



HAL
open science

Methodological and applied contributions to the deformable models framework

Jean-Philippe Pons

► **To cite this version:**

Jean-Philippe Pons. Methodological and applied contributions to the deformable models framework. Human-Computer Interaction [cs.HC]. Ecole des Ponts ParisTech, 2005. English. NNT: . tel-00457461

HAL Id: tel-00457461

<https://pastel.hal.science/tel-00457461>

Submitted on 17 Feb 2010

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

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

THÈSE

présentée à l'
ÉCOLE NATIONALE DES PONTS ET CHAUSSÉES,
Marne-la-Vallée, France

pour l'obtention du titre de
DOCTEUR DE L'ÉCOLE NATIONALE
DES PONTS ET CHAUSSÉES

Spécialité : Informatique

Jean-Philippe Pons

Contributions Méthodologiques et
Appliquées à la Méthode des
Modèles Déformables

Soutenue le 18 novembre 2005.

Directeur	Pr. Olivier Faugeras	INRIA, Sophia-Antipolis, France
Rapporteurs	Pr. Luc Van Gool	Katholieke Universiteit Leuven, Belgique
	Pr. Guillermo Sapiro	University of Minnesota, Minneapolis, USA
	Pr. Stefano Soatto	University of California, Los Angeles, USA
Examineurs	Pr. Jean-Daniel Boissonnat	INRIA, Sophia-Antipolis, France
	Dr. Renaud Keriven	ENPC, Marne-la-Vallée, France

PHD THESIS

presented at the
ÉCOLE NATIONALE DES PONTS ET CHAUSSÉES,
Marne-la-Vallée, France

for obtaining the title of
DOCTOR OF SCIENCE
Specialized in Computer Science

Jean-Philippe Pons

Methodological and Applied
Contributions to the
Deformable Models Framework

November 18, 2005.

Advisor	Pr. Olivier Faugeras	INRIA, Sophia-Antipolis, France
Reviewers	Pr. Luc Van Gool	Katholieke Universiteit Leuven, Belgium
	Pr. Guillermo Sapiro	University of Minnesota, Minneapolis, USA
	Pr. Stefano Soatto	University of California, Los Angeles, USA
Examinators	Pr. Jean-Daniel Boissonnat	INRIA, Sophia-Antipolis, France
	Dr. Renaud Keriven	ENPC, Marne-la-Vallée, France

À Mathieu, Amande et Éléonore.

Remerciements

Je tiens d'abord à remercier Olivier Faugeras de m'avoir accueilli au sein du projet Odyssée et dirigé ces travaux.

J'exprime toute ma gratitude à Renaud Keriven pour avoir codirigé ces travaux, mais aussi pour m'avoir guidé vers le monde passionnant de la recherche, au fil de mes années à l'École Nationale des Ponts et Chaussées. Son dynamisme et son enthousiasme communicatifs sont pour beaucoup dans mon désir de continuer sur la voie de la recherche.

Je remercie également Luc Van Gool, Guillermo Sapiro et Stefano Soatto pour avoir accepté d'être mes rapporteurs, en dépit du travail important que cela représente. Je remercie également Jean-Daniel Boissonnat de sa participation au jury.

Plusieurs personnes ont largement contribué à certains des travaux présentés dans ce manuscrit. Je remercie donc Guillaume Charpiat, Gerardo Hermosillo et Florent Ségonne pour ces fructueuses collaborations.

Merci à tous les membres du projet Odyssée et en particulier à mes compères doctorants et leur indéfectible bonne humeur, grâce à qui je garderai un excellent souvenir de mon séjour à l'INRIA.

Cette section ne serait pas complète si je n'évoquais pas les heureux événements qui ont ponctué ces trois années. Je termine donc sur une pensée toute particulière pour Éléonore, qui a fait de moi l'heureux père du tendre Mathieu et de l'espiègle Amande.

Titre : Contributions méthodologiques et appliquées à la méthode des modèles déformables.

Résumé : Les modèles déformables fournissent un cadre flexible pour traiter divers problèmes de reconstruction de forme en traitement d'images. Ils ont été proposés initialement pour la segmentation d'images, mais ils se sont aussi révélés adaptés dans de nombreux autres contextes en vision par ordinateur et en imagerie médicale, notamment le suivi de régions, la stéréovision, le "shape from shading" et la reconstruction à partir d'un nuage de points. Les éléments clés de cette méthodologie sont l'élaboration d'une fonctionnelle d'énergie, le choix d'une procédure de minimisation et d'une représentation géométrique.

Dans cette thèse, nous abordons ces trois éléments, avec pour but d'élargir le champ d'application des modèles déformables et d'accroître leur performance. En ce qui concerne la représentation géométrique, nous venons à bout de la perte de la correspondance ponctuelle et de l'impossibilité de contrôler les changements de topologie avec la méthode des ensembles de niveau. Nous proposons deux applications associées dans le domaine de l'imagerie médicale : la génération de représentations dépliées du cortex cérébral avec préservation de l'aire, et la segmentation de plusieurs tissus de la tête à partir d'images par résonance magnétique anatomiques. En ce qui concerne la procédure de minimisation, nous montrons que la robustesse aux minima locaux peut être améliorée en remplaçant une descente de gradient traditionnelle par un flot de minimisation spatialement cohérent. Enfin, en ce qui concerne l'élaboration de la fonctionnelle d'énergie, nous proposons une nouvelle modélisation de la stéréovision multi-caméras et de l'estimation du mouvement tridimensionnel non-rigide, fondée sur un critère de mise en correspondance global et basé image.

Mots clés : modèle déformable, contour actif, ensembles de niveau, méthode variationnelle, correspondance ponctuelle, topologie, segmentation, dépliement, cerveau, cortex, stéréovision, mouvement.

Title: Methodological and applied contributions to the deformable models framework.

Abstract: Deformable models constitute a flexible framework to address various shape reconstruction problems in image processing. They have been initially proposed for the purpose of image segmentation, but they have also proven successful in many other contexts in computer vision and in medical imaging, including region tracking, stereovision, shape from shading and shape from unstructured point sets. The key elements of this framework are the design of an energy functional, the choice of a minimization procedure and of a geometric representation.

In this thesis, we tackle these three elements, with the objective of increasing the applicability and efficiency of deformable models. With regard to the geometric representation, we overcome the loss of the point correspondence and the inability to control topology changes with the level set method. We propose two associated applications in the field of medical imaging: the generation of unfolded area preserving representations of the cerebral cortex, and the segmentation of several head tissues from anatomical magnetic resonance images. With regard to the minimization procedure, we show that the robustness to local minima can be improved by substituting a spatially coherent minimizing flow to a traditional gradient descent. Finally, with regard to the design of the energy functional, we propose a novel modeling of multi-view stereovision and scene flow estimation with deformable models, relying on a global image-based matching score.

Keywords: deformable model, active contour, level set, variational method, point correspondence, topology, segmentation, unfolding, brain, cortex, stereovision, motion.

Contents

Remerciements	5
Résumé	7
Abstract	9
1 Introduction (version française)	15
1.1 Représentation géométrique des modèles déformables	16
1.1.1 Modèles explicites	17
1.1.2 Modèles implicites	18
1.2 Minimisation de la fonctionnelle d'énergie	19
1.3 Tendances majeures dans l'élaboration de la fonctionnelle d'énergie	22
1.3.1 Segmentation d'images	22
1.3.2 Stéréovision multi-caméras	25
1.4 Contributions de cette thèse	26
1.4.1 Contributions méthodologiques	27
1.4.2 Contributions appliquées	28
1.4.3 Contributions logicielles	29
2 Introduction	31
2.1 Geometric representation of deformable models	32
2.1.1 Explicit models	33
2.1.2 Implicit models	34
2.2 Minimization of the energy functional	35
2.3 Major trends in the design of the energy functional	37
2.3.1 Image segmentation	37
2.3.2 Multi-view stereovision	41
2.4 Contributions of this thesis	41
2.4.1 Methodological contributions	42
2.4.2 Applied contributions	43
2.4.3 Software contributions	43

I	Methods	45
3	Maintaining the Point Correspondence in the Level Set Framework	47
3.1	Motivation	47
3.2	Methods	48
3.2.1	Previous work on region tracking	49
3.2.2	Previous work on transport and diffusion of a material quantity . . .	50
3.2.3	LSID: Level sets with some interfacial data	50
3.2.4	LSPC: Level sets with a point correspondence	51
3.3	Numerical algorithms	52
3.3.1	Level set reinitialization and data extension	52
3.3.2	Keeping the point correspondence onto the initial interface	54
3.3.3	Finite-difference schemes	54
3.3.4	Overview of the algorithm	55
3.4	Experimental results	55
3.4.1	Definition of the error measures	55
3.4.2	2D experiments	56
3.4.3	3D experiments	59
3.4.4	Comments on the errors	59
3.5	Contributions of this chapter	63
4	Controlling Topology Changes in the Level Set Framework	65
4.1	Background	66
4.1.1	Topology	66
4.1.2	Digital topology	67
4.1.3	The topology preserving level set method	68
4.2	The topology preserving nested level set method	71
4.2.1	Digital topology criterion	71
4.2.2	Description of the algorithm	71
4.3	The genus preserving level set method	73
4.3.1	From simple points to multisimple points	74
4.3.2	Description of the algorithm	74
4.4	Experimental results	76
4.4.1	Synthetic data	76
4.4.2	Real data	79
4.5	Contributions of this chapter	82
5	Improving the Robustness to Local Minima with Spatially Coherent Minimizing Flows	85
5.1	Motivation	85
5.2	Abstract study	87
5.2.1	Designing new inner products	87

5.2.2	Designing new minimizing flows	88
5.3	Spatially coherent minimizing flows	89
5.3.1	The H^1 gradient flow	89
5.3.2	Motion decomposition	89
5.3.3	Intrinsic Gaussian smoothing	91
5.4	Experimental results	92
5.4.1	Shape warping	92
5.4.2	Tracking	93
5.5	Contributions of this chapter	93
II	Applications	97
6	Area Preserving Cortex Unfolding	99
6.1	Motivation	99
6.2	Area preserving surface motion	101
6.2.1	The total, local and relative area preserving conditions	101
6.2.2	Designing a relative area preserving tangential motion	102
6.2.3	The area relaxation term	102
6.2.4	Numerical methods	103
6.3	Experimental results	104
6.3.1	Synthetic data	104
6.3.2	Real data	106
6.4	Contributions of this chapter	108
7	Head Segmentation from MRI under Topological Constraints	109
7.1	Motivation	109
7.1.1	Problem statement	109
7.1.2	Previous work	110
7.1.3	Goals of our approach	112
7.2	Methods	112
7.2.1	Hidden Markov random field classification	112
7.2.2	Topology preserving nested level sets	113
7.2.3	Bayesian region-based deformable models evolution	114
7.3	Experimental results	115
7.4	Contributions of this chapter	118
8	Multi-View Stereo Reconstruction and Scene Flow Estimation with a Global Image-Based Matching Score	119
8.1	Introduction	120
8.1.1	Problem statement	120

8.1.2	Common photometric and geometric assumptions used for shape and motion estimation	120
8.1.3	Previous work on multi-view complete stereovision	121
8.1.4	Previous work on scene flow estimation	123
8.1.5	Motivations of our approach	124
8.2	Minimizing the prediction error	125
8.2.1	Stereovision	126
8.2.2	Scene flow	128
8.3	Some similarity measures	128
8.3.1	Cross correlation	129
8.3.2	Mutual information	130
8.4	Experimental results	130
8.4.1	Stereovision	131
8.4.2	Stereovision + scene flow	133
8.5	Contributions of this chapter	138
	Conclusion	139
	Conclusion (version française)	141
	Appendices	143
	A Formulas of Geometric and Differential Calculus	145
A.1	Some useful identities of intrinsic differential geometry	145
A.2	Some useful expressions for implicit interfaces	145
	B Numerical Schemes	149
B.1	Evolution schemes	149
B.1.1	Normal propagation	149
B.1.2	Passive advection	150
B.1.3	Mean curvature motion	150
B.1.4	Reinitialization	150
B.1.5	Extension	151
B.2	High-order WENO finite-difference schemes	151
B.2.1	Third-order WENO scheme	151
B.2.2	Fifth-order WENO scheme	151
B.3	Finite-difference discretization of the Laplace-Beltrami operator	152
	C Publications of the Author	153
	Bibliography	155

Chapter 1

Introduction (version française)

Les modèles déformable, également connus sous le nom de “snakes”, contours actifs, surfaces actives, contours et surfaces déformables, sont une technique informatique très prisée pour aborder divers problèmes de reconstruction de forme en traitement d’images. Un modèle déformable est une courbe, une surface, ou un objet géométrique de dimension supérieure, qui est positionné initialement par l’utilisateur et qui se déplace sous certaines contraintes et sous l’influence des données d’entrée, jusqu’à converger vers une solution plausible du problème considéré.

Cette méthodologie a plusieurs caractéristiques remarquables. Elle est très polyvalente à travers le choix d’une représentation de forme, et l’élaboration de l’équation d’évolution. De ce fait, elle peut être appliquée à de multiples types de données d’entrée (images bi-dimensionnelles, tri-dimensionnelles et 4D, données de stéréovision multi-caméras, nuages de points, ...) et à un large spectre de problèmes. Elle fut initialement proposée pour la segmentation d’images par Kass, Witkin et Terzopoulos dans [77], mais elle s’est révélée efficace dans de nombreux autres contextes en vision par ordinateur et en imagerie médicale, notamment pour le suivi de régions [115, 101], la reconstruction 3D par stéréoscopie [50, 75, 45, 88, 69, 57], le “shape from shading” [53, 74, 182], et la reconstruction 3D à partir d’un nuage de points [188, 10, 160].

De plus, l’utilisation d’une formulation géométrique et continue est très avantageuse. Elle conduit à un cadre mathématique rigoureux, elle procure une précision sub-voxelique, elle modélise correctement la continuité de la forme, et permet d’incorporer des hypothèses de régularité et des connaissances *a priori* complexes sur la forme recherchée. Il en découle une bonne robustesse aux données bruitées et incomplètes. Finalement, cette formulation permet d’intégrer naturellement de nombreuses informations locales dans une description mathématique cohérente d’une forme.

Les éléments clés de la méthode des modèles déformables sont l’élaboration de la fonctionnelle d’énergie, le choix d’une procédure de minimisation, et d’une représentation géométrique. Dans cette thèse, nous abordons ces trois éléments, avec pour but d’élargir le champ d’application des modèles déformables et d’accroître leur performance. Parmi les divers types de modèles déformables existant, nous nous concentrons sur les évolutions

géométriques provenant de la théorie des évolutions de courbe [54], et nous mettons particulièrement l’accent sur la représentation par ensembles de niveau (*level sets*) [112].

En ce qui concerne la représentation géométrique, nous venons à bout de la perte de la correspondance ponctuelle et de l’impossibilité de contrôler les changements de topologie avec la méthode des ensembles de niveau. Nous proposons deux applications associées dans le domaine de l’imagerie médicale : la génération de représentations dépliées du cortex cérébral avec préservation de l’aire, et la segmentation de plusieurs tissus de la tête à partir d’images par résonance magnétique (IRM) anatomiques. En ce qui concerne la procédure de minimisation, nous montrons que la robustesse aux minima locaux peut être améliorée en remplaçant une descente de gradient traditionnelle par un flot de minimisation spatialement cohérent. Enfin, en ce qui concerne l’élaboration de la fonctionnelle d’énergie, nous proposons une nouvelle modélisation de la stéréovision multi-caméras et de l’estimation du mouvement tridimensionnel non-rigide, fondée sur un critère de mise en correspondance global et basé image.

Le reste de ce chapitre introductif est organisé comme suit. Dans la section 1.1, nous discutons les avantages et les inconvénients des différentes représentations géométriques des modèles déformables. En particulier, nous mettons en lumière plusieurs limitations de la méthode des ensembles de niveau, qui motivent certaines de nos contributions méthodologiques. La section 1.2 est consacrée à l’équation d’évolution, et souligne la dommageable sensibilité des modèles déformables aux conditions initiales, à laquelle nous nous attaquons dans cette thèse. La section 1.3 traite de l’élaboration de la fonctionnelle d’énergie. Comme ce sujet est très dépendant des détails propres à chaque application, nous présentons seulement les tendances majeures dans deux applications importantes de la méthode des modèles déformables auxquelles notre travail contribue : la segmentation d’images et la stéréovision multi-caméras. Finalement, dans la section 1.4, nous résumons les contributions importantes de cette thèse et nous décrivons le contenu des chapitres suivants.

1.1 Représentation géométrique des modèles déformables

La littérature des modèles déformables regorge de différentes représentations géométriques. Le lecteur est prié de se reporter à [105] pour une revue complète de ces représentations. Souvent, on les divise en deux catégories : les modèles “paramétriques” et les modèles “géométriques”. Mais nous pensons que ces deux termes portent à confusion. Tout d’abord, nous préconisons d’utiliser “paramétré” plutôt que “paramétrique” pour désigner les objets géométriques donnés par une paramétrisation, et de réserver le terme “paramétrique” aux familles d’objets indexées par un petit nombre de paramètres. Deuxièmement, dans la littérature, les modèles géométriques réfèrent souvent exclusivement à la méthode des ensembles de niveau. De sorte que des représentations comme les maillages surfaciques non-structurés se retrouveraient classés comme “paramétriques”, alors qu’ils ne sont ni paramétriques, ni paramétrisés. . .

Dans cette thèse, nous faisons la distinction entre les représentations “explicites” et “implicites”. Les modèles explicites sont donnés sous la forme d’un ensemble de coordonnées tandis que les modèles implicites sont définis comme le niveau zéro d’une fonction scalaire de dimension supérieure. Parallèlement, nous distinguons les lois d’évolution “géométriques” et “non-géométriques”. Les évolutions géométriques ne dépendent pas d’une paramétrisation particulière. Elles ont pour origine la théorie des évolutions de courbe [54]. Dans ce type d’évolution, le champ de vitesse du modèle dépend typiquement de quantités géométriques intrinsèques comme la normale, la courbure, ainsi que des données d’entrée.

Les évolutions géométriques peuvent être implémentées très naturellement avec une représentation par ensembles de niveau, mais l’usage d’autres représentations, y compris explicites, est possible. Le choix d’une représentation explicite ou implicite pour implémenter une évolution géométrique est fortement relié au choix d’une perspective lagrangienne ou eulérienne.

1.1.1 Modèles explicites

Paramétrisation

Les courbes et les surfaces paramétrées sont un type important de modèles explicites. C’est ce type de représentation qui est proposé dans le modèle *snake* initial de Kass, Witkin et Terzopoulos [77]. Une multitude d’autres représentations paramétrées existent dans la littérature, fondées sur des fonctions de base (éléments finis [32, 96], B-splines [132], harmoniques de Fourier [151], etc.), ou sur une famille paramétrique (superquadriques [161, 9, 10]).

Un problème typique des modèles déformables paramétrés est que leur évolution et la reconstruction finale dépendent de la paramétrisation de la forme initiale. Plus exactement, ce problème est davantage lié à la définition de la fonctionnelle d’énergie qu’à la représentation géométrique elle-même : dans la plupart des cas, les distorsions métriques entre l’espace des paramètres et le modèle sont négligées. De ce fait, l’énergie n’est pas intrinsèque au modèle.

Il est à noter que certains modèles explicites ne sont pas paramétrisés. C’est le cas des systèmes de particules orientées [159], et des maillages non-structurés (triangulations [104], maillages simplexes [39], polyèdres [147]).

Topologie

Une limitation sérieuse de la plupart des modèles explicites est que leur topologie ne peut pas changer au cours de l’évolution pour s’adapter à la topologie des données. Plusieurs tentatives ont été faites pour résoudre ce problème. McInerney et Terzopoulos [98, 99] proposent des courbes et des triangulations de topologie adaptative, surnommées *T-snakes* et *T-surfaces*. Pendant l’évolution, le modèle est périodiquement rééchantillonné en calculant ses intersections avec une décomposition simpliciale de l’espace. Un étiquetage des

sommets de la grille simpliciale comme étant à l'intérieur ou à l'extérieur du modèle est maintenue. Outre qu'elle est coûteuse en temps de calcul, cette procédure impose une résolution spatiale fixe et uniforme. De plus, tous les mouvements ne sont pas admissibles : cette approche ne fonctionne que quand le modèle gonfle ou dégonfle partout.

