

# THÈSE

Présentée par  
**Mohamed ELAYYADI**

Pour obtenir le titre de  
**Docteur de l'Université Joseph Fourier - Grenoble I.**  
(arrêtés ministériels du 5 juillet 1984 et du 30 mars 1992)

Spécialité : **Mathématiques Appliquées.**

---

## **Equations aux Dérivées Partielles et Réseaux de Neurones pour le Traitement d'Images**

---

Date de soutenance : 14 octobre 1997.

Composition du jury :

M. J. DEMONGEOT	Président
M. V. CASELLES	Rapporteur
M. M. RASCLE	Rapporteur
M. G.-H. COTTET	Directeur de thèse
M. P. CINQUIN	Examineur
M. B. MICHAUX	Examineur
M. P. WITOMSKI	Examineur

Thèse préparée au laboratoire: LMC-IMAG  
en collaboration avec le laboratoire TIMC-IMAG

Mes remerciements les plus vifs et les plus sincères vont d'abord à M. Georges-Henri Cottet, Professeur à l'Université Joseph Fourier, qui a dirigé et soutenu ces travaux de recherches, tant pour ses idées qui sont à l'origine de cette thèse, que pour ses encouragements et conseils tout au long de ces années au sein de l'équipe EDP du laboratoire LMC-IMAG.

Comment ne pas remercier M. Jacques Demongeot, membre de l'Institut Universitaire de France et directeur du laboratoire TIMC-IMAG, pour sa disponibilité, sa gentillesse et son soutien durant la préparation de cette thèse. Je tiens à lui témoigner ma profonde gratitude. Il m'a fait l'honneur de présider ce jury.

J'espère que la collaboration avec ces deux enseignants continuera encore longtemps.

J'exprime toute ma reconnaissance à M. Vicent Caselles, Professeur à l'Université des Iles Baléares. Nos conversations, ainsi que son intérêt à l'égard de mon travail m'ont encouragé. Je le remercie en outre d'avoir accepté de rapporter sur ma thèse.

J'adresse également mes remerciements à M. Michel Rascle, Professeur à l'Université de Nice Sophia-Antipolis pour avoir bien voulu être rapporteur de la thèse, ainsi que pour ses nombreuses remarques fructueuses qui m'ont aidées à améliorer la rédaction du mémoire.

Je remercie MM. Philippe Cinquin, Professeur à la Faculté de Médecine et Patrick Witomski, directeur du laboratoire LMC-IMAG, qui ont bien voulu faire partie de ce jury.

Je remercie aussi M. Bertrand Michaux, Maître de Conférences à l'Université Joseph Fourier, qui a accepté d'être membre de ce jury et qui a toujours été disponible pour des discussions diverses notamment en mathématiques.

Je tiens à remercier Frank Leitner, pour sa collaboration, spécialement, mais pas uniquement en imagerie médicale, et pour sa sympathie.

J'exprime toute ma sympathie à M. Joachim Weickert pour l'aide précieuse qu'il m'a apportée au cours de notre collaboration.

Je n'oublierai pas les thésards et néanmoins amis du LMC sans qui l'ambiance au labo n'aurait pas été la même.

Mes remerciements vont aussi à mes parents et à Alexia, pour leur soutien et leur patience.

*A mes parents,*

*A ma famille.*

## Résumé

CET travail porte sur des techniques à base d'équations aux dérivées partielles et de réseaux de neurones pour le traitement d'images.

L'approximation des réseaux de neurones par des systèmes de réaction-diffusion nous a permis de définir un nouveau modèle de diffusion anisotrope de type Volterra pour le filtrage sélectif d'images bruitées. La loi d'évolution régissant le tenseur de diffusion traduit des lois d'apprentissage synaptiques naturelles.

L'étude de la dynamique de ces réseaux à synapses adaptatives montre qu'ils possèdent des propriétés d'attractivité et de stabilité asymptotique au sens de Lyapunov. Les images traitées sont donc obtenues sur les asymptotiques en temps du modèle.

Les techniques présentées dans cette thèse améliorent de manière importante le pré-traitement d'images car elles ne nécessitent qu'une connaissance *a priori* d'un paramètre de contraste sur l'image désirée et permettent la restauration des images ayant subi jusqu'à 90% de niveau de bruit et la segmentation des images médicales d'échographie.

**Mots clés:** Diffusion anisotrope, équation de Volterra, réseaux de neurones, traitement d'images, segmentation.

## Abstract

THIS work deals with mathematical tools based both on partial differential equations and neural networks for images processing.

Approximation of neural network dynamics by reaction-diffusion equations enable us to introduce a new nonlinear anisotropic diffusion model of selective filters for image processing. This Volterra type equations allows the diffusion tensor to change dynamically in order to process the image by a combination of smoothing and contrast enhancement. Moreover, they exhibit strong stability properties which permit acquisition of the desired filtered image on the asymptotic (in time) state of the model. In other terms, their use requires only an *a priori* knowledge of a contrast parameter about the desired image.

The experimental results on test and medical images shown in this thesis illustrate the ability of the model to detect fine objects in a highly degraded images.

**Key words:** Anisotropic diffusion, Volterra equation, neural network, image enhancement, image segmentation.

# Table des matières

<b>Introduction Générale</b>	<b>0</b>
<b>1 Equations aux Dérivées Partielles pour le Traitement d'Image</b>	<b>3</b>
1.1 Introduction . . . . .	3
1.2 Filtres fondés sur la diffusion linéaire . . . . .	3
1.2.1 Filtre de Gauss . . . . .	4
1.2.2 Variantes du filtrage Gaussien . . . . .	5
1.3 Filtres non linéaires . . . . .	7
1.3.1 Le modèle de Perona et Malik . . . . .	8
1.3.2 Versions régularisées du modèle Perona-Malik . . . . .	10
1.3.3 Filtre limiteur d'oscillation . . . . .	12
1.4 Filtres basés sur la notion de courbure et contours actifs . . . . .	14
1.4.1 Filtres basés sur la notion de courbure ( <i>MCM</i> ) . . . . .	14
1.4.2 <i>MCM</i> et contours actifs . . . . .	16
1.5 Filtres par minimisation d'énergie . . . . .	19
1.5.1 Le modèle de Mumford-Shah . . . . .	19
1.5.2 Le modèle de Nordström . . . . .	21
1.6 Approche morphologique et Analyse multi-échelle . . . . .	22
1.6.1 Une approche morphologique . . . . .	22
1.6.2 Une analyse multi-échelle . . . . .	23
1.7 Diffusion tensorielle . . . . .	23
1.8 Résolutions Numériques . . . . .	26
1.9 Domaines d'applications . . . . .	27
1.10 Conclusions et Discussion . . . . .	27
<b>2 Réseaux de Neurones et Diffusion Anisotrope</b>	<b>29</b>
2.1 Introduction . . . . .	29
2.2 Le modèle dynamique de Hopfield . . . . .	30
2.2.1 Définitions . . . . .	30
2.2.2 Propriétés dynamiques . . . . .	32
2.3 Mécanismes d'apprentissage . . . . .	33
2.3.1 Réseaux à mémoire associative . . . . .	33
2.3.2 Réseaux à synapses adaptatives . . . . .	35

2.4	Approximation par systèmes de réaction-diffusion . . . . .	36
2.4.1	Approximation par diffusion anisotrope . . . . .	36
2.4.2	Approximation par diffusion anisotrope et apprentissage . . . . .	38
2.4.3	Introduction d'un seuil dans l'opérateur de diffusion . . . . .	43
2.4.4	Stabilité asymptotique au sens de Lyapunov . . . . .	44
2.5	Conclusion et Discussion . . . . .	47
<b>3</b>	<b>Etude de la Diffusion Anisotrope</b>	<b>49</b>
3.1	Etude Théorique . . . . .	49
3.2	Résolution numérique . . . . .	56
<b>4</b>	<b>Exemples et Applications</b>	<b>61</b>
4.1	Diffusion anisotrope . . . . .	61
4.1.1	Algorithme . . . . .	61
4.1.2	Images tests . . . . .	62
4.1.3	Images Réelles . . . . .	70
4.1.4	Imagerie Médicale . . . . .	72
4.2	Diffusion anisotrope comme pré-traitement pour la segmentation . . . . .	75
4.3	Diffusion avec réinitialisation . . . . .	76
4.3.1	Principe . . . . .	77
4.3.2	Applications . . . . .	78
4.4	Réseaux de neurones . . . . .	83
4.4.1	Algorithme . . . . .	83
4.4.2	Application . . . . .	84
4.5	Conclusions et Perspectives . . . . .	85
	<b>Annexes</b>	<b>86</b>
<b>A</b>	<b>A Volterra type model for image processing</b>	<b>87</b>
<b>B</b>	<b>Image segmentation using snake-splines and anisotropic diffusion operators</b>	<b>101</b>
<b>C</b>	<b>Une étude tridimensionnelle</b>	<b>119</b>
	<b>Bibliographie Générale</b>	<b>123</b>

# Table des figures

2.1	Architecture de réseaux de neurones . . . . .	31
4.1	Diffusion anisotrope d'un disque ( $256 \times 256$ ). (o): l'image originale et l'image diffusée avec (a): $\tau = 5\Delta t$ et $s = 5$ (b): $\tau = 10\Delta t$ et $s = 5$ (c) $\tau = 10\Delta t$ et $s = 10$ . . . . .	62
4.2	Courbes d'erreur (a) et du résidu (b) correspondantes aux images 4.1. . . . .	63
4.3	Application de la diffusion anisotrope à paramètres variables sur une image <i>triangle – rectangle</i> $128 \times 128$ bruitée à 30%. (a): $\tau = 3\Delta t$ et $s = 4$ (b): $\tau = 3\Delta t$ et $s = 10$ (c): $\tau = 10\Delta t$ et $s = 10$ . Les lignes correspondent à l'image originale, aux images restaurées et au seuillage en noir et blanc de ces dernières. . . . .	64
4.4	Courbes d'erreur et du résidu pour différentes valeurs de $\tau$ avec un seuil fixé pour chaque graphe. . . . .	65
4.5	Courbes d'erreur et du résidu pour différentes valeurs du seuil $s$ avec un temps de relaxation $\tau$ fixé pour chaque graphe. . . . .	66
4.6	Étude de l'erreur et du résidu en fonction du bruit introduit dans l'image originale avec un seuil $s$ et un temps de relaxation $\tau$ fixés pour chaque graphe. . . . .	67
4.7	Diffusion anisotrope d'une image <i>triangle – rectangle</i> $256 \times 256$ bruitée à 70%. (a): image originale (b): image filtrée avec $\tau = 5\Delta t$ et $s = 5$ (c): seuillage de l'image traitée (d): Courbe d'erreur. . . . .	69
4.8	Application de l'algorithme sur une image réelle. (a): l'image originale (b): l'image diffusée (c)et (d): Visualisation de ces deux images à l'aide d'une palette de couleurs aléatoires. . . . .	70
4.9	Diffusion d'images réelles. (a): l'image d'un tissu (b) image d'une planche. . . . .	71
4.10	Diffusion anisotrope d'une image IRM ( $256 \times 256$ ) du cerveau. (a): l'image originale, (b): l'image diffusée avec $\tau = 4\Delta t$ et $s = 5$ . . . . .	72
4.11	Image IRM d'une coupe sagittale du crâne: (a): diffusion isotrope (b): diffusion anisotrope avec $\tau = 20\Delta t$ et $s = 20$ (c): $\tau = 5\Delta t$ et $s = 5$ . les lignes correspondent à 50, 500 et 3000 itérations. . . . .	73
4.12	Extraction d'objets monodimensionnels (vaisseaux sanguins) dans une angiographie numérique. (a): l'image originale (b): l'image filtrée avec $\tau = 10\Delta t$ et $s = 5$ . . . . .	74

4.13	Diffusion anisotrope d'une image échographique du fœtus. (a) : l'image originale (b) : image filtrée avec $\tau = 10\Delta t$ et $s = 10$ (c) et (d) : Extraction des contours de ces deux images. . . . .	75
4.14	Segmentation d' une image échographique du coeur ( $384 \times 256$ ) . . . . .	76
4.15	Principe de réinitialisation. . . . .	77
4.16	Diffusion anisotrope avec réinitialisation de l'image <i>triangle-rectangle</i> ( $256 \times 256$ ) bruitée à 90%. . . . .	78
4.17	Réinitialisation de la diffusion anisotrope sur une image fortement texturée et bruitée à 30% . . . . .	79
4.18	Diffusion anisotrope avec réinitialisation d'une image ( $256 \times 256$ ). (a) : image originale (b) : image bruitée à 30% (c) : filtrage de l'image bruitée (d) : filtrage avec une réinitialisation. . . . .	80
4.19	Réinitialisation de la diffusion anisotrope sur une image échographique du coeur . . . . .	81
4.20	Extraction des contours des images (a) et (c) de la figure 4.19. . . . .	82
4.21	Profil des luminescence de la ligne 250 des images de la figure 4.19. . . . .	82
4.22	Lissage de l'image <i>triangle - rectangle</i> bruitée à 70% avec le réseau neuronal. (a) : image originale (b) : image filtrée (c) : seuillage de l'image lissée (d) : la courbe d'erreur. . . . .	84
C.1	Courbes d'erreur (a) et du résidu (b) . . . . .	122

# Introduction Générale

EN modélisation mathématique du traitement d'images, on peut distinguer en particulier deux domaines d'application. Le premier est la restauration des images, qui consiste à améliorer l'image afin de la rendre plus significative à l'oeil en renforçant les contrastes de l'image originale et en corrigeant les distorsions (bruit) introduites lors de son acquisition. L'autre domaine est celui de la segmentation ou d'une manière plus générale celui de l'interprétation des images qui consiste à détecter les contours des objets significatifs de l'image. D'une manière générale, le pré-traitement représente une étape très importante en traitement d'images et en vision par ordinateur, puisqu'il permet un débruitage et une augmentation des contrastes qui facilitent la segmentation par la suite. Le travail présenté dans cette thèse se situe dans le premier domaine. Les techniques que nous y décrivons combinent équations aux dérivées partielles et réseaux de neurones.

Mathématiquement, une image plane, généralement un rectangle, consiste en un ensemble discret de point (*pixel*) porteur d'une fonction  $u$ , le niveau de gris ou la luminosité. Par ailleurs, en traitement d'image et en vision par ordinateur, il est difficile d'analyser l'information contenue dans une image à partir de l'intensité lumineuse de chaque pixel. Par contre, on peut saisir les changements d'intensité et la taille des zones présentant des propriétés d'homogénéité (ou regroupant des pixels faisant partie d'un "même" objet). D'où, la nécessité d'utiliser des équations aux dérivées partielles EDPs qui respectent la taille et les propriétés de ces zones. Ces approches ont de plus l'avantage de donner des résultats d'existence et d'unicité de la solution (image restaurée) et permettent de mettre en oeuvre des schémas numériques stables.

Par ailleurs, les réseaux de neurones constituent une base naturelle pour le traitement d'images et pour la vision par ordinateur. De part leur capacité à approcher les fonctions continues, ils peuvent servir de guide pour la construction d'équations aux dérivées partielles ayant des états asymptotiques non triviaux pour le pré-traitement d'images.

Cette thèse est organisée de la manière suivante. Le chapitre 1 constitue un résumé des modèles les plus récents en traitement d'image, en particulier ceux fondés sur la diffusion linéaire, non linéaire ou anisotrope pour le pré-traitement et la segmentation. Cela permettra de mettre en perspective le travail présenté ici. Dans le chapitre 2, nous étudierons les propriétés dynamiques des réseaux neuronaux à synapses adaptatives ou non. Nous verrons qu'en transportant le comportement collectif d'un réseau infini de neurones dans

le cas continu, ce dernier peut être une base naturelle pour la construction d'équations aux dérivées partielles pour le traitement d'images. L'approximation de ces réseaux par des systèmes de réaction-diffusion, nous permettra par la suite d'introduire un nouveau modèle de type Volterra qui améliore de manière importante le pré-traitement d'images. Nous montrerons aussi que le réseau équivalent possède des propriétés de stabilité asymptotique au sens de Lyapunov. Dans le chapitre 3, nous donnerons un résumé des résultats théoriques et numériques de ces modèles. Enfin, dans le chapitre 4, nous commencerons par étudier les paramètres du modèle proposé, pour ensuite illustrer par des images synthétiques médicales ou réelles. Nous verrons qu'une réinitialisation de l'algorithme proposé donne des résultats très intéressants sur des images fortement bruitées et texturées.

---

# Chapitre 2

## Equations aux Dérivées Partielles pour le Traitement d'Image

### 2.1 Introduction

**E**N matière de traitement du signal et de l'image, une des principales difficultés est de trouver une méthode capable de filtrer sélectivement le bruit tout en préservant les objets significatifs dans l'image originale. Plusieurs ont vu le jour depuis une dizaine d'années.

Ce chapitre retrace les principales contributions en la matière; en particulier, celles basées sur l'utilisation des équations aux dérivées partielles pour l'amélioration des images, l'élimination du bruit et le renforcement des contrastes. Nous évoquerons aussi les techniques EDP utilisées pour la segmentation d'images car elles sont étroitement liées aux modèles de diffusion.

Après avoir introduit les modèles de base, il s'avèrera intéressant de souligner les différents liens existants entre eux et les difficultés pratiques et théoriques de chacun.

### 2.2 Filtres fondés sur la diffusion linéaire

En traitement d'images par les EDP, le premier et le plus simple des modèles à avoir été utilisé est celui fondé sur la diffusion linéaire. Dans ce paragraphe, nous présenterons ce modèle Gaussien ainsi que quelques filtres basés sur des variantes de l'équation de diffusion linéaire.

### 2.2.1 Filtre de Gauss

Soit  $u_0(x)$  le niveau de gris en un point  $x$  de l'image à traiter<sup>1</sup>, et  $G_\sigma$  la Gaussienne d'écart type  $\sigma$  donnée par la formule suivante:

$$G_\sigma(x) = \frac{1}{4\pi\sigma} \exp\left(-\frac{|x|^2}{4\sigma}\right) \quad (2.1)$$

Le filtrage Gaussien de l'image résulte de la convolution de cette fonction  $u_0$  avec des Gaussiennes en chaque point  $x$  de l'image:

$$u(x, \sigma) = (G_\sigma * u_0)(x) = \int_{\mathbb{R}^2} G_\sigma(x - y)u_0(y)dy \quad (2.2)$$

Puisque  $G_\sigma$  est dans  $C^\infty(\mathbb{R}^2)$  il en est de même pour  $G_\sigma * u$ . La convolution est donc une opération régularisante. En pratique, cette régularisation de  $u$  permet de lisser de manière grossière en supprimant l'information qui présente des variations spatiales sur des échelles inférieures à  $\sigma$ .

La transformée de Fourier des deux membres de l'équation (1.2) nous donne:

$$(\mathcal{F}(u))(x, \sigma) = (\mathcal{F}(G_\sigma))(x) \cdot (\mathcal{F}(u_0))(x) \quad (2.3)$$

Or, la transformée de Fourier d'une Gaussienne ( $\mathcal{F}(G_\sigma)$ ), est une Gaussienne:

$$\mathcal{F}(G_\sigma)(x) = \exp\left(-4\pi|x|^2\sigma\right). \quad (2.4)$$

Donc, en remplaçant dans (1.3) et en prenant la transformée de Fourier inverse de l'équation résultante dans le cas monodimensionnel où le paramètre  $\sigma$  est assimilé au temps  $t$ , on aboutit à l'équation suivante:

$$\frac{du}{dt} = \frac{\partial^2 u}{\partial x^2} \quad (2.5)$$

Cette équivalence reste évidemment valable dans le cas multidimensionnel. L'image filtrée par convolution avec des Gaussiennes  $u$  est donc solution de l'équation de diffusion linéaire, ou équation de la chaleur, complétée par la condition initiale  $u_0(x)$ :

$$\begin{cases} \frac{\partial u(x, t)}{\partial t} = \Delta u(x, t) \\ u(x, 0) = u_0(x) \end{cases} \quad (2.6)$$

Lorsque la condition initiale  $u_0(x)$  est une fonction bornée, ce système admet une solution unique :

$$u(t, x) = (G_t * u_0)(x) \quad \text{pour } t > 0 \quad (2.7)$$

---

1. On notera généralement dans ce chapitre par  $u_0$  le niveau de gris de l'image originale.

L'équivalence entre le filtrage de Gauss et l'équation de la chaleur est classique; elle a été introduite par Marr et Hildreth [79], ensuite développée par Witkin [117] puis par Koenderink [74]. Dans ces équations, on voit facilement que le terme  $t$  correspond bien à une échelle spatiale  $\sigma$  (1.1). Le filtrage (ou lissage) doit être fait durant un temps fini, égale au paramètre  $\sigma$ .

L'équation parabolique linéaire (1.6) permet une diffusion identique dans toutes les directions (diffusion isotrope). Ainsi, dans les zones homogènes d'une image, ce filtre permettra effectivement d'atténuer le bruit considéré comme étant un signal de haute fréquence, mais dans les zones présentant des discontinuités de niveau de gris, celles-ci seront aussi lissées. Ce filtre ne préserve donc pas les objets significatifs de l'image. De plus, en pratique, les objets significatifs d'une image filtrée par un filtre de Gauss donnent une impression de *floue* par rapport à l'image originale, ce qui rend la détermination des contours quasi impossible diminuant en conséquence l'intérêt visuel de ce processus de diffusion. Un seuillage à la fin du traitement est donc nécessaire, par exemple celui à base du gradient tel que:  $(x, y)$  est un point d'une structure représentative pour une échelle spatiale  $\sqrt{t}$  si et seulement si le Laplacien  $\Delta u(x, y, t)$  change de signe et le gradient  $|\nabla u(x, y, t)|$  est très grand.

A titre de compléments, plusieurs études concernant l'application de l'équation de la chaleur aux domaines du traitement des images ont été faites depuis [12, 119, 105, 54]. La partie suivante de ce paragraphe sera consacrée à des variantes du filtrage isotrope Gaussien. Celles-ci ne permettent pas de lisser l'image proprement dit, mais d'en extraire les informations utiles à un traitement postérieur.

### 2.2.2 Variantes du filtrage Gaussien

En analyse non locale du signal ou de l'image, l'endroit où le signal change de tendance est appelé "*bord*" [79]. Dans la théorie classique, les *bords* sont définis comme étant l'ensemble des points, où la norme du gradient du signal atteint un maximum local. L'ensemble des *bords* est donc contenu dans l'ensemble des points où le Laplacien change de signe. Utilisant ces propriétés, plusieurs améliorations des filtres basés sur l'équation de la chaleur ont été introduites pour palier les difficultés citées ci dessus. Parmi ces modèles on trouve :

- le filtre fondé sur les dérivées directionnelles de la Gaussienne qui a été proposé par Marr et Hildreth [79]. Il utilise le Laplacien du signal convolé avec une Gaussienne. Par l'utilisation de cette technique, les *bords* se situent là où le Laplacien s'annule (*zero – crossing*); ce qui permet à ce processus de vérifier le principe du maximum [67] et d'aboutir à des contours fermés. Malheureusement les recherches de zéros peuvent induire de fausses détections dans les zones trop lisses, ces courbes ne représentent donc pas les contours exacts.

- le filtre *passé – bas* de Canny [17] qui consiste à localiser les bords en utilisant les gradients directionnels de l'image régularisée par une Gaussienne. Il permet la localisation

des extrema du gradient et d'isoler ainsi les contours. Plus précisément, supposons que l'on veuille convoluer l'image avec  $G_n$ , la dérivée de la Gaussienne (1.1) dans une direction  $n$ . Le choix idéal de  $n$  est la direction normale au *bord* de l'objet à segmenter; ce processus nécessitera ainsi une connaissance *a priori* des contours. La meilleure estimation de  $n$  est la régularisée de la direction du gradient de la manière suivante:

$$n = \frac{\nabla(G * u)}{|\nabla G * u|} \quad (2.8)$$

Les frontières des objets significatifs sont situées là où  $|G_n * u|$ , la norme du gradient, atteint un maximum local.

Ce filtre améliore la détection et la localisation des contours, par rapport au Laplacien. Par contre, les contours obtenus ne sont généralement pas fermés. Le principal problème est donc de relier les points frontières entre eux pour former un véritable contour. Ce filtre malgré ses limites est considéré comme un outil efficace en élimination du bruit et en détection des *bords*.

Signalons enfin que, dans la lignée de ces méthodes fondées sur des variantes de l'équation de la chaleur, Illner et Neunzert [68] ont introduit une équation de diffusion dirigée de la manière suivante :

$$\begin{cases} \frac{\partial u}{\partial t} &= \tilde{u} \Delta u - u \Delta \tilde{u} \\ u(x, 0) &= u_0(x) \end{cases} \quad (2.9)$$

où  $\tilde{u}$  est une version lisse (*smooth*) de  $u$ , le niveau de gris de l'image. Le problème posé par ce filtre linéaire, étudié par la suite par Illner et Tie [69], réside dans le choix de la fonction  $\tilde{u}$  qui doit être connue à l'avance, ce qui revient à un lissage utilisant une diffusivité adaptative avec une connaissance *a priori* des structures représentatives de l'image.

## Conclusions

Il est bien connu que les *objets* à petite échelle donnent une détermination inexacte des bords. Cela reste vrai pour le filtre *pass - bas* de Canny.

En utilisant les filtres basés sur la diffusion linéaire, la taille d'un objet présentant une forte courbure diminue jusqu'à disparition totale de celui-ci; les fronts se déplacent avec une vitesse proportionnelle à la courbure. En pratique, les coins sont donc particulièrement attaqués.

Si le signal est bruité le gradient aura plusieurs maximums peu utiles, donc à éliminer. Ces oscillations fortes du gradient peuvent être dues à différents facteurs comme la présence de textures. Les frontières des différentes régions texturées ne sont apparentes

que si la valeur moyenne du signal est différente pour chaque texture.

La difficulté commune des filtres linéaires est le lissage excessif qui rend difficile le suivi des bords. On peut donc affirmer que toute amélioration de ces modèles linéaires doit être effectuée à l'intérieur de l'opérateur de diffusion lui-même, sacrifiant ainsi leur linéarité. Nous verrons comment ces difficultés peuvent être surmontées par l'utilisation de modèles non linéaires. Les travaux réalisés dans ce cadre feront l'objet du paragraphe suivant.

## 2.3 Filtres non linéaires

La motivation essentielle des modèles basés sur la diffusion non linéaire est la construction d'un opérateur de diffusion dépendant des propriétés locales de l'image. On peut distinguer deux types de filtres non linéaires:

- (i) Le filtrage isotrope non linéaire, utilisant une équation de diffusion non linéaire avec une diffusivité scalaire adaptée aux propriétés locales du signal.
- (ii) Le filtrage anisotrope non linéaire, utilisant une équation de diffusion non linéaire à diffusivité tensorielle et dont l'opérateur de diffusion s'adapte aux structures de l'image ce qui permet de contrôler les directions de diffusion.

Le filtrage par une diffusion isotrope non linéaire (i) est détaillé dans ce paragraphe et le deuxième type (ii) dans le paragraphe (1.7).

Le premier modèle non linéaire a été introduit par Perona et Malik en 1987 [89]. Ils ont proposé un filtre adaptatif (ou conditionnel), permettant d'atténuer la diffusion dans les régions à fort gradient, où des discontinuités potentiellement significatives peuvent se trouver, et de la maintenir dans les zones à faible gradient. Cela permet d'éviter le seuillage nécessaire pour les modèles linéaires à la fin du traitement. Ce modèle a permis une avancée très importante dans le domaine de l'application des EDP en traitement d'images. Mais, du fait que ce dernier peut être mal posé, plusieurs améliorations basées sur la régularisation (par moyenne locale ou convolution avec une Gaussienne) ont été introduites depuis. La plus importante est celle proposée par Catté et al [26] qui ont montré l'existence et l'unicité de la solution du modèle Perona-Malik régularisé. Notons aussi que dans ces modèles, le terme "anisotrope" n'est pas complètement justifié puisqu'ils utilisent une diffusivité scalaire. Ces modèles ne sont anisotropes que dans un repère local. Une régularisation spatiale utilisant explicitement une diffusivité tensorielle a été introduite, par la suite, par Weickert [108]. Ce filtre sera détaillé dans le paragraphe (1.7).

### 2.3.1 Le modèle de Perona et Malik

#### Le modèle de base [89]

Perona et Malik [91] ont proposé de remplacer l'équation de la chaleur par une équation non linéaire de type milieu poreux :

$$\begin{cases} \frac{\partial u}{\partial t} &= \operatorname{div}(g(\nabla u)\nabla u) \\ u(x, 0) &= u_0(x) \end{cases} \quad (2.10)$$

Dans cette équation  $g$  est une fonction régulière non croissante avec  $g(0) = 1$ ,  $g(s) \geq 0$  et  $\lim_{s \rightarrow \infty} g(s) = 0$ . L'idée est que le traitement obtenu par l'équation (1.10) est conditionnel en chaque point  $x$  de l'image ( $\Omega \subset \mathbb{R}^2$ ):

- Si  $\nabla u(x)$  est grand alors la diffusion s'atténue; par suite on a une localisation exacte des *bords*.
- Si  $\nabla u(x)$  est petit alors la diffusion tend à lisser encore plus au voisinage de  $x$ .

En pratique, le choix de la diffusivité  $g(\nabla u)$  correspond à un seuillage comparable à celui du gradient  $\nabla u(x)$ , généralement utilisé dans l'étape finale des filtres linéaires.

Pour les tests numériques de ce modèle [91], la fonction  $g$  utilisée est de la forme:

$$g(\nabla u) = \frac{1}{1 + \left(\frac{\|\nabla u\|}{\lambda}\right)^2} \quad \lambda > 0, \quad (2.11)$$

ou encore:

$$g(\nabla u) = \exp\left(-\left(\frac{\|\nabla u\|}{\lambda}\right)^2\right) \quad \lambda > 0. \quad (2.12)$$

On peut remarquer que ces deux expressions de la fonction  $g$  présentent la même approximation au premier ordre.

Perona et Malik [91] ont montré que ce modèle améliore la performance du filtre de Canny [17] à détecter les bords dans une image bruitée.

#### Formulation Variationnelle

On peut associer au modèle de Perona-Malik une fonctionnelle d'énergie donnée par:

$$E(u) = \frac{1}{2} \int_{\Omega} \int_0^t g(|\nabla u(x, s)|) ds dx \quad (2.13)$$

Supposons que  $u$  est un minimum de  $E$  et  $v$  une fonction test quelconque, la variation de  $E$  est alors donnée par :

$$\begin{aligned} \lim_{t \rightarrow 0} \frac{E(u + tv) - E(u)}{t} &= \int_{\Omega} g(|\nabla u|) \nabla u \nabla v dx \\ &= - \int_{\Omega} v \operatorname{div}(g(|\nabla u|) \nabla u) dx \end{aligned}$$

Donc l'équation de Perona-Malik peut être interprétée comme une méthode de descente du gradient de la fonctionnelle  $E$ :

$$\frac{\partial u}{\partial t} = -\nabla E \quad (2.14)$$

Récemment Osher et Rudin [86, 78] ont proposé un autre choix de la fonction  $g$  pour le modèle (1.10):

$$g(s) = \frac{1}{s} \text{ et } \sigma = 0$$

le modèle (1.10) s'écrit dans ce cas:

$$\frac{\partial u}{\partial t} = \operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right) \quad (2.15)$$

Cette équation correspond aussi à une méthode de descente (1.14), où  $E$ , la fonctionnelle associée, est donnée par:

$$E(u) = \int_{\Omega} |\nabla u(x)| dx. \quad (2.16)$$

Nous verrons dans le paragraphe (1.5) d'autres équations de diffusion basées sur un principe de minimisation d'énergie.

### Équivalence avec les Réseaux Neuronaux

Il est à noter qu'il existe une relation entre le modèle continu (1.10) et un modèle discret [34, 32], basé sur les réseaux de neurones. En effet, considérons le réseau neuronal suivant:

$$\frac{dX_i}{dt} = -AX_i + (B - X_i) \sum_{k=1}^n I_k G_{ki} - (X_i + D) \sum_{k=1}^n I_k E_{ki} \quad (2.17)$$

où les  $I_k$  sont des fonctions (*patterns*) d'entrée et les  $G_{ki}$  des fonctions Gaussiennes dont l'écart type dépend de la distance inter-cellulaire  $|k - i|$ .

Cohen et Grossberg [34] ont montré que  $u_i$ , activité ou potentiel de la  $i$ ème cellule obéit à l'équation de diffusion non linéaire suivante:

$$\frac{du_i}{dt} = -Hu_i + J_{i+1,i}(u_{i+1} - u_i) + J_{i-1,i}(u_{i-1} - u_i) + F_i \quad (2.18)$$

où la fonction d'entrée  $F_i = X_i/1 + \alpha S_i$ , le seuil  $S_i = \sum_{k=1}^n G_{ik} f(y_k)$  et les coefficients de diffusion  $J_{i+1,i}$  et  $J_{i-1,i}$  sont déterminés en fonction des contours de l'image à traiter de la manière suivante:

$$J_{i+1,i} = \frac{\lambda}{1 + \xi[S_{i+1} - , ]^* + \xi[S_i - , ]^*} \quad \text{et} \quad J_{i-1,i} = \frac{\lambda}{1 + \xi[S_{i-1} - , ]^* + \xi[S_i - , ]^*} \quad (2.19)$$

où  $\lambda$  est un seuil ( $\lambda > 0$ ). Le modèle (1.18) peut être considéré comme une discrétisation sous forme de réseau de neurones du modèle continu (1.10) [110]. Nous reverrons dans le chapitre 2 d'autres analogies entre des réseaux neuronaux et des opérateurs de diffusion linéaire.

Malheureusement, le modèle de Perona-Malik (1.10) présente des difficultés pratiques et théoriques. Si l'on suppose que le signal est bruité et que le bruit est très fort (non borné en théorie), aucune oscillation du gradient n'est visible; le bruit est donc gardé. Dans ce cas, le lissage conditionnel présenté par cette méthode n'est plus efficace. Cela reste vrai pour les images fortement texturées. Pour le choix de la fonction  $g$ , aucune théorie qui puisse montrer l'existence et l'unicité de la solution de (1.10) n'existe. Si la fonction  $xg(x)$  est décroissante sa discrétisation peut conduire à des algorithmes instables. Si l'on introduit les notations  $\tilde{x}$  (la coordonnée suivant la direction parallèle au gradient de  $u$ ) et  $x^\perp$  (celle suivant sa direction orthogonale)  $u = u(\tilde{x}, x^\perp, t)$ , l'équation de Perona-Malik s'écrit :

$$\frac{\partial u}{\partial t} = G'(\nabla u)u_{\tilde{x}\tilde{x}} + g(\nabla u)u_{x^\perp x^\perp} \quad (2.20)$$

où  $G(x) = xg(x)$ . Ce modèle agit donc comme une équation de la chaleur rétrograde au voisinage des très fortes discontinuités car  $u_{x^\perp x^\perp} = 0$  et  $G'(\nabla u) < 0$  si  $\|\nabla u\| > \lambda$  pour un choix de la conductivité  $g(\nabla u)$  correspondant à la formule (1.11), et si  $\|\nabla u\| > \lambda/\sqrt{2}$  pour le choix de la formule (1.12). Ces conditions rendent l'algorithme instable même si l'image de départ est assez lisse [84].

Pour résoudre le problème des instabilités, une solution proposée est d'utiliser des régularisations explicitement dans la fonction de conductivité afin de rendre le problème bien posé et d'obtenir des résultats stables et de meilleures qualités. Cette solution, que nous allons décrire dans le paragraphe suivant, a été indépendamment proposée dans [26] et [84]. Signalons aussi qu'une autre solution numérique a été proposée par Perona, Shiota et Malik [92].

Nous détaillons, dans la suite de ce paragraphe, quelques modèles fondés sur la régularisation du modèle Perona-Malik.

### 2.3.2 Versions régularisées du modèle Perona-Malik

Il est possible d'envisager deux sortes de régularisation: spatiale ou temporelle (et évidemment une combinaison des deux). Nous allons montrer, dans ce paragraphe, qu'une

telle régularisation rend effectivement le modèle Perona-Malik (1.10) bien posé et son approximation numérique stable.

### Le modèle de Catté, Lions, Morel et Coll [26]

Ce modèle consiste à remplacer la diffusivité dans le modèle de Perona et Malik par sa version régularisée. Plus précisément ce filtre est basé sur le système suivant:

$$\begin{cases} \frac{\partial u}{\partial t} &= \operatorname{div}(g(|\nabla G_\sigma * u|)\nabla u) \\ u(x, 0) &= u_0(x) \end{cases} \quad (2.21)$$

où

$$G_\sigma(x) = \frac{1}{4\pi\sigma^2} \exp\left(-\frac{|x|^2}{4\sigma^2}\right).$$

Le terme  $\nabla G_\sigma * u$  est le gradient de la solution (au temps  $t = \sigma$ ) de l'équation de la chaleur (1.6) avec  $u(., t)$  comme condition initiale,  $g(s)$  une fonction de la variable  $s$  qui tend vers 0 quand  $s$  tend vers l'infini et  $\nabla G_\sigma * u$  une somme pondérée de Gaussiennes.

En pratique, comme pour l'équation de la chaleur, on ne distingue pas les gradients forts qui viennent du bruit de ceux des objets significatifs. Pour discriminer cette confusion il est possible d'utiliser un paramètre d'échelle pour le lissage des zones homogènes de l'ordre de  $\sqrt{\sigma}$ . Le temps nécessaire avant de stopper le processus de diffusion est de l'ordre de  $\sigma$ .

L'existence, l'unicité et la régularité de la solution de (1.21) complétée par une condition de Neumann ont été démontrées [26].

Ce modèle, se comportant localement comme une *inversion* de l'équation de la chaleur, peut devenir instable. De plus, la présence d'un terme combinant le gradient et son régularisé dans l'opérateur de diffusion rend difficile de lui trouver une interprétation géométrique. Pour le traitement d'image contenant des segments où une partie peut être partiellement cachée (ex. des vaisseaux sanguins dans le cas de l'imagerie médicale), ce filtre se comporte comme un filtre linéaire sur les objets monodimensionnels; d'autre part, la diffusion s'annulant de part et d'autre des bords, il n'y a pas d'élimination de bruit [109]. Nous verrons dans le paragraphe (1.7) et dans le chapitre 2, deux modèles mieux adaptés au traitement d'images présentant des objets monodimensionnels.

