Réseaux neuronaux convolutifs profonds et représentations hiérarchiques : applications et perspectives pour la pathologie numérique

Arnaud Abreu

Université de Strasbourg

UNIVERSITÉ DE STRASBOURG

msii ÉCOLE DOCTORALE

ÉCOLE DOCTORALE no 269 : Mathématiques, Sciences de l'Information et de l'Ingénieur

Laboratoire ICube, équipe SDC

THÈSE présentée par :

Arnaud ABREU

soutenue le : **1er octobre 2020**

pour obtenir le grade de : **Docteur de l'université de Strasbourg**

Discipline/ Spécialité : Informatique

Réseaux neuronaux convolutifs profonds et représentations hiérarchiques : applications et perspectives pour la pathologie numérique

THÈSE dirigée par :

M. WEMMERT Cédric	Professeur, université de Strasbourg
M. BROUSSET Pierre	Professeur, CHU de Toulouse

RAPPORTEURS :

Mme FROMONT Élisa	Professeur, Université de Rennes 1
M. TALBOT Hugues	Professeur, Université Paris Saclay

AUTRES MEMBRES DU JURY :

M. HEITZ Fabrice	Professeur, Université de Strasbourg
M. WALTER Thomas	Maître de conférence, HDR, Mines ParisTech
M. FRENOIS François-Xavier	Docteur, ingénieur au CHU de Toulouse
M. NAEGEL Benoît	Maître de conférence, HDR, Université de Strasbourg

École doctorale n° 269 : Mathématiques, Sciences de l'Information et de l'Ingénieur

THÈSE

présentée pour obtenir le grade de

Docteur de l'Université de Strasbourg

Discipline "Informatique"

présentée et soutenue publiquement par

Arnaud Abreu

le 1^{er} octobre 2020

Réseaux neuronaux convolutifs profonds et représentations hiérarchiques : applications et perspectives pour la pathologie numérique

Directeur de thèse : **Cédric Wemmert**
Co-directeur de thèse : **Pierre Brousset**
Encadrant de thèse : **Benoît Naegel**
Co-encadrant de thèse : **François-Xavier Frenois**

Jury

Mme Élisa Fromont,	Professeur, Université de Rennes 1	Rapporteur
M. Hugues Talbot,	Professeur, Université Paris Saclay	Rapporteur
M. Fabrice Heitz,	Professeur, Université de Strasbourg	Examinateur
M. Thomas Walter,	Maître de Conférences HDR, Mines ParisTech	Examinateur

Remerciements

Je tiens tout d'abord à remercier Cédric Wemmert qui a accepté de diriger cette thèse. Durant cette période, il a su m'aiguiller lorsque les idées ou la créativité faisaient défaut. Je le remercie également pour sa patience et sa sérénité notamment dans la gestion des crises administratives que j'ai eu le malheur de déclencher. Je le remercie enfin pour la marge de manœuvre qu'il m'a accordée dans la définition des objectifs et des développements réalisés au cours de ce travail.

Je remercie ensuite Pierre Brousset qui co-dirigeait mes travaux. Je le remercie tout particulièrement de m'avoir accueilli au sein du laboratoire d'anatomie et de cytologie pathologique de l'Institut Universitaire du Cancer de Toulouse. Il s'est montré d'une grande patience à mon égard et a su orienter l'ensemble de mes travaux vers des applications biomédicales toujours pertinentes. En formulant les problématiques comme des défis lancés au domaine de l'informatique et de l'« intelligence artificielle », ainsi que par son exigence, il a très largement contribué à la rigueur et à la passion avec laquelle j'ai réalisé ces travaux.

Je remercie ensuite François-Xavier Frenois, qui a encadré mes travaux sur le plan plus technique de l'imagerie microscopique de pointe. Il a lui aussi fait preuve d'une patience à toute épreuve sur la plupart de mes déboires administratives. Je le remercie d'avoir partagé son bureau avec moi et de m'avoir supporté au quotidien sur toute la durée de cette thèse. Je lui suis infiniment reconnaissant de m'avoir fait découvrir en premier lieu les images de la pathologie numérique lors de mon stage de *M1*, ainsi que de m'avoir initié à des problématiques de traitement et d'analyse d'images qui ne s'étaient pas posées durant ma formation universitaire et pour lesquelles je nourris désormais une passion dévorante. Je lui dois également la quasi-totalité de mes compétences en rédaction scientifique. Enfin, je tiens à le remercier d'avoir été un mentor avisé, un ami fidèle et bientôt un collègue avec qui j'aurai grand plaisir à travailler.

Je tiens ensuite à remercier Benoît Naegel, qui a encadré cette thèse sur le plan scientifique informatique. J'ai conscience d'avoir été une source de frustration et n'avoir pas toujours su communiquer efficacement avec lui. Je le remercie donc pour l'indulgence qu'il a eu à mon égard sur ces aspects de mon travail. Je dois à Benoît mon attrait pour les graphes, les arbres et les représentations hiérarchiques. Je ne me serais naturellement pas penché vers ce domaine sans l'intérêt qu'il a suscité en moi en y faisant référence dès les débuts de mon travail. Je tiens également à le remercier pour les bases et la rigueur dans le formalisme qu'il a su m'inculquer, notamment en relisant mes articles, sur ce sujet qui m'est à présent très cher.

Je remercie encore Camille Franchet sans qui bon nombre des développements en *machine learning* de ce travail n'auraient pas pu se faire. Je le remercie pour l'effort qu'il réalise en comprenant les principes fondamentaux à l'œuvre dans mon domaine d'expertise et pour les réflexions constructives qu'il y apporte dans chaque projet que nous réalisons en commun. Je le remercie pour son esprit d'initiative quant à l'élaboration de projets d'« intelligence

artificielle » pour la pathologie et lui suis très reconnaissant pour le temps passé à annoter des lames. J'apprécie beaucoup la teneur de nos échanges scientifiques et les retombées fructueuses de nos diverses collaborations, qui, je l'espère, ne s'arrêteront pas là.

Je remercie également Salvatore Valitutti ainsi que l'ensemble des membres de son équipe de recherche du Centre de Rechercher contre le Cancer de Toulouse. Ils ont su m'apporter l'ouverture d'esprit sur le plan biologique qui échappe aux problématiques purement médicales du service d'anapath. La qualité des échanges scientifiques que nous avons pu avoir m'ont permis de mettre au point certains des concepts-clefs de cette thèse et je leur suis largement redevable pour cela.

Je remercie ensuite Myriam Marty, pour sa joie de vivre, ses conversations animées et le temps qu'elle a su m'accorder tout au long de ce travail.

Je remercie Gaël Gascoin de m'inviter régulièrement à boire le café et à me restaurer dans son bureau.

Je remercie Nathalie Van Acker de me supporter dans le bureau, mais aussi pour son tempérament, la ferveur qu'elle met dans son travail, les gourmandises qu'elle apporte et les échanges scientifiques que nous avons eu et auront d'autant plus par la suite je l'espère.

Je remercie tout particulièrement Frédéric Escudié pour l'ensemble des réflexions et développements scientifiques et techniques que nous avons partagés. Je le remercie surtout pour l'aide considérable qu'il a la gentillesse de m'accorder quotidiennement sur tous les problèmes et questionnements de tout ordre. Je le remercie enfin pour la qualité de ses références cinématographiques, notamment celles se rapportant à la science fiction et à la fantaisie.

Je remercie ensuite Charlotte Syrykh pour le temps considérable que tu m'as accordé sur certains des développements majeurs de ce travail, ainsi que pour ta bonne humeur dans l'ensemble des échanges qu'on a pu avoir et ce quelles que soit les circonstances.

Je remercie finalement l'ensemble du laboratoire d'anapath de l'Institut Universitaire du Cancer de Toulouse pour les efforts qu'ils ont fait pour m'intégrer dans le laboratoire.

Un grand merci à l'ensemble des membres du laboratoire ICube qui ont su m'accueillir lors de mes passages à Strasbourg. J'adresse un remerciement tout particulier à Jennifer pour les efforts qu'elle a déployés pour m'intégrer au sein du laboratoire ainsi que pour ses conversations et ses passions pour la course, la montagne, l'aventure qu'elle sait largement faire partager.

Je remercie ma famille et mes parents qui ont toujours su être présents pour me soutenir et ont grandement facilité l'ensemble de ma scolarité. Je remercie notamment Réjane pour avoir limiter les erreurs d'orthographe dans ce manuscrit.

Je remercie également Clémence Brival pour les super escapades aux Royaume Uni, les concerts, mais surtout pour sa joie de vivre, sa gentillesse et sa grande sincérité qui m'ont fait oublier bien des malheurs ces derniers temps.

Je remercie bien évidemment l'ensemble des membres de la Compagnie du Bátar avec qui j'ai construit un bateau magnifique et suis parti vivre une expédition fantastique.

Je remercie l'ensemble des « Bitch Boys », pour leur accueil durant mes séjours à Strasbourg et pour les moments importants que l'on partage encore depuis nos études à Telecom Physique Strasbourg.

Table des matières

Liste des figures

Liste des tableaux

Introduction générale

La place de l'informatique dans le domaine biomédical

Il est des outils informatiques sans lesquels certains examens médicaux actuels ne pourraient exister. Les analyses génétiques pour lesquelles un séquençage de l'*ADN* est nécessaire par exemple, requièrent bien évidemment un support numérique pour stocker des bases lues par le séquenceur, mais aussi une capacité de calcul et un panel d'algorithmes indispensables à l'identification des gènes, de leurs versions ainsi que de leurs mutations.

En imagerie médicale, les ressources informatiques sont rapidement devenues nécessaires. Les *CT-scanners* ou les *IRMs* ne peuvent former que des images numériques. Ils posent pour cela des problèmes inverses que seuls des ordinateurs sont en mesure de résoudre. Nouveaux standards de la discipline, ces systèmes d'imagerie, accompagnés de leurs infrastructures informatiques, se sont imposés parce qu'ils améliorent considérablement la prise en charge des patients.

En anatomie et cytologie pathologique, discipline médicale dont un aspect important repose sur l'imagerie de microscopie, l'image numérique trouve plus difficilement sa place. L'impact des technologies de numérisation de lames histologiques sur la prise en charge des patients est plus indirect, et le domaine attend notamment beaucoup de l'informatique sur le plan de l'analyse automatisée de ces images.

L'analyse automatique des images de microscopie poursuit deux objectifs. Le premier est la substitution complète au pathologiste sur des tâches à fort degré de pénibilité, telles que le dénombrement de certaines entités comme les noyaux, les mitoses ou toute autre structure présentant un intérêt pour le diagnostic ou le pronostic du patient. Plus généralement, cette conception place la solution informatique comme un moyen d'évolution de la pratique quotidienne de la pathologie.

Un second versant de l'analyse algorithmique de ces images, plus noble, mais infiniment plus ambitieux, projette de faire évoluer la science médicale. « La médecine, en tant que science, écrit Claude Bernard dans son *Introduction à la médecine expérimentale*, a nécessairement des lois qui sont précises et déterminées, qui, comme celles de toutes les sciences, dérivent du critère expérimental. [...] ma pensée est simplement d'appliquer à la médecine les principes de la méthode expérimentale, afin qu'au lieu de rester science conjecturale fondée sur la statistique, elle puisse devenir une science exacte fondée sur le déterminisme expérimental. »

À cette fin, l'informatique se place en « auxiliaire », pour reprendre une nouvelle fois les mots de Claude Bernard, de la médecine en alimentant les réflexions, hypothèses et démonstrations médicales d'arguments quantitatifs et impartiaux. L'informatique en analyse automatique des images de la pathologie, en un sens, se porte garante d'une évaluation objective des hypothèses formulées par les médecins. Elle a vocation à faire obstacle aux erreurs de démarche ainsi qu'au tests biaisés ou délibérément falsifiés.

Analyse des images et apprentissage profond

Parallèlement aux attentes de la pathologie numérique, le domaine informatique de l'analyse des images connait actuellement un essor et une notoriété sans précédent. L'apprentissage automatique, ainsi que l'ensemble des disciplines que l'on regroupe, peut-être un peu trop facilement, sous l'expression « intelligence artificielle », ont relevé avec succès bon nombre des défis majeurs du traitement et de l'analyse des images formulés au cours des dernières décénnies. Ces outils autorisent aujourd'hui des interprétations d'images d'une complexité inespérée et surpassent désormais l'humain sur certaines analyses.

À l'inverse, ce sont ici des travaux en biologie sur l'étude du fonctionnement du cortex visuel qui ont provoqué une révolution dans les pratiques informatiques. Les systèmes de l'« intelligence artificielle » actuelle reposent principalement sur la modélisation des mécanismes de la perception chez l'animal. Ces modèles, formellement instanciés dans ce que l'on appelle des réseaux neuronaux convolutifs profonds (*Convolutional Neural Networks, CNNs*), apprennent par induction, c'est-à-dire sur la base de nombreux exemples, à produire des décisions arbitrairement complexes à partir d'une image.

L'extension de ces techniques d'analyse aux images de la pathologie pour prédire des diagnostics, des pronostics ou la réponse à des thérapies ne se fait pas trivialement. Au-delà des difficultés techniques liées à la structure, la taille ou la variabilité des données, auxquelles les modèles profonds d'apprentissage automatique sont extrêmement sensibles, la médecine est un domaine critique où les bonnes performances de prédiction d'un système n'assurent pas son implémentation dans une application. L'applicatif exige en effet certaines garanties sur la généralisabilité du dispositif, ainsi que sur l'interprétabilité et l'intelligibilité des décisions prises.

Les exigences de la médecine à l'égard des modèles prédictifs font encore largement écho aux écrits de Claude Bernard et à sa véhémence à l'égard des approches statistiques. La représentation par des valeurs moyennes est propre à l'étude des tendances sur de grandes populations mais n'est d'aucun recours au médecin qui soigne des cas particuliers. En plus d'exposer cette base de ce que l'on appelle aujourd'hui la *médecine personnalisée*, il ne manque pas de faire remarquer qu'une science, et donc une connaissance, ne saurait se fonder uniquement sur des statistiques. Ainsi, un modèle prédictif, obtenu par apprentissage statistique (induction) tire sa connaissance de relations de corrélations qui échouent à vérifier les liens de causalité, dans un sens ou dans l'autre, entre les motifs observés dans l'image et les prédictions ou décisions qui en résultent.

Analyse des images et représentations hiérarchiques

Les propriétés marquantes lorsque l'on s'attache aux connaissances *a priori* du pathologiste, autrement dit à ce qu'il s'attend à trouver dans une image de son domaine avant qu'elle ne se présente, sont l'étendue du vocabulaire qu'il utilise ainsi que sa structuration logique. Indépendamment de l'image donc, les concepts de l'expert sont liés les uns aux autres par divers relations d'ordre, telles que l'*implication*, qui induisent des structures hiérarchiques entre les concepts. Bien connues du domaine de l'*ingénierie des connaissances*, l'établissement et l'écriture formelle de ces hiérarchies permettent à un ordinateur de raisonner automatiquement dans l'espace des connaissances humaines.

Sur présentation d'une image, ce sont les relations de contenance et de voisinage entre les régions de l'image qui sont les manifestations des liens logiques entre les concepts de l'expert. Les procédures de *segmentation hiérarchique* exploitent ces relations afin d'assembler des objets *élémentaires* repérés dans l'image pour constituer des structures plus complexes, disons sémantiquement supérieures, dans le langage de l'expert.

L'exploitation du lien logique pour déduire automatiquement de la présence d'un concept dans une image, ou tout autre type de donnée, se fait appeler *reconnaissance syntaxique* des objets. Elle fait partie du domaine plus général de l'« intelligence artificielle symbolique » qui est volontairement distinguée de l'appellation « intelligence artificielle » que l'on réserve souvent, improprement cependant, aux méthodes de l'apprentissage statistique.

C'est sur cette base de structures hiérarchiques formée par des liens logiques que se fonde le concept même d'interprétabilité d'une décision. Lorsque l'analyse est une succession de décisions logiques, autrement dit que toute décision peut être justifiée par la réalisation simultanée de plusieurs évènements qui sont une condition suffisante à sa prédiction, alors seulement, l'analyse est jugée interprétable. L'interprétabilité d'un système décisionnel équivaut à sa capacité à user de la logique pour parvenir à sa conclusion et prôner l'un sans revendiquer l'autre est une position difficile à défendre.

Objectifs

Autour de ces réflexions sur le rôle de l'apprentissage automatique par réseaux neuronaux convolutifs profonds, celui de la structuration des informations extraites dans l'image, mais aussi des informations connues *a priori* par les experts du domaine, dans l'analyse automatique des images de la pathologie numérique, ces travaux de thèse poursuivent plusieurs objectifs :

- ♢ identifier, au contact quotidien des pathologistes, les problématiques cliniques pertinentes de la discipline pouvant être formulées comme des problèmes informatiques ;
- ♢ établir des stratégies algorithmiques, dans l'état de l'art de l'analyse automatique des images, pour répondre à ces problématiques en mettant l'accent sur le potentiel des réseaux neuronaux convolutifs profonds ;
- ♢ identifier les obstacles principaux à la systématisation des réseaux neuronaux profonds pour répondre aux attentes biomédicales ;
- ♢ évaluer le potentiel des représentations hiérarchiques des images, mais aussi des connaissances de l'expert (connaissances structurées), pour surmonter ces obstacles ou compléter une analyse entamée par une solution d'apprentissage profond.

Chapitre 1

Objectifs et terminologie de l'analyse

« What is real ? How do you define 'real' ? If you're talking about what you can feel, what you can smell, what you can taste and see, then 'real' is simply electrical signals interpreted by your brain. »

Morpheus

Sommaire

1.1 Introduction

L'analyse des images numériques est un terme extrêmement général qui regroupe une très grande quantité de tâches d'extraction de données dans les images. Naturellement et le plus généralement accomplies par des humains, certaines d'entre elles peuvent néanmoins être formulées avec rigueur comme des problèmes mathématiques pour lesquels une solution, exacte ou approchée, peut être proposée sous la forme d'un algorithme. Inversement, un algorithme proposé n'est parfois réponse à aucune formulation de problème claire, mais satisfait plutôt qualitativement à un certain nombre de « bonnes propriétés » qui garantissent l'extraction d'informations pertinentes. L'implémentation desdits algorithmes et leur exécution par des machines peuvent permettre, selon le cas, d'extraire une information quantitative inaccessible à la perception humaine ou de se substituer à l'humain lorsqu'une tâche s'avère trop fastidieuse.

Lors de la programmation d'un outil d'analyse automatique, le développeur se heurte à une grande complexité dans la formulation mathématique du problème d'extraction de données. Cette difficulté est principalement imputable à deux facteurs. La structure de l'image, d'une part, représente le continuum de positions et de signaux lumineux de notre monde en un nombre fini de pixels et une échelle discrète d'intensités lumineuses. La décomposition mathématique formelle des connaissances humaines, ou plutôt de leur manifestation visuelle, en motifs décrits par des pixels est particulièrement ardue. D'autre part, la méconnaissance du mécanisme de la perception humaine dans sa globalité, rend bien souvent impossible l'extraction de données par imitation du cheminement de l'expert.

L'objectif de ce chapitre est d'introduire les concepts fondamentaux de l'extraction de connaissances dans les images. Les formalismes, algorithmes et structures de données classiques des différentes approches seront abordées. Une attention particulière sera accordée aux relations d'ordre établies entre les résultats d'une analyse, ainsi que sur les structures hiérarchiques qu'elles induisent. L'accent sera notamment mis sur les types de raisonnements illustrés par ces structures. Nous étudierons d'abord les mécanismes de délinéation d'objets (Section 1.2), puis nous présenterons les stratégies de description des images et de leurs segments (Section 1.3). Enfin, dans une dernière partie, nous aborderons les problématiques de classement et de regroupement des images (Section 1.4) avant de conclure ce chapitre (Section 1.5).

1.2 Images et segmentation

Dans le langage naturel humain, celui des experts du domaine applicatif d'une image, il est rare qu'une information pertinente puisse-être extraite du signal d'un pixel isolé. L'expert s'affranchit des pixels et les regroupe inconsciemment en composantes connexes de l'image qu'il confond instantanément avec des objets usuels de son domaine. La tâche de *segmentation* automatique d'une image est la reproduction de ce mécanisme de groupement et prend la forme d'une procédure de partitionnement de l'image. La segmentation d'une image détermine les confins d'objets spécifiques dont les propriétés visuelles sont plus ou moins clairement formalisées. Il existe un certain nombre d'algorithmes de segmentation qui s'attaquent à des problèmes de délinéation posés de manières radicalement différentes selon le degré de précision de délinéation attendu et la difficulté de formalisation du problème. Loin de dresser un historique ou état de l'art exhaustif de ces méthodes, cette section propose plutôt d'établir les concepts-clefs de la segmentation et de regrouper les approches en grandes catégories.

Une première distinction peut être réalisée sur le degré de contrainte sémantique injecté dans la formulation du problème. Les segmentations peuvent ensuite différer par la structure de leur résultat : un segment de partition peut à son tour être segmenté et la nature récursive de cette procédure peut aboutir à des segmentations dites « hiérarchiques ». Enfin, une distinction s'opère également sur les informations ou caractéristiques de l'image intégrées par

les différentes méthodes pour établir leur partition telles que les textures ou les formes des différents segments. Cette question de l'extraction de caractéristiques n'est cependant pas traitée ici et fait volontairement l'objet d'une section indépendante.

Notations

- Soit $X : \Omega \subseteq \mathbb{Z}^2 \to \mathbb{Z}$ une image en niveaux de gris ;
- On appellera pixel, tout élément $p \in \Omega$ et on notera $X(p)$ sa valeur de luminance ;
- On notera $\boldsymbol{C} = \{c_1, \ldots, c_n\}$ l'ensemble des *classes* que l'expert souhaite délinéer dans les images ;
- On appellera *partition* ou *segmentation* d'une image tout ensemble de régions $\boldsymbol{P} = \{s_1, \ldots, s_n\} \subseteq \mathcal{P}(\Omega)$ tel que $\forall i \neq j, s_i \cap s_j = \emptyset$ et $\bigcup_i s_i = \Omega$;
- On appellera indistinctement *région*, *segment* ou *superpixel* de l'image I tout ensemble connecté de superpixels $s_i \in \boldsymbol{P}$.

1.2.1 Contrainte sémantique faible

Il arrive que certaines applications ne requièrent pas une interprétation sémantique de haut niveau de l'image. Pour ce genre d'analyses, des détections de formes géométriques et de couleurs peuvent s'avérer suffisantes pour accomplir une tâche générale, on parle de *contrainte sémantique faible*. Ces méthodes de partitionnement diffèrent principalement par la position du problème d'optimisation auquel répond la segmentation. Les formulations *locales* fixent la « qualité » d'un segment quelconque de la partition, tandis que les énoncés *globaux* sont des problèmes d'optimisation dans l'espace $\mathcal{P}(\Omega)$ des partitions de l'image.

L'approche locale Les critères locaux de segmentation s'appuient sur la notion d'homogénéité pour mesurer la qualité d'une région quelconque s de la partition. La mesure d'homogénéité la plus commune relève la dispersion σ des valeurs de luminance d'un segment. Les méthodes *absolues* fixent un seuil t de dispersion acceptable sur les segments et visent à produire des partitions $\boldsymbol{P}$ respectueuses du *prédicat d'homogénéité* : $\forall s \in \boldsymbol{P},\ \sigma(s) < t$.

A contrario l'approche dite *relative* étudie un segment s dans son environnement et considère un critère plus objectif de maximisation de contraste, c'est-à-dire de comparaison entre la distribution de luminance *intra-segment* $\boldsymbol{E}_{int}(s) = X(s)$ et une distance *inter-segments* minimale avec les distributions des segments voisins $\boldsymbol{E}_{ext}(s) = \min_{v \in V} \{\, d(X(s), X(v)) \,\}$. Un segment est une région *connexe et contrastée* de l'image encore appelée *cocon* si, et seulement si, elle respecte la relation suivante :

$$\boldsymbol{E}_{int}(s) < \boldsymbol{E}_{ext}(s) \tag{1.1}$$

L'approche globale Une énergie globale de segmentation est une fonction à valeurs réelles définie sur l'espace des partitions de l'image $\boldsymbol{E} : \mathcal{P}(\Omega) \to \mathbb{R}$ et susceptible d'évaluer la qualité d'une partition. La formulation de $\boldsymbol{E}$ est un fondement du paradigme *variationnel* du traitement et de l'analyse des images notamment popularisé par les travaux de Mumford and Shah [1989]. Les principes de ces méthodes sont très naturellement expliqués dans le langage général des problèmes inverses qui propose de voir la segmentation comme un *modèle* simplifié de l'image dans lequel les objets sont bien individualisés, ont des contours réguliers et des textures homogènes. L'image, dans ce cas, est une *observation* du modèle simplifié, que des processus fictifs ou physiques liés à l'acquisition de l'image ont rendu imparfaite. L'énergie d'une partition dans ce cas est toujours exprimée comme un compromis entre deux termes :

$$\boldsymbol{E}(\boldsymbol{P}) = D(X, \boldsymbol{P}) + C(\boldsymbol{P}) \tag{1.2}$$

Le terme d'*attache aux données* $D(X, \boldsymbol{P})$ tout d'abord, rend compte de l'explication des luminances dans l'image X par le modèle $\boldsymbol{P}$ et stipule que les caractéristiques des segments formés dans la partition $\boldsymbol{P}$ sont plausibles compte tenu de leur version « déformée » observée dans l'image X. Ensuite, le terme dit de *régularisation* $C(\boldsymbol{P})$ pénalise les partitions trop complexes ou invraisemblables sur la base des connaissances *a priori* sur les scènes abstraites recherchées.

Exploration de l'espace des partitions et optimisation Quelle que soit la fonction de coût formulée pour l'évaluation d'une segmentation, l'optimisation du critère est toujours un problème particulièrement gourmand en temps de calcul. En effet, l'espace des partitions de l'image est de dimension trop grande pour être exploré entièrement et la plupart des méthodes proposent d'explorer un sous-ensemble de partitions autour d'une solution initiale. L'exploration consiste à appliquer des opérateurs de déformation qui conservent la propriété de partition, tels que les fusions et séparations de régions. Une logique de descente de gradient peut alors être mise en place par applications successives des opérateurs qui minimisent le plus la fonction de segmentation. Ces stratégies d'optimisation gloutonnes ne peuvent cependant garantir qu'un minimum local, atteint lorsque aucune fusion ou découpe ne peut plus diminuer la valeur de $\boldsymbol{E}$.

1.2.2 Contrainte sémantique forte

Lorsque le problème de segmentation prend la forme d'une tâche de classement de pixels, il est question de *segmentation sémantique*. Le terme regroupe les méthodes dites *supervisées*, puisque les *classes* $\{c_1, \ldots, c_n\}$ attribuables aux pixels de l'image sont empruntées au langage de l'expert et sont connues *a priori* par l'algorithme de segmentation. Les classes sont choisies pour leur pertinence à très court terme dans le processus global d'analyse et atteignent, de fait, un haut niveau d'abstraction et de complexité sémantique. Selon cette formulation, la notion de partition de l'image n'est plus si évidente et il semble utile de préciser qu'un segment de l'image dans ce cas est défini comme un ensemble de pixels connectés et de même classe ; on notera par ailleurs $Y : \Omega \subseteq \mathbb{Z}^2 \to \boldsymbol{C}$ la carte des *classes* associée à l'image X.

Lorsque les modalités d'imagerie s'y prêtent, c'est-à-dire lorsqu'une relation quantitative existe entre la valeur de luminance et l'intensité ou, plus simplement, la présence du phénomène à observer, les règles de segmentation de l'image peuvent être énoncées simplement par le biais d'opérations de seuillage ou de fenêtrage de l'histogramme des niveaux de gris comme on pouvait par exemple le lire dans les travaux précurseurs de Ohlander et al. [1978]. Il n'est pas si rare qu'une telle hypothèse simplificatrice puisse-être émise, notamment dans des domaines très spécifiques, comme l'imagerie médicale ou spatiale, pour lesquels les imageurs, microscopes à fluorescence ou imageurs multi/hyperspectraux, ainsi que les prétraitements physico-chimiques, marquages immunohistochimiques et immunohistochimies de fluorescence, sont bien souvent dédiés à la distinction radiométrique ou colorimétrique des objets dans l'image.

Dans une très grande majorité de cas cependant, la segmentation colorimétrique ou par fenêtrage d'histogramme n'est pas applicable puisqu'à une classe recherchée peuvent correspondre une trop grande variété de valeurs de luminance, de la même façon que le niveau de gris d'un pixel peut être une manifestation plausible pour plusieurs classes. La relation entre le signal d'un pixel x_i et sa classe associée y_i n'est donc pas déterministe et il est courant de modéliser les images et cartes de classes comme des champs aléatoires. La segmentation s'énonce alors comme un problème de recherche de maximum *a posteriori MAP* :

$$Y^* = \arg\max_{Y}(\mathbb{P}(Y|X)) \tag{1.3}$$

Cette formulation par champs aléatoires, en plus de tenir compte des distributions de classes plausibles y_i pour des valeurs de luminance d'un pixel isolé x_i, permet d'injecter des informations de contexte en modélisant statistiquement les corrélations spatiales entre le signal ou la classe d'un pixel et celui des pixels avoisinants. Les méthodes de segmentation sémantique proposent une forme paramétrique de la loi postérieure de l'Equation 1.3 et tirent parti de la loi des grands nombres pour approximer la valeur de leurs paramètres en s'appuyant sur de nombreuses *réalisations* de la loi de $Y|X$, on parle ici d'*apprentissage statistique*. Lorsque les réalisations utilisées pour estimer $\mathbb{P}(Y|X)$ prennent la forme d'un ensemble de données préalablement annotées par des experts, c'est-à-dire un ensemble de paires image-segmentation (X, Y) où Y est une segmentation produite manuellement ou semi-automatiquement, il est plus spécifiquement question d'*apprentissage supervisé*. Parmi les méthodes de segmentation sémantique basées sur un apprentissage supervisé, deux sous-types se distinguent encore par leur modélisation de la distribution postérieure des classes d'une image.

L'approche générative Les méthodes *génératives* factorisent l'Equation 1.3 par application de la règle de Bayes, ce qui revient à maximiser le produit $\mathbb{P}(Y)\mathbb{P}(X|Y)$. Le terme *génératif* $\mathbb{P}(X|Y)$, aussi appelé vraisemblance, modélise les caractéristiques du processus d'acquisition, et la loi $\mathbb{P}(Y)$ reflète une connaissance *a priori* sur les textures envisageables dans l'image. Ces techniques doivent leur essor aux travaux de Geman and Geman [1984] et à la modélisation de Y par des champs de Markov qui encodent, pour des voisinages de pixels, les dépendances spatiales entre les classes.

L'approche discriminante Les méthodes *discriminantes* proposent une paramétrisation directe de la distribution postérieure des classes de l'image et considèrent donc l'Equation 1.3 sans passage par la décomposition de Bayes. C'est le cas notamment des réseaux entièrement convolutifs utilisés pour la segmentation comme le réseau *U-net* de Ronneberger et al. [2015], ainsi que des modèles graphiques bien plus généraux tels que les champs aléatoires conditionnels décrits par exemple par He et al. [2004]. Ces techniques sont souvent privilégiées par rapport aux méthodes génératives pour leur meilleur taux de réussite dans la prédiction des classes de pixels. En effet, ces approches se focalisent uniquement sur les motifs permettant la distinction des classes et évitent l'accumulation des erreurs qu'apportent généralement les modélisations multiples et leurs combinaisons. En contrepartie, l'exploitation abusive des motifs différenciants dans ces méthodes est propice à l'apprentissage de biais dans les données aussi appelé *sur-apprentissage*. Les méthodes génératives sont ainsi réputées avoir un meilleur pouvoir de généralisation et prouvent leur intérêt dans les applications d'apprentissage sur de faibles jeux de données comme le font par exemple Rezende et al. [2016].

1.2.3 Contrainte d'individualisation

La segmentation d'individus ou *segmentation d'instances* impose une contrainte supérieure à la sémantique forte en ajoutant l'identification des individus d'une même classe au problème de partitionnement. Sur l'exemple des images de microscopie, la délinéation des noyaux de cellules illustre parfaitement la problématique : les noyaux visibles dans une image appartiennent à la même classe sémantique "noyau", mais plusieurs noyaux distincts ne peuvent être représentés par un même segment, on qualifierait sinon la partition de l'image de *sous-segmentation* et réciproquement, le noyau d'une cellule unique ne saurait être *sur-segmenté*, c'est-à-dire découpé en de multiples segments. Il est entendu qu'un tel problème n'échappe à la segmentation sémantique que dans le cas d'objets contigus ou chevauchant de même classe, puisque pour des objets isolés spatialement, une segmentation sémantique aboutirait naturellement à des segments disjoints.

Dans la littérature, les problèmes de segmentation d'instances sont formulés et résolus de façons variées, mais un trait commun à toutes ces techniques est l'intégration de connaissances *a priori* complexes sur les individus que l'on souhaite détecter, on parle de *segmentations guidées par les connaissances*. Parmi les outils les plus exploités, Vese and Chan [2002] proposent par exemple une approche basée sur les lignes de niveaux, et Caselles et al. [1997] utilisent des modèles de contours actifs qui vont même jusqu'à appliquer des contraintes physiques sur la forme et la taille des objets à segmenter. Lorsque nombre d'objets d'une même catégorie peuvent être observés dans une même image, la concentration des individus et leur répartition spatiale peuvent également entrer dans la liste des paramètres connus *a priori*. Des procédures dérivées du *template matching* de Cooper [1989], telles que les processus ponctuels marqués, proposent notamment de modéliser à la fois les distributions spatiales, mais aussi morphologiques et texturelles des objets d'intérêt. Une description détaillée de ces procédures se trouvent dans les travaux de Descombes and Zerubia [2002].

Toutefois, l'intégration des informations *a priori* sur les objets recherchés alourdit considérablement la complexité algorithmique des problèmes d'optimisation posés ; si bien que la plupart de ces approches disparaissent peu à peu de la littérature au profit de stratégies sémantiques qui intègrent implicitement par apprentissage statistique, *id est* sans qu'elles soient définies ou implémentées manuellement, autant sinon plus d'information sur la répartition spatiale, les textures ou la forme des objets recherchés.

1.2.4 Segmentation et structures hiérarchiques

Bien qu'une majorité de méthodes propose une partition *plate* des images, c'est-à-dire un découpage en segments mutuellement exclusifs, il n'est pas rare que des relations d'inclusion ou d'intersection entre les objets imagés soient porteuses d'une information précieuse pour l'interprétation. Dès lors, une seule segmentation ne permet vraisemblablement pas d'extraire des objets d'intérêt définis ou caractérisés par le fait qu'ils « contiennent » ou se trouvent « à l'intérieur » d'un autre segment de l'image. La résolution automatique d'une telle tâche impose donc l'usage de partitions multiples et il est notamment question de *granularité* de segmentation lorsque les partitions sont susceptibles de mettre en évidence des objets de tailles différentes.

Néanmoins, les multiplicités de segmentations et de « grains » ne sont toujours pas des conditions suffisantes pour analyser les rapports de contenance entre des segments. Très intuitivement, si l'on considère deux partitions $\boldsymbol{P}_i$ et $\boldsymbol{P}_j$ d'une même image et telles que $\boldsymbol{P}_j$ est objectivement de granularité plus fine que $\boldsymbol{P}_i$, autrement dit que les segments de $\boldsymbol{P}_j$ sont plus petits que ceux de $\boldsymbol{P}_i$, les relations entre les segments de l'une et de l'autre ne sont pour autant synonymes de contenance que si les régions de $\boldsymbol{P}_j$ sont obtenues par re-découpage de certaines, sinon de la totalité, des régions de $\boldsymbol{P}_i$ ou inversement si la totalité des régions de $\boldsymbol{P}_i$ peuvent être obtenues par fusion de segments connexes dans $\boldsymbol{P}_j$. Plus rigoureusement, on dira qu'une paire de segmentations $(\boldsymbol{P}_i, \boldsymbol{P}_j)$ d'une image X est propice à l'étude des relations de contenance entre les objets dans X si, et seulement si, $\boldsymbol{P}_j$ est un *raffinement* de $\boldsymbol{P}_i$ et on notera $\boldsymbol{P}_j \leq \boldsymbol{P}_i$, où la relation d'ordre $\leq$ dite de *raffinement* est définie sur l'espace des segmentations à partir de l'ordre naturel d'inclusion sur les parties de Ω :

$$\boldsymbol{P}_j \leq \boldsymbol{P}_i \qquad \Leftrightarrow \qquad \forall s_j \in \boldsymbol{P}_j,\ \exists s_i \in \boldsymbol{P}_i \mid s_j \subseteq s_i \tag{1.4}$$

Par ailleurs, une *segmentation hiérarchique* d'une image est définie comme un ensemble ordonné de partitions :

$$\boldsymbol{H} = \{\boldsymbol{P}_0, \ldots, \boldsymbol{P}_N\} \quad \mid \quad \forall n < N,\ \boldsymbol{P}_n \leq \boldsymbol{P}_{n+1}$$

Une telle hiérarchie est naturellement construite par appels récursifs d'une procédure de segmentation f selon une stratégie dite *descendante* si l'algorithme de segmentation re-découpe des segments en régions plus fines, $\boldsymbol{P}_n = f(\boldsymbol{P}_{n+1})$, ou *ascendante* si, au contraire, la procédure regroupe des segments en structures plus grossières, $\boldsymbol{P}_{n+1} = f(\boldsymbol{P}_n)$. Nous assimilerons une segmentation hiérarchique à la structure arborescente, aussi appelée *arbre de segmentation* et notée $\boldsymbol{T} = (\boldsymbol{N}, \Uparrow, \Downarrow)$. L'ensemble $\boldsymbol{N}$ des *nœuds* de l'arbre est défini par regroupement de tous les segments de toutes les partitions de la hiérarchie, $\boldsymbol{N} = \bigcup_n \boldsymbol{P}_n$ et les liens de parenté sont établis naturellement par relation d'inclusion spatiale stricte entre les segments. Ainsi, le *parent* $\Uparrow(s)$ de tout *noeud* $s \in \boldsymbol{N}$, à l'exception du segment-racine couvrant l'image entière, est défini de manière unique comme le plus petit élément de $\boldsymbol{N}$ contenant strictement s. En s'appuyant sur cette définition, tout segment s_i tel que $\Uparrow(s_i) = s$ sera appelé *enfant* de s et on notera par la suite $\Downarrow(s)$ l'ensemble des enfants du nœud s.

Stratégies de construction Bien que certains algorithmes, à l'instar de Poggi and Ragozini [1999], emploient des stratégies descendantes pour établir l'arbre de segmentation d'une image, une très grande majorité des méthodes suivent une procédure de construction ascendante. Les stratégies les plus communes du domaine étant les approches de Koepfler et al. [1994], Ballester et al. [1996] et Fuchs [2001]. De manière intuitive, la fusion de deux segments d'une partition en une région plus « grossière » constitue une perte d'information puisqu'elle fait disparaître une frontière, tandis que la division d'une région implique nécessairement de créer de l'information en formant une séparation entre plusieurs sous-structures et conduit naturellement à un problème de complexité très supérieure.

Retour à une segmentation plate La structure hiérarchique de la partition n'est bien souvent qu'une conséquence de la procédure de segmentation, notamment dans le cas où la méthode procède par fusions successives de régions de la partition. Malgré tout, l'algorithme peut volontairement faire le choix d'ignorer la structure hiérarchique sous-jacente et produire une segmentation plate de l'image en choisissant un critère d'arrêt sur la minimisation d'une énergie de segmentation par exemple. Cependant, en ne considérant que la partition résultant d'une fusion à un instant donné, les segments d'intérêt formés dans des étapes antérieures ou qui pourraient être formés par des fusions à venir échappent totalement à la méthode. Il est alors évident que le stockage de la structure arborescente complète suivi d'une étape distincte de coupure offre une gamme bien plus large de segmentations. En effet, les coupures ont un degré de liberté supplémentaire : elles deviennent « multi-niveaux » et sont dès lors capables d'adapter localement la granularité de la segmentation à la difficulté de la tâche. Il est donc possible d'aboutir à des partitions de bien meilleure qualité vis-à-vis de la minimisation de l'énergie de segmentation. C'est notamment l'objet des recherches de Guigues [2003] qui sont très complètes sur le sujet.

De manière plus formelle, si $\boldsymbol{T} = (\boldsymbol{N}, \Uparrow, \Downarrow)$ est une segmentation hiérarchique de X, on appellera coupe de l'arbre $\boldsymbol{T}$ toute partition de X dont les segments sont des éléments de $\boldsymbol{N}$. En reprenant la notation utilisée par Guigues [2003], on notera par la suite $\mathbb{C}ut(\boldsymbol{T})$ l'ensemble de ces coupes.

1.2.5 Bilan

L'ensemble des méthodes évoquées ci-dessus témoigne, par la variété des paradigmes, des formulations du problème et des outils de résolution déployés, de l'intérêt que suscite la tâche de segmentation dans la communauté de l'analyse et du traitement automatique des images depuis près d'une cinquantaine d'années. Cependant, la simplicité de l'articulation de cette section, tout particulièrement autour de la contrainte sémantique, se veut un reflet de la maigre

marge de manœuvre actuellement laissée aux développeurs des solutions de segmentation. Auparavant, ce sont les contraintes du domaine d'application, qu'elles portent sur la vitesse d'exécution de l'algorithme, sa performance de détection, sa qualité de délinéation et surtout, la quantité de données annotées disponible pour évaluer et/ou entraîner un modèle supervisé, qui fixaient le choix d'une technique de segmentation. Désormais, l'apprentissage supervisé de modèles discriminants fixe si haut l'état de l'art en la matière, que les techniques alternatives ne sont plus vraiment envisagées. Ainsi, l'effort d'adaptation et de mise en place d'un outil de segmentation automatique repose moins sur l'ingénierie des solutions algorithmiques que sur la capacité des « clients », experts du domaine d'application, à fournir des données annotées en quantité suffisante pour permettre un apprentissage statistique.

Dans cette section, nous n'avons évoqué que vaguement la notion de description ou caractérisation des segments de l'image. Les termes « texture » et « forme », utilisés ci-dessus, se traduisent par des valeurs numériques. Les théories qui étayent ces mécanismes de traduction définissent des concepts fondamentaux indispensables à la compréhension des outils modernes de l'analyse automatique des images, et qu'il convient d'aborder à présent.

1.3 Extraction de caractéristiques

1.3.1 Domaine des connaissances humaines

Une hiérarchie sémantique Dans leurs travaux sur l'apprentissage automatique des représentations, Bengio et al. [2013] affirment qu'une décision ne peut être prise sur la base d'une image que si la machine est capable d'en « identifier et d'en dissocier les paramètres explicatifs sous-jacents cachés au milieu des données sensorielles de bas-niveau ». L'expression « bas-niveau » utilisée ici fait directement référence à une relation d'ordre, et donc à une structure hiérarchique, non pas sur la complexité d'extraction de connaissances par des moyens algorithmiques, qui n'en est qu'une conséquence, mais bien dans l'espace sémantique, celui des concepts que l'humain manipule. Le domaine de l'ingénierie des connaissances étudie particulièrement cette *structuration* et les travaux de Gruber [1995] décrivent toujours une forme d'arbre conceptuel, appelé *ontologie*, dans lequel de nombreuses relations de parenté entre les concepts, telles que la *subsomption* ou la *partinomie* peuvent être définies. Declerck and Charlet [2011] exploite par exemple cette *structuration* du savoir humain pour développer des systèmes-experts pour la prise de décision automatique.

En reprenant les notations de la section précédente, nous noterons plus particulièrement $\boldsymbol{C}_h$ l'ensemble des « concepts humains » manipulés par l'expert et en prenant l'exemple de la partinomie, nous écrirons volontiers $c_i \ \mathcal{P} \ c_j$ pour signifier que le concept c_i contribue à la définition de c_j, c'est-à-dire que c_i est nécessaire, mais souvent insuffisant, pour définir c_j. De manière analogue, $c_i \ \mathcal{S} \ c_j$ désignera la subsomption de c_j sur c_i et on empruntera parfois le vocabulaire de la programmation orientée objets pour désigner c_j comme une *classe-mère* de c_i. Il n'est pas toujours possible d'envisager une structure arborescente en considérant ces relations. En effet, le cas de l'*héritage multiple* pour la subsomption ou de la *partinomie multiple* lorqu'un concept peut entrer dans la définition de deux concepts distincts, n'interdisent pas la construction de graphes cycliques. La définition d'une ontologie garantit toutefois la propriété d'arborescence pour une relation donnée, et nous adopterons naturellement dans ce cas la notation définie dans le cadre de la segmentation hiérarchique : $\boldsymbol{T}_h = (\boldsymbol{C}_h, \Uparrow, \Downarrow)$.

Justification et raisonnement Lorsque l'expert doit justifier formellement de la présence d'un concept $c \in \boldsymbol{C}_h$ dans une image, il doit opérer par déduction, c'est-à-dire réunir l'ensemble des conditions *nécessaires* à l'identification de c de manière à compiler une condition *suffisante*. Il explore donc les liens *causaux* entre les concepts à travers la relation de partinomie de manière

à prédire en premier lieu l'ensemble $\Downarrow(c)$ des concepts constitutifs de c. Lorsque l'établissement d'un concept nécessaire $c_i \in \Downarrow(c)$ devient à son tour discutable, l'expert peut avoir recours à un degré d'explication plus fin en explorant les enfants de c_i, $\Downarrow(c_i) \subset \Downarrow^2(c)$. Récursivement, la justification relève, au pire, de l'exploration exhaustive de tous les descendants du concept c. Par la suite, on nommera *justifications partielles* du concept c, et on notera $\Downarrow^\infty(c)$, l'ensemble des éléments du langage de l'expert qui peuvent participer à la définition de c :

$$\Downarrow^\infty(c) = \{c_i \in \boldsymbol{C}_h \mid c_i < c\} \tag{1.5}$$

Dans le langage des représentations hiérarchiques, il est important de noter qu'une condition *suffisante* pour établir la présence d'un concept c, on parlera notamment de *justification complète* de c, est en fait un élément de l'ensemble des *coupes* $\mathbb{C}ut(\boldsymbol{T}_h^c)$ du sous-arbre $\boldsymbol{T}_h^c = (\Downarrow^\infty(c), \Uparrow, \Downarrow)$.

1.3.2 Description des images et caractéristiques

La structuration en ontologie est propice au raisonnement et à la manipulation des concepts de l'expert indépendamment de la donnée d'une image. Lorsqu'il poursuit son processus de justification de la présence d'un concept-cible $h^* \in \boldsymbol{C}_h$ dans l'image X, l'expert se confronte rapidement à une nécessité de décrire : h^* est présent dans la scène parce que certains signes *caractéristiques* de sa présence sont observables dans l'image. Il est intéressant de relever qu'à cet instant de la justification, une vraie rupture s'opère dans le raisonnement par inversion du lien causal. En effet, l'image n'étant qu'une forme de représentation, la réunion de certaines caractéristiques attendues ne saurait constituer une *condition suffisante* à la présence de h^*, et c'est par *abduction* que procède alors l'expert puisqu'il conclut uniquement à titre d'hypothèse que ses observations sont en fait *dues* à la présence de h^* dans la scène.

Un autre point de rupture avec les connaissances *a priori* formalisées dans l'ontologie, réside dans le rattachement essentiel des *caractéristiques* à des mesures quantitatives de phénomènes. Les *caractéristiques* ou *descripteurs* d'une image, d'une région ou d'un pixel sont ainsi calculés par des transformations opérées sur la luminance des pixels et prennent naturellement la forme de valeurs numériques. La qualité d'une transformation est donnée par son pouvoir *discriminant*, c'est-à-dire sa capacité à produire des valeurs différentes pour des concepts distincts dans $\boldsymbol{C}_h$. Dans cette optique, il est bien souvent pertinent d'appliquer plusieurs transformations dont le pouvoir discriminant combiné est plus important que celui des caractéristiques isolées. Très généralement, on appelle *extracteur de caractéristiques* une famille $\boldsymbol{f} = \{f_1, \ldots, f_n\}$ de transformations à valeurs dans $\mathbb{R}$ applicables à une image X, un de ses segments s ou un de ses pixels p. On nomme également *vecteur de caractéristiques* de X, s ou p, leur image par $\boldsymbol{f}$, que l'on note $\boldsymbol{f}(X)$, $\boldsymbol{f}(s)$ et $\boldsymbol{f}(p)$ respectivement. On appelle enfin *espace des caractéristiques* le sous-espace $Im(\boldsymbol{f}) \subset \mathbb{R}^n$ que décrit $\boldsymbol{f}$ lorsqu'elle est appliquée à l'ensemble des images se rattachant au domaine $\boldsymbol{C}_h$ étudié.

L'extraction de caractéristiques dans la littérature fait souvent aussi bien référence à l'ensemble des techniques de conception des transformations, qu'à leur sélection, leur combinaison, ainsi qu'à l'apprentissage statistique des fonctions de description lorsque ces dernières sont définies de manière paramétrique.

1.3.3 Caractérisation locale

Lorsque les données qui doivent être traduites dans le langage humain sont les pixels, il est couramment admis que l'information de l'image entière n'est pas nécessaire pour fournir une description discriminante. Le contexte spatial utile est alors défini comme une zone restreinte

autour du pixel considéré, telle qu'une imagette ou « patch » centrée sur le pixel d'intérêt. La description peut alors faciliter le classement de tous les pixels de l'image, comme cela était le cas pour la segmentation sémantique évoquée précédemment, on parle notamment de *représentation dense*, mais trouve également un intérêt lorsqu'il s'agit de décrire des objets de plus grande taille ; la description permet alors d'identifier et caractériser uniquement certains points d'intérêt aussi appelés *points saillants*, tels que les coins ou les arêtes des objets.

Filtres linéaires Le filtrage linéaire figure parmi les techniques de description locale les plus classiques. Pour une imagette donnée, il consiste à remplacer la valeur du pixel central par une somme pondérée des pixels de l'imagette. Plus rigoureusement, considérons dans l'image de luminance X, une imagette I centrée sur le pixel p, carrée et dont le côté compte t pixels, $I \in \mathbb{R}^{t\times t}$. Un filtre linéaire appliqué à cette imagette prend donc la forme d'un vecteur de poids $F \in \mathbb{R}^{t\times t}$ et la réponse $\boldsymbol{f}(p)$ du pixel p à ce filtre s'écrit comme le produit scalaire $\langle I, F\rangle$ de l'imagette par le filtre. Selon le motif dessiné par les poids du filtre, différentes propriétés de la zone évaluée peuvent être mises en avant et, puisqu'il est question d'analyser des motifs et donc, des répétitions spatiales de certains signaux, c'est ici leur réponse dans le domaine fréquentiel qui nous permet de catégoriser les filtres :

- *Les filtres passe-bas* sont des filtres moyenneurs, la répartition des poids est très homogène et le signal de tout pixel vient « contaminer » celui de ses voisins. Les variations de signal trop brutales sont atténuées, on dit aussi que le filtre coupe les hautes fréquences et les détails de l'image sont généralement perdus.
- *Les filtres passe-haut* sont généralement des filtres qui produisent une forme de gradient discret de l'intensité du signal au point considéré. Les zones homogènes, donc de signal approximativement constant, ont une dérivée spatiale nulle et ne répondront pas à ce type de filtres, on dit aussi que le filtre coupe les basses fréquences. Le filtre répondra d'autant plus fortement que les changements d'intensité lumineuse s'opèrent rapidement dans la zone considérée, comme cela est le cas pour les contours de l'image par exemple.
- *Les filtres passe-bandes* sont des filtres qui vont permettre de ne conserver que des fréquences dans une bande choisie. Ils peuvent notamment permettre de détecter des contours comme le font les filtres passe-haut, tout en filtrant les fréquences trop hautes, qui correspondent souvent à du bruit dans l'image plutôt qu'à de veritable frontières entre les objets.

Points saillants Il est question de points saillants (*salient points*) lorsque l'image ou un objet dans l'image est repéré par un jeu de points-clefs. Cette approche est particulièrement explorée pour des applicatifs d'estimation de pose ou de recalage souvent nécessaires aux calculs de déplacements des robots, pour lesquelles les contraintes de vitesse d'exécution (temps réel) ou de robustesse (dégâts engendrés par une mauvaise appréciation du positionnement d'un robot) sont très fortes et ont longtemps été incompatibles avec une description *dense*, c'est-à-dire de tous les pixels de l'image. Ces techniques satisfont d'une part la contrainte de vitesse d'exécution en ne calculant des descripteurs que sur un jeu de points-clefs dont le nombre est restreint et la détection rapide. D'autre part, l'approche satisfait à la contrainte de robustesse en décrivant les points-clefs avec des caractéristiques peu sensibles aux transformations de l'image : deux points décrivant une même entité, provenant respectivement d'une image et de sa version transformée, doivent avoir des descriptions similaires de façon à être automatiquement appariés par similarité. Les descripteurs des points saillants doivent donc respecter des propriétés d'invariance à l'échelle, à la rotation ou à l'illumination pour pallier aux différentes transformations envisageables. La stratégie de l'algorithme SIFT (*Scale Invariant Features*

Transform), conçue par Lowe [1999], reste de très loin le travail le plus abouti dans le domaine des points saillants et c'est notamment selon leur approche que nous décrivons la détection et la caractérisation de ces points dans une perspective d'invariance aux transformations.

Les points-clefs sont souvent définis comme des *coins* dans l'image. Les coins des objets présentent le double avantage d'être relativement peu nombreux et d'être détectable formellement par application de filtres linéaires de dérivation spatiale et par calcul d'une matrice hessienne locale. La méthode de Harris et al. [1988] fait notamment office de référence pour la détection des *coins*. Les coins sont intuitivement des descripteurs dotés d'un pouvoir discriminant très important puisqu'ils indiquent, par leur position, une excellente signature géométrique de l'objet à détecter et fournissent du même coup une information essentielle pour en estimer la pose. Lowe [1999] envisage l'invariance d'échelle dès la détection des points de ses descripteurs en lançant une détection sur plusieurs niveaux de détails, matérialisés par des flous gaussiens successifs, et à plusieurs niveaux de sous-échantillonnage de l'image.

Une fois détectés, les coins doivent encore être décrits de manière à augmenter encore leur pouvoir discriminant. La description doit être réalisée sous les contraintes restantes d'invariance à la rotation et à l'illumination.

- *L'invariance à la rotation* est garantie par l'établissement d'une orientation naturelle du coin détecté qui sert ensuite de référentiel à tous les descripteurs angulaires calculés sur le point-clef. L'orientation dite « naturelle » est calculée sur la base de la courbure principale du contour auquel appartient le point d'intérêt. Les autres descripteurs angulaires sont ainsi calculés sur l'orientation des gradients dans une partition du voisinage du point-clef étudié Seul un nombre restreint d'orientation est considéré et un histogramme de ces orientations fait office de description pour une partie du voisinage donné. L'orientation naturelle du point est alors soustraite à l'ensemble des orientations de gradients calculées de manière à fournir un descripteur relatif et donc insensible aux rotations de l'image.
- *L'invariance à l'illumination* est permise par normalisation des histogrammes, les intensités absolues sont alors perdues et seuls les rapports des gradients et des orientations les uns aux autres sont alors conservés. Les transformations globales et uniformes de l'illuminations n'affectent donc plus la description des points-clefs. La sensibilité aux illuminations locales peut également être minimisée en seuillant les valeurs des histogrammes de manière à limiter l'influence de gradients trop prononcés.

Co-occurence des niveaux de gris Suivant les travaux de Haralick [1979], une méthode efficace d'analyse de la texture locale ou globale d'une image consiste à relever les fréquences spatiales des paires de niveaux de gris. Cela est notamment permis par le calcul de matrices de co-occurences qui dénombrent la présence d'une configuration spatiale prédéfinie pour toute paire de niveaux de gris présente dans l'image. Une matrice de co-occurence est calculée par configuration spatiale considérée, et pour alléger la vitesse d'éxecution de la procédure, la méthode préconise de n'utiliser que le 4-voisinage élémentaire des pixels. Ainsi, pour un déplacement élémentaire (dx, dy) donné, le coefficient (i, j) de la matrice de co-occurence $C_{(dx,dy)}$ correspondante s'obtient en dénombrant les paires de pixels (x, y) et $(x + dx, y + dy)$ dont les niveaux de gris valent respectivement i et j :

$$C_{(dx,dy)}(i,j) = \sum_x \sum_y \delta_{i,I(x,y)} \times \delta_{j,I(x+dx,y+dy)} \tag{1.6}$$

Les matrices de co-occurence en elles-mêmes sont des représentations très volumineuses, difficiles à interprêter ou utiliser pour décrire une image et Haralick [1979] proposent ainsi 14 mesures statistiques calculées sur la base de cette matrice afin de caractériser une texture.

Encodage de motifs élémentaires Parmi les méthodes de description locales, les motifs binaires locaux, ou *LBP* (*Local Binary Patterns*), développés par Ojala et al. [1994], forment une méthode à part entière difficilement rattachable aux approches précédentes. La méthode propose de seuiller chaque fenêtre de contexte considérée par la valeur de son pixel central et d'attribuer, en guise de descripteur du patch, une référence au motif binaire produit par l'opération. Pour un patch comptant t pixels de côté, $2^{t\times t}$ motifs différents peuvent être encodés et on appelle *dictionnaire d'unités texturelles* l'ensemble de ces motifs. Chaque pixel de l'image peut alors être décrit par une unique référence à un élément du dictionnaire et une méthode de référencement classique consiste simplement à attribuer au pixel la valeur en base 10 encodée par le motif binaire de l'unité texturelle à laquelle il est rattaché.

1.3.4 Caractérisation globale et stratégies d'agrégation

Des processus de description sont aussi développés pour caractériser des segments d'images ou des images entières. Ces techniques sont majoritairement pensées comme des stratégies d'intégration ou d'*agrégation* des descripteurs locaux décrits ci-dessus.

Originellement mise au point pour analyser des données textuelles par Harris [1954], la méthode des *sacs de mots* (*Bag of Words*), aussi abrégée « BoW », est la méthode emblématique de l'intégration de descripteurs vers des caractéristiques globales. Elle représente un texte ou *corpus* sous forme creuse en répertoriant l'occurence des différents mots ; un mot étant défini comme une chaîne de caractère séparée du reste du texte par des espaces. En dehors des descripteurs à valeurs discrètes, qui rattachent tout pixel à une clef dans un dictionnaire de textures, comme cela est le cas pour les *LBP*, les mots linguistiquement bien définis pour l'applicatif textuel n'ont évidemment pas d'équivalent dans l'espace des images ou des segments d'images. Dans le cas général des descripteurs locaux à valeurs réelles, la méthode procède donc en deux étapes : un **vocabulaire** est d'abord constitué en détectant des motifs statistiquement significatifs dans l'espace des caractéristiques ; les points décrits localement peuvent ensuite être **traduits** en mots du vocabulaire et l'**agrégation** des différents mots dans une zone, notamment par le biais d'un histogramme regroupant les fréquences d'observation des différents mots permet de fournir un descripteur de la zone.

Constitution du vocabulaire Les équivalents des *mots* dans le contexte de l'analyse d'image sont des regroupements statistiquement significatifs de descripteurs locaux relevés dans une banque d'image. Leung and Malik [2001] réalisent cette opération de regroupement ou *clustering* par un algorithme des k-moyennes utilisant la distance euclidienne dans l'espace des descripteurs comme mesure de similarité entre les pixels. À l'issu de ce traitement, une partition aussi appelée *vocabulaire* et notée $\boldsymbol{V}$ de l'espace des caractéristiques est obtenue ; les clusters ou catégories fournies par la méthode définissent alors les *mots* du vocabulaire $\boldsymbol{V} = \{m_1, \ldots, m_k\}$. De manière à les démarquer des caractéristiques locales dites de « bas-niveau », les *mots* définis par la procédure de clustering apparaissent souvent dans la littérature sous l'appellation *caractéristiques de moyen-niveau*. Cette étape est une forme de classement non-supervisé et on notera $\boldsymbol{C}_m$ l'espace des *mots* extraits pour « concepts de la machine » par opposition à l'ensemble $\boldsymbol{C}_h$ des concepts humains.