Lachaud et Montanvert [85] ont recours au concept de δ -triangulation. Un paramètre de longueur δ est utilisé pour contrôler l'échantillonnage de la triangulation et pour détecter les auto-intersections en surveillant la distance entre les paires de sommets voisins et non-voisins. Cette approche est coûteuse, même lorsque les calculs sont optimisés par une structure d'*octree*.

Delingette et Montagnat [39, 40] proposent de modifier la topologie d'un maillage simplexe dynamique grâce à des opérations topologiques élémentaires, mais leur approche nécessite des interventions manuelles en 3D.

Ainsi, une gestion entièrement automatique et efficace des changements de topologie des modèles explicites, en trois dimensions et plus, reste un problème ouvert. Cela explique le grand intérêt suscité par la représentation par ensembles de niveau dans la littérature des modèles déformables.

1.1.2 Modèles implicites

Les modèles implicites ne se limitent pas à la méthode des ensembles de niveau [112]. Les surfaces algébrique polynômiales [160] et les superquadriques/hyperquadriques implicites [9, 30] font également partie de cette catégorie. Pourtant, parmi les modèles implicites, la méthode des ensembles de niveau est de loin la plus puissante. Elle couvre un spectre beaucoup plus large d'applications, car elle peut gérer des géométries complexes, alors que les autres se limitent à une famille réduite de formes.

La méthode des ensembles de niveau, introduite par Osher et Sethian dans [112] (un travail similaire dans le domaine de la mécanique des fluides [41, 42] a récemment refait surface), est une technique établie pour représenter des interfaces en mouvement en deux dimensions ou plus. Elle consiste à représenter l'interface implicitement comme le niveau zéro d'une fonction scalaire de dimension supérieure. Le mouvement de l'interface se traduit alors en une évolution de la fonction de niveau selon une EDP (équation aux dérivées partielles) eulérienne.

D'un côté, cette approche a plusieurs avantages par rapport à une représentation lagrangienne explicite de l'interface : il n'y pas besoin de paramétrisation, les changements de topologie sont gérés automatiquement, n'importe quel nombre de dimensions est possible, les propriétés géométriques intrinsèques comme la normale ou la courbure se calculent aisément à partir de la fonction de niveau. Enfin, et ce n'est pas le moins important, la théorie des solutions de viscosité [34] fournit des schémas numériques robustes et de puissants résultats mathématiques pour l'EDP d'évolution.

D'un autre côté, plusieurs limitations restreignent le domaine d'application de la méthode des ensembles de niveau. D'abord, la dimensionnalité supérieure rend la méthode des ensembles de niveau beaucoup plus coûteuse en temps de calcul que les représentations

explicites. Beaucoup d'efforts ont été faits pour atténuer ce problème, conduisant à la méthodologie de bande mince [1] et plus récemment à la méthode des ensembles de niveau rapide et locale par EDP [118].

De plus, dans sa formulation initiale, la méthode des ensembles de niveau ne peut représenter que des variétés de codimension un sans bords, telles que des courbes fermées dans \mathbb{R}^2 et des surfaces fermées dans \mathbb{R}^3 . Plusieurs approches ont été proposées pour gérer une codimension strictement supérieure à un. Ambrosio et Soner [5] proposent de faire évoluer une hypersurface correspondant à un ϵ -voisinage de la variété. Par exemple, dans le cas d'une courbe dans \mathbb{R}^3 , cela revient à faire évoluer un voisinage tubulaire de la courbe de faible rayon ϵ . Cette idée a été utilisée dans [90] pour segmenter des vaisseaux sanguins à partir d'images médicales. Un inconvénient de cette approche est que la variété considérée ne peut être localisée précisément. Une approche plus rigoureuse de Burchard, Cheng, Merriman et Osher [20] consiste à représenter une courbe comme l'intersection de deux hypersurfaces, et d'exprimer son mouvement comme l'évolution de deux fonctions de niveau. Dans [59], l'auteur envisage l'extension de cette idée à n'importe quelle codimension k par l'évolution simultanée de k fonctions de niveau, au prix de difficultés numériques grandissantes. Une idée similaire est proposée par Solem et Heyden [149] afin de représenter des surfaces à bords avec la méthode des ensembles de niveau.

D'autres écueils de la méthode des ensembles de niveau ont émergé récemment. L'un d'eux est intimement lié au point de vue implicite et à l'absence de paramétrisation : la correspondance ponctuelle est perdue pendant l'évolution. En d'autres termes, on ne sait pas comment chaque point ou chaque partie de l'interface bouge. Ce n'est pas une surprise, puisque la fonction de niveau véhicule une description purement géométrique de l'interface. Plus généralement, il est impossible de gérer des données associées à l'interface en mouvement dans le cadre traditionnel des ensembles de niveau. Cela réduit considérablement le spectre des applications possibles. Dans le chapitre 3, nous proposons une méthode, fondée sur un système couplé d'EDP eulériennes, pour venir à bout de cette limitation.

La faculté de gérer automatiquement les changements de topologie a longtemps été un argument en faveur de la méthode des ensembles de niveau, par rapport aux modèles déformables explicites. Mais ce comportement n'est pas souhaitable dans certaines applications où l'on dispose d'une connaissance *a priori* de la topologie cible. C'est typiquement le cas pour la segmentation des images biomédicales, pour laquelle la topologie des organes et leurs relations topologiques mutuelles sont prescrites par les connaissances anatomiques. Le chapitre 4 traite de méthodes, inspirées de la topologie digitale, visant à exercer un contrôle sur la topologie pendant une évolution par ensembles de niveau.

1.2 Minimisation de la fonctionnelle d'énergie

L'élaboration de l'équation d'évolution régissant le mouvement des modèles déformables suit deux approches principales. Dans la formulation *variationnelle*, le problème s'exprime comme la minimisation d'une fonctionnelle d'énergie. Cette fonctionnelle est définie de

sorte que des configurations de faible énergie indiquent une bonne adéquation avec les données d'entrée et l'information *a priori* (typiquement des hypothèses de régularité). La solution est définie comme une forme réalisant un minimum global de l'énergie. Dans la formulation par *force dynamique*, le mouvement est donné *ad hoc*, souvent comme une combinaison de forces internes, définies à partir du modèle lui-même et destinées à le garder lisse pendant la déformation, et de forces externes, définies à partir des données d'entrée. La solution est alors vue comme un équilibre de ces forces.

Dans cette thèse, nous donnons notre préférence à la formulation variationnelle, car elle offre plusieurs avantages. Premièrement, elle semble moins empirique et apporte une meilleure compréhension de la modélisation. Ceci est particulièrement vrai quand la fonctionnelle d'énergie dérive d'une modélisation statistique, comme le *maximum a posteriori* (MAP) d'une formulation bayésienne [157]. Deuxièmement, dans certains cas, on peut prouver mathématiquement l'existence d'un minimum global de l'énergie et le caractère bien posé de la procédure de minimisation. Enfin, une fois que le problème variationnel est posé, il peut être traité avec diverses procédures de minimisation, selon l'information *a priori* et le temps dont on dispose. Cette section porte sur le choix de cette procédure de minimisation.

En général, une minimisation exacte de la fonctionnelle d'énergie est intraitable de par le nombre considérable d'inconnues. Le *recuit simulé* [152] et la *programmation dynamique* [6] ont été proposés pour calculer un minimum global, mais le premier est très lent en pratique et le deuxième ne s'applique qu'à une forme particulière de fonctionnelles d'énergie. Plus récemment, les *graph cuts* se sont révélés être une puissante méthode de minimisation d'énergie permettant de trouver un minimum global ou un minimum local au sens fort. Au cours des dernières années, cette méthode a été appliquée avec succès à plusieurs problèmes de la vision par ordinateur, notamment la stéréovision multi-caméras [80] et la segmentation d'images [18]. Mais les *graph cuts* ont deux importantes limitations : ils ne s'appliquent pas à n'importe quelle énergie [81], et lorsqu'ils s'appliquent, ils sont coûteux en temps de calcul.

Par conséquent, dans la plupart des cas, une stratégie sous-optimale est adoptée, fondée sur le calcul des variations. Une condition nécessaire d'optimalité déduite de la fonctionnelle d'énergie est substituée au problème variationnel initial. Il s'agit de l'*équation d'Euler-Lagrange*, qui caractérise les minima et les maxima locaux de l'énergie. Celle-ci ne peut être résolue exactement, tout comme le problème variationnel initial. Aussi, il faut utiliser une méthode évolutive partant d'une estimation initiale de la solution. De ce fait, une notion artificielle de temps est ajoutée au problème, ce qui constitue le principe central de la méthode des modèles déformables : ainsi, la résolution du problème se traduit par l'évolution temporelle d'un objet géométrique.

Lorsqu'il s'agit de modèles déformables paramétrés, la résolution de l'équation d'Euler-Lagrange est effectuée typiquement en mettant à jour les paramètres du modèle avec une descente de gradient, ou un schéma numérique à convergence rapide tel que la méthode du gradient conjugué, la méthode de Newton, ou la méthode de Levenberg-Marquardt. Ces

évolutions sont non-géométriques, car elles dépendent de la paramétrisation du modèle. Elles sortent donc du cadre de cette thèse.

Une autre procédure de minimisation a été inaugurée par Chan et Vese dans [25]. Elle consiste à utiliser la méthode des ensembles de niveau à la fois comme une représentation géométrique et comme un cadre d'optimisation. En d'autres termes, la fonctionnelle d'énergie est définie directement à partir de la fonction de niveau ϕ , et la minimisation s'opère également par rapport à ϕ . Dans cette approche, l'intégrale d'une quantité f le long de la frontière du modèle ou dans la région intérieure/extérieure s'expriment à l'aide de la distribution de Dirac δ et de la fonction de Heaviside H :

$$\int_{\Gamma} f(\mathbf{x}) d\mathbf{x} = \int_{\mathbb{R}^n} \delta(\phi(\mathbf{x})) f(\mathbf{x}) \|\nabla\phi(\mathbf{x})\| d\mathbf{x} , \quad (1.1)$$

$$\int_{\text{in}} f(\mathbf{x}) d\mathbf{x} = \int_{\mathbb{R}^n} [1 - H(\phi(\mathbf{x}))] f(\mathbf{x}) d\mathbf{x} , \quad (1.2)$$

$$\int_{\text{out}} f(\mathbf{x}) d\mathbf{x} = \int_{\mathbb{R}^n} H(\phi(\mathbf{x})) f(\mathbf{x}) d\mathbf{x} . \quad (1.3)$$

Cette approche a acquis une certaine popularité en segmentation d'images [25, 170, 114, 56, 130]. Pourtant, elle soulève plusieurs problèmes pratiques et conceptuels. Tout d'abord, elle est spécifique à une représentation géométrique particulière. Ensuite, le recours à une version ϵ -régularisée de δ et H dans l'implémentation est assez inélégante. Enfin, l'évolution qui en découle est non-géométrique. Elle dépend des valeurs initiales de ϕ en dehors du niveau zéro. Des composantes déconnectées du modèle peuvent apparaître, ce qui n'est pas compatible avec une évolution de courbe.

De plus, une erreur fréquente avec cette approche est de supposer que la propriété de distance signée est préservée dans le calcul de la variation de l'énergie par rapport à une variation de ϕ . Par exemple, dans [56], les auteurs proposent de segmenter le cortex cérébral avec deux modèles déformables couplés, et avec un *a priori* sur leur distance mutuelle. Cette distance est obtenue comme la valeur absolue de la fonction distance signée représentant chaque interface. À cause de l'abus mentionné ci-dessus, la dérivation de la fonctionnelle d'énergie dans ce travail est erronée. Une dérivation le long de l'espace des fonctions distances n'est pas possible non plus, car cet espace n'est pas une variété différentielle.

Dans cette thèse, nous nous concentrons sur une procédure de minimisation appelée *flot géométrique de gradient*. Il s'agit de l'évolution géométrique obtenue en suivant la direction de descente la plus rapide d'une énergie intrinsèque. Sa signification mathématique est détaillée au chapitre 5. Par exemple, la minimisation de l'aire d'un modèle conduit au fameux mouvement par courbure moyenne [54]. Un autre important flot géométrique de gradient s'obtient en minimisant l'aire du modèle dans un espace de Riemann, doté d'une métrique dépendante d'une image : ce sont les *contours actifs géodésiques* proposés par Caselles, Kimmel et Sapiro dans [24] pour la détection des contours d'une image.

La plupart des fonctionnelles d'énergie étant non-convexes, les flots géométriques de gradient ont une forte probabilité d'être piégés dans un minimum local. De plus, ce

minimum local dépend de la position de la forme initiale. Si cette dernière est trop éloignée de la configuration finale attendue, l'évolution peut rester coincée dans un état aberrant. Cette sensibilité aux conditions initiales restreint sérieusement le champ d'application et la performance des méthodes à base de modèles déformables.

Il y a essentiellement deux manières de traiter ce problème : positionner le modèle initial très près de la configuration finale attendue, ou bien utiliser une stratégie multi-résolution en allant du grossier vers le plus fin, c'est à dire réaliser l'optimisation sur une série de formes et de données lissées et sous-échantillonnées. Dans le chapitre 5, nous introduisons une troisième manière d'approcher le problème des minima locaux indésirables : une élaboration soigneuse de nouveaux flots géométriques de minimisation.

1.3 Tendances majeures dans l'élaboration de la fonctionnelle d'énergie

1.3.1 Segmentation d'images

Les multiples méthodes à base de modèles déformables dédiées à la segmentation d'images peuvent être classées en deux catégories : les méthodes basées contour et les méthodes basées région. Tandis que les méthodes basées contour se fient uniquement au gradient de l'image à la position courante du modèle, les méthodes basées région utilisent une information globale d'intensité des différents segments de l'image. Nous passons également en revue des travaux importants sur l'introduction d'information de forme *a priori* dans le processus d'extraction, dans le but de lever l'ambiguïté du problème de segmentation. Il est à noter que l'information de contour, l'information de région et les informations *a priori* peuvent être avantageusement combinées dans une même fonctionnelle d'énergie, à l'instar de [70]. Dans la suite, pour plus de clarté, nous présentons ces composants séparément. Nous renvoyons le lecteur à [174, 97] pour des articles consacrés spécifiquement à l'utilisation des modèles déformables pour la segmentation des images médicales.

Méthodes basées contour

Le modèle "snake" de Kass, Witkin et Terzopoulos [77] fait partie de cette catégorie. Cet article fondateur propose de trouver les contours d'une image $I : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ en déplaçant une courbe paramétrée $C : [0, 1] \rightarrow \Omega$ sous l'influence de forces internes et externes, dérivant de la minimisation de la fonctionnelle d'énergie suivante :

$$E(C) = \underbrace{\int_0^1 \alpha(p) |C'(p)|^2 dp + \int_0^1 \beta(p) |C''(p)|^2 dp}_{E_{int}(C)} + \lambda \underbrace{\int_0^1 P(C(p)) dp}_{E_{ext}(C)} . \quad (1.4)$$

L'énergie interne rend compte de l'élasticité et de la rigidité de la courbe, qui peuvent être localement modulées avec les paramètres de pondération $\alpha(p)$ et $\beta(p)$. L'énergie externe est l'intégrale le long de la courbe d'une fonction de potentiel P qui prend des valeurs faibles aux endroits de fort gradient de l'image.

Cette approche procure une remarquable robustesse au bruit et aux données manquantes. Cependant, elle présente d'importantes limitations. Premièrement, comme nous l'avons mentionné précédemment, l'évolution et les résultats dépendent de la paramétrisation, car l'énergie n'est pas intrinsèque à la courbe. Dans (1.4), cette dépendance se manifeste par les dérivées de C et par les intégrales par rapport au paramètre de la courbe. Pour rendre cette formulation intrinsèque, une paramétrisation par l'abscisse curviligne devrait être maintenue tout au long de l'évolution, ce que l'approche ci-dessus ne peut accomplir. Deuxièmement, elle est très sensible à l'initialisation. Étant donnée la portée locale de l'information de gradient, elle peut facilement converger vers de faux contours si la forme initiale n'est pas très proche de la configuration désirée.

Plusieurs extensions ont été proposées dans la littérature pour surmonter ces limitations. L'indépendance vis à vis de la paramétrisation a été obtenue par plusieurs auteurs simultanément [23, 29, 94] grâce à des évolutions géométriques, dérivant du flot par courbure moyenne, et implémentées avec la méthode des ensembles de niveau. Une *force ballon*, qui peut faire gonfler ou dégonfler le modèle, a été proposée par Cohen dans [31] pour lever la contrainte d'initialiser le modèle près des contours de l'objet recherché. En combinant ces deux approches, l'évolution du modèle Γ est de la forme

$$\frac{\partial \Gamma}{\partial t} = g(-H + c) \mathbf{N}. \quad (1.5)$$

Dans l'équation ci-dessus, \mathbf{N} est le vecteur normal unitaire pointant vers l'extérieur, H est la courbure moyenne du modèle, c est l'amplitude de la force ballon, et g est une fonction positive prenant des valeurs faibles aux endroits d'intérêt. Par exemple, si le modèle est censé s'arrêter sur les contours d'une image, on définit généralement g tel que

$$\begin{cases} g \rightarrow 0 & \text{if } \|\nabla I\| \rightarrow +\infty, \\ g \rightarrow 1 & \text{if } \|\nabla I\| \approx 0. \end{cases} \quad (1.6)$$

Le flot (1.5) et l'équation d'évolution associée de la fonction de niveau s'appliquent à un nombre quelconque de dimensions. Γ peut être une courbe en 2D, une surface en 3D, et ainsi de suite. Nous notons n le nombre de dimensions. Un problème de cette approche est que le modèle peut ralentir sans complètement s'arrêter sur les contours de faible contraste. On parle souvent de problème de "fuite". Un autre problème est que (1.5) ne dérive pas d'une formulation variationnelle.

Les *contours actifs géodésiques*, proposés par Caselles, Kimmel et Sapiro dans [24] et par Kichenassamy et al. dans [78, 179], ont représenté un grand progrès. Ils reposent sur la minimisation de la fonctionnelle d'énergie suivante :

$$E(\Gamma) = \int_{\Gamma} g(\mathbf{x}) d\mathbf{x}, \quad (1.7)$$

où $d\mathbf{x}$ désigne l'élément d'aire du modèle (sa longueur en 2D, son aire en 3D, et ainsi de suite). Il est intéressant de constater que cette énergie peut être interprétée comme

l'aire géodésique du modèle dans un espace de Riemman doté de la métrique g . Le flot géométrique de gradient associé, donné par

$$\frac{\partial \Gamma}{\partial t} = [-\nabla g \cdot \mathbf{N} - (n-1)gH] \mathbf{N}, \quad (1.8)$$

est plus robuste aux contours de faible contraste que les approches précédentes, du fait du terme de rappel $\nabla g \cdot \mathbf{N}$ qui retient le modèle s'il dépasse le contour. Cependant, le domaine d'attraction est encore limité, à cause de la portée locale de l'information de gradient. De ce fait, le modèle se déforme difficilement dans les grandes concavités.

Pour atténuer ce problème, Siddiqi et al. proposent des flots minimisant l'aire et la longueur [146]. Xu et Prince élaborent une nouvelle force appelée *flots vectoriels de gradient* [176]. Leur approche consiste à lisser le champ de gradient d'une carte des contours de l'image au moyen d'une EDP non-linéaire, étendant ainsi le domaine d'attraction des frontières à toute l'image.

Malgré de significatives améliorations, les méthodes basées contour nécessitent encore une initialisation manuelle précise. Cette limitation a favorisé l'émergence des méthodes basées région.

Méthodes basées région

La méthode proposée par Chan et Vese dans [25] a sans doute été le travail le plus remarqué dans cette catégorie. Cette approche dérive de la minimisation de la fonctionnelle de Mumford-Shah [108], avec la procédure de minimisation par ensembles de niveau discutée dans la section 1.2. Une partition de l'image en 2^n régions supposées d'intensité constante est effectuée en faisant évoluer n fonctions de niveau. Les régions sont identifiées par les différentes combinaisons de signe des fonctions de niveau. Dans le cas le plus simple de bi-partition, la fonctionnelle d'énergie s'écrit :

$$E(\phi, c_1, c_2) = \int_{\Omega} |I(\mathbf{x}) - c_1|^2 H(\phi(\mathbf{x})) d\mathbf{x} + \int_{\Omega} |I(\mathbf{x}) - c_2|^2 [1 - H(\phi(\mathbf{x}))] d\mathbf{x} + \lambda \int_{\Omega} \delta(\phi(\mathbf{x})) \|\nabla \phi(\mathbf{x})\| d\mathbf{x}, \quad (1.9)$$

où H et δ désignent les fonctions de Heaviside et la distribution de Dirac. c_1 et c_2 sont les estimations des constantes d'intensité dans les deux régions. Leurs valeurs optimales à ϕ constant sont les moyennes empiriques de l'intensité dans les régions correspondantes. Dans [170], les mêmes auteurs proposent une extension de cette approche à des images lisses par morceaux.