### Autres régularisations

Notons qu'une régularisation spatio-temporelle a été introduite par Nitzberg et Shiota [84]. Dans le cas monodimensionnel ils ont proposé le modèle suivant:

$$\partial_t u = \partial_x \left( \frac{\partial_x u}{1 + v} \right) \quad (2.22)$$

$$\partial_t v = \omega \left( (\partial_x u - v) + \frac{\sigma^2}{2} \partial_x^2 v \right) \quad (2.23)$$

où  $u$  est le niveau de gris de l'image,  $\omega$  un réel positif et  $\sigma$  le paramètre d'échelle lié à la régularisation spatiale ( $\sigma > 0$ ). On remarque que lorsque  $\sigma \rightarrow 0$  et  $\omega \rightarrow \infty$ , ce modèle tend à se rapprocher du modèle isotrope non linéaire proposé par Perona et Malik (1.10). Le cas bidimensionnel, détaillé dans [83], donne lieu à un schéma particulièrement bien adapté à la restauration des points anguleux.

Signalons enfin que dans la lignée de ces régularisations du modèle de Perona-Malik, Whitaker et Pizer [115, 113] ont choisi de remplacer le paramètre d'échelle  $\sigma$  par un autre décroissant en temps. Par contre Li et Chen [77] ont proposé de faire décroître la constante  $\lambda$  dans la fonction  $g$ .

### 2.3.3 Filtre limiteur d'oscillation

Utilisant les techniques développées en analyse numérique des équations hyperboliques non linéaires, Rudin [95] a été le premier à introduire les filtres limiteurs d'oscillation dits *Shock Filters* pour la restauration des images. Ces filtres permettent de développer dans l'image restaurée des phénomènes analogues aux ondes de choc en mécanique des fluides [95].

Le premier filtre monodimensionnel est basé sur des modifications de l'équation de Burger de la manière suivante:

$$\begin{cases} \frac{\partial u}{\partial t} + F(u_{xx})|u_x| = 0 \\ u(x, 0) = u_0(x) \end{cases} \quad (2.24)$$

où  $F$  est une fonction telle que  $x F(x) \geq 0$ . Ce modèle développe des lignes de choc là où  $u_{xx}$  change de signe, c'est à dire sur les bords de l'image.

Une généralisation du modèle (1.24) dans le cas bidimensionnel a été introduite par Osher et Rudin, ils ont proposé le modèle suivant:

$$\begin{cases} \frac{\partial u}{\partial t} + |\nabla u| F(\nabla u^t H \nabla u) = 0 \\ u(x, 0) = u_0(x) \end{cases} \quad (2.25)$$

où  $F$  vérifie les mêmes conditions que dans le cas monodimensionnel,  $H$  est le Hessien de  $u$  et le terme  $\nabla u^t H \nabla u$  permet l'extraction des contours.

Pour renforcer les contrastes d'une image, Osher et Rudin [86] ont aussi proposé un schéma conservatif permettant d'inverser l'équation de la chaleur:

$$\begin{cases} \frac{\partial u}{\partial t} &= -|\nabla u|F(\Delta u) \\ u(x, 0) &= u_0(x) \end{cases} \quad (2.26)$$

Lorsque ( $F(x) = x$ ), ce modèle se comporte comme l'équation de la chaleur avec un temps  $t$  inversé, d'où le nom filtrage par inversion de l'équation de la chaleur.

Pour la discrétisation de cette équation, Osher et Rudin ont utilisé un schéma explicite monotone qui préserve la taille, la variation totale et la situation des extrema locaux. Si le signal est bruité, ce modèle devient peu efficace car les fluctuations de  $u_{xx}$  dues au bruit génèrent des "faux" chocs.

Pour maintenir la diffusion dans les zones où le niveau de bruit est important, Alvarez [6, 5] a présenté un processus fondé sur les filtres limiteurs de tension et sur la diffusion non linéaire. Reprenant le modèle monodimensionnel (1.24) introduit par Rudin, il a proposé le schéma suivant:

$$\begin{cases} \frac{\partial u}{\partial t} + F(G_\sigma * u_{xx}, G_\sigma * u_x)|u_x| &= 0 \\ u(x, 0) &= u_0(x) \end{cases} \quad (2.27)$$

où  $G_\sigma$  est la Gaussienne d'écart type  $\sigma$  donnée par la formule (1.1) et  $F$  satisfait la condition suivante:

$$F(x)x_1x_2 \geq 0 \quad \text{pour tout } x = (x_1, x_2) \in \mathbb{R}^2. \quad (2.28)$$

Le modèle (1.27) développe des chocs lors du passage à zéro du signal filtré, ce qui évite d'atténuer la diffusion dans les zones fortement bruitées.

Avant de passer aux filtres basés sur la courbure principale, signalons qu'une version bidimensionnelle de cette dernière équation (1.27) a été introduite par Alvarez et Mazorra [6]; cette généralisation au cas 2D a été effectuée de la manière suivante:

$$\frac{\partial u}{\partial t} = C\Upsilon(u) - u_\eta F(G_\sigma * u_{\eta\eta}, G_\sigma * u_\eta), \quad (2.29)$$

la notation  $\eta = \eta(x)$  est pour la direction du gradient  $\nabla u(x)$ ,  $G_\sigma$  reste la Gaussienne d'écart type  $\sigma$ ,  $C$  une constante strictement positive et  $F$  une fonction vérifiant la condition (1.28).

## 2.4 Filtres basés sur la notion de courbure et contours actifs

La motivation principale des filtres basés sur le déplacement de la courbure principale, dits de *mean curvature motion* (*MCM*), est la construction d'un opérateur de diffusion non linéaire capable de diffuser "plus" dans la direction parallèle aux objets significatifs et "moins" dans leurs directions perpendiculaires. C'est une autre approche qui va nous permettre d'englober les filtres définis précédemment dans un cadre plus large de filtres anisotropes.

### 2.4.1 Filtres basés sur la notion de courbure (*MCM*)

Si l'on introduit les notations  $\tilde{x}$  (la coordonnée suivant la direction parallèle au gradient de  $u$ ) et  $x^\perp$  (celle suivant sa direction orthogonale)  $u = u(\tilde{x}, x^\perp, t)$ , l'équation de la chaleur s'écrit sous la forme :

$$\frac{\partial u}{\partial t} = u_{x^\perp x^\perp} + u_{\tilde{x}\tilde{x}}. \quad (2.30)$$

Le premier terme permet donc le lissage suivant la direction des *bords*; par contre le deuxième diffuse suivant leur direction orthogonale. Il n'est donc pas difficile de voir que, pour ne conserver que la diffusion le long des contours, l'équation de la chaleur doit être remplacée par:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial (x^\perp)^2} \quad (2.31)$$

$$= \Delta u - \frac{D^2 u(\nabla u, \nabla u)}{|\nabla u|^2} \quad (2.32)$$

Une formulation de cette dernière équation sous forme de "quasi divergence" nous donne :

$$\frac{\partial u}{\partial t} = |\nabla u| \operatorname{div} \left( \frac{\nabla u}{|\nabla u|} \right) \quad (2.33)$$

Le *front* de diffusion est une ligne de niveau définie à l'instant  $t$  par :

$$Fr_t = \{(x, y) / u(x, y, t) = Cte\} \quad (2.34)$$

Cette courbe  $Fr_t$  peut être paramétrée par :

$$Fr_t = \{M(s, t) / u(M(s, t)) = Cte\} \quad (2.35)$$

La vitesse  $\frac{\partial \vec{M}}{\partial t}$  de déplacement du point  $M$  et  $\frac{\partial \vec{M}}{\partial s}$ , la direction de la tangente à  $Fr_t$  en  $M$  sont liées par la relation suivante :

$$\left\langle \frac{\partial \vec{M}}{\partial t}, \frac{\partial \vec{M}}{\partial s} \right\rangle = 0 \quad (2.36)$$

Une différenciation de (1.35) par rapport à  $t$  et  $s$  nous permet d'écrire:

$$\left\langle \frac{\partial \vec{M}}{\partial t}, \vec{\nabla} u \right\rangle + \frac{\partial u}{\partial t} = 0 \quad \text{et} \quad \left\langle \frac{\partial \vec{M}}{\partial s}, \vec{\nabla} u \right\rangle = 0 \quad (2.37)$$

(1.36) et (1.37) nous donnent:  $\frac{\partial \vec{M}}{\partial t} = \lambda \vec{\nabla} u$ . En remplaçant dans (1.37) et en utilisant (1.33) on obtient :

$$\frac{\partial \vec{M}}{\partial t} = \left[ -\frac{\vec{\nabla} u}{|\vec{\nabla} u|} \right] \operatorname{div} \left( \frac{\nabla u}{|\nabla u|} \right) \quad (2.38)$$

$$= \left[ -\frac{\vec{\nabla} u}{|\vec{\nabla} u|} \right] \operatorname{courbure}(u) \quad (2.39)$$

où  $\operatorname{courbure}(u)$ , la courbure en chaque point d'une ligne de niveaux, est donnée par :

$$\operatorname{courbure}(u) = \operatorname{div} \left( \frac{\nabla u}{|\nabla u|} \right), \quad (2.40)$$

Les *fronts* de diffusion se déplacent ainsi avec une vitesse proportionnelle à la courbure. Les courbes  $C$ , correspondant aux lignes de niveaux de  $u$ , évoluent donc selon l'équation suivante:

$$C_t(\cdot, t) = k(\cdot, t) \cdot \vec{N} \quad (2.41)$$

où  $k(x, t)$  est la courbure de  $C$  au point  $x$  et  $\vec{N}$  le vecteur normal intérieur à cette courbe. L'équation (1.33) peut être interprétée comme un filtre basé sur le déplacement de la courbure principale. Ce filtre est ainsi fondé sur le système suivant:

$$\begin{cases} \frac{\partial u}{\partial t} = |\nabla u| \operatorname{courbure}(u) \\ u(x, 0) = \tilde{u}_0(x) \end{cases} \quad (2.42)$$

où  $\tilde{u}_0$  est une version lisse de  $u_0$ , le niveau de gris de l'image à traiter. Cette EDP correspond à une diffusion anisotrope dans la direction des lignes de niveaux de l'image. Il est à noter que cette équation est identique à celle proposée par Osher et Sethian [88] pour l'évolution des courbes et à celle étudiée par Rudin, Osher et Fatemi [96] pour la restauration d'image.

On remarque que ce modèle ne comporte aucun élément permettant de stopper le processus de diffusion sur le bord des objets représentatifs.

En coordonnées cartésiennes  $(x, y)$ , l'équation (1.33) peut se réécrire comme suit:

$$\frac{\partial u}{\partial t} = \frac{u_y^2 u_{xx} - 2u_x u_y u_{xy} + u_x^2 u_{yy}}{u_x^2 + u_y^2}. \quad (2.43)$$

Dans un récent papier [4], Alvarez, Guichard, Lions et Morel ont généralisé l'équation précédente en utilisant une approche axiomatique de la théorie des filtres *pass - bas*.

Une autre version du modèle précédent et de celui de Perona et Malik a été introduite par Alvarez, Lions et Morel [5]. Ils ont présenté et étudié le modèle suivant :

$$\begin{cases} \frac{\partial u}{\partial t} &= g(|\nabla G_\sigma * u|)|\nabla u| \operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right), \\ u(x, 0) &= u_0(x) \end{cases} \quad (2.44)$$

Ce modèle fait partie de la classe des filtres sélectifs et dépend de deux paramètres: la fonction contraste qui correspond au seuillage du gradient  $|\nabla u|$  dans la théorie de Marr et Hilreth [79] et le paramètre d'échelle lié à l'écart type de la Gaussienne  $G$  qui est lié au détail minimal à garder dans l'image à traiter. Une interprétation des termes de cette équation est la suivante :

- $|\nabla u| \operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right) = \Delta u - \frac{D^2 u(\nabla u, \nabla u)}{|\nabla u|^2}$  permet de lisser le niveau de gris sur les deux faces des objets représentatifs, avec un minimum de lissage sur l'objet lui même. Ce terme de diffusion donné par (1.33) est anisotrope dans la direction parallèle aux contours.
- la fonction  $g$  étant non croissante en  $x$  et tendant vers 0 quand  $x$  tend vers l'infini,  $g(|\nabla G_\sigma * u|)$  contrôle la vitesse de diffusion et renforce les contrastes. Ainsi, dans les régions de l'image où le gradient est faible le terme  $g(|\nabla G_\sigma * u|)|\nabla u|$  est grand ce qui permet un maintien de la diffusion le long de la direction orthogonale au gradient par contre dans les régions à fort gradient ce terme est faible et la diffusion s'annule.

Comme pour le filtre de Catté et al, sur les segments, la diffusion disparaît, ce type de schéma non linéaire ne permet donc pas d'éliminer le bruit. Ce filtre n'est, par conséquent, pas approprié au pré-traitement d'images contenant des objets monodimensionnels. Une autre difficulté est que si l'image à restaurer correspond à une ellipse alors le mouvement par courbure, qui a tendance à transformer toute courbe fermée en cercles, délocalise les *bords*. Nous verrons dans le paragraphe (1.6.1) une approche morphologique qui permet de garder la forme elliptique de ces courbes de niveaux.

## 2.4.2 MCM et contours actifs

En parallèle avec l'application des EDP au traitement d'image, s'est développée une théorie de la déformation de courbes planes sous l'effet d'une équation de la chaleur

intrinsèque:

$$\begin{cases} C_t &= C_{\sigma\sigma} \\ C(., 0) &= C_0 \end{cases} \quad (2.45)$$

où  $C$  est une courbe fermée du plan affine Euclidien,  $\sigma$  une abscisse curviligne et  $C_0$  une courbe de départ.

Si l'on considère la courbe initiale comme l'isophote d'une image  $u$ , alors son évolution suivant (1.45) est la même que celle produite par l'équation (1.31) [71, 72, 100]. Sapiro et Tannenbaum [99] ont appliqué ces équations d'évolution affines et euclidiennes de courbes planes au lissage de courbes bruitées; ceci permet en particulier de préserver la surface ou le périmètre de la courbe initiale.

Signalons que l'espace d'échelle engendré par cette équation a été utilisée dans le domaine de la reconnaissance des formes par Faugeras et Keriven [53] et récemment généralisée au cas où  $\sigma$  est l'abscisse curviligne projective [50, 53].

Dans le cadre de l'application des EDP au contours déformables (ou contours actifs également connus sous le nom de "snakes") introduits par Wikin et Kass [31, 70], se basant sur les équations (1.31,1.45), Caselles et al [18] ont présenté le modèle géométrique suivant:

$$\begin{cases} \frac{\partial v}{\partial t} &= g(x) |\nabla v| \left( \operatorname{div} \left( \frac{\nabla v}{|\nabla v|} \right) + \nu \right) \\ v(0, x) &= v_0(x) \\ g(x) &= \frac{1}{1 + |\nabla(G_\sigma * u_0)|^2} \end{cases} \quad (2.46)$$

où  $\nu$  est une constante positive,  $G_\sigma * u_0$  étant la convolution de l'image contenant l'objet  $O$  à segmenter, avec la Gaussienne d'écart type  $\sigma$  (1.1), la donnée initiale  $v_0$  est une version lisse de la fonction  $1 - \zeta_C$  et  $\zeta_C$  la fonction caractéristique d'un ensemble  $C$  contenant l'objet  $O$ .

Pour une fonction régularisée  $v$  donnée, la  $k$ -courbe de niveau de  $v$  est donnée par:

$$\partial C(t) = \{(x, y) \in \mathbb{R}^2 : v(x, y, t) = k\}, \quad (2.47)$$

et la courbure en un point  $x$  de la courbe  $\partial C$  est donnée par: courbure( $x$ ) =  $\operatorname{div} \left( \frac{\nabla v}{|\nabla v|} \right)$ .

Dans l'EDP (1.46), une ligne de niveau de l'image  $v$  est une représentation implicite d'une courbe  $C$ , appelée à évoluer dans le temps au sein de l'image  $u_0$  afin de converger

vers les frontières de l'objet  $O$  à segmenter. Une interprétation géométrique de ce modèle, est alors la suivante:

- $\nu$  est un terme de correction qui permet d'assurer la positivité de  $(\operatorname{div}\left(\frac{\nabla v}{|\nabla v|}\right) + \nu)$ , ainsi le niveau de gris en un point  $x$  de la courbe  $\partial C$  croît proportionnellement à la courbure de  $\partial C$  en ce point et  $\partial C$  tend à se rapprocher des bords de l'objet  $O$  lorsque sa courbure devient négative ou nulle.
- La convolution de  $u_0$  avec une Gaussienne permet en effet d'éliminer le bruit lors du mouvement de  $\partial C$ .
- $g(x)$  contrôle la vitesse de déplacement de  $\partial C$ : quand  $\partial C$  est proche des bords de l'objet  $O$ ,  $|\nabla(G_\sigma * u_0)|$  devient très grand et  $\partial C$  ne bouge plus. Ce coefficient permet donc de ralentir l'évolution de  $v$  au voisinage de  $O$ .

Dans le modèles *MCM*, Caselles Kimel et Sapiro [22] ont introduit le concept du contour actif géodésique. Ceci consiste à réécrire l'équation (1.46) sous la forme suivante:

$$\begin{cases} \frac{\partial v}{\partial t} &= g(x) |\nabla v| \operatorname{div}\left(\frac{\nabla v}{|\nabla v|}\right) + \nabla g \cdot \nabla v \\ v(0, x) &= v_0(x) \end{cases} \quad (2.48)$$

dans cette EDP le nouveau terme  $\nabla g \cdot \nabla v$ , permet de pousser encore davantage le contour vers les bords des objets à segmenter et évite le problème du choix de la constante  $\nu$ .

Récemment, utilisant les propriétés géométriques du modèle (1.46), Caselles et Coll [19] ont proposé l'algorithme suivant:

$$\begin{cases} \frac{\partial v}{\partial t} &= g(t, x) |\nabla v| \left( \operatorname{div}\left(\frac{\nabla v}{|\nabla v|}\right) + \nu \right) + (1 - g(t, x)) V \cdot \nabla v \\ v(0, x) &= v_0(x) \end{cases} \quad (2.49)$$

où  $V = (V_1, V_2)$  est la vitesse; et pour  $k$  un seuil donné et  $\varepsilon > 0$ , la fonction  $g$  a été définie par:

$$g(t, x) = \begin{cases} 1 & \text{si } |G_\sigma * u_0| < k - \varepsilon \\ 0 & \text{si } |G_\sigma * u_0| > k + \varepsilon \end{cases} \quad (2.50)$$

Dans cet algorithme, le terme  $(1 - g(t, x))$  est nul au voisinage externe des objet  $O_i$  à segmenter et prend la valeur 1 sur les bords de ces objets. La multiplication par  $g$  permet de réduire l'estimation de la vitesse  $V$  au voisinage de ces bords. Les autres termes ont la même interprétation géométrique que dans (1.46).

L'existence et l'unicité de la solution de viscosité de (1.49) complétée par une condition de Neumann ont été démontrées [19].

## 2.5 Filtres par minimisation d'énergie

Comme on l'a vu précédemment, les solutions des équations de réaction-diffusion isotrope linéaire et non linéaire peuvent être interprétées comme minimisantes de fonctionnelles d'énergie. Inversement les approches par minimisation d'énergie peuvent donner accès, via les équations d'Euler associées, à de nouveaux modèles de diffusion.

### 2.5.1 Le modèle de Mumford-Shah

Pour segmenter une image  $u_0 : \Omega \rightarrow \mathbb{R}$ , se basant sur les modèles discrets introduits par S. et D. Geman [56], Mumford et Shah [81, 82] ont proposé de minimiser la fonctionnelle suivante:

$$E_{u_0}(u, w) = \beta \int_{\Omega} (u - u_0)^2 dx + \int_{\Omega \setminus K} |\nabla u|^2 dx + \alpha |K| \quad (2.51)$$

où  $K$  est l'ensemble des *bords* et  $|K|$  sa distance de Hausdorff,  $\alpha$  et  $\beta$  étant deux constantes positives.

Ce modèle consiste à approcher l'image initiale  $u_0$  définie sur un domaine  $\Omega \subset \mathbb{R}^2$  et à valeurs dans  $[0, 1]$  par une fonction  $u$  présentant des discontinuités dans un sous ensemble  $K$ . Ce filtre combine donc en un même traitement, lissage de l'image et détection des bords.

Dans le livre de Morel et Solimini [80], on trouve une étude très détaillée du modèle de Mumford-Shah et de celle de ses algorithmes dérivés.

Une autre classe de modèles basés sur la minimisation de fonctionnelles a été introduite en approchant la discontinuité sur le sous ensemble  $K$ , par une fonction régulière  $\tilde{u}$ , définie par :

$$\tilde{u}(x) = \begin{cases} 0 & \text{le long des bords de } u \\ 1 & \text{ailleurs} \end{cases} \quad (2.52)$$

En introduisant un paramètre  $c$ , lié à la longueur des *bords*, il est naturel de chercher à minimiser la quantité suivante :

$$\tilde{E}_{u_0}(u, \tilde{u}) = \int_{\Omega} \left( \beta(u - u_0)^2 + \tilde{u}^2 \cdot |\nabla u|^2 + \alpha(c|\nabla \tilde{u}|^2 + \frac{(1 - \tilde{u})^2}{4c}) \right) dx \quad (2.53)$$

La minimisation de cette fonctionnelle  $\tilde{E}_{u_0}$  correspond aux équations de descente du gradient:

$$\begin{cases} \frac{\partial u}{\partial t} = \operatorname{div}(\tilde{u}^2 \nabla u) + \beta(u_0 - u), \\ \frac{\partial \tilde{u}}{\partial t} = c\Delta \tilde{u} - \frac{2\tilde{u}}{\alpha} |\nabla u|^2 - \frac{(1 - \tilde{u})}{2c} \end{cases} \quad (2.54)$$

La fonctionnelle  $\tilde{E}$  n'est pas convexe; l'algorithme de descente du gradient ne converge généralement pas vers le minimum global. Mumford et Shah ont conjecturé [82] que la fonctionnelle  $E_{u_0}$  admet un minimum  $(u, K)$  lorsque  $K$  est un ensemble fini de courbes  $C^1$ .

Récemment, J. Shah [102] a proposé de minimiser la fonctionnelle suivante:

$$E(u, K) = \int \int_{\Omega \setminus K} \|\nabla u\| dx dy + \frac{\beta}{\alpha} \int \int_{\Omega} |u - u_0| dx dy + \int_K \frac{J_u}{1 + \alpha J_u} ds \quad (2.55)$$

où  $\Omega$  est un sous ensemble ouvert borné de  $\mathbb{R}^2$ ,  $K$  une courbe segmentant  $\Omega$  et  $u$  une version lisse de l'image ( $\subset \mathbb{R}^2 \setminus K$ ),  $J_u$  le saut de  $u$  à travers  $K$  ( $J_u = |u^+ - u^-|$  et les notations  $+$  et  $-$  correspondent aux deux faces de  $K$ ). Les poids  $\alpha$  et  $\beta$  sont attribués à chaque point du bord en fonction du contraste.

La fonctionnelle proposée par Shah pour l'implémentation de la méthode de descente du gradient, est donnée par:

$$E_\rho(u, v) = \int \int_{\Omega} \left[ \alpha(1 - v^2) \|\nabla u\| + \beta |u - u_0| + \frac{\rho}{2} \|\nabla v\|^2 + \frac{v^2}{2\rho} \right] dx dy \quad (2.56)$$

où, pour reprendre les notations de (1.53),  $v = 1 - \tilde{u}$  (1.52). L'équation de descente du gradient correspondante s'écrit dans ce cas:

$$\left\{ \begin{array}{l} \frac{\partial u}{\partial t} = -2\nabla v \cdot \nabla u + (1 - v) \|\nabla u\| \text{courbure}(u) - \frac{\beta}{\alpha(1 - v)} \|\nabla u\| \frac{(u - u_0)}{|u - u_0|} \\ \frac{\partial v}{\partial t} = \nabla^2 v - \frac{v}{\rho^2} + \frac{2\alpha}{\rho} (1 - v) \|\nabla u\| \\ \frac{\partial u}{\partial n} |_{\partial\Omega} = 0 \quad ; \quad \frac{\partial v}{\partial n} |_{\partial\Omega} = 0 \end{array} \right. \quad (2.57)$$

Ces modèles résultent du même type d'approche car les bords sont représentés par une fonction qui vaut essentiellement 1 ou 0. La différence réside essentiellement dans le fait que les normes  $L^2$  sont remplacées par les normes  $L^1$ , qui permettent des solutions à variations bornées et discontinues. Une interprétation des termes du système (1.57) est la suivante:

- Le terme  $2\nabla v \cdot \frac{\nabla u}{\|\nabla u\|} - (1 - v) \text{courbure}(u) + \frac{\beta}{\alpha(1 - v)} \frac{(u - u_0)}{|u - u_0|}$  représente la vitesse de déplacement des courbes de niveaux de l'image lissée  $u$ . Le premier terme est l'advection permettant de tirer les courbes de niveaux à travers les grandes valeurs de  $v$ , le second est un terme de lissage [4] et le dernier permet le transfert des courbes de niveaux avec une vitesse  $\pm \beta/\alpha(1 - v)$ .
- Dans ce modèle,  $u$  développe des chocs, étant donné que l'évolution est parabolique le long des courbes de niveaux de  $u$ , et hyperbolique suivant sa direction normale.

- dans le cas monodimensionnel, le terme courbure( $u$ ) disparaît et l'équation d'évolution de  $u$  devient hyperbolique gouvernée par une équation d'advection en  $v$ , d'où un renforcement des contrastes .

La généralisation de ce modèle dans le cas où l'image  $u_0$  est un ensemble de fonctions:  $u_0 = \{u_0^1, \dots, u_0^m\}$ , est proposée dans [103]. L'avantage, par rapport aux filtres précédents, est que ce modèle n'utilise en principe qu'un seul paramètre et fournit des résultats comparables à ceux obtenus par les meilleurs d'entre eux. Il semble cependant que ce modèle n'ait à ce jour pas donné lieu à des résultats théoriques satisfaisants [103].

## 2.5.2 Le modèle de Nordström

Nordström [85] propose de minimiser la fonctionnelle suivante :

$$E_{u_0}(u, w) = \int_{\Omega} \left( \beta \cdot (u - u_0)^2 + w \cdot |\nabla u|^2 + \lambda^2 \cdot (w - \ln w) \right) dx. \quad (2.58)$$

où  $w : \Omega \rightarrow [0, 1]$  avec  $w \approx 1$  à l'intérieur d'une zone homogène et est nulle sur les bords,  $\alpha$  et  $\beta$  étant deux constantes positives.

L'équation d'Euler correspondante est donnée par :

$$\beta \cdot (u - u_0)^2 - \operatorname{div}(w \nabla u) = 0, \quad (2.59)$$

$$\lambda^2 \cdot \left(1 - \frac{1}{w}\right) + |\nabla u|^2 = 0, \quad (2.60)$$

L'équation (1.60) nous permet d'écrire  $w$  sous la même forme que la fonction  $g$  du modèle de Perona-Malik:

$$w = \frac{1}{1 + \frac{|\nabla u|^2}{\lambda^2}} \quad (2.61)$$

La méthode du gradient correspondant à la minimisation de (1.59) conduit à l'équation de réaction-diffusion suivante :

$$\begin{cases} \frac{\partial u}{\partial t} &= \operatorname{div}(g(|\nabla u|^2) \nabla u) + \beta \cdot (u_0 - u) \\ u(x, 0) &= u_0(x) \end{cases} \quad (2.62)$$

Cette équation combine le terme de diffusion de Perona-Malik et un terme de rappel vers l'image initiale, dont le but est de rendre le traitement moins sensible au choix du temps d'arrêt.

D'autre part, en plus du choix des paramètres  $\beta$  et  $\lambda$  qui doivent être appropriés, la fonctionnelle donnée par (1.58) n'étant pas convexe, la méthode peut converger vers un minimum local alors que la méthode de résolution de l'équation (1.54) ne converge pas nécessairement vers un minimum global. C'est d'ailleurs une difficulté assez générale qui concerne le filtrage par minimisation d'énergie.

## 2.6 Approche morphologique et Analyse multi-échelle

### 2.6.1 Une approche morphologique

Selon le principe morphologique, le niveau de gris d'une image est arbitraire et seul l'ordonnance locale des niveaux de gris a un sens [3]. En d'autres termes, dans une image, l'ordre des niveaux de gris constitue une information significative. De plus, une image se décompose en ses courbes de niveau, définies comme les bords des régions où le niveau de gris dépasse une certaine valeur. Il est donc facile de reconstruire une image à partir de lignes de niveaux. Alvarez et al [2, 4] ont proposé l'EDP suivante:

$$\frac{\partial u}{\partial t} = |\nabla u| \left( \operatorname{div} \left( \frac{\nabla u}{|\nabla u|} \right) \right)^{\frac{1}{3}} \quad (2.63)$$

Une interprétation de ce modèle, faisant le lien avec une méthode de lissage itératif et affine des courbes proposée indépendamment par Sapiro et Tannenbaum [99], est donnée dans [4].

Dans un récent papier, Caselles et al [20] ont proposé une autre version basée sur l'équation suivante:

$$\begin{cases} \frac{\partial u}{\partial t} &= \mathbb{1}_Z(x, y) |\nabla u| \left( \operatorname{div} \left( \frac{\nabla u}{|\nabla u|} \right) \right)^{\frac{1}{3}} \\ u(x, 0) &= u_0(x) \end{cases} \quad (2.64)$$

Une version bien posée de (1.64) est obtenue en remplaçant  $\mathbb{1}_Z$  par sa version régularisée  $\mathbb{1}_{Z,\varepsilon}$  de la manière suivante:

$$\mathbb{1}_{Z,\varepsilon}(x, y) = \begin{cases} 1 & \text{si } d((x, y), Z) > \varepsilon \\ 0 & \text{si } d((x, y), Z) < \frac{\varepsilon}{2} \end{cases} \quad (2.65)$$

L'interprétation de l'équation (1.64) est essentiellement la même que pour (1.63). La seule différence est l'altération de la vitesse des lignes de niveau de  $u$  dans un  $\varepsilon$ -voisinage de  $Z$ .

Puisque le temps d'arrêt est lié aux échelles, les contours à grandes échelles subissent un grand déplacement ce qui rend difficile l'application de ces modèles.

Donnons maintenant un dernier point de vue sur le filtrage donné par une analyse multi-échelle.

## 2.6.2 Une analyse multi-échelle

L'analyse multi-échelle d'une image consiste en la génération d'une suite d'images. Chaque terme  $u(\cdot, \lambda)$  de cette suite est une version filtrée à une échelle  $\lambda$  de l'image originale  $u_0$ , ce qui revient à l'application d'un ensemble d'opérateurs  $T_\lambda$  qui, appliqués à une image  $u_0$  conduisent à une séquence d'images  $u(\cdot, \lambda) = T_\lambda(u_0)$ .

Le problème du choix de l'opérateur  $T_\lambda$  a été étudié par l'axiomatisation de l'analyse multi-échelle qui permet d'obtenir l'unicité de cet opérateur. Il est à noter que cette axiomatisation a été largement développée par Alvarez, Guichard, Lions et Morel [2, 3, 1, 7].

Si l'on note par  $u_\lambda$  cette version lissée de  $u_0$ , le niveau de gris de l'image originale, et par  $K_\lambda$  l'ensemble des *bords* présents dans cette image, les principaux axiomes que doit vérifier cette analyse multi-échelles sont les suivants:

- (i) Fidélité:  $\lim_{\lambda \rightarrow 0} u_\lambda = u_0$
- (ii) Causalité: aucun faux détail ne doit être produit au cours de la diminution de la résolution ni lors du passage d'une petite à une grande échelle;  $u_\lambda$  dépend seulement de  $u_{\lambda'}$  pour  $\lambda > \lambda'$ . Cette propriété se formule axiomatiquement par: étant donné  $\lambda$  et  $\lambda' \geq 0$ , il existe un opérateur de transition noté  $T_{\lambda, \lambda'}$  tel que:

$$T_{\lambda'}(u_0) = T_{\lambda, \lambda'} \circ T_\lambda(u_0) \quad \text{et} \quad T_{\lambda, \lambda}(u_0) = u_0$$

- (iii) Causalité forte:  $K_\lambda \subset K_{\lambda'}$  pour  $\lambda > \lambda'$

Cette analyse est un point de vue général dans lequel s'inscrit le filtre Gaussien, entre autre.

Le filtre Gaussien vérifie les deux propriétés (i) et (ii) lorsque  $\lambda$  est le paramètre lié à l'échelle spatiale:  $u_\lambda = u(\cdot, \lambda)$ . Divers travaux se sont intéressés à la vérification de la troisième propriété [117].

## 2.7 Diffusion tensorielle

Toutes les méthodes citées jusqu'à présent utilisent une diffusivité scalaire. Dans ce paragraphe nous présenterons quelques modèles basés sur l'équation de la diffusion à conductivité tensorielle, appelée diffusion anisotrope.

### Le modèle de Cottet et Germain [41]

Les deux traitements réalisés par les filtres basés sur les équations de réaction-diffusion sont d'une part le filtrage du bruit et d'autre part le renforcement des contrastes du signal. Constatant que dans le cas isotrope ces modèles ne permettaient pas de conserver

les objets présentant une courbure très forte, Cottet et Germain [41] ont introduit un opérateur de diffusion anisotrope pour un lissage sélectif en plus d'un terme de réaction pour renforcer les contrastes de l'image. Cet opérateur permet ainsi de contrôler les directions de diffusion.

Soient  $\Omega$  le domaine  $[0, 1]^2$ ,  $\Gamma = \partial\Omega$  sa frontière et  $\varepsilon$  un réel positif. Notons par  $\zeta$  la fonction indicatrice de support, la boule de centre  $O$  et de rayon  $\varepsilon$ ; on peut ainsi définir la fonction cut-off  $\zeta_\varepsilon$  par :

$$\zeta_\varepsilon(x) = \varepsilon^{-2} \zeta\left(\frac{x}{\varepsilon}\right). \quad (2.66)$$

Une version régularisée  $u_\varepsilon$  de  $u$  le niveau de gris de l'image est obtenue par convolution de  $\bar{u}$  (prolongement de  $u$  sur  $\mathbb{R}^2$ ) avec une fonction cut-off:

$$u_\varepsilon(x) = (\bar{u} * \zeta_\varepsilon)(x) = \int_{\mathbb{R}^2} \zeta_\varepsilon(x - y) \bar{u}(y) dy \quad (2.67)$$

Dans ce modèle, l'opérateur de diffusion est défini comme étant la projection orthogonale sur l'orthogonale au gradient de  $u_\varepsilon$ :

$$A_\varepsilon(u) = \frac{1}{|\nabla u_\varepsilon|^2 + \varepsilon^2} \begin{pmatrix} (\partial_x u_\varepsilon)^2 & -\partial_x u_\varepsilon \partial_y u_\varepsilon \\ -\partial_x u_\varepsilon \partial_y u_\varepsilon & (\partial_y u_\varepsilon)^2 \end{pmatrix}$$

L'idée est que l'opérateur de diffusion maîtrise les endroits où la variation de la courbure des *bords* n'est pas très importante. Plus précisément ce modèle est basé sur l'équation de réaction-diffusion anisotrope suivante:

$$\begin{cases} \partial_t u - \varepsilon^2 \operatorname{div}([A_\varepsilon(u)][\nabla u]) = f(u) & \text{dans } \Omega \\ u(\cdot, 0) = u_0 & \text{dans } \Omega \\ u = 0 & \text{sur } \Gamma \end{cases} \quad (2.68)$$

En utilisant ce filtre, l'image traitée est obtenue sur les états asymptotiques de sa solution, le problème du choix d'un temps adéquate pour stopper la diffusion est donc surmonté.

Une analyse du comportement du modèle (1.68) est la suivante :

- En un point où l'image est non bruitée, les directions  $A_\varepsilon(u)[\nabla u]$  et  $\nabla u$  sont approximativement parallèles donc  $[A_\varepsilon(u)][\nabla u] \approx 0$ . L'équation (1.68) se réduit donc à une simple équation de réaction; le lissage est alors supprimé et remplacé par un renforcement des contrastes.
- En un point où l'image est bruitée, les directions ne sont pas corrélées:  $A_\varepsilon(u)\nabla u \approx \nabla u$  d'où  $\operatorname{div}([A_\varepsilon(u)][\nabla u]) \approx \Delta u$ . La diffusion se comporte comme un filtre Gaussien d'où un maintien du lissage.

Le terme de réaction  $f$  a été choisi de classe  $C^1$  satisfaisant :

$$f(1) = 0 \quad \text{et} \quad x f(x) > 0 \quad \text{pour} \quad x \neq 0$$

Le paramètre  $\varepsilon$  détermine la taille minimale de l'objet qu'on veut garder dans l'image originale. C'est donc une constante de régularisation spatiale permettant à ce modèle d'éliminer l'information présentant des variations spatiales sur des échelles inférieures à  $\varepsilon$ .

L'attractivité des états asymptotiques non triviaux et l'existence de solution de ce modèle, dans le cas discret, ont été démontrées [41].

La difficulté posée par ce filtre est liée essentiellement au choix du terme de réaction, qui suppose une connaissance a priori du niveau de gris des objets significatifs. Nous introduirons dans les chapitres suivants un nouveau modèle de diffusion anisotrope avec une régularisation temporelle dans le but de surmonter ces difficultés.