Traduction Chacun des mots m de $\boldsymbol{C}_m$ peut être représenté par sa distribution dans l'espace des caractéristiques. La littérature décrit alors plusieurs stratégies d'*encodage en niveau moyen* qui traduisent les descripteurs locaux en mots du vocabulaire $\boldsymbol{V} \subset \boldsymbol{C}_m$. Il n'y a *a priori* aucune garantie que les mots du vocabulaire puisse constituer une base dans l'espace des caractéristiques et la décomposition d'un concept bas-niveau en mots de niveau sémantique moyen consiste donc à résoudre un problème *mal posé*.

- *L'assignation dure* consiste par exemple à attribuer à chaque descripteur local de la région considérée, le mot de vocabulaire le plus proche en distance euclidienne. Lorsque tous les descripteurs ont été étiquetés, il est possible de calculer le nombre d'occurences de chacun des mots dans la région et de produire un histogramme de $\boldsymbol{V}$ qui sert alors de descripteur du segment. Le reste des méthodes fonctionne sur un système d'*assignation floue* dans laquelle un descripteur est considéré comme une combinaison linéaire des différents mots. La définition des coefficients de pondération varie ainsi selon les formulations et méthodes de résolution du problème.
- *La décomposition linéaire* consiste à traduire chaque descripteur local de la région considérée par une combinaison linéaire des éléments de $\boldsymbol{V}$. Lee et al. [2007], Yang et al. [2009], Wang et al. [2010] calculent une combinaison acceptable par résolution d'un problème des moindres carrés sous divers contraintes et termes de régularisation.
- *La méthode de Parzen* consiste à remplacer les boîtes fixes de l'histogramme, obtenues dans l'assignation dure, par des gaussiennes. Ainsi, Bilmes et al. [1998] pondèrent la part que prend chaque descripteur dans la constitution de l'histogramme par sa probabilité d'appartenance à une distribution gaussienne centrée autour d'un mot du vocabulaire.
- *La méthode de Fisher* envisage plutôt un mélange de gaussiennes. Chacun des mots dans $\boldsymbol{V}$ est décrit par une gaussienne dont les paramètres peuvent notamment être déterminés par apprentissage statistique. À la différence de Parzen, Perronnin and Dance [2007] et Perronnin et al. [2010] pondèrent l'appartenance d'un descripteur aux différents mots est également pondérée par la distribution *a priori* sur les éléments de vocabulaire qui prend la forme d'un jeu de paramètres, aussi appelés *poids* du mélange gaussien.

Agrégation Comme indiqué dans le cas de l'assignation dure, la constitution d'un histogramme reste la méthode d'agrégation ou *pooling* la plus répandue. On parle notamment dans ce cas d'*agrégation par moyenne* ou *average pooling*, comme le décrivent Csurka et al. [2004]. La méthode d'*agrégation par maxima* ou *max pooling* est souvent privilégiée, notamment par Boureau et al. [2010a] et Boureau et al. [2010b], parce qu'elle implémente une logique de détection qui permet notamment de ne pas perdre un signal d'intérêt présent qui par effet de moyenne pourrait être noyé au milieu de signaux peu informatifs.

1.3.4.1 Représentations hiérarchiques

Arbre de subsomption L'édification d'un vocabulaire $\boldsymbol{V}$, dans le processus de caractérisation globale, s'apparente à un partitionnement de l'espace des caractéristiques locales $Im(\boldsymbol{f})$. Dans le cas d'un assignement *dur* et de manière tout à fait analogue à la segmentation des images, la relation d'inclusion naturelle sur les ensembles, ici les parties de l'espace des caractéristiques locales, permet de définir une relation $\leq$ de *raffinement* entre les vocabulaires par respect d'une propriété identique à l'Equation 1.4. Soient $\boldsymbol{V}_i$ et $\boldsymbol{V}_j$ deux vocabulaires définis dans l'espace des mots de la machine, on dira que $\boldsymbol{V}_j$ est un *raffinement* de $\boldsymbol{V}_i$, ou encore, si l'on fait références aux *mots*, que les termes de $\boldsymbol{V}_j$ précisent ceux de $\boldsymbol{V}_i$ lorsque :

$$\forall m_j \in \boldsymbol{V}_j,\ \exists m_i \in \boldsymbol{V}_i \mid m_j \subseteq m_i \tag{1.7}$$

Suivant toujours l'exemple du partitionnement des images, nous envisageons encore des ensembles ordonnés de vocabulaires logiquement assimilés à leur structure arborescente $\boldsymbol{T}_m = (\boldsymbol{C}_m, \Uparrow, \Downarrow)$. Contrairement à la segmentation d'image cependant, l'espace des caractéristiques est dépourvu de voisinage et il n'est pas ici question de contenance spatiale, mais de contenance sémantique : les mots d'un vocabulaire sont regroupés dans un mot de vocabulaire moins raffiné par ressemblance, c'est-à-dire par similarité entre leurs descripteurs. Si $\Uparrow(m_j) = m_i$ alors il est entendu que le mot m_j illustre un cas particulier du terme m_i dont le sens est

plus général. Nous parlerons dans ce cas de *subsomption* comme décrit par Blanchard [2008] plutôt que de *partinomie* et, comme indiqué dans le Sous-section 1.3.1, nous privilégierons la notation $m_j \ \mathcal{S} \ m_i$ à la relation d'inclusion traditionnelle « $\subseteq$ ».

Chaîne de perception Sur la base d'une assignation *floue*, l'opération de **traduction** exprime tout pixel ou segment considéré comme un mélange des mots d'un vocabulaire $\boldsymbol{V}_0$. La différence principale avec l'assignation *dure* réside dans la considération de classes chevauchantes : si un élément $x \in Im(\boldsymbol{f})$ appartient vraisemblablement à une classe $c_i \in \boldsymbol{V}_0$, son appartenance à une autre classe $c_j \in \boldsymbol{V}_0$ n'est pas jugée nulle, mais moins probable. Les proportions du mélange des classes peuvent aussi bien être interprétées par des probabilités d'appartenance d'un segment aux différents mots du vocabulaire, que par une quantification de l'aire occupée par chacun des mots de vocabulaire dans le contexte du segment. Quelle que soit l'interprétation choisie cependant, la traduction prend la forme d'une application qui, pour un pixel ou segment d'image donné, permet de passer d'un vecteur descripteur à un autre $\mathcal{T}_0 : \mathbb{R}^n \to \mathbb{R}^k$.

Dès lors, l'espace des mots $Im(\mathcal{T}_0) \subset \mathbb{R}^k$ peut à son tour faire l'objet d'un partitionnement en vocabulaire $\boldsymbol{V}_1$ pour lequel un nouveau traducteur $\mathcal{T}_1$ est formulé, si bien qu'une suite de paires vocabulaire-traducteur $(\boldsymbol{V}, \mathcal{T})_n$ est récursivement constituée. Dans le cadre de l'assignation *floue*, la règle de récurrence envisage le vocabulaire de rang supérieur $\boldsymbol{V}_{n+1}$, non plus comme un simple partitionnement de données, mais comme une véritable recherche de distributions explicatives postérieures aux observations $Im(\mathcal{T}_n)$. Une relation d'ordre *explicative* existe donc entre un vocabulaire $\boldsymbol{V}_n$ et le vocabulaire de rang supérieur $\boldsymbol{V}_{n+1}$; on dira ainsi que le vocabulaire $\boldsymbol{V}_{n+1}$ *explique* les observations $Im(\mathcal{T}_n)$ des mots du vocabulaire $\boldsymbol{V}_n$ et on notera $\boldsymbol{V}_{n+1} \ \mathcal{E} \ \boldsymbol{V}_n$. Lors de l'inférence sur un pixel ou un segment d'image, l'opération de *traduction* s'assimile à une *abduction* en cela qu'elle détermine l'*explication* la plus probable $\mathcal{T}_{n+1}(x)$ à une conséquence $x \in Im(\mathcal{T}_n)$ observée ; on dira par la suite que x *abduit* $\mathcal{T}_{n+1}(x)$ et on notera $x \ \mathcal{A} \ \mathcal{T}_{n+1}(x)$.

La chaîne abductive $\mathcal{T}_n$ manipule des concepts de complexité sémantique croissante afin d'expliquer la présence d'un pixel ou segment dans une image. Sans en explorer l'implémentation technique, les travaux de Josephson and Josephson [1996] sur le raisonnement abductif envisagent déjà une telle suite pour décrire les mécanismes de la perception et s'expriment d'ailleurs sans détour à ce sujet : « perception is abduction in layers ». Nous ferons à notre tour allusion à la perception lorsqu'il s'agira de détailler la construction des vocabulaires et des traducteurs associés et il sera fréquemment question de *chaîne de perception* lorsque nous ferons référence à la suite $\mathcal{T}_n$.

1.3.5 Bilan

De nombreux outils, construits selon des fondements théoriques variés, existent pour décrire des images ou certaines de leurs zones d'intérêt par un jeu de valeurs numériques. Pour une région d'image, le vecteur caractéristique permet d'accéder à un certain nombre d'informations de niveau sémantique supérieur à celui du signal d'un pixel. Ces techniques sont destinées à faciliter la traduction du signal d'une image vers la connaissance de l'expert, dont les concepts constitutifs, même les plus élémentaires, ne peuvent être exprimés en termes de valeurs de luminance de pixels.

Cette section s'est notamment attachée à mettre en évidence les différents raisonnements à l'œuvre, à la fois dans la construction des descripteurs élémentaires, mais aussi dans l'accès aux niveaux sémantiques supérieurs ainsi que dans l'inférence d'une sémantique abstraite à partir de l'observation des pixels. Le développement ci-dessus souligne notamment le fossé,

aussi connu sous le nom de *fossé sémantique*, existant entre la manipulation des concepts de l'expert et l'extraction de ces mêmes concepts dans une scène représentée par une image numérique. S'il mesure couramment une difficulté de traduction d'un langage vers un autre, disons des concepts abstraits de l'expert vers le langage formel de la machine par exemple, nous matérialisons ici le *fossé sémantique* par la rupture de raisonnement qu'il occasionne entre la *déduction*, qui régit la manipulation des concepts humains, et l'*abduction*, qui règne sur les mécanismes de perception.

Qu'elles soient hiérarchiques et à assignation dure pour construire un *arbre de subsomption*, ou bien plate et à assignation floue pour constituer une *chaîne perceptuelle*, les paragraphes précédents font maintes fois référence à des procédures de partitionnement de l'espace des caractéristiques, que l'on appelle aussi méthodes de *classement* ou *classification*. L'approche des *sacs de mots* rassemble des motifs statistiquement significatifs pour constituer ses classes. intrinsèquement non-supervisé, ce procédé ne garantit pas une superposition optimale des classes obtenues avec celles du langage de l'expert $\boldsymbol{C}_m \neq \boldsymbol{C}_h$, contrairement au cas de la segmentation sémantique des images dont l'objectif principal était un rattachement direct des pixels à des éléments de $\boldsymbol{C}_h$. Le partitionnement de l'espace des caractéristiques apparaît donc indispensable aussi bien à la confection et à la complexification des descripteurs d'une image, qu'à la prédiction des concepts humains qui y sont représentés. L'opération de *classification* semble donc omniprésente dans les processus d'analyse et il s'agit d'en étudier les principaux aspects dans la suite de ce chapitre.

1.4 Partitionnement des caractéristiques : classification

Déjà cités précédemment au sujet de l'extraction de caractéristiques, Bengio et al. [2013] dissocient ce processus d'extraction de « paramètres explicatifs » de l'algorithme décisionnel à proprement parler. Ils ajoutent à ce propos que les descripteurs de l'image doivent être déterminés avec l'objectif de « faciliter la tâche aux algorithmes de classification ou de prédiction ».

Dans leur ouvrage plus ancien, Duda et al. [1973] s'expriment déjà selon des termes similaires à ceux de Bengio et al. [2013], mais ne manquent pas de souligner une certaine subjectivité dans la définition : « La frontière conceptuelle entre l'extraction de caractéristiques et la classification est quelque peu arbitraire : un extracteur de caractéristiques idéal devrait produire une représentation qui rend triviale la tâche du classifieur ; inversement, un classifieur omnipotent n'aurait pas besoin de s'appuyer sur une extraction de caractéristiques sophistiquée. La distinction s'impose plutôt à nous pour des raisons pratiques que pour des considérations théoriques ».

Pour une image X et un jeu de concepts-cibles $\{h_1, \ldots, h_k\} \subset \boldsymbol{C}_h$ de l'espace des connaissances de l'expert, la prédiction sur X consiste à fournir un vecteur $y \in \mathbb{R}^k$ dont les composantes expriment les probabilités de présence des différents concepts recherchés. Pour une chaîne de perception $\mathcal{T}_n$ et un rang N donné, la prédiction de y se définit comme un nouveau traducteur $\mathcal{T}_{N+1} : \mathbb{R}^n \to \mathbb{R}^k$ qui, à partir de la composée des N premiers termes de la chaîne perceptuelle appliquée à X, prédit les probabilités d'appartenance à chacune des classes d'intérêt.

La classification en ce sens, s'intègre parfaitement dans le processus d'extraction de caractéristiques défini dans le cadre des sacs de mots : elle implique, au même titre que tout autre maillon de la chaîne de perception, un partitionnement en vocabulaire et un traducteur. De plus, et comme l'indiquent Bengio et al. [2013] et Duda et al. [1973], l'ensemble de la chaîne est effectivement construite de façon à rendre la définition de ce dernier traducteur $\mathcal{T}_{N+1}$ la plus élémentaire possible.

Néanmoins, ce dernier traducteur est le seul élément de la chaîne à établir explicitement un lien avec le vocabulaire de l'expert. Une distinction s'opère donc si l'on restreint la définition de *classification* aux partitions de données qui reflètent les classes reconnues par l'expert humain. Nous choisissons dans cette section d'explorer le *classement* comme une technique de partitionnement au sens large, qui couvre à la fois l'usage de l'extraction de caractéristiques et celui de la prédiction des concepts de l'expert. Nous articulerons toutefois le développement en deux parties : dans un premier temps nous traiterons la classification *non-supervisée*, qui construit les vocabulaires de la chaîne de perception, puis, dans un second temps, nous aborderons le classement *supervisé*, qui a trait à la prédiction d'informations intelligibles.

Notations

— Soit $X = \{x_1, \ldots, x_n\}$, l'ensemble des descripteurs d'images ou de segments à classer, $\forall i, x_i \in \mathbb{R}^d$ et, afin d'éviter toute confusion entre l'indexation de l'ensemble à classer et celle des composantes d'un individu, nous noterons x_j^i la i-ème composante du j-ème individu de X ;

— Dans le cas supervisé, à tout élément $x \in X$, l'expert associe un élément de son vocabulaire $y^* \in [1, k] \subseteq \boldsymbol{C}_h$;

— L'annotation de l'expert y^* symbolisera aussi bien l'indice de classe de l'individu x, que son encodage *one-hot* donné par le vecteur parcimonieux de $\mathbb{R}^k$ dont toutes les valeurs sont nulles à l'exception de la y^*-ième composante qui prend la valeur 1 ;

— On notera par ailleurs f^* la fonction d'annotation humaine qui associe à tout individu l'étiquette que lui a attribuée l'expert $f^*(x) = y^*$;

— La prédiction y correspond à la sortie d'un modèle de classification automatique f et fera encore une fois aussi bien référence à la classe choisie dans $[1, k]$ qu'à un vecteur de $\mathbb{R}^k$, non parcimonieux cette fois-ci, et qui regroupera les probabilités d'appartenance à chacune des classes de $[1, k]$;

— Soit $\mathcal{C}^* = \{C_1^*, \ldots, C_k^*\}$ tel que $\bigcup_i C_i^* = X$ et tel que $\forall i \in [1, k], C_i^* = \{x \mid f^*(x) = i\}$, le résultat de la classification réalisée par l'expert ;

— Soit $\mathcal{C} = \{C_1, \ldots, C_k\}$ tel que $\bigcup_i C_i = X$ et tel que $\forall i \in [1, k], C_i = \{x \mid f(x) = i\}$, le résultat de la classification réalisée par la machine ;

— De manière très générique, d fera référence aux mesures de dissimilarité, ou *distance*, entre deux individus et l'on notera $d(x_i, x_j) = \|x_i - x_j\|^2$ dans le cas du carré de la norme euclidienne par exemple ;

— On notera enfin μ_i le centroïde du cluster C_i défini par $\mu_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$.

1.4.1 Classement non-supervisé

La classification non-supervisée, que l'on appelle souvent *clustering*, consiste à partitionner un espace de caractéristiques sans formuler explicitement l'objectif d'obtenir des parties coïncidentes avec celles de l'expert des images. Bien entendu, l'objectif premier de l'analyse automatique est de fournir un résultat interprétable afin de trouver une utilité pour les experts des images. L'usage des méthodes de classement non-supervisé est auxiliaire et forme une étape simplificatrice préliminaire à la prédiction utile. Enfin, un avantage majeur de cette approche est évidemment de pouvoir extraire des sous-populations d'individus qui échappent aux experts ou ne sont traditionnellement pas pris en compte dans la prise de décision.

Le problème de partitionnement ressemble donc à celui de la segmentation des images à contrainte sémantique faible, à cela près que la notion de voisinage n'est *a priori* pas définie dans l'espace des caractéristiques. Ainsi, alors qu'un pixel ne pouvait être regroupé qu'avec des

pixels adjacents, un individu de X doit potentiellement être comparé avec tous les autres pour trouver les éléments avec lesquels il peut être regroupé. Ce degré de liberté supplémentaire dans la définition du problème augmentera parfois sensiblement la complexité algorithmique des solutions envisagées.

Il s'agit donc de regrouper des individus de X afin d'en faire émerger des catégories, ou *classes*. Une classe se définit évasivement comme un groupe d'individus partageant certaines caractéristiques qu'ils ne partagent pas avec le reste des données. La classe doit être :

- statistiquement bien représentée ;
- suffisamment homogène ;
- suffisamment distincte des autres membres de l'espace des caractéristiques.

Les différentes techniques de partitionnement vont ainsi différer selon la manière de formuler les bonnes propriétés énoncées ci-dessus et dans le compromis qu'elles imposeront entre ces différents critères dans la formulation du problème.

1.4.2 Algorithmes principaux du classement non-supervisé

Méthodes à base de prototypes Ces techniques de partitionnement s'apparentent notamment aux méthodes basées sur des atlas ou au template-matching. Elles constituent un élément représentant de chacune des classes, appelé *prototype* de classe, auquel tout nouvel élément inconnu est comparé. La classe inférée pour ce nouvel élément est celle du prototype auquel il ressemble le plus.

La méthode phare de cette approche est l'algorithme des *k-moyennes*, ou *k-means*, développée par MacQueen et al. [1967] et qui propose de déterminer les prototypes de k parties dans X. Le nombre k de parties est fixé par l'utilisateur et la méthode place alors aléatoirement k prototypes dans l'espace des caractéristiques qu'elle fait itérativement converger vers les k zones de l'espace des données qui minimisent la variance intra-classe.

Cet algorithme et ses dérivés, tels que sa version floue des *Fuzzy-C-means* mise au point par Dunn [1974], convergent très rapidement et sont d'un coût algorithmique dérisoire. Le problème d'optimisation de la variance intra-classe présuppose cependant des clusters convexes et de même taille, ce qui rend le résultat de ces méthodes peu satisfaisant sur les frontières des classes. Ces procédures sont donc souvent envisagées comme un point de départ à d'autres méthodes itératives plus raffinées, telles que les mixtures de gaussiennes, optimisées par l'algorithme *EM* (*Expectation-Maximization*), pour autoriser des formes de clusters plus hétérogènes.

Méthodes à base de densité En faisant appel à une définition très physique et intuitive des densités de probabilités, les parties d'intérêt dans l'espace des caractéristiques peuvent être naturellement considérées comme des régions à forte concentration d'individus, séparées par des zones peu occupées par les données.

Définie physiquement par le nombre d'individus rapporté au volume considéré, la concentration peut être évaluée localement : l'échantillon $x_i \in X$ et ses voisins dans une sphère de rayon r, $\mathcal{V} = \{x_{j\neq i} \mid d(x_i, x_j) \leq r\}$, appartiennent à une même classe à condition que le nombre d'individus présents dans la sphère, soit supérieur à un seuil de concentration $|\mathcal{V}| > \epsilon$. Le rayon r de la sphère, ainsi que le seuil ϵ sur la concentration des clusters, sont les paramètres de la méthodes devant être fixés au préalable par l'utilisateur.

Le caractère local du critère de regroupement permet notamment de faire abstraction de tout postulat sur la structure des nuages de points à dissocier. Les techniques qui implémentent cette approche, telles que l'algorithme *DBSCAN* de Ester et al. [1996] par exemple, sont particulièrement respectueuses de la géométrie implicite des données et autorisent notamment le découpage de l'espace en parties non-convexes.

Méthodes à base de neurones Les méthodes neuronales sont principalement décrites comme des techniques de réduction des dimensions de l'espace caractéristique. L'algorithme principal *SOM* (*Self Organising Map*), proposé par Kohonen [1982], projette l'espace des données sur une carte dite *auto-adaptive* en deux dimensions.

À chaque case de la carte est associé un neurone qui s'active en réponse à un *stimulus* bien particulier. Les neurones sont d'abord initialisés au hasard puis, la procédure d'apprentissage s'assure que des neurones voisins répondent à des *stimuli* similaires, si bien qu'à la fin de l'apprentissage, des stimuli de même nature activent les mêmes régions de la carte.

Au même titre que les méthodes à base de densité, l'influence très localisée d'un *stimulus* sur la carte permet de cerner la topologie des données, elle va même au-delà du respect de la géométrie interne des classes en faisant également apparaître les ressemblances inter-classes, puisque des clusters proches en termes de description seront voisins sur la carte auto-adaptive.

Le nombre de neurones sur la carte est généralement élevé afin d'enregistrer les groupes, y compris les moins représentés statistiquement, ainsi que les nuances entre les groupes. Cependant, le nombre de catégories attendues est souvent bien inférieur au nombre de neurones de la carte, si bien qu'un algorithme de segmentation d'images est souvent encore appliqué à la carte afin d'en extraire les segments les plus homogènes correspondant à un nombre plus restreint de catégories statistiquement bien représentées.

Classification ascendante hiérarchique La classification ascendante hiérarchique, ou HAC pour *Hierarchical Agglomerative Clustering*, définit la procédure de construction de l'arbre de subsomption décrit précédemment. Au démarrage de la procédure, chacun des individus de X constitue une classe et à chaque étape de la procédure, les deux classes les plus similaires sont fusionnées. Un certain nombre de mesures peuvent servir de similarité, mais le minimum de la distance euclidienne est souvent choisie comme critère de fusion. Quelle que soit la distance choisie, elle n'est définie qu'entre deux individus et la distance entre deux classes peut faire l'objet de différentes stratégies pour le calcul de similarité. Ainsi, pour une paire de clusters (C_i, C_j) :

- le *single linkage* détermine la distance minimale parmi les couples de $C_i \times C_j$:

$$\min_{(u,v)\in C_i\times C_j} d(u,v)$$

- le *complete linkage* calcule la distance maximale entre les couples de $C_i \times C_j$:

$$\max_{(u,v)\in C_i\times C_j} d(u,v)$$

- l'*average linkage* calcule la distance entre les centroïdes de C_i et C_j : $d(\mu_i, \mu_j)$

1.4.3 Evaluation des partitions non-supervisées

Il existe plusieurs mesures d'évaluation des partitions obtenues sans supervision qui consistent globalement à quantifier le respect des bonnes propriétés visées par la tâche de clustering. En outre, la compacité des clusters est utilisée comme indicateur de l'homogénéité

des classes, des valeurs de distances entre les clusters servent à étudier leur séparabilité et le rapport de la compacité sur la séparabilité permet généralement de compiler un critère général intégrant les deux propriétés.

La distance moyenne des individus à leurs prototypes est ainsi souvent utilisée pour évaluer la compacité des clusters comme dans les études menées par Davies and Bouldin [1979] ou Wemmert [2000], mais la distance maximale entre deux individus d'un même cluster, qui ramène à la définition intuitive du diamètre d'un cluster, a aussi été envisagée par Dunn [1974], ainsi que par Rousseeuw and Kaufman [1990].

La séparabilité entre les clusters est également souvent évaluée en moyennant l'ensemble des distances inter-classes selon une stratégie de *linkage* choisie, *average linkage* dans le cas de Davies and Bouldin [1979] ou *single linkage* pour Dunn [1974].

1.4.4 Classement supervisé

À l'expression *classification non-supervisée* se sont beaucoup substitués les termes anglais *clustering* ou, plus généralement, *data mining*, si bien que, sans précision supplémentaire, le terme *classification* fait bien souvent référence à la *classification supervisée*. Cette situation de classification s'avère, et de très loin, être l'approche de partitionnement la plus fréquemment explorée dans la littérature actuelle, et ce pour deux raisons principales. Tout d'abord parce qu'elle fournit un accès direct aux concepts d'intérêt que l'expert souhaite prédire à partir des images ; elle garantit *de facto* une utilité immédiate au domaine applicatif. Ensuite parce qu'elle connaît un essor technique sans précédent. Les méthodes à base de réseaux neuronaux convolutifs profond notamment, produisent désormais des interprétations d'images d'une complexité jusqu'alors inespérées de la part d'un outil automatique.

Ce problème de partitionnement ressemble à celui de la segmentation à contrainte sémantique forte. Dans cette configuration, un algorithme de classement s'appuie sur la donnée d'un ensemble d'entraînement, ou ensemble d'apprentissage, X_t et de son classement $\mathcal{C}_t^*$, déjà réalisé par des experts, pour établir des règles de prédiction applicables sur des données encore inconnues ; on parle notamment de *généralisation* des connaissances rassemblées dans l'ensemble d'apprentisssage.

1.4.5 Algorithmes principaux du classement supervisé

Méthode des K plus proches voisins La classification par la méthode des k plus proches voisins, souvent notée K-NN (*K-nearest neighbors*), repose sur un principe particulièrement simple selon lequel un individu doit avoir la même classe que les individus qui lui ressemblent. Ainsi, pour tout élément x inconnu, les k éléments de X_t les plus proches de x, souvent au sens de la distance euclidienne, votent pour la classe à attribuer à l'élément x. La classe choisie correspond donc à la classe la plus représentée parmi les k éléments annotés sélectionnés.

De manière intéressante, cet algorithme ne comporte aucune phase dite d'*apprentissage* durant laquelle il tente explicitement d'établir une fonction de prédiction à partir de l'ensemble d'entraînement ; il garde le jeu de données d'apprentissage en mémoire pour évaluer la fonction de séparation localement et à la demande de l'inférence. Ce comportement, assez unique parmi les méthodes de classification supervisée, porte le nom de *lazy learning*, qui rend compte de l'absence de généralisation antérieure à l'instant de l'inférence : l'algorithme n'a « pas travaillé » avant d'avoir vu la tâche de prédiction à accomplir.

Dans son implémentation naïve, l'algorithme impose un parcours linéaire du tableau des descripteurs de X_t à chaque inférence et devient rapidement coûteux selon la taille de l'ensemble d'apprentissage considéré. Ainsi, un certain nombre d'algorithmes, souvent plus complexes, mais à n'exécuter qu'une seule fois avant une liste d'inférences, construisent une structure de données telles que les *arbres k-d* de Bentley [1975] ou les arbres *PAT* (*Principal Axis Trees*) de McNames [2001] et permettent de dresser la liste des plus proches voisins selon une complexité moyenne en $O(\log n)$.

Classifieurs linéaires Lorsqu'il s'agit de séparer deux classes dans l'espace des caractéristiques, une méthode courante exprime la frontière, ou plutôt l'appartenance d'un individu x à une classe, comme une combinaison linéaire des composantes de x. Le problème se résume donc à déterminer un vecteur $w \in \mathbb{R}^d$, ainsi qu'un biais $b \in \mathbb{R}$ tel que :

$$\langle w, x \rangle + b \begin{cases} \geq 0 & \Rightarrow f(x) = 1 \\ < 0 & \Rightarrow f(x) = 2 \end{cases} \tag{1.8}$$

La donnée de (w, b) est la définition d'un hyperplan dans $\mathbb{R}^d$ et la fonction de classification consiste simplement à étudier la position relative d'un individu par rapport à cet hyperplan. Si l'équation sépare bien les données, c'est-à-dire qu'elle produit peu ou pas d'erreurs de classement, alors l'ensemble des données est dit *linéairement séparable* et bien que l'hypothèse d'une séparation linéaire des classes puisse paraître simpliste, la chaîne de perception qui aboutit aux descripteurs de X est généralement conçue pour fournir ce genre de bonne propriété. Dans le cas contraire, certaines stratégies ont pu être développées pour transformer le problème en classification linéaire malgré tout.

Classement linéaire par modèle bayésien naïf La tâche de classification linéaire peut être abordée sous l'angle des probabilités. Elle est dans ce cas posée comme un problème de *maximum a posteriori*. À un individu $x \in X$, la méthode associe la classe $\hat{y}$ la plus probable compte tenu des distributions de chacune des classes relevées dans l'ensemble d'apprentissage :

$$\hat{y} = \underset{y \in [1,k]}{\arg\max}\, \mathbb{P}(Y = y | X = x^1, \ldots, x^d) \tag{1.9}$$

La méthode *bayésienne naïve* est une méthode *générative* qui propose de résoudre le problème par estimation de la *vraisemblance* $\mathbb{P}(X|Y)$. À la différence de la formulation exacte de la vraisemblance, le cas de la classification linéaire couvert pas l'approche *baysésienne naïve* fait un certain nombre de suppositions sur la distribution des données afin de simplifier l'apprentissage et de ramener la prédiction à une séparation classique par hyperplan :

- La méthode suppose d'abord une *indépendance conditionnelle* des différentes composantes descriptives des individus. Cette hypothèse autorise notamment, après passage par la règle de Bayes, d'écrire la probabilité d'appartenance à une classe conditionnellement à l'observation d'un descripteur de la manière suivante :

$$\mathbb{P}(Y = y | X = x^1, \ldots, x^d) = \frac{1}{Z} \mathbb{P}(Y = y) \prod_i \mathbb{P}(X = x^i | Y = y) \tag{1.10}$$

- Elle fait ensuite généralement l'hypothèse de distributions conditionnelles *gaussiennes*. Sous ces conditions, l'étape d'apprentissage revient à déterminer, pour chaque classe $y \in [1, k]$ les paramètres de la gaussienne (μ_y^i, σ_y^i) décrite par chaque composante descriptive $i \in [1, d]$. Après passage en *log-probabilité*, un facteur $\mathbb{P}(X = x^i | Y = y)$ de l'Equation 1.10, devient un terme et se développe comme un polynôme d'ordre 2 en x^i :

$$p(X = x^i | Y = y) = \frac{-1}{2 \times (\sigma_y^i)^2} (x^i)^2 + \frac{\mu_y^i}{(\sigma_y^i)^2} x^i - \left(\frac{(\mu_y^i)^2}{2 \times (\sigma_y^i)^2} + ln(\sigma_y^i) \right) \tag{1.11}$$

- Enfin, l'hypothèse d'*homoscédasticité* suppose des gaussiennes de même dispersion : $\forall y, \sigma_y^i = \sigma^i$. Nous constatons alors dans l'expression des termes de la *log-vraisemblance* qu'un certain nombre d'éléments sont indépendants de la classe y. Puisqu'il est question d'établir le maximum selon y sans rechercher une valeur exacte des probabilités, il n'est pas nécessaire de conserver ces éléments. L'expression d'un terme de la *log-vraisemblance* peut alors s'écrire comme l'équation d'une droite fonction de x^i :

$$p(X = x^i | Y = y) = \frac{\mu_y^i}{(\sigma^i)^2} x^i - \left(\frac{(\mu_y^i)^2}{2 \times (\sigma^i)^2} \right) \tag{1.12}$$

Ainsi, la classification bayésienne naïve formule un certain nombre d'hypothèses simplificatrices pour trouver des frontières linéaires au problème de classification énoncé dans l'Equation 1.9.

Classement linéaire par machines à vecteur de support Fondés sur les travaux de VAPNIK [1963], les *SVMs*, pour *Support Vector Machines*, sont également des techniques de calcul d'hyperplans séparateurs de données. Ils introduisent une notion géométrique de *marge* autour des frontières entre les classes qui s'avère d'un intérêt crucial pour la détermination d'hyperplans à *capacité de généralisation maximale*.

La marge de la frontière est définie comme la plus petite distance entre un individu $x \in X_t$ et l'hyperplan séparateur et peut être calculée ainsi :

$$m = \min_{x \in X_t} \frac{\langle w, x \rangle + b}{\|w\|} \tag{1.13}$$

La plupart du temps, et cela est d'autant plus vrai sur de faibles jeux de données, un grand nombre d'hyperplans séparateurs ayant la même performance de classification existent. Les travaux de Vapnik montrent que l'unique hyperplan optimal, c'est-à-dire le moins propice aux erreurs en période de test, est celui qui maximise la marge. La méthode consiste donc à déterminer l'hyperplan affine (w, b) qui maximise la marge, tout en garantissant une bonne classification des individus.

L'hyperplan de marge maximale peut encore être paramétré par une infinité de couples (w, b) et dans le but de simplifier le problème d'optimisation de marge et d'en faciliter l'interprétation géométrique, le paramétrage retenu pour l'hyperplan séparateur optimal est celui tel que pour tout individu x_m situé exactement sur la marge autour de la frontière, on ait : $|\langle w, x_m \rangle + b| = 1$. En introduisant ce paramétrage dans l'Equation 1.13, le problème d'optimisation résolu par la méthode peut s'écrire de la manière suivante :

$$\begin{cases} \min \frac{\|w\|^2}{2} \\ \\ \text{sous contrainte : } \forall x \in X_t, f^*(x) = f(x) \quad \text{(cf.Equation 1.8)} \end{cases} \tag{1.14}$$

L'énoncé ci-dessus est celui d'un problème d'optimisation quadratique traditionnel pour lequel une large variété d'algorithmes de résolution existe.

La théorie des *SVMs* propose également une extension de la classification aux jeux de données qui ne peuvent pas être séparés linéairement. L'*astuce du noyau*, pour *kernel trick*, applique une transformation T, non-linéaire, aux descripteurs afin de placer l'ensemble des données dans un espace linéairement séparable $Im(T)$ avant le calcul de l'hyperplan. On remarque alors dans ce cas qu'au cours de l'optimisation, seuls des produits scalaires des vecteurs de $Im(T)$ sont calculés $\langle T(x_i), T(x_j) \rangle$ et il n'est donc pas nécessaire de définir la transformation T, mais seulement une fonction symétrique semi-définie positive notée $K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ que l'on appelle *noyau*. La définition d'un noyau permet ainsi de définir indirectement une transformation inconnue des données vers un espace que l'on espère linéairement séparable et étend ainsi l'application du *SVM* aux problèmes de classification non-linéaire.

Classement linéaire par réseaux de neurones Ce sont les travaux de Rosenblatt [1958] qui introduisent initialement le neurone formel de McCulloch and Pitts [1943] comme une forme de solution au problème du classement supervisé. Le système proposé est alors constitué d'un unique neurone doté d'un cône d'émergence, appelé *sortie* du neurone, et d'autant de dendrites, ou *entrées*, que les individus à classer comptent de composantes descriptives. Le neurone individuel ainsi décrit se comporte comme une petite unité de calcul autonome.

À chaque dendrite est attribué un poids synaptique w^i et lorsqu'un individu à classer x est connecté à l'entrée du neurone, sa position relative par rapport à l'hyperplan dendritique, $\langle w, x\rangle$ est calculée. Afin de définir un hyperplan affine, comme dans le cas de la classification linéaire générale, une dendrite supplémentaire de poids synaptique b est toujours connectée à une entrée égale à 1. Ainsi, la sortie du neurone peut être calculée de la manière suivante :

$$f(x) = \sigma(\langle w, x\rangle + b)$$

avec σ, une fonction non-linéaire appelée *fonction d'activation* du neurone. Dans le cas du neurone unique, destiné au classement linéaire, l'expression et les propriétés, telles que la dérivabilité de la fonction d'activation servent principalement à faciliter le calcul de l'erreur et à rendre l'apprentissage possible.

Lors de l'apprentissage, les poids sont initialisés au hasard et sont itérativement mis à jour à chaque observation d'un nouvel individu de l'ensemble d'apprentissage. La mise à jour d'un poids repose sur l'application d'une règle de renforcement proposée par Hebb [1961], qui stipule que deux neurones joints dont les activités sont corrélées tendent à réagir métaboliquement en faveur d'une corrélation encore meilleure. L'algorihme adopté par Rosenblatt [1958], plus adapté à l'apprentissage supervisé, utilise l'erreur commise par le neurone, $y^* - y$, pour pondérer la mise à jour d'une synapse :

$$w^i \leftarrow w^i + (y^* - y)x^i$$

Lors de l'observation d'un individu x, le poids synaptique de la i-ème composante descriptive sera d'autant plus modifié que l'erreur commise $y^* - y$ est grande et que l'intensité x^i perçue en entrée est importante.

Le modèle du perceptron, dans sa version dite *multicouche*, est constitué d'un ensemble de neurones connectés en réseau et permet d'étendre le modèle de classification à des frontières arbitrairement complexes. L'organisation en *couches* de neurones et les algorithmes d'optimisation des poids synaptiques les plus élaborés sont à la base du vocabulaire de l'apprentissage profond que nous nous apprêtons à décrire plus en détails dans le chapitre qui suit.

Arbres de décision Parmi les outils de classement supervisé, les arbres décisionnels sont les méthodes les plus propices à l'interprétation humaine. Les systèmes experts par exemple, programment, « en dur », la suite de décisions qui mène l'expert à sa conclusion. Les arbres sont ainsi construits comme des chaînes d'instructions conditionnelles opérant sur les concepts de l'expert par *déduction*. Ils sont en cela une matérialisation locale, propre à la résolution d'un problème bien défini, de l'ontologie des connaissances de l'expert. La définition de l'arbre, par la formulation des conditions sur ses nœuds, n'est d'ailleurs pas réservée qu'à une implémentation informatique en vue d'une automatisation. En effet, les arbres décisionnels servent également à fixer des directives ou des standards d'analyse à destination des experts eux-mêmes, comme cela peut être le cas dans le domaine médical lors de l'émergence de nouvelles pratiques comme celles décrites dans Hendry et al. [2018], Borghaei et al. [2015] ou dans Roach et al. [2016].

Il existe également des techniques d'apprentissage qui consistent à construire l'arbre de manière automatique à partir des données d'entraînement. La solution la plus explorée en la matière est l'algorithme de Breiman et al. [1984], appelé algorithme *CART* (*Classification And Regression Trees*), qui procède par découpages dyadiques successifs de l'ensemble d'apprentissage. À un nœud N non terminal de l'arbre, correspond une partie de l'ensemble d'apprentissage $X_t(N)$ et l'application d'un seuil ϵ sur une composante i descriptive des éléments de cette partie permet de séparer cet ensemble en deux parties : l'enfant gauche, N_g, et l'enfant droit, N_d, qui référencent respectivement $X_t(N_g)$ et $X_t(N_d)$ définis comme suit :

$$X_t(N_g) = \{x \mid x \in X_t(N), x_i \leq \epsilon\}$$

$$X_t(N_d) = \{x \mid x \in X_t(N), x_i > \epsilon\}$$

La composante i et le seuil ϵ sont choisis, parmi l'ensemble des coupures et seuils possibles, pour former les enfants les plus homogènes possibles, la plupart du temps au sens de l'indice de Gini Gini [1921]. L'arbre peut ainsi être construit selon une méthode récursive à logique *descendante* sur la base de cette règle et, pour un nœud considéré, les divisions s'arêtent s'il ne contient plus qu'un seul individu ou si l'ensemble des individus qu'il contient sont de même classe.

L'apprentissage avec terminaison sur le critère d'arrêt évoqué ci-dessus tend à produire des arbres larges, c'est-à-dire formant des frontières complexes entre les classes. Cette précision excessive dans la définition des frontières rend ces méthodes particulièrement sensibles aux données erronées ainsi qu'au *sur-apprentissage*. Afin de minimiser cet effet, des procédures pour limiter le nombre de paramètres, telles que l'*élagage* des arbres, permettent de trouver des frontières lisses avec un meilleur pouvoir de généralisation, mais l'approche la plus répandue reste celle de l'*ensemble learning* qui regroupe les méthodes de combinaison des prédictions de plusieurs arbres.

1.4.6 Stratégies d'ensemble

Combiner plusieurs modèles de classification relève des stratégies d'apprentissage ensemblistes, ou *ensemble learning*. La prédiction combinée peut prendre la forme d'une moyenne ou, plus généralement, d'une combinaison linéaire de la prédiction des modèles individuels et permet d'obtenir de meilleurs résultats de classement à condition que les modèles de classement respectent certaines propriétés.

D'une part, chacun des modèles contribuant à la prédiction doit être porteur d'une information pertinente, c'est-à-dire produire un nombre non-négligeable de bonnes classifications. D'autre part, la constitution des différents modèles individuels doit assurer une certaine *diversité de l'ensemble* sans laquelle toute complémentarité serait impossible. Intuitivement, la combinaison de deux modèles dont les prédictions sont similaires ne saurait apporter un bénéfice réel à la tâche de classement considérée.

Un cadre plus formelle est proposé par Krogh and Vedelsby [1995], qui font intervenir explicitement la notion de diversité dans la formulation de l'erreur commise par un ensemble de classifieurs. Soit $\mathcal{F} = \{f_1, \ldots, f_T\}$ un ensemble de modèles, on confondra notamment l'ensemble $\mathcal{F}$ avec sa fonction de prédiction associée pouvant prendre la forme suivante :

$$\mathcal{F}(x) = \sum_{i=1}^{T} w_i f_i(x) \tag{1.15}$$

Chacun des membres de l'ensemble sont alors susceptibles de proposer une prédiction *ambiguë* vis-à-vis de la prédiction d'ensemble. Krogh and Vedelsby [1995] définissent ainsi l'*ambiguïté* d'un classifieur en mesurant l'écart entre sa prédiction et celle de l'ensemble et l'*ambiguïté*

globale de l'ensemble comme la moyenne des *ambiguïtés* individuelles. On parle aussi de *dispersion* de l'ensemble puisque l'*ambiguïté* globale s'exprime fondamentalement comme une moyenne pondérée des écarts à la prédiction d'ensemble :

$$\mathcal{A}mbi(f_i|x) = (f_i(x) - \mathcal{F}(x))^2 \tag{1.16}$$

$$\mathcal{A}mbi(\mathcal{F}|x) = \sum_{i=1}^{T} w_i \mathcal{A}mbi(f_i|x) \tag{1.17}$$

Il est ensuite possible de relier la performance de classification de l'ensemble à son degré d'ambiguïté. Si l'on choisit par exemple d'exprimier l'erreur de classification comme le carré de la distance euclidienne, on peut écire les relations suivantes :

$$\mathcal{E}rr(f_i|x) = (y^* - f_i(x))^2 \tag{1.18}$$

$$\mathcal{E}rr(\mathcal{F}|x) = (y^* - \mathcal{F}(x))^2 \tag{1.19}$$

$$\overline{\mathcal{E}rr}(\mathcal{F}|x) = \sum_{i=1}^{T} w_i \mathcal{E}rr(f_i|x) \tag{1.20}$$

Il vient alors simplement,

$$\mathcal{E}rr(\mathcal{F}|x) = \overline{\mathcal{E}rr}(\mathcal{F}|x) - \mathcal{A}mbi(\mathcal{F}|x) \tag{1.21}$$

On note alors bien qu'entre deux ensembles $\mathcal{F}_1$ et $\mathcal{F}_2$, dont les performances de classification individuelles sont comparables $\overline{\mathcal{E}rr}_1 = \overline{\mathcal{E}rr}_2$, la meilleur performance sera celle de l'ensemble le plus discordant.

Le Bagging Dans le domaine de l'apprentissage supervisé, le *bagging*, pour *Bootstrap Aggregating*, est une méthode pour la construction efficace d'un ensemble de classifieurs. Mise au point par Breiman [1996], la technique promeut la diversité d'un ensemble de modèles prédictifs par apprentissage sur des jeux de données différents. Chaque classifieur f_i de l'ensemble $\mathcal{F}$ apprend sa règle de classification sur d'un sous-ensemble X_{ti} de l'ensemble d'apprentissage X_t obtenu par une procédure de tirage aléatoire avec remise.

Les forêts aléatoires L'algorithme des forêts aléatoires de Breiman [2001], applique la stratégie du bagging aux ensembles d'arbres décisionnels. Chaque arbre de l'ensemble est donc entraîné à séparer les classes sur la base d'une partie différente de l'ensemble d'apprentissage. Néanmoins, dans le cas des arbres, la stratégie de bagging classique est souvent insuffisante à fournir la diversité requise pour garantir une bonne combinaison des classifieurs. En effet, lorsqu'une composante ou un petit nombre de composantes sont très informatives quant au problème à résoudre, les arbres créés vont naturellement tous faire usage de ces mêmes caractéristiques pour produire leur décision. Les prédictions des arbres individuels seront donc fortement corrélées et ne pourront satisfaire la diversité nécessaire à une bonne prédiction d'ensemble.

Afin de pallier à ce problème, l'algorithme de Breiman [2001] propose un bagging de caractéristiques : chaque arbre de l'ensemble n'a accès qu'à une partie, tirée aléatoirement, des composantes descriptives des individus. La méthode garantit ainsi la diversité de l'ensemble en forçant les arbres à s'appuyer sur des caractéristiques différentes pour prendre leur décision.

Le Boosting Une autre méthode de construction astucieuse d'ensembles de classifieurs est le *boosting* qui propose une construction séquentielle de l'ensemble. À chaque étape $i+1$ de la méthode, un classifieur f_{i+1} dit *faible*, c'est-à-dire pour lequel la seule contrainte est d'avoir des performances de classification supérieures au hasard, construit sa frontière de classification sur l'ensemble d'apprentissage X_t. Le modèle entraîné est ensuite ajouté à l'ensemble et participe désormais à la décision globale qui accorde un poids w_{i+1} à sa prédiction, proportionnel à son taux de réussite sur l'ensemble d'apprentissage. Enfin, les éléments de X_t sur lesquels f_{i+1} commet une erreur de classement sont pondérés et prendront ainsi une part plus importante dans le dessin de la frontière du classifieur f_{i+2} dans l'étape suivante.

1.4.7 Evaluation des partitions supervisées

L'évaluation du classement supervisé est bien plus aisée et plus objective que celle du clustering sans supervision. indépendamment, et après apprentissage sur l'ensemble X_t, l'algorithme prédit le partitionnement $\mathcal{C} = \{C_1, \ldots, C_k\}$ d'un ensemble de données de validation X_v, pour lequel la classification $\mathcal{C}^* = \{C_1^*, \ldots, C_k^*\}$, réalisée par l'expert, est connue. Toute prédiction qui n'est pas en adéquation avec celle de l'expert, $x \in X_v, f(x) \neq f^*(x)$ est une erreur commise par l'algorithme et le taux de ces erreurs, évalué globalement ou détaillé pour chaque classe, permet d'évaluer les performances de l'algorithme.

- Pour une classe i donnée, on appelle *vrais positifs* et on note TP_i, les éléments appartenant à cette classe correctement prédits par l'algorithme de classement,

$$TP_i = C_i^* \cap C_i$$

- Pour une classe i donnée, on appelle *faux positifs* et on note FP_i, les éléments n'appartenant pas à cette classe pourtant prédits comme des membres de cette classe,

$$FP_i = \bigcup_{j \neq i} C_j^* \cap C_i$$

- La *précision* est défini comme un taux de réussite. Pour une classe i donnée, il compare le nombre de bonnes prédictions au nombre total de prédictions pour cette classe :

$$précision_i = \frac{|C_i \cap C_i^*|}{|C_i|}$$

- Le *rappel* se rapporte plus à une métrique de détection. Pour une classe i donnée, il compare le nombre de bonnes prédictions au nombre de prédiction attendues pour cette classe :

$$rappel_i = \frac{|C_i \cap C_i^*|}{|C_i^*|}$$

- Le *F-score* intègre dans une valeur unique la précision et le rappel pour fournir un indice unique de bonne performance :

$$Fscore_i = \frac{précision_i \times rappel_i}{précision_i + rappel_i}$$

- Pour une classe i donnée, l'algorithme de classification produit généralement une probabilité, ou du moins un score d'appartenance des individus de $x \in X_v$ à la classe considérée. Il est alors possible de choisir une valeur-seuil de probabilité au-dessus de laquelle les individus seront automatiquement placés dans la classe i. Pour un cluster C_i, la courbe *ROC* (*Receiver Operating Characteristic*) trace l'évolution du taux TP_i en fonction de FP_i pour des valeurs de seuil décroissantes, c'est-à-dire de plus en plus

tolérantes. Les deux taux augmentent naturellement avec la décroissance du seuil de probabilité de prédiction, mais un classifieur de bonne qualité doit maximiser son taux de vrais positifs pour un faible nombre de faux positifs. En plus de l'allure des courbes qui indiquent la performance du modèle, ces courbes permettent également d'aider l'expert à fixer un seuil de prédiction selon les risque qu'il attribue aux fausses alarmes et aux détections râtées de l'évènement i.

1.5 Conclusion

L'analyse des images regroupe l'ensemble des techniques d'extraction de connaissances utiles à la prise de décision sur la base d'une image. Elle synthétise un processus de perception en relevant les indices de la présence d'objets ou de phénomènes nommés par l'expert. Au cours de ce chapitre, les mécanismes à l'œuvre dans l'analyse automatique des images numériques ont été abordés. Pour chacun d'eux, nous avons expliqué les cheminements qui traduisent ces concepts d'analyse en problèmes d'optimisations, ainsi que les principales pistes explorées pour une résolution algorithmique de ces problèmes. Lorsque cela était possible, nous avons mis en parallèle la structure des connaissances humaines, mais aussi les raisonnements qui conditionnent une décision à l'observation d'une image, avec la structure des résultats des algorithmes et les méthodes statistiques qui les produisent.

De ce développement se dégage également une démarcation profonde entre les notions de description et de prédiction. Les deux procédés, pourtant conceptuellement très proches et naturellement imbriqués pour produire les résultats de l'analyse, sont bien souvent traîtés indépendamment et, de cette absence de coordination entre les deux procédures, naît la limitation principale des approches classiques évoquées tout au long de ce premier chapitre. L'abolition de cette frontière signe l'avènement du connexionisme et c'est sur ce point précis que les réseaux neuronaux convolutifs profonds (*Deep Learning*) font une avancée majeure dans le domaine de l'analyse d'image.

Le chapitre suivant décrit le fonctionnement des méthodes de l'apprentissage profond à la lumière des concepts précédemment évoqués, sans toutefois oublier de souligner les particularités qui font leur superiorité. Il pose également le cadre applicatif de ces travaux de thèse en exposant les contraintes d'adaptation de ces algorithmes aussi bien sur le plan technique, en soulignant la spécificité des images de coupes histologiques, et tout particulièrement de la structure des lames entières numérisées, que sur le plan de leur implantation « utile » dans des applications biomédicales.

1.6 Références

David Mumford and Jayant Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on pure and applied mathematics*, 42 (5) :577–685, 1989. 7

Ron Ohlander, Keith Price, and D Raj Reddy. Picture segmentation using a recursive region splitting method. *Computer graphics and image processing*, 8(3) :313–333, 1978. 8

Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, (6) :721–741, 1984. 9

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net : Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 9

Xuming He, Richard S Zemel, and Miguel Á Carreira-Perpiñán. Multiscale conditional random fields for image labeling. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 2, pages II–II. IEEE, 2004. 9

Danilo Jimenez Rezende, Shakir Mohamed, Ivo Danihelka, Karol Gregor, and Daan Wierstra. One-shot generalization in deep generative models. *arXiv preprint arXiv :1603.05106*, 2016. 9

Luminita A Vese and Tony F Chan. A multiphase level set framework for image segmentation using the mumford and shah model. *International journal of computer vision*, 50(3) : 271–293, 2002. 10

Vicent Caselles, Ron Kimmel, and Guillermo Sapiro. Geodesic active contours. *International journal of computer vision*, 22(1) :61–79, 1997. 10

Martin C Cooper. Formal hierarchical object models for fast template matching. *The computer journal*, 32(4) :351–361, 1989. 10

Xavier Descombes and Josiane Zerubia. Marked point process in image analysis. *IEEE Signal Processing Magazine*, 19(5) :77–84, 2002. 10

Giovanni Poggi and RP Ragozini. Image segmentation by tree-structured markov random fields. *IEEE Signal Processing Letters*, 6(7) :155–157, 1999. 11

Georges Koepfler, Christian Lopez, and Jean-Michel Morel. A multiscale algorithm for image segmentation by variational method. *SIAM journal on numerical analysis*, 31(1) :282–299, 1994. 11

Coloma Ballester, Vicent Caselles, and M Gnzález. Affine invariant segmentation by variational method. *SIAM Journal on Applied Mathematics*, 56(1) :294–325, 1996. 11

Frank Fuchs. *Contribution à la reconstruction du bâti en milieu urbain, à l'aide d'images aériennes stéréoscopiques à grande échelle : étude d'une approche structurelle*. PhD thesis, 2001. 11

Laurent Guigues. Modèles multi-échelles pour la segmentation d'images. *École Doctorale Sciences et Ingénierie de l'Université de Cergy-Pontoise*, 302, 2003. 11

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning : A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8) : 1798–1828, 2013. 12, 19

Thomas R Gruber. Toward principles for the design of ontologies used for knowledge sharing ? *International journal of human-computer studies*, 43(5-6) :907–928, 1995. 12

Gunnar Declerck and Jean Charlet. Intelligence artificielle, ontologies et connaissances en médecine. les limites de la mécanisation de la pensée. *Revue des Sciences et Technologies de l'Information-Série RIA : Revue d'Intelligence Artificielle*, 25(4) :445–472, 2011. 12

David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999. 15

Christopher G Harris, Mike Stephens, et al. A combined corner and edge detector. In *Alvey vision conference*, volume 15, pages 10–5244. Citeseer, 1988. 15

Robert M Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5) :786–804, 1979. 15

Timo Ojala, Matti Pietikainen, and David Harwood. Performance evaluation of texture measures with classification based on kullback discrimination of distributions. In *Proceedings of 12th International Conference on Pattern Recognition*, volume 1, pages 582–585. IEEE, 1994. 16

Zellig S Harris. Distributional structure. *Word*, 10(2-3) :146–162, 1954. 16

Thomas Leung and Jitendra Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International journal of computer vision*, 43(1) : 29–44, 2001. 16

Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Y Ng. Efficient sparse coding algorithms. In *Advances in neural information processing systems*, pages 801–808, 2007. 17

Jianchao Yang, Kai Yu, Yihong Gong, and Thomas Huang. Linear spatial pyramid matching using sparse coding for image classification. In *2009 IEEE Conference on computer vision and pattern recognition*, pages 1794–1801. IEEE, 2009. 17

Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, Thomas Huang, and Yihong Gong. Locality-constrained linear coding for image classification. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3360–3367. IEEE, 2010. 17

Jeff A Bilmes et al. A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. *International Computer Science Institute*, 4(510) :126, 1998. 17

Florent Perronnin and Christopher Dance. Fisher kernels on visual vocabularies for image categorization. In *2007 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2007. 17

Florent Perronnin, Jorge Sánchez, and Thomas Mensink. Improving the fisher kernel for large-scale image classification. In *European conference on computer vision*, pages 143–156. Springer, 2010. 17

Gabriella Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, pages 1–2. Prague, 2004. 17

Y-Lan Boureau, Francis Bach, Yann LeCun, and Jean Ponce. Learning mid-level features for recognition. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2559–2566. IEEE, 2010a. 17

Y-Lan Boureau, Jean Ponce, and Yann LeCun. A theoretical analysis of feature pooling in visual recognition. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 111–118, 2010b. 17

Emmanuel Blanchard. *Exploitation d'une hiérarchie de subsomption par le biais de mesures sémantiques*. PhD thesis, Université de Nantes, 2008. 18

John R Josephson and Susan G Josephson. *Abductive inference : Computation, philosophy, technology*. Cambridge University Press, 1996. 18

Richard O Duda, Peter E Hart, et al. *Pattern classification and scene analysis*. Wiley, New York, 1973. ISBN 978-0-471-22361-0. Open Library ID : OL5287711M. 19

James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967. 21

Joseph C Dunn. Well-separated clusters and optimal fuzzy partitions. *Journal of cybernetics*, 4(1) :95–104, 1974. 21, 23

Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996. 22

Teuvo Kohonen. Self-organized formation of topologically correct feature maps. *Biological cybernetics*, 43(1) :59–69, 1982. 22

David L Davies and Donald W Bouldin. A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence*, (2) :224–227, 1979. 23

Cédric Wemmert. *Classification hybride distribuée par collaboration de méthodes non supervisées*. PhD thesis, Strasbourg 1, 2000. 23

Peter J Rousseeuw and L Kaufman. Finding groups in data. *Hoboken : Wiley Online Library*, 1990. 23

Jon Louis Bentley. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9) :509–517, 1975. 24

James McNames. A fast nearest-neighbor algorithm based on a principal axis search tree. *IEEE Transactions on pattern analysis and machine intelligence*, 23(9) :964–976, 2001. 24

V. VAPNIK. Pattern recognition using generalized portrait method. *Automation and Remote Control*, 24 :774–780, 1963. URL https://ci.nii.ac.jp/naid/10020952249/en/. 25

Frank Rosenblatt. The perceptron : a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6) :386, 1958. 26

Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4) :115–133, 1943. 26

DO Hebb. Brain mechanisms and learning. *Distinctive features of learning in the higher animal*, pages 37–46, 1961. 26

Shona Hendry, David J Byrne, Gavin M Wright, Richard J Young, Sue Sturrock, Wendy A Cooper, and Stephen B Fox. Comparison of four pd-l1 immunohistochemical assays in lung cancer. *Journal of Thoracic Oncology*, 13(3) :367–376, 2018. 26

Hossein Borghaei, Luis Paz-Ares, Leora Horn, David R Spigel, Martin Steins, Neal E Ready, Laura Q Chow, Everett E Vokes, Enriqueta Felip, Esther Holgado, et al. Nivolumab versus docetaxel in advanced nonsquamous non–small-cell lung cancer. *New England Journal of Medicine*, 373(17) :1627–1639, 2015. 26

Charlotte Roach, Nancy Zhang, Ellie Corigliano, Malinka Jansson, Grant Toland, Gary Ponto, Marisa Dolled-Filhart, Kenneth Emancipator, Dave Stanforth, and Karina Kulangara. Development of a companion diagnostic pd-l1 immunohistochemistry assay for pembrolizumab therapy in non–small-cell lung cancer. *Applied Immunohistochemistry & Molecular Morphology*, 24(6) :392, 2016. 26

Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and regression trees*. CRC press, 1984. 27

Corrado Gini. Measurement of inequality of incomes. *The Economic Journal*, 31(121) : 124–126, 1921. 27

Anders Krogh and Jesper Vedelsby. Neural network ensembles, cross validation, and active learning. In *Advances in neural information processing systems*, pages 231–238, 1995. 27

Leo Breiman. Bagging predictors. *Machine learning*, 24(2) :123–140, 1996. 28

Leo Breiman. Random forests. *Machine learning*, 45(1) :5–32, 2001. 28

Chapitre 2

Réseaux neuronaux convolutifs profonds pour l'analyse des lames histologiques

Never send a human
to do a machine's job.

Agent Smith

Sommaire

2.1 Introduction

Le chapitre précédent présentait les composantes constitutives d'une solution algorithmique d'analyse d'images. Il construisait, en suivant notamment la démarche générale des *sacs de mots*, une véritable chaîne perceptuelle en appliquant successivement des processus d'extraction de caractéristiques, suivis de techniques de clustering, dans le but de décomposer l'image en concepts intelligibles par l'humain. Néanmoins, face aux difficultés de décomposition des concepts humains en pixels, ainsi que dans celle de cerner les mécanismes de la perception humaine, les éléments de la chaîne font l'objet d'une large variété d'outils et de fondements théoriques, non seulement pensés, mais aussi implémentés indépendamment les uns des autres.

C'est à ce manque d'interaction entre les maillons de la chaîne perceptuelle que tente de remédier les approches *connexionnistes* de l'apprentissage automatique appliqué à l'analyse des images. Les interactions envisagées historiquement au sein de la chaîne perceptuelle sont principalement unidirectionnelles, en cela que la sortie d'un maillon forme les données d'entrée du suivant. Dans le cadre connexionniste, les interactions sont bidirectionnelles, et un maillon n'est plus optimisé, de manière isolée, à obtenir de bonnes propriétés sur ses éléments de sortie, mais influe également sur l'optimisation du maillon qui le précède.