Rousson, Paragios et Deriche [114, 130, 115] encapsulent la segmentation basée région dans une formulation bayésienne. La fonctionnelle d'énergie provient de la maximisation de la probabilité postérieure du modèle, ce qui est plus connu sous le nom de technique de *maximum a posteriori* (MAP). Les statistiques d'intensité des différentes régions sont modélisées par des densités Gaussiennes ou des mélanges de densités Gaussiennes. Ces statistiques sont soit apprises une fois pour toutes avant le lancement de l'algorithme, soit réestimées itérativement avec un algorithme de *maximisation de l'espérance* (EM).

Parallèlement à ces approches reposant sur une procédure de minimisation par ensembles de niveau, des approches très similaires reposant sur des flots géométriques de gradient ont été proposées [190, 165, 71]. Dans le chapitre 7, nous combinons une méthode de segmentation basée région de ce type avec la méthode des ensembles de niveau avec contrôle de la topologie du chapitre 4, pour extraire plusieurs tissus de la tête à partir d'images par résonance magnétique (IRM).

Information de forme *a priori*

La segmentation d'images est un problème éminemment mal posé, du fait de facteurs de perturbation tels le bruit, les occultations, etc. Pour pouvoir traiter des images complexes, une connaissance *a priori* de la forme peut être requise afin de réduire l'ambiguïté du problème. L'usage d'une telle information au sein de la méthode des modèles déformables s'est longtemps limité à une hypothèse de régularité ou à des familles paramétriques simples de formes. Mais la tendance récente est à l'intégration dans les modèles déformables d'un *a priori* de forme plus élaboré.

Un travail important dans cette direction est le *active shape model* de Cootes et al. [33]. Cette approche réalise une analyse en composantes principales (ACP) sur les positions de points d'amer placés de manière cohérente sur tous les contours de la base d'apprentissage. Le nombre de degrés de liberté du modèle est réduit en ne considérant que les principaux modes de variation. Cette approche est assez générale et a pu être appliquée à divers types de formes (mains, visages, organes). Toutefois, le recours à une représentation paramétrée et le positionnement manuel des points d'amer, particulièrement fastidieux dans les images 3D, limitent sérieusement son applicabilité.

Leventon, Grimson et Faugeras [86] contournent ces limitations en calculant des statistiques de forme indépendantes de la paramétrisation, par le biais de la représentation par ensembles de niveau. Fondamentalement, ils effectuent une ACP sur les fonctions distance signées des contours d'apprentissage, et le modèle statistique obtenu est intégré dans des contours actifs géodésiques. L'équation d'évolution contient un terme qui attire le modèle vers un *a priori* de forme optimal. Ce dernier est la combinaison de la forme moyenne et des principaux modes de variation. Les coefficients des différents modes et les paramètres de pose sont mis à jour par un processus secondaire d'optimisation. Plusieurs améliorations à cette approche ont été proposées depuis [131, 28, 164], et en particulier une élégante intégration du modèle statistique de forme dans une unique optimisation bayésienne. Mentionnons également une autre formulation bayésienne d'*a priori* de forme, reposant sur une représentation B-spline, proposée par Cremers, Kohlberger et Schnörr dans [35].

1.3.2 Stéréovision multi-caméras

La méthode des modèles déformables a été appliquée à la reconstruction 3D complète d'une scène à partir d'un grand nombre de vues. Le travail le plus marquant dans cette catégorie

est sans doute la stéréovision par ensembles de niveau de Keriven et Faugeras [50]. Dans cette approche, le problème de stéréovision est formulé comme la recherche d’une surface minimale, à l’instar de la méthode des contours actifs géodésiques [24]. En d’autres termes, la fonctionnelle d’énergie s’écrit comme l’intégrale sur la surface recherchée d’un critère de fidélité aux données. Ce critère est la corrélation croisée normalisée entre les différentes paires d’images. Enfin, l’évolution de la surface est implémentée avec la méthode des ensembles de niveau [112].

Il y a eu plusieurs variations autour cette approche : une implémentation avec des maillages surfaciques non-structurés [45], l’ajout d’une information de points 3D [45, 88] et d’une information de silhouette [69, 88], et une extension aux scènes spatio-temporelles [57]. La méthode proposée dans [75] pour traiter les scènes non-lambertiennes constitue une amélioration plus significative. Cette méthode est capable d’estimer à la fois la forme et la réflectance non-lambertienne de la scène. Elle fournit en sortie un modèle géométrique et photométrique qui permet de prédire l’apparence de nouvelles vues. La déformation de la surface est pilotée par la minimisation du rang d’un tenseur de radiance.

Dans le chapitre 8, nous remédions à plusieurs limitations communes à toutes ces méthodes, comme l’impossibilité d’incorporer des informations globales d’intensité dans le processus de mise en correspondance, le manque de latitude dans le choix du critère d’appariement, ou la complexité de l’implémentation lorsque la mesure d’appariement dépend de la normale à la surface. Notre approche permet également d’estimer le champ de vitesse non-rigide tri-dimensionnel d’une scène, connu sous le nom de *flot de scène* [167], à partir de plusieurs séquences vidéos.

1.4 Contributions de cette thèse

Cette thèse est divisée en deux parties et chaque partie est composée de trois chapitres. La première partie renferme nos contributions méthodologiques à la méthode des modèles déformables, en ce qui concerne la représentation géométrique et la procédure de minimisation. Ces apports élargissent le champ d’application des modèles déformables, et peuvent accroître leur performance. Le chapitre 3 et le chapitre 4 traitent de la méthode des ensembles de niveau, tandis que le chapitre 5 aborde la procédure de minimisation.

La deuxième partie de cette thèse est consacrée à des applications spécifiques dans les domaines de la vision par ordinateur et de l’imagerie médicale : le dépliage de cortex dans le chapitre 6, la segmentation de la tête à partir de l’IRM dans le chapitre 7, et la stéréovision multi-caméras et l’estimation du flot de scène dans le chapitre 8. Une évaluation expérimentale poussée sur des données réelles est effectuée pour chacune de ces applications. Toutes nos expériences numériques reposent sur une implémentation par ensembles de niveau, mais nous soulignons que la plupart de ces contributions ne sont pas spécifiques à une représentation géométrique particulière.

Il est à noter que l’ordre des chapitres ne reflète pas parfaitement l’ordre chronologique des contributions. En particulier, la procédure de minimisation du chapitre 5 est notre

plus récent apport. C'est la raison pour laquelle les applications des chapitres 6-8 n'avaient pas encore bénéficié de cette méthodologie lorsque cette thèse a été écrite.

1.4.1 Contributions méthodologiques

Chapitre 3

Dans ce chapitre, nous proposons une méthode complètement eulérienne pour surmonter la perte de la correspondance ponctuelle pendant une évolution par ensembles de niveau. Nous décrivons une implémentation numérique robuste de notre approche, compatible avec la technique de bande mince. Nous montrons dans un large éventail d'expériences numériques qu'elle peut gérer à la fois les vitesses normales et tangentielles, les grandes déformations, les chocs, les raréfactions, et les changements de topologie. Dans le chapitre 6, nous utilisons notre méthode pour produire des représentations dépliées du cortex cérébral avec préservation de l'aire.

Chapitre 4

Dans ce chapitre, nous présentons deux nouvelles méthodes pour exercer un contrôle sur la topologie pendant une évolution par ensembles de niveau. Notre première méthode permet de faire évoluer simultanément plusieurs interfaces imbriquées tout en interdisant les changements de topologie et les intersections mutuelles. Contrairement aux méthodes existantes pour les modèles déformables explicites, qui reposent soit sur des forces de répulsion, soit sur une vérification coûteuse de l'absence d'intersection entre deux mailages, notre approche garantit strictement l'absence d'intersections et ne nécessite que des calculs locaux et efficaces. Dans le chapitre 7, nous utilisons notre méthode pour extraire simultanément les interfaces internes et externes du cortex cérébral à partir de l'IRM.

Notre deuxième méthode comble le vide existant entre la méthode des ensembles de niveau standard et la méthode des ensembles de niveau avec préservation de la topologie de Han, Xu et Prince [64]. Notre nouvelle méthode des ensembles de niveau *avec préservation du genre* permet aux différentes composantes connexes de l'objet de fusionner et de se scinder, tout en garantissant qu'aucune nouvelle poignée n'est générée et qu'aucune poignée existante n'est fermée. En comparaison avec la méthode des ensembles de niveau avec préservation de la topologie, la sensibilité aux conditions initiales est grandement réduite. Nous montrons dans des expériences numériques l'intérêt que présente notre méthode des ensembles de niveau avec préservation du genre pour la segmentation des images médicales.

Chapitre 5

Dans ce chapitre, nous abordons un aspect important des modèles déformables variationnels, qui a souvent été négligé jusqu'ici : l'optimisation par les flots de gradient. Nous cherchons à savoir s'il est pertinent d'utiliser d'autres produits scalaires que le produit scalaire canonique L^2 , ce qui conduit à d'autres descentes de gradient, mais aussi d'autres

flots géométriques de minimisation ne dérivant d’aucun produit scalaire. En particulier, nous montrons comment induire différents degrés de cohérence spatiale dans le flot de minimisation, afin de réduire les risques d’être piégé dans des minima locaux inappropriés. Nous montrons dans des expériences numériques que la sensibilité de la méthode des modèles déformables aux conditions initiales est réduite par nos flots géométriques de minimisation spatialement cohérents.

1.4.2 Contributions appliquées

Chapitre 6

Dans ce chapitre, nous proposons une nouvelle méthode à base de modèles déformables pour générer des représentations dépliées du cortex cérébral avec préservation de l’aire. Nous faisons évoluer la surface corticale avec un mouvement normal dépendant de l’application, et avec un mouvement tangentiel construit de sorte à assurer une préservation exacte de l’aire tout au long de l’évolution. Nous décrivons la formulation continue de notre méthode, et son implémentation numérique par ensembles de niveau. Une implémentation directe par ensembles de niveau est impossible du fait de l’absence de paramétrisation. Nous utilisons la méthode proposée au chapitre 3 pour contourner cette écueil. Nous démontrons la faisabilité de notre approche en calculant des représentations gonflées du cortex à partir de données réelles de cerveau.

Chapitre 7

Dans ce chapitre, nous présentons une méthode pour extraire automatiquement et avec précision des maillages surfaciques de différents tissus de la tête à partir d’image IRM anatomiques. Notre méthode met l’accent sur la garantie de certaines propriétés topologiques des maillages, comme la topologie sphérique, l’absence d’auto-intersections et d’intersections mutuelles. Notre méthode est une habile combinaison d’une classification par champs de Markov cachés [185], d’une méthode de modèle déformable basée région, et de notre méthode des ensembles de niveau imbriqués proposée au chapitre 4. Cette dernière permet d’empêcher les intersections mutuelles entre les interfaces internes et externes du cortex cérébral. Nous faisons une démonstration de chaque étape de notre approche sur des données réelles de cerveau.

Chapitre 8

Dans ce chapitre, nous présentons une nouvelle méthode à base de modèles déformables pour la stéréovision multi-caméras et l’estimation du flot de scène. Notre méthode minimise l’erreur de prédiction en se servant d’un score de mise en correspondance global et basé image. Nous transformons les vues d’entrée de manière à soustraire la distorsion projective, et nous recalons les images ainsi obtenues avec une mesure de similarité choisie par l’utilisateur, qui peut incorporer des informations d’intensité de voisinage ou globales. Aucune approximation de la forme, du mouvement, ou de la visibilité n’est faite durant

la mise en correspondance. Nous obtenons des résultats comparables ou supérieurs aux meilleurs méthodes existantes, même sur des images de scènes réelles complexes, comportant des spécularités et de la transparence. Grâce à notre algorithme d'estimation du mouvement, nous parvenons à reconstruire le mouvement 3D d'une scène non-rigide et à synthétiser des séquences 3D interpolées dans le temps.

1.4.3 Contributions logicielles

Le développement logiciel constitue un troisième type de contribution de cette thèse. D'une part, nous avons développé une librairie C++ flexible et efficace, dédiée aux évolutions de courbes et de surfaces par ensembles de niveau. Cette librairie intègre les techniques de bande mince, de réinitialisation, de fast marching, et certaines de nos contributions méthodologiques, comme la gestion des données d'interface et de la correspondance ponctuelle, et le contrôle de la topologie. Ce code a servi de brique de base à d'autres membres du projet de recherche Odysée pour diverses applications, comme la segmentation d'images IRM de diffusion.

D'autre part, notre programme consacré à la segmentation de tissus de la tête à partir d'images IRM, décrit au chapitre 7, est utilisé quotidiennement au sein du projet Odysée et dans la Section de Neurophysiologie de l'Université Catholique de Leuven, en Belgique.

À la fin de chaque chapitre, nous rappellerons les contributions principales et nous indiquerons les publications associées. L'annexe A rassemble des formules utiles de calcul géométrique et différentiel. L'annexe B décrit les schémas numériques nécessaires pour implémenter nos EDP. Enfin, l'annexe C est une liste de nos publications.

Chapter 2

Introduction

Deformable models, also known in the literature as snakes, active contours/surfaces, deformable contours/surfaces, constitute a widely used computerized technique to address various shape reconstruction problems in image processing. A deformable model is a curve, a surface, or a higher-dimensional geometric object, initially positioned by the user, which moves under some constraints and under the influence of the input data, until it converges to a candidate solution to the problem.

This methodology has several outstanding features. It is highly versatile through the choice of a shape representation and the design of the evolution equation. As a result, it can be applied to various types of input data (two-dimensional, three-dimensional and four-dimensional images, multi-view stereovision datasets, unstructured point sets, ...) and to a large range of problems. It has been initially proposed for the purpose of image segmentation by Kass, Witkin and Terzopoulos in [77], but it has proven successful in many other contexts, in computer vision and in medical imaging, including region tracking [115, 101], shape from stereo [50, 75, 45, 88, 69, 57], shape from shading [53, 74, 182], and shape from point clouds [188, 10, 160].

Moreover, the use of a continuous geometric formulation has many benefits. It leads to a neat mathematical framework. It provides a subpixel accuracy. It correctly models the continuity of shape, and allows to incorporate some regularity assumptions or some complex prior information about the target shape, thereby yielding a good robustness to noisy and incomplete data. Finally, it succeeds in naturally integrating many input local features into a consistent mathematical shape description.

The key elements of the deformable models framework are the design of an energy functional, the choice of a minimization procedure and of a geometric representation. In this thesis, we tackle these three elements, with the objective of increasing the applicability and efficiency of deformable models. Among the various existing types of deformable models, we concentrate on geometric evolutions originating from the theory of curve evolution [54] and we put a particular emphasis on the level set representation [112].

With regard to the geometric representation, we overcome the loss of the point correspondence and the inability to control topology changes with the level set method. We

propose two associated applications in the field of medical imaging: the generation of unfolded area preserving representations of the cerebral cortex, and the segmentation of several head tissues from anatomic magnetic resonance images. With regard to the minimization procedure, we show that the robustness to local minima can be improved by substituting a spatially coherent minimizing flow to a traditional gradient descent. Finally, with regard to the design of the energy functional, we propose a novel modeling of multi-view stereovision and scene flow estimation with deformable models, relying on a global image-based matching score.

The rest of this introductory chapter is organized as follows. In Section 2.1, we examine the advantages and the drawbacks of the different geometric representations of deformable models. In particular, we point out several limitations of the level set method, which motivate some of our methodological contributions. Section 2.2 focuses on the evolution equation and emphasizes the detrimental sensitivity of deformable models to initial conditions, which we tackle in this thesis. Section 2.3 deals with the design of the energy functional. As this matter is highly dependent on the details of each application, we only report the major trends in two important applications of the deformable models framework to which our work contributes: image segmentation and multi-view stereovision. Finally, in Section 2.4, we sum up the important contributions of this thesis and we describe the content of the next chapters.

2.1 Geometric representation of deformable models

Many different geometric representations have been proposed in the deformable models literature. The interested reader may refer to [105] for a thorough review. Often, these representations are divided into two categories: “parametric” models and “geometric” models. However, we believe that these two terms are a source of confusion. First, we suggest to use “parameterized” instead of “parametric” to deal with geometric objects given by a parameterization, and to keep the term “parametric” for the families of objects indexed by a small number of parameters. Second, in the literature, “geometric” models often refer exclusively to the level set representation. As a result, some representations like unstructured surface meshes would be classified as “parametric”, whereas they are neither parametric nor parameterized. . .

In this thesis, we make a distinction between “explicit” and “implicit” representations. Explicit models are given through a set of coordinates while implicit models are defined as the zero set of a higher-dimensional scalar function. In parallel, we make a difference between “geometric” and “non-geometric” evolution laws. Geometric evolutions are not dependent on a particular parameterization. They find their origin in the theory of curve evolution [54]. In this type of evolution, the velocity field of the model typically depends on intrinsic geometric quantities such as normal, curvature, and on the input data.

Geometric evolutions can be implemented very naturally with a level set representation, but the use of other representations, including explicit models, is possible. Choosing an

explicit or an implicit representation to implement a geometric evolution is closely related to choosing a Lagrangian or an Eulerian perspective.

2.1.1 Explicit models

Parameterization

Parameterized curves and surfaces are an important type of explicit models. This kind of representation is the one proposed in the original *snake* model of Kass, Witkin and Terzopoulos [77]. A variety of other parameterized representations are found in the literature, either based on some basis functions (finite elements [32, 96], B-splines [132], Fourier harmonics [151], etc.), or on a parametric family (superquadrics [161, 9, 10]).

A typical problem of parameterized deformable models is that their evolution and the final reconstruction are dependent on the parameterization of the initial shape. To be more exact, this problem is related to the definition of the energy functional rather than to the geometric representation itself: in most cases, the metric distortions between the parameterization space and the model are disregarded. As a consequence, the energy is not intrinsic to the model.

Not all explicit models are parameterized. Oriented particle systems [159] and unstructured meshes (triangulated surfaces [104], simplex meshes [39], polyhedrons [147]) do not have an underlying parameterization.

Topology

A serious limitation of most explicit models is that their topology cannot change during the evolution to fit the data topology. Several attempts have been made to overcome this problem. McInerney and Terzopoulos [98, 99] propose topology adaptative deformable curves and triangulations, called *T-snakes* and *T-surfaces*. During the evolution, the model is periodically resampled by computing its intersections with a simplicial decomposition of space. A labeling of the vertices of the simplicial grid as inside or outside of the model is maintained. Besides being computationally expensive, this procedure imposes a fixed uniform spatial resolution. Also, not all motions are admissible: this approach only works when the model inflates or deflates everywhere.

Lachaud and Montanvert [85] use the concept of δ -triangulation. A length parameter δ is used to control the sampling of the triangulation and to detect self-intersections, by monitoring the distance between pairs of neighbor and non-neighbor vertices. This approach is costly, even when optimizing the computations with an octree structure.

Delingette and Montagnat [39, 40] propose to modify the topology of an evolving simplex mesh with some elementary topological operators, but their approach needs manual interaction in 3D.

Thus, a fully automatic and efficient handling of topology changes with explicit models, in three dimensions and more, remains an open issue. This is one important reason why the level set representation has received much interest in the deformable models literature.

2.1.2 Implicit models

Implicit models are not limited to the level set method [112]. Algebraic polynomial surfaces [160] and implicit superquadrics/hyperquadrics [9, 30] also fall into this category. However, among implicit models, the level set method is by far the most powerful. It is applicable to a considerably wider range of applications, because it can handle complex geometries while the others are limited to a small family of shapes.

The level set method, introduced by Osher and Sethian in [112] (a similar work in the area of fluid mechanics [41, 42] has recently surfaced), is an established technique to represent moving interfaces in two or more dimensions. Basically, it consists in representing the interface implicitly as the zero level set of a higher-dimensional scalar function. The movement of the interface can be cast as an evolution of the embedding level set function by an Eulerian PDE (partial differential equation).

