



Université de Caen
Basse-Normandie

Université de Caen Basse-Normandie

UFR sciences

École doctorale SIMEM

Thèse
présentée par

M. Vinh Thong TA

et soutenue
le 2 décembre 2009

en vue de l'obtention du
Doctorat de l'université de Caen Basse-Normandie
Spécialité : informatique et applications
(Arrêté du 07 août 2006)

Équations aux différences partielles définies sur des graphes pour le traitement d'images et de données

Membres du jury

M. Laurent	COHEN	DR CNRS	Université Paris Dauphine	Rapporteur
M. Laurent	NAJMAN	Pr.	Institut Gaspard MONGE	Rapporteur
M. Jean-Michel	JOLION	Pr.	Université de Lyon	Examineur
M. Olivier	LÉZORAY	Pr.	Université de Caen Basse-Normandie	Examineur
M. Petros	MARAGOS	Pr.	Université d'Athènes	Examineur
M. Rémy	MALGOUYRES	Pr.	Université d'Auvergne	Examineur
M. Abderrahim	ELMOATAZ	Pr.	Université de Caen Basse-Normandie	Directeur de thèse

Remerciements

Les travaux de recherches que je présente dans ce manuscrit de thèse ont été initiés et encadrés par Abderrahim ELMOATAZ (mon directeur de thèse) et Olivier LÉZORAY. Je les remercie tous deux pour m'avoir fait découvrir le monde de la recherche. Je les remercie également pour leur investissement et encadrement qui m'ont permis de mener à bien ces trois années de doctorat. J'espère pouvoir continuer à échanger avec eux dans le futur et continuer à profiter de leurs conseils avisés.

Je tiens à remercier Laurent COHEN, Jean-Michel JOLION, Rémi MALGOUYRES, et Petros MARAGOS, Laurent NAJMAN d'avoir accepté de faire partie de mon jury. Je remercie particulièrement Laurent COHEN et Laurent NAJMAN pour avoir accepté de rapporter ma thèse. Je les remercie également d'avoir consacré de leur temps à échanger avec moi sur mes travaux et mon manuscrit.

Je remercie tous les membres du GREYC : permanents, doctorants et l'équipe administrative. Je remercie particulièrement les membres de l'équipe image pour m'avoir accueilli, pour les discussions et les moments de détente que nous avons partagés ensemble.

Je remercie le conseil régional de Basse-Normandie et les membres de l'association « Cœur et Cancer » pour leur soutien financier. Sans eux, toute cette aventure n'aurait jamais pu commencer.

Merci aussi à Lucien Jeannette, à Jacques Vabre et à Jean Nicot pour leur soutien indéfectible durant la rédaction de ce manuscrit et les *deadlines* à respecter.

Finalement, un merci serait bien peu devant tout ce que mes proches, mes amis et ma famille m'ont apportés.

Je dédie ces travaux à mes Parents

Sommaire

Remerciements	i
Sommaire	iii
1 Introduction générale	1
2 Graphes et opérateurs	7
2.1 Introduction	9
2.2 Définitions et notations	9
2.3 Structure et topologie	10
2.4 Opérateurs définis sur des graphes	17
2.5 Une nouvelle famille de différences	22
2.6 Une nouvelle famille de gradients	23
2.7 Propriétés et relations des gradients	24
2.8 Formulation générale du p -Laplacien	29
2.9 Propriétés et relations du p -Laplacien	30
2.10 Conclusion	31
3 Fonctionnelles discrètes de régularisation	33
3.1 Fonctionnelles de régularisation	37
3.2 Modèles de régularisation, filtrage et simplification	47
3.3 Problèmes d'interpolation basés sur la régularisation	73
3.4 Optimisation globale pour la segmentation de données	97
4 Morphologie mathématique	115
4.1 Introduction	118
4.2 Dilatation et érosion	120
4.3 Morphologie mathématique par équations aux différences partielles	122
4.4 Expérimentations	132
4.5 Conclusion	143
5 Équation eikonale	147
5.1 Introduction	149
5.2 Adaptation de l'équation eikonale	153
5.3 Schémas numériques et algorithmes associés	154
5.4 Expérimentations	157
5.5 Conclusion	173
6 Conclusion générale	175

Liste des publications	181
Bibliographie	183
Index	195
Tables des matières	197

Introduction générale

Les récentes avancées technologiques en termes de procédés d'acquisition, de transmission et de stockage génèrent actuellement une profusion de données de tous types et de toutes natures. Celles-ci proviennent de différents domaines comme l'imagerie numérique, la bioinformatique, l'informatique graphique ou encore les réseaux complexes (sociaux, biologiques ou informatiques).

Cette grande quantité d'information produite, loin d'être volatile, est désormais stockée dans des bases de données et les défis actuels que posent ces données consistent à les traiter, les classer, les analyser ou encore à en extraire des connaissances.

Ces données sont, d'une manière générale, de grandes dimensions en terme de volume et de description. Ces aspects constituent un obstacle majeur pour leurs traitements et leurs analyses. Une autre de leurs caractéristiques est qu'elles peuvent être complexes et de différents types rendant le choix d'une représentation commune difficile. Néanmoins, ces données sont discrètes : elles le sont par nature ou par la nécessité de discrétisation d'un phénomène initialement continu.

Ainsi, une structure naturellement adaptée à la représentation de données discrètes réside dans l'utilisation des graphes. Les sommets d'un graphe permettent de représenter un ensemble de données et les arêtes rendent compte de leur proximité (la figure 1.1 montre des exemples de graphes permettant de représenter différents types de données et de domaines). Nous retrouvons cette structure dans différents domaines d'applications tels que l'étude des réseaux complexes [Newman *et al.*, 2006], l'apprentissage des bases de données [Chapelle *et al.*, 2006] ou encore le traitement des images et la vision par ordinateur [Jolion et Rosenfeld, 1994].

Il existe différentes méthodes de traitement et d'analyse dépendantes du type de données et de leur représentation. Nous nous intéressons dans ce manuscrit à deux familles de méthodes : d'une part, celles basées sur la théorie des graphes et l'analyse spectrale et d'autre part, celles basées sur les modèles variationnels et les équations aux dérivées partielles définis dans le domaine continu et appliqués au traitement et à l'analyse des images. La première famille de ces méthodes intègre directement la nature discrète des données alors que la seconde la modélise dans le domaine continu.

Les travaux présentés dans ce manuscrit se situent entre ces deux familles de méthodes et nous proposons de transcrire, d'adapter et d'unifier les modèles définis dans le domaine continu vers des modèles discrets par l'utilisation des graphes.

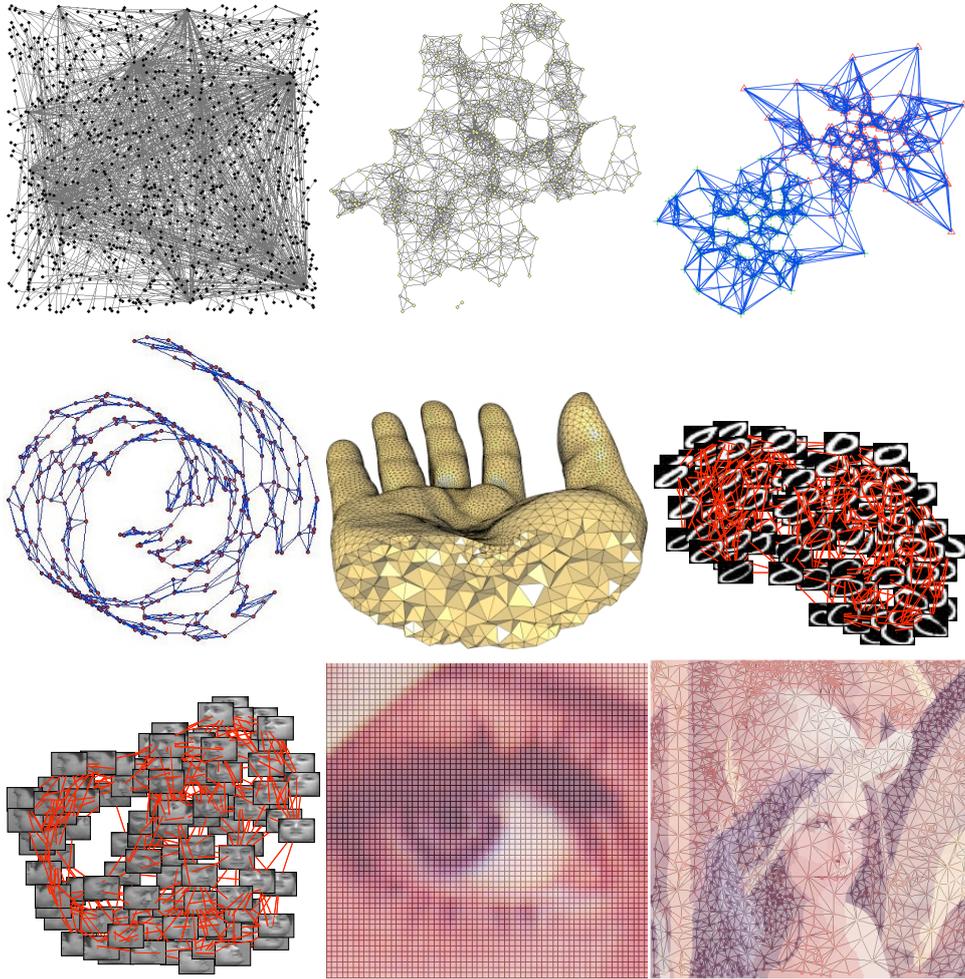


FIG. 1.1: Exemples de graphes représentant des données organisées ou non. Parmi les exemples, nous pouvons trouver des graphes représentant des réseaux sociaux, des nuages de points, des maillages, des bases de données et des images (graphe grille, graphe des composantes connexes).

Méthodes basées sur la théorie des graphes

Dans le domaine de l'analyse, du traitement, de la classification et de l'apprentissage de données, de nombreuses méthodes basées sur des graphes ont été proposées. Elles sont basées sur des concepts et des méthodes provenant de la théorie des graphes et de l'analyse spectrale et exploitent les propriétés spectrales du Laplacien ainsi que les modèles de diffusion associés.

Ces méthodes sont devenues très populaires pour de nombreuses applications telles que la réduction de dimension, le regroupement de données similaires ou la classification. Le lecteur intéressé par plus de détails sur ces méthodes pourra se référer, par exemple [Belkin et Niyogi, 2003, Lafon et Lee, 2006, Shi et Malik, 2000, Belkin *et al.*, 2006, von Luxburg, 2007, Bühler et Hein, 2009] et aux références associées.

Méthodes variationnelles et basées sur les équations aux dérivées partielles

Initialement développées par des physiciens et des mécaniciens, les méthodes basées sur les équations aux dérivées partielles trouvent de plus en plus de champs d'application tels que la biologie, la finance, le traitement des images et la vision par ordinateur.

Dans ces deux derniers domaines, ces approches constituent une part importante des méthodes existantes et sont utilisées dans de nombreuses applications telles que la restauration, le débruitage, la segmentation, le flot optique ou encore le suivi d'objets dans des séquences vidéos. Le lecteur intéressé par ces méthodes pourra se référer par exemple à [Alvarez *et al.*, 1993, Chan et Shen, 2005, Tsai et Osher, 2005, Aubert et Kornprobst, 2006] et aux références associées.

Les modèles définis dans le domaine continu possèdent de nombreux avantages tels qu'une théorie bien établie et des outils mathématiques sophistiqués. Leurs solutions sont obtenues par discrétisation d'opérateurs différentiels afin de pouvoir disposer de schémas numériques applicables.

Cette discrétisation constitue un obstacle non négligeable lorsque les données à traiter sont discrètes et de grande dimension. En effet, les opérateurs différentiels sont difficilement discrétisables lorsque les données sont définies sur des domaines non uniformes ou possèdent une grande dimensionnalité. Ce problème ne permet pas d'utiliser facilement ces modèles dans des domaines de recherche traitant avec ce type de données.

Motivations et contributions principales

Partant de la nécessité de traiter et d'analyser les données discrètes de grande dimension et du constat que les méthodes définies dans le domaine continu sont difficilement applicables pour ce type de données, les travaux présentés dans ce manuscrit proposent de transcrire et d'adapter certains de ces modèles (issus du traitement des images) dans un cadre discret en utilisant les graphes. Ces travaux se situent dans la continuité de ceux récemment proposés par [Bougleux *et al.*, 2007, Elmoataz *et al.*, 2008a].

Notre approche exploite : le formalisme des équations aux différences partielles, le bénéfice des méthodes basées sur les graphes et les outils mathématiques définis dans le domaine continu. Elle permet ainsi de proposer de nouvelles formulations de modèles variationnels et de modèles basés sur des équations aux dérivées partielles issus du traitement des images.

Les modèles abordés dans ce manuscrit sont les suivants : des modèles variationnels de régularisation, des équations aux dérivées partielles pour la morphologie mathématique et l'équation eikonale. Leur adaptation sur des graphes se fait par transcription directe des équations aux dérivées partielles vers des équations aux différences partielles ou en utilisant de nouveaux opérateurs de dérivation définis sur des graphes.

Les principales contributions de notre approche sont les suivantes.

- Contrairement aux équations aux dérivées partielles, nos modèles ne nécessitent pas de discrétisation spatiale grâce à leurs définitions discrètes sur des graphes.

- Nos modèles étendent les modèles définis dans le domaine continu pour le traitement de n'importe quel type de données y compris celles définies sur des domaines irréguliers ou de grande dimension.
- Notre approche permet d'appliquer des méthodes issues du domaine du traitement des images (filtrage, simplification, restauration, segmentation et classification) à des domaines tels que ceux de l'apprentissage ou de l'analyse de données arbitraires (collections d'images, base de données, nuages de points, ...). Ce point permet d'ouvrir de nouvelles perspectives et d'offrir de nouveaux modèles pour ces domaines de recherche.
- Dans le domaine du traitement des images, les schémas non locaux [Buades *et al.*, 2008, Gilboa et Osher, 2007a, Elmoataz *et al.*, 2008a] ont montré la supériorité de ces configurations par rapport aux schémas locaux. Nous montrerons que notre approche unifie (dans une même formulation) ces deux types de schémas (locaux et non locaux) et que les notions de localité et de non localité disparaissent dans le contexte des graphes.
- L'unification des configurations locales et non locales permet d'introduire de nouveaux outils pour le traitement des images tels que des opérations non locales de morphologie mathématique et la segmentation non locale avec l'équation eikonale.
- Finalement, ces transcriptions et ces adaptations sont rendues possibles par l'introduction de nouvelles familles d'opérateurs différentiels définis sur des graphes tels que les différences et les gradients directionnels ou encore une classe de p -Laplacien dont notamment le p -Laplacien anisotrope. Ces opérateurs peuvent être vus comme des extensions de ceux proposés dans [Bougleux *et al.*, 2007, Elmoataz *et al.*, 2008a].

Plan du manuscrit

Ce manuscrit est divisé en quatre chapitres. Au début de chaque chapitre, le lecteur trouvera : son résumé, ses mots clés, un descriptif de son contenu et les publications associées. Hormis le premier chapitre qui donne les définitions, les notions et les notations essentielles à la bonne compréhension de l'ensemble du manuscrit, chaque chapitre peut se lire de manière indépendante, même si certains résultats d'un chapitre peuvent être réutilisés dans un autre.

Dans chacun de ses chapitres, des expérimentations montreront le potentiel de notre méthodologie basée sur des graphes pour le traitement, la segmentation et la classification d'images et plus généralement de données arbitraires.

Chapitre 1 : Graphes et opérateurs

Dans ce premier chapitre, nous introduisons les notions et les définitions nécessaires à la bonne compréhension de la suite de ce manuscrit.

Dans une première partie, nous rappelons de manière brève les notations et les définitions de la théorie des graphes. Dans une seconde partie, à partir des définitions d'opérateurs définis sur des graphes proposés par [Bougleux *et al.*, 2007, Elmoataz *et al.*, 2008a], nous introduisons et étudions une nouvelle famille d'opérateurs différentiels tels que les différences et les gradients directionnels et une classe

de Laplaciens. Ces opérateurs, basés sur des équations aux différences partielles, sont les constituants principaux des différentes approches et thèmes que nous aborderons tout au long de ce manuscrit.

Chapitre 2 : Fonctionnelles discrètes de régularisation

Dans ce chapitre, nous introduisons un cadre formel permettant d'exprimer des fonctionnelles de régularisation définies sur des graphes. Elles sont basées sur les opérateurs différentiels discrets de premier et second ordre présentés dans le chapitre 2.

Les approches que nous proposons dans ce chapitre considèrent la régularisation comme un problème variationnel et plusieurs classes seront présentées et étudiées. Ces classes de modèles basées sur les fonctionnelles discrètes de régularisation permettent de considérer différentes problématiques.

- Les problèmes de filtrage et de simplification de données discrètes arbitraires.
- Les problèmes d'interpolation basés sur la régularisation tels que la colorisation d'images en monochromes ou les méthodes semi supervisées de segmentation, de classification et d'apprentissage.
- Les problèmes d'optimisation globale de fonctionnelles non convexes pour la segmentation d'images et le partitionnement de données arbitraires.

Chapitre 3 : Morphologie mathématique

La morphologie mathématique offre un large choix d'opérateurs permettant de résoudre de nombreuses applications dans les domaines du traitement des images et de la vision par ordinateur. Les opérations morphologiques peuvent être formulées en terme d'opérations algébriques (discrètes) sur des ensembles ou en terme d'équations aux dérivées partielles.

Dans ce chapitre, nous introduisons une nouvelle formulation de la morphologie mathématique formalisée à partir des équations aux différences partielles et des opérateurs discrets directionnels introduits dans le premier chapitre.

Cette nouvelle formulation est une transcription de modèles définis à partir d'équations aux dérivées partielles vers des équations aux différences partielles. Notre nouvelle approche de la morphologie mathématique permet de retrouver les modèles discrets (algébriques) et ceux définis dans le domaine continu. Nous introduisons également une nouvelle classe d'opérateurs morphologiques pour le traitement des images en considérant des configurations non locales. Nous étendons également la morphologie basée sur les équations aux dérivées partielles et introduisons de nouveaux outils pour le traitement de données discrètes arbitraires.

Chapitre 4 : Équation eikonale

Dans ce chapitre, nous présentons une adaptation de l'équation eikonale en considérant cette équation sur des graphes pondérés. Cette nouvelle approche de la résolution de l'équation eikonale est basée sur une famille de gradients directionnels définis par des équations aux différences partielles. Notre approche étend les approches connues de résolution de cette équation pour le traitement de données

non organisées définies sur des domaines irréguliers, sans étape de discrétisation préalable.

L'équation eikonale permet de résoudre de nombreuses applications en traitement des images, en vision par ordinateur ou en informatique graphique. Parmi toutes ces applications, nous nous intéressons dans ce chapitre aux calculs de distances et introduisons des schémas non locaux permettant de segmenter des images. Nous introduisons également de nouvelles applications de cette équation dans des domaines inhabituels tels que le regroupement ou la classification de bases de données.

Chapitre

2

Graphes et opérateurs

Sommaire

2.1	Introduction	9
2.2	Définitions et notations	9
2.3	Structure et topologie	10
2.4	Opérateurs définis sur des graphes	17
2.5	Une nouvelle famille de différences	22
2.6	Une nouvelle famille de gradients	23
2.7	Propriétés et relations des gradients	24
2.8	Formulation générale du p -Laplacien	29
2.9	Propriétés et relations du p -Laplacien	30
2.10	Conclusion	31

Résumé

Ce premier chapitre introduit les notions nécessaires à la bonne compréhension de la suite de ce manuscrit. Nous rappellerons tout d'abord quelques définitions et notations sur les graphes, leur structure et leur construction. Ensuite, nous présenterons une nouvelle famille d'opérateurs définis sur des graphes et basés sur des équations aux différences partielles. Leurs propriétés seront également étudiées. Ces opérateurs sont utilisés dans les approches qui seront présentées dans la suite de ce manuscrit, pour réaliser des traitements des images et des données arbitraires de grande dimension.

Mots clés

Graphes pondérés, structure et topologie, construction des graphes, opérateurs discrets définis sur des graphes, gradients, p -Laplaciens.

Contenu du chapitre

- La section 2.2 rappelle des définitions et des notations essentielles de la théorie des graphes.
- La section 2.3 décrit la construction des graphes et présente leur intérêt dans la représentation des domaines discrets pour le traitement d'images et de données.
- La section 2.4 introduit des opérateurs sur graphes basés sur des équations aux différences partielles telles que le gradient ou le p -Laplacien.
- La section 2.10 conclut ce chapitre.

2.1 Introduction

Dans ce premier chapitre, nous introduisons les notions et les définitions nécessaires à la bonne compréhension de la suite de ce manuscrit.

Dans une première partie, nous rappelons de manière brève les notations et les définitions de la théorie des graphes. Ensuite, nous montrons que la structure de graphe est adaptée pour représenter des données discrètes définies sur des domaines organisés ou non. Nous expliquons également comment nous pouvons obtenir un graphe de voisinage à partir d'un ensemble de données.

Dans une seconde partie, à partir des définitions d'opérateurs définis sur des graphes proposés par [Bougleux *et al.*, 2007, Elmoataz *et al.*, 2008a], nous introduisons et étudions une nouvelle famille d'opérateurs différentiels tels que les différences et gradients directionnels et une classe de Laplaciens.

Nous montrerons également les propriétés de cette nouvelle famille d'opérateurs différentiels et les liens existants avec ceux déjà connus. Ces opérateurs, basés sur des équations aux différences partielles, sont les constituants principaux des différentes approches et thèmes que nous aborderons tout au long de ce manuscrit.

2.2 Définitions et notations

D'une manière générale, nous pouvons considérer que toutes données définies sur un domaine discret peuvent être représentées par un graphe pondéré $G = (V, E, w)$. Dans ce manuscrit, nous ne considérons que des graphes simples (sans arêtes multiples et sans boucles) non-orientés et pondérés [Diestel, 2005].

Un graphe non-orienté G est un couple (V, E) . L'ensemble fini V est composé d'éléments appelés sommets et E est un ensemble de paires non-ordonnées d'éléments où chaque élément est appelé arête. L'ensemble des arêtes E est un sous-ensemble de $V \times V$ défini par

$$E = \{uv \subseteq V \times V \text{ avec } u \neq v \text{ et } u, v \in V\} \quad (2.1)$$

avec uv désignant une arête appartenant à l'ensemble E . Si uv est une arête du graphe non-orienté G alors

- l'arête uv est incidente aux sommets u et v ;
- l'arête uv relie les sommets u et v ;
- les sommets u et v sont adjacents ou voisins.

Par la définition de l'ensemble des arêtes E , il ne peut y avoir d'arête qui relie un sommet à lui-même. Le graphe G est donc considéré sans boucle.

Le voisinage $\mathcal{N}(u)$, d'un sommet u , désigne l'ensemble des voisins de ce sommet. Il est défini par

$$\mathcal{N}(u) = \{v \in V \setminus \{u\} : uv \in E\} \quad (2.2)$$

Dans ce manuscrit, nous considérons des graphes pondérés $G = (V, E, w)$ où $w : E \rightarrow \mathbb{R}^+$ désigne la fonction de poids du graphe. Cette fonction de pondération est appliquée à l'ensemble des arêtes E . Pour une arête $uv \in E$, la notation w_{uv} désigne alors la pondération que nous appliquons à l'arête uv qui satisfait entre

autres

$$w_{uv} \begin{cases} \geq 0 & \text{si } v \in \mathcal{N}(u) \\ = 0 & \text{si } v \notin \mathcal{N}(u) \\ = w_{vu} \end{cases} . \quad (2.3)$$

Le degré $\deg_V : V \rightarrow \mathbb{R}^+$ d'un sommet u est la somme des poids des arêtes incidentes à ce sommet :

$$\deg_V(u) = \sum_{v \sim u} w_{uv} \quad (2.4)$$

où la notation $v \sim u$ signifie que les sommets u et v sont adjacents (ce qui revient à considérer l'ensemble des sommets voisins à u , c'est-à-dire $\forall v \in \mathcal{N}(u)$). Nous pouvons remarquer que dans le cas où le graphe G n'est pas pondéré, alors le degré d'un sommet correspond au nombre d'arêtes incidentes à ce sommet.

À partir de la définition du degré d'un sommet, le volume d'un ensemble de sommets $\mathcal{A} \subseteq V$, $\text{vol} : V \rightarrow \mathbb{R}^+$ se définit par

$$\text{vol}(\mathcal{A}) = \sum_{u \in \mathcal{A}} \deg_{\mathcal{A}}(u) = \sum_{u \in \mathcal{A}} \sum_{v \sim u} w_{uv} \quad (2.5)$$

où $\deg_{\mathcal{A}}$ correspond au degré des sommets relativement à l'ensemble \mathcal{A} .

2.3 Structure et topologie

La structure de graphe est naturellement adaptée pour représenter n'importe quel domaine discret. Ainsi, chaque donnée (ou point) de ce domaine est représentée par un sommet u du graphe. Le domaine considéré peut être organisé ou non organisé (c'est-à-dire avec ou sans à priori sur leur topologie) et les fonctions $f : V \subset \mathbb{R}^m \rightarrow \mathbb{R}^m$ définies sur les sommets du graphe représentent les données à traiter.

2.3.1 Domaines non organisés

Considérons le cas général d'un domaine non organisé. Dans cette situation, l'ensemble des points $V \subset \mathbb{R}^m$ de ce domaine peut être vu comme une fonction définie par $f : V \subset \mathbb{R}^m \rightarrow \mathbb{R}^m$. Chaque sommet du graphe est associé à un vecteur d'attributs (ou de caractéristiques) de \mathbb{R}^m .

Construire un graphe $G = (V, E, w)$ à partir des données de ce domaine consiste donc à définir l'ensemble des arêtes E en modélisant le voisinage de chaque sommet du graphe. Cette modélisation est basée sur des relations de similarité entre les vecteurs caractéristiques des données considérées.

La similarité entre deux sommets du graphe dépend généralement d'une mesure de distance $\mu : E \rightarrow \mathbb{R}^+$. Dans le cas des domaines non organisés, la mesure de distance usuelle est généralement la distance Euclidienne :

$$\mu(uv) = \sqrt{\sum_{i=1}^m (f_i(u) - f_i(v))^2} \quad \text{avec } f : V \subset \mathbb{R}^m \rightarrow \mathbb{R}^m . \quad (2.6)$$

D'autres mesures de distances peuvent être également considérées. Le choix de cette mesure dépend de l'application et des données considérées. Elle peut, par

exemple, être robuste aux transformations géométriques ou prendre en compte les variations des données à traiter. De plus, il n’y a pas de règle générale qui définisse la structure de graphe la plus adaptée à un domaine donné. Cette décision dépend de l’application considérée et des données à traiter [Hein *et al.*, 2008].

Dans ce manuscrit, sauf mention contraire, nous utilisons deux familles de graphes de voisinage :

- le graphe de ν -voisinage (aussi appelé graphe de ϵ -voisinage), noté G_ν .
- le graphe des k plus proches voisins, noté $k\text{-NNG}_\nu$ ¹

Pour une revue des graphes de voisinage, le lecteur intéressé pourra se référer à [Jaromczyk et Toussaint, 1992]

Graphe de ν -voisinage

Le graphe de ν -voisinage, noté $G_\nu = (V, E, w)$, est un graphe pondéré dont le voisinage pour un sommet donné u est défini comme étant l’ensemble des voisins dont la mesure de distance μ est inférieure ou égale au paramètre de seuillage ν . Le voisinage $\mathcal{N}_\nu(u)$ pour un sommet u est alors

$$\mathcal{N}_\nu(u) = \{v \in V \setminus \{u\} : \mu(uv) \leq \nu\} . \quad (2.7)$$

Lorsque le paramètre de seuillage $\nu = \infty$, nous obtenons alors un graphe complet, que nous noterons G_∞ . Le voisinage d’un sommet $u \in V$ se définit alors par

$$\mathcal{N}_\nu(u) = \{v \in V \setminus \{u\}\} . \quad (2.8)$$

Graphe des k plus proches voisins

Le graphe des k -plus proches voisins, noté $k\text{-NNG}_\nu = (V, E, w)$, est un graphe pondéré où un sommet u est connecté à un sommet v si celui-ci v fait parti des k plus proches voisins au sens de la distance μ dans un voisinage \mathcal{N}_ν . Cette définition produit un graphe dirigé. Dans ce manuscrit, nous utiliserons une version non dirigée de ce graphe. Nous considérons l’ensemble des arêtes défini tel que

$$E = \{uv : u \in \mathcal{N}_\nu(v) \text{ ou } v \in \mathcal{N}_\nu(u)\} . \quad (2.9)$$

Dans le cas particulier où le paramètre de seuillage $\nu = \infty$, le $k\text{-NNG}_\infty$ est calculé en prenant en compte l’ensemble des sommets $V \setminus \{u\}$ pour tous les sommets $u \in V$. Pour des raisons de simplicité de notations, nous noterons ce graphe par $k\text{-NNG}$ au lieu de $k\text{-NNG}_\infty$.

Remarque. La détermination du nombre k de voisins pour un sommet donné est dépendante de l’application considéré et aucune règle générale ne peut être donnée [Maier *et al.*, 2009].

La figure 2.1 présente deux exemples de $k\text{-NNG}$ sur un ensemble de données discrètes définies dans \mathbb{R}^2 , pour deux valeurs différentes du paramètre k .

¹de l’anglais *k-Nearest Neighbors Graph*.

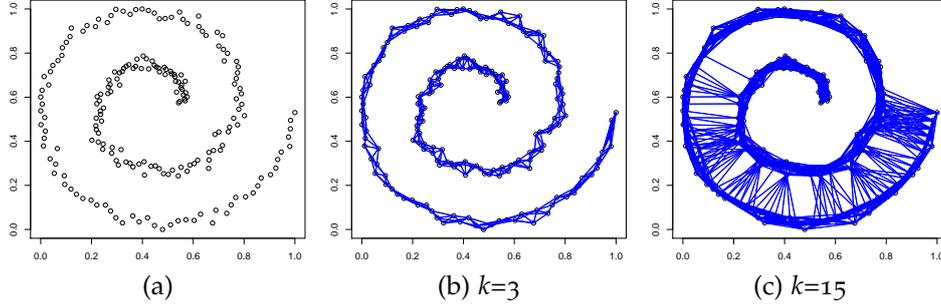


FIG. 2.1: Exemples de graphes des k plus proches voisins. (a) : données initiales dans \mathbb{R}^2 , (b) : 3-NNG, (c) : 15-NNG.

2.3.2 Domaines organisés

Considérons maintenant le cas des domaines organisés. Nous appelons domaines organisés, des données de type signal ou images (2D ou 3D) pour lesquelles l'organisation spatiale est connue a priori. Ainsi, ces domaines peuvent être vus comme des fonctions $f : V \subset \mathbb{Z}^q \rightarrow \mathbb{R}^m$ où $q = 1, 2$ ou 3 , correspondant aux différents cas mentionnés.

Dans le cas des domaines organisés, la mesure de distance μ utilisée pour construire un graphe de voisinage correspond généralement à une distance mesurée entre les coordonnées spatiales associées à chaque sommet. Nous pouvons par exemple citer la distance de « city block » (aussi appelée la distance de Manhattan) ou encore celle de Chebychev.

Cas des images Prenons le cas des images 2D. Si chaque sommet $u \in V$ est associé à un pixel de l'image (avec les coordonnées spatiales), alors les distances de « city-block » et de Chebychev entre deux sommets $u = (x_u, y_u)$ et $v = (x_v, y_v)$ de V sont respectivement

$$\mu(uv) = |x_u - x_v| + |y_u - y_v|, \quad (2.10)$$

$$\mu(uv) = \max(|x_u - x_v|, |y_u - y_v|). \quad (2.11)$$

Avec ces définitions et la famille des graphes de ν -voisinage, non retrouvons les tessellations régulières permettant de former des grilles 2D représentant les images :

- avec la distance de « city block » et $\nu \leq 1$, le graphe grille en 4-voisinage (connexité), noté G_0 ;
- avec la distance de Chebychev et $\nu \leq 1$, le graphe grille en 8-voisinage, noté G_1 .

La figure 2.2 montre deux exemples de graphe grille en 4 et en 8-voisinage.

De manière plus générale, les graphes grilles en $((2s + 1)^2 - 1)$ -voisinage sont obtenus à partir de la distance de Chebychev avec $\nu \leq s$ et $s \geq 1$. Cela correspond à ajouter des arêtes entre le pixel (sommet) central et les autres pixels contenus dans une fenêtre de taille $(2s + 1) \times (2s + 1)$.

Remarque. Les constructions précédemment décrites peuvent être appliquées aux cas des images 3D où chaque sommet du graphe représente un voxel de l'image.

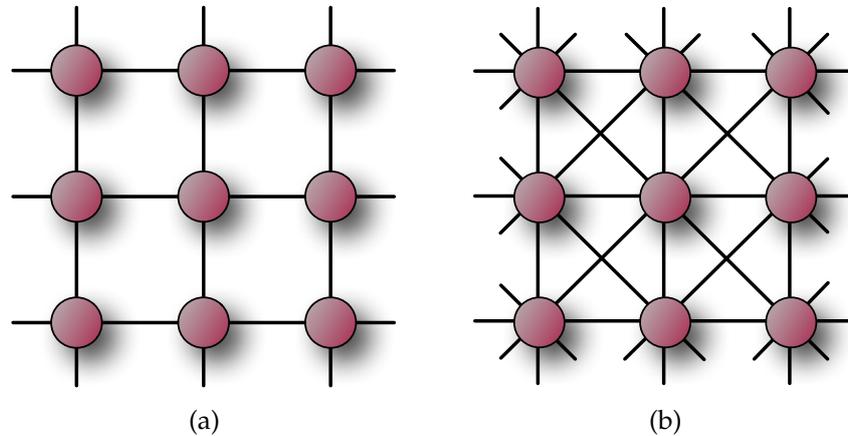


FIG. 2.2: Graphes grille représentant les images. (a) : un graphe grille en 4-voisinage, (b) : un graphe grille en 8-voisinage.

Une autre tessellation possible, permettant la représentation des images par un graphe, est la tessellation de Voronoï qui permet de trouver une partition (ensemble de régions connexes) pour une image donnée. À partir de cette partition, nous pouvons associer un graphe de voisinage : le graphe de Delaunay qui peut également être vu comme un graphe d'adjacence des régions (noté RAG^2) où chaque sommet correspond à une région de la partition. Le calcul d'une partition à partir d'une image peut se faire avec différentes méthodes. Parmi toutes celles existantes, nous pouvons par exemple citer la ligne de partage des eaux [Vincent et Soille, 1991, Cousty *et al.*, 2009a] ou encore les partitions d'énergie [Arbeláez et Cohen, 2004].

Finalement, un RAG peut être défini comme étant un graphe de v -voisinage (avec $v = 1$) où l'ensemble des arêtes E est défini avec la distance $\mu(uv) = 1$ si les régions u et v sont adjacentes et par $\mu(uv) = \infty$ si elles ne le sont pas.

La figure 2.3 montre un exemple de graphe d'adjacence de régions associé à une partition d'une image initiale. La figure 2.3b montre une partition où les contours de chaque région sont superposés en blanc. Chaque région de la partition est caractérisée par la couleur moyenne des pixels de l'image initiale qui la compose. La figure 2.3c montre le RAG associé à la partition. Les arêtes représentent les relations d'adjacences et les sommets sont caractérisés par la couleur de la région qui lui est associée.

Remarque. Dans la suite de ce manuscrit, nous proposerons plusieurs expérimentations basées sur cette représentation des images. Ce type de graphe permet de réduire la complexité des algorithmes (par rapport à un graphe construit à partir des pixels de l'image) et offre des solutions alternatives intéressantes en terme de temps de calcul.

²de l'anglais *Region Adjacency Graph*.

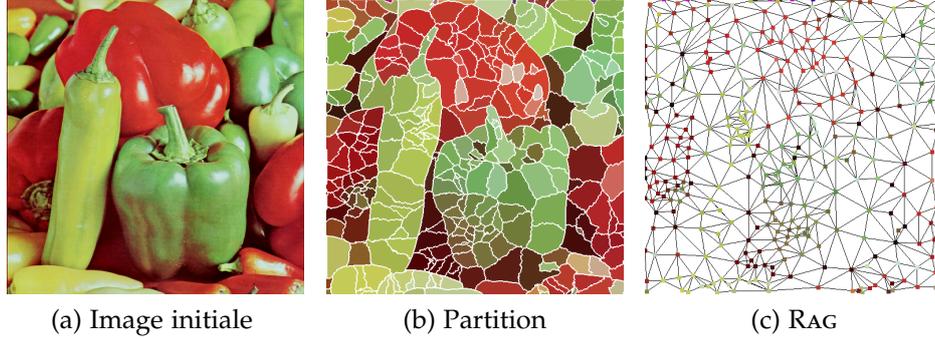


FIG. 2.3: Exemple de graphe d'adjacence des régions associé à une partition d'une image. (a) : image initiale. La partition (b) est représentée avec les contours de ses régions (en blanc) et chaque région est caractérisée par la couleur moyenne des pixels de l'image initiale qui la compose. Chaque sommet du graphe (c) correspond à une région de la partition et est représenté par sa couleur.

2.3.3 Similarités entre les sommets

Les similarités entre les sommets d'un graphe $G = (V, E, w)$ peuvent être incorporées dans la fonction de poids w des arêtes E par le biais d'une fonction de similarité $g : E \rightarrow \mathbb{R}^+$ de telle manière que $w(uv) = g(uv)$ pour une arête $uv \in E$.

Vecteurs d'attributs

Les similarités entre les sommets du graphe reposent sur une comparaison des attributs associés à chacun de ses sommets. Ces attributs dépendent généralement d'une fonction initiale $f^0 : V \subset \mathbb{R}^m \rightarrow \mathbb{R}^m$ définie en chaque sommet. Ainsi, nous définissons le vecteur d'attributs $\mathcal{F}(f^0, u) \in \mathbb{R}^m$ en chaque sommet $u \in V$ du graphe dépendant de la fonction initiale f^0 .

Il existe différents choix possibles pour définir le vecteur d'attributs initial. La définition la plus simple étant de considérer la fonction initiale f^0 elle-même et de définir $\mathcal{F}(f^0, \cdot) = f^0$. D'autres définitions ont été proposées notamment dans le cas particulier des images 2D.

Cas particulier des images 2D Dans le contexte du traitement des images, une grande famille de vecteurs d'attributs est définie par les patches de l'image. Dans ce cas, le vecteur d'attributs \mathcal{F} est défini en un sommet $u \in V$ à l'aide du voisinage \mathcal{N}_v tel que

$$\mathcal{F}_v(f^0, u) = (f^0(v) \in V : v \in \mathcal{N}_v(u) \cup \{u\})^T \quad (2.12)$$

Dans le cas d'une image en niveaux de gris, le vecteur d'attributs $\mathcal{F}_v(f^0, u)$ est de taille $(2v + 1)^2$ et correspond aux valeurs de la fonction f^0 contenues dans une fenêtre de taille $(2v + 1) \times (2v + 1)$ centrée sur le sommet u . Dans le cas d'une image multi variée dans \mathbb{R}^m alors le vecteur d'attributs associé aux sommets du graphe sera de taille $m \times (2v + 1)$ [Tschumperlé, 2009].

La figure 2.4 illustre la définition des vecteurs d'attributs par des patches pour une image en niveaux de gris. Le pixel central (entouré en rouge), au lieu d'être

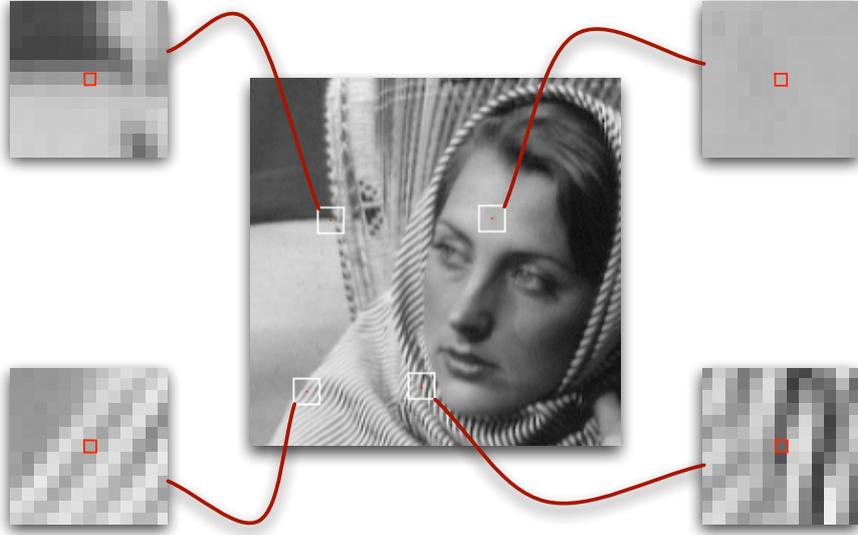


FIG. 2.4: Quatre exemples de patches comme vecteur d'attributs caractérisant les sommets du graphes. Ici, le pixel central (en rouge) est caractérisé par un vecteur de niveaux de gris défini grâce à un patch de taille 13×13 .

uniquement défini par sa valeur de niveau gris, est caractérisé par un vecteur défini par un patch de l'image. Dans le cas de la figure 2.4, les patches sont définis comme des fenêtres symétriques de taille 13×13 . Cette figure présente quatre patches différents de l'image. Avec cette nouvelle représentation, l'espace des patches est alors un espace de grande dimension (\mathbb{R}^{169}).

Afin de pouvoir calculer les poids entre les sommets du graphe qui sont définis par des patches de l'image, la fonction de distance $\rho : \mathbb{R}^{m \times p} \times \mathbb{R}^{m \times p} \rightarrow \mathbb{R}^+$ avec p la taille du vecteur d'attributs $\mathcal{F}_v(f^0, \cdot)$, peut être utilisée

$$\rho^2(\mathcal{F}_v(f^0, u), \mathcal{F}_v(f^0, v)) = \sum_{x=-v}^v \sum_{y=-v}^v \mathcal{G}_a((x, y)) \cdot \|f^0(x_u + x, y_u + y) - f^0(x_v + x, y_v + y)\|_2^2 \quad (2.13)$$

avec \mathcal{G}_a un noyau Gaussien de variance a ; $\mathcal{F}_v(f^0, u)$ et $\mathcal{F}_v(f^0, v)$ correspondent aux patches centrés sur les deux sommets (pixels) $u = (x_u, y_u)$ et $v = (x_v, y_v)$.

Ces vecteurs d'attributs ont été initialement proposés dans le cadre de la synthèse de texture [Efros et Leung, 1999] et repris récemment pour le filtrage et la restauration d'image et de maillage [Gilboa et Osher, 2007a, Buades *et al.*, 2008, Elmoataz *et al.*, 2008a, Bougleux *et al.*, 2009].

Dans le contexte du traitement des images, ces méthodes sont dites « non locales ». Ces derniers travaux ont mis en évidence l'efficacité et la supériorité des méthodes non locales basées sur des patches par rapport aux traitement locaux, en particulier pour capturer les structures fines et répétitives de l'image. Nous voyons ici que la notion de non localité inclut deux notions :

- le voisinage de recherche des candidats les plus similaires par rapport au pixel considéré ;

- le vecteur d’attributs pour comparer ces candidats.

Tel que l’introduisent Buades *et al.* [Buades *et al.*, 2008], la non localité consiste à comparer un pixel à l’ensemble des patchs existants dans l’image. Du fait de l’évidente complexité spatiale et temporelle d’une telle méthode pour le traitement des images, la notion de non localité est usuellement remplacée en pratique par celle de semi localité. Cette dernière consiste à comparer, non plus l’ensemble des patchs de l’image pour un pixel donné, mais uniquement ceux se trouvant dans une fenêtre de recherche centrée sur ce dernier. Des travaux permettant de déterminer, par méthodes statistiques et de manière optimale et automatique, la taille des fenêtres de recherche et la taille des patchs ont été proposés. Nous pouvons également citer de récentes méthodes et algorithmes, basés sur des structures de données particulières, cherchant à réduire la complexité des traitements non locaux des images. Le lecteur intéressé par ces travaux pourra par exemple se référer aux articles suivants ainsi qu’à leurs références associées [Kervrann et Boulanger, 2008, Darbon *et al.*, 2008, Brox *et al.*, 2008].

Comme nous l’avons mentionné, les méthodes non locales ont montré leur efficacité pour le filtrage et la restauration des images. Néanmoins, du fait de la définition des patchs, ces méthodes possèdent des limites, en particulier leur sensibilité aux transformations géométriques.

Finalement, nous pouvons voir que les configurations non locales (comme le définit Buades) n’existent plus dès lors que l’on considère des graphes. En effet, ces notions disparaissent et s’expriment uniquement en terme de topologie du graphe (le voisinage des sommets) et de poids des arêtes (distances entre les attributs des sommets). Les traitements non locaux des images deviennent alors simplement des traitements locaux sur un graphe.

Fonctions de poids

L’utilisation d’une fonction de poids rend compte des similarités entre les données. Cela dépend de l’application considérée et des données elles-même. Ainsi, nous ne pouvons donner de règles générales permettant de déterminer si une fonction de poids est plus pertinente qu’une autre. Le lecteur intéressé peut se référer à [Grady et Jolly, 2008] pour une étude des fonctions de poids et de la topologie des graphes intervenant dans la segmentation des images.

Avec la définition des vecteurs d’attributs en chaque sommet d’un graphe, nous pouvons utiliser les définitions des fonctions de poids suivantes (le lecteur remarquera que cette liste n’est pas exhaustive, elle ne reflète que les différentes fonctions qui seront utilisées dans la suite de ce manuscrit).

Soient un graphe pondéré $G = (V, E, w)$, la mesure de distance ρ permettant de comparer les vecteurs d’attributs $\mathcal{F}(f^0, u)$ et $\mathcal{F}(f^0, v)$ des sommets $u, v \in V$ et $f^0 : V \subset \mathbb{R}^m \rightarrow \mathbb{R}^m$ une fonction initiale. Pour une arête $uv \in E$ donnée, nous

considérons les fonctions de poids suivantes :

$$g_1(uv) = 1 \text{ (poids constant) ,} \quad (2.14)$$

$$g_2(uv) = \left(\rho(\mathcal{F}(f^0, u), \mathcal{F}(f^0, v)) + \epsilon \right)^{-1} , \quad (2.15)$$

$$g_3(uv) = 1 - \left(\rho(\mathcal{F}(f^0, u), \mathcal{F}(f^0, v)) / \max_{\mu} \mu(\mathcal{F}(f^0, \cdot), \mathcal{F}(f^0, \cdot)) \right) , \quad (2.16)$$

$$g_4(uv) = \exp\left(-\rho(\mathcal{F}(f^0, u), \mathcal{F}(f^0, v))^2 / \sigma^2\right) . \quad (2.17)$$

Remarque. En utilisant la distance (2.13) dans la fonction de pondération g_4 nous retrouvons l'expression de la fonction de poids utilisée par le filtre à moyennes non locales pour les images (le NL Means³) [Buades et al., 2008].

La figure 2.5 montre des exemples de distributions de poids et l'influence de l'utilisation des patches dans le contexte des images. Les distributions de poids sont estimées relativement au pixel central de chaque image (représenté par un point blanc). Ce dernier est connecté à tous les autres pixels. Les résultats sont obtenus avec la fonction de poids g_4 . Lors de l'utilisation de patches, les comparaisons sont effectuées avec la mesure de distance (2.13).

Les résultats montrent que l'utilisation d'une fonction de poids (avec ou sans patches) permet de capturer les détails des images et de rendre compte des similarités et des interactions entre les pixels.

L'utilisation des patches permet un filtrage implicite de l'image et capture mieux les similarités et certains détails (par rapport aux cas sans patches). Dans les zones quasi constantes (la première ligne de la figure 2.5), la distribution des poids obtenue avec des patches se comporte comme une convolution par une Gaussienne (résultant de la fonction de poids et de la mesure de distance). Dans l'image comportant un contour (la deuxième ligne de la figure 2.5), la distribution de poids tend à suivre ce dernier. Finalement, dans l'image comportant des motifs répétitifs et oscillants (la dernière ligne de la figure 2.5), les patches permettent de mieux capturer les similarités entre les éléments comportant des textures fines.

2.4 Opérateurs définis sur des graphes

Dans cette section, nous rappelons les définitions des opérateurs différentiels sur graphes introduites par [Bougleux *et al.*, 2007, Elmoataz *et al.*, 2008a]. Ces définitions peuvent être vues comme étant les versions sur graphes pondérés des opérateurs de géométrie différentielle classiques définis dans le domaine continu. Des démarches similaires ont été proposées en apprentissage, en segmentation d'images, en analyse fonctionnelle sur graphe ou encore en théorie spectrale des graphes. Le lecteur intéressé pourra se référer, par exemple, aux articles suivants [Chung, 1997, Bensoussan et Menaldi, 2003, Grady, 2006, Zhou et Schölkopf, 2006] et aux références associées.

³de l'anglais *Non Local Means*.

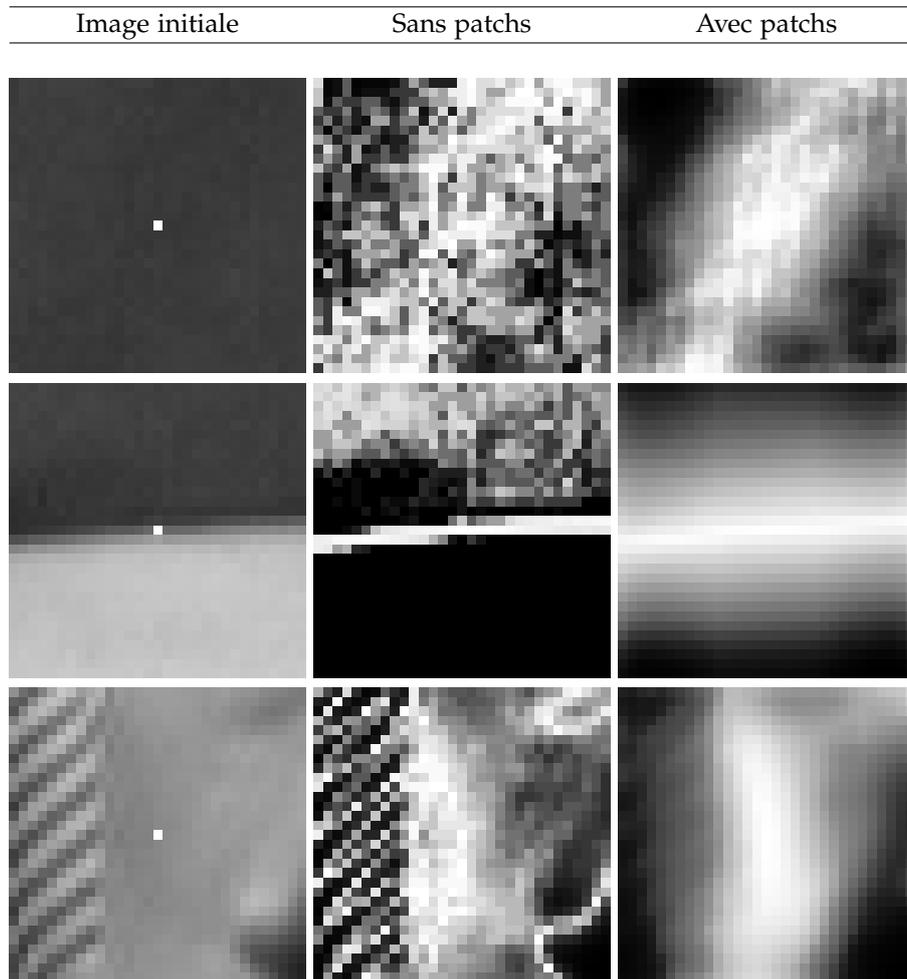


FIG. 2.5: Exemples de distribution des poids sur des images et influence de l'utilisation des patches comme vecteurs d'attributs. Les distributions sont estimées pour le pixel central (représenté par le pixel blanc) de l'image initiale. Le graphe considéré est un graphe reliant le pixel central à tous les autres pixels.

2.4.1 Fonctions définies sur des graphes

Soit $f : V \rightarrow \mathbb{R}$ une fonction scalaire qui associe à chaque sommet $u \in V$ une valeur réelle $f(u)$. Pour un graphe $G = (V, E, w)$, cette fonction peut être vue comme un vecteur colonne défini comme $f = (f(u))_{u \in V}^T$. Par analogie avec l'espace des fonctions continues, l'intégrale discrète de la fonction f sur G est définie par

$$\int_G f = \sum_{u \in V} f(u) . \quad (2.18)$$

Nous définissons l'espace des fonctions sur des graphes. Soit $\mathcal{H}(V)$ l'espace de Hilbert des fonctions à valeurs réelles définies sur les sommets V du graphe G .

L'espace de Hilbert $\mathcal{H}(V)$ est muni du produit scalaire

$$\langle f, h \rangle_{\mathcal{H}(V)} = \sum_{u \in V} f(u)h(u) \quad \text{avec } f, h : V \rightarrow \mathbb{R} \quad (2.19)$$

et la norme \mathcal{L}_2 , $\|f\|_2 = \sqrt{\langle f, f \rangle_{\mathcal{H}(V)}}$.

De la même manière, soit $\mathcal{H}(E)$ l'espace des fonctions $F : E \rightarrow \mathbb{R}^+$ définies sur les arêtes et également muni du produit scalaire

$$\langle F, H \rangle_{\mathcal{H}(V)} = \sum_{u \in V} \sum_{v \sim u} F(uv)H(uv) \quad \text{avec } F, H : E \rightarrow \mathbb{R} \quad (2.20)$$

et la norme \mathcal{L}_2 , $\|F\|_2 = \sqrt{\langle F, F \rangle_{\mathcal{H}(E)}}$.

Remarque. Dans ce chapitre, nous ne considérons que des fonctions scalaires $f : V \rightarrow \mathbb{R}$ afin de simplifier les notations et la compréhension de ce chapitre. Néanmoins, l'extension des approches proposées par la suite est valable pour le cas des fonctions vectorielles $f : V \rightarrow \mathbb{R}^m$.

2.4.2 Opérateur de différence et son adjoint

L'opérateur de différence pondérée d_w d'une fonction $f \in \mathcal{H}(V)$ est une application de $\mathcal{H}(V) \rightarrow \mathcal{H}(E)$. Elle se définit en un sommet u selon une arête $uv \in E$ [Bougleux et al., 2007, Elmoataz et al., 2008a] :

$$(d_w f)(uv) = \frac{f(v) - f(u)}{h_{uv}} \quad (2.21)$$

où h_{uv} un pas d'échantillonnage entre $f(u)$ et $f(v)$ dépendant du poids w entre les sommets u et v de V . Ce pas peut être vu comme la version discrète sur graphe du taux d'accroissement d'une fonction continue.

Par la suite, nous utilisons un pas h_{uv} faisant intervenir le poids w_{uv} d'une arête uv telle que $h_{uv} = 1/\sqrt{w_{uv}}$, donnant ainsi la définition suivante de l'opérateur (2.21) pour une arête $uv \in E$ et une fonction $f \in \mathcal{H}(V)$

$$(d_w f)(uv) = \sqrt{w_{uv}}(f(v) - f(u)) . \quad (2.22)$$

Ce choix permet de retrouver les versions discrètes et pondérées des opérateurs gradients et Laplace définis sur des graphes et d'intégrer les interactions non locales dans le contexte des images.

Par analogie avec le calcul différentiel en continu, nous pouvons voir la différence pondérée comme la version discrète définie sur un graphe de l'opérateur de dérivée appliquée à une arête $uv \in E$:

$$\left. \frac{\partial f}{\partial uv} \right|_u = \partial_v f(u) = (d_w f)(uv) . \quad (2.23)$$

Cette définition est consistante avec la définition de la dérivée d'une fonction continue définie dans l'espace Euclidien et elle possède également les propriétés suivantes

$$\partial_v f(u) = \begin{cases} -\partial_u f(v) \\ 0 & \text{si } v = u \\ 0 & \text{si } f(v) = f(u) \end{cases} . \quad (2.24)$$

Remarque. Avec un graphe représentant une grille régulière et une fonction de poids constante (g_1), la différence pondérée correspond à la définition de la différence classique en analyse numérique.

De plus, il existe d'autres expressions possibles de la différence, par exemple pour une arête $uv \in E$ [Zhou et Schölkopf, 2004],

$$(\mathbf{d}_w f)(uv) = \sqrt{w_{uv}} \frac{f(v)}{\sqrt{\deg(v)}} - \sqrt{w_{uv}} \frac{f(u)}{\sqrt{\deg(u)}} \quad (2.25)$$

Cette dernière définition ne possède pas les mêmes propriétés que la définition (2.22) : elle n'est pas nulle quand la fonction f est localement constante [Hein et al., 2007].

Opérateur adjoint

L'opérateur adjoint $\mathbf{d}_w^* : \mathcal{H}(E) \rightarrow \mathcal{H}(V)$ de l'opérateur de différence pondérée \mathbf{d}_w est défini par

$$\langle \mathbf{d}_w f, H \rangle_{\mathcal{H}(E)} = \langle f, \mathbf{d}_w^* H \rangle_{\mathcal{H}(V)} \quad (2.26)$$

pour toutes les fonctions $f \in \mathcal{H}(V)$ et $H \in \mathcal{H}(E)$. En utilisant les produits scalaires définis dans (2.19) et (2.20), l'expression de l'opérateur adjoint d'une fonction $F \in \mathcal{H}(E)$ est [Bogleux et al., 2007, Elmoataz et al., 2008a]

$$(\mathbf{d}_w^* F)(u) = \sum_{v \sim u} \sqrt{w_{uv}} (F(vu) - F(uv)) \quad (2.27)$$

où la notation $v \sim u$ signifie que v est voisin de u , c'est-à-dire $v \in \mathcal{N}(u)$.

Par analogie avec les opérateurs différentiels continus, l'opérateur de divergence d'une fonction $F \in \mathcal{H}(E)$ est

$$\operatorname{div} F = -\mathbf{d}_w^* F . \quad (2.28)$$

2.4.3 Opérateur gradient

L'opérateur gradient pondéré $\nabla_w : \mathcal{H}(V) \rightarrow \mathbb{R}^m$ d'une fonction $f \in \mathcal{H}(V)$ en un sommet $u \in V$ est défini comme étant le vecteur de toutes les différences pour l'ensemble des arêtes de E

$$(\nabla_w f)(u) = ((\mathbf{d}_w f)(uv))_{uv \in E}^T . \quad (2.29)$$

La définition de ce gradient ne dépend pas de la structure du graphe. Par conséquent, le gradient aura toujours la même expression quelles que soient les données qui seront représentées par le graphe considéré. Cet opérateur est de premier ordre, défini localement en un sommet $u \in V$ et dépend de son voisinage $\mathcal{N}(u)$. Afin de mesurer la variation locale en un sommet $u \in V$, différentes normes peuvent être utilisées telles que les normes \mathcal{L}_q . Ainsi pour $0 < q < +\infty$,

$$\|(\nabla_w f)(u)\|_q = \left[\sum_{v \sim u} w_{uv}^{q/2} |f(v) - f(u)|^q \right]^{1/q} \quad (2.30)$$

et pour $q = \infty$

$$\|(\nabla_w f)(u)\|_\infty = \max_{v \sim u} (\sqrt{w_{uv}} |f(v) - f(u)|) \quad (2.31)$$

Remarque. Le gradient pondéré défini sur un graphe intègre naturellement les interactions non locales dans sa formulation. Avec la définition de la fonction de poids w (pour $u \in V$, $w_{uv} = 0$ pour tout $v \notin \mathcal{N}(u)$) les normes (2.30) et (2.31) peuvent se réécrire

$$\|(\nabla_w f)(u)\|_q = \left[\sum_{v \in V} w_{uv}^{q/2} |f(v) - f(u)|^q \right]^{1/q} \quad (2.32)$$

$$\|(\nabla_w f)(u)\|_\infty = \max_{v \in V} \sqrt{w_{uv}} |f(v) - f(u)| . \quad (2.33)$$

où la notation $v \sim u$ est remplacée par $v \in V$.

2.4.4 p -Laplacien isotrope

Une expression du p -Laplacien pondéré isotrope peut être obtenue [Elmoataz *et al.*, 2008a, Bougleux *et al.*, 2009] à partir de l'opérateur de différence (2.22) et de son adjoint (2.26).

Soit p un nombre réel tel que $0 < p < +\infty$. Le p -Laplacien pondéré isotrope $\Delta_{w,p}^i : \mathcal{H}(V) \rightarrow \mathcal{H}(V)$ d'une fonction $f \in \mathcal{H}(V)$ est défini par

$$\Delta_{w,p}^i f = \mathbf{d}_w^* (\|\nabla_w f\|_2^{p-2} \mathbf{d}_w f) . \quad (2.34)$$

L'expression de cet opérateur en un sommet $u \in V$, pour la fonction f est

$$(\Delta_{w,p}^i f)(u) = \sum_{v \sim u} \sqrt{w_{uv}} \left(\frac{(\mathbf{d}_w f)(vu)}{\|(\nabla_w f)(v)\|_2^{2-p}} - \frac{(\mathbf{d}_w f)(uv)}{\|(\nabla_w f)(u)\|_2^{2-p}} \right) \quad (2.35)$$

$$= \sum_{v \sim u} w_{uv} \left(\frac{f(u) - f(v)}{\|(\nabla_w f)(v)\|_2^{2-p}} - \frac{f(v) - f(u)}{\|(\nabla_w f)(u)\|_2^{2-p}} \right) \quad (2.36)$$

$$= \sum_{v \sim u} w_{uv} (\|(\nabla_w f)(v)\|_2^{p-2} + \|(\nabla_w f)(u)\|_2^{p-2}) (f(u) - f(v)) . \quad (2.37)$$

Propriétés du p -Laplacien isotrope

Le p -Laplacien isotrope (2.34) est relié à différentes formulations de l'opérateur de Laplace [Elmoataz *et al.*, 2008a, Bougleux *et al.*, 2009].

Dans le cas où le paramètre $p = 2$, le p -Laplacien isotrope (2.34) se réduit à l'expression suivante

$$\Delta_{w,2}^i f = \mathbf{d}_w^* (\mathbf{d}_w f) = \frac{1}{2} \Delta_w f \quad (2.38)$$

définissant ainsi le Laplacien pondéré d'une fonction $f \in \mathcal{H}(V)$ sur graphe. En un sommet $u \in V$ son expression est

$$(\Delta_w f)(u) = 2 \sum_{v \sim u} w_{uv} (f(u) - f(v)) \quad (2.39)$$

Ce Laplacien est linéaire et correspond au Laplacien combinatoire [Chung, 1997].

Dans le cas où le paramètre $p = 1$, le p -Laplacien isotrope (2.34) se réduit à l'expression suivante

$$\Delta_{w,1}^i f = \mathbf{d}_w^* (\|\nabla_w f\|^{-1} \mathbf{d}_w f) = \kappa_w f \quad (2.40)$$

définissant ainsi l'opérateur de courbure pondéré d'une fonction $f \in \mathcal{H}(V)$ sur graphe. En un sommet $u \in V$ son expression est

$$(\kappa_w f)(u) = \sum_{v \sim u} w_{uv} \left(\frac{1}{\|(\nabla_w f)(v)\|_2} + \frac{1}{\|(\nabla_w f)(u)\|_2} \right) (f(u) - f(v)) . \quad (2.41)$$

L'opérateur de courbure sur graphe peut être vu comme étant la version discrète de l'opérateur de courbure d'une fonction continue $f : \Omega \subset \mathbb{R}^m \rightarrow \mathbb{R}$ définie comme

$$\kappa f = d_w^* \left(\frac{\nabla f}{\|\nabla f\|_2} \right) = - \operatorname{div} \left(\frac{\nabla f}{\|\nabla f\|_2} \right) \quad (2.42)$$

Dans le cas d'un graphe non pondéré, l'expression de la courbure peut être considérée comme étant la version discrète de l'opérateur de courbure utilisé dans le contexte du traitement des images [Osher et Shen, 2000, Chan *et al.*, 2001].

Finalement, le p -Laplacien isotrope peut être vu comme étant une généralisation du Laplacien combinatoire et de l'opérateur de courbure

$$\Delta_{w,p}^i f = - \operatorname{div} \left(\frac{\nabla f}{\|\nabla f\|_2^{2-p}} \right) \quad (2.43)$$

2.5 Une nouvelle famille de différences

Dans cette section, une nouvelle famille de différences pondérées définies sur des graphes est proposée.

À partir de la définition de la différence pondérée sur graphe (2.22) nous définissons deux nouvelles différences : les différences directionnelles pondérées.

Soit $d_w^+ : \mathcal{H}(V) \rightarrow \mathcal{H}(E)$ la différence pondérée externe basée sur l'opérateur max et $d_w^- : \mathcal{H}(V) \rightarrow \mathcal{H}(E)$ la différence pondérée interne basée sur l'opérateur min. Elles se définissent en un sommet $u \in V$ selon une arête $uv \in E$ telle que

$$(d_w^+ f)(uv) = \sqrt{w_{uv}} \max(f(v), f(u)) - f(u) \quad (2.44)$$

$$(d_w^- f)(uv) = \sqrt{w_{uv}} \left(f(u) - \min(f(u), f(v)) \right) . \quad (2.45)$$

Remarque. Ces deux nouveaux opérateurs constituent la base de nos approches de la morphologie mathématique et de la résolution de l'équation eikonale basées sur des graphes développées dans les chapitres 4 et 5.

Propriété 2.5.1. Ces deux différences directionnelles pondérées possèdent les propriétés suivantes et sont notamment reliées à la définition générale de la différence pondérée (2.22).

$$(d_w^+ f)(uv) = \max(0, (d_w f)(uv)) \quad (2.46)$$

$$(d_w^- f)(uv) = - \min(0, (d_w f)(uv)) \quad (2.47)$$

$$(d_w^- f)(uv) = (d_w^+ f)(vu) . \quad (2.48)$$

Démonstration. La propriété (2.46) s'obtient avec l'identité suivante

$$\max(a, b) - b = \max(0, a - b) . \quad (2.49)$$

En effet,

$$(\mathbf{d}_w^+ f)(uv) = \sqrt{w_{uv}} \max(f(v), f(u)) - f(u) \quad (2.50)$$

$$= \max\left(0, \sqrt{w_{uv}}(f(v) - f(u))\right) \quad (2.51)$$

$$= \max(0, (\mathbf{d}_w f)(uv)) . \quad (2.52)$$

De manière similaire, la propriété (2.47) s'obtient avec l'identité suivante

$$a - \min(a, b) = -\min(0, b - a) . \quad (2.53)$$

En effet,

$$(\mathbf{d}_w^- f)(uv) = \sqrt{w_{uv}} \left(f(u) - \min(f(u), f(v)) \right) \quad (2.54)$$

$$= -\min(0, \sqrt{w_{uv}}(f(v) - f(u))) \quad (2.55)$$

$$= -\min(0, (\mathbf{d}_w f)(uv)) . \quad (2.56)$$

Finalement, avec l'identité

$$-\min(0, b - a) = \max(0, a - b) \quad (2.57)$$

nous obtenons la propriété (2.48)

$$(\mathbf{d}_w^- f)(uv) = -\min(0, \sqrt{w_{uv}}(f(u) - f(v))) \quad (2.58)$$

$$= \max(0, \sqrt{w_{uv}}(f(v) - f(u))) \quad (2.59)$$

$$= (\mathbf{d}_w^- f)(vu) . \quad (2.60)$$

□

Comme pour la différence pondérée, nous avons également les opérateurs de dérivées directionnelles suivants appliqués à une arête $uv \in E$

$$\partial_v^+ f(u) = (\mathbf{d}_w^+ f)(uv) \quad (2.61)$$

$$\partial_v^- f(u) = (\mathbf{d}_w^- f)(uv) . \quad (2.62)$$

2.6 Une nouvelle famille de gradients

Dans cette section, une nouvelle famille de gradients pondérés définis sur des graphes est proposée. Ces gradients sont basés sur les différences directionnelles (2.44) et (2.45) : les gradients directionnels pondérés .

Les opérateurs gradient pondéré externe (noté ∇_w^+) et gradient pondéré interne (noté ∇_w^-) d'une fonction $f \in \mathcal{H}(V)$ sont respectivement définis par

$$(\nabla_w^+ f)(u) = ((\mathbf{d}_w^+ f)(uv))_{uv \in E}^T \quad (2.63)$$

$$(\nabla_w^- f)(u) = ((\mathbf{d}_w^- f)(uv))_{uv \in E}^T . \quad (2.64)$$

Ces définitions sont consistantes avec les définitions des gradients d'une fonction continue et peuvent être vus comme leurs versions discrètes définis sur des graphes.

Comme pour le gradient pondéré symétrique (2.29), ces définitions ne dépendent pas de la structure du graphe et auront donc toujours les mêmes expressions quelles que soient les données à traiter.

Afin de mesurer les variations locales en un sommet $u \in V$, les normes \mathcal{L}_q suivantes sont utilisées. Ainsi, pour des valeurs de $0 < q < +\infty$,

$$\|(\nabla_w^+ f)(u)\|_q = \left[\sum_{v \sim u} w_{uv}^{q/2} |\max(0, f(v) - f(u))|^q \right]^{1/q} \quad (2.65)$$

$$\|(\nabla_w^- f)(u)\|_q = \left[\sum_{v \sim u} w_{uv}^{q/2} |\min(0, f(v) - f(u))|^q \right]^{1/q}. \quad (2.66)$$

Lorsque $q = \infty$,

$$\|(\nabla_w^+ f)(u)\|_\infty = \max_{v \sim u} \sqrt{w_{uv}} |\max(0, f(v) - f(u))| \quad (2.67)$$

$$\|(\nabla_w^- f)(u)\|_\infty = \max_{v \sim u} \sqrt{w_{uv}} |\min(0, f(v) - f(u))|. \quad (2.68)$$

Comme pour le cas du gradient pondéré symétrique, les définitions des gradients pondérés directionnels prennent directement en compte les interactions non locales entre les données traitées. Ainsi, les normes précédentes en un sommet $u \in V$ peuvent se réécrire de manière non locale par

$$\|(\nabla_w^+ f)(u)\|_q = \left[\sum_{v \in V} w_{uv}^{q/2} |\max(0, f(v) - f(u))|^q \right]^{1/q} \quad (2.69)$$

$$\|(\nabla_w^- f)(u)\|_q = \left[\sum_{v \in V} w_{uv}^{q/2} |\min(0, f(v) - f(u))|^q \right]^{1/q} \quad (2.70)$$

$$\|(\nabla_w^+ f)(u)\|_\infty = \max_{v \in V} \sqrt{w_{uv}} |\max(0, f(v) - f(u))| \quad (2.71)$$

$$\|(\nabla_w^- f)(u)\|_\infty = \max_{v \in V} \sqrt{w_{uv}} |\min(0, f(v) - f(u))| \quad (2.72)$$

avec $w_{uv} = 0$ pour tout $v \neq \mathcal{N}(u)$, $\forall u \in V$ et où la notation $v \sim u$ est remplacée par $v \in V$.

2.7 Propriétés et relations des gradients

Nous présentons dans cette section des propriétés des gradients pondérés définis dans les sections précédentes. Nous montrons également les liens existants entre ces gradients et les bords d'un graphe. Pour cela, nous utilisons les définitions des bords d'un graphe, une fonction indicatrice et la décomposition en ensembles de niveaux d'une fonction $f \in \mathcal{H}(V)$.

Soit $\mathcal{A} \subset V$ un ensemble de sommets d'un graphe pondéré $G = (V, E, w)$. Pour tous sommets $u \in \mathcal{A}$, il existe un sommet $v \in \mathcal{A}$ et une arête $uv \in E$. Avec cette définition, nous notons respectivement $\partial^+ \mathcal{A}$ le bord externe et $\partial^- \mathcal{A}$ le bord interne de l'ensemble de sommets \mathcal{A} . Ces bords sont respectivement définis par

$$\partial^+ \mathcal{A} = \{u \notin \mathcal{A} : \exists v \in \mathcal{A} \text{ avec une arête } uv \in E\} \quad (2.73)$$

$$\partial^- \mathcal{A} = \{u \in \mathcal{A} : \exists v \notin \mathcal{A} \text{ avec une arête } uv \in E\}. \quad (2.74)$$

La figure 2.6 montre les bords externe et interne (représentés par les symboles « + » et « - ») d'un ensemble de sommets (sommets représentés en bleu) sur deux graphes différents : un graphe grille en 4-voisinage et un graphe non orienté quelconque.

Remarque. À partir des définitions des bords externe et interne, les bords de V (l'ensemble des sommets d'un graphe) ne peuvent être définis. Dans ce cas, nous considérons qu'ils sont donnés ou connus.

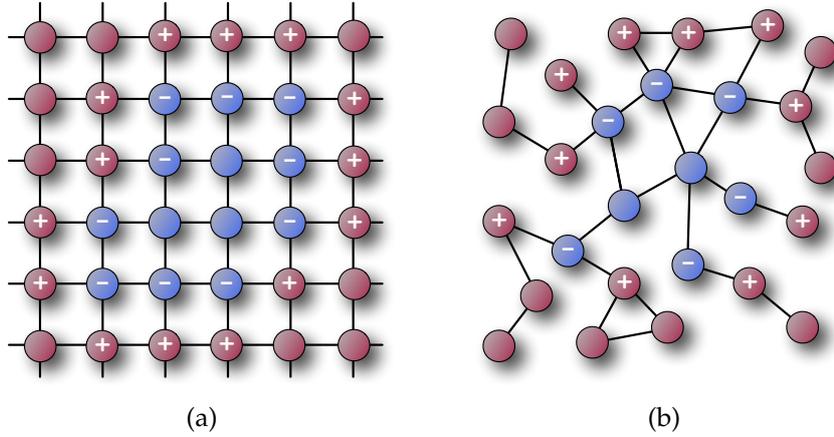


FIG. 2.6: Illustrations des bords d'un graphe sur deux graphes différents. (a) : un graphe grille en 4-voisinage ; (b) : un graphe non orienté quelconque. Les sommets en bleu correspondent à l'ensemble \mathcal{A} . Les sommets marqués par le signe « - » correspondent au bord interne $\partial^- \mathcal{A}$ et ceux marqués par le signe « + » correspondent au bord externe $\partial^+ \mathcal{A}$.

Soit $\chi_{(S)} : V \rightarrow \{0, 1\}$ une fonction indicatrice (caractéristique). Pour un sommet $u \in V$

$$\chi_{(S)}(u) = \begin{cases} 0 & \text{si } u \notin S \\ 1 & \text{si } u \in S \end{cases} \quad (2.75)$$

En utilisant la fonction indicatrice, une fonction $f \in \mathcal{H}(V)$ peut être décomposée en ses ensembles de niveaux, notés f^l avec $f^l = \chi_{(f=l)}$.

