



HAL
open science

Machine learning for high-dimensional and structured problems

Romain Hérault

► **To cite this version:**

Romain Hérault. Machine learning for high-dimensional and structured problems. Machine Learning [stat.ML]. Université de Rouen-Normandie, 2020. tel-03104000

HAL Id: tel-03104000

<https://normandie-univ.hal.science/tel-03104000>

Submitted on 8 Jan 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Normandie Université

HABILITATION À DIRIGER DES RECHERCHES

Spécialité Informatique

École Doctorale Mathématiques, Information, Ingénierie des Systèmes

**Machine learning
for high-dimensional and structured problems**
Apprentissage automatique pour les problèmes de grandes dimensions et structurés

Présentée et soutenue par
Romain HÉRAULT

Dirigée par Sébastien ADAM

Habilitation soutenue publiquement le 11 décembre 2020
devant le jury composé de

Mme Florence D'ALCHÉ-BUC	Professeure, LTCI, Télécom Paris	Rapporteuse
M. Michael BIEHL	Professeur, Institut Bernoulli, Université de Groningue	Rapporteur
M. Nicolas THOME	Professeur, Laboratoire CEDRIC, CNAM, Paris	Rapporteur
M. Thierry ARTIÈRES	Professeur, LIS, École Centrale Marseille	Examineur
M. John Aldo LEE	Professeur, MIRO, Université catholique de Louvain	Examineur
M. Ludovic SEIFERT	Professeur, CETAPS, Université Rouen-Normandie	Examineur
M. Stéphane CANU	Professeur, LITIS, INSA de Rouen-Normandie	Examineur
M. Gilles GASSO	Professeur, LITIS, INSA de Rouen-Normandie	Examineur
M. Sébastien ADAM	Professeur, LITIS, Université Rouen-Normandie	Directeur

Introduction

This report intends to be a view on my research and academical activities since my PhD defense in 2007. It can be split into of three parts.

The first part contains a curriculum vitae, a summary of research and teaching activities and a complete list of publications, all gathered in Chapter 1.

The second part presents the theoretical and practical context of my research as well as the contributions made by the engineers, doctoral and post-doctoral students I have supervised and by my own practice. These research works are in the fields of Artificial Intelligence and more specifically Automatic Learning and Deep Learning. These disciplines fall under the section CNU 61 *Génie informatique, automatique et traitement du signal* but also under the section 27 *Informatique*. In the same way, the application domains I have been able to deal with are diverse: Medical Imaging, Vision, Human Movement, Hydrogeology . . . This is why it appeared that a strict breakdown in terms of model or application would not have been relevant.

The chapter 2) is a general introduction to Machine Learning and Deep Learning.

The chapter 3 gathers the theoretical contributions of our work on machine learning problems with high dimension inputs or outputs (typically images or movies) or with a structure (for example graphs). This framework is then illustrated by our work in the fields of Medical Imaging, Vision and Hydrogeology.

The chapter 4 brings together our work on the application field of Human Movement. These tasks are not necessarily part of the high-dimensional or structured problems that are at the heart of this report, but they nevertheless highlight identical questions that our two communities are asking themselves, notably on the role of bias/variance balancing during learning (of the machine or of the athlete).

The manuscript body ends with a third and final part composed of a single chapter, Chapter 5, addressing the perspectives to this report and to my personal journey.

Finally, in Appendix A, are selected articles of which I am co-author and presented in the body of the document.

Contents

1	Activities Digest	7
1.1	Curriculum vitæ	7
1.2	Syntheses	8
1.3	Research activities	10
1.4	Teaching activities	17
1.5	Administrative and collectives activities	20
1.6	Publications, speeches and communications	21
2	Introduction to Machine Learning and Deep Learning	27
2.1	From Artificial Intelligence (AI) to Deep Learning (DL)	29
2.1.1	Machine learning context and frameworks	30
2.1.2	Supervised learning	30
2.1.3	Unsupervised learning	33
2.2	Artificial Neural Network (ANN) for supervised learning	36
2.2.1	Perceptron	36
2.2.2	Multi Layer Perceptron	37
2.2.3	Recurrent Neural Network (RNN)	39
2.3	Auto-Encoder (AE), an ANN for unsupervised learning	41
2.3.1	Auto-Encoder architecture	41
2.3.2	Auto-encoder training	42
2.4	Deep Learning	44
2.4.1	Definition	44
2.4.2	Tips and tricks to avoid gradient problems	45
2.4.3	Convolutional Neural Networks	51
2.5	Deep Generative Models	56
2.5.1	Variational auto-encoders	57
2.5.2	Generative Adversarial Network	57
3	High-dimensional/structured input/output problems	63
3.1	What are high-dimensional or structured problems ?	65
3.1.1	Image labeling / semantic segmentation : an example of high-dimensional problem	65
3.1.2	A broader approach: structured output problems	69
3.1.3	Toward high-dimensional/structured input/output (HD SIO) problems	70
3.2	Solving HD SIO problems using multi-task regularization	71
3.2.1	The Multi-Task Learning setup	71
3.2.2	Examples of sequential learning	74
3.2.3	Examples of concomitant learning	77
3.2.4	Perspectives and undergoing works	78
3.3	Constrained deep generative models	79
3.3.1	Image synthesis/reconstruction with few constraint	79
3.3.2	Polarimetric conversion	81
3.3.3	Sequence prediction	83

4	Machine Learning applied to Human Movement Science	85
4.1	Movement as dynamical system	87
4.1.1	Importance of the variability in Human Movement	88
4.1.2	Human Movement open questions	89
4.1.3	Why use Machine Learning ?	89
4.1.4	Parallels between human training and machine learning	89
4.2	Movement profiling	90
4.2.1	Change point detection	90
4.2.2	Climber performance evaluation	94
4.2.3	Swimming cycle clustering	98
4.3	Perspectives on Machine Learning applied to Human Movement	101
4.4	Gait recognition	102
4.4.1	Context	102
4.4.2	Proposed framework	102
4.4.3	Results and perspectives	103
5	Perspectives and scientific project	105
5.1	Challenges of Machine Learning	105
5.2	Scientific Perspectives	106
5.3	Personal Project	107
A	Selected Publications	123
A.1	IODA: an Input/Output Deep Architecture for image labeling	125
A.2	Spotting L3 slice in CT scans using deep convolutional network and transfer learning	155
A.3	Deep Neural Networks Regularization for Structured Output Prediction	187
A.4	Pixel-wise Conditioned Generative Adversarial Networks for Image Synthesis and Completion	207
A.5	Temporal dynamics of inter-limb coordination in ice climbing revealed through change-point analysis of the geodesic mean of circular data	237
A.6	Comparing Dynamics of Fluency and Inter-Limb Coordination in Climbing Activities Using Multi-Scale Jensen–Shannon Embedding and Clustering	255
A.7	Key point selection and clustering of swimmer coordination through Sparse Fisher-EM	293
A.8	Improved Model-Free Gait Recognition Based on Human Body Part	307

Chapter 1

Activities Digest

1.1 Curriculum vitæ

First name, last name:	Romain HÉRAULT
Birth date:	September 11th 1981 in Nantes (Loire-Atlantique)
Position:	Associate professor (maître de conférences), normal class
Institution:	Institut National des Sciences Appliquées de Rouen-Normandie (INSA de Rouen-Normandie), Normandie Université
Teaching departement:	Informatique et Traitement de l'Information (ITI), Formerly Architecture des Systèmes d'Information (ASI)
Research laboratory:	Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes (LITIS EA 4108), Machine Learning (<i>App</i>) team

Contact information

Romain HÉRAULT
Département ITI
INSA de Rouen-Normandie
685 avenue de l'université BP 08
76801 SAINT-ETIENNE-DU-ROUVRAY Cedex, FRANCE
Tel : +33 (0)2 32 95 98 38 - Fax : +33 (0)2 32 95 97 08
E-Mail : romain.herault@insa-rouen.fr
<http://asi.insa-rouen.fr/enseignants/~rherault>

Education

- 2004 - 2007 PhD in *System and information technology*, obtained under the supervision of Yves GRANDVALET and Franck DAVOINE at Université de Technologie de Compiègne (UTC), defended in November 2007
- 2003 - 2004 Master (DEA) in Signals and Images in Biology and Medicine (SIBM) at Université d'Angers, defended in September 2004,
- 1999 - 2004 Electronic and signal processing engineer from École Supérieure d'Électronique de l'Ouest (ESEO), Angers, defended in September 2004.

Important dates

- Spring 2019 Invited professor at MIRO, Université catholique de Louvain (UCLouvain), Belgium, 6 months sabbatical, Congés pour Recherche ou Reconversion Thématique (CRCT),
- 2016 - Entitled of the Doctoral Supervision and Research Award (PEDR) since October 2016,
- Spring 2015 Invited professor at ICTEAM, UCLouvain, Belgium, 6 months sabbatical, CRCT,

2008 - Associate professor (maître de conférences) in the computer science department, Architecture des Systèmes d'Information (ASI), now Informatique et Traitement de l'Information (ITI), of INSA de Rouen-Normandie, Normandie Université in section 61 since September 2008,

2007 - 2008 Assistant lecturer (ATER) at the computer science departement of UTC,

Spring 2004 Research internship on lossless watermarks at the nuclear medicine departement of the university hospital of Angers,

Summer 2003 Research internship on machine learning at ÉSÉO, Angers, first approach to Machine Learning (SVM, NN),

Summer 2002 Internship on IT security at ACRIE, Nantes,

Summer 2001 Volunteer teacher in an orphanage at Atlixco, Mexico.

1.2 Syntheses

Research activities synthesis

Research topics

Theoretical keywords

• Machine Learning, • Kernel Methods, • Deep Learning, • Adversarial Learning, • High-dimensional or structured data, • Semantic segmentation.

Application fields

• Machine Learning applied to signal processing, • Medical imaging, • Human movement, • Hydro-geology, • Polarimetric imaging.

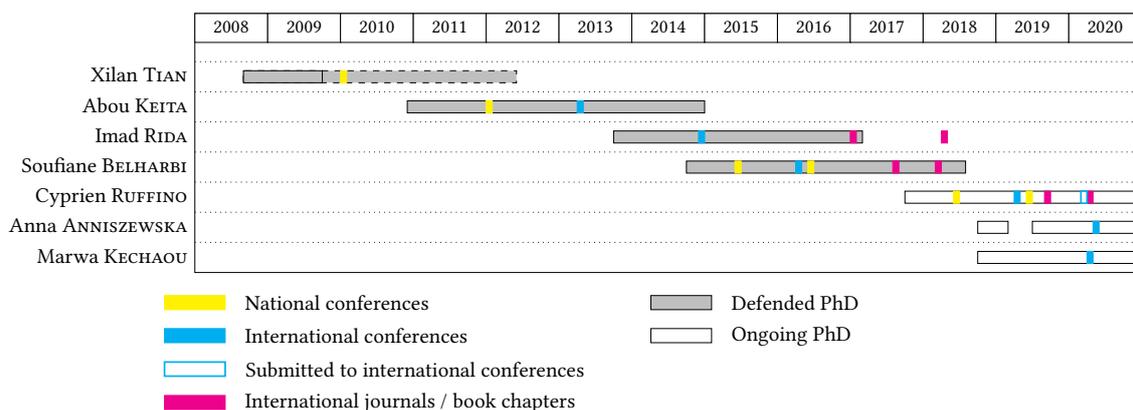
Production

Publication	2006	'07	'08	'09	'10	'11	'12	'13	'14	'15	'16	'17	'18	'19	'20	Total
Awards		2										1				3
Book chapters												1				1
Int. journals						2		3	5	1	1	3	5	1	1	21
Int. conferences	1	1		1	1			1	1	1	1			1		10
Nat. journals			1													1
Nat. conferences	1	1			1		1			1	1		1	1		8

Supervision

- Master thesis: 6
- PhD: 7 (4 defended)
- Postdoc/Engineer: 8
- PhD Jury: 2

PhD details



Other academical activities synthesis

- Formerly managing guest editor (2017-2018) and now associate editor (since 2019) to Neurocomputing, Elsevier,
- Reviewer for more than 20 journals and conferences,
- Member of the local organizing committee for 2 national conferences (RFIA 2014, CAp 2018),
- Coordinator of a local conference (JSecIN),
- Volunteer for two international conferences ICML 2007 and ESANN 2015/2019.
- 4 invited plenary talks,
- 2 times six months visits to abroad university (UCLouvain),
- Selection committee for hiring assistant professors,
- Elected board members of INSA de Rouen-Normandie (since 2010).

Project / Industrial contract / Expertise activities synthesis

- Coordinator of an ongoing regional project (DeepART),
- Member of 6 ongoing projects, local supervisor of one of the national projects,
- Member of 6 completed projects,
- Academical supervisor for 3 industrial contracts,
- Expert for
 - ANR, Agence Nationale de la Recherche ;
 - ANRT, Association Nationale de la Recherche et de la Technologie.

Teaching and administrative activities synthesis

Teaching duties

- Supervisor for 3 ongoing courses,
- Active teacher in 5 other courses,
- Set-up 5 new courses from scratch,
- Author of a MOOC,
- Student internship/project evaluations

Administrative duties

- As a member of the school board:
 - Member of the teacher/employee administrative court (since 2015),
 - Member of the budget commission (from 2015 to 2018).
 - Member of the student administrative court (from 2010 to 2015), 5 cases,
- Supervisor of a Specialized Mastère (BAC+6) since 2019,
- Supervisor of *Contrat de professionnalisation* at the computer science department since 2015,
- Supervisor of the evaluation of *Projets INSA Certifiés iso 9001*,
- Member of the computer science department jury,
- Time schedule for computer science department before 2014,
- Digital correspondent at the regional academical community before 2012.

1.3 Research activities

Activities before LITIS

I started research activities during my two master internships.

The first master internship occurred in the ESEO lab. I had to predict which patients will faint when they get up according to their respiratory impedance and electrocardiogram signals recorded when they are lying. It was my first approach to machine learning using wavelets as feature extraction, and support vector machines (SVM) or neural networks (NN) as classifiers.

The second one took place at the nuclear medicine departement of the university hospital of Angers. During this internship, I implemented a lossless watermark method dedicated to medical images such as MRI. The purpose was to detect where an image could be intentionally damaged. It was my first contact to medical imaging. At that time, I wanted to further develop this technique in a PhD inside the lab that hosted me but the funding of the project was rejected.

Nevertheless, I missed machine learning so I applied to a open PhD position at HEUDIASYC lab at Université de Technologie de Compiègne (UTC) on the detection of driver drowsiness. The first two years were dedicated to features extractions on videos of drivers: We have developed an appearance model that enables the tracking of the head and the inner motions of the face [Hér+06; HDG06]. The last year of the PhD funding and the year as assistant lecturer were spent into building a sparse probabilistic classifier [HG07a; HG07b] that could provide fine grain estimation of the probability of a driver falling asleep around the decision threshold but that could be less precise near 0 or 1.

Research context at LITIS

After one year of assitant lecturer position at UTC, I was hired as an associate professor (Maître de Conférence) in September 2008 at INSA de Rouen, performing the research works at the *Laboratoire d'informatique, de traitement de l'information et des systèmes* (LITIS).

LITIS is one of the two main public computer science labs in the new Normandy region; formerly, the main one in the past region *Haute-Normandie*. It is the fusion of 4 laboratories that occurred in 2006, and is composed of 7 thematic teams.

I belong to the *Apprentissage* team which is dedicated to Machine Learning. Moreover, I have strong collaborations with the *Quantification en imagerie fonctionnelle* (QuantIF), *Multi-agents, Interaction, Décision* (MIND) and *Systèmes de transport intelligent* (STI) teams whose main research subjects are respectively Medical Imaging, Autonomous Agents and Intelligent Vehicules.

LITIS is a medium regional laboratory but with strong industrial local collaborations and international academic partnerships. It hosts around 100 full time position researchers and 70 PhD students.

The lab has the particularity to be part of 3 different entities: Universty of Rouen, University of Le Havre and INSA Rouen. They all belong to the regional university community, the *Normandie Université*, which has in charge the management of PhD students, among other purposes. A LITIS member is part of only one of this entities whatever the research teams he belongs to. The side effects are that within a research team financial and administrative rules may be different from one body to the other.

With the current trend on data science, the machine learning team has few day to day financial problem. Indeed, industrial partnerships enable us to propose every year master and PhD grounds as well as postdoc/Engineer positions. Nevertheless, the long term academical research suffers from the multiplication of short term projects with no guarantees on more theoretical / non-application research or PhD position. Moreover, master, PhD students or postdoc are more and more keen to work on private sectors due to unclear situation of public researches. Thus, ensuring good quality hiring and human resource management is more and more difficult.

Research collaboration within LITIS

When I arrived in Rouen, the *Apprentissage* team used to be composed of two clusters: researchers coming from handwritten document recognition field and researchers coming from theory of machine learning. This distinction faded away along the 10 years spent at this lab. On my side, I had collaboration with both thematics:

- robust statistics, kernel methods, non-convex optimization, domain adaptation and optimal transport with the professors Stéphane CANU, Gilles GASSO, and Dominique FOURDRINIER [Tia+10; KHC12; Kei+13; RHG14; Rid+17; Rid+18; Ruf+18; Ruf+19b; Ruf+19a; Lal+19; Ruf+20; Ani+20; Kec+20; Bli+],

- neural networks with Benoit GAÛZÈRE, Clément CHÂTELAIN and the professor Sébastien ADAM [LHC09; Bel+15a; Bel+15b; Ler+15; Bel+16a; Bel+16b; Bel+17; Amy+18; Bel+18].

Marginally, I have some collaborations with Simon BERNARD and Pierre HÉROUX from the ML Team on mixed optimization problem. The collaboration within the ML team mostly aims at providing general machine learning framework that could be used on different application fields.

My collaborations with members of other teams are linked to different possible applications of machine learning:

- Segmentation and Detection on Medical Imaging with Romain MODZELEWSKI, Isabelle GARDIN as well as professors Sébastien THUREAU and Pierre VERA from the QuantIF team / Centre Henri Becquerel [Ler+15; Bel+17; Amy+18]; I'm supervising the *DEEP learning in Adaptive Radiation Therapy* (DEEPART) project (cf Section 3.3.3),
- Polarimetric image processing and generation with Samia AINOUCZ from the STI team [Bli+],
- Social network analysis and conversational agent with Alexandre PAUCHET from the MIND team.

These works are highlighted in the following project and academical supervision sections and more precisely described in chapters 3 and 4. Selected publications are available in [Ler+15] (Appendix A.1), [Bel+17] (Appendix A.2) and [Bel+18] (Appendix A.3).

External Collaborations

I'm involved into three main collaborations outside of LITIS. One locally at the université de Rouen with CETAPS concerning Machine Learning applied to Human Movement Science, two abroad in Belgium with MLG/MIRO UCLouvain and SCKCEN concerning Machine Learning applied to respectively Medical Imaging and Geosciences.

Let's note that I was given two *congés pour recherche ou conversion thématique* (CRCT), i.e. sabbatical visits, of one semester each which enabled me to consolidate my collaboration with UCLouvain.

CETAPS at Université de Rouen

CETAPS is a laboratory of Université de Rouen whose researches are dedicated to the studies of physical and sports activities. Most notably, one of its aims is to understand how performance and efficiency emerge from training.

Academics at CETAPS and most notably Pr. Ludovic SEIFERT were used to applied statistics and modeling techniques but they wanted to investigate how Machine Learning was good to mitigate recurring problems in Sport Science and notably Human Movement Science such as study of inter- and intra- individual variability, before, during and after expertise acquisition.

The chapter 4 is dedicated to this collaboration and the consecutive published works [Sei+10b; Sei+10c; Sei+11b; Sei+11a; Ort+13; Sei+13b; BHS13; Sei+13a; Sei+13c; Dov+14; Sei+14a; KHS14; Cho+14b; Sei+14c; Sei+14b; Hér+15; Sei+15; Bou+16; Hér+17; Sei+18]. One can find attached to this thesis the publications [Sei+13a] (Appendix A.5), [Hér+17] (Appendix A.6) and [KHS14] (Appendix A.7).

MIRO/MLG at UCLouvain, Belgium

John LEE is my main collaborator at Université catholique de Louvain (UCLouvain). He is a Professor, maître de recherche at the Belgian F.R.S.-FNRS (Fonds National de la Recherche Scientifique). He is at the head of the UCL/IREC/MIRO laboratory.

According to John, The Molecular Imaging, Radiotherapy and Oncology (MIRO) laboratory is a research group where multiple disciplines meet and cross-fertilize. MIRO gathers physicians, physicists, chemists, and engineers. It includes several facilities (radiochemistry, radio-biology, small animal housing) and several preclinical imaging rooms for small animals (PET, SPECT/CT) as well as an irradiator. In addition to its preclinical imaging platform, MIRO is in direct connection with the departments of radiology, nuclear medicine, and radiotherapy in the St Luc university hospital, with access to modern devices for clinical imaging and radiation therapy (4D-CT simulator, MRI, CT and PET-CT with dedicated slots for radiotherapy). MIRO also collaborates tightly with engineers from UCL/SST/ICTEAM (signal image processing in the engineering school), such as the UCL Machine Learning Group (MLG).

This Machine Learning Group gathers researchers from the Applied Mathematics, Computing Science and Engineering, Information Systems and Electrical Engineering departements of UCL. It was founded in 2003 by Pr. Michel VERLEYSEN. More than 15 academics and 10 PhD students belong to it.

In 2015 and 2019, I was granted six months stays at MLG and MIRO by the french CNU (Conseil National des Universités) respectively at the invitation of Michel VERLEYSEN, now dean of the engineering

school of Louvain, and John LEE. Two of my PhD students, Soufiane BELHARBI and Cyprien RUFFINO, were also granted to come for shorter stays in Belgium (≈ 1 month).

The configuration of MLG/MIRO teams for UCL is the same as Apprentissage/QuantIF teams for LITIS where internal exchange arises from the need of machine learning applied to medical images. Swing links between our two institutions already exist with strong collaboration between QuantIF and MIRO inside the *Canceropôle Nord-Ouest*.

Our collaboration has resulted to the following publications [Hér+15; Lal+17] and the attached publication [Hér+17] (Appendix A.6). These works are summed up in Chapters 3 and 4.

SCKCEN, Belgium

SCKCEN stands for *StudieCentrum voor Kernenergie / Centre d'Étude de l'énergie Nucléaire*. It is a research foundation under the supervision of the Belgian Federal Ministry in charge of energy whose main activities aim at Nuclear physics and most notably: nuclear safety, medical and industrial applications of radiation, nuclear reprocessing and management of radioactive waste and Nuclear as well as decommissioning and decontamination of nuclear sites.

Eric LALOY is a researcher at SCKCEN where he is specialized in engineered and geosystems analysis. His work is partially dedicated to inverse modeling for subsurface hydrology and other Earth science disciplines.

John LEE introduced both Eric and I when Eric was attending a Machine Learning lecture of John at UCLouvain. Following this encounter, he has been invited to LITIS lab for 1 week. Eric wanted to try to compare Multiple Point Statistics (MPS) and Machine Learning techniques in the resolution of inverse modeling. We worked together on Auto-Encoders and Generative Adversarial Networks to generate plausible subsurface image under constraints (Chapter 3 Section 3.3). This has led to multiple publications [Lal+17; Lal+18; Lal+19; Ruf+18; Ruf+19a; Ruf+19b; Ruf+20] and a PhD ground (Cyprien RUFFINO) with an international journal publication [Ruf+20] (Appendix A.4).

Projects

Since my beginning at LITIS, I was involved in more than 12 research projects. 6 are still active and I lead one of them.

Ongoing

Name	Type	Start	Length (month)	Budget LITIS/Total
NePTUNE <i>Natation et Paranatation : Tous Unis pour Nos Élites</i> Very high performance sports project (JO 2024) Working packages 2) Tracking and pacing strategy and 3) Coordination, Propulsion and Energy	ANR/STHP	2020	48	70k€/1.56M€
MinMacs/DeepART <i>DEEP learning in Adaptive Radiation Therapy</i> Supervisor of DeepART in the MINMACS program	Label ex. région	2019	36	112k€/224k€
SAPhIRS Système pour l'Analyse de la Propagation d'Information dans les Réseaux Sociaux	DGA-DGE	2017	36	247k€/570k€
DAISI <i>Data science : methodology and applications</i>	GRR/FEDER	2017	48	925k€
DynACEV <i>Dynamics of Learning: Behaviour and Lived Experiences. The role of exploratory strategies</i> Supervisor of WP 5: Analysis of the dynamics of learning	ANR	2017	48	126k€/232k€
Deep in France <i>Machine learning with deep neural networks</i>	ANR	2017	42	124k€/811k€

Completed

Name	Type	Start	Lentgh (months)	Budget
ASAP <i>Learning in deep architecture</i>	ANR	2009	48	
GEN-EASE <i>Study of a continuous biological monitoring and analysis system</i>	ANR	2009	48	
LEMON <i>Learning with Multi-objective Optimization</i> Coordinator of the task « Applications »	ANR JCJC	2011	48	
XTERM <i>Systèmes compleXes, intelligence TERritoriale et Mobilité</i>	GRR/FEDER	2015	48	509k€
NARECA <i>Narrative Embodied Conversational Agent</i>	ANR CONTINT	2013	36	630k€
DYNAMOV <i>Temporal Dynamics of Movement Patterns</i> Supervisor of WP 4 « Movement patterns coordination dynamics and temporal dynamics of learning modelling »	ANR JCJC	2013	48	180k€

Industrial contracts

I have undertaken industrial contracts for more than 50k€:

- Scientific manager of contracts with CILAS and Bertin; Conducting feasibility studies and prototypes ;
- Conduct of feasibility studies for Dynamease company.

Since 2017, we have started a collaboration with the Luxscan company that has eventually led to a PhD thesis financed by them and the Luxemburg state.

Supervision

Master Thesis

- Gautier BIDEAULT, co-supervision 80% avec Ludovic SEIFERT 20%, from February to September 2011. Title : *Modélisation des mouvements de bassin des nageurs de haut niveau en crawl*. This work has led to 2 journal publications [BHS13; Sei+13c].
- Julien LEROUGE, co-supervision 50% Clément CHATELAIN 50%, from March to August 2013. Title : *Segmentation de tumeur de l'œsophage sur des images TEP 18 FDG par des techniques d'apprentissage profond*. The journal publication [Ler+15] (Appendix A.1) comes from this internship work.
- Nar DIOP, co-supervision 80% avec Ludovic SEIFERT 20%, from April to July 2014. Title : *Apprentissage conjoint du dictionnaire de classes supervisées sur des données d'orientation*,
- Houssain ABDESSALEM, supervision 100%, from March to September 2018. Title : *Default detection in wood plank using Convolutional Neural Networks*,
- Robin CONDAT, supervision 100%, from March to September 2018. Title : *Default detection in wood plank using Recurrent Neural Networks*,
- Gaétan BAERT, supervision 100%, from March to September 2019. Title : *Default detection in wood plank using Neural Networks*. Following Robin and Houssain internships. Gaétan is now a research engineer at LITIS.

PhD Supervision

Defended PhDs

Xilan TIAN

PhD defended on May 7th 2012 under the title *Apprentissage et noyau pour les Interfaces Cerveau-machine*.

Supervision: Gilles GASSO (50%) and Stéphane CANU (50%) from September 2008 to August 2012. I had replaced Gilles supervision during his one year stay in NEC Laboratories America, Inc., USA, starting in September 2008.

Publications as co-author: 1 national conference [Tia+10].

Xilan has been granted a Chinese scholarship of 4 years to achieve her PhD. During her first PhD year, we worked together on neural networks pre-training [Tia+10]. Later with Gilles and Stéphane, she focused on non-convex optimization problems [TGC12]. She is now a research engineer at China Electronics Group Corporation.

Abou KEITA

PhD defended on December 15th 2014 under the title *Modèles statistiques précoces et robustes pour l'estimation de la concentration d'agents biologiques dans un système de surveillance en continu dans l'environnement*

Supervision: 50% with Stéphane CANU 50%, from December 2010 to August 2014.

Publications: 1 national conference [KHC12], 1 international conference [Kei+13].

The PhD of Abou was granted by the GEN-EASE ANR project. During this project we aimed to build an automatic alert system for biological hazard. It was a joint initiative of Bertin company, CEA and LITIS. The work of Abou consisted in detecting as soon as possible a change point in a recorded signal of Polymerase Chain Reaction (PCR) to launch an alarm, then to have a precise modelization of the signal curve in order to estimate the biological agent concentration. To do so, we developed robust models, they were published in [KHC12; Kei+13]. Abou is now research engineer at Gesika-LVH Médical.

Imad RIDA

PhD defended on February 3rd 2017 under the title *Temporal Signal Classification*.

Supervision: 35% with Gilles GASSO 65%, from October 2013 to February 2017.

Publications: 1 international conference [RHG14], 1 international journal [Rid+18], 1 book chapter [Rid+17].

Imad was funded by a FUI grant. His work relates to individual identification from biological signals. This encompasses for example palm print recognition or gait classification. With him we had developed classifiers based on sparse matrix decomposition which is described in Chapter 4 Section 4.4 and in the attached publication [Rid+17] (Appendix A.8). He is currently an associate professor (maître de conférences) at Université de Technologie de Compiègne.

Soufiane BELHARBI

PhD defended on July 6th 2018 under the title *Neural networks regularization through representation learning*.

Supervision: 30% with Clément CHATELAIN 30% and Sébastien ADAM 40% (Director), from October 2014 to July 2018.

Publications: 2 national conferences [Bel+15a; Bel+16b], 1 international workshop [Bel+15b], 1 international conference [Bel+16a], 2 international journals [Bel+18; Bel+17].

Soufiane got a state ground (MESR) for realizing his PhD. With him, we worked on Deep Neural Network to solve High-Dimensional and structured Input/Output Problem. These studies are depicted in Chapter 3 and in the attached publications [Bel+18] (Appendix A.3) and [Bel+17] (Appendix A.2). Soufiane is now post-doc at Laboratoire d'imagerie, de vision et d'intelligence artificielle (LIVIA lab) of École de technologie supérieure de Montréal.

Ongoing PhDs

Cyprien RUFFINO

Starting: October 2017 (MESR ground)

Proposed title : *Modèles génératifs de données structurées de type séquences ou tenseurs de grande dimension basés sur une stratégie de réseaux adverses*

Supervision: 50%, Gilles GASSO 50%

Publication: 2 national conferences [Ruf+19b; Ruf+18], 1 international conference [Ruf+19a], 2 international journals [Lal+19; Ruf+20], 1 submitted conference [Bli+].

Cyprien's studies constrained generative models for image synthesis and yet have been very productive. His works is depicted in Chapter 3 Section 3.3 and in the attached publication [Ruf+20] (Appendix A.4).

Anna ANISZEWSKA-STĘPIEŃ

Starting: October 2018 (ANR project DynACEV)

Proposed title: *Clustering et segmentation joints en utilisant la décomposition de tenseurs pour la découverte de comportements humains*

Supervision: 35%, with Gilles GASSO and Ludovic SEIFERT.

Publications: Her preliminary work on HMM has been accepted to MLSA 2020 [Ani+20].

Anna currently works on the automatic annotation of climber video. From a wall equipped of sensors, we know when and where a climber has touched an hold but not with which limb (hands or feet, left or right). We are investigating how to recover this information from the combination of video and sensors recording with HMM and RNN.

Marwa KECHAOU

Starting: November 2018 (Industrial contract with Luxscan/Weining company, financial support from the Luxembourg state)

Proposed title: *Machine Learning for wood defects segmentation and classification*

Supervision: 50%, Gilles GASSO 50%

Publications: A conference paper on Domain adaptation through Optimal Transport has been accepted to ECML-PKDD 2020 [Kec+20].

Marwa studies domain adaptation for defects segmentation and classification. She works 2/3 of her time at the Luxscan compagny in Luxembourg and 1/3 at LITIS.

Research Engineer / Post-doc / Assistant lecturer

- Vlad DOVGALECS, post-doc from September 2013 to August 2014, on the project DYNAMOV, Vlad is nowadays an employee at Oracle ;
- Julien DELPORTE, assistant lecturer (ATER) from September 2014 to august 2015, on the project NARECA ;
- Grégoire MESNIL, post-doc from January 2015 to August 2015, on the project LEMON, he is now CTO of the INCALIA company ;
- Adam SCHMIDT, post-doc from January to December 2016, on the project DYNAMOV and FEDER, Adam works as Senior Project Manager at TNO ;
- Omar RIHAWI, research engineer from June 2017 to January 2018 on the project DynACEV ;
- Imen TRABELSI, post-doc from February 2018 to January 2020, on the project DAISI,
- Gaétan BAERT, research engineer since September 2019 on the project Saphirs,
- Nikolaos ADALOGLOU, research engineer since MArch 2020 on the project DeepART.

Member of PhD Jury

- Perrine BRETIGNY, *L'adaptabilité comme critère d'expertise d'une habileté motrice face aux contraintes : le shoot en hockey sur gazon*, at CETAPS / University of Rouen, June 5th 2013, supervisors Didier CHOLET and Ludovic SEIFERT,

- Dimitri DE SMET D'OLBECKE, *Hybrid Models to Predict Recreational Runners Performance*, at UCLouvain. October 30, 2019, supervisor Michel VERLEYSEN
Member of the jury of the private defense (August 29, 2019) and the public defense (October 30, 2019). In Belgium, at the CIL doctoral school (Brussels, Louvain, Liège, ...), there is no *rapporteur* (reviewer). It is up to the jury of the private defense to study the manuscript in detail beforehand, to question the candidate during a long defense (3 hours) in the sole presence of the candidate, and then to authorize or not the public defense a few months later.

Organization of the research

Locally, I'm an elected member of the school board (Conseil d'Administration) since November 2010. The school board plays an important role in promoting research for local labs, like funding PhD thesis. Moreover we are involved in the selection of local scientific projects and academic promotions.

I have been involved in the organization of the following conferences: • ICML 2007 (as a volunteer), • RFIA 2014, • ESANN 2015, 2019 (as a volunteer), • CAP 2018.

Moreover, every two years, the University of Rouen, the University of Caen and the INSA of Rouen organize the JSecIn *Journée de la Sécurité Informatique en Normandie*¹ which is a regional meeting dedicated to IT security. I'm the INSA supervisor for this event since the 2018 edition.

Expertise

I have been expert/reviewer/assessor for the following organizations : • ANR, Agence Nationale de la Recherche ; • ANRT, Association Nationale de la Recherche et de la Technologie.

Editorial activities

Managing guest editor

I was a managing guest editor of the special issue of Neurocomputing, Elsevier, on the ESANN conferences 2015 and 2017 [Aio+16; ABH17]. This work consists in managing corresponding/associate editors and reviewers.

Associate editor

Following the managing guest editor experience, since January 2019 I'm a full associate editor to Neurocomputing, Elsevier, ISSN: 0925-2312². Here is some bibliometric information of this journal:

CiteScore	5.00
Impact Factor	4.072
5-Year Impact Factor	3.824
Source Normalized Impact per Paper (SNIP)	1.779
SCImago Journal Rank (SJR)	0.996

Reviewer

Since my PhD years, I've been reviewer for the following publications:

Journals

- AIIM, Artificial Intelligence in Medicine, Elsevier, ISSN: 0933-3657,
- AWR, Advances in Water Resources, Elsevier, ISSN: 0309-1708,
- CAGEO, Computers & Geosciences, Elsevier, ISSN: 0098-3004,
- CJAS, Journal of Applied Statistics, Taylor & Francis, ISSN: 0266-4763,
- ESWA, Expert Systems with Applications, Elsevier, ISSN: 0957-4174,
- JMLR, Journal of Machine Learning Research, Microtome, ISSN 1532-4435 / 1533-7928,
- MedPhys, Medical Physics, AAPM, ISSN:2473-4209,
- NEUCOM, Neurocomputing, Elsevier, ISSN: 0925-2312,
- PRL, Pattern Recognition Letters, Elsevier, ISSN: 0167-8655,
- WRR, Water Resources Research, ISSN: 1944-7973.

¹<http://jsecin.insa-rouen.fr/>

²<https://www.journals.elsevier.com/neurocomputing/>

Conferences

- DAMI/DMKD, Data Mining and Knowledge Discovery,
- ECML/PKDD, European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases,
- ESANN, European Symposium on Artificial Neural Network,
- EUSIPCO, European Signal Processing Conference,
- GRETSI, Groupe d'Etudes du Traitement du Signal et des Images,
- ICDAR, International Conference on Document Analysis and Recognition,
- ICLR, International Conference on Learning Representations,
- ICML, International Conference on Machine Learning,
- IJCAI, International Joint Conference on Artificial Intelligence,
- MLSP, Machine Learning for Signal Processing,
- NIPS/NeurIPS, Conference and Workshop on Neural Information Processing Systems,
- RFIA, Reconnaissance de Formes et l'Intelligence Artificielle.

1.4 Teaching activities

Part-time position at Université de technologie de Compiègne

Before being an associate professor at INSA, I taught during the 3 years of my PhD preparation and 1 year as lecturer at the *Université de technologie de Compiègne* in the computer science department, *Génie informatique*. I mainly took part in theoretical exercises, practical works and evaluations.

The teaching were the following:

- Initiation to 3D programming (OpenGL),
- Initiation to computer architecture and assembly programming,
- Object-oriented programming (C++/Java),
- Logical programming (Prolog).

This experience confirm my desire for teaching. That is why I asked for the qualification for being a Maître de conférences (assistant professor) for two sections dedicated to computer science, signal processing and machine learning: 27 *Informatique* and 61 *Génie informatique, automatique et traitement du signal* which I both obtained. Thus, I was able to apply for a full time position at a french university which I eventually got on September 2008 at the *Institut National des Sciences Appliquées de Rouen-Normandie* in short INSA Rouen-Normandie.

Full-time position at INSA Rouen-Normandie

I was hired in the INSA Rouen-Normandie to give teaching mainly at the computer science departement by the time called *Architecture des Systèmes d'Information (ASI)* which is now *Information et Technologie de l'Information (ITI)*. Some side teaching activities occur at the department *Science et Technique Pour l'Ingénieur (STPI)* of INSA and a the University of Rouen.

In the following section part of the courses description is extracted from the official INSA repository. Moreover, to be in accordance with the official naming, the course titles are kept in French.

ITI department

At the computer department, I belong to the data science team.

Most of my teaching duty is connected to signal processing or machine learning where I'm doing lectures (*Cours Magistraux*, CM), theoretical exercises (*Travaux Dirigés*, TD) and practical work (*Travaux pratiques*, TP) for the three last years of the 5 years engineering training. These years are

- ITI3, formerly ASI3, corresponding to Licence 3 (L3) in the French university system,
- ITI4, formerly ASI4, corresponding to Master 1 (M1), and ultimately,
- ITI5, formerly ASI5, corresponding to Master 2 (M2).

The main courses which I have set-up and supervised are

DM2 *Méthodes de Fouilles de Données et d'Apprentissage*, ASI4/5, M1/2, from 2008 to 2013,

This course introduced Kernel Methods, Bayesian networks and Neural networks to students.

TSS *Traitement Statistique du Signal*, ASI4, M1 from 2008 to 2016,

Optimal filtering was the main subject of this course.

TSA *Traitement des signaux aléatoires*, ASI3, L3 starting 2018,
This course is dedicated to basic processing of stochastic signals (Auto-regressive modelization, Kalman filter, . . .).

EDTS *Estimation et Décision Statistique en Traitement du Signal*, ASI5 M2 starting 2014,
Markovian process is the main thematic of this course: Hidden Markov Model, Advanced Kalman Filtering, Particle filtering. A part of the lecture gives also some clues on change point/novelty detection.

DEEP *Deep Learning*, ASI5 , M2 starting 2018 ,
This is a full lecture on neural networks, from perceptron to recurrent and deep network. Recently generative models were also added to the skills presented.

Currently, I'm still supervising **TSA**, **EDTS** and **DEEP**.

I'm also deeply involved in the following course

PIC *Projets INSA Certifiés iso 9001*, ASI4/5, M1/2 starting 2009,
This is not a traditional course: students play a developer team (8 students) for a full year half time of their attendance at school. We introduce them to good developing practice such as documentation, continuous integration, traceability, agile programming. They have to follow a quality reference such as *iso 9001*. The quality management of the team is certified by an external auditor. In this course, I take the role of a technical advisor for one of the 6 teams. In parallel, I'm the person in charge of the evaluation of all the teams.

The side courses where I help other teachers in theoretical exercises or practical work are

Réseaux *Réseaux informatiques*, ASI4, M1,
Theoretical and practical skills in computer networks are presented here.

Algo *Algorithmique avancée et programmation C*, ASI3, L3,
The goal of this course is to study dynamic data structures and advanced algorithms.

STPI department

The *Science et Technique Pour l'Ingénieur (STPI)* correspond to the first cycle after the high school and before the engineer training It takes place in two years. In this cycle, students learn the base of a general engineer background whatever the pathway they will choose whether they will go to chemistry, mechanics, . . . or computer sciences.

These years are

- STPI1, corresponding to Licence 1 (L1) in the French university system,
- STPI2, corresponding to Licence 2 (L2).

In this cycle, I assist other professors mainly in programming initiation:

I1 *Initiation à la programmation impérative*, STPI1, L1,
The goal of the course is to discover imperative programming. The basic concepts of any imperative programming language are seen as well as compiling tools. The Pascal language is used for practical works.

I2 *Algorithmique et programmation structurée*, STPI1, L1,
Following this course a student will be able to write a computer program from the problem statement to its implementation in Pascal, going through algorithm designs.

Let's note that after the retirement of two colleagues and before the hiring of a new colleague at STPI, I was the supervisor of both the above courses for 2 years. STPI is a huge structure and each course enrolls about 350 students, so supervising one course means managing 3 full-time professors for the main lectures as well as 10 to 12 assistants or lecturers for the practical works.

University of Rouen

I was also enrolled at the neighbor university of Rouen for lectures especially in the data science / machine learning thematic.

Master SID

The master *Science et Ingénierie des Données* SID is one of the first french masters to be officially specialized in data science.

Some courses from ITI/INSA are shared with this master. Namely, the **EDTS** and **DEEP** courses described above received students from both training. Practically, lectures are in common and programming session but evaluation are apart.

Master EOPS

The master *Entraînement et Optimisation de la Performance Sportive* (EOPS) ambitions to form national to international sport coaches. Following our research partnership with CETAPS and the increasing trend in sport competition to analyse and interpret past recorded data, they asked me to give an introduction to data analysis to their students.

Ultimately since 3 years, we are opened a joint training on *Game analysis and big data* with the master SID, where advanced students from EOPS can assist lectures at master SID and vice versa.

Outside Rouen campus

Besides INSA and university of Rouen I was implied in activities of other educational structures.

OpenClassrooms MOOC

OpenClassrooms is a private company which proposed an online education platform, a French speaking equivalent to Coursera. The *OpenINSA* initiative, which is supported by the French education ministry and the INSA group (all the INSA schools in France), aims at providing the *OpenClassrooms* platform with high level education contents in the form of massive open online courses (MOOC). With my colleague and friend Clément CHATELAIN, we have set up the *Initiez-vous au Deep Learning* course which consists in a 8 hours lecture on Deep Learning with exercises and evaluation procedure. By the time of writing, 4 200 students were enrolled in it³.

UCLouvain

During my second stay at UCLouvain in spring 2019, I have set-up and teach a 3 day course on neural networks and deep learning [Hér19b]. It took part of the local master curriculum and of the CIL doctoral school training which is doctoral school dedicated to Machine Learning gathering the universities of Brussels and the southern part of the Belgian country.

Other education duties

Mastère ESD

Since 2019, I'm the co-supervisor of the specialized mastère *Expert en science des données* (ESD).⁴

Specialized mastères are Bac + 6 training specific to grandes écoles in connection with a company and a laboratory. The ESD specialized master's training is done through face-to-face courses, an advanced research project (he same kind as a master's thesis in Anglo-Saxon countries), and a work-study program.

I'm in charge of being the principal communication point for students, of managing about 15 teachers from different departments inside and outside INSA, of working with the INSA administration for hiring new students and certifying the training.

Contrat de professionnalisation

Since 2015, I'm the supervisor at the ITI department of students which follows the INSA training under a *Contrat de professionnalisation*. It is a special status where the student is actually an employee of a private company while receiving education program at INSA. Each week, he/she spends 3 days at school and works 2 days at the company. My task goals are to be the interface between the company and the schools, managing contractualization process, evaluating and advising the company on student job.

³<https://openclassrooms.com/fr/courses/5801891-initiez-vous-au-deep-learning>

⁴<https://www.insa-rouen.fr/formation/masteres-specialisesr/expert-en-sciences-des-donnees>

ITI Jury

Since my first years at the computer science department, I'm involved in its jury. It takes place four times a year, two sessions for each semester. Students who are found in difficulties get an audition. The jury then decides whether or not to validate their credits and let them pass the year.

Time schedule

From 2010 to 2014, I was in charge of establishing the time schedule of the computer science department. It consisted in gathering constraints from teachers and from the administration to build the time tables. Actually, it was an intensive task during 2 weeks at the beginning of each semester.

UNR correspondant

L'*Université Numérique en Région* (UNR) was an initiative of the proto *Normandie Université* before its creation. The goal of this project was to propose to the involved institutions a common education web platform. For 3 years, I was the INSA correspondant for the educative section. We set up a *Environnement Numérique de Travail* (ENT), i.e a common website gathering communication and education facilities such as webmail, file sharing, personal schedule. Since then it has been superseded by tools provided by CRIANN or Renater.

Teaching hour duties

The following table indicates the services performed over the last 4 academic years plus a forecast for 2019-2020 (no yet closed). The hours are to be heard in TD (exercice sessions) equivalent and do not include internship follow-ups, nor the reference hours as the responsibilities to the direction of studies of the ITI department such as *Contrat de professionnalisation* or *Mastère ESD*. It should also be noted that for the period 2018-2019 I was given a CRCT of one semester

Years	Teaching hours
2015-2016	199h
2016-2017	210,25h
2017-2018	234,37h
2018-2019	125,5h
2019-2020	281,9h

Including the reference hours, my service equals roughly 290h each years so more than 100h above the legal duty.

1.5 Administrative and collectives activities

Board member

Since November 2010, I'm an elected member of the *Conseil d'Administration* (CA), i.e. the board of the school. There are 4 plenary sessions a year and around 8 to 10 academic sessions a year. During the plenary sessions, our main mission is to define the general policy of the school which is then applied by the direction and its budget. INSA Rouen-Normandy has decided to acquire the *Compétences élargies* in early 2010, which means among other things that the board and direction are also responsible for the wage budget. During the academic sessions, our work is mainly focused on human resources such as promotion.

As a board member, I also endorse the following duties:

- student administrative court from November 2010 to November 2015, 5 cases,
- employee administrative court since November 2015,
- budget commission from November 2015 to November 2018.

Other duties

I have the qualification of first-aid rescue worker, i.e. *sauveteur secouriste du travail*.

1.6 Publications, speeches and communications

Awards

- [BR17] Centre Henri Becquerel and INSA de Rouen. “BodyComp.AI : L’utilisation de l’intelligence Artificielle En Imagerie Médicale, Prix Unicancer de l’innovation 2017, Prix de l’organisation et Des Métiers de La Recherche.” 2017.
- [HG07a] Romain Hérault and Yves Grandvalet. “Régression Logistique Parcimonieuse.” In: *Conférence Sur l’Apprentissage Automatique (CAp)* (Grenoble, France, France). Ed. by Cépaduès. Grenoble, France, July 2007, pp. 265–280. URL: <https://hal.archives-ouvertes.fr/hal-00442755>.
- [Hér+06] Romain Hérault, Franck Davoine, Fadi Dornaika, and Yves Grandvalet. “Suivis Simultanés et Robustes de Visages et de Gestes Faciaux.” In: *Reconnaissance Des Formes et l’Intelligence Artificielle (RFIA)* (Tours, France, France). Tour, France, Jan. 2006. URL: <https://hal.archives-ouvertes.fr/hal-00442758>.

International audience

Book chapters

- [Rid+17] Imad Rida, Noor Al Maadeed, Gian Luca Marcialis, Ahmed Bouridane, Romain Hérault, and Gilles Gasso. “Improved Model-Free Gait Recognition Based on Human Body Part.” In: *Biometric Security and Privacy: Opportunities & Challenges in The Big Data Era*. Ed. by Richard Jiang, Somaya Al-maadeed, Ahmed Bouridane, Prof. Danny Crookes, and Azeddine Beghdadi. Signal Processing for Security Technologies. Cham: Springer International Publishing, 2017, pp. 141–161. ISBN: 978-3-319-47301-7. DOI: 10.1007/978-3-319-47301-7_6. URL: https://doi.org/10.1007/978-3-319-47301-7_6.

Articles in journals

- [Ruf+20] Cyprien Ruffino, Romain Hérault, Eric Laloy, and Gilles Gasso. “Pixel-Wise Conditioned Generative Adversarial Networks for Image Synthesis and Completion.” In: *Neurocomputing* (Apr. 2020). DOI: 10.1016/j.neucom.2019.11.116. arXiv: 2002.01281. URL: <https://hal.archives-ouvertes.fr/hal-02551730>.
- [Lal+19] Eric Laloy, Niklas Linde, Cyprien Ruffino, Romain Hérault, Gilles Gasso, and Diederik Jacques. “Gradient-Based Deterministic Inversion of Geophysical Data with Generative Adversarial Networks: Is It Feasible?” In: *Computers & Geosciences* (Sept. 24, 2019), p. 104333. ISSN: 0098-3004. DOI: 10.1016/j.cageo.2019.104333. URL: <http://www.sciencedirect.com/science/article/pii/S009830041831207X>.
- [Amy+18] Amine Amyar, Su Ruan, Isabelle Gardin, Romain Hérault, Chatelain Clement, Pierre Decazes, and Romain Modzelewski. “Radiomics-Net: Convolutional Neural Networks on FDG PET Images for Predicting Cancer Treatment Response.” In: *Journal of Nuclear Medicine* 59 (supplement 1 2018), p. 324. URL: <https://hal-normandie-univ.archives-ouvertes.fr/hal-02129431>.
- [Bel+18] Soufiane Belharbi, Romain Hérault, Clément Chatelain, and Sébastien Adam. “Deep Neural Networks Regularization for Structured Output Prediction.” In: *Neurocomputing* 281 (Mar. 15, 2018), pp. 169–177. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2017.12.002. URL: <http://www.sciencedirect.com/science/article/pii/S0925231217318295>.
- [Lal+18] Eric Laloy, Romain Hérault, Diederik Jacques, and Niklas Linde. “Training-Image Based Geostatistical Inversion Using a Spatial Generative Adversarial Neural Network.” In: *Water Resources Research* 54.1 (Jan. 1, 2018), pp. 381–406. ISSN: 1944-7973. DOI: 10.1002/2017WR022148. URL: <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2017WR022148>.
- [Rid+18] Imad Rida, Romain Hérault, Gian Luca Marcialis, and Gilles Gasso. “Palmprint Recognition with an Efficient Data Driven Ensemble Classifier.” In: *Pattern Recognition Letters* (Apr. 22, 2018). ISSN: 0167-8655. DOI: 10.1016/j.patrec.2018.04.033. URL: <http://www.sciencedirect.com/science/article/pii/S0167865518301612>.

- [Sei+18] Ludovic Seifert, Dominic Orth, Bruno Mantel, Jérémie Boulanger, Romain Hérault, and Matt Dicks. “Affordance Realization in Climbing: Learning and Transfer.” In: *Frontiers in Psychology* 9 (May 2018). DOI: 10.3389/fpsyg.2018.00820. URL: <https://hal.archives-ouvertes.fr/hal-02094976>.
- [Bel+17] Soufiane Belharbi, Clément Chatelain, Romain Hérault, Sébastien Adam, Sébastien Thureau, Mathieu Chastan, and Romain Modzelewski. “Spotting L3 Slice in CT Scans Using Deep Convolutional Network and Transfer Learning.” In: *Computers in Biology and Medicine* 87 (Aug. 1, 2017), pp. 95–103. ISSN: 0010-4825. DOI: 10.1016/j.compbiomed.2017.05.018. URL: <http://www.sciencedirect.com/science/article/pii/S0010482517301403>.
- [Hér+17] Romain Hérault, Dominic Orth, Ludovic Seifert, Jérémie Boulanger, and John Aldo Lee. “Comparing Dynamics of Fluency and Inter-Limb Coordination in Climbing Activities Using Multi-Scale Jensen–Shannon Embedding and Clustering.” In: *Data Mining and Knowledge Discovery* 31.6 (Nov. 2017), pp. 1758–1792. DOI: 10.1007/s10618-017-0522-1. URL: <https://hal.archives-ouvertes.fr/hal-02094958>.
- [Lal+17] Eric Laloy, Romain Hérault, John Lee, Diederik Jacques, and Niklas Linde. “Inversion Using a New Low-Dimensional Representation of Complex Binary Geological Media Based on a Deep Neural Network.” In: *Advances in Water Resources* 110 (Dec. 2017), pp. 387–405. DOI: 10.1016/j.advwatres.2017.09.029. URL: <https://hal.archives-ouvertes.fr/hal-02094960>.
- [Bou+16] Jeremie Boulanger, Ludovic Seifert, Romain Hérault, and Jean-François Coeurjolly. “Automatic Sensor-Based Detection and Classification of Climbing Activities.” In: *IEEE Sensors Journal* 16.3 (Feb. 2016), pp. 742–749. DOI: 10.1109/JSEN.2015.2481511. URL: <https://hal.archives-ouvertes.fr/hal-01225056>.
- [Ler+15] Julien Lerouge, Romain Hérault, Clément Chatelain, Fabrice Jardin, and Romain Modzelewski. “IODA: An Input/Output Deep Architecture for Image Labeling.” In: *Pattern Recognition* 48.9 (Sept. 1, 2015), pp. 2847–2858. ISSN: 0031-3203. DOI: 10.1016/j.patcog.2015.03.017. URL: <http://www.sciencedirect.com/science/article/pii/S0031320315001181>.
- [Cho+14b] Jia Yi Chow, Ludovic Seifert, Romain Hérault, Shannon Jing Yi Chia, and Miriam Chang Yi Lee. “A Dynamical System Perspective to Understanding Badminton Singles Game Play.” In: *Human Movement Science* 33 (Feb. 1, 2014), pp. 70–84. ISSN: 0167-9457. DOI: 10.1016/j.humov.2013.07.016. URL: <http://www.sciencedirect.com/science/article/pii/S0167945713000985>.
- [Dov+14] Vladislavs Dovgalecs, Jérémie Boulanger, Dominique Orth, Romain Hérault, Jean-François Coeurjolly, Keith Davids, and Ludovic Seifert. “Movement Phase Detection in Climbing.” In: *Sports Technology. Rock Climbing* 7.3-4 (2014), pp. 174–182. DOI: 10.1080/19346182.2015.1064128. URL: <https://hal.archives-ouvertes.fr/hal-01071401>.
- [Sei+14a] Ludovic Seifert, Maxime L’Hermette, John Komar, Dominic Orth, Florian Mell, Pierre Merriaux, Pierre Grenet, Yanis Caritu, Romain Hérault, Vladislavs Dovgalecs, and Keith Davids. “Pattern Recognition in Cyclic and Discrete Skills Performance from Inertial Measurement Units.” In: *Procedia Engineering* 72 (2014), pp. 196–201. DOI: 10.1016/j.proeng.2014.06.033. URL: <https://hal-normandie-univ.archives-ouvertes.fr/hal-02096541>.
- [Sei+14b] Ludovic Seifert, Dominic Orth, Jérémie Boulanger, Vladislavs Dovgalecs, Romain Hérault, and Keith Davids. “Climbing Skill and Complexity of Climbing Wall Design: Assessment of Jerk as a Novel Indicator of Performance Fluency.” In: *Journal of Applied Biomechanics* 30.5 (Oct. 2014), pp. 619–625. DOI: 10.1123/jab.2014-0052. URL: <https://hal.archives-ouvertes.fr/hal-02094928>.
- [Sei+14c] Ludovic Seifert, Léo Wattebled, Romain Hérault, Germain Poizat, David Adé, Nathalie Gal-Petitfaux, and Keith Davids. “Neurobiological Degeneracy and Affordance Perception Support Functional Intra-Individual Variability of Inter-Limb Coordination during Ice Climbing.” In: *PLOS ONE* 9.2 (Feb. 24, 2014), e89865. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0089865. URL: <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0089865>.

- [BHS13] Gautier Bideault, Romain Herault, and Ludovic Seifert. “Data Modelling Reveals Inter-Individual Variability of Front Crawl Swimming.” In: *Journal of Science and Medicine in Sport* 16.3 (May 1, 2013), pp. 281–285. ISSN: 1440-2440. DOI: 10.1016/j.jsams.2012.08.001. URL: <http://www.sciencedirect.com/science/article/pii/S1440244012001715>.
- [Sei+13a] Ludovic Seifert, Jean-François Coeurjolly, Romain Héroult, Leo Wattedled, and Keith Davids. “Temporal Dynamics of Inter-Limb Coordination in Ice Climbing Revealed through Change-Point Analysis of the Geodesic Mean of Circular Data.” In: *Journal of Applied Statistics* 40.11 (Nov. 2013), pp. 2317–2331. DOI: 10.1080/02664763.2013.810194. URL: <https://hal.archives-ouvertes.fr/hal-02094911>.
- [Sei+13c] Ludovic Seifert, Léo Wattedled, Maxime L’Hermette, Gautier Bideault, Romain Herault, and Keith Davids. “Skill Transfer, Affordances and Dexterity in Different Climbing Environments.” In: *Human Movement Science* 32.6 (Dec. 1, 2013), pp. 1339–1352. ISSN: 0167-9457. DOI: 10.1016/j.humov.2013.06.006. URL: <http://www.sciencedirect.com/science/article/pii/S0167945713000766>.
- [Sei+11a] Ludovic Seifert, Hugues Leblanc, Romain Herault, John Komar, Chris Button, and Didier Chollet. “Inter-Individual Variability in the Upper–Lower Limb Breaststroke Coordination.” In: *Human Movement Science* 30.3 (June 1, 2011), pp. 550–565. ISSN: 0167-9457. DOI: 10.1016/j.humov.2010.12.003. URL: <http://www.sciencedirect.com/science/article/pii/S016794571000182X>.
- [Sei+11b] Ludovic Seifert, Leo Wattedled, Maxime L’Hermette, and Romain Herault. “Inter-Limb Coordination Variability in Ice Climbers of Different Skill Level.” In: *Baltic Journal of Sport and Health Sciences* 1.80 (2011). URL: <https://journals.lsu.lt/baltic-journal-of-sport-health/article/download/342/338>.

Articles in conferences

- [Ruf+19a] Cyprien Ruffino, Romain Héroult, Eric Laloy, and Gilles Gasso. “Pixel-Wise Conditioning of Generative Adversarial Networks.” In: European Symposium on Artificial Neural Networks (ESANN). Bruges, Belgium, Apr. 24, 2019. URL: <https://hal.archives-ouvertes.fr/hal-02347732>.
- [Bel+16a] Soufiane Belharbi, Romain Herault, Clement Chatelain, and Sebastien Adam. “Deep Multi-Task Learning with Evolving Weights.” In: European Symposium on Artificial Neural Networks (ESANN). Bruges, Belgium, 2016, p. 6. URL: <https://sbelharbi.github.io/publications/2016/presentation-ESANN2016-bleharbi.pdf>.
- [Sei+15] Ludovic Seifert, Vladislavs Dovgalecs, Jérémie Boulanger, Dominic Orth, Romain Héroult, and Keith Davids. “Full-Body Movement Pattern Recognition in Climbing.” In: *Sports Technology* 7.3-4 (July 2015), pp. 166–173. DOI: 10.1080/19346182.2014.968250. URL: <https://hal.archives-ouvertes.fr/hal-02094936>.
- [RHG14] Imad Rida, Romain Héroult, and Gilles Gasso. “Supervised Music Chord Recognition.” In: 2014 13th International Conference on Machine Learning and Applications (ICMLA). IEEE, Dec. 3, 2014, pp. 336–341. DOI: 10.1109/ICMLA.2014.60. URL: <https://hal-normandie-univ.archives-ouvertes.fr/hal-02096549>.
- [Kei+13] Abou Keita, Romain Héroult, Colas Calbrix, and Stéphane Canu. “Detection and Quantification in Real-Time Polymerase Chain Reaction.” In: European Symposium on Artificial Neural Networks (ESANN). Bruges, Belgium, Apr. 2013, p. 351. URL: <https://hal.archives-ouvertes.fr/hal-00834417>.
- [Sei+10c] Ludovic Seifert, Leo Wattedled, Maxime L’Hermette, and Romain Héroult. “Inter-Limb Coordination Variability in Ice Climbers of Different Skill Level.” In: *3rd International Congress Complex Systems in Medicine and Sport* (Lithuania). Sept. 2010, pp. 105–106. URL: <https://hal.archives-ouvertes.fr/hal-00558152>.
- [LHC09] Benjamin Labbé, Romain Héroult, and Clement Chatelain. “Learning Deep Neural Networks for High Dimensional Output Problems.” In: *ICMLA* (United States). Dec. 2009, 6p. URL: <https://hal.archives-ouvertes.fr/hal-00438714>.

- [HG07b] Romain Héroult and Yves Grandvalet. “Sparse Probabilistic Classifiers.” In: *ICML '07: Proceedings of the 24th International Conference on Machine Learning* (New York, NY, USA, United States). ACM, June 2007, pp. 337–344. doi: 10 . 1145 / 1273496 . 1273539. URL: <https://hal.archives-ouvertes.fr/hal-00442746>.
- [HDG06] Romain Héroult, Franck Davoine, and Yves Grandvalet. “Head and Facial Action Tracking: Comparison of Two Robust Approaches.” In: *7th IEEE International Conference on Automatic Face and Gesture Recognition* (Southampton, UK, United Kingdom). IEEE Computer Society, Apr. 2006, pp. 287–292. URL: <https://hal.archives-ouvertes.fr/hal-00442753>.

Articles in workshops

- [Bel+15b] Soufiane Belharbi, Clement Chatelain, Romain Héroult, and Sébastien Adam. “Learning Structured Output Dependencies Using Deep Neural Networks.” In: Deep Learning Workshop, ICML. 2015. URL: https://www.researchgate.net/profile/Clement_Chatelain/publication/293097934_Learning_Structured_Output_Dependencies_Using_Deep_Neural_Networks/links/56f5a15b08ae7c1fda2eea19/Learning-Structured-Output-Dependencies-Using-Deep-Neural-Networks.pdf.
- [Hér+15] Romain Héroult, Jeremie Boulanger, Ludovic Seifert, and John Aldo Lee. “Valuation of Climbing Activities Using Multi-Scale Jensen-Shannon Neighbour Embedding.” In: Machine Learning and Data Mining for Sports Analytics, ECML/PKDD 2015 workshop (MLSA2015), 2015. URL: <https://hal.archives-ouvertes.fr/hal-01441636>.
- [KHS14] John Komar, Romain Héroult, and Ludovic Seifert. “Key Point Selection and Clustering of Swimmer Coordination through Sparse Fisher-EM.” In: ECML/PKDD 2013 Workshop on Machine Learning and Data Mining for Sports Analytics (MLSA2013). Jan. 7, 2014. arXiv: 1401.1489 [physics, stat]. URL: <http://arxiv.org/abs/1401.1489>.

Editorials in journals

- [ABH17] Fabio Aioli, Gaëlle Bonnet-Loosli, and Romain Héroult. “Advances in Artificial Neural Networks, Machine Learning and Computational Intelligence, Special Issue ESANN 2017 (Editorial).” In: *Neurocomputing*. Advances in Artificial Neural Networks, Machine Learning and Computational Intelligence 268 (Dec. 13, 2017), pp. 1–3. ISSN: 0925-2312. doi: 10.1016/j.neucom.2017.04.038. URL: <http://www.sciencedirect.com/science/article/pii/S0925231217307634>.
- [Aio+16] Fabio Aioli, Kerstin Bunte, Romain Héroult, and Mikhail Kanevski. “Advances in Artificial Neural Networks, Machine Learning and Computational Intelligence, Special Issue ESANN 2015 (Editorial).” In: *Neurocomputing*. Advances in Artificial Neural Networks, Machine Learning and Computational Intelligence 192 (June 5, 2016), pp. 1–2. ISSN: 0925-2312. doi: 10.1016/j.neucom.2016.02.005. URL: <http://www.sciencedirect.com/science/article/pii/S0925231216001831>.

Invited speeches

- [Hér19a] Romain Héroult. “Deep Generative Model.” Séminaire École Doctorale CIL (Université libre de Liège). June 4, 2019.
- [Hér19b] Romain Héroult. “Deep Learning.” Séminaire École Doctorale CIL (Université catholique de Louvain). May 20, 2019.

Communications in conferences

- [Ort+13] Dominic Orth, Keith Davids, Romain Héroult, and Ludovic Seifert. “Indices of Behavioral Complexity over Repeated Trials in a Climbing Task: Evaluating Mechanisms Underpinning Emergence of Skilled Performance,” in: European Conferences on Complex Systems (ECCS). Barcelona, 2013.
- [Sei+13b] Ludovic Seifert, Dominic Orth, Romain Héroult, and Keith Davids. “Metastability in Perception and Action in Rock Climbing.” In: *XVIIIth International Conference on Perception and Action*. Estoril: FMH Editions, Portugal. Estoril, Portugal, 2013.

Communications in workshops

- [Sei+10b] Ludovic Seifert, Leo Wattebled, Maxime L’Hermette, and Romain Hérault. “Effect of Skill Level on Upper/Lower Limb Coordination in Ice Climbers.” In: *11th European Workshop of Ecological Psychology* (France). June 2010, pp. 68–69. URL: <https://hal.archives-ouvertes.fr/hal-00558158>.

Accepted works

- [Ani+20] Anna Aniszewska-Stępień, Romain Hérault, Guillaume Hacques, Ludovic Seifert, and Gilles Gasso. “Learning from Partially Labeled Sequences for Behavioral Signal Annotation.” In: Accepted to Machine Learning and Data Mining for Sports Analytics (MLSA) 2020. Ghent, Belgium, Sept. 14, 2020.
- [Kec+20] Marwa Kechaou, Romain Hérault, Mokhtar Z. Alaya, and Gilles Gasso. “Open Set Domain Adaptation Using Optimal Transport.” In: Accepted in European Conference on Machine Learning, and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD). Ghent, Belgium, Sept. 14, 2020.

Submitted works

- [Bli+] Rachel Blin, Cyprien Ruffino, Stéphane Canu, Gilles Gasso, Samia Ainouz, Fabrice Meriaudeau, and Romain Hérault. “Generating Polarimetric-Encoded Images Using Constrained Cycle-Consistent Generative Adversarial Networks.” In: Submitted to ACCV.

Local / National audience

Articles in journals

- [HG08] Romain Hérault and Yves Grandvalet. “Classifieurs Probabilistes Parcimonieux.” In: *Traitement du Signal* 25.4 (2008), pp. 279–291. URL: <https://hal.archives-ouvertes.fr/hal-00442731>.

Articles in conferences

- [Ruf+19b] Cyprien Ruffino, Romain Hérault, Éric Laloy, and Gilles Gasso. “Approche GAN Pour La Génération d’images Sous Contraintes de Pixel.” In: Conférence Sur l’Apprentissage Automatique (CAp). Toulouse, France, 2019, pp. 439–444.
- [Ruf+18] Cyprien Ruffino, Romain Hérault, Eric Laloy, and Gilles Gasso. “Dilated Spatial Generative Adversarial Networks for Ergodic Image Generation.” In: Conférence Sur l’Apprentissage Automatique (CAp). Rouen, France, June 20, 2018. arXiv: 1905.08613 [cs, eess]. URL: <http://arxiv.org/abs/1905.08613>.
- [Bel+16b] Soufiane Belharbi, Romain Hérault, Clément Chatelain, and Sébastien Adam. “Pondération Dynamique Dans Un Cadre Multi-Tâche Pour Réseaux de Neurones Profonds.” In: *Session Spéciale” Apprentissage et Vision”*. Reconnaissance Des Formes et l’Intelligence Artificielle (RFIA). Clermont-Ferrand, France, 2016. URL: <https://sbelharbi.github.io/publications/2016/RFIA2016-belharbi.pdf>.
- [Bel+15a] Soufiane Belharbi, Clement Chatelain, Romain Hérault, and Sebastien Adam. “A Unified Neural Based Model for Structured Output Problems.” In: Conférence Sur l’Apprentissage Automatique (CAp). Lille, France, 2015, p. 10. URL: <https://sbelharbi.github.io/publications/2015/belharbiCAP2015.pdf>.
- [KHC12] Abou Keita, Romain Hérault, and Stéphane Canu. “Estimation de La Concentration d’un Agent Biologique Par Détection de Rupture Sur Vidéos de Fluorescences Issues de PCR.” In: Reconnaissance Des Formes et l’Intelligence Artificielle (RFIA) (Lyon, France). Lyon, France, Jan. 2012, pp. 978-2-9539515-2-3. URL: <https://hal.archives-ouvertes.fr/hal-00656568>.
- [Tia+10] X Tian, Romain Hérault, Gilles Gasso, and Stephane Canu. “Pré-Apprentissage Supervisé Pour Les Réseaux Profonds.” In: Reconnaissance Des Formes et l’Intelligence Artificielle (RFIA). Caen, France, 2010.

- [HG07a] Romain Hérault and Yves Grandvalet. “Régression Logistique Parcimonieuse.” In: *Conférence Sur l’Apprentissage Automatique (CAp)* (Grenoble, France, France). Ed. by Cépaduès. Grenoble, France, July 2007, pp. 265–280. URL: <https://hal.archives-ouvertes.fr/hal-00442755>.
- [Hér+06] Romain Hérault, Franck Davoine, Fadi Dornaika, and Yves Grandvalet. “Suivis Simultanés et Robustes de Visages et de Gestes Faciaux.” In: *Reconnaissance Des Formes et l’Intelligence Artificielle (RFIA)* (Tours, France, France). Tour, France, Jan. 2006. URL: <https://hal.archives-ouvertes.fr/hal-00442758>.

Invited speeches

- [Hér20] Romain Hérault. “Deep Generative Model.” Séminaire IMVIA, ESIREM (Université de Bourgogne). Jan. 23, 2020.
- [Hér17] Romain Hérault. “Deep Learning.” Research Summer School on Statistics & BigData Science - SBDS (Université de Caen). June 8, 2017.

Thesis

- [Hér07] Romain Hérault. “Vision et Apprentissage Statistique Pour La Reconnaissance d’items Comportementaux.” thesis. Compiègne, Nov. 26, 2007. URL: <http://www.theses.fr/2007COMP1715>.

Selected publications in appendix A

- [Ler+15] Julien Lerouge et al. “IODA: An Input/Output Deep Architecture for Image Labeling.” In: *Pattern Recognition* 48.9 (Sept. 1, 2015), pp. 2847–2858. ISSN: 0031-3203. DOI: 10.1016/j.patcog.2015.03.017. URL: <http://www.sciencedirect.com/science/article/pii/S0031320315001181> (Appendix A.1)
- [Bel+17] Soufiane Belharbi et al. “Spotting L3 Slice in CT Scans Using Deep Convolutional Network and Transfer Learning.” In: *Computers in Biology and Medicine* 87 (Aug. 1, 2017), pp. 95–103. ISSN: 0010-4825. DOI: 10.1016/j.combiomed.2017.05.018. URL: <http://www.sciencedirect.com/science/article/pii/S0010482517301403> (Appendix A.2)
- [Bel+18] Soufiane Belharbi et al. “Deep Neural Networks Regularization for Structured Output Prediction.” In: *Neurocomputing* 281 (Mar. 15, 2018), pp. 169–177. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2017.12.002. URL: <http://www.sciencedirect.com/science/article/pii/S0925231217318295> (Appendix A.3)
- [Ruf+20] Cyprien Ruffino et al. “Pixel-Wise Conditioned Generative Adversarial Networks for Image Synthesis and Completion.” In: *Neurocomputing* (Apr. 2020). DOI: 10.1016/j.neucom.2019.11.116. arXiv: 2002.01281. URL: <https://hal.archives-ouvertes.fr/hal-02551730> (Appendix A.4)
- [Sei+13a] Ludovic Seifert et al. “Temporal Dynamics of Inter-Limb Coordination in Ice Climbing Revealed through Change-Point Analysis of the Geodesic Mean of Circular Data.” In: *Journal of Applied Statistics* 40.11 (Nov. 2013), pp. 2317–2331. DOI: 10.1080/02664763.2013.810194. URL: <https://hal.archives-ouvertes.fr/hal-02094911> (Appendix A.5)
- [Hér+17] Romain Hérault et al. “Comparing Dynamics of Fluency and Inter-Limb Coordination in Climbing Activities Using Multi-Scale Jensen–Shannon Embedding and Clustering.” In: *Data Mining and Knowledge Discovery* 31.6 (Nov. 2017), pp. 1758–1792. DOI: 10.1007/s10618-017-0522-1. URL: <https://hal.archives-ouvertes.fr/hal-02094958> (Appendix A.6)
- [KHS14] John Komar, Romain Hérault, and Ludovic Seifert. “Key Point Selection and Clustering of Swimmer Coordination through Sparse Fisher-EM.” in: *ECML/PKDD 2013 Workshop on Machine Learning and Data Mining for Sports Analytics (MLSA2013)*. Jan. 7, 2014. arXiv: 1401.1489 [physics, stat]. URL: <http://arxiv.org/abs/1401.1489> (Appendix A.7)
- [Rid+17] Imad Rida et al. “Improved Model-Free Gait Recognition Based on Human Body Part.” In: *Biometric Security and Privacy: Opportunities & Challenges in The Big Data Era*. Ed. by Richard Jiang et al. Signal Processing for Security Technologies. Cham: Springer International Publishing, 2017, pp. 141–161. ISBN: 978-3-319-47301-7. DOI: 10.1007/978-3-319-47301-7_6. URL: https://doi.org/10.1007/978-3-319-47301-7_6 (Appendix A.8)

Chapter 2

Introduction to Machine Learning and Deep Learning

In this preliminary chapter, we will recall what are the machine learning framework and its deep learning specialization. I do not intend to be exhaustive but to bring out the notions needed for the two upcoming chapters for my proper work on high dimension / structured problems (Chapter 3) and on human movement (Chapter 4). The first section will be dedicated to describing the machine learning frameworks. Artificial neurons and simple neural networks for supervised learning will be depicted in the second section. We will then present how neural networks can be used for unsupervised learning. Having in mind, the limitation of neural networks highlighted in the two precedent sections, we will present how to overcome them in the deep learning framework. The two following sections will concentrate on the application of deep learning to image processing and to sequence processing respectively. Finally, we will describe how to use neural networks as a generative model.

Contents

2.1 From Artificial Intelligence (AI) to Deep Learning (DL)	29
2.1.1 Machine learning context and frameworks	30
2.1.2 Supervised learning	30
2.1.3 Unsupervised learning	33
2.2 Artificial Neural Network (ANN) for supervised learning	36
2.2.1 Perceptron	36
2.2.2 Multi Layer Perceptron	37
2.2.3 Recurrent Neural Network (RNN)	39
2.3 Auto-Encoder (AE), an ANN for unsupervised learning	41
2.3.1 Auto-Encoder architecture	41
2.3.2 Auto-encoder training	42
2.4 Deep Learning	44
2.4.1 Definition	44
2.4.2 Tips and tricks to avoid gradient problems	45
2.4.3 Convolutional Neural Networks	51
2.5 Deep Generative Models	56
2.5.1 Variational auto-encoders	57
2.5.2 Generative Adversarial Network	57

Notations

To clarify the notations used in the following chapter, a scalar is displayed by a small letter such as x ; small bold letter is used for vectors, thus \mathbf{x} is a vector.

When we deal with matrices or tensor, we write a capital bold letter. For example, when the input represents an image that has not been vectorized but kept in its original shape it is noted by \mathbf{X} . Sets are represented by calligraphic letters, e.g. \mathcal{X} and spaces by blackboard bold letter, for example \mathbb{X} .

An estimation/prediction is indicated by the hat symbol as in $\hat{\mathbf{x}}$, an inferred parameter by the star symbol as in \mathbf{x}^* .

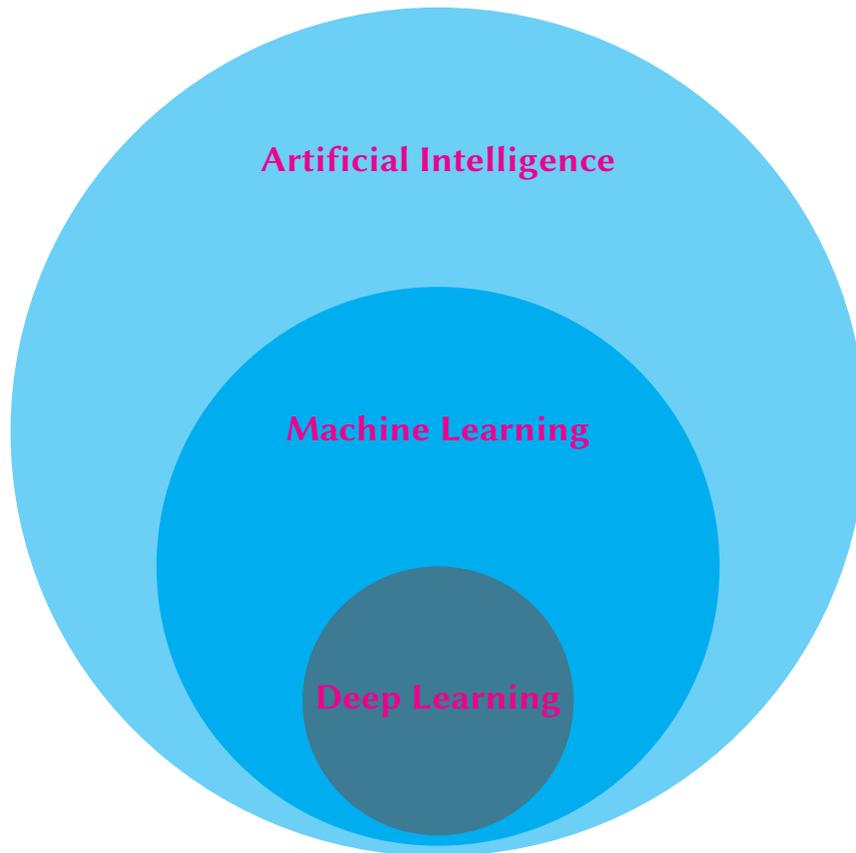


Figure 2.1: Relation between Artificial Intelligence, Machine Learning and Deep Learning.

2.1 From Artificial Intelligence (AI) to Deep Learning (DL)

If the definition of *Intelligence* is yet difficult and controversial, the one of *Artificial Intelligence* (AI) is even more so. In [RN16], the authors have found 4 categories of definitions where the behavior of a program or a device can be attributed to AI if it is 1. acting humanly, 2. thinking humanly, 3. acting rationally, 4. or thinking rationally. Nevertheless, what is the definition of rationality and what can be surely and exclusively attributed to mankind ?

I will take the definition of *Artificial Intelligence* as the fact that a program or a device displays or mimics cognitive behaviors attributed to human mind or intelligent animals. Examples of such cognitive behaviors are deducing or inferring. This definition is still highly imprecise and human-centric.

Machine Learning (ML) is a part of AI dedicated to the latter behavior, *inferring*. ML is not relying on explicit rules, but it builds mathematical models by learning from examples. Thus, a ML program or the resulting model does not consist in a sequence of instructions that explicitly solves a targeted task. Rather, a ML program is mainly an inference procedure where decision/prediction/representation rules are learned from examples trying to optimize a performance criterion linked to the targeted task. As such, ML is closely related to statistics and optimization. Moreover, ML could fall into the *acting rationally* categories of the previous definitions of IA [RN16] : it does not pretend to replace human thought. What counts for evaluating a ML program is **how well** it acts, the **how-to** is not the first target. It can be seen as a black box from which its own reasoning is not accessible.

Deep Learning, in its turn, could be seen as a multiple-stage ML model or a ML procedure that directly processes raw data without doing feature extraction. Therefore, Deep Learning is a part of Machine Learning which is a part of Artificial Intelligence (Fig. 2.1). For more detailed definitions of deep learning, please refer to the section 2.4.

Among the other fields or key-words surrounding Machine Learning, one can find *Data Mining* which is dedicated to exploratory analysis, *Big Data* where local storage or local computation power is not enough for the persistence and the processing of available data, or *Data Science* which gathers all the fields that consist in storing, processing, and extracting knowledge from Data.

2.1.1 Machine learning context and frameworks

Machine learning methods can be split into broad families according to the following use contexts:

Supervised Learning that tries to predict the output linked to an input knowing correct pairs of (input, output),

Active Learning that, as supervised learning, predicts the output linked to an input but this time by interactively asking for information,

Unsupervised Learning that tries to establish a neighborhood or to find common patterns inside a dataset with only a set of (input),

Reinforcement Learning that decides which action to undertake to maximize a reward.

A combination of the preceding contexts can occur. For example, in *semi-supervised learning*, a classifier is learned knowing correct pairs of (input,output) as in supervised learning but also knowing samples of (input) only as in unsupervised learning.

In this work we will focus on *supervised*, *semi-supervised* and *unsupervised* learnings.

2.1.2 Supervised learning

In supervised learning we deal with two spaces: an input space \mathbb{X} and an output space \mathbb{Y} . There is a hidden link $g : \mathbb{X} \rightarrow \mathbb{Y}$ between these two spaces (Fig. 2.2a) and we want to guess that link. On simple setups, such as fixed size output spaces, we can distinguish between:

- *Classification* tasks where the output space \mathbb{Y} is discrete (like recognizing dog versus cat on images),
- *Regression* tasks where \mathbb{Y} is continuous (like predicting the long term benefit of a stock action).

Nevertheless, there is no assumption on both the input and output spaces. They could be (not exclusively) scalars, vectors, matrices (images), tensors, sequences, graphs, ... A task such as building a translator between two languages can be seen as a supervised learning task as long as a corpus of the corresponding sentences in the two languages is known. In this case, it is a *sequence to sequence* problem: both the input and output spaces consist in sequences.

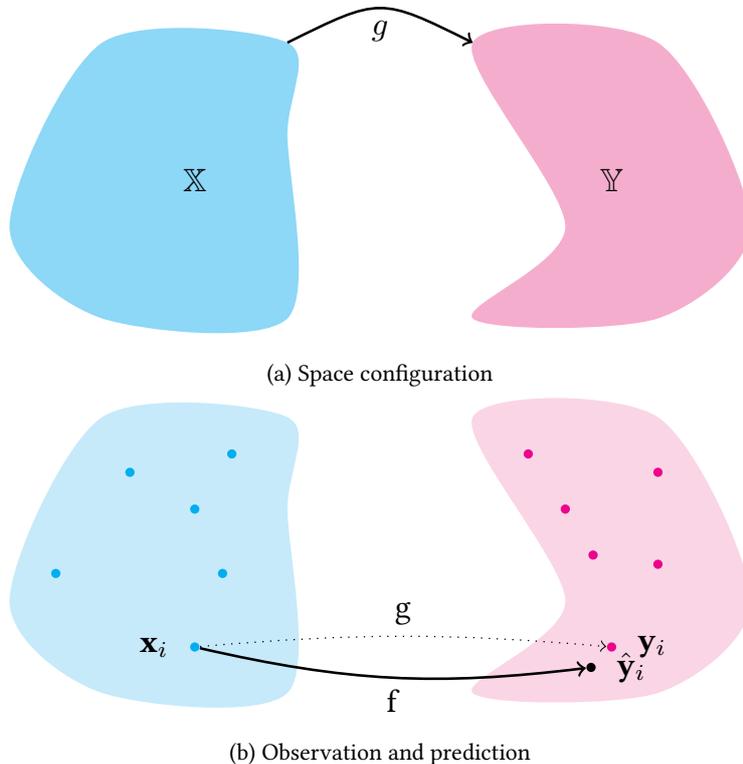


Figure 2.2: Supervised learning setup

Formally, the purpose of supervised learning is to find a function $f \in \mathcal{H}$ that approximates g knowing observed samples (\mathbf{x}, \mathbf{y}) for which \mathbf{y} equals $g(\mathbf{x})$ (Fig. 2.2b). In other words, we are looking for a function f that aims for the prediction $\hat{\mathbf{y}}_i = f(\mathbf{x}_i)$ to be close to $\mathbf{y}_i = g(\mathbf{x}_i)$ for all the observation \mathbf{x}_i . The hypothesis space \mathcal{H} depends on the chosen machine learning model and its hyper-parameters. Usually, the hidden link g does not belong to this space, thereby f can not match perfectly g . Please note that, even if samples \mathbf{x} are i.i.d., it is not enough to guarantee that having perfect matches for the known observations will give us a good prediction for unknown samples due to a bias/variance dilemma called *overfitting* that will be explained in the next paragraph *Learning the model*.

Features extraction

To reduce the complexity of the model, to process high dimension (images), structured (graphs) or unfixed (sequences) inputs, as well as to treat heterogeneity of the data, in standard machine learning the function f is preceded by a feature extraction stage that is not learned (or inferred from data) but rather handcrafted following the scheme presented in Figure 2.3.

Let's say that we have a classification task (dog versus cat) on a dataset which contains photos of different size, a naive feature extraction step could be for example the computation of the color histogram. This transforms high-dimension and possibly various size images into a fixed low dimension vector. Before the advent of representation learning and deep learning, a key part of the expertise of a data scientist was to know which features to extract for a given task.

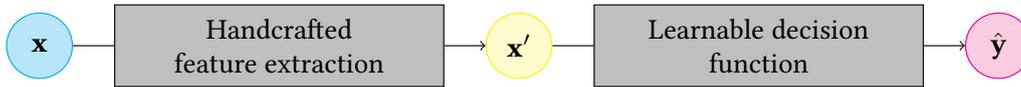


Figure 2.3: Standard machine learning framework using feature extraction

Learning the model

The learning phase consists in choosing a good f among the hypothesis space \mathcal{H} . Most of the machine learning models are parametric models (with k-nn a notable exception) that depend on inner variables or parameters θ . Thus, the learning phase resides in inferring these parameters θ knowing a set of n samples $\mathcal{L} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$, called the training set.

In order to measure if the function f suits the training set \mathcal{L} , we forge a function $L : \mathbb{Y} \times \mathbb{Y} \rightarrow \mathbb{R}^{\geq 0}$ called a loss or cost function. In the case of regression, L is typically the L^2 norm,

$$L(\mathbf{y}, \hat{\mathbf{y}}) = \|\mathbf{y} - \hat{\mathbf{y}}\|_2^2 ,$$

where $\hat{\mathbf{y}} = f(\mathbf{x})$ is the prediction for \mathbf{x} . In the case of classification, L could be the negative log-likelihood.

Inferring the parameters through empirical risk minimization is done by choosing the θ that gives the lowest loss L over the training set \mathcal{L} , that is

$$R_{emp}(f_{\theta}) = \frac{1}{\text{card}(\mathcal{L})} \sum_{(\mathbf{x}, \mathbf{y}) \in \mathcal{L}} L(\mathbf{y}, \hat{\mathbf{y}} = f_{\theta}(\mathbf{x})) ,$$

$$\theta^* = \arg \min_{\theta} R_{emp}(f_{\theta}) ,$$

where R_{emp} stands for the empirical risk.

Let's introduce another set of m samples $\mathcal{T} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^m$, called the test set that is used to evaluate our model.

If the set of the hypothesis \mathcal{H} is small (the model has low capacity), g could be far from \mathcal{H} . The model f will always display a *bias* in its prediction on both samples from the training set \mathcal{L} and unseen samples from the test set \mathcal{T} .

At the opposite, when the set of the hypothesis \mathcal{H} is large, e.g. θ is a high dimension vector, the empirical risk, R_{emp} , will display a great *variance*. Indeed, even if samples are i.i.d., two different training sets \mathcal{L}_a and \mathcal{L}_b will lead to two clearly distinct inferred parameters, θ_a and θ_b , which in turn will perform badly on respectively \mathcal{L}_b and \mathcal{L}_a . Similarly, a high capacity model will give good prediction for \mathcal{L} or yet observed examples, but will give bad scoring for \mathcal{T} or unseen examples. This phenomena is called *overfitting*.

Thereby, at a constant number of training samples, there is a trade-off on the capacity of the model between bias and variance, which is translated by a high loss on unseen samples of \mathcal{T} at low and high capacities (Fig. 2.4). We say that on these two cases the model does not generalize well: *Generalization* is the ability to perform well on unseen samples.

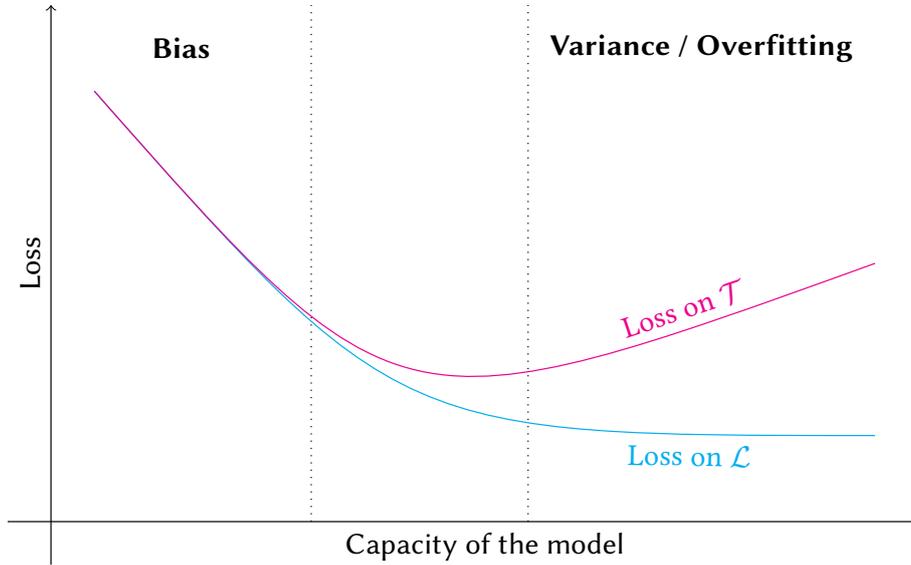


Figure 2.4: Trade-off between *bias* at low capacity (model fails on seen and unseen samples) and *variance/overfitting* at high capacity (model fails on unseen samples) .

In order to prevent overfitting, *Structural risk minimization* introduces a penalization term on the model capacity that plays the role of a regularization. Following the Occam's razor principle, the penalization will force the optimization to choose simpler models over complex ones.

Therefore, the learning optimization problem becomes,

$$\theta^* = \arg \min_{\theta} R_{emp}(f_{\theta}) + \lambda \Omega(f_{\theta}) , \quad (2.1)$$

where Ω is a measure of the capacity of the model and λ a hyper-parameter that controls the bias/variance trade-off. It is similar to the Tikhonov regularization scheme for ill-posed problems.

Typical Ω are L_p norms over the model parameters θ . The L_2 norm is differentiable and so easily manageable by gradient descent, whereas the L_1 norm promotes sparsity among θ but requires careful optimization procedure due to its discontinuity on $\mathbf{0}$.

Selecting the model

A hyper-parameter is a parameter that has been fixed before the training phase described above. We have already seen the λ hyper-parameter that controls the trade-off between the bias and variance. One could note for example that the degree in a polynomial regression is also an hyper-parameter as it is not tuned by the optimization procedure on the training set. In fact, the choice of the model by itself is a hyper-parameter.

When enough examples are available, a validation set \mathcal{V} is created. A sample and try strategy is then applied to select good hyper-parameters. A set of hyper-parameters is chosen, the model is learned on \mathcal{L} and then evaluated on \mathcal{V} . The operation is repeated for many others set of hyper-parameters. The set that gives the best scores on \mathcal{V} is elected for a final learning, the resulting model is then evaluated on the test set \mathcal{T} . This procedure can be systematized by discretizing continuous hyper-parameters and applying a grid-search strategy. In order not to bias model selection, no correlation should be occurring between the training set \mathcal{L} , the validation set \mathcal{V} and the test set \mathcal{T} .

When only the training set \mathcal{L} and the test set \mathcal{T} are available, the evaluation of an hyper-parameter set could be done by a rotation estimation strategy (or out-of-sample testing) on \mathcal{L} which is called *cross-validation*.

2.1.3 Unsupervised learning

Unsupervised learning are machine learning tasks that deal with only one space, the data space \mathbb{X} (Fig. 2.5). A set of n observed samples (\mathbf{x}_i) constitute the training set, $\mathcal{L} = \{(\mathbf{x}_i)\}_{i=1}^n$. These methods rely on clusters or neighborhood inside \mathcal{L} .

Among the possible unsupervised tasks, I will detail 3 main categories: • Clustering, • Representation learning and, • Novelty detection.

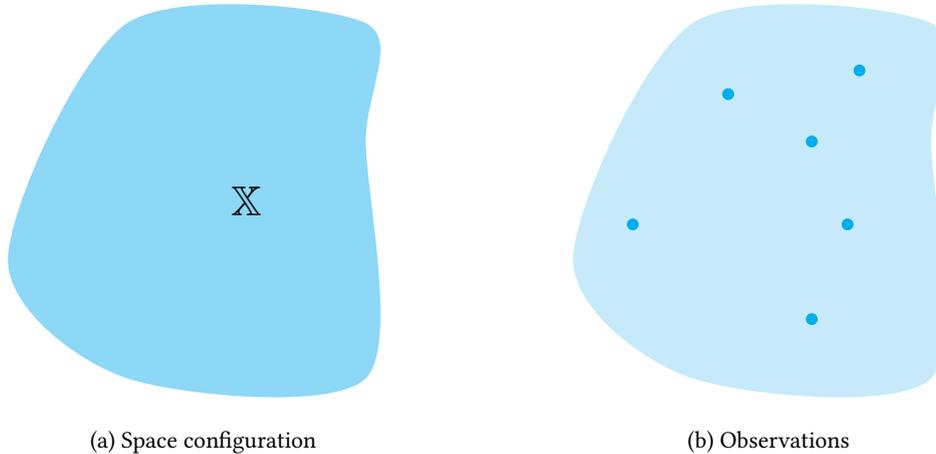


Figure 2.5: Unsupervised learning setup

Clustering

The task of *clustering* or *cluster analysis* consists in splitting a dataset \mathcal{L} into groups or clusters where samples inside one cluster are more similar together than to samples outside this cluster (Fig. 2.6).

Different points of view on what a cluster is lead to different clustering methods:

Gap assumption Higher density areas in the dataset represent clusters. Gaps, void areas, or lower density areas represent separation between clusters. Samples that may appear in these gaps are considered outliers or noise. Methods assuming this point of view are generally based on neighborhood between samples. Two samples far away are likely to be in two different clusters, whereas two samples nearby are likely to be in the same cluster. Hierarchical clustering [Sib73; Def77], DBSCAN [Est+96], MeanShift [FH75] belongs to this category.

Generative process assumption The dataset has been generated by different generative processes. A cluster is a set of samples coming from the same process. This point of view can deal with overlapping clusters and low/scarse densities areas. EM clustering [DLR77], K-means [Ste56; Mac+67] belongs to this category.

Both points of view suffer from the curse of dimensionality for all distances shrink in high dimension spaces and therefore parameters estimation (of the generative processes) is ill-posed.

During clustering, whether a discrete label $y \in \{1..k\}$ or the probability $p(y = i|\mathbf{x})\forall i \in \{1..k\}$ is computed for each sample \mathbf{x} in \mathcal{L} . The number of clusters k is an hyper-parameter of the task.

As a by-product of some clusterings, a labeling function $f : \mathbb{X} \rightarrow \mathbb{Y}$ may be available for new samples not included in the original training set \mathcal{L} .

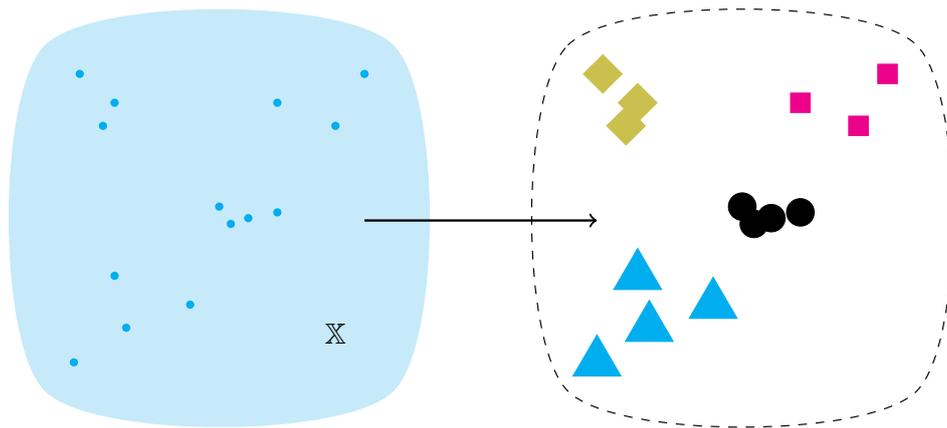


Figure 2.6: The clustering task

Representation learning

The task of *representation learning* consists in automatically learning a feature extraction or more generally finding a new representation of the data to ease later analysis, a supervised task or simply the display of the data (Fig. 2.7). In the supervised framework, this leverages the problem of feature engineering by human knowledge/*a priori* to a machine learning task.

Dimension reduction, such as PCA [Pea01] or ICA [HA84; Com94], Dictionary Learning [EAH99], Vector Quantization [BSB83], Auto-Encoders [Kra91] are examples of representation learning methods.

Formally in *representation learning*, the samples $\mathbf{x} \in \mathbb{X}$ of the training dataset \mathcal{L} are transported to the representation space \mathbb{X}' through a function $f : \mathbb{X} \rightarrow \mathbb{X}'$ learned during the training. New samples in \mathbb{X} observed after the training can also be transported to \mathbb{X}' through this function f .

Embedding is a similar task as *representation learning* but this time the function f is not explicit and not accessible. New samples in \mathbb{X} observed after the training can not be transported to \mathbb{X}' without *a posteriori* approximating f .

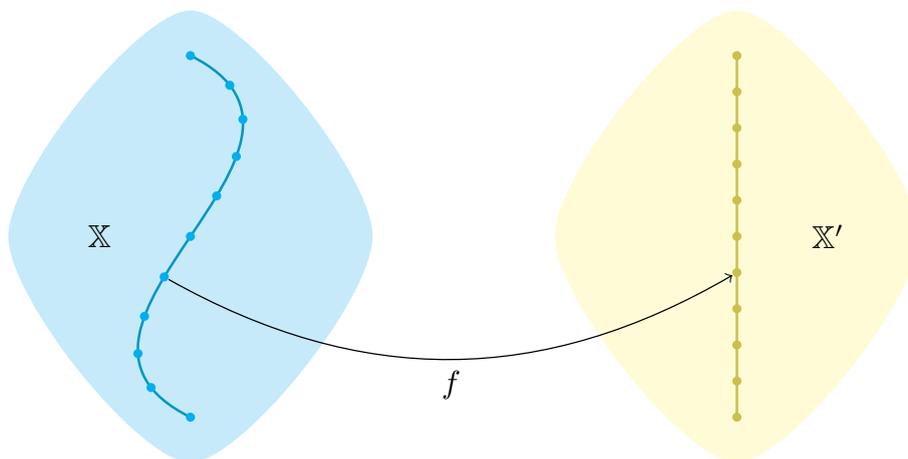


Figure 2.7: An example of representation learning, manifold discovery

Change/Novelty detection

In the task of *novelty detection*, we try to detect when or where there is a change in the (statistical) properties of a signal (stochastic process, time series, 2D or more complex topologies ...). In the example depicted in Figure 2.8, we had to guess if the new samples (magenta cross marks) are generated from the same distribution as past samples (cyan dot marks). In Figure 2.9, the same task is depicted for a 1-D time series. CUSUM [NVK93], filtered derivative [BB84], one class SVM [Rät+00] can solve this problem. Possible applications are fraud or intrusion detection, spam filtering, quality controls Moreover, a image can be seen as a 2D signal, thus the unsupervised image segmentation task can also be classified as a change/novelty detection task.

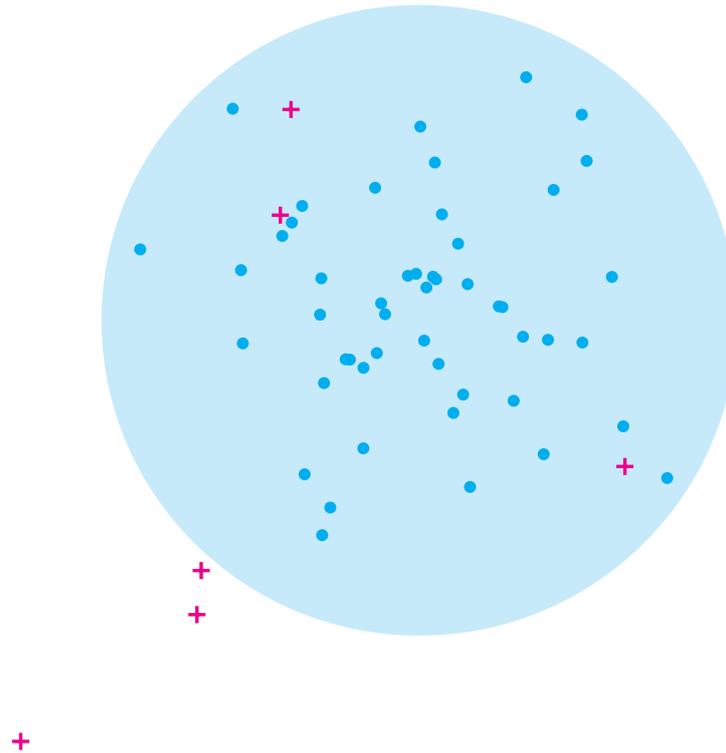


Figure 2.8: An example of novelty detection: do the new samples (magenta cross) belong to the same distribution as old samples (cyan dot) ?

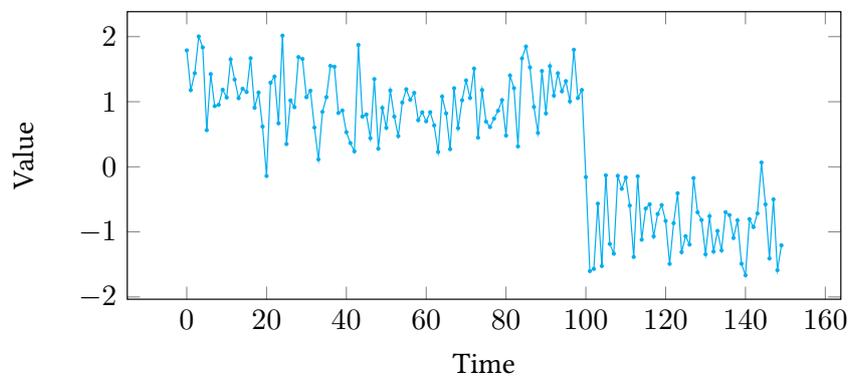


Figure 2.9: Change detection on a time series

2.2 Artificial Neural Network (ANN) for supervised learning

Among the possible models for supervised and unsupervised learnings Artificial Neural Network (ANN) is one of the most popular since the availability of big database and of computation on Graphical Processor Units (GPU). In this section we will describe how they work from the simple perceptron up to deep architectures.

2.2.1 Perceptron

Originally an artificial neuron was aimed to model biological neuron operations: emit signals on its axon when stimuli on its dendrites have reached a threshold. Biologically, a neuron emits pulses, the time between these pulses depends on the stimulation level. Higher is the stimulation, more frequent are the pulses.

In the most used artificial model, the dendrites are represented by an input vector \mathbf{x} and the axon by an output scalar \hat{y} in case of a neuron alone or an output vector $\hat{\mathbf{y}}$ in case of a layer of neurons. This time, the activation of a neuron is depicted by high values on the output, working as an amplitude modulator not as a frequency modulator as its biological counterpart.

Formally, the output of a perceptron (a layer of artificial neurons) is given by

$$\hat{\mathbf{y}} = f(\langle \mathbf{w}, \mathbf{x} \rangle + \mathbf{b}) , \quad (2.2)$$

where f is an activation function, \mathbf{w} the weight of the neuron, \mathbf{b} its bias. \mathbf{w} and \mathbf{b} are the parameters of the model. If $\mathbf{x} \in \mathbb{R}^m$ and $\hat{\mathbf{y}} \in \mathbb{R}^n$, \mathbf{w} belongs to $\mathbb{R}^{n \times m}$ and \mathbf{b} to \mathbb{R}^n . Figure 2.10 represents graphically the equation above, the bias \mathbf{b} is considered as a weight on a fixed input equal to 1. A simplified graphic representation of a perceptron or single layer is represented in Figure 2.11.

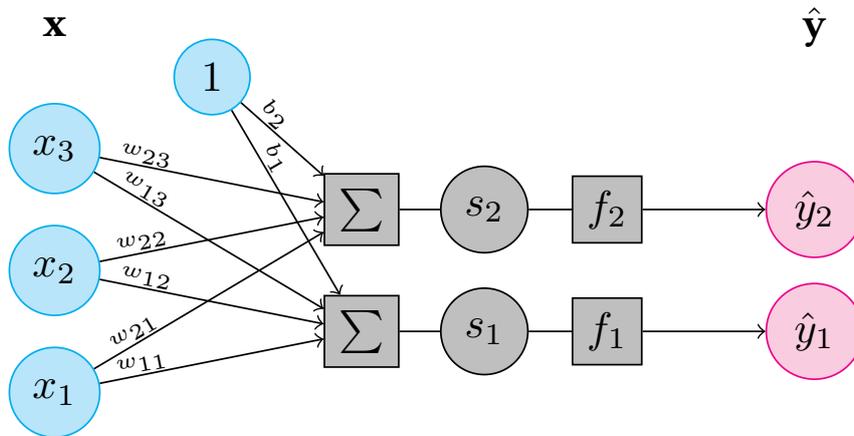


Figure 2.10: The details of a perceptron or a single layer of neurons. It consists in an input representation \mathbf{x} in cyan, an output representation $\hat{\mathbf{y}}$ in magenta and a single computation layer composed of weights \mathbf{w} , bias \mathbf{b} and activation functions f in black.

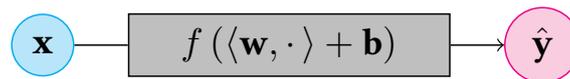


Figure 2.11: Simplified scheme of a perceptron or a single layer of neurons.

The kind of activation function f that is used depends on the targeted task. For example, sigmoid, hyperbolic tangent (Fig. 2.12) or softmax functions are commonly used for classification tasks; hyperbolic tangent or identity functions for regression.

If the activation function is differentiable then the model parameters, \mathbf{w} and \mathbf{b} , are tunable through a gradient descent. For the parameter w_{ij} linking input j to output i , the gradient is composed as followed,

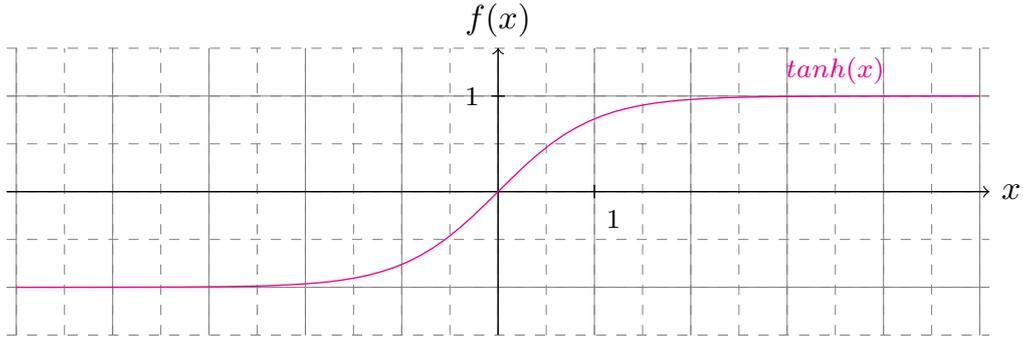


Figure 2.12: Hyperbolic tangent activation function

$$\begin{aligned} \frac{\partial L}{\partial w_{ij}} &= \frac{\partial L}{\partial y_i} \frac{\partial y_i}{\partial s_i} \frac{\partial s_i}{\partial w_{ij}} , \\ &= \frac{\partial L}{\partial y_i} f'_i(s_i) x_j , \end{aligned} \quad (2.3)$$

where L is the loss function, s_i the value of sum before the activation function, and f'_i the derivative of f_i . The gradient of the bias b_i of the output i is computed equivalently by,

$$\frac{\partial L}{\partial b_i} = \frac{\partial L}{\partial y_i} f'_i(s_i) . \quad (2.4)$$

The degenerated model that is composed of identity activation function and a L_2 norm as a loss corresponds to a linear regression learned by a mean square error criterion.

Perceptrons are limited by the fact that they can only solve linearly separable classification problems [MP69]. In order to increase the hypothesis space to other categories of classification problem, we need to chain them in a network. There are infinite possibilities in the topology of such a network. We will restrain ourselves to networks that are organized in layers, they are called feed-forward neural networks or *Multi-Layer Perceptrons* (MLP).

2.2.2 Multi Layer Perceptron

In a Multi-Layer Perceptron (MLP), Perceptrons are stacked together; the output of one perceptron is linked to the input of the following one. The information always flows from one perceptron to the next one. There is no return back, that is why there are also called *Feed-Forward Networks*.

A MLP composed of only two perceptron, as represented in Figure 2.13, should have enough capacity to solve any supervised problems, providing that the dimension of \mathbf{h} may be infinite (an infinite number of hidden units). It has a universal approximation property [Cyb89].

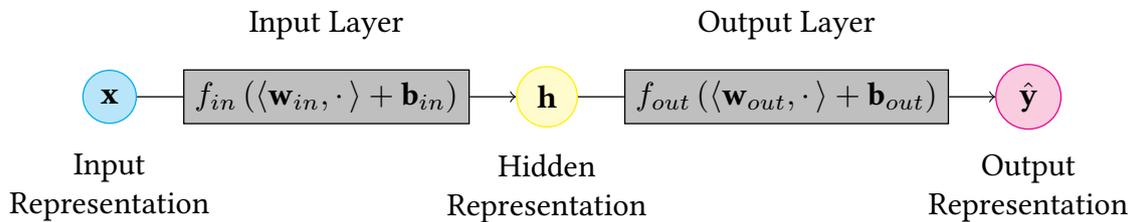


Figure 2.13: An example of a MLP composed of two perceptrons or 2 layers.

Practically to reduce the number of elements in \mathbf{h} meanwhile preserving generalization capacity, a MLP can be composed of more than 2 perceptrons. The Figure 2.14 represents an MLP built out of 3 perceptrons (or 3 layers) and the Figure 2.15 a compact scheme of an MLP with 4 layers.

In order to clarify the vocabulary used in the literature, we will use the following convention in this manuscript:

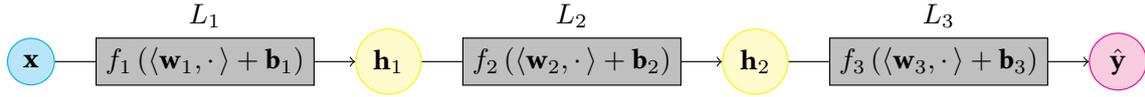


Figure 2.14: An example of a MLP composed of 3 layers.

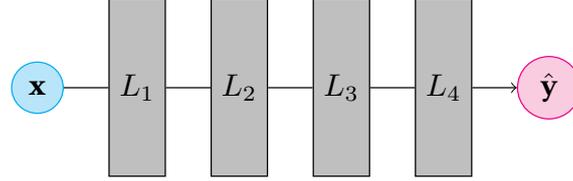


Figure 2.15: Compact scheme of a MLP with many layers.

- a *layer* is the computation component of a feed-forward neural network, it is equivalent to a single perceptron,
- a *representation*, a vector standing for a data state at the input or output of a layer,
- a *unit*, an element of a representation vector,
- a *neuron*, the element (parameters and functions) of a layer that leads to the value of a single output unit.