On the one hand, this approach has several advantages over an explicit Lagrangian representation of the interface: no parameterization is needed, topology changes are handled automatically, any number of dimensions is accommodated, intrinsic geometric properties such as normal or curvature can be computed easily from the level set function. Last but not least, the theory of viscosity solutions [34] provides robust numerical schemes and strong mathematical results to deal with the evolution PDE.

On the other hand, several serious shortcomings limit the applicability of the level set method. First, the higher dimensional embedding makes the level set method much more expensive computationally than explicit representations. Much effort has been done to alleviate this drawback, leading to the narrow band methodology [1] and more recently to the PDE-based fast local level set method [118].

Also, in its basic formulation, the level set method can only represent manifolds of codimension one without borders, such as closed curves in \mathbb{R}^2 and closed surfaces in \mathbb{R}^3 . Several approaches have been proposed to handle a codimension strictly greater than one. Ambrosio and Soner [5] propose to evolve an hypersurface corresponding to an ϵ -neighborhood of the manifold. For instance, in the case of a curve in \mathbb{R}^3 , it consists in evolving a tubular neighborhood of the curve which has a small radius ϵ . This idea was used in [90] to segment blood vessels from medical images. A drawback of this approach is that the manifold of interest cannot be accurately positioned. A more principled approach by Burchard, Cheng, Merriman and Osher [20] consists in representing the curve as the intersection of two hypersurfaces, and to express its motion as the evolution of two level set functions. In [59], the author investigates the extension of this idea to any codimension k by simultaneously evolving k level set functions, at the expense of facing increasing numerical difficulties. A similar idea is proposed by Solem and Heyden [149] to represent surfaces with borders with the level set method.

Some other shortcomings of the level set method have recently surfaced. One is intimately related to the implicit point of view and to the absence of parameterization: the point-wise correspondence is lost during the evolution. In other words, we do not know how each point and each part of the interface moves. This is not a surprise, since the level

set function conveys a purely geometric description of the interface. More generally, it is not possible to handle some data associated with the moving interface, in other words some *interfacial data*, in the traditional level set framework. This considerably restricts the range of possible applications. In Chapter 3, we propose a method, based on a system of coupled Eulerian PDEs, to overcome this limitation.

The ability to automatically handle topology changes is a long claimed advantage of the level set method over explicit deformable models. But it may not be desirable in some applications where some prior knowledge of the target topology is available. This is typically the case in biomedical image segmentation, where the topology of the organs and their mutual topological relations is prescribed by anatomical knowledge. Chapter 4 deals with some methods, inspired by digital topology, to exert a control on the topology during a level set evolution.

2.2 Minimization of the energy functional

The design of the evolution equation driving the motion of deformable models follows two main approaches. In the *variational* formulation, the problem is cast as the minimization of an energy functional. The latter is defined so that low-energy configurations indicate a good fit with the input data and the prior information (typically some regularity assumptions), and the solution is defined as a shape achieving a global minimum of the energy. In the *dynamic force* formulation, the motion is designed *ad hoc*, often as a combination of internal forces, defined from the model itself and dedicated to keep the model smooth during the deformation, and of external forces, defined from the input data. The solution is seen as an equilibrium of the forces.

In this thesis, we favor the variational formulation, because it has several advantages over the dynamic force formulation. First, it appears less empirical and often enables a better understanding of the model. This is particularly true when the energy functional derives from a statistical modeling, such as the *maximum a posteriori* (MAP) of a Bayesian formulation [157]. Second, in some cases, the existence of a global minimum of the energy and the well-posedness of the minimization procedure can be mathematically proven. Finally, once the variational problem is defined, it can be tackled with a variety of minimization procedures, depending on the available prior information and on the allocated computational time. In this section, we deal with the choice of such a minimization procedure.

In general, an exact minimization of the energy functional is computationally unfeasible due to the huge number of unknowns. *Simulated annealing* [152] and *dynamic programming* [6] have been proposed to compute a global minimum, but the former is very slow in practice and the latter only applies to a particular form of energy functionals. More recently, *graph cuts* have emerged as a powerful energy minimization method allowing to find a global minimum or a strong local minimum. In the last few years, this method has been successfully applied to several problems in computer vision, including multi-view

stereovision [80] and image segmentation [18]. However, it has two severe limitations: it cannot be applied to any energy function [81], and, when applicable, is computationally expensive.

Hence, in most cases, a suboptimal strategy is adopted, based on the calculus of variations. A necessary condition of optimality deduced from the energy functional, called the *Euler-Lagrange equation*, is solved instead. The Euler-Lagrange equation characterizes the local minima and the local maxima of the energy. Just like the original variational problem, it cannot be solved exactly, so an evolutive method starting from an initial guess is necessary. As a result, an artificial notion of time is added to the problem. This is the core principle of the deformable models framework: the resolution of the problem translates into the time evolution of a geometric object.

When using parameterized deformable models, the resolution of the Euler-Lagrange equation is typically done by updating the model parameters with a gradient descent or with fast convergence numerical schemes such as the conjugate gradient method, the Newton method or the Levenberg-Marquardt method. These evolutions are non-geometric, i.e. they are dependent on a parameterization of the model, so they fall out of the scope of this thesis.

Another minimization procedure was pioneered by Chan and Vese in [25]. It consists in using the level set method at the same time as a geometric representation and as an optimization framework. In other words, the energy functional is defined directly from the level function ϕ , and the minimization is also performed with respect to ϕ . In this approach, the integral of a quantity f along the boundary of the model or over the inside/outside region write with the Dirac distribution δ and the Heaviside function H :

$$\int_{\Gamma} f(\mathbf{x}) d\mathbf{x} = \int_{\mathbb{R}^n} \delta(\phi(\mathbf{x})) f(\mathbf{x}) \|\nabla\phi(\mathbf{x})\| d\mathbf{x} , \quad (2.1)$$

$$\int_{\text{in}} f(\mathbf{x}) d\mathbf{x} = \int_{\mathbb{R}^n} [1 - H(\phi(\mathbf{x}))] f(\mathbf{x}) d\mathbf{x} , \quad (2.2)$$

$$\int_{\text{out}} f(\mathbf{x}) d\mathbf{x} = \int_{\mathbb{R}^n} H(\phi(\mathbf{x})) f(\mathbf{x}) d\mathbf{x} . \quad (2.3)$$

This approach has become popular in image segmentation [25, 170, 114, 56, 130]. However, it raises several conceptual and practical problems. First, it is specific to a particular geometric representation. Second, the need for an ϵ -regularized version of δ and H in the implementation is quite inelegant. Last but not least, the resulting evolution is non-geometric. It depends on the initial values of ϕ off the zero level. Disconnected components of the model can appear, which is definitely not compatible with a curve evolution.

Moreover, a mistake commonly done with this approach is to assume the preservation of the signed distance property when computing the variation of the energy with respect to a variation of ϕ . For example, in [56], the authors propose to segment the cerebral cortex with two coupled deformable models, and with a prior on their mutual distance. This distance is taken as the absolute value of the signed distance function representing each interface. Because of the aforementioned abuse, the derivation of the energy functional in

this work is not exact. A derivation along the space of signed distance functions is neither possible, because this space is not a well-behaved manifold.

In this thesis, we focus on a minimization procedure called *geometric gradient flows*. It is the geometric evolution obtained by following the direction of steepest descent of an intrinsic energy. Its mathematical meaning is detailed in Chapter 5. For instance, the minimization of the area of the model leads to the well-known motion by mean curvature [54]. Another important geometric gradient flow is obtained by minimizing the area of the model in a Riemannian space with an image-based metric: it is the *geodesic active contours* approach proposed by Caselles, Kimmel and Sapiro in [24] for the detection of image boundaries.

Due to the highly non-convex nature of most energy functionals, geometric gradient flows are very likely to be trapped in a local minimum. Also, this local minimum depends on the position of the initial shape. If the latter is far from the expected final configuration, the evolution may be trapped in a completely irrelevant state. This sensitivity to initial conditions seriously limits the applicability and efficiency of the deformable models framework.

There are essentially two ways of dealing with this problem: positioning the initial model very close to the expected final configuration, or using a multiresolution coarse-to-fine strategy, in other words running the optimization on a series of smoothed and subsampled models and input data. In Chapter 5, we pioneer a third way to tackle the problem of unwanted local minima: the careful design of new geometric minimizing flows.

2.3 Major trends in the design of the energy functional

2.3.1 Image segmentation

The many deformable models methods dedicated to image segmentation can be divided into two categories: boundary-based methods and region-based methods. Whereas boundary-based methods only rely on the gradient of the image at the current position of the model, region-based methods use global intensity information of the different image segments. We also review some important works on introducing prior shape information in the extraction process, in order to cope with the highly ambiguous nature of the image segmentation problem. Note that boundary-based information, region-based information and complex prior information can advantageously be combined in a same energy functional, as in [70]. Here, for sake of clarity, we present these components separately. The reader may also refer to [174, 97] for some specific surveys on deformable models in medical image segmentation.

Boundary-based methods

The original snake model of Kass, Witkin and Terzopoulos [77] falls into this category. This seminal work proposes to find the boundaries of an image $I : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$ by moving

a parameterized curve $C : [0, 1] \rightarrow \Omega$ under the influence of some internal and external forces deriving from the minimization of the following energy functional:

$$E(C) = \underbrace{\int_0^1 \alpha(p) |C'(p)|^2 dp + \int_0^1 \beta(p) |C''(p)|^2 dp}_{E_{int}(C)} + \lambda \underbrace{\int_0^1 P(C(p)) dp}_{E_{ext}(C)}. \quad (2.4)$$

The internal energy accounts for the elasticity and the rigidity of the curve, which can be locally modulated with the weighting parameters $\alpha(p)$ and $\beta(p)$. The external energy is the integral along the curve of a potential function P which takes smaller values at locations of high image gradient.

This approach yields a remarkable robustness to noise and missing data. However, it has important limitations. First, as we mentioned earlier, the evolution and the results are dependent on the parameterization, because the energy is not intrinsic to the curve. In (2.4), this dependency is visible in the derivatives of C and in the integrals with respect to the parameter of the curve. To make the formulation intrinsic, an arc-length parameterization ought to be maintained throughout the evolution, which the above approach cannot achieve. Second, it is very sensitive to initialization. Due to the local relevance of gradient information, it may easily converge towards false edges if the initial shape is not very close to the desired configuration.

Several extensions have been proposed in the literature to overcome these limitations. The parameterization independence was achieved concurrently by several authors [23, 29, 94] with some geometric evolutions, deriving from the mean curvature flow, and implemented with the level set method. A *balloon force* that can either inflate or deflate the model was proposed by Cohen in [31] to remove the requirement to initialize the model near the desired object boundaries. When combining these two approaches, the evolution of the model Γ is of the form

$$\frac{\partial \Gamma}{\partial t} = g(-H + c) \mathbf{N}. \quad (2.5)$$

In the above equation, \mathbf{N} is the outward normal vector, H is the mean curvature of the model, c is the amplitude of the balloon force, and g is a positive function which takes smaller values at features of interest. For instance, if the model should stop on the edges of the image, g is generally defined such that

$$\begin{cases} g \rightarrow 0 & \text{if } \|\nabla I\| \rightarrow +\infty, \\ g \rightarrow 1 & \text{if } \|\nabla I\| \approx 0. \end{cases} \quad (2.6)$$

The flow (2.5) and the associated level set evolution equation apply to any number of dimensions. Γ may denote a curve in 2D, a surface in 3D, and so on. Below, let n be the number of dimensions. A problem with this approach is that the model may slow down but not completely stop on low contrast edges. This is often referred as the leakage problem. Also, (2.5) does not derive from a variational formulation.

A significant step was made with the *geodesic active contours* method proposed by Caselles, Kimmel and Sapiro in [24] and by Kichenassamy et al. in [78, 179]. It is based

on the minimization of the following intrinsic energy functional:

$$E(\Gamma) = \int_{\Gamma} g(\mathbf{x}) d\mathbf{x} , \quad (2.7)$$

where $d\mathbf{x}$ denotes the area element of the model (its length in 2D, its area in 3D, and so on). Interestingly, this energy can be interpreted as the geodesic area of the model in a Riemannian space with metric g . The associated geometric gradient flow, given by

$$\frac{\partial \Gamma}{\partial t} = [-\nabla g \cdot \mathbf{N} - (n-1)gH] \mathbf{N} , \quad (2.8)$$

is more robust to low contrast edges than the previous approach, due to the stopping term $\nabla g \cdot \mathbf{N}$ that pulls back the model if it passes the boundary. However, the attraction range is still limited, due to the local relevance of gradient information. As a result, the model has difficulties to deform into large concavities.

To address this problem, Siddiqi et al. propose area and length minimizing flows [146]. Xu and Prince work out a new force called *gradient vector flows* [176]. Their approach consists in smoothing the gradient field of an edge map of the image with a non-linear PDE, thereby extending the attraction range of image boundaries to the whole image domain.

Despite significant improvements, boundary-based methods still require an accurate manual initialization. This limitation has motivated the emergence of region-based methods.

Region-based methods

A popular work in this category is the *active contours without edges* method proposed by Chan and Vese in [25]. This approach derives from the minimization of the Mumford-Shah functional [108] with the level set based optimization procedure discussed in Section 2.2. A partitioning of the image in 2^n regions assuming to have a constant intensity is performed by evolving n level set functions. The regions are tagged by the different sign combinations of the level set functions. In the simplest case of bi-partitioning, the energy functional writes:

$$E(\phi, c_1, c_2) = \int_{\Omega} |I(\mathbf{x}) - c_1|^2 H(\phi(\mathbf{x})) d\mathbf{x} + \int_{\Omega} |I(\mathbf{x}) - c_2|^2 [1 - H(\phi(\mathbf{x}))] d\mathbf{x} + \lambda \int_{\Omega} \delta(\phi(\mathbf{x})) \|\nabla \phi(\mathbf{x})\| d\mathbf{x} , \quad (2.9)$$

where H and δ denote the usual Heaviside and Dirac functions. c_1 and c_2 are the estimated intensity constants in the two regions. Their optimal value at constant ϕ are the empirical intensity means in the corresponding regions. In [170], the same authors propose an extension of this approach to piecewise smooth images.

Rousson, Paragios and Deriche [114, 130, 115] embed region-based segmentation in a Bayesian formulation. The energy functional originates from the maximization of the

posterior probability of the model. This is known as a *maximum a posteriori* (MAP) technique. The intensity statistics of the different regions are modeled by Gaussian densities or mixtures of Gaussian densities. These statistics are either learned offline or iteratively reestimated with an *expectation-maximization* (EM) algorithm.

In parallel to these approaches relying on a level set based optimization procedure, very similar approaches relying on geometric gradient flows have been proposed [190, 165, 71]. In Chapter 7, we combine such a region-based image segmentation method with the level set method under topology control of Chapter 4 to extract several head tissues from magnetic resonance imaging (MRI).

Prior shape information

Image segmentation is an ill-posed problem due to various perturbing factors such as noise, occlusions, missing parts, cluttered data, etc. When dealing with complex images, some prior shape knowledge may be necessary to disambiguate the segmentation process. The use of such prior information in the deformable models framework has long been limited to a smoothness assumption or to simple parametric families of shapes. But a recent and important trend in this domain is the development of deformable models integrating more elaborate prior shape information.

An important work in this direction is the *active shape model* of Cootes et al. [33]. This approach performs a principal component analysis (PCA) on the position of some landmark points placed in a coherent way on all the training contours. The number of degrees of freedom of the model is reduced by considering only the principal modes of variation. The active shape model is quite general and has been successfully applied to various types of shapes (hands, faces, organs). However, the reliance on a parameterized representation and the manual positioning of the landmarks, particularly tedious in 3D images, seriously limits its applicability.

Leventon, Grimson and Faugeras [86] circumvent these limitations by computing parameterization-independent shape statistics within the level set representation. Basically, they perform a PCA on the signed distance functions of the training shapes, and the resulting statistical model is integrated into a geodesic active contours framework. The evolution equation contains a term which attracts the model towards an optimal prior shape. The latter is a combination of the mean shape and of the principal modes of variation. The coefficients of the different modes and the pose parameters are updated by a secondary optimization process. Several improvements to this approach have been proposed [131, 28, 164], and in particular an elegant integration of the statistical shape model into a unique MAP Bayesian optimization. Let us also mention another neat Bayesian prior shape formulation, based on a B-spline representation, proposed by Cremers, Kohlberger and Schnörr in [35].

2.3.2 Multi-view stereovision

The deformable models framework has been used to obtain a complete 3D reconstruction of a scene from a high number of input views. The most inspiring work in this category is the level set stereovision method of Keriven and Faugeras [50]. In this work, the stereovision problem is formulated as a minimal surface approach, in the spirit of the geodesic active contours method [24]. In other words, the energy functional is written as the integral on the unknown surface of a data fidelity criterion. This criterion is the normalized cross correlation between image pairs. The surface evolution is implemented with the level method [112].

Several variations to this approach have been proposed: an implementation with unstructured surface meshes [45], the addition of 3D points data [45, 88] and of silhouette information [69, 88], and an extension to spatio-temporal scenes [57]. More original is the method proposed in [75] to cope with non-Lambertian scenes. This method can estimate both the shape and the non-Lambertian reflectance of the scene. It outputs a geometric and photometric model which allows to predict the appearance of novel views. The surface deformation is driven by the minimization of the rank of a radiance tensor.

In Chapter 8, we tackle several limitations shared by all these methods, like the inability to incorporate global intensity information in the matching process, the lack of flexibility in the choice of the matching criterion, or the complexity of the implementation when the matching measure depends on the surface normal. Our approach also allows to estimate the dense non-rigid 3D motion field of a scene, often called *scene flow* [167], from multiple video sequences.

2.4 Contributions of this thesis

This thesis is organized in two parts and each part is composed of three chapters. The first part consists in our methodological contributions to the deformable models framework, with regard to the geometric representation and to the minimization procedure. These contributions widen the range of applications of deformable models, and can increase their efficiency. Chapter 3 and Chapter 4 deal with the level set method, while Chapter 5 tackles the minimization procedure.

The second part of this thesis is dedicated to some specific applications in the fields of computer vision and medical imaging, which we believe our work advances: cortex unfolding in Chapter 6, head segmentation from MRI in Chapter 7, and multi-view stereovision and scene flow estimation in Chapter 8. A thorough experimental evaluation on real data is given for each of these applications. All our numerical experiments are based on a level set implementation, but we insist on the fact that most of these contributions are not specific to a particular geometric representation.

Note that the order of the chapters does not reflect perfectly the chronological order of the contributions. In particular, the minimization procedure of Chapter 5 is our more recent contribution. That is the reason why the applications of Chapters 6-8 did not

benefit from this methodology at the time of the writing of this thesis.

2.4.1 Methodological contributions

Chapter 3

In this chapter, we propose a completely Eulerian method to overcome the loss of the point correspondence during a level set evolution. We describe in details a robust numerical implementation of our approach, in accordance with the narrow band methodology. We show in a variety of numerical experiments that it can handle both normal and tangential velocities, large deformations, shocks, rarefactions and topology changes. In Chapter 6, we use our method to generate unfolded area preserving representations of the cerebral cortex.

Chapter 4

In this chapter, we present two novel methods to exert a control on the topology during a level set evolution. Our first method enables to concurrently evolve several nested interfaces while preventing topology changes as well as mutual intersections. Unlike existing methods with explicit deformable models, either based on repulsion forces or on a computationally intensive mesh-to-mesh intersection checking, our approach guarantees the absence of intersections and only requires efficient local computations. In Chapter 7, we use our method to simultaneously extract the inner and outer interfaces of the cerebral cortex from MRI.

Our second method fills the gap between the standard level set method and the topology preserving level set method of Han, Xu and Prince [64]. Our novel *genus preserving* level set method allows the different connected components of the object to merge and split while ensuring that no new handle is generated and no existing handle is closed. Compared to the topology preserving level set method, the sensitivity to initial conditions is greatly alleviated. We show in some numerical experiments the interest of using our genus preserving level set method for the segmentation of medical images.

Chapter 5

In this chapter, we tackle an important aspect of variational deformable models which has been largely overlooked so far: the optimization by gradient flows. We investigate the relevance of using other inner products than the canonical L^2 product, yielding other gradient descents, and some other geometric minimizing flows not deriving from any inner product. In particular, we show how to induce different degrees of spatial coherence into the minimizing flow, in order to decrease the probability of getting trapped into irrelevant local minima. We show in some numerical experiments that the sensitivity of the deformable models framework to initial conditions is alleviated by our application-specific spatially coherent geometric minimizing flows.