Propriété 2.7.1. *Quelque soit le niveau f^l considéré, nous pouvons obtenir un ensemble de sommets $\mathcal{A}^l \subset V$ tel que les normes \mathcal{L}_q des gradients externe (2.65) et interne (2.66) en un sommet $u \in V$ soient définies pour $0 < q < +\infty$ par*

$$\|(\nabla_w^+ f^l)(u)\|_q = \left[\sum_{\substack{v \sim u, \\ u \in \partial^+ \mathcal{A}^l}} w_{uv}^{q/2} \right]^{1/q} \chi_{(\partial^+ \mathcal{A}^l)}(u) \quad (2.76)$$

$$\|(\nabla_w^- f^l)(u)\|_q = \left[\sum_{\substack{v \sim u, \\ u \in \partial^- \mathcal{A}^l}} w_{uv}^{q/2} \right]^{1/q} \chi_{(\partial^- \mathcal{A}^l)}(u) \quad (2.77)$$

Démonstration. En posant $f^l = \chi_{(\mathcal{A}^l)}$ et avec les définitions (2.65) et (2.66), nous obtenons

$$\|(\nabla_w^+ f^l)(u)\|_q = \|(\nabla_w^+ \chi_{(\mathcal{A}^l)})(u)\|_q \quad (2.78)$$

$$= \left[\sum_{v \sim u} w_{uv}^{q/2} |\max(0, \chi_{(\mathcal{A}^l)}(v) - \chi_{(\mathcal{A}^l)}(u))|^q \right]^{1/q} \quad (2.79)$$

$$\|(\nabla_w^- f^l)(u)\|_q = \|(\nabla_w^- \chi_{(\mathcal{A}^l)})(u)\|_q \quad (2.80)$$

$$= \left[\sum_{v \sim u} w_{uv}^{q/2} |\min(0, \chi_{(\mathcal{A}^l)}(v) - \chi_{(\mathcal{A}^l)}(u))|^q \right]^{1/q} . \quad (2.81)$$

En étudiant les différentes configurations possibles selon que u appartienne ou non à \mathcal{A}^l et de même avec son voisinage $\mathcal{N}(u)$, nous étudions l'influence de ces configurations sur la quantité $\chi_{(\mathcal{A}^l)}(v) - \chi_{(\mathcal{A}^l)}(u)$. Nous avons, pour l'équation (2.79),

$$\chi_{(\mathcal{A}^l)}(v) - \chi_{(\mathcal{A}^l)}(u) = \begin{cases} 0 & = \max(0, 0 - 1) & \text{si } u \in \mathcal{A}^l \text{ et } v \notin \mathcal{A}^l \\ 0 & = \max(0, 1 - 1) & \text{si } u \in \mathcal{A}^l \text{ et } v \in \mathcal{A}^l \\ 0 & = \max(0, 0 - 0) & \text{si } u \notin \mathcal{A}^l \text{ et } v \notin \mathcal{A}^l \\ 1 & = \max(0, 1 - 0) & \text{si } u \notin \mathcal{A}^l \text{ et } v \in \mathcal{A}^l \end{cases} . \quad (2.82)$$

Pour l'équation (2.81), nous obtenons

$$\chi_{(\mathcal{A}^l)}(v) - \chi_{(\mathcal{A}^l)}(u) = \begin{cases} -1 & = \min(0, 0 - 1) & \text{si } u \in \mathcal{A}^l \text{ et } v \notin \mathcal{A}^l \\ 0 & = \min(0, 1 - 1) & \text{si } u \in \mathcal{A}^l \text{ et } v \in \mathcal{A}^l \\ 0 & = \min(0, 0 - 0) & \text{si } u \notin \mathcal{A}^l \text{ et } v \notin \mathcal{A}^l \\ 0 & = \min(0, 1 - 0) & \text{si } u \notin \mathcal{A}^l \text{ et } v \in \mathcal{A}^l \end{cases} . \quad (2.83)$$

Les seules configurations possibles pour lesquelles les normes des gradients externe et interne sont non nulles (quand $\chi_{(\mathcal{A}^l)}(v) - \chi_{(\mathcal{A}^l)}(u) > 0$ et $\chi_{(\mathcal{A}^l)}(v) - \chi_{(\mathcal{A}^l)}(u) < 0$) sont les cas où $u \notin \mathcal{A}^l$ et $v \in \mathcal{A}^l$ pour le bord externe et $u \in \mathcal{A}^l$ et $v \notin \mathcal{A}^l$ pour le bord interne. Ces configurations correspondent aux définitions exactes des bords externe (2.73) $\partial^+ \mathcal{A}^l$ et interne (2.74) $\partial^- \mathcal{A}^l$ de l'ensemble \mathcal{A}^l . Finalement,

$$\|(\nabla_w^+ f^l)(u)\|_q = \left[\sum_{v \sim u} w_{uv}^{q/2} |\max(0, \chi_{(\mathcal{A}^l)}(v) - \chi_{(\mathcal{A}^l)}(u))|^q \right]^{1/q} \quad (2.84)$$

$$= \left[\sum_{\substack{v \sim u, \\ u \in \partial^+ \mathcal{A}^l}} w_{uv}^{q/2} \right]^{1/q} \chi_{(\partial^+ \mathcal{A}^l)}(u) \quad (2.85)$$

$$\|(\nabla_w^- f^l)(u)\|_q = \left[\sum_{v \sim u} w_{uv}^{q/2} |\min(0, \chi_{(\mathcal{A}^l)}(v) - \chi_{(\mathcal{A}^l)}(u))|^q \right]^{1/q} \quad (2.86)$$

$$= \left[\sum_{\substack{v \sim u, \\ u \in \partial^- \mathcal{A}^l}} w_{uv}^{q/2} \right]^{1/q} \chi_{(\partial^- \mathcal{A}^l)}(u) . \quad (2.87)$$

ce qui correspond aux équations (2.76) et (2.77). \square

Comme pour la propriété 2.7.1, nous obtenons la propriété suivante.

Propriété 2.7.2. *Quelque soit le niveau f^l considéré, nous pouvons obtenir un ensemble de sommets $\mathcal{A}^l \subset V$ telles que les normes \mathcal{L}_∞ des gradients externe (2.67) et interne (2.68) en un sommet $u \in V$ soient définies par*

$$\|(\nabla_w^+ f^l)(u)\|_\infty = \left(\max_{\substack{v \sim u, \\ u \in \partial^+ \mathcal{A}^l}} \sqrt{w_{uv}} \right) \chi_{(\partial^+ \mathcal{A}^l)}(u) \quad (2.88)$$

$$\|(\nabla_w^- f^l)(u)\|_\infty = \left(\max_{\substack{v \sim u, \\ u \in \partial^- \mathcal{A}^l}} \sqrt{w_{uv}} \right) \chi_{(\partial^- \mathcal{A}^l)}(u) \quad (2.89)$$

Propriété 2.7.3. *Quelque soit le niveau f^l considéré, la norme du gradient pondéré (2.29) peut s'exprimer en un sommet $u \in V$ à l'aide des gradients pondérés directionnels (2.63) et (2.64) par*

$$\|(\nabla_w f^l)(u)\|_q^q = \|(\nabla_w^+ f^l)(u)\|_q^q + \|(\nabla_w^- f^l)(u)\|_q^q \quad (2.90)$$

avec $0 < q < +\infty$.

Démonstration. En utilisant la propriété 2.7.1, les définitions des bords externe (2.73) et interne (2.74) et la définition de la norme \mathcal{L}_q , avec $0 < q < +\infty$, du gradient pondéré (2.30) nous avons en un sommet $u \in V$

$$\|(\nabla_w f^l)(u)\|_q^q \quad (2.91)$$

$$= \sum_{v \sim u} w_{uv}^{q/2} (\chi_{(\mathcal{A}^l)}(v) - \chi_{(\mathcal{A}^l)}(u))^q \quad (2.92)$$

$$= \sum_{\substack{v \sim u, \\ u \in \partial^+ \mathcal{A}^l}} w_{uv}^{q/2} (\chi_{(\mathcal{A}^l)}(v) - \chi_{(\mathcal{A}^l)}(u))^q + \sum_{\substack{v \sim u, \\ u \in \partial^- \mathcal{A}^l}} w_{uv}^{q/2} (\chi_{(\mathcal{A}^l)}(v) - \chi_{(\mathcal{A}^l)}(u))^q \quad (2.93)$$

$$= \|(\nabla_w^+ f^l)(u)\|_q^q + \|(\nabla_w^- f^l)(u)\|_q^q \quad (2.94)$$

ce qui correspond bien à la propriété 2.7.3. \square

De manière similaire à la propriété 2.7.3 et en utilisant la propriété 2.7.2, nous avons la propriété suivante.

Propriété 2.7.4. *Quelque soit le niveau f^l considéré, la norme \mathcal{L}_∞ du gradient pondéré (2.31) peut s'exprimer en un sommet $u \in V$ à l'aide des gradients directionnels (2.63) et (2.64) par*

$$\|(\nabla_w f^l)(u)\|_\infty = \|(\nabla_w^+ f^l)(u)\|_\infty + \|(\nabla_w^- f^l)(u)\|_\infty . \quad (2.95)$$

Propriété 2.7.5. *Selon la position d'un sommet $u \in V$ relativement aux bords externe $\partial^+ \mathcal{A}^l$ et interne $\partial^- \mathcal{A}^l$, la norme \mathcal{L}_q pour $0 < q < +\infty$ du gradient pondéré d'un niveau de la fonction f^l peut se simplifier de la manière suivante*

$$\|(\nabla_w f^l)(u)\|_q = \begin{cases} \|(\nabla_w^+ f^l)(u)\|_q & \text{si } u \in \partial^+ \mathcal{A}^l \\ \|(\nabla_w^- f^l)(u)\|_q & \text{si } u \in \partial^- \mathcal{A}^l \end{cases} . \quad (2.96)$$

Démonstration. De la propriété 2.7.3, nous avons

$$\|(\nabla_w f^l)(u)\|_q^q = \|(\nabla_w^+ f^l)(u)\|_q^q + \|(\nabla_w^- f^l)(u)\|_q^q. \quad (2.97)$$

Si $u \in \partial^+ \mathcal{A}^l$ alors $\|(\nabla_w^- f^l)(u)\|_q^q = 0$. De même, si $u \in \partial^- \mathcal{A}^l$ alors $\|(\nabla_w^+ f^l)(u)\|_q^q = 0$. \square

De la même manière, en utilisant la propriété 2.7.4, nous avons la propriété suivante.

Propriété 2.7.6. *Selon la position d'un sommet $u \in V$ relativement aux bords externe $\partial^+ \mathcal{A}^l$ et interne $\partial^- \mathcal{A}^l$, la norme \mathcal{L}_∞ du gradient pondéré d'un niveau de la fonction f^l peut se simplifier de la manière suivante*

$$\|(\nabla_w f^l)(u)\|_\infty = \begin{cases} \|(\nabla_w^+ f^l)(u)\|_\infty & \text{si } u \in \partial^+ \mathcal{A}^l \\ \|(\nabla_w^- f^l)(u)\|_\infty & \text{si } u \in \partial^- \mathcal{A}^l \end{cases} . \quad (2.98)$$

Propriété 2.7.7. *Pour une fonction $f \in \mathcal{H}(V)$, la norme \mathcal{L}_q , avec $0 < q < +\infty$, du gradient pondéré de f en un sommet $u \in V$ s'exprime en fonction des gradients pondérés directionnels*

$$\|(\nabla_w f)(u)\|_q^q = \|(\nabla_w^+ f)(u)\|_q^q + \|(\nabla_w^- f)(u)\|_q^q . \quad (2.99)$$

Démonstration. En étendant la propriété 2.7.3 à tous les niveaux de la fonction f nous obtenons la propriété 2.7.7.

Cette propriété peut également être obtenue en utilisant l'identité suivante

$$|a - b| = |\max(0, a - b)| + |\min(0, a - b)| . \quad (2.100)$$

En effet,

$$\|(\nabla_w f)(u)\|_q^q = \sum_{v \sim u} w_{uv}^{q/2} |f(v) - f(u)|^q \quad (2.101)$$

$$= \sum_{v \sim u} w_{uv}^{q/2} \left(|\max(0, f(v) - f(u))| + |\min(0, f(v) - f(u))| \right)^q \quad (2.102)$$

Nous pouvons remarquer que

$$|f(v) - f(u)| = \begin{cases} |\max(0, f(v) - f(u))| & \text{si } f(v) - f(u) > 0 \\ |\min(0, f(v) - f(u))| & \text{si } f(v) - f(u) < 0 \end{cases} , \quad (2.103)$$

alors

$$\sum_{v \sim u} w_{uv}^{q/2} \left(|\max(0, f(v) - f(u))| + |\min(0, f(v) - f(u))| \right)^q \quad (2.104)$$

$$= \sum_{v \sim u} w_{uv}^{q/2} \left(|\max(0, f(v) - f(u))|^q + |\min(0, f(v) - f(u))|^q \right) \quad (2.105)$$

$$= \sum_{v \sim u} w_{uv}^{q/2} |\max(0, f(v) - f(u))|^q + \sum_{v \sim u} w_{uv}^{q/2} |\min(0, f(v) - f(u))|^q \quad (2.106)$$

$$= \|(\nabla_w^+ f)(u)\|_q^q + \|(\nabla_w^- f)(u)\|_q^q . \quad (2.107)$$

□

Propriété 2.7.8. *Pour toute fonction $f \in \mathcal{H}(V)$, la norme \mathcal{L}_∞ du gradient pondéré de la fonction f en un sommet $u \in V$ se décompose selon la norme des gradients pondérés directionnels de la manière suivante*

$$\|(\nabla_w f)(u)\|_\infty = \max(\|(\nabla_w^+ f)(u)\|_\infty, \|(\nabla_w^- f)(u)\|_\infty) . \quad (2.108)$$

Démonstration. En utilisant l'identité suivante

$$|a - b| = |\max(0, a - b)| + |\min(0, a - b)|, \quad (2.109)$$

nous avons

$$\|(\nabla_w f)(u)\|_\infty \quad (2.110)$$

$$= \max_{v \sim u} \sqrt{w_{uv}} |f(v) - f(u)| \quad (2.111)$$

$$= \max_{v \sim u} \sqrt{w_{uv}} \left(|\max(0, f(v) - f(u))| + |\min(0, f(v) - f(u))| \right) \quad (2.112)$$

$$= \max_{v \sim u} \sqrt{w_{uv}} |\max(0, f(v) - f(u))| + \max_{v \sim u} \sqrt{w_{uv}} |\min(0, f(v) - f(u))| \quad (2.113)$$

$$= \|(\nabla_w^+ f)(u)\|_\infty + \|(\nabla_w^- f)(u)\|_\infty. \quad (2.114)$$

Si la quantité $(f(v) - f(u)) > 0$ alors $\|(\nabla_w^- f)(u)\|_\infty = 0$ et inversement, si $(f(v) - f(u)) < 0$ alors $\|(\nabla_w^+ f)(u)\|_\infty = 0$. Nous avons donc

$$\|(\nabla_w f)(u)\|_\infty = \|(\nabla_w^+ f)(u)\|_\infty + \|(\nabla_w^- f)(u)\|_\infty \quad (2.115)$$

$$= \max(\|(\nabla_w^+ f)(u)\|_\infty, \|(\nabla_w^- f)(u)\|_\infty) \quad (2.116)$$

□

2.8 Formulation générale du p -Laplacien

2.8.1 p -Laplacien anisotrope

À partir du p -Laplacien isotrope (2.34), une extension possible est le p -Laplacien pondéré anisotrope $\Delta_{w,p}^a : \mathcal{H}(V) \rightarrow \mathcal{H}(V)$ appliqué à une fonction $f \in \mathcal{H}(V)$ pour toutes valeurs réelles $0 < p < +\infty$. Il se définit par

$$\Delta_{w,p}^a f = \mathbf{d}_w^* (|\mathbf{d}_w f|^{p-2} \mathbf{d}_w f) \quad (2.117)$$

avec $|\cdot|$ désignant la valeur absolue.

Comme pour le p -Laplacien isotrope (2.34), nous pouvons déduire l'expression de l'opérateur de p -Laplace anisotrope en un sommet $u \in V$ à l'aide la définition de la différence pondérée et de son adjoint.

$$(\Delta_{w,p}^a f)(u) = \sum_{v \sim u} \sqrt{w_{uv}} \left(\frac{(\mathbf{d}_w f)(vu)}{|(\mathbf{d}_w f)(vu)|^{2-p}} - \frac{(\mathbf{d}_w f)(uv)}{|(\mathbf{d}_w f)(uv)|^{2-p}} \right) \quad (2.118)$$

$$= \sum_{v \sim u} w_{uv}^{p/2} \left(\frac{(f(u) - f(v))}{|f(u) - f(v)|^{2-p}} - \frac{(f(v) - f(u))}{|f(v) - f(u)|^{2-p}} \right) \quad (2.119)$$

$$= \sum_{v \sim u} w_{uv}^{p/2} (f(u) - f(v)) |f(u) - f(v)|^{p-2}. \quad (2.120)$$

Propriété 2.8.1. Lorsque le paramètre $p = 2$, les p -Laplacien isotrope et anisotrope ont la même expression.

$$(\Delta_{w,2}^a f)(u) = (\Delta_{w,2}^i f)(u) = \sum_{v \sim u} w_{uv} (f(u) - f(v)) \quad (2.121)$$

Remarque. Lorsque $p = 1$, l'opérateur de p -Laplace anisotrope peut se réécrire de la manière suivante

$$(\Delta_{w,1}^a f)(u) = \sum_{v \sim u} \sqrt{w_{uv}} \operatorname{sign}(f(u) - f(v)) \quad (2.122)$$

où sign désigne la fonction signe avec l'identité suivante $\operatorname{sign}(a) = a/|a|$.

2.8.2 p -Laplacien

Nous pouvons constater que les p -Laplaciens isotrope et anisotrope peuvent être obtenus à partir d'une formulation générale. Soit $\Delta_{w,p,q}^* : \mathcal{H}(V) \rightarrow \mathcal{H}(V)$ le p -Laplacien pondéré d'une fonction $f \in \mathcal{H}(V)$ pour tout réel $0 < p < +\infty$ et $0 < q < +\infty$. Il se définit comme

$$\Delta_{w,p,q}^* f = \mathbf{d}_w^* (\|\nabla_w f\|_q^{p-q} |\mathbf{d}_w f|^{q-2} \mathbf{d}_w f) \quad (2.123)$$

Avec les définitions de la différence et de l'adjoint, nous pouvons donner l'expression du p -Laplacien généralisé. En un sommet $u \in V$

$$(\Delta_{w,p,q}^* f)(u) \quad (2.124)$$

$$= \sum_{v \sim u} \sqrt{w_{uv}} (\|\nabla_w f\|_q^{p-q} |\mathbf{d}_w f\|_q^{p-q} (\mathbf{d}_w f)(vu) - \|\nabla_w f\|_q^{p-q} |\mathbf{d}_w f\|_q^{p-q} (\mathbf{d}_w f)(uv)) \quad (2.125)$$

$$- \|\nabla_w f\|_q^{p-q} |\mathbf{d}_w f\|_q^{p-q} (\mathbf{d}_w f)(uv) \quad (2.126)$$

$$= \sum_{v \sim u} w_{uv}^{q/2} (\|\nabla_w f\|_q^{p-q} (f(u) - f(v)) |f(u) - f(v)|^{p-2} - \|\nabla_w f\|_q^{p-q} (f(v) - f(u)) |f(v) - f(u)|^{p-2}) \quad (2.127)$$

$$- \|\nabla_w f\|_q^{p-q} (f(v) - f(u)) |f(v) - f(u)|^{p-2}) \quad (2.128)$$

$$= \sum_{v \sim u} w_{uv}^{q/2} (\|\nabla_w f\|_q^{p-q} + \|\nabla_w f\|_q^{p-q}) \cdot (f(u) - f(v)) |f(u) - f(v)|^{p-2} \quad (2.129)$$

$$\cdot (f(u) - f(v)) |f(u) - f(v)|^{p-2} \quad (2.130)$$

2.9 Propriétés et relations du p -Laplacien

Dans cette section, nous montrons certaines propriétés du p -Laplacien et les relations avec les gradients pondérés présentés dans les sections précédentes.

Propriété 2.9.1. Lorsque le paramètre $q = 2$, l'expression du p -Laplacien d'une fonction $f \in \mathcal{H}(V)$ équivaut à celle du p -Laplacien isotrope (2.34)

$$\Delta_{w,p,2}^* f = \mathbf{d}_w^* (\|\nabla_w f\|_2^{p-2} \mathbf{d}_w f) = \Delta_{w,p}^i \quad (2.131)$$

Propriété 2.9.2. Lorsque le paramètre $q = p$, l'expression du p -Laplacien d'une fonction $f \in \mathcal{H}(V)$ équivaut à celle du p -Laplacien anisotrope (2.117)

$$\Delta_{w,p,p}^* f = \mathbf{d}_w^* (|\mathbf{d}_w f|^{p-2} \mathbf{d}_w f) = \Delta_{w,p}^a \quad (2.132)$$

Propriété 2.9.3. Lorsque les paramètres $q = p = 2$, le p -Laplacien et les p -Laplaciens isotrope et anisotrope d'une fonction $f \in \mathcal{H}(V)$ ont la même expression et correspondent au Laplacien combinatoire (2.39)

$$\Delta_{w,2,2}^* f = \Delta_{w,2,2}^i f = \Delta_{w,2,2}^a f = \frac{1}{2} \Delta_w f \quad (2.133)$$

Démonstration. En un sommet $u \in V$

$$(\Delta_{w,2,2}^* f)(u) = \sum_{u \sim v} \sqrt{w_{uv}} ((d_w f)(vu) - (d_w f)(uv)) \quad (2.134)$$

$$= \sum_{u \sim v} \sqrt{w_{uv}} \left(\sqrt{w_{uv}} (f(u) - f(v)) - \sqrt{w_{uv}} (f(v) - f(u)) \right) \quad (2.135)$$

$$= 2 \sum_{u \sim v} w_{uv} (f(v) - f(u)) \quad (2.136)$$

$$= (\Delta_{w,2}^i f)(u) = (\Delta_{w,2}^a f)(u) = \frac{1}{2} (\Delta_w f)(u) . \quad (2.137)$$

□

Propriété 2.9.4. Lorsque les paramètres $q = 2$ et $p = 1$, le p -Laplacien d'une fonction $f \in \mathcal{H}(V)$ correspond à la définition de la courbure pondérée sur graphes (2.41).

$$\Delta_{w,2,1}^* f = \kappa_w f = \Delta_{w,1}^i f . \quad (2.138)$$

TAB. 2.1: Cas particuliers du p -Laplacien

p	q	Opérateurs
p	2	$\Delta_{w,p,2}^* = \Delta_{w,p}^i$ (p -Laplacien isotrope)
p	p	$\Delta_{w,p,2}^* = \Delta_{w,p}^a$ (p -Laplacien anisotrope)
2	2	$\Delta_{w,p,2}^* = \frac{1}{2} \Delta_w$ (Laplacien combinatoire)
1	2	$\Delta_{w,p,2}^* = \kappa_w$ (courbure)

Le tableau 2.1 résume les différentes relations entre le p -Laplacien et les autres opérateurs de Laplace pour différentes valeurs des paramètres p et q .

2.10 Conclusion

Dans ce chapitre, nous avons présenté les notations, notions et définitions nécessaires pour la compréhension de la suite de ce manuscrit. Nous avons rappelé des définitions sur la structure de graphe et les opérateurs différentiels qui lui sont associés.

Nous avons proposé de nouvelles familles d'opérateurs de premier et second ordre tels que les différences et gradients directionnelles et une classe de Laplacien telle que le Laplacien anisotrope. Les problèmes que nous aborderons dans la suite de ce manuscrit utilisent les opérateurs différentiels et la structure de graphe présentés dans ce chapitre.

Finalement, à partir des opérateurs différentiels discrets définis dans ce chapitre, nous pouvons envisager de reformuler à partir des équations aux différences partielles d'autres opérateurs définis dans le domaine continu, par exemple le Laplacien infini.

Fonctionnelles discrètes de régularisation

Sommaire

3.1	Fonctionnelles de régularisation	37
3.2	Modèles de régularisation, filtrage et simplification	47
3.3	Problèmes d'interpolation basés sur la régularisation	73
3.4	Optimisation globale pour la segmentation de données	97

Résumé

Dans ce chapitre, nous introduisons un cadre formel permettant d'exprimer des fonctionnelles de régularisation définies sur des graphes. Elles sont basées sur les opérateurs différentiels discrets du premier et second ordre présentés dans le chapitre 2.

Les approches que nous proposons dans ce chapitre considèrent la régularisation comme un problème variationnel qui consiste à minimiser une énergie. Plusieurs énergies discrètes sont étudiées et des algorithmes linéaires et non linéaires sont proposés afin d'approximer leurs solutions.

Nous présentons deux classes de modèles variationnels basées sur ces fonctionnelles. Elles nous permettent de considérer différentes problématiques liées à la régularisation :

- les problèmes de filtrage et de simplification de données discrètes. Nous appliquons ces traitements sur des données multi-variées de grande dimension et montrons qu'ils peuvent être utilisés comme étape de pré-traitement à des méthodes de regroupement, de classification ou d'apprentissage ;
- les problèmes d'interpolation basés sur la régularisation tels que la colorisation d'images en niveau de gris ou les méthodes semi supervisées de segmentation, de classification et d'apprentissage. Nous montrons également comment notre formalisme, basé sur les graphes, peut être utile et appliqué aux problématiques liées à la segmentation des images cellulaires issues de microscopie ;

- les problèmes d’optimisation globale de fonctionnelles non convexes. À partir d’une méthode de relaxation, nous proposons des modèles généraux discrets permettant de trouver les minima globaux de ces énergies non convexes. Nous appliquons ces modèles pour la segmentation d’images et le partitionnement de données arbitraires en utilisant, par exemple, les modèles de Rudin, Osher et Fatemi ou de Chan et Vese.

À travers différentes expérimentations, nous illustrons le potentiel de nos méthodes.

Mots clés

Modèles discrets de régularisation, graphes de topologie arbitraire, modèles variationnels, minimisation d’énergie, optimisation globale, méthodes semi supervisées, contours actifs globaux, filtrage, restauration, segmentation, traitements non locaux d’images, traitements de données de grande dimension, imagerie cellulaire.

Contenu du chapitre

- La section 3.1 définit nos familles de fonctionnelles discrètes de régularisation et présente certaines de leurs propriétés. Cette section introduit également nos classes de modèles variationnels définis sur des graphes.
- La section 3.2 présente des modèles de régularisation basés sur les fonctionnelles définies dans la section 3.1. En particulier, nous présentons deux modèles de régularisation : les modèles isotrope et anisotrope. Ces deux modèles sont appliqués au filtrage et à la simplification de données arbitraires. Nous présentons le comportement de nos modèles pour la simplification de variétés, le débruitage de collections d’images et de bases de données. Finalement, nous montrons que ces méthodes de filtrage peuvent être utilisées comme étape de pré-traitement afin d’améliorer les résultats de méthodes de classification de données.
- La section 3.3 présente des problèmes d’interpolation qui peuvent se formuler à partir de nos modèles de régularisation. Nous nous intéressons plus particulièrement aux problèmes de la colorisation des images monochromes et de la classification semi supervisée (en introduisant notamment des schémas non locaux pour le traitement des images). Dans cette section, nous appliquerons nos modèles (filtrage et classification) pour le traitement d’images cellulaires provenant de microscopie conventionnelle et d’un dispositif d’acquisition de lames entières.
- La section 3.4 présente une autre classe de modèles variationnels décrivant des problèmes d’optimisation globale. Nous proposons une résolution basée sur une méthode de relaxation permettant de trouver une solution à ces problèmes non convexes. Cette classe générale sera appliquée pour la segmentation de données à travers des modèles tels que celui de Rudin, Osher et Fatemi ou celui de Chan et Vese.

Publications associées à ce chapitre

Chapitre de livre

- [1] Vinh-Thong TA, Olivier LÉZORAY et Abderrahim ELMOATAZ : *Biomedical Microscopic Image Processing by Graphs*. Information Science Reference, 2009. À paraître.

Revue internationale avec comités de lecture

- [1] Vinh-Thong TA, Olivier LÉZORAY, Abderrahim ELMOATAZ et Sophie SCHÜPP : Graph-based tools for microscopic cellular image segmentation. *Pattern Recognition*, 42(6):1113–1125, 2009.

Conférences internationales avec comités de lecture et actes

- [1] Olivier LÉZORAY, Vinh-Thong TA et Abderrahim ELMOATAZ : Manifold and data filtering on graphs. Dans *Proceedings of International Symposium on Methodologies for Intelligent Systems, International Workshop on Topological learning*, pages 19 – 28, 2009.
- [2] Olivier LÉZORAY, Vinh-Thong TA et Abderrahim ELMOATAZ : Nonlocal graph regularization for image colorization. Dans *Proceedings of 19th International Conference on Pattern Recognition (ICPR)*, 2008.
- [3] Olivier LÉZORAY, Vinh-Thong TA et Abderrahim ELMOATAZ : Impulse noise removal by spectral clustering and regularization on graphs. Dans *Proceedings of 19th International Conference on Pattern Recognition (ICPR)*, 2008.
- [4] Abderrahim ELMOATAZ, Olivier LÉZORAY, Sébastien BOUGLEUX et Vinh-Thong TA : Unifying local and nonlocal processing with partial difference operators on weighted graphs. Dans *Proceedings of the International Workshop on Local and Non-Local Approximation in Image Processing (LNLA)*, pages 11 – 26, 2008. **Article invité.**
- [5] Vinh-Thong TA, Olivier LÉZORAY et Abderrahim ELMOATAZ : Graph based semi and unsupervised classification and segmentation of microscopic images. Dans *Proceedings of 7th International Symposium on Signal Processing and Information Technology (ISSPIT)*, pages 1160–1165, 2007.
- [6] Vinh-Thong TA, Olivier LÉZORAY et Abderrahim ELMOATAZ : Graphs regularization for data sets and images : Filtering and semi-supervised classification. Dans *PASCAL Workshop : Learning from and with graphs, 6th IAPR-TC-15 Workshop on Graph based Representations in Pattern Recognition*, 2007.

Rapports techniques et articles soumis

- [1] Vinh-Thong TA, Sébastien BOUGLEUX, Abderrahim ELMOATAZ et Olivier LÉZORAY : Nonlocal anisotropic discrete regularization for image, data filtering and clustering. Rapport technique, GREYC CNRS UMR 6072 - Université de Caen Basse-Normandie – ENSICAEN - hal-00187165, nov 2007.

3.1 Fonctionnelles de régularisation

3.1.1 Introduction

Dans cette section, nous présentons une famille de fonctionnelles de régularisation définies sur des graphes. Cette famille de fonctionnelles est basée sur les normes des gradients présentés dans le chapitre 2.

Bougleux et Elmoataz [Bougleux et Elmoataz, 2005, Elmoataz *et al.*, 2008a] ont proposé une famille de fonctionnelles isotropes de régularisation définies sur des graphes et basées sur la variation totale (Tv^1). Ces fonctionnelles peuvent être vues comme des versions discrètes et des extensions de celles définies dans le domaine continu et proposées par Osher ou Chan [Osher et Shen, 2000, Chan *et al.*, 2001].

Soit $f : V \rightarrow \mathbb{R}$ une fonction à valeurs réelles définie sur les sommets d'un graphe pondéré $G = (V, E, w)$. La famille de fonctionnelles isotropes s'exprime à l'aide de la norme \mathcal{L}_2 du gradient symétrique pondéré :

$$\mathcal{R}_{w,p}^i(f) = \sum_{u \in V} \|(\nabla_w f)(u)\|_2^p = \sum_{u \in V} \left[\sum_{v \sim u} w_{uv} (f(v) - f(u))^2 \right]^{p/2} \quad (3.1)$$

pour $0 < p < +\infty$.

À partir de ces idées, nous proposons deux nouvelles familles de fonctionnelles de régularisation définies sur des graphes. Elles sont basées sur les normes des gradients pondérés symétriques (∇_w) et directionnels (∇_w^+ et ∇_w^-). Dans la suite de cette section, nous présentons leurs définitions ainsi que les propriétés qui les relient à leurs versions définies dans le domaine continu et à la notion de périmètre d'un sous graphe.

3.1.2 Une première famille de fonctionnelles

Une première famille de fonctionnelles discrètes de régularisation définie sur des graphes s'exprime à l'aide de la norme \mathcal{L}_q des gradients pondérés symétriques et directionnels. Pour $0 < q < +\infty$, $0 < p < +\infty$ et $q = p$ et une fonction $f \in \mathcal{H}(V)$, nous définissons

$$\mathcal{R}_{w,p}(f) = \sum_{u \in V} \|(\nabla_w f)(u)\|_p^p = \sum_{u \in V} \sum_{v \sim u} w_{uv}^{p/2} |f(v) - f(u)|^p, \quad (3.2)$$

$$\mathcal{R}_{w,p}^+(f) = \sum_{u \in V} \|(\nabla_w^+ f)(u)\|_p^p = \sum_{u \in V} \sum_{v \sim u} w_{uv}^{p/2} |\max(0, f(v) - f(u))|^p, \quad (3.3)$$

$$\mathcal{R}_{w,p}^-(f) = \sum_{u \in V} \|(\nabla_w^- f)(u)\|_p^p = \sum_{u \in V} \sum_{v \sim u} w_{uv}^{p/2} |\min(0, f(v) - f(u))|^p. \quad (3.4)$$

Nous pouvons remarquer que la fonctionnelle de régularisation (3.2) correspond à la version discrète de la fonctionnelle de régularisation anisotrope définie dans le domaine continu.

Propriété 3.1.1. *Dans le cas où $p = 2$, la famille de fonctionnelles isotropes (3.1) et anisotropes (3.2) ont les mêmes expressions. Pour une fonction $f \in \mathcal{H}(V)$ et à partir des*

¹de l'anglais *Total Variation*.

équations (3.1) et (3.2), nous avons

$$\mathcal{R}_{w,2}(f) = \mathcal{R}_{w,2}^i(f) = \sum_{u \in V} \|(\nabla_w f)(u)\|_2^2 = \sum_{u \in V} \sum_{v \sim u} w_{uv} (f(v) - f(u))^2 . \quad (3.5)$$

3.1.3 Une seconde famille de fonctionnelles

Une seconde famille de fonctionnelles discrètes de régularisation définies sur graphes s'exprime à l'aide de la norme \mathcal{L}_∞ et l'opérateur max. Elle est également basée sur les gradients pondérés symétriques et directionnels. Pour une fonction $f \in \mathcal{H}(V)$, nous définissons

$$\mathcal{R}_{w,\infty}(f) = \max_{u \in V} \|(\nabla_w f)(u)\|_\infty = \max_{u \in V} \max_{v \sim u} \sqrt{w_{uv}} |f(v) - f(u)| , \quad (3.6)$$

$$\mathcal{R}_{w,\infty}^+(f) = \max_{u \in V} \|(\nabla_w^+ f)(u)\|_\infty = \max_{u \in V} \max_{v \sim u} \sqrt{w_{uv}} \max(0, f(v) - f(u)) , \quad (3.7)$$

$$\mathcal{R}_{w,\infty}^-(f) = \max_{u \in V} \|(\nabla_w^- f)(u)\|_\infty = \max_{u \in V} \max_{v \sim u} \sqrt{w_{uv}} \min(0, f(v) - f(u)) . \quad (3.8)$$

3.1.4 Formulation générale

Nous pouvons définir une formulation générale permettant de retrouver les deux familles précédentes de fonctionnelles de régularisation.

Soit $\mathcal{R}_{w,q,\phi}^*(f)$ une famille de fonctionnelles de régularisation, pour un graphe pondéré $G = (V, E, w)$ et une fonction $f \in \mathcal{H}(V)$, définie par

$$\mathcal{R}_{w,q,\phi}^*(f) = \mathcal{O}_V \left(\phi(\|\nabla_w^* f\|_q) \right) \quad (3.9)$$

où

- \mathcal{O}_V est un opérateur s'appliquant sur l'ensemble des sommets V du graphe ;
- ϕ est un noyau de régularisation permettant de pénaliser les variations de la fonction f ;
- $\nabla_w^* = \{\nabla_w, \nabla_w^+, \nabla_w^-\}$ désigne un des opérateurs gradients pondérés définis sur des graphes.

Afin de retrouver les fonctionnelles de régularisation présentées dans le début de cette section, nous utilisons les fonctions ϕ suivantes

$$\phi_1(s) = s^p \text{ avec } 0 < p < +\infty \quad (3.10)$$

$$\phi_2(s) = s . \quad (3.11)$$

Le tableau 3.1 récapitule et montre les différentes valeurs des paramètres de la formulation générale permettant de retrouver la première et la seconde famille de fonctionnelles.

3.1.5 Propriétés des deux familles de fonctionnelles

Dans cette section, nous présentons des relations et des propriétés des différentes fonctionnelles de régularisation proposées précédemment. En particulier, nous montrons les liens existant entre ces fonctionnelles et, d'une part, la notion des périmètres discrets d'un sous graphe et, d'autre part, leurs versions locales et non locales définies dans le domaine continu.

TAB. 3.1: Cas particuliers de la formulation générale des fonctionnelles de régularisation discrètes définies sur des graphes

\mathcal{O}_V	ϕ	∇_w^*	q	fonctionnelles de régularisation
Σ	ϕ_1	∇_w	2	$\mathcal{R}_{w,2,\phi_1}^* = \mathcal{R}_{w,p}^i$ (isotrope)
Σ	ϕ_1	∇_w	p	$\mathcal{R}_{w,p,\phi_1}^* = \mathcal{R}_{w,p}$ (anitrope)
Σ	ϕ_1	∇_w	$p = 2$	$\mathcal{R}_{w,2,\phi_1}^* = \mathcal{R}_{w,2}^i = \mathcal{R}_{w,2}$
Σ	ϕ_1	∇_w^+	p	$\mathcal{R}_{w,p,\phi_1}^* = \mathcal{R}_{w,p}^+$
Σ	ϕ_1	∇_w^-	p	$\mathcal{R}_{w,p,\phi_1}^* = \mathcal{R}_{w,p}^-$
max	ϕ_2	∇_w	∞	$\mathcal{R}_{w,\infty,\phi_2}^* = \mathcal{R}_{w,\infty}$
max	ϕ_2	∇_w^+	∞	$\mathcal{R}_{w,\infty,\phi_2}^* = \mathcal{R}_{w,\infty}^+$
max	ϕ_2	∇_w^-	∞	$\mathcal{R}_{w,\infty,\phi_2}^* = \mathcal{R}_{w,\infty}^-$

Relations entre les fonctionnelles de régularisation

Les propriétés suivantes montrent les relations entre les fonctionnelles basées sur le gradient symétrique (∇_w) et celles basées sur les gradients directionnels (∇_w^+ et ∇_w^-).

Propriété 3.1.2. *Pour toute fonction $f \in \mathcal{H}(V)$, nous avons la relation suivante entre les fonctionnelles de régularisation basées sur le gradient symétrique et celles basées sur les gradients directionnels :*

$$\mathcal{R}_{w,p}(f) = \mathcal{R}_{w,p}^+(f) + \mathcal{R}_{w,p}^-(f) . \quad (3.12)$$

Cette dernière propriété s'obtient en utilisant la propriété 2.7.7 page 28. En un sommet $u \in V$, nous avons

$$\|(\nabla_w f)(u)\|_p^p = \|(\nabla_w^+ f)(u)\|_p^p + \|(\nabla_w^- f)(u)\|_p^p . \quad (3.13)$$

Avec cette dernière relation, nous obtenons

$$\mathcal{R}_{w,p}(f) = \sum_{u \in V} \|(\nabla_w f)(u)\|_p^p \quad (3.14)$$

$$= \sum_{u \in V} \|(\nabla_w^+ f)(u)\|_p^p + \sum_{u \in V} \|(\nabla_w^- f)(u)\|_p^p \quad (3.15)$$

$$= \mathcal{R}_{w,p}^+(f) + \mathcal{R}_{w,p}^-(f) . \quad (3.16)$$

Propriété 3.1.3. *Pour toute fonction $f \in \mathcal{H}(V)$, nous avons la relation suivante entre les fonctionnelles de régularisation basées sur les normes infini du gradient symétrique et celles basées sur les gradients directionnels :*

$$\mathcal{R}_{w,\infty}(f) = \max(\mathcal{R}_{w,\infty}^+(f), \mathcal{R}_{w,\infty}^-(f)) . \quad (3.17)$$

Cette relation s'obtient en utilisant la propriété 2.7.8 page 28. En un sommet $u \in V$, nous avons

$$\|(\nabla_w f)(u)\|_\infty = \max(\|(\nabla_w^+ f)(u)\|_\infty, \|(\nabla_w^- f)(u)\|_\infty) . \quad (3.18)$$

En utilisant cette relation, nous obtenons

$$\max(\mathcal{R}_{w,\infty}^+(f), \mathcal{R}_{w,\infty}^-(f)) \quad (3.19)$$

$$= \max\left(\max_{u \in V} \|(\nabla_w^+ f)(u)\|_\infty, \max_{u \in V} \|(\nabla_w^- f)(u)\|_\infty\right) \quad (3.20)$$

$$= \max\left(\max_{u \in V} (\|(\nabla_w^+ f)(u)\|_\infty, \|(\nabla_w^- f)(u)\|_\infty)\right) \quad (3.21)$$

$$= \max_{u \in V} \left(\max(\|(\nabla_w^+ f)(u)\|_\infty, \|(\nabla_w^- f)(u)\|_\infty)\right) \quad (3.22)$$

$$= \max_{u \in V} \|(\nabla_w f)(u)\|_\infty \quad (3.23)$$

$$= \mathcal{R}_{w,\infty}(f) . \quad (3.24)$$

Relations avec les périmètres discrets d'un graphe

Les propriétés suivantes montrent les relations entre les fonctionnelles de régularisation basées sur la norme \mathcal{L}_q , avec $0 < q < \infty$, et les périmètres discrets d'un sous graphe. Pour cela, nous utilisons d'une part la fonction indicatrice $\chi_{(S)} : V \rightarrow \{0, 1\}$ telle que pour un sommet $u \in V$

$$\chi_{(S)}(u) = \begin{cases} 0 & \text{si } u \notin S \\ 1 & \text{si } u \in S \end{cases} , \quad (3.25)$$

et d'autre part les définitions des bords d'un graphe $G = (V, E, w)$.

Rappelons les définitions des bords interne et externe d'un sous graphe. Soit $\mathcal{A} \subseteq V$ un sous graphe de G . Les définitions des bords externe $\partial^+ \mathcal{A}$ et interne $\partial^- \mathcal{A}$ de l'ensemble \mathcal{A} sont respectivement

$$\partial^+ \mathcal{A} = \{u \notin \mathcal{A} : \exists v \in \mathcal{A} \text{ avec une arête } uv \in E\} \quad (3.26)$$

$$\partial^- \mathcal{A} = \{u \in \mathcal{A} : \exists v \notin \mathcal{A} \text{ avec une arête } uv \in E\} . \quad (3.27)$$

À partir de ces définitions, nous obtenons la définition du bord de l'ensemble \mathcal{A} , noté $\partial \mathcal{A}$

$$\partial \mathcal{A} = \{uv \in E : \text{avec } u \text{ et } v \in \partial^+ \mathcal{A} \text{ ou } \partial^- \mathcal{A}\} . \quad (3.28)$$

Propriété 3.1.4. *La fonctionnelle de régularisation $\mathcal{R}_{w,p}^+$ basée sur le gradient directionnel externe de la fonction indicatrice χ correspond au périmètre discret du bord d'un sous graphe $\mathcal{A} \subseteq V$.*

$$\mathcal{R}_{w,p}^+(\chi_{(\partial \mathcal{A})}) = \frac{1}{2} \text{vol}(\partial \mathcal{A}) . \quad (3.29)$$

Démonstration. En posant $f = \chi_{(\mathcal{A})}$ et en utilisant la propriété 2.7.1 page 25 appliquée sur l'ensemble \mathcal{A} , nous avons

$$\mathcal{R}_{w,p}^+(\chi_{(\mathcal{A})}) = \sum_{u \in V} \|(\nabla_w^+ \chi_{(\mathcal{A})})(u)\|_p^p = \sum_{u \in V} \sum_{\substack{v \sim u, \\ u \in \partial^+ \mathcal{A}}} w_{uv}^{p/2} \chi_{(\partial^+ \mathcal{A})}(u) \quad (3.30)$$

$$= \sum_{u \in \partial^+ \mathcal{A}} \sum_{v \sim u} w_{uv}^{p/2} = \sum_{u \in \partial^+ \mathcal{A}} \sum_{v \in \partial^- \mathcal{A}} w_{uv}^{p/2} \quad (3.31)$$

$$= \sum_{u \in \partial^+ \mathcal{A}} \sum_{v \in \partial^- \mathcal{A}} w'_{uv} = \sum_{u \in \partial^+ \mathcal{A}} \text{deg}_{\partial^- \mathcal{A}}(u) = \frac{1}{2} \text{vol}(\partial \mathcal{A}) . \quad (3.32)$$

avec $w'_{uv} = w_{uv}^{p/2}$. □

De manière duale, nous avons la propriété suivante.

Propriété 3.1.5. *La fonctionnelle de régularisation $\mathcal{R}_{w,p}^-$ basée sur le gradient directionnel externe de la fonction indicatrice χ correspond au périmètre discret du bord d'un sous graphe $\mathcal{A} \subseteq V$.*

$$\mathcal{R}_{w,p}^-(\chi_{(\mathcal{A})}) = \frac{1}{2} \text{vol}(\partial\mathcal{A}) . \quad (3.33)$$

De manière similaire à la propriété 3.1.4, nous avons, avec $w'_{uv} = w_{uv}^{p/2}$

$$\mathcal{R}_{w,p}^-(\chi_{(\mathcal{A})}) = \sum_{u \in V} \|(\nabla_w^- \chi_{(\mathcal{A})})(u)\|_p^p = \sum_{u \in \partial^- \mathcal{A}} \sum_{v \in \partial^+ \mathcal{A}} w'_{uv} = \frac{1}{2} \text{vol}(\partial\mathcal{A}) . \quad (3.34)$$

À partir de ces deux dernières propriétés, nous en déduisons immédiatement la propriété suivante.

Propriété 3.1.6. *La fonctionnelle de régularisation $\mathcal{R}_{w,p}$ basée sur le gradient symétrique de la fonction indicatrice χ s'exprime à partir des gradients directionnels et correspond au périmètre discret du bord d'un sous graphe $\mathcal{A} \subseteq V$.*

$$\mathcal{R}_{w,p}(\chi_{(\mathcal{A})}) = \mathcal{R}_{w,p}^+(\chi_{(\mathcal{A})}) + \mathcal{R}_{w,p}^-(\chi_{(\mathcal{A})}) = \text{vol}(\partial\mathcal{A}) . \quad (3.35)$$

Démonstration. Cette propriété s'obtient à partir des propriétés précédentes (3.1.4 et 3.1.5) et de la propriété 2.7.7 page 28 appliquée à l'ensemble \mathcal{A} .

$$\|(\nabla_w \chi_{(\mathcal{A})})(u)\|_p^p = \|(\nabla_w^+ \chi_{(\mathcal{A})})(u)\|_p^p + \|(\nabla_w^- \chi_{(\mathcal{A})})(u)\|_p^p . \quad (3.36)$$

À partir de cette relation, nous obtenons

$$\mathcal{R}_{w,p}(\chi_{(\mathcal{A})}) \quad (3.37)$$

$$= \sum_{u \in V} \|(\nabla_w \chi_{(\mathcal{A})})(u)\|_p^p = \sum_{u \in V} \|(\nabla_w^+ \chi_{(\mathcal{A})})(u)\|_p^p + \sum_{u \in V} \|(\nabla_w^- \chi_{(\mathcal{A})})(u)\|_p^p \quad (3.38)$$

$$= \mathcal{R}_{w,p}^+(\chi_{(\mathcal{A})}) + \mathcal{R}_{w,p}^-(\chi_{(\mathcal{A})}) = \frac{1}{2} \text{vol}(\partial\mathcal{A}) + \frac{1}{2} \text{vol}(\partial\mathcal{A}) = \text{vol}(\partial\mathcal{A}) . \quad (3.39)$$

□

Remarque. *La propriété 3.1.6 peut également s'écrire de la manière suivante*

$$\mathcal{R}_{w,p}(\chi_{(\mathcal{A})}) = 2\mathcal{R}_{w,p}^+(\chi_{(\mathcal{A})}) = 2\mathcal{R}_{w,p}^-(\chi_{(\mathcal{A})}) = \text{vol}(\partial\mathcal{A}) . \quad (3.40)$$

Finalement, les fonctionnelles basées sur les gradients symétriques $\mathcal{R}_{w,p}$ et directionnels $\mathcal{R}_{w,p}^+$ et $\mathcal{R}_{w,p}^-$ de la fonction indicatrice appliquée à $\mathcal{A} \subseteq V$ peuvent être considérées comme étant les versions discrètes pondérées du périmètre d'un sous graphe [Boykov et Kolmogorov, 2003, Darbon, 2007]. Elle peuvent se réécrire sous la forme suivante

$$\mathcal{R}_{w,p}(\chi_{(\mathcal{A})}) = \text{Per}_{w,p}(\mathcal{A}); \quad \mathcal{R}_{w,p}^+(\chi_{(\mathcal{A})}) = \text{Per}_{w,p}^+(\mathcal{A}); \quad \mathcal{R}_{w,p}^-(\chi_{(\mathcal{A})}) = \text{Per}_{w,p}^-(\mathcal{A}) . \quad (3.41)$$

Ces derniers périmètres sont paramétrés par la fonction de poids w et le paramètre p , avec $0 < p < +\infty$.

Relations avec les fonctionnelles continues locales et non locales

Nous montrons ici que dans le domaine du traitement des images, nos familles de fonctionnelles permettent d'exprimer dans une même formulation les configurations locales ou non locales. De plus, nous montrons que nos fonctionnelles peuvent être vues comme étant les versions discrètes des fonctionnelles de régularisation locales et non locales définies dans le domaine continu et proposées par Gilboa et Osher [Gilboa et Osher, 2007a, Gilboa et Osher, 2007b].

À partir de la définition de la fonction de poids associés aux arêtes (lorsque deux sommets de V ne sont pas connectés $w_{uv} = 0$), nous obtenons la propriété suivante.

Propriété 3.1.7. *Soit $f : V \rightarrow \mathbb{R}$ et un graphe complet $G_\infty = (V, E, w)$. Toutes nos fonctionnelles de régularisation prennent en compte les interactions non locales entre les sommets V du graphe G_∞ .*

Pour simplifier, prenons par exemple les fonctionnelles $\mathcal{R}_{w,p}$ (3.2) et $\mathcal{R}_{w,\infty}$ (3.6) (la propriété présentée ici reste néanmoins valable pour l'ensemble des fonctionnelles proposées). Avec le graphe G_∞ , nous obtenons

$$\mathcal{R}_{w,p}(f) = \sum_{u \in V} \|(\nabla_w f)(u)\|_p^p = \sum_{u \in V} \sum_{v \in V} w_{uv}^{p/2} |f(v) - f(u)|^p, \quad (3.42)$$

$$\mathcal{R}_{w,\infty}(f) = \max_{u \in V} \|(\nabla_w f)(u)\|_\infty = \max_{u \in V} \max_{v \in V} \sqrt{w_{uv}} |f(v) - f(u)|. \quad (3.43)$$

Comme le montre cette propriété, les formulations des fonctionnelles de régularisation sont identiques à leurs expressions initiales ((3.2) et (3.6)). L'opérateur de sommation $\sum_{v \sim u}$ a été remplacé par l'expression $\sum_{v \in V}$. Cette propriété montre bien que les schémas locaux et non locaux (dans le contexte du traitement des images) sont naturellement intégrés dans nos formulations. En effet, nous pouvons constater que les notions de localité et de non localité disparaissent, dans le cas des graphes, au profit de la notion de voisinage des sommets.

Elmoataz *et al.* [Elmoataz *et al.*, 2008a] ont montré que la fonctionnelle de régularisation isotrope

$$\mathcal{R}_{w,p}^i(f) = \sum_{u \in V} \|(\nabla_w f)(u)\|_2^p \quad (3.44)$$

était l'extension discrète pondérée des fonctionnelles basées sur la variation totale isotrope :

$$\mathcal{J}_p^i(f) = \int_{\Omega} \|\nabla f\|_2^p dx \quad (3.45)$$

pour une fonction $f : \Omega \subset \mathbb{R}^m \rightarrow \mathbb{R}$, avec Ω un domaine borné de \mathbb{R}^m .

De la même manière, nous pouvons remarquer que la fonctionnelle de régularisation

$$\mathcal{R}_{w,p}(f) = \sum_{u \in V} \|(\nabla_w f)(u)\|_p^p \quad (3.46)$$

peut être considérée comme une extension discrète et pondérée de celle définie dans le domaine continu

$$\mathcal{J}_p(f) = \int_{\Omega} \|\nabla f\|_p^p dx. \quad (3.47)$$

En utilisant la propriété 3.1.7 page 42, cette fonctionnelle de régularisation discrète se réécrit

$$\mathcal{R}_{w,p}(f) = \sum_{u \in V} \sum_{v \in V} w_{uv}^{p/2} |f(v) - f(u)|^p \quad (3.48)$$

et sa version continue est donnée par

$$\mathcal{J}_{w,p}(f) = \int_{\Omega \times \Omega} w_{xy}^{p/2} |f(y) - f(x)|^p \, dy \, dx . \quad (3.49)$$

Dans le cas particulier où $p = 1$, l'équation (3.49) se réécrit

$$\mathcal{J}_{w,1}(f) = \int_{\Omega \times \Omega} \sqrt{w_{xy}} |f(y) - f(x)| \, dy \, dx \quad (3.50)$$

et correspond à la fonctionnelle non locale anisotrope proposée par Gilboa et Osher [Gilboa et Osher, 2007b]. Lorsque le paramètre $p = 2$, (3.49) se réécrit

$$\mathcal{J}_{w,2}(f) = \int_{\Omega \times \Omega} w_{xy} (f(y) - f(x))^2 \, dy \, dx . \quad (3.51)$$

Cette dernière équation correspond aux fonctionnelles isotropes proposées dans le domaine continu dans le contexte du traitement non local des images [Kindermann *et al.*, 2005, Gilboa et Osher, 2007a, Gilboa et Osher, 2007b]. Cette fonctionnelle est une interprétation variationnelle d'une famille de filtres de voisinages (le filtre à moyennes non locales [Buades *et al.*, 2008]).

Finalement, nous pouvons remarquer que lorsque le paramètre $p = 2$, les fonctionnelles de régularisation isotrope et anisotrope ont les mêmes expressions aussi bien dans le cas discret défini sur des graphes que dans le domaine continu.

Remarque. *Nos fonctionnelles de régularisation ont l'avantage d'être définies sur des graphes. Ce point permet de traiter par régularisation n'importe quelle fonction définie sur les sommets d'un graphe même si les domaines considérés sont irréguliers. De plus, notre approche étend la notion de régularité avec le paramètre p .*

3.1.6 Modèles variationnels discrets

Dans la suite de ce chapitre, nous allons utiliser ces familles de fonctionnelles discrètes de régularisation définies sur des graphes pour définir des modèles variationnels discrets.

Dans le contexte du traitement des images, les modèles variationnels fournissent un cadre formel bien adapté pour résoudre une large variété de problèmes liés à la vision par ordinateur tels que le filtrage, la restauration, l'interpolation ou encore la segmentation.

La solution de ces modèles peut être obtenue en minimisant une énergie adaptée au problème à résoudre. Cette minimisation est couramment formulée en terme d'équations aux dérivées partielles définies dans le domaine continu. Ces équations aux dérivées partielles sont alors discrétisées dans le domaine de l'image pour obtenir une solution numérique au problème de minimisation. Pour avoir un panorama complet sur ces méthodes, le lecteur intéressé peut se référer à [Alvarez *et al.*,

1993, Chan et Shen, 2005, Tsai et Osher, 2005, Aubert et Kornprobst, 2006] et aux références associées.

Inspirés par ces modèles définis dans le domaine continu, nous présentons des modèles variationnels discrets basés sur nos fonctionnelles de régularisation. Nos modèles sont généraux et permettent de régulariser n'importe quelle fonction définie sur les sommets d'un graphe. Ce point permet de traiter par la régularisation n'importe quelles données arbitraires (définies sur un domaine régulier ou irrégulier et de dimension quelconque).

D'une manière générale, soit $f^0 \in \mathcal{H}(V)$ une fonction définie sur les sommets d'un graphe pondéré $G = (V, E, w)$. Dans un contexte donné, la fonction f^0 peut être vue comme une observation d'une fonction dégradée $h \in \mathcal{H}(V)$ par un bruit additif $\eta \in \mathcal{H}(V)$ telle que

$$f^0 = h + \eta . \quad (3.52)$$

Le bruit η est considéré comme étant de moyenne nulle et de variance σ^2 ce qui correspond aux erreurs d'observations résultant du dispositif ayant permis d'observer f^0 .

Ce problème, qui est un problème mal posé [Bakushinsky et Goncharsky, 1994], consiste alors à retrouver la fonction non dégradée h à partir de la fonction f^0 en supprimant le bruit η . Pour retrouver la fonction h , une méthode usuelle est de chercher une fonction $f \in \mathcal{H}(V)$ qui soit assez régulière par rapport au graphe G mais également proche de la fonction initiale f^0 . Ce problème inverse peut être formulé en terme de minimisation d'énergie. Pour des raisons de clarté dans les notations, mais sans perte de généralisation, nous considérons des fonctions scalaires à valeurs réelles telles que $f^0 : V \rightarrow \mathbb{R}$ et $f : V \rightarrow \mathbb{R}$.

Initialement formulé par Tikhonov [Tikhonov et Arsenin, 1977], le problème de minimisation d'énergie peut prendre la forme générale suivante,

$$h \approx \min_{f:V \rightarrow \mathbb{R}} \left\{ \mathcal{E}(f, f^0, \lambda) = \mathcal{E}_R(f) + \lambda \mathcal{E}_D(f, f^0) \right\} \quad (3.53)$$

où l'énergie $\mathcal{E}(f, f^0, \lambda)$ est la somme des deux énergies $\mathcal{E}_R(f)$ et $\mathcal{E}_D(f, f^0)$. L'énergie $\mathcal{E}_R(f)$ est un terme de régularisation qui mesure la régularité de la fonction régularisée f . L'énergie $\mathcal{E}_D(f)$ est un terme d'attache aux données qui mesure la proximité de la fonction régularisée f par rapport à la fonction initiale f^0 . Cette énergie est pondérée par le paramètre $\lambda \in \mathbb{R}$ avec $\lambda \geq 0$ qui contrôle la proportion entre l'énergie de régularisation et cette d'attache aux données. Ce paramètre est également appelé multiplicateur de Lagrange [Aubert et Kornprobst, 2006]. Habituellement choisi comme étant constant, le paramètre λ peut néanmoins être adaptatif [Kang et Katsaggelos, 1995] et déterminé de manière automatique [Lézoray *et al.*, 2007, Chan *et al.*, 2001].

Finalement, nous pouvons également citer des travaux récents, basés sur des modèles variationnels et des énergies discrètes définis sur des graphes dans le contexte du traitement des images. Ces méthodes utilisent des opérateurs et des algorithmes d'optimisation combinatoire afin de solutionner les modèles. Le lecteur intéressé pourra se référer aux articles suivants [Grady, 2006, Couprie *et al.*, 2009, Grady et Alvino, 2009] et aux références associées.

Deux classes de modèles variationnels

Nous proposons ici deux classes de modèles variationnels définis sur des graphes.

- La première classe permet de considérer, d’une part, le filtrage et la simplification de données arbitraires et, d’autre part, des problèmes d’interpolation basés sur la régularisation tels que la colorisation d’images monochromes ou les méthodes semi supervisées de segmentation, de classification ou d’apprentissage.
- Les problèmes d’optimisation globale de fonctionnelles non convexes. À partir d’une méthode de relaxation, nous proposons des modèles généraux discrets permettant de trouver les minima globaux de ces énergies non convexes. Nous appliquons cette seconde classe de modèles pour la segmentation non locale d’images et le regroupement de données discrètes.

La première classe de modèles est formulée de la manière suivante :

$$\min_{f:V \rightarrow \mathbb{R}} \left\{ \mathcal{E}_1(f, f^0, \lambda) = \mathcal{R}_{w,q,\phi}^*(f) + \sum_{u \in V} \frac{\lambda}{2} \|f(u) - f^0(u)\|_2^2 \right\} \quad (3.54)$$

où le terme de régularisation utilise l’opérateur de sommation et le gradient symétrique ∇_w tel que

$$\mathcal{R}_{w,q,\phi}^*(f) = \sum_{u \in V} \phi(\|(\nabla_w f)(u)\|_q) . \quad (3.55)$$

La seconde classe de modèles variationnels considérée s’exprime par

$$\min_{f:V \rightarrow \{0,1\}} \left\{ \mathcal{E}_2(f, f^0, \lambda) = \mathcal{R}_{w,q,\phi}^*(f) + \sum_{u \in V} \lambda g(f^0)(u) f(u) \right\} \quad (3.56)$$

où le terme de régularisation prendra les deux formes suivantes : avec $\phi(s) = s^p$ et $q = p$

$$\mathcal{R}_{w,p,\phi}^*(f) = \sum_{u \in V} \|(\nabla_w^* f)(u)\|_p^p , \quad (3.57)$$

puis avec $\phi(s) = s$ et $q = \infty$

$$\mathcal{R}_{w,\infty,\phi}^*(f) = \max_{u \in V} \|(\nabla_w^* f)(u)\|_\infty . \quad (3.58)$$

Dans cette seconde famille de modèle, la fonction recherchée est une fonction binaire définie dans $\{0, 1\}$.

Dans la suite, nous étudierons ces deux classes de modèles et proposerons des algorithmes permettant d’approximer les solutions pour des valeurs particulières du paramètre q et de la fonction ϕ . L’ensemble des modèles que nous allons décrire dans la suite peuvent être considérés comme étant des versions discrètes et des extensions sur graphes de celles définies dans le domaine continu.

3.1.7 Conclusion

Dans cette section, nous avons présenté deux familles de fonctionnelles discrètes de régularisation définies sur des graphes. Nous avons montré les relations entre

ces différentes fonctionnelles et en particulier les liens existants avec leurs versions définies dans le domaine continu et avec la notion de périmètre d'un sous graphe.

Ces familles de fonctionnelles nous permettent de considérer deux classes de modèles variationnels liées à la régularisation. Ces modèles sont étudiés dans les sections suivantes de ce chapitre.

3.2 Modèles de régularisation, filtrage et simplification

3.2.1 Introduction

Dans cette section, nous étudions et proposons des solutions pour la première famille de modèles de régularisation définie sur un graphe pondéré $G = (V, E, w)$:

$$\min_{f:V \rightarrow \mathbb{R}} \left\{ \mathcal{E}_1(f, f^0, \lambda) = \mathcal{R}_{w,q,\phi}^*(f) + \sum_{u \in V} \frac{\lambda}{2} \|f(u) - f^0(u)\|_2^2 \right\} . \quad (3.59)$$

où le terme de régularisation $\mathcal{R}_{w,q,\phi}^*$ utilise l'opérateur de sommation et le gradient symétrique ∇_w tel que

$$\mathcal{R}_{w,q,\phi}^*(f) = \sum_{u \in V} \phi(\|(\nabla_w f)(u)\|_q) . \quad (3.60)$$

Cette fonctionnelle est basée sur la norme \mathcal{L}_q du gradient de la fonction $f \in \mathcal{H}(V)$ et fait intervenir la fonction $\phi(\cdot)$ (noyau de régularisation) permettant de pénaliser les larges variations de la fonction f dans le voisinage de chaque sommet.

À partir des solutions de cette première famille, nous présentons, dans cette section, deux modèles faisant intervenir des noyaux de régularisation particuliers et permettant d'une part, de retrouver un modèle de régularisation discrète isotrope [Elmoataz *et al.*, 2008a] et d'autre part, de proposer un modèle discret de régularisation anisotrope.

Remarque. *Différentes expressions du noyau $\phi(\cdot)$ ont été proposées dans la littérature. Nous pouvons par exemple citer les cas suivants.*

Nous obtenons la fonctionnelle de régularisation de Tikhonov [Tikhonov et Arsenin, 1977]

$$\sum_{u \in V} \|(\nabla_w f)(u)\|_q^2 \quad \text{avec } \phi(s) = s^2 . \quad (3.61)$$

Nous avons la fonctionnelle de régularisation basée sur la variation totale

$$\sum_{u \in V} \|(\nabla_w f)(u)\|_q \quad \text{avec } \phi(s) = s , \quad (3.62)$$

et sa version régularisée [Rudin et al., 1992, Osher et Shen, 2000]

$$\sum_{u \in V} \sqrt{\|(\nabla_w f)(u)\|_q^2 + \epsilon^2} \quad \text{avec } \phi(s) = \sqrt{s^2 + \epsilon^2} , \quad (3.63)$$

où le paramètre $\epsilon > 0$ et $\epsilon \rightarrow 0$. Finalement, nous pouvons également obtenir la fonctionnelle intervenant dans la diffusion non linéaire anisotrope proposée par Perona et Malik [Perona et Malik, 1990]

$$\sum_{u \in V} \frac{1}{2\|(\nabla_w f)(u)\|_q} \ln \left(1 + \frac{\|(\nabla_w f)(u)\|_q^2}{K^2} \right) \quad \text{avec } \phi(s) = \frac{1}{2s} \ln \left(1 + \frac{s^2}{K^2} \right) , \quad (3.64)$$

où K est une constante.

Les versions définies dans le domaine continu de cette famille de modèles sont bien connus dans le domaine du filtrage, de la simplification et de la restauration des images (voir par exemple [Alvarez *et al.*, 1993, Chan et Shen, 2005, Tsai et Osher, 2005, Aubert et Kornprobst, 2006]).

Un des avantages de nos modèles de régularisation est leur définition sur des graphes. Cela permet de traiter par régularisation des images, mais également n'importe quel type de données définies sur des domaines uniformes ou non.

À travers différentes expérimentations, nous montrerons dans cette section, le comportement des modèles isotrope et anisotrope pour le filtrage et la simplification de données arbitraires (bases de données et d'images, nuages de points). Finalement, nous montrerons également que ces nouvelles approches de traitement de données peuvent être utilisées en tant qu'étapes de pré-traitement pour des méthodes de regroupement, de classification ou d'apprentissage.

3.2.2 Solutions de la première classe de modèles variationnels

En supposant qu'il existe une solution globale et unique du problème (3.59), nous pouvons l'obtenir en calculant le passage par zéro de la dérivée partielle de l'énergie $\mathcal{E}_1(f, f^0, \lambda)$ par rapport à la fonction f en un sommet $u \in V$.

$$\frac{\partial \mathcal{E}_1(f, f^0, \lambda)}{\partial f(u)} = 0 . \quad (3.65)$$

Cette dernière équation peut se réécrire sous la forme suivante. Pour tout sommet $u \in V$

$$\frac{\partial}{\partial f(u)} \left(\sum_{u \in V} \phi(\|(\nabla_w f)(u)\|_q) + \frac{\lambda}{2} \|f(u) - f^0(u)\|_2^2 \right) = \quad (3.66)$$

$$\frac{\partial}{\partial f(u)} \left(\sum_{u \in V} \phi(\|(\nabla_w f)(u)\|_q) \right) + \lambda(f(u) - f^0(u)) = 0 . \quad (3.67)$$

Propriété 3.2.1. Pour un sommet $u \in V$ donné et toute fonction ϕ , la dérivée partielle du terme de régularisation (3.60) d'une fonction $f \in \mathcal{H}(V)$ a pour expression

$$\frac{\partial}{\partial f(u)} \left(\sum_{u \in V} \phi(\|(\nabla_w f)(u)\|_q) \right) = \quad (3.68)$$

$$\sum_{v \sim u} w_{uv}^{q/2} \left(\frac{\phi'(\|(\nabla_w f)(u)\|_q)}{\|(\nabla_w f)(u)\|_q^{q-1}} + \frac{\phi'(\|(\nabla_w f)(v)\|_q)}{\|(\nabla_w f)(v)\|_q^{q-1}} \right) \quad (3.69)$$

$$\cdot (f(u) - f(v)) |f(v) - f(u)|^{q-2} \quad (3.70)$$

où ϕ' correspond à la dérivée de la fonction ϕ .

En posant

$$\alpha_{uv}^{q,\phi,f} = w_{uv}^{q/2} \left(\frac{\phi'(\|(\nabla_w f)(u)\|_q)}{\|(\nabla_w f)(u)\|_q^{q-1}} + \frac{\phi'(\|(\nabla_w f)(v)\|_q)}{\|(\nabla_w f)(v)\|_q^{q-1}} \right) , \quad (3.71)$$

l'équation précédente se réécrit

$$\frac{\partial}{\partial f(u)} \left(\sum_{u \in V} \phi(\|(\nabla_w f)(u)\|_q) \right) = \sum_{v \sim u} \alpha_{uv}^{q,\phi,f} (f(u) - f(v)) |f(v) - f(u)|^{q-2} . \quad (3.72)$$

Démonstration. Rappelons tout d'abord que la norme \mathcal{L}_q , avec $0 < q < \infty$, du gradient symétrique ∇_w d'une fonction $f \in \mathcal{H}(V)$ sur un graphe pondéré $G = (V, E, w)$ en un sommet $u \in V$ est

$$\|(\nabla_w f)(u)\|_q = \left[\sum_{v \sim u} w_{uv}^{q/2} |f(v) - f(u)|^q \right]^{1/q}. \quad (3.73)$$

La dérivée partielle en un sommet particulier $u \in V$ dépend d'une part, de la variation de la fonction f en ce sommet, et d'autre part, des variations des sommets voisins. Nous avons alors la décomposition suivante.

$$\frac{\partial}{\partial f} \sum_{v \in V} \phi(\|(\nabla_w f)(v)\|_q) \Big|_u \quad (3.74)$$

$$= \frac{\partial \phi(\|(\nabla_w f)(u)\|_q)}{\partial f(u)} + \sum_{v \sim u} \frac{\partial \phi(\|(\nabla_w f)(v)\|_q)}{\partial f(u)} \quad (3.75)$$

$$= \phi'(\|(\nabla_w f)(u)\|_q) \frac{\partial \|(\nabla_w f)(u)\|_q}{\partial f(u)} + \sum_{v \sim u} \phi'(\|(\nabla_w f)(v)\|_q) \frac{\partial \|(\nabla_w f)(v)\|_q}{\partial f(u)}. \quad (3.76)$$

Étudions tout d'abord, les variations du premier terme de la décomposition

$$\phi'(\|(\nabla_w f)(u)\|_q) \frac{\partial \|(\nabla_w f)(u)\|_q}{\partial f(u)}. \quad (3.77)$$

En utilisant l'expression de la norme \mathcal{L}_q , avec $0 < q < +\infty$,

$$\phi'(\|(\nabla_w f)(u)\|_q) \frac{\partial}{\partial f(u)} \left[\sum_{v \sim u} w_{uv}^{q/2} |f(v) - f(u)|^q \right]^{1/q} \quad (3.78)$$

$$= \frac{1}{q} \phi'(\|(\nabla_w f)(u)\|_q) \left[\sum_{v \sim u} w_{uv}^{q/2} |f(v) - f(u)|^q \right]^{\frac{1-q}{q}} \quad (3.79)$$

$$\cdot \frac{\partial}{\partial f(u)} \sum_{v \sim u} w_{uv}^{q/2} |f(v) - f(u)|^q \quad (3.80)$$

$$= \frac{\phi'(\|(\nabla_w f)(u)\|_q)}{q \|(\nabla_w f)(u)\|_q^{q-1}} \frac{\partial}{\partial f(u)} \sum_{v \sim u} w_{uv}^{q/2} |f(v) - f(u)|^q. \quad (3.81)$$

En se positionnant en un sommet particulier $u_1 \in V$, la dérivée ne dépend que des arêtes incidentes à u_1 , donc des sommets v_1, \dots, v_k qui sont connectés au sommet u_1 . En utilisant également la dérivée usuelle suivante, $|a|' = a/|a|$, nous obtenons

pour la dernière dérivée restante

$$\frac{\partial}{\partial f(u)} \sum_{v \sim u} w_{uv}^{q/2} |f(v) - f(u)|^q \Big|_{u_1} \quad (3.82)$$

$$= -q \sum_{v \sim u_1} w_{u_1 v}^{q/2} (f(v) - f(u_1)) |f(v) - f(u_1)|^{q-2} \quad (3.83)$$

$$+ q w_{u_1 v_1}^{q/2} (f(u_1) - f(v_1)) |f(u_1) - f(v_1)|^{q-2} \quad (3.84)$$

$$+ \dots + q w_{u_1 v_k}^{q/2} (f(u_1) - f(v_k)) |f(u_1) - f(v_k)|^{q-2} \quad (3.85)$$

$$= -q \sum_{v \sim u_1} w_{u_1 v}^{q/2} (f(v) - f(u_1)) |f(v) - f(u_1)|^{q-2} \quad (3.86)$$

$$+ q \sum_{v \sim u_1} w_{u_1 v}^{q/2} (f(u_1) - f(v)) |f(u_1) - f(v)|^{q-2} \quad (3.87)$$

$$= q \sum_{v \sim u_1} w_{u_1 v}^{q/2} (f(u_1) - f(v)) |f(v) - f(u_1)|^{q-2} . \quad (3.88)$$

En utilisant cette dernière expression dans (3.81), nous avons finalement

$$(3.81) = \frac{\phi'(\|(\nabla_w f)(u)\|_q)}{\|(\nabla_w f)(u)\|_q^{q-1}} \sum_{v \sim u} w_{uv}^{q/2} (f(u) - f(v)) |f(v) - f(u)|^{q-2} . \quad (3.89)$$

Étudions maintenant les variations du second terme de la décomposition (3.74). De manière similaire au premier terme, nous avons

$$\sum_{v \sim u} \phi'(\|(\nabla_w f)(v)\|_q) \frac{\partial \|(\nabla_w f)(v)\|_q}{\partial f(u)} \quad (3.90)$$

$$= \sum_{v \sim u} \phi'(\|(\nabla_w f)(v)\|_q) \frac{\partial}{\partial f(u)} \left[\sum_{u \sim v} w_{uv}^{q/2} |f(u) - f(v)|^q \right]^{1/q} \quad (3.91)$$

$$= \sum_{v \sim u} \frac{\phi'(\|(\nabla_w f)(v)\|_q)}{q \|(\nabla_w f)(v)\|_q^{q-1}} \frac{\partial}{\partial f(u)} \sum_{u \sim v} w_{uv}^{q/2} |f(u) - f(v)|^q . \quad (3.92)$$

Le sommet $v \in V$ ne possède qu'un seul voisin qui correspond au sommet u : le sommet u lui même. De là, l'expression (3.92) devient

$$(3.92) = \sum_{v \sim u} \frac{\phi'(\|(\nabla_w f)(v)\|_q)}{q \|(\nabla_w f)(v)\|_q^{q-1}} \frac{\partial}{\partial f(u)} w_{uv}^{q/2} |f(u) - f(v)|^q \quad (3.93)$$

$$= \sum_{v \sim u} \frac{\phi'(\|(\nabla_w f)(v)\|_q)}{\|(\nabla_w f)(v)\|_q^{q-1}} w_{uv}^{q/2} (f(u) - f(v)) |f(u) - f(v)|^{q-2} . \quad (3.94)$$

Finalement, en utilisant les expressions (3.89) et (3.94) dans l'équation initiale nous avons pour un sommet $u \in V$

$$\frac{\partial}{\partial f(u)} \left(\sum_{u \in V} \phi(\|(\nabla_w f)(u)\|_q) \right) = \quad (3.95)$$

$$\frac{\phi'(\|(\nabla_w f)(u)\|_q)}{\|(\nabla_w f)(u)\|_q^{q-1}} \sum_{v \sim u} w_{uv}^{q/2} (f(u) - f(v)) |f(v) - f(u)|^{q-2} \quad (3.96)$$

$$+ \sum_{v \sim u} \frac{\phi'(\|(\nabla_w f)(v)\|_q)}{\|(\nabla_w f)(v)\|_q^{q-1}} w_{uv}^{q/2} (f(u) - f(v)) |f(u) - f(v)|^{q-2} \quad (3.97)$$

ce qui permet de retrouver la propriété 3.2.1

$$\frac{\partial}{\partial f(u)} \left(\sum_{u \in V} \phi(\|(\nabla_w f)(u)\|_q) \right) = \sum_{v \sim u} \alpha_{uv}^{q,\phi,f} (f(u) - f(v)) |f(v) - f(u)|^{q-2} \quad (3.98)$$

avec

$$\alpha_{uv}^{q,\phi,f} = w_{uv}^{q/2} \left(\frac{\phi'(\|(\nabla_w f)(u)\|_q)}{\|(\nabla_w f)(u)\|_q^{q-1}} + \frac{\phi'(\|(\nabla_w f)(v)\|_q)}{\|(\nabla_w f)(v)\|_q^{q-1}} \right). \quad (3.99)$$

□

En utilisant la propriété 3.2.1 dans l'équation (3.65), nous obtenons le système d'équations suivant en un sommet $u \in V$

$$\frac{\partial \mathcal{E}_1(f, f^0, \lambda)}{\partial f(u)} = \quad (3.100)$$

$$\sum_{v \sim u} \alpha_{uv}^{q,\phi,f} (f(u) - f(v)) |f(v) - f(u)|^{q-2} + \lambda (f(u) - f^0(u)) = 0. \quad (3.101)$$

Cette équation peut être considérée comme étant la version discrète de l'équation d'Euler-Lagrange.

Remarque. Nous pouvons constater que le gradient et la valeur absolue peuvent être nuls. Afin d'éviter des dénominateurs nuls pour des valeurs de $q < 2$, nous utilisons des versions régularisées de la norme du gradient et de la valeur absolue. Pour deux sommets u et $v \in V$

$$\|(\nabla_w f)(u)\|_q = \|(\nabla_w f)(u)\|_{q,\epsilon} = \sqrt{\|(\nabla_w f)(u)\|_q^2 + \epsilon^2} \quad (3.102)$$

$$|f(u) - f(v)| = |f(u) - f(v)|_\epsilon = |f(u) - f(v)| + \epsilon \quad (3.103)$$

avec ϵ un paramètre positif et $\epsilon \rightarrow 0$.

