Contributions à l'apprentissage profond, applications à la reconnaissance d'écriture manuscrite

Clement Chatelain

Habilitation à diriger les recherches

CONTRIBUTIONS À L'APPRENTISSAGE PROFOND, APPLICATIONS À LA RECONNAISSANCE D'ÉCRITURE MANUSCRITE

Clément Chatelain

Composition du jury :

Laurence LIKFORMAN	*Rapporteuse*	*Stéphane CANU*	*Examinateur*
Christian WOLF	*Rapporteur*	*John Aldo LEE*	*Examinateur*
Christian VIARD-GAUDIN	*Rapporteur*	*Thierry PAQUET*	*Examinateur & Garant*

LITIS EA 4108 - NORMANDIE UNIVERSITÉ - INSA ROUEN NORMANDIE

2019

Remerciements

En premier lieu, j'adresse mes plus sincères remerciements aux membres du jury pour avoir accepté de participer à l'évaluation de ce travail, malgré leur emploi du temps que je sais chargé. Merci en particulier à Laurence Likforman, Christian Viard Gaudin et Christian Wolf de m'avoir fait l'honneur de rapporter sur ces travaux, merci pour votre bienveillance, vos encouragements et pour la justesse de vos remarques. Merci également à John Lee pour les questions pertinentes qui donneront j'espère lieu à des collaborations futures. Et je remercie tout aussi chaleureusement les membres locaux du jury : merci Stéphane pour tes conseils et ta disponibilité, et bien sûr un merci particulier à Thierry qui m'a accompagné vers le monde de la recherche depuis mon DEA il y a bien longtemps ...

La plupart des travaux présentés dans ce document est le fruit de collaborations avec mes doctorants Simon, Selma, Gautier, Luc, Soufiane, Bruno, Benjamin et Denis. J'ai eu (et j'ai encore pour certains) beaucoup de plaisir à travailler avec vous tous. Merci également aux stagiaires, ingénieurs et post docs qui ont pris part à ces recherches.

Je profite de ces quelques lignes pour remercier tous mes collègues et amis du département ASI (pardon ITI), de l'équipe Apprentissage, et les secrétaires pour leur travail formidable. Je réalise la chance que j'ai de travailler avec vous tous au quotidien.

Pour finir, ce travail d'HDR n'aurait pas été possible sans le soutien de ma famille et de mes amis. Merci à mes parents, et à mes trois chéris Aurélie, Léonard, Suzie : je vous aime !

Table des matières

Chapitre 1

Synthèse de mes travaux d'enseignement et de recherche

Introduction

Ce chapitre présente la synthèse de mes activités depuis l'obtention de mon poste de maître de conférences à l'INSA de Rouen Normandie en septembre 2007. Un bref curriculum vitae est proposé en section 1.1, suivi de d'une synthèse des activités d'enseignement en section 1.2 puis d'une synthèse des activités de recherche en section 1.3.

1.1 Curriculum Vitae

Clément Chatelain

Maître de Conférences 61ème section

LITIS/INSA de Rouen Normandie/département ASI

Données personnelles

39 ans, né le 18 août 1979 à Mont Saint Aignan. clement.chatelain@insa-rouen.fr

Marié, 2 enfants http://pagesperso.litislab.fr/cchatelain/

Formation & carrière

- 2016 - 2019 Bénéficiaire de la PEDR
- depuis 2007 Maître de conférences à l'INSA de Rouen / LITIS
- 2006 - 2007 ATER à l'Université de Rouen
- 2003 - 2006 Doctorant à l'Université de Rouen. Jury : C. Viard-Gaudin (rapp.), J.-M. Ogier (rapp.), G. Lorette, L.Heutte (coDir.), T.Paquet (coDir.)
- 2002 - 2003 DEA Perception Traitement de Information (Univ. Rouen, classé 1er)
- 2002 - 2003 DESS GEII (Univ. Rouen, classé 1er)

Activités de recherche

Mots clefs recherche :

- Architectures profondes, modèles statistiques de séquences, réseaux récurrents
- Reconnaissance d'écriture manuscrite, analyse et reconnaissance d'images de document
- Imagerie médicale, segmentation sémantique

Encadrement doctoral (voir frise 1.1) :

- 1 thèse en cours : B.Deguerre
- 6 thèses soutenues : S.Thomas, S.Belgacem, L.Mioulet, G.Bideault, B.Stuner, S.Belharbi

Récapitulatif des publications :

- 13 revues internationales : Pattern Recognition ×3 (IF=5.898), IJDAR ×2 (IF=0.846), Neurocomputing (IF=4.072), Image & Vision Computing (IF=2.747), Journal of Nuclear Medicine (IF=7.439), Computers in Biology & Medicine (IF=2.286), etc.
- 2 revues nationales (Traitement du Signal et Documents Numériques)
- 33 conférences internationales dont 8 de rang A et 12 de rang B (source CORE)

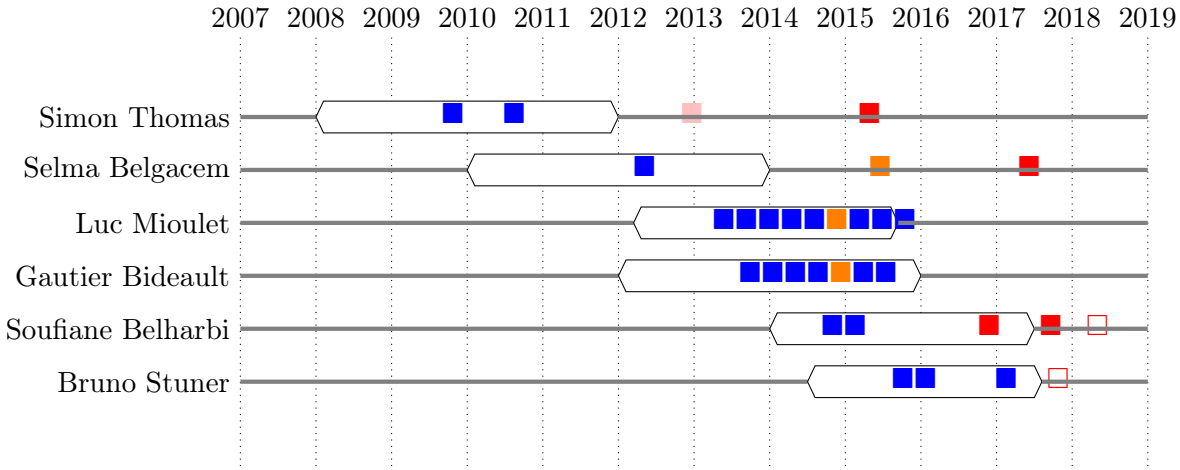

Figure 1.1: Résumé des thèses encadrées soutenues et publications associées.
■ : revue internationale (□ : soumise), ■ : conférence internationale,
■ : participation à un ouvrage collectif, ■ : revue nationale.

Répartition des publications par type et par années (Seuls les articles/communications avec comité de lecture sont comptabilisés) :

Publication	≤2007	'08	'09	'10	'11	'12	'13	'14	'15	'16	'17	'18	2019	**Total**
Revues Int.				2		1			2	2	2	3	1	**13**
Revues Nat.	1						1							**2**
Conf. Int.	5	1	1	2		1	2	3	8	4	2	2	2	**33**
Conf. Nat.	3	1		1		1		3		2		3		**14**
Ouvrage coll.									2					**2**

1.2 Synthèse des activités d'enseignement

1.2.1 Contexte

J'enseigne au sein de l'INSA Rouen Normandie, première école d'ingénieurs publique de Normandie qui compte environ 2000 étudiants répartis sur 5 ans.

Je suis rattaché au département Architecture des Systèmes d'Information (ASI) qui est l'un des 7 départements de spécialité de l'INSA. Créé en 2000, le département ASI forme en 3 ans des ingénieurs en informatique spécialisés dans les domaines des systèmes d'information, de l'ingénierie des données ou de la vision. Les Projets INSA Certifiés (PIC), projets étudiants originaux et innovants en terme de pédagogie, constituent l'un des piliers du département. Depuis 2015, je suis directeur de l'unité « Pédagogie Par Projet »(P3) qui pilote les PIC, responsabilité qui constitue une part importante de mon activité d'enseignement (part estimée à 30 - 40% du temps effectif selon les années).

1.2.2 Les Projets INSA Certifiés

Les PIC [1] ont pour objet de confronter les élèves-ingénieurs à un projet à caractère industriel ou recherche et développement, de taille et de durée significatives. Ils impliquent des équipes de 8 à 9 étudiants travaillant à mi-temps pendant deux semestres. Le défi offert aux élèves-ingénieurs est de proposer et/ou réaliser des solutions techniques à des problématiques réelles soumises par des industriels et en rapport avec leur formation d'architecte des systèmes d'information.

Le succès de ces projets repose selon moi sur plusieurs spécificités qui les démarquent des projets étudiants plus traditionnels. Ces spécificités sont présentées ci-après.

Certification ISO : Les projets PIC sont chaque année certifiés ISO 9001 (version 2015) par un auditeur AFNOR. La certification concerne l'unité P3 qui pilote les projets, ainsi que les projets en eux-mêmes. Cette certification constitue pour les étudiants un apport pédagogique important, qui impose de respecter un cadre de travail en terme de méthodologie de développement, de documentation et de bonnes pratiques de programmation. Cette certification demande un investissement important pour accompagner les étudiants et mener des audits internes préparatoires à l'audit AFNOR. Je suis aidé dans cette tâche par le directeur qualité de l'INSA Pascal Meslier ainsi que par un tuteur qualité Benjamin Coelho De Matos.

Autonomie des équipes PIC : La deuxième spécificité des PIC est l'autonomie des équipes vis-à-vis de leur client, afin de se rapprocher d'une relation client-fournisseur réelle. La

1. Page web de présentation des PIC à destination des clients

négociation du cahier des charges, les livraisons et autres recettes se font donc directement entre l'équipe et le client, sans intervention de l'unité P3. Les équipes sont également autonomes dans leur structuration : attribution des postes (chef PIC, responsable qualité, responsable développement, etc.), organisation de la salle PIC, configuration réseau, etc.

Tutorat des équipes PIC : En début de projet, j'attribue trois tuteurs à chaque équipe, qui interviennent à raison d'une réunion par semaine environ. Le tuteur pédagogique - généralement un enseignant chercheur du département - est garant de l'utilisation d'une démarche ingénieur et scientifique correcte : identification de la problématique, analyse de l'existant, choix justifié d'une solution, méthodologie de développement adoptée. Le tuteur qualité accompagne les équipes et notamment le responsable qualité afin de respecter la norme ISO. Enfin le tuteur communication issu du monde industriel conseille les équipes sur leur communication écrite et orale, et travaille autour de la gestion des conflits et de l'évaluation par les pairs.

La direction de l'Unité P3

En tant que directeur de l'Unité P3, je supervise le pilotage des trois processus de l'Unité : 1) « Rechercher, choisir et contractualiser les PIC », 2) « Réaliser les PIC », et 3) « Manager la qualité ». Je suis également pilote du processus 1 qui consiste à rechercher les sujets PIC auprès des entreprises. Il s'agit de proposer des sujets intéressants couvrant les trois thématiques du département (informatique, vision et machine learning), en respectant si possible une diversité dans la nature des entreprises (PME, grands groupes, secteur public/parapublic ou associations). Je supervise également la contractualisation des contrats INSA/entreprise en collaboration avec la direction de la recherche et de la valorisation l'INSA. Une participation financière de l'ordre de 10 k€est demandée à chaque client. Cette participation nous permet d'équiper les salles (machines, écran TV, mobilier), mais aussi à participer au salaire du tuteur qualité à hauteur de 50%. Les entreprises ayant participé aux dernières campagnes PIC sont les suivantes :

Campagne 2019 : Actemium, Aerow, Essilor, HAPPY, Mediarithmics, Sopra Steria
Campagne 2018 : Champollion, HAPPY, Saagie, Thalès Optronics, Ville de Paris/RATP
Campagne 2017 : a2ia, Actemium, Aerow, Centre H. Becquerel, Saagie, Thalès Avionics
Campagne 2016 : Cap Gemini, Creative Data, Powow.Box, Unicef 76

(orange : startup, bleu : grand groupe, rose : secteur public ou association, vert : PME)

La recherche des participants se fait essentiellement par deux moyens : le premier est le réseau des anciens étudiants ASI qui ont pour la plupart vécu l'expérience PIC, le second est mon carnet d'adresse issu de relations avec le monde industriel, dans un cadre recherche ou personnel. Ce deuxième levier permet d'instaurer régulièrement des PIC orientés recherche faisant le lien entre mes activités de recherche et pédagogiques. Ce fût par exemple le cas avec les clients *Centre Henri Becquerel* (Conception et apprentissage de réseaux totalement

convolutifs (FCN) pour la segmentation d'images médicales) ou *a2ia* (mise en place d'un benchmark des moteurs de reconnaissance d'écriture du commerce sur des terminaux mobiles). Signalons également un brevet en cours de dépôt avec la société Essilor.

Il est à noter que l'interaction recherche/pédagogie fonctionne dans les deux sens : l'année dernière le client PIC actemium est devenu un partenaire de recherche avec lequel nous avons débuté la thèse CIFRE de Benjamin Deguerre concernant l'utilisation de réseaux profonds pour l'analyse de scènes routières.

En conclusion sur les PIC, ceux-ci constituent de par leur ampleur l'une des pierres angulaires du département, ce qui m'amène à travailler de manière resserrée avec la direction ASI. Les PIC permettent d'avoir des relations avec de nombreux acteurs industriels pouvant déboucher sur des émulations intéressantes. D'un point de vue plus personnel, la direction de l'Unité P3 me conduit à avoir des rapports particuliers avec les étudiants, dépassant les rapport étudiant/enseignant classiques. Bien que particulièrement chronophage (recherche des sujets, contractualisation, gestion des problèmes avec les clients, pilotage des processus, préparation de l'audit AFNOR, etc.), l'activité PIC est passionnante et motivante.

1.2.3 Résumé de mes activités d'enseignement au département ASI

Traitement du signal (3ème année) :

Je suis depuis mon arrivée en ASI responsable de l'EC traitement du signal pour laquelle j'assure les cours et TD (21h CM, 21h TD) . Les notions abordées sont les distributions, la transformée de Fourier, les systèmes linéaires et le filtrage aussi bien pour les signaux analogiques que numériques. Je délègue généralement les TP à des ATER ou à des doctorants en mission enseignement.

Machine Learning (4ème année) :

Je suis responsable de cette EC depuis 2018. Nous y abordons les méthodes de machine learning non linéaires à l'état de l'art : apprentissage Profond, méthodes d'ensemble et SVM. Pour ces deux dernières classes de méthodes, je fais intervenir des collègues spécialisés en recherche dans le domaine. J'essaye de sensibiliser les étudiants à la recherche en proposant des projets personnels consistant à choisir des articles scientifiques, à les résumer puis à mener des expérimentations en lien avec les travaux en question. Ces projets font l'objet de soutenances publiques devant toute la promotion afin de restituer et partager les connaissances.

Automatique (3ème année) :

Cette EC concerne l'automatique traditionnelle et aborde notamment la stabilité, la performance et la correction des systèmes asservis continus. Elle est suivie d'une seconde partie sur les modèles d'état. J'ai été responsable de cette EC (CM+TD) à plusieurs reprises entre 2008 et 2016.

Autres interventions

De manière plus ponctuelle, j'interviens également dans les EC « Documents »(cours d'analyse et reconnaissance d'image de documents) et « interaction homme machine évoluées »(cours sur les modèles de séquences).

1.2.4 Résumé de mes activités d'enseignement hors département ASI

MOOC "Initiez-vous au Deep Learning"

En 2017 l'INSA Rouen Normandie a répondu a un appel à projet pour la création de MOOC en collaboration avec Openclassroom. Avec Gilles Gasso, Stéphane Canu et Romain Hérault nous nous sommes positionnés sur la conception d'un MOOC sur le machine learning. J'ai réalisé les 5 cours de la partie "Découvrez les réseaux de neurones adaptés au traitement de séquences" avec vidéos introductives associées, ainsi qu'une évaluation via un quizz. Ce MOOC est finalisé et devrait sortir courant juillet 2019.

Deep Learning en master 2 Science et Ingéniérie des Données (SID)

En 2018 j'ai conçu avec Romain Hérault cette nouvelle EC du master SID (master co-habilité Université de Rouen/INSA de Rouen). Dans cette EC nous abordons l'apprentissage profond à travers les méthodes statiques (DNN, CNN) et dynamiques (RNN, LSTM, modèles d'attention, etc.), ainsi que leurs applications en CM (8h) et TP (16h).

Algorithmique et programmation structurée en premier cycle INSA (STPI)

En 2015/2016 j'ai assuré des TD/TP de programmation en deuxième année à l'INSA de Rouen : analyse descendante, programmation pascal et encadrement de mini projets.

Machine learning avancé en Mastère Spécialisé Science des Données (INSA)

J'interviens ponctuellement pour un cours sur les modèles récurrents de séquences dans le mastère spécialisé Science des données (CM uniquement).

1.2.5 Synthèse de mes activités administratives liées à l'enseignement

Je suis impliqué dans les tâches courantes et la vie du département Architecture des Systèmes d'Information de l'INSA de Rouen. Je suis ou ai été en charge des responsabilités présentées dans le tableau 1.1.

Table 1.1: Responsabilités assumées au département ASI depuis 2007.

Date	Responsabilité
Depuis 2008	Responsable du Master SID à l'INSA de Rouen (voir ci-dessous)
Depuis 2015	Directeur de l'Unité P3 (Cf. section 1.2.2)
	Responsable du processus « Rechercher, choisir et contractualiser les PIC »
Depuis 2007	Membre élu du conseil de département
	Membre du jury
2007 - 2015	Responsable de la thématique « Ingénierie des données »
	(répartition des enseignements, recrutement ATER/missions enseignement,
	cohérence pédagogique de la thématique)
2008 - 2015	Pilote du processus « mesurer et évaluer la qualité de l'unité P3 »
2015 - 2016	Responsable des relations internationales

Responsabilité à l'INSA du Master SID (anciennement STIM)

Depuis 2008 je suis le responsable INSA du master Science et Ingénierie des Données co-habilité INSA de Rouen/Université de Rouen. Les étudiants du département ASI ont la possibilité de suivre ce master à coloration recherche en parallèle de leur 5ème année (entre 4 et 12 étudiants/an). Je suis en charge des fonctions suivantes pour les étudiants inscrits à l'INSA : sélection des candidats, établissement des contrats d'étude de M2 en fonction de leur parcours ASI, participation aux jurys, membre du conseil de perfectionnement.

1.3 Synthèse des activités de recherche

1.3.1 Contexte des travaux

J'ai obtenu mon poste de Maître de Conférences en 2007 à l'INSA Rouen Normandie, enseignant au département Architecture des systèmes d'information (ASI) et chercheur au sein du Laboratoire d'Informatique, du Traitement de l'Information et des Systèmes (LITIS).

Le Laboratoire d'Informatique, du Traitement de l'Information et des Systèmes (LITIS) est l'unité de recherche haut-normande dans le domaine des Sciences et Technologies de l'Information et de la Communication (STIC). Il implique les trois principaux établissements d'enseignement supérieur de la région : l'Université de Rouen, l'Université du Havre et l'INSA de Rouen.

1.3.2 Parcours de recherche

Après un double diplôme DEA/DESS obtenu en 2003, j'ai effectué un doctorat au laboratoire LITIS à l'Université de Rouen encadré par Laurent Heutte et Thierry Paquet, en convention CIFRE avec la société EMC sur la reconnaissance d'écriture manuscrite. J'y ai abordé des problématiques d'extraction d'information dans des documents manuscrits faiblement contraints de type courriers entrants. Pour cela, nous avons proposé des modèles de lignes de texte combinant un modèle de reconnaissance classique décrivant l'information à extraire, et un modèle surfacique de l'information décrivant l'information non pertinente. Durant ces travaux, j'ai étudié et implémenté des réseaux de neurones et les ai appliqué à la reconnaissance d'écriture manuscrite. Cet intérêt pour les réseaux de neurones ne s'est jamais éteint depuis.

Depuis mon arrivée à l'INSA de Rouen en 2007, j'ai maintenu mon activité de recherche autour de la reconnaissance d'écriture manuscrite, notamment à travers la thèse CIFRE de Simon Thomas faisant suite à mon doctorat, puis à l'occasion des thèses de Gautier Bideault, Luc Mioulet et Bruno Stuner. En parallèle, j'ai continué à m'intéresser aux réseaux de neurones que j'avais déjà étudiés pendant ma thèse. Dès 2008, nous avons développé nos premières architectures "profondes" basées sur des auto-encodeurs, qui paraîtraient aujourd'hui assez peu profondes puisque limitées à quelques couches. Nous avons assez naturellement appliqué ces modèles à la reconnaissance d'écriture, mais aussi à l'imagerie médicale, à l'analyse de gestes ou encore plus récemment à l'analyse de scènes routières. Notre premier article sur l'apprentissage profond date ainsi de 2009 [44], quelques années avant l'engouement de la communauté scientifique pour l'apprentissage profond.

1.3.3 Encadrement doctoral

Depuis l'obtention de mon poste de maître de conférences, j'ai co-encadré 7 doctorants dont 6 ont soutenu et 1 est en cours. La somme de mes encadrements doctoraux s'élève à 310%, dont 260% sur les thèses soutenues et 50% sur la thèse en cours. Ces encadrements sont détaillés ci-après par ordre chronologique inverse (les rapporteurs sont soulignés parmi les membres du jury).

Thèse de Benjamin Deguerre [débutée en avril 2018, en cours]

- Sujet : « Apprentissage profond pour la détection et le tracking dans des flux vidéo compressés »
- Thèse dirigée par Gilles Gasso (50%) et co-encadrée par Clément Chatelain (50%)
- Financement : Convention CIFRE avec la société Actemium
- Publications : 1 conf. internationale [18] sur la détection d'objets dans des images compressées.

Thèse de Bruno Stuner [01/2015 – 06/2018]

- Sujet : « Cohorte de Réseaux de Neurones Récurrents »
- Thèse dirigée par Thierry Paquet (50%) et co-encadrée par Clément Chatelain (50%)
- Jury hors encadrants : C. Wolf, L. Likforman, F. D'Alché-Buc, N. Vincent, É. Anquetil
- Publications : 3 conf. internationales [23, 27, 26], 1 revue internationale soumise.
- Poste actuel : Ingénieur R&D chez Solystic (Paris)

Thèse de Soufiane Belharbi [09/2014 – 07/2018]

- Financement : Bourse INSA Rouen Normandie
- Sujet : « Régularisation des réseaux de neurones via l'apprentissage des représentations »
- Thèse dirigée par Sébastien Adam (40%), co-encadrée par Romain Hérault (30%) et Clément Chatelain (30%)
- Jury hors encadrants : É.Fromont, T. Artières, J. Lee, D. Picard
- Publications : 2 revues internationales [3, 6], 2 conf. Internationales [28, 25], 2 conf nationales [55, 57]
- Poste actuel : Post Doc ÉTS/Univ. McGill (Montréal, Canada)

Thèse de Gautier Bideault [11/2011 – 12/2015]

- Sujet : « Détection de mots clés et d'expressions régulières en vue de la reconnaissance d'entités nommées dans des documents manuscrits »

- Thèse dirigée par Thierry Paquet (50%) et co-encadrée par Clément Chatelain (50%)
- Financement : Contrat de recherche lié à la société Itesoft
- Jury hors encadrants : N. Vincent, L. Likforman, J.M. Ogier, V. Poulain d'Ancy
- Publications : 1 chapitre de livre [17], 6 conf. internationales [31, 29, 30, 34, 33, 37]
- Poste actuel : Ingénieur R&D chez Sanofi Pasteur (Val-de-Reuil)

Thèse de Luc Mioulet [02/2012 – 07/2015]

- Sujet : « Réseaux de neurones récurrents pour la reconnaissance d'écriture »
- Thèse dirigée par Thierry Paquet (50%) et co-encadrée par Clément Chatelain (50%)
- Financement : Convention CIFRE avec Airbus defense & space
- Jury hors encadrants : C. Viard-Gaudin, L. Prevost, C. Garcia, B. Coüasnon
- Publications : 1 chapitre de livre [17], 8 conf. internationales [32, 35, 31, 29, 30, 34, 33, 37]
- Poste actuel : Ingénieur R&D chez Cubeyou (Milan, Italie)

Thèse de Selma Belgacem [03/2010 – 06/2014]

- Sujet : « Caractérisation et Reconnaissance de Gestes dans des vidéos à l'aide de Modèles Markoviens »
- Thèse dirigée par Thierry Paquet (50%) et co-encadrée par Clément Chatelain (50%)
- Financement : Bourse gouvernement tunisien
- Jury hors encadrants : P. Dalle, É. Anquetil, S. Ruan, N. Vincent
- Publications : 1 revue internationale [5], 1 chapitre de livre [16], 1 conf. internationale [41].
- Poste actuel : Maître de conférences (Kasserine, Tunisie)

Thèse de Simon Thomas [04/2008 – 07/2012]

- Sujet : « Extraction d'information dans des couriers manuscrits faiblement contraints »
- Thèse dirigée par Laurent Heutte (40%), co-encadrée par Thierry Paquet (30%) et Clément Chatelain (30%)
- Financement : Convention CIFRE avec la société EMC-Captiva
- Jury hors encadrants : C. Viard-Gaudin,T. Artières, F. Jurie
- Publications : 1 revue internationale [10], 1 revue nationale [14], 2 conf. internationales [43, 42], 2 conf. nationales [62, 61]
- Poste actuel : Ingénieur R&D chez Nagravision (Genève, Suisse)

1.3.4 Encadrements de post-doctorants et stagiaires

Co-encadrements de post-doctorants

— **Yann soullard (2017-2019)** sur l'ANR labcom INKS puis sur le projet Asturias : "Analyse d'images de documents avec BLSTM/CTC et FCN".
— **Simon Bernard (2015)** sur l'ANR LeMOn : "Front ROC multiclasse".
— **Thotreingam Kasar (2015)** sur le projet Maurdor : "Détection de tableaux dans des images de documents".
— **Julien Delporte (2014)** sur le projet Nareca : "Annotation automatique d'éléments dialogiques".
— **Yousri Kessentini (2013)** sur le projet DoD avec la société Itesoft : "extraction d'expressions régulières dans des documents manuscrits".

Co-encadrements de stagiaires

— **Denis Coquenet (2019)** : "Reconnaissance d'écriture par réseaux profonds totalement convolutifs".
— **Guillaume Renton (2017)** : "Segmentation de documents en lignes par réseaux totalement convolutifs". Le travail de ce stage a été publié dans un journal international [4] et dans un workshop international [22].
— **Bruno Stuner (2014)** : "Champs Conditionnels Aléatoires pour la reconnaissance d'écriture".
— **Julien Lerouge (2013)** : "Combinaison d'apprentissage statistique et d'un recalage pour la segmentation de coupes de scanner". Ce travail a donné lieu à une publication dans une revue internationale [9].
— **Élodie Dellier (2013)** : "Détection de ROI à l'aide de Deep Belief Network (DBN)".
— **Soufiane Belharbi (2013)** : "Reconnaissance de l'écriture avec des CRF".
— **Rami Kessentini (2012)** : "Discrimination d'écritures arabe/latine, imprimée/manuscrite".
— **Yannick Ouffella (2008)** : "Sélection de modèles multiobjectifs". Ce travail a été présenté dans une conférence nationale [63]
— **Quentin Suire (2007)** : "Champs Conditionnels Aléatoires pour le traitement d'images de documents".

1.3.5 Collaborations académiques

Robert Sabourin (École de Technologie Supérieure, Montréal, Canada)

En octobre/novembre 2012 j'ai effectué une visite de recherche de 15 jours à l'*École de Technologie Supérieure* de Montréal chez le professeur R. Sabourin, sur des problématiques d'apprentissage multiobjectif. J'ai eu l'occasion de donner un séminaire « Sélection de modèle dans des environnements incertains [12] ». Cette collaboration s'est poursuivie dans le cadre du post doctorat de Simon Bernard et a donné lieu à un article dans la revue *Pattern Recognition* en 2016 [8] et un article dans la conférence nationale *SFC 2014* [59]. Cette collaboration est toujours active puisque R. Sabourin est régulièrement invité au LITIS depuis.

Utpal Garain (Indian Statistical Institute, Calcutta, Inde)

Depuis 2013 je travaille en collaboration avec U. Garain, qui a effectué une visite de recherche au LITIS de 10 jours en septembre 2014, puis de 3 semaines en août/septembre 2015. Cette collaboration s'est matérialisée par deux articles dans la conférence *ICDAR 2015* sur la reconnaissance d'écriture Bengali [32] et sur l'identification de langue dans des documents manuscrits [35], en collaboration avec mon doctorant Luc Mioulet. Dans ces travaux nous avons notamment proposé les premiers résultats de reconnaissance d'écriture sur l'Assamese (dialecte indien comportant un très grand nombre de classe de caractères) à l'aide de réseaux BLSTM, et produit la première base annotée publique de mots manuscrits Bengali.

Grégoire Mesnil (Université de Montréal, Canada)

En 2012 j'ai travaillé en collaboration avec Grégoire Mesnil de l'équipe de machine learning de Yoshua Bengio. Cette visite a donné lieu à un séjour de 4 mois de Grégoire Mesnil au LITIS dans le cadre de l'ANR LeMOn à partir de mars 2015.

Maxime Martineau & Romain Raveaux (Equipe RFAI, LI Tours)

En 2017 nous avons collaboré avec Romain Raveaux au sujet de l'élaboration d'une méthode d'identification d'insectes sur des images en environnement contrôlé. Nous avons pour cela utilisé des architectures convolutives et un transfert d'apprentissage. Cette collaboration a donné lieu à un article dans une conférence internationale [20].

Romain Modzelewski & Fabrice Jardin (équipe Quantif, LITIS Rouen)

Bien que locale, je mentionne cette collaboration avec l'équipe de médecine nucléaire du centre Henri Becquerel car elle constitue une part importante de mon activité de recherche, de manière continue depuis ma prise de fonction en 2007. Cette collaboration vise à concevoir des méthodes de machine learning pour l'aide à la décision dans le contexte médical de la lutte contre le cancer, en utilisant notamment l'apprentissage profond. Les bases d'images y sont assez limitées, de résolution moyenne ou basse, et présentent de très fortes variabilités. La collaboration a naturellement débuté par une longue phase de discussions avec les radiologues au sujet des annotations des images pour constituer des bases exploitables. Nous avons ensuite abordé les problématiques de segmentation sémantique sur des coupes de scanner CT, l'étiquetage de voxels dans des volumes 3D, ou encore la détection de coupe dans des scanners complets. Cette collaboration a donné lieu à plusieurs articles dans des revues internationales [9, 5, 2, 1], plusieurs conférences [21, 53, 52] et à l'obtention du prix Unicancer de l'innovation 2017 pour le projet "Bodycomp.AI" (lien).

1.3.6 Participation à des projets de recherche académiques et industriels

Participation à des projets de recherche académiques

- **[Asturias, 2018-2021, 300 k€]**
  Projet financé par la région : « Analyse STructURelle et Indexation sémantique d'ArticleS de presse »porté par Pierrick Tranouez. Implication à hauteur de 10% de mon temps de recherche.
- **[ANR HBDEX, 2017-2021, 661 k€]**
  « Extraction de Big Data Historiques pour les Humanités Numériques : application aux données financières »porté par Pierre Cyrille Hautcœur. Implication à hauteur de 10% de mon temps de recherche.
- **[ANR Deep in France, 2017-2020, 811 k€]**
  « Réseaux de neurones profonds pour l'apprentissage »porté par Stéphane Canu. Implication à hauteur de 15% de mon temps de recherche.
- **[ANR LabCom INKS, 2016-2018, 300 k€]**
  « LabCom INtelligent notebooKS »porté par Thierry Paquet. Implication à hauteur de 20% de mon temps de recherche. Dans ce contexte, co encadrement du postdoc Yann Soulard et du stagiaire Guillaume Renton.
- **[ANR NARECA, 2013-2016, 629 k€]**
  « Narrative Embodied Conversational Agent »porté par Alexandre PAUCHET. Implication à hauteur de 22% de mon temps de recherche. Dans ce contexte, co-encadrement d'un post doc (Julien Delporte) en 2014.
- **[ANR JCJC LeMOn, 2012 - 2015, 140 k€]**
  « Learning with multiple objective optimization »porté par Sébastien Adam. Participation à hauteur de 50% de mon temps de recherche, responsable d'un Work Package, co-encadrement à 50% d'un post-doctorant sur un an (Simon Bernard).
- **[ANR ASAP, 2010-2012, 151 k€]**
  « Apprentissage Statistique par des Architectures Profondes »porté par Alain Rakotomamonjy. Implication à hauteur de 10% de mon temps de recherche.

Participation à des projets de recherche industriels

- **[DGA, projet "SAPhIRS", 2017-2020, 170 k€]**
  Le projet SAPhIRS financé par la DGA regroupe le LITIS et deux entreprises (Saagie et Airbus) pour l'analyse de la propagation d'information dans les réseaux sociaux. Dans ce contexte, je co-encadre un postdoc avec Romain Hérault et Benoit Gauzère.
- **[Solystic, 2012-2018, 22 k€]**
  Dans ce partenariat visant à développer les réseaux récurrents pour la reconnaissance d'écriture, le co-encadrement de deux stagiaires de master (Bruno Stuner et Élodie Dellier) a été poursuivi par une convention CIFRE pour la thèse de Bruno Stuner.

— **[Itesoft, projet "DocOnDemand", 2011-2015, 360 k€]**
Dans le cadre de son projet DocOnDemand, nous avons mené une étude pour le compte de la société Itesoft, visant à extraire des entités nommées dans des documents manuscrits. La collaboration s'est matérialisée par le stage de Quentin Suire, la convention CIFRE du doctorant Gautier Bideaut, et le recrutement de Yousri Kessentini en post-doctorat, que j'ai tous co-encadrés.

— **[Airbus Defense&Space, projet "MAURDOR", 2011-2014, 360 k€]**
En collaboration avec Airbus, nous avons répondu à un appel d'offre de la DGA concernant un Programme d'Étude Amont (PEA) que nous avons remporté. Nous avons ainsi constitué le consortium "Maurdor" piloté par Airbus et le LITIS, regroupant plusieurs partenaires académiques et privés pour développer une plateforme de reconnaissance d'images de document à l'état de l'art. Cette collaboration nous a permis de mettre en place la bourse CIFRE de Luc Mioulet et de recruter le post doctorant Thotreingam Kasar.

— **[EMC, 2007-2012, 75 k€]**
Initiée avec ma thèse en convention CIFRE, la collaboration avec la société EMC s'est poursuivie par une seconde convention CIFRE avec Simon Thomas, que j'ai co-encadré. Il s'agissait de généraliser mes travaux de thèse pour concevoir des modèles génériques d'extraction d'information dans des couriers manuscrits pour localiser des séquences alphanumériques données (voir section 2.3.3.2).

1.3.7 Activités de relecture et d'évaluation

Participation à un jury de thèse externe

J'ai été invité en tant qu'examinateur à participer au Jury d'Adeline Granet pour sa soutenance de thèse le 12 décembre 2018 au LS2N de Nantes. L'intitulé de la thèse est « Extraction d'informations dans des collections historiques manuscrites », soutenue devant le jury composé d'Antoine Doucet, Frédéric Béchet, Solène Quiniou, Harold Mouchère, Christian Viard-Gaudin, Emmanuel Morin et moi même.

Relecture d'articles scientifiques

— Relecteur régulier pour les revues *International Journal on Document Analysis and recognition* et *Pattern recognition.*

— Relecteur occasionnel pour les revues internationales *Pattern Analysis and Machine Intelligence*, *Pattern Recognition Letters*, *Neurocomputing*, *Engineering Applications of Artificial Intelligence*, *Journal of Machine Learning Research*, *Traitement du Signal*, *Documents Numériques.*

— Relecteur régulier pour les conférences internationales *ICDAR, ICFHR, ICPR, DAS, ICIC*, et la conférence nationale *SIFED* (anciennement *CIFED*).

Relecture de projets scientifiques

— Relecture de projets ANR (2017, 2018)
— Expert auprès de l'ANRT pour la relecture de demandes de subventions CIFRE (5 dossiers en 2017 et 2018)

Comités d'organisation/de programme de conférences, comités de sélection

— Membre des comités de programme des conférences internationales ICDAR 2009 et ICDAR 2013.
— Membre des comités de programme des conférences nationales CIFED 2014 et 2016.
— Membre des comités d'organisation des conférences CIFED 2008, RFIA 2014, CAp 2018.
— Membre d'un comité de sélection pour le recrutement d'un MCF 61ème section à l'INSA Rouen Normandie en 2015.
— Membre du comité de suivi de thèse de Robien Condat, débutée en 2019 au LITIS.

Affiliation à des groupes de recherche

Je suis membre du Groupe de Recherche en Communication Écrite (GRCE), de l'Association Française de Reconnaissance des Formes (AFRIF), de la Société Savante Francophone d'Apprentissage Machine (SSFAM) et de l'International Association for Pattern Recognition (IAPR).

1.4 Bibliographie personnelle

Revues internationales (13)

[1] A. Amyar, S. Ruan, I. Gardin, C. Chatelain, P. Decazes et R. Modzelewski. "3-D RPET-NET : Development of a 3-D PET Imaging Convolutional Neural Network for Radiomics Analysis and Outcome Prediction". In : *IEEE Transactions on Radiation and Plasma Medical Sciences* 3.2 (2019), p. 225-231 (cf. p. 13).

