HAL
open science

Extraction de caractéristiques et apprentissage statistique pour l'imagerie biomédicale cellulaire et tissulaire

Alexis Zubiolo

To cite this version:

Alexis Zubiolo. Extraction de caractéristiques et apprentissage statistique pour l'imagerie biomédicale cellulaire et tissulaire. Autre. Université Nice Sophia Antipolis, 2015. Français. NNT : 2015NICE4117 . tel-01290131

HAL Id: tel-01290131

https://theses.hal.science/tel-01290131v1

Submitted on 6 Jul 2017

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

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

Université Nice Sophia Antipolis
École doctorale STIC
Sciences et Technologies de l'Information et de la Communication

Thèse de doctorat

pour obtenir le grade de

Docteur en sciences

Spécialité : Automatique, traitement du signal et de l'image

présentée et soutenue par

Alexis Zubiolo

EXTRACTION DE CARACTERISTIQUES ET APPRENTISSAGE STATISTIQUE POUR L'IMAGERIE BIOMEDICALE TISSULAIRE ET CELLULAIRE

Thèse préparée au sein de l'équipe Morpheme
(Inria Sophia Antipolis, Laboratoire I3S (UMR CNRS 7271), iBV (UMR CNRS 7277))

défendue le vendredi 11 décembre 2015, devant le jury composé de :

Philippe Pognonec	CNRS	(Président)
Jenny Benois-Pineau	Université de Bordeaux	(Rapporteur)
Jean-Denis Durou	Université Paul Sabatier – Toulouse III	(Rapporteur)
Xavier Descombes	Inria Sophia Antipolis	(Examinateur)
Thomas Walter	MINES ParisTech	(Examinateur)
Éric Debreuve	CNRS	(Encadrant)
Damien Ambrosetti	CHU de Nice	(Invité)

REMERCIEMENTS

Je tiens à remercier en premier lieu mon directeur, Éric Debreuve, pour m'avoir donné l'opportunité de découvrir le monde académique. Tout au long de mon stage et de ma thèse, j'ai eu l'occasion de me pencher sur de nombreux domaines d'application du traitement de l'image.

Mes sincères remerciements vont également aux membres de mon jury de thèse, Jenny Benois-Pineau, Jean-Denis Durou, Thomas Walter, Philippe Pognonec, Damien Ambrosetti et Xavier Descombes, pour avoir pris le temps d'évaluer mes travaux, m'avoir fait part de leurs remarques et avoir fait le déplacement pour ma soutenance.

J'ai eu l'occasion de collaborer avec de nombreuses personnes venant de divers horizons pendant ces années passées au laboratoire. Je tiens à faire part de ma gratitude à Damien Ambrosetti, Philippe Pognonec, Thierry Pourcher, Kawssar HARB, Michèle Studer, Christian Alfano, Barbara André, Xavier Descombes et Grégoire Malandain pour les échanges que nous avons pu avoir. Ils ont su faire preuve d'une grande patience lorsqu'il s'agissait de m'expliquer leurs domaines de recherche et je leur en suis reconnaissant.

Merci aussi aux amis que je me suis faits pendant ces années de thèse. Il y a bien entendu les membres de l'équipe Morpheme, mais aussi des personnes que j'ai eu la chance de connaître à Inria et au laboratoire I3S. Je pense notamment à Éric, Laure, Grégoire, Xavier, Alejandro, Agustina, Yuliya, Olivier, Rita, Froso, Arnaud, Yasmine, Marine, Lola, Sylvain, Saïma, Aurélie, Vladimir (les deux), Nektaria, Yaqun, Sen, Wei, Emmanuelle, Huei-Fang, Matias, Alvaro, Pablo, Juan, Julien, Mohammed, Manon, Nicolas et Adrien. Je n'oublie pas les assistants de l'équipe pour leur aide : Christine, Micheline, Jane, Nadia et Frédéric. Bien entendu, une mention spéciale revient à mes collègues de bureau Emmanuel et Gaël avec qui j'ai pu avoir des discussion sur des sujets si variés que leur énumération augmenterait de façon considérable (et inutile) la longueur de ce manuscrit. Je pense bien entendu à toutes les personnes que j'ai pu rencontrer lors de mon aventure niçoise. J'ai passé de belles années dans cette belle région et je le leur dois en grande partie.

Enfin, je remercie bien évidemment ma famille et mes amis (d'Auvergne et d'ailleurs) pour leur soutien.

TABLE DES MATIÈRES

1 INTRODUCTION GÉNÉRALE 1
1.1 Le traitement d'images dans le domaine biomédical 1
1.1.1 La classification 2
1.1.2 L'extraction de caractéristiques 2
1.2 Organisation du manuscrit 3
1.2.1 Extension d'un classifieur binaire au cas multiclasse 3
1.2.2 Analyse morphologique de neurones 3
1.2.3 Étude du réseau vasculaire de carcinomes rénaux 3

I Classification supervisée 5

2 GÉNÉRALISATION D'UN CLASSIFIEUR BINAIRE 7
2.1 Introduction 7
2.1.1 Classification supervisée d'objets 7
2.1.2 Classification binaire et classification multiclasse 9
2.1.3 Passage du cas binaire du cas multiclasse 9
2.1.4 Le cas des Margin Trees 11
2.1.5 Un schéma récursif d'extension général 13
2.2 Une extension fondée sur la coupure de graphe 14
2.2.1 Motivations et défis 14
2.2.2 Approche proposée : Graph Cut SVM (GCSVM) 16
2.2.3 Équilibrage de l'arbre binaire 18
2.2.4 Complexité 19
2.3 Résultats expérimentaux 20
2.3.1 Choix d'implémentation 20
2.3.2 Résultats sur des données 2D synthétiques 20
2.3.3 Résultats en reconnaissance de lettres 23
2.3.4 Résultats sur un ensemble de vidéos d'endomicroscopie 23
2.4 Conclusions et Perspectives 26

II Classification non supervisée 29

3 ANALYSE MORPHOLOGIQUE DE NEURONES DE SOURIS 31
3.1 Introduction 31
3.1.1 Contexte biologique : le néocortex mammifère 31
3.1.2 Présentation des données 33
3.1.3 Défis et enjeux 34

3.2 Étude des images x40 36
3.2.1 Segmentation des neurones 36
3.2.2 Calcul des descripteurs 36
3.3 Étude des images x10 40
3.3.1 Détection des neurones 40
3.3.2 Reconstruction des neurones 41
3.4 Mise en correspondance des neurones 43
3.4.1 Définition du problème 43
3.4.2 Motivation 44
3.4.3 Formalisation du problème : appariement d'ensembles de points de l'espace 44
3.4.4 Résolution des problèmes d'optimisation linéaire 46
3.4.5 Cas d'ambiguïté 47
3.4.6 Appariement de données synthétiques 47
3.4.7 Application à la correspondance de neurones 49
3.5 Interprétation des résultats 49
3.5.1 Histogrammes des distributions 49
3.5.2 Classification par k-moyennes 53
3.6 Conclusion et perspectives 53

4 ANALYSE DE L'ARCHITECTURE D'UN CARCINOME RÉNAL 55
4.1 Introduction 55
4.1.1 Contexte biomédical : le carcinome à cellules rénales 55
4.1.2 Présentation des données 56
4.1.3 Motivation 57
4.1.4 Défis et enjeux 59
4.2 Images à grossissement minimal 59
4.2.1 Segmentation du tissu tumoral 60
4.2.2 Résultats de segmentation 62
4.2.3 Découpage de la région d'intérêt 63
4.3 Analyse de l'architecture 65
4.3.1 Pré-traitements 65
4.3.2 Extraction du réseau vasculaire 71
4.3.3 Graphe représentant le réseau vasculaire 75
4.3.4 Combinaison d'information : Fusion de graphes voisins 77
4.4 Analyse du graphe 79
4.4.1 Extraction des données 79
4.4.2 Interprétations biomédicales des données 79
4.5 Conclusion 80
4.5.1 Chaîne de traitement proposée 80
4.5.2 Temps de calcul 80
4.5.3 Perspectives 82

5 CONCLUSION GÉNÉRALE 83
5.1 Revue des contributions 83

5.2 Perspectives 84
5.2.1 Extension d'un classifieur binaire 84
5.2.2 Analyse de la morphologie de neurones de souris 84
5.2.3 Analyse de l'architecture d'un carcinome rénal 85

III Annexes 87

A SÉPARATEURS À VASTE MARGE 89
A.1 Classifieur à marge optimale (SVM) 89
A.2 Dualité lagrangienne 92
A.2.1 Du problème primal au problème dual 92
A.2.2 Le cas des SVM 94
A.2.3 Classification non-linéaire 95
A.3 Cas non-séparable : la marge souple 96

B L'ALGORITHME DE COUPE MINIMALE 99
B.1 Notations et définitions 99
B.2 The Algorithm 99
B.3 Équilibrage de la coupe 100
B.4 Temps de calcul 101

BIBLIOGRAPHIE 103

1 INTRODUCTION GÉNÉRALE

Sommaire

1.1 Le traitement d'images dans le domaine biomédical 1

1.1.1 La classification 2

1.1.2 L'extraction de caractéristiques 2

1.2 Organisation du manuscrit 3

1.2.1 Extension d'un classifieur binaire au cas multiclasse 3

1.2.2 Analyse morphologique de neurones 3

1.2.3 Étude du réseau vasculaire de carcinomes rénaux 3

1.1 LE TRAITEMENT D'IMAGES DANS LE DOMAINE BIOMÉDICAL

Le traitement d'images biomédicales [Abràmoff et al., 2004 ; Bankman, 2008 ; Demirkaya et al., 2008] est une branche du traitement de l'image de plus en plus étudiée et que l'on retrouve dans de nombreux domaines applicatifs (microscopie, imagerie par résonance magnétique, ...). Son objectif principal est d'accompagner les biologistes et les médecins dans le traitement des données auxquelles ils sont confrontés au quotidien. Par exemple, certaines techniques permettent d'améliorer la qualité des images observées (débruitage [Delpretti et al., 2008 ; Luisier et al., 2009], défloutage [Choi et al., 2006 ; Tristán-Vega et al., 2012], ...) ou de réaliser certaines tâches (semi-) automatiquement et à grande échelle (comme détecter et compter des cellules [Kothari et al., 2009 ; Poulain et al., 2015]).

Le traitement de telles images présente des spécificités que l'on ne retrouve pas systématiquement dans le domaine du traitement de l'image. Par exemple, il est généralement plus simple de connaître les ordres de grandeurs de ce que l'on observe dans la mesure où on connaît le matériel qui a fait l'acquisition de l'image ainsi que l'échantillon observé.

Dans ce travail de thèse, deux principaux aspects du traitement d'images biomédicales seront étudiés : la classification et l'extraction de caractéristiques.

1.1.1 La classification

La classification est une étape que l'on retrouve de façon récurrente dans le domaine biomédical. En effet, les biologistes et médecins sont régulièrement amenés à établir un diagnostic, ce qui revient généralement attribuer une classe générale y au phénomène x observé (pathologie, cellule, ...). En apprentissage statistique, on distingue deux grandes familles d'algorithmes de classification :

— La *classification supervisée* (qui sera discutée plus en détails en Partie I) consiste à apprendre une règle de classification à partir de données (telles que des images) préalablement annotées, c'est-à-dire auxquelles un expert[1] a attribué une classe parmi un ensemble (prédéfini) de classes. Un algorithme de classification supervisée prendra donc en entrée un ensemble d'objets $\mathcal{X}$ et l'ensemble $\mathcal{Y}$ des classes correspondantes.

— La *classification non-supervisée* (ou *clustering*, Cf. Partie II) [Hastie, Tibshirani et Friedman, 2009] cherche à réaliser la même tâche, mais en n'ayant recours qu'à des données non-annotées, c'est-à-dire sans connaissance *a priori* sur l'ensemble des classes. En d'autres termes, un algorithme de classification non-supervisée n'aura que l'ensemble d'objets $\mathcal{X}$ en entrée.

Ces deux types de classification ont leur place dans le domaine biomédical. En effet, la classification supervisée permet à l'algorithme d'apprendre une règle de classification fondée sur des classes bien définies. À l'inverse, l'apprentissage non-supervisé peut quant à lui s'utiliser lorsque les classes ne sont pas clairement définies, qu'on cherche à les définir, ou lorsque l'acquisition d'une vérité terrain sur un nombre significatif d'échantillons n'est pas envisageable.

1.1.2 L'extraction de caractéristiques

L'extraction de caractéristiques est une composante du traitement de l'image qui va souvent de paire avec la classification. En effet, pour établir une règle de classification (supervisée ou non), on se base généralement sur un ensemble de critères numériques décrivant l'objet ou le phénomène observé $x \in \mathcal{X}$. En pratique et selon le contexte, deux types de caractéristiques peuvent être extraites :

— Des descripteurs génériques n'ayant pas nécessairement d'interprétation physique ou biologique (SIFT [Lowe, 2004], SURF [Bay et al., 2008], histogrammes de gradients orientés [Dalal et Triggs, 2005], contexte de forme [Belongie et al., 2002], sacs de mots [Sivic et Zisserman, 2003], ...),

1. Ici, le mot *expert* fait référence à une personne suffisamment compétente pour attribuer une classe à une donnée. Ce terme, très général, peut aussi bien désigner un biologiste lorsqu'il s'agit de donner un type de cellule particulier qu'une personne sans compétences techniques spécifiques pour reconnaître un chiffre ou une lettre dans une image.

— Des descripteurs ayant un sens physique. Typiquement, dans le domaine biomédical, il peut s'agir de caractéristiques morphologiques décrivant les objets observés (taille d'une cellule, épaisseur d'un vaisseau, ...).

1.2 ORGANISATION DU MANUSCRIT

Le présent manuscrit se décompose en trois principales parties correspondant à chacune des trois branches de l'imagerie biomédicale abordées : l'endomicroscopie, la microscopie confocale en fluorescence et l'histologie digitale.

1.2.1 Extension d'un classifieur binaire au cas multiclasse

Le Chapitre 2 s'intéresse à la classification supervisée multiclasse (c'est-à-dire dans le cas à $n \geqslant 3$ classes), et plus spécifiquement à l'extension d'un classifieur binaire au cas multiclasse. Un schéma général d'extension fondée sur des outils de partitionnement de graphe permettant de construire un arbre de classifieurs est proposé. Il est ensuite appliqué aux séparateurs à vaste marge (SVM, Cf. Annexe A) et à la coupe minimale de graphe (Cf. Annexe B) et testé sur un ensemble de vidéos endomicroscopiques sur lesquelles il faut prédire la classe pathologique des polypes.

1.2.2 Analyse morphologique de neurones

Le Chapitre 3 fait l'objet de l'analyse morphologique des neurones présents dans la couche V du cortex de la souris. Le cortex est observé par microscopie confocale en fluorescence et à deux niveaux de grossissement afin de disposer d'une information riche permettant de caractériser à la fois le corps cellulaire du neurone et sa dendrite apicale. Sur ces deux types d'images tridimensionnelles, un ensemble de descripteurs morphologiques est extrait de manière automatique pour chaque neurone. Une étape de mise en correspondance entre neurones aux deux grossissements est ensuite proposée afin de combiner les informations locales extraites en une information extraites sur le corps cellulaire et la dendrite apicale.

1.2.3 Étude du réseau vasculaire de carcinomes rénaux

Dans le Chapitre 4, nous nous intéresserons à l'étude des carcinomes à cellules rénales. Nous disposons d'un ensemble de lames numérisées sur lesquelles il est possible d'observer l'architecture du tissu tumoral à fort grossissement. Nous proposons d'analyser le ré-

seau vasculaire, critère important lors de l'établissement du diagnostic du patient, en le représentant par un graphe duquel un ensemble de caractéristiques seront extraites et mises en corrélation avec la classe histologique de la tumeur observée.

Première partie

Classification supervisée

2 GÉNÉRALISATION D'UN CLASSIFIEUR BINAIRE

Sommaire

2.1 Introduction 7
- 2.1.1 Classification supervisée d'objets 7
- 2.1.2 Classification binaire et classification multiclasse 9
- 2.1.3 Passage du cas binaire du cas multiclasse 9
- 2.1.4 Le cas des Margin Trees 11
- 2.1.5 Un schéma récursif d'extension général 13

2.2 Une extension fondée sur la coupure de graphe **14**
- 2.2.1 Motivations et défis 14
- 2.2.2 Approche proposée : Graph Cut SVM (GCSVM) 16
- 2.2.3 Équilibrage de l'arbre binaire 18
- 2.2.4 Complexité 19

2.3 Résultats expérimentaux **20**
- 2.3.1 Choix d'implémentation 20
- 2.3.2 Résultats sur des données 2D synthétiques 20
- 2.3.3 Résultats en reconnaissance de lettres 23
- 2.3.4 Résultats sur un ensemble de vidéos d'endomicroscopie 23

2.4 Conclusions et Perspectives **26**

2.1 INTRODUCTION

2.1.1 Classification supervisée d'objets

Le problème de la *classification* d'objets consiste à établir une procédure qui associe une classe (appartenant à un ensemble de classes) à une donnée. La *classification supervisée*[1] se fait à partir d'un *ensemble d'apprentissage*, composée d'objets et des classes qui leur correspondent, et d'un *ensemble de test*, composée d'objets dont la classe est inconnue. Une classification supervisée se réalise alors en deux temps. Elle commence par la *phase d'apprentissage*, pendant laquelle la procédure (généralement appelée *classifieur*) permettant d'associer un objet à une classe est définie. S'en suit alors la *phase de classification* (ou

1. La classification supervisée est à opposer à la *classification non supervisée* (ou *clustering*), pour laquelle on ne dispose que d'un seul ensemble d'objets non annotés (c'est-à-dire sans classe attribuée).

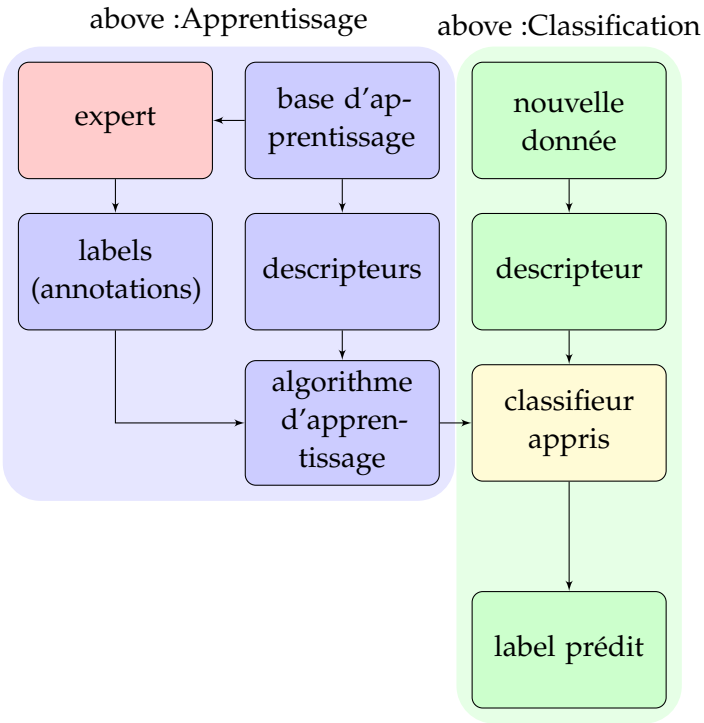


FIGURE 1 – Schéma général de l'apprentissage supervisé.

de *prédiction*) pendant laquelle les règles de classification établies (ou *apprises*) lors de la phase précédente sont utilisées pour en déduire la classe, *a priori* inconnue, d'un objet.

Pour ce faire, une donnée quelconque (que ce soit un document texte, un fichier audio, une image, une vidéo, un maillage, ...) est généralement convertie en un vecteur de valeurs réelles, pouvant appartenir à un espace de grande dimension, parfois appelé *signature* [OLIVA et TORRALBA, 2001 ; SIVIC et ZISSERMAN, 2003].

Dans le cas dit *offline*, l'apprentissage supervisé est fondé sur un ensemble d'apprentissage et un *algorithme d'apprentissage*. L'ensemble d'apprentissage est constitué de signatures associées à leur classe respective par un expert, et peut être écrit sous la forme :

$$\mathcal{S} = \left\{ (x^{(i)}, y^{(i)}) \mid x^{(i)} \in \mathbb{R}^d, y^{(i)} \in \{1, \ldots, n\}, 1 \leqslant i \leqslant m \right\}$$

où m est le nombre de signatures et $n \geqslant 2$ le nombre de classes. L'algorithme d'apprentissage, quant à lui, résulte de cet ensemble d'apprentissage couplé à d'éventuels paramètres et établit une règle de prédiction permettant d'estimer la classe y d'un nouvel objet x. La Figure 1 schématise le mode de fonctionnement global d'un algorithme de classification supervisée.

2.1.2 Classification binaire et classification multiclasse

La *classification binaire* correspond au cas où il n'y a que 2 classes. C'est une situation courante dans le contexte biomédical, par exemple lorsqu'il s'agit de faire la distinction entre malin et bénin [Jiang et al., 1996 ; Stavros et al., 1995]. Par conséquent, certains algorithmes d'apprentissage s'appliquent uniquement au cas où il n'y a que deux classes. On parle alors de *classifieurs binaires*.

Parmi les plus populaires dans la littérature, on retrouve les séparateurs à vaste marge (en anglais Support Vector Machines (SVM), cette méthode est détaillée en Annexe A) et le Boosting (et plus particulièrement AdaBoost, un meta-algorithme consistant à combiner des classifieurs dits « faibles » afin d'obtenir un classifieur dit « fort » [Schapire, 2003]).

Par opposition au cas binaire, on parle de classification « multiclasse » lorsque le nombre de classe est supérieur ou égal à 3. Là encore, les applications dans le domaine biomédical ne manquent pas. On peut citer à titre d'exemple la classification histologique de la tumeur [Kleihues et al., 2002 ; Sobin et Fleming, 1997] ou la classification de séquences d'ADN [J. T. Wang et al., 1999].

Certains classifieurs binaires peuvent être reformulés afin d'être applicables au cas multiclasse (par exemple AdaBoost [Zhu et al., 2009] et les SVM [Crammer et Singer, 2002]). D'autres sont directement multiclasses. C'est le cas notamment pour les réseaux de neurones artificiels [Bengio et al., 2013], ainsi que les arbres de décision et les forêts d'arbres décisionnels [Breiman, 2001 ; Quinlan, 1986].

2.1.3 Passage du cas binaire du cas multiclasse

Bien que les classifieurs, dont ceux évoqués en section 2.1.2, soient pour la plupart adaptés au cas multiclasse (éventuellement à une reformulation près), il est courant de se ramener au cas binaire. C'est par exemple le cas lorsque le classifieur binaire dispose de propriétés intéressantes (implémentation performante disponible, passage à l'échelle, ...), ce qui a motivé l'introduction de stratégies d'extension d'une méthode de classification binaire au cas multiclasse [Rocha et Klein Goldenstein, 2014].

Pour chaque extension, un nombre arbitraire $K \in \mathbb{N}^*$ de classifieurs binaires sont appris entre des paires de classes (élémentaires [2] et/ou virtuelles [3]). Une partie ou la totalité de ces classifieurs [4] est alors appliquée à une nouvelle signature $x \in \mathbb{R}^n$ pour en prédire son étiquette $y \in \{1, \ldots, n\}$.

2. Une *classe élémentaire* est une des n classes du problème de classification initial.
3. Une *classe virtuelle* est la réunion de classe(s) élémentaires.
4. Une telle extension n'utilise pas nécessairement tous les classifieurs appris pour réaliser la classification, à l'instar de la méthode qui sera proposée en section 2.2.

De nombreuses extensions ont été proposées dans la littérature [Allwein et al., 2001 ; Aly, 2005 ; Hastie, Tibshirani et al., 1998 ; Hsu et Lin, 2002]. Les plus populaires d'entre elles peuvent être classées dans 4 principales catégories : un contre tous, un contre un, code correcteur et division pour régner.

UN CONTRE TOUS Souvent abrégée OVA (de l'anglais « one versus all »), cette famille d'extensions [Athitsos et al., 2007 ; Manikandan et Venkataramani, 2009 ; Mota et Thome, 2009 ; Rifkin et Klautau, 2004] consiste à comparer chaque classe à son complémentaire, c'est-à-dire la classe virtuelle composée de l'ensemble des $n-1$ autres classes. Dans ce cas, le nombre de classifieurs binaires appris K est égal à n. La classe prédite y d'une nouvelle signature x est alors celle qui maximise le critère de confiance [5].

UN CONTRE UN Dans ce cas (OVO – « one versus one »), le classifieur est appliqué à chaque paire de classes, ce qui donne un total de $K = \binom{n}{2} = O(n^2)$ classifieurs binaires utilisés. Lors de la phase de classification, deux principales stratégies peuvent être adoptées pour déterminer la classe de l'objet :

— en appliquant un vote majoritaire (la classe qui récolte le plus de votes l'emporte) [Liu et al., 2007 ; Moreira et Mayoraz, 1998],
— en ayant recours à un graphe orienté acyclique [Kijsirikul et al., 2002 ; Phetkaew et al., 2003 ; J. C. Platt et al., 2000 ; VanderWeele et Robins, 2007].

D'autres approches, souvent moins utilisées, sont envisageables, comme [Garcia-Pedrajas et Ortiz-Boyer, 2006] proposant une stratégie hybride entre les schémas un contre un et un contre tous.

CODE CORRECTEUR Cette méthode, généralement nommée ECOC (pour « error-correcting output-coding ») [Allwein et al., 2001 ; Dietterich et Bakiri, 1995 ; Klautau et al., 2003], fait appel à une matrice ECOC $M \in \{-1, 0, 1\}^{n \times K}$ dans laquelle chaque ligne correspond à une classe et chaque colonne correspond à un classifieur. Pour chaque colonne, et donc pour chaque classifieur, -1 signifie que la classe correspondante appartient à la première classe virtuelle, $+1$ qu'elle appartient à la seconde et 0 qu'elle est ignorée.

Les K classifieurs ainsi définis sont ensuite appliqués à un nouvel objet x. Comme dans le cas précédent, il existe plusieurs façons de prédire la classe $y \in \{1, \ldots, n\}$ de x, parmi lesquelles on retrouve le vote majoritaire et une approche cherchant à minimiser une certaine fonction de coût [Windeatt et Ghaderi, 2003].

5. Généralement, un classifieur binaire retourne une valeur indiquant à quel point la prédiction faite est fiable. Dans le cas des SVM, cette valeur peut être la distance du point à l'hyperplan séparateur : intuitivement, plus le point en est loin, plus les chances d'erreur sont faibles.

Notons que les stratégies OVO et OVA peuvent être vues comme des cas particuliers de la classification par ECOC avec $K = \binom{n}{2}$ et $K = n - 1$, respectivement.

DIVISION POUR RÉGNER Il est naturel d'envisager l'utilisation de la division pour régner pour réaliser une telle généralisation [Chen et al., 2004 ; Madzarov et al., 2009 ; F. Takahashi et Abe, 2002]. Il est en effet possible de construire une *arbre binaire entier*[6] de classifieurs où, à chaque nœud, la classe virtuelle courante est séparée en deux sous-classes virtuelles non-vides. La racine de l'arbre correspond à la classe virtuelle regroupant toutes les classes élémentaires et les feuilles de l'arbre correspondent à chacune des classes élémentaires. Lors d'un parcours en profondeur (donc de la racine à une des feuilles), le nombre de classes dans le nœud courant décroit strictement et passe de n à 1.

Du classifieur binaire utilisé résulte alors une règle de décision permettant, à chaque nœud du graphe, de choisir entre le fils de gauche et le fils droit, et ce récursivement jusqu'à atteindre une feuille. Alors, pour prédire la classe y d'un objet x, il suffit de réaliser un parcours en profondeur de l'arbre en partant de sa racine et en suivant le chemin indiqué par les règles de décisions découlant du classifieur binaire jusqu'à arriver à une feuille.

Cette catégorie correspond typiquement au cas où le nombre de classifieurs utilisé lors de la phase de classification est inférieur au nombre K de classifieurs appris lors de la phase d'apprentissage car seule une partie des nœuds d'un arbre binaire sont parcourus lors d'un parcours en profondeur. Cette propriété s'avère intéressante en apprentissage supervisé puisque la phase de classification est généralement celle dont le temps de calcul importe le plus (Cf. Section 2.1.4).

[Vural et Dy, 2004] propose par exemple de séparer la classe courante en deux sous-classes de même taille (c'est-à-dire contenant le même nombre de classes élémentaires). Ici, le but est de diminuer le nombre de classifieurs utilisés lors de la phase de classification en faisant en sorte d'obtenir un arbre binaire équilibré. Dans ce cas, ce nombre est logarithmique car égal à la profondeur de l'arbre de décision.

2.1.4 Le cas des Margin Trees

Le nombre de classifieurs utilisé par les méthodes évoquées en section 2.1.3 augmente avec le nombre de classes n. La table 1 résume les complexités[7] de ces extensions.

6. Un arbre binaire entier est un arbre dont tous les nœuds possèdent soit zéro, soit deux fils.

7. en terme de nombre de classes

8. Quadratique dans le cas du vote majoritaire, linéaire dans celui des graphes orientés acycliques [Phetkaew et al., 2003].

Table 1 – Nombre de classifieurs appris (phase d'apprentissage) et utilisés (phase de classification) pour les principales extensions.

Méthode	**Complexité**	
	Apprentissage	**Classification**
Un contre un	Quadratique	Linéaire ou quadratique [8]
Un contre tous	Linéaire	Linéaire
ECOC	Polynomial	Polynomial
Division pour régner	Quelconque [9]	Logarithmique ou linéaire

Pour chacune des deux phases, plus le nombre de classifieurs appris ou utilisés est important, plus le temps de calcul risque d'être important également. Par ailleurs, en apprentissage statistique supervisé, le temps de calcul de la phase de classification importe beaucoup plus que celui de la phase d'apprentissage. En effet, une fois l'algorithme et l'ensemble d'apprentissage $\mathcal{S}$ choisis, la phase d'apprentissage n'a besoin d'être réalisée qu'une seule fois tandis que la phase de classification l'est autant de fois que de nouvelles données à étiqueter sont disponibles. La plupart des algorithmes d'apprentissage privilégient donc de temps de calcul de la phase de classification, parfois aux dépens de celui de la phase d'apprentissage.

C'est la raison pour laquelle les approches récursives (de type *diviser pour régner*) ont reçu une attention particulière dans le cadre de l'extension d'un classifieur binaire. Motivés par de tels arguments, Tibshirani et Hastie, 2007 proposent une approche appelée « Margin Trees » inspirée du partitionnement de données (clustering), sous 3 variantes. La variante optimale (en terme de partitionnement maximisant l'épaisseur de la marge [10]), dite gloutonne (« greedy »), est de complexité combinatoire (itération sur l'ensemble des bipartitions de l'ensemble des classes, de cardinal 2^n) lors de la phase d'apprentissage. Bien que la complexité de cette première phase soit d'importance moindre, le cas combinatoire peut s'avérer rédhibitoire lorsque le nombre de classes devient important. Pour pallier ce problème, deux alternatives sous-optimales sont proposées, mais la perte de performances par rapport à l'approche gloutonne est significative.

Dans l'approche qui sera proposée dans ce manuscrit en section 2.2, une autre alternative sera proposée pour s'affranchir d'une complexité combinatoire. Celle-ci se base sur un algorithme de coupe minimale permettant un partitionnement optimal de graphes pondérés, décrit en détail en Annexe B. Cet algorithme, offrant une complexité négligeable par rapport à l'approche gloutonne, ne représente en pratique qu'une partie infime du temps de calcul de la procédure d'apprentissage finale. Une version modifiée de la coupure minimale de graphe est également proposée en section 2.2.3 afin d'encourager un bipar-

9. Exponentiel dans le cas des Margin Trees (Cf. Section 2.1.4), quadratique pour la méthode proposée (Cf. Section 2.2)

10. Plus de détails sur ce critère seront donnés en Section 2.2.

titionnement du graphe équilibré et ainsi approcher une complexité logarithmique par rapport au nombre de classes. La méthode proposée sera illustrée avec les SVM (Cf. Section A) en tant que classifieur binaire et comparée de manière expérimentale à d'autres extensions multiclasses de complexité de classification généralement supérieure.

2.1.5 Un schéma récursif d'extension général

On peut naturellement dégager un schéma d'extension générique permettant de passer d'un classifieur binaire $C_{binaire}$ à un classifieur multiclasse C_{multi} en s'inspirant des algorithmes de type « diviser pour régner » . En effet, si on considère une fonction de bipartition d'un ensemble d'apprentissage B qui, appliquée à un ensemble d'apprentissage $\mathcal{S}$ (ayant $n \geqslant 2$ classes), renvoie deux sous-ensembles d'apprentissage $\mathcal{S}_1$ et $\mathcal{S}_2$ satisfaisant les conditions :

— $\mathcal{S}_1$ et $\mathcal{S}_2$ n'ont aucune élément de même classe en commun : $(x_1, y_1) \in \mathcal{S}_1, (x_2, y_2) \in \mathcal{S}_2 \implies y_1 \neq y_2$,

— $\mathcal{S}_1$ et $\mathcal{S}_2$ forment une bipartition non triviale de $\mathcal{S}$: $\mathcal{S}_1 \subseteq \mathcal{S}$, $\mathcal{S}_2 \subseteq \mathcal{S}$, $\mathcal{S}_1 \cup \mathcal{S}_2 = \mathcal{S}$, $\mathcal{S}_1 \cap \mathcal{S}_2 = \emptyset$, $\mathcal{S}_1 \neq \emptyset$ et $\mathcal{S}_2 \neq \emptyset$,

alors l'algorithme 1 permet d'obtenir un arbre de classifieurs T.

Algorithme 1 Schéma de construction d'un arbre de classifieurs

```
1:  procedure ARBRE_DE_CLASSIFIEURS(𝒮)
2:      L ← ensemble des classes de 𝒮
3:      m ← #L
4:      if m == 1 then
5:          Renvoyer une feuille labellisée par L
6:      else
7:          T ← nœud racine
8:          Créer un nœud labellisé par L à la racine de T
9:          (𝒮₁, 𝒮₂) ← B(𝒮)
10:         Déterminer le classifieur binaire C_binaire entre 𝒮₁ et 𝒮₂)
11:         Associer C_binaire à T
12:         Créer T₁ et T₂, respectivement fils gauche et droit de T
13:         T₁ = ARBRE_DE_CLASSIFIEURS(𝒮₁)
14:         T₂ = ARBRE_DE_CLASSIFIEURS(𝒮₂)
15:         Renvoyer T
16:     end if
17: end procedure
```

Un tel arbre de classifieurs peut être appris à partir d'un classifieur binaire quelconque $C_{binaire}$. En effet, à chaque nœud de T, l'ensemble d'apprentissage courant S est partitionné en deux sous-ensembles d'apprentissage S_1 et S_2 par l'intermédiaire de B. Il est par conséquent possible d'appliquer le classifieur binaire sur S où la première classe virtuelle correspond à la réunion de classes de S_1 et la seconde

à la réunion de celles de S_2. Il en résulte une règle de décision entre S_1 et S_2 permettant de choisir un chemin entre le fils gauche et le fils droit du nœud courant de T. La figure 2 montre un exemple d'arbre obtenu à partir d'un ensemble d'apprentissage à 5 classes.

Pour estimer la classe d'un nouveau point $x \in \mathbb{R}^d$, il suffit donc de réaliser un parcours en profondeur de l'arbre : en partant de la racine, on suit le chemin défini par le classifieur binaire à chaque nœud jusqu'à atteindre une feuille. Son label correspond à la classe prédite $\hat{y}$ de x.

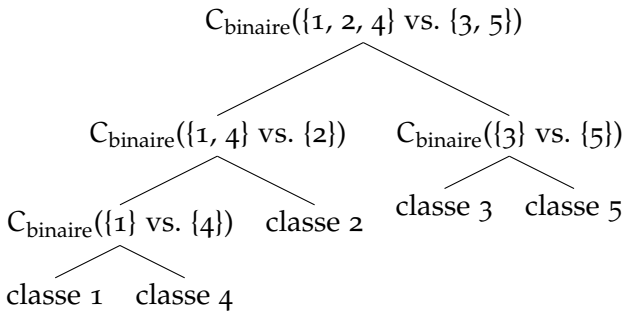


Figure 2 – Exemple d'arbre construit par la procédure générale d'extension de classifieur. Les chiffres correspondent aux indices des classes.

2.2 UNE EXTENSION FONDÉE SUR LA COUPURE DE GRAPHE

2.2.1 Motivations et défis

La possibilité d'atteindre une complexité de classification logarithmique rend les approches hiérarchiques (ou récursives, c'est-à-dire faisant appel à la division pour régner) très compétitives parmi les extensions de classifieurs binaires.

Par ailleurs, lors de la phase de classification, le parcours en profondeur de l'arbre peut être interprété comme un procédé d'affinement de l'ensemble des classes. En effet, à l'instant initial (donc à la racine de l'arbre), un nouvel objet x à classifier fait partie de l'ensemble des classes initial, puis, à chaque niveau de l'arbre descendu, une partie des classes est éliminée, jusqu'à arriver à une feuille correspondant à la classe estimée de x. L'extension OVO n'est pas aussi cohérente dans la mesure où, si la classe de x est $y = k$, elle appliquera chacun des classifieurs binaires C_{ij} entre les classes i et j, y compris dans le cas où $i \neq k$ et $j \neq k$, et les résultats de ces classifications contribueront (de manière équitable) au vote majoritaire.

En revanche, la conception d'une approche récursive pose trois défis majeurs : la pertinence des bipartitions de l'ensemble d'apprentis-

sage, le coût de calcul des bipartitions et l'équilibrage de l'arbre de classifieurs.

ÉQUILIBRAGE DE L'ARBRE L'équilibrage de l'arbre a un impact direct sur la complexité de classification puisqu'elle résulte d'un parcours en profondeur. La figure 3 montre deux cas sur un exemple élémentaire à 4 classes :

— Figure 3a : un *arbre équilibré* garantissant une complexité logarithmique dans tous les cas,
— Figure 3b : un *arbre dégénéré* (également appelé *arbre peigne*, c'est-à-dire dont chaque nœud a au moins un fils étant une feuille), pour lequel la complexité (dans le cas moyen et le pire des cas) est linéaire.

La possibilité d'aboutir à un arbre déséquilibré limite donc l'intérêt d'une approche récursive par rapport au cas OVA.