### Le modèle de Weickert [110]

Weickert a proposé d'utiliser une diffusion tensorielle pour régulariser le modèle de Perona-Malik [107, 108, 109, 110], comme alternative au modèle de Catté et al [26]. Dans ce modèle le paramètre  $\lambda$ , donné dans (1.11), dépend de la direction du gradient  $\nabla u$ . Plus précisément celui-ci est basé sur l'équation de diffusion anisotrope suivante :

$$\begin{cases} \partial_t u = \nabla \cdot (D(\nabla u_\sigma) \nabla u) & \text{dans } \Omega \\ u(., 0) = u_0 & \text{dans } \Omega \\ \langle D(\nabla u_\sigma) \cdot \nabla u, n \rangle = 0 & \text{sur } \partial \Omega \end{cases} \quad (2.69)$$

où  $n$  est un vecteur normal,  $\Omega$  un domaine rectangulaire de  $\mathbb{R}^2$  et  $\partial \Omega$  sa frontière. La notation  $\langle, \rangle$  représente le produit scalaire Euclidien de  $\mathbb{R}^2$ ;  $u_\sigma$  est défini par :

$$u_\sigma(x, t) = (G_\sigma * \tilde{u}(., t))(x) \quad \text{avec} \quad \sigma > 0 \quad (2.70)$$

où  $\tilde{u}$  est une extension de  $u$ , le niveau de gris de l'image, de  $\Omega$  dans  $\mathbb{R}^2$ . La convolution avec une Gaussienne d'écart type  $\sigma$  rend le filtre insensible au bruit correspondant aux échelles inférieures à  $\sigma$ .

L'opérateur de diffusion  $D$  est symétrique, défini, positif,  $(v_1, v_2)$  étant une base orthonormale de ses vecteurs propres :

$$v_1 \parallel \nabla u_\sigma \quad v_2 \perp \nabla u_\sigma \quad (2.71)$$

Les deux valeurs propres correspondantes  $(\lambda_1, \lambda_2)$  sont données par la formule suivante:

$$\lambda_1(\nabla G_\sigma * u) = g(|\nabla G_\sigma * u|^2) \quad (2.72)$$

$$\lambda_2(\nabla G_\sigma * u) = 1. \quad (2.73)$$

L'existence, l'unicité et la régularité de la solution de ce modèle ont été démontrées [110] (c'est une extension de la démonstration de Catté et al [26]).

Récemment, Weickert [111] a proposé de choisir comme valeurs propres :

$$\lambda_1(\nabla G_\sigma * u) = \begin{cases} 1 & \text{si } |\nabla u_\sigma| = 0 \\ 1 - \exp\left(\frac{-C}{|\nabla u_\sigma|^{2m}}\right) & \text{si non} \end{cases} \quad (2.74)$$

$$\lambda_2(\nabla G_\sigma * u) = 1. \quad (2.75)$$

où  $m \in \mathbb{N}$  et  $C$  est une constante positive.

Ainsi, sur les objets significatifs  $\lambda_1$  décroît rapidement favorisant la diffusion de part et d'autre de ces objets sur la diffusion à l'intérieur de l'objet lui même.

L'avantage par rapport au filtre conditionnel de Catté et al est que ce modèle lisse mieux les images bruitée et renforce les contrastes en diffusant sélectivement. Il se comporte ainsi comme isotrope dans les zones homogènes, et comme anisotrope le long des objets significatifs (*edges*). Lorsque  $\sigma \rightarrow 0$ , le modèle (1.69) tend à se rapprocher du modèle isotrope non linéaire proposé par Perona et Malik (1.10).

Ce modèle, comme celui de Catté et al, est strictement et uniformément (par rapport au temps) dissipatif et par conséquent ne donne que des états asymptotiques triviaux. Le problème du choix du temps approprié pour stopper le processus de diffusion n'est donc pas surmonté.

## 2.8 Résolutions Numériques

Puisque l'image a été définie comme une grille fixe, alors, en analyse numérique des modèles basés sur les EDP, les schémas explicites de différence finie sont les mieux adaptés. De plus, de part leur caractère local, ils sont faciles à utiliser sur des machines parallèles. Cohignac et al [35] ont proposé de modifier l'approximation spatiale des dérivées afin d'obtenir des propriétés de stabilité. Une approximation sous forme de réseaux de neurones a été introduite par Cottet [39, 40], elle a la particularité d'avoir des images filtrées sur les asymptotiques en temps et une stabilité au sens de Lyapunov. Pour une comparaisons entre différentes méthodes on pourra se référer à [35].

Enfin on ne peut pas terminer ce chapitre sans mentionner que plusieurs des modèles cités plus haut ont été adaptés pour l'analyse d'images tridimensionnelles (3-D)

[52, 51, 24].

Dans le paragraphe suivant, nous donnons quelques exemples concernant les domaines d'applications des modèles introduits plus haut.

## 2.9 Domaines d'applications

Les modèles d'équations de diffusion linéaire, non linéaire ou anisotrope sont généralement appliqués au pré-traitement des images, comme étape avant la segmentation [90, 50, 101, 104], ou d'une manière plus générale, celle de l'interprétation des images [80, 98]. Entre autres domaines d'application, on trouve évidemment le domaine médical: l'Imagerie par Résonance magnétique (IRM) [41, 40, 57, 112, 8, 14, 73] et les images échographiques [30, 16, 36] ainsi que l'imagerie cellulaire. L'imagerie satellite en météorologie et en cartographie apparaît aussi comme un champs d'application tout comme l'imagerie stratégique (e.g. détection et reconnaissance de forme).

Dans le chapitre 4, nous traiterons plusieurs de ces applications, notamment des images réelles, qui représentent une variété assez importante de problèmes particuliers. De plus, nous verrons que les filtres basés sur la diffusion anisotrope [40] sont les mieux adaptés au lissage d'image présentant des objets monodimensionnels (vaisseaux sanguins dans le cas des images médicales).

## 2.10 Conclusions et Discussion

On a vu que les filtres linéaires, outils efficaces pour le débruitage, délocalisent les objets significatifs et ne permettent pas le renforcement des contrastes. Ils rendent donc difficile la segmentation des objets.

Les filtres non linéaires dépendant des propriétés locales des images ont permis une avancée significative. Leur développement a été accompagné de la mise en oeuvre de techniques sophistiquées de schémas numériques. Néanmoins, certaines difficultés pratiques subsistent. Une de ces difficultés est le choix d'un temps nécessaire et suffisant pour le lissage des images. Les autres filtres anisotropes basés sur la minimisation de fonctionnelles qui ne sont pas toujours convexes, d'où une absence de convergence vers un minimum global.

Après avoir présenté différents modèles, ainsi que les idées de base de l'application des EDP pour le traitement d'images et les difficultés pratiques et théorique de chacun de ces filtres, nous présenterons dans le chapitre suivant un nouveau modèle de diffusion anisotrope capable d'éviter beaucoup de ces complications.



# Chapitre 3

## Réseaux de Neurones et Diffusion Anisotrope

### 3.1 Introduction

UN réseau de neurones est une description, en termes de variables simplifiées, d'un système neurobiologique. Cette description comporte deux classes de variables: l'une représente l'activité des cellules nerveuses (ou neurones), l'autre décrit les connexions (ou synapses) entre ces cellules nerveuses. Un modèle complet de réseaux de neurones adaptatif contient donc en général deux équations d'évolution, une pour chaque variable.

On dit qu'un réseau de neurones s'adapte ou subit un apprentissage, s'il développe les deux traitements simultanément. Il y a donc interaction entre les deux processus. L'un des apprentissages les plus populaires est l'apprentissage Hebbien [61]. Il consiste à lier les forces des connexions des neurones à la corrélation de leurs activités.

En 1943, Mc Culloch et Pitts ont proposé un modèle de neurones formels organisés dans un réseau interconnecté et effectuant de simples opérations. Ce modèle a été à la base de nombreuses études. Mais le modèle qui a suscité le plus d'intérêt est celui proposé, par la suite, par J.J Hopfield [65]. Il s'agit d'un réseau de neurones dont l'architecture est semblable à celle proposée par Mc Culloch et Pitts avec une matrice synaptique symétrique à diagonale nulle, et dont l'activation (fonction de transfert) est à seuil, ou sigmoïdale. Ce modèle possède par ailleurs des analogies avec ceux étudiés dans le domaine des verres de spin [10, 106] en physique statistique, et dans le domaine de l'imagerie [39, 15].

Lorsque les paramètres (taille du réseau, poids synaptiques, fonction de transfert, seuil, ...) sont convenablement choisis, les réseaux de neurones peuvent être une base naturelle pour la construction d'équations aux dérivées partielles pour le traitement d'images. De plus, ils possèdent une capacité de stabilisation sur les attracteurs ponctuels (en neurobiologie les attracteurs représentent les faits mémorisés ou «*patterns*») ou sur les trajectoires périodiques de la dynamique des réseaux, qui peuvent servir de guide pour la définition

de modèles EDP ayant des états asymptotiques non triviaux.

Dans ce chapitre, nous commençons par donner quelques rappels concernant les propriétés dynamiques des réseaux de neurones, pour ensuite donner des approximations par équations de diffusion anisotrope. Enfin nous donnerons une étude de stabilité asymptotique au sens de Lyapunov.

Signalons enfin que Grossberg [58, 60] et Cohen & Grossberg [33, 32] ont développé des processus d'apprentissage ou d'adaptation des systèmes biologiques en introduisant une partie inhibitrice et une autre excitatrice dans les équations d'état.

## 3.2 Le modèle dynamique de Hopfield

### 3.2.1 Définitions

Un réseau de neurones est un ensemble discret de points  $i$  porteurs d'une fonction d'entrée (*input*) et d'une fonction de sortie ou réponse (*output*).

Les neurones sont interconnectés entre eux par l'intermédiaire de sommes pondérées par des poids synaptiques formant généralement une matrice carrée.

La fonction d'entrée de chaque neurone  $i$  provient de deux sources différentes, soit de l'extérieur dans le cas d'un réseau sous influence externe, soit des autres neurones  $j$  ( $j \neq i$ ). La fonction totale d'entrée  $T$  s'écrit donc comme suit:

$$T_i = \sum_{j \neq i} J_{ij} V_j + E_i \quad \text{pour chaque neurone } i, \quad (3.1)$$

Dans cette équation,  $J_{ij}$  est la connexion (ou poids synaptique) entre les neurones  $i$  et  $j$ . Le réseau est dit excitateur si  $J_{ij} > 0$  et inhibiteur si  $J_{ij} < 0$ . Quand  $J_{ij} = 0$ , il n'y a pas de connexion entre les unités  $i$  et  $j$ .

Pour chaque neurone  $i$ , le paramètre  $E_i$  représente une entrée externe provenant généralement de l'environnement dans lequel évolue le réseau. Si pour tout  $i$ ,  $E_i = 0$  ou dépend uniquement de l'état initial, le réseau est dit *isolé* ou sans influence externe.

Suivant la définition de McCulloch et Pitts, à chaque neurone  $i$  correspond un seuil d'activation  $S_i$ , par rapport auquel l'état de chaque neurone est donné par la formule suivante:

$$V_i = \begin{cases} V^0 & \text{si } T_i < S_i \\ V^1 & \text{si } T_i \geq S_i \end{cases} \quad (3.2)$$

Dans le cas d'un réseau booléen, on a  $V^0 = 0$  et  $V^1 = 1$

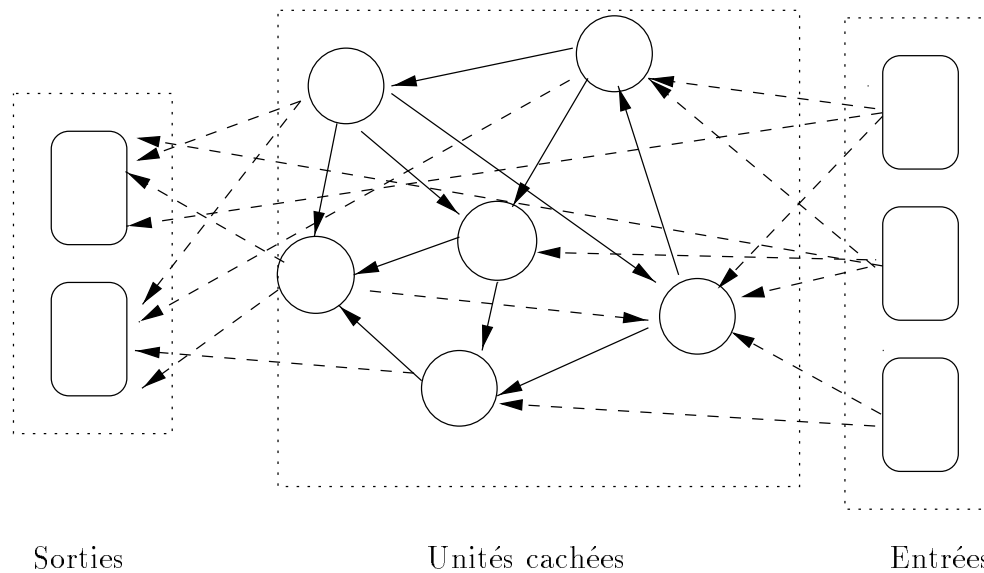


FIG. 3.1 – Architecture de réseaux de neurones

Dans le cas déterministe, le premier et le plus simple des modèles de réseaux neuronaux a été celui introduit par le physicien J.-J. Hopfield en 1982 [65]. Dans ce modèle, la fonction de sortie et celle d'entrée instantanée du neurone sont liées par la relation suivante:

$$V_i = g_i(\lambda U_i), \quad \text{pour chaque neurone } i. \quad (3.3)$$

La fonction de transfert  $g_i$  est monotone et sigmoïde avec deux asymptotes  $V^+$  et  $V^-$ , où le paramètre de gain  $\lambda$  est un nombre qui contrôle et qui, dans la limite  $\lambda \rightarrow \infty$ , redonne un réseau booléen (pour  $V^- = 0$  et  $V^+ = 1$ ).

L'équation décrivant la dynamique de l'activité des neurones est telle que l'état de chaque neurone  $i$  soit la somme d'une combinaison linéaire des activités des autres neurones  $j$  ( $j \neq i$ ) et d'un facteur extérieur au réseau  $E_i$ . Plus précisément, les états de neurones évoluent selon une loi dite de résistance-capacitance [66]:

$$\begin{cases} \frac{dU_i}{dt} = \sum_j J_{ij} V_j - \frac{U_i}{R_i} + E_i \\ V_i = g_i(\lambda U_i) \end{cases} \quad \text{pour chaque neurone } i \quad (3.4)$$

où les  $R_i$  sont des coefficients de résistances et  $E_i$  une fonction externe au réseau.

### 3.2.2 Propriétés dynamiques

Choisissons comme système compétitif de réseau de neurones, le réseau de Hopfield (2.4) avec  $\lambda > 0$  et  $g$  sigmoïdale, strictement croissante sur  $\mathbb{R}$  et bornée. Le réseau est supposé fonctionner sans influence externe. Le choix de la fonction de transfert (*input – output*)  $g$  est tel que:

$$g : \mathbb{R} \rightarrow (-1, 1), \quad g'(x) > 0, \quad \text{et} \quad g'(x) \leq g'(0) = \frac{1}{\beta} \quad \forall x \neq 0.$$

La fonction de transfert  $g$  est continue et strictement croissante, on peut donc définir sa réciproque  $G$  ( $G = g^{-1}$ ):

$$U_i = \frac{1}{\lambda} G(V_i)$$

Concernant la stabilité, les méthodes les plus appliquées en systèmes dynamiques sont les méthodes dites de Lyapunov [116].

Considérons le système dynamique suivant:

$$\frac{dU}{dt} = F(U) \quad \text{où} \quad U = (U_1, \dots, U_n) \quad \text{et} \quad F = (f_i : \mathbb{R}^n \rightarrow \mathbb{R})_{i=1, \dots, n} \quad (3.5)$$

Soit  $\mathcal{V}$  un voisinage d'un état d'équilibre  $S$  du système (2.5) ( $f_i(S) = 0$ ,  $i = 1, \dots, n$ ) et  $E$  une fonction de  $\mathcal{V}$  dans  $\mathbb{R}$ .

**Définition 1**  $E$  est une fonction de Lyapunov associée au système (2.5) dans un voisinage  $\mathcal{V}$  de  $S$  si:

- $E(S) = 0$  et  $E(U) > 0$  si  $U \in \mathcal{V} \setminus \{S\}$
- $E'(U) \leq 0$ , pour tout  $U \in \mathcal{V}$ .

**Théorème 1** Un état d'équilibre  $S$  est dit stable au sens de Lyapunov s'il existe une fonction de Lyapunov  $E$  définie sur un voisinage  $\mathcal{V}$  de  $S$ . Et, si de plus  $E'(U) < 0$ , pour tout  $U \in \mathcal{V} \setminus \{S\}$ , alors  $S$  est dit asymptotiquement stable.

En 1982, en rapportant la dynamique du réseau (2.4) à une dynamique d'interaction dans les verres de spin en physique statistique, Hopfield [66] a montré qu'il existe une fonction énergie (fonction de Lyapunov) pour les deux équations (2.4), dans le cas où  $J$  est symétrique en indice ( $J_{ij} = J_{ji}$ ):

$$E = -\frac{1}{2} \sum_i \sum_j J_{ij} V_i V_j + \frac{1}{\lambda} \sum_i \int_0^{V_i} G(V) dV \quad (3.6)$$

**Théorème 2 J.J. Hopfield [65]**

Si les  $J_{ij}$  sont constants et symétriques, alors la fonctionnelle  $E$  définie par (2.6) est une fonction de Lyapunov associée au système (2.4).

La fonctionnelle  $E$  est bornée et décroissante en temps. C'est donc une fonction de Lyapunov associée au système (2.4) [116]. Cette fonctionnelle  $E$  n'est autre que l'énergie totale du système (circuit électrique) qui décroît dans le temps [66].

De plus, la convergence de l'énergie  $E$  vers un minimum local correspond à la convergence du réseau (2.4) vers un état stable, les attracteurs du réseau sont donc des états stationnaires [28, 29].

Par contre, lorsque les poids synaptiques  $J_{ij}$  ne sont pas symétriques le système de Hopfield (2.4) n'admet pas de fonction de Lyapunov et peut donc avoir des cycles limites.

Avant d'étudier les propriétés dynamiques des réseaux de neurones avec apprentissage (ou à synapses adaptatives), il est important de noter que Grossberg [59] a montré la convergence d'un réseau neuronal compétitif pour lequel aucune fonction de Lyapunov n'est connue [62, 63, 64].

### 3.3 Mécanismes d'apprentissage

Les méthodes d'apprentissage dépendent de la dynamique utilisée pour calculer les poids successifs du réseau. Selon le type d'apprentissage, on distingue deux classes de réseaux:

#### 3.3.1 Réseaux à mémoire associative

Un réseau à mémoire associative est un réseau pour lequel, pour chaque état initial  $U(0) = (u_1(0), \dots, u_n(0))$ , il existe un état d'équilibre  $\bar{U}$  tel que  $\lim_{t \rightarrow \infty} U(t) = \bar{U}$ . Il est souvent décrit par l'équation suivante:

$$\frac{du_i}{dt} = -c_i u_i + \sigma_i \left( \sum_j J_{ij} u_j - \theta_i \right) + I_i \quad (3.7)$$

où les  $u_i(t)$  sont dans  $[0, 1]$ . A chaque unité  $i$  est associé un coefficient de *fatigue*  $c_i$ , une fonction de transfert  $\sigma_i$ , un seuil (ou *biais*)  $\theta_i$  et un facteur externe  $I_i$ .

Pour prouver la convergence de ce réseau vers un état d'équilibre  $\bar{U}$ , il convient de montrer qu'il existe une fonction de Lyapunov associée au système qui soit bornée et décroissante dans le temps.

Pour mémoriser des vecteurs  $\xi_1, \xi_2, \dots, \xi_p$  dans le réseau (2.7), il suffit de calculer la matrice synaptique  $J$  qui permettra au réseau d'avoir ces vecteurs comme états stables. Une méthode de calcul de cette matrice  $J$  est fondée sur la règle d'apprentissage Hebbienne

[61]:

$$J_{ij} = \sum_{k=1}^p \xi_{ik} \xi_{jk} \quad (3.8)$$

où  $\xi_{ik}$  est la  $k$  ème composante du vecteur  $\xi_p$ .

Une autre méthode de calcul de la matrice  $J$ , fondée aussi sur l'apprentissage Hebbien, s'écrit sous une forme différentielle comme suit [75]:

$$\frac{dJ_{ij}}{dt} = \xi_p \xi_j - J_{ij} \quad (3.9)$$

Plus généralement, Cohen et Grossberg [33] ont montré qu'un réseau de neurones est à mémoire associative, si on peut l'écrire sous la forme suivante:

$$\frac{du_i}{dt} = \alpha_i(u_i) \left[ \beta_i(u_i) - \sum_{j=1}^n \zeta_{ij} d_j(u_j) \right] \quad (3.10)$$

avec:  $\alpha_i(u_i) \geq 0$ ,  $\zeta_{ij} = \zeta_{ji}$  et  $d'_j(u_j) \geq 0$ .

Cette généralisation a permis de montrer l'équivalence mathématique de plusieurs modèles malgré leur forme différente.

Le réseau récurrent (2.7) peut être aussi réécrit sous la forme suivante:

$$U(t+1) = G(JU(t) - \theta) \quad (3.11)$$

où  $J$  est la matrice synaptique,  $U$  et  $\theta$  des vecteurs dont les  $i$ èmes composantes sont respectivement  $u_i$  et  $\theta_i$  et  $G : \mathbb{R}^n \rightarrow \mathbb{R}^n$  une fonction vérifiant  $G(x) = [g_i(x_i)]_i$ .

Pour les problèmes d'ajustement du réseau à un ensemble de données, Pineda a démontré que, si le réseau (2.11) converge vers un point fixe  $\phi$ , il est possible de calculer sa différentielle  $\frac{\partial \phi}{\partial J_{ij}}$  par une méthode de *rétro-propagation* [93]. Pour cela, on peut utiliser la dérivée partielle  $V = (\partial U / \partial J_{ij})_{ij}$ , pour obtenir:

$$V(t+1) = DF(JV(t) - \theta) (JV(t) + E(U(t))) \quad (3.12)$$

où  $E = (E_{ij})$  est la matrice dont les coefficients sont nuls sauf celui en  $(i, j)$ , qui est égal à 1.

Pour obtenir une approximation de la différentielle de  $\phi(x)$  par rapport à  $J$ , on utilise donc le réseau secondaire (2.12) dont les activations sont  $V$ . Cette approche est appelée apprentissage par descente du gradient.

La pratique a montré que la méthode d'apprentissage par descente du gradient est très lente en raison de nombreux minima locaux (points où la différentielle du réseau s'annule). Il est donc utile d'utiliser d'autres approches, par exemple les algorithmes génétiques [55], permettant de modifier la mise à jour des poids lors de la descente du gradient.

### 3.3.2 Réseaux à synapses adaptatives

Dans ce cas, les connexions synaptiques  $J_{ij}$  entre les neurones  $i$  et  $j$  ne sont plus constantes mais suivent l'évolution des activités des neurones (on dit que les synapses s'adaptent). Plusieurs adaptations de la matrice synaptique ont été utilisées [75, 97], mais un des premiers modèles avec apprentissage Hebbien a été introduit par le physicien D. Dong [45, 46, 47]. C'est un réseau neuronal du type Hopfield avec des synapses s'adaptant de la manière suivante:

$$\begin{cases} \frac{dU_i}{dt} &= \sum_j T_{ij} V_j - U_i + I_i \\ V_i &= g_i(\lambda U_i) \\ \beta \frac{dS_{ij}}{dt} &= -S_{ij} + V_i V_j \\ T_{ij} &= H_{ij}(S_{ij}) \end{cases} \quad (3.13)$$

où  $T_{ij}$  est une fonction continue croissante et bornée de la corrélation des potentiels  $V_i$  et  $V_j$  et  $\beta$  une constante positive. Une intégration de l'équation d'apprentissage des synapses nous donne:

$$S_{ij} = \int_{-\infty}^t \exp\left(\frac{\tau - t}{\beta}\right) V_i(\tau) V_j(\tau) d\tau \quad (3.14)$$

Ici, l'apprentissage est bel et bien du type Hebbien (ou apprentissage local) puisque les poids synaptiques  $J_{ij}$  de chaque neurone  $i$  ne dépendent que de l'activation des neurones voisins  $j$  ( $j \neq i$ ). Ce réseau peut s'interpréter de la manière suivante: lorsque la corrélation entre deux neurones est positive, le terme excitateur croît et, inversement lorsque la corrélation est négative, le terme inhibiteur décroît.

De plus, ce modèle (2.13) admet comme fonctionnelle énergie, la fonction  $E$  suivante:

$$E = -\frac{1}{2} \sum_{ij} T_{ij} V_i + \sum_i \int u_i - \sum_i I_i + \frac{1}{2} \sum_{ij} \int S_{ij} dT_{ij} \quad (3.15)$$

Par des différentiations élémentaires, et en supposant que les entrées  $I_i$  sont constantes et que les poids synaptiques  $T_{ij}$  sont symétriques en indices, on obtient le résultat suivant:

$$\left(\frac{dE}{dt} \leq 0\right) \quad \text{et} \quad \left(\frac{dE}{dt} = 0 \Rightarrow \frac{dU_i}{dt} = 0 \quad \text{et} \quad \frac{dS_{ij}}{dt} = 0\right) \quad (3.16)$$

Ce modèle est donc asymptotiquement stable au sens de Lyapunov.

Nous présenterons, dans le paragraphe suivant, une autre forme d'apprentissage fondée sur l'analogie entre les réseaux de Hopfield et les systèmes de réaction-diffusion.

## 3.4 Approximation par systèmes de réaction-diffusion

Ce paragraphe sera consacré à un problème classique de reconnaissance; celui-ci consiste en l'approximation de systèmes de réaction diffusion par les réseaux neuronaux et inversement. Pour cela nous reprendrons l'analyse introduite par Cottet [39, 40], et reprise par Edwards [48].

Nous nous intéresserons ainsi aux réseaux bidimensionnels de taille infinie, ce qui revient à faire tendre vers zéro la distance entre deux neurones voisins. Pour éviter toute confusion nous réservons les notations  $i, j$  pour les indices de tenseur, et nous supposons que les neurones sont situés sur les noeuds d'un maillage uniforme comme suit:

$$x_p = ph, \quad p \in \mathbb{Z}^2 \quad \text{où } x_p \text{ est la position du neurone.} \quad (3.17)$$

### 3.4.1 Approximation par diffusion anisotrope

Une première manière de comprendre le rapport entre les réseaux de neurones et les équations de réaction diffusion est de prendre:

$$J_{pq} = J_\varepsilon(x_p, x_q) = K\left(\frac{x_q - x_p}{\varepsilon}\right) \quad (3.18)$$

où  $K$  est une fonction à symétrie radiale, à support compact, et  $\varepsilon$  un petit paramètre qui donne la portée synaptique.

Pour déterminer le terme de diffusion de la fonction activité continue  $v$  (analogue de l'activité discrète  $V$ ), supposée assez régulière, on calcule l'intégrale suivante:

$$\int J_\varepsilon(x, y) [v(y) - v(x)] dy = \int K\left(\frac{y - x}{\varepsilon}\right) v(y) dy - v(x) \int K\left(\frac{y - x}{\varepsilon}\right) dy \quad (3.19)$$

Si on note par  $\tau_0$  et  $\tau_2$  les moments d'ordre 0 et 2 de la fonction  $K$ , un développement limité à l'ordre 1 de la fonction  $v$  au voisinage de  $x$  nous permet d'approcher cette intégrale par  $\varepsilon^4 \tau_2 \Delta v(x)$ , et une quadrature de l'opérateur intégral (2.19) nous donne l'approximation suivante:

$$\varepsilon^4 \tau_2 \Delta v(x_p) = \frac{1}{n} \sum_q J_\varepsilon(x_p, x_q) v(x_q) - v(x_p) \tau_0 \quad (3.20)$$

Dans (2.20) nous retrouvons la somme des activités des neurones pondérées avec les poids synaptiques. Le terme de droite du réseau de Hopfield est donc une discrétisation d'un opérateur de diffusion non linéaire. Le terme  $v(x_p) \tau_0$  correspond à celui de la réaction au point  $x_p$ . Nous pouvons donc introduire l'équation de réaction-diffusion suivante:

$$\frac{\partial v}{\partial t} - a_\varepsilon(v) \Delta v = f(v) \quad (3.21)$$

où

$$a_\varepsilon(v) = \varepsilon^4 \tau_2 = 2\varepsilon^4 \int_{\mathbb{R}^2} y^2 K(y) dy$$

L'équation de réaction-diffusion (2.21) est équivalente au réseau de Hopfield (2.4) dont les poids synaptiques vérifient la règle d'apprentissage (2.18).

Cependant, une discrétisation de  $\tau_0$  dans (2.20) nous permet d'éliminer le terme de réaction dans (2.21), et l'équation (2.20) peut être réécrite sous la forme suivante:

$$\frac{1}{2} \tau_2 \Delta v(x_p) = h^2 \sum_q J_{pq} [v(x_q) - v(x_p)] + \varepsilon^2 O(h^2) \quad (3.22)$$

Le second membre de cette équation contient une combinaison linéaire des potentiels des neurones qui nous permet d'introduire le réseau suivant:

$$\begin{cases} \frac{dU_p}{dt} &= \frac{1}{n} \sum_q J_{pq} (V_q - V_p) \\ V_p &= g(\lambda U_p) \end{cases} \quad (3.23)$$

Dans ce système, la fonction de transfert  $g$  est monotone et continue, on peut alors définir sa fonction réciproque  $G = g^{-1}$ , et ce réseau se réécrit comme suit:

$$\frac{dV_p}{dt} = \frac{\lambda}{G'(V_p)} \left[ \frac{1}{n} \sum_q J_{pq} (V_q - V_p) \right] \quad (3.24)$$

Nous pouvons annoncer la proposition suivante:

**Proposition 1** *Si les coefficients  $a_\varepsilon$  et  $n$  vérifient les propriétés suivantes:*

$a_\varepsilon(v) = \frac{\lambda h^{-2}}{G'(V)}$  *et,  $n$  est de l'ordre de  $\frac{h^2}{\varepsilon^2}$ . Si  $v$  et  $V$  les solutions de (2.21) et (2.24) sont assez régulières, alors:*

$$\max_p |V_p - v(x_p)| \leq O(\varepsilon^2) + \varepsilon^2 O(h^2). \quad (3.25)$$

**preuve:** Pour le neurone de position  $x_p$ , l'équation (2.22) se note sous la forme:

$$\Delta v(x_p) = \frac{1}{\tau_2} \left[ \frac{h^2}{\varepsilon^2} \sum_q K(x_p, x_q) (v(x_q) - v(x_p)) \right] + \varepsilon^2 O(h^2) \quad (3.26)$$

En écrivant l'équation (2.21) sans le terme de réaction au point  $x_p$ , et en utilisant l'approximation de l'opérateur Laplacien (2.26), on arrive au résultat suivant:

$$\frac{\partial v(x_p)}{\partial t} = a_\varepsilon(v) \frac{2h^2}{\tau_2} \sum_q J_{pq} [v(x_q) - v(x_p)] + \varepsilon^2 O(h^2) \quad (3.27)$$

Une identification des seconds membres des deux équations (2.24) et (2.27) nous donne l'équivalence entre les réseaux de neurones et les équations de diffusion anisotrope. ■

Signalons enfin que R. Edwards [48, 49] a introduit une généralisation des résultats de Cottet sur l'équivalence entre les réseaux de Hopfield:

$$\frac{dv_p}{dt} = \frac{1}{G'(v_p)} \left[ \lambda \sum_q T_{pq} v_q - \alpha G(v_p) \right], \quad (3.28)$$

et l'équation de réaction-diffusion suivante:

$$\frac{dv}{dt} = \frac{1}{G'(v)} \left[ \gamma \varepsilon^2 D_2 v - \alpha G(v) \right] \quad (3.29)$$

$$= \frac{1}{G'(v)} \left[ \frac{\gamma \varepsilon^2}{2} \operatorname{div}_y (\tau \nabla_y [\mu(x, y) v(y)]) \Big|_{(x, x)} + \gamma \tau_0 \mu(x, x) v(x) - \alpha G(v) \right] \quad (3.30)$$

où

$$\tau = \operatorname{diag}(\tau_1, \tau_2) \quad (3.31)$$

les  $\tau_p$  étant les moments d'ordre  $i$  de la fonction  $K$  définie dans (2.18). Pour  $\tau_p = \text{constante}$ , on retrouve donc l'équivalence introduite par Cottet [39].

### 3.4.2 Approximation par diffusion anisotrope et apprentissage

Dans cette partie, on cherche à exploiter l'analogie entre les équations de réaction-diffusion et les réseaux de neurones. Nous allons construire une loi d'évolution sur la matrice de diffusion qui traduise une loi d'apprentissage naturelle pour les réseaux de neurones. Le type général d'une telle loi peut être cherché sous la forme :

$$\frac{dJ_{pq}}{dt} + \frac{1}{\tau} J_{pq} = \frac{1}{\tau} G(u, x_p, x_q) \quad (3.32)$$

où  $G$  est une fonction qui permet de renforcer ou d'inhiber les connexions entre différents neurones selon la cohérence locale de l'activité du réseau, mesurée par  $G(u, x_p, x_q)$ , et  $\tau$  est un petit paramètre qui donne l'échelle de temps sur laquelle ces cohérences sont moyennées.

A la lumière de l'analyse qui précède il est alors naturel de chercher une loi d'évolution de la matrice de diffusion sous la forme suivante :

$$\frac{dL}{dt} + \frac{1}{\tau} L = \frac{1}{\tau} F(\nabla u) \quad (3.33)$$

la raison pour laquelle  $F$  apparaît comme une fonction de  $\nabla u$  et non de  $u$ , apparaîtra dans la dérivation qui suit.

Le but est alors de préciser la forme de  $G$  et d'en déduire celle de  $F$  de telle sorte que l'équation de réaction diffusion :

$$\frac{\partial u}{\partial t} - \operatorname{div}(L \nabla u) = f(u), \quad (3.34)$$

couplée avec (2.33) traduit, dans la limite d'une distance entre neurones voisins tendant vers 0, la dynamique du réseau de neurones dont les connexions obéissent à (2.32). En utilisant les techniques d'approximation par diffusion similaires à celles données au début de ce paragraphe, nous allons partir d'une fonction  $G$  s'écrivant sous la forme :

$$G(u, x_p, x_q) = \varepsilon^{-6} \theta\left(\frac{x_p - x_q}{\varepsilon}\right) |x_p - x_q|^2 \left[ s^2 - \frac{|u(x_p) - u(x_q)|^2}{|x_p - x_q|^2} \right] \quad (3.35)$$

Le facteur  $\theta$  est un facteur de forme isotrope qui est quelconque, à une constante multiplicative près, qui décroît vite et que nous supposons à support compact. Le seuil  $s$  est un paramètre de contraste, nous reviendrons par la suite sur l'interprétation de (2.35). Montrons que la fonction  $F$  qui découle de (2.35) est donnée par :

$$F(\nabla u) = |\nabla u|^2 \mathbf{P}_{\nabla u^\perp} + \frac{3}{2} (s^2 - |\nabla u|^2) \text{Id} \quad (3.36)$$

où  $\mathbf{P}_{\nabla u^\perp}$  est la matrice de la projection orthogonale sur l'orthogonale au gradient de  $u$  :

$$\mathbf{P}_{\nabla u^\perp} = \frac{1}{|\nabla u|^2} \begin{pmatrix} (\partial_x u)^2 & -\partial_x u \partial_y u \\ -\partial_x u \partial_y u & (\partial_y u)^2 \end{pmatrix} \quad (3.37)$$

Pour trouver le réseau de neurone discrétisant (2.33)-(2.34), il faut nous inspirer des techniques décrites dans [42] pour traiter le cas de la diffusion anisotrope.