[2] A. Amyar, S. Ruan, I. Gardin, R. Herault, C. Chatelain, P. Decazes et R. 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 (cf. p. 13).

[3] S. Belharbi, R. Hérault, C. Chatelain et S. Adam. "Deep Neural Networks Regularization for Structured Output Prediction". In : *Neurocomputing* 281 (2018), p. 169-177 (cf. p. 10).

[4] G. Renton, Y. Soullard, C. Chatelain, S. Adam, C. Kermorvant et T. Paquet. "Fully Convolutional Network with dilated convolutions for Handwritten text line segmentation". In : *International Journal on Document Analysis and Recognition (IJDAR)* (2018) (cf. p. 12).

[5] S. Belgacem, C. Chatelain et T. Paquet. "Gesture sequence recognition with one shot learned CRF/HMM hybrid model". In : *Image and Vision Computing* 61 (2017), p. 12-21 (cf. p. 11, 13).

[6] S. Belharbi, C. Chatelain, R. Hérault, S. Adam, S. Thureau, M. Chastan et R. Modzelewski. "Spotting L3 slice in CT scans using deep convolutional network and transfer learning". In : *Computers in Biology and Medicine* 87 (2017), p. 95-103 (cf. p. 10).

[7] P. Barlas, D. Hebert, C. Chatelain, S. Adam et T. Paquet. "Language identification in document images". In : *Journal of Imaging Science and Technology* 60.1 (2016), p. 1-16.

[8] S. Bernard, C. Chatelain, S. Adam et R. Sabourin. "The Multiclass ROC Front method for cost-sensitive classification". In : *Pattern Recognition* 52 (2016), p. 46-60 (cf. p. 12).

[9] J. Lerouge, R. Herault, C. Chatelain, F. Jardin et R. Modzelewski. "IODA : An Input Output Deep Architecture for image labeling". In : *Pattern Recognition* 48.9 (2015), p. 2847-2858 (cf. p. 12, 13).

[10] S. Thomas, C. Chatelain, L. Heutte, T. Paquet et Y. Kessentini. "A Deep HMM model for multiple keywords spotting in handwritten documents". In : *Pattern Analysis and Applications* 18.4 (2015), p. 1003-1015 (cf. p. 11).

[11] T. Paquet, L. Heutte, G. Koch et C. Chatelain. "A Categorization System for Handwritten Documents". In : *International Journal on Document Analysis and Recognition (IJDAR)* 15.4 (2012), p. 315-330.

[12] C. Chatelain, S. Adam, Y. Lecourtier, L. Heutte et T. Paquet. "A multi-model selection framework for unknown and/or evolutive misclassification cost problems". In : *Pattern Recognition* 43.3 (2010), p. 815-823 (cf. p. 12).

[13] C. Chatelain, S. Adam, Y. Lecourtier, L. Heutte et T. Paquet. "Non-cost-sensitive SVM training using Multiple Model Selection". In : *Journal of Circuits, Systems and Computers* 19.1 (2010), p. 231-242.

Revues nationales (2)

[14] S. Thomas, C. Chatelain, L. Heutte et T. Paquet. "Un modèle neuro markovien profond pour l'extraction de séquences dans des documents manuscrits". In : *Documents Numériques* 16.2 (2013), p. 49-69 (cf. p. 11).

[15] C. Chatelain, G. Koch, L. Heutte et T. Paquet. "Une methode dirigee par la syntaxe pour l'extraction de champs numeriques dans les courriers entrants". In : *Traitement du Signal* 23.2 (2006), p. 179-198.

Participation à des ouvrages collectifs (2)

[16] S. Belgacem, C.Chatelain et T. Paquet. "A Hybrid CRF/HMM for One-Shot Gesture Learning". In : *Adaptive Biometric Systems*. Sous la dir. d'A. Rattani, F. Roli et E. Granger. Advances in Computer Vision and Pattern Recognition. Springer International Publishing, 2015, p. 51-72. isbn : 978-3-319-24863-9 (cf. p. 11).

[17] G. Bideault, L. Mioulet, C. Chatelain et T. Paquet. "Using BLSTM for Spotting Regular Expressions in Handwritten Documents". In : *Pattern Recognition : Applications and Methods : 4th International Conference, ICPRAM 2015, Lisbon, Portugal, January 10-12, 2015, Revised Selected Papers*. Sous la dir. d'A. Fred, M. De Marsico et M. Figueiredo. Cham : Springer International Publishing, 2015, p. 143-157. isbn : 978-3-319-27677-9 (cf. p. 11).

Conférences internationales (33)

[18] B. Deguerre, C. Chatelain et G. Gasso. "Fast object detection in compressed JPEG Images". In : *IEEE Intelligent Transportation Systems Conference (ITSC)*. 2019 (cf. p. 10).

[19] Y. Soullard, W. Swaileh, C. Chatelain, P. Tranouez et T. Paquet. "Improving text recognition using optical and language model writer adaptation". In : *ICDAR*. 2019.

[20] M.Martineau, R.Raveaux, C. Chatelain, D.Conte et G.Venturini. "Effective Training of Convolutional Neural Networks for Insect Image Recognition". In : *Advanced Concepts for Intelligent Vision Systems*. Springer International Publishing, 2018, p. 426-437 (cf. p. 13).

[21] R. Modzelewski, C. Chatelain, R. Herault, F. Jardin, S. Thureau et P. Vera. "BodyComp.ai : Toward automatic L3 computed tomography bodycomposition quantification in clinical practice ?" In : *International Conference on Frailty and Sarcopenia Research*. 2018 (cf. p. 13).

[22] G. Renton, C. Chatelain, S. Adam, C. Kermorvant et T. Paquet. "Handwritten text line segmentation using Fully Convolutional Network". In : *ICDAR Workshop on Machine Learning*. 2017 (cf. p. 12).

[23] B. Stuner, C. Chatelain et T. Paquet. "Self-Training of BLSTM with Lexicon Verification For Handwriting Recognition". In : *ICDAR*. Kyoto, Japan, 2017 (cf. p. 10).

[24] P. Barlas, D. Hebert, C. Chatelain, S. Adam et T. Paquet. "Language Identification in Document Images". In : *Document Recognition and Retrieval XXIII*. San Francisco, USA, 2016, p. 1-16.

[25] S. Belharbi, R. Hérault, C. Chatelain et S. Adam. "Deep multi-task learning with evolving weights". In : *European Symposium on Artificial Neural Networks (ESANN)*. Bruges, Belgium, 2016 (cf. p. 10).

[26] B. Stuner, C. Chatelain et T. Paquet. "A Lexicon Verification Strategy in a BLSTM Cascade Framework". In : *International Conference on Frontiers in Handwriting Recognition (ICFHR)*. Shenzhen, China, 2016, p. 234-239 (cf. p. 10).

[27] B. Stuner, C. Chatelain et T. Paquet. "Cascading BLSTM networks for handwritten word recognition". In : *International Conference on Pattern Recognition (ICPR)*. Cancún, Mexico, 2016, p. 3416-3421 (cf. p. 10).

[28] S. Belharbi, C. Chatelain, R. Herault et S. Adam. "Learning Structured Output Dependencies Using Deep Neural Networks". In : *Deep Learning Workshop, ICML*. Lille, France, 2015 (cf. p. 10).

[29] G. Bideault, L. Mioulet, C. Chatelain et T. Paquet. "A Hybrid BLSTM-HMM for spotting Regular Expressions". In : *ICPRAM*. Lisbon, Portugal, 2015, p. 5-12 (cf. p. 11).

[30] G. Bideault, L. Mioulet, C. Chatelain et T. Paquet. "Benchmarking discriminative approaches for word spotting in handwritten documents". In : *ICDAR*. Washington, USA, 2015, p. 201-205 (cf. p. 11).

[31] G. Bideault, L. Mioulet, C. Chatelain et T. Paquet. "Spotting handwritten words and REGEX using a two stage BLSTM-HMM architecture". In : *Document Recognition and Retrieval*. T. 9402. San Francisco, USA, 2015, 94020G-11 (cf. p. 11).

[32] U. Garain, L. Mioulet, B. Chaudhuri, C. Chatelain et T. Paquet. "Unconstrained Bengali Handwriting Recognition with Recurrent Models". In : *ICDAR*. Washington, USA, 2015, p. 1056-1060 (cf. p. 11, 13).

[33] L. Mioulet, G. Bideault, C. Chatelain et T. Paquet. "BLSTM-CTC Combination Strategies for Off-line Handwriting Recognition". In : *ICPRAM*. Lisbon, Portugal, 2015, p. 173-180 (cf. p. 11).

[34] L. Mioulet, G. Bideault, C. Chatelain et T. Paquet. "Exploring multiple feature combination strategies with a recurrent neural network architecture for off-line handwriting recognition". In : *Document Recognition and Retrieval*. San Francisco, USA, 2015, 94020F (cf. p. 11).

[35] L. Mioulet, U. Garain, C. Chatelain, T. Paquet et P. Barlas. "Language Identification from Handwritten Documents". In : *ICDAR*. Washington, USA, 2015, p. 676-680 (cf. p. 11, 13).

[36] P. Barlas, S. Adam, C. Chatelain et T. Paquet. "A typed and handwritten text block segmentation system for heterogeneous and complex documents". In : *Document Analysis Systems, Tours, France*. 2014, p. 46-50.

[37] G. Bideault, L. Mioulet, C. Chatelain et T. Paquet. "A hybrid CRF/HMM approach for handwriting recognition". In : *International Conference on Image Analysis and Recognition (ICIAR), Vilamoura, Portugal*. T. 1. Lecture Notes in Computer Science. 2014, p. 403-410 (cf. p. 11).

[38] D. Hébert, P. Barlas, C. Chatelain, S. Adam et T. Paquet. "Writing type and language identification in heterogeneous and complex documents". In : *International Conference on Frontiers in Handwriting Recognition (ICFHR), Crete, Greece*. 2014, p. 411-416.

[39] T. Kasar, P. Barlas, S. Adam, C. Chatelain et T. Paquet. "Learning to Detect Tables in Scanned Document Images using Line Information". In : *ICDAR, Washington DC, USA*. 2013, p. 1185-1189.

[40] Y. Kessentini, C. Chatelain et T. Paquet. "Word Spotting and Regular Expression Detection in Handwritten Documents". In : *ICDAR, Washington DC, USA*. 2013, p. 516-520.

[41] S. Belgacem, C. Chatelain, A. Ben-Hamadou et T. Paquet. "Hand Tracking using Optical-Flow embedded Particle Filter in sign language scenes". In : *Computer Vision and Graphics*. T. 7594. Lecture Notes in Computer Science. 2012, p. 288-295 (cf. p. 11).

[42] S. Thomas, C. Chatelain, L. Heutte et T. Paquet. "Alpha-numerical sequences extraction in handwritten documents". In : *International Conference on Frontiers in Handwriting Recognition (ICFHR), Kolkata, India*. 2010, p. 232-237 (cf. p. 11).

[43] S. Thomas, C. Chatelain, L. Heutte et T. Paquet. "An Information Extraction model for unconstrained handwritten documents". In : *International Conference on Pattern Recognition (ICPR), Istanbul, Turkey*. 2010, p. 4 (cf. p. 11).

[44] B. Labbe, R. Herault et C. Chatelain. "Learning Deep Neural Networks for High Dimensional Output Problems". In : *IEEE International Conference on Machine Learning and Applications (ICMLA), Miami, USA*. 2009, p. 63-68 (cf. p. 9).

[45] C. Chatelain, L. Heutte et T. Paquet. "Recognition-based vs syntax-directed models for numerical field extraction in handwritten documents". In : *International Conference on Frontiers in Handwriting Recognition (ICFHR), Montreal, Canada*. 2008, p. 159-164.

[46] E. Barbu, C. Chatelain, S. Adam, P. Heroux et E. Trupin. "A simple one class classifier with rejection strategy : application to symbol classification". In : *Seventh*

IAPR International Workshop on Graphics Recognition - GREC 2007, Curitiba, Brazil. 2007.

[47] C. Chatelain, S. Adam, Y. Lecourtier, L. Heutte et T. Paquet. "Multi-Objective Optimization for SVM Model Selection". In : *ICDAR, Curitiba, Brazil*. T. 1. 2007, p. 427-431.

[48] C. Chatelain, L. Heutte et T. Paquet. "A two-stage outlier rejection strategy for numerical field extraction in handwritten documents". In : *International Conference on Pattern Recognition (ICPR), Hong Kong, China*. T. 3. 2006, p. 224-227.

[49] C. Chatelain, L. Heutte et T. Paquet. "Discrimination between digits and outliers in handwritten documents applied to the extraction of numerical fields". In : *International Workshop on Frontiers in Handwriting Recognition (IWFHR), La Baule, France*. 2006, p. 475-480.

[50] C. Chatelain, L. Heutte et T. Paquet. "Segmentation-driven recognition applied to numerical field extraction from handwritten incoming mail documents". In : *Document Analysis System, Nelson, New Zealand, LNCS 3872, Springer*. 2006, p. 564-575.

[51] C. Chatelain, L. Heutte et T. Paquet. "A syntax-directed method for numerical field extraction using classifier combination". In : *International Workshop on Frontiers in Handwriting Recognition (IWFHR), Tokyo, Japan*. 2004, p. 93-98.

Conférences nationales (14)

[52] A. Amyar, S. Ruan, I. Gardin, C. Chatelain, R. Herault, P. Decazes et R. Modzelewski. "Étude de la radiomique en imagerie TEP oncologique basée sur réseau de neurones 3D". In : *CAp, Rouen, France*. 2018 (cf. p. 13).

[53] S. Belharbi, C. Chatelain, R. Hérault, S. Adam, S. Thureau, M. Chastan et R. Modzelewski. "Spotting L3 slice in CT scans using deep convolutional network and transfer learning". In : *CAp, Rouen, France*. 2018 (cf. p. 13).

[54] S. Belharbi, R. Hérault, C. Chatelain et S. Adam. "Deep Neural Networks Regularization for Structured Output Prediction". In : *CAp, Rouen, France*. 2018.

[55] S. Belharbi, R.Herault, C. Chatelain et S. Adam. "Pondération dynamique dans un cadre multi-tâche pour réseaux de neurones profonds". In : *RFIA, Clermont-Ferrand, France*. 2016 (cf. p. 10).

[56] B. Stuner, C. Chatelain et T. Paquet. "Cascade de réseaux BLSTM vérifiée par le lexique pour la reconnaissance d'écriture". In : *RFIA, Clermont-Ferrand, France*. 2016.

[57] S. Belharbi, C. Chatelain, R.Herault et S. Adam. "A Unified Neural Based Model for Structured Output Problems". In : *Conférence d'apprentissage (CAP), Lille, France*. 2015 (cf. p. 10).

[58] P. Barlas, C. Chatelain, S. Adam et T. Paquet. "Détection et segmentation des blocs de texte manuscrits et imprimés dans des documents complexes". In : *CIFED, Nancy, France*. 2014.

[59] S. Bernard, C. Chatelain, S. Adam et R. Sabourin. "Apprentissage multiclasse en environnement incertain". In : *Societe Francophone de Classification, Rabat, Maroc*. 2014 (cf. p. 12).

[60] T. Kasar, P. Barlas, S. Adam, C. Chatelain et T. Paquet. "Détection de tableaux dans des documents complexes". In : *CIFED, Nancy, France*. 2014.

[61] S. Thomas, C. Chatelain, L. Heutte et T. Paquet. "Combinaison architecture profonde/HMM pour l'extraction de sequences dans des documents manuscrits". In : *CIFED, Bordeaux, France*. 2012 (cf. p. 11).

[62] S. Thomas, C. Chatelain, L. Heutte et T. Paquet. "Un systeme generique d'extraction d'information dans des documents manuscrits non-contraints". In : *CIFED, Sousse, Tunisia*. 2010 (cf. p. 11).

[63] C. Chatelain, S. Adam, Y. Lecourtier, L. Heutte, T. Paquet et Y.Oufella. "Optimisation multi-objectif pour la selection de modeles SVM". In : *RFIA, Amiens, France*. 2008 (cf. p. 12).

[64] C. Chatelain, L. Heutte et T. Paquet. "Discrimination chiffre/rejet pour l'extraction de champs numeriques dans des documents manuscrits". In : *CIFED, Fribourg, Suisse*. 2006, p. 55-60.

[65] C. Chatelain, L. Heutte et T. Paquet. "Reconnaissance de champs numeriques dans des documents manuscrits libres". In : *RFIA, Tours, France*. 2006, 45-63 (Actes sur CDROM).

[66] C. Chatelain, L. Heutte et T. Paquet. "Une methode d'extraction automatique de champs numeriques dans des documents manuscrits". In : *EGC, Lille, France*. T. 1. 2006, p. 23-34.

Thèse

[67] C. Chatelain. "Numerical sequences extraction in handwritten documents". Thèse de doct. Université de Rouen, déc. 2006.

Chapitre 2

Contributions

2.1 Introduction

À la fin du deuxième millénaire, « l'intelligence artificielle »(IA) relevait encore de la science fiction. Il existait bien des applications qui laissaient entrevoir le potentiel de l'IA telles que la reconnaissance d'écriture ou de la parole, mais la plupart des applications industrielles étaient limitées à l'automatisation de tâches dans des environnements très contrôlés. La situation a radicalement changée depuis une dizaine d'années, avec l'explosion des applications grands public telles que la conduite autonome de véhicules ou les joueurs de jeu de Go artificiels, excitant particulièrement la sphère médiatique. Nous pourrions discuter longuement du terme intelligence artificielle qui est incontestablement prétentieux (il y a aujourd'hui assez peu d'intelligence dans l'IA [1]), mais qui a le mérite de bien délimiter l'ensemble des méthodes qui se situent entre les méthodes à base de règles et l'intelligence humaine. Pour des raisons pratiques, nous clorons le débat en admettant ce terme communément employé dans la vie quotidienne.

Paradoxalement, les bases des méthodes employées ont été fondés il y a plusieurs décennies. Les raisons de cette révolution ne viennent donc pas tant de la science fondamentale que de plusieurs autres facteurs relevant davantage du monde socio-économique. On peut notamment citer l'explosion de la quantité de données numériques, l'émergence des géants du web particulièrement actifs dans le développement de l'IA, ou encore l'augmentation des puissances de calcul à disposition des entreprises et des particuliers.

Cœur de l'intelligence artificielle, l'apprentissage statistique ou Machine Learning est une discipline scientifique récente visant à faire digérer des données par une machine, dans le but d'effectuer une prédiction dans le monde réel. Ce domaine regroupe les statistiques et l'informatique pour les appliquer aux grandes bases de données. L'apprentissage à partir d'exemples doit permettre à la machine de s'adapter à de nouvelles situations, c'est à dire d'extraire automatiquement des règles les plus universelles possibles permettant d'effectuer les

1. L'intelligence peut se définir selon le Larousse comme l' « Ensemble des fonctions mentales ayant pour objet la connaissance conceptuelle et rationnelle ».

bonnes prédictions sur des données jamais rencontrées. Cette notion centrale en apprentissage machine est la généralisation.

Basée sur cette notion de généralisation, la théorie de l'apprentissage statistique [213] a favorisé l'émergence de modèles évitant l'écueil du sur-apprentissage (apprentissage par cœur). Elle a donné naissance aux Machines à Vecteurs de Support (SVM), qui ont régné pendant une dizaine d'années sur le machine learning, grâce à des fondements théoriques forts et des schémas d'optimisation convexes efficaces.

Depuis les travaux de G. Hinton [163] en 2006, on assiste à un véritable raz de marée des réseaux de neurones dans le machine learning, sous la dénomination de « deep learning »ou « apprentissage profond »[64]. Cette classe de méthodes neuronales, pourtant ancienne, possède une grande modularité, s'adaptant ainsi plus naturellement que les SVM à la nature dynamique des signaux réels. Ils disposent également d'algorithmes d'apprentissage efficaces, permettant d'optimiser des modèles extrêmement complexes. L'apprentissage profond a permis la mise en production d'applications que l'on n'imaginait pas réalisables il y a seulement quelques années, telles que la traduction automatique, la génération automatique de textes ou encore le traitement de vidéos complexes en temps réel.

À l'origine de la révolution de l'intelligence artificielle, les architectures profondes performantes comportent toutefois des millions de paramètres dont l'apprentissage requiert de grandes quantités de données étiquetées. Les géants du web, riches en données, sont donc particulièrement attirés par le deep learning, au point de développer leur propres librairies ou leurs architectures matérielles dédiées. En dehors de ces puissants acteurs, la situation est différente. Si le manque de puissance de calcul est encore parfois une barrière pour les particuliers, la parallélisation de certains calculs sur des processeurs GPU a permis de démocratiser l'utilisation de l'apprentissage profond dans les petites structures du monde socio-économique. En revanche, il est rare d'avoir les quantités de données étiquetées nécessaires à l'apprentissage de modèles comportant fréquemment plusieurs centaines de millions de paramètres.

L'application des architectures profondes à des situations où la quantité de données est limitée constitue ainsi le premier verrou abordé dans ce document. Nous nous intéressons ainsi dans la partie suivante (2.2) aux méthodes de régularisation, et présentons plusieurs contributions dans ce domaine.

Un deuxième verrou majeur lié aux architectures profondes concerne l'intégration de connaissances à priori afin de contraindre les modèles et ainsi alléger le besoin en données. Nous addressons ce problème à travers la reconnaissance de l'écriture manuscrite, sujet qui m'occupe depuis le début de ma thèse de doctorat. Nous proposons dans la section 2.3 plusieurs contributions permettant d'intégrer des connaissances linguistiques dans les modèles statistiques profonds dédiés à la reconnaissance d'écriture.

2.2 Apprentissage profond et régularisation

Le levier majeur permettant d'améliorer les capacités de généralisation d'une machine d'apprentissage est la régularisation. Cette section présente plusieurs travaux concernant la régularisation d'architectures profondes, principalement développés entre 2013 et 2018 à travers les travaux de Soufiane Belharbi, Julien Lerouge, Guillaume Renton et Yann Soullard. Nous présentons dans la section 2.2.1 une introduction aux méthodes de régularisation dédiées aux architectures profondes, puis nous présentons trois contributions basées sur ce principe de régularisation. La première est une méthode de régularisation dédiée aux problèmes à sorties structurées (section 2.2.2) ; la deuxième concerne l'exploitation du transfert d'apprentissage (section 2.2.3), et la troisième est la proposition d'une architecture totalement convolutive basée sur les convolutions dilatées (section 2.2.4).

2.2.1 Apprentissage profond

Les réseaux de neurones sont une classe particulière de modèles d'apprentissage statistique, dont la particularité est d'avoir connu une popularité très inégale tout au long de leur histoire. Après des débuts prometteurs avec l'invention du neurone formel [236] et du perceptron [234], les travaux sont tombés en désuétude à la fin des années 60 à cause de leur inaptitude à l'époque à traiter des problèmes non linéairement séparable. L'introduction de la fameuse rétropropagation du gradient au milieu des années 80 [228] contribua à populariser de nouveau les réseaux de neurones désormais munis de plusieurs non linéarités. Bien que proposant des résultats à l'état de l'art, il était déjà bien connu à l'époque que l'apprentissage par rétropropagation était limité à des réseaux à 2 couches cachées, à cause du problème de "disparition du gradient". Au milieu des années 90, l'arrivée de la théorie de l'apprentissage statistique et des SVM [213] placèrent une nouvelle fois le connexionnisme parmi les "méthodes anciennes" pour une dizaine d'années, jusqu'à l'avènement du deep learning [163, 153, 138, 151, 64]. Le milieu des années 2000 fût une période charnière car plusieurs travaux des "pères du deep learning" permirent d'augmenter significativement le nombre de couches cachées (et donc de non linéarités) des modèles, par l'introduction d'algorithmes complémentaires à la rétropropagation, contournant le problème de disparition du gradient en utilisant notamment le pré-apprentissage.

Dès lors, on peut considérer le deep learning comme l'ensemble des méthodes permettant d'apprendre des architectures avec de nombreuses couches, c'est à dire de contourner les problèmes de disparition du gradient. Parmi ces méthodes, on peut citer les plus importantes telles que le pré-apprentissage des couches basses par des méthodes non supervisées [163, 153, 138, 151], l'utilisation de fonctions d'activation de type ReLU (Rectified Linear Unit) pour conserver la dynamique du gradient [131], les méthodes de régularisation issues de la théorie de l'apprentissage statistique [213], ou encore certaines méthodes visant à contrôler l'énergie

du gradient à l'intérieur des couches [61] ou d'astuces telles que le Dropout [85, 110].

L'ensemble de ces méthodes est aujourd'hui parfaitement opérationnel, et on peut considérer que la disparition du gradient est un problème résolu puisque certaines architectures possèdent désormais plus de 100 couches cachées [39]. L'apprentissage profond est souvent qualifié d'apprentissage de représentations puisqu'il permet de construire au fur et à mesure des couches de représentations de haut niveau des données, pilotées par la tâche supervisée.

Le problème de disparition du gradient étant réglé, la communauté scientifique du machine learning se concentre désormais sur l'amélioration des capacités de généralisation des modèles profonds afin d'alléger le besoin en données des apprentissages. Issue de la théorie de l'apprentissage statistique, la régularisation est la solution la plus communément employée pour diminuer l'erreur de généralisation.

2.2.1.1 Régularisation des architectures profondes

L'apprentissage des réseaux de neurones est connu pour être difficile [130, 37] car ils ont tendance à surapprendre les données d'apprentissage, notamment lorsqu'elles sont peu nombreuses.

La regularisation peut être considérée comme l'ensemble des méthodes permettant de réduire l'erreur de généralisation, indépendemment de l'erreur empirique. La régularisation a longtemps consisté à réduire la complexité des modèles à l'aide d'un terme de pénalité sur la norme ℓ_p des paramètres. Aujourd'hui, d'autres approches sont proposées pour réduire l'erreur de généralisation, sans forcément réduire la complexité des modèles. Même si la frontière entre les deux n'est pas très précise, on peut considérer deux types de régularisation :

— La régularisation explicite, où on cherche explicitement à réduire la complexité des modèles.
— La régularisation implicite, où l'on cherche à améliorer les performances en généralisation, sans se préoccuper de la complexité du modèle.

Régularisation explicite

La limitation de la capacité des modèles se fait généralement par l'ajout d'une pénalité $\Omega(\boldsymbol{\theta})$ à la fonction objectif J [116, 37], où $\boldsymbol{\theta}$ désigne les paramètres du modèle. L'objectif régularisé $\widetilde{J}$ peut ainsi s'exprimer :

$$\widetilde{J}(\boldsymbol{\theta}; \boldsymbol{X}, \boldsymbol{y}) = J(\boldsymbol{\theta}; \boldsymbol{X}, \boldsymbol{y}) + \alpha\Omega(\boldsymbol{\theta}) \tag{2.1}$$

où $\alpha \in [0, \infty]$ est un hyper-paramètre contrôlant l'équilibre entre l'objectif J et le terme de régularisation $\Omega(\boldsymbol{\theta})$ qui est généralement une pénalisation de la norme des paramètres du modèles. Ce terme de pénalité s'exprime sous la forme :

$$\Omega(\boldsymbol{\theta}) = \frac{1}{p}\|\boldsymbol{\theta}\|_p^p \,. \tag{2.2}$$

Cette régularisation sur la norme ℓ_p des paramètres du modèle n'est pas spécifique aux réseaux de neurones. Dans le cas des réseaux de neurones, la pénalité est naturellement appliquée aux poids des connections $\boldsymbol{w}$. Les deux régularisations les plus courantes sont La régularisation ℓ_2 et ℓ_1 :

$$\Omega(\boldsymbol{\theta}) = \frac{1}{2}\|\boldsymbol{w}\|_2^2 \,, \qquad \Omega(\boldsymbol{\theta}) = \|\boldsymbol{w}\|_1 = \sum_i |w_i| \,. \tag{2.3}$$

Dans le cas d'une régularisation ℓ_2, les grandes valeurs de paramètres sont pénalisées et ainsi maintenues proches de l'origine. Cette régularisation est connue sous le nom de *weight decay*, *ridge regression* ou *régularisation de Tikhonov* [233]. Elle tend à annuler les paramètres contribuant peu à la fonction objectif.

La normalisation ℓ_1 est quand à elle une approximation de la « norme »ℓ_0. Par rapport à la norme ℓ_2, la norme ℓ_1 apporte davantage de parcimonie en forcant l'annulation d'un certain nombre de paramètres, tout en limitant également la norme du reste des paramètres [59, 216].

Régularisation implicite

Il existe de nombreuses méthodes spécifiques aux réseaux de neurones permettant d'améliorer la généralisation des architectures, sans toutefois prendre la forme d'un terme de régularisation ajouté à la fonction objectif. Bien qu'elles ne soient pas toutes considérées comme régularisation dans la littérature, nous les considérons comme telles dans la mesure où elles permettent d'améliorer la généralisation des modèles. Nous dressons maintenant une brève liste non exhaustive de ces méthodes.

Augmentation de données : C'est la méthode la plus simple pour améliorer les performances en généralisation des réseaux. Confronté au manque de données étiquetées, les bases d'apprentissage sont augmentées en utilisant des déformations par un bruitage des données [150, 137], ou par un modèle physique[223].

Transfert d'apprentissage : Cette approche vise également à combler le manque de données étiquetées en exploitant un modèle déjà appris pour le transférer sur une nouvelle tâche plus ou moins proche [133, 112].

Réseaux convolutionnels : Dans la mesure où les réseaux convolutionnels [211] limitent forte-

ment le nombre de paramètres d'un modèle profond [2], cette régularisation peut s'apparenter aux effets d'une régularisation ℓ_0. Remarquons qu'ils auraient également pu être considérés comme une régularisation explicite, le partage des poids étant imposé lors de l'apprentissage.

Dropout : Cette méthode consiste à annuler des connexions ou des neurones de manière aléatoire pour contraindre le modèle à s'adapter à de nouvelles situations [85]. Il peut s'apparenter à du bruitage des données.

Batch Normalization : proposé récemment [61], cette normalisation interne au réseau permet de résoudre le problème du "covariance shift" [188] au sein des représentations cachées des réseaux de neurones. Cette méthode améliore la généralisation des modèles et rend parfois l'utilisation du dropout inutile [85].

De nombreuses méthodes récentes visent donc à régulariser les architectures profondes dans le but d'améliorer leur capacités en généralisation, tout en considérant des jeux de données limités.

Nous proposons dans la suite de cette section nos contributions à la régularisation de modèles profonds. La première contribution est une nouvelle méthode de régularisation explicite dédiée aux problèmes à sorties structurées, présentée dans la section 2.2.2. Nous montrons comment l'apprentissage des dépendances entre les variables de sorties peut améliorer les capacités de généralisation des architectures profondes, et validons l'approche proposée sur un problème de régression (détection de points d'intérêt sur des visages, section 2.2.2.3) et sur un problème de classification (segmentation sémantique en imagerie médicale, section 2.2.2.4). Dans la seconde contribution, nous abordons un cas pratique d'apprentissage d'architecture profonde sur un problème d'imagerie médicale avec peu de données étiquetées (section 2.2.3). Le modèle proposé est un CNN appris avec transfert d'apprentissage. La troisième contribution propose d'exploiter des architectures totalement convolutives très légères afin de compenser le manque de données sur un problème de segmentation en lignes d'images de documents (section 2.2.4). Une variante basée sur les convolutions dilatées est présentée.

2.2.2 Régularisation de modèles profonds dédiés aux problèmes à sortie structurées

Cette contribution est basée sur les travaux de Julien Lerouge et Soufiane Belharbi. Nous y proposons une méthode de régularisation capable d'apprendre les dépendances entre les sorties au sein de problèmes à sorties structurées.

Les problèmes à sorties structurées sont des problèmes où le nombre de sorties est supérieur à 1. Dans ces problèmes, il existe généralement des dépendances structurelles entre les sorties,

2. Par exemple, l'architecture à 16 couches VGG16 [88] contient 138 millions de paramètres, dont 120 millions (87%) pour les 3 seules couches denses.

qu'il convient d'exploiter pour réaliser la meilleure prédiction possible. Ces dépendances peuvent être locales ou plus lointaines. Si l'on considère une tâche de segmentation sémantique sur une image naturelle, il est probable que les classes de pixels adjacents soient identiques, car les zones sont généralement homogènes (dépendances locales). En considérant le même problème de segmentation, il est très probable que des pixels de la classe "herbe" se situent en dessous des pixels de la classe "ciel" (dépendances lointaine). De très nombreux domaines sont confrontés aux problèmes à sorties structurées, tel que le traitement automatique des langues (Traduction [176], part-of-speech tagging [215]), la segmentation sémantique d'images [96, 66, 33, 40], la bioinformatique (prédiction de structure de protéines [193]) ou encore l'analyse de la parole (reconnaissance [225] ou synthèse vocale [145]).

Dans les problèmes à sorties structurées, la difficulté majeure réside dans le nombre exponentiel de combinaisons possibles de la structure de sortie, qui demande un nombre important d'exemples. Dans ce travail, nous proposons de considérer la prédiction de sorties structurées comme un apprentissage de représentation, où le modèle devra modéliser les dépendances entre les entrées $\mathbf{x}$ et les sorties $\mathbf{y}$, mais aussi capturer les dépendances entre les variables de chaque espace.

Plusieurs modèles ont été proposés dans la littérature pour ce type de problèmes. On peut citer notamment les méthodes à noyaux [179] et les approches discriminantes [146] qui cherchent à découvrir la structure cachée de l'espace de sortie, et les modèles graphiques qui intègrent des connaissances à priori sur la structure des données. Malheureusement, les deux premières approches souffrent du problème de pré-image et du passage à l'échelle. Bien que très utilisés pour les séquences à une [225, 189, 183], deux [170, 165] voire trois dimensions [155], les modèles graphiques de type HMM et CRF ne possèdent qu'une non linéarité et sont ainsi limitées dans la complexité des frontières de décision qu'elles sont capables de modéliser. Les architectures profondes ont naturellement été utilisées sur des problèmes à sorties structurées, à l'aide d'architecture convolutionnelles pour la segmentation sémantique [66, 71, 69], ou encore avec des architectures récurrentes sur du texte [84, 89, 94], en reconnaissance de la parole [80] ou d'écriture [141]. Cependant, aucune de ces approches ne modélisent explicitement les dépendances entre les variables de sortie.

La contribution de ce travail est un cadre multitâche dédié à l'apprentissage d'architecture profonde pour les prédictions structurées. Nous proposons de combiner deux tâches non supervisées pour l'apprentissage des dépendances au sein des deux espaces d'entrée et de sortie avec la tâche supervisée. Les tâches non supervisées sont instanciées à l'aide d'auto-encodeurs [137, 151] pour apprendre les représentations idoines. Nous montrons l'efficacité de la méthode proposée sur deux tâches : la première est une tâche de régression pour la détection de points d'intérêt sur des images de visage, la deuxième est une tâche de segmentation sémantique appliquée à l'imagerie médicale.

2.2.2.1 Cadre multitâche pour la prédiction de sorties structurées

Considérons un ensemble d'apprentissage $\mathbb{D}$ contenant trois types d'exemples : des exemples avec le signal d'entrée et la sortie cible $(\boldsymbol{x}, \boldsymbol{y})$, des exemples avec le signal d'entrée uniquement $(\boldsymbol{x}, \_)$, et des exemples avec la sortie cible uniquement $(\_, \boldsymbol{y})$. Considérons également l'ensemble $\mathbb{F}$ des exemples contenant au moins les signaux d'entrée $\boldsymbol{x}$, l'ensemble $\mathbb{L}$ contenant au moins les sorties cibles $\boldsymbol{y}$, et l'ensemble $\mathbb{S}$ contenant les signaux d'entrée $\boldsymbol{x}$ et les sorties cibles $\boldsymbol{y}$.

Capture des dépendances dans les signaux d'entrée :

Il s'agit d'une tâche non supervisée cherchant à obtenir une représentation de haut niveau des entrées, supposée plus robuste et informative que $\boldsymbol{x}$. C'est le schéma de pré-apprentissage traditionnel proposé dans la littérature [163, 153, 138, 151]. Cette tâche projette les données d'entrée $\boldsymbol{x}$ dans un espace de représentation intermédiaire $\tilde{\boldsymbol{x}}$ à travers une fonction $\mathcal{P}_{in}$ généralement appelée *encodeur*. On cherche alors à retrouver le signal initial $\hat{\boldsymbol{x}}$ à partir de $\tilde{\boldsymbol{x}}$ à travers un décodeur $\bar{\mathcal{P}}_{in}$.

$$\hat{\boldsymbol{x}} = \mathcal{R}_{in}\left(\boldsymbol{x}; \boldsymbol{w}_{in}\right) = \bar{\mathcal{P}}_{in}\left(\tilde{\boldsymbol{x}} = \mathcal{P}_{in}\left(\boldsymbol{x}; \boldsymbol{w}_{cin}\right); \boldsymbol{w}_{din}\right) \ , \tag{2.4}$$

où $\boldsymbol{w}_{in} = \{\boldsymbol{w}_{cin}, \boldsymbol{w}_{din}\}$. Si les paramètres du décodeur $\boldsymbol{w}_{din}$ sont propres à cette tâche, les paramètres de l'encodeur $\boldsymbol{w}_{cin}$ sont en revanche partagés avec la tache principale (voir Fig.2.1) Ce cadre multitâche est ainsi prévu pour attirer les paramètres partagés entre les deux tâches vers une région où les représentations sont plus robustes, et ainsi gagner en généralisation. Cette tâche peut ainsi être utilisée pour amorcer le processus d'apprentissage de la tache principale. Le critère permettant d'apprendre cette tâche peut ainsi s'écrire :

$$J_{in}(\mathbb{F}; \boldsymbol{w}_{in}) = \frac{1}{\operatorname{card} \mathbb{F}} \sum_{\boldsymbol{x} \in \mathbb{F}} \mathcal{C}_{in}(\mathcal{R}_{in}(\boldsymbol{x}; \boldsymbol{w}_{in}), \boldsymbol{x}) \ , \tag{2.5}$$

où $\mathcal{C}_{in}(\cdot, \cdot)$ est une fonction de coût non supervisée pouvant être calculée sur tous les exemples disposant des signaux d'entrée (i.e. $\in \mathbb{F}$), par exemple un critère des moindres carrés.