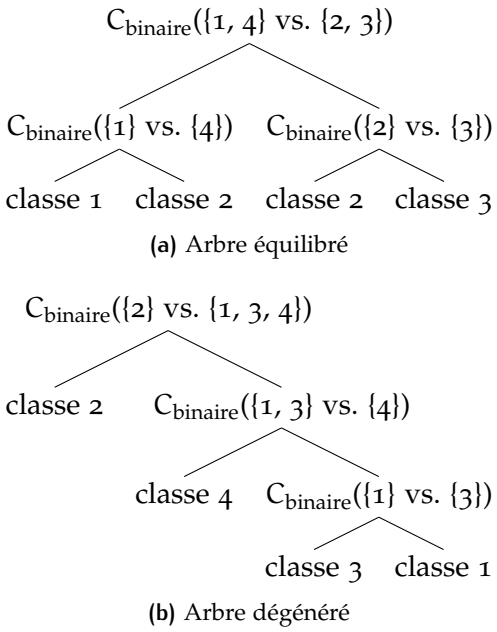


FIGURE 3 – Exemples d'arbres binaires de classifieurs.

PERTINENCE DE LA BIPARTITION C'est un critère que l'on retrouve régulièrement en classification non supervisée (clustering). De manière intuitive, il semble préférable de regrouper des classes « proches » à chaque bipartition [11].

COÛT DE CALCUL DE LA BIPARTITION Comme évoqué en section 2.1.4 dans le cas glouton des Margin Trees, le temps de construction de l'arbre doit rester raisonnable.

11. La notion de proximité de classes sera expliquée plus en détail en Section 2.2.2

2.2.2 Approche proposée : Graph Cut SVM (GCSVM)

Tout en se basant sur le schéma évoqué en Section 2.2.1, l'idée de la méthode proposée est d'utiliser les outils de théorie des graphes pour partitionner l'ensemble des classes de manière efficace et pertinente. Pour cela, on construit un graphe G complet et pondéré de classes dans lequel :

— chaque nœud représente une des n classes,
— le poids c_{ij} de l'arête liant les nœuds i et j est d'autant plus grand que les classes i et j sont « proches » .

Une fois construit, il suffit de lui appliquer un algorithme de bipartionnement (de préférence équilibré) de graphe pour séparer l'ensemble des classes en deux. Le partitionnement de graphe est un problème complexe [Feder et al., 1999 ; Feldmann, 2012] généralement résolu de façon heuristique [Even et al., 1999]. Parmi les approches existantes, on retrouve les algorithmes de *coupe minimale* [Stoer et Wagner, 1997], qui ont l'avantage de proposer une implémentation simple et performante (Cf. Annexe B). La coupe minimale de G donne une bipartition des nœuds minimisant la coupe du graphe, c'est-à-dire la somme des poids des arêtes coupées.

Comme pour les Margin Trees [Tibshirani et Hastie, 2007], nous considérerons dès à présent le SVM comme classifieur binaire (Cf. Annexe A). Ce choix est particulièrement pertinent dans ce cas dans la mesure où la marge d'un SVM appris entre deux classes donne une estimation de la « distance » entre deux classes : plus son épaisseur (donnée par $\frac{2}{\|w\|}$, Cf. Annexe A) est importante, plus les classes peuvent être considérées comme distantes.

Pour la construction du graphe de classe décrit précédemment, on peut alors définir le poids c_{ij} liant les nœuds i et j comme étant égal à l'inverse de la marge du SVM appris entre les classes i et j, c'est-à-dire $c_{ij} = \frac{\|w_{ij}\|}{2}$. Il ne reste donc plus qu'à utiliser l'algorithme de bipartition défini en Annexe B à chaque étape de construction de l'arbre binaire de classifieurs. Cet algorithme va en effet faire en sorte de couper les arêtes de poids faible, c'est-à-dire correspondant à une marge importante.

On peut distinguer deux types de SVM appris lors de l'apprentissage de l'algorithme GCSVM :

1. Phase 1 : Apprentissage des $\frac{n(n-1)}{2}$ SVM entre chaque paire de classes pour la construction du graphe,
2. Phase 2 : Apprentissage des $n-1$ SVM entre les bipartitions de classes à chaque nœud non terminal (Cf. Proposition 2.2.1) pour la construction de l'arbre de classifieurs.

Proposition 2.2.1 (Relation entre le nombre de nœuds et le nombre de feuilles d'un arbre binaire entier [Mehta et Sahni, 2004]). *Pour*

un arbre binaire entier non-vide à n_0 feuille(s) (nœud(s) de degré 0) et n_2 nœud(s) à 2 fils (nœud(s) de degré 2), on a $n_0 = n_2 + 1$.

Démonstration. La preuve se fait par induction sur la construction de l'arbre. Puisque l'on a un arbre binaire entier non vide, le cas de base correspond à une feuille, pour laquelle $n_0 = 1$ et $n_2 = 0$. Pour le cas général, supposons que l'on ait un arbre binaire entier T de fils gauche T^g et de fils droit T^d non vides. Par hypothèse d'induction, T^g et T^d vérifient $n_0^g = n_2^g + 1$ et $n_0^d = n_2^d + 1$, respectivement. Pour T, on a $n_0 = n_0^g + n_0^d$ et $n_2 = n_2^g + n_2^d + 1$, et donc $n_0 = n_2 + 1$. □

À noter qu'entre 1 (dans le cas dégénéré) et $n/2$ (dans le cas équilibré) des classifieurs de la seconde phase ont déjà été appris pendant la première phase. Au total, le nombre de SVM calculé lors de la phase d'apprentissage est donc quadratique. L'algorithme 2, prenant en entrée le graphe complet de classes décrit ci-dessus, résume toutes les étapes de la construction de l'arbre de l'algorithme GCSVM. On peut remarquer que sa structure n'est qu'un cas particulier de l'Algorithme 1.

Algorithme 2 Construction de l'arbre GCSVM

```
procedure ARBRE_GCSVM(G)
    L ← ensemble des nœuds de G
    n ← #L
    if n == 1 then
        Renvoyer une feuille labellisée par L
    else
        T ← nœud racine
        a ← un nœud quelconque de G
        (G1, G2) ← MinCut(G, a)
        Calculer le SVM C_SVM entre S1 et S2)
        Associer C_SVM à T
        Créer T1 et T2, respectivement fils gauche et droit de T
        T1 = ARBRE_GCSVM(G1)
        T2 = ARBRE_GCSVM(G2)
        Renvoyer T
    end if
end procedure
```

Pour la classfication d'un nouvel objet, il ne reste plus qu'à réaliser un parcours en profondeur de l'arbre en suivant le chemin indiqué par les classifieurs. La Figure 4 montre un exemple d'arbre de SVM obtenu dans cas à cinq classes.

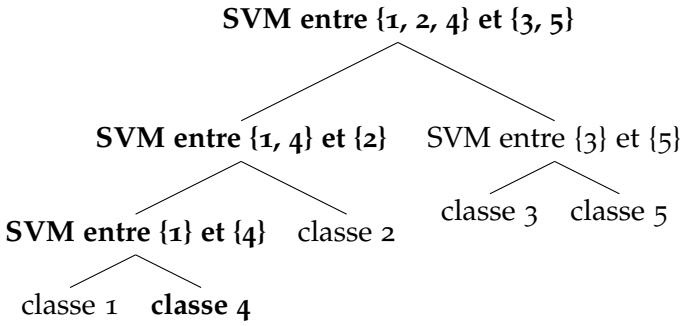


FIGURE 4 – Type d'arbre résultant de la phase d'apprentissage de la méthode proposée. Un exemple de classification prédisant la classe 4 est indiqué **en gras** sur l'arbre.

2.2.3 Équilibrage de l'arbre binaire

Il s'agit d'une étape optionnelle pouvant être utilisée dans le but d'encourager l'équilibrage de l'arbre de décision construit. Comme précisé en Section 2.2.1, l'idée générale est de décomposer un ensemble de n classes en deux sous-ensembles de n_1 et n_2 classes, respectivement, tels que $n_1 + n_2 = n$, et ce de manière récursive jusqu'à arriver au cas de base où le sous-ensemble obtenu n'a qu'une seule classe. Or, pour réduire le nombre de classifieurs utilisés pendant la phase de classification, il suffit de minimiser la profondeur de l'arbre. Ceci peut être réalisé en ajoutant une contrainte d'équilibrage encourageant la construction d'un arbre binaire équilibré, par exemple $|n_1 - n_2| \leqslant 1$.

Formellement, cela revient à définir un nouveau terme d'énergie e égal à la somme de la capacité c, que l'on cherche à minimiser, et d'un *terme de déséquilibre* comme suggéré dans [DELL'AMICO et TRUBIAN, 1998]. Plus précisément, si la coupe de graphe est donnée par $(A, \bar{A})$, alors le terme d'énergie peut être défini par :

$$e(A, \bar{A}) = c(A, \bar{A}) + \lambda \left| |A| - |\bar{A}| \right|$$

où λ est un *hyperparamètre d'équilibrage*, à fixer manuellement, et qui sera d'autant plus grand que l'on souhaite forcer l'équilibrage de la bipartition. Idéalement, on aimerait trouver l'ensemble de classes A minimisant l'énergie $e(A, \bar{A})$, ou alors la bipartition $(A, \bar{A})$ équilibrée de coupe minimale.

Or, l'algorithme de coupe minimale tel qu'il est proposé dans [STOER et WAGNER, 1997] ne permet pas de minimiser une telle énergie. Nous proposons donc, dans l'Annexe B, une heuristique permettant de trouver une bipartition équilibrée de coupe faible.

2.2.4 Complexité

En ce qui concerne la phase d'apprentissage, le nombre de SVM calculés est quadratique par rapport au nombre total de classes.

Proposition 2.2.2. *Le nombre de SVM calculés pendant la phase d'apprentissage est* $O(n^2)$.

Démonstration. Le nombre de SVM calculés est égal à la somme :
- du nombre de SVM binaires calculés lors de la construction du graphe $O(n^2)$,
- Le nombre de nœufs de l'arbre, c'est-à-dire $O(n)$.

Par conséquent, le nombre total de SVM appris pendant la phase d'apprentissage est $O(n^2)$. □

Pour l'étape de classification, le nombre de SVM utilisés dépend de l'arbre binaire de classifieurs construit puisqu'il est égal à la profondeur de la classe estimée. C'est la raison pour laquelle on distingue les complexités dans le meilleur des cas et dans le pire des cas.

Proposition 2.2.3 (Complexité de classification dans le pire des cas)**.** *La complexité de classification dans le pire des cas est* $O(n)$.

Démonstration. Le pire des cas correspond au scénario où l'arbre construit est dégénéré et où la classe prédite est celle de profondeur maximale. La complexité est alors $O(n)$. □

Proposition 2.2.4 (Complexité de classification dans le meilleur des cas)**.** *La complexité de classification dans le meilleur des cas est* $O(\log(n))$.

Démonstration. Dans le meilleur des cas, l'arbre est équilibré, ce qui correspond à une profondeur d'arbre logarithmique. □

Le tableau 2 résume les complexités d'apprentissage et de classification pour la méthode proposée et les extensions les plus souvent utilisées.

TABLE 2 – Complexité d'apprentissage et de classification pour les différentes extensions de SVM : OVO, OVA, ECOC, GCSVM « non équilibré » (Cf. Section 2.2) et GCSVM « équilibré » (Cf. Section 2.2.2).

Méthode	**Complexité**	
	Apprentissage	**Classification**
Un contre un	Quadratique	Linéaire ou quadratique
Un contre tous	Linéaire	Linéaire
ECOC	Polynomial	Polynomial
GCSVM (déf.)	Quadratique	Linéaire/Logarithmique
GCSVM (éq.)	Quadratique	Logarithmique

La méthode proposée offre la complexité de classification la plus faible tout en assurant une complexité d'apprentissage raisonnable.

Par ailleurs, le temps de calcul de la phase d'apprentissage peut être réduit de plusieurs manières, par exemple en sous-échantillonnant l'ensemble d'apprentissage [Bottou et al., 2004].

2.3 RÉSULTATS EXPÉRIMENTAUX

2.3.1 Choix d'implémentation

L'algorithme GCSVM a été implémenté sous Matlab, en utilisant l'interface Matlab de la librairie libSVM [Chang et Lin, 2011]. Les tests ont été faits avec le noyau gaussien (sauf mention contraire explicite), et les hyperparamètres de l'Annexe A ont été choisis par validation croisée [Arlot, Celisse et al., 2010 ; Kohavi et al., 1995].

2.3.2 Résultats sur des données 2D synthétiques

Avant d'appliquer les algorithmes à des données réelles, il est pertinent de s'intéresser à des données générées aléatoirement afin d'avoir une intuition du comportement de la méthode lors de la construction de l'arbre de classifieurs et de faire le lien entre le paramètre d'équilibrage de l'algorithme de coupe minimale et l'arbre de classifieurs construit.

Un total de 8 classes de 100 points du plan sont générées. Chaque classe est définie par son centre, un angle de rotation (par rapport à l'axe des abscisses) et deux variances (une par axe). Pour chacune des classes, les 100 points sont générés selon une distribution gaussienne. Compte tenu du fait que $8 = 2^3$, la version équilibrée de la méthode proposée devrait donner un arbre de profondeur 3 parfaitement équilibré.

L'apprentissage de l'algorithme GCSVM est lancé sur cet ensemble de 800 points. Le classifieur ainsi appris est ensuite appliqué à chaque point d'un échantillonnage régulier du plan de sorte à pouvoir visualiser le partitionnement de l'espace que réalise l'algorithme à chaque niveau de profondeur de l'arbre.

La Figure 5 montre les différentes étapes de l'algorithme GCSVM sans que l'équilibrage ne soit forcé. L'arbre qui en résulte est dégénéré : à chaque bipartition, une classe est éliminée de l'ensemble des classes courant. L'arbre correspondant est donc de profondeur maximale (égale à 7), et la complexité de classification moyenne est linéaire.

La Figure 6, quant à elle, illustre les étapes successives de la classification dans le cas où l'équilibrage est encouragé. On passe alors d'un arbre dégénéré à un arbre parfaitement équilibré : à chaque partitionnement, l'ensemble des classes est séparé en deux sous-ensembles de même taille (en terme de nombre de classes).

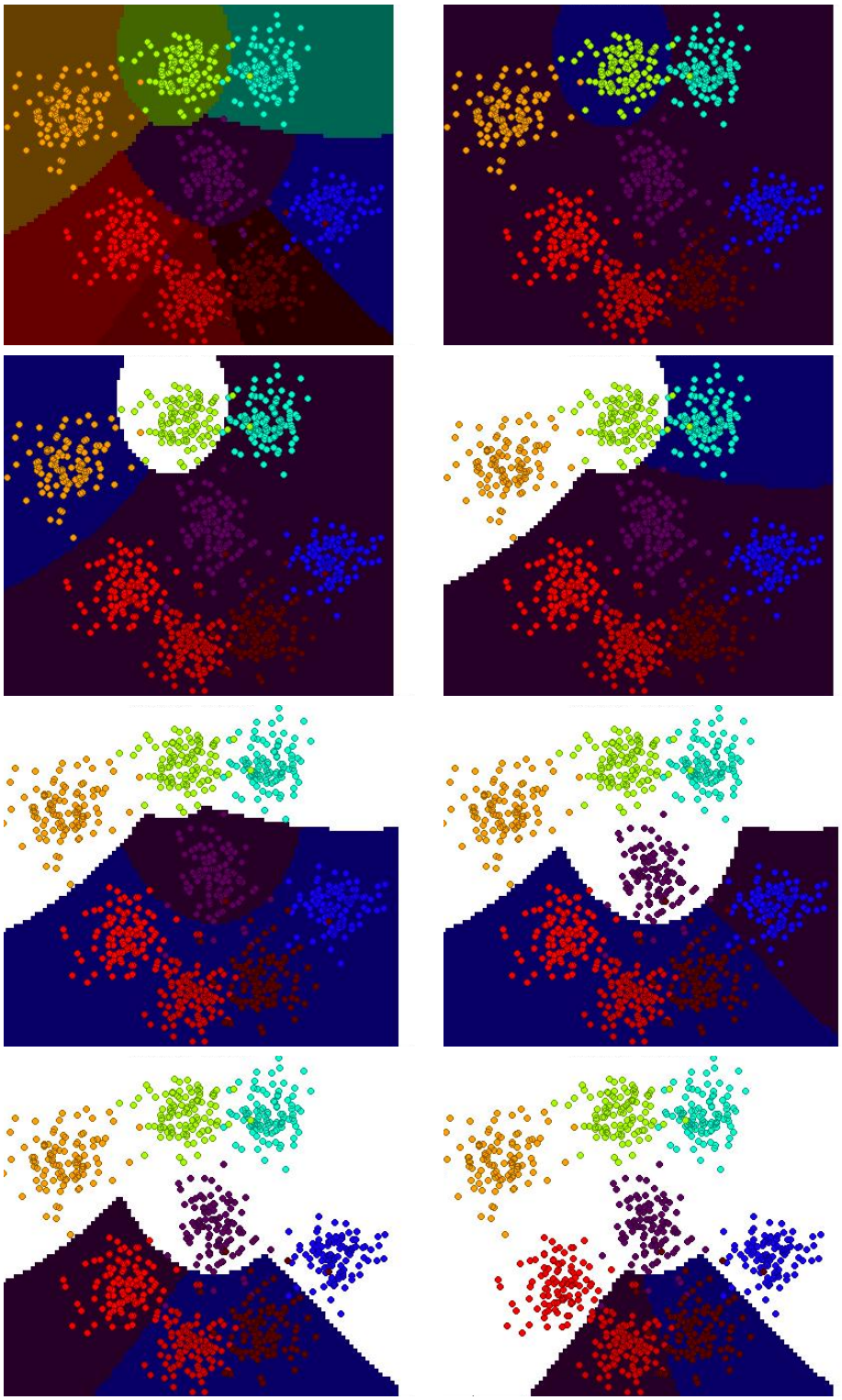


FIGURE 5 – Les étapes de l'algorithme GCSVM standard.

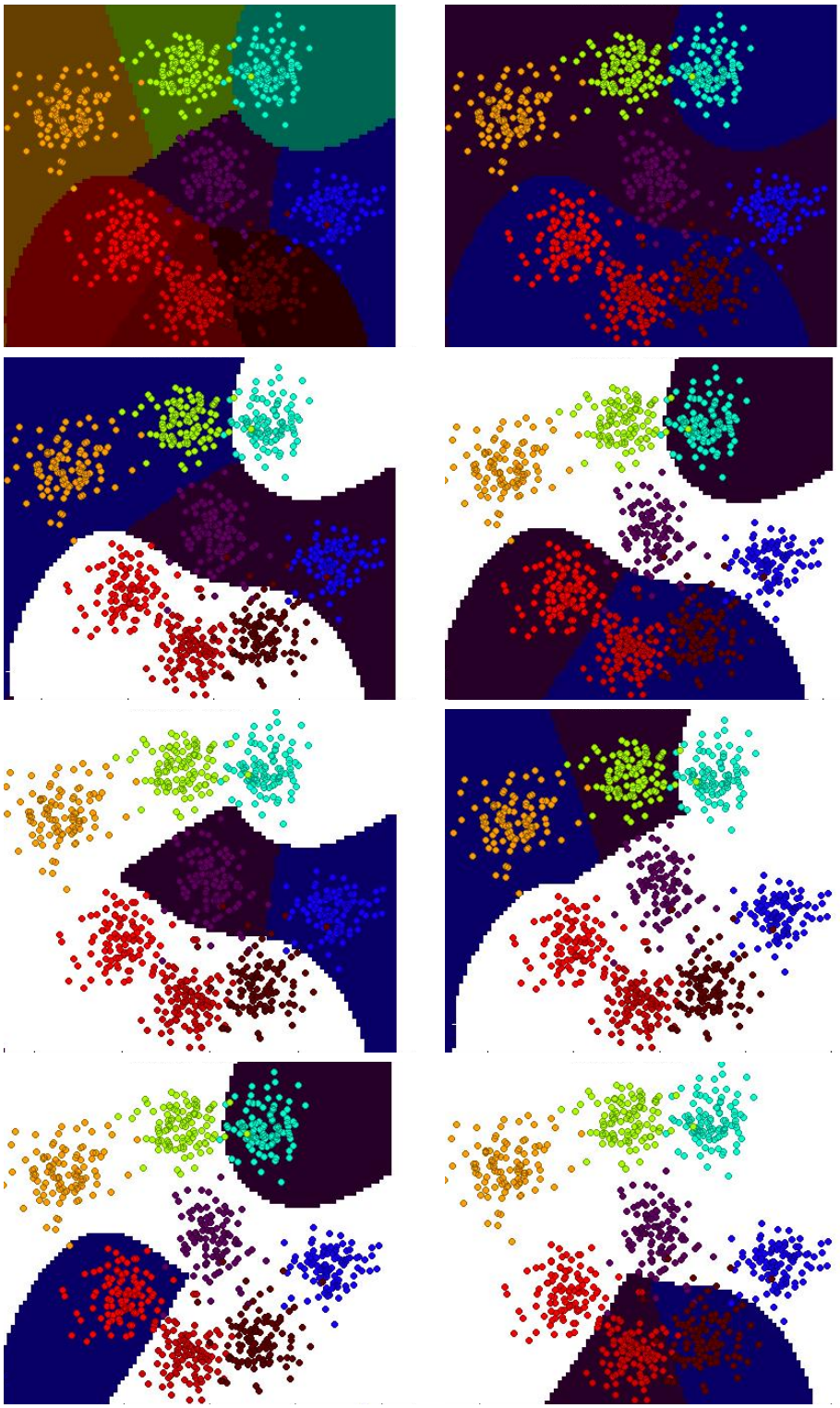


Figure 6 – Les étapes de l'algorithme GCSVM équilibré.

Sur chacune des figures 5 et 6, la première image correspond au résultat final (c'est-à-dire le partitionnement de l'espace en classes estimé par l'algorithme). Les autres images décrivent les étapes successive de décomposition du plan. Il y a 3 couleurs de fond :

— Violet : la première sous-classe,
— Bleu : la seconde sous-classe,
— Blanc : les classes ignorées (*i.e.* les classes faisant partie d'une autre classe virtuelle et donc traitées à un autre nœud de l'arbre de décision).

2.3.3 Résultats en reconnaissance de lettres

On applique désormais l'algorithme sur un ensemble d'images de lettres [Frey et Slate, 1991] à reconnaître issu de dépôt de l'Université d'Irvine [Lichman, 2013]. Cette base de données est constituée de :

— 26 classes (correspondant aux 26 lettres de l'alphabet latin),
— 20,000 images (entre 734 et 813 par classe),
— 16 valeurs décrivant chaque image (comme la position, la hauteur et la largeur de la plus petite *bounding box* délimitant la lettre, . . .).

La base de données a été séparée (de manière aléatoire) en deux parties de même taille, la première moitié servant à l'apprentissage, et l'autre à la classification. Le tableau 3 montre les résultats obtenus par différentes méthodes.

On constate que toutes les extension multiclasse du SVM, y compris la méthode proposée, montrent d'excellents résultats sur cet ensemble de test, que ce soit en terme de précision ou de temps de calcul. On peut également remarquer que le temps de calcul de la méthode GCSVM équilibré est supérieur à celui de la version non équilibrée. Ceci peut paraître surprenant dans la mesure où la complexité de la version équilibrée est inférieure. La différence s'explique par le fait que la complexité donne l'ordre de grandeur asymptotique alors que l'exemple donné ne compte 26 classes.

2.3.4 Résultats sur un ensemble de vidéos d'endomicroscopie

Cette base de données médicale est composée de 116 vidéos d'endomicroscopie issues de coloscopies réalisées sur 65 patients. Chacune d'entre elle montre un polype du colon *in vivo* au niveau cellulaire. Les vidéos ont été acquises à la clinique Mayo (à Jacksonville), en utilisant une technologie appelée pCLE (pour *Probe-based Confocal Laser Endomicroscopy* [Wallace et Fockens, 2009]) développée par Mauna Kea Technologies.

Chaque vidéo est associée à une des 5 classes pathologiques possibles issue du diagnostique histologique établi par un expert suite à

Méthode	Option	Précision (%)	Temps d'apprentissage (s)	Temps de classification (s)
Forêts aléatoires	50 arbres	89.68	62	8.0
	100 arbres	90.21	122	15.4
	500 arbres	90.81	596	74.1
	1 000 arbres	90.88	1192	146.1
	5 000 arbres	91.02	6114	754.6
	10 000 arbres	91.06	12460	1533.8
Margin Trees		89.43	26798	4.4
SVM OVO		96.29	8026	57.3
SVM OVA		96.61	251239	11.4
DAGSVM		96.30	8035	4.8
GCSVM	sans équilibrage	**95.52**	72129	4.6
	avec équilibrage	**95.97**	19257	8.7

TABLE 3 – Précision pour les forêts d'arbres décisionnels [BREIMAN, 2001], les principales extensions pour le SVM, les Margin Trees [TIBSHIRANI et HASTIE, 2007], et la méthode proposée (avec ou sans équilibrage).

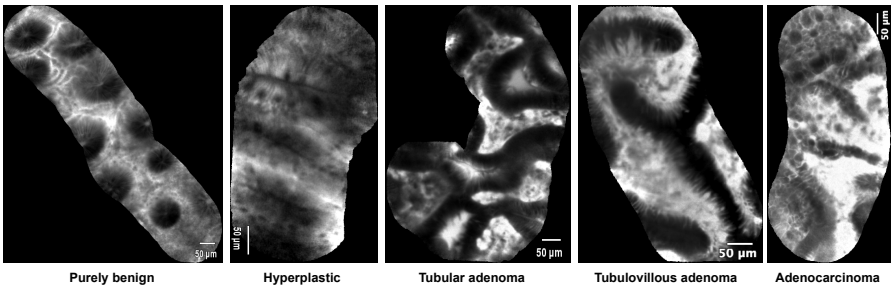


FIGURE 7 – Exemples de vidéos endomicroscopiques montrant des polypes. De gauche à droite : purement bénin, hyperplastique, adénome tubuleux, adénome tubulo-villeux, adénocarcinome.

une biopsie optique. Les 5 classes pathologiques de polypes présentes sont :

— purement bénin (14 vidéos),
— hyperplastique (21 vidéos),
— adénome tubuleux (62 vidéos),
— adénome tubulo-villeux (15 vidéos),
— adénocarcinome (4 vidéos).

La Figure 7 montre une image prise d'une vidéo correspondant à chacun des cas pathologiques.

Dans [André et al., 2011], une méthode basée sur les sacs de mots visuels (*bag of visterms* ou *bag of visual words* [Deselaers et al., 2008 ; J. Yang et al., 2007]) a été proposée afin d'obtenir des descripteurs de ces vidéos dans le but d'en prédire la classe pathologique. Cette prédiction est réalisée en utilisant un algorithme de k plus proches voisins (k-*Nearest Neighbors* – k-NN en anglais) utilisant un vote majoritaire pondéré et la distance du χ^2.

Nous proposons alors d'utiliser les mêmes descripteurs visuels et de comparer les résultats de classification obtenus dans [André et al., 2011] à ceux des extensions multiclasses de SVM, dont la méthode proposée.

Compte tenu de la petite taille de la base de donnée, les méthodes de classification ont été appliquées aux descripteurs par une validation croisée de type *Leave one patient out* (LOPO) [André et al., 2011]. Ce protocole, inspiré du *Leave one out* (LOO) [Kearns et Ron, 1999], consiste, pour chaque patient :

— à réaliser l'apprentissage sur l'ensemble de vidéos correspondant aux autres patients,
— à classifier la (les) vidéo(s) correspondant au patient.

Dans [André et al., 2011], le LOPO est préféré au LOO pour ne pas biaiser les résultats de classification. En effet, deux polypes du même patient ont des chances de présenter des caractéristiques morphologiques communes, et donc d'avoir des vecteurs de descripteurs relativement proches les uns des autres. Par ailleurs, c'est une procédure plus fidèle à la réalité dans la mesure où, pour une nouvelle vidéo

d'endomicroscopie à classer, on ne dispose d'aucune information à propos des autres polypes du patient.

Les résultats de classification de vidéo de chacune des méthodes sont donnés dans le Tableau 4.

TABLE 4 – Taux de bonne classification pour [ANDRÉ et al., 2011] et quelques extensions multiclasses.

Méthode	Précision
ANDRÉ et al., 2011	62.9%
OVO	63.5%
OVA	65.0%
DAGSVM	62.7%
GCSVM	**67.1%**

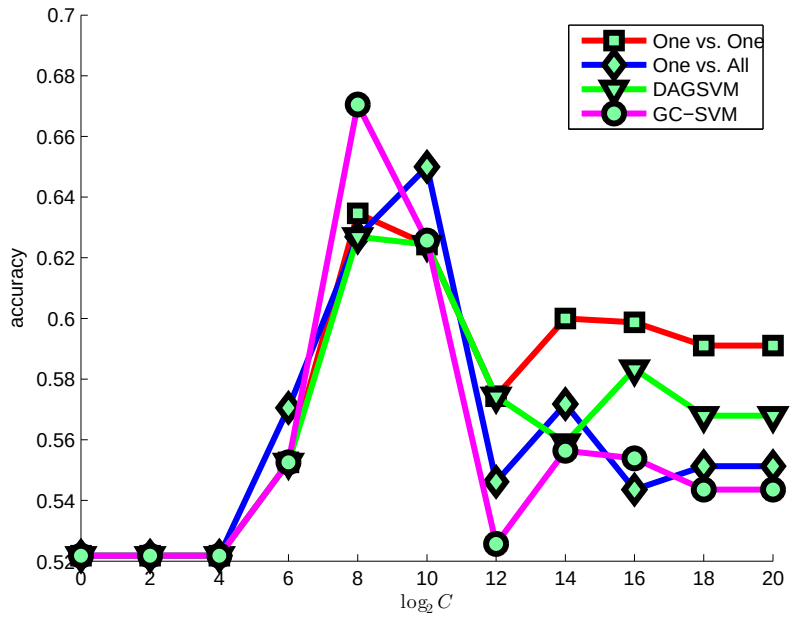
On constate que les extensions des SVM sont toutes plus précises que la version des k-NN utilisés dans [ANDRÉ et al., 2011]. Parmi elles, la méthode proposée réalise de meilleurs résultats sur cette base de données.

2.4 CONCLUSIONS ET PERSPECTIVES

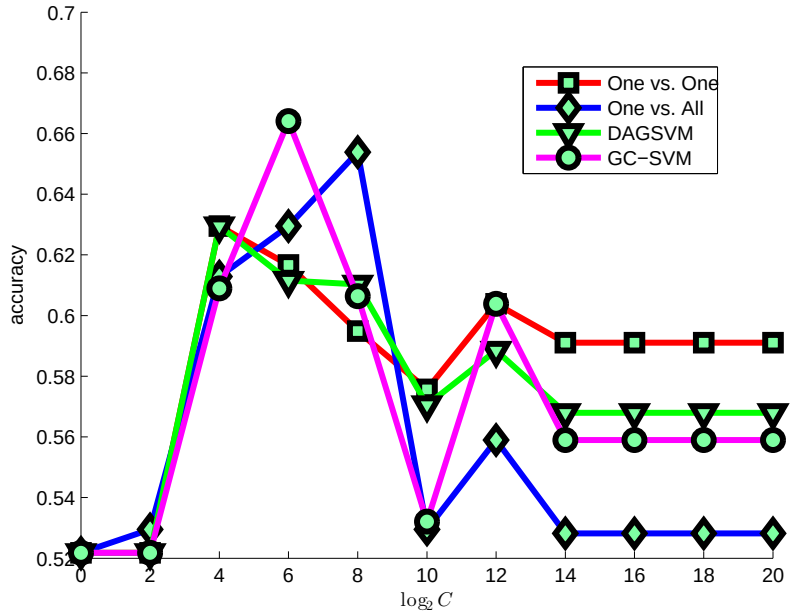
Les résultats de la Section 2.3 sont encourageants puisque l'algorithme GCSVM fait aussi bien (Section 2.3.3) voire mieux (Section 2.3.4) que les extensions classiques de classifieurs binaires, tout en invoquant un nombre moindre de classifieurs binaires (Section 2.3.2) et donc en garantissant une meilleure complexité de classification.

Le choix des hyperparamètres (dont l'équilibrage) est laissé à l'utilisateur ou peut être réalisé de façon automatique par validation croisée. Ce problème de calibrage de paramètre concerne toutes les extensions de classifieurs binaires (et plus généralement les algorithmes d'apprentissage). La Figure 8 illustre ce problème en montrant l'évolution de la précision en fonction du choix du paramètre C et du noyau (Cf. Annexe A) dans le cas de la classification des vidéos d'endomicroscopie (Cf. Section 2.3.4). Certaines approches ont été proposées dans le but d'automatiser cette tâche par estimation directe [EITRICH et LANG, 2006] ou par apprentissage [CORTES, MOHRI et al., 2008], dans le but d'éviter la validation croisée.

Enfin, puisque l'Algorithme 1 de construction d'arbre de classifieurs est générique, il pourrait être intéressant d'essayer l'implémentation d'autres algorithmes de bipartition (potentiellement équilibrés) [ANDREEV et RACKE, 2006 ; KRAUTHGAMER et al., 2009 ; H. H. YANG et D. WONG, 1996] et d'autres classifieurs binaires.



(a) Noyau gaussien : $k(x_i, x_j) = \exp\left(\gamma \left\|x_i - x_j\right\|^2\right), \gamma \in \mathbb{R}$



(b) Noyau polynomial (de degré 2) : $k(x_i, x_j) = \left(\langle x_i, x_j \rangle + 1\right)^2$

FIGURE 8 – Précision en fonction du paramètre C (discrétisé selon une échelle logarithmique) pour les extensions du SVM (OVO, OVA, DAGSVM, et la méthode proposée).

Deuxième partie

Classification non supervisée

3 ANALYSE MORPHOLOGIQUE DE NEURONES DE SOURIS

Sommaire

3.1 Introduction 31
- 3.1.1 Contexte biologique : le néocortex mammifère 31
- 3.1.2 Présentation des données 33
- 3.1.3 Défis et enjeux 34

3.2 Étude des images x40 36
- 3.2.1 Segmentation des neurones 36
- 3.2.2 Calcul des descripteurs 36

3.3 Étude des images x10 40
- 3.3.1 Détection des neurones 40
- 3.3.2 Reconstruction des neurones 41

3.4 Mise en correspondance des neurones 43
- 3.4.1 Définition du problème 43
- 3.4.2 Motivation 44
- 3.4.3 Formalisation du problème : appariement d'ensembles de points de l'espace 44
- 3.4.4 Résolution des problèmes d'optimisation linéaire 46
- 3.4.5 Cas d'ambiguïté 47
- 3.4.6 Appariement de données synthétiques 47
- 3.4.7 Application à la correspondance de neurones 49

3.5 Interprétation des résultats 49
- 3.5.1 Histogrammes des distributions 49
- 3.5.2 Classification par k-moyennes 53

3.6 Conclusion et perspectives 53

3.1 INTRODUCTION

3.1.1 Contexte biologique : le néocortex mammifère

Le *néocortex* des mammifères s'organise en 6 couches (numérotées de I à VI, I étant la plus externe et VI la plus profonde) au sein desquelles les populations de neurones peuvent exprimer différents gènes [Gage, 2000]. Comme le montre la Figure 9, ces neurones sont composés :

— d'un *corps cellulaire* (ou *péricaryon*, ou *soma*), correspondant à la partie centrale,
— de dendrites, qui sont des prolongements du corps cellulaire, pouvant présenter un aspect arborescent. Chaque neurone présente deux types de dendrites : une dendrite principale (dite *apicale*), allant du corps cellulaire à la surface du cortex, et des dendrites secondaires (ou *basales*), plus courtes, plus fines, et réparties autour du corps cellulaire.

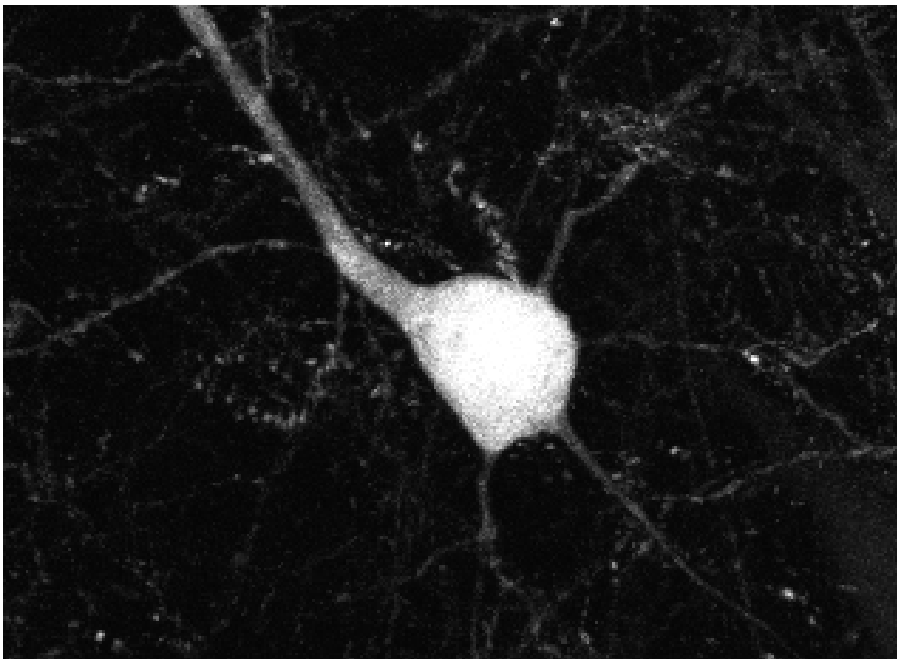


FIGURE 9 – Principales parties d'un neurone : le corps cellulaire et les dendrites (apicale et basales).

Les différentes expressions génétiques entraînent, entre autres, une variabilité au niveau de la connectivité et de la morphologie des neurones du cortex [HATTOX et NELSON, 2007]. En effet, pour ce qui est de la connectivité, la dendrite apicale peut commencer sa ramification à une distance plus ou moins grande du corps cellulaire (et donc dans différentes couches du néocortex). La Figure 10 illustre différents types de connectivité que l'on peut retrouver entre des neurones de la couche V (c'est-à-dire ceux dont le corps cellulaire appartient à la couche V).

En ce qui concerne la morphologie, le corps cellulaire peut être plus ou moins gros, présenter une géométrie [1] différente et avoir un nombre de dendrites secondaires qui varie d'un neurone à l'autre (Cf. Figure 12).