Un classifieur ne s'appuie plus sur des caractéristiques générales observées dans les images pour prendre sa décision, mais participe véritablement au façonnement des descripteurs les plus adaptés à la résolution de son problème de classement. Cette manœuvre ne garantit pas seulement un outil d'analyse performant, mais remplace presque entièrement l'humain dans son travail d'ingénierie des différents blocs de traitement des données. La conception de la solution relève désormais essentiellement de la connaissance des règles pertinentes d'association des opérations de filtrage et de réduction de dimensions, ainsi que de la bonne constitution des ensembles de données destinés à l'apprentissage et à l'évaluation objective du système.

Compte tenu de l'intervention marginale de l'humain dans la construction et l'intégration des caractéristiques, ainsi que du nombre colossal de paramètres incorporés dans le modèle, ce que l'ingénieur et l'expert du domaine applicatif ont gagné en simplicité de conception et en performance de classification, ils l'ont vraisemblablement perdu en interprétabilité des étapes intermédiaires empruntées par le modèle. À cette complexité d'interprétation, s'ajoute encore un grand nombre d'hyper-paramètres devant être ajustés, et dont l'influence sur la bonne résolution du problème demeure très indirecte.

Dans ce contexte, à l'heure où l'utilité applicative ne requiert guère de maîtrise des concepts sous-jacents du design et de l'influence des hyper-paramètres sur le modèle, le travail de mise en place des solutions d'apprentissage profond s'appuie souvent bien plus sur des relevés expérimentaux et des stratégies d'exploration empiriques que sur de véritables socles théoriques. Ce « travail » fastidieux et coûteux de déploiement et d'ajustement des modèles cessera d'ailleurs bientôt de mobiliser des experts du domaine puisque des initiatives telles qu'AutoML [1] systématisent la recherche des modèles optimaux.

L'attention de la communauté de l'apprentissage automatique revient donc se porter sur l'élaboration de nouveaux concepts de construction ou d'optimisation de cette chaîne d'apprentissage profond, afin de pallier aux principaux défauts qui freinent son implémentation industrielle, notamment dans les domaines les plus critiques comme celui de la santé. Ce chapitre s'attache tout particulièrement à l'applicabilité des réseaux neuronaux convolutifs profonds dans le développement d'applications spécialement dédiées à l'analyse des images de l'histopathologie. Nous aborderons d'abord les principes de fonctionnement des réseaux neuronaux convolutifs profonds à la lumière des descriptions fournies dans le premier chapitre (Section 2.2), puis nous décrirons le contexte et les contraintes du domaine de la pathologie numérique (Section 2.3). Les deux dernières sections couvriront deux applications que j'ai été

1. https://cloud.google.com/automl/

amené à développer pour répondre aux attentes du laboratoire d'anatomocythopathologie de l'Institut Universitaire du Cancer de Toulouse, l'une pour l'analyse de marquages d'immunohistochimie complexes (Section 2.4, Abreu et al. [2019]), l'autre pour l'aide au diagnostic des lymphomes folliculaires (Section 2.5, Syrykh et al. [2020]), chacune ayant fait l'objet d'une publication dans une conférence ou une revue à comité de lecture.

2.2 Réseaux neuronaux convolutifs et apprentissage profond

Les réseaux neuronaux convolutifs profonds sont parfaitement décrits comme un cas particulier des chaînes perceptuelles présentées dans le chapitre précédent. En utilisant le jargon des *sacs de mots*, nous notons que les maillons de la chaîne, ici construits comme des *couches* de neurones, sont tantôt descriptifs, pour **traduire** une représentation de l'image selon un nouveau **vocabulaire**, tantôt **agrégatifs**, pour résumer l'information spatiale de la représentation. Cette section décrit la construction paramétrique des différents types de couches. Elle précise également comment les liens *symboliques* entre les couches permettent d'optimiser le modèle en respectant les relations d'interdépendance entre les paramètres. Des observations architecturales élémentaires, notamment sur les réseaux les plus classiques, je propose ici de créer une véritable *grammaire formelle* pour la construction efficace de classifieurs et d'outils de segmentation d'images. Bien que la version développée au cours de ce travail ne couvre pas la totalité des concepts architecturaux, elle permet néanmoins de couvrir la totalité des cas d'usages pour fixer des performances de base sur un jeu de données d'images.

2.2.1 Extraction de caractéristiques, vocabulaire et traduction

2.2.1.1 Inspiration biologique, filtrage linéaire et activation

Bon nombre de méthodes visant à extraire des caractéristiques chiffrées pour un pixel ou un segment d'image ont déjà été décrites dans la Sous-section 1.3.3 du premier chapitre. Dans le cas des réseaux neuronaux convolutifs, la méthodologie de description d'un pixel ou d'une région d'image est *bio-inspirée*. Les réseaux neuronaux convolutifs profonds doivent ainsi leur inspiration aux résultats de Hubel and Wiesel [1962], portant sur l'étude du cortex visuel du chat. Ces travaux ont montré que des cellules dites « simples » de ce cortex sont activées en réponse à des stimuli visuels particuliers. Une cellule donnée est ainsi sensible à une orientation, une épaisseur et une position de stimuli bien spécifiques sur la rétine.

Des travaux menés par la suite, Marĉelja [1980] et Daugman [1985], modélisent fidèlement la réponse des cellules « simples » par la famille des filtres linéaires de Gabor. De manière simplifiée, un filtre de Gabor défini en 2 dimensions présente une frange lumineuse (motif rectiligne) répétée périodiquement, avec atténuation, dans la direction orthogonale à la frange. Plus précisément, dans sa formulation continue, l'intensité en tout point de l'espace est donnée par :

$$g(x,y) = exp(-\frac{x^2+\gamma^2y^2}{2\sigma^2})cos(2\pi\frac{x}{\lambda}+\phi) \tag{2.1}$$

avec le paramétrage suivant :

- La longueur d'onde λ de la sinusoïde de répétition du motif.
- L'angle θ de la direction (normale à la frange) de répétition du motif.
- La phase ϕ de la sinusoïde de répétition.
- L'écart-type σ de la fonction (gaussienne) d'atténuation des oscillations.

- Un paramètre γ de régulation de la longueur de la frange.

Par la suite, Turner [1986], Mallat [1989], Bovik et al. [1990], Olshausen and Field [1996], Serre et al. [2007] ont notamment beaucoup utilisé et conçu certains de ces filtres afin de définir des bases de décomposition pour caractériser des textures et certains des systèmes développés alors présentaient des architectures extrêmement proches des réseaux neuronaux convolutifs profonds actuels, tels que l'étude menée par Serre et al. [2007]. Les filtres linéaires de Gabor ont néanmoins été remplacés par des filtres quelconques dont la similarité avec l'Equation 2.1 n'est désormais plus constatée qu'après apprentissage statistique des paramètres des filtres. Il est pourtant intéressant de noter un certain regain d'intérêt pour la définition des filtres de Gabor dans certaines architectures récentes, comme cela est le cas pour Luan et al. [2018] ou Alekseev and Bobe [2019], qui permet notamment d'aboutir à des modèles plus aptes à généraliser et comportant moins de paramètres que les architectures classiques.

Nous modélisons ici une cellule simple du cortex visuel primaire par une matrice carrée $\boldsymbol{w}$, de taille $t \times t \times (3)$ (la dernière dimension est pour le nombre de canaux, 3 pour la synthèse additive du rouge, du vert et du bleu) et dont les coefficients, encore appelés *poids synaptiques* de la cellule, sont fixés en discrétisant spatialement l'Equation 2.1. Sur une image numérique $\boldsymbol{I}$, interprétée ici comme une version discrète de celle produite sur la rétine, les *dendrites* câblent physiquement la cellule $\boldsymbol{w}$ à une fenêtre $\boldsymbol{i}$, centrée sur un pixel $p \in \boldsymbol{I}$ et de taille $t \times t \times (3)$. La réponse de la cellule $\boldsymbol{w}$ au stimulus $\boldsymbol{i}$ est d'autant plus grande que le signal capté sur $\boldsymbol{i}$ via les *dendrites* ressemble au motif dessiné par $\boldsymbol{w}$ et cette mesure de ressemblance est mathématiquement fournie par le produit scalaire du stimulus par les *poids synaptiques* : $\langle \boldsymbol{i}, \boldsymbol{w} \rangle$.

Dans l'objectif d'une modélisation correcte sur le plan biologique, mais aussi dans le but de rapprocher le fonctionnement des ordinateurs de celui d'un cerveau, le neurone formel de McCulloch and Pitts [1943], encore utilisé à ce jour comme base pour l'implémentation des réseaux neuronaux les plus profonds, a un mode de fonctionnement binaire. Au repos, la sortie du neurone, aussi appelée *axone*, donne une valeur nulle, mais lorsque la ressemblance du stimulus avec le motif attendu par le neurone est suffisante, c'est-à-dire supérieure à une valeur-seuil, un *potentiel d'action* est déclenché et l'axone prend la valeur 1.

La valeur-seuil, aussi nommée *biais* et notée b, est propre au neurone. Dans son expression la plus simple, la sortie y de la cellule $\boldsymbol{w}$ peut-être formulée comme une instruction conditionnelle, ou *fonction de Heaviside* :

$$\langle \boldsymbol{i}, \boldsymbol{w} \rangle + b \begin{cases} \geq 0 & \Rightarrow y = 0 \\ < 0 & \Rightarrow y = 1 \end{cases} \tag{2.2}$$

Bien que totalement inadapté aux procédures d'apprentissage statistique par descente de gradient, qui requièrent des fonctions dont la dérivée est calculable et non-nulle, l'*échelon de Heaviside* imite néanmoins le comportement non-linéaire des *potentiels d'action*. Cybenko [1989] montre notamment que cette propriété s'avère cruciale pour l'efficacité des réseaux neuronaux artificiels puisque sans elle, toute interconnexion de neurones serait inutile car atteignable par un neurone, ou du moins une couche de neurones, unique.

Afin de s'en convaincre, plaçons-nous par exemple dans le cas de neurones purement linéaires. Pour une cellule dont les entrées sont connectées aux *axones* d'autres cellules d'une couche précédente, la sortie globale du réseau ainsi formé est calculée comme une composition d'applications linéaires qui ne peut être elle-même qu'une application linéaire. La sortie s'écrirait donc sous la forme $y = \langle \boldsymbol{i}, \boldsymbol{w} \rangle + b$ et serait aisément modélisable avec un neurone unique.

Parmi les premiers choix de fonctions d'activation compatibles avec l'apprentissage par descente de gradient, ce sont des versions lisses de l'*échelon de Heaviside* qui ont été proposées telles que la fonction logistique ou la tangente hyperbolique pour Rumelhart et al. [1986]. Les fonctions d'activation actuelles, telle que l'unité de rectification linéaire, *ReLU* utilisée par Nair and Hinton [2010], disposent généralement d'une plage d'activation non-bornée, leur permettant de pénaliser les poids indépendamment de leur valeur. Les mises à jour impactent ainsi fortement un plus grand nombre de paramètres et l'apprentissage en devient plus rapide. De plus, la fonction *ReLU* propose une interprétation naturelle de l'inactivité d'un neurone, puisque les valeurs négatives sont ramenées à 0 de sorte que la sortie du neurone ne puisse véritablement pas être perçue comme un signal. Le neurone s'est spécialisé dans la détection d'un motif bien particulier et sa sortie n'est simplement pas prise en compte si le motif n'est pas présent dans l'image. Quelle que soit la fonction d'activation choisie, nous la noterons a par la suite et l'on pourra écire la sortie d'un neurone sous la forme :

$$y = a(\langle \boldsymbol{i}, \boldsymbol{w} \rangle + b) \tag{2.3}$$

2.2.1.2 Convolution et cartes caractéristiques

Neurones et convolution La description proposée précédemment n'envisage la détection d'un motif qu'en une position donnée de l'image formée sur la rétine. Il va sans dire qu'un objet discriminé, ou partiellement décrit, par le motif de la cellule $\boldsymbol{w}$ doit pouvoir être détecté en tout point du champ de vision et que le filtre linéaire ne saurait être appliqué qu'en un pixel particulier p de l'image $\boldsymbol{I}$. Cette nécessaire invariance à la translation dans la reconnaissance du motif n'est possible qu'en applicant le filtre linéaire sur tous les pixels de l'image.

On parle ici de filtrage sur *fenêtre glissante*, opération apparentée à la *convolution*, qui justifie l'emploi du qualificatif « convolutif » pour caractériser les architectures de réseaux neuronaux utilisées en analyse d'image. La *convolution* trouve une définition mathématique rigoureuse dans le domaine général du traitement du signal. Dans ce formalisme, une *image filtrée*, $\boldsymbol{F}$, est le résultat du *produit de convolution*, noté « $*$ », entre l'image $\boldsymbol{I}$ et une matrice $\boldsymbol{h}$, de taille $t \times t \times (3)$, appelée *réponse impulsionnelle* du filtre :

$$\boldsymbol{F} = \boldsymbol{I} * \boldsymbol{h} \tag{2.4}$$

À la notation $\boldsymbol{i}$ utilisée précédemment pour définir une fenêtre de taille $t \times t \times (3)$ dans l'image, nous ajoutons l'indexation p pour identifier le pixel de $\boldsymbol{I}$ sur lequel est centrée l'imagette $\boldsymbol{i}_p$. En considérant la taille $n \times n$ de l'image filtrée $\boldsymbol{F}$, comme identique à celle de l'image d'origine, $\boldsymbol{I}$, le produit de convolution $\boldsymbol{I} * \boldsymbol{h}$ peut-être reformulé selon le produit scalaire de la section précédente :

$$\boldsymbol{F}_p = \langle \boldsymbol{i}_p, \boldsymbol{w} \rangle \tag{2.5}$$

où $\boldsymbol{w}$ est ici appelé *masque de convolution* associé au filtre $\boldsymbol{h}$ et est défini par permutation spatiale des coefficients de $\boldsymbol{h}$:

$$\forall i, j \in [1 \dots t],\ \boldsymbol{w}_{i,j} = \boldsymbol{h}_{t-i,t-j}$$

La réponse impulsionnelle $\boldsymbol{h}$ du filtre correspond à la sortie que produirait le filtre sur une imagette $\boldsymbol{i}$ dont seul le pixel central serait lumineux. Cette matrice porte également le nom de *fonction d'étalement du point* et exprime comment le signal acquis pour un pixel de l'image a pu « contaminer » celui de ses voisins. Son interprétation et sa formulation reposent sur la modélisation des systèmes physiques impliqués dans le processus d'acquisition des images. Elle est largement utilisée dans les procédures de restauration d'images, telles que le débruitage ou la déconvolution, pour son rôle central dans la formulation du terme d'attache aux données des problèmes inverses.

L'interprétation de $\boldsymbol{w}$ comme permutation spatiale de la réponse impulsionnelle d'un filtre linéaire nous paraît cependant d'une complexité inadaptée à la description des mécanismes artificiels de la perception, y compris du point de vue de leur héritage biologique. Nous choisissons ici de ne jamais faire référence à $\boldsymbol{h}$, que ce soit pour parler du filtre ou de sa réponse impulsionnelle. Les termes « filtres », mais aussi « noyaux/matrices de convolution », ou encore « neurones », seront employés indistinctement pour faire référence à la matrice $\boldsymbol{w}$.

Cartes caractéristiques Considérons la taille $n \times n$ de l'image $\boldsymbol{I}$. On appelle *image filtrée* par le neurone $\boldsymbol{w}$, ou encore *carte caractéristique* produite par $\boldsymbol{w}$, la matrice $\boldsymbol{Y}$ de taille $n \times n$ dont les pixels sont calculés par stimulation de $\boldsymbol{w}$ avec les pixels de $\boldsymbol{I}$ correspondants. En adaptant les notations précédentes, nous notons $\boldsymbol{i}_p$ la fenêtre de pixels de $\boldsymbol{I}$ de taille $s \times s$ centrée sur le pixel p. Les pixels de la matrice $\boldsymbol{Y}$ peuvent alors être calculés de la manière suivante :

$$\forall p,\ \boldsymbol{Y}_p = a(\langle \boldsymbol{i}_p, \boldsymbol{w} \rangle + b) \tag{2.6}$$

Un stimulus de taille $s \times s$ ne peut rigoureusement pas être produit pour tout pixel de $\boldsymbol{I}$, puisque les pixels situés sur les bords de l'image ne disposent pas du voisinage suffisant. La *carte caractéristique* produite par le neurone $\boldsymbol{w}$ a donc pour véritables dimensions $(n-s+1) \times (n-s+1)$. Afin de simplifier l'interprétation des *cartes caractéristiques*, ainsi que certaines démarches de construction des réseaux neuronaux, des stratégies de bourrage, ou *padding*, sont mises en place pour ajouter des pixels à $\boldsymbol{Y}$ (*post-processing*), ou à $\boldsymbol{I}$ (*pre-processing*), de manière à assurer une invariance de la dimension de l'image par l'opération de filtrage.

Pour simplifier les notations, mais également pour rappeler le caractère linéaire des opérations réalisées au sein du réseau, nous réécrirons l'Equation 2.6 sous la forme d'un produit matriciel. En effet, la répétition du motif $\boldsymbol{w}$ selon une logique doublement circulante par bloc, aussi appelée *matrice de Toeplitz* de $\boldsymbol{w}$ et notée $\boldsymbol{W}$, permet notamment de réécrire l'obtention d'une *carte caractéristique* comme suit :

$$\boldsymbol{Y} = a(\boldsymbol{W}\boldsymbol{I} + b) \tag{2.7}$$

Nous l'avons évoqué dans la Sous-section 1.3.2 du premier chapitre, le pouvoir discriminant d'un filtre linéaire seul n'est généralement pas suffisant pour conclure à la présence d'un objet d'intérêt. Il est d'usage, comme préconisé par la stratégie des *sacs de mots*, d'établir une batterie de descripteurs dont la combinaison garantit, de manière bien plus robuste, l'appartenance d'un pixel à un objet particulier.

Dans un réseau convolutif, une couche de convolution contient donc un ensemble de cellules $\boldsymbol{V} = \{\boldsymbol{W}^1, \ldots, \boldsymbol{W}^k\}$, chacune responsable de la reconnaissance d'un élément texturel particulier. Dans le cadre général de la *chaîne perceptuelle* définie dans le premier chapitre, il est pertinent de relever que cette collection de cellules associée à son ensemble de *cartes caractéristiques*, $\mathcal{T} = \{\boldsymbol{Y}^1, \ldots, \boldsymbol{Y}^k\}$, constitue naturellement une paire **vocabulaire-traducteur**, $(\boldsymbol{V}, \mathcal{T})$. Tout pixel p est assigné de manière *floue* aux k *mots* de vocabulaire de la couche : $\{\boldsymbol{Y}^1_p, \ldots, \boldsymbol{Y}^k_p\}$. On notera également que les cellules de la N-ième couche de convolution, $\boldsymbol{V}_N$, d'un réseau convolutif, ont pour dimensions $s \times s \times k$, où k est le nombre de *mots* du vocabulaire $\boldsymbol{V}_{N-1}$, c'est-à-dire le nombre de neurones dans la couche de convolution précédente.

2.2.2 Caractérisation globale et stratégies d'agrégation

De manière générale, l'extraction de connaissances dans les images correspond à une opération de réduction de dimensions. Lorsqu'il s'agit de classer, de détecter ou de décrire des phénomènes dans les images, la matrice de pixels est « réduite » à un vecteur de probabilité d'appartenance, de présence, ou une liste de descripteurs. Ces différentes représentations ont

en commun la perte de la localisation spatiale précise du phénomène étudié dans l'image. Cela indique notamment qu'une grande partie de la réduction de dimensions s'opère sur l'information spatiale.

L'agrégation spatiale consiste à compiler les descripteurs d'une région de l'image en un descripteur unique pour cette région. Elle a été brièvement abordée dans le premier chapitre Sous-section 1.3.4. Les stratégies de *max pooling* ou d'*average pooling* que nous avions alors présentées sont utilisées de manière rigoureusement identique dans le cadre des réseaux neuronaux convolutifs. Les briques architecturales des réseaux comportent des couches d'agrégation, mais le *pooling* effectué demeure cependant restreint à l'agrégation de fenêtres régulières et aucune autre forme de segment, obtenue via une procédure de segmentation d'image par exemple, ne peut être prise en charge par les couches d'agrégation. Soit $\mathcal{T}_N = \{\boldsymbol{Y}^1, \ldots, \boldsymbol{Y}^k\}$ le tenseur des *cartes caractéristiques* sorti d'une couche de convolution comportant k neurones, $\boldsymbol{V}_N = \{\boldsymbol{W}^1, \ldots, \boldsymbol{W}^k\}$. En notant $n \times n \times k$ les dimensions du tenseur $\mathcal{T}_N$, l'application d'une couche de *pooling* est généralement réalisée avant toute autre étape de filtrage $N + 1$.

La réduction est paramétrée par une taille d'agrégation r, telle que la largeur n des cartes caractéristiques est divisible par r. Les cartes $\boldsymbol{Y}_i$ peuvent ainsi être découpées en blocs de $r \times r$ pixels. Chaque bloc est ensuite réduit à une valeur unique, ou *valeur d'agrégation*, qui peut être la valeur maximale, *max pooling*, ou la valeur moyenne, *average pooling*, rencontrée dans le bloc. Un tenseur agrégé, $\mathcal{T}_{N,r}$, de dimensions $\frac{n}{r} \times \frac{n}{r} \times k$ est alors formé en remplaçant chaque bloc dans chaque carte caractéristique par sa *valeur d'agrégation*. Le cas particulier des réseaux neuronaux convolutifs, notamment la restriction de l'agrégation à des segments aussi réguliers que des fenêtres carrées, permet également d'envisager une autre forme de réduction spatiale.

Une troisième stratégie d'agrégation consiste à utiliser le paramètre r comme *pas de convolution*. La couche d'agrégation est alors implémentée comme une couche de convolution où la sortie des neurones $\boldsymbol{w}_i$ n'est calculée que pour un pixel sur r dans chacune des directions. Un filtrage et une réduction de dimensions spatiales sont alors réalisés simultanément. L'avantage ici est d'apprendre une logique spécifique d'agrégation plutôt que d'appliquer une transformation fixe qui ne peut pas être optimisée. Lorsque le *pas de convolution* vient remplacer l'enchaînement d'une couche de convolution et d'une couche de *pooling*, le temps d'exécution est souvent réduit et cette stratégie est employée avec succès dans un certain nombre d'architectures récentes, telles que He et al. [2016] ou encore Iandola et al. [2016]. Néanmoins, la force des couches de convolution par rapport aux couches d'agrégation, peut parfois également faire leur faiblesse. En effet, les couches d'agrégation, dépourvues de paramètres optimisables, peuvent, dans le cas des architectures les plus profondes, grandement faciliter le passage de l'information durant l'apprentissage en évitant d'ajouter des paramètres et des dépendances supplémentaires entre les variables du modèle comme cela est présenté dans l'étude de Sun et al. [2018].

Des couches d'agrégation spatiale sont ainsi régulièrement appliquées après les couches de convolution et cette alternance se poursuit quasiment systématiquement jusqu'à l'obtention de cartes caractéristiques de taille unitaire, c'est-à-dire ne comportant qu'un seul pixel. Ce jeu de descripteurs, sans localisation spatiale particulière, porte généralement le nom d'*encodage* de l'image et les blocs dans lesquels se succèdent une ou plusieurs couches de convolution et une couche de *pooling* sont appelés *encodeurs*.

Ce mode de construction « standard » des architectures neuronales, hérité de l'architecture de LeCun et al. [1989], m'invite à construire une grammaire formelle pour la production des chaînes de perception basées sur réseaux neuronaux convolutifs profonds. Nous fixons pour cela un jeu de *symboles terminaux* en donnant des références explicites, `Conv2D` et `Pooling`,

aux couches de neurones évoquées précédemment. Les *blocs* et le *réseau*, définis comme des enchaînements de symboles terminaux, feront l'objet de *règles de production*, pour lesquelles nous adopterons les notations suivantes :

- « $\rightarrow$ » débutera la définition d'une règle de production,
- « ; » symbolisera la fin d'une règle de production,
- « | » symbolisera un choix possible entre les caractères de part et d'autre,
- le suffixe « $*$ » symbolisera la répétition du caractère précédent 0 ou plusieurs fois,
- le suffixe « $+$ » symbolisera la répétition du caractère précédent au moins 1 fois,
- le suffixe « ? » symbolisera un caractère optionnel ne pouvant être répété qu'une fois au maximum,
- les parenthèses « () » seront utilisées pour grouper des expressions.

La construction de la chaîne perceptuelle par réseaux neuronaux convolutifs profonds, du moins jusqu'à l'encodage de l'image, s'énonce alors simplement selon les notations précédentes :

$$\texttt{Perception} \rightarrow \texttt{Encodeur}+ \ ;$$
$$\texttt{Encodeur} \rightarrow \texttt{Conv2D} + \ \texttt{Pooling}? \ ;$$

où le bloc `Encodeur` de la grammaire correspond à un élément de la suite $\mathcal{T}_n$. Nous verrons par la suite que ces deux règles doivent souvent être complétées pour constituer des classifieurs ou des outils de segmentation d'image. On notera le caractère optionnel « ? » de la couche de `Pooling`, puisque nous avons remarqué qu'un *pas de convolution* pouvait remplacer cette couche dans certains cas d'usage.

2.2.3 Rattachement aux connaissances humaines

Comme cela était déjà le cas dans le premier chapitre, le rattachement aux connaissances humaines, qui se manifeste par la résolution d'un problème de segmentation ou de classement, fait l'objet d'une attention particulière, bien que, nous allons le voir, leur intégration à la grammaire formelle de construction de la chaîne perceptuelle s'opère naturellement.

2.2.3.1 Classement

Les éléments architecturaux du *classement* par réseaux neuronaux ont, encore une fois, déjà été abordés dans le premier chapitre lors de la description du *perceptron*. La classification est donc réalisée en calculant des séparations linéaires entre les classes dans l'espace des images encodées. Nous nous plaçons dans un espace encodé $\boldsymbol{Y} = \{y_1, \ldots, y_k\}$ de cartes caractéristiques y_i unitaires, c'est-à-dire privées de localisation spatiale, et considérons pour la suite un problème de classification supervisée à m classes.

Les séparations linéaires entre les classes sont encodées par des couches dites *denses* et l'on utilisera d'ailleurs le symbole `Dense` dans notre grammaire formelle de construction pour identifier ces couches. Ces dernières sont constituées de neurones *entièrement connectés* aux axones de la couche précédente. Entendons par là que ces couches n'impliquent pas de notion 1-D de fenêtre glissante, si bien que le stimulus d'un neurone *entièrement connecté* est toujours constitué de l'ensemble des signaux de sortie de la couche qui précède.

Le fonctionnement des neurones de cette couche ne diffère pour autant pas de la modélisation décrite précédemment : les *poids synaptiques* encodent toujours un motif caractéristique et la sortie du neurone s'exprime toujours comme une affinité entre le stimulus proposé et le motif pré-enregistré via le calcul d'un produit scalaire suivi d'une activation non-linéaire.

Pour les opérations de classement d'images, la dernière couche du réseau est bien souvent construite comme une couche *dense* de m neurones, $\boldsymbol{W} = \{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_m\}$, chacun dédié à la reconnaissance du motif d'une classe particulière dans l'espace $\boldsymbol{Y}$ des images encodées.

À cette construction s'ajoute également une forme d'activation particulière propre à la couche, et plus seulement au neurone individuel, chargée de transformer le vecteur $\boldsymbol{z}$ des m sorties du réseau en un vecteur de probabilités. La fonction d'activation la plus répandue pour cela est la fonction *softmax* dont l'expression pour une composante i de sortie du réseau est exprimée comme suit :

$$a(\boldsymbol{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{m} e^{z_j}} \tag{2.8}$$

Elle garantit notamment la sommation unitaire du vecteur, d'où l'interprétation par un vecteur de probabilité, mais gère difficilement les problèmes de classes chevauchantes puisqu'elle tend à former un dirac sur le vecteur de sortie dont seule la composante la plus probable est activée (non nulle).

La construction des classifieurs se dote également de *règles de production* qui viennent compléter notre grammaire formelle d'architecture :

$$\begin{aligned} \texttt{Perception} &\rightarrow \texttt{Encodeur} + \texttt{Perceptron}? \ ; \\ \texttt{Encodeur} &\rightarrow \texttt{Conv2D} + \texttt{Pooling}? \ ; \\ \texttt{Perceptron} &\rightarrow \texttt{Dense} + \ ; \end{aligned}$$

Deux remarques peuvent alors être faites vis-à-vis de cette nouvelle grammaire. Le symbole du `Perceptron` tout d'abord, comporte plus d'une couche `Dense` ce qui lui offre la possibilité d'empiler les couches entièrement connectées. Cette stratégie permet notamment d'étendre les frontières entre les classes à des frontières linéaires par partie et permet une approximation plus fine des séparations entre les classes. Néanmoins, le reste de la chaîne, c'est-à-dire la succession des `Encodeurs` permet généralement d'aboutir à une représentation $\boldsymbol{Y}$ susceptible d'être séparée par une couche `Dense` unique. Les architectures modernes les plus performantes ne sont bien souvent dotées que d'une unique couche `Dense` pour produire les résultats de classification, comme cela est le cas pour l'architecture *Xception* développée par Chollet [2017].

Ensuite, nous notons également le caractère optionnel du symbole `Perceptron` pour la classification. Cela signifie notamment que l'opération de classification peut également être réalisée sans recours aux couches entièrement connectées. On parle dans ce cas de réseaux *entièrement convolutionnels*. Ces architectures fonctionnent de manière très similaire aux classifieurs que nous venons d'aborder, mais remplacent la couche `Dense` par une couche `Conv2D` à m neurones dont les dimensions spatiales d'entrée sont unitaires, $1 \times 1 \times k$, et où k est la dimension du vecteur encodé $\boldsymbol{Y}$.

2.2.3.2 Segmentation

Les réseaux neuronaux convolutifs profonds disposent également de briques architecturales propres à la segmentation sémantique d'images. Si l'opération implique également un encodage de l'image, qui peut être réalisé selon un enchaînement de couches de convolution et de *pooling*, la seconde partie du réseau nécessite la reconstruction d'une sortie de même dimension spatiale que l'image d'origine (masque de segmentation).

Le processus de *décodage* de l'image est pensé symétriquement à celui de l'encodage et alterne ainsi des opérations de convolution et d'extension des dimensions spatiales ou *upsampling*, qui consistent à interpoler le signal entre deux pixels pour augmenter la taille

de l'image selon un facteur d'échelle r. Une certaine version de cette opération porte le nom de *convolution transposée* et présente les mêmes avantages que le *pas de convolution* pour la réduction de dimensions. Elle consiste à intercaler r lignes et r colonnes de pixels nuls entre les pixels de l'image et à appliquer une couche de convolution traditionnelle à cette image transformée. À la différence des formules d'interpolation classiques, celles-ci sont entraînées avec le modèle et garantissent une certaine optimalité de la formule de grandissement appliquée. Radford et al. [2015] et Noh et al. [2015] utilisent d'ailleurs cette stratégie pour la segmentation ou la génération d'images.

En intégrant le symbole `Upsampling`, notre grammaire de construction de réseaux s'élargit encore et couvre désormais les problématiques de segmentation aussi bien que celles de classification :

$$
\begin{aligned}
\texttt{Perception} &\rightarrow \texttt{Encodeur}+\ (\texttt{Décodeur}*\ \mid\ \texttt{Perceptron})\ ;\\
\texttt{Encodeur} &\rightarrow \texttt{Conv2D}+\ \texttt{Pooling}?\ ;\\
\texttt{Décodeur} &\rightarrow \texttt{Upsampling}?\ \texttt{Conv2D}+\ ;\\
\texttt{Perceptron} &\rightarrow \texttt{Dense}+\ ;
\end{aligned}
$$

Qu'elles soient dédiées à des problèmes de segmentation pour le modèle de Ronneberger et al. [2015] ou de classification pour le modèle de Szegedy et al. [2016], les architectures récentes bénéficient également de connexions *skip*, qui consistent à donner au réseau la possibilité de court-circuiter une sous-chaîne de couches, ou d'autoriser une couche à accéder à l'information de sortie d'une couche plus profonde que celle qui la précède. Cette formulation sous forme de grammaire générative, rend difficilement compte de la présence de ce type de blocs dans le réseau, mais elle fournit cependant un ensemble de règles suffisant pour générer des architectures pertinentes sur un grand nombre de problématiques de classement ou de segmentation et suffit souvent, en première approche, à fixer des performances de base sur un jeu de données.

2.2.4 Apprentissage

Bien entendu, les valeurs des *poids synaptiques* des cellules de convolution, ou des neurones entièrement connectés décrits plus haut, ne sont pas déterminées au moment de la construction de l'architecture. C'est une procédure d'apprentissage statistique qui permet au modèle de fixer les valeurs de ces différents paramètres. Certaines méthodes, à l'origine de l'optimisation des modèles connexionnistes, ont été évoquées dans le premier chapitre lorsque nous avons abordé l'optimisation du *perceptron*. Ces méthodes d'apprentissage, étaient essentiellement basées sur des considérations biologiques qui ont conduit Hebb [1961] à énoncer une règle d'apprentissage des paramètres d'une assemblée de neurones par renforcement des liens synaptiques entre des neurones dont les activités sont corrélées. Cette formulation exprime fidèlement les interactions entre les cellules dans les mécanismes d'apprentissage, et tout particulièrement celui de l'apprentissage associatif. Bien qu'elle ait pu être adaptée à certaines tâches de classification supervisée, elle demeure fondamentalement, dans sa forme générale, un principe de clustering sans supervision, qui associe des chemins d'activation similaires à des sorties similaires.

Les travaux de Werbos and John [1974] présentent les premières notions de *dynamic feedback* qui proposent des formules de descente du gradient de l'erreur de classification/régression pour optimiser de manière supervisée des modèles aux paramètres entrelacés. Ce sont finalement les travaux de Parker [1985], de Lecun [1985] puis de Rumelhart et al. [1986] qui définissent les lignes directrices de l'apprentissage supervisé des réseaux neuronaux profonds en introduisant l'algorithme de *rétropropagation* (ou *back-propagation* dans sa version anglaise plus commune) de l'erreur de classification/régression.

Notations

— $\frac{\partial f}{\partial x}$ fera référence à la dérivée partielle de f par rapport à la variable x,

— selon les notations de la chaîne perceptuelle définies précédemment, $\mathcal{T}_n$ fera référence à la n-ième couche, ou du moins bloc, sans distinction du type, `Conv2D` ou `Dense`,

— sous un système d'indexation analogue, $\boldsymbol{w}_n$, $\boldsymbol{b}_n$, $\boldsymbol{z}_n$ dénoteront respectivement les neurones, les valeurs de biais et le vecteur de sortie du bloc $\mathcal{T}_n$ et l'on écrira volontiers $\boldsymbol{z}_n = \mathcal{T}_n(\boldsymbol{z}_{n-1})$ pour résumer les transformations réalisées par la couche N selon l'Equation 2.7,

— $\boldsymbol{w}_n^k$, $\boldsymbol{b}_n^k$ et $\boldsymbol{z}_n^k$ feront référence au k-ième neurone, au k-ième biais et à la k-ième sortie de la n-ième couche du réseau,

— $\boldsymbol{x} = \{x_1, \ldots, x_t\}$ correspondra à l'ensemble des images d'apprentissage,

— $\boldsymbol{y}^* = \{y_1^*, \ldots, y_t^*\}$ correspondra à l'ensemble des vecteurs de sortie attendus pour les éléments de $\boldsymbol{x}$

— $\boldsymbol{z}_N = \{z_{N1}, \ldots, z_{Nt}\}$ correspondra à la prédiction globale (sortie) du réseau à N blocs sur l'ensemble $\boldsymbol{x}$.

2.2.4.1 Descente de gradient

L'idée principale consiste à formuler le problème d'optimisation comme la minimisation d'une différence entre les prédictions $\boldsymbol{z}_N$ du réseau sur l'ensemble d'apprentissage et les valeurs $\boldsymbol{y}^*$ de sortie attendues sur ces images. Cette différence, communément appelée *erreur* et notée $\mathcal{E}$, s'exprime formellement comme une fonction des paramètres, c'est-à-dire des *poids synaptiques*, de chacun des neurones du réseau. Sous réserve de convexité et de dérivabilité de la fonction par rapport aux poids synaptiques, il est théoriquement possible de déterminer un paramétrage du modèle qui minimise l'erreur de prédiction par un algorithme de descente de gradient.

L'erreur $\mathcal{E}(\boldsymbol{w}, \mathcal{D})$ se définit ici comme une fonction de la totalité des paramètres de l'architecture neuronale considérée, $\boldsymbol{w} = \bigcup_n \boldsymbol{w}_n$ et de l'ensemble d'apprentissage $\mathcal{D} = (\boldsymbol{x}, \boldsymbol{y}^*)$. Pour un paramétrage donné, $\boldsymbol{w}$, la logique de la descente de gradient indique que la valeur de la fonction $\mathcal{E}$ décroît plus rapidement dans la direction opposée au gradient de la fonction. Intuitivement, et selon une interprétation « physique » de la dérivée d'une fonction, cette technique propose à qui souhaiterait trouver le minimum d'une fonction, de se diriger vers le sens descendant de la direction présentant la plus forte pente, cette dernière étant fournie par la dérivée de la fonction, dans le cas univarié, ou par son équivalent, le *gradient*, dans le cas multi-varié. Plus ou moins arbitrairement, la stratégie de descente consiste à faire un pas d'amplitude η, ici appelé *taux d'apprentissage*, dans cette direction.

Descente globale Suivant la logique ci-dessus se dégage naturellement un algorithme itératif dit de *descente de gradient* qui consiste à faire un certain nombre de pas dans le sens descendant de la pente la plus abrupte de notre fonction $\mathcal{E}$. En notant $\nabla_{\boldsymbol{w}}\mathcal{E}(\boldsymbol{w}, \mathcal{D})$ le gradient de la fonction $\mathcal{E}$, vecteur des dérivées partielles $\frac{\partial \mathcal{E}}{\partial \boldsymbol{w}_n^k}$, une étape de l'algorithme de descente de gradient peut s'écrire comme suit :

$$\boldsymbol{w} \leftarrow \boldsymbol{w} + \eta \nabla_{\boldsymbol{w}} \mathcal{E}(\boldsymbol{w}, \mathcal{D}) \tag{2.9}$$

Cette formulation a l'avantage de produire l'exact gradient de $\mathcal{E}(\boldsymbol{w}, \mathcal{D})$. Néanmoins, elle n'est quasiment jamais utilisée, car toute mise à jour des paramètres impose une prédiction et un calcul de gradient sur l'ensemble des éléments de l'ensemble d'apprentissage, ce qui peut s'avérer extrêmement coûteux pour de larges jeux de données.

Descente stochastique Une alternative moins coûteuse à l'algorithme de minimisation précédent consiste à pratiquer une mise à jour des poids pour chaque individu $x_i \in \boldsymbol{x}$:

$$\boldsymbol{w} \leftarrow \boldsymbol{w} + \eta \nabla_{\boldsymbol{w}} \mathcal{E}(\boldsymbol{w}, (x_i, y_i)) \tag{2.10}$$

Cependant, cette méthode peut s'avérer sensible aux biais dans les données et suivant l'ordre dans lequel ces dernières sont présentées, le système peut rapidement tomber dans un minimum local de la fonction de coût. À cette échelle, le problème d'optimisation devient un problème d'*apprentissage incrémental* et le réseau encourt le risque, à chaque itération de mise à jour de ses poids, d'écraser l'information pertinente apprise jusqu'alors. On parle dans ce cas du phénomène de *catastrophic forgetting*, mis en évidence par McCloskey and Cohen [1989], qui expriment les difficultés de persistance de la connaissance dans un modèle connexionniste lors d'un apprentissage incrémental. Ces travaux soulignent notamment l'extrême difficulté dans ce cas à trouver un compromis entre *plasticité* du réseau, c'est-à-dire sa capacité à oublier ce qu'il sait pour apprendre une information plus générale, et *stabilité* des connaissances, c'est-à-dire sa capacité à ne pas écraser les principes pertinents déjà acquis.

Descente par paquets La stratégie de descente la plus utilisée, généralement appelée *batch gradient descent*, ou *mini-batch gradient descent*, est un compromis entre les deux solutions extrêmes proposées précédemment. Elle consiste à diviser le jeu d'apprentissage $\mathcal{D}$ en *paquets*, ou *batches*, $d \in \mathcal{P}(\mathcal{D})$, et à appliquer la moyenne des gradients calculée sur le paquet :

$$\boldsymbol{w} \leftarrow \boldsymbol{w} + \eta \nabla_{\boldsymbol{w}} \mathcal{E}(\boldsymbol{w}, d) \tag{2.11}$$

Cette stratégie permet donc d'approcher plus fidèlement, disons de manière moins bruitée, le gradient de la véritable fonction $\mathcal{E}(\boldsymbol{w}, \mathcal{D})$ à minimiser, sans pour autant être obligé de calculer les gradients de tous les échantillons d'apprentissage avant de pouvoir mettre à jour les paramètres du modèle.

Descente et inertie La plupart des algorithmes actuels de mise à jour des poids du réseau diffèrent cependant un peu de l'équation Equation 2.9. La variabilité du signal dans les images, de même que la taille du vocabulaire m devant être couvert par les prédictions du système, produisent souvent des directions de gradient contradictoires qui peuvent considérablement ralentir l'apprentissage, y compris en adoptant une stratégie de descente par paquets. Pour pallier ce problème, la plupart des méthodes récentes, Nesterov [1983], Duchi et al. [2011], Zeiler [2012], Kingma and Ba [2014], Bengio [2015], modifient l'équation de mise à jour de manière à lisser le gradient au cours des différentes itérations d'apprentissage. La mise à jour n'est alors plus seulement réalisée dans la direction préconisée à l'itération t par le paquet d, mais dans une direction hybride qui combine la suggestion actuelle et la ou les directions empruntées précédemment, $t-1, t-2, \ldots$, on parle notamment d'*inertie* du gradient.

Sans entrer dans les détails des implémentations de chacune des méthodes utilisées dans les réseaux actuels, nous précisons succinctement le fonctionnement de l'algorithme *Adaptive Moment Estimation* de Kingma and Ba [2014], abrégé *Adam*, qui exploite les axes principaux d'amélioration de la descente de gradient. Cet algorithme propose, comme plusieurs de ses prédecesseurs, Duchi et al. [2011], Zeiler [2012], d'utiliser un pas de descente, η, différent pour chaque paramètre du modèle. À la différence des autres techniques d'optimisation en revanche, la direction de descente ne dépend pas seulement de l'inertie du gradient cumulée sur quelques itérations, mais sur la valeur prise par ce paramètre dans toutes les étapes de la descente. De plus, le pas de descente η est atténué proportionnellement à la variance du gradient, elle aussi relevée sur toutes les itérations d'apprentissage qui précèdent. Un gradient très dispersé, jugé instable, sera d'autant moins crédité dans la mise à jour du poids correspondant.

2.2.4.2 Algorithme de rétropropagation

La descente de gradient décrite précédemment implique, à chaque étape de mise à jour des poids, le calcul du gradient $\nabla_{\boldsymbol{w}}\mathcal{E}(\boldsymbol{w}, d)$ de la fonction de coût. Afin de simplifier la lecture, nous abandonnerons parfois la notation des chaînes perceptuelles, dans laquelle la transformation opérée par le n-ième bloc de l'architecture est notée $\mathcal{T}_n$, pour la remplacer par $\boldsymbol{z}_n$ qui, par abus de notation, fera également référence au tenseur de sortie de cette même couche. Sous cette convention, la transformation globale $\boldsymbol{z}_N$ appliquée à un tenseur d'entrée $\boldsymbol{x}$ et la fonction d'erreur $\mathcal{E}$ du réseau s'écrirons de la manière suivante :

$$\begin{aligned}\boldsymbol{z}_N(\boldsymbol{w}, \boldsymbol{x}) &= (\mathcal{T}_1 \circ \cdots \circ \mathcal{T}_N)(\boldsymbol{w}, \boldsymbol{x}) \\ \mathcal{E}(\boldsymbol{w}, d) &= \boldsymbol{z}_N - \boldsymbol{y}^*\end{aligned}$$

Le développement du gradient de $\mathcal{E}$, consiste donc à dériver la composée d'une multitude de fonctions. Ce calcul est donné par la formule de dérivation dite de *Leibniz*, qui stipule que pour deux fonctions $\boldsymbol{z}_n = \mathcal{T}_n$ et $\boldsymbol{z}_{n+1} = \mathcal{T}_{n+1}$, telles que $\boldsymbol{z}_n$ est dérivable au point w_n et $\boldsymbol{z}_{n+1}$ est dérivable au point $\boldsymbol{z}_n(w_n)$, la fonction composée $\boldsymbol{z}_{n+1}(\boldsymbol{z}_n) = \boldsymbol{z}_{n+1} \circ \boldsymbol{z}_n$ est dérivable en w_n et que sa dérivée en ce point s'écrit ainsi :

$$\frac{\partial \boldsymbol{z}_{n+1}(\boldsymbol{z}_n)}{\partial w_n} = \frac{\partial \boldsymbol{z}_{n+1}(\boldsymbol{z}_n)}{\partial \boldsymbol{z}_n} \times \frac{\partial \boldsymbol{z}_n(w_n)}{\partial w_n} \tag{2.12}$$

L'algorithme de rétropropagation de l'erreur propose alors une méthode de mise à jour des poids dans les différentes couches en calculant *récursivement* les différentes composantes du gradient de la fonction de coût :

$$\nabla_{\boldsymbol{w}}\mathcal{E}(\boldsymbol{w}, d) = [\frac{\partial \boldsymbol{z}_N}{\partial \boldsymbol{w}_1}, \ldots, \frac{\partial \boldsymbol{z}_N}{\partial \boldsymbol{w}_N}]$$

L'algorithme de rétropropagation calcule les composantes de ce tenseur les unes après les autres en commençant par celle de rang N et en reculant progressivement vers la première couche du réseau, d'où l'appellation *rétropropagation*. Par opposition, la suite de ce développement utilisera le terme *propagation* pour évoquer la stimulation du réseau de neurones par un tenseur d'images $\boldsymbol{x}$ et le relevé successif des activations $\boldsymbol{z}_n$ qui en découlent.

La procédure de *rétropropagation* considère donc la suite inversée des composantes du gradient de l'erreur du réseau. Elle exploite notamment une règle de définition récurrente de cette suite :

1. **Initialisation** Pour un *batch* $d = (\boldsymbol{x}, \boldsymbol{y}^*)$, une *propagation* du tenseur $\boldsymbol{x}$ est réalisée et la sortie $\boldsymbol{z}_n$ des différentes couches est enregistrée. On remarque alors que les paramètres de la dernière couche du réseau ne font pas l'objet d'une composition de fonctions. La N-ième composante du gradient de $\mathcal{E}$ correspondante se calcule donc par une dérivation simple de la fonction $\boldsymbol{z}_n = \mathcal{T}_N$, par rapport aux paramètres $\boldsymbol{w}_N$ de cette même couche :
$$\frac{\partial \mathcal{E}}{\partial \boldsymbol{w}_N} = \frac{\partial \boldsymbol{z}_N}{\partial \boldsymbol{w}_N}$$
2. **Généralisation au rang n** Afin de mieux cerner le fonctionnement de la récursion à l'œuvre dans le calcul des termes de la suite, observons la forme de la composante de rang $N-1$, du gradient de $\mathcal{E}$:
$$\frac{\partial \mathcal{E}}{\partial \boldsymbol{w}_{N-1}} = \frac{\partial \boldsymbol{z}_N(\boldsymbol{z}_{N-1})}{\partial \boldsymbol{w}_{N-1}}$$
Puis, en rappelant que $\boldsymbol{z}_{N-1}$ est une fonction de $\boldsymbol{w}_{N-1}$, l'exacte application de la règle de dérivation de Leibniz (Equation 2.12) conduit à :
$$\frac{\partial \mathcal{E}}{\partial \boldsymbol{w}_{N-1}} = \frac{\partial \boldsymbol{z}_N}{\partial \boldsymbol{z}_{N-1}} \times \frac{\partial \boldsymbol{z}_{N-1}}{\partial \boldsymbol{w}_{N-1}}$$

On conjecture alors rapidement la propriété suivante au rang n :

$$\frac{\partial \mathcal{E}}{\partial \boldsymbol{w}_n} = \mathcal{M}_n \times \frac{\partial \boldsymbol{z}_n}{\partial \boldsymbol{w}_n}$$

Où $\mathcal{M}_n$, que l'on appellera ici *rétro-message* de rang n, est un facteur cumulé des dérivées des couches de rang supérieur à n par rapport à leur entrée :

$$\mathcal{M}_n = \prod_{k=n+1}^{k=N} \frac{\partial \boldsymbol{z}_k}{\partial \boldsymbol{z}_{k-1}}$$

En respectant les règles d'**initialisation** et de **généralisation** précédentes, la procédure de rétropropagation, détaillée dans Algorithme 1, permet de calculer les différentes composantes du gradient nécessaires à la mise à jour des différents poids du réseau.

Algorithme 1 : Algorithme de rétropropagation

Données : $\forall n, \boldsymbol{z}_n, \boldsymbol{w}_n$; `// après propagation de` $\boldsymbol{x}$
Résultat : $\forall n, \nabla_n$;

$\mathcal{M} \leftarrow 1$;
$\nabla_N \leftarrow \frac{\partial \boldsymbol{z}_N}{\partial \boldsymbol{w}_N}$;
$n \leftarrow N - 1$;
tant que $n > 0$ **faire**
  $\mathcal{M} \leftarrow \mathcal{M} \times \frac{\partial \boldsymbol{z}_{n+1}}{\partial \boldsymbol{z}_n}$; `// Mise à jour du rétro-message`
  $\nabla_n \leftarrow \mathcal{M} \times \frac{\partial \boldsymbol{z}_n}{\partial \boldsymbol{w}_n}$; `//` n`-ième composante du gradient`
  $n \leftarrow n - 1$;
fin

Les réseaux neuronaux convolutifs les plus performants comptent plusieurs dizaines à plusieurs centaines de millions de poids devant être mis à jour un grand nombre de fois au cours des itérations d'apprentissage (voir Figure 2.1). L'algorithme de *rétropropagation* peut alors devenir une opération extrêmement coûteuse et les architectures neuronales profondes ne doivent leur popularité, voire même leur viabilité, qu'à un très haut degré de parallélisation de l'apprentissage. En effet, Nordström and Svensson [1992] remarquent que la structure des réseaux permet de relever de nombreuses opérations indépendantes dans la procédure de construction du gradient par *rétropropagation*.

Parmi les opérations indépendantes les plus évidentes, on note par exemple que les neurones d'une couche donnée, qu'elle soit `Dense` ou `Conv2D`, ne sont pas connectés les uns aux autres. Il est ainsi possible de calculer les sorties et dérivées pour chacun d'eux en même temps. Chellapilla et al. [2006], Ciresan et al. [2011] et Krizhevsky et al. [2012] établissent les premiers succès retentissants de l'apprentissage profond en relevant que beaucoup de ces opérations sont élémentaires et peuvent matériellement être exécutées simultanément. Ils détournent ainsi des processeurs graphiques, *Graphics Processing Units, GPU*, de leur usage premier (génération rapide d'images) pour paralléliser massivement ces calculs.

Un autre point important à soulever lorsque l'on parle du déploiement d'un réseau de neurones pour analyser des images est le recueil de données annotées. En tant que méthode d'apprentissage statistique, les réseaux neuronaux convolutifs profonds requièrent un grand nombre d'exemples qui doivent exprimer toute la variabilité des images devant être prédites après déploiement. C'est donc également à la construction de grandes bases de données

d'images, telles que ImageNet Deng et al. [2009], pouvant être rangées dans un grand nombre de catégories, plus de 1000 pour ImageNet Fellbaum [2005], que le recours aux modèles d'apprentissage statistique s'est généralisé.

Ces recueils de données massifs font aussi émerger un nouveau paradigme d'apprentissage *par transfert*, qui consiste à exploiter les poids d'un réseau déjà entraîné sur une tâche similaire ou *auxiliaire*. De manière assez intuitive, des réseaux entraînés à distinguer des centaines de classes dans des images aussi variées que celles de la base de données ImageNet, sont supposés extraire des caractéristiques très générales qui peuvent s'avérer utiles pour analyser tout type d'images. L'idée de base de l'adaptation d'un modèle est d'initialiser les poids du réseau avec ceux déjà appris sur ImageNet afin de ne réaliser qu'un minimum d'itérations d'apprentissage sur l'ensemble spécifique des données d'intérêt. Carreira et al. [2016], Chen et al. [2017], Ren et al. [2015] et Weinzaepfel et al. [2013] sont autant d'exemples d'applications, parmi les plus innovants, qui exploitent ce principe pour bâtir leur modèle prédictif sur le *transfert* de descripteurs appris sur ImageNet.

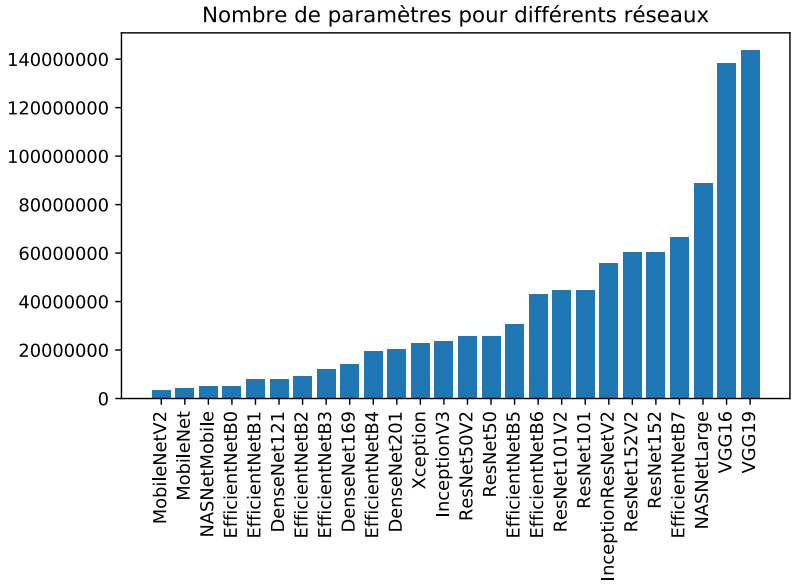


FIGURE 2.1 – Différentes architectures de réseaux neuronaux convolutifs ordonnées par nombre croissant de paramètres.

2.2.5 Bilan

Cette section présentait succinctement les principes qui permettent de construire et d'optimiser les réseaux neuronaux convolutifs profonds. Les principes de fonctionnement de ces modèles ont toujours été définis et expliqués dans un souci de rattachement formel à la méthodologie des *sacs de mots* et à la notion plus spécifique de *chaîne perceptuelle* que nous avions définie dans le premier chapitre. Les maillons de cette *chaîne*, ici instanciés par des *couches* de neurones, disposent d'un **vocabulaire** pré-enregistré, sous la forme de motifs élémentaires appris et reconnus par les neurones, d'une procédure de **traduction** qui transcrit un stimulus-image en *cartes caractéristiques*, et d'outils d'**agrégation** qui limitent la dimension finale de l'encodage de l'image.

Le schéma classique des *sacs de mots* fait usage de méthodes de clustering non-supervisées, lancées indépendamment sur les différents maillons de la chaîne, afin d'apprendre des **vocabulaires** qui présentent de « bonnes » propriétés, mais dont le lien avec l'objectif d'analyse final reste très indirect. À l'inverse, les réseaux neuronaux exploitent, durant l'apprentissage,

le lien formel qui lie les différents maillons de leur *chaîne perceptuelle* et s'assurent que chaque mot de vocabulaire appris par une couche l'a été pour servir l'objectif global de l'analyse. Ce fonctionnement, en plus de garantir la construction d'une solution sur mesure pour la résolution d'un problème d'analyse posé, simplifie drastiquement les efforts de développement de la solution. En effet, le travail ne consiste plus à définir et optimiser des critères plus ou moins arbitraires pour apprendre les différents vocabulaires de la chaîne, mais simplement à énoncer une unique fonction d'erreur de prédiction du réseau qui retranscrit fidèlement la performance d'analyse du système dans sa globalité et la *rétropropagation* se charge d'optimiser en conséquence les différentes couches du réseau.

Ces outils constituent actuellement l'état de l'art de l'analyse d'image automatique, mais la compatibilité de ces solutions avec des applications biomédicales reste à prouver. L'implémentation de ces méthodes prédictives sur des données de patients doit par exemple pouvoir garantir un taux de réussite et de reproductibilité souvent incompatibles avec l'application de méthodes statistiques. De plus, le coût d'entraînement des réseaux, principalement lié au coût du rassemblement des données biologiques, peut être significativement plus élevé que pour des applications plus « classiques ». Dans ces travaux de thèse, nous nous concentrons particulièrement sur l'application de ces techniques au service de l'anatomie et de la cytologie pathologique. Cette spécialité médicale consiste à reconnaître les anomalies des cellules et des tissus d'un organisme, appelées *lésions*, afin d'établir le diagnostic des pathologies, porter un pronostic, evaluer la réponse d'un patient à une thérapie et, plus généralement, étudier et comprendre les mécanismes des pathologies. Il convient donc à présent de décrire un peu plus précisément la spécificité technique des images étudiées et des différentes attentes et contraintes liées à la pratique de l'Anatomo-cyto-pathologie.

2.3 Contexte de la pathologie numérique

Le champ d'application de ces travaux de thèse est le matériel de base sur lequel s'appuient quotidiennement les pathologistes pour établir des diagnostics : les images microscopiques de coupes histologiques. Aussi forte que puisse paraître la restriction aux applications de l'anatomie et cytologie pathologique, elle conduit néanmoins à se pencher sur une variété d'images objectivement plus conséquente que celle de la plupart des secteurs de l'imagerie médicale. Des images issues d'une large gamme de techniques histologiques et d'imagerie ont d'ailleurs été explorées durant ces travaux, notamment grâce aux infrastructures de pointe intégrées dans la plateforme *Imag'IN* [2] de l'Institut Universitaire de Cancer de Toulouse Oncopole. Cette section fait un tour d'horizon rapide de certaines techniques histologiques et des principaux concepts d'imagerie utilisés quotidiennement dans le service d'Anatomo-cyto-pathologie de l'Oncopole de Toulouse pour le diagnostic des cancers.

2.3.1 Techniques histologiques

Les techniques histologiques ont pour objectif d'obtenir des coupes fines et colorées de matériel biologique propices à l'observation au microscope. Nous présentons ici quelques éléments se rapportant à la préparation de lames histologiques. Ce développement n'a pas la prétention d'entrer précisément dans le détail d'une technique, ni même de dresser un catalogue exhaustif de toutes les techniques existantes, mais plutôt de décrire sommairement les étapes de préparation des échantillons afin de comprendre les sources principales de variabilité dans l'aspect des images. Les différentes modalités d'imagerie seront également abordées pour

2. https://www.imagin.univ-tlse3.fr/ImagIn

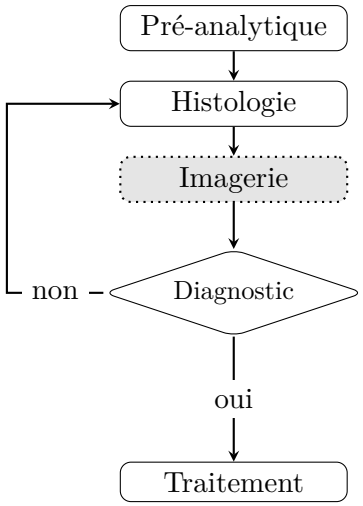


FIGURE 2.2 – Parcours d'un prélèvement dans le laboratoire d'Anapath.

exprimer la variété de structures et d'informations représentée dans les images étudiées dans la suite de ce travail. Le parcours suivi dans le laboratoire par un tissu prélevé sur un patient est résumé dans la Figure 2.2.