2.4.2 Applied contributions

Chapter 6

In this chapter, we propose a novel deformable models method to generate unfolded area preserving representations of the cerebral cortex. The cortical surface is evolved with an application-specific normal motion, and an adequate tangential motion is constructed in order to ensure an exact area preservation throughout the evolution. We describe the continuous formulation of our method as well as its numerical implementation with level sets. A straightforward level set implementation of cortex unfolding is not feasible due to the absence of parameterization. We use the method proposed in Chapter 3 to circumvent this limitation. We show the applicability of our approach by computing inflated representations of the cortex from real brain data.

Chapter 7

In this chapter, we present a method to automatically and accurately extract surface meshes of several head tissues from anatomical MR images. The emphasis of our method is on guaranteeing some topological properties of the meshes, such as spherical topology, absence of self-intersections and mutual intersections. Our method is a successful combination of hidden Markov random field classification [185], of a region-based deformable models method, and of our topology preserving nested level set method proposed in Chapter 4. The latter allows to prevent mutual intersections between the inner and outer interfaces of the cerebral cortex. We demonstrate each step of our approach on real brain data.

Chapter 8

In this chapter, we present a novel deformable models method for multi-view stereovision and scene flow estimation. Our method minimizes the prediction error using a global image-based matching score. We adequately warp the input views and we register the resulting distortion-free images with a user-defined image similarity measure, which can include neighborhood and global intensity information. No approximation of shape, motion or visibility is made in the matching process. We obtain results comparing favorably with state-of-the-art methods, even on complex non-Lambertian real-world images including specularities and translucency. Using our algorithm for motion estimation, we successfully recover the 3D motion of a non-rigid scene and we synthesize time-interpolated 3D sequences.

2.4.3 Software contributions

A third type of contribution of this thesis is software development. First, we have written a flexible and efficient C++ templated library to implement curve and surface evolutions with the level set method. This library includes narrow banding, reinitialization, fast

marching, and some of our methodological contributions, such as the handling of interfacial data, of the point correspondence and of topology control. This code is used as a building block by the other members of the Odyssee Laboratory, for various applications, such as the segmentation of DT (Diffusion Tensor) images.

Second, our program dedicated to the segmentation of head tissues from MRI, described in Chapter 7, is used routinely both in the Odyssee Laboratory and in the Neurophysiology Section of the Katholieke Universiteit Leuven, Belgium.

At the end of each chapter, we will recall the main contributions that were presented and we will indicate the associated publications. In Appendix A, we have gathered some useful formulas of geometric and differential calculus. Appendix B describes the numerical schemes needed to implement our PDEs. Finally, Appendix C lists all our publications.

Part I

Methods

Chapter 3

Maintaining the Point Correspondence in the Level Set Framework

In this chapter, we tackle a major shortcoming of the level set method: the loss of the point correspondence and the inability to handle interfacial data. We propose an approach to maintain an explicit backward correspondence from the evolving interface to the initial one. Our method consists in advecting the initial point coordinates with the same velocity as the level set function. It leads to a system of coupled Eulerian partial differential equations.

We describe in details a robust numerical implementation of our approach, in accordance with the narrow band methodology. We show in a variety of numerical experiments that it can handle both normal and tangential velocities, large deformations, shocks, rarefactions and topology changes. Our method has a very wide range of applications since it can be used to upgrade virtually any level set evolution.

In Chapter 6, we will use this method to generate unfolded area preserving representations of the cerebral cortex.

3.1 Motivation

A serious shortcoming of the level set method, which has been studied only recently, is intimately related to the implicit point of view and to the absence of parameterization: the point-wise correspondence is lost during the evolution. In other words, we do not know how each point and each part of the interface moves. This is not a surprise, since the level set function conveys a purely geometric description of the interface. More generally, it is not possible to handle some data associated with the moving interface, in other words some *interfacial data*, in the traditional level set framework. This considerably restricts the range of possible applications.

Some hybrid Lagrangian-Eulerian methods have been proposed to circumvent this

limitation in some particular applications such as the unfolding of the surface of the cerebral cortex [68], the construction of transverse lines in grid generation [143], and image registration with level sets [169]. Basically, in these works, the level set equation is complemented with a set of Lagrangian ODEs which track the points during the evolution. This particle-based approach leads to well-known numerical difficulties. Most of the advantages of using an implicit representation are lost. For sake of numerical stability and topology independence, a completely Eulerian approach must be preferred.

The earliest Eulerian approach related to the handling of some interfacial data in the level set framework appears in [14]. This method allows to track a region on a deforming level set interface. But it cannot handle either the explicit point correspondence or some arbitrary interfacial data. As a result, it has a rather limited range of applications. Also related is a method to evolve a curve in three dimensions with a level set approach [20]. More recently, some Eulerian methods have been proposed to evolve an interfacial material quantity in the level set framework [177, 3]. The data are simultaneously advected, scaled by the local compression/expansion of the interface as a result of mass conservation, and diffused along the interface. But mass conservation is not relevant to all types of interfacial data. Moreover, the explicit point correspondence is not addressed. In other words, the mapping between the initial and the final interface is not available.

In this chapter, we propose a method to maintain an explicit point correspondence during a level set evolution. Our method consists in a system of coupled Eulerian PDEs. Our approach has a very wide range of applications, since it can be used to upgrade virtually any level set evolution. The rest of this chapter is organized as follows. In Section 3.2, we analyze in details the previous work on level sets with some interfacial data and we elaborate a novel method for handling passively advected interfacial data, and above all for maintaining an explicit point correspondence. Section 3.3 describes in detail its numerical implementation. Finally, in Section 3.4, we report on some numerical experiments that demonstrate the effectiveness of our method in a wide range of situations including normal and tangential velocity fields, large deformations, shocks, rarefactions and topology changes.

3.2 Methods

In the following, we note $\Gamma(t), t \in \mathbb{R}^+$ a moving closed and embedded hypersurface in \mathbb{R}^n . Γ is represented by a level set function $\phi : \mathbb{R}^n \times \mathbb{R}^+ \rightarrow \mathbb{R}$ such that:

$$\begin{cases} \phi(\mathbf{x}, t) < 0 & \text{if } \mathbf{x} \text{ is inside } \Gamma(t), \\ \phi(\mathbf{x}, t) = 0 & \text{if } \mathbf{x} \in \Gamma(t), \\ \phi(\mathbf{x}, t) > 0 & \text{if } \mathbf{x} \text{ is outside } \Gamma(t). \end{cases} \quad (3.1)$$

The geometric properties of Γ can be easily expressed from the ϕ function:

- The outward unit normal at any point of the interface is given by $\mathbf{N} = \frac{\nabla\phi}{\|\nabla\phi\|}$.

- The mean curvature is given by $H = \frac{1}{n-1} \operatorname{div} \left(\frac{\nabla \phi}{\|\nabla \phi\|} \right)$.

For the reader's convenience, we have gathered in Appendix A.2 some other useful formulas of geometric and differential calculus in the implicit framework.

A deformation of Γ with a velocity field \mathbf{v} has a direct counterpart in the level set representation:

$$\frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi = 0. \quad (3.2)$$

It is well known that the geometry of the interface is only affected by the normal component of the velocity. The corresponding property in the level set formulation is that the tangential component of the velocity cancels in (3.2). That is the reason why the velocity is often taken normal to the interface: $\mathbf{v} = \beta \mathbf{N}$, where \mathbf{N} denotes the outward normal. This can be done without any loss of generality when geometry only is of interest. Given that $\mathbf{N} = \nabla \phi / \|\nabla \phi\|$, we get the classical form of the level set evolution equation:

$$\frac{\partial \phi}{\partial t} + \beta \|\nabla \phi\| = 0. \quad (3.3)$$

A tangential velocity does not affect the geometry, but it does affect the point correspondence and the data associated with the interface. Hence, in this work, it would be erroneous to restrict to a normal velocity field. We refer the reader to [143, 110, 111] for more details about the theory, the recent developments, the implementation and the applications of the level set method.

3.2.1 Previous work on region tracking

In [14], the problem of tracking a region in Γ during the level set evolution is addressed. The boundary of the region of interest is represented as the intersection of Γ with the interior of an auxiliary hypersurface $\hat{\Gamma}$ defined as the zero level set of a function $\hat{\phi}$. The two functions ϕ and $\hat{\phi}$ are evolved according to the following system of coupled Eulerian PDEs:

$$\begin{cases} \frac{\partial \phi}{\partial t} + \beta \|\nabla \phi\| = 0, \\ \frac{\partial \hat{\phi}}{\partial t} + \left(\beta \frac{\nabla \phi}{\|\nabla \phi\|} \cdot \frac{\nabla \hat{\phi}}{\|\nabla \hat{\phi}\|} \right) \|\nabla \hat{\phi}\| = 0. \end{cases} \quad (3.4)$$

where β is again the magnitude of the outward normal speed of the interface. Rewriting (3.4) using a form similar to (3.2) allows an easier interpretation:

$$\begin{cases} \mathbf{v} = \beta \frac{\nabla \phi}{\|\nabla \phi\|}, & (3.5a) \\ \frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi = 0, & (3.5b) \\ \frac{\partial \hat{\phi}}{\partial t} + \mathbf{v} \cdot \nabla \hat{\phi} = 0. & (3.5c) \end{cases} \quad (3.5)$$

So (3.5b) and (3.5c) simply state that Γ and $\hat{\Gamma}$ move with the same velocity field \mathbf{v} . In [14], only a normal velocity is considered. But a tangential component is perfectly

possible. Again, we emphasize that a tangential velocity does not affect the geometry of the interface, but it does affect the motion of $\hat{\Gamma}$ and hence of the region of interest.

This method is closely related to the work of [20] on the motion of a curve in three dimensions with a level set approach. The curve is represented as the intersection of Γ and $\hat{\Gamma}$ and its motion can be expressed as the evolution of the two level set functions ϕ and $\hat{\phi}$ according to (3.5b) and (3.5c). But in contrast with region tracking and (3.5a), the velocity then depends on the geometric properties of the curve (e.g. tangent, normal, binormal, curvature, torsion), and not only on Γ .

3.2.2 Previous work on transport and diffusion of a material quantity

In [177, 3], two similar methods are proposed to model the evolution of a material quantity f along a moving interface in the level set framework. Besides being passively advected, the interfacial data are also scaled due to the local compression/expansion of the interface, as a result of mass conservation, and diffused along the interface. This is achieved by the following system of coupled Eulerian PDEs:

$$\begin{cases} \frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi = 0, \\ \frac{\partial f}{\partial t} + \operatorname{div}_{\Gamma}(f\mathbf{v}) = \Delta_{\Gamma} f, \end{cases} \quad (3.6)$$

where $\operatorname{div}_{\Gamma}$ is the intrinsic divergence operator and Δ_{Γ} is the intrinsic Laplacian operator on the interface, often called the Laplace-Beltrami operator.

This method requires to define the material quantity on the whole space. Just like the definition of the level set function off Γ is arbitrary, any extension f which agrees with the data on the interface can be considered. Such an embedding of the data was previously used to solve variational problems and PDEs on *fixed* implicit interfaces [13]. In some applications, the data may have a natural extension off the interface. In other applications, one may build an extension f by some numerical procedure, that we discuss in Section 3.3.

The mass conservation behavior, responsible for the divergence form in (3.6), and the diffusion behavior are both related to the physical interpretation of the data as concentrations. However, this interpretation is not relevant to all types of interface data. For example, mass conservation does not make sense in the case of an evolving textured surface. In the next subsection, we study the case of passively advected interfacial data by taking the region tracking method of Subsection 3.2.1 as a source of inspiration.

3.2.3 LSID: Level sets with some interfacial data

If we go back to Subsection 3.2.1 and take a closer look at (3.5), we notice that the zero level set of $\hat{\phi}$ does not play a particular role. All the level sets of $\hat{\phi}$ evolve according to \mathbf{v} . Actually, the evolution equation for $\hat{\phi}$ is nothing but a passive advection equation with an extrinsic velocity field.

Hence we can go beyond the interpretation of $\hat{\phi}$ as the level set function of a hyper-surface and substitute to it a general scalar or vector-valued function f coding for some interfacial data. This slight change immediately generalizes the tracking method of [14] to the evolution of any quantity passively advected with a moving level set interface. The corresponding PDEs are

$$\begin{cases} \frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi = 0, & (3.7a) \\ \frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f = 0. & (3.7b) \end{cases} \quad (3.7)$$

In the sequel, we will refer to this approach as *level sets with some interfacial data* (LSID). We can now regard (3.4) and (3.5) as a particular case of LSID with f restricted to a scalar function, the sign of which tags the region of interest. Interestingly, the above method can be used to upgrade a traditional level set evolution without any modification of the existing PDE.

At first sight, the level set function and the data play symmetric roles in (3.7) and the two sub-equations are decoupled. Actually, this is only the case when the velocity field is given *a priori*. In most problems, the velocity field depends on the geometric properties of Γ (e.g. normal and curvature) and hence on ϕ . In some problems, the velocity field may also depend on the values of the interfacial data.

3.2.4 LSPC: Level sets with a point correspondence

LSID is versatile but it suffers from several limitations. First, it does not provide an explicit point correspondence, which makes it inadequate for some applications. Second, it requires to solve a PDE for each scalar component of the data, which may be prohibitive in some applications.

In order to overcome these problems we propose to maintain, rather than the interfacial data themselves, some unambiguous coordinates of the points of the interface. This implies a choice of a coordinate system, for example a global parameterization of the interface. However, obtaining a global parameterization of a complex shape is difficult. Moreover, critical points and periodic conditions would be tricky to handle. Finally, reintroducing a parameterization is quite unnatural in the implicit framework.

A very convenient alternative is to use the initial Cartesian coordinates of the points of the interface in the embedding space \mathbb{R}^n . We regard them as vector-valued interfacial data that we evolve with LSID. This is equivalent to considering a function $\psi : \mathbb{R}^n \times \mathbb{R}^+ \mapsto \mathbb{R}^n$ such as

$$\psi(\mathbf{x}, 0) = \mathbf{x} \quad (3.8)$$

and

$$\begin{cases} \frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi = 0, & (3.9a) \\ \frac{\partial \psi}{\partial t} + D\psi \mathbf{v} = 0, & (3.9b) \end{cases} \quad (3.9)$$

where $D\psi$ stands for the Jacobian matrix of ψ . For each point \mathbf{x} of the interface at time t , $\psi(\mathbf{x}, t)$ holds the position that this point was occupying at time $t = 0$. In other words,

$\psi(\cdot, t)$ provides an explicit backward point correspondence from the current interface $\Gamma(t)$ to the initial one $\Gamma(0)$. Off the interface, the point correspondence is driven by the extension of the velocity. In most applications, the latter is arbitrary, so the values of ψ off $\Gamma(t)$ do not have a physical meaning.

Once the point correspondence is available, the evolution of any other passively advected interfacial data with (3.7b) can be bypassed. We build any such data by composition of the initial data f_0 with the correspondence function ψ . As a matter of fact, $f = f_0 \circ \psi$ formally satisfies (3.7b):

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f = (\nabla f_0 \circ \psi) \cdot \left(\frac{\partial \psi}{\partial t} + D\psi \mathbf{v} \right) = 0. \quad (3.10)$$

In the sequel, we will refer to this approach as *level sets with a point correspondence* (LSPC). Equation (3.9b) is the Eulerian counterpart of the Lagrangian ODE which gives the forward point correspondence. The latter can be represented by a function $\hat{\psi} : \mathbb{R}^n \times \mathbb{R}^+ \mapsto \mathbb{R}^n$ such as

$$\begin{cases} \hat{\psi}(\mathbf{x}, 0) = \mathbf{x}, \\ \frac{\partial \hat{\psi}}{\partial t} = \mathbf{v} \circ \hat{\psi}. \end{cases} \quad (3.11)$$

This Lagrangian approach is used in [143, 68, 169] to circumvent the loss of the point correspondence in the level set method. But the Eulerian PDE (3.9b) has two important advantages over the Lagrangian ODE (3.11). First, it is numerically more stable since the computations are performed on a fixed grid. More importantly, it performs automatic deleting of merging characteristics, whereas this task requires intricate delooping algorithms in the Lagrangian approach. Moreover, a forward correspondence may not exist if the evolution forms shocks; the interface may even collapse and merely disappear. In such cases, (3.11) is not relevant. That is the reason why LSPC only focuses on the backward point correspondence. In some applications however, the forward correspondence is needed and it is necessary to invert the ψ map at a postprocessing stage.

If the velocity field is sufficiently smooth [46, 163], (3.9b) generates a one-parameter family of diffeomorphisms. So does (3.11) and by the chain rule we get $\psi(\cdot, t) \circ \hat{\psi}(\cdot, t) = \hat{\psi}(\cdot, t) \circ \psi(\cdot, t) = \text{Id}$, $\forall t$. This is not true in general, as illustrated in some of the numerical experiments that we report in Section 3.4. Typically, ψ fails to be surjective and develops a discontinuity in the presence of a shock (see Experiments 2 and 4) and fails to be injective in the presence of a rarefaction (see Experiment 3).

3.3 Numerical algorithms

In this section, we describe in details a numerical implementation of LSID and LSPC.

3.3.1 Level set reinitialization and data extension

On the one hand, the definition of the level set function and of the data off the interface is arbitrary in the continuous formulation. But on the other hand, from a numerical point

of view, flat and/or steep regions that develop in the level set function and in the data during the evolution can dramatically decrease the accuracy of the computed solution. This motivates the use of the signed distance function to the interface as the level set function.

The signed distance function to the interface is defined by:

$$\begin{cases} \phi(\mathbf{x}, t) = -d(\mathbf{x}, \Gamma(t)) & \text{if } \mathbf{x} \text{ is inside } \Gamma(t), \\ \phi(\mathbf{x}, t) = 0 & \text{if } \mathbf{x} \in \Gamma(t), \\ \phi(\mathbf{x}, t) = d(\mathbf{x}, \Gamma(t)) & \text{if } \mathbf{x} \text{ is outside } \Gamma(t), \end{cases} \quad (3.12)$$

where $d(\mathbf{x}, \Gamma(t)) = \min_{\mathbf{y} \in \Gamma(t)} \|\mathbf{x} - \mathbf{y}\|$. The signed distance function verifies

$$\|\nabla\phi\| = 1 \quad (3.13)$$

almost everywhere (more precisely outside of the skeleton of the interface).

Similarly, an extension of the data constant along the normal to the isolevels of ϕ , i.e. of the form

$$\nabla f \cdot \nabla\phi = 0 \quad (3.14)$$

is numerically advantageous.

An interesting approach to counteract the loss of resolution of the level set function is to use a particular extension of the velocity that maintains the signed distance property [2, 60]. However, this approach does not have a counterpart for the data function. A more common approach is to occasionally apply a reinitialization procedure which restores the signed distance property. This can be done either by applying a Fast Marching technique [142] or by considering the steady state solution to the following PDE [156]:

$$\frac{\partial\phi}{\partial\tau} + \text{sign}(\phi_0) (\|\nabla\phi\| - 1) = 0. \quad (3.15)$$

Similarly, an extension of the data fulfilling (3.14) can be obtained either in combination with a reinitialization by Fast Marching [2] or by running the following PDE [187]:

$$\frac{\partial f}{\partial\tau} + \text{sign}(\phi) (\nabla f \cdot \nabla\phi) = 0. \quad (3.16)$$

Independently of numerical accuracy, a reinitialization procedure and an extension procedure are required when using a localized version of the level set method, such as the narrow band methodology [1] or the PDE-based fast local level set method [118]. Indeed, when the narrow band is rebuilt, proper values must be assigned to the level set function and to the data at the new active grid points.

In our implementation, we perform both the reinitialization and the extension with a single pass of Fast Marching, as proposed in [2]. In this work, the authors build an extension of the velocity and a temporary signed distance function as a byproduct, at each iteration. In our case, we run the Fast Marching procedure only occasionally: when the narrow band needs rebuilding, and otherwise periodically to keep the interface and the data well-resolved.

3.3.2 Keeping the point correspondence onto the initial interface

This subsection specifically applies to LSPC. In this case, the interfacial data are the coordinates of the points of the initial interface. Moreover, due to the extension procedure described in Subsection 3.3.1, the values of ψ in the whole domain belong to $\Gamma(0)$.