Les différences observées motivent par conséquent l'étude morphologique de tels neurones afin d'en extraire des descripteurs caractéris-

1. La notion de *géométrie* du corps cellulaire sera expliquée de manière plus détaillée en Section 3.2.

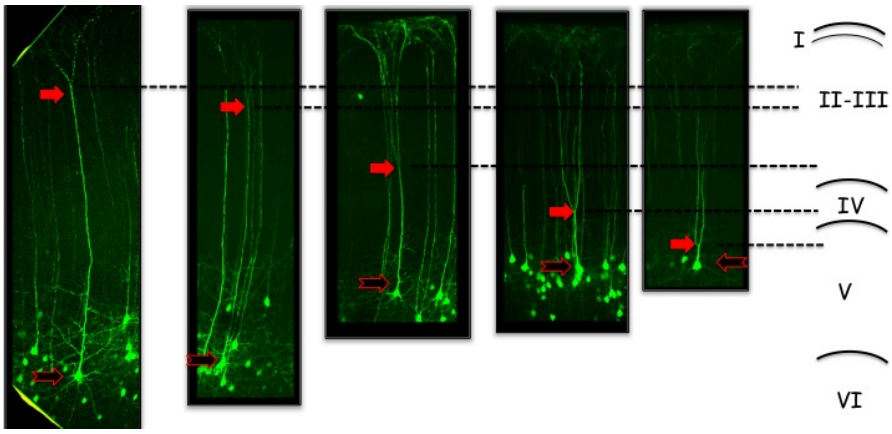


FIGURE 10 – Exemples de types de connectivité pour les neurones de la couche V du néocortex

tiques permettant de les distinguer et de les classifier. Nous nous restreindrons, dans ce manuscrit, à l'étude des neurones de la couche V.

3.1.2 Présentation des données

Microscopie confocale en fluorescence

Les neurones ont été observés grâce à un *microscope confocal à balayage laser* (*MCBL*, de modèle ZEISS LSM 710) [WILSON, 1990]. Il s'agit d'un microscope optique permettant l'acquisition d'images tridimensionnelles. Pour ce faire, le plan focal de l'objectif est positionné à différents niveaux de profondeur dans l'échantillon observé. L'image est alors numérisée et peut être observée par ordinateur.

La fluorescence des neurones est rendue possible par la protéine Thy1-eYFP-H [FENG et al., 2000]. Il est important de remarquer que tous les neurones ne réagissent pas de manière égale à cette protéine (Cf. Figures 11 et 12) et apparaissent alors plus ou moins intensément sur les images produites. Nous reviendrons sur ce détail en Section 3.1.3.

Images x10

Le premier type d'image auquel nous sommes confrontés correspond au zoom x10, pour lequel la résolution est de $1.38\mu m \times 1.38\mu m \times 1\mu m$. Ces images montrent les neurones en entier, du corps cellulaire à son extrémité, la *touffe apicale* (*apical tuft* en anglais). Une des images x10 mises à disposition se trouve en Figure 11.

Ces images permettent en particulier d'observer la dendrite apicale dans son intégralité, et plus précisément d'en localiser la première ramification (Cf. Figure 10).

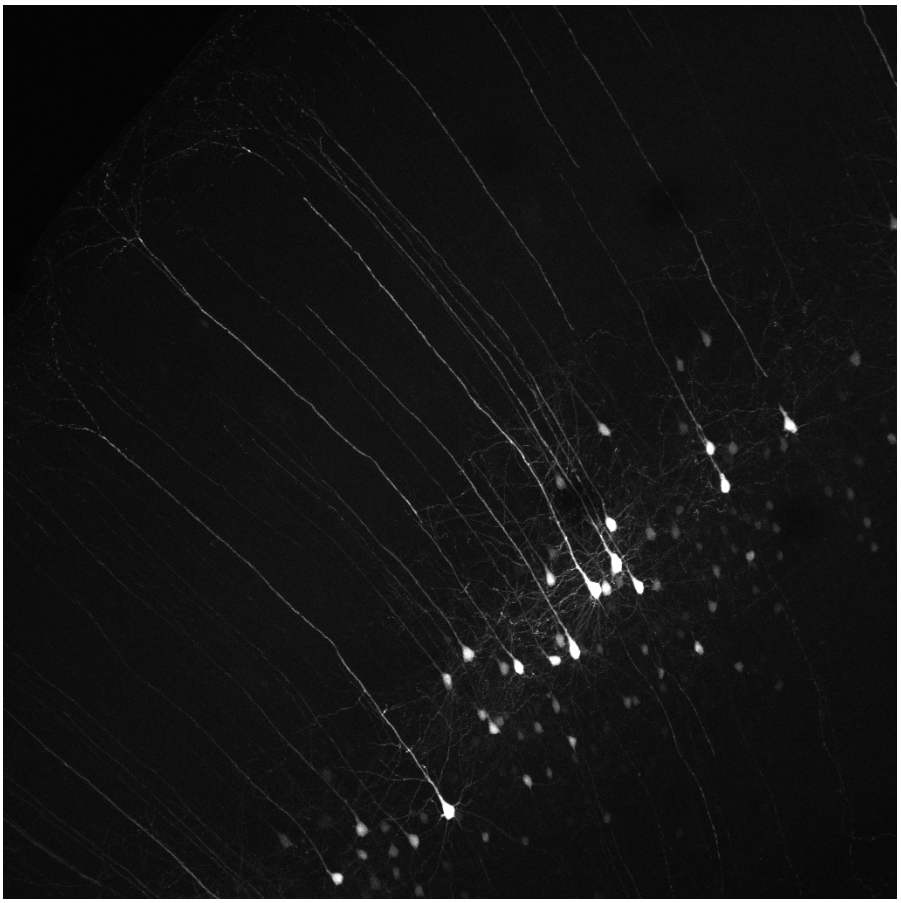


FIGURE 11 – Projection du maximum d'intensité d'une des images x10.

Images x40

Pour chaque image x10, une série de 2 à 6 images x40 correspondantes est acquise. Ces images x40 se focalisent sur le corps cellulaire des neurones et permettent donc de l'observer de façon plus détaillée. En l'occurrence, contrairement aux images x10, les dendrites basales sont visibles (Cf. Figure 12) et peuvent donc être étudiées, ce qui n'était pas possible sur l'image x10.

Taille des données

Nous disposons d'un total de 8 ensembles d'images tridimensionnelles. Chaque ensemble d'image est composé d'une image au zoom x10 ainsi que de plusieurs (entre 2 et 6) images x40 correspondantes. Chaque image compte entre 30 et 100 coupes de taille 1024×1024 et au format 8-bits.

3.1.3 Défis et enjeux

L'étude de telles images pose un nombre de problèmes dont il faut tenir compte. Tout d'abord, comme évoqué en Section 3.1.2, les neu-

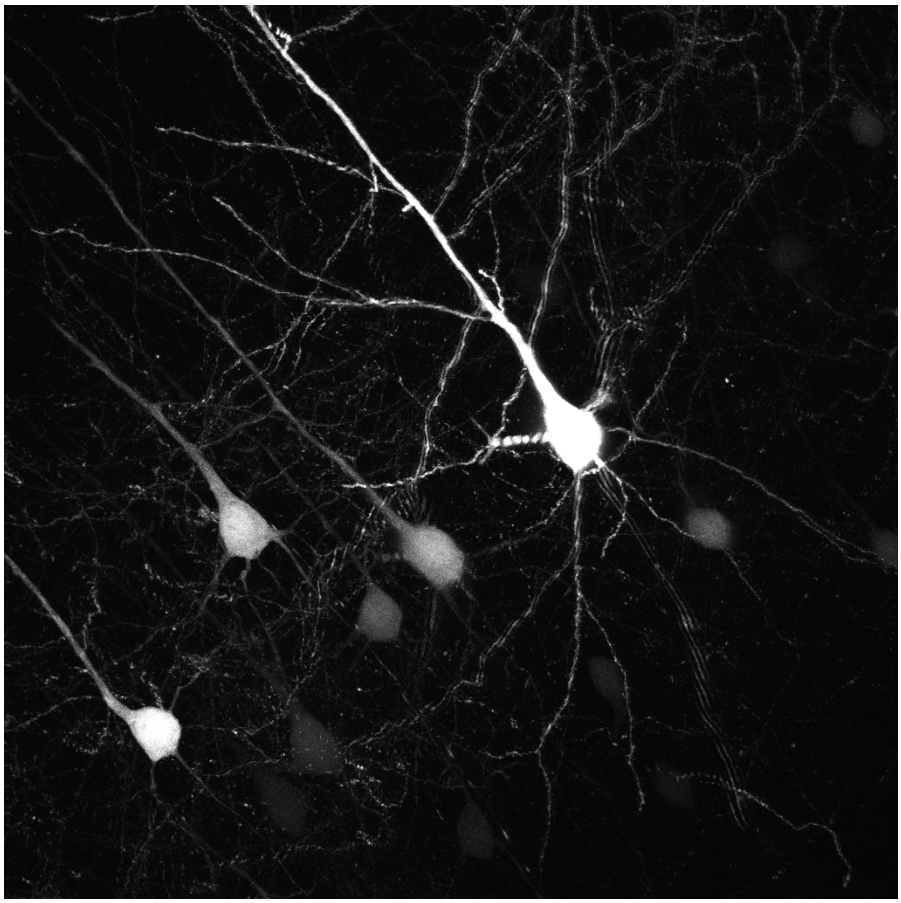


FIGURE 12 – Projection du maximum d'intensité d'une des images x40.

rones ne réagissent pas tous de façon égale à la protéine Thy1-eYFP-H, et n'apparaissent donc pas avec la même intensité sur l'image (Cf. Figure 12). Par ailleurs, au sein d'un même neurone, l'intensité est hétérogène puisque le corps cellulaire est systématiquement plus lumineux que les dendrites (Cf. Figure 9). Cette *double hétérogénéité* demande donc une attention particulière lors de la détection et de la segmentation des neurones (Cf. Section 3.2.1).

Les images présentent également une propriété d'*anisotropie* puisque la résolution en z diffère de celles en x et y, et ce pour les deux types d'images mis à disposition (Cf. Section 3.1.2).

Les deux types d'images sont également bruités, ce qui rend parfois la distinction entre les dendrites (en particulier lorsqu'elles sont ramifiées) et le bruit délicate.

Enfin, il existe une dualité entre les images x10 et x40. Les deux types d'images permettent en effet d'observer des parties différentes d'un même neurone (la dendrite apicale pour les images x10, le corps cellulaire et les dendrites basales pour les images 40x), et les caractéristiques extraites de chaque image sont complémentaires. Par conséquent, en plus d'analyser les images x40 (Cf. Section 3.2) et les images x10 (Cf. Section 3.3), il est nécessaire d'établir une correspon-

dance entre les neurones présents sur les deux types d'image (Cf. Section 3.4).

3.2 ÉTUDE DES IMAGES X40

Nous commençons par l'étude des images x40 montrant de façon détaillée le corps cellulaire des neurones. L'objectif final de cette étape est d'en extraire un vecteur de descripteurs morphologiques caractérisant chacun des neurones visible. Avant de calculer ces paramètres (Section 3.2.2), une étape de détection des neurones est nécessaire (Section 3.2.1).

3.2.1 Segmentation des neurones

Compte tenu du bruit et de la variation d'intensité au sein d'un même neurone, la détection des neurones est réalisée en plusieurs temps : détection du corps cellulaire (plus gros et plus lumineux) suivi de la détection progressive des dendrites (plus fines et moins intenses). Les détails de la procédure sont donnés par l'Algorithme 3 et les résultats de la détection des neurones sont donnés en Figure 13.

Algorithme 3 Les étapes de la détection de neurones sur les images x40.

1: **Seuillage :** Soit I l'image initiale et $T_1(I)$ (respectivement $T_2(I)$ and $T_3(I)$) les résultats du seuillage de I avec un seuil t_1 (respectivement t_2 and t_3) manuellement choisis tels que $t_1 > t_2 > t_3$.

2: **Seuillage par hystérésis :** Soit $\{C_1(i), i = 1, \ldots, n_1\}$ et $\{C_2(j), j = 1, \ldots n_2\}$. Les composantes connexes respectives de $T_1(I)$ et $T_2(I)$, les neurones sont initiallements définis par :

$$N = \{N_k : \exists i, j, N_k = C_2(j), N_k \cap C_1(i) \neq \emptyset\} \tag{1}$$

3: **Reconstruction :** Soit $u = 0$, $\forall k, N_k^{(0)} = N_k$. Itérer, jusqu'à ce qu'aucun changement n'apparaisse : pour $k = 1, \ldots, n_k$

$$N_k^{(u+1)} = D(N_k^{(u)}, b_1) \cap \left(T_3(I) / \cup_{v=1}^{k-1} N_v^{(u+1)}\right) \tag{2}$$

où $D(A, b_1)$ est la dilatation de A par la sphère unité.

3.2.2 Calcul des descripteurs

Une liste de descripteurs morphologiques a été calculée à partir de la segmentation des neurones décrite en Section 3.2 :

— La taille du corps cellulaire (en μm^3),

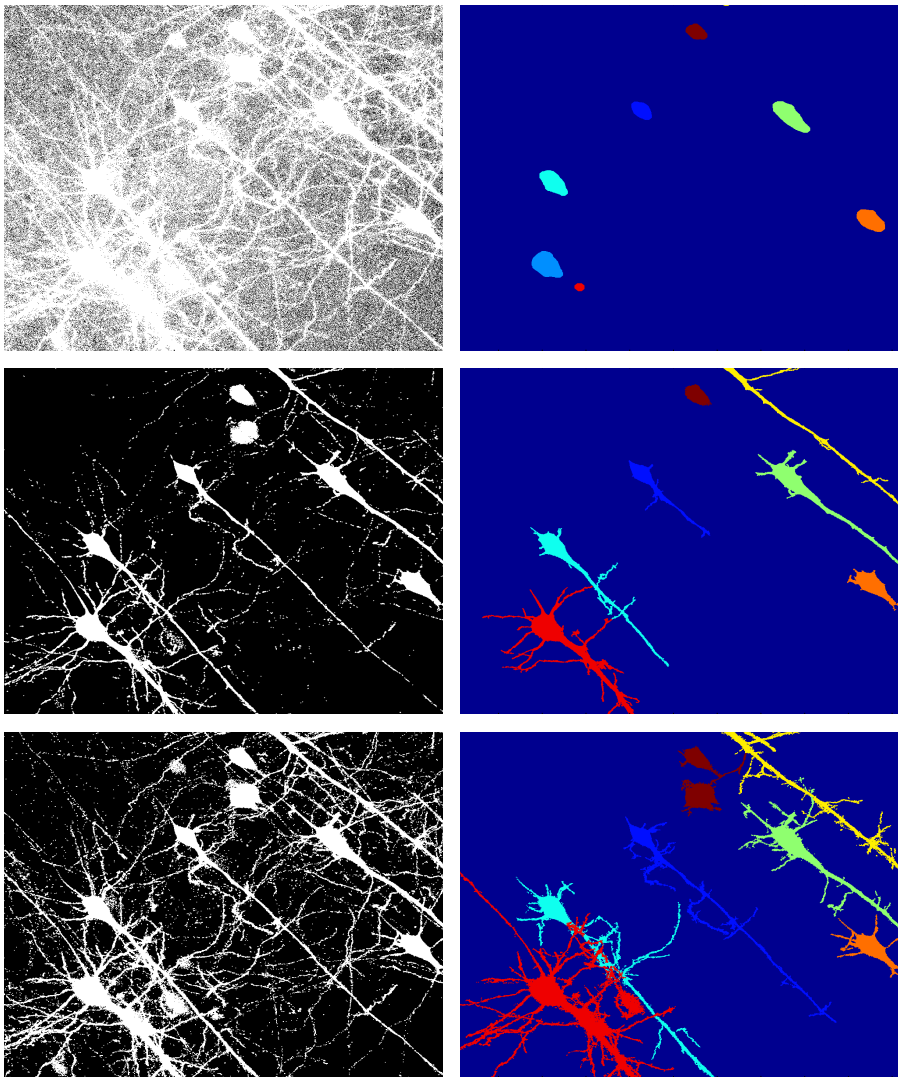


FIGURE 13 – Les étapes de la segmentation des neurones. Image initiale (haut gauche), détection des corps cellulaires correspondant au premier seuil (haut droite), deuxième seuil (milieu gauche), première segmentation (milieu droite), troisième seuil (bas gauche) et résultat final (bas droite). Les images montrées correspondent à la projection du maximum d'intensité.

— La forme du corps cellulaire (détaillé ci-après),
— Le nombre de dendrites,
— La section de la dendrite apicale (en μm^2),
— La distribution angulaire des dendrites secondaires (détaillé ci-après).

Taille du corps cellulaire

Ce critère découle directement de la segmentation du corps cellulaire (premier seuil) et est donc égal au produit entre le nombre de voxels de la segmentation et la taille d'un voxel.

Forme du corps cellulaire

Au dela de la taille du corps cellulaire, sa forme peut constituer un critère permettant de distinguer deux neurones : celui-ci peut en effet être plus ou moins allongé, ou sphérique selon les cas. Pour estimer leur forme, les sept descripteurs de forme proposés dans [Hu, 1962] ont été appliqués à l'image binaire du neurone segmenté. Il s'agit de moments de forme centrés invariants par translation, rotation, et changement d'échelle. Pour une image $I = (I(j,k))_{\substack{1\leqslant j\leqslant J\\ 1\leqslant k\leqslant K}}$, ils sont définis par :

— $h_1 = N_{2,0} + N_{0,2}$,
— $h_2 = (N_{2,0} - N_{0,2})^2 + 4N_{1,1}^2$,
— $h_3 = (N_{3,0} - 3N_{1,2})^2 + (3N_{2,1} - N_{0,3})^2$,
— $h_4 = (N_{3,0} - N_{1,2})^2 + (N_{2,1} - N_{0,3})^2$,
— $h_5 = (N_{3,0} - 3N_{1,2})(N_{3,0} + N_{1,2})[(N_{3,0} - N_{1,2})^2 - 3(N_{2,1} + N_{0,3})^2]$
$+ (3N_{2,1} - N_{0,3})(N_{2,1} + N_{0,3})[3(N_{3,0} + N_{1,2})^2 - (N_{2,1} + N_{0,3})^2]$,
— $h_6 = (N_{2,0} + N_{0,2})[(N_{3,0} + N_{1,2})^2 - (N_{2,1} + N_{0,3})^2]$
$+ 4N_{1,1}(N_{3,0} + N_{1,2})(N_{2,1} + N_{0,3})$,
— $h_7 = (3N_{2,1} - N_{0,3})(N_{3,0} + N_{1,2})[(N_{3,0} + N_{1,2})^2 - 3(N_{2,1} + N_{0,3})^2]$
$- (N_{3,0} - 3N_{1,2})(N_{2,1} + N_{0,3})[3(N_{3,0} + N_{1,2})^2 - (N_{2,1} + N_{0,3})^2]$,

où

$$N_{n,m} = \frac{J^m K^n}{M_{0,0}} M_{m,n}$$

et

$$M_{m,n} = \frac{1}{J^m K^n} \sum_{j=1}^{J} \sum_{k=1}^{K} (j - X_C)^m (k - Y_C)^n I(j,k).$$

Étant donné que ces descripteurs sont définis pour une image bidimensionnelle, ils ont été appliqués à la projection de la détection du corps cellulaire selon l'axe z.

Nombre de dendrites

Pour calculer le nombre de dendrites, il suffit d'en détecter la base. Pour cela, deux dilatations C_{D_1} et C_{D_2} sont appliquées successive-

ment à la segmentation du corps cellulaire C faite en Section 3.2.1 de sorte que $C \subset C_{D_1} \subset C_{D_2}$ (Cf. Figure 14c).

Le nombre de dendrites du neurone est alors donné par le nombre de composantes connexes de l'ensemble $B = N^f \cap (C_{D_2} \backslash C_{D_1})$ des bases détectées où N^f est la segmentation finale du neurone (Cf. Section 3.2.1 et Figure 14d).

(a) Image initiale **(b)** N_f **(c)** C (en blanc) et $C_{D_2} \backslash C_{D_1}$ (en rouge) **(d)** N^f (en blanc) et B (en rouge)

FIGURE 14 – Les étapes de la détection des bases de dendrites. Pour cette illustration, les opérations ont été réalisées sur la projection du maximum d'intensité.

Section de la dendrite apicale

L'aire de la section de la dendrite apicale découle directement de la détection de la base des dendrites réalisée pour compter le nombre de dendrites. La dendrite apicale étant la dendrite de plus grande section, la composante connexe de B qui lui correspond est celle qui compte le plus grand nombre de voxels.

Distribution angulaire des dendrites secondaires

L'angle entre deux dendrites i et j est défini par l'angle (non orienté) entre les vecteurs $\overrightarrow{SD_i}$ et $\overrightarrow{SD_j}$, où S est le barycentre du corps cellulaire, et D_i et D_j sont les barycentres respectifs des bases des dendrites i et j.

À partir de cette définition, deux paramètres de distribution angulaire sont calculés :

— L'écart angulaire θ_{tot} entre la première et la dernière dendrite secondaires (dans l'ordre de parcours dans le sens horaire).

— Le paramètre $\frac{|\theta_{max} - \theta_{min}|}{\theta_{tot}}$ où θ_{max} et θ_{min} sont respectivement l'angle (non orienté) minimal et maximal entre deux dendrites secondaires voisines. Ce paramètre caractérise la régularité de la distribution des dendrites secondaires autour du corps cellulaire.

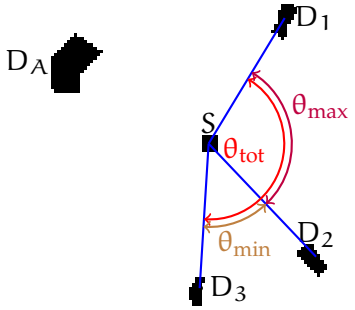


FIGURE 15 – Schéma de calcul des angles θ_{tot}, θ_{min} et θ_{max} à partir des résultats de la Figure 14d. S est le barycentre du corps cellulaire, D_A est la dendrite apicale, D_1, D_2 et D_3 sont les dendrites basales.

La Figure 15 illustre le calcul de ces valeurs sur les résultats de segmentation de la Figure 14.

3.3 ÉTUDE DES IMAGES X10

Vient ensuite l'étude des images x10, au sein desquelles l'intérêt est porté sur la ramification de la dendrite apicale, et plus particulièrement sur la distance entre le corps cellulaire du neurone et la première ramification [HATTOX et NELSON, 2007]. Le calcul de ce paramètre se fait par l'intermédiaire d'une étape de détection (Cf. Section 3.3.1) et d'une étape de reconstruction (Cf. Section 3.3.2) des neurones.

3.3.1 Détection des neurones

Une manière de segmenter le neurone serait d'utiliser des algorithmes de suivi de structures tubulaires [KIRBAS et F. QUEK, 2004 ; F. K. QUEK et KIRBAS, 2001 ; B. ZHANG et al., 2010], mais le fort bruit que présentent les images rendrait une telle approche inappropriée.

Or, comme le montre la Figure 11, les dendrites apicales vont du corps cellulaire (situé dans la couche V dans notre cas) jusqu'à la couche extérieure du cortex, en suivant une direction de propagation globalement commune. Nous proposons par conséquent de considé-

rer les coupes d'axe (y, z) de l'image, sur lesquelles apparaissent les sections des neurones. Il suffit donc de détecter ces intersections pour détecter les neurones.

Pour cela, nous utilisons un processus ponctuel marqué sur un ensemble de rectangles [Descombes, 2013]. Soit donc $r = \{r_i, i = 1, \ldots, n\}$ un ensemble de rectangles de taille arbitraire, et une fonction d'énergie définie par :

$$\mathcal{U}(r) = \sum_i \mathcal{U}_1(r_i) + \sum_{i,j} \mathcal{U}_2(r_i, r_j) \tag{3}$$

où $\mathcal{U}_2$ est un *a priori* de non recouvrement des rectangles (c'est-à-dire $\mathcal{U}_2(r_i, r_j) = \infty$ si r_i et r_j s'intersectent, et 0 sinon), et $\mathcal{U}_1$ est un terme d'attache aux données mesurant le contraste entre l'objet et le fond, défini par :

$$\mathcal{U}_1(u) = \left(\exp\left(-\frac{d_B(u) - d_0}{100}\right) - 1\right) \delta(d_B(u) \geqslant d_0) \tag{4}$$

où :

$$d_B(u) = \frac{(\mu_1 - \mu_2)^2}{4\sqrt{\sigma_1^2 + \sigma_2^2}} - \frac{1}{2}\log\frac{2\sigma_1\sigma_2}{\sigma_1^2 + \sigma_2^2} \tag{5}$$

et d_0 est un seuil sur le contraste minimal toléré, μ_1 (respectivement σ_1^2) et μ_2 (respectivement σ_2^2) sont les moyennes (respectivement les variances) des pixels à l'intérieur et sur le bord des rectangles.

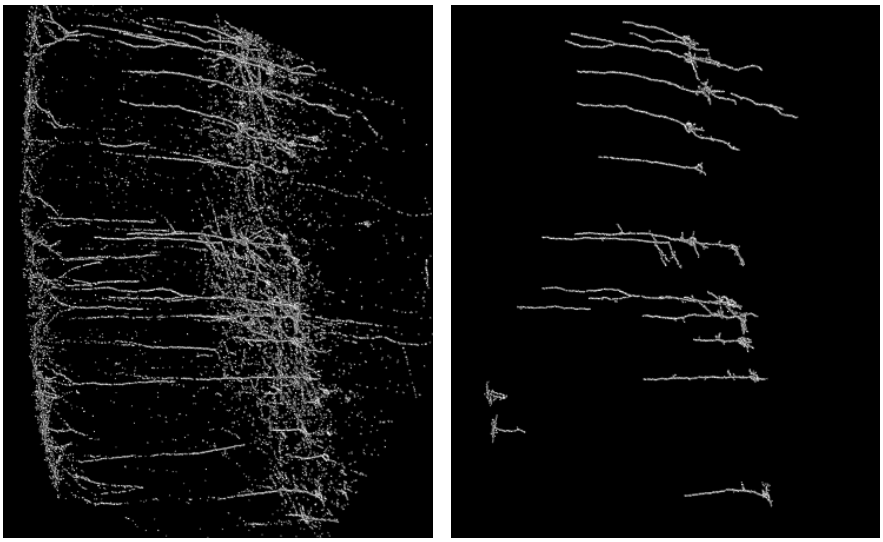
Pour minimiser l'énergie $\mathcal{U}$, nous utilisons l'algorithme de naissances et de coupes multiples (en anglais *Multiple Birth and Cut* – MBC) défini dans [Eldin et al., 2012]. Les résultats de la détection des sections est disponible en Figure 17. On constate que les sections sont pour la plupart convenablement détectées, et le fait que quelques sections ne soient pas détectées sera compensé pendant l'étape de reconstruction décrite en Section 3.3.2.

3.3.2 Reconstruction des neurones

Une fois les sections de neurones détectées, une étape de reconstruction du neurone est nécessaire afin d'obtenir la détection complète des neurones. Pour cela, nous construisons l'image binaire constituée des centres de l'ensemble des rectangles détectés en Section 3.3.1, à laquelle nous appliquons, successivement :

— une dilatation morphologique avec un élément structurant parallélépipèdique (plus long selon les axes x et y que selon z compte tenu du caractère anisotrope des images) [Meyer et Beucher, 1990],

— une suppression des composantes connexes de taille inférieure à un certain seuil, permettant de supprimer les surdétections de sections de neurones,



(a) Sections détectées (b) Reconstruction des neurones

FIGURE 16 – Les étapes de la reconstruction des neurones.

— une fermeture morphologique, afin de compléter les neurones et ainsi pallier les sous-détections évoquées en Section 3.3.1.

Les résultats de cette reconstruction sont donnés en Figure 16.

On peut constater que tous les neurones n'ont pas été segmentés entièrement. Ceci est dû à la fois au bruit présent dans les images et au fait que tous les neurones ne réagissent pas de manière égale à la protéine YFP (et par conséquent, certains neurones, et plus particulièrement leur dendrite apicale, apparaissent de manière très peu lumineuse, Cf. Section 3.1.3).

Il est néanmoins possible de calculer la distance entre le corps cellulaire et la première ramification de la dendrite apicale des neurones détectés. Pour cela, on procède en deux étapes :

— Détection des corps cellulaires des neurones (de manière analogue à la Section 3.2) et de l'éventuel point de branchement, obtenu par analyse du squelette du neurone [HOWE, 2013 ; TELEA et VAN WIJK, 2002],
— Calcul de la distance entre le corps cellulaire et le branchement comme étant le plus court chemin entre ces deux points d'intérêt du neurone [DIJKSTRA, 1959].

Des résultats de détection de corps cellulaire et de premier branchement sont donnés en Figure 17, sur laquelle le corps cellulaire des neurones et leur première ramification sont entourés en vert et en rouge, respectivement.

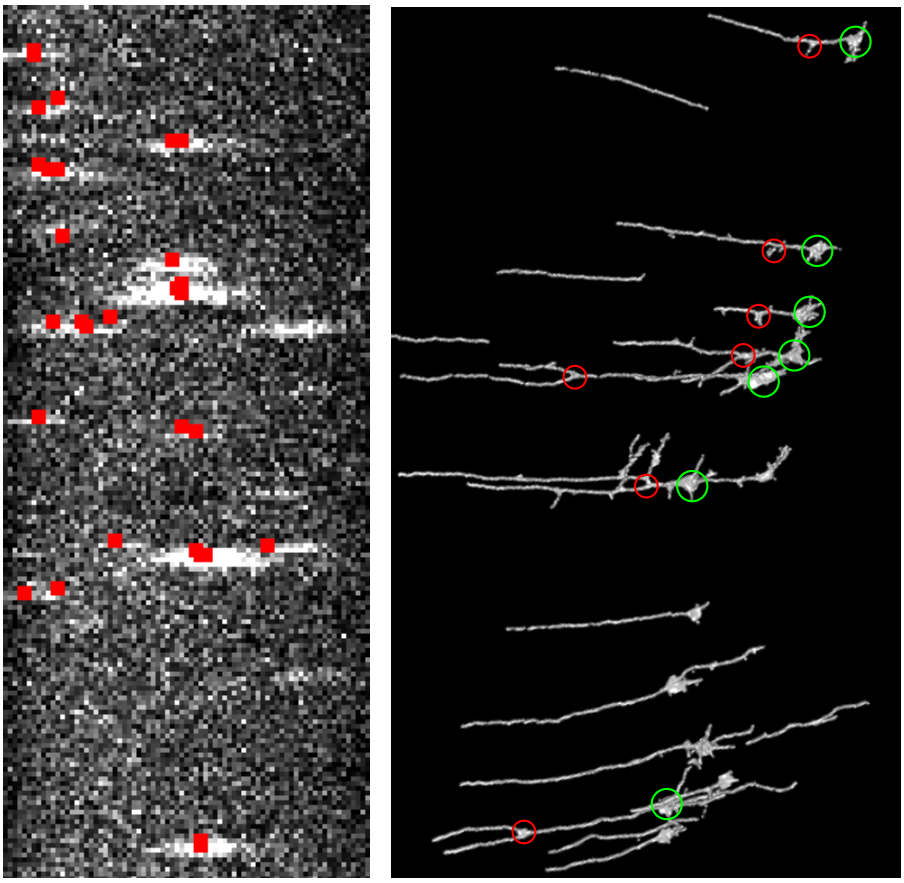


FIGURE 17 – Les deux étapes de la reconstruction de neurones : détection des sections de dendrites (à gauche) et reconstruction des dendrites (à droite).

3.4 MISE EN CORRESPONDANCE DES NEURONES

3.4.1 Définition du problème

Il s'agit d'établir une mise en correspondance entre les neurones détectés sur une image x40 et sur une image x10 afin d'en fusionner les descripteurs calculés en Sections 3.2 et 3.3.

Considérons donc une image x10 et une de ses images x40 correspondantes. Puisque l'image x40 n'est qu'un zoom d'une partie de l'image x10, tous les neurones de l'image x40 apparaissent sur l'image x10. En revanche, la réciproque est fausse. Il en découle que le nombre N de neurones sur l'image x10 est supérieur à n, le nombre de neurones sur l'image x40.

Pour combiner l'information issue des images x10 et x40, il suffit de savoir à quel neurone $\mathcal{N}_i^{10x}$ $(1 \leqslant i \leqslant N)$ de l'image x10 correspond un certain neurone $\mathcal{N}_j^{40x}$ $(1 \leqslant j \leqslant n)$ de l'image x40. En d'autres termes, nous cherchons une fonction de correspondance $\phi : [\![1, n]\!] \to [\![1, N]\!]$ qui, à l'indice d'un neurone de l'image x40, associe celui du neurone correspondant sur l'image x10.

3.4.2 Motivation

Pour répondre à la question énoncée précédemment, deux principales approches sont envisageables :

— une approche fondée sur des méthodes de recalage (rigide) d'images [Pratt, 1974 ; Zitova et Flusser, 2003],

— une approche fondée sur des outils de couplage de graphes bipartis [Duff et al., 2011 ; Kaya et al., 2011 ; Kuhn, 1955 ; Munkres, 1957].

Dans le cas d'images tridimensionnelles, le recalage d'images, cherchant par exemple à maximiser un critère de corrélation, peut s'avérer lourd en calculs. Par ailleurs, nous disposons déjà d'informations à propos des neurones détectés en Sections 3.2 et 3.3. La seconde approche semble alors plus appropriée pour réaliser cette opération d'appariement.

Pour calculer ϕ, nous proposons donc d'utiliser des outils d'appariement de graphes bipartis. Soit $\mathcal{N}^{10x} = \{\mathcal{N}_j^{10x}, 1 \leqslant j \leqslant N\}$ l'ensemble des neurones de l'image x10 et $\mathcal{N}^{40x} = \{\mathcal{N}_i^{40x}, 1 \leqslant i \leqslant n\}$ l'ensemble des neurones de l'image x40. Soit également un graphe biparti complet $\mathcal{G} = (\mathcal{N}^{40x}, \mathcal{N}^{10x}, \mathcal{E})$ où $\mathcal{E} = (e_{ij})_{(i,j)\in[\![1,n]\!]\times[\![1,N]\!]}$ est la matrice des arêtes pondérées entre les neurones de $\mathcal{N}^{40x}$ et ceux de $\mathcal{N}^{10x}$. Comment définir ces poids afin d'obtenir l'appariement de neurones souhaité ? Une réponse à cette question est proposée en Section 3.4.3.

3.4.3 Formalisation du problème : appariement d'ensembles de points de l'espace

Nous allons dès à présent formaliser le problème en considérant des points de l'espace[2]. Soit donc $\mathcal{X} = \{x_i, 1 \leqslant i \leqslant n\}$ et $\mathcal{Y} = \{y_j, 1 \leqslant j \leqslant N\}$ deux ensembles tels que $\mathcal{X} \subset \mathcal{Y}$ (et donc $n \leqslant N$) et munis d'une même distance d. Nous cherchons à trouver une fonction de correspondance $\phi : [\![1, n]\!] \to [\![1, N]\!]$ permettant de retrouver chaque point de $\mathcal{X}$ dans $\mathcal{Y}$. Pour cela, nous proposons de minimiser la fonction

$$\sum_{1\leqslant i_1<i_2\leqslant n} \left| d(x_{i_1}, x_{i_2}) - d(x_{\phi(i_1)}, x_{\phi(i_2)}) \right| .$$

Comme évoqué en Section 3.4.2, nous allons utiliser des outils d'appariement de graphes bipartis. Soit donc un graphe biparti complet $\mathcal{G} = (\mathcal{X}, \mathcal{Y}, \mathcal{E})$ où $\mathcal{E} = (e_{ij})_{(i,j)\in[\![1,n]\!]\times[\![1,N]\!]}$ est une matrice d'arêtes pondérées (que l'on va définir par la suite) entre les points de $\mathcal{X}$ et ceux de $\mathcal{Y}$. Nous allons proposer des valeurs de e_{ij} pour lesquelles ϕ résulte d'un appariement de $\mathcal{G}$.

Chaque $x_i \in \mathcal{X}$ (respectivement $y_j \in \mathcal{Y}$) peut être décrit par une liste de distances aux autres points de $\mathcal{X}$ (respectivement de $\mathcal{Y}$) $d_i^{\mathcal{X}} =$

2. Ces points correspondront en fait au barycentre des corps cellulaires segmentés sur les images x10 et x40 (Cf. Section 3.4.6).

$(d^{\mathcal{X}}_{ii'})_{1\leqslant i'\leqslant n} = (d(x_i, x_{i'}))_{1\leqslant i'\leqslant n}$ (respectivement $d^{\mathcal{Y}}_j = (d^{\mathcal{Y}}_{jj'})_{1\leqslant j'\leqslant N} = (d(y_j, y_{j'}))_{1\leqslant j'\leqslant N}$).

Soit $(x_i, y_j) \in \mathcal{X} \times \mathcal{Y}$. Si x_i et y_j se correspondent (c'est-à-dire si $\phi(i) = j$), alors leurs listes de distances respectives $d^{\mathcal{X}}_i$ et $d^{\mathcal{Y}}_j$ ont des valeurs communes (ou relativement proches). Nous proposons donc d'envisager un problème d'appariements des distances. Soit $M^{ij} = (m^{ij}_{i'j'})_{(i',j')\in[\![1,n]\!]\times[\![1,N]\!]}$ la matrice solution de ce problème d'appariement (c'est-à-dire pour laquelle $m^{ij}_{i'j'} = 1$ si $d^{\mathcal{X}}_{ii'}$ et $d^{\mathcal{Y}}_{jj'}$ se sont égales ou très proches et 0 sinon). On peut alors définir le poids e_{ij} entre x_i et y_j comme étant le coût optimal du problème d'optimisation linéaire suivant :

$$e_{ij} = \min_{M^{ij}\in\mathcal{M}_{n,N}(\{0,1\})} \sum_{i'=1}^{n}\sum_{j'=1}^{N} \left| d^{\mathcal{X}}_{ii'} - d^{\mathcal{Y}}_{jj'} \right| m^{ij}_{i'j'} \tag{P^{ij}}$$

$$\text{avec} \qquad \forall i' \in [\![1,n]\!], \sum_{j'=1}^{N} m^{ij}_{i'j'} = 1 \tag{6}$$

$$\text{et} \qquad \forall j' \in [\![1,N]\!], \sum_{i'=1}^{n} m^{ij}_{i'j'} \leqslant 1 \tag{7}$$

où :

— (6) signifie que chaque distance de $d^{\mathcal{X}}_i$ correspond exactement à une distance de $d^{\mathcal{Y}}_j$,
— (7) signifie que chaque distance de $d^{\mathcal{Y}}_j$ correspond à au plus une distance de $d^{\mathcal{X}}_i$,
— (P^{ij}) signifie que l'ont souhaite minimiser la somme des différences entre les distances correspondantes.