Capture des dépendances dans les sorties cibles :

La capture des dépendances dans l'espace de sortie constitue la contribution de ce travail. L'originalité consiste à appliquer une seconde tâche de pré-apprentissage, cette fois ci sur les sorties cibles. Cette tache $\mathcal{R}_{out}$ est donc également une tâche de reconstruction non-supervisée qui projette les sorties cibles $\boldsymbol{y}$ dans une représentation intermédiaire $\tilde{\boldsymbol{y}}$ à travers une fonction encodeur $\mathcal{P}_{out}$.

$\hat{\boldsymbol{y}}$ est reconstruit à partir de $\tilde{\boldsymbol{y}}$ par une fonction décodeur $\bar{\mathcal{P}}_{out}$. Dans les problèmes à sorties structurées, $\tilde{\boldsymbol{y}}$ devrait ainsi encoder la structure cachée entre les variables de sortie $\boldsymbol{y}$. Cette structure est découverte de manière non supervisée, sans avoir à

les formaliser vis-à-vis de la machine. Signalons que ce deuxième pré-apprentissage autorise l'utilisation de données de type $(\_, \boldsymbol{y})$, c'est à dire contenant les sorties cibles sans les signaux d'entrée. En fonction des problèmes considérés, la quantité de ce type de données peut être très important, comme par exemple en reconnaissance de l'écriture ou de la parole, où il est facile de trouver des données textuelles en masse.

$$\hat{\boldsymbol{y}} = \mathcal{R}_{out}\left(\boldsymbol{y}; \boldsymbol{w}_{out}\right) = \bar{\mathcal{P}}_{out}\left(\tilde{\boldsymbol{y}} = \mathcal{P}_{out}\left(\boldsymbol{y}; \boldsymbol{w}_{cout}\right); \boldsymbol{w}_{dout}\right) \ , \tag{2.6}$$

où $\boldsymbol{w}_{out} = \{\boldsymbol{w}_{cout}, \boldsymbol{w}_{dout}\}$. Contrairement à la tâche d'encodage des signaux d'entrée, les paramètres de l'encodeur $\boldsymbol{w}_{cout}$ sont propres à cette tâche, et les paramètres du décodeur $\boldsymbol{w}_{dout}$ sont partagés avec ceux de la tâche cible (see Fig.2.1). Le critère d'apprentissage de cette tâche peut s'écrire :

$$J_{out}(\mathbb{L}; \boldsymbol{w}_{out}) = \frac{1}{\operatorname{card} \mathbb{L}} \sum_{\boldsymbol{y} \in \mathbb{L}} \mathcal{C}_{out}(\mathcal{R}_{out}(\boldsymbol{y}; \boldsymbol{w}_{out}), \boldsymbol{y}) \ , \tag{2.7}$$

où $\mathcal{C}_{out}(\cdot, \cdot)$ est une fonction de coût non supervisée pouvant être calculée sur tous les exemples disposant de leur sorties cibles (i.e. $\in \mathbb{L}$).

Tâche principale :

La tâche principale est l'apprentissage supervisé de la fonction de mapping $\mathcal{M}$ entre les signaux d'entrée $\boldsymbol{x}$ et les sorties cibles $\boldsymbol{y}$. Pour cela, la fonction dispose de paramètres partagés avec l'encodeur des entrées $\mathcal{P}_{in}$, de paramètres partagés avec le décodeur des sorties $\bar{\mathcal{P}}_{out}$, et des paramètres qui lui sont propres $\mathcal{L}$.

$$\hat{\boldsymbol{y}} = \mathcal{M}\left(\boldsymbol{x}; \boldsymbol{w}_{sup}\right) = \bar{\mathcal{P}}_{out}\left(\mathcal{L}\left(\mathcal{P}_{in}\left(\boldsymbol{x}; \boldsymbol{w}_{cin}\right); \boldsymbol{w}_{s}\right); \boldsymbol{w}_{dout}\right) \ , \tag{2.8}$$

où $\boldsymbol{w}_{sup} = \{\boldsymbol{w}_{cin}, \boldsymbol{w}_{s}, \boldsymbol{w}_{dout}\}$. Les paramètres $\boldsymbol{w}_{cin}$ et $\boldsymbol{w}_{dout}$ sont respectivement partagés avec la tâche d'encodage des entrées et de décodage des sorties. L'apprentissage de cette tâche est réalisé en minimisant le critère J_s, où $\mathcal{C}_s(\cdot, \cdot)$ est une fonction de coût :

$$J_s(\mathbb{S}; \boldsymbol{w}_{sup}) = \frac{1}{\operatorname{card} \mathbb{S}} \sum_{(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{S}} \mathcal{C}_s(\mathcal{M}(\boldsymbol{x}; \boldsymbol{w}_{sup}), \boldsymbol{y}) \tag{2.9}$$

L'apprentissage des trois tâches est réalisé en parallèle, en agrégeant les trois critères au sein d'un objectif unique. Afin de pouvoir balancer l'importance des trois tâches, nous avons proposé d'utiliser une somme pondérée par des poids d'importance λ_{sup}, λ_{in} et λ_{out} respectivement pour les tâches principale, d'encodage des entrées et de décodage des sorties. Finalement, si $\boldsymbol{w} = \{\boldsymbol{w}_{cin}, \boldsymbol{w}_{din}, \boldsymbol{w}_{s}, \boldsymbol{w}_{cout}, \boldsymbol{w}_{dout}\}$ est l'ensemble complet des paramètres du modèle, le critère d'apprentissage général peut s'exprimer comme :

$$J(\mathbb{D}; \boldsymbol{w}) = \lambda_{sup} \times J_s(\mathbb{S}; \boldsymbol{w}_{sup}) + \lambda_{in} \times J_{in}(\mathbb{F}; \boldsymbol{w}_{in}) + \lambda_{out} \times J_{out}(\mathbb{L}; \boldsymbol{w}_{out}) \ , \tag{2.10}$$

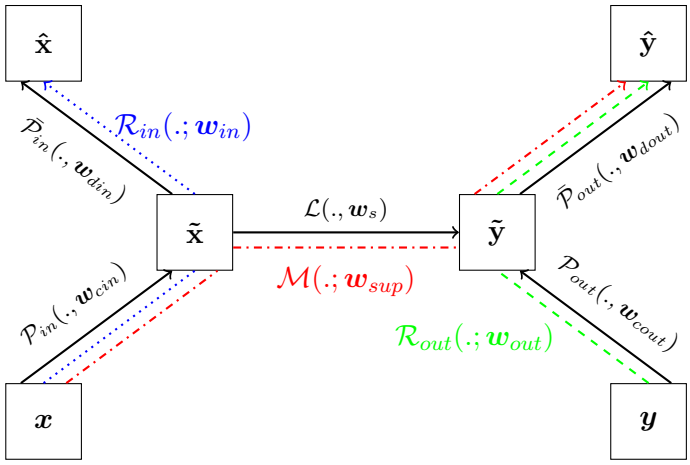

FIGURE 2.1: Cadre multitâche proposé. Les flèches noires pleines désignent les fonctions intermédiaires, les flèches bleues en pointillés désignent la tâche de reconstruction auxiliaire $\mathcal{R}_{in}$, les flèches en tirets verts désignent la tâche de reconstruction auxiliaire $\mathcal{R}_{out}$, et les flèches rouge en pointillé/tirets désignent la tâche principale de mapping $\mathcal{M}$.

Plutôt que d'utiliser des poids fixes pouvant être difficiles à régler, nous avons proposé dans [34] d'utiliser des poids variables au cours du temps. L'équation 2.10 est ainsi modifiée en conséquence :

$$\begin{aligned} J(\mathbb{D};\boldsymbol{w}) = \lambda_{sup}(t) \times J_s(\mathbb{S};\boldsymbol{w}_{sup}) \\ + \lambda_{in}(t) \times J_{in}(\mathbb{F};\boldsymbol{w}_{in}) + \lambda_{out}(t) \times J_{out}(\mathbb{L};\boldsymbol{w}_{out}) \ , \end{aligned} \tag{2.11}$$

où les $t \geq 0$ indicent les époques d'apprentissage. La motivation derrière l'évolution dynamique des poids est que si les deux tâches secondaires doivent permettre d'orienter le début de l'apprentissage afin de mieux généraliser, elles deviennent inutiles voire nuisibles en fin d'apprentissage, lorsque le modèle a suffisamment convergé. Nous avons donc naturellement essayé plusieurs schémas de décroissance de ces poids liés aux tâches secondaires. Le sur-apprentissage de ces tâches secondaires étant difficile à contrôler [93], il est important de les arrêter suffisamment tôt avec une stratégie d'*early stopping*. Nous renvoyons à [34] pour une évaluation expérimentale de différentes politiques d'évolution des poids.

2.2.2.2 Instanciation de l'approche par une architecture profonde

Dans ce travail, le modèle a été implémenté au sein d'une architecture profonde. La tâche principale supervisée est réalisée par un réseau de neurones profond comportant K couches, et les tâches secondaires sont réalisées par des auto-encodeurs comportant K_{in} et K_{out} couches. Au moins une couche doit être totalement réservée à la tâche supervisée de manière à relier $\tilde{\boldsymbol{x}}$ et $\tilde{\boldsymbol{y}}$ dans un espace intermédiaire.

L'optimisation du critère 2.11 est réalisée en utilisant une descente de gradient stochastique. Dans le cas où les données sont toutes étiquetées, l'optimisation peut être réalisée en parallèle sur toutes les tâches. Lorsqu'on utilise des données non étiquetées $(\boldsymbol{x}, \_)$ (ou des données contenant uniquement des sorties cibles $(\_, \boldsymbol{y})$), le gradient ne peut être calculé en une fois et il doit être séparé selon chaque critère et pour chaque mini batch. Le schéma d'optimisation est décrit dans l'algorithme 1 pour un apprentissage en ligne, mais l'apprentissage Mini-batch peut être effectué de la même manière.

Algorithm 1 Our training strategy for one epoch

$\mathbb{D}$ is the shuffled training set. B a sample.
for B in $\mathbb{D}$ **do**
  if B contains $\boldsymbol{x}$ **then**
    `Update` $\boldsymbol{w}_{in}$: Make a gradient step toward $\lambda_{in} \times J_{in}$ using B ▷ Eq.2.5
  if B contains $\boldsymbol{y}$ **then**
    `Update` $\boldsymbol{w}_{out}$: Make a gradient step toward $\lambda_{out} \times J_{out}$ using B ▷ Eq.2.7
  `# parallel parameters update`
  if B contains $\boldsymbol{x}$ and $\boldsymbol{y}$ **then**
    `Update` $\boldsymbol{w}$: Make a gradient step toward J using B ▷ Eq.2.11
  `Update` λ_{sup}`,` λ_{in} `and` λ_{out}.

Nous présentons maintenant dans les deux sections suivantes l'application de cette méthode à un problème de détection de points d'intérêt dans des visages, et à un problème de segmentation sémantique d'images médicales.

2.2.2.3 Application à la détection de points d'intérêt dans des images de visage

Nous avons appliqué le cadre multitâche à la détection de points d'intérêt dans des images de visage. La figure 2.2.2.3 présente quelques exemples issus de la base LFPW [123], où il s'agit de trouver la position de 68 points définis par leurs coordonnées (x, y) dans des images de taille fixe. Il s'agit d'une tâche typique de prédiction à sorties structurées puisque les points sont spatialement interdépendants. Nous avons utilisé dans nos expérimentations deux bases publiques : LFPW [123] (1132 images d'apprentissage, 300 images de test), et HELEN [117] (2000 images d'apprentissage, 330 images de test). Dans les deux bases, les visages sont croppés en patchs de taille fixe 50×50 et les boites englobantes ainsi que les coordonnées des 68 points d'intérêt sont donnés dans [107].

L'évaluation des modèles est réalisée avec les métriques standards du domaine, qui sont l'erreur des moindres carrés entre la prédiction s_p et la vérité terrain s_g normalisée par la distance intra-oculaire D (Normalized Root Mean Squared Error, NRMSE), et la fonction de densité cumulative pour une valeur spécifique de NMSE (Cumulative Distribution Function CDF_{NRMSE}) value calculée sur toute la base (Eq.2.12). CDF_{NRMSE} représente la proportion d'images dont l'erreur est inférieure à la valeur NMSE fixée. Nous donnons également l'aire

FIGURE 2.2: Exemples de visages avec leur points d'intérêt issus de la base LFPW.

sous la courbe CDF, désignée AUC dans la suite de cette section.

$$NRMSE(s_p, s_g) = \frac{1}{N * D} \sum_{i=1}^{N} ||s_{pi} - s_{gi}||_2 \ , \quad CDF_x = \frac{card(NRMSE \leq x)}{n} \ , \qquad (2.12)$$

Nous avons implémenté un modèle comportant $K = 4$ couches avec des fonctions d'activation sigmoïde (tanh sur la dernière couche) selon l'architecture suivante :

$$50 \times 50 = 2500 \ \texttt{entrées} \rightarrow 1024 \rightarrow 512 \rightarrow 64 \rightarrow 136 \ \texttt{sorties} \ (68 \times 2)$$

Nous comparons cette même architecture entraînée selon les 4 configurations suivantes :

- **MLP**, MLP classique ;
- **MLP + in**, MLP avec AE sur les relations en entrée ;
- **MLP + out**, MLP avec AE sur les relations en sortie ;
- **MLP + in + out**, MLP avec AE sur les relations en entrée et en sortie.

L'auto-encodeur d'entrée est un *denoising AE* avec un degré de corruption de 20% ; l'auto-encodeur de sortie est un simple AE. Les deux auto-encodeurs sont entraînés avec une régularisation ℓ_2 et un weight decay de 10^{-2}. Tous les hyper-paramètres ont été optimisés sur la base de validation LFPW. Pour l'importance des poids nous avons réutilisé le schéma que nous avions déterminé dans [34].

Les tableaux 2.1 et 2.2 présentent les résultats obtenus sur les base LFPW et HELEN. Pour les deux bases, nous avons simulé une augmentation de données $(\boldsymbol{x}, \_)$ et $(\_, \boldsymbol{y})$ en utilisant les images de visages sans les points d'intérêt d'une base vers une autre, et les points d'intérêt sans visages d'une base vers une autre.

On peut constater sur les deux bases l'apport des auto-encodeurs pour l'apprentissage des dépendances entre les entrées, ainsi qu'entre les sorties. Les meilleurs résultats sont obtenus en combinant les trois tâches, ce qui montre l'intérêt de l'approche proposée. L'autre résultat

Table 2.1: AUC and $\mathbf{CDF_{0.1}}$ performance sur la base de test LFPW sans et avec augmentation de données.

	Sans augmentation		Avec augmentation	
	AUC	$\mathbf{CDF_{0.1}}$	**AUC**	$\mathbf{CDF_{0.1}}$
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 2.2: AUC and $\mathbf{CDF_{0.1}}$ performance sur la base de test HELEN sans et avec augmentation de données.

	Sans augmentation		Avec augmentation	
	AUC	$\mathbf{CDF_{0.1}}$	**AUC**	$\mathbf{CDF_{0.1}}$
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%**

intéressant concerne l'ajout des données de données non étiquetées $(\boldsymbol{x}, \_)$ ou $(\_, \boldsymbol{y})$, qui permettent d'améliorer significativement les résultats.

2.2.2.4 Application à la segmentation sémantique d'images médicales

Dans cette section, nous présentons l'application du cadre multitâche présenté précédemment à un problème réel de segmentation d'images médicales. Ce travail est issu d'une collaboration avec Romain Modzelewski et le Pr. Fabrice Jardin du centre Henri Becquerel initiée par les stages de Julien Lerouge et Denis Rousselle, et a été publié dans la revue Pattern Recognition [65].

Le problème abordé ici est l'estimation du degré de sarcopénie des patients à partir de scanners CT (Computerized Tomography) complets. La sarcopénie est une perte de masse musculaire, et constitue pour les médecins un facteur important dans le pronostic de plusieurs cancers [102, 83]. Le degré de sarcopénie d'un patient peut être calculé en estimant les surfaces musculaires sur la coupe de scanner correspondant à la lombaire n° 3 (appelée coupe L3 dans la

suite de cette section). Cette tâche est aujourd'hui effectuée manuellement par un radiologue expérimenté et prend environ 5 minutes par patient. Il y a donc un besoin d'automatiser cette tâche, notamment pour les muscles squelettiques qui sont particulièrement difficiles à identifier sur des images CT. Il s'agit donc d'un problème de segmentation sémantique (attribution d'une classe à chaque pixel d'une image), qui est un autre cas typique de problème à sortie structurée, ici à deux classes : {muscle squelettique, autre}.

Cette tâche est difficile car elle comporte beaucoup de variabilité. Au niveau de la population des patients, on observe une forte variabilité intrinsèque des morphologies et de la position des organes, mais aussi de l'état médical, de l'âge ou du sexe qui modifient les formes et textures des muscles. Au niveau des images, il y a également une grande variabilité de la qualité de reconstruction des images sur le scanner, des quantités de radiations reçues par le patient, et de la quantité d'agent de contraste administrée. La distribution des niveaux de gris entre les muscles et les organes internes sont par ailleurs en grandes partie confondues. La figure 2.3 donne quelques exemples de coupes L3 et les masques correspondant aux muscles squelettiques. Afin d'apprendre notre modèle, nous disposons d'une base de 128 coupes L3 512×512 en niveaux de gris 16 bit qui ont été annotées manuellement par un radiologue. La faible taille de la base est un vrai défi pour l'apprentissage d'une méthode de segmentation sémantique, et permettra d'éprouver efficacement les capacités de généralisation de l'approche proposée.

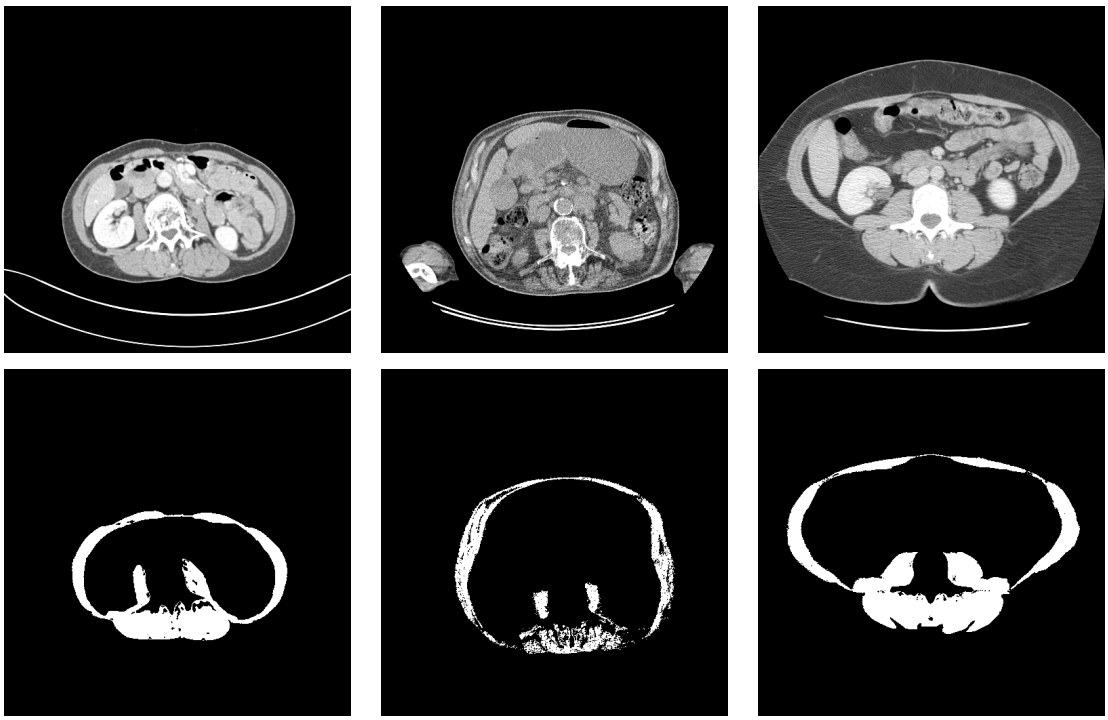

Figure 2.3: En haut : Exemples de coupes L3 ; **En bas :** étiquetage correspondant aux muscles squelettiques.

Application de l'approche multitâche à la segmentation sémantique

Si la détection des points d'intérêt dans les visages (Cf. section 2.2.2) était un problème de régression, nous sommes ici confrontés à une tâche de classification d'un espace de dimensions 512×512 vers une matrice d'étiquettes de la même taille. Afin de réduire la taille de l'architecture, nous avons ignoré les parties des images CT qui sont vides pour toutes images, pour considérer des images 311×457 sans perte d'information.

L'architecture mise en place pour cette tâche comporte $K = 3$ couches dont la taille a été choisie empiriquement sur la base de validation. Les fonctions d'activations sont des *tanh*, sauf pour la dernière couche qui est une softmax. La géométrie retenue est la suivante :

$$311 \times 457 \texttt{ entrées} \rightarrow 1500 \rightarrow 1500 \rightarrow 311 \times 457 \texttt{ sorties}$$

L'apprentissage non supervisé de la première couche est effectué sur les images CT de la coupe L3, celui de la dernière couche sur les étiquettes seules. La taille de la base étant très limitée, nous avons effectué une descente de gradient en ligne.

Résultats

Afin d'évaluer notre approche, nous l'avons comparée à la méthode de référence pour ce type de données proposée par Chung *et al*[139]. Il s'agit d'une méthode ad-hoc classique en imagerie médicale, basée sur un recalage rigide d'un modèle moyen appris sur la base d'apprentissage, suivi d'un recalage non rigide de type FFD (Free Form Deformation). Le code n'étant pas disponible, nous avons réimplémenté cette méthode en optimisant ses paramètres (paramètre de régularisation de la FFD et seuil de décision en sortie) par une procédure de cross validation sur 4 folds.

La métrique utilisée est l'indice de Jaccard qui mesure l'intersection entre la prédiction et la vérité terrain pour les deux méthodes (approche proposée et méthode de Chung). Nous donnons également la différence d'aire relative (appelée Diff.) qui permet de mesurer le taux de sur/sous segmentation.

Concernant la base, nous avons fait un tirage aléatoire des 128 images L3 en 28 images de test, et 100 images pour l'ensemble de cross validation, lui même séparé en 4 folds de 25 images. Le tableau 2.3 donne les résultats obtenus pour les deux méthodes automatiques. Comme dans la section précédente, nous présentons les résultats pour un MLP appris de manière classique (MLP), un MLP avec apprentissage non supervisé des entrées (MLP+in), et un MLP avec apprentissage non supervisé des entrées et des sorties (MLP+in+out).

Il apparait que la méthode de référence fonctionne assez mal sur la base de données qui nous a été fournie. Ceci s'explique par la forte variabilité qu'elle comporte, et par l'impossibilité de cette méthode à prendre en compte les cas s'éloignant du modèle moyen. Par ailleurs, la différence d'aire relative montre la sous estimation de la quantité de muscle estimée par cette

Méthode	Diff. (%)	Jaccard (%)
Reference method (Chung)	-10.6±40.7	60.3±32.5
MLP	0.12±9.78	85.88±5.44
MLP+in	0.15±9.79	85.91±5.45
MLP + in + out	3.37±9.69	**88.47±4.76**

Table 2.3: Performance en test des deux méthodes de segmentation automatique (moyenne ± deviation standard).

méthode. En revanche, on peut constater le bon comportement de l'approche proposée, et notamment l'apport de l'apprentissage des dépendances entre les sorties sur les performances générales. Un résultat remarquable est la très faible différence de performance entre le MLP et MLP+in, qui est dû selon nous à l'impossibilité d'apprendre des textures discriminantes sur des images de type CT (Computerized Tomography). En effet, rappelons que les images CT sont des images reconstruites par le scanner à partir des rayons X et non des images anatomiques réelles telles que les IRM. Nous avions déjà observé lors du stage de Denis Rousselle l'impossibilité d'extraire des descripteurs de texture efficaces sur de telles images.

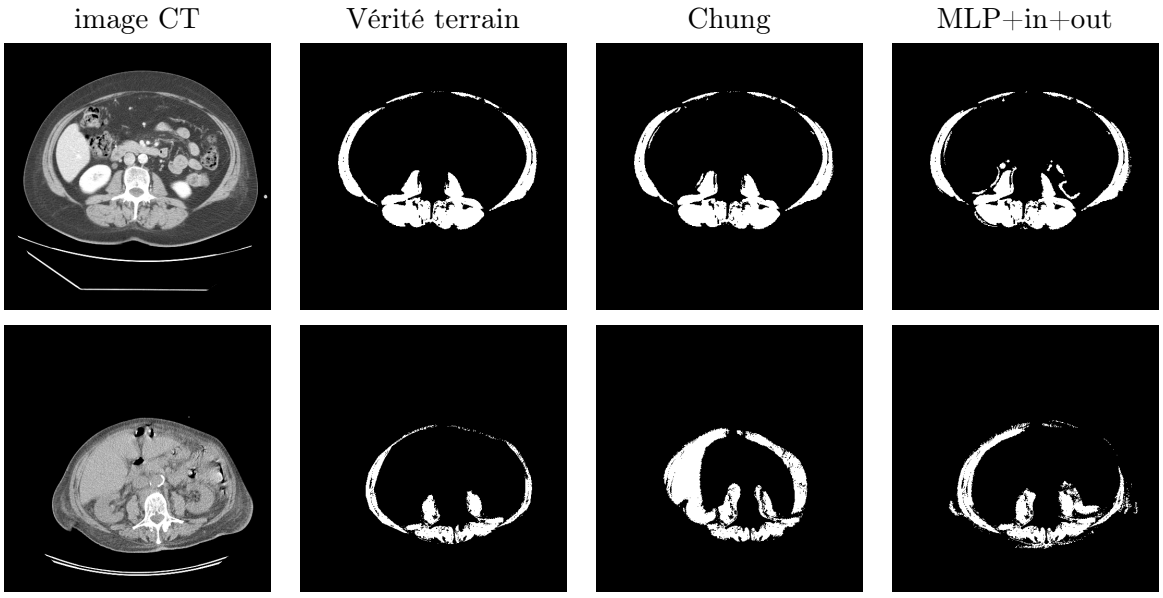

Figure 2.4: En haut : patient non sarcopénique ; en bas : patient sarcopénique

La figure 2.4 montre deux exemples d'images segmentées par la méthode de référence et l'approche proposée. L'image sur la première ligne est considérée comme propre et facile à segmenter par un expert médical (muscles, organes et graisses bien délimités), alors que

l'image de la seconde ligne est bruitée et considérée comme difficile à segmenter (morphologie complexe, bruit de reconstruction). Si les deux méthodes sont performantes dans le cas facile, on peut constater que l'approche proposée est moins sensible aux fortes variations sur le cas difficile.

2.2.2.5 Conclusion et perspectives

Dans ces travaux, nous avons proposé une méthode de régularisation pour les problèmes à sorties structurées basée sur un cadre multitâche générique. L'approche a été validée en régression sur un problème de détection de points d'intérêt et en classification sur un problème de segmentation sémantique.

Il existe plusieurs perspectives à ce travail. La plus évidente concerne l'utilisation de convolutions qui pourraient avantageusement remplacer les auto-encodeurs pour la tâche de reconstruction des entrées. Pour cela, des couches convolutives apprises de manière non supervisées devront être mises en place [70, 78]. Dans le cadre de la segmentation sémantique d'images, les architectures de type "Fully Convolutional Networks" apparus peu après ces travaux (FCN, voir section 2.2.4.3) constituent aujourd'hui une solution efficace et plus légère que l'approche "dense" proposée. L'introduction du terme de régularisation proposé dans nos travaux pourraient cependant être utilisé pour apprendre la déconvolution des FCN.

Par ailleurs, nous avons montré qu'il pouvait être intéressant de faire évoluer les poids des différentes tâches au cours de l'apprentissage, mais nous avons uniquement proposé une politique fixée à priori. Il serait intéressant d'automatiser le réglage des poids au fur et à mesure de l'apprentissage, en se basant par exemple sur l'erreur des différentes tâches sur la base de validation.

2.2.3 Architecture convolutionnelle et transfert d'apprentissage

2.2.3.1 Introduction

Comme nous avons pu le voir dans la section 2.2, l'amélioration des capacités de généralisation des architectures profondes peut s'effectuer à l'aide de couches convolutionnelles qui, à travers un partage des poids, allège considérablement le nombre de paramètres des modèles. Il en résulte un besoin moindre en exemples d'apprentissage par rapport à une architecture dense. Mais l'empilement des couches convolutionnelles nécessaires à l'extraction de représentations de haut niveau (généralement plus d'une dizaine de couches) reste difficile à régulariser sans un minimum de données d'apprentissage.

Ces dernières années, le transfert d'apprentissage a été proposé afin de pallier ce problème. Le transfert d'apprentissage consiste à exploiter une architecture entraînée sur une tâche où les données sont abondantes vers une autre tâche plus ou moins lointaine qui possède un

nombre limité de données. L'idée générale est que si la tâche source dispose de suffisamment de données, les représentations apprises seront suffisamment génériques pour traiter un problème connexe, notamment dans les couches basses. Dans la pratique, cette idée est généralement appliquée en réutilisant des modèle pré-appris sur la base imageNet [116] pour en exploiter leurs couches convolutionnelles, sur lesquelles on applique des couches denses de décision adaptées au problème cible.

Dans le cadre de la thèse de Soufiane Belharbi, nous avons abordé ce thème appliqué à un problème d'imagerie médicale, dans le cadre de notre collaboration avec le centre Henri Becquerel. Ce travail a donné lieu à une publication dans une revue [12] et a été présenté en conférences [6, 3]. Nous présentons dans la section suivante le cadre applicatif de ces travaux, qui sont dans la continuité de la contribution précédente sur la segmentation sémantique de coupe L3.

2.2.3.2 Détection de coupe dans un scanner complet

Le problème abordé dans ces travaux concerne la détection d'une coupe particulière dans un scanner complet : la coupe correspondant à la vertèbre lombaire n^{o} 3 (appelée L3 dans la suite de cette section). Cette tâche est illustrée en figure 2.5. Nous avons vu dans la section précédente (section 2.2.2.4) que cette opération pouvait être utile dans le cadre du calcul automatique du degré de sarcopénie chez les patients atteints de cancer [102, 75], mais il y a de nombreux autres exemples d'utilisation, par exemple pour les maladies cardiovasculaires [77], en chirurgie [128, 100] ou encore pour le calcul du dosage de chimiothérapie à administrer [98, 83]. Ce besoin vient d'un intérêt croissant de la part des médecins à analyser la composition corporelle pour estimer le pronostic des patients. Plutôt que d'analyser les scanners complets, l'estimation des indicateurs se fait souvent à partir d'une seule coupe connue pour être représentative du corps complet. La coupe L3 est connue pour être un bon estimateur de la composition corporelle complète [198, 169].

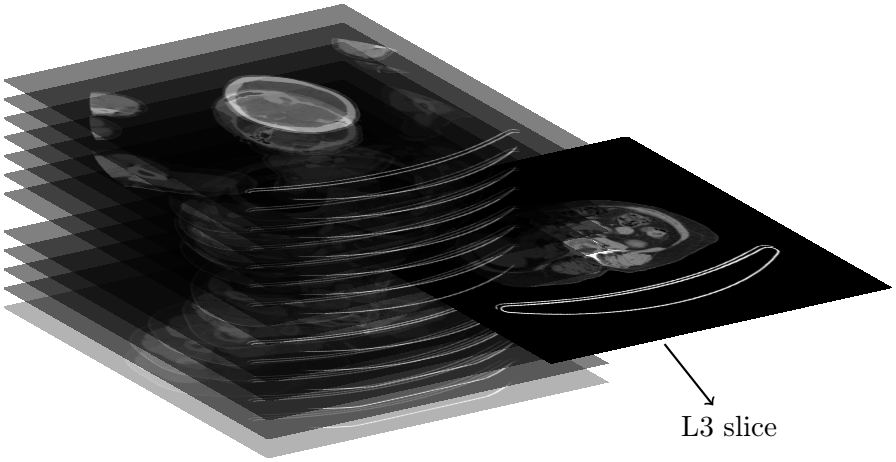

Figure 2.5: Identification d'une coupe L3 dans un scanner complet.

Rappelons qu'un scanner CT est composé de coupes de taille fixe (généralement 512 × 512), dont le nombre et l'épaisseur sont variables en fonction du type de scanner et de l'examen réalisé. On retrouve évidemment dans les images CT toute la variabilité abordée dans la section précédente : variabilité anatomique des patients, type de machine, qualité de la reconstruction, etc. La détection d'une coupe dans une pile d'images CT est plus spécifiquement confrontée aux problèmes suivants :

— Variabilité du champ de vision des scanners : les examens ne concernent pas tous la même partie du corps (corps complet, thorax uniquement, tête uniquement, etc.). Il en résulte un nombre de coupes variable.
— Variabilité inter-patients : du fait de la position variable des organes dans le corps, deux coupes L3 de deux patients différents peuvent être totalement différentes (voir image 2.6).
— Nature répétitive des vertèbres : inversement, au sein d'un même patient deux coupes issues de différentes vertèbres peuvent être très similaires (voir image 2.7).

Qu'elle soit manuelle ou automatique, la détection de la coupe L3 nécessite donc l'analyse du scanner dans son ensemble. Afin de trouver la L3, les médecins parcourent généralement les coupes de haut en bas en partant du sacrum facilement identifiable, pour remonter en comptant le nombre de vertèbres jusqu'à arriver à la L3. Si la segmentation d'images médicales a été extensivement traité dans la littérature, la détection d'une coupe particulière dans un scanner complet a été très peu abordé. Nous présentons un bref état de l'art de la littérature pour ce problème difficile.

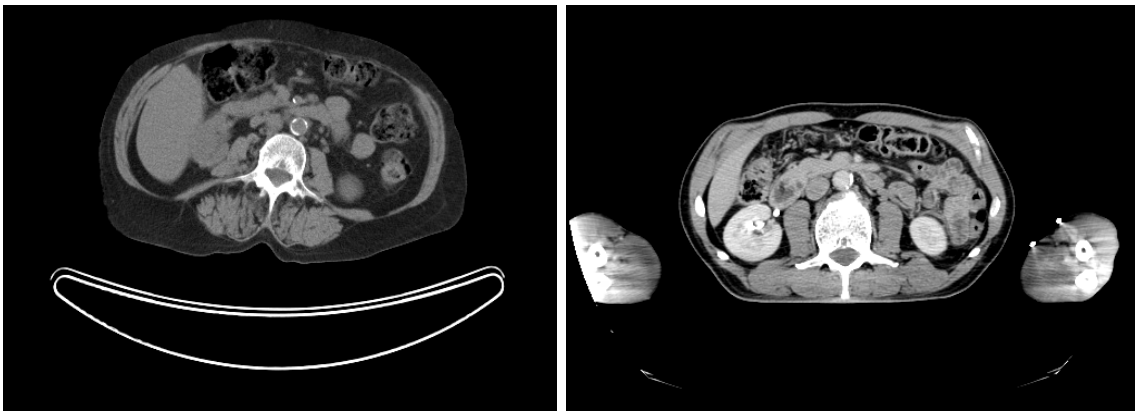

Figure 2.6: Deux coupes L3 de patients différents présentant de nombreuses disparités.

D'un point de vue machine learning, la localisation d'une coupe dans un scanner complet peut être vu de trois manières différentes.

Une problématique de classification de coupes : Il s'agit de décider pour chaque coupe composant le scanner s'il s'agit de la coupe recherchée ou pas. Comme nous l'avons montré précédemment (voir figure 2.7), le manque de contexte rend impossible la

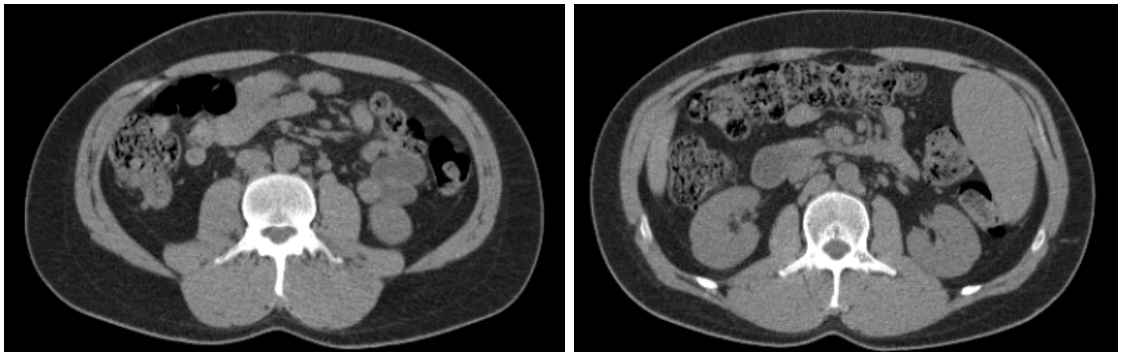

Figure 2.7: Deux coupes issues d'un même patient : L3 (à gauche) et L2 (à droite). La similarité des formes ne permet pas de décider en se basant uniquement sur cette coupe.

décision avec ce type d'approche [3]. À notre connaissance, cette approche n'a pas été utilisée dans la littérature.