Thereby, the MLP presented in Figure 2.13 has 3 representations:

1. an *input representation*, usually noted \mathbf{x} , representing features of a sample,
2. a *hidden representation*, usually noted \mathbf{h} , giving the internal hidden/latent state of the network,
3. and an *output representation*, usually noted as $\hat{\mathbf{y}}$, stating for the estimation of the target \mathbf{y} by the network

But, it has only two layers:

1. an *input layer*, which stands for a function f_{in} parameterized by weights \mathbf{w}_{in} and bias b_{in} that computes \mathbf{h} from \mathbf{x} ,
2. an *output layer*, which stands for a function f_{out} parameterized by weights \mathbf{w}_{out} and bias b_{out} that computes $\hat{\mathbf{y}}$ from \mathbf{h} ,

Therefore, in this convention, there is no *hidden layer* in this MLP.

Moreover, we called the layer attached to the input vector \mathbf{x} the *first* or the *deepest layer*, and the layer giving an estimation of the output vector $\hat{\mathbf{y}}$ the *last* or the *highest layer*. When the indexation is negative, it means that it goes backward from the last layer, L_{-1} representing the last layer itself.

As for a single perceptron, a MLP is trained using gradient descent methods. A clever technique called the *gradient back-propagation* takes into account the layered nature of an MLP to iteratively compute the parameter gradient.

We look at a single layer number (l) inside a MLP, as depicted in Figure 2.16. Its inputs or entrances $\mathbf{e}^{(l)}$ are connected to the outputs of the preceding layer $\mathbf{o}^{(l-1)}$ and its outputs $\mathbf{o}^{(l)}$ to the inputs of the next layer $\mathbf{e}^{(l+1)}$.

Let us assume that we already know $\frac{\partial L}{\partial o_i^{(l)}}$, the gradient of the criterion L over the output i of the layer l . The gradient of the parameters for this layer are composed as for the parameters of a perceptron (cf Eq. 2.3),

$$\begin{aligned} \frac{\partial L}{\partial w_{ij}^{(l)}} &= \frac{\partial L}{\partial o_i^{(l)}} \frac{\partial o_i^{(l)}}{\partial s_i^{(l)}} \frac{\partial s_i^{(l)}}{\partial w_{ij}^{(l)}} , \\ &= \frac{\partial L}{\partial o_i^{(l)}} f_i^{\prime(l)} \left(s_i^{(l)} \right) e_j^{(l)} . \end{aligned} \quad (2.5)$$

Let us compute the gradient of the criterion over the input j of the layer l . Please note that the gradient for an input j comes from all the output branches i . Thus we have to sum the gradient over i ,

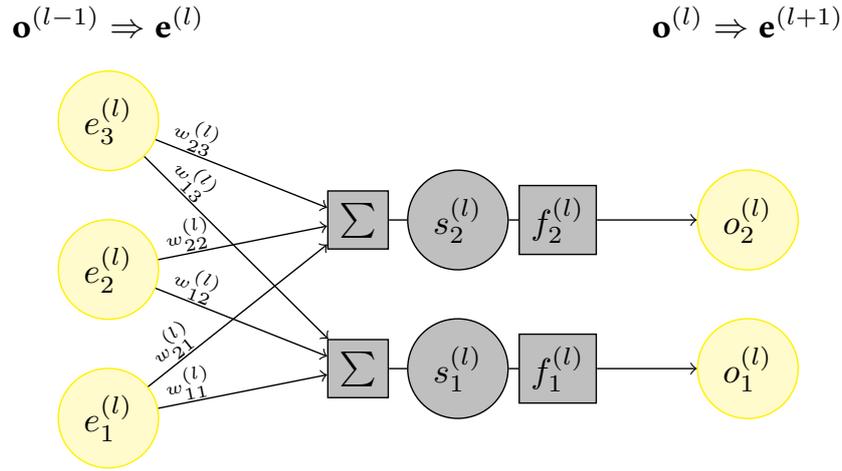


Figure 2.16: A layer (l) of neurons inside a MLP. To simplify the scheme, the bias b is not represented.

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial e_j^{(l)}} &= \sum_i \frac{\partial \mathcal{L}}{\partial o_i^{(l)}} \frac{\partial o_i^{(l)}}{\partial s_i^{(l)}} \frac{\partial s_i^{(l)}}{\partial e_j^{(l)}} , \\ &= \sum_i \frac{\partial \mathcal{L}}{\partial o_i^{(l)}} f_i^{\prime(l)} \left(s_i^{(l)} \right) w_{ij}^{(l)} . \end{aligned} \quad (2.6)$$

As $\mathbf{o}^{(l-1)}$ equals $\mathbf{e}^{(l)}$, we have $\frac{\partial \mathcal{L}}{\partial o_i^{(l-1)}}$ equal to $\frac{\partial \mathcal{L}}{\partial e_j^{(l)}}$. We can then compute $\frac{\partial \mathcal{L}}{\partial w_{ij}^{(l-1)}}$ of the preceding layer, applying Equation 2.5 for the layer $l - 1$. The first iteration of *gradient back-propagation* algorithm starts with $\frac{\partial \mathcal{L}}{\partial o_i^{(l-1)}} = \frac{\partial \mathcal{L}}{\partial y_i}$ to compute $\frac{\partial \mathcal{L}}{\partial w_{ij}^{(l-1)}}$ for the parameters of the last layer. Afterwards, it goes backward down to the first layer, layer by layer.

2.2.3 Recurrent Neural Network (RNN)

A Recurrent Neural Network (RNN) is a neural network where at least an output of a neuron A is connected to an input of a neuron B (possibly through others neurons) and at the same time where an output of B is connected to an input of A (possibly through others neurons also). This definition also works when A and B are the same neuron that is when an output of A is linked to an input of A. As an example, networks based on a MLP scheme where outputs of a layer n return to an input of a layer below n are RNN. Figure 2.17 shows such an example.

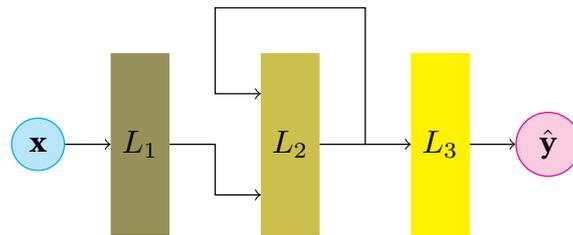


Figure 2.17: A simple RNN, based on a 3 layer MLP. The outputs of layer 2 are concatenated with the outputs of layer 1 to form the input of layer 2.

In a RNN, there is an internal state : a given input \mathbf{x} will not always lead to the same output $\hat{\mathbf{y}}$. RNN are particularly suited for sequence processing, but they require careful training as gradient back-propagation can not be directly applied to them because of the internal state.

Nevertheless, some RNN, such as the one presented in Figure 2.17, can be unfolded to form a MLP that can be trained by gradient back-propagation. The Figure 2.18 presents the result of the unfolding of the 3 layer RNN on a 4 step sequence. Therefore, the unfolded RNN can be interpreted as a 6 layer MLP where its input \mathbf{x} is the concatenation of the all the input \mathbf{x}_t for all the time steps t , its output \mathbf{y} is the concatenation of the all the output \mathbf{y}_t and the unfolded layers U are composed of the concatenation of layers L from the folded RNN. For example, the third layer U_3 is the concatenation of the layers $L_{3,1}$, $L_{2,2}$, $L_{1,3}$ and the identity function. This MLP can be trained by gradient back-propagation taking care of the tied parameters. Indeed, the blocks in the same color are tied together as they correspond to the same layer in the folded representation. That is, when the parameters of $L_{3,1}$ evolve during the gradient descent, the parameters of $L_{3,2}$, $L_{3,3}$ and $L_{3,4}$ evolve in the same manner as they all correspond to the layer L_3 of the folded RNN (Fig. 2.17).

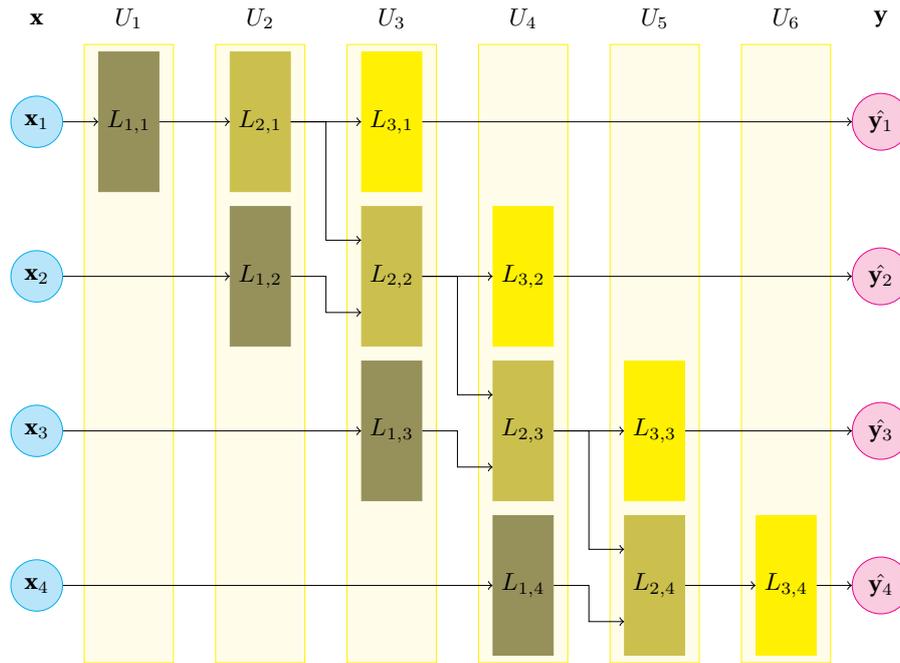


Figure 2.18: The RNN presented in Figure 2.17 unfolded on a 4 step sequence.

Nonetheless, this trick may suffer from the vanishing gradient as we will see later in the section 2.4 dedicated to Deep Learning. Memory units (or *cells*) that composed LSTM [HS97], BLSTM [GS05] and GRU [Cho+14a] can address this gradient lock while preserving long-term dependencies.

Speech to text is an example of task where RNN performs at the state of the art (Figure 2.19).

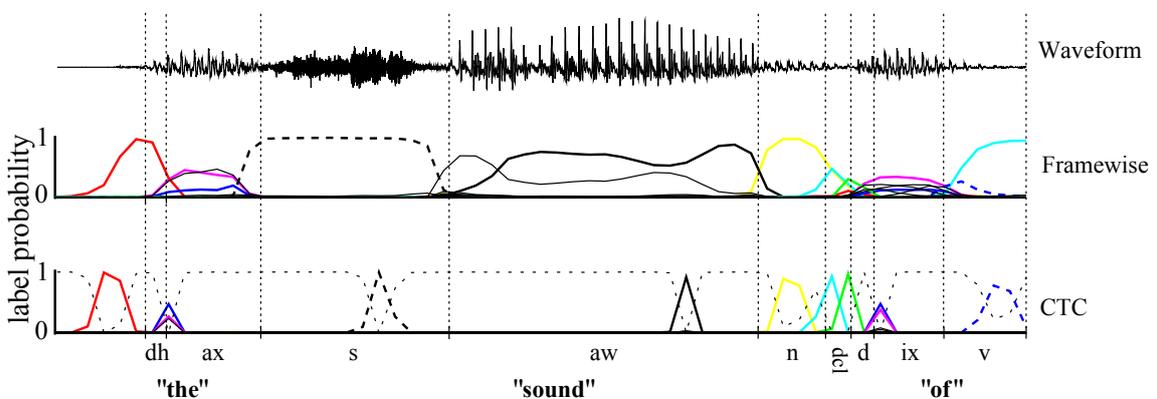


Figure 2.19: Speech to text with RNN (Fig.1 of [Gra+06]).

In this work, we will stick to Feed-Forward Networks or MLP that can directly be trained by back-propagation.

2.3 Auto-Encoder (AE), an ANN for unsupervised learning

In this section we briefly describe what are Auto-Encoders (AE), a special kind of Feed-Forward Networks dedicated to unsupervised learning, especially for representation learning. There are many categories of ANN specialized in unsupervised learning other than AE such as Self Organizing Map (SOM) [Koh82] and Adaptive Resonance Theory (ART) [Gro87]. Nevertheless, we only detail here the building blocks which are needed for the next sections.

2.3.1 Auto-Encoder architecture

An Auto-Encoder (AE) is a special kind of neural network which aims at recovering the input at its output. No label is needed to train it, and so it belongs to the unsupervised learning models.

The simplest AE (Fig. 2.20) is composed of 3 representations (input, latent representation/code and input reconstruction) and 2 layers (an encoder and a decoder). Please note that in this 2-layer case, the dimensions of \mathbf{w}_{dec} , the weights of the decoder, are the reverse of the dimension of \mathbf{w}_{enc} , the weights of the encoder. Optionally, we can decide to tie \mathbf{w}_{dec} to the transposition of \mathbf{w}_{enc} in order to have a PCA-like projection.

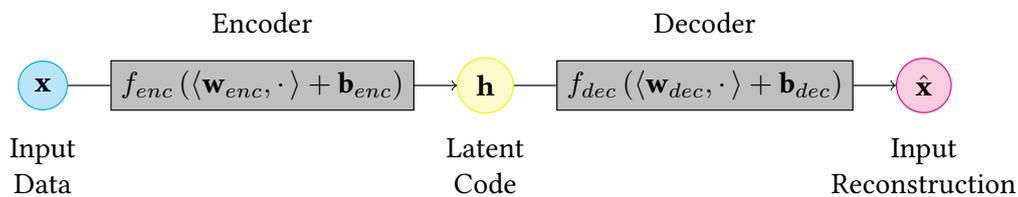


Figure 2.20: A simple 2-layer Auto-Encoder.

An *under-complete* AE is a two layer AE where the latent code, $\mathbf{h} \in \mathbb{R}^n$, has a smaller dimension than the input representation, $\mathbf{x} \in \mathbb{R}^m$, and the input reconstruction, $\hat{\mathbf{x}} \in \mathbb{R}^m$; i.e $m > n$. Such an AE is represented in Figure 2.21 with the typical *diabolo* shape. When the dimension of \mathbf{h} is greater than the one of \mathbf{x} , we have an *over-complete* AE.

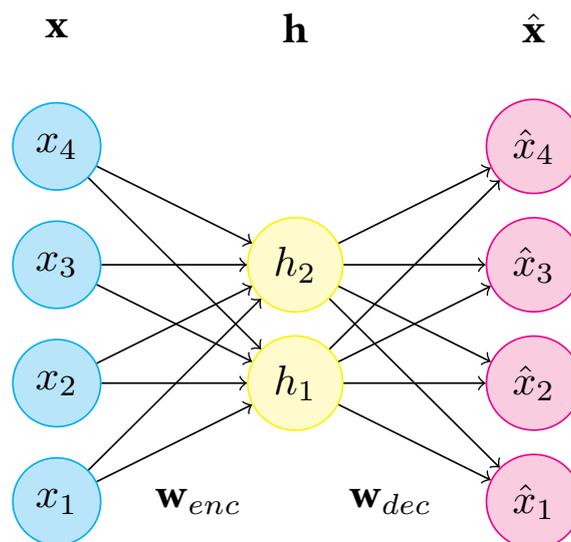


Figure 2.21: An under-complete AE or diabolo network.

The encoder and the decoder can be composed of more than one layer each. In that case, the representation at the output of the last encoder layer is the latent code (Fig.2.22).

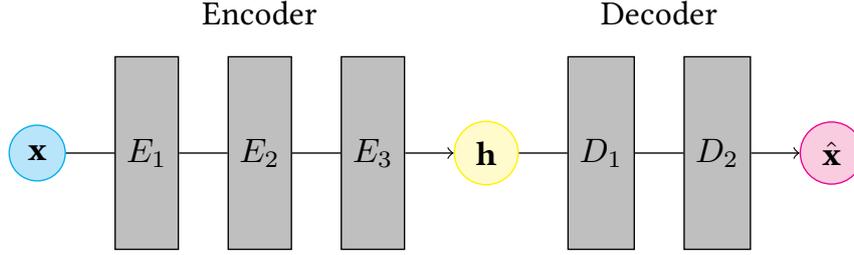


Figure 2.22: An AE with an encoder composed of 3 layers (E_1 to E_3) and a decoder composed of 2 layers (D_1, D_2).

2.3.2 Auto-encoder training

As stated in the preceding sub-section, auto-encoders are learned by trying to recover the input \mathbf{x} , leading to the following empirical risk,

$$R_{emp}(\boldsymbol{\theta}_{enc}, \boldsymbol{\theta}_{dec}) = \frac{1}{\text{card}(\mathcal{U})} \sum_{\mathbf{x} \in \mathcal{U}} L(\mathbf{x}, \hat{\mathbf{x}} = \text{dec}(\text{enc}(\mathbf{x}; \boldsymbol{\theta}_{enc}); \boldsymbol{\theta}_{dec})) \quad , \quad (2.7)$$

where \mathcal{U} is an unlabeled training set, enc the encoder and $\boldsymbol{\theta}_{enc}$ its parameters as well as dec the decoder and $\boldsymbol{\theta}_{dec}$ its parameters. The loss function L is typically the L_2 norm.

As for supervised learning, in order to increase the generalization power, a regularization term is added to the empirical risk leading to the following optimization problem for training the model,

$$\boldsymbol{\theta}^* = \arg \min_{\boldsymbol{\theta}} R_{emp}(\boldsymbol{\theta}) + \lambda \Omega(\boldsymbol{\theta}) \quad , \quad (2.8)$$

where $\boldsymbol{\theta}$ is the concatenation of $\boldsymbol{\theta}_{enc}$ and $\boldsymbol{\theta}_{dec}$.

Besides this explicit regularization scheme, one may use implicit regularization methods: whether we can add noise/transformation to the data or to the model.

In a denoising auto-encoder, a transformation T is introduced between the input representation and the encoder. We still aim at recovering the original input. The empirical risk then becomes,

$$R_{emp} = \frac{1}{\text{card}(\mathcal{U})} \sum_{\mathbf{x} \in \mathcal{U}} L(\mathbf{x}, \hat{\mathbf{x}} = \text{dec}(\text{enc}(T(\mathbf{x})))) \quad . \quad (2.9)$$

For example, the transformation T can consist in adding noise or applying a translation to the input image, if we work on a 2D input representation.

We can also apply noise not only at its input but also inside the model: [Hin+12] proposed to randomly disconnect units in the hidden representation during the training. This technique called *Dropout* is the equivalent to putting 0 to the value of the unit and not applying a gradient descent to its linked parameters (incoming and outgoing), as in Figure 2.23. The authors argue that it helps avoiding perceptron of the encoder to co-adapt, i.e. preventing them from learning the same features and dependencies.

The first trick, i.e. the denoising AE, is used on under-complete AE whereas the later trick, i.e. disconnecting units, is used on over-complete AE.

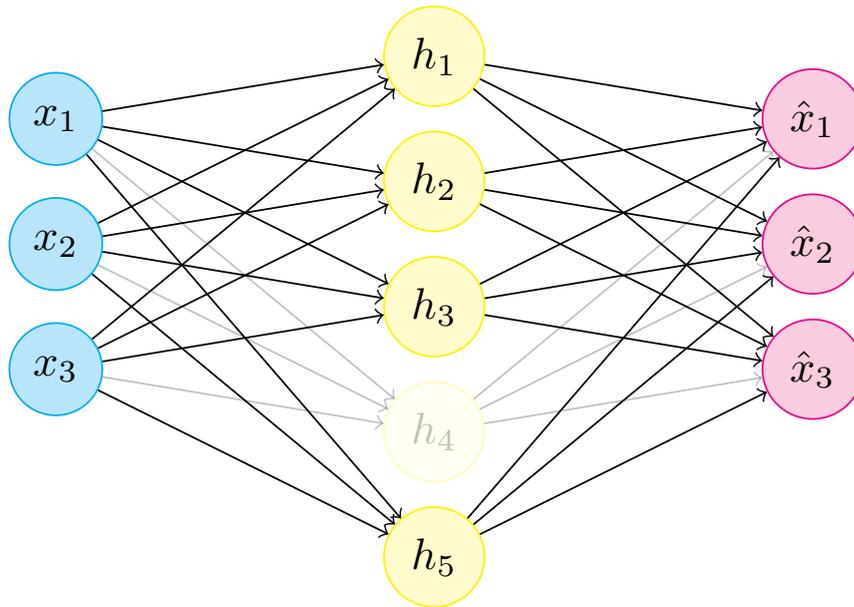


Figure 2.23: An AE with the hidden unit h_4 disconnected.

Auto-encoder application examples

An AE has different purpose such as data compression, dimension reduction denoising, or inpainting [XXC12] (Figure 2.24).

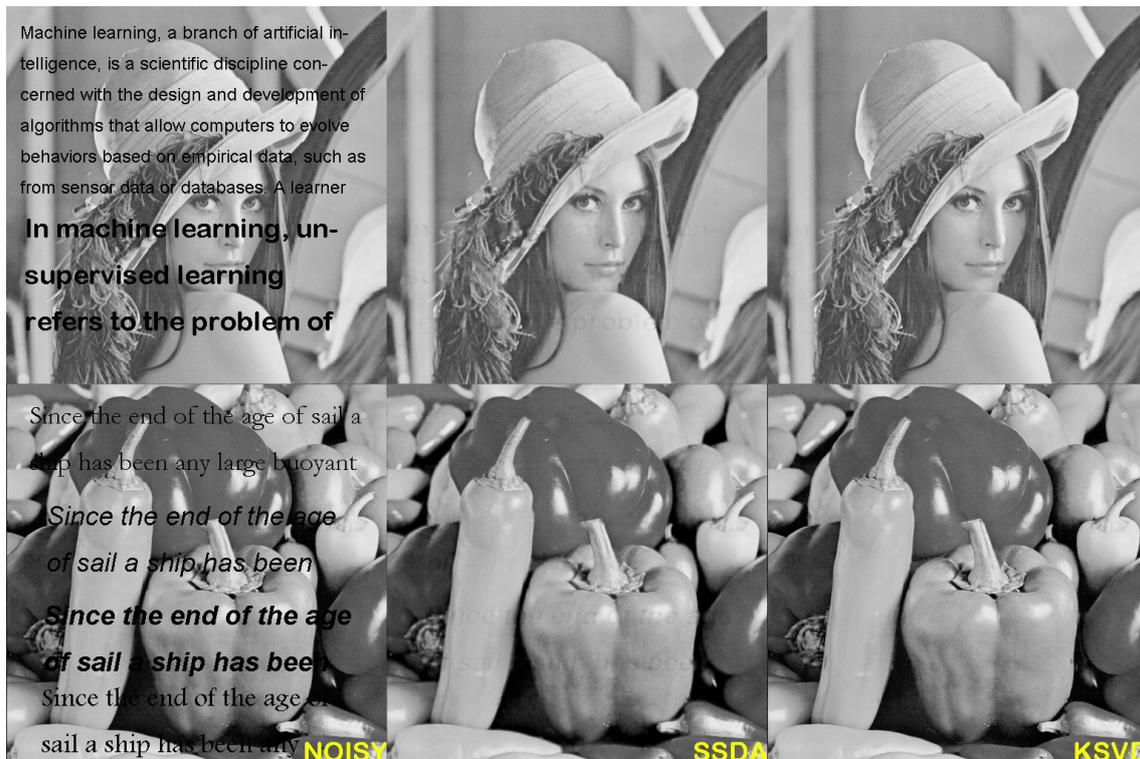


Figure 2.24: Examples of image denoising/inpainting using deep auto-encoders (SSDA) and K Single Value Decomposition (K-SVD) (Fig.3 of [XXC12]).

2.4 Deep Learning

Since the availability of Graphical Processing Unit (GPU) and of open API/libraries [Ber+10; Aba+15; The16; Pas+17] to take advantage of their matrix computation power, Machine Learning based on Neural Networks has exploded. In this trend, a lot of research projects, commercial products, civil society debates refer to *Deep Learning* to gather attention, putting aside past Artificial intelligence and Machine Learning. In this section, we will first try to debunk what *Deep Learning* is, show their architectural particularities and how they can be trained.

As for the precedent sections, the following paragraphs do not claim to be exhaustive but present the needed information to understand Deep Neural Network (DNN) presented in the published articles appended to this manuscript.

2.4.1 Definition

To my point of view, at least three definitions of the Deep Learning can be given: whether we take into account the way the network is built, the way the network is taught, or the way the data are represented.

In the first first definition, we look at the architecture of the network (Fig. 2.25a). A network is said to be *deep* when it is composed of more than two layers. At the opposite a one or two layer network is said to be *shallow*.

In the second definition, a network is said to be *deep* when a *vanishing or exploding gradient* arises from its training. This kind of problems appears due to numerical imprecision inherent to the finite nature of computer calculation. When back-propagating the gradient to deeper layers, differentiation and numerical precision errors are accumulated (Fig. 2.25b). These errors may lead to low or high values of the gradient, preventing it from learning or converging.

In the third definition which has my preference, a deep framework is a framework that address directly in itself the data representation step of machine learning, avoiding handcrafted feature extraction or a separate unsupervised representation learning (Fig. 2.25c). Consequently, raw or lightly processed data are directly fed to a deep network.

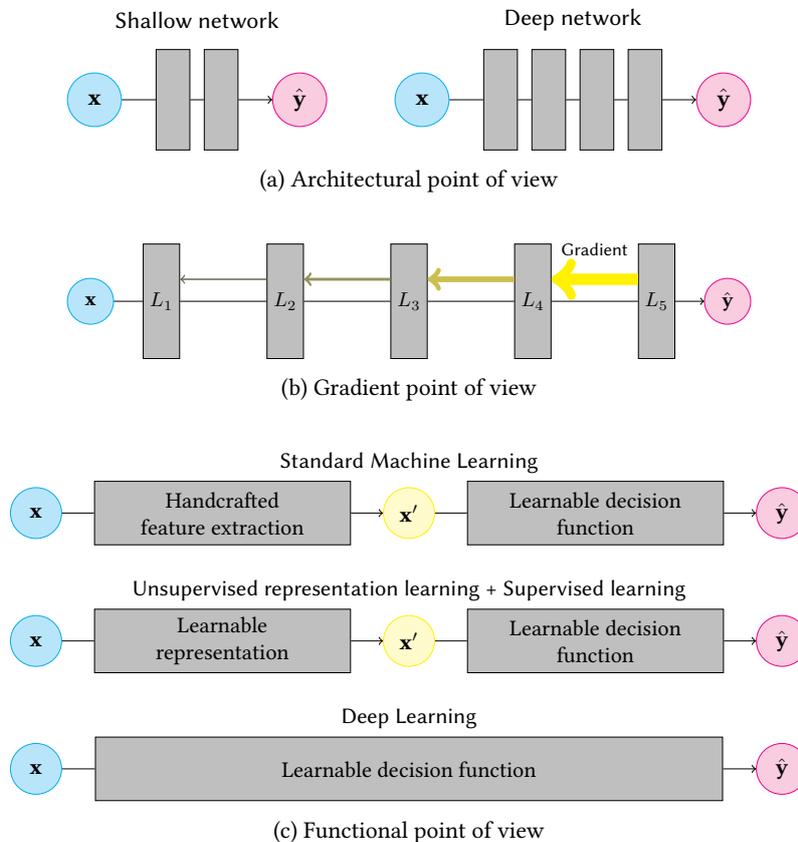


Figure 2.25: Different definitions of Deep Learning

Focus on the vanishing gradient problem

Along the training, a neuron tends to push its output toward saturated parts of the activation function, far from the decision boundary (Fig. 2.26). For samples that reach these parts, the gradient is low and prone to numerical errors. Ultimately it may lead to a vanishing gradient problem when back-propagating and accumulating gradients over several layers.

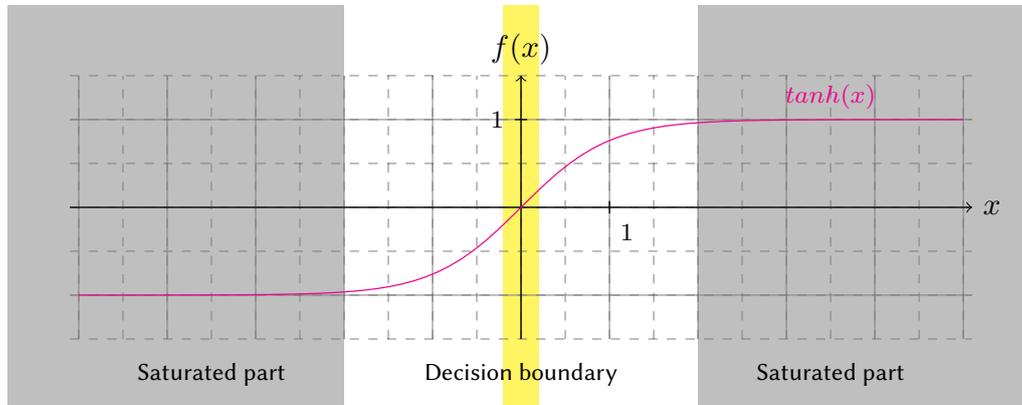


Figure 2.26: Hyperbolic tangent activation function with saturated parts highlighted in gray.

2.4.2 Tips and tricks to avoid gradient problems

Due to the fact that they process directly raw data and to the vanishing/exploding gradient, deep models with standard layers (dense plus sigmoid/tanh activations) are hardly trained directly by a gradient back-propagation. They need modifications to the nature of their layers or to the training process. We will describe these tips and tricks in the following paragraphs.

AE Pretraining

In order to tackle the vanishing gradient problem [HS06] propose to initialize DNN by learning unsupervisedly stacked auto-encoders (AE) from the input space. The method consists in two steps, a *pre-training* which builds the network from encoders of AE trained in an unsupervised manner, followed by a *fine-tuning*, i.e. a supervised training of the full network on the actual targeted task.

Algorithm 1 presents a simplified version of the method. It does not take into account different layer types or different training losses between AE and the full network. Some functions are supposed to be known and not detailed in the algorithm:

MLPFORWARD is a primitive that does a forward pass on a MLP taking as arguments the list of parameters (one element per layer) and the network input. For example $\text{MLPFORWARD}([\mathbf{w}_1, \mathbf{w}_2], \mathbf{X})$ stands for the evaluation of a 2-layer MLP on the data set \mathbf{X} .

MLPTRAIN is also a primitive that trains a MLP taking as arguments the list of initial parameters (one element per layer), the network input and the desired target. For example $\text{MLPTRAIN}([\mathbf{w}_1, \mathbf{w}_2], \mathbf{X}, \mathbf{Y})$ stands for the training a 2-layer MLP on the training data set (\mathbf{X}, \mathbf{Y}) .

Moreover, $\text{MLPTRAIN}([\mathbf{w}_{enc}, \mathbf{w}_{dec}], \mathbf{X}, \mathbf{X})$ with $\text{shape}(\mathbf{w}_{dec}) = \text{shape}(\mathbf{w}_{enc}^T)$ simulates the training of a 2-layer auto-encoder on \mathbf{X} .

Algorithm 1 Simplified algorithm of stacked auto-encoders.

Input: \mathbf{X} , a training feature set of size examples \times features

Input: \mathbf{Y} , a corresponding training label set of size examples \times labels

Input: N , the number of layers of the deep network

Input: N_{input} , the number of input layers to be pre-trained ($N_{\text{input}} < N$)

Output: $[\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_N]$, the parameters (weights and bias) for all the N layers

Pre-training

Randomly initialize $[\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_N]$

$\mathbf{R} \leftarrow \mathbf{X}$

for $i \leftarrow 1..N_{\text{input}}$ **do**

 {Training an AE on \mathbf{R} and keeps its encoding parameters}

$[\mathbf{w}_i, \mathbf{w}_{\text{dummy}}] \leftarrow \text{MLPTRAIN}([\mathbf{w}_i, \mathbf{w}_i^T], \mathbf{R}, \mathbf{R})$

 Drop $\mathbf{w}_{\text{dummy}}$ that corresponds to the decoder part

$\mathbf{R} \leftarrow \text{MLPFORWARD}([\mathbf{w}_i], \mathbf{R})$

end for

Fine-tuning

$[\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_N] \leftarrow \text{MLPTRAIN}([\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_N], \mathbf{X}, \mathbf{Y})$

Activation functions

We have seen that the vanishing gradient problem arises when neuron outputs lie in the saturated part of the activation function where the gradient is low. One solution is to use non-saturating functions, or functions that are only saturating in one direction.

In [GBB11], the authors proved that the *rectifier* activation function, $f(x) = \max(0, x)$, leads to fewer gradient problem and enables the learning of deeper network without the pre-training trick (Fig. 2.27, cyan curve). Let's note that the often used term *ReLU* comes from *REctified Linear Unit*. In addition to its non saturating behavior on one direction, it also provides a sparse initialization. Indeed, for randomly initialized weights and for a particular sample, half of the neurons are not activated. There are progressively introduced along the training by changes in underlying layers. This activation function has been already in use since [HSS03] based on mathematical motivation from its non-symmetry. Nevertheless, It has a discontinuity on 0 that must be addressed for gradient descent. In practice, for values closed to this point, an arbitrary derivative between 0 and 1 is randomly chosen.

An other possibility to manage the discontinuity is to relax the ReLU by a differentiable version: the *softplus* (Fig. 2.27, magenta curve). It has the same properties except the sparse initialization. Among other activation functions adapted to tackle vanishing gradient, we can find the *exponential linear unit* (ELU), the *softsign* or the *LeakyReLU* which displays a small slope for negative values.

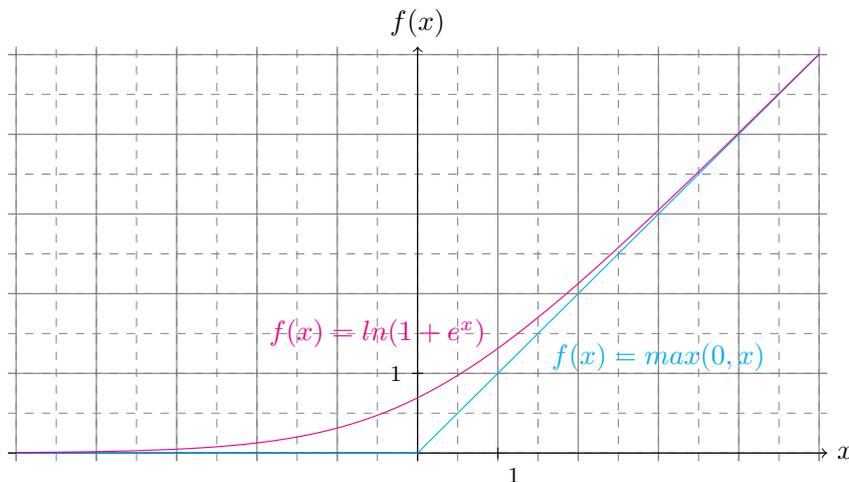


Figure 2.27: The rectifier $f(x) = \max(0, x)$ activation function (cyan curve) and its differentiable relaxation, the softplus $f(x) = \ln(1 + e^x)$ (magenta curve).

Normalization

An other possibility to prevent low gradient is to retain samples in the non-saturated part of the activation, usually around the decision boundary (Fig. 2.26). When one normalizes a dataset prior the training to a unit normal distribution, it is exactly what one is doing for the first layer. Why not apply the same trick in between each layer as in Figure 2.28 ?

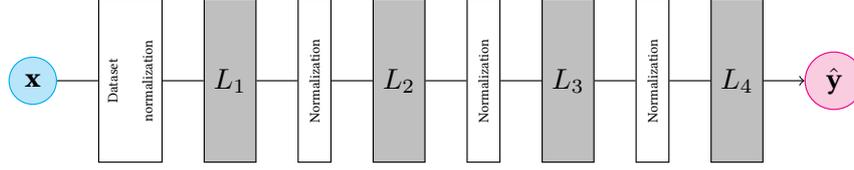


Figure 2.28: A DNN with normalization layers.

In a *Batch Normalization* layer [IS15], each component h of the representation \mathbf{h} is rescaled using the mean $\mu_{\mathcal{B}}$ and the variance $\sigma_{\mathcal{B}}^2$ of each sample batch \mathcal{B} by the following formula,

$$h' = \frac{h - \mu_{\mathcal{B}}(h)}{\sqrt{\sigma_{\mathcal{B}}^2(h) + \epsilon}}, \quad (2.10)$$

where h' the component of the layer's output corresponding to the input component h . ϵ is a small value to prevent from a numerical explosion of the division.

The statistical indicators $\mu_{\mathcal{B}}(h)$ and $\sigma_{\mathcal{B}}^2(h)$ depend on h and are not parameters of the layer ! Thus it is not a simple linear transformation and the back-propagation to h must take into account the two paths going through the indicators. These two paths are composing the second term of the equation of the gradient over h in the following equation, the first term coming from the linear scaling,

$$\frac{\partial \mathcal{L}}{\partial h} = \frac{\partial \mathcal{L}}{\partial h'} \frac{1}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} + \frac{1}{m} \left(\frac{\partial \mathcal{L}}{\partial \sigma_{\mathcal{B}}^2} 2(h - \mu_{\mathcal{B}}) + \frac{\partial \mathcal{L}}{\partial \mu_{\mathcal{B}}} \right) \quad (2.11)$$

given the gradient of the mean,

$$\frac{\partial \mathcal{L}}{\partial \sigma_{\mathcal{B}}^2} = \frac{-1}{2} (\sigma_{\mathcal{B}}^2 + \epsilon)^{-3/2} \sum_{i=1}^m \frac{\partial \mathcal{L}}{\partial h'_i} (h_i - \mu_{\mathcal{B}}), \quad (2.12)$$

where m is the number of examples in \mathcal{B} , and given the gradient of the variance,

$$\frac{\partial \mathcal{L}}{\partial \mu_{\mathcal{B}}} = -\frac{1}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \sum_{i=1}^m \frac{\partial \mathcal{L}}{\partial h'_i} - \frac{1}{m} \frac{\partial \mathcal{L}}{\partial \sigma_{\mathcal{B}}^2} \sum_{i=1}^m 2(h_i - \mu_{\mathcal{B}}). \quad (2.13)$$

At the end of the training, a mean μ and a variance σ^2 are computed on the full train set. These two values will replace $\mu_{\mathcal{B}}$ and $\sigma_{\mathcal{B}}^2$ respectively at decision stages (validation or test).

Here each component of \mathbf{h} is computed separately without taking into account inner dependencies. A straight forward extension is to do the normalization to a unit multi-variate normal distribution taking into account covariances of the components. Nevertheless, back-propagation is here tricky as the covariance matrix has to be inverted.

Different normalization strategies arise from different mean and variance computation choices [WH18]. Figure 2.29 illustrates possible strategies on a multi-channel image batch.

In *Batch Normalization*, the statistics are computed along the batch, feature by feature (Fig. 2.29a). This strategy can be extended to a multivariate normalization. A covariance matrix is computed on features of the same channel (Fig. 2.29b, light cyan), but the mean is still performed feature by feature (dark cyan).

In a *Layer Normalization* mean and variance are computed along all the features of a giving sample but separately sample by sample (Fig. 2.29c).

On multi-channel images, the layer normalization mixes all channels. The *Instance Normalization* takes the same principle as the layer normalization but this time computing the statistics channel by channel independently (Fig. 2.29d).

The *Group Normalization* is a trade-off between layer and instance normalization where channels can be grouped together (Fig. 2.29e).

Thereby, the batch normalization corresponds to normalizing a pixel $[i, j]$ with pixels at the same position $[i, j]$ on other images in the batch without taking into account pixels located elsewhere in the same image; whereas the layer/instance/group normalizations perform the normalization on pixels of the same image wherever they are, without taking into account other images.

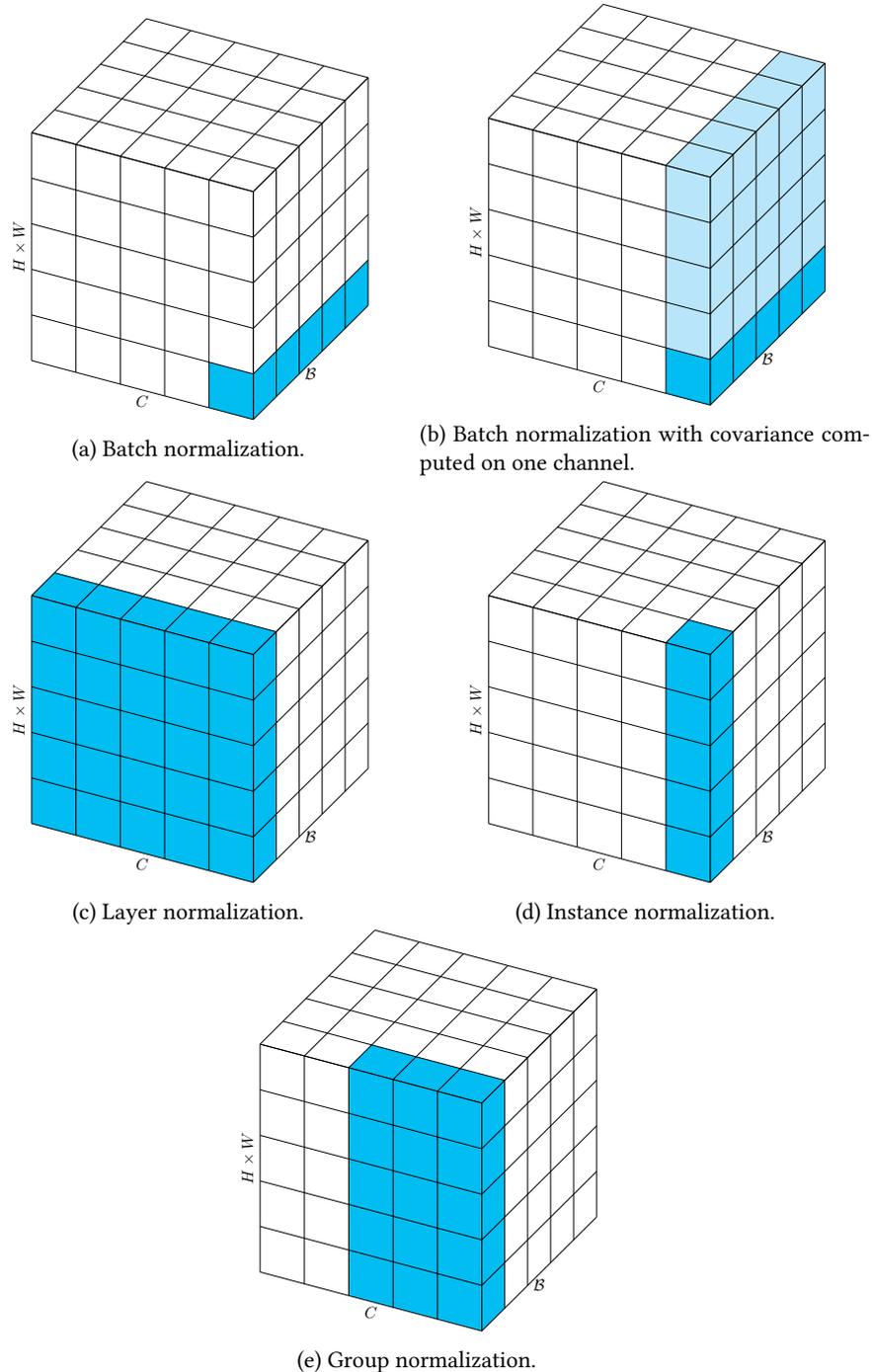


Figure 2.29: Different normalization strategies on a batch of multi-channel images, figure inspired by [WH18]. A first axis represents the topology of the sample, i.e. its height (H) times its width (W) in the case of an image, a second axis the channels (C), e.g. RGB in a color image, and a third axis the batch samples (B). The cyan items represent the part of the data that is used to compute the mean and the variance.

Skip connection

To help the gradient flow to deeper layers, one can add a direct connection from a layer to layers deeper than its preceding layer. Figure 2.30 shows such a typical *skip connection* from a layer $l - 1$ to a layer $l + 1$; the layer l has been skipped. The output $\mathbf{o}^{(l+1)}$ of the layer $l + 1$ is then given by

$$\mathbf{o}^{(l+1)} = f(\langle \mathbf{w}_a, \mathbf{o}^{(l)} \rangle + \langle \mathbf{w}_b, \mathbf{o}^{(l-1)} \rangle + \mathbf{b}) , \quad (2.14)$$

where $\mathbf{o}^{(l)}$ is the output of layer $l - 1$, and $\mathbf{o}^{(l-1)}$ the output of layer $l - 1$, as well as \mathbf{w}_a is the weight matrix from layer l to $l + 1$ and \mathbf{w}_b from $l - 1$ to l .

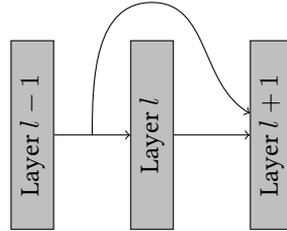


Figure 2.30: An example of skip connection over layer l .

When the layer l has the same input and output size, we can add its input directly to its output to get a *Residual Layer* (Fig. 2.31). Indeed in this setup a residual layer models the difference or *residue* between its input and output [He+16]. This is a special case of skip connection with \mathbf{w}_a and \mathbf{w}_b equal and tied together to \mathbf{w} ,

$$\mathbf{o}^{(l+1)} = f(\langle \mathbf{w}, \mathbf{o}^{(l)} + \mathbf{o}^{(l-1)} \rangle + \mathbf{b}) . \quad (2.15)$$

Networks built with residual layers are called *Residual Neural Networks*.

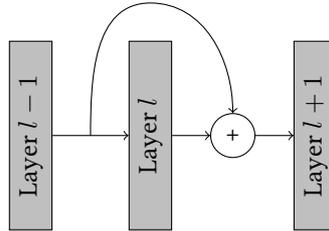


Figure 2.31: An example of residual layer.

When inside a group of layers, all the layers are directly connected to the higher layers of the group, this group is called a *dense block* (Fig. 2.32). A *densely connected network* [Hua+17] is the succession of several dense blocks.

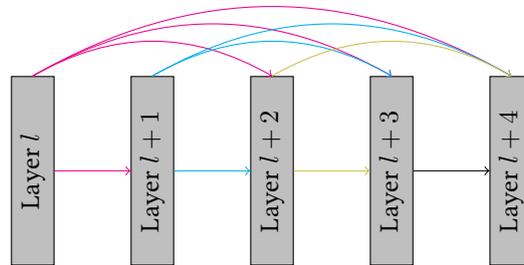
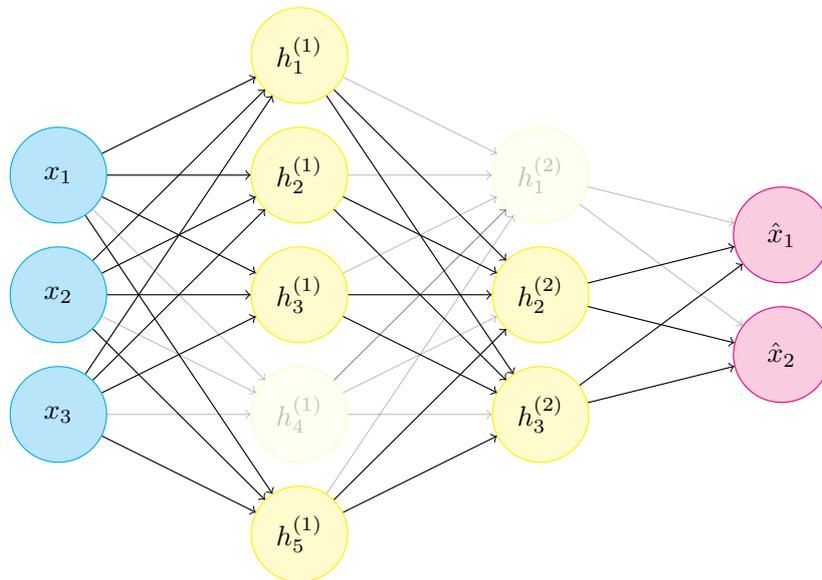


Figure 2.32: An example of dense block.

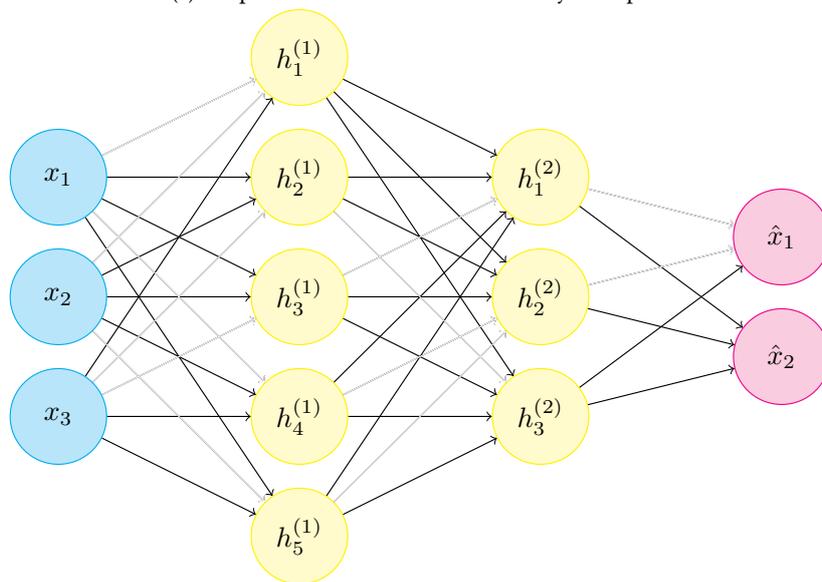
Dropout and Dropconnect

Following the idea presented in [Hin+12] for auto-encoders and shallow networks (c.f. sub-section 2.3.2), in [Sri+14] the same authors extend the *Dropout* idea to deep architecture. Formally, during the training phase, a unit is put to 0 at a probability p (Fig. 2.33a). Meanwhile during decision phase (validation or test), the output of the unit is multiplied by p .

Dropconnect [Wan+13] is based on the same principle but this time the disconnection operates on a link (a weight) not a unit (Fig. 2.33b).



(a) Dropout: disconnections on units/layer outputs



(b) Dropconnect: disconnection on links/weights

Figure 2.33: Different disconnections strategies

These two techniques reduce artificially the capacity of a deep network during training to prevent over-fitting but they don't address directly the vanishing gradient problem. They perform an implicit regularization.

2.4.3 Convolutional Neural Networks

Convolutional Neural Networks (CNN) are a special class of DNN that particularly suit image processing and more generally n-D signals. They take into account the topology of images and, at a layer scale, mimic the behaviour of a learnable convolution filter.

In this architecture, there are two specific layers: *convolution layers* which actually perform convolutions and *pooling layers* which down-sample the image space. A convolution layer is in fact a regularized version of a standard layer : not all input units are connected to output units and neurons of the same layer share their parameters together. By its sparse architecture and its parameter sharing, a convolution layer is less prone to overfitting and gradient problems. In the CNN point of view, a standard layer is called a *fully connected layer* or *dense layer*, in opposition to a convolution layer.

Convolution layer

Mathematically a convolutional layer performs a cross-correlation between its input representation and a *kernel*. It consists in moving the kernel around the input representation in a sliding manner and for each possible position performing a dot product between the kernel and the underlying part of the input representation. The result of the dot product is affected to the pixel of output representation corresponding to the kernel position (Fig. 2.34).

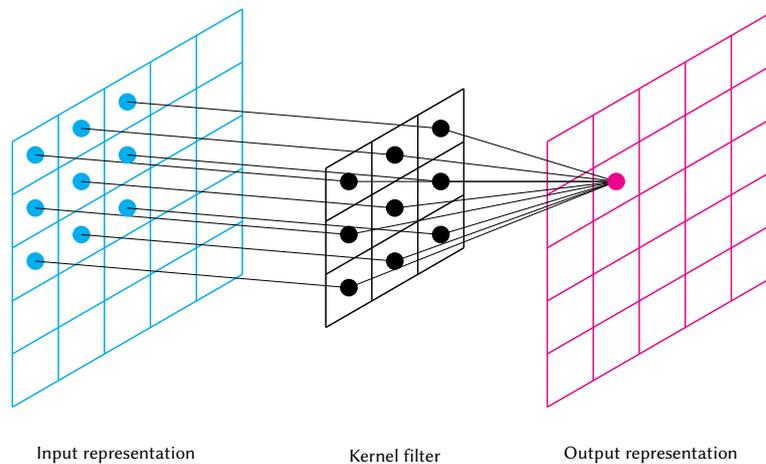


Figure 2.34: A convolutional layer on 1 channel 2D image with a convolution kernel of size 3×3 .

The sliding window can overlap with its preceding position depending on the movement drift call the *stride*. If the stride is lower than the size of the kernel, two consecutive frames share a part of their input (Fig. 2.35). If the stride is equal to the kernel size or greater, they do not overlap.

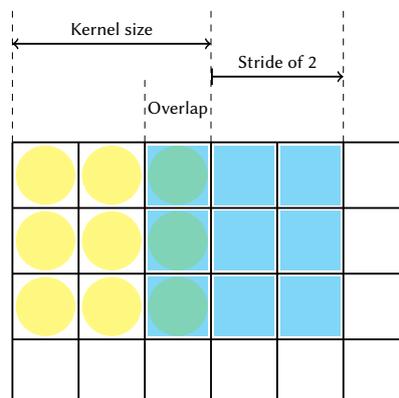


Figure 2.35: Two consecutive frames (yellow circles followed by cyan rectangles) can overlap if the stride is lower than the kernel size.

In term of CNN, an image channel is called a *feature map*. The input can have more than one feature map (Fig. 2.37a), the output representation can also have multiple output channels (Fig. 2.37b); moreover

the convolution may address both multiple input and output maps at the same time. Beware, all the input channels are taken into account for each output maps.

Formally, if the kernel has a size of $K \times L$ in the image 2D topology, I the number of input feature maps, O the number of output channels then the kernel parameters \mathbf{W} have a total of $K \times L \times I \times O$ elements. For an input image \mathbf{X} and output image \mathbf{Y} , the value of the output pixel is given by

$$\mathbf{Y}[m, n, o] = f \left(\sum_{i=1}^I \sum_{k=1}^K \sum_{l=1}^L \mathbf{W}[k, l, i, o] \mathbf{X} \left[m + k - \frac{K-1}{2}, n + l - \frac{L-1}{2}, i \right] + \mathbf{B}[m, n, o] \right), \quad (2.16)$$

where $i \in [1, \dots, I]$ is an input feature map, $o \in [1, \dots, O]$ an output feature map, $[m, n]$ the pixel position in the output representation, f an activation function, and \mathbf{B} the bias. In order, to simplify the equation it is assumed that the kernel sizes K and L are odd, that the stride is 1, and that the input and output representation sizes are compatible, i.e. no padding has to be added.

As for a standard layer, \mathbf{W} and \mathbf{B} , the convolutional layer parameters, are tunable by gradient descent. \mathbf{B} can have different values for each pixel or it can be shared between all pixels of the same output feature map or between all pixels of all output feature maps.

A convolutional layer drastically reduces the number of parameters. Let's take as example an RGB input image $512 \times 512 \times 3$ that we want to process to a $512 \times 512 \times 1$ output image. With a fully connected layer it would mean $(512 \times 512 \times 3) \times (512 \times 512)$ parameters for \mathbf{W} and 512×512 parameters for \mathbf{B} , more than 200G parameters. With a convolutional layer consisting in a 3×3 kernel, there are only $3 \times 3 \times 3$ parameters for \mathbf{W} and 512×512 for the bias as it is specific to each convolution position, leading to less than 265k parameters. With a shared bias configuration this can be reduced to only 1 parameter for \mathbf{B} , so only 10 parameters for this layer!

When stacking convolutional layers, all the input pixel from the first layer that influence the output of a pixel in the last layer is called the *field of perception*. This field of perception can be increased by using a *dilated convolution layer* which contains a sparse kernel where non zero values are separated by a dilation rate (Fig. 2.36). Enlarging the field of perception enables to take into account a larger context for the decision.

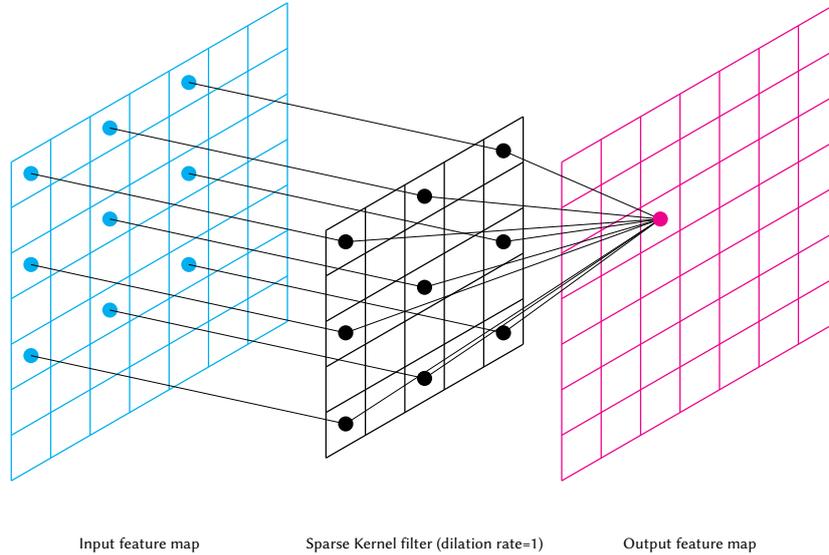
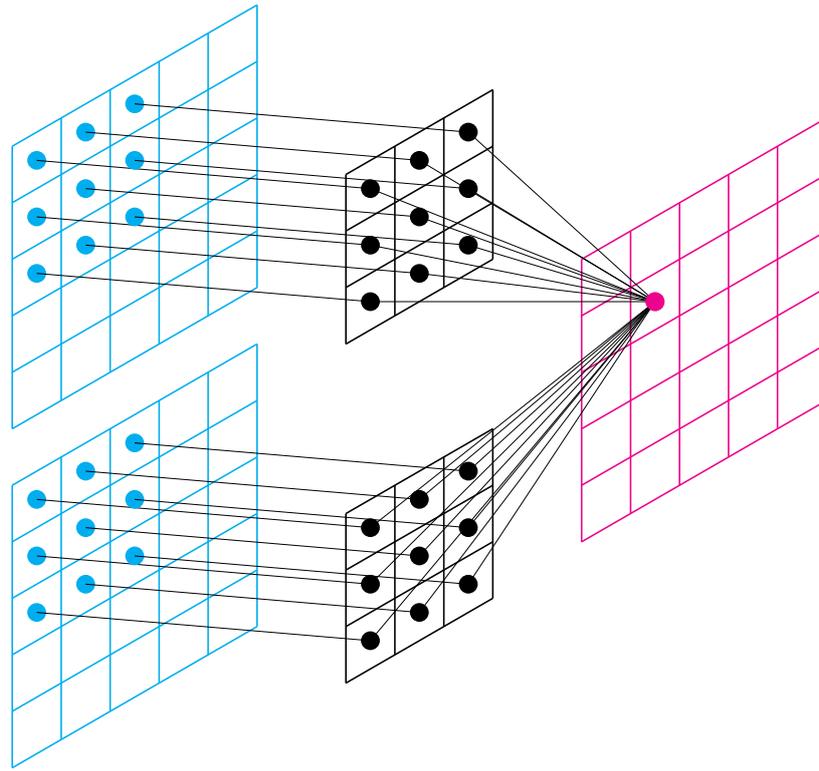
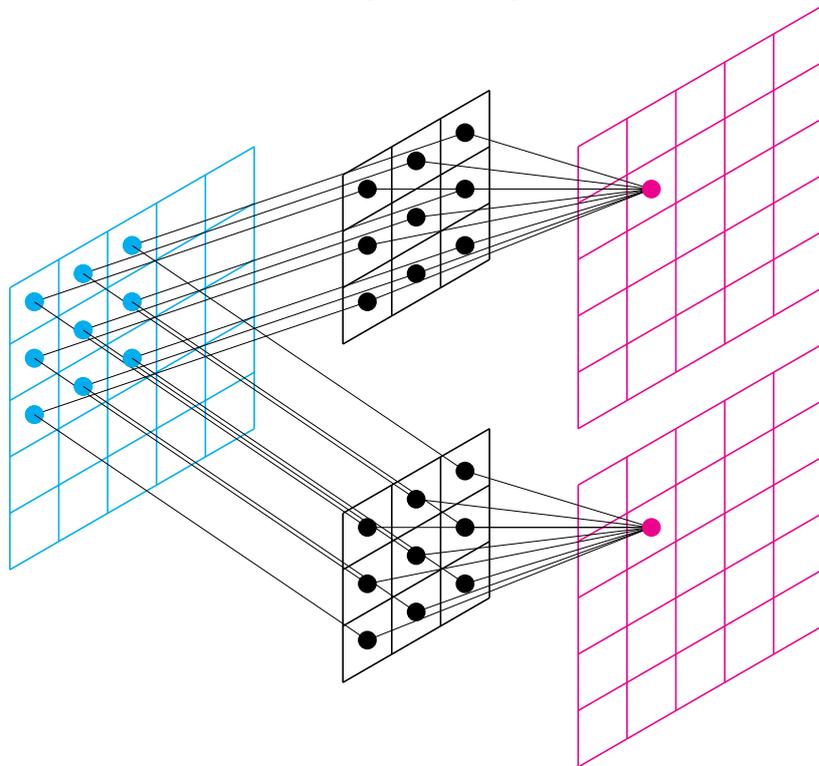


Figure 2.36: A dilated convolutional layer with a dilation rate of 1.



$M \times N \times 2$ input feature maps $3 \times 3 \times 2 \times 1$ kernel filter $M \times N \times 1$ output feature map

(a) Two input feature maps.



$M \times N \times 1$ input feature map $3 \times 3 \times 1 \times 2$ kernel filter $M \times N \times 2$ output feature maps

(b) Two output feature maps.

Figure 2.37: Multiple feature map cases.

Pooling layer

A pooling layer performs a down-sampling on its input representation by splitting it into non-overlapping partitions and then computing a sumup, typically a max operation, on each of these sub-regions.

By doing so, the network loses in localization precision but gains in translation invariance. It is also a way of controlling overfitting and gradient problem by reducing the number of parameters of higher layers.

A pooling layer does not have in itself parameters tunable by gradient back-propagation.

In Figure 2.38, we can see a pooling layer performing a max operation, on a 2×2 sub-region with a stride of 2. It results in a down-sampling by a factor 2, i.e. the output representation is 2 times smaller in height and width, resulting to 4 times less elements than in the input representation.

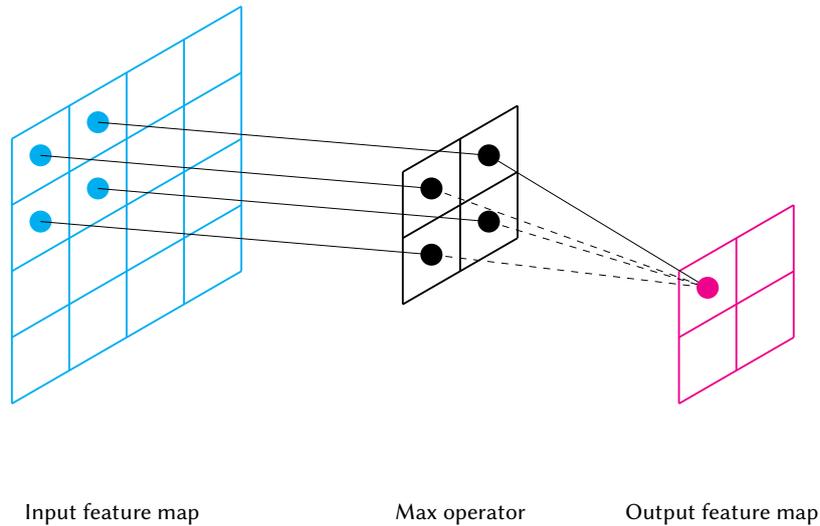


Figure 2.38: A pooling layer down-sampling its input by a factor 2.

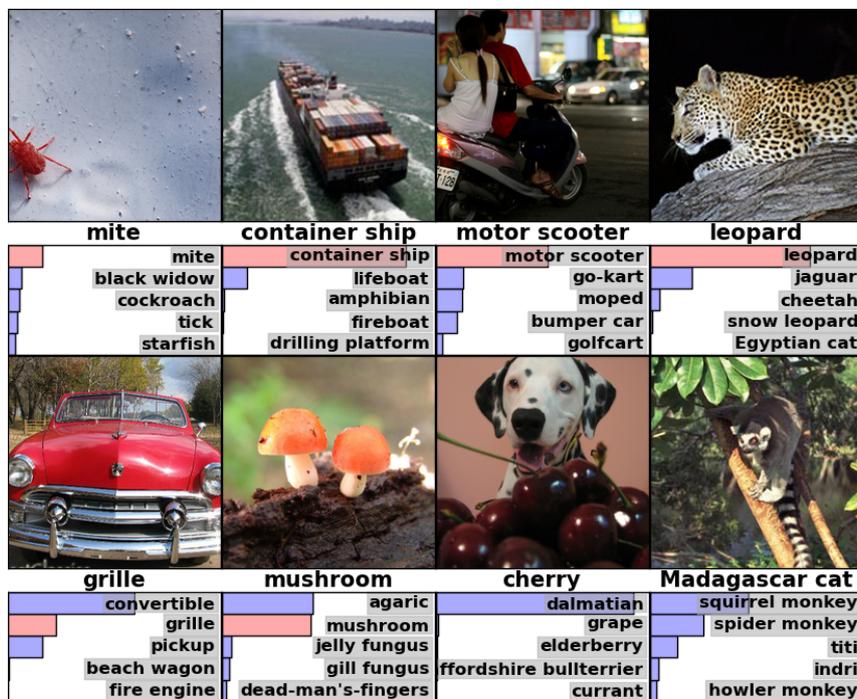


Figure 2.39: One of the first DNN results on Imagenet competition (Fig.4 of [KSH12]).

General architecture

As depicted in Figure 2.40, on its basic form, the architecture of a CNN consists in,

- Several *convolutional blocks*,
- 1, 2 or more fully connected / dense layers.

The composition of each *convolutional block* may vary, but the standard set up is the concatenation of

- a convolutional layer,
- a ReLU activation layer,
- a pooling layer.

One may find normalization layer in between convolutional blocks, as well as skip connections.

The convolutional blocks build higher semantic representation of the data while preserving a part of the localization. The fully connected layers lose the topology and take a global decision.

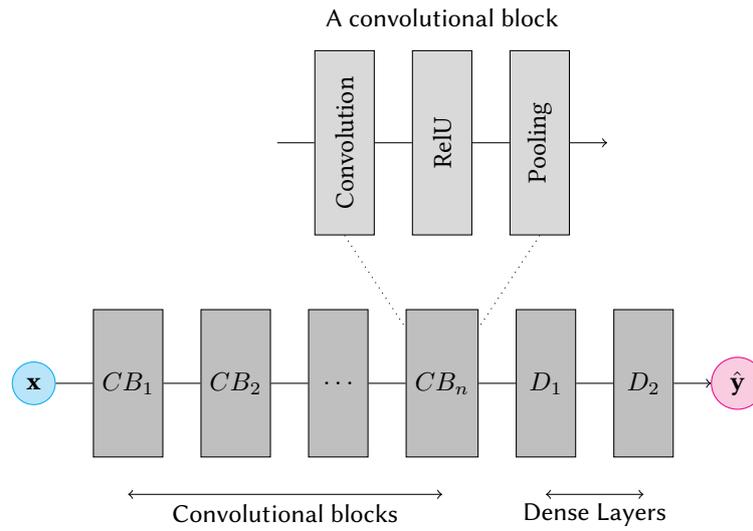


Figure 2.40: A basic form of convolutional neural network

Deep architecture with convolutional blocks is perfectly suited for image classification task such as the ImageNet competition where they have overtopped all other competitors (Figure 2.39).

Fully Convolutional Networks

A *Fully Convolutional Network* (FCN) [LSD15] is a CNN which does not contain any dense layer but only convolutional blocks. The targeted loss is directly plugged to the activation layer of the last convolutional block. It takes a decision for each pixel of the output representation (semantic segmentation). Moreover, it has the advantage of having very few parameters.

Nonetheless, with no dense layer the global dependencies are harder to take into account, and the decision at the output pixel level is limited to its field of perception. To increase the field of perception and catch more global information, one may use dilated convolutional layers as presented above.

2.5 Deep Generative Models

Deep Generative models are deep architecture dedicated to the generation problem. Let be a training set \mathcal{U} that contains real samples \mathbf{x} thrown from an unknown distribution P . Generally \mathbf{x} lies in a high dimension space such as images. The purpose of a generative model is to generate artificial samples $\tilde{\mathbf{x}}$ from a built distribution Q . This distribution Q should be as close as possible to the unknown distribution P (Figure 2.41).

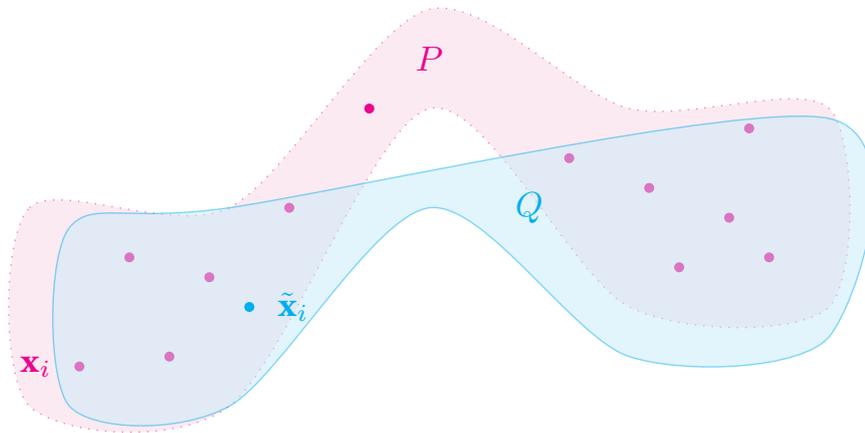


Figure 2.41: The distribution Q of the deep generative model should modelize the distribution P only knowing samples \mathbf{x}_i .

The architecture of a deep generative model consists in throwing a low dimension noise \mathbf{z} from a known distribution Z , typically a uniform or normal distribution and then feeding the input of deep network with that noise. The DNN is built to have a high dimension output representation in the same space as the real samples \mathbf{x} thrown from the unknown distribution P . Thus the input of the DNN, \mathbf{z} is in low dimension whereas its output $\tilde{\mathbf{x}}$ is in high dimension (Figure 2.42).

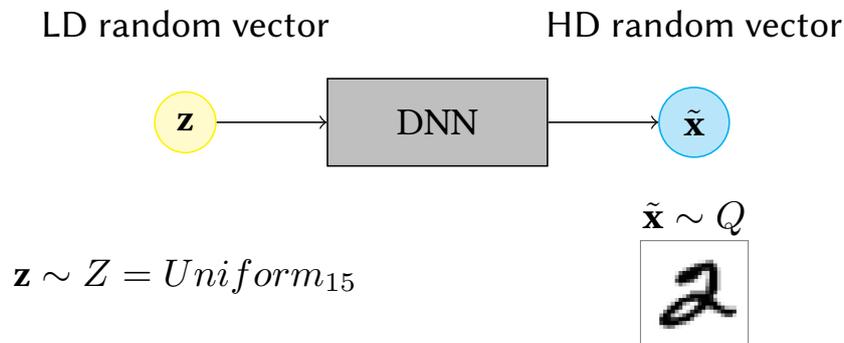


Figure 2.42: The deep generative model setup.

The input distribution Z is known, thus the computation of the likelihood $Z(\mathbf{z})$ of an input sample \mathbf{z} is direct. Nevertheless, the likelihood of an artificial sample, $Q(\tilde{\mathbf{x}})$, is not directly accessible. Indeed, the inversion of the DNN is not known. Moreover, multiple \mathbf{z} can lead to the same $\tilde{\mathbf{x}}$. This is a pre-image problem.

2.5.1 Variational auto-encoders

The LD to HD configuration of a deep generative model is very similar to the decoder part of an auto-encoder. The latent code \mathbf{h} is equivalent to the LD sample \mathbf{z} of a deep generative model. So why not using the decoder part of a AE trained on the real sample set \mathcal{U} as our generative model? The problem is that the distribution H of the latent code \mathbf{h} inside an AE is unknown. Nonetheless, we need to throw samples \mathbf{h} from H in order to generate artificial samples $\tilde{\mathbf{x}}$ from Q that should be as close as possible to the real sample distribution P . Even worse, nothing guarantees by construction that H could be approximated by a simple parametric distribution.

In a Variational Auto Encoder (VAE) [AC15], a new term is added to the training loss that push the H distribution to a known distribution, namely a normal distribution \mathcal{N} (Figure 2.43). This term consists in a Kullback-Leibler divergence between H and the targeted distribution,

$$L = \mathbb{E}_{\mathbf{x} \in \mathcal{U}} L_2(\tilde{\mathbf{x}}, \mathbf{x}) + D_{KL}(H|\mathcal{N}) . \quad (2.17)$$

The decoder part of the VAE can then be used as a generative model by sampling a new \mathbf{h} in the known normal distribution \mathcal{N} .

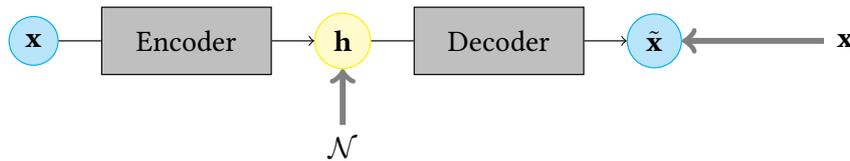


Figure 2.43: A variational auto-encoder.

☒

A VAE has the tendency not to choose between modes when multiple modes are present in P . It results in average modes in Q . For example, if \mathcal{U} contains faces with opened and closed eyes. A VAE trained on this dataset will generate samples with blurred eyes that stand for a mix between opened/closed eyelids [LKC16].

2.5.2 Generative Adversarial Network

Generative Adversarial Networks (GAN) [Goo+14] overcome the mode averaging problem using a clever setup called *Adversarial learning*.

A GAN is composed of two DNN, a *generator* which is the generative model in itself, and a *discriminator* (Figure 2.44). The *discriminator* is trained to distinguish between real samples \mathbf{x} and fake/artificial samples $\tilde{\mathbf{x}}$ coming from the generator. The *generator* is trained to fool the discriminator. As for generative model setup, the real samples are from a training dataset \mathcal{U} and should follow an unknown distribution P .

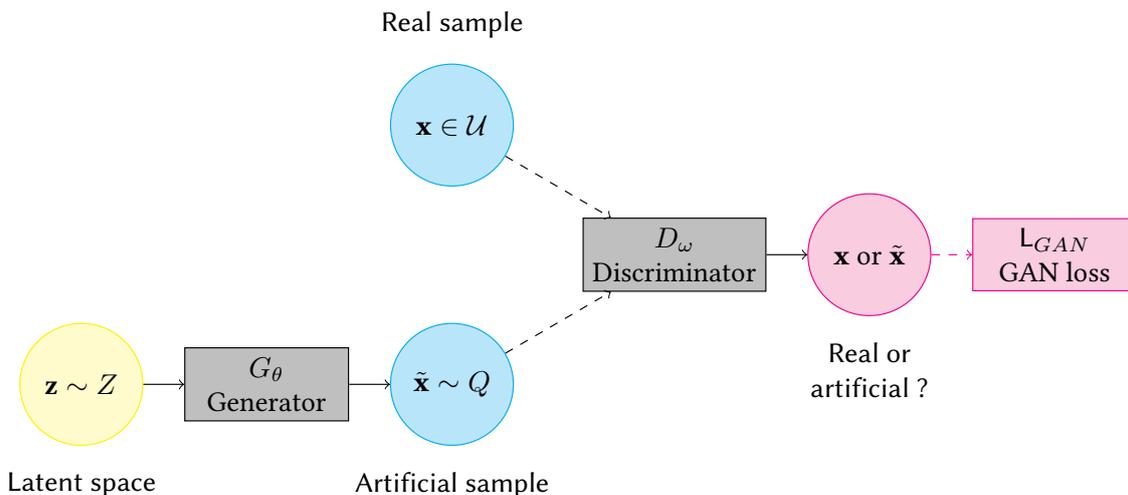


Figure 2.44: The generative adversarial network (GAN) setup. The dashed arrows mean that a sample is given whether from \mathbf{x} or exclusively from $\tilde{\mathbf{x}}$ to the discriminator.

Formally, this is a mini-max game between the two players, the generator G and the discriminator D , on the following loss V ,

$$V(\omega, \theta) = -\frac{1}{2}\mathbb{E}_{\mathbf{x}\sim P}\log(D_{\omega}(\mathbf{x})) - \frac{1}{2}\mathbb{E}_{\mathbf{z}\sim Z}\log(1 - D_{\omega}(G_{\theta}(\mathbf{z}))) , \quad (2.18)$$

$$= -\frac{1}{2}\mathbb{E}_{\mathbf{x}\sim P}\log(D_{\omega}(\mathbf{x})) - \frac{1}{2}\mathbb{E}_{\tilde{\mathbf{x}}\sim Q_{\theta}}\log(1 - D_{\omega}(\tilde{\mathbf{x}})) , \quad (2.19)$$

where ω are the parameters of the D and θ those of G . The discriminator minimizes $V(\omega, \theta)$ over ω at fixed θ ; whereas the generator maximizes $V(\omega, \theta)$ over θ at fixed ω .

In this setup, the optimal discriminator D^* is given by

$$D^*(\mathbf{x}) = \frac{P(\mathbf{x})}{Q(\mathbf{x}) + P(\mathbf{x})} , \quad (2.20)$$

where $P(\mathbf{x})$ and $Q(\mathbf{x})$ stand for the likelihood of \mathbf{x} respectively in the unknown distribution P and the built distribution Q .

When D^* is plugged into the loss V , it leads to the following loss J for the generator,

$$J(\theta) = \frac{1}{2}\mathbb{E}_{\mathbf{x}\sim P}\log(D^*(x)) + \frac{1}{2}\mathbb{E}_{\tilde{\mathbf{x}}\sim Q_{\theta}}\log(1 - D^*(\tilde{\mathbf{x}})) , \quad (2.21)$$

$$= JSD(P|Q_{\theta}) - \log(2) , \quad (2.22)$$

where $JSD(P|Q) = \frac{1}{2}KL\left(P|\frac{P+Q}{2}\right) + \frac{1}{2}KL\left(Q|\frac{P+Q}{2}\right)$ is the Jensen-Shannon divergence. Thus, for an optimal discriminator, minimizing generator loss is equivalent to minimizing Jensen-Shannon divergence between the real distribution P and the fake distribution Q .

For example, a GAN can synthesize completely artificial faces [Kar+20] that are photo-realistic (Figure 2.45).



Figure 2.45: An artificial face generated from a GAN at <https://thispersondoesnotexist.com/>.

Going back to the latent space

The Figure 2.46 shows an extension of GAN that enables going back from the HD to the LD space: the *adversarial autoencoder* [Mak+15]. A new network E acts as an encoder and projects HD \mathbf{x} to the LD \mathbf{z} . The generator G plays the role of the decoder, back-projecting the LD \mathbf{z} to $\hat{\mathbf{x}}$ an estimation of \mathbf{x} . The training is performed by the addition of an adversarial loss, L_{GAN} , and a reconstruction loss, L_{Rec} , typically a ℓ_2 distance between \mathbf{x} and $\hat{\mathbf{x}} = G(E(\mathbf{x}))$,

$$L_{Rec}(\mathbf{x}) = \|\mathbf{x} - G(E(\mathbf{x}))\|_2^2. \quad (2.23)$$

Adversarial autoencoders can be used for example to do face morphing, by performing a linear regression between two latent codes of two real faces (Figure 2.47). Moreover, adding conditioning information such as label or age to the generator/discriminator enables the system to synthesize a modified image corresponding to a query, for example produces a baby face from an elderly person [ZSQ17].

Other modifications of the GAN architecture [DKD16; Dum+16; Zha+18] propose to learn a mapping from the HD to the LD/latent space with other tricks than the autoencoder.

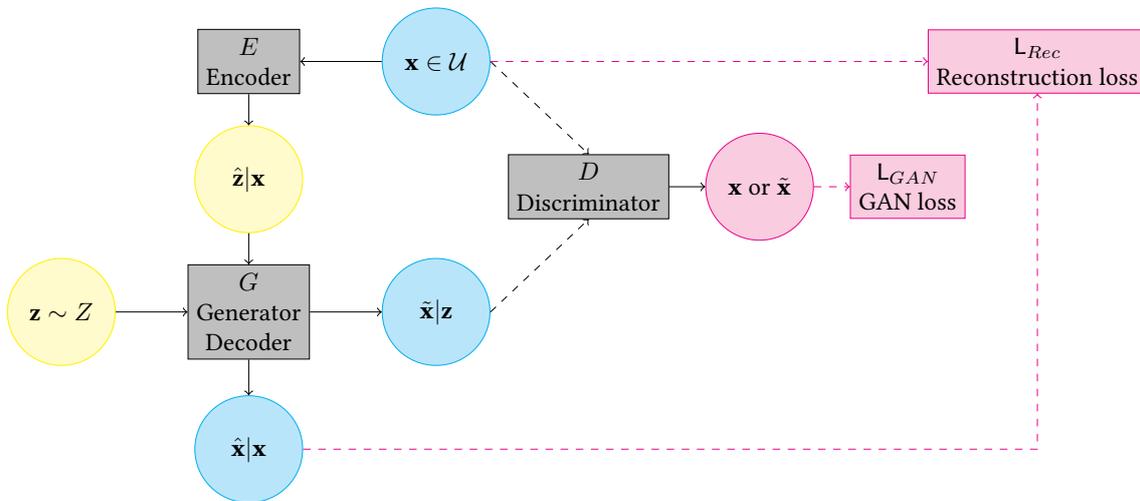


Figure 2.46: The adversarial auto-encoder.

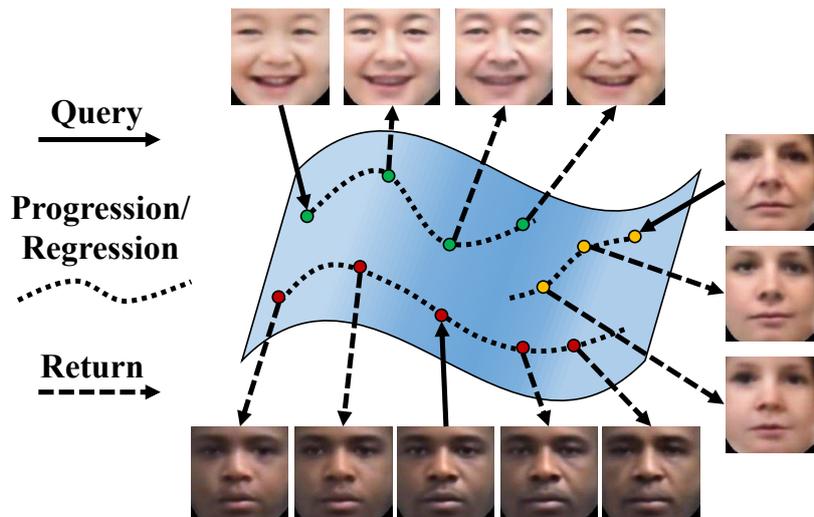


Figure 2.47: Examples of morphing and synthesis from query (Fig.1 of [ZSQ17]).

Domain to domain mapping with cycle consistency

Let's say that we want to build a mapping $M_{\mathbb{X}\mathbb{Y}}$ from the High Dimension space / source domain \mathbb{X} to an other HD space / target domain \mathbb{Y} (Figure 2.48). Nevertheless, we don't have access to a supervised set $\mathcal{L} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$ of n samples where $(\mathbf{x}_i, \mathbf{y}_i) \in \mathbb{X} \times \mathbb{Y}$ but we do own a set of n samples in \mathbb{X} alone, $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^n | \mathbf{x}_i \in \mathbb{X}$, and a set of m samples in \mathbb{Y} , $\mathcal{Y} = \{\mathbf{y}_j\}_{j=1}^m | \mathbf{y}_j \in \mathbb{Y}$ alone.

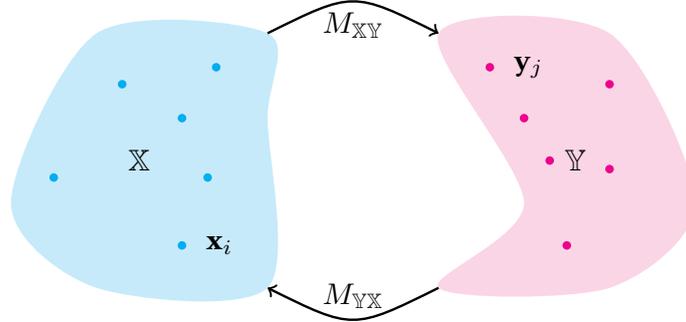


Figure 2.48: Unsupervised domain to domain mapping.

In this setup, we cannot apply standard supervised learning machinery to learn $M_{\mathbb{X}\mathbb{Y}}$ as no couple (\mathbf{x}, \mathbf{y}) are known. CycleGAN [Zhu+17], DualGAN [Yi+17], and DiscoGAN [Kim+17] introduce a similar answer to this problem using GAN.

Let's modify the GAN framework so that the input noise \mathbf{z} is replaced by \mathbf{x} , a sample from \mathcal{X} , and that the real dataset is replaced by \mathcal{Y} , the known sample set of the targeted domain (Figure 2.49). The discriminator $D_{\mathbb{Y}}$ and the mapping $M_{\mathbb{X}\mathbb{Y}}$ are trained optimizing a GAN loss $L_{GAN\mathbb{X}\mathbb{Y}}$.

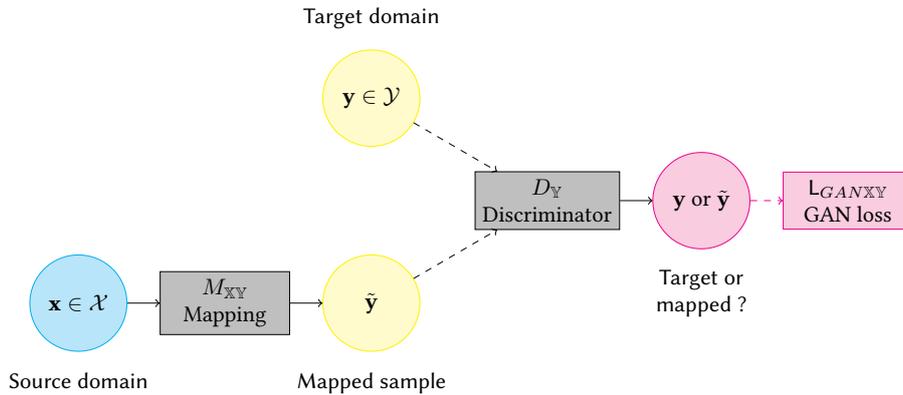


Figure 2.49: The generative adversarial network (GAN) setup diverted to learn a mapping $M_{\mathbb{X}\mathbb{Y}}$ from a source domain \mathbb{X} to a target domain \mathbb{Y} without knowing supervised couple $(\mathbf{x}, \mathbf{y}) \in \mathbb{X} \times \mathbb{Y}$.

Learning the mapping $M_{\mathbb{X}\mathbb{Y}}$ this way only guarantees that $\tilde{\mathbf{y}}$, the result of mapping from the sample \mathbf{x} , lies in the target distribution. For example, there is nothing that pushes the mapping to produce the same amount of variance and modes that is shown in the target samples \mathcal{Y} . In order to at least enforce a bijection between the source and target domain and so to preserve variances/modes, [Zhu+17; Yi+17; Kim+17] proposed to also learn the opposite mapping $M_{\mathbb{Y}\mathbb{X}}$. Furthermore, they ensure that a source sample mapped to the target domain by $M_{\mathbb{X}\mathbb{Y}}$ and then back-mapped to the source domain by $M_{\mathbb{Y}\mathbb{X}}$ stays the same.

To the former GAN presented in Fig. 2.49 is added a dual GAN to ensure that the M_{XY} produces a sample that lies in the source distribution (Figure 2.50). It is learned optimizing the dual GAN loss $L_{GAN_{YX}}$

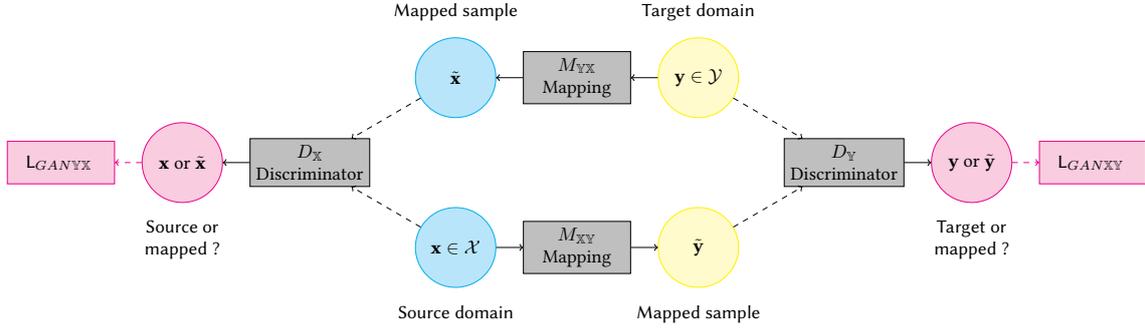


Figure 2.50: A GAN and its dual GAN.

Moreover, the cycle consistency, i.e the fact that $\hat{\mathbf{x}} = M_{XY}(M_{YX}(\mathbf{x}))$ stays close to \mathbf{x} , is checked by typically ℓ_2 distance between $\hat{\mathbf{x}}$ and \mathbf{x} as a training loss L_{cycleX} ,

$$L_{cycleX} = \|\hat{\mathbf{x}} - \mathbf{x}\|_2^2 . \quad (2.24)$$

A dual cycle is also formed to ensure that $\hat{\mathbf{y}} = M_{YX}(M_{XY}(\mathbf{y}))$ stays close to \mathbf{y} as well, leading to its own cycle loss L_{cycleY} (Figure 2.51).

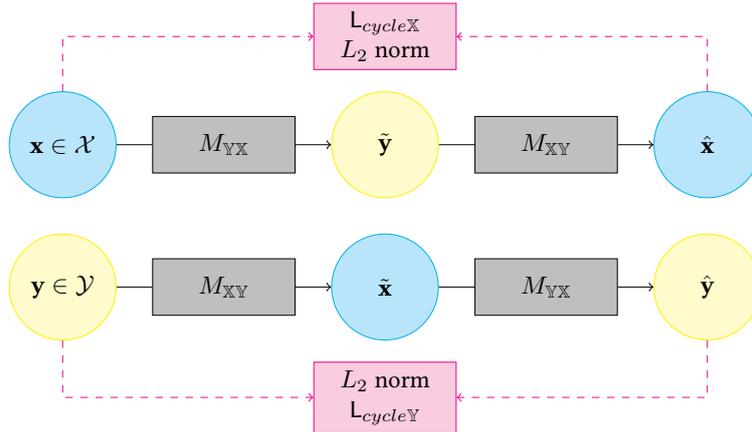


Figure 2.51: The two consistency cycles of CycleGAN.

In total, there are 4 losses to optimize at the same time, the two GAN losses $L_{GAN_{XY}}$ and $L_{GAN_{YX}}$ as well as the two cycle losses L_{cycleX} and L_{cycleY} , over the parameters of the two discriminators D_Y and D_X as well as over the parameters of the two mappings M_{XY} and M_{YX} .

In the same manner as GAN, CycleGAN shows great achievement in working on images. It could be used as unsupervised image to image translation [Zhu+17] when supervised image couples have no meaning or are not accessible such as converting image of horses to zebras and vice versa (Figure 2.52).

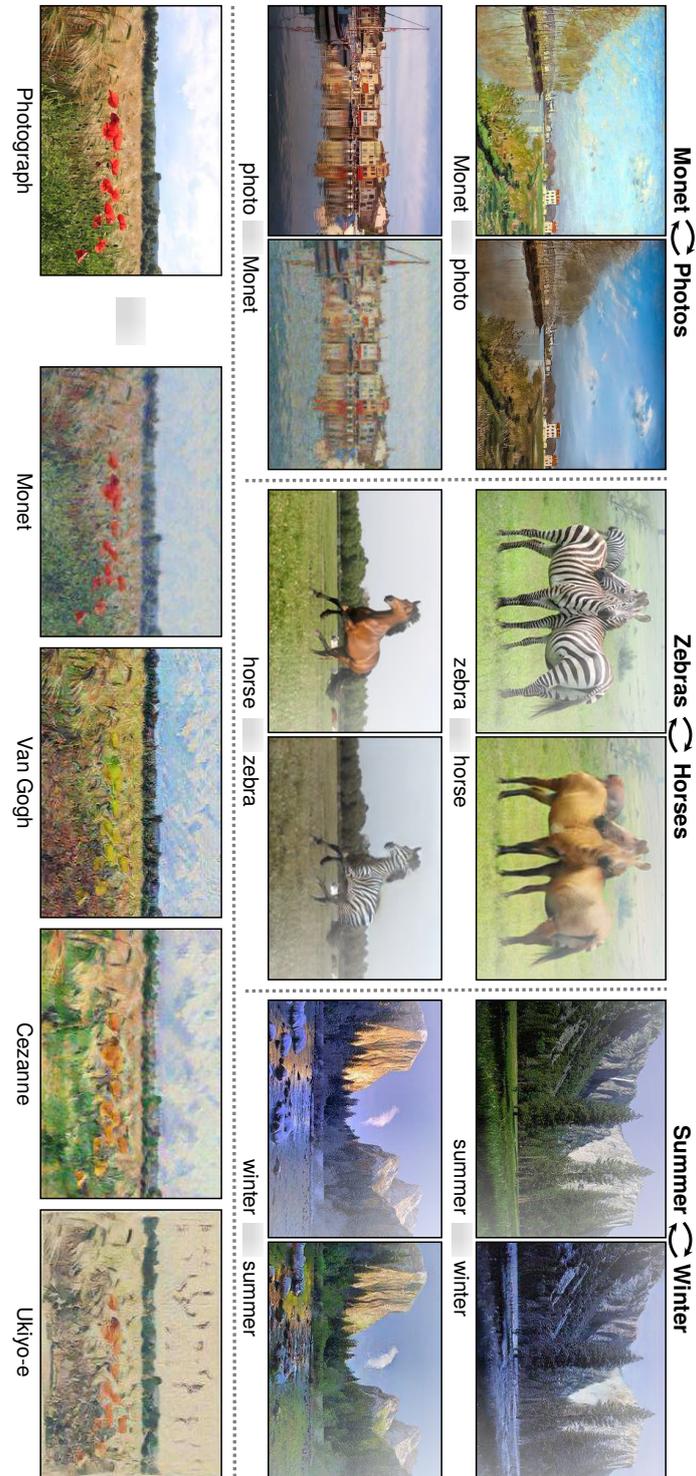


Figure 2.52: Examples of image to image translations done by CycleGAN (Fig.1 of [Zhu+17]).

Chapter 3

High-dimensional/structured input/output problems

In this chapter, I shall present the work that we have undertaken on a special class of machine learning problems, namely tasks where the input and/or the output spaces lie on high dimension or contained dependent/ structured information.

We are aiming at taking into account input dependencies $p(\mathbf{x})$ and output dependencies $p(\mathbf{y})$ to help learning a targeted supervised task $p(\mathbf{y}|\mathbf{x})$ where \mathbf{x} and \mathbf{y} belong to HD or structured spaces. These three goals, learning $p(\mathbf{x})$, $p(\mathbf{y})$ and $p(\mathbf{y}|\mathbf{x})$, will be formulated as learning three models, respectively M_{in} , M_{out} and M_{sup} . We have developed different strategies to link these models which are explained in details in the publications [Ler+15; Bel+17; Bel+18] attached respectively in the appendices A.1, A.2, and A.3.

In the following sections we will show examples of such problems and how they are solved in the literature, formalize the proposed framework and detail the different training strategies.

At the end of this chapter, we will open up on other possible strategies to solve this class of problems based this time on generative models. This is linked to our recent work [Ruf+20] presented in appendix A.4.

Contents

3.1	What are high-dimensional or structured problems ?	65
3.1.1	Image labeling / semantic segmentation : an example of high-dimensional problem	65
3.1.2	A broader approach: structured output problems	69
3.1.3	Toward high-dimensional/structured input/output (HD SIO) problems	70
3.2	Solving HD SIO problems using multi-task regularization	71
3.2.1	The Multi-Task Learning setup	71
3.2.2	Examples of sequential learning	74
3.2.3	Examples of concomitant learning	77
3.2.4	Perspectives and undergoing works	78
3.3	Constrained deep generative models	79
3.3.1	Image synthesis/reconstruction with few constraint	79
3.3.2	Polarimetric conversion	81
3.3.3	Sequence prediction	83

Selected publications

[LHC09] Benjamin Labbé, Romain Hérault, and Clement Chatelain. “Learning Deep Neural Networks for High Dimensional Output Problems.” In: *ICMLA* (United States). Dec. 2009, 6p. URL: <https://hal.archives-ouvertes.fr/hal-00438714>

[Ler+15] Julien Lerouge et al. “IODA: An Input/Output Deep Architecture for Image Labeling” In: *Pattern Recognition* 48.9 (Sept. 1, 2015), pp. 2847–2858. ISSN: 0031-3203. DOI: 10.1016/j.patcog.2015.03.017. URL: <http://www.sciencedirect.com/science/article/pii/S0031320315001181> (Appendix A.1)

[Bel+17] Soufiane Belharbi et al. “Spotting L3 Slice in CT Scans Using Deep Convolutional Network and Transfer Learning.” In: *Computers in Biology and Medicine* 87 (Aug. 1, 2017), pp. 95–103. ISSN: 0010-4825. DOI: 10.1016/j.combiomed.2017.05.018. URL: <http://www.sciencedirect.com/science/article/pii/S0010482517301403> (Appendix A.2)

[Bel+18] Soufiane Belharbi et al. “Deep Neural Networks Regularization for Structured Output Prediction.” In: *Neurocomputing* 281 (Mar. 15, 2018), pp. 169–177. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2017.12.002. URL: <http://www.sciencedirect.com/science/article/pii/S0925231217318295> (Appendix A.3)

[Ruf+20] Cyprien Ruffino et al. “Pixel-Wise Conditioned Generative Adversarial Networks for Image Synthesis and Completion.” In: *Neurocomputing* (Apr. 2020). DOI: 10.1016/j.neucom.2019.11.116. arXiv: 2002.01281. URL: <https://hal.archives-ouvertes.fr/hal-02551730> (Appendix A.4)

3.1 What are high-dimensional or structured problems ?

3.1.1 Image labeling / semantic segmentation : an example of high-dimensional problem

Here we will place ourselves in the context of building a classifier that gives a label to each pixel of an image (Fig. 3.1). This problem is called image labeling or semantic segmentation. Namely, for an image \mathbf{X} of size $m \times n$ with p channels/feature maps we are looking for a label map \mathbf{Y} of size $m \times n$ that gives to each pixel of the input image one of the q possible classes. The p channels can represent grayscale/color space information or features extracted locally around the pixel, each of them are real values. This can be formulated by

$$f : \begin{array}{ccc} \mathbf{X} & \rightarrow & \mathbf{Y} \\ \mathbb{R}^{m \times n \times p} & \rightarrow & \mathbb{Y}^{m \times n} \end{array}, \quad (3.1)$$

where $m \times n$ is the size of the image, p the number of channels, and $\mathbb{Y} = \{c_1, c_2, \dots, c_q\}$ the set of the q possible classes. Let's note $x \in \mathbb{R}^p$ one of the $m \times n$ pixels of the image \mathbf{X} , and y one of $m \times n$ elements of the label map \mathbf{Y} .

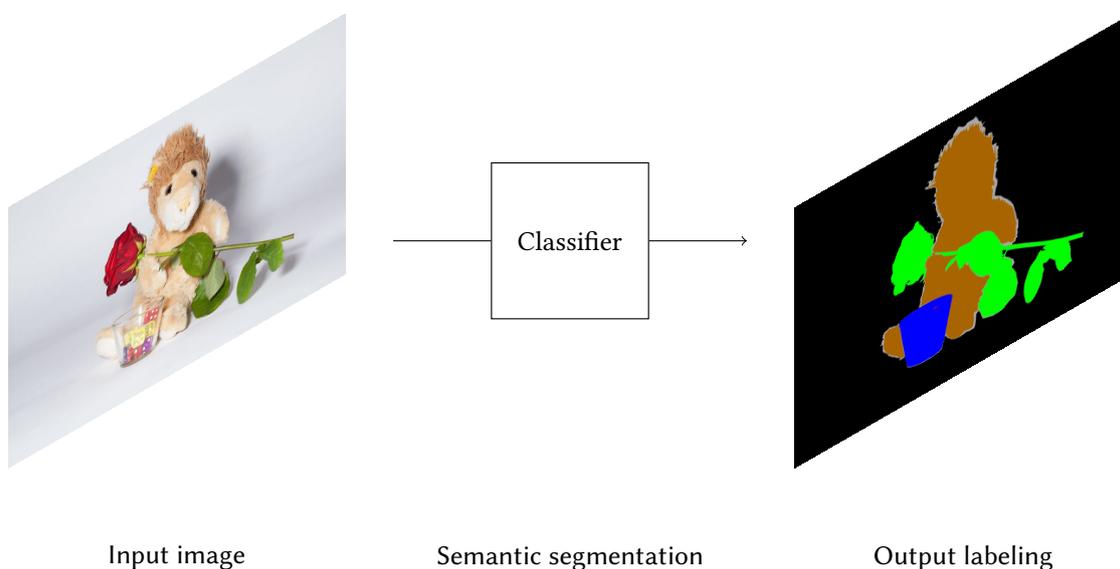


Figure 3.1: The image labeling / semantic segmentation problem.
Source Wikimedia commons[Tho03; Mar17]

In the literature, one can find two kinds of approaches for this semantic segmentation problem:

- A local, independent labeling performed on each pixel, taking into account the local distribution $p(y|x)$,
- A global labeling that directly gives the full label map, considering the global conditional distribution $p(\mathbf{Y}|\mathbf{X})$.

Independent pixel approaches

A first simple attempt to perform image labeling is to consider the task as a local classification problem where a label y is given to each pixel x independently. Actually, the features of a pixel are usually computed taking into account the local neighborhood of the pixel increasing the *field of perception* of the local labeling (Fig. 3.2).

This first local classification can be followed by a post-processing on the full label map [KLT09] in order to render smoothed and homogeneous label regions.

In these approaches, neither the full image distribution $p(\mathbf{X})$ nor the full label map distribution $p(\mathbf{Y})$ are considered but only the pixel/label dependency $p(y|x)$.

Among the possible classifiers using this approach, we can find Logistic Regression [CP11], SVM [Fer+08], or Artificial Neural Networks. In fact, fully Convolutional Network (FCN) [LSD15] can also be viewed as local classifiers though their *field of perception* is increased due to the stacked convolutions or dilated convolutions [YK15; Luo+16].

These methods suffer from the *Sayre's paradox* which states that an object cannot be recognized before being segmented but cannot be segmented before being recognized.

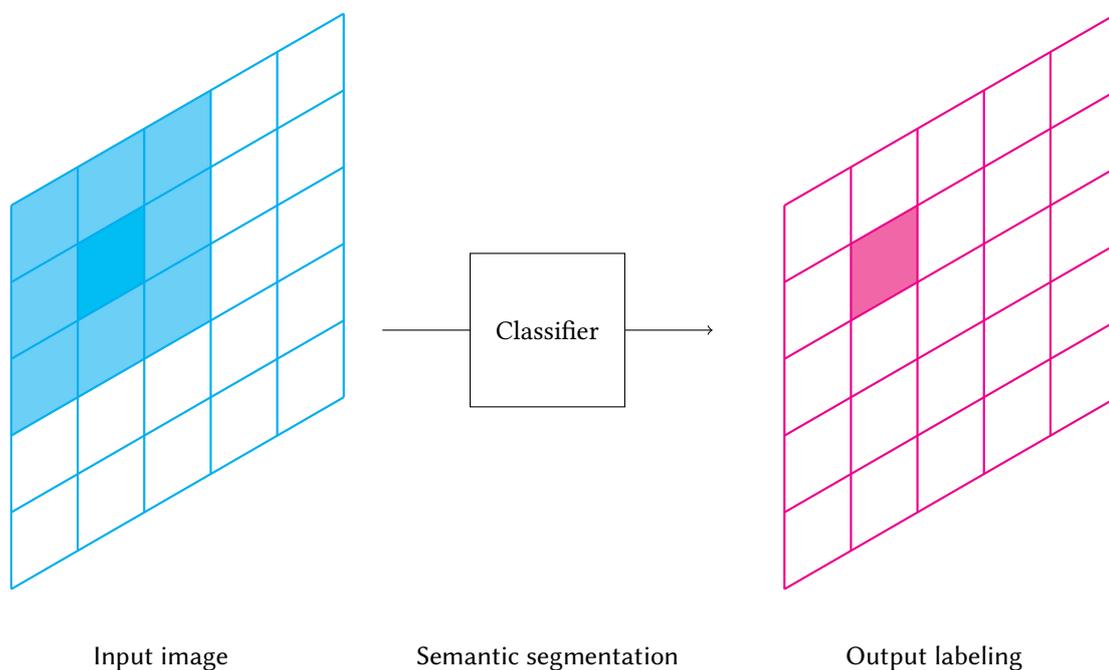


Figure 3.2: Semantic segmentation with an independent pixel approach. Only the features of the pixel in cyan (and possibly its surroundings in light cyan) are used to compute its label in magenta.

Global Approaches

The global approach framework is depicted in Figure 3.3.

Most of the popular models for global image labeling before the rise of deep neural network were 2D-probabilistic models inheriting from 1D method such as Hidden Markov Model (HMM) [RJ86] or Conditional Random Field (CRF) [LMP01]. This time the global distributions $p(\mathbf{X}, \mathbf{Y})$ in case of HMM or $p(\mathbf{Y}|\mathbf{X})$ in case of CRF are targeted. These models have proven to be efficient on 1D sequence task such as text, handwriting and voice recognition. Nevertheless, Markov Random Field (MRF) [Kin80] or 2D-CRF [Nic+07], their 2D counter-parts suffers from a high decoding complexity leading to high decision time even when using sub-optimal label assignment.

For small 2D lattices, we can also explore structured output SVM [Tso+04; BL08] or kernel joint projection [Wes+02] in order to compute $p(\mathbf{X}, \mathbf{Y})$. Nonetheless, if the likelihood is easily accessible when \mathbf{X} and \mathbf{Y} are known, the classifiers are not made to infer \mathbf{Y} when \mathbf{X} is known; letting the user with the obligation to explore the output space. Thus, these models are not suitable for large image labeling.

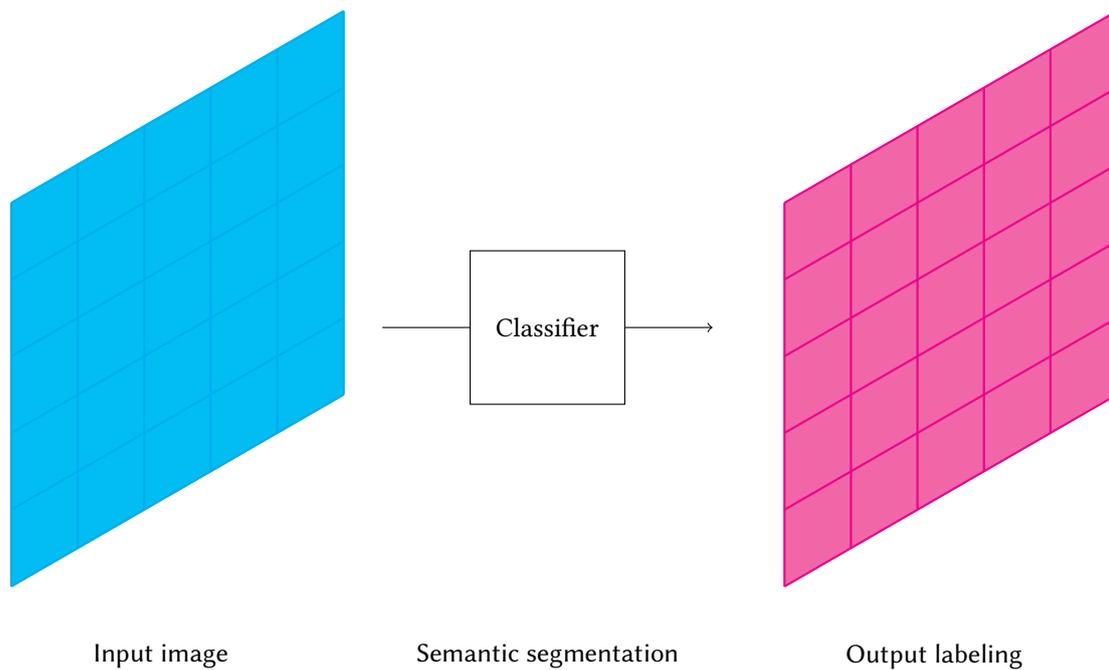


Figure 3.3: Semantic segmentation with a global approach.
All pixel features are involved for the liberalization of each pixel.

Hybrid Approaches

Recently, a hybrid approach called U-Net [RFB15] was proposed by combining in a same architecture a fully convolutional network, that is taking decision locally, and an auto-encoder, that is taking global decisions. The network consists in a fully convolutional network with a diablo shape. The latent code in the middle of the diablo accounts for global informations on the image, enabling global decisions at the decoder stage. In order to preserve local information to the decoder, skip connections were added between encoder layers and decoder layers of the same size (see Fig. 3.4). This method originally created for medical images has been proven to be efficient for semantic segmentation in many domains.

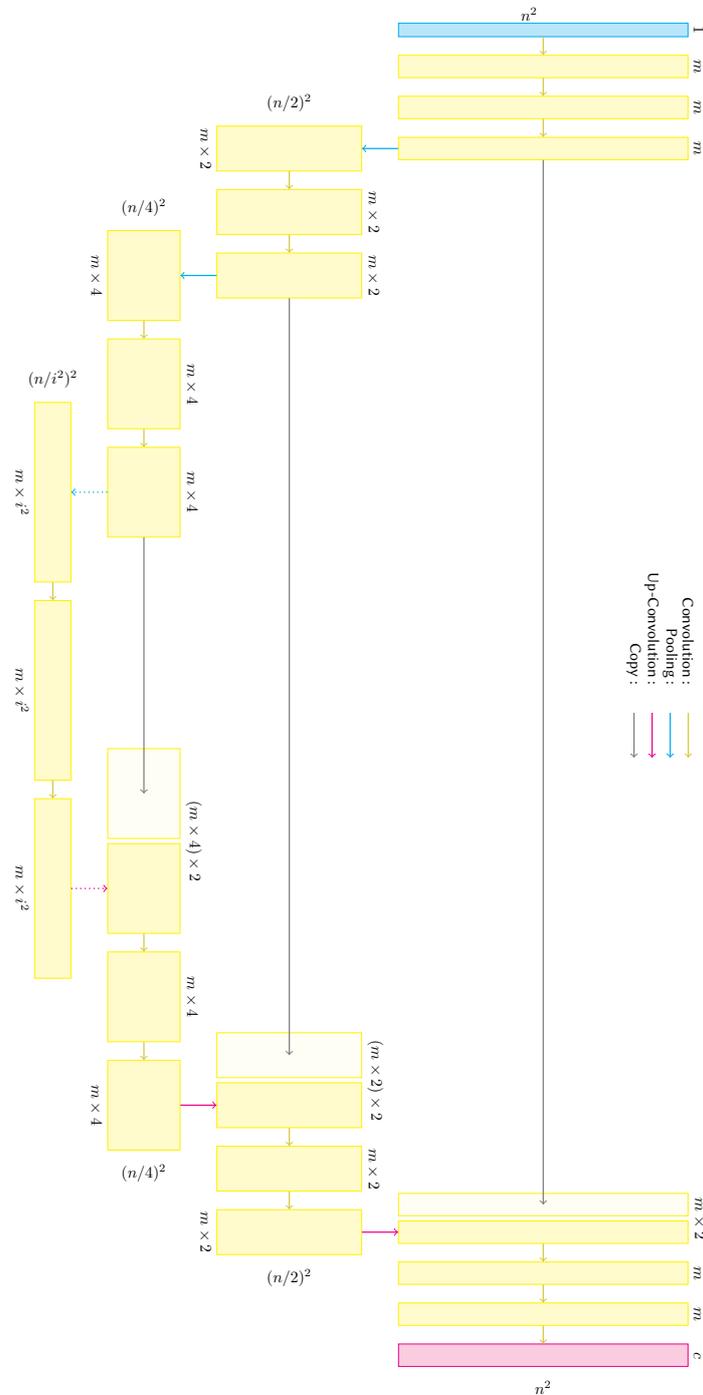


Figure 3.4: U-Net general principles.

n^2 is the size of the input image, m number of feature map for the first convolution blocks, i the number of Pooling/Up-Convolution, and c the number of classes in the labeling.