2.3.1.1 Le pré-analytique

Le prélèvement est d'abord effectué sur le patient par frottis, biopsie, résection, *etc.* Il s'ensuit deux grandes étapes de traitements regroupées sous le nom de « Pré-analytique » :

- **La fixation** de l'échantillon a pour objectif d'empêcher la dégradation naturelle des tissus et de « durcir » la pièce afin d'en faciliter la découpe très fine. L'opération doit être réalisée en conservant au maximum l'intégrité structurale des tissus, c'est-à-dire en limitant les effets de rétractation, de dilatation et de distorsion des structures. Cette opération de fixation est souvent réalisée en plongeant les tissus dans du formol, puis en les enrobant de paraffine ou de résine.
- **La coupe** consiste à trancher le bloc de paraffine en sections de quelques μm. Cette procédure, réalisée à l'aide d'un *microtome*, permet d'obtenir des coupes de tissus très fines que l'on dépose sur des lames de verre avant de les colorer et de les recouvrir d'une lamelle de verre ou d'un film plastique pour permettre l'observation microscopique et l'acquisition d'images. Ce processus permet de laisser passer la lumière lorsque le tissu est imagé en lumière transmise (microscopie « classique ») ou de ne récupérer que le signal d'une fine couche cellulaire dans le cas de l'imagerie en fluorescence.

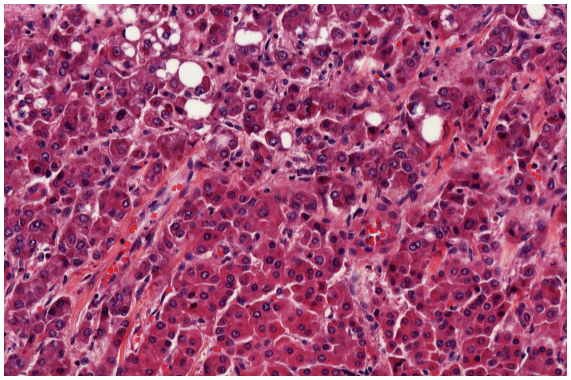
2.3.1.2 La coloration

Après le pré-analytique et avant de pouvoir être examinée au microscope ou numérisée, la lame doit encore subir une étape de *coloration*. L'objectif de cette étape dépend alors notamment de ce que le pathologiste désire observer dans l'image. Les colorations peuvent ainsi être envisagées selon les deux cas d'application suivants :

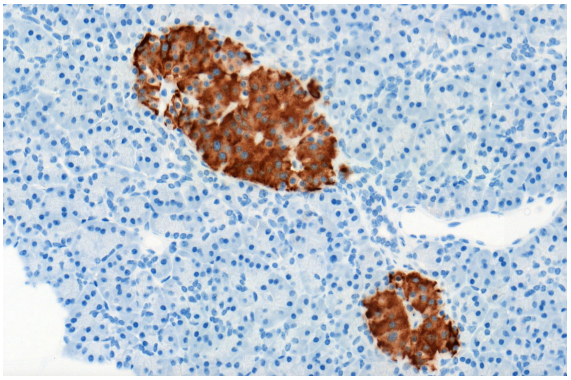
- **L'examen morphologique** qui consiste simplement à observer l'architecture générale des tissus. Il est rendu possible par un marquage *Hématoxyline et Éosine, H&E*, où l'*hématoxyline* va colorer les noyaux de cellules en bleu, et l'*éosine* va colorer les cytoplasmes en rose, Figure 2.3 (a).

- **L'examen fonctionnel** ou *immunohistochimie, IHC*, qui permet d'observer certaines protéines différentiellement exprimées selon les types cellulaires. Pour cela, la lamelle de tissu est mise en présence d'un *anticorps* choisi spécifiquement pour se fixer à l'*antigène* d'intérêt dont on souhaite observer la présence. Cet *anticorps* est conjugué à une enzyme qui, lors de la formation de la liaison *anticorps-antigène*, provoque une réaction colorée. Ce type de marquage va ainsi teinter la membrane, le cytoplasme ou encore le noyau de types cellulaires spécifiques et permettre ainsi de les identifier et de les localiser dans le tissu, Figure 2.3 (b).

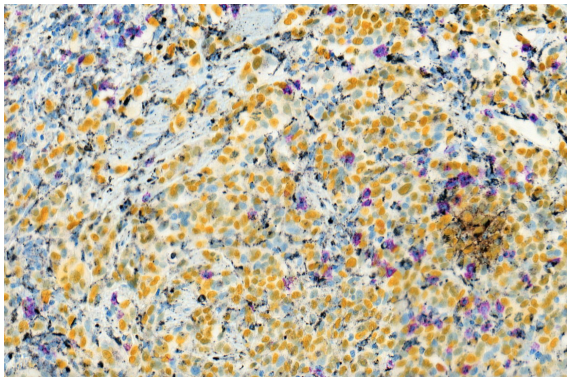
Dans le cas de l'*IHC*, il est également possible d'utiliser simultanément plusieurs marqueurs de couleurs différentes. Le marquage peut alors discriminer un type cellulaire ou permettre d'observer des interactions entre différentes populations de cellules, comme par exemple l'attaque des cellules tumorales par les cellules du système immunitaire. Lorsque plusieurs marqueurs sont impliqués, on parle d'*IHC multiplexe* et les marquages, bien que plus informatifs, sont à la fois plus difficiles à mettre en œuvre, car il faut trouver des couleurs différentes et des protocoles compatibles pour les différents *anticorps*, et plus difficiles à analyser par l'expert, car l'image contient plus d'informations colorimétriques à recouper pour identifier des cellules, Figure 2.3 (c).



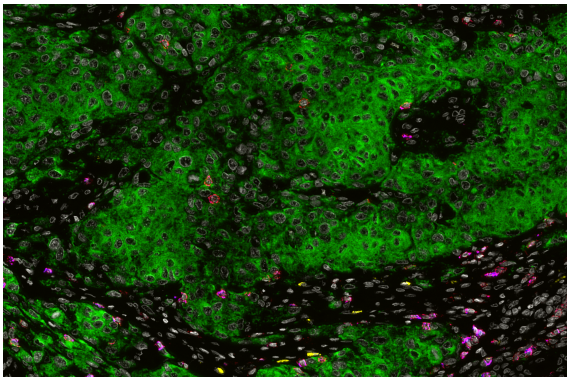
(a) Hématoxyline & Éosine



(b) Simplexe Chromogénique



(c) Multiplexe Chromogénique



(d) Multiplexe Fluorescent

FIGURE 2.3 – (a) Coupe de foie colorée à l'hématoxyline et à l'éosine. (b) Coupe de pancréas colorée à l'hématoxyline (noyaux en bleu) et marquée pour l'expression de l'insuline (marron) qui se manifeste dans le cytoplasme des cellules β des ilôts de Langerhans. (c) Multiplexe chromogénique, les noyaux sont marqués en bleu, les noyaux de mélanomes sont marqués en jaune, le cytoplasme des lymphocytes sont marqués en violet et les lysosomes sont marqués en noir. (d) Multiplexe fluorescent, le cytoplasme des cellules de mélanome est marqué en vert, les membranes des lymphocytes sont marquées en rouge, les vésicules de certaines cellules sont marquées en violet (cytoplasme) et les noyaux apparaissent en gris.

2.3.1.3 L'imagerie

En sortie de l'étape de coloration, la lame est prête à être examinée par le pathologiste. Cette étape est le plus souvent réalisée en observant la lame directement au microscope *optique* ou *photonique*, on parle aussi d'imagerie *fond clair* ou en *lumière transmise*. Cette modalité d'imagerie microscopique, la plus ancienne et la plus répandue, consiste à observer le signal lumineux (lumière blanche) ayant traversé l'échantillon.

Pour certaines applications cependant, notamment dans le cas des *multiplexes*, la microscopie optique présente certaines limitations. En effet, dans ce cas d'usage la restriction des couleurs au spectre visible, à laquelle s'ajoute l'incompatibilité des protocoles de marquage par anticorps, vient rapidement restreindre le panel d'antigènes observables et différentiables simultanément sur une même lame. Une première façon de contourner ce problème est l'usage de coupes dites *sériées*, c'est-à-dire d'utiliser plusieurs coupes d'un même bloc de paraffine (et donc d'un même patient). Si les coupes successives sont réalisées avec peu d'intervalle, la ressemblance d'une coupe à la suivante est suffisante pour une comparaison et il est possible d'appliquer des protocoles différents d'une lame à l'autre. Le pathologiste observe ainsi dans des lames similaires la présence de multiples antigènes pourtant incompatibles au multiplexage.

Si le détournement précédent permet de contrôler l'expression simultanée de plusieurs marquages dans un même échantillon, il se limite cependant à une allure générale de l'expression de l'antigène. En effet, sur la base de deux lames différentes, le recalage des structures ne saurait être parfait et les cellules observées sur une coupe sont souvent différentes de celles observées sur la suivante dans la série. La superposition des marquages et l'étude des co-expressions, c'est-à-dire de la position simultanée de plusieurs marquages sur une même structure, deviennent alors difficiles, voire impossibles.

Lorsque des comparaisons précises, notamment pour l'extraction de grandeurs quantitatives d'expression des marquages pour chaque structure deviennent nécessaires, un multiplexe en *fluorescence*, souvent plus coûteux, peut encore être envisagé. Dans cette procédure, chacun des anticorps d'intérêt $a_1, \ldots, a_n$, on parle ici de *n-plexe*, est marqué par un fluorochrome différent $f_1, \ldots, f_n$. Chaque fluorochrome, f_i, n'émet de la lumière que dans une longueur d'onde spécifique, em_i, et ne peut le faire qu'après avoir été excité avec un signal lumineux de longueur d'onde tout aussi spécifique, ex_i, telle que $ex_i < em_i$, Figure 2.3 (d). Ces spécificités complexifient grandement le processus d'imagerie : le microscope doit désormais se doter de filtres optiques et de miroirs dichroiques, qui vont permettre d'une part l'excitation de chacun des fluorochromes à une longueur d'onde spécifique, et d'autres part, de récupérer la lumière émise par chaque fluorochrome préalablement excité, et ceci de manière séquentielle.

Une dernière technologie d'imagerie présente sur la plateforme *Imag'IN*, plutôt réservée aux activités de recherche, est la microscopie confocale. Sans entrer dans le détail des montages optiques qui amènent ces propriétés, nous dirons que les microscopes confocaux sont capables, en imagerie de fluorescence, de conserver uniquement la lumière émise dans leur plan focal. Cela permet de se débarrasser d'une grande partie du signal parasite habituellement intégré sur toute l'épaisseur de la lame de verre. Le plan focal de ce genre de systèmes est mobile et permet un déplacement dans l'épaisseur de l'échantillon. Le tissu du patient peut alors être observé selon une troisième dimension spatiale et ce dispositif permet une quantification plus précise des marquages en donnant accès à leur expression en 3 dimensions.

2.3.1.4 Numérisation et structure des images

De l'intérêt de la numérisation L'aspect grisé et pointillé de l'étape d'*imagerie* dans la Figure 2.2 du parcours de l'échantillon, marque le caractère optionnel, non pas de l'obtention d'une image, qui est nécessaire à l'examen par le pathologiste, mais plutôt de sa numérisation.

En effet, dans sa pratique quotidienne, le pathologiste dispose d'un microscope personnel avec lequel il examine directement les lames de verre et l'étape de numérisation, qui permet le stockage informatique et la restitution de l'image par le biais d'une application, demeure optionnelle.

Plusieurs faits expliquent le caractère optionnel de la numérisation des lames. Tout d'abord, contrairement aux scanners rayons-X ou aux IRM, qui reposent sur des images numériques depuis leur invention, l'imagerie histologique a historiquement toujours été pratiquée avec des microscopes. Remplacer soudainement cet outil, sur lequel reposent des siècles de pratique de la pathologie, par des visualisations sur écrans d'ordinateurs est une entreprise particulièrement ardue. Ensuite, l'investissement de départ est très conséquent, ce qui constitue souvent un frein important pour un laboratoire dont l'activité fonctionne déjà très bien avec des microscopes. Enfin d'autres arguments concernant la résolution, toujours un peu inférieure à celle de la microscopie classique, ou le temps de numérisation, qui pourrait ralentir le débit d'un laboratoire, sont de moins en moins recevables compte tenu des technologies utilisées de nos ajouterons.

Scanner	fond clair	fluorescence	confocal	quantité
3DHISTECH Pannoramic Confocal	oui	oui	oui	1
3DHISTECH Pannoramic 250 Flash 2	oui	oui	non	2
ZEISS AxioScan.Z1	oui	oui	non	1

TABLEAU 2.1 – Scanners présents sur la plateforme Imag'IN

Le laboratoire d'anapath de l'Oncopole de Toulouse s'est ainsi doté de scanners capables de numériser des lames pour chacune des modalités d'imagerie présentées ci-dessus (voir Tableau 2.1). La numérisation de lames a d'abord été systématisée en routine pour l'imagerie de fluorescence nécessaire aux diagnostics de *FISH* (*Fluorescent In Situ Hybridization*), pour laquelle le support numérique présente l'avantage de ne jamais être dégradé, alors que la lame physique perd ses propriétés fluorescentes sur le long terme.

Au-delà de cet avantage, la généralisation de la numérisation est également un gage de qualité et de traçabilité des pratiques. Il est toujours plus simple de rattacher ou corriger l'affectation d'une lame à un patient et de retracer l'historique de tous les examens et toutes les techniques réalisées sur ce patient lorsque l'ensemble des données sont dans un format numérique.

Enfin, et c'est le postulat sur lequel ce travail de thèse fonde une bonne partie de son intérêt industriel, les lames entières numérisées offrent l'opportunité d'utiliser des algorithmes d'analyse automatique de ces images dont l'ambition est de fiabiliser la décision médicale, qu'elle soit diagnostique, pronostique ou prédictive de la réponse à la thérapie.

De la structure des images numérisées Les différents scanners sur le marché présentent des stratégies d'acquisition variées. Les appareils de la plateforme *Imag'IN* acquièrent une série d'images en différents points ou *champs* de la lame. Lorsque la caméra a pris un cliché à la position (x, y) sur la lame, elle est déplacée vers la position $(x + d, y)$ par exemple, où elle prend un nouveau cliché. Cette translation se fait sur une distance d très courte, quelques μm, et par le biais d'une motorisation mécanique. La mécanique de précision rencontre néanmoins ici ses limites, puisqu'il est impossible de garantir que la transformation entre deux champs successifs est bien une translation de taille d. Certaines perturbations, comme de petites rotations ou des translations de tailles inexactes, doivent être informatiquement compensées *a posteriori* (ou au cours du processus d'acquisition). Les *champs* acquis par la machine sont

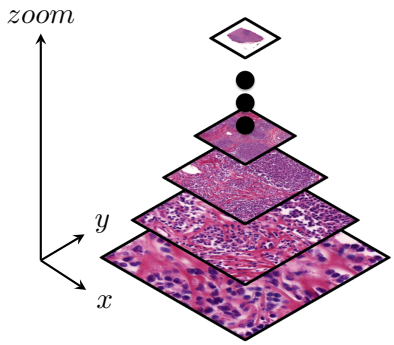


FIGURE 2.4 – Structure pyramidale des images histologiques numérisées.

donc chevauchants pour que la redondance d'information entre deux tuiles voisines permette d'estimer la transformation mécanique exacte de manière à la compenser sur l'image finale. L'estimation et la compensation des transformations sont réalisées par un algorithme de *stitching*.

Sur les lames histologiques, les tissus observés peuvent occuper plusieurs cm^2 et la résolution des images acquises par les scanners de lames de la plateforme est de $0.24\mu m^2$ par pixel en général. Il est cependant possible d'atteindre des résolutions de $0.12\mu m^2$ par pixel en utilisant un objectif de plus fort grossissement. Le nombre total de pixels dans une image, de l'ordre de 10^{10}, pose de véritables problèmes de gestion de la mémoire vive des ordinateurs, tant pour le traitement des images que pour la « simple » tâche de visualisation. Les lames numérisées sont stockées et compressées sous des formes variables selon les constructeurs de scanners. Afin d'optimiser l'accession aux pixels et de rendre possible le développement d'outils de visualisation qui imitent l'utilisation du microscope, les images sont généralement stockées dans des structures pyramidales, Figure 2.4. Le niveau le plus bas de la pyramide correspond à l'image entière à pleine résolution et les dimensions spatiales de l'image, pour un niveau donné, sont divisées par deux pour passer au niveau supérieur.

L'accession aux pixels de l'image consiste à requêter une *imagette* ou *patch* dans la lame, en utilisant 5 paramètres :

- x et y, qui définissent la position du *patch*.
- dx et dy, qui définissent les dimensions spatiales du *patch*.
- s, qui définit le niveau de zoom du *patch* dans la pyramide.

Cette forme de requête est notamment standardisée par certaines APIs [3] pour tous les formats propriétaires de lames numérisées. Selon le type d'imagerie utilisée, l'image obtenue est un tenseur à 3 dimensions, $\{dx, dy, n\}$, pour l'imagerie fond clair et la fluorescence, où n est le nombre de canaux de l'image (fluorochromes dans le cas de la fluorescence) ; ou un tenseur à 4 dimensions, $\{dx, dy, dz, n\}$ dans le cas de la microscopie confocale, qui acquière une troisième dimension spatiale dans l'épaisseur de la lame.

2.3.1.5 Bilan

Cette section présentait de manière synthétique les différentes étapes suivies par un échantillon dans le laboratoire avant l'obtention d'une image numérique. La connaissance, même approximative, de ces étapes est extrêmement importante lorsqu'il s'agit d'analyser les images par des approches d'apprentissage statistique car elle explique les causes, et d'une certaine façon nous fait estimer l'étendue, de la variabilité qui peut exister dans ces images.

3. https://openslide.org/api/python/

Elle nous oblige à envisager des descripteurs et classifieurs d'images insensibles à l'épaisseur de coupe, aux variations de concentration des réactifs de coloration, au volume de formol utilisé pour la fixation, aux modèles, aux optiques, aux filtres d'émission et d'acquisition des scanners, *etc.* Dans le reste de ce chapitre, deux exemples de problématiques biomédicales concrètes que j'ai développées sont présentées. Elles montrent comment les réseaux neuronaux convolutifs profonds, fleuron de l'apprentissage statistique sur les images, peuvent être utilisés pour analyser les lames numérisées et dans quelles mesures ces solutions peuvent être déployées pour répondre aux attentes de la clinique. Abreu et al. [2019] présentent un détecteur de vaisseaux *HEVs* dans des lames marquées par le *MECA-79* en immunohistochimie. Syrykh et al. [2020] proposent un algorithme de diagnostic automatique entre des cas de lymphomes et d'hyperplasie folliculaire.

2.4 Analyse des marquages immunohistochimiques complexes

À l'image de la plupart de ces travaux de thèse, le projet que nous allons décrire ici est construit autour d'une véritable problématique clinique. Il est mené à l'initiative d'un médecin, le docteur Camille Franchet de l'Institut Universitaire du Cancer de Toulouse, et répond à un besoin d'estimation pronostique dans le cancer du sein. Il s'agit de repérer des vaisseaux de tailles et aspects très variés, sur la base d'un marquage immunohistochimique largement exprimé par un grand nombre de structures sans intérêt. Il s'agit de déceler la présence des vaisseaux recherchés au milieu du marquage parasite et de les dénombrer. L'analyse par la machine trouve ici tout son sens puisque ce dénombrement, vraisemblablement bien corrélé à la survie des patients, est jugé trop complexe et fastidieux pour être effectivement réalisé par les pathologistes.

La volonté pragmatique de résoudre cette tâche pour répondre à un besoin clinique, donne lieu à une longue liste de développements informatiques. Il est important de souligner dans ce cas que ce sont les observations et réflexions menées durant l'optimisation itérative du procédé qui ont apporté les clefs d'une contribution plus générale à la méthode. Les métriques de validation définies pour l'occasion, ainsi que leur relevé rigoureux dans diverses expériences, ont permis de constater des écarts de performances entre des modèles aux architectures neuronales et aux ensembles d'apprentissage scrupuleusement identiques. Cette dispersion dans les résultats nous a naturellement conduits à des considérations plus générales sur le consensus entre prédicteurs et à élaborer des stratégies d'optimisation de ce consensus. C'est le fruit de ces réflexions et leur implémentation dans une solution d'analyse que nous décrivons dans la suite de cette section.

2.4.1 Motivations

Les veinules à endothélium épais (*High Endothelial Venules HEVs*) se rencontrent généralement dans les ganglions lymphatiques, dans lesquels elles régulent le transfert des lymphocytes depuis le sang périphérique jusqu'au tissu lymphoïde et ce phénomène est particulièrement étudié par Girard et al. [2012]. Les *HEVs* peuvent également se trouver dans des tissus non-lymphoïdes dans bon nombre de maladies inflammatoires chroniques, ainsi que dans les tumeurs solides chez l'humain (tumeurs du sein, mélanomes primaires, carcinome à cellules squameuses buccal), où elles semblent agir comme une passerelle pour l'infiltration des cellules immunitaires dans la tumeur, puisque certaines études cliniques ont observé de fortes concentrations de *HEVs* dans les tumeurs corrélées avec de bonnes infiltrations des lymphocytes et de meilleurs pronostics. Martinet et al. [2011, 2012], Wirsing et al. [2016] et Allen et al. [2017] ont notamment beaucoup contribuer à décrire ces aspects pronostics.

Dans ce contexte, le dénombrement précis des *HEVs* dans les tumeurs solides pourrait devenir un biomarqueur pertinent pour la clinique. Les *HEVs* peuvent être détectés dans les tumeurs solides humaines via immunomarquage par le *MECA-79*, mais l'évaluation de leur concentration est rendue difficile. D'une part par le « bruit de fond » et l'aspécificité du marquage qui se retrouve fréquemment dans des régions carcinomateuses ou adipeuses, et d'autre part par la répartition non-homogène en taille et en position de ces vaisseaux dans la tumeur, qui contraint à un dénombrement à haute résolution sur l'ensemble de la coupe histologique du tissu du patient.

Nous présentons ici une solution pour la détection automatique des *HEVs* par classification du marquage MECA-79 sur lames entières. Nous proposons un outil de classification, ainsi qu'une métrique pour l'évaluation des performances générales du détecteur proposé. La démarche met en place un algorithme d'entraînement d'assemblées discordantes de réseaux neuronaux et propose d'en évaluer l'intérêt par comparaison avec d'autres approches d'*ensemble learning*. Ce travail souligne également la pertinence de la méthode pour l'analyse de l'immunomarquage *MECA-79*, ainsi que de l'approche par classification pour l'analyse générale des marquages immunohistochimiques.

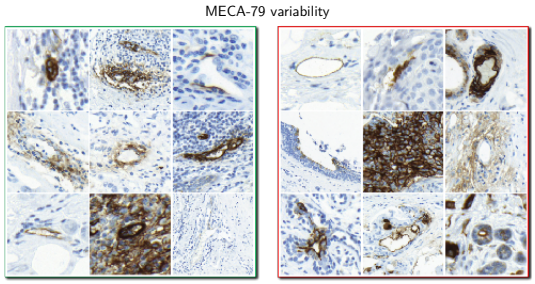


FIGURE 2.5 – MECA-79 marquage spécifique (gauche) et marquage non-spécifique (droite).

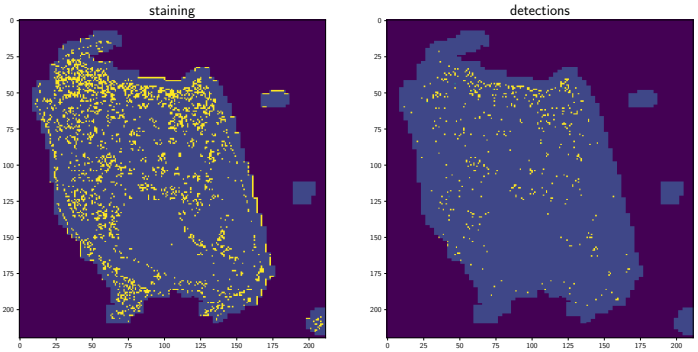


FIGURE 2.6 – MECA-79 seuil sur l'intensité du marquage (gauche) et marquage détecté spécifique (droite). Le nombre de champs que devrait observer le pathologiste a diminué de 83%.

2.4.2 Détection par classement sur lames entières

L'analyse automatique de biomarqueurs immunohistochimiques (*IHC*) est très largement explorée dans la littérature. Shi et al. [2016], Varghese et al. [2014] et Fernández-Carrobles et al. [2017] y voient d'ailleurs un moyen de gagner du temps et d'éviter des erreurs de diagnostic. Cependant, toutes les méthodes explorées sont basées sur une normalisation des couleurs, suivie d'une segmentation, souvent effectuée en positionnant une valeur-seuil de positivité sur l'intensité du marquage. À notre connaissance, au moment de l'écriture la distinction formelle entre le marquage d'une structure recherchée et celui d'une zone de peu d'intérêt n'est jamais

effectuée. Cependant, pour un certain nombre de marqueurs tels que *MECA-79* ou encore *PD-L1*, l'intensité du signal du marquage n'est pas suffisante pour assurer la présence de la structure d'intérêt, voir Figure 2.5 et 2.6.

Nous considérons la détection des *HEVs* comme un problème de classification à trois classes. La lame entière est découpée en un ensemble de patches, chevauchants ou non, $x = \{x_1, \ldots, x_N\}$ où $x_i \in \mathbb{R}^{s \times s \times 3}$ est une image de taille $s \times s$ pixels. Un modèle de classification dont la fonction prédictive, notée f, produit un vecteur de classes correspondant $f(x) = \{y_1, \ldots, y_N\}$ dans lequel $y_i \in [0, 2]$, la prédiction de la tuile x_i, se trouve dans l'une des catégories suivantes : (0) le patch n'est pas marqué par le *MECA-79*, (1) la tuile est marquée mais ne contient pas de *HEV* (nous parlerons de *marquage non-spécifique*), (2) le patch est marqué et contient un *HEV* (nous parlerons de *marquage spécifique*). Enfin, les probabilités d'appartenance à la classe (2) prédites par le modèle, jointes aux positions des tuiles x où elles ont été relevées fournissent une carte de la probabilité de présence des *HEVs* sur la lame entière.

2.4.3 Assemblage de réseaux neuronaux

La combinaison des prédictions de plusieurs réseaux de neurones peut produire de meilleurs résultats de classification qu'un réseau isolé. La supériorité de l'ensemble par rapport au modèle individuel n'est permise qu'en garantissant la diversité de l'ensemble et un certain nombre de stratégies ont été développées pour promouvoir la diversité dans une assemblée de réseaux.

Les techniques les plus classiques de *boosting* développées par Schwenk and Bengio [2000] et Drucker et al. [1993] ainsi que de *bagging* explorées par Opitz and Shavlik [1996] et Zhou et al. [2002a] ont naturellement déjà été implémentées avec succès pour augmenter la diversité d'une assemblée de réseaux neuronaux, et Zhou et al. [2002a] l'ont même appliqué à des problématiques de classement d'images de pathologie. Comme cela a pu être décrit dans le premier chapitre, le *bagging* entraîne différents réseaux sur différentes parties de l'ensemble d'apprentissage, tandis que le *boosting* entraîne les réseaux séquentiellement afin qu'un réseau individuel ne reproduise pas les erreurs de son prédécesseur.

D'autres méthodes comme celle de Maji et al. [2016] proposent d'introduire un processus aléatoire dans la définition de l'architecture des réseaux de l'ensemble, ou dans la configuration des paramètres d'apprentissage des réseaux, ce qui impose généralement l'exploration d'une grille très large de paramètres aléatoires et donc l'entraînement d'un très grand nombre de réseaux. Zhou et al. [2002b] fournit une large gamme de stratégies pour l'assemblage « optimal » de classifieurs. Elles consistent principalement à définir des heuristiques de sélection et de combinaison des réseaux sans laquelle aucune amélioration de l'ensemble par rapport à l'inidvidu ne serait observée la plupart du temps.

Garipov et al. [2018] et Izmailov et al. [2018], parmi les techniques les plus récentes, utilisent la descente de gradient afin de déplacer un réseau entraîné dans l'espace des poids et explorent de cette façon un ensemble de paramétrages des poids restreint aux modèles ayant de bonnes performances de classement. En exploitant, parfois même en forçant, l'imperfection du taux d'apprentissage, il est possible de trouver un ensemble de réseaux suffisamment distants les uns des autres dans l'espace des poids pour améliorer la diversité de l'ensemble.

Contrairement à ces approches, nous choisissons de traiter la diversité comme un problème d'optimisation et proposons une fonction de coût pour maintenir la diversité entre un réseau et un classifieur, ou ensemble de classifieurs, préalablement entraîné. De cette façon, nous réduisons la part de l'aléatoire dans la construction de l'ensemble et améliorons la performance de l'assemblée avec un plus petit nombre de réseaux entraînés. À la différence de Liu and

Yao [1999] par exemple, d'autres travaux définissent également des fonctions de coût d'ensemble, l'implémentation que j'ai proposée est triviale, impose un apprentissage séquentiel très avantageux pour la gestion des ressources mémoire et calcul, et facilite sa combinaison avec toute autre stratégie d'assemblage déjà évoquée. De plus, cette étude montre une amélioration d'ensemble intéressante même avec de petits ensembles de 2 réseaux.

Notons $\mathcal{A}$ une architecture de réseau de neurones, paramétrée par le vecteur de poids w, et $f(w, .)$ sa fonction de prédiction. Un ensemble de n réseaux d'architecture $\mathcal{A}$ est ensuite défini comme un ensemble de vecteurs de poids $\{w_1, \ldots, w_n\}$ et les prédictions $\bar{f}(x)$ de l'ensemble sont généralement calculées en moyennant les prédictions sur les n membres de l'assemblée :

$$\bar{f}(x) = \frac{1}{n} \sum_{i=1}^{n} f(w_i, x) \tag{2.13}$$

Par simplicité, nous faisons le choix, dans le cadre de cette étude, de nous placer dans le cas d'un ensemble de deux réseaux ($n = 2$) et proposons une procédure d'apprentissage séquentielle. Soient x_t, l'ensemble des images d'apprentissage, et y_t l'ensemble des étiquettes correspondantes, la première étape de la procédure d'apprentissage consiste à entraîner un réseau de référence, noté w_r, à minimiser la fonction $\mathcal{H}(f(w_r, x_t), y_t)$, où $\mathcal{H}$ correspond à la fonction d'*entropie croisée*.

Nous entraînons ensuite le second réseau, noté w_c, de manière à assurer une bonne performance de l'assemblage $\{w_r, w_c\}$. Si l'on considère la relation entre l'erreur de classement et l'*ambiguïté* de l'ensemble, vue en détail dans la Sous-section 1.4.6 du premier chapitre, w_c doit avoir de « bonnes » performances de classification et augmenter la diversité de l'ensemble. Si la définition de la diversité d'un ensemble est parfois confuse, comme le montrent notamment les travaux de Didaci et al. [2013] et de Kuncheva and Whitaker [2003], nous avons pensé, inspirés par l'approche dite *à corrélation négative* de Liu and Yao [1999], qu'un moyen indirect, mais néanmoins pratique, d'augmenter la diversité de notre ensemble est de maximiser la distance entre les distributions de sortie des classifieurs individuels $f(w_r, x_t)$ et $f(w_c, x_t)$. Nous en tirons donc l'expression d'une fonction de coût très simple pour l'apprentissage de w_c :

$$\mathcal{L}(w_c, w_r) = \mathcal{H}(f(w_c, x_t), y_t) - k \times \mathcal{H}(f(w_c, x_t), f(w_r, x_t)) \tag{2.14}$$

Le premier terme doit garantir une bonne performance de classification de w_c, tandis que le deuxième terme, que nous appellerons *terme de discordance*, maintient la diversité de l'ensemble. Le *coefficient de discordance* k est un hyperparamètre de la méthode dont nous étudierons rapidement l'impact (Paragraphe 2.4.4.3).

Très proche, dans sa logique séquentielle d'entraînement, de la stratégie de *boosting*, la technique que j'ai développée ici en diffère cependant par deux aspects. Dans sa forme tout d'abord, puisqu'elle intervient directement dans la position du problème d'optimisation, alors que le *boosting* influe sur l'apprentissage du second réseau par pondération des échantillons d'apprentissage.

Elle diffère enfin par la sélectivité du processus de décorrélation : le *boosting* n'impose un désaccord que sur l'erreur commise par les deux modèles, tandis que le terme de discordance s'applique identiquement à tous les échantillons vus au cours du processus d'apprentissage. La présente méthode peut tirer un certain avantage de ce dernier point. En effet, lorsque les bonnes prédictions du réseau référent reposent sur la prise en compte d'un paramètre non-pertinent (pourtant fortement corrélé), le nouveau réseau à tout intérêt à produire une sortie globalement différente afin d'assurer un maximum d'indépendance vis-à-vis des paramètres pris en compte par la référence. De plus, le bénéfice d'une pondération des échantillons, tout particulièrement lorsque le taux d'erreur de la référence devient faible, est statistiquement discutable et peut conduire à de grossières erreurs de sur-apprentissage qui, bien que diluées dans la prédiction d'ensemble, peuvent engendrer une perte de généralisation.

2.4.4 Cadre expérimental et évaluation

2.4.4.1 Classement

Le jeu de données utilisé dans ce travail fait partie d'une cohorte internationale initialement construite par *UNICANCER* : l'étude *PACS04*. Des lames entières (*Whole slide images, WSI*), sont numérisées à forte résolution et stockées dans une structure pyramidale qui autorise la requête d'extraction de pixels d'une position donnée à une résolution donnée. Sans information *a priori* sur le contexte et la résolution nécessaires au modèle pour prendre une décision, nous avons extrait des tuiles de taille $s = 125$ pixels, à une résolution de $0.88\mu m$/pixel, parce qu'elles semblaient rassembler la quantité minimale d'information requise par le pathologiste pour classer les objets marqués. Pour les données annotées, nous avons pu extraire des tuiles disposées sur des annotations humaines effectuées sur 100 *WSI*, et avons ainsi récupéré un total de 4500 patches pour l'apprentissage et 1500 patches pour constituer un ensemble de test, chacun de ces ensembles annotés compte autant de patches dans chacune des classes que le modèle doit prédire.

En guise de modèle individuel de classement, nous utilisons une architecture de réseau très simple qui compte 3 blocks de Convolution-MaxPooling de 32, 64 et 128 filtres respectivement, suivis d'un perceptron à deux couches, chacune de 1024 unités. La sortie du modèle est une couche de 3 neurones avec une fonction d'activation de type *softmax* de manière à produire des probabilités d'appartenance aux 3 classes du problème.

2.4.4.2 Détection

Dans un objectif de détection, chaque zone de la lame contenant du tissu doit être couverte par une prédiction. Compte tenu de la dimension et de la résolution d'extraction des patches, nous choisissons ici, afin d'économiser un peu de temps sur le traitement d'une lame, de découper les lames entières en patches contigus. Comme indiqué précédemment, un classifieur de l'ensemble produit, pour chaque patch, un vecteur de 3 valeurs. La prédiction d'ensemble pour un patch est le vecteur moyen des prédictions individuelles (Equation 2.13) et nous interprétons ses composantes comme des probabilités d'appartenance aux classes de notre problème. Dès lors, à chaque tuile prédite est attribuée la valeur de la composante de prédiction codant pour la classe (2). Ainsi, une carte de probabilité de présence de la classe (2), c'est-à-dire de la présence de *HEV*, est dressée et un seuil doit encore être appliqué à cette carte pour obtenir un masque binaire de détection des vaisseaux. Le seuil est un hyperparamètre de la méthode. Ce dernier peut être ajusté *a posteriori* pour atteindre un niveau satisfaisant de sensibilité et de spécificité du détecteur (Paragraphe 2.4.4.4).

2.4.4.3 Assemblage de réseaux

Nous avons mis en place trois types d'expériences, chacune menée une centaine de fois sur des séparations apprentissage-validation différentes, afin d'évaluer la méthode d'assemblage proposée. Nous montrons d'abord, qu'un entraînement suivant l'Equation 2.14 produit une meilleure performance d'ensemble que des entraînements individuels indépendants et nous évaluons l'impact du *coefficient de discordance* k sur le taux de réussite de l'ensemble. Nous nous plaçons pour cela dans un cadre suffisamment simple et général afin de fournir des résultats reproductibles. Un modèle d'architecture *LeNet* est donc entraîné sur le jeu de données *CIFAR10* et la procédure d'apprentissage est stoppée prématurément lorsque w_c atteint 66% de réussite. Le gain de performance à l'assemblage est alors confirmé pour le

modèle discordant, puisque la performance de classification est toujours meilleure lorsque $k > 0$. On notera également que le taux de réussite de l'ensemble augmente avec les valeurs de k (tableau 2.2).

discordance k	**0.0**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**
%réussite	0.674	0.673	0.678	0.685	0.696	**0.700**

TABLEAU 2.2 – Coefficient de discordance et taux de réussite

Nous proposons ensuite de comparer notre approche avec la méthode récente d'assemblage par moyenne stochastique des poids (*Stochastic Weight Averaging, SWA*) qui repose sur les déplacements liés à la descente de gradient pour trouver des réseaux distants dans l'espace des poids susceptibles de produire des ensembles ayant une bonne diversité. En plus du détail de la méthode, les travaux de Izmailov et al. [2018] fournissent une démarche d'évaluation des méthodes d'assemblage séquentiel : l'idée est de mettre en regard l'amélioration des prédictions avec le surplus d'itérations d'entraînement occasionné. On appelle ainsi *Budget* le nombre d'itérations nécessaires à l'apprentissage du réseau individuel. On exprime alors un surcoût relatif de la méthode d'assemblage en divisant la somme des itérations d'entraînement de tous les réseaux par le *Budget*.

Nous utilisons une nouvelle fois *LeNet* sur *CIFAR10* et employons les paramètres optimaux de la méthode *SWA*. Le *Budget* est fixé à 300 itérations et l'assemblage *SWA* est effectué à partir de l'itération $0.75Budget$ à $1.75Budget$ avec un taux d'apprentissage constant, fixé à 0.001. Nous invitons le lecteur à lire les travaux de Izmailov et al. [2018] pour plus de précisions.

k	**SWA**	**0.0**	**0.1**	**0.2**
budget	1.75	1.66	1.70	1.73
accuracy	0.795	0.798	0.798	**0.799**

TABLEAU 2.3 – Comparison with SWA on CIFAR10 with LeNet

Nous pouvons observer que la méthode proposée fait mieux que *SWA*, le tout avec moins d'itérations (tableau 2.3). Il est intéressant de noter que même un assemblage de réseaux entraînés indépendamment fait mieux que la technique *SWA*. Bien entendu, il faut rappeler ici que *SWA* présente l'avantage de ne produire qu'un seul modèle et que les performances d'ensemble sont atteintes avec un réseau unique, contrairement au cas $k = 0$. Cette infériorité de *SWA* face à un ensemble d'individus entraînés indépendamment a déjà été observée par Garipov et al. [2018] et semble se produire lorsque la méthode n'est plus capable d'assurer une diversité suffisante de l'ensemble. Un changement des paramètres d'apprentissage de la méthode pourrait peut-être réduire cet écart, mais ce sujet n'a pas été exploré par la suite.

Enfin, nous entraînons le détecteur de *HEVs* avec l'Equation 2.14 pour $k = 0.2$, $k = 0.5$ et comparons les taux de réussite avec celui de l'apprentissage indépendant des réseaux individuels. Tous les réseaux ont été entraînés selon l'algorithme du gradient stochastique (*Stochastic Gradient Descent, SGD*) avec un taux d'apprentissage fixé à 0.001. Sans surprise, un retard à la convergence de plus en plus grand est observé lorsque la valeur de k augmente (Figure 2.7) puisque les deux termes de la fonction de coût 2.14 se contredisent sur la majeure partie des échantillons d'apprentissage. Nous améliorons cependant la performance de l'ensemble pour les deux valeurs de $k > 0$ (tableau 2.4 et Figure 2.8).

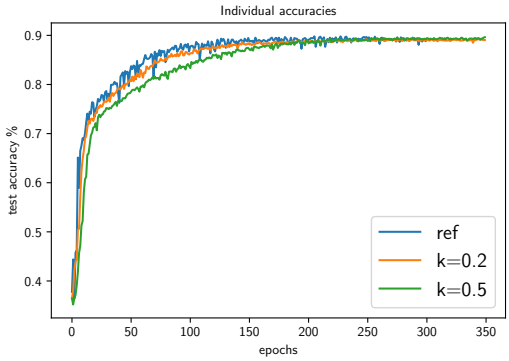


FIGURE 2.7 – Test accuracy during the training of individual w_c with different values of k.

k	0.0	0.2	0.5
acc $\mathbf{w}_c$	0.891	0.890	0.893
acc ens	0.894	0.895	**0.896**

TABLEAU 2.4 – Test accuracy comparison on HEVs data

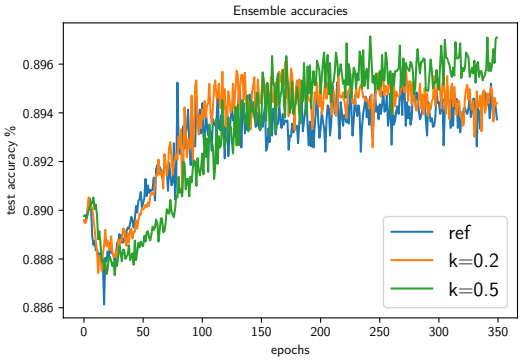


FIGURE 2.8 – Test ensemble accuracy during training

2.4.4.4 Métrique de détection

Pour toutes les lames de la cohorte *PACS04*, nous avons demandé à un expert de placer un point sur chaque *HEV* qu'il parvenait à trouver dans la lame. Pour une lame donnée, soient $d = \{d_1, \ldots, d_n\}$ les positions des détections réalisées par le système, et $a = \{a_1, \ldots, a_m\}$ les positions des annotations expertes sur la lame. En guise de mesure de *sensibilité* du détecteur, nous proposons de calculer la distance entre une annotation et la détection automatique la plus proche :

$$se = \frac{1}{m} \sum_{i=1}^{m} |a_i - nearest(d)|$$

De manière analogue, une mesure de la *spécificité* du détecteur s'écrirait :

$$spe = \frac{1}{n} \sum_{i=1}^{n} |d_i - nearest(a)|$$

Les moyennes de ces métriques ont été relevées sur les 100 lames de l'ensemble de test pour des détecteurs entraînés avec différents taux de discordance (tableau 2.5). Pour les métriques de sensibilité et spécificité, des valeurs de $k > 0$ peuvent améliorer les performances et le meilleur rapport sensibilité-spécificité est obtenu pour une valeur maximale de $k = 0.5$.

k	**0.0**	**0.2**	**0.5**
sens10^3	1.34	**1.31**	1.40
spe10^4	1.54	1.54	**1.45**

Tableau 2.5 – Detector performance on HEVs detection

2.4.5 Bilan

Dans le cas de marquages immunohistochimiques complexes, la détection de structures dans les images de lames entières sur la base d'une classification supervisée des zones marquées est possible et semble plus pertinente qu'une simple quantification du marquage. L'évaluation de la détection peut sembler discutable parce que le relevé manuel des *HEVs* dans les lames entière est une tâche particulièrement fastidieuse lorsque les images peuvent compter plusieurs centaines de *HEVs* dont l'aspect et la taille peuvent beaucoup varier. De plus, aucune convention claire n'avait été fixée sur la position exacte à pointer en cas de présence d'un vaisseau. Cependant, bien que les métriques de spécificité et de sensibilité définies précédemment soient des signaux bruités, les valeurs moyennées sur l'ensemble des lames de validation constituent tout de même de bons indices quant à la performance des détecteurs.

La stratégie d'ensemble proposée devrait faire l'objet d'une évaluation plus poussée, alors que le présent travail ne la présente que comme un outil pour améliorer la classification du marquage immunohistochimique. Pourtant, la simplicité de formulation et d'implémentation de la méthode, ainsi que sa décorrélation non-spécifique, sont autant d'avantages qui justifieraient son utilisation dans les applications de l'*ensemble learning*.

la méthode requiert cependant toujours une classification avec deux réseaux de neurones. Une amélioration conséquente de la méthode, et une perspective de recherche intéressante, serait d'analyser en détail les unités neuronales les plus décorrélées afin de les assembler dans une architecture unifiée et obtenir une prédiction d'ensemble en n'utilisant qu'un seul réseau.

2.5 Aide au diagnostic automatisée

Cette section traite une nouvelle fois un cas particulier, celui du diagnostic différentiel du lymphome folliculaire et de l'hyperplasie folliculaire sur la base de lames en coloration standard hématoxyline et éosine. Si un doute peut exister entre ces deux pathologies, il est rapidement levé par le biais d'un marquage immunohistochimique et ne constitue pas nécessairement un véritable défi de diagnostic. Dans ce cas, l'erreur dans la décision médicale prend éventuellement la forme d'un oubli : le lymphome folliculaire n'a pas été diagnostiqué parce qu'il n'a pas été suspecté à partir de la lame *H&E*.

Si dans cette application particulière l'enjeu biomédical est plus restreint, elle nous permet néanmoins de nous placer dans le contexte plus sensible du diagnostic. Elle constitue tout de même un véritable défi technique et répond également à une interrogation plus générale formulée par les experts pathologistes : un modèle peut-il différencier des lésions très similaires sur le plan morphologique et éviter le recours à une technique immunohistochimique coûteuse ? De manière similaire au problème précédent, ce sont les différentes itérations d'amélioration de la solution d'analyse qui ont progressivement conduit à formuler des problématiques techniques plus spécifiques, notamment sur la mesure du risque, de l'incertitude, mais aussi sur l'interprétation du diagnostic automatisé sur lames entières. Certaines expériences réalisées

au cours de ce travail, dont les résultats sont présentés dans la suite de cette section, nous poussent notamment à adopter un point de vue sceptique, sinon critique, sur une grande partie des travaux menés en *Deep Learning* dans le domaine de la pathologie numérique.

2.5.1 Motivations et travaux associés

2.5.1.1 Analyse de lames entières par réseaux neuronaux convolutifs

Cruz-Roa et al. [2017], Janowczyk and Madabhushi [2016], Komura and Ishikawa [2018], Levine et al. [2019], Khosravi et al. [2018] ou encore Zhang et al. [2019] témoignent de l'ompniprésence des réseaux neuronaux pour la classification et l'analyse automatique des lames entières. Ces outils pourraient notamment réduire la variabilité *inter-* et *intra*-observateurs dans le diagnostic des cancers en permettant une analyse plus objective des coupes histologiques Ehteshami Bejnordi et al. [2017]. De plus, la numérisation des lames et l'analyse par des modèles de *Deep Learning* peuvent potentiellement relever des caractéristiques jusque là inaccessibles à l'inspection visuelle humaine. Levine et al. [2019] et Khosravi et al. [2018] indiquent que de nouvelles caractéristiques morphologiques pourraient ainsi être établies et que la décision médicale, prise sur la base des lames histologiques, en serait grandement fiabilisée.

Dans les travaux les plus récents, notamment ceux de Coudray et al. [2018a] ou de Szegedy et al. [2015], les auteurs conçoivent des chaînes d'analyse automatique qui permettent d'entraîner des réseaux neuronaux convolutifs profonds à distinguer la tumeur du tissu pulmonaire sain sur des images de lames entières *WSI*. Ils présentent ainsi, sur l'exemple du cancer du poumon, une méthodologie de développement d'outils de diagnostic efficaces basés sur des modèles d'apprentissage profond. Les approches de Janowczyk and Madabhushi [2016] et Khosravi et al. [2018] similaires à celle de Coudray et al. [2018a] ont également été utilisées dans d'autres études récentes pour détecter d'autres types de tumeurs ou leurs biomarqueurs associés.

Ces réseaux neuronaux, bien que très performants dans le cadre des études faisant leur présentation, ont tous été conçus sans méthode pour contrôler le degré d'incertitude vis-à-vis de leur prédiction. Cependant, la différence entre des tissus bénins et malins n'est pas toujours tranchée et certaines images sont plus difficiles à analyser que d'autres. Aussi, s'il doit être intégré dans un système d'aide au diagnostic utilisé couramment dans la pratique médicale, nous pensons qu'un système doit être en mesure de fournir une appréciation de la difficulté de la tâche. Cette notion d'incertitude a déjà été abordée dans les travaux récents de Leibig et al. [2017a] sur la détection automatique de rétinopathie diabétique à partir d'images du fond de l'œil. Dans ce travail, les auteurs ont utilisé un réseau dit « bayésien » pour montrer qu'une mesure d'incertitude dans la décision de la machine peut être fournie en même temps que la prédiction. Ils montrent notamment que la prise en compte de cette mesure peut améliorer les performance de diagnostic.

2.5.1.2 Diagnostic des lymphomes

Le diagnostic des lymphomes sur lames histologiques reste une pratique difficile. Laurent et al. [2017], étude récente du laboratoire et menée dans le cadre du réseau français Lymphopath, a pu montrer que 20% des diagnostics de lymphomes en France sont erronés et impactent négativement la prise en charge du patient. Actuellement, le diagnostic des lymphomes est basé sur l'examen d'échantillons de tissus à différents grossissements par un pathologiste dont les suspicions, c'est-à-dire la réduction des solutions possibles à un petit ensemble de pathologies, sont formulées sur la base de caractéristiques morphologiques observées dans les

tissus marqués à l'hématoxyline et à l'éosine (*H&E*). Cependant, Wilkins [2011] montrent que la réalisation de techniques supplémentaires, telles que l'immunohistochimie ou les analyses moléculaires, sont bien souvent nécessaires pour fixer un diagnostic définitif. La mise en place de ces techniques retarde évidemment la prise en charge des patients et l'évaluation qualitative des lames *H&E* demeure subjective et dépend largement de l'expertise du pathologiste.

Dans le domaine de l'hématopathologie, la distinction entre le lymphome folliculaire (*LF*), qui est le deuxième sous-type de lymphome le plus fréquent, et l'hyperplasie folliculaire(*HF*), bénigne, peut être difficile à réaliser, particulièrement sans recours à l'immunohistochimie (voir la Figure 2.9). Dans ce travail, nous proposons une analyse par apprentissage profond pour classer les changements morphologiques d'un ganglion en *LF* ou *HF* uniquement sur la base de *WSIs* sous coloration H&E standard en utilisant un réseau de neurones dit « bayésien » (*BNN*). Similaire, dans les techniques de classification utilisées, aux travaux de Leibig et al. [2017a], l'objectif ici est moins de proposer un outil spécifique pour le diagnostic du lymphome folliculaire que d'évaluer de manière plus approfondie le potentiel ainsi que les limites des *BNNs* pour la distinction de lésions morphologiquement très proches.

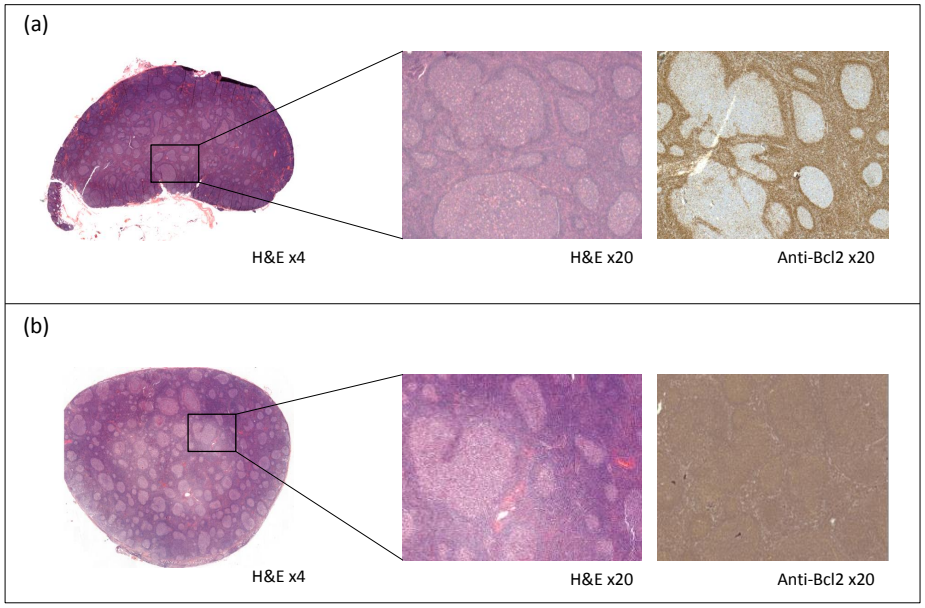


FIGURE 2.9 – Distinction entre l'hyperplasie folliculaire (HF) (a) et le lymphome folliculaire (LF) (b), les lames sont présentées en coloration H&E standard (gauche) et colorées par immunohistochimie (droite) pour révéler l'expression de la protéine Bcl2 qui permet de distinguer les deux pathologies.

2.5.2 Estimation du risque et de l'incertitude

Très générale, voire floue au premier abord, la définition de l'« incertitude » reste pourtant très claire dans le domaine de l'inférence bayésienne et se distingue notamment du « risque ». Ces notions sont toutes deux porteuses d'une information sur les limites de la capacité prédictive d'un modèle probabiliste, mais n'ont pas nécessairement vocation à être corrélées.

- Même pour un modèle ayant convergé, c'est-à-dire dont les paramètres n'évoluent plus sous les itérations d'optimisation, il n'est pas difficile d'envisager une erreur de classement ou de régression irréductible sur l'ensemble de données considéré. On fait ici référence au cas de classement dans lequel les données ne sont effectivement pas

séparables, ou le cas de régression sur des mesures bruitées. On appelle *risque* ou, dans une appellation anglaise plus technique, *aleatoric uncertainty*, cette erreur incompressible sur les prédictions du modèle. Le risque est donc une propriété intrinsèque au problème posé et est indépendant du modèle prédictif. Ajouter des observations, c'est-à-dire des exemples d'apprentissage, ne permettra pas de le faire diminuer. L'évaluation du *risque* dans ce cas est intimement liée à la calibration des modèles et aux travaux de Guo et al. [2017] qui permet d'estimer, après entraînement, le taux d'erreur effectif pour chaque valeur de prédiction du modèle.

- Dans sa définition bayésienne, l'*incertitude*, ou *epistemic uncertainty*, est en fait un indicateur de la méconnaissance du *risque*. Contrairement au *risque*, l'incertitude témoigne d'un manque de connaissance sur le modèle qui peut être levé par l'observation de données supplémentaires. De manière géométrique, ce cas de figure correspond à une grande liberté dans la définition de la frontière entre les classes et rejoint la problématique de généralisation abordée par les *SVMs* et évoquée dans le premier chapitre de ce mémoire, Sous-section 1.4.5. Tandis que la méthode des *SVMs* déduisait un critère géométrique de *marge* qui permettait dans ce cas d'établir le classifieur « le plus général » parmi l'ensemble des classifieurs acceptables, la présente démarche tente plutôt d'établir si le risque est bien évalué statistiquement pour les échantillons présents dans cette zone de l'espace des caractéristiques. Nous allons le voir, cette évaluation est difficile à mettre en place, notamment dans le cas des réseaux de neurones, et repose sur une exploration plus exhaustive de l'ensemble des classifieurs recevables et des propriétés de cet ensemble comme le font notamment Blundell et al. [2015] et Gal and Ghahramani [2016].

Pour une application sensible telle que l'aide au diagnostic, les deux informations s'avèrent précieuses et peuvent à la fois orienter l'expert dans sa décision et permettre l'amélioration du système. Pour une valeur de *risque* élevée, l'expert saura que l'information sur laquelle se base le classifieur pour prendre sa décision ne lui permet pas de se prononcer sur ce cas. Si l'ajout de ce cas à l'ensemble d'apprentissage ne permettra pas d'améliorer le modèle par la suite, cela met toutefois en évidence un manque d'information dans la représentation utilisée par le modèle. Une redéfinition éclairée du modèle ou du problème d'optimisation qu'il résout peut par exemple être mise en place. Pour une valeur d'*incertitude* élevée, l'expert saura que la machine sort de la zone bien couverte par son ensemble d'apprentissage et que la définition de la frontière dans cette zone reste à faire. Les individus difficiles à classer peuvent alors constituer un nouvel ensemble d'apprentissage destiné à améliorer la connaissance du modèle, on parlera ici d'*apprentissage incrémental*.

2.5.2.1 Réseaux neuronaux et inférence bayésienne

Réseaux neuronaux « classiques » Suivant des notations adoptées dans la Section 1.4 du premier chapitre, nous noterons $\mathcal{D} = (X_t, y_t^*)$ l'ensemble d'apprentissage. Comme dans la section précédente, et sans perte de généralité, nous nous cantonnons à une architecture de réseau donnée $\mathcal{A}$, paramétrée par un jeu de poids w et dont la fonction de prédiction sera notée $f(w, .)$. Dans le formalisme probabiliste, le problème d'apprentissage du réseau de neurones relève de l'établissement d'un maximum de la vraisemblance du modèle :

$$w^* = \arg\max_w \mathbb{P}(\mathcal{D}|w) \tag{2.15}$$

À chaque itération d'apprentissage, lorsque les valeurs de poids sont fixées, la vraisemblance du modèle au regard des données d'entraînement est mesurée par la distance entre la distribution des prédictions $f(w, X_t) = y_t$ et la distribution réelle des classes y_t^*. Plus cette distance est élevée, plus le modèle échoue à expliquer les données d'entraînement et l'on reconsidère le problème 2.15 comme un problème de minimisation de cette distance, généralement mesurée

par la fonction $\mathcal{H}$ d'*entropie croisée* :

$$w^* = \arg\min_w \mathcal{H}(f(w, X_t), y_t) \tag{2.16}$$

Réseaux neuronaux bayésiens L'inférence bayésienne appliquée au modèle considéré consiste plutôt à calculer la distribution postérieure des poids sur le jeu de données d'entraînement $\mathbb{P}(w|\mathcal{D})$. L'échantillonnage d'une telle loi permet alors de tirer des paramétrages w du modèle qui sont majoritairement pertinents vis-à-vis de la tâche de classification demandée. Suivant ce modèle, la prédiction d'un nouvel échantillon x est obtenue en moyennant les prédictions de plusieurs réseaux tirés suivant la loi $\mathbb{P}(w|\mathcal{D})$. Ainsi, pour un série de réalisations $\mathbf{w} = \{w_1, \ldots, w_n\}$ de cette loi, nous noterons $f(x, \mathbf{w}) = \mathbf{y}$ le vecteur des prédictions réalisées par les n tirages de poids. La prédiction finale est celle d'une assemblée de réseaux (similaire à l'Equation 2.13) et est calculée comme une prédiction moyenne que l'on notera $\mu(\mathbf{y})$ par la suite.

Contrairement à la section précédente, ce n'est pas la performance de classement de l'ensemble qui est recherchée dans ce cas, mais plutôt sa capacité de généralisation et une forme d'expression de son incertitude. Les réseaux tirés selon la distribution postérieure des poids ont souvent des performances similaires sur l'ensemble d'apprentissage, mais le dessin de leur frontière, notamment dans les zones peu représentées de l'espace des données, peut grandement différer.

Dès lors, si un échantillon-test x est tiré d'une zone peu couverte par les données d'apprentissage, les réseaux individuels auront tendance à avoir des avis divergents sur la prédiction, parce que le degré de liberté dans le tracé de la frontière de décision est élevé dans cette zone. Il est alors possible d'évaluer l'incertitude du modèle, c'est-à-dire évaluer localement si les données d'apprentissage étaient suffisantes pour définir une frontière décisionnelle stable, en mesurant la dispersion des prédictions au sein de l'ensemble, que l'on notera $\sigma(\mathbf{y})$. Le modèle dans ce cas est dit « confiant » dans sa prédiction si la dispersion des réseaux individuels reste faible.