We show that if the initial correspondences belong to the zero level set of ϕ_0 , this remains true throughout the evolution. We note $\nu = \phi_0 \circ \psi$. We obtain from (3.9b)

$$\frac{\partial \nu}{\partial t} + \mathbf{v} \cdot \nabla \nu = 0. \quad (3.17)$$

So if $\nu(\cdot, 0) \equiv 0$, we have $\nu(\cdot, t) \equiv 0, \forall t$. However, a numerical scheme for (3.9b) will in general move ψ outside of $\Gamma(0)$. We propose two adaptations to counteract this numerical drift. We take our inspiration in the work of [100] on solving PDEs mapping into an implicitly-defined target manifold.

The first adaptation consists in projecting the variation of ψ on the tangent plane of the initial interface. Without loss of generality, we can assume that ϕ_0 is a signed distance function. In this case, the projection operator can be written in a very simple form. Thus, (3.9b) is replaced with

$$\frac{\partial \psi}{\partial t} + \left[\mathbf{I} - (\nabla \phi_0 \circ \psi) (\nabla \phi_0 \circ \psi)^T \right] D\psi \mathbf{v} = 0. \quad (3.18)$$

Equation (3.18) is mathematically equivalent to (3.9b), but it turns out to be numerically advantageous. In particular, the preservation of ν is enforced more directly than in (3.17) since we now have

$$\frac{\partial \nu}{\partial t} = (\nabla \phi_0 \circ \psi) \cdot \frac{\partial \psi}{\partial t} = 0. \quad (3.19)$$

The second adaptation consists in projecting the correspondences on the initial interface, in other words replacing each values of ψ by the closest point on Γ_0 .

$$\psi \leftarrow P_{\Gamma_0} \circ \psi. \quad (3.20)$$

Generally, a closest point algorithm must be used to compute the projection operator P_{Γ_0} . However, if ϕ_0 is a signed distance function, a simple expression can be used outside of the skeleton of Γ_0 :

$$P_{\Gamma_0}(\mathbf{x}) = \mathbf{x} - \phi_0(\mathbf{x}) \nabla \phi_0(\mathbf{x}). \quad (3.21)$$

In our implementation of LSPC, we use these two adaptations in combination. We apply the modified PDE (3.18) at each iteration, and we reproject the values of the correspondence function with (3.21) before starting each reinitialization/extension procedure.

3.3.3 Finite-difference schemes

On the one hand, the proper numerical scheme for the level set evolution equation (3.7a) and (3.9a) depends on the properties of the velocity field. All the schemes needed in the numerical experiments of Section 3.4 are taken from [112, 143, 156, 187] and are described in details in Appendix B.1. On the other hand, a passive advection scheme (*cf* Appendix

B.1) can be adopted for (3.7b) and (3.9b) as soon as the velocity field does not depend either on the interfacial data or on the correspondence, in other words as soon as \mathbf{v} is extrinsic to f and ψ .

Moreover, to estimate one-sided space derivatives we adopt the third-order weighted essentially non-oscillatory (WENO) scheme derived in [73, 72] (*cf* Appendix B.2). Finally, with respect to time differencing, a simple Euler scheme has proven sufficient in our experiments. Of course, we could resort to higher-order schemes, such as the total-variation diminishing Runge-Kutta scheme of third order devised in [145].

3.3.4 Overview of the algorithm

For sake of clarity, we give an overview of our implementation of LSPC in Algorithm 1 below.

Algorithm 1 Algorithm for level sets with a point correspondence

for all iterations **do**

 Compute the velocity field.

 Evolve the level set function by (3.9a).

 Evolve the correspondence function by (3.18).

if the narrow band need rebuilding, otherwise periodically **then**

 Reproject the correspondence function by (3.21).

 Run the reinitialization/extension procedure.

3.4 Experimental results

In this section, we report on some numerical experiments that demonstrate the robustness and accuracy of LSID and LSPC in a wide range of situations, including normal and tangential velocity fields, large deformations, shocks, rarefactions and topology changes. Whenever the exact solution for the position of the interface, for the data and for the correspondence are known, we measure and we discuss the numerical error.

3.4.1 Definition of the error measures

The computation of the numerical error is not straightforward. In particular, a direct comparison of the computed data and correspondence with the exact solution is not possible since the computed interface and the exact interface differ. In all our experiments, the error is measured as follows.

To get the error on the position of the interface, we extract the zero isocontour of the computed level set function using the marching cubes algorithm [89]. At the resulting vertices, we compute the distance to the exact interface. Then we accumulate the point-wise errors along the contour to get the mean error (L_1 norm), the mean square error (L_2 norm), and the maximum error (L_∞ norm).

To get the error of LSID on the interfacial data, we sample the exact interface and we interpolate the computed data at these points using cubic spline interpolation. Due to the extension procedure (Subsection 3.3.1), this is equivalent to evaluating the computed data at the closest points on the computed interface. Then we subtract the exact data and we compute the L_1 , L_2 and L_∞ norms. We proceed similarly to get the error of LSPC on the correspondence.

3.4.2 2D experiments

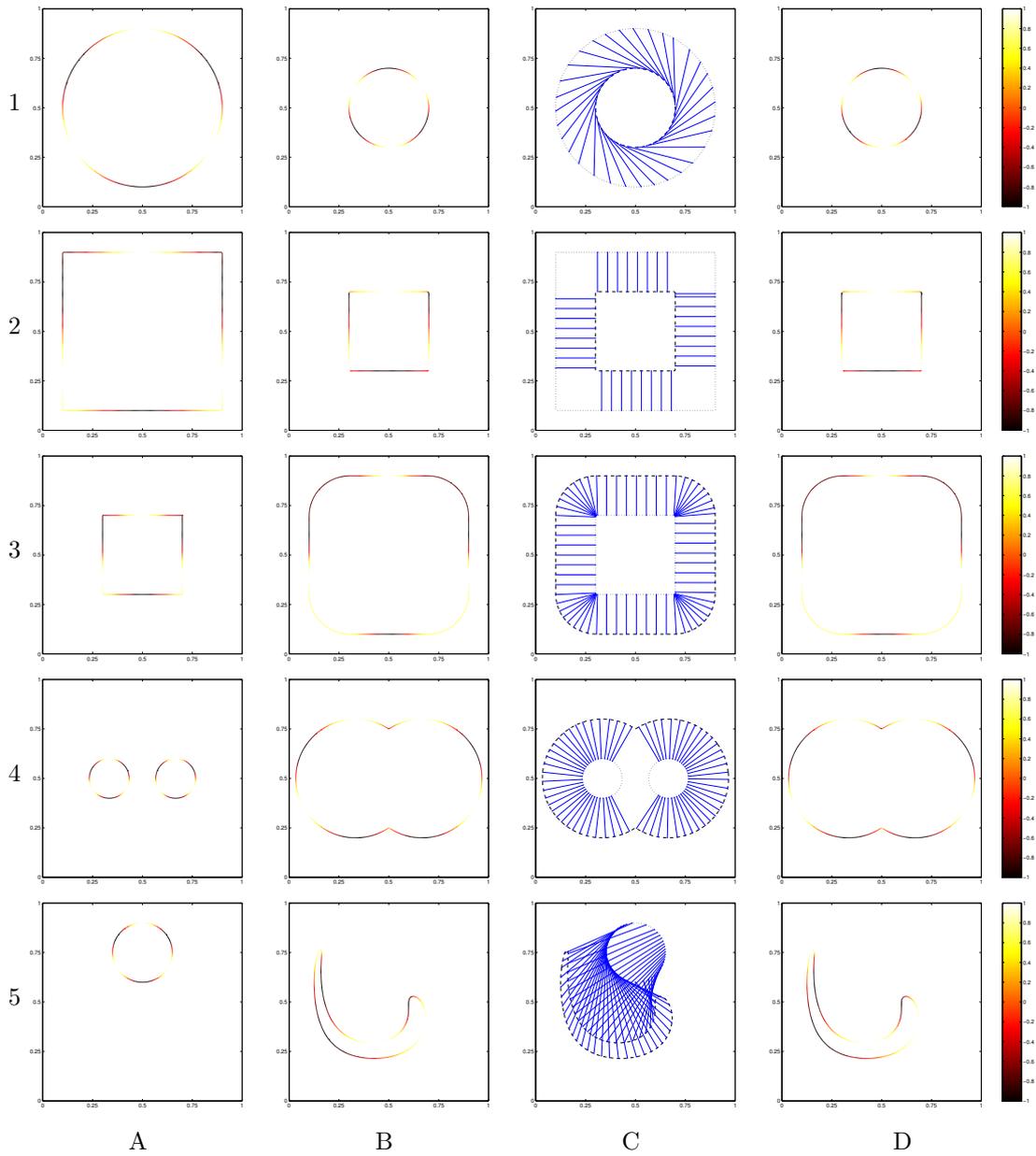


Figure 3.1: Results of LSID (column B) and LSPC (columns C&D) in several 2D test cases.

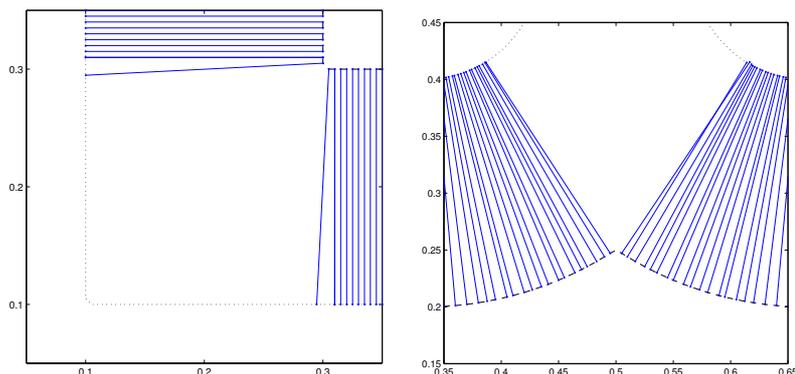


Figure 3.2: Some detailed views of Figure 3.1-C2 (left) and Figure 3.1-C4 (right).

Throughout this subsection, the spatial domain is the unit square $[0, 1] \times [0, 1]$ and we use a time step of $\Delta t = \Delta x/10$ and a $6\Delta x$ -thick narrow band. The reinitialization/extension procedure is started when the distance of the interface to the borders of the narrow band gets below $3\Delta x$. In all five experiments, the initial data are a function of the central angle θ : $f_0(\theta) = \sin(3\theta)$.

Figure 3.1 demonstrates LSID and LSPC in several 2D test cases. The curves displayed in this figure are obtained from the computed level set function using a marching squares algorithm [89]. At the resulting vertices, we sample the computed data and correspondence function using cubic spline interpolation.

Each row shows a different experiment. In each experiment, column A shows the initial interface colored with the initial data. Column B shows the interface at $t = 0.2$ colored with the data computed with LSID. The last two columns show the results of LSPC at the same time instant. In column C, the correspondence function is plotted at one out of ten vertices. The initial and the current interfaces are plotted too, with a dotted line and a dashed line respectively. Column D shows the interface colored with the transformed data $f_0 \circ \psi$.

The errors for the different experiments at $t = 0.2$ for different grid sizes (50^2 , 100^2 , 200^2) are given in Tables 3.1-3.5. In the next paragraphs, we describe in detail the purpose, the setting and the results of each experiment.

Experiment 1: a rotating and shrinking circle.

The initial interface is a circle with radius 0.4 centered at $(0.5, 0.5)$ and the velocity field is a combination of a unit inward normal speed and of an extrinsic rigid rotation of angular velocity 5. This example demonstrates that both normal and tangential velocities can be handled.

Experiment 2: a shrinking square.

The initial interface is a square with side length 0.4 centered at $(0.5, 0.5)$ and the velocity field is a unit inward normal speed. The evolution forms shocks at the angles of the square.

Discontinuities develop in the data and in the correspondence. Figure 3.2-left provides a detailed view of Figure 3.1-C2 with a denser representation for the correspondence. It shows that the expected discontinuities are fairly well recovered by LSPC.

Experiment 3: an expanding square.

The initial interface is a square with side length 0.2 centered at (0.5, 0.5) and the velocity field is a unit outward normal speed. The evolution forms rarefaction cones at the angles of the square. The many-to-one correspondence is successfully recovered by LSPC.

Experiment 4: the merging of two expanding circles.

The initial interface is composed of two circles with radius 0.1 centered at (1/3, 0.5) and (2/3, 0.5) and the velocity field is a unit outward normal speed. The two circles merge at $t = 2/15$ and two discontinuities develop in the data and in the correspondence. With no surprise, the topology change is handled automatically by the level set method. The discontinuities in the correspondence are also recovered by LSPC, as illustrated in Figure 3.2-right.

Experiment 5: a circle in a vortex velocity field.

The initial interface is a circle with radius 0.15 centered at (0.5, 0.75) and the evolution is driven by a non-constant vorticity velocity field that varies sinusoidally in time. It is defined by

$$\mathbf{v}(x, y) = 4 \cos \frac{\pi t}{T} \begin{pmatrix} \sin^2(\pi x) \sin(2\pi y) \\ -\sin(2\pi x) \sin^2(\pi y) \end{pmatrix}. \quad (3.22)$$

This velocity field at $t = 0$ is shown in Figure 3.3. This experiment is numerically more

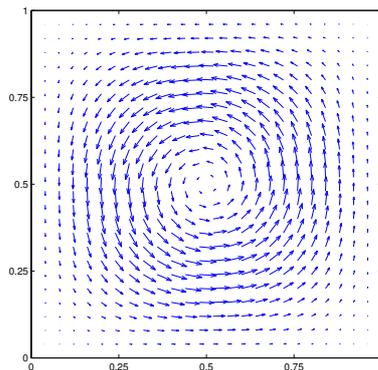


Figure 3.3: Velocity field used in Experiment 5.

challenging than the previous ones because the flow considerably stretches the interface. The exact solution is not available at all time, so at first sight we cannot measure the error. But the velocity reverses at time $T/2$, so the initial interface and the initial data should be recovered at time T . This provides a convenient way to evaluate the accuracy of our results. We take $T = 0.4$. The errors at $t = 0.4$ are given in Table 3.5.

3.4.3 3D experiments

We now demonstrate our method in three dimensions. In this subsection, the spatial domain is the unit cube $[0, 1] \times [0, 1] \times [0, 1]$ and we use a time step of $\Delta t = \Delta x/5$. In order to test the robustness to large deformations, we consider the following incompressible velocity field proposed by LeVeque [87]:

$$\mathbf{v}(x, y, z) = \cos \frac{\pi t}{T} \begin{pmatrix} 2 \sin^2(\pi x) \sin(2\pi y) \sin(2\pi z) \\ - \sin(2\pi x) \sin^2(\pi y) \sin(2\pi z) \\ - \sin(2\pi x) \sin(2\pi y) \sin^2(\pi z) \end{pmatrix}. \quad (3.23)$$

This flow is a surimposition of a deformation in the xy plane with a deformation in the xz plane. As in Experiment 5, the initial interface and the initial data should be recovered at time T . In the next paragraphs, we apply this velocity field to two different interfaces.

Experiment 6: a deforming plane.

The initial interface is the plane $x = 0.5$ and the initial data are given by

$$f_0(x, y, z) = \sin(10\pi y) \sin(10\pi z). \quad (3.24)$$

In this experiment, we take $T = 0.8$. Figure 3.4 shows the computed interface, the data and the correspondence at $t=0, 0.2$ and 0.4 . Rather than plotting the point correspondence which is difficult to visualize in three dimensions, we color the interface with a checkerboard texture obtained by composition with ψ . The errors at $t = 0.8$ for different grid sizes (50^3 , 100^3 and 200^3) are given in Table 3.6.

Experiment 7: a deforming sphere.

The initial interface is a sphere with radius 0.15 centered at $(0.35, 0.35, 0.35)$ and the initial data are given by

$$f_0(x, y, z) = (x - 0.35)(y - 0.35)(z - 0.35) / 0.15^3. \quad (3.25)$$

In this experiment, we take $T = 1.6$. Figure 3.5 shows the results at $t=0, 0.4$ and 0.8 . The errors at $t = 1.6$ are given in Table 3.7.

3.4.4 Comments on the errors

In Tables 3.1-3.7, we observe a regular decrease of the L_1 , L_2 and L_∞ norms of the errors of LSID and LSPC when the dimension of the computational grid increases. There are a few exceptions to this behavior, shown in bold fonts. They can be explained easily: in Experiment 2 and Experiment 4, the discontinuities that develop in the data and in the correspondence cause a stagnation of the L_∞ norm of the error.

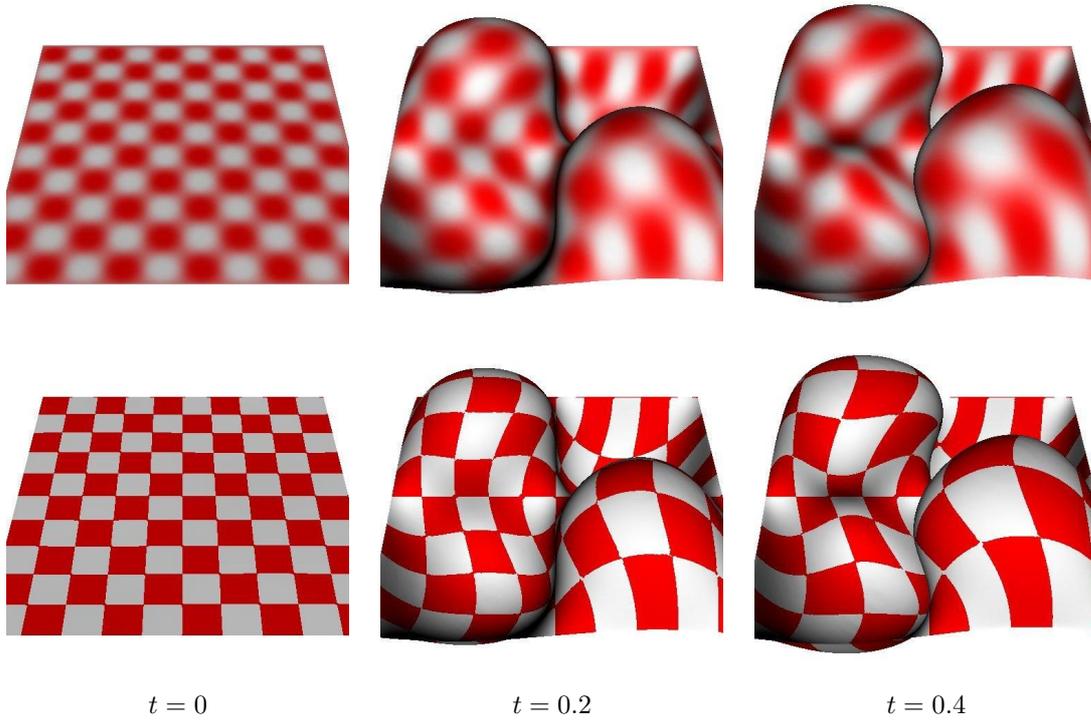


Figure 3.4: Results of LSID (top) and LSPC (bottom) in Experiment 6 at different times.

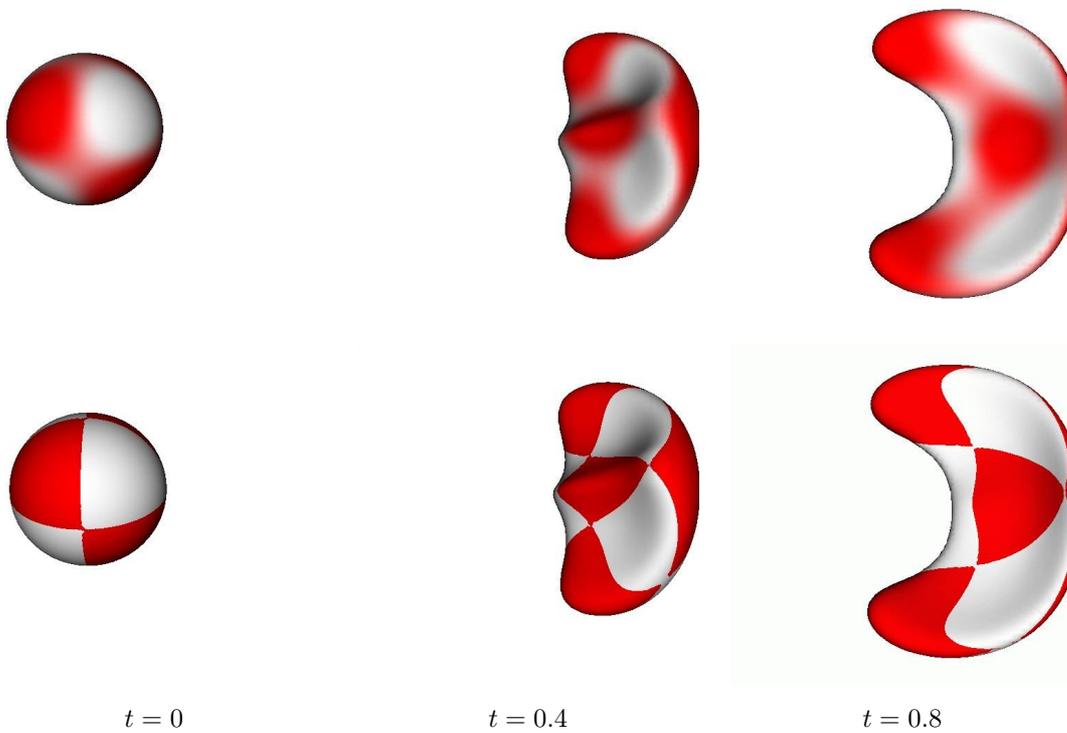


Figure 3.5: Results of LSID (top) and LSPC (bottom) in Experiment 7 at different times.