3.1.2 A broader approach: structured output problems

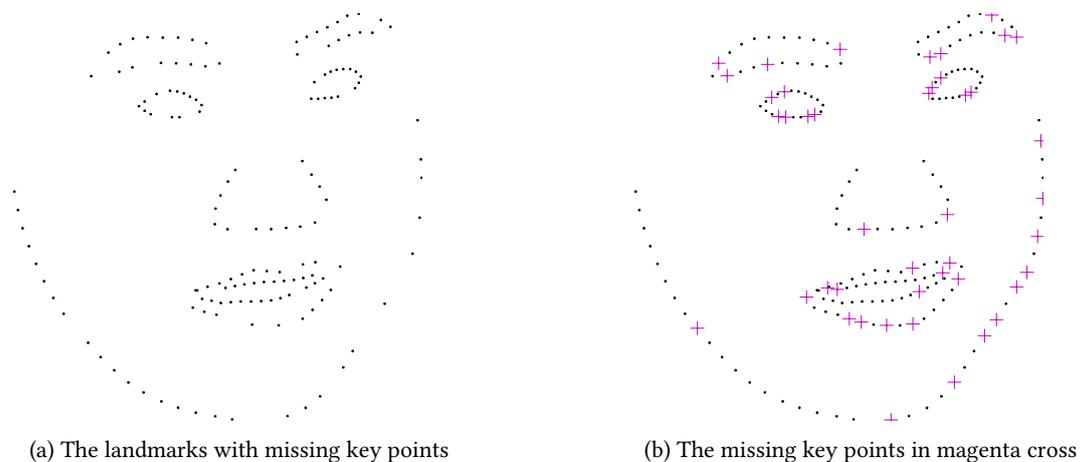
Image labeling / Semantic Segmentation is a task where the output lies in a multi-dimensional space describing discrete or continuous variables that are inter-dependent. In fact, this problem can be cast into a broader family where one tries to address a supervised task in a structured/inter-dependent output space.

Let's take the example of facial landmark detection (Fig. 3.5) The task consists in predicting the coordinates of a set of key points such as eye centers, lips contours, nose position, ... given a face image as input.



Figure 3.5: The facial landmark detection problem. Image and landmarks from the HELEN dataset [Le+12].

The set of points are interdependent throughout geometric relations induced by the face structure. If one or few key points are hidden, you will be able to reconstruct the full structure by only seeing the remaining anchors without knowing the underlying face image (Fig. 3.6). Your own *a priori* of the body structure gives you the distribution of the key points, $p(\mathbf{y})$, from which you can infer the missing ones.



(a) The landmarks with missing key points

(b) The missing key points in magenta cross

Figure 3.6: Interdependence induced by the face structure.

The brain is able to reconstruct the missing key points by its *a priori* on the face structure.

That is why not only the distribution of the input space $p(\mathbf{X})$ but also the distribution of the output space $p(\mathbf{y})$ shall be important to determine the distribution $p(\mathbf{y}|\mathbf{X})$ which is the task goal.

Many other tasks can be seen as structured output tasks. One may cite Natural Language Processing, speech processing, handwriting recognition, image captioning or, as seen in previous section, image labeling. In fact, a task that involves for its output graphs, sequences, high dimension space (image/tensor) could be seen as a *structured output* task.

Two main families arise from the literature to address these tasks: historically, *graphical models* lead the state-of-the-art until 2010 when *Neural Networks / Deep Learning* start to challenge then overcome traditional methods in tackling structured output problems.

Graphical models

HMM is known to be suited for processing temporal/sequential data. Nonetheless, the Markovian hypothesis on which they rely assumes that outputs are independent. Conditional Random Fields (CRF) have been proposed to overcome this issue, thanks to its capability to learn large dependencies of the observed output data.

Nevertheless, as with image labeling, their extension to 2D or n-D typologies such as MRF does not scale well and sub-optimal strategies must be used [Bes86; CB90]. Moreover, by the time we were exploring this task family very few works addressed continuous space / regression task such as the facial landmark detection [Fri93; NC08].

Neural Networks / Deep Learning

Besides the deep architectures already cited in the previous sections that are targeting image labeling problem [RFB15; LSD15], one can cite Recurrent Neural Network (RNN) that is more suited to sequence processing such as handwriting recognition, translation, or speech [GS05; Cho+14a]. However, these DNN or RNN models do not consider explicitly the output dependencies.

Conditional Restricted Boltzmann Machines, a particular case of neural networks and probabilistic graphical models have been used with different training algorithms according to the size of the plausible output configurations [MLH11]. Training and inferring using such models remain a difficult task.

In this same direction, [BM16] tackles structured output problems as an energy minimization through two feed-forward networks. The first is used for feature extraction over the input. The second is used for estimating an energy by taking as input the extracted features and the current state of the output labels. This allows learning the inter-dependencies within the output labels. The prediction is performed using an iterative backpropagation-based method with respect to the labels through the second network which remains computationally expensive.

In order to deal with high-dimensional and structured problems, in our work we make the hypothesis that learning the output dependencies $p(\mathbf{y})$, and possibly the learning of the input dependencies $p(\mathbf{x})$, should help the learning of the targeted supervised task $p(\mathbf{y}|\mathbf{x})$.

3.1.3 Toward high-dimensional/structured input/output (HD SIO) problems

We proposed a framework that could be used to address problems where both input and output can have fixed (tensor) or unfixed sizes (sequences), high-dimensional (images, scan) or structured information (graphs). The input and the output can be of different natures such as in the facial landmark detection problem where the input is an image and the output a graph. We call this problem the high-dimensional/structured input/output problem or HD SIO problem.

Most of the approaches presented above concentrate on learning the targeted supervised task $p(\mathbf{y}|\mathbf{x})$ and the input dependencies $p(\mathbf{x})$ or exclusively the output dependencies $p(\mathbf{y})$. In our work in order to deal with high-dimensional and structured problems, we make the hypothesis that learning explicitly both the input dependencies $p(\mathbf{x})$ and the output dependencies $p(\mathbf{y})$ should help the learning of the targeted supervised task $p(\mathbf{y}|\mathbf{x})$.

For the sake of readability, we use the vector notation (bold small letters) for \mathbf{x}, \mathbf{y} but they can be of any forms: scalars as in y notation (though it will lead to a degenerated HD/SIO problem), tensors such as in \mathbf{X} notation, sequences or graphs.

3.2 Solving HD SIO problems using multi-task regularization

We are aiming at taking into account input dependencies $p(\mathbf{x})$ and output dependencies $p(\mathbf{y})$ to help learn the targeted supervised task $p(\mathbf{y}|\mathbf{x})$. In order to do so, these three goals are formulated as three models:

- Two side reconstruction/representation models:
 - M_{in} which will try to find on which manifold lies the input,
 - M_{out} , which in turn will try to learn the dependencies among the output space,
- One main supervised model, M_{sup} , which tries to guess the correct output given the input.

We have developed different strategies to link these models:

1. Learn M_{in} and M_{out} separately and then used part of their parameters to initialize the parameters of M_{sup} ,
2. Link part of the parameters of M_{in} and M_{out} to the ones of M_{sup} , and then learn the tasks concurrently,
3. Still link the models as previously, and learn the tasks concurrently but evolving the weight of the different tasks during the training.

They are described in details in the following sub-sections with experimental results.

3.2.1 The Multi-Task Learning setup

Let us consider a training set \mathcal{D} containing examples with both features and targets (\mathbf{x}, \mathbf{y}) , features without target $(\mathbf{x}, _)$, and targets without features $(_, \mathbf{y})$. We split this set into:

- a set \mathcal{F} which is the subset of \mathcal{D} containing examples with at least features x ,
- a set \mathcal{L} which is the subset of \mathcal{D} containing examples with at least targets y ,
- and a set \mathcal{S} which is the subset of \mathcal{D} containing examples with both features x and targets y .

One can note that all examples in \mathcal{S} are also in \mathcal{F} and in \mathcal{L} .

Let us also consider a side training set \mathcal{D}' containing examples with both features and targets (\mathbf{x}, \mathbf{z}) from another supervised problem where the feature \mathbf{x} belongs to the same feature/input space as \mathcal{D} .

From these sets we learned 3 models M_{in} , M_{out} and M_{sup} , through 3 different tasks.

Input task

The input model M_{in} is inferred in a supervised or unsupervised manner depending on the availability of a side training set \mathcal{D}' from an other supervised task which relies on the same input space \mathcal{X} as the main targeted task.

Supervised learning

If we have a side supervised set \mathcal{D}' , the input model M_{in} can be learned as a supervised kick task s_{in} which is composed of two parts. The input data \mathbf{x} is projected into an intermediate representation space $\tilde{\mathbf{x}}$ through f_{in} which plays the role of a *feature extractor* or learned representation. Then $\hat{\mathbf{z}}$, the guessed label, is estimated from $\tilde{\mathbf{x}}$ by the function f'_{in} which plays the role of the actual *discriminator*.

$$\hat{\mathbf{z}} = s_{in}(\mathbf{x}; \mathbf{w}_{in}) = f'_{in}(\tilde{\mathbf{x}} = f_{in}(\mathbf{x}; \mathbf{w}_{cin}); \mathbf{w}_{din}) \quad , \quad (3.2)$$

where $\mathbf{w}_{in} = \{\mathbf{w}_{cin}, \mathbf{w}_{din}\}$. The discriminator parameters \mathbf{w}_{din} are proper to this task however the feature extractor parameters \mathbf{w}_{cin} are shared with the main task.

The training loss for this task is given by :

$$L_{in}(\mathcal{D}'; \mathbf{w}_{in}) = \frac{1}{\text{card}\mathcal{D}'} \sum_{(\mathbf{x}, \mathbf{z}) \in \mathcal{D}'} C_{in}(s_{in}(\mathbf{x}; \mathbf{w}_{in}), \mathbf{z}) \quad , \quad (3.3)$$

where C_{in} is a supervised learning cost which can be computed on all the side samples (i.e. on \mathcal{D}').

Unsupervised learning

If we don't have a side supervised set \mathcal{D}' , which is generally the case, the input model M_{in} can be learned by an unsupervised reconstruction task r_{in} which aims at learning global and more robust input representation based on the original input data \mathbf{x} . This task projects the input data \mathbf{x} into an intermediate

representation space $\tilde{\mathbf{x}}$ through a coding function f_{in} , known as *encoder*. Then, it attempts to recover the original input by reconstructing $\hat{\mathbf{x}}$ from $\tilde{\mathbf{x}}$ through a decoding function f'_{in} , known as *decoder*:

$$\hat{\mathbf{x}} = r_{in}(\mathbf{x}; \mathbf{w}_{in}) = f'_{in}(\tilde{\mathbf{x}} = f_{in}(\mathbf{x}; \mathbf{w}_{cin}); \mathbf{w}_{din}) , \quad (3.4)$$

where $\mathbf{w}_{in} = \{\mathbf{w}_{cin}, \mathbf{w}_{din}\}$. The decoder parameters \mathbf{w}_{din} are proper to this task however the encoder parameters \mathbf{w}_{cin} are shared with the main task.

The training loss for this task is given by :

$$L_{in}(\mathcal{F}; \mathbf{w}_{in}) = \frac{1}{\text{card}\mathcal{F}} \sum_{\mathbf{x} \in \mathcal{F}} C_{in}(r_{in}(\mathbf{x}; \mathbf{w}_{in}), \mathbf{x}) , \quad (3.5)$$

where C_{in} is an unsupervised learning cost which can be computed on all the samples with features (i.e. on \mathcal{F}).

Output task

As for the input model, the output model M_{out} could be learned in a supervised manner if a side supervised task and dataset which relies on the same output space \mathcal{Y} as the targeted supervised task is available. In our applications, we had never faced this case so let's only keep the description of the unsupervised learning.

Therefore, the output task r_{out} is an unsupervised reconstruction task which has the same goal as the unsupervised version of the input task. Similarly, this task projects the output data \mathbf{y} into an intermediate representation space $\tilde{\mathbf{y}}$ through a coding function f'_{out} , i.e. a coder. Then, it attempts to recover the original output data by reconstructing $\hat{\mathbf{y}}$ based on $\tilde{\mathbf{y}}$ through a decoding function f_{out} , i.e. a decoder. In structured output data, $\tilde{\mathbf{y}}$ can be seen as a code that contains many aspects of the original output data \mathbf{y} , most importantly, its hidden structure that describes the global relation between the components of \mathbf{y} . This hidden structure is discovered in an unsupervised way without priors fixed beforehand which makes it simple to use. Moreover, it allows using labels only (without input \mathbf{x}) which can be helpful in tasks with abundant output data such as in speech recognition or translation task :

$$\hat{\mathbf{y}} = r_{out}(\mathbf{y}; \mathbf{w}_{out}) = f_{out}(\tilde{\mathbf{y}} = f'_{out}(\mathbf{y}; \mathbf{w}_{cout}); \mathbf{w}_{dout}) , \quad (3.6)$$

where $\mathbf{w}_{out} = \{\mathbf{w}_{cout}, \mathbf{w}_{dout}\}$. In the opposite manner of the input task, the encoder parameters \mathbf{w}_{cout} are proper to this task while the decoder parameters \mathbf{w}_{dout} are shared with the main task.

The training loss for this task is given by :

$$L_{out}(\mathcal{L}; \mathbf{w}_{out}) = \frac{1}{\text{card}\mathcal{L}} \sum_{\mathbf{y} \in \mathcal{L}} C_{out}(r_{out}(\mathbf{y}; \mathbf{w}_{out}), \mathbf{y}) , \quad (3.7)$$

where C_{out} is an unsupervised learning cost which can be computed on all the samples with labels (i.e. on \mathcal{L}).

Main task

The main model M_{sup} is built upon a supervised task that attempts to learn a decision function f_{sup} between features \mathbf{x} and labels \mathbf{y} .

In order to do so, the first part of this function is shared with the encoding/feature extraction part f_{in} of the input task and the last part is shared with the decoding part f_{out} of the output task. The mapping function f_{map} between $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$, i.e. the middle part of the decision function f_{sup} , is specific to this model and is parameterized by \mathbf{w}_{map} ,

$$\hat{\mathbf{y}} = f_{sup}(\mathbf{x}; \mathbf{w}_{sup}) = f_{out}(f_{map}(f_{in}(\mathbf{x}; \mathbf{w}_{cin}); \mathbf{w}_{map}); \mathbf{w}_{dout}) , \quad (3.8)$$

where $\mathbf{w}_{sup} = \{\mathbf{w}_{cin}, \mathbf{w}_{map}, \mathbf{w}_{dout}\}$. Accordingly, \mathbf{w}_{cin} and \mathbf{w}_{dout} parameters are respectively shared with the input and output tasks.

Learning this task consists in minimizing its training loss L_{sup} ,

$$L_{sup}(\mathcal{S}; \mathbf{w}_{sup}) = \frac{1}{\text{card}\mathcal{S}} \sum_{(x,y) \in \mathcal{S}} C_{sup}(f_{sup}(\mathbf{x}; \mathbf{w}_{sup}), \mathbf{y}) , \quad (3.9)$$

where C_{sup} is a supervised learning cost which can be computed on all the samples with features and labels (i.e. on \mathcal{S}).

Framework full picture

As a synthesis, our proposal is formulated as a multi-task learning framework (MTL) [Car97] which gathers a main task and two secondary tasks. This framework is illustrated in Figure 3.7 and in Figure 3.8 for respectively the supervised or unsupervised learning of input model.

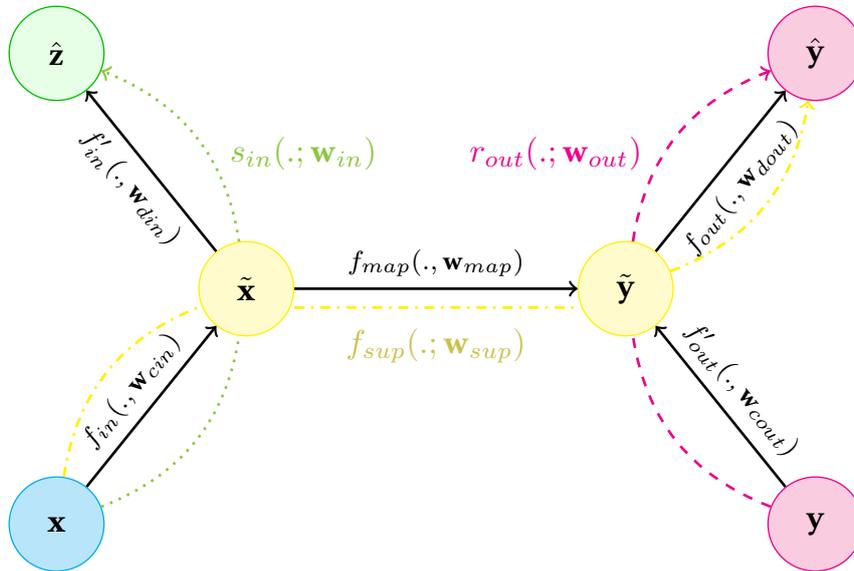


Figure 3.7: How models are nested : supervised input version

The input model M_{in} is s_{in} , the output model M_{out} is r_{out} , and the main model M_{sup} is g_{sup} .

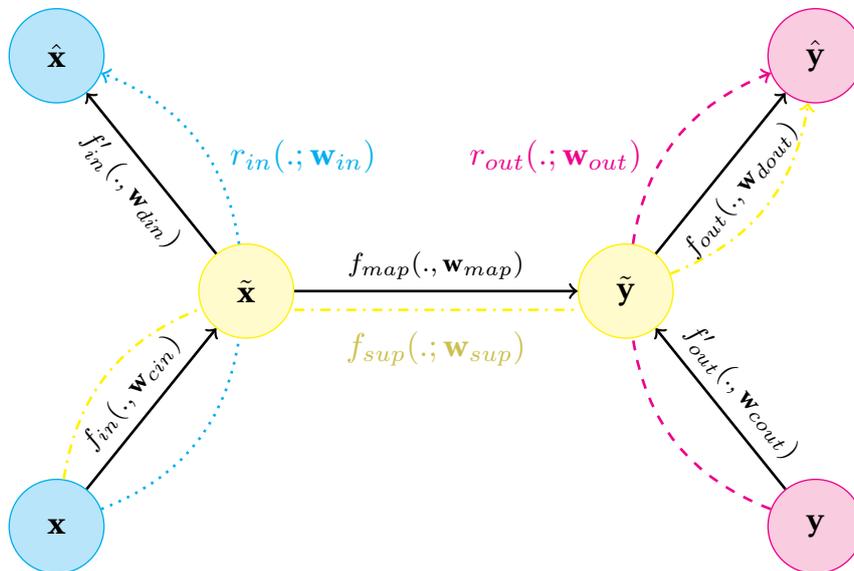


Figure 3.8: How models are nested : unsupervised input version

The input model M_{in} is r_{in} , the output model M_{out} is r_{out} , and the main model M_{sup} is g_{sup} .

Learning strategies

We can distinguish between two main learning strategies for the aforementioned models:

Sequential learning

In this methodology, the parameters of models M_{in} and M_{out} are trained separately without taking into account the main task. Then, the models parameters are (partially) transferred to the main model M_{sup} as initialization parameters.

The input M_{in} and output M_{out} models are *pre-trained* using a unsupervised or supervised task. Then, their parameters \mathbf{w}_{cin} and \mathbf{w}_{dout} are taken out to initialize the corresponding parameters in M_{sup} which is at its turn trained or *fine-tuned* to the supervise task. Let's note that \mathbf{w}_{cin} and \mathbf{w}_{dout} are generally not fixed during the fine tuning. Thus, all the \mathbf{w}_{sup} parameters, not only \mathbf{w}_{map} proper to M_{sup} , are adjusted by this last training.

The Algorithm 1 presented in Chapter 2 and introduced by [HS06] for pre-training deep architectures is a candidate for such training, but there only M_{in} is pre-trained. In [Ler+15] (Appendix A.1), we have extended this methodology to the output model M_{out} . The premise of this extension could be found in our earlier publication [LHC09].

Transfer learning is also part of this class of strategies [Bel+17] (Appendix A.2).

Concomitant learning

Rather than being trained separately all the models can be trained at the same time in a multi-objective scalarized straining loss :

$$L(\mathbf{w}) = \lambda_{sup} \cdot L_{sup}(\mathcal{S}; \mathbf{w}_{sup}) + \lambda_{in} \cdot L_{in}(\mathcal{F}; \mathbf{w}_{in}) + \lambda_{out} \cdot L_{out}(\mathcal{L}; \mathbf{w}_{out}), \quad (3.10)$$

where $\mathbf{w} = \{\mathbf{w}_{cin}, \mathbf{w}_{din}, \mathbf{w}_{map}, \mathbf{w}_{cout}, \mathbf{w}_{dout}\}$ is the complete set of parameters of the framework. Let's remember that $\mathbf{w}_{in} = \{\mathbf{w}_{cin}, \mathbf{w}_{din}\}$, $\mathbf{w}_{out} = \{\mathbf{w}_{cout}, \mathbf{w}_{dout}\}$ and $\mathbf{w}_{sup} = \{\mathbf{w}_{cin}, \mathbf{w}_{map}, \mathbf{w}_{dout}\}$. Given that the tasks have different importance, they are balanced using a importance weight λ_{sup} , λ_{in} and λ_{out} multiplied by their corresponding cost, respectively for the main/supervised, the input and output tasks.

In [Bel+16a], we have extended this approach by combining *sequential learning* as presented in the above paragraph and *concomitant learning* as depicted in equation 3.10. Instead of using fixed importance weights that can be difficult to optimally set, λ_{sup} , λ_{in} and λ_{out} are modified along training epochs. Different evolution schemes have been experienced in [Bel+18] (Appendix A.3).

Sequential and concomitant strategies can be seen as a special case of this methodology. The sequential strategy corresponds to λ_{sup} sets to 0 and λ_{in} as well as λ_{out} set to 0.5 up for n pre-training epochs followed by λ_{sup} sets to 1 and λ_{in} as well as λ_{out} set to 0 for m fine-tuning epochs. Whereas standard concomitant learning corresponds to fixed weights.

3.2.2 Examples of sequential learning

We have used the sequential learning strategy in different publications . As such, we were among the first ones to use transfer learning from natural images to medical image processing [Bel+17] (Appendix A.2). Moreover as radiotherapists refer to organ atlas/map to segment CT-scan, we have proposed to learn dependencies in organ position by themselves [Ler+15] (Appendix A.1) using an output model, extending [HS06] strategy to the output space.

Tackle few available data through transfer Learning

The aim of the work published in [Bel+17] (Appendix A.2) was to select from all the slices of CT scan a special slice corresponding to the position of the third lumbar (Figure 3.9). This slice is called *L3 slice* and is a referee to anchor a CT or to compute biological indicator such as the sarcopenias index.

Due to medical compliance and privacy protection, dataset in medical imaging are usually scarce. To learn our model, we had very few labeled images, only 642 available samples in our L3CT1 database. Even if we use convolutional layers and a reduced number of parameters, it is too short to train a neural network.

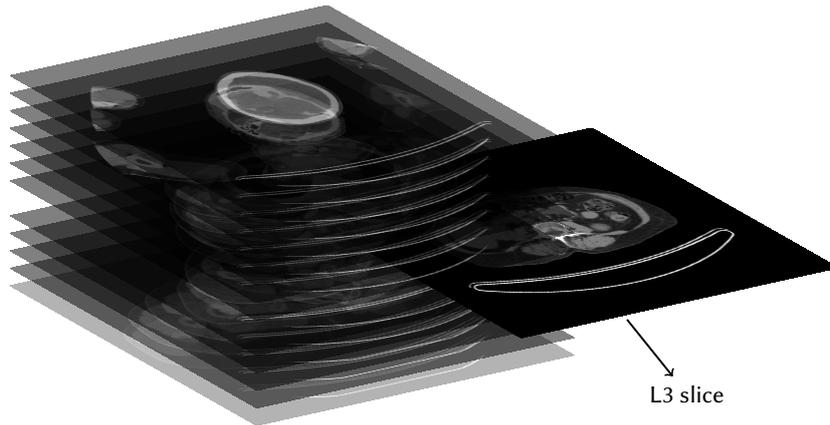


Figure 3.9: Guessing which CT slice corresponds to the third lumbar.

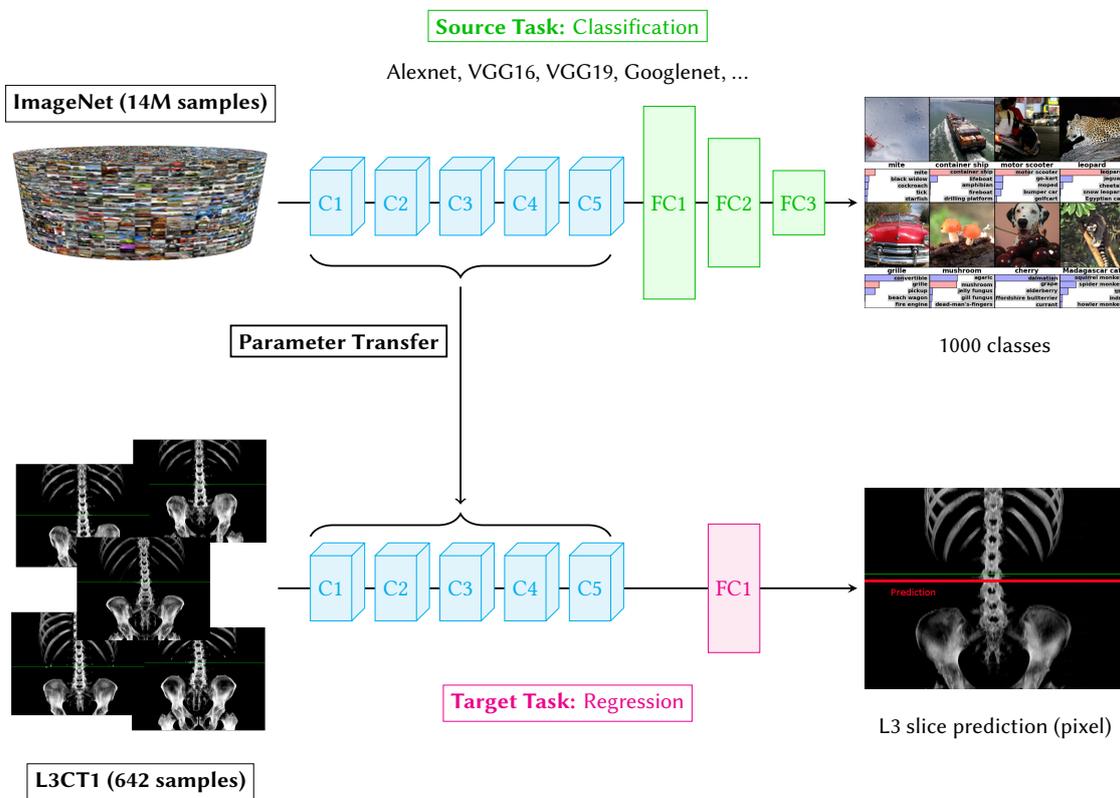


Figure 3.10: Transfer learning from a natural image classification to regression on medical images.

In order to overcome this lock, we have decided to use pre-trained convolutional layers from a model trained on a classification task on natural images and copy them to the deepest layers of our model for our regression task (Figure 3.10).

We achieve an error less than 2 slices in precision which is the same level of error a human specialist does. The details of the implementation and results are available in the original publication [Bel+17] (Appendix A.2).

Stacked Auto-Encoders extended to output space

In [Ler+15] (Appendix A.1), we proposed a deep neural network (DNN) architecture called Input Output Deep Architecture (IODA) to solve the problem of semantic segmentation in the context of medical images.

The originality of this work is to transpose DNN input pre-training trick to the output space, in order to learn a high level representation of labels $p(\mathbf{y})$. We extend the Algorithm 1 presented in the preceding chapter at sub-section 2.4.2. Not only input layers are pre-trained using encoder parts of stacked AE. This time output layers are also pre-trained using decoder parts of stacked AE. The resulting procedure is depicted in Algorithm 2.

Algorithm 2 Simplified IODA training algorithm

Input: \mathbf{X} , a training feature set of size $Nb_{\text{examples}} \times Nb_{\text{features}}$
Input: \mathbf{Y} , a corresponding training label set of size $Nb_{\text{examples}} \times Nb_{\text{labels}}$
Input: N_{input} , the number of input layers to be pre-trained
Input: N_{output} , the number of output layers to be pre-trained
Input: N , the number of layers in the IODA, $N_{\text{input}} + N_{\text{output}} < N$
Output: $[\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_N]$, the parameters for all the layers

Randomly initialize $[\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_N]$

Input pre-training

$\mathbf{R} \leftarrow \mathbf{X}$

for $i \leftarrow 1..N_{\text{input}}$ **do**

{Training an AE on \mathbf{R} and keeps its encoding parameters}

$[\mathbf{w}_i, \mathbf{w}_{\text{dummy}}] \leftarrow \text{MLPTRAIN}([\mathbf{w}_i, \mathbf{w}_i^T], \mathbf{R}, \mathbf{R})$

Drop $\mathbf{w}_{\text{dummy}}$

$\mathbf{R} \leftarrow \text{MLPFORWARD}([\mathbf{w}_i], \mathbf{R})$

end for

Output pre-training

$\mathbf{R} \leftarrow \mathbf{Y}$

for $i \leftarrow N..N - N_{\text{output}} + 1$ **step** -1 **do**

{Training an AE on \mathbf{R} and keeps its decoding parameters}

$[\mathbf{u}, \mathbf{w}_i] \leftarrow \text{MLPTRAIN}([\mathbf{w}_i^T, \mathbf{w}_i], \mathbf{R}, \mathbf{R})$

$\mathbf{R} \leftarrow \text{MLPFORWARD}([\mathbf{u}], \mathbf{R})$

Drop \mathbf{u}

end for

Final supervised learning

$[\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_N] \leftarrow \text{MLPTRAIN}([\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_N], \mathbf{X}, \mathbf{Y})$

This method was experimented in the segmentation of skeletal muscles on CT scans, a key but time consuming step of the computation of Sarcopenia indicator (loss of skeletal muscle mass). It may take more than 5 minutes for an experimented physician to compute it. The Sarcopenia indicator is linked to a good nutrition and good state of the fat/muscle ratio. It helps oncologists to propose an adapted treatment plan to the patient, as fatty or at the opposite undernourished subjects may have more side-effect to chemotherapy or radiotherapy.

Learning the correlation in labels $p(\mathbf{y})$ can be seen as learning an organ atlas, a common practice in medical training. Indeed, organs are not randomly positioned in the body. Thus when one kidney is found somewhere there is a high probability to also have the second one in the symmetrical position.

Standard ad-hoc methods such as Chung [Chu+09] is not able to cope with strong variability of patient's morphologies as it is based on a single average model, that is to say a single atlas. In opposition, our model embeds the variability of the patient morphologies through a learning process over the training set and gives better qualitative (Figure. 3.11) and quantitative results.

Extended quantitative results on both synthetic and real data are presented in the original publication [Ler+15] (Appendix A.1).

The same IODA strategy was also deployed for a preliminary work on facial landmark detection in [Bel+15b].

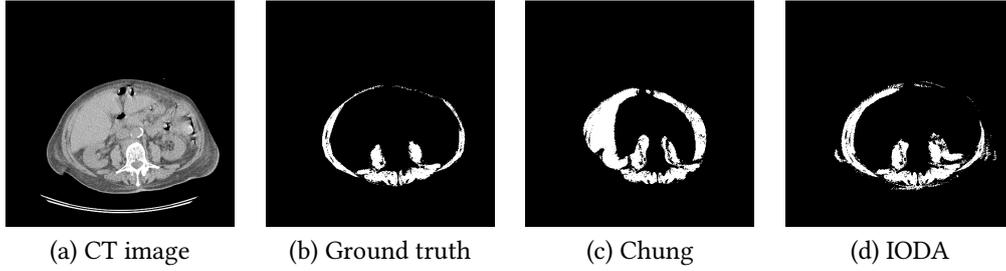


Figure 3.11: Patient with fatty mass making muscle mass segmentation difficult for ad-hoc methods such as in Chung [Chu+09].

3.2.3 Examples of concomitant learning

As presented in sub-section 3.1.2, Facial Landmark Detection is a typical structured output problem: the labeling consists in a graph of key points in the face (Figure 3.12). For a human, knowing partially the graph is sufficient to recover the full graph as she/he owns an *a priori* face model. This task is a perfect candidate for our framework, we have test with concomitant learning strategy using fixed or evolving weights in publications [Bel+16a] and [Bel+18] (Appendix A.3).

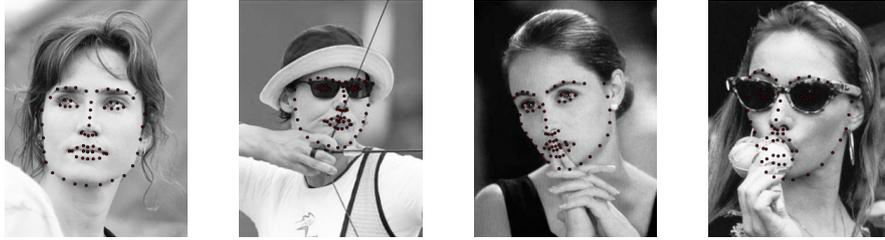


Figure 3.12: Examples of facial landmarks from LFPW [Bel+13] training set.

Here the models are separated like this:

- M_{in} , the input model, consists in learning face image representation,
- M_{out} , the output model, in its turn, tries to catch the structure and correlation in the facial landmark key points,
- M_{sup} , the main model, is the targeted task which aims at finding the key points from the picture.

The two side tasks for tuning M_{in} and M_{out} are unsupervised reconstruction tasks. This is a traditional multi-task scheme [Car97] that is hoping to help a main targeted task by side tasks, here the reconstruction of the input and the reconstruction of the output. The two side tasks are used as regulation for the main task.

Fixed weights

The first strategy to concomitantly train the model is to use fixed weights for the Equation 3.10.

Four different weighting schemes are presented in [Bel+18] (Appendix A.3):

1. λ_{in} and λ_{out} equal 0, this reverts to training the targeted task M_{sup} , as in a standard learning,
2. λ_{out} is null, this means that both the M_{in} and M_{sup} , i.e. $p(\mathbf{x})$ (the face image distribution) is mod-
elized to help get $p(\mathbf{y}|\mathbf{x})$ but not $p(\mathbf{y})$,
3. the opposite, λ_{in} is put to 0, $p(\mathbf{y})$ (the correlation in the key points) is learnt but not $p(\mathbf{x})$ concur-
rently to $p(\mathbf{y}|\mathbf{x})$.
4. no λ are null, all models are learnt.

The results of the last setup outmatches the results of the three firsts.

Evolving weights

One down side of the MTL framework presented in Equation 3.10 is that it introduced 3 hyper-parameters, the weights λ_{in} , λ_{out} and λ_{sup} , that need to be chosen, usually by cross-validation. Moreover, we are ultimately interested in learning the main model M_{sup} but not the side model M_{in} and M_{out} . Nevertheless, the same computation power is given to these three models all along the training.

That is why we have proposed to use an evolving weight scheme along the training epochs to grant importance to side tasks at the beginning of the learning but few on last epochs.

Sequential training (pre-training followed by fine-tuning) is nothing more than an evolving weight strategy but with step functions as temporal modulation (Figure 3.13): λ_{in} and λ_{out} are set to 0.5 and λ_{sup} to 0 until epoch E , then this last parameter for the supervised task is put to 1 and the two others for the unsupervised tasks to 0

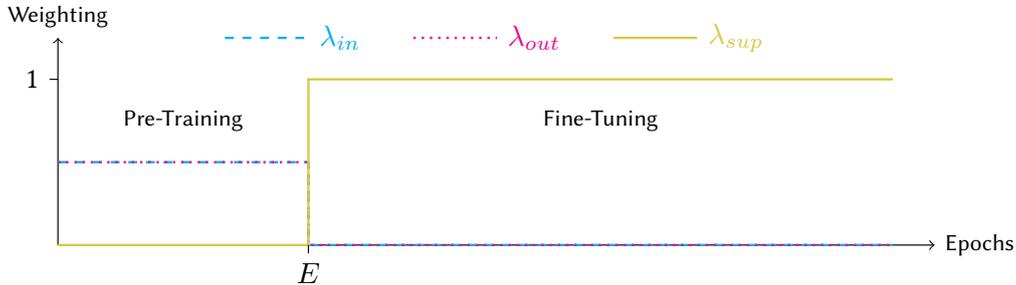


Figure 3.13: Sequential training as an evolving weight strategy.

A smoother version is to use linear evolution of side task weights, λ_{in} and λ_{out} as in Figure 3.14. Their value linearly decreases up until a certain epoch, epoch that could be different for both tasks. At its turn λ_{sup} starts from 0 and increases gradually in opposition to the two other weights. After λ_{in} and λ_{out} have extinguished (respectively at E_1 and E_2), only the main task remains, i.e. λ_{sup} equals to 1.

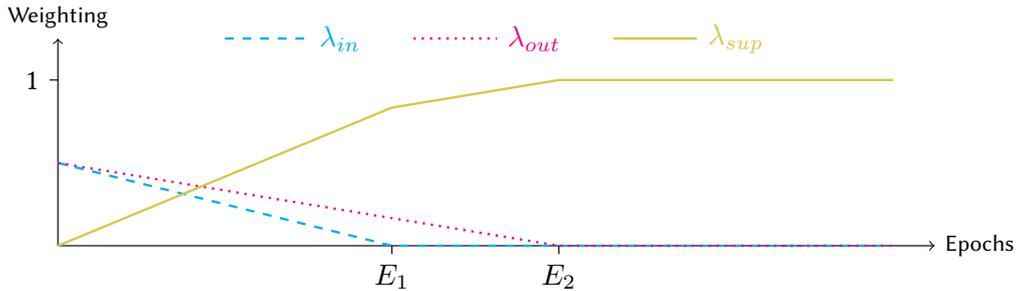


Figure 3.14: Linear evolution of weights.

More evolution scheme and corresponding results on the facial landmark detection problem could be found in [Bel+18] (Appendix A.3).

3.2.4 Perspectives and undergoing works

Since the publication of our works on SIO HD problem, advances occurred in representation learning and distribution modeling. Namely, we could take the benefit of new deep learning architecture such as Fully Convolutional Networks [LSD15] and Adversarial Learning [Goo+14].

Fully Convolutional Networks (FCN) are deep networks that do not include any fully connected layers, they suit very well tasks with images as input. It has the advantage that it can process no fixed size images and has very few parameters. U-Net presented in [RFB15] is built upon both AE and FCN, which enable it to have at the same time good high level features recognition and local precision. The input model M_{in} could take advantage of these two architectures.

Generative Adversarial Network (GAN) presented in [Goo+14] is perfectly suited as a generative model for HD or structured space. We could modify our framework to have a conditional or constrained GAN as the output model M_{out} .

In the next section, I shall present undergoing works on constrained GAN that could be in future work used in the presented HD/SIO framework, especially for the output model M_{out} .

3.3 Constrained deep generative models

Generative Adversarial Networks (GAN) are a special kind of deep generative models based on the min max game between a discriminator that aims at distinguishing between real or artificial samples and a generator that tries to fool it. They are presented in the chapter 2 at sub-section 2.5.2.

In this section, we will address two problems using extension of GAN: how to synthesize images with some constraint on their pixel and how to perform synthesis of polarimetric images for data augmentation. This is a first step in using GAN as an output model M_{out} in the HD/SIO framework.

3.3.1 Image synthesis/reconstruction with few constraint

Problem setup

We place ourselves in the context of a degraded version of the in-painting problem (Figure 3.15). In the traditional in-painting problem a structured part of an image has been erased and the purpose of the task is to reconstruct the full image from information known in the remaining part of the picture (Fig. 3.15b). In our setup, only scarce information of the original picture is known, such as some value of pixel in a constraint map (Fig. 3.15c). Nevertheless, to help us recover the original image, we own a dataset of images similar in nature to the image to recover.

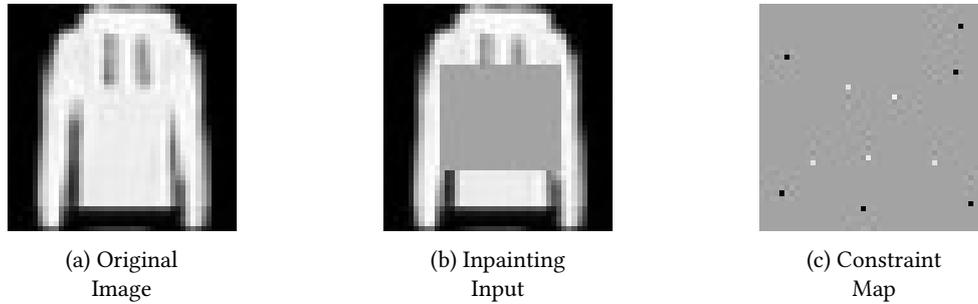


Figure 3.15: Difference between regular inpainting (b) and the problem undertaken in our work (c) from an original image (a).

For our experimentation we have used 3 different data set:

1. the fashion MNIST dataset composed of clothe articles (Fig. 3.15),
2. an ergodic texture dataset,
3. a real black and white image composed of ergodic underground hydrologic maps (Fig. 3.16).

The two first ones are toy sets to validate our methods, the latter is our real application where geologist aims at reconstructing unreachable underground maps from sparse probes from ground fields. When applicable, the ergodicity of the images will be enforced by the generator architecture.

VAE architecture and inversion problems

In [Lal+17], we first investigate ergodic image generation by using Variational Auto-Encoders (VAE). The architecture is composed of convolutional layers only as in fully convolutional networks. This enables all parts of the generated images to obey the same distribution. The VAE is trained on a small piece of a big underground water map that is used as a gold standard in geophysics publications.

The (low order) moments and connectivity indicators computed on the resulting artificial images are within the margin of the same indicators computed on the underground water map. This is the way to qualify the quality of the generated samples in this domain.

As such, the model does not contain any means to constrain certain pixels as in Figure 3.16. In this first work, we use the by-the-time standard for this kind of constrain problems : using a guess and try strategy in an inversion problem framework. The latent space \mathbf{z} is moved by a Monte-Carlo process up-until the resulting image respect the targeted constraints. Using a VAE reduces the exploration space of the inversion problem, which was an improvement in par with the state of the art of the time.

Even so, this is a slow process that we wanted to accelerate. We thought to back-propagate the constraints to the latent space \mathbf{z} with the parameters of the decoder network fixed to guide the problem inversion. Preliminary results with back-propagation were not satisfying and do not lead to publication.

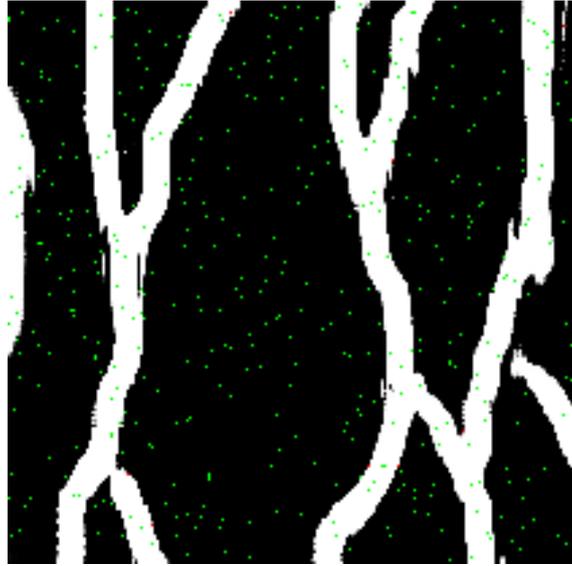


Figure 3.16: Underground water map with enforce pixels in green (achieved) or red (un-achieved).

GAN architecture and inversion problems

We follow our preliminary work but this time using a GAN and not a VAE as the deep model. As for VAE, the generator of the GAN is composed of only convolutional layers to enforce ergodicity of synthetic images, this is called a spacial GAN (SGAN) [JBV17]. To satisfy the constraints, we both tested Monte-Carlo Inversion strategy [Lal+17] and by back-propagation of the constraints to the LD space [Lal+19]. This time we achieve good results with both procedures.

The article *Training-image based geostatistical inversion using a spatial generative adversarial neural network* [Lal+17] was a breakthrough in using neural networks and GAN in the water resource community which leads to multiple linked works (more than 60 citations).

GAN architecture with constraint regularization

In the previous setup, we still need to adopt an inversion problem strategy to respect the constraints. Why not use a GAN that could directly output an image where constraint pixels are close to the correct value ?

In [Ruf+20] (Appendix A.4), we proposed to modify the GAN framework in the style of Conditional GAN [MO14] to input the constraint map to both the generator and the discriminator. Nevertheless, we explicitly added a loss term between the synthetic image from the generator and the constrain map (Figure 3.17). The loss is added to the training loss of the GAN as a regularization term through a weight parameter λ . This parameter acts as a trade-off between image quality (correct modelization of the real set distribution) and respect of the constraints (Figure 4 of [Ruf+20]).

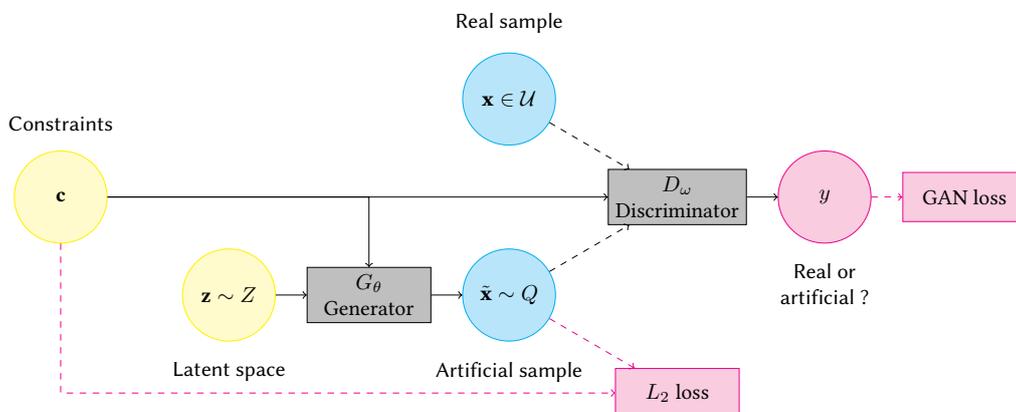


Figure 3.17: How to constraint a GAN with a pixel map.

3.3.2 Polarimetric conversion

Problem setup

Acquiring polarimetric images is a heavy process and few database are available. Why not use data augmentation and GAN ? Nevertheless, the problem relies in the fact that such modality has strong physical constraints.

Each pixel in polarimetric image is composed of 3 scalars forming a Stokes vector,

$$\mathbf{s} = [s_0 \quad s_1 \quad s_2]^\top, \quad (3.11)$$

where $s_0 > 0$ represents the total intensity, s_1 the amount of horizontally and vertically linearly polarized light and s_2 the amount of linearly polarized light at $\pm 45^\circ$.

It is important to note that by design, the Stokes vector is physically admissible if and only if the two following conditions are met,

$$s_0 > 0 \quad \text{and} \quad s_0^2 \geq s_1^2 + s_2^2. \quad (3.12)$$

Moreover, the Stoke vector is not directly recorded from a camera, it is obtained by a linear combination of grey level images recorded with different angles of the polarizer. For example, if the records occur when the polarizer is at rotation angles 0, 45, 90 and 135°, then the Strokes vector \mathbf{s} is obtained via the following formula,

$$\mathbf{s} = \mathbf{P} \cdot \mathbf{i} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} i_0 \\ i_{45} \\ i_{90} \\ i_{135} \end{bmatrix} = \begin{bmatrix} i_0 + i_{90} \\ i_0 - i_{90} \\ i_{45} - i_{135} \end{bmatrix}, \quad (3.13)$$

where $\mathbf{i} = [i_0, i_{45}, i_{90}, i_{135}]^\top$ are the actual intensities recorded by the camera at the different positions of the polarizer.

The Strokes vector \mathbf{s} contains all the polarimetric information and the intensity vector \mathbf{i} should be retrievable from it solely. That is, there exists a matrix \mathbf{A} such that,

$$\mathbf{i} = \mathbf{A} \cdot \mathbf{s} \quad \text{and} \quad (3.14)$$

$$\mathbf{i} = \mathbf{A} \cdot \mathbf{P} \cdot \mathbf{i}. \quad (3.15)$$

This matrix, \mathbf{A} , is a calibration matrix proper to the angles recorded by the camera.

A generated polarimetric images should obey equations 3.12 and 3.15 which respectively stand for the physical admissibility constraints and the calibration constraint.

CycleGAN and admissibility constraints

In order to leverage this problem, in the work submitted to ACCV [Bli+], we propose to synthesize polarimetric images from their RGB counter parts using a CycleGAN framework (cf. Sub-Section 2.5.2 of Chapter 2) where \mathbb{X} is the polarimetric domain and \mathbb{Y} the RGB domain.

In order to enforce physical admissibility (Eq.3.12), a rectified linear penalty is considered,

$$L_{phys} = \mathbb{E}_{\mathbf{y} \in \mathcal{Y}} \max(s_1^2 + s_2^2 - s_0^2, 0), \quad (3.16)$$

where \mathbf{s} is a pixel belonging to the synthesized polarimetric image converted in Strokes vectors, i.e. $\mathbf{P} \cdot \tilde{\mathbf{x}} = \mathbf{P} \cdot M_{\mathbb{Y}\mathbb{X}}(\mathbf{y})$.

Moreover, to enforce calibration properties (Eq. 3.15), a ℓ_2 distance between the synthetic polarimetric image $\tilde{\mathbf{x}} = M_{\mathbb{Y}\mathbb{X}}(\mathbf{y})$ and its calibrated reconstruction $\mathbf{A} \cdot \mathbf{P} \cdot \tilde{\mathbf{x}}$ is added to the optimization problem,

$$L_{cal} = \mathbb{E}_{\mathbf{y} \in \mathcal{Y}} \|\tilde{\mathbf{x}} - \mathbf{A} \cdot \mathbf{P} \cdot \tilde{\mathbf{x}}\|_2. \quad (3.17)$$

The framework is depicted in Figure 3.18. This framework has been tested with success on the KITTI dataset [Gei+13] to obtain new polarimetric images.

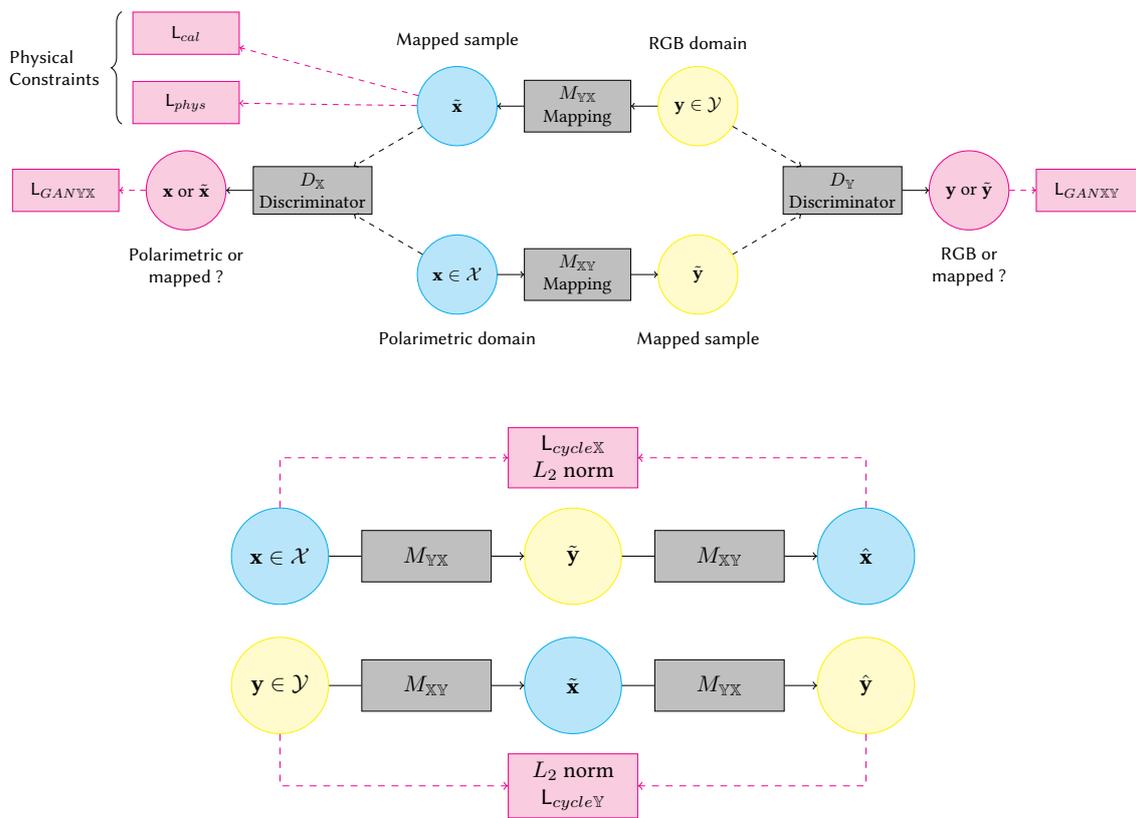


Figure 3.18: Adding admissibility constraints to the CycleGAN.

3.3.3 Sequence prediction

For now, we have only worked on static images for both HD/SIO and GAN frameworks. We are currently extending them to sequence prediction thanks to recurrent networks (such as LSTM) on the latent representation. This is a preliminary work on medical images for adaptive radiotherapy which has not led yet to publication.

The need of adaptive radiotherapy

External radiotherapy aims at treating cancer by irradiating malignant cells, causing irreparable damage to their DNA and eventually their death, as their ability to divide being impaired. However, ionizing beams used in external radiotherapy must also pass through healthy tissues surrounding the tumor, which may induce very side effects and non-negligible toxicity. The treatment is then analogous to a ballistic problem, where a trade-off must be found between maximally irradiating the tumor and minimally irradiating neighboring organs at risk (OARs).

In practice, dose optimization and treatment fractionation are the main keys to reach this objective. Fractionation over time spreads the delivery of the total dose into several sessions, allowing healthy cells to recover (they better repair their DNA than cancerous cells that divide continuously). Dose optimization, also known as treatment planning, deals with the geometric aspect of the problem and consists in focusing the beams on the target, while avoiding other tissues. The treatment plan must comply with a dose prescription on the target volumes and constraints on OARs. It can lead to about thirty sessions of irradiation at the rate of 5 sessions per week. Typically, Computed Tomography (CT) images with high resolution are used to segment the lesion, also called Gross Tumor Volume (GTV) and OARs. The beams are positioned on CT images in order to optimize the processing by calculating the absorbed dose in 3D according to intensity modulation of beams. Overall plan elaboration (OARs and tumor segmentation, dose calculation, optimization, and validation) takes several hours and can span several days.

In this scheme, the treatment plan is fixed before the 1st session and is kept unchanged during the treatment course. Nevertheless, during treatment, the patient can lose weight. The tumor often shrinks significantly (or grows, exceptionally), which can compromise treatment quality, with a volume that is then too large, too small, or mis-aligned if these changes also affect the tumor position. Figure 3.19 shows such evolution on a given patient. This can induce insufficient tumor irradiation, responsible for recurrence, and/or side effects to OARs.

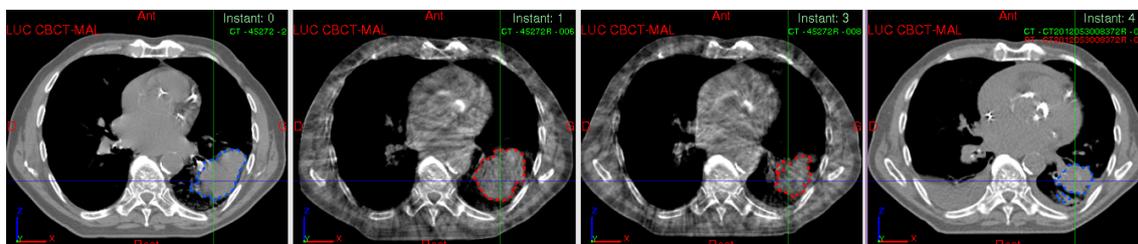


Figure 3.19: Examples of temporal exams of the same patient (NSCLC): planning CT, CBCT at the beginning of radiotherapy, CBCT at the end of radiotherapy, CT follow up 3 month after radiotherapy. Only GTV segmentations are shown (in blue for high resolution CT, in red for CBCT)

Nonetheless, we could take advantage of unexploited information. Indeed, in order to guarantee the reproducibility of the positioning of the patient and of the tumor, as well as its evolution, low resolution CT images are acquired regularly from one session to the other or weekly, typically using an imaging device embedded in the linear accelerator. However, this information is not yet taken into account in the treatment plan.

For a given patient, improvement is possible by personalizing the treatment plan and adapting it to the evolution of the tumor and OARs in the course of its delivery. This research direction, coined adaptive radiotherapy (ART) aims at taking into account tumor and OARs changes occurring during treatment. As the tumor response to the treatment requires its delivery not to be interrupted, and due to the long duration of treatment simulation, anticipation of any necessary treatment adaptation is a major health issue.

Therefore, it would be highly beneficial for the patient and the medical staff to trigger any adaptation as soon as possible, in order to ensure a seamless treatment delivery and avoid rescheduling.

DEEP learning in Adaptive Radiation Therapy (DEEPART) project

We propose to use deep learning methods to predict sequences of CT (possibly along with the GTV and OAR segmentation) over the treatment sessions. This project involves peoples from GREYC (Caen) and LITIS (Rouen) machine learning teams as well as from Centre François Baclesse (Caen), Centre Henri sBecquerel (Rouen) and MIRO (Brussels) radiotherapy / medical imaging teams. It is supported by the Normandy region under its label of excellence initiative (MINMACS program).

Figure 3.20 presents a global overview of the DEEPART system. It can be described as follows:

Inputs that contain the high-resolution CT involved in treatment planning, with the contours of the GTV and OARs, as well as a sequence of daily or weekly low-resolution CT images used for patient positioning at each treatment session.

Prediction system that involves deep neural networks (DNN) designed to process sequences (of different temporal size) in a generative mode (i.e. producing an image or a sequence of images). The idea is to use a Long Short Term Memory network (LSTM) to condition a FCN decoder or a GAN (Figure 3.21).

Outputs that represents an estimation of the low resolution CT that would be acquired at the next radiotherapy session. Moreover, if the GTV and OARs segmentation are available in the input image, then the system may also return the predicted GTV and OARs on the estimated CT.

This predictive system and workflow are evaluated on two different types of tumor sites: head and neck cancers (H& N), and non small cell lung cancer (NSCLC).

By the time of writing, Nikolaos ADALOGLOU, the research engineer working on the project, has implemented the system using 2D MIP projection of the CT and FCN as encoder/decoder. The 2 next steps are to switch from FCN to constrained GAN for the decoder part and from 2D MIP to full 3D CT. Self-supervised learning is prospected for training the encoder/decoder FCN in order to cope with the few data available.

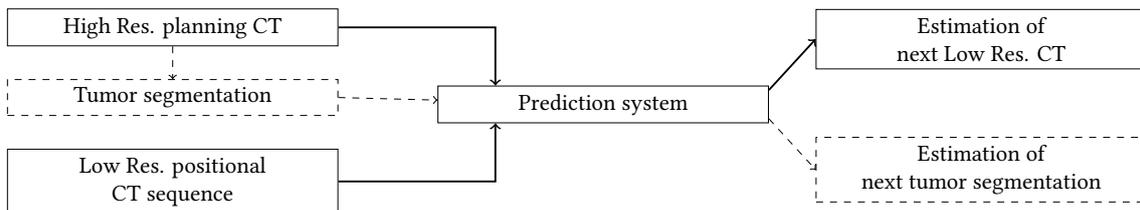


Figure 3.20: Global view of the proposed DEEPART system

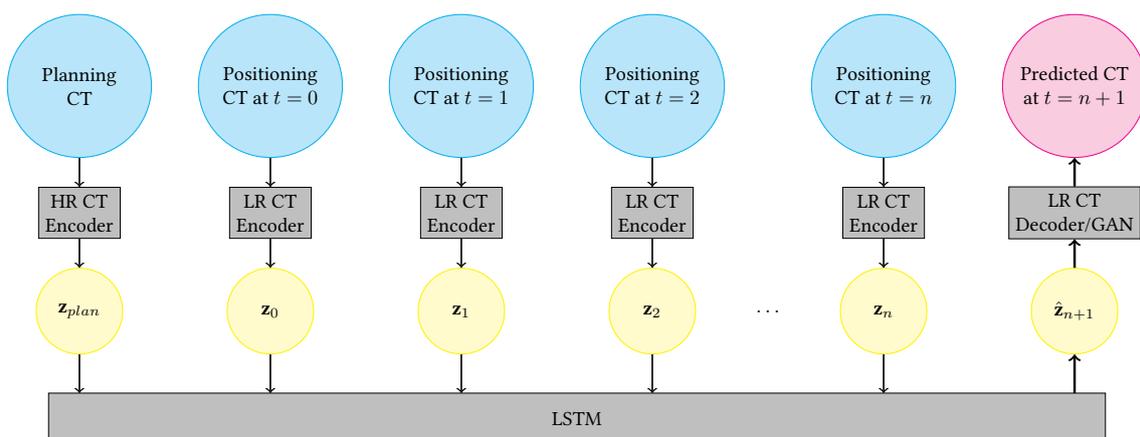


Figure 3.21: Detailed implementation of the prediction system by the mean of LSTM and GAN

Chapter 4

Machine Learning applied to Human Movement Science

This chapter will be dedicated to diverse applications of Machine Learning to Human Movement Science.

Soon after my installation at LITIS in 2008, Pr. Ludovic SEIFERT, a colleague from the CETAPS laboratory¹, contacted me to establish a collaboration at the border between Human Movement Science and Statistical Learning.

CETAPS researches are dedicated to the studies of physical and sports activities. Most notably, one of its aims is to understand how performance and efficiency emerge from training. Academics at CETAPS were used to applied statistics and modeling techniques but they wanted to investigate how Machine Learning was good to mitigate recurring problems in Sport Science and notably Human Movement Science such as study of inter- and intra- individual variability, before, during and after expertise acquisition.

I'm interested in this partnership as their fundamental interrogation on how a human could be trained for sport activities meet questions of the machine learning community, most notably on the bias/variance undercapacity/over-fitting dilemma.

The following *Movement as dynamical system* section will present you why the study of the variability is an important question in Human Movement Science [vEvW00; Dav+03; BWR07]. The next section *Movement profiling* depicts three works [Sei+13a] (Appendix A.5), [Hér+17] (Appendix A.6), [KHS14] (Appendix A.7) where Machine Learning has been applied to help Human Movement Science. The last section of this chapter presents a study on *Gait Recognition* [Rid+17] (Appendix A.8). It was not part of our partnership with CETAPS but was inspired by Human Movement knowledge acquired by this collaboration.

Contents

4.1	Movement as dynamical system	87
4.1.1	Importance of the variability in Human Movement	88
4.1.2	Human Movement open questions	89
4.1.3	Why use Machine Learning ?	89
4.1.4	Parallels between human training and machine learning	89
4.2	Movement profiling	90
4.2.1	Change point detection	90
4.2.2	Climber performance evaluation	94
4.2.3	Swimming cycle clustering	98
4.3	Perspectives on Machine Learning applied to Human Movement	101
4.4	Gait recognition	102
4.4.1	Context	102
4.4.2	Proposed framework	102
4.4.3	Results and perspectives	103

¹<http://cetaps.univ-rouen.fr/>

Selected publications

[Sei+13a] Ludovic Seifert et al. “Temporal Dynamics of Inter-Limb Coordination in Ice Climbing Revealed through Change-Point Analysis of the Geodesic Mean of Circular Data.” In: *Journal of Applied Statistics* 40.11 (Nov. 2013), pp. 2317–2331. doi: 10.1080/02664763.2013.810194. URL: <https://hal.archives-ouvertes.fr/hal-02094911> (Appendix A.5)

[Hér+17] Romain Hérault et al. “Comparing Dynamics of Fluency and Inter-Limb Coordination in Climbing Activities Using Multi-Scale Jensen–Shannon Embedding and Clustering” In: *Data Mining and Knowledge Discovery* 31.6 (Nov. 2017), pp. 1758–1792. doi: 10.1007/s10618-017-0522-1. URL: <https://hal.archives-ouvertes.fr/hal-02094958> (Appendix A.6)

[KHS14] John Komar, Romain Hérault, and Ludovic Seifert. “Key Point Selection and Clustering of Swimmer Coordination through Sparse Fisher-EM.” in: ECML/PKDD 2013 Workshop on Machine Learning and Data Mining for Sports Analytics (MLSA2013). Jan. 7, 2014. arXiv: 1401.1489 [physics, stat]. URL: <http://arxiv.org/abs/1401.1489> (Appendix A.7)

[Rid+17] Imad Rida et al. “Improved Model-Free Gait Recognition Based on Human Body Part.” In: *Biometric Security and Privacy: Opportunities & Challenges in The Big Data Era*. Ed. by Richard Jiang et al. Signal Processing for Security Technologies. Cham: Springer International Publishing, 2017, pp. 141–161. ISBN: 978-3-319-47301-7. doi: 10.1007/978-3-319-47301-7_6. URL: https://doi.org/10.1007/978-3-319-47301-7_6 (Appendix A.8)

4.1 Movement as dynamical system

The analysis of the movement as a dynamical system comes as far as the work of a Russian Physiologist, Nikolai Bernstein, in the 1960's [Ber66]. In this point of view, the human body can be thought as a bio-mechanical apparatus where movement chains depend on actuators controlled by parameters or Degrees Of Freedom (DOF).

Degrees Of Freedom (DOF) can be defined [NV01] as

the number of independent coordinates required to uniquely describe the configuration of a system.

For example, the arm chain can be decomposed into 7 parameters :

- 3 DOF at the shoulder,
- 1 at the elbow,
- 3 at the wrist.

Figure 4.1 illustrates the position of the seven degrees of freedom in the arm kinematic chain.

These seven DOF are far more than needed to perform most of the arm movements. Indeed, the variation of these parameters or degrees of freedom are correlated and they lie on manifold constrained by the body limits. The correlation can be broken when an *error* appears on an actuator, the remaining correlated DOF reorganize themselves to achieve the movement. Thus, the body is able to recover from a glitch by the redundancy of the DOF.

The hidden parameters, on which the DOF depend, are sometimes referred as *Virtual DOF* or *coordinative structures*. For a particular task, the DOF and/or *Virtual DOF* will stay on a restricted part of their manifold or will describe a reproducible path in this manifold. This path or less precisely the restricted part of the manifold is called a *coordination mode* for the task.

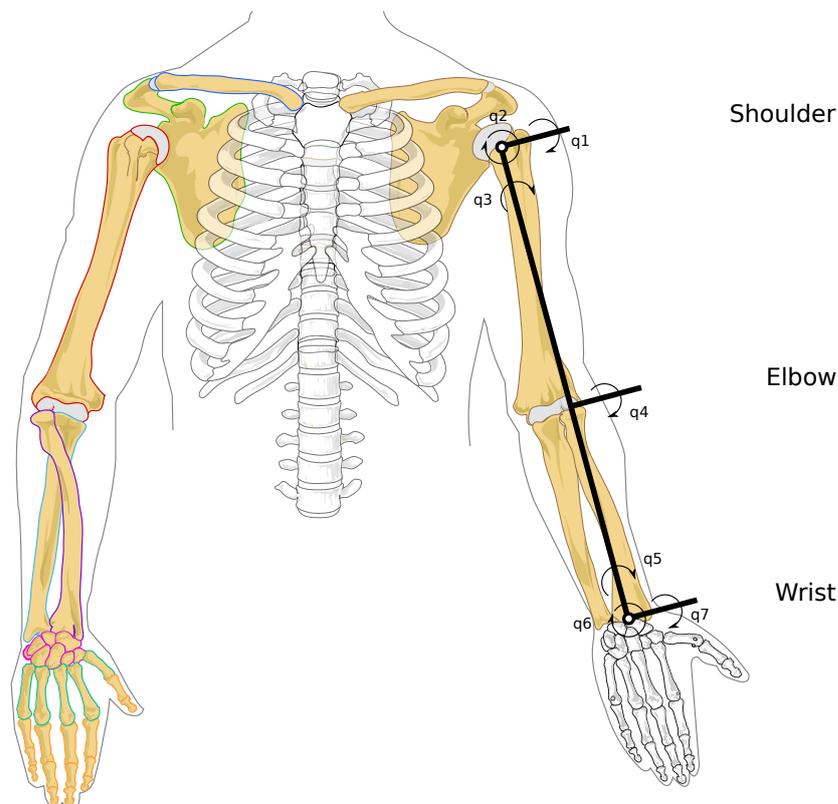


Figure 4.1: Illustration of the 7 degrees of freedom in the arm kinematic chain.

Source Wikimedia ^a

^aModified version of public domain LadyoffHats figure, https://commons.wikimedia.org/wiki/File:Human_arm_bones_diagram.svg

4.1.1 Importance of the variability in Human Movement

According to the dynamical system framework, non-expert individuals have few coordination structures and their parameters rely on a large manifold. The body does not yet know how to organize the bio-mechanical chain to perform efficiently the given task and thus is composed of a lot of degrees of freedom [Ber66; NV01].

In the first stage of the training, unwanted body and/or virtual DOF in the kinematic chain are progressively frozen. The reduced explored manifold leads to a more efficient coordination mode.

Trained individuals have low intra-variability and the remaining variability could be seen as a system noise that should be reduced to increase efficiency [BWR07]. Moreover, one optimal coordination mode should be available for a given task, also reducing inter-individual variability.

Effectively, in experimentation, experts shows lower inter and intra individual variability than non-expert or recreational practitioners. This tends to the idea that experts should react as an automaton and always respond by the same action to the same stimulus.

Nevertheless, some phenomena do not match this model [vEvW00; NV01; Dav+03; BWR07; Sei+11a].

Experts with low variability are not able anymore to adapt to new/changing environments. For example, expert pool swimmers perform very badly in free waters. Moreover, recreational practitioners with low variability behavior profit less from training: they hardly gain in efficiency and performance.

In addition, an inverse phenomenon arises when you look at elite subjects. Elites are subjects playing at a national or international competition level. For a given task, each player has developed multiple efficient coordination modes leading to higher intra-individual variability.

To sum-up, the training and the performance skill curve shows an hour glass shape (Fig. 4.2) where intra and inter individual variability is high for non-experts, low for experts, and higher intra-individual variability for elite practitioners [Sei+11a].

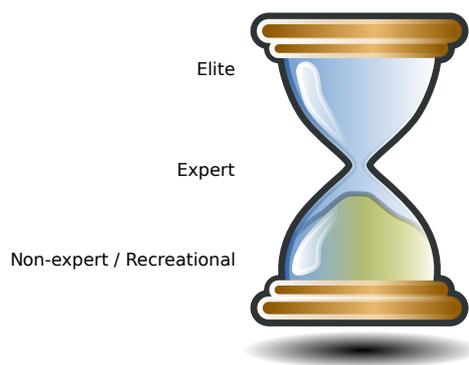


Figure 4.2: Schematic representation of amount of variability as a function of expertise
Source Wikimedia ^a.

^aModified version of cc-by-sa 3.0 RRZEicons figure, https://commons.wikimedia.org/wiki/File:Hourglass_2.svg

Contrary to the idea of considering movement variability as a plague that should be reduced in order to increase efficiency, an ecological point of view will see variability as a pre-condition to adaptation and thus to learning as well as to efficiency [vEvW00]. For example, elite subjects converge to multi modal behavior. This enables them to choose between a repertoire of movements to adapt their behavior to external conditions, such as change in the nature of the playground for the tennis player, or internal conditions, such as strain, mental stress, . . .

Moreover, recent studies show that intra-variability reduces injuries risk by involving different tissues in the repetition of the movement [BWR07].

4.1.2 Human Movement open questions

Thereby, studying variability is of high interest for Human Movement Science. Nevertheless, most of the studies available at the beginning of the collaboration with CETAPS were static studies or studies where coordination modes were only investigated on a reduced number of limbs seen as two oscillators [HHM00; Bar+02; TD07]. In these experiences, recreational/non-expert, expert and elite subjects were observed independently. In addition, few works were undertaking an investigation on the dynamic of the variability during the training.

To this point, we can list open questions that need to be answered (partially quoted from [BWR07]):

1. What is the link between intra and inter individual variability ?
2. How do coordination modes evolve during training at each skill levels ?
3. How to separate *bad* variability from recreational subjects to *good* variability as displayed by elite practitioners needed for adaptation ?
4. How instructions given during training or the training environment itself can temper or favorise the acquired variability as well as the induced adaptability and efficiency?

We do not claim that we can definitively answer these long-term questions of Human Movement Science but rather we want to address them from a new point of view by using Machine Learning techniques.

4.1.3 Why use Machine Learning ?

How Machine Learning could help answering the aforementioned open questions ?

First, as stated in previous sections, coordination is a highly multi-modal phenomenon so interpretations of standard statistical indicators are hard. On the other side, clustering seems a good and simple manner to detect these coordination patterns.

Secondly, not all the possible coordination modes are known from Human Movement Science experts. Moreover, *a priori* knowledges on the human movement may lead to choice bias among possible coordination modes. Therefore, we aim to work in an unsupervised manner with few/no ad-hoc features extraction so that behaviors/coordination modes unknown or pent-up by the human movement community can arise more objectively.

Lastly, the processed data may lie in non-euclidean spaces such as angle in a circular manifold. Machine Learning methods can easily address different data types providing the use of adapted similarities and losses.

4.1.4 Parallels between human training and machine learning

Interestingly, one can build links between the role of the variability in Human Movement and in Machine Learning.

Indeed, some recreational/non-expert with low variability are unable to be trained; it can be seen as a low capacity model unable to learn a task. No degree of freedom in both cases means no learning capabilities.

Moreover, experts with low variability did not seem to be able to adapt to new contexts. Thus, they can not generalize their body knowledge to a new situation. This can be viewed as an over-fitting phenomenon where a model performs well on known examples from the training set but poorly on new examples.

Changing/demanding environments, strict or open given instructions can therefore be seen as noise or regularization to help the training.

4.2 Movement profiling

In this section, I will present three main works from our collaboration with CETAPS, all about *movement profiling*.

The first work aims at a local analysis of the variability in ice climbing through change point detection [Sei+13a] (Appendix A.5); the second targets the discovery of a link between coordination mode evolution and fluency evolution in artificial wall climbing for climber performance evaluation [Hér+17] (Appendix A.6); the last one stages the uncovering of coordination mode in breakstroke swim using swimming cycle clustering [KHS14] (Appendix A.7).

They are not organized in chronological order of publication but rather thematically so as to ease the understanding of the reader.

4.2.1 Change point detection

This study on a local analysis of the variability of ice climbers was published in the Journal of Applied Statistics [Sei+13a] (Appendix A.5). It is the fruit of the collaboration between CETAPS, LITIS in Rouen, Laboratory Jean Kuntzmann in Grenoble and the Queensland University of Technology in Brisbane, Australia and involves skills in Human Movement Science, Geodesic Mathematics and Statistical Learning.

Context

According to Wikipedia²,

Ice climbing is the activity of ascending inclined ice formations. Usually, ice climbing refers to roped and protected climbing of features such as icefalls, frozen waterfalls, and cliffs and rock slabs covered with ice refrozen from flows of water.

It is a particular climbing activity in that sense that,

1. Ice falls are non-reproducible structures that can evolve over time, and that,
2. Climber can dig his own anchorages in the ice.

It results in a high interaction between the performer and his environment both having influence on each other. Thus, climbers are pushed to exploit affordances, i.e. possibilities for action offered by a particular performance environment [Gib14], in order to gain in fluency and performance.

In this situation, non-expert/recreational climbers tend to concentrate to stabilize their body position in order not to fall, whereas experts probe the environment in order to find a good place to dig an anchorage, for example. As a consequence, affordances induce variation in motor coordination patterns in expert climbers.

Therefore, contrary to the general rules of dynamical system, experts exhibit more variability than recreational climbers.

Study

Seven expert climbers and eight beginners took part in the study. The collected data consist in time series of two angles:

1. the angle formed by the upper limb anchorages (picks) and the horizontal,
2. the angle formed by the lower limb anchorages (studs) and the horizontal.

Figure 4.3 describes the recorded angles.

As experts display a greater movement repertoire, a global analysis of the variance on a full climb would not enable us to distinguish in details between expert and beginner behavior. That's why we intended to perform a local analysis of the variability along the climb.

²https://en.wikipedia.org/wiki/Ice_climbing

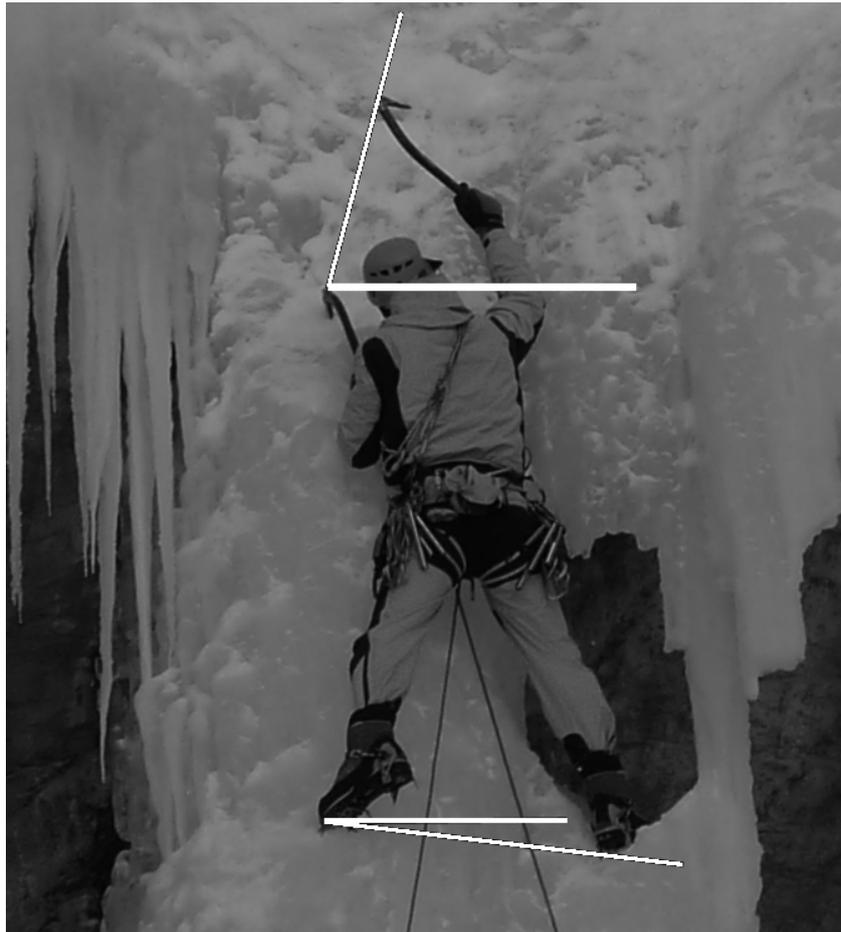


Figure 4.3: Angular position of the limbs

Proposed framework and contribution

Climbing is not a cycling activity like, for example, swimming, there is no immediate semantic cutting available. Hence, in order to undertake a local analysis, we have to segment a climb into shorter time sequences representing each a coherent portion of a climb. Then the variability will be assessed on each segment.

One problem we faced on these particular data is that they are composed of angular data. Hence, the notion of mean and variance should be taken with great care. If one uses a scalar mean between the angle, the result will depend on the chosen reference as depicted in Figure 4.4.

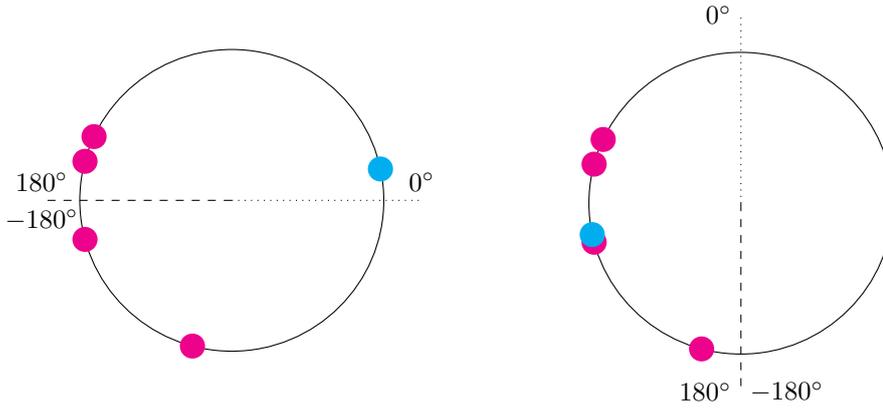


Figure 4.4: Scalar mean (in blue) of angular points (in red) depends on the chosen reference

In order to alleviate this problem, we used geodesic indicators based on the geodesic distance between the two angles α and β ,

$$d_G(\alpha, \beta) = 180^\circ - |180^\circ - |\alpha - \beta|| .$$

This distance corresponds to the shortest path in the circle manifold between the two points whose coordinates are α and β (Fig. 4.5).

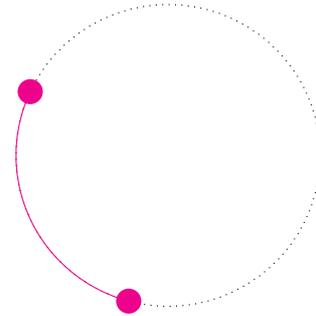


Figure 4.5: Geodesic distance between two points on the circle manifold

From this distance, we can derive the definition of the geodesic mean and variance,

$$\hat{\mu}_G = \underset{\mu \in S^1}{\operatorname{Argmin}} \frac{1}{n} \sum_{i=1}^n d_G(y_i, \mu)^2 ,$$

$$\hat{\sigma}_G^2 = \frac{1}{n} \sum_{i=1}^n d_G(y_i, \hat{\mu}_G)^2 .$$

The geodesic mean is chosen to give the lowest geodesic variance. The variance is unique by definition; nevertheless the mean may be multiple. For example, with two points at 0° and 180° , the means are at -90° and 90° as these two angles lead to the same minimal variance (Figure 4.6).

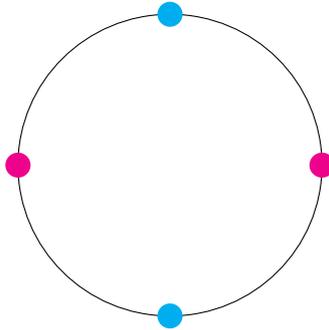


Figure 4.6: Two possible means (cyan) of a set of two points (magenta).

In order to segment the angular signals, we will use a filtered derivative method [BN93] which generally consists in computing a local estimation of a parameter of interest via a *filter* like a mean or a M-estimator, and in detecting changes in these local estimations through *derivation*. Figure 4.7 exhibits how to compute the drift at time k between the estimation parameter $\hat{\theta}$ computed on the \mathbf{x} signal on A_1 points before k and A_2 points after k .

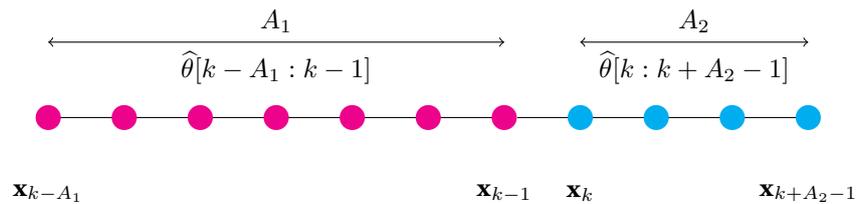


Figure 4.7: Detection of change in a parameter of interest through derivation

Our contribution resides in using geodesic mean as the parameter of interest and geodesic distance as measure of the drift when the signal consists in an angular time series.

Results and interpretation

By using the filtered derivative segmentation, we obtained for each climb a set of change points in the climber behavior occurring during the climb.

Most notably, we have identified that

- Experts have more changing points than non-experts, and,
- Non-experts spent a lot of time without any movement of the limbs.

By looking qualitatively, segment by segment, some facts can be noticed:

- Non-experts had to repeat punches to create an anchorage (2 punches for pick and 5 punches for stud),
- Non-experts do want and spend more time to stabilize their body,
- Non-experts take longer time to catch the next grasp.

Indeed, non experts don't know how to use affordances in the icefall.

In consequence, this study participated to the development of the open question how to separate variability from non-expert subjects to the acquired variability needed for adaptation by expert/elite subjects.

4.2.2 Climber performance evaluation

The past study has shown us the interest of using Machine Learning techniques into climbing by studying the variability of limb angles along an icefall. We wanted to go further and detect higher semantic level coordination behaviors. In order to do so we have proposed to cluster segment by segment climbs, considering the resulting cluster as a coordination mode and then trying to establish a link between clusters and performance [Hér+17] (Appendix A.6).

Context

The efficiency of a climb can be assessed by the fluency or smoothness of the ascent [Sei+14b]. The fluency is computed for instance by the entropy or by the *jerk* which is the integration of the absolute value of the acceleration. Figure 4.8 shows the decrease of the *jerk* trial after trial, depicting increasing performance along the training.

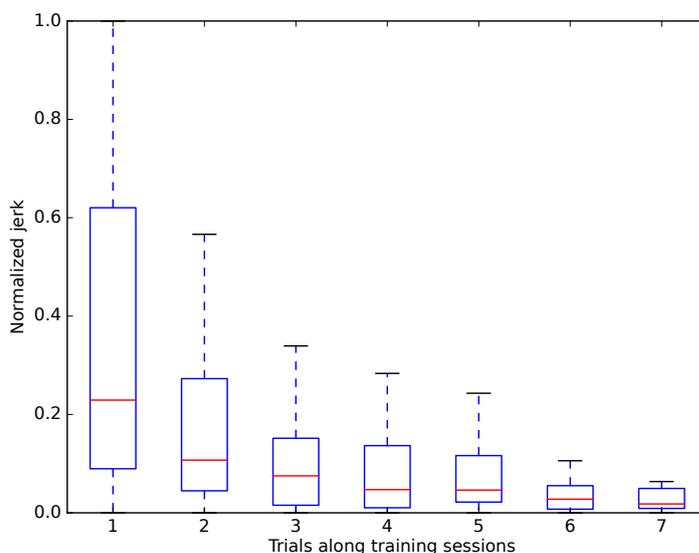


Figure 4.8: Jerk distribution, trial after trial

Nevertheless, smoothness indicators such as the jerk are usually computed globally on the body, not taking into account limb coordination. It is then hard for a practitioner to interpret his own fluency evolution and thus to increase his climbing efficiency. Therefore we proposed to use a clustering based on limb orientation rather than body activity to detect explainable coordination modes.

In taking a *Human Movement* perspective, the novelty of this work is to adapt machine learning methods to overcome current methodological limitations in linking movement variability with performance over the timescale of practice and at the individual level of analysis. Namely, we had addressed 3 objectives,

1. to go on **full body analysis**, taking into account the 4 limbs and related trunk movement; in order to do so, we have reduced the dimension of the data-set to visualize the climbing actions into features and categorize these by clustering.
2. to analyze how the clusters are distributed in time, i.e. to address the **dynamics of learning at the behavioral level**, in order to know whether some patterns are present at the beginning of the learning process, which could correspond to the existing repertoire; while other clusters appear later in the learning process, emerging through exploratory processes.
3. to analyze the **individual specificity** during learning. We expect that some participants learn faster than others, meaning that they switch more rapidly to a new pattern because they demonstrate a more effective exploration. Conversely, we also expect that some participants will exhibit a tendency to resist to change. Thus, we anticipate a link between the emergence (or lack thereof) of new actions and the improvement in performance. The latter is suggesting that putting, together the dynamics of the climbing fluency (performance outcome) and the dynamics of behavioral skills acquisition might reveal whether exploration is effective or not.

Proposed framework

From a *Machine Learning* point of view, this work does not propose any new dimension reduction nor clustering techniques. Nevertheless, due to the nature of the data (temporal signal, 3D rotations ...), we adapted unsupervised methods with special care (geodesic distance, ...).

We have first cut the recorded signals into coherent segments using the method that we have previously developed in [Bou+16]. For each of the four limbs and the hip, we detect if a sensor is immobile or moving using a CUSUM algorithm. Using the aggregation of the segmentations on each 5 sensors, based on heuristical rules given by human movement experts, a global body state is determined among:

- Immobility,
- Postural regulation
- Hold Exploration,
- Hold Change,
- Traction.

On the same segmentation, a clustering is performed this time not using sensor activities nor the determined body state but only limb and hip orientations. This separation between body activities and limb orientation is needed as we want to investigate the link between performance (body activity) and coordination (limb orientation) and so we must not introduce correlation between them before doing the clustering. A Gaussian mixture model was preferred over HAC, which can hardly handle the number of samples we have, and over k-means, which is not able to manage clusters with different spreads.

Moreover, in our climbing data, structures are unknown and may appear on different scales: climbers, holds, paths, climbing order, learning curve ... Nevertheless, standard clustering or dimension reduction methods, such as Stochastic Neighbor Embedding (SNE), are known to be good at structure preservation only for a particular scale. Recently, Multi-Scale Jensen-Shannon Neighborhood Embedding (MsJSE) [LPV15] solves this problem by opting for multi-similarity approaches. This Multi-Scale method will be applied to the output of motion sensors in order to help the visualization of behaviors even if they appear at different scales.

Results

Figure 4.10 exhibits a typical evolution of the fluency, body states and clustering for one climber along a practicing session. The decrease of the fluency indicators (Jerk, Entropy, Immo ratio in first plot) starting at segment 325 indicates a gain in performance. In the meantime you can see a reorganization of the body states (second plot) and the coordination modes found by the clustering (third plot). Most notably, we can notice a switch between coordination modes C1 and C2 to coordination modes C8 and C12.

Moreover, we would have expected that fluency would have a direct link with clusters, i.e. the coordination behavior. But, surprisingly, this climber comes back to his beginner repertoire (C1 and C2 coordination mode) between segment 375 and 400 but this change did not impact its fluency. Please refer to [Hér+17] (Appendix A.6) in order to have a detailed analysis of the clustering and its link with fluency.

Let's look at prototypes of limb orientations cluster by cluster. For the aforementioned clusters, the mean body position is illustrated in Figure 4.9. The new patterns show the trunk going from an orientation with front of the body facing the wall (clusters 1 and 2; Figures 4.9a and 4.9b), to more of an oblique orientation (clusters 8 and 12; Figures 4.9c and 4.9d). Additionally, clusters 8 and 12 differ to each other in so far that the feet are orientated either in a pigeon toed fashion, or where the outer edge of the foot is orientated to be used as support.

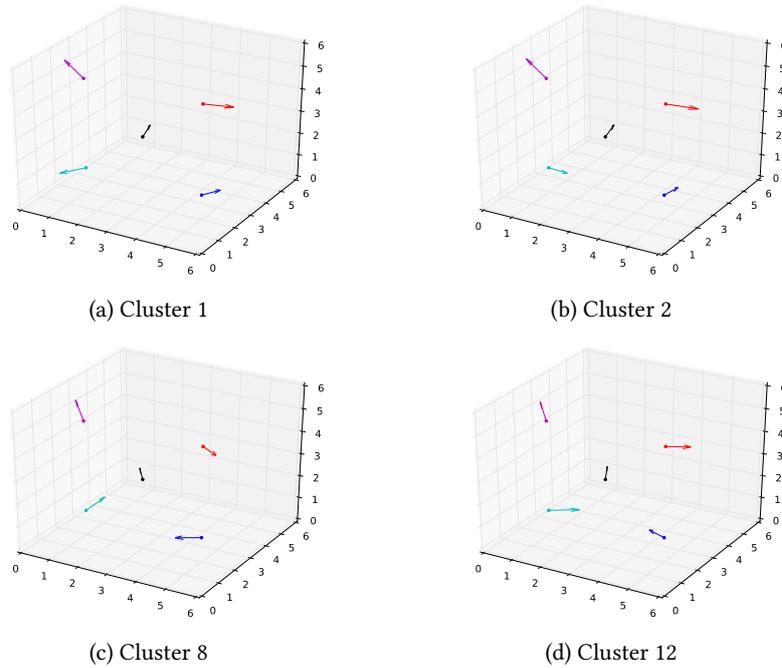


Figure 4.9: Sensor orientations for selected clusters

Perspectives

Cluster analysis appeared as a promising way to investigate the dynamics of climbing practice in order to highlight the individual pathway of learning. Indeed, it outmatches past studies based on oscillator models [HHM00; Bar+02; TD07] by taking into account the full body dynamics and not only a sub-set of limbs (namely, upper limbs). It has helped to investigate the open question on *how do coordination modes evolve during training at each skills levels*.

In particular, clustering of discrete activity states enables us to discover learning dynamics and changes in orientation and dynamics of lower limbs and trunk along the time-scale of ongoing practice, and, specifically, changes that coincide with more fluent traction. Interpretation of each climber cluster time-line highlights individual specificity such as a lack of acquisition during practicing, blind search and exploration followed by temporary return to original repertoire.

Moreover, we could investigate the link between the behavioral skill and fluency: the coordination time-line (obtained through clustering) is not adequately described as linear and proportional to the climbing fluency.

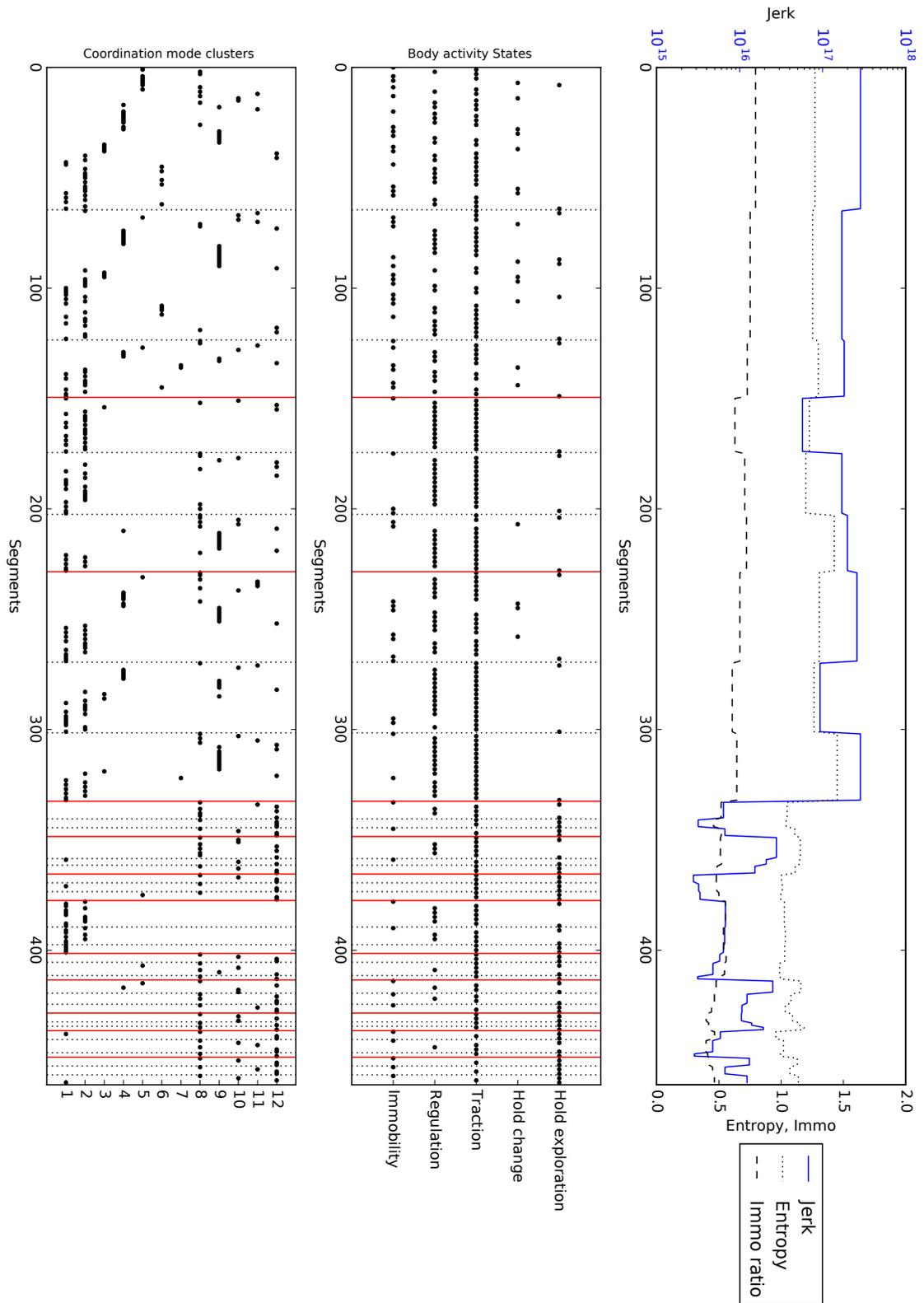


Figure 4.10: An example of fluency indicators, body state segmentation, and clustering along multiple trials during the training of one climber

4.2.3 Swimming cycle clustering

This work [KHS14] (Appendix A.7) is dedicated to breaststroke swimmers. Contrary to previous studies, the swimming pool represents a controlled environment with the same affordances at each trial. Indeed, in this study we want to investigate more precisely the learning process of the arm and leg coordination and not the ability of a performer to adapt to its environment and to gain from affordances.

This work took part in the first experimentations of the then PhD student John KOMAR on the application of Machine Learning to Human Movement Science which eventually led him to focus on these tools for his own PhD research [Kom13].

Context

In breaststroke swimming, achieving high performance requires a particular management of both arm and leg movements, in order to maximize propulsive effectiveness and optimize the glide and recovery times [Sei+10a]. Therefore, expertise in breaststroke is defined by adopting a precise coordination pattern between arms and legs (i.e. a specific spatial and temporal relationship between elbow and knee oscillations). Continuous relative phase between elbow and knee is typically used to measure the coordination (Fig. 4.11).

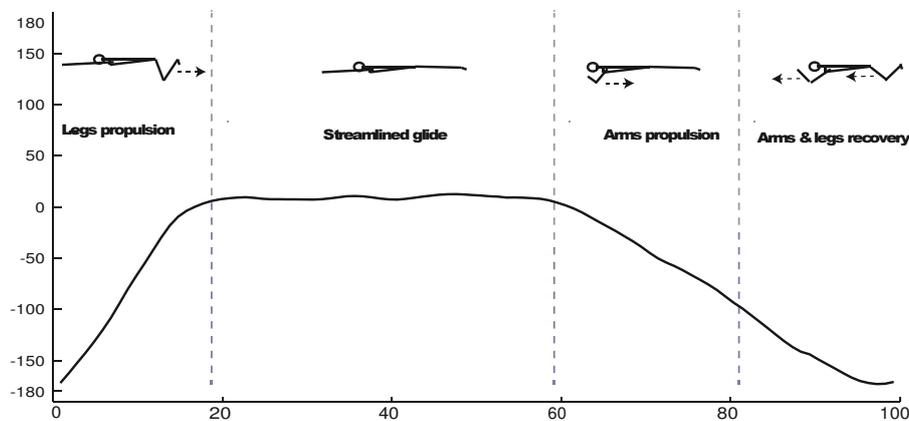


Figure 4.11: A prototype of continuous relative phase (in degree) between the knee and the elbow during one breaststroke cycle

Previous studies on breaststroke swimmers mainly focused on a static analysis, not defining the different behaviors exhibited during learning. As a matter of fact, a major interest in the field of motor learning resides in the definition of different pathways of learning, namely different possible learning strategies. Such an interest in investigating the existence of different "routes of learning" needs to focus on a dynamical analysis, namely the analysis of the successions of different behaviors. An unanswered question to date concerns the existence of optimal learning strategies (i.e. strategies that would appear more effective). Thus, the discovery of optimal learning strategies could have a huge impact on the pedagogical approach of practitioners.

Collected data

For this study, 26 novices were involved in 16 lessons of breaststroke swimming, with two sessions per week for a total duration of two months. The general goal of learning for all the 26 swimmers was to increase the distance per stroke, while maintaining the speed stable. Then the 26 learners were divided into four different groups, each group receiving a different instruction during the learning process:

Control group This group received only the general goal of learning, increase the distance per stroke

Analogy group In addition to the general goal of learning, this group received a single additional instruction: "glide two seconds with your arms outstretched"

Pacer group In addition to the general goal of learning, this group had to follow an auditory metronome trying to perform one cycle every single auditory signal.

Prescription group In addition to the general goal of learning, this group received multiple additional instructions: "keep your arms outstretched forward when you extend your legs; then glide with your

arms and legs outstretched; then keep your legs outstretched when you flex your arms; recover both arms and legs together”.

These different instructions were supposed to have a specific impact on the learning strategies of the learners.

In total, we have recorded 4160 trials (26 swimmers \times 16 sessions \times 10 trials) with an average of 8 cycles per trials. Thus, the dataset is composed by 33280 cycles, each cycle is represented by 100 continuous relative phase samples.

Study expectations

From a sport sciences point of view, the specific aims of the study were twofold:

- Assessing the dynamics of learning: In other words, the aim was to assess not only the different behaviors used during learning but also the transitions between these behaviors, that is the potential search strategy exhibited by learners.
- Assessing the impact of different learning conditions on the dynamics of learning: In other words, the aim was to investigate the possible existence of different behaviors exhibited by the learners regarding their learning condition, as well as the possible existence of different search strategy exhibited by different instructions given by the coach during the learning process.

A third side target of this study was then to define highly discriminative key points within the swimming cycle and that might be the target of the instruction in order to orient the attention of learners.

From a machine learning point of view, there are two locks to tackle:

- Each cycle is described by 100 features which are highly correlated due to the fact that they are samples of the relative phase which is a continuous time signal. Nevertheless, we don't want to bias the study by preprocessing the data, a transformation like filters, wavelet transform or sample selection that will embed a priori knowledge.
- The number of cycles are not equal on all the trials, that is why a trial cannot be directly described by a fixed number of features.

Fisher-EM Clustering

The proposed framework to solve the aforementioned ML locks is

1. a clustering by Fisher-EM [BB12] that also performs dimension reduction and features selection,
2. a two stage clustering: on cycles then on trials; a procedure similar to *Bags of words* to have fixed size features on trial.

A clustering can be derived from a mixture of Gaussians generative models. A Gaussian, which is parameterized by a covariance matrix and a mean in the observation space, represents a cluster. An observation is labeled according to its ownership (likelihood ratio) to each Gaussian. Knowing the number of clusters, the mixture and Gaussian parameters are learned from the observation data through an Expectation-Maximization (EM) algorithm.

The Fisher-EM algorithm [BB12] is based on the same principles but the mixture of Gaussians does not lie directly on the observation space but on a lower dimension latent space. This latent space is chosen to maximize the Fisher criterion between clusters and thus be discriminative and its dimension is bounded by the number of clusters. This reduction of dimension leads to more efficient computation on medium to large datasets (here 33280 examples by 100 features) as operations can be held in the smaller latent space.

Let us consider n observations y_1, y_2, \dots, y_n that are realizations of a random vector $Y \in \mathbb{R}^p$. We want to cluster these observations into K groups. For each observation y_i , a variable $z_i \in Z = \{1, \dots, K\}$ indicates which cluster it belongs to. This clustering will be decided upon a generative model, namely a mixture of K Gaussians which lies in a discriminative latent space $X \in \mathbb{R}^d$ where $d \leq K - 1$.

This latent space is linked to the observation space through a linear transformation,

$$Y = UX + \epsilon, \quad (4.1)$$

where $U \in \mathbb{R}^{p \times d}$ and $U^t U = Id(d)$ where $Id(d)$ is the identity matrix of size d , i.e. U is an orthogonal matrix and ϵ non-discriminative noise.

Let be $W = [U, V] \in \mathbb{R}^{p \times p}$ such that $W^t W = Id(p)$. V is the orthogonal complement of U . Thus, a projection $U^t y$ of an observation y from space Y of dimension p belongs to the latent discriminative

subspace X of dimension d and the projection $V^t y_i$ lies on the non-discriminative complement subspace of dimension $p - d$.

Conditionally to $Z = k$, random variables X and Y are assumed to be Gaussian, $X_{|Z=k} \sim \mathcal{N}(\mu_k, \Sigma_k)$, and $Y_{|Z=k} \sim \mathcal{N}(m_k, S_k)$, where $\mu_k \in \mathbb{R}^d$, $\Sigma_k \in \mathbb{R}^{d \times d}$, $m_k \in \mathbb{R}^p$ and $S_k \in \mathbb{R}^{p \times p}$.

Yet, the use of latent space introduces dimension reduction and computation efficiency. Nevertheless the back-projection from the latent space to the observation space can involve all the original features. To do feature selection, the projection matrix U has to be sparse. [BB14] proposed 3 methods to enforce sparsity based on sparse approximation, L_1 regularization or SVD penalization.

Please refer to [KHS14] (Appendix A.7) or the original publications of the Fisher-EM algorithm [BB12; BB14] to have a detailed description of the algorithm, the optimization procedure and its sparsification.

Results and interpretation

In the work published in [KHS14], for the first clustering level, analysis of the BIC highlights the existence of 11 clusters within the whole set of data. This result advocates for qualitative reorganizations of motor behavior during motor learning, as each learner visited between 9 and 11 different clusters during their sessions. Figure 4.12 shows successive stabilizations of different patterns of coordination (from 2 and 10 to 3 and 7) including feed-back to the initial pattern (10) of a participant of the control group.

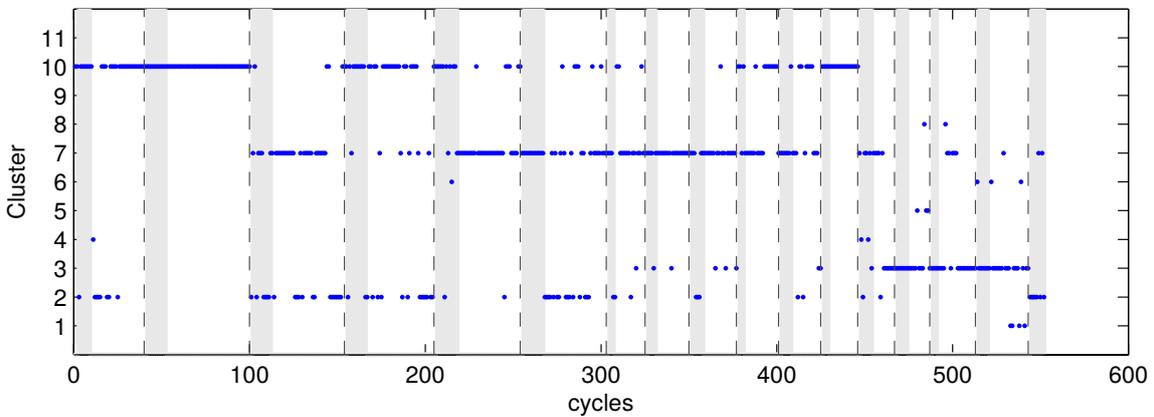


Figure 4.12: Patterns of coordination exhibited by the participant 4 of the control group (in dashed line, separations between the sessions) [Kom13]

More interesting, using the Sparse version of Fisher-EM we are able to discover what are the key points in the swimming cycle. On Figure 4.13, we have superimposed a typical coordination curve and, in gray bars, the back-projection of latent space into observation space to see induced sparsity from the first level. The height of a bar at a feature $i \in [1 \dots p]$ is proportional to $\sum_{j=1}^d |U_{ij}|$. A null value shows that the corresponding feature is not involved in the projection to the latent space, i.e. it is not selected by the F-Step or it is squeezed by the sparsity; therefore it can be considered not relevant to build the clusters. Compelling, only key points of the movement have high values, thus the Fisher-EM algorithm is able to select key points without any prior knowledge !

The second level of cluster analysis, based on the transition matrix during each trial, showed the existence of six different clusters. Interestingly, the group who showed the highest number of preferred transition was associated with the learning group that did not receive any instruction (i.e. the **control group**). In that sense, this second level of cluster analysis allowed to highlight the use of temporary additional information during learning in order to modify the learning search strategy, namely by impacting the preferred transitions.

After the work published in [KHS14], John KOMAR has undertaken for his PhD thesis [Kom13] an extensive qualitative and quantitative analysis on clustering results in par with the swimming condition and learning instructions given to the breakstroke swimmers. He linked the exploratory ability to the number of clusters visited and stabilized by each performer. He noticed that the prescription and analogy groups showed the earliest improvement in performance and exploratory ability but not necessarily the

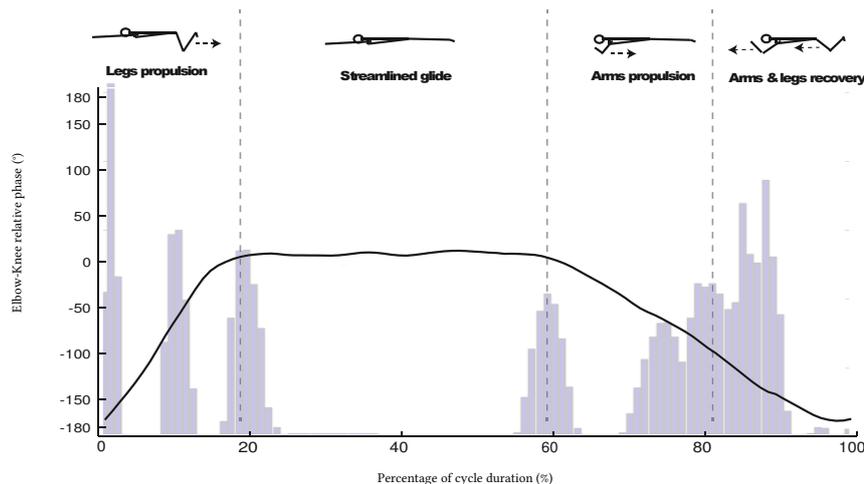


Figure 4.13: A typical coordination and superimposed induced sparsity

best in regard to the control group. Whereas the most constrained group, the pacer group, took more time to exhibit the same performance and hardly reached the same exploration rate. He concluded that

Thus, it seems that the increase in performance (i.e. the decrease in frequency in our case) might be associated to the learner's exploratory ability, and that the decrease in exploration leads to a stagnation of the performance.

Hence, this study had participated to answering on how do evolve coordination modes during training at each skills levels and how instructions given during training can reduce or increase the acquired variability.

4.3 Perspectives on Machine Learning applied to Human Movement

My collaboration with CETAPS keeps running. It is a challenging opportunity as HMS questions are still mainly opened and we had to clearly formalize their problem into machine learning frameworks. One of the frustrations is that we can rarely cast their needs to supervised learning tasks but most of the time into unsupervised ones. Therefore, the outcomes of the methods are evaluated in a more qualitative rather than a quantitative manner.

Currently and in the near future, we are focusing on performance prediction for competition (e.g. rank) and not only on individual performance (cf. NePTUNE project in Chapter 1 section 1.3). Consequently, we are more prone to cast their need into supervised regression problem where ML methods can be evaluated quantitatively.

4.4 Gait recognition

The way or manner a person walks constitutes a set of patterns of movements called the *gait*. *Gait recognition* consists in discriminating among people using these patterns.