Les réseaux bayésiens en pratique Les modèles de réseaux neuronaux « bayésiens » sont difficiles à mettre en œuvre en pratique, car la distribution postérieure des poids du réseau, $\mathbb{P}(w|\mathcal{D})$, s'avère particulièrement coûteuse à établir. La stratégie la plus répandue dans le domaine, développée par Blundell et al. [2015], consiste à évaluer cette distribution en approximant la distribution des poids par un jeu de gaussiennes, paramétrées par leur valeur moyenne et leur dispersion $\theta = (\mu, \sigma)$. Les valeurs de ces paramètres sont alors mises à jour au cours de l'apprentissage afin de minimiser l'écart entre la distribution approximée par θ et celle observée en appliquant le modèle aux données d'apprentissage. L'écart entre les distributions est généralement mesuré par la divergence de Kullback-Leibler (KL) :

$$\theta^* = \arg\min_\theta KL(\mathbb{P}(\mathbf{w}|\theta), \mathbb{P}(w|\mathcal{D})) \tag{2.17}$$

Blundell et al. [2015], font alors remarquer que ce problème d'optimisation, après développement de la divergence KL et application de la règle de Bayes sur la loi postérieure, est décrit simplement dans le langage des problèmes inverses par le traditionnel compromis entre attache aux données et satisfaction d'une connaissance *a priori* sur le modèle :

$$\theta^* = \arg\min_\theta KL(\mathbb{P}(\mathbf{w}|\theta), \mathbb{P}(\mathbf{w})) - \mathbb{E}_{\mathbb{P}(\mathbf{w}|\theta)}(log\mathbb{P}(\mathcal{D}|\mathbf{w})) \tag{2.18}$$

Le premier terme, $KL(\mathbb{P}(\mathbf{w}|\theta), \mathbb{P}(\mathbf{w}))$, correspond à un terme de régularisation, qui pénalise les distributions trop complexes, tandis que le second terme, $\mathbb{E}_{\mathbb{P}(\mathbf{w}|\theta)}(log\mathbb{P}(\mathcal{D}|\mathbf{w}))$, s'assure de la bonne classification des échantillons d'apprentissage. Il est possible de dériver cette fonction de coût par rapport aux paramètres de chacun des poids du réseau et de procéder à l'optimisation par descente du gradient selon l'algorithme de *rétropropagation* traditionnel.

La méthode de Blundell et al. [2015] fait l'objet des développements principaux en matière de réseaux bayésiens, car elle permet de conserver l'algorithme de *rétropropagation* et ne fait que doubler le nombre de paramètres à optimiser dans le réseau. Ainsi, la méthode est au centre des implémentations des frameworks les plus utilisés tels que Edward [4] ou encore Tensorflow-Probability [5]. La technique n'en reste pas moins coûteuse. En effet, le terme d'attache aux données de l'Equation 2.18 contient une espérance qui est évaluée en pratique par la méthode de Monte-Carlo qui impose, à chaque itération d'apprentissage, de nombreux tirages de modèles suivant la distribution postérieure des poids. L'algorithme d'apprentissage s'en voit considérablement ralenti, ce qui constitue le principal frein à l'application généralisée de cette méthode dans le domaine de l'apprentissage profond.

Plus récemment, Gal and Ghahramani [2016] proposent d'utiliser la régularisation *dropout*, une technique qui retire aléatoirement des neurones dans un réseau, afin d'échantillonner la distribution postérieure des poids et approximer l'inférence bayésienne. Bien que la méthode soit un sujet de controverse, pour son utilisation du terme « incertitude » peut-être inadaptée à la définition bayésienne de l'incertitude selon Osband [2016], elle s'est cependant révélée d'un intérêt pratique dans l'automatisation de certaines décisions médicales et cela sans efforts supplémentaires d'implémentation ni surcharge de mémoire ou de temps d'exécution comme l'indiquent Leibig et al. [2017b]. Pour ces raisons pratiques, nous choisissons d'utiliser cette solution de *dropout* et proposons des protocoles d'évaluation de son intérêt dans l'estimation du *risque* et de l'*incertitude* pour notre cas d'application. Les détails de ces protocoles et le paramétrage utilisé pour la méthode sont donnés par la suite dans la description du cadre expérimental, Sous-section 2.5.3.

Suivant cette méthode, la prédiction d'une seule image est réalisée en passant l'image dans le réseau pour plusieurs tirages de dropout. En tant qu'échantillonnage de la distribution postérieure des poids, la prédiction finale selon cette méthode ne diffère pas de celle décrite dans le cadre général de l'inférence bayésienne par réseaux de neurones, elle est alors calculée comme la moyenne des prédictions pour les différents tirages de dropout $\mu(\mathbf{y})$ et l'incertitude est naturellement donnée par une mesure de la dispersion des prédiction ici notée $\sigma(\mathbf{y})$.

2.5.3 Cadre expérimental et évaluation

2.5.3.1 Recueil des données

Pour le développement de notre algorithme d'analyse, nous avons rétrospectivement recueilli un total de 378 *WSIs* en coloration standard *H&E* de ganglions lymphatiques diagnostiqués *LF* ($n = 197$) ou *HF* ($n = 181$). Dans une autre étape destinée à tester la capacité du système à estimer son *incertitude*, nous avons également recueilli 65 *WSIs* colorées *H&E* de ganglions lymphatiques impliqués dans d'autres lymphomes B à petites cellules, que nous appellerons *non-LF/HF* dans la suite du texte.

Les sections de tissu de lymphomes (*LF* et *non-LF/HF*) ont été récupérées dans la base de données *Lymphopath*. Les diagnostics de ces lames ont été préalablement confirmés par immunohistochimie dans les services de pathologie de deux centres experts (le service de

4. http://edwardlib.org/
5. https://www.tensorflow.org/probability

pathologie de l'Institut Universitaire du Cancer de Toulouse et celui de l'Hôpital Universitaire de Dijon). Les cas *HF* ont pour leur part tous été récupérés dans le service de pathologie de l'Institut Universitaire du Cancer de Toulouse. L'étude a été menée en accord avec les principes de la Déclaration d'Helsinki et les lames *H&E* ont toutes été anonymisées. Les lames ont ensuite été numérisées avec un scanner Panoramic 250 Flash II à une résolution de $0.49\mu m^2$/pixel.

Les *WSIs* de *LF* et *HF* ont été divisées aléatoirement en ensemble d'apprentissage (50%), de validation (25%) et de test (25%). Le classifieur utilisé pour des raisons déjà évoquées dans ce chapitre est entraîné, validé et testé sur de petites parties des lames aussi appelées *patches*. Nous avons donc recueilli, pour chacune des lames de l'ensemble de données, des *patches* de taille 299×299 pixels selon une grille régulière en ignorant les patches couverts par moins de 50% de tissu. Sans connaissance de la quantité de contexte et de la résolution spatiale nécessaire au réseau pour distinguer le motif *LF* du motif *HF* sur un patch donné, tous les patches ont été extraits à 8 niveaux de résolution différents allant de $0.49\mu m$/pixel à $125.44\mu m$/pixel. Nous avons donc pu entraîner, évaluer et stocker un modèle sur chacune de ces différentes résolutions afin d'autoriser une sélection, *a posteriori*, de la meilleure solution ou de la meilleure combinaison. Nous avons ainsi pu récupérer 100 à 1000 patches par *WSI* selon la quantité de tissu présente sur la lame. Globalement, 320000 (20000 pour chacun des niveaux de résolution considérés) patches ont pu être extraits, parmi lesquels 160000 ont servi à l'apprentissage du modèle, 80000 à sa validation et 80000 autres à le tester.

2.5.3.2 Classification de patches et diagnostic sur lame entière

Classification de patches Le modèle a une architecture similaire à celle de la Section 2.4 pour la détection des *HEV*. Comme indiqué précédemment, le système évalue sa confiance sur la prédiction qu'il réalise en utilisant l'échantillonnage *dropout* de Gal and Ghahramani [2016]. Pour la prédiction, nous appliquons la régularisation *dropout* sur les deux couches de perceptron, c'est-à-dire les couches entièrement connectées du réseau, à un taux de 50%. La prédiction du réseau sur un patch, $\mu(\mathbf{y})$, est une moyenne sur 100 tirages de *dropout*, et la confiance du réseau en sa prédiction est donnée par l'écart-type des prédictions, $\sigma(\mathbf{y})$. À la résolution la plus adaptée, nous parvenons à classer 72895 patches sur les 80000 de l'ensemble test, ce qui correspond à un taux de réussite de 91%. L'ensemble des performances aux différentes résolutions ainsi que leur évolution au cours des itérations d'apprentissage sont rassemblés dans la Figure 2.10.

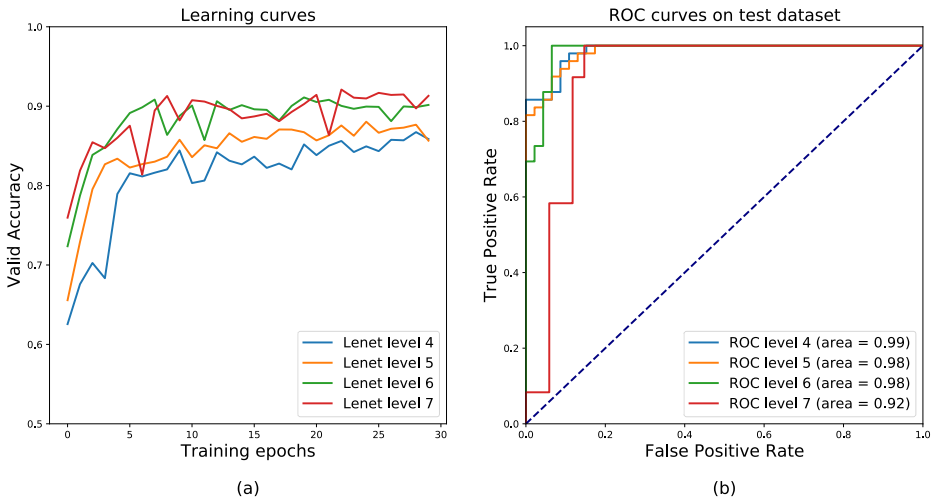


FIGURE 2.10 – Courbes d'apprentissage sur les patches (a) et courbes ROC sur l'intégration à l'échelle de la lame entière et la prédiction du diagnostic (b).

Décision sur lame entière Le diagnostic du patient se fait en intégrant les prédictions réalisées par le modèle sur les patches extraits dans la lame considérée. Plusieurs stratégies d'intégration peuvent évidemment être envisagées ; Coudray et al. [2018b] par exemple, relèvent que dans leur application, la stratégie produisant le moins d'erreurs de diagnostic est celle du vote à la majorité : la catégorie comptant le plus de patches détermine le diagnostic à prédire pour le patient étudié.

Le problème majeur des stratégies d'intégration vient souvent de la présence locale du motif caractéristique des lésions. Tous les patches de la lame ne sont généralement pas porteurs de l'information d'intérêt. Par conséquent, le vote à la majorité, qui étudie des rapports de surfaces entre les différentes pathologies prédites sur la lame, aboutit souvent à une logique d'intégration subjective : selon l'impact d'une fausse alarme ou d'un manque à la détection d'une pathologie, des valeurs minimales de surface des classes doivent être atteintes pour déclencher la suspicion du système.

Afin d'éviter cette surcouche de paramètres, il est souhaitable d'extraire les patches dans des zones restreintes de la lame, connues spécifiquement pour contenir l'information nécessaire à la classification. Pour le système de Coudray et al. [2018b], le fonctionnement est optimal lorsque les patches sont exclusivement extraits dans la zone tumorale. Une procédure de détourage, réalisée par un pathologiste, est préalablement effectuée afin de définir la zone dans laquelle les prédictions doivent être réalisées.

De manière intéressante, cette restriction spatiale de l'analyse conduit à une logique d'intégration vraisemblablement très différente du vote majoritaire proposé par Coudray et al. [2018b] et qui trouve sa justification, non pas dans l'incertitude du modèle de classement des patches, mais bien dans l'incertitude sur la prédiction du diagnostic global de la lame. Considérons ici l'architecture $\mathcal{A}$ dont le jeu de paramètres w est optimisé pour la tâche de classification *LF/HF*, et notons $f(w, .)$ sa fonction prédictive associée. Nous tentons de résoudre le problème de diagnostic d'une lame test (non étudiée par le classifieur durant l'apprentissage) sur la base de patches non-chevauchants, $x = \{x_1, \ldots, x_n\}$, tous extraits dans une zone d'intérêt choisie par l'expert pour sa pertinence vis-à-vis de la décision à prendre.

De manière intuitive, il ne semble pas déraisonnable, pour un classifieur de patches objectivement performant, de considérer le vecteur des prédictions, $y = f(w, x)$, comme une distribution centrée sur le bon diagnostic y^*. Les prédictions des patches, $f(w, x_i)$, sont alors perçues comme des observations indépendantes d'un même phénomène. La tâche de diagnostic, sous cet angle, n'est plus vraiment formulée comme une tâche de classification, mais comme un problème d'estimation ou de régression. Le bon diagnostic, y^*, est l'*explication* des observations y et son estimation est théoriquement donnée par le calcul de la moyenne des observations, et non par l'adoption de la classe majoritairement prédite. Dans le cadre théorique de l'inférence bayésienne rappelé par Osband [2016], cette interprétation stipule également que le doute sur la prédiction de y^* prend bien la forme d'une *incertitude* et que l'ajout d'observations dans y fiabilise d'autant plus l'estimation de y^*.

Diagnostic des lames HF et LF En accord avec des notions d'histopathologie communes, nous supposons que le motif caractéristique du lymphome folliculaire *LF*, s'il est présent dans le ganglion examiné, est observable sur la quasi-totalité de la surface du tissu. Ce postulat de base semble être confirmé par l'allure des courbes d'apprentissage de la Figure 2.10. Les patches d'entraînement sont extraits en tout point des tissus, ce qui n'empêche pas le taux de réussite d'augmenter au fil de l'apprentissage pour atteindre une valeur toujours supérieure à 80%. La cohérence spatiale des prédictions, que l'on peut observer dans la Figure 2.11, indique également que la prédiction *HF/LF* est indépendante de la position du patch dans la lame.

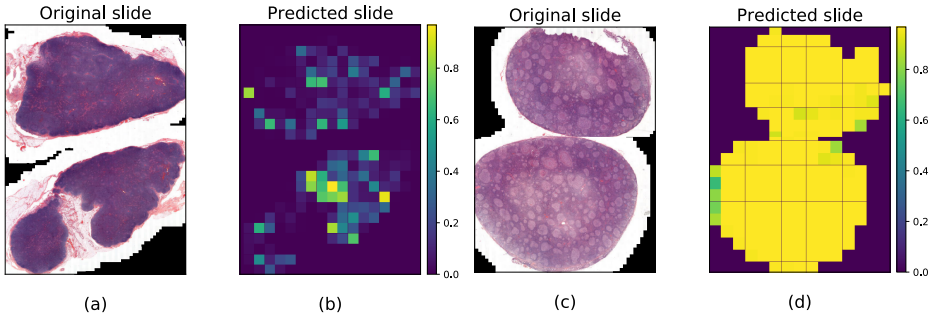


FIGURE 2.11 – Visualisation de la classfication des patches sur les lames entières, pour une hyperplasie folliculaire et un lymphome folliculaire de l'ensemble de test. Chaque patch extrait à la résolution $7.84\mu m$/pixel est coloré par sa probabilité prédite d'appartenance à la classe *FL*. (b) Une lame *HF* présente une probabilité de lymphome très basse quel que soit le patch considéré. (d) La probabilité de lymphome est proche de 1 en tout patch de la lame *LF*.

Cette hypothèse simplifie grandement la constitution des ensembles d'apprentissage et de validation, puisque tout patch extrait d'une lame de patient diagnostiqué *LF* se voit attribuer l'étiquette « *LF* » et inversement, l'étiquette « *HF* » est attribuée à tous les patches extraits d'une lame diagnostiquée *HF*, le tout sans aucune spécification de région d'intérêt par le pathologiste. Sous ce postulat et en suivant la stratégie d'intégration par la moyenne des prédictions locales, notre modèle atteint d'excellentes performances de diagnostic, comme en témoignent les valeurs d'aire sous la courbe ROC, allant de 0.92 à 0.99 dans la Figure 2.10.

De manière intéressante, on relève que ce sont les modèles les moins performants sur la classification des patches qui produisent pourtant les meilleurs résultats pour la prédiction du diagnostic global (Figure 2.10). Ce sont en réalité les réseaux opérant au plus fort niveau de grossissement qui sont en fait les plus mauvais prédicteurs de patches et cela est directement lié à la perte de contexte spatial. Néanmoins, ces patches pris à fort grossissement couvrent, à dimension égale (299×299 pixels), moins de surface de tissu. Il est donc possible d'extraire beaucoup plus de patches non-chevauchants à fort grossissement et l'estimation de y^* en devient d'autant plus robuste car, bien que les prédictions de patches soient plus *risquées* qu'à faible grossissement, l'estimation du diagnostic repose sur un plus grand nombre de réalisations y.

Dispersion dropout et amélioration des prédictions Pour notre application, les réseaux neuronaux bayésiens, ou du moins l'échantillonnage *dropout* de la distribution postérieure des poids du réseau, produisent, pour un patch donné x_i, une série de prédictions, y_i. La prédiction retenue est le centre de la distribution, $\mu(y_i)$, et la dispersion des valeurs prédites, $\sigma(y_i)$, donne un indice de confiance à donner à cette prédiction.

À l'échelle de la lame entière, et pour des raisons similaires à celles de l'intégration des prédictions locales, l'incertitude est également calculée comme une incertitude moyenne relevée sur les différents patches de la lame :

$$\frac{1}{n}\sum_{i=1}^{n}\sigma(y_i)$$

Nous tentons dans cette partie de préciser l'intérêt de cet indice pour la prédiction du diagnostic d'une lame entière. Cet indice, calculé ici dans le cadre du diagnostic *HF/LF*, présente une excellente corrélation avec les erreurs de diagnostic du système, comme cela est présenté dans la Figure 2.12.

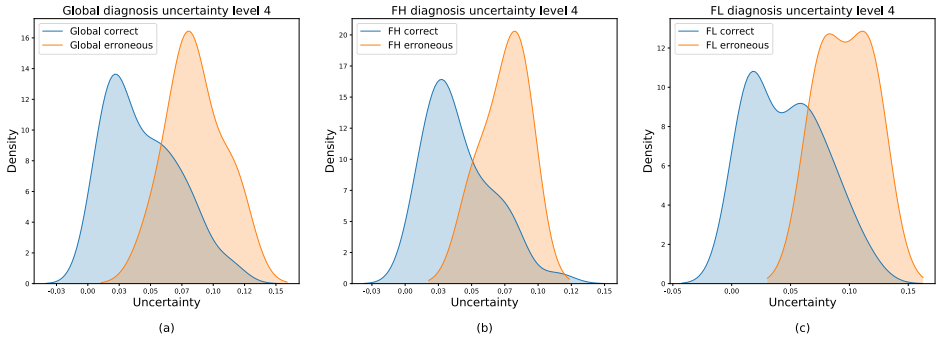


FIGURE 2.12 – Les prédictions erronées sont corrélées à de plus hautes valeurs de dispersion dropout. (a) Densités de dispersion pour toute lame prédite (*HF* ou *LF* sur le diagnostic global), (b) prédiction des lames *HF* uniquement, (c) prédiction des lames *LF* uniquement. Dans chaque cas, la distribution pour un diagnostic correct (bleue) ou un diagnostic faux (orange) est présentée.

Déjà observé par Leibig et al. [2017a], cette propriété permet notamment de renvoyer les cas sur lesquels le système est peu confiant vers l'expert. Pour les niveaux 4 et 5 de la pyramide de résolution, les quartiles du score de confiance nous montrent qu'une sensibilité parfaite dans la détection du lymphome folliculaire (*LF*) est atteinte lorsque 10% des cas les moins confiants sont retirés et la spécificité du système reste élevée puisque les fausses alarmes ne rassemblent que 2% des cas restants. De manière plus générale, on observe que les performances de classification (aire sous la courbe ROC) augmentent pour des valeurs de confiance élevées (Figure 2.13).

Dans le contexte du diagnostic, l'objectif prioritaire est de ne rater aucun lymphome. Il est donc intéressant de noter que le système prédit la classe *LF* ($\sigma = 0.02$) avec une confiance plus élevée que la classe *HF* ($\sigma = 0.04$) et il paraît logique d'utiliser plusieurs seuils de confiance en fonction de la sortie du modèle. Ainsi, un seuil de confiance spécifique à chacune des classes conduit à une amélioration plus rapide des performances de diagnostic, le taux de réussite optimal étant atteint en retirant 23% des cas sur lesquels le réseau est le moins confiant. En revanche, en utilisant un seuil commun à toutes les classes, le taux de réussite optimal n'est obtenu qu'après avoir retiré 36% des cas (Figure 2.13).

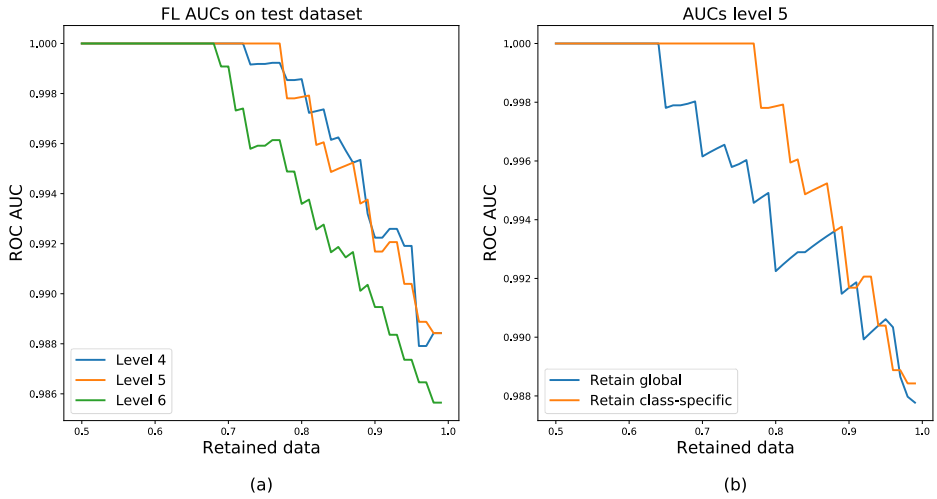


FIGURE 2.13 – Amélioration de l'aire sous la courbe ROC en retirant les cas de plus forte dispersion. (a) L'aire sous la courbe ROC augmente lorsque l'on retire les données les moins sûres pour différents niveaux de résolution. (b) Pour le modèle entraîné à la résolution $15.68\mu m$/pixel (niveau 5), retirer les lames les moins sûres augmente la performance sur les lames restantes. Placer un seuil de confiance spécifique à chaque classe (courbes oranges) s'avère meilleur que la stratégie du seuil global (courbe bleue) à cause des différences de confiance lors de la prédiction des différentes classes.

Dispersion dropout et incertitude Le précédent paragraphe explore l'usage de la dispersion des prédictions pour améliorer le taux de réussite de diagnostic en écartant les prédictions les plus indécises. Selon les définitions de la Sous-section 2.5.2, l'indice de confiance, dans ce cas, était bien étudié comme un indicateur du *risque* de prédiction. Afin de tester la pertinence de la dispersion dropout dans l'estimation de l'*incertitude* de prédiction, nous proposons deux expériences supplémentaires qui consistent à confronter le modèle à des données éloignées de son ensemble d'apprentissage :

- La première expérience, *Exp1* est un scénario de déploiement de la solution. Pour cela, nous construisons délibérément un ensemble de données biaisé, de sorte que les données d'entraînement et de validation sont désormais exclusivement des lames de notre centre, et les données de test comportent essentiellement des cas externes.
- La seconde expérience, *Exp2* est un cas de mauvaise suspicion de départ. Pour cela, nous utilisons le réseau entraîné préalablement sans biais, et observons son comportement sur un jeu de lames dont le diagnostic n'est ni *HF*, ni *LF*.

Dans le cas de *Exp1*, après entraînement, tous les modèles montrent un taux de réussite parfait, $AUC = 1.0$, mais s'avèrent inutilisables sur des cas externes, avec une AUC située entre 0.63 et 0.69 selon le niveau de résolution (Figure 2.14). Néanmoins, une différence significative entre les distributions de l'indice de confiance est observée entre les cas internes et les cas externes dans la Figure 2.14. Dans cette expérience, placer un seuil $t = 0.03$ sur les valeurs d'indice de confiance ne conduit à retirer que 10% des cas internes prédits, tandis que plus de 50% des cas externes ont pu être écartés. Ainsi, bien qu'une distinction parfaite au cas par cas soit impossible, on peut considérer que l'indice de confiance fourni par la dispersion dropout sépare statistiquement favorablement les données internes prédictibles, des données externes méconnues.

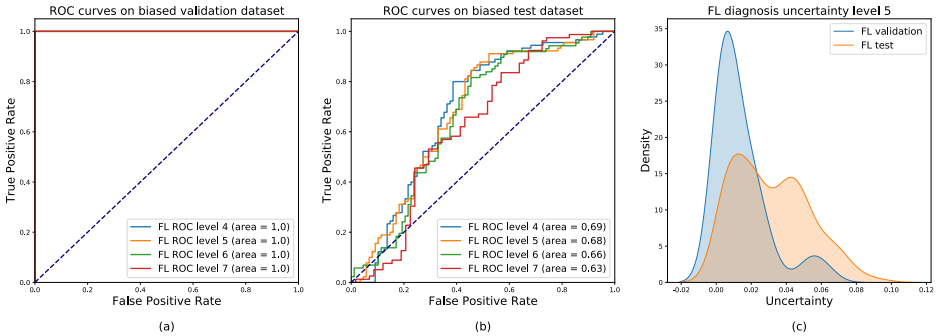


FIGURE 2.14 – Receiver-Operating Characteristics (ROC) et distributions de la dispersion pour l'ensemble de données biaisé. (a) Courbes ROC sur l'ensemble de validation biaisé (composé uniquement de cas internes). (b) Courbes ROC sur l'ensemble de validation biaisé (composé uniquement par des cas externes). (c) Distribution de la dispersion pour les données prédites comme des cas *LF* sur la validation interne (courbe bleue) et sur la validation externe (courbe orange).

L'*Exp2* compare les distributions de l'indice de confiance entre un ensemble de test de diagnostic familier *HF/LF* et un ensemble de lames aux diagnostics étrangers, *non-HF/LF* et les résultats de cette expérience sont présentés dans la Figure 2.15. Les lames *non-HF/LF* sont évidemment toujours prédites comme des cas *LF* ou *HF* par le système, mais les décisions corrèlent encore avec des valeurs de dispersion dropout plus élevées que celles obtenues avec des cas *HF/LF*. La variance dropout montre donc encore ici certaines propriétés discriminantes vis-à-vis de données étrangères à son domaine d'application.

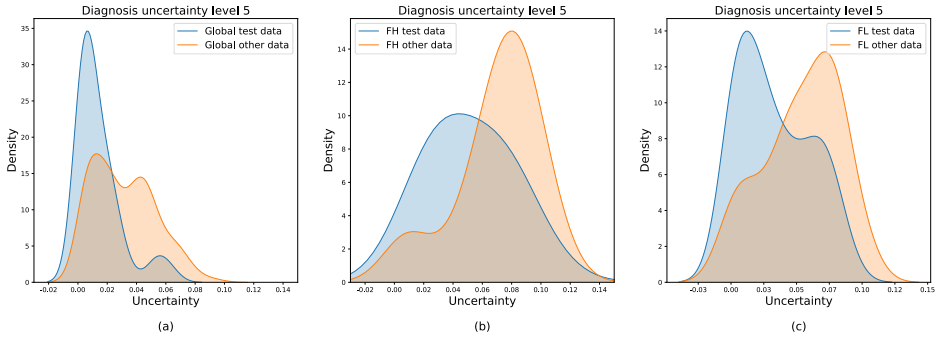


FIGURE 2.15 – Comparaison des distributions de l'indice de confiance entre les données de test familières, *HF/LF*, et d'autres données moins familières *non-HF/LF*. Les distributions sont données pour le diagnostic global (a) et le diagnostic spécifique à chaque classe (b et c). Nous séparons les distributions de test (courbes bleues) et les distributions non familières (courbes oranges) sur chaque graphe. Les lames *non-HF/LF* sont diagnostiquées avec une plus grande dispersion (indice de confiance plus faible) que celle observée dans les données *HF/LF*.

2.5.4 Bilan

Cette section présente un algorithme que j'ai conçu durant ces travaux de thèse. Il constitue un outil efficace d'aide au diagnostic différentiel entre le lymphome folliculaire et l'hyperplasie folliculaire. Bien que le diagnostic en lui-même, dans la pratique médicale quotidienne, ne soit pas nécessairement une tâche difficile, le classifieur développé ici se passe des techniques d'immunohistochimie complémentaires souvent nécessaires pour confirmer le diagnostic du lymphome.

Le présent travail présente également une méthodologie très générale d'analyse de lames entières et aborde des problématiques essentielles pourtant rarement prises en compte dans d'autres travaux similaires. Nous mettons particulièrement l'accent sur l'intégration de l'information locale à l'échelle de la lame entière en justifiant notamment le choix de la moyenne des prédictions dans le cadre de l'inférence bayésienne.

Ce travail porte également une réflexion sur le risque et l'incertitude de prédiction des systèmes. Nous pensons que ces grandeurs, d'autant plus dans le cadre d'applications aussi critiques que le diagnostic médical, doivent absolument être calculées et fournies à l'expert chargé de prendre la décision. Sans garantie sur la sortie des modèles prédictifs et tant que la responsabilité du diagnostic revient au pathologiste, il semble absolument nécessaire de fournir à l'expert toutes les données dont dispose le système sur la prédiction à réaliser.

De manière plus technique, le calcul de la confiance qu'accorde le système à sa propre décision fait l'objet de nombreux développements et de controverses dans le domaine de l'apprentissage profond. Ici, le choix de la méthode est principalement motivé par sa transparence du point de vue de l'implémentation et du temps d'exécution. Cependant, nous l'avons vu, la dispersion des prédictions par échantillonnage dropout ne semble pas couvrir parfaitement toutes les attentes d'une véritable estimation du risque et de l'incertitude. Il reste un bon indicateur statistique, mais ne permet souvent pas d'écarter au cas par cas une mauvaise décision. Son cas d'application le plus efficace s'est avéré être l'estimation du *risque* et ses performances pour une véritable estimation de l'*incertitude* restent à prouver.

L'usage et l'évaluation de cet indicateur de confiance nous a également conduits à envisager des scénarios de déploiement de la solution à d'autres centres. Ces expériences, au-delà d'évaluer notre indicateur de confiance, permettent de se confronter aux réelles difficultés de l'exploitation des solutions de *deep learning* dans des applications médicales. En dépit de leur évolution en dehors du cadre bayésien, les réseaux neuronaux profonds se plient aux lois de l'apprentissage

statistique et requièrent des jeux de données représentatifs afin de pouvoir produire un raisonnement inductif cohérent. Pour atteindre un tel objectif, l'adaptation du système à différents protocoles de marquage ou à différents types de scanners est notamment permise en constituant des jeux de données multi-centriques. Cependant, l'applicabilité universelle de l'algorithme n'est jamais garantie.

À ce problème majeur de généralisation de la solution, c'est encore la confiance du système dans sa décision, dans son estimation de l'*incertitude* qui peut permettre une adaptation rapide aux pratiques d'un nouveau centre. En effet, le modèle peut bénéficier de l'évaluation de l'incertitude sur les échantillons pour constituer, lors d'une phase de pré-production et sans aucune supervision, un ensemble d'apprentissage dit d'« adaptation » qui accoutumerait le système à des données encore non-couvertes par sa précédente phase d'entraînement.

2.6 Conclusion

Le domaine du *deep learning* propose des classifieurs efficaces et susceptibles, à court terme, de remplacer ou compléter les pathologistes sur les tâches les plus pénibles de leur activité. En suivant certaines règles élémentaires de construction de réseaux neuronaux, ainsi qu'en respectant les logiques fondamentales de l'analyse des *WSIs* par *patches*, il est possible d'intégrer rapidement ces outils dans des applications biomédicales pertinentes. Sur le plan technique, une piste d'amélioration prometteuse, tant du point de vue des performances de classification que des attentes biomédicales quant à la certitude du système, pourrait se trouver dans les stratégies de l'*ensemble learning*.

Dans cette section, nous avons proposé une procédure originale destinée à améliorer l'apprentissage des asssemblées de réseaux neuronaux pour des tâches de classification. Nous avons également fait usage d'une stratégie, toujours interprétable sous l'angle de l'*ensemble learning*, pour mesurer le degré de certitude d'un classifieur. Ce domaine de l'apprentissage automatique est intimement lié au pouvoir de généralisation des outils statistiques de classification et permet également d'envisager de nouvelles techniques d'*apprentissage incrémental*. Les classifieurs peuvent être entraînés sur des sous-ensembles « biaisés », ici des centre hospitaliers et des pathologistes différents, et combiner leurs prédictions pour former des consensus d'experts.

Le chapitre suivant énumère les limitations des deux approches de classification qui précèdent pour l'analyse des *WSIs*. Il met notamment l'accent sur le manque de généralisation des représentations apprises par les réseaux neuronaux convolutifs profonds et sur les efforts déployés pour mettre en place ces solutions. Il tente également de développer des procédures pour passer outre ces limitations et rapidement tirer le meilleur parti de ces représentations.

2.7 Références

Arnaud Abreu, Camille Franchet, FX Frenois, P Brousset, JP Girard P Denéfle, P Denéfle, Benoît Naegel, and Cédric Wemmert. Ensemble of neural networks for high endothelial venules detection in meca-79 immunohistochemistry images. In *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pages 938–942. IEEE, 2019. 37, 56

Charlotte Syrykh, Arnaud Abreu, Nadia Amara, Aurore Siegfried, Véronique Maisongrosse, François X Frenois, Laurent Martin, Cédric Rossi, Camille Laurent, and Pierre Brousset. Accurate diagnosis of lymphoma on whole-slide histopathology images using deep learning. *npj Digital Medicine*, 3(1) :1–8, 2020. 37, 56

David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1) :106–154, 1962. 37

S Marĉelja. Mathematical description of the responses of simple cortical cells. *JOSA*, 70(11) : 1297–1300, 1980. 37

John G Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *JOSA A*, 2(7) :1160–1169, 1985. 37

Mark R Turner. Texture discrimination by gabor functions. *Biological cybernetics*, 55(2-3) : 71–82, 1986. 38

Stephane G Mallat. A theory for multiresolution signal decomposition : the wavelet representation. *IEEE transactions on pattern analysis and machine intelligence*, 11(7) :674–693, 1989. 38

Alan C. Bovik, Marianna Clark, and Wilson S. Geisler. Multichannel texture analysis using localized spatial filters. *IEEE transactions on pattern analysis and machine intelligence*, 12 (1) :55–73, 1990. 38

Bruno A Olshausen and David J Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583) :607–609, 1996. 38

Thomas Serre, Lior Wolf, Stanley Bileschi, Maximilian Riesenhuber, and Tomaso Poggio. Robust object recognition with cortex-like mechanisms. *IEEE transactions on pattern analysis and machine intelligence*, 29(3) :411–426, 2007. 38

Shangzhen Luan, Chen Chen, Baochang Zhang, Jungong Han, and Jianzhuang Liu. Gabor convolutional networks. *IEEE Transactions on Image Processing*, 27(9) :4357–4366, 2018. 38

Andrey Alekseev and Anatoly Bobe. Gabornet : Gabor filters with learnable parameters in deep convolutional neural networks. *arXiv preprint arXiv :1904.13204*, 2019. 38

Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4) :115–133, 1943. 38

George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4) :303–314, 1989. 38

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088) :533–536, 1986. 39, 44

Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010. 39

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 41

Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. Squeezenet : Alexnet-level accuracy with 50x fewer parameters and< 0.5 mb model size. *arXiv preprint arXiv :1602.07360*, 2016. 41

Shuyang Sun, Jiangmiao Pang, Jianping Shi, Shuai Yi, and Wanli Ouyang. Fishnet : A versatile backbone for image, region, and pixel level prediction. In *Advances in neural information processing systems*, pages 754–764, 2018. 41

Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4) :541–551, 1989. 41

François Chollet. Xception : Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017. 43

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv :1511.06434*, 2015. 44

Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1520–1528, 2015. 44

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net : Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 44

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 44

DO Hebb. Brain mechanisms and learning. *Distinctive features of learning in the higher animal*, pages 37–46, 1961. 44

Paul Werbos and Paul John. Beyond regression : new tools for prediction and analysis in the behavioral sciences /. 01 1974. 44

D.B. Parker. *Learning-logic : Casting the Cortex of the Human Brain in Silicon.* Technical report : Center for Computational Research in Economics and Management Science. Massachusetts Institute of Technology, Center for Computational Research in Economics and Management Science, 1985. URL https://books.google.fr/books?id=2kS9GwAACAAJ. 44

Yann Lecun. Une procedure d'apprentissage pour reseau a seuil asymmetrique (a learning scheme for asymmetric threshold networks). In *Proceedings of Cognitiva 85, Paris, France*, pages 599–604, 1985. 44

Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks : The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989. 46

Yurii Nesterov. A method for unconstrained convex minimization problem with the rate of convergence. 1983. 46

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(Jul) :2121–2159, 2011. 46

Matthew D Zeiler. Adadelta : an adaptive learning rate method. *arXiv preprint arXiv :1212.5701*, 2012. 46

Diederik P Kingma and Jimmy Ba. Adam : A method for stochastic optimization. *arXiv preprint arXiv :1412.6980*, 2014. 46

Yoshua Bengio. Rmsprop and equilibrated adaptive learning rates for nonconvex optimization. *corr abs/1502.04390*, 2015. 46

Tomas Nordström and Bertil Svensson. Using and designing massively parallel computers for artificial neural networks. *Journal of parallel and distributed computing*, 14(3) :260–285, 1992. 48

Kumar Chellapilla, Sidd Puri, and Patrice Simard. High performance convolutional neural networks for document processing. 2006. 48

Dan Claudiu Ciresan, Ueli Meier, Jonathan Masci, Luca Maria Gambardella, and Jürgen Schmidhuber. Flexible, high performance convolutional neural networks for image classification. In *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011. 48

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 48

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet : A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 49

Christiane Fellbaum. Wordnet and wordnets. 2005. 49

Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human pose estimation with iterative error feedback. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4733–4742, 2016. 49

Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab : Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4) :834–848, 2017. 49

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn : Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 49

Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, and Cordelia Schmid. DeepFlow : Large displacement optical flow with deep matching. In *Proceedings of the IEEE international conference on computer vision*, pages 1385–1392, 2013. 49

Jean-Philippe Girard, Christine Moussion, and Reinhold Förster. HEVs, lymphatics and homeostatic immune cell trafficking in lymph nodes. *Nature Reviews Immunology*, 12(11) : 762, 2012. 56

Ludovic Martinet, Ignacio Garrido, Thomas Filleron, Sophie Le Guellec, Elisabeth Bellard, Jean-Jacques Fournie, Philippe Rochaix, and Jean-Philippe Girard. Human solid tumors contain high endothelial venules : association with t-and b-lymphocyte infiltration and favorable prognosis in breast cancer. *Cancer research*, 2011. 56

Ludovic Martinet, Sophie Le Guellec, Thomas Filleron, Laurence Lamant, Nicolas Meyer, Philippe Rochaix, Ignacio Garrido, and Jean-Philippe Girard. High endothelial venules (HEVs) in human melanoma lesions : major gateways for tumor-infiltrating lymphocytes. *Oncoimmunology*, 1(6) :829–839, 2012. 56

Anna M Wirsing, Oddveig G Rikardsen, Sonja E Steigen, Lars Uhlin-Hansen, and Elin Hadler-Olsen. Presence of tumour high-endothelial venules is an independent positive prognostic factor and stratifies patients with advanced-stage oral squamous cell carcinoma. *Tumor Biology*, 37(2) :2449–2459, 2016. 56

Elizabeth Allen, Arnaud Jabouille, Lee B Rivera, Inge Lodewijckx, Rindert Missiaen, Veronica Steri, Kevin Feyen, Jaime Tawney, Douglas Hanahan, Iacovos P Michael, et al. Combined antiangiogenic and anti–pd-l1 therapy stimulates tumor immunity through HEV formation. *Science translational medicine*, 9(385) :eaak9679, 2017. 56

Peng Shi, Jing Zhong, Jinsheng Hong, Rongfang Huang, Kaijun Wang, and Yunbin Chen. Automated ki-67 quantification of immunohistochemical staining image of human nasopharyngeal carcinoma xenografts. *Scientific reports*, 6 :32127, 2016. 57

Frency Varghese, Amirali B Bukhari, Renu Malhotra, and Abhijit De. Ihc profiler : an open source plugin for the quantitative evaluation and automated scoring of immunohistochemistry images of human tissue samples. *PloS one*, 9(5) :e96801, 2014. 57

M Milagro Fernández-Carrobles, Gloria Bueno, Marcial García-Rojo, Lucía González-López, Carlos López, and Oscar Déniz. Automatic quantification of ihc stain in breast tma using colour analysis. *Computerized Medical Imaging and Graphics*, 61 :14–27, 2017. 57

Holger Schwenk and Yoshua Bengio. Boosting neural networks. *Neural computation*, 12(8) : 1869–1887, 2000. 58

Harris Drucker, Robert Schapire, and Patrice Simard. Boosting performance in neural networks. In *Advances in Pattern Recognition Systems using Neural Network Technologies*, pages 61–75. World Scientific, 1993. 58

David W Opitz and Jude W Shavlik. Generating accurate and diverse members of a neural-network ensemble. In *Advances in neural information processing systems*, pages 535–541, 1996. 58

Zhi-Hua Zhou, Yuan Jiang, Yu-Bin Yang, and Shi-Fu Chen. Lung cancer cell identification based on artificial neural network ensembles. *Artificial Intelligence in Medicine*, 24(1) : 25–36, 2002a. 58

Debapriya Maji, Anirban Santara, Pabitra Mitra, and Debdoot Sheet. Ensemble of deep convolutional neural networks for learning to detect retinal vessels in fundus images. *arXiv preprint arXiv :1603.04833*, 2016. 58

Zhi-Hua Zhou, Jianxin Wu, and Wei Tang. Ensembling neural networks : many could be better than all. *Artificial intelligence*, 137(1-2) :239–263, 2002b. 58

Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry P Vetrov, and Andrew Gordon Wilson. Loss surfaces, mode connectivity, and fast ensembling of dnns. *arXiv preprint arXiv :1802.10026*, 2018. 58, 61

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv :1803.05407*, 2018. 58, 61

Yong Liu and Xin Yao. Ensemble learning via negative correlation. *Neural networks*, 12(10) : 1399–1404, 1999. 58, 59

Luca Didaci, Giorgio Fumera, and Fabio Roli. Diversity in classifier ensembles : Fertile concept or dead end ? In *International workshop on multiple classifier systems*, pages 37–48. Springer, 2013. 59

Ludmila I Kuncheva and Christopher J Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine learning*, 51(2) :181–207, 2003. 59

Angel Cruz-Roa, Hannah Gilmore, Ajay Basavanhally, Michael Feldman, Shridar Ganesan, Natalie N. C. Shih, John Tomaszewski, Fabio A. González, and Anant Madabhushi. Accurate and reproducible invasive breast cancer detection in whole-slide images : A Deep Learning approach for quantifying tumor extent. *Scientific Reports*, 7 :46450, April 2017. ISSN 2045-2322. doi : 10.1038/srep46450. 64

Andrew Janowczyk and Anant Madabhushi. Deep learning for digital pathology image analysis : A comprehensive tutorial with selected use cases. *Journal of Pathology Informatics*, 7 :29, 2016. ISSN 2229-5089. doi : 10.4103/2153-3539.186902. 64

Daisuke Komura and Shumpei Ishikawa. Machine Learning Methods for Histopathological Image Analysis. *Computational and Structural Biotechnology Journal*, 16 :34–42, 2018. ISSN 2001-0370. doi : 10.1016/j.csbj.2018.01.001. 64

Adrian B. Levine, Colin Schlosser, Jasleen Grewal, Robin Coope, Steve J.M. Jones, and Stephen Yip. Rise of the Machines : Advances in Deep Learning for Cancer Diagnosis. *Trends in Cancer*, 5(3) :157–169, March 2019. ISSN 24058033. doi : 10.1016/j.trecan.2019.02.002. URL https://linkinghub.elsevier.com/retrieve/pii/S2405803319300184. 64

Pegah Khosravi, Ehsan Kazemi, Marcin Imielinski, Olivier Elemento, and Iman Hajirasouliha. Deep Convolutional Neural Networks Enable Discrimination of Heterogeneous Digital Pathology Images. *EBioMedicine*, 27 :317–328, January 2018. ISSN 2352-3964. doi : 10.1016/j.ebiom.2017.12.026. 64

Zizhao Zhang, Pingjun Chen, Mason McGough, Fuyong Xing, Chunbao Wang, Marilyn Bui, Yuanpu Xie, Manish Sapkota, Lei Cui, Jasreman Dhillon, Nazeel Ahmad, Farah K. Khalil, Shohreh I. Dickinson, Xiaoshuang Shi, Fujun Liu, Hai Su, Jinzheng Cai, and Lin Yang. Pathologist-level interpretable whole-slide cancer diagnosis with deep learning. *Nature Machine Intelligence*, 1(5) :236, May 2019. ISSN 2522-5839. doi : 10.1038/s42256-019-0052-1. URL https://www.nature.com/articles/s42256-019-0052-1. 64

Babak Ehteshami Bejnordi, Mitko Veta, Paul Johannes van Diest, Bram van Ginneken, Nico Karssemeijer, Geert Litjens, Jeroen A. W. M. van der Laak, the CAMELYON16 Consortium, Meyke Hermsen, Quirine F. Manson, Maschenka Balkenhol, Oscar Geessink, Nikolaos Stathonikos, Marcory Crf van Dijk, Peter Bult, Francisco Beca, Andrew H. Beck, Dayong Wang, Aditya Khosla, Rishab Gargeya, Humayun Irshad, Aoxiao Zhong, Qi Dou, Quanzheng Li, Hao Chen, Huang-Jing Lin, Pheng-Ann Heng, Christian Haß, Elia Bruni, Quincy Wong, Ugur Halici, Mustafa Ümit Öner, Rengul Cetin-Atalay, Matt Berseth, Vitali Khvatkov, Alexei Vylegzhanin, Oren Kraus, Muhammad Shaban, Nasir Rajpoot, Ruqayya Awan, Korsuk Sirinukunwattana, Talha Qaiser, Yee-Wah Tsang, David Tellez, Jonas Annuscheit, Peter Hufnagl, Mira Valkonen, Kimmo Kartasalo, Leena Latonen, Pekka Ruusuvuori, Kaisa Liimatainen, Shadi Albarqouni, Bharti Mungal, Ami George, Stefanie Demirci, Nassir Navab, Seiryo Watanabe, Shigeto Seno, Yoichi Takenaka, Hideo Matsuda, Hady Ahmady Phoulady, Vassili Kovalev, Alexander Kalinovsky, Vitali Liauchuk, Gloria Bueno, M. Milagro Fernandez-Carrobles, Ismael Serrano, Oscar Deniz, Daniel Racoceanu, and Rui Venâncio. Diagnostic Assessment of Deep Learning Algorithms for Detection of Lymph Node Metastases in Women With Breast Cancer. *JAMA*, 318(22) :2199–2210, 2017. ISSN 1538-3598. doi : 10.1001/jama.2017.14585. 64

Nicolas Coudray, Paolo Santiago Ocampo, Theodore Sakellaropoulos, Navneet Narula, Matija Snuderl, David Fenyö, Andre L. Moreira, Narges Razavian, and Aristotelis Tsirigos. Classification and mutation prediction from non-small cell lung cancer histopathology images using deep learning. *Nature Medicine*, 24(10) :1559–1567, October 2018a. ISSN 1546-170X. doi : 10.1038/s41591-018-0177-5. 64

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the Inception Architecture for Computer Vision. *arXiv :1512.00567 [cs]*, December 2015. URL http://arxiv.org/abs/1512.00567. arXiv : 1512.00567. 64

Christian Leibig, Vaneeda Allken, Murat Seçkin Ayhan, Philipp Berens, and Siegfried Wahl. Leveraging uncertainty information from deep neural networks for disease detection. *Scientific reports*, 7(1) :17816, 2017a. 64, 65, 72

Camille Laurent, Marine Baron, Nadia Amara, Corinne Haioun, Mylène Dandoit, Marc Maynadié, Marie Parrens, Beatrice Vergier, Christiane Copie-Bergman, Bettina Fabiani, Alexandra Traverse-Glehen, Nicole Brousse, Marie-Christine Copin, Patrick Tas, Tony Petrella, Marie-Christine Rousselet, Josette Brière, Fréderic Charlotte, Catherine Chassagne-Clement, Thérèse Rousset, Luc Xerri, Anne Moreau, Antoine Martin, Diane Damotte, Peggy Dartigues, Isabelle Soubeyran, Michel Peoch, Pierre Dechelotte, Jean-François Michiels, Antoine de Mascarel, Françoise Berger, Céline Bossard, Flavie Arbion, Isabelle Quintin-Roué, Jean-Michel Picquenot, Martine Patey, Blandine Fabre, Henri Sevestre, Cécile Le Naoures, Marie-Pierre Chenard-Neu, Claire Bastien, Sylvie Thiebault, Laurent Martin, Manuela Delage, Thomas Filleron, Gilles Salles, Thierry Jo Molina, Georges Delsol, Pierre Brousset, and Philippe Gaulard. Impact of Expert Pathologic Review of Lymphoma Diagnosis : Study of Patients From the French Lymphopath Network. *Journal of Clinical Oncology : Official Journal of the American Society of Clinical Oncology*, 35(18) :2008–2017, June 2017. ISSN 1527-7755. doi : 10.1200/JCO.2016.71.2083. 64

Bridget S. Wilkins. Pitfalls in lymphoma pathology : avoiding errors in diagnosis of lymphoid tissues. *Journal of Clinical Pathology*, 64(6) :466–476, June 2011. ISSN 1472-4146. doi : 10.1136/jcp.2010.080846. 65

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1321–1330. JMLR. org, 2017. 66

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks. *arXiv preprint arXiv :1505.05424*, 2015. 66, 67, 68

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation : Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059, 2016. 66, 68, 69

Ian Osband. Risk versus uncertainty in deep learning : Bayes, bootstrap and the dangers of dropout. In *NIPS Workshop on Bayesian Deep Learning*, 2016. 68, 70

Christian Leibig, Vaneeda Allken, Murat Seçkin Ayhan, Philipp Berens, and Siegfried Wahl. Leveraging uncertainty information from deep neural networks for disease detection. *Scientific Reports*, 7(1) :17816, December 2017b. ISSN 2045-2322. doi : 10.1038/s41598-017-17876-z. 68

Nicolas Coudray, Paolo Santiago Ocampo, Theodore Sakellaropoulos, Navneet Narula, Matija Snuderl, David Fenyö, Andre L. Moreira, Narges Razavian, and Aristotelis Tsirigos. Classification and mutation prediction from non–small cell lung cancer histopathology images

using deep learning. *Nature Medicine*, 24(10) :1559, October 2018b. ISSN 1546-170X. doi : 10.1038/s41591-018-0177-5. URL https://www.nature.com/articles/s41591-018-0177-5. 70

Chapitre 3

Une approche plus symbolique de l'analyse des images histologiques

That's how it is with people.
Nobody cares how it works
as long as it works.

Councillor Hamann

Sommaire

3.1 Introduction

Dans ses aspects techniques, le chapitre qui précède traitait du fonctionnement des réseaux neuronaux convolutifs et les décrivait comme une forme élaborée de chaîne perceptuelle dont chacun des maillons était automatiquement paramétré pour contribuer de manière optimale à la prédiction attendue par le réseau. Une seconde partie, plus applicative, implémentait, étudiait et améliorait ces solutions pour analyser des images de lames entières afin de répondre à des problématiques biomédicales. Ces parties, et principalement celle sur le diagnostic des lymphomes, laissent pourtant entrevoir quelques limitations dans la démarche employée.

La « démarche » à laquelle nous faisons référence ici est la méthodologie quasi-systématique d'entraînement et d'application d'un réseau neuronal convolutif profond, en tant que classifieur, sur les tuiles d'une image numérisée. Sans écrire une diatribe à l'encontre de ce mode d'analyse, nous débutons tout de même ce chapitre par un constat qui fait planer un lourd sentiment d'insatisfaction sur les développements précédents. Il s'agit dans cet état d'esprit de mesurer le rapport entre l'énergie mobilisée et la banalité, ou disons plutôt l'intérêt opérationnel limité, du résultat obtenu.

Énergie mobilisée Comptons d'abord le temps d'étude clinique utilisé à rassembler les patients, leur accord de participation à l'étude et la constitution des jeux de données. Comptons ensuite le temps de récupération des lames des patients et leur numérisation. Comptons également le temps passé par le pathologiste expert à annoter les lames numérisées, qui est considérable notamment dans le cas des vaisseaux *HEVs*. Comptons encore le temps de conception, d'entraînement et d'évaluation des performances du modèle neuronal utilisé pour analyser les *patches*, ainsi que l'élaboration de stratégies d'agrégation d'information à l'échelle du patient. Comptons enfin les compétences qui doivent être rassemblées et doivent travailler de concert pour mener à bien ce genre d'expérience.

Faiblesse des résultats obtenus Notons dans un premier temps le caractère très spécifique de la décision, surtout dans le cas du lymphome, qui correspond souvent à un diagnostic différentiel très restreint et n'intervient que très tardivement et très rarement dans l'arbre décisionnel emprunté par les pathologistes. Notons ensuite que les cas précédemment abordés sont des cas simples, puisque la quantité d'annotations nécessaire à l'apprentissage était relativement faible pour une tâche de classification de ce genre et que la plupart des diagnostics nécessiteraient beaucoup plus de travail de la part des pathologistes. Notons enfin la difficulté de généralisation et de déploiement de la solution si durement obtenue. Sur ce dernier point, il est important d'insister sur le fait qu'il n'a rien de spécifique à notre cas d'usage, le lymphome, et n'est pas non plus imputable à une défaillance dans notre conception de l'outil d'analyse. Des difficultés similaires sont observées lorsque des sociétés autrement plus équipées et expertes de l'« intelligence artificielle » tentent de déployer des outils similaires en routine clinique [1].

Ce constat fait sérieusement douter de la viabilité de cette approche pourtant très explorée dans l'analyse des lames entières de pathologie, comme cela est décrit dans les travaux de Janowczyk and Madabhushi [2016]. Loin d'être gratuite, cette critique est une prise de recul vis-à-vis de la conception et du déploiement des solutions de *Deep Learning* appliquées à la pathologie numérique. Elle nous amène à identifier et à faire tomber des verrous d'ordre supérieur, qui ne reposent pas uniquement sur les arcanes de l'architecture ou de l'optimisation des réseaux neuronaux, mais sur la démarche d'analyse dans sa globalité lorsque des solutions d'apprentissage automatique sont mises en jeu. Elle conduit également à voir sous un jour

1. https://www.technologyreview.com/2020/04/27/1000658/google-medical-ai-accurate-lab-real-life-clinic-covid-diabetes-retina-disease/

nouveau les premiers algorithmes mis au point dans ces travaux de thèse qui, loin de l'état de l'art des solutions de segmentation d'images, illustrent pourtant parfaitement certains concepts-clefs qui animent l'écriture de ce chapitre.

Avec l'objectif de ré-équilibrer le rapport décrit ci-dessus, le présent chapitre propose des approches d'analyse plus générales dans lesquelles le *Deep Learning* occupe une place moins centrale et cède du terrain aux algorithmes de fouille et de structuration des données. Une première partie décrira ainsi des stratégies d'extraction et de structuration de données destinées à alléger considérablement la charge de l'annotation médicale des lames (Section 3.2). La seconde moitié de ce chapitre revisitera les travaux préliminaires de cette thèse sur les arbres de segmentation, qui ont fait l'objet de deux communications dans des conférences internationales. Ces travaux seront revisités sous l'angle de la détection *syntaxique* d'objets dans les images et illustreront notamment comment une connaissance sur la structure des objets peut orienter des méthodes de détection et de segmentation (Section 3.4).

3.2 Extraction et structuration de données dans les WSIs

En préservant le mode de fonctionnement de prédilection des CNNs, la plupart des questions cliniques pertinentes du domaine de l'anatomie et de la cytologie pathologique ont été abordées comme des problèmes de classification supervisée. Cependant, contrairement aux problèmes de classification traditionnels tels que les études menées par Bayramoglu et al. [2016], Bejnordi et al. [2017] et Saha et al. [2018], les dimensions spatiales des images de pathologie numérique ne permettent pas de les placer en entrée des réseaux de neurones. À l'instar des stratégies développées dans le cadre de la télédétection et des images satellitaires, la plupart des solutions prédictives sont développées pour la classification de patches qui, une fois prédits et repositionnés dans la lame, conduisent à une forme de segmentation sémantique, à partir de laquelle un résultat à l'échelle du patient peut être compilé comme le décrivent notamment Coudray et al. [2018].

Avec la volonté d'élargir le champ d'application de ces méthodes à plus de tissus et de pathologies, le nombre de structures et de lésions devant être reconnues par les algorithmes de classification supervisée ne cesse d'augmenter. Dans ce contexte, il n'est pas surprenant de voir les ensembles de données annotées, comme ceux présentés par Aresta et al. [2019] et Tambe et al. [2019], couvrir un pan toujours plus large du vocabulaire technique des pathologistes. Cette recherche de l'exhaustivité dans la reconnaissance des structures, plus qu'une quête de l'omniscience du système d'analyse, témoigne d'une tendance naturelle à vouloir établir un vocabulaire commun entre l'humain et la machine. Elle est intuitivement nécessaire à la production d'une décision automatique *explicable* et *interprétable* par l'utilisateur humain qui, n'en déplaise à ses principaux détracteurs que l'on a pu voir s'exprimer lors de la conférence *NIPS*, Wilson et al. [2017], demeurent des principes-clefs à respecter pour implémenter avec succès ces technologies dans un domaine aussi critique que la santé. Cette problématique a notamment déjà été largement décrite par les travaux de Kelly et al. [2019], Holzinger et al. [2017a] et Holzinger et al. [2017b].

Malheureusement, la classification, dans sa forme supervisée, s'adapte mal aux grands nombres de classes. Cela n'a évidemment rien à voir avec la nature même des modèles d'apprentissage statistique, puisque la multiplicité des classes réduit souvent la variabilité intra-classe des données d'apprentissage et accélère les procédures d'entraînement de ces modèles. Les difficultés surviennent plutôt lorsqu'il s'agit de faire annoter des quantités astronomiques d'images à des experts hautement qualifiés.

Pour passer outre cette limitation, certaines méthodologies d'analyse de lames histologiques reposent sur des briques de classification non-supervisée et parviennent à extraire des *motifs*, *classes* ou *concepts*, qui permettent de prédire les résultats d'un patient et trouvent du sens dans le langage des pathologistes, tels que les *clusters* extraits dans les procédures décrites par Yamamoto et al. [2019], Rajpoot et al. [2013] ou Cruz-Roa et al. [2011]. Cependant, bon nombre de ces implémentations apprennent des dictionnaires *plats*, c'est-à-dire sans structure hiérarchique, comptant un nombre fixé et arbitraire de mots de *vocabulaire* Yamamoto et al. [2019], Cruz-Roa et al. [2011]. Ce choix de représentation des connaissances, en plus d'être inadapté à la comparaison avec la connaissance de l'expert, intrinsèquement hiérarchique comme le décrivent Smith et al. [2007] et Roullier et al. [2010], induit le risque de masquer des concepts importants par moyennage excessif du signal, voir Figure 3.2 et Figure 3.3. Si l'on poursuit la lecture, le terme *explicable* est souvent employé dans les articles qui définissent ces méthodes et sert aussi bien à qualifier l'analyse, la connaissance ou les descripteurs extraits par le système. Alors que l'« intelligence artificielle explicable » (*explainable-AI*) devient un sujet de recherche en vogue, l'*explicabilité*, même pour des applications transversales de l'*IA*, ne devrait pas être revendiquée sans appui sur une définition claire, comme énoncée par Gilpin et al. [2018] et Rudin [2019], et des métriques d'évaluation rigoureuses pour lesquelles on trouve des pistes de définition dans les travaux de Nauck [2003] ou de Gilpin et al. [2018].