Un opérateur de diffusion  $Du = \text{div}(L \nabla u)$  peut être discrétisé par un opérateur intégral sous la forme :

$$Q_\varepsilon u(x) = \int \sigma^\varepsilon(x, y, t) [u(x) - u(y)] dy \quad (3.38)$$

où  $\sigma^\varepsilon$  ( $\sigma^\varepsilon$  ressemble à  $K$  dans (2.18)) est construit à partir de fonctions cut-off de la manière suivante [42] :

$$\sigma^\varepsilon(x, y, t) = \varepsilon^{-2} \cdot \sum_{\substack{1 \leq i \leq 2 \\ 1 \leq j \leq 2}} m_{ij} \left( \frac{x+y}{2}, t \right) \Psi_{ij}^\varepsilon(y-x) \quad (3.39)$$

où  $m = [m_{ij}]$  est une matrice à déterminer en fonction de la matrice  $L$  et :

$$\Psi_{ij}^\varepsilon(x) = \varepsilon^{-2} \Psi_{ij} \left( \frac{x}{\varepsilon} \right) \quad (3.40)$$

En pratique, on choisit d'abord la forme des  $\Psi_{ij}$  puis on déduit les  $m_{ij}$  à partir de conditions sur les moments des  $\Psi_{ij}^\varepsilon$  pour assurer la consistance. Dans notre cas particulier, il s'avérera fructueux de choisir  $\Psi_{ij}$  sous la forme :

$$\Psi_{ij}(x) = x_i^\perp x_j^\perp \theta(x) \quad (3.41)$$

avec la notation  $\perp$  pour l'orthogonal :

$$\text{pour } x = (x_1, x_2) \text{ on a } x_1^\perp = x_2, \quad x_2^\perp = -x_1 \quad (3.42)$$

et  $\theta$  est une fonction à symétrie radiale qui doit vérifier les conditions de moments suivantes :

(C<sub>1</sub>) pour les moments d'ordre  $\alpha$ , avec  $|\alpha| \neq 2$ , il existe un réel  $r$  tel que:

$$\int x_i^\perp x_j^\perp \theta(x) x^\alpha dx = 0 \quad \text{pour } 1 \leq |\alpha| \leq r + 1 \quad (3.43)$$

(C<sub>2</sub>) les moments d'ordre 2 ( $|\alpha| = 2$  donc  $\alpha = e_k + e_l$  avec  $k$  et  $l$  dans  $[1, 2]$  (2.45)) vérifient la relation suivante:

$$\sum_{ij} \left( \int x_i^\perp x_j^\perp \theta(x) x^{e_k + e_l} dx \right) m_{ij}(x) = 2L_{kl}(x) \quad (3.44)$$

où  $\alpha = (\alpha_1, \alpha_2)$  est un multi-indice de  $\mathbb{N}^2$ ,  $x = (x_1, x_2)$  est un élément de  $\mathbb{R}^2$  et  $\{e_1, e_2\}$  est la base canonique de  $\mathbb{R}^2$ . On pose:

$$|\alpha| = \alpha_1 + \alpha_2 \quad \text{et} \quad x^\alpha = x_1^{\alpha_1} + x_2^{\alpha_2} \quad (3.45)$$

Dans ce cas, en suivant l'analyse de [42] (en remplaçant  $x_p x_q$  par  $x_p^\perp x_q^\perp$ ) on montre que la matrice  $m = [m_{ij}]$  se déduit de  $L$  par :

$$m = -L + \frac{3}{4} \text{Tr} L \text{Id} \quad (3.46)$$

Si  $L$  est donnée par :

$$L = |\nabla u|^2 \mathbf{P}_{\nabla u^\perp} + \beta \text{Id}, \quad (3.47)$$

on en déduit :

$$m = -|\nabla u|^2 \mathbf{P}_{\nabla u^\perp} + \left( \frac{\beta}{2} + \frac{3}{4} |\nabla u|^2 \right) \text{Id} \quad (3.48)$$

Cette forme spécifique se trouvera justifiée à posteriori par la loi d'apprentissage à laquelle elle conduit. Le paramètre  $\beta$  est lié au seuil  $s$  par la relation suivante :

$$s^2 = \frac{4}{3} \left( \frac{3}{4} |\nabla u|^2 + \frac{\beta}{2} \right) \quad (3.49)$$

d'où

$$\beta = \frac{3}{2} (s^2 - |\nabla u|^2), \quad (3.50)$$

et

$$L = \left[ |\nabla u|^2 \mathbf{P}_{\nabla u^\perp} + \frac{3}{2}(s^2 - |\nabla u|^2) \text{Id} \right] \quad (3.51)$$

Dans l'analogie entre les opérateurs de diffusion et les réseaux de neurones, les poids synaptiques  $J_{pq}$  reliant deux neurones de positions  $x_p$  et  $x_q$  seront donnés par :

$$J_{pq} = \sigma^\varepsilon(x_p, x_q, t) \quad (3.52)$$

en posant  $\mathbf{P}_{\nabla u^\perp} = [\mathbf{P}_{ij}]$ , une application de (2.48) au point  $(\frac{x_p+x_q}{2})$  nous permet d'écrire:

$$m_{ij}(\frac{x_p+x_q}{2}) = -|\nabla u(\frac{x_p+x_q}{2})|^2 \mathbf{P}_{ij}(\frac{x_p+x_q}{2}) + s^2 \text{Id} \quad (3.53)$$

d'où

$$J_{pq} = \varepsilon^{-6} \theta\left(\frac{x_p - x_q}{\varepsilon}\right)$$

$$\left[ \sum_{\substack{1 \leq i \leq 2 \\ 1 \leq j \leq 2}} \left( -|\nabla u(\frac{x_p+x_q}{2})|^2 \mathbf{P}_{ij}(\frac{x_p+x_q}{2}) (x_p - x_q)_i^\perp (x_p - x_q)_j^\perp \right) + s^2 \|(x_p - x_q)\|^2 \right]$$

or

$$\begin{aligned} & \sum_{\substack{1 \leq i \leq 2 \\ 1 \leq j \leq 2}} \left( -|\nabla u(\frac{x_p+x_q}{2})|^2 \mathbf{P}_{ij}(\frac{x_p+x_q}{2}) (x_p - x_q)_i^\perp (x_p - x_q)_j^\perp \right) \\ &= \sum_{\substack{1 \leq i \leq 2 \\ 1 \leq j \leq 2}} \left( \partial_i^\perp u(\frac{x_p+x_q}{2}) \partial_j^\perp u(\frac{x_p+x_q}{2}) (x_p - x_q)_i^\perp (x_p - x_q)_j^\perp \right) \\ &= \left[ \frac{\partial u}{\partial x_1}(\frac{x_p+x_q}{2}) \right]^2 \cdot (x_p - x_q)_1^2 + \left[ \frac{\partial u}{\partial x_2}(\frac{x_p+x_q}{2}) \right]^2 \cdot (x_p - x_q)_2^2 \\ &= |\nabla u(\frac{x_p+x_q}{2}) \cdot (x_p - x_q)|^2 \end{aligned}$$

d'où

$$J_{pq} = \varepsilon^{-6} \theta\left(\frac{x_p - x_q}{\varepsilon}\right) \left[ |\nabla u(\frac{x_p+x_q}{2}) \cdot (x_p - x_q)|^2 + s^2 \cdot \|(x_p - x_q)\|^2 \right]$$

En utilisant un développement limité de  $u$  à l'ordre 1, cette dernière équation peut être réécrite comme suit :

$$J_{pq} \approx \varepsilon^{-6} \cdot \theta\left(\frac{x_p - x_q}{\varepsilon}\right) \|(x_p - x_q)\|^2 \left[ s^2 - \frac{|u(x_p) - u(x_q)|^2}{\|(x_p - x_q)\|^2} \right] \quad (3.54)$$

Ce résultat nous donne la forme de la fonction  $G$  donnée par (2.35) et celle de  $F$  (2.36).

## Remarques

Les poids synaptiques vérifient les propriétés suivantes:

- ils sont bornés de la manière suivante:

$$(s^2 - |\nabla u|^2) \varepsilon^{-6} \cdot \theta\left(\frac{x_p - x_q}{\varepsilon}\right) \leq \frac{J_{pq}}{\|x_p - x_q\|^2} \leq s^2 \varepsilon^{-6} \cdot \theta\left(\frac{x_p - x_q}{\varepsilon}\right) \quad (3.55)$$

et ne sont non nuls que pour  $|x_p - x_q| \leq C \cdot \varepsilon$ .

- la valeur maximale est atteinte lorsque les états des sites  $u(x_p)$  et  $u(x_q)$  sont les mêmes. Quand la variation de  $u$  entre  $x_p$  et  $x_q$  est maximale,  $J_{pq}$  est minimal, alors :

$$J_{pq} = (s^2 - |\nabla u|^2) \varepsilon^{-6} \cdot \theta\left(\frac{x_p - x_q}{\varepsilon}\right) \|x_p - x_q\|^2 \quad (3.56)$$

Les connexions sont donc favorisées entre neurones présentant des réponses cohérentes.  $s$  correspond au seuil de cohérence au delà duquel ces connexions sont favorisées.

On aboutit alors à un système continu du type réaction-diffusion anisotrope:

$$\begin{cases} \frac{\partial u}{\partial t} - \operatorname{div}(L \nabla u) = f(u) \\ \frac{dL}{dt} + \frac{1}{\tau} L = \frac{1}{\tau} (|\nabla u|^2 \mathbf{P}_{\nabla u^\perp} + \frac{3}{2} (s^2 - |\nabla u|^2) \operatorname{Id}) \end{cases} \quad (3.57)$$

Notons ici, comme signalé dans [40], que l'équivalent "exact" qui résulterait de la régularisation spatiale de Cottet-Germain ( $F(\nabla u) = \mathbf{P}_{\nabla u^\perp}$ ) serait une loi d'apprentissage non complètement satisfaisante.

Il découle de l'analyse qui précède qu'un modèle de réseau de neurones équivalent est donné par:

$$\begin{cases} \frac{dU_p}{dt} = \sum_j J_{pq} V_q - U_p \\ \frac{dJ_{pq}}{dt} + \frac{1}{\tau} J_{pq} = \frac{1}{\tau} \varepsilon^{-6} \cdot \theta\left(\frac{x_p - x_q}{\varepsilon}\right) \|x_p - x_q\|^2 \left[ s^2 - \frac{|u(x_p) - u(x_q)|^2}{|x_p - x_q|^2} \right] \\ V_p = g_p(\lambda U_p) \end{cases} \quad (3.58)$$

Dans ce système, l'apprentissage est du type Hebbien puisque les poids synaptiques de chaque neurone ne dépendent que de l'activation des neurones voisins.

Comme nous l'avons indiqué, le choix du terme de réaction présuppose une connaissance a priori des niveaux de gris significatifs de l'image. En fait, comme nous le verrons en particulier dans nos illustrations numériques, le paramètre de contraste  $s$  rend le terme de réaction non nécessaire pour permettre des états asymptotiques non triviaux.

### 3.4.3 Introduction d'un seuil dans l'opérateur de diffusion

Dans ce paragraphe, nous rappelons l'équation d'apprentissage sur la matrice de diffusion :

$$\frac{dL}{dt} + \frac{1}{\tau}L = \frac{1}{\tau} \left( \frac{|\nabla u|^2}{s^2} \mathbf{P}_{\nabla u^\perp} + \frac{3}{2} s^2 \left( 1 - \frac{|\nabla u|^2}{s^2} \right) \text{Id} \right) \quad (3.59)$$

Lorsque  $|\nabla u| > s$ , cette équation peut être source d'instabilité. Nous pouvons remédier à cette possible antidiffusion par troncature en remplaçant le second membre dans (2.59) par  $F(\nabla u)$ , défini par :

$$F(\nabla u) = \begin{cases} \mathbf{P}_{\nabla u^\perp} & \text{si } |\nabla u| > s \\ \frac{|\nabla u|^2}{s^2} \mathbf{P}_{\nabla u^\perp} + \frac{3}{2} s^2 \left( 1 - \frac{|\nabla u|^2}{s^2} \right) \text{Id} & \text{si } |\nabla u| \leq s. \end{cases} \quad (3.60)$$

Nous verrons dans le paragraphe suivant que le réseau de neurone équivalent admet des propriétés de stabilité asymptotique au sens de Lyapunov qui donnent la possibilité d'un renforcement des contrastes, ce que nous reverrons dans la partie application.

Finalement, nous pouvons introduire un nouveau modèle pour le traitement d'image. Ce modèle est basé sur une équation de diffusion non linéaire et anisotrope dont la matrice de diffusion obéit à une règle d'apprentissage comme suit :

$$\begin{cases} \frac{\partial u}{\partial t} - \text{div}(L \nabla u) = 0 \\ \frac{dL}{dt} + \frac{1}{\tau}L = \frac{1}{\tau} \begin{cases} \mathbf{P}_{\nabla u^\perp} & \text{si } |\nabla u| > s \\ \frac{|\nabla u|^2}{s^2} \mathbf{P}_{\nabla u^\perp} + \frac{3}{2} s^2 \left( 1 - \frac{|\nabla u|^2}{s^2} \right) \text{Id} & \text{si } |\nabla u| \leq s. \end{cases} \end{cases} \quad (3.61)$$

où  $\tau$  est un paramètre de relaxation et  $s$  un seuil du gradient.

Une étude théorique de ce modèle est donnée dans [37] en annexe 1. En pratique, pour le traitement d'images, ce modèle sera complété par deux conditions initiales :

- $u(., 0) = u_0(x)$ , le niveau de gris de l'image originale.
- $L(., 0) = L_0 = \text{Id}$ , ce qui correspond à un filtrage isotrope du type Gaussien.

Dans le chapitre 4 on trouvera une étude détaillée de ce modèle, ainsi que de ses paramètres.

### 3.4.4 Stabilité asymptotique au sens de Lyapunov

Dans ce paragraphe nous montrons quelques propriétés de stabilité en faisant une analyse de type Lyapunov. Notre réseau admet l'architecture suivante:

$$\begin{cases} \frac{dU_p}{dt} &= \sum_j J_{pq} (V_q - V_p) \\ \tau \frac{dJ_{pq}}{dt} &= -J_{pq} + \varepsilon^{-6} \theta\left(\frac{x_p - x_q}{\varepsilon}\right) \|x_p - x_q\|^2 \left[ s^2 - \frac{|u(x_p) - u(x_q)|^2}{|x_p - x_q|^2} \right] \\ V_p &= g_p(\lambda U_p) \end{cases} \quad (3.62)$$

Considérons le changement de variables suivant:

$$T_{pq} = J_{pq} - \varepsilon^{-6} \theta\left(\frac{x_p - x_q}{\varepsilon}\right) \|x_p - x_q\|^2 s^2 \quad (3.63)$$

ce qui nous permet d'écrire:

$$\begin{aligned} \tau \frac{\partial T_{pq}}{\partial t} &= \tau \frac{\partial J_{pq}}{\partial t} \\ &= -T_{pq} - \varepsilon^{-6} \theta\left(\frac{x_p - x_q}{\varepsilon}\right) (V_p - V_q)^2 \end{aligned}$$

Et, en posant:

$$S_{pq} = \frac{\theta\left(\frac{x_p - x_q}{\varepsilon}\right)}{\varepsilon^6} T_{pq} = A_{pq} T_{pq}, \quad (3.64)$$

le système discret (2.62) se réécrit:

$$\begin{cases} \frac{dU_p}{dt} &= \sum_j J_{pq} (V_q - V_p) \\ V_p &= g_p(\lambda U_p) \\ \tau \frac{dS_{pq}}{dt} &= -S_{pq} - (V_p - V_q)^2 \end{cases} \quad (3.65)$$

**Théorème 3** *Si les  $J_{pq}$  sont symétriques ( $J_{pq} = J_{qp}$ ), et si  $g$  est continue et strictement croissante, alors les états d'équilibre du système (2.65) sont asymptotiquement stables au sens de Lyapunov.*

**preuve:**

Considérons la fonctionnelle suivante:

$$E = \frac{1}{2} \sum_{pq} J_{pq} (V_q - V_p)^2 + \frac{1}{2} \sum_{pq} S_{pq}^2 \quad (3.66)$$

- Montrons d'abord que la fonctionnelle  $E$  est bornée.  
Pour cela, intéressons nous à l'équation d'apprentissage de la matrice synaptique :

$$\tau \frac{dS_{pq}}{dt} = -S_{pq} - (V_p - V_q)^2 \quad (3.67)$$

Une intégration directe de la première équation de ce système nous donne:

$$S_{pq} = \int_0^t -\exp\left(\frac{s-t}{\tau}\right) (V_p(s) - V_q(s))^2 ds + S_{pq}(0) \exp\left(\frac{-t}{\tau}\right) \quad (3.68)$$

D'autre part, les potentiels  $V_p$  et  $V_q$  sont dans  $(-1, 1)$ ; ceci donne alors l'inégalité suivante:

$$(V_p - V_q)^2 \leq 4 \quad (3.69)$$

Ceci entraîne:

$$|S_{pq}| \leq 4 + \exp\left(\frac{-t}{\tau}\right) [ |S_{pq}(0)| - 4 ]$$

Finalement, on a la majoration suivante:

$$|S_{pq}| \leq \max(4, |S_{pq}(0)|) \quad (3.70)$$

Par ailleurs, l'équation (2.64) nous donne:

$$|J_{pq}| \leq A_{pq} \left( \max(4, |S_{pq}(0)|) + C_{pq} \right) = J_{pq}^{max} \quad (3.71)$$

où la constante  $C_{pq}$  est donnée par:

$$C_{pq} = \max_{pq} [ \|x_p - x_q\|^2 s^2 ] \quad (3.72)$$

Ceci nous permet d'écrire:

$$\begin{aligned} |E| &\leq 2 \sum_{pq} J_{pq} + \frac{1}{2} \sum_{pq} \int S_{pq} dJ_{pq} \\ &\leq \left[ 2 + \frac{1}{2} \max(4, |S_{pq}(0)|) \right] \sum_{pq} J_{pq}^{max} = E_{max} \end{aligned}$$

La fonctionnelle énergie  $E$  est donc bornée. De plus, elle vérifie la propriété suivante:

$$\max_t |E(t)| \leq E_{max} \quad (3.73)$$

- Montrons maintenant que la fonctionnelle  $E(t)$  est décroissante.

Une dérivation de cette fonctionnelle par rapport à  $V_p$  et  $J_{pq}$  nous permet d'écrire:

$$\begin{aligned}\frac{\partial E}{\partial V_p} &= -2 \sum_j J_{pq} (V_q - V_p) \\ &= -2 \frac{dU_p}{dt}\end{aligned}$$

nous avons également:

$$\begin{aligned}\frac{\partial E}{\partial J_{pq}} &= S_{pq} + (V_p - V_q)^2 \\ &= -\tau \frac{dS_{pq}}{dt}.\end{aligned}$$

Ce qui nous permet d'écrire:

$$\begin{aligned}\frac{dE}{dt} &= \sum_i \frac{\partial E}{\partial V_p} \frac{dV_p}{dt} + \sum_{pq} \frac{\partial E}{\partial J_{pq}} \frac{dJ_{pq}}{dt} \\ &= -2 \sum_i G'(V_p) \left(\frac{dV_p}{dt}\right)^2 - \sum_{pq} A_{pq} \left(\frac{dJ_{pq}}{dt}\right)^2\end{aligned}$$

La fonction de transfert  $g$  étant continue strictement croissante,  $G' = (g^{-1})'$  est donc positive. De plus, les constantes  $A_{pq}$  sont positives. On peut donc conclure que  $E$  est décroissante.

En résumé, la fonctionnelle  $E(t)$  est bornée et vérifie la propriété suivante:

$$\left(\frac{dE}{dt} \leq 0\right) \quad \text{et} \quad \left(\frac{dE}{dt} = 0 \Rightarrow \frac{dU_p}{dt} = 0 \quad \text{et} \quad \frac{dJ_{pq}}{dt} = 0\right) \quad (3.74)$$

$E$  est donc une fonction de Lyapunov associée au système discret (2.62), et les états d'équilibre de ce système sont asymptotiquement stables au sens de Lyapunov. ■

Le caractère éventuellement antidiffusif du réseau, mis en évidence dans le système de réaction diffusion avant seuillage (2.57), donne à ce résultat de stabilité une importance particulière.

### Comportement du réseau adaptatif (2.62)

Dans l'espace  $\mathcal{E}$  des couples  $(V_p(t), J_{pq}(t))$  tels que:  $\frac{dV_p}{dt} \neq 0$  ou  $\frac{dJ_{pq}}{dt} \neq 0$ , le réseau neuronal (2.62) peut être écrit sous la forme dynamique suivante:

$$\frac{d\mathcal{X}_p}{dt} = F(\mathcal{X}_p); \quad p = 1, \dots, n; \quad \text{où } \mathcal{X}_p \in \mathcal{E} \quad (3.75)$$

L'évolution des états dans l'espace  $\mathcal{E}$  aboutit à des attracteurs qui sont situés sur les minima locaux de  $E$ . En effet, la fonctionnelle  $E(t)$  définie par (2.66) a une limite finie, et sa dérivée par rapport au temps converge vers 0 :

$$\lim_{t \rightarrow \infty} E(t) = \text{constante} \quad \text{et} \quad \lim_{t \rightarrow \infty} \frac{dE(t)}{dt} = 0 \quad (3.76)$$

Si l'on suppose que dans l'espace  $\mathcal{E}$ , le point  $(V_i^0, J_{pq}^0)$  est fixe, alors une perturbation  $(\delta V_i^0, \delta J_{pq}^0)$  nous donne le système suivant :

$$\begin{cases} \frac{d\delta U_p}{dt} = \sum_j (\delta J_{pq} V_q + J_{pq} g'_p(U_q) \delta U_p) - \delta U_p \\ \frac{d\delta J_{pq}}{dt} = -\delta J_{pq} + 2V_p g'_q(U_q) \delta V_q + 2V_q g'_p(U_p) \delta V_p + 2V_p g'_p(U_p) \delta V_p + 2V_q g'_q(U_q) \delta V_q \end{cases}$$

Et pour  $(V_p = Cst, J_{pq} = Cst)$ , en supposant que  $g(0) = 0$ , ce système devient :

$$\begin{cases} \frac{d\delta U_p}{dt} = -\delta U_p \\ \frac{d\delta J_{pq}}{dt} = -\delta J_{pq} \end{cases}$$

ceci donne alors :

$$\lim_{t \rightarrow +\infty} \delta U_p = \lim_{t \rightarrow +\infty} \delta J_{pq} = 0$$

Les états constants sont donc des points d'équilibre stables.

### 3.5 Conclusion et Discussion

Dans ce chapitre, nous avons présenté trois modèles pour le traitement d'images : un modèle fondé sur les réseaux de neurones et deux sur les équations aux dérivées partielles, plus précisément sur les équations de diffusion anisotrope.

Après avoir établi l'analogie entre les réseaux de neurones et les équations de diffusion anisotrope, nous avons obtenu quelques résultats de stabilité asymptotique au sens de Lyapunov.



# Chapitre 4

## Etude de la Diffusion Anisotrope

DANS le chapitre précédent nous avons introduit un nouveau filtre pour le traitement d'images. Ce modèle est basé sur une équation de diffusion anisotrope avec un apprentissage sur l'opérateur de diffusion. Pour des raisons techniques, nous ne pouvons prouver l'existence et l'unicité de la solution que pour un modèle régularisé (résultat détaillé en annexe 1). Dans ce chapitre, nous donnerons quelques résultats de stabilité sur le modèle "brut" et des résultats partiels sur le modèle régularisé. Nous présenterons aussi des schémas numériques stables. L'application de ces algorithmes au traitement d'images ainsi que l'étude des paramètres seront détaillés dans le chapitre suivant.

### 4.1 Etude Théorique

Dans cette partie, nous présenterons des résultats théoriques partiels sur une version régularisée du modèle de diffusion anisotrope introduit dans le chapitre précédent. Cette régularisation est faite de la manière suivante :

$$\frac{\partial u}{\partial t} - \operatorname{div}(L \nabla u) = 0 \quad (4.1)$$

$$\frac{dL}{dt} + \frac{1}{\tau} L = \frac{1}{\tau} F(\nabla_\varepsilon u) \quad (4.2)$$

avec

$$\nabla_\varepsilon u = \nabla(u \star f_\varepsilon), \quad f_\varepsilon(y) = \varepsilon^{-2} f\left(\frac{y}{\varepsilon}\right), \quad \int f dx = 1, \quad (4.3)$$

et

$$F(\xi) = \begin{cases} \mathbb{P}_{\xi^\perp} & \text{si } |\xi| > s \\ \frac{|\xi|^2}{s^2} \mathbb{P}_{\xi^\perp} + \frac{3}{2} s^2 \left(1 - \frac{|\xi|^2}{s^2}\right) Id & \text{si } |\xi| \leq s \end{cases} \quad (4.4)$$

$L$  et  $F(\nabla_\varepsilon u)$  sont deux matrices  $2 \times 2$  et  $P$  est la matrice de la projection orthogonale sur l'orthogonale à  $\xi$  ( $\xi = (\xi_1, \xi_2) \in \mathbb{R}^2$ ), donnée par :

$$\mathbb{P}_{\xi^\perp} = \frac{1}{|\xi|^2} \begin{pmatrix} \xi_2^2 & -\xi_1 \xi_2 \\ -\xi_1 \xi_2 & \xi_1^2 \end{pmatrix}, \quad (4.5)$$

## Propriétés de l'opérateur $F(\nabla u)$

L'opérateur  $F(\nabla u)$  vérifie les propriétés suivantes:

(P<sub>1</sub>)  $F(\nabla u)$  est une matrice symétrique définie positive.

La symétrie de la matrice  $\mathbb{P}_{\nabla u^\perp}$  nous donne celle de  $F(\nabla u)$ .

$\forall v, w \in \mathbb{R}^2$  on a:

$$\langle F(\nabla u)v, w \rangle = \begin{cases} \langle \mathbb{P}_{\nabla u^\perp} v, w \rangle & \text{si } |\nabla u| > s \\ \frac{|\nabla u|^2}{s^2} \langle \mathbb{P}_{\nabla u^\perp} v, w \rangle + \frac{3}{2}s^2 \left(1 - \frac{|\nabla u|^2}{s^2}\right) \langle v, w \rangle & \text{si } |\nabla u| \leq s \end{cases}$$

en posant  $\nabla u = (u_x, u_y)$  et  $w = (w_1, w_2)$ , nous pouvons écrire

$$\begin{aligned} \langle \mathbb{P}_{\nabla u^\perp} v, w \rangle &= \frac{1}{|\nabla u|^2} (u_y^2 v_1 w_1 - u_x u_y v_2 w_1 - u_x u_y v_1 w_2 + u_x^2 w_2^2) \\ &= \frac{\langle \text{rot } u, v \rangle \langle \text{rot } u, w \rangle}{|\nabla u|^2} \end{aligned}$$

où  $\text{rot } u = (u_y, -u_x)$ . En faisant  $v = w$  on voit que  $\mathbb{P}_{\nabla u^\perp}$  et par suite  $F(\nabla u)$  sont positives.

La positivité de  $F(\nabla u)$  permet de prévenir toute antidiffusion dans le système (3.1). Pour  $L_0$  une valeur initiale de  $L$  vérifiant:

$$L_0 \geq \beta Id \quad \text{avec } \beta > 0 \quad (4.6)$$

une intégration de (3.2) nous donne:

$$L(t) = \exp\left(\frac{-t}{\tau}\right)L_0 + \frac{1}{\tau} \int_0^t \exp\left(-\frac{t-s}{\tau}\right) F(\nabla u(\cdot, s)) ds \quad (4.7)$$

l'opérateur  $F(\nabla u)$  étant positif, donc:

$$L(\cdot, t) \geq \beta Id \quad (4.8)$$

La matrice  $L(u)$  est symétrique définie positive, l'équation (3.1) est donc parabolique.

(P<sub>2</sub>)  $F(\nabla u)$  est borné:

$$|F(\nabla u)| \leq C \quad \forall u \in \mathbb{R}^2 \quad (4.9)$$

(P<sub>3</sub>) Les dérivées de  $F(\nabla u)$  sont bornées et s'annulent à l'infini

$$|F(\nabla u)| \leq C(1 + |u|)^{-1} \quad \forall u \in \mathbb{R}^2 \quad (4.10)$$

Pour  $\varepsilon = 0$ , l'intégration de (3.7) dans l'équation (3.1) entraîne :

$$\frac{\partial u}{\partial t} - \exp\left(\frac{-t}{\tau}\right) \operatorname{div}(L_0[\nabla u]) - \frac{1}{\tau} \int_0^t \exp\left(-\frac{t-s}{\tau}\right) \operatorname{div}\left(F(\nabla u(.,s)) \nabla u(.,t)\right) ds = 0 \quad (4.11)$$

Notre modèle est donc de type Volterra non standard puisque les variables  $s$  et  $t$  sont couplées dans l'opérateur intégral. Notons aussi la forte non linéarité introduite par  $F(\nabla u)$ .

Commençons par donner les notations suivantes :  $\Omega = ]-1, 1[ \times ]-1, 1[$  est un carré qui représente le niveau de gris de l'image,  $Q = \Omega \times [0, T]$ ,  $H^1(\Omega)$  l'espace de Sobolev des fonctions de  $L^2$  à dérivées dans  $L^2$  muni de la norme suivante :

$$\|u\|_{H^1(\Omega)} = \left( \|u\|_{L^2(\Omega)}^2 + \sum_{i=1}^2 \|\partial_{x_i} u\|_{L^2(\Omega)}^2 \right)^{1/2}$$

et

$$\|u\|_{L^2(0,T;H^1(\Omega))} = \left( \int_0^T \|u(t)\|_{H^1(\Omega)}^2 dt \right)^{1/2}$$

**Théorème 4** *Supposons que  $(u_0, L_0)$  est borné dans  $H^1(\Omega) \times (H^1(\Omega))^4$ , alors le système (3.1)-(3.2) admet une solution unique :*

$$(u, L) \text{ dans } \left[ L^\infty(0, T, H^1(\Omega)) \cap L^2(0, T, H^1(\Omega)) \right] \times L^\infty(0, T, (H^1(\Omega))^4) \quad (4.12)$$

*Cette solution dépend continuellement des données initiales et vérifie le principe du maximum :*

$$\text{si } |u_0| \leq 1, \text{ alors } |u(., t)| \leq 1 \quad \forall t. \quad (4.13)$$

L'existence et l'unicité de la solution sont démontrées en annexe 1 [37].

### Principe du maximum

Soit  $u_0$ , le niveau de gris de l'image à restaurer. Si l'on note par  $u^n$ , la valeur de  $u$  à l'instant  $t_n = n\Delta t$ , alors l'équation (3.11) peut être écrite sous la forme semi discrète suivante :

$$\frac{u^{n+1} - u^n}{\Delta t} - e^{\left(\frac{-t_n}{\tau}\right)} \Delta u_{n+1} - \frac{1}{\tau} \int_0^{t_n} \exp\left(-\frac{t_n-s}{\tau}\right) \operatorname{div}\left(F(u^n(.,s)) \nabla u^{n+1}(.,t)\right) ds = 0$$

La propriété (P<sub>1</sub>) montre que ce problème est bien posé. Cette équation peut être réécrite comme suit :

$$u^{n+1} - \Delta t e^{\left(\frac{-t_n}{\tau}\right)} \Delta u_{n+1} - \frac{\Delta t}{\tau} \int_0^{t_n} \exp\left(-\frac{t_n-s}{\tau}\right) \operatorname{div}\left(F(u^n(.,s)) \nabla u^{n+1}(.,t)\right) ds = u_n$$

Si  $|u^n| \leq 1$  alors le principe du maximum nous donne  $|u^{n+1}| \leq 1$ . Les résultats de convergence [37], permettant le passage à la limite quand  $\Delta t \rightarrow 0$  nous donnent  $|u(., t)| \leq 1 \forall t$ .

Cette propriété est très importante en traitement d'images par les EDP. Si l'on veut restaurer une image répartie sur 256 niveaux de gris ( $u_0 \in [0, 255]$ ), un changement de variable entraîne  $u_0 \in [-1, 1]$ . L'image traitée  $u(., t)$  restera alors dans  $[-1, 1]$  et un changement de variables inverse nous permet d'avoir la même répartition en niveaux de gris ( $u(., t) \in [0, 255]$ ).

### Dépendance continue des données initiales

Soient  $(u_0, L_0)$ ,  $(v_0, M_0)$  deux données initiales et  $(u, L)$ ,  $(v, M)$  les solutions correspondantes. Comme pour la démonstration de l'unicité (annexe 1), nous posons  $e = u - v$ ,  $E = L - M$ ; une soustraction des deux systèmes en  $u$  et  $v$  nous permet d'écrire :

$$\frac{\partial e}{\partial t} - \operatorname{div}(M \nabla e) = -\operatorname{div}(E \nabla u) \quad (4.14)$$

$$\frac{\partial E}{\partial t} + E = F(\nabla_\varepsilon u) - F(\nabla_\varepsilon v) \quad (4.15)$$

L'équation (3.15) nous donne :

$$\|E(., t)\|_{L^\infty} \leq e^{-T} \|E_0\|_{L^\infty} + C \int_0^t \|e(., s) \star \nabla f_\varepsilon\|_{L^\infty} ds \quad (4.16)$$

$$\leq e^{-T} \|E_0\|_{L^\infty} + C \int_0^t \|e(., s)\|_{L^2} ds \quad (4.17)$$

Si l'on multiplie (3.14) par  $e$  et si l'on intègre sur  $\Omega$  on obtient :

$$\begin{aligned} \frac{1}{2} \frac{d}{dt} \|e\|_{L^2}^2 + \beta e^{-T} \|\nabla e\|_{L^2}^2 &\leq \|E \nabla u\|_{L^2} \|\nabla e\|_{L^2} \\ &\leq \frac{1}{2} \left( \frac{e^T}{\beta} \|E \nabla u\|_{L^2}^2 + \beta e^{-T} \|\nabla e\|_{L^2}^2 \right) \end{aligned}$$

et

$$\frac{d}{dt} \|e\|_{L^2}^2 \leq C \|E \nabla u\|_{L^2}^2 \leq C \|E\|_{L^\infty}^2 \|\nabla u\|_{L^2}^2 \quad (4.18)$$

$\nabla u$  étant dans  $L^2(Q)$ , les inégalités (3.17) et (3.18) montrent que la solution  $(u, L)$  dépend continuellement des données initiales  $u_0$  et  $L_0$ .

Dans le lemme suivant, nous montrons un résultat de stabilité uniforme par rapport à  $\varepsilon$ .

**Lemme 1** *Si  $u_0 \in H^1(\Omega)$  et  $L_0 \in (H^1(\Omega))^4$  alors on a l'estimation suivante:*

$$u_\varepsilon \text{ est bornée dans } L^\infty(0, T, H^1(\Omega)) \quad (4.19)$$

**preuve:** La matrice  $L_\varepsilon$  étant symétrique, définie positive, d'où l'existence d'une matrice  $M_\varepsilon$  symétrique et uniformément définie positive vérifiant :

$$L_\varepsilon = M_\varepsilon (M_\varepsilon)^t \quad (4.20)$$

En multipliant l'équation (3.1) par  $(-\operatorname{div}(L_\varepsilon \nabla u_\varepsilon))$  on aura :

$$\begin{aligned} \left\langle \frac{\partial u_\varepsilon}{\partial t}, -\operatorname{div}(L_\varepsilon \nabla u_\varepsilon) \right\rangle &= \left\langle \frac{\partial \nabla u_\varepsilon}{\partial t}, L_\varepsilon \nabla u_\varepsilon \right\rangle \\ &= \left\langle M_\varepsilon \frac{\partial \nabla u_\varepsilon}{\partial t}, M_\varepsilon \nabla u_\varepsilon \right\rangle \\ &\leq 0 \end{aligned}$$

et

$$\|\operatorname{div}(L_\varepsilon \nabla u_\varepsilon)\|_{L^2}^2 = \int_\Omega \left\langle \frac{\partial \nabla u_\varepsilon}{\partial t}, L_\varepsilon \nabla u_\varepsilon \right\rangle \quad (4.21)$$

$$\leq \frac{1}{2} \frac{d}{dt} \|M_\varepsilon \nabla u_\varepsilon\|_{L^2}^2 + C \|\nabla u_\varepsilon\|_{L^2}^2 \quad (4.22)$$

or

$$M_\varepsilon \frac{\partial \nabla u_\varepsilon}{\partial t} = \frac{\partial (M_\varepsilon \nabla u_\varepsilon)}{\partial t} - \frac{\partial M_\varepsilon}{\partial t} \nabla u_\varepsilon$$

alors

$$\left\langle \frac{\partial (M_\varepsilon \nabla u_\varepsilon)}{\partial t}, M_\varepsilon \nabla u_\varepsilon \right\rangle \leq \left\langle \frac{\partial M_\varepsilon}{\partial t} \nabla u_\varepsilon, M_\varepsilon \nabla u_\varepsilon \right\rangle$$

Ceci entraîne :

$$\frac{1}{2} \frac{d}{dt} \|M_\varepsilon \nabla u_\varepsilon\|_{L^2}^2 \leq \int_\Omega \left\langle \frac{\partial M_\varepsilon}{\partial t} \nabla u_\varepsilon, M_\varepsilon \nabla u_\varepsilon \right\rangle dx \quad (4.23)$$

de plus

$$M_\varepsilon \frac{\partial M_\varepsilon}{\partial t} = \frac{1}{2} \frac{\partial L_\varepsilon}{\partial t} \quad (4.24)$$

en utilisant

$$\frac{\partial L_\varepsilon}{\partial t} = F(\nabla u_\varepsilon) - L_\varepsilon$$

L'inégalité (3.23) peut être réécrite comme suit :

$$\frac{1}{2} \frac{d}{dt} \|M_\varepsilon \nabla u_\varepsilon\|_{L^2}^2 \leq \int_\Omega \left\langle F(\nabla u_\varepsilon) \nabla u_\varepsilon, \nabla u_\varepsilon \right\rangle - \int_\Omega \left\langle L_\varepsilon \nabla u_\varepsilon, \nabla u_\varepsilon \right\rangle \quad (4.25)$$

En utilisant la définie positivité de  $L_\varepsilon$  et la propriété (P<sub>2</sub>), on peut écrire :

$$\begin{aligned} \frac{d}{dt} \|M_\varepsilon \nabla u_\varepsilon\|_{L^2}^2 &\leq \int_\Omega \langle F(\nabla u_\varepsilon) \nabla u_\varepsilon, \nabla u_\varepsilon \rangle dx \\ &\leq C \|\nabla u_\varepsilon\|_{L^2}^2 \end{aligned}$$

$M_\varepsilon$  étant définie positive alors

$$\frac{d}{dt} \|M_\varepsilon \nabla u_\varepsilon\|_{L^2}^2 \leq C \|\nabla u_\varepsilon\|_{L^2}^2 \leq C' \|M_\varepsilon \nabla u_\varepsilon\|_{L^2}^2 \quad (4.26)$$

on peut en déduire que  $M_\varepsilon \nabla u_\varepsilon$  et par suite  $\nabla u_\varepsilon$  sont bornés dans  $L^\infty(0, T; L^2(\Omega))$ ; d'où l'estimation (3.19). ■

La forte non linéarité dans l'opérateur de diffusion (3.1) ne nous a pas permis d'avoir des estimations a priori sur les dérivées secondes de  $\nabla u_\varepsilon$  ou sur les dérivées premières de  $L_\varepsilon$  nécessaires pour pouvoir montrer l'existence d'une solution du modèle "brut".