La résolution de ces problèmes d'optimisation (P^{ij}) (pour $1 \leqslant i \leqslant n$ et $1 \leqslant j \leqslant N$) sera détaillée en Section 3.4.4.

Si e_{ij} est proche de 0, cela veut dire que les ensembles de distances $d^{\mathcal{X}}_i$ et $d^{\mathcal{Y}}_j$ se correspondent et que l'hypothèse $j = \phi(i)$ est raisonnable. Sinon, cela peut signifier qu'il existe au moins une distance de $d^{\mathcal{X}}_i$ qui n'apparaît pas dans $d^{\mathcal{Y}}_j$, ce qui impliquerait $j \neq \phi(i)$.

Une fois que tous les e_{ij} sont calculés, l'ensemble des arêtes $\mathcal{E}$ est défini et la correspondance ϕ est le résultat de l'appariement du graphe biparti $\mathcal{G}$, consistant à trouver la matrice d'appariement $\mathcal{A} = (a_{ij})_{(i,j)\in[\![1,n]\!]\times[\![1,N]\!]}$ solution du problème d'optimisation linéaire :

$$\min_{A \in \mathcal{M}_{n,N}(\{0,1\})} \sum_{i'=1}^{n} \sum_{j'=1}^{N} e_{ij} a_{ij} \tag{Q}$$

$$\text{avec} \qquad \forall i' \in [\![1, n]\!], \sum_{j'=1}^{N} m_{i'j'}^{ij} = 1 \tag{8}$$

$$\text{et} \qquad \forall j' \in [\![1, N]\!], \sum_{i'=1}^{n} m_{i'j'}^{ij} \leqslant 1 \tag{9}$$

où :

— (8) signifie que chaque point x_i de $\mathcal{X}$ correspond exactement à un point y_j de $\mathcal{Y}$,
— (9) signifie que chaque point y_j de $\mathcal{Y}$ correspond à au plus un point x_i de $\mathcal{X}$,
— (Q) signifie que l'ont souhaite minimiser la somme des coûts e_{ij} choisis.

La résolution de ce problème d'optimisation (Q) sera détaillée en Section 3.4.4.

Enfin, la fonction de correspondance ϕ estimée découle directement de la matrice $\mathcal{A}$ calculée : $\phi(i) = j$ si, et seulement si, $a_{ij} = 1$. ϕ est bien définie pour tout $i \in [\![1, n]\!]$ en vertu de la condition (8).

3.4.4 Résolution des problèmes d'optimisation linéaire

Les problèmes (P^{ij}) (pour $1 \leqslant i \leqslant n$ et $1 \leqslant j \leqslant N$) et (Q) sont des problèmes d'optimisation linéaire liés à un problème de couplage de graphe biparti.

Parmi toutes les méthodes proposées dans la littérature pour résoudre cette famille de problèmes [Duff et al., 2011 ; Kaya et al., 2011, 2013 ; Kuhn, 1955 ; Munkres, 1957], l'algorithme le plus couramment utilisé est l'algorithme hongrois [Kuhn, 1955] (garantissant une résolution du problème en temps polynômial). En revanche, cet algorithme est initialement conçu pour le cas équilibré (c'est-à-dire dans le cas où $\mathcal{X}$ et $\mathcal{Y}$ sont de même cardinal), ce qui n'est pas le cas de notre application pour laquelle $n \leqslant N$.

Il existe une extension naturelle de l'algorithme hongrois au cas non équilibré ($n < N$), pour laquelle la matrice d'adjacence du graphe biparti est rendue carrée par ajout de valeurs arbitrairement grandes (strictement supérieures à sa valeur maximale, par exemple). Ce procédé fait donc passer d'un problème de taille $n \times N$ à un problème de taille N^2, ce qui peut engendrer un temps de calcul bien plus conséquent lorsque $n \ll N$.

Il est par ailleurs possible de résoudre de tels problèmes d'appariement de graphes en ayant recours à des outils d'optimisation linéaire en variables entières. En effet, les problèmes (P^{ij}) et (Q) peuvent être

réécrits sous forme d'un problème d'optimisation linéaire à contraintes linéaires standard :

$$\begin{array}{lll} \inf\limits_{x} & c^T x & \\ \text{avec} & Ax & = b, \\ \text{et} & x & \geqslant 0, \end{array}$$

et ainsi être résolus grâce à des algorithmes de programmation linéaire [Dantzig, 1998 ; Hadley, 1962] tels que le simplexe [Nelder et Mead, 1965] ou la méthode des points intérieurs [Nesterov et al., 1994].

Bien que la méthode des points intérieurs soit généralement plus rapide, l'algorithme du simplexe est à préférer dans ce cas de figure dans la mesure où il garantit l'obtention d'une solution exacte au problème d'optimisation [Schrijver, 2003].

L'impact du choix de l'approche pour résoudre le problème d'appariement sur le temps de calcul sera détaillé en Section 3.4.6.

3.4.5 Cas d'ambiguïté

Dans la Section 3.4.3, nous partons du principe que si $x_i \in \mathcal{X}$ et $y_j \in \mathcal{Y}$ se correspondent, alors nous retrouverons dans $d_j^{\mathcal{Y}}$ des valeurs proches de celles de $d_i^{\mathcal{X}}$. La réciproque est fausse, notamment dans le cas où le motif $\mathcal{X}$ est répété dans $\mathcal{Y}$.

Cependant, c'est un phénomène rare (et d'autant plus rare que n est grand), et quand bien même il se produirait, la méthode proposée est capable de le détecter. En effet, dans un tel cas, il existerait $m \geqslant 2$ motifs $\mathcal{X}_1, \cdots, \mathcal{X}_m$ de même cardinal pour lesquels $d_{i_1}^{\mathcal{X}}, \cdots, d_{i_m}^{\mathcal{X}}$ ont des valeurs similaires à celles de $d_j^{\mathcal{Y}}$ pour un certain $(i_1, \cdots, i_m, j) \in [\![1, n]\!]^m \times [\![1, N]\!]$, si bien que les pénalités respectives $e_{i_1 j}, \cdots, e_{i_m j}$ seraient faibles. On en déduirait alors les ensembles $\mathcal{X}_1, \cdots, \mathcal{X}_m$ des matrices $M^{i_1 j}, \cdots, M^{i_m j}$ des problèmes $P^{i_1 j}, \cdots, P^{i_m j}$.

Dans un tel cas de figure, l'algorithme proposé permet alors d'extraire des appariements possibles de points. L'appariement final peut donc être choisi *a posteriori* parmi les candidats donnés par la méthode, par exemple en cherchant à maximiser un critère de corrélation [Pratt, 1974] parmi les appariements proposés.

3.4.6 Appariement de données synthétiques

Avant d'appliquer la méthode d'appariement aux neurones (Cf. Section 3.4.7), commençons par en juger l'efficacité, en termes de temps de calcul et de robustesse au bruit, sur des données générées aléatoirement.

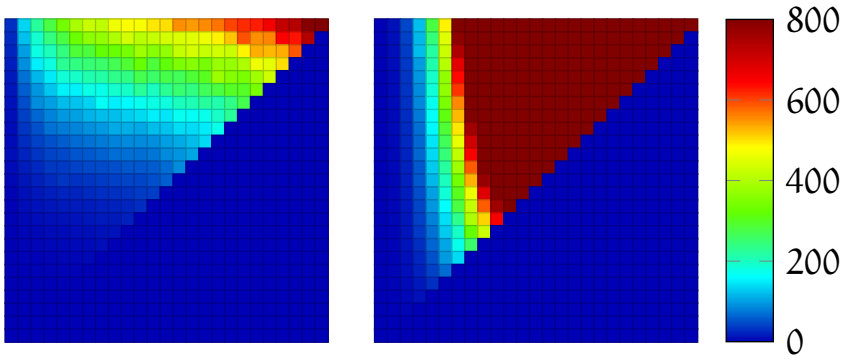


FIGURE 18 – Temps de calcul (en secondes) de la résolution du problème d'appariment pour $1 \leqslant n \leqslant N \leqslant 100$ pour l'algorithme hongrois (à gauche), et pour le simplexe (à droite).

Temps de calcul

Nous générons un ensemble $\mathcal{X}$ de n points en dimension 2 dont chacune des composantes suit une loi uniforme sur $[0, 1]$, puis un ensemble $\mathcal{Y}$ composé des points de $\mathcal{X}$ et de $N - n$ autres points eux aussi générés selon une loi uniforme sur $[0, 1]^2$. Nous comparons dans la Figure 18 les temps de calculs mis par la méthode en ayant recours à l'algorithme hongrois d'un côté, et au simplexe de l'autre.

Pour chaque graphique de la Figure 18, l'axe des abscisses correspond à n et l'axe des ordonnées à N. La partie triangulaire inférieure droite est forcée à zéro parce qu'elle correspond au cas où $n > N$. Nous avons imposé la valeur maximale du temps de calcul à 800 secondes pour que le graphe correspondant à l'algorithme hongrois soit lisible, ce qui explique la saturation du graphe correspondant au simplexe. En réalité, le simplexe met plusieurs heures pour le cas $n = N = 100$.

Comme annoncé en Section 3.4.4, le simplexe devient plus intéressant lorsque la différence entre n et N est importante ($n \leqslant N/5$ sur la plage de données testée). L'algorithme hongrois, quant à lui, l'emporte largement lorsque n s'approche de N, et correspond donc à un choix plus sûr dans la plupart des cas.

Robustesse au bruit

Nous générons d'abord les ensembles $\mathcal{X}$ et $\mathcal{Y}$ de la même manière que précédemment. Nous fixons $N = 100$ et considérons plusieurs valeurs de n : $\{5, 10, 15, 20\}$. Nous appliquons ensuite un bruit additif de loi uniforme sur $[-\epsilon, +\epsilon]$ (avec $\epsilon \in \left[0, 10^{-1}\right]$) sur chaque composante des n premiers vecteurs de $\mathcal{Y}$ (c'est-à-dire celles correspondant aux points de $\mathcal{X}$). Nous souhaitons observer la robustesse de la méthode proposée en suivant l'évolution du taux de réussite de l'algorithme en fonction du bruit ϵ. Le taux de réussite est simplement défini comme étant la moyenne des rapports entre le nombre de points de $\mathcal{X}$ correc-

tement appariés avec leurs correspondants dans $\mathcal{Y}$ et n sur 50 réalisations de bruit. Les résultats sont donnés dans la Figure 19.

On constate que la robustesse de la méthode augmente avec n, ce qui est dû au fait que plus n est grand plus le motif $\mathcal{X}$ sera difficile à reproduire dans $\mathcal{Y}$.

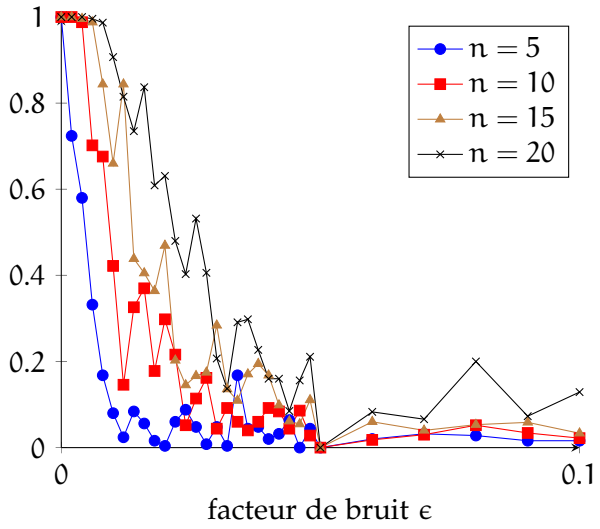


FIGURE 19 – Taux de réussite en fonction du facteur de bruit ϵ, pour $N = 100$ et différentes valeurs de n.

3.4.7 Application à la correspondance de neurones

Il faut représenter les neurones détectés en Sections 3.2 et 3.3 par des points pour pouvoir appliquer la méthode proposée. Pour cela, nous proposons d'assimiler chaque neurone au barycentre de son corps cellulaire, dont une segmentation a déjà été calculée.

Le résultat d'appariement est illustré par la figure 20. Il est important de noter que, dans cet exemple, toutes les distances sont exprimées en μm et non pas en pixels, puisque l'échelle est différente entre les deux types d'images.

Le calcul du barycentre du corps cellulaire est éventuellement à l'origine d'un bruit entre les deux types d'image, mais celui-ci est suffisamment faible dans la mesure où tous les neurones ont été correctement appariés.

3.5 INTERPRÉTATION DES RÉSULTATS

3.5.1 Histogrammes des distributions

Au total, 317 neurones ont été extraits sur l'ensemble des images mises à disposition. Les descripteurs donnés en section 3.2 ont ensuite

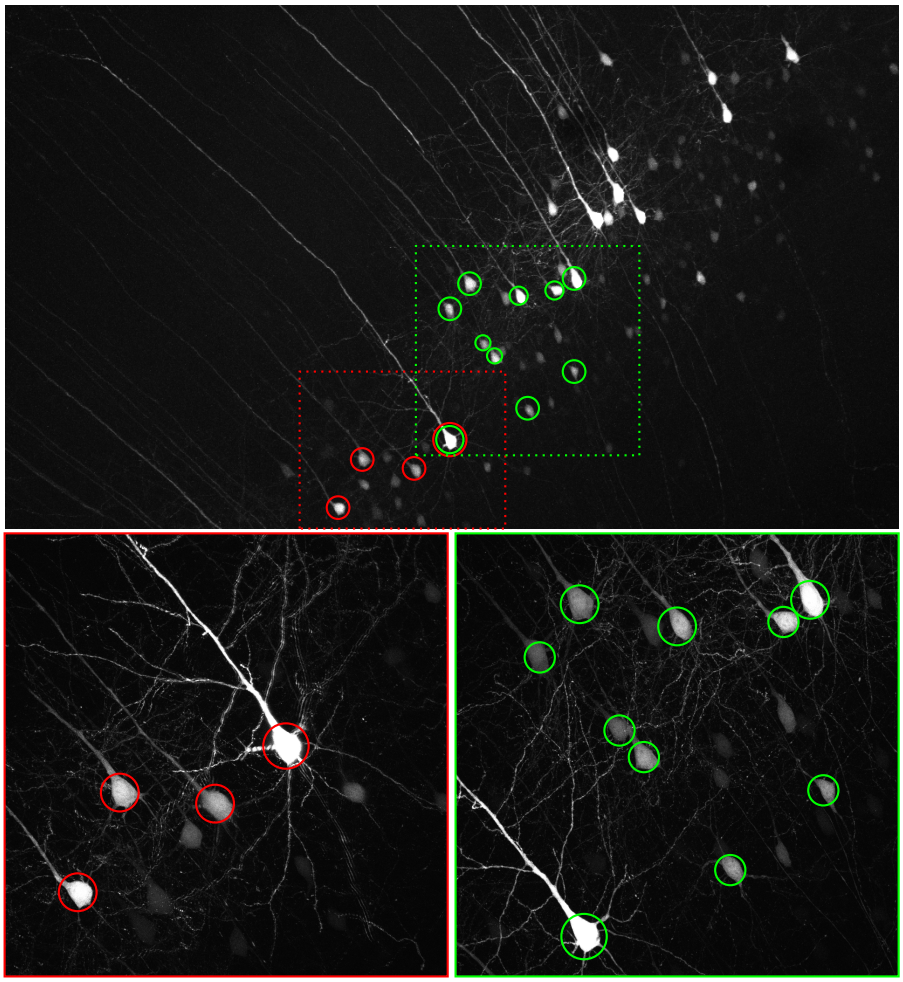
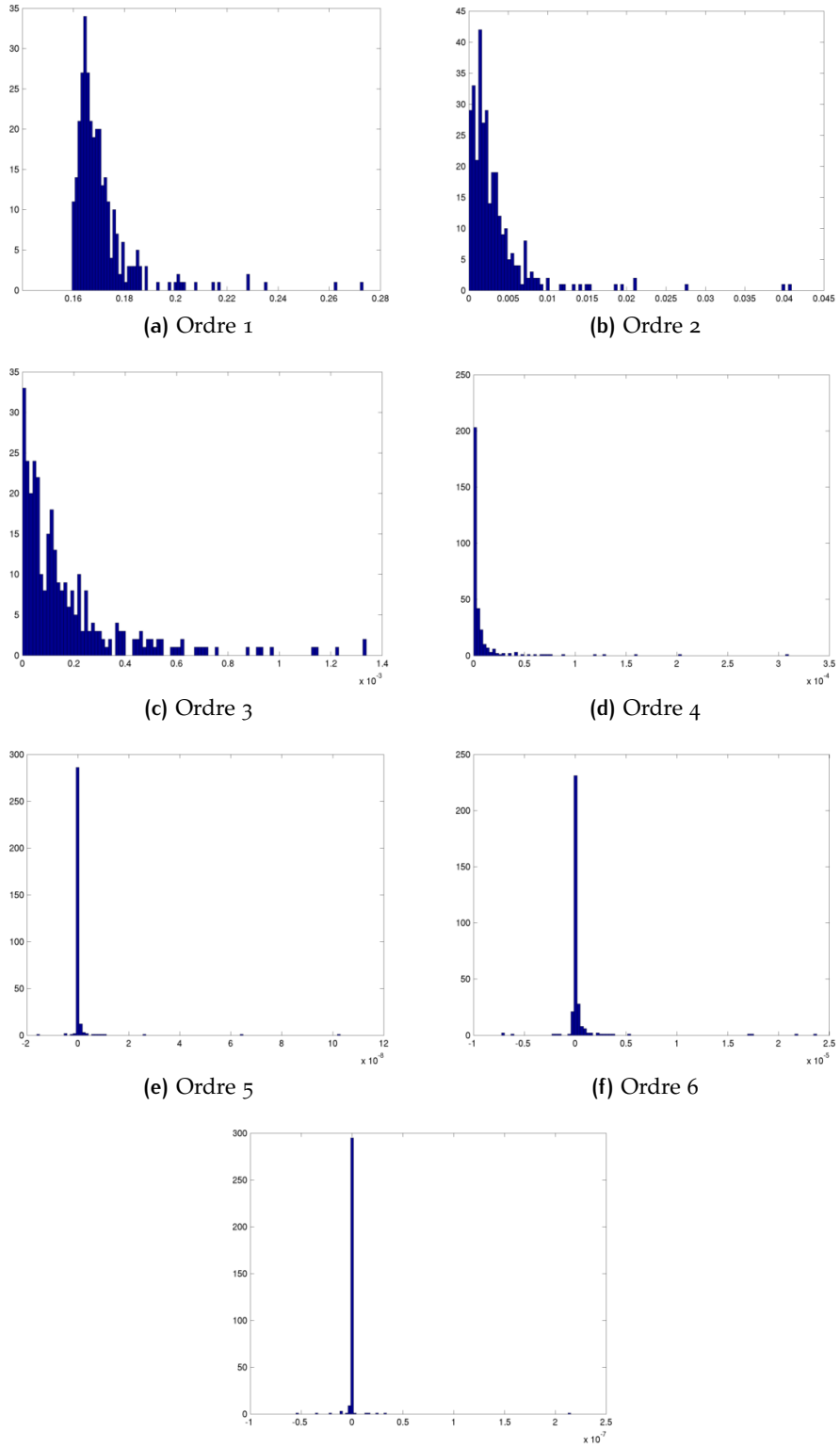


FIGURE 20 – Illustration de la carte de correspondance sur la projection du maximum d'intensité d'images. Les têtes des neurones correspondants sont entourées de la même couleur sur l'image 10x (en haut) que sur les images 40x (en bas).

été calculés pour chacun d'entre eux. Les Figures 21 et 22 montrent les histogrammes des différentes caractéristiques obtenues.

Pour ce qui est des descripteurs de forme décrits en Section 3.2.2, on constate sur la Figure 21 qu'à partir de l'ordre 5, les moments extraits semblent relativement peu discriminants sur l'échantillon observé.

Il est important de noter que, pour les Figures 22b, 22c, 22f, si de nombreuses valeurs sont à zéro, c'est parce qu'elles ont été forcées à zéro puisqu'aucune dendrite basale n'a pu être détectée (soit parce qu'elles n'existent pas, soit parce que l'algorithme n'a pas pu les segmenter). En fait, le tableau de descripteurs obtenu est relativement creux, c'est-à-dire que certaines valeurs sont manquantes.



(a) Ordre 1

(b) Ordre 2

(c) Ordre 3

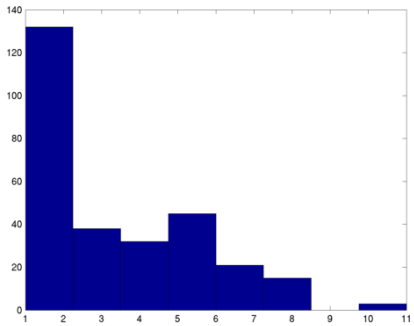
(d) Ordre 4

(e) Ordre 5

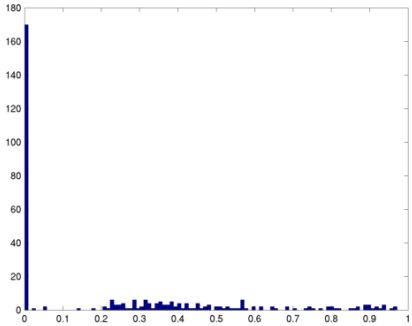
(f) Ordre 6

(g) Ordre 7

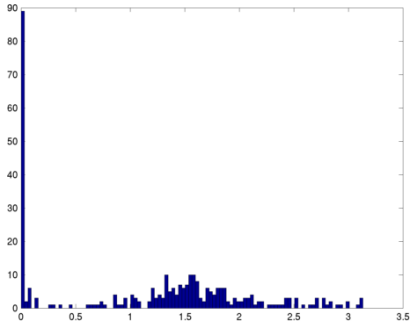
FIGURE 21 – Histogrammes de répartition des moments de forme [HU, 1962] pour l'ensemble des neurones segmentés.



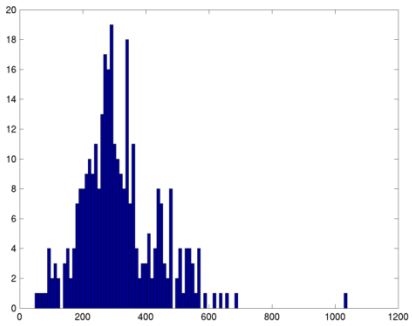
(a) Nombre de dendrites



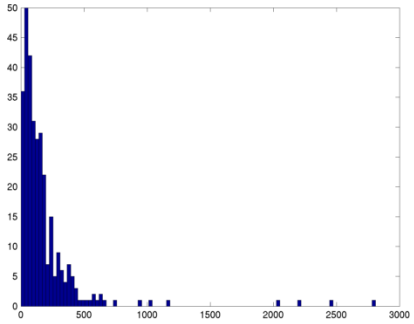
(b) θ_{tot}



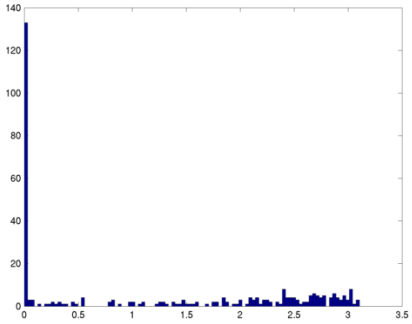
(c) $\frac{|\theta_{max} - \theta_{min}}{\theta_{tot}}$



(d) Taille du corps cellulaire



(e) Épaisseur de la dendrite apicale



(f) Épaisseur moyenne des dendrites secondaires

FIGURE 22 – Histogrammes de répartition des différents descripteurs calculés.

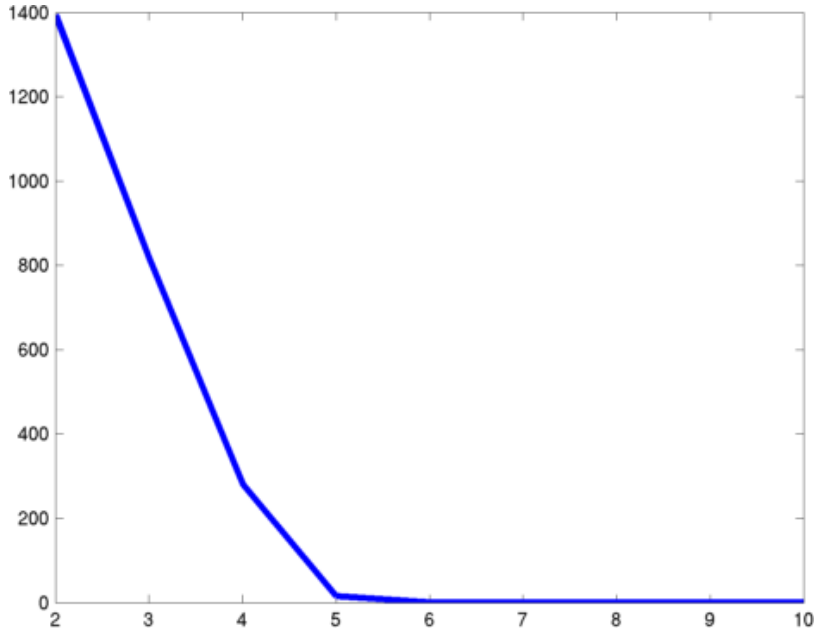


FIGURE 23 – Évolution de la somme des distances intra-cluster (c'est-à-dire des points au barycentre du cluster) en fonction du nombre de classes k donné à l'algorithme des k-moyennes.

3.5.2 Classification par k-moyennes

Afin de voir si les paramètres calculés permettent de distinguer des classes de neurones, l'algorithme des k-moyennes (décrit en Section 4.2.1) a été appliqué à l'ensemble des vecteurs extraits [3], pour des valeurs de k comprises entre 2 et 10.

La Figure 23 montre l'évolution de la somme des distances entre les points et le barycentre de leur classe, en fonction du nombre de classes k. On remarque que cette valeur décroît très rapidement jusqu'à une valeur proche de 0 pour $k \leqslant 5$, puis reste proche de 0 au-délà. Il semble donc possible de dégager jusqu'à 5 différentes classes de neurones parmi l'ensemble des neurones observés.

3.6 CONCLUSION ET PERSPECTIVES

Dans ce chapitre, nous proposons une méthode permettant d'extraire des caractéristiques décrivant des neurones d'images de sources différentes en s'intéressant aux deux types d'images donnés, puis en en combinant les données extraites.

Comme dans [NAM et al., 2014], la méthode d'appariement proposée n'utilise qu'une information ponctuelle. Elle a cependant l'avantage d'utiliser un critère de distance entre paires de points et évite

3. Les vecteurs ont été centrés et réduits au préalable afin de ne pas biaiser l'algorithme des k-moyennes avec des données exprimées dans des unités différentes.

donc d'avoir à minimiser un critère de corrélation entre pixels (ou voxels). Des algorithmes d'appariements de points autres que l'algorithme hongrois [Aiger et Kedem, 2009, 2010]pourraient être testés et comparés en terme de performance et de robustesse.

Il serait néanmoins intéressant d'envisager une généralisation de cet algorithme d'appariement au cas où il existe un facteur d'échelle λ inconnu entre les deux types d'image. Le problème d'optimisation à résoudre serait donc de la forme :

$$e_{ij} = \min_{M^{ij} \in \mathcal{M}_{n,N}(\{0,1\}),\ \lambda > 0} \sum_{i'=1}^{n} \sum_{j'=1}^{N} \left| d^{\mathcal{X}}_{ii'} - \lambda d^{\mathcal{Y}}_{jj'} \right| m^{ij}_{i'j'} \tag{P'^{ij}}$$

$$\text{avec} \qquad \forall i' \in [\![1, n]\!],\ \sum_{j'=1}^{N} m^{ij}_{i'j'} = 1 \tag{10}$$

$$\text{et} \qquad \forall j' \in [\![1, N]\!],\ \sum_{i'=1}^{n} m^{ij}_{i'j'} \leqslant 1 \tag{11}$$

Enfin, il est envisageable d'appliquer la chaîne de traitement proposée à un nombre plus important d'images. Cela permettrait d'obtenir des résultats plus significatifs, mais aussi à des images provenant de souris mutantes et atteintes d'une pathologie cérébrale pour voir si la mutation a un impact sur la morphologie des neurones.

4 ANALYSE DE L'ARCHITECTURE D'UN CARCINOME RÉNAL

Sommaire

4.1 Introduction 55
- 4.1.1 Contexte biomédical : le carcinome à cellules rénales 55
- 4.1.2 Présentation des données 56
- 4.1.3 Motivation 57
- 4.1.4 Défis et enjeux 59

4.2 Images à grossissement minimal 59
- 4.2.1 Segmentation du tissu tumoral 60
- 4.2.2 Résultats de segmentation 62
- 4.2.3 Découpage de la région d'intérêt 63

4.3 Analyse de l'architecture 65
- 4.3.1 Pré-traitements 65
- 4.3.2 Extraction du réseau vasculaire 71
- 4.3.3 Graphe représentant le réseau vasculaire 75
- 4.3.4 Combinaison d'information : Fusion de graphes voisins 77

4.4 Analyse du graphe 79
- 4.4.1 Extraction des données 79
- 4.4.2 Interprétations biomédicales des données 79

4.5 Conclusion 80
- 4.5.1 Chaîne de traitement proposée 80
- 4.5.2 Temps de calcul 80
- 4.5.3 Perspectives 82

4.1 INTRODUCTION

4.1.1 Contexte biomédical : le carcinome à cellules rénales

Le *carcinome à cellules rénales* (*renal cell carcinoma* – *RCC*) représente environ 90% des cancers du rein et compte plusieurs centaines de milliers de nouveaux cas chaque année dans le monde [Jonasch et al., 2014]. Les RCC sont subdivisés en plusieurs sous types, les *carcinomes rénaux à cellules claires* (*clear cell RCC* – *CCRCC*) étant les plus fréquents.

Dans certaines situations, une intervention chirurgicale (*néphrectomie* partielle ou totale) à but curatif est nécessaire. Cette opération

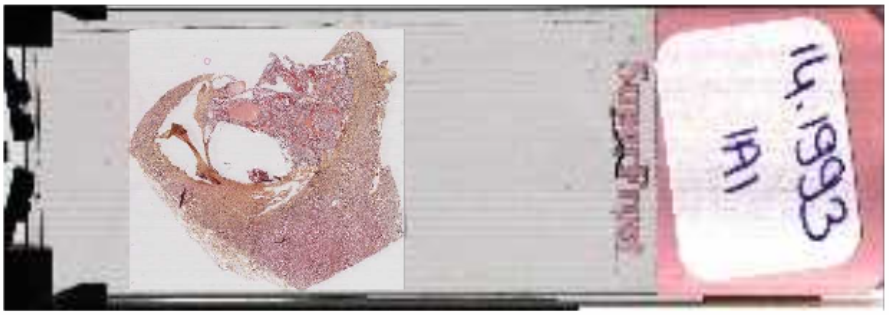


FIGURE 24 – Un exemple de lamelle scannée.

consiste à ouvrir la paroi de l'abdomen afin d'extraire la tumeur (ainsi qu'une partie voire la totalité du rein). Une fois enlevée, la tumeur est ensuite analysée dans l'objectif de fournir un diagnostic et donc un pronostic précis.

Le diagnostic consiste à déterminer le type histologique et l'avancement de la tumeur. Il résulte de l'analyse de 4 critères :

— La morphologie cellulaire du tissu tumoral [CHEVILLE et al., 2003],
— L'architecture, correspondant principalement à l'organisation vasculaire du tissu tumoral [SABO et al., 2001],
— L'expression protéique [SARTO et al., 1997],
— L'expression génétique [GOLUB et al., 1999 ; HIGGINS et al., 2003 ; M. TAKAHASHI et al., 2001]

Ces analyses permettent de définir le sous-type histologique de la tumeur [STÖRKEL et al., 1997]. Cette classification permet ainsi aux médecins d'établir le pronostic vital [KEEGAN et al., 2012] et de proposer un traitement adapté au patient.

En 2004, l'*Organisation Mondiale de la Santé* (*OMS*) recensait 12 types de tumeurs à cellules rénales [LOPEZ-BELTRAN et al., 2006]. Seulement 2 d'entre elles sont bénignes (*adénome papillaire* et *oncocytome*) et les 10 autres malignes. Parmi les types de tumeurs malignes, le plus courant est le *carcinome à cellules claires* (75% des cas), le *carcinome papillaire* (10%), et le *carcinome chromophobe* (5%). Il existe également le type *carcinome inclassé* (4 à 6%), correspondant aux tumeurs ne pouvant pas être placées dans une des 9 autres catégories malignes, ce qui laisse penser que cette classification risque de s'étoffer à l'avenir. Les autres types sont, quand à eux, d'incidence rare (inférieure à 1%).

4.1.2 Présentation des données

Le tissu extrait lors de l'exérèse chirurgicale est pris en charge par l'anatomopathologiste. Son analyse consiste d'abord, en un temps macroscopique, à découper le prélèvement en tranches qui feront par la suite l'objet de coupe micrométrique (d'une épaisseur de 2 à 5 microns) et seront placées sur une lame en vue d'une observation *a poste-*

riori. Pour que les détails structurels soient visibles, le tissu est coloré. En histologie, les colorants les plus souvent utilisés sont l'éosine, un acide qui colore en rose le cytoplasme et la matrice cellulaire, et l'hématoxyline, une base qui colore en bleu-noir les noyaux [Bancroft et Gamble, 2008] (Cf. Figure 28b). L'observation du tissu se fait alors soit au microscope, soit sur ordinateur, une fois la lame numérisée.

Nous disposons d'une série de 12 lames numérisées par un scanner Leica SCN400 (Cf. Figure 24). L'image résultante est compressée et enregistrée au format .scn, un format de *Whole-Slide Imaging* [Ho et al., 2006] permettant d'observer le tissu à plusieurs niveaux de grossissements (Cf. Figure 25).

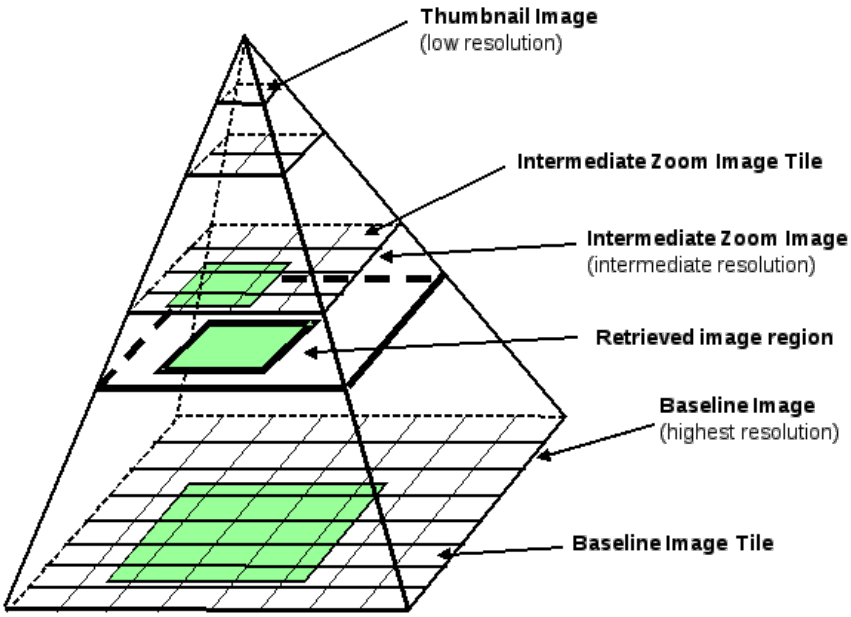
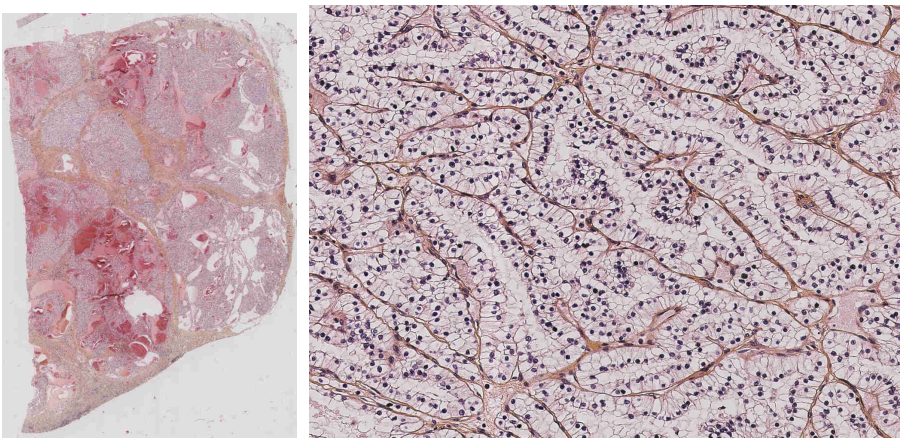


Figure 25 – La structure pyramidale du *Whole-Slide Imaging*.

Le premier niveau de grossissement permet d'observer le tissu dans son ensemble à basse résolution et donc de distinguer les différentes régions composant l'échantillon (tissus sain, tumoral, ou adipeux, nécrose, sang, ... Cf. Figure 26a), tandis que grossissement le plus fort (résolution de 0.25µm) rend possible l'observation du tissu à l'échelle cellulaire (Cf. Figure 26b). À un tel grossissement, les images comptent un total d'environ 100 000 pixels par axe.

4.1.3 Motivation

L'*histopathologie digitale* est une technique relativement récente et en plein essor, à l'intersection entre l'histopathologie et l'imagerie médicale, apportant de nouvelles problématiques en traitement de l'image (Cf. Section 4.1.4). Plus spécifiquement, concernant les tumeurs rénales, un grand nombre d'applications retrouvées dans la



(a) Premier niveau (b) Dernier niveau

FIGURE 26 – Exemple d'images prises aux niveaux de grossissement minimal et maximal.

littérature s'intéresse à la segmentation des différentes parties de la tumeur [HUANG et al., 2011 ; NAYAK et al., 2013] ou encore à l'analyse de la morphologie cellulaire [IRSHAD et al., 2014 ; YEH et al., 2014].