Un problème d'étiquetage de séquence : Étant donné la séquence de coupes, on cherche à déterminer la séquence des étiquettes de vertèbre correspondante. On prend ainsi une décision contextuelle pour l'ensemble du scanner. Ce type d'approche a été proposé avec succès pour l'étiquetage des vertèbres à partir d'une image de colonne vertébrale complète [125, 103, 97, 127, 115] à l'aide de modèles graphiques HMM [115] ou MRF [127]. L'inconvénient de ce type d'approche est qu'il nécessite un étiquetage de chacune des coupes du scanner pour apprendre les modèles de séquence. Dans la pratique, un scanner comporte environ 800 coupes, ce qui rend improbable l'obtention de telles données.

Un problème de régression : Étant donné l'ensemble des coupes, on cherche à estimer la position de la L3, c'est à dire la hauteur ou le numéro de la coupe correspondant. Cette approche permet d'obtenir une décision globale puisqu'elle prend en compte le scanner complet, et ne nécessite qu'un étiquetage léger des données : le numéro de la coupe L3 pour chaque scanner. À notre connaissance, ce problème de détection de coupe n'a jamais été formulé comme un problème de régression dans la littérature. C'est l'approche que nous proposons de mettre en œuvre à l'aide d'architectures profondes.

2.2.3.3 Régression pour la détection de coupe dans des scanners complet

Nous souhaitons donc utiliser une approche convolutionnelle afin de générer les bonnes représentations des données à partir d'un scanner complet de taille $512 \times 512 \times N$, N étant

3. Par acquis de conscience, nous avons tout de même essayé, cette approche a donné des résultats catastrophiques.

le nombre de coupes (variable) d'un scanner, pour calculer la hauteur de la coupe L3. Nous sommes donc confrontés à plusieurs problématiques : les dimensions importantes de l'espace d'entrée, le nombre de coupes variable et le manque de données étiquetées.

L'approche proposée permet de prendre en compte toutes ces contraintes. La figure 2.8 donne un aperçu de la chaîne de traitement. Le scanner CT complet est dans un premier temps converti dans une autre représentation en utilisant une transformation par projection d'intensité maximale (MIP) afin de réduire la dimension de l'espace d'entrée. L'image MIP est ensuite analysée par un régresseur CNN à l'aide d'une fenêtre glissante afin de prendre en compte le nombre de coupes variable. Ce régresseur est appris par transfert d'apprentissage afin de résoudre le problème de manque de données étiquetées. Un post traitement sur le signal résultant de l'application du CNN sur l'ensemble du MIP est ensuite effectué pour calculer la prédiction finale du modèle et estimer la position de la coupe L3. Les trois prochaines sections détaillent ces trois étapes.

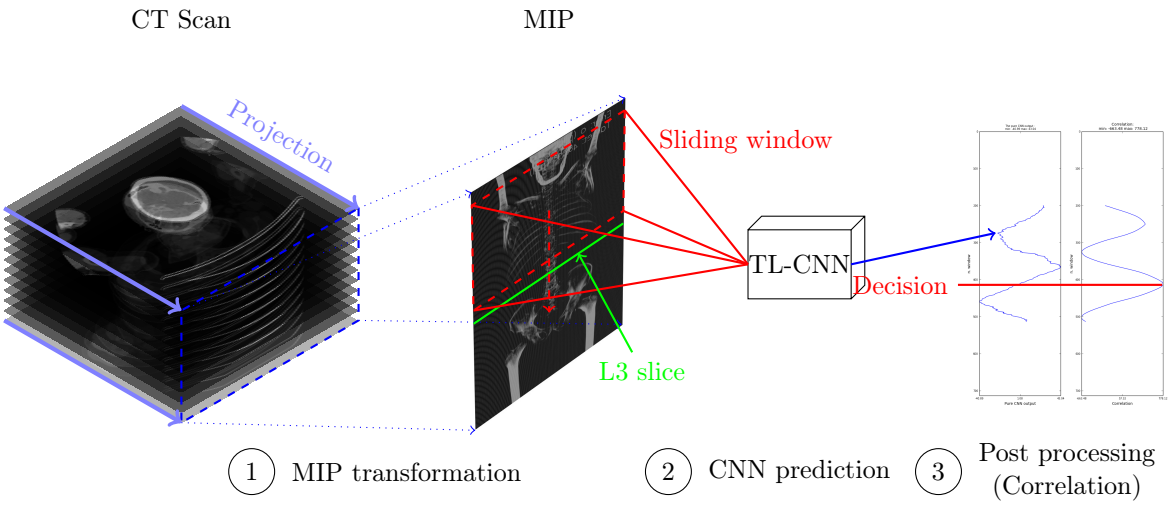

Figure 2.8: Aperçu général de l'approche proposée : transformation MIP, prédiction CNN, post traitement du signal.

Transformation MIP

Si l'on considère le volume 3D d'un scanner brut, la dimension de l'espace d'entrée est d'environ 200 millions. Bien que structurellement un CNN soit capable de digérer de telles données, il convient de compresser l'information afin d'alléger l'architecture et ainsi le nombre de données requis. Comme nous cherchons la coupe correspondant à une vertèbre, il est naturel de se concentrer sur l'information squelettique du scanner. Celle ci peut être obtenue par une transformation appelée MIP [226, 222] (Maximum Intensity Projection), qui réduit considérablement la dimension du signal. Cette transformation est une projection frontale du volume sur une image dont l'intensité des pixels est obtenue par le maximum d'intensité rencontrée dans les voxels traversés. La figure 2.9 donne des exemples d'images MIP frontales.

Cette transformation permet de réduire la dimension d'entrée de $512^2 \times N$ à $512 \times N$.

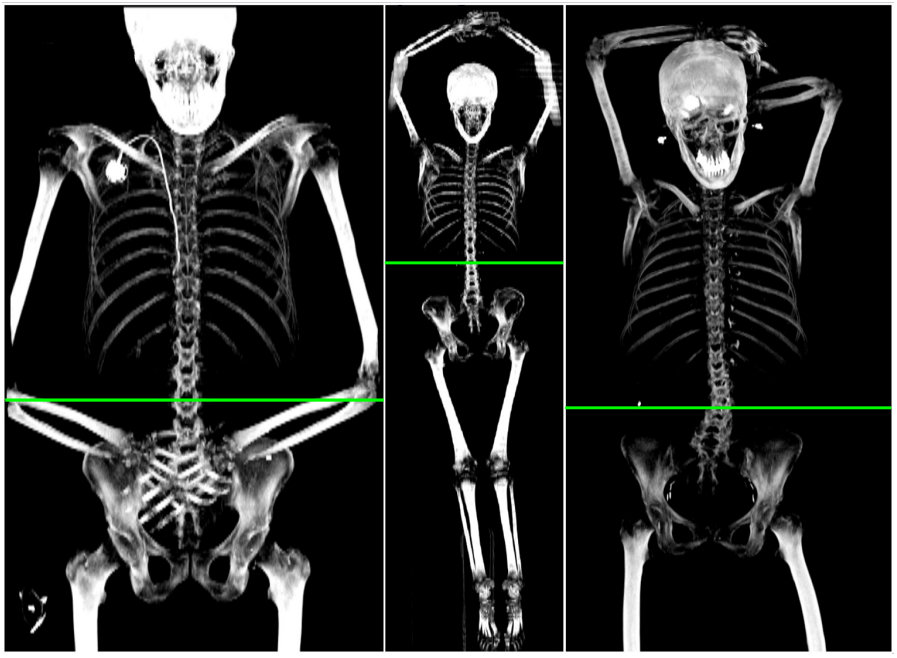

Figure 2.9: Exemples de projection MIP frontale, avec la position de la L3 annotée.

Régresseur CNN et transfert d'apprentissage

L'utilisation des architectures convolutionnelles s'est rapidement répandue dans le domaine médical ces dernières années, par exemple pour le diagnostic du cancer [87, 91], la segmentation [99, 60, 63], en encore en histologie [148]. Dans tous ces travaux, les auteurs sont confrontés au manque de données annotées et les nombreuses méthodes de régularisation présentées en section 2.2 sont utilisées, comme par exemple dans [63].

Comme déjà abordé dans la section 2.2, le transfert d'apprentissage permet de combler le manque de données étiquetées, et a été appliqué avec succès sur des tâches variées telles que la reconnaissance de caractères [62, 112], l'identification de signatures [38], ainsi qu'en imagerie médicale [53, 46].

Nous avons ainsi transféré les couches convolutionnelles d'un modèle pré-appris sur imageNet dans une architecture profonde complète destinée à traiter notre tâche de régression. Pour cela, nous avons remplacé toutes les couches denses des modèles pré-appris par une seule couche dense. Ce transfert d'apprentissage est illustré en figure 2.10.

Les images MIP étant de taille variable, une étape supplémentaire est nécessaire avant d'appliquer un régresseur CNN statique. Bien qu'un redimensionnement de l'image soit possible, il détruit considérablement l'image d'entrée. Nous avons donc choisi de traiter l'image MIP par l'intermédiaire d'une fenêtre glissante de taille fixe. On apprend au modèle à prédire la position relative de la L3 par rapport au centre de la fenêtre.

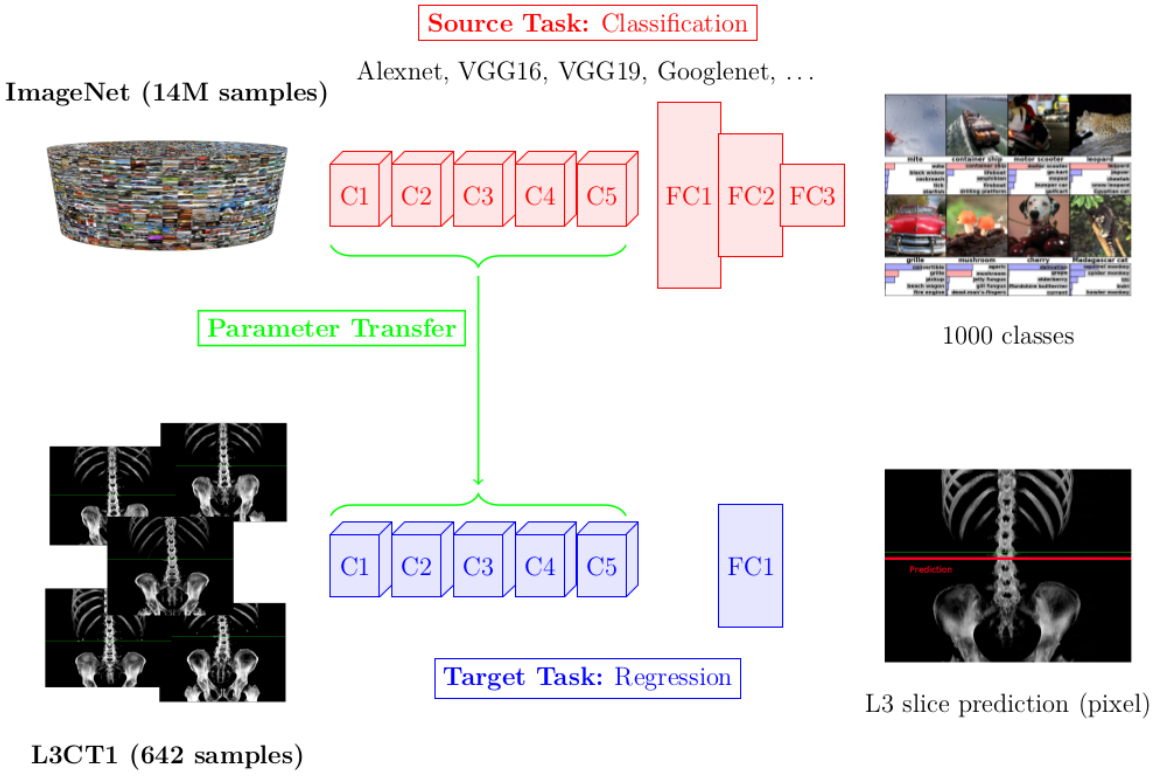

Figure 2.10: Transfert d'apprentissage entre une tâche de classification d'images naturelles et une tâche de régression pour la détection de coupe L3. Les C_i sont les couches convolutionnelles, les FC_i sont des couches denses de décision. Les C_i initiaux sont utilisés pour initialiser le modèle de la tâche cible afin de compenser le manque d'exemple.

Estimation de la L3 par le traitement du signal issu du CNN

En phase de décision, la fenêtre glissante est appliquée sur l'image MIP et le régresseur CNN produit une séquence d'estimations de position relatives de L3. La figure 2.11 montre un exemple d'une telle séquence (au centre). On peut distinguer 3 phases différentes dans le signal de sortie : i) lorsque la fenêtre parcourt le MIP au dessus de la L3, le régresseur a tendance à produire un signal aléatoire. ii) lorsque la fenêtre contient la coupe L3, le CNN estime plus précisément sa position relative, qui diminue linéairement par rapport à son déplacement le long de la L3. iii) en sortant de la zone contenant la L3, le signal tend à redevenir aléatoire. On s'attend alors à observer dans le signal complet un morceau de signal proche d'une droite théorique de pente $\frac{\Delta y}{\Delta x} = -1$, qui est caractéristique de la présence de la coupe L3. Une inter-corrélation classique entre la droite théorique et le le signal de sortie du régresseur permet de détecter efficacement ce phénomène. L'estimation de la position de la L3 est donc faite sur le maximum de cette fonction d'inter-corrélation, ce qui permet d'obtenir une prédiction robuste en moyennant les décisions du CNN. La figure 2.11 (droite) illustre cette estimation.

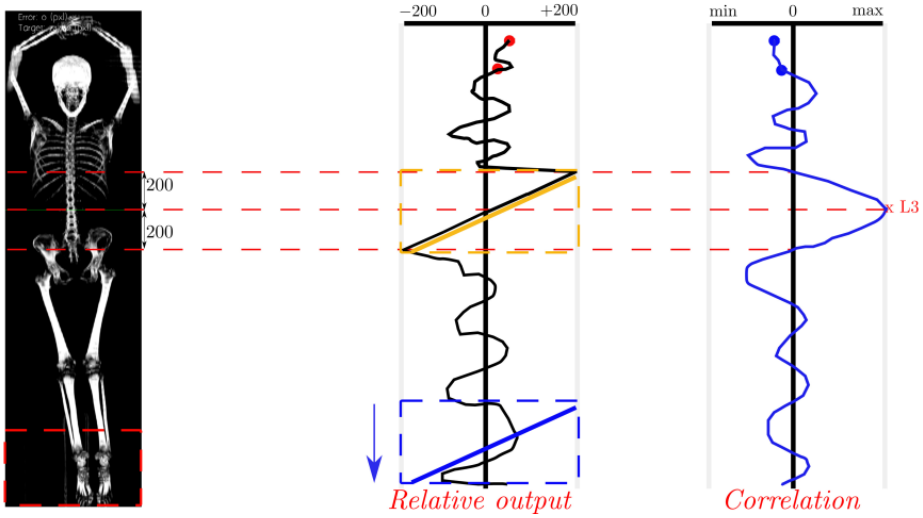

Figure 2.11: À gauche : Le régresseur CNN parcourt le MIP complet de haut en bas, et produit un signal de positions de coupe L3 relatives au centre de la fenêtre (au centre). À doite : le résultat de l'inter-corrélation du signal avec la droite théorique de pente -1. La corrélation maximum correspond à la position de la L3.

2.2.3.4 Expériences et résultats

L'approche est validée sur une base de 642 scanners CT provenant de différents patients, annotés par un radiologue. Les MIP frontaux correspondants sont générés, de largeur fixe 512 pixels et de hauteur variant entre 659 et 1862 pixels. Des exemples sont donnés en figure 2.9. Un séparation de la base en 5 folds est effectuée pour la validation croisée, avec une séparation effectuée au niveau des patients.

Pour nos expériences, nous avons comparé plusieurs modèles de régression.

- **RF500 :** Nous avons implémenté une forêt aléatoire avec 500 arbres (RF500) utilisée en régression [182, 210], combinée avec des caractéristiques Local Binary Pattern (LBP) particulièrement réputées en imagerie médicale [132].
- **CNN4 :** Il s'agit d'un CNN appris sans transfert d'apprentissage, avec 4 couches de convolution avec max-pooling vertical et une couche de décision. Les hyper paramètres ont été optimisés sur la base de validation.
- **CNNs préappris :** Plusieurs modèles pré-appris sur imageNet [140] ont été récupérés pour procéder au transfert d'apprentissage : Alexnet [116], VGG16 et VGG19 [88], Googlenet (Inception V1) [90]. Seule une couche dense remplace les couches de décision précédentes [4]. Les modèles pré-appris demandant des images RGB, les MIP en niveau de gris sont dupliqués sur les trois canaux. Une régularisation ℓ_2 a été utilisée pour tous ces modèles.

4. l'ajout de plusieurs couches denses n'a pas permis d'améliorer les résultats.

Ces modèles ont été évalués en validation croisée en calculant l'erreur de prédiction exprimée en nombre de coupes. Les moyennes et écarts type (μ_e, σ_e) produits par chaque approche sont présentées dans le tableau 2.4.

Table 2.4: Moyennes et écarts type de l'erreur exprimée en coupes pour plusieurs méthodes (pour chaque fold et en moyenne).

	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
Moyenne	10.50 ± 10.80	2.78 ± 2.48	2.45 ± 2.42	$\mathbf{1.82 \pm 2.32}$	1.83 ± 1.83	2.54 ± 4.22

On peut constater que les modèles pré-appris produisent de meilleures performances que les modèles appris uniquement sur les images MIP. En particulier, VGG16 donne les meilleurs résultats avec une erreur de 1.82 ± 2.32 coupes. À titre de comparaison, l'écart entre les coupes est de 2, 3 ou 5mm suivant les scanners CT. L'épaisseur moyenne d'une vertèbre étant d'environ 2.5 cm, ce résultat est satisfaisant. Un des retours des radiologues sur ce travail est que même quand la méthode génère une erreur de quelques coupes, la prédiction ne sort jamais de la vertèbre L3.

Machine learning *vs.* radiologues : le match

S'il est évident que les modèles pré-appris donnent les meilleurs résultats, il est plus difficile de savoir si ces résultats sont exploitables en environnement clinique. Pour cela, nous avons utilisé un nouveau jeu de données de 43 scanners annotés par les même médecins que la base de 642 scanners, que nous avons soumis à 3 autres radiologues. Nous avons comparé leurs prédictions à deux temps différents t_1 et t_2 avec celles de l'approche de machine learning proposée. Les résultats sont présentés dans le tableau 2.5.

Table 2.5: Comparaison des performances des systèmes automatiques avec 3 radiologues. Les annotations sont fournies par un quatrième radiologue, qui elles aussi varient entre t_1 et t_2 (Erreurs exprimées en coupes).

	Erreur par rapport à Radiologue #4 (en coupes)				
	CNN4	VGG16	Radiologue #1	Radiologue #2	Radiologue #3
Expérience à t_1	2.37 ± 2.30	1.70 ± 1.65	0.81 ± 0.97	0.72 ± 1.51	0.51 ± 0.62
Expérience à t_2	2.53 ± 2.27	1.58 ± 1.83	0.77 ± 0.68	0.95 ± 1.61	0.86 ± 1.30

Ces résultats montrent que les radiologues sont plus précis que la meilleure des méthodes

automatiques (VGG16) d'environ 50%. Il est toutefois intéressant de remarquer qu'il existe une erreur inter-annotateur non négligeable, et même une erreur intra-annotateur puisque les prédictions entre les temps t_1 et t_2 ne sont pas identiques. Remarquons finalement que la référence a elle même changée puisque les résultats des méthodes déterministes ont également changé. Au delà du gain de temps que procure l'automatisation, ces méthodes ont donc l'avantage de faire des prédictions stables.

2.2.3.5 Conclusion

Dans cette contribution, nous avons partagé un outil de perception générique que sont des couches convolutionnelles entre un problème de classification d'images naturelles couleurs et un problème d'imagerie médicale sur des images en niveau de gris reconstruites à partir de rayon X. Bien que les signaux d'entrée soient totalement différents, le transfert d'apprentissage a permis d'améliorer considérablement les résultats, participant ainsi à la régularisation du modèle.

En terme de perspectives, l'aspect séquentiel de l'image MIP pourrait être efficacement pris en compte à l'aide d'architectures dynamique de type réseaux récurrents. Une approche plus ambitieuse serait de reposer le problème différemment afin de proposer une approche totalement "end-to-end" pour relier le volume 3D de hauteur variable à la valeur cible. Pour cela, la projection MIP ainsi que le post-traitement devraient être intégrés au sein du processus d'apprentissage. Concernant la projection MIP, elle pourrait être remplacée par des convolutions 3D comme il en existe dans les architectures U-net [71]. Concernant le post traitement, il s'agirait d'effondrer les trois dimensions (dont l'une est variable : la hauteur) du scanner pour calculer la valeur cible. Un pooling hiérarchique [81] sur l'axe z pourrait par exemple être mis en œuvre. On se rapprocherait ainsi d'une architecture totalement convolutionnelle particulièrement légère et adaptée aux signaux dynamiques.

2.2.4 Architectures totallement convolutives et convolution dilatées

2.2.4.1 Introduction

Comme nous venons de le constater, les architectures convolutives permettent de réduire drastiquement le nombre de paramètres d'une architecture profonde. Proposées très récemment, les architectures totalement convolutionnelles suppriment les couches denses, et réalisent un pas supplémentaire vers la régularisation des architectures profondes. Dans cette contribution réalisée dans le cadre du projet INKS et publiée dans *IJDAR* [7], nous avons proposé une variante efficace des architectures totalement convolutionnelles basée sur les convolutions dilatées. L'approche est validée sur un problème de segmentation d'images de documents en lignes, en utilisant un transfert d'apprentissage.

2.2.4.2 Architectures totalement convolutives

Une architecture totalement convolutive (ou FCN pour "Fully Convolutional Network") est une architecture convolutive (CNN) dont les couches denses ont été retirées [66, 51]. Cette spécificité apporte plusieurs avantages. Tout d'abord, cela permet d'appréhender des signaux de tailles variables, puisque contrairement aux couches denses, les couches convolutionnelles s'adaptent à la taille du signal d'entrée. Deuxièmement, comme vu en section 2.2.1.1, l'élimination des couches denses permet de limiter fortement le nombre de paramètres des architectures. Troisièmement, les FCN sont capables de conserver une information spatiale, contrastant avec les CNN qui agrègent ces informations dans les couches denses afin de calculer les valeurs cibles. Appliqués aux images, les FCN permettent ainsi de produire efficacement des cartes de probabilités contenant une description spatiale de l'image d'entrée. Les FCN sont ainsi particulièrement adaptés aux tâches de segmentation sémantique.

Le problème majeur concernant les FCN est la manière dont sont reconstruites les sorties à partir des représentations calculées dans les couches internes. En effet, l'application d'une couche de convolution est généralement couplée à une étape de pooling qui réduit la résolution afin d'étendre les champs réceptifs des filtres sans augmenter le nombre de paramètres. La représentation interne des signaux (souvent des images) est donc de haut niveau mais de basse résolution. Lorsqu'on cherche à étiqueter les pixels d'une image, il faut donc augmenter cette résolution pour produire une sortie de la même taille que l'image d'entrée. Trois méthodes ont été proposées dans la littérature pour cela : la déconvolution, l'unpooling [5] et les convolutions dilatées. Nous décrivons brièvement ces trois méthodes.

Déconvolution

Historiquement, la déconvolution a été la première méthode proposée par Long *et al.* [66], et est toujours très utilisée [24, 49, 72]. L'idée est d'apprendre les déconvolutions comme les transformations inverses des convolutions. En décision, on applique les filtres de déconvolution avec une décalage égal à $\frac{1}{f}$ pour sur-échantillonner la sortie d'un facteur f (voir figure 2.12 gauche). Les filtres de déconvolution sont entraînés en même temps que les filtres de convolution, rendant le réseau plus profond, donc plus expressif mais aussi plus gourmand en données.

Unpooling

Le deuxième approche pour augmenter la résolution des représentations interne est l'unpooling. L'unpooling est la transformation inverse du pooling, et est appliquée de manière symétrique aux couches de pooling. L'idée est de stocker la position des activations gagnantes

5. Une proposition de traduction moyennement satisfaisante serait "agrégation inverse"

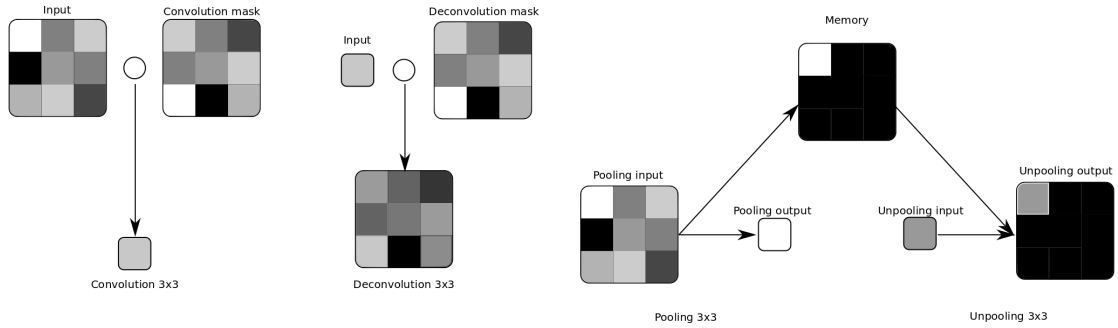

Figure 2.12: *À gauche :* Convolution et déconvolution. La déconvolution est effectuée par une couche convolutionnelle appliquée avec un décalage (stride) de 1/3.
À droite : Pooling et unpooling. Les positions des valeurs remportant le max-pooling sont stockées et réutilisées pour l'unpooling (les pixels noirs représentent des valeurs nulles).

dans les différentes couches de pooling. Le sur-échantillonnage est effectué en plaçant la valeur du pixel considéré à la place de l'activation gagnante sauvegardée. Les pixels voisins sont laissés à 0 (voir figure 2.12 droite). Contrairement à la déconvolution, l'unpooling ne nécessite pas de paramètres supplémentaires. En revanche la reconstruction de l'image est partielle puisqu'elle comprend de nombreux pixels nuls. Une couche de convolution [52] ou de déconvolution [69] est donc souvent employée pour "lisser" l'image de sortie, augmentant de nouveau le nombre de paramètres.

Convolutions dilatées

Contrairement à la déconvolution et à l'unpooling qui cherchent à sur-échantillonner les représentations internes, la convolution dilatée évite l'étape de sous échantillonnage, ce qui permet de conserver une résolution constante pour toutes les représentations internes de l'architecture. En effet, rappelons qu'une convolution appliquée avec un décalage unitaire et sans couche de pooling produit une image de même taille que l'image d'entrée (aux effets de bord près, pouvant être contournés par un padding de l'image d'entrée). Mais l'absence de pooling provoque des champs réceptifs réduits, limitant la prise en compte du contexte lors du calcul des représentations. Il est possible d'augmenter la taille des champs réceptifs en augmentant la taille des filtres, mais cela affecte grandement le nombre de paramètres du réseau.[6] Les convolutions dilatées permettent de contourner ce problème. Elles reposent sur l'algorithme "Atrous" proposé par Holschneider *et al.* [224], initialement proposé dans la transformée en ondelettes. Le principe consiste à sous-échantillonner les éléments pris

6. Un filtre 9×9 comporte 81 paramètres, contre 9 seulement pour un filtre 3×3 associé à un pooling 3×3. Pour obtenir les champs réceptifs de VGG16 sans pooling, il faudrait 4225 paramètres pour chaque filtre, contre 9 seulement avec le pooling.

en compte par le noyau de convolution, de manière à élargir le contexte sans augmenter le nombre de paramètres. La figure 2.13 illustre le champ réceptif d'une convolution dilatée pour plusieurs valeurs du facteur r.

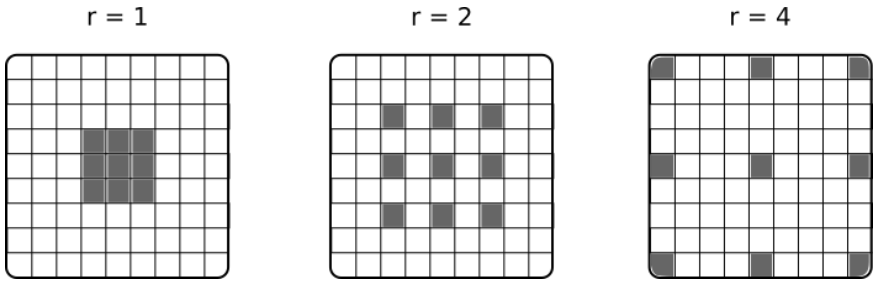

Figure 2.13: Champs réceptifs de convolutions dilatées pour différents facteurs de dilatation r.

La convolution dilatée est donc une généralisation de la convolution, qui a montré des résultats intéressants dans plusieurs travaux de segmentation sémantique [36, 15, 76]. Ces bons résultats s'expliquent selon nous par les avantages que procurent la convolution dilatée : i) faible nombre de paramètres et donc bonnes capacités de généralisation, ii) ajustement facilité de la taille des champs réceptifs, et iii) résolution constante évitant la difficile étape de sur-échantillonnage.

2.2.4.3 FCN avec Convolutions dilatées pour la segmentation de documents en lignes

Dans cette section, nous présentons une contribution pour la segmentation d'images de documents manuscrits en lignes de texte. L'extraction des lignes de texte est un problème central en analyse d'images de document car il est généralement le point d'entrée des moteurs de reconnaissance de texte [154]. Dans ce problème, nous avons choisi de définir les lignes de texte par la zone appelée "X-height" qui contient le corps de texte, sans les hampes ni les jambages. En effet, nous avons montré dans [26] que cette représentation était la plus adaptée car elle évitait le chevauchement des lignes de texte que l'on peut observer avec des boites englobantes, tout en autorisant les lignes courbes. La figure 2.14 montre un exemple d'un document et la vérité terrain "X-height" associée.

La segmentation d'un document pouvant être vue comme une segmentation sémantique en deux classes (ligne de texte/fond), nous avons choisi d'utiliser les FCN. Nos données étant en nombre limité, un maximum de régularisation est requis. Au regard des trois types de FCN présentés dans la section précédente, nous avons donc opté pour les convolutions dilatées, ce qui n'avait jamais été proposé à l'époque de ce travail.

Architecture proposée

Dans l'architecture proposée, les couches de convolution dilatées sont définies de la manière

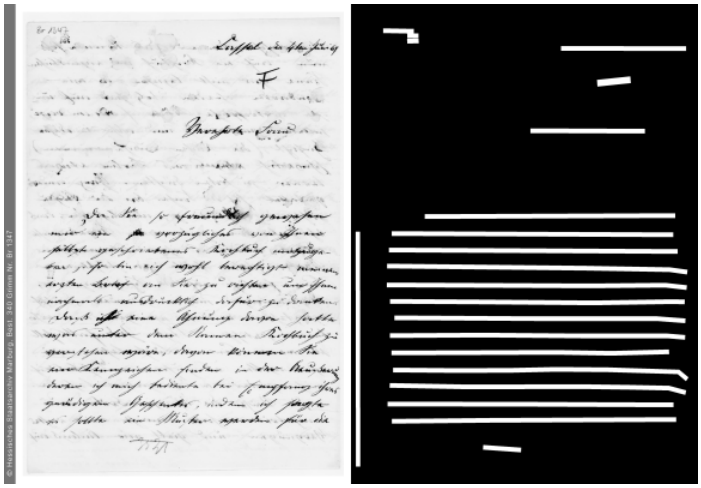

Figure 2.14: Exemple d'un étiquetage "X-height"

suivante.

Soit x de dimension $H \times W \times D^I$, le signal d'entrée d'une couche convolutionnelle, où H, W et D^I sont respectivement la hauteur, la largeur et le nombre de canaux de l'image. Soit f le filtre pondéré (le filtre convolutionnel) de taille $H^f \times W^f \times D^I$. Pour préserver la taille de l'image d'entrée, un stride (décalage) $s = 1$ est considéré, et un complément de l'entrée avec des colonnes et des lignes de 0 est effectué ($\lfloor \frac{H^f-1}{2} \rfloor$ lignes en haut, $\lfloor \frac{H^f}{2} \rfloor$ lignes en bas, $\lfloor \frac{W^f-1}{2} \rfloor$ colonnes à gauche et $\lfloor \frac{W^f}{2} \rfloor$ colonnes à droite). À partir d'une entrée x paddée, le calcul d'une convolution standard est obtenu par :

$$y[i,j,d^o] = \sum_{k=0}^{H^f} \sum_{l=0}^{W^f} \sum_{d=0}^{D^I} f[k,l,d,d^o] \times x[i+k, j+l, d] \tag{2.13}$$

où d^o est l'indice du canal de sortie, et i et j sont les indices de ligne et de colonne, tels que $\lfloor \frac{H^f-1}{2} \rfloor \leqslant i \leqslant H$ et $\lfloor \frac{W^f-1}{2} \rfloor \leqslant j \leqslant W$.

Pour la convolution dilatée, on définit un terme supplémentaire r appelé taux de dilatation, correspondant à l'échelle du filtre. En considérant un filtre de convolution de taille $H^f \times W^f \times D^I$ comme ci-dessus, la convolution est appliquée sur des fenêtres de hauteur $\tilde{H}^f = H^f + (H^f - 1) \times (r - 1)$ et de largeur $\tilde{W}^f = W^f + (W^f - 1) \times (r - 1)$ dans l'image. L'image d'entrée est donc complétée en ajoutant $\lfloor \frac{\tilde{H}^f-1}{2} \rfloor$ lignes au dessus, $\lfloor \frac{\tilde{H}^f}{2} \rfloor$ lignes au dessous, $\lfloor \frac{\tilde{W}^f-1}{2} \rfloor$ colonnes à gauche et $\lfloor \frac{\tilde{W}^f}{2} \rfloor$ colonnes à droite. Comme pour l'équation 2.13, la sortie de la convolution dilatée s'exprime comme suit :

$$y[i,j,d^o] = \sum_{k=0}^{H^f} \sum_{l=0}^{W^f} \sum_{d=0}^{D^I} f[k,l,d,d^o] \times x[i+r \times k, j+r \times l, d] \tag{2.14}$$

où d^o est l'indice du canal de sortie, $\lfloor \frac{\tilde{H}^f - 1}{2} \rfloor \leqslant i \leqslant H$ et $\lfloor \frac{\tilde{W}^f - 1}{2} \rfloor \leqslant j \leqslant W$.

Nous avons proposé deux architectures totalement convolutives à 7 et à 11 couches. L'architecture à 7 couches, illustrée en figure 2.15, est inspirée de l'architecture VGG16 [88] dont les paramètres des 6 premières couches convolutives ont été repris. Des facteurs de dilatation respectifs de (1,1,2,2,4,4) remplacent les couches de pooling. L'architecture est complétée par une couche de sortie avec un taux de dilatation de 1 et des filtres de taille 1×1. L'architecture à 11 couches reprend l'architecture à 7 couches, dans laquelle s'insèrent entre la 6ème et la 7ème couche 4 couches convolutionnelles supplémentaires avec des taux de dilatation de (2,2,1,1).

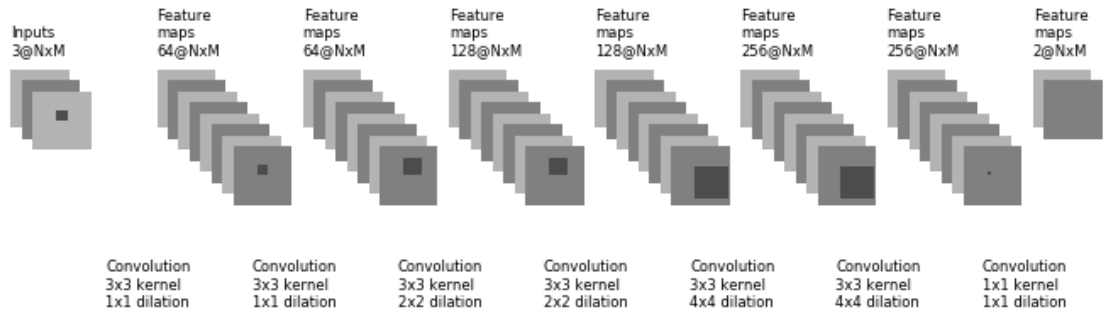

Figure 2.15: Notre architecture FCN pour la segmentation de documents en lignes. La résolution reste constante tout au long de l'architecture, et les facteurs de dilatation successifs permettent agrandir les champs réceptifs.

2.2.4.4 Expériences et résultats

Nous avons participé à la compétition de segmentation en lignes cBAD [7] de la conférence ICDAR 2017. Il s'agissait de documents provenant de 7 différentes archives, répartis en 216 images de documents pour l'apprentissage et 539 documents pour le test. Nous avons réservé 40 images de l'ensemble d'apprentissage pour la validation. Les métriques utilisées sont celles de la compétition cBAD [21] : précision, rappel et F-mesure.

Nous évaluons les trois types de FCN (déconvolution, unpooling, convolutions dilatées) pour la segmentation de documents en lignes de texte, ainsi que l'influence du nombre de couches. Nous proposons enfin quelques améliorations à l'architecture, fournissant un gain significatif en performances.