Dans la suite, pour des raisons de clarté, nous continuerons à noter la norme du gradient et la valeur absolue dans leur version non régularisée.

3.2.3 Algorithme discret d'approximation

Nous présentons ici une approche permettant d'approximer la solution du système (3.100).

Le système d'équations (3.100) peut se réécrire sous la forme suivante.

$$f(u) = \frac{\lambda f^0(u) + \sum_{u \sim v} \alpha_{uv}^{q,\phi,f} |f(v) - f(u)|^{q-2} f(v)}{\lambda + \sum_{u \sim v} \alpha_{uv}^{q,\phi,f} |f(v) - f(u)|^{q-2}}, \quad \forall u \in V. \quad (3.104)$$

En utilisant l'algorithme itératif de Gauss Jacobi, nous obtenons l'algorithme suivant permettant d'approximer la solution du système d'équations (3.104).

$$\begin{cases} f^{n+1}(u) &= \frac{\lambda f^0(u) + \sum_{u \sim v} \alpha_{uv}^{q,\phi,f^n} |f^n(v) - f^n(u)|^{q-2} f^n(v)}{\lambda + \sum_{u \sim v} \alpha_{uv}^{q,\phi,f^n} |f^n(v) - f^n(u)|^{q-2}}, \quad \forall u \in V. \\ f^{(0)}(u) &= f^0(u) \end{cases} \quad (3.105)$$

L'algorithme 1 résume l'approximation du système (3.105). La convergence est obtenue pour un nombre d'itérations N donné ou en calculant à chaque itération le critère d'arrêt suivant $\|f^{n+1} - f^n\|_2 < \tau$, avec $\tau > 0$ et $\tau \rightarrow 0$.

Données : une fonction initiale $f^0 : V \rightarrow \mathbb{R}$
Résultat : une fonction $f : V \rightarrow \mathbb{R}$ approximant le système (3.106)
Entrées : un graphe pondéré $G = (V, E, w)$

- 1 **pour tous les** sommets $u \in V$ **faire**
- 2 $f^{(0)}(u) = f^0(u)$
- 3 **tant que** $n \neq N$ ou $\|f^{n+1} - f^n\|_2 < \tau$ **faire**
- 4 **pour tous les** sommets $u \in V$ **faire**
- 5 $\beta_{uv}^{q,\phi,f^n} = \sum_{v \sim u} \alpha_{uv}^{q,\phi,f^n} |f^n(v) - f^n(u)|^{q-2}$
- 6 $f^{n+1}(u) = \frac{\lambda f^0(u) + \sum_{u \sim v} \beta_{uv}^{q,\phi,f^n} f^n(v)}{\lambda + \sum_{u \sim v} \beta_{uv}^{q,\phi,f^n}}$

Algorithme 1 : Algorithme discret approximant le système (3.104).

Remarque. Une autre manière de résoudre le système d'équations (3.100) est (comme dans le domaine continu) d'utiliser une méthode basée sur une descente à pas infinitésimal

$$\begin{cases} \frac{d f(u, t)}{d t} + \sum_{v \sim u} \alpha_{uv}^{q,\phi,f} (f(u) - f(v)) |f(v) - f(u)|^{q-2} + \lambda (f(u) - f^0(u)) = 0 \\ f(u, 0) = f^0(u) \end{cases} \quad (3.106)$$

Ce système décrit alors une famille de processus de diffusion discrets qui, contrairement aux équations aux dérivées partielles, n'a pas besoin d'étape de discrétisation spatiale.

Remarque. Le système (3.105) décrit une famille de filtres de voisinage. En effet, à chaque itération, la nouvelle valeur de la fonction f en un sommet $u \in V$ ne dépend que de deux quantités : sa valeur initiale f^0 et la moyenne pondérée des valeurs filtrées de la fonction f dans son voisinage $\mathcal{N}(u)$.

Dans la suite de cette section, nous appliquons cette famille de filtres pour le filtrage et la simplification de données avec différentes fonctions de régularisation ϕ et valeurs du paramètre q .

3.2.4 Cas des données vectorielles

Pour des raisons de clarté, les modèles que nous avons précédemment proposés ont été formulés pour des fonctions scalaires $f : V \rightarrow \mathbb{R}$.

Dans le cas des données multi variées, c'est-à-dire des fonctions vectorielles $f : V \rightarrow \mathbb{R}^m$, $f(u)$ est alors représentée par le vecteur $f(u) = (f_i(u))_{i=1, \dots, m}^T$ pour un sommet $u \in V$ donné. Dans ces conditions, la régularisation de ce type de fonction est réalisée sur chacune des composantes f_i du vecteur f (composante par composante). Le calcul de la variation locale du gradient de f_i est différent pour chacune des composantes, ce qui revient à appliquer m processus de régularisation en parallèle.

L'avantage de cette approche est de permettre la parallélisation des algorithmes. L'inconvénient est de ne pas prendre en compte (pour des applications particulières) l'aspect initialement vectoriel de la fonction initiale.

Pour palier ce problème et pour prendre en compte la corrélation entre les différentes composantes du vecteur, nous remplaçons la norme \mathcal{L}_q du gradient pondéré en un sommet u par une version multi dimensionnelle

$$\|(\nabla_w f)(u)\|_q = \|(\nabla_w f)(u)\|_{q,m} = \sqrt{\sum_{i=1}^m \|(\nabla_w f_i)(u)\|_q} . \quad (3.107)$$

De plus, la fonction de poids w sert également de terme de couplage permettant ainsi de prendre en compte l'aspect vectoriel des données à traiter.

Dans la suite, nous nous intéressons à deux cas particuliers du modèle variationnel (3.59) : les modèles de régularisation isotrope et anisotrope définis sur des graphes.

3.2.5 Modèle p -Tv isotrope

Nous présentons ici un cas particulier du modèle variationnel (3.59) correspondant au modèle p -Tv isotrope défini sur des graphes et proposé par Bougleux et Elmoataz [Bougleux et Elmoataz, 2005, Elmoataz *et al.*, 2008a, Bougleux *et al.*, 2009].

En effet, lorsque le noyau de régularisation $\phi(s) = \frac{1}{p}s^p$ et que $q = 2$, nous obtenons le modèle p -Tv isotrope

$$\min_{f:V \rightarrow \mathbb{R}} \left\{ \mathcal{E}_i(f, f^0, \lambda) = \frac{1}{p} \sum_{u \in V} \|(\nabla_w f)(u)\|_2^p + \sum_{u \in V} \frac{\lambda}{2} \|f(u) - f^0(u)\|_2^2 \right\} \quad (3.108)$$

qui se réécrit sous la forme

$$\min_{f:V \rightarrow \mathbb{R}} \left\{ \mathcal{E}_i(f, f^0, \lambda) = \frac{1}{p} \mathcal{R}_{w,p}^i(f) + \sum_{u \in V} \frac{\lambda}{2} \|f(u) - f^0(u)\|_2^2 \right\} \quad (3.109)$$

avec la fonctionnelle de régularisation $\mathcal{R}_{w,p}^i$ (3.1)

$$\mathcal{R}_{w,p}^i(f) = \sum_{u \in V} \|(\nabla_w f)(u)\|_2^p = \sum_{u \in V} \sum_{v \sim u} [w_{uv}(f(v) - f(u))^2]^{p/2} . \quad (3.110)$$

Propriété 3.2.2. La dérivée partielle de la fonctionnelle de régularisation $\mathcal{R}_{w,p}^i$ d'une fonction $f \in \mathcal{H}(V)$ correspond à l'opérateur de p -Laplace isotrope (2.34) à la page 21

$$\frac{1}{p} \frac{\partial \mathcal{R}_{w,p}^i(f)}{\partial f(u)} = (\Delta_{w,p}^i f)(u) . \quad (3.111)$$

Avec la fonction $\phi(s) = \frac{1}{p}s^p$, sa dérivée $\phi'(s) = s^{p-1}$ et le paramètre $q = 2$, la fonction $\alpha_{uv}^{q,\phi,f}$ (3.99) s'écrit

$$\alpha_{uv}^{2,\phi,f} = w_{uv} \left(\frac{\phi'(\|(\nabla_w f)(u)\|_2)}{\|(\nabla_w f)(u)\|_2} + \frac{\phi'(\|(\nabla_w f)(v)\|_2)}{\|(\nabla_w f)(v)\|_2} \right) \quad (3.112)$$

$$= w_{uv} (\|(\nabla_w f)(u)\|_2^{p-2} + \|(\nabla_w f)(v)\|_2^{p-2}) , \quad (3.113)$$

à partir de l'équation (3.98), la dérivée partielle en un sommet $u \in V$ de cette fonctionnelle est alors

$$\frac{1}{p} \frac{\partial \mathcal{R}_{w,p}^i(f)}{\partial f(u)} = \sum_{v \sim u} w_{uv} (\|(\nabla_w f)(u)\|_2^{p-2} + \|(\nabla_w f)(v)\|_2^{p-2}) (f(u) - f(v)) \quad (3.114)$$

Remarque. Nous pouvons remarquer que lorsque le paramètre $p < 1$, le problème d'optimisation n'est pas convexe, ce qui ne garantit pas d'obtenir le minimum global de la minimisation. Cependant, dans la suite, nous montrerons des expérimentations où le paramètre $p \rightarrow 0$ afin d'illustrer le comportement de ce type de filtres dans ce cas particulier.

La solution du problème (3.108) en un sommet $u \in V$, s'obtient en utilisant la propriété 3.2.2 page 53

$$\frac{\partial \mathcal{E}_i(f, f^0, \lambda)}{\partial f(u)} = \frac{\partial}{\partial f(u)} \left(\frac{1}{p} \mathcal{R}_{w,p}^i(f) + \frac{\lambda}{2} \|f(u) - f^0(u)\|_2^2 \right) \quad (3.115)$$

$$= (\Delta_{w,p}^i f)(u) + \lambda (f(u) - f^0(u)) = 0 . \quad (3.116)$$

En utilisant l'expression du p -Laplacien isotrope nous obtenons l'équation d'Euler-Lagrange suivante

$$\sum_{v \sim u} w_{uv} (\|(\nabla_w f)(v)\|_2^{p-2} + \|(\nabla_w f)(u)\|_2^{p-2}) (f(u) - f(v)) + \lambda (f(u) - f^0(u)) = 0 \quad (3.117)$$

ce qui est équivalent au système suivant

$$f(u) = \frac{\lambda f^0(u) + \sum_{v \sim u} w_{uv} (\|(\nabla_w f)(v)\|_2^{p-2} + \|(\nabla_w f)(u)\|_2^{p-2}) f(v)}{\lambda + \sum_{v \sim u} w_{uv} (\|(\nabla_w f)(v)\|_2^{p-2} + \|(\nabla_w f)(u)\|_2^{p-2})} . \quad (3.118)$$

Afin d'approximer la solution du système (3.118), nous pouvons utiliser l'algorithme discret (3.105). Pour tout $u \in V$, nous obtenons le processus itératif suivant

$$\begin{cases} f^{n+1}(u) = \frac{\lambda f^0(u) + \sum_{v \sim u} w_{uv} (\|(\nabla_w f^n)(v)\|_2^{p-2} + \|(\nabla_w f^n)(u)\|_2^{p-2}) f^n(v)}{\lambda + \sum_{v \sim u} w_{uv} (\|(\nabla_w f^n)(v)\|_2^{p-2} + \|(\nabla_w f^n)(u)\|_2^{p-2})} \\ f^{(0)}(u) = f^0(u) \end{cases} . \quad (3.119)$$

Remarque. Dans le cas où le paramètre $p = 1$, la version dans le domaine continu de l'énergie ainsi formulé peut également être résolu à partir de l'algorithme de projection de Chambolle [Chambolle, 2004].

Remarque. En utilisant les processus de diffusion décrits par le système (3.106) pour approximer la solution de (3.118), nous obtenons une famille de diffusions isotropes qui inclut et étend des processus bien connus dans les domaines du traitement des images ou de l'informatique graphique.

Ces processus de diffusion sont généralement formulés sans terme d'attache aux données, c'est-à-dire que $\lambda = 0$, et ont été analysés par Weickert dans [Weickert, 1998] dans le contexte du traitement des images.

Liens avec des filtres de voisinage

L'algorithme (3.119) décrit une famille de filtres de voisinage. Bougleux et Elmoataz [Bougleux, 2007, Elmoataz et al., 2008a, Bougleux et al., 2009] font le lien

entre cette famille de filtres et un certain nombre de filtres connus pour le traitement des images dans le domaine spatial ou spectral.

Soit $G = (V, E, w)$ un graphe pondéré associé à une image en 2D définie comme une fonction $f^0 \in \mathcal{H}(V)$.

Lorsqu'il n'y a pas d'attache aux données ($\lambda = 0$) et quand le paramètre $p = 2$, le système (3.119) se réécrit

$$\begin{cases} f^{n+1}(u) = \frac{\sum_{v \sim u} w_{uv} f^n(v)}{\sum_{v \sim u} w_{uv}} \\ f^{(0)}(u) = f^0(u) \end{cases} . \quad (3.120)$$

À la convergence de l'algorithme, la fonction f obtenue est alors la solution de l'équation de la chaleur

$$(\Delta_{w,2}^i f)(u) = (\Delta_w f)(u) = 0, \quad \forall u \in V . \quad (3.121)$$

La fonction f est également la solution de la fonctionnelle de régularisation

$$\mathcal{R}_{w,2}^i(f) = \sum_{u \in V} \|(\nabla_w f)(u)\|_2^2 \quad (3.122)$$

correspondant à la solution de la minimisation du problème de Dirichlet.

De manière non exhaustive, une itération de l'algorithme (3.120) correspond aux filtres suivants pour le traitement des images (les sommets u et v sont associés aux coordonnées spatiales des pixels de l'image).

- Le filtre Gaussien si la fonction de poids associée à une arête uv est de la forme

$$w_{uv} = \exp\left(-\frac{\|u - v\|_2^2}{2\sigma^2}\right) . \quad (3.123)$$

- Le filtre σ [Yaroslavsky, 1985, Lee, 1983] avec

$$w_{uv} = \exp\left(-\frac{\|f^0(u) - f^0(v)\|_2^2}{2\sigma^2}\right) . \quad (3.124)$$

- Le filtre bilatéral [Tomasi et Manduchi, 1998, Barash, 2002] avec

$$w_{uv} = \exp\left(-\frac{\|u - v\|_2^2}{2\sigma^2}\right) \exp\left(-\frac{\|f^0(u) - f^0(v)\|_2^2}{\sigma^2}\right) . \quad (3.125)$$

- Le filtre à moyennes non locales [Buades *et al.*, 2008] avec

$$w_{uv} = \exp\left(-\frac{\rho(\mathcal{F}(f^0, u), \mathcal{F}(f^0, v))^2}{\sigma^2}\right) . \quad (3.126)$$

Ce poids correspond à la fonction de poids g_4 (2.17) page 17 décrite dans le chapitre 2. Le lecteur intéressé pourra également se référer à [Szlam *et al.*, 2008, Peyré, 2009] pour d'autres méthodes utilisant le filtrage non local avec des variétés de patches issus d'une image.

Quand le paramètre $\lambda \neq 0$ et que la fonction de poids w est constante, alors le système (3.119) correspond à la version discrète de filtres définis dans le domaine continu et basés sur des équations aux dérivées partielles. Quand $p = 1$, il s'agit de la régularisation basée sur la variation totale [Osher et Shen, 2000, Chan *et al.*, 2001]. Quand $p = 2$, le système (3.119) correspond à la régularisation de Tikhonov [Tikhonov et Arsenin, 1977].

Finalement, en réécrivant le système (3.119) sous forme matricielle, nous obtenons un filtrage par diffusion sur des graphes (basé sur des matrices de Markov) et qui correspond à un filtrage dans le domaine spectral. On notera que l'étude des propriétés des Laplaciens matriciels trouvent de nombreuses applications en analyse de données discrètes telles que la segmentation d'images, la réduction de dimensions, ou encore la l'apprentissage semi supervisé. Le lecteur intéressé pourra, par exemple, se référer à [Chung, 1997, Shi et Malik, 2000, Coifman *et al.*, 2005, Lafon et Lee, 2006, Belkin *et al.*, 2006, Szlam *et al.*, 2008, Bougleux *et al.*, 2007] ainsi qu'aux références associées.

3.2.6 Modèle p -TV isotrope, filtrage et simplification de données

Comme nous l'avons déjà mentionné, un des avantages de notre approche est que nos modèles sont définis sur des graphes. Ceci nous permet de traiter n'importe quel type de données pouvant être représentées par un graphe sans étape de discrétisation spatiale, contrairement aux méthodes basées sur les équations aux dérivées partielles.

Nous proposons ici une série d'expérimentations montrant le comportement du modèle isotrope pour le filtrage et la simplification de données arbitraires. Nous montrons, d'une part, que nos méthodes permettent de débruiter des variétés dans le domaine spatial sans avoir recours à une projection dans le domaine spectral ou à une réduction de dimension préalable. D'autre part, nous montrons que le filtrage et le débruitage de données peuvent servir de pré-traitement dans l'amélioration des performances des méthodes de regroupement, de classification ou d'apprentissage.

Les expérimentations suivantes n'ont pas pour but de résoudre une problématique particulière dans le domaine de l'apprentissage ou du traitement de données. Elles permettent uniquement d'illustrer les comportements des filtres basés sur le modèle isotrope et de montrer le potentiel de notre méthodologie dans le domaine du traitement de données.

Remarque. *Dans les expérimentations suivantes, nous illustrons le comportement des filtres pour des valeurs de $p < 1$. Dans ce cas, seul un minimum local est obtenu et les différents résultats doivent être comparés à ce titre.*

Débruitage de collections d'images

La figure 3.2 montre un exemple de débruitage de variété d'images. Les données initiales proviennent de la base de données de l'Usps (*United States Postal Service handwritten digits data set*) contenant des images de chiffres manuscrits allant du chiffre « 0 » au chiffre « 9 ». Chaque image de la base est de taille 16×16 .

Nous utilisons un échantillon de test de 100 images provenant des chiffres « 0 » et « 1 » choisis de manière aléatoire. Ces données sont ensuite bruitées par un bruit

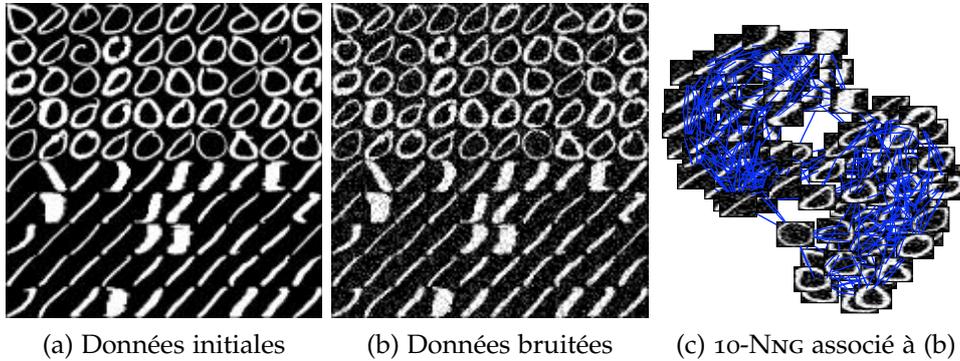


FIG. 3.1: Échantillon de tests pour le débruitage des données de la base USPS par filtrage isotrope (figure 3.2) et anisotrope (figure 3.9 page 67). (b) : données bruitées à partir de (a) par un bruit Gaussien de variance égale à 40. (c) : 10-NNG pondéré par la fonction g_3 , $\text{card}(V) = 100$ et chaque sommet est caractérisé par une fonction initiale $f^0 : V \rightarrow \mathbb{R}^{16 \times 16}$.

Gaussien de variance de 40 et de moyenne nulle. Les figures 3.1a et 3.1b montrent les données initiales et celles bruitées. Afin de traiter cette variété, un 10-NNG pondéré par la fonction de poids g_3 est construit à partir des données bruitées. Chaque sommet est caractérisé par une fonction initiale $f^0 : V \rightarrow \mathbb{R}^{16 \times 16}$ correspondant aux niveaux de gris des pixels des images.

La figure 3.2 montre les résultats du filtrage à partir du modèle p -Tv isotrope pour différentes valeurs des paramètres p et λ . Chaque résultat présenté est composé

- du résultat du filtrage (la fonction finale f);
- de la différence entre le filtrage et la fonction initiale ($|f^0 - f|$);
- d’une valeur de rapport signal sur bruit en crête : PSNR^2 . Les trois meilleures valeurs sont mises en gras.

Nous pouvons constater que les résultats, pour les paramètres $p = 2$ (n’importe quelle valeur de λ) et $\lambda = 0$ (n’importe quelle valeur de p), donnent approximativement les mêmes effets de filtrage. Le traitement a tendance à uniformiser les données pour obtenir de nouveaux modèles moyens. Ce type de traitement peut être intéressant dans le cadre d’une méthode de classification dans le sens où les deux classes de chiffres deviennent visuellement distinctes.

Les deux meilleurs valeurs de PSNR sont obtenues pour les cas où les paramètres p et λ sont faibles. Malgré cela, nous pouvons voir que lorsque les paramètres p et λ sont proches ou égaux à 1, nous obtenons une bonne valeur du PSNR et le filtrage tend à mieux respecter la structure des données (au vue des images de différence).

Ce type de traitement peut être considéré comme une alternative intéressante aux méthodes utilisées pour résoudre les problèmes liés à la reconstruction de variétés ou les problèmes de la pré-image dans le domaine de l’apprentissage. Ces méthodes utilisent usuellement une projection dans le domaine spectral ou une réduction de dimension comme étape préalable à la reconstruction (pour obtenir la

²de l’anglais *Peak Signal to Noise Ratio*

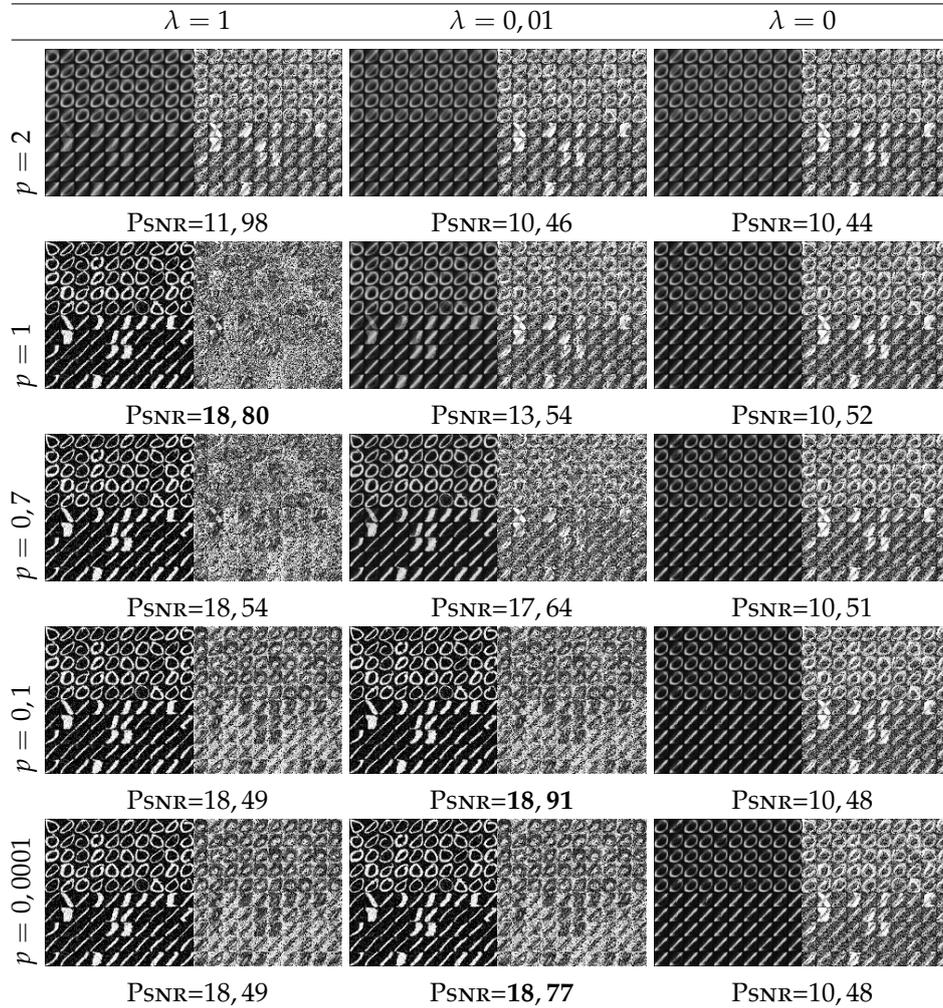


FIG. 3.2: Filtrage isotrope d'une variété d'image (base USPS) avec différentes valeurs des paramètres p et λ . Les données filtrées sont celles de la figure 3.1. Chaque résultat présente le résultat du filtrage (la fonction finale f), la différence avec les données initiales ($|f^0 - f|$) et une valeur de PSNR.

variété considérée). Notre approche a l'avantage de permettre le débruitage de la variété directement dans le domaine spatial sans étape préalable (contrairement aux méthodes proposées par [Coifman *et al.*, 2005, Szlam *et al.*, 2008]). Le lecteur intéressé par ces méthodes de filtrage dans le domaine spectral pourra, par exemple, se référer aux articles suivants [Park *et al.*, 2007, Thorstensen *et al.*, 2009] ainsi qu'à leurs références associées.

Simplification de variétés

Les figures 3.4 et 3.5 montrent les effets du filtrage et du débruitage basés sur la régularisation isotrope sur une variété bruitée représentant une hélice toroïdale.

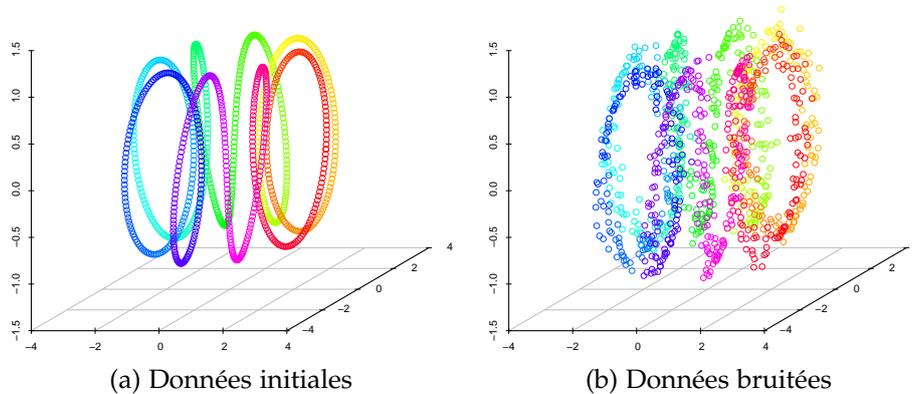


FIG. 3.3: Hélice toroïdale bruitée pour illustrer la simplification de variétés par filtrage isotrope (figures 3.4 et 3.5) et anisotrope (figures 3.10 page 68 et 3.11 page 69). (b) : données bruitées à partir de (a) avec un bruit Gaussien de variance égale à 10.

La figure 3.3 montre les données initiales et les données bruitées par un bruit Gaussien de variance égale à 10. Afin de régulariser ces données, un 15-NNG pondéré par la fonction de poids g_3 est construit à partir des données bruitées. Chaque sommet du graphe (représentant un point de l'hélice) est associé aux coordonnées spatiales de celui-ci ($f^0 : V \rightarrow \mathbb{R}^3$).

Les figures 3.4 et 3.5 montrent les effets et l'évolution du filtrage sur les données pour différentes itérations n et différentes valeurs des paramètres p et λ .

La figure 3.4 présente des résultats pour une valeur de $\lambda = 1$ et, pour la figure 3.5, une valeur de $\lambda = 0$

Nous pouvons constater que lorsque le paramètre $\lambda = 1$ (figure 3.4), le filtrage a tendance à respecter la topologie initiale de la variété tout en la débruitant. Nous pouvons également remarquer que, en fonction du paramètre de régularité p , les courbures de l'hélice initiale sont plus ou moins respectées. Quand le paramètre $p = 2$, le filtrage lisse l'ensemble des données y compris dans les zones de fortes courbures. Quand le paramètre $p = 1$, les courbures initiales sont mieux préservées. Quand le paramètre p tend vers zéro, la topologie de l'hélice (représentée par la fonction f) tend à devenir constante par morceaux.

La figure 3.5 montre l'effet du filtrage isotrope lorsque il n'y a pas de terme d'attache aux données ($\lambda = 0$). Cette expérimentation illustre également le comportement du filtrage pour différentes valeurs du paramètre p . D'une manière générale, quand $\lambda = 0$, le filtrage a pour effet de déplier complètement la variété initiale, en particulier pour les cas où $p = 2$ et $p = 1$. Lorsque le paramètre p tend vers zéro, nous pouvons observer le même effet, mais le filtrage a tendance à préserver les points où les courbures sont fortes.

Finalement, nous pouvons voir que le filtrage de ce type de données à partir du modèle de régularisation isotrope peut être considéré comme un processus de projection ou de réduction de dimension de la variété. En effet, notre approche du filtrage par le p -Laplacien isotrope peut être considéré comme une généralisation des méthodes d'apprentissage de variétés basées sur le 2-Laplacien [Belkin et Niyogi, 2003]. Notre approche possède l'avantage de permettre de réaliser cette

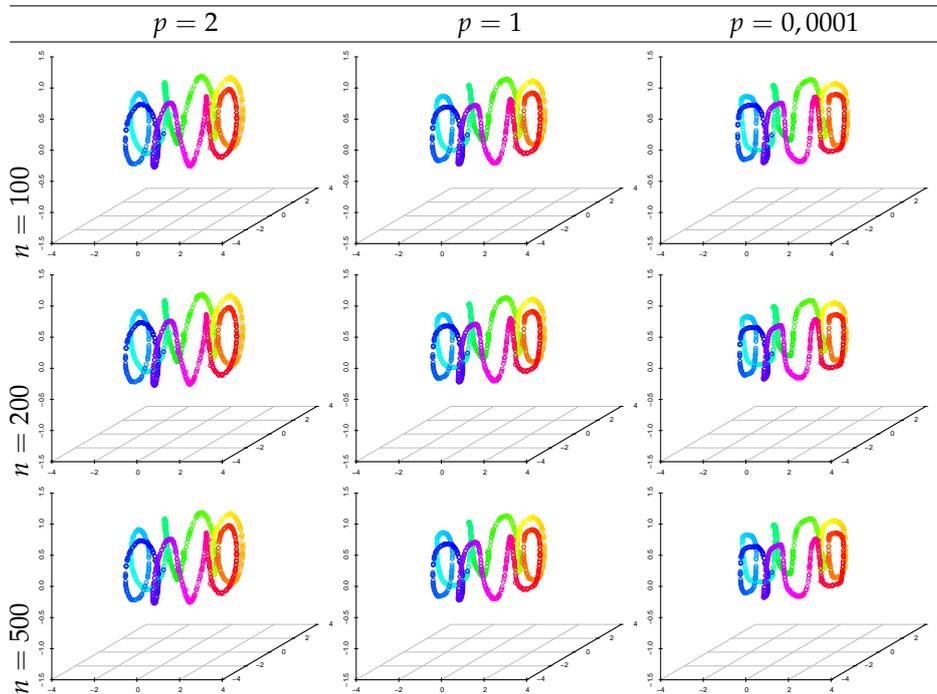


FIG. 3.4: Évolution (en fonction du nombre d'itérations n) du filtrage isotrope d'une variété avec différentes valeurs des paramètres p et $\lambda = 1$. Les données traitées correspondent à celles représentées par la figure 3.3 page 59.

simplification des données directement dans le domaine spatial.

Filtrage et simplification comme pré-traitement

Dans le contexte du traitement des images, le filtrage et la simplification permettent souvent de faciliter une méthode donnée de segmentation.

Dans le même esprit, nous proposons d'utiliser la famille de filtres basée sur le modèle isotrope, pour filtrer et simplifier des données avant une étape de regroupement ou de classification.

La figure 3.6 montre deux exemples de bases de données réelles servant à illustrer l'intérêt du filtrage et de la simplification comme étape de pré-traitement. Ces bases proviennent du dépôt de l'université de Californie, Irvine (UCI) [Asuncion et Newman, 2007] et sont bien connues dans la communauté de l'apprentissage.

La figure 3.6a montre un premier exemple de base de données (les données « iris »). Cette base contient des données décrivant 3 classes de fleurs. Elle contient 150 instances (50 par classes) et chaque entrée est décrite par 4 attributs. La figure 3.6b montre un second exemple de base de données (les données « wine »). Cette base contient des données décrivant les composés chimiques du vin et est composée de 3 classes. Chaque instance de cette base est décrite par 13 attributs.

La figure 3.6 montre des projections des données initiales par paires d'attributs. Pour la base « wine », pour des raisons de clarté, seuls les 4 premiers attributs sur

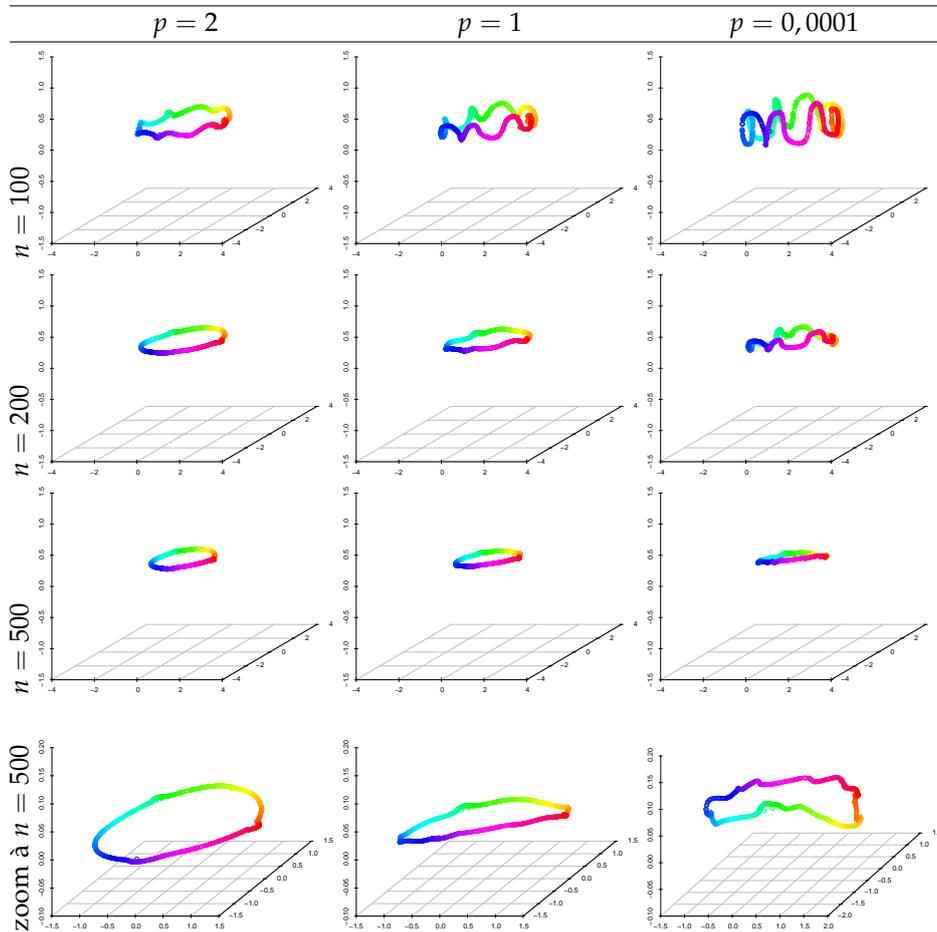


FIG. 3.5: Évolution (en fonction du nombre d'itérations n) du filtrage isotrope d'une variété avec différentes valeurs des paramètres p et $\lambda = 0$. Les données traitées correspondent à celles représentées par la figure 3.3. Les figures de la dernière ligne correspondent à un zoom pour l'itération $n = 500$.

les 13 sont représentés. Chacune des classes des bases de données est représentée par une couleur différente.

Afin de traiter ces bases de données, des 30-NNG pondérés par la fonction de poids g_3 sont construits à partir des données initiales. Pour les données « iris », le graphe contient 150 sommets et chaque sommet est décrit par une fonction $f^0 : V \rightarrow \mathbb{R}^4$. Pour la base « wine », le graphe contient 178 sommets et chaque sommet est décrit par une fonction $f^0 : V \rightarrow \mathbb{R}^{13}$.

Les figures 3.7 et 3.8 montrent les résultats du filtrage par le modèle de régularisation isotrope pour différentes valeurs des paramètres p et λ des bases « iris » et « wine ». Ces résultats montrent l'effet du filtrage sur les données qui ont tendance à se regrouper (par classe) dans des parties distinctes de l'espace des attributs (notamment pour une des classe de la base « iris »). Comme dans le cas du débruitage de l'hélice toroïdale (figures 3.4 et 3.5), les filtrages peuvent être vus comme une

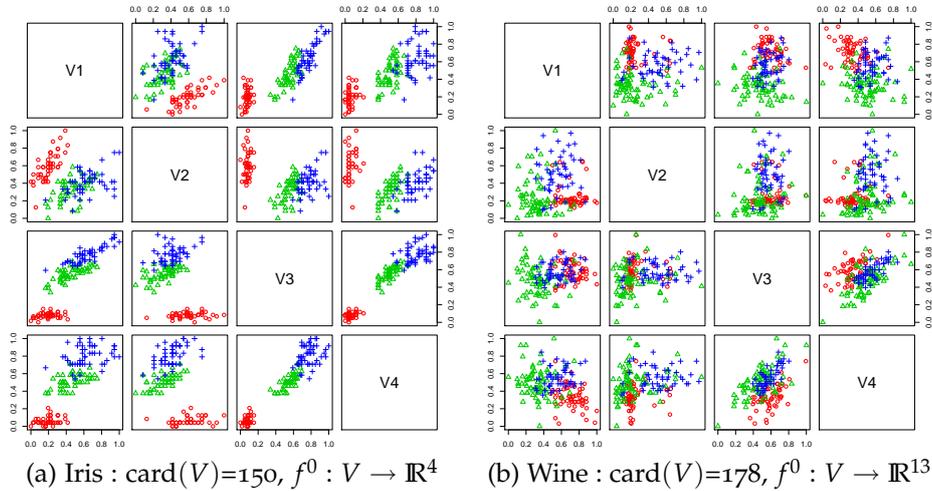


FIG. 3.6: Bases de données provenant du dépôt de l'UCI pour illustrer le filtrage et la simplification de données (isotrope : figures 3.7 et 3.8, anisotrope : figures 3.12 page 70 et 3.13 page 70) comme pré-traitement à une étape de regroupement ou de classification. Les figures représentent des projections par paires selon les attributs de chaque base. Pour la base « wine », la projection est faite sur les 4 premiers attributs des 13 contenus dans la base initiale.

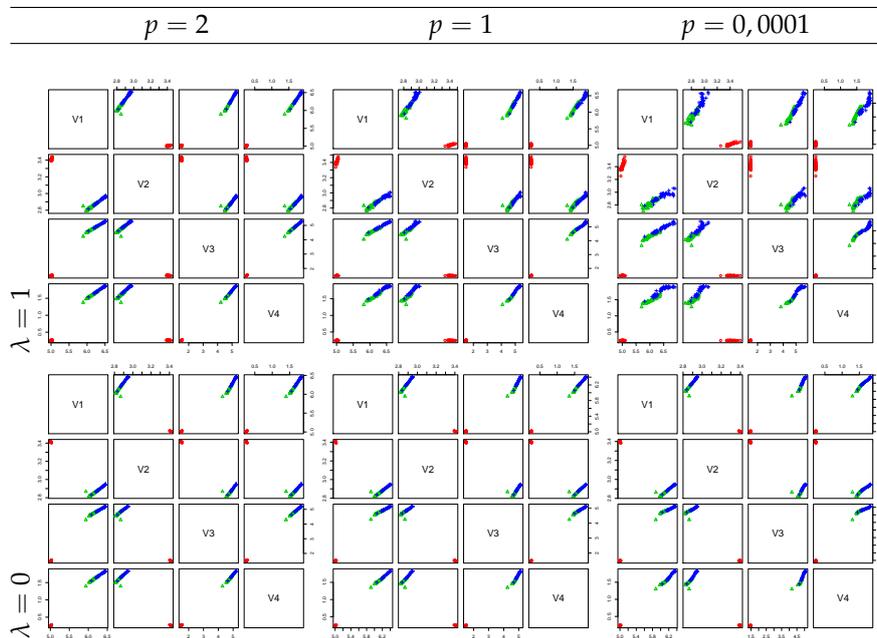


FIG. 3.7: Filtrage isotrope de la base de données « iris » avec différentes valeurs des paramètres p et λ . La première ligne montre les résultats pour $\lambda = 1$ et la seconde avec $\lambda = 0$.

réduction de dimension calculée dans le domaine spatial.

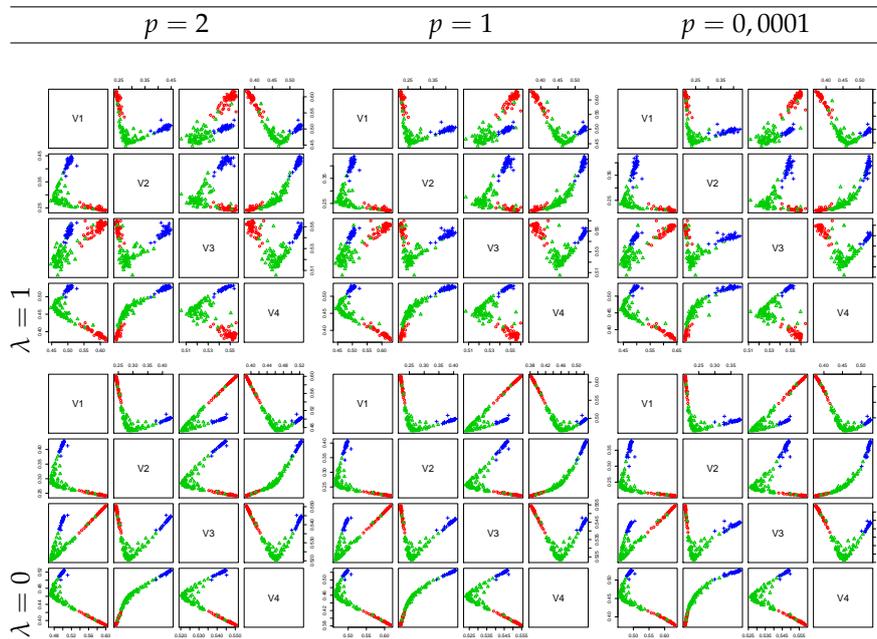


FIG. 3.8: Filtrage isotrope de la base de données « wine » avec différentes valeurs des paramètres p et λ . La première ligne montre les résultats pour $\lambda = 1$ et la seconde avec $\lambda = 0$.

TAB. 3.2: Taux de classification de la base de données « iris » et « wine » pour différentes valeurs des paramètres p et λ (comparaison entre les données initiales et les données filtrées avec le modèle isotrope). Les meilleurs taux sont présentés en gras. La classification est obtenue avec l'algorithme des k moyennes où le nombre d'attributs utilisés est spécifié dans la deuxième colonne.

Base	Attributs utilisés	Données initiales	Données filtrées			
			$p = 2$	$p = 1$	$p = 0,0001$	
Iris	4	89,3%	$\lambda = 1$	94,0%	90,7%	90,7%
			$\lambda = 0$	94,0%	94,7%	91,3%
wine	13	96,6%	$\lambda = 1$	94,4%	94,9%	94,9%
			$\lambda = 0$	94,4%	94,9%	94,9%
wine	4	83,7%	$\lambda = 1$	97,2%	97,8%	97,2%
			$\lambda = 0$	96,6%	97,2%	97,8%

Le tableau 3.2 montre et compare des résultats de classification entre les données initiales et les données filtrées pour les bases « iris » et « wine ». La classification est obtenue à partir de l'algorithme des k moyennes. La deuxième colonne de ce tableau spécifie le nombre d'attributs utilisés dans l'algorithme. Les résultats présentés correspondent à la valeur maximale des taux de classification obtenus sur 200 tours de tests (l'algorithme de k moyennes utilise une initialisation aléatoire).

Pour la base « iris », nous pouvons constater le bénéfice du filtrage. Chaque résultat de classification, à partir des données filtrées, est meilleur que celui obtenu à partir des données non filtrées. En ce qui concerne la base « wine », nous pouvons

constater (à la deuxième ligne du tableau 3.2) que le filtrage n'a pas amélioré le taux de classification lorsque nous utilisons l'ensemble des attributs de la base. Cela provient du fait que certains attributs de la base initiale sont suffisamment discriminants pour séparer les différentes classes. Par contre, nous pouvons noter que lorsque nous utilisons moins d'attributs (4 au lieu des 13, à la dernière ligne du tableau 3.2), nous obtenons de meilleurs résultats dans le cas des données filtrées que dans le cas des données non filtrées. Les attributs utilisés dans l'algorithme des k moyennes sont ceux présentés dans la figure 3.8. Nous pouvons remarquer que ces derniers résultats sont également meilleurs que ceux obtenus en utilisant l'ensemble des attributs.

Remarque. *La sélection des attributs pour la classification des bases de données est un problème dépendant des données et de l'application considérée. Dans cet exemple, nous avons sélectionné les 4 attributs de manière empirique dans le seul but de montrer la potentialité de notre méthodologie.*

Finalement, nous pouvons conclure sur l'intérêt et le bénéfice du filtrage et de la simplification de données qui permettent d'améliorer les résultats de classification. Des travaux connexes [Hein et Maier, 2007a, Hein et Maier, 2007b], basés sur le filtrage de variétés avec le 2-Laplacien matriciel, ont également montré l'intérêt de ce type de traitement dans les méthodes de classification. Nous pouvons considérer que nos approches sont une extension de ces travaux (avec le terme général de régularité p).

3.2.7 Modèle p -Tv anisotrope

Nous présentons ici un autre cas particulier du modèle variationnel (3.59) correspondant au modèle variationnel p -Tv anisotrope défini sur des graphes.

Avec le noyau de régularisation $\phi(s) = \frac{1}{2^p} s^p$ et pour $p = q$, nous obtenons le modèle p -Tv anisotrope suivant

$$\min_{f:V \rightarrow \mathbb{R}} \left\{ \mathcal{E}_a(f, f^0, \lambda) = \frac{1}{2^p} \sum_{u \in V} \|(\nabla_w f)(u)\|_p^p + \sum_{u \in V} \frac{\lambda}{2} \|f(u) - f^0(u)\|_2^2 \right\} \quad (3.127)$$

qui peut se réécrire de la manière suivante

$$\min_{f:V \rightarrow \mathbb{R}} \left\{ \mathcal{E}_a(f, f^0, \lambda) = \frac{1}{2^p} \mathcal{R}_{w,p}(f) + \sum_{u \in V} \frac{\lambda}{2} \|f(u) - f^0(u)\|_2^2 \right\} \quad (3.128)$$

où la fonctionnelle de régularisation $\mathcal{R}_{w,p}$ correspond à celle définie par l'équation (3.2) page 37

$$\mathcal{R}_{w,p}(f) = \sum_{u \in V} \|(\nabla_w f)(u)\|_p^p = \sum_{u \in V} \sum_{v \sim u} w_{uv}^{p/2} |f(v) - f(u)|^p . \quad (3.129)$$

Propriété 3.2.3. *La dérivée partielle de la fonctionnelle de régularisation $\mathcal{R}_{w,p}$ d'une fonction $f \in \mathcal{H}(V)$ correspond à l'opérateur de p -Laplace anisotrope (2.117) en page 29 et défini sur des graphes*

$$\frac{1}{2^p} \frac{\partial \mathcal{R}_{w,p}(f)}{\partial f(u)} = (\Delta_{w,p}^a f)(u) . \quad (3.130)$$

Avec la fonction $\phi(s) = \frac{1}{2p}s^p$, sa dérivée $\phi'(s) = s^{p-1}$ et le paramètre $q = p$, la fonction $\alpha_{uv}^{q,\phi,f}$ s'écrit

$$\alpha_{uv}^{p,\phi,f} = w_{uv}^{p/2} \left(\frac{\phi'(\|(\nabla_w f)(u)\|_p)}{\|(\nabla_w f)(u)\|_p^{p-1}} + \frac{\phi'(\|(\nabla_w f)(v)\|_p)}{\|(\nabla_w f)(v)\|_p^{p-1}} \right) \quad (3.131)$$

$$= 2pw_{uv}^{p/2} . \quad (3.132)$$

La dérivée partielle en un sommet $u \in V$ de cette fonctionnelle est

$$\frac{1}{2p} \frac{\partial \mathcal{R}_{w,p}(f)}{\partial f(u)} = \sum_{v \sim u} w_{uv}^{p/2} (f(u) - f(v)) |f(v) - f(u)|^{p-2} \quad (3.133)$$

$$= (\Delta_{w,p}^a f)(u) . \quad (3.134)$$

La solution du problème (3.127) en un sommet $u \in V$ s'obtient en utilisant la propriété 3.2.3 page 64

$$\frac{\partial \mathcal{E}_a(f, f^0, \lambda)}{\partial f(u)} = \frac{\partial}{\partial f(u)} \left(\frac{1}{2p} \mathcal{R}_{w,p}(f) + \frac{\lambda}{2} \|f - f^0\|_2^2 \right) \quad (3.135)$$

$$= (\Delta_{w,p}^a f)(u) + \lambda(f(u) - f^0(u)) = 0 . \quad (3.136)$$

En utilisant l'expression du p -Laplacien anisotrope, nous obtenons l'équation d'Euler-Lagrange suivante

$$\sum_{v \sim u} w_{uv}^{p/2} (f(u) - f(v)) |f(u) - f(v)|^{p-2} + \lambda(f(u) - f^0(u)) = 0 , \quad (3.137)$$

ce qui est équivalent au système suivant

$$f(u) = \frac{\lambda f^0(u) + \sum_{v \sim u} w_{uv}^{p/2} (f(u) - f(v)) |f(u) - f(v)|^{p-2} f(v)}{\lambda + \sum_{v \sim u} w_{uv}^{p/2} (f(u) - f(v)) |f(u) - f(v)|^{p-2}} . \quad (3.138)$$

Afin d'approximer la solution du système (3.138), nous pouvons utiliser l'algorithme discret (3.105). Pour tout $u \in V$, nous obtenons le processus itératif suivant.

$$\begin{cases} f^{n+1}(u) = \frac{\lambda f^0(u) + \sum_{v \sim u} w_{uv}^{p/2} (f^n(u) - f^n(v)) |f^n(u) - f^n(v)|^{p-2} f^n(v)}{\lambda + \sum_{v \sim u} w_{uv}^{p/2} (f^n(u) - f^n(v)) |f^n(u) - f^n(v)|^{p-2}} \\ f^{(0)}(u) = f^0(u) \end{cases} . \quad (3.139)$$

Remarque. Nous pouvons remarquer que lorsque le paramètre $q = 2$, les modèles p -Tv isotrope et anisotrope ont la même expression, ainsi que leurs solutions et leurs algorithmes.

De plus, comme pour le modèle isotrope, dans le cas des données vectorielles, la fonction de poids w sert de terme de couplage. Il est également possible d'intégrer le caractère vectoriel des données en considérant la version vectorielle de la valeur absolue dans le système (3.138) [Bougleux, 2007].

3.2.8 Modèle p -Tv anisotrope, filtrage et simplification de données

Comme pour le modèle p -Tv isotrope, notre formalisme basé sur les graphes permet de régulariser n'importe quel type de données discrètes.

Nous proposons ici une série d'expérimentations montrant le comportement du modèle anisotrope pour le filtrage et la simplification de données arbitraires. Les expérimentations suivantes sont effectuées avec les mêmes données que pour le cas isotrope afin de pouvoir comparer le comportement des deux modèles.

Comme pour le modèle isotrope, ces expérimentations n'ont pas pour but de résoudre une problématique particulière dans le domaine de l'apprentissage ou du traitement de données. Elles permettent uniquement d'illustrer les comportements des filtres basés sur le modèle anisotrope et de montrer le potentiel de notre méthodologie.

Remarque. *Dans les expérimentations suivantes, comme pour le modèle isotrope, nous illustrons le comportement des filtres pour des valeurs de $p < 1$. Dans ce cas, seul un minimum local est obtenu et les différents résultats doivent être comparés à ce titre.*

Débruitage de collections d'images

La figure 3.9 montre un exemple de débruitage de variété d'images avec le modèle anisotrope. Afin de pouvoir comparer les effets du filtrage avec le modèle isotrope, nous utilisons les mêmes données que celles présentées par la figure 3.1 page 57. Nous utilisons également les mêmes paramètres et les mêmes graphes que pour les résultats obtenus avec la figure 3.2 page 58.

La figure 3.9 montre les résultats du filtrage à partir du modèle p -Tv anisotrope pour différentes valeurs des paramètres p et λ . Chaque résultat présenté est composé

- du résultat du filtrage (la fonction finale f);
- de la différence entre le filtrage et la fonction initiale ($|f^0 - f|$);
- d'une valeur PSNR où les trois meilleures valeurs sont mises en gras.

Nous pouvons constater que les résultats du filtrage par le modèle anisotrope sont meilleurs que ceux obtenus par le modèle isotrope, du point de vue des valeurs de PSNR; excepté pour les résultats obtenus avec le paramètre $p = 2$ où les deux modèles coïncident.

À partir des images de différences, nous pouvons voir que le filtrage anisotrope tend à mieux préserver la structure initiale de la variété tout en filtrant le bruit. Le comportement du traitement de ce type de variété par le modèle anisotrope est supérieur au modèle isotrope.

Simplification de variétés

Les figures 3.10 et 3.11 montrent les effets du filtrage par un débruitage basé sur la régularisation anisotrope.

Afin de pouvoir comparer les effets du filtrage avec le modèle isotrope, nous utilisons les mêmes données, celles de l'hélice toroïdale présentée par la figure 3.3 page 59. Nous utilisons également les mêmes paramètres et les mêmes graphes que pour les résultats obtenus avec les figures 3.4 page 60 et 3.5 page 61.

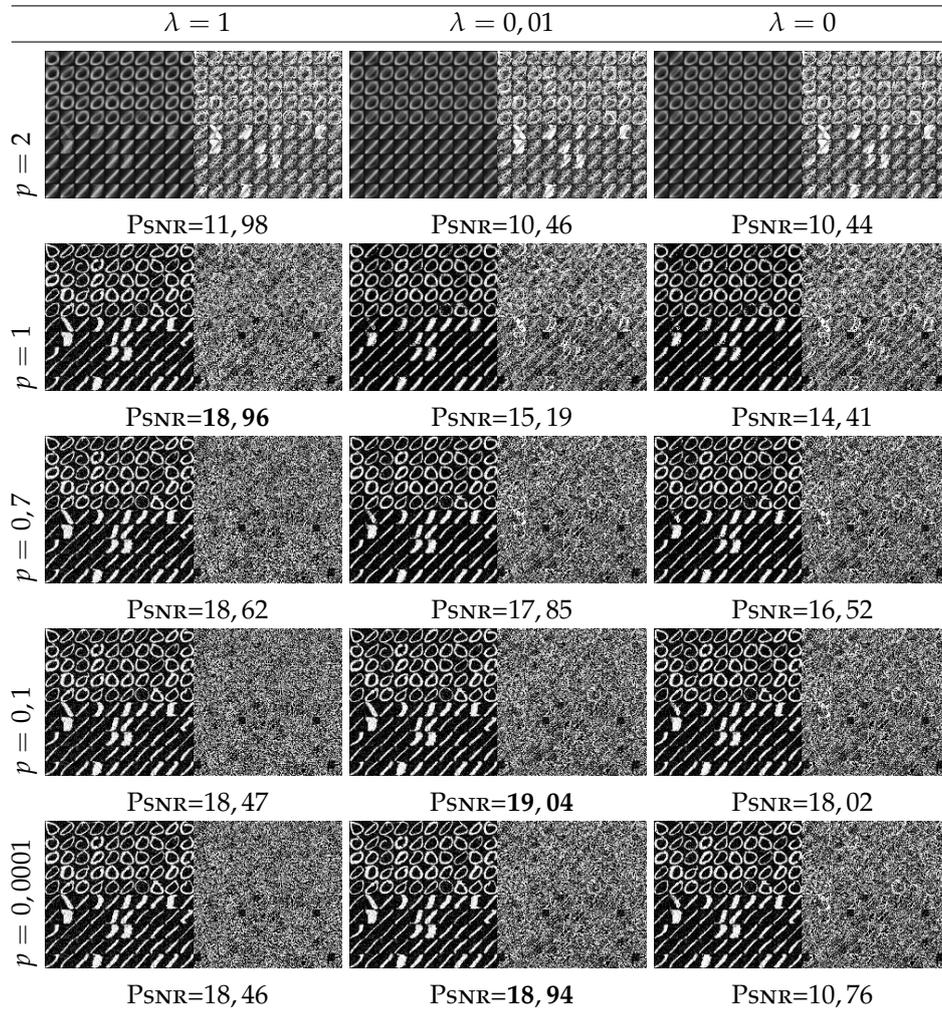


FIG. 3.9: Filtrage anisotrope d'une variété d'images (base Usps). Les données filtrées sont celles de la figure 3.1 page 57 avec différentes valeurs des paramètres p et λ . Chaque résultat présente le résultat du filtrage (la fonction finale f), la différence avec les données initiales ($|f^0 - f|$) et une valeur de PSNR.

Les figures 3.10 et 3.11 montrent les effets et l'évolution du filtrage sur les données pour différentes itérations n et différentes valeurs des paramètres p et λ .

La figure 3.10 présente des résultats pour une valeur de $\lambda = 1$ et la figure 3.11 ceux pour une valeur de $\lambda = 0$.

Nous pouvons constater que l'ensemble des résultats du filtrage par le modèle anisotrope sont différents de ceux obtenus avec le modèle isotrope hormis quand le paramètre $p = 2$ où les deux modèles coïncident. Comme pour le débruitage de la variété d'images présenté dans figure 3.9, le modèle anisotrope (pour des valeurs du paramètre $p \neq 2$) tend à préserver la structure initiale de la variété et filtre moins les données que le modèle isotrope.

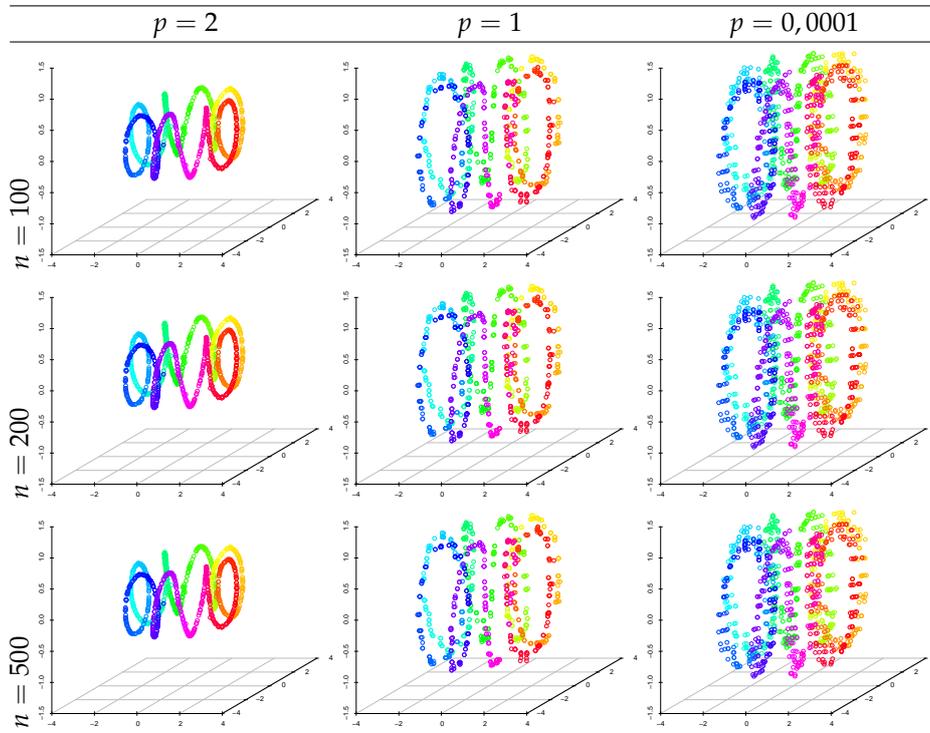


FIG. 3.10: Évolution (en fonction du nombre d'itérations n) du filtrage anisotrope de la variété avec différentes valeurs des paramètres p et $\lambda = 1$. Les données traitées correspondent à celles représentées par la figure 3.3 page 59.

Filtrage et simplification comme pré-traitement

Les figures 3.12 et 3.13 montrent les effets du filtrage du modèle anisotrope sur des bases de données.

Afin de pouvoir comparer les effets du filtrage avec le modèle isotrope, nous utilisons les mêmes données, celles représentant les bases de données « iris » et « wine » provenant du dépôt de l'UCI. Nous utilisons également les mêmes paramètres et les mêmes graphes que pour les résultats obtenus avec les figures 3.7 page 62 et 3.8 page 63.

Les figures 3.12 et 3.13 montrent les résultats du filtrage des bases « iris » et « wine » par le modèle de régularisation anisotrope pour différentes valeurs des paramètres p et λ .

Contrairement au modèle isotrope, l'effet de regroupement semble visuellement moins prononcé. Les points se regroupent moins dans l'espace des attributs. Nous pouvons constater que ces résultats sont similaires à ceux obtenus dans le cas du débruitage de la variété représentée par l'hélice toroïdale (figures 3.10 et 3.11), où le filtrage anisotrope tend à mieux préserver la topologie initiale des données initiales tout en les filtrant.

Comme pour le cas isotrope, le tableau 3.3 montre et compare des résultats de classification entre les données initiales et les données filtrées. La classification est

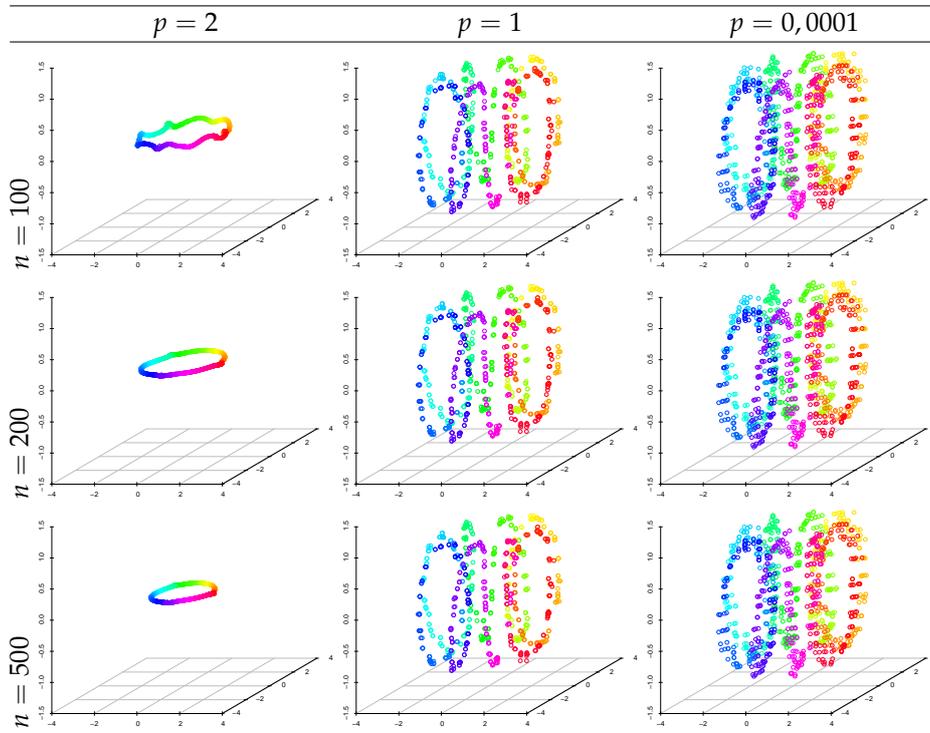


FIG. 3.11: Évolution (en fonction du nombre d'itérations n) du filtrage anisotrope de la variété avec différentes valeurs des paramètres p et $\lambda = 0$. Les données traitées correspondent à celles représentées par la figure 3.3 page 59.

également obtenue à partir de l'algorithme des k moyennes où le nombre d'attributs utilisé est spécifié dans la deuxième colonne de ce tableau. Les résultats affichés correspondent à la valeur maximale des taux de classification obtenus sur 200 tours de tests.

Pour la base « iris », nous pouvons constater que le bénéfice du filtrage est moins important que pour le cas isotrope. Cependant, les taux de classification (à partir des données filtrées) restent meilleurs (ou égaux) à ceux obtenus sans l'étape de filtrage. En ce qui concerne la base « wine », nous pouvons remarquer que le taux de classification est amélioré en utilisant l'ensemble des attributs de la base (à la deuxième ligne du tableau 3.3), contrairement au filtrage basé sur le modèle isotrope. Lorsque nous utilisons un nombre d'attributs réduit, les taux de classification sont meilleurs (ou égaux) avec les données filtrées que ceux obtenus sans l'étape de filtrage, mais restent moins élevés que les résultats obtenus avec l'ensemble des attributs. Les taux de classification sont proches, mais restent moins élevés que dans le cas isotrope.

3.2.9 Conclusion

Dans cette section, nous avons présenté les solutions d'une première classe de modèles variationnels définis sur des graphes. Afin d'approximer ces solutions,

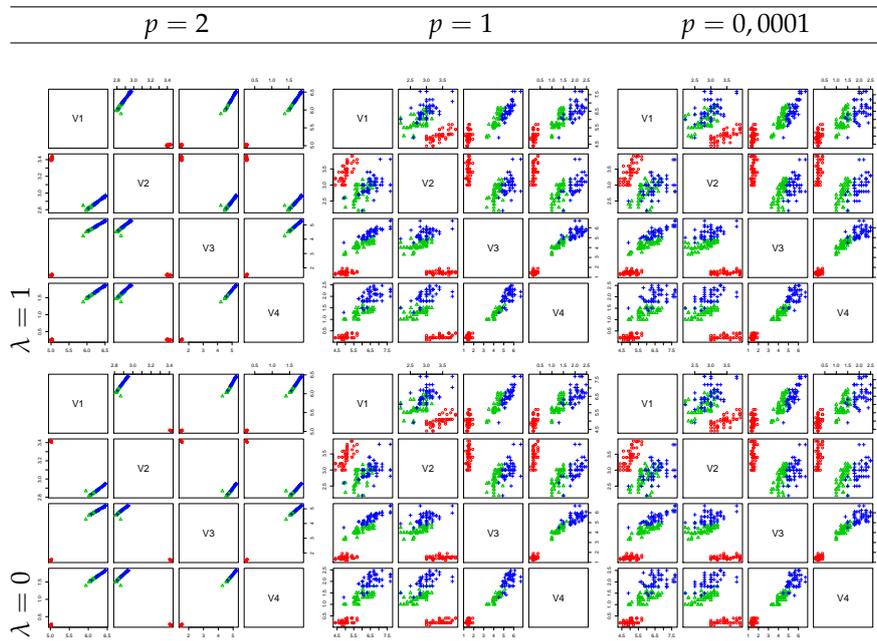


FIG. 3.12: Filtrage anisotrope de la base de données « iris » avec différentes valeurs des paramètres p et λ . La première ligne montre les résultats pour $\lambda = 1$ et la seconde pour $\lambda = 0$.

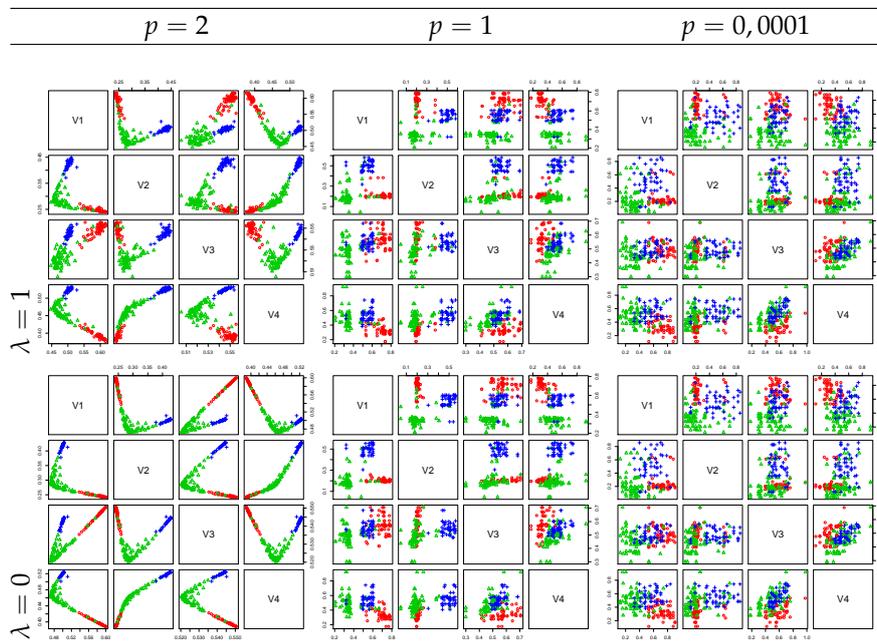


FIG. 3.13: Filtrage anisotrope de la base de données « wine » avec différentes valeurs des paramètres p et λ . La première ligne montre les résultats pour $\lambda = 1$ et la seconde pour $\lambda = 0$.

TAB. 3.3: Taux de classification de la base de données « iris » et « wine » pour différentes valeurs des paramètres p et λ (comparaison entre les données initiales et les données filtrées avec le modèle anisotrope). Les meilleurs taux sont présentés en gras. La classification est obtenue avec l'algorithme des k moyennes où le nombre d'attributs utilisés est spécifié dans la deuxième colonne.

Base	Attributs utilisés	Données initiales	Données filtrées			
			$p = 2$	$p = 1$	$p = 0,0001$	
Iris	4	89,3%	$\lambda = 1$	94,0%	90,7%	89,3%
			$\lambda = 0$	94,0%	90,7%	89,3%
wine	13	96,6%	$\lambda = 1$	94,4%	97,2%	97,2%
			$\lambda = 0$	94,4%	97,2%	97,2%
wine	4	83,7%	$\lambda = 1$	97,2%	94,4%	83,7%
			$\lambda = 0$	96,6%	94,9%	83,7%

nous avons proposé des algorithmes discrets. Nos algorithmes ne nécessitent pas d'étape de discrétisation contrairement aux modèles basés sur les équations aux dérivées partielles. Nos modèles permettent le traitement par régularisation de n'importe quel type de données arbitraires pouvant être représentées par un graphe.

Nous avons présenté deux cas particuliers de la classe de modèles variationnels : les modèles isotrope et anisotrope conduisant à deux familles de filtres linéaires et non linéaires. À partir de ces modèles, nous avons montré l'intérêt de ce type de traitement pour le filtrage et la simplification de données discrètes telles que des bases de données, des variétés d'images ou des nuages de points.

Nous avons également montré que les modèles isotrope et anisotrope se comportaient de manière différente. Chacun d'entre eux peut apporter une solution particulière à une application donnée.

Finalement, les deux modèles de régularisation peuvent s'exprimer dans un modèle variationnel général en utilisant la formule du p -Laplacien défini sur des graphes $\Delta_{w,p,q}^*$ par l'équation (2.124) page 30.

Dans la section suivante, nous montrerons comment les modèles isotrope et anisotrope peuvent être utilisés pour considérer des problèmes d'interpolation.

3.3 Problèmes d'interpolation basés sur la régularisation

3.3.1 Introduction

Dans cette section, nous abordons certains problèmes d'interpolation qui peuvent être exprimés et résolus à l'aide de nos modèles de régularisation définis sur des graphes. Dans le domaine du traitement des images, différents problèmes tels que la retouche des images, la super résolution, la colorisation ou encore la segmentation semi supervisée peuvent être interprétés comme étant des problèmes d'interpolation.

Le problème d'interpolation consiste à prédire des données manquantes (dans un ensemble) à partir de celles existantes. Le lecteur intéressé par plus de détails sur ces problèmes pourra se référer, par exemple, aux articles suivants [Boykov et Jolly, 2001, Levin *et al.*, 2004, Tschumperlé et Deriche, 2005, Grady, 2006, Peyré *et al.*, 2008, Roussos et Maragos, 2009] et à leurs références associées. Ces problèmes peuvent être modélisés à partir de notre formalisme basé sur des graphes.

Soient $G = (V, E, w)$ un graphe pondéré et $f^0 \in \mathcal{H}(V_0)$ une fonction initiale définie sur un ensemble de sommets V_0 tel que $V_0 \subset V$. Le problème d'interpolation consiste à calculer une fonction f définie sur $V \setminus V_0$ à partir de f^0 . À l'aide de la première classe de modèles variationnels discrets, ce problème se formalise par

$$\min_{f:V \rightarrow \mathcal{H}(V)} \mathcal{R}_{w,q,\phi}^*(f) + \sum_{u \in V} \frac{\lambda(u)}{2} (f(u) - f^0(u))^2 \quad (3.140)$$

où le terme de régularisation utilise l'opérateur de sommation et le gradient symétrique ∇_w tel que

$$\mathcal{R}_{w,q,\phi}^*(f) = \sum_{u \in V} \phi(\|(\nabla_w f)(u)\|_q) \quad (3.141)$$

avec $0 < q < +\infty$. La fonction $\lambda : V \rightarrow \mathbb{R}^+$ est définie, pour un sommet $u \in V$, par

$$\lambda(u) = \begin{cases} \lambda_u & \text{si } u \in V_0 \\ 0 & \text{sinon} \end{cases} \quad (3.142)$$

Dans la suite de cette section, nous présentons deux exemples de problèmes d'interpolation : la colorisation des images monochromes et la classification semi supervisée. Dans le contexte du traitement des images, nous montrons les bénéfices des configurations non locales avec des patches par rapport aux schémas locaux. Nous illustrons également la généralité de nos approches par l'utilisation de différentes topologies de graphes et l'application à différentes données.

Nous montrons également que nos modèles de régularisation forment un ensemble d'outils adaptatifs (filtrage, simplification et classification) pouvant être utilisés dans le cadre du traitement des images cellulaires.

3.3.2 Colorisation d'images

La colorisation consiste à coloriser (colorier, ajouter des couleurs) à des images monochromes. Dans la majorité des cas, ce traitement est réalisé manuellement

par des experts du domaine rendant la tâche répétitive et fastidieuse. Récemment, plusieurs méthodes de colorisation semi automatiques ont été proposées afin de faciliter ce travail [Levin *et al.*, 2004, Yatziv et Sapiro, 2006]. Elles sont basées sur des marqueurs initiaux placés par l'utilisateur et utilisent des processus de diffusion pour réaliser la colorisation. Ces processus de diffusion ne considèrent généralement, que les interactions locales entre les pixels ne permettant pas de rendre compte des structures plus complexes contenues dans les images. Afin de palier ce problème, nous proposons d'utiliser notre formalisme basé sur des graphes et d'incorporer les schémas non locaux dans le processus de colorisation des images.

Le problème de colorisation peut se formaliser à l'aide de la minimisation (3.140) et du modèle p -Tv isotrope présenté dans les sections précédentes. Nous avons alors

$$\min_{f:V \rightarrow \mathcal{H}(V)} \frac{1}{p} \sum_{u \in V} \|(\nabla_w f)(u)\|_2^p + \sum_{u \in V} \frac{\lambda(u)}{2} (f(u) - f^0(u))^2 \quad (3.143)$$

où f^0 correspond à la fonction initiale. Cette dernière utilise l'image monochrome initiale $g^0 : V \rightarrow \mathbb{R}$ à coloriser et des marqueurs couleurs initiaux $m : V_0 \subset V \rightarrow \mathbb{R}^3$ (avec V_0 , l'ensemble des sommets initialement marqués). La fonction m est définie comme une application des sommets V vers un vecteur de couleurs défini dans l'espace RGB . Pour un sommet $u \in V$, $m(u) = (m_i(u))_{i=\{R,G,B\}}^T$ où $m_i : V \rightarrow \mathbb{R}$ représente une des trois composantes de l'espace couleur.