Les expériences (chapitre 4) montrent que le modèle "brut" (2.61) admet des propriétés d'attractivité mais pour des raisons techniques, on ne peut prouver que l'existence d'une sphère d'attraction comme suit :

**Proposition 2** *La solution du modèle (2.61) vérifie la propriété suivante :*

$$\overline{\lim}_{t \rightarrow \infty} \left| \frac{du}{dt}(t) \right|_{L^2}^2 \leq \frac{3s^2}{2} |\Omega| \left( \frac{1}{2\tau} + 1 \right) \quad (4.27)$$

**preuve:**

Une dérivation de l'équation (3.1) par rapport au temps nous donne:

$$u'' - \operatorname{div}(L' \nabla u) - \operatorname{div}(L \nabla u') = 0$$

avec la notation ( $'$ ) pour la dérivée par rapport à  $t$ . (3.2) entraîne :

$$u'' - \operatorname{div}\left(\frac{1}{\tau}(-L + F(\nabla u))\nabla u\right) - \operatorname{div}(L \nabla u') = 0 \quad (4.28)$$

si l'on multiplie (3.28) par  $u'$  et si l'on intègre sur  $\Omega$ , on aura :

$$\frac{1}{2} \frac{d}{dt} |u'|^2 - \frac{1}{\tau} \int_\Omega \langle L \nabla u, \nabla(u') \rangle + \frac{1}{\tau} \int_\Omega \langle F(\nabla u) \nabla u, \nabla u' \rangle + \int_\Omega \langle L \nabla u, \nabla(u') \rangle + |N \nabla u'|_{L^2}^2 = 0$$

maintenant, si l'on multiplie (3.1) par  $u'$  et si l'on intègre sur  $\Omega$ , on aura :

$$|u'|^2 + \int \langle L \nabla u, \nabla(u') \rangle = 0$$

la matrice  $L$  étant symétrique, définie positive. D'où l'existence d'une matrice  $M$  symétrique et uniformément définie positive vérifiant :

$$L = M M^t \quad (4.29)$$

ceci entraîne :

$$\frac{1}{2} \frac{d}{dt} |u'|_{L^2}^2 + \frac{1}{\tau} |u'|_{L^2}^2 + |M \nabla u'|_{L^2}^2 + \frac{1}{\tau} \int_{\Omega} \langle F(\nabla u) \nabla u, \nabla u' \rangle = 0 \quad (4.30)$$

d'où

$$\frac{d}{dt} |u'|_{L^2}^2 + \frac{2}{\tau} |u'|_{L^2}^2 \leq -\frac{1}{2\tau} \int_{\Omega} \langle F(\nabla u) \nabla u, \nabla u' \rangle$$

le théorème de Gronwall nous donne:

$$|u'|_{L^2}^2 \leq |u'(0)|_{L^2}^2 \exp\left(\frac{-2t}{\tau}\right) + \frac{1}{2\tau} \exp\left(\frac{-2t}{\tau}\right) \int_{\Omega} \int \exp\left(\frac{2s}{\tau}\right) \frac{d}{ds} \langle F(\nabla u) \nabla u, \nabla u' \rangle ds$$

Une intégration par partie de cette dernière équation amène :

$$\begin{aligned} |u'(t)|_{L^2}^2 &\leq \exp\left(\frac{-2t}{\tau}\right) |u'(0)|_{L^2}^2 \\ &+ \frac{1}{2\tau} \int_{\Omega} \left[ \langle F(\nabla u) \nabla u(t), \nabla u'(t) \rangle - \langle F(\nabla u) \nabla u(0), \nabla u'(0) \rangle \right] \\ &+ \frac{1}{\tau} \int_{\Omega} \int_0^t \exp\left(\frac{2(s-t)}{\tau}\right) \langle F(\nabla u) \nabla u(s), \nabla u'(s) \rangle ds \end{aligned}$$

- Pour  $|\nabla u| > s$ , on a  $F(\nabla u) \nabla u = 0$ . La dernière inégalité se réécrit comme suit:

$$|u'|^2 \leq \exp\left(\frac{-2t}{\tau}\right) |u'(0)|_{L^2}^2$$

Si  $u'(0)$  est dans  $L^2(\Omega)$  alors  $\Delta u(0)$  est dans  $L^2(\Omega)$ ; on peut conclure que :

$$u' \text{ est borné dans } L^\infty(0, T, L^2(\Omega)) \cap L^2(0, T, H^1(\Omega))$$

- Pour  $|\nabla u| < s$ , on a  $F(\nabla u) \nabla u = \frac{3}{2} s^2 \left(1 - \frac{|\nabla u|^2}{s^2}\right) \nabla u$ .

or

$$\begin{aligned} \int_{\Omega} \left[1 - \frac{|\nabla u|^2}{s^2}\right] \langle \nabla u, \nabla(u') \rangle &= \frac{1}{2} \int_{\Omega} \left[1 - \frac{|\nabla u|^2}{s^2}\right] \frac{d}{dt} |\nabla u|^2 \\ &= -\frac{1}{4} \int_{\Omega} \frac{d}{dt} \left(\left[1 - \frac{|\nabla u|^2}{s^2}\right]\right)^2 \end{aligned}$$

donc en remplaçant dans l'inégalité précédente on aura :

$$\begin{aligned} \left|\frac{du}{dt}(t)\right|_{L^2}^2 &\leq \exp\left(\frac{-2t}{\tau}\right) \left|\frac{du}{dt}(0)\right|_{L^2}^2 + \frac{3}{4\tau} s^2 \left|1 - \frac{|\nabla u(t)|^2}{s^2}\right|_{L^2}^2 \\ &+ \frac{3}{2\tau} \int \exp\left(\frac{2(r-t)}{\tau}\right) s^2 \left|1 - \frac{|\nabla u(r)|^2}{s^2}\right|_{L^2}^2 dr \end{aligned}$$

donc

$$\left|\frac{du}{dt}(t)\right|_{L^2}^2 \leq \exp\left(\frac{-2t}{\tau}\right) \left|\frac{du}{dt}(0)\right|_{L^2}^2 + \frac{3s^2}{2} |\Omega| \left(\frac{1}{2\tau} + 1 - \exp\left(\frac{-2t}{\tau}\right)\right)$$

finalment on a :

$$\overline{\lim}_{t \rightarrow \infty} \left|\frac{du}{dt}(t)\right|_{L^2}^2 \leq \frac{3s^2}{2} |\Omega| \left(\frac{1}{2\tau} + 1\right).$$

■

## 4.2 Résolution numérique

Pour une étude numérique nous allons discrétiser le système suivant:

$$\frac{\partial u}{\partial t} - \operatorname{div}(L[\nabla u]) = 0 \quad (4.31)$$

$$\frac{dL}{dt} + \frac{1}{\tau}L = \frac{1}{\tau} \begin{cases} \mathbb{P}_{\nabla u^\perp} & \text{si } |\nabla u| > s \\ \frac{|\nabla u|^2}{s^2} \mathbb{P}_{\nabla u^\perp} + \frac{3}{2} s^2 \left(1 - \frac{|\nabla u|^2}{s^2}\right) Id & \text{si } |\nabla u| \leq s \end{cases} \quad (4.32)$$

Dans la partie application,  $N \times N$  représente la taille de l'image. La discrétisation en espace est faite avec un pas  $h = 1/N$  et celle en temps est régulière avec un pas de temps  $\Delta t$ .

Soit  $u_{ij}^n$  une approximation de  $u$  au point  $(ih, jh)$  (avec  $0 \leq i \leq N$  et  $0 \leq j \leq N$ ) à l'instant  $t = n\Delta t$ . La matrice de diffusion  $L$  est approchée par:

$$L_h^n = \begin{pmatrix} L_{(ij)xx}^n & L_{(ij)xy}^n \\ L_{(ij)xy}^n & L_{(ij)yy}^n \end{pmatrix} \quad (4.33)$$

La divergence et le gradient sont discrétisés avec des décentrages différents comme suit:

$$\operatorname{div}_h(u_1, u_2) = h^{-1}(\Delta_+^x u_1 + \Delta_+^y u_2) \quad (4.34)$$

et

$$\nabla_h u = h^{-1}(\Delta_-^x u, \Delta_-^y u) \quad (4.35)$$

où

$$\Delta_+^x u_{i,j} = u_{i+1,j} - u_{i,j} \quad , \quad \Delta_-^x u_{i,j} = u_{i,j} - u_{i-1,j},$$

pour  $\Delta_+^y$  et  $\Delta_-^y$ , on utilise la même discrétisation en intervertissant les rôles de  $i$  et  $j$ .

Avec cette discrétisation, une approximation de  $L(\nabla u)$  est la suivante:

$$L_h(\nabla u_h) = h^{-1} \left( L_{(ij)xx}^n \Delta_-^x u_{ij}^n + L_{(ij)xy}^n \Delta_-^y u_{ij}^n, L_{(ij)yx}^n \Delta_-^x u_{ij}^n + L_{(ij)yy}^n \Delta_-^y u_{ij}^n \right) \quad (4.36)$$

et  $\operatorname{div}(L(\nabla u))$  est approchée par:

$$h^{-2} [\Delta_+^x (L_{(ij)xx}^n \Delta_-^x u_{ij}^n + L_{(ij)xy}^n \Delta_-^y u_{ij}^n) + \Delta_+^y (L_{(ij)yx}^n \Delta_-^x u_{ij}^n + L_{(ij)yy}^n \Delta_-^y u_{ij}^n)]. \quad (4.37)$$

## Schéma mixte

Pour la résolution numérique, nous avons choisi un schéma mixte:

- Schéma implicite pour l'équation (3.32):

$$L_{ij}^{n+1} = \frac{\tau}{\tau + \Delta t} L_{ij}^n + \frac{\Delta t}{\tau + \Delta t} F_{ij}^n \quad (4.38)$$

où  $F_{ij}^n$  est une approximation à l'instant  $t = n\Delta t$  au point  $(ih, jh)$  du membre droit de (3.32) donnée par :

$$F_{ij}^n = \begin{cases} (\mathbf{P}_{\nabla u^\perp})_{ij}^n & \text{si } |\nabla u^n| > s \\ \frac{|\nabla u^n|^2}{s^2} (\mathbf{P}_{\nabla u^\perp})_{ij}^n + \frac{3}{2} s^2 \left(1 - \frac{|\nabla u^n|^2}{s^2}\right) \delta_{ij} & \text{si } |\nabla u^n| \leq s \end{cases} \quad (4.39)$$

Comme on l'a vu au début de ce chapitre, nous avons  $F_{ij}^n$ , une matrice définie positive  $\forall i, j$ . L'équation (3.38) assure la positivité de  $L$  qui permet d'éviter l'antidiffusion dans l'équation (3.31).

- Schéma explicite pour l'équation (3.31):

$$\begin{aligned} \frac{u_{ij}^{n+1} - u_{ij}^n}{\Delta t} - h^{-2} [\Delta_+^x (L_{(ij)xx}^n \Delta_-^x u_{ij}^n + L_{(ij)xy}^n \Delta_-^y u_{ij}^n) + \\ + \Delta_+^y (L_{(ij)yx}^n \Delta_-^x u_{ij}^n + L_{(ij)yy}^n \Delta_-^y u_{ij}^n)] = 0 \end{aligned}$$

Ceci nous donne l'algorithme suivant:

$$(Pd) \left\{ \begin{aligned} u_{ij}^{n+1} &= u_{ij}^n \left(1 - \frac{\Delta t}{h^2} (L_{(i+1j)xx}^n + L_{(ij)xx}^n + L_{(ij)xy}^n + L_{(ij+1)yy}^n + L_{(ij)yx}^n \right. \\ &+ L_{(ij)yy}^n) + \frac{1}{h^2} [L_{(i+1j)xx}^n u_{i+1j}^n + L_{(i+1j)xy}^n u_{i+1j}^n + L_{(ij)xx}^n u_{i-1j}^n \\ &+ L_{(ij)xy}^n u_{ij-1}^n + L_{(ij+1)yx}^n u_{ij+1}^n + L_{(ij+1)yy}^n u_{ij+1}^n + L_{(ij)yx}^n u_{i-1j}^n \\ &+ L_{(ij)yy}^n u_{ij-1}^n - L_{(ij+1)yx}^n u_{i-1j+1}^n - L_{(i+1j)xy}^n u_{i+1j-1}^n] \Big) \\ u_{ij}^0 &= u_0(ih, jh) \quad , \quad 0 \leq i \leq N \quad \text{et} \quad 0 \leq j \leq N \\ L_{ij}^{n+1} &= \frac{\tau}{\tau + \Delta t} L_{ij}^n + \frac{\Delta t}{\tau + \Delta t} F_{ij}^n \end{aligned} \right. \quad (4.40)$$

C'est un système fortement anisotrope qui peut être réécrit comme suit:

$$\begin{cases} u^{n+1} &= [Id + \Delta t h^2 A(u^n)] u^n \\ L^{n+1} &= \frac{\tau}{\tau + \Delta t} L^n + \frac{\Delta t}{\tau + \Delta t} F^n \end{cases} \quad (4.41)$$

**Proposition 3** (*Principe du maximum*)

Soient  $u_{ij}^n$  le niveau de gris de l'image et  $L_{(ij)xy}^n$  une composante de la matrice de diffusion à l'instant  $t = n \cdot \Delta t$ . Alors,

si

$$\frac{\Delta t}{h^2} \max_{ij} |L_{ij}| \leq \frac{1}{6} \quad \text{et} \quad m \leq u_{ij}^0 \leq M \quad \forall i, j \quad (4.42)$$

alors

$$m \leq u_{ij}^n \leq M \quad \forall i, j$$

**Preuve:**

Supposons que:

$$\frac{\Delta t}{h^2} \max_{ij} |L_{ij}| \leq \frac{1}{6}$$

alors (4.1) nous permet d'écrire:

$$\begin{aligned} \max_{ij} |u_{ij}^{n+1}| &\leq \max_{ij} |u_{ij}^n| \cdot \left(1 - \frac{6\Delta t}{h^2} \max_{ij} |L_{ij}|\right) + \frac{\Delta t}{h^2} \left(6 \cdot \max_{ij} |u_{ij}^n| \max_{ij} |L_{ij}|\right) \\ &\leq \max_{ij} |u_{ij}^n| \end{aligned}$$

Donc (3.42) est une condition suffisante pour que le principe du maximum soit satisfait.

**Schéma implicite**

Dans ce cas nous utilisons un schéma implicite pour l'équation (3.31):

$$\begin{aligned} &\frac{u_{ij}^{n+1} - u_{ij}^n}{\Delta t} - \frac{1}{h^2} \left[ (L_{(ij)xx}^n + L_{(ij)xy}^n) u_{i-1j}^{n+1} + (L_{(ij)xy}^n + L_{(ij)yy}^n) u_{ij-1}^{n+1} \right. \\ &\quad + (L_{(i+1j)xy}^n + L_{(i+1j)yx}^n) u_{i+1j}^{n+1} + (L_{(ij+1)xy}^n + L_{(ij+1)yy}^n) u_{ij+1}^{n+1} \\ &\quad + (L_{(i+1j)xy}^n + L_{(i+1j)yx}^n) u_{i+1j}^{n+1} + (L_{(ij+1)xy}^n + L_{(ij+1)yy}^n) u_{ij+1}^{n+1} \\ &\quad \left. - L_{(i+1j)xy}^n u_{i+1j-1}^{n+1} - L_{(ij+1)yx}^n u_{i-1j+1}^{n+1} \right. \\ &\quad \left. - (L_{(i+1j)xx}^n + L_{(ij)xx}^n + 2L_{(ij)xy}^n + L_{(ij+1)yy}^n + L_{(ij)yy}^n) u_{ij}^{n+1} \right] = 0 \end{aligned} \quad (4.43)$$

Cette équation peut être réécrite comme suit:

$$\frac{u_{ij}^{n+1} - u_{ij}^n}{\Delta t} - \frac{1}{h^2} \left[ a^n u_{i-1j}^{n+1} + b^n u_{ij-1}^{n+1} + c^n u_{i+1j}^{n+1} + d^n u_{ij+1}^{n+1} + \tau^n u_{i+1j-1}^{n+1} + \beta^n u_{i-1j+1}^{n+1} + e^n u_{ij}^{n+1} \right] = 0$$

Finalement, utilisant une autre numérotation, ce problème discret peut être réécrit sous la forme:

$$\begin{cases} \frac{u^{n+1} - u^n}{\Delta t} + A_{\Delta t}(L^n)u^{n+1} = 0 \\ L^{n+1} = \left( \frac{\tau}{\tau + \Delta t} L^n + \frac{\Delta t}{\tau + \Delta t} F^n \right) \end{cases} \quad (4.44)$$

où

$$A_h(L^n) = \begin{pmatrix} e^n & c^n & . & . & . & b^n & \alpha^n & . & . & . & . & . & . \\ a^n & e^n & c^n & . & . & . & b^n & \alpha^n & . & . & . & . & . \\ . & a^n & e^n & c^n & . & . & . & b^n & \alpha^n & . & . & . & . \\ . & . & a^n & e^n & c^n & . & . & . & b^n & . & . & . & . \\ d^n & . & . & a^n & e^n & c^n & . & . & . & b^n & . & . & . \\ \alpha^n & d^n & . & . & . & a^n & e^n & c^n & . & . & . & b^n & \beta^n \\ . & \alpha^n & d^n & . & . & . & a^n & e^n & c^n & . & . & . & b^n \\ . & . & . & d^n & . & . & . & a^n & e^n & c^n & . & . & . \\ . & . & . & \alpha^n & d^n & . & . & . & a^n & e^n & c^n & . & . \\ . & . & . & . & \alpha^n & d^n & . & . & . & a^n & e^n & c^n & . \\ . & . & . & . & . & \alpha^n & d^n & . & . & . & a^n & e^n & c^n \\ . & . & . & . & . & . & \alpha^n & d^n & . & . & . & a^n & e^n \end{pmatrix}$$

Avec cette numérotation particulière on obtient une matrice bande tridiagonale par bloc. On pourra alors résoudre efficacement le problème discret (3.44) à l'aide d'une méthode directe comme par exemple celle de Cholesky qui préserve la structure bande de la matrice. Dans le paragraphe suivant nous utiliserons l'algorithme (4.1) puisqu'il est moins coûteux en temps de calcul.



# Chapitre 5

## Exemples et Applications

LES modèles de diffusion linéaire, non linéaire ou anisotrope sont généralement utilisés comme une étape intermédiaire préparant l'image à la segmentation. Ce chapitre est consacré à l'application des modèles de diffusion anisotrope et des méthodes de réseaux de neurones décrits dans les chapitres précédents

### 5.1 Diffusion anisotrope

#### 5.1.1 Algorithme

Dans cette partie nous étudierons différents cas d'application du filtre fondé sur l'équation de type Volterra (3.11). Pour cela rappelons l'algorithme détaillé dans le chapitre 3 :

$$(P_d) \quad \begin{cases} u^{n+1} &= [Id + \Delta t h^2 A(u^n)] u^n \\ L_{ij}^{n+1} &= \left(\frac{k}{1+k}\right) \left(\frac{1}{k} F_{ij}^n + L_{ij}^n\right) \end{cases} \quad (5.1)$$

où  $k$  est un paramètre multiplicatif donnant le temps de relaxation par la relation  $\tau = k\Delta t$  et  $F_{ij}^n$  une approximation du membre de droite de l'équation d'apprentissage :

$$F_{ij}^n = \begin{cases} (P_{\nabla u^\perp})_{ij}^n & \text{si } |\nabla u^n| > s \\ |\nabla u^n|^2 (P_{\nabla u^\perp})_{ij}^n + \frac{3}{2}(s^2 - |\nabla u^n|^2)\delta_{ij} & \text{si } |\nabla u^n| \leq s \end{cases} \quad (5.2)$$

où nous rappelons que  $s$  est un paramètre de contraste qui permet de sélectionner les états stationnaires.

Cet algorithme est utilisé de façon itérative jusqu'à ce que le résidu (ou la différence en norme  $L^2$  des solutions correspondantes à deux itérations successives) tombe en dessous d'un certain seuil prédéterminé. Dans nos applications nous prendrons comme valeur de ce seuil typiquement  $10^{-4}$ .

Nous montrerons, dans la partie application que cet algorithme fortement anisotrope permet d'éliminer le bruit, de préserver les bords et les points anguleux, et même de réhausser les contrastes des bords flous.

### 5.1.2 Images tests

Dans la première application (Fig.4.1), l'image exacte que nous voulons reconstituer consiste en un disque noir sur un fond gris. L'image de départ n'étant pas bruitée et étant constituée de plages de niveaux de gris homogènes, c'est la condition initiale qui doit être idéalement préservée par notre traitement. Nous avons traité cette image en utilisant 100 itérations de l'algorithme (4.1) et différentes valeurs du temps de relaxation  $\tau$  et du seuil  $s$ :  $\tau = 5\Delta t$   $s = 5$  pour (a),  $\tau = 5\Delta t$   $s = 10$  pour (b), et  $\tau = 10\Delta t$   $s = 10$  pour l'image (c). Notons aussi qu'un seuillage de ces images finales montre que l'image originale n'a pas changé, ceci étant possible puisque celle-ci n'est pas bruitée.

Le temps de filtrage d'une image  $256 \times 256$ , par notre code de calcul, sur une station de travail Sun Ultra 1, est de deux itérations par seconde.

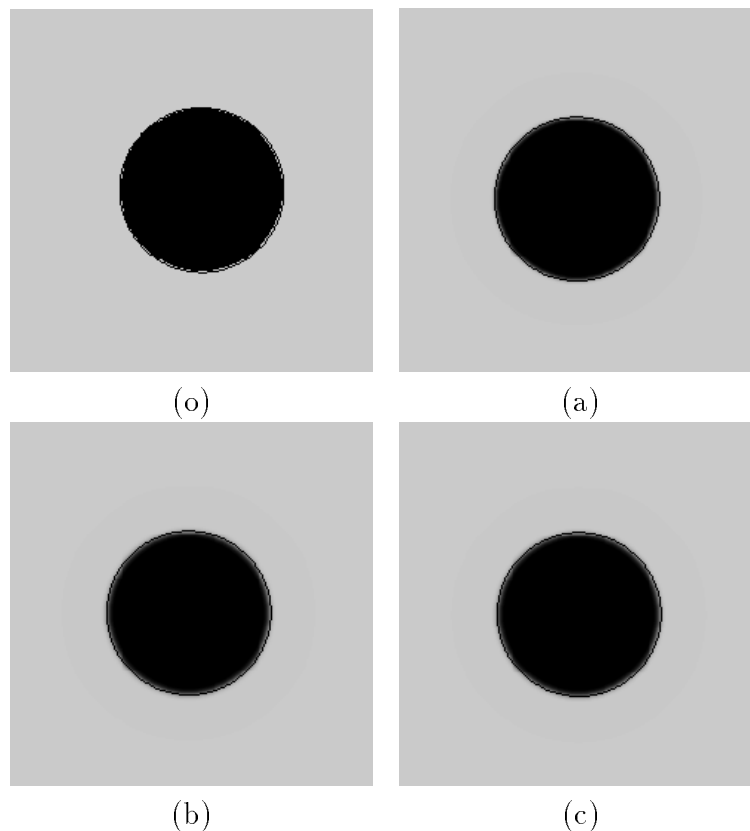


FIG. 5.1 – Diffusion anisotrope d'un disque ( $256 \times 256$ ). (o) : l'image originale et l'image diffusée avec (a) :  $\tau = 5\Delta t$  et  $s = 5$  (b) :  $\tau = 10\Delta t$  et  $s = 5$  (c)  $\tau = 10\Delta t$  et  $s = 10$ .

Ensuite, nous mesurons pour différentes valeurs de  $\tau$  et  $s$ , par la norme  $L^2$ , la distance entre cette solution exacte (l'image originale (4.1.o) et la solution calculée (l'image traitée) ainsi que la distance entre les solutions correspondante à deux itérations successives (résidu). La courbe A correspond à  $\tau = 5\Delta t$   $s = 5$ , B à  $\tau = 5\Delta t$   $s = 10$  et C à  $\tau = 10\Delta t$   $s = 10$ . Cette expérience montre que lorsque l'image de départ n'est pas bruitée, pour une même valeur de  $\tau$  l'erreur n'est pas très sensible au choix du seuil  $s$  (les courbes B et C de la figure 4.2.a). Par contre pour une même valeur de  $s$  l'erreur devient sensible au choix du temps de relaxation  $\tau$  (les courbes A et B de la figure 4.2.a).

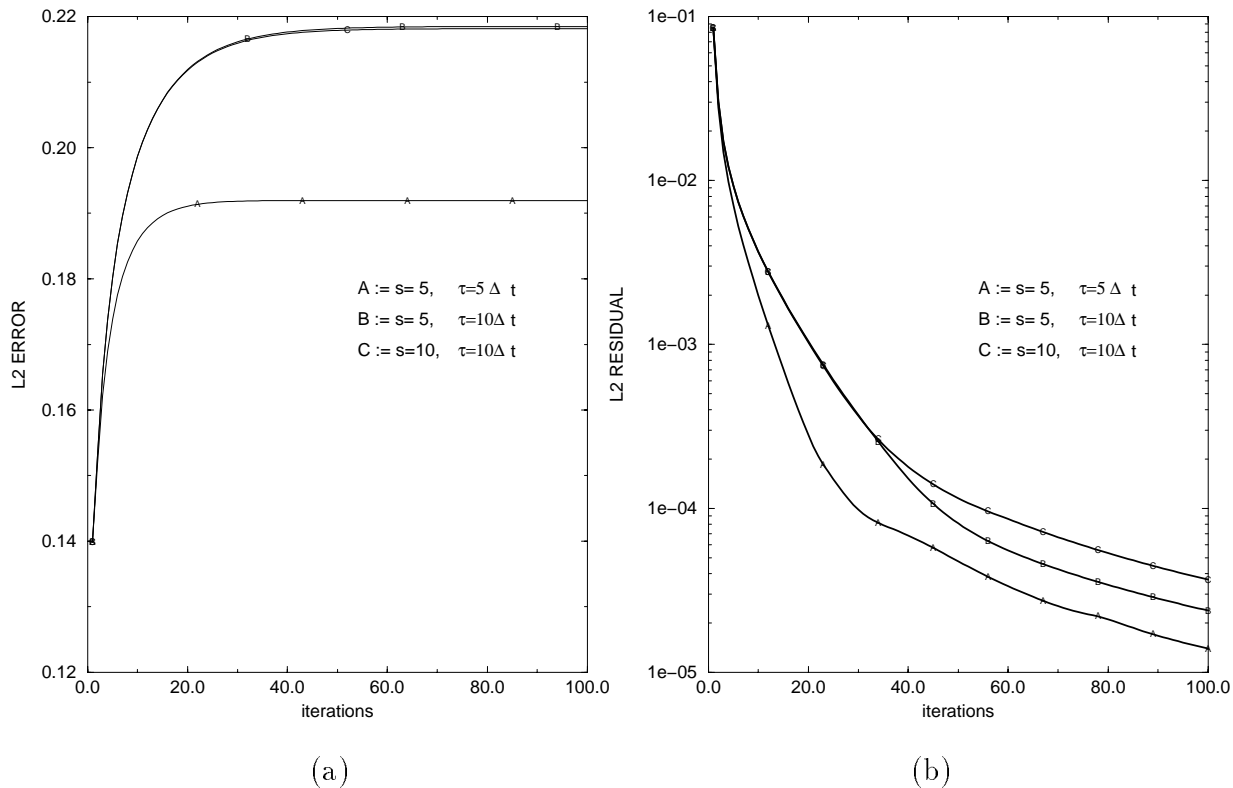


FIG. 5.2 – Courbes d'erreur (a) et du résidu (b) correspondantes aux images 4.1.

La deuxième expérience traite une image synthétique contenant un triangle noir au dessus d'un rectangle noir étroit sur un fond blanc (l'image ayant une taille de  $128 \times 128$ ), auquel nous avons ajouté différents niveaux de bruit en détruisant un certain pourcentage de pixels et en remplaçant le niveau de gris des pixels détruits par des valeurs aléatoires. Nous étudierons ensuite les courbes d'erreur et du résidu en fonction des paramètres  $\tau$ ,  $s$  et du pourcentage du bruit, ce qui nous donnera une idée sur le choix des paramètres pour la suite des applications.

Dans la figure (4.3) nous avons détruit l'image originale à 30% puis traité en fai-

sant varier séparément les valeurs de  $\tau$  et de  $s$ . Après un seuillage en noir et blanc des images diffusées nous retrouvons à quelques pixels près, les mêmes résultats. Une première conclusion est donc que, si le taux de dégradation de l'image originale est faible alors la sensibilité de l'image filtrée au choix des paramètres  $\tau$  et  $s$  l'est aussi.

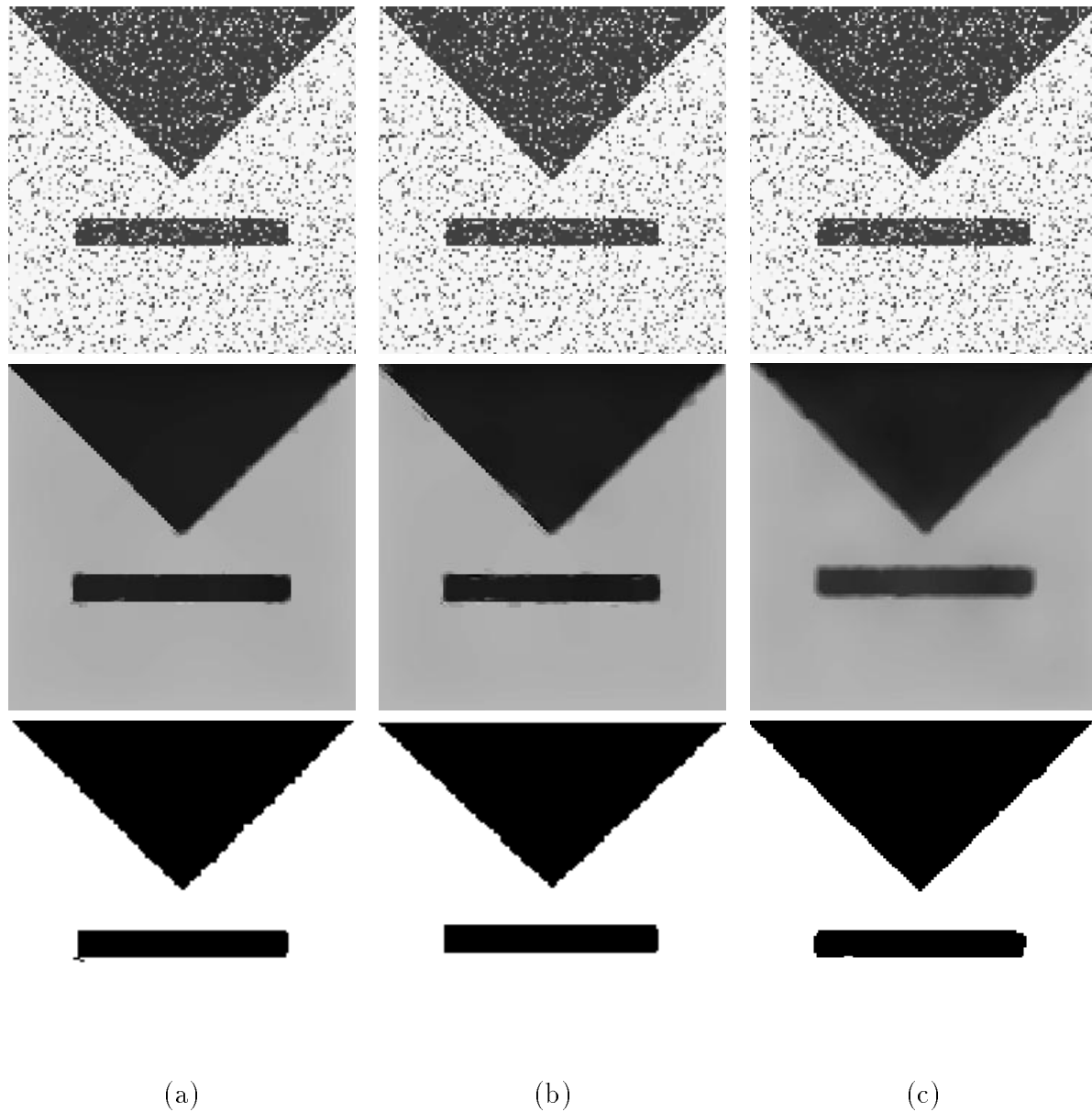
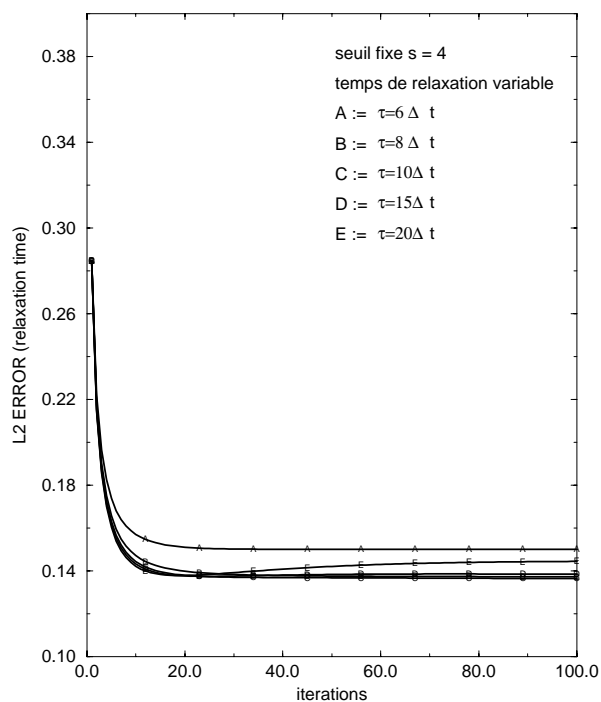
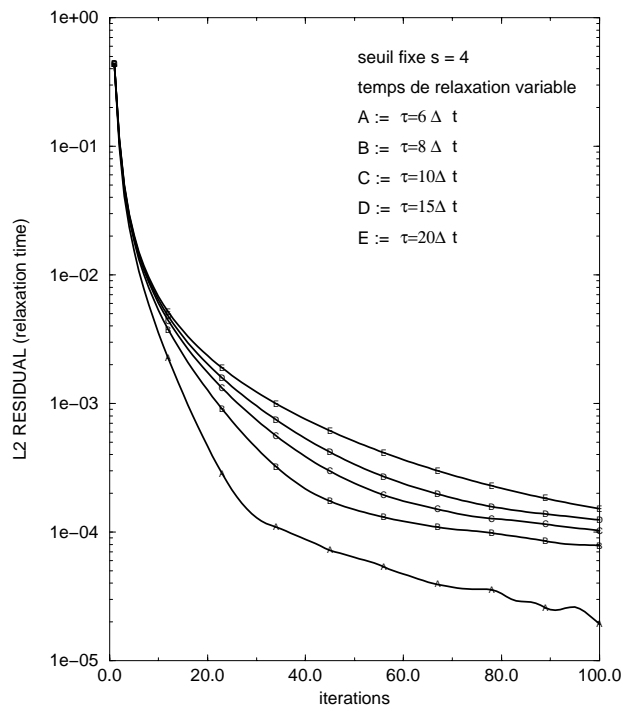


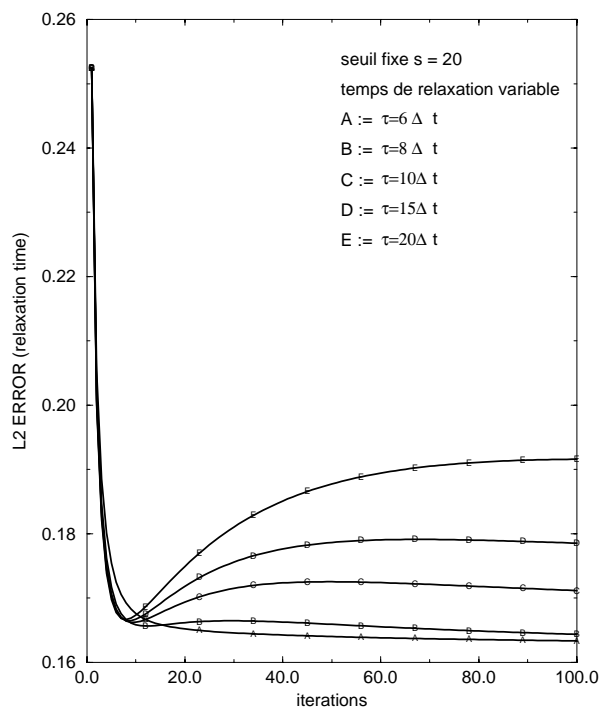
FIG. 5.3 – Application de la diffusion anisotrope à paramètres variables sur une image triangle – rectangle  $128 \times 128$  bruitée à 30%. (a) :  $\tau = 3\Delta t$  et  $s = 4$  (b) :  $\tau = 3\Delta t$  et  $s = 10$  (c) :  $\tau = 10\Delta t$  et  $s = 10$ . Les lignes correspondent à l'image originale, aux images restaurées et au seuillage en noir et blanc de ces dernières.