Or, comme évoqué en Section 4.1.1, la morphologie cellulaire n'est pas le seul critère morphologique permettant d'établir le diagnostic. En effet, lors de la formation de la tumeur, les cellules se disposent le long d'un réseau vasculaire (visible en marron sur la Figure 26b). On parle alors d'*angiogénèse*. Ce réseau, définissant en partie l'architecture de la tumeur, s'organise différemment en fonction du type histologique, ce qui en fait un critère capital du diagnostic [SABO et al., 2001]. Intuitivement, plus le réseau est régulier, plus la tumeur se développe de façon structurée et, par conséquent, c'est un argument en faveur de la bégninité ou d'un faible grade. Inversement, plus le réseau est chaotique et complexe, plus elle se développe de manière agressive et plus la tumeur risque d'être maligne.

Le but du travail proposé est donc l'analyse de la morphologie du réseau vasculaire à grande échelle, piste relativement peu explorée jusqu'à présent, puis l'étude de l'éventuel lien qui peut exister avec le type histologique de la tumeur. De plus, son originalité repose sur une analyse des images telles qu'elles sont "vues" par le pathologiste, sans coloration préalable pouvant mettre en évidence le réseau vasculaire. L'approche proposée se découpe alors en 3 parties :

— Représentation du réseau vasculaire sous forme de graphe (Sections 4.2 et 4.3),
— Étude du graphe (Section 4.4),
— Mise en correspondance avec le type histologique (Section 4.4.2).

4.1.4 Défis et enjeux

L'analyse d'image histologiques pose plusieurs difficultés. Premièrement, puisque seules les zones tumorales sont intéressantes, il est nécessaire de segmenter au préalable ces régions d'intérêt (Cf. Section 4.2). Cette segmentation peut être réalisée à n'importe quel niveau de grossissement, mais en raison de la taille des images au grossissement maximal, un compromis raisonnable entre temps de calcul et qualité de segmentation est à trouver.

Par ailleurs, comme évoqué en Section 4.1.2, les images à fort grossissement sont de très grande taille, ce qui impose, dans un premier temps, un découpage de l'image entière en sous-images sur lesquelles on extrait des informations locales, et, dans un deuxième temps, une fusion globale des données. Il s'agit très clairement d'un problème qui fait appel à une étude multi-échelle, d'autant que c'est l'approche adoptée par les anatomopathologistes, qui alternent entre les différents niveaux de grossissement lors de l'analyse de tels tissus.

Ensuite, certaines images présentent des défauts d'acquisition suite à la numérisation de la lame. Il arrive que certaines bandes de l'image soient floues (à cause d'une erreur lors de la mise au point automatique du capteur), comme le montre la Figure 27. Dans un tel cas de figure, il devient difficile, même à l'œil nu, de distinguer les différentes parties de l'image. En pratique, ces zones sont tout simplement ignorées pour deux raisons : l'information s'y trouvant est trop peu fiable et, en raison de la taille des images, il y a des zones suffisamment vastes à explorer, même en ignorant les zones où l'acquisition d'image est d'une qualité insatisfaisante.

Enfin, les images mises à disposition sont relativement hétérogènes. On observe effectivement une variabilité inter-images (deux lames présentent des tissus de morphologie et coloration différentes, Cf. Figure 28a), ainsi qu'une variabilité intra-image (densité du réseau variable, différents composants du tissu, ... Cf. Figure 28b).

4.2 IMAGES À GROSSISSEMENT MINIMAL

Avant de s'intéresser à l'étude du réseau vasculaire à proprement parler, une première étape de segmentation du tissu tumoral est à effectuer (Section 4.2.1). En effet, certaines zones de l'échantillon (masse grasse, sang, ...) n'ont aucun intérêt dans l'analyse de l'angiogénèse et n'apporteraient aucune information pertinente. Cette segmentation permettra ensuite de procéder au découpage des régions d'intérêt en sous-images de taille suffisamment petite 4.2.3).

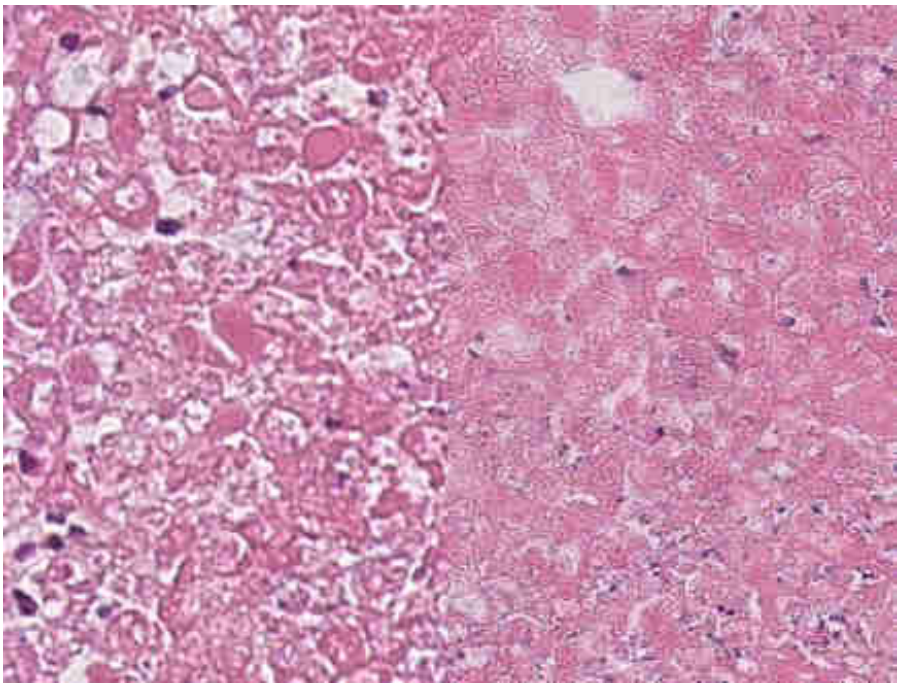


FIGURE 27 – Bande floue apparaissant dans l'image.

4.2.1 Segmentation du tissu tumoral

La segmentation du tissu tumoral est une voie légèrement explorée en traitement d'images histologiques. Parmi les approches proposées, la plupart s'orientent vers l'apprentissage supervisé [APOU et al., 2014 ; C.-W. WANG et al., 2011]. Cela demande donc l'intervention préalable d'un expert pour annoter manuellement certains pixels (ou certaines régions) de l'image pour constituer un ensemble d'apprentissage sur lequel l'algorithme de classification peut s'appuyer.

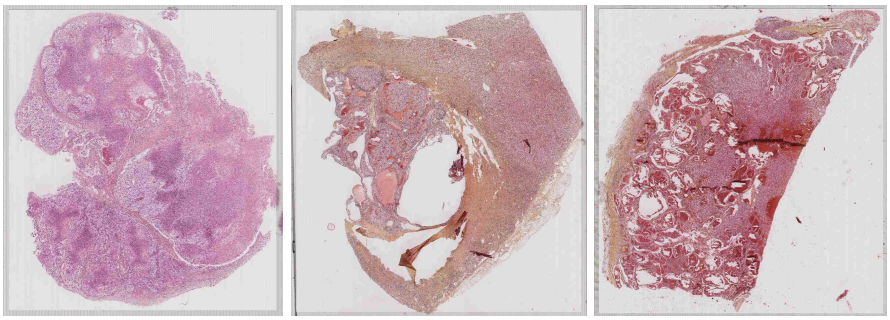
Nous proposons d'aborder la piste de l'apprentissage non-supervisé (ou *clustering*) dans le but d'automatiser la détection de zones d'intérêt. Pour cela, nous partons du constat que les zones tumorales sont plus foncées et plus hétérogènes (en raison de la présence des noyaux, apparaissant en bleu-noir grâce à la coloration à l'hématoxyline).

La notion d'hétérogénéité peut être quantifiée par l'entropie locale d'une image [GONZALEZ et al., 2004] : on peut définir l'entropie $E(p)$ d'un pixel p sur un voisinage $\mathcal{V}(p)$ par :

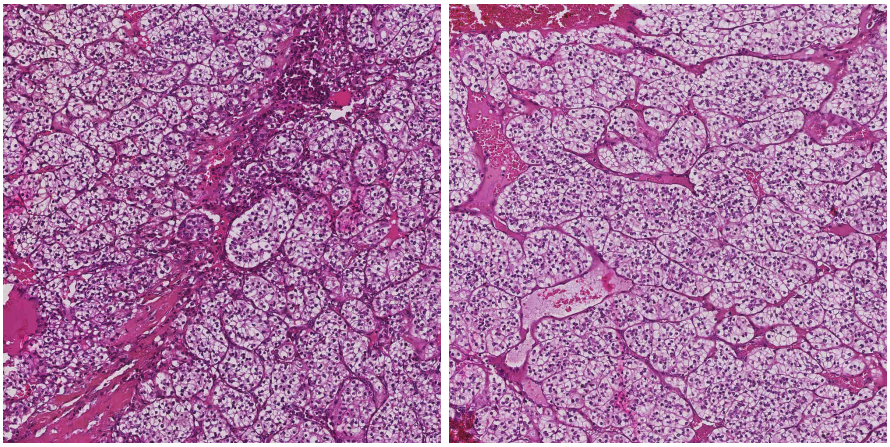
$$E(p) = -\sum_{q \in \mathrm{Hist}(\mathcal{V}(p))} q \log q. \tag{12}$$

où $\mathrm{Hist}(\mathcal{V}(p))$ est l'histogramme des niveaux de gris sur le voisinage $\mathcal{V}(p)$.

En pratique, $\mathcal{V}(p)$ sera le voisinage $n \times n$ (avec n impair) autour de p. Il en découle donc une image d'entropie locale $E_{loc}(I)$ de l'image I à laquelle on applique la formule 12 à chacun de ses pixels. Toujours dans le but de caractériser l'hétérogénéité de certaines zones



(a) Variabilité inter-images



(b) Variabilité intra-image

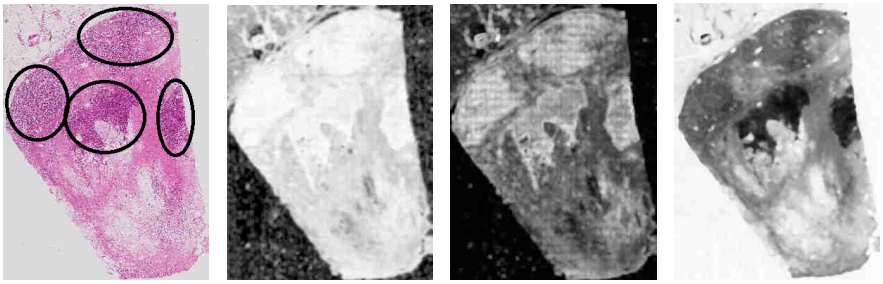
FIGURE 28 – Illustration de la variabilité des images données.

de l'image et d'ajouter une information complémentaire à l'entropie, nous proposons d'utiliser la notion de variance sur le voisinage d'un pixel.

Pour définir le caractère foncé de certaines zones de l'image, nous utilisons la valeur médiane sur le voisinage d'un pixel. Nous avons ainsi défini trois mesures locales correspondant à l'entropie, la variance et la médiane de l'image. Un exemple de réponse à ces mesures est donné en Figure 29. Désormais, chaque pixel de l'image est défini par une entropie, une variance et une médiane. Nous proposons alors d'utiliser un algorithme de classification non supervisée, la méthode des k-moyennes [HARTIGAN et M. A. WONG, 1979], pour segmenter les zones tumorales.

L'algorithme des k-moyennes cherche à trouver un partitionnement à k classes d'un ensemble de points X minimisant la somme des carrés des distances au barycentre pour chaque partition. En d'autres termes, l'algorithme cherche un partitionnement $\Pi = \{\Pi_1, \ldots, \Pi_k\}$ de l'ensemble X en cherchant une solution du problème :

$$\arg\min_{\Pi} \sum_{i=1}^{k} \sum_{x \in \Pi_i} \|x - \mu_i\|^2 \tag{13}$$



(a) Image initiale (b) Entropie (c) Variance (d) Médiane

FIGURE 29 – Les différentes mesures appliquées à l'image. Sur l'image initiale, les zones d'intérêt sont grossièrement entourées en noir.

où, pour tout $i \in [\![1, k]\!]$, μ_i est le barycentre de Π_i. La méthode des k-moyennes propose d'atteindre un minimum (potentiellement local) de (13) par itération, jusqu'à convergence [1], de deux étapes :

— Création des partitions :

$$\forall i \in [\![1, k]\!], \Pi_i^t = \{x \in X, \forall j \in [\![1, k]\!], \|x - \mu_i^t\| \leqslant \|x - \mu_j^t\|\}.$$

Il s'agit en fait d'une partition de Voronoi selon les moyennes [AURENHAMMER, 1991].

— Mise à jour des barycentres :

$$\forall i \in [\![1, k]\!], \mu_i^{t+1} = \frac{1}{\#\Pi_i^t} \sum_{x \in \Pi_i^t} x,$$

où $\#\Pi_i$ désigne le cardinal de l'ensemble Π_i et t l'indice de l'itération courante. Les barycentres initiaux μ_i^0 ($1 \leqslant i \leqslant k$) peuvent être choisis aléatoirement.

Bien que la convergence soit garantie, les partitions obtenues dépendent de l'initialisation des barycentres. Pour cette raison, l'algorithme peut être répété m fois, auquel cas le choix de la partition finale d'un point se fait par vote majoritaire sur les différentes répétitions de l'algorithme. Dans ce qui suit, le nombre m de répétition des k-moyennes sera égal à 3.

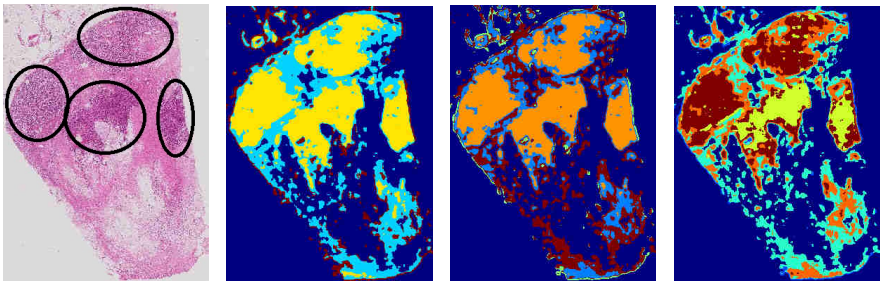
4.2.2 Résultats de segmentation

Avant d'appliquer la méthode proposée en section précédente aux images de la base de données, il est pertinent de calculer un masque éliminant les zones vides de la lame observée (apparaissant en blanc sur la Figure 30a) dans le but d'éviter de biaiser le classifieur avec des pixels n'apportant aucune information. Ce masque est calculé de

1. En pratique, la convergence est supposée atteinte lorsqu'aucun changement n'apparait entre deux itérations consécutives (propriété garantie théoriquement puisque chaque itération permet d'atteindre une meilleure partition au sens de l'équation (13)), ou lorsqu'un nombre maximal d'itérations est atteint

manière simple : il correspond aux pixels de trop grande intensité (supérieure à 0.9 en pratique) sur les canaux rouge, vert et bleu de l'image (normalisés entre 0 et 1). Les pixels ignorés apparaissent en bleu marine sur les Figures 30b, 30c et 30d.

Le partitionnement en k-moyennes est ensuite appliqué à l'ensemble des pixels de l'image, pour différentes valeurs de k. Les résultats de segmentation sont donnés en Figure 30. On constate que les zones d'intérêt (entourées en noir sur la Figure 30a) sont bien mises en évidence par la méthode de segmentation proposée pour des valeurs raisonnablement faibles de k. En revanche, une valeur de k trop élevée (Cf. Figure 30d) mène à un surapprentissage et oblige donc à devoir sélectionner plusieurs classes pour segmenter la zone d'intérêt. En pratique, nous avons donc choisi $k = 3$ ou $k = 4$.



(a) Image initiale **(b)** $k = 3$ **(c)** $k = 4$ **(d)** $k = 5$

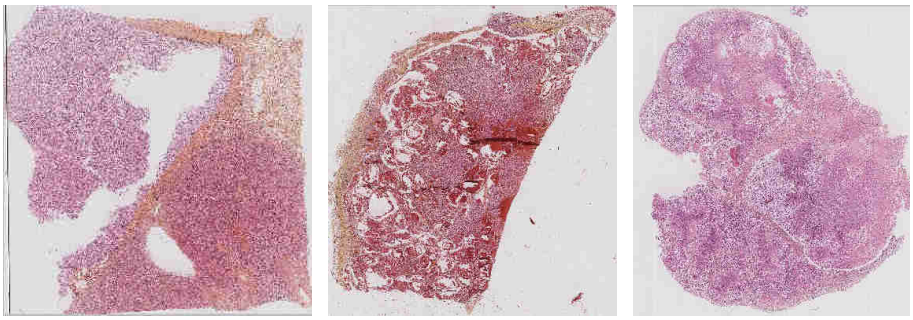
Figure 30 – Résultats de segmentation grâce au partitionnement par k-moyennes pour différentes valeurs de k. Sur l'image initiale, les zones d'intérêt sont grossièrement entourées en noir.

Parmi les différentes classes données par les k-moyennes (Cf. Figure 30), celle qui sera choisie est celle pour laquelle l'entropie moyenne est la plus élevée. D'autres résultats de partitionnement sont donnés en Figure 31 et montrent quelques limites de la méthode proposée. Celles-ci sont en partie dues aux zones floues que l'on peut retrouver sur certaines images.

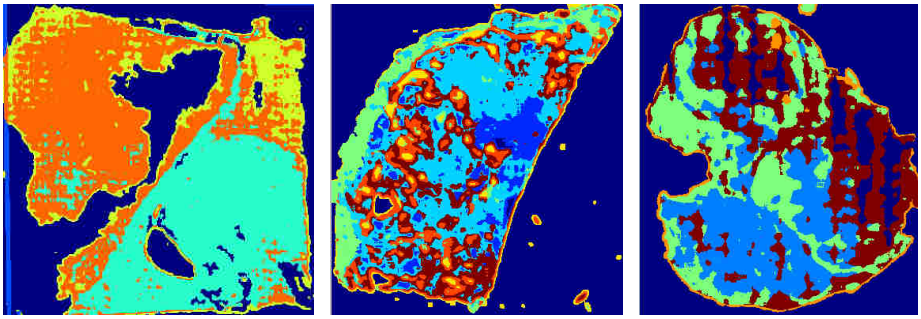
L'étape de segmentation proposée, bien que donnant des résultats intéressants pour certaines images, mériterait donc d'être améliorée pour pouvoir être utilisée de façon systématique. En pratique, nous utiliserons une segmentation manuelle puisque le nombre d'images utilisées est faible.

4.2.3 Découpage de la région d'intérêt

Nous découpons la segmentation des régions d'intérêt de la lame en sous-images de taille raisonnable sur lesquelles l'analyse du réseau vasculaire pourra être effectuée, avec un recouvrement entre sous-images voisines pour pouvoir combiner l'information locale extraite en une seule information globale. La taille des images choisies est de



(a) Acquisitions



(b) Partitionnements obtenus

FIGURE 31 – Résultats de segmentation sur trois lames.

3200×3200, avec un recouvrement (horizontal et vertical) de 10% sur les quatre côtés de l'image.

Entre deux images voisines, il existe donc un recouvrement de 320 pixels. Comme évoqué précédemment, cette zone d'intersection permet de combiner l'information entre les images voisines. Une question se pose alors naturellement : De laquelle de ces deux images prend-on l'information se trouvant sur la zone de recouvrement ? Nous proposons alors de découper le recouvrement en deux zones de confiance, comme illustré par la Figure 32 dans le cas d'un recouvrement horizontal. L'information se situant sur la première moitié de la zone de confiance (à gauche dans le cas horizontal) sera extraite de la première image, tandis que l'information située sur la seconde moitié du recouvrement (à droite dans le cas horizontal) sera extrait sur la seconde image. Le même raisonnement peut être transposé au cas vertical. Le but de cette manipulation est d'éviter les effets de bord qui peuvent survenir lors de l'extraction de squelette permettant de construire le graphe décrivant le réseau vasculaire (Cf. Section 4.3.3). Formellement, cela revient à ne pas prendre en considération la bordure extérieure d'épaisseur de l'image $I = (I(i,j))_{1 \leqslant i,j \leqslant 3200}$ lors de l'analyse du graphe et donc à ne s'intéresser qu'à la sous-image $I' = (I(i,j))_{1+160 \leqslant i,j \leqslant 3200-160}$.

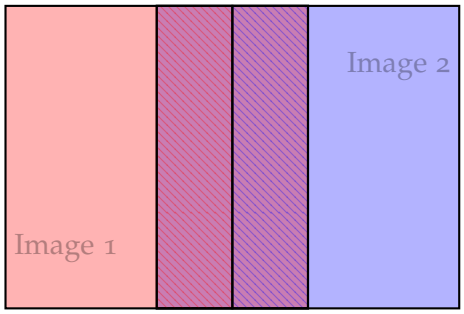


FIGURE 32 – Gestion du recouvrement entre deux images voisines. Les parties hachées correspondent aux zones de confiance de chacune d'entre elles.

4.3 ANALYSE DE L'ARCHITECTURE

Dans cette section, nous nous intéressons à l'architecture de la tumeur, c'est-à-dire à la structure de son réseau vasculaire. Pour cela, nous observons le tissu tumoral sur les images de grossissement maximal extraites en Section 4.2 et proposons une méthode permettant de décrire l'architecture du réseau vasculaire.

4.3.1 Pré-traitements

Combinaison de canaux

Nous disposons initialement d'une série d'images RGB encodées sur 8 bits. Puisque les vaisseaux apparaissent en violet sur les images (Cf Figure 33a), il est important d'utiliser un canal (ou une combinaison de canaux) sur lequel les structures vasculaires apparaissent de façon la plus claire possible.

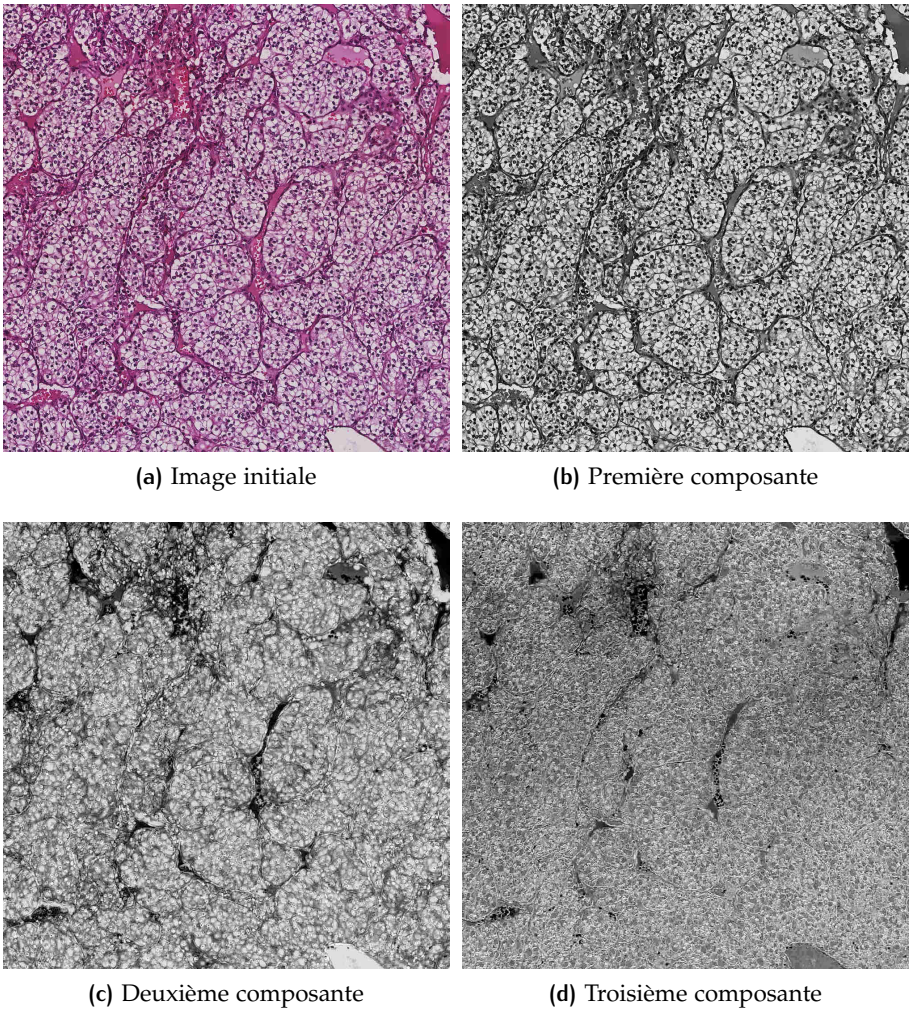
De tels pré-traitements pour les images histologiques existent dans la littérature. Le plus couramment utilisé est une simple combinaison linéaire empirique entre les canaux rouge et bleu correspondant à la teinte violette souhaitée, par exemple une moyenne (normalisée) entre ces deux canaux :

$$\text{Violet}(p) = 0.5\frac{\text{Rouge}(p) + \text{Bleu}(p)}{\sqrt{\text{Rouge}(p)^2 + \text{Vert}(p)^2 + \text{Bleu}(p)^2}} \tag{14}$$

pour chaque pixel p de l'image (Cf. Figure 34).

D'autres utilisent une analyse en composante principale (ACP) [JOLLIFFE, 2002] sur l'ensemble des pixels [PENG et al., 2011] (Cf. Figure 33).

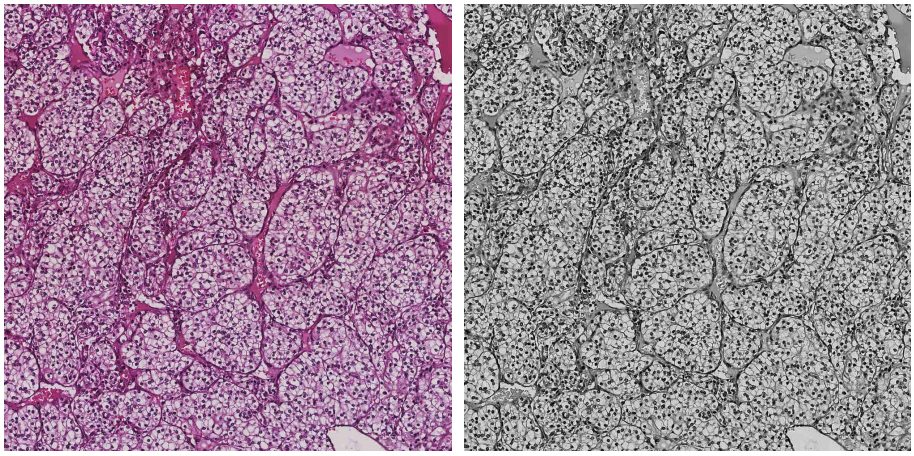
Plus récemment, des formules empiriques correspondant aux marqueurs utilisés en histologie (dont l'éosine et l'hématoxyline) ont été proposées [ALSHEH ALI, 2015]. Ces formules ont recours aux canaux invariants par contraste proposés dans [GEVERS et SMEULDERS, 2000].



(a) Image initiale **(b)** Première composante

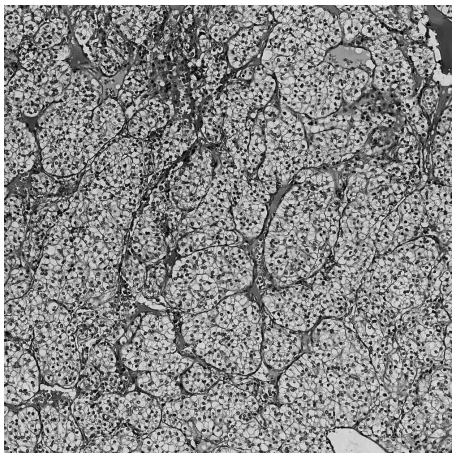
(c) Deuxième composante **(d)** Troisième composante

FIGURE 33 – Un exemple d'image et les trois composantes données par l'ACP.

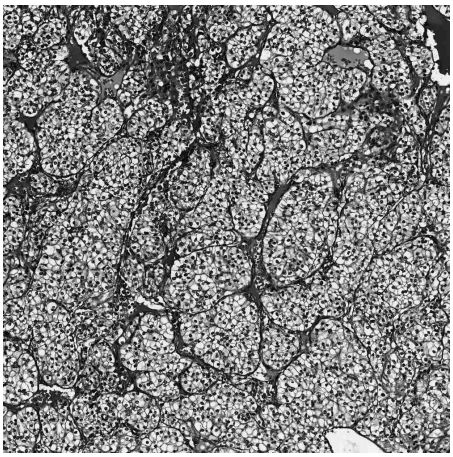


(a) Image initiale

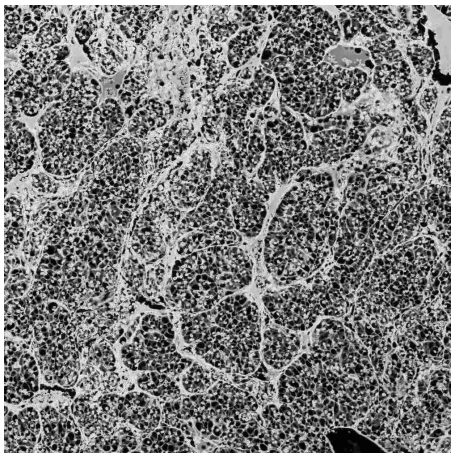
(b) Rouge



(c) Bleu

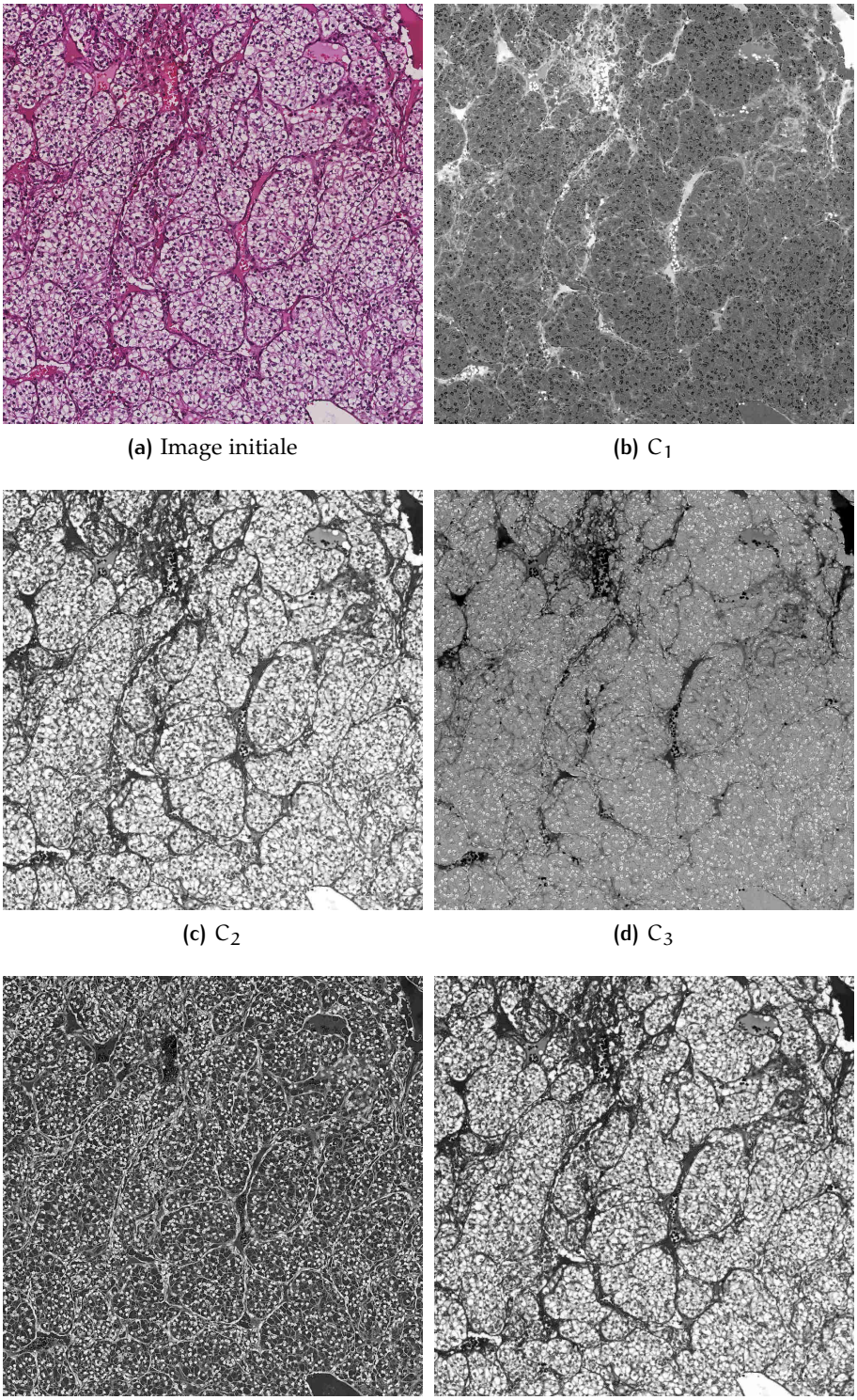


(d) Vert



(e) Violet

Figure 34 – Un exemple d'image, les canaux rouge, vert et bleu correspondant et l'image violette calculée.



(a) Image initiale **(b)** C_1

(c) C_2 **(d)** C_3

(e) Hématoxyline **(f)** Éosine

FIGURE 35 – Un exemple d'image, les canaux C_1, C_2, C_3, et les résultats des canaux empiriques correspondant à l'hématoxyline et l'éosine.

Pour chaque pixel p de l'image, ces canaux C_1, C_2 et C_3 sont définis par :

$$C_1(p) = \arctan\left(\frac{\text{Red}(p)}{\max(\text{Vert}(p), \text{Bleu}(p))}\right), \tag{15a}$$

$$C_2(p) = \arctan\left(\frac{\text{Vert}(p)}{\max(\text{Rouge}(p), \text{Bleu}(p))}\right), \tag{15b}$$

$$C_3(p) = \arctan\left(\frac{\text{Bleu}(p)}{\max(\text{Vert}(p), \text{Rouge}(p))}\right). \tag{15c}$$

Les canaux H et E proposés dans [Alsheh Ali, 2015] et correspondant respectivement à l'hématoxyline et à l'éosine, sont alors donnés par :

$$H(p) = \frac{\text{Bleu}(p)}{C_3(p)}, \tag{16a}$$

$$E(p) = \frac{\text{Vert}(p)}{\text{Rouge}(p)}. \tag{16b}$$

La Figure 35 montre le résultat de ces canaux sur une des images mises à disposition.

Parmi les pré-traitements évoqués, deux seront retenus pour la suite des travaux :

— le canal V, correspondant à la teinte violette (Cf. équation (14)), mettant en évidence le réseau vasculaire,
— le canal H, correspondant à l'hématoxyline (Cf. équation (16a)), mettant en évidence les noyaux.

Détection des noyaux

Une étape de détection des noyaux peut paraître contre-intuitive dans le cadre de l'étude du réseau vasculaire. En réalité, les noyaux des cellules peuvent apporter une information pertinente et complémentaire à l'architecture tumorale.

Dans un premier temps, il est important de noter que les noyaux, en réagissant avec l'hématoxyline, apparaissent de manière prononcée sur le canal H, mais également sur le canal V (Cf. Figure 36). Sur ce dernier canal, qui sera utilisé lors de l'extraction du réseau vasculaire puisque que c'est celui qui le fait apparaître le plus clairement, les noyaux apparaitront donc aussi intensément que les vaisseaux, ce qui posera problème lors de la phase d'extraction de structures tubulaires (Cf. Section 4.3.2). La détection des noyaux cellulaires constitue donc une étape de pré-traitement permettant de ne pas prendre en considération les noyaux lors de la détection des vaisseaux. Par ailleurs, puisque les cellules tumorales prolifèrent le long du réseau vasculaire, les noyaux ont tendance à s'aligner le long des vaisseaux. La détection des noyaux peut par conséquent apporter une information supplémentaire sur les vaisseaux.

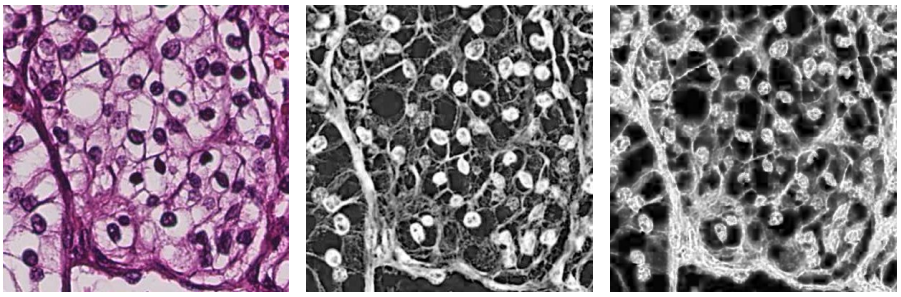


FIGURE 36 – Exemple d'image montrant un agrégat de noyaux dans un vaisseau. De gauche à droite : l'image initiale, le canal H, et le canal V.

Or, comme le montrent les Figures 36 (en particulier le canal H) et 37, les vaisseaux sont également composés de noyaux. Ces noyaux sont, quant à eux, à prendre en compte lors de la segmentation des vaisseaux puisque les ignorer pourrait engendrer des discontinuités indésirables dans détection du réseau vasculaire. La détection des noyaux doit donc se faire en deux étapes :

— Détection de l'ensemble des noyaux,
— Distinction entre les noyaux des cellules et les noyaux des vaisseaux.

La procédure de détection des noyaux se fait sur le canal H, où ils sont le plus visibles, et utilise des outils de morphologie mathématique [HARALICK et al., 1987]. Le but est de détecter les zones suffisamment larges (au moins de la taille minimale d'un noyau de cellule) et intenses. Pour cela, une ouverture morphologique, un seuillage, et une fermeture morpholgique sont successivement appliqués à l'image pour en segmenter les noyaux. À ce stade, les noyaux sont détectés, qu'ils soient dans les cellules ou dans les vaisseaux (Cf. Figure 37).