Grid size		50×50	100×100	200×200
Surface error	L_1	2.6e-4	1.6e-4	1.1e-4
	L_2	2.7e-4	1.7e-4	1.2e-4
	L_∞	3.9e-4	3.0e-4	1.9e-4
LSID Data error	L_1	1.8e-2	1.2e-2	6.3e-3
	L_2	2.1e-2	1.4e-2	7.3e-3
	L_∞	3.9e-2	2.1e-2	1.2e-2
LSPC Correspondence error	L_1	7.3e-4	4.7e-4	2.7e-4
	L_2	8.5e-4	5.6e-4	3.3e-4
	L_∞	1.4e-3	1.3e-3	6.4e-4

Table 3.1: Errors for Experiment 1 at $t = 0.2$.

Grid size		50×50	100×100	200×200
Surface error	L_1	2.0e-5	6.3e-6	3.6e-6
	L_2	3.4e-5	1.7e-5	7.4e-6
	L_∞	1.0e-4	6.7e-5	4.0e-5
LSID Data error	L_1	2.4e-2	1.3e-2	6.5e-3
	L_2	1.1e-1	7.7e-2	5.5e-2
	L_∞	7.4e-1	6.7e-1	6.5e-1
LSPC Correspondence error	L_1	7.1e-3	3.2e-3	1.6e-3
	L_2	3.1e-2	2.0e-2	1.4e-2
	L_∞	1.6e-1	1.4e-1	1.4e-1

Table 3.2: Errors for Experiment 2 at $t = 0.2$.

Grid size		50×50	100×100	200×200
Surface error	L_1	3.1e-3	1.5e-3	7.2e-4
	L_2	4.7e-3	2.3e-3	1.1e-3
	L_∞	9.1e-3	4.4e-3	2.1e-3
LSID Data error	L_1	1.9e-2	1.0e-2	5.4e-3
	L_2	3.1e-2	1.8e-2	9.6e-3
	L_∞	1.0e-1	6.6e-2	4.3e-2
LSPC Correspondence error	L_1	5.8e-3	2.8e-3	1.4e-3
	L_2	8.2e-3	4.3e-3	2.3e-3
	L_∞	1.9e-2	1.3e-2	8.3e-3

Table 3.3: Errors for Experiment 3 at $t = 0.2$.

Grid size		50×50	100×100	200×200
Surface error	L_1	2.6e-4	7.7e-5	3.8e-5
	L_2	3.8e-4	1.4e-4	6.5e-5
	L_∞	1.3e-3	1.3e-3	6.8e-4
LSID Data error	L_1	6.7e-2	1.8e-2	5.6e-3
	L_2	8.3e-2	5.4e-2	2.7e-2
	L_∞	3.7e-1	5.2e-1	3.9e-1
LSPC Correspondence error	L_1	4.3e-3	1.6e-3	8.1e-4
	L_2	2.6e-2	1.4e-2	9.8e-3
	L_∞	2.1e-1	1.9e-1	1.9e-1

Table 3.4: Errors for Experiment 4 at $t = 0.2$.

Grid size		50×50	100×100	200×200
Surface error	L_1	6.3e-3	2.2e-3	1.1e-3
	L_2	7.0e-3	2.6e-3	1.3e-3
	L_∞	1.1e-2	5.3e-3	3.1e-3
LSID Data error	L_1	2.7e-1	8.5e-2	3.1e-2
	L_2	3.4e-1	1.3e-1	5.3e-2
	L_∞	7.2e-1	3.7e-1	1.9e-1
LSPC Correspondence error	L_1	1.4e-2	4.2e-3	2.2e-3
	L_2	2.2e-2	6.7e-3	3.3e-3
	L_∞	5.7e-2	2.0e-2	9.8e-3

Table 3.5: Errors for Experiment 5 at $t = 0.4$.

Grid size		50^3	100^3	200^3
Surface error	L_1	5.5e-3	2.9e-3	1.5e-3
	L_2	7.4e-3	3.9e-3	2.1e-3
	L_∞	1.9e-2	9.5e-3	5.5e-3
LSID Data error	L_1	1.3e-1	5.6e-2	2.4e-2
	L_2	1.9e-1	7.3e-2	3.1e-2
	L_∞	5.6e-1	1.9e-1	1.1e-1
LSPC Correspondence error	L_1	2.1e-3	1.6e-3	9.7e-4
	L_2	2.5e-3	1.9e-3	1.1e-3
	L_∞	6.3e-3	6.2e-3	3.9e-3

Table 3.6: Errors for Experiment 6 at $t = 0.8$.

Grid size		50^3	100^3	200^3
Surface error	L_1	7.2e-3	3.4e-3	1.9e-3
	L_2	1.1e-2	5.2e-3	2.8e-3
	L_∞	4.4e-2	2.4e-2	1.2e-2
LSID Data error	L_1	4.5e-2	1.7e-2	7.1e-3
	L_2	5.6e-2	2.1e-2	9.1e-3
	L_∞	1.5e-1	8.3e-2	3.6e-2
LSPC Correspondence error	L_1	1.5e-2	6.3e-3	3.4e-3
	L_2	1.8e-2	7.2e-3	4.0e-3
	L_∞	4.4e-2	2.0e-2	1.2e-2

Table 3.7: Errors for Experiment 7 at $t = 1.6$.

3.5 Contributions of this chapter

In this chapter, we have extended the applicability of the level set method to the problems involving interfacial data and/or an explicit point correspondence.

- We have proposed two systems of coupled Eulerian PDEs:
 - the first one can handle passively advected interfacial data,
 - the second one can maintain an explicit backward point correspondence from the current interface to the initial one.
- We have described in details a robust numerical implementation of our approach:
 - a procedure for reinitializing the level set function and for extending the interfacial data and the point correspondence,
 - two adaptations to keep the point correspondence onto the initial interface,
 - the adequate numerical finite-difference schemes for the different PDEs.
- We have successfully tested our approach, in 2D and in 3D, in a wide range of situations, including normal and tangential velocities, large deformations, shocks, rarefactions and topology changes.
- In all our experiments, we have computed the numerical error and we have commented its variation with respect to the grid size.

A preliminary version of this work has appeared in the *International Conference on Computer Vision* [119] and in a technical report [120]. A journal version has been submitted to *Journal of Computational Physics* [121].

Chapter 4

Controlling Topology Changes in the Level Set Framework

In this chapter, we present two novel methods to exert a control on the topology during a level set evolution. The ability to automatically handle topology changes is a long claimed advantage of the level set method over explicit deformable models, but may not be desirable in some applications where some prior knowledge of the target topology is available. This is typically the case in biomedical image segmentation, where the topology of the organs and their mutual topological relations is prescribed by anatomical knowledge.

A topology preserving variant of the level set method has been proposed in [64] to overcome this problem. The level set function is evolved using a modified update procedure based on the concept of *simple point*, borrowed from digital topology [15]. The final mesh is extracted with a modified topology-consistent marching cubes algorithm. This method ensures that the resulting mesh has the same topology as the user-defined initial level set.

In this chapter, we propose two improvements to the topology preserving level set method. Our first extension allows to concurrently evolve several nested interfaces while preventing topology changes as well as mutual intersections. Unlike existing methods with explicit deformable models, either based on repulsion forces [116] or on a computationally intensive mesh-to-mesh intersection checking [91], our approach guarantees the absence of intersections and only requires efficient local computations. In Chapter 7, we will use this method to simultaneously extract the inner and outer interfaces of the cerebral cortex from MRI.

Our second extension applies specifically to the three-dimensional case. It allows to exert a more subtle control on the topology during a level set evolution. As a matter of fact, while a strict topology preservation is desired in some applications, it is often too restrictive. Since the different components of the object are not allowed to merge or to split up, the number of connected components remains constant throughout the evolution. This number must be known by the user *a priori* and the initial interface must be designed accordingly. Also, the sensitivity to initial conditions, which already limits the applicability and efficiency of the deformable models framework, is considerably increased. The initial

model must both have the same topology as the target shape and be close enough to the final configuration, otherwise the evolution is likely to be trapped in topological deadlocks including large geometric inconsistencies (see Figure 4.6 and Figure 4.7, row 4, column B).

In contrast, with our method, some prior knowledge of the target topology can be integrated without requiring that the topology be known exactly. Compared to [64], the sensitivity to initial conditions is greatly alleviated. Our method allows connected components to merge and split under certain conditions that ensure that no new handle is generated and no existing handle is closed. For example, an initial object with a spherical topology may split into several pieces, go through one or several mergings, and finally produce a certain number of surfaces, all of which are topologically equivalent to a sphere.

Our approach is based on an extension of the concept of *simple point* to multi-label images, that we have called *multisimple point*. It fills the gap between the standard level set method and the topology preserving level set method. We demonstrate the strength of our method in a wide range of numerical experiments, including the segmentation of the cortical surface and of blood vessels from medical images.

4.1 Background

4.1.1 Topology

Topology is the branch of mathematics that studies the properties of geometric figures that are invariant through *homeomorphisms* (bijective bicontinuous mappings), and in particular without regard to size and absolute position.

In two dimensions, the situation is particularly simple: any closed connected curve is homeomorphic to a circle. In three dimensions, a strong result of this theory is that any closed connected orientable surface is homeomorphic to a sphere with some number of handles. This number of handles is a topological invariant called the *genus*. For example, a sphere is of genus 0 and a torus is of genus 1. The genus g is directly related to another topological invariant called the *Euler characteristic* χ by the formula $\chi = 2 - 2g$. The Euler characteristic is of great practical interest because it can be calculated from any polyhedral decomposition of the surface by $\chi = V - E + F$, where V , E and F denote respectively the number of vertices, edges and faces of the polyhedron.

The curves and surfaces accounted for by the level set representation are closed and embedded, so they are ruled by the above results. However, this notion of topology based on homeomorphisms is unadapted to solid objects, because it completely disregards the embedding space. For this reason, it is sometimes referred as *intrinsic topology*.

For instance, a hollow disk has the same intrinsic topology as two disks, a hollow ball as two balls, and a knotted torus as a simple torus. We have to substitute the notion of *homotopy* to the notion of homeomorphism to be able to differentiate between these solids: two solids are said to be *homotopic* if one can be continuously deformed into the other. This alternate notion of topology satisfies our needs. For more details on homotopies, we refer the reader to an excellent course on algebraic topology [65].

4.1.2 Digital topology

The purpose of *digital topology* is to transpose the continuous concepts of topology to discrete spaces, for example to assign a precise topological meaning to binary images. In this theory, the central concept of homotopic deformation is clearly defined through the notion of *simple point*. An extensive discussion of these concepts can be found in [95]. In this section, some basic notions of digital topology are presented. All the following definitions are from [15].

A binary image I is composed of a foreground object X and its complement \bar{X} . Before trying to define a notion of topology, we need the concept of *connectivity*, which specifies the condition of adjacency that two points must fulfill to be regarded as connected. In 2D, two types of connectivity are commonly used: 4- and 8- connectivity. Two pixels are 4-adjacent if they share a face and 8-adjacent if they share at least a corner. In 3D, three types of connectivity prevail: 6-, 18- and 26-connectivity. Two voxels are 6-adjacent if they share a face, 18-adjacent if they share at least an edge and 26-adjacent if they share at least a corner. We note $N_n(x)$ the n -neighborhood of a point x , i.e. the set of grid points which are n -adjacent to x . We also set $N_n^*(x) = N_n(x) \setminus x$. The set of all n -connected components of a set X is denoted by $C_n(X)$.

In order to avoid topological paradoxes, different connectivities n and \bar{n} , must be used for X and \bar{X} . This leaves us with two pairs of compatible connectivities in 2D, (4,8) and (8,4), and four pairs in 3D, (6,26), (6,18), (18,6) and (26,6).

The concept of *simple point* is central to the method of [64] and to our work. A point of a binary object is said to be *simple* if it can be added or removed without changing the topology of both the object and the background, i.e. without changing the number of connected components, cavities and handles of both X and \bar{X} .

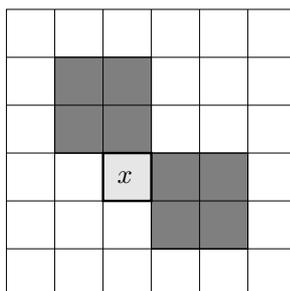


Figure 4.1: Illustration of the concept of simple point in a 2D example (see text).

Figure 4.1 illustrates in a simple 2D example the notion of simple point and its dependency on the choice of a compatible connectivity pair. The binary object X is colored in dark grey and the point x in light grey. If we choose the (4,8) connectivity pair, x is not a simple point since its addition/removal connects/disconnects the two dark squares. On the contrary, if we choose the (8,4) connectivity pair, x is a simple point: the two dark squares are already connected because they share a corner.

To formally define simple points, we need a few more definitions. The *geodesic neigh-*

borhood of x with respect to X of order k is defined recursively by:

$$\begin{aligned} N_n^1(x, X) &= N_n^*(x) \cap X, \\ N_n^k(x, X) &= N_M^*(x) \cap X \cap \bigcup_{y \in N_n^{k-1}(x, X)} N_n(y), \end{aligned}$$

where $M = 8$ in 2D and $M = 26$ in 3D.

Finally, the *topological numbers of a point x relative to X* are:

- In 2D:

$$\begin{aligned} T_4(x, X) &= |C_4 [N_4^2(x, X)]|, \\ T_8(x, X) &= |C_8 [N_8^1(x, X)]|. \end{aligned} \tag{4.1}$$

- In 3D:

$$\begin{aligned} T_6(x, X) &= |C_6 [N_6^2(x, X)]|, \\ T_{6+}(x, X) &= |C_6 [N_6^3(x, X)]|, \\ T_{18}(x, X) &= |C_{18} [N_{18}^2(x, X)]|, \\ T_{26}(x, X) &= |C_{26} [N_{26}^1(x, X)]|. \end{aligned} \tag{4.2}$$

The topological numbers are the number of connected components within certain geodesic neighborhoods. Following [15], in the above definition of topological numbers in the 3D case, there are two notations for the 6-connectivity: the notation “6+” and “6” are used when the dual connectivity is 18 and 26, respectively.

Interestingly, the topological numbers $T_n(x, X)$ and $T_{\bar{n}}(x, \bar{X})$ with respect to a binary object X and a compatible connectivity pair (n, \bar{n}) allow to differentiate between isolated, interior and border points as well as different kinds of junctions. For instance, an isolated point is unambiguously characterized by $T_n(x, X) = 0$ while an interior point by $T_{\bar{n}}(x, \bar{X}) = 0$. Similarly, a point is simple if and only if

$$T_n(x, X) = T_{\bar{n}}(x, \bar{X}) = 1. \tag{4.3}$$

The computation of the topological numbers is described in [16]. It only involves the 8-neighborhood in 2D and the 26-neighborhood in 3D, so it is very cheap computationally.

4.1.3 The topology preserving level set method

In [64], the authors use the concept of simple point to design a topology preserving variant of the level set method. The binary object of interest is the interior of the interface, i.e. the domain where the level set function ϕ is strictly negative: $X = \{x \mid \phi(x) < 0\}$. The digital topology of X is preserved during the evolution thanks to a modified update procedure detailed in Algorithm 2. Basically, the algorithm prevents non-simple grid points from changing sign in order to retain the initial digital topology throughout the evolution.

The isocontour extraction step also has to be adapted, in order to ensure that the topology of the generated explicit mesh representation is the same as the digital topology of the embedding level set function. A standard marching squares or marching cubes

Algorithm 2 Algorithm for the topology preserving level set method of [64]

```
for all iterations do
  for all grid points do
    Compute the new value of the level set function
    if the sign does not change then
      Accept the new value
    else {sign change}
      Compute the topological numbers
      if the point is simple then
        Accept the new value
      else
        Discard the new value
        Set instead a small value of the adequate sign
```

algorithm [89] does not meet this critical requirement. As a consequence, a topology-consistent isocontour algorithm must be designed.

The marching cubes algorithm is an efficient isocontour algorithm which generates a triangulated surface by tessellating each cubic cell of the domain depending on the sign of the level set function at its corners. The vertices of the triangulated surface are generated on the cell edges intersecting the isosurface, in other words on those having values of different signs, and the position of a vertex along an intersected edge is determined by linear interpolation of the two edge values. The way the vertices are connected to form triangles is stored in a case table generated off-line. In 3D, the 256 existing cases can be reduced, modulo symmetries, rotations and sign inversion, to the 15 representative cases shown in Figure 4.2. The corresponding algorithm in 2D is called the marching squares algorithm.

In [64], the authors work out a topology-consistent variant of the marching cubes algorithm by building a specialized case table for each choice of connectivity pair. Each case table is designed carefully to ensure that the topology of the output triangulated surface is the same as the digital topology of the level set function. Figure 4.3 illustrates that the mesh generated by this algorithm highly depends on the chosen connectivity pair. In this example, when using the (6,26), (6,18), (18,6) or (26,6) pairs, we obtain four different meshes from the same level set function.

Using the topology preserving level set method and the topology-consistent marching cubes algorithm in conjunction, with the same connectivity pair (n, \bar{n}) throughout the process, guarantees that the output mesh is topologically equivalent to the user-defined initial level set.

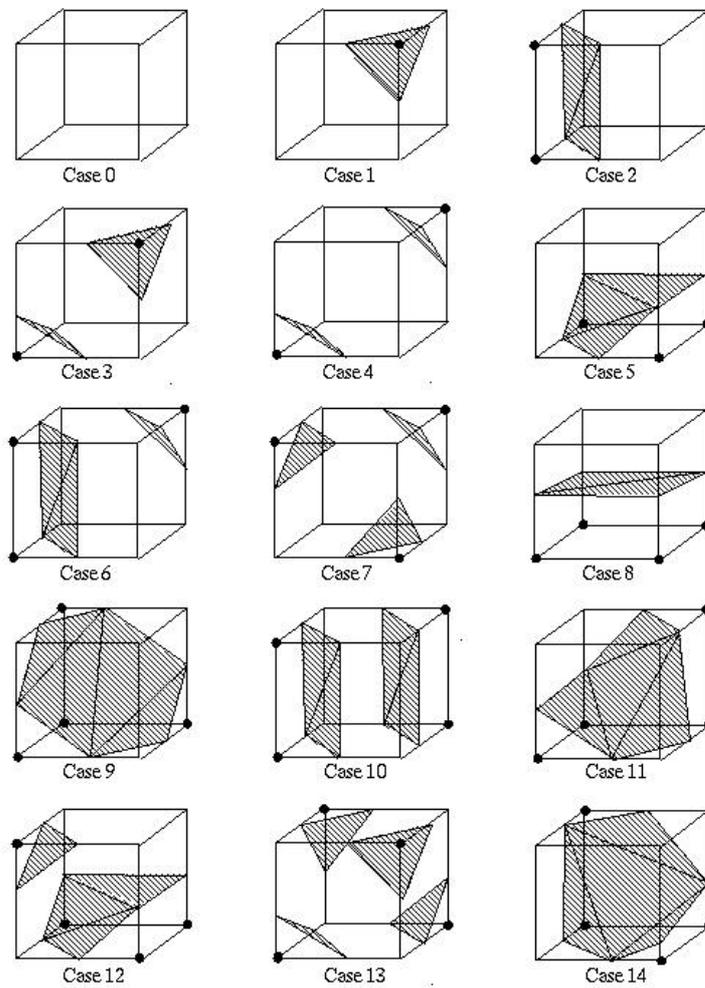


Figure 4.2: The 15 representative cases of the marching cubes algorithm.