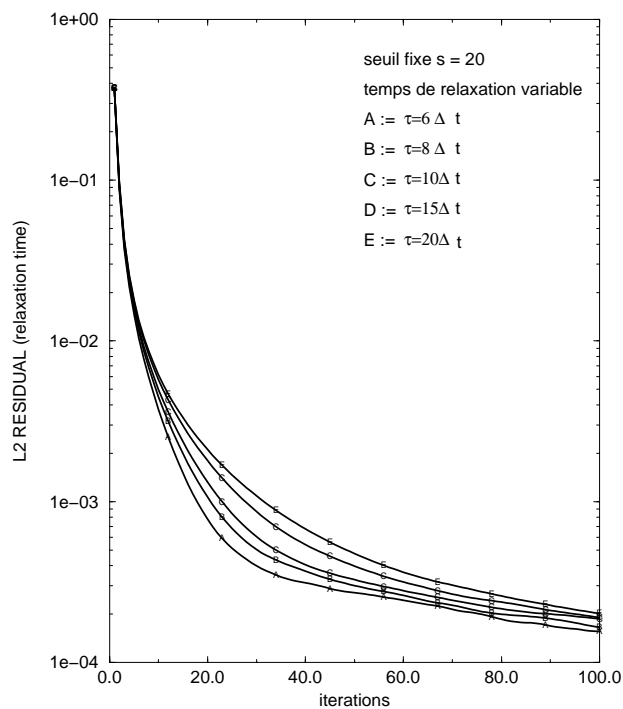
(a)



(b)

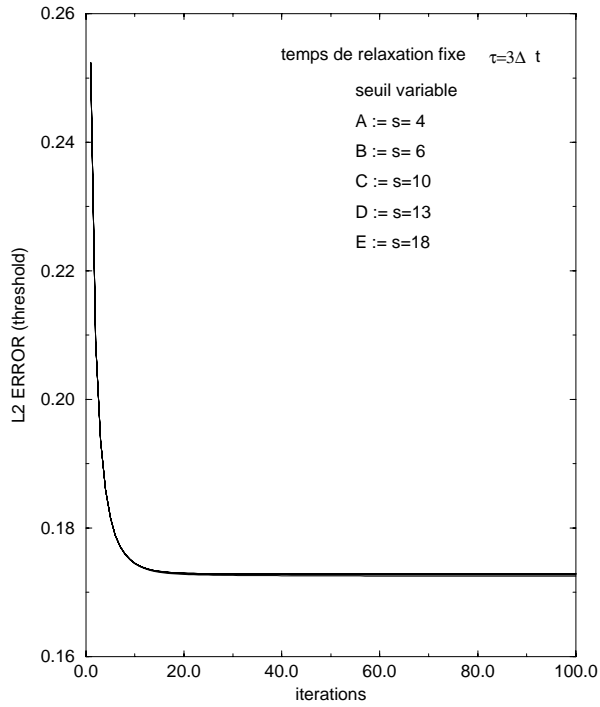


(c)

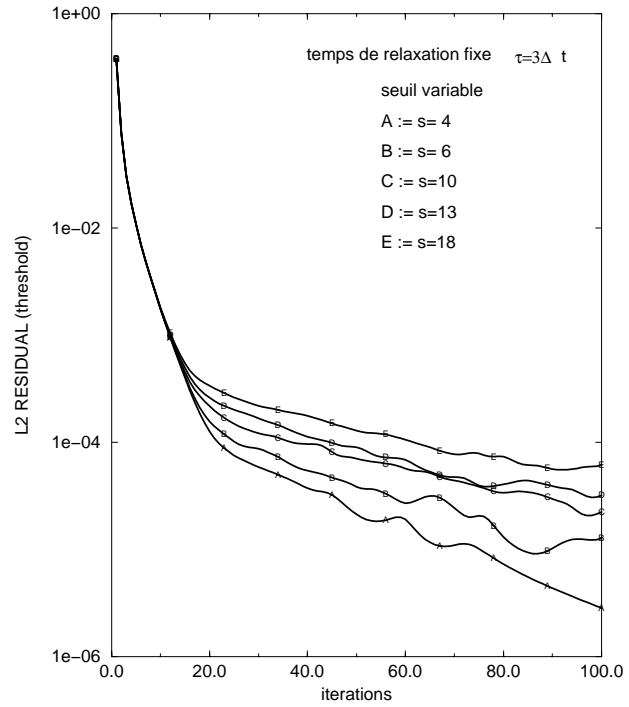


(d)

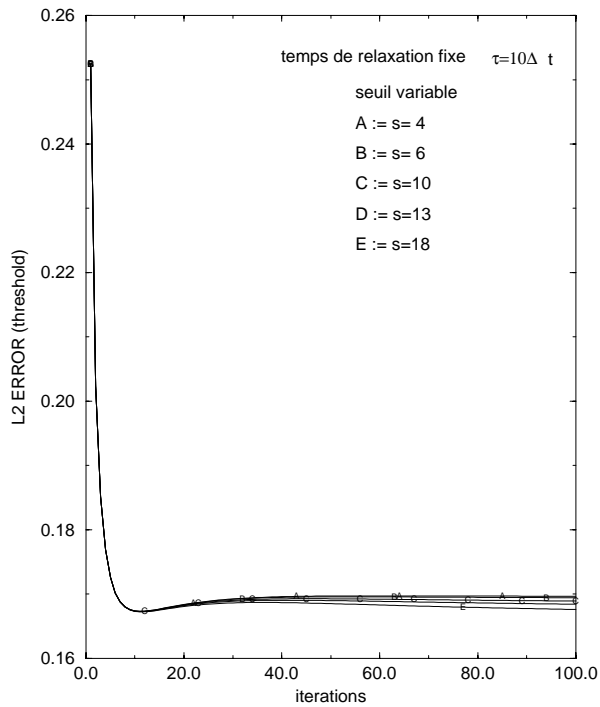
FIG. 5.4 – Courbes d'erreur et du résidu pour différentes valeurs de  $\tau$  avec un seuil fixé pour chaque graphe.



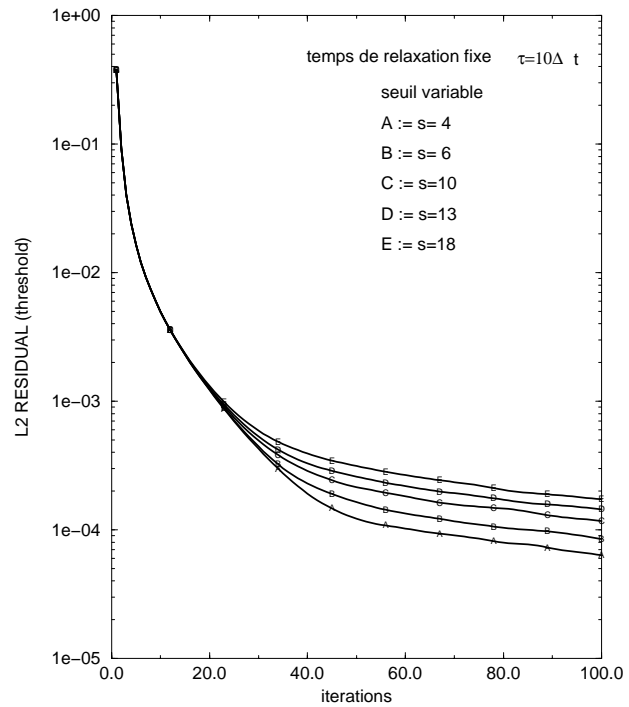
(a)



(b)

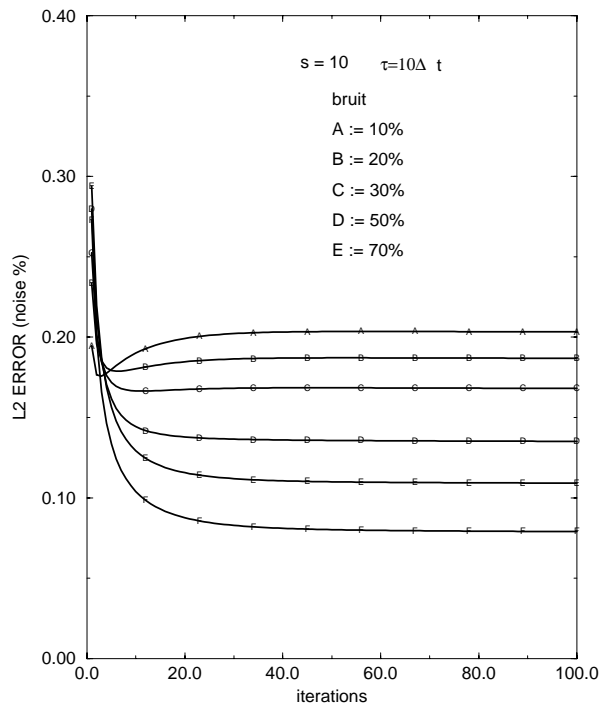


(c)

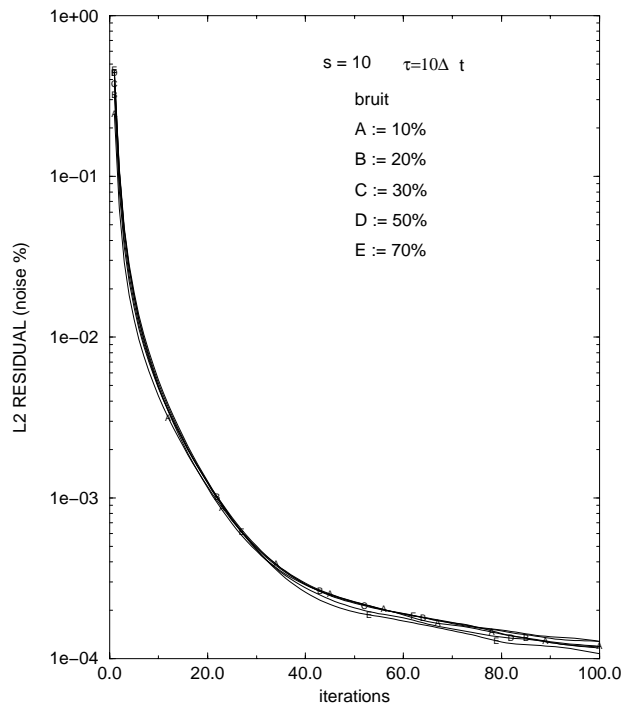


(d)

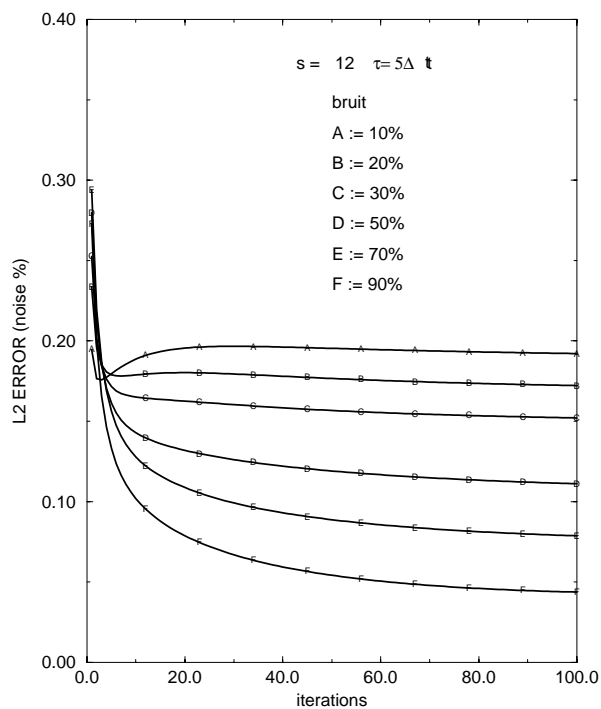
FIG. 5.5 – Courbes d'erreur et du résidu pour différentes valeurs du seuil  $s$  avec un temps de relaxation  $\tau$  fixé pour chaque graphe.



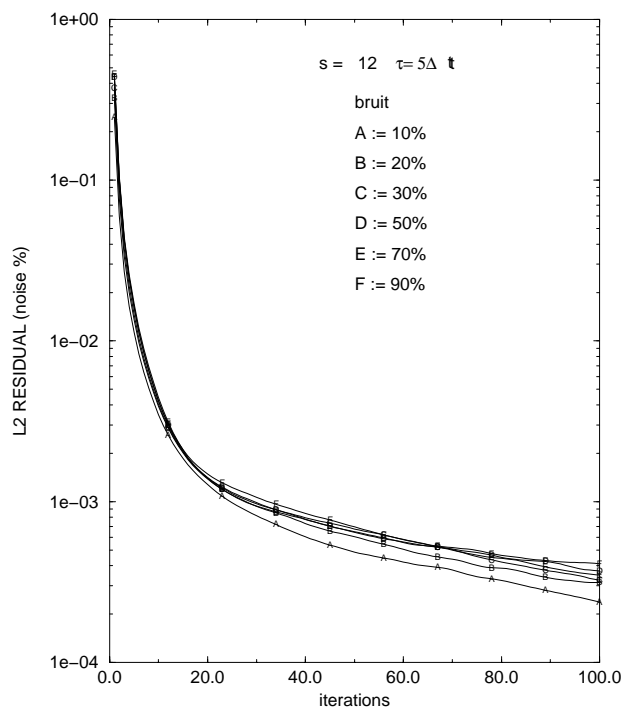
(a)



(b)



(c)



(d)

FIG. 5.6 – Étude de l'erreur et du résidu en fonction du bruit introduit dans l'image originale avec un seuil  $s$  et un temps de relaxation  $\tau$  fixés pour chaque graphe.

Dans la deuxième application (Fig.4.4 et 4.5), nous avons calculé l'erreur et le résidu en fixant un paramètre et en faisant varier l'autre. Nous rappelons que l'erreur a été calculée par rapport à une image où le noir et le blanc sont remplacés par des valeurs proportionnelles au niveau de bruit (i.e. la moyenne statistique des niveaux de gris effectivement dans l'image). Cette expérience traite l'image du *triangle-rectangle* bruitée à 30% (Fig.4.3(a)). Dans la figure (4.4) nous avons fixé le seuil  $s$  ( $s = 4$  pour (a) et (b)) et ( $s = 20$  pour (c) et (d)). Les courbes  $A, B, C, D$  et  $E$  correspondent à  $\tau = 6, 8, 10, 15, 20\Delta t$  respectivement. Par contre, dans la figure (4.5), nous avons fixé le paramètre de relaxation  $\tau$  ( $\tau = 3\Delta t$  pour (a) et (b)) et ( $\tau = 10\Delta t$  pour (c) et (d)). Les courbes  $A, B, C, D$  et  $E$  correspondent à  $s = 4, 6, 10, 13$  et  $18$  respectivement. Nous pouvons remarquer que le choix d'une grande valeur de  $\tau$  permet un débruitage efficace et une "convergence" vers l'image exacte en seulement dix itérations (Fig.4.4). Par la suite, la diffusion conduit à une perte des angles, qui se traduit par une remontée de la courbe d'erreurs. Notons aussi que lorsque le temps de relaxation  $\tau$  est fixe, les courbes d'erreur sont insensibles au choix du seuil  $s$  même pour un niveau de bruit de 30% (Fig.4.5). Ceci peut s'expliquer par le fait que l'image originale n'est pas très bruitée et que les grandes valeurs de  $\tau$  et de  $s$  favorisent le filtrage Gaussien d'où une dépendance de la condition initiale  $L_0 = Id$ ; nous verrons dans le paragraphe suivant comment nous pouvons surmonter cette difficulté. Les petites valeurs de  $\tau$  et  $s$  permettent une "convergence" lente et l'image filtrée est sur une asymptotique du modèle proposé.

Dans la figure (Fig.4.6), nous avons calculé l'erreur et le résidu pour différentes valeurs du niveau de dégradation de la même image (Fig.4.3(a)), nous avons pris ( $s = 10$  et  $\tau = 10\Delta t$  pour (a) et (b)) et ( $s = 5$  et  $\tau = 5\Delta t$  pour (c) et (d)). Les courbes  $A, B, C, D$  et  $E$  correspondent à 10%, 20%, 30%, 50%, 70% et 90% respectivement. Les courbes d'erreur montrent que plus le bruit est fort et plus la valeur de  $\tau$  doit être grande.

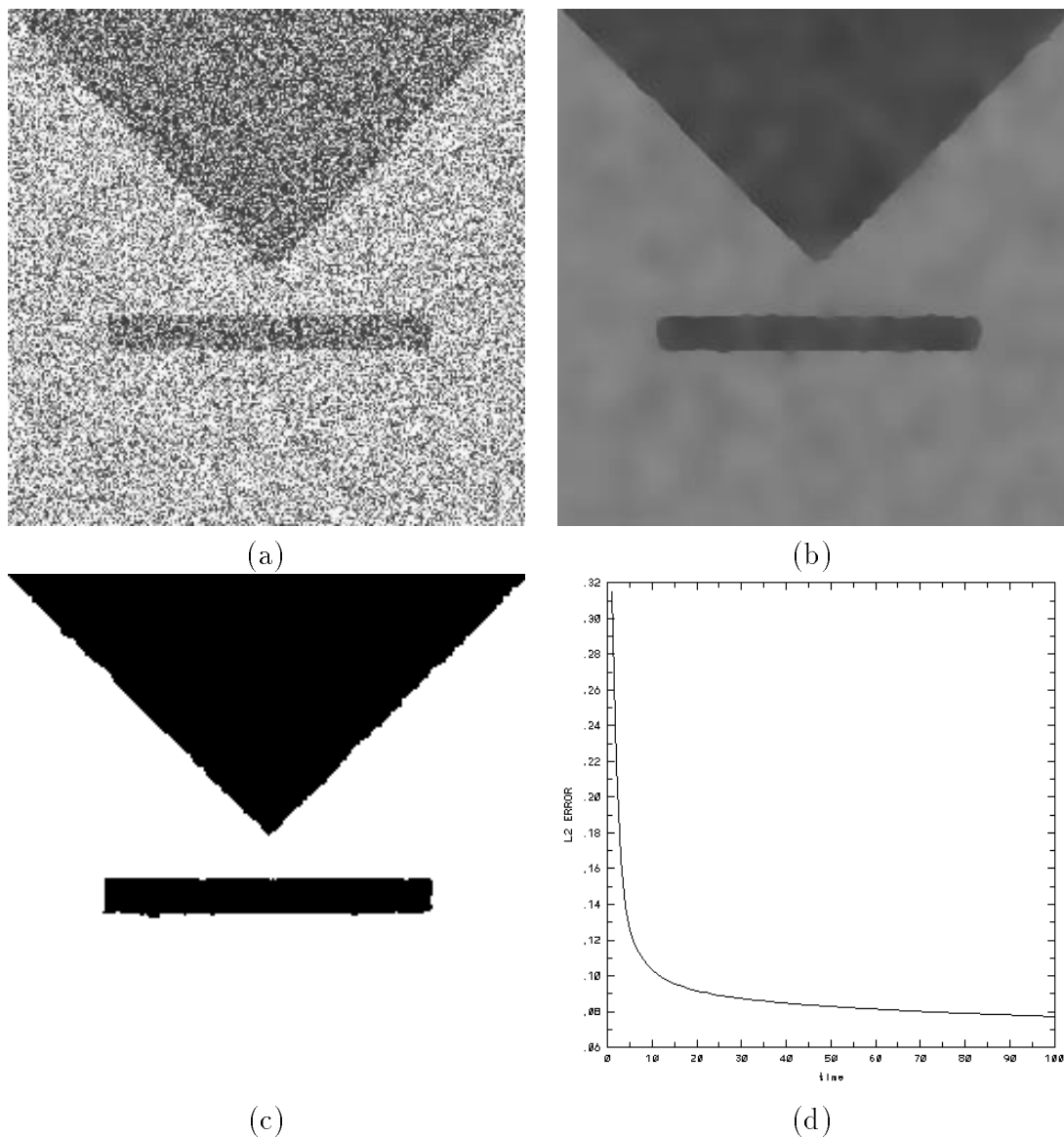


FIG. 5.7 – Diffusion anisotrope d’une image triangle – rectangle  $256 \times 256$  bruitée à 70%. (a) : image originale (b) : image filtrée avec  $\tau = 5\Delta t$  et  $s = 5$  (c) : seuillage de l’image traitée (d) : Courbe d’erreur.

Dans cette dernière application (Fig.4.7), nous avons bruité l’image *triangle-rectangle* à 70%. Pour avoir une ”convergence” rapide de l’algorithme présenté, en se basant sur le courbe ( $E$ ) de la figure (Fig.4.6 (a) et (b)), nous avons pris un temps de relaxation sur dix pas de temps et un seuil  $s = 10$ . Un seuillage de l’image diffusée montre que ce filtre permet l’élimination du bruit et le renforcement des contrastes tout en préservant les bords et les angles. Cette dernière propriété est très intéressante dans le sens où elle permet un lissage intra-région plus continu.

### 5.1.3 Images Réelles

Notre deuxième application traite une image fortement texturée: Lenna ( $256 \times 256$ ). C'est une image standard «*benchmark*» d'extraction des contours qui a été largement utilisée en traitement d'images. La difficulté initiale est de segmenter l'épaule qui a un niveau de gris se confondant avec celui du visage. Une visualisation de cette image à l'aide d'une palette de couleurs aléatoires nous permet de voir le bruit impulsionnel assez intense sur l'image originale (Fig.4.8(c)). Nous avons lissé cette image avec un temps de relaxation ( $\tau = 10\Delta t$ ), un seuil  $s = 10$  et 50 itérations (Fig.4.8(b)). Visualisée à l'aide de la même palette de couleurs aléatoires (d), cette image met en évidence la meilleure qualité de lissage du filtre développé dans le chapitre 2. Les différentes zones d'isoluminescence (ou zones homogènes) qui étaient impossibles à détecter sur l'image originale sont maintenant nettement visibles (Fig.4.8(d)). Nous verrons dans le paragraphe suivant que la réinitialisation de notre algorithme permet d'améliorer encore l'image traitée.

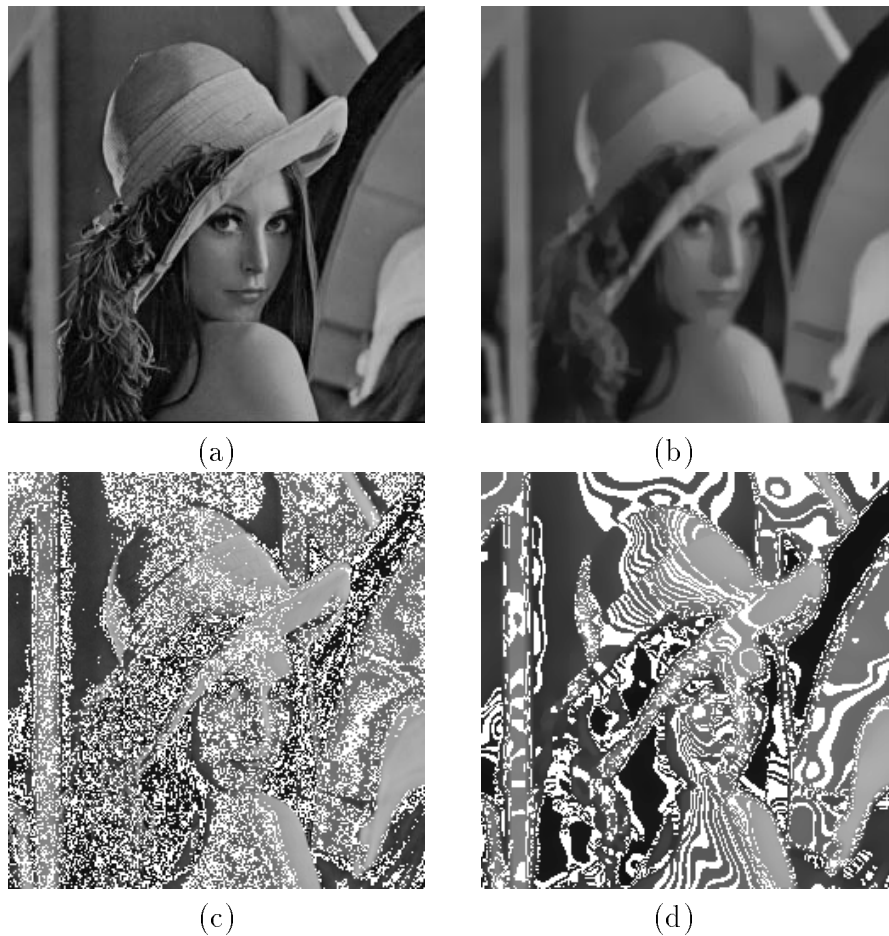


FIG. 5.8 – Application de l'algorithme sur une image réelle. (a) : l'image originale (b) : l'image diffusée (c) et (d) : Visualisation de ces deux images à l'aide d'une palette de couleurs aléatoires.

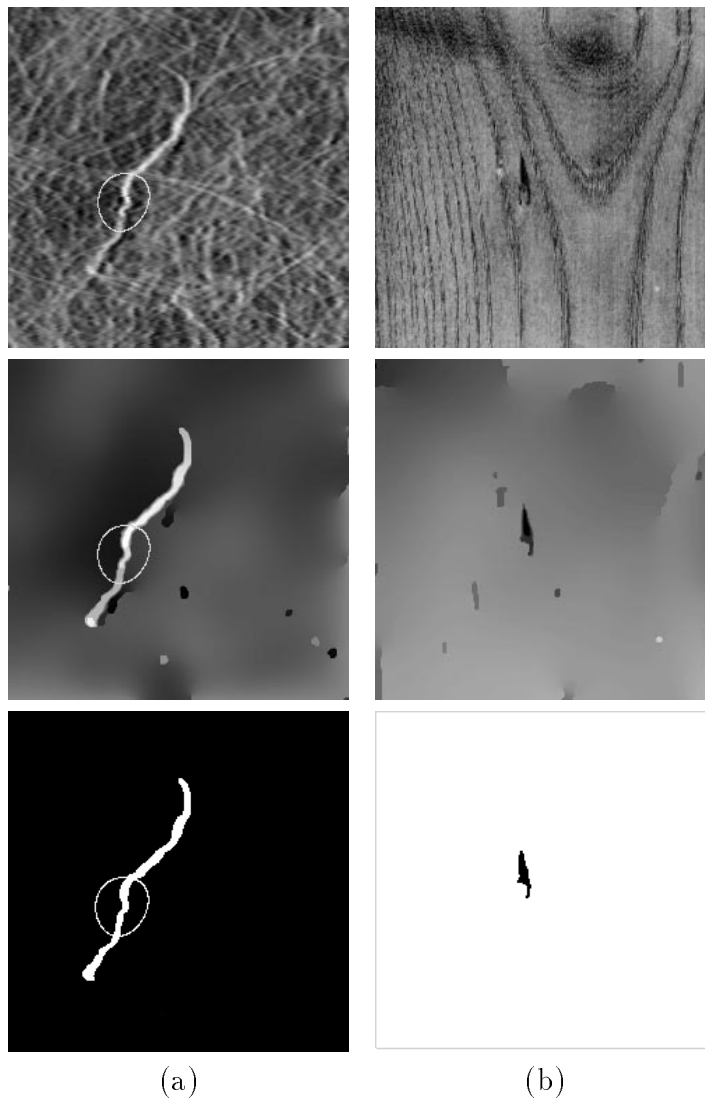


FIG. 5.9 – Diffusion d'images réelles. (a) : l'image d'un tissu (b) image d'une planche.

La diffusion anisotrope des deux images<sup>1</sup> de la figure (Fig.4.9) met en évidence la capacité du filtre proposé à renforcer les contrastes des objets significatifs d'une image. La colonne (a) traite une image de tissu ( $257 \times 257$ ). Après 300 itérations de l'algorithme avec  $\tau = 4\Delta t$  et  $s = 6$ , nous arrivons à récupérer la raie principale. La difficulté réside dans la région encerclée où cette raie, ayant une forme compliquée, présente une coupure et une dégradation de l'intensité du niveau de gris qui se confond de plus en plus à celle du fond de l'image. La colonne (b) traite une planche de bois avec un petit défaut au centre (l'image ayant une taille de  $256 \times 256$ ); après 600 itérations nous arrivons à extraire cet objet avec le petit segment qui le dépasse en bas à droite.

1. Images amicalement fournies par Joachim Weickert, Utrecht University Hospital, Netherlands

Nous illustrerons, dans le paragraphe suivant la possibilité de l'application de notre algorithme (4.1) dans le domaine de l'imagerie médicale. Cette application est considérée comme une étape de pré-traitement préparant les images à la segmentation (voir annexe B pour plus de détails).

### 5.1.4 Imagerie Médicale

La première image est une IRM (Imagerie par Résonance Magnétique) du cerveau (l'image ayant une taille de  $256 \times 256$ ). Dans cette image, extraite d'une série de coupes axiales au niveau du cerveau (Laboratoire TIMC-IMAG)<sup>2</sup>, nous cherchons à obtenir un contour précis de la tumeur cérébrale qui apparaît comme une zone nettement plus foncée dans la partie en bas à gauche de la photo (b). Nous avons traité cette image avec le filtre présenté ci-dessus en utilisant une régularisation temporelle sur quatre pas de temps ( $\tau = 4\Delta t$ ) et un seuil  $s = 5$  et avec seulement dix itérations. Ceci est possible car l'image de départ n'est pas très bruitée. La photo (b) montre l'image traitée où il est facile de localiser la tumeur ainsi que les zones d'oedème l'entourant. Pour d'autres traitements de cette image voir : [41, 40] pour les approches par réaction-diffusion, Lions et al [26] pour une approche par une version régularisée du modèle de Perona-Malik détaillé dans le chapitre 1, et Demongeot et al [15] pour une approche par les réseaux de neurones et par les systèmes dynamiques.

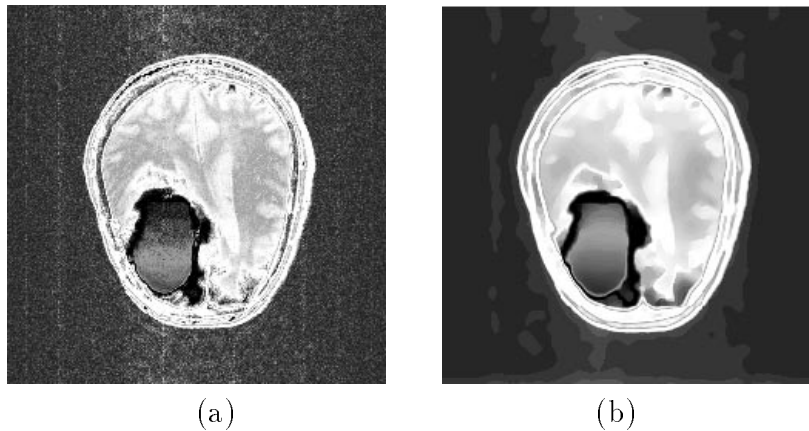


FIG. 5.10 – Diffusion anisotrope d'une image IRM ( $256 \times 256$ ) du cerveau. (a) : l'image originale, (b) : l'image diffusée avec  $\tau = 4\Delta t$  et  $s = 5$ .

La deuxième expérience traite aussi une image IRM d'une coupe sagittale de la tête. La figure (Fig.4.11) montre le comportement asymptotique de l'algorithme (4.1). Le filtrage Gaussien (voir chap 1 1.2.1) réduit les structures significatives de l'image en taches sombres (deuxième image colonne (a)) qui diminuent au cours des itérations jusqu'à disparaître (dernière image colonne (a)). Notons aussi que l'utilisation de grandes valeurs de  $s$  et

<sup>2</sup>. Faculté de Médecine, Service d'IRM, CHU de Grenoble

$\tau$  ( $s = 20$  et  $\tau = 20\Delta t$ ) nous donne un lissage grossier et une "convergence" très lente (colonne (b)). Une application de l'algorithme (4.1) sur cette image originale avec une régularisation temporelle sur cinq pas de temps ( $\tau = 5\Delta t$ ) et un seuil  $s = 5$  permet de préserver les structures significatives; même après 3000 itérations l'image diffusée ne change pas par rapport à celle obtenue avec 500 itérations. Une comparaison entre les colonnes (b) et (c) montre que, contrairement aux autres filtres anisotropes, les niveaux de gris et les positions des arêtes correspondant à un même dessin sont conservés par le filtre proposé (le nez par exemple).

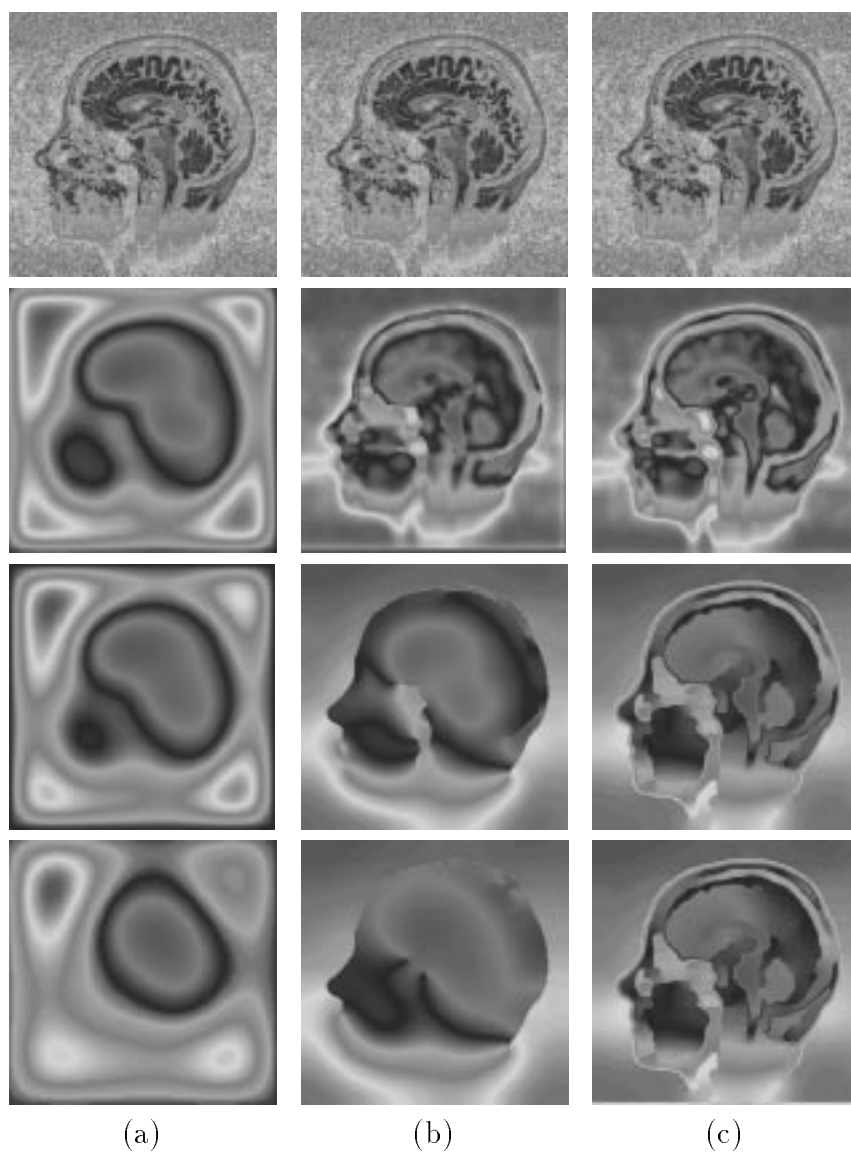


FIG. 5.11 – Image IRM d'une coupe sagittale du crâne : (a) : diffusion isotrope (b) : diffusion anisotrope avec  $\tau = 20\Delta t$  et  $s = 20$  (c) :  $\tau = 5\Delta t$  et  $s = 5$ . les lignes correspondent à 50, 500 et 3000 itérations.

La troisième expérience traite un aspect beaucoup plus délicat du traitement d'images médicales. Les objets significatifs de l'image originale<sup>3</sup> (Fig.4.12a) concernent des vaisseaux sanguins qui dessinent des courbes monodimensionnelles compliquées qui commencent à partir noyau (bas gauche de l'image (b)). Même si l'image originale ne semble pas très bruitée, un seuillage classique avec une valeur très basse en une image contenant seulement deux niveaux de gris (noir et blanc) échoue dans la zone correspondant aux tissus inhomogènes qui ne peuvent être distingués des vaisseaux. Notons aussi que l'utilisation de grandes valeurs pour le même seuillage a pour effet de déconnecter les vaisseaux. La photo (b) montre l'image traitée avec une régularisation temporelle sur cinq pas de temps ( $\tau = 5\Delta t$ ) et un seuil  $s = 5$  avec seulement 20 itérations. L'image filtrée montre qu'il est possible de suivre les vaisseaux loin du noyau.

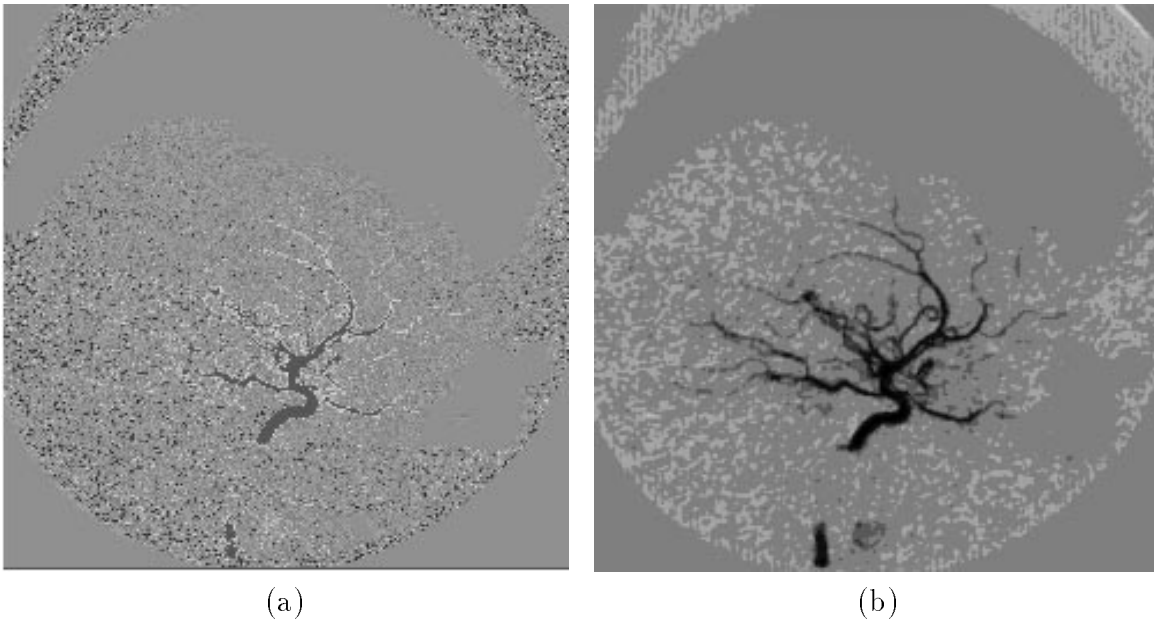


FIG. 5.12 – *Extraction d'objets monodimensionnels (vaisseaux sanguins) dans une angiographie numérique. (a) : l'image originale (b) : l'image filtrée avec  $\tau = 10\Delta t$  et  $s = 5$ .*

Notre dernière expérience toujours dans le cadre de l'imagerie médicale, traite une image échographique (échographie du fœtus  $286 \times 384$ ) qui est aussi un aspect difficile en pré-traitement et en segmentation. L'image originale est très bruitée, assez floue, donc difficile à segmenter (photo (c)). L'image (b) a été obtenue en appliquant l'algorithme (4.1) avec  $\tau = 5\Delta t$  et  $s = 10$ .

Les images (c) et (d) qui représentent les contours des deux images montrent à quel point les gradients de l'image originale sont inexploitablement car très bruités, alors que ceux de l'image filtrée par une diffusion anisotrope (photo (b)) sont bien restaurés. Ils sont très

3. Faculté de Médecine, Service de Radiologie, CHU de Grenoble

visibles au niveau du fœtus (centre de l'image (d)). Notons aussi que nous avons utilisé la même valeur du seuil pour l'extraction des gradients dans (c) et (d).