Pour distinguer les noyaux de cellules tumorales des noyaux apparaissant dans les vaisseaux, plusieurs critères sont à prendre en compte sur la segmentation. Tout d'abord, les noyaux apparaissant dans les vaisseaux sont généralement sous forme d'agrégats, et/ou ont tendance à s'aligner le long des vaisseaux (Cf. Figures 36 et 37). La taille des composantes connexes de la segmentation des noyaux constitue donc un premier critère distinguant les deux types de noyaux : une composante de taille supérieure à un certain seuil (par exemple égal à 5 fois l'aire d'un noyau moyen) correspond à un agrégat de noyaux dans un vaisseau. De plus, puisque les noyaux ont tendance à s'aligner, un critère intéressant est l'excentricité de la composante connexe (définie comme l'excentricité de l'ellipse ayant les mêmes moments d'ordre 2) : une composante trop allongée dans une certaine direction sera alors interprétée comme un agrégat de noyaux dans un vaisseau.

La suppression des composantes connexes de taille trop grande et d'excentricité arbitrairement faible devant 1 donne une segmen-

tation des noyaux de cellules n'appartenant pas aux vaisseaux (Cf. Figure 37).

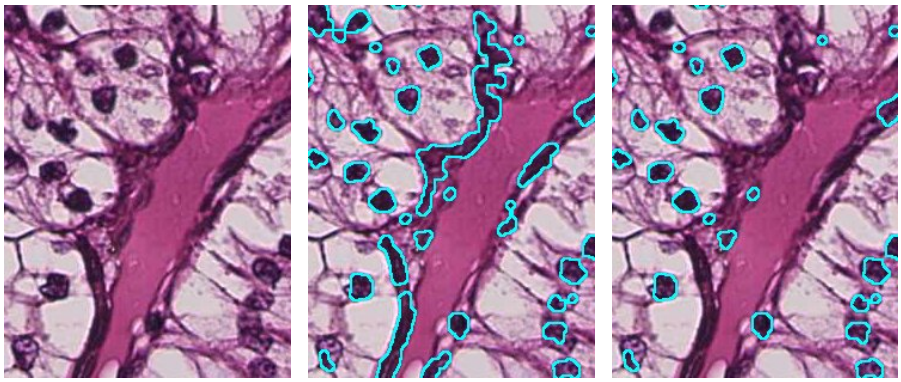


FIGURE 37 – Étapes de la segmentation des noyaux n'appartenant pas aux vaisseaux. De gauche à droite : l'image initiale, la segmentation de tous les noyaux, et la segmentation des noyaux en dehors des vaisseaux.

Les pixels des noyaux de cellules hors vaisseaux détectés sont ensuite forcés à zéro sur le canal V. Il en résulte une image où une grande partie des noyaux hors vaisseaux sont absents et donc où les vaisseaux sont mis en évidence. Un exemple de résultats obtenus est donné en Figure 38.

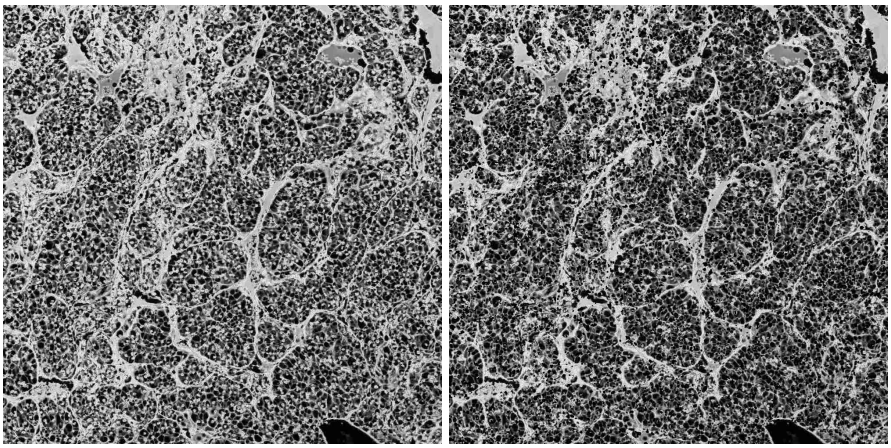
4.3.2 Extraction du réseau vasculaire

Seuillage du canal V

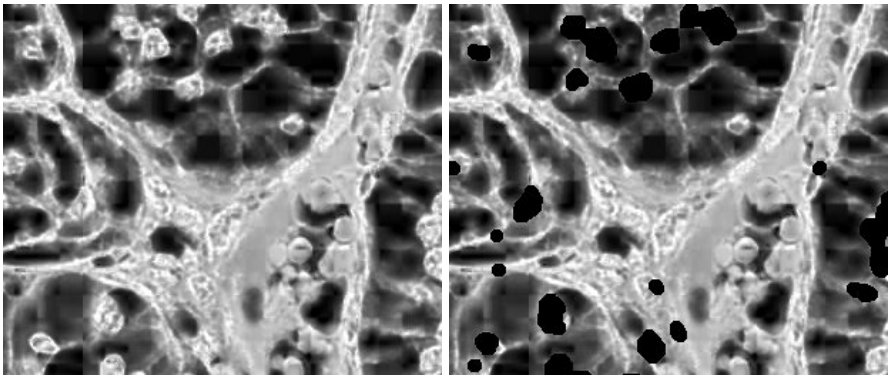
Une étape de seuillage est nécessaire afin de limiter le bruit autour des vaisseaux. Compte tenu du fait que l'on cherche à extraire les structures tubulaires du canal V, le seuillage par hystérésis est plus adapté qu'un seuillage simple [CONDURACHE et AACH, 2005]. Le seuil haut permet d'isoler les principales parties du réseau, plus intenses sur le canal V (privé des noyaux de cellules), tandis que le seuil bas permet de les relier entre elles. Les dernières traces de bruit sont enlevées par suppression des plus petites composantes connexes. Les résultats de cette opération sont donnés en Figure 39.

Détection de structures tubulaires par filtrage

Il s'agit désormais de détecter des structures tubulaires sur une image bidimensionnelle en niveaux de gris. De nombreux filtres permettant de détecter de tels motifs ont été proposés et utilisés dans la littérature [FRANGI et al., 1998 ; JONES et PALMER, 1987 ; KRISSIAN et al., 2000 ; POCK et al., 2005 ; SOFKA et STEWART, 2006]. Parmi eux, le



(a) L'ensemble de l'imagette



(b) Partie de l'imagette

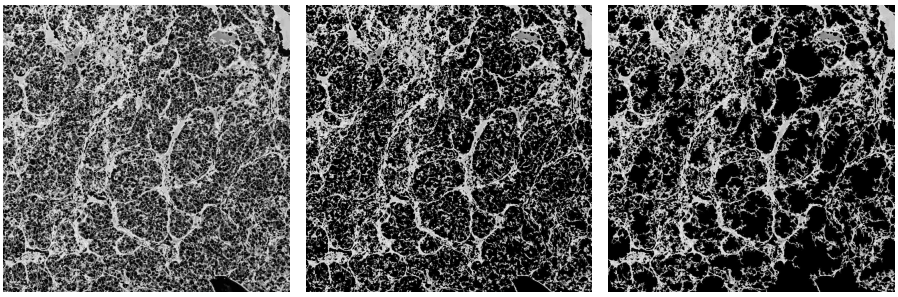
FIGURE 38 – Exemple de suppression des noyaux hors vaisseaux sur le canal V : à gauche, le canal V, à droite, le canal V où les pixels correspondant aux noyaux hors vaisseaux ont été mis à zéro.

filtre de Gabor fait partie des plus couramment utilisés. Dans le cas 2D, il peut être défini par :

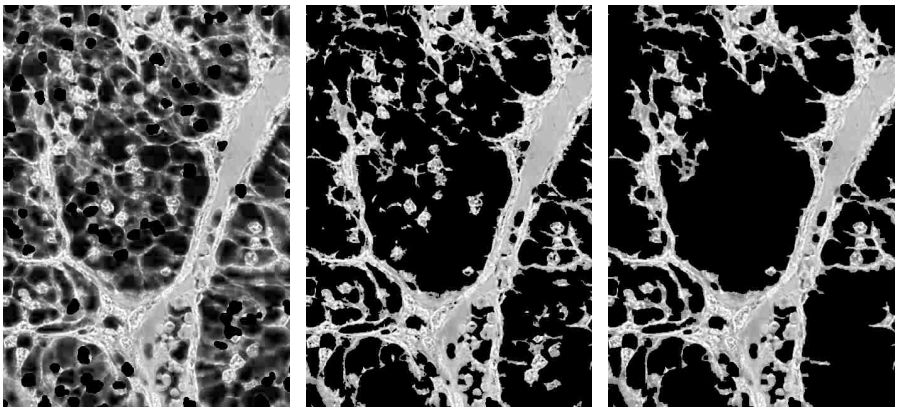
$$G(x,y) = \cos\left(\frac{x'\pi}{\sqrt{2}\sigma}\right)\exp\left(-\frac{x'^2+y'^2}{2\sigma^2}\right) \tag{17}$$

où $x' = x\cos\theta + y\sin\theta$, $y' = x\sin\theta - y\cos\theta$ et σ et θ sont respectivement l'étendue et l'angle d'orientation du filtre. La Figure 40 donne quelques exemples de filtres de Gabor pour différentes valeurs des paramètres σ et θ. Filtrer une image I revient donc à calculer la convolution de I par G pour différentes valeurs de θ et σ, puis à prendre la réponse R maximale à chaque pixel.

Afin d'apporter quelques améliorations aux performances du filtre de Gabor, nous proposons d'en utiliser une version légèrement modifiée [MOTTINI, 2014]. L'approche consiste à couper le filtre de Gabor



(a) Sur l'ensemble de l'imagette



(b) Sur une partie de l'imagette

FIGURE 39 – Pré-détection des vaisseaux par seuillage par hystérésis. De gauche à droite : l'image d'entrée (Cf. Figure 38), la réponse du seuillage par hystérésis, et le résultat final, après suppression des plus petites composantes connexes.

en deux au niveau de son axe de symétrie. Formellement, cela revient à définir deux demi-filtres G^+ et G^- par :

$$G^+(x,y) = \begin{cases} G(x,y) & \text{si } x' \geqslant 0, \\ 0, & \text{sinon.} \end{cases} \tag{18}$$

$$G^-(x,y) = \begin{cases} G(x,y), & \text{si } x' \leqslant 0, \\ 0, & \text{sinon.} \end{cases} \tag{19}$$

puis à appliquer chacun de ces filtres à l'image pour différentes valeurs de θ et σ. La Figure 41 donne une illustration des filtres de Gabor proposés en comparaison au filtre de Gabor classique. Le filtre, appliqué à une image donnée, renvoie donc deux réponses R^+ et R^- aux demi-filtres G^+ et G^-, respectivement. La réponse finale R^F est alors définie comme étant le minimum (pixel à pixel) de ces deux réponses. Pour que la réponse finale soit élevée en un pixel donné, il faut alors que les deux réponses R^+ et R^- soient élevées en ce pixel. La Figure 42 (colonnes 2 et 3) donne un résultat de réponse au filtre utilisé sur une des images mises à disposition.

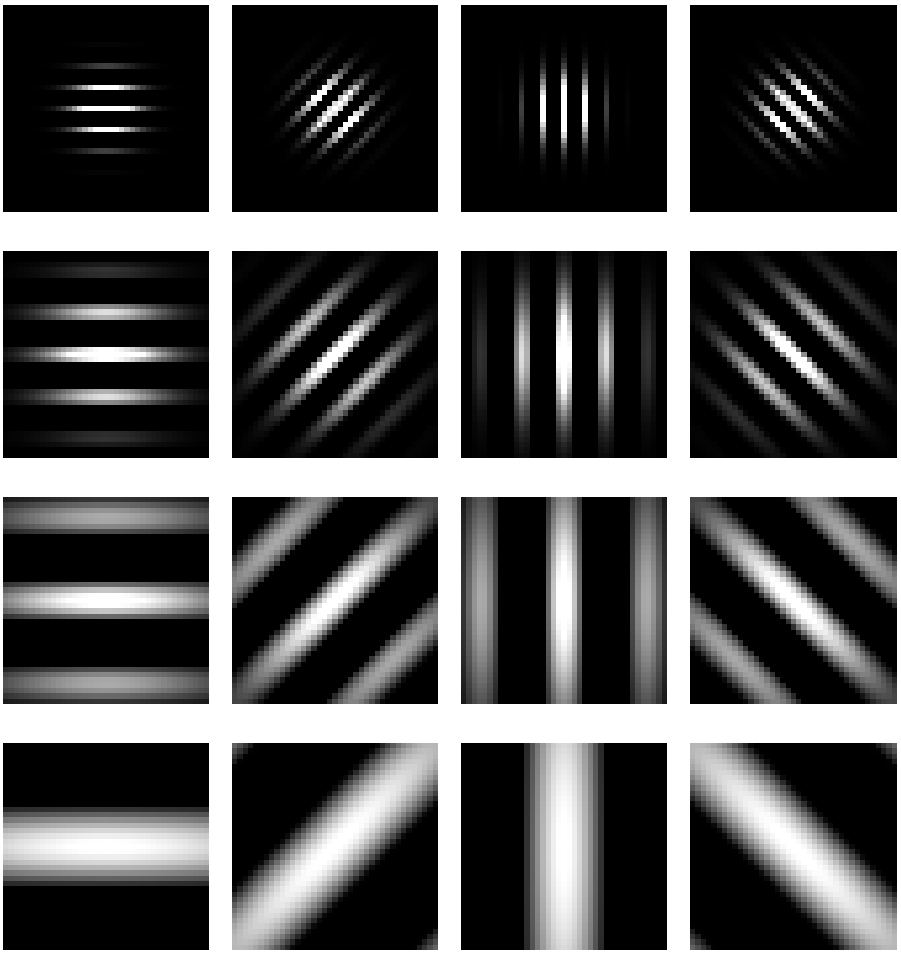


FIGURE 40 – Filtres de Gabor pour différentes valeurs de θ (colonnes) et σ (lignes).

Segmentation de la réponse au filtre de Gabor

Une fois le filtre appliqué, la prochaine étape consiste à segmenter la réponse résultante. Pour cela, un autre seuillage par hystérésis est appliqué à l'image. L'objectif de cette étape reste le même : il s'agit, dans un premier temps, d'appliquer un seuil haut pour mettre en exergue les zones de forte réponse au filtre de Gabor, puis, dans un second temps, de sélectionner les zones au dessus d'un seuil bas, uniquement si elles sont connexes aux zones de forte réponse.

Afin d'obtenir un résultat plus propre et moins bruité, quelques post-traitements sont ensuite appliqués à la segmentation obtenue :

— Une fermeture morphologique [HARALICK et al., 1987] permettant de lisser la segmentation,
— Un remplissage des trous dans la segmentation de taille inférieure à un certain seuil [2],

2. Cette étape est importante puisque nous allons calculer le squelette de la segmentation (Cf. Section 4.3.3). En effet, chaque trou ajoutera des boucles au squelette, ce qui compliquera inutilement le graphe souhaité.

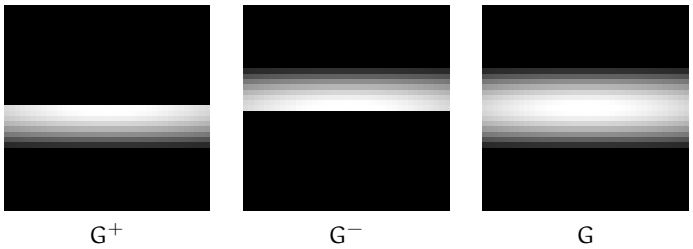


FIGURE 41 – Les demi-filtres de Gabor utilisés et le filtre de Gabor (complet) correspondant.

— Une suppression des trop petites composantes connexes, censées représenter du bruit de segmentation.

La Figure 42 (colonnes 4 et 5) illustre la segmentation de la réponse au filtre de Gabor.

4.3.3 Graphe représentant le réseau vasculaire

Le réseau vasculaire étant détecté, il suffit d'en extraire un squelette afin de pouvoir le représenter sous forme de graphe.

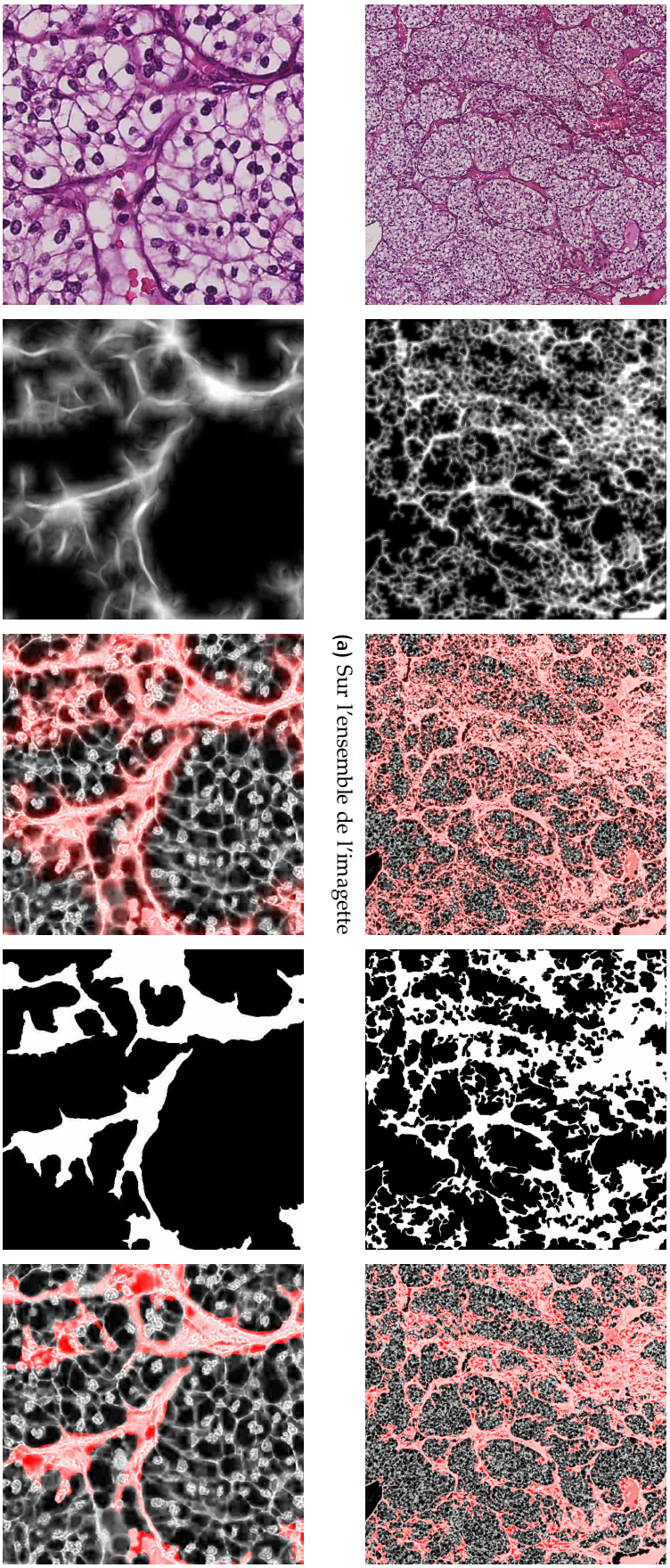
Extraction du squelette de la segmentation

De nombreux algorithmes d'extraction de squelette d'une composante connexe ont été proposés dans la littérature [HOWE, 2013 ; NÄF et al., 1997 ; OGNIEWICZ et ILG, 1992 ; TELEA et VAN WIJK, 2002]. Parmi eux, les méthodes d'amincissement [JANG et CHIN, 1990 ; LAM et al., 1992 ; T. ZHANG et SUEN, 1984], qui seront appliquées par la suite, sont parmi les plus robustes et utilisées. La Figure 44 (colonne 2) donne le résultat d'un tel algorithme sur nos échantillons.

Post-traitements du squelette obtenu

Quelques post-traitements sont à appliquer pour rendre le squelette plus fidèle à la réalité. Dans un premier temps on constate que de nombreuses branches parasites apparaissent malgré les efforts fournis en Section 4.3.2 pour lisser la segmentation des vaisseaux. Il s'agit d'un problème récurrent lorsque l'on cherche à extraire le squelette d'une surface. Cela est généralement dû à deux facteurs :

— Les aspérités de la surface, lorsqu'elles sont trop importantes, peuvent être interprétées comme des branches (Cf. Figure 44b, colonne 2),
— Les surfaces trop étendues (des vaisseaux épais ou des intersections de vaisseaux dans notre cas) sont généralement mal gérés par de tels algorithmes qui retournent souvent des branches verticales, horizontales et/ou diagonales afin que le squelette recouvre la surface (Cf. Figure 44c, colonne 2).



(a) Sur l'ensemble de l'imagette

(b) Sur une partie de l'imagette

Figure 42 – Segmentation du réseau vasculaire. De gauche à droite : l'image initiale, la réponse au filtre de Gabor modifié, la réponse du filtre superposée au canal V, le seuillage de la réponse au filtre et le seuillage de l'image superposé au canal V.

Pour palier ces sur-détections, il faudrait supprimer les branches terminales plus petites qu'un certain seuil ainsi que celles ayant un nombre trop important de pixels voisins alignés sur une droite (verticale, horizontale ou diagonale). Pour cela, nous utilisons la méthode utilisée dans [Howe, 2013] pour extraire les branches du squelette. Il suffit en effet de détecter l'ensemble J des jonctions du squelette et de forcer à zéro la valeur des pixels du voisinage 3×3 de chaque jonction $j \in J$, comme illustré en Figure 43.

$$\begin{array}{c|c|c} x & x & x \\ \hline x & j & x \\ \hline x & x & x \end{array} \rightarrow \begin{array}{c|c|c} 0 & 0 & 0 \\ \hline 0 & 0 & 0 \\ \hline 0 & 0 & 0 \end{array}$$

Figure 43 – Procédé de découpage d'un squelette en branches. j est le pixel de jonction (et vaut donc systématiquement 1 puisqu'appartenant au graphe), et les x représentent le voisinage 3×3 et valent soit 0, soit 1.

Dès lors, chaque composante connexe de l'image résultante correspond à une branche du squelette. Il est alors aisé d'en extraire les branches non-terminales puisque ce sont les branches ayant une jonction à chacune de ses deux extrémités. Les autres, n'ayant qu'une jonction à l'une de leurs extrémités, sont des branches terminales.

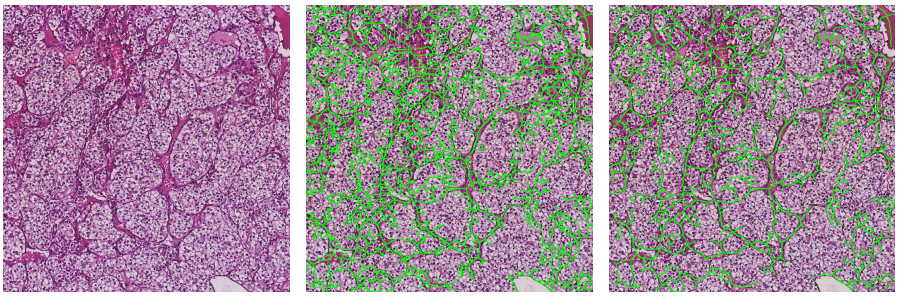
Il ne reste donc plus qu'à supprimer les branches de taille inférieure à un certain seuil, ou celles comptant un nombre trop grand de pixels voisins alignés. Ce post-traitement est à itérer jusqu'à ce que le squelette reste inchangé. En effet, une seule itération ne supprimerait que les ramifications de petites branches. En pratique, sur les données fournies, quelques itérations (au plus 5) suffisent. Un résultat de squelette final est donné en Figure 44 (colonne 3).

4.3.4 Combinaison d'information : Fusion de graphes voisins

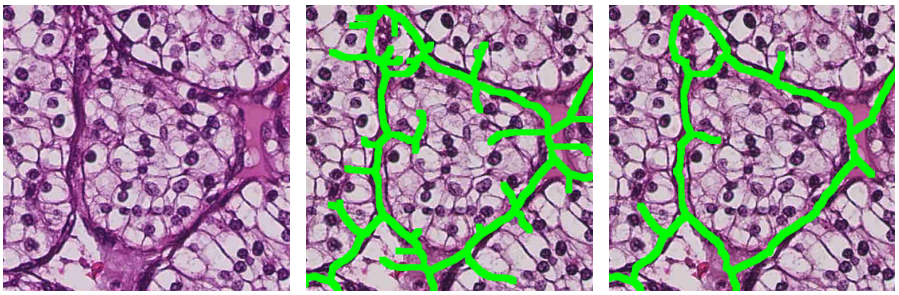
Dans un contexte de découpage des images, une étape de fusion de l'information est nécessaire pour résumer les multiples résultats locaux en un seul résultat global. Ici, il suffit de fusionner les graphes extraits sur chaque image afin d'obtenir un graphe décrivant l'ensemble de l'image.

La principale problématique de cette étape réside dans la gestion des effets de bords dus à l'extraction du squelette d'une sous-image. En effet, l'analyse du squelette aura tendance à considérer les branches en bord d'image comme des branches terminales, ce qui n'est pas le cas si la branche se prolonge dans l'image voisine.

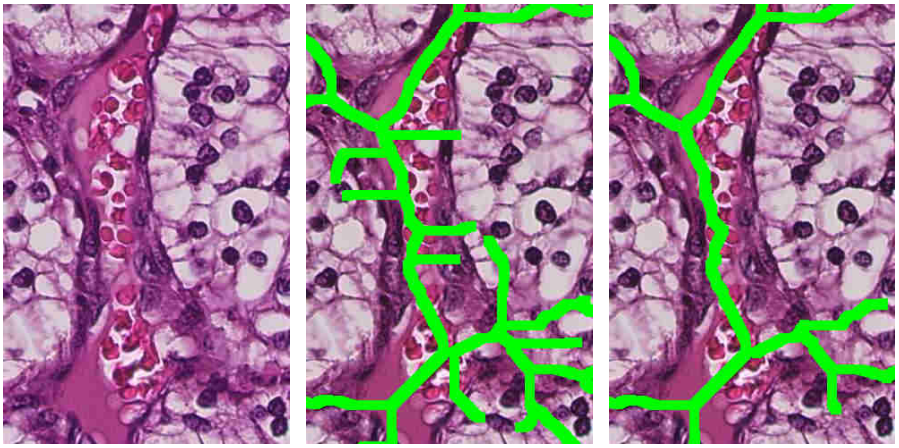
Cette difficulté peut être surmontée simplement dans la mesure où il existe un recouvrement (entièrement déterminé après le découpage fait en Section 4.2.3) entre les images voisines. Il n'y a donc aucune



(a) Sur l'ensemble de l'imagette



(b) Sur une partie de l'imagette



(c) Sur une (autre) partie de l'imagette

FIGURE 44 – Extraction du squelette correspondant au réseau vasculaire. De gauche à droite : l'image initiale, le premier squelette détecté et le squelette post-traité.

tâche de recalage préalable à réaliser. Il suffit par conséquent de privilégier une branche non terminale à une branche terminale lors de la fusion de graphes voisins.

4.4 ANALYSE DU GRAPHE

Une fois que le graphe représentant les vaisseaux est calculé, il reste à en calculer des caractéristiques permettant de représenter le réseau vasculaire.

4.4.1 Extraction des données

Des caractéristiques ayant un sens biomédical peuvent être extraites du graphe. Tout d'abord, comme évoqué en Section 4.1.3, nous sommes en particulier intéressés par l'aspect incohérent du développement du réseau vasculaire, caractéristique pressentie indicatrice de la malignité d'une tumeur. Nous proposons pour cela de prendre en compte deux critères dans l'analyse du graphe global obtenu à l'issue de la Section 4.3.4 : le nombre de branches terminales ainsi que leur longueur. L'idée sous-jacente à cette proposition est que plus le réseau se développe de manière chaotique, plus les chances de voir apparaître des branches terminales seront importantes.

En résumé, les caractéristiques suivantes seront calculées :

— le nombre de branches terminales (NE),
— la longueur moyenne des branches terminales (LE),
— le nombre de jonctions (NJ),
— la longueur moyenne des branches non-terminales (LJ).

L'implémentation de ce calcul est immédiate dans la mesure où elle ne demande que de compter les composantes connexes d'un ensemble de branches, et le nombre de pixels de chacune d'entre elles.

4.4.2 Interprétations biomédicales des données

Le tableau 5 résume les caractéristiques numériques ayant été extraites pour chaque lame et les confronte aux données histologiques (grade de Fuhrman[3] [Fuhrman et al., 1982] et type histologique). Bien que le nombre de cas étudiés soit relativement faible, les résultats obtenus permettent de confirmer quelques intuitions biomédicales intéressantes.

En effet, les mesures semblent relativement homogènes pour les patients pour lesquels on dispose de plusieurs lames. Par ailleurs, les valeurs du ration NE/NJ des carcinomes rénaux à cellules claires

3. Il s'agit d'un score entier compris entre 1 et 4 caractérisant la malignité de la tumeur, 1 étant le plus bénin, et 4 le plus malin.

(ccRCC) semble inférieur à celui des carcinomes papillaires (pap). Ensuite, la longueur moyenne des branches terminales est plus longue dans le cas des carcinomes papillaires que dans le cas du carcinome à cellules claires.

4.5 CONCLUSION

4.5.1 Chaîne de traitement proposée

Dans ce chapitre, une chaîne de traitement permettant l'étude au niveau microscopique à grande échelle du réseau vasculaire d'une tumeur a été proposée dans un domaine encore peu exploré dans la littérature. L'approche permet d'analyser l'architecture de la tumeur à faible grossissement (segmentation des différentes parties du tissu : tissu sain et tumoral, sang, ... Cf. Section 4.2), puis de représenter le réseau vasculaire de manière simple (par un graphe, Cf. Section 4.3) et d'en extraire des caractéristiques descriptives du tissu observé de manière semi-automatique.

La méthode imite la démarche qu'ont les anatomopathologistes lorsqu'ils sont confrontés à de telles données, mais a pour avantage d'extraire une grande quantité d'information en un temps raisonnable. En effet, dégager le squelette représentant le réseau vasculaire apparaissant sur de si grandes images serait une tâche extrêmement longue et fastidieuse à réaliser manuellement.

4.5.2 Temps de calcul

Compte tenu de la taille des données à disposition et du nombre d'étapes menant à l'obtention du réseau vasculaire, la question du temps de calcul se pose naturellement. Chaque sous-image (de taille 3200×3200 issue du découpage décrit en Section 4.2.3) est traitée en environ 3 minutes, ce qui donne un temps de calcul total d'environ 15 heures pour un ensemble de 300 images sur un ordinateur personnel équipé d'un processeur à 4 cœurs cadencés à 3GHz et de 16Go de mémoire vive. Quant à l'implémentation, elle a été réalisée en Python (pour l'interface OpenSlide permettant de lire les fichiers de Whole-Slide Imaging [Goode et al., 2013]) et en Matlab pour les opérations de traitement d'image.

Les étapes de segmentation des régions tumorales (Section 4.2.1), de fusion de graphe (Section 4.3.4) et de calcul des descripteurs du réseau (Section 4.4) sont de durée négligeable devant l'extraction du graphe local.

Informations patient			Données extraites							
Patient	Type histologique	Grade	sous-images	jonctions	terminaux	jonction	terminaux	Ratios		
			NI	NJ	NE	LJ	LE	NJ/NI	NE/NI	NE/NJ
1	incla	4	127	44558	7031	130,477	84,8623	350,85	55,36	0,15
2	ccRCC	3	213	139032	17794	101,308	61,1211	652,73	83,53	0,12
3	ccRCC	3	309	118645	49945	105,199	67,8457	383,96	161,63	0,42
	ccRCC	3	149	76513	26850	100,555	63,7481	513,51	180,20	0,35
	ccRCC	3	45	25354	6204	101,399	62,3908	563,42	137,86	0,24
	ccRCC	3	111	68264	28489	88,876	56,1132	614,99	256,65	0,41
4	pap	3	137	23724	13933	101,565	72,8841	173,16	101,70	0,58
	pap	3	158	50085	23164	100,572	67,4742	316,99	146,60	0,46
5	incla	4	210	92921	36994	99,864	65,612	442,48	176,16	0,39
	incla	4	318	189736	37391	103,122	60,2544	596,65	117,58	0,19
	incla	4	341	94380	57198	105,883	73,5482	276,77	167,73	0,60
6	ccRCC + pap	2	79	27382	13349	105,971	63,7832	346,60	168,97	0,48

Table 5 – Données extraites pour chacune des lames observées. NI correspond au nombre de sous-images dans le découpage de la zone tumorale (et est donc un critère directement lié à son aire).

4.5.3 Perspectives

Bien que les résultats obtenus jusqu'alors soient encourageants et semblent confirmer les intuitions généralement suivies lors de l'analyse d'images histologiques, il serait pertinent d'appliquer cette méthode à un ensemble d'images plus important (c'est-à-dire présentant plus de cas mais aussi des cas plus variés) afin que les statistiques soient plus significatives et représentatives de la réalité.

Les différentes étapes menant au calcul du graphe peuvent aussi être améliorées. Si plusieurs filtres de détection de vaisseaux [Frangi et al., 1998 ; Jones et Palmer, 1987 ; Krissian et al., 2000] et d'extraction de squelette [Gonzalez et al., 2004 ; Näf et al., 1997 ; Ogniewicz et Ilg, 1992 ; Telea et Van Wijk, 2002] ont été testés, la détection du réseau vasculaire est encore perfectible en raison des quelques sur- et sous-détections qu'elle présente.

De surcroît, la méthode proposée fait appel à de nombreux paramètres. Si certains d'entre eux peuvent être déterminés par des arguments biomédicaux (tailles minimale et maximale des noyaux, épaisseurs minimale et maximale des vaisseaux, ...), d'autres (seuils de segmentation notamment) n'ont pas été calculés automatiquement et doivent donc être fixés de manière empirique, principalement en raison de la variabilité importante de teinte entre deux échantillons observés. Les méthodes d'estimation de seuil classiques [Kapur et al., 1985 ; Kittler et Illingworth, 1985 ; Kittler, Illingworth et Föglein, 1985 ; Otsu, 1975] n'ont par ailleurs pas donné de résultats suffisamment robustes pour que leur utilisation soit envisagée.

Ces améliorations permettraient d'obtenir des résultats plus fidèles à la réalité et de comprendre plus en détail les mécanismes qui régissent l'angiogénèse dans le cas du carcinome à cellules rénales (voire d'autres types de cancers dans la mesure où la chaîne de traitement proposée pourrait être adaptée aux images issues d'autres spécialités de l'histopathologie).

5 CONCLUSION GÉNÉRALE

Sommaire

5.1 Revue des contributions 83
5.2 Perspectives 84
5.2.1 Extension d'un classifieur binaire 84
5.2.2 Analyse de la morphologie de neurones de souris 84
5.2.3 Analyse de l'architecture d'un carcinome rénal 85

5.1 REVUE DES CONTRIBUTIONS

Dans ce travail de thèse, nous avons abordé le traitement et la classification d'images biomédicales issues de sources diverses (microscopie confocale, histopathologie digitale, endoscopie) et touchants des domaines de la biologie et de la médecine variés (études du cortex mammifère, analyse de l'architecture d'une tumeur rénale, classification de vidéos d'endomicroscopie).

Nous avons commencé par définir un cadre général permettant d'étendre un classifieur binaire au cas à $n \geqslant 3$ classes, et dont l'intérêt est la possibilité d'atteindre un nombre logarithmique de classifieurs utilisés lors de la phase de classification tout en conservant une complexité d'apprentissage raisonnable (Cf. Chapitre 2). Cette méthode fait appel à des outils de bipartitionnement de graphes pondérés (Cf. Annexe B) permettant de calculer de manière efficace l'arbre de classifieurs, et est appliquée au cas des SVM (Cf. Annexe A) comme algorithme de classification binaire. Le classifieur généralisé ainsi obtenu est soumis à un ensemble de vidéos d'endomicroscopie sur lesquelles des polypes du colon sont à classer. La méthode proposée donne des résultats légèrement supérieurs aux autres méthodes de généralisation de classifieurs binaires classiques (un contre un, un contre tous, ...).

Ensuite, nous avons proposé une méthode permettant l'analyse d'images tridimensionnelles de cortex de souris issus de microscopie confocale (Cf. Chapitre 3). La chaîne de traitement proposée consiste à analyser les neurones à deux niveaux de grossissement différents afin d'en extraire des grandeurs caractéristiques spécifiques décrivant leur morphologie. Une étape de mise en correspondance invariante par translation et par rotation a été proposée afin d'établir une carte

de correspondance des neurones et de combiner les information extraites sur les deux types d'image.

La dernière partie (Cf. Chapitre 4) de cette thèse est consacrée à l'étude d'images d'histologie digitale de carcinomes rénaux. Alors que les approches proposées dans la littérature se concentrent, de façon quasi-exclusive, sur l'étude des noyaux des cellules tumorales, nous proposons une analyse à grande échelle de l'architecture vasculaire qui est un critères important intervenant dans l'analyse de telles images par les médecins. La méthode décrite permet de calculer le graphe représentant le réseau vasculaire du carcinome grâce à une analyse à plusieurs niveaux de grossissement. En effet, l'échantillon est d'abord analysé dans son ensemble à faible grossissement dans le but de détecter les régions tumorales, puis à un grossissement maximal afin d'extraire le réseau vasculaire local. Les informations ainsi calculées sur des sous-parties de l'image sont enfin combinées dans le but d'obtenir une information synthétique de l'ensemble de l'échantillon observé permettant d'en décrire son architecture. Le travail réalisé permet une analyse du tissu prélevé à un niveau de détail et une échelle qui ne seraient pas envisageables de manière manuelle.

5.2 PERSPECTIVES

De manière générale, les résultats obtenus, en particulier les segmentations des Chapitres 3 et 4, mériteraient d'être évalués de manière plus poussée, même si cela demande au préalable un certain nombre de vérités terrain réalisées par des experts.

5.2.1 Extension d'un classifieur binaire

Puisque le cadre général d'extension du classifieur binaire proposé dépend de deux algorithmes (un classifieur binaire et une méthode de bipartionnement de graphe pondéré), il serait intéressant d'étendre l'étude réalisée en sélectionnant divers algorithmes pour pouvoir comparer les performances (en temps de calcul comme en précision).

5.2.2 Analyse de la morphologie de neurones de souris

Dans le cadre de l'analyse de la morphologie de neurones de souris, il serait intéressant d'appliquer la méthode proposée à un ensemble de données plus important afin de pouvoir calculer des statistiques plus significatives.

En ce qui concerne la méthode d'appariement de points, la généralisation au cas où le facteur d'échelle entre les deux types d'images est inconnu constitue une perspective importante.