This section presents the work realized on this discriminative task by Imad RIDA, a PhD Student under the supervision of Gilles GASSO and myself. Even if it is not part of the collaboration with CETAPS, it is still an application of Machine Learning to Human Movement. It eventually led to publication [Rid+17] and a significant part of his PhD work [Rid17].

4.4.1 Context

Taking the gait as a biometric trait can show advantageous properties over other biometric identification techniques. One may cite that:

- No need to have a contact with the analyzed person. The gait can be caught from distance by a camera,
- No need to get high-resolution pictures, the information of the gait mainly resides in movement that a low-definition video surveillance camera can capture,
- Gait is an unconscious behavior. Trying to masquerade it will lead to suspicious conduct.

Nevertheless, state of the art *gait recognition* methods suffer from clothing changes, carrying conditions, and modification of the point of view of the camera.

4.4.2 Proposed framework

The method is composed of three stages:

1. In a first stage, we segment the body into horizontal parts using a Group Lasso,
2. In the second stage, a Canonical Discriminant Analysis (CDA) [HHN99] selects which segments of the body part are the most useful for the discrimination,
3. In the last third stage, ultimately test samples are labeled using a nearest-neighbor classifier.

Our contribution mainly resides in using a Group Lasso to select the features that are robust to these condition changes between the sample in learning set that identified a person and the recorded sample to be discriminated.

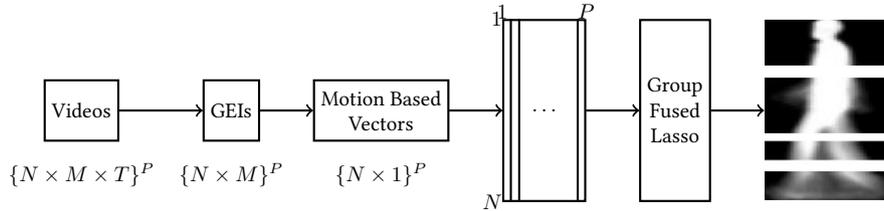


Figure 4.14: Processing flow of body segmentation into parts based on group fused Lasso of motion

Let's take a look into more details of the first stage of the proposed framework (Fig. 4.14). We consider that the training set is composed of P videos, each composed of T pictures of dimension $N \times M$. These videos are summed-up into Gait Energy Image (GEI), $G \in \mathbb{R}^{N \times M}$, which is a spatio-temporal representation of the gait obtained by averaging the silhouettes over a gait cycle [JB06]. The GEI are further reduced by averaging them line by line into a Motion Based Vector $\mathbf{e} \in \mathbb{R}^N$. Afterwards, all \mathbf{e} vectors are concatenated into a matrix $\mathbf{E} \in \mathbb{R}^{N \times P}$.

The segmentation of the body into horizontal parts is then performed by a multi-point change detection on E . It can be achieved by resolving the following convex optimization problem [VB10]:

$$\min_{\mathbf{V} \in \mathbb{R}^{N \times P}} \|\mathbf{E} - \mathbf{V}\|_F^2 + \lambda \sum_{i=1}^{N-1} \|\mathbf{v}_{i+1, \cdot} - \mathbf{v}_{i, \cdot}\|_1 \quad (4.2)$$

where $\mathbf{V} \in \mathbb{R}^{N \times P}$ is an approximation of \mathbf{E} and where $\mathbf{v}_{i, \cdot}$ is the i -th row of \mathbf{V} and $\lambda > 0$ a regularization parameter. Intuitively, increasing λ enforces many increments $\mathbf{v}_{i+1} - \mathbf{v}_i$ to converge towards zero. This implies that the position of non-zeros increments will be the same for all vectors \mathbf{e}_k . Therefore, the

solution of (4.2) provides an approximation of \mathbf{E} by a matrix \mathbf{V} of piecewise-constant vectors with shared change-points (Fig. 4.15).

The problem (4.2) is reformulated as a group Lasso regression problem as follows:

$$\min_{\beta \in \mathbb{R}^{(N-1) \times P}} \|\bar{\mathbf{E}} - \bar{\mathbf{X}}\beta\|_F^2 + \lambda \sum_{i=1}^{N-1} \|\beta_{i,\cdot}\|_1 \quad (4.3)$$

where $\bar{\mathbf{X}}$ and $\bar{\mathbf{E}}$ are obtained by centering each column from \mathbf{X} and \mathbf{E} knowing that:

$$\begin{cases} \mathbf{X} \in \mathbb{R}^{N \times (N-1)}; & x_{i,j} = \begin{cases} 1 & \text{for } i > j \\ 0 & \text{otherwise} \end{cases} \\ \beta_{i,\cdot} = \mathbf{v}_{i+1,\cdot} - \mathbf{v}_{i,\cdot} \end{cases} \quad (4.4)$$

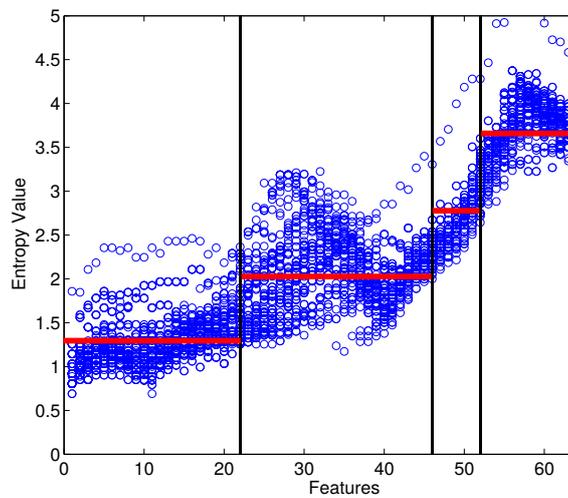


Figure 4.15: Example of shared change points across motion based vectors. Blue dots correspond to the motion-based vectors \mathbf{E} and red lines stand for the piecewise approximation \mathbf{V}

4.4.3 Results and perspectives

Intensive experiments [Rid+17; Rid17] have shown that the proposed method not only significantly outperforms other approaches in the case of clothing variations but also achieves the overall best performance among all approaches on the whole testing dataset that contains normal, carrying, clothing and view angle variations.

Nevertheless, the classification step uses a very simple technique (nearest-neighbor) and the Euclidian distance could be replaced by more adequate similarity measures. Actually, GEI can be seen as distributions of the movement patterns. Moreover the changing conditions (normal, clothing, carrying) affect heavily the statistics. In that context, as an interesting perspective we plan to lift our body part-selection approach in domain adaptation techniques. Particularly, we have intended to explore novel method such as optimal transport for domain adaptation based on a manifold regularization inspiring from the work in [Cou+17].

Chapter 5

Perspectives and scientific project

5.1 Challenges of Machine Learning

Over the last decade, the framework of Machine Learning has been a key element in addressing the problems of multiple scientific communities. It has taken precedence over most ad-hoc methods, particularly in signal processing.

For example, today, in almost all work in the fields of computer vision, imaging or sound processing there is a component related to statistical or deep learning, recently including the field of synthesis thanks to the contributions of generative models by adverse learning [Goo+14].

Artificial Intelligence Limits

Natural Language Processing (NLP) was one of the last areas to make the switch. Indeed, it is only recently that methods have been developed that can create a language model from a weakly labeled corpus [Dev+19]. These learned models now show their superiority over symbolic formal models : [Dev+19; Le+20; Mar+20].

The case of NLP is typical of the challenges of Machine Learning and beyond Artificial Intelligence in general.

On the one hand, we can legitimately wonder about the fact that if a formal model needs more than 8 000 rules (without counting the vocabulary) to explain a language it is because the modeling is not optimal. Indeed, it is quite unlikely that the neurological mechanisms involved in the production and interpretation of the language are based on such a large number of constraints; a statistical part must therefore intervene in our biological functioning (personal interpretation). This is why deep models have surpassed the state of the art in language processing recently.

On the other hand, to achieve good generalization performance, deep models need to be trained on hundreds of thousands or even millions of examples. However, a human being or an animal does not need to be confronted with such a number of cases to be able to learn and infer.

The number of rules needed for good modeling for symbolic methods is counterbalanced by the number of examples needed for statistical methods: resulting in a dead end on both sides, beyond finding a middle term between formal and automatic approaches to build a better modeling of the phenomena that surround us. This is why Artificial Intelligence methods are far from having reached a *common sense*, what is called strong AI [Cun19].

Limits of Supervised Learning and Deep Learning

Thus, the availability of data remains a major challenge for supervised learning methods and especially deep learning methods notably because of their number of parameters. Methods aimed at reducing the number of parameters in these models (convolutional layer, connection jump, . . .) do not meet all applications, especially when not only a small amount of data is accessible but also when annotations are rare and expensive (typically in medical imaging [Bel+17] or in NLP).

Before the advent of deep learning, much of a *data scientist's* expertise came from his/her knowledge in feature extraction: what were the preparatory steps to be applied to the raw data in order to extract the information needed by the classifiers. Today this work is done by the neural network itself. However, deep learning has exacerbated the other major task facing the expert: which model to choose, how to

set the hyper-parameters? Whereas in a support vector machine (SVM), the choice is limited to the type of kernel, its hyper-parameters and its regularization; in a neural network, each neuron in itself has an amount of variations equivalent to a single SVM and the number of possible network arrangements is infinite. This difficulty makes engineering deep neural networks particularly difficult for neophytes.

A more general problem of machine learning methods but which is also amplified by the complexity of the architecture of deep neural networks is the interpretability of models. It is possible to indicate *how* a network gives an answer but it is difficult to indicate *why*. This limits its use in fields requiring explainable (medicine) or proven results (critical system, nuclear power, aviation, . . .). Research in this field is still in its infancy, as stated above (see [MJ18; ZNZ18]).

On Ethical, Social and Environmental Issues

As I am not a specialist in social issues, I do not wish to develop here the ethical aspects raised by artificial intelligence or machine learning, and even less do I wish to take a position beyond my scientific competence. However, as with all applications of statistics, I think it is important to distinguish between what is descriptive and what is normative. I don't think it's ethical for a decision made by an artificial intelligence to be applied directly to an individual. Human intervention seems to me indispensable and must be helped in this by the interpretability of the models learned (cf. previous section).

One aspect that is more easily accessible to me is the environmental aspect. Supervised or deep learning methods have two kinds of needs that have a strong ecological impact:

- large amount of data . . . so it requires computer storage and physical space;
- large computing capacity due to the large number of parameters . . . thus it needs energy.

Here, the ecological challenges meet the scientific locks cited in the previous section. Indeed, having learning methods that require fewer examples would reduce the need for storage. In addition, constrained models or interpretable models require fewer parameters and therefore less calculation and energy.

Companies also have an economic need to reduce the energy cost of learning and using such models. Hardware solutions are emerging. The first have been GPU, which, with equivalent matrix computing capacity, are more energy efficient than CPU. FPGA and dedicated chips further reduce requirements. Examples include TPU on Google's servers or the Apple Neural Engine on smartphones, where reducing the energy footprint is essential.

5.2 Scientific Perspectives

I have isolated two recent advances in machine learning that I believe can partially address the above challenges: *self-supervised learning* and *optimal transport*. We have recently been able to use them in my team and our first publications on these topics are accepted [Kec+20] or submitted [Bli+].

Self-Supervised Learning

In order to compensate for the lack of annotated data, most current approaches use a semi-supervised framework where unlabeled data participate in the learning of non-supervised ancillary tasks that support the learning of the main supervised task. Nevertheless, monitoring the training of an unsupervised task is difficult because of the non-availability of an explicit evaluation criterion. These methods are then more easily exposed to over-fitting. (The work presented in this habilitation has attempted to address partially this problem by gradually giving more importance to the supervised task during learning).

The self-supervised learning [DZ17] proposes to keep a scheme of main target task and auxiliary secondary tasks but this time staying in a supervised framework for all tasks! How to do then when one wants to take advantage of unlabeled data? The solution is to create fake supervised tasks from the unlabeled data. Here are some examples of self-supervised tasks [WKZ18; Ser+18; KZB19; JT20]:

1. Arbitrarily flip natural images and ask which ones are still in the right place (classification),
2. Rotate natural images and ask which angle they were rotated from (regression),
3. Extract a sample of an image around a position (x, y) and ask to find the coordinates (x, y) by giving the image and the sample,
4. Convert an image to gray scale and ask to retrieve the color version,
5. Reverse time arbitrarily on videos and ask which ones are always in the right time direction,
6. Predict intermediate frames of a video and then from the intermediate frames predict the originals.

The tasks described in points 4 and 6 are very close to what is done in unsupervised domain mapping (CycleGAN) : we study the coherence of a transformation/back-transformation cycle between a domain

A (color image) and a domain B (grayscale image). One of my research perspectives is to work on the resonances between domain mapping in adverse learning (GAN) and self-supervised learning.

Optimal Transport

Optimal transport or transport theory is a very ancient mathematical and economic field that was originally formulated in the 18th century by Gaspard MONGE for the optimal transfer of materials and the optimal allocation of resources in the context of the nascent industrial revolution. However, it was only belatedly that non-trivial solutions could be formulated. KANTOROVITCH was awarded the 1971 Nobel Prize in Economics for his work on the subject. This field continues to be an important field of mathematical innovation with two Fields Medal winners in the last 10 years (VILLANI in 2010 and FIGALI in 2018).

Optimal transport provides us with tools to compare probability measures of all kinds. This is a critical point for generative models (GAN), structured or large dimensional data because of the curse of dimensionality. Indeed, in large dimensions all distances crush and standard tools for comparing measurements are no longer robust. It is therefore a promising framework for machine learning [Cou+17; Suv+].

In addition, the *transport plan* resulting from the optimization of the problem is a representation that is easily humanly interpretable and can provide clues as to *why* a model works.

We had a first approach to optimal transport during our work to address the problem of non-stationarity between the learning set and the test set [Kec+20]. I wish to continue to develop the use of these tools, especially in very degenerate situations where few examples are available but lie in large spaces. This is typically the case in the context of sequence processing in medical imaging.

5.3 Personal Project

Collaboration on Machine Learning and Health

I am currently the coordinator of the DeepART (Deep learning for Adaptive RadioTherapy) axis of the MINMACS regional excellence project. This involves applying deep learning techniques to help medical doctor adapt a radiotherapy plan during treatment.

This project and more generally the interface between Health / Imaging / Machine Learning bring into play the above-mentioned limitations of Artificial Intelligence: few data, structured/large data, and the need for explanations of the model. This is why it is important to me to continue to make progress on this topic within LITIS through collaborations between the *App*, *QuantIF* teams and the Henri BECQUEREL center. On a regional and international scale, I would like to develop our collaborations with the University of Caen, the François BACLESSE center and the UCLouvain in Belgium within the framework of the North-Western cancer center.

In the short term, I will devote part of my research funding to the exchange of PhD students between LITIS and UCLouvain as I was able to benefit from it through sabbaticals. Beyond the sporadic exchanges, if I obtain the HDR, I will be able to set up doctoral co-supervision projects between Pr. John LEE of UCLouvain and myself.

In addition, I would like to be able to create and lead a collaboration between my colleagues specialized in medical imaging and my colleagues specialized in high performance sports. This is a medium-term project that will have to rely on ANR or European funding, given the number of people and institutions involved.

Research and Development at LITIS

I wish to be able to fully direct my theoretical research towards the two challenges presented in this chapter, namely *self supervised learning* and *optimal transport*. I'm not alone to target these directions in the *App* team. In the medium term, we may be able to form a spin-off team, especially if the current team continues to grow in size and missions in the face of the Artificial Intelligence craze.

As far as funding for research work is concerned, I envisage two distinct approaches:

1. I prefer to concentrate my efforts in seeking funding for PhD students on establishment grants and industrial collaborations (CIFRE-type scheme). This will ensure the continuation of the research themes whatever the hazards of project calls. To this end, obtaining the HDR will make it easier for me to choose industrial collaborations as a priority in relation to the research areas I wish to develop.

2. On the other hand, I will reserve responses to project calls for the financing exchanges, post-docs and research engineers through inter-thematic or inter-institutional requests.

Beyond the LITIS

In order to consider applying for a university professorship, I still need to develop my skills in team management and project management. LITIS is a very favorable ground and the head of the lab is committed to the training of its members. To this end, I have no doubt that I will be able to take on responsibilities in the laboratory once I have obtained my habilitation.

Bibliography

- [Aba+15] Martín Abadi et al. “TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems.” In: (2015). URL: <https://www.tensorflow.org/>.
- [ABH17] Fabio Aioli, Gaëlle Bonnet-Loosli, and Romain Héroult. “Advances in Artificial Neural Networks, Machine Learning and Computational Intelligence, Special Issue ESANN 2017 (Editorial).” In: *Neurocomputing*. Advances in Artificial Neural Networks, Machine Learning and Computational Intelligence 268 (Dec. 13, 2017), pp. 1–3. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2017.04.038. URL: <http://www.sciencedirect.com/science/article/pii/S0925231217307634>.
- [Aio+16] Fabio Aioli, Kerstin Bunte, Romain Héroult, and Mikhail Kanevski. “Advances in Artificial Neural Networks, Machine Learning and Computational Intelligence, Special Issue ESANN 2015 (Editorial).” In: *Neurocomputing*. Advances in Artificial Neural Networks, Machine Learning and Computational Intelligence 192 (June 5, 2016), pp. 1–2. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2016.02.005. URL: <http://www.sciencedirect.com/science/article/pii/S0925231216001831>.
- [Amy+18] Amine Amyar, Su Ruan, Isabelle Gardin, Romain Héroult, Chatelain Clement, Pierre Decazes, and Romain Modzelewski. “Radiomics-Net: Convolutional Neural Networks on FDG PET Images for Predicting Cancer Treatment Response.” In: *Journal of Nuclear Medicine* 59 (supplement 1 2018), p. 324. URL: <https://hal-normandie-univ.archives-ouvertes.fr/hal-02129431>.
- [AC15] Jinwon An and Sungzoon Cho. “Variational Autoencoder Based Anomaly Detection Using Reconstruction Probability.” In: *Special Lecture on IE 2.1* (2015). URL: <http://dm.snu.ac.kr/static/docs/TR/SNUDM-TR-2015-03.pdf>.
- [Ani+20] Anna Aniszewska-Śtepien, Romain Héroult, Guillaume Hacques, Ludovic Seifert, and Gilles Gasso. “Learning from Partially Labeled Sequences for Behavioral Signal Annotation.” In: Accepted to Machine Learning and Data Mining for Sports Analytics (MLSA) 2020. Ghent, Belgium, Sept. 14, 2020.
- [Bar+02] Benoît G. Bardy, Olivier Oullier, Reinoud J. Bootsma, and Thomas A. Stoffregen. “Dynamics of Human Postural Transitions.” In: *Journal of Experimental Psychology: Human Perception and Performance* 28.3 (2002), pp. 499–514. ISSN: 1939-1277(Electronic),0096-1523(Print). DOI: 10.1037/0096-1523.28.3.499.
- [BWR07] Roger Bartlett, Jon Wheat, and Matthew Robins. “Is Movement Variability Important for Sports Biomechanists?” In: *Sports Biomechanics* 6.2 (May 1, 2007), pp. 224–243. ISSN: 1476-3141. DOI: 10.1080/14763140701322994. PMID: 17892098. URL: <https://doi.org/10.1080/14763140701322994>.
- [BN93] Michèle Basseville and Igor Nikiforov. *Detection of Abrupt Changes - Theory and Application*. Prentice Hall Englewood Cliffs. Vol. 104. 1993. URL: <https://hal.archives-ouvertes.fr/hal-00008518>.
- [BR17] Centre Henri Becquerel and INSA de Rouen. “BodyComp.AI : L’utilisation de l’intelligence Artificielle En Imagerie Médicale, Prix Unicancer de l’innovation 2017, Prix de l’organisation et Des Métiers de La Recherche.” 2017.
- [BM16] David Belanger and Andrew McCallum. “Structured Prediction Energy Networks.” In: *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48* (New York, NY, USA). ICML’16. JMLR.org, 2016, pp. 983–992. URL: <http://dl.acm.org/citation.cfm?id=3045390.3045495>.

- [Bel+15a] Soufiane Belharbi, Clement Chatelain, Romain Herault, and Sebastien Adam. “A Unified Neural Based Model for Structured Output Problems.” In: Conférence Sur l’Apprentissage Automatique (CAp). Lille, France, 2015, p. 10. URL: <https://sbelharbi.github.io/publications/2015/belharbiCAP2015.pdf>.
- [Bel+15b] Soufiane Belharbi, Clement Chatelain, Romain Hérault, and Sébastien Adam. “Learning Structured Output Dependencies Using Deep Neural Networks.” In: Deep Learning Workshop, ICML. 2015. URL: https://www.researchgate.net/profile/Clement_Chatelain/publication/293097934_Learning_Structured_Output_Dependencies_Using_Deep_Neural_Networks/links/56f5a15b08ae7c1fda2eea19/Learning-Structured-Output-Dependencies-Using-Deep-Neural-Networks.pdf.
- [Bel+17] Soufiane Belharbi, Clément Chatelain, Romain Hérault, Sébastien Adam, Sébastien Thureau, Mathieu Chastan, and Romain Modzelewski. “Spotting L3 Slice in CT Scans Using Deep Convolutional Network and Transfer Learning.” In: *Computers in Biology and Medicine* 87 (Aug. 1, 2017), pp. 95–103. ISSN: 0010-4825. DOI: 10.1016/j.combiomed.2017.05.018. URL: <http://www.sciencedirect.com/science/article/pii/S0010482517301403>.
- [Bel+16a] Soufiane Belharbi, Romain Herault, Clement Chatelain, and Sebastien Adam. “Deep Multi-Task Learning with Evolving Weights.” In: European Symposium on Artificial Neural Networks (ESANN). Bruges, Belgium, 2016, p. 6. URL: <https://sbelharbi.github.io/publications/2016/presentation-ESANN2016-bleharbi.pdf>.
- [Bel+16b] Soufiane Belharbi, Romain Hérault, Clément Chatelain, and Sébastien Adam. “Pondération Dynamique Dans Un Cadre Multi-Tâche Pour Réseaux de Neurones Profonds.” In: *Session Spéciale” Apprentissage et Vision”*. Reconnaissance Des Formes et l’Intelligence Artificielle (RFIA). Clermont-Ferrand, France, 2016. URL: <https://sbelharbi.github.io/publications/2016/RFIA2016-belharbi.pdf>.
- [Bel+18] Soufiane Belharbi, Romain Hérault, Clément Chatelain, and Sébastien Adam. “Deep Neural Networks Regularization for Structured Output Prediction.” In: *Neurocomputing* 281 (Mar. 15, 2018), pp. 169–177. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2017.12.002. URL: <http://www.sciencedirect.com/science/article/pii/S0925231217318295>.
- [Bel+13] Peter N. Belhumeur, David W. Jacobs, David J. Kriegman, and Neeraj Kumar. “Localizing Parts of Faces Using a Consensus of Exemplars.” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.12 (Dec. 2013), pp. 2930–2940. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2013.23.
- [BB84] Albert Benveniste and Michèle Basseville. “Detection of Abrupt Changes in Signals and Dynamical Systems : Some Statistical Aspects.” In: *Analysis and Optimization of Systems*. Ed. by Alain Bensoussan and J. L. Lions. Lecture Notes in Control and Information Sciences. Berlin, Heidelberg: Springer, 1984, pp. 143–155. ISBN: 978-3-540-39007-7. DOI: 10.1007/BFb0004951.
- [Ber+10] James Bergstra, Olivier Breuleux, Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, Guillaume Desjardins, Joseph Turian, David Warde-Farley, and Yoshua Bengio. “Theano: A CPU and GPU Math Expression Compiler.” In: *Proceedings of the Python for Scientific Computing Conference (SciPy)*. Austin, TX, June 2010.
- [Ber66] Nikolai Bernstein. “The Co-Ordination and Regulation of Movements.” In: *The co-ordination and regulation of movements* (1966).
- [Bes86] Julian Besag. “On the Statistical Analysis of Dirty Pictures.” In: *Journal of the Royal Statistical Society: Series B (Methodological)* 48.3 (1986), pp. 259–279. ISSN: 2517-6161. DOI: 10.1111/j.2517-6161.1986.tb01412.x. URL: <https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.2517-6161.1986.tb01412.x>.
- [BHS13] Gautier Bideault, Romain Herault, and Ludovic Seifert. “Data Modelling Reveals Inter-Individual Variability of Front Crawl Swimming.” In: *Journal of Science and Medicine in Sport* 16.3 (May 1, 2013), pp. 281–285. ISSN: 1440-2440. DOI: 10.1016/j.jsams.2012.08.001. URL: <http://www.sciencedirect.com/science/article/pii/S1440244012001715>.
- [BL08] Matthew B. Blaschko and Christoph H. Lampert. “Learning to Localize Objects with Structured Output Regression.” In: *Computer Vision – ECCV 2008*. Ed. by David Forsyth, Philip Torr, and Andrew Zisserman. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2008, pp. 2–15. ISBN: 978-3-540-88682-2. DOI: 10.1007/978-3-540-88682-2_2.

- [Bli+] Rachel Blin, Cyprien Ruffino, Stéphane Canu, Gilles Gasso, Samia Ainouz, Fabrice Meriaudeau, and Romain Hérault. “Generating Polarimetric-Encoded Images Using Constrained Cycle-Consistent Generative Adversarial Networks.” In: Submitted to ACCV.
- [Bou+16] Jeremie Boulanger, Ludovic Seifert, Romain Hérault, and Jean-François Coeurjolly. “Automatic Sensor-Based Detection and Classification of Climbing Activities.” In: *IEEE Sensors Journal* 16.3 (Feb. 2016), pp. 742–749. doi: 10.1109/JSEN.2015.2481511. URL: <https://hal.archives-ouvertes.fr/hal-01225056>.
- [BB12] Charles Bouveyron and Camille Brunet. “Theoretical and Practical Considerations on the Convergence Properties of the Fisher-EM Algorithm.” In: *Journal of Multivariate Analysis* 109 (Aug. 1, 2012), pp. 29–41. issn: 0047-259X. doi: 10.1016/j.jmva.2012.02.012. URL: <http://www.sciencedirect.com/science/article/pii/S0047259X1200053X>.
- [BB14] Charles Bouveyron and Camille Brunet-Saumard. “Discriminative Variable Selection for Clustering with the Sparse Fisher-EM Algorithm.” In: *Computational Statistics* 29.3 (June 1, 2014), pp. 489–513. issn: 1613-9658. doi: 10.1007/s00180-013-0433-6. URL: <https://doi.org/10.1007/s00180-013-0433-6>.
- [BSB83] David Burton, J. Shore, and J. Buck. “A Generalization of Isolated Word Recognition Using Vector Quantization.” In: *ICASSP ’83. IEEE International Conference on Acoustics, Speech, and Signal Processing*. ICASSP ’83. IEEE International Conference on Acoustics, Speech, and Signal Processing. Vol. 8. Apr. 1983, pp. 1021–1024. doi: 10.1109/ICASSP.1983.1171915.
- [Car97] Rich Caruana. “Multitask Learning.” In: *Machine Learning* 28.1 (July 1, 1997), pp. 41–75. issn: 1573-0565. doi: 10.1023/A:1007379606734. URL: <https://doi.org/10.1023/A:1007379606734>.
- [Cho+14a] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. “Learning Phrase Representations Using RNN Encoder–Decoder for Statistical Machine Translation.” In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. EMNLP 2014. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1724–1734. doi: 10.3115/v1/D14-1179. URL: <https://www.aclweb.org/anthology/D14-1179>.
- [CB90] Paul B. Chou and Christopher M. Brown. “The Theory and Practice of Bayesian Image Labeling.” In: *International Journal of Computer Vision* 4.3 (June 1, 1990), pp. 185–210. issn: 1573-1405. doi: 10.1007/BF00054995. URL: <https://doi.org/10.1007/BF00054995>.
- [Cho+14b] Jia Yi Chow, Ludovic Seifert, Romain Hérault, Shannon Jing Yi Chia, and Miriam Chang Yi Lee. “A Dynamical System Perspective to Understanding Badminton Singles Game Play.” In: *Human Movement Science* 33 (Feb. 1, 2014), pp. 70–84. issn: 0167-9457. doi: 10.1016/j.humov.2013.07.016. URL: <http://www.sciencedirect.com/science/article/pii/S0167945713000985>.
- [Chu+09] Howard Chung, Dana Cobzas, Laura Birdsell, Jessica Lieffers, and Vickie Baracos. “Automated Segmentation of Muscle and Adipose Tissue on CT Images for Human Body Composition Analysis.” In: *Medical Imaging 2009: Visualization, Image-Guided Procedures, and Modeling*. Medical Imaging 2009: Visualization, Image-Guided Procedures, and Modeling. Vol. 7261. International Society for Optics and Photonics, Mar. 13, 2009, 72610K. doi: 10.1117/12.812412. URL: <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/7261/72610K/Automated-segmentation-of-muscle-and-adipose-tissue-on-CT-images/10.1117/12.812412.short>.
- [Com94] Pierre Comon. “Independent Component Analysis, A New Concept?” In: *Signal Processing. Higher Order Statistics* 36.3 (Apr. 1, 1994), pp. 287–314. issn: 0165-1684. doi: 10.1016/0165-1684(94)90029-9. URL: <http://www.sciencedirect.com/science/article/pii/0165168494900299>.
- [Cou+17] Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. “Optimal Transport for Domain Adaptation.” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.9 (Sept. 2017), pp. 1853–1865. issn: 0162-8828. doi: 10.1109/TPAMI.2016.2615921.
- [CP11] Gabriela Csurka and Florent Perronnin. “An Efficient Approach to Semantic Segmentation.” In: *International Journal of Computer Vision* 95.2 (Nov. 1, 2011), pp. 198–212. issn: 1573-1405. doi: 10.1007/s11263-010-0344-8. URL: <https://doi.org/10.1007/s11263-010-0344-8>.

- [Cun19] Yann Le Cun. *Quand la machine apprend: La révolution des neurones artificiels et de l'apprentissage profond*. Odile Jacob, Oct. 16, 2019. 388 pp. ISBN: 978-2-7381-4932-9. Google Books: 78m2DwAAQBAJ.
- [Cyb89] G. Cybenko. "Approximation by Superpositions of a Sigmoidal Function." In: *Mathematics of Control, Signals and Systems* 2.4 (Dec. 1, 1989), pp. 303–314. ISSN: 1435-568X. DOI: 10.1007/BF02551274. URL: <https://doi.org/10.1007/BF02551274>.
- [Dav+03] Keith Davids, Paul Glazier, Duarte Araújo, and Roger Bartlett. "Movement Systems as Dynamical Systems." In: *Sports Medicine* 33.4 (Apr. 1, 2003), pp. 245–260. ISSN: 1179-2035. DOI: 10.2165/00007256-200333040-00001. URL: <https://doi.org/10.2165/00007256-200333040-00001>.
- [Def77] D. Defays. "An Efficient Algorithm for a Complete Link Method." In: *The Computer Journal* 20.4 (Jan. 1, 1977), pp. 364–366. ISSN: 0010-4620. DOI: 10.1093/comjnl/20.4.364. URL: <https://academic.oup.com/comjnl/article/20/4/364/393966>.
- [DLR77] A. P. Dempster, N. M. Laird, and D. B. Rubin. "Maximum Likelihood from Incomplete Data Via the EM Algorithm." In: *Journal of the Royal Statistical Society: Series B (Methodological)* 39.1 (1977), pp. 1–22. ISSN: 2517-6161. DOI: 10.1111/j.2517-6161.1977.tb01600.x. URL: <https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.2517-6161.1977.tb01600.x>.
- [Dev+19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding." In: NAACL-HLT 2019. Minneapolis, Minnesota: Association for Computational Linguistics, May 24, 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. arXiv: 1810.04805 [cs]. URL: <http://arxiv.org/abs/1810.04805>.
- [DZ17] Carl Doersch and Andrew Zisserman. "Multi-Task Self-Supervised Visual Learning." In: Proceedings of the IEEE International Conference on Computer Vision. 2017, pp. 2051–2060. URL: https://openaccess.thecvf.com/content_iccv_2017/html/Doersch_Multi-Task_Self-Supervised_Visual_ICCV_2017_paper.html.
- [DKD16] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. "Adversarial Feature Learning." In: ICLR 2017. Toulon, France, May 31, 2016. arXiv: 1605.09782 [cs, stat]. URL: <http://arxiv.org/abs/1605.09782>.
- [Dov+14] Vladislavs Dovgalecs, Jérémie Boulanger, Dominique Orth, Romain Herault, Jean-François Coeurjolly, Keith Davids, and Ludovic Seifert. "Movement Phase Detection in Climbing." In: *Sports Technology*. Rock Climbing 7.3-4 (2014), pp. 174–182. DOI: 10.1080/19346182.2015.1064128. URL: <https://hal.archives-ouvertes.fr/hal-01071401>.
- [Dum+16] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, and Aaron Courville. "Adversarially Learned Inference." In: ICLR 2016. San Juan, Puerto Rico, June 2, 2016. arXiv: 1606.00704 [cs, stat]. URL: <http://arxiv.org/abs/1606.00704>.
- [EAH99] K. Engan, S.O. Aase, and J. Hakon Husoy. "Method of Optimal Directions for Frame Design." In: *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No.99CH36258)*. 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No.99CH36258). Vol. 5. Mar. 1999, 2443–2446 vol.5. ISBN: 978-0-7803-5041-0. DOI: 10.1109/ICASSP.1999.760624.
- [Est+96] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise." In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. AAAI Press, 1996, pp. 226–231.
- [Fer+08] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid. "Groups of Adjacent Contour Segments for Object Detection." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30.1 (Jan. 2008), pp. 36–51. ISSN: 0162-8828, 2160-9292, 1939-3539. DOI: 10.1109/TPAMI.2007.1144.
- [Fri93] Moshe Fridman. *Hidden Markov Model Regression*. 1993.
- [FH75] K. Fukunaga and L. Hostetler. "The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition." In: *IEEE Transactions on Information Theory* 21.1 (Jan. 1975), pp. 32–40. ISSN: 1557-9654. DOI: 10.1109/TIT.1975.1055330.

- [Gei+13] A Geiger, P Lenz, C Stiller, and R Urtasun. “Vision Meets Robotics: The KITTI Dataset.” In: *The International Journal of Robotics Research* 32.11 (Sept. 1, 2013), pp. 1231–1237. ISSN: 0278-3649. DOI: 10 . 1177 / 0278364913491297. URL: <https://doi.org/10.1177/0278364913491297>.
- [Gib14] James J. Gibson. *The Ecological Approach to Visual Perception : Classic Edition*. Psychology Press, Nov. 20, 2014. ISBN: 978-1-315-74021-8. DOI: 10.4324/9781315740218. URL: <https://www.taylorfrancis.com/books/9781315740218>.
- [GBB11] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. “Deep Sparse Rectifier Neural Networks.” In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics. June 14, 2011, pp. 315–323. URL: <http://proceedings.mlr.press/v15/glorot11a.html>.
- [Goo+14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. “Generative Adversarial Nets.” In: *Advances in Neural Information Processing Systems 27* (2014). In collab. with Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, pp. 2672–2680. URL: <http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf>.
- [Gra+06] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. “Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks.” In: *Proceedings of the 23rd International Conference on Machine Learning*. ICML ’06. Pittsburgh, Pennsylvania, USA: Association for Computing Machinery, June 25, 2006, pp. 369–376. ISBN: 978-1-59593-383-6. DOI: 10 . 1145 / 1143844 . 1143891. URL: <https://doi.org/10.1145/1143844.1143891>.
- [GS05] Alex Graves and Jürgen Schmidhuber. “Framewise Phoneme Classification with Bidirectional LSTM and Other Neural Network Architectures.” In: *Neural Networks*. IJCNN 2005 18.5 (July 1, 2005), pp. 602–610. ISSN: 0893-6080. DOI: 10.1016/j.neunet.2005.06.042. URL: <http://www.sciencedirect.com/science/article/pii/S0893608005001206>.
- [Gro87] Stephen Grossberg. “Competitive Learning: From Interactive Activation to Adaptive Resonance.” In: *Cognitive Science* 11.1 (Jan. 1, 1987), pp. 23–63. ISSN: 0364-0213. DOI: 10.1016/S0364-0213(87)80025-3. URL: <http://www.sciencedirect.com/science/article/pii/S0364021387800253>.
- [HSS03] Richard H. R. Hahnloser, H. Sebastian Seung, and Jean-Jacques Slotine. “Permitted and Forbidden Sets in Symmetric Threshold-Linear Networks.” In: *Neural Computation* 15.3 (Mar. 2003), pp. 621–638. ISSN: 0899-7667, 1530-888X. DOI: 10.1162/089976603321192103. URL: <http://www.mitpressjournals.org/doi/10.1162/089976603321192103>.
- [HHM00] Joseph Hamill, Jeffrey M. Haddad, and William J. McDermott. “Issues in Quantifying Variability from a Dynamical Systems Perspective.” In: *Journal of Applied Biomechanics* 16.4 (Nov. 1, 2000), pp. 407–418. ISSN: 1065-8483. DOI: 10.1123/jab.16.4.407. URL: <https://journals.humankinetics.com/doi/abs/10.1123/jab.16.4.407>.
- [He+16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep Residual Learning for Image Recognition.” In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 770–778. URL: http://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html.
- [HA84] Jeanny Hérault and Bernard Ans. “Réseau de Neurones à Synapses Modifiables: Décodage de Messages Sensoriels Composites Par Apprentissage Non Supervisé et Permanent.” In: *Comptes rendus des séances de l’Académie des sciences. Série 3, Sciences de la vie* 299.13 (1984), pp. 525–528.
- [Hér07] Romain Hérault. “Vision et Apprentissage Statistique Pour La Reconnaissance d’items Comportementaux.” thesis. Compiègne, Nov. 26, 2007. URL: <http://www.theses.fr/2007COMP1715>.
- [Hér17] Romain Hérault. “Deep Learning.” Research Summer School on Statistics & BigData Science - SBDS (Université de Caen). June 8, 2017.
- [Hér19a] Romain Hérault. “Deep Generative Model.” Séminaire École Doctorale CIL (Université libre de Liège). June 4, 2019.

- [Hér19b] Romain Hérault. “Deep Learning.” Séminaire École Doctorale CIL (Université catholique de Louvain). May 20, 2019.
- [Hér20] Romain Hérault. “Deep Generative Model.” Séminaire IMVIA, ESIREM (Université de Bourgogne). Jan. 23, 2020.
- [Hér+15] Romain Hérault, Jeremie Boulanger, Ludovic Seifert, and John Aldo Lee. “Valuation of Climbing Activities Using Multi-Scale Jensen-Shannon Neighbour Embedding.” In: *Machine Learning and Data Mining for Sports Analytics, ECML/PKDD 2015 workshop (MLSA2015)*, 2015. URL: <https://hal.archives-ouvertes.fr/hal-01441636>.
- [Hér+06] Romain Hérault, Franck Davoine, Fadi Dornaika, and Yves Grandvalet. “Suivis Simultanés et Robustes de Visages et de Gestes Faciaux.” In: *Reconnaissance Des Formes et l’Intelligence Artificielle (RFIA)* (Tours, France, France). Tour, France, Jan. 2006. URL: <https://hal.archives-ouvertes.fr/hal-00442758>.
- [HDG06] Romain Hérault, Franck Davoine, and Yves Grandvalet. “Head and Facial Action Tracking: Comparison of Two Robust Approaches.” In: *7th IEEE International Conference on Automatic Face and Gesture Recognition* (Southampton, UK, United Kingdom). IEEE Computer Society, Apr. 2006, pp. 287–292. URL: <https://hal.archives-ouvertes.fr/hal-00442753>.
- [HG07a] Romain Hérault and Yves Grandvalet. “Régression Logistique Parcimonieuse.” In: *Conférence Sur l’Apprentissage Automatique (CAp)* (Grenoble, France, France). Ed. by Cépaduès. Grenoble, France, July 2007, pp. 265–280. URL: <https://hal.archives-ouvertes.fr/hal-00442755>.
- [HG07b] Romain Hérault and Yves Grandvalet. “Sparse Probabilistic Classifiers.” In: *ICML ’07: Proceedings of the 24th International Conference on Machine Learning* (New York, NY, USA, United States). ACM, June 2007, pp. 337–344. DOI: 10.1145/1273496.1273539. URL: <https://hal.archives-ouvertes.fr/hal-00442746>.
- [HG08] Romain Hérault and Yves Grandvalet. “Classifieurs Probabilistes Parcimonieux.” In: *Traitement du Signal* 25.4 (2008), pp. 279–291. URL: <https://hal.archives-ouvertes.fr/hal-00442731>.
- [Hér+17] Romain Hérault, Dominic Orth, Ludovic Seifert, Jérémie Boulanger, and John Aldo Lee. “Comparing Dynamics of Fluency and Inter-Limb Coordination in Climbing Activities Using Multi-Scale Jensen–Shannon Embedding and Clustering.” In: *Data Mining and Knowledge Discovery* 31.6 (Nov. 2017), pp. 1758–1792. DOI: 10.1007/s10618-017-0522-1. URL: <https://hal.archives-ouvertes.fr/hal-02094958>.
- [HS06] G. E. Hinton and R. R. Salakhutdinov. “Reducing the Dimensionality of Data with Neural Networks.” In: *Science* 313.5786 (July 28, 2006), pp. 504–507. ISSN: 0036-8075, 1095-9203. DOI: 10.1126/science.1127647. pmid: 16873662. URL: <https://science.sciencemag.org/content/313/5786/504>.
- [Hin+12] Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R. Salakhutdinov. *Improving Neural Networks by Preventing Co-Adaptation of Feature Detectors*. Version 1. July 3, 2012. arXiv: 1207.0580 [cs]. URL: <http://arxiv.org/abs/1207.0580>.
- [HS97] Sepp Hochreiter and Jürgen Schmidhuber. “Long Short-Term Memory.” In: *Neural Computation* 9.8 (Nov. 1, 1997), pp. 1735–1780. ISSN: 0899-7667. DOI: 10.1162/neco.1997.9.8.1735. URL: <https://doi.org/10.1162/neco.1997.9.8.1735>.
- [Hua+17] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. “Densely Connected Convolutional Networks.” In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 4700–4708. URL: http://openaccess.thecvf.com/content_cvpr_2017/html/Huang_Densely_Connected_Convolutional_CVPR_2017_paper.html.
- [HHN99] P. S. Huang, C. J. Harris, and M. S. Nixon. “Recognising Humans by Gait via Parametric Canonical Space.” In: *Artificial Intelligence in Engineering* 13.4 (Oct. 1, 1999), pp. 359–366. ISSN: 0954-1810. DOI: 10.1016/S0954-1810(99)00008-4. URL: <http://www.sciencedirect.com/science/article/pii/S0954181099000084>.
- [IS15] Sergey Ioffe and Christian Szegedy. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. Mar. 2, 2015. arXiv: 1502.03167 [cs]. URL: <http://arxiv.org/abs/1502.03167>.

- [JBV17] Nikolay Jetchev, Urs Bergmann, and Roland Vollgraf. *Texture Synthesis with Spatial Generative Adversarial Networks*. Sept. 8, 2017. arXiv: 1611.08207 [cs, stat]. URL: <http://arxiv.org/abs/1611.08207>.
- [JT20] Longlong Jing and Yingli Tian. “Self-Supervised Visual Feature Learning with Deep Neural Networks: A Survey.” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020), pp. 1–1. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2020.2992393.
- [JB06] Ju Han and Bir Bhanu. “Individual Recognition Using Gait Energy Image.” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28.2 (Feb. 2006), pp. 316–322. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2006.38.
- [Kar+20] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. “Analyzing and Improving the Image Quality of StyleGAN.” In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Mar. 23, 2020, pp. 8110–8119. arXiv: 1912.04958 [cs, eess, stat]. URL: <http://arxiv.org/abs/1912.04958>.
- [Kec+20] Marwa Kechaou, Romain Herault, Mokhtar Z. Alaya, and Gilles Gasso. “Open Set Domain Adaptation Using Optimal Transport.” In: *Accepted in European Conference on Machine Learning, and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*. Ghent, Belgium, Sept. 14, 2020.
- [Kei+13] Abou Keita, Romain Héroult, Colas Calbrix, and Stéphane Canu. “Detection and Quantification in Real-Time Polymerase Chain Reaction.” In: *European Symposium on Artificial Neural Networks (ESANN)*. Bruges, Belgium, Apr. 2013, p. 351. URL: <https://hal.archives-ouvertes.fr/hal-00834417>.
- [KHC12] Abou Keita, Romain Héroult, and Stéphane Canu. “Estimation de La Concentration d’un Agent Biologique Par Détection de Rupture Sur Vidéos de Fluorescences Issues de PCR.” In: *Reconnaissance Des Formes et l’Intelligence Artificielle (RFIA)* (Lyon, France). Lyon, France, Jan. 2012, pp. 978-2-9539515-2-3. URL: <https://hal.archives-ouvertes.fr/hal-00656568>.
- [Kim+17] Taeksoo Kim, Moon-su Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. “Learning to Discover Cross-Domain Relations with Generative Adversarial Networks.” In: *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*. Vol. 70. *Proceedings of Machine Learning Research*. PMLR, Mar. 15, 2017, pp. 1857–1865. arXiv: 1703.05192. URL: <http://arxiv.org/abs/1703.05192>.
- [Kin80] Ross Kindermann. *Markov Random Fields and Their Applications*. American mathematical society. 1980. ISBN: 978-0-8218-5001-5.
- [KLT09] Pushmeet Kohli, L’ubor Ladický, and Philip H. S. Torr. “Robust Higher Order Potentials for Enforcing Label Consistency.” In: *International Journal of Computer Vision* 82.3 (May 1, 2009), pp. 302–324. ISSN: 0920-5691, 1573-1405. DOI: 10.1007/s11263-008-0202-0. URL: <https://link.springer.com/article/10.1007/s11263-008-0202-0>.
- [Koh82] Teuvo Kohonen. “Self-Organized Formation of Topologically Correct Feature Maps.” In: *Biological Cybernetics* 43.1 (Jan. 1, 1982), pp. 59–69. ISSN: 1432-0770. DOI: 10.1007/BF00337288. URL: <https://doi.org/10.1007/BF00337288>.
- [KZB19] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. “Revisiting Self-Supervised Visual Representation Learning.” In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 1920–1929. URL: https://openaccess.thecvf.com/content_CVPR_2019/html/Kolesnikov_Revisiting_Self-Supervised_Visual_Representation_Learning_CVPR_2019_paper.html.
- [Kom13] John Komar. “Dynamique de l’apprentissage Moteur : Apprendre Loin de l’équilibre.” thesis. Rouen, Jan. 1, 2013. URL: <http://www.theses.fr/2013ROUEL009>.
- [KHS14] John Komar, Romain Héroult, and Ludovic Seifert. “Key Point Selection and Clustering of Swimmer Coordination through Sparse Fisher-EM.” In: *ECML/PKDD 2013 Workshop on Machine Learning and Data Mining for Sports Analytics (MLSA2013)*. Jan. 7, 2014. arXiv: 1401.1489 [physics, stat]. URL: <http://arxiv.org/abs/1401.1489>.
- [Kra91] Mark A. Kramer. “Nonlinear Principal Component Analysis Using Autoassociative Neural Networks.” In: *AIChE Journal* 37.2 (1991), pp. 233–243. ISSN: 1547-5905. DOI: 10.1002/aic.690370209. URL: <https://aiche.onlinelibrary.wiley.com/doi/abs/10.1002/aic.690370209>.

- [KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “ImageNet Classification with Deep Convolutional Neural Networks.” In: *Advances in Neural Information Processing Systems 25*. Ed. by F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger. Curran Associates, Inc., 2012, pp. 1097–1105. URL: <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>.
- [LHC09] Benjamin Labbé, Romain Héroult, and Clément Chatelain. “Learning Deep Neural Networks for High Dimensional Output Problems.” In: *ICMLA (United States)*. Dec. 2009, 6p. URL: <https://hal.archives-ouvertes.fr/hal-00438714>.
- [LMP01] John Lafferty, Andrew McCallum, and Fernando Pereira. “Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data.” In: *Departmental Papers (CIS)* (June 28, 2001). URL: https://repository.upenn.edu/cis_papers/159.
- [Lal+18] Eric Laloy, Romain Héroult, Diederik Jacques, and Niklas Linde. “Training-Image Based Geostatistical Inversion Using a Spatial Generative Adversarial Neural Network.” In: *Water Resources Research* 54.1 (Jan. 1, 2018), pp. 381–406. ISSN: 1944-7973. DOI: 10.1002/2017WR022148. URL: <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2017WR022148>.
- [Lal+17] Eric Laloy, Romain Héroult, John Lee, Diederik Jacques, and Niklas Linde. “Inversion Using a New Low-Dimensional Representation of Complex Binary Geological Media Based on a Deep Neural Network.” In: *Advances in Water Resources* 110 (Dec. 2017), pp. 387–405. DOI: 10.1016/j.advwatres.2017.09.029. URL: <https://hal.archives-ouvertes.fr/hal-02094960>.
- [Lal+19] Eric Laloy, Niklas Linde, Cyprien Ruffino, Romain Héroult, Gilles Gasso, and Diederik Jacques. “Gradient-Based Deterministic Inversion of Geophysical Data with Generative Adversarial Networks: Is It Feasible?” In: *Computers & Geosciences* (Sept. 24, 2019), p. 104333. ISSN: 0098-3004. DOI: 10.1016/j.cageo.2019.104333. URL: <http://www.sciencedirect.com/science/article/pii/S009830041831207X>.
- [Le+20] Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoît Crabbé, Laurent Besacier, and Didier Schwab. “FlauBERT: Unsupervised Language Model Pre-Training for French.” In: *Proceedings of the 12th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, Mar. 12, 2020, pp. 2479–2490. arXiv: 1912.05372. URL: <http://arxiv.org/abs/1912.05372>.
- [Le+12] Vuong Le, Jonathan Brandt, Zhe Lin, Lubomir Bourdev, and Thomas S. Huang. “Interactive Facial Feature Localization.” In: *Computer Vision – ECCV 2012*. Ed. by Andrew Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2012, pp. 679–692. ISBN: 978-3-642-33712-3. DOI: 10.1007/978-3-642-33712-3_49.
- [LPV15] John A. Lee, Diego H. Peluffo-Ordóñez, and Michel Verleysen. “Multi-Scale Similarities in Stochastic Neighbour Embedding: Reducing Dimensionality While Preserving Both Local and Global Structure.” In: *Neurocomputing. Learning for Visual Semantic Understanding in Big Data* 169 (Dec. 2, 2015), pp. 246–261. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2014.12.095. URL: <http://www.sciencedirect.com/science/article/pii/S0925231215003641>.
- [Ler+15] Julien Lerouge, Romain Héroult, Clément Chatelain, Fabrice Jardin, and Romain Modzelewski. “IODA: An Input/Output Deep Architecture for Image Labeling.” In: *Pattern Recognition* 48.9 (Sept. 1, 2015), pp. 2847–2858. ISSN: 0031-3203. DOI: 10.1016/j.patcog.2015.03.017. URL: <http://www.sciencedirect.com/science/article/pii/S0031320315001181>.
- [LSD15] Jonathan Long, Evan Shelhamer, and Trevor Darrell. “Fully Convolutional Networks for Semantic Segmentation.” In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. June 2015, pp. 3431–3440. DOI: 10.1109/CVPR.2015.7298965. URL: https://www.cv-foundation.org/openaccess/content_cvpr_2015/html/Long_Fully_Convolutional_Networks_2015_CVPR_paper.html.
- [LKC16] William Lotter, Gabriel Kreiman, and David Cox. *Unsupervised Learning of Visual Structure Using Predictive Generative Networks*. Jan. 20, 2016. arXiv: 1511.06380 [cs, q-bio]. URL: <http://arxiv.org/abs/1511.06380>.

- [Luo+16] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. “Understanding the Effective Receptive Field in Deep Convolutional Neural Networks.” In: *Advances in Neural Information Processing Systems 29*. Curran Associates, Inc., 2016, pp. 4898–4906. arXiv: 1701.04128. URL: <http://arxiv.org/abs/1701.04128>.
- [Mac+67] James MacQueen et al. “Some Methods for Classification and Analysis of Multivariate Observations.” In: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*. Vol. 1. 14. Oakland, CA, USA. 1967, pp. 281–297.
- [Mak+15] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. *Adversarial Autoencoders*. Nov. 17, 2015. arXiv: 1511.05644 [cs]. URL: <http://arxiv.org/abs/1511.05644>.
- [Mar+20] Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villemonte de la Clergerie, Djamel Seddah, and Benoît Sagot. “CamemBERT: A Tasty French Language Model.” In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, May 21, 2020, pp. 7203–7219. arXiv: 1911.03894. URL: <http://arxiv.org/abs/1911.03894>.
- [Mar17] MartinThoma. *Deutsch: Maske Zur Semantischen Segmentierung von File:EmiMa-099.jpg*. June 8, 2017. URL: <https://commons.wikimedia.org/wiki/File:EmiMa-099-semantic-segmentation.png>.
- [MJ18] David Alvarez Melis and Tommi Jaakkola. “Towards Robust Interpretability with Self-Explaining Neural Networks.” In: *Advances in Neural Information Processing Systems 31*. Ed. by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett. Curran Associates, Inc., 2018, pp. 7775–7784. URL: <http://papers.nips.cc/paper/8003-towards-robust-interpretability-with-self-explaining-neural-networks.pdf>.
- [MP69] Marvin Minsky and Seymour Papert. *Perceptrons: An Introduction to Computational Geometry*. Cambridge Trass., HIT. 1969.
- [MO14] Mehdi Mirza and Simon Osindero. *Conditional Generative Adversarial Nets*. Nov. 6, 2014. arXiv: 1411.1784 [cs, stat]. URL: <http://arxiv.org/abs/1411.1784>.
- [MLH11] Volodymyr Mnih, Hugo Larochelle, and Geoffrey E. Hinton. “Conditional Restricted Boltzmann Machines for Structured Output Prediction.” In: *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*. Barcelona, Spain, July 14, 2011, pp. 514–522. arXiv: 1202.3748 [cs, stat]. URL: <http://arxiv.org/abs/1202.3748>.
- [NV01] Karl M. Newell and David E. Vaillancourt. “Dimensional Change in Motor Learning.” In: *Human Movement Science* 20.4 (Nov. 1, 2001), pp. 695–715. ISSN: 0167-9457. DOI: 10.1016/S0167-9457(01)00073-2. URL: <http://www.sciencedirect.com/science/article/pii/S0167945701000732>.
- [Nic+07] S. Nicolas, J. Dardenne, T. Paquet, and L. Heutte. “Document Image Segmentation Using a 2D Conditional Random Field Model.” In: *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*. Ninth International Conference on Document Analysis and Recognition (ICDAR 2007). Vol. 1. Sept. 2007, pp. 407–411. DOI: 10.1109/ICDAR.2007.4378741.
- [NVK93] I. Nikiforov, V. Varavva, and V. Kireichikov. “Application of Statistical Fault Detection Algorithms to Navigation Systems Monitoring.” In: *Automatica* 29.5 (Sept. 1, 1993), pp. 1275–1290. ISSN: 0005-1098. DOI: 10.1016/0005-1098(93)90050-4. URL: <http://www.sciencedirect.com/science/article/pii/0005109893900504>.
- [NC08] Keith Noto and Mark Craven. “Learning Hidden Markov Models for Regression Using Path Aggregation.” In: *Uncertainty in artificial intelligence : proceedings of the ... conference. Conference on Uncertainty in Artificial Intelligence 2008* (July 9, 2008), pp. 444–451. ISSN: 1525-3384. PMID: 21785575. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3141580/>.
- [Ort+13] Dominic Orth, Keith Davids, Romain Héroult, and Ludovic Seifert. “Indices of Behavioral Complexity over Repeated Trials in a Climbing Task: Evaluating Mechanisms Underpinning Emergence of Skilled Performance,” in: *European Conferences on Complex Systems (ECCS)*. Barcelona, 2013.
- [Pas+17] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. “Automatic Differentiation in PyTorch.” In: (Oct. 28, 2017). URL: <https://openreview.net/forum?id=BJJsrmfCZ>.

- [Pea01] Karl Pearson. “On Lines and Planes of Closest Fit to Systems of Points in Space.” In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2.11 (1901), pp. 559–572.
- [RJ86] L. Rabiner and B. Juang. “An Introduction to Hidden Markov Models.” In: *IEEE ASSP Magazine* 3.1 (Jan. 1986), pp. 4–16. ISSN: 1558-1284. DOI: 10.1109/MASSP.1986.1165342.
- [Rät+00] Gunnar Rätsch, Bernhard Schölkopf, Sebastian Mika, and Klaus-Robert Müller. *SVM and Boosting: One Class*. GMD-Forschungszentrum Informationstechnik, 2000.
- [Rid17] Imad Rida. “Temporal Signals Classification.” thesis. Normandie, Feb. 3, 2017. URL: <http://www.theses.fr/2017NORMIR01>.
- [Rid+17] Imad Rida, Noor Al Maadeed, Gian Luca Marcialis, Ahmed Bouridane, Romain Héault, and Gilles Gasso. “Improved Model-Free Gait Recognition Based on Human Body Part.” In: *Biometric Security and Privacy: Opportunities & Challenges in The Big Data Era*. Ed. by Richard Jiang, Somaya Al-maadeed, Ahmed Bouridane, Prof. Danny Crookes, and Azeddine Beghdadi. Signal Processing for Security Technologies. Cham: Springer International Publishing, 2017, pp. 141–161. ISBN: 978-3-319-47301-7. DOI: 10.1007/978-3-319-47301-7_6. URL: https://doi.org/10.1007/978-3-319-47301-7_6.
- [Rid+18] Imad Rida, Romain Héault, Gian Luca Marcialis, and Gilles Gasso. “Palmprint Recognition with an Efficient Data Driven Ensemble Classifier.” In: *Pattern Recognition Letters* (Apr. 22, 2018). ISSN: 0167-8655. DOI: 10.1016/j.patrec.2018.04.033. URL: <http://www.sciencedirect.com/science/article/pii/S0167865518301612>.
- [RHG14] Imad Rida, Romain Héault, and Gilles Gasso. “Supervised Music Chord Recognition.” In: 2014 13th International Conference on Machine Learning and Applications (ICMLA). IEEE, Dec. 3, 2014, pp. 336–341. DOI: 10.1109/ICMLA.2014.60. URL: <https://hal-normandie-univ.archives-ouvertes.fr/hal-02096549>.
- [RFB15] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation.” In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Ed. by Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2015, pp. 234–241. ISBN: 978-3-319-24574-4. DOI: 10.1007/978-3-319-24574-4_28.
- [Ruf+18] Cyprien Ruffino, Romain Héault, Eric Laloy, and Gilles Gasso. “Dilated Spatial Generative Adversarial Networks for Ergodic Image Generation.” In: *Conférence Sur l’Apprentissage Automatique (CAp)*. Rouen, France, June 20, 2018. arXiv: 1905.08613 [cs, eess]. URL: <http://arxiv.org/abs/1905.08613>.
- [Ruf+19a] Cyprien Ruffino, Romain Héault, Eric Laloy, and Gilles Gasso. “Pixel-Wise Conditioning of Generative Adversarial Networks.” In: *European Symposium on Artificial Neural Networks (ESANN)*. Bruges, Belgium, Apr. 24, 2019. URL: <https://hal.archives-ouvertes.fr/hal-02347732>.
- [Ruf+20] Cyprien Ruffino, Romain Héault, Eric Laloy, and Gilles Gasso. “Pixel-Wise Conditioned Generative Adversarial Networks for Image Synthesis and Completion.” In: *Neurocomputing* (Apr. 2020). DOI: 10.1016/j.neucom.2019.11.116. arXiv: 2002.01281. URL: <https://hal.archives-ouvertes.fr/hal-02551730>.
- [Ruf+19b] Cyprien Ruffino, Romain Héault, Éric Laloy, and Gilles Gasso. “Approche GAN Pour La Génération d’images Sous Contraintes de Pixel.” In: *Conférence Sur l’Apprentissage Automatique (CAp)*. Toulouse, France, 2019, pp. 439–444.
- [RN16] Stuart J. Russell and Peter Norvig. *Artificial Intelligence : A Modern Approach*. Malaysia; Pearson Education Limited, 2016. URL: http://thuvienso.thanglong.edu.vn/handle/DHTL_123456789/4010.
- [Sei+13a] Ludovic Seifert, Jean-François Coeurjolly, Romain Héault, Leo Wattebled, and Keith Davids. “Temporal Dynamics of Inter-Limb Coordination in Ice Climbing Revealed through Change-Point Analysis of the Geodesic Mean of Circular Data.” In: *Journal of Applied Statistics* 40.11 (Nov. 2013), pp. 2317–2331. DOI: 10.1080/02664763.2013.810194. URL: <https://hal.archives-ouvertes.fr/hal-02094911>.

- [Sei+15] Ludovic Seifert, Vladislavs Dovgalecs, Jérémie Boulanger, Dominic Orth, Romain Hérault, and Keith Davids. “Full-Body Movement Pattern Recognition in Climbing.” In: *Sports Technology* 7.3-4 (July 2015), pp. 166–173. doi: 10.1080/19346182.2014.968250. URL: <https://hal.archives-ouvertes.fr/hal-02094936>.
- [Sei+14a] Ludovic Seifert, Maxime L’Hermette, John Komar, Dominic Orth, Florian Mell, Pierre Merriaux, Pierre Grenet, Yanis Caritu, Romain Hérault, Vladislavs Dovgalecs, and Keith Davids. “Pattern Recognition in Cyclic and Discrete Skills Performance from Inertial Measurement Units.” In: *Procedia Engineering* 72 (2014), pp. 196–201. doi: 10.1016/j.proeng.2014.06.033. URL: <https://hal-normandie-univ.archives-ouvertes.fr/hal-02096541>.
- [Sei+10a] Ludovic Seifert, Hugues Leblanc, Didier Chollet, and Didier Delignières. “Inter-Limb Coordination in Swimming: Effect of Speed and Skill Level.” In: *Human Movement Science* 29.1 (Feb. 1, 2010), pp. 103–113. ISSN: 0167-9457. doi: 10.1016/j.humov.2009.05.003. URL: <http://www.sciencedirect.com/science/article/pii/S0167945709000694>.
- [Sei+11a] Ludovic Seifert, Hugues Leblanc, Romain Hérault, John Komar, Chris Button, and Didier Chollet. “Inter-Individual Variability in the Upper-Lower Limb Breaststroke Coordination.” In: *Human Movement Science* 30.3 (June 1, 2011), pp. 550–565. ISSN: 0167-9457. doi: 10.1016/j.humov.2010.12.003. URL: <http://www.sciencedirect.com/science/article/pii/S016794571000182X>.
- [Sei+14b] Ludovic Seifert, Dominic Orth, Jérémie Boulanger, Vladislavs Dovgalecs, Romain Hérault, and Keith Davids. “Climbing Skill and Complexity of Climbing Wall Design: Assessment of Jerk as a Novel Indicator of Performance Fluency.” In: *Journal of Applied Biomechanics* 30.5 (Oct. 2014), pp. 619–625. doi: 10.1123/jab.2014-0052. URL: <https://hal.archives-ouvertes.fr/hal-02094928>.
- [Sei+13b] Ludovic Seifert, Dominic Orth, Romain Hérault, and Keith Davids. “Metastability in Perception and Action in Rock Climbing.” In: *XVIIIth International Conference on Perception and Action*. Estoril: FMH Editions, Portugal. Estoril, Portugal, 2013.
- [Sei+18] Ludovic Seifert, Dominic Orth, Bruno Mantel, Jérémie Boulanger, Romain Hérault, and Matt Dicks. “Affordance Realization in Climbing: Learning and Transfer.” In: *Frontiers in Psychology* 9 (May 2018). doi: 10.3389/fpsyg.2018.00820. URL: <https://hal.archives-ouvertes.fr/hal-02094976>.
- [Sei+10b] Ludovic Seifert, Leo Wattebled, Maxime L’Hermette, and Romain Hérault. “Effect of Skill Level on Upper/Lower Limb Coordination in Ice Climbers.” In: *11th European Workshop of Ecological Psychology* (France). June 2010, pp. 68–69. URL: <https://hal.archives-ouvertes.fr/hal-00558158>.
- [Sei+10c] Ludovic Seifert, Leo Wattebled, Maxime L’Hermette, and Romain Hérault. “Inter-Limb Coordination Variability in Ice Climbers of Different Skill Level.” In: *3rd International Congress Complex Systems in Medicine and Sport* (Lithuania). Sept. 2010, pp. 105–106. URL: <https://hal.archives-ouvertes.fr/hal-00558152>.
- [Sei+11b] Ludovic Seifert, Leo Wattebled, Maxime L’Hermette, and Romain Hérault. “Inter-Limb Coordination Variability in Ice Climbers of Different Skill Level.” In: *Baltic Journal of Sport and Health Sciences* 1.80 (2011). URL: <https://journals.lsu.lt/baltic-journal-of-sport-health/article/download/342/338>.
- [Sei+14c] Ludovic Seifert, Léo Wattebled, Romain Hérault, Germain Poizat, David Adé, Nathalie Gal-Petitfaux, and Keith Davids. “Neurobiological Degeneracy and Affordance Perception Support Functional Intra-Individual Variability of Inter-Limb Coordination during Ice Climbing.” In: *PLOS ONE* 9.2 (Feb. 24, 2014), e89865. ISSN: 1932-6203. doi: 10.1371/journal.pone.0089865. URL: <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0089865>.
- [Sei+13c] Ludovic Seifert, Léo Wattebled, Maxime L’Hermette, Gautier Bideault, Romain Hérault, and Keith Davids. “Skill Transfer, Affordances and Dexterity in Different Climbing Environments.” In: *Human Movement Science* 32.6 (Dec. 1, 2013), pp. 1339–1352. ISSN: 0167-9457. doi: 10.1016/j.humov.2013.06.006. URL: <http://www.sciencedirect.com/science/article/pii/S0167945713000766>.

- [Ser+18] Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, Sergey Levine, and Google Brain. “Time-Contrastive Networks: Self-Supervised Learning from Video.” In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. 2018 IEEE International Conference on Robotics and Automation (ICRA). May 2018, pp. 1134–1141. DOI: 10.1109/ICRA.2018.8462891.
- [Sib73] R. Sibson. “SLINK: An Optimally Efficient Algorithm for the Single-Link Cluster Method.” In: *The Computer Journal* 16.1 (Jan. 1, 1973), pp. 30–34. ISSN: 0010-4620. DOI: 10.1093/comjnl/16.1.30. URL: <https://academic.oup.com/comjnl/article/16/1/30/434805>.
- [Sri+14] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. “Dropout: A Simple Way to Prevent Neural Networks from Overfitting.” In: *The Journal of Machine Learning Research* 15.1 (Jan. 1, 2014), pp. 1929–1958. ISSN: 1532-4435.
- [Ste56] Hugo Steinhaus. “Sur La Division Des Corp Materiels En Parties.” In: *Bull. Acad. Polon. Sci* 1.804 (1956), p. 801.
- [Suv+] Alexandra Suvorikova, Marco Cuturi, Gabriel Peyré, and Rémi Flamary. *OTML Workshop NeurIPS 2019*. URL: <https://sites.google.com/view/otml2019/home>.
- [TD07] Caroline Teulier and Didier Delignières. “The Nature of the Transition between Novice and Skilled Coordination during Learning to Swing.” In: *Human Movement Science* 26.3 (June 1, 2007), pp. 376–392. ISSN: 0167-9457. DOI: 10.1016/j.humov.2007.01.013. URL: <http://www.sciencedirect.com/science/article/pii/S0167945707000152>.
- [The16] Theano Development Team. “Theano: A Python Framework for Fast Computation of Mathematical Expressions.” In: *arXiv e-prints* abs/1605.02688 (May 2016). URL: <http://arxiv.org/abs/1605.02688>.
- [Tho03] Emanuel Jöbstl und Martin Thoma. *Deutsch: Stofflöwe, Der Eine Rote Rose Hält*. 30 November 2016, 07:20:03. URL: <https://commons.wikimedia.org/wiki/File:EmiMa-099.jpg>.
- [Tia+10] X Tian, Romain Héroult, Gilles Gasso, and Stephane Canu. “Pré-Apprentissage Supervisé Pour Les Réseaux Profonds.” In: *Reconnaissance Des Formes et l’Intelligence Artificielle (RFIA)*. Caen, France, 2010.
- [TGC12] Xilan Tian, Gilles Gasso, and Stéphane Canu. “A Multiple Kernel Framework for Inductive Semi-Supervised SVM Learning.” In: *Neurocomputing. Advances in Artificial Neural Networks, Machine Learning, and Computational Intelligence (ESANN 2011)* 90 (Aug. 1, 2012), pp. 46–58. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2011.12.036. URL: <http://www.sciencedirect.com/science/article/pii/S0925231212001877>.
- [Tso+04] Ioannis Tsochantaridis, Thomas Hofmann, Thorsten Joachims, and Yasemin Altun. “Support Vector Machine Learning for Interdependent and Structured Output Spaces.” In: *Proceedings of the Twenty-First International Conference on Machine Learning. ICML ’04*. Banff, Alberta, Canada: Association for Computing Machinery, July 4, 2004, p. 104. ISBN: 978-1-58113-838-2. DOI: 10.1145/1015330.1015341. URL: <https://doi.org/10.1145/1015330.1015341>.
- [vEvW00] Richard E.A. van Emmerik and Erwin E.H. van Wegen. “On Variability and Stability in Human Movement.” In: *Journal of Applied Biomechanics* 16.4 (Nov. 2000), pp. 394–406. ISSN: 1065-8483, 1543-2688. DOI: 10.1123/jab.16.4.394. URL: <http://journals.humankinetics.com/doi/10.1123/jab.16.4.394>.
- [VB10] Jean-philippe Vert and Kevin Bleakley. “Fast Detection of Multiple Change-Points Shared by Many Signals Using Group LARS.” In: *Advances in Neural Information Processing Systems 23*. Ed. by J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta. Curran Associates, Inc., 2010, pp. 2343–2351. URL: <http://papers.nips.cc/paper/4157-fast-detection-of-multiple-change-points-shared-by-many-signals-using-group-lars.pdf>.
- [Wan+13] Li Wan, Matthew Zeiler, Sixin Zhang, Yann Le Cun, and Rob Fergus. “Regularization of Neural Networks Using DropConnect.” In: *International Conference on Machine Learning*. International Conference on Machine Learning. Feb. 13, 2013, pp. 1058–1066. URL: <http://proceedings.mlr.press/v28/wan13.html>.
- [Wes+02] Jason Weston, Olivier Chapelle, André Elisseeff, Bernhard Schölkopf, and Vladimir Vapnik. “Kernel Dependency Estimation.” In: *Proceedings of the 15th International Conference on Neural Information Processing Systems. NIPS’02*. Cambridge, MA, USA: MIT Press, Jan. 1, 2002, pp. 897–904.

- [WKZ18] Olivia Wiles, A. Sophia Koepke, and Andrew Zisserman. “Self-Supervised Learning of a Facial Attribute Embedding from Video.” In: British Machine Vision Conference (BMVC). Aug. 21, 2018. arXiv: 1808.06882 [cs]. URL: <http://arxiv.org/abs/1808.06882>.
- [WH18] Yuxin Wu and Kaiming He. “Group Normalization.” In: Proceedings of the European Conference on Computer Vision (ECCV). 2018, pp. 3–19. URL: http://openaccess.thecvf.com/content_ECCV_2018/html/Yuxin_Wu_Group_Normalization_ECCV_2018_paper.html.
- [XXC12] Junyuan Xie, Linli Xu, and Enhong Chen. “Image Denoising and Inpainting with Deep Neural Networks.” In: *Advances in Neural Information Processing Systems 25*. Ed. by F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger. Curran Associates, Inc., 2012, pp. 341–349. URL: <http://papers.nips.cc/paper/4686-image-denoising-and-inpainting-with-deep-neural-networks.pdf>.
- [Yi+17] Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. “DualGAN: Unsupervised Dual Learning for Image-to-Image Translation.” In: *2017 IEEE International Conference on Computer Vision (ICCV)*. 2017 IEEE International Conference on Computer Vision (ICCV). Venice: IEEE, Oct. 2017, pp. 2868–2876. ISBN: 978-1-5386-1032-9. DOI: 10.1109/ICCV.2017.310. URL: <http://ieeexplore.ieee.org/document/8237572/>.
- [YK15] Fisher Yu and Vladlen Koltun. “Multi-Scale Context Aggregation by Dilated Convolutions.” In: ICLR 2016. San Juan, Puerto Rico, Nov. 23, 2015. arXiv: 1511.07122 [cs]. URL: <http://arxiv.org/abs/1511.07122>.
- [Zha+18] Jiyi Zhang, Hung Dang, Hwee Kuan Lee, and Ee-Chien Chang. *Flipped-Adversarial AutoEncoders*. Feb. 13, 2018. arXiv: 1802.04504 [cs]. URL: <http://arxiv.org/abs/1802.04504>.
- [ZNZ18] Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. “Interpretable Convolutional Neural Networks.” In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018, pp. 8827–8836. URL: https://openaccess.thecvf.com/content_cvpr_2018/html/Zhang_Interpretable_Convolutional_Neural_CVPR_2018_paper.html.
- [ZSQ17] Zhifei Zhang, Yang Song, and Hairong Qi. “Age Progression/Regression by Conditional Adversarial Autoencoder.” In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017, pp. 5810–5818. URL: http://openaccess.thecvf.com/content_cvpr_2017/html/Zhang_Age_ProgressionRegression_by_CVPR_2017_paper.html.
- [Zhu+17] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. “Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks.” In: *2017 IEEE International Conference on Computer Vision (ICCV)*. 2017 IEEE International Conference on Computer Vision (ICCV). Venice: IEEE, Oct. 2017, pp. 2242–2251. ISBN: 978-1-5386-1032-9. DOI: 10.1109/ICCV.2017.244. URL: <http://ieeexplore.ieee.org/document/8237506/>.

Appendix A

Selected Publications

A.1 IODA: an Input/Output Deep Architecture for image labeling

Reference

[Ler+15] Julien Lerouge et al. “IODA: An Input/Output Deep Architecture for Image Labeling.” In: *Pattern Recognition* 48.9 (Sept. 1, 2015), pp. 2847–2858. ISSN: 0031-3203. DOI: 10.1016/j.patcog.2015.03.017. URL: <http://www.sciencedirect.com/science/article/pii/S0031320315001181>

IODA: an Input/Output Deep Architecture for image labeling

J. Lerouge^{a,*}, R. Herault^a, C. Chatelain^{a,*}, F. Jardin^c, R. Modzelewski^{b,a}

^aLITIS EA 4108, INSA de Rouen, Saint Étienne du Rouvray 76800, France

^bDepartment of Nuclear Medicine, Henri Becquerel Cancer Center

^cDepartment of Hematology & U918 (INSERM), Henri Becquerel Cancer Center

Abstract

In this article, we propose a deep neural network (DNN) architecture called Input Output Deep Architecture (IODA) for solving the problem of image labeling. IODA directly links a whole image to a whole label map, assigning a label to each pixel using a single neural network forward step. Instead of designing a handcrafted a priori model on labels (such as an atlas in the medical domain), we propose to automatically learn the dependencies between labels. The originality of IODA is to transpose DNN input pre-training trick to the output space, in order to learn a high level representation of labels. It allows a fast image labeling inside a fully neural network framework, without the need of any preprocessing such as feature designing or output coding.

In this article, IODA is applied on both a toy texture problem and a real-world medical image dataset, showing promising results. We provide an open source implementation of IODA¹².

Keywords: Deep learning architectures, deep neural network, image labeling, machine learning, medical imaging, sarcopenia

1. Introduction

When dealing with a huge amount of images, the classical computer vision problems can be either i) assigning a class to an image, known as the image classification problem; ii) partitioning an image into non-overlapping regions, known as the image segmentation problem; or iii) assigning a class or a label to each pixel of an image, known as the image labeling problem (sometimes called semantic segmentation). This last problem have received a lot of attention

*Corresponding author

Email addresses: `julien.lerouge@insa-rouen.fr` (J. Lerouge),
`romain.herault@insa-rouen.fr` (R. Herault), `clement.chatelain@insa-rouen.fr` (C. Chatelain), `romain.modzelewski@chb.unicancer.fr` (R. Modzelewski)

¹<http://mloss.org/software/view/562/>

²<https://github.com/jlerouge/crino>

during the last years, with important needs in the analysis of medical images, natural scenes or document images.

Depending on the application domain, an image labeling problem can be very challenging. It has to deal with a lot of variability, especially when tackling a real-world domain such as medical images or natural scenes labeling. Other difficulties may also include poor quality images or a large number of classes.

One can oppose two kinds of approaches for image labeling: dedicated approaches and learning-based approaches. Dedicated approaches often rely on a priori models of the images and/or of the labels. These models can be either handcrafted, unsupervisedly learned, or statistically computed on a database. In opposition, learning-based approaches directly estimate a decision function that links pixels to their labels by exploiting a labeled image database. It makes the system more versatile, at the expense of an offline supervised learning procedure.

In many difficult application domains, dedicated methods are still the state-of-the-art methods, using strong priors on the data. It is the case in medical imaging, where 2D models (atlas) are generally fitted on the new data in order to label its pixels [1, 2, 3, 4]. However, recent advances in machine learning and computer vision make the learning-based approaches more and more accurate, and we believe that they will be able to outperform dedicated methods when they are able to efficiently handle the a priori knowledge on the data.

The Input/Output Deep Architecture (IODA) is an original learning-based approach for image labeling that relies on deep neural network architectures. It directly links a whole image to a whole label map, assigning a label to each pixel using a unique neural network forward. Instead of designing a handcrafted a priori model on labels, we propose to automatically learn the dependencies between labels. The originality of IODA is to transpose DNN pre-training input trick to outputs, in order to learn a high level representation of labels. We apply it on a medical imaging labeling problem on which we outperform the state-of-the-art method achieved by a dedicated approach based on an a priori model [5].

The article is organised as follows: section 2 is dedicated to a review of existing learning-based approaches for image labeling tasks. In section 3 we recall the principles of neural networks and deep architectures, and we describe our IODA approach for image segmentation and labeling. The method is evaluated on a toy problem in section 4, and on a real-world medical image segmentation problem in section 5.

2. Related works on image labeling methods

From a machine learning point of view, the image labeling process is seen as a classification process, trying to find the best function f over a labeled image dataset, that minimizes the criterion $J = \mathcal{L}(Y, f(X))$, \mathcal{L} being a loss function, and the domain of f is given by

$$\begin{aligned}
f : X = \{x\}^{n \times m} &\rightarrow Y = \{y\}^{n \times m} \\
\mathcal{X}^{n \times m} &\rightarrow \mathcal{Y}^{n \times m}
\end{aligned} \tag{1}$$

where $n \times m$ is the image size, $x \in \mathcal{X}$ are features extracted from a pixel, and $y \in \mathcal{Y}$ is the label of the corresponding pixel. For example, one can consider the raw pixels of a greyscale image as input ($\mathcal{X} = \mathbb{R}$), the raw pixels of a color image ($\mathcal{X} = \mathbb{R}^3$), or a set of p features extracted from the neighbourhood of the current pixel ($\mathcal{X} = \mathbb{R}^p$). For this latter example, the domain of f becomes $\mathbb{R}^{p \times n \times m} \rightarrow \mathcal{Y}^{n \times m}$.

In the literature, one can oppose two kinds of approach for learning-based image labeling methods:

- performing a local, independent labeling of the pixels of an image, through the distribution $p(y|x)$
- performing a global image labeling method at the image level, through the distribution $p(Y|X)$

We now describe these two kinds of approaches.

2.1. Independent pixel labeling approaches

A first straightforward method for performing image labeling using a learning approach is to perform pixel labeling using a suitable feature set (textures, color, etc.) and a classifier [6, 7] that learns the local dependencies $p(y|x)$. Features are generally computed on the neighbourhood of the current pixel. Thus only local decisions are taken and the global function f is not sought.

Moreover, as the pixel classification stage does not output homogeneous regions, these methods are often followed by a post processing segmentation stage whose aim is to reconstruct smoothed label map based on a local decision [8, 9] Nevertheless, these sequential classification-then-segmentation approaches do not modelize the whole input distribution $p(X)$, nor the whole output distribution $p(Y)$.

As shape and label areas are strongly dependent, the pixel classification and the area segmentation should be performed together. Therefore, local pixel labeling approaches appear sub-optimal. This is related to the famous segmentation/recognition issue (also known as Sayre's paradox) saying that an object cannot be recognized before being segmented, but cannot be segmented before being recognized.

2.2. Global image labeling approaches

In the general pattern recognition domain, the segmentation/recognition issue is classically circumvented using global approaches taking a whole segmentation and recognition decision.

In this kind of approaches, the global function f is estimated. Unlike the independent pixel labeling approaches, we expect from the learning process to

model the input and output distributions, $p(X)$ and $p(Y)$. State-of-the-art learning methods for image labeling are 2D-probabilistic approaches extended from 1D method such as Hidden Markov Model (HMM) and Conditionnal Random Field (CRF). Structured output SVM approaches has also been explored for sequence labeling.

In the HMM framework, the joint probability $p(X, Y)$ is modeled, implying the (false) assumption that observations X are independent [10]. The CRF overcome this problem by modeling the conditional probability $p(Y|X)$ instead of $p(X, Y)$ [11]. Probabilistic methods have proven to be effective on 1D sequences with numerous applications such as information extraction in text, handwriting and voice recognition, or even 1D-signal segmentation. These methods have been adapted to 2D-signals through either Markov Random Field (MRF) [12, 13] or 2D-CRF [14, 15, 9], but they both suffer from a time consuming and sub-optimal decoding process such as HCF or ICM [16, 17]. Indeed, one has to search for the best path among the huge number of possible paths in the observation trellis which dramatically increases with the signal and output size.

In structured output SVM approaches [18] and kernel dependency estimation [19], a kernel joint projection evaluates the co-occurrence probability of an observation X and a label Y . Although these approaches can theoretically handle complex output spaces, the inference problem of finding the best label sequence knowing the model is a hard problem. It prevents the approach from tackling problems where the dimension of the sequence is large, as it is the case for image segmentation.

In this paper, we assume that other machine learning methods such as neural network are able to perform a global image segmentation and labeling task, modeling the underlying problem of estimating $p(Y|X)$.

Estimating $p(y|X)$, even if X has a great dimension, can be achieved through Deep Neural Network (DNN) using unsupervised pre-training or regularized learning process, through the modelization of $p(X)$. The learning of $p(X)$ can be performed either independently [20, 21] or jointly [22, 23] to the learning of $p(y|X)$. These approaches have shown to be efficient on numerous problems such as natural language processing [24], speech recognition [25] or handwriting recognition [26].

In this work, we propose to address directly the image labeling problem, that is the estimation of $p(Y|X)$. Our key idea is to extend the DNN input pre-training and adapt it to the output pre-training, providing the label distribution $p(Y)$.

While input pre-training has given to neural networks the ability to deal with high dimensional input space, we assume that output pre-training allows neural networks to deal with high dimensional output space.

The next section is dedicated to a recall on neural networks, before presenting our approach.

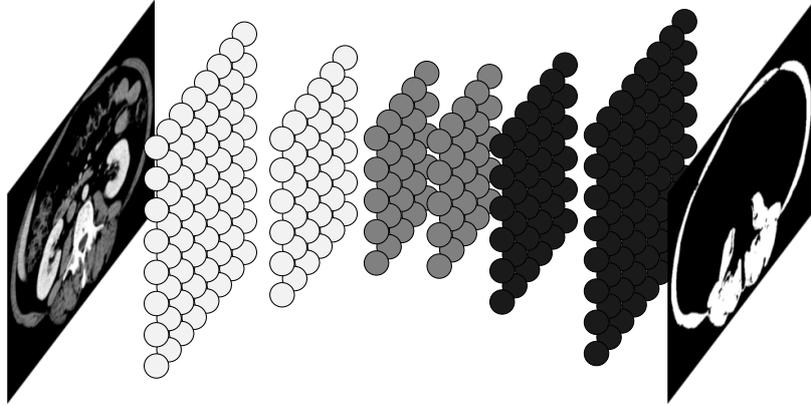


Figure 1: The IODA architecture. It directly links the pixel matrix to the label matrix. The input layers (left, light) are pre-trained to provide a high level representation of the pixels, while the output layers (right, dark) are pre-trained to learn the a priori knowledge of the problem.

3. Input/Output Deep Architecture (IODA)

For the image labeling task, we choose a global approach where a Deep Neural Network (DNN), a kind of Feedforward Artificial Neural Network (FANN), is used as the global decision function f .

The FANN architecture is a common artificial neural network topology used for supervised learning. In this architecture, information is processed by a sequence of computational layers. At decision step, information always flows from input to output, without any feedback. A FANN can be composed of 1, 2 or many layers [27]. In order to model all continuous functions on compact subsets of \mathbb{R}^n or logical function like XOR, at least 2 computational layers must be involved. A FANN with 2 or more layers is called a Multi-Layer Perceptron (MLP). A MLP with more or far more than 2 layers can also be called a Deep Neural Network (DNN). DNN are typically used in image problems like character recognition [28]. A MLP is usually learned by an algorithm called gradient back-propagation, that cleverly performs a gradient descent through the layers.

Nevertheless, the deepest layers of a DNN are hardly trained by this technique. To help the learning of a DNN, an unsupervised pre-training is performed on deepest layers, through the use of auto-encoders (AE) which learn feature distribution [20, 21, 29, 30, 31].

In Input Output Deep Architecture (IODA), we propose to use the pre-training trick with AE not only for the input space but also for the output space, in order to learn the labels distribution as well as the features distribution. The global architecture of IODA is presented in Figure 1.

3.1. Notations and building blocks

In this section, we present the notations that will be used in this work. Then we discuss how to build a DNN using the input pre-training trick with AE, and eventually how to build a IODA with the same principle adapted to the output space.

In the preceding section, the input was a two-dimensional image matrix X and the output a label map Y . In this section, we will consider that input and output are one-dimensional flatten versions of the data, respectively an input vector \mathbf{x} and a label vector \mathbf{y} .

3.1.1. Baseline Multi Layer Perceptron

We denote :

- a *layer*, the unit of computational operations,
- a *representation*, the unit of data.

Within this framework, the smallest MLP with universal approximation property has 2 layers (an input layer and an output layer), whereas this very same MLP has 3 representations (an input, a hidden and an output representations).

Let us consider a MLP of N layers. Each of the $N + 1$ representations is denoted \mathbf{r}_l with $l \in [0 \dots N]$. \mathbf{r}_0 is the input representation, i.e. the features \mathbf{x} , and \mathbf{r}_N the output representation, i.e. the estimated labels $\hat{\mathbf{y}}$.

Each layer l performs the following operation at forward step,

$$\mathbf{r}_l = f_l(\mathbf{W}_l \times \mathbf{r}_{l-1} + \mathbf{b}_l) \quad (2)$$

where \mathbf{r}_{l-1} and \mathbf{r}_l are respectively the input and the output of the layer l , \mathbf{W}_l a matrix representing a linear transformation corresponding to neuron *weights*, \mathbf{b}_l a vector of offsets corresponding to neuron *biases*, and f_l a non-linear differentiable transformation corresponding to neuron *activation function*. If $\mathbf{r}_{l-1} \in \mathbb{R}^m$ and $\mathbf{r}_l \in \mathbb{R}^n$, then \mathbf{W}_l is in $\mathbb{R}^{n \times m}$ and \mathbf{b}_l in \mathbb{R}^n . The lower the index l of a representation \mathbf{r}_l or of a layer $(f_l, \mathbf{W}_l, \mathbf{b}_l)$ is, the *deeper* it is. At the opposite the greater the index is, the *higher* the representation or the layer is. Figure 2 sums up the adopted notations on a 2-layer perceptron.

This is a gradient machine, i.e. the criterion (e.g. squared error, negative log likelihood, cross entropy ...) that is used to train the machine is differentiable according to its parameters. Thus the machine can be trained by gradient descent. In MLP, parameters are modified layer by layer backward from the output layer to the input layer. This is the so-called gradient back-propagation algorithm.

3.1.2. Auto-Encoder

An Auto-Encoder (AE) consists in a 2-layer MLP (see Figure 3). It tries to recover its input \mathbf{x} at its output $\hat{\mathbf{x}}$ [32].

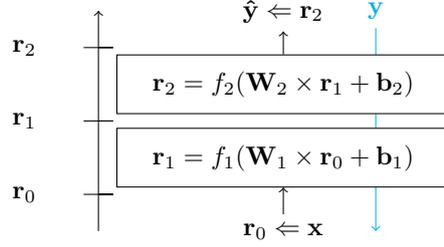


Figure 2: A 2-layer perceptron with adopted notation, input \mathbf{x} , output $\hat{\mathbf{y}}$ and target \mathbf{y}

The first layer applies a transformation from the input space \mathbb{R}^m to a hidden space \mathbb{R}^n , the second layer inverts this transformation, i.e. does a back-projection in the original input space. When $m < n$ the first layer performs a compression of the data, and the second layer a decompression. The first layer is called *encoding layer*, the last layer is called *decoding layer*.

We denote the linear transformations ($\mathbf{U} \in \mathbb{R}^{n \times m}$, $\mathbf{a} \in \mathbb{R}^n$) and ($\mathbf{V} \in \mathbb{R}^{m \times n}$, $\mathbf{c} \in \mathbb{R}^m$), and the non-linear transformations g and h , for the encoding layer and for the decoding layer respectively. The compressed or encoded representation at the output of the encoding layer is noted \mathbf{e} and conversely the decompressed or decoded output representation of the decoding layer is noted \mathbf{d} . Obviously, the input representation \mathbf{x} and the decoded representation \mathbf{d} have the same size m .

An AE is learned through the back-propagation algorithm with \mathbf{x} as input and as target, and $\mathbf{V} = \mathbf{U}^\top$ at initialization. Noise or transformations can be applied to \mathbf{x} solely at the input to increase its generalization power.

At decision, when you give an example \mathbf{x}_k to an AE it estimates the example itself $\hat{\mathbf{x}}_k$. If \mathbf{x}_k and $\hat{\mathbf{x}}_k$ are similar, \mathbf{x}_k is likely to happen according to the training set \mathcal{X} ; if \mathbf{x}_k and $\hat{\mathbf{x}}_k$ are dissimilar, \mathbf{x}_k is not likely to come from the same phenomenon that gives the training set \mathcal{X} : that is the modeling of $p(\mathbf{x})$.

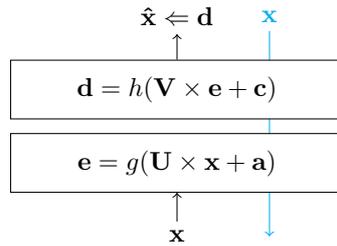


Figure 3: A 2-layer auto-encoder, input \mathbf{x} , output $\hat{\mathbf{x}}$ and target \mathbf{x}

3.2. DNN extension and Input/Output Deep Architecture (IODA)

In DNN, the deepest layers are used to transform the input space into a simpler yet more suitable space for the supervised task. Nevertheless, back-propagation is not efficient to train these deepest layers from random initialized parameters.

To prevent the DNN to fall into a local minimum far from a good solution, a smart initialization of the parameters is undertaken. This *pre-training* strategy consists in learning auto-encoders in an unsupervised way and stacking only their first layer to build a DNN with the desired architecture. Thereafter, the DNN is trained with a standard back-propagation supervised learning. In this way, the pre-training enables the first layers of the DNN to build a smarter representation of the input space to simplify the supervised task.

In order to address high dimensional but correlated output space, such as label map in image labeling problem, we propose to use the same AE trick as for DNN this time not only on the input space but also on the output space. AE are learned backward from the targets, in order to learn their distribution over the output space $p(\mathbf{y})$. The aim is to simplify the final supervised task by reducing the output representation. We call this architecture *Input/Output Deep Architecture (IODA)*.

To sum up, the IODA training involves:

- an unsupervised pre-training of the input layers,
- an unsupervised pre-training of the output layers, which is specific to IODA,
- a final standard back-propagation supervised learning.

Figure 4 displays the whole process on a 5-layer MLP.

3.2.1. Pre-training of input layers

Input pre-training with AEs occurs forward from the deepest layer. At each step, the encoding part of AEs are kept aside to initialize the final IODA.

Figure 4(a) shows the pre-training of the two first (deepest) layers of a 5-layer IODA. We note m the size of the input representation of the IODA, n the size of the first hidden representation and o the size of the second one.

For the first step of the input pre-training (Figure 4(a) left), an auto-encoder is trained with back-propagation on the input representation of the IODA. The size of the input representation and of encoded representation in this AE should be respectively m and n to mimics the final IODA. Thus if \mathbf{W}_1 of the IODA is in $\mathbb{R}^{n \times m}$ then \mathbf{U}_1 of the auto-encoder must be in $\mathbb{R}^{n \times m}$. Moreover, the non linear transformation g_1 of AE should be the same as f_1 the first non-linear transformation of the IODA.

After the training of the AE, the linear transformation $(\mathbf{U}_1, \mathbf{a}_1)$ of the encoding layer is kept aside to initialize the first layer of the IODA. Furthermore, the encoded representation \mathbf{e}_1 of all the training examples is also kept in order to feed the second step.

For the second step of the pre-training (Figure 4(a) right), we repeat the latter operation for the second layer of the IODA. An other auto-encoder is trained with back-propagation on \mathbf{e}_1 the encoded representation from the first auto-encoder. This time, the size of the input representation and of encoded representation in the AE should be respectively n and o ; and the non-linear transformation g_2 be the same as f_2 .

At the end of that step, the linear transformation $(\mathbf{U}_2, \mathbf{a}_2)$ of the encoding layer is kept aside to initialize the second layer of the IODA.

Stacking more AE can be repeated if more input layers than in the given example are involved.

Eventually, input layers of desired final IODA is initialized by the weights computed on this input pre-training step (Fig 4(c)),

- $\mathbf{W}_1 \leftarrow \mathbf{U}_1, \mathbf{b}_1 \leftarrow \mathbf{a}_1$, for the first layer,
- $\mathbf{W}_2 \leftarrow \mathbf{U}_2, \mathbf{b}_2 \leftarrow \mathbf{a}_2$, for the second layer.

3.2.2. Pre-training of output layers specific to IODA

Operations are the same than in the latter input pre-training with two exceptions: they are undertaken backward from the highest layer and the parameters kept aside for the initialization of the final IODA are from the decoding part of AEs.

Figure 4(b) shows the pre-training of the two last (highest) layers of a 5-layer IODA.

A first pre-training step (Fig 4(b) left) is done with an auto-encoder on the label vector \mathbf{y} . The second linear transformation \mathbf{V}_5 of the AE should have the same shape as \mathbf{W}_5 , the last linear transformation of the IODA. Furthermore, the second non-linear transformation h_5 of the AE should be the same as f_5 , the last non-linear transformation of the IODA.

Let's be careful, this time it is the parameters of the second layer of the AE, i.e. the decoding layer, which are kept contrary to the standard DNN pre-training. Thus, after training the AE, the linear transformation $(\mathbf{V}_5, \mathbf{c}_5)$ is saved to initialize the last layer of the IODA. Moreover, \mathbf{e}_5 the encoded representation for all the training examples is kept in order to feed the next step.

A second pre-training step (Fig 4(b) right) is done with an other auto-encoder on \mathbf{e}_5 . The second non-linear transformation h_4 of this AE should be the same as the penultimate non-linear transformation f_4 of the IODA, as well as the shape of its second linear transformation \mathbf{V}_4 should be the same shape as \mathbf{W}_4 the penultimate linear transformation of the IODA.

At the end of that step, the linear transformation $(\mathbf{V}_4, \mathbf{c}_4)$ of the decoding layer is kept aside to initialize the penultimate layer of the DNN.

As for the input pre-training, these operations can be repeated and more AEs stacked if the architecture consists in more output layers.

Eventually output layers of the desired IODA are initialized by the weights computed on precedent pre-training steps:

- $\mathbf{W}_4 \leftarrow \mathbf{V}_4, \mathbf{b}_4 \leftarrow \mathbf{c}_4$, for the penultimate layer,
- $\mathbf{W}_5 \leftarrow \mathbf{V}_5, \mathbf{b}_5 \leftarrow \mathbf{c}_5$, for the last layer.

3.2.3. Final supervised training

After pre-trainings of input layers and output layers, a standard back-propagation is undertaken with target \mathbf{y} (Fig 4(c)) on the whole MLP.

Let note that in the 5-layer architecture given as an example it exists a *link* layer, the layer number 3, between the input layers and output layers. It is not pre-trained and thus it has randomized parameters before the last back-propagation. A slightly different approach may supervisedly train this link layer before doing a last full back-propagation at a risk of over-fitting.

3.2.4. IODA training algorithm

The algorithm 1 describes the whole learning procedure for training a IODA. We assume the existence of these two functions:

- $X' \leftarrow \text{MLPFORWARD}([\mathbf{W}_1, \dots, \mathbf{W}_K], X)$ that propagates X through layers $[\mathbf{W}_1, \dots, \mathbf{W}_K]$,
- $[\mathbf{W}'_1, \dots, \mathbf{W}'_K] \leftarrow \text{MLPTRAIN}([\mathbf{W}_1, \dots, \mathbf{W}_K], X, Y)$ that trains layers $[\mathbf{W}_1, \dots, \mathbf{W}_K]$ using back-propagation algorithm according to a labeled dataset (X, Y) .

With this notation an AE is trained by $[\mathbf{U}, \mathbf{V}] \leftarrow \text{MLPTRAIN}([\mathbf{W}, \mathbf{W}^\top], X, Y)$ where \cdot^\top denotes the transposition. Then we can drop \mathbf{V} if we want to keep the encoding part only, or drop \mathbf{U} if we want to keep the decoding part.

For the sake of clarity, hyperparameters such as non-linear function of each layer does not appear in the algorithm, and all the parameters of a layer i (the linear transformation \mathbf{W} and the bias b) are gathered into the generic variable \mathbf{W}_i .

Algorithm 1 Simplified IODA training algorithm

Input: X , a training feature set of size $Nb_{\text{examples}} \times Nb_{\text{features}}$

Input: Y , a corresponding training label set of size $Nb_{\text{examples}} \times Nb_{\text{labels}}$

Input: N_{input} , the number of input layers to be pre-trained

Input: N_{output} , the number of output layers to be pre-trained

Input: N , the number of layers in the IODA, $N_{\text{input}} + N_{\text{output}} < N$

Output: $[\mathbf{W}_1, \mathbf{W}_2, \dots, \mathbf{W}_N]$, the parameters for all the layers

Randomly initialize $[\mathbf{W}_1, \mathbf{W}_2, \dots, \mathbf{W}_N]$

Input pre-training

$R \leftarrow X$

for $i \leftarrow 1..N_{\text{input}}$ **do**

 {*Training an AE on R and keeps its encoding parameters*}

$[\mathbf{W}_i, \mathbf{W}_{\text{dummy}}] \leftarrow \text{MLPTRAIN}([\mathbf{W}_i, \mathbf{W}_i^T], R, R)$

 Drop $\mathbf{W}_{\text{dummy}}$

$R \leftarrow \text{MLPFORWARD}([\mathbf{W}_i], R)$

end for

Output pre-training

$R \leftarrow Y$

for $i \leftarrow N..N - N_{\text{output}} + 1$ **step** -1 **do**

 {*Training an AE on R and keeps its decoding parameters*}

$[\mathbf{U}, \mathbf{W}_i] \leftarrow \text{MLPTRAIN}([\mathbf{W}_i^T, \mathbf{W}_i], R, R)$

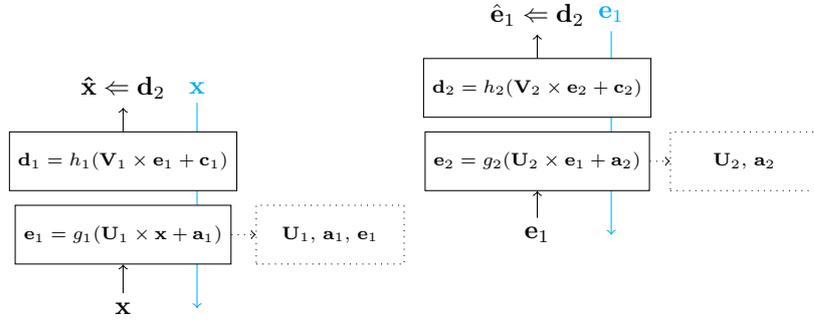
$R \leftarrow \text{MLPFORWARD}([\mathbf{U}], R)$

 Drop \mathbf{U}

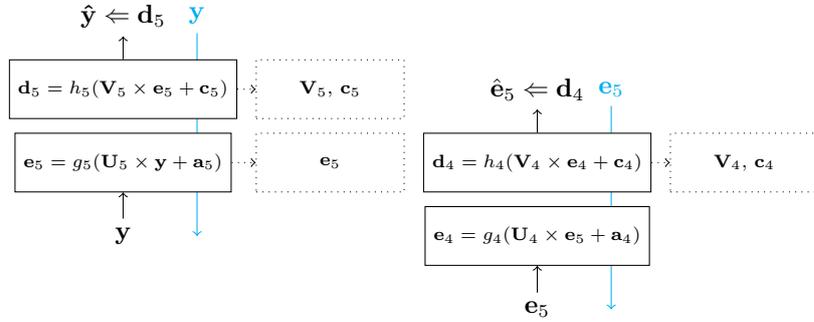
end for

Final supervised learning

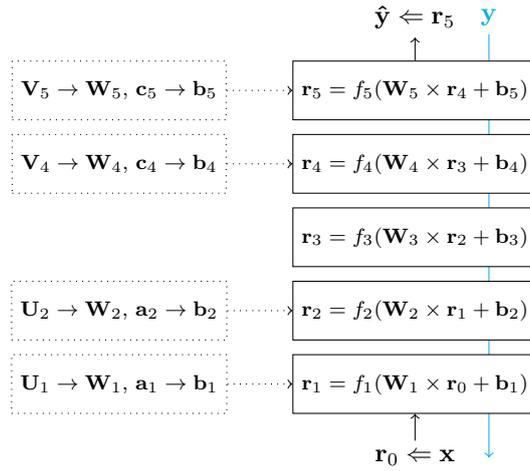
$[\mathbf{W}_1, \mathbf{W}_2, \dots, \mathbf{W}_N] \leftarrow \text{MLPTRAIN}([\mathbf{W}_1, \mathbf{W}_2, \dots, \mathbf{W}_N], X, Y)$



(a) Input pre-training. Left : Learning of the first AE, input \mathbf{x} , output $\hat{\mathbf{x}}$, target \mathbf{x} and $g_1 \leftarrow f_1$. Right : Learning of the second AE, input \mathbf{e}_1 which is the encoded representation of the first AE, output $\hat{\mathbf{e}}_1$, target \mathbf{e}_1 and $g_2 \leftarrow f_2$.



(b) Output pre-training Left : Learning of the first AE, input \mathbf{y} , output $\hat{\mathbf{y}}$, target \mathbf{y} and $h_5 \leftarrow f_5$. Right : Learning of the second AE, input \mathbf{e}_5 which is encoded representation of the first AE, output $\hat{\mathbf{e}}_5$, target \mathbf{e}_5 and $h_4 \leftarrow f_4$.



(c) Final IODA, with pre-computed initial weights, input \mathbf{x} , output $\hat{\mathbf{y}}$ and target \mathbf{y}

Figure 4: Pre-trainings and training of a 5-layer IODA

4. Texture recognition experiments

We developed a Python library, named Crino, based on the Theano library[33]. It allows to build and train neural networks with a modular architecture, including IODA. Crino is available online³⁴ and is free to use for further research.

To demonstrate the validity of our proposition, we have performed experiments on a toy image dataset. We first describe the dataset, then the different experimental setups. We finally present and discuss the results we have obtained.

4.1. Toy dataset

We have generated a toy image dataset for a texture recognition task. The input examples are artificial images composed of two textures, taken from the Brodatz texture archive⁵. The background is taken from Texture 77, on top of which is drawn the foreground with Texture 17. The foreground consists in the portion of a disk included between two concentric circles whose center and radii are variable (randomly chosen for each sample). The labels are binary images denoting the class of the pixels, 0's are for the background pixels and 1's are for the foreground pixels. All images are 128×128 pixels, and inputs are normalized between 0 and 1. Our training and validation sets are composed of 500 images each. Two examples of the validation set are shown on Figure 5.

On this kind of images, the internal dependencies among the label structures are very high, and therefore constitutes a suitable problem for evaluating the IODA abilities. A better representation of the output space should be available and pre-training on output should improve the results of this supervised task.

4.2. Experimental setup

For this toy problem, we have built using Crino a 3-layer, 4-layer and 5-layer neural networks with a MSE criterion. For all of them, the size of the input and output representations is 128×128 .

For 3-layer architectures, we have tested four hidden representation geometries: (256,256), (512,512), (1024,1024) and (2048,2048) neurons. Input pre-training has been performed from 0 to 2 layers, and output pre-training from 0 to 2 layers also. Let us emphasise that the total number of pre-trainings can not exceed 2 since at least one layer must be free of autoencoding pre-training. Finally, 4×6 setups have been trained and evaluated. Setups that share the same number of hidden neurons starts with the same initialisation weights. The results are gathered in Table 1.

For 4-layer architectures, the same procedure has been applied. Four hidden representation geometries have been evaluated: (256,128,256), (512,256,512), (1024,512,1024) and (2048,1024,2048) neurons. Input and output pre-trainings

³<http://mloss.org/software/view/562/>

⁴<https://github.com/jlerouge/crino>

⁵<http://www.ux.uis.no/~tranden/brodatz.html>

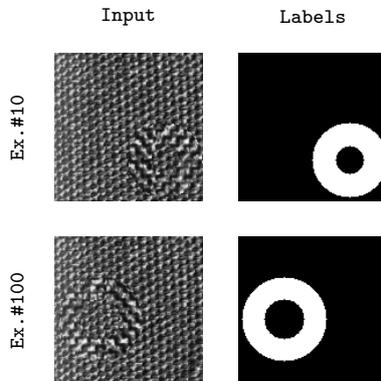


Figure 5: Two random validation examples (left) artificial image inputs (right) image labels

vary each from 0 to 3 layers, with a total number of 3 pre-trainings. The 4×10 setups results are presented in table 2.

For 5-layer architectures, four hidden representation geometries have been evaluated: (256,128,128,256), (512,256,256,512), (1024,512,512,1024) and (2048,1024,1024,2048). Only some pre-training configurations have been tested, including the most successful strategies within 3 and 4 layers architectures (please refer to the next subsection).

If the setup contains at least one input pre-trained layer and one output pre-trained layer, it falls into our proposed IODA framework.

4.3. Parameterization

For each autoencoder, the weights are randomly initialized according to the work of [34] in order to perform a faster convergence of the gradient backpropagation algorithm. It uses a uniform distribution $\mathcal{U}(-l, l)$, where $l = \sqrt{\frac{6}{mn}}$ and where m and n are respectively the input and the output sizes of the autoencoder. The biases are initially null.

For input and output pre-trainings, auto-encoders are trained with a batch gradient descent of 100 images, controlled by a validation set in order to minimize error while avoiding overfitting. Input pre-training has therefore been stopped after 300 iterations, while only 100 iterations were enough for output pre-training since a strong overfit appeared around 200 iterations. The final supervised learning is also performed with a batch gradient descent. As we have 500 training examples, it means that the parameters are updated five times per iteration.

We have chosen an adaptive learning rate, i.e. that is varying at each iteration. Our strategy consists in increasing the learning rate after some fixed number of consecutive iterations that improves the learning criterion. In case of a degradation, we keep decreasing the learning rate until it provides a better

value of the criterion in comparison to the previous iteration. The initial value of the learning rate is 10 (same value for all the architectures).

4.4. First qualitative results

For a first qualitative result, we propose to focus on a given geometry with three different pre-training strategies. The considered geometry has 3 layers and 256 units in each hidden representations. We call these three strategies are NDA, IDA and IODA :

NDA - DNN without pre-training : No pre-training is done at all, solely a supervised learning with standard back-propagation is achieved.

IDA - DNN with input pre-training : The input layer is pre-trained using an auto-encoder on the input data. Then a supervised fine tuning is achieved.

IODA - DNN with input and output pre-training : The input and output layers are pre-trained using respectively an auto-encoder on the input data and an auto-encoder on the output data. A supervised fine tuning is then achieved.

Figure 6 shows the output of each architecture for the first example shown in Figure 5, after an increasing number of supervised iterations (pre-training has already been performed). As one can see, the best results are achieved with the input and output pre-trained architecture (IODA), while input pre-trained architecture (IDA) outperforms the non pre-trained architecture (NDA). One can also observe that the global output structure has already been learned by IODA after only 10 supervised learning iterations. It shows that the IODA strategy is much more efficient than the IDA strategy, as it speeds up the supervised learning. Finally, after a significant number of iterations, IODA is able to locate more accurately the texture change.

4.5. Quantitative results

Tables 1 and 2 shows a quantitative evaluation over the whole test dataset (500 images), using all the setups defined above. Several comments can be made out of these experiments.

First, one can observe that the results are strongly dependent from the setup. The pre-training strategies seems to have more influence on the results than the number of layer and the number of hidden units. One can notice that 3-layer or 4-layer architectures lead to very close performance. Globally, input pre-trained setups are ranked first. Among them, IODA setups, i.e. with at least one pre-trained input layer and one pre-trained output layer, achieve the best results. This demonstrates the interest of our approach. Moreover, setups with pre-trained output layers only do not guarantee good results.

The results of 5 layer architectures are globally worse than those of 3 and 4 layer architectures and for this reason not entirely reported in this article.

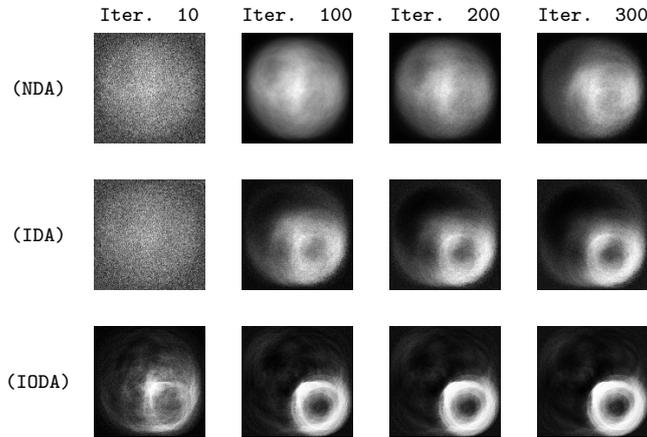


Figure 6: Evolution of the output image of the architecture according to the number of batch gradient descent iterations for the three learning strategies, using the validation example #10.

Among them, the best result is achieved using 3 pre-trained input and 1 pre-trained output, with the (2048,1024,1024,2048) geometry. It leads to a MSE of $4.42e - 2$ over the test dataset (*vs.* $3.48e - 02$ for the best 3-layer architecture). The behaviour with respect to the geometry and the pre-training configurations follows those of the 3 and 4 layer architecture.

In the next section, we propose the application of IODA to a real-world problem.

5. Application to a medical imaging problem

In this section, we apply the IODA architecture to a real-world problem which consists of labeling each pixel of scanner images into 2 classes. We compare the results of our approach with the state-of-the-art method for this challenging task.

5.1. Medical image segmentation task

The real-world problem addressed in this section consists in estimating the sarcopenia level, based on the scanner image labeling which is usually manually performed. Sarcopenia (loss of skeletal muscle mass) is of interest in the medical research field because this data could predict the prognosis of multiple cancers[35, 36]. Sarcopenia is assessed by manually segmenting the skeletal muscles on Computer Tomography (CT) scan slices taken at the third lumbar vertebra (L3) level.

This task is very time-consuming since the overall segmentation process take in average 4 minutes per patient for an experimented physician. Therefore, there

Table 1: Experiments on toy data set for a 3-layer architecture with different hidden sizes and pre-training setups, sorted by ascending test error. Test errors provided by IODA are in bold.