Dans ce travail nous décrivons une approche générale pour extraire des *motifs* pertinents dans des cohortes non-annotées de lames entières numérisées. Le système construit, selon des règles d'agglomération très simples, une représentation hiérarchique des concepts visuels statistiquement significatifs observés dans les lames, Sous-section 3.3.3. Il garde en mémoire la trace de tous les concepts rencontrés et susceptibles d'avoir un intérêt pour le pathologiste. Pour des raisons de complexité, l'algorithme de regroupement utilisé fonctionne sur une stratégie *single-linkage* qui a tendance à construire un très grand nombre de classes dont les effectifs sont déséquilibrés, Paragraphe 3.3.3.1. Une bonne partie des contributions techniques de cette section est ainsi dédiée à la description d'un algorithme destiné à compacter la représentation hiérarchique obtenue, Paragraphe 3.3.3.2. Bien que la structure soit conçue dans le but de maximiser sa correspondance avec la connaissance humaine, nous faisons l'hypothèse qu'un recoupement parfait des concepts est inatteignable. Nous proposons donc de calculer une forme intuitive de *fossé sémantique* entre deux représentations qui peut être utilisée comme un indicateur d'*interprétabilité*, Paragraphe 3.3.4.1. Nous ajoutons à cela la création d'une application web pour visualiser la structure des connaissances de la machine, qui permet également de « combler » ce *fossé sémantique* en énonçant des règles de traduction basées sur des opérateurs logiques, Paragraphe 3.3.4.2. Enfin, la pertinence du dispositif comme un auxiliaire général d'analyse est soulignée au travers du développement concret d'un outil de gradation automatique des tumeurs du sein, Sous-section 3.3.5.

3.2.1 Motivations et principes fondateurs

Le cas d'usage ciblé est celui d'un projet d'apprentissage automatique basé sur un ensemble conséquent de lames numérisées *non-annotées*. Entendons par *non-annotées* des images dans lesquelles aucune région, zone tumorale ou tout autre type de lésion d'intérêt, n'a encore été détourée par un expert. Plusieurs informations sur le suivi des patients sont néanmoins à disposition et le système doit bien évidemment apprendre à prédire certaines d'entre elles à partir des images.

Sans perte de généralité, considérons la tâche de prédiction du diagnostic du patient à partir de sa *WSI*. Dans une approche supervisée, tous les *patches* des *WSIs* d'entraînement sont *étiquetés* avec la pathologie connue du patient. Cependant, à moins que le *motif discriminant* de la maladie soit observable en tout point de la lame, une très grande majorité des imagettes

extraites seront dépourvues de l'information nécessaire au système pour prendre sa décision. Dans ces conditions, il est peu probable que le modèle puisse véritablement établir une séparation entre les différents diagnostics.

Devant cette situation, deux options s'offrent alors à l'analyste en charge du projet. La première consiste à demander aux experts des pathologies concernées de passer des heures à segmenter manuellement des régions d'intérêt dans les lames. En échantillonnant les *patches* d'apprentissage uniquement dans ces zones, le modèle disposera de l'information nécessaire à la décision dans chacun des échantillons d'entraînement et l'apprentissage fonctionnera sans problème. La seconde alternative est d'utiliser une stratégie de classification non-supervisée pour regrouper dans des *clusters* les *patches* avec des aspects visuels similaires. Certains de ces *clusters* pourront ensuite être identifiés, automatiquement ou manuellement, comme porteurs de l'information nécessaire à la classification attendue. Une solution d'apprentissage supervisée peut enfin être entraînée à séparer les différentes pathologies uniquement sur la base des *clusters* identifiés.

Cette deuxième approche n'est pas seulement bien plus élégante d'un point de vue de l'extraction de données, mais elle est surtout considérablement économe en temps d'annotation experte et apporte une information de description précieuse pour la suite de l'analyse. La machine pourrait avoir isolé des structures, encore inexplorées ou perceptuellement inaccessibles aux pathologistes, susceptibles de corréler avec le pronostic ou la réponse des patients à une thérapie.

Bien entendu, l'extraction de *motifs* pertinents, aussi appelée *fouille*, dans des banques d'images reste un problème ouvert, tout particulièrement lorsqu'il s'agit de *WSIs*. Nous basons notre travail dans ce domaine sur trois principes fondamentaux qui aspirent à garantir de bonnes propriétés sur les concepts extraits. La persistance du concept dans l'*espace caractéristique* est la propriété pincipale. Elle est d'ailleurs le critère d'optimisation le plus répandu dans les méthodes de *clustering* classiques, dans lesquelles un *motif* est défini comme une région de forte densité dans l'*espace caractéristique*. Tirant avantage de l'analyse orientée *patches* des *WSIs*, un autre principe fondamental utilise une connaissance *a priori* bien connue de la communauté du traitement des images : la persistance spatiale d'un concept. Deux pixels voisins, ou *patches* dans notre cas, appartiennent vraisemblablement au même objet et une structure significative va très probablement couvrir une large composante connexe de *patches*. Enfin, un indicateur important de généralisabilité des concepts assimilés est leur persistance d'un patient à un autre. Ce principe s'assure qu'un concept observé sur une *WSI* d'un patient est également observable dans les *WSIs* de plusieurs autres.

3.2.2 Etat de l'art de la fouille dans les images

3.2.2.1 Extraction de caractéristiques sans supervision

Le point de départ de la méthode est une bonne représentation des *patches* de lames hisologiques pour un tissu donné. Comme nous l'avons déjà évoqué dans le premier chapitre, la *représentation* est un encodage d'image qui rend plus aisée toute tâche de prédiction automatique sur les images considérées. C'est précisément sur ce point, nous l'avons vu, que le *Deep Learning* vient se positionner au-dessus des autres méthodes d'analyse d'images. Il remplace les conceptions astucieuses de filtres, dont les méthodes de Jain and Farrokhnia [1990] et Lowe [1999a] sont des exemples magistraux, et les structures d'intégration ingénieuses, comme la détection des coins de Harris [1954], par de vastes champs de paramètres automatiquement optimisés sur les données brutes. Il est capable de produire des *représentations* arbitrairement complexes et abstraites des données étudiées. Dans cette partie, contrairement aux applications déjà explorées, l'apprentissage de ces modèles n'est envisagé que sous l'angle de la description

des images, et non de la classification supervisée. Nous allons voir que plusieurs approches non-supervisées ont été développées pour entraîner des réseaux convolutifs profonds dans cette perspective.

Les modèles génératifs Ces modèles capturent la topologie des *représentations* qu'ils construisent. Ils approximent la distribution des données dans l'*espace caractéristique* et en tirent une mesure pertinente de similarité entre les échantillons. Lorsqu'il s'agit d'utiliser un algorithme de *clustering* dans le processus d'analyse, le recours préalable à un modèle génératif apparaît donc comme un prérequis important. Il est cependant rarement garanti dans les travaux de ce genre, à l'image de Janowczyk et al. [2017] et Hou et al. [2019], où des *auto-encodeurs* standards sont souvent utilisés (aussi relevés exhaustivement dans les travaux de Raza and Singh [2018]). Pourtant, l'*apprentissage de métriques* couvre aussi bien le cas d'usage de l'extraction non-supervisée de descripteurs, via l'entraînement d'*auto-encodeurs variationnels* et en suivant l'exemple de Kingma and Welling [2014] ou de Rezende et al. [2014], que celui de la classification supervisée, avec l'exemple des *réseaux siamois* entraînés selon la fonction de coût de *contraste* ou du *triplet* développées par Chopra et al. [2005] et Weinberger and Saul [2009]. Toutefois, ces techniques présentent souvent un surcoût de mise en place : elles imposent bien souvent des architectures et procédures d'entraînement plus lourdes. Le cadre de travail étudié par Wojke and Bewley [2018] constitue désormais une solution à ce problème. Les auteurs font remarquer qu'en normalisant la sortie et les poids de la dernière couche, ainsi qu'en supprimant le biais de cette couche, toute architecture entraînée selon une démarche de classification supervisée classique, est en mesure d'apprendre une métrique sur l'*espace caractéristique* construit par le réseau.

Transfert de caractéristiques multi-usages Le transfert des poids d'un classifeur très général pour initialiser un modèle destiné à une nouvelle tâche plus spécifique peut drastiquement raccourcir le temps d'apprentissage du nouveau modèle. Cette technique, si efficace et simple à mettre en œuvre, est désormais la brique principale de la plupart des applications les plus populaires de vision par ordinateur. Les modèles développés par Carreira et al. [2016], Chen et al. [2017], Ren et al. [2015] ou encore Weinzaepfel et al. [2013] sont parmi les exemples les plus populaires. Originellement dédiées à la classification supervisée, certaines approches de pointe du *transfer-learning* étendent à présent ces stratégies d'adaptation à un domaine spécifique à travers des algorithmes d'apprentissage dits *auto-supervisés*. Ces techniques définissent des *tâches auxiliaires* telles que deviner la couleur de pixels manquants dans les images du domaine spécifique cible ainsi que le proposent Pathak et al. [2016], ou apprendre à un classifieur à retrouver les classes identifiées par un algorithme des *k-moyennes* comme Caron et al. [2018a], Fan et al. [2018] l'ont déjà envisagé. Les images encodées par le classifieur ou le réseau générateur de pixels manquants sont alors souvent des descripteurs généraux très pertinents obtenus sans le moindre effort d'étiquetage manuel.

Nous proposons ici d'obtenir rapidement, et sans effort d'annotation, une *représentation* des images histologiques qui soit propice au clustering, en mettant en place un algorithme d'*apprentissage de métrique auto-supervisé*. La procédure d'entraînement est identique à celle décrite par Fan et al. [2018], mais dispose en plus d'une dernière couche neuronale re-paramétrée à la façon de Wojke and Bewley [2018].

3.2.2.2 Persistance spatiale

Afin d'extraire des concepts d'empreinte spatiale significative dans les images, l'utilisation de méthodes de segmentation non-supervisées est inévitable. Cette thématique a déjà été explorée dans le premier chapitre, Sous-section 1.2.1, et la plupart de ces techniques, à l'instar de Bejnordi et al. [2015], de Albayrak and Bilgin [2019] ou encore de Fouad et al. [2017],

ont notamment été adaptées pour segmenter efficacement des *WSIs*. Cependant, la plupart de ces méthodes opèrent sur des pixels et doivent, pour des raisons d'occupation mémoire, faire le choix de s'exécuter à faible grossissement ou de remplacer les traditionnels pixels par de larges *patches* extraits selon une grille régulière dans la lame. L'un des objectifs de cette segmentation étant d'obtenir des descripteurs raffinés, c'est-à-dire calculés par des *CNN* et qui intègrent beaucoup d'informations de contexte, nous choisissons naturellement de nous placer dans le second cas.

Segmentation sémantique La forme de segmentation la plus répandue sur les *WSIs* est la segmentation sémantique. Les outils de partitionnement d'images conçus par Xu et al. [2014, 2017] et Coudray et al. [2018] n'en sont que les exemples les plus plebiscités, au centre d'une littérature particulièrement étendue sur le sujet. Comme évoqué dans le premier chapitre à l'échelle du pixel, elle consiste, dans le cas d'une grille de *patches*, à classer chacun des *patches*. Un ensemble de *patches* connectés et de même classe constitue alors un segment. Généralement traitée comme un algorithme de classification supervisée, Thomas et al. [2010], Khan et al. [2013] et Fouad et al. [2017] ont toutefois montré que ce mode de segmentation s'adapte parfaitement à l'étiquetage non-supervisé des *patches*. Cependant, il ne trouve dans ce cas qu'une utilité pour la visualisation, puisque la position des imagettes et leurs relations de voisinnage ne sont jamais exploitées pour déterminer la classe des *patches*.

Segmentation par super-pixels Définis comme des segmentations *à sémantique faible* dans le premier chapitre, les super-pixels, et particulièrement l'algorithme *SLIC* proposé par Achanta et al. [2012], sont régulièrement utilisés pour segmenter des *WSIs* ainsi que le montrent Bejnordi et al. [2015], Albayrak and Bilgin [2019] et Fouad et al. [2017]. *SLIC* présente par exemple l'avantage d'être récursif et peut ainsi être utilisé à différents niveaux de grossissement, ce qui permet par exemple à Bejnordi et al. [2015], travaux les plus aboutis dans le domaine, de segmenter plus ou moins finement une zone selon son intérêt.

Segmentation orientée graphes Ces méthodes ont beaucoup été traitée, souvent implicitement dans le premier chapitre, parce qu'elles tiennent facilement compte de la hiérarchie de contenance des segments dans une image. Elles font donc l'objet d'un intérêt particulier dans le domaine de la segmentation des *WSIs* car elles permettent d'imiter fidèlement les processus de détection et de classification des pathologistes : en exploitant les relations de contenance entre différents niveaux de résolution, les zones suspectes sont détectées aux niveaux les plus grossiers, puis confirmées ou infirmées en inspectant localement les niveaux plus fins. Cette méthode d'analyse épargne ainsi la complexité algorithmique des méthodes de recherche exhaustive telle que la fenêtre glissante fortement résolue sur la lame. Au-delà de l'application à la segmentation des *WSIs*, les approches orientées graphes offrent une large variété d'algorithmes de faibles complexités en temps et qui optimisent un grand nombre de fonctions de coût de segmentation. Sharma et al. [2015] identifient ainsi dans les graphes des approches très prometteuses pour appréhender l'analyse des *WSIs*. Nou notons de plus que leur formulation est adaptée à la description de lames par *patches*, où le positionnement et la description dans l'espace des caractéristiques prennent très naturellement la forme de nœuds et d'arêtes pondérées. L'ensemble de ces raisons, doublé d'une certaine affection pour ces structures de données, orientent naturellement le choix de notre méthode de segmentation vers ce type de solutions.

3.2.2.3 Evaluer et combler le fossé sémantique

Le terme *fossé sémantique* n'est pas encore apparu dans la littérature associée à l'analyse des *WSIs*. Il est généralement utilisé par la communauté de la requête générale d'images, dans laquelle les études menées par Djeraba [2003], Hare et al. [2006] ou Ma et al. [2010] ont pu se distinguer, qui interroge de vastes banques d'images par le biais d'expressions formulées en langage naturel ou avec des images proches de celles recherchées. Une bonne partie de ce domaine se concentre sur la requête d'images annotées, ce qui, comme l'illustrent notamment Ma et al. [2010], constitue déjà une tâche difficile, mais ce travail se rapproche d'avantage des techniques de sélection d'images sur la base du contenu visuel. « Combler » le *fossé sémantique*, dans ce contexte, consiste à relier la *représentation* d'une image, formulée par la machine, avec des concepts abstraits de la connaissance humaine.

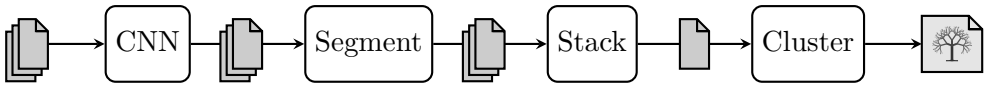
Traduction automatique Parmi les techniques de requête d'images, notre travail s'inscrit parfaitement dans le paradigme de la *traduction automatique*, dans lequel tous les *mots* de vocabulaire de la machine sont appris sans supervision et le traducteur prend la forme d'un *lexique* appris statistiquement à partir d'annotations humaines. Cet aspect est d'ailleurs particulièrement détaillé dans les travaux de Duygulu et al. [2002].

Fossé sémantique À notre connaissance, combler le *fossé sémantique* dans le cadre de ces études revient toujours à la formulation et à la résolution d'un problème de classification supervisée dans lequel le *lexique* relève statistiquement des corrélations existantes entre des *mots visuels* assimilés par la machine et des termes du langage humain. Nous proposons un principe d'association simplifié qui formule des traducteurs uniquement par l'intermédiaire d'opérateurs logiques sur les ensembles d'images. Avec la présence de données annotées encore une fois, Djeraba [2003], Hare et al. [2006] ou encore Ma et al. [2010] évaluent le *fossé sémantique* par le calcul des métriques de classification traditionnels : précision, rappel, F-scores. Dans ce travail, l'estimation du *fossé sémantique* est fournie par la procédure d'annotation elle-même : comme l'utilisateur formule explicitement les règles de traduction, avec très peu de marge laissée pour la subjectivité, le *fossé* est précisément quantifié en terme de complexité de formulation des règles.

3.2.3 Une méthodologie générale

Un *encodeur*, qui transforme les images en *vecteurs caractéristiques* de faible dimension, est d'abord obtenu. Ensuite, toutes les images sont traitées de la même façon : la lame est tuilée et chaque tuile est *encodée* dans la représentation apprise à l'étape précédente. Les *patches* de la lame sont alors regroupés en segments, décrits à leur tour par le vecteur moyen de leur *patches* constitutifs. Tous les segments rencontrés dans toutes les *WSIs* de la cohorte étudiée sont alors utilisés comme des *concepts-feuilles* de la représentation de la machine. Le reste de l'arborescence est finalement construit par un regroupement hiérarchique des *feuilles* en utilisant la distance euclidienne entre les *vecteurs caractéristiques* comme mesure de similarité entre deux *clusters*.

Un diagramme complet de l'architecture du système est fourni dans la Figure 3.1. La solution dans son intégralité est pensée comme une adaptation du paradigme d'analyse *Object Based Image Analysis OBIA*, emprunté à Hay and Castilla [2008] et Blaschke et al. [2014] qui l'ont formalisé précisémment pour la communauté de la télédétection. Tandis que la section suivante justifie et décrit en détail l'implémentation de chaque brique de la solution, la méthode se veut une démarche très générale et ne se restreint pas à une combinaison spécifique de *représentation*, de segmentation, de classification ou de *clustering*.



(a) {slide}.mrxs (b) {slide}_desc.csv (c) {slide}_seg.csv (d) f_space.csv (e) knowledge.json

FIGURE 3.1 – Diagramme global de l'analyse. L'entrée de la méthode est un large ensemble de lames numérisées (a). Pour chaque lame, nous récupérons la position des patches, ainsi que leur description dans l'espace caractéristique d'un *CNN* (b). Les couples position-description forment naturellement des nœuds, des arêtes pondérées, et une segmentation de graphe est perpétrée pour chaque fichier de description (c). Chaque segment de lame est ensuite décrit par le vecteur caractéristique moyen calculé sur ses patches constitutifs et les segments décrits sont identifés et rassemblés dans un fichier unique (d). Enfin, un regroupement hiérarchique est réalisé sur cet ensemble de segments décrits dans l'espace caractéristique (e).

3.3 Extraction de concepts visuels

3.3.1 Apprentissage de représentations

3.3.1.1 Apprentissage d'une métrique

Lorsqu'il est question de *transfer-learning* et d'adaptation à un domaine spécifique, le fonctionnement d'un classifieur de type *CNN* est souvent scindé en deux étapes. La première, que l'on retrouve récemment sous l'appellation *embeding*, pour signifier le fait que l'image est « projetée » dans un *espace caractéristique*, transforme une image $x \in \mathbb{R}^{s\times s\times 3}$ en un vecteur de descripteurs $f \in \mathbb{R}^n$. La seconde étape est un partitionnement de l'*espace caractéristique* en *concepts*, et cette classification est généralement opérée par un *perceptron* (voir Sous-section 2.2.3).

Soit k le nombre de concepts devant être distingués par le classifieur et y, une variable aléatoire, prédiction réalisée par le modèle sur l'image x. La couche de sortie du réseau compte donc également k neurones que l'on indicera par $c \in [1, \ldots, k]$, chacun responsable d'encoder l'appartenance de x à sa classe c.

Suivant le schéma d'une couche `Dense` (voir Sous-section 2.2.3), chaque neurone c encode un *motif*, $w_c \in \mathbb{R}^n$, qu'il est en charge de reconnaître dans le signal f. En notant b_c le *biais* de ce neurone, l'affinité, ou *logit* z_c, de ce neurone pour son entrée f, ou pour l'entrée du *CNN* x, est exprimée comme suit :

$$z_c = w_c f + b_c$$

Une fois que chacun des k neurones de sortie a exprimé son affinité vis-à-vis du *vecteur caractéristique*, la fonction *softmax*, notée $\sigma(.)$ et déjà définie à l'Equation 2.8, appliquée au vecteur $\boldsymbol{z}$ des *logits*, fournit une probabilité d'appartenance de l'image à chacun des k concepts assimilés :

$$\mathcal{P}(y = c|x) = \sigma(\boldsymbol{z})_c$$

Dans le cas d'un apprentissage classique, entendons *discriminant*, le réseau est entraîné à produire des *vecteurs caractéristiques* d'une classe c_i qui soient les plus éloignés, en terme de distance euclidienne par exemple, de ceux des autres classes $\{c_j, \forall j \neq i\}$. Cette logique d'apprentissage interdit généralement la nuance. Ainsi, une image contenant l'information caractéristique de plusieurs classes bénéficiera tout de même d'une représentation « tranchée », c'est-à-dire qui favorisera une des classes observée, qui ne laissera pas douter de la présence d'autres motifs dans l'image. C'est le cas de l'apprentissage du perceptron par descente du gradient de la fonction de *cross-entropy* entre la prédiction $\sigma(\boldsymbol{z})$ du réseau et la vérité annotée par l'expert des images (voir la Sous-section 1.4.4 et la Section 2.2 pour plus de précisions).

Il est cependant possible d'entraîner un réseaux de neurones classifieur à former un *espace caractéristique* propice à la nuance. Ces modèles sont alors qualifiés de *génératifs* (voir la Sous-section 1.2.2 pour une explication plus détaillée) et donne une information numérique claire de la proximité, toujours en terme de distance euclidienne sur l'*espace caractéristique* dans la plupart des cas, d'un individu à toutes les classes.

Nous employons ici la méthode de Wojke and Bewley [2018] qui propose une re-paramétrisation de la prédiction, $z_c^{'} = w_c^{'} f^{'}$, basée sur la normalisation des vecteurs de poids de chaque classe, et la normalisation des vecteurs de représentation prédits :

$$f^{'} = \frac{f}{\|f\|}$$
$$\forall c,\ w_c^{'} = \frac{w_c}{\|w_c\|}$$

Dans leurs travaux, ils montrent rigoureusement qu'avec une descente classique du gradient de la fonction de *cross-entropy*, l'apprentissage conduit ici à un *espace caractéristique* doté d'une métrique de similarité entre les individus. En effet, ce paramétrage garantit, en plus de la distinction entre les classes par éloignement de leurs vecteurs caractéristiques, un rapprochement des individus d'une même classe autour d'un centroïde. Ils rappellent également que, dans un *espace caractéristique* restreint à la sphère-unité, la distance euclidienne et la distance cosinus entre deux vecteurs sont directement proportionnelles. Par la suite, nous ferons souvent référence à la *similarité* ou la *dissimilarité* entre les individus et les deux distances dans ce cas seront interchangeables.

3.3.1.2 Apprentissage « auto-supervisé »

L'adaptation d'une architecture existante et performante à des données qu'elle ne connait pas sous-entend généralement un ré-apprentissage, souvent appelé *fine-tuning*, c'est-à-dire une re-paramétrisation du modèle. Cette modalité est compatible avec les architectures dédiées à la classification qui se terminent par un perceptron et doivent être entraînées à réaliser des tâches de classification supervisée uniquement.

Cette configuration d'adaptation semble incompatible avec l'absence d'annotations qui caractérise pourtant le positionnement de notre problème. Fan et al. [2018] et Caron et al. [2018b] se placent précisément dans ce contexte et proposent une stratégie itérative d'*auto-supervision* du classifieur. Pour cela, ils considèrent la fraction $\boldsymbol{x}$ des images du domaine-cible dédiée à l'adaptation d'un modèle neuronal pré-entraîné sur une tâche de classification générale. Dans ce cas, nous notons $\boldsymbol{f}_0$ l'ensemble des représentations fournies par ce modèle pour l'ensemble des images dédiées à l'adaptation. La méthode propose alors de réaliser un partitionnement non-supervisé de l'*espace caractéristique* $\boldsymbol{f}_0$ en un nombre arbitraire, k, de *clusters*. Ce *clustering* sera alors noté $\mathcal{C}_0$ pour reprendre les notations de la Sous-section 1.4.1.

Après initialisation, la méthode propose de constituer une nouvelle représentation $\boldsymbol{f}_n$ en réalisant un apprentissage supervisé du modèle dans lequel la vérité terrain, ou *ground truth*, est donnée par $\mathcal{C}_{n-1}$. Autrement dit, le modèle est optimisé par descente du gradient de la *cross-entropy* entre ses prédictions et les classes définies par $\mathcal{C}_{n-1}$. Le nouveau partitionnement, $\mathcal{C}_n$ est alors obtenue en exécutant une nouvelle procédure de partitionnement non-supervisé sur le nouvel espace caractéristique constitué $\boldsymbol{f}_n$. Ce partitionnement est généralement réalisé *via* un algorithme des *k-moyennes* sur $\boldsymbol{f}_n$. Avant de procéder à ce *clustering*, les *outliers*, c'est-à-dire les individus de $\boldsymbol{f}_n$ les plus difficiles à classer, sont retirés de l'*espace caractéristiques*. Le nombre d'*outliers* diminue au cours des itérations de la méthode et un critère d'arrêt relativement arbitraire est placé sur leur effectif.

3.3.2 Persistance spatiale des concepts

Suivant les préceptes de l'analyse orientée *patches*, l'image entière est découpée en un ensemble d'imagettes centrées sur les positions $\boldsymbol{P} = \{(x_1, y_1), \ldots, (x_K, y_K)\}$. Toute image centrée sur le point $p_i \in \boldsymbol{P}$ est décrite par un vecteur de taille M noté f_i de l'*espace caractéristique* assimilé précédemment. La fouille de motifs dans l'*espace caractéristique* utilise généralement chaque *patch* relevé dans les lames de la cohorte comme une donnée d'entraînement pour l'algorithme des *k-moyennes*. Nous pensons que l'information de position du *patch* peut contribuer à construire des clusters plus pertinents.

Lorsque l'on compare le caractère très général des représentations apprises par les modèles non-supervisés à la pluralité des structures et motifs pouvant être distingués dans les *patches*, il semble raisonnable de penser le *vecteur caractéristique* d'une imagette isolée comme une information bruitée. Dès lors, la moyenne des vecteurs descripteurs de plusieurs *patches* appartenant à la même structure, c'est-à-dire rattachés au même concept dans le langage du pathologiste, offre un signal de reconnaissance beaucoup plus fiable. Par conséquent, et suivant un *a priori* bien connu en segmentation sémantique ainsi qu'en traitement du signal, nous proposons d'initialiser une procédure de clustering hiérarchique avec des descripteurs moyennés sur de petits regroupements spatiaux de *patches* (segments de la lame).

Regrouper des pixels, ou plutôt des *patches* dans notre cas, en segments homogènes de taille significative fait naturellement appel à des algorithmes de segmentation non-supervisés, auxquels nous faisions référence sous la dénomination *segmentations à contrainte sémantique faible* dans la Sous-section 1.2.1. Nous rappelons que dans la formulation de ce genre de problèmes de segmentation, l'idée n'est pas de détourer exactement les objets d'intérêt présents dans l'image, mais d'extraire des segments *relativement* homogènes aux contours *relativement* réguliers. Alors que beaucoup de ces méthodes sont bien étudiées pour les tableaux denses de pixels, comme cela est le cas pour les procédures de *super-pixels* de Achanta et al. [2012] et de Machairas et al. [2015], les *arbres couvrants* construits selon la méthode de Kruskal, et notamment sa version déclinée dans les travaux de Felzenszwalb and Huttenlocher [2004] s'adaptent bien à la représentation creuse que nous avons adoptée pour définir une lame entière : $\mathcal{S} = \{(p_1, \boldsymbol{f}_1), \ldots, (p_K, \boldsymbol{f}_K)\}$, et pour laquelle les approches orientées graphes nous semblent bien optimisées.

Définissons $N(p_i)$, le voisinage d'un *patch*, comme l'ensemble des *patches* adjacents à p_i. Soit alors $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ le graphe d'adjacence des *patches* défini sur la lame $\mathcal{S}$. Chaque noeud $v \in \boldsymbol{V}$ de $\boldsymbol{G}$ est un élément de $\boldsymbol{P}$, autrement dit $\boldsymbol{V} = \boldsymbol{P}$ et chaque arête correspond à un couple de *patches* adjacents, autrement dit $\forall e \in \boldsymbol{E}$, $e = (p_i, p_j) \Leftrightarrow p_j \in N(p_i) \wedge p_i \in N(p_j)$. La méthode de Kruskal construit alors un arbre couvrant sur le graphe $\boldsymbol{G}$ qui minimise la somme des poids w, attribués à chaque arête $e \in \boldsymbol{E}$ dans le graphe. Par conséquent, afin de débruiter l'*espace caractéristique*, w est simplement défini par la métrique apprise par le modèle sur la représentation. Cette dernière peut, dans notre cas (voir la Sous-section 3.3.1), aussi bien faire référence à la distance euclidienne, $w(p_i, p_j) = \|\boldsymbol{f}_i - \boldsymbol{f}_j\|$, qu'à la distance cosinus, $w(p_i, p_j) = 1 - \langle \boldsymbol{f}_i, \boldsymbol{f}_j \rangle$. Cette mesure de similarité encourage ainsi le regroupement de *patches* dont les aspects visuels sont proches pour le modèle d'extraction de caractéristiques entraîné précédemment.

L'Algorithme 2 de fusion de régions suivant le critère de Kruskal regroupe progressivement tous les segments d'une image jusqu'à la constitution d'un segment unique qui couvre toute l'image. Il est tout de même possible de produire une partition plus fine en ajoutant un critère d'arrêt à la procédure de fusion, comme cela a notamment été proposé par Felzenszwalb and Huttenlocher [2004]. Tout sous-ensemble connecté de *patches*, $s = \bigcup_i p_i \subseteq \boldsymbol{V}$, est appelé région ou segment de l'image et en considérant le sous-ensemble d'arêtes correspondantes, $\boldsymbol{E}_s \subseteq \boldsymbol{E}$, nous définissons l'énergie interne d'un segment comme la dissimilarité maximale rencontrée

Algorithme 2 : Algorithme de fusion de régions suivant Kruskal

Données : $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$; `// Le graphe pondéré`
Résultat : $parents$; `// Un dictionnaire de parenté`

$parents \leftarrow \{\}$;
$id \leftarrow taille(\boldsymbol{E})$;
$\mathcal{U} \leftarrow$ union-find$(\boldsymbol{V})$; `// Union-find initialisé avec les nœuds`
$\boldsymbol{E} \leftarrow tri(\boldsymbol{E})$; `// Arêtes par similarité décroissante`

pour chaque $e \in \boldsymbol{E}$ **faire**
  $n_1, n_2 \leftarrow e$;
  $r_1 \leftarrow \mathcal{U}.racine(n_1)$;
  $r_2 \leftarrow \mathcal{U}.racine(n_2)$;
  si $r_1 \neq r_2$ **alors**
    $parents(r_1) \leftarrow id$;
    $parents(r_2) \leftarrow id$;
    $\mathcal{U}.ré\text{-}enracine(r_1, id)$;
    $\mathcal{U}.ré\text{-}enracine(r_2, id)$;
    $id \leftarrow id + 1$;
  fin
fin

dans $\boldsymbol{E}_s$. Le critère de Kruskal fusionne en priorité les régions les plus similaires. Soient s_1 et s_2 les deux régions devant être fusionnées à une itération quelconque de la méthode par l'intermédiaire de l'arête $e = (p_i, p_j)$ avec $p_i \in s_1$ et $p_j \in s_2$. Le *coût* ou *énergie de fusion* de ces deux composantes corresond au poids de l'arête qui les relie, $Fuse(s_1, s_2) = w(e)$ et la fusion de s_1 et s_2 n'est ainsi permise que si le *coût* est faible par rapport à l'énergie interne minimale :

$$Fuse(s_1, s_2) \leq \min_{1,2}(Int(s_i) + \frac{\tau}{|s_i|}) \tag{3.1}$$

Le second terme dans la partie droite de cette équation correspond à la politique de fusion. Il impose un critère d'arrêt qui, dans la formulation ci-dessus, repose uniquement sur la taille des régions : les composantes les plus petites, jugées instables car leur empreinte spatiale est trop faible, ont de plus grandes chances de fusionner que les composantes plus larges. Le facteur d'échelle, noté τ, est un hyper-paramètre de la méthode qui place une préférence sur la taille des composantes finales. Il est bon ici de rappeler qu'une faible valeur de τ conduira à un grand nombre de segments et, *de facto*, un très grand nombre de concepts-feuilles dans la structure hiérarchique de connaissances. Bien que nous conseillions fortement au lecteur de garder cette valeur relativement basse, le véritable impact de τ n'est pas exploré en profondeur dans ce travail. Voir les travaux de Felzenszwalb and Huttenlocher [2004] pour plus de précisions.

Finalement, nous décrirons un segment en moyennant le vecteur descripteur de ses *patches* constitutifs et définirons les concepts-feuilles, c'est-à-dire les mots élémentaires du vocabulaire de la machine, en recueillant l'ensemble des vecteurs descripteurs de tous les segments de toutes les *WSIs* de notre jeu de données : $\mathcal{L} = \{\boldsymbol{f}_1, \ldots, \boldsymbol{f}_N\}$.

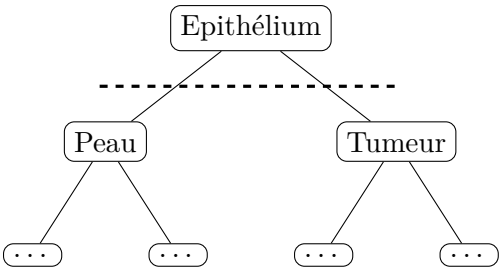
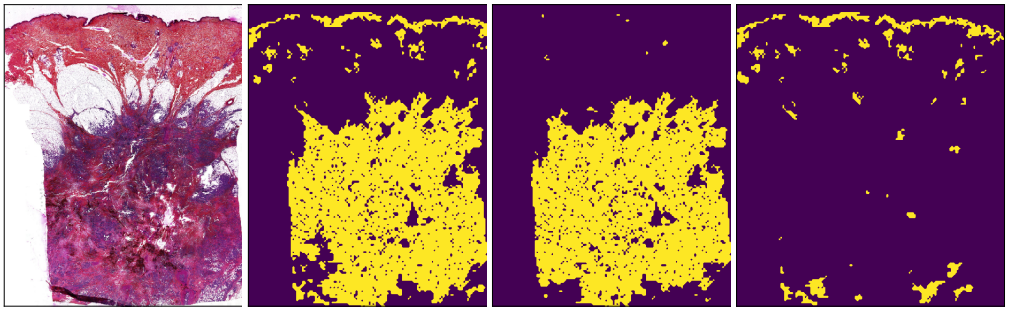


FIGURE 3.2 – Choisir arbitrairement le nombre de classes pour un regroupement « plat », peut aboutir à un partitionnement sémantique trop grossier de l'espace caractéristique, ligne en **pointillés**, voir la Figure 3.3 pour illustration.



(a) originale (b) épithélium (c) tumeur (d) peau

FIGURE 3.3 – (a) Miniature d'une lame de cancer du sein. (b) Masque correspondant à un concept trouvé par la machine : les patches appartenant à ce concept sont en sur-brillance. (c) et (d) Masques des « enfants directs » du concept (b) dans le regroupement hiérarchique. Si des sous-types intéressants d'un concept existent, le clustering hiérarchique les aura probablement mémorisés dans une étape antérieure.

3.3.3 Clustering et hiérarchie de subsomption

Comme déjà évoqué précédemment, le partitionnement de l'*espace des caractéristiques* devrait être hiérarchique afin de détecter chaque concept potentiellement pertinent et favoriser les comparaisons avec les connaissances humaines, voir Figure 3.2 et Figure 3.3. En tant qu'algorithme de *clustering* hiérarchique, la fusion de régions selon le critère de Kruskal apparaît encore comme une solution adaptée. Bien sûr, au-delà de son élégance, la procédure présente le double avantage d'être simple à implémenter et d'être extrêmement rapide à s'exécuter du fait de sa faible complexité algorithmique. De plus, utiliser un algorithme unique pour la segmentation d'image et le partitionnement de l'*espace des caractéristiques* économise de nombreux efforts de développement et homogénéise la structure des données manipulées le long de la chaîne de traitement.

Malheureusement, la notion de voisinage dans l'ensemble des concepts-feuilles est loin d'être aussi naturelle qu'entre les pixels, ou du moins les *patches*, d'une image. Une recherche des *k plus proches voisins* dans $\mathcal{L}$ devient donc nécessaire pour retrouver une structure de graphe dans l'*espace des caractéristiques* avant de pouvoir procéder aux fusions de régions selon Kruskal. Néanmoins, doter l'espace des descripteurs d'une structure de graphe reste une opération indolore et les *arbres-kd*, parmi d'autres structures, construisent très efficacement des graphes des *k plus proches voisins* et sont désormais implémentés dans la plupart des bibliothèques de manipulation de graphes [2].

2. https://scikit-learn.org/stable/modules/classes.html#module-sklearn.neighbors

3.3.3.1 Arbre de subsomption binaire

De manière similaire à la Sous-section 3.3.2, soit $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ le graphe des *k plus proches voisins* dont les nœuds sont des points de l'*espace des caractéristiques*. $v \in \boldsymbol{V}$ dans $\boldsymbol{G}$ est un élément de $\mathcal{L}$, autrement dit $\boldsymbol{V} = \mathcal{L}$ et chaque arête correspond à un couple de *plus proches voisins* pour lesquels un poids de dissimilarité, $w(\boldsymbol{f}_i, \boldsymbol{f}_j)$, est calculé par distance euclidienne ou cosinus.

La procédure de clustering, contrairement à la recherche de segments dans les lames, ne tronque pas la hiérarchie puisqu'elle a pour objectif de conserver tous les concepts d'intérêt. Elle peut donc exécuter l'algorithme de fusion de régions sur $\boldsymbol{G}$ sans application de critère d'arrêt. La seule contrainte ici est de stocker l'ensemble $\boldsymbol{C}$ de toutes les entités fusionnées au cours de la procédure dans une structure arborescente. Tout concept dans l'arbre est défini par un indice unique $l \in \boldsymbol{C}$ et un vecteur descripteur $\boldsymbol{f}_l$. La structure arborescente est obtenue en mémorisant, pour un concept donné, une référence sur son *parent* et ses deux *enfants*. Nous ré-utiliserons d'ailleurs avantageusement les notations employées dans la Sous-section 1.2.4 où les fonctions $\Uparrow(.)$, $\Downarrow(.)$, $\Leftrightarrow(.)$, retournent respectivement le *parent*, la paire d'*enfants* et le *frère* d'un concept. Plus spécifiquement, soient i et j, deux concepts devant être fusionnés selon le critère de Kruskal, et l le concept résultant de leur fusion. Nous avons $\Leftrightarrow(i) = j$, ainsi que $\Uparrow(i) = \Uparrow(j) = l$ et réciproquement, $\Downarrow(l) = \{i, j\}$. De plus, la relation de parenté peut faire l'objet d'une composition dans le but de définir l'ensemble des *ancêtres* d'un concept comme la trace des compositions successives de la fonction *parent* jusqu'à ce que la racine de l'arbre soit atteinte :

$$\Uparrow^{\infty}(l) = \{\Uparrow^{k}(l)\}_{k \in [0 \ldots \infty]}$$

Les *descendants* à leur tour, sont définis comme l'ensemble des concepts qui partagent un ancêtre commun :

$$\Downarrow^{\infty}(l) = \{k, l \in \Uparrow^{\infty}(k)\}$$

Pour calculer le *vecteur caractéristique* d'un concept, une dernière référence sur la *population* doit être conservée lorsqu'un nouveau concept est formé. À une *feuille* donnée dans $\mathcal{L}$ correspond un ensemble de *patches* dans une lame. La population du k-ième concept dans $\mathcal{L}$ est le nombre de *patches* utilisés pour constituer le segment correspondant, $population(k) = |k| = |s_k|$. Récursivement, pour tout concept l né de la fusion de i et j, nous écrivons $|l| = |i| + |j|$ et définissons le vecteur caractéristique $\boldsymbol{f}_l$ du concept l comme la moyenne de $\boldsymbol{f}_i$ et $\boldsymbol{f}_j$ pondérée par les populations de i et j :

$$\boldsymbol{f}_l = \frac{|i| \times \boldsymbol{f}_i + |j| \times \boldsymbol{f}_j}{|l|} \tag{3.2}$$

L'algorithme de fusion sur critère de Kruskal va ensuite regrouper séquentiellement les paires de concepts les plus similaires selon la *métrique* apprise par le *CNN* dans l'espace caractéristique. La connaissance de la machine prendra finalement la forme d'un *arbre binaire strict* noté $\boldsymbol{T} = (\boldsymbol{C}, \Uparrow, \Downarrow)$.

3.3.3.2 Arbre le plus significatif

Gagolewski et al. [2016] le font remarquer, l'algorithme de fusion de régions exploré précédemment présente un dernier inconvénient commun à toutes les stratégies d'aggrégation *single-linkage* : il a tendance à créer des groupes d'effectifs radicalement différents. L'effectif, dans notre cas, est directement lié au paramètre de *population* d'un concept. Dès lors, en accord avec les principes énoncés dans la Sous-section 3.2.1, les concepts faiblement peuplés ne représentent vaisemblablement pas des structures d'intérêt puisqu'ils n'ont, soit pas été observés dans suffisamment de lames, soit pas occupé une surface suffisante lorsqu'ils ont été détectés.

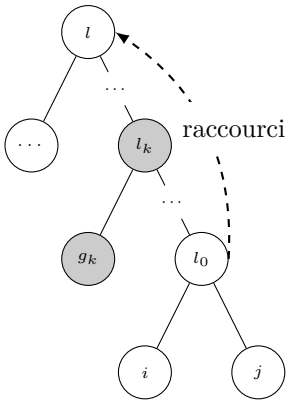


FIGURE 3.4 – Ramasse-miettes. Après la fusion de deux concepts aux effectifs significatifs, i et j, un certain nombre de fusions ne font qu'agglomérer des « miettes » g_k dont la population est moindre. Ceci correspond d'avantage à l'extension de l vers sa forme définitive, plutôt qu'à l'émergence d'un concept plus abstrait.

Très intuitivement, deux catégories de fusions se distinguent dans une procédure de groupement hiérarchique *single-linkage*. La première s'apparente à une sorte de *ramasse-miettes* où une classe comportant beaucoup d'individus, i, absorbe un groupe, j, qui n'en comporte presque aucun. Le concept résultant l, dans ce cas, ne s'interprète pas comme une nouvelle entité distincte, mais plutôt comme une extension du concept i. Le second type de fusion réunit deux groupes aux effectifs statistiquement significatifs en un nouveau concept plus abstrait et fait réellement de i et j des sous-types spécifiques d'un concept l plus général, voir Figure 3.4.

Selon les considérations ci-dessus, un bon indicateur de « significativité » pour la fusion d'une paire de concepts-frères (i, j) est naturellement fourni par l'effectif du concept le moins peuplé et nous écrirons, $|(i, j)| = min(|i|, |j|)$. De cette façon, les enfants les *plus significatifs* d'un concept l sont la paire de concepts-frères de significativité maximale parmi les descendants de l :

$$\Downarrow^{*}(l) = \Downarrow(\arg\max_{c \in \Downarrow^{\infty}(l)}(|\Downarrow(c)|)$$

Les enfants les plus significatifs d'un concept peuvent alors être trouvés en examinant la significativité de toutes les paires de concepts-frères descendants. Cependant, afin d'éviter une recherche exhaustive, nous proposons une descente sélective de $\boldsymbol{T}$ qui ne considère que les concepts les plus peuplés à chaque étape. Pour cela, soit $\Downarrow^{+}(.)$ une version sélective de $\Downarrow(.)$ qui ne retourne que l'enfant le plus peuplé d'un concept $\Downarrow^{+}(l) = \arg\max_{c \in \Downarrow(l)}|c|$. La trace des descendants les plus peuplés correspondante, $\Downarrow^{+\infty}(.)$, présente des propriétés particulièrement intéressantes lorsqu'il s'agit de trouver les enfants les plus significatifs d'un concept.

Propriété 1. *Tout descendant d'un concept l a un ancêtre dans la trace des descendants les plus peuplés de l : $\forall c \in \Downarrow^{\infty}(l), \exists n \in \mathbb{N}, \Uparrow^{n}(c) \in \Downarrow^{+\infty}(l)$.*

Lemme 1. *Le parent direct des enfants les plus significatifs de l est un élément de la trace des descendants les plus peuplés de l : $\Uparrow(\Downarrow^{*}(l)) \in \Downarrow^{+\infty}(l)$.*

Démonstration. Supposons que ce ne soit pas le cas, soit $c^{*} = \Downarrow^{*}(l)$ et étant donné la Propriété 1, $\exists n > 1 \in \mathbb{N}, \Uparrow^{\mathrm{n}}(c^{*}) \in \Downarrow^{+\infty}(l)$. Alors, par additivité de l'attribut de population, l'ancêtre de c^{*} dans la trace des descendants les plus peuplés aurait des enfants plus significatifs que c^{*} : $|\Downarrow(\Uparrow^{\mathrm{n}}(c^{*}))| > |c^{*}|$, ce qui, par définition de c^{*}, est une contradiction. □

En démarrant de la racine de $\boldsymbol{T}$ et en cherchant récursivement les enfants les plus significatifs, une version plus compacte de $\boldsymbol{T}$, qui court-circuite tous les ramassages de miettes, peut être construite. Le nouvel *arbre binaire strict* obtenu est appelé *arbre le plus significatif* et nous l'écrirons $\boldsymbol{T}^* = (\boldsymbol{C}, \Uparrow^*, \Downarrow^*)$.

Il est intéressant de se pencher sur les modes et propriétés de l'inférence dans une structure de connaissances arborescente comme celles construites dans les paragraphes précédents. L'arbre de subsomption est un résultat de clustering hiérarchique *à base de prototypes*, voir le Section 1.4.2, où chaque concept $c \in \boldsymbol{C}$ présent dans l'arbre et décrit par un *vecteur caractéristique* f_c formé comme la moyenne des descripteurs de tous ses *patches* constitutifs. Ce modèle génératif procède donc à l'inférence par association : le rattachement d'un *patch* inconnu x, décrit par f_x, à un nœud de l'arbre est déterminé par sa *similarité* avec le nœud considéré en calculant $w(f_x, f_c)$. Bien entendu, la relation de subsomption établie entre les différents concepts va éviter la comparaison de x avec la totalité des concepts représentés dans l'arbre.

Dans un arbre de subsomption, la relation d'implication pour déduire l'appartenance à une classe est ascendante, autrement dit, si un *patch* x appartient à un concept c, alors il appartient à tous les ancêtres de c : $\forall n,\ x \in c \Rightarrow x \in \Uparrow^{\mathrm{n}}(c)$. Une règle de prédiction qui exploite cette propriété nous mène à comparer x seulement aux conepts-feuilles, puis à tirer parti de la structure de $\boldsymbol{T}$ ou $\boldsymbol{T}^*$ pour déduire l'ensemble des concepts qui lui sont attribuables.

En pratique, le nombre de concepts-feuilles peut lui aussi être conséquent. De plus, par construction, les concepts les plus éloignés de la racine sont représentés par moins de *patches*. Cela signifie que leur description est statistiquement moins fiable que celle des concepts les plus peuplés. Dès lors, en comparant les individus à prédire aux feuilles de l'arbre, les erreurs de classement surviendront plus fréquemment et une erreur commise au niveau des feuilles sera répercutée sur l'ensemble des prédictions de super-classes qui en découlent.

Il est donc préférable de procéder à une inférence descendante : les différentes sous-classes d'un *patch* sont prédites de la plus abstraite (racine de l'arbre) à la plus spécifique (feuille de l'arbre) par rattachements successifs aux concepts-enfants les plus similaires. Lors de la prédiction de x, les classes auxquelles il est successivement rattaché sont mémorisées dans une suite, $tr(x)$, que nous appellerons la *trace de prédiction* du *patch* et qui est définie récursivement de la manière suivante :

$$\begin{cases} tr_0(x) = racine(\boldsymbol{T}^*) \\ \\ tr_{n+1}(x) = \underset{c \in \Downarrow(tr_n(x))}{\arg\min}\ w(f_x, f_c) \end{cases} \tag{3.3}$$

3.3.4 Interprétation et visualisation

3.3.4.1 Interprétation

L'adjectif « explicable » est souvent employé pour qualifier un *espace caractéristique* dans lequel certains groupements d'individus, autrement dit des classes ou motifs, sont intelligibles pour l'expert humain. Sabol et al. [2019] et Yamamoto et al. [2019] ont par exemple recours à cette terminologie. Sans aucune volonté d'attiser des controverses sur le choix des termes, nous privilégierons par la suite l'emploi du terme *interprétabilité* qui semble plus approprié au développement qui va suivre selon Rudin [2019]. L'*interprétabilité* donc, comme le soulignent déjà implicitement d'autres études telle que Yamamoto et al. [2019], ne

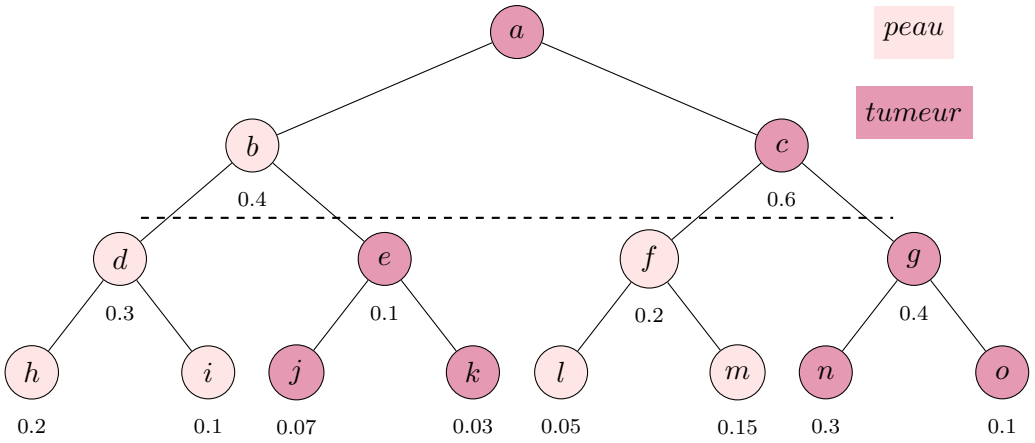


FIGURE 3.5 – Arbre de subsomption. Chaque nœud est un concept identifié par une lettre. Les couleurs indiquent la présence d'une classe majoritaire, « peau » ou « tumeur » pour reprendre l'exemple de la Figure 3.2 et de la Figure 3.3. Les concepts situés sous la ligne en pointillés sont *purs*, c'est-à-dire que leur *définition* $\boldsymbol{I}(concept)$ ne contient que des éléments de la classe dans laquelle ils sont colorés. Sous chaque concept se trouve la fraction de *patches* du concept-racine a qui lui est rattachée.

peut être mesurée qu'en visualisant des clusters. Cela implique de montrer à l'expert des images représentatives des différentes classes extraites. Lorsque l'expert retrouve majoritairement l'un de ses propres concepts dans un cluster de la machine, l'*interprétabilité* du modèle est subjectivement proclamée sans qu'aucune mesure quantitative de cette propriété n'ait véritablement été étudiée. Avec l'objectif de produire quelques métriques objectives pour quantifier l'*interprétabilité*, nous structurons les connaissances de l'expert $\boldsymbol{T}_{humain} = (\boldsymbol{C}_h, \Uparrow, \Downarrow)$ et celles de la machine $\boldsymbol{T}_{machine} = (\boldsymbol{C}_m, \Uparrow, \Downarrow)$ avec les arbres binaires définis précédemment et proposons d'étudier les procédures de traduction entre ces deux représentations.

La connaissance, dans les deux cas, dispose intrinsèquement d'un dictionnaire dans lequel les mots $h \in \boldsymbol{C}_h$, $m \in \boldsymbol{C}_m$ du vocabulaire de l'humain et de la machine sont implicitement rattachés à des ensembles d'images représentatives que nous appellerons *définitions* des mots de vocabulaire. Une *définition* rattachée à un mot $c \in \boldsymbol{C}_h \cup \boldsymbol{C}_m$ sera notée $\boldsymbol{I}(c)$. Suivant la structure de dictionnaire, la factorisation d'un mot entre différentes représentations s'effectue par l'application d'opérateurs logiques sur les *définitions*. Par exemple, il est possible de mesurer le sens partagé entre deux mots en calculant l'intersection de leur *définitions*, ce qui s'avère être assez proche d'un calcul de taux d'erreur communément utilisé pour évaluer la performance d'un modèle de classification, $\boldsymbol{I}(h) \cap \boldsymbol{I}(m)$.

Nous nommons d'abord *hyperonymie* d'un concept humain, tous les sous-ensembles de $\boldsymbol{C}_m$ dans lesquels sa *définition* est incluse. Plus rigoureusement, l'*hyperonymie* de h est le sous-ensemble $\mathcal{P}_h = \{p_1, \ldots, p_n\} \subseteq \mathcal{P}(\boldsymbol{C}_m)$ tel que $\forall i, \boldsymbol{I}(h) \subseteq \boldsymbol{I}(p_i)$. Ensuite, nous appelons *factorisations* de h ses *hyperonymes* les plus spécifiques $\mathcal{F}_h = \arg\min_{p \in \mathcal{P}_h}(\boldsymbol{I}(p) - \boldsymbol{I}(h))$. Finalement, l'*interprétation* de h est définie, de manière non-équivoque, comme sa *factorisation* la plus succincte :

$$interp(h) = \arg\min_{f \in \mathcal{F}_h} Card(f)$$

L'*interprétation* est la traduction d'une expression qui est aussi brève que possible dans le domaine ciblé. Lorsque les mots de l'expert sont *interprétés* avec des expressions courtes dans le langage de la machine, nous dirons que les deux représentations sont proches. Dès lors, la cardinalité moyenne de l'*interprétation* sur tous les mots du dictionnaire humain caractérise assez fidèlement le *fossé sémantique* en comptant le nombre moyen de mots nécessaires à la machine pour couvrir entièrement le sens d'un mot humain.

Pour fixer les idées, considérons plus en détails l'exemple fourni par la Figure 3.5. Selon cet arbre, nous pouvons dire que le concept a de la machine contient aussi bien la *définition* du concept *peau* que du concept *tumeur*. Le concept-racine a dans cet arbre est donc un *hyperonyme* de *peau* ainsi que de *tumeur* et nous notons $a \in \mathcal{P}_{peau}$ et $a \in \mathcal{P}_{tumeur}$. Le concept c, en revanche, ne contient pas tous les *patches* de *tumeur* dans sa descendance et n'est donc pas un *hyperonyme* de ce concept. On peut également noter que $I(j) \cup I(k) \cup I(n) \cup I(o) \in \mathcal{F}_{tumeur}$ puisque cet ensemble contient tous les *patches* du concept *tumeur* et aucun du concept *peau*. Pour autant, l'ensemble $\{j, k, n, o\}$ n'est pas l'*interprétation* de la *tumeur*, car un autre ensemble de nœuds, de cardinalité plus faible, regroupe également la *définition* de ce concept : $interp(tumeur) = \{e, g\}$. Notons enfin que le fossé sémantique entre cette représentation et la connaissance humaine prend la valeur 2, puisqu'il faut en moyenne deux mots de la machine pour exprimer un mot humain, $\{d, f\}$ pour la *peau* et $\{e, g\}$ pour la *tumeur*. Le fossé, dans ce cas, est donc supérieur aux représentations idéales présentées dans la Figure 3.2 et la Figure 3.3 qui à un mot humain associent un nœud unique de l'arbre.

3.3.4.2 Combler le fossé sémantique

Les *interprétations* ne peuvent être obtenues qu'à partir d'annotations expertes puisque $\bigcup_i \boldsymbol{I}(h_i)$ est simplement un ensemble de *patches* étiquetés. Par chance ici, la procédure d'*annotation* n'a rien de comparable à la traditionnelle corvée manuelle de segmentation ou d'étiquetage *patch* par *patch*. Contrairement à la démarche supervisée, la connaissance de la machine est structurée avant l'étape de classification et l'*annotation* de l'expert consiste simplement à placer quelques identifiants sur des concepts déjà partiellement assimilés par la machine. Le nombre de concepts devant être étiquetés pour englober un mot unique h dans le langage de l'expert correspond exactement à la cardinalité de son *interprétation*. Puisque la plupart des stratégies de *clustering* ont pour objectif, souvent implicite, de réduire le *fossé sémantique*, une poignée de concepts étiquetés suffit généralement à classer quelques dizaines de milliers de *patches* et le travail d'*annotation* s'en trouve considérablement allégé, voir la Figure 3.6.

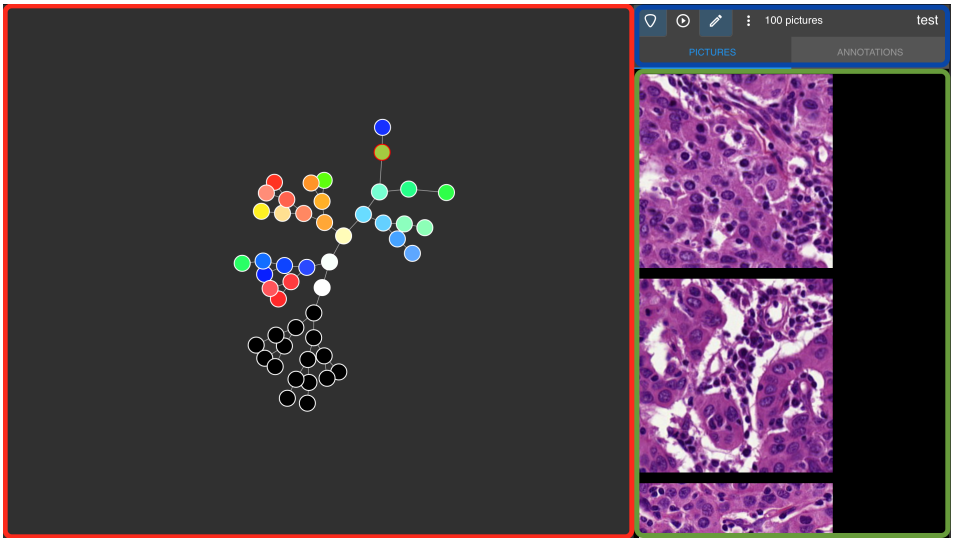


FIGURE 3.6 – Vue de l'application web pour l'annotation. La partie centrale, encadrée en rouge, contient une représentation interactive de l'arbre de subsomption (connaissance de la machine). Lorsque l'utilisateur expert clique sur un nœud n de l'arbre, il voit apparaître, dans la partie de droite, encadrée en vert, la collection de *patches* correspondant à la *définition* $\boldsymbol{I}(n)$. Le menu, situé en haut à droite, encadré en bleu, autorise l'utilisateur à étiqueter un nœud avec les termes de son choix (champ textuel à remplir).

La traversée du *fossé sémantique* repose donc sur une tâche d'annotation qui, bien que conceptuellement hautement simplifiée, requiert un outil software qui dépasse la simple visualisation des *patches* représentatifs de $\boldsymbol{C}_m$, voir la Figure 3.6. Un outil dédié au franchissement du *fossé sémantique* doit donc répondre à un certain nombre d'attentes formulées par les experts. Tout d'abord, tous les concepts significatifs de $\boldsymbol{C}_m$, ainsi que leur structure arborescente $\boldsymbol{T}^*$, doivent être visibles par l'utilisateur. Ensuite, l'interface doit pouvoir montrer, à la demande de l'utilisateur, tous les éléments de la *définition* d'un concept. Autrement dit, il doit être en mesure de présenter, sans biais, $\boldsymbol{I}(h), \forall h \in C_h$. Un champ textuel doit encore être mis à disposition du pathologiste pour lui donner la possibilité de placer un ou plusieurs mots de son jargon sur un concept établi par la machine. Un aspect important de la gestion des utilisateurs doit également être pris en compte : tous les utilisateurs ne sont pas nécessairement des experts. Lorsqu'un novice se connecte au système, il est identifié en tant que tel et ne peut émettre que des suggestions d'annotations. Seuls certains utilisateurs habilités peuvent effectivement impacter l'injection de connaissances dans la machine. On note également l'intérêt de pouvoir identifier plusieurs experts et de les faire annoter séparément afin de quantifier le *fossé sémantique* humain-humain qui peut exister entre deux pathologistes. Enfin, une dernière attention doit être apportée à l'esthétique de la solution : l'interface doit être attirante, aussi ludique que possible pour l'expert qui, dans le cas contraire, ne prendra pas le temps de s'en servir.