À partir des marqueurs initiaux, la fonction initiale f^0 est définie comme une application des sommets vers un vecteur de chrominance telle que

$$f^0(u) = \begin{cases} \left(\begin{matrix} m_i(u) \\ g^0(u) \end{matrix} \right)_{i=\{R,G,B\}}^T & \text{si } u \in V_0 \\ (0, 0, 0)^T & \text{si } u \in V \setminus V_0 \end{cases} . \quad (3.144)$$

La minimisation du problème lié à la colorisation s'obtient de la même manière que pour le modèle p -Tv isotrope. La solution peut être approximée en utilisant l'algorithme discret suivant. Pour tout $u \in V$,

$$\begin{cases} f_i^{n+1}(u) &= \frac{\lambda(u)f_i^0(u) + \sum_{v \sim u} w_{uv} (\|(\nabla_w f_i^n)(v)\|_2^{p-2} + \|(\nabla_w f_i^n)(u)\|_2^{p-2}) f_i^n(v)}{\lambda(u) + \sum_{v \sim u} w_{uv} (\|(\nabla_w f_i^n)(v)\|_2^{p-2} + \|(\nabla_w f_i^n)(u)\|_2^{p-2})} \\ f^{(0)}(u) &= f^0(u) \end{cases} . \quad (3.145)$$

À la convergence de l'algorithme, nous obtenons la fonction interpolée $f : V \rightarrow \mathbb{R}^3$. Les couleurs finales à appliquer à l'image monochrome sont obtenues par

$$g^0(u) \cdot (f_1^{n \rightarrow \infty}(u), f_2^{n \rightarrow \infty}(u), f_3^{n \rightarrow \infty}(u)), \quad \forall u \in V . \quad (3.146)$$

La figure 3.14 montre un exemple d'application de notre approche de la colorisation. Notre modèle intègre les configurations locales et non locales dans une même formulation. Cette figure montre une comparaison entre ces deux types de schémas. La figure 3.14 montre deux exemples où, de gauche à droite, sont présentées les images initiales à coloriser (provenant d'images couleurs converties en

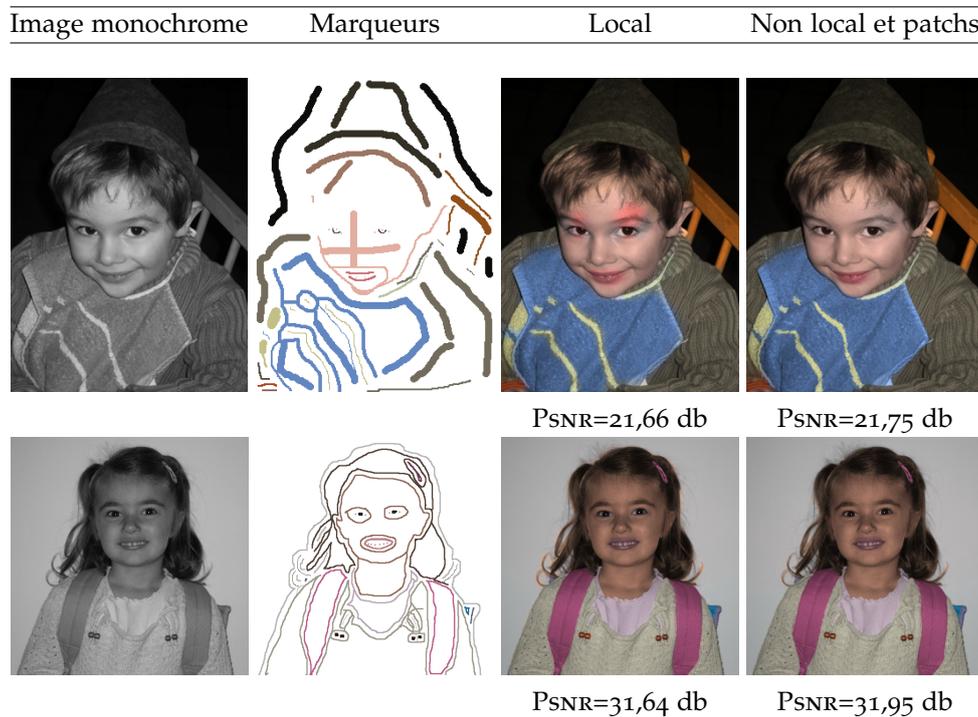


FIG. 3.14: Exemples de colorisation d'images à partir du modèle de régularisation p -TV isotrope. De gauche à droite et à chaque ligne : l'image monochrome à coloriser (provenant d'une image couleur convertie en niveaux de gris), les marqueurs couleurs initiaux, les résultats obtenus avec des schémas locaux et non locaux. Pour chaque résultat une valeur de PSNR est donnée (valeur calculée sur l'image initiale couleur avant la conversion en niveaux de gris).

niveaux de gris), les marqueurs couleurs initiaux et les résultats obtenus avec des schémas locaux et non locaux. Les résultats utilisant les configurations locales sont obtenus avec un graphe grille en 8-voisinage pondéré par la fonction de poids g_4 avec $\mathcal{F} = g^0$. Pour les configurations non locales, le graphe associé est un graphe grille G_5 (99-voisinage) pondéré par la fonction de poids g_4 avec comme vecteur d'attributs $\mathcal{F}_2(g^0, \cdot)$ (des patches de taille 5×5). L'ensemble des résultats est obtenu avec une valeur de $p = 1$ pour l'algorithme (3.145) et une valeur de $\lambda_u = 0,01$ pour la définition de la fonction (3.142). Cette valeur faible du terme d'attache aux données permet de garantir que les marqueurs initiaux puissent être modifiés durant le processus de colorisation afin d'obtenir un résultat final plus lisse et homogène avec le reste de l'image.

Nous pouvons voir que, avec les schémas non locaux, nous avons des valeurs de PSNR plus élevées (ces valeurs ont été calculées à partir des images initiales en couleurs avant leur conversion en niveaux de gris) et des résultats visuellement plus satisfaisants comparativement aux configurations locales. Dans le premier exemple, nous pouvons voir qu'avec les schémas locaux, les couleurs dans certaines zones ont été trop diffusées (notamment autour des yeux). Avec les configurations non locales avec des patches, la colorisation de ces zones est satisfaisante grâce aux patches qui

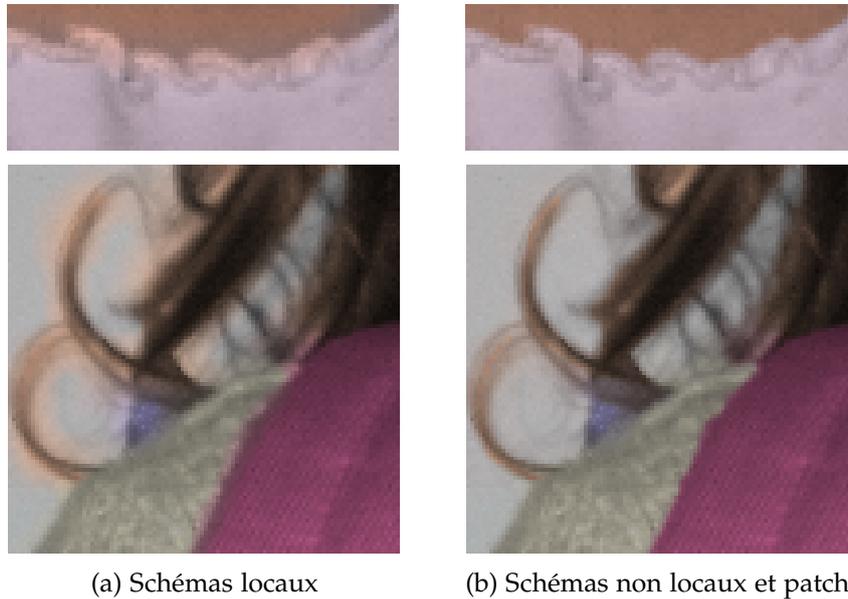


FIG. 3.15: Zoom sur certains détails de la colorisation de l'image de la fille. Comparaison avec les schémas locaux (a) et non locaux avec des patches (b). Les configurations non locales permettent de mieux stopper la diffusion des couleurs : par exemple au niveau des cheveux ou du col.

permettent de mieux capturer les structures fines et similaires contenues dans les images.

Afin d'apprécier visuellement les différences entre la colorisation locale et non locale avec des patches, la figure 3.15 montre des zooms de certains détails du second exemple. Nous pouvons remarquer que même avec des marqueurs initiaux précisément placés, les schémas locaux ont tendance à diffuser davantage la couleur (au niveau du col et des cheveux). Avec les configurations non locales, nous n'obtenons pas ces effets rendant la colorisation plus satisfaisante.

Dans cette section, nous avons présenté un premier problème d'interpolation formulé à partir de nos modèles de régularisation basés sur les graphes : la colorisation d'image monochrome. Nous avons montré que notre modèle pouvait intégrer les schémas non locaux avec des patches dans le processus de colorisation. Ces dernières configurations permettent d'obtenir de meilleurs résultats qualitatifs et quantitatifs par rapport aux schémas locaux.

Dans la section suivante, nous nous intéressons à un autre problème d'interpolation : la segmentation et la classification semi supervisée d'images et de données.

3.3.3 Segmentation et classification semi supervisées

La classification semi supervisée consiste à prédire les labels (marqueurs) manquants à partir d'un ensemble de données initialement et partiellement labélisé. Nous proposons ici de considérer ce problème d'interpolation à partir de nos modèles de régularisation définis sur des graphes : en particulier, d'utiliser les modèles variationnels p -Tv isotrope et anisotrope.

Comme nous l'avons mentionné dans les sections précédentes, nos modèles de régularisation sont reliés à des méthodes de diffusion basées sur le p -Laplacien isotrope et anisotrope définies sur des graphes. Des travaux récents ont montré l'efficacité de ce type d'approche (notamment le p -Laplacien isotrope) pour résoudre les problèmes de classification et d'apprentissage de données ou de variétés [Zhou et Schölkopf, 2004, Belkin *et al.*, 2006]. Le lecteur intéressé par plus de détails sur ces approches peut se référer à [Zhu, 2008] pour une revue complète des différentes méthodes existantes.

Dans le domaine du traitement des images, la classification (segmentation) semi supervisée consiste à extraire des objets d'intérêts à partir de labels initiaux placés par l'utilisateur. La segmentation est alors réalisée par propagation de ces marqueurs. Différentes méthodes ont été utilisées pour résoudre ce problème dans la littérature. Nous pouvons, par exemple, citer les méthodes de coupe de graphes, les méthodes variationnelles incluant des configurations non locales, celles utilisant des méthodes d'optimisation combinatoire ou la théorie spectrale. Le lecteur intéressé par ces méthodes pourra se référer aux articles suivants [Boykov et Jolly, 2001, Yu et Shi, 2004, Grady, 2006, Gilboa et Osher, 2007a, Sinop et Grady, 2007, Szlam *et al.*, 2008, Couprie *et al.*, 2009, Houhou *et al.*, 2009] et à leurs références associées.

Approches basées sur les modèles p -Tv isotrope et anisotrope

Le problème d'interpolation que représente la classification semi supervisée peut se formaliser à l'aide des graphes de la manière suivante.

Soit $G = (V, E, w)$ un graphe pondéré qui représente des données à traiter. La classification semi supervisée des sommets V consiste à regrouper et à classer cet ensemble en c classes, où le nombre c est connu a priori. L'ensemble des sommets V est composé initialement de sommets labélisés et non labélisés. L'objectif est alors d'estimer les sommets non marqués à partir de ceux qui le sont.

Soit C_i , avec $i = 1, \dots, c$, l'ensemble des sommets de V qui sont labélisés et appartenant à la $i^{\text{ème}}$ classe. Soit $V_0 = \{C_i\}_{i=1, \dots, c}$ l'ensemble des sommets qui sont initialement marqués et $V \setminus V_0$ l'ensemble des sommets qui ne le sont pas. Chaque sommet u de V est alors décrit par un vecteur de labels $f^0 : V \subset \mathbb{R}^m \rightarrow \mathbb{R}^c$ tel que

$$f^0(u) = (f_i^0(u))_{i=1, \dots, c}^T \quad (3.147)$$

et initialisé de la manière suivante

$$f_i^0(u) = \begin{cases} 1 & \text{si } u \in C_i, \text{ avec } i = 1, \dots, c \\ 0 & \text{sinon} \\ 0 & \forall u \in V \setminus V_0 \end{cases} \quad (3.148)$$

Pour résoudre ce problème d'interpolation, nous utilisons la minimisation formulée par l'équation (3.140) et nos modèles variationnels : les modèles p -Tv isotrope

$$\min_{f \in \mathcal{H}(V)} \frac{1}{p} \sum_{u \in V} \|(\nabla_w f)(u)\|_2^p + \sum_{u \in V} \frac{\lambda(u)}{2} (f(u) - f^0(u))^2 \quad (3.149)$$

et anisotrope

$$\min_{f \in \mathcal{H}(V)} \frac{1}{2p} \sum_{u \in V} \|(\nabla_w f)(u)\|_p^p + \sum_{u \in V} \frac{\lambda(u)}{2} (f(u) - f^0(u))^2 \quad (3.150)$$

avec $0 < p < +\infty$.

En utilisant les algorithmes discrets approximant les solutions de ces deux modèles, nous obtenons, dans le cas de la classification semi supervisée, les algorithmes suivants. Pour tout $u \in V$ et $i = \{1, \dots, c\}$, le modèle isotrope donne

$$\begin{cases} f_i^{n+1}(u) = \frac{\lambda(u)f_i^0(u) + \sum_{v \sim u} w_{uv} (\|\nabla_w f_i^n(v)\|_2^{p-2} + \|\nabla_w f_i^n(u)\|_2^{p-2}) f_i^n(v)}{\lambda(u) + \sum_{v \sim u} w_{uv} (\|\nabla_w f_i^n(v)\|_2^{p-2} + \|\nabla_w f_i^n(u)\|_2^{p-2})} \\ f_i^{(0)}(u) = f_i^0(u) \end{cases} \quad (3.151)$$

et le modèle anisotrope donne

$$\begin{cases} f_i^{n+1}(u) = \frac{\lambda(u)f_i^0(u) + \sum_{v \sim u} w_{uv}^{p/2} (f_i^n(u) - f_i^n(v)) |f_i^n(u) - f_i^n(v)|^{p-2} f_i^n(v)}{\lambda(u) + \sum_{v \sim u} w_{uv}^{p/2} (f_i^n(u) - f_i^n(v)) |f_i^n(u) - f_i^n(v)|^{p-2}} \\ f_i^{(0)}(u) = f_i^0(u) \end{cases} \quad (3.152)$$

En partant des sommets labélisés de V_0 , la classification semi supervisée (isotrope et anisotrope) consiste à calculer c régularisations. À convergence de ces processus, nous affectons à un sommet u la classe i la plus plausible en calculant une probabilité d'appartenance

$$\operatorname{argmax}_i \left(\frac{f_i^{n \rightarrow \infty}(u)}{\sum_i f_i^{n \rightarrow \infty}(u)} \right) \quad \text{avec } i = \{1, \dots, c\} \quad (3.153)$$

Remarque. Nous pouvons remarquer l'importance de la fonction de poids et de la topologie du graphe dans ce type de problème. En effet, ce sont eux qui permettent d'intégrer les informations provenant des données initiales dans le processus de classification ou de segmentation étant donné que la fonction f estimée est la fonction label.

La figure 3.16 illustre un exemple de classification semi supervisée sur un ensemble de données par propagation de labels. Elle montre également l'évolution de la labélisation en fonction du nombre d'itérations n . Le graphe associé (figure 3.16b) aux données initiales est un 20-NNG pondéré par la fonction de poids g_4 .

Expérimentations

Nous présentons ici une série d'expérimentations montrant l'application de nos modèles de classification semi supervisée définis sur des graphes.

Nous nous intéressons tout d'abord à la segmentation des images avec des configurations non locales avec des patches et basée sur des graphes grilles. Ensuite, nous présentons des expérimentations sur des graphes de régions construits à partir d'une partition d'une image. Nous verrons que ce type de graphes associés aux images offrent des avantages tels que la segmentation des objets non connexes, un nombre de marqueurs réduit ou encore la réduction du nombre de sommets entraînant une diminution de la complexité des algorithmes. Finalement, nous présentons des applications de la classification semi supervisée pour traiter des données discrètes telles que des bases de données ou des variétés.

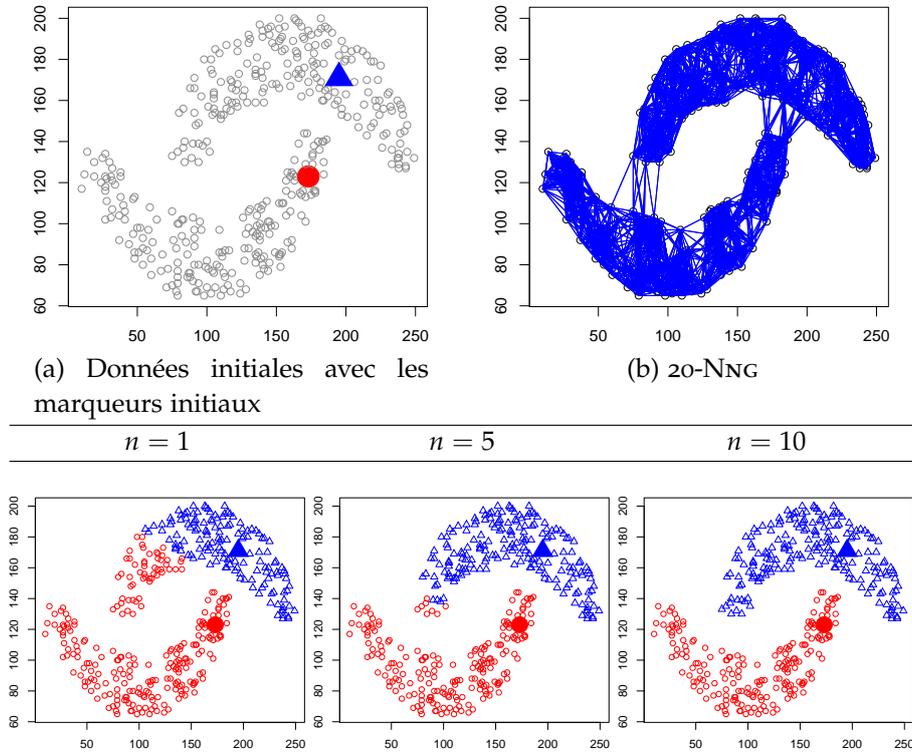


FIG. 3.16: Exemple de propagation de labels sur un nuage de points pour la classification semi supervisée de données. (a) : données initiales avec les marqueurs initiaux et (b) : le graphe associé. À la dernière ligne : évolution de la classification en fonction du nombre d'itérations n .

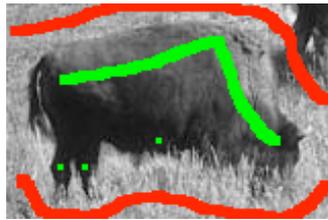


FIG. 3.17: Image initiale avec marqueurs initiaux servant d'image de test pour la segmentation semi supervisée non locale isotrope et anisotrope présentée dans la figure 3.18.

Nos modèles isotrope et anisotrope de la classification semi supervisée sont définis avec des graphes et unifient configurations locales et non locales dans le contexte du traitement des images. Nous pouvons remarquer que la segmentation non locale avec des patches est peu étudiée et utilisée dans la littérature. Nous pouvons citer quelques travaux récents définis dans le domaine continu [Gilboa et Osher, 2007a, Houhou *et al.*, 2009].

La figure 3.18 illustre ce type de segmentation des images et compare les modèles isotrope et anisotrope avec différents graphes et différentes valeurs du para-

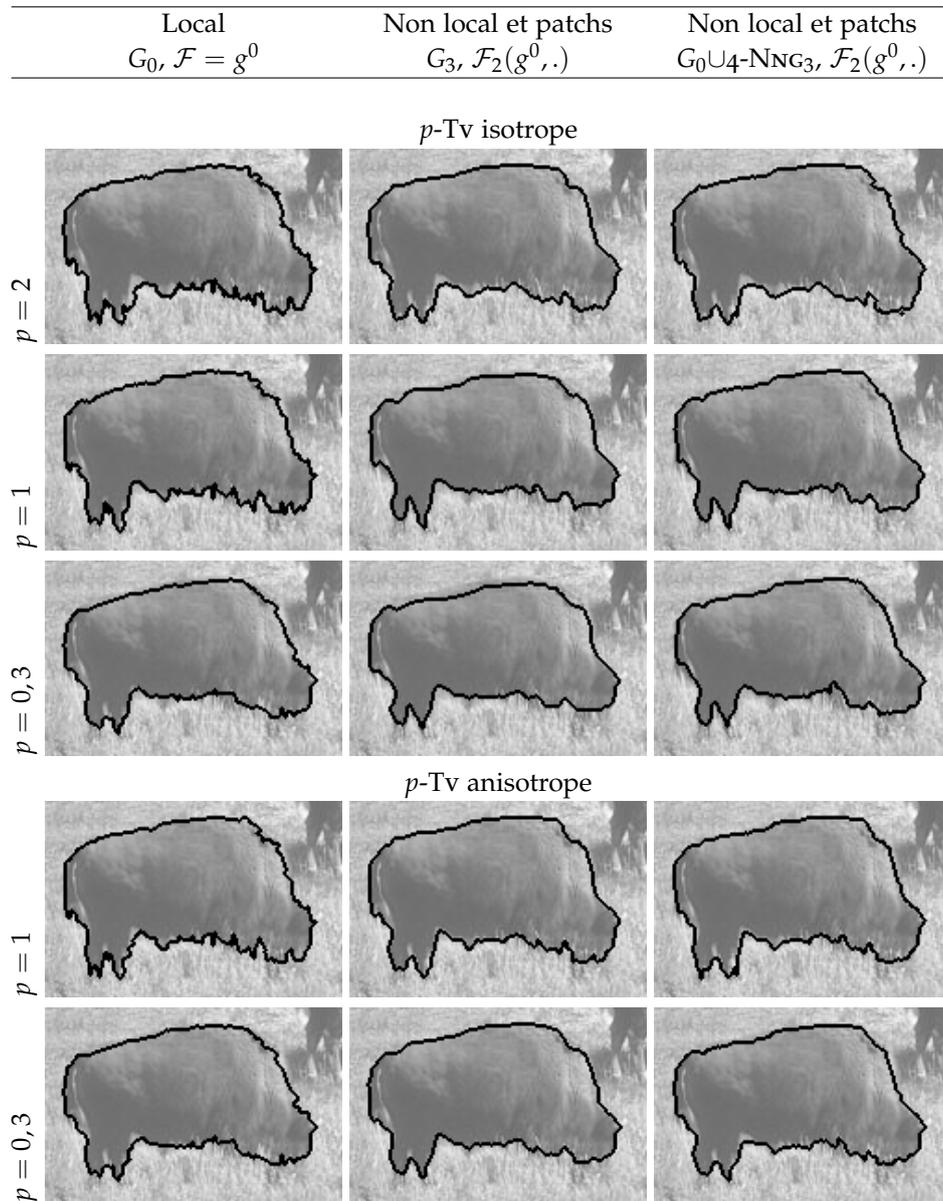


FIG. 3.18: Segmentation non locale d'images. Les graphes associés sont construits à partir de l'image initiale $g^0 : V \rightarrow \mathbb{R}$ de la figure 3.17. Les segmentations sont obtenues avec les marqueurs initiaux présentés dans la figure 3.17. Chaque résultat correspond à une structure de graphe différente et à une valeur du paramètre p différent. Les contours des régions obtenus sont superposés en noir sur l'image initiale (volontairement blanchie afin d'avoir une meilleure visualisation).

mètre p (dans le cas où le paramètre $p = 2$, les modèles isotrope et anisotrope sont équivalents).

Les résultats de la figure 3.18 sont obtenus avec différentes topologies de graphes calculées à partir de l'image initiale (figure 3.17) définie comme une fonction g^0 :

$V \rightarrow \mathbb{R}$. Les résultats de la figure 3.18 montrent les segmentations finales (obtenues à partir des marqueurs initiaux de la figure 3.17) où les contours des régions sont superposés à l'image initiale.

La première colonne de la figure 3.18 présente les résultats des segmentations obtenues à partir d'un graphe grille en 4-voisinage pondéré par la fonction de poids g_4 associée aux vecteurs $\mathcal{F}(g^0, \cdot) = g^0(\cdot)$. Cette configuration correspond aux schémas locaux usuels employés pour ce type de méthode de segmentation. Nous pouvons remarquer que dans les cas isotrope et anisotrope, plus la valeur du paramètre p diminue et plus le contour final se rapproche de l'objet désiré. L'aspect du contour final reste néanmoins peu lisse.

La seconde colonne de la figure 3.18 présente les résultats des segmentations obtenues à partir d'une configuration non locale avec des patches. Il s'agit d'un graphe grille en 49-voisinage (G_3) pondéré par la fonction de poids g_4 associée à des vecteurs d'attributs $\mathcal{F}_2(g^0, \cdot)$ définis par des patches de taille 5×5 . Comme pour le cas de la configuration locale, plus la valeur du paramètre p diminue et plus le contour final tend à se rapprocher de l'objet désiré. Contrairement au cas local, l'utilisation d'une grande fenêtre de recherche (ici 7×7) associée à des patches comme vecteur d'attributs permet d'obtenir des contours plus lisses et plus réguliers grâce à la capacité des schémas non locaux à mieux capturer les similarités et à lisser implicitement les informations contenues dans les images. Avec cette topologie de graphe, nous pouvons constater que le cas isotrope tend à légèrement sous segmenter l'image alors que le modèle anisotrope permet au contour de mieux correspondre à l'objet.

Finalement, la dernière colonne de la figure 3.18 présente également des résultats de segmentation obtenus à partir d'une configuration non locale avec des patches. Un des inconvénients des méthodes non locales est la grande complexité des algorithmes résultant de la connexité du graphe, et plus particulièrement du degré élevé de chaque sommet $u \in V$. Afin de palier ce problème, nous proposons d'utiliser une autre topologie de graphe permettant de réduire le nombre de voisins de chaque sommet. Les résultats de la dernière colonne de la figure 3.18 sont obtenus à partir d'un graphe grille en 4-voisinage G_0 couplé avec un graphe des 4-NNC₃ et pondéré par la fonction de poids g_4 . Chaque sommet est connecté à ses quatre voisins spatialement connexes et à ses quatre plus proches voisins dans une fenêtre de recherche de taille 7×7 . Le calcul des distances permettant de pondérer le graphe et de déterminer les plus proches voisins est basé sur des vecteurs d'attributs $\mathcal{F}_2(g^0, \cdot)$. Les résultats de segmentation montrent un comportement similaire à ceux obtenus avec le graphe G_3 , mais avec cette configuration, la complexité des algorithmes de traitement est diminuée grâce à la réduction du nombre de voisins de chaque sommet du graphe.

La plupart des méthodes de segmentation semi supervisées sont basées sur les pixels de l'image. Si nous considérons que les pixels de l'image ne sont pas les seuls éléments représentatifs, alors nous pouvons utiliser des structures de plus haut niveau telles que les régions d'une partition de l'image également appelées « super pixels » [Ren et Malik, 2003].

La figure 3.19 montre des exemples de segmentation semi supervisée basée sur des graphes de régions. La première colonne de la figure 3.19 montre les images initiales avec les marqueurs initiaux servant à la segmentation.

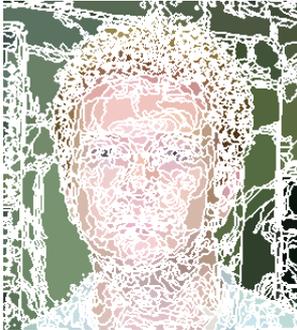
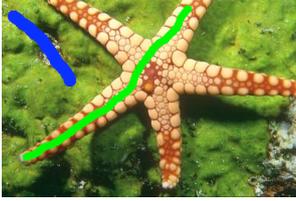
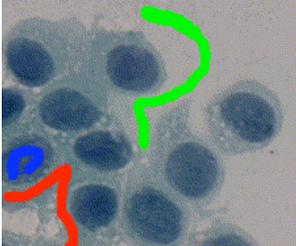
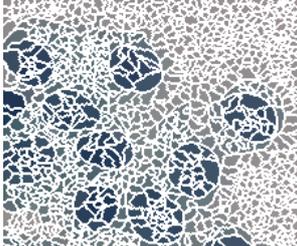
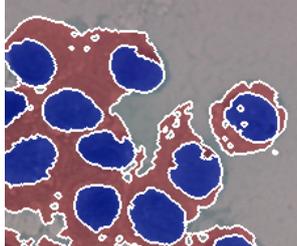
Images initiales et marqueurs	Partitions	Résultats
		
$\text{card}(V) = 85\,593$	$\text{card}(V) = 2\,370$ (97,2%)	
		
$\text{card}(V) = 154\,401$	$\text{card}(V) = 3\,174$ (97,9%)	
		
$\text{card}(V) = 44\,748$	$\text{card}(V) = 1\,148$ (97,4%)	

FIG. 3.19: Segmentation semi supervisée avec des graphes de régions. Les résultats sont obtenus avec des graphes complets pondérés et construits à partir des partitions (*deuxième colonne*). Les pourcentages entre parenthèses représentent le pourcentage de réduction en terme de sommets du graphe de régions par rapport au nombre de sommets d'un graphe qui aurait été construit à partir des pixels de l'image.

À partir des images initiales, des partitions sont calculées. Différentes méthodes efficaces permettant d'obtenir une partition sont disponibles dans la littérature [Meila et Shi, 2000, Meyer, 2001, Arbeláez et Cohen, 2004, Cousty *et al.*, 2009a]. Des exemples de partitions obtenues avec la méthode de [Arbeláez et Cohen, 2004] sont présentés dans la seconde colonne de la figure 3.19 où chaque région de la partition est remplacée par la couleur moyenne des pixels qui la composent et son contour est superposé en blanc. Cette colonne présente également les pourcentages de réduction (en terme de sommets) entre un graphe construit à partir des régions des partitions et un graphe qui aurait été construit à partir des pixels de l'image. Nous

pouvons remarquer que cette réduction est de l'ordre de 97% sur les trois exemples présentés.

La dernière colonne de la figure 3.19 montre les résultats de segmentation obtenus à partir des marqueurs de la première colonne. Pour obtenir ces résultats, des graphes complets ont été construits à partir des régions des partitions et la fonction de poids associée (g_4) utilise les moyennes couleurs (dans l'espace RGB) des régions comme attributs des sommets du graphe (chaque sommet étant associé à une région de la partition). L'utilisation d'une telle structure n'est pas envisageable si les sommets du graphe sont associés aux pixels de l'image. Par contre, dans le cas de l'utilisation des régions d'une partition, cette topologie permet d'obtenir les propriétés suivantes.

- L'utilisation d'un graphe complet pondéré permet d'intégrer toutes les informations de l'image (approche complètement non locale), ce qui a pour conséquence d'avoir une convergence rapide de l'algorithme en peu d'itérations.
- Peu de marqueurs initiaux sont nécessaires pour obtenir une segmentation satisfaisante.
- La possibilité de segmenter des objets non spatialement connexes ou adjacents. Par exemple, dans l'image des cellules, seulement une cellule a été marquée et la méthode de segmentation a réussi à délimiter toutes les autres.
- Finalement, en considérant une version simplifiée de l'image, le nombre de sommets du graphe se trouve réduit permettant de diminuer la complexité des algorithmes et les temps de calcul.

La figure 3.20 montre un exemple de classification semi supervisée appliquée à la recherche par le contenu dans des bases de données. Les données initiales sont celles de l'USPS. La base de test labélisée comporte 800 instances de chaque chiffre allant de « 0 » à « 9 », tirés de manière aléatoire, représentant un total de 8000 chiffres. Chaque chiffre est labélisé avec sa classe correspondante. La figure 3.20a montre une partie de la base de test. Les données non labélisées (figures 3.20b, 3.20c, 3.20d et 3.20e) sont sélectionnées aléatoirement parmi les chiffres allant de « 0 » à « 3 » des données restantes non utilisées pour constituer la base de test.

La classification semi supervisée appliquée à la recherche par le contenu consiste alors à estimer et à ordonner les instances de la base les plus similaires par rapport aux données non marquées initiales. Les figures 3.20f, 3.20g, 3.20h et 3.20i montrent les 50 résultats les plus proches obtenus à partir de la requête correspondante. Le graphe utilisé ici est un graphe complet où chaque sommet correspond à une image et est décrit par un vecteur d'attributs de $\mathbb{R}^{16 \times 16}$ correspondant aux niveaux de gris de l'image qui lui est associée. Nous pouvons constater la pertinence des réponses obtenues pour chacune des requêtes test.

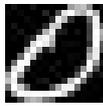
3.3.4 Outils pour le traitement d'images cellulaires

Dans cette section, nous montrons que nos modèles de régularisation forment un ensemble d'outils adaptatifs (filtrage, simplification et classification) pouvant être utilisés dans le cadre du traitement des images cellulaires.

L'objectif des traitements présentés ici consiste à extraire des objets tissulaires ou cellulaires. Cette extraction a pour but de faciliter le travail d'analyse de médecins anatomo-pathologistes. Les sous domaines médicaux concernés sont la cytologie



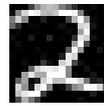
(a) Extrait de la base de test initiale labellisée



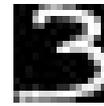
(b)



(c)



(d)



(e)



(f) résultats à partir de (b)



(g) résultats à partir de (c)



(h) résultats à partir de (d)



(i) résultats à partir de (e)

FIG. 3.20: Exemple d'application de la classification semi supervisée pour la recherche par le contenu. Les résultats présentés dans les figures (f), (g), (h) et (i) correspondent au 50 premiers exemples les plus proches en accord avec les requêtes initiales correspondantes (figure (b), (c), (d) et (e)).

et l'histologie pathologiques. Le traitement et la segmentation de telles images est un problème complexe du fait de la grande variété de contenus. Néanmoins, différentes approches existent pour traiter et segmenter des images médicales dont, par exemple, celles basées sur : l'analyse des histogrammes, la classification de pixels, les opérations morphologiques ou bien les méthodes basées sur les équations aux dérivées partielles. Le lecteur intéressé par plus de détails pourra se référer par exemple aux articles suivants [Malladi et Sethian, 1996, Schüpp *et al.*, 2001, Lézoray et Cardot, 2002, Lézoray *et al.*, 2009b] et aux références associées.

Dans cette section, nous abordons le problème de la segmentation des images microscopiques en utilisant nos modèles définis sur des graphes. Nous ne propo-

sons pas une méthode directe permettant de segmenter une classe d'images particulières, mais plutôt un ensemble d'outils et de schémas flexibles permettant de s'adapter à un problème donné. Ces outils sont basés sur les filtres et les schémas de segmentation semi supervisée proposés dans les sections 3.2 et 3.3.

Des schémas de segmentation automatique et semi automatique pour des images de cytologie et d'histologie seront présentés ainsi qu'une comparaison avec des approches proposées dans la littérature [Lézoray et Cardot, 2002, Dumont *et al.*, 2007].

Images cellulaires

De manière simple, nous pouvons dire que le domaine de pathologie se divise en deux parties : la cytologie et l'histologie. Dans ces deux parties, l'imagerie microscopique joue un rôle important car les différents prélèvements (issus de fluides ou de tissus humains) sont déposés sur une lame afin d'être analysés et diagnostiqués par un expert à l'aide d'un microscope.

- La cytologie consiste à étudier la morphologie des cellules à travers les modifications des noyaux ou des cytoplasmes contenues dans des prélèvements de fluides humains. Ces modifications peuvent s'exprimer en terme de taille, couleur, texture ou forme. Les figures 3.21a et 3.21b montrent des exemples d'images de cytologie.
- L'histologie consiste à étudier l'organisation et l'arrangement des cellules (provenant de prélèvements tissulaires humains) regroupées en amas et formant des structures complexes. Les figures 3.21c et 3.21d montrent des exemples d'images d'histologie.

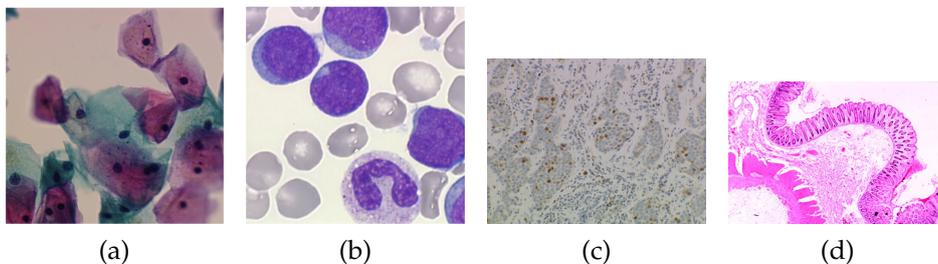


FIG. 3.21: Exemples d'images de cytologie et d'histologie. (a) : gynécologie, (b) : hématologie, (c) : tissu mammaire et (d) : colon.

Du fait du grand nombre d'objets cellulaires pouvant être contenus dans les lames, l'analyse visuelle par l'expert est une tâche fastidieuse et consommatrice en terme de temps. Nous pouvons ainsi voir l'intérêt du traitement des images qui peut permettre d'apporter une aide aux médecins par l'extraction et l'analyse automatique ou semi automatique des différentes structures cellulaires ou tissulaires.

Extraction d'objets cellulaires

Le traitement des images cellulaires doit respecter au mieux la manière dont l'expert analyse. Pour cela, les images sont acquises en couleur et l'extraction consiste

à séparer le fond du reste de l'image considérée. Le reste représente les objets à extraire (noyaux, cytoplasme ou tissus).

En terme de traitement des images, l'extraction des objets contenus dans ces images peut se décomposer en deux niveaux : le niveau cellulaire et le niveau tissulaire. Au niveau cellulaire, nous pouvons avoir des cellules isolées, qui se touchent ou qui se chevauchent. Au niveau tissulaire, nous pouvons trouver des groupes de cellules ou des regroupements (groupes de groupes). Cette hiérarchie peut se formaliser de la manière suivante : cellules \subset groupes \subset regroupements. Nous pouvons voir que la structure de graphe est adaptée à la représentation de cette hiérarchie où les sommets peuvent aisément représenter chaque niveau. Ainsi, nos modèles basés sur des graphes permettent d'unifier dans une même formulation les différents traitements applicables aux différents niveaux de la représentation des images.

Dans la suite, nous proposons d'utiliser nos modèles de régularisation comme un ensemble d'outils adaptatifs (filtrage, simplification et classification) permettant de traiter et de segmenter ce type d'images. Ces outils peuvent être combinés entre eux pour former des schémas de traitement et de segmentation automatiques ou semi automatiques. Le filtrage, la simplification et la segmentation sont basées sur les familles de filtres (section 3.2) et la classification semi supervisée (section 3.3) décrites dans les sections précédentes.

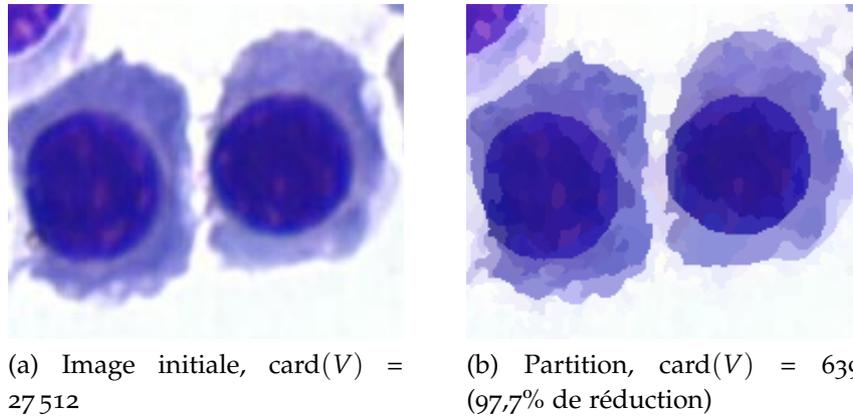
Comme nous l'avons dit dans la section 3.3, les modèles de segmentation d'images semi supervisée basés sur la propagation de labels utilisent habituellement une représentation basée sur les pixels pour effectuer la classification. Cette représentation devient inefficace, en terme de temps de calcul, lorsque les images sont de grande taille à cause de la grande quantité de données à traiter. Pour palier ce problème, nous proposons d'utiliser la méthode de segmentation basée sur des graphes de régions décrite dans la section 3.3 et faisant intervenir une version simplifiée de l'image à traiter (une partition).

La figure 3.22 illustre cette méthode pour la segmentation en trois classes d'une image cytologique (fond, noyaux et cytoplasme). Cette expérimentation compare également, les temps de calcul³ et les résultats obtenus avec différentes structures de graphes (basés sur les pixels de l'image ou les régions d'une partition). Ces temps incluent le calcul de la partition 3.22b et la construction du graphe associé.

La première colonne (3.22c) montre le résultat obtenu à partir d'un graphe grille en 8-voisinage associé aux pixels de l'image. Nous pouvons remarquer la nécessité de positionner des marqueurs entre les deux cellules afin de pouvoir les séparer. La deuxième colonne (3.22d) montre le résultat obtenu à partir d'un RAG pondéré calculé à partir de la partition 3.22b où la fonction de poids utilise avec la couleur moyenne des régions. Le résultat est obtenu à partir des même marqueurs que ceux utilisés dans le cas du graphe grille (colonne 3.22c). Nous pouvons constater que nous avons des résultats similaires tout en réduisant, dans le cas du RAG, le nombre d'itérations et les temps de calcul.

Les deux dernières colonnes de la figure 3.22 montrent des résultats avec des graphes complets calculés à partir des régions de la partition. Comme pour le cas

³Temps obtenus à partir un ordinateur équipé d'un processeur Intel Xeon quatre cœurs avec 16 Gb de mémoire vive.



Marqueurs initiaux et résultats (en colonne) avec le nombre d'itérations n et le temps d'exécution

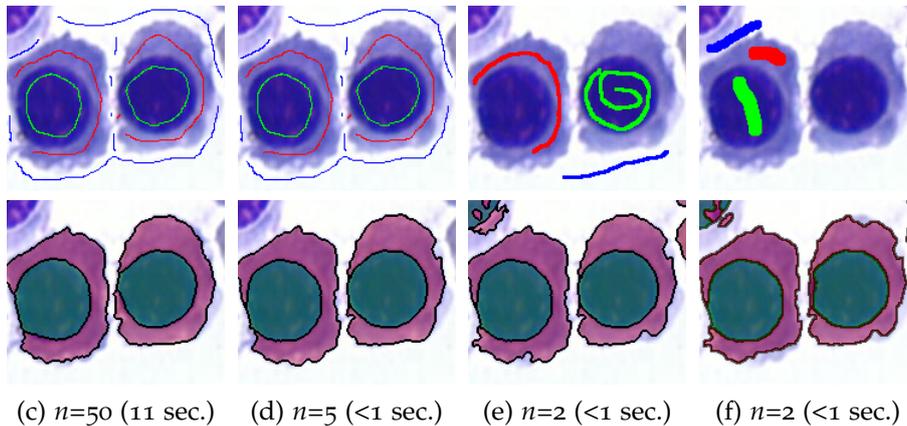


FIG. 3.22: Segmentation semi supervisée et comparaison avec différentes structures de graphes. De gauche à droite : les structures de graphes sont les suivantes. (c) : graphe grille en 8-voisinage ; (d) : RAG calculé à partir de la partition (b) ; (e) et (f) : graphes complets calculés à partir de la partition (b).

des expérimentations présentées dans la section 3.3 qui utilisaient le même type de graphe, nous obtenons les mêmes propriétés. (i) Peu de marqueurs sont nécessaires pour obtenir une segmentation totale des objets non connexes, rendant la méthode semi automatique moins fastidieuse pour l'utilisateur. (ii) Une convergence rapide de l'algorithme et une diminution du temps d'exécution sont obtenues du fait de la réduction du nombre de données à traiter (avec la partition nous réduisons le nombre de sommets à traiter de 97,7%). (iii) Les deux derniers résultats montrent également la robustesse de la méthode vis à vis de la position initiale des marqueurs.

Dans les sections suivantes, nous présentons l'utilisation de nos modèles de régularisation comme outils permettant de traiter et de segmenter des images cellulaires. Ces outils sont combinés entre eux (ou avec d'autres méthodes) pour obtenir des schémas de segmentation automatique ou semi automatique d'images couleurs de cytologie ou d'histologie. Dans la suite, nous présentons des schémas pour illus-

trer le potentiel de la méthodologie et une comparaison avec des méthodes proposées dans la littérature. Nous ouvrons également des pistes en proposant un schéma permettant de segmenter des images de lames entières issues d'un scanner à lames.

Schéma de segmentation interactive

La figure 3.23 montre un exemple de segmentation interactive (semi automatique) d'image couleur d'histologie et illustre les avantages d'une approche guidée par l'utilisateur. Lorsque les images à traiter deviennent plus complexes, les méthodes de segmentation automatiques ne sont parfois pas assez précises et une correction de l'utilisateur devient nécessaire.

La méthodologie utilisée dans cet exemple est la suivante.

- Simplification de l'image initiale par filtrage.
- Construction d'une partition.
- Segmentation par propagation de labels guidée par l'utilisateur.

Dans l'exemple de la figure 3.23, la première étape consiste à filtrer l'image initiale par régularisation discrète isotrope avec un graphe pondéré en 8-voisinage afin d'obtenir une image filtrée et très simplifiée (figure 3.23b) par rapport à l'image initiale (figure 3.23a). À partir de cette nouvelle image filtrée, une partition (figure 3.23c) est calculée afin d'obtenir une version très réduite de l'image initiale. Un RAG est ensuite associé à cette partition où chaque sommet correspond à une région caractérisée par sa couleur moyenne. La segmentation est calculée à partir de ce graphe.

La figure 3.23d montre les marqueurs initiaux déposés par l'utilisateur ainsi que la segmentation obtenue. Nous pouvons constater que le résultat obtenu n'est pas satisfaisant. Les structures tissulaires ont été mal extraites car les marqueurs initiaux n'ont pas été précisément placés pour séparer les regroupements de cellules. La figure 3.23e montre les marqueurs additionnels déposés par l'utilisateur afin de corriger la segmentation. Finalement, grâce à ces nouveaux labels, la segmentation désirée a pu être obtenue.

Comparaisons avec d'autres méthodes

Nous comparons ici notre schéma de segmentation interactive et rapide basée sur des graphes de régions avec d'autres méthodes proposées dans la littérature.

- Deux approches basées sur : l'algorithme des k moyennes d'une part, et une classification Bayésienne d'autre part [Lézoray et Cardot, 2002].
- Deux approches basées sur : la classification de pixels d'une part, et l'utilisation d'arbres avec fenêtres de recherches aléatoires d'autre part [Dumont *et al.*, 2007].

Pour plus de détails sur ces méthodes, le lecteur intéressé pourra se référer aux articles cités ci-dessus.

La comparaison est effectuée à partir d'une base d'images de tests composée de 10 images de cytologie des séreuses [Lézoray et Cardot, 2002]. Les cellules sont colorées avec la coloration standard internationale de Papanicolaou et la classification effectuée est constituée de deux classes : les pixels correspondant aux noyaux des cellules et les autres pixels de l'image.

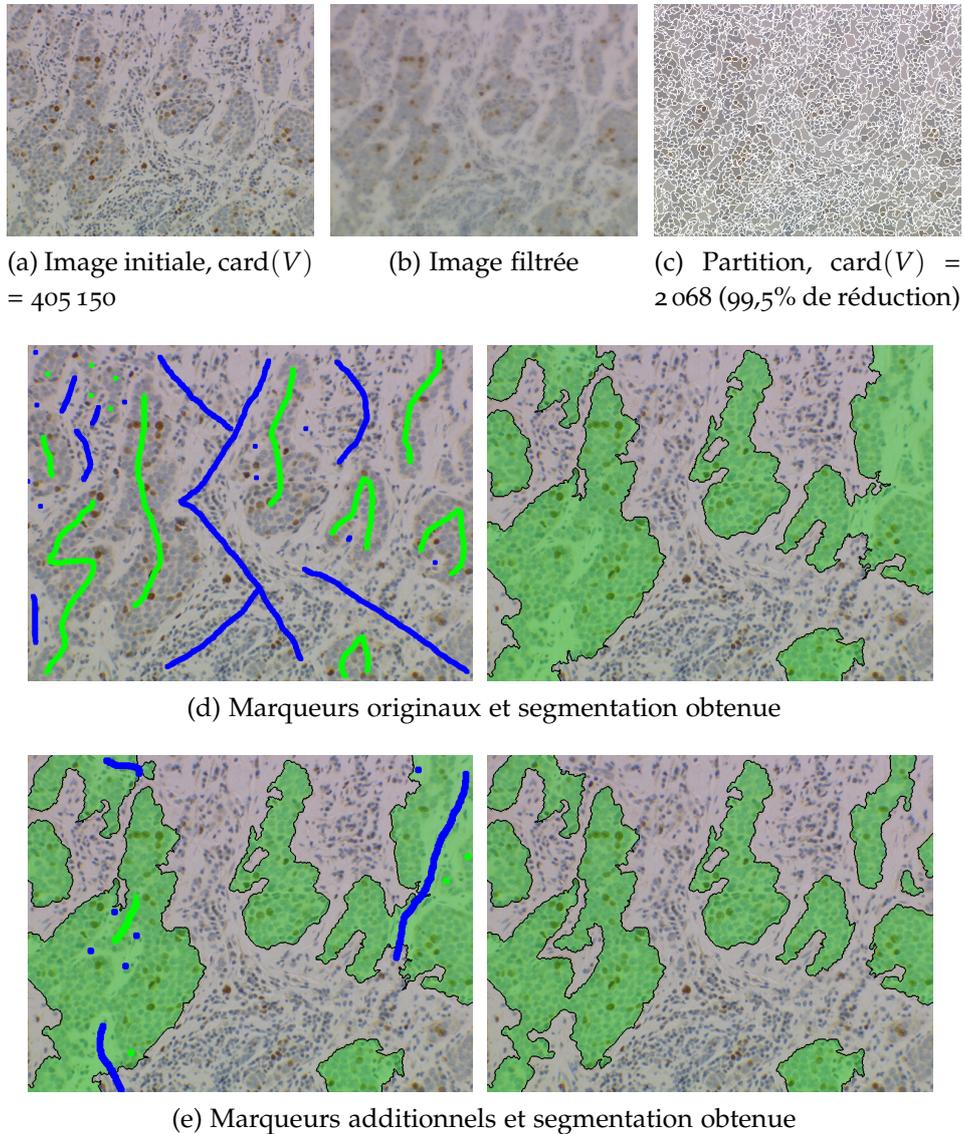


FIG. 3.23: Segmentation interactive d'image d'histologie couleur. La partition (c) est calculée à partir de (b) obtenue par filtrage de l'image initiale (a).

La figure 3.24 montre trois exemples de la base d'images de tests utilisées dans notre comparaison. La première ligne de cette figure présente les images initiales. La deuxième ligne de cette figure montre les vérités terrain réalisées manuellement par les experts du domaine. La dernière ligne montre, quant à elles, les segmentations obtenues à partir de notre méthode basée sur la segmentation guidée par l'utilisateur.

Afin de comparer les différentes approches, nous utilisons quatre mesures de taux de classification basées sur la proportions des pixels bien classés.

Les tableaux 3.4 et 3.5 montrent les taux de classification et comparent les résultats obtenus entre notre schéma de segmentation et deux autres approches [Lézoray

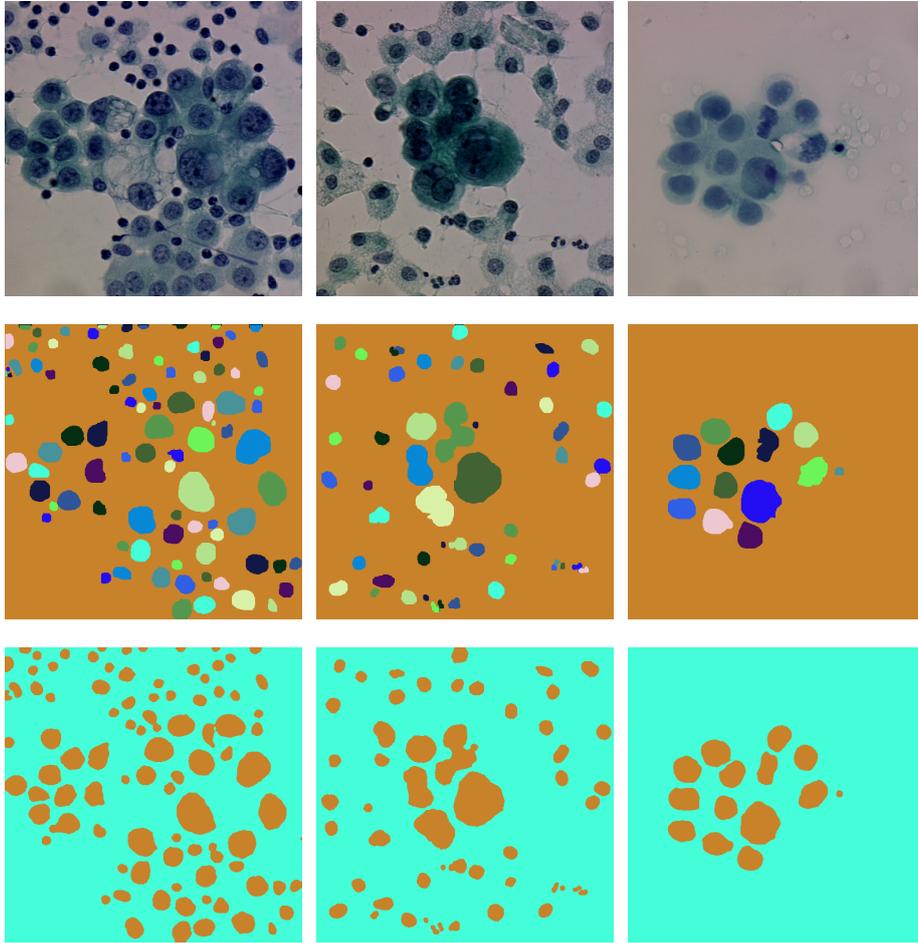


FIG. 3.24: Trois exemples de la base de tests utilisée. De la première à la dernière ligne : les images initiales, les vérités terrain, et les segmentations obtenues à partir de notre méthode semi automatique.

TAB. 3.4: Taux de classification et comparaison avec les approches de Dumont *et al.* [Dumont *et al.*, 2007].

	Basée pixel	Basé fenêtre aléatoire	Notre approche
R_0	95.93%	96.39%	98.41%

TAB. 3.5: Taux de classification et comparaison avec les approches de Lézoray *et al.* [Lézoray et Cardot, 2002].

	Basée k moyennes	Basée Bayes	Notre approche
R_1	88.7%	97.53%	92.31%
R_2	98.65%	95.40%	99.15%
R_3	93.67%	96.47%	95.73%

et Cardot, 2002, Dumont *et al.*, 2007].

Les meilleurs taux de classification sont mis en gras. Les mesures de taux de classification R_0 , R_1 , R_2 et R_3 sont définies par

$$R_0 = \frac{\text{nombre de pixels bien classés}}{\text{nombre total de pixels de l'image}} \quad (3.154)$$

$$R_1 = \frac{\text{nombre de pixels de la classe noyau bien classés}}{\text{nombre total de pixels de la classe noyau de l'image}} \quad (3.155)$$

$$R_2 = \frac{\text{nombre de pixels de la classe fond bien classés}}{\text{nombre total de pixels de la classe fond de l'image}} \quad (3.156)$$

$$R_3 = \frac{R_1 + R_2}{2} \quad (3.157)$$

Les résultats présentés par le tableau 3.4 montrent que notre schéma a le meilleur taux de classification en comparaison aux approches proposées par Dumont *et al.*

Les résultats présentés par le tableau 3.5 montrent les taux de classification par classe pour les différentes approches considérées. Notre approche permet d'obtenir de meilleurs résultats que ceux obtenus par l'algorithme des k moyennes, mais des résultats moins bons que ceux obtenus par la classification Bayésienne pour les pixels classés comme étant de la classe noyau. Ces résultats peuvent être interprétés de la manière suivante. Un des avantages de notre approche est que l'utilisateur peut revenir et corriger les marqueurs initiaux jusqu'à obtenir les résultats de segmentation désirés. La conséquence est que nous pouvons obtenir de meilleurs taux de classification comparativement à une méthode automatique. En ce qui concerne les taux de classification des pixels de la classe fond ou de la classe noyau, nous pouvons noter que l'approche Bayésienne a tendance à sur-segmenter les noyaux et que notre schéma tend à les sous-segmenter.

Schéma de segmentation automatique de lames entières

Les images cellulaires que nous avons présentées jusqu'à maintenant proviennent d'une caméra embarquée sur un microscope et correspondent uniquement à un champ (sous partie) de la lame entière. Cette méthode d'acquisition permet d'obtenir des images de champs qui, une fois recalées, permet de reconstituer l'image de la lame entière.

Récemment, les avancées technologiques de l'acquisition dans le domaine de la pathologie et de la biologie ont fait émerger de nouvelles stations d'imagerie. Ce nouveau type de matériel permet de disposer d'interfaces pour la microscopie virtuelle par l'imagerie d'une lame entière (ILE). Ces ILE permettent de scanner des prélèvements dans leur totalité et de disposer de différentes résolutions de l'image scannée. La figure 3.25 montre un exemple de lame scannée à partir d'un dispositif d'ILE et l'image obtenue à différentes résolutions.

Les images obtenues sont représentées par un seul fichier contenant l'image à pleine résolution et par une séquence codant différentes résolutions de l'image de base. Ce type d'images a donc la particularité de faire apparaître différentes structures à différentes résolutions.

Le fichier contenant les images peut atteindre la taille de 30Go pour une unique lame scannée. La gestion de telles masses de données représente un réel défi de cette nouvelle ère de la microscopie numérique. Les images contenues dans les fichiers

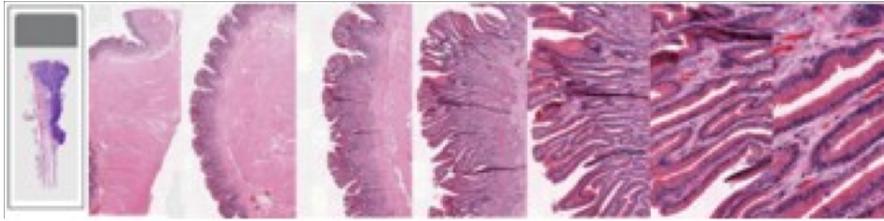


FIG. 3.25: Exemple de lame acquise à partir un scanner avec des zooms à différentes résolutions

représentant les lames entières sont en fait des pyramides régulières. La figure 3.26 montre un schéma illustrant la pyramide représentant les différentes résolutions de l'image de la lame entière.

Des travaux récents montrent différentes approches afin de traiter ces images de lame entière. Nous pouvons par exemple citer une méthode basée sur la décomposition en ondelettes [Signolle *et al.*, 2009] ou encore celle basée sur l'apprentissage et une sélection d'attributs à chaque niveau de la pyramide [Sertel *et al.*, 2009].

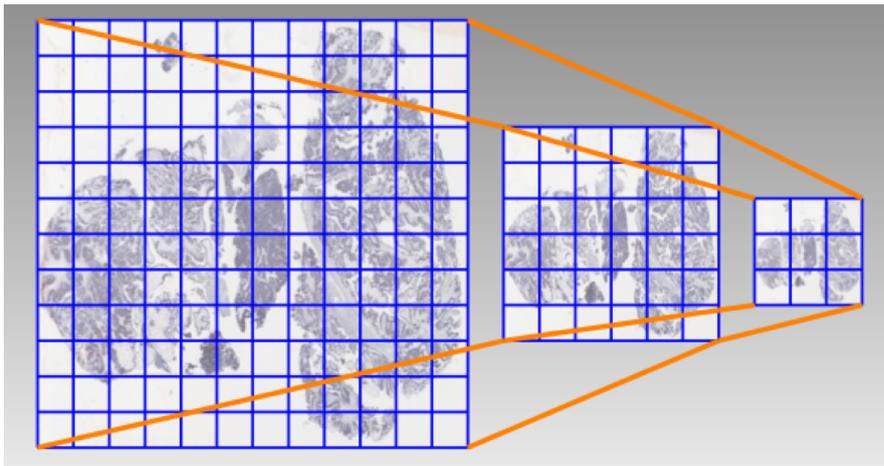


FIG. 3.26: Schéma de la pyramide représentant les images d'une lame entière acquise par un dispositif d'ILE.

Dans cette section, nous proposons d'utiliser la pyramide et un schéma de segmentation automatique multi échelles des images de lames entières basé sur nos modèles de régularisation définis sur des graphes. Les figures 3.28 et 3.29 montrent un exemple de segmentation multi échelles d'une lame entière pour les deux premiers niveaux de la pyramide. L'image à traiter (figure 3.27) correspond à un prélèvement de cancer du sein coloré par immuno-histochimie. Le but est d'extraire les tissus malins et de localiser les cellules malignes afin de pouvoir calculer un taux de prolifération du cancer.

La méthodologie utilisée pour la segmentation d'un niveau donné de la pyramide est la suivante.

- Filtrage de l'image par régularisation isotrope.
- Classification par k moyennes robuste de l'image [Kanungo *et al.*, 2002].

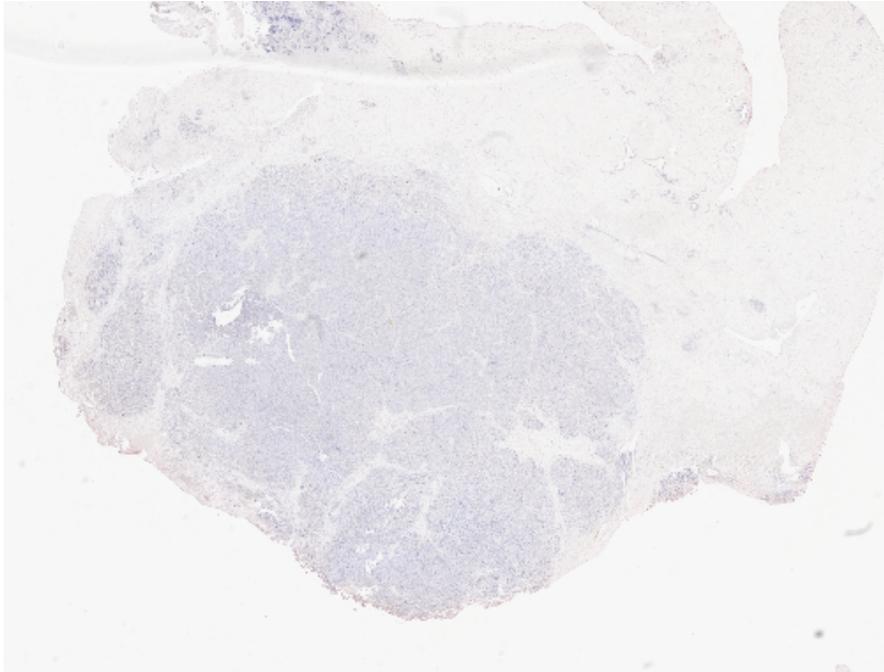


FIG. 3.27: Prélèvement original intervenant pour la segmentation multi échelles présentée dans les figures 3.28 et 3.29.

- Détection et construction d'une bande étroite autour des contours des régions obtenues.
- Classification semi supervisée dans la bande étroite autour des contours afin de les affiner.

La figure 3.28 montre la segmentation du premier niveau de la pyramide en utilisant la méthodologie décrite précédemment. La figure 3.28a montre le résultat du filtrage par régularisation isotrope. La figure 3.28b montre le résultat de la classification par l'algorithme robuste des k moyennes. À partir du résultat de cette classification, une détection des contours est réalisée et une bande étroite (de taille relative au facteur d'agrandissement) est construite autour de ce contour. Un graphe grille en 8-voisinage est alors construit à partir des pixels de la bande étroite et les labels initiaux sont initialisés à partir de la classification obtenue par l'algorithme des k moyennes. La figure 3.28c montre la carte de régions obtenue après la classification semi supervisée (avec le paramètre $\lambda = 0$) permettant de modifier et d'affiner les contours précédemment calculés. La figure 3.28d montre la segmentation finale superposée à l'image initiale.

La figure 3.29 montre la segmentation du deuxième niveau de la pyramide représentant la figure 3.27. Cette segmentation est réalisée à partir du niveau précédent et utilise la même méthodologie. La figure 3.29a montre le résultat de la première étape : le filtrage sur le deuxième niveau de la pyramide. Afin d'appliquer l'étape de classification par l'algorithme des k moyennes, nous utilisons une étape préliminaire de super résolution permettant de passer de la carte de régions du premier niveau vers une image de la taille du deuxième niveau. À partir de cette nouvelle

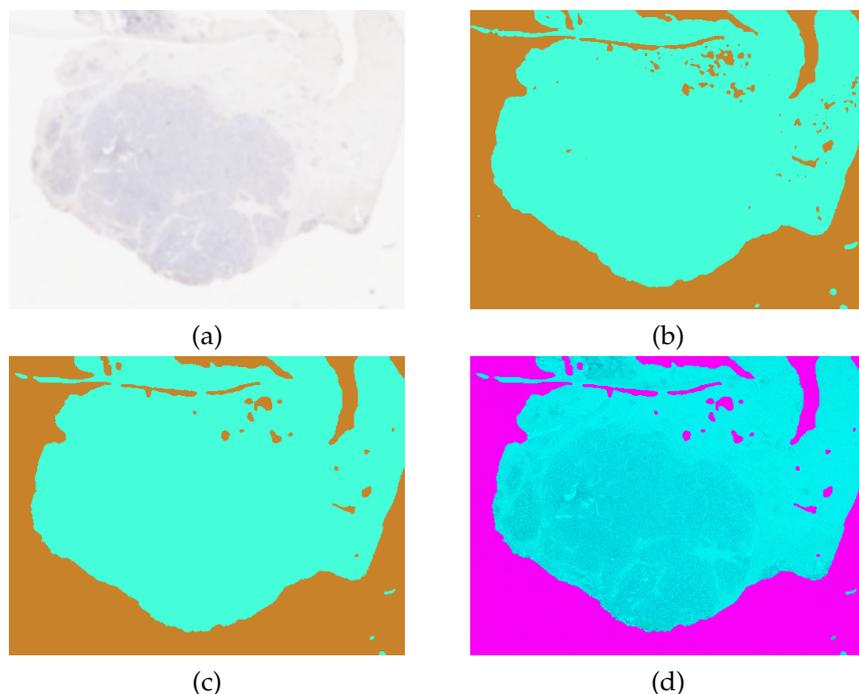


FIG. 3.28: Segmentation du premier niveau de la pyramide représentant la lame entière 3.27. (a) : filtrage par régularisation isotrope, (b) : classification par k moyennes ($k=2$), (c) : classification semi supervisée pour affiner les contours, (d) : segmentation finale superposée sur l’image initiale.

carte de régions, l’algorithme des k moyennes (avec $k = 2$) est appliqué uniquement dans la zone précédemment détectée comme étant du tissu (zone vert clair de la figure 3.28c). La figure 3.29b montre le résultat de cette première classification. La figure 3.29c montre le résultat de la segmentation semi supervisée (afin d’affiner les contours) obtenue avec une bande étroite calculée à partir du résultat de la figure 3.29b. La figure 3.29d montre la segmentation finale superposée sur l’image initiale.

Finalement, la même méthodologie peut être appliquée à l’ensemble des niveaux de la pyramide pour obtenir une segmentation des structures tissulaires et cellulaires contenues dans la lame entière.

3.3.5 Conclusion

Dans cette section, nous avons présenté des problèmes d’interpolation formalisés à partir de nos modèles de régularisation définis sur des graphes. Nous nous sommes intéressés plus particulièrement aux problèmes de la colorisation des images monochromes et de la classification semi supervisée.

Dans le cadre du traitement des images, nous avons montré que les configurations non locales avec des patches permettaient d’obtenir de meilleurs résultats que les schémas locaux. Nous avons également illustré les bénéfices de l’utilisation des graphes de régions pour la segmentation des images. Nos modèles étant défi-

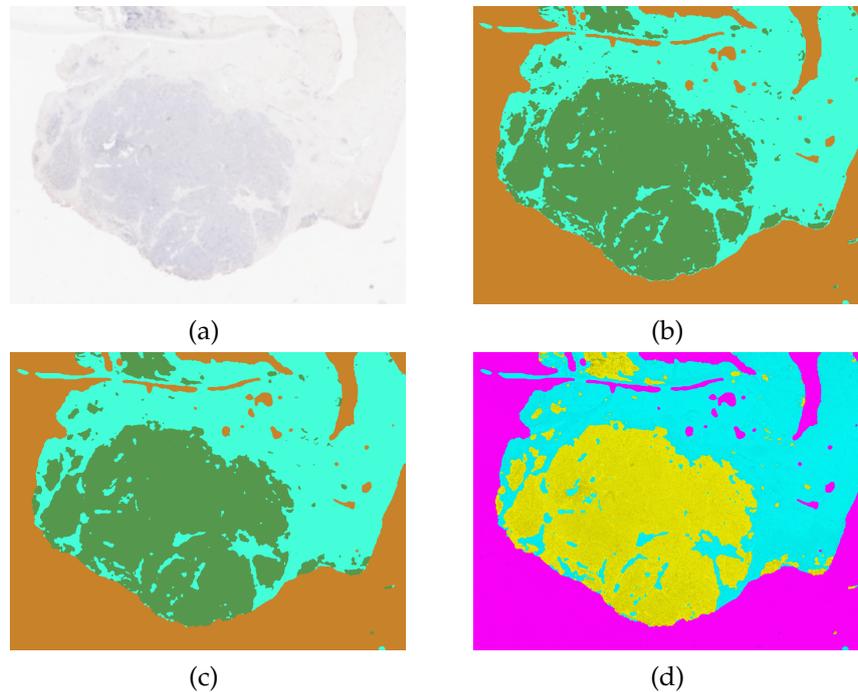


FIG. 3.29: Segmentation du deuxième niveau de la pyramide représentant la lame entière 3.27. (a) : filtrage par régularisation isotrope, (b) : classification par k moyennes ($k=2$), (c) : classification semi supervisée pour affiner les contours et (d) : segmentation finale superposée sur l'image initiale.

nis avec des graphes, l'extension naturelle de ces méthodes permet d'envisager le traitement de tout type de données pouvant être représenté par un graphe.

À partir de nos modèles, une perspective intéressante serait d'adapter, avec les équations aux différences partielles définies sur des graphes, d'autres problèmes d'interpolation tels que la retouche d'image ou encore la super résolution. Dans le domaine du traitement des images, cette adaptation permettrait d'intégrer les schémas non locaux avec des patches dans la résolution de ces problématiques. Dans le domaine de l'apprentissage, elle permettrait d'envisager de nouvelles pistes telles que l'interpolation des valeurs manquantes ou encore le traitement multi échelles des bases de données.

De plus, nous avons présenté des schémas de segmentation d'images microscopiques de cellules. Ces schémas utilisent nos modèles de régularisation comme des outils adaptatifs pouvant être combinés entre eux ou avec d'autres méthodes.

Nous avons montré des expérimentations permettant de traiter et de segmenter des images biomédicales de manière automatique ou semi automatique. Nous nous sommes également intéressés à une nouvelle génération d'images cellulaires : les images de lame entière. Ces images constituent un nouveau défi dans le domaine du traitement des images du fait de la très grande masse de données à traiter. Nous avons considéré cette problématique et avons proposé un schéma de segmentation multi échelles permettant de les traiter.

Enfin, une perspective intéressante, serait d'utiliser ces mêmes outils pour ana-

lyser, catégoriser et classer les objets dans des grandes bases de données. La figure 3.30 montre deux exemples de classification semi supervisée d'objets contenus dans des bases de données. Les figures 3.30a et 3.30c montrent les bases de données d'images initiales (bases de cellules segmentées et de l'USPS) avec les marqueurs initiaux. Les figures 3.30b et 3.30d montrent les classifications finales.

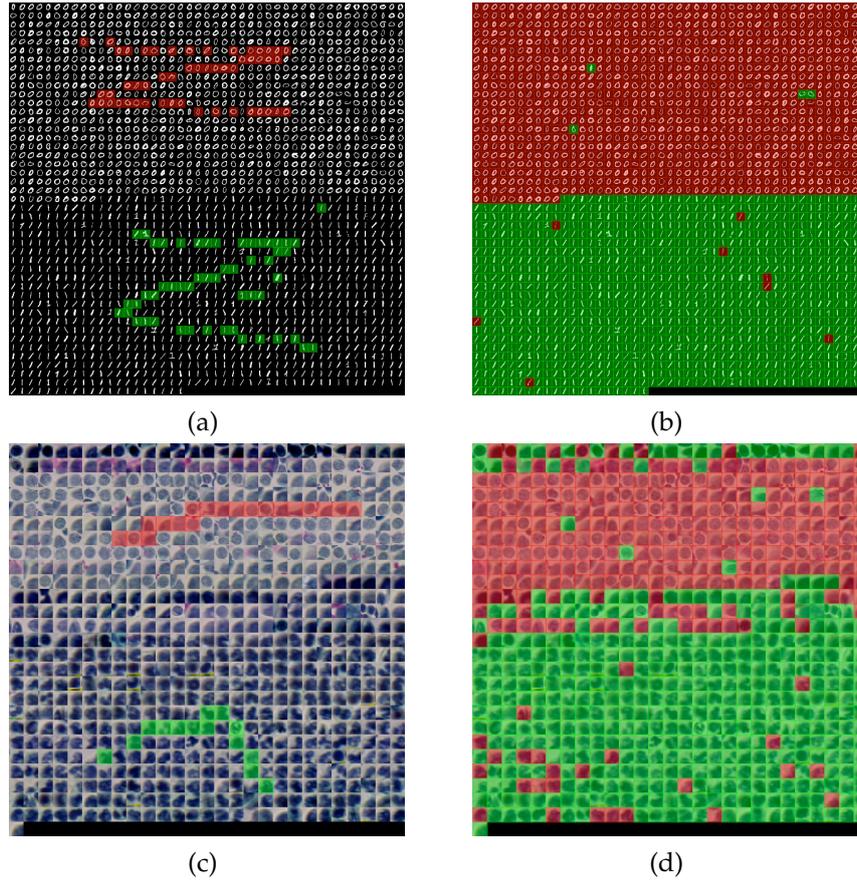


FIG. 3.30: Classification semi supervisée d'objets contenus dans des bases de données (bases d'images de chiffres et de cellules). (a) et (c) : données et marqueurs initiaux. (b) et (d) : classifications finales.

3.4 Optimisation globale pour la segmentation de données

3.4.1 Introduction

En segmentation d'images, les méthodes variationnelles basées sur la régularisation et les équations aux dérivées partielles sont très utilisées. Cependant, comme nous l'avons déjà mentionné, peu de travaux intégrant les configurations non locales dans les modèles de segmentation existent dans la littérature. Les travaux qui ont tout de même abordé ceci reposent principalement sur une extension de la variation totale vers des schémas non locaux. Nous pouvons citer des méthodes semi supervisées définies dans le domaine continu [Gilboa et Osher, 2007a] ou discret [Szlam *et al.*, 2008, Houhou *et al.*, 2009], notre approche présentée dans les sections précédentes et les modèles de contours actifs globaux [Bresson et Chan, 2008].

Cette dernière approche de segmentation repose sur une minimisation globale continue [Chan *et al.*, 2006] et sur la formule de la co-aire [Chambolle, 2005] pour rendre convexe ce problème initialement non convexe et trouver un minimum global. L'obtention d'un minimum global est important pour obtenir des algorithmes de segmentation robustes et indépendants de la position des contours initiaux, comme cela est fait dans les méthodes de contours actifs [Chan et Vese, 2001].