X	Architecture					Train error	Test error	
	\mathbf{r}_1	\mathbf{r}_2	\hat{Y}					
128^2	■	2048	□	2048	⊗	128^2	2.64e-02	3.48e-02
128^2	■	1024	□	1024	⊗	128^2	3.11e-02	3.91e-02
128^2	■	512	□	512	⊗	128^2	3.26e-02	4.10e-02
128^2	□	2048	□	2048	⊗	128^2	3.86e-02	4.59e-02
128^2	■	256	□	256	⊗	128^2	4.05e-02	4.85e-02
128^2	□	1024	□	1024	⊗	128^2	4.44e-02	5.13e-02
128^2	□	512	□	512	⊗	128^2	4.81e-02	5.50e-02
128^2	■	2048	■	2048	□	128^2	5.20e-02	5.75e-02
128^2	□	256	□	256	⊗	128^2	6.16e-02	6.75e-02
128^2	■	1024	■	1024	□	128^2	6.29e-02	6.77e-02
128^2	■	2048	□	2048	□	128^2	6.30e-02	6.79e-02
128^2	■	1024	□	1024	□	128^2	7.09e-02	7.55e-02
128^2	■	512	■	512	□	128^2	7.13e-02	7.60e-02
128^2	■	256	■	256	□	128^2	7.52e-02	7.98e-02
128^2	■	512	□	512	□	128^2	8.03e-02	8.48e-02
128^2	■	256	□	256	□	128^2	8.31e-02	8.75e-02
128^2	□	2048	⊗	2048	⊗	128^2	8.86e-02	9.37e-02
128^2	□	2048	□	2048	□	128^2	9.03e-02	9.40e-02
128^2	□	1024	⊗	1024	⊗	128^2	9.60e-02	1.01e-01
128^2	□	1024	□	1024	□	128^2	1.03e-01	1.06e-01
128^2	□	512	⊗	512	⊗	128^2	1.06e-01	1.10e-01
128^2	□	256	⊗	256	⊗	128^2	1.25e-01	1.28e-01
128^2	□	512	□	512	□	128^2	1.26e-01	1.28e-01
128^2	□	256	□	256	□	128^2	1.41e-01	1.41e-01

■ : input pre-training, □ : no pre-training, ⊗ : output pre-training.

Table 2: Experiments on toy data set for a 4-layer architecture with different hidden sizes and pre-training setups, sorted by ascending test error. Test errors provided by IODA are in bold.

X	Architecture							Train criterion	Test criterion	
	r_1	r_2	r_3	\hat{Y}						
128^2	■	2048	■	1024	□	2048	⊗	128^2	2.74e-02	3.62e-02
128^2	■	2048	□	1024	□	2048	⊗	128^2	2.95e-02	3.75e-02
128^2	■	1024	■	512	□	1024	⊗	128^2	3.59e-02	4.44e-02
128^2	■	1024	□	512	□	1024	⊗	128^2	3.70e-02	4.47e-02
128^2	■	512	□	256	⊗	512	⊗	128^2	3.45e-02	4.52e-02
128^2	■	1024	□	512	⊗	1024	⊗	128^2	3.52e-02	4.53e-02
128^2	■	2048	□	1024	⊗	2048	⊗	128^2	3.91e-02	4.85e-02
128^2	■	512	□	256	□	512	⊗	128^2	4.24e-02	5.00e-02
128^2	■	512	■	256	□	512	⊗	128^2	4.36e-02	5.21e-02
128^2	■	256	□	128	⊗	256	⊗	128^2	4.18e-02	5.23e-02
128^2	□	2048	□	1024	□	2048	⊗	128^2	4.64e-02	5.33e-02
128^2	■	256	□	128	□	256	⊗	128^2	4.59e-02	5.36e-02
128^2	■	256	■	128	□	256	⊗	128^2	4.76e-02	5.68e-02
128^2	□	1024	□	512	□	1024	⊗	128^2	5.07e-02	5.75e-02
128^2	□	512	□	256	□	512	⊗	128^2	5.34e-02	6.02e-02
128^2	■	2048	■	1024	■	2048	□	128^2	6.01e-02	6.50e-02
128^2	■	2048	■	1024	□	2048	□	128^2	6.37e-02	6.85e-02
128^2	□	2048	□	1024	⊗	2048	⊗	128^2	6.23e-02	6.96e-02
128^2	■	1024	■	512	■	1024	□	128^2	6.82e-02	7.28e-02
128^2	□	512	□	256	⊗	512	⊗	128^2	6.60e-02	7.39e-02
128^2	□	256	□	128	□	256	⊗	128^2	6.95e-02	7.50e-02
128^2	□	1024	□	512	⊗	1024	⊗	128^2	7.10e-02	7.74e-02
128^2	■	2048	□	1024	□	2048	□	128^2	7.72e-02	8.14e-02
128^2	■	512	■	256	■	512	□	128^2	7.84e-02	8.27e-02
128^2	■	1024	■	512	□	1024	□	128^2	7.94e-02	8.37e-02
128^2	■	256	■	128	■	256	□	128^2	8.02e-02	8.48e-02
128^2	■	512	■	256	□	512	□	128^2	8.37e-02	8.80e-02
128^2	■	256	■	128	□	256	□	128^2	8.52e-02	8.98e-02
128^2	□	2048	⊗	1024	⊗	2048	⊗	128^2	8.70e-02	9.23e-02
128^2	■	1024	□	512	□	1024	□	128^2	9.08e-02	9.42e-02
128^2	■	512	□	256	□	512	□	128^2	9.28e-02	9.63e-02
128^2	■	256	□	128	□	256	□	128^2	1.07e-01	1.10e-01
128^2	□	1024	⊗	512	⊗	1024	⊗	128^2	1.06e-01	1.12e-01
128^2	□	2048	□	1024	□	2048	□	128^2	1.12e-01	1.15e-01
128^2	□	256	□	128	⊗	256	⊗	128^2	1.11e-01	1.15e-01
128^2	□	512	⊗	256	⊗	512	□	128^2	1.16e-01	1.20e-01
128^2	□	256	⊗	128	⊗	256	⊗	128^2	1.33e-01	1.37e-01
128^2	□	1024	□	512	□	1024	□	128^2	1.41e-01	1.41e-01
128^2	□	512	□	256	□	512	□	128^2	1.41e-01	1.41e-01
128^2	□	256	□	128	□	256	□	128^2	1.41e-01	1.41e-01

■ : input pre-training, □ : no pre-training, ⊗ : output pre-training.

is a real need in automating the pixel labeling into two classes: skeletal muscle or not.

It is particularly challenging owing to numerous difficulties. More precisely, the method has to handle :

- The variability in the patients population:
 - the intrinsic variability in the anatomy of the patients, due to their variable genders, ages, morphologies (thin/fat) and medical states (healthy/ill) which modify significantly the shapes and the textures of the muscular, organic and fat tissues;
 - the variable organ positions : for example, kidney and liver can be present, partially or totally absent of the L3 slice;
 - the greyscale distribution overlap between muscle and internal organs.
- The variability of the images:
 - the variable quality of reconstructed CT images, due to the variable dose of radiations received by the patients during the CT acquisition (low dose / high dose);
 - the variable quantity of contrast agent that enhances the perfused tissues appearance;
 - the variable slice thickness (from submillimetric to 5mm);
 - the variable reconstruction filter used to reconstruct the images;

Figure 7 shows several images and their label. We believe that a machine learning approach could efficiently learn the intrinsic variability of this image labeling problem. For that, we dispose of a labeled dataset described thereafter.

5.2. Dataset and evaluation metrics

Our dataset consisted of 128 512×512 CT 16bit gray-level images. Each image has been manually labeled at the pixel level by a senior radiologist. Among them, 40 images come from lymphoma patients, the 88 others from breast cancer patients. As said previously, the database is composed of a wide variability of morphology, contrast and SNR between images.

We evaluated the proposed IODA automatic segmentation in comparison with the manual segmentation, and also with a referenced method[5] proposed by Chung *et al* briefly described below. In order to evaluate and compare their performance, the Jaccard index was used to measure the overlap between IODA (respectively Chung's) segmentation and the manual segmentation. We also provide the area relative difference (denoted as Diff.) metric which measures the rate of over/under-segmentation.

We now describe the Chung's dedicated method for skeletal muscle segmentation.

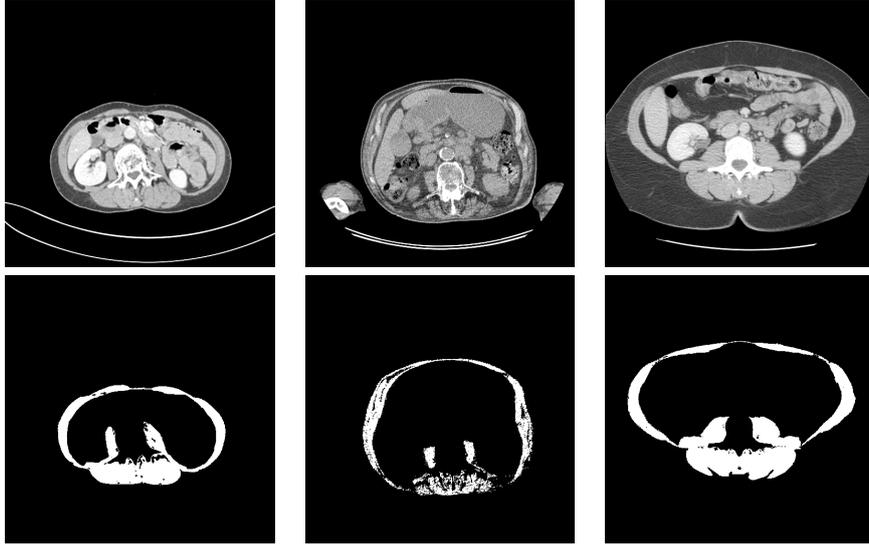


Figure 7: Examples of scanner images with their labeling below. Let us emphasize the morphology and image quality variability.

5.3. Reference automatic method

To the best of our knowledge, the only automated method for skeletal muscle segmentation at L3 was proposed by Chung *et al.* [5]. The method is based on standard shape prior coupled with an appearance model. The shape prior consists in computing the mean muscle shape on a labeled dataset, while the appearance model consists in estimating the probability distributions of both classes with a kernel density estimation method (Parzen window). In the decision process, the image is first filtered by thresholding the appearance model probability density. The final muscle segmentation is performed by an affine registration followed by a Free Form Deformation (FFD) based on a non-rigid registration. We reimplemented this method, since the original code is not available. The MATLAB toolbox MIRT⁶ has been used for the non-rigid registration performed by FFD.

The next subsection describes the application of IODA to the L3 skeleton muscle image labeling.

5.4. Application and setting of IODA for CT image labeling

The IODA architecture maps the 512×512 input greyscale images into a 512×512 label matrix. We have ignored the large parts of the CT images that are non informative (black areas) for every image of the dataset. It leads to

⁶<https://sites.google.com/site/myronenko/research/mirt>

smaller 311×457 -sized images around the patient body. This crop allows to reduce significantly the size of the architecture. Moreover, each training and test images are rigidly registered to a CT slice reference in order to reduce their size, shape and position variabilities.

For the task of learning the input and output dependencies, we have turned toward the use of 3-layer network with a MSE criterion, leading to 4 representations :

- one 311×457 -sized input representation,
- two 1500-sized hidden representations,
- one 311×457 -sized output representation.

The dimension of the hidden representations were empirically chosen, i.e. several geometry have been tried, and the best one have been chosen w.r.t. their performance obtained in validation. The whole resulting network is made of $145K$ hidden and output representation values, and contains $428M$ parameters. The first layer (between input and first hidden representations) and the mid-layer (between first hidden and second hidden representations) classically use *tanh* activation function; whereas the last layer (between second hidden and output representations) uses a *sigmoid* activation function. The first layer is pre-trained on the images, while the last layer is pre-trained on the groundtruth labels. Once pre-trained, a standard back-propagation has been performed on the whole network so as to fine tune the architecture. Since the medical imaging dataset is rather small, we have performed a gradient descent without batches, i.e. the parameters are updated at each iteration using the gradient computed on the whole training dataset. As the last layer gives a probability-like image output for the muscle tissue. This probability image must be thresholded in order to perform the final decision. This threshold has been chosen using a validation procedure in order to maximize the Jaccard index. It leads to an optimal value of 0.5. This value is the center of the output interval, but further experiments are needed to know if this value is a coincidence, or if it can be generalized.

As for the toy problem, we used our neural-network Python library, Crino, based on Theano which has not only a CPU backend, but also a GPU backend compatible with NVidia's CUDA technology. Thanks to this, we were able to run our tests on a range of different systems :

- A desktop computer featuring a NVidia Tesla C1060 GPU card with 4GB of onboard GDDR3 RAM;
- A laptop computer featuring an Intel Core i7-2760QM CPU (quad-core, 2.4GHz) and 8GB of DDR3 RAM.

Using the latter hardware setup, the overall training process of the IODA took about 35 minutes, split as follows :

- 15 minutes for pre-training of the first layer,

- 13 minutes for pre-training of the last layer,
- 7 minutes for fine-tuning the whole network.

With the same setup, the IODA forward step of the muscle tissue segmentation process takes in average 201.2 ± 8.6 milliseconds per image, in comparison to 4 minutes (± 2 min) needed by a senior radiologist on a homemade software [36]. However, the loading in memory of the network takes approximately 10 seconds, therefore it is better to process the images in batch mode.

5.5. Results

In this section we present qualitative and quantitative comparison between our neural network and state of the art approaches.

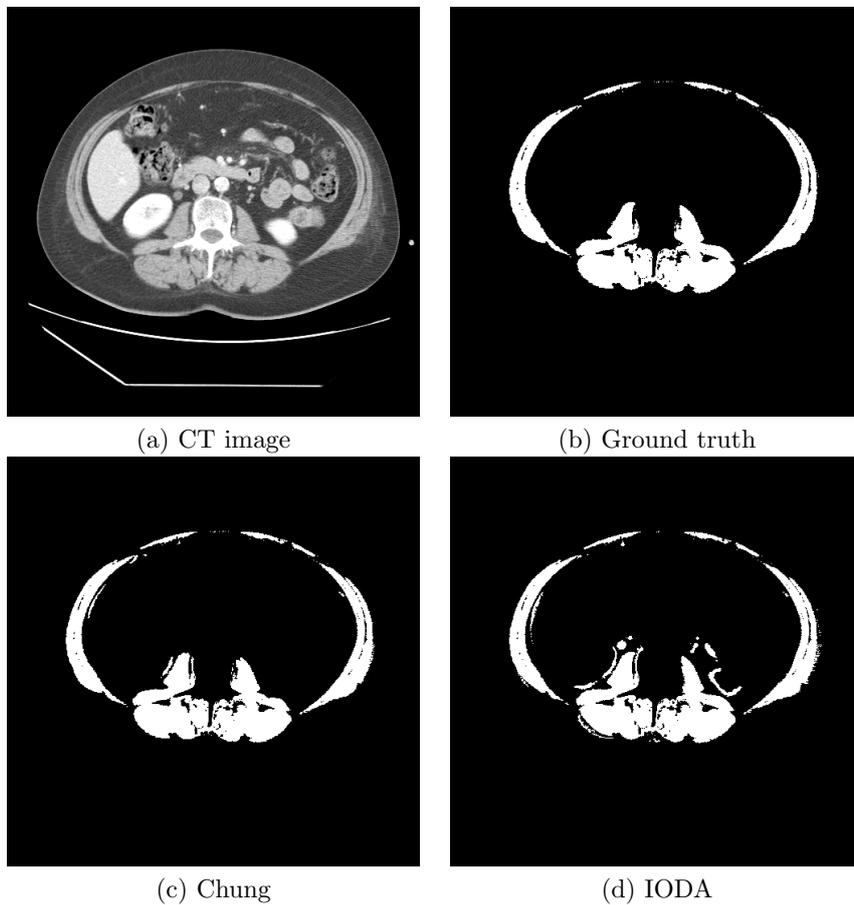


Figure 8: Non-sarcopenic patient

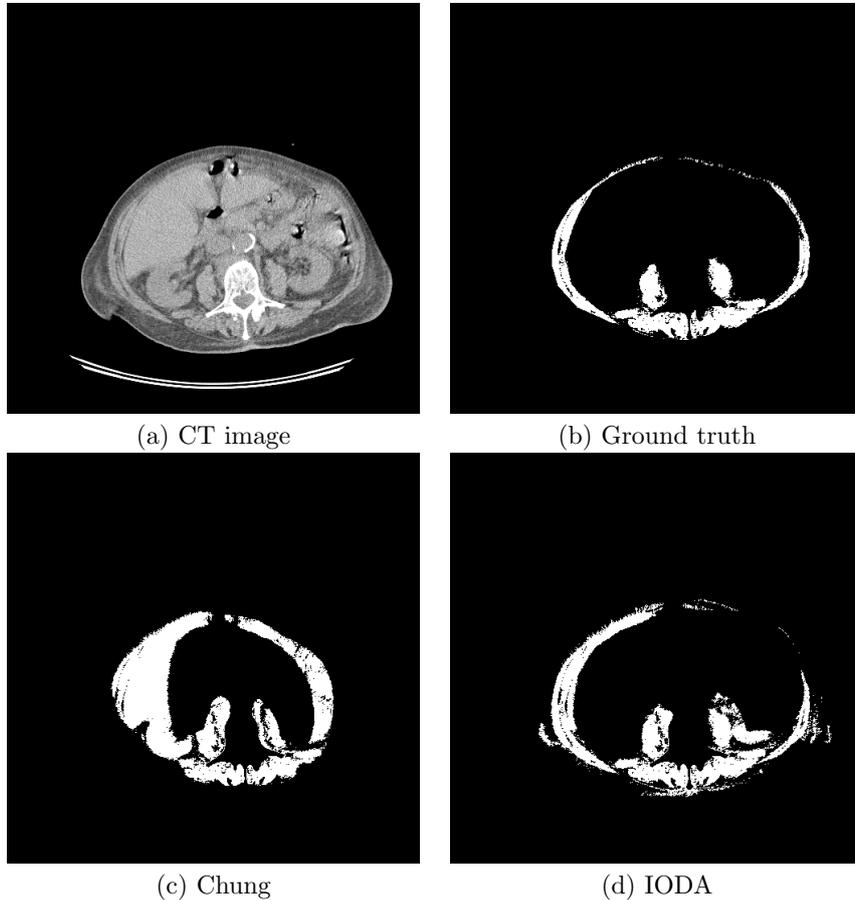


Figure 9: Sarcopenic patient

For a first experiment, 100 images have been used for learning the system parameters, and two images have been selected for displaying qualitative results. The first one has been considered as clean and "easy to segment" by a medical expert (See Figure 8), while the second is noisy and considered as "hard to segment" (see Figure 9). On each figure the raw image is represented in (a), ground truth labeling in (b), Chung labeling in (c) and IODA labeling in (d).

In the first case (see Figure 8), skeletal muscles, organs and fat are well delimited. Both approaches perform well : the state of the art method (Chung) achieves a Jaccard index of 97.2% whereas our proposed method (IODA) achieves 90.4%.

In the second case (see Figure 9), patient morphology is complex, and the image shows acquisition and reconstruction noise. Here, the Chung's method fails to correctly label the image with a Jaccard Index of 27.5% whereas the

IODA framework provides a much more accurate labeling leading to a Jaccard index of 64.0%.

We interpret these qualitative result by making the hypothesis that Chung method is not able to cope with strong variability of patient morphologies as it is based on a single average model, that is to say a single atlas. In opposition, our model embeds the variability of the patient morphologies through a learning process over the training set.

In order to confirm this hypothesis, we tested both methods on a large dataset, with significant variability in image contrast and in skeletal muscle shapes as suggested in Figure 7.

The regularization parameter for the FFD (λ) of Chung’s method and the output muscle probability threshold of our method have been subjected to a systematic search through a 4-fold cross validation procedure, in order to maximize the Jaccard Index of these methods in validation.

We have randomly split our 128 L3 images in cross-validation and test sets as follows :

- the cross-validation set contains 100 images, itself split in 4 subsets of 25 images each,
- the test set contains the remaining 28 images.

During the cross-validation step, and for each fold, 3 subsets (= 75 images) are used for training and the remaining subset (= 25 images) is used for validation. During the test step, the entire cross-validation set is used for training and the test set is used to compute the test performance.

Method	Diff. (%)	Jaccard (%)
Chung	-10.6±40.7	60.3±32.5
NDA	0.12±9.78	85.88±5.44
IDA	0.15±9.79	85.91±5.45
IODA	3.37±9.69	88.47±4.76

Table 3: Test performance of the automatic segmentation methods. All values are reported as mean \pm standard deviation.

Table 3 presents the test performance of this setup of IODA and Chung’s methods. For the sake of comparison, we have also reported the results achieved by NDA and IDA strategies, as defined in 4.4, using the same setup as IODA. Chung’s state of the art method gives worse results than one can expect. It confirms the hypothesis that shape and appearance prior of this method are not able to deal with too much variability. Moreover, the Diff. metric emphasizes an underestimation of the muscle tissue areas by Chung’s method.

On the other hand, IODA clearly outperforms Chung’s method according to both metrics. The Diff. metric suggests that IODA approach gives an average area close to the manual segmentation area. The Jaccard metric shows that IODA proposes a much better overlap of the skeletal muscles areas, and the behaviour of IODA is more stable than Chung’s method since the standard

deviations are significantly lower. Let us remark that NDA and IDA approaches perform much better than Chung’s method, but significantly worse than IODA. It is also of interest that NDA and IDA approaches give extremely similar results on this experiment. This is certainly due to the noisy texture of reconstructed scanner images which prevents from learning the features of the data.

6. Conclusion

In this article, we have presented a new method for image labeling that allows to learn prior knowledge on input images and output labels. The novelty lies in the automatically modelization of the output dependencies through a learning machine, whereas it usually relies on a static model like an atlas in medical applications.

As a feedforward neural network, IODA has a static architecture which implies that the input and output sizes cannot vary from an example to another. Therefore, IODA cannot be considered as a dynamic method: the processing of variable input size problems would require an image resampling preprocessing stage. Moreover, as the efficiency of IODA relies on embedding the output space, it is designed for dataset where outputs are correlated.

From a computational point of view, our approach does not require a huge amount of resources, that makes it affordable for standard desktop computers. Nevertheless, as a lot of parameters are tuned during learning, a significant amount of memory is needed (3GB for our medical application).

Unlike other 2D-approach (CRF or HMM), our neural-based approach does not require a time consuming and suboptimal decoding process as the decision is performed using a light forward propagation through the network. Another advantage is that high order output dependencies can be modeled, while 2D approaches are generally limited to the first order dependency due to computational complexity. Indeed, IODA allows each label to depend on all other labels from the image.

From an applicative point of view, IODA could be applied on other image labeling problems. One can expect significant improvements on problems where dependencies between the output labels can be observed. This condition is often verified in medical imaging, or by instance in document image structure analysis [37, 38] or natural scene processing such as road sign detection [39, 40].

7. Acknowledgement

This work has been partly supported by the ANR-11-JS02-010 project LeMon.

References

- [1] H. Wang, J. Suh, S. Das, J. Pluta, C. Craige, P. Yushkevich, Multi-atlas segmentation with joint label fusion, *Pattern Analysis and Machine Intelligence*, IEEE Transactions on 35 (3) (2013) 611–623.

- [2] B. Zitov, J. Flusser, Image registration methods: a survey, *Image and Vision Computing* 21 (11) (2003) 977 – 1000.
- [3] J. B. A. Maintz, M. A. Viergever, A survey of medical image registration, *Medical Image Analysis* 2 (1) (1998) 1–36.
- [4] A. Goshtasby, Piecewise linear mapping functions for image registration, *Pattern Recognition* 19 (6) (1986) 459 – 466.
- [5] H. Chung, D. Cobzas, L. Birdsell, J. Lieffers, V. Baracos, Automated segmentation of muscle and adipose tissue on ct images for human body composition analysis, *Proc. SPIE* 7261 (2009) 72610K–72610K–8.
- [6] O. Chum, A. Zisserman, An Exemplar Model for Learning Object Classes, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2007) 1–8.
- [7] V. Ferrari, L. Fevrier, F. Jurie, C. Schmid, Groups of Adjacent Contour Segments for Object Detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30 (1) (2008) 36–51.
- [8] G. Csurka, F. Perronnin, An Efficient Approach to Semantic Segmentation, *International Journal of Computer Vision* 95 (2) (2011) 198–212.
- [9] P. Kohli, L. Ladický, P. H. Torr, Robust Higher Order Potentials for Enforcing Label Consistency, *International Journal of Computer Vision* 82 (3) (2009) 302–324.
- [10] L. R. Rabiner, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, in: *Readings in Speech Recognition*, Kaufmann, 1990, pp. 267–296.
- [11] J. Lafferty, A. McCallum, F. Pereira, Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data, *ICML* (2001) 282–289.
- [12] S. Nicolas, T. Paquet, L. Heutte, A Markovian Approach for Handwritten Document Segmentation, *ICPR* 3 (2006) 292–295.
- [13] S. Z. Li, *Markov Random Field Modeling in Image Analysis*, 3rd Edition, Springer Publishing Company, Incorporated, 2009.
- [14] A. Ion, J. Carreira, C. Sminchisescu, Probabilistic Joint Image Segmentation and Labeling by Figure–Ground Composition, *International Journal of Computer Vision* 107 (1) (2014) 40–57.
- [15] P. Krähenbühl, V. Koltun, Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials, *NIPS* (2011) 109–117.
- [16] J. Besag, On the statistical analysis of dirty pictures, *Journal of the Royal Society series B*, vol 48 (1986) 259–302.

- [17] P. B. Chou, B. C. M., The theory and practice of Bayesian image labeling, *International Journal of Computer Vision* 4 (1990) 185–210.
- [18] M. B. Blaschko, C. H. Lampert, Learning to Localize Objects with Structured Output Regression, *ECCV* (2008) 2–15.
- [19] J. Weston, O. Chapelle, A. Elisseeff, B. Scholkopf, V. Vapnik, Kernel dependency estimation, *NIPS* (2002) 873–880.
- [20] G. E. Hinton, S. Osindero, Y.-W. Teh, A Fast Learning Algorithm for Deep Belief Nets, *Neural Computation* 18 (7) (2006) 1527–1554.
- [21] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle, Greedy Layer-Wise Training of Deep Networks, *NIPS* (2007) 153–160.
- [22] J. Weston, F. Ratle, R. Collobert, Deep learning via semi-supervised embedding, *ICML* (2008) 1168–1175.
- [23] N. Srivastava, Improving Neural Networks with Dropout, Master’s thesis, University of Toronto, Toronto, Canada (January 2013).
- [24] R. Sarikaya, G. E. Hinton, A. Deoras, Application of deep belief networks for natural language understanding, *IEEE/ACM Transactions on Audio, Speech & Language Processing* 22 (4) (2014) 778–784.
- [25] L. Deng, G. E. Hinton, B. Kingsbury, New types of deep neural network learning for speech recognition and related applications: an overview, *ICASSP* (2013) 8599–8603.
- [26] A. Graves, M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke, J. Schmidhuber, A novel connectionist system for unconstrained handwriting recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31 (5) (2009) 855–868.
- [27] D. E. Rumelhart, G. E. Hinton, R. J. Williams, Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1, MIT Press, Cambridge, MA, USA, 1986, Ch. Learning Internal Representations by Error Propagation, pp. 318–362.
- [28] D. C. Ciresan, U. Meier, L. M. Gambardella, J. Schmidhuber, Deep big simple neural nets excel on handwritten digit recognition, *Neural computation* 22 (12) (2010) 3207–3220.
- [29] B. Labbe, R. Herault, C. Chatelain, Learning Deep Neural Networks for High Dimensional Output Problems, in: *ICMLA*, Miami, USA, 2009, p. 6p.
- [30] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion, *Journal of Machine Learning Research* 11 (2010) 3371–3408.

- [31] Y. Bengio, A. Courville, P. Vincent, Representation Learning: A Review and New Perspectives, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35 (8) (2013) 1798–1828.
- [32] H. Bourlard, Y. Kamp, Auto-association by multilayer perceptrons and singular value decomposition, *Biological Cybernetics* 59 (4-5) (1988) 291–294.
- [33] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley, Y. Bengio, Theano: a CPU and GPU Math Expression Compiler, *Python for Scientific Computing Conference (SciPy)*.
- [34] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: *In Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS10)*. Society for Artificial Intelligence and Statistics, 2010.
- [35] L. Martin, L. Birdsell, N. MacDonald, T. Reiman, M. T. Clandinin, L. J. McCargar, R. Murphy, S. Ghosh, M. B. Sawyer, V. E. Baracos, Cancer Cachexia in the Age of Obesity: Skeletal Muscle Depletion Is a Powerful Prognostic Factor, Independent of Body Mass Index, *Journal of Clinical Oncology* 31 (12) (2013) 1539–1547.
- [36] H. Lanic, J. Kraut-Tauzia, R. Modzelewski, F. Clatot, S. Mareschal, J.-M. Picquenot, A. Stamatoullas, S. Leprêtre, H. Tilly, F. Jardin, Sarcopenia is an Independent Prognostic Factor in Elderly Patients with Diffuse Large B-Cell Lymphoma Treated with Immunochemotherapy, *Leukemia and Lymphoma* (2013) Epub ahead of a print.
- [37] V. Papavassiliou, T. Stafylakis, V. Katsouros, G. Carayannis, Handwritten document image segmentation into text lines and words, *Pattern Recognition* 43 (1) (2010) 369–377.
- [38] P. Barlas, S. Adam, C. Chatelain, T. Paquet, A typed and handwritten text block segmentation system for heterogeneous and complex documents, *Document Analysis Systems* (2014) 46–50.
- [39] S. M. Bascón, J. A. Rodríguez, S. L. Arroyo, A. F. Caballero, F. López-Ferreras, An optimization on pictogram identification for the road-sign recognition task using SVMs, *Computer Vision and Image Understanding* 114 (3) (2010) 373–383.
- [40] A. Ruta, Y. Li, X. Liu, Real-time traffic sign recognition from video by class-specific discriminative features, *Pattern Recognition* 43 (1) (2010) 416–430.

A.2 Spotting L3 slice in CT scans using deep convolutional network and transfer learning

Reference

[Bel+17] Soufiane Belharbi et al. "Spotting L3 Slice in CT Scans Using Deep Convolutional Network and Transfer Learning." In: *Computers in Biology and Medicine* 87 (Aug. 1, 2017), pp. 95–103. issn: 0010-4825. doi: 10.1016/j.combiomed.2017.05.018. URL: <http://www.sciencedirect.com/science/article/pii/S0010482517301403>

Spotting L3 slice in CT scans using deep convolutional network and transfer learning

Soufiane Belharbi^a, Clément Chatelain^{a,d}, Romain Hérault^{a,d}, Sébastien Adam^{a,*}, Sébastien Thureau^{c,a}, Mathieu Chastan^b, Romain Modzelewski^{a,b}

^a*Normandie Univ, UNIROUEN, UNIHAVRE, INSA Rouen, LITIS, 76000 Rouen, France*

^b*Henri Becquerel center, Department of Nuclear Medicine, 76000 Rouen, France.*

^c*Henri Becquerel center, Department of Radiotherapy, 76000 Rouen, France.*

^d*These authors contributed equally*

Abstract

In this article, we present a complete automated system for spotting a particular slice in a complete 3D Computed Tomography exam (CT scan). Our approach does not require any assumptions on which part of the patient's body is covered by the scan. It relies on an original machine learning regression approach. Our models are learned using the transfer learning trick by exploiting deep architectures that have been pre-trained on imageNet database, and therefore it requires very little annotation for its training. The whole pipeline consists of three steps : i) conversion of the CT scans into Maximum Intensity Projection (MIP) images, ii) prediction from a Convolutional Neural Network (CNN) applied in a sliding window fashion over the MIP image, and iii) robust analysis of the prediction sequence to predict the height of the desired slice within the whole CT scan. Our approach is applied to the detection of the third lumbar vertebra (L3) slice that has been found to be representative to the whole body composition. Our system is evaluated on a database collected in our clinical center, containing 642 CT scans from different patients. We obtained an average localization error of 1.91 ± 2.69 slices (less than 5 mm) in an average time of less than 2.5 seconds/CT scan, allowing integration of the proposed system into daily clinical routines.

*Corresponding author

Email address: Sebastien.Adam@univ-rouen.fr (Sébastien Adam)

Keywords: Convolutional neural networks, deep learning, slice detection, maximum intensity projection, sarcopenia

1. Introduction

In recent years, there has been an increasing interest in the analysis of body composition for estimating patient outcomes in many pathologies. For instance, sarcopenia (loss of muscle), visceral and subcutaneous obesity are known prognostic factors in cancers [MBM⁺13, YDM⁺15], cardiovascular diseases [AWM⁺14] and surgical procedures [PVT⁺11, KOF⁺13]. Body composition can also be used to improve individual nutritional care and chemotherapy dose calculation [GLC⁺13, LKTM⁺14]. It is usually assessed by CT and Magnetic Resonance Imaging (MRI). Moreover, It has been shown that the composition of the third lumbar vertebra (L3) slice is a good estimator of the whole body measurements [MBH⁺98, SPW⁺04]. To assess the patient's body composition, radiologists usually have to manually find the corresponding L3 slice in the whole CT exam (spotting step, see Figure 1), and then to segment the fat and muscle on a dedicated software platform (segmentation step). These two operations take more than 5 minutes for an experienced radiologist and are prone to errors. Therefore, there is a need for automating these two tasks.

The segmentation step has been extensively addressed in the literature among the medical imaging community [PXP00, MT96]. Dedicated approaches for L3 slice have been proposed such as atlas based methods [CCB⁺09] or deep learning [LHC⁺15]. On the other hand, to the best of our knowledge, the automatic spotting of a specific slice within the whole CT scan has not been investigated in the literature. The spotting task is particularly challenging since it has to handle:

- The intrinsic variability in the patient's anatomy (genders, ages, morphologies or medical states).

- The various acquisition/reconstruction protocols (low/high X-rays dose, slice thickness, reconstruction filtering, enhanced/non enhanced contrast agent).
- The arbitrary field-of-view scans, displaying various anatomical regions.
- The strong similarities between the L3 slice and other slices, due to the repetitive nature of vertebrae (Fig.2).

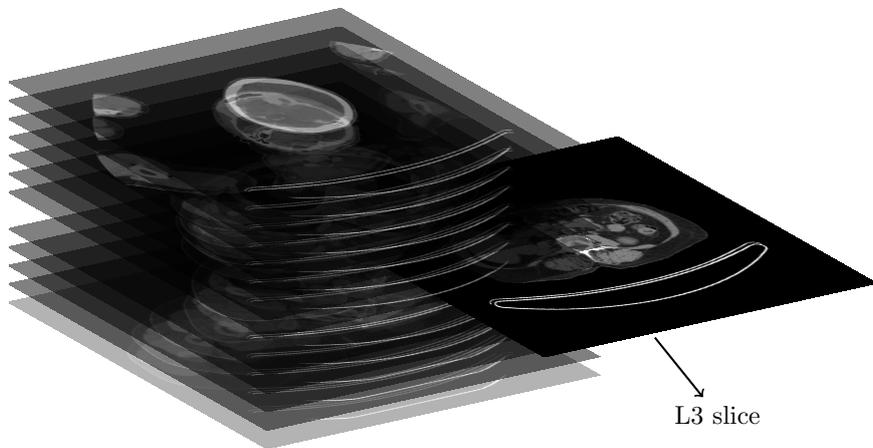


Figure 1: Finding the L3 slice within a whole CT scan.

In the literature, spotting tasks are often achieved using ad hoc approaches such as registration which are not suitable for high variability problems [GZH14, CWJ⁺15]. In particular, a 3D registration on a whole CT scan would require a large amount of computation at decision time [SEM16]. Here, we suggest a more generic strategy based on machine learning in order to handle high variability context, while maintaining a fast decision process.

In this work, spotting a slice within a CT scan is tackled as a regression problem, where we try to estimate the slice position height. An efficient processing flow is proposed, including a Convolutional Neural Network (CNN) learned using transfer learning. Our approach tackles the classical issues faced in medical image analysis: the data representation issue is addressed using Maximum In-



Figure 2: Two slices from the same patient: a L3 (up) and a non L3 (L2) (down). The similar shapes of both vertebrae prevent from taking a robust decision given a single slice.

tensity Projection (MIP); the variability of the shapes in CT scans is handled using a CNN; and the lack of annotated data is circumvented using transfer learning.

The article is organized as follows: section 2 presents the related work and the general framework for applying machine learning for L3 detection in a CT scan. The third section presents the proposed approach and describes each stage of the whole processing flow. The fourth section describes the experiments and the obtained results.

2. Related Work

Machine learning approaches provide generic and flexible systems, provided enough annotated data is available. From a machine learning perspective, the localization of the L3 slice given a whole CT scan can either be considered as a slice-classification problem, a sequence labeling problem or a regression problem. Let us now consider these three options.

The classification paradigm consists of deciding for each slice of the whole CT scan whether the L3 vertebra is present or not. However, the repetitive nature of individual vertebra induces a similarity between the L3 slice and its neighbors, which prevents to efficiently classify an isolated slice without any context (see Fig. 2). This explains why even experienced radiologists need to browse the CT scan to infer the relative position and precisely identify the L3 slice. To the best of our knowledge, the classification paradigm has not been used in the literature to detect the L3 slice within a whole CT scan.

The sequence labeling paradigm consists of estimating the label (L1, L2, etc.) of every slice of a complete CT scan, then, choose the one that is more likely to correspond to the L3. The advantage of this approach is that the decision is globally taken on the whole CT scan by analyzing the dependencies between the slices. This kind of approach has been recently investigated for labeling the vertebrae of complete spine images [GACD11, GVC09, MWK⁺13, GDE⁺13, KLP11, GFC⁺12, HCLN09, ML13, OA11]. The dependencies are modeled using graphical models, such as Hidden Markov Models (HMMs) [GFC⁺12] or Markov Random Fields (MRFs) [KLP11]. A full review of the spine labelization methods can be found in [MHSB13]. The major drawback of sequence labeling approaches is that they require a fully annotated learning database where every slice of the CT scan is labeled, which is very time consuming. Such a dataset is

proposed by [GZH14], but this dataset cannot be easily exploited for our problem since i) the data are cropped images of the whole spine, and ii) it contains only 224 CT scan.

The regression problem consists of directly estimating the L3 slice number given the whole CT scan, in a spotting fashion. Like the previous paradigm, it has the advantage of performing a global decision by taking into account the dependencies within the entire exam. Another major advantage of a spotting approach is that it does not require a full labeling of the exams. Indeed, the only annotation needed for learning such a model is the L3 position within the whole exam. For radiologists, this annotation is more lightweight than a full annotation and may lead to creating large datasets easily.

In this work, we retain the third paradigm and propose a machine learning approach for spotting the L3 slice in heterogeneous arbitrary field-of-view CT scans. To the best of our knowledge, this is the first time that slice spotting is addressed as a machine learning regression problem.

Usually, traditional machine learning methods exploit generic hand-designed features which are fed to a learning model with the assumption that they are suitable for describing the image. To achieve high accuracy, usually one ends up combining many types of features which require extensive computation, more time and large memory size. Ideally, it would be better if the model is capable of learning on its own task-dependent features.

Deep neural networks (DNN) are a specific category of models in machine learning which are capable of learning on their own hierarchical features based on the raw image. Convolutional neural networks (CNN) are a particular type of DNN which gained a large reputation in computer vision due to their high performance for many tasks on natural scene images [STE13, ESTA14, RHGS15, KSH12].

In the last years, the use of machine learning, in general, and using CNN, in particular, has grown in various medical domains such as cancer diagnosis [RYL⁺14, UBHK14], segmentation [HJ13, HDW⁺15, Lai15] or histological [MMB⁺08] and drusen identification [CC06]. In all these works, the authors are faced with a common issue which is the lack of annotated data. Although extremely powerful, CNN architectures require a huge amount of data to avoid the “learning by heart” phenomenon, also known as overfitting in machine learning. The classical techniques to limit these issues are dropout, data augmentation or the use of regularization. All these technical tricks are exploited in [Lai15], but the lack of data is still a limitation to train such large models. Recently, a more efficient way has been proposed to circumvent the lack of annotated data in vision. This method consists of exploiting models that have been pre-trained on a huge amount of annotated data on another task and is known as “transfer learning”.

In this work, we explore the idea of using a CNN model for the localization of the L3 slice using transfer learning. A full description of our approach is presented in section 3.

3. Proposed approach

Using a CNN for solving the L3 detection task formulated as a regression problem (see fig. 1) is not straightforward, and requires the alleviation of some constraints which are inherent to the medical domain and to the data that is being processed (i) Training a CNN on 3D data such as CT scans requires very large computing and memory resources that can even exceed the memory limit of most accelerator cards, while such cards are essential for learning a CNN in a reasonable time; (ii) Training a CNN requires fixed size inputs, while the size of the CT scans can vary from one exam to another because of an arbitrary field of view; (iii) Training a CNN requires a large amount of labeled data.

In this paper, we propose to overcome these limitations by using the approach depicted in figure 3. In this approach, the CT scan is first converted into another representation using Maximum Intensity Projection (MIP), in order to reduce the dimension of the input from 3D to 2D, without loss of important information. Then, the MIP image is processed in a sliding window fashion to be fed to a CNN with a fixed-size input. This CNN is trained with Transfer Learning (TL-CNN) to solve the requirement of a large amount of labeled examples. Once the trained TL-CNN has computed its prediction for each position of a sliding window, the resulting prediction sequence is processed in order to estimate the final L3 position in the full CT scan. The following subsections detail the three important contributions of the proposed system.

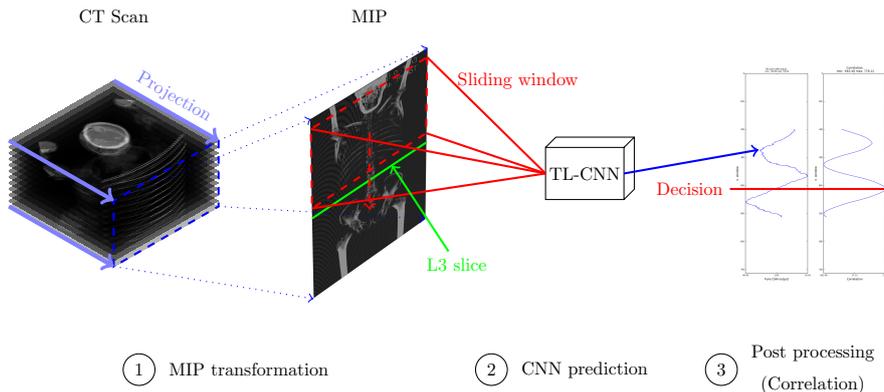


Figure 3: System overview describing the three important stage of our approach : MIP transformation, TL-CNN prediction, and post processing.

3.1. MIP transformation

Ideally, one can use the raw 3D scan image to feed the CNN. If N is the number of slices of the arbitrary field of view CT scan, the input size is $512^2 \times N$. For example, a CT-scan with 1000 slices represents 262M inputs. However, the input size of CNN models strongly impacts their number of parameters. Therefore it would require a very large number of training samples to efficiently

learn the CNN. Thus, in the case of few training samples, using the 3D scan directly as an input is not efficient. We believe that the patient’s skeleton carries enough visual information in order to detect the L3.

For these reasons, we propose to use a different data representation which focuses on the patient’s skeleton and dramatically reduces the size of the input space. This representation is based on a frontal Maximum Intensity Projection (MIP) [WMLK89, WM91, Wal92]. The idea is to project a line from a frontal view of the CT scan and retain the maximum intensity over all the voxels that fall into that line. We experimented using different views such as frontal and lateral views, as well as their combination but they did not work well as compared to the frontal view alone.

Since the slice thickness can vary within the same scan and the voxels are not squared, the projection often generates a distorted MIP. Visually, this gives an unrealistic image where the skeleton is shrunk or enlarged. The cause of this distortion is that, often, the resulting pixel from the projection does not correspond to one voxel. Often, one voxel can be represented by more than one pixel. In order to obtain an equal correspondence (i.e. one pixel corresponds to one voxel), we resize (normalize) the 2D MIP image using an estimated ratio r and average slice thickness s where r represents the number of pixels corresponding to one voxel (slice).

Fig.4 shows an example of a normalized frontal MIP image. The MIP transformation reduces the input size from $512^2 \times N$ to $512 \times N$.

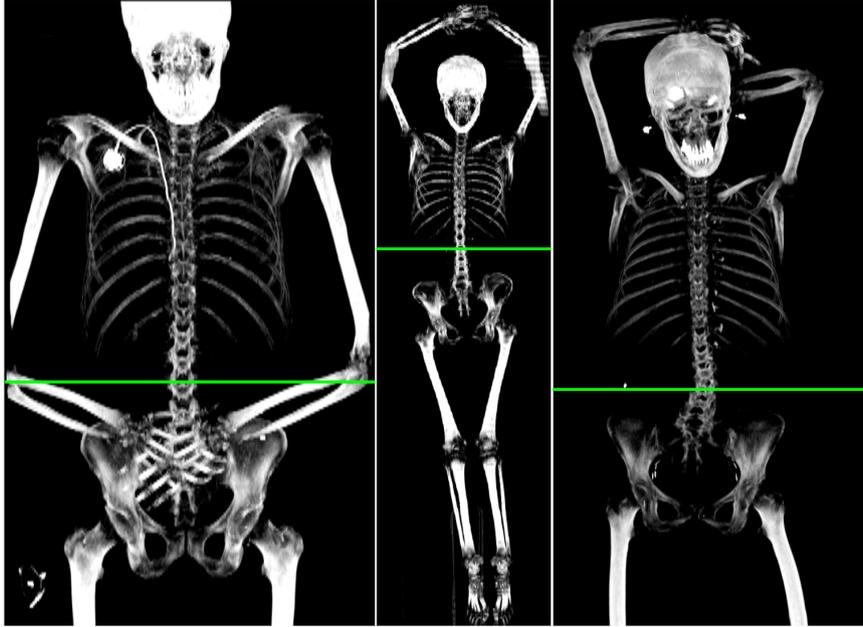


Figure 4: Examples of normalized frontal MIP images with the L3 slice position.

3.2. Learning the TL-CNN

Convolutional neural networks (CNN) are particular architecture of neural networks. Their main building block is a convolution layer that performs a non-linear filtering operation. This convolution can be viewed as a feature extractor applied identically over a plane. The values of the convolution kernel constitute the layer parameters. Several convolution layers can be stacked to extract hierarchical features, where each layer builds a set of features from the previous layer. After the convolutional layers, fully connected layers can be stacked to perform the adequate task such as the classification or the regression.

In the learning phase, both parameters of convolutional layers and fully connected layers are optimized according to a loss function. The optimization of these huge number of parameters is generally performed using stochastic gradient descent method. This process requires a very large number of training samples.

Recently, there has been a growing interest in the exploration of transfer learning methods to overcome the lack of training data. Transfer learning consists in adapting models, trained for different task, to the task in hand (target). It has been applied with success for various applications such as character recognition [Jia15, CMS12], signature identification [HSO16] or medical imaging [BDWG15, SRG⁺16]. All these contributions exploit CNN architectures which have been pre-trained on computer vision problems, where huge labeled datasets exist. In this framework, the weights of the convolutional layers are initialized with the weights of a pre-trained CNN on another dataset, and then fine-tuned to fit the target application. The fine-tuning starts by transferring only the weights of the convolutional layers from a pre-trained network to the target network. Then, randomly initialized fully connected layers are stacked over the pre-trained convolutional layers and the optimization process is performed on the whole network. This transfer learning framework carried out for our application is illustrated by Figure 5 .

A well-known difficulty when using the transfer learning paradigm is to fit the data to the input size of the pre-trained architecture. Since the size of the normalized MIP images varies from one patient to another, two solutions can be considered. The first one consists of resizing the whole scan to a given fixed size. This solution is straightforward but it dramatically impacts the image quality and the output precision. The second solution consists in decomposing the input MIP into a set of fixed-size windows with a sampling strategy. In this paper, we adopt the second approach which enables to preserve the initial quality of the image data.

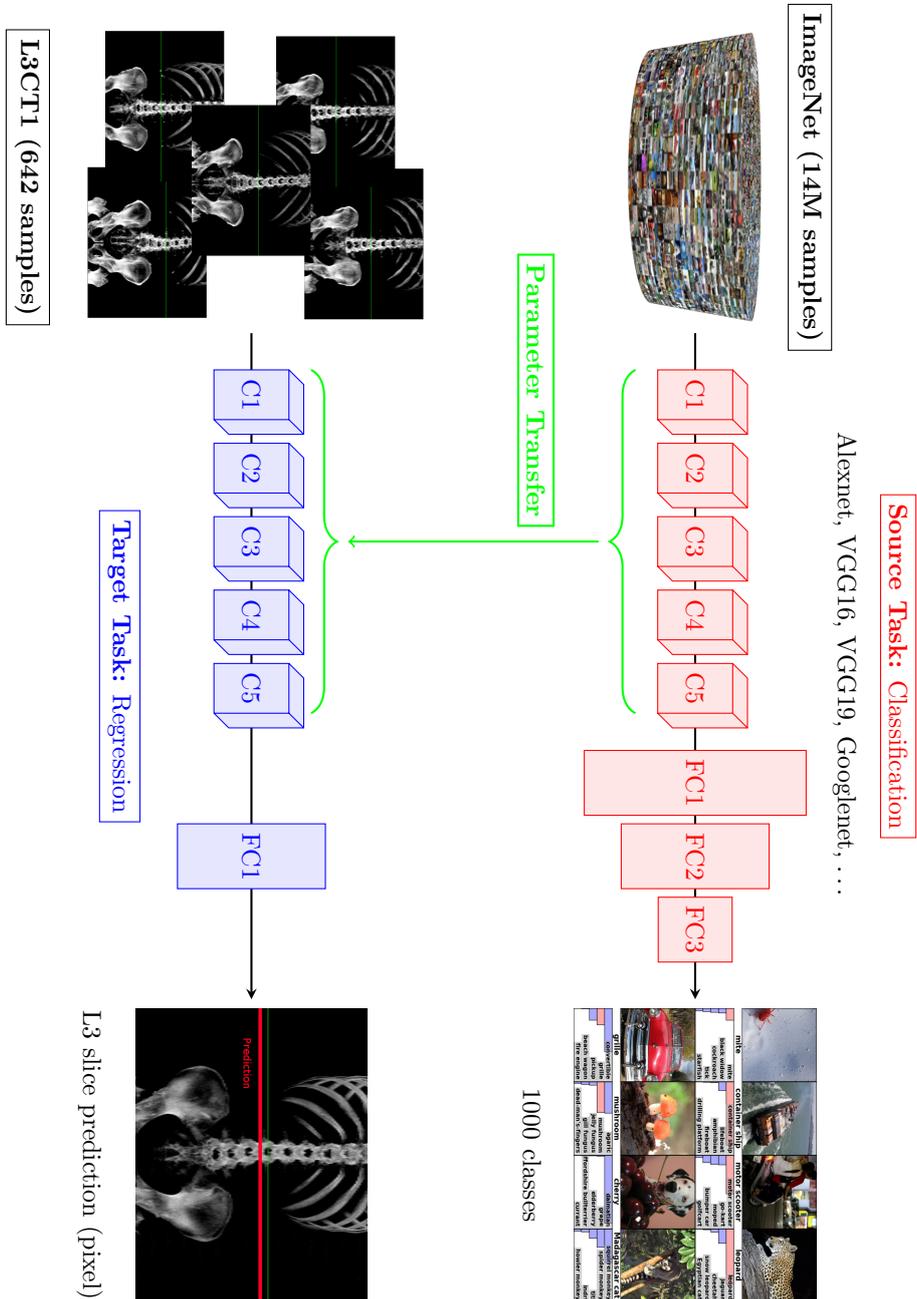


Figure 5: System overview. Layers C_i are Convolutional layers, while FC_i denote Full Connected layers. Convolution parameters of previously learnt ImageNet classifier are used as initial values of corresponding L3 regressor layers to overcome the lack of CT examples.

When sampling windows from the MIP image, two sets of window images can be produced. The first one is made of windows containing the L3, and the other one is made of windows without the L3. This raises the question whether the windows without L3 should be present or not in the CNN learning dataset. As we propose a regression approach, adding the non-L3 images in the learning dataset would imply that the CNN learns (and outputs in the decision stage) the offset of the L3 with respect to the current window. Obviously, this offset can be very difficult to learn, particularly if the current window is far from the L3 position. Thus, we have decided to include only the windows containing the L3 in the learning dataset.

Thus, for building the training dataset, we sample all the possible windows of height H such that the L3 position is in the support $[-a, +a]$ where 0 denotes the center of the window. This leads to $2a + 1$ possible windows from each MIP image to be included in the training set. All windows from all MIP are then shuffled: it is highly improbable that two neighboring windows from the same MIP will appear next to each other in the optimization procedure.

3.3. Decision process using a sliding window over the MIP images

A sliding window procedure is applied at the decision phase on the entire MIP image, leading to a sequence of relative L3 position predictions. Such a sequence is illustrated in the left of figure 6.

In this sequence, one can observe two distinct behaviors depending on the presence of the L3 in the corresponding window: i) If the L3 is not in the window, the CNN tends to output random values since it has been trained only on images containing L3. This behavior is illustrated in Figure 6 at the beginning and (less clearly) at the end of the sequence. ii) If the L3 is within the window, the CNN is expected to predict (correctly) the relative L3 position within the window. Since the L3 position is fixed in the MIP and the window slides line by line on the region of interest, the true relative L3 position should decrease one by one. In consequence, the CNN output should evolve linearly along the sequence of windows, leading to a noisy straight line with a slope of

–1. The noise may come from local imprecision or error on an individual slide. This behavior can be observed in figure 6 between offset 500 and 600, and it is highlighted with a theoretical green line.

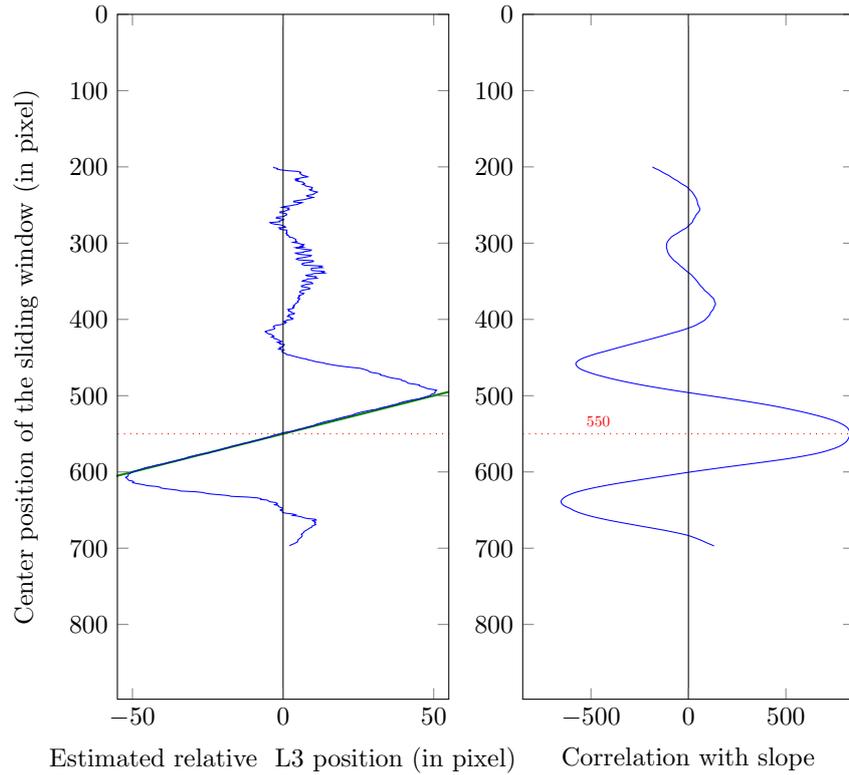


Figure 6: [left]: CNN output sequence obtained for $H = 400$ and $a = 50$ on a test CT scan. The sequence contains the typical straight line of slope -1 centered on the L3 (the theoretical line is plotted in green), surrounded by random values. [right]: correlation between the CNN output sequence and the theoretical slope. We retain the maximum of correlation as an estimation of the L3 position.

Therefore, at decision stage, the L3 position can be estimated through the localization of the middle of this particular straight segment. This estimation can easily be achieved by searching the maximum of a simple correlation between

the sequence and the expected slope. This procedure, illustrated at the bottom of Fig. 6, easily filters out boundary windows which do not contain the L3, and shows robustness by averaging several predictions of the CNN.

4. Experimental protocol

4.1. CT exams database description

In order to validate the proposed approach, a database named L3CT1 has been collected¹. The main part of the dataset is composed of 642 CT exams from different patients. All patients were included in this study after being informed of the possible use of their images in a retrospective research. The institutional ethical board of the Rouen Henri Becquerel Center approved this study². The CT exams show a high heterogeneity of patients in terms of anatomy, sex, cancer pathologies, position and properties of the reconstructed CT images: 4 scanner models (PET/CT modalities) and 2 manufacturer, acquisition protocols (low dose acquisition (100 to 120 kV) and modulated mAs along the body) axial field of view (FOV) (400 to 500 mm), reconstruction algorithms (Filtered Back Projection (FBP) or iterative reconstruction) and slice thickness (2 to 5 mm).

On each CT scan, the L3 slice was located by an expert radiologist on a dedicated software [LKT⁺14], providing the annotation for the position of the L3 through its distance in (mm) from the first slice in the scan (top).

Moreover, 43 supplementary CT scans have been annotated by the same radiologist and 3 other experts, in order to evaluate the variability of annotations among experts.

To be as reproducible and precise as possible, detailed guidelines were given to all radiologists for annotation.

¹This dataset is available on demand, please contact the corresponding author

²IRB Number 1604B.

From all the scans, frontal MIP images have been computed using the process described in 3.1. This results in a set of 642 images of constant width (512 pixels) and variable height, varying from 659 to 1862 pixels. Fig 4 shows some examples of frontal MIP images extracted from three patients of the L3CT1 database.

4.2. Datasets preparation

The first step consists in splitting the dataset into 5 folds, in order to allow a cross-validation procedure. The split is applied at the patient level, in order to prevent that a given CT-scan provides windows in different sets (learning, validation, test), what should lead to biased results. Moreover, due to variable slice thickness in the dataset, we make sure when dividing the dataset to obtain stratified folds. Thus, we end up with the same number of samples from each slice thickness in each set.

Once the MIP images folds have been generated, learning, validation and test windows are sampled as explained in section 3.3, where the value of a has been experimentally set to $a = 50$ using a cross validation procedure. For the validation set, in order to speed up the training, we take only 300 random windows from different patients.

4.3. Neural networks models

In order to conduct our experiments, two types of convolutional neural networks have been compared:

- **Homemade CNN (CNN4):** We have designed and trained a CNN from scratch, with specific architecture of four convolutional layers followed by a fully connected output layer. In each convolution layer, a horizontal max-pooling is performed. We found in practice that vertical max-pooling distorts the target position. The number of kernels that we used in the four convolution layers are [10, 3, 3, 5], with respective sizes [5, 7, 9, 3]. The hyper-parameters of our CNN were tuned on the validation set [Ben12]. We refer to our model as *CNN4*.

- **Pre-trained CNNs:** In our study, we have collected a set of pre-trained convolutional neural networks over ImageNet dataset [DDS⁺09]: Alexnet [KSH12], VGG16 [SZ14], VGG19[SZ14], Googlenet (Inception V1) [SLJ⁺14]³. The models are created using the library Keras [Cho15]. For each model, we keep only the convolutional layers which are considered as shared perception layers that may be used for different tasks. On top of that, we add one fully connected layer to be specialized in our specific task (i.e. L3 detection). Our experiments have shown that adding more fully connected layers does not improve the results.

The input of pre-trained models is supposed to be an RGB image (i.e. a 3D matrix), while in the other hand, our sampled windows are 2D matrix. In order to match the required input, we duplicate the 2D matrix in each color channel. Then, each channel is normalized using its mean from the ImageNet Dataset.

We use L_2 regularization for training all the models with value of $\lambda = 10^{-3}$, except for Googlenet where we used the original regularization values.

5. Results

5.1. Data view: Frontal Vs. Lateral

The use of the MIP representation allows us to access to different views of the CT scan, such as the frontal and lateral views (other views with different angles are possible). In order to choose the best view, we re-train a VGG16 model with one fully connected layer using different input views. We recall that the input of the VGG16 is an image with 3 plans. We experimented three configurations. In the first and second cases, we repeat the frontal and lateral views, respectively, in the three input channels. In the last case, we mixed

³The weights of Googlenet were obtained from: <https://gist.github.com/joelouismarino/a2ede9ab3928f999575423b9887abd14>, and the weights of the rest of the models were obtained from <https://github.com/heuritech/convnets-keras>

the frontal and the lateral view. The motivation behind the combination of the views is that each view will provide an additional information (hopefully complementary) that will help the model to decide. The sampling margin of the windows is done over the range $[-50, +50]$. Tab.1 shows that using frontal view alone is more suitable. One possible explanation of this results is that the frontal view contains more structural context (ribs, pelvis) which helps to locate the L3 slice, in the opposite of the lateral view. Combining lateral and frontal views gave better results than lateral alone but worse than frontal alone. One may think that lateral view adds noise to the frontal view.

View	VGG16
	Error m_c (slices)
Frontal	1.71 ± 1.59
Lateral	4.29 ± 14.90
Frontal Lateral Frontal	1.89 ± 2.05

Table 1: Test error (mean \pm standard deviation) over the test set of fold 0, expressed in slices, using VGG16 model with frontal and lateral views.

5.2. Detection performance

All the models described in section 4.3 have been evaluated in a cross validation procedure on the L3CT1 dataset by computing the prediction error. The prediction error for one CT scan is computed as the absolute difference between the prediction y_{pred} and the target y : $e = |y - y_{pred}|$. The error is expressed in slices. We report the mean and the standard deviation of the test error (μ_e, σ_e) , respectively in the form $\mu_e \pm \sigma_e$, over the entire test set. Obtained results are reported in Tab.2.

For the sake of comparison, we used Random Forest Regression (RF) [Bre01, Ho95] as a regressor instead of our CNN. As in most pattern recognition problems, we need to extract input features to train our Random Forest Regression. Local Binary Patterns (LBP) features have shown to be very efficient in many computer vision tasks [OPM02], especially in medical imaging [NLB10]. Therefore, we have retained this feature descriptor. To extract the LBP features we

	RF500	CNN4	Alexnet	VGG16	VGG19	Googlenet
fold 0	7.31 ± 6.52	2.85 ± 2.37	2.21 ± 2.11	2.06 ± 4.39	1.89 ± 1.77	1.81 ± 1.74
fold 1	11.07 ± 11.42	3.12 ± 2.90	2.44 ± 2.41	1.78 ± 2.09	1.96 ± 2.10	3.84 ± 12.86
fold 2	13.10 ± 13.90	3.12 ± 3.20	2.47 ± 2.38	1.54 ± 1.54	1.65 ± 1.73	2.62 ± 2.52
fold 3	12.03 ± 14.34	2.98 ± 2.38	2.42 ± 2.23	1.96 ± 1.62	1.76 ± 1.75	2.22 ± 1.79
fold 4	8.99 ± 7.83	1.87 ± 1.58	2.69 ± 2.41	1.74 ± 1.96	1.90 ± 1.83	2.20 ± 2.20
Average	10.50 ± 10.80	2.78 ± 2.48	2.45 ± 2.42	1.82 ± 2.32	1.83 ± 1.83	2.54 ± 4.22

Table 2: Error expressed in slice over all the folds using different models: RF500, CNN4 (Homemade model), and Alexnet/VGG16/VGG19/GoogleNet (Pre-trained models).

used a number of neighbors of 8 and a radius of 3 which creates an input feature vector with dimension of $2^8 = 256$. From each sampled window, we extract LBP features. We investigated different number of trees: 10, 100 and 500. The obtained results showed that random forests do not perform well over this task. We report in Tab.2 the results using 500 (RF500) trees which are in the same order of performance compared the other cases (i.e. 10 and 100 trees).

From Tab.2, one can see that pre-trained models perform better than our homemade CNN4 with an improvement of about 35%. In particular, VGG16 showed the best results by an average error of 1.82 ± 2.32 followed by VGG19 with 1.83 ± 1.83 . This result confirms the strong benefit of transfer learning between two different tasks. Moreover, it shows that the convolutional layers can be shared as a perception tool between different tasks with slight adaptation. On the other hand, this illustrates the capability for modeling such task using the pre-trained models.

5.3. Processing time issues

One must mention that the price we paid in order to reach the performance mentioned above is to increase the complexity of the model. In Table 3, we present the number of parameters of each model and the average required time for the prediction of the L3 slice. We observe that VGG16 contains approximately 264 times more parameters than CNN4. Beside the required memory

for such models, the real paid cost is the evaluation time during the test phase. Computed on a GPU (Tesla K40), VGG16 requires an average of 13.28 seconds per CT scan while our CNN4 only needs 4.46 second per CT scan.

	Number of parameters	Average forward pass time (seconds/CT scan)
CNN4	55,806	04.46
Alexnet	2,343,297	06.37
VGG16	14,739,777	13.28
VGG19	20,049,473	16.02
Googlenet	6,112,051	17.75

Table 3: Number of parameters for different models and average forward pass time per CT scan.

An important factor which affects the evaluation time in these experiments is the number of windows processed by the CNN for a given CT scan. Thus, it is possible to dramatically reduce the computation time by shifting the window by a bigger value than 1 pixel. An experimental evaluation of this strategy with VGG16 has shown that a good compromise between processing time and performance could be obtained for a shift value up to 6 pixels without affecting the localization precision. This sub-sampling reduces the evaluation time from 13.28 seconds/CT scan to 2.36 seconds/CT scan and moved the average localization error from 1.82 ± 2.32 slices to 1.91 ± 2.69 slices, respectively. This shows the robustness of the proposed correlation post-processing.

5.4. Comparison with radiologists

In order to further assess the performance of the proposed approach, an extra set of 43 CT scans was used for test. This particular dataset was annotated by the same radiologist who annotated L3CT1 dataset and also by three other experts. Each annotation was performed at two different times, in order to evaluate the intra-annotator variability. We refer to both annotations by the same expert by *Review 1* and *Review 2*.

Obtained results are illustrated in Tab.4. It compares the error made by CNN models with those made by the radiologists, using the radiologist who annotated the L3CT1 dataset as reference. These results corroborate the results

provided in Table 2 since VGG16 is better than CNN4 with an improvement of about 35% in average for both reviews. The results also demonstrate that radiologists are in average more precise than automatic models with an improvement of about 50%. However, they also show that there exists some variabilities among radiologist annotations and even an intra-annotator variability. This latter is visible in Tab. 4 since computed errors for automatic systems vary between both reviews while the automatic system gives the same output, showing that reference values have changed. This illustrates the difficulty of the task of precisely locating the L3 slice and the interest of CNN which does not change its prediction.

Errors (slices) / operator	CNN4	VGG16	Radiologist #1	Radiologist #2	Radiologist #3
Review1	2.37 ± 2.30	1.70 ± 1.65	0.81 ± 0.97	0.72 ± 1.51	0.51 ± 0.62
Review2	2.53 ± 2.27	1.58 ± 1.83	0.77 ± 0.68	0.95 ± 1.61	0.86 ± 1.30

Table 4: Comparison of the performance of both the automatic systems and radiologists. The L3 annotations given by the reference radiologist vary between the two reviews.

6. Conclusion

In this paper, we proposed a new and generic pipeline for spotting a particular slice in a CT scan. In our work, we applied our approach to the L3 slice, but it can easily be generalized to other slices, provided a labeled dataset is available.

First, the CT scan is converted into a frontal Maximum Intensity Projection (MIP) image. Afterwards, this representation is processed in a sliding window fashion to be fed to a CNN which is trained using Transfer Learning. In the test phase, all the predictions concerning the position of the L3 within the sliding windows are merged into a robust post-processing stage to take the final decision about the position of the L3 slice in the full CT scan.

Obtained results show that the approach is efficient to precisely detect the target slice. Using a fine-tuned VGG16 network coupled with an adequate decision strategy, the average error is under 2 slices where experienced radi-

ologists can provide annotations that differ of about 1 slice. The computing time is within an acceptable range for clinical applications, and can be further reduced by (i) increasing the shift value (ii) adapting the network architecture by pre-training smaller networks over ImageNet, for example, which has not been studied in this work (iii) and pruned the final trained CNN by dropping the less important filters. Recently, pruning CNNs has seen a lot of attention in order to deploy large CNNs on devices with less computation resource. We are currently working on this idea to speedup more the computation.

This contribution confirms the interest of using machine learning and more particularly deep learning in medical problems. One of the main reasons deep learning is not popular in medical domain is the lack of training data. Pre-training the networks over other large dataset will strongly alleviate this problem and encourage the use of such efficient models.

References

- [AWM⁺14] Janice L Atkins, Peter H Whincup, Richard W Morris, Lucy T Lennon, Olia Papacosta, and S Goya Wannamethee. Sarcopenic obesity and risk of cardiovascular disease and mortality: a population-based cohort study of older men. *Journal of the American Geriatrics Society*, 62(2):253–60, February 2014.
- [BDWG15] Yaniv Bar, Idit Diamant, Lior Wolf, and Hayit Greenspan. Deep learning with non-medical training used for chest pathology identification. *Proc. SPIE, Medical Imaging: Computer-Aided Diagnosis*, 9414:94140V–7, 2015.
- [Ben12] Yoshua Bengio. Practical recommendations for gradient-based training of deep architectures. In Grégoire Montavon, Genevieve B. Orr, and Klaus-Robert Müller, editors, *Neural Networks: Tricks of the Trade (2nd ed.)*, volume 7700 of *Lecture Notes in Computer Science*, pages 437–478. Springer, 2012.

- [Bre01] Leo Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, October 2001.
- [CC06] P. Checco and F. Corinto. Cnn-based algorithm for drusen identification. In *International Symposium on Circuits and Systems*, 2006.
- [CCB⁺09] Howard Chung, Dana Cobzas, Laura Birdsell, Jessica Lieffers, and Vickie Baracos. Automated segmentation of muscle and adipose tissue on CT images for human body composition analysis. *Proceedings of SPIE*, 7261:72610K–72610K–8, 2009.
- [Cho15] François Chollet. Keras. <https://github.com/fchollet/keras>, 2015.
- [CMS12] D. C. Cireşan, U. Meier, and J. Schmidhuber. Transfer learning for latin and chinese characters with deep neural networks. In *International Joint Conference on Neural Networks*, pages 1–6, 2012.
- [CWJ⁺15] A.R. Cunliffe, B. White, J. Justusson, C. Straus, R. Malik, Al-H.A. Hallaq, and S.G. Armato. Comparison of Two Deformable Registration Algorithms in the Presence of Radiologic Change Between Serial Lung CT Scans. *Journal of Digital Imaging*, 28(6):755–760, 2015.
- [DDS⁺09] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.
- [ESTA14] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. In *CVPR*, pages 2155–2162, 2014.

- [GACD11] Subarna Ghosh, Raja' S. Alomari, Vipin Chaudhary, and Gurmeet Dhillon. Automatic lumbar vertebra segmentation from clinical CT for wedge compression fracture diagnosis. *Proceedings of the SPIE*, 3:796303–9, 2011.
- [GDE⁺13] B. Glocker, D.Zikic, E.Konukoglu, D.R. Haynor, and A. Criminisi. Vertebrae localization in pathological spine CT via dense classification from sparse annotations. *MICCAI*, 16(Pt 2):262–70, 2013.
- [GFC⁺12] Ben Glocker, J. Feulner, Antonio Criminisi, D. R. Haynor, and E. Konukoglu. *Automatic Localization and Identification of Vertebrae in Arbitrary Field-of-View CT Scans*, pages 590–598. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
- [GLC⁺13] S Gouérand, M Leheurteur, M Chaker, R Modzelewski, O Rigal, C Veyret, G Lauridant, and F Clatot. A higher body mass index and fat mass are factors predictive of docetaxel dose intensity. *Anticancer research*, 33(12):5655, 2013.
- [GVC09] S. Golodetz, I. Voiculescu, and S. Cameron. Automatic spine identification in abdominal CT slices using image partition forests. *International Symposium on Image and Signal Processing and Analysis*, 2009.
- [GZH14] Ben Glocker, Darko Zikic, and David R. Haynor. *Robust Registration of Longitudinal Spine CT*, pages 251–258. Springer International Publishing, 2014.
- [HCLN09] Szu H. Huang, Yi Hong Chu, Shang Hong Lai, and Carol L. Novak. Learning-Based Vertebra Detection and Iterative Normalized-Cut Segmentation for Spinal MRI. *IEEE Transactions on Medical Imaging*, 28(10):1595–1605, 2009.

- [HDW⁺15] M. Havaei, A. Davy, D. Warde-Farley, A. Biard, A.C. Courville, Y. Bengio, C. Pal, P.-M. Jodoin, and H. Larochelle. Brain tumor segmentation with deep neural networks. *CoRR*, abs/1505.03540, 2015.
- [HJ13] Gary B. Huang and Viren Jain. Deep and wide multiscale recursive networks for robust image labeling. *CoRR*, abs/1310.0354, 2013.
- [Ho95] Tin Kam Ho. Random decision forests. In *Proceedings of the Third International Conference on Document Analysis and Recognition (Volume 1) - Volume 1*, ICDAR '95, pages 278–, Washington, DC, USA, 1995. IEEE Computer Society.
- [HSO16] Luiz G. Hafemann, Robert Sabourin, and Luiz S. Oliveira. Writer-independent feature learning for offline signature verification using deep convolutional neural networks. *CoRR*, abs/1604.00974, 2016.
- [Jia15] Xiang Jiang. Representational transfer in deep belief networks. In *28th Canadian Conference on Artificial Intelligence*, pages 338–342, 2015.
- [KLP11] Samuel Kadoury, Hubert Labelle, and Nikos Paragios. Automatic inference of articulated spine models in CT images using high-order markov random fields. *Medical Image Analysis*, 15(4):426–437, 2011.
- [KOF⁺13] Toshimi Kaido, Kohei Ogawa, Yasuhiro Fujimoto, Y Ogura, K Hata, T Ito, K Tomiyama, S Yagi, A Mori, and S Uemoto. Impact of sarcopenia on survival in patients undergoing living donor liver transplantation. *American Journal of Transplantation*, 13(6):1549–1556, 2013.

- [KSH12] A. Krizhevsky, I. Sutskever, and G.E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *NIPS 25*, pages 1097–1105. 2012.
- [Lai15] Matthew Lai. Deep learning for medical image segmentation. *CoRR*, abs/1505.02000, 2015.
- [LHC⁺15] J Lerouge, R Herault, C Chatelain, F Jardin, and R Modzelewski. IODA : An input / output deep architecture for image labeling. *Pattern Recognition*, 48(9):2847–2858, 2015.
- [LKT⁺14] H el ene Lanic, Jer ome Kraut-Tauzia, Romain Modzelewski, Florian Clatot, Sylvain Mareschal, Jean Michel Picquenot, Aspasia Stamatoullas, St ephane Lepr etre, Herv e Tilly, and Fabrice Jardin. Sarcopenia is an independent prognostic factor in elderly patients with diffuse large b-cell lymphoma treated with immunochemotherapy. *Leukemia & Lymphoma*, 55(4):817–823, 2014.
- [MBH⁺98] N Mitsiopoulos, R N Baumgartner, S B Heymsfield, W Lyons, D Gallagher, and R Ross. Cadaver validation of skeletal muscle measurement by magnetic resonance imaging and computerized tomography. *Journal of applied physiology*, 85(1):115–122, 1998.
- [MBM⁺13] Lisa Martin, Laura Birdsell, Neil MacDonald, Tony Reiman, M. Thomas Clandinin, Linda J. McCargar, Rachel Murphy, Sunita Ghosh, Michael B. Sawyer, and Vickie E. Baracos. Cancer cachexia in the age of obesity: Skeletal muscle depletion is a powerful prognostic factor, independent of body mass index. *Journal of Clinical Oncology*, 31(12):1539–1547, 2013.
- [MHSB13] David Major, Jiří Hladůvka, Florian Schulze, and Katja B uhler. Automated landmarking and labeling of fully and partially scanned spinal columns in CT images. *Medical Image Analysis*, 17(8):1151–1163, 2013.

- [ML13] Jun Ma and Le Lu. Hierarchical segmentation and identification of thoracic vertebra using learning-based edge detection and coarse-to-fine deformable model. *Computer Vision and Image Understanding*, 117(9):1072–1083, 2013.
- [MMB⁺08] Christopher Malon, Matthew Miller, Harold Christopher Burger, Eric Cosatto, and Hans Peter Graf. Identifying histological elements with convolutional neural networks. In *Int. Conf. on Soft Computing As Transdisciplinary Science and Technology*, pages 450–456, 2008.
- [MT96] T. McInerney and D. Terzopoulos. Deformable models in medical image analysis: a survey. *Medical image analysis*, 1(2):91–108, 1996.
- [MWK⁺13] B. Michael Kelm, Michael Wels, S. Kevin Zhou, Sascha Seifert, Michael Suehling, Yefeng Zheng, and Dorin Comaniciu. Spine detection in CT and MR using iterated marginal space learning. *Medical Image Analysis*, 17(8):1283–1292, 2013.
- [NLB10] Loris Nanni, Alessandra Lumini, and Sheryl Brahnem. Local binary patterns variants as texture descriptors for medical image analysis. *Artificial Intelligence in Medicine*, 49(2):117 – 125, 2010.
- [OA11] Ayse Betul Oktay and Yusuf Sinan Akgul. Localization of the lumbar discs using machine learning and exact probabilistic inference. In *Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics*, pages 158–165, 2011.
- [OPM02] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, Jul 2002.

- [PVT⁺11] Peter D. Peng, Mark G. Van Vledder, Susan Tsai, Mechteld C. De Jong, Martin Makary, Julie Ng, Barish H. Edil, Christopher L. Wolfgang, Richard D. Schulick, Michael A. Choti, Ihab Kamel, and Timothy M. Pawlik. Sarcopenia negatively impacts short-term outcomes in patients undergoing hepatic resection for colorectal liver metastasis. *HPB*, 13(7):439–446, 7 2011.
- [PXP00] Dzung L Pham, Chenyang Xu, and Jerry L Prince. Current methods in medical image segmentation 1. *Annual review of biomedical engineering*, 2(1):315–337, 2000.
- [RHGS15] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *NIPS* 28, pages 91–99, 2015.
- [RYL⁺14] Holger R. Roth, Jianhua Yao, Le Lu, James Stieger, Joseph E. Burns, and Ronald M. Summers. Detection of sclerotic spine metastases via random aggregation of deep convolutional neural network classifications. *CoRR*, abs/1407.5976, 2014.
- [SEM16] Antonis D. Savva, Theodore L. Economopoulos, and George K. Matsopoulos. Geometry-based vs. intensity-based medical image registration: A comparative study on 3D CT data. *Computers in Biology and Medicine*, 69:120–133, 2016.
- [SLJ⁺14] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going Deeper with Convolutions. *CoRR*, abs/1409.4842, 2014.
- [SPW⁺04] Wei Shen, Mark Punyanitya, ZiMian Wang, Dympna Gallagher, Marie-Pierre St-Onge, Jeanine Albu, Steven B Heymsfield, and Stanley Heshka. Total body skeletal muscle and adipose tissue volumes: estimation from a single abdominal cross-sectional image. *Journal of applied physiology*, 97(6):2333–2338, 2004.

- [SRG⁺16] H. C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers. Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning. *IEEE Transactions on Medical Imaging*, 35(5):1285–1298, 2016.
- [STE13] C. Szegedy, A. Toshev, and D. Erhan. Deep neural networks for object detection. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *NIPS 26*, pages 2553–2561. 2013.
- [SZ14] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.
- [UBHK14] G. Urban, M. Bendszus, Fred A. Hamprecht, and J. Kleesiek. Multi-modal brain tumor segmentation using deep convolutional neural networks. In *MICCAI BraTS Challenge Proceedings*, pages 31–35, 2014.
- [Wal92] Jerold W. Wallis. *Cardiovascular Nuclear Medicine and MRI: Quantitation and Clinical Applications*, pages 89–100. Springer Netherlands, 1992.
- [WM91] JW Wallis and TR Miller. Three-dimensional display in nuclear medicine and radiology. *Journal of nuclear medicine : official publication, Society of Nuclear Medicine*, 32(3):534–546, March 1991.
- [WMLK89] J. W. Wallis, T. R. Miller, C. A. Lerner, and E. C. Kleerup. Three-dimensional display in nuclear medicine. *IEEE Trans. on Medical Imaging*, 8(4):297–230, Dec 1989.

[YDM⁺15] Connie Yip, Charlotte Dinkel, Abhishek Mahajan, Musib Siddique, Gary Cook, and Vicky Goh. Imaging body composition in cancer patients: visceral obesity, sarcopenia and sarcopenic obesity may impact on clinical outcome. *Insights into Imaging*, pages 489–497, 2015.

A.3 Deep Neural Networks Regularization for Structured Output Prediction

Reference

[Bel+18] Soufiane Belharbi et al. “Deep Neural Networks Regularization for Structured Output Prediction.” In: *Neurocomputing* 281 (Mar. 15, 2018), pp. 169–177. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2017.12.002. URL: <http://www.sciencedirect.com/science/article/pii/S0925231217318295>

Deep Neural Networks Regularization for Structured Output Prediction

Soufiane Belharbi*

Normandie Univ, UNIROUEN, UNIHAVRE,
INSA Rouen, LITIS
76000 Rouen, France
soufiane.belharbi@insa-rouen.fr

Romain Hérault

Normandie Univ, UNIROUEN, UNIHAVRE,
INSA Rouen, LITIS
76000 Rouen, France
romain.herault@insa-rouen.fr

Clément Chatelain

Normandie Univ, UNIROUEN, UNIHAVRE,
INSA Rouen, LITIS
76000 Rouen, France
clement.chatelain@insa-rouen.fr

Sébastien Adam

Normandie Univ, UNIROUEN, UNIHAVRE,
INSA Rouen, LITIS
76000 Rouen, France
sebastien.adam@univ-rouen.fr

Abstract

A deep neural network model is a powerful framework for learning representations. Usually, it is used to learn the relation $x \rightarrow y$ by exploiting the regularities in the input x . In structured output prediction problems, y is multi-dimensional and structural relations often exist between the dimensions. The motivation of this work is to learn the output dependencies that may lie in the output data in order to improve the prediction accuracy. Unfortunately, feedforward networks are unable to exploit the relations between the outputs. In order to overcome this issue, we propose in this paper a regularization scheme for training neural networks for these particular tasks using a multi-task framework. Our scheme aims at incorporating the learning of the output representation y in the training process in an unsupervised fashion while learning the supervised mapping function $x \rightarrow y$.

We evaluate our framework on a facial landmark detection problem which is a typical structured output task. We show over two public challenging datasets (LFPW and HELEN) that our regularization scheme improves the generalization of deep neural networks and accelerates their training. The use of unlabeled data and label-only data is also explored, showing an additional improvement of the results. We provide an opensource implementation² of our framework.

1 Introduction

In machine learning field, the main task usually consists in learning general regularities over the input space in order to provide a specific output. Most of machine learning applications aim at predicting a single value: a label for classification or a scalar value for regression. Many recent applications address challenging problems where the output lies in a multi-dimensional space describing discrete or continuous variables that are most of the time interdependent. A typical example is speech recognition, where the output label is a sequence of characters which are interdependent, following the statistics of the considered language. These dependencies generally constitute a regular structure such as a sequence, a string, a tree or a graph. As it provides constraints that may help the prediction,

*<https://sbelharbi.github.io>

²<https://github.com/sbelharbi/structured-output-ae>

this structure should be either discovered if unknown, or integrated in the learning algorithm using prior assumptions. The range of applications that deal with structured output data is large. One can cite, among others, image labeling [12, 26, 31, 35, 49, 16, 24, 39], statistical natural language processing (NLP) [17, 33, 38, 37], bioinformatics [18, 43], speech processing [34, 47] and handwriting recognition [15, 40]. Another example which is considered in the evaluation of our proposal in this paper is the facial landmark detection problem. The task consists in predicting the coordinates of a set of keypoints given the face image as input (Fig.1). The set of points are interdependent throughout geometric relations induced by the face structure. Therefore, facial landmark detection can be considered as a structured output prediction task.



Figure 1: Examples of facial landmarks from LFPW [4] training set.

One main difficulty in structured output prediction is the exponential number of possible configurations of the output space. From a statistical point of view, learning to predict accurately high dimensional vectors requires a large amount of data where in practice we usually have limited data. In this article we propose to consider structured output prediction as a representation learning problem, where the model must i) capture the discriminative relation between x (input) and y (output), and ii) capture the interdependencies laying between the variables of each space by efficiently modeling the input and output distributions. We address this modelization through a regularization scheme for training neural networks. Feedforward neural networks lack exploiting the structural information between the y components. Therefore, we incorporate in our framework an unsupervised task which aims at discovering this hidden structure. The advantage of doing so is there is no need to fix beforehand any prior structural information. The unsupervised task learns it on itself.

Our contributions is a multi-task framework dedicated to train feedforward neural networks models for structured output prediction. We propose to combine unsupervised tasks over the input and output data in parallel with the supervised task. This parallelism can be seen as a regularization of the supervised task which helps it to generalize better. Moreover, as a second contribution, we demonstrate experimentally the benefit of using the output labels y without their corresponding inputs x . In this work, the multi task framework is instantiated using auto-encoders [46, 5] for both representations learning and exploiting unlabeled data (input) and label-only data (output). We demonstrate the efficiency of our proposal over a real-world facial landmark detection problem.

The rest of the paper is organized as follows. Related works about structured output prediction is proposed in section 2. Section 3 presents the proposed formulation and its optimization details. Section 4 describes the instantiation of the formulation using a deep neural network. Finally, section 5 details the conducted experiments including the datasets, the evaluation metrics and the general training setup. Two types of experiments are explored: with and without the use of unlabeled data. Results are presented and discussed for both cases.

2 Related work

We distinguish two main categories of methods for structured output prediction. For a long time, graphical models have showed a large success in different applications involving 1D and 2D signals. Recently, a new trend has emerged based on deep neural networks.

2.1 Graphical Models Approaches

Historically, graphical models are well known to be suitable for learning structures. One of their main strength is an easy integration of explicit structural constraints and prior knowledge directly into the model's structure. They have shown a large success in modeling structured data thanks to their capacity to capture dependencies among relevant random variables. For instance, Hidden Markov Models (HMM) framework has a large success in modeling sequence data. HMMs make an assumption that the output random variables are supposed to be independent which is not the case in many real-world applications where strong relations are present. Conditional Random Fields (CRF) have been proposed to overcome this issue, thanks to its capability to learn large dependencies of the observed output data. These two frameworks are widely used to model structured output data represented as a 1-D sequence [11, 34, 6, 21]. Many approaches have also been proposed to deal with 2-D structured output data as an extension of HMM and CRF. [29] propose a Markov Random Field (MRF) for document image segmentation. [44] provide an adaptation of CRF to 2-D signals with hand drawn diagrams interpretation. Another extension of CRF to 3-D signal is presented in [45] for 3-D medical image segmentation. Despite the large success of graphical models in many domains, they still encounter some difficulties. For instance, due to their inference computational cost, graphical models are limited to low dimensional structured output problems. Furthermore, HMM and CRF models are generally used with discrete output data where few works address the regression problem [32, 13].

2.2 Deep Neural Networks Approaches

More recently, deep learning based approaches have been widely used to solve structured output prediction, especially proposed for image labeling problems. Deep learning domain provides many different architectures. Therefore, different solutions were proposed depending on the application in hand and what is expected as a result.

In image labeling task (also known as semantic segmentation), one needs models able to adapt to the large variations in the input image. Given their large success in image processing related tasks [20], convolutional neural networks is a natural choice. Therefore, they have been used as the core model in image labeling problems in order to learn the relevant features. They have been used either combined with simple post-processing in order to calibrate the output [8] or with more sophisticated models in structure modeling such as CRF [12] or energy based models [30]. Recently, a new trend has emerged, based on the application of convolution [26, 35] or deconvolutional [31] layers in the output of the network which goes by the name of fully convolutional networks and showed successful results in image labeling. Despite this success, these models does not take in consideration the output representation.

In many applications, it is not enough to provide the output prediction, but also its probability. In this case, Conditional Restricted Boltzmann Machines, a particular case of neural networks and probabilistic graphical models have been used with different training algorithms according to the size of the plausible output configurations [28]. Training and inferring using such models remains a difficult task. In this same direction, [2] tackle structured output problems as an energy minimization through two feed-forward networks. The first is used for feature extraction over the input. The second is used for estimating an energy by taking as input the extracted features and the current state of the output labels. This allows learning the interdependencies within the output labels. The prediction is performed using an iterative backpropagation-based method with respect to the labels through the second network which remains computationally expensive. Similarly, Recurrent Neural Networks (RNN) are a particular architecture of neural networks. They have shown a great success in modeling sequence data and outputting sequence probability for applications such as Natural Language Processing (NLP) tasks [25, 42, 1] and speech recognition [14]. It has also been used for image captioning [19]. However, RNN models doe not consider explicitly the output dependencies.

In [23], our team proposed the use of auto-encoders in order to learn the output distribution in a pre-training fashion with application to image labeling with promising success. The approach consists in two sequential steps. First, an input and output pre-training is performed in an unsupervised way using autoencoders. Then, a finetune is applied on the whole network using supervised data. While this approach allows incorporating prior knowledge about the output distribution, it has two main issues. First, the alteration of a network output layer is critical and must be performed carefully. Moreover, one needs to perform multiple trial-error loops in order to set the autoencoder's training

hyper-parameters. The second issue is overfitting. When pre-training the output auto-encoder, there is actually no information that indicates if the pre-training is helping the supervised task, nor when to stop the pre-training.

The present work proposes a general and easy to use multi-task training framework for structured output prediction models. The input and the output unsupervised tasks are embedded into a regularization scheme and learned in parallel with the supervised task. The rationale behind is that the unsupervised tasks should provide a *generalization* aspect to the main supervised task and should limit overfitting. This parallel transfer learning which includes an output reconstruction task constitutes the main contribution of this work. In structured output context, the role of the output task is to learn the hidden structure within the original output data, in an unsupervised way. This can be very helpful in models that do not consider the relations between the components of the output representation such as feedforward neural networks. We also show that the proposed framework enables to use labels without input in an unsupervised fashion and its effect on the generalization of the model. This can be very useful in applications where the output data is abundant such as in a speech recognition task where the output is ascii text which can be easily gathered from Internet. In this article, we validate our proposal on a facial landmark prediction problem over two challenging public datasets (LFPW and HELEN). The performed experiments show an improvement of the generalization of deep neural networks and an acceleration of their training.

3 Multi-task Training Framework for Structured Output Prediction

Let us consider a training set \mathcal{D} containing examples with both features and targets (x, y) , features without target $(x, _)$, and targets without features $(_, y)$. Let us consider a set \mathcal{F} which is the subset of \mathcal{D} containing examples with at least features x , a set \mathcal{L} which is the subset of \mathcal{D} containing examples with at least targets y , and a set \mathcal{S} which is the subset of \mathcal{D} containing examples with both features x and targets y . One can note that all examples in \mathcal{S} are also in \mathcal{F} and in \mathcal{L} .

Input task

The input task \mathcal{R}_{in} is an unsupervised reconstruction task which aims at learning global and more robust input representation based on the original input data \mathbf{x} . This task projects the input data \mathbf{x} into an intermediate representation space $\tilde{\mathbf{x}}$ through a coding function P_{in} , known as encoder. Then, it attempts to recover the original input by reconstructing $\hat{\mathbf{x}}$ from $\tilde{\mathbf{x}}$ through a decoding function P'_{in} , known as decoder:

$$\hat{\mathbf{x}} = \mathcal{R}_{in}(\mathbf{x}; \mathbf{w}_{in}) = P'_{in}(\tilde{\mathbf{x}} = P_{in}(\mathbf{x}; \mathbf{w}_{cin}); \mathbf{w}_{din}) , \quad (1)$$

where $\mathbf{w}_{in} = \{\mathbf{w}_{cin}, \mathbf{w}_{din}\}$. The decoder parameters \mathbf{w}_{din} are proper to this task however the encoder parameters \mathbf{w}_{cin} are shared with the main task (see Fig.2). This multi-task aspect will attract, hopefully, the shared parameters in the parameters space toward regions that build more general and robust input representations and avoid getting stuck in local minima. Therefore, it promotes generalization. This can be useful to start the training process of the main task.

The training criterion for this task is given by :

$$\mathcal{J}_{in}(\mathcal{F}; \mathbf{w}_{in}) = \frac{1}{\text{card } \mathcal{F}} \sum_{x \in \mathcal{F}} \mathcal{C}_{in}(\mathcal{R}_{in}(\mathbf{x}; \mathbf{w}_{in}), \mathbf{x}) , \quad (2)$$

where \mathcal{C}_{in} is an unsupervised learning cost which can be computed on all the samples with features (i.e. on \mathcal{F}). Practically, it can be the mean squared error.

Output task

The output task \mathcal{R}_{out} is an unsupervised reconstruction task which has the same goal as the input task. Similarly, this task projects the output data \mathbf{y} into an intermediate representation space $\tilde{\mathbf{y}}$ through a coding function P_{out} , i.e. a coder. Then, it attempts to recover the original output data by reconstructing $\hat{\mathbf{y}}$ based on $\tilde{\mathbf{y}}$ through a decoding function P'_{out} , i.e. a decoder. In structured output data, $\tilde{\mathbf{y}}$ can be seen as a code that contains many aspect of the original output data \mathbf{y} , most importantly, its hidden structure that describes the global relation between the components of \mathbf{y} . This hidden structure is discovered in an unsupervised way without priors fixed beforehand which makes it simple to use. Moreover, it allows using

labels only (without input \mathbf{x}) which can be helpful in tasks with abundant output data such as in speech recognition task (Sec.2):

$$\hat{\mathbf{y}} = \mathcal{R}_{out}(\mathbf{y}; \mathbf{w}_{out}) = P'_{out}(\tilde{\mathbf{y}} = P_{out}(\mathbf{y}; \mathbf{w}_{cout}); \mathbf{w}_{dout}) . \quad (3)$$

where $\mathbf{w}_{out} = \{\mathbf{w}_{cout}, \mathbf{w}_{dout}\}$. In the opposite of the input task, the encoder parameters \mathbf{w}_{cout} are proper to this task while the decoder parameters \mathbf{w}_{dout} are shared with the main task (see Fig.2).

The training criterion for this task is given by :

$$\mathcal{J}_{out}(\mathcal{L}; \mathbf{w}_{out}) = \frac{1}{\text{card } \mathcal{L}} \sum_{y \in \mathcal{L}} \mathcal{C}_{out}(\mathcal{R}_{out}(\mathbf{y}; \mathbf{w}_{out}), \mathbf{y}) , \quad (4)$$

where \mathcal{C}_{out} is an unsupervised learning cost which can be computed on all the samples with labels (i.e. on \mathcal{L}), typically, the mean squared error.

Main task

The main task is a supervised task that attempts to learn the mapping function \mathcal{M} between features \mathbf{x} and labels \mathbf{y} . In order to do so, the first part of the mapping function is shared with the encoding part P_{in} of the input task and the last part is shared with the decoding part P'_{out} of the output task. The middle part m of the mapping function \mathcal{M} is specific to this task:

$$\hat{\mathbf{y}} = \mathcal{M}(\mathbf{x}; \mathbf{w}_{sup}) = P'_{out}(m(P_{in}(\mathbf{x}; \mathbf{w}_{cin}); \mathbf{w}_s); \mathbf{w}_{dout}) . \quad (5)$$

where $\mathbf{w}_{sup} = \{\mathbf{w}_{cin}, \mathbf{w}_s, \mathbf{w}_{dout}\}$. Accordingly, \mathbf{w}_{cin} and \mathbf{w}_{dout} parameters are respectively shared with the input and output tasks.

Learning this task consists in minimizing its learning criterion \mathcal{J}_s ,

$$\mathcal{J}_s(\mathcal{S}; \mathbf{w}_{sup}) = \frac{1}{\text{card } \mathcal{S}} \sum_{(x,y) \in \mathcal{S}} \mathcal{C}_s(\mathcal{M}(x; \mathbf{w}_{sup}), y) , \quad (6)$$

where $\mathcal{C}_s(\cdot, \cdot)$ can be the mean squared error.

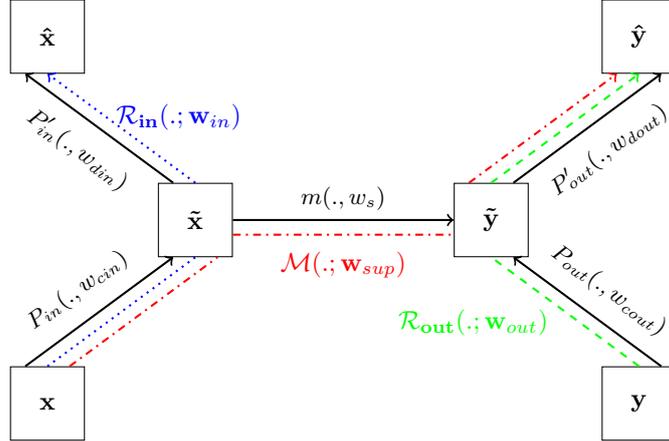


Figure 2: Proposed MTL framework. Black plain arrows stand for intermediate functions, blue dotted arrow for input auxiliary task \mathcal{R}_{in} , green dashed arrow for output auxiliary task \mathcal{R}_{out} , and red dash-dotted arrow for the main supervised task \mathcal{M} .

As a synthesis, our proposal is formulated as a multi-task learning framework (MTL) [7], which gathers a main task and two secondary tasks. This framework is illustrated in Fig. 2.

Learning the three tasks is performed in parallel. This can be translated in terms of training cost as the sum of the corresponding costs. Given that the tasks have different importance, we weight each

cost using a corresponding importance weight λ_{sup} , λ_{in} and λ_{out} respectively for the supervised, the input and output tasks. Therefore, the full objective of our framework can be written as:

$$\mathcal{J}(\mathcal{D}; \mathbf{w}) = \lambda_{sup} \cdot \mathcal{J}_s(\mathcal{S}; \mathbf{w}_{sup}) + \lambda_{in} \cdot \mathcal{J}_{in}(\mathcal{F}; \mathbf{w}_{in}) + \lambda_{out} \cdot \mathcal{J}_{out}(\mathcal{L}; \mathbf{w}_{out}), \quad (7)$$

where $\mathbf{w} = \{\mathbf{w}_{cin}, \mathbf{w}_{din}, \mathbf{w}_s, \mathbf{w}_{cout}, \mathbf{w}_{dout}\}$ is the complete set of parameters of the framework.

Instead of using fixed importance weights that can be difficult to optimally set, we evolve them through the learning epochs. In this context, Eq. 7 is modified as follows :

$$\mathcal{J}(\mathcal{D}; \mathbf{w}) = \lambda_{sup}(t) \cdot \mathcal{J}_s(\mathcal{S}; \mathbf{w}_{sup}) + \lambda_{in}(t) \cdot \mathcal{J}_{in}(\mathcal{F}; \mathbf{w}_{in}) + \lambda_{out}(t) \cdot \mathcal{J}_{out}(\mathcal{L}; \mathbf{w}_{out}), \quad (8)$$

where $t \geq 0$ indicates the learning epochs. Our motivation to evolve the importance weights is that we want to use the secondary tasks to start the training and avoid the main task to get stuck in local minima early in the beginning of the training by moving the parameters towards regions that generalize better. Then, toward the end of the training, we drop the secondary tasks by annealing their importance toward zero because they are no longer necessary for the main task. The early stopping of the secondary tasks is important in this context of multi-tasking as shown in [50] otherwise, they will overfit, therefore, they will harm the main task. The main advantage of Eq.8 is that it allows an interaction between the main supervised task and the secondary tasks. Our hope is that this interaction will promote the generalization aspect of the main task and prevent it from overfitting.

4 Implementation

In this work, we implement our framework throughout a deep neural network. The main supervised task is performed using a deep neural network (DNN) with K layers. Secondary reconstruction tasks are carried out by auto-encoders (AE): the input task is achieved using an AE that has K_{in} layers in its encoding part, with an encoded representation of the same dimension as $\tilde{\mathbf{x}}$. Similarly, the output task is achieved using an AE that has K_{out} layers in its decoding part, with an encoded representation of the same dimension as $\tilde{\mathbf{y}}$. At least one layer must be dedicated in the DNN to link $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ in the intermediate spaces. Therefore, $K_{in} + K_{out} < K$.

Parameters \mathbf{w}_{in} are the parameters of the whole input AE, \mathbf{w}_{out} are the parameters of the whole output AE and \mathbf{w}_{sup} are the parameters of the main neural network (NN). The encoding layers of the input AE are tied to the first layers of the main NN, and the decoding layers of the output AE are in turn tied to the last layers of the main NN. If \mathbf{w}_i are the parameters of layer i of a neural network, then \mathbf{w}_1 to $\mathbf{w}_{K_{in}}$ parameters of the input AE are shared with \mathbf{w}_1 to $\mathbf{w}_{K_{in}}$ parameters of the main NN. Moreover, if \mathbf{w}_{-i} are the parameters of last minus $i - 1$ layer of a neural network, then parameters $\mathbf{w}_{-K_{out}}$ to \mathbf{w}_{-1} of the output AE are shared with the parameters $\mathbf{w}_{-K_{out}}$ to \mathbf{w}_{-1} of the main NN.

During training, the loss function of the input AE is used as \mathcal{J}_{in} , the loss function of the output AE is used as \mathcal{J}_{out} , and the loss function of the main NN is used as \mathcal{J}_s .

Optimizing Eq.8 can be performed using Stochastic Gradient Descent. In the case of task combination, one way to perform the optimization is to alternate between the tasks when needed [9, 50]. In the case where the training set does not contain unlabeled data, the optimization of Eq.8 can be done in parallel over all the tasks. When using unlabeled data, the gradient for the whole cost can not be computed at once. Therefore, we need to split the gradient for each sub-cost according to the nature of the samples at each mini-batch. For the sake of clarity, we illustrate our optimization scheme in Algorithm 1 using on-line training (i.e. training one sample at a time). Mini-batch training can be performed in the same way.

5 Experiments

We evaluate our framework on a facial landmark detection problem which is typically a structured output problem since the facial landmarks are spatially inter-dependent. Facial landmarks are a set of key points on human face images as shown in Fig. 1. Each key point is defined by the coordinates (x, y) in the image $(x, y \in \mathbb{R})$. The number of landmarks is dataset or application dependent.

It must be emphasized here that the purpose of our experiments in this paper was not to outperform the state of the art in facial landmark detection but to show that learning the output dependencies helps improving the performance of DNN on that task. Thus, we will compare a model with/without

Algorithm 1 Our training strategy for one epoch

```
1:  $\mathcal{D}$  is the shuffled training set.  $B$  a sample.
2: for  $B$  in  $\mathcal{D}$  do
3:   if  $B$  contains  $x$  then
4:     Update  $w_{in}$ : Make a gradient step toward  $\lambda_{in} \times \mathcal{J}_{in}$  using  $B$  (Eq.2).
5:   end if
6:   if  $B$  contains  $y$  then
7:     Update  $w_{out}$ : Make a gradient step toward  $\lambda_{out} \times \mathcal{J}_{out}$  using  $B$  (Eq.4).
8:   end if
9:   # parallel parameters update
10:  if  $B$  contains  $x$  and  $y$  then
11:    Update  $w$ : Make a gradient step toward  $\mathcal{J}$  using  $B$  (Eq.8).
12:  end if
13:  Update  $\lambda_{sup}$ ,  $\lambda_{in}$  and  $\lambda_{out}$ .
14: end for
```

input and output training. [48] use a cascade of neural networks. In their work, they provide the performance of their first global network. Therefore, we will use it as a reference to compare our performance (both networks has close architectures) except they use larger training dataset.

We first describe the datasets followed by a description of the evaluation metrics used in facial landmark problems. Then, we present the general setup of our experiments followed by two types of experiments: without and with unlabeled data. An opensource implementation of our MTL deep instantiation is available online³.

5.1 Datasets

We have carried out our evaluation over two challenging public datasets for facial landmark detection problem: LFPW [4] and HELEN [22].

LFPW dataset consists of 1132 training images and 300 test images taken under unconstrained conditions (in the wild) with large variations in the pose, expression, illumination and with partial occlusions (Fig.1). This makes the facial point detection a challenging task on this dataset. From the initial dataset described in LFPW [4], we use only the 811 training images and the 224 test images provided by the ibug website⁴. Ground truth annotations of 68 facial points are provided by [36]. We divide the available training samples into two sets: validation set (135 samples) and training set (676 samples).

HELEN dataset is similar to LFPW dataset, where the images have been taken under unconstrained conditions with high resolution and collected from Flickr using text queries. It contains 2000 images for training, and 330 images for test. Images and face bounding boxes are provided by the same site as for LFPW. The ground truth annotations are provided by [36]. Examples of dataset are shown in Fig.3.



Figure 3: Samples from HELEN [22] dataset.

All faces are cropped into the same size (50×50) and pixels are normalized in $[0,1]$. The facial landmarks are normalized into $[-1,1]$.

³<https://github.com/sbelharbi/structured-output-ae>

⁴300 faces in-the-wild challenge <http://ibug.doc.ic.ac.uk/resources/300-W/>

5.2 Metrics

In order to evaluate the prediction of the model, we use the standard metrics used in facial landmark detection problems.

The Normalized Root Mean Squared Error (NRMSE)[10] (Eq.9) is the Euclidean distance between the predicted shape and the ground truth normalized by the product of the number of points in the shape and the inter-ocular distance D (distance between the eyes pupils of the ground truth),

$$NRMSE(s_p, s_g) = \frac{1}{N * D} \sum_{i=1}^N \|s_{pi} - s_{gi}\|_2, \quad (9)$$

where s_p and s_g are the predicted and the ground truth shapes, respectively. Both shapes have the same number of points N . D is the inter-ocular distance of the shape s_g .

Using the NRMSE, we can calculate the Cumulative Distribution Function for a specific NRMSE (CDF_{NRMSE}) value (Eq.10) overall the database,

$$CDF_x = \frac{CARD(NRMSE \leq x)}{n}, \quad (10)$$

where $CARD(\cdot)$ is the cardinal of a set. n is the total number of images.

The CDF_{NRMSE} represents the percentage of images with error less or equal than the specified NRMSE value. For example a $CDF_{0.1} = 0.4$ over a test set means that 40% of the test set images have an error less or equal than 0.1. A CDF curve can be plotted according to these CDF_{NRMSE} values by varying the value of $NRMSE$.

These are the usual evaluation criteria used in facial landmark detection problem. To have more numerical precision in the comparison in our experiments, we calculate the Area Under the CDF Curve (AUC), using only the NRMSE range [0,0.5] with a step of 10^{-3} .

5.3 General training setup

To implement our framework, we use: - a DNN with four layers $K = 4$ for the main task; - an input AE with one encoding layer $K_{in} = 1$ and one decoding layer; - an output AE with one encoding layer and one decoding layer $K_{out} = 1$. Referring to Fig.2, the size of the input representation \mathbf{x} and estimation $\hat{\mathbf{x}}$ is $2500 = 50 \times 50$; the size of the output representation \mathbf{y} and estimation $\hat{\mathbf{y}}$ is $136 = 68 \times 2$, given the 68 landmarks in a 2D plane; the dimension of intermediate spaces $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ have been set to 1025 and 64 respectively; finally, the hidden layer in the m link between $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}$ is composed of 512 units. The size of each layer has been set using a validation procedure on the LFPW validation set.

Sigmoid activation functions are used everywhere in the main NN and in the two AEs, except for the last layer of the main NN and the tied last layer of output AE which use a hyperbolic tangent activation function to suite the range $[-1, 1]$ for the output \mathbf{y} .

We use the same architecture through all the experiments for the different training configurations. To distinguish between the multiple configurations we set the following notations:

1. **MLP**, a DNN for the main task with no concomitant training;
2. **MLP + in**, a DNN with input AE parallel training;
3. **MLP + out**, a DNN with output AE parallel training;
4. **MLP + in + out**, a DNN with both input and output reconstruction secondary tasks.

We recall that the auto-encoders are used only during the training phase. In the test phase, they are dropped. Therefore, the final test networks have the same architecture in all the different configurations.

Beside these configurations, we consider the mean shape (the average of the \mathbf{y} in the training data) as a simple predictive model. For each test image, we predict the same estimated mean shape over the train set.

To clarify the benefit of our approach, all the configurations must start from the same initial weights to make sure that the obtained improvement is due to the training algorithm, not to the random initialization.

For the input reconstruction tasks, we use a denoising auto-encoder with a corruption level of 20% for the first hidden layer. For the output reconstruction task, we use a simple auto-encoder. To avoid overfitting, the auto-encoders are trained using L_2 regularization with a weight decay of 10^{-2} .

In all the configurations, the update of the parameters of each task (supervised and unsupervised) is performed using Stochastic Gradient Descent with momentum [41] with a constant momentum coefficient of 0.9. We use mini-batch size of 10. The training is performed for 1000 epochs with a learning rate of 10^{-3} .

In these experiments, we propose to use a simple linear evolution scheme for the importance weights λ_{sup} (supervised task), λ_{in} (input task) and λ_{out} (output task). We retain the evolution proposed in [3], and presented in Fig.4.

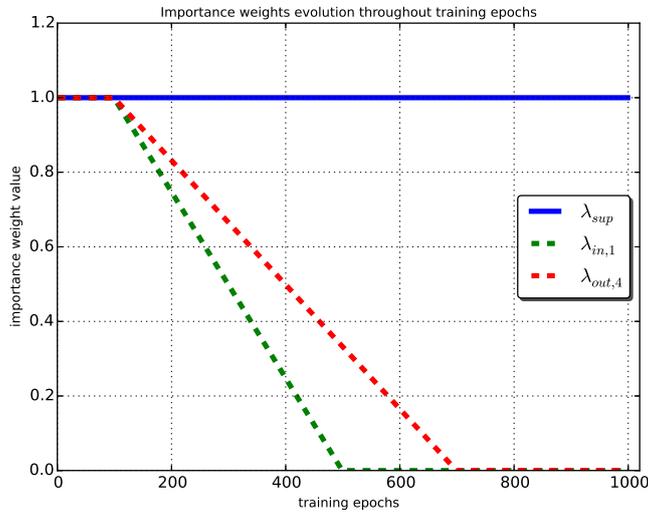


Figure 4: Linear evolution of the importance weights during training.

The hyper-parameters (learning rate, batch size, momentum coefficient, weight decay, the importance weights) have been optimized on the LFPW validation set. We apply the same optimized hyper-parameters for HELEN dataset.

Using these configurations, we perform two types of experiments: with and without unlabeled data. We present in the next sections the obtained results.

5.3.1 Experiments with fully labeled data

In this setup, we use the provided labeled data from each set in a classical way. For LFPW set, we use the 676 available samples for training and 135 samples for validation. For HELEN set, we use 1800 samples for training and 200 samples for validation.

In order to evaluate the different configurations, we first calculate the Mean Squared Error (MSE) of the best models found using the validation during the training. Column 1 (no unlabeled data) of Tab.1, 2 shows the MSE over the train and valid sets of LFPW and HELEN datasets, respectively. Compared to an MLP alone, adding the input training of the first hidden layer slightly reduces the train and validation error in both datasets. Training the output layer also reduces the train and validation error, with a more important factor. Combining the input train of the first hidden layer and output train of the last layer gives the best performance. We plot the tracked MSE over the train and valid sets of HELEN dataset in Fig.7(a), 7(b). One can see that the input training reduces slightly the validation MSE. The output training has a major impact over the training speed and the generalization of the model which suggests that output training is useful in the case of structured output problems.

Combining the input and the output training improves even more the generalization. Similar behavior was found on LFPW dataset.

At a second time, we evaluate each configuration over the test set of each datasets using the $CDF_{0.1}$ metric. The results are depicted in Tab.3, 4 in the first column for LFPW and HELEN datasets, respectively. Similarly to the results previously found over the train and validation set, one can see that the joint training (supervised, input, output) outperforms all the other configurations in terms of $CDF_{0.1}$ and AUC. The CDF curves in Fig.8 also confirms this result. Compared to the global DNN in [48] over LFPW test set, our joint trained MLP performs better ([48]: $CDF_{0.1} = 65\%$, ours: $CDF_{0.1} = 69.64\%$), despite the fact that their model was trained using larger supervised dataset (combination of multiple supervised datasets beside LFPW).