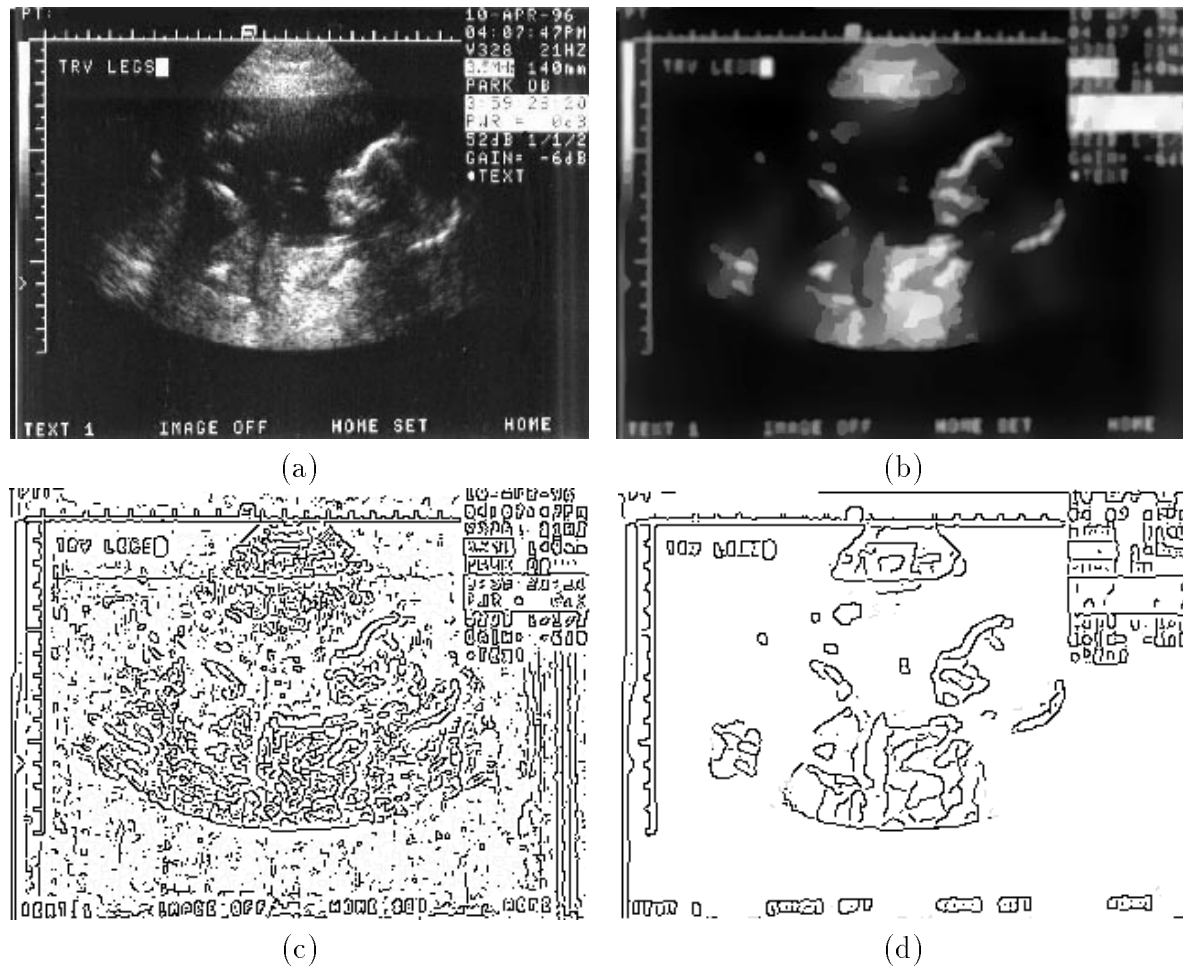


FIG. 5.13 – Diffusion anisotrope d'une image échographique du fœtus. (a) : l'image originale (b) : image filtrée avec  $\tau = 10\Delta t$  et  $s = 10$  (c) et (d) : Extraction des contours de ces deux images.

## 5.2 Diffusion anisotrope comme pré-traitement pour la segmentation

L'application suivante (Fig.4.14) concerne la segmentation d'une image échographique du coeur. L'extraction des gradients de l'image par le filtre de Canny-Deriche, présenté dans le chapitre 1, ne suffit pas pour déterminer les contours des deux ventricules (Fig.4.14.(b)). La photo (c) montre l'image diffusée par le filtre présenté avec  $\tau = 10\Delta t$  et  $s = 5$ . Une coopération avec la méthode des *snakes-splines* détaillée en annexe B, permet

d'extraire les contours des deux ventricules (Fig.4.14.(f)).

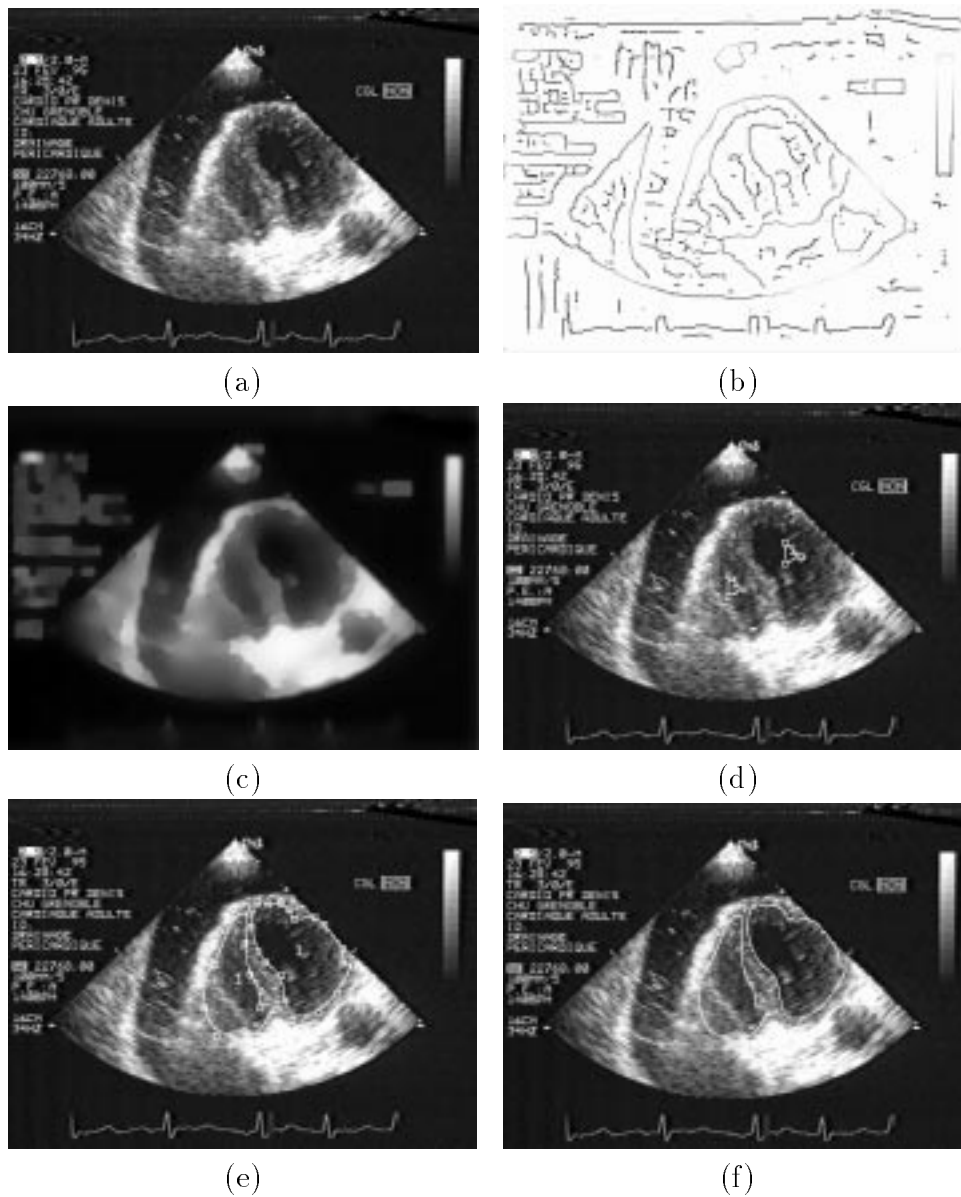


FIG. 5.14 – Segmentation d'une image échographique du coeur ( $384 \times 256$ ).

### 5.3 Diffusion avec réinitialisation

Nous avons vu dans le paragraphe précédent que pour une image fortement bruitée ou fortement texturée, il est nécessaire d'utiliser de grandes valeurs de  $\tau$ . En terme de réseaux de neurones cela s'explique par le fait que le réseau rappelle l'information originale, qui est toujours disponible, tout en conservant les connexions ayant subi l'apprentissage. Dans

ces conditions, l'algorithme (4.1) devient largement dépendant du choix initial pour la matrice de diffusion. La technique de réinitialisation que nous allons maintenant décrire permet dans une certaine mesure de s'affranchir de cette dépendance.

### 5.3.1 Principe

Après convergence de l'algorithme présenté vers une solution  $(\bar{u}, \bar{L})$  le résidu descend au-dessous de  $10^{-4}$ ; nous le réinitialisons avec les mêmes paramètres  $\tau$  et  $s$ , et avec la même image originale  $u_0$  mais avec  $\bar{L}$  comme condition initiale de la matrice de diffusion ( $L_0 = \bar{L}$ ). Ce procédé peut être répété jusqu'à convergence (résidu descend en de sous de  $10^{-4}$ ). En pratique, une ou deux réinitialisations ont été suffisantes pour donner des résultats que l'on peut juger satisfaisants.

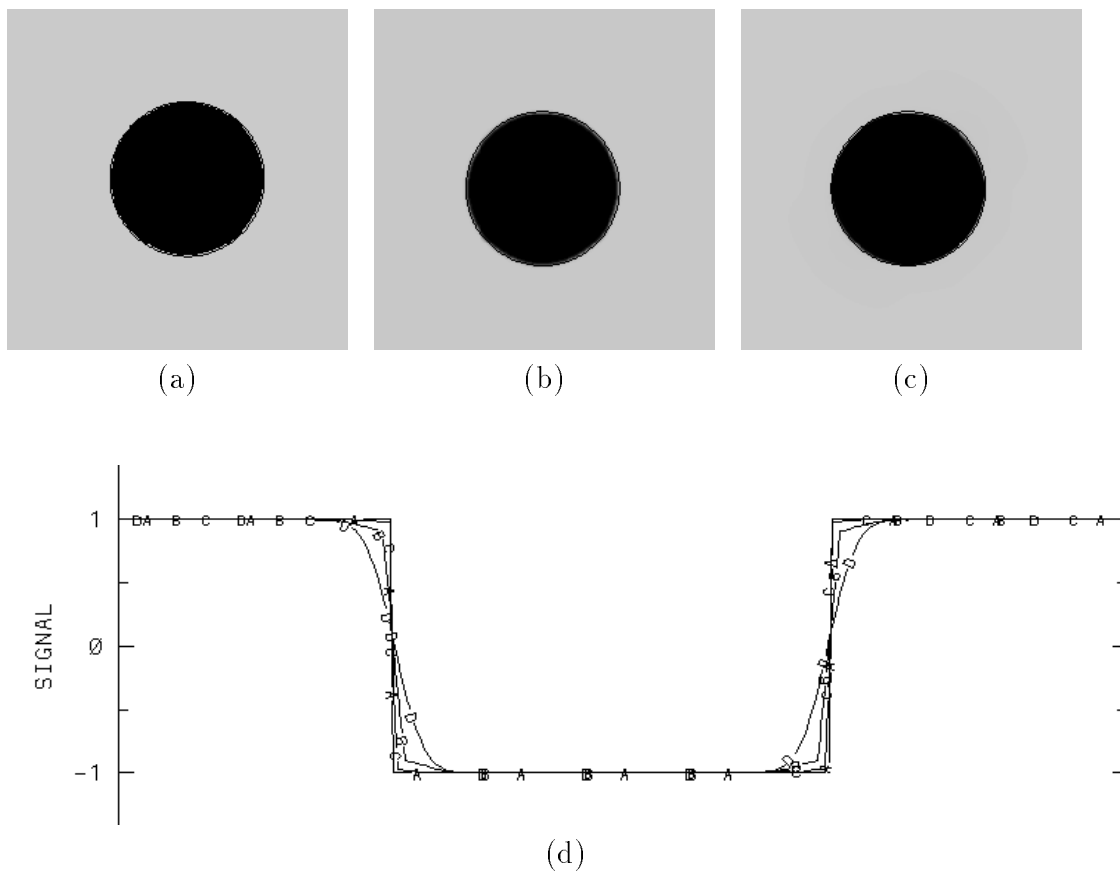


FIG. 5.15 – *Principe de réinitialisation.*

Dans la figure 4.15, nous traitons un disque noir sur un fond gris (la même image que 4.1). La photo (b) montre l'image diffusée avec  $s = 3$  et  $\tau = 3\Delta t$ . Fig. 4.15(c) montre l'image diffusée avec une réinitialisation. Avec l'algorithme réinitialisé nous obtenons l'image de départ. Il est naturel d'obtenir la solution exacte après réinitialisation

car les fronts n'ont pas été détruits lors de la première passe, et "recouvrent" la position exacte des bords. Dans cette application nous avons utilisé le même nombre d'itérations avant et après la réinitialisation pour comparer les deux images diffusées. Par la suite, et en pratique, nous obtenons la convergence de l'algorithme réinitialisé avec seulement quelques itérations.

Dans la figure 4.15 (d), nous avons pris une coupe transversale des disques (a) (b) et (c). La courbe  $A$  représente le signal original,  $B$  le signal diffusé,  $C$  le signal diffusé avec réinitialisation et  $D$  le signal lissé avec un filtre isotrope Gaussien. Cette figure montre qu'après réinitialisation nous retrouvons le signal de départ, ceci étant possible puisque l'image originale n'est pas bruitée.

### 5.3.2 Applications

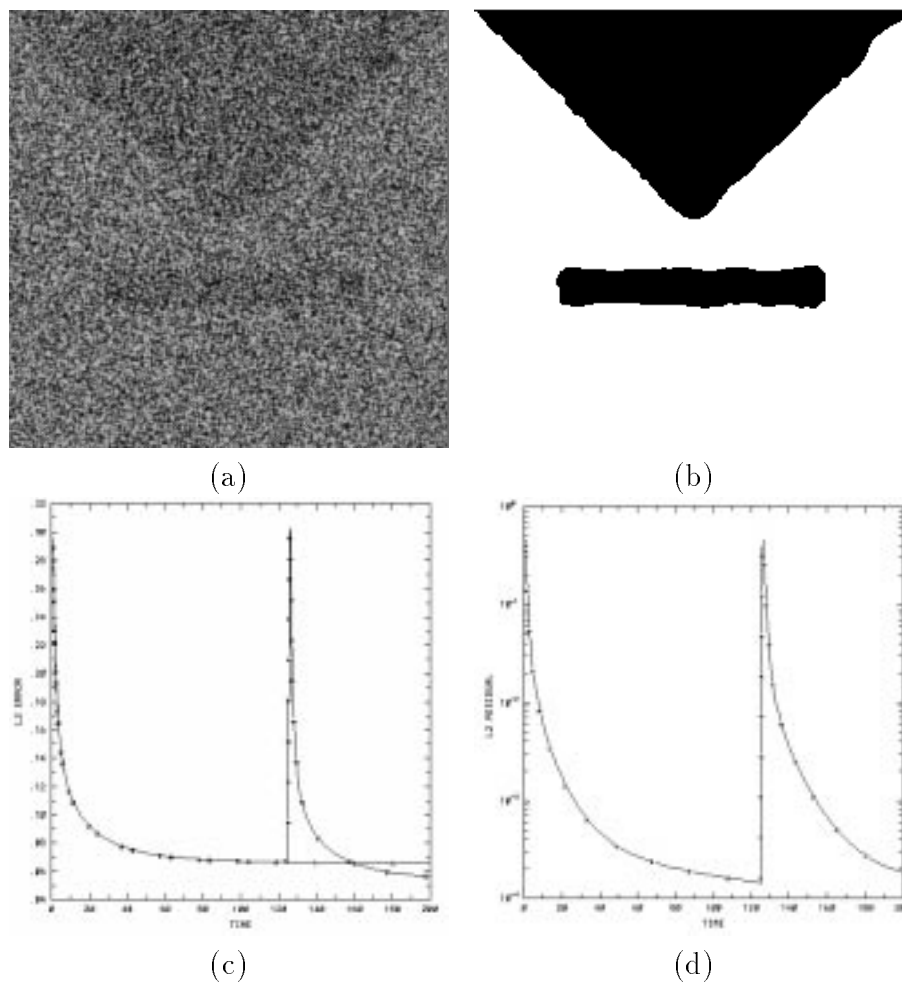


FIG. 5.16 – Diffusion anisotrope avec réinitialisation de l'image triangle-rectangle ( $256 \times 256$ ) bruitée à 90%.

Dans la première application, nous reprenons l'image du *triangle – rectangle*, avec cette fois-ci un niveau de bruit de 90% (Fig.4.16.a). A notre connaissance aucun filtre "conventionnel" ne permet de restaurer une image bruitée à ce point. Un filtrage avec réinitialisation ( $\tau = 10\Delta t$  et  $s = 10$ ) est nécessaire pour la restauration; la photo (b) montre l'image diffusée et seuillée. La courbe d'erreur (c) montre que l'erreur après réinitialisation descend au dessous de celle obtenue avant, après seulement 40 itérations. La courbe du résidu (d) confirme cette propriété; elle montre la convergence de l'algorithme (résidu  $\approx 10^{-4}$ ) avant et après la réinitialisation.



FIG. 5.17 – Réinitialisation de la diffusion anisotrope sur une image fortement texturée et bruitée à 30%

Dans la deuxième expérience (Fig.4.17 et 4.18), nous avons repris l'image de Lenna (l'image ayant une taille de  $256 \times 256$ ), dans laquelle nous avons détruit 30% des pixels (

photo b). La figure (Fig.4.17c) montre l'image lissée avec un temps de relaxation  $\tau = 10\Delta t$  et un seuil  $s = 10$ . Une comparaison entre les photos (c) et (d) montre que la diffusion anisotrope réinitialisée lisse le bruit et renforce les contrastes (photo d).

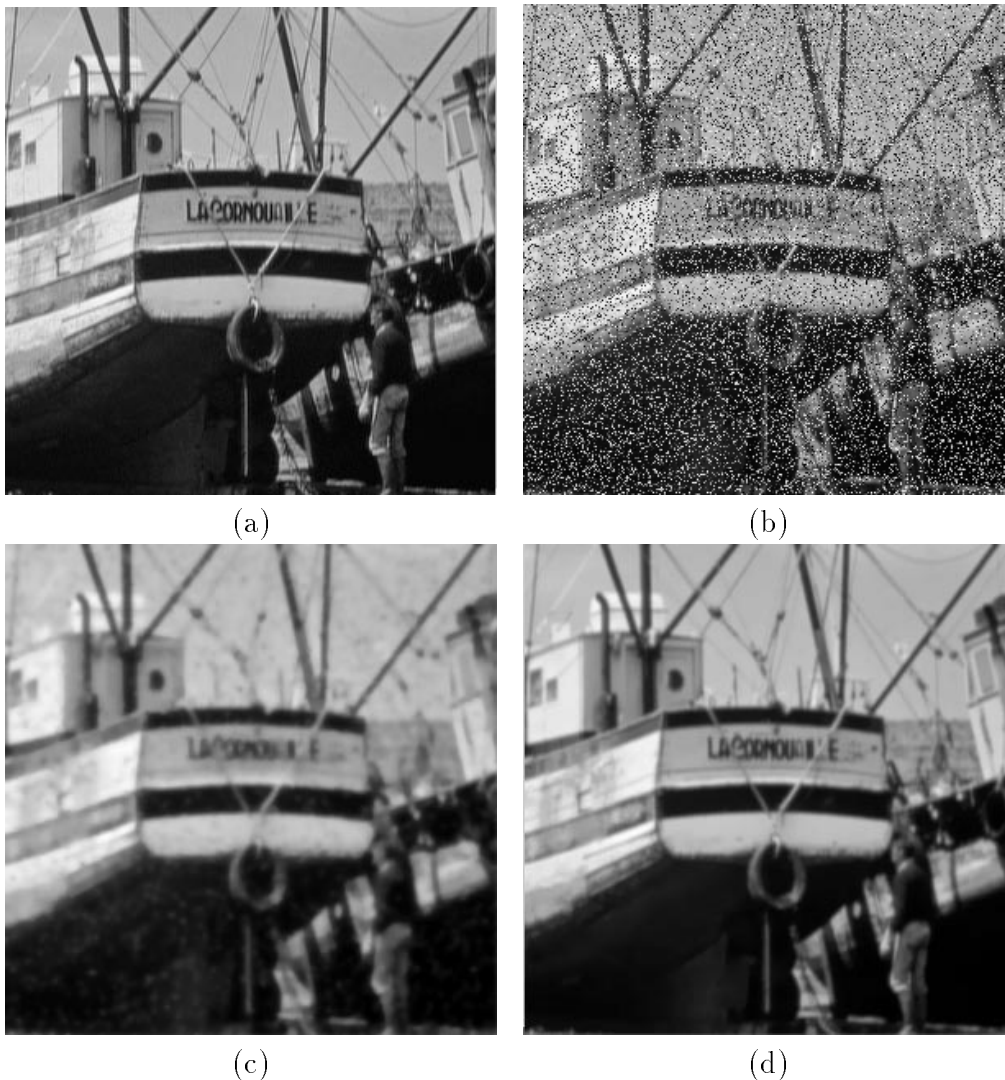


FIG. 5.18 – Diffusion anisotrope avec réinitialisation d'une image ( $256 \times 256$ ). (a) : image originale (b) : image bruitée à 30% (c) : filtrage de l'image bruitée (d) : filtrage avec une réinitialisation.

La figure (Fig.4.18) montre la restauration d'une image de bateaux<sup>4</sup> ( $256 \times 256$ ) bruitée à 30%. La photo (d) montre l'effet du renforcement des contrastes. La réinitialisation donne un résultat comparable à celui d'Alvarez et Esclarín qui utilisent un traitement basé sur les méthodes de réaction-diffusion [9] pour un niveau de bruit plus faible. i

4. Image amicalement fournie par Julio Esclarín, Universidad de Las Palmas de Gran Canaria, Spain

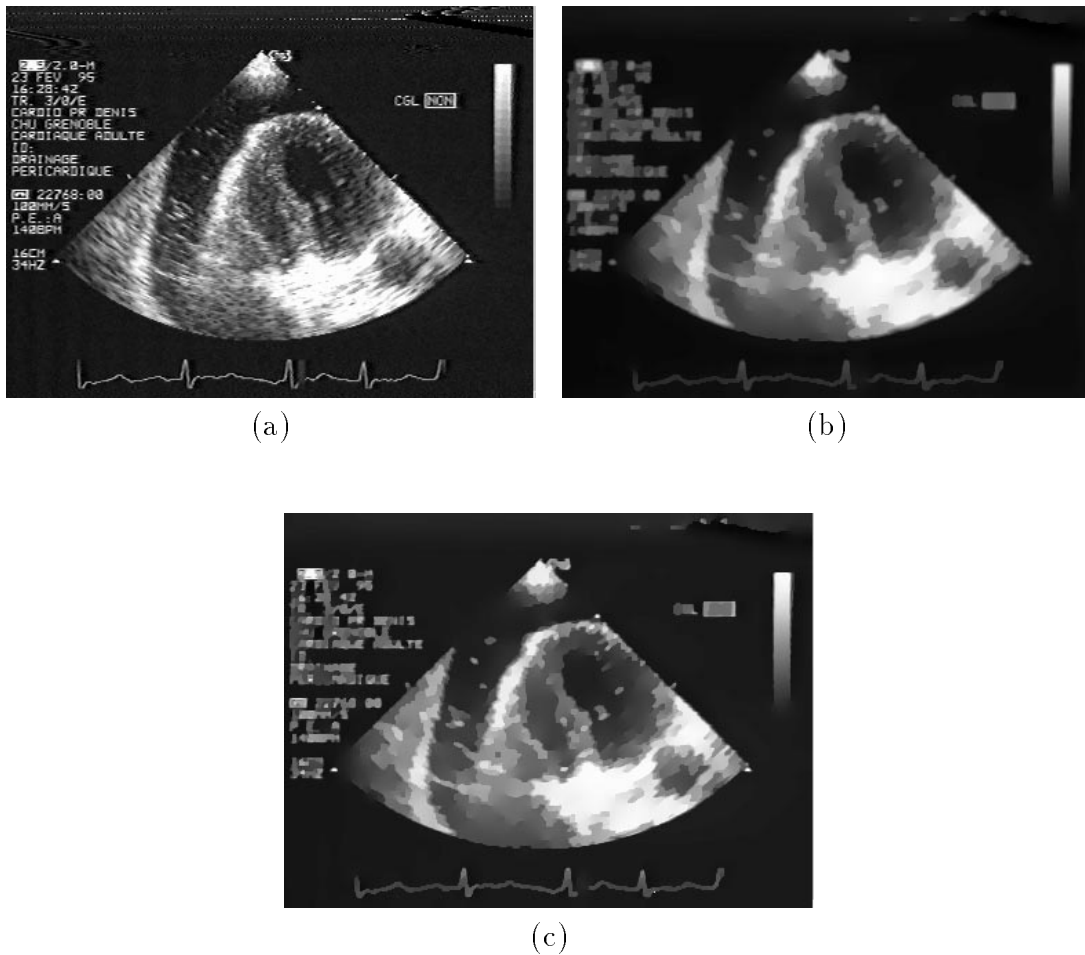


FIG. 5.19 – Réinitialisation de la diffusion anisotrope sur une image échographique du coeur

La dernière expérience (Fig.4.19) concerne une image échographique du coeur (l'image ayant une taille de  $286 \times 384$ ). La figure (b) montre l'image filtrée avec  $\tau = 6$  et  $s = 10$ . La photo (c) montre l'image diffusée avec une réinitialisation, où on voit bien l'amélioration au niveau des deux ventricules. Les images (e) et (f) de la figure (Fig.4.20), qui représentent les contours des images (a) et (c), permettent de voir que les gradients de l'image originale sont impossibles à exploiter car très bruités, alors que ceux de l'image filtrée avec réinitialisation sont bien restaurés (ils sont bien visibles au niveau des deux ventricules). Les images (f) et (g) de la figure (Fig.4.21) représentent les profils de luminescence de la ligne 250 de l'image (a) et de celles des images (c) et (d), respectivement. Dans (a) on voit que le signal original est très bruité. Dans (g) on remarque que la diffusion anisotrope (signal A) atténue bien le bruit mais ne donne pas encore une image parfaite, et que la réinitialisation (signal B) permet une nette amélioration du signal, tout en respectant les discontinuités significatives dans le signal original (4.21 f).

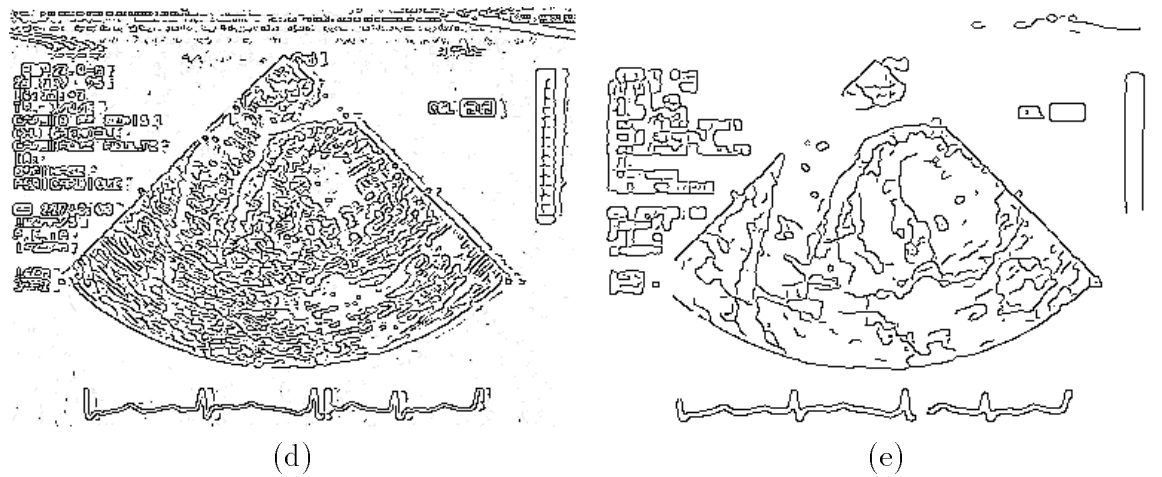


FIG. 5.20 – *Extraction des contours des images (a) et (c) de la figure 4.19.*

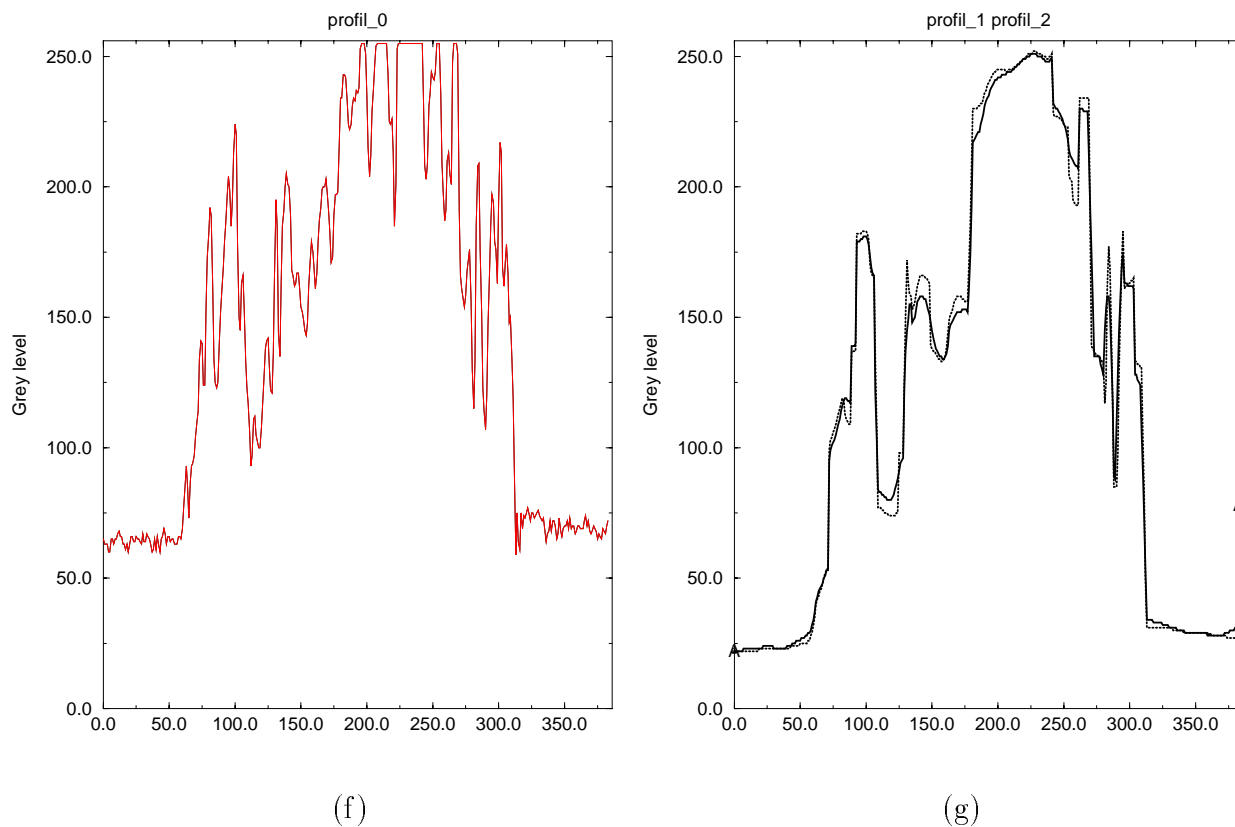


FIG. 5.21 – *Profil des luminescence de la ligne 250 des images de la figure 4.19.*

Dans le paragraphe suivant nous introduisons un filtre basé sur le réseau neuronal à synapses adaptative détaillé dans le chapitre 2.

## 5.4 Réseaux de neurones

### 5.4.1 Algorithme

Dans ce paragraphe, nous considérons le réseau de neurones décrit dans le chapitre 2 comme alternative au modèle EDP étudié précédemment. Pour cela nous rappelons le système suivant :

$$\begin{cases} \frac{dU_i}{dt} &= \sum_j J_{ij}(V_j - V_i) \\ \tau \frac{dJ_{ij}}{dt} &= -J_{ij} + \varepsilon^{-6} \cdot \theta\left(\frac{x_i - x_j}{\varepsilon}\right) \|x_i - x_j\|^2 \left[ s^2 - \frac{|U_i - U_j|^2}{\|x_i - x_j\|^2} \right] \\ V_i &= g_i(\lambda U_i) \end{cases} \quad (5.3)$$

Comme pour le système de diffusion anisotrope, nous utiliserons un schéma mixte pour la discrétisation : explicite pour l'équation de résistance et implicite pour l'équation d'apprentissage.

Dans ce cas, l'image sera considérée comme une grille de  $N^2$  neurones. Chaque neurone  $i$  de position  $x_i = ih$  ( $h = 1/N^2$ ) est porteur d'une fonction d'entrée  $U_i$  le niveau de gris de l'image et d'une fonction de sortie  $V(x_i) = V_i = g(U_i)$ . Pour simplifier les calculs nous avons pris  $g_i = g$ , donnée par :

$$g(x) = \frac{1}{2} [|x + 1| - |x - 1|] \quad \forall x \in \mathbb{R} \quad (5.4)$$

Le poids synaptique reliant le neurone  $i$  au neurone  $j$  sera approché par  $J_{ij}^n$ . La discrétisation de l'équation de résistance nous donne :

$$U_i^{n+1} = U_i^n + \Delta t \sum_j J_{ij}^n (V_j^n - V_i^n), \quad (5.5)$$

et celle d'apprentissage est discrétisée par :

$$J_{ij}^{n+1} = \left(\frac{k}{1+k}\right) (J_{ij}^n + \frac{1}{k} C_{ij}^n) \quad (5.6)$$

où  $C_{ij}^n$  est donnée par :

$$C_{ij}^n = \theta\left(\frac{x_i - x_j}{\varepsilon}\right) \|x_i - x_j\|^2 \left[ s^2 - \frac{|U_i^n - U_j^n|^2}{\|x_i - x_j\|^2} \right], \quad (5.7)$$

$x_i$  par  $(i_1 h, i_2 h)$  et la fonction  $\theta$ , à symétrie radiale normalisée comme ( chapitre 2.4.2), par :

$$\theta(x) = \frac{4}{\pi} \exp(-|x|^2) \quad \text{pour tout } x \in \mathbb{R}^2. \quad (5.8)$$

Remarquons que, pour chaque pixel (neurone)  $i$  de l'image, il est inutile de prendre tous les autres pixels comme étant en liaison synaptique avec  $i$ , car au delà d'une certaine distance, l'information apportée (ou liaison synaptique) est négligeable voir numériquement nulle. Ainsi, pour chaque pixel  $i$  nous prenons une portée synaptique  $\varepsilon$  de l'ordre de quelques pixels  $\varepsilon = mh$ , ce qui correspond à une cellule de  $m^2$  neurones en liaison synaptique avec  $i$ .

### 5.4.2 Application

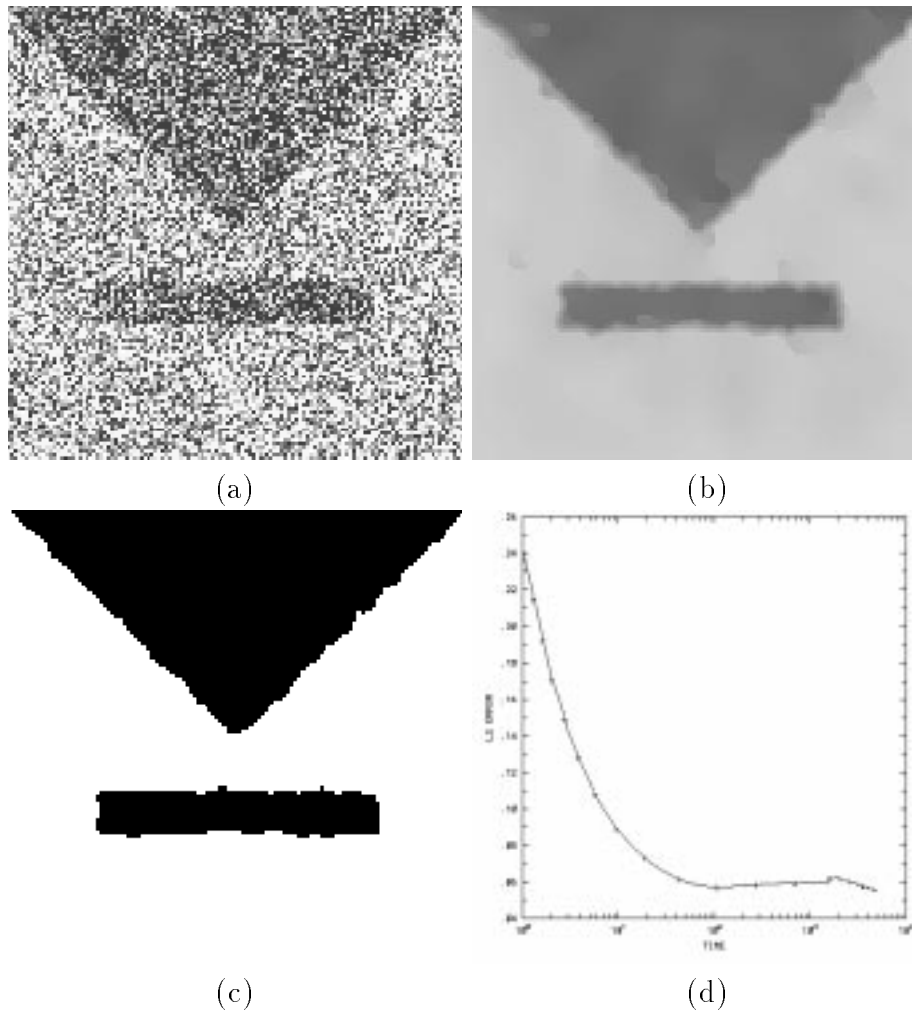


FIG. 5.22 – Lissage de l'image triangle – rectangle bruitée à 70% avec le réseau neuronal. (a) : image originale (b) : image filtrée (c) : seuillage de l'image lissée (d) : la courbe d'erreur.

Pour les applications, la forme particulière de la fonction de transfert  $g$  (4.4) nous oblige à considérer des images à deux niveaux de gris. Nous reprenons l'image du *triangle-*

*rectangle* dégradée à 70% (Fig.4.22). Nous avons filtré cette image avec un temps de relaxation  $\tau = 5\Delta t$  et un seuil  $s = 2$ . Pour la restauration de cette image, nous avons utilisé un nombre d'itération égale à 5000 et une portée synaptique  $\varepsilon = 5h$ , ce qui correspond à des boîtes de 25 neurones.