Cette classe d'optimisation globale peut s'exprimer par la seconde classe de modèles variationnels que nous avons précédemment proposée :

$$\min_{f:V \rightarrow \{0,1\}} \left\{ \mathcal{E}_2(f, f^0, \lambda) = \mathcal{R}_{w,q,\phi}^*(f) + \sum_{u \in V} \lambda g(f^0)(u) f(u) \right\} \quad (3.158)$$

où la fonction $g \in \mathcal{H}(V)$ est une fonction dépendant de la fonction initiale $f^0 \in \mathcal{H}(V)$. Le terme de régularisation prendra les deux formes suivantes : avec $\phi(s) = \frac{1}{p}s^p$ et $q = p$

$$\mathcal{R}_{w,p,\phi}^*(f) = \frac{1}{2p} \sum_{u \in V} \|(\nabla_w^* f)(u)\|_p^p = \frac{1}{2p} \mathcal{R}_{w,p}^*(f) , \quad (3.159)$$

et $\phi(s) = s$ et $q = \infty$

$$\mathcal{R}_{w,\infty,\phi}^*(f) = \max_{u \in V} \|(\nabla_w^* f)(u)\|_\infty = \mathcal{R}_{w,\infty}^*(f) . \quad (3.160)$$

Les fonctionnelles de régularisation $\mathcal{R}_{w,p}^*(f)$ et $\mathcal{R}_{w,\infty}^*(f)$ désignent les différentes fonctionnelles que nous avons proposées dans les sections précédentes. Pour une fonction $f \in \mathcal{H}(V)$

$$\mathcal{R}_{w,p}^*(f) = \{ \mathcal{R}_{w,p}(f), \mathcal{R}_{w,p}^+(f), \mathcal{R}_{w,p}^-(f) \} \quad (3.161)$$

$$\mathcal{R}_{w,\infty}^*(f) = \{ \mathcal{R}_{w,\infty}(f), \mathcal{R}_{w,\infty}^+(f), \mathcal{R}_{w,\infty}^-(f) \} . \quad (3.162)$$

Cette classe de modèles peut être considérée comme la version discrète des modèles similaires définis dans le domaine continu.

Dans cette section, nous proposons des solutions et des algorithmes itératifs permettant de résoudre ce problème non convexe et obtenir un minimum global. Ces solutions sont basées sur l'utilisation de la formule de la co-aire définie sur des graphes et une méthode discrète de relaxation.

Notre approche unificatrice permet de segmenter des images en intégrant des schémas non locaux mais également de traiter tout type de données même celles définies sur des domaines irréguliers et de grande dimension.

Remarque. Dans cette section, les algorithmes utilisés pour résoudre le problème de minimisation sont itératifs ce qui permet d'obtenir une implantation facilitée. Cependant, des méthodes d'optimisation combinatoire permettant de minimiser des énergies binaires existent, nous pouvons par exemple citer les travaux suivants [Boykov et Kolmogorov, 2004, Grady et Alvino, 2009].

3.4.2 Solutions de la deuxième classe de modèles variationnels

Le problème de minimisation (3.158) est non convexe tout comme sa version définie dans le domaine continu [Chan et Esedoglu, 2005, Burger et Hintermüller, 2005, Chan *et al.*, 2006]. Afin de pouvoir trouver un minimum global à ce problème, nous pouvons reformuler l'énergie discrète en une version relaxée convexe

$$\min_{f:V \rightarrow [0,1]} \left\{ \tilde{\mathcal{E}}_2(f, f^0, \lambda) = \mathcal{R}_{w,q,\phi}^*(f) + \sum_{u \in V} \lambda g(f^0)(u) f(u) \right\} \quad (3.163)$$

où la fonction f recherchée est maintenant définie dans l'intervalle $[0, 1]$.

Dans cette section, nous allons montrer que pour les cas particuliers où $q = 1$ et $q = \infty$ des fonctionnelles (3.161) et (3.162), nous pouvons obtenir la solution pour le problème (3.158) en utilisant le problème discret relaxé (3.163) et la formule de la co-aire discrète définie sur des graphes.

Formule de la co-aire sur des graphes

Nous montrons ici que les fonctionnelles de régularisation $\mathcal{R}_{w,p}^*$ avec $p = 1$ satisfont la formule de la co-aire.

Propriété 3.4.1. Pour toute fonction $f : V \rightarrow \mathbb{R}$, les fonctionnelles de régularisation $\mathcal{R}_{w,p}^*(f)$, avec $p = 1$ satisfont la formule de la co-aire

$$\mathcal{R}_{w,1}^*(f) = \int_{-\infty}^{+\infty} \mathcal{R}_{w,1}^*(\chi_{(f>t)}) \, dt \quad (3.164)$$

$$(3.165)$$

avec χ la fonction indicatrice et $t \geq 0$.

Démonstration. Soit f une fonction définie par $f : V \rightarrow \mathbb{R}$. Pour tout $t \geq 0$, la fonction indicatrice $\chi_{(f>t)} : V \rightarrow \{0, 1\}$ se définit, en un sommet $u \in V$, par

$$\chi_{(f>t)}(u) = \begin{cases} 0 & \text{si } f(u) < t \\ 1 & \text{si } f(u) \geq t \end{cases} . \quad (3.166)$$

Par définition de la fonction de poids w , nous avons pour n'importe quelle arête $uv \in E : w_{uv} \geq 0$.

Prenons tout d'abord le cas suivant où

$$\mathcal{R}_{w,1}^* = \mathcal{R}_{w,1} = \sum_{u \in V} \sum_{v \sim u} \sqrt{w_{uv}} |f(v) - f(u)| \quad (3.167)$$

en utilisant l'identité suivante

$$|a - b| = \int_{-\infty}^{+\infty} |\chi_{(a>t)} - \chi_{(b>t)}| \, dt \quad (3.168)$$

nous avons

$$\sum_{u \in V} \sum_{v \sim u} \sqrt{w_{uv}} |f(v) - f(u)| = \int_{-\infty}^{+\infty} \sum_{u \in V} \sum_{v \sim u} \sqrt{w_{uv}} |\chi_{(f>t)}(v) - \chi_{(f>t)}(u)| \, dt \quad (3.169)$$

ce qui permet d'obtenir la propriété 3.164

$$\mathcal{R}_{w,1}(f) = \int_{-\infty}^{+\infty} \mathcal{R}_{w,1}(\chi_{(f>t)}) \, dt . \quad (3.170)$$

Prenons maintenant les deux cas où

$$\mathcal{R}_{w,1}^* = \mathcal{R}_{w,1}^+ = \sum_{u \in V} \sum_{v \sim u} \sqrt{w_{uv}} |\max(0, f(v) - f(u))| \quad (3.171)$$

$$\mathcal{R}_{w,1}^* = \mathcal{R}_{w,1}^- = \sum_{u \in V} \sum_{v \sim u} \sqrt{w_{uv}} |\min(0, f(v) - f(u))| . \quad (3.172)$$

En utilisant les identités suivantes

$$\max(0, a - b) = \int_{-\infty}^{+\infty} \max(0, \chi_{(a>t)} - \chi_{(b>t)}) \, dt \quad (3.173)$$

$$\min(0, a - b) = \int_{-\infty}^{+\infty} \min(0, \chi_{(a>t)} - \chi_{(b>t)}) \, dt \quad (3.174)$$

$$\min(0, a - b) = \max(0, b - a) , \quad (3.175)$$

nous obtenons la propriété 3.164 pour les fonctionnelles directionnelles

$$\mathcal{R}_{w,1}^+(f) = \int_{-\infty}^{+\infty} \mathcal{R}_{w,1}^+(\chi_{(f>t)}) \, dt \quad (3.176)$$

$$\mathcal{R}_{w,1}^-(f) = \int_{-\infty}^{+\infty} \mathcal{R}_{w,1}^-(\chi_{(f>t)}) \, dt \quad (3.177)$$

□

Basé sur une approche similaire, nous pouvons déduire les mêmes propriétés pour les fonctionnelles basées sur la norme \mathcal{L}_∞

$$\mathcal{R}_{w,\infty}^* = \mathcal{R}_{w,\infty} = \max_{u \in V} \max_{v \sim u} \sqrt{w_{uv}} |f(v) - f(u)| = \int_{-\infty}^{+\infty} \mathcal{R}_{w,\infty}(\chi_{(f>t)}) \, dt \quad (3.178)$$

$$\mathcal{R}_{w,\infty}^* = \mathcal{R}_{w,\infty}^+ = \max_{u \in V} \max_{v \sim u} \sqrt{w_{uv}} |\max(0, f(v) - f(u))| = \int_{-\infty}^{+\infty} \mathcal{R}_{w,\infty}^+(\chi_{(f>t)}) \, dt \quad (3.179)$$

$$\mathcal{R}_{w,\infty}^* = \mathcal{R}_{w,\infty}^- = \max_{u \in V} \max_{v \sim u} \sqrt{w_{uv}} |\min(0, f(v) - f(u))| = \int_{-\infty}^{+\infty} \mathcal{R}_{w,\infty}^-(\chi_{(f>t)}) \, dt . \quad (3.180)$$

Les fonctionnelles $\mathcal{R}_{w,1}^*$ et $\mathcal{R}_{w,\infty}^*$ satisfaisant la propriété de la co-aire, ont alors les propriétés suivantes.

- Convexité. Pour toutes fonctions f et h définies dans $\mathcal{H}(V)$, nous avons les relations

$$\mathcal{R}_{w,1}^*(tf + (1-t)) \leq t\mathcal{R}_{w,1}^*(f) + (1-t)\mathcal{R}_{w,1}^*(h) \quad (3.181)$$

$$\mathcal{R}_{w,\infty}^*(tf + (1-t)) \leq t\mathcal{R}_{w,\infty}^*(f) + (1-t)\mathcal{R}_{w,\infty}^*(h) . \quad (3.182)$$

- Semi continuité inférieure.
- Positivité et homogénéité de degré 1. Pour toute fonction $f \in \mathcal{H}(V)$ et $t \geq 0$

$$\mathcal{R}_{w,1}^*(tf) = t\mathcal{R}_{w,1}^*(f) \quad (3.183)$$

$$\mathcal{R}_{w,\infty}^*(tf) = t\mathcal{R}_{w,\infty}^*(f) \quad (3.184)$$

Méthode discrète de relaxation

À partir de la propriété de la co-aire et du problème de minimisation relaxé (3.163) nous pouvons trouver le minimum global satisfaisant le problème (3.158) pour les fonctionnelles de régularisation basées sur les normes \mathcal{L}_1 (3.161) et \mathcal{L}_∞ (3.162). En suivant une approche similaire à la résolution de la version définie dans le domaine continu des problèmes (3.158) et (3.163) [Burger et Hintermüller, 2005, Chan *et al.*, 2006], nous pouvons montrer que chaque ensemble de niveaux de la solution du problème relaxé (3.163) est une solution du problème d'optimisation original (3.158).

Propriété 3.4.2. Avec une fonction $\phi(s) = \frac{1}{p}s^p$, et pour $q = p = 1$ la solution du problème convexe suivant

$$\min_{f:V \rightarrow [0,1]} \left\{ \tilde{\mathcal{E}}_3(f, f^0, \lambda) = \frac{1}{2}\mathcal{R}_{w,1}^*(f) + \sum_{u \in V} \lambda g(f^0)(u)f(u) \right\} \quad (3.185)$$

est également une solution du problème original suivant

$$\min_{f:V \rightarrow \{0,1\}} \left\{ \mathcal{E}_3(f, f^0, \lambda) = \frac{1}{2}\mathcal{R}_{w,1}^*(f) + \sum_{u \in V} \lambda g(f^0)(u)f(u) \right\} \quad (3.186)$$

où

$$\mathcal{R}_{w,1}^*(f) = \{ \mathcal{R}_{w,1}(f), \mathcal{R}_{w,1}^+(f), \mathcal{R}_{w,1}^-(f) \} . \quad (3.187)$$

Démonstration. Supposons que \hat{f} soit la solution du problème initial (3.186) et que \tilde{f} soit la solution du problème relaxé (3.185). Avec l'identité suivante pour un sommet $u \in V$

$$\tilde{f}(u) = \int_0^{\tilde{f}(u)} dt = \int_0^1 \chi_{(\tilde{f} > t)}(u) dt \quad (3.188)$$

et un changement de variable, nous obtenons pour le terme d'attache aux données

$$\lambda \sum_{u \in V} g(f^0)(u) \tilde{f}(u) = \lambda \sum_{u \in V} \left(\int_0^{\tilde{f}(u)} g(f^0)(u) dt \right) \quad (3.189)$$

$$= \lambda \int_0^1 \left(\sum_{u \in V} g(f^0)(u) \chi_{(\tilde{f} > t)}(u) \right) dt \quad (3.190)$$

À partir de la propriété de la co-aire pour des fonctions à variations bornées, nous avons la relation suivante

$$\mathcal{R}_{w,1}^*(\tilde{f}) = \int_0^1 \mathcal{R}_{w,1}^*(\chi_{(\tilde{f}>t)}) \, dt . \quad (3.191)$$

En utilisant ces deux dernières relations, nous avons

$$\tilde{\mathcal{E}}_3(\tilde{f}, f^0, \lambda) = \frac{1}{2} \mathcal{R}_{w,1}^*(\tilde{f}) + \sum_{u \in V} \lambda g(f^0)(u) \tilde{f}(u) \quad (3.192)$$

$$= \int_0^1 \left(\frac{1}{2} \mathcal{R}_{w,1}^*(\chi_{(\tilde{f}>t)}) + \sum_{u \in V} \lambda g(f^0) \chi_{(\tilde{f}>t)}(u) \right) dt \quad (3.193)$$

$$= \int_0^1 \tilde{\mathcal{E}}_3(\chi_{(\tilde{f}>t)}, f^0, \lambda) \, dt . \quad (3.194)$$

Puisque $\chi_{(\tilde{f}>t)} \in \{0, 1\}$, pour tout t , nous avons

$$\tilde{\mathcal{E}}_3(\chi_{(\tilde{f}>t)}, f^0, \lambda) \geq \tilde{\mathcal{E}}_3(\hat{f}, f^0, \lambda) . \quad (3.195)$$

De plus, résultant du problème relaxé, nous avons l'inégalité suivante

$$\tilde{\mathcal{E}}_3(\hat{f}, f^0, \lambda) \geq \tilde{\mathcal{E}}_3(\tilde{f}, f^0, \lambda) , \quad (3.196)$$

donnant ainsi

$$\tilde{\mathcal{E}}_3(\tilde{f}, f^0, \lambda) = \int_0^1 \tilde{\mathcal{E}}_3(\chi_{(\tilde{f}>t)}, f^0, \lambda) \, dt \quad (3.197)$$

$$\geq \int_0^1 \tilde{\mathcal{E}}_3(\hat{f}, f^0, \lambda) \, dt = \tilde{\mathcal{E}}_3(\hat{f}, f^0, \lambda) \geq \tilde{\mathcal{E}}_3(\tilde{f}, f^0, \lambda) \quad (3.198)$$

À partir de cette dernière relation, nous obtenons finalement

$$\tilde{\mathcal{E}}_3(\tilde{f}, f^0, \lambda) = \tilde{\mathcal{E}}_3(\hat{f}, f^0, \lambda) = \tilde{\mathcal{E}}_3(\chi_{(\tilde{f}>t)}, f^0, \lambda) . \quad (3.199)$$

Nous pouvons ainsi dire que pour chaque solution \tilde{f} minimisant le problème relaxé, (3.185) nous pouvons calculer une solution pour le problème original (3.186) en considérant les ensembles de niveaux $\chi_{(\tilde{f}>t)}$ de la fonction \tilde{f} . La convexité du problème relaxé garantit que chaque solution du problème relaxé soit un minimum global correspondant aussi à un minimum global du problème non convexe initial. \square

De la même manière que pour la propriété 3.4.2, nous pouvons déduire la propriété suivante.

Propriété 3.4.3. Avec une fonction $\phi(s) = s$, et pour $q = \infty$ la solution du problème convexe suivant

$$\min_{f: V \rightarrow [0,1]} \left\{ \tilde{\mathcal{E}}_3(f, f^0, \lambda) = \mathcal{R}_{w,\infty}^*(f) + \sum_{u \in V} \lambda g(f^0)(u) f(u) \right\} \quad (3.200)$$

est également une solution du problème initial

$$\min_{f: V \rightarrow \{0,1\}} \left\{ \mathcal{E}_3(f, f^0, \lambda) = \mathcal{R}_{w,\infty}^*(f) + \sum_{u \in V} \lambda g(f^0)(u) f(u) \right\} \quad (3.201)$$

où

$$\mathcal{R}_{w,\infty}^*(f) = \{ \mathcal{R}_{w,\infty}(f), \mathcal{R}_{w,\infty}^+(f), \mathcal{R}_{w,\infty}^-(f) \} . \quad (3.202)$$

Algorithme itératif d'approximation

Les seuls cas où nous pouvons trouver une solution au problème (3.158) sont ceux qui satisfont la propriété de la co-aire, c'est-à-dire les fonctionnelles $\mathcal{R}_{w,1}^*$ et $\mathcal{R}_{w,\infty}^*$.

Dans cette section, nous résolvons le problème de minimisation (3.158) en ne considérant que le cas de la fonctionnelle $\mathcal{R}_{w,1}^*$. Pour approximer la solution de cette minimisation, nous utilisons un algorithme itératif discret.

Remarque. Dans ce manuscrit, nous ne considérons pas le cas de la fonctionnelle $\mathcal{R}_{w,\infty}^*$. Pour résoudre ce problème de minimisation, une méthode possible est de considérer le problème de manière duale et d'utiliser des techniques d'optimisation combinatoire. Le lecteur intéressé pourra se référer par exemple à l'article suivant [Sinop et Grady, 2007] et aux références associées.

Propriété 3.4.4. Les fonctionnelles de régularisation directionnelles $\mathcal{R}_{w,1}^+$ et $\mathcal{R}_{w,1}^-$ peuvent s'exprimer en fonction de la fonctionnelle $\mathcal{R}_{w,1}$. Pour une fonction $f \in \mathcal{H}(V)$, nous avons

$$\mathcal{R}_{w,p}^+(f) = \mathcal{R}_{w,p}^-(f) = \frac{1}{2} \mathcal{R}_{w,p}(f) . \quad (3.203)$$

Démonstration. En utilisant les identités suivantes

$$|\max(0, a - b)| = \frac{|a - b| + (a - b)}{2} \quad (3.204)$$

$$|\min(0, a - b)| = \frac{|a - b| - (a - b)}{2} \quad (3.205)$$

nous avons pour la fonctionnelle $\mathcal{R}_{w,1}^+(f)$

$$\mathcal{R}_{w,1}^+(f) = \sum_{u \in V} \sum_{v \sim u} \sqrt{w_{uv}} |\max(0, f(v) - f(u))| \quad (3.206)$$

$$= \frac{1}{2} \sum_{u \in V} \sum_{v \sim u} \sqrt{w_{uv}} |f(v) - f(u)| + \frac{1}{2} \sum_{u \in V} \sum_{v \sim u} \sqrt{w_{uv}} (f(v) - f(u)) \quad (3.207)$$

$$= \frac{1}{2} \mathcal{R}_{w,1}(f) + \frac{1}{2} \sum_{u \in V} \sum_{v \sim u} \sqrt{w_{uv}} (f(v) - f(u)) \quad (3.208)$$

$$= \frac{1}{2} \mathcal{R}_{w,1}(f) \quad (3.209)$$

du fait que le terme

$$\frac{1}{2} \sum_{u \in V} \sum_{v \sim u} \sqrt{w_{uv}} (f(v) - f(u)) = 0 . \quad (3.210)$$

De la même manière, nous avons pour la fonctionnelle $\mathcal{R}_{w,1}^-$

$$\mathcal{R}_{w,1}^-(f) = \sum_{u \in V} \sum_{v \sim u} \sqrt{w_{uv}} |\min(0, f(v) - f(u))| \quad (3.211)$$

$$= \frac{1}{2} \sum_{u \in V} \sum_{v \sim u} \sqrt{w_{uv}} |f(v) - f(u)| - \frac{1}{2} \sum_{u \in V} \sum_{v \sim u} \sqrt{w_{uv}} (f(v) - f(u)) \quad (3.212)$$

$$= \frac{1}{2} \mathcal{R}_{w,1}(f) - \frac{1}{2} \sum_{u \in V} \sum_{v \sim u} \sqrt{w_{uv}} (f(v) - f(u)) \quad (3.213)$$

$$= \frac{1}{2} \mathcal{R}_{w,1}(f) . \quad (3.214)$$

Finalement, nous retrouvons bien la relation

$$\mathcal{R}_{w,1}^+(f) = \mathcal{R}_{w,1}^-(f) = \frac{1}{2}\mathcal{R}_{w,1}(f) . \quad (3.215)$$

□

À partir de la propriété 3.2.3 (page 64) qui montre le lien entre la dérivée de la fonctionnelle de régularisation $\mathcal{R}_{w,p}$ et le p -Laplacien anisotrope :

$$\frac{1}{2p} \frac{\partial \mathcal{R}_{w,p}(f)}{\partial f(u)} = (\Delta_{w,p}^a f)(u) \quad (3.216)$$

et de la propriété 3.4.4, nous avons immédiatement la propriété suivante.

Propriété 3.4.5. *Les dérivées partielles des fonctionnelles de régularisation $\mathcal{R}_{w,1}^*$ correspondent à la définition du 1-Laplacien anisotrope (2.117) présenté dans le chapitre 2*

$$\frac{1}{2} \frac{\partial \mathcal{R}_{w,1}(f)}{\partial f(u)} = \frac{\partial \mathcal{R}_{w,1}^+(f)}{\partial f(u)} = \frac{\partial \mathcal{R}_{w,1}^-(f)}{\partial f(u)} = (\Delta_{w,1}^a f)(u) . \quad (3.217)$$

La solution du problème (3.185) pour $p = 1$ en un sommet $u \in V$, s'obtient en utilisant cette dernière propriété

$$\frac{\partial \tilde{\mathcal{E}}_3(f, f^0, \lambda)}{\partial f(u)} = \frac{\partial}{\partial f(u)} \left(\frac{1}{2} \mathcal{R}_{w,1}(f) + \lambda g(f^0) f \right) \quad (3.218)$$

$$= \frac{\partial}{\partial f(u)} \left(2\mathcal{R}_{w,1}^+(f) + \lambda g(f^0) f \right) \quad (3.219)$$

$$= \frac{\partial}{\partial f(u)} \left(2\mathcal{R}_{w,1}^-(f) + \lambda g(f^0) f \right) \quad (3.220)$$

$$= (\Delta_{w,1}^a f)(u) + \lambda g(f^0)(u) = 0 . \quad (3.221)$$

En utilisant l'expression du 1-Laplacien anisotrope, nous obtenons l'équation d'Euler-Lagrange suivante

$$c \sum_{v \sim u} \sqrt{w_{uv}} (f(u) - f(v)) |f(u) - f(v)|^{-1} + \lambda g(f^0)(u) = 0 , \quad (3.222)$$

ce qui est équivalent au système suivant

$$f(u) = \frac{c \sum_{v \sim u} \sqrt{w_{uv}} |f(u) - f(v)|^{-1} f(v) - \lambda g(f^0)(u)}{c \sum_{v \sim u} \sqrt{w_{uv}} |f(u) - f(v)|^{-1}} \quad (3.223)$$

où c est une constante. Avec $c = 1$ (3.223) correspond à l'approximation du modèle basé sur la fonctionnelle $\mathcal{R}_{w,1}$. Quand $c = 2$, le système (3.223) correspond à l'approximation du modèle basé sur les fonctionnelles $\mathcal{R}_{w,1}^+$ et $\mathcal{R}_{w,1}^-$.

L'approximation de la solution du système (3.223) en un sommet $u \in V$ se fait par l'algorithme discret

$$\begin{cases} f^{n+1}(u) = \frac{c \sum_{v \sim u} \sqrt{w_{uv}} |f^n(u) - f^n(v)|^{-1} f^n(v) - \lambda g(f^0)(u)}{c \sum_{v \sim u} \sqrt{w_{uv}} |f^n(u) - f^n(v)|^{-1}} \\ f^{(0)}(u) = m(u) \end{cases} \quad (3.224)$$

où $m : V \rightarrow \{0, 1\}$ est une fonction binaire représentant un marqueur initial.

Afin d'éviter une division par zéro du dénominateur, nous utilisons une version régularisée de la valeur absolue telle que pour deux sommets u et v

$$|f(u) - f(v)|_\epsilon = \sqrt{(f(u) - f(v))^2 + \epsilon^2} \quad (3.225)$$

avec $\epsilon > 0$ et $\epsilon \rightarrow 0$.

Du fait de la relaxation du problème, à chaque itération, nous projetons la solution en un sommet $u \in V$ la solution $f^n(u)$ sur l'intervalle $[0, 1]$ telle que

$$f^n(u) = \begin{cases} 0 & \text{si } f^n(u) < 0 \\ 1 & \text{si } f^n(u) \geq 1 \\ f^n(u) & \text{sinon} \end{cases} \quad (3.226)$$

L'algorithme 2 résume l'approximation de la solution du système (3.223). La convergence de l'algorithme est obtenue pour un nombre N d'itérations ou quand $\|f^{n+1} - f^n\|_2 < \tau$

Données : une fonction initiale $f^0 : V \rightarrow [0, 1]$
Résultat : une fonction $f : V \rightarrow \{0, 1\}$ approximant le système (3.223)
Entrées : un graphe pondéré $G = (V, E, w)$, un marqueur initial $m : V \rightarrow \{0, 1\}$

- 1 **pour tous les** sommets $u \in V$ **faire**
- 2 $f^{(0)}(u) = m(u)$
- 3 boucle principale :
- 4 **tant que** $n \neq N$ ou $\|f^{n+1} - f^n\|_2 < \tau$ **faire**
- 5 **pour tous les** sommets $u \in V$ **faire**
- 6 $\beta_{uv}^{f^n} = c\sqrt{w_{uv}}|f^n(u) - f^n(v)|_\epsilon^{-1}$
- 7 $f^{n+1}(u) = \max\left(\min\left(\frac{\sum_{u \sim v} \beta_{uv}^{f^n} f^n(v) - \lambda g(f^0)(u)}{\sum_{u \sim v} \beta_{uv}^{f^n}}, 1\right), 0\right)$

Algorithme 2 : Algorithme discret approximant le système (3.223).

Dans les sections suivantes, nous appliquons nos modèles généraux définis sur des graphes pour la segmentation non locale des images ou le regroupement de nuages de points ou de bases de données. Ces applications sont obtenues en utilisant deux modèles particuliers issus du traitement des images : le modèle de Rudin Osher et Fatemi (ROF) [Rudin *et al.*, 1992] et celui de Chan et Vese (Cv) [Chan et Vese, 2001].

3.4.3 Modèle discret de Rudin, Osher et Fatemi

Le modèle de Rudin, Osher et Fatemi (ROF) [Rudin *et al.*, 1992] (basé sur des équations aux dérivées partielles et la variation totale) est un des modèles les plus connus pour débruiter les images.

Soit $f^0 : \Omega \subset \mathbb{R}^m \rightarrow [0, 1]$ une fonction initiale. La minimisation du modèle de ROF défini dans le domaine continu s'exprime alors de la manière suivante

$$\mathcal{J}_{\text{ROF}}(f, f^0, \lambda) = \int_{\Omega} \|\nabla f\|_1 + \lambda \int_{\Omega} (f(x) - f^0(x))^2 dx . \quad (3.227)$$

Une application intéressante de ce modèle consiste à considérer des images binaires, c'est-à-dire $f^0 \in \{0, 1\}$. Dans ce cas, le terme d'attache aux données se réécrit

$$(f - f^0)^2 = f^2 - 2ff^0 + f^{02} = f - 2ff^0 + f^0 = (1 - 2f^0)f \quad (3.228)$$

et l'énergie \mathcal{J}_{ROF} s'exprime alors

$$\mathcal{J}_{\text{ROF}}(f, f^0, \lambda) = \int_{\Omega} \|\nabla f\|_1 + \lambda \int_{\Omega} g(f^0)(x)f(x) dx \quad (3.229)$$

avec la fonction

$$g(f^0) = (1 - 2f^0) . \quad (3.230)$$

La fonction f recherchée est une fonction binaire $f \in \{0, 1\}$.

Nous pouvons remarquer que ce modèle correspond à un cas particulier défini dans le domaine continu du modèle variationnel discret (3.186). Nous pouvons donc naturellement considérer la version discrète définie sur des graphes du modèle de ROF. Pour une fonction initiale $f^0 : V \rightarrow \{0, 1\}$

$$\min_{f: V \rightarrow \{0,1\}} \left\{ \mathcal{E}_{\text{ROF}}(f, f^0, \lambda) = \mathcal{R}_{w,1}^*(f) + \lambda(1 - 2f^0)f \right\} \quad (3.231)$$

Ce modèle généralise le modèle de ROF pour des graphes arbitraires. Cela permet de traiter n'importe quel type de données pouvant être représentées par un graphe (images ou des données de grande dimension). Notre approche étend également ce modèle pour des schémas de traitements non locaux des images et pour des fonctionnelles de régularisation basées sur des gradients directionnels.

À partir de l'équation d'Euler-Lagrange (3.222) et du système (3.223), nous obtenons, pour le modèle discret de ROF défini sur des graphes, le système suivant

$$f(u) = \frac{c \sum_{v \sim u} \sqrt{w_{uv}} |f(u) - f(v)|^{-1} f(v) - \lambda(1 - 2f^0(u))}{c \sum_{v \sim u} \sqrt{w_{uv}} |f(u) - f(v)|^{-1}} \quad (3.232)$$

où c est une constante. $c = 1$ correspond à l'approximation du modèle basé sur la fonctionnelle $\mathcal{R}_{w,1}$, quand $c = 2$, cela correspond à l'approximation du modèle de ROF basé sur les fonctionnelles $\mathcal{R}_{w,1}^+$ et $\mathcal{R}_{w,1}^-$. L'approximation de la solution se fait alors par l'algorithme suivant. Pour un sommet $u \in V$

$$\begin{cases} f^{n+1}(u) = \frac{c \sum_{v \sim u} \sqrt{w_{uv}} |f^n(u) - f^n(v)|^{-1} f^n(v) - \lambda(1 - 2f^0(u))}{c \sum_{v \sim u} \sqrt{w_{uv}} |f^n(u) - f^n(v)|^{-1}} \\ f^{(0)}(u) = m(u) \end{cases} . \quad (3.233)$$

L'algorithme 3 résume l'approximation de la solution du modèle discret de ROF défini sur des graphes.

La figure 3.31 présente une application du modèle discret de ROF. Dans cet exemple, nous traitons une image en niveaux de gris définie par la fonction f^0 :

Entrées : un graphe pondéré $G = (V, E, w)$, un contour initial $m : V \rightarrow \{0, 1\}$,
une fonction initiale $f^0 : V \rightarrow [0, 1]$

Données : une fonction $l^0 : V \rightarrow \mathbb{R}$.

Résultat : une fonction $f : V \rightarrow \{0, 1\}$ approximant le système (3.223)

- 1 **pour tous les** sommets $u \in V$ **faire**
- 2 $f^{(0)}(u) = m(u)$
- 3 $l^0(u) = \lambda(1 - 2f^0(u))$
- 4 **tant que** $n \neq N$ **ou** $\|f^{n+1} - f^n\|_2 < \tau$ **faire**
- 5 **pour tous les** sommets $u \in V$ **faire**
- 6 $\beta_{uv}^{f^n} = c\sqrt{w_{uv}}|f^n(u) - f^n(v)|^{-1}$
- 7 $f^{n+1}(u) = \max\left(\min\left(\frac{\sum_{u \sim v} \beta_{uv}^{f^n} f^n(v) - l^0(u)}{\sum_{u \sim v} \beta_{uv}^{f^n}}, 1\right), 0\right)$
- 8 **pour tous les** sommets $u \in V$ **faire**
- 9 **si** $f(u) > \frac{1}{2}$ **alors** $f(u) = 1$ **sinon** $f(u) = 0$

Algorithme 3 : Algorithme discret approximant le modèle de ROF (3.231).



(a) Image initiale et contour initial



(b) $n=600$



(c) $n=800$



(d) $n=1200$

FIG. 3.31: Modèle de ROF discret défini sur des graphes appliqué à une image bruitée par un bruit Gaussien de moyenne nulle et de variance égale à 60. Les images (b), (c) et (d) présentent l'évolution du marqueur initial représenté en rouge pour un nombre n d'itérations de l'algorithme 3.

$V \rightarrow \mathbb{R}$. L'image initiale est bruitée par un bruit Gaussien de moyenne nulle et de variance $\sigma = 60$. La figure 3.31a présente l'image initiale avec le contour initial représenté en rouge. Afin de traiter cette image, nous utilisons un graphe grille

non pondéré en 4-voisinage $G_0 = (V, E, g_1)$. Les images suivantes présentées à la figure 3.31 montrent l'évolution de la fonction binaire $f \in \{0, 1\}$ pour un nombre d'itérations n donné avec $\lambda = 1$. Nous pouvons remarquer que le modèle de débruitage de ROF permet d'obtenir les contours des objets désirés.

3.4.4 Modèle discret de Chan et Vese

Un autre modèle très populaire permettant de réaliser une segmentation d'image est le modèle proposé par Chan et Vese (Cv) [Chan et Vese, 2001] basé sur la fonctionnelle de Mumford et Shah [Mumford et Shah, 1989].

Soit $f^0 : \Omega \subset \mathbb{R}^m \rightarrow [0, 1]$ une fonction initiale, le modèle de Cv défini dans le domaine continu s'exprime alors avec la fonctionnelle suivante

$$\mathcal{J}_{Cv}(f, f^0, \lambda) = \int_{\Omega} \|\nabla f\|_1 \quad (3.234)$$

$$+ \lambda \int_{\Omega} \left(f(x)(c_1 - f^0(x))^2 + (1 - f(x))(c_2 - f^0(x))^2 \right) dx \quad (3.235)$$

où la fonction f recherchée est définie par $f : \Omega \subset \mathbb{R}^m \rightarrow \{0, 1\}$ et $c_1, c_2 \in \mathbb{R}$. Ces deux dernières variables peuvent être calculées de manière explicite durant la minimisation. Dans ce cas, la fonctionnelle \mathcal{J}_{Cv} peut se réécrire sous la forme

$$\mathcal{J}_{Cv}(f, f^0, \lambda) = \int_{\Omega} \|\nabla f\|_1 + \lambda \int_{\Omega} g(f^0)(x)f(x) dx \quad (3.236)$$

avec la fonction

$$g(f^0) = (c_1 - f^0)^2 - (c_2 - f^0)^2 . \quad (3.237)$$

Nous pouvons remarquer que ce modèle correspond à un cas particulier défini dans le domaine continu du modèle variationnel discret (3.186). Nous pouvons donc naturellement considérer la version discrète définie sur des graphes du modèle de Cv, pour une fonction initiale $f^0 : V \rightarrow [0, 1]$

$$\min_{f: V \rightarrow \{0,1\}} \left\{ \mathcal{E}_{Cv}(f, f^0, \lambda) = \mathcal{R}_{w,1}^*(f) + \lambda \left((c_1 - f^0)^2 - (c_2 - f^0)^2 \right) f \right\} \quad (3.238)$$

Comme pour le modèle de ROF, ce modèle discret de Cv est défini pour des graphes de topologie arbitraire permettant de traiter n'importe quel type de données pouvant être représenté par un graphe. Notre approche étend également ce modèle pour des schémas de traitements non locaux des images et pour des fonctionnelles de régularisation basées sur des gradients directionnels.

La résolution du modèle discret de Cv se base à nouveau sur notre approche pour la recherche de minimum global avec le problème relaxé (3.185).

Nous obtenons pour le modèle discret de Cv défini sur des graphes, le système suivant en un sommet u

$$f(u) = \frac{c \sum_{v \sim u} \sqrt{w_{uv}} |f(u) - f(v)|^{-1} f(v) - \lambda \left((c_1 - f^0(u)) - (c_2 - f^0(u)) \right)}{c \sum_{v \sim u} \sqrt{w_{uv}} |f(u) - f(v)|^{-1}} \quad (3.239)$$

où c est une constante. $c = 1$ correspond à l'approximation du modèle basé sur la fonctionnelle $\mathcal{R}_{w,1}$. $c = 2$ correspond à l'approximation du modèle de Cv basé sur les fonctionnelles $\mathcal{R}_{w,1}^+$ et $\mathcal{R}_{w,1}^-$. L'approximation de la solution du modèle discret de Cv se fait alors par l'algorithme suivant. En un sommet $u \in V$, nous avons

$$\begin{cases} f^{n+1}(u) = \frac{c \sum_{v \sim u} \sqrt{w_{uv}} |f^n(u) - f^n(v)|^{-1} f^n(v) - \lambda \left((c_1 - f^0(u))^2 - (c_2 - f^0(u))^2 \right)}{c \sum_{v \sim u} \sqrt{w_{uv}} |f^n(u) - f^n(v)|^{-1}} \\ f^{(0)}(u) = m(u) \end{cases} \quad (3.240)$$

Entrées : un graphe pondéré $G = (V, E, w)$, un contour initial $m : V \rightarrow \{0, 1\}$, une fonction initiale $f^0 : V \rightarrow [0, 1]$

Données : une fonction $l^0 : V \rightarrow \mathbb{R}$.

Résultat : une fonction $f : V \rightarrow \{0, 1\}$ approximant le système (3.223)

- 1 **pour tous les** sommets $u \in V$ **faire** $f^{(0)}(u) = m(u)$
- 2 $c_1 = \text{moyenne}(f^0)$ si $m = 1$; $c_2 = \text{moyenne}(f^0)$ si $m = 0$
- 3 **tant que** $n \neq N$ ou $\|f^{n+1} - f^n\|_2 < \tau$ **faire**
- 4 **pour tous les** sommets $u \in V$ **faire**
- 5 $\beta_{uv}^{f^n} = c \sqrt{w_{uv}} |f^n(u) - f^n(v)|_c^{-1}$
- 6 $\gamma^{f^n}(u) =$
 $\max \left(\min \left(\frac{\sum_{u \sim v} \beta_{uv}^{f^n} f^n(v) - \lambda \left((c_1 - f^0(u))^2 - (c_2 - f^0(u))^2 \right)}{\sum_{u \sim v} \beta_{uv}^{f^n}}, 1 \right), 0 \right)$
- 7 $f^{n+1}(u) = \gamma^{f^n}(u)$
- 8 $c_1 = \text{moyenne}(f^0)$ si $\gamma^{f^n} \leq 0.5$; $c_2 = \text{moyenne}(f^0)$ si $\gamma^{f^n} > 0.5$
- 9 **pour tous les** sommet $u \in V$ **faire**
- 10 **si** $f(u) > \frac{1}{2}$ **alors** $f(u) = 1$ **sinon** $f(u) = 0$

Algorithme 4 : Algorithme discret approximant le modèle de Cv (3.238).

L'algorithme 4 résume l'approximation de la solution du modèle discret Cv. Afin d'améliorer les performances de l'algorithme, une alternative possible à la mise à jour (à chaque itération) des variables c_1 et c_2 est de la considérer uniquement au bout de quelques itérations.

Dans la suite, nous présentons des applications du modèle de Cv défini sur des graphes pour de la segmentation d'images (locale et non locales) et du partitionnement de données (nuages de points et bases de données). Les expérimentations suivantes montrent les potentialités de notre méthodologie, en particulier pour le traitement des données de grande dimension définies dans des espaces non uniformes et la segmentation non locale des images.

Segmentation d'images

Les figures 3.32 et 3.33 montrent deux exemples de segmentation d'images utilisant le modèle de Cv défini sur des graphes.

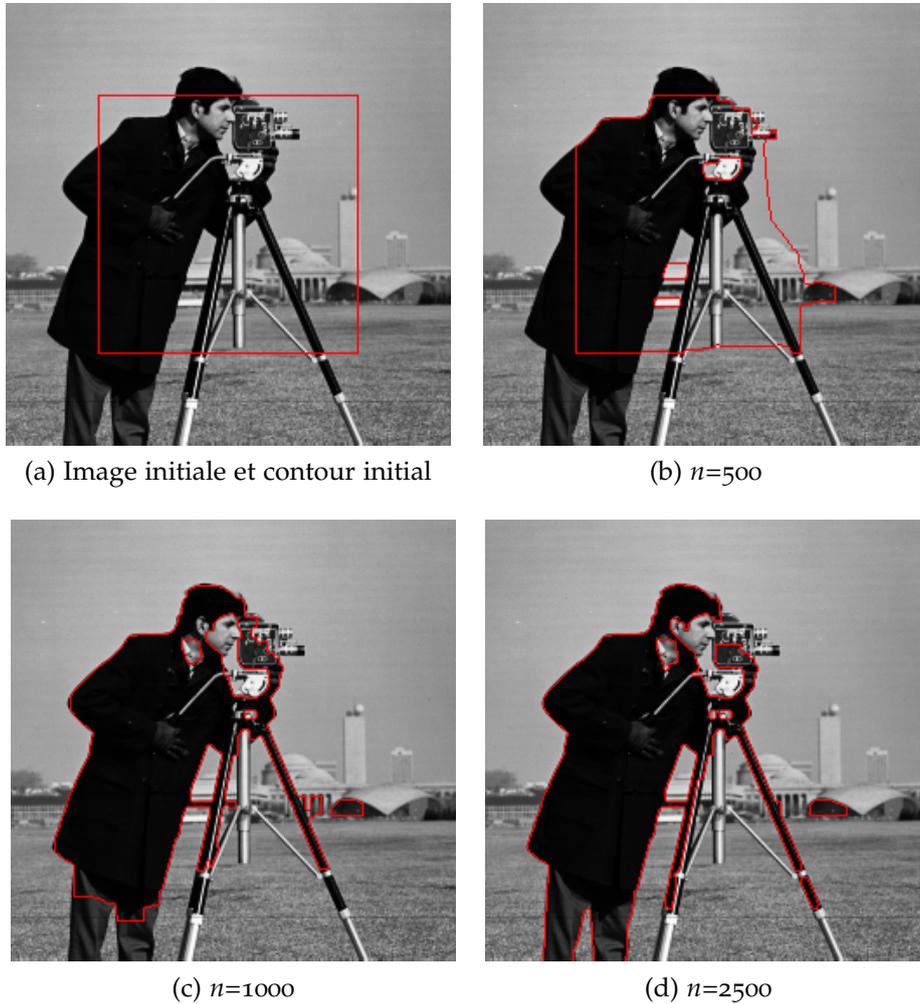


FIG. 3.32: Modèle discret de C_v défini sur des graphes appliqué à la segmentation d'images. Les images (b), (c) et (d) présentent l'évolution de la courbe initiale représentée en rouge pour un nombre n d'itérations de l'algorithme 4.

Les images initiales de ces deux figures sont définies comme des fonction $f^0 : V \rightarrow \mathbb{R}$ et les marqueurs initiaux sont représentés en rouge. Ces deux figures montrent les évolutions des contours en fonction du nombre d'itérations n de l'algorithme 4, où les contours sont obtenus par seuillage de la fonction $f : \partial\{f > 0.5\}$.

La figure 3.32 présente un exemple de segmentation basé sur un graphe grille en 4-voisinage non pondéré $G_0 = (V, E, g_1)$. Les autres images de la figure 3.32 montrent l'évolution de la segmentation. Notre algorithme permet d'obtenir un résultat satisfaisant en un nombre faible d'itérations.

La figure 3.33 montre deux segmentations d'une image texturée et bruitée par un bruit Gaussien de variance égale à 30. Cette figure compare les résultats obtenus avec des configurations locales (première ligne) et des configurations non locales avec des patches. Les résultats de la première ligne sont obtenus avec un graphe grille

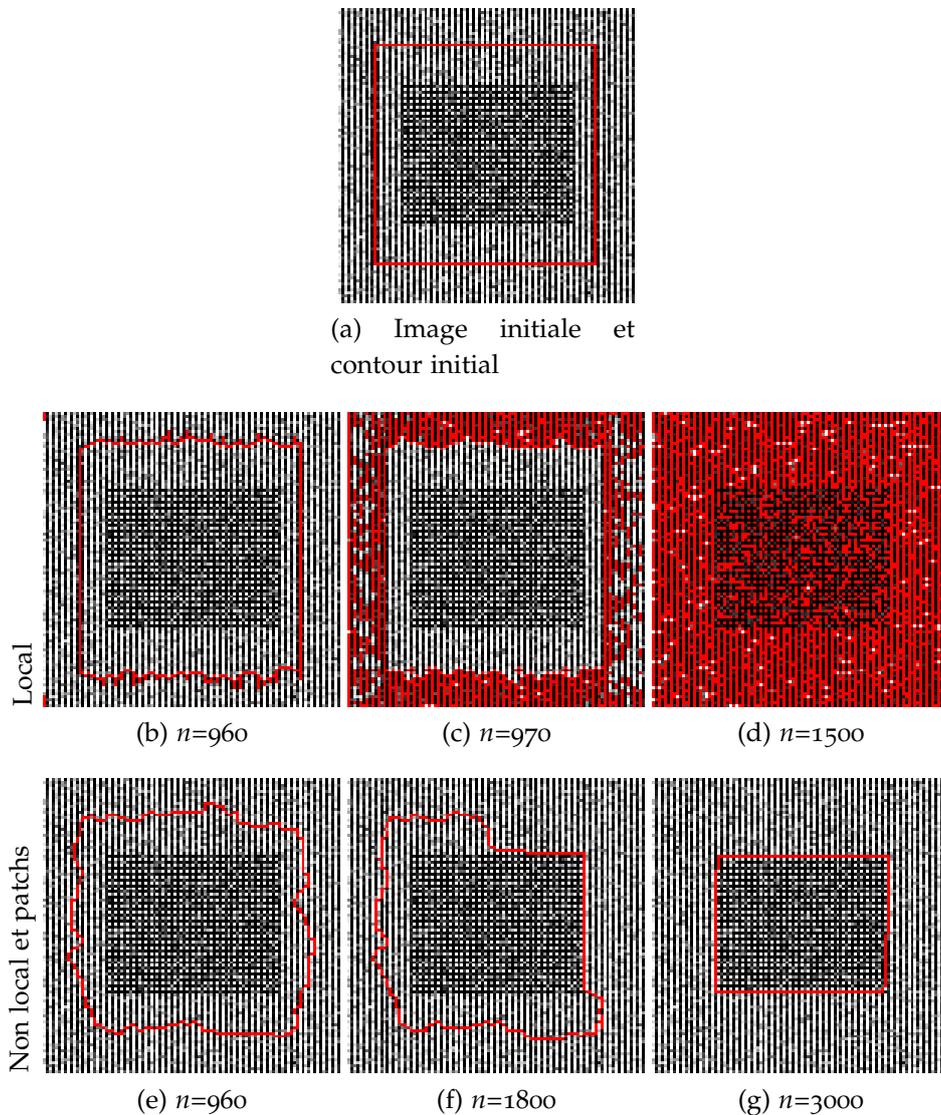


FIG. 3.33: Modèle de Cv discret défini sur des graphes appliqué à la segmentation non locale d'images. (a) : Image initiale et marqueur initial. À la première ligne : résultats avec une configuration locale. À la seconde ligne : résultats avec une configuration non locale et patches. De gauche à droite évolution de la segmentation en fonction du nombre n d'itérations.

en 4-voisinage non pondéré alors que ceux de la seconde ligne sont obtenus avec un graphe G_3 pondéré par la fonction de poids g_4 utilisant des vecteurs d'attributs définis par des patches de taille 5×5 ($\mathcal{F}_2(f^0, \cdot)$).

Cette expérimentation montre clairement le bénéfice des configurations non locales pour détecter les motifs répétitifs contenus dans l'image. Cela permet de segmenter la sous région alors que la segmentation obtenue avec les schémas locaux n'est pas correcte. Nous constatons que le nombre d'itérations, dans le cas de la segmentation non locale est plus élevé que dans le cas local. Cela provient simplement

du degré plus élevé des sommets.

Partitionnement de données

Un des apports de nos approches est que nos modèles sont basés sur des graphes. Cet avantage permet de traiter n'importe quel type de données même celles définies sur des domaines irréguliers ou de grande dimension.

Les figures 3.34 et 3.35 illustrent cet avantage et montrent des applications inhabituelles du modèle de segmentation de C_v : le partitionnement de nuages de points de base de données.

La figure 3.34 montrent un partitionnement réalisé à partir d'un nuage de points (deux Gaussiennes). Afin de traiter ces données un 20- NNG pondéré par la fonction g_2 (figure 3.34b) est construit à partir des données initiales. Les marqueurs (figure 3.34c) sont initialisés de manière aléatoire (en rouge et en noir).

Les figures (d) et (e) montrent l'évolution du partitionnement en fonction du nombre n d'itérations de l'algorithme 4. Enfin, la partition finale (figure (f)) est obtenue à convergence de l'algorithme. Nous pouvons remarquer finalement le résultat satisfaisant de la partition finale.

La figure 3.35 montre un second exemple de partitionnement mais cette fois sur une base de données d'images. Les données proviennent de la base de l'USPS et ont été sélectionnées de manière aléatoire parmi les chiffres « 0 » et « 1 » (figure 3.35a). Un 20- NNG pondéré par la fonction de poids g_2 est construit à partir des données initiales (figure 3.35b).

L'initialisation des marqueurs est faite de manière aléatoire. La figure 3.35c montre cette initialisation où $m(\cdot) = 0$ et $m(\cdot) = 1$ correspondent aux partitions initiales. Les figures 3.35d et 3.35e montrent l'évolution du partitionnement en fonction du nombre n d'itérations de l'algorithme approximant la solution. La figure 3.35f montre le partitionnement final où $f(\cdot) = 0$ et $f(\cdot) = 1$ correspondent aux partitions trouvées.

Nous pouvons remarquer que les résultats sont satisfaisants malgré l'utilisation d'une simple distance Euclidienne pour comparer les sommets du graphe.

3.4.5 Conclusion

Dans cette section, nous nous sommes intéressés à une seconde classe de modèles variationnels discrets basés sur la régularisation proposée au début de ce chapitre.

Cette classe de modèles correspond à une classe de problèmes d'optimisation globale de fonctionnelles de régularisation non convexes. Ces fonctionnelles sont à la base de nombreuses applications telles que la segmentation par contours actifs globaux.

Notre approche de l'optimisation globale peut être considérée comme la version discrète des approches définies dans le domaine continu, mais également comme une extension sur des graphes de ces modèles.

Pour trouver un minimum global à ces problèmes qui ne sont pas convexes, nous avons utilisé les propriétés de la formule de la co-aire définie sur des graphes

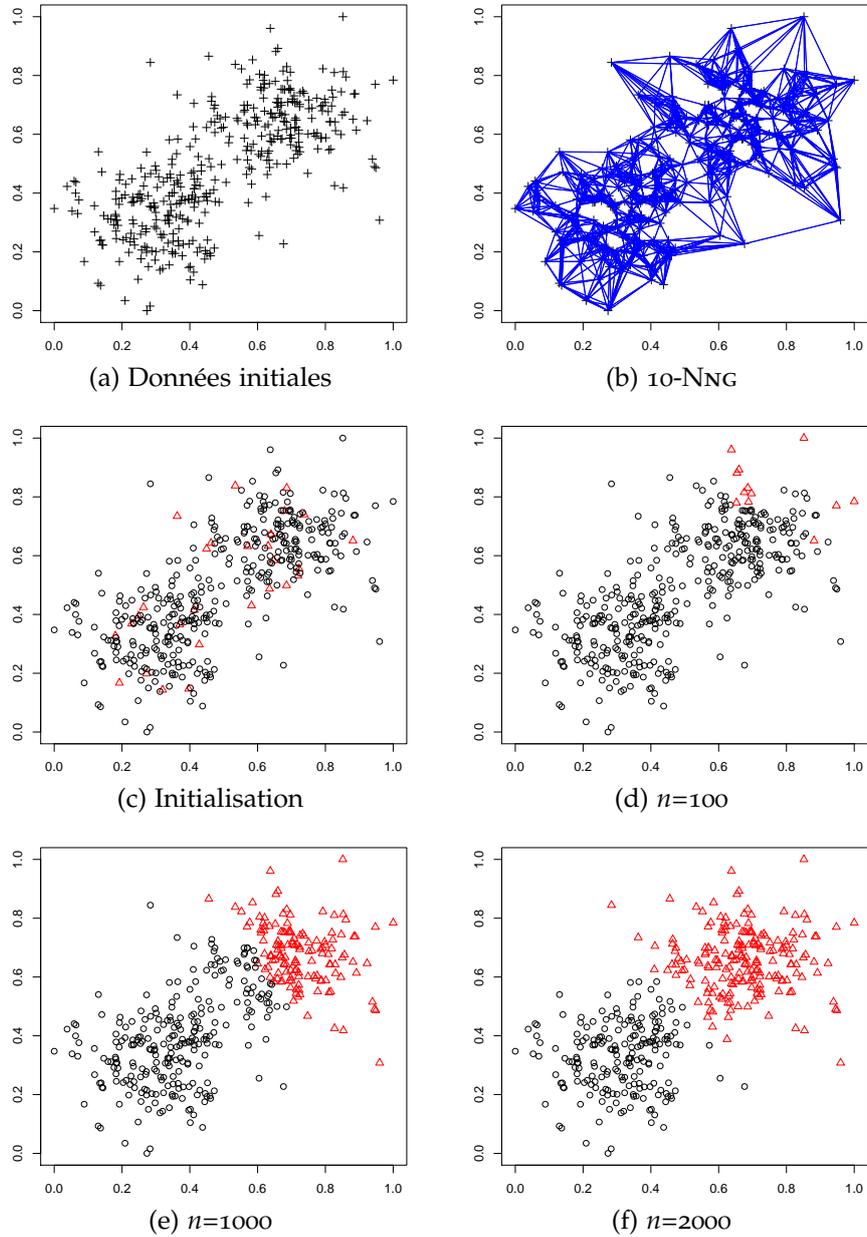


FIG. 3.34: Partitionnement de nuages de points (deux Gaussiennes) avec le modèle discret de C_v défini sur des graphes. (a) : les données initiales ($f^0 : V \rightarrow \mathbb{R}^2$). (b) : 10-NNG associé aux données initiales. (c) : initialisation aléatoire des marqueurs. (d) et (e) évolution du partitionnement en fonction du nombre n d'itérations (f) : partitionnement final.

et une méthode discrète de relaxation afin de les rendre convexes. À partir de la formulation du p -Laplacien anisotrope, nous avons proposé un algorithme permettant d'approximer la solution de ces problèmes d'optimisation globale.

Notre formulation générale nous a permis de transcrire des modèles continus

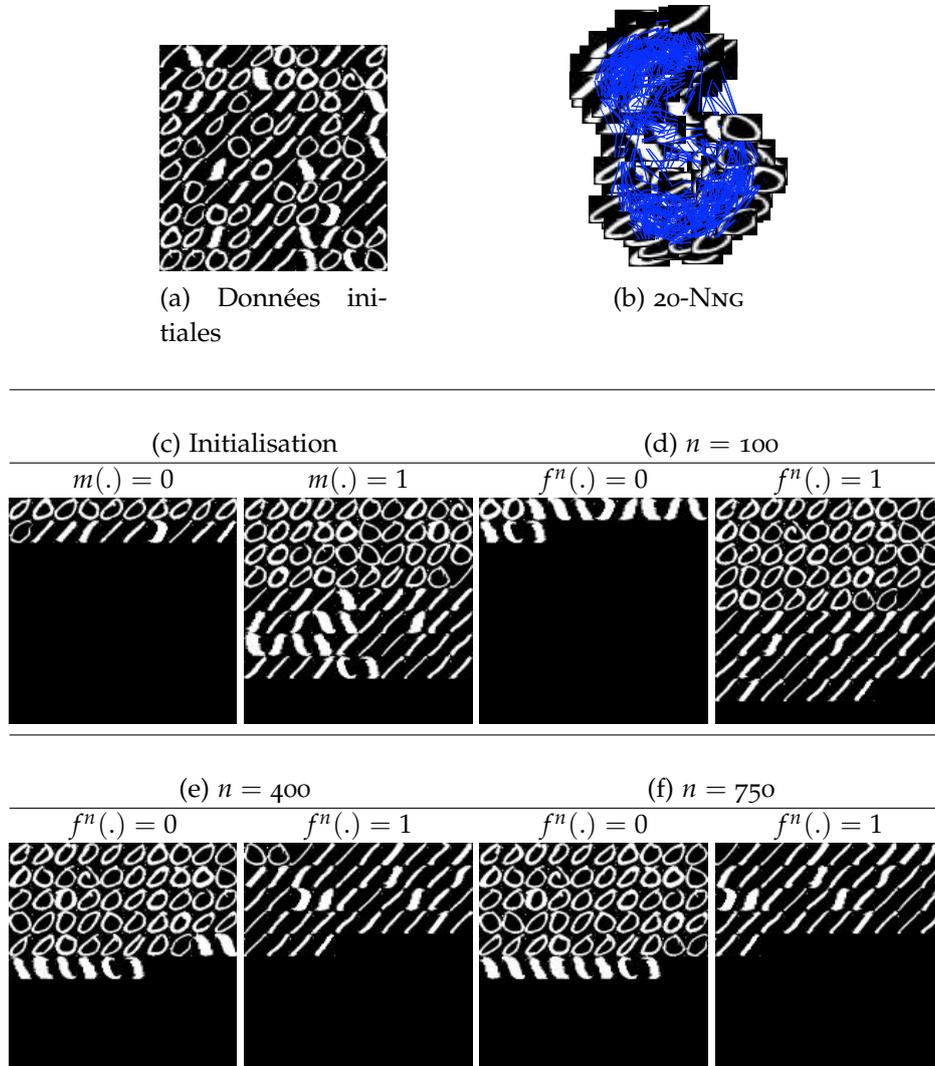


FIG. 3.35: Partitionnement de base de données (base de l'USPS) avec le modèle de Cv discret défini sur des graphes. (a) : données initiales ($f^0 : V \rightarrow \mathbb{R}^{256}$). (b) : 20-NNG associé aux données initiales. (c) : initialisation où $m(\cdot) = 0$ et $m(\cdot) = 1$ représentent les partitions initiales. (d) et (e) : évolution du partitionnement. en fonction du nombre n d'itérations. (f) : partitionnement final.

(issus du traitement des images) vers des schémas discrets tels que le modèle de RoF ou celui de Cv. À partir de ces modèles, nous avons montré les potentialités de notre méthodologie pour réaliser de la segmentation non locale d'images ou encore du partitionnement de données. Ce dernier point ouvre des pistes intéressantes pour des domaines tels que l'apprentissage ou la fouille de données. Nous pouvons envisager de proposer de nouveaux algorithmes de classification. En effet, nous pouvons constater que le modèle discret de Cv que nous avons proposé correspond à l'algorithme des k moyennes dans sa version régularisée.

Morphologie mathématique

Sommaire

4.1	Introduction	118
4.2	Dilatation et érosion	120
4.3	Morphologie mathématique par équations aux différences partielles	122
4.4	Expérimentations	132
4.5	Conclusion	143

Résumé

La morphologie mathématique offre un large choix d'opérateurs permettant de résoudre de nombreuses applications dans les domaines du traitement des images et de la vision par ordinateur. Les opérations morphologiques peuvent être formulées en terme d'opérations sur des ensembles ou en terme d'équations aux dérivées partielles.

Dans ce chapitre, nous introduisons une nouvelle formulation inspirée de la morphologie mathématique formalisée à partir d'équations aux différences partielles définies sur des graphes et des opérateurs discrets directionnels introduits dans le premier chapitre.

Cette nouvelle formulation est une transcription des modèles définis à partir d'équations aux dérivées partielles vers des équations aux différences partielles. Notre nouvelle approche de la morphologie mathématique permet de retrouver, sous certaines conditions, les modèles définis dans le domaine discret et ceux définis dans le domaine continu. Nous introduisons également une nouvelle classe d'opérateurs morphologiques adaptatifs pour le traitement des images en considérant des configurations non locales. Nous étendons également la morphologie basée sur les équations aux dérivées partielles et introduisons de nouveaux outils pour le traitement de données discrètes arbitraires.

Mots clés

Morphologie mathématique, graphes pondérés arbitraires, traitements morphologiques non locaux des images, traitements morphologiques de données de grande dimension, données discrètes.

Contenu du chapitre

- La section 4.1 introduit ce chapitre en rappelant brièvement les grands concepts de la morphologie mathématique classique et celle formulée par les équations aux dérivées partielles. Cette section décrit également comment notre approche se place vis-à-vis de ces deux formulations et explicite nos principales contributions.
- La section 4.2 rappelle les approches morphologiques classiques et celles basées sur les équations aux dérivées partielles qui permettent de définir les opérations fondamentales de la morphologie mathématique que sont la dilatation et l'érosion
- La section 4.3 présente notre approche basée sur les équations aux différences partielles et montre les liens entre notre formulation et les approches de la morphologie mathématique définies sur des graphes et celles définies dans le domaine continu.
- La section 4.4 présente des expérimentations et l'application de notre formalisme inspiré de la morphologie mathématique pour des traitements non locaux des images et de données arbitraires.
- La section 4.5 conclut ce chapitre et donne des perspectives par rapport à l'approche proposée.

Publications associées à ce chapitre

Conférences internationales avec comités de lecture et actes

- [1] Olivier LÉZORAY, Vinh-Thong TA et Abderrahim ELMOATAZ : Manifold and data filtering on graphs. *Dans Proceedings of International Symposium on Methodologies for Intelligent Systems, International Workshop on Topological learning*, pages 19 – 28, 2009.
- [2] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Nonlocal graph morphology. *Dans Proceedings of 9th International Symposium on Mathematical Morphology (ISMM), Abstract Book*, pages 5 – 9, 2009.
- [3] Vinh-Thong TA, Olivier LÉZORAY et Abderrahim ELMOATAZ : Nonlocal morphological levelings by partial difference equations over weighted graphs. *Dans Proceedings of 19th International Conference on Pattern Recognition (ICPR)*, 2008.
- [4] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Partial difference equations over graphs : morphological processing of arbitrary discrete data. *Dans Proceedings of 10th European Conference on Computer Vision (ECCV)*, LNCS 5304, pages 668 – 680. Springer, 2008.
- [5] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Partial difference equations on graphs for mathematical morphology operators over images and manifolds. *Dans Proceedings of 15th IEEE International Conference on Image Processing (ICIP)*, pages 801 – 804, 2008. **Prix IBM du meilleur article étudiant.**
- [6] Abderrahim ELMOATAZ, Olivier LÉZORAY, Sébastien BOUGLEUX et Vinh-Thong TA : Unifying local and nonlocal processing with partial difference operators on weighted graphs. *Dans Proceedings of the International Workshop on Local*

and Non-Local Approximation in Image Processing (LNLA), pages 11 – 26, 2008.

Article invité.

- [7] Olivier LÉZORAY, Abderrahim ELMOATAZ et Vinh-Thong TA : Learning graph neighborhood topological order for image and manifold morphological processing. *Dans Proceedings of 8th IEEE International Conference on Computer and Information Technology (CIT)*, pages 350 – 355, 2008.

Conférences nationales avec comités de lecture et actes

- [1] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Morphologie mathématique sur graphes pondérés de topologie arbitraire — traitements non locaux d’images et de données multi-variées. *Dans Actes du groupe de recherche et d’études du traitement du signal et des images (GRETSI)*, 2009.

Journées nationales sans comités de lecture ni actes

- [1] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Morphologie mathématique sur graphes pondérés de topologie arbitraire. *Dans 32ième journée de l’International Society for Stereology – France (Iss), école des mines, Paris*, 2009.

Rapports techniques et articles soumis

- [1] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Mathematical morphology and eikonal equations on graphs for nonlocal image and data processing. Rapport technique, GREYC CNRS UMR 6072 - Université de Caen Basse-Normandie – ENSICAEN - hal-00365431, mars 2009. Soumis à IEEE Transactions on Image Processing.

4.1 Introduction

La morphologie mathématique est une approche non linéaire du traitement d'image qui est beaucoup utilisée dans de nombreuses applications, par exemple l'analyse de formes et de textures, le traitement d'images biomédicales, la reconnaissance de documents ou encore les méthodes d'analyse multi échelles [Serra, 1982, Soille, 2002, Najman et Talbot, 2008].

La morphologie mathématique a été initialement développée par Matheron [Matheron, 1967] et Serra [Serra, 1967]. Elle repose sur la structure fondamentale de treillis complet. Un treillis complet L est défini de la manière suivante [Ronse, 1990] :

- c'est un ensemble non vide ;
- qui possède une relation d'ordre ;
- pour tout sous ensemble fini K de L , il existe un supremum et un infimum.

Dans ce contexte, les images sont modélisées comme étant des applications de leur domaine de définition Ω vers le treillis complet L . Cela permet ainsi de définir des opérations morphologiques pour n'importe quel type d'image à condition de définir une relation d'ordre consistante. Avec cette approche, les opérations morphologiques sont représentées comme des applications entre treillis complets combinées avec des motifs de références définis dans Ω et appelés éléments structurants.

Les deux opérations fondamentales en morphologie mathématique sont la dilatation et l'érosion. Elles constituent une base permettant de former de nombreuses autres opérations morphologiques telles que l'ouverture ou la fermeture, les filtres par reconstruction ou encore les nivellements [Heijmans, 1994, Soille, 2002]. Une des conséquences de la notion de treillis complet pour la définition des opérations morphologiques est que leurs implantations se font généralement directement dans le domaine discret.

Une alternative à ces dernières formulations repose sur l'utilisation d'équations aux dérivées partielles non linéaires. C'est ainsi que des travaux ont transposé les opérations de dilatation et d'érosion dans un cadre formel utilisant la théorie espace échelle (*scale space theory*). Ces deux opérations sont alors définies comme étant des opérations multi échelles élémentaires définies dans le domaine continu [van den Boomgaard et Smeulders, 1994, Brockett et Maragos, 1994, Alvarez *et al.*, 1993].

Même si ce cadre théorique basé sur les équations aux dérivées partielles ne permet pas de retrouver l'ensemble des propriétés morphologiques, il possède néanmoins plusieurs avantages.

- Il permet une meilleure modélisation mathématique et montre des liens avec des phénomènes physiques.
- De meilleurs résultats sont obtenus pour des éléments structurants qui sont difficilement représentables sur une grille discrète.
- Ces approches permettent une précision sous pixellaire.
- Elles peuvent être adaptatives en introduisant un terme local qui contrôle la vitesse d'évolution du processus morphologique ou alors définies comme étant des opérateurs visqueux [Breuß *et al.*, 2007, Maragos et Vachier, 2008].

Pour être applicables aux images, les approches basées sur les équations aux dérivées partielles sont généralement discrétisées et de nombreux schémas numériques ont été développés pour approximer leurs solutions.

Nous pouvons mentionner des schémas numériques très utilisés dans la réso-

lution des équations aux dérivées partielles liées aux processus de dilatation et d'érosion : les schémas de Osher et Sethian, le schéma de Rouy et Tourin, des schémas d'ordre supérieur ou encore des méthodes de transport. Le lecteur intéressé pourra se référer aux articles suivants ainsi qu'à leurs références associés [Osher et Sethian, 1988, Sethian, 1999b, Rouy et Tourin, 1992, Siddiqi *et al.*, 1997, Breuß et Weickert, 2006, Pizarro *et al.*, 2009]

La morphologie mathématique est bien définie pour le traitement des images binaires et en niveaux de gris, mais, quelque soit la formulation choisie, l'extension pour le traitement de données multi variées n'est pas évidente. En effet, la morphologie mathématique classique requiert la définition d'un treillis complet et il n'existe pas d'ordre naturel pour les données vectorielles. Plusieurs ordres ont été proposés dans la littérature (le lecteur intéressé pourra se référer à [Aptoula et Lefèvre, 2007] pour une revue complète). Nous pouvons noter que ces ordres sont généralement créés pour résoudre des problèmes liés à des types d'images particulières telles que les images de couleurs [Angulo, 2005, Angulo, 2009] ou des images de tenseurs de diffusion d'imagerie par résonance magnétique [Burgeth *et al.*, 2007]. Le traitement de données multi variées de grande dimension n'est également pas évidente avec les approches basées sur les équations aux dérivées partielles car la discrétisation spatiale de domaine non uniforme peut être difficile.

Le traitement de données discrètes de grande dimension par morphologie mathématique est peu étudié et peu de travaux s'intéressent à ce problème. Nous pouvons néanmoins évoquer des travaux récents basés sur une méthode de réduction de dimension non linéaire permettant de traiter morphologiquement des données multi variées arbitraires [Lézoray *et al.*, 2009a].

Dans le cadre du traitement des images, nous pouvons également remarquer que ces deux formulations (discrète et continue) ne considèrent que les interactions locales des images alors que les schémas non locaux deviennent de plus en plus populaires dans la communauté du traitement des images. En effet, comme nous l'avons montré dans le chapitre précédent, le traitement d'images basé sur des schémas non locaux permet de mieux capturer des structures complexes telles que les structures fines et répétitives.

L'utilisation de telles méthodes pour le traitement morphologique des images n'a jamais été, à notre connaissance, étudié.

Finalement, la morphologie mathématique a déjà été définie avec des graphes. Le lecteur intéressé pourra, par exemple, se référer aux articles suivants [Heijmans *et al.*, 1992, Postaire *et al.*, 1993, Falcão *et al.*, 2004, Meyer et Lerallut, 2007, Cousty *et al.*, 2009b] et aux références associées.

Dans ce chapitre, à partir des opérateurs définis dans le chapitre 2 et en se basant sur les mêmes idées que pour la transcription de modèles variationnels continus, nous proposons d'adapter les opérations morphologiques basée sur les équations aux dérivées partielles dans un cadre discret en exploitant le formalisme des équations aux différences partielles définies sur des graphes.

Contributions du chapitre

Pour transcrire les opérations de dilatation et d'érosion définies par des équations aux dérivées partielles, nous utilisons les opérateurs différentiels directionnels

introduits dans le premier chapitre. Cela nous permet de définir une nouvelle formulation de ces opérateurs basées sur des équations aux différences partielles. La formulation que nous proposons peut être considéré comme de la morphologie mathématique adaptative utilisant des processus de diffusion. De ce fait, comme pour le cas des équations aux dérivées partielles, notre formulation ne permet pas de retrouver l'ensemble des propriétés de la morphologie classique mais permet d'introduire de nouveaux concepts dans le cadre du traitement des images et de données discrètes de grande dimension.

Les contributions de notre approche sont les suivantes.

- Notre approche est générale. En effet, dans certains cas particuliers, les schémas numériques que nous proposons permettent de retrouver des schémas connus de la littérature tels que le schéma numérique de Osher et Sethian mais également la formulation de la morphologie mathématique définie sur des graphes.
- Notre approche unifie schémas locaux et non locaux dans une même formulation. Cette unification permet, dans le cadre du traitement des images, d'introduire une nouvelle classe opérations morphologiques basées sur les configurations non locales.

4.2 Dilatation et érosion

Nous rappelons ici la définition des opérations de dilatation et d'érosion du point de vue de la morphologie mathématique algébrique et celui basé sur les équations aux dérivées partielles.

Pour plus de détails sur ces approches, le lecteur intéressé pourra se référer aux ouvrages et articles suivants [Serra, 1982, Serra, 1988, Sapiro *et al.*, 1993, Heijmans, 1994, Brockett et Maragos, 1994, Soille, 2002, Guichard *et al.*, 2007, Sapiro, 2001] ainsi qu'à leurs références associées.

En considérant la morphologie fonctionnelle, c'est-à-dire en utilisant un élément structurant non planaire ou « fonction structurante », une dilatation $\delta : \mathbb{R}^m \rightarrow \mathbb{R}^m$ et une érosion $\varepsilon : \mathbb{R}^m \rightarrow \mathbb{R}^m$ sont généralement formulées de la manière suivante [Serra, 1988]

$$\delta_s(f)(x) = (f \oplus s)(x) = \sup\{f(x - y) + s(y) : y \in B\}, \quad x \in \mathbb{R}^m \quad (4.1)$$

$$\varepsilon_s(f)(x) = (f \ominus s)(x) = \inf\{f(x + y) - s(y) : y \in B\}, \quad x \in \mathbb{R}^m. \quad (4.2)$$

La fonction traitée $f : \mathbb{R}^m \rightarrow \mathbb{R}^m$ représente un signal m dimensionnel avec $m = 1, 2, 3, \dots$ et la fonction $s : B \rightarrow \mathbb{R}$ est une fonction structurante avec un support $B \subseteq \mathbb{R}^m$.

Pour des raisons de clarté dans les formules, nous noterons également l'opération de dilatation δ par \oplus et l'opération d'érosion ε par \ominus .

Lorsque le support de la fonction structurante s est un ensemble compact, symétrique, convexe et défini comme constant et égal à zéro tel que $s : B \rightarrow \{0\}$ avec

$$s(x) = \begin{cases} 0 & \text{si } x \in B \\ -\infty & \text{si } x \notin B \end{cases} \quad (4.3)$$

alors, la fonction structurante est dite « planaire » (et est appelé élément structurant). Nous obtenons la définition de la morphologie mathématique planaire. Ainsi, la dilatation et l'érosion de la fonction f par l'élément structurant B sont définies par

$$\delta_B(f)(x) = (f \oplus s)(x) = \sup\{f(x - y) : y \in B\}, \quad x \in \mathbb{R}^m \quad (4.4)$$

$$\varepsilon_B(f)(x) = (f \ominus s)(x) = \inf\{f(x + y) : y \in B\}, \quad x \in \mathbb{R}^m. \quad (4.5)$$

Dans ce chapitre, nous nous restreignons à considérer uniquement les opérations de dilatation et d'érosion planaires.

4.2.1 Approche basée sur les équations aux dérivées partielles

L'idée de la morphologie mathématique définie par des équations aux dérivées partielles est de voir les opérations de dilatation et d'érosion d'un point de vue multi échelle [Brockett et Maragos, 1992,?, Brockett et Maragos, 1994, Guichard *et al.*, 2007]. Afin de simplifier les notations, nous prendrons le cas d'une fonction en deux dimensions définie comme $f : \mathbb{R}^2 \rightarrow \mathbb{R}$.

Soit K une fonction structurante de \mathbb{R}^2 unitaire utilisée comme noyau pour générer des dilatations et des érosions. Mettre à l'échelle les valeurs et le support de la fonction structurante K par un paramètre d'échelle $t \geq 0$ correspond à produire une famille de fonctions structurantes multi échelles définies comme

$$K_t(x, y) = \begin{cases} tK(x/t), & \text{pour } t > 0 \\ -\infty, & \text{sinon} \end{cases}, \quad (4.6)$$

avec la condition initiale à $t = 0$, $K_0(0, 0) = 0$.

Le remplacement de la fonction structurante s dans les équations (4.1) et (4.2) par le noyau K_t amène à la définition de la dilatation et de l'érosion multi échelles de la fonction f par K_t comme étant

$$\delta_K(f)(x, y, t) = f \oplus K_t(x, y) \quad (4.7)$$

$$\varepsilon_K(f)(x, y, t) = f \ominus K_t(x, y) \quad (4.8)$$

avec comme conditions initiales, à $t = 0$, $\delta_K(f)(x, y, 0) = \varepsilon_K(f)(x, y, 0) = f(x, y)$.

Si la famille des fonctions K correspond à des fonctions structurantes planaires

$$K(x, y) = \begin{cases} 0, & \text{si } (x, y) \in B \\ -\infty, & \text{sinon} \end{cases} \quad (4.9)$$

et que le support B est défini comme étant des boules unitaires $B_q = \{(x, y) : \|(x, y)\|_q \leq 1\}$ induites par la norme \mathcal{L}_q , avec $q = 1, 2, \dots, \infty$, alors les équations aux dérivées partielles permettant de générer la dilatation et l'érosion planaires multi échelles de la fonction f par B_q sont

$$\frac{\partial \delta(f)}{\partial t} = + \|\nabla \delta\|_q = + \|\nabla f\|_q \quad (4.10)$$

$$\frac{\partial \varepsilon(f)}{\partial t} = - \|\nabla \varepsilon\|_q = - \|\nabla f\|_q \quad (4.11)$$

où ∇ désigne l'opérateur nabla de la fonction f . Les conditions initiales de ces équations d'évolution sont $\delta(f)(x, y, 0) = \varepsilon(f)(x, y, 0) = f(x, y)$.

Pour des valeurs spéciales de q , nous retrouvons les éléments structurants suivants pour le cas de la dilatation (les mêmes équations sont applicables pour le cas de l'érosion)

$$q = 1, \quad \frac{\partial \delta(f)}{\partial t} = \|\nabla f\|_1 = \left| \frac{\partial f}{\partial x} \right| + \left| \frac{\partial f}{\partial y} \right|, \quad B = \text{un carré} \quad (4.12)$$

$$q = 2, \quad \frac{\partial \delta(f)}{\partial t} = \|\nabla f\|_2 = \sqrt{\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2}, \quad B = \text{un disque} \quad (4.13)$$

$$q = \infty, \quad \frac{\partial \delta(f)}{\partial t} = \|\nabla f\|_\infty = \max \left\{ \left| \frac{\partial f}{\partial x} \right|, \left| \frac{\partial f}{\partial y} \right| \right\}, \quad B = \text{un losange} . \quad (4.14)$$

Les équations d'évolution de la dilatation et de l'érosion peuvent créer des discontinuités dans les dérivées. Afin de palier les problèmes de « chocs » (*shocks*) engendrés par ces équations de type hyperbolique, les schémas numériques utilisés pour approximer la solution de ces équations aux dérivées partielles utilisent la notion de solution de viscosité. Le lecteur intéressé par ce type de solution peut se référer à [Crandall *et al.*, 1992, Osher et Sethian, 1988].