An illustrative result of our method is presented in Fig.5, 6 for LFPW and HELEN using an MLP and MLP with input and output training.



Figure 5: Examples of prediction on LFPW test set. For visualizing errors, red segments have been drawn between ground truth and predicted landmark. Top row: MLP. Bottom row: MLP+in+out. (no unlabeled data)



Figure 6: Examples of prediction on HELEN test set. Top row: MLP. Bottom row: MLP+in+out. (no unlabeled data)

5.3.2 Data augmentation using unlabeled data or label-only data

In this section, we experiment our approach when adding unlabeled data (input and output). Unlabeled data (i.e. image faces without the landmarks annotation) are abundant and can be found easily for

Table 1: MSE over LFPW: train and valid sets, at the end of training with and without unlabeled data.

	No unlabeled data		With unlabeled data	
	MSE train	MSE valid	MSE train	MSE valid
Mean shape	7.74×10^{-3}	8.07×10^{-3}	7.78×10^{-3}	8.14×10^{-3}
MLP	3.96×10^{-3}	4.28×10^{-3}	-	-
MLP + in	3.64×10^{-3}	3.80×10^{-3}	1.44×10^{-3}	2.62×10^{-3}
MLP + out	2.31×10^{-3}	2.99×10^{-3}	1.51×10^{-3}	2.79×10^{-3}
MLP + in + out	2.12×10^{-3}	2.56×10^{-3}	1.10×10^{-3}	2.23×10^{-3}

Table 2: MSE over HELEN: train and valid sets, at the end of training with and without data augmentation.

	Fully labeled data only		Adding unlabeled or label-only data	
	MSE train	MSE valid	MSE train	MSE valid
Mean shape	7.59×10^{-3}	6.95×10^{-3}	7.60×10^{-3}	0.95×10^{-3}
MLP	3.39×10^{-3}	3.67×10^{-3}	-	-
MLP + in	3.28×10^{-3}	3.42×10^{-3}	2.31×10^{-3}	2.81×10^{-3}
MLP + out	2.48×10^{-3}	2.90×10^{-3}	2.00×10^{-3}	2.74×10^{-3}
MLP + in + out	2.34×10^{-3}	2.53×10^{-3}	1.92×10^{-3}	2.40×10^{-3}

Table 3: AUC and $CDF_{0.1}$ performance over LFPW test dataset with and without unlabeled data.

	Fully labeled data only		Adding unlabeled or label-only data	
	AUC	$CDF_{0.1}$	AUC	$CDF_{0.1}$
Mean shape	68.78%	30.80%	77.81%	22.33%
MLP	76.34%	46.87%	-	-
MLP + in	77.13%	54.46%	80.78%	67.85%
MLP + out	80.93%	66.51%	81.77%	67.85%
MLP + in + out	81.51%	69.64%	82.48%	71.87%

Table 4: AUC and $CDF_{0.1}$ performance over HELEN test dataset with and without unlabeled data.

	Fully labeled data only		Adding unlabeled or label-only data	
	AUC	$CDF_{0.1}$	AUC	$CDF_{0.1}$
Mean shape	64.60%	23.63%	64.76%	23.23%
MLP	76.26%	52.72%	-	-
MLP + in	77.08%	54.84%	79.25%	63.33%
MLP + out	79.63%	66.60%	80.48%	65.15%
MLP + in + out	80.40%	66.66%	81.27%	71.51%

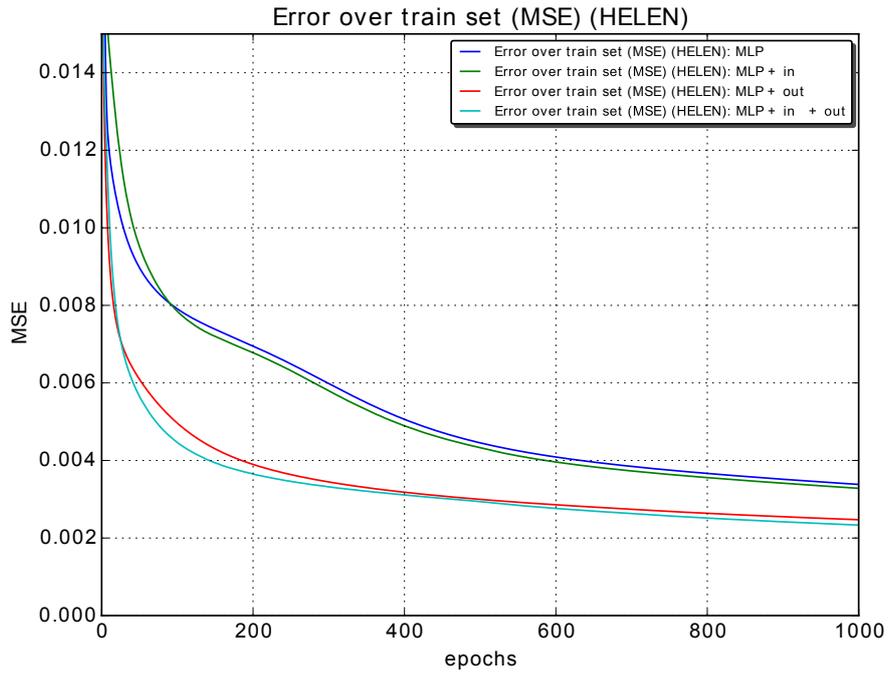
example from other datasets or from the Internet which makes it practical and realistic. In our case, we use image faces from another dataset.

In the other hand, label-only data (i.e. the landmarks annotation without image faces) are more difficult to obtain because we usually have the annotation based on the image faces. One way to obtain accurate and realistic facial landmarks without image faces is to use a 3D face model as a generator. We use an easier way to obtain facial landmarks annotation by taking them from another dataset.

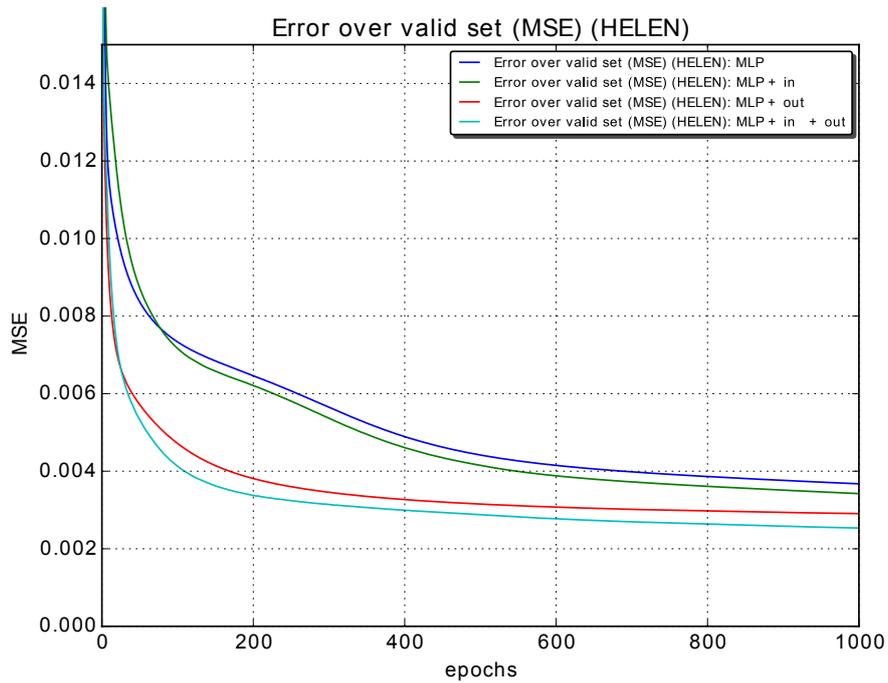
In this experiment, in order to add unlabeled data for LFPW dataset, we take all the image faces of HELEN dataset (train, valid and test) and vice versa for HELEN dataset by taking all LFPW image faces as unlabeled data. The same experiment is performed for the label-only data using the facial landmarks annotation. We summarize the size of each train set in Tab.5..

Table 5: Size of augmented LFPW and HELEN train sets.

Train set / size of	Supervised data	Unsupervised input x	Unsupervised output y
LFPW	676	2330	2330
HELEN	1800	1035	1035



(a)

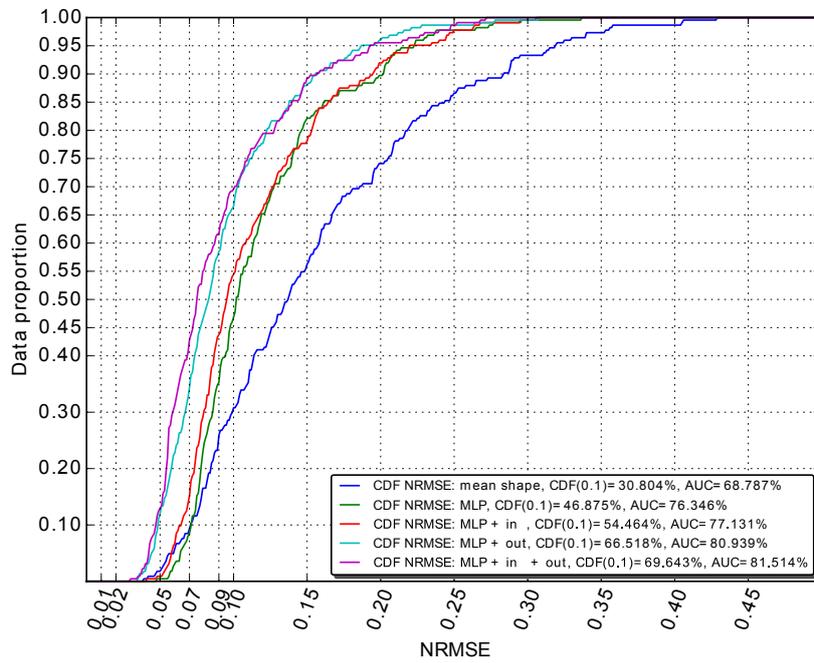


(b)

Figure 7: MSE during training epochs over HELEN train (a) and valid (b) sets using different training setups for the MLP.

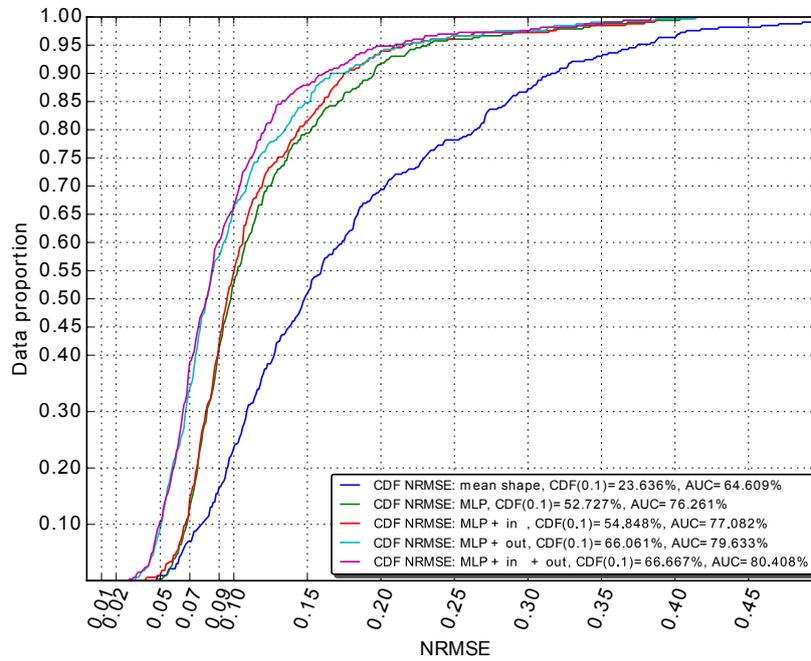
We use the same validation sets as in Sec.5.3.1 in order to have a fair comparison. The MSE are presented in the second column of Tab.1, 2 over LFPW and HELEN datasets. One can see that

Cumulative distribution function (CDF) of NRMSE over LFPW test set.



(a)

Cumulative distribution function (CDF) of NRMSE over HELEN test set.



(b)

Figure 8: CDF curves of different configurations on: (a) LFPW, (b) HELEN.

adding unlabeled data decreases the MSE over the train and validation sets. Similarly, we found that the input training along with the output training gives the best results. Identically, these results are translated in terms of $CDF_{0.1}$ and AUC over the test sets (Tab.3, 4). All these results suggest that adding unlabeled input and output data can improve the generalization of our framework and the training speed.

6 Conclusion and Future Work

In this paper, we tackled structured output prediction problems as a representation learning problem. We have proposed a generic multi-task training framework as a regularization scheme for structured output prediction models. It has been instantiated through a deep neural network model which learns the input and output distributions using auto-encoders while learning the supervised task $\mathbf{x} \rightarrow \mathbf{y}$. Moreover, we explored the possibility of using the output labels \mathbf{y} without their corresponding input data \mathbf{x} which showed more improvement in the generalization. Using a parallel scheme allows an interaction between the main supervised task and the unsupervised tasks which helped preventing the overfitting of the main task.

We evaluated our training method on a facial landmark detection task over two public datasets. The obtained results showed that our proposed regularization scheme improves the generalization of neural networks model and speeds up their training. We believe that our approach provides an alternative for training deep architectures for structured output prediction where it allows the use of unlabeled input and label of the output data.

As a future work, we plan to evolve automatically the importance weights of the tasks. For that and in order to better guide their evolution, we can consider the use of different indicators based on the training and the validation errors instead of the learning epochs only. Furthermore, one may consider other kind of models instead of simple auto-encoders in order to learn the output distribution. More specifically, generative models such as variational and adversarial auto-encoders [27] could be explored.

Acknowledgments

This work has been partly supported by the grant ANR-11-JS02-010 LeMon, the grant ANR-16-CE23-0006 “Deep in France” and has benefited from computational means from CRIANN, the contributions of which are greatly appreciated.

References

- [1] Michael Auli, Michel Galley, Chris Quirk, and Geoffrey Zweig. Joint language and translation modeling with recurrent neural networks. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1044–1054, 2013.
- [2] David Belanger and Andrew McCallum. Structured prediction energy networks. In *Proceedings of the 33rd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, pages 983–992, 2016.
- [3] S. Belharbi, R.Hérault, C. Chatelain, and S. Adam. Deep multi-task learning with evolving weights. In *European Symposium on Artificial Neural Networks (ESANN)*, 2016.
- [4] Peter N. Belhumeur, David W. Jacobs, David J. Kriegman, and Neeraj Kumar. Localizing parts of faces using a consensus of exemplars. In *CVPR*, pages 545–552. IEEE, 2011.
- [5] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle. Greedy Layer-Wise Training of Deep Networks. In B. Schölkopf, J.C. Platt, and T. Hoffman, editors, *NIPS*, pages 153–160. 2007.
- [6] Daniel M Bikel, Richard Schwartz, and Ralph M Weischedel. An algorithm that learns what’s in a name. *Machine learning*, 34(1-3):211–231, 1999.
- [7] R. Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997.

- [8] Dan C. Ciresan, Alessandro Giusti, Luca Maria Gambardella, and Jürgen Schmidhuber. Deep neural networks segment neuronal membranes in electron microscopy images. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States.*, pages 2852–2860, 2012.
- [9] R. Collobert and J. Weston. A unified architecture for natural language processing: deep neural networks with multitask learning. In *ICML*, pages 160–167, 2008.
- [10] D. Cristinacce and T. Cootes. Feature Detection and Tracking with Constrained Local Models. In *BMVC*, pages 95.1–95.10, 2006.
- [11] M. El-Yacoubi, M. Gilloux, and J-M Bertille. A statistical approach for phrase location and recognition within a text line: An application to street name recognition. *IEEE PAMI*, 24(2):172–188, 2002.
- [12] Clément Farabet, Camille Couprie, Laurent Najman, and Yann LeCun. Learning Hierarchical Features for Scene Labeling. *IEEE PAMI*, 35(8):1915–1929, 2013.
- [13] Moshe Fridman. *Hidden markov model regression*. PhD thesis, Graduate School of Arts and Sciences, University of Pennsylvania, 1993.
- [14] Alex Graves and Navdeep Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, pages 1764–1772, 2014.
- [15] Alex Graves, Marcus Liwicki, Santiago Fernández, Roman Bertolami, Horst Bunke, and Jürgen Schmidhuber. A novel connectionist system for unconstrained handwriting recognition. *IEEE transactions on pattern analysis and machine intelligence*, 31(5):855–868, 2009.
- [16] Judy Hoffman, Dequan Wang, Fisher Yu, and Trevor Darrell. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation. *CoRR*, abs/1612.02649, 2016.
- [17] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep structured output learning for unconstrained text recognition. *CoRR*, abs/1412.5903, 2014.
- [18] David T. Jones. Protein secondary structure prediction based on position-specific scoring matrices. *Journal of Molecular Biology*, 292(2):195–202, 1999.
- [19] Andrej Karpathy and Fei-Fei Li. Deep visual-semantic alignments for generating image descriptions. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 3128–3137, 2015.
- [20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.
- [21] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *ICML*, pages 282–289, 2001.
- [22] Vuong Le, Jonathan Brandt, Zhe Lin, Lubomir D. Bourdev, and Thomas S. Huang. Interactive Facial Feature Localization. In *ECCV, 2012, Proceedings, Part III*, pages 679–692, 2012.
- [23] J. Lerouge, R. Herault, C. Chatelain, F. Jardin, and R. Modzelewski. IODA: An Input Output Deep Architecture for image labeling. *Pattern Recognition*, 2015.
- [24] Xirong Li, Tiberio Uricchio, Lamberto Ballan, Marco Bertini, Cees G. M. Snoek, and Alberto Del Bimbo. Socializing the semantic gap: A comparative survey on image tag assignment, refinement, and retrieval. *ACM Comput. Surv.*, 49(1):14:1–14:39, 2016.
- [25] Shujie Liu, Nan Yang, Mu Li, and Ming Zhou. A recursive recurrent neural network for statistical machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1491–1500, Baltimore, Maryland, June 2014. Association for Computational Linguistics.
- [26] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 3431–3440, 2015.

- [27] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, and Ian J. Goodfellow. Adversarial autoencoders. *CoRR*, abs/1511.05644, 2015.
- [28] Volodymyr Mnih, Hugo Larochelle, and Geoffrey E. Hinton. Conditional restricted boltzmann machines for structured output prediction. In *UAI 2011, Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence, Barcelona, Spain, July 14-17, 2011*, pages 514–522, 2011.
- [29] Stéphane Nicolas, Thierry Paquet, and Laurent Heutte. A Markovian Approach for Handwritten Document Segmentation. In *ICPR (3)*, pages 292–295, 2006.
- [30] F. Ning, D. Delhomme, Yann LeCun, F. Piano, Léon Bottou, and Paolo Emilio Barbano. Toward automatic phenotyping of developing embryos from videos. *IEEE Trans. Image Processing*, 14(9):1360–1371, 2005.
- [31] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 1520–1528, 2015.
- [32] Keith Noto and Mark Craven. Learning Hidden Markov Models for Regression using Path Aggregation. *CoRR*, abs/1206.3275, 2012.
- [33] Franz Josef Och. Minimum error rate training in statistical machine translation. In *Proceedings of the ACL*, volume 1, 2003.
- [34] Lawrence Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.
- [35] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015 - 18th International Conference Munich, Germany, October 5 - 9, 2015, Proceedings, Part III*, pages 234–241, 2015.
- [36] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. A semi-automatic methodology for facial landmark annotation. In *CVPR Workshops*, pages 896–903, 2013.
- [37] H. Schmid. Part-of-speech tagging with neural networks. *conference on Computational linguistics*, 12:44–49, 1994.
- [38] Daniel Dominic Sleator and David Temperley. Parsing English with a Link Grammar. *CoRR*, 1995.
- [39] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *NIPS 2015*, pages 3483–3491, 2015.
- [40] Bruno Stuner, Clément Chatelain, and Thierry Paquet. Cohort of LSTM and lexicon verification for handwriting recognition with gigantic lexicon. *CoRR*, abs/1612.07528, 2016.
- [41] I. Sutskever, J. Martens, G. Dahl, and G. Hinton. On the importance of initialization and momentum in deep learning. In *ICML*, volume 28, pages 1139–1147, 2013.
- [42] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112, 2014.
- [43] U. Syed and G. Yona. Enzyme function prediction with interpretable models. *Computational Systems Biology. Humana press*, pages 373–420, 2009.
- [44] M. Szummer and Y. Qi. Contextual Recognition of Hand-drawn Diagrams with Conditional Random Fields. In *IWFHR*, pages 32–37, 2004.
- [45] G. Tsechpenakis, Jianhua Wang, B. Mayer, and D. Metaxas. Coupling CRFs and Deformable Models for 3D Medical Image Segmentation. In *ICCV*, pages 1–8, 2007.
- [46] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P. Manzagol. Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion. *JMLR*, 11:3371–3408, 2010.
- [47] H. Zen, K. Tokuda, and A. Black. Statistical parametric speech synthesis. *Speech Communication*, 51(11):1039–1064, 2009.

- [48] J. Zhang, S. Shan, M. Kan, and X. Chen. Coarse-to-Fine Auto-Encoder Networks (CFAN) for Real-Time Face Alignment. In *ECCV, Part II*, pages 1–16, 2014.
- [49] Yang Zhang, Philip David, and Boqing Gong. Curriculum domain adaptation for semantic segmentation of urban scenes. *CoRR*, abs/1707.09465, 2017.
- [50] Z. Zhang, P. Luo, C. C. Loy, and X. Tang. Facial landmark detection by deep multi-task learning. In *ECCV*, pages 94–108, 2014.

A.4 Pixel-wise Conditioned Generative Adversarial Networks for Image Synthesis and Completion

Reference

[Ruf+20] Cyprien Ruffino et al. “Pixel-Wise Conditioned Generative Adversarial Networks for Image Synthesis and Completion.” In: *Neurocomputing* (Apr. 2020). DOI: 10.1016/j.neucom.2019.11.116. arXiv: 2002.01281. URL: <https://hal.archives-ouvertes.fr/hal-02551730>

Pixel-wise Conditioned Generative Adversarial Networks for Image Synthesis and Completion

Cyprien Ruffino¹, Romain Hérault¹, Eric Laloy², Gilles Gasso¹

1- Normandie Univ, UNIROUEN, UNIHAVRE, INSA Rouen, LITIS
76 000 Rouen, France

2- Belgian Nuclear Research, Institute Environment, Health and Safety,
Boeretang 200 - BE-2400 Mol, Belgium

Abstract

Generative Adversarial Networks (GANs) have proven successful for unsupervised image generation. Several works have extended GANs to image inpainting by conditioning the generation with parts of the image to be reconstructed. Despite their success, these methods have limitations in settings where only a small subset of the image pixels is known beforehand. In this paper we investigate the effectiveness of conditioning GANs when very few pixel values are provided. We propose a modelling framework which results in adding an explicit cost term to the GAN objective function to enforce pixel-wise conditioning. We investigate the influence of this regularization term on the quality of the generated images and the fulfillment of the given pixel constraints. Using the recent PacGAN technique, we ensure that we keep diversity in the generated samples. Conducted experiments on FashionMNIST show that the regularization term effectively controls the trade-off between quality of the generated images and the conditioning. Experimental evaluation on the CIFAR-10 and CelebA datasets evidences that our method achieves accurate results both visually and quantitatively in term of Fréchet Inception Distance, while still enforcing the pixel conditioning. We also evaluate our method on a texture image generation task using fully-convolutional networks. As a final contribution, we apply the method to a classical geological simulation application.

Keywords: deep generative models, generative adversarial networks, conditional GAN

1. Introduction

Generative modelling is the process of modelling a distribution in a high-dimension space in a way that allows sampling in it. Generative Adversarial Networks (GANs) [1] have been the state of the art in unsupervised image generation for the past few years, being able to produce realistic images with high

resolution [2] without explicitly modelling the samples distribution. GANs learn a mapping function of vectors drawn from a low dimensional latent distribution (usually normal or uniform) to high dimensional ground truth images issued from an unknown and complex distribution. By using a discrimination function that distinguishes real images from generated ones, GANs setups a min max game able to approximate a Jensen-Shannon divergence between the distributions of the real samples and the generated ones.

Among extensions of GANs, Conditional GAN (CGAN) [3] attempts to condition the generation procedure on some supplementary information y (such as the label of the image x) by providing y to the generation and discrimination functions. CGAN enables a variety of conditioned generation, such as class-conditioned image generation [3], image-to-image translation [4, 5], or image inpainting [6]. On the other side, Ambient GAN [7] aims at training an unconditional generative model using only noisy or incomplete samples y . Relevant application domain is high-resolution imaging (CT scan, fMRI) where image sensing may be costly. Ambient GAN attempts to produce unaltered images \tilde{x} which distribution matches the true one without accessing to the original images x . For the sake, Ambient GAN considers lossy measurements such as blurred images, images with removed patch or removed pixels at random (up to 95%). Following this setup, Pajot et al.[8] extend the learning strategy to enable the reconstruction instead of the generation of realistic images from similarly altered samples.

In the spirit of Ambient GAN, we consider in this paper an extreme setting of image generation when only a few pixels, less than a percent of the image size, are known and are randomly scattered across the image (see Fig.1c). We refer to these conditioning pixels as a constraint map y . To reconstruct the missing information, we design a generative adversarial model able to generate high quality images coherent with given pixel values by leveraging on a training set of similar, but not paired images. The model we propose aims to match the distribution of the real images conditioned on a highly scarce constraint map, drawing connections with Ambient GAN while, in the same manner as CGAN, still allowing the generation of diverse samples following the underlying conditional distribution.

To make the generated images honoring the prescribed pixel values, we use a reconstruction loss measuring how close real constrained pixels are to their generated counterparts. We show that minimizing this loss is equivalent to maximizing the log-likelihood of the constraints given the generated image. Thereon we derive an objective function trading-off the adversarial loss of GAN and the reconstruction loss which acts as a regularization term. We analyze the influence of the related hyper-parameter in terms of quality of generated images and the respect of the constraints. Specifically, empirical evaluation on FashionMNIST [9] evidences that the regularization parameter allows for controlling the trade-off between samples quality and constraints fulfillment.

Additionally to show the effectiveness of our approach, we conduct experiments on CIFAR10 [10], CelebA [11] or texture [12] datasets using various deep architectures including fully convolutional network. We also evaluate our

method on a classical geological problem which consists of generating 2D geological images of which the spatial patterns are consistent with those found in a conceptual image of a binary fluvial aquifer[13][14]. Empirical findings reveal that the used architectures may lack stochasticity from the generated samples that is the GAN input is often mapped to the same output image irrespective of the variations in latent code [15]. We address this issue by resorting to the recent PacGAN [16] strategy. As a conclusion, our approach performs well both in terms of visual quality and respect of the pixel constraints while keeping diversity among generated samples. Evaluations on CIFAR-10 and CelebA show that the proposed generative model always outperforms the CGAN approach on the respect of the constraints and either come close or outperforms it on the visual quality of the generated samples.

The remainder of the paper is organized as follows. In Section 2, we review the relevant related work focusing first on generative adversarial networks, their conditioned version and then on methods dealing with image generation and reconstruction from highly altered training samples. Section 3 details the overall generative model we propose. In Section 4, we present the experimental protocol and evaluation measures while Section 5 gathers quantitative and qualitative effectiveness of our approach. The last section concludes the paper.

The contributions of the paper are summarized as follows:

- We propose a method for learning to generate images with a few pixel-wise constraints.
- A theoretical justification of the modelling framework is investigated.
- A controllable trade-off between the image quality and the constraints' fulfillment is highlighted,
- We showcase a lack of diversity in generating high-dimensional images which we solve by using PacGAN[16] technique. Several experiments allow to conclude that the proposed formulation can effectively generate diverse and high visual quality images while satisfying the pixel-wise constraints.

2. Image reconstruction with GAN in related works

The pursued objective of the paper is image generation using generative deep network conditioned on randomly scattered and scarce (less than a percent of the image size) pixel values. This kind of pixel constraints occurs in application domains where an image or signal need to be generated from very sparse measurements.

Before delving into the details, let introduce the notations and previous work related to the problem. We denote by $X \in \mathcal{X}$ a random variable and x its realization. Let p_X be the distribution of X over \mathcal{X} and $p_X(x)$ be its evaluation at x . Similarly $p_{X|Y}$ represents the distribution of X conditioned on the random variable $Y \in \mathcal{Y}$.

Given a set of images $x \in \mathcal{X} = [-1, 1]^{n \times p \times c}$ (see Figure 1a) drawn from an unknown distribution p_X and a sparse matrix $y \in \mathcal{Y} = [-1, 1]^{n \times p \times c}$ (Figure 1c) as the given constrained pixels, the problem consists in finding a generative

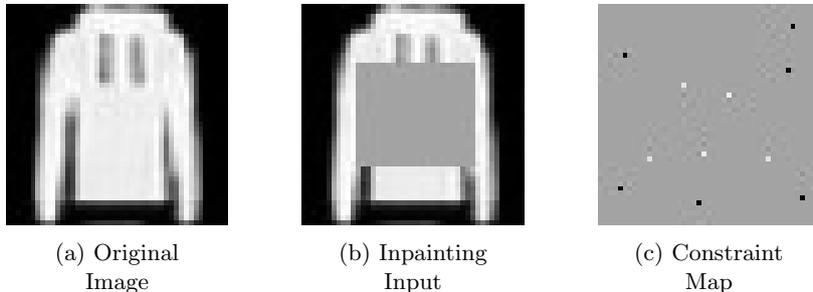


Figure 1: Difference between regular inpainting (1b) and the problem undertaken in this work (1c) on a real sample (1a).

model G with inputs z (a random vector sampled from a known distribution p_Z over the space \mathcal{Z}) and constrained pixel values $y \in [-1, 1]^{n \times p \times c}$ able to generate an image satisfying the constraints while likely following the distribution p_X (see Figure 3).

One of the state-of-the-art modelling framework for image generation is the Generative Adversarial Network. The seminal version of GAN [1] learns the generative models in an unsupervised way. It relies on a game between a generation function G and a discrimination network D , in which G learns to produce realistic samples while D learns to distinguish real examples from generated ones (Figure 2a). Training GANs amounts to find a Nash equilibrium to the following min-max problem,

$$\min_G \max_D L(D, G) = \mathbb{E}_{x \sim p_X} [\log(D(x))] + \mathbb{E}_{z \sim p_Z} [\log(1 - D(G(z)))] , \quad (1)$$

where p_Z is a known distribution, usually normal or uniform, from which the latent input z of G is drawn, and p_X is the distribution of the real images.

Among several applications, the GANs was adapted to image inpainting task (Figure 1b). For instance Yeh et al. [17] propose an inpainting approach which considers a pre-trained generator, and explores its latent space \mathcal{Z} through an optimization procedure to find a latent vector z , which induces an image with missing regions filled in by conditioning on the surroundings available information. However, the method requires to solve a full optimization problem at inference stage, which is computationally expensive.

Other approaches (Figure 2) rely on Conditional variant of GAN (CGAN) [3] in which additional information y is provided to the generator and the discriminator (see Figure 2b). This leads to the following optimization problem adapted to CGAN

$$\min_G \max_D L(D, G) = \mathbb{E}_{\substack{x \sim p_X \\ y \sim p_{Y|X}}} [\log(D(x, y))] + \mathbb{E}_{\substack{z \sim p_Z \\ y \sim p_Y}} [\log(1 - D(G(y, z), y))] . \quad (2)$$

Although CGAN was initially designed for class-conditioned image generation by setting y as the class label of the image, several types of conditioning

information can apply such as a full image for image-to-image translation [4] or partial image as in inpainting [18]. CGAN-based inpainting methods rely on generating a patch that will fill up a structured missing part of the image and achieve impressive results. However they are not well suited to reconstruct very sparse and unstructured signal [19]. Additionally, these approaches learn to reconstruct a single sample instead of a full distribution, implying that there is no sampling process for a given constraint map or highly degraded image.

AmbientGAN [7] (Figure 2c) trains a generative model capable to yield full images from only lossy measurements. One of the image degradations considered in this approach is the random removal of pixels leading to sparse pixel map y . It is simulated with a differentiable function f_θ whose parameter θ indicates the pixels to be removed. The underlying optimization problem solved by AmbientGAN is therefore stated as

$$\min_G \max_D L(D, G) = \mathbb{E}_{y \sim p_Y} \left[\log(D(y)) \right] + \mathbb{E}_{\substack{z \sim p_Z \\ \theta \sim p_\theta}} \left[\log(1 - D(f_\theta(G(z)))) \right]. \quad (3)$$

Pajot et al. [8] combined the AmbientGAN approach with an additional reconstruction task that consists in reconstructing the $f_\theta(G(y))$ from the twice-altered image $\tilde{y} = f_\theta(G(y))$ and $\hat{y} = f_\theta(G(f_\theta(G(y))))$,

$$\min_G \max_D L(D, G) = \mathbb{E}_{y \sim p_Y} \left[\log(D(y)) \right] + \mathbb{E}_{y \sim p_Y} \left[\log(1 - D(\hat{y})) \right] + \|\hat{y} - \tilde{y}\|_2^2. \quad (4)$$

The ℓ_2 norm term ensures that the generator is able to learn to revert f_θ i.e. to revert the alteration process on a given sample. This allows the reconstruction of realistic image only from a given constraint map y . However the reconstruction process is deterministic and does not provide a sampling mechanism.

Compressed Sensing with Meta-Learning [20] is an approach that combines the exploration of the latent space \mathcal{Z} to recover images from lossy measurements with the enforcing of the Restricted Isometric Property [21], which states that for two samples $x_1, x_2 \sim p_X$,

$$(1 - \alpha)\|x_1 - x_2\|_2^2 \leq \|f_\theta(x_1 - x_2)\|_2^2 \leq (1 + \alpha)\|x_1 - x_2\|_2^2$$

where α is a small constant. It replaces the adversarial training of the generative model G (Eq. 1) by searching, for a given degraded image y , a vector \hat{z} such that $\hat{y} = f_\theta(G(\hat{z}))$ minimizes the ℓ_2 distance between y and \hat{y} while still enforcing the RIP. The overall problem induced by this approach can be formulated as:

$$\min_G L(G) = \mathbb{E}_{\substack{x \sim p_X \\ y \sim p_Y \\ z \sim p_Z}} \left(\sum_{\substack{x_1, x_2 \in \mathcal{S} \\ x_1 \neq x_2}} (\|f_\theta(x_1 - x_2)\|_2^2 - \|x_1 - x_2\|_2^2)^2 \right) / 3 + \|y - f_\theta(G(\hat{z}))\|_2^2$$

where $\hat{z} = \min_z \|y - f_\theta(G(z))\|_2^2$. (5)

where \mathcal{S} contains the three samples $x, G(z), G(\hat{z})$. In practice, \hat{z} is computed with gradient descent on z by minimizing $\|y - f_\theta(G(z))\|_2^2$, and starting from a

random $z \sim p_Z$. As a benefit, this approach may generate an image $\hat{x} = G(\hat{z})$ from a noisy information y but at a high computation burden since it requires to solve an optimization problem (computing \hat{z}) at inference stage for generating an image.

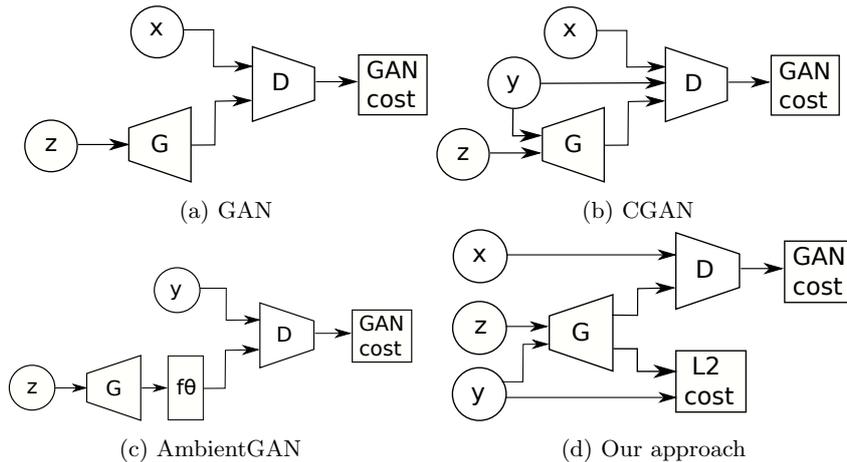


Figure 2: Different GAN Setups. G and D are the generator and discriminator networks, x and z are samples from the distributions P_x and P_r , y is a label/constraint map sampled from P_y and f_θ is an image degradation function.

3. Proposed approach

Let introduce the formal formulation of the addressed problem. Assume y is the given set of constrained pixel values. To ease the presentation, let consider y as a $n \times p \times c$ image with only a few available pixels (less than 1% of $n \times p \times c$). We will also encode the spatial location of these pixels using a corresponding binary mask $M(y) \in \{0, 1\}^{n \times p \times c}$. We intend to learn a GAN whose generation network takes as input the constraint map y and the sampled latent code $z \in \mathcal{Z}$ and outputs a realistic image that fulfills the prescribed pixel values. Within this setup, the generative model can sample from the unknown distribution p_X of the training images $\{x_1, \dots, x_N\}$ while satisfying unseen pixel-wise constraints at training stage. Formally our proposed GAN can be formulated as

$$\min_G \max_D L(D, G) = \mathbb{E}_{x \sim p_x} \left[\log(D(x)) \right] + \mathbb{E}_{\substack{z \sim p_Z \\ y \sim p_Y}} \left[\log(1 - D(G(y, z))) \right], \quad (6)$$

$$\text{s.t. } y = M(y) \odot G(y, z)$$

where \odot stands for the Hadamard (or point-wise) product and $M(y)$ for the mask, a sparse matrix with entries equal to one at constrained pixels location.

As the equality constraint in Problem (6) is difficult to enforce during training, we rather investigate a relaxed version of the problems. Following Pajot et

al. [8] we assume that the constraint map is obtained through a noisy measurement process

$$y = f_M(x) + \varepsilon . \quad (7)$$

Here f_M is the masking operator yielding to $y = M(y) \odot x$. Also the constrained pixels are randomly and independently selected. ε represents an additive i.i.d noise corrupting the pixels. Therefore we can formulate the Maximum A Posteriori (MAP) estimation problem, which, given the constraint map y , consists in finding the most probable image x^* following the posterior distribution $p_{X|Y}$,

$$x^* = \arg \max_x \log p_{X|Y}(x|y) \quad (8)$$

$$= \arg \max_x \log p_{Y|X}(y|x) + \log p_X(x) . \quad (9)$$

$p_{Y|X}(y|x)$ is the likelihood that the constrained pixels y are issued from image x while $p_X(x)$ represents the prior probability at x . Assuming that the generation network G may sample the most probable image $G(y, z)$ complying with the given pixel values y , we get the following problem

$$G^* = \arg \max_G \mathbb{E}_{\substack{y \sim p_Y \\ z \sim p_Z}} \log p_{Y|X}(y|G(y, z)) + \log p_X(G(y, z)) . \quad (10)$$

The first term in Problem (10) measures the likelihood of the constraints given a generated image. Let rewrite Equation (7) as $\text{vect}(y) = \text{vect}(f_M(x)) + \text{vect}(\varepsilon)$ where $\text{vect}(\cdot)$ is the vectorisation operator that consists in stacking the constrained pixels. Therefore, assuming $\text{vect}(\varepsilon)$ is an i.i.d Gaussian noise with distribution $\mathcal{N}(0, \sigma^2 I)$, we achieve the expression of the conditional likelihood

$$\log p_{Y|X}(y|G(y, z)) \propto - \|\text{vect}(y) - \text{vect}(M(y) \odot G(y, z))\|_2^2 \quad (11)$$

which evaluates the quadratic distance between the conditioning pixels and their predictions by G . In other words, using a matrix notation of (7), the likelihood of the constraints given a generated image equivalently writes

$$\log p_{Y|X}(y|G(y, z)) \propto - \|y - M(y) \odot G(y, z)\|_F^2 . \quad (12)$$

$\|A\|_F^2$ represents the squared Frobenius norm of matrix A that is the sum of its squared entries.

The second term in Problem (10) is the likelihood of the generated image under the true but unknown data distribution p_X . Maximizing this term can be equivalently achieved by minimizing the distance between p_X and the marginal distribution of the generated samples $G(y, z)$. This amounts to minimizing with respect to G , the GAN-like objective function $\mathbb{E}_{x \sim p_X} \log(D(x)) + \mathbb{E}_{\substack{z \sim p_Z \\ y \sim p_Y}} \log(1 - D(G(y, z)))$ [1]. Putting altogether these elements, we can propose a relaxation of the hard constraint optimization problem (6) (Figure 2d) as follows

$$\begin{aligned} \min_G \max_D L(D, G) &= \mathbb{E}_{x \sim p_X} \left[\log(D(x)) \right] \\ &+ \mathbb{E}_{\substack{z \sim p_Z \\ y \sim p_Y}} \left[\log(1 - D(G(y, z))) + \lambda \|y - M(y) \odot G(y, z)\|_F^2 \right] . \end{aligned} \quad (13)$$

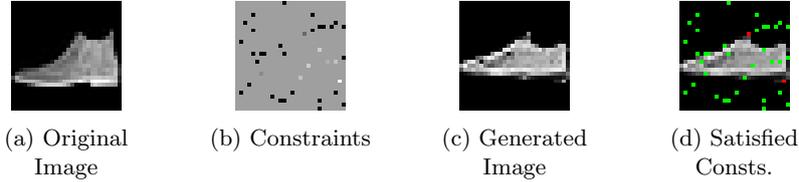


Figure 3: Generation of a sample during training. We first sample an image from a training set (3a) and we sample the constraints (3b) from it. Then our GAN generates a sample (3c). The constraints with squared error smaller than $\epsilon = 0.1$ are deemed satisfied and shown by green pixels in (3d) while the red pixels are unsatisfied.

Remarks:

- The assumption of Gaussian noise measurement leads us to explicitly turn the pixel value constraints into the minimization of the ℓ_2 norm between the real enforced pixel values and their generated counterparts (see Figure 2d).
- This additional term acts as a regularization over prescribed pixels by the mask $M(y)$. The trade-off between the distribution matching loss and the constraint enforcement is assessed by the regularization parameter $\lambda \geq 0$.
- It is worth noting that the noise ε can be of any other distribution, according to the prior information, one may associate to the measurement process. We only require this distribution to admit a closed-form solution for the maximum likelihood estimation for optimization purpose. Typical choices are distributions from the exponential family [22].

To solve Problem (13), we use the stochastic gradient descent method. The overall training procedure is detailed in Algorithm 1 and ends up when a maximal number of training epochs is attained.

When implementing this training procedure we experienced, at inference stage, a lack of diversity in the generated samples (see Figure 5) with deeper architectures, most notably the encoder-decoder architectures. This issue manifests itself through the fact that the learned generation network, given a constraint map y , outputs almost deterministic image regardless the variations in the input z . The issue was also pointed out by Yang et al. [15] as characteristic of CGANs.

To avoid the problem, we exploit the recent PacGAN [16] technique: it consists in passing a set of samples to the discrimination function instead of a single one. PacGAN is intended to tackle the mode collapse problem in GAN training. The underlying principle being that if a set of images are sampled from the same training set, they are very likely to be completely different, whereas if the generator experiences mode collapse, generated images are likely to be similar. In practice, we only give two samples to the discriminator, which is sufficient to overcome the loss of diversity as suggested in [16]. The resulting training procedure is summarized in Algorithm 2.

Algorithm 1 Proposed training algorithm

Require: \mathcal{D}_X the set of unaltered images, \mathcal{D}_Y the set of constraint maps, G the generation network, and D the discrimination function

repeat

sample a mini-batch $\{x_i\}_{i=1}^m$ from \mathcal{D}_X

sample a mini-batch $\{y_i\}_{i=1}^m$ from \mathcal{D}_Y

sample a mini-batch $\{z_i\}_{i=1}^m$ from distribution p_Z

update D by stochastic gradient ascent of

$$\sum_{i=1}^m \log(D(x_i)) + \log(1 - D(G(y_i, z_i)))$$

sample a mini-batch $\{y_j\}_{j=1}^n$ from \mathcal{D}_Y

sample a a mini-batch $\{z_j\}_{j=1}^n$ from distribution p_Z ;

update G by stochastic gradient descent of

$$\sum_{j=1}^n \log(1 - D(G(y_j, z_j))) + \|y_j - M(y_j) \odot G(y_j, z_j)\|_F^2$$

until a stopping condition is met

Algorithm 2 Our training algorithm including PacGAN

Require: \mathcal{D}_X the set of unaltered images, \mathcal{D}_Y the set of constraint maps, G the generation network, and D the discrimination function

repeat

sample two mini-batches $\{x_i^a\}_{i=1}^m, \{x_i^b\}_{i=1}^m$ from \mathcal{D}_X

sample a mini-batch $\{y_i\}_{i=1}^m$ from \mathcal{D}_Y

sample two mini-batches $\{z_i^a\}_{i=1}^m, \{z_i^b\}_{i=1}^m$ from distribution p_Z

update D by stochastic gradient ascent of

$$\sum_{i=1}^m \log(D(x_i^a, x_i^b)) + \log(1 - D(G(y_i, z_i^a), G(y_i, z_i^b)))$$

sample a mini-batch $\{y_j\}_{j=1}^n$ from \mathcal{D}_Y

sample two mini-batches $\{z_j^a\}_{j=1}^n, \{z_j^b\}_{j=1}^n$ from distribution p_Z

update G by stochastic gradient descent of

$$\sum_{j=1}^n \log(1 - D(G(y_j, z_j^a), G(y_j, z_j^b))) + \|y_j - M(y_j) \odot G(y_j, z_j^a)\|_F^2$$

until a stopping condition is met

4. Experiments

We have conducted a series of empirical evaluation to assess the performances of the proposed GAN. Used datasets, evaluation protocol and the tested deep architectures are detailed in this section while Section 5 is devoted to the results presentation.

4.1. Datasets

We tested our approach on several datasets listed hereafter. Detailed information on these datasets are provided in the Appendix A.

FashionMNIST [9] consists of 60,000 28×28 small grayscale images of fashion items, split in 10 classes and is a harder version of the classical MNIST dataset [23]. The very small size of the images makes them particularly appropriate for large-scale experiments, such as hyper-parameter tuning.

CIFAR10 [10] consists of 60,000 32×32 colour images of 10 different and varied classes. It is deemed less easy than MNIST and FashionMnist

CelebA[11] is a large dataset of celebrity portraits labeled by identity and a variety of binary features such as eyeglasses, smiling... We use 100,000 images cropped to a size of 128×128 , making this dataset appropriate for a high dimension evaluation of our approach in comparison with related work.

Texture is a custom dataset composed of 20,000 160×160 patches sampled from a large brick wall texture, as recommended in [12]. It is worth noting that this procedure can be reproduced on any texture image of sufficient size. Texture is a testbed of our approach on fully-convolutional networks for constrained texture generation task.

Subsurface is a classical dataset in geological simulation [13] which consists, similarly to the Texture dataset, of 20,000 160×160 patches sampled from a model of a subsurface binary domain. These models are assumed to have the same properties as a texture, mainly the property of global ergodicity of the data.

To avoid learning explicit pairing of real images seen by the discrimination function with constraint maps provided to the generative network, we split each dataset into training, validation and test sets, to which we add a set composed of constraint maps that should remain unrelated to the three others. In order to do so, a fifth of each set is used to generate the constrained pixel map y by randomly selecting 0.5% of the pixels from a uniform distribution, composing a set of constraints for each of the train, test and validation sets. The images from which these maps are sampled are then removed from the training, testing and validation sets. For each carried experiment the best model is selected based on some performance measures (see Section 4.3) computed on the validation set, as in the standard of machine learning methodology [24]. Finally, reported results are computed on the test set.

4.2. Network architectures

We use a variety of GAN architectures in order to adapt to the different scales and image sizes of our datasets. The detailed configuration of these architectures are exposed in Appendix B.

For the experiments on the FashionMNIST [9], we use a lightweight network for both the discriminator and the generator similarly to DCGAN [25] due to the small resolution of FashionMnist images.

To experiment on the Texture dataset, we consider a set of fully-convolutional generator architectures based on either dilated convolutions [26], which behave well on texture datasets [27], or encoder-decoder architectures that are commonly used in domain-transfer applications such as CycleGAN [28]. We selected these architectures because they have very large receptive fields without using pooling, which allow the generator to use a large context for each pixel.

We keep the same discriminator across all the experiments with these architectures, the PatchGAN discriminator [4], which is a five-layer fully-convolutional network with a sigmoid activation.

The Up-Dil architecture consists in a set of transposed convolutions (the upscaling part), and a set of dilated convolutional layers [26], while the Up-EncDec has an upscaling part followed by an encoder-decoder section with skip-connections, where the constraints are downscaled, concatenated to the noise, and re-upscaled to the output size.

The UNet [29] architecture is an encoder-decoder where skip-connections are added between the encoder and the decoder. The Res architecture is an encoder-decoder where residual blocks [30] are added after the noise is concatenated to the features. The UNet-Res combines the UNet and the Res architectures by including both residual blocks and skip-connections.

Finally, we will evaluate our approach on the Subsurface dataset using the architecture that yields to the best performances on the Texture dataset.

4.3. Evaluation

We evaluate our approach based on both the satisfaction of the pixel constraints and the visual quality of sampled images. From the assumption of Gaussian measurement noise (as discussed in Section 3), we assess the constraint fulfillment using the following mean square error (MSE)

$$MSE = \frac{1}{L} \sum_{i=1}^L \|y_i - M(y_i) \odot G(y_i, z_i)\|_F^2 \quad (14)$$

This metric should be understood as the mean squared error of reconstructing the constrained pixel values.

Visual quality evaluation of an image is not a trivial task [31]. However, Fréchet Inception Distance (FID) [32] and Inception Score [33], have been used to evaluate the performance of generative models. We employ FID since the Inception Score has been shown to be less reliable [34]. The FID consists in computing a distance between the distributions of relevant features extracted from generated and real samples. To extract these features, a pre-trained Inception v3 [35] classifier is used to compute the embeddings of the images at a chosen layer. Assuming these embeddings shall follow a normal distribution, the quality of the generated images is assessed in term of a Wasserstein-2 distance between the distribution of real samples and generated ones. Hence the FID writes

$$FID = \|\mu_r - \mu_g\|^2 + Tr(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2}), \quad (15)$$

where Tr is the trace operator, (μ_r, Σ_r) and (μ_g, Σ_g) are the pairs of mean vector and covariance matrix of embeddings obtained on respectively the real and the generated data. Being a distance between distributions, a small FID corresponds to a good matching of the distributions.

Since the FID requires a pre-trained classifier adapted to the dataset in study, we trained simple convolutional neural networks as classifiers for the FashionMNIST and the CIFAR-10 datasets. For the Texture dataset, since the dataset is not labeled, we resort to a CNN classifier trained on the Describable Textures Dataset (DTD) [36], which is a related application domain.

However, since we do not have labels for the Subsurface dataset, we could not train a classifier for this dataset, thus we cannot compute the FID. To evaluate the quality of the generated samples, we use metrics based on a distance between feature descriptors extracted from real samples and generated ones. Similarly to [27], we rely on a χ^2 distance between the Histograms of Oriented Gradients (HOG) or Local Binary Patterns (LBP) features computed on generated and real images.

Histograms of Oriented Gradients (HOG) [37] and Local Binary Patterns (LBP) [38] are computed by splitting an image into cells of a given radius and computing on each cell the histograms of the oriented gradients for HOGs and of the light level differences for each pixel to the center of the cell for LBPs. Additionally, we consider the domain-specific metric, the connectivity function [39] which is presented in Appendix C.

Finally, we check by visual inspection if the trained model G is able to generate diverse samples, meaning that for a given y and for a set of latent codes $(z_1, \dots, z_n) \sim p_Z$, the generated samples $G(y, z_1), \dots, G(y, z_n)$ are visually different.

5. Experimental results

5.1. Quality-fidelity trade-off

We first study the influence of the λ regularization hyper-parameter on both the quality of the generated samples and the respect of the constraints. We experiment on the FashionMNIST [9] dataset, since such a study requires intensive simulations permitted by the low resolution of FashionMnist images and the used architectures (see Section 4.2).

To overcome classical GANs instability, the networks are trained 10 times and the median values of the best scores on the test set at the best epoch are recorded. The epoch that minimizes:

$$\sqrt{\left(\frac{FID - FID_{min}}{FID_{max} - FID_{min}}\right)^2 + \left(\frac{MSE - MSE_{min}}{MSE_{max} - MSE_{min}}\right)^2}$$

on the validation set is considered as the best epoch, where FID_{min} , MSE_{min} , FID_{max} and MSE_{max} are respectively the lowest and highest FIDs and MSEs obtained on the validation set.

Empirical evidences (highlighted in Figure 4) show that with a good choice of λ , the regularization term helps the generator to enforce the constraints, leading to smaller MSEs than when using the CGAN ($\lambda = 0$) without compromising on the quality of generated images. Also, we can note that using the regularization

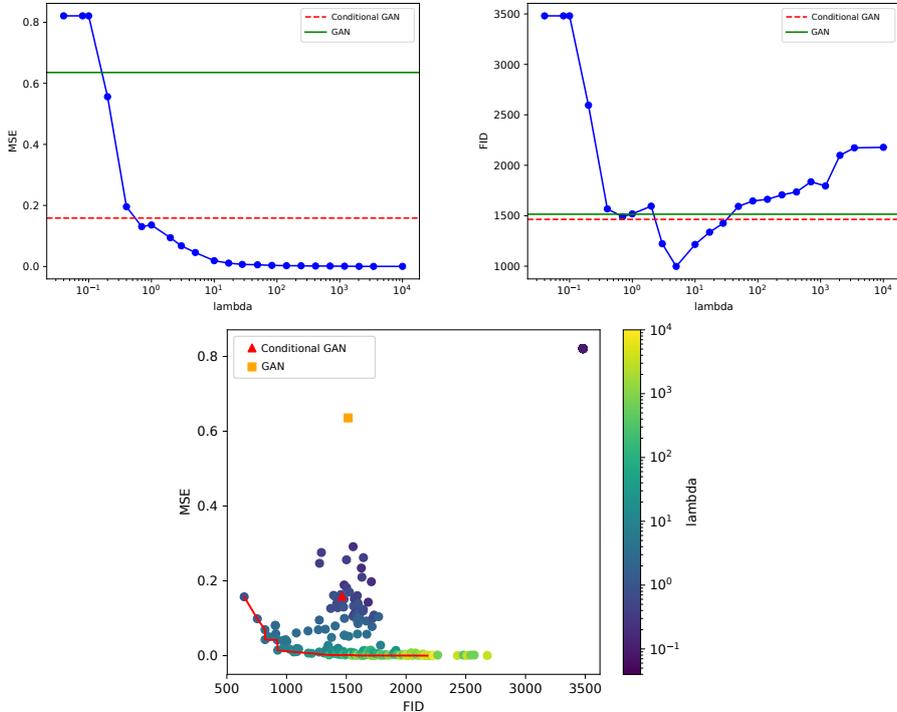


Figure 4: Our approach compared to the GAN and CGAN baselines. MSE (left) and FID (right) w.r.t. the regularization parameter λ , MSE w.r.t the FID (bottom).

term even leads to a better image quality compared to GAN and CGAN. The bottom panel in Figure 4 illustrates that the trade-off between image quality and the satisfaction of the constraints can be controlled by appropriately setting the value of λ . Nevertheless, for small values of λ (less or equal to 10^{-1}), our GAN model fails to learn meaningful distribution of the training images and only generates uniformly black images. This leads to the plateaus on the MSE and FID plots (top panels in Figure 4).

5.2. Texture generation with fully-convolutional architectures

Fully-convolutional architectures for GANs are widely used, either for domain-transfer applications [28][4] or for texture generation [12]. In order to evaluate the efficiency of our method on relatively high resolution images, we experiment the fully-convolutional networks described in Section 4.2 on a texture generation task using Texture dataset. We investigate the upscaling-dilatation network, the encoder-decoder one and the resnet-like architectures.

Our training algorithm was run for 40 epochs on all reported results. We provide a comparison to CGAN[3] approach by using the selected best architectures. The models are evaluated in terms of best FID (visual quality of sampled

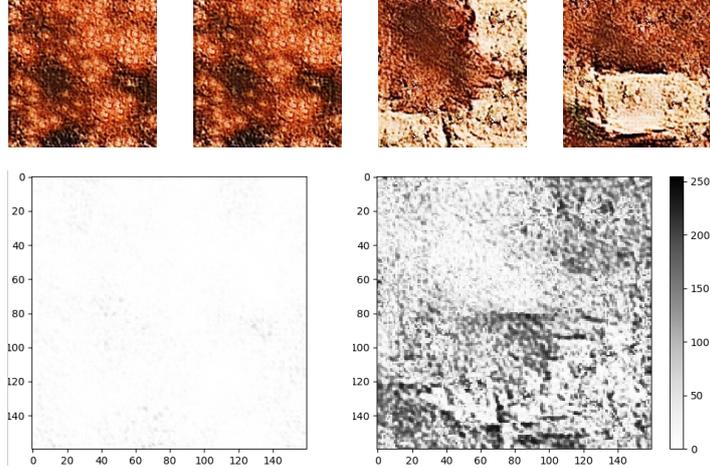


Figure 5: An example of a loss of diversity when generating Texture samples with a trained UNetRes network using two different random noises z and a single constraint map y . The two samples on the top left are generated using the classical GAN discriminator whereas the samples on the top right are generated using the PacGAN approach. The loss of diversity is clearly visible on the absolute differences between the greyscaled images (bottom).

images) at each epoch and MSE (conditioning on fixed pixel values). We also compute the FID score of the models at the epochs where the MSE is the lowest. In the other way around, the MSE is reported at epoch when the FID is the lowest. The obtained quantitative results are detailed in Table 1.

For the encoder-decoder models, we can notice that the models using ResNet blocks perform better than just using a UNet generator. A trade-off can also be seen between the FID and MSE for the ResNet models and the UNet-ResNet, which could mean that skip-connections help the generator to fulfill the constraints but at the price of lowered visual quality.

Although the encoder-decoder models perform the best, they tend to lose diversity in the generated samples (see Figure 5), whereas the upscaling-based models have high FID and MSE but naturally preserve diversity in the generated samples.

Changing the discriminator for a PacGAN discriminator with 2 samples in the encoder-decoder based architectures allows to restore diversity, while keeping the same performances as previously or even increasing the performances for the UNetRes (see Table 1).

Table 2 compares our proposed approach to CGAN using fully convolutional networks. It shows that our approach is more able to comply with the pixel constraints while producing realistic images. Indeed, our approach outperforms CGAN (see Table 2) by a large margin on the respect of conditioning pixels (see the achieved MSE metrics by our UNetPAC or UNetResPAC) and gets close FID performance on the generated samples. This finding is in accordance of the obtained results on FashionMnist experiments.

Model	Best FID	Best MSE	FID at best MSE	MSE at best FID	Diversity
Up-Dil	0.0949	0.4137	1.0360	0.7057	✓
Up-EncDec	0.1509	0.7570	0.2498	0.9809	✓
UNet	0.0442	0.1789	0.0964	0.4559	✗
Res	0.0458	0.0474	0.0590	0.0476	✗
UNetRes	0.0382	0.0307	0.0499	0.0338	✗
ResPAC	0.0350	0.0698	0.0466	0.4896	✓
UNetPAC	0.0672	\leq 0.0001	0.3120	0.2171	✓
UNetResPAC	0.0431	0.0277	0.0447	0.0302	✓

Table 1: Results obtained by the different fully-convolutional architectures on the Texture dataset. We can remark that the encoder-decoder greatly outperforms the upscaling ones and that using the PacGAN technique helps keeping the performance of these models while restoring the diversity in the samples. The bottom part of the table refers to PacGAN architectures.

Model	Best FID	Best MSE	FID at best MSE	MSE at best FID
CGAN-ResPAC	0.0234	0.1337	0.0340	0.2951
CGAN-UNetPAC	0.0518	0.2010	0.0705	0.4828
CGAN-UNetResPAC	0.0428	0.1060	0.0586	0.2250
Ours-ResPAC	0.0350	0.0698	0.0466	0.4896
Ours-UNetPAC	0.0672	\leq 0.0001	0.3120	0.2171
Ours-UNetResPAC	0.0431	0.0277	0.0447	0.0302

Table 2: Results obtained by the selected best fully-convolutional architectures on the Texture dataset for both the CGAN approach and our approach.

5.3. Extended architectures

We extend the comparison of our approach to CGAN on the CIFAR10 and CelebA datasets (Table 3). We investigated the architectures described in Section 4.2. All reported results are obtained with the regularization parameter fixed to $\lambda = 1$. We train the networks for 150 epochs using the same dataset split as stated previously in order to keep independence between the images constraint maps. The evaluation procedure remains also unchanged. We use the PacGAN approach to avoid the loss of diversity issues. The experiments on both datasets show that though CGAN provides better results in terms of visual quality, our approach outperforms it according to the respect of the pixel constraints.

5.4. Application to hydro-geology

Finally, we evaluate our approach on the Subsurface dataset. We use the UNetResPAC architecture, since it performed the best on Texture data as exposed in Section 5.2. As previously, we simply set the regularization parameter at $\lambda = 1$ and, the network is trained for 40 epochs using the same experimental

	Model	Best FID	Best MSE	FID at best MSE	MSE at best FID
CIFAR-10	CGAN	2,68	0.081	2.68	0.081
	Ours	3.120	0.010	3.530	0.011
CelebA	CGAN	1.34e-4	0.0209	1.81e-4	0.0450
	Ours	2.09e-4	0.0053	5.392e-4	0.0249

Table 3: Results on the CIFAR10 and CelebA datasets. The reported performances compare CGAN to our proposed GAN conditioned on scarce constraint map.

	Model	Best HOG	Best MSE	HOG at best MSE	MSE at best HOG
Subsurface	CGAN	2.92e-4	0.2505	3.06e-4	1.1550
	Ours	4.31e-4	0.0325	5.69e-4	0.2853

Table 4: Evaluation of the trade-off between the visual quality of the generated samples and the respect of the constraints for the CGAN approach and ours on the Subsurface dataset.

protocol. To evaluate the trade-off between the visual quality and the respect of the constraints, instead of FID we rather compute distances between visual Histograms of Oriented Gradients (see Section 4), extracted from real and generated samples. We also evaluate the visual quality of our approach with a distance between Local Binary Patterns. Indeed, Subsurface application lacks labelled data in order to learn a deep network classifier from which the FID score can be computed.

The obtained results are summarized in Tables 4 and 5. They are coherent with the previous experiments since the generated samples are diverse and have a low error regarding the constrained pixels. The conditioning have a limited impact on the visual quality of the generated samples and compares well to unconditional approaches [27]. Evaluation of the generated images using the domain-connectivity function highlights this fact on Figures 7 and 7 in the supplementary materials. Also examples of generated images by our approach pictured in Figure 9 (see appendix D) show that we preserve the visual quality and honor the constraints.

	Model	Best HOG	Best MSE	Best LBP (radius=1)	Best LBP (radius=2)
Subsurface	CGAN	2.92e-4	0.2505	2.157	3.494
	Ours	4.31e-4	0.0325	10.142	16.754

Table 5: Evaluation of the visual quality between the CGAN approach and ours on the Subsurface dataset using several metrics.

Conclusion

In this paper, we address the task of learning effective generative adversarial networks when only very few pixel values are known beforehand. To solve this pixel-wise conditioned GAN, we model the conditioning information under a probabilistic framework. This leads to the maximization of the likelihood of the constraints given a generated image. Under the assumption of a Gaussian distribution over the given pixels, we formulate an objective function composed of the conditional GAN loss function regularized by a ℓ_2 -norm on pixel reconstruction errors. We describe the related optimization algorithm.

Empirical evidences illustrate that the proposed framework helps obtaining good image quality while best fulfilling the constraints compared to classical GAN approaches. We show that, if we include the PacGAN technique, this approach is compatible with fully-convolutional architectures and scales well to large images. We apply this approach to a common geological simulation task and show that it allows the generation of realistic samples which fulfill the prescribed constraints.

In future work, we plan to investigate other prior distributions for the given pixels as the Laplacian or β -distributions. We are also interested in applying the developed approach to other applications or signals such as audio inpainting [40].

Acknowledgements

This research was supported by the CNRS PEPS I3A REGGAN project and the ANR-16-CE23-0006 grant *Deep in France*. We kindly thank the CRIANN for the provided high-computation facilities.

References

- [1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.
- [2] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.
- [3] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.
- [4] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.
- [5] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.

- [6] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016.
- [7] Ashish Bora, Eric Price, and Alexandros G Dimakis. Ambientgan: Generative models from lossy measurements. In *International Conference on Learning Representations (ICLR)*, 2018.
- [8] Arthur Pajot, Emmanuel de Bezenac, and Patrick Gallinari. Unsupervised adversarial image reconstruction. In *International Conference on Learning Representations*, 2019.
- [9] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.
- [10] Alex Krizhevsky. Learning multiple layers of features from tiny images. In *Technical report, University of Toronto*, 2009.
- [11] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015.
- [12] Nikolay Jetchev, Urs Bergmann, and Roland Vollgraf. Texture synthesis with spatial generative adversarial networks. *arXiv preprint arXiv:1611.08207*, 2016.
- [13] Sebastien Strebelle. Conditional simulation of complex geological structures using multiple-point statistics. *Mathematical Geology*, 34(1):1–21, Jan 2002.
- [14] Eric Laloy, Romain Hérault, Diederik Jacques, and Niklas Linde. Training-image based geostatistical inversion using a spatial generative adversarial neural network. *Water Resources Research*, 54(1):381–406, 2018.
- [15] Dingdong Yang, Seunghoon Hong, Yunseok Jang, Tiangchen Zhao, and Honglak Lee. Diversity-sensitive conditional generative adversarial networks. In *International Conference on Learning Representations*, 2019.
- [16] Zinan Lin, Ashish Khetan, Giulia Fanti, and Sewoong Oh. Pacgan: The power of two samples in generative adversarial networks. In *Advances in Neural Information Processing Systems*, pages 1498–1507, 2018.
- [17] Raymond A Yeh, Chen Chen, Teck-Yian Lim, Alexander G Schwing, Mark Hasegawa-Johnson, and Minh N Do. Semantic image inpainting with deep generative models. In *CVPR*, volume 2, page 4, 2017.
- [18] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. *arXiv preprint arXiv:1801.07892*, 2018.
- [19] Ugur Demir and Gozde Unal. Patch-based image inpainting with generative adversarial networks. *arXiv preprint arXiv:1803.07422*, 2018.
- [20] Yan Wu, Mihaela Rosca, and Timothy Lillicrap. Deep compressed sensing. In *Proceedings of the 36th International Conference on Machine Learning*, 2019.
- [21] Emmanuel J. Candes and Terrence Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 51(12):4203–4215, Dec 2005.
- [22] Lawrence D Brown. *Fundamentals of statistical exponential families: with applications in statistical decision theory*. Ims, 1986.
- [23] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

- [24] Luca Oneto. *Model Selection and Error Estimation in a Nutshell*, pages 25–31. Springer International Publishing, Cham, 2020.
- [25] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.
- [26] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.
- [27] Cyprien Ruffino, Romain Hérault, Eric Laloy, and Gilles Gasso. Dilated spatial generative adversarial networks for ergodic image generation. In *Conférence sur l'Apprentissage*, 2018.
- [28] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.
- [29] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
- [30] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [31] Lucas Theis, Aäron van den Oord, and Matthias Bethge. A note on the evaluation of generative models. *arXiv preprint arXiv:1511.01844*, 2015.
- [32] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, pages 6626–6637, 2017.
- [33] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, pages 2234–2242, 2016.
- [34] Shane Barratt and Rishi Sharma. A note on the inception score. *arXiv preprint arXiv:1801.01973*, 2018.
- [35] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.
- [36] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, , and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.
- [37] Navneet Dalal and Bill Triggs. Histograms of Oriented Gradients for Human Detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.
- [38] Matti Pietikäinen, Abdenour Hadid, Guoying Zhao, and Timo Ahonen. *Computer Vision Using Local Binary Patterns*, volume 40 of *Computational Imaging and Vision*. Springer London, London, 2011.
- [39] Laurent Lemmens, Bart Rogiers, Mieke De Craen, Eric Laloy, Diederik. Jacques, Marijeke Huysmans, and al. Effective structural descriptors for natural and engineered radioactive waste confinement barrier, 2017.

- [40] Andrés Marafioti, Nathanaël Perraudin, Nicki Holighaus, and Piotr Majdak. A context encoder for audio inpainting. *arXiv preprint arXiv:1810.12138*, 2018.
- [41] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.
- [42] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

Appendices

A. Details of the datasets

Dataset	Size (in pixels)	Training set	Validation set	Test set
FashionMNIST	28x28	55,000	5,000	10,000
Cifar-10	32x32	55,000	5,000	10,000
CelebA	128x128	80,000	5,000	15,000
Texture	160x160	20,000	2,000	4,000
Subsurface	160x160	20,000	2,000	4,000

Additional information:

- For FashionMNIST and Cifar-10, we keep the original train/test split and then sample 5000 images from the training set that act as validation samples.
- For the Texture dataset, we sample patches randomly from a 3840x2400 image of a brick wall.

B. Detailed deep architectures

B.1. DCGAN for FashionMNIST

Layer type	Units	Scaling	Activation	Output shape
Input z	-	-	-	7x7
Input y	-	-	-	28x28
Dense	343	-	ReLU	7x7
Conv2DTranspose	128 3x3	x2	ReLU	14x14
Conv2DTranspose	64 3x3	x2	ReLU	28x28
Conv2DTranspose	1 3x3	x1	tanh	28x28
Input x	-	-	-	28x28
Input y	-	-	-	28x28
Conv2D	64 3x3	x1/2	LeakyReLU	14x14
Conv2D	128 3x3	x1/2	LeakyReLU	7x7
Conv2D	1 3x3	x1	tanh	28x28
Dense	1	-	Sigmoid	1

Additional information:

- Batch normalization[41] is applied across all the layers
- A Gaussian noise is applied to the input of the discriminator

B.2. UNet-Res for CIFAR10

Layer type	Units	Scaling	Activation	Output shape
Input y	-	-	-	32x32
Conv2D*	64 5x5	x1	ReLU	32x32
Conv2D*	128 3x3	x1/2	ReLU	16x16
Conv2D*	256 3x3	x1/2	ReLU	8x8
Input z	-	-	-	8x8
Dense	256	-	ReLU	8x8
Residual block	3x256 3x3	x1	ReLU	8x8
Residual block	3x256 3x3	x1	ReLU	8x8
Residual block	3x256 3x3	x1	ReLU	8x8
Residual block	3x256 3x3	x1	ReLU	8x8
Conv2DTranspose*	256 3x3	x2	ReLU	16x16
Conv2DTranspose*	128 3x3	x2	ReLU	32x32
Conv2DTranspose*	64 3x3	x1	ReLU	32x32
Conv2D	3 3x3	x1	tanh	32x32
Input x	-	-	-	32x32
Input y	-	-	-	32x32
Conv2D	64 3x3	x1/2	LeakyReLU	16x16
Conv2D	128 3x3	x1/2	LeakyReLU	8x8
Conv2D	256 3x3	x1/2	LeakyReLU	4x4
Dense	1	-	Sigmoid	1

Additional information:

- Instance normalization[42] is applied across all the layers instead of Batch normalization. This is involved by the use of the PacGAN technique.
- A Gaussian noise is applied to the input of the discriminator
- The layers noted with an asterisk are linked with a skip-connection

B.3. UNet-Res for CelebA

Layer type	Units	Scaling	Activation	Output shape
Input y	-	-	-	128x128
Conv2D	64 5x5	x1	ReLU	128x128
Conv2D*	128 3x3	x1/2	ReLU	64x64
Conv2D*	256 3x3	x1/2	ReLU	32x32
Conv2D*	512 3x3	x1/2	ReLU	16x16
Input z	-	-	-	16x16
Dense	256	-	ReLU	16x16
Residual block	3x256 3x3	x1	ReLU	16x16
Residual block	3x256 3x3	x1	ReLU	16x16
Residual block	3x256 3x3	x1	ReLU	16x16
Residual block	3x256 3x3	x1	ReLU	16x16
Residual block	3x256 3x3	x1	ReLU	16x16
Residual block	3x256 3x3	x1	ReLU	16x16
Conv2DTranspose*	256 3x3	x2	ReLU	32x32
Conv2DTranspose*	128 3x3	x2	ReLU	64x64
Conv2DTranspose*	64 5x5	x2	ReLU	128x128
Conv2D	3 3x3	x1	tanh	128x128
Input x	-	-	-	128x128
Input y	-	-	-	128x128
Conv2D	64 3x3	x1/2	LeakyReLU	64x64
Conv2D	128 3x3	x1/2	LeakyReLU	32x32
Conv2D	256 3x3	x1/2	LeakyReLU	16x16
Conv2D	512 3x3	x1/2	LeakyReLU	32x32
Dense	1	-	Sigmoid	1

This network follows the same additional setup as described in Appendix (B.2).

B.4. Architectures for Texture

B.4.1. PatchGAN discriminator

Layer type	Units	Scaling	Activation	Output shape
Input x	-	-	-	160x160
Input y	-	-	-	160x160
Conv2D	64 3x3	x1/2	LeakyReLU	80x80
Conv2D	128 3x3	x1/2	LeakyReLU	40x40
Conv2D	256 3x3	x1/2	LeakyReLU	20x20
Conv2D	512 3x3	x1/2	LeakyReLU	10x10

B.4.2. UpDil Texture

Layer type	Units	Scaling	Activation	Output shape
Input z	-	-	-	20x20
Conv2DTranspose	256 3x3	x2	ReLU	40x40
Conv2DTranspose	128 3x3	x2	ReLU	80x80
Conv2DTranspose	64 3x3	x2	ReLU	160x160
Input y	-	-	-	160x160
Conv2D	64 3x3 dil. 1	x1	ReLU	160x160
Conv2D	128 3x3 dil. 2	x1	ReLU	160x160
Conv2D	256 3x3 dil. 3	x1	ReLU	160x160
Conv2D	512 3x3 dil. 4	x1	ReLU	160x160
Conv2D	3 3x3	x1	tanh	160x160

B.4.3. UpEncDec Texture

Layer type	Units	Scaling	Activation	Output shape
Input z	-	-	-	20x20
Conv2DTranspose	256 3x3	x2	ReLU	40x40
Conv2DTranspose	128 3x3	x2	ReLU	80x80
Conv2DTranspose	64 5x5	x2	ReLU	160x160
Input* y	-	-	-	160x160
Conv2D*	64 3x3	x1/2	ReLU	80x80
Conv2D*	128 3x3	x1/2	ReLU	40x40
Conv2D	256 3x3	x1/2	ReLU	20x20
Conv2DTranspose*	256 3x3	x2	ReLU	40x40
Conv2DTranspose*	128 3x3	x2	ReLU	80x80
Conv2DTranspose*	64 3x3	x2	ReLU	160x160
Conv2D	3 3x3	x1	tanh	160x160

B.4.4. UNet Texture

Layer type	Units	Scaling	Activation	Output shape
Input y	-	-	-	160x160
Conv2D	64 5x5	x1	ReLU	160x160
Conv2D*	128 3x3	x1/2	ReLU	80x80
Conv2D*	256 3x3	x1/2	ReLU	40x40
Conv2D*	512 3x3	x1/2	ReLU	20x20
Input z	-	-	-	20x20
Conv2DTranspose*	256 3x3	x2	ReLU	40x40
Conv2DTranspose*	128 3x3	x2	ReLU	80x80
Conv2DTranspose*	64 5x5	x2	ReLU	160x160
Conv2D	3 3x3	x1	tanh	160x160

B.4.5. Res Texture

Layer type	Units	Scaling	Activation	Output shape
Input y	-	-	-	160x160
Conv2D	64 5x5	x1	ReLU	160x160
Conv2D	128 3x3	x1/2	ReLU	80x80
Conv2D	256 3x3	x1/2	ReLU	40x40
Conv2D	512 3x3	x1/2	ReLU	20x20
Input z	-	-	-	20x20
Residual block	3x256 3x3	x1	ReLU	20x20
Residual block	3x256 3x3	x1	ReLU	20x20
Residual block	3x256 3x3	x1	ReLU	20x20
Residual block	3x256 3x3	x1	ReLU	20x20
Residual block	3x256 3x3	x1	ReLU	20x20
Residual block	3x256 3x3	x1	ReLU	20x20
Conv2DTranspose	256 3x3	x2	ReLU	40x40
Conv2DTranspose	128 3x3	x2	ReLU	80x80
Conv2DTranspose	64 5x5	x2	ReLU	160x160
Conv2D	3 3x3	x1	tanh	160x160

B.4.6. UNet-Res Texture

Layer type	Units	Scaling	Activation	Output shape
Input y	-	-	-	160x160
Conv2D	64 5x5	x1	ReLU	160x160
Conv2D*	128 3x3	x1/2	ReLU	80x80
Conv2D*	256 3x3	x1/2	ReLU	40x40
Conv2D*	512 3x3	x1/2	ReLU	20x20
Input z	-	-	-	20x20
Residual block	3x256 3x3	x1	ReLU	20x20
Residual block	3x256 3x3	x1	ReLU	20x20
Residual block	3x256 3x3	x1	ReLU	20x20
Residual block	3x256 3x3	x1	ReLU	20x20
Residual block	3x256 3x3	x1	ReLU	20x20
Residual block	3x256 3x3	x1	ReLU	20x20
Conv2DTranspose*	256 3x3	x2	ReLU	40x40
Conv2DTranspose*	128 3x3	x2	ReLU	80x80
Conv2DTranspose*	64 5x5	x2	ReLU	160x160
Conv2D	3 3x3	x1	tanh	160x160

As for Cifar10, this network follows the same additional setup described in Appendix (B.2).

C. Domain-specific metrics for underground soil generation

In this section, we compute the connectivity function [39] of generated soil image, a domain-specific metric, which is the probability that a continuous pixel path exists between two pixels of the same value (called Facies) in a given

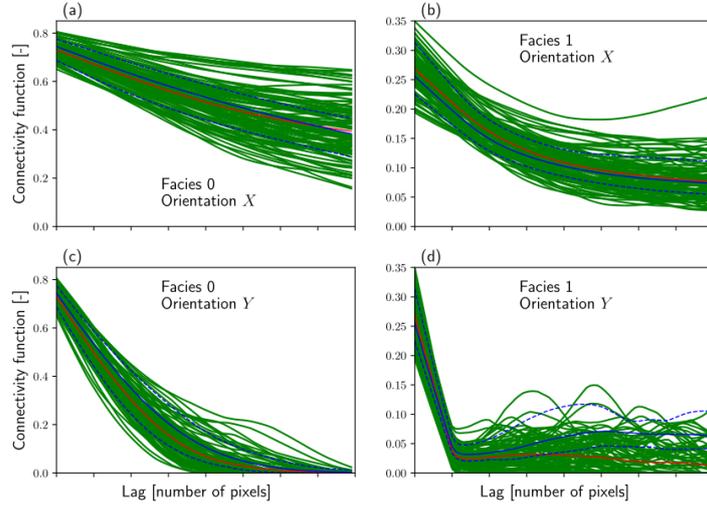


Figure 6: Connectivity curves obtained on 100 samples generated with the CGAN approach.

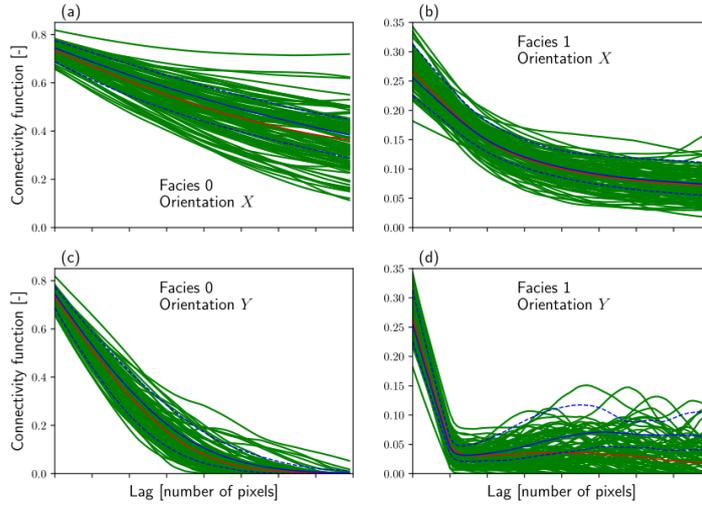


Figure 7: Connectivity curves obtained on 100 samples generated with our approach.

direction and a given distance (called Lag). This connectivity function should be similar to the one obtained on real-world samples. In this application, the connectivity function models the probability that two given pixels are from the same sand brick or clay matrix zone.

We sampled 100 real and 100 generated images using the UNetResPAC architecture (see Section 4.2) on which the connectivity function was evaluated for

both the CGAN and our approach. The obtained graphs are shown respectively in Figures 6 and 7.

The blue curves are the mean value for the real samples, and the blue dashed curves are the minimum and maximum values on these samples. The green curves are the connectivity functions for each of the 100 synthetic samples and the red curves are their mean connectivity functions. From these curves we observe that that our approach has similar connectivity functions as the CGAN approach while being significantly better at respecting the given constraints (see Section Table 4).

D. Additional samples from the Texture and Subsurface datasets

In this section, we show some samples generated with the UNetResPAC architecture, which performs the best in our experiments (see Section 5) compared to real images sampled from the Texture (Figure 8) and Subsurface (Figure 9) datasets. For the generated samples, the enforced pixel constraints are colored in the images, green corresponding to a squared error less than 0.1 and red otherwise.

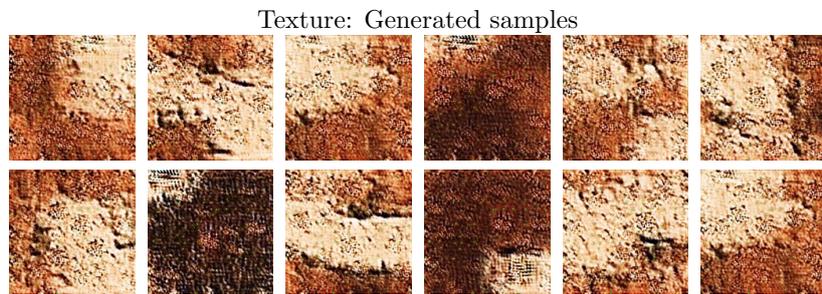


Figure 8: Real and generated samples from the Texture dataset.

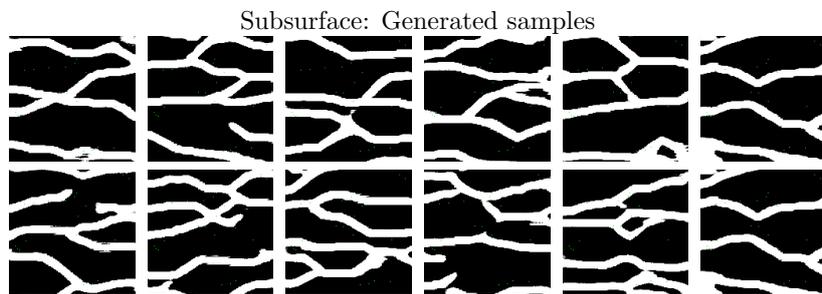
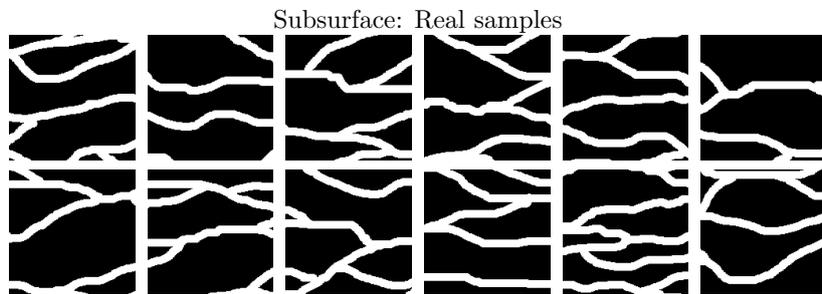


Figure 9: Real and generated samples from the Subsurface dataset.

A.5 Temporal dynamics of inter-limb coordination in ice climbing revealed through change-point analysis of the geodesic mean of circular data

Reference

[Sei+13a] Ludovic Seifert et al. “Temporal Dynamics of Inter-Limb Coordination in Ice Climbing Revealed through Change-Point Analysis of the Geodesic Mean of Circular Data.” In: *Journal of Applied Statistics* 40.11 (Nov. 2013), pp. 2317–2331. doi: 10.1080/02664763.2013.810194. URL: <https://hal.archives-ouvertes.fr/hal-02094911>

Temporal dynamics of inter-limb coordination in ice climbing revealed through change-point analysis of the geodesic mean of circular data

Ludovic Seifert^a, Jean-François Coeurjolly^{b*}, Romain Hérault^c,
Léo Wattebled^a and Keith Davids^d

^aFaculty of Sports Sciences, Centre d'Etude des Transformations des Activités Physiques et Sportives (CETAPS) – EA 3832, University of Rouen, Mont-Saint-Aignan, France; ^bLaboratory Jean Kuntzmann – UMR CNRS 5224, Grenoble University, Grenoble, France; ^cLaboratoire d'Informatique, de Traitement de l'Information et des Systèmes (LITIS) – EA 4108, National Institute of Applied Sciences (INSA), Rouen, France; ^dSchool of Human Movement Studies, Queensland University of Technology, Brisbane, Australia

(Received 9 July 2012; accepted 28 May 2013)

This study examined the temporal dynamics of the inter-limb angles of skilled and less skilled ice climbers to determine how they explored ice fall properties to adapt their coordination patterns during performance. We observed two circular time series corresponding to the upper- and lower-limbs of seven expert and eight inexperienced ice climbers. We analyzed these data through a multiple change-point analysis of the geodesic (or Fréchet) mean on the circle. Guided by the nature of the geodesic mean obtained by an optimization procedure, we extended the filtered derivative method, known to be computationally very cheap and fast, to circular data. Local estimation of the variability was assessed through the number of change-points computed via the filtered derivatives with p -value method for the time series and integrated squared error (ISE). Results of this change-point analysis did not reveal significant differences of the number of change-points between groups but indicated higher ISE that supported the existence of plateaux for beginners. These results emphasized higher local variability of limb angles for experts than for beginners suggesting greater dependence on the properties of the performance environment and adaptive behaviors in the former. Conversely, the lower local variance of limb angles assessed in beginners may reflect their independence of the environmental constraints, as they focused mainly on controlling body equilibrium.

Keywords: variability analysis; inter-limb coordination; climbing; change-point analysis; geodesic mean; circular data

*Corresponding author. Email: jean-francois.coeurjolly@upmf-grenoble.fr

1. Context of the application and main objective

Traditionally, it has been argued that the acquisition of movement expertise is characterized by a linear progression towards invariance in motor output performance. From this perspective, movement pattern variability is an errorful by-product of noise in the central nervous system that should be minimized or eliminated with practice. This view is compounded by the types of motor tasks used to study movement performance in experiments, emphasizing deviations from an ideal performance template, characteristic of expert behavior [32,34]. However, from an ecological dynamics perspective, observed variability in motor output in both novice and expert individuals in performance domains like sport may not necessarily be a reflection of system error or noise. Expertise in sport results from the adaptation of behaviors to interacting constraints, individually perceived and encountered. Indeed, the intertwined relationship between perceptions and actions constrains the direction, and restrains the range of movement possibilities available for each individual performer. With this emphasis on perception and action to constrain behaviors, the role of movement pattern stability and functional intra-individual performance variability is paramount. Beek *et al.* [5] suggested that the nature of relationship and the coupling of perception and action is not the same for non-experts and experts, since the expert is more capable of exploiting information about task-related constraints in order to organize their behaviors.

In traditional research, movement expertise has been captured statistically through calculations of the magnitude of variance measures like the standard deviation of the mean distribution and the coefficient of variation [17,28]. These statistical indicators attempt to characterize the data distribution and the amount of noise in a single measurement pertaining to performance. However, such statistical measurements only indicate the magnitude of system variability (i.e. the amplitude and the spatial distribution of performance outcomes over trials), but not the dynamical structure of the data series [28]. Recent studies have explored the structure of variability for performance outcomes through identifying the learning dynamics, showing multiple time scales of variability (such as exponential, power law and S-shaped performance curves) [24,27,29,30]. According to the dynamical systems' approach, these multiple time scales emphasize the discontinuities that typify learning, based on the assumptions that learning is constructed from spontaneous manifestations of motor coordination and that often it is necessary to destabilize an established motor pattern in order to provoke the attainment of expert coordination [37]. For instance, it has been hypothesized [29] that when a system is close to its stable state, it will change at a constant time scale (i.e. an exponential function), while multiple time scales (i.e. a power law) are expected to be observed when movement coordination goes through transition. The emphasis in our study was on discovering statistical measures which capture the structure of movement pattern variability through observing the temporal dynamics of motor variability during an ice climbing ascent. In this paper we discuss new data analysis approaches which can demonstrate how studying expertise differences in sport can benefit from new statistical methodologies. Since movement patterns, during ice climbing, are predicated on ice fall properties (e.g. shape, steepness, temperature, thickness and ice density), an important question concerns how ice climbers of various skill levels exploit affordances (i.e. possibilities for action offered by a particular performance environment [18]) to organize their upper and lower movements over time. Our hypothesis is that ice fall properties contain affordances that induce variable motor coordination patterns in expert climbers, whereas learners use a basic and functionally stable motor organization to achieve their main goal of maintaining body equilibrium with respect to gravity.

The selected characteristics of different levels of expertise in ice climbing will be statistically described in Section 2. Our analysis has been based on the collection of angular time series data for eight beginners and seven experts. Such data fall into the general domain of circular statistics, for which there are general references [22,25,26]. The concepts of location and dispersion for circular statistics are really specific and, on a general manifold, we refer to other publications for a deep understanding of these concepts [9,10,20]. In this article, we have focused on intrinsic

characteristics, i.e. on the geodesic mean and the geodesic variance. These parameters are the natural extensions of the standard Euclidean mean and the standard Euclidean variance when substituting the Euclidean distance by the geodesic distance on the circle; a distance which is referred to as the arc-length distance. We have recalled the main definitions of these concepts and apply this measure of dispersion in Section 2.4. As demonstrated in this section, based on computations of the sample geodesic variance, there was a clear distinction in behaviors between beginners and experts. However, this difference was mainly due to the fact that expert climbers explore a larger range of angle values than beginners. To have a better understanding of behaviors for both groups of climbers at different levels of expertise, we have turned to a local analysis of these data. In particular, we aimed to develop a change-point analysis of the geodesic mean for circular data.

The problem of change-point detection is certainly one of the most investigated issues by statisticians which has led to the development of a huge body of literature; for example [3,12,16] or, for more recent review, the article by Hušková and Meintanis [21]. It consists of detecting one or more time points where parameters of a process change. One of these methods, the filtered derivative, was initially introduced by Benveniste and Basseville [6], Basseville and Nikiforov [3], and Antoch and Hušková [1]. Generally speaking, this method consists of computing local estimations of the parameter of interest via a *filter* like a mean or a M -estimator, and in detecting changes in these local estimations through *derivation*. Extensions and theoretical studies have been considered by Bertrand *et al.* [7,8]. The statistical contribution of this article is to extend understanding of the filtered derivative with the p -value method in order to detect multiple change-points on the geodesic mean for circular data; this method is referred to as **fdpv** in the following sections. Regarding our data, and in particular the parameter of interest, it has raised the idea that the sample geodesic mean was actually an M -estimate obtained through an optimization procedure. Moreover, no sequential formula was available for such an estimate, i.e. the sample geodesic of the data set (y_1, \dots, y_n) could not be obtained using the sample geodesic mean of the data set (y_1, \dots, y_{n-1}) and y_n . This convinced us to turn to a method that shows a very low complexity and a short running time. The main interest of the **fdpv** method that is used to detect abrupt changes in the standard Euclidean mean, variance or parameters of a simple linear regression is the fact that its time and complexity memory are both of order $\mathcal{O}(n)$.

The rest of the paper is organized as follows. We present in Section 2 the protocol from which the data have been obtained and the main characteristics of the climbers included in this study. We also specify the contribution of our data which is the observation of angular time series and introduce the concepts of geodesic mean and geodesic variance for circular data which constitute the parameters of interest of this paper. Section 3 turns to the core of the paper which is the extension to circular data of the filtered derivative method with p -values allowing us to propose an efficient change-point analysis method of the geodesic mean of a circular time series. Finally, we apply the developed methodology and discuss and interpret the results from a practical point of view in Section 4.

2. Description of the data and global variability analysis

2.1 Participants

Fifteen male ice climbers, divided into two groups, volunteered for this study. Seven expert climbers with mean age: 32.1 ($\sigma = 6.1$); mean height: 176.4 cm ($\sigma = 6.2$ cm); mean weight: 68.4 kg ($\sigma = 6.7$ kg); skill level in rock climbing: grade 7a+ to 7c on the French rating scale, which ranges from 1 to 9; mean number of years practicing rock climbing: 17.1 ($\sigma = 5.6$); skill level in ice fall climbing: grade 6–7 on the French rating scale, which goes from 1 to 7 [4]; mean number of years of practice in ice climbing: 10.4 ($\sigma = 4.7$); mean number of days of ice climbing

per year: 20.6 ($\sigma = 9.3$). They were considered as skilled climbers since they were (i) mountain guides, certified by the International Federation of Mountain Guides Association (IFMGA) or/and (ii) instructors at the French National School of Skiing and Alpinism (ENSA). The eight beginners (mean age: 28.5 ($\sigma = 6.4$); mean height: 177.2 cm ($\sigma = 5.8$ cm); mean weight: 71.8 kg ($\sigma = 8.9$ kg)) were students in a Faculty of Sport Sciences at a local university, with 20 h of practice on an artificial climbing wall and were inexperienced at ice climbing.

2.2 Protocol

To impose a similar task constraint on both groups [28], a sub-maximal level of effort was imposed that corresponded to a 30 m ice fall climb at grade 5+ for expert climbers (which is a regular grade for them). The beginners climbed a 30 m ice fall at grade 4 (a common grade assigned to that skill level). Grade 5 + /6 signifies vertical climbing for most of the ice fall, while grade 4 involves alternation of steep sections around 80–85°, with ramps around 60–70°. For this protocol, the ice fall selected for the beginners was in three sections: 20 m at 85°, ramp of 5 m at 70°, then 5 m at 80°. Although a similar task constraint was imposed on the participants, these differences of grade between the two groups would represent different environmental constraints (i.e. in terms of steepness). Consequently, to enable a valid comparison between skilled climbers and beginners, the first 20 m part of the ice fall that corresponded to 85° of steepness for both groups was selected to analyze the motor behavior. Performance data were collected in two sessions during which the air temperature was, respectively, -8°C and -12°C . All climbers were equipped with the same crampons and ice tools and were instructed to climb at their normal pace. The protocol was approved by the University ethics committee and followed the declaration of Helsinki. Procedures were explained to the climbers, who then gave their informed consent to participate.

2.3 Data collection

A frontal camera (25 Hz), positioned 15 m behind the climber perpendicular to the ice fall, digitally recorded the first 20 m of the climb. A calibration frame delimited the recorded space of climbing performance and was composed of one vertical rope with marks every 2 m and two horizontal ropes (at 5 m and at 20 m) with marks every 1 m (total of 20 marks for calibration). Five key points (the head of left and right ice tools, and the extremity of left and right crampons) were digitized using Simi Motion Systems®(2004). Since climbing was self-paced, the time of ascent was not considered in assessing performance.

The nature and the number of ice tool and crampon actions completed during the ascent were counted, including (i) the ratio between definitive anchorage and repetitive ice tool swinging, and (ii) the ratio between definitive anchorage and repetitive crampon kicking.

Upper-limb coordination patterns were assessed by using the angle between the horizontal line and the displacement of the heads of the left and right hand ice tools. Lower-limb coordination patterns corresponded to the angle between the horizontal line and the displacement of the left and right crampons (Figure 1). These two signals were smoothed by a Butterworth low-pass filter (cut-off frequency 6.25 Hz) by Matlab 7.7®(1984–2008, The MathWorks, Inc.) as suggested by Winter [36] to address noise introduced by body marks digitizing from the video yet preserving movement information.

We highlighted eight angle modes, each of them are 45° span. When the angle was $0 \pm 22.5^{\circ}$, the two limbs were horizontal, meaning that they were simultaneously flexed or simultaneously extended, corresponding to an in-phase mode of coordination. When the angle was $\pm 90 \pm 22.5^{\circ}$, one limb was vertically located above the other limb, meaning that one was flexed, while the other was extended, corresponding to an anti-phase mode of coordination. Between these values, the limbs showed a diagonal angle so that coordination was considered in an intermediate mode. The

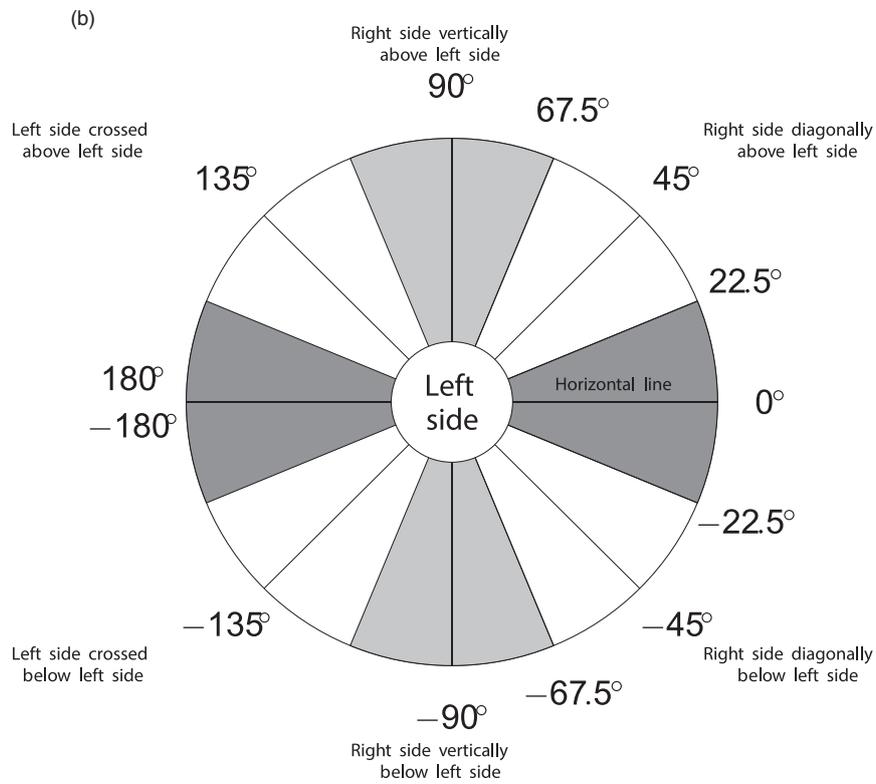
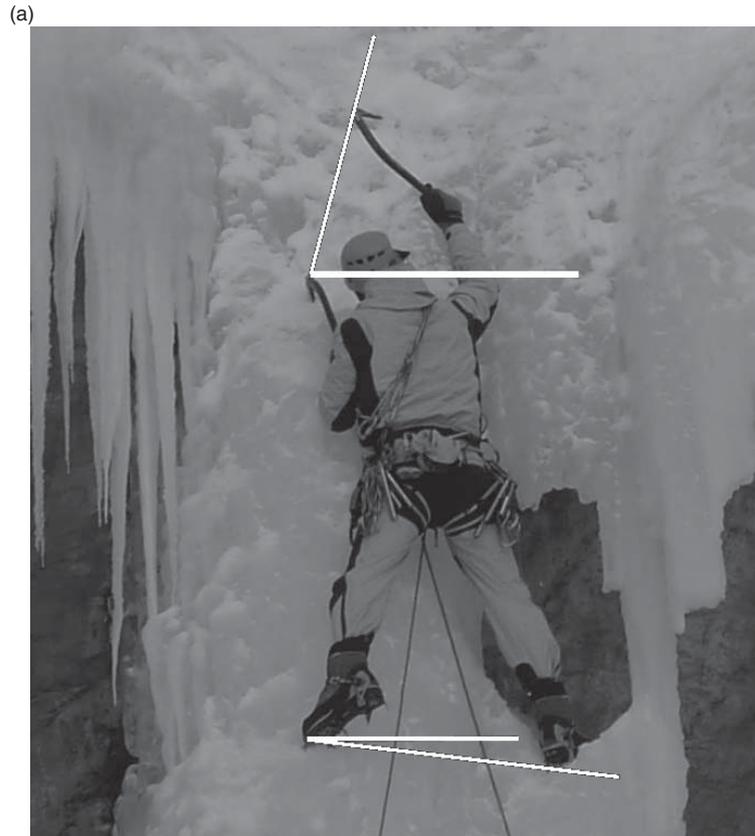


Figure 1. Angular position of limbs. (a) Angle between horizontal, left limb and right limb. (b) Modes of limbs' coordination as regards the angle value between horizontal, left limb and right limb.

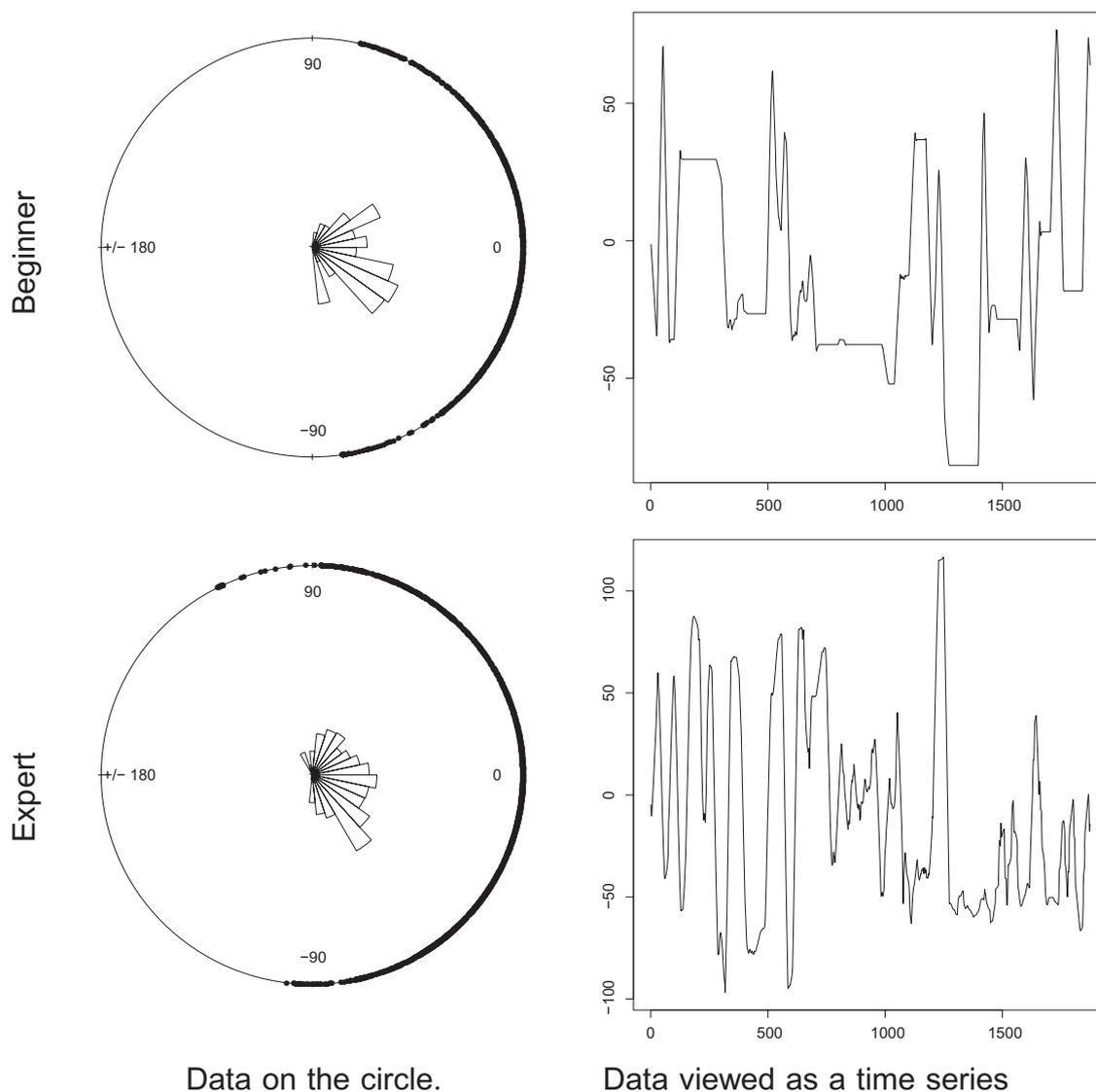


Figure 2. Angular data of the hand ice tools (i.e. time series ULC) for a beginner climber (top) and an expert one (bottom). The left plots represent the data on the circle. These plots and the associated rose diagram have been generated using the R package `circular`. The right plots represent the same data viewed as a time series (note that the vertical axis is not the same for the beginner and the expert climbers).

angle between the horizontal line and the left and right limbs was positive when the right limb was above the left limb and negative when the right limb was below the left limb, see Figure 1. To summarize, we collected two time series of angular data on performance for eight beginners and seven expert climbers: the first time series was related to the use of ice tools and represents upper-limb coordination, whereas the second set of data was related to the use of crampons, corresponding to lower-limb coordination. These time series, whose lengths are more or less 1874 data points (with a duration of 4 s between two points), will be, respectively, denoted by upper-limb coordination (ULC) and lower-limb coordination (LLC) in the following. Figure 2 illustrates these data for two different climbers and the next section aims to explore the statistical analysis of the variability characteristics in these data.

2.4 Global variability analysis

Due to the compactness of the circle (particular case of a compact manifold), the standard notions of mean and variance are not suited for circular data. Refs. [22,25,26] of the extant literature

contain some detailed analyses of the concepts of location and dispersion for data on a circle (and for some of the references on a general manifold) (see [9,10,20]). Among these different concepts of location, we have focused in this article on the notions of a geodesic mean (or intrinsic mean) and geodesic variance (or intrinsic variability), that is on characteristics which are intrinsically defined via a distance on the manifold, here the circle S^1 . In our opinion, these concepts seemed more natural than the classical extrinsic mean (obtained as the projection on the circle of the point with abscissa (resp. ordinate) equal to the mean of the cosine (resp. sine) of the angle) and extrinsic variance which are the concepts on which the general references [22,25,26] are based on.

For the specific manifold of the circle, denoted by S^1 , the geodesic distance is the arc-length distance (expressed in degrees) which for two angles $(\alpha, \beta) \in [-180^\circ, 180^\circ]^2$ is expressed as

$$d_G(\alpha, \beta) = 180^\circ - |180^\circ - |\alpha - \beta||. \quad (1)$$

Now, given n observations y_1, \dots, y_n of angles (expressed in radians units), the geodesic sample mean is defined by

$$\hat{\mu}_G = \underset{\mu \in S^1}{\text{Argmin}} \frac{1}{n} \sum_{i=1}^n d_G(y_i, \mu)^2 \quad (2)$$

and the geodesic sample variance by

$$\hat{\sigma}_G^2 = \frac{1}{n} \sum_{i=1}^n d_G(y_i, \hat{\mu}_G)^2, \quad (3)$$

which obviously satisfy $0 \leq \hat{\sigma}_G^2 \leq (180^\circ)^2$. The sample geodesic mean and sample geodesic variance generalize in a very natural way the standard (Euclidean) sample mean and sample variance: the Euclidean distance is simply replaced by the geodesic distance. The notation μ_G and σ_G^2 stand for the theoretical geodesic mean and variance. The geodesic mean μ_G may not be unique, see [13,19,23] for a complete survey of this topic (on the circle). For example, the geodesic mean can be any point of the circle for an uniform distribution on $[-180^\circ, 180^\circ)$. Despite this, the sample geodesic mean is almost surely unique. Regarding the variances, the theoretical and the geodesic sample variances are necessarily unique. We applied this concept of dispersion for circular data to the time series ULC and LLC for all ice climbers.

Results are depicted in Figure 3 (top left). These findings make it clear that the global variance (i.e. the variance computed on the overall time series) was really linked to the intrinsic performance level of the climber. This clustering effect was actually mainly due to the large range of angles used by experts in contrast to beginners (see the top right plot of Figure 3). Indeed, if we had artificially rescaled the data such that the range of all the time series was $[-90^\circ, 90^\circ]$ (affine transformation applied to each angular time series), the geodesic variance would be less discriminant (see the right plot of Figure 3). To confirm these visual characteristics, we conducted a one-way analysis of variance (ANOVA; fixed factor: skill level). We obtained significantly higher geodesic variances of ULC ($F_{1,13} = 27.28, p = 0.0002$) and LLC ($F_{1,13} = 7.52, p = 0.0017$) for expert climbers than for beginners if we consider the raw data. However, based on the rescaled data, there was no clear evidence of significant different variances ($F_{1,13} = 2.72, p = 0.123$ for the ULC time series and $F_{1,13} = 1.81, p = 0.201$ for the LLC time series).

In conclusion, the global variability was almost linked to the global performance of the climber and did not really reflect the climber's style of performance. Figure 2 shows that expert climbers explored a larger range of angular positions of limbs than non-experts. Notably, according to the angular position classification of Figure 1, expert climbers exploited horizontal, diagonal, vertical and crossed angular positions while non-experts mostly used horizontal and diagonal angular positions.

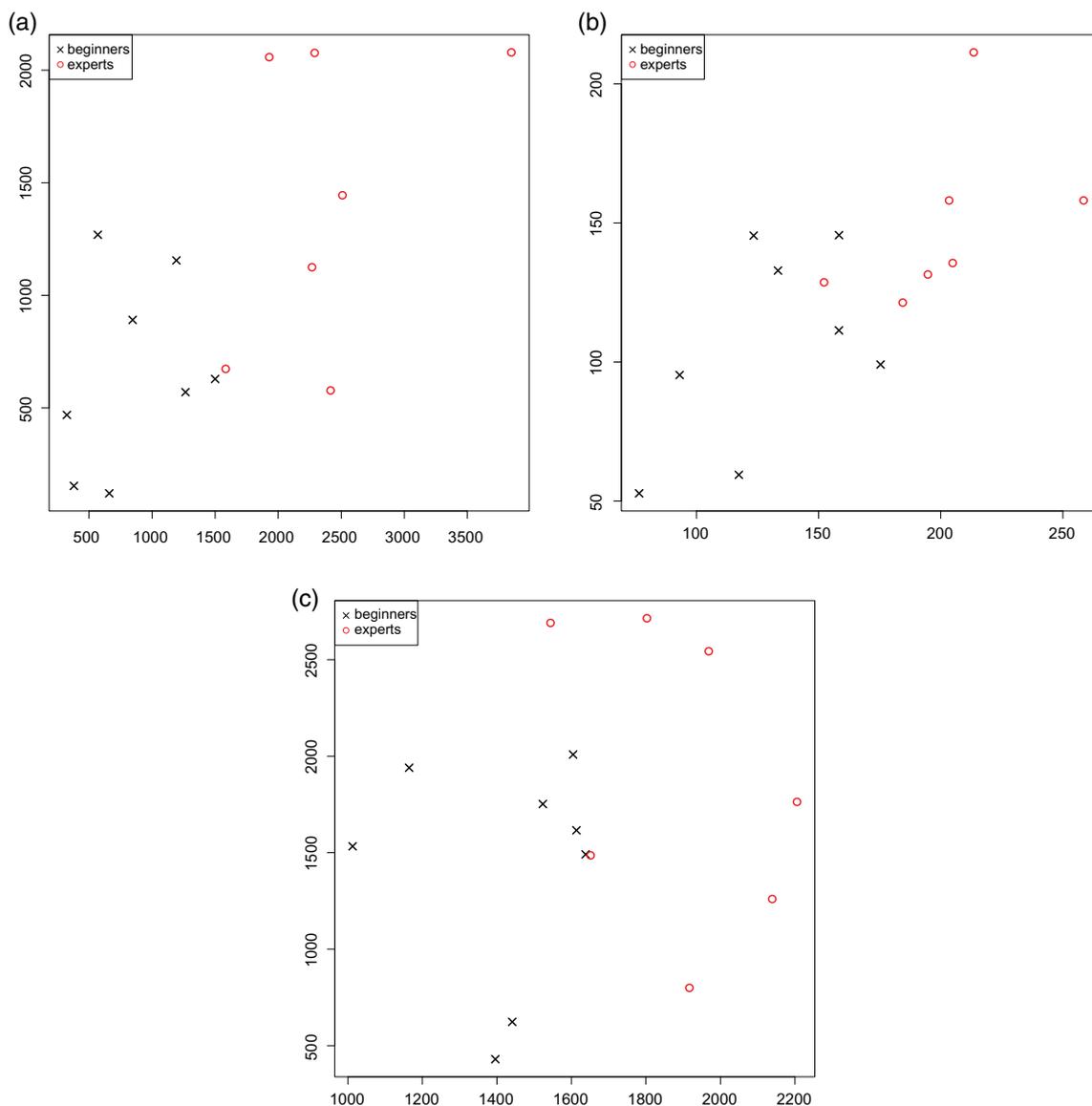


Figure 3. Upper-limb coordination versus lower-limb coordination. The abscissa (resp. the ordinates) correspond to the time series ULC (resp. LLC). (a) Geodesic variances of raw data. (b) Ranges of the time series (in $^{\circ}$). (c) Geodesic variances on data artificially rescaled.