Une suite naturelle à cette procédure d'annotation serait d'entraîner l'architecture neuronale à adapter sa représentation et la métrique associée pour s'adapter au nouveau classement que lui impose l'utilisateur. Cependant, cette pratique fait courir le risque de sur-apprendre un sous-ensemble trop restreint de C_m incomplètement étiqueté par l'expert. De plus, et comme nous pouvons notamment le lire dans Bilal et al. [2017], le problème de classification dans ce cas relève du domaine de la *classification hiérarchique* où les stratégies d'apprentissage de réseaux profonds sont lourdes à mettre en place. Enfin, du point de vue de l'inférence, il n'est pas nécessaire de ré-apprendre un modèle si les *interprétations* annotées sont suffisamment pures. En effet, si certaines *interprétations* peuvent être de cardinalité élevée, c'est-à-dire que l'expert a tout de même dû annoter un certain nombre de concepts dans l'arbre, l'inférence selon l'Equation 3.3 n'en est absolument pas impactée. Les annotations de l'expert sont mémorisées sous la forme d'une structure clefs-valeurs qui associe, ou non, à chaque concept $m \in \boldsymbol{C}_m$ un élément $h \in \boldsymbol{C}_h$. Ainsi, lorsqu'un *patch* x se présente, un simple parcours des termes de sa *trace de prédiction* $tr(x)$ permet de lui attribuer les mots humains qui lui correspondent.

3.3.5 Application à la gradation des cancers du sein

3.3.5.1 Jeu de données

Nous disposons d'un large ensemble de lames numérisées constitué par des cohortes *UNICANCER* dans le cadre de leur Programme Adujvant dans le Cancer du Sein (*PACS*), *PACS04*, *PACS05*, *PACS06* et *PACS08*. La problématique posée est la suivante : un réseau de neurones convolutif profond peut-il apprendre à distinguer les grades les plus extrêmes (grades 1 et 3) du cancer du sein dans des lames en coloration standard *H&E* ? Le travail décrit ici ne présente pas la version la plus aboutie du système et ne s'intéresse que modérément à la performance absolue du modèle prédictif. Cette description insiste d'avantage sur l'impact bénéfique que peut avoir notre méthodologie d'extraction de connaissances sur la vitesse de mise en place d'une solution de ce genre.

3.3.5.2 Apprentissage de métrique

Suivant la démarche *PUL* développée de manière très similaire et presque simultanément par Fan et al. [2018] et Caron et al. [2018b], nous choisissons d'adapter la représentation d'un réseau existant pré-entraîné aux images de sein en coloration standard *H&E*. Dans cet objectif, nous utilisons l'architecture Xception proposée par Chollet [2017] parce que son nombre de paramètres est bien inférieur à celui d'autres architectures aux performances comparables Sengupta et al. [2019] et il a prouvé son efficacité sur des jeux de données très larges et diversifiés tel que Imagenet Deng et al. [2009]. Il est d'ailleurs communément admis que produire d'aussi bons résultats sur des ensembles de données de cette envergure, garantit l'apprentissage d'une représentation d'images très générale et utile à de multiples tâches de reconnaissance.

Dès lors, nous décrivons l'entièreté de l'ensemble des *patches* relevés dans les *WSI* dans l'*espace caractéristique* de Xception, c'est-à-dire en récupérant la sortie de sa dernière couche de convolution, et exécutons un algorithme des *k-moyennes* sur la base de ces descripteurs. Les *k-moyennes* dans notre expérience sont toujours lancés avec le paramètre $k = 10$ parce qu'il reflète grossièrement le nombre de mots prononcés par le pathologiste lorsqu'il inspecte les lames avec une fenêtre de taille p extraite au niveau de résolution r.

Au cours d'une itération de *PUL*, Xception est entraîné à adapter sa représentation sur 20 *epochs* en utilisant la fonction de coût *cross entropy* entre ses prédictions et les étiquettes fournies par les *k-moyennes* en guise de vérité terrain. Les poids du réseau son mis à jour selon la règle d'optimisation *Adam*, proposée par Kingma and Ba [2014], et le *taux d'apprentissage* est initialisé avec une valeur de 10^{-3}. Le seuil d'acceptation des individus pour la constitution des ensembles d'apprentissage dans la méthode *PUL* est fixé à $0,75$ et 5 itérations étaient généralement suffisantes pour arriver à convergence de la méthode.

3.3.5.3 Segmentation et construction des arbres

Comme nous avons pu le voir au cours de cette section, les tâches de groupement sont majoritairement réalisées par des algorithmes de fusions de régions sur critère de Kruskal. Lorsque l'algorithme est exécuté dans sa forme originale, comme cela est le cas pour construire l'arbre binaire des connaissances de la machine, aucun hyperparamètre ne doit être ajusté manuellement. La version de Felzenszwalb and Huttenlocher [2004] que nous utilisons pour la segmentation spatiale en revanche, impose de définir le paramètre τ dans l'Equation 3.1. Le terme contenant ce paramètre teinte l'Equation 3.1 d'inhomogénéité en mêlant ressemblance dans l'*espace caractéristique* avec taille des segments formés par la procédure de fusion rendant ainsi l'influence exacte du paramètre de taille τ difficile à interpréter. Le risque ici est la perte des signaux qui occupent de petits segments et, de manière très subjective, nous posons $\tau = 10$ après inspection qualitative des segments produits sur les lames entières.

3.3.5.4 Annotation accélérée

La pertinence et l'efficacité de l'approche sont mises en avant sur le cas concret de l'annotation accélérée. Sur la base de 1125 *WSIs* tirées des cohortes *PACS04*, *PACS06* et *PACS08*, nous souhaitons entraîner un *CNN* à faire la différence entre les tumeurs de grade bas et celles de grade haut. Le grade des tumeurs a été réalisé sur l'ensemble des lames par des pathologistes experts de la pathologie mammaire. Pour constituer l'ensemble d'apprentissage, un sous-ensemble de 898 lames est découpé en tuiles non-chevauchantes $x = \{x_1, \ldots, x_N\}$ de taille $s = 299$ pixels extraites selon une grille régulière dont le pas vaut également s. Une étiquette $y_i \in [0, 1]$, vérité terrain assignée à un *patch* d'entraînement x_i, prend l'une des

valeurs suivantes : (0) la tuile appartient à la lame d'une patiente dont la tumeur est de bas grade, (1) l'imagette appartient à la lame d'une patiente dont la tumeur est de haut grade. De manière analogue, 227 lames indépendantes des 898 *WSIs* d'apprentissage sont utilisées pour extraire de nouveaux *patches* annotés et constituent un ensemble de données inconnu du classifieur et destiné à son évaluation.

Le jeu de données final contient 38093 *patches* de la classe (0) et 4215697 de la classe (1) pour l'ensemble d'apprentissage, et 11810 *patches* de la classe (0) et 106216 de la classe (1) pour l'ensemble de validation. Comme nous l'avons évoqué dans la Sous-section 3.2.1, sans plus de filtrage des *patches*, l'information discriminante entre la classe (0) et la classe (1) sera probablement perdue au milieu des données non-informatives des *patches* les moins pertinents. Bien entendu, la gradation des tumeurs repose bien plus sur l'examen des *patches* extraits dans la zone tumorale que sur celui des imagettes prises dans les structures alentours. Nous avons donc demandé à un pathologiste de déterminer l'*interprétation* du concept humain *tumeur* en prenant quelques minutes pour étiqueter les nœuds de l'arbre binaire de la machine dans notre application web. Pour une évaluation plus objective, les images utilisées pour extraire l'arbre de connaissances de la machine proviennent de 128 *WSIs* choisies en dehors des ensembles d'apprentissage et de validation utilisés pour la classification des grades. Il est important de noter comment « quelques minutes » d'annotation auraient pu devenir plusieurs heures s'il avait fallu segmenter ou pointer manuellement des zones tumorales dans les images entières.

3.3.5.5 Classification de patches

En utilisant les *interprétations* fournies par le pathologiste, trois versions des ensembles d'entraînement et de validation ont été formées et un apprentissage a été réalisé sur chacune d'elles. Le premier classifieur est ainsi entraîné sur des *patches* pris uniquement en dehors de la zone tumorale, autrement dit qui ne tombent pas dans l'*interprétation* de *tumeur* pour la machine. Le second classifieur est entraîné sur la totalité des *patches*, c'est-à-dire sans qu'aucun filtre n'ait été appliqué aux données. Enfin, une troisième expérience entraîne le réseau uniquement sur les *patches* issus de la zone tumorale.

Chacune des trois expériences est menée avec la même architecture, ResNet18 He et al. [2016], entraînée depuis zéro avec la fonction de coût *cross entropy*, en utilisant la règle *Adam* pour l'optimisation des poids. Le taux d'apprentissage est toujours initialisé à 10^{-3} et l'entraînement est réalisé sur 50 *epochs* avec une taille de *batch* de 64 *patches*. Le nombre d'*epochs* est volontairement placé haut de manière à étudier le comportement du modèle jusque dans une situation de sur-apprentissage flagrante. La valeur de la fonction de coût sur la validation est relevée au cours de l'apprentissage pour les 3 expériences et résumée dans les courbes de la Figure 3.7. Elles montrent exactement les tendances auxquelles nous pouvions nous attendre : plus la concentration en *patches* étiquetés *tumeur* est importante dans le jeu de données, plus la décroissance de la *cross entropy* est importante. Ces courbes montrent d'une part que l'information nécessaire à la gradation des tumeurs, y compris pour des solutions dans l'état de l'art de la classification des images, est uniquement présente dans la zone tumorale. D'autre part, elles mettent en évidence la pertinence des annotations réalisées avant la classification et, couplées à la rapidité d'étiquetage, confirment l'efficacité du franchissement de *fossé sémantique* pour le partage rapide de connaissances entre l'humain et la machine.

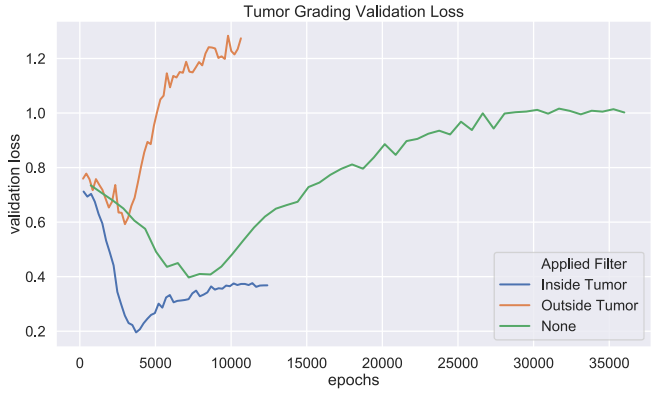


FIGURE 3.7 – Évolution de la *cross entropy* sur l'ensemble de validation au cours de l'apprentissage. De plus fortes concentrations en *patches* de tumeur conduisent à de meilleurs profils d'apprentissage. On note, quantitativement, qu'une fois passé le cap du sur-apprentissage, la fonction de coût des *patches* de tumeur seule (**blue**) reste plus basse que la meilleure valeur obtenue pour l'ensemble de *patches* non-filtré (**green**).

3.3.6 Bilan

Cette section propose une méthodologie générale pour aborder les projets d'analyses automatiques de *WSIs*. Elle précède et se place en support de la résolution par un modèle de classification supervisée. Pour un tissu donné et sans aucune supervision, elle extrait dans les images des concepts visuels dont les empreintes, spatiale et *caractéristique*, sont statistiquement significatives. Ces concepts sont regroupés hiérarchiquement par ressemblance au sens d'une métrique apprise par le modèle descripteur dans l'*espace caractéristique*. Pour chaque concept, le système mémorise un *prototype* qui, par association, permet d'étiqueter de nouveaux individus (*patches*) qui lui sont similaires. La structure hiérarchique, accessible à travers une interface, permet à l'utilisateur de cerner rapidement la logique de groupement utilisée par la machine et de traduire facilement ces concepts dans son propre langage. La traditionnelle tâche d'annotation d'images par le pathologiste en est grandement facilitée, puisqu'en traduisant une poignée de concepts dans la connaissance de la machine, il étiquette et regroupe simultanément sous une même classe des milliers de *patches*.

De manière intéressante, la structure souligne les biais présents dans les données et les fait immédiatement visualiser à l'utilisateur. Dans la partie traitant de l'application à la gradation du cancer du sein, nous avons évoqué la présence de différentes cohortes *PACS04*, *PACS05*, *PACS06* et *PACS08*, chacune ayant des caractéristiques visuelles reconnaissables. Dans nos premiers développements, ces différentes cohortes sont nettement séparées dans trois branches indépendantes de l'arbre de subsomption. Cela multiplie naturellement par trois le nombre d'annotations que doit réaliser l'expert. En effet, lorsqu'un concept d'intérêt comme *tumeur* est relevé, il est vraisemblablement présent dans toutes les cohortes et si ces dernières sont dans des branches séparées, alors l'annotation *tumeur* devra être réalisée dans chacune de ces branches.

En effet, la plupart de ces biais dans le jeu de données passent inaperçus lors de l'entraînement et de l'évaluation des modèles de classification, tout particulièrement lorsqu'ils sont inconnus *a priori* par le développeur de la solution. Cette problématique fait d'ailleurs actuellement l'objet d'une attention toute particulière de la part de la communauté de l'apprentissage automatique depuis les observations faites par Amazon sur leur algorithme de recrutement [3] et

3. https ://www.reuters.com/article/us-amazon-com-jobs-automation-insight/amazon-scraps-secret-ai-recruiting-tool-that-showed-bias-against-women-idUSKCN1MK08G

l'on entend de plus en plus parler de *fairness in AI* ou « intelligence artificielle équitable ». La solution de construction de connaissances et de visualisation proposée offre ainsi une manière de détecter qualitativement la présence de ces biais. De plus, les considérations précédentes sur les cohortes et la multiplication des annotations en présence de biais portent à croire que le *fossé sémantique* serait un bon indice d'amélioration de *fairness* de la représentation d'un réseau de neurones.

Nous l'avons évoqué, si un ensemble de données contient n sous-populations, disons des cohortes dans notre cas, une représentation biaisée multiplie vraisemblablement par un facteur n la plupart des *interprétations* annotées par le pathologiste. Ainsi, la représentation apprise par le réseau de neurone étudié devient *fair* lorsque le *fossé sémantique* est diminué d'un facteur n. Si l'information reste relative, le *fossé sémantique*, tel que nous l'avons défini dans cette section, serait une métrique objective et quantitative de l'équité du modèle.

Un peu plus avant, nous avons vite rejeté l'idée d'une ré-adaptation après annotation du modèle de représentation des images. Bien que cette procédure d'adaptation soit risquée, nous pensons qu'elle constitue une perspective d'amélioration importante dans la démarche globale d'analyse. La procédure dans son entièreté serait alors pensée comme une boucle fermée dans laquelle la sortie de l'annotation serait utilisée pour améliorer la représentation. Par *amélioration*, il faut entendre ici une réduction du *fossé sémantique* qui permettra non-seulment de rendre les classifications plus pertinentes, mais surtout de rendre les représentations insensibles aux biais.

3.4 Détection et segmentation syntaxique

La section précédente s'attristait de la légèreté des définitions d'*interprétabilité* rencontrées dans la littérature et, bien qu'elle ait défini et utilisé à maintes reprises des *interprétations*, elle n'établit pas une définition claire de ce qu'il faut appeler *interprétabilité* en apprentissage automatique.

À la question « qu'appelle-t-on *interprétabilité* en *machine learning* ? », la réponse la plus répandue parle de décisions « intelligibles » ou « compréhensibles » par l'humain. Tandis que la section précédente et les articles auxquels elle fait référence s'attachent à qualifier d'*interprétables* des résultats de classification ou des *espaces caractéristiques*, nous notons plutôt que la conception intuitive fait d'avantage allusion à la *décision* lorsqu'elle tente de définir l'*interprétabilité*.

Toujours intuitivement, une décision *interprétable* correspond intrinsèquement à un temps d'analyse postérieur au franchissement du *fossé sémantique* dans lequel seul l'ensemble $\boldsymbol{C}_h$ des concepts humains est considéré et, comme l'indiquaient déjà certaines réflexions des Section 1.3 et Section 1.4, la décision s'appuie sur des relations entre les concepts de $\boldsymbol{C}_h$ et relève de raisonnements logiques déductifs. Dans le domaine de l'analyse de données, seuls les *arbres de décisions* implémentés dans des *systèmes experts* permettent de produire de telles analyses de manière automatique.

L'arbre de subsomption construit précédemment, fait état d'une relation de ressemblance visuelle entre les concepts. Elle est basée sur une métrique de similarité acquise statistiquement qui place naturellement la structure du « mauvais » côté du *fossé sémantique*. L'attribution d'une ou plusieurs classes à un *patch* en descendant l'arbre ne relève absolument pas d'une succession de décisions logiques et ne s'appuie sur des éléments de $\boldsymbol{C}_h$ qu'à condition que la structure ait été annotée : cette décision ne saurait donc être qualifiée d'*interprétable*.

L'arbre $\boldsymbol{T}_{machine} = (\boldsymbol{C}_m, \Uparrow, \Downarrow)$ offre néanmoins la capacité de reconnaître très rapidement et à coût d'annotation quasi-nul la totalité des concepts humains élémentaires sans lesquels toute déduction dans $\boldsymbol{C}_h$ serait impossible, ce que la classification supervisée ne permettrait qu'au prix d'efforts démesurés. De plus, l'arbre de subsomption et son outil d'annotation ont la particularité de franchir le *fossé sémantique* en utilisant des opérateurs logiques d'union entre des ensembles de *patches*. Ils posent ainsi les bases de ce qui pourrait être une forme d'injection rapide des règles humaines de déduction au sein des concepts de $\boldsymbol{C}_h$.

Les autres parties de ce travail de thèse se sont appliquées à détailler des mécanismes de perception et de franchissement du *fossé sémantique*. Afin de compléter nos explorations et contributions à l'ensemble de la chaîne d'analyse automatique, cette section est dédiée à $\boldsymbol{C}_h$, aux relations entre ses éléments et leur exploitation pour déduire la présence d'objets ou de phénomènes dans les images. Une première partie tentera de mettre en évidence les relations entre les concepts humains qui peuvent permettre de guider une analyse automatique d'images Sous-section 3.4.1. Une deuxième partie présentera les approches dans la littérature qui ont parfois recours à des connaissances humaines structurées pour résoudre un problème de segmentation Sous-section 3.4.2. Une troisième partie fournira certaines propriétés que doit respecter une solution de segmentation syntaxique Sous-section 3.4.3. Ensuite, nous noterons comment le problème de segmentation d'objets s'inscrit dans le formalisme du problème classique d'isomorphisme de sous-graphes Sous-section 3.4.4 qui peut être abordé avec des techniques d'optimisation gloutonnes. Enfin, la méthodologie étudiée sera appliquée au cas concret de la segmentation de noyaux de cellules dans des images de fluorescence en revisitant un algorithme mis au point dans les premiers temps de ce travail de thèse Sous-section 3.4.5.

3.4.1 Raisonnement sur les images

3.4.1.1 Raisonnement dans l'espace sémantique

Le postulat des développements qui vont suivre est celui d'un *fossé sémantique* franchi. *Franchi* ici ne signifie pas qu'un classifieur est capable de répondre directement à une problématique très spécifique, comme faire la distinction entre deux pathologies par exemple Section 2.5. Un système d'analyse automatique d'images ne peut pas prétendre avoir véritablement franchi le *fossé sémantique* s'il se contente de connaître seulement deux mots de vocabulaire du pathologiste. Comme nous l'avons évoqué au début de ce chapitre [ref], c'est une forme d'*érudition* ou d'*exhaustivité* de vocabulaire qui caractérise le *franchissement* et garantit l'*interprétabilité* des décisions.

Ne nous méprenons pas, l'*exhaustivité* dans la perception ne signifie pas pour autant que le système est capable de reconnaître la totalité du vocabulaire de l'humain. L'*érudition* du classifieur se limite à un sous-groupe d'*axiomes* dans $\boldsymbol{C}_h$ à partir desquels tous les autres concepts présents dans l'image peuvent être inférés, non plus par des rapprochements statistiques, mais bien par un mode de raisonnement déductif.

Cette section reprend les notations et définitions de *justifications* employées dans la [ref] du premier chapitre. Nous rappelons ici qu'un concept $c_i \in \boldsymbol{C}_h$ est une *justification partielle* de $c \in \boldsymbol{C}_h$ ou encore que c_i est une condition *nécessaire* à c si, et seulement si, la présence de c dans l'image conditionne absolument la présence de c_i dans cette même image. Autrement dit, $c \Rightarrow c_i$, et l'on notera plus facilement $c_i \in \Downarrow^{\infty}(c)$ par souci d'harmonie avec les notations de hiérarchie adoptées tout au long de ce travail.

Nous disons ensuite d'un ensemble $\boldsymbol{j} = \{c_1, \ldots, c_n\} \subset \boldsymbol{C}_h$ qu'il est une *justification complète* de c et l'on écrira $\boldsymbol{j} \in \mathbb{J}(c)$ si, et seulement si, tous les éléments de $\boldsymbol{j}$ sont des *justifications partielles* de c et que la présence simultanée de tous les éléments de $\boldsymbol{j}$ conditionne

absolument la présence de c :

$$\forall c_i \in \boldsymbol{j}, \quad c_i \in \Downarrow^{\infty}(c) \quad | \quad \bigwedge_i c_i \Rightarrow c$$

Nous ajouterons une notion d'indépendance aux *justifications complètes* d'un concept et appellerons ensemble des *justifications exactes* de c, $\mathbb{J}^{+}(c)$, toutes les *justifications complètes* $\boldsymbol{j}$ qui satisfont :

$$\forall (c_i, c_k)_{i \neq k} \in \boldsymbol{j} \times \boldsymbol{j}, \quad c_i \notin \Downarrow^{\infty}(c_k)$$

À la différence de toutes les autres classes, les *axiomes* de $\boldsymbol{C}_h$, notés $\mathbb{A}xioms(\boldsymbol{C}_h)$, sont des concepts qui n'ont pas de descendant par la relation d'implication, $\mathbb{A}xioms(\boldsymbol{C}_h) = \{a \in \boldsymbol{C}_h \mid \Downarrow^{\infty}(a) = \phi\}$. Ces concepts se trouvent naturellement au bord du *fossé sémantique*, ils ne sont pas obtenus par déduction, mais sont justement le produit du *franchissement* par classification supervisée ou annotation de *clusters* obtenus sans supervision. Trivialement, tout élément de $\boldsymbol{C}_h$ se décompose en concepts *axiomatiques*. Par extension, nous notons $\mathbb{A}xioms(c)$ la décomposition d'un concept $c \in \boldsymbol{C}_h$ et la définissons comme la *justification exacte* de c dont tous les éléments sont des *axiomes* de $\boldsymbol{C}_h$.

3.4.1.2 Le problème de la structure spatiale

Une hiérarchie sur $\boldsymbol{C}_h$, basée sur la relation d'implication, permet de raisonner par déduction sur les concepts de l'expert. Cependant, elle ne tient absolument pas compte de la nature structurée des images, c'est-à-dire du positionnement relatif des segments d'images étiquetés comme des *axiomes*.

Contrairement aux manipulations purement sémantiques décrites précédemment, la seule présence d'éléments de $\mathbb{A}xioms(c)$ dans l'image ne forme pas une *condition suffisante* pour prédire la présence d'un objet de classe c. Une décomposition élémentaire destinée à la détection ou à la segmentation de c doit en fait regrouper ses parties au sens de l'inclusion spatiale. De manière plus précise, si un objet de catégorie c occupe un segment s de l'image, alors des segments classés dans $\mathbb{A}xioms(c)$ n'indiquent pas seulement qu'un segment d'image s_i existe pour chaque *axiome* $c_i \in \mathbb{A}xioms(c)$, mais elle garantit que ces régions sont connexes et que leur fusion produit exactement le segment s. Une relation supplémentaire de *partinomie* relie donc les segments classés comme des *axiomes* aux segments de leurs concepts parents et l'on écrira, pour récupérer les notations de la [ref], $\forall s_i,\ s_i\ \mathcal{P}\ s$.

De plus, sauf cas particuliers, certaines relations de voisinage entre les segments d'*axiomes*, $\boldsymbol{V} = \{Sur, Sous, Gauche, Droite\}$ par exemple, doivent être satisfaites pour valider la présence effective d'un objet de classe c. Pour fixer les idées, prenons l'exemple d'un *visage* humain dont une décomposition *axiomatique* pourrait s'écrire $\mathbb{A}xioms(visage) = \{yeux, nez, bouche\}$, les détections simultanées des *yeux*, d'un *nez* et d'une *bouche* ne suffisent pas à conclure à la présence d'un visage humain dans l'image. En effet, si les segments attribués à ces structures *axiomatiques* prennent des configurations spatiales incompatibles avec la structure ordinaire du *visage*, disons par exemple la relation « le *nez* se trouve *sous* la *bouche* », alors un algorithme ne devrait pas conclure à la présence d'un *visage* dans l'image, bien qu'il puisse attester de la présence simultanée et connexe de tous les constituants *axiomatiques* de cet objet.

Nous remarquons alors que deux types de hiérarchies sont intriqués dans la prise de décision pour une segmentation déductive des objets, la hiérarchie des concepts humains, pour la relation d'implication, et la hiérarchie des segments d'image pour la relation d'inclusion. Lorsqu'elles étaient explicitement évoquées, ces hiérarchies ont toujours été traitées indépendamment, ce qui nous a notamment permis d'utiliser une notation commune pour les structures arborescentes

sans gestion de conflits. Un enjeu de cette section sera d'entremêler les deux structures pour formuler des règles de segmentation d'objets et ce contexte impose un effort d'adaptation des notations utilisées jusqu'à présent.

Notations

Sémantique

- On notera $\boldsymbol{C}_h = \{c_1, \ldots, c_n\}$ l'ensemble des concepts, nœuds de l'arbre sémantique ;
- Les notations relatives à la structure arborescente seront indicées par *sem*, et l'on notera ainsi l'arbre des concepts pour la relation d'implication $\boldsymbol{T}_{sem}(\boldsymbol{C}_h, \Uparrow_{sem}, \Downarrow_{sem})$;
- On notera $\mathbb{A}\text{xioms}(\boldsymbol{C}_h)$ les feuilles de $\boldsymbol{T}_{sem}$, aussi appelées *axiomes* de $\boldsymbol{C}_h$ et par extension, pour $c \in \boldsymbol{C}_h$, nous écrirons $\mathbb{A}\text{xioms}(c) = \{a \in \Downarrow^{\infty}_{sem}(c) \mid \Downarrow^{\infty}_{sem}(a) = \phi\}$;

Spatial

- On notera $\boldsymbol{S} = \{s_1, \ldots, s_m\}$ l'ensemble des segments, nœuds de l'arbre de segmentation ;
- Les notations relatives à la structure arborescente seront indicées par *space*, et l'on notera ainsi l'arbre des concepts pour la relation d'inclusion $\boldsymbol{T}_{space}(\boldsymbol{S}, \Uparrow_{space}, \Downarrow_{space})$;
- On notera $\mathbb{A}\text{toms}(\boldsymbol{S})$ les feuilles de $\boldsymbol{T}_{space}$, aussi appelées *atomes* de $\boldsymbol{N}$ et par extension, pour $s \in \boldsymbol{S}$, nous écrirons $\mathbb{A}\text{toms}(s) = \{a \in \Downarrow^{\infty}_{space}(s) \mid \Downarrow^{\infty}_{space}(a) = \phi\}$;

3.4.2 La connaissance dans les outils de segmentation

De nombreux algorithmes de détection et de segmentation automatique exploitent des connaissances, parfois structurées, de l'expert, telles que des liens d'implication ou bien des relations géométriques entre les *parties des objets* observables afin de *guider* l'extraction des *objets entiers* dans les images. Dans la littérature, on regroupe fréquemment l'ensemble de ces techniques sous l'appellation *part-based segmentation/detection* et on trouve notamment ces solutions implémentées pour résoudre des problèmes de *segmentation d'instances* pour lesquels une approche de *segmentation sémantique* seule ne permet pas d'individualiser un grand nombre d'objets compactés dans une foule. Si l'implémentation des méthodes et l'aspect technique des briques de la chaîne de détection diffèrent souvent d'un algorithme à l'autre, l'ensemble des solutions proposées semble reposer sur un socle commun : celui de la généralisation de la *transformée de Hough* Hough [1962] et des diverses approches dites de *template matching* qui en découlent telle que Duda et al. [1973], les modèles *en constellation*, et particulièrement ceux introduits par Weber et al. [2000] et Fergus et al. [2003], étant les plus connus.

Tout d'abord, un modèle génératif des objets à détecter ou segmenter, aussi appelé *atlas* dans le cas particulier de la segmentation, est produit sous la supervision de l'expert avant d'être encodé puis mémorisé par la machine. Une description détaillée des *atlas* et leur utilisation pour la segmentation d'images biomédicales est notamment fournie par les travaux de Rohlfing et al. [2004] ou de Rohlfing et al. [2005]. Ensuite, lorsque l'on présente une nouvelle image X au système, celui-ci commence par extraire un jeu de *points saillants* dans l'image, en utilisant notamment des approches d'extraction robuste comme l'algorithme SIFT Lowe [1999b], voir la Sous-section 1.3.3. Chacun des *points saillants* $\{x_1, \ldots, x_n\}$ de l'image renferme la position à laquelle il a été relevé, ainsi que le descripteur SIFT associé. Un jeu de points similaire $\{p_1, \ldots, p_m\}$ caractérise déjà chacun des objets P, ou *prototypes* de l'atlas, stockés en mémoire par la machine et l'étape suivante consiste à trouver le prototype P^* dont la configuration de points est la plus proche de celle de X. La comparaison des configurations ici n'est pas un simple calcul de score d'appariement par similarité des descripteurs, mais prend

aussi en compte les positions relatives : lorsque les x_i, x_j, x_k sont appariés avec les p_l^*, p_m^*, p_n^* respectivement, les angles orientés doivent varier le moins possible $(\overrightarrow{p_l^* p_m^*}, \overrightarrow{p_m^* p_n^*}) \approx (\overrightarrow{x_i x_j}, \overrightarrow{x_j x_k})$. Finalement, si le score d'appariement est suffisamment élevé, le système conclut à la présence d'un objet de type P^* dans l'image et le localise précisément.

La plupart des outils de *segmentation d'instances*, y compris les plus actuels dont le plus courant est certainement celui développé par He et al. [2017], fonctionnent sur le même principe. Les techniques les plus performantes utilisent des *CNNs* discriminants pour déterminer les points d'intérêt. La sortie de ces réseaux fournit généralement plus qu'une probabilité de présence de la partie d'objet reconnue, mais imite également le vote de la *transformée de Hough* en formulant une proposition de positionnement et de dimensionnement (*boîtes englobantes*) de l'objet entier. Une fois toutes les parties détectées, un algorithme élimine les propositions les plus redondantes puis, un second *CNN* est utilisé sur chacune des *boîtes englobantes* restantes pour réaliser une *segmentation sémantique* et détourer précisément l'instance repérée précédemment.

Aucune de ces approches ne se place véritablement dans notre cas d'usage. Elles implémentent toutes une forme complète de chaîne perceptuelle sans se soucier du point de franchissement effectif du *fossé sémantique* et donc de l'interprétabilité de la décision. Bien que des connaissances sur la structure géométrique des objets soient intégrées dans la méthode, on parle plus précisément dans ce cas de segmentations *model-driven* ou *knowledge-driven*, aucune d'elles n'est formulée explicitement par un expert. Dans les formalismes les plus courants, principalement explorés dans les études de Lin et al. [2007a], Felzenszwalb et al. [2010] et Zhang et al. [2014], le guidage de la segmentation par ces connaissances se matérialise toujours sous forme de contraintes appliquées à un problème d'optimisation, mais ne fait jamais appel à des opérateurs logiques et donc à une forme de raisonnement.

Le domaine de la télédétection utilise encore fréquemment le raisonnement logique pour détecter et classer des structures dans les images de manière plus rapide et plus robuste. Bon nombre de ces approches ont notamment été recensées dans les études de Forestier [2010] et de Kurtz [2012]. Il faut cependant remonter à des travaux plus anciens, comme ceux de Ehrig et al. [1992] et Fuchs [2001], dont certains sont développés à destination des images biomédicales comme cela est le cas de Ogiela and Tadeusiewicz [2002], pour voir la déduction prendre véritablement le pas sur la classification dans certaines méthodes que l'on appelle segmentations *syntaxiques*. Nous proposons donc, et détaillons dans la suite de cette section, une méthodologie originale de segmentation syntaxique.

3.4.3 Segmentation hiérarchique indicée sémantiquement

3.4.3.1 Structure des données d'entrée

Il s'agit ici de préciser la structure du **point de départ** de notre algorithme de segmentation syntaxique. Le franchissement du fossé sémantique, tel que nous avons pu le décrire dans les paragraphes précédents, initialise bien évidemment la méthode avec une partition *axiomatique*, puisque les membres de $\mathbb{A}\text{xioms}(\boldsymbol{C}_h)$ sont, par définition, les seules classes que le modèle de perception est capable de prédire. De plus, nous notons que la segmentation d'un objet par ses parties, repose sur un principe fondamental d'*intrication* entre les hiérarchies spatiale et sémantique : les segments constitutifs d'un objet lui sont *sémantiquement* inférieurs. Cela implique donc nécessairement que la partition de départ de la méthode est également une segmentation *atomique*. On appellera *partition atomique axiomatique* et on notera $\boldsymbol{S}_0$ ce point d'entrée de la méthode.

3.4.3.2 Structure des solutions

Hiérarchie C'est l'*exhaustivité* sémantique, nécessaire à l'interprétabilité de la procédure, qui impose une structure hiérarchique à la solution de notre problème de segmentation. Soit un objet de catégorie c détouré par un segment s d'une image X. Si c admet une *justification complète*, autrement dit $\mathbb{A}\text{xioms}(c) \neq \phi$, alors la procédure de segmentation **doit** détourer chacun des éléments $c_i \in \mathbb{J}(c)$ dans un segment s_i. L'objet de classe c_i est alors interprété comme une partie de l'objet de catégorie c, $s_i \in \Downarrow_{space}(s)$. Par conséquent, si tous les éléments de C_h **doivent** être détectés, le résultat de la segmentation syntaxique est obligatoirement hiérarchique.

Hiérarchie indicée Soit X une image et $\boldsymbol{T}_{space}(\boldsymbol{S}, \Uparrow_{space}, \Downarrow_{space})$ une segmentation hiérarchique de cette image. Pour une fonction f des segments d'images et à valeurs dans $\mathbb{R}$, on dira du couple $(\boldsymbol{T}_{space}, f)$ qu'il forme une *segmentation hiérarchique indicée* de X si, et seulement si, les propriétés suivantes sont respectées :

Propriété 2. *L'image par f d'un segment est inférieure à celle de n'importe quel autre segment qui le contient :*

$$\forall s_i, s_j \in \boldsymbol{S}, s_i \subset s_j \;\Rightarrow\; f(s_i) < f(s_j)$$

Propriété 3. *L'image par f d'un segment atomique est nulle :*

$$\forall s \in \mathbb{A}\text{toms}(\boldsymbol{S}), f(s) = 0$$

Les *hiérarchies indicées* sont particulièrement explorées dans les approches de segmentation d'images à *contrainte sémantique faible* pour lesquelles certaines fonctions f, aussi appelées *critères croissants* en morphologie mathématique, permettent d'évaluer des *énergies de segmentation* et d'établir des coupes optimales de ces hiérarchies (voir la Sous-section 1.2.1). Les *critères croissants* sont généralement à valeurs réelles et dépendent souvent des descripteurs des segments pour quantifier localement une énergie de fusion. Néanmoins, la définition de *critères croissants*, et donc de *hiérarchie indicée*, s'étend très facilement à toutes les fonctions d'un espace ordonné dans un autre puisqu'une notion de *croissance* de la fonction peut toujours y être définie.

Hiérarchie indicée sémantiquement Nous précisons ici une propriété, très similaire à la *hiérarchie indicée*, que doit satisfaire la **solution** à notre problème de segmentation. Soit $\mathcal{S}em : \boldsymbol{S} \rightarrow \boldsymbol{C}_h$ la fonction de classification de l'expert, aussi appelée *vérité terrain*, qui rattache un mot de vocabulaire du langage de l'expert à tout segment d'un partitionnement hiérarchique $\boldsymbol{S}$ sans commettre d'erreur. Cette fonction permet d'écrire la propriété d'*intrication* entre les hiérarchies spatiale et sémantique dans un formalisme très proche de celui de la Propriété 2 et de la Propriété 3 :

Propriété 4. *L'image par $\mathcal{S}em$ d'un segment est inférieure à celle de n'importe quel autre segment qui le contient :*

$$\forall s_i, s_j \in \boldsymbol{S}, s_i \in \Downarrow^{\infty}{}_{space}(s_j) \;\Rightarrow\; \mathcal{S}em(s_i) \in \Downarrow^{\infty}{}_{sem}(\mathcal{S}em(s_j))$$

Propriété 5. *L'image par $\mathcal{S}em$ d'un segment atomique est un axiome :*

$$\forall s \in \mathbb{A}\text{toms}(\boldsymbol{S}), \mathcal{S}em(s) \in \mathbb{A}\text{xioms}(\boldsymbol{C}_h)$$

La Propriété 5 interprète la valeur nulle rencontrée sur les feuilles dans la Propriété 3 comme la plus petite valeur que peut prendre le *critère croissant* dans l'espace ordonné ciblé, ici les *axiomes* du langage humain. Par la suite, on appellera *critère sémantique croissant* toute fonction de classement $\mathcal{S}em$ qui respecte les propriétés ci-dessus et on dira du triplet $(\boldsymbol{T}_{space}, \boldsymbol{T}_{sem}, \mathcal{S}em)$ qu'il forme une *segmentation hiérarchique indicée sémantiquement*.

3.4.4 Un problème d'optimisation

3.4.4.1 Isomorphisme de sous-graphes

La segmentation syntaxique est un problème d'*isomorphisme de sous-graphes*. Elle consiste à rechercher, dans une image dotée d'une structure de graphe, un sous-graphe d'intérêt appelé *motif* et noté m. Soit X une image et $\boldsymbol{S}$ une segmentation *plate* de cette image. Cette segmentation de X est naturellement dotée d'une structure de graphe, aussi appelée *graphe d'adjacence des régions*, notée $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ dont les nœuds sont les segments de $\boldsymbol{S}$, autrement dit $\boldsymbol{V} = \boldsymbol{S}$, et dont les arêtes sont les paires de segments adjacents dans $\boldsymbol{S}$. On dira ensuite qu'un graphe $\boldsymbol{G}_d = (\boldsymbol{V}_d, \boldsymbol{E}_d)$ est un sous-graphe de $\boldsymbol{G}$ si, et seulement si :

$$\boldsymbol{V}_d \subseteq \boldsymbol{V} \qquad \wedge \qquad \boldsymbol{E}_d = \boldsymbol{E} \cap \boldsymbol{V}_d^2$$

Le *motif* recherché, m, prend lui aussi la forme d'un graphe $\boldsymbol{G}_m = (\boldsymbol{V}_m, \boldsymbol{E}_m)$ dont les nœuds sont des segments hypothétiques, identifiés comme des parties de l'objet m. Le problème de segmentation est une recherche de l'ensemble $\boldsymbol{D}$ des sous-graphes de $\boldsymbol{G}$ tel que pour tout élément $\boldsymbol{G}_d \in \boldsymbol{D}$, il existe une *bijection* f de $\boldsymbol{V}_m$ dans $\boldsymbol{V}_d$ qui conserve la topologie du *motif*. Plus formellement,

$$\boldsymbol{D} = \{\boldsymbol{G}_d \mid \exists f : \boldsymbol{V}_m \to \boldsymbol{V}_d,\ \textit{bijective} \mid \forall (v_1, v_2) \in \boldsymbol{E}_m,\ (f(v_1), f(v_2)) \in \boldsymbol{E}_d\} \tag{3.4}$$

3.4.4.2 Contraintes supplémentaires

Dans le cas général, la recherche exhaustive d'un *motif* dans un graphe est un problème de la classe *NP-complet*, mais un certain nombre de stratégies, basées sur des heuristiques, permettent de réduire la complexité pour certains types de graphes et de motifs. Dans le cas de *motifs* fixes et de graphes *planaires* par exemple, Nešetřil and De Mendez [2008] proposent par exemple un algorithme pour résoudre le problème en temps linéaire.

Au coût de résolution pour un *motif*, viennent s'ajouter certaines contraintes relatives à notre tâche de segmentation. La première difficulté réside dans la pluralité des *motifs* rattachables à une catégorie d'objets $c \in \boldsymbol{C}_h$. En effet, les manifestations des concepts de l'expert dans une image tolèrent certaines variations dans le nombre ou l'agencement spatial de leurs briques constitutives. Cette liberté dans le nombre de nœuds et la disposition des arêtes d'un objet de classe c se décline en définissant une famille potentiellement infinie de *motifs*.

Enfin, pour garantir l'*interprétabilité* totale des décisions que prendra le système, il semble important de conserver la propriété d'*exhaustivité* dans la détection des structures. La procédure de segmentation syntaxique utilisée doit donc pouvoir détecter, dans un temps raisonnable, des *motifs* qui couvrent la totalité des catégories recherchées $\boldsymbol{C}_h$.

3.4.4.3 Grammaire et optimisation gloutonne

Une grammaire des motifs La gestion de la variabilité intra-classe dans les motifs ne peut être modélisée que par l'intermédiaire de règles de production, qui sont le pendant symbolique des *modèles génératifs* et *atlas* utilisés dans les approches non-syntaxiques. Une règle de production doit donc absolument être formulée par l'expert pour tous les éléments de $\boldsymbol{C}_h$ qu'il souhaite reconnaître automatiquement. Sur un plan plus technique, les notations propres à la grammaire que nous avions notamment utilisées dans la Sous-section 2.2.2 étaient bien adaptées à la topologie des chaînes, mais des voisinages supérieurs à 2 nous incitent fortement à modifier et ajouter certains symboles. L'utilisation d'un opérateur de voisinage

$\Leftrightarrow$ entre les symboles s'impose et l'utilisation des suffixes pourra parfois être reportée sur cet opérateur pour témoigner de voisinage multiple entre les symboles situés de part et d'autre de l'opérateur.

- « $\rightarrow$ » débutera la définition d'une règle de production,
- « ; » symbolisera la fin d'une règle de production,
- « | » symbolisera un choix possible entre les caractères de part et d'autre,
- le suffixe « $*$ » symbolisera la répétition du caractère précédent 0 ou plusieurs fois,
- le suffixe « $+$ » symbolisera la répétition du caractère précédent au moins 1 fois,
- le suffixe « ? » symbolisera un caractère optionnel ne pouvant être répété qu'une fois au maximum,
- les parenthèses « () » seront utilisées pour grouper des expressions.
- « $\Leftrightarrow$ » symbolisera une arête entre les motifs de part et d'autre, cet opérateur a la priorité sur « | »,
- « $\overset{*}{\Leftrightarrow}$ » symbolisera l'existence de 0 ou plusieurs arêtes entre une instance du motif de gauche et des instances du motif de droite,
- « $\overset{+}{\Leftrightarrow}$ » symbolisera l'existence d'au moins 1 arête entre une instance du motif de gauche et des instances du motif de droite,
- « $\overset{?}{\Leftrightarrow}$ » symbolisera l'existence optionnelle d'au plus 1 arête entre une instance du motif de gauche et des instances du motif de droite.

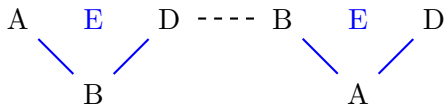
Construction de la hiérarchie par optimisation gloutonne Nous ne cessons de l'évoquer plus ou moins explicitement depuis le début de cette section, la solution $\boldsymbol{T}_{space}$ à notre problème de segmentation se construit selon une stratégie *ascendante*, c'est-à-dire par une méthode itérative de fusion des segments de l'image. Le critère de fusion, ou plutôt l'ordre des priorités de fusion, est établi par la grammaire des motifs. La règle élémentaire est de commencer par les productions de sémantique faible : la détection de tous les segments de catégorie c n'est garantie que si la totalité des segments dont la classe *justifie* c ont été reconnus. Les fusions qui mènent aux segments de classes $\mathbb{J}(c)$ sont donc prioritaires par rapport à celles qui impliquent des segments de classe c.

Un exemple Soit $\{A, B, C, D, E\} \subset \boldsymbol{C}_h$ tel que $\{A, B, D\} \subset \mathbb{A}\text{xioms}(\boldsymbol{C}_h)$ et tel que $\Downarrow_{sem}(C) = \{A, B\}$ et $\Downarrow_{sem}(E) = \{C, D\}$. On propose pour ce sous-espace sémantique la grammaire suivante :

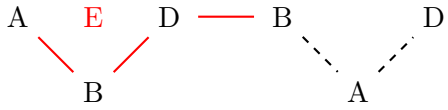
$$\texttt{C} \rightarrow \texttt{A} \Leftrightarrow \texttt{B} \ ;$$
$$\texttt{E} \rightarrow \texttt{C} \Leftrightarrow \texttt{D} \overset{?}{\Leftrightarrow} \texttt{B} \ ;$$

Si elle veut récupérer la totalité des instances du concept E dans l'image, la procédure de fusion doit absolument commencer par détecter toutes les instances de la classe C avant d'intégrer la règle de production de E. La Figure 3.8 présente notamment l'effet d'une inversion de priorité sur ce cas particulier.

Un problème mal posé La recherche de motifs par fusion de régions laisse cependant des configurations ambiguës dans l'image. Dans ce cas, le respect strict des priorités de fusions de régions ne conduit pas à une partition unique, Figure 3.9. La meilleure option est évidemment celle qui permet par la suite de construire les objets de niveaux sémantiques les plus élevés. Cependant, aucune information ne permet de le savoir *a priori* et le conflit ne peut être résolu qu'après construction complète de l'arbre de segmentation. Malheureusement, ces évènements



(a) Détection de toutes les instances de `E`



(b) Instance de `E` manquée

FIGURE 3.8 – Règles de priorité des fusions. Les traits pleins représentent des fusions réalisées. Les traits en pointillés sont des fusions possibles mais non-réalisées. (a) Les fusions construisent les concepts de plus faible sémantique en priorité, aucun segment de catégorie `E` n'est manqué. (b) Une fusion optionnelle $\texttt{D} \overset{?}{\Leftrightarrow} \texttt{B}$ a été réalisée avant une fusion $\texttt{A} \Leftrightarrow \texttt{B}$ de sémantique plus faible, toutes les instances de `E` n'ont pas été détectées.

FIGURE 3.9 – Ambiguité dans la segmentation des instances du concept `E`. (a) et (b) sont toutes deux des configurations recevables qui peuvent être obtenues en respectant l'ordre des priorités. Les deux arêtes $\texttt{B} \Leftrightarrow \texttt{D}$ sont de même priorité et aucune règle ne permet *a priori* de choisir l'une plutôt que l'autre.

restent difficiles à détecter que ce soit avant, pendant ou après la construction de l'arbre. De plus, l'algorithme pour trouver la solution optimale explore naïvement toutes les possibilités. Cela revient à reconstruire un arbre différent pour chaque configuration ambiguë ainsi que pour chaque combinaison de configurations ambiguës. La complexité algorithmique du problème est donc exponentielle en nombre d'ambiguités et il est souvent préférable de modifier la grammaire des motifs pour éviter ou limiter l'apparition de ces évènements au cours de la procédure de fusion.

Le problème des motifs absurdes On dira d'un motif qu'il est *absurde* si aucune règle de production de la grammaire ne permet de le générer. L'apparition de telles structures semble contre-intuitive, mais l'évènement survient lorsque seuls les *axiomes* situés de part et d'autre d'une arête sont pris en compte pour décider de sa fusion au lieu d'utiliser les catégories de plus haut niveau sémantique des segments qui les contiennent, voir la Figure 3.10. Les motifs absurdes peuvent donc être évités à condition qu'une opération de *parsing* soit réalisée sur chaque nouveau segment créé pour s'assurer que sa structure n'a pas été générée par un motif d'ordre sémantique supérieur.

3.4.5 Segmentation de noyaux de cellules en fluorescence

Ce sont des réflexions menées au début de ces travaux de thèse qui conduisent à la méthodologie de segmentation *syntaxique* que nous venons d'aborder. Les algorithmes développés alors sont détaillés dans Abreu et al. [2017]. Ils sont dédiés à la segmentation de noyaux de

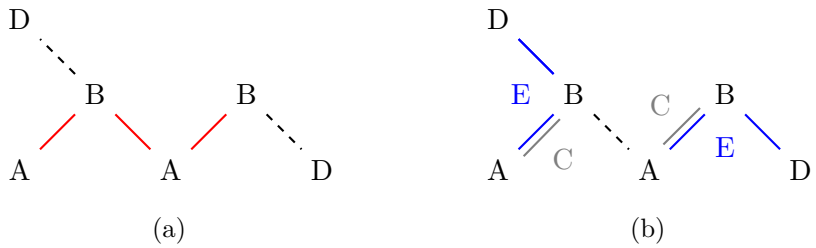


FIGURE 3.10 – Génération d'un motif absurde. (a) Les lignes rouges décrivent un modèle absurde. Il n'est pas décrit par la grammaire, bien qu'il soit visiblement obtenu en effectuant les fusions `A` ⇔ `B` qui semblent prioritaires dans la grammaire donnée en exemple. (b) Lorsqu'une fusion `A` ⇔ `B` est perpétrée, le segment obtenu est automatiquement étiqueté avec le concept `C`, rendant impossible toute nouvelle liaison avec une instance de `A` ou de `B`.

cellules dans des images de microscopie de fluorescence, intègrent dès le départ une notion de hiérarchie spatiale et de détection d'objets par assemblage de leurs parties. Les notions de parties ou segments étiquetés par des termes du langage humain et d'assemblage guidé par les connaissances n'apparaissent que dans la seconde version de la méthode, développée dans Abreu et al. [2018]. L'ordre des fusions ne dépend que des catégories des segments, mais n'a pas encore été formulé par des opérateurs logiques. La connaissance des objets, relative à la structuration de leurs parties, n'est pas non plus formalisée dans une grammaire et le choix des priorités de fusions n'est donc justifié que par l'intuition.

Nous l'avons déjà annoncé plus haut, cet algorithme trouve difficilement sa place dans l'état de l'art de la segmentation des noyaux en fluorescence. Ce domaine applicatif semble désormais régi par des réseaux neuronaux convolutifs. Le problème principal de la segmentation de noyaux abordée avec des modèles d'apprentissage automatique est celui du manque d'annotations. Les noyaux sont des structures petites, nombreuses et d'aspect extrêmement variable dans les images de microscopie. Le décompte et le détourage manuel de ces objets est si fastidieux qu'aucune base d'apprentissage décente n'a pu être constituée pour entraîner des modèles.

Les *CNNs* utilisés dans ce cas sont ce qu'on appelle des réseaux *antagonistes génératifs* (*Generative Adversarial Networks, GANs*) et, sans entrer dans le détail de l'architecture ou de l'entraînement de ces modèles, nous notons qu'ils ont la particularité de générer des images synthétiques, qui peuvent permettre d'accroître la taille des ensembles d'apprentissage pour des réseaux de segmentation classiques, comme cela est le cas dans les travaux de Fu et al. [2018]. Dans leur version dite *cyclique*, les *GANs* peuvent même être entraînés sans appariement des données d'apprentissage (image de noyaux et sa segmentation associée). Dans ce cas, Mahmood et al. [2019] montre que l'annotation n'est plus vraiment un problème. Il suffit de produire des masques de segmentation aléatoires plausibles pour entraîner le modèle.

Notre algorithme ne saurait donc être comparé avec de telles approches. Sa méthode de construction d'arbres par fusions de régions l'inscrit dans l'ensemble abondant des segmentations de noyaux orientées graphes. Si elle s'en démarque par un rattachement plus franc au raisonnement et à la reconnaissance syntaxique, la comparaison et l'évaluation dans ce contexte étaient bien plus légitimes.

Nous tentons donc ici de revisiter cette procédure comme un cas particulier du formalisme déployé dans la première partie de cette section. Cette rétrospective est l'occasion de présenter une version plus complète de la méthode et d'en combler certaines lacunes que le manque de recul et la brièveté imposée des communications ont laissé s'installer.

3.4.5.1 Introduction

La plupart des approches que nous avons jugées « comparables » sont basées sur la segmentation par *ligne de partage des eaux* et, plus précisément, sur une version de l'algorithme initialisée par des *graines*, Ortiz De Solórzano et al. [1999], Chawla et al. [2004], Lin et al. [2003, 2005], Adiga and Chaudhuri [2001], Wählby et al. [2004], Lin et al. [2007b]. Dans ce cas, une procédure doit détecter les centres des noyaux sur lesquels sont placés des *bassins*. Ces bassins vont progressivement se remplir, en commençant par les zones de gradient faible, c'est-à-dire les zones les plus homogènes de l'image, et la croissance des bassins cesse lorsque deux bassins entrent en collision.

Cette méthode à une tendance à produire des sur-segmentations de l'image, ce qui signifie qu'un objet est généralement séparé en plusieurs segments. Cela est principalement lié aux méthodes de détection des centres qui ont tendance à produire des faux positifs. Chawla et al. [2004], Lin et al. [2003, 2005] et Adiga and Chaudhuri [2001] corrigent ce défaut, en utilisant des modèles de noyaux pour guider une procédure de fusion de segments, vers une partition qui place chaque noyau dans un segment unique. Cependant, l'exploration des configurations de fusion pour reconnaître un modèle de noyau est combinatoirement défavorable et repose sur des heuristiques difficiles à mettre en place et dont l'exécution est lente. C'est à ce problème que la présente méthode tente d'apporter une solution originale.

3.4.5.2 Franchissement du fossé sémantique

Bien que ce ne soit pas exactement le propos de cette section, il paraît tout de même pertinent d'expliquer comment le point de départ, la *segmentation atomique axiomatique* $\boldsymbol{S}_0$, est constitué. En microscopie de fluorescence, le signal lumineux émis par la molécule fluorescente (*DAPI*), qui se lie fortement à l'*ADN*, révèle la présence des noyaux des cellules. L'hétérogénéité en densité du matériel génétique dans le noyau produit un signal lumineux dont la variance intra/inter-noyaux est très importante. De plus, les noyaux des cellules se regroupent généralement sous forme d'amas compacts et deviennent difficiles à individualiser. L'entrée de la méthode est un relevé du signal lumineux émis par le *DAPI* dans une image numérique X. Le partitionnement de X en $\boldsymbol{S}_0$ se déroule en trois temps.

Partition spatiale initiale Tout d'abord, les pixels de l'image sont regroupés, par le biais d'une méthode de segmentation à *contrainte sémantique faible*. Les segments obtenus doivent être de taille inférieure à celle des noyaux de cellules, on parle ici de *sur-segmentation*. Cette condition sur la taille est de première importance. En effet, si des objets sont construits par assemblage de segments, ces derniers doivent être plus petits que les objets recherchés. Une autre « bonne » propriété des segments est de respecter au mieux les contours des noyaux. Un certain nombre d'algorithmes de *superpixels* peuvent être utilisés et paramétrés pour remplir les conditions ci-dessus et la plupart des outils présentés dans la Sous-section 1.2.1 sont susceptibles de fournir des résultats satisfaisants. Notre choix s'est porté sur la méthode des *waterpixels* de Machairas [2016] et Machairas et al. [2015] parce que son implémentation est aisée, son exécution rapide et son respect des propriétés précédentes est très bon. Le fonctionnement de la méthode est simple : une grille régulière est placé sur l'image, puis déformée pour que ses lignes épousent les bords des objets (gradients élevés de l'image). La déformation est modérée par un terme de régularisation qui doit être ajusté manuellement.

Description des segments Une fois la partition obtenue, un vecteur descripteur est calculé pour chaque segment. Ici encore, nous pouvons envisager une large gamme de méthodes parmi les solutions évoquées dans la Section 1.3. Dans ce travail nous utilisons 3 types de caractéristiques : des descripteurs basés sur l'intensité des pixels, des descripteurs basés sur les

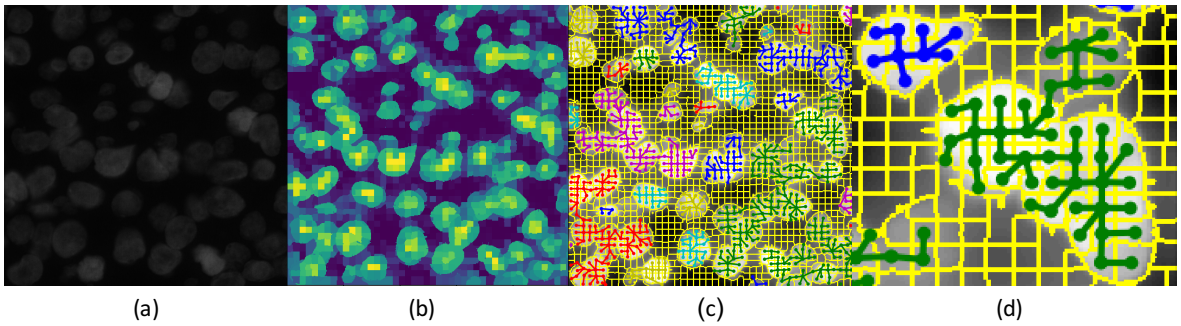


FIGURE 3.11 – (a) Image *DAPI* en niveaux de gris. (b) Classement des superpixels, les superpixels sombres sont du fond, les superpixels verts sont des bords et les plus jaunes sont des centres. (c) Arbre couvrant de poids minimal. (d) Zoom sur une partie de l'arbre couvrant, les noyaux sont séparés par une unique arête, y compris dans les amas les plus compacts.

gradients rencontrés dans le segment et des descripteurs morphologiques du segment. Chaque région s de l'image est ainsi décrite par un vecteur caractéristique x :

$$\mathbf{x}(s) = ((h_i)_{i\in[1...10]}, (g_i)_{i\in[1...10]}, a, c, dx, dy, r),$$

dans lequel (h_i) et (g_i) sont respectivement des histogrammes à 10 boîtes des valeurs d'intensité et de gradient du segment, $a = |s|$ est l'aire du segment, c est une mesure de convexité du segment, calculée comme le rapport entre l'aire de s et celle de son enveloppe convexe, dx et dy sont les dimensions de la boîte englobante de s et r est une mesure de rectangularité du segment, calculée comme le rapport entre l'aire de s et celle de sa boîte englobante. Le vecteur caractéristique résultant compte 25 composantes.

Partition sémantique initiale Pour constituer $\boldsymbol{S}_0$ il faut encore réaliser un classement des segments de l'image sur la base des descripteurs. Le classement des segments doit rendre possible leur assemblage en structures d'intérêt. Pour la détection des noyaux, un segment de la partition initiale peut appartenir à l'une des 3 catégories suivantes : centre de noyau (0), bord de noyau (1) et fond de l'image (2). L'apprentissage d'un modèle de classification est rendu possible par annotation manuelle de superpixels. La plupart des classifieurs étudiés dans la Section 1.4 peuvent être entraînés à réaliser cette tâche, mais sans connaissance sur la structure de l'espace caractéristique et pour un fonctionnement rapide sans qu'une trop grande quantité de superpixels n'ait besoin d'être annotée, nous choisissons l'algorithme des *Forêts Aléatoires* (*Random Forests, (RF)*). Un exemple de partition *atomique axiomatique* produite par succession de ces trois étapes de franchissement du fossé sémantique est présenté dans la Figure 3.11.