7. https ://scriptnet.iit.demokritos.gr/competitions/5/

Table 2.6: Résultats obtenus sur la base de test de la compétition cBad pour 4 types de FCN : déconvolution, unpooling, déconvolution/unpooling, et convolutions dilatées.

Méthode	nombre de couches	Précision	Rappel	F-mesure
Déconvolution	7	62.2	76.5	68.6
	11	**72.7**	83.9	77.9
Unpooling	7	56.6	74.7	64.4
	11	72.5	83.4	77.6
Déconv + Unpool	7	71.3	47.2	56.8
	11	72.1	84.0	77.6
Convolutions dilatées	7	70.8	82.3	76.1
	11	72.2	**85.5**	**78.3**

Comparaison des trois types de FCN

Nous avons comparé les trois types de FCN ainsi qu'une combinaison à base de déconvolution et d'unpooling qu'on retrouve parfois dans la littérature.

Pour avoir une comparaison valide, nous avons utilisé des architectures dont les tailles des champs réceptifs et le nombre de filtres étaient similaires. Nous renvoyons à [7] pour le détail des architectures à 7 et 11 couches pour les quatre types de FCN.

Chaque réseau a été entraîné sur l'ensemble d'apprentissage de la compétition cBAD jusqu'à convergence de la base de validation. Le meilleur réseau sur la base de validation a été sélectionné pour soumission à la compétition. Les résultats bruts sans post traitements sont présentés dans le tableau 2.6.

On peut constater que les convolutions dilatées obtiennent généralement les meilleurs résultats, que ce soit pour l'architecture à 7 ou à 11 couches. De même, les architectures à 11 couches proposent de meilleures performances, au prix d'un nombre supérieur de paramètres. L'architecture basée sur les convolutions dilatées passe ainsi de 1.15M de paramètres pour 7 couches à 1.70M pour 11 couches, soit un facteur 1.5.

Amélioration des résultats

Nous avons proposé quelques améliorations à la méthode proposée afin de maximiser le score sur la compétition cBAD.

Premièrement, nous avons effectué un transfert d'apprentissage en utilisant des données

Table 2.7: Performances des meilleurs systèmes à la compétition cBAD.

Participant	Précision	Rappel	F-mesure
DMRZ	97.3	97.0	97.1
Approche proposée (11 couches)	**94.9**	**88.1**	**91.3**
Approche proposée (7 couches)	**89.7**	**89.9**	**89.8**
UPVLC	93.7	85.5	89.4
BYU	87.8	90.7	89.2
IRISA	88.3	87.7	88.0

provenant de la compétition READ [8], une autre compétition ICDAR sur la reconnaissance d'écriture, mais qui donnait l'annotation des lignes de texte sur les documents. Ces documents supplémentaires (8000 en apprentissage, 2000 en validation) ont été utilisés pour pré-apprendre le FCN. Ce préapprentissage de l'architecture proposé a permis d'obtenir une F-mesure de 86.25%, contre 76.1% initialement pour l'architecture à 7 couches. Ce gain très significatif confirme l'intérêt du transfert d'apprentissage sur la généralisation des modèles, comme déjà observé dans la section 2.2.3.

Nous avons également procédé à une optimisation du seuil de décision en sortie du FCN pour décider des classes lignes de texte/fond. Cela permet de rééquilibrer la décision dans le cas où une différence de balance entre les deux classes est observée entre l'apprentissage et le test. Nous avons finalement ajouté une étape de post-traitement consistant à regrouper des petites lignes pour éviter certains problèmes de sous segmentation qui subsistaient.

Résultats finaux sur la base de la competition cBAD

Les résultats finaux des architectures à 7 et 11 couches avec toutes les améliorations proposées (transfert d'apprentissage, seuil optimisé, post-traitements) obtenus sur les données de la compétition cBAD sont présentés dans le tableau 2.7.

L'approche proposée basée sur les FCN avec convolutions dilatées se classe deuxième. Il est intéressant de constater que le participant DMRZ a utilisé une architecture profonde de type U-net, combinée avec des post-traitements optimisés pour chacune des 7 archives dont les documents étaient extraits, ce que les autres participants n'ont pas fait. L'approche proposée par BYU était à base de FCN à 10 couches avec déconvolutions, tandis que les approches de l'UPVLC et de l'IRISA étaient des approches ad-hoc.

8. https ://scriptnet.iit.demokritos.gr/competitions/ icdar2017htr/

2.2.4.5 Conclusion

Nous avons proposé dans cette contribution l'application d'une variante des architectures totalement convolutives sur une problématique d'analyse d'images de documents. De par leur conception, les FCN poussent à son paroxysme la notion de convolution, et permet l'apprentissage d'architectures particulièrement légères qui s'adaptent aux signaux de tailles variables.

Une perspective intéressante à ce travail est l'utilisation de filtres multi-résolution afin de gérer les objets de différentes tailles, ici la variabilité dans les styles d'écriture. On peut pour cela facilement regrouper plusieurs filtres identiques mais avec des facteurs de dilatation différents. Le nombre de paramètres devrait s'en trouver une nouvelle fois réduit.

2.2.5 Conclusion sur la régularisation des architectures profondes

À travers les contributions présentées dans cette section, nous avons exploré un certain nombre de méthodes pour la régularisation d'architectures profondes : régularisation explicite pour les problèmes à sorties structurées, transfert d'apprentissage, architectures convolutionnelles et totalement convolutionnelles. Dans la littérature, on observe souvent une combinaison de plusieurs approches pour la régularisation des modèles. Mais il est connu que certaines d'entre elles sont parfois incompatibles ou déconseillées (Batch norm./dropout, LSTM/ReLU, etc.). On revient alors à un problème de recherche d'hyper-paramètres, particulièrement difficile pour des modèles dont les apprentissages sont longs. On observe également dans la littérature récente une course à l'amaigrissement des couches, en utilisant par exemples des convolution factorisées ou les "depthwise separable convolutions" [16]. Selon nous, deux pistes importantes pourraient également permettre d'alléger le besoin de régulariser les modèles.

La première piste consiste à utiliser des données non étiquetées et donc à développer les approches non supervisées. En effet, de telles données sont souvent disponibles à faible coût et en grande quantité. Nous avons encore peu abordé cette problématique mais elle constitue certainement une perspective prometteuse.

La deuxième piste permettant de limiter le nombre d'exemple nécessaire à l'apprentissage d'architectures profondes est l'intégration de connaissances à priori dans les modèles. En effet, suivant les problèmes certaines règles simples valent mieux qu'un grand nombre d'exemples. Nous avons exploré cette deuxième piste dans le cadre de la reconnaissance d'écriture manuscrite, où les connaissances à priori liées à la langue sont fondamentales.

2.3 Apprentissage profond et Reconnaissance de l'écriture

2.3.1 Introduction

La reconnaissance d'écriture a largement bénéficié des progrès en machine learning ces dernières années, notamment grâce aux réseaux de neurones récurrents. Contrairement à leurs éditions précédentes, lors les conférences ICDAR 2017 [9] et ICFHR 2018 [10], une très grande majorité des articles proposait des approches exploitant les architectures profondes. Pour autant, la reconnaissance d'écriture manuscrite est un domaine à part entière, nécessitant la prise en compte des contraintes linguistiques qui sont primordiales pour obtenir des systèmes de reconnaissance exploitables. Ces connaissances sont principalement des lexiques et des modèles de langue, permettant de filtrer et/ou de réévaluer les hypothèses de reconnaissance du modèle optique. Aujourd'hui, les moteurs de reconnaissance optique se sont considérablement améliorés, et atteignent des performances très élevées, probablement proches des performances humaines. En revanche, la prise en compte des connaissances à priori n'est pas résolue, et la question de l'apprentissage conjoint du modèle optique et du modèle de langue est encore ouverte.

Dans cette section nous présentons les contributions concernant la reconnaissance d'écriture à l'aide de réseaux de neurones profonds qui ont été proposées dans les thèses de Simon Thomas, Gautier Bideault, Luc Mioulet et Bruno Stuner. Après avoir donné un aperçu des méthodes à l'état de l'art pour la reconnaissance d'écriture (2.3.2), nous proposons deux thèmes de recherche ayant trait à la prise en compte des informations linguistiques : l'extraction d'information (section 2.3.3), et la reconnaissance d'écriture avec des très grands lexiques (section 2.3.4).

2.3.2 Reconnaissance d'écriture

La reconnaissance d'écriture manuscrite est l'opération qui consiste à transcrire une image contenant du texte en chaine de caractères. Débutée dans les années 60, la reconnaissance d'écriture a connu ses premiers succès dans les années 90 avec les applications de reconnaissance des montants de chèques [218, 192] et de reconnaissance de blocs adresse [217, 219, 180] pour le tri postal. Aujourd'hui l'avènement de l'apprentissage profond a permis d'améliorer grandement la reconnaissance optique, permettant le traitement de documents complexes, parfois dégradés avec des lexiques importants. De nouvelles applications telles que l'extraction d'information [159, 129, 135] ou le traitement de documents historiques [45, 82] sont désormais opérationnelles. La figure 2.16 montre l'évolution des bases publiques annotées, où l'on constate bien l'augmentation de la difficulté.

9. http://u-pat.org/ICDAR2017/index.php

10. http://icfhr2018.org/

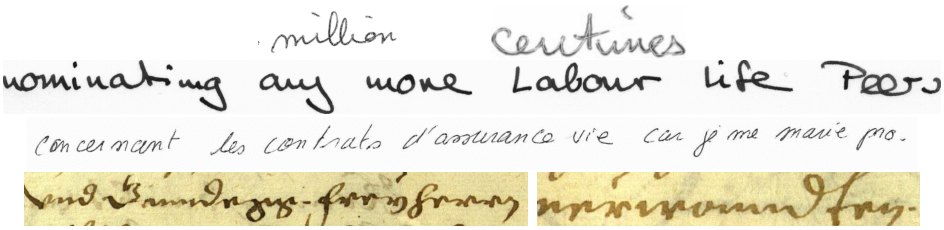

Figure 2.16: Mots isolés et lignes de texte issus des bases publiques Ironoff [194] (1999), IAM [178] (2002), Rimes [157](2006) et READ [27] (2017). On observe la difficulté croissante pour ces quatre bases, principalement due *i)* au contenu (mots isolés vs. lignes de texte), *ii)* aux conditions de numérisation, *iii)* à la graphie plus ou moins complexe et *vi)* à la disponibilité des ressources linguistiques permettant de contraindre la reconnaissance.

Les difficultés liées à la reconnaissance d'écriture manuscrite sont nombreuses : variabilité intrinsèque de l'écriture (style d'écriture, écriture cursive ou scripte, variabilité de taille et d'inclinaison, etc.), numérisation et/ou supports imparfaits (luminosité trop faible/trop importante, transparence du support, usure du temps, tâches, etc.), ressources linguistiques pas toujours disponibles (vocabulaire ouvert, langue potentiellement inconnue, entitées nommées, etc.).

D'une manière générale, la reconnaissance consiste à déterminer la séquence de caractères ou de mots W à partir d'une séquence d'observation O. La meilleure séquence de sortie $\hat{W}$ est celle qui maximise la probabilité $P(W|O)$, qui peut être réécrite à l'aide du théorème de Bayes :

$$\hat{W} = \underset{W}{\operatorname{argmax}}\, P(W|O) = \underset{W}{\operatorname{argmax}}\, \frac{P(O|W) \times P(W)}{P(O)}$$

Le terme $P(O)$ étant constant pour tous les W, on peut réécrire la maximisation :

$$\hat{W} = \underset{W}{\operatorname{argmax}} \underbrace{P(O|W)}_{\textit{estime par le modele optique}} \times \underbrace{P(W)}_{\textit{estime par le modele de langue}} \tag{2.15}$$

Les systèmes de reconnaissance d'écriture reposent ainsi généralement sur deux étapes[187] permettant d'estimer les deux termes de l'équation 2.15 : le reconnaissance optique de caractères et le traitement linguistique. La reconnaissance optique de caractère consiste à estimer les hypothèses de reconnaissance $P(O|W)$ en se basant uniquement sur la forme des caractères. Le traitement linguistique cherche lui à estimer les probabilités $P(W)$ d'apparition de la séquence de sortie. Elle permet ainsi de réestimer les hypothèses de reconnaissance optique en prenant en compte les connaissances linguistiques à priori afin de fiabiliser la reconnaissance.

2.3.2.1 Reconnaissance optique de caractères

Historiquement, les premiers systèmes de reconnaissance ont cherché à reconnaître des caractères isolés [181], à l'aide de caractéristiques [208] et d'un classifieur statique de type KPPV, réseau de neurones ou SVM [174]. Ces techniques ont ensuite été étendues pour reconnaître des mots cursifs, avec la difficulté supplémentaire liée à la segmentation des mots en caractères. Pour cela des méthodes à segmentation dite explicite ont été mises en places, visant à identifier puis évaluer différents points de segmentation à l'aide d'un classifieur de caractères et d'un algorithme de programmation dynamique [214, 201]. Ces méthodes ont progressivement été remplacées par des modèles de Markov cachés (Hidden Markov Model ou HMM), réalisant une segmentation implicite à l'aide d'une fenêtre glissante [205, 124, 144, 168]. Dans le milieu des années 90 les approches "neuro-markoviennes" ont vu le jour, combinant les capacités modélisantes des HMM avec les capacités discriminantes des méthodes neuronales de type MLP, RNN [207] ou CNN [197]. Des algorithmes d'apprentissage "bout-en-bout" ont même été proposés, permettant d'obtenir à l'époque les résultats à l'état de l'art [209, 196, 184]. Aujourd'hui, les méthodes reposant sur l'apprentissage profond sont incontournables. Pour les caractères isolés les CNN produisent naturellement les meilleurs résultats, et leur combinaison avec des réseaux récurrents constituent la méthode de référence pour la reconnaissance de mots isolés [142, 143], de lignes [50, 86, 92] ou même de paragraphes [13]. Comme pour leurs homologues statiques (MLP, CNN), les réseaux de neurones récurrents (Recurrent Neural Networks, RNN) reposent sur un principe fondateur ancien [230], mais ont bénéficié des immenses progrès des algorithmes d'apprentissage ces dernières années.

Réseaux de neurones récurrents

Un réseau de neurone récurrent est un réseau de neurone dynamique dédié au traitement de séquences, observées à travers une fenêtre glissante par pas de temps t. Leur nature dynamique réside dans la présence de connexions récurrentes, qui réinjectent les sorties d'une couche à l'instant $t-1$ en entrée de celle ci à l'instant t.

Mis au point dès les années 80 [230, 229], les RNN ont eu un succès limité, principalement à cause des difficultés à contrôler leur apprentissage. En effet, les connexions récurrentes les rendent particulièrement sujets au problème de disparition du gradient (voir [195] et section 2.2.1), mais aussi à l'explosion gradient [106]. Comme pour les architectures non récurrentes, ces problèmes ont été bien identifiés et des solutions efficaces ont été trouvées ces dernières années pour contrôler le gradient à travers un "clipping" [119], par les techniques de régularisation classiques ℓ_1 et ℓ_2 (voir section 2.2) ou plus spécifiques comme privilégier les paramètres qui n'induisent pas une variation trop forte du gradient [106]. Une autre solution efficace pour lutter contre la disparition du gradient est l'utilisation de cellules Long short term memory (LSTM) [200].

Les LSTM sont des neurones particuliers qui contiennent une mémoire (souvent appelée

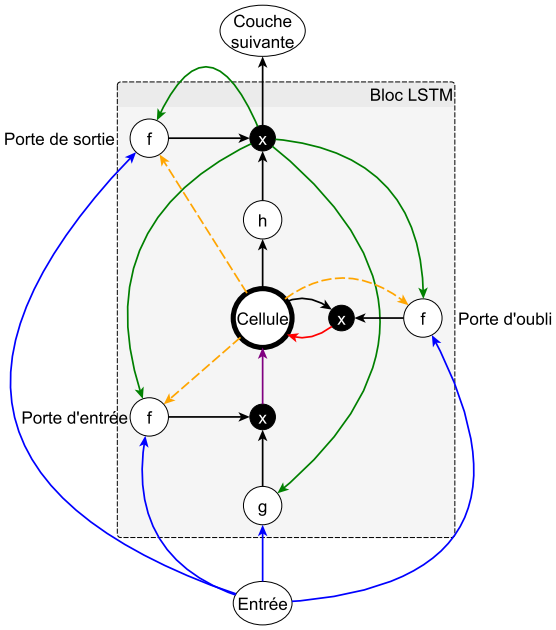

Figure 2.17: Modélisation d'une cellule LSTM issue de la thèse de Bruno Stuner [8].

Cell) dont le contenu est mis à jour à travers trois portes de contrôle (Voir Figure 2.17). Les portes d'entrée, d'oubli et de sortie permettent respectivement d'autoriser l'action de l'entrée sur la mémoire, de remettre la mémoire à zéro, et d'autoriser l'action de la mémoire sur la sortie. Ces trois actions sont modélisées par des sigmoïdes dont les paramètres sont appris par rétropropagation du gradient. Contrairement aux RNN classiques, la présence d'une mémoire explicite permet une mémorisation de signaux beaucoup plus longue, pouvant aller jusqu'à un millier de pas de temps selon les problèmes. En revanche, les cellules LSTM possèdent un nombre important de paramètres, ce qui invite à la parcimonie.

En 2006, Alex Graves a proposé l'algorithme Connectionist Temporal Classification [162], version connectionniste de l'algorithme forward backward des HMM, permettant l'apprentissage de RNN sans nécessiter de séquences étiquetées au niveau fenêtres. En 2009, Alex Graves remporte haut la main la compétition Rimes 2009 pour la reconnaissance de mots manuscrits [143], avec une version bidirectionnelle des LSTM (BLSTM) [141]. Une version 2D des LSTM a également été proposée [142], afin de capturer les dépendances dans les deux directions du signal écriture. Bien que cette architecture MDLSTM permettent d'apprendre à la fois la représentation des données et la modélisation dynamique des séquences, on leur préfère aujourd'hui des architectures plus légères à base de CNN et LSTM. L'une des raisons avancées dans [25] est que la modélisation des dépendances verticales n'est pas suffisamment utile pour lui dédier des cellules LSTM.

La combinaison CNN/LSTM (typiquement une dizaine de couches convolutives associées à 1

ou 2 couches de LSTM [32, 25, 18]) constitue aujourd'hui un outil idéal pour la reconnaissance optique de caractère. Elle permet l'apprentissage conjoint d'un outil de perception efficace et d'un modéle dynamique de décision dédié au signal complexe d'écriture.

Il reste néammoins de nombreux cas où le meilleur modèle optique est insuffisant, et où des connaissances linguistiques sont indispensables pour effectuer la bonne prédiction. Reprenons par exemple le dernier exemple de la figure 2.16, où un lecteur sans connaissance en haut allemand sera incapable de déchiffrer le moindre caractère.

2.3.2.2 Prise en compte des ressources linguistiques

Aussi performant soit il, un moteur de reconnaissance optique de caractère pourra toujours être mis en défaut par une information visuelle insuffisante. Le but des ressources linguistique est de réévaluer les hypothèses de reconnaissance du moteur optique au regard de connaissances à priori.

Les deux types d'approche permettant ces corrections linguistiques sont les méthodes dirigées par le lexique et les modèles de langage. Ces deux méthodes sont basées sur la recherche de la meilleure solution au regard d'un lexique donné.

Décodage dirigé par le lexique

Un décodage dirigé par le lexique permet de contraindre les sorties du système de reconnaissance à appartenir à un lexique de mots donné. Pour cela, un alignement entre les sorties probabilisées du reconnaisseur optique et les différents mots du lexique est généralement effectué par une recherche en faisceau de Viterbi [227, 221].

S'il permet de corriger un certain nombre d'erreurs de reconnaissance, le lexique doit couvrir l'ensemble des mots susceptibles d'être rencontrés. Dans le cas contraire, un mot dont la reconnaissance optique est correcte sera confondu avec son entrée la plus proche dans le lexique. Le choix du lexique n'est donc pas anodin. Dans un environnement suffisamment contrôlé, il est facile de constituer un lexique adapté à la tâche de reconnaissance (reconnaissance des mois d'une date ou du montant litéral d'un montant de chèque par exemple). Dans un environnement plus libre, il faut souvent plusieurs millions de mots pour obtenir une couverture suffisante et limiter le nombre de mots hors vocabulaire (on parle d'OOV pour Out-Of-Vocabulary) qui ne pourront être reconnus. Il y a par exemple 336531 mots dans le dictionnaire français Gutenberg et plus de 3 millions de mots différents sur les pages françaises de Wikipedia et du Wiktionnaire [8]. L'augmentation de la taille des lexiques pose deux problèmes bien connus [173]. Tout d'abord, l'utilisation d'un très grand lexique demande davantage de calculs pour le décodage. Deuxièmement, un lexique trop grand est moins contraignant et donc moins efficace pour la correction des erreurs de reconnaissance optique.

La constitution du lexique doit alors se faire sous la double contrainte de maximisation du taux de couverture, et de minimisation de la taille du lexique. Bien sûr, dans les environnements non contrôlés, ces taux sont difficiles à déterminer.

Afin de traiter plus efficacement les grands lexiques, il existe de nombreuses modélisations dont la plus utilisée est le transducteur à états finis [206] (ou Finite State Transducer, FST). Issu de la théorie des automates, cet outil modélise les transformations nécessaires au passage d'un alphabet d'entrée (les hypothèses de caractères) vers un alphabet de sortie (les mots du lexique). Bien que permettant d'accélerer fortement la recherche du meilleur mot dans le lexique, ces outils ne permettent pas de compenser le manque de contraintes d'un lexique trop grand.

Modélisation de la langue

Une autre manière de bénéficier de ressources linguistiques consiste à modéliser le langage de manière probabliste, souvent sur un corpus externe conséquent.

L'intérêt de ces modèles est d'estimer les séquences de mots ou de caractères les plus probables de manière à rescorer les probabilité du modèle optique selon l'équation 2.15. La probabilité d'observer la séquence $W = w_i, ..., w_m$ peut être estimée par l'équation 2.16 :

$$P(W) = P(w_i, \ldots, w_m) = \prod_{i=1}^{m} P(w_i | w_1, \ldots, w_{i-1}) \tag{2.16}$$

Ces probabilités sont cependant difficiles à estimer, y compris sur des corpus importants. La solution communément employée est de limiter l'historique à des séquences de n caractères ou mots appelés n-grams [235]. La modélisation du langage est ainsi réduite à la modélisation des enchaînement de mots ou de caractères selon une hypothèse de Markov d'ordre n. L'estimation de la probabilité $P(w_i, ..., w_m)$ est alors approximée par l'équation :

$$P(w_i, \ldots, w_m) = \prod_{i=1}^{m} P(w_i | w_{i-n+1}, \ldots, w_{i-1}) \tag{2.17}$$

En décision, les n-grams les plus probables feront ainsi remonter les hypothèses issues de la reconnaissance optique. L'apprentissage consiste à estimer les probabilités des n-grams, qui est généralement réalisé sur des corpus plus grands que les corpus dédiés à la reconnaissance optique. En effet, une estimation par un simple décompte des observations conduit à une majorité de probabilités nulles, notamment pour des valeur de n élevées. De nombreuses méthodes existent pour solutionner ce problèmes, telles que les modèles à base de classes de mots [220] ou les techniques de lissage [204] dont les plus utilisées sont les méthodes de replis.

Une alternative consiste à estimer les probabilités des n-grams à l'aide de réseaux de neurones

récurrents [172]. Pour cela, il est nécessaire de plonger les mots dans un espace numérique de type *word2vec* [104, 105]. Malgré des problèmes de complexité de calcul liés à la normalisation [108], ce type d'approche tend à se généraliser aussi bien en reconnaissance d'écriture [92] qu'en reconnaissance de la parole [80].

2.3.2.3 conclusion

Dans ce bref panorama de la reconnaissance d'écriture, nous avons constaté les progrès récents considérables des systèmes de reconnaissance optique de caractères. Sur des bases difficiles telles que READ [27], certains modèles dépassent les performances humaines. Inversement, un humain confronté à sa langue native est toujours capable de déchiffrer des signaux très dégradés bien mieux qu'une machine. Il nous semble qu'il existe un décalage important entre la maturité des systèmes optiques et la difficulté des systèmes actuels à prendre en compte des ressources linguistiques externe, notamment dans le cas de très grands lexiques. Dans la suite de cette section, nous proposons plusieurs contributions liées à la prise en compte des ressources linguistiques externes.

Dans la première partie (section 2.3.3), nous posons la question de l'utilité d'une reconnaissance intégrale de documents dans un contexte à très grand lexique, et proposons une approche orientée extraction d'information.

Dans la seconde partie (section 2.3.4), nous proposons un nouveau paradigme pour la reconnaissance d'écriture avec des lexiques gigantesques, de l'ordre de plusieurs millions de mots. Ce nouveau paradigme remplace la reconnaissance dirigée par le lexique par une vérification lexicale combinée à des cohortes de réseaux de neurones récurrents.

2.3.3 Extraction d'information dans des documents manuscrits

2.3.3.1 Contexte des travaux

Cette section présente des travaux effectués dans la continuité de ma thèse, lors des thèses de Simon Thomas avec la société EMC (2008-2012), puis de la thèse de Gautier Bideault (2011-2015) et du post doctorat de Yousri Kessentini (2013), tous les deux avec la société Itesoft.

Le contexte de ces travaux est le traitement automatique du courrier entrant, reçu en grande quantité par des grands goupes. Ces courriers manuscrits constituaient encore à l'époque un moyen important de communication entre des clients et une entreprise ou une administration, afin de notifier un changement d'adresse, une réclamation, une demande de résiliation de contrat, etc.

L'automatisation du traitement de ces courriers requiert une lecture intégrale du texte,

suivie de traitement de plus haut niveau tels que l'identification et l'extraction des entités nommées, la catégorisation des documents, etc. Il est vite apparu que cette enchaînement de traitements poserait deux principaux problèmes que sont la reconnaissance d'écriture à très grand lexique pour reconnaître les entitées nommées, et l'application des traitements de plus haut niveau dans un contexte bruité à cause des erreurs de reconnaissance. Lors de ma thèse et dans les travaux ultérieurs présentés dans cette partie, nous avons proposé une alternative à la reconnaissance intégrale des documents, basée sur l'extraction d'information. L'idée générale est d'effectuer une extraction directe de l'information pertinente telles que des séquences numériques, un lexique de mots-clefs prédéfini ou encore des entitées nommées. Pour cela, nous avons conçu des modèles inspirés de ceux généralement utilisés en recherche d'information, que nous avons adapté pour prendre en compte les incertitudes liées aux erreurs de reconnaissance.

Nous avons décliné cette stratégie dans plusieurs travaux entre 2008 et 2015 :

— L'extraction de séquences numériques telles que des numéros de téléphone, des codes postaux, etc. a été abordée lors de ma thèse [161] et ne sont pas détaillés dans ce document (voir aussi les travaux [160, 147, 159]).

— Dans la thèse de Simon Thomas [121], nous avons proposé l'extraction de mots clefs d'un lexique prédéfini à l'aide de modèles d'extraction d'information génériques basés sur des HMM. Nous y avons intégré des réseaux de neurones profonds préappris à l'aide d'auto-encodeurs. Ces travaux, publiés dans [73, 109, 122, 136, 135, 134], sont présentés dans la section 2.3.3.2.

— Dans la thèse de Gautier Bideault [54], nous avons étendu ces modèles pour permettre l'extraction d'expressions régulières, et nous y avons intégré des LSTM afin d'améliorer les performances [55, 56, 57]. Cette contribution est décrite dans la section 2.3.3.4.

2.3.3.2 Extraction de mots clefs d'un lexique prédéfini

En 2010 lors de la thèse de Simon Thomas, nous avons proposé l'un des premiers modèles d'extraction d'information pour des images de documents, basé sur l'extraction de mots clefs d'un lexique prédéfini. Il s'agit de détecter et à reconnaître des mots "requête" d'un lexique dans des images de documents [136, 114, 113].

Cette extraction peut servir à interroger une base de documents, dans ce cas le lexique n'est composé que d'un mot. Pour d'autres tâches telles qu'une catégorisation [171], une indexation ou à l'identification d'entités nommées, il convient de définir un lexique adapté. Nous avons par exemple proposé ultérieurement une méthode de catégorisation de documents manuscrits à l'aide d'un lexique d'environ 400 mots clefs [120]. À travers le lexique de mots pertinents, le problème de reconnaissance basée sur un très grand lexique est contourné.

Pour mettre en œuvre la tâche d'extraction de mots clefs, nous avons proposé un modèle de ligne de texte mettant en compétition l'information pertinente (les mots clefs du lexique)

avec l'information non pertinente (tous les mots hors lexique, la ponctuation, etc.). Nous avons initialement proposé d'implémenter le modèle de ligne de texte à l'aide de modèles de Markov cachés [136, 135, 134], que nous avons ensuite augmenté d'un réseau de neurones profond pour proposer un modèle neuro-markovien [73, 109].

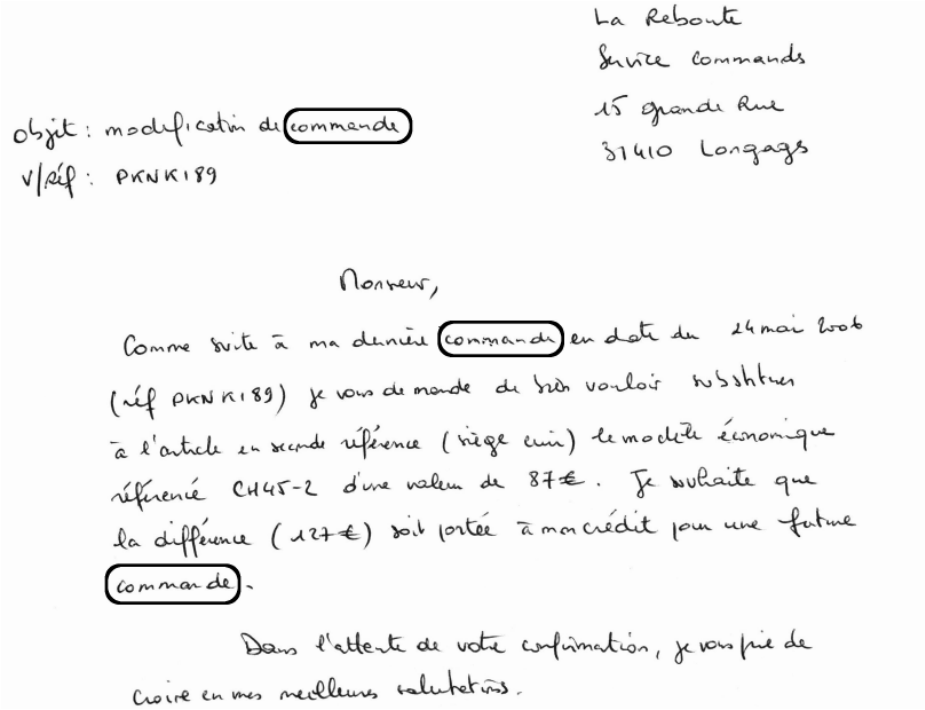

Figure 2.18: Exemple d'extraction du mot "commande" dans un courrier manuscrit.

Modèle de ligne de texte générique pour l'extraction de mots clefs

En considérant qu'on soit capable d'identifier les lignes de texte d'un document (par exemple avec un FCN, voir section 2.2.4), le modèle d'extraction doit décider de la présence dans la ligne des mots clefs du lexique. Dès lors, il est nécessaire de concevoir un modèle de ligne de texte capable de mettre en compétiion un modèle des mots du lexique avec un modèle du reste de la langue. Afin de modéliser cette deuxième partie, nous avons proposé d'utiliser un modèle dit ergodique, autorisant tous les enchaînements de caractères de la langue considérée. Ce type de modèle, aussi appelé modèle de remplissage, est bien connu des chercheurs en recherche d'information, notamment sur des signaux sonores [190, 167]. Le modèle de ligne que nous avons proposé pour l'extraction de mots clefs est ainsi présenté en figure 2.19.

Le *lexique de mots clefs (KL)* est représenté par un modèle parallèle des mots du lexique qui sont constitués de leur séquence de caractères. Tous les mots ont la même probabilité d'apparition.

L'ensemble des *mots hors lexique (OKL)* est représenté par un modèle ergodique autorisant toute les transitions entre les symboles de la langue (ici le francais), incluant les caractères

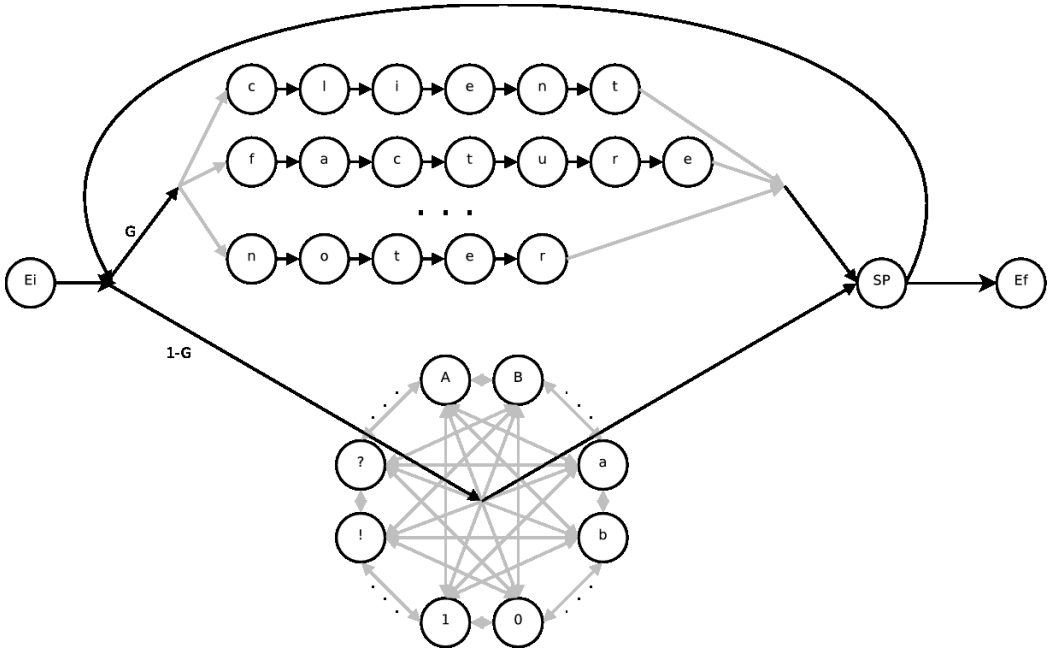

Figure 2.19: Modèle de ligne de texte générique pour l'extraction de mots clefs. La partie haute modélise les mots clefs du lexique (KL : keyword lexicon) ; la partie basse modélise les mots hors lexiques (OKL : Out of Keyword Lexicon. Le modèle contient également un modèle d'espace ("SP"), un début et une fin de ligne ("Ei" et "Ef").

minuscules, majuscules, les chiffres et symboles de ponctuation.

Le modèle doit être capable de décrire les enchaînements de mots (KL ou OKL) séparés par des espaces inter-mots. Le modèle contient donc un modèle d'espace ("SP"), un début et une fin de ligne ("Ei" et "Ef"). La compétition entre les modèles KL et OKL est arbitrée par un paramètre G, qui est supposé représenter la probabilité d'observer un mot du lexique.

La manière la plus naturelle d'implémenter ce modèle de séquence probabiliste est l'utilisation des modèles de Markov cachés. L'apprentissage du modèle complet consiste à apprendre les modèles de caractères et les probabilités de transitions entre caractères. Les modèles de caractères sont des modèles gauche-droite classiques à 4 états, dont les vraisemblances sont estimées soit par des gaussiennes, soit calculées a partir des probabilités a posteriori d'un réseau de neurones. Les probabilités de transition entre caractères dans le modèle KL sont toutes à 1, comme dans n'importe quel décodage dirigé par le lexique. Les transitions du modèle de surface OKL doivent être représentatives du langage considéré et sont donc estimées à partir d'un lexique général.

Architecture neuro-markovienne

Considérons $O = \{o_1 \dots o_T\}$ une séquence d'observation de taille T extraite d'une ligne de texte W. Comme vu en section 2.3.2, la recherche de la meilleure séquence $\hat{W}$ se fait par la

maximisation de l'équation :

$$\hat{W} = \underset{W}{\text{argmax}}\{P(O|W)P(W)\} \tag{2.18}$$

où le premier terme est calculé par le modèle optique, et le second terme peut être estimé par un modèle de langage. Dans ces travaux, nous avons simplement considéré toutes les séquences équiprobables et avons donc ignoré le deuxième terme.

Considérons également que la ligne W contient N mots $\{w_1, \ldots, w_N\}$ du lexique, M mots hors lexique $\{w_1, \ldots, w_M\}$ et P espaces S. étant donné notre modèle, on peut ainsi trouver la meilleure séquence en maximisant la quantité suivante sur toutes les hypothèses de segmentation/reconnaissance :

$$\hat{W} = \text{argmax}\left\{ G^N \times \prod_{n=1}^{N} P(o_n|w_n) \times (1-G)^M \times \prod_{m=1}^{M} P(o_m|w_m) \times \prod_{p=1}^{P} P(o_p|s_p) \right\} \tag{2.19}$$

où les $P(o|w)$ (respectivement $P(o_n|w_n)$, $P(o_m|w_m)$ and $P(o_p|s_p)$) sont les vraisemblances d'observer la sequence o (resp. o_n, o_m and o_p) sachant la séquence de caractère w (resp. w_n, w_m and s_p). Dans les HMM, ces vraisemblances sont calculées au niveau trame en utilisant l'hypothèse de Markov :

$$P(o|w) = \prod_{t=1}^{T} P(o_t|q_i)P(q_i|q_{i-1}) \tag{2.20}$$

Où les $\{q_i\}$ sont les états du mot w, et T est le nombre d'observations. Finalement, la meilleure séquence d'étiquette $\hat{W}$ est calculée avec l'algorithme "Time Synchronous Beam Search" [202], en maximisant la quantité $P(O|W)P(W)$ sur toute la ligne. Cette quantité dépend :

- du paramètre G du modèle et des contraintes lexicales modelisées par les probabilités de transitions entre états $P(q_i|q_{i-1})$;
- de la capacité de discrimination locale du modèle $P(o_t|q_i)$.