We demonstrate in the next section that a local estimation of the variability, based on a segmentation of the circular time series, provides more information on the behavioral differences in-between beginners and in-between expert climbers. And we highlight that the local estimation is almost independent of the global performance, i.e. independent of the support of the data.

3. Filtered derivative method for circular data

In this section, we extended a procedure based on filtered derivatives with a p -value to detect changes in the geodesic mean of circular data. Let y_1, y_2, \dots, y_n be the sample of a circular time series of length n . We decompose the methodology into two different steps: detection of potential change-points and then deletion of false alarms.

Step 1 Detection of potential change-points.

Roughly speaking, the filtered derivative method consists of computing local estimations of the parameter of interest and to detect changes on these local estimations. We define for some integer

$A \geq 1$ and for $k \in \{A + 1, \dots, n - A\}$ the statistic $D(k, A)$ by

$$D(k, A) := d_G(\hat{\mu}_G[(k - A + 1) : k], \hat{\mu}_G[(k + 1) : (k + A)]), \tag{4}$$

where $\hat{\mu}_G[i : j]$ (for $1 \leq i < j \leq n$) is the geodesic sample mean based on the observations $y_i, y_{i+1}, \dots, y_{j-1}, y_j$. In [1] or [7], potential change-points are selected as times corresponding to the local maxima of the absolute value of the filtered derivative time series $D(\cdot, A)$, moreover, when this last quantity exceeds a given threshold. Then, in [8], the authors used a slightly different approach: a probability of a type I error is fixed at a level $p_1 \in (0, 1)$ and the corresponding threshold C_1 is given by

$$P \left(\underbrace{\max_{k \in [A, n-A]} D(k, A)}_{:=M(A)} > C_1 \mid H_0 \text{ is true} \right) = p_1, \tag{5}$$

where H_0 represents the null hypothesis, corresponding here to the absence of change-points. More specifically, H_0 corresponds to the situation where y_1, \dots, y_n are independent realizations of the same random variable Y . Bertrand *et al.* [8] considered the problem of detecting changes in the Euclidean mean, variance and parameters of the simple linear regression. For these problems, we managed to determine the asymptotic survival function of $M(A)$ under the assumption of independence of the observations when Y followed a distribution satisfying some moments conditions. The translation to circular data is not straightforward, and we address the estimation of parameters through two data-driven methods. Thus, regarding the nature of the parameter of interest (actually an M -estimate), we considered a parametric bootstrap approach and a non-parametric re-sampling method to estimate the survival function of $M(A)$:

- (1) *Parametric bootstrap approach.* The asymptotic distribution in [8] obtained for the parameters (mean and variance) is obtained for a mild assumption on the distribution of the data. In this vein, we modeled the data by a specific parametric circular distribution, estimated the parameters using the maximum likelihood method and estimated the distribution of $M(A)$ using B replications of the circular distribution with estimated parameters. For this method, we have chosen the geodesic normal distribution on the circle, whose circular density rewrites $f(\theta) = k^{-1}(\gamma) e^{-(\gamma/2)(180/\pi)^2 d_G(\mu, \theta)^2}$ for some angle $\mu \in [-180^\circ, 180^\circ)$ and some real number $\gamma \geq 0$ and where $k(\gamma)$ is a normalizing constant given by $k(\gamma) = \sqrt{2\pi/\gamma} \operatorname{erf}(\pi \sqrt{\gamma}/2)$. The geodesic normal distribution was first introduced by Pennec [31] and defined for general Riemannian manifolds. In the particular case of the circle, Coeurjolly and Le Bihan [15] studied its statistical properties (moments, simulation, asymptotic properties of the maximum likelihood estimates, etc.). The choice of this circular distribution is guided by the fact that parameter μ corresponds to the geodesic theoretical mean of this distribution (our parameter of interest) and the MLE of μ is the sample geodesic mean.
- (2) *Non-parametric re-sampling method.* Following advice in [2], the survival function is estimated by replications of the data obtained by permutations (B replications are used).

In our application (presented in Section 4), we set p_1 to 10% and used $B = 5000$ replications to estimate C_1 for each circular time series. We observed that for different values of A , both approaches lead to quite similar results (for each climber). Therefore, we only kept the permutation approach in the presentation of our empirical results.

Step 2. Deletion of false alarms.

Let $A \leq \tau_1 < \dots < \tau_{\tilde{K}} \leq n - A$ be the \tilde{K} change-points defined after Step 1. In this step, the signal is segmented into $\tilde{K} + 1$ subsamples. The subsample k consists in the set $\{y_i \mid i \in [\tau_{k-1} +$

$1, \dots, \tau_k\}$, called Y_k , which are the records of the time period $[\tau_{k-1} + 1, \tau_k]$. As there is \tilde{K} change-points, k can vary from 1 to $\tilde{K} + 1$ setting $\tau_0 = 0$ and $\tau_{\tilde{K}+1} = n$.

To delete false alarms, Bertrand *et al.* [8] then proposed to test the parameter of interest (mean, variance) between two successive subsamples Y_k and Y_{k+1} , that is between $\{y_i | i \in [\tau_{k-1} + 1, \dots, \tau_k]\}$ and $\{y_j | j \in [\tau_k + 1, \dots, \tau_{k+1}]\}$.

In other words, \tilde{K} statistical tests are formed and change points were kept if the corresponding p -value was lower than a fixed value p_2 . In our setting, we modeled each signal on the period $[\tau_{k-1} + 1, \tau_k]$ by independent observations of a circular random variable Y_k for $k = 1, \dots, \tilde{K} + 1$, we let $\tilde{\mu}_k$ denote the geodesic mean of Y_k and defined θ_k as $d_G(\tilde{\mu}_k, \tilde{\mu}_{k+1})$. Then, we considered the \tilde{K} statistical tests

$$H_0 : \theta_k = 0 \quad \text{versus} \quad H_1 : \theta_k \neq 0. \quad (6)$$

To be close to the nature of the data, we again proceeded with these different statistical tests using re-sampling methods ($B = 5000$ permutation tests). For this second step, we followed the advice in [8] setting p_2 to the value 10^{-6} . With a slight alteration of notation, the final change-points are denoted by $\tau_1, \dots, \tau_{\check{K}}$ with $A \leq \tau_1 < \dots < \tau_{\check{K}} \leq n - A$ and $\check{K} \leq \tilde{K}$.

4. Results and discussion

4.1 Numerical results

The algorithm implemented in the R software and described in the previous section was applied to the $2 \times 15 = 30$ circular time series. The window parameter has been set to $A = 40$ for the 30 time series. We selected $A = 40$ because the realization of one action of each limb (i.e. left arm swing, right arm swing, left foot kick and right foot kick) took in average 40 points (i.e. a duration of 10 s). Apart from this empirical choice, we would like to underline that the procedures have also been applied with window sizes from $A = 25$ to $A = 60$. The obtained results were quite similar to the findings presented later with the choice $A = 40$, this stability can be explained by the fact that the Step 2 cancels many false discoveries.

We denote by $\hat{\mu}_{G,t}$ the piecewise constant function at time t estimated from the change-point analysis given by

$$\hat{\mu}_{G,t} = \sum_{k=1}^{\check{K}+1} \hat{\mu}_G[(\tau_{k-1} + 1) : \tau_k] \mathbf{1}(t \in [\tau_{k-1} + 1, \tau_k]),$$

where, we set by convention, $\tau_0 = 0$ and $\tau_{\check{K}+1} = n$. In order to quantify the local variations of the data around $\hat{\mu}_G$, we propose to define the following criterion:

$$\text{ISE} = \sum_{t=1}^n d_G(y_t, \hat{\mu}_{G,t})^2 = \sum_{k=1}^{\check{K}+1} (\tau_k - (\tau_{k-1} + 1)) \sum_{t=\tau_{k-1}}^{\tau_k} d_G(y_t, \hat{\mu}_G[(\tau_{k-1} + 1) : \tau_k])^2. \quad (7)$$

As a general comment, we noted that the second step of the **fdpv** method has allowed us to delete between 1 and 3 potential change-points proposed by the first step. The computational aspect was not negligible since due to the huge number of calculations of geodesic sample means and due to the re-sampling procedures, the **fdpv** method required about 1 h for one time series (of length $n = 1874$).

Figure 4 presents an example of the segmentation method with two data sets and their related time series $D(t, A)$ allowing to detect changes. All the segmentations can be found in the supplementary material available online accompanying this paper. Table 1 aims at summing up the numerical results. The number of change-points (after the second step) and the integrated squared

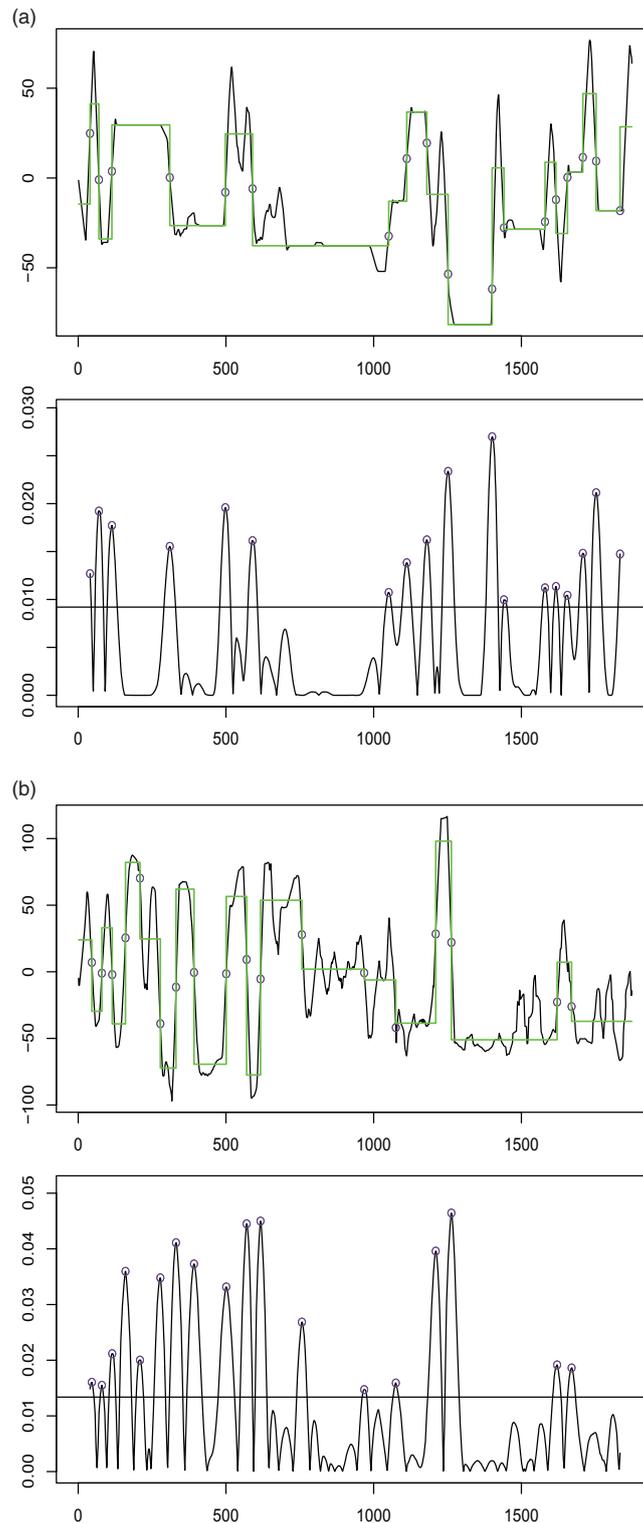


Figure 4. Examples of change-point analysis of the circular time series based on the filtered derivatives method with p -value. The data correspond to Figure 2. The two upper plots represent the selected change-points in blue circle and the resulting piecewise constant function (corresponding to the geodesic means computed on each segment). The bottom plots represent the related plot of the statistic $D(k, A)$ in terms of k . The horizontal line corresponds to the threshold C_1 . For both examples, all change-points selected after the Step 1 were kept after the second step. The parameters of the method were $A = 40, p_1 = 10\%$ and $p_2 = 10^{-4}\%$.

Table 1. Summary results for the 15 ice climbers: number of change points \check{K} computed via the filtered derivatives with p -value method for the time series ULC and LLC (first two columns) and integrated squared error denoted by ISE and defined by Equation (7).

	\check{K}		ISE	
	ULC	LLC	ULC	LLC
Beginner 1	18	13	65.4	59.4
2	9	12	27.7	3.0
3	9	12	24.2	5.9
4	20	16	21.3	28.8
5	18	15	65.8	69.0
6	15	20	61.5	31.7
7	24	16	106.6	30.9
8	23	14	79.7	61.0
Average	17	14.8	56.5	36.2
Expert 1	18	12	160.9	141.4
2	23	17	205.9	144.3
3	24	24	120.3	83.7
4	16	17	227.6	134.8
5	22	17	131.6	104.0
6	14	14	153.3	66.0
7	16	7	150.3	28.7
Average	19	15.4	164.3	97.5

error (ISE) criterion are presented. Whereas the number of change-points was not really different between a beginner and an expert climber, we highlight that the ISE was very discriminating. The low values of the ISE criteria were related to the existence of plateaux for beginners. This will be discussed in the next section.

Figure 3, presented in Section 2, shows in particular that the discrimination power of the global geodesic variance (i.e. computed on the overall data set) was essentially due to the larger range used by experts. After artificially rescaling the data, the global geodesic variance became much less discriminant. Figure 5 illustrates the counterpart of such analysis based on the ISE criterion. Simple statistics have been applied to the data in Table 1: a one-way ANOVA (fixed factor: skill level) showed significantly higher ISE of ULC ($F_{1,13} = 36.53, p < 0.0001$) and LLC ($F_{1,13} = 10.89, p = 0.006$) for expert climbers than for beginners. We highlight that our analysis is much less affected by a rescaling of the data. Indeed, we applied our general methodology to the artificially rescaled data (such that the range is $[-90^\circ, 90^\circ]$). We did not report the results but after undertaking the similar one-way ANOVA, we still obtained significantly higher ISE of ULC ($F_{1,13} = 19.83, p = 0.0006$) and LLC ($F_{1,13} = 9.57, p = 0.009$) for expert climbers than for beginners. There was clear evidence that the change-point analysis is less sensitive to the support of the time series which pertinently signifies in particular that locally the variability of angles for a beginner is much lower than for an expert.

As suggested previously, differences about ULC and LLC numbers could come from different number of action ratios between arm and foot; notably expert climbers often realized 1–3 foot kicks for 1 arm swing; while beginners realized 1–3 arm swings for 1 foot kick. However, we did not compute these data for this study as we did not examine arm to leg coupling.

4.2 Interpretation and discussion

Even if the global geodesic variance indicated significant differences between the two groups of climbers, these differences are linked to the range of angles used by climbers rather than to

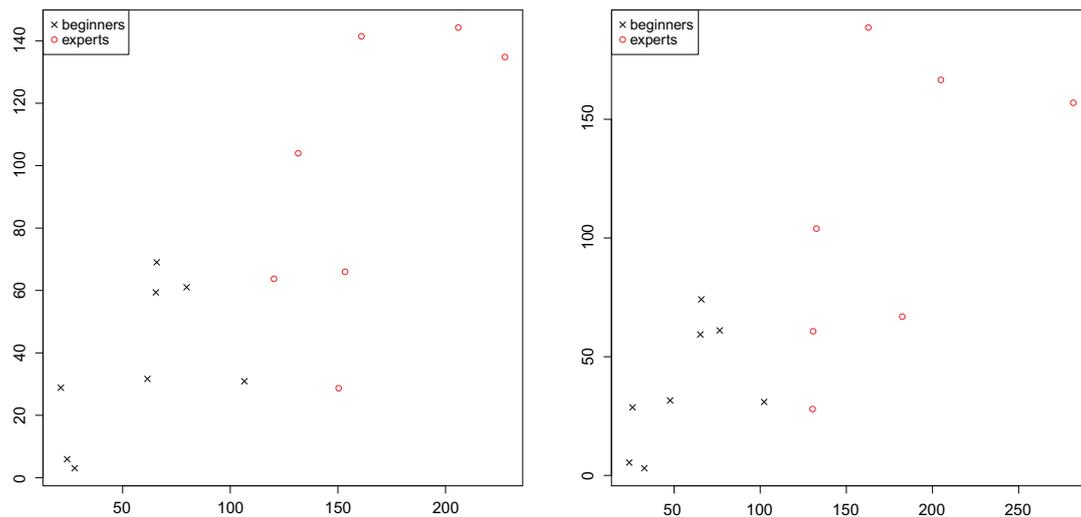


Figure 5. ISE defined by Equation (7) based on the change-point analysis of the circular time series for the 15 ice climbers. The left plot corresponds to the results based on raw data, also presented in the last two columns of Table 1. The right plot corresponds to the methodology based on data artificially rescaled in the interval $[-90^\circ, 90^\circ]$. The abscissa (resp. the ordinates) correspond to the time series ULC (resp. LLC).

the variability of the angles distribution. Therefore, the analysis of the local variance was more powerful to highlight significantly higher ISE for expert climbers than for non-experts, reflecting different behavioral adaptations to environmental constraints (i.e. ice fall properties). The longer duration spent without any movement of limbs statistically identified in the non-expert could have several causes:

- (1) They spend more time to determine their climbing path and their next point of anchorage, suggesting their difficulty to perceive an affordance. Qualitative analysis of video footage revealed that the non-experts swung their ice tool to create a hole in the ice fall whereas natural holes existed close to them. Conversely, expert climbers showed a greater dependence on the environment as they were able to exploit the ice fall properties (e.g. hole in the ice fall) to vary their limb angular positions and their limb movement patterns (e.g. swinging, kicking and hooking). Behavior variability corresponds to adaptive perception–action coupling to climb quickly, efficiently and safely. For instance, video footage showed that experts adopted vertical limb angular positions and sometimes crossed their limbs to hook existing hole in the ice fall and to use with their crampons the holes previously created with their ice tools.
- (2) Non-experts spent more time to stabilize their body as they focused on keeping their body equilibrium under control (according to the findings of Bourdin *et al.* [11] in rock climbing), suggesting their relative independence of the environment. Body movements could be perceived as a potential cause of a fall; therefore beginners seemed to try and control body roll, yaw and pitch by freezing the motor system's degrees of freedom (as already observed in ski simulator tasks, [35]). Conversely, experts released the degrees of freedom to reach greater range of limb motion and length of vertical body displacement. Moreover, experts were able to exploit gravity (environmental constraint) by yaw and roll body motion leading the body to move like a pendulum or a door. The capacity of the expert to vary their movement patterns and limb angular positions revealed multi-stability of movement, that is a property of non-linear dynamical systems [14].
- (3) Last, the non-experts needed a confident anchorage that was often synonymous with a deep anchorage. The high number of ice tool swinging and crampon kicking movements, and the

high ratio between swinging actions and definitive anchorages supported this impression. In particular, our results indicated that experts realized one ice tool swinging for one definitive anchorage, and one crampon kicking for one definitive anchorage. Conversely, for non-expert climbers, the ratio between definitive anchorages and swinging or kicking actions was 0.6 for ice tools and 0.2 for crampons. The non-expert climbers swung their ice tools two times and their crampons five times before a definitive anchorage. Therefore they spent a long time in a static body position leading to the onset of fatigue. This observation is in accordance with previous research in rock climbing [33] which relates to the ‘three-holds-rule’: if a rock climber uses a smaller number of holds he/she has to be quick enough to maintain equilibrium on the surface. Conversely, if the number of holds is equal to or greater than three, it is more likely that the rock climber will climb slowly, because his/her equilibrium is always under control [33].

5. Conclusion

Our study provided a valuable method to assess circular data in the sport performance domain, in particular the structure of variability through break-points in the upper-limbs and lower-limbs angle time series. Our results of this change-point analysis indicated higher levels of variability of limb angles for experts than for beginners suggesting greater dependence on the properties of the performance environment and adaptive behaviors in expert climbers. Conversely, the lower variance of limb angles assessed in beginners may reflect their independence of the environmental performance constraints, since they focused on controlling body equilibrium. Finally, by a structural analysis of variability, our method enabled the detection and understanding of the break-points causing plateaux and a lack of climbing fluency in order to improve the learning process in sport.

References

- [1] J. Antoch and M. Hušková, *Procedures for the detection of multiple changes in series of independent observations*, *Insur.: Math. Econ.* 16 (1995), pp. 268–268.
- [2] J. Antoch and M. Hušková, *Permutation tests in change point analysis*, *Stat. Probab. Lett.* 53 (2001), pp. 37–46.
- [3] M. Basseville and I. Nikiforov, *Detection of Abrupt Changes: Theory and Application*, Prentice Hall Information and System Sciences Series, Prentice Hall Inc., Englewood Cliffs, NJ, 1993.
- [4] P. Batoux and L. Seifert, *Ice Climbing and Dry Tooling: From Leman to Mont Blanc*, JM Editions, Chamonix, France, 2007.
- [5] P. Beek, D.M. Jacobs, A. Daffertshofer, and R. Huys, *Expert performance in sport: Views from the joint perspectives of ecological psychology and dynamical systems theory*, in *Expert Performance in Sports: Advances in Research on Sport Expertise*, J.L. Starkes and K.A. Ericsson, eds., Human Kinetics Publishers, Champaign, IL, 2003, pp. 321–344.
- [6] A. Benveniste and M. Basseville, *Detection of abrupt changes in signals and dynamical systems: Some statistical aspects*, in *Analysis and Optimization of Systems*, Lecture Notes in Control and Information Sciences, LNCIS-62, A. Bensoussan and J.L. Lions, eds., Springer-Verlag, Berlin, DE, 1984, pp. 145–155.
- [7] P. Bertrand, *A local method for estimating change points: The ‘hat-function’*, *Stat.: J. Theor. Appl. Stat.* 34 (2000), pp. 215–235.
- [8] P. Bertrand, M. Fhima, and A. Guillin, *Off-line detection of multiple change points by the filtered derivative with p-value method*, *Sequential Anal.* 30 (2011), pp. 172–207.
- [9] R. Bhattacharya and V. Patrangenaru, *Large sample theory of intrinsic and extrinsic sample means on manifolds: I*, *Ann. Stat.* 31 (2003), pp. 1–29.
- [10] R. Bhattacharya and V. Patrangenaru, *Large sample theory of intrinsic and extrinsic sample means on manifolds: II*, *Ann. Stat.* 33 (2005), pp. 1225–1259.
- [11] C. Bourdin, N. Teasdale, V. Nougier, C. Bard, and M. Fleury, *Postural constraints modify the organization of grasping movements*, *Hum. Mov. Sci.* 18 (1999), pp. 87–102.
- [12] B. Brodsky and B. Darkhovsky, *Nonparametric Methods in Change-Point Problems*, vol. 243, Mathematics and Its Applications, Kluwer, Dordrecht, 1993.

- [13] B. Charlier, *Necessary and sufficient condition for the existence of a fréchet mean on the circle*, Arxiv preprint (2011). Available at arXiv:1109.1986.
- [14] J. Chow, K. Davidsb, R. Hristovskic, D. Araújo, and P. Passos, *Nonlinear pedagogy: Learning design for self-organizing neurobiological systems*, *New Ideas Psychol.* 29 (2011), pp. 189–200.
- [15] J.F. Coeurjolly and N. Le Bihan, *Geodesic normal distribution on the circle*, *Metrika* 75 (2012), pp. 977–995.
- [16] M. Csörgö and L. Horváth, *Limit Theorems in Change-Point Analysis*, Wiley, Chichester, 1997.
- [17] K. Davids, S. Bennett, and K. Newell, *Movement System Variability*, Human Kinetics Publishers, Champaign, IL, 2006.
- [18] J. Gibson, *The Ecological Approach to Visual Perception*, Houghton-Mifflin, Boston, MA, 1979.
- [19] T. Hotz and S. Huckemann, *Intrinsic means on the circle: Uniqueness, locus and asymptotics*, Arxiv preprint (2011). Available at arXiv:1108.2141.
- [20] S. Huckemann, T. Hotz, and A. Munk, *Intrinsic shape analysis: Geodesic PCA for Riemannian manifolds modulo isometric lie group actions*, *Stat. Sin.* 20 (2010), pp. 1–58.
- [21] M. Hušková and S. Meintanis, *Change-point analysis based on empirical characteristic functions of franks*, *Sequential Anal.* 25 (2006), pp. 421–436.
- [22] S. Jammalamadaka and A. Sengupta, *Topics in Circular Statistics*, World Scientific Pub. Co. Inc., Singapore, 2001.
- [23] D. Kaziska and A. Srivastava, *The Karcher mean of a class of symmetric distributions on the circle*, *Stat. Probab. Lett.* 78 (2008), pp. 1314–1316.
- [24] Y.T. Liu, G. Mayer-Kress, and K.M. Newell, *Qualitative and quantitative change in the dynamics of motor learning*, *J. Exp. Psychol.: Hum. Percept. Perform.* 32 (2006), pp. 380–393.
- [25] K. Mardia, *Statistics of Directional Data*, Academic Press, London, 1972.
- [26] K. Mardia and P. Jupp, *Directional Statistics*, Wiley, Chichester, 2000.
- [27] P. Molenaar and K. Newell, *Individual Pathways of Change: Statistical Models for Analyzing Learning and Development*, American Psychological Association, Washington, DC, 2010.
- [28] K. Newell, *Constraints on the development of coordination*, in *Motor Development in Children: Aspects of Coordination and Control*, M. Wade and H.T.A. Whiting, eds., vol. 34, Martinus Nijhoff, Dordrecht, 1986, pp. 341–360.
- [29] K.M. Newell, Y.T. Liu, and G. Mayer-Kress, *Time scales in motor learning and development*, *Psychol. Rev.* 108 (2001), pp. 57–82.
- [30] K.M. Newell, G. Mayer-Kress, S.L. Hong, and Y.T. Liu, *Adaptation and learning: Characteristic time scales of performance dynamics*, *Hum. Mov. Sci.* 28 (2009), pp. 655–687.
- [31] X. Pennec, *Intrinsic statistics on Riemannian manifolds: Basic tools for geometric measurements*, *J. Math. Imaging Vis.* 25 (2006), pp. 127–154.
- [32] R. Schmidt and T. Lee, *Motor Control and Learning: A Behavioral Emphasis*, 5th ed., Human Kinetics Publishers, Champaign, IL, 2011.
- [33] F. Sibella, I. Frosio, F. Schena, and N.A. Borghese, *3d analysis of the body center of mass in rock climbing*, *Hum. Mov. Sci.* 26 (2007), pp. 841–852.
- [34] J. Summers and J. Anson, *Current status of the motor program: Revisited*, *Hum. Mov. Sci.* 28 (2009), pp. 566–577.
- [35] B. Vereijken, R.E.A. van Emmerika, H.T.A. Whitingb, and K.M. Newellc, *Freezing degrees of freedom in skill acquisition*, *J. Motor Behav.* 24 (1992), pp. 133–142.
- [36] D. Winter, *Biomechanics of Human Movement*, Wiley, New York, 1979.
- [37] P. Zanone and J. Kelso, *Evolution of behavioral attractors with learning: Nonequilibrium phase transitions*, *J. Exp. Psychol.: Hum. Percept. Perform.* 18 (1992), pp. 403–421.

A.6 Comparing Dynamics of Fluency and Inter-Limb Coordination in Climbing Activities Using Multi-Scale Jensen-Shannon Embedding and Clustering

Reference

[Hér+17] Romain Hérault et al. “Comparing Dynamics of Fluency and Inter-Limb Coordination in Climbing Activities Using Multi-Scale Jensen-Shannon Embedding and Clustering.” In: *Data Mining and Knowledge Discovery* 31.6 (Nov. 2017), pp. 1758–1792. DOI: 10.1007/s10618-017-0522-1. URL: <https://hal.archives-ouvertes.fr/hal-02094958>

Comparing dynamics of fluency and inter-limb coordination in climbing activities using multi-scale Jensen–Shannon embedding and clustering

Romain Hernaut¹ · Dominic Orth² ·
Ludovic Seifert³ · Jeremie Boulanger⁴ ·
John Aldo Lee⁵

Received: 29 February 2016 / Accepted: 12 June 2017 / Published online: 22 June 2017
© The Author(s) 2017

Abstract This paper reports the results of two studies carried out in a controlled environment aiming to understand relationships between movement patterns of coordination that emerge during climbing and performance outcomes. It involves a recent method of nonlinear dimensionality reduction, multi-scale Jensen–Shannon neighbor embedding (Lee et al., 2015), which has been applied to recordings of movement sensors in order to visualize coordination patterns adapted by climbers. Initial clustering at the climb scale provides details linking behavioral patterns with climbing fluency/smoothness (i.e., the performance outcome). Further clustering on shorter time intervals, where individual actions within a climb are analyzed, enables more detailed exploratory data analysis of behavior. Results suggest that the nature of individual learning curves (the global, trial-to-trial performance) corresponded to certain behavioral patterns (the within trial motor behavior). We highlight and discuss three distinctive learning curves and their corresponding relationship to behavioral pattern emergence, namely: no improvement and a lack of new motor behavior emergence;

Responsible editors: Ulf Brefeld and Albrecht Zimmermann.

✉ Romain Hernaut
romain.herault@insa-rouen.fr

¹ LITIS, INSA Rouen, UNIROUEN, UNIHAVRE, Normandie Univ, 76 000 Rouen, France

² Faculty of Behavioral and Movement Sciences, Amsterdam Vrije Universiteit, Amsterdam, The Netherlands

³ Normandie Univ, UNIROUEN, CETAPS, 76000 Rouen, France

⁴ CRISAL, Bâtiment P2, Avenue Carl Gauss, Université Lille 1, 59655 Villeneuve d’Ascq Cedex, France

⁵ Research Associate with the Belgian F.R.S-FNRS, Molecular Imaging, Radiotherapy, and Oncology — IREC, Université catholique de Louvain, Avenue Hippocrate 55, 1200 Brussels, Belgium

sudden improvement and the emergence of new motor behaviors; and gradual improvement and a lack of new motor behavior emergence.

Keywords Performance management · Climbing skills profile · Climbing patterns dynamics · Non-linear dimension reduction

1 Introduction

Valuation of climbing activities can be estimated through climbing efficiency and climbing behavioral skills, such as inter-limb coordination. On the one hand, climbing efficiency, which partially relates to fluency, can be relatively easily assessed by performance indicators based on spatial and temporal computations of the hip trajectory. These include the jerk coefficient (spatial-temporal indicator), geometric entropy (spatial indicator), and a ratio of immobility to mobility (temporal indicator) (Seifert et al. 2014b; Orth et al. 2016). On the other hand, behavioral patterns which could be examined through limb kinematics (3D orientation, angular velocity, linear acceleration, etc) is straight forward to retrieve, but, extremely challenging to analyze. Therefore, few studies have tackled the qualification of inter-limb coordination. This level of analysis, however, provides a mechanistic insight into how the climber appropriately coordinates their four limbs together with trunk movement (Orth et al. 2016). Thus, detection and qualification of inter-limb coordination in climbing activities play a key role in performance management. In practical terms, correct qualification of inter-limb coordination may help the practitioner to guide individuals toward learning more efficient behavioral patterns. In order to do so, empirical tools are needed for determining how the coordination and regulation of action influences performance efficiency. These concerns also highlight the scales of analysis needed for determining how increased movement variability can be associated with skill (Bernstein et al. 1996). Indeed, a larger repertoire of inter-limb coordination patterns appears to help experienced/skilled climbers adapt to variations in constraints (such as size, shape of hold, distance between holds, onset of fatigue, use of chalk). For example, skilled climbers are able to both, switch between more patterns of movement coordination (Seifert et al. 2014b), and, within these patterns, make a greater magnitude of adjustment (Seifert et al. 2013). In both cases, through an increase in functional, or goal supportive, movement variability, skilled climbers are more fluent in terms of overall performance (Seifert et al. 2014c). Observing the practice over time is a necessary step to understand whether individuals are able to learn new and adaptive techniques (where adaptivity helps the individual to improve performance). For example, individual differences in the rate and nature (e.g., presence of discontinuities) of learning can be revealed by observing performance on a trial-to-trial basis, sometimes termed assessment of learning dynamics (Nourrit et al. 2003; Kostrubiec et al. 2012). Indeed, when considering behavior at the group level, the nature of individual learning curves are not apparent, potentially leading to the (incorrect) assumption that learning follows a linear progression. Ideally, the practitioner can use the individual as the frame of reference when modifying practice constraints, and, therefore, accurate understanding of the *nature* of learning complex multi-articular skill *at the individual level* is essential.

The main question this study addresses is *do climbers improve simultaneously, linearly and in a proportional way both their climbing fluency and their behavioral skills?* Our hypothesis is *no* because the learning process can include an alternation of *exploitation* of the initial repertoire of skills and the *exploration* of new skills. This means that when climbers exploit their repertoire of behavioral patterns, they optimize known skills and climbing fluency improves as well. Conversely, when climbers explore new possibilities, climbing fluency might temporarily decrease, but over longer timescales, their performance may improve.

To achieve that aim, automatic and theoretically consistent methodologies are required to track climbing actions throughout practice. Past studies on coordination focus on statistics of the relative phase of two limbs that are seen as two oscillators. (Kelso 1984; Bardy et al. 2002; Teulier and Delignieres 2007). However, climbing involves more than two limbs and is not a cyclic task (where in cyclical tasks limbs act as oscillators following a periodic signal, obviously climbing breaks this assumption).

In a previous study (Boulanger et al. 2016), the task of climbing was defined as an alternation of moments of movement and partial or total immobility. Movement means that the climber's limbs support the hip in upward progression, whereas immobility means that the whole body does not move. However, partial immobility, where only some limbs are interacting with the environment (e.g., to explore how to grasp a new hold) while the hip does not move, can also be taken into account (Pijpers et al. 2006). This allows the comparison of exploratory as performatory movements as functional actions. Where in exploratory actions the end effector comes into contact with a hold but is not subsequently used to support the body, with the ensuing action being to withdraw the limb in order to make contact with the same or different hold. Of additional concern is that, as climbers have to regulate their body equilibrium, suggesting that partial immobility can also be observed when the hip is moving whilst the four limbs remain stationary. This former work has enabled automatic segmentation of a climb into 5 general activities states: Immobility; Postural regulation; Hold Exploration; Hold Change; Traction. Nevertheless, it has not been applied to a long dataset involving an extended period of practice. Moreover, the approach is exclusively based on trunk/limb activities and not limb orientation or coordination. Hence, what is currently unclear, is how discrete actions, such as a traction or an exploratory reach, are related to a particular pattern of movement coordination.

Therefore, traditional methods and tools used to study coordination dynamics are currently limited in terms of the analysis of full body dynamics especially where a range of degenerate solutions can emerge that may (or may not) lead to improved performance through practice (Davids et al. 2006).

In taking a *Human Movement* perspective, the novelty of this article is to adapt machine learning methods to overcome current methodological limitations in linking movement variability with performance over the timescale of practice and at the individual level of analysis. Namely, we address 3 objectives,

- (A) to go on **full body analysis**, taking into account the 4 limbs and related trunk movement (as opposed to 2 limb oscillators); in order to do so, we will reduce the dimension of the data-set to visualize the climbing actions into features and categorize these by clustering.

- (B) to analyze how the clusters are distributed in time, i.e. to address the **dynamics of learning at the behavioral level**, in order to know whether some patterns are present at the beginning of the learning process, which could correspond to the existing repertoire; while other clusters appear later in the learning process, emerging through exploratory processes.
- (C) to analyze the **individual specificity** during learning. We expect that some participants learn faster than others, meaning that they switch more rapidly to a new pattern because they demonstrate a more effective exploration. Conversely, we also expect that some participants will exhibit a tendency resist change. Thus, we anticipate a link between the emergence (or lack thereof) of new actions and the improvement in performance. The later suggesting that putting, together the dynamics of the climbing fluency (performance outcome) and the dynamics of behavioral skills acquisition might reveal whether exploration is effective or not.

On the other hand, from a *Machine Learning* point of view, this article does not propose any new dimension reduction nor clustering techniques. Nevertheless, due to the nature of the data (temporal signal, 3D rotations, ...), we describe with special care how methods are adapted:

- We will recall how can be defined the geodesic distance, the mean, and the variance on the particular group of 3D rotations (Hall 2015).
- In our climbing data, structures are unknown and may appear on different scales: climbers, holds, paths, climbing order, learning curve, ... Nevertheless, standard clustering or dimension reduction methods, such as stochastic neighbor embedding (SNE), are known to be good at structure preservation only for a particular scale. Recently, multi-scale Jensen–Shannon neighborhood embedding (Lee et al. 2015) solves this problem by opting for multi-similarity approaches. This multi-scale method will be applied to the output of motion sensors in order to help the visualization of behaviors even if they appears at different scales.

Two studies from two practicing/learning protocols are presented. In the first one, the semantic unit that will be clustered is a full climb, in order to have a general idea of the link between behavioral skills and the fluency. In the second study, we step down one level, and do the analysis on a segment scale which is a time interval in the climb where coherent actions are performed.

This work is decomposed in five sections:—a first section exposed how the climbing protocols, i.e., *Learning protocols*, are set up, and how signals are recorded;—the next section, *Building features*, is dedicated on how raw signals from the sensors are segmented and how features are extracted;—in the third section, *Dimensionality reduction*, we briefly recall how single- and multi-scale stochastic neighbor embedding works;—then, the results of the dimension reduction and the clustering on the extracted features are exhibited in *Per climb* and *Per segment* experimentations.

2 Learning/practicing protocols

Two separate climbing protocols were undertaken. The first campaign was based on an experiment originally designed to test the effect of ability level when practicing on

different routes over a small number of repetitions. The second campaign was based on a beginner group of individuals who practiced over an extended period of time on the same route. In each experiment, participants were required to undertake a climbing task designed to represent conditions normally met in commercial climbing gyms and pedagogical settings. Indeed they took place at a local climbing wall and globally across both experiments the task involved climbing to the top of a 10.3 m high vertically aligned wall, using artificial holds bolted to the surface. Prior to each ascent a 3 min period to visually inspect the route was afforded and between trails a 5 min seated rest was enforced. In order to prevent any risk of injury should a fall occur, each climb was top-roped, meaning a rope was passed through a bolt at the top of the route, with one end connected to the climber and the other counterbalanced by a belayer. Participants were instructed in all cases to self-pace their ascent, with the following task-goal: explore the way to climb as fluently as possible, i.e. without falling down while minimizing pauses and twitches of the body displacement. Instructions were not made too specific to allow new coordination patterns to emerge during exploratory behavior under the varying task constraints. The protocols were approved by the local University ethics committee and comply with the declaration of Helsinki ([World Medical Association 2013](#)), a set of ethical rules that apply for research on Humans. Procedures were explained to the climber, who then gave written informed consent to participate.

Please note that in the following the term *learning* is used synonymous with practice, where observations are made at each trial of practice, thus, in this context, the general nature of behavior and performance exhibited from one trial to the next indicating the effect of learning.

2.1 First campaign specificity

The learning protocol for the first recorded dataset consisted of four climbing sessions, separated by 2 days of rest. Each session consisted of ascending randomly three different routes graded 5b–5c in the French Rating Scale of Difficulty (F-RSD) (ranging from 1 to 9). Each path was identifiable by color and was set on an artificial indoor climbing wall by three professional certified route setters who ensured that routes match intermediate climbing ability.

The three routes had the same height (10.3 m) and they included the same number of hand-holds (20), which were bolted to a flat surface inclined at 90° from the horizontal. The holds were located at the same place on the artificial wall; only the orientation of the hold was changed:

- (i) the horizontal-edge route was designed to allow horizontal hold grasping,
- (ii) the vertical-edge route was designed to allow vertical hold grasping, and
- (iii) the double-edge route was designed to allow both horizontal and vertical hold grasping.

To emphasize, in this later condition, each hold had two edges: a horizontal edge that could be grasped in a manner with the knuckles in-line running parallel to the horizontal axis, and a vertical edge that could be grasped in a manner with the knuckles parallel to the vertical axis. Each edge could also be grasped by the left and/or the right hand. At the fourth session, the participants climbed a fourth path, which mixed

the hold types of the three previous routes. The first six holds only allowed horizontal grasping, then the seven next holds only allowed vertical grasping, while the seven last holds allowed horizontal and vertical grasping. These routes were originally designed to assess the capability of the climbers to reinvest the grasping patterns they learned in the first routes to the last and corresponded to a transfer test for further studies. This question is not developed in this work.

Fourteen participants voluntarily took part in this study, with mean age 22.7 ± 2.9 years, mean height: 176 ± 5 cm, mean weight: 64.2 ± 5.8 kg. Seven individuals in this group, at the time of participating had practiced indoor climbing for 3 years, 3 h per week and had a skill level in rock climbing of grade 6a–6b in the F-RSD, which represents an intermediate level of performance. Seven other participants had only practiced for 10 h and have a skill level of climbing of grade 5b–5c, which corresponds to a novice level of performance. In the following sections, participant names have been masqueraded.

2.2 Second campaign specificity

The learning protocol consisted of 14 climbing sessions in total. These were distributed twice weekly over 7 weeks and separated by no less than 2 days of rest. Each session consisted in ascending the same route three times. The route was graded at 5b F-RSD (as for first campaign, a typical grade used to challenge beginner adults) and consisted of a total of 40 green colored artificial holds. The hold characteristics and relative positioning were designed to allow the possibility of exploration of different grasping actions, body orientations and pathways through the route. Specifically, a single horizontal edge (running parallel to the ground plane) allowed use of an overhand grip and two vertical edges (running perpendicular to the ground plane) allowed opposing grips. Additionally, different route pathways were designed into the route allowing the exploration of a left, middle and right pathway through the route (see the middle image in Fig. 1).

Eight individuals (mean age: 20.2 ± 2.2 years, mean height: 173.9 ± 8.8 cm, mean weight: 60.3 ± 10.7 kg) voluntarily participated in this study. Inclusion criteria required that participants be within the healthy BMI range (<25) and have an arm span of no less than 140 cm. The participants were also required to have practiced no more than 10 h.

On a separate day prior to the experimentation, participants were screened by asking them to climb other beginner routes at a level of 5b to ensure they had a skill level of climbing within 5a–5c. This procedure also served to familiarize the participants with the data collection equipment and the safety procedures involved in top-roped climbing.

2.3 Instrumentation and recorded data

In both campaigns, the directions of the trunk and the limbs (3D unit vectors in Earth reference) have been collected from small, wearable, inertial measurement units (IMU). IMUs corresponded to a combination of a tri-axial accelerometer ($\pm 8G$),

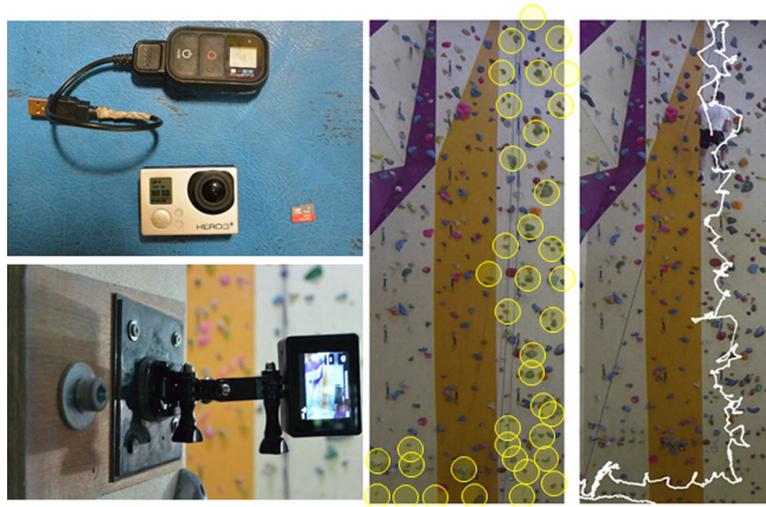


Fig. 1 Instrumentation for collecting trajectory and route's relative hold positions

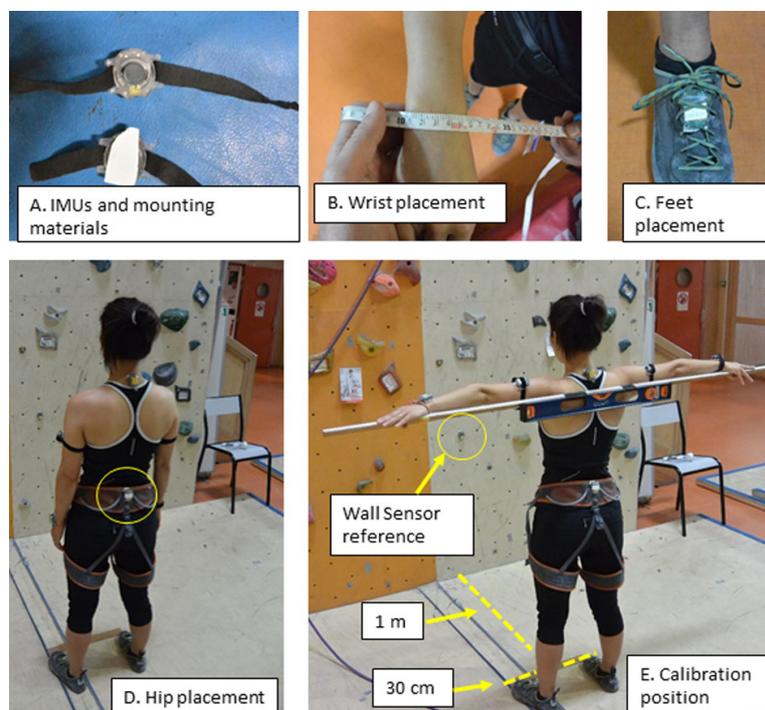


Fig. 2 Sensor placement and relative position calibration procedures

tri-axial gyroscope (1600/s) and a tri-axial magnetometer (MotionPod, Movea©, Grenoble, France; Seifert et al. 2014a). Data collected from the IMUs were recorded with North magnetic reference at 100 Hz and transmitted by wireless connection with a control unit run off a desktop operating system. IMUs were attached to five locations (wrists and feet and hip) chosen to ensure that climbing movements would not be interfered with, whilst, also minimizing displacement artifact due to underlying muscle (see Fig. 2). The sensors and their relative placement locations, orientations and procedures were used throughout the entirety of the experimentation.

In order that behavior could be qualitatively contextualized with respect to our analyses, each ascent was also captured with a frontal camera fixed 9.5 m away from

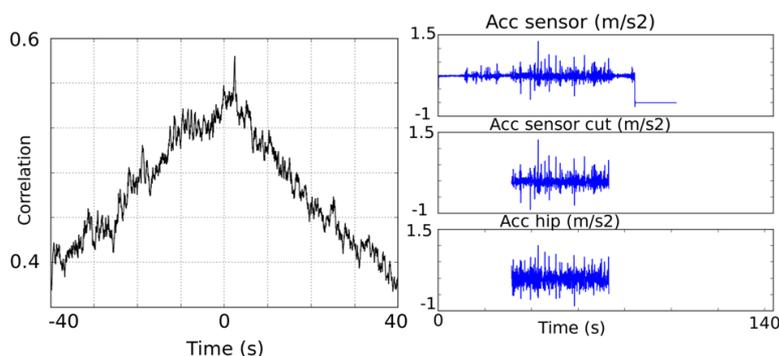


Fig. 3 Synchronization of the sensor and video based time series

the climbing wall and at a distance of 5.4 m from the ground and operated via remote wi-fi with images recorded directly to a SD card. As the back of the climber harness was equipped with a red light, we could easily track the position of the climber with simple video technique: Using color spotting and a Kalman filter to eliminate false detection (in this case, the presence of televisions in the recorded picture), we obtain for each video frame the position of the harness and thus the hip position (Boulanger et al. 2016). Nevertheless, the camera and worn IMU sensors were not synchronized, which means for example that the frame at 10s for the camera does not correspond to the sample at 10s for the sensors. The correction delay could be computed using the correlation of the hip/pelvis acceleration signals which can be found in the two different modalities. The top right plot of Fig. 3 presents the hip/pelvis acceleration recorded from the IMU sensors for one climb, the bottom right plot, shows the same signal coming from the video tracking. Their correlation is displayed at the left part of the figure. As the maximum of the curve is not on 0, these signals are not in sync and IMU needs to be delayed to around 3 seconds which is the position of the correlation extreme. The IMU drifted signal is then cut to the interval of interest: first discernible contact made from a quadruped support to first discernible contact made with the final hold with both hands. In this example, the resulting signal is displayed on the middle right plot of Fig. 3.

3 Building features

This section describes all the pre-processing steps occurring prior to dimensionality reduction and the clustering.

From the raw signals recorded on the climbers (gyroscope, accelerometer, and magnetometer on the hip and on each limb), three kinds of features are extracted:

- general smoothness/fluency information (namely, Entropy, Jerk, and immobility ratio), one per climb;
- climb segmentation and segment classification;
- relative orientation of each sensors through rotations.

Further pre-processing combines the last two items by computing segment by segment the mean and variance of the rotations. Because of the group-wise nature of

rotations, their statistics are not standard statistics and they must be carefully aggregated for the next stage of the analysis: dimensionality reduction and clustering.

3.1 Fluency indicators

The index of the geometric *Entropy* (GE) is a ratio between the length of a trajectory and the perimeter length of its convex hull (Cordier et al. 1993). For a given trajectory from time 0 to time T , $x : [O, T] \rightarrow \mathbb{R}^3$, let's be Δx the trajectory length and $\Delta c(x)$ the convex hull perimeter. The index of the geometric entropy is given by

$$S_x = \frac{\log(2 * \Delta x) - \log(\Delta c(x))}{\log(2)},$$

where the division by $\log(2)$ is here to provide the entropy in bits.

Smoothness of the hip trajectory can be measured by estimating the *Jerk*, which is the derivative of the acceleration. For a given trajectory from time 0 to time T , $x : [O, T] \rightarrow \mathbb{R}^3$, the dimensionless jerk is defined as

$$J_x = \frac{T^5}{(\Delta x)^2} \int_0^T \left\| \frac{d^3 x}{dt^3}(s) \right\|^2 ds,$$

where Δx is the length of the trajectory. The jerk proves to be an indicator of the expertise skills (Seifert et al. 2014b) in climbing activities.

GE is only based on spatial measurement (hip trajectory) whereas jerk is a spatial-temporal measurement. As we want to compare fluency indicators and behavioral clusters fairly, smoothness information are not aggregated with other features in data to be projected and clustered.

3.2 Segmentation

Regarding each 5 sensors separately, signals are segmented into 2 states:—the sensor is *moving*,—or the sensor is *immobile*.

In order to perform this segmentation, the acceleration and the angular velocity are passed through a CUSUM algorithm (Basseville and Nikiforov 1993), which is a sequential analysis method based on a recursive hypothesis test. Empirical histograms of *moving* and *immobile* states were obtained through manual labeling by an expert climber on some climbing videos. From the shape of these histograms, Γ -distributions have been elected for the statistical test. Their parameters are estimated on these very same human annotations.

An example of signal segmentation is displayed on Fig. 4. From the top to the bottom, it shows the norm acceleration signal (in green) along with the synchronized manual annotation from the video record (in blue); below this, the log likelihood ratio curve (in black), or LLR, is the recursive test ratio given by the CUSUM algorithm which leads to the detected segments (in red).

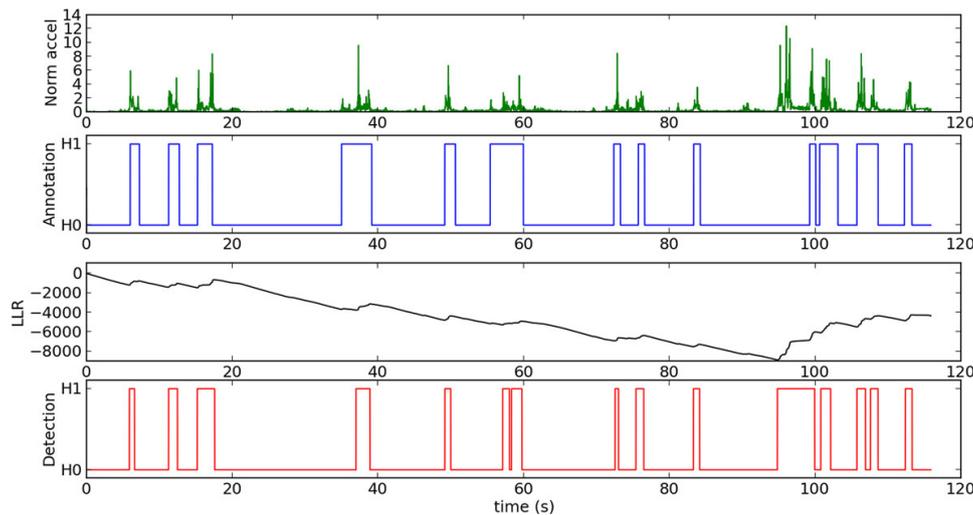


Fig. 4 CUSUM segmentation based on the acceleration norm

Using the aggregation of the segmentations on each 5 sensors, a global body state is then determined by the following rules:

1. All sensors are immobile \Rightarrow **Immobility**;
2. Only the hip sensor is mobile \Rightarrow **Postural regulation**;
3. Hip sensor is immobile, limbs sensors are mobile \Rightarrow **Hold Exploration**;
4. Last *Hold Exploration* before a traction \Rightarrow **Hold Change**;
5. Some limbs sensors and the hip sensors are mobile \Rightarrow **Traction**.

The exact details of this procedure is not the scope of this article. One can find extensive description in [Boulanger et al. \(2016\)](#).

On one hand, an immobility ratio is computed per climb from this segmentation. This indicator joins the jerk and entropy as smoothness indicators and is not directly used in the up-coming clustering.

On the other hand, this global body segmentation gives the timing frames for extracting statistical indicators; that is for each kind of metrics, one is computed per segment.

3.3 From sensors to rotations

In order to prepare dimension reduction for qualitative human interpretation, gyroscope, accelerometer and magnetometer information are converted into a 3×3 rotation matrix that describes each sensor in an Earth frame (North, West, vertical).

Small angular changes, instant acceleration or magnetic field direction given directly by sensors for each limb are good to segment the climber ascent by an automatic process but they are hardly directly interpretable in a human perspective. Moreover, gyroscope is accurate for angular changes over very a short time duration but due to a drift its direct integration to get the absolute angle is not reliable. In the meantime, the measurement of acceleration is noisy and also can not be directly integrated to get speed and position.

Therefore, combining the two sensors accelerometer and gyroscope in a rotation signal provides us a better signal/noise ratio; the third sensor, the magnetometer, is used

to get the Earth reference. The transformation is performed through a complementary filter based algorithm described in [Madgwick \(2010\)](#) and [Madgwick et al. \(2011\)](#).

The resulting informations (axis,angle) are the components of a 3D rotation which is encoded into a 3×3 matrix. Here, it does not mean the limb is rotated around the *axis* by the *angle*; but, if a sensor were put a finger, the rotation *axis* gives where the finger points at, and the rotation *angle* tell us how much the palm points at the sky. Hence, the limb direction is parallel to the rotation axis! This is enough to reconstruct the relative limb positions if needed.

3.4 Computing metrics and statistical indicators on rotations

We want to extract the rotation mode and the rotation variance for each limb on each segment in order to have an idea of the body position and the body variation scenario over a climb.

Nevertheless, the mean of rotations is not the element-wise mean of rotation matrices. Indeed, rotation matrices belong to a compact manifold, the Lie group of rotations $\mathfrak{so}(3)$, and thus standard metrics and statistics do not apply. This subsection is dedicated on how can be defined the geodesic distance, the mean, and the variance on that particular group ([Hall 2015](#)).

3.4.1 Rotation distance

The geodesic distance between rotations A and B is defined as the angle of the composition C of rotation B and the inverse of rotation A . If A and B are the same rotations then the angle of C is null. In the rotation group, the inverse of a matrix is simply its transpose. This gives,

$$d(A, B) = \arccos \left(\frac{\text{tr}(A^T \cdot B) - 1}{2} \right).$$

3.4.2 Rotation mean

A rotation geodesic mean M is defined as a rotation that minimizes the sum of geodesic distances between itself and the studied set of rotations. It may not be unique!

The computation of the rotation mean M of n rotations R_i with $i \in [1 \dots n]$ involved the following iteration process ([Manton 2004](#)):

1. Initialize M_0 to one of the R_i ,
2. Project each R_i to the tangent space of the rotation manifold in M_t ,

$$P_i = \log (M_t^T \cdot R_i),$$

3. Compute the mean of P_i and project it back to the manifold, leading to the new mean estimation,

$$M_{t+1} = M_t \cdot \exp\left(\frac{1}{n} \sum_i^n P_i\right),$$

4. go back to 2 until convergence.

As being a rotation itself, this mean also belongs to the Lie group of rotations.

Beware, the log and exp operators are the matrix operations not the element-wise operations. They are computationally costly. Nevertheless, the log of a rotation matrix R can be efficiently computed (Engø 2001) by

$$\log(R) = \frac{\arcsin(\|S\|)}{\|S\|} S, \quad \text{where } S = \frac{R - R^T}{2}.$$

3.4.3 Rotation variance

The variance V of a rotation set is defined by the mean of the squared geodesic distances of each rotation to the rotation mean M , namely,

$$V = \frac{1}{n} \sum_{i=1}^n d(M, R_i)^2.$$

As rotation mean M must minimize the geodesic distances, variance V is unique even if the rotation mean M is not. Due to the fact that rotation distances are expressed in angle, the unit of the variance is a squared angle.

3.4.4 Implication on the data metrics

Besides the computation of the statistics segment by segment, these tools are also used inside forth-coming machine learning algorithms. The dimension reduction and the clustering need distances between examples, but features are composed of

- Rotation means that are rotation themselves,
- Rotation variances that are squared angles.

That is why using a Euclidean distance between vectors of two examples is not suitable. One must combine two kinds of geodesic distances (between rotations or between angles on each corresponding part of the features) to get an accurate distance between examples.

4 Dimensionality reduction

Direct observation of the rotation climbing features is hardly interpretable. Indeed, rotations are easier to understand for humans than the raw sensors signals but their

aggregation leads to more than 200 real values per example in our first experimental setup.

This issue can be addressed by dimensionality reduction (DR). Features in the high-dimensional (HD) space are represented in a low-dimensional space that can be easily visualized, typically with two or three dimensions only. If this embedding reproduces correctly relevant structure from data in the HD space in the LD space, then visualization allows for a meaningful preliminary qualitative interpretation of data.

Almost all DR methods can be characterized by the particular kind of structure they try to preserve in the LD embedding. For instance, principal component analysis (PCA) attempts to best preserve the observed data variance. Most methods of multidimensional scaling try to reproduce dot-products (Torgerson 1952), (Euclidean) distances (Sammon 1969; Demartines and Hérault 1997), or just a monotonic function of those (Shepard 1962; Kruskal 1964). Here we focus on recent DR methods that preserve the neighborhood of data points (Hinton and Roweis 2002), which is less constraining and much more successful (van der Maaten and Hinton 2008) than distance preservation (Lee and Verleysen 2014). In practice, these DR methods try to embed dissimilar points far from each other and similar ones close to each other, taking into account only relative ordering of neighbors, not plain distances.

This work uses a variant of stochastic neighbor embedding (SNE) (Hinton and Roweis 2002). In this family of methods, soft neighborhoods are defined in both HD and LD space with normalized Gaussian similarities, which can be interpreted as the probability of some point to be a neighbor of some other point of reference. This relation is not symmetric, a bit like Korea would likely be the neighbor of China, while the opposite would not necessarily hold true.

Before reducing dimensionality, the user has to adjust hyper-parameter B , called *perplexity*, which can be interpreted as the size of the (soft) neighborhoods around each point. A value of 10 determines the individualized bandwidths of the Gaussian function centered on each data points, such that it covers about 10 neighbors, in spite of local density variations. The perplexity hence defines a relevant scale in data, usually rather small, and SNE attempts to preserve neighborhoods on that scale mainly. In our case, though, structure may arise on different scales: the climber, the path, or the order of the climbs. For this reason, this study relies on multi-scale Jensen–Shannon embedding (Ms.JSE) (Lee et al. 2015), a variant of SNE that involves banks of similarities with a range of several different bandwidths, in order to capture both local and global structure.

The rest of this section introduces briefly SNE and Ms.JSE.

4.1 Single-scale approaches

For the sake of the notational simplicity, Greek and Roman symbols refer to variables in the HD and LD spaces, respectively. The data set, consisting of N HD points, is written $\Xi = [\xi_1 \dots \xi_N]$, whereas corresponding LD points are written $X = [x_1 \dots x_N]$.

Normalized Gaussian similarities in the HD space can be defined as

$$\sigma_{ij} = \frac{\exp\left(-\delta_{ij}^2/(2\lambda_i(B)^2)\right)}{\sum_{k,k \neq i} \exp\left(-\delta_{ij}^2/(2\lambda_i(B)^2)\right)},$$

with $\sigma_{ii} = 0$ and where δ_{ij} is the distance between ξ_i and ξ_j . Bandwidth $\lambda_i(B)$ corresponds to the radius of the soft neighborhood centered on ξ_i th, in order to encompass B neighbors. In practice, $\lambda_i(B)$ is determined by forcing σ_{ij} to have an entropy equal to $\log(B)$, namely,

$$\log(B) = -\sum_j \sigma_{ij} \log(\sigma_{ij}).$$

Individualization of the bandwidth for each data point allows adapting to local density variations at the expense of breaking symmetry. Once all bandwidths are determined, HD similarities remain fixed in SNE.

The LD counterparts of HD similarities σ_{ij} can be defined in mostly the same way as

$$s_{ij} = \frac{\exp\left(-d_{ij}^2/2\right)}{\sum_{k,k \neq i} \exp\left(-d_{ik}^2/2\right)},$$

where d_{ij} is the Euclidean distance between between x_i and x_j . Note that no bandwidths are used here, in contrast with to HD ones.

The embedding process starts by initializing the LD points in X , either randomly or along principal components.

Now that we have the LD and HD similarities, how can we qualify the embedding? A Kullback–Leibler divergence D_{KL} measures the mismatch between HD and LD similarities, considered here as probability distribution, since they are normalized. This divergence is given by

$$D_{KL}(\sigma_i || s_i) = \sum_j \sigma_{ij} \ln\left(\frac{\sigma_{ij}}{s_{ij}}\right).$$

Lee et al. (2013) proposes to use the Jensen–Shannon divergence, a type-2 mixture of KL divergences,

$$D_{JS}^\kappa(\sigma || s) = \kappa D_{KL}(\sigma || \mathbf{z}) + (1 - \kappa) D_{KL}(s || \mathbf{z}),$$

where $\mathbf{z} = \kappa \sigma + (1 - \kappa)s$ and κ is the mixture parameter.

The embedding can then be seen as an optimization problem where the objective function to minimize over the LD points X is

$$J(X) = \sum_i^N D_{JS}(\sigma_i || s_i).$$

Iterative optimization techniques like gradient descent can be run until convergence or for a fixed number of iterations (Lee et al. 2013).

4.2 Multi-scale approach

One caveat of single-scale methods is that the result depends on the value of perplexity B . Lee et al. (2015) overcomes this issue by using a bank of similarities, accounting for neighborhoods on $L = \lfloor \log_2(N) \rfloor$ different scales, with preset perplexities ranging from $B_1 = 2$ to $B_L = 2^L$.

A global HD similarity is given by

$$\sigma_{ij} = \frac{1}{L} \sum \sigma_{ijh},$$

where σ_{ijh} is a similarity for bandwidth λ_{ih} , obtained with perplexity $B_h = 2^h$, namely,

$$\sigma_{ijh} = \frac{\exp\left(-\delta_{ij}^2 / (2\lambda_{ih}^2)\right)}{\sum_{k, k \neq i} \exp\left(-\delta_{ij}^2 / (2\lambda_{ih}^2)\right)}.$$

In contrast to the single-scale approach, bandwidths are introduced in the LD similarities, which are written as

$$s_{ijh} = \frac{\exp\left(-d_{ij}^2 / (2l_{ih}^2)\right)}{\sum_{k, k \neq i} \exp\left(d_{ik}^2 / (2l_{ih}^2)\right)}.$$

Here bandwidths l_{ih} cannot be computed by imposing the entropy value, since the LD coordinates in X are not known yet either. Instead, we fix $l_{ih} = 2^{h/P}$, where P is the target dimension. In other words the bandwidth grows like the radius of discs (or spheres) with doubling surfaces (or volumes). A global LD similarity is then computed as

$$s_{ij} = \frac{1}{L} \sum s_{ijh}.$$

The optimization procedure is mostly the same as in single-scale approach, except that small scale components of the similarities are introduced progressively, mainly to avoid getting stuck in poor local minima.

5 Preliminary per climb experiment

For the first experiment, the analysis is worked out at the level of a climb in the sense that the unit to be projected and clustered is a whole climb. Yet signal segmentation is

performed but segment indicators are gathered climb by climb. Moreover no distinction is made between *Hold Exploration* and *Hold Change* full body states.

5.1 Data set

The features extracted segment by segment in a climb are aggregated this way:

- Rotation signals are split into 20 sets corresponding to the 5 sensors and to the segmentation in 4 high-level states. For each of these sets, the rotation mean and variance are computed.
- Body states are summarized by state count and state transitions probabilities, which corresponds to Markov model parameters.

Thereby, each climb is represented by a vector of 220 continuous features decomposed in—20 rotation mean matrices in $\mathbb{R}^{3 \times 3}$,—20 rotation variances in \mathbb{R} ,—a state count vector in \mathbb{R}^4 ,—and a transition matrix in $\mathbb{R}^{4 \times 4}$. A dimension reduction and a clustering is performed on the aforementioned features.

5.2 Fluency analysis

As our objective is to compare the result of clustering to performance, the fluency indicators are not included into data to be reduced and clustered. Indeed, jerk alone during practicing has already been investigated in [Seifert et al. \(2014b\)](#). For the sake of completeness, we provide here a brief analysis of jerk in the first dataset. Figure 5 shows the distribution of jerk for each climber, on a log scale. In particular, the distribution from one climber to the other is seen to differ to a large amount. Beginners like Gerard can have low jerk. In order to see the effect of practicing without any climber bias on fluency, jerk can be normalized for each climber separately, with 1 then representing the highest value a climber has reached, and zero his lowest. The jerk distribution is then plotted, trial after trial, for all climbers, in Fig. 6. Jerk clearly decreases over successive trials, indicating that climbers globally improve their fluency through practice.

5.2.1 Embedding

Figure 7 shows the same Ms.JSE projection with two different annotations, the first one with climber labels, the second one with path labels. Each point represents a different climb.

For a particular climber (Fig. 7a), most of its climbs form between 1 to 3 clusters with few outliers. Each of these clusters can be seen as a coordination pattern specific to the climber. A general path/route effect appears in the projection with higher density zone for each of the path even if their instances are not clearly separated (Fig. 7b). Thus, Ms.JSE has succeeded in preserving these two scopes.

Now looking at the three clusters of climber *Henry* that have been highlighted in both scatter plots, we see each of the consolidations contains more than one path. This suggests that the clusters observed for one climber are not the consequence of a route

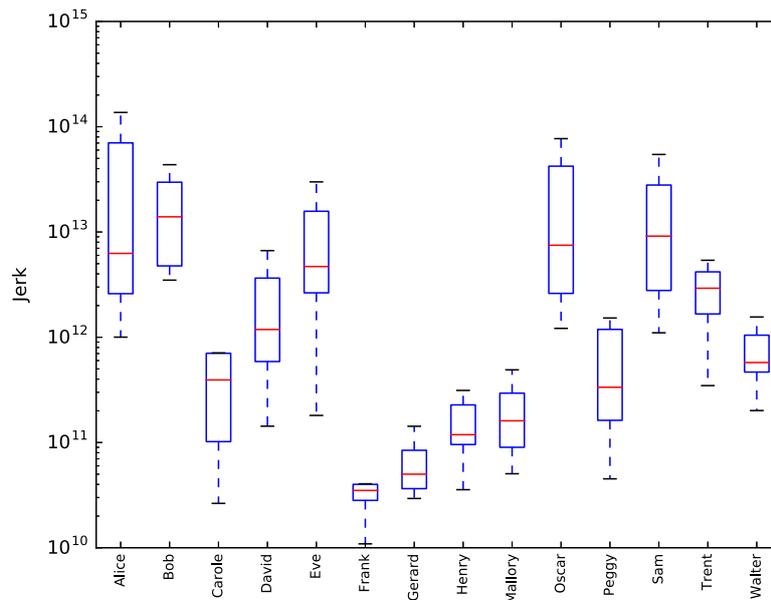


Fig. 5 Jerk distribution for each climber

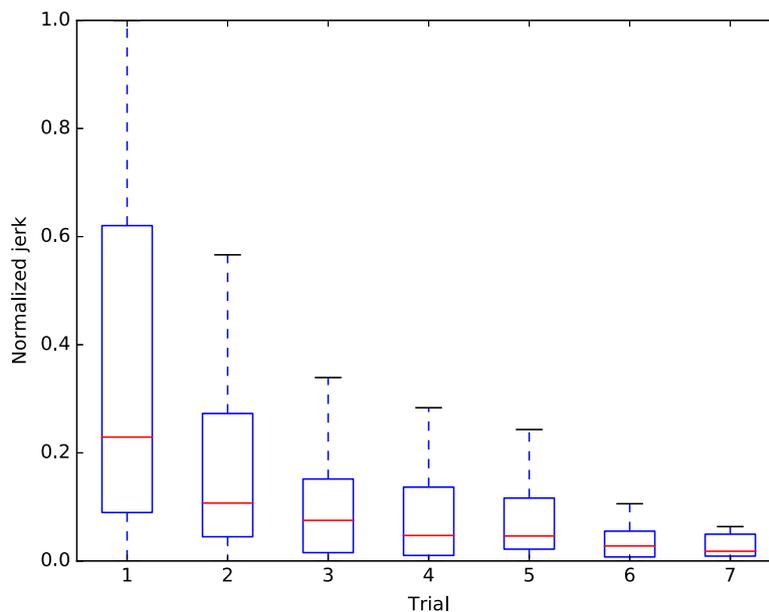


Fig. 6 Normalized jerk distribution, trial after trial

effect but may be due to a time effect. The route effects will not be discussed further in this article and we focus rather on the practice dynamics.

5.2.2 Clustering

To have a better picture of behavioral patterns exhibited during practice, we have applied hierarchical agglomerative clustering (HAC) with complete linkage, using as metric symmetrized LD similarities, i.e., the geometric mean of s_{ij} and s_{ji} . The tree has been cut to get 6 clusters. This number has been chosen by a Bayesian information criterion (BIC), a standard tool in clustering model selection.

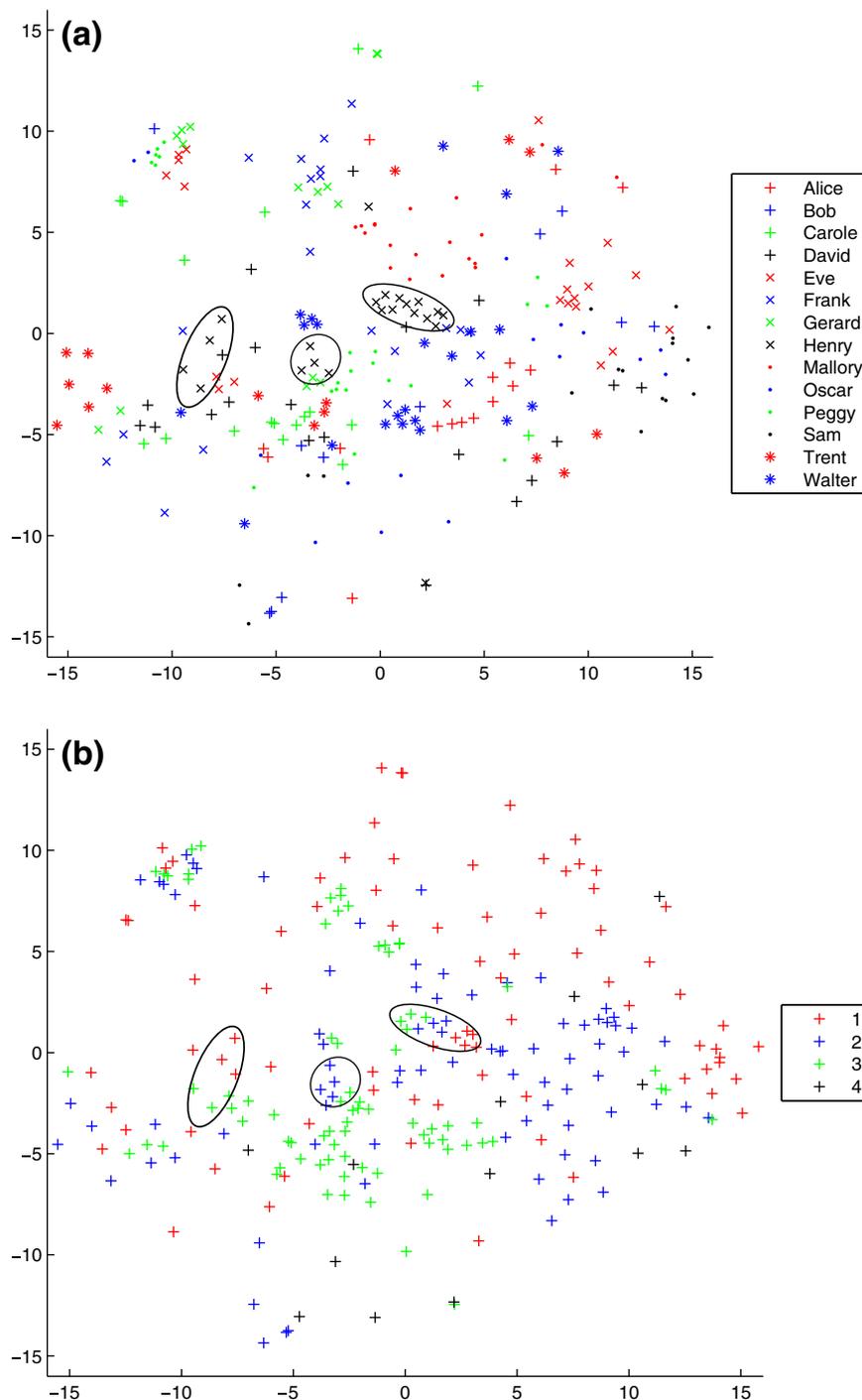


Fig. 7 Ms.JSE projection with *Henry* climbs surrounded. **a** By climbers. **b** By paths

Figure 8 shows the distribution of jerk cluster by cluster. Here, clusters are arranged in terms of mean trial position. *Light blue, orange* and *brown* dominate in the first trials. *Dark blue, cyan* and *green* are more often seen in the latter trials. This figure highlights that clusters and jerk do not correlate directly. Next, let us look at some subjects individually.

Figure 9 shows the result of clustering in a time-line for four selected climbers. Each graph is split into 4 parts: a time-line on all climbs, and 3 time-lines for each climbed route. The last fourth path is intended to test the transfer of skill by examining

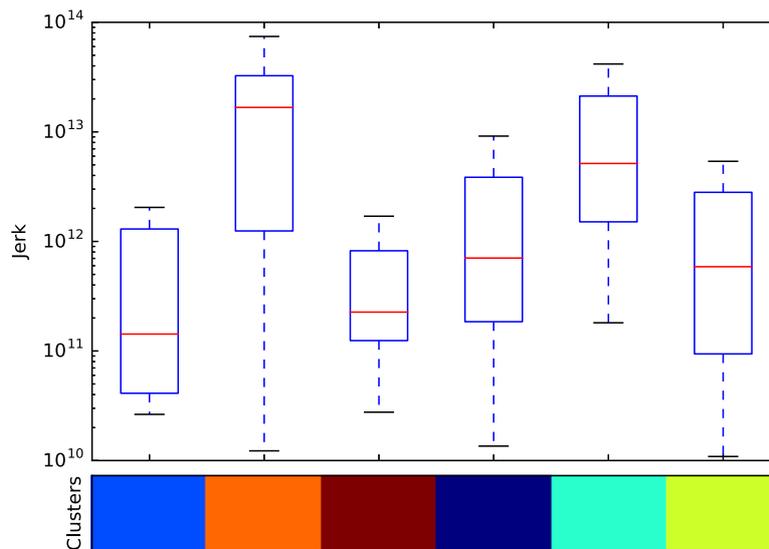


Fig. 8 Jerk distribution for each clusters

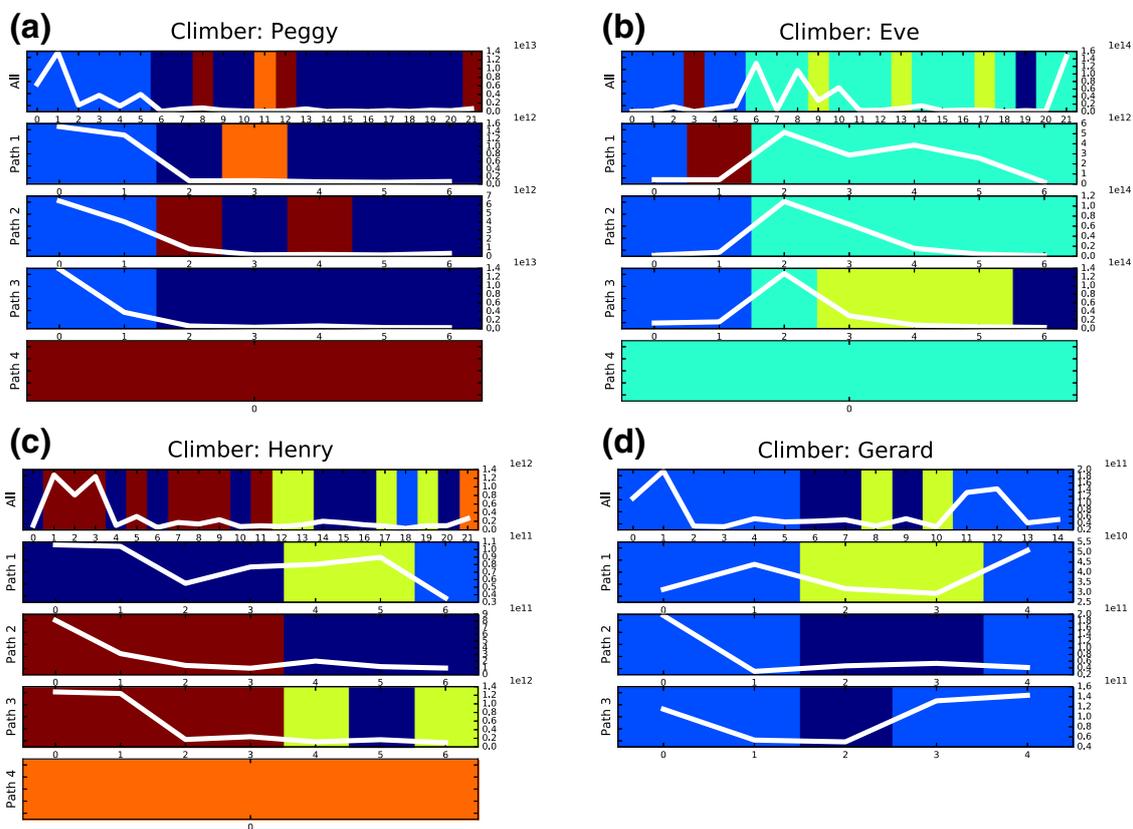


Fig. 9 Clustering and jerk displayed over time for selected climbers. The *first path* is the vertical grasp route, the *second path* the horizontal one, the *third path* contains both and the *fourth path* is the transfer route. *Peggy* and *Eve* are beginners while *Henry* and *Gerard* are experts. **a** *Peggy*. **b** *Eve*. **c** *Henry*. **d** *Gerard* (no path 4)

performance of a single trial on a new, unfamiliar route (corresponding to on-sight conditions in climbing). The background color indicates to which cluster a climb belongs, where colors are consistent across the climbers. The jerk has been plotted in white over the cluster time-line.

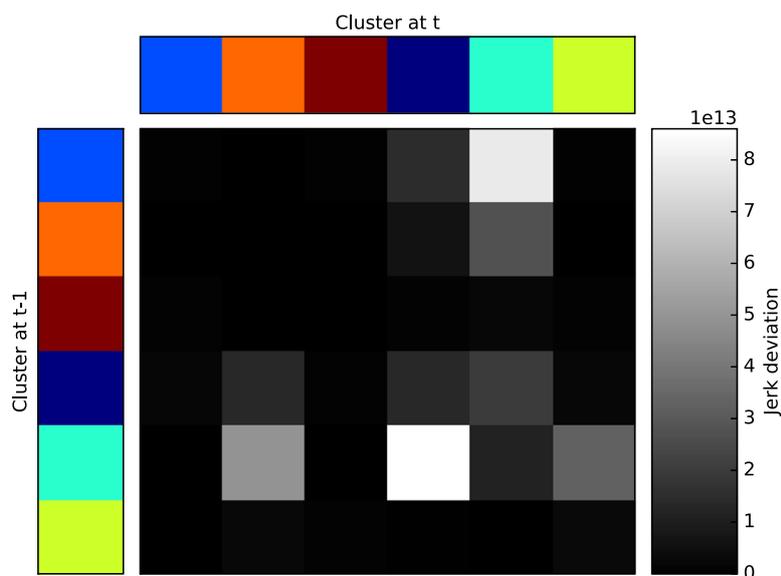


Fig. 10 Jerk change concurrent to cluster transition

Jerk variations are seen to be concomitant with cluster changes for *Peggy* and *EVE*. In order to confirm this observation over the entire data set, we have computed the mean over all climbers of the jerk deviation at cluster transition. When a climber stays in the same cluster, a mean jerk deviation of $7.89e12$ is displayed; when cluster changes, the value increases to $12.24e12$. Figure 10 shows a more detailed view of jerk deviation: rows indicate cluster at $t - 1$, columns cluster at t . A black cell in the central matrix means that no transition has occurred or a low jerk deviation. Conversely, a white cell reflects a high deviation value (positive or negative). The diagonal has lower deviation values, meaning that jerk is not likely to change much when a climber stays within the same cluster. The highest jerk deviation occurs when a climber is going from cluster *light blue* to *cyan* or from *cyan* to *dark blue*. More generally, when a climber enters *dark blue* or *cyan* clusters (acquired clusters), a higher jerk deviation is observed as well.

5.3 Preliminary analysis on the per climb clustering

For the first dataset clustering was undertaken on each trial to have a first evaluation of the degree of behavioral consistencies within and between individuals. Observing Fig. 9 the distinct clusters are exhibited for four individuals on trial-by-trial basis with the jerk indicator superimposed.

Jerk tends to decrease throughout practice (Seifert et al. 2014b), confirmed in this data set by Fig. 6. This assumption holds true in our individual inspection with the exception of the fourth route. At an individual level, Gerard demonstrates no improvement in his indicator at all; moreover, for Eve at trial 2 in all paths and for Henry for trials 3–4 in path 1, a local increase of the jerk occurs. The general trend is consistent with the fact that the more the subjects train, the better their fluency.

All individuals share some clusters (e.g., the light blue cluster). Conversely, some behaviors are individually specific: Eve for example shows unique orientation features,

that emerged with practice, not shared by any other climbers. These data highlight that certain behavioral consistencies can be expected within or between individuals whereas others can be unique, perhaps as a function of the individuals constraints.

Moreover, a change of repertoire (i.e., a change in used clusters) arises for Peggy, Eve, and Henry. Nevertheless, transitions either come up at the beginning or at the end of the learning process, suggesting that exploration during learning was experienced individually (as already highlighted by [Chow et al. 2008](#)). On his side, Gerard starts and finishes the experiment using the same repertoire, even if he temporarily enters new coordination clusters.

As a consequence, the per climb clustering indicates, that each climber displays different cluster profiles, and, that no static link can be made between cluster, practice and performance. For instance, the *Light Blue* cluster appears as a beginners repertoire with low performance for Peggy and Eve, but conversely, as a stable repertoire with high fluency for Henry. All the same, on the other hand, fluency and behavioral skill present certain consistencies:—changes of repertoire can be concomitant to jerk increase (Eve and Henry),—a climber that does not benefit from practicing (Gerard) do not acquire new behavioral skills and their fluency remains plateaued.

Each time a climber explores a new cluster (i.e., a pattern which does not exist in the initial repertoire) we can expect higher Jerk. Conversely, each time a climber exploits an existing cluster of his/her repertoire we can expect a decrease of Jerk, because the behavioral pattern is reinforced through practice.

Thereby, this preliminary study provided general support that a full body analysis through clustering (objective A) enhances the understanding of the general dynamics of the learning (objective B), with sufficient sensitivity to individual specificity (objective C) and, furthermore, qualifying the relation between fluency and behavioral skills (main question). However, one skill label per climb, as attributed for the first dataset, is not enough to determine a direct link with performance. These questions were thus followed up in the second data set.

6 Extended per segment experiment

For the second experiment, a more local analysis is undertaken. This time the unit to be clustered is the segment itself. Moreover, contrary to the first data set, *Hold Exploration* and *Hold Change* body states are separated.

6.1 Data set

In the data set, we only keep trials where all the five sensors have been correctly recorded. Finally, 287 trials are scrutinized belonging to 100 sessions from 8 participants. These trials are divided in a total of 15,412 segments. By segment, the features are composed of the rotation mean and the rotation variance of each of the 5 sensors. Thereby a segment is described by 50 real values:—5 rotation mean matrices in $\mathbb{R}^{3 \times 3}$,—5 rotation variances in \mathbb{R} . In the end, the whole data set is composed of 15,412 instances of 50 features each.

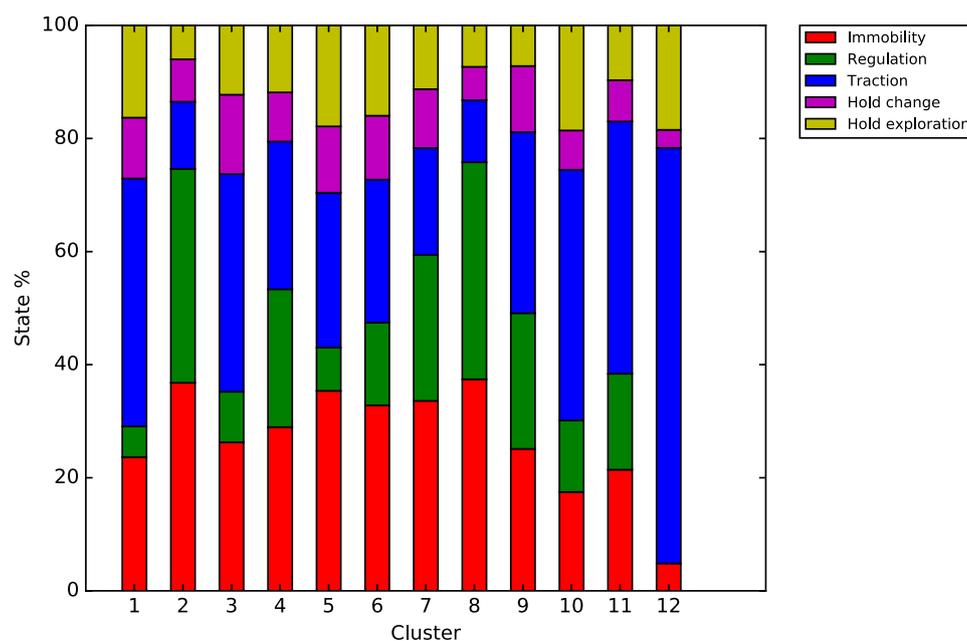


Fig. 11 Cluster per state (% along one cluster)

Notice that the first data set has not been included in this second study. Mixing the sets together bears the risk of biased clustering, since the recording and performing conditions were not the same.

6.2 Embedding and clustering

In clustering, all segments from all climbs from all sessions are pooled together, whoever the climber is. Features of each segment are composed of its LD similarities to other segments, provided directly by Ms.JSE. Therefore, a Gaussian mixture model was preferred over HAC, which can hardly handle so many instances, and over k-means, which can are not easily able to handle clusters with different spreads. The number of clusters is determined through a BIC whose curve gives an optimal number of 12. For the visualization and analysis, an instance is assigned to the cluster with the highest membership.

The repartition of the 12 clusters versus the 5 body states is reviewed in Fig. 11. Moreover, the count session per session of clusters and body states is detailed in Figs. 12 and 13, showing the evolution of the practice over these categories. All figures are computed on all the segments whoever the climber is. In Fig. 14, we can have a look at the general body position for clusters 1, 2, 8 and 12. More exactly at the sensor orientations. On these plots, a sensor is represented by an arrow. The tail of the arrow does not move, it is fixed according to the nature of the sensor:—the right hand, draw in red, is at the north east position,—the left hand (in magenta) at the north west,—the right foot (in blue) at the south east,—the left foot (in cyan) at the south west,—finally, the hip (in black) stares at the center. The head of the arrow is the moving part, it aims at the same direction the corresponding sensor points at.

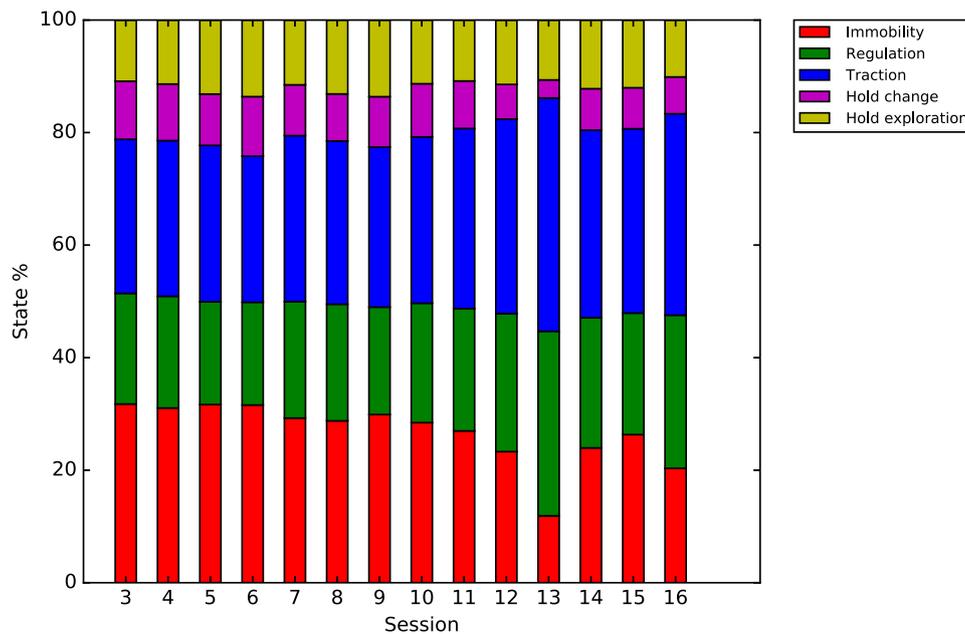


Fig. 12 State per session (% along one session)

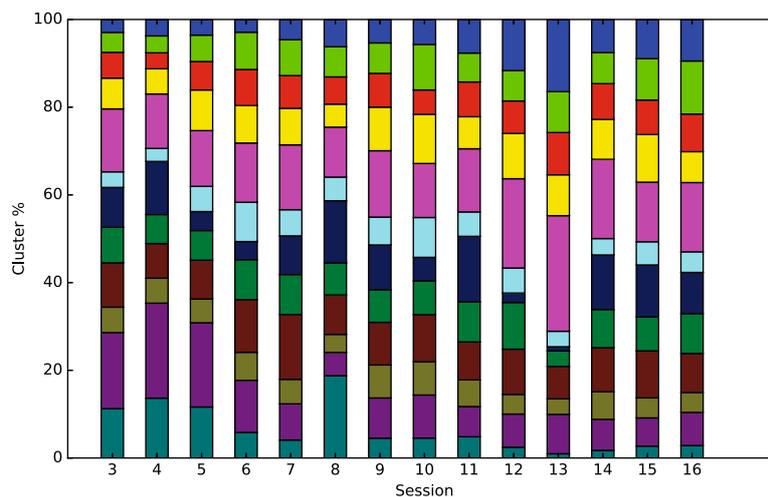


Fig. 13 Cluster per session (% along one session)

The cluster time-line for the 4 climbers is exhibited on “Appendix A”. Each of the figures consist in three plots, their abscissa are aligned and represent the segment rank over time. Dotted black vertical bars are positioned between two climb trials. A red bar dissociates two sessions. As indicated above, one or more trials are deleted out because of recording hazard, this leads to sessions with less than 3 trials. The top plot shows global smoothness/fluency indicators with the jerk (on log scale), the entropy and the immobility ratio (on linear scale). The stair appearance of these curves come from the fact that there is only one indicator set per climb (i.e., since these are computed on the entire trajectory). The middle plot gives the state of each segment. Finally, the bottom plot indicates in which cluster each segment was attributed. To ease the interpretation of the reader, clusters are sorted: clusters that are positioned at the bottom of the plot appear more at the beginning of the protocol; conversely, clusters that occur more at latter sessions are moved to the top of the plot. In all these graphs, the abscissa unit

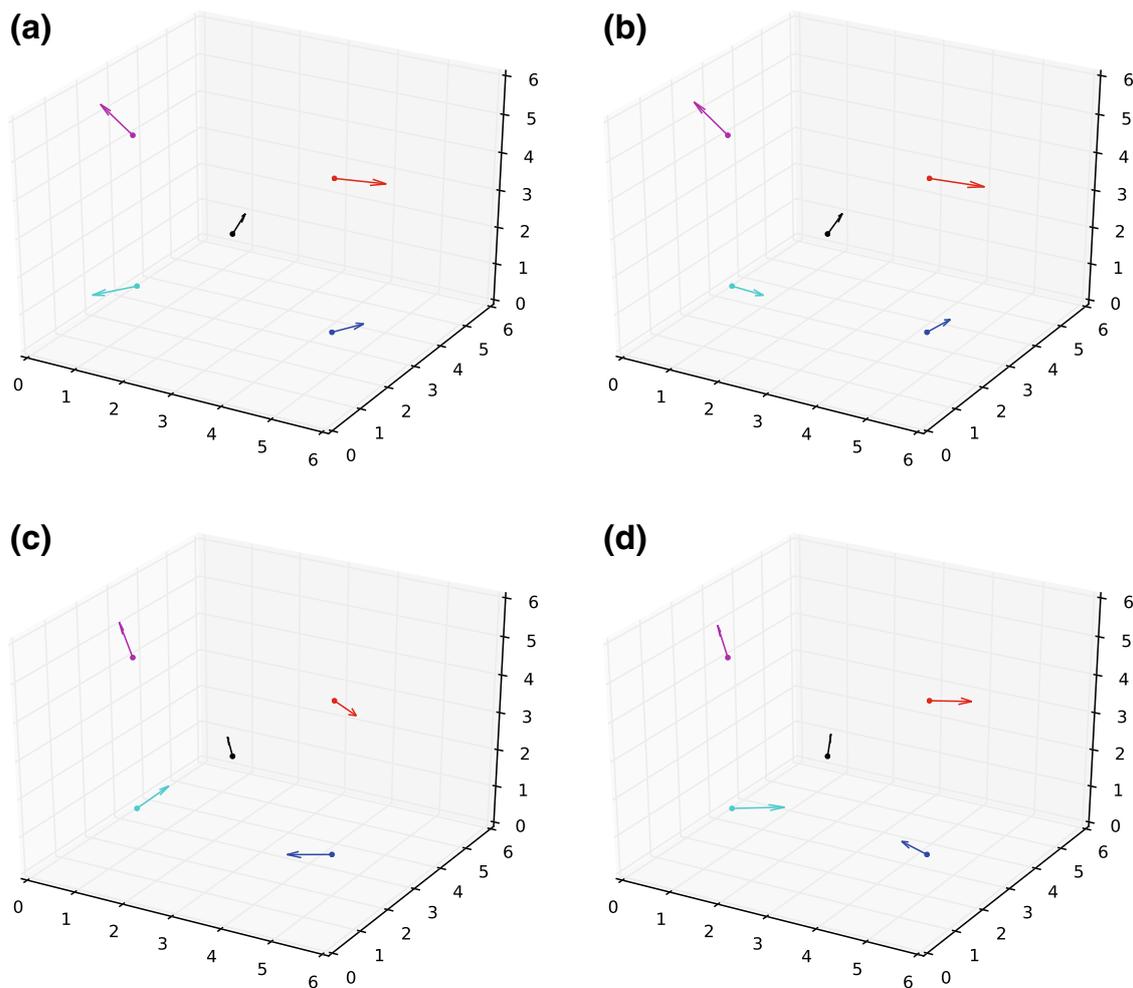


Fig. 14 Sensor orientations for selected clusters. **a** Cluster 1. **b** Cluster 2. **c** Cluster 8. **d** Cluster 12

is the segment, which does not have a fixed duration. For example, early trials look larger; it does not mean they last longer but rather they are more segmented (i.e., earlier trials contain more discrete actions, such as more exploratory reaches, compared to later trials, where exploratory actions are reduced).

The hip position decomposed in segments is displayed in “Appendix B” for 4 typical climbers. The segment limits are indicated by points; higher the number of points, higher the number of activity changes is. The segment color corresponds to the cluster it belongs.

6.3 General dynamics of behavioral skills retrieved by the full-body clustering on a per segment basis

In this sub-section, we will look at the general dynamics of activity states and of the behavioral skills on the second dataset. The analysis is conducted on all the individuals at the same time but using the per segment clustering, recalling that each segment corresponds to a discrete activity state: immobility, postural regulation, hold exploration, hold change or traction. In the following, the notation Cx will be adopted for the cluster number x .

Considering activity states only, three different evolution profiles can be observed in Fig. 12: *Immobility* and *Hold change* decrease over sessions, whereas *Hold exploration* remains stable and *Regulation* and *Traction* get more and more present. The states that appear less effective for performance are *Immobility* and *Hold change*, whereas, the last two states (*Regulation* and *Traction*) appear to improve immediate performance. Thus, the evolution profiles of activity state indicate that climbers generally learn from their practice, wasting less time in non-profitable activities (*Immobility* and *Hold change*) and spending more effort in gainful ones (*Regulation* and *Traction*).

Considering now the evolution of the coordination clusters over the sessions (Fig. 13), the presence of C1 and C2 is weakening along sessions whereas C11 and C12 are raising. Other clusters are stable or with temporary gain, such as C8, or temporary decrease, such as C6. One may yet think about a behavioral transfer between clusters but it has to be specified.

In order to have a clearer view on the coordination dynamics, we look at the repartition of the state activities cluster by cluster in Fig. 11:—Clusters C1, C3, C9, C10, C11, and C12 have a maximum of segments in state *Traction*, with high extremum for C12,—C2, C8 in state *Regulation*, and,—C4, C5, C6, C7 in state *Immobility*. Cluster are ordered by their presence along sessions, thus, practicing go through *Immobility* clusters in medium position (C4–C7), *Traction* is mostly performed through C1 at the beginning and by C12 at the end of the sessions; *Regulation* by C2 at the beginning and by C8 at the end.

What happens there in term of full body representation during these transfers from C1 to C12 and from C2 to C8? In both cases, the new patterns, show the trunk going from an orientation with front of the body facing the wall (clusters 1 and 2; Fig. 14a, b), to more of an oblique orientation (clusters 8 and 12; Fig. 14c, d). Additionally, clusters 8 and 12 differ to each other in so far that the feet are orientated either in a pigeon toed fashion, or where the outer edge of the foot is orientated to be used as support. As a consequence, it would seem that transitions were characterized by the discovery of a movement pattern during traction/regulation where the body was orientated more side-on to the wall.

This analysis on all the individuals presents a general view of the *dynamics of learning* (objective B). During practicing, three phases occur: 1) **Exploitation of a beginner repertoire**, marked by C1 and C2, with the body orientated facing the wall, 2) **Exploration**, marked by C4–C7, 3) **Exploitation of an acquired repertoire**, marked by C8 and C12, with the body orientated more side-on to the wall. The outcome of the full body clustering is to brings us the distinction within exploitation/exploration phases and their interpretation in terms of orientation - such informations cannot be provided by state activities alone.

Indeed we expect that states that do not immediately improve performance can still be functional but at longer time-scales. For example exploration (and perhaps postural regulation and immobility) could lead to a temporary reduction in the rate of performance improvement, corresponding to stable or worsening in Jerk, however, over longer timescales these behaviors might uncover information for action. In doing so, after a period of exploration, the learner may discover more efficient and possibly entirely new coordination patterns. These, they can then exploit and refine to perhaps

dramatically improve performance beyond which would have been possible if they had remained using the same coordination patterns from the beginning of practice.

6.4 Individual consistencies in learning dynamics involving the emergence of skilled climbing behavior

In order to examine individual specificity, we will first focus on 4 climbers (14, 12, 19 and 21) and then extend on the remaining subjects.

Figure 16 exhibits a representation of the evolution of the climbing fluency and the climbing behavior (found clusters and states), along practicing session for **climber 14**. The clusters' evolution, lower part of the figure, can be split into 4 phases: 1) *Exploitation of the beginner repertoire*, from the first session to the third: there are a high number of activity changes (which is given by the width of a trial on the graph), and, each trial finishes by a predominance of C1 and C2, 2) *Exploration A*, From the fourth to the sixth sessions: the graph indicates much less activity segments, and the orientations lie mainly in clusters C8, C10 and C12, 3) *Exploration B*, At the seventh session, the climber returns back to a C1 and C2 limb coordination repertoire while keeping the number of activity changes low (as in the second phase), 4) *Exploitation of the acquired repertoire*, From the eighth until the final trial, again, C8, C10 and C12 are the main clusters. In the meantime, when looking at the fluency in the upper part of the figure, the Immobility ratio is slowly and smoothly decreasing, whereas Jerk and Entropy demonstrate an abrupt change at the beginning of the second phase (recalling that the lower these indicators are, the better the performance is). In Fig. 19, the simplification of the segmentation is clearly noticeable from the first to the last trials. The change in behavioral repertoire, the decrease of the fluency indicators and the decrease of the number of activity segments indicate that the subject is learning. Moreover, this detailed view confirms the general orientation transition from C1,C2 at the beginning to C8,C12 at the end of the practicing, as mentioned in the previous section. Notably, the abrupt change in the jerk corresponds to the transition from the first to the second behavioral phase. More interesting is the fact that when the subject return back to the original repertoire at the third phase the fluency indicator stay low. Thereby, the limb coordination and orientation are not in direct relation to the fluency.

Now, let us look at **climber 12** at Fig. 15. This time the behavioral skills, can be depicted into 3 phases: 1) *Exploitation of the beginner repertoire*, from the first session to the third: high numbers of activity changes, C1, C2 and C8 are predominant, 2) *Exploration*, from fourth session to the seventh session: the number of activity changes is decreasing, where all the clusters are used with a predominance of C8, 3) *Exploitation of the acquired repertoire*, From the ninth until the end, the most present clusters are C8 and C12. Here, the fluency indicators are globally diminishing with an exception of the jerk that is fluctuating at the beginning of the exploration phase. As the previous subject, Fig. 19 demonstrates a clarification of the activity for that climber. The general profile of the limb coordination, the evolution of the activity changes as well as the evolution of the fluency demonstrate that the subject is learning. Nevertheless, contrary to the previous climber, we see an exploration phase that is not clearly structured with respect organized in term of behavioral skills, correlated with a

temporary increase in the jerk. This would suggest a blind search scheme as depicted in Gel'fand and Tsetlin (1962).

Climber 19 has a simpler organization scheme (Fig. 17). All goes smoothly. In the behavioral profile, C1 is slowly decreasing, C12 slowly increasing, C2 and C8 stay present across all sessions. As with previous subjects, the number of activity changes (Fig. 19) and fluency indicator are decreasing, demonstrating the learning curve. It also remains that the climber goes from a facing wall orientation (C1) to a side-on wall orientation (C12); however, it is harder to split the time-line into exploitation/exploration phases.

Some of the subjects do not always improve performance through practice. This is typically the case of **climber 21** (Fig. 18). Fluency indicators stay high, whether they stand for spatial (Entropy), temporal (Immo ratio) or spatial-temporal (Jerk) measurements. Moreover, no noticeable cluster dynamics is shown from one session to the other (with exception of session 2) and the number of activity changes remains high (Fig. 19).

The other climbers present in the studies (subjects 13, 15, 17 and 18) display similar profile than one of the fourth aforementioned profiles. Their climbing time-line can be obtained upon request to authors.

Thereby, the individual study of coordination cluster time-line enables us to qualify *subject specificity* in the practicing dynamics (objective C):—which climbers perform a quick and efficient exploration phase (ongoing improvement),—which ones are more disoriented in their re-organization during learning phase, eventually leading to a more operative coordination (sudden improvement),—which subjects do not receive any clear advantage of practicing (no improvement).

Our *main question* was: Do climbers improve simultaneously, linearly and in a proportional way both their climbing fluency and their behavioral skills? Climbers 12, 13, 14, and 17 confirms our hypothesis and show that, no, it does not:—Exploration and transition phases, even if smooth in term of behavioral skills may induce a bump in the jerk and so a decrease of fluency (subjects 12 and 13).—After practicing, a temporary return to original beginner repertoire may not weaken the fluency (climber 14).

7 Conclusion and perspectives

Cluster analysis appeared as a promising way to investigate the dynamics of climbing practice in order to highlight the individual pathway of learning. Indeed, it outmatches past studies based on oscillator models by taking into account the full body dynamics and not only a sub-set of limbs (namely, upper limbs) (objective A). In particular, clustering of discrete activity states enables us to discover learning dynamics and changes in orientation and dynamics of lower limbs and trunk along the time-scale of ongoing practice, and, specifically, changes that coincide with more fluent traction (objective B). Interpretation of each climber cluster time-line highlights individual specificity (objective C) such as a lack of acquisition during practicing, blind search and exploration followed by temporary return to original repertoire. Moreover, we can answer our *main question* about the link between the behavioral skill and fluency: the coordi-

nation time-line (obtained through clustering) is clearly not adequately described as linear and proportional to the climbing fluency.

Future research would help to emphasize how the different learning dynamics (i.e., abrupt transition, gradual transition, and no transition) are linked to the activity states (i.e., immobility, traction, hold exploration, hold transition, postural regulation). For instance a particular question is whether we can predict which pattern of change the individual is predisposed to by using information of their early behavioral dynamics. Also related to this concern is the need to identify more precisely points during performance where new skills emerge (higher climbing fluency and change in behavior). Is it, for example, during states of immobility that climbers can discover more effective traction states, or do they emerge as a refinement to previous traction states? Another possibility is that visual-motor coordination is a key relationship to address. Examining how the climber coordinates where and when they look at holds dedicated for a particular state will enable us to qualify more precisely phases of mobility. For example, is the climber projecting himself upward on the climb by looking at upper grasps, planning next moves? or is she/he concentrating on the current position?

Of additional interest is, that, given the tendency for each cluster to be visited within each climb and the apparent regularity in terms of cluster sequencing from one trial to the next, it is unclear how the individual's location on the route is influencing the regularity of within trial cluster dynamics. It is possible for example that the emergence of more effective traction states were not so much a global response to the route, but, emerged from experience with specific holds and their local configuration.

In order to address these questions it would be useful:—to localize clustering patterns with respect to hold locations and consider whether patterns can be identified with respect to the sequencing of clusters;—to consider the fuzzy ownership of clusters given by the Gaussian mixture;—to analyze the variance of limb position;—to use data mining methods or Markov models on cluster time-line to discover recurrent patterns in behavioral dynamics.

Acknowledgements This work was partially funded by the Agence Nationale de la Recherche with the ANR-13-JSH2-004 Dynamov Project. and Belgian F.R.S.-FNRS (National Fund of Scientific Research).

Appendix A: Cluster and fluency time-lines for the second data set

See Figs. 15, 16, 17 and 18.