5.2.3 Analyse de l'architecture d'un carcinome rénal

Les perspectives d'amélioration de la méthode d'analyse d'images de carcinomes rénaux proposée sont nombreuses dans la mesure où l'histologie digitale – et en particulier l'analyse du réseau vasculaire de tumeurs – est une piste qui n'a reçu que peu d'attention jusqu'alors.

En effet, au niveau du grossissement le plus faible, la segmentation des régions tumorales, bien que ne demandant pas l'intervention d'un expert, pourrait gagner en précision pour devenir suffisamment fiable et ainsi éviter d'avoir recours à une segmentation manuelle. La deuxième piste à envisager concerne la construction du graphe : la segmentation et l'extraction du squelette peuvent encore être parfaites dans le sens où des problèmes de sous- et sur-détection persistent.

Enfin, une interprétation plus poussée du graphe extrait pourrait être envisagée afin d'aboutir à des arguments biologiques de classification plus précis. À titre d'exemple, l'étude de l'orientation des branches pourrait donner quelques indications quant à la direction de propagation du réseau vasculaire.

Troisième partie

Annexes

A SÉPARATEURS À VASTE MARGE

Sommaire

A.1 Classifieur à marge optimale (SVM) 89
A.2 Dualité lagrangienne 92
A.2.1 Du problème primal au problème dual 92
A.2.2 Le cas des SVM 94
A.2.3 Classification non-linéaire 95
A.3 Cas non-séparable : la marge souple 96

Dans cette partie, comme précedemment dans le document, nous utiliserons l'acronyme SVM pour désigner les machines à vecteurs de support, du nom de la méthode en anglais : Support Vector Machines [Vapnik, 2013]. Nous utiliserons également l'anglicisme *classifieur* comme raccourci pour « fonction prédisant la classe d'une donnée d'entrée ».

L'approche SVM [Cortes et Vapnik, 1995] est parmi les méthodes d'apprentissage supervisé les plus couramment utilisées [Burges, 1998 ; Joachims, 1998]. Il s'agit d'un classifieur binaire qui a peu de paramètres à régler : sans sans noyau (Section A.2.3) ni marge souple (Section A.3), il n'y en a tout simplement pas. Par ailleurs, comme tout classifieur supervisé, ses paramètres peuvent être ajsutés automatiquement par validation croisée [Kohavi et al., 1995]. Notons qu'il existe plusieurs extensions dites multi-classes (nombre de classes supérieur ou égal à trois, Cf. Chapitre 2).

A.1 CLASSIFIEUR À MARGE OPTIMALE (SVM)

Avant d'exposer les intuitions et le principe des SVM, voici quelques notations et définitions valables dans cette annexe. Soit un ensemble de données

$$\mathcal{S} = \left\{ (x^{(i)}, y^{(i)}), x^{(i)} \in \mathbb{R}^d, y^{(i)} \in \{-1, +1\}, 1 \leqslant i \leqslant m \right\}$$

et un classifieur linéaire pour un problème de classification binaire sur des caractéristiques $x \in \mathbb{R}^n$ labélisées par $y \in \{-1, +1\}$, où -1 et $+1$ représentent les deux classes. Dans un premier temps, on suppose que les deux ensembles de points correspondant à ces labels sont linéairement séparables, comme illustré sur la Figure 45.

Le principe d'un classifieur SVM est de chercher l'hyperplan séparateur offrant la plus grande marge (en un sens qui sera défini plus

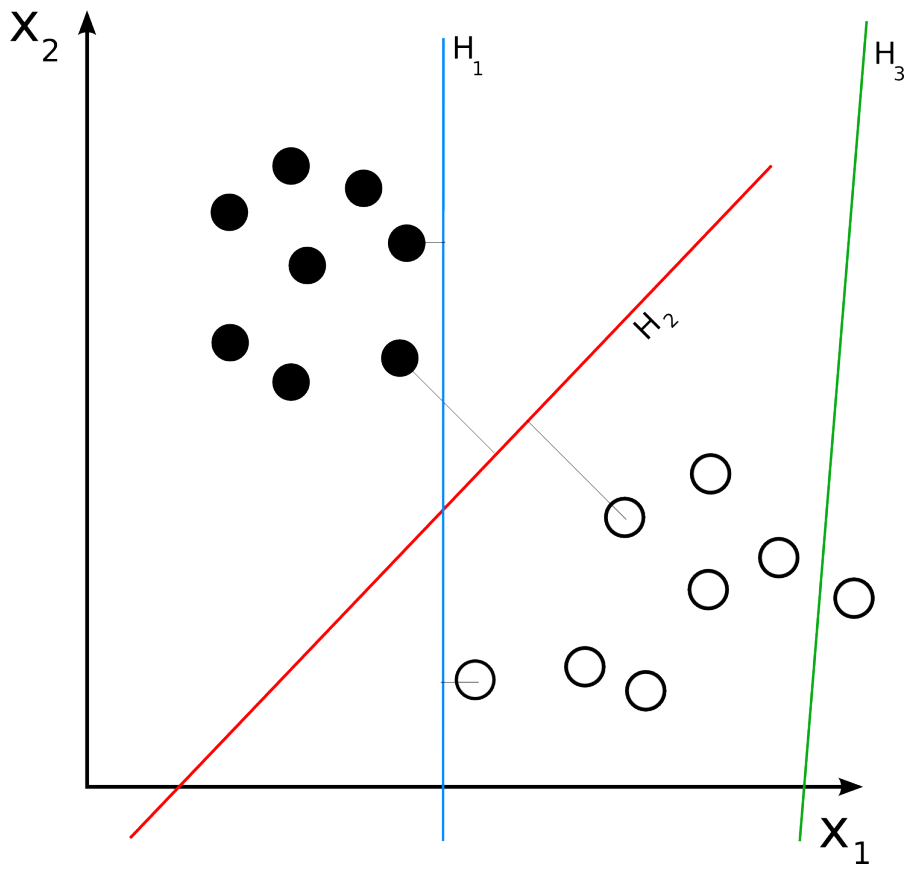


FIGURE 45 – Un exemple simple de deux ensembles de données linéairement séparables. H_1 n'est pas un hyperplan séparateur, H_2 est un hyperplan séparateur, and H_3 est l'hyperplan séparateur offrant la plus grande marge.

loin) entre les points labélisés $+1$ et ceux labélisés -1. Cet hyperplan optimal $\mathcal{H}$ peut être défini par l'équation suivante :

$$\langle w, x \rangle - b = 0,$$

où $w \in \mathbb{R}^d$ est un vecteur normal à l'hyperplan et $b \in \mathbb{R}$.

L'objectif étant de maximiser la marge entre les deux ensembles de points, nous considérons les deux hyperplans $\mathcal{H}^-$ et $\mathcal{H}^+$, parallèles à $\mathcal{H}$, respectivement définis par :

$$\langle w, x \rangle - b = -1$$

et :

$$\langle w, x \rangle - b = +1.$$

Ces deux hyperplans délimitent la marge de laquelle nous voulons exclure tout point (voir Figure 46), ce qui se traduit par, pour tout $i \in [\![1, m]\!]$:

$$y^{(i)} = -1 \implies \left\langle w, x^{(i)} \right\rangle - b \leqslant -1$$

et

$$y^{(i)} = +1 \implies \left\langle w, x^{(i)} \right\rangle - b \leqslant +1.$$

Les deux conditions précédentes peuvent être résumées en la suivante :

$$\forall i \in [\![1, m]\!], y^{(i)} \left(\left\langle w, x^{(i)} \right\rangle - b \right) \geqslant 1.$$

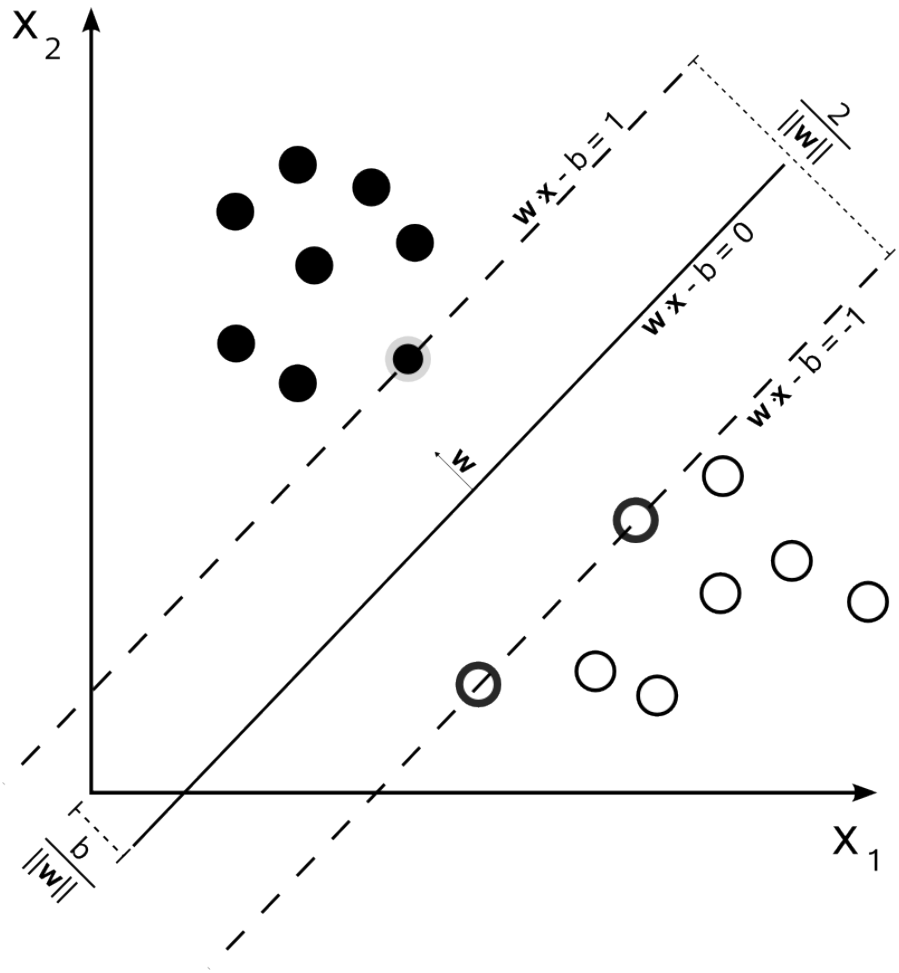


FIGURE 46 – Le classifieur SVM dans un cas bidimensionnel linéairement séparable.

Puisque la distance entre les deux hyperplans $\mathcal{H}^-$ et $\mathcal{H}^+$ est égale à $\frac{2}{\|w\|}$ [SCHOLKOPF et SMOLA, 2001], nous devons résoudre le problème d'optimisation suivant :

$$\begin{aligned} \underset{w \in \mathbb{R}^n, b \in \mathbb{R}}{\text{maximiser}} \quad & \frac{2}{\|w\|} \\ \text{avec} \quad & y^{(i)}(\left\langle w, x^{(i)} \right\rangle - b) \geqslant 1, \quad i = 1, \ldots, m, \end{aligned} \tag{20}$$

ce qui est équivalent à :

$$\begin{aligned} \underset{w\in\mathbb{R}^n, b\in\mathbb{R}}{\text{minimiser}} \quad & \frac{1}{2}\|w\|^2 \\ \text{avec} \quad & y^{(i)}(\left\langle w, x^{(i)}\right\rangle - b) \geqslant 1, \quad i = 1, \ldots, m. \end{aligned} \tag{21}$$

La formulation (21) est préférée à (20) car elle est plus facile à résoudre [Scholkopf et Smola, 2001].

A.2 DUALITÉ LAGRANGIENNE

Bien que le problème d'optimisation établi en Section A.1 puisse être résolu dans un cadre classique d'optimisation convexe, il est préférable d'utiliser la dualité lagrangienne pour profiter pleinement du contexte correspondant aux SVM.

A.2.1 Du problème primal au problème dual

Laissons pour l'instant de côté le problème des SVM et considérons le problème d'optimisation contrainte suivant :

$$\begin{aligned} \underset{w}{\text{minimiser}} \quad & f(w) \\ \text{avec} \quad & g_i(w) \leqslant 0, \quad i = 1, \ldots, k \\ & h_i(w) = 0, \quad i = 1, \ldots, l. \end{aligned} \tag{$\mathcal{P}$}$$

Nous qualifierons de primal ce problème. Un tel problème peut se résoudre par une approche classique dite de dualité lagrangienne en utilisant le lagrangien défini comme suit.

Définition A.1 (Lagrangien)**.** *Le lagrangien d'une fonction* f *a pour expression*

$$\mathcal{L}(w, \alpha, \beta) = f(w) + \sum_{i=1}^{k} \alpha_i g_i(w) + \sum_{i=1}^{l} \beta_i h_i(w),$$

où les α_i *et les* β_i *sont les multiplicateurs de Lagrange.*

Définissons

$$\theta_{\mathcal{P}}(w) = \max_{\alpha\geqslant 0, \beta} \mathcal{L}(w, \alpha, \beta).$$

($\mathcal{P}$ désigne le qualificatif « primal ».) Il en découle que

$$\theta_{\mathcal{P}}(w) = \begin{cases} f(w), & \text{si } w \text{ satisfait les constraintes primales} \\ \infty, & \text{sinon.} \end{cases}$$

Ainsi, le problème d'optimisation

$$\min_{w} \theta_{\mathcal{P}}(w) = \min_{w} \max_{\alpha\geqslant 0, \beta} \mathcal{L}(w, \alpha, \beta) \tag{$\mathcal{P}'$}$$

a les mêmes solutions que le problème primal. On note $p^\star = \min_w \theta_{\mathcal{P}}(w)$ sa valeur optimale.

Considérons maintenant la quantité

$$\theta_{\mathcal{D}}(\alpha, \beta) = \min_w \mathcal{L}(w, \alpha, \beta)$$

où $\mathcal{D}$ désigne le qualificatif « dual ». Nous pouvons alors poser le problème d'optimisation dual :

$$\max_{\alpha \geqslant 0, \beta} \theta_{\mathcal{D}}(\alpha, \beta) \tag{$\mathcal{D}$}$$

et la valeur correspondante de ce problème :

$$d^\star = \max_{\alpha \geqslant 0, \beta} \theta_{\mathcal{D}}(\alpha, \beta).$$

Faisons à présent le lien entre $p^\star$ et $d^\star$. Pour cela, dans la suite, nous supposerons que :

— f est convexe,
— Pour tout $i = 1, \ldots, k$, g_i est convexe,
— Pour tout $i = 1, \ldots, l$, h_i est affine,
— Pour tout $i = 1, \ldots, k$, les contraintes sur g_i sont (fortement) réalisables [1].

Le théorème suivant fait le lien entre problème primal et problème dual.

Théorème A.2.1. *Sous les conditions précédentes, il existe $w^\star$, $\alpha^\star$ et $\beta^\star$ tels que $w^\star$ est la solution du problème primal, $\alpha^\star$ et $\beta^\star$ sont les solutions du problème dual, et $p^\star = d^\star = \mathcal{L}(w^\star, \alpha^\star, \beta^\star)$.*

En conséquence, résoudre le problème primal permet d'obtenir immédiatement la solution du problème dual, et vice-versa. De plus, les conditions suivantes dites de KKT pour « Karush, Kuhn et Tucker » (et le theorème associé) offrent le moyen de calculer effectivement les valeurs optimales $w^\star$, $\alpha^\star$ et $\beta^\star$:

Définition A.2 (Conditions de KKT [Kuhn, 2014]). *$w^\star$, $\alpha^\star$ et $\beta^\star$ satisfont les conditions de KKT si et seulement si :*

$$\frac{\partial}{\partial w_i} \mathcal{L}(w^\star, \alpha^\star, \beta^\star) = 0, \; i = 1, \ldots, n \tag{KKT1}$$

$$\frac{\partial}{\partial \beta_i} \mathcal{L}(w^\star, \alpha^\star, \beta^\star) = 0, \; i = 1, \ldots, l \tag{KKT2}$$

$$\alpha^\star g_i(w^\star) = 0, \; i = 1, \ldots, k \tag{KKT3}$$

$$g_i(w^\star) \leqslant 0, \; i = 1, \ldots, k \tag{KKT4}$$

$$\alpha^\star \geqslant 0, \; i = 1, \ldots, k \tag{KKT5}$$

Théorème A.2.2 (KKT). *$w^\star$ est la solution du problème primal, $\alpha^\star$ et $\beta^\star$ sont les solutions du problème dual, et $p^\star = d^\star = L(w^\star, \alpha^\star, \beta^\star)$ si et seulement si $w^\star$, $\alpha^\star$ et $\beta^\star$ satisfont les conditions de KKT.*

1. Cela signifie que pour tout $i = 1, \ldots, k$, il existe w_i tel que $g_i(w_i) < 0$

A.2.2 Le cas des SVM

En Section A.1, nous avons posé le problème d'optimisation suivant permettant de trouver le classifieur à marge optimale :

$$\begin{aligned} \underset{w,b}{\text{minimize}} \quad & \frac{1}{2}\|w\|^2 \\ \text{avec} \quad & y^{(i)}(w^T x^{(i)} + b) \geqslant 1, \quad i = 1, \ldots, m. \end{aligned} \tag{22}$$

Notons que (22) peut être interprété comme le problème primal ($\mathcal{P}'$) avec :

$$f(w) = \frac{1}{2}\|w\|^2 \tag{23}$$

$$g_i(w) = -y^{(i)}(w^T x^{(i)} + b) + 1 \leqslant 0, \; i = 1, \ldots, l \tag{24}$$

Il n'y a pas dans ce cas de contraintes d'égalité définies par des fonctions h_i. De plus, les fonctions g_i satisfont les conditions du théorème A.2.1, ce qui assure l'existence de $w^\star$.

Le lagrangien de ce problème d'optimisation primal est

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\| - \sum_{i=1}^{m} \alpha_i \left[y^{(i)}(w^T x^{(i)} + b - 1 \right].$$

Pour obtenir la forme duale correspondant au problème (22), nous devons déterminer $\theta_{\mathcal{D}}$, qui peut être calculé en minimisant $\mathcal{L}(w, b, \alpha)$ par rapport à w et b (voir la condition (KKT1)). Pour ce faire, les dérivées de $\mathcal{L}$ par rapport à w et b sont mises à zéro :

$$\frac{\partial}{\partial b}\mathcal{L}(w, b, \alpha) = \sum_{i=1}^{m} \alpha_i y^{(i)} = 0 \tag{25}$$

et

$$\nabla_w \mathcal{L}(w, b, \alpha) = w - \sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)} = 0 \tag{26}$$

ce qui implique que :

$$w = \sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)}. \tag{27}$$

Après simplification [Scholkopf et Smola, 2001], cela conduit à l'expression suivante du lagrangien :

$$\mathcal{L}(w, b, \alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j (x^{(i)})^T x^{(j)}.$$

En rassemblant les éléments précédents, nous aboutissons au problème d'optimisation dual suivant :

$$\begin{aligned} \underset{\alpha}{\text{maximiser}} \quad & W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j \left\langle x^{(i)}, x^{(j)} \right\rangle \\ \text{avec} \quad & \alpha_i \geqslant 0, \ i = 1, \ldots, m \\ & \sum_{i=1}^{m} \alpha_i y^{(i)} = 0. \end{aligned} \tag{28}$$

En outre, les conditions de KKT (définition A.2) étant satisfaites [Scholkopf et Smola, 2001], nous pouvons résoudre le problème d'optimisation dual en lieu et place du problème primal.

Il existe plusieurs algorithmes d'optimisation permettant de résoudre le problème SVM dual. L'un des plus couramment employé est l'algorithme SMO (pour « Sequential Minimal Optimization ») [J. Platt et al., 1999].

A.2.3 Classification non-linéaire

L'un des avantages du problème dual par rapport au problème primal est qu'il peut être écrit en termes de produits scalaires $\langle x^{(i)}, x^{(j)} \rangle$. Par conséquent, dans le cas où les ensembles de points ne sont pas linéairement séparables, il serait intéressant de pouvoir reconsidérer le problème dans un espace de dimension supérieure (voire infinie) dans lequel les données seraient linéairement séparables. Dans ce cas, chaque donnée $x^{(i)}$ serait remplacée par $\phi(x^{(i)})$ où ϕ est une transformation vers un espace de plus grande dimension (parfois appelée *espace de redescription*). Dans (22), le produit scalaire $\langle x^{(i)}, x^{(j)} \rangle$ serait alors remplacé par

$$K(\phi(x^{(i)}), \phi(x^{(j)})) = \left\langle \phi(x^{(i)}), \phi(x^{(j)}) \right\rangle$$

où K est le noyau correspondant à la transformation ϕ. On parle alors de *noyau valide*.

Définition A.3 (Noyau valide)**.** *Un noyau* K *est* valide *si, et seulement si, il existe une transformation* ϕ *telle que* $K(x^{(i)}, x^{(j)}) = \phi(x^{(i)})^T \phi(x^{(j)})$.

Une propriété importante des noyaux est que l'on peut les utiliser sans pour autant connaître la transformation correspondante ϕ. Le Théorème A.2.3 donne une condition nécessaire et suffisante de validité.

Théorème A.2.3 (Mercer [Scholkopf et Smola, 2001])**.** *Soit* $K : \mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}$ *un noyau.* K *est un noyau valide si, et seulement si,* K *est symétrique et semi-définie positive.*

Nous pouvons alors utiliser un noyau K défini par une transformation ϕ ou satisfaisant le théorème de Mercer. Parmi les noyaux les plus utilisés, on retrouve le noyau polynômial de degré k :

$$K(x^{(i)}, x^{(j)}) = ((x^{(i)})^T x^{(j)} + c)^k$$

et le noyau gaussien :

$$K(x^{(i)}, x^{(j)}) = \exp\left(-\frac{\left\|x^{(i)} - x^{(j)}\right\|^2}{2\sigma^2}\right)$$

(qui correspond à une transformation vers un espace de dimension infinie [Scholkopf et Smola, 2001]).

A.3 CAS NON-SÉPARABLE : LA MARGE SOUPLE

Il arrive que les données ne soient pas séparables, même en utilisant un noyau. Cela peut être d'autant plus vrai lorsque celles-ci sont bruitées et/ou lorsqu'elles présentent des observations aberrantes. Dans un tel cas, le problème (21) peut être modifié en relâchant la contrainte sur l'ensemble d'apprentissage par l'ajout de variables « ressort » $\xi_i \geqslant 0$ afin de tolérer, dans une certaine mesure, des mauvaises classifications. Cela revient à trouver un compromis entre l'épaisseur de la marge et la proportion de données bien classifiées. On parle alors de « marge douce » (par opposition à la « marge dure » du problème (22)).

Les contraintes du problème (22) s'écrivent alors, pour tout $i = 1, \ldots, m$:

$$y^{(i)}(\left\langle w, x^{(i)} \right\rangle - b) \geqslant 1 - \xi_i.$$

Ensuite, nous pénalisons les variables ξ_i non nulles, ce qui donne le problème d'optimisation primal suivant :

$$\begin{aligned} \underset{w,b,\xi}{\text{minimiser}} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m} \xi_i \\ \text{avec} \quad & y^{(i)}(w^T x^{(i)} + b) \geqslant 1 - \xi_i, \quad i = 1, \ldots, m. \end{aligned} \tag{29}$$

où C est un hyperparamètre d'autant plus grand que l'on souhaite pénaliser les données mal classifiées. En ce qui concerne le problème dual, il devient :

$$\begin{aligned} \underset{\alpha}{\text{maximiser}} \quad & W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j \left\langle x^{(i)}, x^{(j)} \right\rangle \\ \text{avec} \quad & 0 \leqslant \alpha_i \leqslant C, \ i = 1, \ldots, m \\ & \sum_{i=1}^{m} \alpha_i y^{(i)} = 0. \end{aligned} \tag{30}$$

Ce problème d'optimisation est très proche de (28) puisque la constante C n'apparaît que comme borne supérieure des multiplicateurs de Lagrange.

B L'ALGORITHME DE COUPE MINIMALE

Sommaire

B.1 Notations et définitions 99
B.2 The Algorithm 99
B.3 Équilibrage de la coupe 100
B.4 Temps de calcul 101

B.1 NOTATIONS ET DÉFINITIONS

Commençons par définir les concepts et les notations nécessaires au fonctionnement de l'algorithme de coupe minimale. Soit $G = (V, E)$ un graphe pondéré où V est l'ensemble des nœuds et E l'ensemble des arêtes (pondérées).

Définition B.1 (s-t-coupe)**.** *Une* s-t-*coupe (*s-t-cut *en anglais)* $C = (S, T)$ *est une partition de* V *en deux sous-graphes non-vides* S *et* T *tels que* $s \in S$ *et* $t \in T$.

Définition B.2 (capacité)**.** *La capacité* c *d'une* s-t-*coupe* $C = (S, T)$ *est définie par* $c(S, T) = \sum_{(u,v) \in S \times T} c_{uv}$.

Définition B.3 (problème de s-t-coupe minimale)**.** *Une* s-t-*coupe minimale consiste à déterminer* S *et* T *minimisant la capacité de la* s-t-*coupe* $C = (S, T)$.

B.2 THE ALGORITHM

D'après [Stoer et Wagner, 1997], l'algorithme de coupe minimale MinCutPhase présenté dans cette annexe permet d'obtenir une coupe de G de capacité minimale. Il est basé sur le théorème de Stoer-Wagner :

Théorème B.2.1 (Stoer-Wagner)**.** *Soit* s *et* t *deux nœuds du graphe* $G = (V, E)$. *Soit* $G/\{s, t\}$ *le graphe obtenu en fusionnant* s *et* t. *Une coupe minimale de* G *peut être obtenue en prennant la plus petite coupe entre une* s-t-*coupe minimale et une coupe minimale de* $G/\{s, t\}$.

En vertue de ce théorème, il suffit donc de pouvoir calculer une s-t-coupe de G pour réaliser la partition souhaitée. L'algorithme 4,

Algorithme 4 L'algorithme MinCutPhase

```
1: procedure MinCutPhase(G, a)
2:     A ← {a}
3:     while A ≠ V do
4:         Ajouter à A son plus proche voisin
5:         Stocker la coupe courante et réduire G en fusionnant les
   deux derniers nœuds ajoutés
6:     end while
7:     Renvoyer la réduction de G et la coupe courante.
8: end procedure
```

également tiré de [Stoer et Wagner, 1997], permet de calculer une telle coupe.

Il est possible de prouver, par induction (Cf. [Stoer et Wagner, 1997]), que MinCutPhase renvoie s-t-coupe minimale de G. L'algorithme contracte ensuite les nœuds s et t en un seul nœud. Dès lors, il ne reste qu'à stocker la s-t-coupe minimale et à itérer le procédé jusqu'à ce que G n'ait plus qu'une arête pour trouver la coupe minimale globale. C'est ce que l'Algorithme 5 permet de faire.

Algorithme 5 L'algorithme MinCut

```
1: procedure MinCut(G, a)
2:     A ← {a}
3:     while |V| > 1 do
4:         CP ← MinCutPhase(G, a)
5:         if La coupe courante est plus faible que la coupe minimale
   then
6:             La coupe courante devient la coupe minimale
7:         end if
8:         Renvoyer la partition minimale
9:     end while
10: end procedure
```

On peut remarquer que les Algorithmes 4 et 5 prennent un nœud a de G entrée. Le choix de a n'a pas d'impact sur le résultat de la coupe renvoyée par MinCut.

B.3 ÉQUILIBRAGE DE LA COUPE

Dans cette section, nous proposons une manière de forcer l'équilibrage de la coupe obtenue. Pour cela, l'Algorithme 4 est modifié : parmi les coupes stockées, celle qui sera sélectionnée est la partition équilibrée pour laquelle la valeur de la coupe est minimale.

Après une telle modification, chaque appel de MinCutPhase retourne une partition équilibrée et par conséquent la coupe donnée par MinCut sera, elle aussi, équilibrée.

B.4 TEMPS DE CALCUL

[Stoer et Wagner, 1997] montre que la complexité de l'algorithme MinCut est polynomiale :

Proposition B.4.1 (complexité de MinCut)**.** *La complexité de* MinCut *pour partitionner* $G = (V, E)$ *est* $O\left(|V||E| + |V|^2 \log |V|\right)$.

Dans le cas de GCSVM (Cf. Chapitre 2), le nombre de nœuds est $|V| = m$ où m est le nombre de classe et le nombre d'arêtes est $|E| = O(m^2)$ puisque le graphe est complet et non orienté. Par conséquent, la formule de la proposition précédente donne alors une complexité de $O\left(m^3 + m^2 \log m\right) = O\left(m^3\right)$ pour MinCut. En pratique, cette complexité importe peu puisque le temps de calcul des coupes de graphes permettant de construire l'arbre de classifieurs binaires est largement négligeable devant le temps mis pour réaliser l'apprentissage des classifieurs.

BIBLIOGRAPHIE

Abràmoff, Michael D, Paulo J Magalhães et Sunanda J Ram

2004 "Image processing with ImageJ", *Biophotonics international*, 11, 7, p. 36–42. (Cf. p. 1.)

Aiger, Dror et Klara Kedem

2009 "Geometric pattern matching for point sets in the plane under similarity transformations", *Information Processing Letters*, 109, 16, p. 935–940. (Cf. p. 54.)

2010 "Approximate input sensitive algorithms for point pattern matching", *Pattern Recognition*, 43, 1, p. 153–159. (Cf. p. 54.)

Allwein, Erin L, Robert E Schapire et Yoram Singer

2001 "Reducing multiclass to binary : A unifying approach for margin classifiers", *The Journal of Machine Learning Research*, 1, p. 113–141. (Cf. p. 10.)

Alsheh Ali, Maya

2015 *Analyse statistique de populations pour l'interprétation d'images histologiques*, thèse de doct., Université Paris-Descartes. (Cf. p. 65, 69.)

Aly, Mohamed

2005 "Survey on multiclass classification methods", *Neural Netw*, p. 1–9. (Cf. p. 10.)

André, Barbara, Tom Vercauteren, Anna M Buchner, Michael B Wallace et Nicholas Ayache

2011 "A smart atlas for endomicroscopy using automated video retrieval", *Medical image analysis*, 15, 4, p. 460–476. (Cf. p. 25, 26.)

Andreev, Konstantin et Harald Racke

2006 "Balanced graph partitioning", *Theory of Computing Systems*, 39, 6, p. 929–939. (Cf. p. 26.)

Apou, Gregory, Benoit Naegel, Germain Forestier, Friedrich Feuerhake et Cédric Wemmert

2014 "Fast segmentation for texture-based cartography of Whole Slide Images", in *Proceedings of 9th International Conference on Computer Vision Theory and Applications (VISAPP) : 5-8 January 2014 ; Lisbon*. (Cf. p. 60.)

ARLOT, Sylvain, Alain CELISSE et al.

2010 "A survey of cross-validation procedures for model selection", *Statistics surveys*, 4, p. 40–79. (Cf. p. 20.)

ATHITSOS, Vassilis, Alexandra STEFAN, Quan YUAN et Stan SCLAROFF

2007 "ClassMap : Efficient multiclass recognition via embeddings", in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, IEEE, p. 1–8. (Cf. p. 10.)

AURENHAMMER, Franz

1991 "Voronoi diagrams—a survey of a fundamental geometric data structure", *ACM Computing Surveys (CSUR)*, 23, 3, p. 345–405. (Cf. p. 62.)

BANCROFT, John D et Marilyn GAMBLE

2008 *Theory and practice of histological techniques*, Elsevier Health Sciences. (Cf. p. 57.)

BANKMAN, Isaac

2008 *Handbook of medical image processing and analysis*, academic press. (Cf. p. 1.)

BAY, Herbert, Andreas ESS, Tinne TUYTELAARS et Luc VAN GOOL

2008 "Speeded-up robust features (SURF)", *Computer vision and image understanding*, 110, 3, p. 346–359. (Cf. p. 2.)

BELONGIE, Serge, Jitendra MALIK et Jan PUZICHA

2002 "Shape matching and object recognition using shape contexts", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24, 4, p. 509–522. (Cf. p. 2.)

BENGIO, Yoshua, Aaron COURVILLE et Pierre VINCENT

2013 "Representation learning : A review and new perspectives", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35, 8, p. 1798–1828. (Cf. p. 9.)

BOTTOU, Léon, Jason WESTON et Gökhan H BAKIR

2004 "Breaking SVM complexity with cross-training", in *Advances in neural information processing systems*, p. 81–88. (Cf. p. 20.)

BREIMAN, Leo

2001 "Random forests", *Machine learning*, 45, 1, p. 5–32. (Cf. p. 9, 24.)

BURGES, Christopher JC

1998 "A tutorial on support vector machines for pattern recognition", *Data mining and knowledge discovery*, 2, 2, p. 121–167. (Cf. p. 89.)

CHANG, Chih-Chung et Chih-Jen LIN

2011 "LIBSVM : A library for support vector machines", *ACM Transactions on Intelligent Systems and Technology*, 2 (3 2011), Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm, 27 :1–27 :27. (Cf. p. 20.)

CHEN, Yangchi, Melba M CRAWFORD et Joydeep GHOSH

2004 "Integrating support vector machines in a hierarchical output space decomposition framework", in *Geoscience and Remote Sensing Symposium, 2004. IGARSS'04. Proceedings. 2004 IEEE International*, IEEE, t. 2, p. 949–952. (Cf. p. 11.)

CHEVILLE, John C, Christine M LOHSE, Horst ZINCKE, Amy L WEAVER et Michael L BLUTE

2003 "Comparisons of outcome and prognostic features among histologic subtypes of renal cell carcinoma", *The American journal of surgical pathology*, 27, 5, p. 612–624. (Cf. p. 56.)

CHOI, Hyohoon, Samuel CHENG, Qiang WU, Kenneth R CASTLEMAN et AlanC BOVIK

2006 "Extended depth-of-field using adjacent plane deblurring and MPP wavelet fusion for microscope images", in *Biomedical Imaging : Nano to Macro, 2006. 3rd IEEE International Symposium on*, IEEE, p. 774–777. (Cf. p. 1.)

CONDURACHE, Alexandru-Paul et Til AACH

2005 "Vessel segmentation in angiograms using hysteresis thresholding", in *IAPR Conference on Machine Vision Applications*, Citeseer. (Cf. p. 71.)

CORTES, Corinna, Mehryar MOHRI et Afshin ROSTAMIZADEH

2008 "Learning sequence kernels", in *Machine Learning for Signal Processing, 2008. MLSP 2008. IEEE Workshop on*, IEEE, p. 2–8. (Cf. p. 26.)

CORTES, Corinna et Vladimir VAPNIK

1995 "Support-vector networks", *Machine learning*, 20, 3, p. 273–297. (Cf. p. 89.)

CRAMMER, Koby et Yoram SINGER

2002 "On the algorithmic implementation of multiclass kernel-based vector machines", *The Journal of Machine Learning Research*, 2, p. 265–292. (Cf. p. 9.)

DALAL, Navneet et Bill TRIGGS

2005 "Histograms of oriented gradients for human detection", in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, IEEE, t. 1, p. 886–893. (Cf. p. 2.)

Dantzig, George Bernard

1998 *Linear programming and extensions*, Princeton university press. (Cf. p. 47.)

Dell'Amico, Mauro et Marco Trubian

1998 "Solution of large weighted equicut problems", *European journal of operational research*, 106, 2, p. 500–521. (Cf. p. 18.)

Delpretti, Saskia, Florian Luisier, Sathish Ramani, Thierry Blu et Michael Unser

2008 "Multiframe sure-let denoising of timelapse fluorescence microscopy images", in *Biomedical Imaging : From Nano to Macro, 2008. ISBI 2008. 5th IEEE International Symposium on*, IEEE, p. 149–152. (Cf. p. 1.)

Demirkaya, Omer, Musa H Asyali et Prasanna K Sahoo

2008 *Image processing with MATLAB : applications in medicine and biology*, CRC Press. (Cf. p. 1.)

Descombes, Xavier

2013 *Stochastic geometry for image analysis*, John Wiley & Sons. (Cf. p. 41.)

Deselaers, Thomas, Lexi Pimenidis et Hermann Ney

2008 "Bag-of-visual-words models for adult image classification and filtering", in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, IEEE, p. 1–4. (Cf. p. 25.)

Dietterich, Thomas G. et Ghulum Bakiri

1995 "Solving multiclass learning problems via error-correcting output codes", *Journal of artificial intelligence research*, p. 263–286. (Cf. p. 10.)

Dijkstra, Edsger W

1959 "A note on two problems in connexion with graphs", *Numerische mathematik*, 1, 1, p. 269–271. (Cf. p. 42.)

Ding, Y et T Zheng

2007 "Tree-Based Integration of One-versus-Some (OVS) Classifiers for Multiclass Classification", in *Joint Statistical Meeting*.

Duff, Iain S, Kamer Kaya et Bora Uçcar

2011 "Design, implementation, and analysis of maximum transversal algorithms", *ACM Transactions on Mathematical Software (TOMS)*, 38, 2, p. 13. (Cf. p. 44, 46.)

Eitrich, Tatjana et Bruno Lang

2006 "Efficient optimization of support vector machine learning parameters for unbalanced datasets", *Journal of computational and applied mathematics*, 196, 2, p. 425–436. (Cf. p. 26.)

ELDIN, Ahmed Gamal, Xavier DESCOMBES, Guillaume CHARPIAT et Josiane ZERUBIA

2012 "Multiple birth and cut algorithm for multiple object detection", *Journal of Multimedia Processing and Technologies*. (Cf. p. 41.)

EVEN, Guy, Joseph NAOR, Satish RAO et Baruch SCHIEBER

1999 "Fast approximate graph partitioning algorithms", *SIAM Journal on Computing*, 28, 6, p. 2187–2214. (Cf. p. 16.)

FEDER, Tomas, Pavol HELL, Sulamita KLEIN et Rajeev MOTWANI

1999 "Complexity of graph partition problems", in *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, ACM, p. 464–472. (Cf. p. 16.)

FELDMANN, Andreas Emil

2012 "Fast balanced partitioning is hard even on grids and trees", in *Mathematical Foundations of Computer Science 2012*, Springer, p. 372–382. (Cf. p. 16.)

FENG, Guoping, Rebecca H MELLOR, Michael BERNSTEIN, Cynthia KELLER-PECK, Quyen T NGUYEN, Mia WALLACE, Jeanne M NERBONNE, Jeff W LICHTMAN et Joshua R SANES

2000 "Imaging neuronal subsets in transgenic mice expressing multiple spectral variants of GFP", *Neuron*, 28, 1, p. 41–51. (Cf. p. 33.)

FRANGI, Alejandro F, Wiro J NIESSEN, Koen L VINCKEN et Max A VIERGEVER

1998 "Multiscale vessel enhancement filtering", in *Medical Image Computing and Computer-Assisted Interventation—MICCAI'98*, Springer, p. 130–137. (Cf. p. 71, 82.)