Concernant l'estimation des quantités $P(o_t|q_i)$, celles ci ont été dans un premier temps calculées par des gaussiennes, comme c'est classiquement le cas dans les HMM continus. Dans un second temps, nous avons remplacé les gaussiennes par des réseaux de neurones. Dans ce second cas, les probabilités a posteriori des réseaux de neurones sont transformées en vraisemblances normalisées $\frac{P(q_i|o_t)}{P(q_i)}$.

Lorsqu'un réseau de neurones a été utilisé, l'apprentissage du HMM et du réseau de neurones a été fait séparemment. Bien qu'il existe des algorithmes d'apprentissage joint [196, 184], le déséquilibre entre le nombre d'itérations du HMM et du réseau de neurones ne nous a pas permis d'améliorer les résultats.

2.3.3.3 Implémentation des modèles et résultats

Plusieurs architectures ont été testées, basées sur un HMM "pur" avec GMM, ou sur des modèles neuro-markoviens MLP/HMM ou DNN/HMM. Pour les approches DNN/HMM, le DNN (Deep Neural Network) a été appris avec un pré-apprentissage des couches d'entrée à l'aide d'auto-encodeurs. Toutes les approches sont entrainées à estimer les vraisemblances des 4 états des 82 caractères (minuscules + majuscules + caractères accentués + chiffre + espace + ponctuation), soit $4 \times 82 = 284$ classes [11]. La plupart des méthodes sont appliquées sur 26 caractéristiques de type concavité et densité inspiré de [152].

Méthode	Caractéristiques	#entrées	N^1	N^2	N^3	#sorties
GMM	[152]	26	-	-	-	284
MLP-26	[152]	26	155	-	-	284
MLP-78	[152] $\times 3$	78	181	-	-	284
DNN-26	[152]	26	200	225	250	284
DNN-78	[152] $\times 3$	78	200	225	250	284
DNN-432	RAW pixels	432	400	350	300	284

Table 2.8: Les différentes méthodes pour estimer les vraisemblances des caractères. Pour les MLP et DNN, les N^i désignent le nombre de neurones des couches cachées. Les méthodes DNN-78 et MLP-78 sont entrainés sur les caractéristiques extraites de 3 trames consécutives aux instants $t-1$, t et $t+1$.

Pour appliquer notre modèle de ligne, les images de documents sont segmentées en ligne par une méthode à base de composantes connexes [159] donnant des résultats satisfaisant.

Tous les hyperparamètres des méthodes ont été optimisés sur une base de validation (taille et recouvrement de la fenetre, nombre d'états par caractère, nombre de gaussienne par état, nombre de neurones des couches cachées des MLP et DNN).

Les différents systèmes sont évalués sur la base RIMES[157] qui contient 1150 courriers dont 36000 mots ont été annotés. Nous avons choisi un découpage 950/100/100 en apprentissage/validation/test. Les méthodes sont évaluées par le calcul du rappel et de la précision.

11. Pour permettre l'apprentissage des réseaux de neurones, une base étiquetée au niveau trame a été générée par alignement forcé d'un HMM pur sur la base d'apprentissage.

Pour chaque courrier, un lexique est constitué en choisissant au hasard 10 mots présents dans le courrier, complété par des mots choisis au hasard non présents dans le courrier. Trois tailles de lexique sont utilisées : 10, 100 et 500 mots clefs. Des résultats avec un seul mot clef sont également présentés. Rappelons que la constante G permet d'arbitrer l'extraction en favorisant le lexique ($G \to 1$) ou le *filler model* ($G \to 0$).

Le tableau 2.9 donne le *Break-Even-Point*[12].

On peut constater la supériorité des approches neuro-markoviennes sur le HMM classique. On note également une amélioration sensible des résultats entre un MLP entrainé sur des caractéristique et un DNN entrainé sur directement sur des pixels. Quelle que soit l'approche utilisée, les performances baissent lorsque l'on augmente la taille du lexique de mots clefs.

# de mots clefs	1	10	100	500
GMM	68.7	57,5	50,9	28,2
MLP-26	70.4	60,6	53,9	37,6
DNN-26	71.9	61,8	55,2	39,7
MLP-78	54.6	62,7	55,5	39,9
DNN-78	74.0	64,1	56,3	41,0
DNN-432	75.8	65,8	57,5	42,4

Table 2.9: Break even point (%) de plusieurs systèmes pour différentes tailles de lexiques.

conclusion sur l'extraction de mots clefs

Ces résultats constituait à l'époque (2012) une confirmation de la capacité des architectures "profondes" (pour l'époque toujours, 4 couches !) à générer une représentation discriminante à partir des données brutes. Ces modèles ont depuis été adoptés dans la communauté [114, 113], et améliorés en utilisant des architectures plus récentes telles que des CNN et LSTM. Les performances en extraction sont intéressantes, et permettent d'envisager des traitements de plus haut niveau, sans pour autant être confronté au problème des très grands lexiques nécessaire à la lecture intégrale des documents. Par ailleurs, la généricité du modèle de ligne nous a permis d'envisager d'autres requêtes plus complexes de type expressions régulières, présentées dans la section suivante.

2.3.3.4 Extraction d'expressions régulières

On peut définir une expression régulière comme un motif modélisant un ensemble de séquences qui respectent une syntaxe particulière. La syntaxe des séquences cibles repose sur

12. Le *Break-Even-Point* est le point tel que le rappel=précision.

des caractères spéciaux qui décrivent des sous ensembles de l'alphabet complet des caractères. Il est également possible de spécifier la position et le nombre de caractères ou de caractères spéciaux dans les séquences recherchées, ou encore des opérateurs logiques tels que le "ou" (présence du caractère 'a' *ou* 'b').

Les notations adoptées dans ce travail pour les caractères spéciaux et opérateurs classiques sont les suivantes :

- # modélise la fin ou le début d'une expression régulière.
- **a** représente le caractère 'a'.
- **[a-z]** représente l'ensemble des caractères compris entre 'a' et 'z'.
- **a ?** représente 0 ou 1 occurence du caractère 'a'.
- **a*** représente 1 ou plusieurs occurences du caractère 'a'.
- **a{n}** représente n occurences du caractère 'a'.
- **a|b** représente le "ou" logique entre les caractères 'a' et 'b'.

à partir de ces caractères spéciaux et opérateurs, il est possible de créer des motifs plus ou moins complexes pour detecter des séquences particulières. Voici quelques exemples d'expressions régulières :

- #**[0-9]***# modélise toutes les séquences de chiffres.
- #**[a-z]***# et #**[A-Z]***# modélisent les séquences de minuscules et de majuscules
- #**[a-z]*tion**# modélise les mots terminant par *tion*
- #**[0-3] ?[0-9]/[0-1] ?[0-9]/[1-2][0-9]{3}**# modélise une date de type *31/12/2018*.
- #**[0-9]{5} [A-Z]***# permet de détecter les code postaux français suivis du nom de de ville écrit en majuscule

La détection d'expressions régulières peut être vue comme une généralisation de la détection de mots clés abordés précédemment.

Les expressions régulières sont très largement utilisées sur des textes numériques [164, 185, 191]. La détection est généralement réalisée en convertissant l'expression régulière recherchée en un automate à états finis [232, 231]. La recherche est ensuite résolue en parcourant l'automate pour conserver uniquement l'ensemble des états atteignables. La recherche se termine lorsque la chaîne d'entrée est épuisée.

La détection d'expression régulière dans des images de documents est un problème beaucoup moins abordé. Il s'agit d'un problème difficile, puisque la recherche du motif #**[a-z]*cha[a-z]***# devra par exemple retourner des séquences aussi variées que "<u>cha</u>ngement", "pe<u>cha</u>rmant" ou "pa<u>cha</u>". À la variabilité des séquences à retourner s'ajoute la nécessité d'une étape de reconnaissance, apportant du bruit et/ou de l'incertitude sur les caractères observés. Ainsi, une substitution, une délétion ou une insertion conduiront toutes à un échec d'une recherche exacte dans la sortie d'un OCR. Il est évidemment possible d'exploiter un lexique pour corriger les différentes hypothèses locales de reconnaissance avant d'effectuer la recherche. Cette approche a été proposée sur des documents imprimés [212, 203, 177],

mais les problèmes liés aux grands lexique réapparaissent. On peut également mentionner la détection de dates [175] à l'aide d'un MMC, ou la détection de champs numériques effectués pendant ma thèse [147, 158], mais ces travaux ne sont pas suffisamment génériques et sont limités à des types de champs particuliers.

Dans ces travaux, nous proposons une généralisation du modèle de ligne pour l'extraction de mots clefs de la section 2.3.3.2 à un modèle de ligne pour la détection d'expressions régulières. Ces travaux ont été initiés par Yousri Kessentini en 2013 [101] puis prolongés lors de la thèse de Gautier Bideault [57, 56].

2.3.3.5 Modèles hybrides pour la détection de mots clés et d'expressions régulières

Le modèle de ligne pour la détection d'expressions régulières est inspiré du modèle de ligne décrit dans la section 2.3.3.2. Il reprend la stratégie des modèles de remplissage, en revanche le modèle de l'information pertinente (KL) doit être adapté pour modéliser les motifs recherchés.

Par rapport à une recherche de mots clefs, les contraintes lexicales sont relachées et le modèle de ligne doit pouvoir s'aligner sur des séquences de contenu et de taille variable (#**re[a-z]***#, #**[a-z]*tion**#, etc.). Afin de modéliser les contenus variables, nous avons mis en place des modèle ergodiques spécifiques appelés méta-modèles, qui sont créés à la volée lors de la recherche. La figure 2.20 présente les trois méta-modèles les plus utilisés qui sont le modèle de minuscules #**[a-z]***#, le modèle de majuscules #**[A-Z]***# et le modèle de chiffres #**[0-9]***#, mais il est évidemment possible de créer d'autres modèles tels qu'un modèle de voyelle par exemple (**a|e|i|o|u|y**).

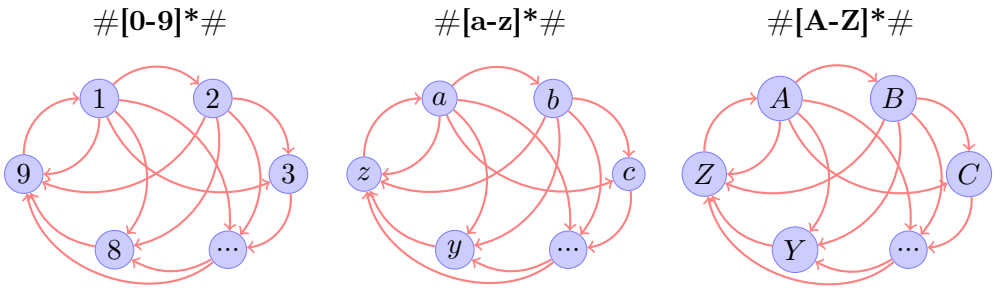

Figure 2.20: Méta-modèles de chiffres, de minuscules et de majuscules utilisés pour la détection d'expressions régulières.

La longueur variable des séquences est modélisée par des auto-transitions sur le ou les méta-modèles suivis de l'opérateur *. La Figure 2.21 présente un exemple de modèle de ligne pour la détection d'une expression régulière. Concernant l'étape de décision, les modèles

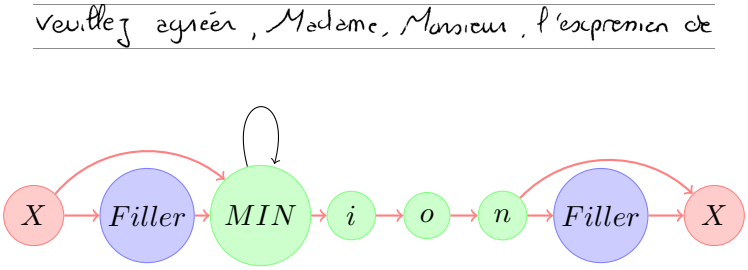

FIGURE 2.21: Modèle de ligne généré pour la détection de l'expression régulière #**[a-z]*ion**# (séquences de caractères minuscules se terminant par "ion"). La séquence de caractères minuscule de taille quelconque **[a-z]*** est modélisée par le méta modèle de minuscules MIN présenté en figure 2.20. Les $FILLER$ sont identiques à ceux de la section 2.3.3.2. Sur l'exemple de ligne de texte, cette requête est supposée renvoyer la séquence "expression".

de ligne sont mis en compétition avec un modèle ergodique de type $FILLER$ sur toute la ligne. Contrairement à la détection de mots clefs où la compétition était arbitrée par une constante G, ici les expressions régulières sont trop variables pour pouvoir fixer une valeur à priori. La décision s'effectue donc plus classiquement par un seuillage du ratio des deux vraisemblances avec/sans l'expression. Les probabilités de transitions à l'intérieur des méta-modèles du MMC étant uniformes, la recherche du meilleur chemin - toujours réalisé par un alignement du Viterbi - est uniquement dirigé par les hypothèses de reconnaissance locales. Nous présentons dans la section suviante les expériences permettant d'évaluer ces modèles.

Expériences et résultats

Comme il n'existe pas à notre connaissance d'autres travaux sur la détection d'expressions régulières dans des images de documents, nous avons réalisé un benchmark de plusieurs modèles pour l'estimation des vraisemblances. Ces modèles exploitent tous des Histogrammes de Gradients Orientés [149] extrait d'une fenêtre glissante de taille 8x64 pixels, formant un vecteur de taille 64. Comme dans la section 2.3.3.2, l'évaluation des performances de nos systèmes a été effectuée sur la base de phrases RIMES [126]. Les modèles évalués sont un HMM pur, et des méthodes neuro-markoviennes couplant le HMM avec un MLP, un RNN, et un BLSTM. Nous renvoyons à [54] pour le paramétrage détaillé des différents modèles.

L'évaluation de la détection pose également la question du choix des expressions régulières à rechercher. Là encore, comme il n'existe pas d'autres travaux sur le sujet, nous avons choisi plusieurs expressions régulières, dont la seule contrainte était qu'elles soient suffisamment présente dans la base de test pour procurer une estimation correcte des résultats. Nous avons ainsi

recherché les expressions régulières suivantes : #**effe[a-z]***#, #**pa[a-z]***#, #**com[a-z]***#, #**cha[a-z]***#, #**[a-z]*er**#, #**[a-z]*tion**#, #**[a-z]*tt[a-z]***#, #**[a-z]*mm[a-z]***#.

La figure 2.22 présente les courbes rappel précision obtenues en faisant varier le seuil d'acceptation sur le ratio des vraisemblances du modèle de remplissage et du modèle contenant l'expression régulière. Une première constatation concerne la comparaison des différents modèles, où le BLSTM fournit les meilleurs résultats, suivi du RNN, du MLP et du HMM avec gaussiennes. Si ce classement ne constitue pas une surprise, l'écart entre le LSTM et les autres approches est particulièrement marqué, surtout sur les expressions régulières peu contraintes telles que les les séquences de majuscules où le break even point varie d'environ 20% (HMM) à 76% (LSTM). La seconde constatation concerne les performances obtenues qui semblent intéressantes. On peut remarquer que plus la séquence est contraignante, plus les résultats sont bons avec des BEP (*Break-Even-Point*) compris entre 50 et 80%. Certaines requêtes peu contraignantes telles que les séquences de chiffre ou les séquences de majuscules proposent même des BEP supérieurs à 70%. En revanche, d'autres requêtes telle que #**[a-z]*tt[a-z]***# admettent des performances plus faibles, certainement due à la double lettre 't' qui par ailleurs est une lettre assez courte.

Conclusion sur la détection d'expressions régulières

Le modèle proposé dans ce travail permet d'extraire des expressions régulières dans des documents manuscrits, en prenant en compte les incertitudes liées à la reconnaissance. Pour cela, des modèles de lignes traduisant le motif recherché sont construits à la volée, en se basant sur des méta modèles. Si la qualité de la détection est difficile à évaluer en l'absence de tâche exploitant le résultat des requêtes, les performances atteintes semblent intéressantes et laissent entrevoir des perspectives telles que la détection d'entitées nommées, en détectant par exemple des séquences du type *"Prénom NOM"* respectant la casse.

2.3.4 Reconnaissance d'écriture avec très grand lexique

2.3.4.1 Introduction

En fin de section 2.3, nous avons constaté que l'intégration des informations linguistiques dans les moteurs de reconnaissance avait connu moins d'intérêt que la reconnaissance optique qui avait elle bénéficié de progrès remarquables. Nous présentons dans cette section une contribution pour la reconnaissance d'écriture en très grand lexique, developpée pendans la thèse de Bruno Stuner [8]. L'idée générale de ces travaux est de contourner les approches "dirigée par le lexique" (voir section 2.3) dont les performances chutent quand la taille du lexique augmente.

Dans ces travaux, nous proposons un nouveau paradigme de reconnaissance permettant de prendre en compte des très grands lexiques. La strategie est basée sur une opération

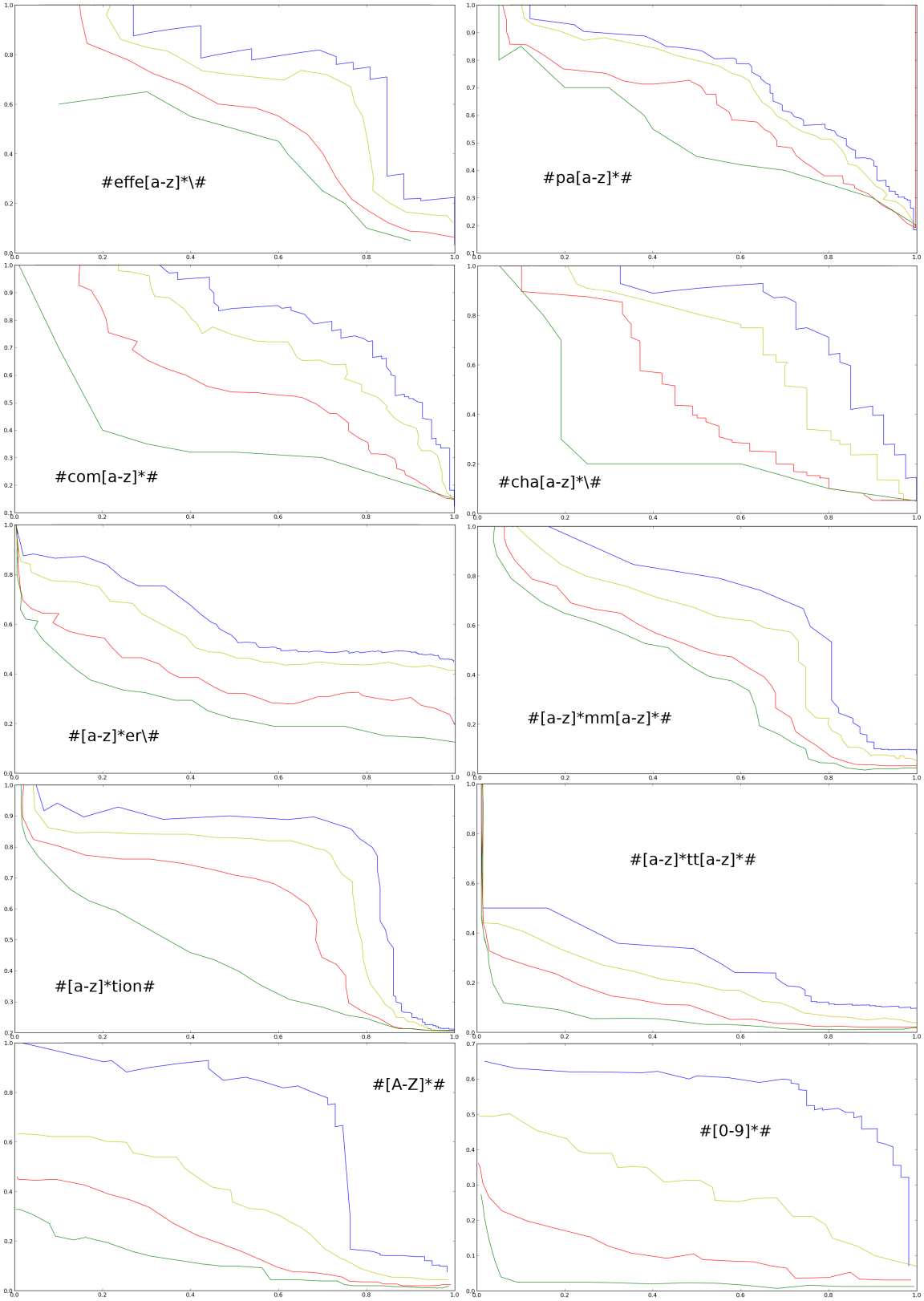

Figure 2.22: Courbes rappel (abscisse) précision (ordonnée) de plusieurs expressions régulières, pour les 4 systèmes testés : HMM en vert, MLP en rouge, RNN en jaune et LSTM en bleu.

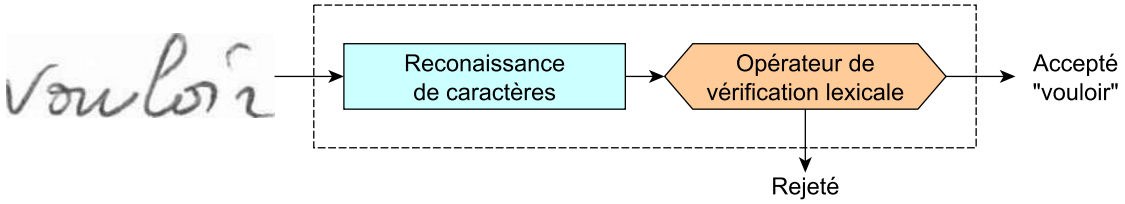

Figure 2.23: Nouveau paradigme pour la reconnaissance d'écriture.

de vérification combinée avec une cascade de classifieurs complémentaires. Plutôt que de contraindre un reconnaisseur optique à ne considérer que des hypothèses possibles ou probables, nous générons une grande quantité d'hypothèses de reconnaissance, sans se préoccuper de la vraisemblance linguistique. Dans un deuxième temps, les hypothèses sont validées ou rejetées avec un opérateur binaire rapide que nous avons baptisé "vérification lexicale".

2.3.4.2 Approche proposée

Opérateur de vérification lexicale

Dans les approches dirigées par le lexique, les hypothèses de reconnaissance optique sont alignées sur les différentes entrées du lexique, généralement par une programmation dynamique. Il existe différentes variantes plus ou moins optimisées de ces algorithmes, mais tous ont un coût calculatoire qui dépend de la taille du dictionnaire.

La vérification lexicale proposée dans ces travaux consiste à simplifier la mise en correspondance des hypothèses de reconnaissance optique avec les entrées du lexique. Pour cela, nous proposons de ne pas effectuer d'alignement entre les séquences, mais seulement de vérifier si l'hypothèse de reconnaissance brute appartient au lexique considéré. La règle de décision s'ennonce ainsi : « *Si la séquence de caractère proposée par le reconnaisseur optique appartient au lexique, alors elle est acceptée, sinon, elle est rejetée* ». L'idée sous jacente de cette vérification est que si la séquence fournie par le reconnaisseur appartient au lexique, alors il y a très peu de risque pour qu'elle soit erronée. La probabilité P_{FA} que cette règle produise une Fausse Acceptation (FA) peut être modélisée de la manière suivante. Si W est une hypothèse du reconnaisseur optique $Reco$ et L est un lexique, alors :

$$P_{FA} = P(W\text{est erronée } \wedge W \in L \mid Reco) \tag{2.21}$$

où "West erronée" renvoie *vrai* si l'hypothèse W est fausse.

On se propose d'observer le comportement de cet opérateur de vérification sur un problème préliminaire de reconnaissance de mots isolés dans le tableau 2.10. Cette expérience compare

les performances obtenues par trois reconnaisseurs de type BLSTM sur la base Rimes. Le premier reconnaisseur fournit des hypothèses qui sont correctes ou erronées. Le second reconnaisseur fournit des hypothèses qui sont soumises à l'opérateur de vérification : si l'hypothèse appartient au lexique, elle est acceptée, à tort (erreur) ou à raison. Sinon, elle est rejetée. Pour le troisième reconnaisseur, nous avons reporté les résultats d'une des meilleure méthode de la litérature [43] basée sur un paradigme de reconnaissance dirigée par le lexique. Cette dernière méthode propose de loin la meilleure précision, 25 points devant la vérification lexicale. En revanche, l'erreur proposée par l'approche par vérification est la plus faible (2.25% contre 8.52%).

Table 2.10: Précision, erreur et rejet au niveau mot sur la base Rimes, produits par un reconnaisseur BLSTM avec ou sans vérification lexicale, ainsi qu'une méthode dirigée par le lexique.

Network	Précision	Erreur	Rejet
BLSTM	66.37	33.63	0
BLSTM + décodage dirigé [43]	**91.48**	8.52	0
BLSTM + vérification lexicale	66.37	**2.25**	31.38

Cette expérience préliminaire confirme que la probabilité P_{FA} est très faible, et pose deux questions principales : i) comment augmenter significativement la précision de l'approche par vérification, et ii) que faire des rejets ? La seconde proposition de notre approche consiste à mettre en œuvre une cascade de reconnaisseurs afin de traiter les rejets.

Cascade de LSTM vérifiée par le lexique

L'idée de la cascade [111, 156, 186] est de successivement soumettre les rejets d'un reconnaisseur à un autre reconnaisseur. Le processus continue tant que l'hypothèse de reconnaissance n'est pas acceptée par la règle de vérification lexicale (voir figure 2.24). Nous avons choisi d'utiliser des classifieurs de type BLSTM, non spécialisés sur les rejets du reconnaisseur précédent, car trop peu de données seraient disponibles. Les reconnaisseurs sont ordonnés par erreur de délétion décroissante sur l'ensemble de validation.

Afin de rendre les décisions prises au sein de la cascade plus robustes, nous avons introduit la règle suivante : pour qu'un mot soit définitivement accepté, il faut qu'il passe par un nombre minimum de réseaux ayant pris la même décision. Ce nombre, appelé NMAD, permet de ne pas être dépendant d'un seul classifieur. En revanche, cette règle ralenti la décision puisque le NMAD doit être atteint avant de sortir de la cascade.

Pour que cette cascade soit efficace, il est nécessaire d'obtenir un ensemble de classifieur complémentaires, pour permettre d'explorer plusieurs hypothèses de reconnaissance tout au long de la cascade. Nous proposons dans la section suivante une contribution pour générer

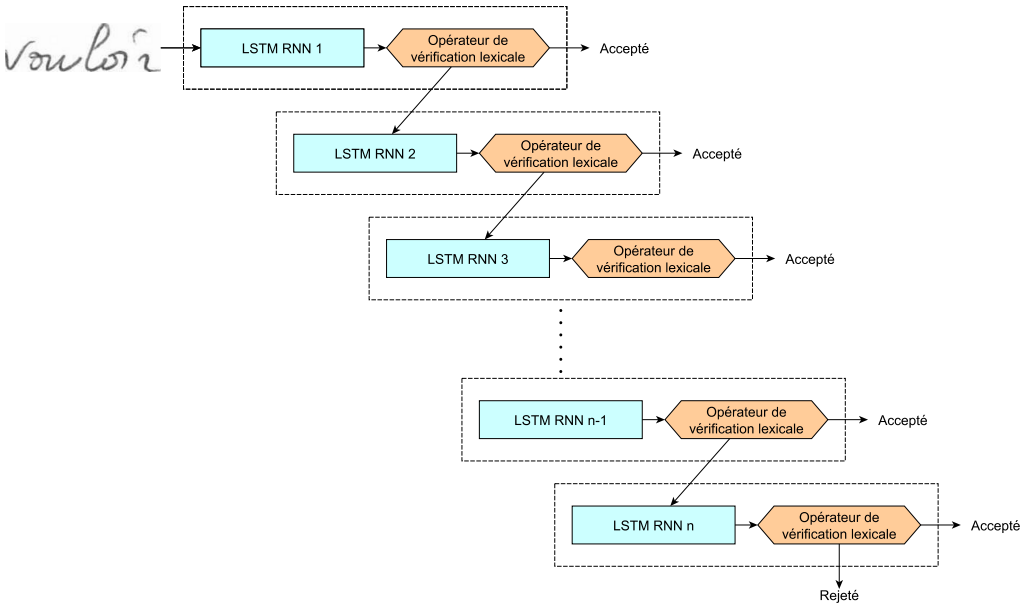

Figure 2.24: Schéma en cascade proposéChaque classifieur traite les rejets de l'étage précédent.

un nombre important de classifieurs complémentaires, basé sur les propriété d'apprentissage des architectures profondes.

Generation de cohortes de classifieurs LSTM complémentaires

Il existe de nombreuses manière de générer de nombreux classifieurs. La méthode la plus évidente consiste à utiliser des architectures variées (LSTM, MDLSTM, géométrie du réseau, etc.) ainsi que des caractéristiques variées (HOG, pixels, etc.). Cependant ces modifications sont couteuses en terme de conception et de temps d'apprentissage, et sont limitées en nombre. Une alternative est de générer de nombreux réseaux identiques, mais dont les initialisations sont différents [118]. Cette méthode reste cependant longue à entraîner, surtout si le nombre de classifieurs attendu est important.

Afin d'obtenir un grand nombre de classifieurs complémentaire avec un effort réduit sur la conception et le temps d'apprentissage, notre idée est de nous baser sur un apprentissage unique mais controlé, dont sont extraits tous les classifieurs. Cette idée nous a été inspirée par les travaux de Choromanska [58], où un parallèle est fait entre la fonction de perte des réseaux de neurones et des polynômes aléatoires de haut degré, disposant de minimums locaux de même magnitude. Les auteurs arrivent à la conclusion qu'à l'issue d'un apprentissage, la plupart des minimums d'un réseau de neurones profond sont équivalents. Ce résultat, qui a également été démontré dans [41] dans le cas linéaire, va à l'encontre de la croyance selon laquelle il y aurait un minimum global à atteindre et des minimums locaux à éviter à tout prix. Notre proposition consiste simplement à exploiter ces nombreux minimums de

performances équivalentes comme solution pour générer des ensembles de classifieurs. Nous avons baptisé ces ensembles de classifieurs issus d'un apprentissage unique des "cohortes".

Cependant, cette stratégie de génération de cohorte pour alimenter la cascade demande un certain contrôle de l'apprentissage. En effet, lors de celui ci nous aimerions atteindre une zone de l'espace des paramètres où plusieurs minimums sont visités, et éviter d'être bloqués dans un minimum unique, aussi bon soit il. Ce contrôle peut être obtenu par trois paramètres importants que sont le momentum [13], le mélange des données et le pas d'apprentissage. Dans nos expérimentations nous avons utilisé un momentum de 0.9, valeur classique utilisée dans la littérature, afin de ressortir des minimums locaux. Le mélange des données est également important car il permet de modifier l'état du réseau entre chaque époque et ainsi d'obtenir de la diversité. Le paramètre le plus important est certainement le pas d'apprentissage, car s'il est trop grand la convergence sera mauvaise, et s'il est trop petit les modifications entre deux époques seront trop faibles pour apporter la complémentarité recherchée. Dans nos expériences, le comportement espéré est obtenu pour un pas de 10^{-4}. En choisissant ces paramètres, les réseaux sont sélectionnés à chaque époque d'un apprentissage unique pour constituer une cohorte, comme illustré figure 2.25. Nous décrivons maintenant l'implémentation de la cascade complète et les résultats obtenus.

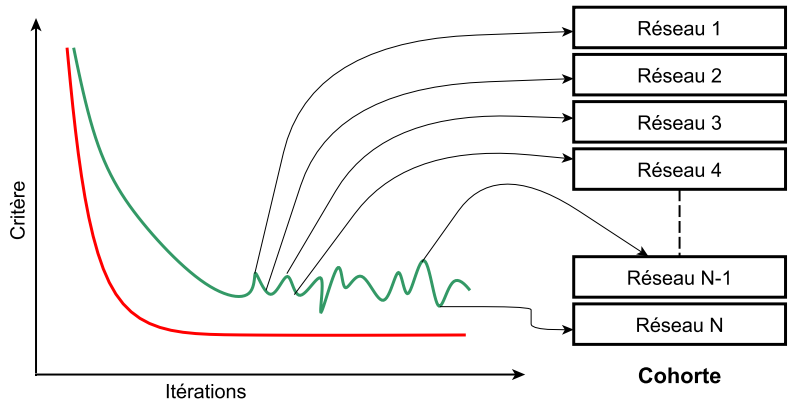

Figure 2.25: La cohorte est finalement constituée de nombreux réseaux extraits des époques d'un seul apprentissage. Plutôt qu'une convergence rapide et stable (en rouge), on recherche une convergence plus oscillante (en vert) permettant de visiter l'espace des paramètres, à la recherche de nombreux minimums.

2.3.4.3 Implémentation et résultats

Le système proposé est basé sur les principes méthodologiques décrits dans la section précédente : vérification lexicale, cascade de classifieurs et génération de cohorte en un seul

13. Le momentum est un facteur qui contrôle la mise à jour des poids et ainsi la convergence de l'apprentissage.

apprentissage. Nous présentons maintenant les détails d'implémentation et une comparaison des résultats avec les méthodes à l'état de l'art.

Les expérimentations ont été menées sur la base de mots isolés Rimes [143] et IAM [178] avec les lexiques et découpages officiels. Dans toutes nos expérimentations, les performances sont mesurées avec l'erreur caractère (Character Error Rate, CER), calculée avec la distance de Levenshtein. Nous mesurons également la précision, le taux d'erreur et le taux de rejet au niveau mots.

Génération de cohortes

Pour la génération de cohortes, deux architectures ont été utilisées. La première est un réseau BLSTM à deux couches de 70 et 120 LSTM séparées par une couche dense de 100 neurones cachés sans biais issu de [68], et entraîné sur des histogrammes de gradient orientés [166]. La deuxième architecture est un MDLSTM identique à celui présenté dans [142], composé de trois couches de 2, 10 et 50 LSTM, et deux couches de sous-échantillonnage de 6 et 20 neurones. Les images normalisées sont directement données en entrée du réseau. Nous avons utilisé des transformations (rotations et étirements) afin d'augmenter la taille de la base d'apprentissage ($\times 3$). 100 réseaux sont extraits des deux apprentissages (BLSTM et MDLSTM).

Concernant le *Nombre Minimum d'Accord de Décision (NMAD)*, celui ci a été fixé en distinguant les mots courts qui sont plus risqués, des mots longs. Le NMAD a donc été fixé à 10 pour les mots de moins de 3 caractères et à 3 pour les mots plus longs. Nous avons observé que les performances sont très peu dépendantes de ces valeurs.

Génération de cascades basée sur plusieurs cohortes

Bien que les réseaux complémentaire puisse être générés à partir d'un seul apprentissage, nous avons voulu essayer de constituer une cascade issue de plusieurs cohortes. Nous avons donc effectué des apprentissage des deux architectures présentées avec des initialisations différentes. Finalement, 2100 réseaux différents sont extraits de 10 apprentisages sur la base Rimes :

— 2 apprentissages de BLSTM ;
— 1 apprentissage de MDLSTM ;
— 4 apprentissages de BLSTM avec transformations ;
— 3 apprentissages de MDLSTM avec transformations.

L'apprentissage de ces 10 réseaux sur 3 machines a pris 10 jours, alors qu'il aurait fallu 2 ans et 9 mois pour apprendre les 2100 réseaux. Pour IAM, la cohorte est constituée de 1039 réseaux.

Résultats Le tableau 2.11 reprend les résultats de l'expérience préliminaire, et les compare aux résultats obtenus par plusieurs cascades. La première est constituée de 3 réseaux BLSTM, la seconde est constituée de 100 réseaux BLSTM et MDLSTM issue de a même cohorte, et la troisième est construite des 2100 réseaux issus des 10 cohortes présentées.

On peut remarquer que la précision augmente lorqu'on ajoute des réseaux dans la cascade. Les BLSTM donnent de meilleurs résultats que les MDLSTM. L'approche à 2100 réseaux donne 95,60% de précision, ce qui se rapproche du résultat à l'état de l'art (96,10%). Lorsqu'on applique un Viterbi sur les rejets subsistants en bout de cascade, nous dépassons cette performance (96,52%).

Table 2.11: Performances de plusieurs approches sur la base Rimes avec le lexique à 5744 mots.

System	Précision	Erreur	Rejet	CER
BLSTM	66.37	33.63	-	11.24
BLSTM + viterbi	91.48	8.52	-	2.77
BLSTM + vérification	66.37	2.25	31.38	x
Cascade de 2 BLSTM + vérification	73.84	3.38	22.78	x
Cascade de 3 BLSTM + vérification	77.76	3.76	18.48	x
Cascade de 100 BLSTM	84.53	3.13	12.34	0.99
Cascade de 100 BLSTM (+transformations)	89.56	2.74	7.70	0.82
Cascade de 100 MDLSTM	80.27	4.09	15.64	1.40
Cascade 2100 réseaux	95.60	3.00	1.40	0.99
Cascade 2100 + viterbi	**96.52**	3.48	-	**1.34**
Menasri *et al.* [118]	95.25	4.75	-	-
Poznanski *et al.* [43]	96.10	3.90	-	1.90

Pour la base IAM, nous avons également obtenus des résultats à l'état de l'art, et les mêmes tendances sont observées. Nous renvoyons vers [8] pour les résultats détaillés sur cette base.