4.3 Morphologie mathématique par équations aux différences partielles

Dans cette section, nous introduisons notre approche de la dilatation et de l'érosion basée sur les équations aux différences partielles définies sur des graphes. Nous nous intéressons également à une classe particulière de filtre morphologique : les nivellements.

À partir des équations d'évolution des opérations de dilatation et d'érosion définies par les équations aux dérivées partielles (4.10) et (4.11), notre but est de définir les versions discrètes de ces définitions. Pour cela, nous utilisons les définitions des opérateurs différentiels directionnels présentés dans le chapitre 2.

Dans cette section, nous allons montrer que la version définie sur des graphes basée sur les équations d'évolution (4.10) et (4.11), peuvent s'exprimer de la manière suivante avec des équations aux différences partielles.

Soit $G = (V, E, w)$ un graphe pondéré et f^0 une fonction initiale définie dans l'espace $\mathcal{H}(V)$. Alors, la dilatation et l'érosion de la fonction f^0 sont

$$\frac{\partial \delta(f)}{\partial t} = \frac{\partial f}{\partial t} = +\|\nabla_w^+ f\|_q \quad (4.15)$$

$$\frac{\partial \varepsilon(f)}{\partial t} = \frac{\partial f}{\partial t} = -\|\nabla_w^- f\|_q . \quad (4.16)$$

Les conditions initiales sont, à $t = 0$, $f = f^0$. f est une version modifiée de la fonction initiale f^0 . Les opérateurs ∇_w^+ et ∇_w^- correspondent aux gradients externe et interne pondérés (équations (2.63) et (2.64) à la page 23). Enfin, $\|\cdot\|_q$ désigne la norme \mathcal{L}_q pour des valeurs de $q = 1, 2, \dots, \infty$.

Les équations (4.15) et (4.16) représentent des équations de diffusion permettant de retrouver, sous certaines conditions, les équations de dilatation et d'érosion basées sur les équations aux dérivées partielles. L'utilisation des gradients externe et interne pondérés permettent d'introduire de l'adaptabilité dans nos processus de traitement. Les opérations que nous proposons peuvent être considéré comme des opérations morphologiques adaptatives. De ce fait, les opérations de dilatation et d'érosion que nous proposons dans le cadre de notre formalisme perdent certaines propriétés morphologiques et deviennent donc par conséquence des pseudo dilatation et érosion. Par abus de langage et pour des raisons de clarté, nous continuerons à appeler opérations morphologiques (dilatation, érosion, etc) les opérations que nous proposons.

Afin de clarifier et de faciliter la lecture de la suite de ce chapitre, des définitions et des propriétés du chapitre 2 sont rappelées.

4.3.1 Différences et gradients morphologiques

Afin d'établir les équations d'évolution de la dilatation et de l'érosion (4.15) et (4.16) définies sur des graphes, nous utilisons les définitions des différences et gradients directionnels pondérés présentées dans le chapitre 2. Nous rappelons ici leurs définitions.

Différences directionnelles pondérées morphologiques

Les différences pondérées externe et interne ont été définies, respectivement, comme

$$(\mathbf{d}_w^+ f)(uv) = \frac{\max(f(v), f(u)) - f(u)}{h_{uv}} = \frac{\max(0, f(v) - f(u))}{h_{uv}} \quad (4.17)$$

$$(\mathbf{d}_w^- f)(uv) = \frac{f(u) - \min(f(v), f(u))}{h_{uv}} = -\frac{\min(0, f(v) - f(u))}{h_{uv}} \quad (4.18)$$

pour un graphe pondéré $G = (V, E, w)$ et une fonction $f \in \mathcal{H}(V)$.

Avec le choix du pas d'échantillonnage $h_{uv} = 1/\sqrt{w_{uv}}$ que nous avons fait, nous obtenons

$$(\mathbf{d}_w^+ f)(uv) = \sqrt{w_{uv}} \max(0, f(v) - f(u)) \quad (4.19)$$

$$(\mathbf{d}_w^- f)(uv) = -\sqrt{w_{uv}} \min(0, f(v) - f(u)) \quad (4.20)$$

Remarque. Ces définitions des différences directionnelles correspondent à des différences directionnelles morphologiques. En effet, dans le cas d'un graphe non pondéré (fonction de poids g_1), nous retrouvons les définitions des différences morphologiques. Dans notre cas, nous les étendons à des graphes pondérés permettant plus d'adaptation dans le calcul des différences.

Gradients directionnels pondérés morphologiques

Les gradients directionnels externe et interne pondérés ont été respectivement définis en un sommet $u \in V$, comme

$$(\nabla_w^+ f)(u) = ((d_w^+ f)(uv))_{uv \in E}^T \quad (4.21)$$

$$(\nabla_w^- f)(u) = ((d_w^- f)(uv))_{uv \in E}^T . \quad (4.22)$$

pour un graphe pondéré $G = (V, E, w)$ et une fonction $f \in \mathcal{H}(V)$. Ces définitions des gradients directionnels correspondent à des gradients directionnels morphologiques. Ces gradients peuvent être utilisés sur des graphes arbitraires. Nous allons montrer dans la suite que, avec une fonction de poids et une structure de graphe particulière, nos formulations des gradients directionnels permettent de retrouver des formulations connues de la littérature.

Propriété 4.3.1. *Avec la norme infini du gradient externe morphologique (4.21) et dans le cas particulier d'un graphe non pondéré $G = (V, E, g_1)$, nous retrouvons l'expression du gradient externe morphologique pour un sommet $u \in V$ donné, c'est-à-dire $\delta(f)(u) - f(u)$.*

Cette propriété s'obtient de la manière suivante. Pour un sommet $u \in V$,

$$\begin{aligned} \|(\nabla_w^+ f)(u)\|_\infty &= \max_{v \sim u} \left(\sqrt{w_{uv}} \max(0, f(v) - f(u)) \right) \\ &= \max_{v \sim u} \left(\max(0, f(v) - f(u)) \right) \quad \text{avec } w = g_1 \\ &= \max_{v \sim u} \left(\max(f(u), f(v)) - f(u) \right) \\ &= \max_{v \sim u} (f(u), f(v)) - f(u) \\ &= \delta(f)(u) - f(u) . \end{aligned} \quad (4.23)$$

Propriété 4.3.2. *Avec la norme infini du gradient interne morphologique (4.22) et dans le cas particulier d'un graphe non pondéré $G = (V, E, g_1)$, nous retrouvons l'expression du gradient interne morphologique pour un sommet $u \in V$ donné, c'est-à-dire $f(u) - \varepsilon(f)(u)$.*

Cette propriété s'obtient de la manière similaire à la propriété précédente. Pour un sommet $u \in V$,

$$\begin{aligned} \|(\nabla_w^- f)(u)\|_\infty &= \max_{v \sim u} \left(\sqrt{w_{uv}} |\min(0, f(v) - f(u))| \right) \\ &= \max_{v \sim u} \left(|\min(0, f(v) - f(u))| \right) \quad \text{avec } w = g_1 \\ &= \max_{v \sim u} \left(\max(0, f(u) - f(v)) \right) \\ &= \max_{v \sim u} (0, f(u) - f(v)) \\ &= -\min_{v \sim u} (0, f(v) - f(u)) \\ &= f(u) - \min_{v \sim u} (f(u), f(v)) \\ &= f(u) - \varepsilon(f)(u) . \end{aligned} \quad (4.24)$$

Avec ces deux dernières propriétés, nous pouvons facilement remarquer que nos gradients directionnels pondérés permettent de retrouver les opérateurs gradient et

Laplacien morphologiques [Rivest *et al.*, 1993]. Ainsi, le gradient morphologique en un sommet $u \in V$, $\delta(f)(u) - \varepsilon(f)(u)$ s'exprime par

$$\|(\nabla_w^+ f)(u)\|_\infty + \|(\nabla_w^- f)(u)\|_\infty = \delta(f)(u) - \varepsilon(f)(u) . \quad (4.25)$$

De la même manière, le Laplacien morphologique en un sommet $u \in V$, $\delta(f)(u) + \varepsilon(f)(u) - 2f(u)$ s'exprime par

$$\|(\nabla_w^+ f)(u)\|_\infty - \|(\nabla_w^- f)(u)\|_\infty = \delta(f)(u) + \varepsilon(f)(u) - 2f(u) . \quad (4.26)$$

Nous pouvons remarquer que nos gradients externe et interne peuvent être également considérés, comme étant les versions discrètes des dérivées morphologiques sup et inf (*sup-derivative* et *inf-derivative*) d'une fonction continue f [Brockett et Maragos, 1992, Brockett et Maragos, 1994]. La dérivée sup $\mathcal{M}^+(f)$ d'une fonction continue $f : \mathbb{R}^m \rightarrow \mathbb{R}$ en un point x est définie comme

$$\mathcal{M}^+(f)(x) = \lim_{r \rightarrow 0} \frac{\sup_{\|y\| < r} (f(x+y)) - f(x)}{r} . \quad (4.27)$$

En appliquant \mathcal{M} à $-f$, et avec la dualité entre la dilatation et l'érosion, nous obtenons la dérivée morphologique inf en un point x

$$\mathcal{M}^-(f)(x) = \lim_{r \rightarrow 0} \frac{f(x) - \inf_{\|y\| < r} (f(x+y))}{r} . \quad (4.28)$$

4.3.2 Processus de dilatation et d'érosion

Dans cette section, nous allons présenter comment nous établissons les équations d'évolution de nos opérations inspirés par la dilatation et l'érosion définies par des équations aux différences partielles.

Soit f une fonction de $\mathcal{H}(V)$. Nous décomposons la fonction en ses ensembles de niveaux. Pour un niveau donné l , la fonction f s'écrit $f^l = \chi_{(f-l)}$ avec $\chi_{(S)} : V \rightarrow \{0, 1\}$, une fonction indicatrice telle que $\chi_{(S)}(u) = 1$ si $u \in S$ et $\chi_{(S)}(u) = 0$ sinon. Nous rappelons les définitions des bords externe $\partial^+ \mathcal{A}$ et interne $\partial^- \mathcal{A}$ (équations (2.73) et (2.74) page 24) d'un ensemble $\mathcal{A} \subseteq V$.

$$\partial^+ \mathcal{A} = \{u \notin \mathcal{A} : \exists v \in \mathcal{A} \text{ avec une arête } uv \in E\} \quad (4.29)$$

$$\partial^- \mathcal{A} = \{u \in \mathcal{A} : \exists v \notin \mathcal{A} \text{ avec une arête } uv \in E\} . \quad (4.30)$$

De manière intuitive, nous pouvons constater qu'une opération de dilatation sur l'ensemble \mathcal{A} peut être interprétée comme un processus d'expansion qui va venir ajouter des sommets du bord externe $\partial^+ \mathcal{A}$ vers \mathcal{A} . Par dualité, une opération d'érosion peut être vue comme étant une opération de contraction qui va retirer des sommets du bord interne de l'ensemble \mathcal{A} .

Comme pour les modèles définis dans le domaine continu, une simple définition variationnelle d'une dilatation appliquée à un niveau f^l d'une fonction f peut être interprétée comme étant la maximisation d'un gain de surface proportionnellement à la norme de son gradient $+\|\nabla f^l\|_q$. De la même manière, une érosion du niveau f^l peut être vue comme étant la minimisation d'un gain de surface proportionnellement à la norme de son gradient $-\|\nabla f^l\|_q$.

Rappelons la propriété (2.7.5) page 27 du chapitre 2. En fonction de la position d'un sommet $u \in V$ relativement aux bords externe et interne $\partial^+ \mathcal{A}^l$ et $\partial^- \mathcal{A}^l$, la norme \mathcal{L}_q du gradient de la fonction f^l (pour $0 < q < +\infty$ et $q = \infty$) se simplifie de la manière suivante

$$\|(\nabla_w f^l)(u)\|_q = \begin{cases} \|(\nabla_w^+ f^l)(u)\|_q, & \text{si } u \in \partial^+ \mathcal{A}^l \\ \|(\nabla_w^- f^l)(u)\|_q, & \text{si } u \in \partial^- \mathcal{A}^l \end{cases} . \quad (4.31)$$

En utilisant cette propriété, nous pouvons constater qu'une dilatation de la fonction f^l sur l'ensemble \mathcal{A}^l correspond uniquement à considérer les sommets du bord externe $\partial^+ \mathcal{A}$. De là, l'équation d'évolution de l'opération de dilatation peut s'écrire $\partial \delta(f^l) / \partial t = +\|\nabla_w^+ f^l\|_q$ où la norme du gradient $\|\nabla_w f\|_q$ se réduit à celle du gradient externe $\|\nabla_w^+ f\|_q$.

De la même manière, l'équation d'évolution de l'opération d'érosion de la fonction f^l sur l'ensemble \mathcal{A}^l peut s'écrire $\partial \varepsilon(f^l) / \partial t = -\|\nabla_w^- f^l\|_q$ où la norme du gradient $\|\nabla_w f\|_q$ se réduit à celle du gradient interne $\|\nabla_w^- f\|_q$.

Finalement, en généralisant ces équations à tous les niveaux de la fonction f , nous obtenons les équations d'évolution des opérations de dilatation et d'érosion d'une fonction f sur un graphe $G = (V, E, w)$ telles que pour un sommet $u \in V$, nous obtenons une famille de processus de diffusion paramétrée par q et la fonction de poids w

$$\frac{\partial \delta(f)(u)}{\partial t} = \frac{\partial f(u)}{\partial t} = +\|(\nabla_w^+ f)(u)\|_q \quad (4.32)$$

$$\frac{\partial \varepsilon(f)(u)}{\partial t} = \frac{\partial f(u)}{\partial t} = -\|(\nabla_w^- f)(u)\|_q . \quad (4.33)$$

Ces deux équations peuvent s'exprimer sous une équation générale d'évolution. Pour un sommet u

$$\frac{\partial f(u)}{\partial t} = \max(0, \mu) \|(\nabla_w^+ f)(u)\|_q + \min(0, \mu) \|(\nabla_w^- f)(u)\|_q \quad (4.34)$$

où μ est une valeur constante. Quand $\mu > 0$, nous utilisons le gradient externe et le processus d'évolution correspond à une pseudo-dilatation. Quand $\mu < 0$, nous utilisons le gradient interne et le processus d'évolution correspond à une pseudo-érosion.

4.3.3 Schémas numériques pour la dilatation et l'érosion

Les expressions de nos gradients externe et interne permettent d'obtenir des schémas numériques résolvant les équations d'évolution de la dilatation et de l'érosion définies sur des graphes.

Contrairement aux méthodes basées sur les équations aux dérivées partielles, aucune étape de discrétisation spatiale n'est nécessaire aux expressions discrètes de nos opérateurs différentiels.

En utilisant une discrétisation temporelle et avec la notation conventionnelle $f^n(u) \approx f(u, n\Delta t)$, les solutions des équations d'évolution (4.32) et (4.33) sont res-

pectivement, pour un sommet $u \in V$

$$\frac{f^{n+1}(u) - f^n(u)}{\Delta t} = +\|(\nabla_w^+ f)(u)\|_q \quad (4.35)$$

$$\frac{f^{n+1}(u) - f^n(u)}{\Delta t} = -\|(\nabla_w^- f)(u)\|_q . \quad (4.36)$$

La condition initiale est $f = f^0$ et $f^0 \in \mathcal{H}(V)$ correspond à la fonction initiale définie sur les sommets d'un graphe pondéré $G = (V, E, w)$.

Avec les expressions des normes des gradients, nous obtenons pour le processus de dilatation les algorithmes suivants pour les cas particuliers de $q = 1, 2$, et ∞

$$\frac{f^{n+1}(u) - f^n(u)}{\Delta t} = \|(\nabla_w^+ f^n)(u)\|_1 = \sum_{v \sim u} \sqrt{w_{uv}} \max(0, f^n(v) - f^n(u)) \quad (4.37)$$

$$\frac{f^{n+1}(u) - f^n(u)}{\Delta t} = \|(\nabla_w^+ f^n)(u)\|_2 = \sqrt{\sum_{v \sim u} w_{uv} \max(0, f^n(v) - f^n(u))^2} \quad (4.38)$$

$$\frac{f^{n+1}(u) - f^n(u)}{\Delta t} = \|(\nabla_w^+ f^n)(u)\|_\infty = \max_{v \sim u} (\sqrt{w_{uv}} |\max(0, f^n(v) - f^n(u))|) \quad (4.39)$$

À chaque étape des algorithmes, nous pouvons voir que la nouvelle valeur en un sommet $u \in V$ ne dépend uniquement que de son ancienne valeur au temps n et des valeurs des fonctions des ses voisins contenus dans son voisinage $\mathcal{N}(u)$.

Les mêmes algorithmes peuvent être écrits pour le cas de l'opération d'érosion.

Il est important de noter que les schémas numériques que nous proposons sont définis sur des graphes arbitraires permettant ainsi de traiter n'importe quel type de données définies sur des domaines réguliers ou irréguliers pouvant être représentés par un graphe.

De plus, la structure de graphe unifie naturellement les schémas locaux et non locaux permettant de définir de nouveaux opérateurs morphologiques basés sur ce type de configurations.

4.3.4 Relations avec la morphologie mathématique continue

Nous montrons ici qu'avec une fonction de poids particulière et une structure de graphe adaptée, nos schémas numériques sont reliés à des approches morphologiques connues de la littérature telles que le schéma numérique de Osher et Sethian [Osher et Sethian, 1988].

Schéma numérique de Osher et Sethian

Soit $G = (V, E, w)$ un graphe pondéré. Dans le cas où $q = 2$, le schéma numérique de l'opération de dilatation (4.38) est

$$\frac{f^{n+1}(u) - f^n(u)}{\Delta t} = \|(\nabla_w^+ f^n)(u)\|_2 = \sqrt{\sum_{v \sim u} w_{uv} \max(0, f^n(v) - f^n(u))^2} . \quad (4.40)$$

Si le graphe G représente une grille dans \mathbb{R}^m de dimensions m alors ce schéma numérique correspond aux modèles de discrétisation de Osher et Sethian pour une grille de dimension m [Osher et Sethian, 1988, Sethian, 1999b].

Dans ce cas, soit $u \in V$ un sommet du graphe G définissant un vecteur de coordonnées spatiales de dimensions m tel que $u = (i_1 h_1, \dots, i_m h_m)^T$ où les h_j représentent l'espacement de la grille et les $i_j \in \mathbb{N}$ avec $j = 1, \dots, m$.

Le voisinage $\mathcal{N}(u)$ de u peut alors être défini comme

$$\mathcal{N}(u) = \{v | v = u \pm h_j e_j \text{ avec } j = 1, \dots, m\} \quad (4.41)$$

où $e_j = (q_k)_{k=1, \dots, m}^T$ est un vecteur de \mathbb{R}^m tel que $q_k = 1$ si $j = k$ et $q_k = 0$ sinon. En remplaçant dans le schéma numérique les sommets par leurs nouvelles coordonnées, et en posant que $w_{uv_j} = 1/h_j$ pour tout $v_j \in \mathcal{N}(u)$, nous pouvons écrire

$$\frac{f^{n+1}(u) - f^n(u)}{\Delta t} = \sqrt{\sum_{j=1}^m \max\left(0, \frac{f^n(u \pm h_j e_j) - f^n(u)}{h_j}\right)^2}. \quad (4.42)$$

En utilisant les notations suivantes pour un sommet $u \in V$

$$\mathcal{D}_j^+ f(u) = \frac{f(u + h_j e_j) - f(u)}{h_j} \quad \text{et} \quad \mathcal{D}_j^- f(u) = \frac{f(u) - f(u - h_j e_j)}{h_j} \quad (4.43)$$

et la propriété que $\min(0, a - b)^2 = \max(0, b - a)^2$, nous avons le schéma de discrétisation de premier ordre avant (*upwind first order*) de Osher et Sethian sur une grille de dimension m

$$\frac{f^{n+1}(u) - f^n(u)}{\Delta t} = \sqrt{\sum_{j=1}^m \min(0, \mathcal{D}_j^- f^n(u))^2 + \max(0, \mathcal{D}_j^+ f^n(u))^2} \quad (4.44)$$

Remarque. La même propriété peut être obtenue avec l'équation décrivant l'opération d'érosion.

Cas particulier d'une grille régulière 2D

Soit $G_0 = (V, E, g_1)$ un graphe grille en 4-voisinage non pondéré (fonction de poids g_1 , implique que le pas $h_j = 1$) associé à une image 2D en niveaux de gris définie comme une fonction $f : V \subset \mathbb{Z}^2 \rightarrow \mathbb{R}$, l'équation (4.44) devient

$$\frac{f^{n+1}(u) - f^n(u)}{\Delta t} = \sqrt{\sum_{j=1}^2 \min(0, f(u) - f(u - e_j))^2 + \max(0, f(u + e_j) - f(u))^2}. \quad (4.45)$$

Si nous remplaçons les sommets $u \in V$ et leurs voisinages $\mathcal{N}(u)$ par leurs coordonnées spatiales (x, y) nous obtenons

$$\frac{f^{n+1}((x, y)) - f^n((x, y))}{\Delta t} = \left[\begin{aligned} &\min\left(0, f((x, y)) - f((x-1, y))\right)^2 + \\ &\max\left(0, f((x+1, y)) - f((x, y))\right)^2 + \\ &\min\left(0, f((x, y)) - f((x, y-1))\right)^2 + \\ &\max\left(0, f((x, y+1)) - f((x, y))\right)^2 \end{aligned} \right]^{1/2}. \quad (4.46)$$

Ce schéma numérique correspond aux schémas utilisés par la morphologie mathématique basée sur les équations aux dérivées partielles pour traiter les images en niveaux de gris [Guichard *et al.*, 2007].

4.3.5 Relations avec la morphologie mathématique définie sur des graphes

Notre formulation basée sur les équations aux différences partielles est liée à la formulation de la morphologie mathématique définie sur des graphes [Heijmans *et al.*, 1992, Heijmans et Vincent, 1992].

En effet, si nous prenons le cas de la dilatation avec le paramètre $q = \infty$ (cas de la norme \mathcal{L}_∞ du gradient) alors nous avons

$$\frac{f^{n+1}(u) - f^n(u)}{\Delta t} = \|(\nabla_w^+ f^n)(u)\|_\infty = \max_{v \sim u} (\sqrt{w_{uv}} |\max(0, f^n(v) - f^n(u))|). \quad (4.47)$$

Si nous considérons une discrétisation temporelle constante ($\Delta t = 1$), alors le schéma numérique devient, pour un graphe non pondéré $G = (V, E, g_1)$,

$$\begin{aligned} f^{n+1}(u) &= f^n(u) + \max_{v \sim u} (\max(0, f^n(v) - f^n(u))) \\ f^{n+1}(u) &= f^n(u) + \max_{v \sim u} (f^n(v) - f^n(u)) \end{aligned} \quad (4.48)$$

Ainsi, la valeur de la fonction f en un sommet u au temps $n + 1$ vaut

$$f^{n+1}(u) = \begin{cases} \max_{v \sim u} (f^n(v)) & \text{si } f^n(v) - f^n(u) > 0 \\ f^n(u) & \text{sinon} \end{cases}. \quad (4.49)$$

Ce qui permet de déduire, dans tous les cas, l'expression d'une dilatation définie sur un graphe

$$\delta^n(f)(u) = f^{n+1}(u) = \max_{v \sim u} (f^n(v), f^n(u)). \quad (4.50)$$

Remarque. Dans ce cas, l'élément structurant est exprimé par la topologie du graphe et le voisinage des sommets. Par exemple, si nous considérons un graphe grille en 8-voisinage associé à une image alors, l'élément structurant correspond à un carré de taille 3×3 .

De manière similaire, nous pouvons montrer que, dans les mêmes conditions, nous obtenons la formulation d'une érosion définie sur un graphe

$$\varepsilon^n(f)(u) = f^{n+1}(u) = \min_{v \sim u} (f^n(v), f^n(u)) . \quad (4.51)$$

Quand $n = 1$, nous obtenons une dilatation et une érosion élémentaires ce qui correspond aux opérations $\delta(f)$ et $\varepsilon(f)$. Comme en morphologie mathématique classique et avec un graphe non pondéré, nous avons les propriétés suivantes

$$\delta^n(f) = \underbrace{\delta(f) \circ \delta(f) \circ \dots \circ \delta(f)}_{n \text{ fois}} \quad (4.52)$$

$$\varepsilon^n(f) = \underbrace{\varepsilon(f) \circ \varepsilon(f) \circ \dots \circ \varepsilon(f)}_{n \text{ fois}} . \quad (4.53)$$

4.3.6 Filtrage par nivellements

De manière similaire à la méthodologie que nous avons utilisée pour adapter les opérations de dilatation et d'érosion définies par des équations aux dérivées partielles, nous présentons ici une transcription des filtres par nivellements (*levelings*) définis dans le domaine continu.

Nivellements basés sur les équations aux dérivées partielles

Dans le contexte du traitement des images, les filtres par reconstruction (reconstruction par ouverture ou fermeture) sont capables de reconstruire les objets présents dans les images, tout en respectant exactement les contours de ces objets (voir par exemple [Salembier et Serra, 1995] et les références associées). Les objets sont reconstruits à partir d'une image appelée fonction marqueur qui est produite à partir d'une image initiale (fonction de référence). Durant le processus de reconstruction, ces filtres simplifient les objets à reconstruire en éliminant les parties où le marqueur ne peut accéder.

Afin de palier les problèmes d'asymétrie de ces filtres, Maragos et Meyer ont proposé une classe générale de filtres morphologiques que sont les filtres par nivellements [Maragos et Meyer, 1999, Meyer et Maragos, 2000, Maragos, 2003]. La formulation par équations aux dérivées partielles permettant d'obtenir les nivellements d'une image de référence définie comme une fonction $f^0 : \Omega \subset \mathbb{Z}^2 \rightarrow \mathbb{R}$ à partir d'une fonction marqueur $m : \Omega \subset \mathbb{Z}^2 \rightarrow \mathbb{R}$ et d'un disque infinitésimal est

$$\begin{cases} \frac{\partial f}{\partial t} = \text{sign}(f^0 - f) \|\nabla f\| \\ f = m, \quad \text{à } t = 0 \end{cases} . \quad (4.54)$$

où $\text{sign} : \mathbb{R} \rightarrow \{-1, 0, 1\}$ est la fonction signe définie comme

$$\text{sign}(x) \begin{cases} -1 & \text{si } x < 0 \\ 0 & \text{si } x = 0 \\ 1 & \text{si } x > 0 \end{cases} . \quad (4.55)$$

Au temps $t \rightarrow \infty$, nous obtenons une fonction f qui correspond au nivellement de la fonction f^0 en accord avec la fonction marqueur m .

Si f^0 et m ne possède pas d'ordre spécifique alors (4.54) décrit un processus dont le signe varie dans le temps permettant ainsi de contrôler la croissance ou la décroissance de la fonction m (dilatation-érosion conditionnelle).

Reconstruction par ouverture et fermeture Si la fonction marqueur $m \leq f^0$ sur l'ensemble de son domaine de définition, alors (4.54) décrit une succession de dilatations planaires par un disque infinitésimal et la fonction marqueur va itérativement croître. Dans cette situation, à $t \rightarrow \infty$ nous obtenons avec la fonction f une reconstruction par ouverture de la fonction de référence f^0 en accord avec le marqueur m .

Si maintenant la fonction marqueur $m \geq f^0$, alors (4.54) décrit une succession d'érosions planaires, la fonction marqueur va itérativement décroître. Finalement, à $t \rightarrow \infty$ nous obtenons avec la fonction f une reconstruction par fermeture de la fonction de référence f^0 en accord avec le marqueur m .

Adaptation des filtres par nivellements

Comme pour les cas de la dilatation et l'érosion, nous transcrivons les filtres par nivellements sur des graphes en exploitant le formalisme des équations aux différences partielles.

Considérons une fonction de référence $f^0 \in \mathcal{H}(V)$ et sa fonction marqueur $m \in \mathcal{H}(V)$. Le nivellement $f \in \mathcal{H}(V)$ de la fonction f^0 en accord avec le marqueur m s'exprime, en un sommet $u \in V$ d'un graphe $G = (V, E, w)$ par

$$\begin{cases} \frac{\partial f(u)}{\partial t} &= \max\left(0, \text{sign}(f^0(u) - f(u))\right) \|(\nabla_w^+ f)(u)\|_q \\ &+ \min\left(0, \text{sign}(f^0(u) - f(u))\right) \|(\nabla_w^- f)(u)\|_q \\ f(\cdot) &= m(\cdot), \quad \text{à } t = 0 \end{cases} \quad (4.56)$$

L'équation d'évolution (4.56) décrit une famille de nivellements définie sur des graphes paramétrée par la valeur de q et la fonction de poids w .

Lorsque la quantité $\text{sign}(f^0 - f) > 0$, le schéma (4.56) agit comme une dilatation et nous utilisons le schéma numérique (4.35). De même, lorsque $\text{sign}(f^0 - f) < 0$, le schéma (4.56) agit comme une opération d'érosion et nous utilisons la résolution (4.36).

De manière similaire aux opérations de dilatation et d'érosion, notre formulation des filtres par nivellements intègre naturellement les configurations locales et non locales pour le traitement des images. Notre approche permet également d'utiliser ce type de filtre pour traiter tout type de données arbitraires.

Finalement, dans le cas d'un graphe grille non pondéré associé à une image en niveaux de gris et avec $q = 2$, le schéma numérique permettant de résoudre l'équation (4.56) correspond exactement aux schémas utilisés par les méthodes basées sur les équations aux dérivées partielles pour résoudre l'équation d'évolution des filtres par nivellements [Maragos et Meyer, 1999, Meyer et Maragos, 2000].

4.4 Expérimentations

Dans cette section, nous appliquons notre nouveau formalisme basée sur les équations aux différences partielles pour le traitement d'images et de données non organisées de grande dimension. Nous présentons des expérimentations pour montrer le comportement de nos opérations inspirées de la morphologie mathématique. L'objectif des résultats présentés dans cette section n'est pas de résoudre un problème particulier ou une application particulière mais de montrer les potentialités de notre approche pour traiter des données discrètes arbitraires.

Nous nous intéressons tout d'abord aux cas des images en utilisant différents type de graphes : graphes de pixels et de régions. Ensuite, nous présenterons des expérimentations de traitement de données telles que des nuages de points ou des bases de données.

4.4.1 Cas des données vectorielles

Notre approche permet de traiter les données multi variées c'est-à-dire vectorielles $f : V \rightarrow \mathbb{R}^m$. Pour un sommet $u \in V$ d'un graphe $G = (V, E, w)$, sa fonction représentative $f(u)$ est alors le vecteur $f(u) = (f_i(u))_{i=1, \dots, m}^T$.

Le traitement de ce type de fonction est réalisé sur chacune des composantes f_i du vecteur f (composante par composante). Cela revient donc à avoir m traitements morphologiques en parallèle. Appliquer les traitements composante par composante a l'avantage de permettre la parallélisation des algorithmes. Pour des applications particulières, ce type de traitements peut avoir des inconvénients par rapport aux solutions purement vectorielles. En effet, contrairement à la morphologie classique, ce type de traitement introduit une modification des données initiales. Par exemple dans le cadre du traitement des images couleurs, les opérations morphologiques ainsi réalisées introduisent de « fausses couleurs ». Afin de s'affranchir de ce problème, les traitements morphologiques des fonctions multi variées doivent prendre en compte la corrélation entre les différentes composantes du vecteur. Dans notre cas, nous utilisons la fonction de poids w (calculée de manière vectorielle) comme terme de couplage permettant ainsi de prendre en compte les différentes composantes du vecteur à traiter.

Finalement, ce type de traitement permet, d'une manière simple, de traiter des données de grande dimension non organisées sans avoir à définir d'ordre sur les données vectorielles.

4.4.2 Traitements morphologiques des images

Dans cette section, nous illustrons notre formulation basée sur les graphes à travers des opérations morphologiques simples sur des images en niveaux de gris : dilatation, érosion, ouverture ou fermeture.

Remarque. *L'ouverture et la fermeture sont des opérations morphologiques réalisées à partir des opérations élémentaires de dilatation et d'érosion. La manière la plus simple de générer les opérations d'ouverture et de fermeture est de composer en série les opérations de dilatation et d'érosion. Ainsi, pour une fonction $f \in \mathcal{H}(V)$, l'opération d'ouverture est $\delta(\varepsilon(f))$ et la fermeture de f correspond à $\varepsilon(\delta(f))$.*



FIG. 4.1: Images tests initiales servant pour les figures 4.2 (figure (a)) et 4.3 (figure (b)).

Nous faisons également remarquer au lecteur que dans le cas des opérations de dilatation et d'érosion pondérées, les poids du graphe sont calculé à partir de l'image initiale et ne sont pas recalculé au cours des itérations. Dans le cas de la fermeture (resp. ouverture) les poids sont recalculés lorsque nous appliquons la dilatation (resp. érosion).

Une image en niveaux de gris est considéré comme une fonction $f^0 : V \subset \mathbb{Z}^2 \rightarrow \mathbb{R}$ qui définit une application des sommets d'un graphe $G = (V, E, w)$ vers les valeurs de niveaux de gris des pixels de l'image.

La figure 4.1 montre les deux images de tests en niveaux de gris servant d'images initiales pour les traitements morphologiques des figures 4.2 et 4.3.

Les figures 4.2 à la page 134 et 4.3 à la page 135 illustrent la flexibilité et l'adaptabilité de notre approche des opérations morphologiques. En effet, les résultats présentés dans ces figures ont été obtenus pour différentes valeurs du paramètre q , fonctions de poids w et topologies du graphe G .

La figure 4.2 montre des résultats pour une dilatation et une fermeture pour le cas particulier où $q = 2$. La première ligne de cette dernière figure montre le résultat obtenu avec une discrétisation temporelle $\Delta t = 1$, une valeur de $q = \infty$. Cela correspond à une dilatation et une fermeture algébrique sur des graphes. Dans ce cas, le graphe utilisé est un graphe G_2 avec la distance de « city block ». Le voisinage des sommets permet d'obtenir un élément structurant de type disque de rayon égale à 2.

La figure 4.3 montre des résultats pour une dilatation et une érosion et l'influence des valeurs particulières du paramètre $p = \infty, 1$ et 2. La première colonne de cette figure et la deuxième ligne de la figure 4.2 montrent les résultats obtenus avec un graphe grille en 4-voisinage G_0 non pondéré $w = g_1$. Ce cas correspond au cas obtenu par les méthodes basées sur les équations aux dérivées partielles. Nous pouvons constater dans la figure 4.3 l'influence du paramètre q permettant de générer les éléments structurants que sont le losange, le carré et le disque. Nous pouvons aussi voir que les méthodes basées sur les équations aux dérivées partielles ont tendance à introduire un effet de flou notamment au niveau des contours.

La deuxième colonne de la figure 4.3 et la troisième ligne de la figure 4.2

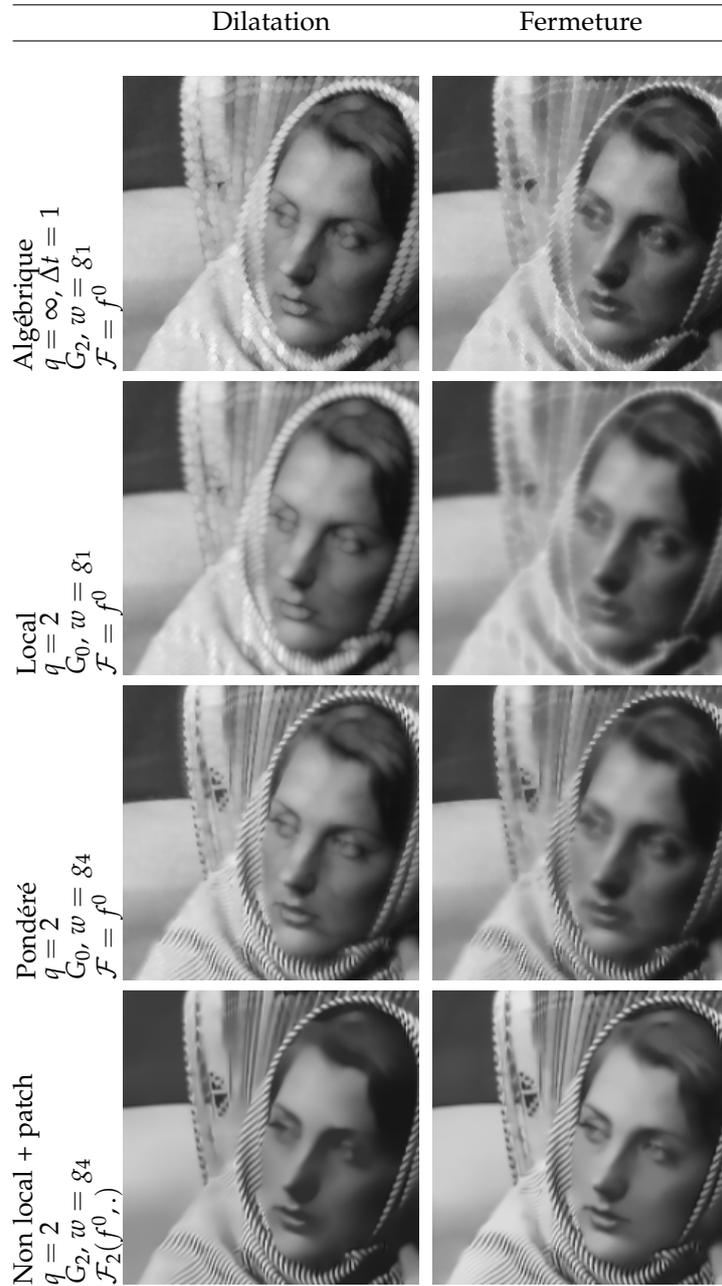


FIG. 4.2: Dilatation et fermeture avec différentes valeurs de q , de fonctions de poids w et de topologies du graphe G . À la première et seconde colonne, respectivement : dilatation et fermeture. Algébrique (première ligne). Local non pondéré (deuxième ligne). Local pondéré (troisième ligne). Non local avec patches (dernière ligne).

montrent des résultats avec un graphe grille en 4-voisinage G_0 pondéré $w = g_4$. Dans ces cas, l'utilisation d'une fonction de poids permet de mieux préserver les informations de contours en comparaison avec les résultats montrant des cas non pondérés.

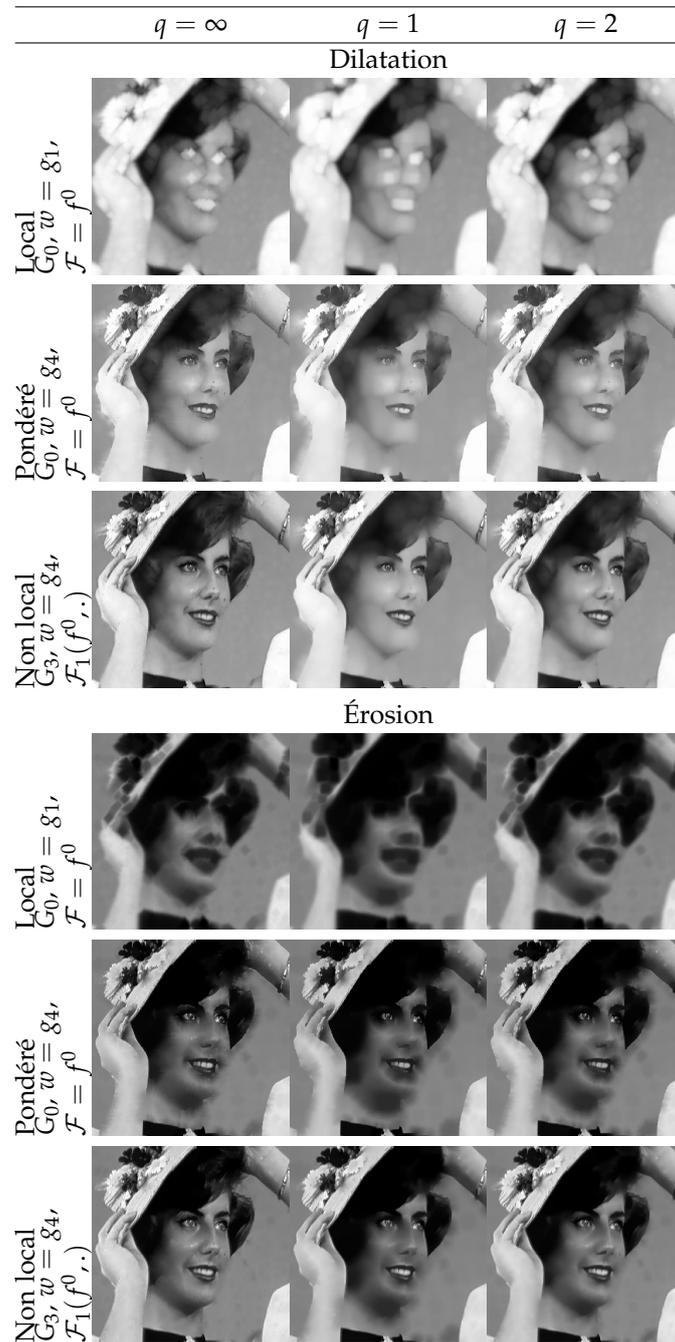


FIG. 4.3: Dilatation et érosion avec différentes valeurs de q , de fonctions de poids w et de topologies de graphe G . Pour chaque opération : local non pondéré (première ligne); local pondéré (deuxième ligne); non local avec des patches (dernière ligne). En colonne, de gauche à droite : $q = \infty$, $q = 1$ et $q = 2$.

La dernière colonne de la figure 4.3 et la dernière ligne de la figure 4.2 montrent des opérations morphologiques avec des schémas non locaux et des patches. Dans

le cas de la figure 4.3, le graphe utilisé est un graphe en 48-voisinage G_3 avec des patches de taille 3×3 . Dans le cas de la figure 4.2, le graphe utilisé est un graphe en 25-voisinage G_2 avec des patches de taille 5×5 . Ces deux graphes sont obtenus à l'aide de la distance de Chebychev et pondérés par la fonction g_4 où les distances sont calculées en utilisant les patches comme vecteur d'attributs.

Ces résultats montrent clairement le bénéfice des schémas non locaux dans les opérations morphologiques. Ces traitements permettent de mieux préserver les informations de contours et les structures fines et répétitives telles que les textures. On peut également remarquer l'effet filtrant de ce type de traitement, notamment dans les zones quasi homogènes.

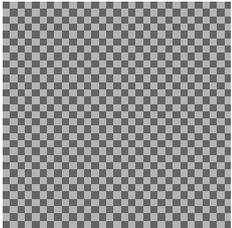
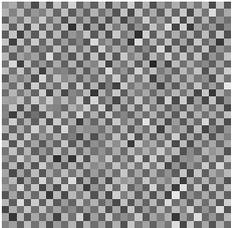
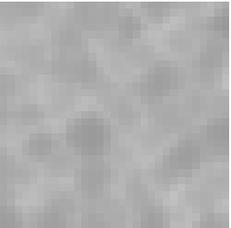
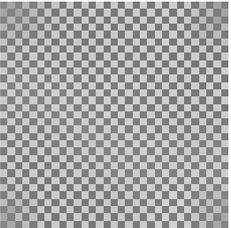
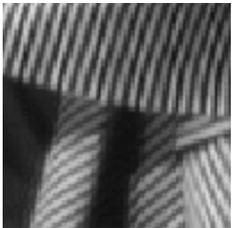
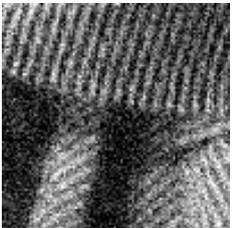
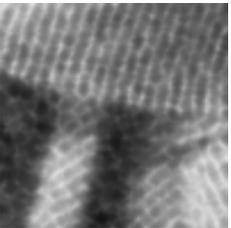
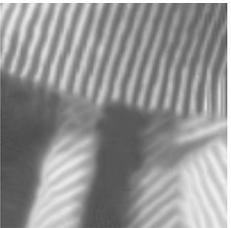
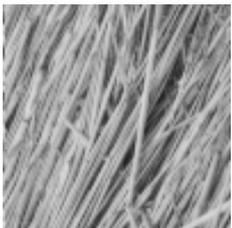
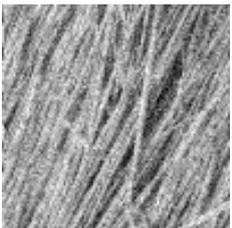
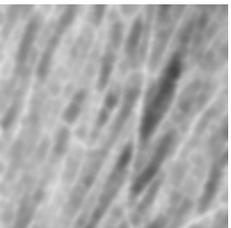
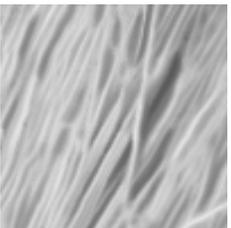
Images initiales	Images bruitées	Local $G_0, w = g_1,$ $\mathcal{F} = f^0$	Non local 10-NNG ₁₀ , $w = g_1,$ $\mathcal{F}_3(f^0, \cdot)$
			
			
			

FIG. 4.4: Comparaison entre fermetures locales et non locales avec des patches. Images initiales (première colonne). Images bruitées (Gaussien de variance égale à 20) (deuxième colonne). Local (troisième colonne). Non Local avec des patches (dernière colonne).

La figure 4.4 compare les schémas locaux et non locaux avec une opération de fermeture pour des images bruitées comportant des éléments fins et répétitifs.

Cette figure montre des résultats pour trois images de tests (une par ligne). La première colonne montre les images initiales. La deuxième colonne montre les images corrompues auxquelles nous avons ajouté un bruit Gaussien de variance $\sigma = 20$. La troisième colonne présente les résultats d'une fermeture locale, corres-

pondant aux méthodes basées sur les équations aux dérivées partielles. La dernière colonne présente les résultats d'une fermeture basée sur des schémas non locaux. Pour obtenir ces résultats, nous avons utilisé un graphe des 10- NNG_{10} non pondéré (fonction g_1). Chaque pixel est connecté à ses 10 plus proches voisins sélectionnés dans une fenêtre de recherche de taille 21×21 (centrée sur le pixel étudié). Afin de calculer la distance entre les voisins, nous utilisons des patches de taille 7×7 comme vecteur d'attributs ($\mathcal{F}_3(f^0, \cdot)$). Ces résultats montrent une fois de plus la supériorité des schémas non locaux dans les opérations morphologiques par rapport aux schémas locaux qui détruisent tous les éléments fins et répétitifs.

4.4.3 Nivellements des images

Notre approche de la morphologie mathématique permet l'intégration des schémas non locaux dans le traitement morphologique des images. Dans cette section, nous montrons les avantages de ce type de schémas dans les filtres morphologiques par nivellements.

La figure 4.5 présente des résultats de filtrage par nivellements obtenus à partir de l'équation (4.56) sur trois images de test. La première colonne de la figure 4.5 montre les images initiales servant d'images de référence dans le processus de nivellement. Elles sont définies comme des fonctions $f^0 : V \subset \mathbb{Z}^2 \rightarrow \mathbb{R}$. La deuxième colonne montre les images marqueurs à partir desquelles les nivellements sont obtenus. Les marqueurs sont obtenus simplement par filtrage Gaussien.

La troisième colonne montre les résultats des nivellements obtenus à partir des équations aux dérivées partielles locales classiques. Nous avons utilisé dans ce cas un graphe grille en 4-voisinage G_0 non pondéré ($w = g_1$) avec comme vecteur d'attributs $\mathcal{F} = f^0$.

La dernière colonne montre les résultats des nivellements obtenus avec nos équations aux différences partielles qui intègrent des schémas non locaux basés sur des patches. Pour obtenir ces résultats, nous utilisons des graphes basés sur le k - NNG où les distances entre les voisins sont calculées avec des vecteurs d'attributs de taille 7×7 ($\mathcal{F} = \mathcal{F}_3(f^0, \cdot)$). Pour les résultats de la première ligne, nous utilisons un 10- NNG_{10} non pondéré et pour les deux autres résultats nous utilisons respectivement un 2- NNG_2 et un 4- NNG_2 . Ces deux derniers graphes sont pondérés par la fonction de poids g_4 couplés avec un graphe grille en 4-voisinage G_0 formant ainsi les deux graphes de voisinage $G_0 \cup 2\text{-NNG}_2$ et $G_0 \cup 4\text{-NNG}_2$.

Les résultats montrent clairement la supériorité des nos schémas non locaux dans la reconstruction des éléments fins et répétitifs par rapport aux méthodes classiques. Dans les cas où les méthodes locales ne parviennent pas à reconstruire l'image de référence, notre approche non locale basée sur les patches des filtres par nivellements permet d'obtenir un résultat satisfaisant.

4.4.4 Traitements morphologiques à partir de graphes de régions

Comme dans le chapitre 3 pour le cas de la segmentation semi supervisée des images, nous proposons d'utiliser des graphes de régions à la place des graphes de pixels dans les processus morphologiques. La figure 4.6 montre l'utilisation d'une partition de l'image et de ses régions associées à un graphe pour effectuer des

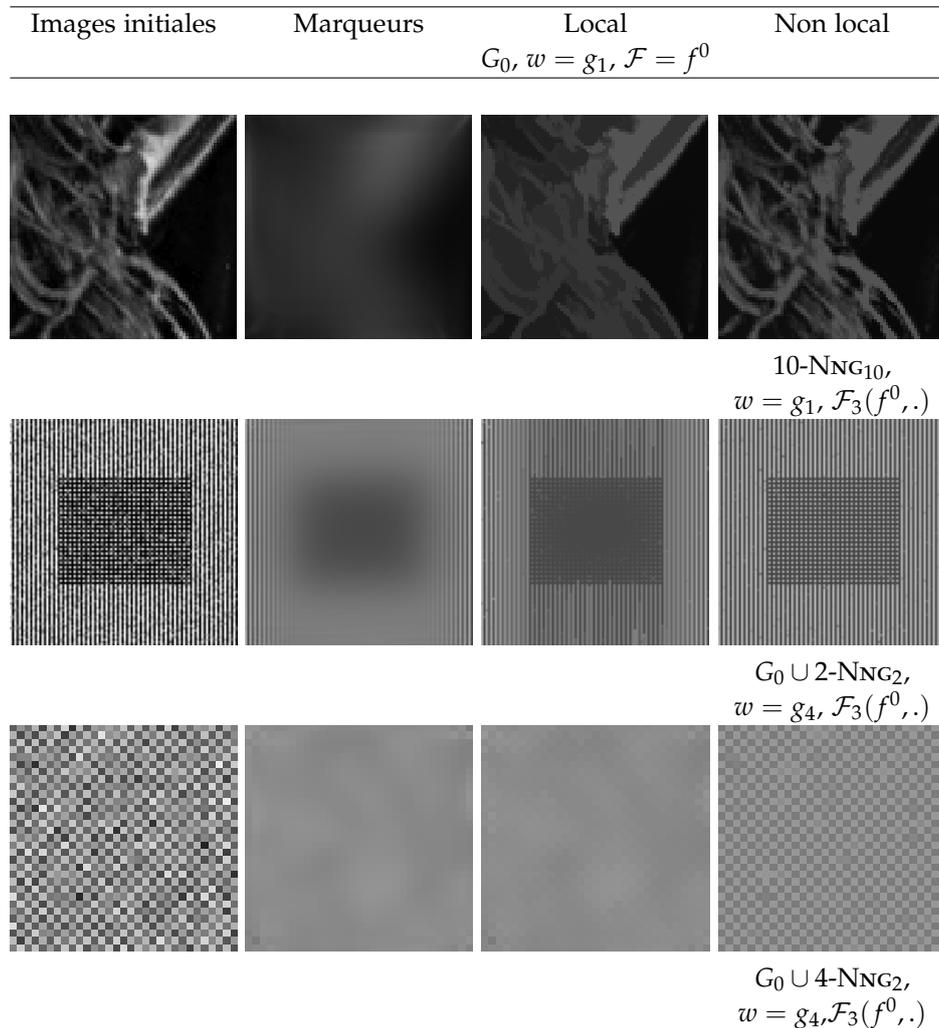


FIG. 4.5: Comparaisons entre nivellements locaux et non locaux. *Première colonne* : images initiales; *deuxième colonne* : images de marqueur; *troisième colonne* : local; *dernière colonne* : non local avec patches.

opérations morphologiques. Cette partition a été calculée à partir de la méthode des partitions d'énergies proposée par [Arbeláez et Cohen, 2004].

La figure 4.6b montre une partition calculée à partir de l'image initiale 4.6a définie comme une fonction $f^0 : V \subset \mathbb{Z}^2 \rightarrow \mathbb{R}$. À cette partition, nous associons un RAG. Chaque sommet du RAG représente une région de la partition et est décrite par la valeur moyenne des pixels qui la composent. La figure 4.6c montre l'image reconstruite à partir des régions correspondantes. Nous pouvons constater la réduction non négligeable en terme de nombre de sommets de la partition par rapport à l'image initiale. Cette réduction de sommets permet de réduire la complexité des traitements à effectuer.

La figure 4.6 montre des exemples d'opérations morphologiques (dilatation, érosion et fermeture) et compare les résultats entre un graphe pondéré G_0 de pixels et

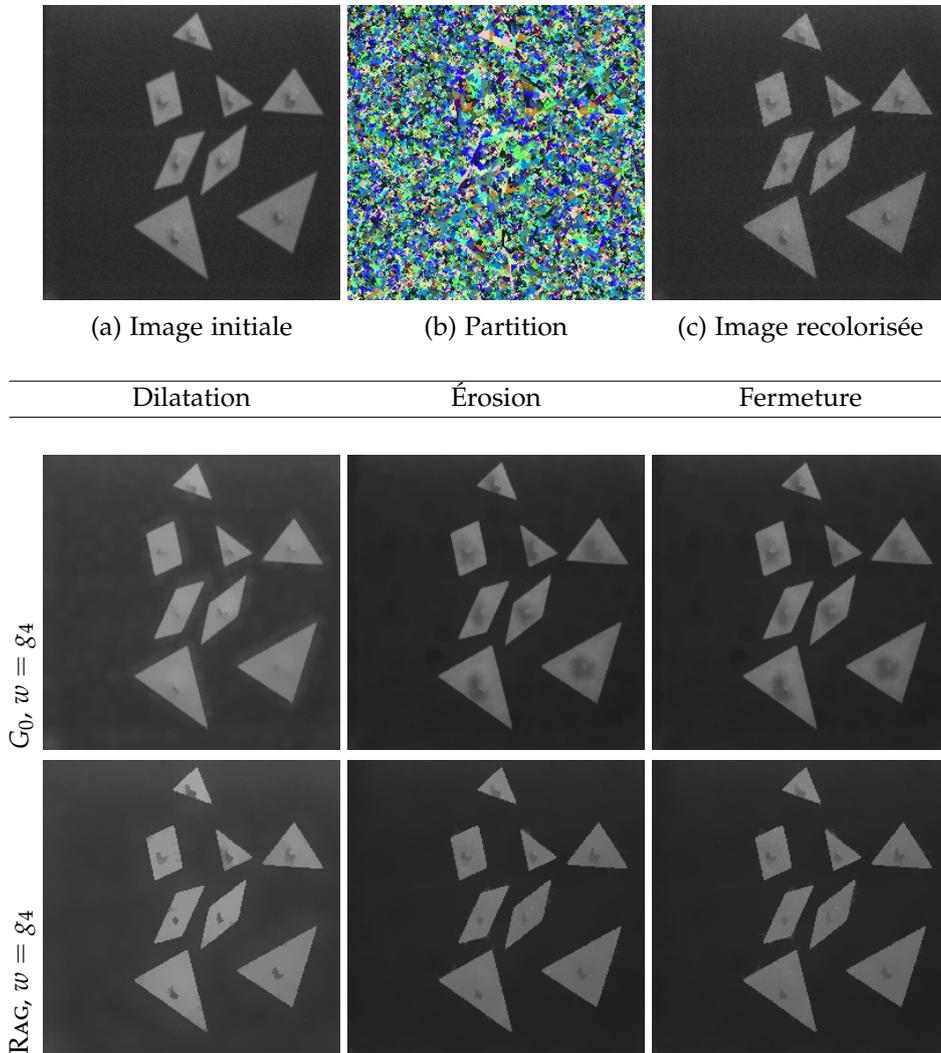


FIG. 4.6: Extension de la dilatation, de l'érosion et de la fermeture sur graphe de régions. (a) : image initiale (65 536 pixels). (b) : partition calculée à partir de (a) (11 853 régions soit 82% de réduction). (c) : image reconstruite à partir de la partition et de la valeur moyenne des régions. *Deuxième ligne* : résultats avec un graphe grille en 4-voisinage (G_0). *Dernière ligne* : résultats avec un RAG.

un RAG pondéré. Nous pouvons constater que nous obtenons dans les deux cas des résultats similaires tout en réduisant, dans le cas du RAG, le nombre de données à considérer et à traiter.

4.4.5 Traitements morphologiques de données arbitraires

Notre formulation permet de traiter n'importe quel type de données pouvant être représentées par un graphe.

Dans les expérimentations qui vont suivre, nous montrons la généricité et les

potentialités de notre méthodologie pour le traitement morphologique de données de grande dimension (bases de données, nuages de points, collections d'images).

Dilatation et érosion de base de données

La figure 4.7 montre un exemple de traitements morphologiques de base de données. Comme dans le chapitre 3 (filtrage par régularisation de base de données), nous utilisons les données « iris ».

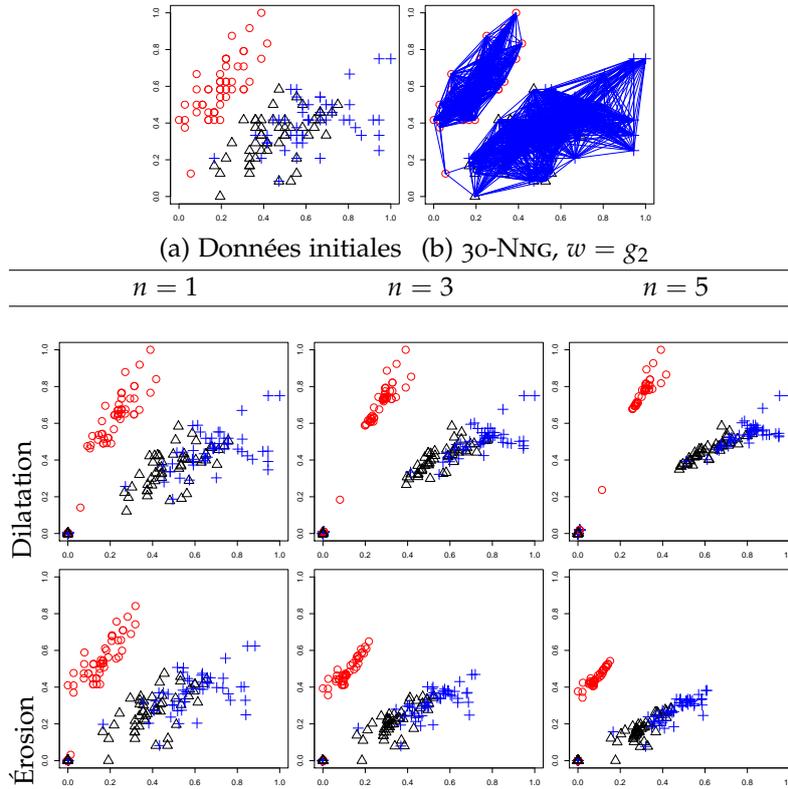


FIG. 4.7: Évolution des opérations de dilatation et d'érosion sur la base « iris ». La fonction f^0 associée aux données est définie (et traitée) comme une fonction $f^0 : V \rightarrow \mathbb{R}^4$. Les résultats présentent uniquement une projection sur deux attributs. (a) : données initiales ; (b) : 30-NNG. La première ligne : évolution d'une dilatation ; la deuxième ligne : évolution d'une érosion.

La figure 4.7 montre une projection sur deux des quatre attributs de la base initiale où chaque classe est représentée par une couleur et un symbole différent. Les données sont définies et sont traitées comme des fonctions $f^0 : V \rightarrow \mathbb{R}^4$. Afin d'appliquer des traitements morphologiques sur ces données, nous construisons un 30-NNG pondéré par la fonction de poids g_2 . La figure 4.7b présente le graphe associé aux données. La fonction de poids utilisée joue donc le rôle de terme de couplage permettant de prendre en compte l'aspect vectoriel des données. Les deux lignes suivantes de la figure 4.7 montrent l'évolution d'une dilatation et d'une érosion en fonction du nombre d'itérations n . Nous pouvons constater que la dilatation

et l'érosion ont un effet de regroupement sur les données. Les données sont projetées dans des parties différentes de l'espace des attributs et se déplacent vers un maximum ou un minimum de cet espace. Cela correspond bien aux comportements attendus pour une dilatation ou une érosion.

Filtrage par nivellements de nuages de points

La figure 4.8 montre des traitements morphologiques sur quatre nuages de points différents. Chaque nuage de points est considéré comme une fonction $f^0 : V \rightarrow \mathbb{R}^2$, (4.8a) où chaque sommet du graphe est associé aux coordonnées spatiales de chaque point. Le graphe associé à chaque ensemble de données est un 8-NNG pondéré par la fonction de poids g_4 (4.8b).

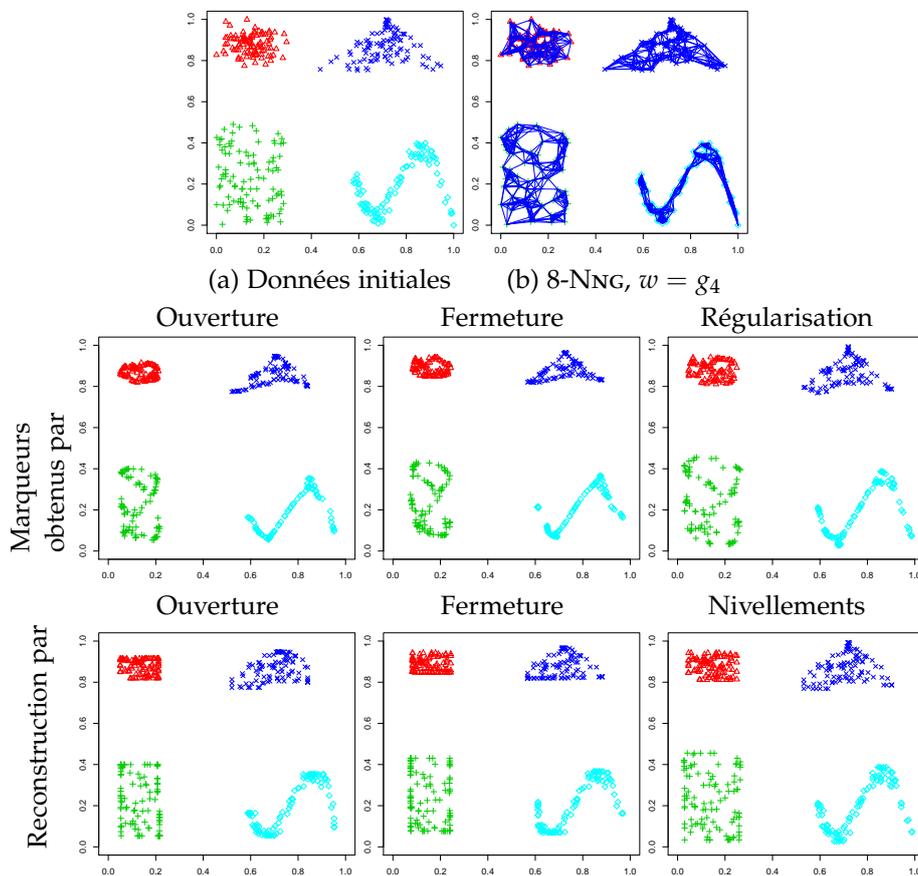


FIG. 4.8: Filtrage par reconstruction de nuages de points. Les quatre nuages de points sont définies comme quatre fonctions $f^0 : V \rightarrow \mathbb{R}^2$ associées aux données. (a) : données initiales. (b) : les 8-NNG. Les deux dernières lignes, de gauche à droite : reconstruction par ouverture, fermeture et nivellements. Les marqueurs sont obtenus (de gauche à droite) par ouverture, fermeture et régularisation discrète isotrope.

La deuxième partie de la figure 4.8 montre des opérations de filtrage par reconstruction. La première ligne de cette partie, présente les données marqueurs à partir

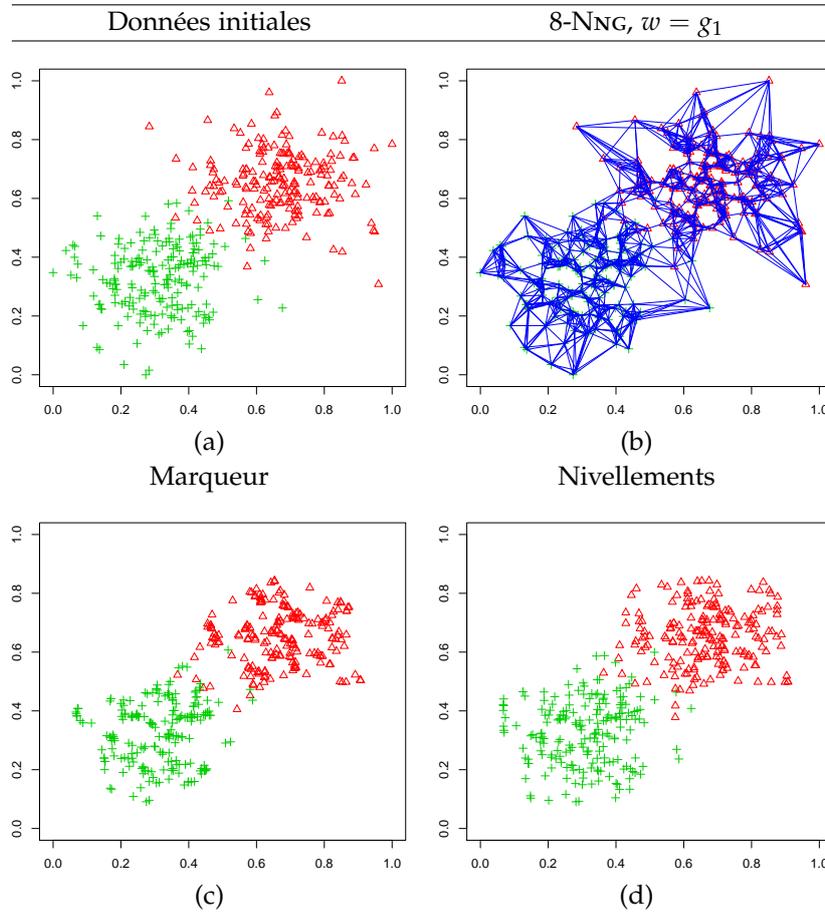


FIG. 4.9: Filtrage par nivellements de deux Gaussiennes. La fonction f^0 associée aux données est définie et traitée comme une fonction $f^0 : V \rightarrow \mathbb{R}^2$. (a) : les données initiales. (b) : 8-NNG associée aux données. (c) : marqueur obtenu par régularisation discrète isotrope. (d) : résultat du filtrage par nivellements.

desquelles les reconstructions sont réalisées. Ces marqueurs ont été obtenus à partir de nos formulations définies sur des graphes de la morphologie mathématique et de la régularisation discrète. Ils correspondent à une ouverture, une fermeture et le dernier marqueur a été obtenu par un filtrage basé sur la régularisation isotrope présentée dans le chapitre 3. Les reconstructions obtenues sont présentées à la dernière ligne de la figure 4.8 qui montre respectivement, de gauche à droite, une reconstruction par ouverture, par fermeture et un nivellement.

La figure 4.9 présente également le résultat d'un nivellement sur un nuage de points représentant deux Gaussiennes (figure 4.9a). Le graphe associé (voir la figure 4.9b) est un 8-NNG pondéré. Le marqueur (figure 4.9c) est obtenu par un filtrage basé sur la régularisation discrète définie sur des graphes (chapitre 3). Le résultat du nivellement, en accord avec le marqueur, est présenté à la figure 4.9d.

Nous pouvons constater que cela soit pour la figure 4.8 ou pour la figure 4.9, l'effet filtrant et simplificateur des filtres par reconstruction sur les données. Les structures principales de chaque jeu de données ont bien été reconstruites et les

données aux extrémités ont été filtrées. Ce type de traitement présente un intérêt pour la suppression et la détection des *outliers* dans des bases de données.

Traitements morphologiques de collections d'images

La figure 4.10 présente l'application des traitements morphologiques pour des données de grande dimension non organisées : des bases de données d'images qui peuvent être considérées comme des variétés d'images.

Afin de traiter ces données, des k -NNG pondérés par la fonction de poids g_2 sont construits à partir des données initiales. La première ligne de la figure 4.10 montre une représentation de ces graphes. Les trois dernières lignes de cette figure présentent de gauche à droite, les données initiales et les opérations de dilatation, d'érosion et d'ouverture sur ces variétés d'images.

Les images de chiffres proviennent de la base de chiffres manuscrits de l'USPS (déjà utilisée dans le chapitre précédent).

La figure 4.10 montre deux exemples de test de 100 échantillons chacun, où les échantillons ont été tirés de manière aléatoire. Le premier exemple provient de la base du chiffre « 0 » et le deuxième exemple provient des bases des chiffres « 3 » et « 1 ». Quant aux images de visages (dernière ligne de la figure 4.10), elles proviennent d'une base contenant 133 images en niveaux de gris d'une personne avec différentes postures de tête [Sherrah et Gong, 2001]. Chaque image de cette base est de taille 29×29 . La dernière ligne de la figure 4.10 montre un exemple de test de 100 échantillons tirés au hasard dans la base initiale.

Les graphes associés à ces trois exemples sont respectivement un 30-NNG et un 15-NNG pour les images de chiffres et un 30-NNG pour les images de visages. Chaque sommet du graphe est associé à une image ($\text{card}(V) = 100$) et est décrit par un vecteur d'attributs de 256 dimensions ($\mathbb{R}^{16 \times 16}$) dans le cas des chiffres, et un vecteur de 841 dimensions ($\mathbb{R}^{29 \times 29}$) pour les images de visages (chaque attribut du vecteur correspond à un niveau de gris d'une imagerie).

Les résultats présentés montrent les opérations de dilatation, d'érosion et d'ouverture sur ces variétés d'images. Nous pouvons constater l'effet simplificateur et filtrant de l'opération d'ouverture sur les données. En effet, cette opération tend à réduire les données et à créer de nouveaux modèles moyens simplifiant ainsi les données. Nous pouvons remarquer que ces résultats sont comparables aux résultats présentés dans le filtrage de données avec une régularisation discrète sur graphes présentés dans le chapitre 3 (figures 3.2 à la page 58 et 3.9 à la page 67). Finalement, les traitements morphologiques sur les variétés et bases de données peuvent être appliqués comme des étapes de pré-traitements et peuvent être utiles pour des méthodes de fouille de données telles que la classification, le regroupement ou l'apprentissage de variétés.

4.5 Conclusion

Dans ce chapitre, nous avons présenté une nouvelle approche inspirée des opérations de dilatation et d'érosion basée sur des équations aux différences partielles définies sur des graphes. Notre formulation étend les approches basées sur les équations

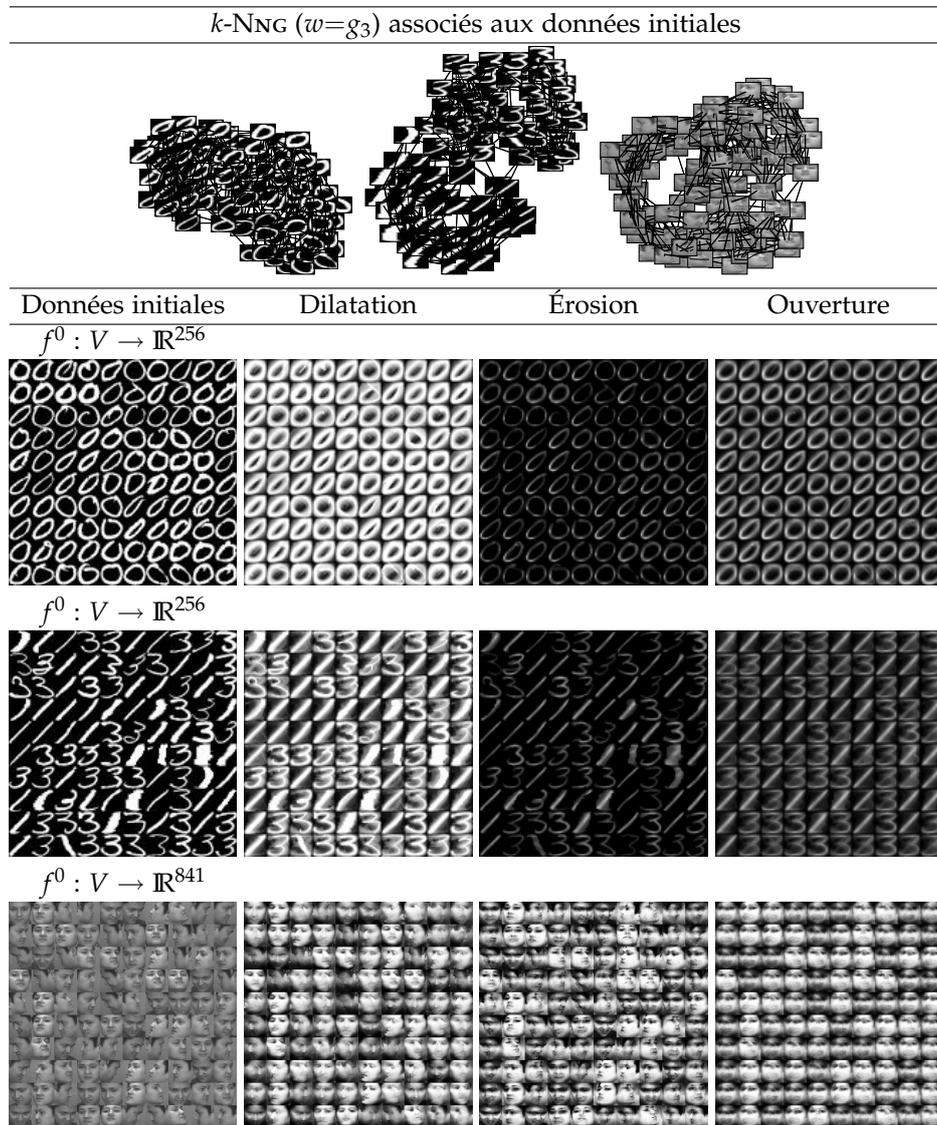


FIG. 4.10: Traitements morphologiques de données de grande dimension non organisées. À la première ligne : les k -NNG pondérés associés aux données initiales : 30-NNG pour les chiffres « 0 » ; 15-NNG pour les chiffres « 3-1 » et 30-NNG pour les visages. Au trois dernières lignes et de gauche à droite : les données initiales et les résultats d'une dilatation, d'une érosion et d'une ouverture. Les données initiales sont définies comme des fonctions $f^0 : V \rightarrow \mathbb{R}^{256}$ pour les images de chiffres et comme $f^0 : V \rightarrow \mathbb{R}^{841}$ pour les images de visages.

tions aux dérivées partielles vers des schémas discrets locaux et non locaux. Basée sur des graphes, notre approche permet de traiter n'importe quelles données discrètes et notamment des données de grande dimension définies sur des domaines réguliers ou irréguliers.

Notre formalisme unifie les schémas locaux et non locaux et nous permet d'in-

roduire une nouvelle classe d'opérateurs morphologiques pour le traitement des images en considérant des configurations non locales. Nous avons montré que ces nouveaux opérateurs permettent de mieux capturer et préserver les structures fines et répétitives dans les images.

Notre formulation constitue un cadre simple et unificateur permettant de retrouver des schémas numériques employés dans les approches de la morphologie mathématique définie par des équations aux dérivées partielles mais également par la morphologie mathématique définie sur des graphes.

À travers différentes expérimentations, nous avons présenté les potentialités et l'adaptabilité de notre méthodologie pour résoudre certains problèmes liés aux traitements des images. Nous avons également montré que l'utilisation de graphes basés sur les régions de l'image permet de réduire la complexité des algorithmes rendant le traitement morphologique des images plus rapide. Finalement, nous avons présenté le traitement morphologique de données arbitraires telles que des bases de données ou des variétés. Ce type de traitement peut être utile et être appliqué comme une étape de pré-traitement à des méthodes de classification ou de regroupement.