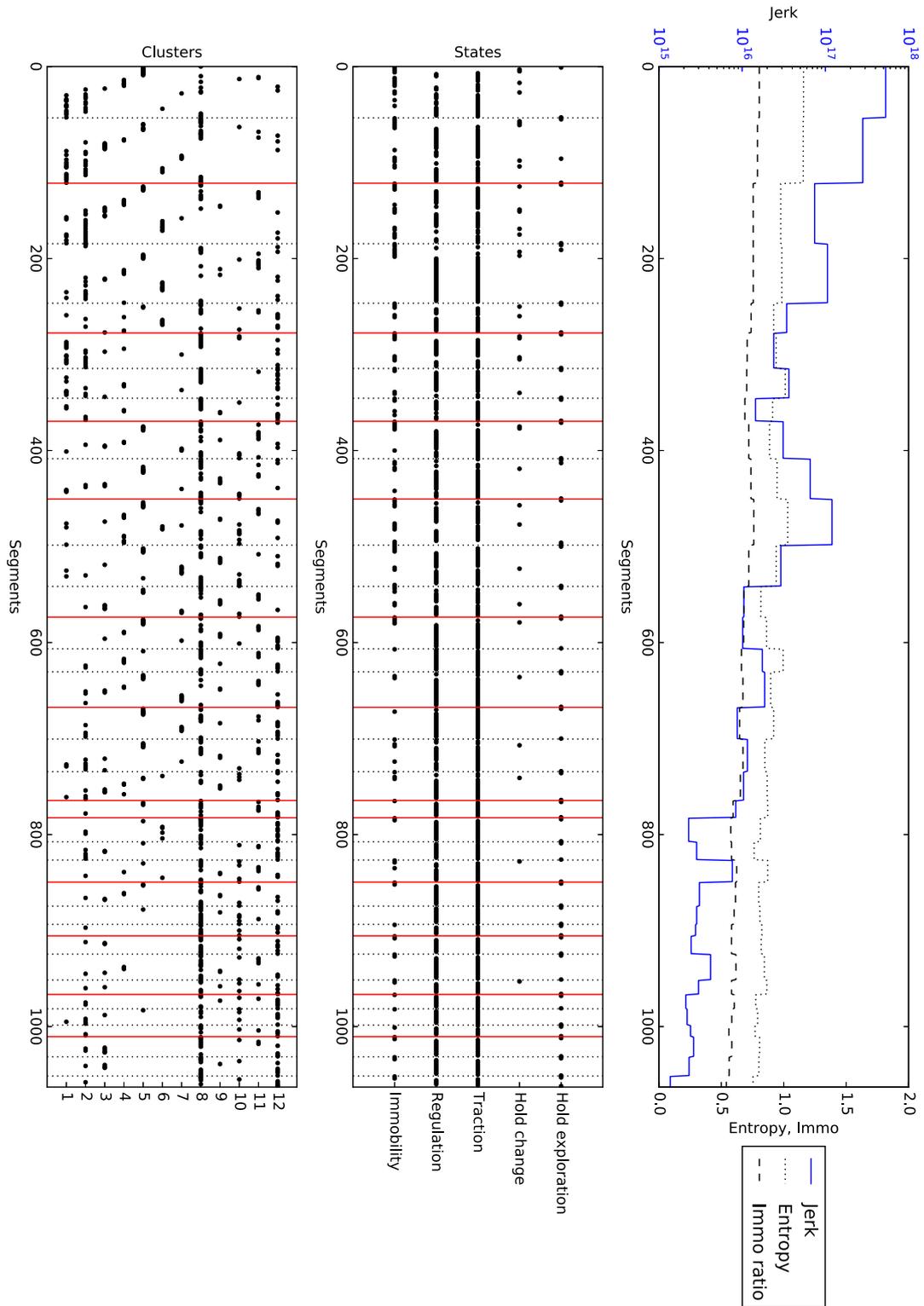


Fig. 15 Climber id #12

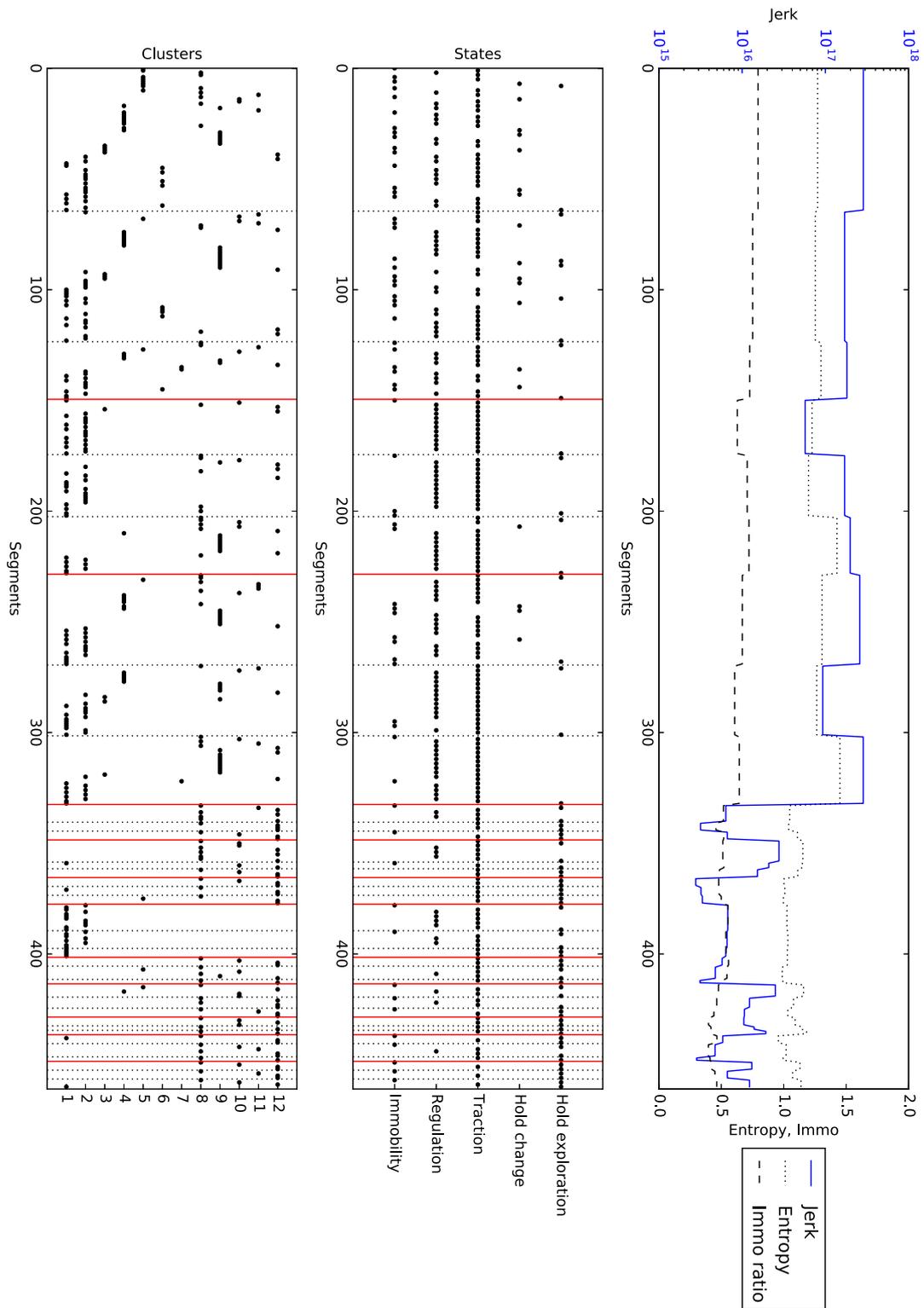


Fig. 16 Climber id #14

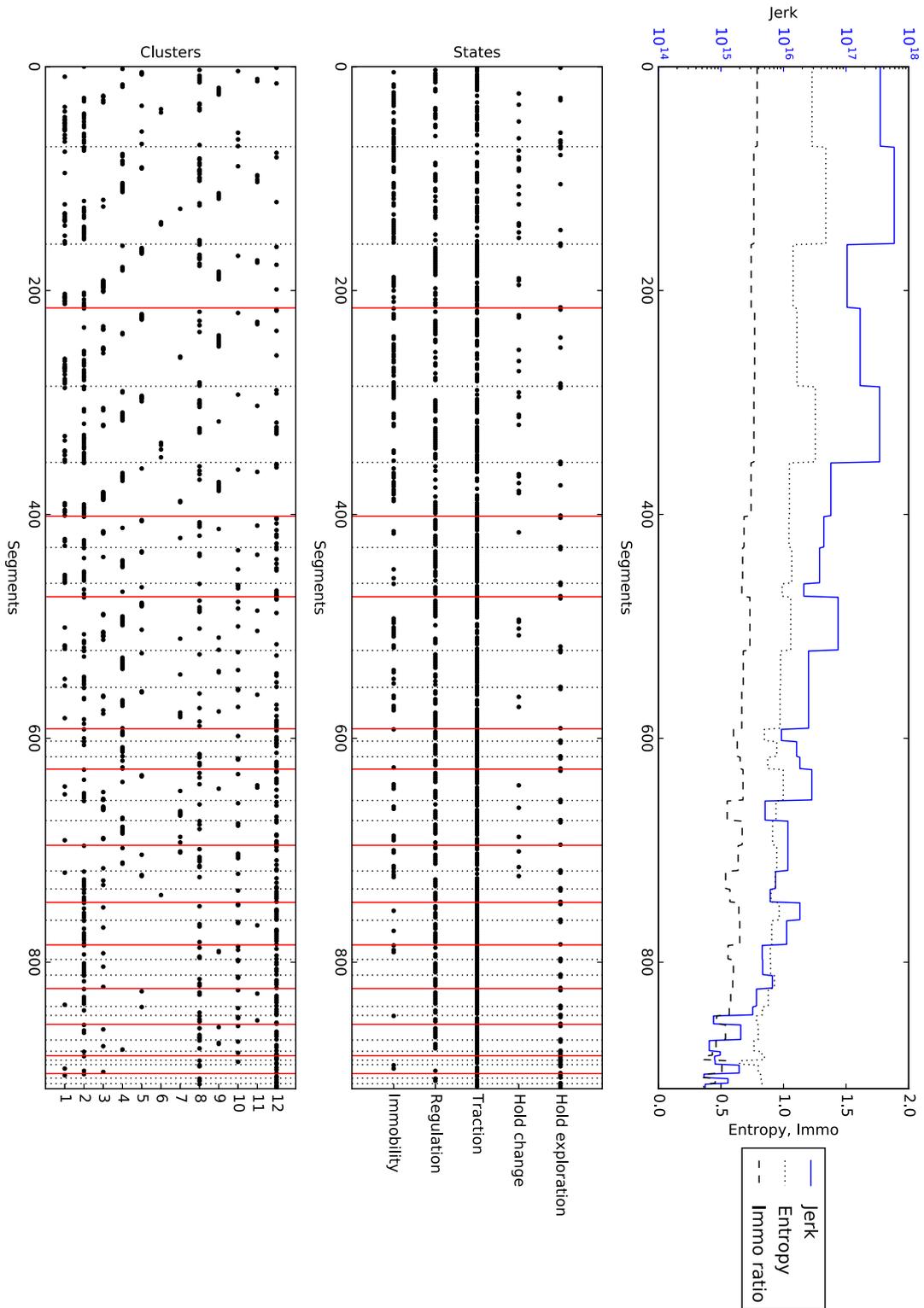


Fig. 17 Climber id #19

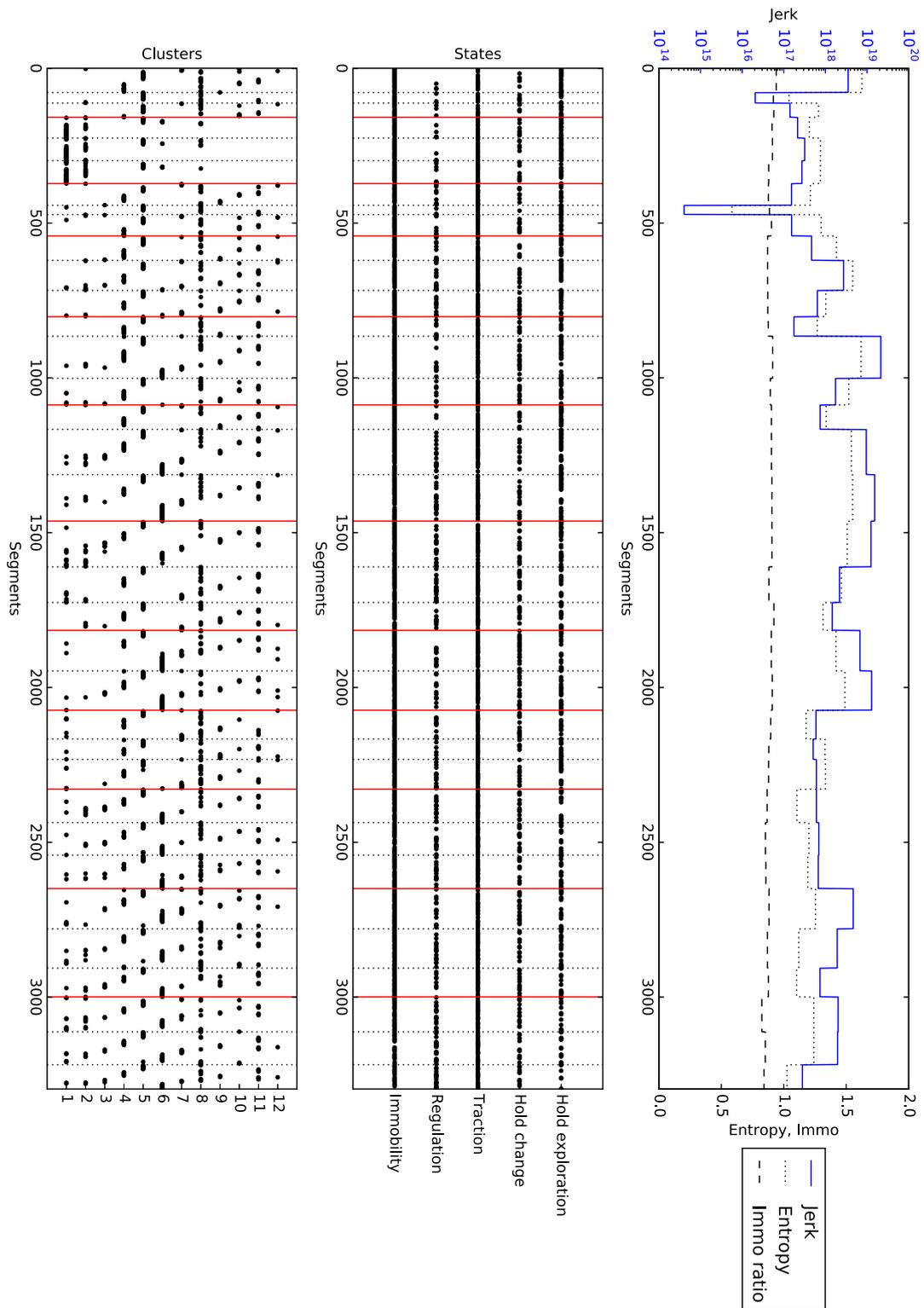


Fig. 18 Climber id #21

Appendix B: Hip position linked to segment cluster for the second data set

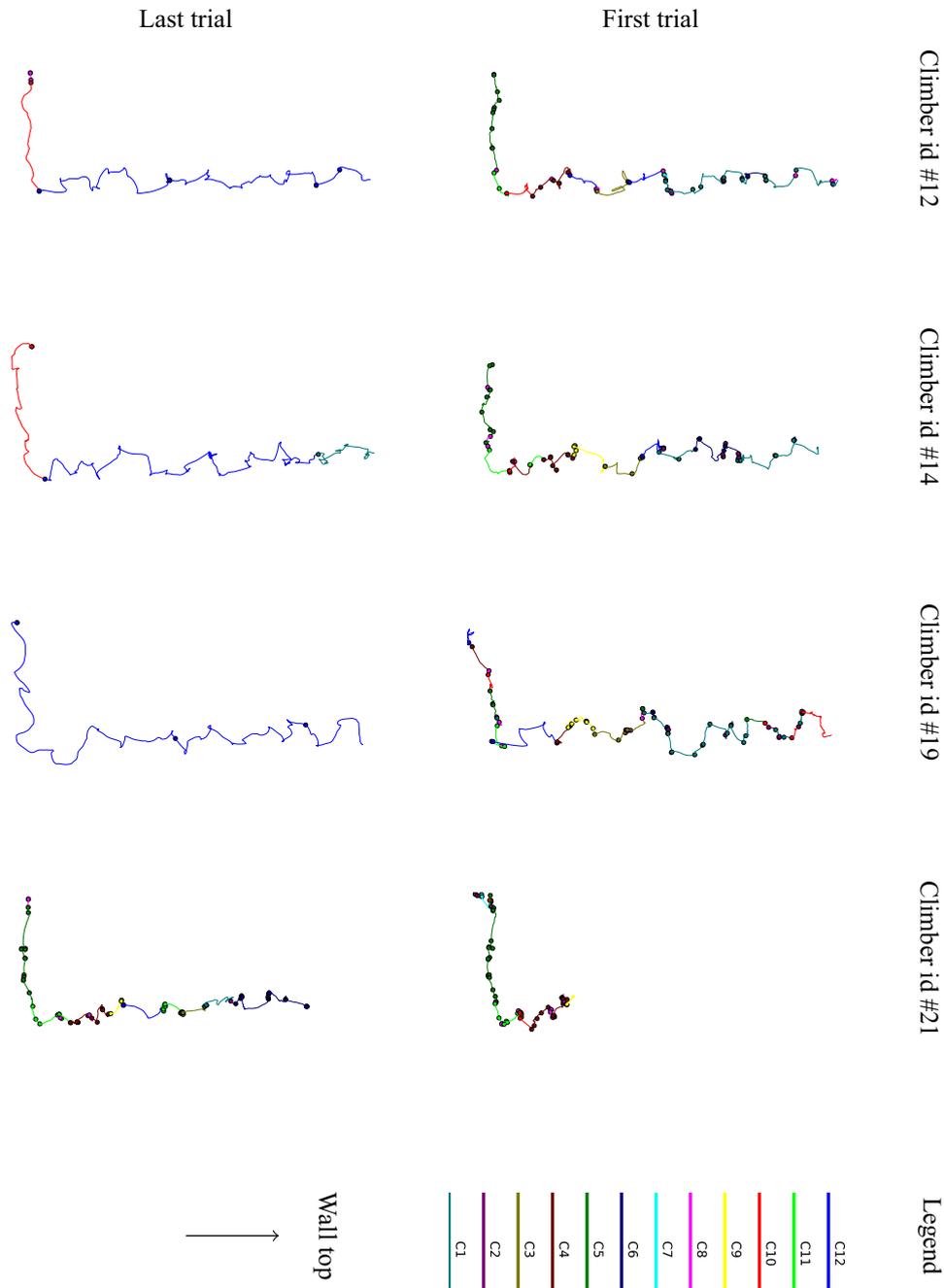


Fig. 19 Hip position linked to segment cluster for the second data set

References

Bardy B, Oullier O, Bootsma RJ, Stoffregen TA (2002) Dynamics of human postural transitions. *J Exp Psychol Hum Percept Perform* 28(3):499

- Basseville M, Nikiforov IV et al (1993) Detection of abrupt changes: theory and application, vol 104. Prentice Hall, Englewood Cliffs
- Bernstein NA, Latash ML, Turvey M (1996) Dexterity and its development. Taylor & Francis, New York
- Boulanger J, Seifert L, Héroult R, Coeurjolly JF (2016) Automatic sensor-based detection and classification of climbing activities. *IEEE Sens J* 16(3):742–749. doi:[10.1109/JSEN.2015.2481511](https://doi.org/10.1109/JSEN.2015.2481511)
- Chow JY, Davids K, Button C, Koh M (2008) Coordination changes in a discrete multi-articular action as a function of practice. *Acta Psychol* 127(1):163–176
- Cordier P, France MM, Bolon P, Pailhous J (1993) Entropy, degrees of freedom, and free climbing: a thermodynamic study of a complex behavior based on trajectory analysis. *Int J Sport Psychol* 24(4):370–378
- Davids K, Button C, Araújo D, Renshaw I, Hristovski R (2006) Movement models from sports provide representative task constraints for studying adaptive behavior in human movement systems. *Adapt Behav* 14(1):73–95
- Demartines P, Héroult J (1997) Curvilinear component analysis: a self-organizing neural network for non-linear mapping of data sets. *IEEE Trans Neural Netw* 8(1):148–154
- Engø K (2001) On the BCH-formula in $so(3)$. *BIT Numer Math* 41(3):629–632
- Gel'fand IM, Tsetlin M (1962) Some methods of control for complex systems. *Russ Math Sur* 17(1):95
- Hall B (2015) Lie groups, Lie algebras, and representations: an elementary introduction. Springer, Berlin
- Hinton GE, Roweis ST (2002) Stochastic neighbor embedding. In: *Advances in neural information processing systems*, pp 833–840
- Kelso J (1984) Phase transitions and critical behavior in human bimanual coordination. *Am J Physiol Regul Integr Comp Physiol* 246(6):R1000–R1004
- Kostrubiec V, Zanone PG, Fuchs A, Kelso JS (2012) Beyond the blank slate: routes to learning new coordination patterns depend on the intrinsic dynamics of the learner—experimental evidence and theoretical model. *Front Hum Neurosci* 6:1–14
- Kruskal J (1964) Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika* 29:1–28
- Lee J, Verleysen M (2014) Two key properties of dimensionality reduction methods. In: *IEEE SSCI 2014—2014 IEEE symposium series on computational intelligence—CIDM 2014: 2014 IEEE symposium on computational intelligence and data mining*, pp 163–170
- Lee JA, Renard E, Bernard G, Dupont P, Verleysen M (2013) Type 1 and 2 mixtures of kullback-leibler divergences as cost functions in dimensionality reduction based on similarity preservation. *Neurocomputing* 112:92–108
- Lee JA, Peluffo-Ordóñez DH, Verleysen M (2015) Multi-scale similarities in stochastic neighbour embedding: reducing dimensionality while preserving both local and global structure. *Neurocomputing* 169:246–261
- Madgwick S (2010) An efficient orientation filter for inertial and inertial/magnetic sensor arrays. Report x-io and University of Bristol, UK
- Madgwick S, Harrison A, Vaidyanathan R (2011) Estimation of imu and mag orientation using a gradient descent algorithm. In: *2011 IEEE international conference on rehabilitation robotics (ICORR)*, pp 1–7
- Manton JH (2004) A globally convergent numerical algorithm for computing the centre of mass on compact lie groups. In: *Control, automation, robotics and vision conference, 2004. ICARCV 2004 8th, IEEE*, vol 3, pp 2211–2216
- Nourrit D, Delignières D, Caillou N, Deschamps T, Lauriot B (2003) On discontinuities in motor learning: a longitudinal study of complex skill acquisition on a ski-simulator. *J Mot Behav* 35(2):151–170
- Orth D, Davids K, Seifert L (2016) Coordination in climbing: effect of skill, practice and constraints manipulation. *Sports Med* 46(2):255–268
- Pijpers J, Oudejans RR, Bakker FC, Beek PJ (2006) The role of anxiety in perceiving and realizing affordances. *Ecol Psychol* 18(3):131–161
- Sammon J (1969) A nonlinear mapping algorithm for data structure analysis. *IEEE Trans Comput* 18(5):401–409
- Seifert L, Coeurjolly JF, Héroult R, Wattedled L, Davids K (2013) Temporal dynamics of inter-limb coordination in ice climbing revealed through change-point analysis of the geodesic mean of circular data. *J Appl Stat* 40(11):2317–2331
- Seifert L, L'Hermette M, Komar J, Orth D, Mell F, Merriault P, Grenet P, Caritu Y, Héroult R, Dovgalecs V, Davids K (2014a) Pattern recognition in cyclic and discrete skills performance from inertial measurement units. *Procedia Eng* 72:196–201 (**the Engineering of Sport 10**)

- Seifert L, Orth D, Boulanger J, Dovgalecs V, Herault R, Davids K (2014b) Climbing skill and complexity of climbing wall design: assessment of jerk as a novel indicator of performance fluency. *J Appl Biomech* 30(5):619–625
- Seifert L, Wattedled L, Herault R, Poizat G, Adé D, Gal-Petitfaux N, Davids K (2014c) Neurobiological degeneracy and affordance perception support functional intra-individual variability of inter-limb coordination during ice climbing. *PLoS ONE* 9(2):e89865
- Shepard R (1962) The analysis of proximities: multidimensional scaling with an unknown distance function (parts 1 and 2). *Psychometrika* 27(125–140):219–249
- Teulier C, Delignieres D (2007) The nature of the transition between novice and skilled coordination during learning to swing. *Hum Mov Sci* 26(3):376–392
- Torgerson W (1952) Multidimensional scaling, I: theory and method. *Psychometrika* 17:401–419
- van der Maaten L, Hinton G (2008) Visualizing data using t-SNE. *J Mach Learn Res* 9:2579–2605
- World Medical Association (2013) World Medical Association Declaration of Helsinki: ethical principles for medical research involving human subjects. *JAMA* 310(20):2191

A.7 Key point selection and clustering of swimmer coordination through Sparse Fisher-EM

Reference

[KHS14] John Komar, Romain Hérault, and Ludovic Seifert. “Key Point Selection and Clustering of Swimmer Coordination through Sparse Fisher-EM.” in: ECML/PKDD 2013 Workshop on Machine Learning and Data Mining for Sports Analytics (MLSA2013). Jan. 7, 2014. arXiv: 1401.1489 [physics, stat]. URL: <http://arxiv.org/abs/1401.1489>

Key point selection and clustering of swimmer coordination through Sparse Fisher-EM

John Komar¹, Romain Herault², and Ludovic Seifert¹

¹ CETAPS EA-3832 Université de Rouen,
Boulevard Siegfried, 76821 Mont Saint Aignan, France

`firstname.lastname@univ-rouen.fr`

² LITIS EA-4108, INSA de Rouen,
Avenue de l'Université - BP 8,
76801 Saint-Étienne-du-Rouvray Cedex, France

`firstname.lastname@insa-rouen.fr` *

Abstract. To answer the existence of optimal swimmer learning/teaching strategies, this work introduces a two-level clustering in order to analyze temporal dynamics of motor learning in breaststroke swimming. Each level have been performed through Sparse Fisher-EM, a unsupervised framework which can be applied efficiently on large and correlated datasets. The induced sparsity selects key points of the coordination phase without any prior knowledge.

Keywords: Clustering, Variable selection, Temporal dynamics of motor learning, Sparse Fisher-EM

1 Introduction

The development of Dynamical Systems Theory [1] in understanding motor learning has increased the interest of sports scientists in focusing on temporal dynamics of human motor behavior. Broadly speaking, the investigation of motor learning traditionally implied the assessment of both a pre-learning behavior and a post-learning behavior [2], but the deep understanding of the process of motor learning requires a continuous and long term assessment of the behavior rather than previous traditional discrete assessments. Indeed, such a continuous assessment of behavioral data enables to investigate the nature of the learning process and might highlight the paramount role played by motor variability in optimizing learning [2].

From a theoretical point of view, motor learning is viewed as a process involving active exploration of a so-called perceptual-motor workspace which is learner dependent and defines all the motor possibilities available to him. Few studies have already highlighted this exploratory behavior during learning a ski

* Authors would like to thank the *Agence Nationale de la Recherche* (ANR) for its financial support to§ the project LeMOn (Learning with Multi-objective Optimization, ANR-11-JS02-10).

simulator task [3] or a soccer kicking task [4]. These authors showed that learners exhibited different qualitative motor organizations during skill acquisition. Nevertheless, these principles studies mainly focused on a static analysis, defining the different behaviors exhibited during learning. As a matter of fact, a major interest in the field of motor learning resides in the definition of different pathways of learning, namely different possible learning strategies [5]. Such an interest in investigating the existence of different "routes of learning" needs to focus on a dynamical analysis, namely the analysis of the successions of different behaviors. An unanswered question to date concerns the existence of optimal learning strategies (i.e. strategies that would appear more effective). Thus, the discovery of optimal learning strategies could have a huge impact on the pedagogical approach of practitioners.

The article will describe at first the context of the research insisting on the way data have been collected, what are the long-term expectations in sport science field and what are the short term locks in machine learning field. Then we will give a brief view of the Fisher-EM algorithm [6] which is an unsupervised learning method used in this work. In the end, preliminary results of the data clustering will be analyzed.

2 Context of the Research

2.1 Previous work

In breaststroke swimming, achieving high performance requires a particular management of both arm and leg movements, in order to maximize propulsive effectiveness and optimize the glide and recovery times [7]. Therefore, expertise in breaststroke is defined by adopting a precise coordination pattern between arms and legs (i.e. a specific spatial and temporal relationship between elbow and knee oscillations). Indeed, when knees are flexing, elbows should be fully extended (180°), whereas knees should be fully extended (180°) when elbows are flexing, in order to ensure a hydrodynamic position of the non-propulsive limbs when the first pair of limbs is actually propulsive [8,9].

Based on this context, the breaststroke swimming task was deemed as suitable in investigating the dynamics of learning, mainly as it implies at a macroscopic scale the acquisition of an expert arm-leg coordination that can be easily assessed. However, the investigation of potential differences in learning strategies required a continuous movement assessment. In that sense, the use of motion sensors allowed a fast, accurate and cycle per cycle movement assessment.

Previously, two analysis methods were used in the cycle per cycle study of motor learning. A previous study [3] highlighted the unstable character of the transition between novice and expert, but not really an exploration as experimental setup assumes that novices left their initial behavior to adopt the expert one. Therefore, no search strategies were really investigated. In order to overcome this issue, [4] used a cluster analysis (Hierarchical Cluster Analysis) in their experiment on football kicking and highlighted different behaviors used

by each participant during learning to kick a ball. The authors therefore linked these different behaviors to a search strategy. However, the cluster analysis was performed individually and there was no comparison done between the learners (e.g. did they use identical behaviors?), it implied only few participants (i.e. four learners), it was performed only with 120 kicks per learner (i.e. 10 kicks per session during 12 sessions) and like the previous study of [3] it only defined the behavior from a static point of view (i.e. defining what behavior was adopted).

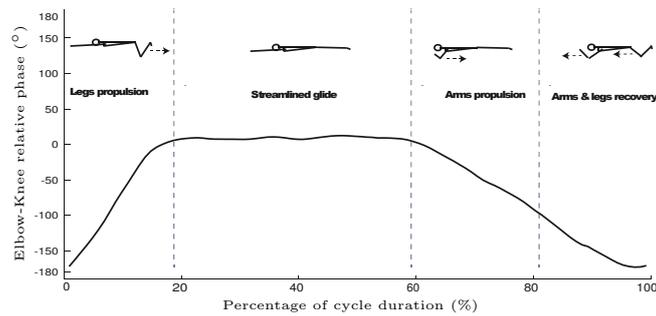


Fig. 1: A typical continuous relative phase between the knee and the elbow

2.2 Data collection

For this study, 26 novices were involved in 16 lessons of breaststroke swimming, with two sessions per week for a total duration of two months. The general goal of learning for all the 26 swimmers was to increase the distance per stroke, while maintaining the speed stable. Then the 26 learners were divided into four different groups, each group receiving a different instruction during the learning process:

- 1) Control group (N=7): This group received only the general goal of learning, increase the distance per stroke
- 2) Analogy group (N=7): In addition to the general goal of learning, this group received a single additional instruction: "glide two seconds with your arms outstretched"
- 3) Pacer group (N=6): In addition to the general goal of learning, this group had to follow an auditory metronome trying to perform one cycle every single auditory signal. The frequency of the metronome was decreased every two sessions, in order to promote a decrease in the stroke frequency of the learners that should lead to an increase in the distance per stroke
- 4) Prescription group (N=6): In addition to the general goal of learning, this group received multiple additional instructions: "keep your arms outstretched forward when you extend your legs; then glide with your arms and legs outstretched; then keep your legs outstretched when you flex your arms; recover both arms and legs together". These different instructions were supposed to have a specific impact on the learning strategies of the learners.

Each learner performed 10 trials of 25-m swim during each session, with 1 x 25-m consisting approximatively in 8 recorded cycles (one cycle correspond to the period between two successive maximal knee flexion). During every learning session, all learners were equipped with small motion sensors on both arms and legs (3-D gyroscopes, 3-D magnetometers, 3-D accelerometers) including a data logger and recording elbow and knee angles at a frequency of 200 Hz.

Following the literature in coordination dynamics [1], the coordination between elbow and knee was defined by the continuous relative phase between these two oscillators [10], considering elbows and knees as acting like individual pendulums [7]. A value of relative phase close to -180° or 180° defined an anti-phase relationship (i.e. opposite movements of knee and elbow) while a value close to 0° defined an in-phase mode of coordination (i.e. identical movements of knee and elbow); here, each cycle will be described by a time series of 100 normalized values of continuous relative phase between the knee and the elbow (Fig. 1).

To sum-up, we have recorded 4160 trials ($26 \text{ swimmers} \times 16 \text{ sessions} \times 10 \text{ trials}$) and there is an average of 8 cycles per trials. Thus, the dataset is composed by 33280 cycles, each cycle is represented by 100 continuous relative phase samples.

2.3 Study expectations

From a sport sciences point of view, the specific aims of the study were twofold: – Assessing the dynamics of learning: In other words, the aim was to assess not only the different behaviors used during learning but also the transitions between these behaviors, that is the potential search strategy exhibited by learners (e.g. they used preferably behavior n° 1 then n° 4, then n° 3 ...). – Assessing the impact of different learning conditions on the dynamics of learning: In other words, the aim was to investigate the possible existence of different behaviors exhibited by the learners regarding their learning condition, as well as the possible existence of different search strategy exhibited by the different groups.

A last point in this experiment was the possibility to transfer the results of the analysis towards practical application or guidelines for teachers. From a pedagogical point of view, it appeared difficult to teach novice swimmers by giving instruction on the arm-leg coordination during all the cycle and the definition of key points within the entire cycle reflects a paramount aspect for teaching. Indeed, a strong literature in sports pedagogy highlights the role played by attentional focalization during motor learning, as a focalization on a key point of the swimming cycle may be highly beneficial in seeking to reorganize the entire arm-leg coordination [11]. A third aim of this study was then to define highly discriminative key points within the swimming cycle and that might be the target of the instruction in order to orient the attention of learners.

From a machine learning point of view, there are two locks to tackle: 1) Each cycle is described by 100 features which are highly correlated due to the fact that they are samples of the relative phase which is a continuous time signal.

Nevertheless, we don't want to bias the study by preprocessing the data, a transformation like filters, wavelet transform or sample selection that will embed our a priori knowledge. 2) The number of cycles are not equal on all the trials, that is why a trial can not be directly described by a fixed number of features.

Those two problems were address by 1) using a clustering by Fisher-EM [6] that also performs dimension reduction and features selection, 2) doing a two stage clustering: on cycles then on trials; a procedure similar to *Bags of words* to have fixed size features on trial.

3 Fisher-EM Algorithm

A clustering can be derived from a mixture of Gaussians generative model. A Gaussian, which is parameterized by a covariance matrix and a mean in the observation space, represents a cluster. An observation is labeled according to its ownership (likelihood ratio) to each Gaussian. Knowing the number of clusters, the mixture and Gaussian parameters are learned from the observation data through an Expectation-Maximization (EM) algorithm.

The Fisher-EM algorithm [6] is based on the same principles but the mixture of Gaussians does not lie directly on the observation space but on a lower dimension latent space. This latent space is chosen to maximize the Fisher criterion between clusters and thus be discriminative and its dimension is bounded by the number of clusters. This reduction of dimension leads to more efficient computation on medium to large datasets (here 33280 examples by 100 features) as operations can be held in the smaller latent space.

3.1 Generative Model

We consider that the n observations y_1, y_2, \dots, y_n are realizations of a random vector $Y \in \mathbb{R}^p$. We want to cluster these observations into K groups. For each observation y_i , a variable $z_i \in Z = \{1, \dots, K\}$ indicates which cluster its belong to. This clustering will be decided upon a generative model, namely a mixture of K Gaussians which lies in a discriminative latent space $X \in \mathbb{R}^d$ where $d \leq K - 1$.

This latent space is linked to the observation space through a linear transformation,

$$Y = UX + \epsilon, \quad (1)$$

where $U \in \mathbb{R}^{p \times d}$ and $U^t U = Id(d)$ where $Id(d)$ is the identity matrix of size d , i.e. U is an orthogonal matrix and ϵ non-discriminative noise.

Let be $W = [U, V] \in \mathbb{R}^{p \times p}$ such that $W^t W = Id(p)$. V is the orthogonal complement of U . Thus, a projection $U^t y$ of an observation y from space Y of dimension p , lies on the latent discriminative subspace X of dimension d and the projection $V^t y_i$ lies on the non-discriminative complement subspace of dimension $p - d$.

Conditionally to $Z = k$, random variables X and Y are assumed to be Gaussian, $X|_{Z=k} \sim \mathcal{N}(\mu_k, \Sigma_k)$, and $Y|_{Z=k} \sim \mathcal{N}(m_k, S_k)$, where $\mu_k \in \mathbb{R}^d$, $\Sigma_k \in \mathbb{R}^{d \times d}$, $m_k \in \mathbb{R}^p$ and $S_k \in \mathbb{R}^{p \times p}$.

With the help of equation 1, we can deduce parameters of the distribution $Y_{|Z=k}$ in the observation space from the parameters of the distribution $X_{|Z=k}$ in the latent space, $m_k = U\mu_k$ and $S_k = U\Sigma_k U^t + \Psi$, where $\Psi \in \mathbb{R}^{p \times p}$ is the covariance matrix of ϵ which is assumed to follow a 0-centered Gaussian distribution. To ensure that ϵ represents non-discriminative noise, we will impose that the covariance of ϵ , Ψ , projected into the discriminative space is null, i.e. $U\Psi U^t = \mathbf{0}(d)$, and that Ψ projected into the non-discriminative subspace is diagonal, i.e. $V\Psi V^t = \beta Id(p-d)$. Thus,

$$W^t S_k W = \begin{pmatrix} \Sigma_k & \mathbf{0} \\ \mathbf{0} & \beta_k Id(p-d) \end{pmatrix}. \quad (2)$$

All the Gaussian distributions are mixed together, the density of the generative model is given by $f(y) = \sum_{k=1}^K \pi_k \phi(y; m_k, S_k)$ where π_k are mixing proportion and m_k, S_k are deduced from $\{U, \beta, \mu_k, \Sigma_k\}$.

Finally, the model is parameterized by: $- U$ the projection from discriminative subspace to observation space, $-\beta_k$ variance of ϵ in the non-discriminative subspace, $-\pi_k$ the mixing parameter, $-\mu_k$ and Gaussian parameter $\{\mu_k, \Sigma_k\}$, where the 3 last parameters are repeated by the number of Gaussians.

Model variations, that lead to reduced numbers of parameters, can be achieved by enforcing shared covariances β and/or Σ between Gaussians, diagonalization of the covariance Σ without or with constant diagonal, and combination of these enforcements.

3.2 Parameter estimation

The iterative Expectation-Maximization (EM) algorithm can be extended by a Fisher Step (*F-Step*) in-between the *E-Step* and the *M-Step* where the latent discriminative subspace is computed [6]. The Fisher criterion computed at the *F-Step* is used as a stopping criterion. Convergences properties can be found in [12].

E-Step In this step, for each observation i , its posterior probability to each cluster k is computed by

$$o_{ik} \leftarrow \frac{\pi_k \phi(y_i, \hat{\theta}_k)}{\sum_{l=1}^K \pi_l \phi(y_i, \hat{\theta}_l)},$$

where $\hat{\theta}_k = \{U, \beta, \mu_k, \Sigma_k\}$. From these probabilities, each observation can be given to a cluster by $z_i = \arg \max_k o_{ik}$.

F-Step The projection matrix U is computed such that Fisher's criterion is maximized in the latent space,

$$U \leftarrow \underset{U}{\arg \max} \operatorname{trace} \left((U^t S U)^{-1} U^t S_B U \right), \quad \text{w.r.t.} \quad U^t U = Id(d)$$

where S is the variance of the whole dataset and $S_B = \frac{1}{n} \sum_{k=1}^K n_k (m_k - \bar{y})(m_k - \bar{y})^t$ where $n_k = \sum_i o_{ik}$ and \bar{y} the mean of the dataset.

M-Step Knowing the posterior probabilities o_{ik} and the projection matrix U , we compute the new Gaussian parameters by maximizing the likelihood of the observations,

$$\hat{\pi}_k \leftarrow \frac{n_k}{n}, \quad \hat{\mu}_k \leftarrow \frac{1}{n_k} \sum_{i=1}^n o_{ik} U^t y_i, \quad \hat{\Sigma}_k \leftarrow U^t C_k U, \quad \hat{\beta}_k \leftarrow \frac{\text{trace}(C_k) - \sum_{j=1}^d u_j^t C_k u_j}{p - d},$$

where u_j is the j -th column of U and $C_k = \frac{1}{n_k} \sum_{i=1}^n o_{ik} (y_i - m_k)(y_i - m_k)^t$ the empirical covariance matrix of the cluster k .

3.3 Sparse version

Yet, the use of latent space introduces dimension reduction and computation efficiency. Nevertheless the back-projection from the latent space to the observation space can involve all the original features. To do feature selection, the projection matrix U has to be sparse. [13] proposed 3 methods to enforce sparsity: 1) After a standard F-step, compute a sparse approximation of U independently of the Fisher criterion, 2) Compute the projection with a modified Fisher criterion with a L_1 penalty on U , 3) Compute U from the Fisher criterion using a penalized SVD algorithm.

4 Application to swimmer coordination

The clustering is done in two steps: 1) A clustering on cycle data. Here an observation is just one swimming cycle. This clustering has two purposes, a) give a label to each cycle b) select which phase samples over the 100 are informative through sparsity. 2) A clustering on trials. Each trial can be described now by a sequence of cycle labels learned at the first step. Features for this clustering consist in the transition matrix of the sequence with its diagonal put to zero. The number of cluster is chosen by analysis of the Bayesian information criterion (BIC).

For the first clustering level, analysis of the BIC (Tab. 1) highlights the existence of 11 clusters within the whole set of data. The mean coordination of these clusters are represented at Figure 2a.

This result advocates for qualitative reorganizations of motor behavior during motor learning, as each learner visited between 9 and 11 different clusters during their sessions. For instance, the mean and standard deviation of one cluster (n°8) is presented in Figure 2b.

In order to differentiate the effect of the different instructions on the learning process, Table 2 shows the distribution of each emerging cluster across the different learning conditions. Interestingly, the use of different additional instructions

led to the exhibition of different preferred patterns of coordination. For instance, the group who received an analogy exhibited preferably clusters 3, 7, 8 and 9, whereas clusters 2, 4 and 10 were inhibited. In the meantime, the use of the prescriptive instruction preferably led to the use of cluster 5 and inhibited the use of clusters 2, 6 and 10. This result is a key point of the experiment, validating the possibility of guiding the exploration during learning and by extension the result of the learning process with using different types of instructions during the practice.

On Figure 2c, we have superimposed a typical coordination curve and, in gray bars, the back-projection of latent space into observation space to see induced sparsity from the first level. The height of a bar at a feature $i \in [1 \dots p]$ is proportional to $\sum_{j=1}^d |U_{ij}|$. A null value shows that the corresponding feature is not involved in the projection to the latent space, i.e. it is not selected by the F-Step or it is squeezed by the sparsity; therefore it can be considered not relevant to build the clusters. Interestingly, only key points of the movement have high values, thus the Fisher-Em algorithm is able to select key points without any prior knowledge.

The second level of cluster analysis, based on the transition matrix during each trial showed the existence of six different clusters. More specifically, Figure 3 highlights the preferred transitions exhibited by each emerging cluster. Interestingly, the group who showed the highest number of preferred transition (i.e. cluster 6) was associated with the learning group that did not receive any instruction. In that sense, this second level of cluster analysis allowed to highlight the use of temporary additional information during learning in order to modify the learning search strategy, namely by impacting the preferred transitions.

5 Perspectives

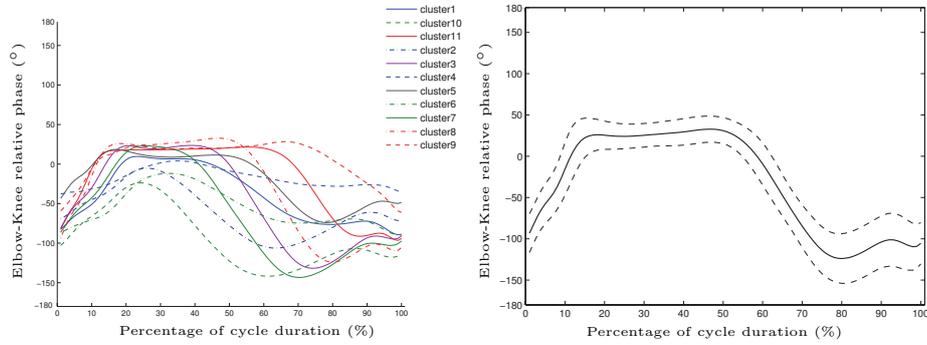
These preliminary experiments show that we can apply efficiently the Fisher-EM clustering on highly correlated features. Interestingly, the induced sparsity corresponds to key points of the coordination phase. Now, a qualitative work needs to be undertaken to qualify clusters of trials in term of learning condition and learning dynamics.

Table 1: Analysis of the BIC for the first level showing a plateau at 11 clusters

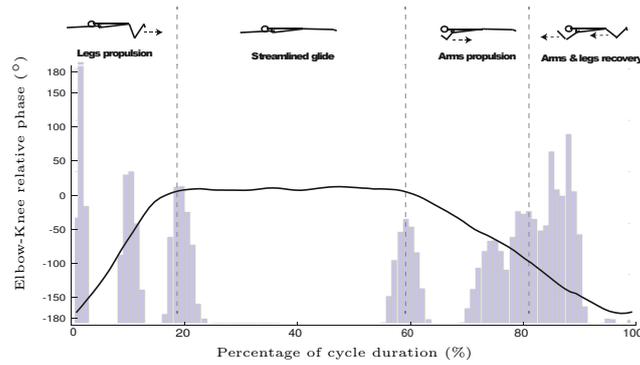
Number of clusters	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
BIC value ($\times 10^7$)	-1.23	-1.21	-1.18	-1.18	-1.15	-1.14	-1.13	-1.11	-1.08	-1.04	-1.05	-1.05	-1.07	-1.04	-1.04	-1.05

Table 2: Distribution (in %) of each cluster according to learning conditions

Cluster	Control	Analogy	Pacer	Prescription	Total	Cluster	Control	Analogy	Pacer	Prescription	Total
1	24.62	35.15	14.39	25.84	100	7	23.12	39.03	17.25	20.60	100
2	47.85	7.16	28.77	16.22	100	8	16.72	46.56	17.41	19.31	100
3	17.60	45.59	12.07	24.74	100	9	14.69	41.91	18.04	25.36	100
4	61.18	4.59	10.98	23.26	100	10	27.81	5.95	64.36	1.87	100
5	28.73	25.73	1.86	43.69	100	11	19.46	26.18	26.34	28.01	100
6	44.25	16.70	23.95	15.09	100						



(a) a) Mean patterns of coordination for each cluster (b) b) Mean pattern for cluster 8 (black line), standard deviation (dotted line)



(c) c) A typical coordination and superimposed induced sparsity

Fig. 2: First clustering level

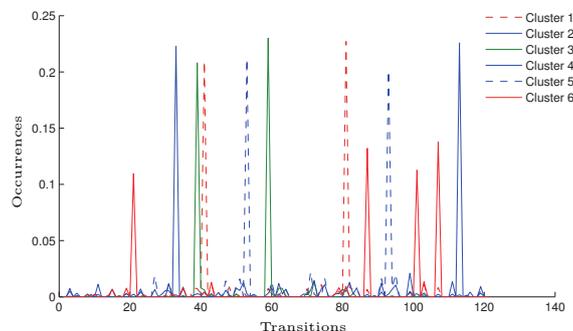


Fig.3: Mean patterns of possible transitions within a trial for the 2nd level clustering, please note that there are $121 = 11 \times 11$ possible transitions as there is 11 clusters at first level

References

1. Scott Kelso, J.: *Dynamic Patterns: the self-organization of brain and behavior*. MIT Press (1995)
2. Müller, H., Sternad, D.: Decomposition of variability in the execution of goal-oriented tasks: Three components of skill improvement. *Journal of Experimental Psychology: Human Perception and Performance* **30**(1) (2004) 212–233
3. Nourrit, D., Delignières, D., Caillou, N., Deschamps, T., Lauriot, B.: On discontinuities in motor learning: A longitudinal study of complex skill acquisition on a ski-simulator. *Journal of Motor Behavior* **35**(2) (2003) 151–170
4. Chow, J.Y., Davids, K., Button, C., Rein, R.: Dynamics of movement patterning in learning a discrete multiarticular action. *Motor Control* **12**(3) (2008) 219–240
5. Gel'fand, I.M., Tsetlin, M.L.: Some methods of control for complex systems. *Russian Mathematical Survey* **17** (1962) 95–116
6. Bouveyron, C., Brunet, C.: Simultaneous model-based clustering and visualization in the fisher discriminative subspace. *Statistics and Computing* **22**(1) (2012) 301–324
7. Seifert, L., Leblanc, H., Chollet, D., Delignières, D.: Inter-limb coordination in swimming: Effect of speed and skill level. *Human Movement Science* **29**(1) (2010) 103–113
8. Seifert, L., Chollet, D.: A new index of flat breaststroke propulsion: A comparison of elite men and women. *Journal of Sports Sciences* **23** (March 2005) 309–320
9. Tagaki, H.: Differences in stroke phases, arm-leg coordination and velocity fluctuation due to event, gender and performance level in breaststroke. *Sports Biomechanics* **3** (2004) 15–27
10. Hamill, J., Haddad, J.M., McDermott, W.J.: Issues in quantifying variability from a dynamical systems perspective. *Journal of Applied Biomechanics* **16** (2000) 407–418
11. Komar, J., Chow, J.Y., Chollet, D., Seifert, L.: Effect of analogy instruction on learning a complex motor skill. *Journal of Applied Sport Psychology* (in press)
12. Bouveyron, C., Brunet, C.: Theoretical and practical considerations on the convergence properties of the Fisher-EM algorithm. *J. Multivariate Analysis* **109** (2012) 29–41

13. Bouveyron, C., Brunet, C.: Discriminative variable selection for clustering with the sparse Fisher-EM algorithm. Technical report, <http://hal.archives-ouvertes.fr/hal-00685183>

A.8 Improved Model-Free Gait Recognition Based on Human Body Part

Reference

[Rid+17] Imad Rida et al. “Improved Model-Free Gait Recognition Based on Human Body Part.” In: *Biometric Security and Privacy: Opportunities & Challenges in The Big Data Era*. Ed. by Richard Jiang et al. Signal Processing for Security Technologies. Cham: Springer International Publishing, 2017, pp. 141–161. ISBN: 978-3-319-47301-7. DOI: 10.1007/978-3-319-47301-7_6. URL: https://doi.org/10.1007/978-3-319-47301-7_6

Chapter 6

Improved Model-Free Gait Recognition Based on Human Body Part

**Imad Rida, Noor Al Maadeed, Gian Luca Marcialis, Ahmed Bouridane,
Romain Herault, and Gilles Gasso**

6.1 Introduction

In the past years with frequent terrorist attacks, a considerable number of surveillance cameras have been installed in public places, train stations, and airports and many research efforts have been devoted to build intelligent systems able to analyze the visual data in order to extract information about the behavior of humans in scenes. Ideal intelligent monitoring system should be able to automatically analyze the collected video data, detect the suspicious or endangering human behavior, and give out an early warning before the adverse event happens. A system which detects abnormal behavior should also be able to identify all the suspicious persons in the scene, and track them across the zones. Monitoring system requires not only to estimate the location and behavior, but also to obtain the identity information.

I. Rida (✉) • R. Herault • G. Gasso
LITIS EA 4108 - INSA de Rouen, Avenue de l'Université, Saint Etienne du
Rouvray 76801, France
e-mail: imad.rida@insa-rouen.fr; romain.herault@insa-rouen.fr; gilles.gasso@insa-rouen.fr

N. Al Maadeed
Department of Computer Science and Engineering, Qatar University, Doha, Qatar
e-mail: n.alali@qu.edu.qa

G.L. Marcialis
Department of Electrical and Electronic Engineering (DIEE), University of Cagliari,
09123 Cagliari, Italy
e-mail: marcialis@diee.unica.it

A. Bouridane
Department of Computer Science and Digital Technologies, Northumbria University,
Newcastle upon Tyne, UK
e-mail: ahmed.bouridane@northumbria.ac.uk

Gait is the most suitable biometric modality in the case of intelligent video surveillance. In monitoring scenes, people are usually distant from cameras, which makes most of biometric features not suitable even the use of face for identification. The drawbacks are obvious, for example, view angle variations and occlusions cause the impossibility to capture the full faces and distance brings low-resolution face images. Therefore, face cannot always achieve good performances in practice. In contrast, gait is a behavioral biometric, including not only individual appearance, such as limb, leg length, and width, but also the dynamic information of individual walking. Gait as a biometric trait can be seen as advantageous over other forms of biometric identification techniques for the following reasons:

- The gait of a person walking can be extracted and analyzed from distance without any contact with the sensor.
- The images used in gait recognition can be easily provided by low-resolution, video-surveillance cameras.
- The gait of an individual is difficult to disguise, by trying to do so the individual will probably appear more suspicious.

Gait recognition consists on discriminating among people by the way or manner they walk. Techniques can be classified into two main categories: model based and model-free approach. Model based approach [5, 27] models the person body structure that estimates static body parameters over time (i.e., trajectory, limb lengths, etc.). This process is computationally intensive since it needs to model and track the subject body. The model-free approach does not recover a structural model of human motion. It uses the features extracted from the motion or shape and hence requires much less computation (see Table 6.1). Furthermore, dynamic information results in better recognition performance than its static counterpart [34]. These motivate researchers to develop new feature representations in model-free approach context.

There exists a considerable amount of work in the context of model-free approach. Benabdelkader et al. [4] introduced a self similarity representation to measure the similarity between pairs of silhouettes. Collins et al. [7] proposed a template based silhouette matching in some key frames. Hayfron-Acquah et al. [11] suggested a contour based representation by analyzing the symmetry of human motion using the Generalized Symmetry Operator. Lee et al. [23] introduced a novel spatio-temporal representation called Shape Variation Based Frieze Pattern which aims to capture the motion information over time. Kobayashi and Otsu [20] used Cubic Higher-order Local Auto-Correlation to extract gait features. Lu and Zhang [25] used multiple gait feature representations based on Independent Component

Table 6.1 Comparison between model based and model-free approach gait recognition

	Model-free	Model based
Complexity	✓	✗
Covariates	✗	✓
Calculation	✓	✗

Analysis and Genetic Fuzzy Support Vector Machine. Huang et al. [16] presented a manifold based approach for cross-speed recognition. Hu et al. [13] proposed an incremental framework based on optical flow, including dynamics learning, pattern retrieval, and recognition. Liu et al. [24] integrated gait recognition in person re-identification. Hu et al. [14] suggested a View-invariant Discriminative Projection method by a unitary linear projection to improve the discriminative ability of multiview gait features. Hu et al. [12] introduced a gait modeling method for gender classification.

Recent trends seem to favor Gait Energy Image (GEI) representation suggested by Han and Bhanu [10]. It is a spatio-temporal representation of the gait obtained by averaging the silhouettes over a gait cycle. For the recognition step, Component Discriminant Analysis (CDA) was applied, which applies Principal Component Analysis (PCA) followed by Multiple Discriminant Analysis (MDA). A considerable amount of works use GEI representation. Yu et al. [36] applied a template matching on GEI without any dimensionality reduction and feature selection. Tao et al. [32] used Gabor filters to extract information from GEI and a General Tensor Discriminant Analysis for recognition. Xu et al. [35] presented an extension of Marginal Fisher analysis to address the problem of gait recognition.

The main challenge of model-free gait recognition is coping with various intra-class variations caused by the presence of shadows, clothing variations, and carrying conditions. Segmentation and view angle are further causes of recognition error [10, 26, 36]. To overcome the limitations of GEI presentation, several approaches have been proposed. Bashir et al. [3] introduced a feature selection method named Gait Entropy Image (GEnI). It computes entropy for each pixel to distinguish static and dynamic pixels of GEI. The GEnI represents a measure of feature significance. In the same context Bashir et al. [2] suggested a gait representation by a weighted sum of the optical flow corresponding to each direction of human motion. An unsupervised method is used to select GEI pixels based on their intensity value [1]. Dupuis et al. [8] introduced a feature selection method based on Random Forest feature rank algorithm. Rida et al. [29] estimated a mask based on pixel variations. In the same context, Rida et al. [28] proposed a method which selects the human dynamic body part. Jeevan et al. [17] introduced a gait representation called Gait Pal and Pal Entropy Image. Kusakunniran [21, 22] proposed a framework to construct gait feature directly from a raw video. Rokanujjaman et al. [31] introduced a novel frequency-domain gait entropy representation. Choudhury and Tjahjadi [6] proposed a View-Invariant Multiscale Gait Recognition (VI-MGR) method. Zeng and Wang [38] introduced a novel method to cope with the problem of walking speed. Recently, Rida et al. [30] used the Modified Phase Only Correlation which is an improved version of the Phase Only Correlation algorithm using a band-pass-type spectral weighting function in order to achieve superior performances.

This chapter proposes a new framework to mitigate the effect of the intra-class variation of GEI representation. Contributions are summarized as follows:

- A horizontal motion vector is proposed that is more reliable and better characterizes the gait than the pixel-wise motion.

- A human body-part selection method is proposed based on group Lasso to cluster the individual dynamic lines into homogeneous parts of human body.
- Feature selection set is separated from the training set to enhance the generalization of body-part selection.
- For view angle variations, a pose estimation method is proposed which is capable to compare a query gait sample without prior knowledge of its view angle with the corresponding gait sequences with the same view angle in the training dataset.

6.2 Proposed Method

Among the available feature representations we choose GEI that is an effective representation, a good compromise between the computational cost and the recognition performance [3]. Figure 6.1 shows our framework of part selection, training and testing, divided into two modules. The first one estimates the human body parts based on motion and group Lasso and selects the discriminative part that is also robust to the intra-class variation. The estimated body parts should not be overspecialized for a particular training set [8]. Therefore, we perform it on a separated feature selection set. The second module applies CDA to the part of GEI features of the training data selected in the first module. Gait recognition performance is measured by Correct Classification Rate (CCR) on the testing dataset.

It has been found that the gait of an individual is characterized much more by the horizontal than the vertical motion [9]. Therefore, instead to estimate the motion of each pixel [3], we propose to estimate the horizontal motion by taking the Shannon entropy of each row from the GEI. The resulting column vector is named as motion based vector.

To generalize the contiguous human body parts from the motion based vector, we further propose to apply group Lasso learning algorithm to segment the motion based vector into shared blocks with similar motion value. The body part with the highest average motion value over the selection dataset is selected, which is discriminative and robust to the intra-class variation.

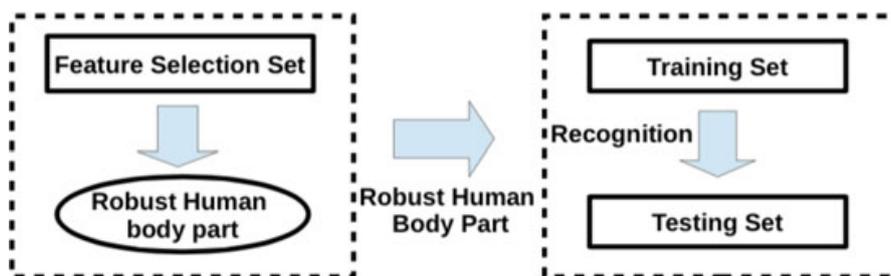


Fig. 6.1 Scheme of our part selection, training and testing

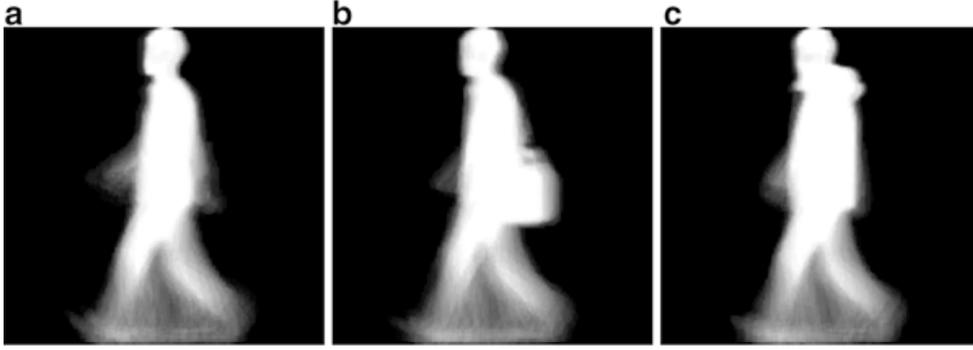


Fig. 6.2 Gait energy image of an individual under different conditions. (a) Normal walk. (b) Carrying bag. (c) Wearing coat

6.2.1 The Proposed Motion Based Vector

GEI is a spatio-temporal representation of gait pattern. It is a single grayscale image (see Fig. 6.2) obtained by averaging the silhouettes extracted over a complete gait cycle [10] as

$$\mathbf{G} = \frac{255}{T} \sum_{t=1}^T \mathbf{B}(t) \quad (6.1)$$

where $\mathbf{G} = \{g_{i,j}\}$ is GEI, $1 \leq i \leq N$ and $1 \leq j \leq M$ are the spatial coordinates, T is the number of the frames of a complete gait cycle, and $\mathbf{B}(t)$ is the silhouette image of frame t .

For each GEI, a motion based vector $\mathbf{e} \in \mathbb{R}^N$ shown in Fig. 6.3 is generated by computing the Shannon entropy of each row of GEI. The element of the motion based vector \mathbf{e} is given by:

$$e_i = - \sum_{k=0}^{255} p_k^i \log_2 p_k^i \quad (6.2)$$

where p_k^i is the probability that the pixel value k occurs in the i th row of image \mathbf{G} , which is estimated by:

$$p_k^i = \frac{\#(g_{i,j} = k)}{M}; \forall j \in [1, M] \quad (6.3)$$

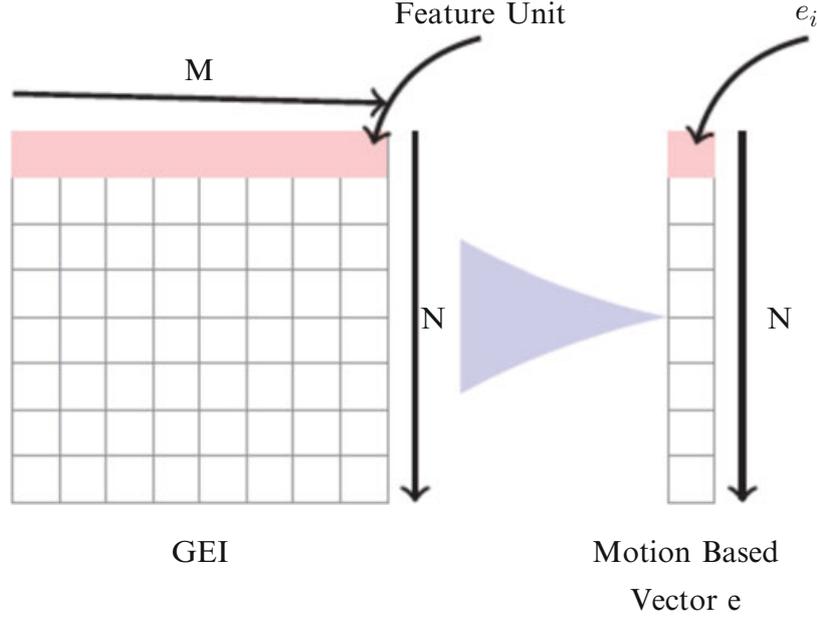


Fig. 6.3 Illustration of the motion based vector

6.2.2 Group Lasso for Multiple Change-Point Detection

Let P motion based vectors $\{\mathbf{e}_k\}_{k=1}^P$ of P GEIs stored in $N \times P$ matrix \mathbf{E} . The aim is to detect the shared change-point locations across all motion based vectors $\{\mathbf{e}_k\}_{k=1}^P$ by approximating matrix $\mathbf{E} \in \mathbb{R}^{N \times P}$ by a matrix $\mathbf{V} \in \mathbb{R}^{N \times P}$ of piecewise-constant vectors that share change-points. This can be achieved by resolving the following convex optimization problem:

$$\min_{\mathbf{V} \in \mathbb{R}^{N \times P}} \|\mathbf{E} - \mathbf{V}\|_F^2 + \lambda \sum_{i=1}^{N-1} \|\mathbf{v}_{i+1} - \mathbf{v}_i\|_1 \quad (6.4)$$

where \mathbf{v}_i is the i th row of \mathbf{V} and $\lambda > 0$. Intuitively, when increasing λ enforces many increments $\mathbf{v}_{i+1} - \mathbf{v}_i$ to converge to zero. This implies that the position of nonzeros increments will be same for all vectors. Therefore, the solution of (6.4) provides an approximation of \mathbf{E} by a matrix \mathbf{V} of piecewise-constant vectors which share change-points. The problem (6.4) is reformulated as a group Lasso regression problem as follows:

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^{(N-1) \times P}} \left\| \bar{\mathbf{E}} - \bar{\mathbf{X}}\boldsymbol{\beta} \right\|_F^2 + \lambda \sum_{i=1}^{N-1} \|\boldsymbol{\beta}_i\|_1 \quad (6.5)$$

where $\bar{\mathbf{X}}$ and $\bar{\mathbf{E}}$ are obtained by centering each column from \mathbf{X} and \mathbf{E} knowing that:

$$\begin{cases} \mathbf{X} \in \mathbb{R}^{N \times (N-1)}; x_{ij} = \begin{cases} 1 & \text{for } i > j \\ 0 & \text{otherwise} \end{cases} \\ \beta_i = \mathbf{v}_{i+1} - \mathbf{v}_i \end{cases} \quad (6.6)$$

The problem (6.5) can be solved based on the group LARS described in [37] which approximates the solution path with a piecewise-affine set of solutions and iteratively finds change-points independently of λ value. The full derivation of the method can be found in [33].

6.2.3 Canonical Discriminant Analysis

On the training dataset, Canonical Discriminant Analysis (CDA) is applied to the GEI features of the robust human body part determined by the group Lasso on the feature selection dataset. The CDA applies PCA followed by a MDA. PCA removes unreliable dimensions that adversely affect the robustness of the classification [18, 19] and hence improves the classification accuracy. MDA maximizes the distance between classes and preserves the distance inside the classes. As suggestion in [10] we retain $2C$ eigenvectors after applying PCA, where C corresponds to the number of classes (the full explanation is found in [15]). The performance of our method is measured by the CCR that is the ratio of the number of correctly classified samples over the total number of samples.

Let n d -dimensional training GEI templates $\{\mathbf{g}_1, \dots, \mathbf{g}_n\}$, where each template is a column vector obtained by concatenating the rows of the corresponding GEI. The discriminative human body part with highest motion is selected using the group Lasso to obtain n d' -dimensional GEI templates $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ where $d' < d$. PCA aims to minimize the following objective function:

$$J_{d''} = \sum_{k=1}^n \left\| \left(\mathbf{m} + \sum_{l=1}^{d''} a_{kl} \mathbf{u}_l \right) - \mathbf{x}_k \right\|^2 \quad (6.7)$$

where $d'' < d' < d$, $\mathbf{m} = \frac{1}{n} \sum_{k=1}^n \mathbf{x}_k$, $\{\mathbf{u}_1, \dots, \mathbf{u}_{d''}\}$ set of orthogonal unit vectors representing new coordinate system of the subspace and a_{kl} is the projection of the k th data over \mathbf{u}_l .

$J_{d''}$ is minimized when $\mathbf{u}_1, \dots, \mathbf{u}_{d''}$ are eigenvectors of the largest eigenvalues of the covariance matrix Σ given by:

$$\Sigma = \sum_{k=1}^n (\mathbf{x}_k - \mathbf{m})(\mathbf{x}_k - \mathbf{m})^T \quad (6.8)$$

The d'' -dimensional feature vector \mathbf{y}_k obtained from \mathbf{x}_k is given by:

$$\mathbf{y}_k = \mathbf{M}_{\text{PCA}} \mathbf{x}_k = [a_1, \dots, a_{d''}]^T = [\mathbf{u}_1, \dots, \mathbf{u}_{d''}]^T \mathbf{x}_k, \quad k = 1, \dots, n \quad (6.9)$$

Suppose that the n d' -dimensional principal vectors $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ belong C classes, as suggestion in [2] we retain $d'' = 2C$ eigenvectors after applying PCA. MDA is a supervised learning method which seeks a transformation matrix \mathbf{W} that maximizes the ratio of the between-class scatter matrix \mathbf{S}_B to the within-class scatter matrix \mathbf{S}_W given by:

$$J(\mathbf{W}) = \frac{|\mathbf{W}^T \mathbf{S}_B \mathbf{W}|}{|\mathbf{W}^T \mathbf{S}_W \mathbf{W}|} \quad (6.10)$$

The within-class scatter matrix in the PCA subspace \mathbf{S}_W is defined as $\mathbf{S}_W = \sum_{c=1}^C \mathbf{S}_c$ where:

$$\left\{ \begin{array}{l} \mathbf{S}_c = \sum_{\mathbf{y} \in \mathcal{D}_c} (\mathbf{y} - \mathbf{m}_c)(\mathbf{y} - \mathbf{m}_c)^T \\ \mathbf{m}_c = \frac{1}{n_c} \sum_{\mathbf{y} \in \mathcal{D}_c} \mathbf{y} \\ \{\mathcal{D}_c\}_{c=1}^C \text{ training data of class } c \text{ of size } n_c \end{array} \right. \quad (6.11)$$

The between-class scatter in the PCA subspace \mathbf{S}_B is given by:

$$\mathbf{S}_B = \sum_{c=1}^C n_c (\mathbf{m}_c - \mathbf{m})(\mathbf{m}_c - \mathbf{m})^T \quad (6.12)$$

where $\mathbf{m} = \frac{1}{n} \sum_{\mathbf{y} \in \mathcal{D}} \mathbf{y}$. $J(\mathbf{W})$ is maximized when the columns of \mathbf{W} are the generalized eigenvectors that correspond to $C - 1$ nonzero eigenvalues in:

$$\mathbf{S}_B \mathbf{w}_r = \lambda_r \mathbf{S}_W \mathbf{w}_r \quad (6.13)$$

where \mathbf{w}_r is the r th column of the matrix \mathbf{W} . The corresponding generalized eigenvectors are denoted by $\mathbf{v}_1, \dots, \mathbf{v}_{C-1}$. The $(C - 1)$ -dimensional feature vector \mathbf{z}_k in the MDA subspace is obtained from the d'' -dimensional principal component vector \mathbf{y}_k :

$$\mathbf{z}_k = \mathbf{M}_{\text{MDA}} \mathbf{y}_k = [\mathbf{v}_1, \dots, \mathbf{v}_{C-1}]^T \mathbf{y}_k, \quad k = 1, \dots, n \quad (6.14)$$

For each training gait template, its gait feature vector is obtained as follows:

$$\mathbf{z}_k = \mathbf{M}_{\text{MDA}} \mathbf{M}_{\text{PCA}} \mathbf{x}_k \quad k = 1, \dots, n \quad (6.15)$$

6.3 Experiments and Results

In this section, we will introduce the dataset, the different experiments performed on it as well as the obtained results.

6.3.1 Dataset

The proposed method is tested on CASIA dataset B [36] to evaluate its ability to handle the carrying, clothing, and view angle variations. CASIA dataset B is a large multiview gait database created in January 2005 containing 124 subjects captured from 11 different view angles using 11 USB cameras around the left-hand side of the walking subject starting from 0° to 180° (see Fig. 6.4).

Each subject has six normal walking sequences (SetA), two carrying conditions sequences (SetB), and two clothing variations sequences (SetC). The first four sequences of setA noted as SetA1 are used for training. The two remaining sequences of SetA noted as SetA2 as well as SetB and SetC are used for testing normal, carrying, and clothing conditions, respectively. For each sequence, GEI of size 64×64 is computed (see Figs. 6.5 and 6.6).

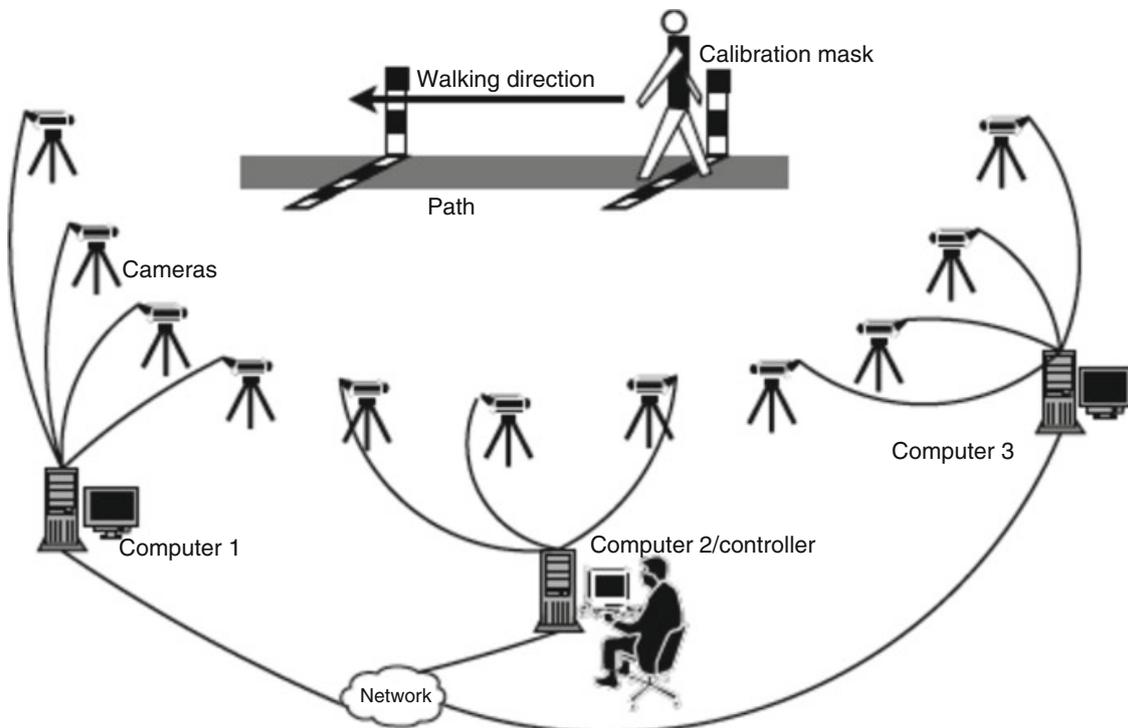


Fig. 6.4 Set-up for gait data collection in CASIA [36]

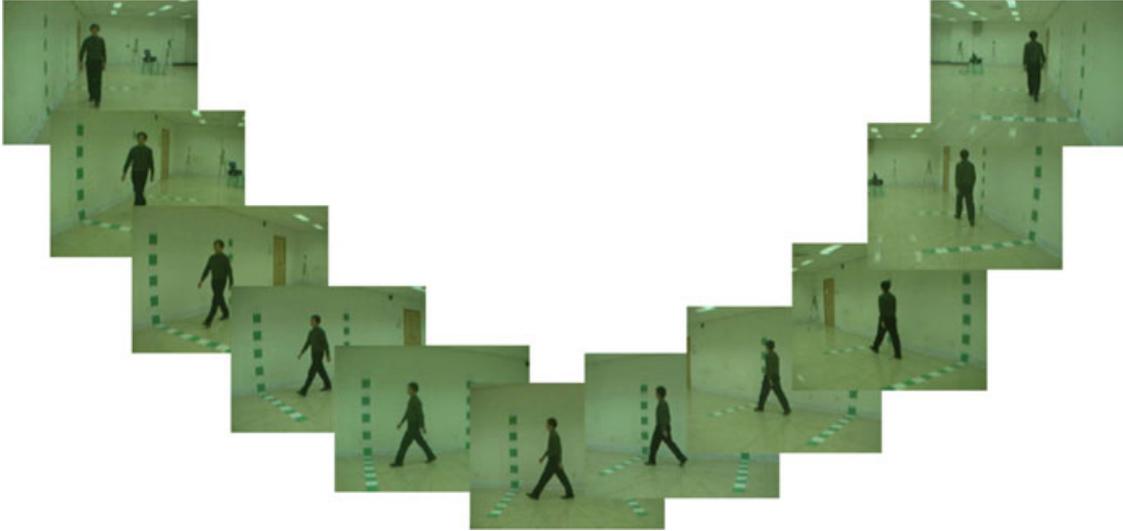


Fig. 6.5 Normal walking conditions under different view angles [36]



Fig. 6.6 Normal, clothing, and carrying conditions under 90° angle [36]

6.3.2 Selected Robust Human Body Part

As we have already mentioned previously, the selected robust human body part shouldn't be overspecialized for a specific training dataset, as a consequence, human body parts are estimated on a feature selection dataset independent from training and testing datasets. To create our body-part selection dataset, we randomly selected 24 GEIs for each variant (normal, carrying, and clothing). All selected GEIs for the feature selection dataset were removed from the training and testing sets. We performed a bagging without replacement of 45 GEIs on the feature selection dataset. The operation was repeated $L = 5$ times.

Figure 6.7 shows the entropy value (y -axis) of all GEIs against feature index (x -axis) for the $L = 5$ experiments. The vertical lines represent the limits of human body parts learnt by the group Lasso on the feature selection dataset.

From Fig. 6.7 we can see that the group Lasso divides the horizontal motion of human body into four parts (the corresponding parts of GEI are shown in Fig. 6.8). It can be seen also in Fig. 6.7 that the part formed by feature units (rows of GEI) from 46 to 64 has the highest mean motion value. It corresponds to the most dynamic part from the human body which contains discriminative information to

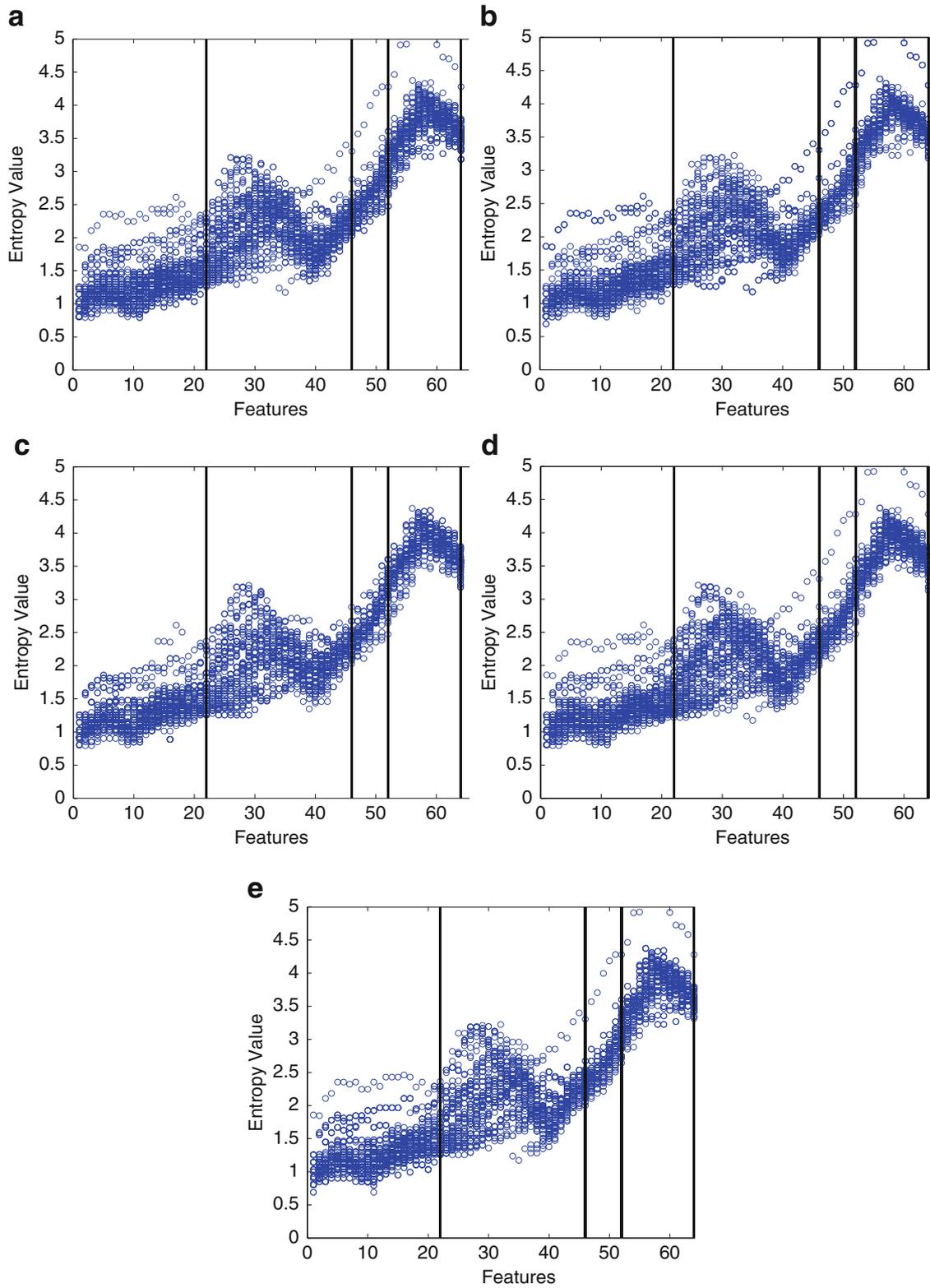


Fig. 6.7 Values of motion based vectors in selection datasets and parts of shared motion value separated by group Lasso. (a) Experiment 1. (b) Experiment 2. (c) Experiment 3. (d) Experiment 4. (e) Experiment 5

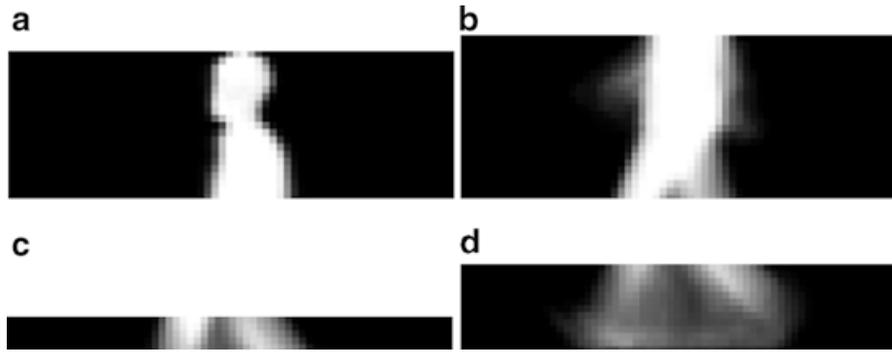


Fig. 6.8 Human body parts of GEI separated by group Lasso. (a) Part 1. (b) Part 2. (c) Part 3. (d) Part 4

differentiate between different people, furthermore it is also robust to the intra-class variations caused by different conditions including clothing, carrying, and view angle variations (see Fig. 6.8c, d).

In the following we will perform experiments under different conditions using the selected human body part.

6.3.3 *Clothing and Carrying Conditions*

In this part, we focus on the effect of the body variations caused by carrying conditions and clothing variations (see Fig. 6.6), so we carried out our experiments under 90° view angle. Table 6.2 compares the performance of our proposed method against the reported by other methods under 90° view angle. It shows that the CCR of our method is marginally lower in the normal and carrying conditions and significantly higher in the clothing variations than all other methods.

It is common in real life that people have different clothes depending on days (warm or cool days) and seasons (summer or winter). Unfortunately, the intra-class variation of the static features (low motion) is mainly caused by the clothing variation that greatly affects the recognition accuracy adversely. It has been demonstrated by Matovski et al. [26] that clothing is the factor that drastically affects the performance of gait recognition. Thus, alleviating the problems caused by the clothing variation has significant meaning for gait recognition.

The proposed method alleviates the clothing variation problem very well as it significantly outperforms all other approaches as shown in Table 6.2. In the normal and carrying conditions, different persons have different clothing conditions but all samples of a same person always have the same clothing condition in the dataset. Thus, the cloths in the normal and carrying conditions in fact undesirably contribute to differentiate persons. Therefore, these recognition rates could be misleading as they do not well reflect the real gait recognition performance. In the next sections, we will further see the problems of testing the gait recognition performance using the training and test data in the same cloth for the same persons. Nevertheless, the

Table 6.2 Comparison of CCRs (in %) from several different algorithms on CASIA database using 90° view

Method	Normal conditions	Carrying conditions	Clothing conditions	Overall	Std
Yu et al. [36]	97.60	32.70	52.00	60.77	33.33
Han and Bhanu [10]	99.60	57.20	23.80	60.20	37.99
Bashir et al. [3]	100.00	78.30	44.00	74.10	28.24
Bashir et al. [2]	97.50	83.87	48.80	76.63	25.09
Bashir et al.[1]	99.40	79.90	31.30	70.20	35.07
Dupuis et al. [8]	98.80	73.80	63.70	78.77	18.07
Rida et al. [30]	93.60	81.70	68.80	81.37	12.40
Rida et al. [29]	95.97	63.39	72.77	77.38	16.77
Hu et al. [13]	94.00	45.20	42.90	60.70	28.86
Kusakunniran [22]	95.40	60.90	52.00	69.43	22.92
Rakanujjaman et al. [31]	97.61	83.87	51.61	77.70	23.61
Kusakunniran [21]	94.50	60.90	58.50	71.30	20.13
Jeevan et al. [17]	93.36	56.12	22.44	57.31	35.47
Our proposed method	98.39	75.89	91.96	88.75	11.59

The bold values correspond to the best results

proposed method performs the best among all approaches on the whole test dataset that contains one-third samples with cloth variation and two-third samples without the cloth variation.

6.3.4 Cross-View Gait Recognition

In real life, subjects are often captured under different view angles; to simulate these conditions we perform experiments in the so-called cross-view gait recognition. In this case, different view angle combinations between training and testing data are used to estimate the recognition performances. Tables 6.3, 6.4, and 6.5 show the results of the body-part cross-view under normal, carrying conditions, and clothing variations, respectively, when Tables 6.6, 6.7, and 6.8 show the same results of whole-body under the same conditions.

The results demonstrate that our body-part method significantly outperforms the whole-body one under cloth variations; however, it has marginally lower performances in normal conditions due to the undesirable contribution of clothing in recognition which was already pointed out previously. From the same results it can be seen that both the whole-body and body-part give good performances when the training view angle is similar to the testing one; however, the performances significantly decrease when the difference between the training view angle and the

Table 6.3 Cross-view body-part recognition under normal conditions (%)

Gallery angle (°)	Probe angle (°)										
	0	18	36	54	72	90	108	126	144	162	180
0	98.37	5.24	1.61	1.21	0.40	0.81	0.81	1.61	0.81	0.81	9.27
18	6.10	98.79	17.74	1.61	0.81	0.81	1.21	1.61	4.44	2.42	2.82
36	3.66	23.79	95.97	32.66	5.65	0.81	1.21	0.81	0.40	3.63	2.42
54	2.03	5.24	33.87	96.77	11.69	4.84	1.61	1.21	0.40	1.61	2.02
72	1.22	2.02	3.23	10.08	98.39	82.26	20.16	1.21	0.81	1.61	2.02
90	1.22	1.21	2.82	7.66	67.74	98.39	48.79	4.84	3.23	1.61	1.21
108	2.03	2.82	4.44	4.44	23.79	67.34	97.18	30.24	4.84	3.63	1.61
126	0.81	2.42	2.42	4.03	5.65	7.26	29.03	95.56	38.31	3.63	1.61
144	0.81	2.02	1.21	2.42	5.24	4.44	6.05	47.18	97.18	2.02	0.81
162	3.66	3.23	0.81	0.81	0.81	0.81	0.81	0.81	1.21	97.98	6.85
180	10.57	2.42	1.61	0.40	0	0.40	0.81	1.61	2.42	3.63	97.58

Bold value correspond to CCR when gallery angle is similar to probe angle

Table 6.4 Cross-view body-part recognition under carrying conditions (%)

Gallery angle (°)	Probe angle (°)										
	0	18	36	54	72	90	108	126	144	162	180
0	72.36	2.02	0.81	0.81	0.40	0	0.40	2.02	1.62	2.04	8.50
18	5.28	73.79	9.68	2.03	2.02	1.79	1.61	2.02	1.62	3.67	2.02
36	4.07	16.94	77.02	27.64	4.44	1.34	2.02	0.81	0	5.31	1.62
54	1.63	6.45	25.40	75.61	10.48	3.57	1.21	1.21	0.81	2.04	2.02
72	1.63	1.61	1.61	10.16	75.00	56.70	15.32	2.02	0.81	2.04	2.83
90	0.81	1.61	2.42	5.69	45.16	75.89	25.00	4.86	2.43	0.82	1.21
108	0.81	0.81	4.03	3.66	14.92	53.57	75.00	22.27	6.88	3.27	2.43
126	1.22	1.21	2.42	2.44	6.85	6.25	29.84	76.52	28.34	2.04	1.21
144	1.22	0.81	1.61	2.03	4.84	4.46	5.24	33.60	77.33	0	0.81
162	2.85	1.21	1.21	1.22	1.21	1.34	0.81	0.81	0.40	74.69	3.24
180	9.76	2.42	0.81	0.81	0.40	0.89	0.81	2.02	1.62	4.08	75.71

Bold value correspond to CCR when gallery angle is similar to probe angle

testing one increases. This makes us conclude that there is an invert relationship between the view angle difference between training and testing data and the performance.

Based on the obtained results, we can clearly understand that conventional methods fail to give good recognition performances in case of the large view angle variations between the training and testing data. Unfortunately, the latter is frequently encountered in real life gait recognition applications. This clearly shows the mandatory to introduce new methods capable to address the very challenging problem of view angle variations.

Table 6.5 Cross-view body-part recognition under clothing variations (%)

Gallery angle (°)	Probe angle (°)										
	0	18	36	54	72	90	108	126	144	162	180
0	80.89	4.03	2.42	1.62	0.81	0.89	0.81	2.43	2.02	0.82	9.27
18	5.28	83.06	12.90	2.02	0.81	0.89	0.81	1.62	2.83	2.04	3.23
36	2.44	19.35	85.08	29.55	6.85	2.68	1.61	1.62	0.40	2.45	1.21
54	1.63	5.65	30.24	87.04	10.08	4.02	1.21	0.81	0	0.82	0.81
72	1.22	1.61	2.42	12.96	91.13	62.95	18.55	0.40	0	0.82	0.81
90	0.41	1.61	3.23	6.07	60.48	91.96	40.32	4.05	2.43	1.63	1.61
108	1.63	3.23	1.61	3.64	18.95	56.25	88.71	31.58	4.45	3.67	1.61
126	1.22	1.61	1.61	4.05	4.44	4.91	22.18	87.04	40.08	3.67	1.61
144	2.03	1.21	1.61	2.02	5.65	1.79	4.03	27.13	90.28	2.86	1.61
162	3.25	2.82	2.02	1.62	1.21	1.34	1.21	1.62	1.21	86.94	6.85
180	9.35	2.02	2.02	0.81	0.81	0.89	0.81	1.62	0.81	2.86	84.27

Bold value correspond to CCR when gallery angle is similar to probe angle

Table 6.6 Cross-view whole-body recognition under normal conditions (%)

Gallery angle (°)	Probe angle (°)										
	0	18	36	54	72	90	108	126	144	162	180
0	100	70.16	14.92	5.24	2.42	2.02	0.81	0.81	4.44	15.32	40.32
18	82.11	100	92.74	16.13	3.63	1.21	2.42	4.84	15.32	21.77	31.85
36	38.21	94.76	99.19	85.89	30.24	15.73	12.50	22.58	20.97	21.77	9.27
54	9.76	27.82	92.34	99.19	70.97	35.48	21.77	27.42	23.79	6.05	6.45
72	6.10	4.03	16.13	63.31	99.19	98.79	74.19	14.92	4.84	5.24	4.44
90	2.03	2.02	6.45	17.34	98.79	100	97.18.79	22.98	6.05	2.82	2.42
108	2.44	0.81	8.06	33.06	79.84	97.98	99.60	91.53	22.58	3.63	2.42
126	6.50	4.84	12.10	31.45	47.58	50.81	90.73	98.39	94.76	15.32	6.45
144	13.01	15.73	27.02	19.35	8.87	6.45	31.45	95.16	99.19	34.68	11.29
162	20.73	25.00	15.32	6.05	0.81	0.81	1.21	2.42	6.05	99.60	70.56
180	52.44	18.55	12.10	4.84	3.23	1.61	0.81	2.42	9.27	77.42	100

Bold value correspond to CCR when gallery angle is similar to probe angle

Starting from the observation that the view angle similarity between the training and testing data helps to give good recognition performances, we introduce in the following section a novel method named “gait recognition without prior knowledge of the view angle” capable to estimate the view angle of the testing samples to compare them to training samples with similar view angle and as a consequence improve the recognition performances.

Table 6.7 Cross-view whole-body recognition under carrying conditions (%)

Gallery angle (°)	Probe angle (°)										
	0	18	36	54	72	90	108	126	144	162	180
0	83.74	45.56	14.92	6.50	4.44	2.23	1.61	2.02	2.83	6.53	21.46
18	54.07	79.44	54.03	11.79	4.44	0.45	1.21	4.45	5.67	10.20	10.53
36	27.64	55.24	74.60	46.34	16.13	6.70	3.63	7.69	6.48	8.98	5.26
54	4.88	14.52	48.79	69.11	37.90	23.21	10.08	11.74	9.31	8.98	5.67
72	5.69	4.44	7.66	24.80	59.68	47.77	23.79	8.91	4.86	3.67	5.26
90	2.03	2.42	3.63	11.79	47.98	55.80	39.92	9.72	4.05	2.86	2.43
108	2.44	0.81	4.44	15.45	40.73	50.89	59.27	35.22	12.55	4.08	2.83
126	4.07	3.23	9.68	20.73	27.02	28.57	38.31	62.35	43.32	8.57	4.45
144	5.69	8.87	15.32	11.38	5.24	5.36	8.47	48.58	70.45	17.96	8.10
162	10.98	13.71	5.24	2.44	1.61	1.79	1.61	2.43	4.05	67.35	31.17
180	29.27	13.71	6.05	3.66	2.42	0.45	2.02	2.02	6.48	34.29	76.11

Bold value correspond to CCR when gallery angle is similar to probe angle

Table 6.8 Cross-view body-part recognition under clothing variations (%)

Gallery angle (°)	Probe angle (°)										
	0	18	36	54	72	90	108	126	144	162	180
0	28.05	14.52	5.65	2.02	1.21	0.45	1.21	1.62	3.64	6.94	7.66
18	11.38	25.81	21.37	6.48	4.03	3.57	2.82	4.05	6.07	6.94	5.65
36	8.94	18.95	31.05	23.48	8.87	6.70	4.44	6.88	5.26	7.76	2.42
54	1.22	7.66	20.97	28.34	16.53	7.59	6.85	6.88	4.45	2.45	0.40
72	0.81	1.61	2.42	9.31	29.44	22.32	12.50	4.86	1.62	1.63	2.02
90	2.85	1.61	2.02	7.29	16.53	25.45	14.92	5.67	1.62	2.04	0
108	0.81	1.61	3.23	5.26	13.71	17.86	24.60	12.96	5.26	1.63	0.40
126	1.22	2.02	3.23	5.26	10.48	11.61	23.39	31.58	19.43	1.22	1.21
144	5.28	5.65	7.26	8.50	6.45	3.13	6.05	25.91	37.25	4.08	3.23
162	5.28	6.45	7.26	5.67	1.21	1.34	0.81	2.02	4.45	31.02	12.10
180	10.16	7.66	5.24	1.21	1.61	1.79	2.02	2.83	4.45	12.24	30.65

Bold value correspond to CCR when gallery angle is similar to probe angle

6.3.5 Gait Recognition Without Prior Knowledge of the View Angle

The framework in Fig. 6.9 is designed to recognize individuals without a prior knowledge of the viewpoint. Towards this end, the first step consists on estimating the pose of the query subject using the selected human body part, i.e., row 46–64 (it has been explained above how the body part is selected using the group Lasso of motion). For this aim, a simple knn with $k = 1$ is used to find the group of training samples that have the similar pose to that of the query subject. The results of pose

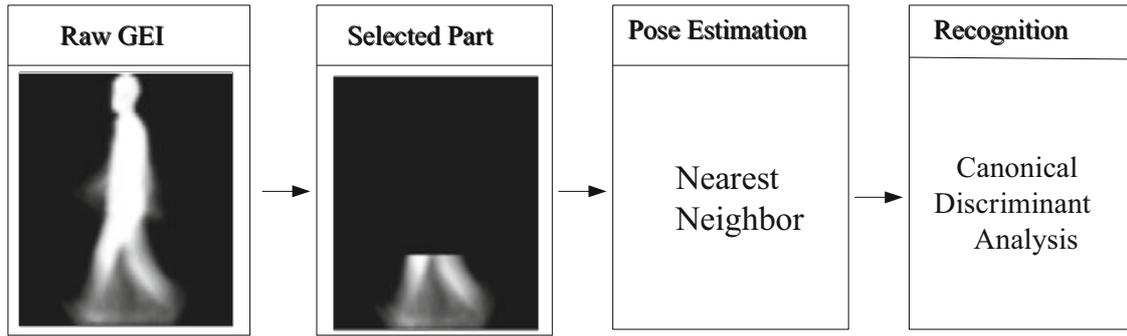


Fig. 6.9 Framework of view angle variation without prior knowledge of the view angle

Table 6.9 Pose estimation–confusion matrix (%)

Real angle (°)	Predicted angle (°)										
	0	18	36	54	72	90	108	126	144	162	180
0	98.78	0.27	0	0	0	0	0	0	0.40	0	0.54
18	0.40	97.58	1.34	0	0	0.13	0	0.13	0.26	0	0.13
36	0.26	1.20	97.31	0.80	0	0	0	0	0.40	0	0
54	0.13	0.13	0.8	98.65	0	0	0.13	0	0.13	0	0
72	0	0.26	0.13	0	98.92	0.13	0.40	0.13	0	0	0
90	0	0.14	0	0.43	0.43	98.41	0.57	0	0	0	0
108	0	0	0	0.13	0	1.34	97.71	0.53	0	0.26	0
126	0	0	0	0.13	0	0	0.40	98.92	0	0.26	0.26
144	0	0.13	0.13	0	0	0	0.13	0.26	97.57	1.48	0.26
162	0	0.27	0.13	0.13	0	0	0	0	1.62	97.83	0
180	1.07	0.26	0	0	0	0	0	0.13	0	0	98.51

Bold values correspond to performance of well-predicted angles

estimation are shown in Table 6.9, it can be seen that the human body part selected by the group Lasso is very discriminative and we are able to estimate the pose of the query subjects of all the dataset with an error less than 3 % for all view angles from 0° to 180°.

The next step consists in identifying the query subject among the group of training samples with the same pose using CDA, which corresponds to PCA followed by MDA (it has been well introduced above in Sect. 6.2.3). Results are shown in Tables 6.10, 6.11, and 6.12, which, respectively, record the CCR of our proposed body-part selection approach, the approach that uses the whole-body, and the VI-MGR method which has been introduced by Choudhury and Tjahjadi in [6] especially to deal with the problem of view angle variations.

Results in these tables clearly show that our proposed body-part selection method significantly outperforms VI-MGR and the approach without the part selection (whole GEI template) for all 11 view angle variations in the case of the clothing variation (see Fig. 6.10). On the whole test dataset that contains one-third samples

Table 6.10 Selected body-part CCR (%) without prior knowledge of view angle

	Test angle (°)										
	0	18	36	54	72	90	108	126	144	162	180
Normal	97.97	98.79	96.37	96.77	98.39	97.98	97.18	95.56	96.77	97.98	97.58
Carrying	72.76	72.58	75.81	76.42	75.81	73.66	74.60	76.92	76.11	75.10	76.11
Clothing	80.49	83.47	85.08	87.85	91.53	91.07	87.90	86.23	87.45	84.90	83.06
Overall	83.74	84.95	85.75	87.02	88.58	87.57	86.56	86.24	86.78	85.99	85.59

Table 6.11 Whole-body CCR (%) without prior knowledge of view angle

	Test angle (°)										
	0	18	36	54	72	90	108	126	144	162	180
Normal	100	100	99.19	99.19	99.19	100	99.60	97.98	99.60	99.19	100
Carrying	82.11	77.42	75.81	68.70	59.68	52.68	53.63	60.73	68.02	66.53	72.47
Clothing	26.83	25.81	28.63	27.53	28.63	22.77	23.79	31.17	34.01	29.39	29.84
Overall	69.65	67.74	67.88	65.14	62.50	58.48	59.01	63.30	67.21	65.04	67.44

Table 6.12 VI-MGR CCR (%) without prior knowledge of view angle

	Test angle (°)										
	0	18	36	54	72	90	108	126	144	162	180
Normal	100	99	100	99	100	100	99	99	100	100	99
Carrying	93	89	89	90	77	80	82	84	92	93	89
Clothing	67	56	70	80	71	75	77	75	65	64	66
Overall	86.66	81.33	86.33	89.66	82.66	85	86	86	85.66	85.66	84.66

with cloth variation and two-third samples without the cloth variation, the proposed approach outperforms the no-part selection approach for all view angle variations and outperforms VI-MGR in 8 of the 11 view angle variations (see Fig. 6.11).

The problems of the CCR for normal and carrying conditions are shown in Tables 6.11 and 6.12. It is well known that the maximum gait information is captured for the view angle near 90° and the minimum gait information is captured for the view angle near 0° or 180° . However, while perfect or near perfect CCR is achieved by almost all view angles in normal condition, in carrying condition, visibly higher CCR is achieved for view angles near 0° or near 180° than that for view angles near 90° . This shows that the cloths in the normal and carrying conditions in fact undesirably contribute to differentiate persons. Therefore, these recognition rates could be misleading as they do not well reflect the real gait recognition performance. Figure 6.10 shows CCR of the three approaches on all test data with cloth and view angle variations, which clearly shows the significant performance gain achieved by the proposed approach.

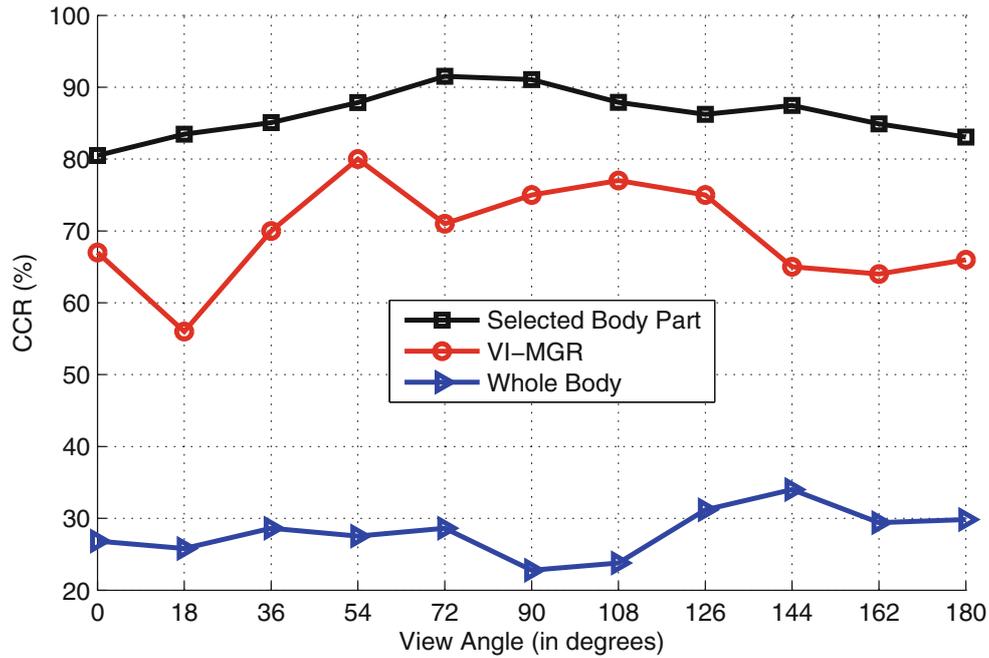


Fig. 6.10 CCR of different approaches on test data with cloth and view angle variations

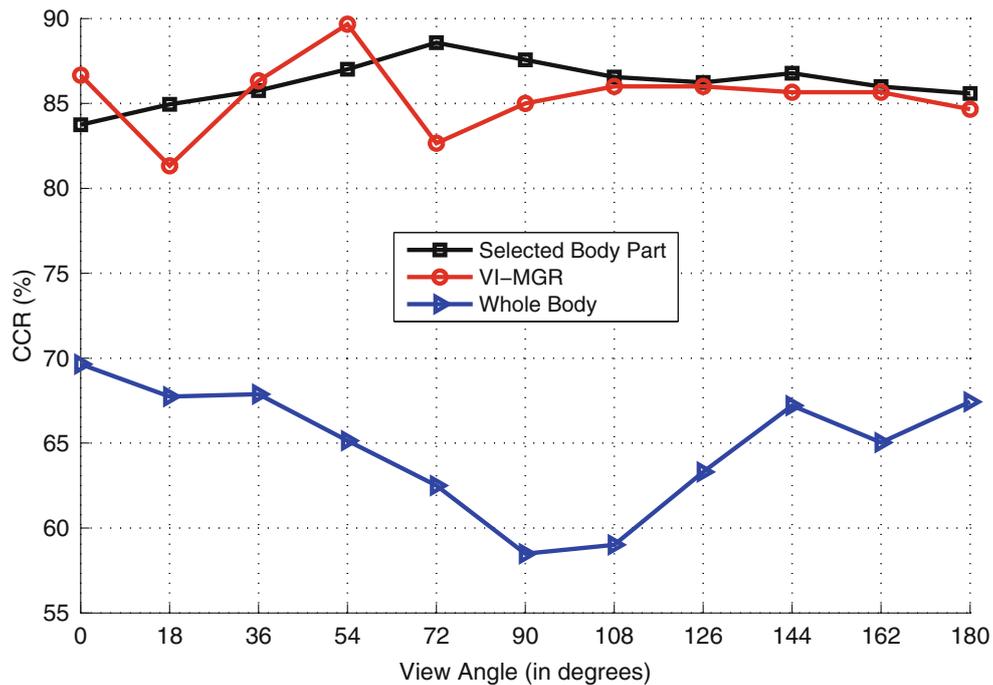


Fig. 6.11 The mean CCR of different approaches on test data under different conditions

6.4 Conclusion

In this chapter we proposed a method that finds the discriminative human body part that is also robust to the intra-class variations for improving the human gait recognition. The proposed method first generates a horizontal motion based vector

from GEI and then applies the group Lasso on the horizontal motion based vectors of a feature selection dataset to learn the discriminative human body part for gait recognition. The learnt human body part is applied to the independent training and test datasets. The proposed method significantly improves the recognition accuracy in the case of large intra-class variation such as the clothing variation. This is verified by the experiments, which show that the proposed methods not only significantly outperforms other approaches in the case of clothing variations but also achieves the overall best performance among all approaches on the whole testing dataset that contains normal, carrying, clothing, and view angle variations.

Acknowledgements Imad Rida and Gilles Gasso would like to acknowledge support and funding from French FUI AAP 15 - Project RCSM, Risk Credit Chain & Supply Chain Management.

References

1. K. Bashir, T. Xiang, S. Gong, Feature selection on gait energy image for human identification, in *IEEE International Conference on Acoustics, Speech and Signal Processing, 2008. ICASSP 2008* (IEEE, New York, 2008), pp. 985–988
2. K. Bashir, T. Xiang, S. Gong et al., Gait representation using flow fields, in *BMVC* (2009), pp. 1–11
3. K. Bashir, T. Xiang, S. Gong, Gait recognition without subject cooperation. *Pattern Recogn. Lett.* **31**(13), 2052–2060 (2010)
4. C. Benabdelkader, R.G. Cutler, L.S. Davis, Gait recognition using image self-similarity. *EURASIP J. Adv. Signal Process.* **2004**(4), 1–14 (2004)
5. I. Bouchrika, M.S. Nixon, Model-based feature extraction for gait analysis and recognition, in *Computer Vision/Computer Graphics Collaboration Techniques* (Springer, Berlin, Heidelberg, 2007), pp. 150–160
6. S.D. Choudhury, T. Tjahjadi, Robust view-invariant multiscale gait recognition. *Pattern Recogn.* **48**(3), 798–811 (2015)
7. R.T. Collins, R. Gross, J. Shi, Silhouette-based human identification from body shape and gait, in *Proceedings of Fifth IEEE International Conference on Automatic Face and Gesture Recognition, 2002* (IEEE, New York, 2002), pp. 366–371
8. Y. Dupuis, X. Savatier, P. Vasseur, Feature subset selection applied to model-free gait recognition. *Image Vis. Comput.* **31**(8), 580–591 (2013)
9. J.P. Foster, M.S. Nixon, A. Prügel-Bennett, Automatic gait recognition using area-based metrics. *Pattern Recogn. Lett.* **24**(14), 2489–2497 (2003)
10. J. Han, B. Bhanu, Individual recognition using gait energy image. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(2), 316–322 (2006)
11. J.B. Hayfron-Acquah, M.S. Nixon, J.N. Carter, Automatic gait recognition by symmetry analysis. *Pattern Recogn. Lett.* **24**(13), 2175–2183 (2003)
12. M. Hu, Y. Wang, Z. Zhang et al., Gait-based gender classification using mixed conditional random field. *IEEE Trans. Syst. Man Cybern. B Cybern.* **41**(5), 1429–1439 (2011)
13. M. Hu, Y. Wang, Z. Zhang et al., Incremental learning for video-based gait recognition with LBP flow. *IEEE Trans. Cybernet.* **43**(1), 77–89 (2013)
14. M. Hu, Y. Wang, Z. Zhang et al., View-invariant discriminative projection for multi-view gait-based human identification. *IEEE Trans. Inf. Forensics Secur.* **8**(12), 2034–2045 (2013)
15. P.S. Huang, C.J. Harris, M.S. Nixon, Recognising humans by gait via parametric canonical space. *Artif. Intell. Eng.* **13**(4), 359–366 (1999)
16. S. Huang, A. Elgammal, J. Lu et al., Cross-speed gait recognition using speed-invariant gait templates and globality? Locality preserving projections. *IEEE Trans. Inf. Forensics Secur.* **10**(10), 2071–2083 (2015)

17. M. Jeevan, N. Jain, M. Hanmandlu et al., Gait recognition based on gait pal and pal entropy image, in *2013 20th IEEE International Conference on IEEE Image Processing (ICIP)* (2013), pp. 4195–4199
18. X. Jiang, Asymmetric principal component and discriminant analyses for pattern classification. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(5), 931–937 (2009)
19. X. Jiang, Linear subspace learning-based dimensionality reduction. *IEEE Signal Process. Mag.* **28**(2), 16–26 (2011)
20. T. Kobayashi, N. Otsu, Action and simultaneous multiple-person identification using cubic higher-order local auto-correlation, in *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004* (IEEE, New York, 2004), pp. 741–744
21. W. Kusakunniran, Attribute-based learning for gait recognition using spatio-temporal interest points. *Image Vis. Comput.* **32**(12), 1117–1126 (2014)
22. W. Kusakunniran, Recognizing gaits on spatio-temporal feature domain. *IEEE Trans. Inf. Forensics Secur.* **9**(9), 1416–1423 (2014)
23. S. Lee, Y. Liu, R. Collins, Shape variation-based frieze pattern for robust gait recognition, in *IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR'07* (IEEE, New York, 2007), pp. 1–8
24. Z. Liu, Z. Zhang, Q. Wu et al., Enhancing person re-identification by integrating gait biometric. *Neurocomputing* **168**, 1144–1156 (2015)
25. J. Lu, E. Zhang, Gait recognition for human identification based on ICA and fuzzy SVM through multiple views fusion. *Pattern Recogn. Lett.* **28**(16), 2401–2411 (2007)
26. D.S. Matovski, M.S. Nixon, S. Mahmoodi et al., The effect of time on gait recognition performance. *IEEE Trans. Inf. Forensics Secur.* **7**(2), 543–552 (2012)
27. S.A. Niyogi, E.H. Adelson, Analyzing and recognizing walking figures in XYT, in *1994 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1994. Proceedings CVPR'94* (IEEE, New York, 1994), pp. 469–474
28. I. Rida, S. Almaadeed, A. Bouridane, Improved gait recognition based on gait energy images, in *2014 26th International Conference on Microelectronics (ICM)* (IEEE, New York, 2014), pp. 40–43
29. I. Rida, A. Bouridane, G.L. Marcialis et al., Improved human gait recognition, in *Image Analysis and Processing-ICIAP 2015* (Springer International Publishing, Cham, 2015), pp. 119–129
30. I. Rida, S. Almaadeed, A. Bouridane, Gait recognition based on modified phase-only correlation. *Signal Image Video Process.* **10**(3), 463–470 (2016)
31. M. Rokanujjaman, M.S. Islam, M.A. Hossain et al., Effective part-based gait identification using frequency-domain gait entropy features. *Multimedia Tools Appl.* **74**(9), 3099–3120 (2015)
32. D. Tao, X. Li, X. Wu et al., General tensor discriminant analysis and gabor features for gait recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(10), 1700–1715 (2007)
33. J.-P. Vert, K. Bleakley, Fast detection of multiple change-points shared by many signals using group LARS, in *Advances in Neural Information Processing Systems* (2010), pp. 2343–2351
34. L. Wang, H. Ning, T. Tan, W. Hu, Fusion of static and dynamic body biometrics for gait recognition. *IEEE Trans. Circuits Syst. Video Technol.* **14**(2), 149–158 (2004)
35. D. Xu, S. Yan, T. Dacheng et al., Marginal fisher analysis and its variants for human gait recognition and content-based image retrieval. *IEEE Trans. Image Process.* **16**(11), 2811–2821 (2007)
36. S. Yu, D. Tan, T. Tan, A framework for evaluating the effect of view angle, clothing and carrying condition on gait recognition, in *IEEE 18th International Conference on Pattern Recognition, ICPR 2006* (2006), pp. 441–444
37. M. Yuan, Y. Lin, Model selection and estimation in regression with grouped variables. *J. R. Stat. Soc. Ser. B (Stat Methodol.)* **68**(1), 49–67 (2006)
38. W. Zeng, C. Wang, Gait recognition across different walking speeds via deterministic learning. *Neurocomputing* **152**, 139–150 (2015)