Évaluation de l'approche avec des très grands lexiques

Rappelons que l'objectif de cette approche était d'être capable de gérer des lexiques de très grande taille. En plus des lexiques officiels de Rimes et IAM, nous avons donc évalué notre approche avec un très grand lexique, plus réaliste vis-à-vis des applications industrielles ayant à gérer les entités nommées par exemple. Pour cela, nous avons constitué deux lexiques (qui sont disponibles sur demande) :

— *Un lexique francais de 342 275 mots* constitué de l'union du dictionnaire francais

Gutemberg et du lexique Rimes.

— *Un lexique francais de 3 276 994 mots* obtenu par l'union des mots du Wikipedia francais, du Wiktionnaire, du dictionnaire francais Gutemberg et du lexique Rimes.

— *Un lexique anglais de 2,439,432 mots* formé par l'union des mots du *one billion word data* [95] et du lexique IAM.

Le tableau 2.12 présente les résultats obtenus sur Rimes par la cascade de 2100 réseaux pour différentes tailles de lexique allant de 1692 à 3276994 mots. Remarquons que la différence de performance entre les lexiques à 1692 et 5744 est très faible, comparée à la traditionnelle chute rencontrée par les méthodes dirigées par le lexique. La différence augmente lorsque l'on considère les plus grands lexiques, mais restent limitées. On obtient notamment plus de 90% de bonne reconnaissance avec un lexique à plus de 3 millions de mots, soit 600 fois plus de mot que le lexique Rimes. Tous ces résultats montrent la très faible sensibilité de l'approche proposée à la taille du lexique. Nous n'avons pas trouvé dans la littérature de résultats avec des lexiques aussi grands.

Table 2.12: Résultat de la cascade à 2100 réseaux sur la base Rimes avec des lexiques de différentes tailles.

Taille du lexique	Précision	Erreur	Rejet	CER
1692	96.08	2.32	1.60	0.76
5744	95.60	**3.00**	1.40	0.99
342 275	93.41	5.47	1.12	1.72
3 276 994	90.14	9.18	0.68	2.67

d

Temps de traitement Nous analysons ici un inconvénient potentiel de notre approche, qui requiert de nombreux réseaux et pourrait donc demander des temps de traitement conséquents. L'approche est dans la pratique assez efficace puisque la grande majorité des mots "sortent" de la cascade bien avant les 2100 étages, le NMAD ayant été atteint plus tôt. Concernant la vérification lexicale, celle ci est effectuée en moins d'une microseconde, même sur le lexique à plus de 3 millions de mots. En utilisant la cascade complète, on observe que 80% des mots sont reconnus en moins de 0.17s (14 réseaux ou moins), et 90% en moins d'une seconde (41 réseaux ou moins). Les expériences ont été menées sur un processeur de type Intel CPU i7-3740QM, avec un code non optimisé. Concernant la mémoire, l'ensemble des 2100 réseaux encodé en double précision tient dans 6GB.

Nous avons par ailleurs montré qu'il était possible de ne conserver qu'un sous ensemble des 2100 réseaux, sans baisse notable des résultats. Pour cela, nous avons supprimé les réseaux produisant les performances les plus faibles sur une base de validation pour ne conserver que

118 réseaux. Avec cette cascade réduite, nous obtenons une erreur de **3.64%** (contre 3.48% avec les 2100 réseaux) et une CER de **1.49%** (contre 1.34%). Le temps de traitement moyen est ainsi réduit de 729 ms à 197 ms.

Conclusion En évitant la reconnaissance dirigée par le lexique, nous avons proposé un nouveau paradigme capable d'obtenir des performances élevées, et qui autorise une reconnaissance avec de très grands lexiques. La limitation principale de l'approche proposée est qu'elle ne concerne que les mots isolés. La section suivante propose ainsi une extension de l'approche aux lignes de texte qui a été proposée dans [28, 8]

2.3.4.4 Extension du principe de cohorte à la reconnaissance de lignes de texte

Les principes de cohorte et de vérification lexicale ne peuvent être appliqués tels quels à la reconnaissance des lignes de texte. En effet, on pourrait imaginer l'application successive de la cascade sur tous les mots d'une ligne de texte, mais les hypothèses de reconnaissance issues des classifieurs d'une cohorte ne sont a priori pas alignées. Si l'on souhaite combiner différents moteurs de reconnaissance, il faut donc réaliser un consensus sur la segmentation en mots entre les différents moteurs.

La méthode de combinaison à l'état de l'art pour la combinaison de systèmes séquentiels est *Recognizer Output Voting Error Reduction* (ROVER) [199]. ROVER combine les sorties de plusieurs moteurs en deux étapes : un alignement et un vote. L'alignement des sorties est réalisé par programmation dynamique, alors que le module de vote sélectionne la meilleur solution en se basant sur la fréquence des hypothèses de mots et leur score de reconnaissance.

La méthode proposée pour la reconnaissance de lignes de texte est une hybridation entre l'algorithme ROVER et les principes évoqués précédemment. Le module de vote de ROVER ressemble fortement au vote que nous avons introduit dans notre schéma de combinaison en cascade, à l'étape de vérification près. Dans ce travail, nous proposons donc de modifier le module de vote de ROVER afin d'introduire la vérification lexicale. Cette modification est simplement réalisée en attribuant un score à chaque hypothèse de mots, en fonction de l'appartenance ou non du mot au lexique. Ce score est substitué au score de reconnaissance de chaque mot. Nous appliquons ensuite ce nouveau schéma de vérification ROVER pour la combinaison de nombreux classifieurs LSTM issus des cohortes.

Implémentation et résultats

Nous avons réutilisé le schéma d'apprentissage des cohortes de la section précédente, et les avons appliqué à un MDLSTM et 3 BLSTM appris avec des initialisations différentes sur la base d'apprentissage de la base de lignes de texte Rimes. 454 réseaux ont été extraits de ces 4 cohortes, et sont utilisés pour la combinaison ROVER.

L'évaluation des résultats est effectuée comme dans [86, 50] pour calculer le taux d'erreur mot (Word Error Rate, WER) et le taux d'erreur caractère (CER). Les résultats sont présentés dans le tableau 2.13, et comparés à ces deux méthodes à l'état de l'art pour la reconnaissance de lignes de texte [50, 86]. Ces deux systèmes relativement classiques sont constitués de réseaux à base de LSTM suivis d'une reconnaissance dirigée par le lexique, et d'un modèle de langage n-gram appris sur le corpus Rimes.

Table 2.13: Résultats sur la base Rimes de la combinaison ROVER modifiée pour prendre en compte la vérification lexicale des 454 réseaux.

Système	CER	WER
ROVER + cohort + LV (approche proposée)	**2.6**	**8.4**
Voigtlaender et al. [50]	2.8	9.6
Pham et al. [86]	3.3	12.3

Notre approche basée sur le ROVER modifié, le principe de cohorte et l'opérateur de vérification lexicale permet d'obtenir de meilleurs résultats que l'état de l'art, alors que nous n'utilisons pas de modèle de langage.

Ces résultats montrent que le concept de cohorte associé à l'opérateur de vérification peut être facilement implémenté pour la reconnaissance de lignes de texte. L'approche peut dans ce cas être utilisée avec ou sans modèle de langage.

2.3.4.5 Conclusion

Dans cette contribution nous avons présenté un nouveau paradigme se substituant à la traditionnelle reconnaissance dirigée par le lexique. L'efficacité de la méthode repose sur la vérification lexicale qui est un opérateur robuste et immédiat, permettant d'exploiter au mieux les hypothèses générées par des cohortes de classifieurs organisés en cascade. Une méthode originale pour la génération de cohorte extrait de nombreux classifieurs d'un apprentissage de modèle unique a été présentée. En contrôlant la convergence de cet apprentissage, il est possible d'obtenir un grand nombre de classifieurs complémentaires. Nous avons montré l'intérêt de la méthode pour la reconnaissance de mots isolés et de lignes de texte, avec des performances à l'état de l'art. Nous avons également montré la capacité de la méthode à travailler dans des temps très raisonnables avec un lexique contenant plusieurs millions de mots.

De tels lexiques permettent d'envisager de nouvelles applications telles que la reconnaissance d'entités nommées, ou encore la reconnaissance dans un contexte multilingue. En terme de perspectives méthodologiques, nous avons assez peu exploré l'ordonnancement des classifieurs dans la cascade. Bien que la méthode ne semble pas très sensible à cet ordonnancement, il

pourrait être intéressant d'exploiter les complémentarités plus en détail afin d'en déduire le meilleur ranking de classifieurs.

Chapitre 3

Conclusion et perspectives

3.1 État des lieux sur l'apprentissage profond

En moins d'une dizaine d'années, l'apprentissage profond a révolutionné l'apprentissage statistique et son utilisation est désormais généralisée. Au delà de l'apprentissage, de nombreuses communautés scientifiques ont également été largement transformées par l'avènement des réseaux de neurones. Citons l'exemple notable de la vision par ordinateur, où les deux conférences majeures du domaine (CVPR [1] et ICCV [2]) peuvent désormais être considérées comme des conférences d'apprentissage, le plus souvent profond. Les raisons de ce succès sont la capacité des modèles neuronaux à résoudre des problèmes particulièrement complexes autrefois traités par des méthodes ad-hoc, leur simplicité de mise en œuvre ainsi que leur modularité.

Si les possibilités offertes par l'apprentissage profond semblent aujourd'hui infinies, l'utilisation des modèles neuronaux conséquents n'est pas sans poser un certain nombre de questions fondamentales, d'ingénierie et environnementales.

La question la plus évidente concerne les capacités de généralisation des architectures profondes. Si les SVM sont fondés sur cette notion de généralisation et possèdent de remarquables propriétés à ce niveau, l'obtention d'une bonne capacité de généralisation avec les modèles profonds est plus délicate à obtenir. En effet, le très grand nombre de paramètres rend les modèles profonds particulièrement sujets au sur-apprentissage. Une régularisation implicite ou explicite efficace des modèles est ainsi impérative.

Cette notion de généralisation est également liée à la quantité de données étiquetées disponible puisqu'une quantité de donnée infinie doit théoriquement permettre au modèle d'être parfaitement générique. Les recherches concernant l'apprentissage de modèles profonds avec de faibles quantités de données ou des données partiellement étiquetées sont très actives, comme discuté dans le manuscrit.

1. http ://cvpr2019.thecvf.com/
2. http ://iccv2019.thecvf.com/

Un problème fondamental lié à la nature « boite noire »des modèles neuronaux concerne l'interprétabilité des résultats. Bien qu'il existe des travaux visant à rendre les modèles neuronaux interprétables [2, 10], cette question est toujours ouverte et constitue une limitation des modèles profonds pour certains domaines requérant des résultats explicables tel que le diagnostic médical.

Une autre question aussi ancienne que les réseaux de neurones concerne la recherche des hyperparamètres permettant de construire le meilleur modèle. Là aussi, ce thème de recherche est très actif, et récemment de réelles avancées ont été faites avec les méthodes de type autoML [11, 5] permettant d'optimiser automatiquement l'architecture et les hyperparamètres des modèles. Les résultats semblent prometteurs et devraient permettre d'améliorer encore davantage l'accessibilité des architectures profondes aux non-spécialistes.

Enfin, à l'heure de l'optimisation énergétique à tous les niveaux de la société, on peut se poser la question de l'impact environnemental des architectures profondes et notamment de leur apprentissage. Il est courant d'entraîner pendant plusieurs jours des modèles à plusieurs millions de paramètres sur des GPU dont la consommation électrique est importante. Là aussi, de nombreuses pistes ont été développées ou sont en cours de développement, telles que l'utilisation de processeurs dédiés pour réduire les accès aux données particulièrement énergivores. On peut citer le TPU de Google [22] ou les FPGA [44] dont la consommation est très inférieure à celle des GPU. Notons que l'utilisation de modèles bien régularisés conduit également à des architectures *a priori* plus économes d'un point de vue énergétique.

La plupart des questions regroupées ci-dessus trouvent une partie de leur réponse dans l'utilisation de modèles plus légers. Si la profondeur des modèles n'a cessé d'augmenter, on observe une forte tendance à limiter le nombre de paramètres des couches neuronales, à travers bien sûr les couches convolutionnelles, mais aussi en étendant le mécanisme de poids partagés aux couches partagées [35], par des connexions résiduelles [48] ou encore en cherchant à éviter l'utilisation de récurrences qui possèdent un nombre important de paramètres [20].

Nous nous placons totalement dans cette tendance, et proposons dans la suite des perspectives pour des modèles plus légers, en particulier en ce qui concerne le traitement de séquences.

3.2 Perspectives

3.2.1 Proposition d'un modèle purement convolutif pour la reconnaissance d'écriture

La première perspective que nous souhaitons aborder est l'abandon des structures récurrentes pour le traitement de séquence, et son application à la reconnaissance d'écriture. En effet, les couches récurrentes sont à la fois lourdes en terme de nombre de paramètres, et longues

à entraîner du fait de l'impossibilité de paralléliser l'apprentissage. Comme proposé dans [17] pour la traduction automatique, une solution est de remplacer les récurrences par des convolutions afin de capturer les dépendances au sein des signaux d'entrée. Naturellement, la mémoire potentiellement infinie des réseaux récurrents est réduite à une mémoire finie, limitée à la taille des champs réceptifs de la structure convolutive. Dans ces travaux, le modèle sans récurrence est utilisé au sein d'un modèle encodeur-décodeur adapté à la traduction. Dans le cadre de la reconnaissance d'écriture, l'aspect encodeur-décodeur n'est a priori pas nécessaire car la séquence d'entrée (image de caractères) et la séquence de sortie (séquence ascii de caractères) sont synchronisées.

Nous souhaitons ainsi développer un modèle convolutif sans récurrence, capable de gérer les tailles variables des signaux aussi bien en entrée qu'en sortie. Deux travaux récents [9, 1] constituent des tentatives intéressantes et se rapprochent du but recherché. Le premier [1] propose un système hybride CNN + CTC qui ne s'affranchit pas des couches denses et propose donc un système qui pourrait être allégé. Le second [9] est un système complexe basé sur une architecture totalement convolutive avec une utilisation importante de normalisation des couches, qui a obtenu des performances à l'état de l'art sur les données de la compétition compétition READ 2018. Toutefois le système reste assez lourd et semble nécessiter une utilisation intensive d'augmentation de données, ce qui va à l'encontre des principe de légèreté que l'on souhaite proposer pour toutes les raisons évoquées ci-dessus.

La solution envisagée est donc un modèle purement convolutif basé sur les avancées récentes permettant de limiter fortement le nombre de paramètres du modèle : convolution séparables et mécanisme de couches partagées. Les convolutions séparables peuvent être utilisées au niveau spatial (optimisation de la taille des filtres de convolution comme dans les architectures inception [48, 29]), comme au niveau des canaux (depthwise convolutions [16]) pour rechercher des motifs dans la profondeur et réduire le nombre de canaux successifs. Le mécanisme de couches partagées [35] consiste à empiler plusieurs fois la même couche de convolution, avec des connexions résiduelles. Le but est d'augmenter la profondeur de l'architecture et d'obtenir une analyse multi échelle, sans pour autant augmenter le nombre de paramètres. On pourra également exploiter les portes (gates) que l'ont retrouve dans les neurones LSTM et qui ont été adaptées aux convolutions [17], qui semblent efficaces selon des premiers essais sur la reconnaissance d'écriture [14].

Nous proposons également l'apprentissage du modèle purement convolutif complet avec une fonction de perte de type CTC [162] afin d'aligner les sorties du modèle optique sur les étiquettes désirées. La séquence d'étiquettes étant à une seule dimension contre deux pour l'image convoluée, il conviendra d'effondrer la dimension verticale du signal, à l'aide d'un pooling par exemple.

3.2.2 Modèle d'attention pour l'accès au lexique en reconnaissance d'écriture

Ces dernières années, la reconnaissance optique de caractères basée sur les modèles profonds à bénéficié de progrès considérables, dépassant probablement les performances humaines brutes. En revanche, les modèles profonds pour la modélisation du langage sont encore limités, et peinent à atteindre les performances des modèles de langues traditionnels. Un des verrous concerne la prise en charge des lexiques de très grande taille, problème que nous avons abordé dans les travaux de Bruno Stuner [47].

Parallèlement à ces avancées, on a vu apparaitre avec un certain succès les modèles d'attention [31] qui permettent d'adapter la vue d'un modèle sur le signal d'entrée en fonction du contexte. Ces modèles d'attention sont très utilisés pour les applications de génération de légende [74] ou en traduction automatique [67, 20] par exemple, où l'ordre et le nombre de mots dans la langue source est généralement différent de ceux de la langue cible.

Notre proposition consiste à détourner les modèles d'attention traditionnels pour une aide au décodage avec des lexiques de grande taille. Pour cela, un modèle d'attention réaliserait le focus sur la sous partie du lexique la plus plausible étant donné i) la reconnaissance optique (l'attache aux données), et ii) le contexte, c'est à dire les caractères précédemment reconnus. Ce modèle d'attention hybride devrait ainsi permettre d'accelérer considérablement un processus de décodage ultérieur en limitant la taille du faisceau de recherche à chaque instant.

3.2.3 Apprentissage auto-supervisé

Une perspective intéressante à l'apprentissage des modèles profonds est l'apprentissage non supervisé afin de s'affranchir des besoins grandissant en données annotées. Il existe une solution ancienne consistant à pré-apprendre de manière totalement non supervisée les couches basses des modèles à l'aide d'auto-encodeurs. Ces approches sont aujourd'hui progressivement abandonnées, à cause de la difficulté à contrôler le sur-apprentissage [3]. On retrouve également ces problèmes dans les architectures plus récentes de type GAN [79], à travers l'instabilité entre le discriminateur et le générateur.

Notre point de vue est qu'un apprentissage non supervisé sera toujours confronté à l'impossibilité de vérifier la capacité de généralisation des modèles appris, celle-ci nécessitant le calcul d'une fonction de perte sur des données inconnues et annotées. Partant de ce constat, une approche alternative est l'apprentissage auto supervisé [19] (self-supervision), qui consiste à utiliser des étiquettes générées automatiquement sur une grande quantité de

3. Nous avons constaté ces difficultés lors de la thèse de Soufiane Belharbi, et avons proposé des solutions partielles via un apprentissage multitâche [4].

données, même si celles ci ne semblent pas très pertinentes au regard de la tâche considérée. Il existe plusieurs approches pour le préapprentissage d'architectures convolutives, telle que le « Jigsaw Puzzle »[42] qui consiste à faire apprendre à un modèle profond à remettre dans l'ordre les patchs d'une image mélangés aléatoirement.

En revanche, les approches auto supervisées sont moins répandues sur les architecture récurrentes [23, 30]. Nous pensons qu'il serait intéressant d'exploiter la nature séquentielle des données pour préapprendre les modèles, par exemple en cherchant à prédire le signal à l'instant suivant à partir des données courantes et passées. Outre le fait que la génération des étiquettes peut être obtenue à un coût nul, signalons qu'il peut fonctionner du passé vers le futur et inversement, autorisant l'apprentissage de réseaux bidirectionnels.

3.3 Conclusion

Après plus de quinze ans à manipuler les réseaux de neurones, nous avons étudié une grande variété de modèles, des plus simples aux plus complexes, des plus légers aux plus lourds, comprenant des connexions récurrentes, partagées, régularisées. Il est frappant de constater que quelle que soit l'architecture considérée, elle est strictement constituée des deux opérations basiques que sont la somme pondérée et l'activation non linéaire. On peut ainsi considérer l'optimisation d'un modèle neuronal comme la recherche du meilleur ordonnancement de ces deux fonctions au regard d'une fonction de perte, sous la contrainte de minimisation du nombre de connexions.

D'un point de vue plus personnel, ma convergence à l'issue de ces quinze époques passées dans l'enseignement supérieur et la recherche m'a amené à la conclusion (sur une base de validation évidemment) que j'ai eu - et j'ai toujours - la chance incroyable de travailler en équipe avec des personnes que j'apprécie et que j'estime. Pourvu que ça dure !

Références

[1] R. R. Ingle, Y. Fujii, T. Deselaers, J. Baccash et A. C. Popat. "A Scalable Handwritten Text Recognition System". In : *arXiv preprint arXiv :1904.09150* (2019) (cf. p. 86).

[2] D. Alvarez-Melis et T. S. Jaakkola. "Towards Robust Interpretability with Self-Explaining Neural Networks". In : *CoRR* abs/1806.07538 (2018) (cf. p. 85).

[3] S. Belharbi, C. Chatelain, R. Hérault, S. Adam, S. Thureau, M. Chastan et R. Modzelewski. "Spotting L3 slice in CT scans using deep convolutional network and transfer learning". In : *CAp, Rouen, France*. 2018 (cf. p. 40).

[4] S. Belharbi, R. Hérault, C. Chatelain et S. Adam. "Deep Neural Networks Regularization for Structured Output Prediction". In : *Neurocomputing* 281 (2018), p. 169-177 (cf. p. 87).

[5] H. Jin, Q. Song et X. Hu. "Efficient Neural Architecture Search with Network Morphism". In : *CoRR* abs/1806.10282 (2018) (cf. p. 85).

[6] R. Modzelewski, C. Chatelain, R. Herault, F. Jardin, S. Thureau et P. Vera. "BodyComp.ai : Toward automatic L3 computed tomography bodycomposition quantification in clinical practice ?" In : *International Conference on Frailty and Sarcopenia Research*. 2018 (cf. p. 40).

[7] G. Renton, Y. Soullard, C. Chatelain, S. Adam, C. Kermorvant et T. Paquet. "Fully Convolutional Network with dilated convolutions for Handwritten text line segmentation". In : *International Journal on Document Analysis and Recognition (IJDAR)* (2018) (cf. p. 48, 54).

[8] B. Stuner. "Cohorte de Réseaux de Neurones Récurrents pour la Reconnaissance de l'Écriture". Thèse de doct. Université de Rouen Normandie, 2018 (cf. p. 60, 61, 74, 80, 82).

[9] M. Yousef, K. F. Hussain et U. S. Mohammed. "Accurate, Data-Efficient, Unconstrained Text Recognition with Convolutional Neural Networks". In : *arXiv preprint arXiv :1812.11894* (2018) (cf. p. 86).

[10] Q. Zhang, Y. Nian Wu et S.-C. Zhu. "Interpretable Convolutional Neural Networks". In : *CVPR*. 2018 (cf. p. 85).

[11] B. Zoph, V. Vasudevan, J. Shlens et Q. V. Le. "Learning transferable architectures for scalable image recognition". In : *CVPR*. 2018, p. 8697-8710 (cf. p. 85).

[12] S. Belgacem, C. Chatelain et T. Paquet. "Gesture sequence recognition with one shot learned CRF/HMM hybrid model". In : *Image and Vision Computing* 61 (2017), p. 12-21 (cf. p. 40).

[13] T. Bluche, J. Louradour et R. O. Messina. "Scan, Attend and Read : End-to-End Handwritten Paragraph Recognition with MDLSTM Attention". In : *14th IAPR*

International Conference on Document Analysis and Recognition, Kyoto, Japan. 2017, p. 1050-1055 (cf. p. 59).

[14] T. Bluche et R. Messina. "Gated convolutional recurrent neural networks for multilingual handwriting recognition". In : *ICDAR*. T. 1. IEEE. 2017, p. 646-651 (cf. p. 86).

[15] L. C. Chen, G. Papandreou, F. Schroff et H. Adam. "Rethinking Atrous Convolution for Semantic Image Segmentation". In : *arXiv preprint :1706.05587* (2017) (cf. p. 51).

[16] F. Chollet. "Xception : Deep Learning With Depthwise Separable Convolutions". In : *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017 (cf. p. 56, 86).

[17] Y. N. Dauphin, A. Fan, M. Auli et D. Grangier. "Language modeling with gated convolutional networks". In : *ICML*. 2017, p. 933-941 (cf. p. 86).

[18] H. Ding, K. Chen, Y. Yuan, M. Cai, L. Sun, S. Liang et Q. Huo. "A Compact CNN-DBLSTM Based Character Model for Offline Handwriting recognition with Tucker Decomposition". In : *14th IAPR International Conference on Document Analysis and Recognition, ICDAR*. 2017, p. 507-512 (cf. p. 60).

[19] C. Doersch et A. Zisserman. "Multi-task Self-Supervised Visual Learning". In : *CoRR* abs/1708.07860 (2017) (cf. p. 87).

[20] J. Gehring, M. Auli, D. Grangier, D. Yarats et Y. N. Dauphin. "Convolutional sequence to sequence learning". In : *ICML*. 2017, p. 1243-1252 (cf. p. 85, 87).

[21] T. Grüning, R. Labahn, M. Diem, F. Kleber et S. Fiel. "READ-BAD : A New Dataset and Evaluation Scheme for Baseline Detection in Archival Documents". In : *arXiv preprint :1705.03311* (2017) (cf. p. 53).

[22] N. P. Jouppi, C. Young, N. Patil, D. Patterson, G. Agrawal, R. Bajwa, S. Bates, S. Bhatia, N. Boden, A. Borchers et al. "In-datacenter performance analysis of a tensor processing unit". In : *2017 ACM/IEEE 44th Annual International Symposium on Computer Architecture (ISCA)*. IEEE. 2017, p. 1-12 (cf. p. 85).

[23] D. Pathak, P. Agrawal, A. Efros et T. Darrell. "Curiosity-Driven Exploration by Self-Supervised Prediction". In : *CVPR Workshops*. Juil. 2017 (cf. p. 88).

[24] C. Peng, X. Zhang, G. Yu, G. Luo et J. Sun. "Large Kernel Matters–Improve Semantic Segmentation by Global Convolutional Network". In : *arXiv preprint :1703.02719* (2017) (cf. p. 49).

[25] J. Puigcerver. "Are Multidimensional Recurrent Layers Really Necessary for Handwritten Text Recognition ?" In : (2017), p. 67-72 (cf. p. 60).

[26] G. Renton, C. Chatelain, S. Adam, C. Kermorvant et T. Paquet. "Handwritten text line segmentation using Fully Convolutional Network". In : *ICDAR Workshop on Machine Learning*. 2017 (cf. p. 51).

[27] J. A. Sánchez, V. Romero, A. H. Toselli, M. Villegas et E. Vidal. "ICDAR2017 Competition on Handwritten Text Recognition on the READ Dataset". In : *14th*

IAPR International Conference on Document Analysis and Recognition (ICDAR). T. 01. 2017, p. 1383-1388 (cf. p. 58, 63).

[28] B. Stuner, C. Chatelain et T. Paquet. "LV-ROVER : Lexicon Verified Recognizer Output Voting Error Reduction". In : *CoRR* abs/1707.07432 (2017) (cf. p. 82).

[29] C. Szegedy, S. Ioffe, V. Vanhoucke et A. A. Alemi. "Inception-v4, inception-resnet and the impact of residual connections on learning". In : *AAAI Conference on Artificial Intelligence*. 2017 (cf. p. 86).

[30] H. F. Tung, H. Tung, E. Yumer et K. Fragkiadaki. "Self-supervised Learning of Motion Capture". In : *CoRR* abs/1712.01337 (2017) (cf. p. 88).

[31] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser et I. Polosukhin. "Attention is all you need". In : *NIPS*. 2017, p. 5998-6008 (cf. p. 87).

[32] C. Wigington, S. Stewart, B. L. Davis, B. Barrett, B. L. Price et S. Cohen. "Data Augmentation for Recognition of Handwritten Words and Lines Using a CNN-LSTM Network". In : *14th IAPR International Conference on Document Analysis and Recognition, ICDAR*. 2017, p. 639-645 (cf. p. 60).

[33] Y. Zhang, P. David et B. Gong. "Curriculum Domain Adaptation for Semantic Segmentation of Urban Scenes". In : *CoRR* abs/1707.09465 (2017) (cf. p. 29).

[34] S. Belharbi, R. Hérault, C. Chatelain et S. Adam. "Deep multi-task learning with evolving weights". In : *European Symposium on Artificial Neural Networks (ESANN)*. Bruges, Belgium, 2016 (cf. p. 31, 32, 34).

[35] J. Bradbury, S. Merity, C. Xiong et R. Socher. "Quasi-recurrent neural networks". In : *arXiv preprint arXiv :1611.01576* (2016) (cf. p. 85, 86).

[36] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy et A. L. Yuille. "Deeplab : Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs". In : *arXiv preprint :1606.00915* (2016) (cf. p. 51).

[37] I. Goodfellow, Y. Bengio et A. Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016 (cf. p. 26).

[38] L. G. Hafemann, R. Sabourin et L. S. Oliveira. "Writer-independent Feature Learning for Offline Signature Verification using Deep Convolutional Neural Networks". In : *CoRR* abs/1604.00974 (2016) (cf. p. 44).

[39] K. He, X. Zhang, S. Ren et J. Sun. "Deep Residual Learning for Image Recognition". In : *CVPR*. IEEE Computer Society, 2016, p. 770-778 (cf. p. 26).

[40] J. Hoffman, D. Wang, F. Yu et T. Darrell. "FCNs in the Wild : Pixel-level Adversarial and Constraint-based Adaptation". In : *CoRR* abs/1612.02649 (2016) (cf. p. 29).

[41] K. Kawaguchi. "Deep Learning without Poor Local Minima". In : *Advances in Neural Information Processing Systems 29*. Sous la dir. de D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon et R. Garnett. Curran Associates, Inc., 2016, p. 586-594 (cf. p. 77).

[42] M. Noroozi et P. Favaro. “Unsupervised learning of visual representations by solving jigsaw puzzles”. In : *European Conference on Computer Vision*. Springer. 2016, p. 69-84 (cf. p. 88).

[43] A. Poznanski et L. Wolf. “CNN-N-Gram for Handwriting Word Recognition”. In : *CVPR*. 2016, p. 2305-2314 (cf. p. 76, 80).

[44] J. Qiu, J. Wang, S. Yao, K. Guo, B. Li, E. Zhou, J. Yu, T. Tang, N. Xu, S. Song et al. “Going deeper with embedded fpga platform for convolutional neural network”. In : *Proceedings of the 2016 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*. ACM. 2016, p. 26-35 (cf. p. 85).

[45] J. A. Sánchez, V. Romero, A. H. Toselli et E. Vidal. “ICFHR2016 Competition on Handwritten Text Recognition on the READ Dataset”. In : *Frontiers in Handwriting Recognition (ICFHR), 2016 15th International Conference on*. IEEE. 2016, p. 630-635 (cf. p. 57).

[46] H. C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura et R. M. Summers. “Deep Convolutional Neural Networks for Computer-Aided Detection : CNN Architectures, Dataset Characteristics and Transfer Learning”. In : *IEEE Transactions on Medical Imaging* 35.5 (2016), p. 1285-1298 (cf. p. 44).

[47] B. Stuner, C. Chatelain et T. Paquet. “Cohort of LSTM and lexicon verification for handwriting recognition with gigantic lexicon”. In : *CoRR* abs/1612.07528 (2016) (cf. p. 87).

[48] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens et Z. Wojna. “Rethinking the Inception Architecture for Computer Vision”. In : *CVPR*. 2016 (cf. p. 85, 86).

[49] Q. N. Vo et G. Lee. “Dense prediction for text line segmentation in handwritten document images”. In : *ICIP*. 2016, p. 3264-3268 (cf. p. 49).

[50] P. Voigtlaender, P. Doetsch et H. Ney. “Handwriting recognition with large multidimensional long short-term memory recurrent neural networks”. In : *Frontiers in Handwriting Recognition (ICFHR), 2016 15th International Conference on*. IEEE. 2016, p. 228-233 (cf. p. 59, 82, 83).

[51] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu et X. Bai. “Multi-oriented text detection with fully convolutional networks”. In : *arXiv preprint arXiv :1604.04018* (2016) (cf. p. 49).

[52] V. Badrinarayanan, A. Kendall et R. Cipolla. “Segnet : A deep convolutional encoder-decoder architecture for image segmentation”. In : *arXiv preprint :1511.00561* (2015) (cf. p. 50).

[53] Y. Bar, I. Diamant, L. Wolf et H. Greenspan. “Deep learning with non-medical training used for chest pathology identification”. In : *Proc. SPIE, Medical Imaging : Computer-Aided Diagnosis* 9414 (2015), p. 94140V-7 (cf. p. 44).

[54] G. Bideault. “Détection de mots clés et d’expressions régulières en vue de la reconnaissance d’entités nommées dans des documents manuscrits”. Thèse de doct. Université de Rouen Normandie, 2015 (cf. p. 64, 72).

[55] G. Bideault, L. Mioulet, C. Chatelain et T. Paquet. “A Hybrid BLSTM-HMM for spotting Regular Expressions”. In : *ICPRAM*. Lisbon, Portugal, 2015, p. 5-12 (cf. p. 64).

[56] G. Bideault, L. Mioulet, C. Chatelain et T. Paquet. “Spotting handwritten words and REGEX using a two stage BLSTM-HMM architecture”. In : *Document Recognition and Retrieval*. T. 9402. San Francisco, USA, 2015, 94020G-11 (cf. p. 64, 71).

[57] G. Bideault, L. Mioulet, C. Chatelain et T. Paquet. “Using BLSTM for Spotting Regular Expressions in Handwritten Documents”. In : *Pattern Recognition : Applications and Methods : 4th International Conference, ICPRAM 2015, Lisbon, Portugal, January 10-12, 2015, Revised Selected Papers*. Sous la dir. d’A. Fred, M. De Marsico et M. Figueiredo. Cham : Springer International Publishing, 2015, p. 143-157. isbn : 978-3-319-27677-9 (cf. p. 64, 71).

[58] A. Choromanska, M. Henaff, M. Mathieu, G. B. Arous et Y. LeCun. “The Loss Surfaces of Multilayer Networks.” In : *AISTATS*. 2015 (cf. p. 77).

[59] T. Hastie, R. Tibshirani et M. Wainwright. *Statistical Learning with Sparsity : The Lasso and Generalizations*. Chapman & Hall/CRC, 2015. isbn : 1498712169, 9781498712163 (cf. p. 27).

[60] M. Havaei, A. Davy, D. Warde-Farley, A. Biard, A. Courville, Y. Bengio, C. Pal, P. Jodoin et H. Larochelle. “Brain Tumor Segmentation with Deep Neural Networks”. In : *CoRR* abs/1505.03540 (2015) (cf. p. 44).

[61] S. Ioffe et C. Szegedy. “Batch Normalization : Accelerating Deep Network Training by Reducing Internal Covariate Shift”. In : *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*. 2015, p. 448-456 (cf. p. 26, 28).

[62] X. Jiang. “Representational Transfer in Deep Belief Networks”. In : *28th Canadian Conference on Artificial Intelligence*. 2015, p. 338-342 (cf. p. 44).

[63] M. Lai. “Deep Learning for Medical Image Segmentation”. In : *CoRR* abs/1505.02000 (2015) (cf. p. 44).

[64] Y. LeCun, Y. Bengio et G. Hinton. “Deep learning”. In : *Nature* 521.7553 (2015), p. 436-444 (cf. p. 24, 25).

[65] J. Lerouge, R. Herault, C. Chatelain, F. Jardin et R. Modzelewski. “IODA : An Input Output Deep Architecture for image labeling”. In : *Pattern Recognition* 48.9 (2015), p. 2847-2858 (cf. p. 35).

[66] J. Long, E. Shelhamer et T. 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*. 2015, p. 3431-3440 (cf. p. 29, 49).

[67] M. Luong, H. Pham et C. Manning. “Effective approaches to attention-based neural machine translation”. In : *arXiv preprint arXiv :1508.04025* (2015) (cf. p. 87).

[68] L. Mioulet, G. Bideault, C. Chatelain, T. Paquet et S. Brunessaux. "Exploring multiple feature combination strategies with a recurrent neural network architecture for off-line handwriting recognition". In : *Document Recognition and Retrieval*. 2015, 94020F-94020F (cf. p. 79).

[69] H. Noh, S. Hong et B. Han. "Learning Deconvolution Network for Semantic Segmentation". In : *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. 2015, p. 1520-1528 (cf. p. 29, 50).

[70] A. Radford, L. Metz et S. Chintala. "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks". In : *CoRR* abs/1511.06434 (2015) (cf. p. 39).

[71] O. Ronneberger, P. Fischer et T. Brox. "U-Net : Convolutional Networks for Biomedical Image Segmentation". In : *CoRR* abs/1505.04597 (2015). arXiv : 1505.04597 (cf. p. 29, 48).

[72] S. Shuai, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang et P. Torr. "Conditional random fields as recurrent neural networks". In : *ICCV*. 2015, p. 1529-1537 (cf. p. 49).

[73] S. Thomas, C. Chatelain, L. Heutte, T. Paquet et Y. Kessentini. "A Deep HMM model for multiple keywords spotting in handwritten documents". In : *Pattern Analysis and Applications* 18.4 (2015), p. 1003-1015 (cf. p. 64, 65).

[74] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel et Y. Bengio. "Show, attend and tell : Neural image caption generation with visual attention". In : *ICML*. 2015, p. 2048-2057 (cf. p. 87).