FREY, Peter W et David J SLATE

1991 "Letter recognition using Holland-style adaptive classifiers", *Machine Learning*, 6, 2, p. 161–182. (Cf. p. 23.)

FUHRMAN, Susan A, Larry C LASKY et Catherine LIMAS

1982 "Prognostic significance of morphologic parameters in renal cell carcinoma." *The American journal of surgical pathology*, 6, 7, p. 655–664. (Cf. p. 79.)

GAGE, Fred H

2000 "Mammalian neural stem cells", *Science*, 287, 5457, p. 1433–1438. (Cf. p. 31.)

GARCIA-PEDRAJAS, Nicolas et Domingo ORTIZ-BOYER

2006 "Improving multiclass pattern recognition by the combination of two strategies", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28, 6, p. 1001–1006. (Cf. p. 10.)

Gevers, Theo et Arnold WM Smeulders

2000 "Pictoseek : Combining color and shape invariant features for image retrieval", *Image Processing, IEEE Transactions on*, 9, 1, p. 102–119. (Cf. p. 65.)

Golub, Todd R, Donna K Slonim, Pablo Tamayo, Christine Huard, Michelle Gaasenbeek, Jill P Mesirov, Hilary Coller, Mignon L Loh, James R Downing, Mark A Caligiuri et al.

1999 "Molecular classification of cancer : class discovery and class prediction by gene expression monitoring", *science*, 286, 5439, p. 531–537. (Cf. p. 56.)

Gonzalez, Rafael C, Richard Eugene Woods et Steven L Eddins

2004 *Digital image processing using MATLAB*, Pearson Education India. (Cf. p. 60, 82.)

Goode, Adam, Benjamin Gilbert, Jan Harkes, Drazen Jukic et Mahadev Satyanarayanan

2013 "OpenSlide : A vendor-neutral software foundation for digital pathology", *Journal of pathology informatics*, 4. (Cf. p. 80.)

Hadley, George

1962 "Linear programming." (Cf. p. 47.)

Haralick, Robert, Stanley R Sternberg et Xinhua Zhuang

1987 "Image analysis using mathematical morphology", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 4, p. 532–550. (Cf. p. 70, 74.)

Hartigan, John A et Manchek A Wong

1979 "Algorithm AS 136 : A k-means clustering algorithm", *Applied statistics*, p. 100–108. (Cf. p. 61.)

Hastie, Trevor, Robert Tibshirani et al.

1998 "Classification by pairwise coupling", *The annals of statistics*, 26, 2, p. 451–471. (Cf. p. 10.)

Hastie, Trevor, Robert Tibshirani et Jerome Friedman

2009 *Unsupervised learning*, Springer. (Cf. p. 2.)

Hattox, Alexis M et Sacha B Nelson

2007 "Layer V neurons in mouse cortex projecting to different targets have distinct physiological properties", *Journal of neurophysiology*, 98, 6, p. 3330–3340. (Cf. p. 32, 40.)

Higgins, John PT, Rajesh Shinghal, Harcharan Gill, Jeffrey H Reese, Martha Terris, Ronald J Cohen, Michael Fero, Jonathan R Pollack, Matt van de Rijn et James D Brooks

2003 "Gene expression patterns in renal cell carcinoma assessed by complementary DNA microarray", *The American journal of pathology*, 162, 3, p. 925–932. (Cf. p. 56.)

Ho, Jonhan, Anil V Parwani, Drazen M Jukic, Yukako Yagi, Leslie Anthony et John R Gilbertson

2006 "Use of whole slide imaging in surgical pathology quality assurance : design and pilot validation studies", *Human pathology*, 37, 3, p. 322–331. (Cf. p. 57.)

Howe, Nicholas R

2013 "Document binarization with automatic parameter tuning", *International Journal on Document Analysis and Recognition (IJDAR)*, 16, 3, p. 247–258. (Cf. p. 42, 75, 77.)

Hsu, Chih-Wei et Chih-Jen Lin

2002 "A comparison of methods for multiclass support vector machines", *Neural Networks, IEEE Transactions on*, 13, 2, p. 415–425. (Cf. p. 10.)

Hu, Ming-Kuei

1962 "Visual pattern recognition by moment invariants", *Information Theory, IRE Transactions on*, 8, 2, p. 179–187. (Cf. p. 38, 51.)

Huang, Chao-Hui, Antoine Veillard, Ludovic Roux, Nicolas Lomé-nie et Daniel Racoceanu

2011 "Time-efficient sparse analysis of histopathological whole slide images", *Computerized medical imaging and graphics*, 35, 7, p. 579–591. (Cf. p. 58.)

Irshad, Humayun, Antoine Veillard, Ludovic Roux et Daniel Racoceanu

2014 "Methods for Nuclei Detection, Segmentation, and Classification in Digital Histopathology : A Review – Current Status and Future Potential", *Biomedical Engineering, IEEE Reviews in*, 7, p. 97–114. (Cf. p. 58.)

Jang, Ben-Kwei et Roland T Chin

1990 "Analysis of thinning algorithms using mathematical morphology", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 12, 6, p. 541–551. (Cf. p. 75.)

Jiang, Yulei, Robert M Nishikawa, Dulcy E Wolverton, Charles E Metz, Maryellen L Giger, Robert A Schmidt, Carl J Vyborny et Kunio Doi

1996 "Malignant and benign clustered microcalcifications : automated feature analysis and classification." *Radiology*, 198, 3, p. 671–678. (Cf. p. 9.)

Joachims, Thorsten

1998 *Text categorization with support vector machines : Learning with many relevant features*, Springer. (Cf. p. 89.)

Jolliffe, Ian

2002 *Principal component analysis*, Wiley Online Library. (Cf. p. 65.)

Jonasch, Eric, Jianjun Gao et W Kimryn Rathmell

2014 "Renal cell carcinoma", *BMJ*, 349, doi : 10.1136/bmj.g4797. (Cf. p. 55.)

Jones, Judson P et Larry A Palmer

1987 "An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex", *Journal of neurophysiology*, 58, 6, p. 1233–1258. (Cf. p. 71, 82.)

Kapur, Jagat Narain, Prasanna K Sahoo et Andrew KC Wong

1985 "A new method for gray-level picture thresholding using the entropy of the histogram", *Computer vision, graphics, and image processing*, 29, 3, p. 273–285. (Cf. p. 82.)

Kaya, Kamer, Johannes Langguth, Fredrik Manne et Bora Uçar

2011 "Experiments on push-relabel-based maximum cardinality matching algorithms for bipartite graphs", *CERFACS, Toulouse, France, Technical Report TR/PA/11/33*. (Cf. p. 44, 46.)

2013 "Push-relabel based algorithms for the maximum transversal problem", *Computers & Operations Research*, 40, 5, p. 1266–1275. (Cf. p. 46.)

Kearns, Michael et Dana Ron

1999 "Algorithmic stability and sanity-check bounds for leave-one-out cross-validation", *Neural Computation*, 11, 6, p. 1427–1453. (Cf. p. 25.)

Keegan, Kirk A, Clayton W Schupp, Karim Chamie, Nicholas J Hellenthal, Christopher P Evans et Theresa M Koppie

2012 "Histopathology of surgically treated renal cell carcinoma : survival differences by subtype and stage", *The Journal of urology*, 188, 2, p. 391–397. (Cf. p. 56.)

Kijsirikul, Boonserm, Nitiwut Ussivakul et Surapant Meknavin

2002 "Adaptive Directed Acyclic Graphs for Multiclass Classification", in *Pacific Rim International Conference on Artificial Intelligence (PRICAI) 2002 : Trends in Artificial Intelligence*, Lecture Notes in Computer Science, t. 2417, p. 158–168. (Cf. p. 10.)

Kirbas, Cemil et Francis Quek

2004 "A review of vessel extraction techniques and algorithms", *ACM Computing Surveys (CSUR)*, 36, 2, p. 81–121. (Cf. p. 40.)

Kittler, Josef et John Illingworth

1985 "On threshold selection using clustering criteria", *Systems, Man and Cybernetics, IEEE Transactions on*, 5, p. 652–655. (Cf. p. 82.)

KITTLER, Josef, John ILLINGWORTH et J FÖGLEIN

1985 "Threshold selection based on a simple image statistic", *Computer vision, graphics, and image processing*, 30, 2, p. 125–147. (Cf. p. 82.)

KLAUTAU, Aldebaro, Nikola JEVTIĆ et Alon ORLITSKY

2003 "On nearest-neighbor error-correcting output codes with application to all-pairs multiclass support vector machines", *The Journal of Machine Learning Research*, 4, p. 1–15. (Cf. p. 10.)

KLEIHUES, Paul, David N LOUIS, Bernd W SCHEITHAUER, Lucy B RORKE, Guido REIFENBERGER, Peter C BURGER et Webster K CAVENEE

2002 "The WHO classification of tumors of the nervous system", *Journal of Neuropathology & Experimental Neurology*, 61, 3, p. 215–225. (Cf. p. 9.)

KOHAVI, Ron et al.

1995 "A study of cross-validation and bootstrap for accuracy estimation and model selection", in *Ijcai*, 2, t. 14, p. 1137–1145. (Cf. p. 20, 89.)

KOTHARI, Sonal, Qaiser CHAUDRY et May D WANG

2009 "Automated cell counting and cluster segmentation using concavity detection and ellipse fitting techniques", in *Biomedical Imaging : From Nano to Macro, 2009. ISBI'09. IEEE International Symposium on*, IEEE, p. 795–798. (Cf. p. 1.)

KRAUTHGAMER, Robert, Joseph Seffi NAOR et Roy SCHWARTZ

2009 "Partitioning graphs into balanced components", in *Proceedings of the twentieth Annual ACM-SIAM Symposium on Discrete Algorithms*, Society for Industrial et Applied Mathematics, p. 942–949. (Cf. p. 26.)

KRISSIAN, Karl, Grégoire MALANDAIN, Nicholas AYACHE, Régis VAILLANT et Yves TROUSSET

2000 "Model-based detection of tubular structures in 3D images", *Computer vision and image understanding*, 80, 2, p. 130–171. (Cf. p. 71, 82.)

KUHN, Harold W

1955 "The Hungarian method for the assignment problem", *Naval research logistics quarterly*, 2, 1-2, p. 83–97. (Cf. p. 44, 46.)

2014 "Nonlinear programming : a historical view", in *Traces and Emergence of Nonlinear Programming*, Springer, p. 393–414. (Cf. p. 93.)

LAM, Louisa, Seong-Whan LEE et Ching Y SUEN

1992 "Thinning methodologies-a comprehensive survey", *IEEE Transactions on pattern analysis and machine intelligence*, 14, 9, p. 869–885. (Cf. p. 75.)

LICHMAN, M.

2013 *UCI Machine Learning Repository*, http://archive.ics.uci.edu/ml. (Cf. p. 23.)

LIU, Yang, Rui WANG et Ying-Sheng ZENG

2007 "An improvement of one-against-one method for multi-class support vector machine", in *Machine Learning and Cybernetics, 2007 International Conference on*, IEEE, t. 5, p. 2915–2920. (Cf. p. 10.)

LOPEZ-BELTRAN, Antonio, Marina SCARPELLI, Rodolfo MONTIRONI et Ziya KIRKALI

2006 "2004 WHO classification of the renal tumors of the adults", *European urology*, 49, 5, p. 798–805. (Cf. p. 56.)

LOWE, David G

2004 "Distinctive image features from scale-invariant keypoints", *International journal of computer vision*, 60, 2, p. 91–110. (Cf. p. 2.)

LUISIER, Florian, Cédric VONESCH, Thierry BLU et Michael UNSER

2009 "Fast Haar-wavelet denoising of multidimensional fluorescence microscopy data", in *Biomedical Imaging : From Nano to Macro, 2009. ISBI'09. IEEE International Symposium on*, IEEE, p. 310–313. (Cf. p. 1.)

MADZAROV, Gjorgji, Dejan GJORGJEVIKJ et Ivan CHORBEV

2009 "A multi-class SVM classifier utilizing binary decision tree", *Informatica*, 33, 2. (Cf. p. 11.)

MANIKANDAN, J et B VENKATARAMANI

2009 "Design of a modified one-against-all SVM classifier", in *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on*, IEEE, p. 1869–1874. (Cf. p. 10.)

MEHTA, Dinesh P et Sartaj SAHNI

2004 *Handbook of data structures and applications*, CRC Press. (Cf. p. 16.)

MEYER, Fernand et Serge BEUCHER

1990 "Morphological segmentation", *Journal of visual communication and image representation*, 1, 1, p. 21–46. (Cf. p. 41.)

MOREIRA, Miguel et Eddy MAYORAZ

1998 "Improved pairwise coupling classification with correcting classifiers", in *Machine Learning : ECML-98*, Springer, p. 160–171. (Cf. p. 10.)

Mota, Tiago C et Antonio Carlos G Thome

2009 "One-against-all-based multiclass SVM strategies applied to vehicle plate character recognition", in *Neural Networks, 2009. IJCNN 2009. International Joint Conference on*, IEEE, p. 2153–2159. (Cf. p. 10.)

Mottini, Alejandro

2014 *Axon Morphology Analysis : from Image Processing to Modelling*, Theses, Universite de Nice Sophia-Antipolis (UNS), https://hal.inria.fr/tel-01074620. (Cf. p. 72.)

Munkres, James

1957 "Algorithms for the assignment and transportation problems", *Journal of the Society for Industrial and Applied Mathematics*, 5, 1, p. 32–38. (Cf. p. 44, 46.)

Näf, Martin, Gábor Székely, Ron Kikinis, Martha Elizabeth Shenton et Olaf Kübler

1997 "3D Voronoi skeletons and their usage for the characterization and recognition of 3D organ shape", *Computer Vision and Image Understanding*, 66, 2, p. 147–161. (Cf. p. 75, 82.)

Nam, David, Judith Mantell, Lorna Hodgson, David Bull, Paul Verkade et Alin Achim

2014 "Feature-based registration for correlative light and electron microscopy images", in *Image Processing (ICIP), 2014 IEEE International Conference on*, IEEE, p. 3567–3571. (Cf. p. 53.)

Nayak, Nandita, Hang Chang, Alexander Borowsky, Paul Spellman et Bahram Parvin

2013 "Classification of tumor histopathology via sparse feature learning", in *Biomedical Imaging (ISBI), 2013 IEEE 10th International Symposium on*, IEEE, p. 410–413. (Cf. p. 58.)

Nelder, John A et Roger Mead

1965 "A simplex method for function minimization", *The computer journal*, 7, 4, p. 308–313. (Cf. p. 47.)

Nesterov, Yurii, Arkadii Nemirovskii et Yinyu Ye

1994 *Interior-point polynomial algorithms in convex programming*, SIAM, t. 13. (Cf. p. 47.)

Ogniewicz, R et M Ilg

1992 "Voronoi skeletons : Theory and applications", in *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR'92., 1992 IEEE Computer Society Conference on*, IEEE, p. 63–69. (Cf. p. 75, 82.)

Oliva, Aude et Antonio Torralba

2001 "Modeling the Shape of the Scene : A Holistic Representation of the Spatial Envelope", *International Journal of Computer Vision*, 42, p. 145–175. (Cf. p. 8.)

Otsu, Nobuyuki

1975 "A threshold selection method from gray-level histograms", *Automatica*, 11, 285-296, p. 23–27. (Cf. p. 82.)

Peng, Yahui, Yulei Jiang, Laurie Eisengart, Mark A Healy, Francis H Straus et Ximing J Yang

2011 "Computer-aided identification of prostatic adenocarcinoma : Segmentation of glandular structures", *Journal of pathology informatics*, 2. (Cf. p. 65.)

Phetkaew, Thimaporn, Boonserm Kijsirikul et Wanchai Rivepiboon

2003 "Reordering adaptive directed acyclic graphs : an improved algorithm for multiclass support vector machines", in *Neural Networks, 2003. Proceedings of the International Joint Conference on*, IEEE, t. 2, p. 1605–1610. (Cf. p. 10, 11.)

Platt, John et al.

1999 "Fast training of support vector machines using sequential minimal optimization", *Advances in kernel methods-support vector learning*, 3. (Cf. p. 95.)

Platt, John C., Nello Cristianini et John Shawe-Taylor

2000 "Large Margin DAGs for Multiclass Classification", in *Advances in Neural Information Processing Systems 12 (NIPS-99)*, p. 547–553. (Cf. p. 10.)

Pock, Thomas, Reinhard Beichel et Horst Bischof

2005 "A novel robust tube detection filter for 3D centerline extraction", in *Image Analysis*, Springer, p. 481–490. (Cf. p. 71.)

Poulain, Emmanuelle, Sylvain Prigent, Emmanuel Soubies et Xavier Descombes

2015 "Cells Detection Using Segmentation Competition", in *ISBI - International Symposium on Biomedical Imaging*, IEEE, Brooklyn, United States, doi : 10.1109/ISBI.2015.7164090, https://hal.inria.fr/hal-01113167. (Cf. p. 1.)

Pratt, William K

1974 "Correlation techniques of image registration", *Aerospace and Electronic Systems, IEEE Transactions on*, 3, p. 353–358. (Cf. p. 44, 47.)

Quek, Francis KH et Cemil Kirbas

2001 "Vessel extraction in medical images by wave-propagation and traceback", *Medical Imaging, IEEE Transactions on*, 20, 2, p. 117–131. (Cf. p. 40.)

QUINLAN, J. ROSS

1986 "Induction of decision trees", *Machine learning*, 1, 1, p. 81–106. (Cf. p. 9.)

RIFKIN, Ryan et Aldebaro KLAUTAU

2004 "In defense of one-vs-all classification", *The Journal of Machine Learning Research*, 5, p. 101–141. (Cf. p. 10.)

ROCHA, Anderson et Siome KLEIN GOLDENSTEIN

2014 "Multiclass from binary : Expanding one-versus-all, one-versus-one and ecoc-based approaches", *Neural Networks and Learning Systems, IEEE Transactions on*, 25, 2, p. 289–302. (Cf. p. 9.)

SABO, Edmond, Albina BOLTENKO, Yanina SOVA, Avi STEIN, Shira KLEINHAUS et Murray B RESNICK

2001 "Microscopic analysis and significance of vascular architectural complexity in renal cell carcinoma", *Clinical Cancer Research*, 7, 3, p. 533–537. (Cf. p. 56, 58.)

SARTO, Cecilia, Alessandro MAROCCHI, Jean-Charles SANCHEZ, Daniela GIANNONE, Séverine FRUTIGER, Olivier GOLAZ, Marc R WILKINS, Giancarlo DORO, Francesco CAPPELLANO, Graham HUGHES et al.

1997 "Renal cell carcinoma and normal kidney protein expression", *Electrophoresis*, 18, 3-4, p. 599–604. (Cf. p. 56.)

SCHAPIRE, Robert E

2003 "The boosting approach to machine learning : An overview", in *Nonlinear estimation and classification*, Springer, p. 149–171. (Cf. p. 9.)

SCHOLKOPF, Bernhard et Alexander J SMOLA

2001 *Learning with kernels : support vector machines, regularization, optimization, and beyond*, MIT press. (Cf. p. 91, 92, 94–96.)

SCHRIJVER, Alexander

2003 *Combinatorial optimization : polyhedra and efficiency*, Springer Science & Business Media, t. 24. (Cf. p. 47.)

SIVIC, Josef et Andrew ZISSERMAN

2003 "Video Google : A text retrieval approach to object matching in videos", in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, IEEE, p. 1470–1477. (Cf. p. 2, 8.)

SOBIN, Leslie H et Irvin D FLEMING

1997 "TNM classification of malignant tumors, (1997)", *Cancer*, 80, 9, p. 1803–1804. (Cf. p. 9.)

Sofka, Michal et Charles V Stewart

2006 "Retinal vessel centerline extraction using multiscale matched filters, confidence and edge measures", *Medical Imaging, IEEE Transactions on*, 25, 12, p. 1531–1546. (Cf. p. 71.)

Stavros, A Thomas, David Thickman, Cynthia L Rapp, Mark A Dennis, Steve H Parker et Gale A Sisney

1995 "Solid breast nodules : use of sonography to distinguish between benign and malignant lesions." *Radiology*, 196, 1, p. 123–134. (Cf. p. 9.)

Stoer, Mechthild et Frank Wagner

1997 "A simple min-cut algorithm", *Journal of the ACM (JACM)*, 44, 4, p. 585–591. (Cf. p. 16, 18, 99–101.)

Störkel, Stephan, John N. Eble, K. Adlakha, Mahul Amin, Michael L. Blute, David G. Bostwick, M. Darson, Brett Delahunt et K. Iczkowski

1997 "Classification of renal cell carcinoma", *Cancer*, 80, 5, p. 987–989, issn : 1097-0142. (Cf. p. 56.)

Takahashi, Fumitake et Shigeo Abe

2002 "Decision-tree-based multiclass support vector machines", in *Neural Information Processing, 2002. ICONIP'02. Proceedings of the 9th International Conference on*, IEEE, t. 3, p. 1418–1422. (Cf. p. 11.)

Takahashi, Masayuki, Daniel R Rhodes, Kyle A Furge, Hiro-omi Kanayama, Susumu Kagawa, Brian B Haab et Bin Tean Teh

2001 "Gene expression profiling of clear cell renal cell carcinoma : gene identification and prognostic classification", *Proceedings of the National Academy of Sciences*, 98, 17, p. 9754–9759. (Cf. p. 56.)

Telea, Alexandru et Jarke J Van Wijk

2002 "An augmented fast marching method for computing skeletons and centerlines", in *Proceedings of the symposium on Data Visualisation 2002*, Eurographics Association, 251–ff. (Cf. p. 42, 75, 82.)

Tibshirani, Robert et Trevor Hastie

2007 "Margin trees for high-dimensional classification", *The Journal of Machine Learning Research*, 8, p. 637–652. (Cf. p. 12, 16, 24.)

Tristán-Vega, Antonio, Santiago Aja-Fernández et Carl-Fredrik Westin

2012 "Deblurring of probabilistic ODFs in quantitative diffusion MRI", in *Biomedical Imaging (ISBI), 2012 9th IEEE International Symposium on*, IEEE, p. 932–935. (Cf. p. 1.)

VanderWeele, Tyler J et James M Robins

2007 "Four types of effect modification : A classification based on directed acyclic graphs", *Epidemiology*, 18, 5, p. 561–568. (Cf. p. 10.)

Vapnik, Vladimir

2013 *The nature of statistical learning theory*, Springer Science & Business Media. (Cf. p. 89.)

Vural, Volkan et Jennifer G Dy

2004 "A hierarchical method for multi-class support vector machines", in *Proceedings of the twenty-first international conference on Machine learning*, ACM, p. 105. (Cf. p. 11.)

Wallace, Michael B et Paul Fockens

2009 "Probe-based confocal laser endomicroscopy", *Gastroenterology*, 136, 5, p. 1509–1513. (Cf. p. 23.)

Wang, Ching-Wei, Dean Fennell, Ian Paul, Kienan Savage et Peter Hamilton

2011 "Robust automated tumour segmentation on histological and immunohistochemical tissue images", *PloS one*, 6, 2, e15818. (Cf. p. 60.)

Wang, Jason TL, Steve Rozen, Bruce A Shapiro, Dennis Shasha, Zhiyuan Wang et Maisheng Yin

1999 "New techniques for DNA sequence classification", *Journal of Computational Biology*, 6, 2, p. 209–218. (Cf. p. 9.)

Wilson, Tony

1990 "Confocal microscopy", *Academic Press : London, etc*, 426, p. 1–64. (Cf. p. 33.)

Windeatt, Terry et Reza Ghaderi

2003 "Coding and decoding strategies for multi-class learning problems", *Information Fusion*, 4, 1, p. 11–21. (Cf. p. 10.)

Yang, Hannah Honghua et DF Wong

1996 "Balanced partitioning", *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 15, 12, p. 1533–1540. (Cf. p. 26.)

Yang, Jun, Yu-Gang Jiang, Alexander G Hauptmann et Chong-Wah Ngo

2007 "Evaluating bag-of-visual-words representations in scene classification", in *Proceedings of the international workshop on Workshop on multimedia information retrieval*, ACM, p. 197–206. (Cf. p. 25.)

Yeh, Fang-Cheng, Anil V Parwani, Liron Pantanowitz et Chien Ho

2014 "Automated grading of renal cell carcinoma using whole slide imaging", *Journal of pathology informatics*, 5. (Cf. p. 58.)

Zhang, Bob, Lin Zhang, Lei Zhang et Fakhri Karray

2010 "Retinal vessel extraction by matched filter with first-order derivative of Gaussian", *Computers in biology and medicine*, 40, 4, p. 438–445. (Cf. p. 40.)

Zhang, TY et Ching Y. Suen

1984 "A fast parallel algorithm for thinning digital patterns", *Communications of the ACM*, 27, 3, p. 236–239. (Cf. p. 75.)

Zhu, Ji, Hui Zou, Saharon Rosset et Trevor Hastie

2009 "Multi-class adaboost", *Statistics and its Interface*, 2, 3, p. 349–360. (Cf. p. 9.)

Zitova, Barbara et Jan Flusser

2003 "Image registration methods : a survey", *Image and vision computing*, 21, 11, p. 977–1000. (Cf. p. 44.)

TABLE DES FIGURES

Figure 1 Schéma général de l'apprentissage supervisé. 8

Figure 2 Exemple d'arbre construit par la procédure générale d'extension de classifieur. Les chiffres correspondent aux indices des classes. 14

Figure 3 Exemples d'arbres binaires de classifieurs. 15

Figure 4 Type d'arbre résultant de la phase d'apprentissage de la méthode proposée. Un exemple de classification prédisant la classe 4 est indiqué **en gras** sur l'arbre. 18

Figure 5 Les étapes de l'algorithme GCSVM standard. 21

Figure 6 Les étapes de l'algorithme GCSVM équilibré. 22

Figure 7 Exemples de vidéos endomicroscopiques montrant des polypes. De gauche à droite : purement bénin, hyperplastique, adénome tubuleux, adénome tubulo-villeux, adénocarcinome. 25

Figure 8 Précision en fonction du paramètre C (discrétisé selon une échelle logarithmique) pour les extensions du SVM (OVO, OVA, DAGSVM, et la méthode proposée). 27

Figure 9 Principales parties d'un neurone : le corps cellulaire et les dendrites (apicale et basales). 32

Figure 10 Exemples de types de connectivité pour les neurones de la couche V du néocortex 33

Figure 11 Projection du maximum d'intensité d'une des images x10. 34

Figure 12 Projection du maximum d'intensité d'une des images x40. 35

Figure 13 Les étapes de la segmentation des neurones. Image initiale (haut gauche), détection des corps cellulaires correspondant au premier seuil (haut droite), deuxième seuil (milieu gauche), première segmentation (milieu droite), troisième seuil (bas gauche) et résultat final (bas droite). Les images montrées correspondent à la projection du maximum d'intensité. 37

Figure 14 Les étapes de la détection des bases de dendrites. Pour cette illustration, les opérations ont été réalisées sur la projection du maximum d'intensité. 39

Figure 15 Schéma de calcul des angles θ_{tot}, θ_{min} et θ_{max} à partir des résultats de la Figure 14d. S est le barycentre du corps cellulaire, D_A est la dendrite apicale, D_1, D_2 et D_3 sont les dendrites basales. 40
Figure 16 Les étapes de la reconstruction des neurones. 42
Figure 17 Les deux étapes de la reconstruction de neurones : détection des sections de dendrites (à gauche) et reconstruction des dendrites (à droite). 43
Figure 18 Temps de calcul (en secondes) de la résolution du problème d'appariment pour $1 \leqslant n \leqslant N \leqslant 100$ pour l'algorithme hongrois (à gauche), et pour le simplexe (à droite). 48
Figure 19 Taux de réussite en fonction du facteur de bruit ϵ, pour $N = 100$ et différentes valeurs de n. 49
Figure 20 Illustration de la carte de correspondance sur la projection du maximum d'intensité d'images. Les têtes des neurones correspondants sont entourées de la même couleur sur l'image 10x (en haut) que sur les images 40x (en bas). 50
Figure 21 Histogrammes de répartition des moments de forme [Hu, 1962] pour l'ensemble des neurones segmentés. 51
Figure 22 Histogrammes de répartition des différents descripteurs calculés. 52
Figure 23 Évolution de la somme des distances intra-cluster (c'est-à-dire des points au barycentre du cluster) en fonction du nombre de classes k donné à l'algorithme des k-moyennes. 53
Figure 24 Un exemple de lamelle scannée. 56
Figure 25 La structure pyramidale du *Whole-Slide Imaging*. 57
Figure 26 Exemple d'images prises aux niveaux de grossissement minimal et maximal. 58
Figure 27 Bande floue apparaissant dans l'image. 60
Figure 28 Illustration de la variabilité des images données. 61
Figure 29 Les différentes mesures appliquées à l'image. Sur l'image initiale, les zones d'intérêt sont grossièrement entourées en noir. 62
Figure 30 Résultats de segmentation grâce au partitionnement par k-moyennes pour différentes valeurs de k. Sur l'image initiale, les zones d'intérêt sont grossièrement entourées en noir. 63
Figure 31 Résultats de segmentation sur trois lames. 64

Figure 32 Gestion du recouvrement entre deux images voisines. Les parties hachées correspondent aux zones de confiance de chacune d'entre elles. 65

Figure 33 Un exemple d'image et les trois composantes données par l'ACP. 66

Figure 34 Un exemple d'image, les canaux rouge, vert et bleu correspondant et l'image violette calculée. 67

Figure 35 Un exemple d'image, les canaux C_1, C_2, C_3, et les résultats des canaux empiriques correspondant à l'hématoxyline et l'éosine. 68

Figure 36 Exemple d'image montrant un agrégat de noyaux dans un vaisseau. De gauche à droite : l'image initiale, le canal H, et le canal V. 70

Figure 37 Étapes de la segmentation des noyaux n'appartenant pas aux vaisseaux. De gauche à droite : l'image initiale, la segmentation de tous les noyaux, et la segmentation des noyaux en dehors des vaisseaux. 71

Figure 38 Exemple de suppression des noyaux hors vaisseaux sur le canal V : à gauche, le canal V, à droite, le canal V où les pixels correspondant aux noyaux hors vaisseaux ont été mis à zéro. 72

Figure 39 Pré-détection des vaisseaux par seuillage par hystérésis. De gauche à droite : l'image d'entrée (Cf. Figure 38), la réponse du seuillage par hystérésis, et le résultat final, après suppression des plus petites composantes connexes. 73

Figure 40 Filtres de Gabor pour différentes valeurs de θ (colonnes) et σ (lignes). 74

Figure 41 Les demi-filtres de Gabor utilisés et le filtre de Gabor (complet) correspondant. 75

Figure 42 Segmentation du réseau vasculaire. De gauche à droite : l'image initiale, la réponse au filtre de Gabor modifié, la réponse du filtre superposée au canal V, le seuillage de la réponse au filtre et le seuillage de l'image superposé au canal V. 76

Figure 43 Procédé de découpage d'un squelette en branches. j est le pixel de jonction (et vaut donc systématiquement 1 puisqu'appartenant au graphe), et les x représentent le voisinage 3×3 et valent soit 0, soit 1. 77

FIGURE 44 Extraction du squelette correspondant au réseau vasculaire. De gauche à droite : l'image initiale, le premier squelette détecté et le squelette post-traité. 78

FIGURE 45 Un exemple simple de deux ensembles de données linéairement séparables. H_1 n'est pas un hyperplan séparateur, H_2 est un hyperplan séparateur, and H_3 est l'hyperplan séparateur offrant la plus grande marge. 90

FIGURE 46 Le classifieur SVM dans un cas bidimensionnel linéairement séparable. 91

LISTE DES TABLEAUX

Table 1 Nombre de classifieurs appris (phase d'apprentissage) et utilisés (phase de classification) pour les principales extensions. 12

Table 2 Complexité d'apprentissage et de classification pour les différentes extensions de SVM : OVO, OVA, ECOC, GCSVM « non équilibré » (Cf. Section 2.2) et GCSVM « équilibré » (Cf. Section 2.2.2). 19

Table 3 Précision pour les forêts d'arbres décisionnels [Breiman, 2001], les principales extensions pour le SVM, les Margin Trees [Tibshirani et Hastie, 2007], et la méthode proposée (avec ou sans équilibrage). 24

Table 4 Taux de bonne classification pour [André et al., 2011] et quelques extensions multiclasses. 26

Table 5 Données extraites pour chacune des lames observées. NI correspond au nombre de sous-images dans le découpage de la zone tumorale (et est donc un critère directement lié à son aire). 81

PUBLICATIONS ET ACTIVITÉS D'ENSEIGNEMENT

PUBLICATIONS

Conférences internationales avec actes

— **Alexis Zubiolo**, Barbara André, Éric Debreuve and Grégoire Malandain : *A Recursive Approach For Multiclass Support Vector Machine : Application to Automatic Classification of Endomicroscopic Videos.* International Conference on Computer Vision Theory and Applications (VISAPP), Lisbon, Portugal, 2014.

— **Alexis Zubiolo**, Kawssar Harb, Michèle Studer, Éric Debreuve and Xavier Descombes : *Morphological Analysis and Feature Extraction of Neurons from Mouse Cortices Multiscale 3D Microscopic Images.* International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Milano, Italy, 2015.

Conférences nationales avec actes

— **Alexis Zubiolo**, Kawssar Harb, Michèle Studer, Éric Debreuve and Xavier Descombes : *Comment retrouver une constellation dans la galaxie ?* Colloque Gretsi, Lyon, France, 2015.

Conférences sans actes

— **Alexis Zubiolo**, Kawssar Harb, Nikita Lukianets, Franck Grammont, Michèle Studer, Éric Debreuve and Xavier Descombes : *Towards Neuron Classification From 3D Confocal Microscopic Images.* SIGNALIFE Workshop on Cell Signaling, Nice, France, 2014.

— Christian Alfano, Kawssar Harb, Elia Magrinelli, Nikita Lukianets, **Alexis Zubiolo**, Celine Nicolas, Éric Debreuve, Guillaume Sandoz, Xavier Descombes, Franck Grammont and Michèle Studer : *Concomitant expression of subcerebral- and callosal-specific genes defines a deep layer cell population in the postnatal mouse neocortex.* Society for Neuroscience, Washington DC., USA, 2014.

ACTIVITÉS D'ENSEIGNEMENT

Au total, 205 heures d'enseignement ont été dispensées à l'Université Nice Sophia Antipolis et l'école Polytech Nice-Sophia.

Année universitaire 2012-2013

— Informatique générale (L1 – 21 heures)
— Systèmes d'exploitation (L1 – 42 heures)
— Introduction au traitement d'images (M2 – 10 heures)
Sous-total : 73 heures.

Année universitaire 2013-2014

— Informatique générale (L1 – 36 heures)
— Programmation orientée objet en Python (L2 – 18 heures)
— Programmation web (L1 – 10 heures)
Sous-total : 66 heures.

Année universitaire 2014-2015

— Informatique générale (L1 – 18 heures)
— Programmation web (L1 – 48 heures)
Sous-total : 66 heures.

RÉSUMÉ

L'objectif de cette thèse est de s'intéresser à la classification de cellules et de tissus au sein d'images d'origine biomédicales en s'appuyant sur des critères morphologiques. Le but est de permettre aux médecins et aux biologistes de mieux comprendre les lois qui régissent certains phénomènes biologiques. Ce travail se décompose en trois principales parties correspondant aux trois problèmes typiques des divers domaines de l'imagerie biomédicale abordés.

L'objet de la première est l'analyse de vidéos d'endomicroscopie du colon dans lesquelles il s'agit de déterminer automatiquement la classe pathologique des polypes qu'on y observe. Cette tâche est réalisée par un apprentissage supervisé multiclasse couplant les séparateurs à vaste marge à des outils de théorie des graphes. Le bipartitionnement d'un graphe représentant l'ensemble des classes permet de construire un arbre de classifieurs de manière récursive.

La deuxième partie s'intéresse à l'étude de la morphologie de neurones de souris observés par microscopie confocale en fluorescence. Afin de disposer d'une information riche, les neurones sont observés à deux grossissements, l'un permettant de bien caractériser les corps cellulaires, l'autre, plus faible, pour voir les dendrites apicales dans leur intégralité. Sur ces deux types d'images, des descripteurs morphologiques des neurones sont extraits automatiquement en vue d'une classification. Une étape de mise en correspondance des neurones est proposée afin de combiner les informations extraites en une information unique.

La dernière partie concerne le traitement multi-échelle d'images d'histologie digitale dans le contexte du cancer du rein. Sur ces images de lamelles marquées à l'hématoxyline et à l'éosine, le réseau vasculaire est extrait et mis sous forme de graphe afin de pouvoir établir un lien entre l'architecture de la tumeur et sa classe pathologique. L'approche proposée combine un découpage de l'image en sous-images sur lesquelles le réseau est extrait de manière locale avant d'être fusionné en une information globale décrivant l'ensemble de l'échantillon.

MOTS-CLÉS : Classification, apprentissage statistique, imagerie biomédicale, endomicroscopie, histologie, microscopie confocale en fluorescence.

ABSTRACT

The purpose of this Ph.D. thesis is to study the classification based on morphological features of cells and tissues taken from biomedical images. The goal is to help medical doctors and biologists better understand some biological phenomena. This work is spread in three main parts corresponding to the three typical problems in biomedical imaging tackled.

The first part consists in analyzing endomicroscopic videos of the colon in which the pathological class of the polyps has to be determined. This task is performed using a supervised multiclass machine learning algorithm combining support vector machines and graph theory tools. The bipartitioning of a graph representing the set of classes leads to the construction of a tree of binary classifiers in a recursive fashion.

The second part concerns the study of the morphology of mice neurons taken from fluorescent confocal microscopy. In order to obtain a rich information, the neurons are imaged at two different magnifications, the higher magnification where the soma appears in details, and the lower showing the whole cortex, including the apical dendrites. On these images, morphological features are automatically extracted with the intention of performing a classification. A matching between corresponding neurons is proposed to gather the information extracted from both sources.

The last part is about the multi-scale processing of digital histology images in the context of kidney cancer. The vascular network is extracted and modeled by a graph to establish a link between the architecture of the tumor and its pathological class. The proposed approach combines a sampling of the main images in subimages on which the networks are locally extracted and merged to build the full graph describing the whole sample.

KEYWORDS : Classification, machine learning, biomedical imaging, endomicroscopy, histology, fluorescent confocal microscopy.