Nous avons montré dans le chapitre 2 que le système (4.3) est stable au sens de Lyapunov, que les attracteurs du réseau sont les images filtrées, obtenues sur les asymptotiques en temps du réseau neuronal. La particularité et l'intérêt de ce filtre (spatio-temporel) réside dans ses deux paramètres  $\varepsilon$  et  $\tau$ , le premier étant lié à la notion d'échelle spatiale, le second à la notion d'échelle temporelle. L'intérêt réside aussi dans la nature des systèmes biologiques où l'on retrouve la mémoire qui permet de reconnaître un objet dans une image dégradée, à partir de la liaison entre les faits mémorisés et les informations relatives à cet objet.

En appliquant le réseau de neurones on retrouve comme prévu (Proposition 1) des résultats comparables à ceux de l'équation de Volterra, mais le modèle (4.3) est moins flexible (2 niveaux de gris, correspondant à un modèle de réaction diffusion) et plus coûteux.

## 5.5 Conclusions et Perspectives

Dans ce travail, nous avons introduit et étudié un nouveau modèle, fondé sur une équation de diffusion anisotrope à diffusivité tensorielle, pour le traitement d'images. Ce filtre semble répondre à certains problèmes rencontrés par les modèles présentés dans le chapitre 1. Par ailleurs, contrairement aux autres filtres, ce modèle ne contient pas de paramètres liés aux détails à garder dans l'image à restaurer.

L'existence, l'unicité, la dépendance continue par rapport à l'image originale et la conservation de la valeur moyenne des niveaux de gris présent dans l'image à restaurer ont été démontrés. L'équivalence avec les réseaux de neurones a permis d'avoir des propriétés de stabilité asymptotique au sens de Lyapunov et des états asymptotiques non triviaux.

L'application du filtre proposé sur des images synthétiques, réelles ou médicales illustrent la capacité de ce modèle anisotrope à fournir un lissage sélectif. Par rapport aux filtres fondés sur la diffusion linéaire, non linéaire ou anisotrope présentés dans le chapitre 1 il possède des capacités intéressantes comme la préservation d'objets monodimensionnels dans une image bruitée ou le lissage intra-région plus continu ou encore la préservation et l'extraction d'objets monodimensionnels dans une image bruitée; mais, et c'est là le plus important, l'image filtrée est obtenue sur les asymptotes du modèle, évitant ainsi le choix d'un paramètre du temps suffisant pour stopper le processus de diffusion.

Certains points de ce travail semblent intéressants en vue de recherches futures : ainsi l'extension de nos résultats aux domaines du flot optique et des contours actifs. Cela représenterait un réel avantage en comparaison avec les méthodes déjà existantes.

## Annexe A

# A Volterra type model for image processing

Cette annexe reprend un article écrit en collaboration avec Georges-Henri Cottet et accepté par IEEE Transaction on Images Processing Special Issue on Partial Differential Equations (PDE's) and Geometry-Driven Diffusion in Image Processing and Analysis (édité par Morel, Sapiro, Tannenbaum et Caselles).



## Annexe B

# Image segmentation using snake-splines and anisotropic diffusion operators

Cette annexe reprend un article écrit en collaboration avec Georges-Henri Cottet, François Leitner et Jaques Demongeot du Laboratoire TIMC Faculté de Médecine et soumis à IEEE Transaction on Pattern Analysis and Machine Intelligence.



## Annexe C

### Une étude tridimensionnelle

Dans cette annexe, nous donnerons une étude tridimensionnelle utilisant le parallélisme pour cela rappelons le modèle de type Volterra introduit dans le chapitre 2

$$\frac{\partial u}{\partial t} - \operatorname{div}(L[\nabla u]) = 0 \quad (\text{C.1})$$

$$\frac{dL}{dt} + \frac{1}{\tau}L = \frac{1}{\tau} \begin{cases} P_{\nabla u^\perp} & \text{si } |\nabla u| > s \\ |\nabla u|^2 P_{\nabla u^\perp} + \frac{3}{2}(s^2 - |\nabla u|^2)Id & \text{si } |\nabla u| \leq s \end{cases} \quad (\text{C.2})$$

### Algorithme

Pour la discrétisation de ce système (C.1)-(C.2), nous utilisons un schéma mixte comme pour le cas bidimensionnel (chapitre 3).

La discrétisation en espace est faite avec deux pas  $h_1 = 1/N$  et  $h_2 = 1/M$  où  $(N \times N \times M)$  représente la taille de l'image; et celle en temps est régulière avec un pas de temps  $\Delta t$ .

Soit  $u_{ijk}^n$  une approximation de  $u$  au point  $(ih_1, jh_1, kh_2)$  (avec  $0 \leq i \leq N$ ,  $0 \leq j \leq N$  et  $0 \leq k \leq M$ ) à l'instant  $t = n\Delta t$ .

La matrice de diffusion  $L$  est approchée par:

$$L_h^n = \begin{pmatrix} L_{(ijk)xx}^n & L_{(ijk)xy}^n & L_{(ijk)xz}^n \\ L_{(ijk)yx}^n & L_{(ijk)yy}^n & L_{(ijk)yz}^n \\ L_{(ijk)zx}^n & L_{(ijk)zy}^n & L_{(ijk)zz}^n \end{pmatrix} \quad (\text{C.3})$$

En 3-D, la matrice de projection s'écrit comme suit :

$$P_{\nabla u^\perp} = \frac{1}{u_x^2 + u_y^2 + u_z^2} \begin{pmatrix} u_y^2 + u_z^2 & -u_x u_y & -u_x u_z \\ -u_y u_x & u_x^2 + u_z^2 & -u_y u_z \\ -u_x u_z & -u_y u_z & u_x^2 + u_y^2 \end{pmatrix} \quad (\text{C.4})$$

La divergence et le gradient sont discrétisés avec des décentrages différents comme suit:

$$\text{div}_h(u_1, u_2, u_3) = h_1^{-1}(\Delta_+^x u_1 + \Delta_+^y u_2) + h_2^{-1} \Delta_+^z u_3 \quad (\text{C.5})$$

et

$$\nabla_h u = h^{-1}(\Delta_-^x u, \Delta_-^y u, \Delta_-^z u) \quad (\text{C.6})$$

où

$$\Delta_+^x u_{i,j,k} = u_{i+1,j,k} - u_{i,j,k} \quad , \quad \Delta_-^x u_{i,j,k} = u_{i,j,k} - u_{i-1,j,k}$$

pour  $\Delta_+^y$ ,  $\Delta_-^y$ ,  $\Delta_+^z$  et  $\Delta_-^z$  on utilise la même discrétisation en intervertissant les rôles de  $i$ ,  $j$  et  $k$ . Avec cette discrétisation, une approximation de  $L(\nabla u)$  est la suivante:

$$L_h(\nabla u_h) = \left( h_1^{-1}(L_{(ijk)xx}^n \Delta_-^x u_{ijk}^n + L_{(ijk)xy}^n \Delta_-^y u_{ijk}^n + L_{(ijk)xz}^n \Delta_-^z u_{ijk}^n), h_1^{-1}(L_{(ijk)yx}^n \Delta_-^x u_{ijk}^n + L_{(ijk)yy}^n \Delta_-^y u_{ijk}^n + L_{(ijk)yz}^n \Delta_-^z u_{ijk}^n), h_2^{-1}(L_{(ijk)zx}^n \Delta_-^x u_{ijk}^n + L_{(ijk)zy}^n \Delta_-^y u_{ijk}^n + L_{(ijk)zz}^n \Delta_-^z u_{ijk}^n) \right)$$

et  $\text{div}(L(\nabla u))$  est approchée par:

$$h_1^{-2} \left[ \Delta_+^x (L_{(ijk)xx}^n \Delta_-^x u_{ijk}^n + L_{(ijk)xy}^n \Delta_-^y u_{ijk}^n + L_{(ijk)xz}^n \Delta_-^z u_{ijk}^n) + \Delta_+^y (L_{(ijk)yx}^n \Delta_-^x u_{ijk}^n + L_{(ijk)yy}^n \Delta_-^y u_{ijk}^n + L_{(ijk)yz}^n \Delta_-^z u_{ijk}^n) \right] + h_2^{-2} \left[ \Delta_+^z (L_{(ijk)zx}^n \Delta_-^x u_{ijk}^n + L_{(ijk)zy}^n \Delta_-^y u_{ijk}^n + L_{(ijk)zz}^n \Delta_-^z u_{ijk}^n) \right]$$

Pour la résolution numérique, nous avons choisi un schéma mixte:

- Schéma implicite pour l'équation (C.2):

$$L_{ijk}^{n+1} = \frac{\tau}{\tau + \Delta t} L_{ijk}^n + \frac{\Delta t}{\tau + \Delta t} F_{ijk}^n \quad (\text{C.7})$$

où  $F_{ijk}^n$  est une approximation à l'instant  $t = n\Delta t$  au point  $(ih_1, jh_1, kh_2)$  du membre droit de (C.2) donnée par :

$$F_{ijk}^n = \begin{cases} (P_{\nabla u^\perp})_{ijk}^n & \text{si } |\nabla u^n| > s \\ |\nabla u^n|^2 (P_{\nabla u^\perp})_{ijk}^n + \frac{3}{2}(s^2 - |\nabla u^n|^2) \delta_{ijk} & \text{si } |\nabla u^n| \leq s \end{cases} \quad (\text{C.8})$$

Comme on l'a vu dans le chapitre 3, nous avons  $F_{ijk}^n$ , une matrice définie positive  $\forall i, j$  et  $k$ . L'équation (C.7) assure la positivité de  $L$  qui permet d'éviter l'antidiffusion dans l'équation (C.1).

- Schéma explicite pour l'équation (C.1):

$$\begin{aligned} \frac{u_{ijk}^{n+1} - u_{ijk}^n}{\Delta t} - h_1^{-2} \left[ \Delta_+^x (L_{(ijk)xx}^n \Delta_-^x u_{ijk}^n + L_{(ijk)xy}^n \Delta_-^y u_{ijk}^n + L_{(ijk)xz}^n \Delta_-^z u_{ijk}^n) + \right. \\ \left. \Delta_+^y (L_{(ijk)yx}^n \Delta_-^x u_{ijk}^n + L_{(ijk)yy}^n \Delta_-^y u_{ijk}^n + L_{(ijk)yz}^n \Delta_-^z u_{ijk}^n) \right] + \\ h_2^{-2} \left[ \Delta_+^z (L_{(ijk)zx}^n \Delta_-^x u_{ijk}^n + L_{(ijk)zy}^n \Delta_-^y u_{ijk}^n + L_{(ijk)zz}^n \Delta_-^z u_{ijk}^n) \right] = 0 \end{aligned}$$

Ceci nous donne l'algorithme suivant:

$$P_d \left\{ \begin{array}{l} u_{ijk}^{n+1} = u_{ijk}^n + \frac{\Delta t}{h_1^2} \left[ \Delta_+^x (L_{(ijk)xx}^n \Delta_-^x u_{ijk}^n + L_{(ijk)xy}^n \Delta_-^y u_{ijk}^n + L_{(ijk)xz}^n \Delta_-^z u_{ijk}^n) + \right. \\ \left. \Delta_+^y (L_{(ijk)yx}^n \Delta_-^x u_{ijk}^n + L_{(ijk)yy}^n \Delta_-^y u_{ijk}^n + L_{(ijk)yz}^n \Delta_-^z u_{ijk}^n) \right] + \\ \frac{\Delta t}{h_2^2} \left[ \Delta_+^z (L_{(ijk)zx}^n \Delta_-^x u_{ijk}^n + L_{(ijk)zy}^n \Delta_-^y u_{ijk}^n + L_{(ijk)zz}^n \Delta_-^z u_{ijk}^n) \right] \\ u_{ijk}^0 = u_0(ih_1, jh_1, kh_2) \quad , \quad 0 \leq i \leq N \quad \text{et} \quad 0 \leq j \leq N \\ L_{ijk}^{n+1} = \frac{\tau}{\tau + \Delta t} L_{ijk}^n + \frac{\Delta t}{\tau + \Delta t} F_{ijk}^n \end{array} \right. \quad (C.9)$$

Pour implémenter cet algorithme, nous avons utilisé le langage PVM (Parallel Virtual Machines).

**Proposition 4** (*Principe du maximum*)

Soient  $u_{ijk}^n$  le niveau de gris de l'image et  $L_{(ijk)xy}^n$  une composante de la matrice de diffusion à l'instant  $t = n \cdot \Delta t$

Si

$$\Delta t \max_{ijk} |L_{ijk}| \leq \frac{\min(h_1^2, h_2^2)}{9} \quad (C.10)$$

Alors

$$m \leq u_{ijk}^n \leq M \quad \forall i, j, k \in \mathbb{R}$$

**Preuve:**

Supposons que:

$$\Delta t \max_{ijk} |L_{ijk}| \leq \frac{\min(h_1^2, h_2^2)}{9}$$

alors (C.9) nous permet d'écrire:

$$\begin{aligned} \max_{ij} |u_{ijk}^{n+1}| &\leq \max_{ijk} |u_{ijk}^n| \cdot \left( 1 - \frac{9\Delta t}{h_1^2} \max_{ijk} |L_{ijk}| - \frac{9\Delta t}{h_2^2} \max_{ijk} |L_{ijk}| \right) + \frac{\Delta t}{h_1^2} \left( 9 \cdot \max_{ijk} |u_{ijk}^n| \max_{ijk} |L_{ijk}| \right) + \\ &\quad + \frac{\Delta t}{h_2^2} \left( 9 \cdot \max_{ijk} |u_{ijk}^n| \max_{ijk} |L_{ijk}| \right) \\ &\leq \max_{ij} |u_{ij}^n| \end{aligned}$$

Donc (C.10) est une condition suffisante pour que le principe du maximum soit satisfait.

## Applications

Cette expérience traite une image synthétique contenant un cube noir dans un fond blanc (l'image ayant une taille  $(128 \times 128 \times 128)$ ) auquel nous avons ajouté un niveau de bruit de 70%. La figure (C.1) montre les courbes d'erreur et du résidu correspondant au traitement de cette image par l'algorithme (C.9) avec un temps de relaxation sur dix pas de temps et un seuil  $s = 10$ .

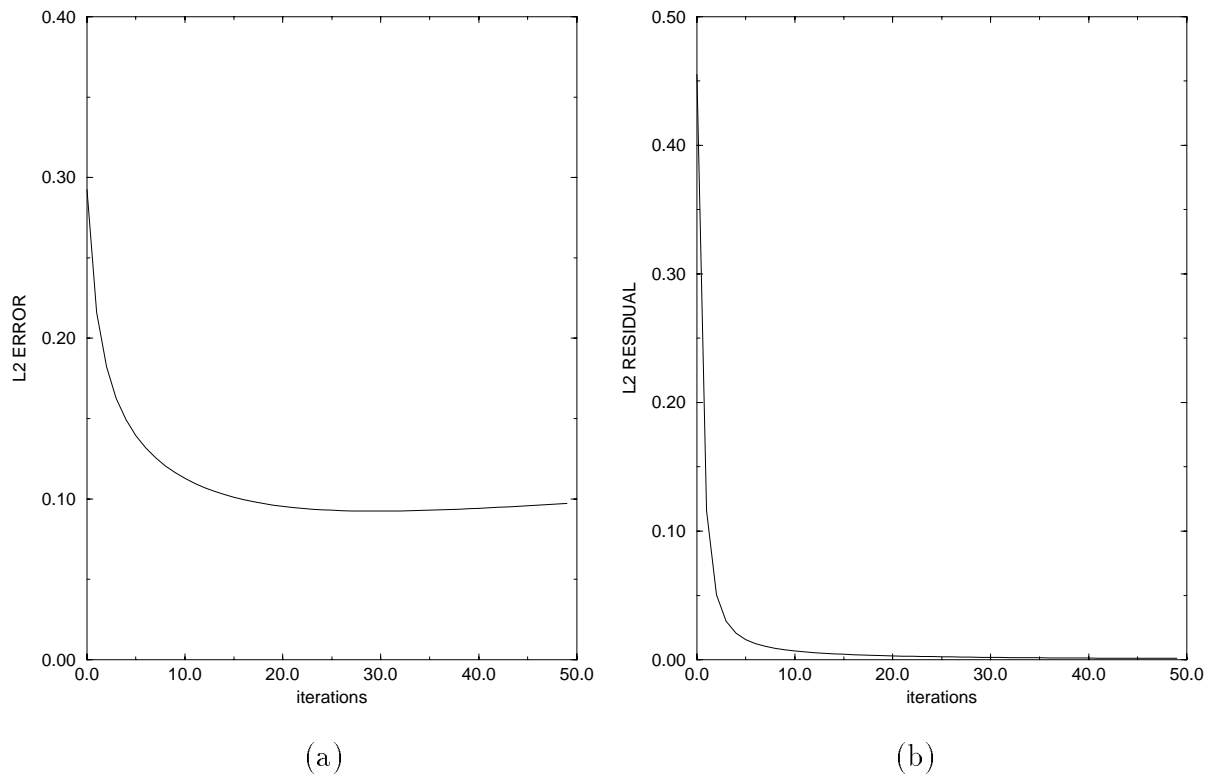


FIG. C.1 – Courbes d'erreur (a) et du résidu (b) .

# Bibliographie

- [1] L. Alvarez, F. Guichard, P.L. Lions, and J.M. Morel. Axiomatisation et nouveaux opérateurs de la morphologie mathématique . *C. R. Acad. Sci. Paris*, t. 315, Série I:265–268, 1992.
- [2] L. Alvarez, F. Guichard, P.L. Lions, and J.M. Morel. Axiomes et équations fondamentales du traitement d'images. (Analyse multiéchelle et E.D.P.). *C. R. Acad. Sci. Paris*, t. 315, Série I:135–138, 1992.
- [3] L. Alvarez, F. Guichard, P.L. Lions, and J.M. Morel. Équations fondamentales de l'analyse multiéchelle des films. *C. R. Acad. Sci. Paris*, t. 315, Série I:1145–1148, 1992.
- [4] L. Alvarez, F. Guichard, P.L. Lions, and J.M. Morel. Axioms and fundamental equations in image processing. *Arch. Rational Mech. Anal.*, 123:199–257, 1993.
- [5] L. Alvarez, P.L. Lions, and J.M. Morel. Image selective smoothing and edge detection by nonlinear diffusion.II. *SIAM J. Numer. Anal.*, 29(3):845–866, 1992.
- [6] L. Alvarez and L. Mazorra. Signal and image restoration using shock filters and anisotropic diffusion. *SIAM J. Numer. Anal.*, 31(2):590–605, 1994.
- [7] L. Alvarez and J.M. Morel. *Anisotropic diffusion*. In Bart M. ter Haar Romeny (Ed.), *Geometry-driven diffusion in computer vision*, Kluwer, Dordrecht, 1994.
- [8] L. Alvarez and J.M. Morel. Formalization and computational aspect of image analysis. *Acta Numerica*, pages 1–59, 1994.
- [9] L. Alvarez and J. Esclarín. Image Quantization using Reaction-Diffusion Equations. *SIAM J. Appl. Math.*, 57(1):153–175, 1997.
- [10] D. Amit, H. Gutfreund, and H. Sompolinsky. Storing infinite numbers of patterns in a spin glass model of neural network. *Phys. Rev. Lett.*, 55:1530–1533, 1985.
- [11] V. Anh, J.Y. Shi, and H.T. Tsui. Scaling Theorems for Zero Crossings of Bandlimited Signals. *IEEE Trans. Pattern Anal. & Machine Intell.*, 18:309–320, 1996.

- [12] J. Babaud, A.P. Witkin, M. Baudin, and R.O. Duda. Uniqueness of the Gaussian Kernel for Scale-Space Filtering. *IEEE Trans. Pattern Anal. & Machine Intell.*, 8:26–33, 1986.
- [13] C. Ballester, V. Caselles, and M. González. Affine Invariant Segmentation by Variational Method. *SIAM J. Appl. Math.*, 56(1):294–325, 1996.
- [14] A.L. Benabid, P. Cinquin, J.F. Lebas, J. Demongeot, and J. de Rougemont. A computer driven robot for stereotactic surgery connected to cat-scan magnetic resonance imaging. Technological design and preliminary results. *Applied Neurophysiology*, 50:153–154, 1987.
- [15] F. Berthommier, O. Francois, T. Coll, T. Herve, I. Marque, and J. Demongeot. Asymptotic behavior of neural networks and image processing. In A. Babloyantz, editor, *Self-Organization, Emerging properties and Learning*, pages 219–230. NATO Series, Plenum Press, 1991.
- [16] T. Blankenship. *Real-time enhancement of medical ultrasound images*. Plenum, New York, 1987.
- [17] J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. & Machine Intell.*, 8:679–698, 1986.
- [18] V. Caselles, F. Catté, T. Coll, and F. Dibos. A geometric model for active contours in image processing. *Numer. Math.*, 66:1–33, 1993.
- [19] V. Caselles and B. Coll. Snakes in Movement. *SIAM J. Numer. Anal.*, 33:2445–2456, 1996.
- [20] V. Caselles, B. Coll, and J.M. Morel. Junction detection and filtering: a morphological approach. In *Proc. third IEEE Int. Conf. on Image Processing*, pages I–493–496, (ICIP-96, Lausanne september 16-19) 1996.
- [21] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. In *Proc. 5th IEEE Int. Conf. on Computer Vision*, pages 694–699, ICCV’95, Boston, MA, June 1995.
- [22] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *Int. J. of Computer Vision*, 22(1):61–79, 1997.
- [23] V. Caselles and J.M. Morel. An Axiomatic Approach to Image Interpolation. *IEEE Trans. Image Processing*, page to appear, 1997.
- [24] V. Caselles and C. Sbert. What is the Best Causal Scale Space for Three-Dimensional Images? *SIAM J. Appl. Math.*, 56(4):1199–1246, 1996.

- [25] F. Catté and F. Dibos. A Morphological Scheme for Mean Curvature Motion and Application to Anisotropic Diffusion and Motion of Level Sets. In *Proc. third IEEE Int. Conf. on Image Processing*, pages I-26-30, ICIP-94, Austin (Texas), November 13-16 1994.
- [26] F. Catté, J.-M. Morel, P.-L. Lions, and T. Coll. Image selectives smoothing and edge detection by nonlinear diffusion. *SIAM J. Numer. Anal.*, 29:182-193, 1992.
- [27] A. Chambolle and P.-L. Lions. Image Recovery via Total Variation Minimization and Related Problems. *Numer. Math.*, 76:167-188, 1997.
- [28] L.-O. Chua and L. Yang. Cellular Neural Network: Applications. *IEEE Trans. Circuits & Systems*, 35:1273-1290, 1988.
- [29] L.-O. Chua and L. Yang. Cellular Neural Network: Theory. *IEEE Trans. Circuits & Systems*, 35:1257-1272, 1988.
- [30] I. Cohen. Modèles déformables 2D et 3D : application à la segmentation d'images médicales, 1992. Ph.D. thesis, Université Paris IX-Dauphine.
- [31] L.D. Cohen. On active contour models and balloons. *CVGIP: Image Understanding*, 53:211-218, 1991.
- [32] M.-A. Cohen. Sustained Oscillation in a Symetric Cooperative-Competitive Neural Networ: Disproof of a conjecture about Content Adressable Memory. *Neural Network.*, 1:217-221, 1988.
- [33] M.-A. Cohen and S. Grossberg. Absolute stability of global pattern formation and parallel memory storage bye competitive neural networks. *IEEE Trans. Systems Man & Cybernetics.*, SMC-12, n.5:815, 1983.
- [34] M.-A. Cohen and S. Grossberg. Neural Dynamics of brightness perception: Features, boundaries, diffusion and resonance. *Perception and Psychophysics*, 36:428-456, 1984.
- [35] T. Cohignac, F. Eve, F. Guichard, C. Lopez, and J.M. Morel. Numerical analysis of the fundamental equation of image processing, 1992. RT 9254, Ceremade ParisIX - Dauphine.
- [36] G.-H. Cottet and M. El Ayyadi. Nonlinear PDE operators with memory terms for image processing. In *Proc. third IEEE Int. Conf. Image Processing*, pages I-481-484, ICIP-96, Lausanne september 16-19 1996.
- [37] G.-H. Cottet and M. El Ayyadi. A Volterra type model for image processing. *IEEE Trans. Image Processing*, 1997, To appear.

- [38] G.-H. Cottet, J. Demongeot, M. El Ayyadi, and F. Leitner. Image segmentation using snake-splines and anisotropic diffusion operators. *submitted to IEEE Trans. PAMI.*, 1996.
- [39] G.H. Cottet. Modèle de réaction-diffusion pour des réseaux de neurones stochastiques et déterministes. *C.R.Acad.Sc.*, 312:217–221, 1991.
- [40] G.H. Cottet. Neural networks: continuous approach and applications to image processing. *Journal of Biological Systems*, 3:1131–1139, 1995.
- [41] G.H. Cottet and L. Germain. Image processing through reaction combined with nonlinear diffusion. *Math. Comp.*, 61:659–673, 1993.
- [42] P. Degond and S. Mas-Gallic. The weighted particle method for convection-diffusion equations. II: The anisotropic case. *Math. Comp.*, 53:485–508, 1989.
- [43] R. Deriche. Using Canny’s Criteria to derive a recursively implemented optimal edge detector. *Int. J. of Comp. Vision*, pages 167–187, 1987.
- [44] F. Dibos. Analyse multiéchelle invariante par transformations projective, 1995. RT 9505, Ceremade ParisIX - Dauphine, Janvier.
- [45] D.W. Dong. Dynamic Properties of Neural Networks, 1991. Ph.D. thesis, California Institute of technology, Pasadena, CA.
- [46] D.W. Dong and J.J. Hopfield. Dynamic properties of neural networks with adapting synapses. In *Network: Computation in Neural Systems*, volume 3(3), pages 267–283, 1992.
- [47] D.W. Dong and J.J. Hopfield. Dynamics of interconnection development within visual cortex. In *Proc IJCNN, Baltimore*, volume 3, pages 85–90, 1992.
- [48] R. Edwards. Neural Networks and Neural Fields: Discrete and Continuous Space Neural Models, 1994. Ph.D. thesis, University of Victoria, Victoria.
- [49] R. Edwards. Approximation of Neural Network Dynamics by Reaction-Diffusion Equations. *Math. Meth. Appl. Sci.*, 19, 1996.
- [50] O. Faugeras. Sur l’évolution de courbes simples du plan projectif réel. *C. R. Acad. Sci. Paris*, t. 317, Série I:565–570, 1993.
- [51] O. Faugeras. Three-Dimensional Computer Vision; a Geometric Viewpoint. *MIT Press*, 1993.
- [52] O. Faugeras and M. Hebert. The representation, recognition and locating of 3-d objects. *Int. J. of Robotic Res.*, 5:25–52, 1986.

- [53] O. Faugeras and R. Keriven. Some recent results on the projective evolution of 2-D curves. In *Proc. third IEEE Int. Conf. on Image Processing*, pages III-13-16, ICIP-95, Lausanne October 16-19 1995.
- [54] L.M.J. Florack. The Syntactical Structure of Scalar Image, 1993. Ph.D. thesis, Universiteit Utrecht, Faculteit Geneeskunde, Netherland.
- [55] C. Gégout. Techniques Évolutionnaires pour l'apprentissage des Réseaux de Neurones à Coefficients Réels, 1992. Ph.D. thesis, École Polytechnique Paris.
- [56] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. & Machine Intell.*, 6:721-741, 1984.
- [57] G. Gerig, O. Kübler, R. Kikinis, and F.A. Jolesz. Nonlinear anisotropic filtering of MRI data. *IEEE Trans. Medical Imaging*, 11:221-232, 1992.
- [58] S. Grossberg. Adaptive pattern classification and universal recoding: I. Parallel development and coding of neural feature detectors. *Biological Cybernetics.*, 23:121-134, 1976.
- [59] S. Grossberg. Competition, decision, and consensus. *Journal of Mathematical Analysis and Applications.*, 66:470-493, 1978.
- [60] S. Grossberg. Nonlinear Neural Networks: Principles, Mecanisms, and Architectures. *Neural Network.*, 1:17-61, 1988.
- [61] D.O. Hebb. *The Organization of Behavior: A Neuropsychological Theory*. John Wiley, New York, 1948.
- [62] M.-W Hirsch. Systems of differential equation that are competitive or cooperative I: Limit sets. *SIAM Journal of Mathematical Analysis.*, 13:167-179, 1982.
- [63] M.-W Hirsch. Systems of differential equation that are competitive or cooperative II: Convergence almost everywhere. *SIAM Journal of Mathematical Analysis.*, 16:423-439, 1985.
- [64] M.-W Hirsch. Convergent Activation Dynamics in Continuous Time Networks. *Neural Network.*, 2:331-349, 1989.
- [65] J.J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proc. Nat. Acad. Sci. U.S.A.*, 79:2554-2558, 1982.
- [66] J.J. Hopfield. Neurons with graded response have collective computational properties like those of two-states neurons. *Proc. Nat. Acad. Sci. U.S.A.*, 81:3088-3092, 1984.

- [67] R.A. Hummel. Representations based on zero-crossings in scale space. *IEEE Conf. Computer Vision & Pattern Recognition (CVPR'86) Miami Beach.*, pages 204–209, 1986.
- [68] R. Illner and H. Neunzert. Relative entropy maximization and directed diffusion equations. *Math. Meth. Appl. Sci.*, 16:545–554, 1993.
- [69] R. Illner and J. Tie. On directed diffusion with measurable background. *Math. Meth. Appl. Sci.*, 16:681–690, 1993.
- [70] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *Int. J. of Computer Vision*, 1:321–332, 1988.
- [71] B. Kimia, A.R. Tannenbaum, and S.W. Zucker. On the Evolution of Curves via a Function of Curvature. I. The Classical Case . *J. of Mathematical Analysis and Applications*, 163(2):438–458, 1992.
- [72] B. Kimia, A.R. Tannenbaum, and S.W. Zucker. Shapes, shocks and deformations i: The compenents of two-dimensional shape and the reaction- diffusion space. *Int. J. of Computer Vision*, 15:189–224, 1995.
- [73] R. Kimmel. Curve Evolution on Surfaces, 1995. Ph.D. thesis, Technion Institute of technology, Israel.
- [74] J.J. Koenderink. The sructure of images. *Biol. Cybernet.*, 50:363–370, 1984.
- [75] T. Kohonen. *Self-Organization and Associative Memory*. New York, Springer-Verlag, 1984.
- [76] T. Kurosawa, H. Tsuchiya, Y. Maruyama, H. Ohtsuka, and K. Nakazato. A new bi-level reproduction of continuous tone images. In *Proc. 3rd Interna. Conf. Image Processing and its Applications. IEE Publ. London*, pages 82–86, 1986.
- [77] X. Li and T. Chen. Nonlinear diffusion with multiple edginess thresholds. *Pattern Recognition.*, 27:1029–1037, 1994.
- [78] P.L. Lions, S. Osher, and L. Rudin. Denoising and Deblurring Algorithms with Constrained Nonlinear PDE's. *SIAM J. Numer. Anal.*, submitted.
- [79] D. Marr and E. Hildreth. Theory of edge detection. *Pro. Roy. Soc. London Ser. B*, 207:187–217, 1980.
- [80] J.-M. Morel and S. Solimini. *Variational methods in imagesegmentation*. Birkhäuser, Boston, 1994.
- [81] D. Mumford and J. Shah. Boundary detection by minimizing functionals. *IEEE Conf. Computer Vision & Pattern Recognition (CVPR'85) San Francisco.*, pages 22–26, 1985.

- [82] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math.*, 42:577–685, 1989.
- [83] M. Nitzberg, D. Mumford, and T. Shiota. *Filtering, segmentation and depth*. Lecture Notes in Comp. Science, 662, Springer, Berlin, 1993.
- [84] M. Nitzberg and T. Shiota. Nonlinear Image Filtering with Edge and Corner Enhancement. *IEEE Trans. Pattern Anal. & Machine Intell.*, 14(7):(826–833), 1992.
- [85] N. Nordström. Biased anisotropic diffusion - a unified regularization and diffusion approach to edge detection. *Image Vision Comput.*, 8:(318–327), 1990.
- [86] S. Osher and L. Rudin. Feature-oriented image enhancement using shock filters. *SIAM J. Numer. Anal.*, 27:919–940, 1990.
- [87] S. Osher and L. Rudin. Shock and Other Nonlinear Filtering Applied to Image Processing. In *Proc. SPIE Application of Digital Image Processing XIV, San-Diego*, volume 1567, pages 16–22, 1991.
- [88] S. Osher and J. Sethian. Fronts propagating with curvature dependent speed: algorithms based on the Hamilton-Jacobi formulation. *J. Comp. Physics*, 79:12–49, 1988.
- [89] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. In *Proc. IEEE Comp. Soc. Workshop on Computer Vision, IEEE Computer Society Press, Washington*, pages 16–22, 1987.
- [90] P. Perona and J. Malik. A network for multiscale image segmentation. *Proc. IEEE Int. Symp. Circuit and Systems (ISCAS-88)*, pages 2565–2568, 1988.
- [91] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. & Machine Intell.*, 12:629–639, 1990.
- [92] P. Perona, T. Shiota, and J. Malik. *Anisotropic diffusion*. In Bart M. ter Haar Romeny (Ed.), *Geometry-driven diffusion in computer vision*, Kluwer, Dordrecht, 1994.
- [93] F. J. Pineda. Generalization of Backpropagation to Recurrent and Higher Order Neural Networks. In *Neural Information Processing Systems Conference*, volume 1, pages 602–611. American Institute of Physics, 1987.
- [94] C.B. Price, P. Wanback, and A. Oosterlinck. Application of reaction-diffusion equations to image processing. In *Proc. 3rd Interna. Conf. Image Processing and its Applications. IEE Publ. London*, pages 49–53, 1989.
- [95] L. Rudin. Images, numerical analysis of singularities, and shock filters, 1987. Ph.D. thesis, California Institute of technology, Pasadena, CA.

- [96] L.I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica-D*, 60:259–268, 1992.
- [97] D.-E. Rumelhart, G.-E. Hinton, and R.-J. Williams. learning Internal Representations by Error Propagation. In D.E. Rumelhart & J.L. McClelland), editor, *Parallel Distributed Processing*, pages 219–230. Volume 1, Cambridge, MA: MIT Press, 1986.
- [98] P. Saint-Marc, J.S. Chen, and G. Medioni. Adaptive smoothing: a general tool for early vision. *IEEE Trans. Pattern Anal. & Machine Intell.*, 13:514–529, 1990.
- [99] G. Sapiro and A. Tannenbaum. Affine invariant scalespace. *Int. J. of Computer Vision*, 11:24–44, 1993.
- [100] G. Sapiro and A. Tannenbaum. On Affine Plane Curve Evolution. *J. of Functional Analysis*, 119(1):79–119, 1993.
- [101] G. Sapiro and A. Tannenbaum. Area and length preserving geometric invariant scale-space. *IEEE Trans. Pattern Anal. & Machine Intell.*, 17:67–72, 1995.
- [102] J. Shah. A Common Framework for Curve Evolution, Segmentation and Anisotropic Diffusion. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 1996.
- [103] J. Shah. Curve Evolution and Segmentation Functionals: Application to Color Images. In *Proc. third IEEE Int. Conf. Image Processing*, pages I-461–484, ICIP-96, Lausanne september 16-19 1996.
- [104] B.M. ter Haar Romeny. *Geometry-driven diffusion in computer vision*. Kluwer, Dordrecht, 1994.
- [105] B.M. ter Haar Romeny, L.M.J. Florack, J.J. Koenderink, and M.A. Viergever. *Scale space: Its natural operators and differential invariants*. A.C.F. Colchester, D.J Hawkes (Eds.), Information processing in medical imaging, Lecture Notes in Comp. Science, 511, Springer, Berlin, 1991.
- [106] J.-L. van Hemmen. Spin-glass models of neural network. *Phys. Rev. Lett.*, 34:3435–3445, 1986.
- [107] J. Weickert. Anisotropic Diffusion Filters for Image Processing: Based Quality Control, 1993. preprint.
- [108] J. Weickert. Scale-space properties of nonlinear diffusion filtering with a diffusion tensor, 1994. preprint.
- [109] J. Weickert. *Multiscale texture enhancement*, pages 230–237. V. Hlavác, R. Sara (Eds.), Computer analysis of images and patterns, Lecture Notes in Comp. Science, 970, Springer, Berlin, 1995.

- [110] J. Weickert. Anisotropic Diffusion in Image Processing, January 1996. Ph.D. thesis, Dept. of Mathematics, University of Kaiserslautern, Germany.
- [111] J. Weickert and M.A. Viergever B.M. ter Haar Romeny. Conservative image transformations with restoration and scale-space properties. In *Proc. third IEEE Int. Conf. Image Processing*, pages I-465-468, ICIP-96, Lausanne september 16-19 1996.
- [112] R. Whitaker and G. Gerig. *Vector-valued diffusion*. B.M. ter Haar Romeny (Ed.), Geometry-driven diffusion in computer vision, Kluwer, Dordrecht, 1994.
- [113] R.T. Whitaker. Geometry-Limited Diffusion in the characterization of Geometric Patches in Images. *CVGIP: Image Understanding*, pages 111-120, 1993.
- [114] R.T. Whitaker and M.A. Abidi. A Weighted Sum of Curvatures for Image Denoising: A Comparative Analysis. *IEEE Trans. Image Processing*, page submitted, 1996.
- [115] R.T. Whitaker and S.M. Pizer. A multiscale approach to nonuniform diffusion. *CVGIP: Image Understanding*, pages 99-110, 1993.
- [116] S. Wiggins. Global Bifurcations and Chaos: Analytic Methods. *Springer Verlag*, New York, 1988.
- [117] A.P. Witkin. Scale-space filtering. In *Proc. of IJCAI*, pages 1019-1021, Karlsruhe 1983.
- [118] Z.B. Xu. Global Convergence and Asymptotic Stability of Asymmetric Hopfield Neural Networks. *Journal of Mathematical Analysis and Applications.*, 191:405-427, 1995.
- [119] A. Yuille and T. Poggio. Scaling theorems for zero crossings. *IEEE Trans. Pattern Anal. & Machine Intell.*, 8, 1986.