Ce dernier point permet d'ouvrir de nouvelles perspectives intéressantes pour la morphologie mathématique : l'application d'opérateurs morphologiques dans des communautés telles l'apprentissage ou la fouille de données. L'avantage de notre approche est de pouvoir appliquer ces méthodes à d'autres domaines que celui du traitement des images et d'ouvrir d'autres applications à la morphologie mathématique.

De plus, notre formalisme général et simple permet de revisiter d'autres opérateurs ou formulations de la morphologie mathématique que ceux que nous avons présenté dans ce chapitre tels que la morphologie mathématique visqueuse ou la morphologie fonctionnelle.

Finalement, une autre extension de ces travaux serait de considérer une nouvelle classe d'opérateurs morphologiques algébriques intégrant la notion de non localité, au travers de la fonction de poids ou de la configuration des graphes pour le traitement des images. Cette notion de pondération des arêtes du graphe permet également d'envisager une formulation simple pour la généralisation de la notion d'élément structurant rendant les opérations de morphologie mathématique naturellement adaptatives.

Dans le chapitre suivant, nous montrons comment à partir de notre approche des opérations de dilatation et d'érosion, nous pouvons résoudre l'équation eikonale en exploitant le formalisme des équations aux différences partielles définies sur des graphes.

Chapitre

5

Équation eikonale

Sommaire

5.1	Introduction	149
5.2	Adaptation de l'équation eikonale	153
5.3	Schémas numériques et algorithmes associés	154
5.4	Expérimentations	157
5.5	Conclusion	173

Résumé

Dans ce chapitre, nous présentons une adaptation de l'équation eikonale en considérant cette équation sur des graphes pondérés. Cette nouvelle approche de la résolution de l'équation eikonale est basée sur la famille de gradients directionnels définis par des équations aux différences partielles. Notre approche étend les approches de résolution connues de cette équation pour le traitement de données non organisées définies sur des domaines irréguliers sans étape de discrétisation préalable.

L'équation eikonale permet de résoudre de nombreuses applications en traitement des images, en vision par ordinateur ou en informatique graphique. Parmi toutes ces applications, nous nous intéressons dans ce chapitre aux calculs de distances et introduisons les schémas non locaux permettant de segmenter des images. Nous introduisons également de nouvelles applications de cette équation dans des domaines inhabituels tels que le regroupement ou la classification de bases de données.

Mots clés

Équation eikonale, graphes pondérés, calculs de distances généralisées, segmentations non locales des images, traitements de données non organisées de grande dimension.

Contenu du chapitre

- La section 5.1 introduit ce chapitre en rappelant des définitions sur les équations d’Hamilton-Jacobi et replace l’équation eikonale vis-à-vis d’elles. Cette section rappelle également les principales méthodes de la littérature permettant de résoudre l’équation eikonale.
- La section 5.2 présente notre adaptation de l’équation eikonale sur des graphes et fait le lien avec les processus morphologiques. Ces liens nous permettent d’exploiter nos résultats sur la morphologie mathématique pour transcrire l’équation eikonale vers des équations aux différences partielles.
- La section 5.3 présente nos schémas numériques permettant d’approximer la solution de l’équation eikonale. Cette section présente également les liens entre nos schémas et certaines approches de la littérature.
- La section 5.4 présente l’application de notre résolution de l’équation eikonale à travers différentes expérimentations. Nous exploitons notre formalisme basé sur les graphes pour traiter différents type de données. Nous présentons des exemples de calculs de distances et des segmentations non locales pour des images. Nous montrons également l’extension de ces méthodes pour le traitement de données non organisées de grande dimension et notamment pour des applications telles que l’ordonnancement, ou la classification de bases de données.
- La section 5.5 conclut ce chapitre et donne des perspectives par rapport à l’approche proposée.

Publications associées à ce chapitre

Conférences internationales avec comités de lecture et actes

- [1] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Adaptation of eikonal equation over weighted graphs. *Dans Proceedings of 2nd International Conference on Scale Space and Variational Methods in Computer Vision (SsVM)*, LNCS, pages 187 – 199. Springer, 2009.

Conférences nationales avec comités de lecture et actes

- [1] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Morphologie mathématique sur graphes pondérés de topologie arbitraire — traitements non locaux d’images et de données multi-variées. *Dans Actes du groupe de recherche et d’études du traitement du signal et des images (GRETSI)*, 2009.

Rapports techniques et articles soumis

- [1] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Mathematical morphology and eikonal equations on graphs for nonlocal image and data processing. Rapport technique, GREYC CNRS UMR 6072 - Université de Caen Basse-Normandie – ENSICAEN - hal-00365431, mars 2009. Soumis à IEEE Transactions on Image Processing.

5.1 Introduction

La résolution de l'équation eikonale a de nombreuses applications dans des domaines variés tels l'optique géométrique, l'analyse des images, la vision par ordinateur, ou encore l'informatique graphique.

Plus particulièrement, en vision par ordinateur, nous pouvons citer des applications telles que l'extraction de l'axe médian ou de squelettes, la reconstruction de formes à partir des ombres (*shape from shading*) ou l'extraction de structures tubulaires et de chemins minimaux dans des images 2D ou 3D [Siddiqi *et al.*, 1999, Rouy et Tourin, 1992, Verbeek et Verwer, 1990, Benmansour et Cohen, 2009, Deschamps et Cohen, 2001]. En informatique graphique, la résolution de l'équation eikonale permet le calcul des distances géodésiques sur des surfaces paramétriques et discrètes à partir de maillages structurés ou non définis dans des domaines Cartésiens ou non [Bronstein *et al.*, 2007, Sethian et Vladimirsky, 2003, Abgrall, 1996, Shu et Zhang, 2003].

L'équation eikonale est un cas particulier des équations d'Hamilton-Jacobi statiques [Zhang *et al.*, 2006] définies par

$$\begin{cases} H(x, f, \nabla f) = 0 & x \in \Omega \setminus \Gamma \\ f(x) = \phi(x) & x \in \Gamma \subset \Omega \end{cases} \quad (5.1)$$

où Ω est un domaine défini dans \mathbb{R}^m , Γ est un sous ensemble de Ω et ϕ une fonction permettant initialisation des points appartenant à Γ . L'Hamiltonien H est une fonction Lipschitz continue non linéaire.

Une autre manière de voir la famille d'équations d'Hamilton-Jacobi (5.1) est de considérer une version temporelle de ces équations (*time marching approach*) :

$$\begin{cases} \frac{\partial f(x, t)}{\partial t} + H(x, f, \nabla f) = 0 & x \in \Omega \setminus \Gamma \\ f(x, t) = \phi(x) & x \in \Gamma \subset \Omega \\ f(x, 0) = \phi_0(x) & x \in \Omega \end{cases} \quad (5.2)$$

et de faire évoluer ce système jusqu'à la convergence (c'est-à-dire que quand $t \rightarrow \infty$, le système (5.2) satisfait l'équation statique (5.1)). La fonction ϕ_0 est une fonction permettant d'initialiser l'ensemble des points du domaine Ω . La fonction ϕ est une fonction permettant de maintenir les valeurs des points de Γ durant l'évolution.

L'équation eikonale est un cas particulier de cette famille d'équations statiques d'Hamilton-Jacobi. La version isotrope de l'équation eikonale peut se définir en utilisant l'Hamiltonien suivant

$$H(x, f, \nabla f) = \|\nabla f(x)\|_2 - P(x) \quad (5.3)$$

où P est une fonction strictement positive (ou négative), ∇ correspond à l'opérateur gradient et $\|\cdot\|_2$ est la norme \mathcal{L}_2 usuelle. En remplaçant cet Hamiltonien dans les systèmes (5.1) et (5.2), la version isotrope stationnaire de l'équation eikonale est

$$\begin{cases} \|\nabla f(x)\|_2 = P(x) & x \in \Omega \setminus \Gamma \\ f(x) = \phi(x) & x \in \Gamma \subset \Omega \end{cases} \quad (5.4)$$

et sa version isotrope temporelle est

$$\begin{cases} \frac{\partial f(x, t)}{\partial t} = P(x) - \|\nabla f(x)\|_2 & x \in \Omega \setminus \Gamma \\ f(x, t) = \phi(x) & x \in \Gamma \subset \Omega \\ f(x, 0) = \phi_0(x) \end{cases} \quad (5.5)$$

Ces systèmes sont des équations différentielles partielles de premier ordre non linéaires. Les notions de solutions de viscosité ont été introduites afin de résoudre ce type d'équations [Crandall et Lions, 1983, Osher et Sethian, 1988, Crandall *et al.*, 1992].

De plus, en utilisant la méthode des ensembles de niveaux (*level sets*) et en considérant le problème avec une dimension plus élevée, [Osher, 1993] a montré les liens entre les équations d'Hamilton-Jacobi statiques et leur version temporelles. En effet, l'équation eikonale traduit la propagation d'un front pouvant être modélisé par l'évolution d'une courbe initiale $\Gamma_{t=0} \subset \Omega$ partant du temps $t = 0$ dans la direction de la normale sous l'influence d'une force (également appelée vitesse) égale à $1/P(x) > 0$. La solution $f(x)$ de l'équation eikonale peut alors être interprétée comme le temps nécessaire à la courbe Γ_t pour évoluer et atteindre le point x pour la première fois.

En modélisant l'évolution de la courbe Γ_t par les ensembles de niveaux, nous pouvons considérer que Γ_t est l'ensemble de niveau zéro d'une fonction $\psi(t)$. Cette fonction ψ est initialisée, à $t = 0$, comme étant négative à l'intérieur et positive à l'extérieur de la courbe initiale $\Gamma_{t=0}$. L'évolution isotrope de la fonction ψ est donnée par l'équation aux dérivées partielles suivante

$$\frac{\partial \psi}{\partial t} = -\frac{1}{P} \|\nabla \psi\|_2 \quad (5.6)$$

En résolvant cette équation aux dérivées partielles, la solution $f(x)$ peut être trouvée comme étant le temps minimal auquel le niveau zéro de la courbe ψ passe par le point x . $f(x)$ peut également être interprété comme étant la distance par rapport à la courbe initiale Γ_0 .

Remarque. *Dans ce chapitre, nous ne considérons que le cas où la fonction P est strictement positive (ou négative) ce qui implique que le front se propage avec une vitesse qui ne change pas de signe. Le lecteur intéressé par le cas où la vitesse peut changer de signe, peut se référer à [Carlini et al., 2008].*

De nombreux schémas numériques monotones basés sur des différences finies ont été développés afin de calculer de manière stable les solutions de viscosité des équations d'Hamilton-Jacobi statiques et temporelles. Il y a principalement deux classes de méthodes pour résoudre les équations d'Hamilton-Jacobi (5.1).

La première considère la version temporelle (5.2). De nombreux schémas numériques ont été développés basés pour des maillages structurées ou non. Le lecteur intéressé pourra se référer à une revue de Shu ([Shu, 2007] et les références associées) sur les méthodes numériques d'ordre supérieur pour résoudre les versions temporelles des équations d'Hamilton-Jacobi. Nous pouvons également citer un schéma numérique basé sur un semi Lagrangien et la programmation dynamique [Falcone et Ferretti, 2002].

Le second type de méthode de résolution des équations statiques d'Hamilton-Jacobi considère une version stationnaire en discrétisant le problème (5.1) en un système d'équations non linéaires et construit un algorithme permettant de le résoudre. Afin de construire un schéma numérique dans une direction *upwind* et une approximation correcte de la solution de viscosité des équations d'Hamilton-Jacobi, il est nécessaire d'avoir une bonne approximation des dérivées. Nous pouvons mentionner des approximations telles que celle de Godunov ou celle de Lax-Freidrich utilisé par [Osher et Sethian, 1988, Sethian, 1999b].

En considérant pour simplifier un domaine $\Omega \subset \mathbb{R}^2$, l'équation eikonale (5.4) se réécrit

$$\sqrt{f(x)^2 + f(y)^2} = P(x, y) . \quad (5.7)$$

Les notations standards utilisées dans les différences finies sont

$$\begin{aligned} \mathcal{D}_{i,j}^{-x} f(x_i, y_j) &= \frac{f(i, j) - f(i-1, j)}{\Delta x} & \mathcal{D}_{i,j}^{+x} f(x_i, y_j) &= \frac{f(i+1, j) - f(i, j)}{\Delta x} \\ \mathcal{D}_{i,j}^{-y} f(x_i, y_j) &= \frac{f(i, j) - f(i, j-1)}{\Delta y} & \mathcal{D}_{i,j}^{+y} f(x_i, y_j) &= \frac{f(i, j+1) - f(i, j)}{\Delta y} \end{aligned} \quad (5.8)$$

avec $f(i, j) = f(x_i, y_j)$ et $(x_i, y_j) = (i\Delta x, j\Delta y)$ où Δx et Δy correspondent au pas de discrétisation spatiale dans les directions x et y .

Avec ces notations, l'équation eikonale (5.7) s'écrit, en utilisant l'approximation de Godunov

$$\left[\max(0, \mathcal{D}_{i,j}^{-x} f, -\mathcal{D}_{i,j}^{+x} f)^2 + \max(0, \mathcal{D}_{i,j}^{-y} f, -\mathcal{D}_{i,j}^{+y} f)^2 \right]^{1/2} = P(i, j) \quad (5.9)$$

et avec l'approximation de Osher et Sethian, (5.7) s'écrit

$$\begin{aligned} & \left[\max(0, \mathcal{D}_{i,j}^{-x} f)^2 + \min(0, \mathcal{D}_{i,j}^{+x} f)^2 + \right. \\ & \left. \max(0, \mathcal{D}_{i,j}^{-y} f)^2 + \min(0, \mathcal{D}_{i,j}^{+y} f)^2 \right]^{1/2} = P(i, j) \end{aligned} \quad (5.10)$$

À partir de ces approximations, de nombreux algorithmes ont été proposés pour résoudre de manière efficace le système d'équations non linéaires décrit par l'équation eikonale. Parmi ces méthodes nous pouvons citer les suivantes.

Une méthode à point fixe résolvant l'équation quadratique de la version isotrope de l'équation eikonale [Rouy et Tourin, 1992, Jeong et Whitaker, 2008]. Cette méthode met à jour tous les points du domaine de manière itérative jusqu'à convergence.

Une méthode appelée le *fast sweeping* [Zhao, 2005, Tsai *et al.*, 2004] permettant de résoudre l'équation eikonale sur une grille de m -dimensions. Elle utilise des itérations de type Gauss-Seidel dans le schéma numérique pour mettre à jour la carte de distance. Le point clé de cette méthode est l'alternance des directions de balayage des points. Cette idée a été initialement proposée pour calculer la distance euclidienne sur des images binaires [Danielsson, 1980].

Basée sur un algorithme optimal [Tsitsiklis, 1995], la méthode du *fast marching* [Sethian, 1999a, Sethian et Vladimirovsky, 2003] est à l'heure actuelle l'algorithme le plus utilisé pour résoudre l'équation eikonale. Cette méthode met à jour les points

du domaine de manière séquentielle et toujours vers une solution strictement croissante (ou décroissante). La mise en œuvre de cette méthode nécessite, d'une part, une structure de données de type tas avec un algorithme de tri permettant d'obtenir un minimum de manière efficace et d'autre part, un schéma numérique de type *upwind*.

Remarque. Nous pouvons remarquer que des modèles digitaux issus de la morphologie mathématique peuvent être considérés comme étant des versions discrètes de l'équation eikonale. En particulier, la ligne de partage des eaux peut être interprétée comme étant une version digitale de l'équation eikonale [Najman et Schmitt, 1994].

Contributions du chapitre

Comme pour les modèles variationnels (chapitre 3) et la morphologie mathématique (chapitre 4), nous proposons dans ce chapitre de transcrire l'équation eikonale définie par des équations aux dérivées partielles vers des schémas discrets en utilisant le formalisme des équations aux différences partielles définies sur des graphes. Nous nous intéressons ici uniquement à la version temporelle de l'équation eikonale (5.5).

À partir des liens existants avec la morphologie mathématique, l'adaptation de la version temporelle de l'équation eikonale sur un graphe pondéré $G = (V, E, w)$ peut s'écrire

$$\begin{cases} \frac{\partial f(u, t)}{\partial t} = P(u) - \|(\nabla_w^- f)(u)\|_q & u \in V \setminus V_0 \\ f(u, t) = \phi(u) & u \in V_0 \subset V \\ f(u, 0) = \phi_0(u) \end{cases} \quad (5.11)$$

où V correspond à l'ensemble des sommets du graphe G , V_0 est l'ensemble des sommets source, ∇_w^- est le gradient directionnel interne pondéré défini dans le chapitre 2 et $\|\cdot\|_q$ correspond à la norme \mathcal{L}_q .

Cette transcription de l'équation eikonale possède les mêmes avantages que les différents modèles (régularisation et morphologie mathématique) que nous avons adaptés sur des graphes et abordés dans les chapitres précédents.

Les principales contributions de ce chapitre sont les suivantes.

- Notre formulation fournit un cadre simple et unifié pour traiter n'importe quelle type de données discrètes.
- Notre approche, contrairement aux équations aux dérivées partielles, ne nécessite pas d'étape préalable de discrétisation spatiale ou de triangulation. Cette discrétisation est complexe lorsque les données à traiter sont de grande dimension ou définies sur un domaine non uniforme. Cette difficulté rend difficile l'application de l'équation eikonale à ce type de données pour les méthodes définies dans le domaine continu.
- Comme pour les modèles de régularisation et de la morphologie mathématique, notre approche de l'équation eikonale unifie les configurations locales et non locales. Dans le contexte du traitement des images, cette unification permet d'introduire par exemple des schémas non locaux dans la segmentation des images.

- Finalement, notre approche permet d'appliquer l'équation eikonale pour le traitement de données ouvrant ainsi de nouvelles applications de l'équation dans les domaines de l'apprentissage ou de la fouille de données telles que l'ordonnement ou la classification de bases de données ou de variétés d'images.

5.2 Adaptation de l'équation eikonale

Dans cette section, nous rappelons quelques définitions permettant d'arriver à l'adaptation de l'équation eikonale. Cette adaptation est basée sur les opérateurs directionnels définis dans le chapitre 2. Nous rappelons également les liens existants entre l'équation eikonale, l'évolution d'une courbe et les processus morphologiques de dilatation et d'érosion.

La version isotrope de l'équation eikonale définie par les équations aux dérivées partielles est

$$\begin{cases} P(x) - \|\nabla f(x)\|_2 = 0 & x \in \Omega \setminus \Gamma \\ f(x) = \phi(x) & x \in \Gamma \subset \Omega \end{cases} . \quad (5.12)$$

Cette équation traduit l'évolution d'une courbe (ou d'une surface) en utilisant la méthode des ensembles de niveaux [Osher et Sethian, 1988, Sethian, 1999b]. La fonction f calculée peut être interprétée comme la distance à la courbe initiale.

L'évolution de la courbe peut être interprétée comme étant des processus morphologiques [Arehart *et al.*, 1993]. En effet, les contours obtenus par des dilata-tions ou des érosions planaires multi échelles par des disques unitaires peuvent être considérés comme la propagation d'un front dépendante d'une force et diri-gée dans la direction de la normale. La dilatation (δ) et l'érosion (ε) définies par des équations aux dérivées partielles, d'une fonction f par des disques unitaires $B = \{x : \|x\|_2 \leq 1\}$ induites par la norme \mathcal{L}_2 sont

$$\frac{\partial \delta(f)}{\partial t} = +\|\nabla f\|_2 \quad \text{et} \quad \frac{\partial \varepsilon(f)}{\partial t} = -\|\nabla f\|_2 . \quad (5.13)$$

où ∇ désigne l'opérateur gradient de la fonction f . Les conditions initiales de ces équations d'évolution sont $\delta(f)(x, 0) = \varepsilon(f)(x, 0) = f(x)$.

En utilisant la méthode des ensembles de niveaux, l'évolution de la courbe peut s'exprimer par l'équation aux dérivées partielles suivante :

$$\frac{\partial \psi}{\partial t} = -\frac{1}{P} \|\nabla \psi\|_2 \quad (5.14)$$

où ψ correspond à la fonction de niveau et ψ_0 peut être considérée comme étant la fonction distance à la courbe initiale. La solution f représente alors le temps minimal auquel le niveaux zéro de la courbe de $\psi(x, t)$ passe par x .

Nous pouvons voir que si $P = 1$ ou -1 alors l'équation d'évolution (5.14) cor-respond aux équations aux dérivées partielles d'évolution permettant d'obtenir la dilatation et l'érosion planaires (5.13). Cela revient à dilater ou à éroder la courbe avec un disque de taille tB et de considérer le bord extérieur de la nouvelle courbe au temps t .

Dans le chapitre 4, nous avons vu que les versions des équations d'évolution (5.13) définies sur un graphe pondéré $G = (V, E, w)$ étaient

$$\frac{\partial \delta(f)}{\partial t} = \frac{\partial f}{\partial t} = +\|\nabla_w^+ f\|_q \quad \text{et} \quad \frac{\partial \varepsilon(f)}{\partial t} = \frac{\partial f}{\partial t} = -\|\nabla_w^- f\|_q . \quad (5.15)$$

où f^0 est une fonction initiale (définie dans l'espace $\mathcal{H}(V)$) et les conditions initiales sont à $t = 0$, $f = f^0$. $\|\cdot\|_q$ désigne la norme \mathcal{L}_q avec $0 < q < +\infty$.

À la lumière des liens existants entre les processus morphologiques, l'évolution des courbes et l'équation eikonale, nous pouvons transcrire directement l'équation eikonale basée sur les équations aux dérivées partielles en utilisant les équations aux différences partielles définies sur des graphes grâce aux processus morphologiques associés.

Ainsi, pour n'importe quelle norme \mathcal{L}_q , nous avons pour un graphe $G = (V, E, w)$ et une fonction $f \in \mathcal{H}(V)$

$$\begin{cases} \frac{\partial f(u, t)}{\partial t} = P(u) - \|(\nabla_w^- f)(u)\|_q & u \in V \setminus V_0 \\ f(u, t) = \phi(u) & u \in V_0 \subset V \\ f(u, 0) = \phi_0(u) \end{cases} . \quad (5.16)$$

où V_0 correspond à l'ensemble des sommets source. À convergence de ce processus ($t \rightarrow \infty$) la solution de cette équation satisfait l'équation eikonale. Nous pouvons constater que lorsque la fonction potentielle P est constante, cette équation d'évolution correspond au processus d'érosion défini sur des graphes (chapitre 4).

Remarque. L'équation eikonale peut également s'exprimer à l'aide de l'Hamiltonien suivant

$$H(x, f, \nabla f) = P(x) - \|\nabla f(x)\|_q . \quad (5.17)$$

Dans ce cas, la version temporelle de l'équation eikonale définie sur des graphes est similaire à un processus morphologique de dilatation et l'opérateur gradient est remplacé par le gradient morphologique externe ∇_w^+ . Le système se réécrit de la manière suivante

$$\begin{cases} \frac{\partial f(u, t)}{\partial t} = \|(\nabla_w^+ f)(u)\|_q - P(u) & u \in V \setminus V_0 \\ f(u, t) = \phi(u) & u \in V_0 \subset V \\ f(u, 0) = \phi_0(u) \end{cases} . \quad (5.18)$$

La solution (à convergence) de cette équation satisfait également l'équation eikonale. Néanmoins, dans ce cas, la solution obtenue est une fonction monotone strictement décroissante. Comme nous désirons obtenir une fonction monotone strictement croissante qui est plus consistante avec la notion de distance, nous privilégions et utilisons l'Hamiltonien de la forme

$$H(u, f, \nabla f) = \|\nabla f(u)\|_q - P(u) . \quad (5.19)$$

5.3 Schémas numériques et algorithmes associés

Dans cette section, nous nous intéressons à la résolution de la version temporelle de l'équation eikonale définie sur des graphes et présentons les schémas numériques et les algorithmes permettant d'approximer la solution de l'équation eikonale.

Nous montrons également qu'avec une structure de graphe et une fonction de poids adaptées, nos schémas de résolution permettent de retrouver des schémas connus de la littérature tels que le schémas de Osher et Sethian sur une grille de m dimensions et les algorithmes de plus court chemins de type Dijkstra.

Avec l'expression discrète du gradient directionnel pondéré interne, nous obtenons directement le schéma numérique permettant d'approximer l'équation eikonale sans étape de discrétisation spatiale.

En utilisant une discrétisation temporelle et avec la notation conventionnelle telle que $f^n(u) \approx f(u, n\Delta t)$ pour un sommet donné $u \in V$, l'approximation de la solution de l'équation eikonale, pour un graphe pondéré $G = (V, E, w)$, est obtenue avec l'algorithme suivant

$$\frac{f^{n+1}(u) - f^n(u)}{\Delta t} = P(u) - \|\nabla_w^- f^n(u)\|_p . \quad (5.20)$$

Avec les normes \mathcal{L}_q correspondantes du gradient directionnel interne, nous obtenons pour les valeurs particulières de $q = 1, 2$ et ∞ les expressions suivantes. Pour $p = 1$

$$\frac{f^{n+1}(u) - f^n(u)}{\Delta t} = P(u) - \sum_{u \sim v} \sqrt{w_{uv}} |\min(0, f^n(v) - f^n(u))| , \quad (5.21)$$

pour $p = 2$

$$\frac{f^{n+1}(u) - f^n(u)}{\Delta t} = P(u) - \sqrt{\sum_{u \sim v} w_{uv} |\min(0, f^n(v) - f^n(u))|^2} , \quad (5.22)$$

et pour $p = \infty$

$$\frac{f^{n+1}(u) - f^n(u)}{\Delta t} = P(u) - \max_{v \sim u} \left(\sqrt{w_{uv}} |\min(0, f^n(v) - f^n(u))| \right) . \quad (5.23)$$

La convergence de ces processus itératifs satisfait la solution de l'équation eikonale. La convergence est obtenue soit avec un nombre fixé de n itérations ou en considérant un critère d'arrêt tel que $|f^{n+1} - f^n| \leq \epsilon$ avec $\epsilon > 0$ et $\epsilon \rightarrow 0$.

Notre approche de la résolution de l'équation eikonale permet de considérer tous types de données. Dans la section 5.4, nous montrerons l'application de ces schémas sur différents types de données tels que des images, des nuages de points, des maillages ou encore des bases de données.

5.3.1 Relations avec le schéma de Osher et Sethian

Comme pour le cas de la morphologie mathématique (chapitre 4), nos schémas numériques permettent de retrouver le schéma d'approximation de proposé par Osher et Sethian sur une grille à m -dimensions [Osher et Sethian, 1988].

Soit $G = (V, E, w)$ un graphe pondéré. Dans le cas où $q = 2$, le schéma numérique permettant de résoudre l'équation eikonale est

$$\begin{aligned} \frac{f^{n+1}(u) - f^n(u)}{\Delta t} &= P(u) - \|(\nabla_w^- f)(u)\|_2 \\ &= P(u) - \left[\sum_{u \sim v} w_{uv} |\min(0, f^n(v) - f^n(u))|^2 \right]^{1/2} . \end{aligned} \quad (5.24)$$

Si le graphe G est associé à une grille dans \mathbb{R}^m recouvrant le domaine, alors ce schéma correspond à la discrétisation de Osher et Sethian [Osher et Sethian, 1988] permettant de résoudre l'équation eikonale sur une grille de dimension m . Soit $u \in V$ un sommet du graphe G . u peut être défini comme étant un vecteur de coordonnées spatiales vecteur de coordonnées spatiales de dimension m tel que $u = (i_1 h_1, \dots, i_m h_m)^T$ où les h_k représentent l'espacement de la grille et les $i_k \in \mathbb{N}$ avec $k = 1, \dots, m$.

Le schéma (5.24) peut alors se réécrire (voir le chapitre 4 pour plus de détails)

$$\frac{f^{n+1}(u) - f^n(u)}{\Delta t} = P(u) - \left[\sum_{k=1}^m \max \left(0, \frac{f^n(u) - f^n(u - h_k e_k)}{h_k} \right)^2 + \min \left(0, \frac{f^n(u + h_k e_k) - f^n(u)}{h_k} \right)^2 \right]^{1/2} . \quad (5.25)$$

ce qui correspond au schéma de Osher et Sethian pour une grille de dimension m , avec $w_{uv_k} = 1/h_k$ pour tous les sommets du graphe et $e_k = (q_l)_{l=1, \dots, m}$ est un vecteur de \mathbb{R}^m tel que $q_l = 1$ si $k = l$ et $q_l = 0$ sinon.

5.3.2 Algorithme du type Dijkstra

Dans cette section, nous montrons que pour le cas particulier où $p = \infty$, nos schémas numériques correspondent à des algorithmes de recherches du plus court chemin dans un graphe du type algorithme de Dijkstra.

Dans le cas particulier où $p = \infty$, le schéma numérique permettant de résoudre l'équation eikonale est

$$\begin{aligned} \frac{f^{n+1}(u) - f^n(u)}{\Delta t} &= P(u) - \|(\nabla_w^- f)(u)\|_\infty \\ &= P(u) - \max_{v \sim u} \left(\sqrt{w_{uv}} \min(0, f^n(v) - f^n(u)) \right) . \end{aligned} \quad (5.26)$$

Si nous considérons une discrétisation temporelle constante ($\Delta t = 1$) et un graphe $G = (V, E, g_1)$ non pondéré pouvant représenter des données dans \mathbb{R}^m alors le schéma précédent se réécrit

$$f^{n+1}(u) = f^n(u) - \max_{v \sim u} \left(\min(0, f^n(v) - f^n(u)) \right) + P(u) . \quad (5.27)$$

Avec $|\min(0, a - b)| = |\max(0, b - a)|$ et $\max(0, a - b) = -\min(0, b - a)$, nous avons

$$\begin{aligned} f^{n+1}(u) &= f^n(u) - \max_{v \sim u} \left(\max(0, f^n(u) - f^n(v)) \right) + P(u) \\ &= f^n(u) - \max_{v \sim u} \left(0, f^n(u) - f^n(v) \right) + P(u) \\ &= f^n(u) + \min_{v \sim u} \left(0, f^n(v) - f^n(u) \right) + P(u) \end{aligned} \quad (5.28)$$

Finalement, avec $\min(0, a - b) = \min(b, a) - a$, nous avons

$$f^{n+1}(u) = f^n(u) + \min_{v \sim u} (f^n(u), f^n(v)) - f^n(u) + P(u) . \quad (5.29)$$

En considérant que le voisinage d'un sommet u est défini comme $\mathcal{N}(u) \cup u$, l'algorithme devient

$$f^{n+1}(u) = \min_{v \sim u} (f^n(v)) + P(u) . \quad (5.30)$$

Cet algorithme correspond à un algorithme de type Dijkstra de recherche de plus court chemin dans un graphe de topologie arbitraire. En effet, à chaque étape de l'algorithme, la distance $f(u)$ en un sommet u correspond à la distance minimale d'un de ses voisins.

5.4 Expérimentations

Dans cette section, nous appliquons notre adaptation de l'équation eikonale pour le traitement de données arbitraires représentables par des graphes.

Dans la suite, différentes expérimentations utilisant l'équation eikonale seront proposées notamment pour le calcul de distance et la segmentation d'image et de données. Nous proposerons également des applications de l'équation eikonale dans des domaines tels que l'apprentissage ou la fouille de données pour réaliser de l'ordonnement ou la classification de bases de données.

L'objectif des expérimentations qui vont suivre n'est pas de résoudre une application particulière mais uniquement de montrer le comportement et les potentialités de notre méthodologie.

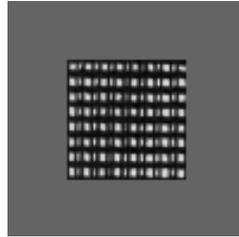
5.4.1 Calcul de distances généralisées

Dans les expérimentations de cette section, nous montrons l'adaptabilité de notre méthodologie en terme de structures de graphes, de fonctions de poids pour la propagation de fronts et le calcul de distances généralisées pour les images et leurs extensions à des graphes arbitraires. Tous les résultats présentés dans cette section sont obtenus avec un potentiel $P = 1$.

Calculs de distances pour les images

La figure 5.1 montre un exemple de calcul de distances sur une image en niveaux de gris définie comme une fonction $f^0 : V \subset \mathbb{Z}^2 \rightarrow \mathbb{R}$ (figure 5.1a). Cette expérimentation montre l'adaptabilité de notre méthodologie en terme de valeur du paramètre q , structures de graphes et de fonctions de poids.

Le marqueur initial est situé dans le coin en haut à gauche de l'image initiale. Les trois lignes de la figure 5.1 montrent respectivement les résultats pour les cas particuliers de la norme \mathcal{L}_q avec $q = 2, 1$ et ∞ . Tous les résultats correspondent à des cartes de distances où pour chaque image la couleur rouge illustre une distance faible et la couleur bleue illustre une distance élevée du pixel par rapport au marqueur initial. Pour une meilleure visualisation, nous avons également superposé en blanc certaines lignes d'iso-contours. Le deux premières colonnes de la figure 5.1 montrent des résultats obtenus à partir de graphes non pondérés ($w = g_1$). La première colonne utilise un graphe grille en 4-voisinage G_0 ce qui correspond aux cas pouvant être obtenus à partir des méthodes de la littérature.



(a) Image initiale
 $f^0 : V \subset \mathbb{Z}^2 \rightarrow \mathbb{R}$

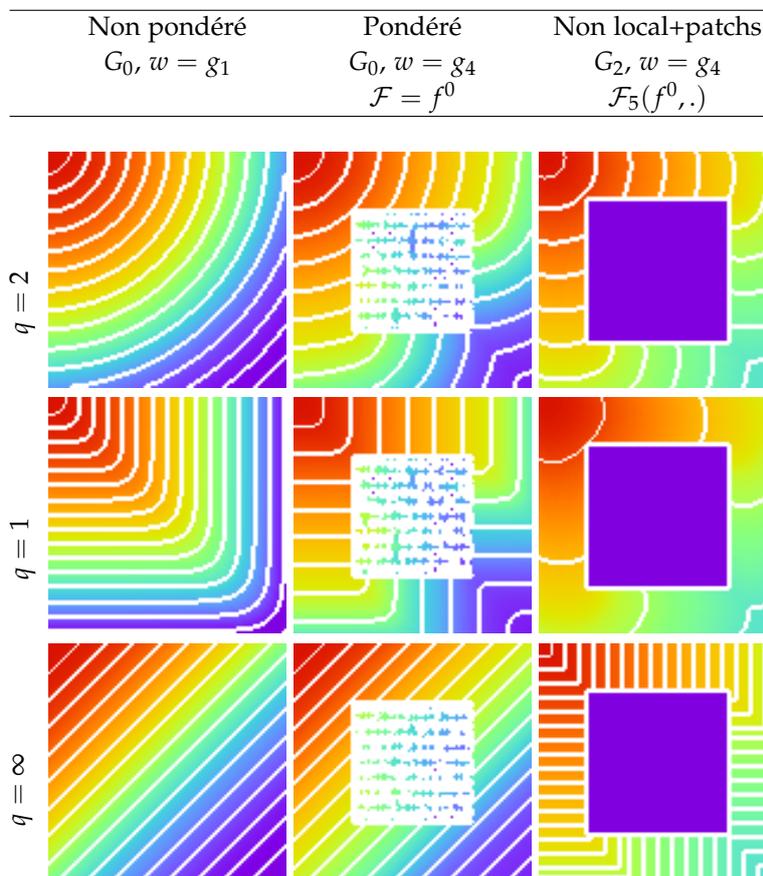


FIG. 5.1: Calculs de distances adaptatifs avec différentes valeurs de q , de fonctions de poids w et de topologies de graphe G . (a) : l'image initiale. À la première, la deuxième et la dernière ligne respectivement : cas où $q = 2$, $q = 1$ et $q = \infty$. Chaque colonne présente des résultats obtenus avec des structures de graphes et des fonctions de poids différentes calculée à partir de (a). Chaque résultat correspond à une carte de distances où les iso-contours sont superposés. Le marqueur initial correspond à un pixel qui est situé dans le coin en haut et à gauche de l'image initiale.

Les deux dernières colonnes de la figure 5.1 montrent l'utilisation de graphes pondérés où la fonction de poids associées aux arêtes est calculée à partir de l'image

initiale 5.1a.

La deuxième colonne montre les résultats avec un graphe grille en 4-voisinage G_0 pondéré par la fonction de poids g_4 . Nous pouvons voir que l'utilisation d'un poids permet de modifier la propagation du front et plus particulièrement dans la sous région comportant de la texture. La dernière colonne présente des résultats avec des schémas non locaux. Le graphe utilisé est un graphe en 25-voisinage pondéré (G_2) avec la fonction de poids (g_4) et utilisant comme vecteur d'attributs, en chaque sommet, un patch de taille 11×11 ($\mathcal{F}_5(f^0, \cdot)$) centré sur le sommet considéré. Ces résultats montrent clairement que l'intégration de schémas non locaux dans la résolution de l'équation eikonale permet d'obtenir des distances prenant en compte les structures répétitives contenues dans les images. Cela a pour effet de stopper la propagation du front autour de la sous région texturée rendant les pixels contenus dans cette partie de l'image initiale très éloignés par rapport au marqueur initial. Nous pouvons également remarquer qu'un simple seuillage de la carte de distances ainsi obtenue permet de réaliser une segmentation de la sous région.

Finalement, nous pouvons constater que l'utilisation d'une fonction de poids non constante permet d'intégrer automatiquement les informations provenant de l'image dans le calcul des distances sans l'utilisation de potentiel. Nous rappelons que le potentiel P utilisé pour obtenir l'ensemble des résultats de la figure 5.1 est égal à 1.

Calculs de distances pour des données organisées

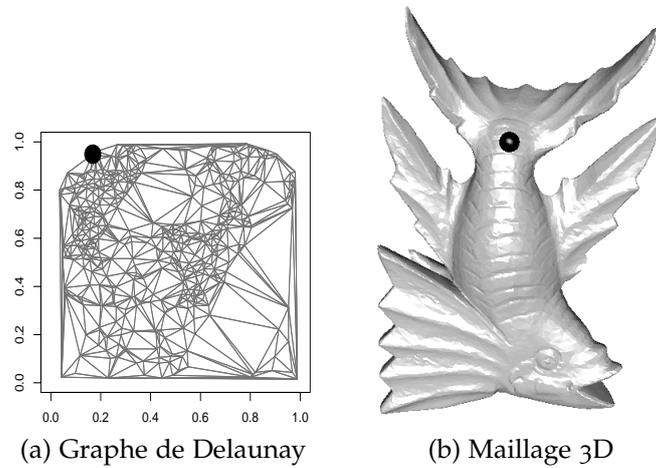
Notre méthodologie permet d'appliquer l'équation eikonale sur n'importe quel type de graphes.

La figure 5.2 montre un exemple de calculs de distances sur des graphes représentant des données organisées 2D et 3D. Les graphes utilisés sont un graphe de Delaunay et un maillage 3D.

Le graphe de Delaunay (figure 5.2a) est obtenu avec un diagramme de Voronoï sur le nuage de points 2D. Dans le cas des données 3D (figure 5.2b), nous utilisons la représentation naturelle en graphe du maillage, c'est-à-dire que les sommets et les arêtes du graphe correspondent à ceux du maillage. Afin de tenir compte de la position spatiale des sommets, ces deux graphes sont pondérés avec la fonction de poids g_2 .

Les deux premières lignes de la figure 5.2 montrent, de gauche à droite, l'évolution du front sur les graphes. Chaque résultat montre la carte de distance obtenue à partir de notre résolution de l'équation eikonale où la couleur rouge représente des distances proches et la couleur bleu, des distances éloignées par rapport aux marqueurs initiaux (disque noir sur les graphes originaux figures 5.2a et 5.2b). La dernière colonne de cette figure montre les cartes de distances finales obtenues à convergence.

Finalement, cette expérimentation illustre bien que notre méthodologie pour résoudre l'équation eikonale peut être facilement utilisée pour calculer des distances sur n'importe quel type de graphe.



De gauche à droite : évolution du front

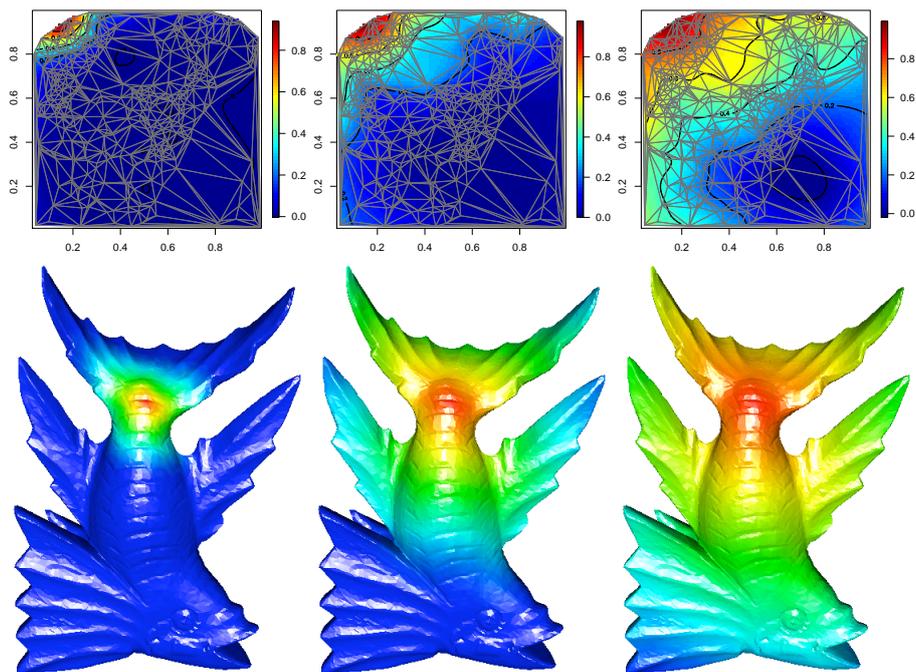


FIG. 5.2: Propagation et évolution de fronts sur un graphe de Delaunay et un maillage 3D. (a) et (b) : graphes originaux. Sur chaque ligne et de gauche à droite : évolution du front pour Les marqueurs initiaux sont représentés par un disque noir sur les graphes originaux. Chaque résultat correspond à une carte de distance. La dernière colonne correspond à la carte de distance finale.

Calculs de distances pour des bases de données

Les expérimentations suivantes illustrent l'application de l'équation eikonale sur des bases de données d'images. Le but est de montrer comment, à l'aide de l'équa-

tion eikonale, nous pouvons calculer des distances sur ces variétés permettant ainsi d'envisager des applications telles que l'ordonnement de données (*data ranking*) ou l'indexation.

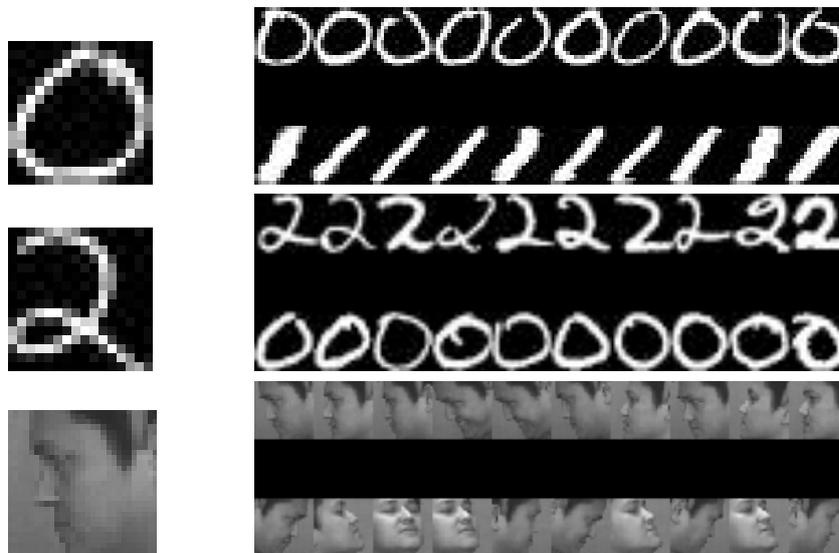
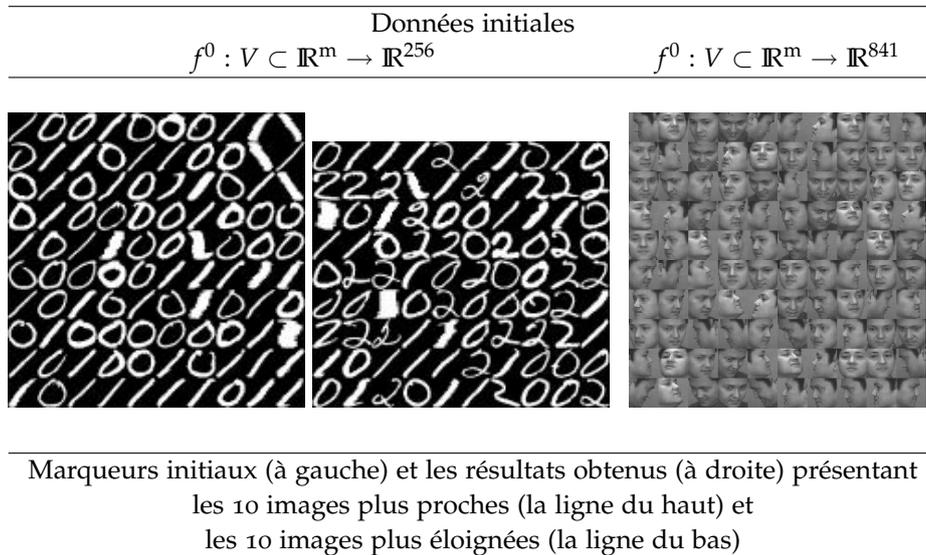
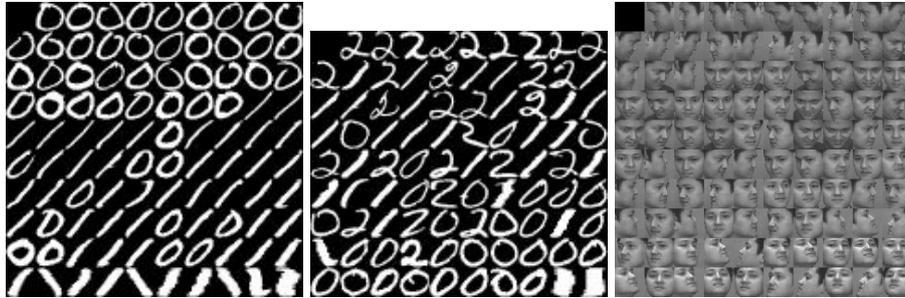


FIG. 5.3: Calcul de distance avec l'équation eikonale sur des bases de données d'images — application à l'ordonnement de variétés d'images. Dans la première partie : les données initiales définies comme des fonctions de $f^0 : V \subset \mathbb{R}^m \rightarrow \mathbb{R}^{256}$ pour la base des chiffres et $f^0 : V \subset \mathbb{R}^m \rightarrow \mathbb{R}^{841}$ pour la base des visages. Dans la seconde partie : les marqueurs initiaux, les 10 images les plus proches et les 10 images les plus éloignés.

Les figures 5.3 et 5.4 montrent un exemple d'application du calcul de distance sur des données de grande dimension non organisées : l'ordonnement de bases de données. Le but est ici de partir d'un élément (la requête) de la base de données

 Résultats obtenus avec l'algorithme des k plus proches voisins



 Résultats obtenus avec notre approche et l'équation eikonale (en haut) et les cartes de distance associées (en bas)

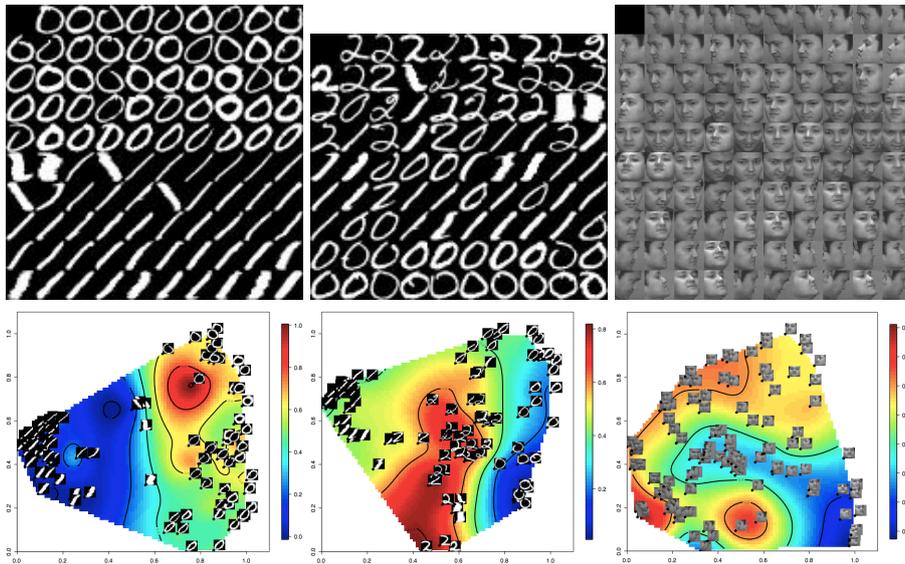


FIG. 5.4: Calcul de distance avec l'équation eikonale sur des bases de données d'images — application à l'ordonnancement de variétés d'images. Comparaison avec les résultats obtenus avec l'algorithme des k plus proches voisins utilisant la distance Euclidienne. *Chaque résultat* présenté est ordonné du plus proche (en haut à gauche) au plus éloigné (en bas à droite) en terme de distance et en accord avec les marqueurs présentés à la figure 5.3. *Dans la première partie* : les résultats obtenus avec l'algorithme des k plus proches voisins. *Dans la seconde partie* : les résultats obtenus avec notre approche (*la première ligne*) et les cartes de distances obtenues avec l'équation eikonale (*la seconde ligne*).

et de trouver dans la base les éléments qui lui sont le plus proches. Pour réaliser cet ordonnancement, nous utilisons notre approche de l'équation eikonale.

La première partie de la figure 5.3 présente les bases de données de test (qui ont déjà été utilisées dans les chapitres précédents). Les images de chiffres proviennent

de la base de l'Usps. Nous utilisons respectivement 100 et 90 échantillons tirés de manière aléatoire parmi les bases de chiffres « 0 » et « 1 » pour la première base de test et les bases de « 0 », « 1 » et « 2 » pour la seconde. Quant aux images de visages (troisième cas de la figure 5.3), nous utilisons 100 échantillons tirés au hasard dans la base initiale. Les graphes associés à ces trois exemples sont respectivement un 20-NNG et un 5-NNG pour les images de chiffres et un 3-NNG pour les images de visages. Ces trois graphes sont pondérés par la fonction de poids g_3 . Chaque sommet des graphes est associé à une image et est décrit par un vecteur d'attributs de 256 dimensions ($\mathbb{R}^{16 \times 16}$) dans le cas des chiffres, et un vecteur de 841 dimensions ($\mathbb{R}^{29 \times 29}$) pour les images de visages (chaque attribut du vecteur correspond à un niveau de gris).

La seconde partie de la figure 5.3 montre les marqueurs initiaux (les images requêtes) ainsi que les résultats obtenus. Pour chaque marqueur, les résultats montrent les 10 images les plus proches (à la ligne du haut) et les 10 images les plus éloignées (à la ligne du bas) en accord avec l'image requête. Ces images résultats proviennent de l'ordonnancement réalisé à l'aide des cartes de distances calculées à partir de l'équation eikonale. Les cartes de distances et les ordonnancements associés sont présentés dans la seconde partie de la figure 5.4.

La figure 5.4 présente également une comparaison entre l'ordonnancement réalisé à partir de notre approche de l'équation eikonale et celui obtenu avec l'algorithme des k plus proches voisins pour une distance Euclidienne.

Pour chaque image requête (marqueur) de la figure 5.3, les résultats présentés sont ordonnés de la manière suivante : l'image résultat la plus proche se trouve en haut à gauche et celle la plus éloignée se trouve en bas à droite. Nous pouvons remarquer que notre approche permet d'obtenir de meilleurs résultats (visuellement plus cohérent) que celui obtenue par l'algorithme des k -plus proches voisins. Cette exemple montre que notre approche constitue une méthode permettant d'utiliser l'équation eikonale afin de calculer des distances géodésiques dans des variétés.

Les ordres obtenus par notre approche sont réalisés simplement en triant les cartes de distances calculées à partir de l'équation eikonale.

Nous pouvons voir, dans le cas des images de visages, que l'ordre obtenu permet quasiment de retrouver la rotation complète du visage de la personne c'est-à-dire en partant d'un côté du visage à l'autre en passant par les images de faces (même si les images plus contrastées semblent influencer sur l'ordre final). Cet effet provient de la méthode et de la distance que nous utilisons pour comparer deux images. En effet, pour construire les graphes associés à chacune des bases de données, la distance utilisée dans le calcul des similarités entre les images est simplement la distance Euclidienne. La comparaison entre deux images se fait alors pixel à pixel.

Nous pouvons remarquer que, malgré l'utilisation de cette simple méthode (distance Euclidienne pixel à pixel), les résultats que nous obtenons sont visuellement satisfaisants. L'utilisation d'une distance permettant de prendre en compte les différences d'illumination ou les transformations géométriques permettraient une amélioration globale des résultats.

Ces expérimentations montrent les potentialités de notre méthodologie basée sur des graphes pour calculer de manière simple des distances géodésiques dans des variétés.

Finalement, l'application de l'équation eikonale sur des bases de données ou des variétés d'images permet d'envisager de nouvelles applications de cette équation dans des domaines tels que l'apprentissage ou la fouille de données.

5.4.2 Segmentation

La segmentation d'image est un problème fondamental en vision par ordinateur et en traitement des images. L'objectif de la segmentation d'image est de diviser l'image en régions d'intérêts ou d'extraire un ou plusieurs objets dans l'image à traiter. Parmi toutes les nombreuses méthodes de segmentation existantes dans la littérature (voir par exemple dans [Russ, 2007] et les références associées), les méthodes variationnelles et en particulier les approches basées sur les contours actifs sont très populaires (voir par exemple dans [Aubert et Kornprobst, 2006]).

Il est intéressant de noter que Cohen et Kimmel dans [Cohen et Kimmel, 1997] ont montré que les fonctionnelles mises en jeu dans les méthodes de contours actifs sont liées à l'équation eikonale et notamment à la recherche de chemins minimaux. Ceci replace donc nos travaux dans le cadre de la segmentation des images.

De plus, nous pouvons également remarquer que les méthodes de segmentation basées sur des schémas non locaux sont encore très peu étudiées dans la littérature [Gilboa et Osher, 2007a, Bresson et Chan, 2008, Houhou *et al.*, 2009] alors que ces approches ont montré leur efficacité dans les domaines du filtrage et de la restauration des images, des maillages ou de données [Buades *et al.*, 2008, Gilboa et Osher, 2007a, Elmoataz *et al.*, 2008a]

Dans cette section, nous abordons la segmentation d'image basée sur notre résolution de l'équation eikonale. Notre approche a l'avantage d'intégrer grâce à la structure de graphe les schémas locaux et non locaux dans une même formulation. Cela nous permet d'obtenir une méthode de segmentation basée sur des schémas non locaux avec l'équation eikonale. Dans la suite, nous allons montrer le bénéfice de ce type de schéma pour la segmentation d'image et plus particulièrement les images contenant des structures fines et répétitives ou du bruit.

Dans les expérimentations précédentes, nous avons montré les potentialités de nos schémas numériques pour résoudre l'équation eikonale et pour calculer des distances pondérées généralisées. En particulier, la figure 5.1 a montré le bénéfice des schémas non locaux dans le calcul des distances. Dans cette dernière expérimentation, nous avons également pu remarquer qu'une manière simple de segmenter la sous région texturée est de procéder à un simple seuillage de la carte de distances ainsi calculée.

Pour des images plus complexes nécessitant plus d'informations à priori, nous proposons d'utiliser l'équation eikonale avec une méthode de labélisation. Pour réaliser cette segmentation, nous utilisons l'équation eikonale afin de calculer c cartes de distance, où c correspond au nombre de classes (de type) d'objets à extraire. Une valeur usuelle du nombre de classes est $c = 2$, une classe désignant l'arrière-plan et une autre l'objet à extraire. La segmentation finale est alors obtenue en utilisant les distances calculées. Ce type de segmentation peut être considéré comme un problème d'apprentissage, de regroupement semi supervisé sur graphe.

Soit $G = (V, E, w)$ un graphe pondéré représentant des données qui doivent être segmentées. L'apprentissage semi supervisé des sommets V du graphe consiste à

regrouper et à classer l'ensemble V de sommets en c classes, où le nombre c est connu. L'ensemble des sommets V est composé initialement de sommets labélisés et non labélisés. L'objectif de l'apprentissage semi supervisé est alors d'estimer les sommets non labélisés à partir de ceux qui le sont. Nous pouvons remarquer que ce problème peut être vu comme un problème d'interpolation (voir chapitre 3).

Soit C_i , avec $i = 1, \dots, c$, l'ensemble des sommets de V qui sont labélisés et appartenant à la $i^{\text{ème}}$ classe. Soit $V_0 = \{C_i\}_{i=1, \dots, c}$ l'ensemble des sommets qui sont initialement labélisés et $V \setminus V_0$ l'ensemble des sommets qui ne le sont pas. Chaque sommet u de V est alors décrit par un vecteur de labels $f : V \subset \mathbb{R}^m \rightarrow \mathbb{R}^c$ tel que

$$f(u) = (f_i(u))_{i=1, \dots, c}^T . \quad (5.31)$$

En partant des sommets labélisés, l'apprentissage semi supervisé consiste à calculer c cartes de distances. Avec l'équation eikonale et pour tout sommet $u \in V$, nous avons

$$\begin{cases} \frac{\partial f_i(u, t)}{\partial t} = P(u) - \|\nabla_w^- f_i(u)\|_q \\ f_i(u, t) = \phi(u) \quad \text{si } u \in C_i, \quad \text{avec } i = 1, \dots, c \end{cases} . \quad (5.32)$$

Chaque f_i , avec $i = 1, \dots, c$, correspond à une classe donnée. À convergence de chacun de ces processus, c'est-à-dire à l'obtention des cartes de distances résultante f_i , nous affectons à un sommet u la classe i ayant la distance minimale :

$$\underset{i}{\operatorname{argmin}}(f_i(u)) \quad \text{avec } i = 1, \dots, c . \quad (5.33)$$

Notre approche est comparable au calcul d'un diagramme de Voronï.

Segmentation non locale d'images

La figure 5.5 présente un exemple d'application de la segmentation basée sur l'équation eikonale pour des images. Elle montre également le comportement et le bénéfice que peuvent apporter les schémas non locaux basés sur des patches dans l'extraction des objets contenus dans les images.

La première colonne de la figure 5.5 montre trois images initiales sur lesquelles nous avons superposé les labels initiaux. Ces images en niveaux de gris sont définies comme des fonctions $f^0 : V \subset \mathbb{Z}^2 \rightarrow \mathbb{R}$. Les marqueurs ayant la couleur verte représentent l'arrière-plan et ceux ayant la couleur rouge les objets que nous souhaitons extraire. Chaque couleur représente une classe différente. La deuxième colonne montre les images de gradient calculées à partir des images initiales. Ces gradients sont utilisés comme potentiel P . Les trois colonnes suivantes montrent les résultats de la segmentation obtenue à partir des cartes de distances calculées à convergence des processus (5.32). Sur ces résultats, nous avons superposé les frontières des régions trouvées à l'issue de la segmentation. Ces résultats sont obtenus avec le paramètre $q = 2$.

Les résultats de la troisième colonne de la figure 5.5 sont obtenus à partir d'un graphe grille en 4-voisinage G_0 non pondéré (fonction de poids g_1). Ces résultats correspondent à ceux que nous pourrions obtenir avec les méthodes usuelles de la

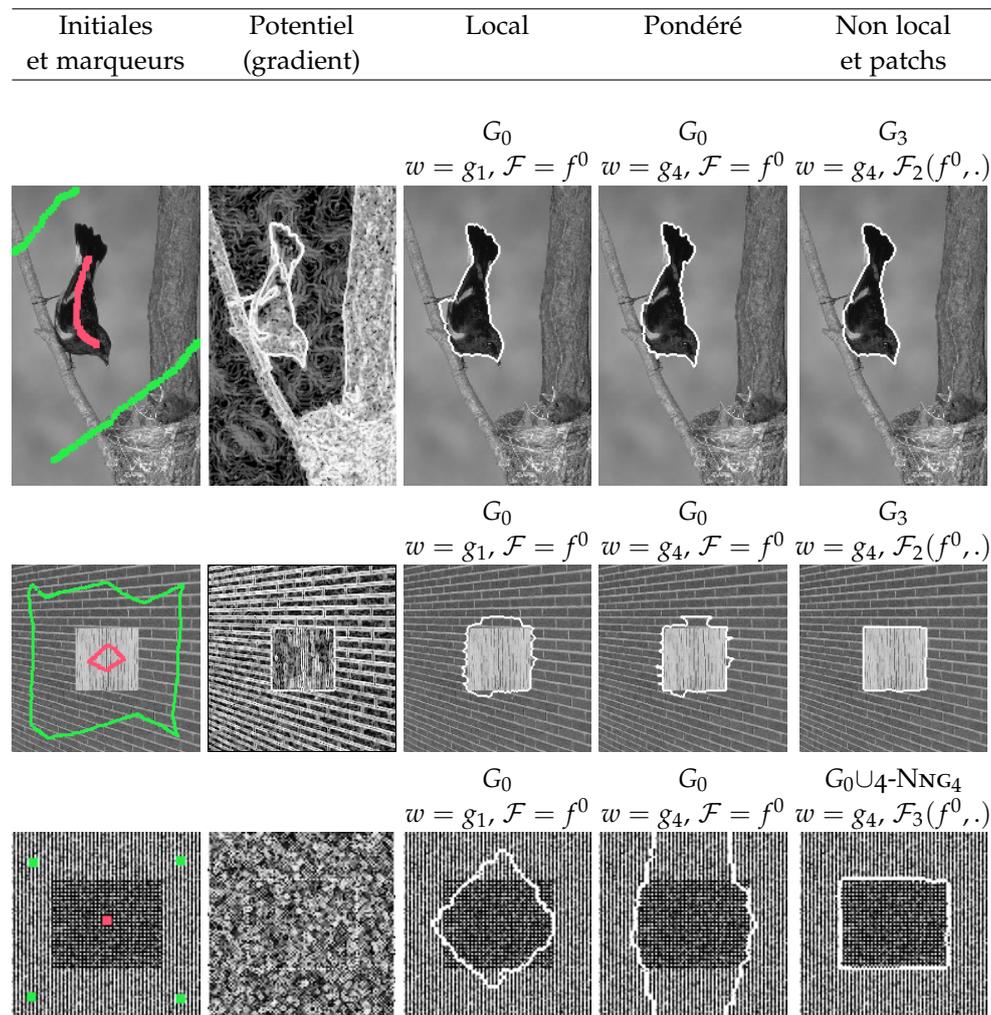


FIG. 5.5: Segmentation d'images avec l'équation eikonale avec différentes structures de graphes. À chaque ligne et de gauche à droite : l'image initiale avec les marqueurs superposés (la couleur verte représente l'arrière plan et la couleur rouge l'objet à extraire); l'image de gradient utilisée comme potentiel P ; les résultats obtenus (avec les différents graphes spécifiés) où les contours des régions trouvées sont superposées en blanc. Les images initiales sont définies comme des fonctions $f^0 : V \subset \mathbb{Z}^2 \rightarrow \mathbb{R}$.

littérature. Nous pouvons constater que les résultats sont satisfaisants même si l'oiseau n'est pas totalement segmenté et que pour l'image de texture bruitée (troisième ligne de la figure 5.5) la segmentation n'est pas correcte.

Les résultats de la quatrième colonne sont également obtenus à partir d'un graphe grille en 4-voisinage G_0 mais cette fois pondéré par la fonction de poids g_4 . Nous pouvons constater des similitudes avec le cas non pondéré même si les résultats sont meilleurs : l'oiseau est complètement segmenté et les contours sont plus proches des objets pour les deux dernières images contenant de la texture.

La dernière colonne de la figure 5.5 montre une segmentation basée sur des schémas non locaux avec patches. Les graphes associés sont pondérés avec la fonction de poids g_4 . Pour les deux premières images de test, nous avons construit des graphes grilles en 49-voisinage et utilisé, en chaque sommet, un patch de taille 5×5 comme vecteur d'attributs $(\mathcal{F}_5(f^0, \cdot))$. Ces deux premiers résultats montrent visuellement de meilleures segmentations par rapport aux deux cas précédents (local non pondéré et local pondéré) : le contour de l'oiseau est plus lisse et celui de la deuxième image vient coller aux bords de la sous région texturée. Dans le cas de la dernière image comportant de la texture bruitée, nous pouvons constater que les schémas non locaux avec des patches ont permis d'obtenir un résultat satisfaisant. Le graphe utilisé ici est un $G_0 \cup 4\text{-NNG}_4$ où les 4 plus proches voisins sont sélectionnés dans une fenêtre de taille 9×9 et pondérés par la fonction de poids (g_4) associée à des patches de taille 7×7 .

Finalement, les résultats proposés dans la figure 5.5 montrent le potentiel de notre approche de l'équation eikonale pour réaliser de la segmentation d'image et, en particulier, ils illustrent les bénéfices des schémas non locaux notamment pour des images bruitées ou comportant des structures fines et répétitives. Nous remarquons que ces derniers schémas, grâce à une configuration particulière de graphe et une fonction de poids adaptée, induisent un filtrage sous-jacent des images permettant une meilleure extraction des objets.

Segmentation avec des graphes de régions

Comme dans les chapitres précédents, nous utilisons des graphes de régions afin de traiter et segmenter les images.

Les figures 5.6, 5.7 et 5.8 montrent des exemples de segmentation avec ce type de graphe.

Tous les résultats présentés dans cette section sont obtenus en utilisant un potentiel $P = 1$. Ce sont les fonctions de poids associées aux graphes utilisés qui permettent de rendre compte des interactions entre les régions et de capturer les informations contenues dans les images à traiter.

Les premières parties des figures 5.6, 5.7 et 5.8 montrent les partitions calculées à partir des images initiales. Ces figures présentent également les images reconstruites à partir des partitions correspondantes et où chaque pixel de l'image est remplacé par la couleur moyenne de la région lui correspondant.

La figure 5.6 compare les résultats obtenus à partir d'un graphe grille de pixels en 4-voisinage et un RAG pondéré par la fonction de poids g_4 .

La seconde partie de figure 5.6 montre les résultats obtenus avec le graphe G_0 (à la première ligne) et le RAG (à la seconde ligne). Dans chaque cas, nous présentons les cartes de distances calculées en accord avec les marqueurs initiaux et la segmentation finale obtenue par seuillage de ces cartes. Les marqueurs initiaux sont symbolisés par des points en blanc sur l'image initiale 5.6a. Cette expérimentation montre des comportements et des résultats similaires tant au niveau des cartes de distance que des segmentations finales tout en réduisant le nombre de sommets à traiter.

La figure 5.7 compare l'utilisation d'un RAG et d'un RAG non local pour réaliser la segmentation.

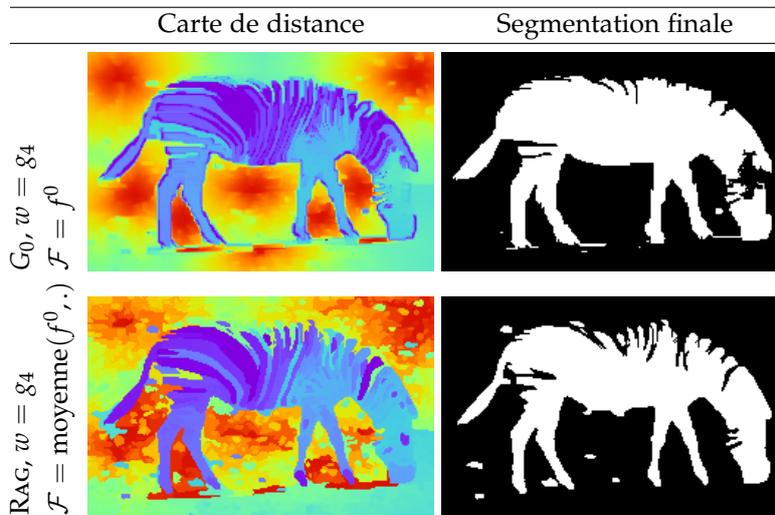
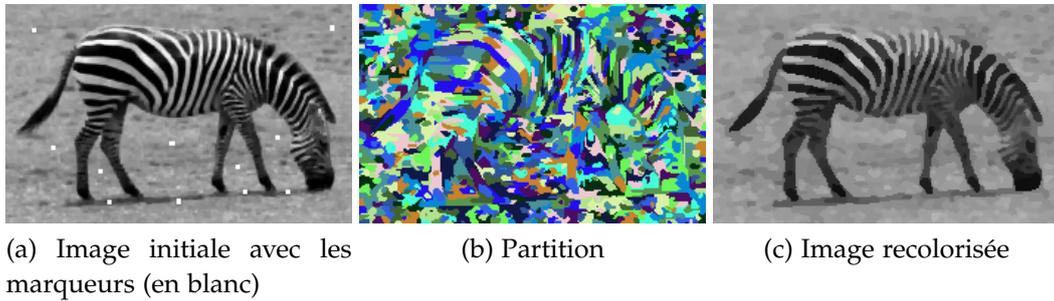


FIG. 5.6: Segmentation d'image avec graphes basés pixels et basés régions. (a) : image initiale (32 250 pixels) et les marqueurs initiaux en blanc. (b) : partition calculée à partir de (a) (999 régions soit 97% de réduction). (c) : image reconstruite à partir de la partition et de la valeur moyenne des régions. Dans la seconde partie de cette figure : à la première ligne, résultats avec un graphe grille de pixels et à la seconde ligne, résultats un RAG. À la première colonne : les cartes de distances à la seconde colonne : les segmentations obtenues par seuillage.

Les deux graphes sont construits à partir de la partition (figure 5.7b) et sont pondérés par la fonction de poids g_4 .

À chaque ligne de la seconde partie de la figure 5.7, nous présentons la carte de distance, la segmentation finale obtenue par seuillage de la carte et les contours des objets segmentés superposés en blanc sur l'image initiale.

La première ligne de la figure 5.7 montre le cas de l'utilisation d'un RAG. Le triangle marqué a été segmenté, ce qui correspond bien aux courtes distances (de couleur rouge) décrites par la carte de distance. Les autres objets ont été détectés comme éloignés (de couleur bleue) ce qui aboutit à la segmentation du triangle marqué uniquement. La seconde ligne de la figure 5.7 montre le cas de l'utilisation d'un RAG non local. Comme pour le cas précédent, les distances à l'intérieur du triangle marqué sont calculées comme proches des marqueurs initiaux. Les distances aux autres triangles contenus dans l'image ont également été calculées comme proches

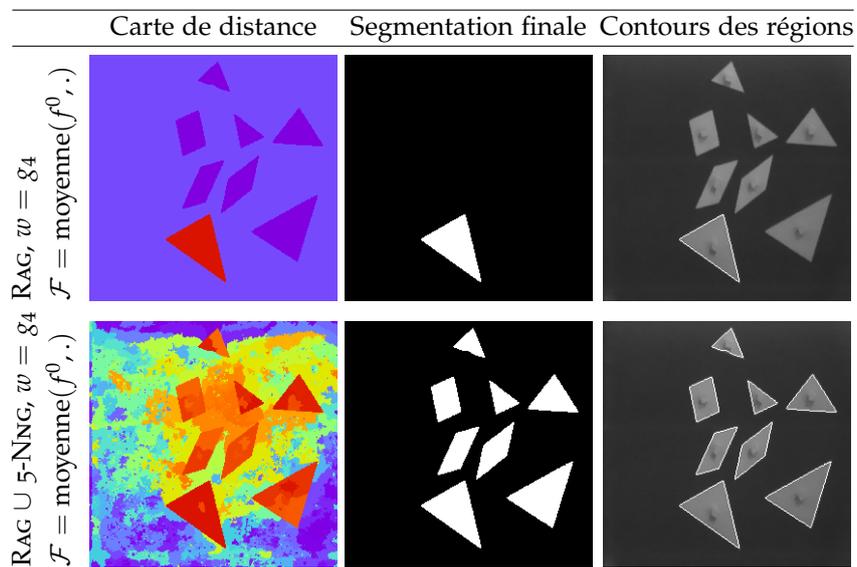
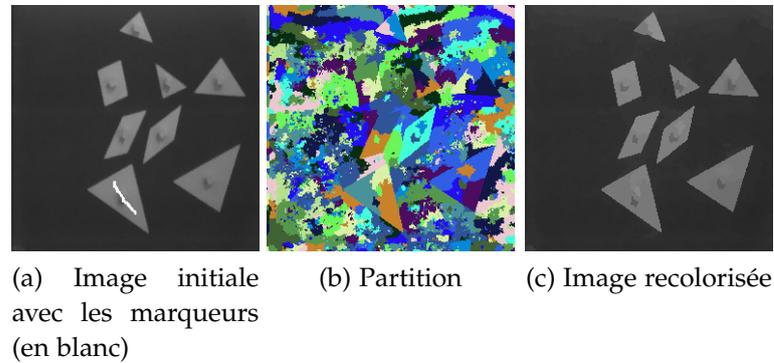


FIG. 5.7: Segmentation non locale d'image avec graphes basés régions. (a) : image initiale (65 536 pixels) et marqueurs originaux en blanc. (b) : partition calculée à partir de (a) (1 324 régions soit 98% de réduction). (c) : image reconstruite à partir de la partition et la valeur moyenne des régions. Dans la seconde partie de cette figure : à la première ligne, résultats obtenus avec un RAG à la seconde ligne, résultats obtenus avec un $\text{RAG} \cup 5\text{-NNG}$. Les deux graphes sont construits à partir de la partition et de l'image recolorisée. Pour chaque cas, de gauche à droite : la carte de distance, la segmentation calculée par seuillage et les contours des régions obtenues superposés en blanc sur l'image initiale.

des marqueurs initiaux. Finalement, par seuillage de cette nouvelle carte de distance, nous obtenons une segmentation complète de l'ensemble des objets contenus dans la scène. Pour obtenir ce résultat, nous utilisons un graphe de régions non local. À partir d'un RAG calculé avec la partition de l'image initiale, nous ajoutons à chaque sommet ses 5 plus proches voisins (régions) parmi l'ensemble des régions dans la partition. Le graphe ainsi construit correspond à un $\text{RAG} \cup 5\text{-NNG}$. Cette expérimentation montre que l'utilisation d'une telle structure, permet d'obtenir des

schémas rapides de segmentation non locaux, c'est-à-dire l'extraction d'éléments non adjacents ou non connexes, avec un minimum de marqueurs initiaux.