[75] C. Yip, C. Dinkel, A. Mahajan, M. Siddique, G. Cook et V. Goh. "Imaging body composition in cancer patients : visceral obesity, sarcopenia and sarcopenic obesity may impact on clinical outcome". In : *Insights into Imaging* (2015), p. 489-497 (cf. p. 40).

[76] F. Yu et V. Koltun. "Multi-scale context aggregation by dilated convolutions". In : *arXiv preprint arXiv :1511.07122* (2015) (cf. p. 51).

[77] J. L. Atkins, P. H. Whincup, R. W. Morris, L. Lennon, O. Papacosta et S. G. Wannamethee. "Sarcopenic obesity and risk of cardiovascular disease and mortality : a population-based cohort study of older men." In : *Journal of the American Geriatrics Society* 62.2 (2014), p. 253-60 (cf. p. 40).

[78] A. Dosovitskiy, J. T. Springenberg, M. Riedmiller et T. Brox. "Discriminative Unsupervised Feature Learning with Convolutional Neural Networks". In : *Advances in Neural Information Processing Systems 27*. 2014, p. 766-774 (cf. p. 39).

[79] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville et Y. Bengio. "Generative Adversarial Nets". In : (2014), p. 2672-2680 (cf. p. 87).

[80] A. Graves et N. 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. 2014, p. 1764-1772 (cf. p. 29, 63).

[81] K. He, X. Zhang, S. Ren et J. Sun. "Spatial pyramid pooling in deep convolutional networks for visual recognition". In : *European conference on computer vision*. Springer. 2014, p. 346-361 (cf. p. 48).

[82] D. Hebert, T. Palfray, S. Nicolas, P. Tranouez et T. Paquet. "PIVAJ : displaying and augmenting digitized newspapers on the web experimental feedback from the "Journal de Rouen" collection". In : *DATeCH*. ACM, 2014, p. 173-178 (cf. p. 57).

[83] H. Lanic, J. Kraut-Tauzia, R. Modzelewski, F. Clatot, S. Mareschal, J.-M. Picquenot, A. Stamatoullas, S. Leprêtre, H. Tilly et F. Jardin. "Sarcopenia is an independent prognostic factor in elderly patients with diffuse large B-cell lymphoma treated with immunochemotherapy". In : *Leukemia & Lymphoma* 55.4 (2014), p. 817-823 (cf. p. 35, 40).

[84] S. Liu, N. Yang, M. Li et M. Zhou. "A Recursive Recurrent Neural Network for Statistical Machine Translation". In : *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*. Baltimore, Maryland : Association for Computational Linguistics, 2014, p. 1491-1500 (cf. p. 29).

[85] S. N., G. Hinton, A. Krizhevsky, I. Sutskever et R. Salakhutdinov. "Dropout : A Simple Way to Prevent Neural Networks from Overfitting". In : *Journal of Machine Learning Research* 15 (2014), p. 1929-1958 (cf. p. 26, 28).

[86] V. Pham, T. Bluche, C. Kermorvant et J. Louradour. "Dropout improves recurrent neural networks for handwriting recognition". In : *International Conference on Frontiers in Handwriting Recognition*. 2014, p. 285-290 (cf. p. 59, 82, 83).

[87] H. R. Roth, J. Yao, L. Lu, J. Stieger, J. E. Burns et R. M. Summers. "Detection of Sclerotic Spine Metastases via Random Aggregation of Deep Convolutional Neural Network Classifications". In : *CoRR* abs/1407.5976 (2014) (cf. p. 44).

[88] K. Simonyan et A. Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In : *CoRR* abs/1409.1556 (2014) (cf. p. 27, 46, 53).

[89] I. Sutskever, O. Vinyals et Q. V. Le. "Sequence to sequence learning with neural networks". In : *NIPS*. 2014, p. 3104-3112 (cf. p. 29).

[90] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke et A. Rabinovich. "Going Deeper with Convolutions". In : *CoRR* abs/1409.4842 (2014) (cf. p. 46).

[91] G. Urban, M. Bendszus, F. A. Hamprecht et J. Kleesiek. "Multi-modal Brain Tumor Segmentation using Deep Convolutional Neural Networks". In : *MICCAI BraTS Challenge Proceedings*. 2014, p. 31-35 (cf. p. 44).

[92] F. Zamora-Martinez, V. Frinken, S. España-Boquera, M. J. Castro-Bleda, A. Fischer et H. Bunke. "Neural network language models for off-line handwriting recognition". In : *Pattern Recognition* 47.4 (2014), p. 1642-1652 (cf. p. 59, 63).

[93] J. Zhang, S. Shan, M. Kan et X. Chen. "Coarse-to-Fine Auto-Encoder Networks (CFAN) for Real-Time Face Alignment". In : *ECCV, Part II*. 2014, p. 1-16 (cf. p. 32).

[94] M. Auli, M. Galley, C. Quirk et G. 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, Seattle, Washington, USA*. 2013, p. 1044-1054 (cf. p. 29).

[95] C. Chelba, T. Mikolov, M. Schuster, Q. Ge, T. Brants, P. Koehn et T. Robinson. "One billion word benchmark for measuring progress in statistical language modeling". In : *arXiv preprint arXiv :1312.3005* (2013) (cf. p. 81).

[96] C. Farabet, C. Couprie, L. Najman et Y. LeCun. "Learning Hierarchical Features for Scene Labeling". In : *IEEE PAMI* 35.8 (2013), p. 1915-1929 (cf. p. 29).

[97] B. Glocker, D.Zikic, E.Konukoglu, D. Haynor et A. Criminisi. "Vertebrae localization in pathological spine CT via dense classification from sparse annotations." In : *MICCAI* 16.Pt 2 (2013), p. 262-70 (cf. p. 42).

[98] S. Gouérant, M. Leheurteur, M. Chaker, R. Modzelewski, O. Rigal, C. Veyret, G. Lauridant et F. Clatot. "A higher body mass index and fat mass are factors predictive of docetaxel dose intensity." In : *Anticancer research* 33.12 (2013), p. 5655 (cf. p. 40).

[99] G. B. Huang et V. Jain. "Deep and Wide Multiscale Recursive Networks for Robust Image Labeling." In : *CoRR* abs/1310.0354 (2013) (cf. p. 44).

[100] T. Kaido, K. Ogawa, Y. Fujimoto, Y. Ogura, K. Hata, T. Ito, K. Tomiyama, S. Yagi, A. Mori et S. Uemoto. "Impact of sarcopenia on survival in patients undergoing living donor liver transplantation". In : *American Journal of Transplantation* 13.6 (2013), p. 1549-1556 (cf. p. 40).

[101] Y. Kessentini, C. Chatelain et T. Paquet. "Word Spotting and Regular Expression Detection in Handwritten Documents". In : *ICDAR, Washington DC, USA*. 2013, p. 516-520 (cf. p. 71).

[102] L. Martin, L. Birdsell, N. MacDonald, T. Reiman, M. T. Clandinin, L. J. McCargar, R. Murphy, S. Ghosh, M. B. Sawyer et V. E. Baracos. "Cancer Cachexia in the Age of Obesity : Skeletal Muscle Depletion Is a Powerful Prognostic Factor, Independent of Body Mass Index". In : *Journal of Clinical Oncology* 31.12 (2013), p. 1539-1547 (cf. p. 35, 40).

[103] B. Michael Kelm, M. Wels, S. Kevin Zhou, S. Seifert, M. Suehling, Y. Zheng et D. Comaniciu. "Spine detection in CT and MR using iterated marginal space learning". In : *Medical Image Analysis* 17.8 (2013), p. 1283-1292 (cf. p. 42).

[104] T. Mikolov, K. Chen, G. Corrado et J. Dean. "Efficient estimation of word representations in vector space". In : *arXiv preprint arXiv :1301.3781* (2013) (cf. p. 62).

[105] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado et J. Dean. “Distributed representations of words and phrases and their compositionality”. In : *Advances in neural information processing systems*. 2013, p. 3111-3119 (cf. p. 62).

[106] R. Pascanu, T. Mikolov et Y. Bengio. “On the difficulty of training recurrent neural networks”. In : *ICML (3)*. T. 28. JMLR Workshop and Conference Proceedings. 2013, p. 1310-1318 (cf. p. 59).

[107] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou et M. Pantic. “A semi-automatic methodology for facial landmark annotation”. In : *CVPR Workshops*. 2013, p. 896-903 (cf. p. 33).

[108] M. Sundermeyer, I. Oparin, J. L. Gauvain, B. Freiberg, R. Schlüter et H. Ney. “Comparison of feedforward and recurrent neural network language models”. In : *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. 2013, p. 8430-8434 (cf. p. 63).

[109] S. Thomas, C. Chatelain, L. Heutte et T. Paquet. “Un modèle neuro markovien profond pour l’extraction de séquences dans des documents manuscrits”. In : *Documents Numériques* 16.2 (2013), p. 49-69 (cf. p. 64, 65).

[110] S. Wager, S. Wang et P. S. Liang. “Dropout Training as Adaptive Regularization”. In : *Advances in Neural Information Processing Systems 26*. 2013, p. 351-359 (cf. p. 26).

[111] B. Zhang. “Reliable classification of vehicle types based on cascade classifier ensembles”. In : *Intelligent Transportation Systems* 14.1 (2013), p. 322-332 (cf. p. 76).

[112] D. C. Cireşan, U. Meier et J. Schmidhuber. “Transfer learning for Latin and Chinese characters with Deep Neural Networks”. In : *International Joint Conference on Neural Networks*. 2012, p. 1-6 (cf. p. 27, 44).

[113] A. Fischer, A. Keller, V. Frinken et H. Bunke. “Lexicon-free handwritten word spotting using character HMMs”. In : *Pattern Recognition Letters* 33.7 (2012), p. 934-942 (cf. p. 64, 69).

[114] V. Frinken, A. Fischer, R. Manmatha et H. Bunke. “A Novel Word Spotting Method Based on Recurrent Neural Networks”. In : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34 (2012), p. 211-224. issn : 0162-8828 (cf. p. 64, 69).

[115] B. Glocker, J. Feulner, A. Criminisi, D. R. Haynor et E. Konukoglu. “Automatic Localization and Identification of Vertebrae in Arbitrary Field-of-View CT Scans”. In : *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2012 : 15th International Conference, Nice, France, October 1-5, 2012, Proceedings, Part III*. Berlin, Heidelberg, 2012, p. 590-598 (cf. p. 42).

[116] A. Krizhevsky, I. Sutskever et G. Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In : *Advances in Neural Information Processing Systems 25*. 2012, p. 1097-1105 (cf. p. 26, 40, 46).

[117] V. Le, J. Brandt, Z. Lin, L. D. Bourdev et T. S. Huang. “Interactive Facial Feature Localization”. In : *ECCV, 2012, Proceedings, Part III*. 2012, p. 679-692 (cf. p. 33).

[118] F. Menasri, J. Louradour, A.-L. Bianne-Bernard et C. Kermorvant. “The A2iA French handwriting recognition system at the Rimes-ICDAR2011 competition”. In : *Document Recognition and Retrieval XIX*. 2012, 82970Y-82970Y (cf. p. 77, 80).

[119] T. Mikolov. “Statistical language models based on neural networks”. In : *Brno University of Technology* (2012) (cf. p. 59).

[120] T. Paquet, L. Heutte, G. Koch et C. Chatelain. “A Categorization System for Handwritten Documents”. In : *International Journal on Document Analysis and Recognition (IJDAR)* 15.4 (2012), p. 315-330 (cf. p. 64).

[121] S. Thomas. “Extraction d’information dans des courriers manuscrits”. Thèse de doct. Université de Rouen Normandie, 2012 (cf. p. 64).

[122] S. Thomas, C. Chatelain, L. Heutte et T. Paquet. “Combinaison architecture profonde/HMM pour l’extraction de sequences dans des documents manuscrits”. In : *CIFED, Bordeaux, France*. 2012 (cf. p. 64).

[123] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman et N. Kumar. “Localizing parts of faces using a consensus of exemplars.” In : *CVPR*. IEEE, 2011, p. 545-552 (cf. p. 33).

[124] A.-L. Bianne-Bernard, F. Menasri, R. A.-H. Mohamad, C. Mokbel, C. Kermorvant et L. Likforman-Sulem. “Dynamic and contextual information in HMM modeling for handwritten word recognition”. In : *IEEE transactions on pattern analysis and machine intelligence* 33.10 (2011), p. 2066-2080 (cf. p. 59).

[125] S. Ghosh, R. Alomari, V. Chaudhary et G. Dhillon. “Automatic lumbar vertebra segmentation from clinical CT for wedge compression fracture diagnosis”. In : *Proceedings of the SPIE* 3 (2011), p. 796303-9 (cf. p. 42).

[126] E. Grosicki et H. El-Abed. “ICDAR 2011-french handwriting recognition competition”. In : *ICDAR*. 2011, p. 1459-1463 (cf. p. 72).

[127] S. Kadoury, H. Labelle et N. Paragios. “Automatic inference of articulated spine models in CT images using high-order Markov Random Fields”. In : *Medical Image Analysis* 15.4 (2011), p. 426-437 (cf. p. 42).

[128] P. Peng et al. “Sarcopenia negatively impacts short-term outcomes in patients undergoing hepatic resection for colorectal liver metastasis”. In : *HPB* 13.7 (2011), p. 439-446. issn : 1365-182X (cf. p. 40).

[129] A. Fischer, A. Keller, V. Frinken et H. Bunke. “HMM-based Word Spotting in Handwritten Documents Using Subword Models”. In : *International Conference on Pattern Recognition* (2010), p. 3416-3419. issn : 1051-4651 (cf. p. 57).

[130] X. Glorot et Y. Bengio. “Understanding the difficulty of training deep feedforward neural networks”. In : *International conference on artificial intelligence and statistics*. 2010, p. 249-256 (cf. p. 26).

[131] V. Nair et G. E. Hinton. “Rectified Linear Units Improve Restricted Boltzmann Machines”. In : *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*. Omnipress, 2010, p. 807-814 (cf. p. 25).

[132] L. Nanni, A. Lumini et S. Brahnam. “Local binary patterns variants as texture descriptors for medical image analysis”. In : *Artificial Intelligence in Medicine* 49.2 (2010), p. 117-125 (cf. p. 46).

[133] S. J. Pan et Q. Yang. “A Survey on Transfer Learning”. In : *IEEE Trans. on Knowl. and Data Eng.* 22.10 (2010), p. 1345-1359. issn : 1041-4347 (cf. p. 27).

[134] S. Thomas, C. Chatelain, L. Heutte et T. Paquet. “Alpha-numerical sequences extraction in handwritten documents”. In : *International Conference on Frontiers in Handwriting Recognition (ICFHR), Kolkata, India*. 2010, p. 232-237 (cf. p. 64, 65).

[135] S. Thomas, C. Chatelain, L. Heutte et T. Paquet. “An Information Extraction model for unconstrained handwritten documents”. In : *International Conference on Pattern Recognition (ICPR), Istanbul, Turkey*. 2010, p. 4 (cf. p. 57, 64, 65).

[136] S. Thomas, C. Chatelain, L. Heutte et T. Paquet. “Un systeme generique d’extraction d’information dans des documents manuscrits non-contraints”. In : *CIFED, Sousse, Tunisia*. 2010 (cf. p. 64, 65).

[137] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio et P. Manzagol. “Stacked Denoising Autoencoders : Learning Useful Representations in a Deep Network with a Local Denoising Criterion”. In : *JMLR* 11 (2010), p. 3371-3408 (cf. p. 27, 29).

[138] Y. Bengio. “Learning Deep Architectures for AI”. In : *Found. Trends Mach. Learn.* 2.1 (2009), p. 1-127. issn : 1935-8237 (cf. p. 25, 30).

[139] H. Chung, D. Cobzas, L. Birdsell, J. Lieffers et V. Baracos. “Automated segmentation of muscle and adipose tissue on CT images for human body composition analysis”. In : *Proc. SPIE* 7261 (2009) (cf. p. 37).

[140] J. Deng, W. Dong, R. Socher, L. Li, K. Li et F.-F. Li. “ImageNet : A large-scale hierarchical image database.” In : *CVPR*. 2009, p. 248-255 (cf. p. 46).

[141] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke et J. Schmidhuber. “A novel connectionist system for unconstrained handwriting recognition”. In : *IEEE transactions on pattern analysis and machine intelligence* 31.5 (2009), p. 855-868 (cf. p. 29, 60).

[142] A. Graves et J. Schmidhuber. “Offline handwriting recognition with multidimensional recurrent neural networks”. In : *NIPS*. 2009, p. 545-552 (cf. p. 59, 60, 79).

[143] E. Grosicki et H. El Abed. “Icdar 2009 handwriting recognition competition”. In : *ICDAR*. 2009, p. 1398-1402 (cf. p. 59, 60, 79).

[144] T. Plötz et G. A. Fink. “Markov models for offline handwriting recognition : a survey”. In : *International Journal on Document Analysis and Recognition* 12.4 (2009), p. 269-298 (cf. p. 59).

[145] H. Zen, K. Tokuda et A. Black. “Statistical parametric speech synthesis”. In : *Speech Communication* 51.11 (2009), p. 1039-1064 (cf. p. 29).

[146] M. B. Blaschko et C. H. Lampert. “Learning to Localize Objects with Structured Output Regression”. In : *ECCV 2008*. 2008, p. 2-15 (cf. p. 29).

[147] C. Chatelain, L. Heutte et T. Paquet. “Recognition-based vs syntax-directed models for numerical field extraction in handwritten documents”. In : *International Conference on Frontiers in Handwriting Recognition (ICFHR), Montreal, Canada*. 2008, p. 159-164 (cf. p. 64, 71).

[148] C. Malon, M. Miller, H. Burger, E. Cosatto et H. P. Graf. “Identifying Histological Elements with Convolutional Neural Networks”. In : *Int. Conf. on Soft Computing As Transdisciplinary Science and Technology*. 2008, p. 450-456 (cf. p. 44).

[149] J. A. Rodriguez et F. Perronnin. “Local gradient histogram features for word spotting in unconstrained handwritten documents”. In : *Int. Conf. on Frontiers in Handwriting Recognition*. 2008 (cf. p. 72).

[150] P. Vincent, L. Hugo, Y. Bengio et P.-A. Manzagol. “Extracting and composing robust features with denoising autoencoders”. In : *Proceedings of the 25th international conference on Machine learning*. ICML ’08. 2008, p. 1096-1103. isbn : 978-1-60558-205-4 (cf. p. 27).

[151] Y. Bengio, P. Lamblin, D. Popovici et H. Larochelle. “Greedy Layer-Wise Training of Deep Networks”. In : *NIPS*. Sous la dir. de B. Schölkopf, J. Platt et T. Hoffman. 2007, p. 153-160 (cf. p. 25, 29, 30).

[152] R. Al-Hajj. “Reconnaissance hors ligne de textes manuscrits cursifs par l’utilisation de systèmes hybrides et de techniques d’apprentissage automatique”. Thèse de doct. ENST de Paris, 2007, p. 1-155 (cf. p. 68).

[153] G. E. Hinton. “Learning multiple layers of representation”. In : *Trends in Cognitive Sciences* 11 (2007), p. 428-434 (cf. p. 25, 30).

[154] L. Likforman-Sulem, A. Zahour et B. Taconet. “Text line segmentation of historical documents : a survey”. In : *International Journal of Document Analysis and Recognition (IJDAR)* 9.2-4 (2007), p. 123-138 (cf. p. 51).

[155] G. Tsechpenakis, J. Wang, B. Mayer et D. Metaxas. “Coupling CRFs and Deformable Models for 3D Medical Image Segmentation”. In : *ICCV*. 2007, p. 1-8 (cf. p. 29).

[156] P. Zhang, T. D. Bui et C. Y. Suen. “A novel cascade ensemble classifier system with a high recognition performance on handwritten digits”. In : *Pattern Recognition* 40.12 (2007), p. 3415-3429 (cf. p. 76).

[157] E. Augustin, M. Carré, E. Grosicki, J.-M. Brodin, E. Geoffrois et F. Prêteux. “RIMES evaluation campaign for handwritten mail processing”. In : *International Workshop on Frontiers in Handwriting Recognition (IWFHR’06),* 2006, p. 231-235 (cf. p. 58, 68).

[158] C. Chatelain, L. Heutte et T. Paquet. “A two-stage outlier rejection strategy for numerical field extraction in handwritten documents”. In : *International Conference on Pattern Recognition (ICPR), Hong Kong, China*. T. 3. 2006, p. 224-227 (cf. p. 71).

[159] C. Chatelain, L. Heutte et T. Paquet. “Segmentation-driven recognition applied to numerical field extraction from handwritten incoming mail documents”. In : *Document Analysis System, Nelson, New Zealand, LNCS 3872, Springer*. 2006, p. 564-575 (cf. p. 57, 64, 68).

[160] C. Chatelain, G. Koch, L. Heutte et T. Paquet. “Une methode dirigee par la syntaxe pour l’extraction de champs numeriques dans les courriers entrants”. In : *Traitement du Signal* 23.2 (2006), p. 179-198 (cf. p. 64).

[161] C. Chatelain. “Numerical sequences extraction in handwritten documents”. Thèse de doct. Université de Rouen, déc. 2006 (cf. p. 64).

[162] A. Graves, S. Fernández, F. J. Gomez et J. Schmidhuber. “Connectionist temporal classification : labelling unsegmented sequence data with recurrent neural networks”. In : *ICML*. 2006, p. 369-376 (cf. p. 60, 86).

[163] G. E. Hinton, S. Osindero et Y.-W. Teh. “A fast learning algorithm for deep belief nets”. In : *Neural Comput.* 18.7 (2006), p. 1527-1554 (cf. p. 24, 25, 30).

[164] W. Holzinger, B. Krüpl et M. Herzog. *Using ontologies for extracting product features from web pages*. Springer, 2006 (cf. p. 70).

[165] S. Nicolas, T. Paquet et L. Heutte. “A Markovian Approach for Handwritten Document Segmentation.” In : *ICPR (3)*. 2006, p. 292-295 (cf. p. 29).

[166] N. Dalal et B. Triggs. “Histograms of oriented gradients for human detection”. In : *CVPR*. T. 1. IEEE. 2005, p. 886-893 (cf. p. 79).

[167] I. Szoke, P. Schwarz, P. Matejka, L. Burget, M. Karafiát, M. Fapso et J. Cernocky. “Comparison of keyword spotting approaches for informal continuous speech”. In : *Ninth European Conference on Speech Communication and Technology*. 2005 (cf. p. 65).

[168] A. R. Ahmad, M. Khalia, C. Viard-Gaudin et E. Poisson. “Online handwriting recognition using support vector machine”. In : *2004 IEEE Region 10 Conference TENCON 2004*. T. 1. 2004, p. 311-314 (cf. p. 59).

[169] W. Shen, M. Punyanitya, Z. Wang, D. Gallagher, M.-P. St-Onge, J. Albu, S. Heymsfield et S. Heshka. “Total body skeletal muscle and adipose tissue volumes : estimation from a single abdominal cross-sectional image.” In : *Journal of applied physiology* 97.6 (2004), p. 2333-2338 (cf. p. 40).

[170] M. Szummer et Y. Qi. “Contextual Recognition of Hand-drawn Diagrams with Conditional Random Fields”. In : *IWFHR*. 2004, p. 32-37 (cf. p. 29).

[171] A. Vinciarelli, S. Bengio et H. Bunke. “Offline Recognition of Unconstrained Handwritten Texts Using HMMs and Statistical Language Models”. In : *IEEE Trans. on PAMI* 26.6 (2004), p. 709-720 (cf. p. 64).

[172] Y. Bengio, R. Ducharme, P. Vincent et C. Janvin. “A Neural Probabilistic Language Model”. In : *J. Mach. Learn. Res.* 3 (2003), p. 1137-1155 (cf. p. 62).

[173] A. L. Koerich, R. Sabourin et C. Y. Suen. “Large vocabulary off-line handwriting recognition : A survey”. In : *Pattern Analysis & Applications* 6.2 (2003), p. 97-121 (cf. p. 61).

[174] C. L. Liu, K. Nakashima, H. Sako et H. Fujisawa. “Handwritten digit recognition : benchmarking of state-of-the-art techniques”. In : *Pattern Recognition* 36.10 (2003), p. 2271-2285 (cf. p. 59).

[175] M. E. Morita, R. Sabourin, F. Bortolozzi et C. Y. Suen. “Segmentation and recognition of handwritten dates : an HMM-MLP hybrid approach.” In : 2003, p. 248-262 (cf. p. 70).

[176] F. J. Och. “Minimum error rate training in statistical machine translation”. In : *Proceedings of the ACL*. T. 1. 2003 (cf. p. 29).

[177] A. R. Dengel et B. Klein. “smartFIX : A requirements-driven system for document analysis and understanding”. In : *Document Analysis Systems V*. Springer, 2002, p. 433-444 (cf. p. 70).

[178] U.-V. Marti et H. Bunke. “The IAM-database : an English sentence database for offline handwriting recognition”. In : *International Journal on Document Analysis and Recognition* 5.1 (2002), p. 39-46 (cf. p. 58, 79).

[179] J. Weston, O. Chapelle, V. Vapnik, A. Elisseeff et B. Schölkopf. “Kernel dependency estimation”. In : *NIPS*. 2002, p. 873-880 (cf. p. 29).

[180] M. El-Yacoubi, M. Gilloux et J.-M. Bertille. “A statistical approach for phrase location and recognition within a text line : An application to street name recognition”. In : *IEEE PAMI* 24.2 (2002), p. 172-188 (cf. p. 57).

[181] N. Arica et F. T. Yarman-Vural. “An Overview of Character Recognition Focused on Off-line Handwriting”. In : *Trans. Sys. Man Cyber Part C* 31.2 (2001), p. 216-233 (cf. p. 59).

[182] L. Breiman. “Random Forests”. In : *Mach. Learn.* 45.1 (2001), p. 5-32. issn : 0885-6125 (cf. p. 46).

[183] J. D. Lafferty, A. McCallum et F. C. Pereira. “Conditional Random Fields : Probabilistic Models for Segmenting and Labeling Sequence Data”. In : *ICML*. USA, 2001, p. 282-289 (cf. p. 29).

[184] S. Marukatat, T. Artières, P. Gallinari et B. Dorizzi. “Sentence Recognition through hybrid neuro-markovian modeling”. In : *In International Conference on Document Analysis and Recognition (ICDAR* (2001), p. 731-735 (cf. p. 59, 67).

[185] R. Sidhu et V. K. Prasanna. “Fast regular expression matching using FPGAs”. In : *Field-Programmable Custom Computing Machines, 2001. FCCM'01. The 9th Annual IEEE Symposium on*. IEEE. 2001, p. 227-238 (cf. p. 70).

[186] P. Viola et M. Jones. “Rapid object detection using a boosted cascade of simple features”. In : *CVPR*. T. 1. IEEE. 2001, p. I-511 (cf. p. 76).

[187] R. Plamondon et S. N. Srihari. “Online and off-line handwriting recognition : a comprehensive survey”. In : *IEEE PAMI* 22.1 (2000), p. 63-84 (cf. p. 58).

[188] H. Shimodaira. “Improving predictive inference under covariate shift by weighting the log-likelihood function”. In : *Journal of Statistical Planning and Inference* 90.2 (2000), p. 227-244 (cf. p. 28).

[189] D. M. Bikel, R. Schwartz et R. M. Weischedel. “An algorithm that learns what’s in a name”. In : *Machine learning* 34.1-3 (1999), p. 211-231 (cf. p. 29).

[190] J. Foote. “An overview of audio information retrieval”. In : *Multimedia systems* 7.1 (1999), p. 2-10 (cf. p. 65).

[191] M. N. Garofalakis, R. Rastogi et K. Shim. “SPIRIT : Sequential pattern mining with regular expression constraints”. In : *VLDB*. T. 99. 1999, p. 7-10 (cf. p. 70).

[192] N. Gorski, V. Anisimov, E. Augustin, O. Baret, D. Price et J.-C. Simon. “A2ia check reader : A family of bank check recognition systems”. In : *Document Analysis and Recognition, 1999. ICDAR’99. Proceedings of the Fifth International Conference on*. IEEE. 1999, p. 523-526 (cf. p. 57).

[193] D. T. Jones. “Protein secondary structure prediction based on position-specific scoring matrices”. In : *Journal of Molecular Biology* 292.2 (1999), p. 195-202 (cf. p. 29).

[194] C. Viard-Gaudin, P. M. Lallican, S. Knerr et P. Binter. “The IRESTE On/Off (IRONOFF) dual handwriting database”. In : *Proceedings of the Fifth International Conference on Document Analysis and Recognition. ICDAR ’99 (Cat. No.PR00318)*. 1999, p. 455-458 (cf. p. 58).

[195] S. Hochreiter. “The vanishing gradient problem during learning recurrent neural nets and problem solutions”. In : *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 6.02 (1998), p. 107-116 (cf. p. 59).

[196] S. Knerr, E. Augustin, O. Baret et D. Price. “Hidden Markov model based word recognition and its application to legal amount reading on French checks”. In : *Computer Vision and Image Understanding* 70.3 (1998), p. 404-419 (cf. p. 59, 67).

[197] Y. LeCun, L. Bottou, Y. Bengio et P. Haffner. “Gradient-based learning applied to document recognition”. In : *Proceedings of the IEEE* 86.11 (1998), p. 2278-2324 (cf. p. 59).

[198] N. Mitsiopoulos, R. N. Baumgartner, S. B. Heymsfield, W. Lyons, D. Gallagher et R. Ross. “Cadaver validation of skeletal muscle measurement by magnetic resonance imaging and computerized tomography.” In : *Journal of applied physiology* 85.1 (1998), p. 115-122 (cf. p. 40).

[199] J. Fiscus. “A post-processing system to yield reduced word error rates : Recognizer output voting error reduction (ROVER)”. In : *Automatic Speech Recognition and Understanding, 1997. Proceedings., 1997 IEEE Workshop on*. IEEE. 1997, p. 347-354 (cf. p. 82).

[200] S. Hochreiter et J. Schmidhuber. “Long short-term memory”. In : *Neural computation* 9.8 (1997), p. 1735-1780 (cf. p. 59).

[201] S. Knerr, V. Asimov, O. Baret, N. Gorsky et J. S. D. Price. "The A2iA intercheque system : Courtesy amount and legal amount recognition for french checks". In : Automatic bankcheck processing. World Scientific, 1997, p. 43-86 (cf. p. 59).

[202] S. Ortmanns, A. Eiden, H. Ney et N. Coenen. "Look-ahead techniques for fast beam search". In : *Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on*. T. 3. IEEE. 1997, p. 1783-1786 (cf. p. 67).

[203] L. Spitz. "Determination of the script and language content of document images". In : *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 19.3 (1997), p. 235-245 (cf. p. 70).

[204] S. F. Chen et J. Goodman. "An empirical study of smoothing techniques for language modeling". In : *Proceedings of the 34th annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics. 1996, p. 310-318 (cf. p. 62).

[205] M. Mohammed et P. Gader. "Handwritten Word Recognition Using Segmentation-Free Hidden Markov Modeling and Segmentation-Based Dynamic Programming Techniques". In : *IEEE Trans. on Pattern Analysis and Machine Intelligence* 18.5 (1996), p. 548-554 (cf. p. 59).

[206] M. Mohri. "On some applications of finite-state automata theory to natural language processing". In : *Natural Language Engineering* 2.1 (1996), p. 61-80 (cf. p. 62).

[207] A. Senior et T. Robinson. "Forward-backward retraining of recurrent neural networks." In : *NIPS* (1996), p. 743-749 (cf. p. 59).

[208] Ø. D. Trier, A. K. Jain et T. Taxt. "Feature extraction methods for character recognition-A survey". In : *Pattern Recognition* 29.4 (1996), p. 641-662. issn : 0031-3203 (cf. p. 59).

[209] Y. Bengio, Y. L. Cun, C. Nohl et C. Burges. "LeRec : A NN-HMM hybrid for on-line handwriting recognition". In : *Neural Computation* 7.6 (1995), p. 1289-1303 (cf. p. 59).

[210] T. K. Ho. "Random Decision Forests". In : *Proceedings of the Third International Conference on Document Analysis and Recognition (Volume 1) - Volume 1*. ICDAR '95. Washington, DC, USA : IEEE Computer Society, 1995, p. 278- (cf. p. 46).

[211] Y. LeCun, Y. Bengio et al. "Convolutional networks for images, speech, and time series". In : *The handbook of brain theory and neural networks* 3361.10 (1995), p. 1995 (cf. p. 27).

[212] L. Spitz. "Using character shape codes for word spotting in document images". In : *Shape, Structure and Pattern Recognition*. 1995, p. 382-389 (cf. p. 70).

[213] V. N. Vapnik. *The Nature of Statistical Learning Theory*. New York, NY, USA : Springer-Verlag New York, Inc., 1995. isbn : 0-387-94559-8 (cf. p. 24, 25).

[214] F. Kimura, S. Tsuruoka, Y. Miyake et M. Shridhar. "A Lexicon Directed Algorithm for Recognition of Unconstrained Handwritten Words". In : *IEICE Trans. on Information & Syst.* E77-D.7 (1994), p. 785-793 (cf. p. 59).

[215] H. Schmid. “Part-of-speech tagging with neural networks”. In : *conference on Computational linguistics* 12 (1994), p. 44-49 (cf. p. 29).

[216] R. Tibshirani. “Regression Shrinkage and Selection Via the Lasso”. In : *Journal of the Royal Statistical Society, Series B* 58 (1994), p. 267-288 (cf. p. 27).

[217] M. Gilloux. “Research into the new generation of character and mailing address recognition systems at the French post office research center”. In : *Pattern Recognition Letters* 14.4 (1993), p. 267-276 (cf. p. 57).

[218] T. Paquet et Y. Lecourtier. “Recognition of handwritten sentences using a restricted lexicon”. In : *Pattern Recognition* 26.3 (1993), p. 391-407 (cf. p. 57).

[219] S. N. Srihari. “Recognition of handwritten and machine-printed text for postal address interpretation”. In : *Pattern recognition letters* 14.4 (1993), p. 291-302 (cf. p. 57).

[220] P. F. Brown, P. V. Desouza, R. L. Mercer, V. J. D. Pietra et J. C. Lai. “Class-based n-gram models of natural language”. In : *Computational linguistics* 18.4 (1992), p. 467-479 (cf. p. 62).

[221] H. Ney, D. Mergel, A. Noll et A. Paeseler. “Data driven search organization for continuous speech recognition”. In : *IEEE Transactions on Signal Processing* 40.2 (1992), p. 272-281 (cf. p. 61).

[222] J. W. Wallis et T. R. Miller. “Three-dimensional display in nuclear medicine and radiology”. In : *Journal of nuclear medicine : official publication, Society of Nuclear Medicine* 32.3 (1991), p. 534-546 (cf. p. 43).

[223] H. Baird. “Document Image Defect Models”. In : *Proceddings, IAPR Workshop on Syntactic and Structural Pattern Recognition*. Murray Hill, NJ, 1990 (cf. p. 27).

[224] M. Holschneider, R. Kronland-Martinet, J. Morlet et P. Tchamitchian. “A real-time algorithm for signal analysis with the help of the wavelet transform”. In : *Wavelets*. Springer, 1989, p. 286-297 (cf. p. 50).

[225] L. Rabiner. “A tutorial on hidden Markov models and selected applications in speech recognition”. In : *Proceedings of the IEEE* 77.2 (1989), p. 257-286 (cf. p. 29).

[226] J. W. Wallis, T. R. Miller, C. A. Lerner et E. C. Kleerup. “Three-dimensional display in nuclear medicine”. In : *IEEE Trans. on Medical Imaging* 8.4 (1989), p. 297-230 (cf. p. 43).

[227] L. Fissore, G. Micca, R. Pieraccini et P. Palace. “Strategies for lexical access to very large vocabularies”. In : *Speech Communication* 7.4 (1988), p. 355-366 (cf. p. 61).

[228] D. E. Rumelhart, G. E. Hinton et R. J. Williams. “Learning Internal Representations by Error Propagation”. In : *Parallel Distributed Processing*. Sous la dir. de D. E. Rumelhart et J. L. McClelland. T. 1. MIT Press, 1986, p. 318-362 (cf. p. 25).

[229] D. H. Ackley, G. E. Hinton et T. J. Sejnowski. “A learning algorithm for boltzmann machines*”. In : *Cognitive science* 9.1 (1985), p. 147-169 (cf. p. 59).

[230] J. Hopfield. “Neural networks and physical systems with emergent collective computational abilities”. In : *Proceedings of the national academy of sciences* 79.8 (1982), p. 2554-2558 (cf. p. 59).

[231] J. E. Hopcroft. *Introduction to automata theory, languages, and computation.* Pearson Education India, 1979 (cf. p. 70).

[232] T. L. Booth. *Sequential machines and automata theory.* T. 3. Wiley New York, 1967 (cf. p. 70).

[233] A. Tikhonov. “On solving ill-posed problem and method of regularization”. In : *Doklady Akademii Nauk USSR* 153 (1963), p. 501-504 (cf. p. 27).

[234] F. Rosenblatt. “The Perceptron : A Probabilistic Model for Information Storage and Organization in The Brain”. In : *Psychological Review* (1958), p. 65-386 (cf. p. 25).

[235] C. E. Shannon. “A mathematical theory of communication”. In : *Bell System technical journal* (1948), p. 379-423 (cf. p. 62).

[236] W. S. McCulloch et W. Pitts. “A logical calculus of the ideas immanent in nervous activity”. In : *The bulletin of mathematical biophysics* 5.4 (déc. 1943), p. 115-133 (cf. p. 25).