3.4.5.3 Logique d'assemblage

Intuition et topologie en étoile Sur la base de notre jeu d'*axiomes*, $\mathbb{A}\text{xioms}(\boldsymbol{C}_h) = \{\texttt{Centre}, \texttt{Bord}, \texttt{Fond}\}$, il s'agit à présent de définir les règles de priorité des fusions. Lorsque l'on considère la tâche de segmentation des noyaux par fusions de régions, la problématique sera de s'assurer que deux noyaux ne sont regroupés qu'après qu'ils ont été construits individuellement correctement. Cela donne naturellement la priorité aux liaisons $\texttt{Centre} \Leftrightarrow \texttt{Bord}$ avant que les liaisons $\texttt{Bord} \Leftrightarrow \texttt{Bord}$, susceptibles de réunir deux noyaux, ne soient considérées, voir la Figure 3.12. Cette idée aboutit à une logique d'agrégation extrêmement simple. Soient les segments axiomatiques voisins s_i et s_j de catégories $c_i, c_j \in [0, 2]$ respectivement, alors la priorité de fusion entre un segment contenant s_i et un autre contenant s_j est donnée par :

$$priorité(s_i, s_j) = \max(c_i, c_j)$$

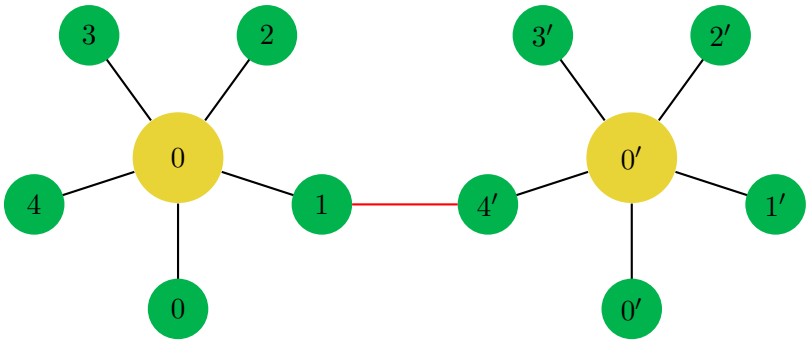


FIGURE 3.12 – Les configurations en étoiles sont révélatrices de la présence des noyaux. Les segments de `Bord` sont représentés en vert et ceux de `Centre` sont en jaune. Les arêtes en noir sont les fusions prioritaires. La fusion rouge est une liaison `Bord` ⇔ `Bord` et doit être réalisée plus tard sous peine de créer des motifs absurdes.

et cette logique d'agrégation amène une topologie de segmentation en étoile. Elle est observable dans les étapes intermédiaires de la procédure, Figure 3.11, et est généralement caractéristique des tâches d'individualisation d'objets.

Une grammaire pour la segmentation des noyaux L'analyse de l'image *DAPI* complète selon notre méthodologie ne peut pas se contenter de segmenter les noyaux de cellules. L'*exhaustivité* impose également la déduction de toutes les structures d'ordres supérieurs. Parmi ces structures, nous compterons les `Synapses` qui correspondent à un contact physique entre deux cellules, puis les `Amas` qui sont des chaînes de noyaux connexes. Nous en convenons, ces images ne sont sémantiquement pas très riches et ne permettent probablement pas d'étudier tout le potentiel de l'algorithme. Elles font toutefois l'objet d'une grammaire, ce qui nous autorise à étudier notre méthode dans son intégralité et à relever quelques propriétés destinées à faciliter sa formulation pour des cas plus généraux.

```
Image → Fond ⇔ ( Amas | Synapse | Noyau ) | Image ⇔ Quelconque ;
 Fond → Fond | Fond ⇔ Fond ;
 Amas → Synapse ⇔ ( Noyau | Synapse ) | Amas ⇔ ( Amas | Synapse | Noyau ) ;
Synapse → Noyau ⇔ Noyau ;
Noyau → Centre ⇔ Bord | Noyau ⇔ Bord ;
```

FIGURE 3.13 – La grammaire propice à la segmentation syntaxique des images de *DAPI*.

Cette grammaire est présentée dans la Figure 3.13. Elle montre notamment la récurrence d'une certaine forme d'expression dans son écriture :

```
A → Préfixe | ( A ⇔ Suffixe ) ;
```

Cette règle de production rejoint parfaitement le raisonnement du Paragraphe 3.3.3.2 qui séparait alors les fusions de construction de l'arbre de subsomption en deux catégories, les *ramasse-miettes* et les véritables créations de concepts. Ici, la partie gauche de la formule, appelée `Préfixe`, correspond à ce cas de création de concept : le segment de concept `A` est formé par la fusion de deux concepts qui, pris individuellement, ne suffisent pas à prédire sa présence. En revanche, la partie droite de la règle, correspond à un cas d'expansion d'un segment de type `A`. L'objet de catégorie `A` y est déjà détecté et il ne fait que se compléter.

Il est particulièrement avantageux de définir les règles de production des motifs de cette façon, parce qu'elle simplifie grandement l'étape de *parsing* qui doit être exécutée après chaque fusion pour ré-étiqueter le nouveau segment produit. En effet, si l'on reprend s, issu de la fusion de s_i et s_j de classes c_i et c_j telles que le segment s_i est sémantiquement plus élevé que le segment s_j, autrement dit $c_i > c_j$, alors il s'agit simplement de savoir si c_j est un `Suffixe` de c_i, auquel cas la classe de s sera naturellement c_i, ou si $c_i \Leftrightarrow c_j$ se trouve dans les `Préfixes` des classes sémantiquement supérieures à c_i.

3.4.5.4 Algorithme de segmentation

La procédure décrite jusqu'à présent est initialisée par la *segmentation atomique axiomatique* $\boldsymbol{S}_0$. Nous considérons ensuite le *graphe d'adjacence des régions* de cette partition, $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ dans lequel les arêtes se voient attribuer une valeur de priorité de fusion motivée par les liaisons $\Leftrightarrow$ de la grammaire des images, de sorte que les objets de niveau sémantique faible sont construits les premiers. Chaque arête est ensuite parcourue en suivant l'ordre des priorités. Tout segment créé est automatiquement étiqueté en identifiant le `Préfixe` ou le `Suffixe` d'un concept dans la grammaire. En conservant une référence sur chaque objet créé lors de la procédure, une segmentation hiérarchique indicée sémantiquement est formée. L'algorithme est en tout point similaire à l'Algorithme 2 et la structure de $\boldsymbol{T}_{spatial}$ est celle d'un **arbre couvrant de poids sémantique maximal**.

C'est sur ce point que l'algorithme initial diffère et perd finalement son lien avec le raisonnement logique. En effet, dans sa version originale notre méthode construisait un *Arbre couvrant* de segmentation sans étiqueter les segments au fur et à mesure. Une seconde procédure parcourait ensuite la totalité des sous-arbres afin de détecter des structures d'intérêt. Un classifieur supervisé était par exemple entraîné à distinguer les noyaux de cellules de tout autre type de segment. C'est la structuration de la segmentation en arbre binaire qui rendait la visite des sous-arbres rapide et la détection des noyaux efficace puisqu'un seul parcours des arêtes (coupes) de l'arbre suffit à explorer exhaustivement tous les segments de la hiérarchie. Une version de cette procédure est donnée par l'Algorithme 3.

Algorithme 3 : Algorithme de coupe

Données : $\boldsymbol{T}_{spatial} = (\boldsymbol{V}, \boldsymbol{E})$; `// Segmentation hiérarchique (arbre binaire)`
Résultat : N ; `// Ensemble des segments de noyaux`

$N = \emptyset$;
tant que $\boldsymbol{V} \neq \emptyset$ **faire**
  pour chaque $e = (s_1, s_2) \in E$ **faire**
    $(V', E') \leftarrow (V, E)$; `// Copie de l'arbre`
    $E' \leftarrow E \setminus e$; `// L'arête` e `est retirée de l'arbre`
    $(t_1, t_2) \leftarrow$ Composantes-connexes$((V', E'))$; `// Extraction des sous-arbres`
    $S \leftarrow S \cup \{T, t_1, t_2\}$; `// Stockage des sous-arbres`
    $t^* \leftarrow \arg\max_{(t \in S)}(P[t = \text{nucleus}])$; `// Étiquetage`
    $T \leftarrow T \setminus t^*$;
    $N \leftarrow N \cup s(t^*)$;
  fin
fin
retourner N ;

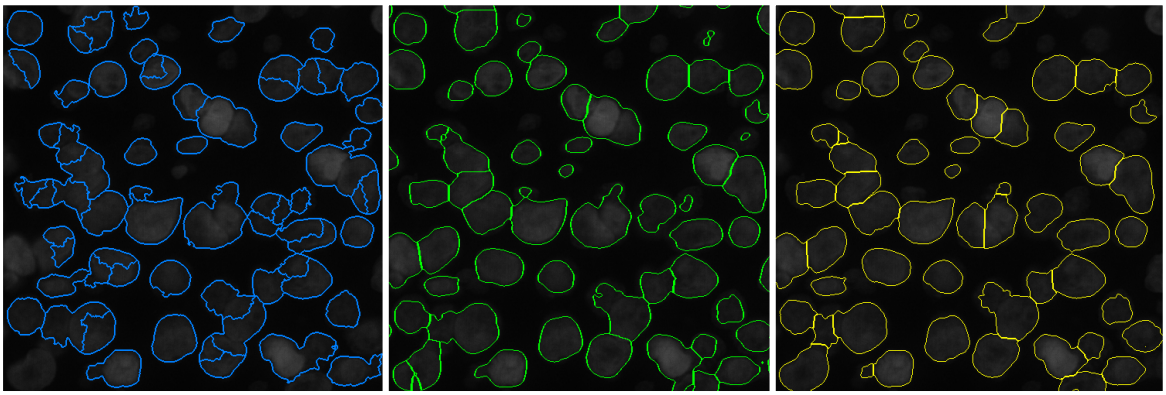


FIGURE 3.14 – (a) Segmentation CellProfiler. Les noyaux sont globalement sur-segmentés et l'on observe également un certain nombre d'agglomérats sous-segmentés. (b) Segmentation Fiji. La sur-segmentation est considérablement réduite par rapport à la procédure de CellProfiler. (c) Segmentation par *Arbre couvrant*. La méthode semble combler les lacunes des procédures précédentes.

TABLEAU 3.1 – Jeux de données de *superpixels* destinés au franchissement du *fossé sémantique*

Data sets	**Background**	**Inter**	**Borders**	**Centers**
Training	44142	8466	22296	4536
Testing	22071	4233	11148	2268
Total	66213	12699	33444	6804

3.4.5.5 Expériences et évaluation

Acquisition Les *WSI* de *DAPI* ont été acquises avec un Pannoramic 250 Flash (P250 Flash digital microscopes, 3DHISTECH, Hungary) en fluorescence, à une résolution de 0.16 μm/pixel.

Entraînement Un ensemble d'apprentissage comptant un total de 13240 waterpixels a été annoté manuellement pour entraîner le classifieur chargé de franchir le *fossé sémantique*, et 518 régions ont été sélectionnées pour entraîner le modèle de noyaux. Dans l'objectif de fiabiliser les classifications, des transformations simples ont été appliquées au jeu de données tels que l'ajout d'un bruit et d'un flou gaussien. Pour les annotations considérées, plusieurs partitions initiales sont proposées pour différents paramétrages des waterpixels de manière à rendre la méthode relativement insensible à la forme des superpixels initiaux. En appliquant ces opérations de bruit, de flou et d'altérations morphologiques des partitions initiales, nous parvenons à multiplier par un facteur 9 le nombre d'échantillons dans les ensembles d'apprentissage et de validation des classifieurs, voir le Tableau 3.1 et le Tableau 3.2.

TABLEAU 3.2 – Jeux de données de segments destinés à la distinction des noyaux

Data sets	**Nuclei**	**Non-nuclei**
Training	1140	1968
Testing	570	984
Total	1710	2952

Validation et évaluation Les segmentations tests ont été réalisées sur 28 images *DAPI* de taille 500×500 pixels qui ne faisaient pas partie de l'ensemble d'apprentissage. Nous avons travaillé sur des images utilisées en routine clinique, toutes choisies pour la difficulté de la tâche de segmentation, autrement dit, ces images sont bruitées, de faible contraste et présentent des noyaux densément agglomérés. La partition en *waterpixels* a été réalisée en appliquant un coefficient de régularisation $k = 0.3$ et le nombre de *superpixels* moyen obtenu sur une image est de 2000, nous redirigeons ici le lecteur vers les travaux de Machairas [2016] et Machairas et al. [2015] pour obtenir plus de précisions sur l'algorithme des *waterpixels*. Tous les classifieurs utilisés dans ce travail sont des *Forêts Aléatoires* initialisées avec 2000 arbres, ce qui fournit un bon compromis entre vitesse d'apprentissage et précision dans la prédiction.

Nous comparons notre approche avec 2 logiciels *open-source*, CellProfiler [4] Kamentsky et al. [2011] et Fiji [5] Schindelin et al. [2012], ainsi qu'avec le logiciel commercial HALO [6]. Pour chacune des méthodes de segmentation utilisées, le Tableau 3.3 et la Figure 3.14 présentent la *précision*, le *rappel* et le *F-score*. Pour la segmentation selon *Fiji*, nous utilisons la méthode d'Otsu pour fixer le paramètre de seuil de la méthode afin de rendre la méthode la moins subjective possible. Pour la méthode implémentée dans *CellProfiler*, nous avons également limité le diamètre standard des objets (en pixels) à l'intervalle $[20, 80]$, en accord avec la taille moyenne des noyaux relevée dans l'ensemble d'apprentissage.

Pour constituer la vérité terrain, nous avons demandé à un expert de placer un point au centre de chaque noyau dans nos 28 images. Un masque binaire a ensuite été formé en plaçant des disques de diamètre 5 pixels centrés sur les annotations humaines. Un vrai-positif, VP, correspond à un segment qui contient exactement un disque. Tout autre segment est considéré comme un faux-positif, FP, et un disque qui ne se trouve dans aucun segment est compté comme un faux-négatif, FN. Selon ces notations, nous rappelons comment des métriques de détection peuvent être calculées pour les segmentations produites :

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

$$Fscore = \frac{2TP}{2TP + FP + FN}$$

Bien que les *F-scores* soient inférieurs à 0.5 (Tableau 3.3), nous notons que dans un grand nombre d'applications, tel que le décompte des évènements de *break-apart* dans les lames de *FISH*, dont l'exemple précis est présenté par Mueller et al. [2013], les segments obtenus n'en demeurent pas moins utiles puisque les *WSI* contiennent des millions de cellules et même un faible taux de bonnes détections suffit à établir des résultats statistiquement fiables. Cela étant dit, le Tableau 3.3 indique que notre algorithme affiche des scores plus satisfaisants que les logiciels *open-source* et commerciaux utilisés dans cette expérience sur les métriques de détection envisagées.

D'un point de vue algorithmique, notre intuition initiale était que le parcours des arêtes de l'arbre binaire de segmentation serait moins coûteux que les explorations de fusions plus classiques, tel que l'*Arbre des fusions* proposé par Lin et al. [2005] et Adiga and Chaudhuri [2001] qui consiste à explorer de manière exhaustive, mais non-redondante, les arbres de fusions pour chaque nœud du graphe d'*adjacence des régions* de la partition initiale en fixant

4. http://cellprofiler.org/examples/#HumanCells
5. https://imagej.net/Nuclei_Watershed_Separation
6. http://www.indicalab.com/

TABLEAU 3.3 – Évalution des algorithmes

	CellProfiler	Fiji	HALO	Arbre couvrant
Recall	0.41	**0.45**	0.40	0.41
Precision	0.40	0.40	0.46	**0.50**
F-score	0.40	0.42	0.43	**0.45**

arbitrairement une profondeur maximale d'exploration. Sur la base du pseudo-code fourni dans les travaux de Lin et al. [2005] et leur concept de *root path sets*, nous avons comparé les temps d'exécution en *Python* pur de leur structure d'*Arbre des fusions* avec le parcours de notre *Arbre couvrant* pour différentes profondeurs (Figure 3.15) et avons remarqué que l'exploration des arêtes de l'*Arbre couvrant* est vraisemblablement plus rapide que l'exploration exhaustive des fusions autour de chaque segment de la partition initiale.

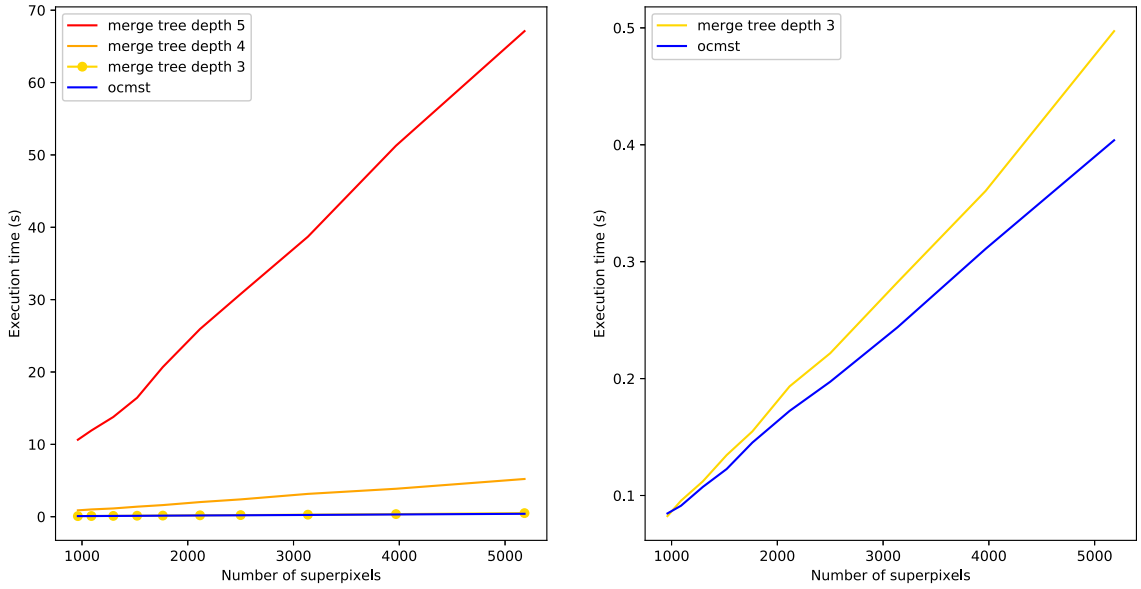


FIGURE 3.15 – Comparaison des temps d'exécution. Gauche : l'*Arbre couvrant* est comparé à l'*Arbre des fusions* pour différentes profondeurs. Droite : l'*Arbre couvrant* est comparé à l'*Arbre des fusions* pour une profondeur de 3. Dans une partition de *superpixels*, le nombre de configurations de fusions augmente exponentiellement avec le paramètre de profondeur. De plus, pour des nombres de *superpixels* élevés, les noyaux sont généralement coupés en plus de 3 fragments et l'*Arbre des fusions* dans ce cas peut conduire à des durées d'exécution excessives.

3.4.6 Bilan

Cette section faisait d'avantage référence à des approches de l'« Intelligence Artificielle » dites *symboliques*. La méthode de segmentation que j'ai présentée propose un fanchissement exhaustif, mais précoce, du fossé sémantique de façon à déduire les structures les plus complexes, et non à les reconnaître ou à les percevoir directement comme le font les modèles d'apprentissage profond. La méthode détourne l'usage des structures arborescentes de segmentations hiérarchiques, obtenues par des algorithmes de fusion de régions de l'image, en véritables arbres décisionnels.

Ces principes, destinés à améliorer l'interprétabilité des outils de segmentation, nous font reconsidérer les premiers développements réalisés dans ces travaux de thèse en inscrivant les intuitions qui guidaient alors la conception de mes premiers algorithmes dans un formalisme plus rigoureux et plus général.

La segmentation des noyaux de cellules dans les images de microscopie de fluorescence ne constitue peut-être pas l'exemple le plus adapté pour étudier le potentiel d'interprétabilité de la méthode. En effet, les étapes intermédiaires de la constitution d'un noyau de cellules sont d'un intérêt extrêmement limité pour l'expert. La technique devrait donc être implémentée pour la segmentation de structures plus complexes pour lesquelles une justification est effectivement nécessaire.

Il est cependant important de noter la complexité dans l'énonciation de la grammaire formelle utilisée pour segmenter des structures aussi simples. La création de cette grammaire doit évidemmment être compatible avec l'algorithme de fusion utilisé pour construire la segmentation hiérarchique. Il faut alors avouer que des règles claires pour écrire cette grammaire n'ont pas vraiment été données dans la description de la méthode et relèvent encore de l'intuition. La validation n'est possible qu'en évaluant la méthode sur diverses métriques de segmentation, détection ou classification de structures. La correction de la grammaire n'est pas non plus aisée, puisque de multiples causes (mauvaise définition de la grammaire ou grammaire incompatible avec l'algorithme de construction de l'arbre de segmentation) peuvent expliquer les mauvaises performances de la solution.

Loin de vouloir supplanter les approches purement orientées sur la perception, cette méthodologie a pour objectif de se placer en complément. Elle vise a apporter des justifications pour les cas les plus difficiles ou litigieux vis-à-vis de l'avis de l'expert. En plus d'être un moyen d'expliquer les décisions de la machine, elle apporte un retour vers l'expert qui permet notamment une correction efficace de la connaissance du système.

3.5 Références

Andrew Janowczyk and Anant Madabhushi. Deep learning for digital pathology image analysis : A comprehensive tutorial with selected use cases. *Journal of pathology informatics*, 7, 2016. 84

Neslihan Bayramoglu, Juho Kannala, and Janne Heikkila. Deep learning for magnification independent breast cancer histopathology image classification. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 2440–2445, Cancun, December 2016. IEEE. ISBN 978-1-5090-4847-2. doi : 10.1109/ICPR.2016.7900002. URL http://ieeexplore.ieee.org/document/7900002/. 85

Babak Ehteshami Bejnordi, Jimmy Lin, Ben Glass, Maeve Mullooly, Gretchen L Gierach, Mark E Sherman, Nico Karssemeijer, Jeroen van der Laak, and Andrew H Beck. DEEP LEARNING-BASED ASSESSMENT OF TUMOR-ASSOCIATED STROMA FOR DIAGNOSING BREAST CANCER IN HISTOPATHOLOGY IMAGES. *Proceedings. IEEE International Symposium on Biomedical Imaging*, 2017 :929–932, April 2017. ISSN 1945-7928. doi : 10.1109/ISBI.2017.7950668. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6802272/. 85

Monjoy Saha, Chandan Chakraborty, and Daniel Racoceanu. Efficient deep learning model for mitosis detection using breast histopathology images. *Computerized Medical Imaging and Graphics*, 64 :29–40, March 2018. ISSN 08956111. doi : 10.1016/j.compmedimag.2017.12.001. URL https://linkinghub.elsevier.com/retrieve/pii/S0895611117301222. 85

Nicolas Coudray, Paolo Santiago Ocampo, Theodore Sakellaropoulos, Navneet Narula, Matija Snuderl, David Fenyö, Andre L. Moreira, Narges Razavian, and Aristotelis Tsirigos. Classification and mutation prediction from non–small cell lung cancer histopathology images using deep learning. *Nature medicine*, 24(10) :1559–1567, 2018. 85, 89

Guilherme Aresta, Teresa Araújo, Scotty Kwok, Sai Saketh Chennamsetty, Mohammed Safwan, Varghese Alex, Bahram Marami, Marcel Prastawa, Monica Chan, Michael Donovan, Gerardo Fernandez, Jack Zeineh, Matthias Kohl, Christoph Walz, Florian Ludwig, Stefan Braunewell, Maximilian Baust, Quoc Dang Vu, Minh Nguyen Nhat To, Eal Kim, Jin Tae Kwak, Sameh Galal, Veronica Sanchez-Freire, Nadia Brancati, Maria Frucci, Daniel Riccio, Yaqi Wang, Lingling Sun, Kaiqiang Ma, Jiannan Fang, Ismael Kone, Lahsen Boulmane, Aurélio Campilho, Catarina Eloy, António Polónia, and Paulo Aguiar. BACH : Grand Challenge on Breast Cancer Histology Images. *Medical Image Analysis*, 56 :122–139, August 2019. ISSN 13618415. doi : 10.1016/j.media.2019.05.010. URL http://arxiv.org/abs/1808.04277. arXiv : 1808.04277. 85

Rucha Tambe, Sarang Mahajan, Urmil Shah, Mohit Agrawal, and Bhushan Garware. Towards Designing an Automated Classification of Lymphoma subtypes using Deep Neural Networks. pages 143–149, January 2019. doi : 10.1145/3297001.3297019. 85

Andrew Gordon Wilson, Jason Yosinski, Patrice Simard, Rich Caruana, and William Herlands. Proceedings of NIPS 2017 Symposium on Interpretable Machine Learning. *arXiv :1711.09889 [stat]*, December 2017. URL http://arxiv.org/abs/1711.09889. arXiv : 1711.09889 version : 3. 85

Christopher J. Kelly, Alan Karthikesalingam, Mustafa Suleyman, Greg Corrado, and Dominic King. Key challenges for delivering clinical impact with artificial intelligence. *BMC medicine*, 17(1) :195, 2019. 85

Andreas Holzinger, Bernd Malle, Peter Kieseberg, Peter M. Roth, Heimo Müller, Robert Reihs, and Kurt Zatloukal. Towards the Augmented Pathologist : Challenges of Explainable-AI in Digital Pathology. *arXiv :1712.06657 [cs, stat]*, December 2017a. URL http://arxiv.org/abs/1712.06657. arXiv : 1712.06657. 85

Andreas Holzinger, Chris Biemann, Constantinos S. Pattichis, and Douglas B. Kell. What do we need to build explainable AI systems for the medical domain ? *arXiv :1712.09923 [cs, stat]*, December 2017b. URL http://arxiv.org/abs/1712.09923. arXiv : 1712.09923. 85

Yoichiro Yamamoto, Toyonori Tsuzuki, Jun Akatsuka, Masao Ueki, Hiromu Morikawa, Yasushi Numata, Taishi Takahara, Takuji Tsuyuki, Kotaro Tsutsumi, Ryuto Nakazawa, Akira Shimizu, Ichiro Maeda, Shinichi Tsuchiya, Hiroyuki Kanno, Yukihiro Kondo, Manabu Fukumoto, Gen Tamiya, Naonori Ueda, and Go Kimura. Automated acquisition of explainable knowledge from unannotated histopathology images. *Nature Communications*, 10(1) :5642, December 2019. ISSN 2041-1723. doi : 10.1038/s41467-019-13647-8. URL http://www.nature.com/articles/s41467-019-13647-8. 86, 98

NasirM Rajpoot, Hesham El-Daly, AdnanM Khan, and Emma Simmons. HyMaP : A hybrid magnitude-phase approach to unsupervised segmentation of tumor areas in breast cancer histology images. *Journal of Pathology Informatics*, 4(2) :1, 2013. ISSN 2153-3539. doi : 10.4103/2153-3539.109802. URL http://www.jpathinformatics.org/text.asp?2013/4/2/1/109802. 86

Angel Cruz-Roa, Juan C. Caicedo, and Fabio A. González. Visual pattern mining in histology image collections using bag of features. *Artificial Intelligence in Medicine*, 52(2) :91–106, June 2011. ISSN 09333657. doi : 10.1016/j.artmed.2011.04.010. URL https://linkinghub.elsevier.com/retrieve/pii/S0933365711000510. 86

Barry Smith, Michael Ashburner, Cornelius Rosse, Jonathan Bard, William Bug, Werner Ceusters, Louis J. Goldberg, Karen Eilbeck, Amelia Ireland, Christopher J. Mungall, Neocles Leontis, Philippe Rocca-Serra, Alan Ruttenberg, Susanna-Assunta Sansone, Richard H. Scheuermann, Nigam Shah, Patricia L. Whetzel, and Suzanna Lewis. The OBO Foundry : coordinated evolution of ontologies to support biomedical data integration. *Nature Biotechnology*, 25(11) :1251–1255, November 2007. ISSN 1546-1696. doi : 10.1038/nbt1346. URL https://www.nature.com/articles/nbt1346. 86

V. Roullier, V-T. Ta, O. Lezoray, and A. Elmoataz. Graph-based multi-resolution segmentation of histological whole slide images. In *2010 IEEE International Symposium on Biomedical Imaging : From Nano to Macro*, pages 153–156, Rotterdam, April 2010. IEEE. ISBN 978-1-4244-4125-9. doi : 10.1109/ISBI.2010.5490390. URL http://ieeexplore.ieee.org/document/5490390/. 86

Leilani H. Gilpin, David Bau, Ben Z. Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. Explaining Explanations : An Approach to Evaluating Interpretability of Machine Learning. *ArXiv*, abs/1806.00069, 2018. 86

Cynthia Rudin. Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead. *arXiv :1811.10154 [cs, stat]*, September 2019. URL http://arxiv.org/abs/1811.10154. arXiv : 1811.10154. 86, 98

D.D. Nauck. Measuring interpretability in rule-based classification systems. In *The 12th IEEE International Conference on Fuzzy Systems, 2003. FUZZ '03.*, volume 1, pages 196–201 vol.1, May 2003. doi : 10.1109/FUZZ.2003.1209361. ISSN : null. 86

Anil K. Jain and Farshid Farrokhnia. Unsupervised texture segmentation using Gabor filters. In *1990 IEEE international conference on systems, man, and cybernetics conference proceedings*, pages 14–19. IEEE, 1990. 87

D.G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, pages 1150–1157 vol.2, Kerkyra, Greece, 1999a. IEEE. ISBN 978-0-7695-0164-2. doi : 10.1109/ICCV.1999.790410. URL http://ieeexplore.ieee.org/document/790410/. 87

Zellig S. Harris. Distributional Structure. *WORD*, 10(2-3) :146–162, August 1954. ISSN 0043-7956, 2373-5112. doi : 10.1080/00437956.1954.11659520. URL http://www.tandfonline.com/doi/full/10.1080/00437956.1954.11659520. 87

Andrew Janowczyk, Ajay Basavanhally, and Anant Madabhushi. Stain normalization using sparse autoencoders (stanosa) : application to digital pathology. *Computerized Medical Imaging and Graphics*, 57 :50–61, 2017. 88

Le Hou, Vu Nguyen, Ariel B Kanevsky, Dimitris Samaras, Tahsin M Kurc, Tianhao Zhao, Rajarsi R Gupta, Yi Gao, Wenjin Chen, David Foran, et al. Sparse autoencoder for unsupervised nucleus detection and representation in histopathology images. *Pattern recognition*, 86 :188–200, 2019. 88

Khalid Raza and Nripendra Kumar Singh. A tour of unsupervised deep learning for medical image analysis. *arXiv preprint arXiv :1812.07715*, 2018. 88

Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. *arXiv :1312.6114 [cs, stat]*, May 2014. URL http://arxiv.org/abs/1312.6114. arXiv : 1312.6114. 88

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic Backpropagation and Approximate Inference in Deep Generative Models. *arXiv :1401.4082 [cs, stat]*, May 2014. URL http://arxiv.org/abs/1401.4082. arXiv : 1401.4082. 88

Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546. IEEE, 2005. 88

Kilian Q. Weinberger and Lawrence K. Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10(Feb) :207–244, 2009. 88

Nicolai Wojke and Alex Bewley. Deep Cosine Metric Learning for Person Re-identification. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 748–756, Lake Tahoe, NV, March 2018. IEEE. ISBN 978-1-5386-4886-5. doi : 10.1109/WACV.2018.00087. URL https://ieeexplore.ieee.org/document/8354191/. 88, 92

Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human pose estimation with iterative error feedback. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4733–4742, 2016. 88

Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab : Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4) :834–848, 2017. 88

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn : Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 88

Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, and Cordelia Schmid. DeepFlow : Large displacement optical flow with deep matching. In *Proceedings of the IEEE international conference on computer vision*, pages 1385–1392, 2013. 88

Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders : Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016. 88

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep Clustering for Unsupervised Learning of Visual Features. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, volume 11218, pages 139–156. Springer International Publishing, Cham, 2018a. ISBN 978-3-030-01263-2 978-3-030-01264-9. doi : 10.1007/978-3-030-01264-9_9. URL http://link.springer.com/10.1007/978-3-030-01264-9_9. 88

Hehe Fan, Liang Zheng, Chenggang Yan, and Yi Yang. Unsupervised Person Re-identification : Clustering and Fine-tuning. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(4) :83 :1–83 :18, October 2018. ISSN 1551-6857. doi : 10.1145/3243316. URL https://doi.org/10.1145/3243316. 88, 92, 102

Babak Ehteshami Bejnordi, Geert Litjens, Meyke Hermsen, Nico Karssemeijer, and Jeroen AWM van der Laak. A multi-scale superpixel classification approach to the detection of regions of interest in whole slide histopathology images. In *Medical Imaging 2015 : Digital Pathology*, volume 9420, page 94200H. International Society for Optics and Photonics, 2015. 88, 89

Abdulkadir Albayrak and Gokhan Bilgin. Automatic cell segmentation in histopathological images via two-staged superpixel-based algorithms. *Medical & biological engineering & computing*, 57(3) :653–665, 2019. 88, 89

Shereen Fouad, David Randell, Antony Galton, Hisham Mehanna, and Gabriel Landini. Unsupervised superpixel-based segmentation of histopathological images with consensus clustering. In *Annual Conference on Medical Image Understanding and Analysis*, pages 767–779. Springer, 2017. 88, 89

Yan Xu, Jun-Yan Zhu, I Eric, Chao Chang, Maode Lai, and Zhuowen Tu. Weakly supervised histopathology cancer image segmentation and classification. *Medical image analysis*, 18 (3) :591–604, 2014. 89

Yan Xu, Zhipeng Jia, Liang-Bo Wang, Yuqing Ai, Fang Zhang, Maode Lai, I Eric, and Chao Chang. Large scale tissue histopathology image classification, segmentation, and visualization via deep convolutional activation features. *BMC bioinformatics*, 18(1) :1–17, 2017. 89

Kristine A Thomas, Matthew J Sottile, and Carolyn M Salafia. Unsupervised segmentation for inflammation detection in histopathology images. In *International Conference on Image and Signal Processing*, pages 541–549. Springer, 2010. 89

Adnan M Khan, Hesham El-Daly, Emma Simmons, and Nasir M Rajpoot. Hymap : A hybrid magnitude-phase approach to unsupervised segmentation of tumor areas in breast cancer histology images. *Journal of pathology informatics*, 4(Suppl), 2013. 89

Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11) :2274–2282, 2012. 89, 93

Harshita Sharma, Norman Zerbe, Sebastian Lohmann, Klaus Kayser, Olaf Hellwich, and Peter Hufnagl. A review of graph-based methods for image analysis in digital histopathology. *Diagnostic pathology*, 1(1), 2015. 89

C. Djeraba. Association and content-based retrieval. *IEEE Transactions on Knowledge and Data Engineering*, 15(1) :118–135, January 2003. ISSN 2326-3865. doi : 10.1109/TKDE.2003.1161586. 90

Jonathon S. Hare, Paul H. Lewis, Peter G. B. Enser, and Christine J. Sandom. Mind the gap : another look at the problem of the semantic gap in image retrieval. page 607309, San Jose, CA, January 2006. doi : 10.1117/12.647755. URL http://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/12.647755. 90

Hao Ma, Jianke Zhu, Michael Rung-Tsong Lyu, and Irwin King. Bridging the Semantic Gap Between Image Contents and Tags. *IEEE Transactions on Multimedia*, 12(5) :462–473, August 2010. ISSN 1520-9210, 1941-0077. doi : 10.1109/TMM.2010.2051360. URL http://ieeexplore.ieee.org/document/5473143/. 90

Pinar Duygulu, Kobus Barnard, Joao FG de Freitas, and David A Forsyth. Object recognition as machine translation : Learning a lexicon for a fixed image vocabulary. In *European conference on computer vision*, pages 97–112. Springer, 2002. 90

Geoffrey J Hay and Guillermo Castilla. Geographic object-based image analysis (geobia) : A new name for a new discipline. In *Object-based image analysis*, pages 75–89. Springer, 2008. 90

Thomas Blaschke, Geoffrey J Hay, Maggi Kelly, Stefan Lang, Peter Hofmann, Elisabeth Addink, Raul Queiroz Feitosa, Freek Van der Meer, Harald Van der Werff, Frieke Van Coillie, et al. Geographic object-based image analysis–towards a new paradigm. *ISPRS journal of photogrammetry and remote sensing*, 87 :180–191, 2014. 90

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 132–149, 2018b. 92, 102

Vaïa Machairas, Matthieu Faessel, David Cárdenas-Peña, Théodore Chabardes, Thomas Walter, and Etienne Decencière. Waterpixels. *IEEE Transactions on Image Processing*, 24 (11) :3707–3716, 2015. 93, 115, 120

Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient Graph-Based Image Segmentation. *International Journal of Computer Vision*, 59(2) :167–181, September 2004. ISSN 0920-5691. doi : 10.1023/B:VISI.0000022288.19776.77. URL http://link.springer.com/10.1023/B:VISI.0000022288.19776.77. 93, 94, 102

Marek Gagolewski, Maciej Bartoszuk, and Anna Cena. Genie : A new, fast, and outlier-resistant hierarchical clustering algorithm. *Information Sciences*, 363 :8–23, 2016. 96

Patrik Sabol, Peter Sinčák, Kana Ogawa, and Pitoyo Hartono. Explainable Classifier Supporting Decision-making for Breast Cancer Diagnosis from Histopathological Images. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, July 2019. doi : 10.1109/IJCNN.2019.8852070. ISSN : 2161-4393. 98

Alsallakh Bilal, Amin Jourabloo, Mao Ye, Xiaoming Liu, and Liu Ren. Do convolutional neural networks learn class hierarchy ? *IEEE transactions on visualization and computer graphics*, 24(1) :152–162, 2017. 101

François Chollet. Xception : Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017. 102

Abhronil Sengupta, Yuting Ye, Robert Wang, Chiao Liu, and Kaushik Roy. Going deeper in spiking neural networks : Vgg and residual architectures. *Frontiers in neuroscience*, 13 :95, 2019. 102

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet : A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 102

Diederik P Kingma and Jimmy Ba. Adam : A method for stochastic optimization. *arXiv preprint arXiv :1412.6980*, 2014. 102

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 103

Paul VC Hough. Method and means for recognizing complex patterns, December 18 1962. US Patent 3,069,654. 108

Richard O Duda, Peter E Hart, et al. *Pattern classification and scene analysis*. Wiley, New York, 1973. ISBN 978-0-471-22361-0. Open Library ID : OL5287711M. 108

Markus Weber, Max Welling, and Pietro Perona. Unsupervised learning of models for recognition. In *European conference on computer vision*, pages 18–32. Springer, 2000. 108

Robert Fergus, Pietro Perona, and Andrew Zisserman. Object class recognition by unsupervised scale-invariant learning. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 2, pages II–II. IEEE, 2003. 108

Torsten Rohlfing, Robert Brandt, Randolf Menzel, and Calvin R Maurer Jr. Evaluation of atlas selection strategies for atlas-based image segmentation with application to confocal microscopy images of bee brains. *NeuroImage*, 21(4) :1428–1442, 2004. 108

T Rohlfing, R Brandt, R Menzel, DB Russakoff, J Maurer, and R Calvin. The handbook of medical image analysis–volume iii : Registration models. *Vol. chapter 11. Kluwer Academic/Plenum Publishers ; Quo vadis, atlas-based segmentation*, pages 435–486, 2005. 108

David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999b. 108

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 109

Zhe Lin, Larry S Davis, David Doermann, and Daniel DeMenthon. Hierarchical part-template matching for human detection and segmentation. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE, 2007a. 109

P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9) :1627–1645, 2010. 109

Ning Zhang, Jeff Donahue, Ross Girshick, and Trevor Darrell. Part-based r-cnns for fine-grained category detection. In *European conference on computer vision*, pages 834–849. Springer, 2014. 109

Germain Forestier. *Connaissances et clustering collaboratif d'objets complexes multisources*. PhD thesis, Strasbourg, 2010. 109

Camille Kurtz. *Une approche collaborative segmentation-classification pour l'analyse descendante d'images multirésolutions*. PhD thesis, 2012. 109

Hartmut Ehrig, Annegret Habel, and Hans-JÃ¶rg Kreowski. Introduction to graph grammars with applications to semantic networks. *Computers and Mathematics with Applications*, 23 (6) :557 – 572, 1992. ISSN 0898-1221. doi : https://doi.org/10.1016/0898-1221(92)90124-Z. URL http://www.sciencedirect.com/science/article/pii/089812219290124Z. 109

Frank Fuchs. *Contribution à la reconstruction du bâti en milieu urbain, à l'aide d'images aériennes stéréoscopiques à grande échelle : étude d'une approche structurelle*. PhD thesis, 2001. 109

Marek R Ogiela and Ryszard Tadeusiewicz. Syntactic reasoning and pattern recognition for analysis of coronary artery images. *Artificial Intelligence in Medicine*, 26(1-2) :145–159, 2002. 109

Jaroslav Nešetřil and Patrice Ossona De Mendez. Grad and classes with bounded expansion ii. algorithmic aspects. *European Journal of Combinatorics*, 29(3) :777–791, 2008. 111

Arnaud Abreu, F-X Frenois, S Valitutti, P Brousset, P Denefle, Benoît Naegel, and Cédric Wemmert. Optimal cut in minimum spanning trees for 3-d cell nuclei segmentation. In *Proceedings of the 10th International Symposium on Image and Signal Processing and Analysis*, pages 195–199. IEEE, 2017. 113

Arnaud Abreu, FX Frenois, S Valitutti, P Brousset, P Denefle, Benoît Naegel, and Cédric Wemmert. Model-based graph segmentation in 2-d fluorescence microscopy images. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3844–3849. IEEE, 2018. 114

Chichen Fu, Soonam Lee, David Joon Ho, Shuo Han, Paul Salama, Kenneth W Dunn, and Edward J Delp. Three dimensional fluorescence microscopy image synthesis and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 2221–2229, 2018. 114

Faisal Mahmood, Daniel Borders, Richard Chen, Gregory N McKay, Kevan J Salimian, Alexander Baras, and Nicholas J Durr. Deep adversarial training for multi-organ nuclei segmentation in histopathology images. *IEEE transactions on medical imaging*, 2019. 114

Carlos Ortiz De Solórzano, E Garcia Rodriguez, Arthur Jones, Dan Pinkel, Joe W Gray, Damir Sudar, and Stephen J Lockett. Segmentation of confocal microscope images of cell nuclei in thick tissue sections. *Journal of Microscopy*, 193(3) :212–226, 1999. ISSN 1365-2818. 115

Monica K Chawla, Gang Lin, Kathy Olson, Almira Vazdarjanova, Sara N Burke, Bruce L McNaughton, Paul F Worley, John F Guzowski, Badrinath Roysam, and Carol A Barnes. 3D-catFISH : a system for automated quantitative three-dimensional compartmental analysis of temporal gene transcription activity imaged by fluorescence in situ hybridization. *Journal of neuroscience methods*, 139(1) :13–24, 2004. ISSN 0165-0270. 115

Gang Lin, Umesh Adiga, Kathy Olson, John F Guzowski, Carol A Barnes, and Badrinath Roysam. A hybrid 3D watershed algorithm incorporating gradient cues and object models for automatic segmentation of nuclei in confocal image stacks. *Cytometry Part A*, 56(1) : 23–36, 2003. ISSN 1552-4930. 115

Gang Lin, Monica K Chawla, Kathy Olson, John F Guzowski, Carol A Barnes, and Badrinath Roysam. Hierarchical, model-based merging of multiple fragments for improved three-dimensional segmentation of nuclei. *Cytometry Part A*, 63(1) :20–33, 2005. 115, 120, 121

PS Umesh Adiga and BB Chaudhuri. An efficient method based on watershed and rule-based merging for segmentation of 3-D histopathological images. *Pattern Recognition*, 34(7) : 1449–1458, 2001. 115, 120

Carolina Wählby, I-M Sintorn, Fredrik Erlandsson, Gunilla Borgefors, and Ewert Bengtsson. Combining intensity, edge and shape information for 2D and 3D segmentation of cell nuclei in tissue sections. *Journal of Microscopy*, 215(1) :67–76, 2004. 115

Gang Lin, Monica K Chawla, Kathy Olson, Carol A Barnes, John F Guzowski, Christopher Bjornsson, William Shain, and Badrinath Roysam. A multi-model approach to simultaneous segmentation and classification of heterogeneous populations of cell nuclei in 3D confocal microscope images. *Cytometry Part A*, 71(9) :724–736, 2007b. ISSN 1552-4930. 115

Vaïa Machairas. *Waterpixels and their application to image segmentation learning*. PhD thesis, Université de recherche Paris Sciences et Lettres, 2016. 115, 120

Lee Kamentsky, Thouis R Jones, Adam Fraser, Mark-Anthony Bray, David J Logan, Katherine L Madden, Vebjorn Ljosa, Curtis Rueden, Kevin W Eliceiri, and Anne E Carpenter. Improved structure, function and compatibility for cellprofiler : modular high-throughput image analysis software. *Bioinformatics*, 27(8) :1179–1180, 2011. 120

Johannes Schindelin, Ignacio Arganda-Carreras, Erwin Frise, Verena Kaynig, Mark Longair, Tobias Pietzsch, Stephan Preibisch, Curtis Rueden, Stephan Saalfeld, Benjamin Schmid, et al. Fiji : an open-source platform for biological-image analysis. *Nature methods*, 9(7) : 676–682, 2012. 120

Florian Mueller, Adrien Senecal, Katjana Tantale, Hervé Marie-Nelly, Nathalie Ly, Olivier Collin, Eugenia Basyuk, Edouard Bertrand, Xavier Darzacq, and Christophe Zimmer. Fish-quant : automatic counting of transcripts in 3D fish images. *Nature methods*, 10(4) :277–278, 2013. 120

Conclusion

Bilan

Dans ces travaux de thèse, de multiples aspects de l'analyse automatique des images ont été abordés dans le but d'extraire des informations cliniquement pertinentes dans les images de la pathologie numérique. La première partie présentait les différentes tâches, segmentation, description, classement, qui se rapportent à l'analyse des images. Elle énonçait ces tâches sous la forme de problèmes informatiques et présentait les solutions algorithmiques classiquement mises en œuvre pour les résoudre. En attachant un intérêt particulier aux raisonnements qui motivent la conception et guident l'exécution de ces algorithmes, nous avons remarqué que toute solution complexe d'analyse synthétise un mécanisme de perception. Les étapes de cette analysent constituent une chaîne de traitements, ou *chaîne perceptuelle* dont les maillons sont construits de manière systématique selon des règles édictées par le cadre général de la méthodologie des *sacs de mots*.

Toujours dans les limites de ce cadre, la deuxième partie de ce travail a commencé par décrire les réseaux neuronaux convolutifs profonds, fleuron de l'apprentissage automatique pour l'analyse des images, comme un cas particulier de la méthodologie des *sacs de mots*. Sans toutefois omettre les attributs de ces modèles qui font leur supériorité face aux autres solutions d'analyse, l'inscription dans un formalisme plus général a également permis de fournir des règles élémentaires de production de ces modèles pour la classification des images. Le reste de ce chapitre s'est ensuite attardé sur deux cas particuliers d'application de ces réseaux neuronaux à la résolution de problèmes biomédicaux concrets. Soucieux de se placer dans des conditions cliniques « réalistes », ces exemples nous ont poussé à raffiner ces outils, à construire des réflexions plus générales sur l'*assemblage* de classifieurs et la *certitude* des modèles prédictifs.

Certaines déconvenues de cette deuxième partie ont mis en lumière les limites des réseaux neuronaux convolutifs, et plus généralement de la classification supervisée, pour l'analyse pragmatique des images de la pathologie numérique. Plusieurs obstacles s'opposent à l'expansion de ces modèles dans les applications biomédicales : la généralisation, le passage à l'échelle (en termes de couverture sémantique du savoir humain et de pénibilité d'annotation), mais aussi le raisonnement logique pour une interaction constructive entre le système d'analyse et l'utilisateur. Dans la troisième partie, nous avons préconisé l'apprentissage de métriques par des modèles génératifs pour augmenter le pouvoir de généralisation du système. Nous avons ensuite affirmé qu'un aperçu de cette métrique, présenté par un jeu fini de concepts acquis sans supervision par la machine, et reliés entre eux par similarité pour former un arbre de subsomption, facilite et raccourci le temps d'annotation des données. Enfin, une fois les concepts élémentaires assimilés, nous avons introduit une méthode de segmentation d'images qui déduit la présence de structures complexes à partir de leurs segments constitutifs élémentaires et garantit ainsi l'interprétabilité totale dans la détection d'objets dans les images. Cette approche a notamment été employé pour segmenter des noyaux de cellules dans des images de microscopie de fluorescence.

Contributions et développements

Nous avons mis au point un outil d'analyse des marqueurs immunohistochimiques complexes. À la différence des approches traditionnelles du domaine, la solution développée considère une tâche de classification plutôt qu'une problématique de quantification du signal du marqueur. Sa performance repose sur une stratégie originale d'entraînement des ensembles de réseaux neuronaux. Nous avons proposé une description détaillée de la procédure Abreu et al. [2019] et avons mis à disposition le code source [7] qui a permis de réaliser les expériences communiquées. Le classifieur a été testé sur une tâche de dénombrement de vaisseaux *HEVs* dont le marquage par le *MECA-79* est visuellement difficile à caractériser.

Nous avons développé une procédure de diagnostic automatique de lames en coloration standard *H&E*. Le problème est énoncé comme une tâche de classification locale de *patches* suivi d'un problème de compilation des prédictions locales vers une décision diagnostic sur le patient Syrykh et al. [2020]. La méthodologie a été appliquée au diagnostic différentiel entre l'hyperplasie folliculaire et le lymphome folliculaire. Le code source [8] est une fois encore mis à disposition sur un dépôt publique. Contrairement à beaucoup de travaux du domaine, cette étude intègre un cadre décisionnel bayésien. Ce caractère transparait aussi bien dans la prédiction du réseau de neurones, construit pour exprimer l'incertitude quant à sa prise de décision, que dans la stratégie de diagnostic qui, sous certaines conditions, peut fournir de solides garanties sur la fiabilité de la prédiction. Bien qu'elle ne soit pas purement technique sur le plan informatique, une autre contribution importante de cette étude est la mise en garde contre certaines difficultés majeures de déploiement des solutions d'apprentissage profond vers des applications biomédicales concrètes.

Nous avons également présenté deux études sur la segmentation par arbres couvrants Abreu et al. [2017, 2018]. Ces algorithmes sont implémentés pour segmenter des noyaux de cellules dans des images de microscopie de fluorescence l'un d'eux a fait l'objet d'un dépôt sur la plateforme GitLab privée de l'université de Strasbourg. Loin de l'état de l'art de la segmentation des noyaux, ces algorithmes posent néanmoins les bases d'une procédure de segmentation entièrement interprétable qui, appliquée à des structures plus larges et sémantiquement plus complexes rendra les système d'analyse capables de se justifier et d'échanger de manière constructive avec l'expert.

Perspectives

Les travaux réalisés au cours de cette thèse ouvrent de nombreuses pistes de recherche. La méthodologie d'assemblage de réseaux neuronaux proposée tout d'abord, a été pensée pour l'amélioration d'un classifieur. Elle pourrait par exemple être complétée par l'analyse *a posteriori* des poids et chemins d'activation dans les modèles obtenus. Le résultat pourrait permettre de constituer des représentations d'images *désentrelacées*, c'est-à-dire dont les composantes ont les activations les plus décorrélées possibles. Cela serait un moyen de construire des classifieurs à redondance d'information minimale, ce qui aboutirait à des modèles économes en paramètres et beaucoup plus généraux.

Le potentiel des méthodes d'évaluation de la certitude des modèles prédictifs, comme celle que nous avons utilisée pour le diagnostic du lymphome, devrait encore être évalué pour des applications de déploiement. Lorsque des lacunes sont repérées de cette façon dans les données d'apprentissage, une stratégie de raffinement d'un réseau en vue de sa généralisations aux

7. https://github.com/ArnaudAbreu/neuralyzer
8. https://github.com/ArnaudAbreu/DiagFLFH

nouvelles données peut être envisagée. En effet, le regroupement automatique des données les moins certaines constitue automatiquement un nouvel ensemble auquel les poids du modèle sont adaptés, on parle de *fine tuning* dans ce cas précis.

Toujours sur ce cas particulier, une stratégie multi-experts fonctionnerait également : plutôt que d'utiliser une stratégie d'adaptation, un second modèle est entraîné selon une procédure d'assemblage semblable à celle que nous avons mise au point dans ce travail. D'ailleurs, la mesure d'incertitude développée par Gal and Ghahramani [2016] à laquelle nous avons eu recours, est notamment interprétée comme la *diversité* ou *ambiguïté* (voir la Sous-section 1.4.6) d'un ensemble de classifieurs. Il serait alors intéressant d'étudier comment notre technique d'augmentation de la diversité d'un ensemble pourrait fiabiliser ou accélérer la quantification de l'incertitude.

Enfin, la procédure de clustering décrite dans la Sous-section 3.3.2 et dans la Sous-section 3.3.3 n'ont pas exploré le principe de régularisation inter-patients. En effet, l'un des principes du début de ces sections stipulait que l'existence d'un concept relevé chez un patient doit être confirmée par son observation sur un grand nombre d'autres patients. La méthode pourrait donc bénéficier d'un recalage par similarité des structures entre les patients au moment de la construction du vocabulaire. Il serait également pertinent d'envisager dans ce cas d'inclure les différentes régularisations (regroupement en segments, regroupements des segments de même nature entre les patients) directement dans la démarche *PUL* utilisée pour adapter la perception du système, c'est-à-dire la représentation des images par le réseau de neurones descripteur.

Références

Arnaud Abreu, Camille Franchet, FX Frenois, P Brousset, JP Girard P Denéfle, P Denéfle, Benoît Naegel, and Cédric Wemmert. Ensemble of neural networks for high endothelial venules detection in meca-79 immunohistochemistry images. In *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pages 938–942. IEEE, 2019. 132

Charlotte Syrykh, Arnaud Abreu, Nadia Amara, Aurore Siegfried, Véronique Maisongrosse, François X Frenois, Laurent Martin, Cédric Rossi, Camille Laurent, and Pierre Brousset. Accurate diagnosis of lymphoma on whole-slide histopathology images using deep learning. *npj Digital Medicine*, 3(1) :1–8, 2020. 132

Arnaud Abreu, F-X Frenois, S Valitutti, P Brousset, P Denefle, Benoît Naegel, and Cédric Wemmert. Optimal cut in minimum spanning trees for 3-d cell nuclei segmentation. In *Proceedings of the 10th International Symposium on Image and Signal Processing and Analysis*, pages 195–199. IEEE, 2017. 132

Arnaud Abreu, FX Frenois, S Valitutti, P Brousset, P Denefle, Benoît Naegel, and Cédric Wemmert. Model-based graph segmentation in 2-d fluorescence microscopy images. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3844–3849. IEEE, 2018. 132

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation : Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059, 2016. 133

Résumé

Ce travail de thèse a pour objectif de développer des méthodes d'analyse automatique des images de pathologie numérique. Il évalue tout particulièrement le potentiel des réseaux neuronaux convolutifs profonds et des représentations hiérarchiques dans la reconnaissance d'objets ou de concepts biologiques d'intérêt dans ces images. Les réseaux neuronaux convolutifs profonds (Convolutional Neural Networks, CNNs) se sont attaqués, avec grand succès, à la plupart des problèmes de reconnaissance d'objets dans les images. Bien que les images de microscopie soient, depuis presque 60 ans, un champ d'application important pour les algorithmes de traitement d'image et de vision par ordinateur, ce sont les avancées récentes dans le domaine de la numérisation des lames histologiques (Whole Slide Imaging) qui permettent aujourd'hui d'envisager l'intégration de ces algorithmes dans de véritables applications biomédicales. Les plateformes de pathologie numérique peuvent désormais scanner les lames à des débits et des résolutions compatibles avec l'activité diagnostique des services hospitaliers. Comme une suite logique à la démarche de numérisation, des solutions d'analyse automatiques, notamment basées sur des CNNs, sont naturellement développées avec l'ambition de réduire les erreurs de diagnostic, de pronostic ou de prédiction de réponse à la thérapie.

L'adaptation des CNNs aux applications de la pathologie numérique pose cependant de nombreux défis. Tout d'abord, la nature critique de la décision médicale impose des contraintes fortes sur les performances du système, mais aussi sur son ergonomie et la transparence de son intégration dans la pratique quotidienne des laboratoires de pathologie. Ensuite, et de manière plus technique, les images de lames entières numérisées occupent un espace mémoire conséquent et, au même titre que les images spatiales de télédétection, doivent être analysées localement avant qu'une décision à l'échelle du patient puisse être compilée. Enfin, et de manière toujours très similaire aux applications de télédétection, les systèmes d'analyse d'images de pathologie doivent reconnaître un grand nombre d'entités biologiques et connaître leurs relations vis-à-vis des pathologies étudiées afin de proposer des résultats interprétables aux experts pathologistes.

Dans le premier chapitre du mémoire, les mécanismes à l'œuvre dans l'analyse automatique des images numériques sont abordés. Pour chacun d'eux, nous expliquons les cheminements qui traduisent les concepts d'analyse en problèmes d'optimisation, ainsi que les principales pistes explorées pour leur résolution algorithmique. L'accent est particulièrement mis sur les relations d'ordre établies entre les résultats fournis par ces méthodes et les structures hiérarchiques qui en découlent. Lorsque cela est possible, nous mettons notamment en parallèle ces structures avec celles des connaissances humaines, mais aussi avec les raisonnements qui conditionnent une décision à l'observation d'une image. Ce chapitre aboutit à un regroupement général du processus d'analyse sous la forme d'une chaîne perceptuelle. La principale faiblesse des approches d'analyse traditionnelles apparait comme un manque d'interaction entre les processus de construction des différents maillons de cette chaîne, notamment entre la partie descriptive et la partie dédiée au classement, à laquelle remédie l'approche connexionniste à l'origine de l'apprentissage profond.

Le second chapitre expose le fonctionnement des CNNs à la lumière des définitions précédentes en soulignant les aspects qui en font une chaîne perceptuelle plus performante que les versions évoquées dans le premier chapitre. Les spécificités des images de la pathologie numérique, ainsi que les différentes contraintes de l'environnement biomédical relatives à l'implantation de systèmes automatiques d'analyse d'images sont ensuite présentées. Enfin, cette partie présente en détail comment des applications biomédicales peuvent être envisagées et implémentées dans le respect de ces contraintes. Cet aspect est illustré au travers de deux applications développées durant cette thèse, l'une pour l'analyse de marquage immunohistochimiques complexes, l'autre pour l'assistance dans le diagnostic des lymphomes, chacune ayant fait l'objet d'une communication dans un congrès international ou dans une revue à comité de lecture.

Le dernier chapitre s'attache à relever les limites de l'apprentissage profond, en particulier sous la forme de modèles de classification, pour la résolution des problématiques biomédicales. Ces observations critiques conduisent à des conceptions et développements plus généraux dans lesquels la place de l'apprentissage profond est repensée comme une brique de soutien à des outils de fouille de données. Les algorithmes de fouille sont des outils propices à la structuration (entendons ici hiérarchisation) et à la classification exhaustive des données, qui sont autant de propriétés partagées avec la connaissance humaine. Au travers de la construction d'arbres de segmentations ou de subsomptions, nous montrons comment la fouille de données, soutenue par des représentations construites par apprentissage profond, peuvent constituer des éléments d'analyse compatibles avec l'interprétation humaine et économes en termes d'annotations et d'apprentissage.

Arnaud ABREU

Réseaux neuronaux convolutifs profonds et représentations hiérarchiques : applications et perspectives pour la pathologie numérique

Résumé

Les réseaux neuronaux convolutifs profonds excellent à résoudre les problèmes de reconnaissance dans les images. Les avancées récentes dans le domaine de la numérisation des lames histologiques permettent aujourd'hui d'utiliser ces algorithmes dans de véritables applications biomédicales en microscopie. Des solutions d'analyse automatiques sont donc naturellement développées pour réduire les erreurs de diagnostic. Nous présentons deux applications, l'une pour l'analyse de marquages immunohistochimiques, l'autre pour assister le diagnostic des lymphomes. Nous présentons enfin les limites de l'apprentissage profond pour résoudre les problématiques biomédicales. Cette critique conduit à repenser l'apprentissage profond comme un soutien aux outils de fouille de données. Par le biais d'arbres de segmentations ou de subsomptions, ces techniques, soutenues par l'apprentissage profond, sont compatibles avec l'interprétation humaine, économes en annotation et en apprentissage.

Résumé en anglais

Deep convolutional neural networks excel at solving recognition problems in images. Recent advances in the field of digitisation of histological slides now make it possible to use these algorithms in real biomedical applications in microscopy. Automatic analysis solutions are therefore naturally developed to reduce diagnostic errors. We present two applications, one for the analysis of immunohistochemical markers, the other to assist in the diagnosis of lymphomas. Finally, we present the limits of deep learning to solve biomedical problems. This critique leads us to rethink deep learning as a support for data mining tools. By means of segmentation trees or subsumptions, these techniques, supported by deep learning, are compatible with human interpretation, economical in annotation and learning.