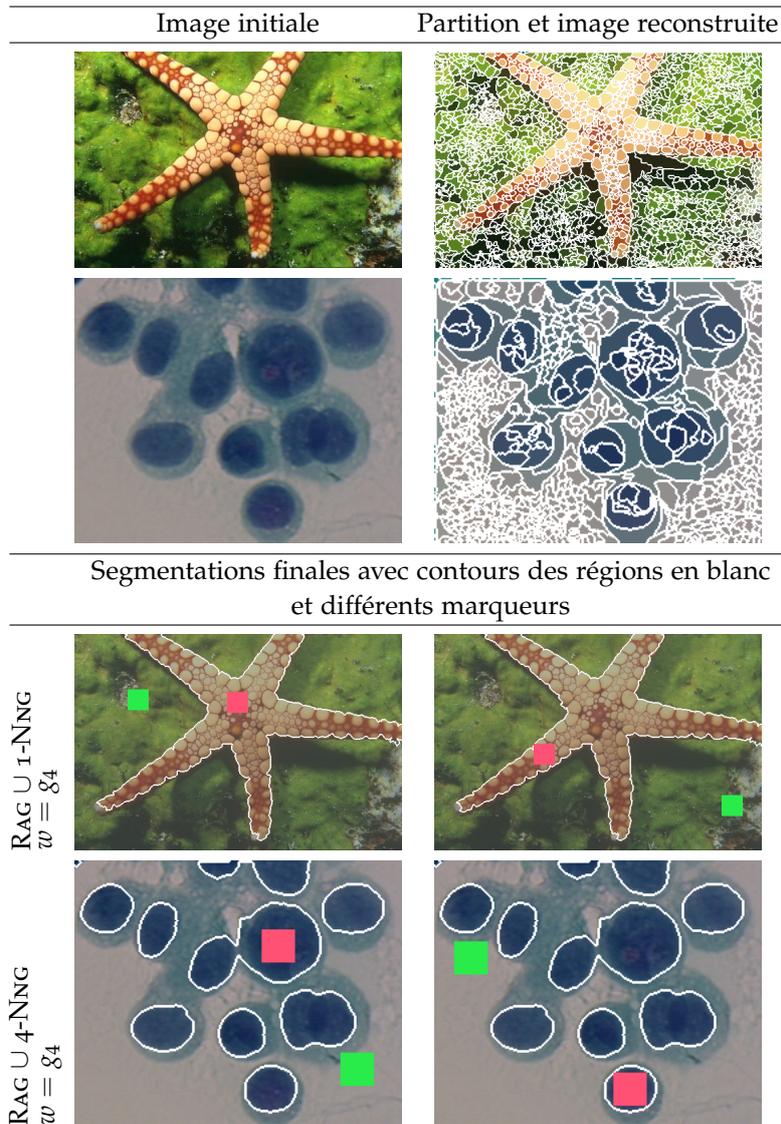


FIG. 5.8: Segmentation non locale d'image avec graphes basés régions. Dans la première partie, pour chaque ligne et de gauche à droite : l'image initiale et l'image reconstruite avec la couleur moyenne. Les régions de la partition sont également superposées en blanc. La première image contient 154 401 pixels et la seconde 49 000 pixels. Les partitions contiennent respectivement 2 391 régions et 741 régions (98% de réduction pour les deux cas). Dans la seconde partie, pour chaque ligne et de gauche à droite : les segmentations obtenues où les contours des régions segmentées sont superposés en blanc sur l'image initiale. Les marqueurs originaux sont également représentés, de couleur rouge pour les objets à segmenter et de couleur verte pour l'arrière-plan.

La figure 5.8 montre deux autres exemples de segmentation d'image avec des graphes de régions. Elle illustre également la robustesse de la notre approche vis-à-vis de la localisation des marqueurs initiaux. Les graphes utilisées dans la figure 5.8 sont des $RAG \cup k\text{-NNG}$ où $k = 1$ pour le premier exemple et $k = 4$ pour le second. Les voisins sont sélectionnés parmi toutes les régions des partitions des images initiales en utilisant la moyenne couleur pour calculer les distances. Ces graphes sont pondérés par la fonction de poids g_4 . Dans la seconde partie de la figure 5.8, nous présentons pour chaque exemple deux résultats de segmentation. Nous avons superposé en blanc, les contours des régions finales obtenues. Sont également symbolisés avec des carrés colorés les marqueurs initiaux ayant servi à obtenir les résultats présentés. Les carrés rouges représentent les objets à extraire et les carrés blancs l'arrière-plan des images. Comme pour la figure 5.7, nous pouvons constater que notre approche permet d'obtenir une segmentation satisfaisante de l'ensemble des objets avec un nombre minimal de marqueurs. Nous pouvons également apprécier la sensibilité et la robustesse de notre approche vis-à-vis du positionnement des marqueurs initiaux (peu de différences entre les résultats obtenus avec deux initialisations différentes).

Dans cette section, nous avons montré les potentialités de notre méthode basée sur l'équation eikonale et les graphes de régions pour réaliser de la segmentation d'images. Comme pour le cas de la segmentation semi supervisée (voir chapitre 3), l'utilisation de graphes de régions comporte plusieurs avantages : des schémas de segmentation rapides grâce à la réduction du nombre de sommets à traiter, une extraction des objets non connexes et un minimum de marqueurs initiaux nécessaires.

Classification de bases de données

La figure 5.9 montre une nouvelle (à notre connaissance) utilisation de l'équation eikonale pour la classification de données de grande dimension non organisées. Les données initiales (la première partie de la figure 5.9) proviennent de la même base que celle utilisée dans les figures 5.3 et 5.4 pour réaliser de l'ordonnancement : la base de l'USPS. Ces données initiales représentent des variétés d'images. La figure 5.9 présente deux exemples de 100 et 90 échantillons provenant de la base initiale.

Les graphes associés à ces deux exemples sont respectivement un 20-NNG pour le premier exemple et un 5-NNG pour le second pondérés par la fonction de poids g_3 . Chaque sommet des graphes est associé à une image et est décrit par un vecteur d'attributs de 256 dimensions ($\mathbb{R}^{16 \times 16}$) correspondant à la taille des images (16×16).

La seconde partie de la figure 5.9 montre les marqueurs utilisés ainsi que les résultats de la classification obtenus en accord avec les marqueurs. La classification est réalisée de manière similaire à la segmentation multi labels des images, c'est-à-dire que nous calculons une carte de distance par classe (problème à deux classes pour le premier exemple et à trois classes pour le second) et la classification finale est réalisée par l'utilisation de la fonction de décision définie par l'équation (5.33). Nous pouvons constater que nous obtenons des résultats de classification satisfaisants : le taux de bonne classification est de 100% dans le premier cas et de 93,1% dans le second. Il est également important de noter que ces taux de classification et ces résultats sont obtenus en utilisant une méthode simple pour calculer les si-

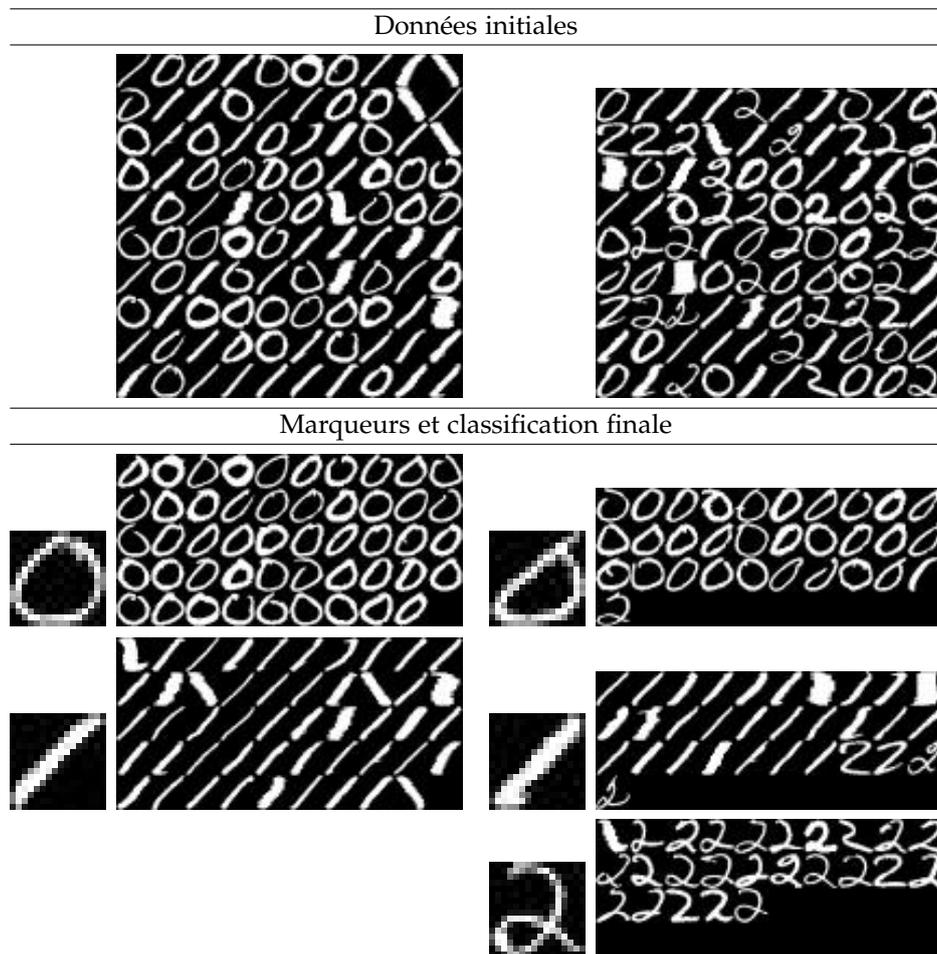


FIG. 5.9: Classification de variétés d'images avec l'équation eikonale. *Chaque colonne présente deux exemples différents. Dans la première partie : les données initiales définies comme des fonctions $f^0 : V \subset \mathbb{R}^m \rightarrow \mathbb{R}^{256}$. Dans la seconde partie, pour chaque exemple : les marqueurs (à gauche) et les résultats de la classification (à droite) en accord avec le marqueur initial.*

milarités entre les images et construire les graphes associés aux données (distance Euclidienne pixel à pixel). Comme nous l'avons déjà précisé, cette distance ne prend pas en compte les transformations géométriques et n'est pas robuste aux différences d'illumination des images.

Cette expérimentation montre les potentialités de notre méthodologie basée sur des graphes pour segmenter, à l'aide de l'équation eikonale, des données de grande dimension définies sur des domaines non uniformes. Nous montrons également une nouvelle application de l'équation eikonale qui peut être utile pour l'apprentissage ou la fouille de données : la classification de données arbitraires.

5.5 Conclusion

Dans ce chapitre, nous avons présenté une nouvelle formulation pour résoudre l'équation eikonale sur des graphes. Cette résolution est basée sur des gradients définis par des équations aux différences partielles. Ces derniers gradients interviennent également dans notre approche de la morphologie mathématique présentée dans le chapitre 4.

Notre approche permet d'utiliser l'équation eikonale sur tout type de données y compris celles définies sur des domaines non uniformes ou de grande dimension sans étape de discrétisation spatiale.

Notre approche pour résoudre équation eikonale constitue un cadre simple, adaptatif permettant de retrouver des approches connues de la littérature. De plus, notre formalisme unifie les traitements locaux et non locaux dans le cadre du traitement des images. L'intégration des schémas non locaux basés sur des patches dans le calcul des distances et la segmentation permet d'obtenir des résultats supérieurs aux schémas locaux, notamment pour des images comportant des structures fines et répétitives. Nous avons également montré que l'utilisation de graphes basés sur les régions permet d'une part, de réduire la complexité des algorithmes rendant la segmentation des images plus rapide et d'autre part, d'extraire des objets non adjacents avec un minimum de marqueurs initiaux.

Finalement, nous avons utilisé l'équation eikonale pour traiter des données arbitraires telles que les bases de données et pour calculer (de manière simple) des distances géodésiques dans des variétés. Ce type de traitement peut être utile et être appliqué pour réaliser de l'indexation par le contenu ou encore de la classification dans des bases de données. Ce dernier point permet d'ouvrir de nouvelles perspectives intéressantes pour appliquer l'équation eikonale dans des domaines tels que ceux de l'apprentissage ou de la fouille de données.

Ce travail initial sur l'équation eikonale ouvre de nombreuses pistes. Nous pouvons, par exemple, envisager de résoudre par la même formalisme la version stationnaire de l'équation eikonale. Une autre piste serait d'utiliser notre formalisme pour résoudre des applications liées aux problèmes des chemins minimaux dans des domaines tels que la robotique ou la planification de chemins. Finalement, dans le domaine de l'apprentissage ou de la fouille de données, nous pouvons également envisager d'utiliser notre méthodologie pour proposer de nouveaux algorithmes de classification : un k moyennes ou un k plus proches voisins géodésiques.

6

Conclusion générale

Dans cette conclusion générale, nous présentons une synthèse des principales contributions de ce manuscrit ainsi que les perspectives de nos travaux.

Synthèse, contributions et perspectives

L'idée maîtresse qui a guidé les travaux présentés dans ce manuscrit est la transcription de certains modèles issus du traitement des images définis dans le domaine continu vers des schémas discrets en exploitant le formalisme des équations aux différences partielles définies sur des graphes arbitraires. Cette adaptation permet d'étendre ces modèles pour d'une part, le traitement de données de grande dimension définies sur des domaines irréguliers (sans étape de discrétisation spatiale), et d'autre part, pour le traitement des images, d'unifier dans une même formulation les schémas locaux et non locaux. Ces transcriptions permettent d'envisager de nouvelles applications de ces modèles dans des domaines tels que l'apprentissage ou la fouille de données.

Nous nous sommes intéressés à certains modèles variationnels et basés sur les équations aux dérivées partielles connues de la littérature : les modèles variationnels basés sur la régularisation, les équations aux dérivées partielles définissant la morphologie mathématique et l'équation eikonale. Dans le contexte du traitement des images, ces approches ont été utilisées avec succès dans des applications telles que la segmentation, le filtrage, la restauration ou encore la simplification.

Opérateurs différentiels discrets

Dans le chapitre 2, nous avons proposé de nouvelles familles d'opérateurs de premier et second ordre définis sur des graphes tels que les différences et gradients directionnels et une classe de Laplaciens telle que le Laplacien anisotrope. Ces opérateurs différentiels sont les constituants principaux qui nous ont permis de transcrire et d'adapter les modèles précédemment cités vers des schémas discrets définis sur des graphes en utilisant le formalisme des équations aux différences partielles.

Finalement, à partir des opérateurs différentiels discrets définis dans le chapitre 2, nous pouvons envisager de reformuler d'autres opérateurs définis dans le domaine continu, par exemple le Laplacien infini.

Fonctionnelles de régularisation

Dans le chapitre 3, nous avons abordé des modèles variationnels basés sur la régularisation pour le filtrage et la simplification mais également pour la segmentation semi et non supervisée.

Modèles variationnels Dans un premier temps, nous avons défini deux familles de fonctionnelles discrètes de régularisation basées sur des graphes à partir desquelles nous avons défini deux classes de modèles variationnels discrets.

À partir de la première classe de modèles variationnels, nous nous sommes intéressés à deux cas particuliers : les modèles de régularisation isotrope et anisotrope. Nous avons proposé des algorithmes discrets permettant d'approximer les solutions de ces modèles et conduisant à des familles de filtres linéaires et non linéaires. Nous avons présenté l'intérêt de ces filtres pour la simplification et le filtrage de données discrètes (des bases de données, des variétés d'images ou des nuages de points). Nous avons montré que ces traitements pouvaient améliorer les résultats des méthodes de classification et de regroupement de données. Nous avons également montré que le débruitage de variétés à partir de nos modèles de régularisation se comportaient comme une réduction de dimension dans le domaine spatial.

Interpolation Dans un second temps, nous avons utilisé nos modèles de régularisation pour formuler des problèmes d'interpolation tels que la colorisation des images monochromes et la segmentation et classification semi supervisées.

Dans le contexte du traitement des images, nous avons introduit les configurations non locales basées sur des patchs pour la colorisation et la segmentation. L'utilisation des schémas non locaux est peu étudiée pour ces domaines d'application alors qu'ils ont montré leur supériorité par rapport aux schémas locaux dans la prise en compte des éléments fins et répétitifs contenus dans les images.

Nous avons également illustré les bénéfices de l'utilisation des graphes de régions pour la segmentation des images : la réduction de la complexité des algorithmes ou la segmentation semi supervisée d'objets non connexes ou adjacents.

Nous avons également présenté l'application de ces méthodes pour le traitement des images cellulaires issues de microscopie conventionnelle ou de dispositifs d'acquisition de lames entières.

À partir de nos modèles, une perspective intéressante serait d'adapter, avec les équations aux différences partielles définies sur des graphes, d'autres problèmes d'interpolation tels que la retouche d'images ou encore la super résolution et d'intégrer les schémas non locaux dans la résolution de ces problématiques.

Une autre perspective serait d'ouvrir de nouvelles pistes dans le domaine de l'apprentissage telles que l'interpolation des valeurs manquantes ou encore le traitement multi échelles des bases de données.

Finalement, les schémas de segmentation que nous avons proposés utilisent des graphes et permettent d'unifier les traitements pour tout type de données. Une perspective intéressante, serait d'utiliser ces mêmes outils pour analyser, catégoriser et classer les objets dans des grandes bases de données.

Optimisation globale Finalement, dans le chapitre 3, nous nous sommes intéressés à une seconde classe de modèles variationnels discrets basés sur la régularisation. Cette classe de modèles correspond à une classe de problèmes d'optimisation globale de fonctionnelles de régularisation non convexes. Notre approche de l'optimisation globale peut être considérée comme la version discrète des approches définies dans le domaine continu, mais également comme une extension sur des graphes de ces modèles.

Pour trouver un minimum global à ces problèmes nous avons utilisé les propriétés de la formule de la co-aire définie sur des graphes et une méthode discrète de relaxation afin de les rendre convexes. À partir de la formulation du p -Laplacien anisotrope, nous avons proposé un algorithme permettant d'approximer la solution de ces problèmes d'optimisation globale.

Notre formulation générale nous a permis de transcrire des modèles continus (issus du traitement des images) vers des schémas discrets tels que le modèle de ROF ou celui de Cv. À partir de ces modèles, nous avons montré les potentialités de notre méthodologie pour réaliser de la segmentation non locale des images ou encore du partitionnement de bases de données.

Une perspective intéressante, dans le domaine de l'apprentissage ou celui de la fouille de données, serait de proposer de nouveaux algorithmes de classification. En effet, nous pouvons constater que le modèle discret de Cv que nous avons proposé correspond à l'algorithme des k moyennes dans sa version régularisée.

Morphologie mathématique

Dans le chapitre 4, nous avons présenté une nouvelle approche de la morphologie mathématique basée sur des équations aux différences partielles définies sur des graphes. Notre formulation de la morphologie mathématique étend les approches basées sur les équations aux dérivées partielles vers des schémas discrets locaux et non locaux et les généralise pour le traitement de tout type de données discrètes.

Notre approche unifie les schémas locaux et non locaux et nous permet d'introduire une nouvelle classe d'opérateurs morphologiques pour le traitement des images en considérant les configurations non locales. Nous avons montré que ces nouveaux opérateurs permettent de mieux capturer et préserver les structures fines et répétitives dans les images.

Notre approche de la morphologie mathématique constitue un cadre simple et unificateur permettant de retrouver des schémas numériques employés dans les approches définies par les équations aux dérivées partielles, mais également en morphologie mathématique algébrique définie sur des graphes.

Notre formalisme général et simple permet de revisiter d'autres opérateurs ou formulations de la morphologie mathématique que ceux que nous avons présentés dans ce chapitre tels que la morphologie mathématique visqueuse ou la morphologie fonctionnelle.

Une autre perspective serait l'extension de ces travaux vers une nouvelle classe d'opérateurs morphologiques algébriques intégrant la notion de non localité, à travers la fonction de poids, pour le traitement des images. Cette notion de pondération des arêtes du graphe permet également d'envisager une formulation simple pour la généralisation de la notion d'élément structurant rendant les opérations de

morphologie mathématique naturellement adaptatives.

Finalement, une des perspectives de ce travail sur la morphologie mathématique est l'application d'opérateurs morphologiques dans des communautés telles que l'apprentissage ou la fouille de données.

Équation eikonale

Dans le chapitre 5, nous avons exploité les résultats obtenus dans le chapitre 4 pour transcrire sur des graphes l'équation eikonale et proposer une nouvelle formulation de la résolution de cette équation.

Notre approche permet d'utiliser l'équation eikonale sur tout type de données y compris celles définies sur des domaines non uniformes ou de grande dimension sans étape de discrétisation spatiale.

Notre approche pour résoudre l'équation eikonale constitue un cadre simple et adaptatif permettant de retrouver des approches connues de la littérature. De plus, notre formalisme unifie les traitements locaux et non locaux dans le cadre du traitement des images.

Nous avons utilisé l'équation eikonale pour traiter des données arbitraires telles que les bases de données et pour calculer (de manière simple) des distances géodésiques dans des variétés. Ce type de traitement peut être utile et être appliqué pour réaliser de l'indexation par le contenu ou encore de la classification dans des bases de données.

Ce travail initial sur l'équation eikonale ouvre de nombreuses pistes. Nous pouvons, par exemple, envisager de résoudre dans le même formalisme la version stationnaire de l'équation eikonale. Une autre piste serait d'utiliser notre formalisme pour résoudre des applications liées aux problèmes des chemins minimaux dans des domaines tels que la robotique ou la planification de chemins. Finalement, dans le domaine de l'apprentissage ou de la fouille de données, nous pouvons également envisager d'utiliser notre méthodologie pour proposer de nouveaux algorithmes de classification : k moyennes ou k plus proches voisins géodésiques.

Perspectives générales

Nous présentons ici des perspectives moins spécifiques à chaque chapitre, mais plus générales sur l'ensemble des travaux de ce manuscrit.

Tout d'abord, nous pouvons envisager un étude théorique pour montrer les liens entre les opérateurs tels que le p -Laplacien et les opérateurs morphologiques que nous avons introduits, permettant de résoudre par notre formalisme de la morphologie mathématique certains traitements utilisant ces opérateurs.

Nous pouvons également considérer des problèmes de décomposition spectrale d'un point de vue variationnel et utiliser notre formalisme basé sur les graphes afin de proposer des algorithmes simples de résolution.

Dans le domaine du traitements des images, nous avons vu l'efficacité des schémas non locaux. Malheureusement, ces méthodes ont une grande complexité. Une perspective possible serait de réduire cette complexité en utilisant des méthodes issues de l'apprentissage telles que les représentations parcimonieuses ou des réductions de dimensions afin de sélectionner uniquement les voisins pertinents. Ces

méthodes utilisent des patchs habituellement carrés. Une des limitations est leur sensibilité aux transformations géométriques. Il serait possible d'envisager d'utiliser une autre représentation, une transformation de l'information contenue dans les patchs afin de palier ces limitations. D'une manière plus générale, nous pouvons également envisager l'étude de la définition de patchs sur des graphes arbitraires et de comparer (comme pour le cas du traitement des images) ces méthodes aux méthodes usuelles dans le cadre du traitement de données.

Nous avons évoqué les problèmes d'interpolation et de l'extension possible de ces méthodes (retouche, super résolution) sur des graphes. Cette extension pose des problèmes intéressants à étudier, à savoir comment interpoler une information sur un domaine non uniforme et plus généralement sur un graphe arbitraire.

L'avantage de notre approche est de pouvoir appliquer les méthodes que nous avons définies sur des graphes à d'autres domaines que celui du traitement des images et ainsi d'ouvrir d'autres applications pour ces modèles. Par exemple, nous pouvons envisager d'appliquer l'ensemble de nos modèles et algorithmes pour le traitement de données dans le domaine de l'apprentissage, de la classification ou de la fouille de données.

Finalement, les travaux présentés dans ce manuscrit ouvrent de nombreuses pistes et permettent de fournir un cadre général et simple qui unifie les modèles définis dans le domaine continu et ceux définis dans le domaine discret.

Liste des publications

Dans cette partie, le lecteur trouvera la liste des publications réalisées à partir des travaux de thèse présentés dans ce manuscrit.

Chapitre de livre

- [1] Vinh-Thong TA, Olivier LÉZORAY et Abderrahim ELMOATAZ : *Biomedical Microscopic Image Processing by Graphs*. Information Science Reference, 2009. À paraître.

Revue internationale avec comités de lecture

- [1] Vinh-Thong TA, Olivier LÉZORAY, Abderrahim ELMOATAZ et Sophie SCHÜPP : Graph-based tools for microscopic cellular image segmentation. *Pattern Recognition*, 42(6):1113–1125, 2009.

Conférences internationales avec comités de lecture et actes

- [1] Olivier LÉZORAY, Vinh-Thong TA et Abderrahim ELMOATAZ : Manifold and data filtering on graphs. *Dans Proceedings of International Symposium on Methodologies for Intelligent Systems, International Workshop on Topological learning*, pages 19 – 28, 2009.
- [2] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Nonlocal graph morphology. *Dans Proceedings of 9th International Symposium on Mathematical Morphology (ISMM), Abstract Book*, pages 5 – 9, 2009.
- [3] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Adaptation of eikonal equation over weighted graphs. *Dans Proceedings of 2nd International Conference on Scale Space and Variational Methods in Computer Vision (SSVM), LNCS*, pages 187 – 199. Springer, 2009.
- [4] Vinh-Thong TA, Olivier LÉZORAY et Abderrahim ELMOATAZ : Nonlocal morphological levelings by partial difference equations over weighted graphs. *Dans Proceedings of 19th International Conference on Pattern Recognition (ICPR)*, 2008.
- [5] Olivier LÉZORAY, Vinh-Thong TA et Abderrahim ELMOATAZ : Nonlocal graph regularization for image colorization. *Dans Proceedings of 19th International Conference on Pattern Recognition (ICPR)*, 2008.
- [6] Olivier LÉZORAY, Vinh-Thong TA et Abderrahim ELMOATAZ : Impulse noise removal by spectral clustering and regularization on graphs. *Dans Proceedings of 19th International Conference on Pattern Recognition (ICPR)*, 2008.
- [7] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Partial difference equations over graphs : morphological processing of arbitrary discrete data. *Dans Proceedings of 10th European Conference on Computer Vision (ECCV), LNCS 5304*, pages 668 – 680. Springer, 2008.

- [8] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Partial difference equations on graphs for mathematical morphology operators over images and manifolds. *Dans Proceedings of 15th IEEE International Conference on Image Processing (ICIP)*, pages 801 – 804, 2008. **Prix Ibm du meilleur article étudiant.**
- [9] Abderrahim ELMOATAZ, Olivier LÉZORAY, Sébastien BOUGLEUX et Vinh-Thong TA : Unifying local and nonlocal processing with partial difference operators on weighted graphs. *Dans Proceedings of the International Workshop on Local and Non-Local Approximation in Image Processing (LNLA)*, pages 11 – 26, 2008. **Article invité.**
- [10] Olivier LÉZORAY, Abderrahim ELMOATAZ et Vinh-Thong TA : Learning graph neighborhood topological order for image and manifold morphological processing. *Dans Proceedings of 8th IEEE International Conference on Computer and Information Technology (CIT)*, pages 350 – 355, 2008.
- [11] Vinh-Thong TA, Olivier LÉZORAY et Abderrahim ELMOATAZ : Graph based semi and unsupervised classification and segmentation of microscopic images. *Dans Proceedings of 7th International Symposium on Signal Processing and Information Technology (ISSPIT)*, pages 1160–1165, 2007.
- [12] Vinh-Thong TA, Olivier LÉZORAY et Abderrahim ELMOATAZ : Graphs regularization for data sets and images : Filtering and semi-supervised classification. *Dans PASCAL Workshop : Learning from and with graphs, 6th IAPR-TC-15 Workshop on Graph based Representations in Pattern Recognition*, 2007.

Conférences nationales avec comités de lecture et actes

- [1] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Morphologie mathématique sur graphes pondérés de topologie arbitraire — traitements non locaux d’images et de données multi-variées. *Dans Actes du groupe de recherche et d’études du traitement du signal et des images (GRETSI)*, 2009.

Journées nationales sans comités de lecture ni actes

- [1] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Morphologie mathématique sur graphes pondérés de topologie arbitraire. *Dans 32ième journée de l’International Society for Stereology – France (Iss)*, école des mines, Paris, 2009.

Rapports techniques et articles soumis

- [1] Vinh-Thong TA, Sébastien BOUGLEUX, Abderrahim ELMOATAZ et Olivier LÉZORAY : Nonlocal anisotropic discrete regularization for image, data filtering and clustering. Rapport technique, GREYC CNRS UMR 6072 - Université de Caen Basse-Normandie – ENSICAEN - hal-00187165, nov 2007.
- [2] Vinh-Thong TA, Abderrahim ELMOATAZ et Olivier LÉZORAY : Mathematical morphology and eikonal equations on graphs for nonlocal image and data processing. Rapport technique, GREYC CNRS UMR 6072 - Université de Caen Basse-Normandie – ENSICAEN - hal-00365431, mars 2009. Soumis à IEEE Transactions on Image Processing.

Bibliographie

- [Abgrall, 1996] ABGRALL, R. (1996). Numerical discretization of the first-order hamilton-jacobi equations on triangular meshes. *Comm. Pure and Applied Math.*, 49 : 1339–1373.
- [Alvarez et al., 1993] ALVAREZ, L., GUICHARD, F., LIONS, P.-L. et MOREL, J.-M. (1993). Axioms and fundamental equations of image processing. *Archive of Rational Mechanics and Analysis*, 123(3) : 199 – 257.
- [Angulo, 2005] ANGULO, J. (2005). Unified morphological color processing framework in lum/sat/hue representation. In *Proceedings of International Symposium on Mathematical Morphology (ISMM)*, pages 387 – 396.
- [Angulo, 2009] ANGULO, J. (2009). From scalar-valued images to hypercomplex representations and derived total orderings for morphological operators. In *In the Proceedings of International Symposium on Mathematical Morphology (ISMM)*, LNCS 5720, pages 238 – 249.
- [Aptoula et Lefèvre, 2007] APTOULA, E. et LEFÈVRE, S. (2007). A comparative study on multivariate mathematical morphology. *Pattern Recognition*, 40(11) : 2914 – 2929.
- [Arbeláez et Cohen, 2004] ARBELÁEZ, P. A. et COHEN, L. (2004). Energy partitions and image segmentation. *Journal of Mathematical Imaging and Vision*, 20 : 43–57.
- [Arehart et al., 1993] AREHART, A. B., VINCENT, L. et KIMIA, B. B. (1993). Mathematical morphology : The hamilton jacobi connection. In *Proceedings of International Conference on Computer Vision ICCV*, pages 215–219.
- [Asuncion et Newman, 2007] ASUNCION, A. et NEWMAN, D. J. (2007). UCI machine learning repository. <http://archive.ics.uci.edu/ml/>, University of California, Irvine, School of Information and Computer Sciences.
- [Aubert et Kornprobst, 2006] AUBERT, G. et KORNPBST, P. (2006). *Mathematical Problems in Image Processing : Partial Differential Equations and the Calculus of Variations*, volume 147 de *Applied Mathematical Sciences*. Springer, deuxième édition.
- [Bakushinsky et Goncharsky, 1994] BAKUSHINSKY, A. B. et GONCHARSKY, A. (1994). *Ill-Posed Problems : Theory and Applications*. Kluwer Academic Publishers.
- [Barash, 2002] BARASH, D. (2002). A fundamental relationship between bilateral filtering, adaptive smoothing, and the nonlinear diffusion equation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(6) : 844 – 847.
- [Belkin et Niyogi, 2003] BELKIN, M. et NIYOGI, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6) : 1373 – 1396.
- [Belkin et al., 2006] BELKIN, M., NIYOGI, P. et SINDHWANI, V. (2006). Manifold regularization : A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7 : 2399 – 2434.
- [Benmansour et Cohen, 2009] BENMANSOUR, F. et COHEN, L. (2009). Fast object segmentation by growing minimal paths from a single point on 2d or 3d images. *Journal of Mathematical Imaging and Vision*.

- [Bensoussan et Menaldi, 2003] BENSOUSSAN, A. et MENALDI, J.-L. (2003). Difference equations on weighted graphs. *Journal of Convex Analysis*, 12(1) : 13 – 44.
- [Bougleux, 2007] BOUGLEUX, S. (2007). *Reconstruction, Détection et Régularisation de Données Discrètes*. Thèse de doctorat, Université de Caen-Basse Normandie.
- [Bougleux et Elmoataz, 2005] BOUGLEUX, S. et ELMOATAZ, A. (2005). Image smoothing and segmentation by graph regularization. In *Proceedings of International Symposium on Visual Computing (ISCV)*, LNCS 3804, pages 745–752. Springer.
- [Bougleux et al., 2007] BOUGLEUX, S., ELMOATAZ, A. et MELKEMI, M. (2007). Discrete regularization on weighted graphs for image and mesh filtering. In *Proceedings of International Conference on Scale Space and Variational Methods in Computer Vision (SSVM)*, LNCS 4485, pages 128–139. Springer.
- [Bougleux et al., 2009] BOUGLEUX, S., ELMOATAZ, A. et MELKEMI, M. (2009). Local and nonlocal discrete regularization on weighted graphs for image and mesh processing. *International Journal of Computer Vision*, 83(2) : 220 – 236. Special Issue of SSVM 2007.
- [Boykov et Jolly, 2001] BOYKOV, Y. et JOLLY, M.-P. (2001). Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. In *Proceedings of International Conference on Computer Vision (ICCV)*, pages 105 – 112.
- [Boykov et Kolmogorov, 2003] BOYKOV, Y. et KOLMOGOROV, V. (2003). Computing geodesics and minimal surfaces via graph cuts. In *Proceedings of International Conference on Computer Vision ICCV*, pages 26 – 33.
- [Boykov et Kolmogorov, 2004] BOYKOV, Y. et KOLMOGOROV, V. (2004). An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9) : 1124 – 1137.
- [Boykov et al., 2001] BOYKOV, Y., VEKSLER, O. et ZABIH, R. (2001). Fast approximation energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11) : 1222 – 1239.
- [Bresson et Chan, 2008] BRESSON, X. et CHAN, T. F. (2008). Non-local unsupervised variational image segmentation models. CAM Report 08-67, University of California Los Angeles.
- [Breuß et al., 2007] BREUSS, M., BURGETH, B. et WEICKERT, J. (2007). Anisotropic continuous-scale morphology. In *Proceedings of Iberian Conference on Pattern Recognition and Image Analysis IBPRIA*, LNCS 4478, pages 515 – 522. Springer.
- [Breuß et Weickert, 2006] BREUSS, M. et WEICKERT, J. (2006). A shock-capturing algorithm for the differential equations of dilation and erosion. *Journal of Mathematical Imaging and Vision*, 25 : 187 – 201.
- [Brockett et Maragos, 1992] BROCKETT, R. W. et MARAGOS, P. (1992). Evolution equations for continuous-scale morphology. In *Proceedings of ICASSP 1992*, volume 3, pages 125–128.
- [Brockett et Maragos, 1994] BROCKETT, R. W. et MARAGOS, P. (1994). Evolution equations for continuous-scale morphological filtering. *IEEE Transactions on Signal Processing*, 42(12) : 3377 – 3386.

- [Bronstein *et al.*, 2007] BRONSTEIN, A. M., BRONSTEIN, M. M. et KIMMEL, R. (2007). Weighted distance maps computation on parametric three-dimensional manifolds. *J. Comput. Phys.*, 225(1).
- [Brox *et al.*, 2008] BROX, T., KLEINSCHMIDT, O. et CREMERS, D. (2008). Efficient non-local means for denoising of textural patterns. *IEEE Transactions on Image Processing*, 17(17) : 1083 – 1092.
- [Buades *et al.*, 2008] BUADES, A., COLL, B. et MOREL, J.-M. (2008). Nonlocal image and movie denoising. *International Journal of Computer Vision*, 76(2) : 123 – 139.
- [Bühler et Hein, 2009] BÜHLER, T. et HEIN, M. (2009). Spectral clustering based on the graph p -laplacian. In *In Proceedings of International Conference on Machine Learning (ICML)*, pages 81 – 88.
- [Burger et Hintermüller, 2005] BURGER, M. et HINTERMÜLLER, M. (2005). Projected gradient flows for bv /level set relaxation. *Proceedings of Applied Mathematics and Mechanics*, 5 : 11 – 14.
- [Burgeth *et al.*, 2007] BURGETH, B., BRUHN, A., DIDAS, S., WEICKERT, J. et WELK, M. (2007). Morphology for matrix data : Ordering versus pde-based approach. *Journal of Image Vision Computing*, 25 : 496 – 511.
- [Carlini *et al.*, 2008] CARLINI, E., FALCONE, M., FORCADEL, N. et MONNEAU, R. (2008). Convergence of a generalized fast marching method for an eikonal equation with a velocity changing sign. *SIAM Journal on Numerical Analysis*, 46(6) : 2920 – 2952.
- [Chambolle, 2004] CHAMBOLLE, A. (2004). An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20(1 – 2) : 89 – 97.
- [Chambolle, 2005] CHAMBOLLE, A. (2005). Total variation minimization and a class of binary MRF models. In *in Proceedings of Energy Minimization Methods in in Computer Vision and Pattern Recognition EMMCVPR, LNCS 3757*, pages 136 – 152.
- [Chan et Esedoglu, 2005] CHAN, T. F. et ESEDOGLU, S. (2005). Aspects of total variation regularized l^1 function approximation. *SIAM Journal on Applied Mathematics*, 65(5) : 1817 – 1837.
- [Chan *et al.*, 2006] CHAN, T. F., ESEDOGLU, S. et NIKOLOVA, M. (2006). Algorithms for finding global minimizers of image segmentation and denosing models. *SIAM Journal on Applied Mathematics*, 66(5) : 1632 – 1648.
- [Chan *et al.*, 2001] CHAN, T. F., OSHER, S. et SHEN, J. (2001). The digital tv filter and nonlinear denoising. *IEEE Transactions on Image Processing*, 10(2) : 231 – 241.
- [Chan et Shen, 2005] CHAN, T. F. et SHEN, J. (2005). *Image processing and analysis : variational, PDE, wavelet, and stochastic methods*. SIAM.
- [Chan et Vese, 2001] CHAN, T. F. et VESE, L. A. (2001). Active contours without edges. *IEEE Transactions on Image Processing*, 10(2) : 266 – 277.
- [Chapelle *et al.*, 2006] CHAPELLE, O., SCHÖLKOPF, B. et ZIEN, A. (2006). *Semi-Supervised Learning*. MIT Press.
- [Chung, 1997] CHUNG, F. R. K. (1997). *Spectral Graph Theory*. CBMS Regional Conference Series in Mathematics.

- [Cohen et Kimmel, 1997] COHEN, L. et KIMMEL, R. (1997). Global minimum for active contour models : A minimal path approach. *International Journal of Computer Vision*, 24(1) : 57–78.
- [Coifman *et al.*, 2005] COIFMAN, R. D., LAFON, S., LEE, A. B., MAGGIONI, M., NADLER, B., WARNER, F. et ZUCKER, S. W. (2005). Geometric diffusions as a tool for harmonic analysis and and structure definition of data. *Proceedings of the National Academy of Sciences*, 102(2) : 7426 – 7432.
- [Couprie *et al.*, 2009] COUPRIE, C., GRADY, L., NAJMAN, L. et TALBOT, H. (2009). Power watersheds : A new image segmentation framework extending graph cuts, random walker and optimal spanning forest. *In In Proceedings of International Conference on Computer Vision (ICCV)*.
- [Cousty *et al.*, 2009a] COUSTY, J., BERTRAND, G., NAJMAN, L. et COUPRIE, M. (2009a). Watershed cuts : Minimum spanning forests and the drop of water principle. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(8) : 1362 – 1374.
- [Cousty *et al.*, 2009b] COUSTY, J., NAJMAN, L. et SERRA, J. (2009b). Some morphological operators in graph spaces. *In In the Proceedings of International Symposium on Mathematical Morphology (ISMM)*, LNCS 5720, pages 149 – 160.
- [Crandall, 1995] CRANDALL, M. G. (1995). *Viscosity solutions : a primer*, chapitre 1, pages 1 – 43. Springer.
- [Crandall *et al.*, 1992] CRANDALL, M. G., ISHII, H. et LIONS, P.-L. (1992). User’s guide to viscosity solutions of second order partial differential equations. *Bulletin of the American Mathematical Society*, 27 : 1 – 67.
- [Crandall et Lions, 1983] CRANDALL, M. G. et LIONS, P.-L. (1983). Viscosity solutions of hamilton-jacobi equations. *Trans. Amer. Math. Soc.*, 277 : 1–42.
- [Danielsson, 1980] DANIELSSON, P.-E. (1980). Euclidean distance mapping. *Computer Graphics and Image Processing*, 14 : 227–248.
- [Darbon, 2007] DARBON, J. (2007). A note on the discrete binary mumford-shah model. *In in Proceedings of Computer Vision/Computer Graphics Collaboration Techniques (MIRAGE)*, pages 283 – 294.
- [Darbon *et al.*, 2008] DARBON, J., CUNHA, A., CHAN, T. F., OSHER, S. et JENSEN, G. J. (2008). Fast nonlocal filtering applied to electron cryomicroscopy. *In In the Proceedings of the IEEE International Symposium on Biomedical Imaging (ISBI)*, pages 1331 – 1334.
- [Darbon et Sigelle, 2005] DARBON, J. et SIGELLE, M. (2005). Exact optimization of discrete constrained total variation minimization problems. *In In Proceedings of International Workshop on Combinatorial Image Analysis (IWCIA)*, LNCS 3322, pages 548 – 557.
- [Deschamps et Cohen, 2001] DESCHAMPS, T. et COHEN, L. (2001). Fast extraction of minimal paths in 3d images and application to virtual endoscopy. *Medical Image Analysis*, 5(4).
- [Diestel, 2005] DIESTEL, R. (2005). *Graph Theory*, volume 173 de *Graduate Texts in Mathematics*. Springer-Verlag Heidelberg, third édition.

- [Dumont *et al.*, 2007] DUMONT, M., MARÉE, R., GEURTS, P. et WEHENKEL, L. (2007). Random subwindows and multiple output decision trees for generic image annotation. In *Proceedings of 6th Annual Machine Learning Conference of Belgium and The Netherlands*.
- [Efros et Leung, 1999] EFROS, A. A. et LEUNG, T. K. (1999). Texture synthesis by non-parametric sampling. In *Proceedings of International Conference on Computer Vision (Iccv)*, volume 2, pages 1033 – 1038.
- [Elmoataz *et al.*, 2008a] ELMOATAZ, A., LÉZORAY, O. et BOUGLEUX, S. (2008a). Non-local discrete regularization on weighted graphs : A framework for image and manifold processing. *IEEE Transactions on Image Processing*, 17(7) : 1047 – 1060.
- [Elmoataz *et al.*, 2008b] ELMOATAZ, A., LÉZORAY, O., BOUGLEUX, S. et TA, V.-T. (2008b). Unifying local and nonlocal processing with partial difference operators on weighted graphs. In *Proceedings of Local and Non-Local Approximation in Image Processing (LNLA)*, pages 11–26.
- [Falcão *et al.*, 2004] FALCÃO, A. X., STOLFI, J. et LOTUFO, A. (2004). The image foresting transform : Theory, algorithms, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(1) : 19 – 29.
- [Falcone et Ferretti, 2002] FALCONE, M. et FERRETTI, R. (2002). Semi-lagrangian schemes for hamilton-jacobi equations, discrete representation formulae and godunov methods. *Journal of Computational Physics*, 175(2) : 559 – 575.
- [Fleming et Rishel, 1960] FLEMING, W. H. et RISHEL, R. (1960). An integral formula for total gradient variation. *Archiv der Mathematik*, 11(1) : 218 – 222.
- [Gilboa et Osher, 2007a] GILBOA, G. et OSHER, S. (2007a). Nonlocal linear image regularization and supervised segmentation. *SIAM Multiscale Modeling and Simulation*, 6(2) : 595 – 630.
- [Gilboa et Osher, 2007b] GILBOA, G. et OSHER, S. (2007b). Nonlocal operators with applications to image processing. Rapport technique 07-23, University of California Los Angeles.
- [Grady, 2006] GRADY, L. (2006). Random walks for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11).
- [Grady et Alvino, 2009] GRADY, L. et ALVINO, C. (2009). The piecewise smooth mumford-shah functional on an arbitrary graph. *IEEE Transactions on Image Processing*, 18(11) : 2547 – 2561.
- [Grady et Jolly, 2008] GRADY, L. et JOLLY, M.-P. (2008). Weights and topology : A study of the effects of graph construction on 3D image segmentation. In *Proceedings of Medical Image Computing and Computer Assisted Intervention MICCAI*, pages 153 – 161.
- [Guichard *et al.*, 2007] GUICHARD, F., MARAGOS, P. et MOREL, J.-M. (2007). *Partial Differential Equations for Morphological Operators*, volume 183 de LNCS, pages 369–390. Springer.
- [Heijmans, 1994] HEIJMANS, H. J. A. M. (1994). *Morphological Image Operators*. Boston : Academic Press.

- [Heijmans *et al.*, 1992] HEIJMANS, H. J. A. M., NACKEN, P., TOET, A. et VINCENT, L. (1992). Graph morphology. *Journal of Visual Communication and Image Representation*, 3(1) : 24 – 38.
- [Heijmans et Vincent, 1992] HEIJMANS, H. J. A. M. et VINCENT, L. (1992). *Graph Morphology in Image Analysis*, chapitre 6, pages 171 – 203. E. Dougherty.
- [Hein *et al.*, 2007] HEIN, M., AUDIBERT, J.-Y. et von LUXBURG, U. (2007). Graph laplacians and their convergence on random neighborhood graphs. *Journal of Machine Learning Research*, 8 : 1325 – 1368.
- [Hein et Maier, 2007a] HEIN, M. et MAIER, M. (2007a). Manifold denoising. In *Advances in Neural Information Processing Systems*, pages 561 – 568.
- [Hein et Maier, 2007b] HEIN, M. et MAIER, M. (2007b). Manifold denoising as pre-processing for finding natural representations of data. In *In Proceedings of Conference on Artificial Intelligence*, pages 1646 – 1649.
- [Hein *et al.*, 2008] HEIN, M., von LUXBURG, U. et MAIER, M. (2008). Influence of graph construction on graph-based clustering measures. In *Advances in Neural Information Processing Systems*, pages 1025 – 1032.
- [Houhou *et al.*, 2009] HOUHOU, N., BRESSON, X., SZLAM, A., CHAN, T. F. et THIRAN, J.-P. (2009). Semi-supervised segmentation based on non-local continuous min-cut. In *Proceedings of 2nd International Conference on Scale Space and Variational Methods in Computer Vision (SsvM)*, LNCS 5567, pages 112–123.
- [Jaromczyk et Toussaint, 1992] JAROMCZYK, J. W. et TOUSSAINT, G. T. (1992). Relative neighborhood graphs and their relatives. *Proceedings of the IEEE*, 80(9).
- [Jeong et Whitaker, 2008] JEONG, W.-K. et WHITAKER, R. T. (2008). A fast iterative method for eikonal equations. *SIAM Journal on Scientific Computing*, 30(5) : 2512–2534.
- [Jolion et Rosenfeld, 1994] JOLION, J.-M. et ROSENFELD, A. (1994). *A Pyramid Framework for Early Vision*. Kluwer.
- [Kang et Katsaggelos, 1995] KANG, M. G. et KATSAGGELOS, A. K. (1995). General choice of the regularization functional in regularized image restoration. *IEEE Transactions on Image Processing*, 4(5) : 594 – 602.
- [Kanungo *et al.*, 2002] KANUNGO, T., MOUNT, D. M., NETANYAHU, N. S., PIATKO, C. D., SILVERMAN, R. et WU, A. Y. (2002). An efficient k -means clustering algorithm : Analysis and implementation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7) : 881 – 892.
- [Kao *et al.*, 2004] KAO, C. Y., OSHER, S. et QIAN, J. (2004). Lax-friedrichs sweeping scheme for static hamilton-jacobi equations. *Journal of Computational Physics*, 196(1) : 367 – 391.
- [Kervrann et Boulanger, 2008] KERVRANN, C. et BOULANGER, J. (2008). Local adaptivity to variable smoothness for exemplar-based image regularization and representation. *International Journal of Computer Vision*, 79(1) : 45 – 69.
- [Kindermann *et al.*, 2005] KINDERMANN, S., OSHER, S. et JONES, P. W. (2005). Deblurring and denoising of images by nonlocal functionals. *SIAM Multiscale Modeling and Simulation*, 4(4) : 1091 – 1115.

- [Kolmogorov et Zabih, 2004] KOLMOGOROV, V. et ZABIH, R. (2004). What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2) : 147 – 159.
- [Lafon et Lee, 2006] LAFON, S. et LEE, A. B. (2006). Diffusion maps and coarse-graining : A unified framework for dimensionality reduction, graph partitioning, and data set parameterization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(9) : 1393 – 1403.
- [Lee, 1983] LEE, J.-S. (1983). Digital image smoothing and the sigma filter. *Computer Vision, Graphics and Image Processing*, 24(2) : 255 – 269.
- [Leveque, 2002] LEVEQUE, R. J. (2002). *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press.
- [Levin et al., 2004] LEVIN, A., LISCHINSKI, D. et WEISS, Y. (2004). Colorization using optimization. *ACM Transactions on Graphics*, 23 : 689 – 694.
- [Lézoray et al., 2007] LÉZORAY, O., BOUGLEUX, S. et ELMOATAZ, A. (2007). Parameterless discrete regularization on graphs for color image filtering. In *Proceedings of International Conference on Image Analysis and Recognition ICIAR*, LNCS 4633, pages 46 – 57.
- [Lézoray et Cardot, 2002] LÉZORAY, O. et CARDOT, H. (2002). Cooperation of color pixel classification schemes and color watershed : a study for microscopic images. *IEEE Transactions on Image Processing*, 11(7) : 783 – 789.
- [Lézoray et al., 2009a] LÉZORAY, O., CHARRIER, C. et ELMOATAZ, A. (2009a). Learning complete lattices for manifold mathematical morphology. In *Proceedings of International Symposium on Mathematical Morphology ISMM*. accepté.
- [Lézoray et al., 2009b] LÉZORAY, O., LEBRUN, G., MEURIE, C., CHARRIER, C., ELMOATAZ, A., LECLUSE, M. et ELIE, H. (2009b). *Machine Learning In Morphological Segmentation*, chapitre 21, pages 320 – 334. Information Science Reference. in Handbook of Research on Advanced Techniques in Diagnostic Imaging and Biomedical Applications.
- [Lézoray et al., 2008] LÉZORAY, O., MEURIE, C. et ELMOATAZ, A. (2008). Graph-based ordering scheme for color image filtering. *International Journal of Image and Graphics*, 8(3) : 472 – 493.
- [Maier et al., 2009] MAIER, M., HEIN, M. et von LUXBURG, U. (2009). Optimal construction of k-nearest neighbor graphs for identifying noisy clusters. *Theoretical Computer Science*, 410 : 1749 – 1764.
- [Malladi et Sethian, 1996] MALLADI, R. et SETHIAN, J. A. (1996). A unified approach to noise removal, image enhancement and shape recovery. *IEEE Transactions on Image Processing*, 5(11) : 1554 – 1568.
- [Maragos, 2003] MARAGOS, P. (2003). Algebraic and pde approaches for lattice scale-spaces with global constraints. *International Journal of Computer Vision*, 52(2/3) : 121 – 137.
- [Maragos et Meyer, 1999] MARAGOS, P. et MEYER, F. (1999). Nonlinear pdes and numerical algorithms for modeling levelings and reconstruction filters. In *Proceedings of Scale-Space*, LNCS 1682, pages 363 – 374.

- [Maragos et Vachier, 2008] MARAGOS, P. et VACHIER, C. (2008). A pde formulation for viscous morphological operators with extensions to intensity-adaptive operators. *In Proceedings of International Conference on Image Processing ICIP*, pages 2200 – 2202.
- [Matheron, 1967] MATHERON, G. (1967). *Éléments pour une théorie des milieux poreux*. Masson, Paris.
- [Matheron, 1975] MATHERON, G. (1975). *Random Sets and Integral Geometry*. New York : Wiley.
- [Meila et Shi, 2000] MEILA, M. et SHI, J. (2000). Learning segmentation by random walks. *In Advances in Neural Information Processing Systems*, pages 873 – 879.
- [Meyer, 2001] MEYER, F. (2001). An overview of morphological segmentation. *Journal of Pattern Recognition and Artificial Intelligence*, 15(7) : 1089 – 1118.
- [Meyer et Lerallut, 2007] MEYER, F. et LERALLUT, R. (2007). Morphological operators for flooding, leveling and filtering images using graphs. *In Proceedings of Workshop on Graph-based Representations in Pattern Recognition (GBR)*, pages 158 – 167.
- [Meyer et Maragos, 2000] MEYER, F. et MARAGOS, P. (2000). Nonlinear scale-space representation with morphological levelings. *Journal of Visual Communication and Image Representation*, 11 : 245 – 265.
- [Mumford et Shah, 1989] MUMFORD, D. et SHAH, J. (1989). Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42 : 577 – 685.
- [Najman et Schmitt, 1994] NAJMAN, L. et SCHMITT, M. (1994). Watershed of a continuous function. *Signal Processing*, 38 : 99 – 112.
- [Najman et Talbot, 2008] NAJMAN, L. et TALBOT, H. (2008). *Morphologie mathématique 1 : approches déterministes*. Hermes.
- [Newman et al., 2006] NEWMAN, M., BARABASI, A.-L. et WATTS, D. J. (2006). *The Structure and Dynamics of Networks*. Princeton University Press.
- [Osher, 1993] OSHER, S. (1993). A level set formulation for the solution of the dirichlet problem for hamilton-jacobi equations. *SIAM Journal on Mathematical Analysis*, 24(5) : 1145–1152.
- [Osher et Sethian, 1988] OSHER, S. et SETHIAN, J. A. (1988). Fronts propagating with curvature dependent speed : Algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics*, 79 : 12 – 49.
- [Osher et Shen, 2000] OSHER, S. et SHEN, J. (2000). Digitized pde method for data restoration. *Analytical-Computational Methods in Applied Mathematics*, pages 751 – 771.
- [Park et al., 2007] PARK, J., KANG, D., KIM, J., KWOK, J. T. et TSANG, I. W. (2007). Svdd-based pattern denoising. *Neural Computation*, 19(7) : 1919 – 1938.
- [Perona et Malik, 1990] PERONA, P. et MALIK, J. (1990). Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7) : 629 – 639.
- [Peyré, 2009] PEYRÉ, G. (2009). Manifold models for signals and images. *Computer Vision and Image Understanding*, 113(2) : 249 – 260.

- [Peyré *et al.*, 2008] PEYRÉ, G., BOUGLEUX, S. et COHEN, L. (2008). Non-local regularization of inverse problems. *In Proceedings of 10th ECCV 2008*, LNCS 5304, pages 57 – 68. Springer.
- [Pizarro *et al.*, 2009] PIZARRO, L., BURGETH, B., BREUSS, M. et WEICKERT, J. (2009). A directional rouy-tourin scheme for adaptive matrix-valued morphology. *In In the Proceedings of International Symposium on Mathematical Morphology (ISMM)*, LNCS 5720, pages 102 – 114.
- [Postaire *et al.*, 1993] POSTAIRE, J.-G., ZHANG, R.-D. et LECOCQ-BOTTE, C. (1993). Cluster analysis by binary morphology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(2) : 170 – 180.
- [Ren et Malik, 2003] REN, X. et MALIK, J. (2003). Learning a classification model for segmentation. *In In Proceedings of International Conference on Computer Vision (ICCV)*, pages 10 – 17.
- [Rivest *et al.*, 1993] RIVEST, J.-F., SOILLE, P. et BEUCHER, S. (1993). Morphological gradients. *Journal of Electronic Imaging*, 2(4) : 326 – 336.
- [Ronse, 1990] RONSE, C. (1990). Why mathematical morphology needs complete lattices. *Signal Processing*, 21(2) : 129 – 154.
- [Roussos et Maragos, 2009] ROUSSOS, A. et MARAGOS, P. (2009). Reversible interpolation of vectorial images by anisotropic diffusion-projection pde. *International Journal of Computer Vision*, 83(2) : to appear.
- [Rouy et Tourin, 1992] ROUY, E. et TOURIN, A. (1992). A viscosity solutions approach to shape-from-shading. *SIAM Journal on Numerical Analysis*, 3 : 867 – 884.
- [Rudin *et al.*, 1992] RUDIN, L. I., OSHER, S. et FATEMI, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D*, 60(1 – 4) : 259 – 268.
- [Russ, 2007] RUSS, J. C. (2007). *The Image Processing Handbook*. CRC Press.
- [Salembier et Serra, 1995] SALEMBIER, P. et SERRA, J. (1995). Flat zones filtering, connected operators, and filters by reconstruction. *IEEE Transactions on Image Processing*, 4(8) : 1153 – 1160.
- [Sapiro, 2001] SAPIRO, G. (2001). *Geometric Partial Differential Equations and Image Processing*. Cambridge University Press.
- [Sapiro *et al.*, 1993] SAPIRO, G., KIMMEL, R., SHAKED, D., KIMIA, B. B. et BRUCKSTEIN, A. M. (1993). Implementing continuous-scale morphology via curve evolution. *Pattern Recognition*, 26(9) : 1363 – 1372.
- [Schüpp *et al.*, 2001] SCHÜPP, S., ELMOATAZ, A., FADILI, M.-J. et BLOYET, D. (2001). Fast statistical level sets image segmentation for biomedical applications. *In In Proceedings of Scale-Space and Morphology in Computer Vision*, LNCS 2106, pages 380 – 388.
- [Serra, 1967] SERRA, J. (1967). *Échantillonnage et estimation des phénomènes de transition minier*. Thèse de doctorat, Université de Nancy – France.
- [Serra, 1982] SERRA, J. (1982). *Image Analysis and Mathematical Morphology*. London : Academic.

- [Serra, 1988] SERRA, J. (1988). *Image Analysis and Mathematical Morphology vol. 2 : Theoretical Advances*. Academic Press.
- [Sertel et al., 2009] SERTEL, O., KONG, J., SHIMADA, H., CATALYUREK, U. V., SALTZ, J. H. et GURCAN, M. N. (2009). Computer-aided prognosis of neuroblastoma on whole-slide images : classification of stromal development. *Pattern Recognition*, 42 : 1093 – 1103.
- [Sethian, 1999a] SETHIAN, J. A. (1999a). A fast marching level set methods for monotonically advancing fronts. *Proc. Nar. Acad. Sci.*, 41(2) : 199–235.
- [Sethian, 1999b] SETHIAN, J. A. (1999b). *Level Set Methods and Fast Marching Methods : Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision and Materials Science*. Cambridge University Press.
- [Sethian et Vladimirsky, 2003] SETHIAN, J. A. et VLADIMIRSKY, A. (2003). Ordered upwind methods for static hamilton-jacobi equations : Theory and algorithms. *SIAM J. Num. Anal.*, 41(1) : 325–363.
- [Sherrah et Gong, 2001] SHERRAH, J. et GONG, S. (2001). Fusion of perceptual cues for robust tracking of head pose and position. *Pattern Recognition*, 34 : 1565 – 1572.
- [Shi et Malik, 2000] SHI, J. et MALIK, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8) : 888 – 905.
- [Shu, 2007] SHU, C.-W. (2007). High order numerical methods for time dependent hamilton-jacobi equations. *Mathematics and Computation in Imaging Science and Information Processing*, 11 : 47–91.
- [Shu et Zhang, 2003] SHU, C.-W. et ZHANG, Y.-T. (2003). High order weno schemes for hamilton-jacobi equations on triangular meshes. *SIAM J. Scien. Comp.*, 24 : 1005–1030.
- [Siddiqi et al., 1999] SIDDIQI, K., BOUIX, S., TANNENBAUM, A. et ZUCKER, S. W. (1999). The hamilton-jacobi skeleton. In *Proceedings of International Conference on Computer Vision (ICCV)*, pages 828–834.
- [Siddiqi et al., 1997] SIDDIQI, K., KIMIA, B. B. et SHU, C.-W. (1997). Geometric shock-capturing eno schemes for subpixel interpolation, computation and curve evolution. *Graphical Models and Image Processing*, 59(5) : 278 – 30.
- [Signolle et al., 2009] SIGNOLLE, N., REVENU, M., PLANCOULAIN, B. et HERLIN, P. (2009). Wavelet-based multiscale texture segmentation : Application to stromal compartment characterization on virtual slides. *Signal Processing*. A paraître.
- [Sinop et Grady, 2007] SINOP, A. K. et GRADY, L. (2007). A seeded image segmentation framework unifying graph cuts and random walker which yields a new algorithm. In *In Proceedings of International Conference on Computer Vision (ICCV)*, pages 1 – 8.
- [Smith et Brady, 1997] SMITH, S. M. et BRADY, J. M. (1997). Susan - a new approach to low level image processing. *International Journal of Computer Vision*, 23(1) : 45 – 78.
- [Soille, 2002] SOILLE, P. (2002). *Morphological Image Analysis, Principles and Applications*. Springer, deuxième édition.

- [Szlam *et al.*, 2008] SZLAM, A. D., MAGGIONI, M. et COIFMAN, R. D. (2008). Regularization on graphs with function-adapted diffusion processes. *Journal of Machine Learning Research*, 9 : 1711 – 1739.
- [Ta *et al.*, 2007] TA, V.-T., BOUGLEUX, S., ELMOATAZ, A. et LÉZORAY, O. (2007). Non-local anisotropic discrete regularization for image, data filtering and clustering. Rapport technique, GREYC CNRS UMR 6072 - Université de Caen Basse-Normandie - ENSICAEN - hal-00187165.
- [Thorstensen *et al.*, 2009] THORSTENSEN, N., SEGONNE, F. et KERIVEN, R. (2009). Pre-image as karcher mean using diffusion maps : Application to shape and image denoising. In *In Proceedings of International Conference on Scale Space and Variational Methods in Computer Vision (Ssvm)*, LNCS 5567, pages 721 – 732.
- [Tikhonov et Arsenin, 1977] TIKHONOV, A. N. et ARSENIN, V. Y. (1977). *Solutions of Ill-Posed Problems*. Wiley.
- [Tomasi et Manduchi, 1998] TOMASI, C. et MANDUCHI, R. (1998). Bilateral filtering for gray and color images. In *Proceedings of International Conference on Computer Vision (Iccv)*, pages 839 – 846.
- [Tsai *et al.*, 2004] TSAI, Y.-H. R., CHENG, L.-T., OSHER, S. et ZHAO, H.-K. (2004). Fast sweeping algorithms for a class of hamilton-jacobi equations. *SIAM Journal on Numerical Analysis*, 41(2) : 673 – 694.
- [Tsai et Osher, 2005] TSAI, Y.-H. R. et OSHER, S. (2005). Total variation and level set methods in image science. *Acta Numerica*, pages 509–573.
- [Tschumperlé, 2009] TSCHUMPERLÉ, D. (2009). Lissage non local d'images multi-valuées par diffusion anisotrope dans l'espace des patches. In *Groupe de recherche sur le traitement du signal et des Images (GRETSI)*.
- [Tschumperlé et Deriche, 2005] TSCHUMPERLÉ, D. et DERICHE, R. (2005). Vector-valued image regularization with pdes : A common framework for different applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(4) : 506 – 517.
- [Tsitsiklis, 1995] TSITSIKLIS, J. N. (1995). Efficient algorithms for globally optimal trajectories. *IEEE Transactions on Automatic Control*, 40(9) : 1528–1538.
- [van den Boomgaard et Smeulders, 1994] van den BOOMGAARD, R. et SMEULDERS, A. (1994). The morphological structure of images : the differential equations of morphological scale-space. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(11) : 1101 – 1113.
- [Verbeek et Verwer, 1990] VERBEEK, P. W. et VERWER, B. J. H. (1990). Shading from shape, the eikonal equation solved by grey-weighted distance transform. *Pattern Recognition Letters*, 11(10) : 681 – 690.
- [Vincent, 1993] VINCENT, L. (1993). Morphological gray scale reconstruction in image analysis : Applications and efficient algorithms. *IEEE Transactions on Image Processing*, 2(2).
- [Vincent et Soille, 1991] VINCENT, L. et SOILLE, P. (1991). Watersheds in digital spaces : an efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6) : 583 – 598.

- [von Luxburg, 2007] VON LUXBURG, U. (2007). A tutorial on spectral clustering. *Statistics and Computing*, 17(4) : 395 – 416.
- [Weickert, 1998] WEICKERT, J. (1998). *Anisotropic Diffusion in Image Processing*. ECMI Series. Teubner-Verlag.
- [Yaroslavsky, 1985] YAROSLAVSKY, L. P. (1985). *Digital Picture Processing*. Springer-Verlag.
- [Yatziv et Sapiro, 2006] YATZIV, L. et SAPIRO, G. (2006). Fast image and video colorization using chrominance blending. *IEEE Transactions on Image Processing*, 15(5) : 1120 – 1129.
- [Yu et Shi, 2004] YU, S. X. et SHI, J. (2004). Segmentation given partial grouping constraints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2) : 173 – 183.
- [Zhang et al., 2006] ZHANG, Y.-T., ZHAO, H.-K. et QIAN, J. (2006). High order fast sweeping methods for static hamilton-jacobi equations. *Journal of Scientific Computing*, 29(1) : 25–56.
- [Zhao, 2005] ZHAO, H. (2005). Fast sweeping method for eikonal equations. *Mathematics of Computation*, 74 : 603–627.
- [Zhou et Schölkopf, 2004] ZHOU, D. et SCHÖLKOPF, B. (2004). A regularization framework for learning from graph data. In *Proceedings of International Conference on Machine Learning 2004 Workshop on Statistical Relational Learning and Its Connections to Other Fields*, pages 132 – 137.
- [Zhou et Schölkopf, 2006] ZHOU, D. et SCHÖLKOPF, B. (2006). *Discrete Regularization*, chapitre 3.13, pages 221 – 232. Adaptive computation and machine learning. MIT Press.
- [Zhu, 2008] ZHU, X. (2008). Semi-supervised learning literature survey. Rapport technique 1530, University of Wisconsin – Madison.

Index

- A**
adjacents 9
adjoint 20
apprentissage semi supervisé 164
arête 9
- B**
bord externe 24
bord interne 24
- C**
Chan-Vese discret 107
classification semi supervisé 76
co-aire 98
colorisation 73
courbure 22
courbure pondéré 22
cytologie 85
- D**
degré 10
différence pondérée 19
différence pondérée externe 22
différence pondérée interne 22
différences directionnelles pondérées 22
dilatation 118, 120
divergence 20
donnée 10
- E**
élément structurant 118, 121
énergie 43
équation eikonale 149
érosion 118, 120
espace de Hilbert 18
- F**
fonction de poids 9, 16
fonction structurante 120
fonctionnelles isotropes 37
fonctions sur des graphes 18
- G**
gradient morphologique 125
gradient pondéré 20
gradient pondéré externe 23
gradient pondéré interne 23
gradients directionnels pondérés 23
graphe d'adjacence des régions 13
graphe grille 12
graphe non-orienté 9
graphe pondéré 9
graphes de voisinage 11
graphes des images 12
- H**
Hamilton-Jacobi statiques 149
histologie 85
- I**
imagerie de lame entière 91
images cellulaires 83
interpolation 73
- L**
la requête 161
labels 76
Laplacien combinatoire 21
Laplacien morphologique 125
Laplacien pondéré 21
le calcul de distances généralisées .. 157
- M**
marqueur 130
marqueurs 74
minimisation d'énergie 44
morphologie mathématique planaire 121
dilatation et érosion multi échelles . 121
- N**
nivellements 130
non locale 15
non organisé 10
noyau de régularisation 47
- O**
optimisation globale 97
ordonnancement 161

organisé 10
ouverture 132

P

p-Laplacien pondéré 30
p-Laplacien pondéré anisotrope 29
p-Laplacien pondéré isotrope 21
périmètre 41
partitionnement 111
patches 14
propagation de fronts 157

R

référence 130
reconstruction par fermeture 131
reconstruction par ouverture 131
ROF discret 104

S

Schéma de segmentation automatique 91
schéma de segmentation interactive . 88
segmentation semi supervisé 76
semi supervisé 164
similarité 10
sommets 9

T

treillis 118

V

variation totale 37
vecteur d'attributs 10, 14
voisinage 9
voisins 9

Table des matières

Remerciements	i
Sommaire	iii
1 Introduction générale	1
2 Graphes et opérateurs	7
2.1 Introduction	9
2.2 Définitions et notations	9
2.3 Structure et topologie	10
2.3.1 Domaines non organisés	10
Graphe de ν -voisinage	11
Graphe des k plus proches voisins	11
2.3.2 Domaines organisés	12
2.3.3 Similarités entre les sommets	14
Vecteurs d'attributs	14
Fonctions de poids	16
2.4 Opérateurs définis sur des graphes	17
2.4.1 Fonctions définies sur des graphes	18
2.4.2 Opérateur de différence et son adjoint	19
2.4.3 Opérateur gradient	20
2.4.4 p -Laplacien isotrope	21
Propriétés du p -Laplacien isotrope	21
2.5 Une nouvelle famille de différences	22
2.6 Une nouvelle famille de gradients	23
2.7 Propriétés et relations des gradients	24
2.8 Formulation générale du p -Laplacien	29
2.8.1 p -Laplacien anisotrope	29
2.8.2 p -Laplacien	30
2.9 Propriétés et relations du p -Laplacien	30
2.10 Conclusion	31
3 Fonctionnelles discrètes de régularisation	33
3.1 Fonctionnelles de régularisation	37
3.1.1 Introduction	37
3.1.2 Une première famille de fonctionnelles	37
3.1.3 Une seconde famille de fonctionnelles	38
3.1.4 Formulation générale	38
3.1.5 Propriétés des deux familles de fonctionnelles	38
Relations entre les fonctionnelles de régularisation	39
Relations avec les périmètres discrets d'un graphe	40
Relations avec les fonctionnelles continues locales et non locales	42
3.1.6 Modèles variationnels discrets	43
Deux classes de modèles variationnels	45
3.1.7 Conclusion	45

3.2	Modèles de régularisation, filtrage et simplification	47
3.2.1	Introduction	47
3.2.2	Solutions de la première classe de modèles variationnels	48
3.2.3	Algorithme discret d'approximation	51
3.2.4	Cas des données vectorielles	52
3.2.5	Modèle p -TV isotrope	53
	Liens avec des filtres de voisinage	54
3.2.6	Modèle p -TV isotrope, filtrage et simplification de données . . .	56
	Débruitage de collections d'images	56
	Simplification de variétés	58
	Filtrage et simplification comme pré-traitement	60
3.2.7	Modèle p -TV anisotrope	64
3.2.8	Modèle p -TV anisotrope, filtrage et simplification de données .	66
	Débruitage de collections d'images	66
	Simplification de variétés	66
	Filtrage et simplification comme pré-traitement	68
3.2.9	Conclusion	69
3.3	Problèmes d'interpolation basés sur la régularisation	73
3.3.1	Introduction	73
3.3.2	Colorisation d'images	73
3.3.3	Segmentation et classification semi supervisées	76
	Approches basées sur les modèles p -TV isotrope et anisotrope	77
	Expérimentations	78
3.3.4	Outils pour le traitement d'images cellulaires	83
	Images cellulaires	85
	Extraction d'objets cellulaires	85
	Schéma de segmentation interactive	88
	Comparaisons avec d'autres méthodes	88
	Schéma de segmentation automatique de lames entières	91
3.3.5	Conclusion	94
3.4	Optimisation globale pour la segmentation de données	97
3.4.1	Introduction	97
3.4.2	Solutions de la deuxième classe de modèles variationnels	98
	Formule de la co-aire sur des graphes	98
	Méthode discrète de relaxation	100
	Algorithme itératif d'approximation	102
3.4.3	Modèle discret de Rudin, Osher et Fatemi	104
3.4.4	Modèle discret de Chan et Vese	107
	Segmentation d'images	108
	Partitionnement de données	111
3.4.5	Conclusion	111
4	Morphologie mathématique	115
4.1	Introduction	118
4.2	Dilatation et érosion	120
4.2.1	Approche basée sur les équations aux dérivées partielles	121

4.3	Morphologie mathématique par équations aux différences partielles	122
4.3.1	Différences et gradients morphologiques	123
	Différences directionnelles pondérées morphologiques	123
	Gradients directionnels pondérés morphologiques	124
4.3.2	Processus de dilatation et d'érosion	125
4.3.3	Schémas numériques pour la dilatation et l'érosion	126
4.3.4	Relations avec la morphologie mathématique continue	127
	Schéma numérique de Osher et Sethian	127
	Cas particulier d'une grille régulière 2D	128
4.3.5	Relations avec la morphologie mathématique définie sur des graphes	129
4.3.6	Filtrage par nivellements	130
	Nivellements basés sur les équations aux dérivées partielles	130
	Adaptation des filtres par nivellements	131
4.4	Expérimentations	132
4.4.1	Cas des données vectorielles	132
4.4.2	Traitements morphologiques des images	132
4.4.3	Nivellements des images	137
4.4.4	Traitements morphologiques à partir de graphes de régions	137
4.4.5	Traitements morphologiques de données arbitraires	139
	Dilatation et érosion de base de données	140
	Filtrage par nivellements de nuages de points	141
	Traitements morphologiques de collections d'images	143
4.5	Conclusion	143
5	Équation eikonale	147
5.1	Introduction	149
5.2	Adaptation de l'équation eikonale	153
5.3	Schémas numériques et algorithmes associés	154
5.3.1	Relations avec le schéma de Osher et Sethian	155
5.3.2	Algorithme du type Dijkstra	156
5.4	Expérimentations	157
5.4.1	Calcul de distances généralisées	157
	Calculs de distances pour les images	157
	Calculs de distances pour des données organisées	159
	Calculs de distances pour des bases de données	160
5.4.2	Segmentation	164
	Segmentation non locale d'images	165
	Segmentation avec des graphes de régions	167
	Classification de bases de données	171
5.5	Conclusion	173
6	Conclusion générale	175
	Liste des publications	181

Bibliographie	183
Index	195
Tables des matières	197

Résumé Cette thèse s'intéresse aux traitements d'images et de données non uniformes en utilisant le formalisme des équations aux différences partielles définies sur des graphes pondérés. Nous exploitons ce formalisme afin de transcrire et d'adapter des modèles définis dans le domaine continu vers des formulations discrètes. Les modèles continus considérés dans ce manuscrit proviennent du domaine du traitement des images et sont définis comme des modèles variationnels ou des approches basées sur des équations aux dérivées partielles. Nous nous sommes intéressés à des modèles de régularisation, à la morphologie mathématique et à l'équation eikonale. Afin de transcrire ces modèles définis dans le domaine continu vers des formulations discrètes, nous avons introduit une large famille de nouveaux opérateurs différentiels discrets définis sur des graphes pondérés : différences pondérées, gradients discrets, p -Laplacien. Ces opérateurs permettent de redéfinir les modèles continus considérés dans un cadre discret mais également de proposer un formalisme général permettant de considérer de nombreux problèmes liés aux traitements des images et, plus généralement, de données arbitraires.

À partir des modèles discrets de régularisation, de morphologie mathématique et de l'équation eikonale, nous montrons dans ce manuscrit les potentialités de notre formalisme pour des applications telles que le filtrage, la simplification, la segmentation, le regroupement et la classification d'images et de données. Notre formalisme unifie également les traitements locaux et non locaux basés sur des patches. Nous avons généralisé l'utilisation de ce type de configuration dans les problématiques considérées et montré la supériorité de ces schémas dans le contexte du traitement des images. Notre formalisme est basé sur des graphes pondérés. Cela nous permet d'étendre les modèles définis dans le domaine continu aux traitements de n'importe quel type de donnée pouvant être représenté par cette structure (par exemple des images, des collections d'images, des nuages de points, des variétés, des bases de données, etc.). Finalement, ces travaux de thèse permettent d'envisager de nombreuses pistes de recherche tant dans le domaine du traitement des images que dans des domaines tels que celui de l'apprentissage ou de la fouille de données.

Mots-clés *Indexation RAMEAU* : Équations aux différences / Traitement d'images – Techniques numériques / Morphologie mathématique / Calculs des variations
Indexation libre : Graphes pondérés / Données arbitraires / Régularisation et équation eikonale / Filtrage et segmentation

Title Partial difference equations on graphs for image and data processing.

Abstract This work deals with images and non uniform data processing by partial difference equations (PdE) over weighted graphs. Transcription and adaption of continuous models to discrete formulations are considered within this PdEs-based framework. The considered continuous models (from image processing domain) are defined as variational models or approaches based on partial differential equations. The continuous models considered in this work are: regularization models, mathematical morphology, and the eikonal equation. To adapt these latter models within a discrete setting, we introduce a large family of discrete differential operators defined on weighted graphs: weighted differences, discrete gradients and p -Laplacian. These operators enable the transcription and the adaption of continuous models and provide a general formulation for considering numerous applications for images and arbitrary data.

Potentialities of our discrete regularization, mathematical morphology and eikonal equation models are shown in applications such as image and data filtering, simplification, segmentation, clustering and classification. Our formulation also unifies local and non local patch-based processing. We have intensively used this latter configuration and shown the superiority of such a scheme in the context of image processing. Our approach is based on weighted graphs. This point provides a natural extension of continuous models for the processing of arbitrary data that can be represented by a weighted graph (for instance: images, manifolds, data sets, data bases, etc.). Finally, this work opens new insights for image processing and new possible applications in machine learning.

Keywords *Rameau index* : Difference equations / Image processing – numerical schemes / Mathematical morphology / Calculus of variations
Free index : Weighted graphs / Arbitrary data / Regularization and Eikonal equation / Filtering and segmentation

Discipline Informatique et applications

Université de Caen Basse-Normandie, ENSICAEN, CNRS
GREYC – équipe image
6 boulevard Maréchal Juin, 14050 Caen cedex, France