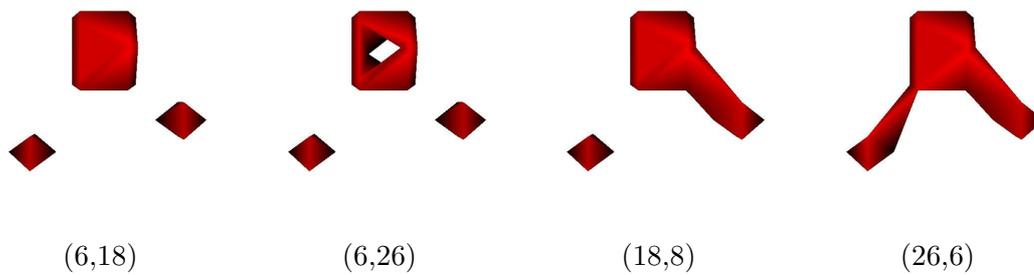


Figure 4.3: Mesh extracted from the same level set function by the topology-consistent marching cubes algorithm, when using different connectivity pairs.

4.2 The topology preserving nested level set method

In this section, we propose an extension of the topology preserving level set method which allows to concurrently evolve several nested interfaces in the level set framework, while preventing topology changes as well as mutual intersections. Unlike existing methods with explicit deformable models, either based on repulsion forces [116] or on a computationally intensive mesh-to-mesh intersection checking [91], our approach guarantees the absence of intersections and only require efficient local computations.

4.2.1 Digital topology criterion

We consider N nested digital objects X_1, X_2, \dots, X_N such that

$$X_1 \subset X_2 \subset \dots \subset X_N. \quad (4.4)$$

In order to avoid topological paradoxes, subsequent objects are assigned dual connectivity pairs. More precisely, the connectivity pair used for X_i is (n_i, \bar{n}_i) where

- if i is *odd*, $(n_i, \bar{n}_i) = (n, \bar{n})$.
- if i is *even*, $(n_i, \bar{n}_i) = (\bar{n}, n)$.

Moreover, the borders of the subsequent objects do not intersect. In other words,

$$\forall 1 \leq i < N, \forall x \in X_i, x \text{ is not } n_i\text{-adjacent to } \overline{X_{i+1}}. \quad (4.5)$$

In the above condition, the relevant connectivity types for X_i and $\overline{X_{i+1}}$ coincide thanks to the alternation of the connectivity pairs. An equivalent condition is

$$\forall 1 \leq i < N, \forall x \in X_i, N_{n_i}(x) \subset X_{i+1}. \quad (4.6)$$

The conditions for the preservation of the digital topology of each object and the prevention of mutual intersections are the following:

- x can be added to X_i if and only if $N_{n_i}(x) \subset X_{i+1}$ and x is a simple point relatively to X_i and the (n_i, \bar{n}_i) connectivity pair.
- x can be removed from X_i if and only if $N_{\bar{n}_i}(x) \subset \overline{X_{i-1}}$ and x is a simple point relatively to X_i and the (n_i, \bar{n}_i) connectivity pair.

4.2.2 Description of the algorithm

We now describe our algorithm for topology preserving nested level sets. The principle of our approach is similar to Subsection 4.1.3. We consider N interfaces represented as the zero level sets of some functions ϕ_i , $1 \leq i \leq N$, and we monitor the topology of the digital objects $X_i = \{x \mid \phi_i(x) < 0\}$. Basically, the algorithm prevents the sign changes that do not satisfy the conditions stated in the previous subsection. See Algorithm 3 for

more details. For the algorithm to work properly, the initial level set interfaces must be strictly nested, in the sense that

$$\forall 1 \leq i < N, \forall x, \quad \phi_i(x) < 0 \Rightarrow \forall y \in N_{n_i}(x), \phi_{i+1}(y) < 0 \quad (4.7)$$

Algorithm 3 Algorithm for the topology preserving nested level set method

for all iterations **do**

for all grid points **do**

for all $1 \leq i \leq N$ **do**

 Compute the new value of the level set function ϕ_i

if the sign does not change **then**

 Accept the new value

else {sign change}

if strictly negative value **then**

if $\forall y \in N_{n_i}(x), \phi_{i+1}(y) < 0$ and x is a simple point **then**

 Accept the new value

else

 Discard the new value, set instead a small positive value

else {positive value}

if $\forall y \in N_{\bar{n}_i}(x), \phi_{i-1}(y) \geq 0$ and x is a simple point **then**

 Accept the new value

else

 Discard the new value, set instead a small negative value

The isocontour extraction step also has to be considered. We propose to run the topology-consistent marching cubes algorithm separately on each level set function, using the relevant connectivity pairs. However, we have to check carefully that this procedure guarantees the absence of mutual intersections between the different output meshes. To this purpose, we examine all the sign configurations of ϕ_i and ϕ_{i+1} in a cubic cell, for all the possible connectivity pairs. First, we note that an intersection is possible only if several level set functions change sign in the same cell, so that some triangles of several interfaces are generated in this cell. Due to the strictly nested property (4.7), this happens only in a subset of the cases. More precisely:

- in 2D, it only happens when $n_i = 4$ and with the sign configuration shown in Figure 4.4. In this figure, the points of X_i are marked with black dots and the points of X_{i+1} with both black and white dots. The corresponding tessellations, plotted with a dashed line and a dotted line, do not intersect.
- In 3D:
 - When $n_i = 26$, only a single level set function can change sign in the same cubic cell.

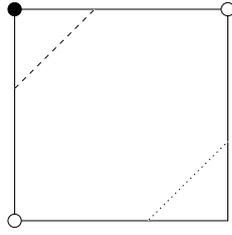


Figure 4.4: The only sign configuration in 2D yielding two interfaces in a same cubic cell. As expected, there is no intersection between the two output tessellations.

- When $n_i = 18$, two interfaces can traverse a same cell only in the sign configuration shown in Figure 4.5. Again, there is no intersection.

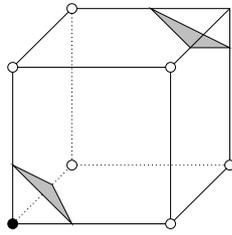


Figure 4.5: (see text).

- When $n_i = 6$, the representative cases 1, 2, 3, 5, 7 and 9 for X_i (*cf* Figure 4.2) must be examined, for all the possible configurations of X_{i+1} . We do not display all the cases here. The interested reader can make sure himself that no intersection is generated.

Finally, we have shown that using our topology preserving nested level set method along with the modified topology-consistent marching cubes algorithm, with the adequate connectivity pairs (n_i, \bar{n}_i) throughout the process, allows to evolve several nested interfaces and obtain some mutual intersection free meshes with a certified topology.

4.3 The genus preserving level set method

The simple point condition is a very efficient way to detect and prevent topology changes during a level set evolution. However, in many applications, a strict topology preservation is too restrictive.

On the one hand, preventing the creation of topological defects such as handles, which are difficult to retrospectively correct [63, 51, 82], is very advantageous. But on the other hand, the changes in the number of connected components during the evolution are less problematic, because the different components can easily be identified at post-processing time using standard region growing algorithms. This is also the case for the cavities, which can be interpreted as background \bar{n} -connected components.

Consequently, in three dimensions, we extend the simple point criterion to allow the different connected components of the object and of the background to merge, split, appear or disappear while ensuring that no new handle is generated and no existing handle is closed. Hence, the total genus is preserved during the evolution. With our criterion, for example, an initial object with a spherical topology may undergo various topology changes and finally produce a certain number of objects, potentially including one or several cavities. Our method ensures that all the surfaces of these objects are topologically equivalent to a sphere, whatever the number and history of splittings, mergings, etc. In some cases, at post-processing time, a subset of these components is selected by the user as the final output, typically the largest one if a single spherical component is needed, the others being imputable to some noise in the input data.

4.3.1 From simple points to multisimple points

We say that a point is *multisimple* if it can be added or removed without changing the number of handles. Contrarily to the case of simple points, the addition/removal of a multisimple point may merge/split the connected components of X and \bar{X} .

We note $C_n(x, X)$ the set of n -connected components of $X \setminus \{x\}$ that are n -adjacent to a point x . If the cardinality of this set is strictly greater than one, the addition or removal of x involves a merge or a split in X respectively. In order to capture these connectivity changes, we define two numbers, $T_n^+(x, X)$ and $T_{\bar{n}}^+(x, \bar{X})$, which we call *extended topological numbers*. The extended topological numbers of a point x relatively to an object X and a compatible connectivity pair (n, \bar{n}) are

$$\begin{aligned} T_n^+(x, X) &= |C_n(x, X)|, \\ T_{\bar{n}}^+(x, \bar{X}) &= |C_{\bar{n}}(x, \bar{X})|. \end{aligned} \tag{4.8}$$

It can be proven that a point is multisimple if and only if

$$\begin{cases} T_n^+(x, X) = T_n(x, X), \\ T_{\bar{n}}^+(x, \bar{X}) = T_{\bar{n}}(x, \bar{X}). \end{cases} \tag{4.9}$$

The interested reader may refer to [136] for a complete proof by Florent Ségonne. We note that an isolated point of the object or the background is a multisimple point, so that foreground components or cavities can be created or disappear during the evolution.

4.3.2 Description of the algorithm

With the concept of multisimple point in hand, we are now ready to describe our genus preserving level set method. Once again, the principle of our approach is similar to Subsection 4.1.3. Some additional bookkeeping is required to maintain a map L of labels encoding the different connected components of X and \bar{X} . The detailed procedure is given in Algorithm 4. When a point changes sign, the simple point condition, more restrictive, is checked first, because it is computationally cheaper. If the point is non-simple, then $C_n(x, X)$ and $C_{\bar{n}}(x, \bar{X})$ are computed in order to check the multisimple criterion.

Algorithm 4 Algorithm for the genus preserving level set method

```
for all iterations do
  for all grid points do
    Compute the new value of the level set function
    if the sign does not change then
      Accept the new value
    else {sign change}
      Compute the topological numbers
      if the point is simple then
        Accept the new value
        Update the map of connected components
      else {non-simple point}
        Compute the extended topological numbers
        if the point is multisimple then
          Accept the new value
          Update the map of connected components
        else
          Discard the new value
          Set instead a small value of the adequate sign
```

If the point x is candidate for addition to X , $C_n(x, X)$ can be deduced directly from the map L , while $C_{\bar{n}}(x, \bar{X})$ must be computed using a region growing algorithm. Indeed, the addition of x can change the global connectivity of \bar{X} , and the number of resulting connected components generally cannot be determined with local computations. Reciprocally, if the point x is candidate for removal from X , $C_{\bar{n}}(x, \bar{X})$ can be deduced directly from the map L , while $C_n(x, X)$ must be entirely recomputed.

Hopefully, these non-local computations are not necessary when dealing only with spherical components, the most common situation in practice. In this case, the computation of the extended topological numbers only involves local computations. Since no handle is present in the initial volume, the geodesic neighborhood $N_n^k(x, X)$ contains the necessary information about the connectivity of the different components of $C_n(x, X)$.

Some care must be taken in order to ensure that the map of labels L is correctly updated. In particular, when new connected components of X or \bar{X} are created by the addition or the removal of a multisimple point, some unused labels must be assigned to them.

Using our variant of the level set method along with the modified topology-consistent marching cubes algorithm of [64], with the same connectivity pair (n, \bar{n}) throughout the process, guarantees that the total genus of the output mesh is equal to the number of handles of the initial user-defined level set, while the number of connected components (including cavities) can vary.

4.4 Experimental results

Our topology preserving nested level set method is not demonstrated in this chapter. We will use it in Chapter 7 to simultaneously extract the inner and outer interfaces of the cerebral cortex from MRI.

In this section, we show the interest of using the genus preserving level set method for image segmentation. We present some numerical experiments on phantom and real data. In the following, we use a simplistic velocity field, which is a combination of an intensity-based term and a mean curvature term:

$$\mathbf{v}(\mathbf{x}, t) = [\lambda(I(\mathbf{x}) - I_{thres}) - H(\mathbf{x}, t)] \mathbf{N}(\mathbf{x}, t), \quad (4.10)$$

where I denotes the scalar image to be segmented, I_{thres} is a suitable intensity threshold which we assume to separate the object from the background, and λ is a weighting coefficient. Note that this evolution law, often called the *signed pressure force*, must be regarded as a special case of the region-based image segmentation methods reviewed in Subsection 2.3.1. The corresponding level set evolution equation is

$$\frac{\partial \phi}{\partial t} = \left[-\lambda(I - I_{thres}) + \frac{1}{n-1} \operatorname{div} \left(\frac{\nabla \phi}{\|\nabla \phi\|} \right) \right] \|\nabla \phi\|. \quad (4.11)$$

More complex images would require more elaborate evolution laws, like the ones described in Subsection 2.3.1. However, the choice of a particular segmentation method is not the matter here. We rather focus on the improvements brought by our approach, with respect to the management of topology, relatively to the standard level set method and to the topology preserving method of [64].

4.4.1 Synthetic data

Experiment 1: Segmentation of a 'C' shape

Figure 4.6 and Figure 4.7 show the segmentation of a phantom 'C' shape. We compare the behavior of three different methods: the standard level set method with automated topology changes (column A), the topology preserving level set method of [64] (column B) and our genus preserving level set method (column C). The differences of behavior are circled in the images. We have used two different initializations (a little sphere in Figure 4.6 and a larger box in Figure 4.7) to test the sensitivity of each method to initial conditions.

In this simple synthetic example, both standard level sets and genus preserving level sets yield the expected result. With the first initialization (Figure 4.6), these two methods behave exactly the same, because no handle is to be created: the surface splits into three pieces, one piece disappears and the two other pieces eventually merge. With the second initialization (Figure 4.7), they behave differently: whereas standard level sets temporarily generate a toroidal topology (row 3, column A), our method prevents the formation of a

handle (row 3, column B) by delaying a merging until a splitting in an other part of the object eliminates this eventuality.

In contrast, topology preserving level sets yield poor results. For the two different initializations, they get trapped in a topological deadlock. Although the final surface has the correct topology, it has large geometric errors (Figure 4.6 and Figure 4.7, row 4, column B): a filament linking the two ends of the 'C' and a separating membrane at the middle of the 'C'. These topological barriers, generated during the evolution to prevent topology changes, are difficult to retrospectively correct.

The behavior of our approach corresponds to a trade-off in between standard level sets and topology preserving level sets. Compared to the former, the formation or closing of handles is prevented. Compared to the latter, the ability to change topology under certain conditions greatly alleviates the sensitivity to initial conditions.

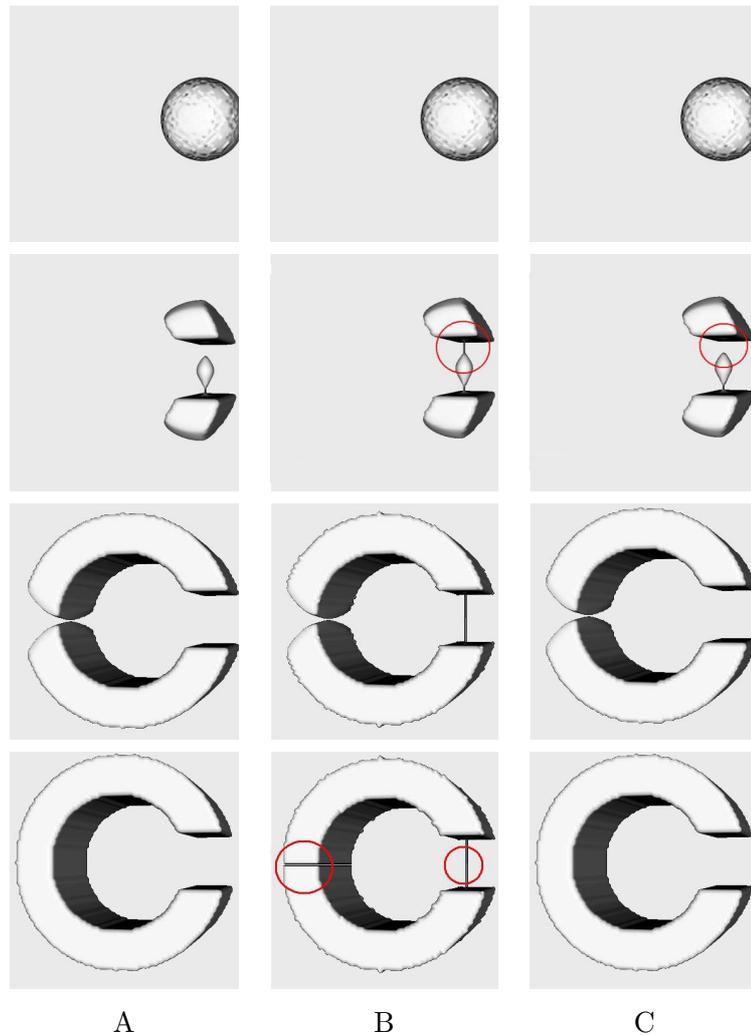


Figure 4.6: Segmentation of a 'C' shape with standard level sets (column A), with topology preserving level sets (column B) and with genus preserving level sets (column C).

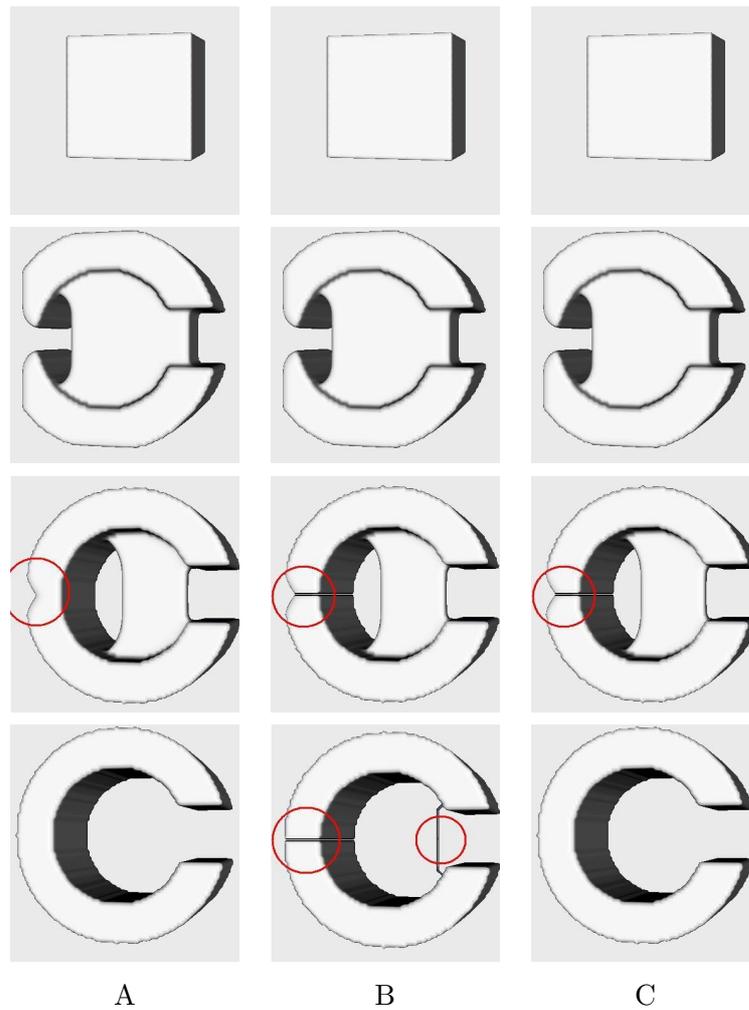


Figure 4.7: Segmentation of a 'C' shape with standard level sets (column A), with topology preserving level sets (column B) and with genus preserving level sets (column C).

Experiment 2: Formation of cavities

The second synthetic experiment, shown in Figure 4.8, illustrates the ability of our approach to generate cavities. The object to be segmented is a cube containing 3 large cavities. The initialization is composed of 10 seed points randomly selected in the whole domain. During the evolution, the components merge, vanish and produce cavities. As expected, the final surface is composed of 4 spherical components: the 3 cavities and the main object. Note that these different components can be easily extracted from the map of labels maintained during the evolution (*cf* Algorithm 4).

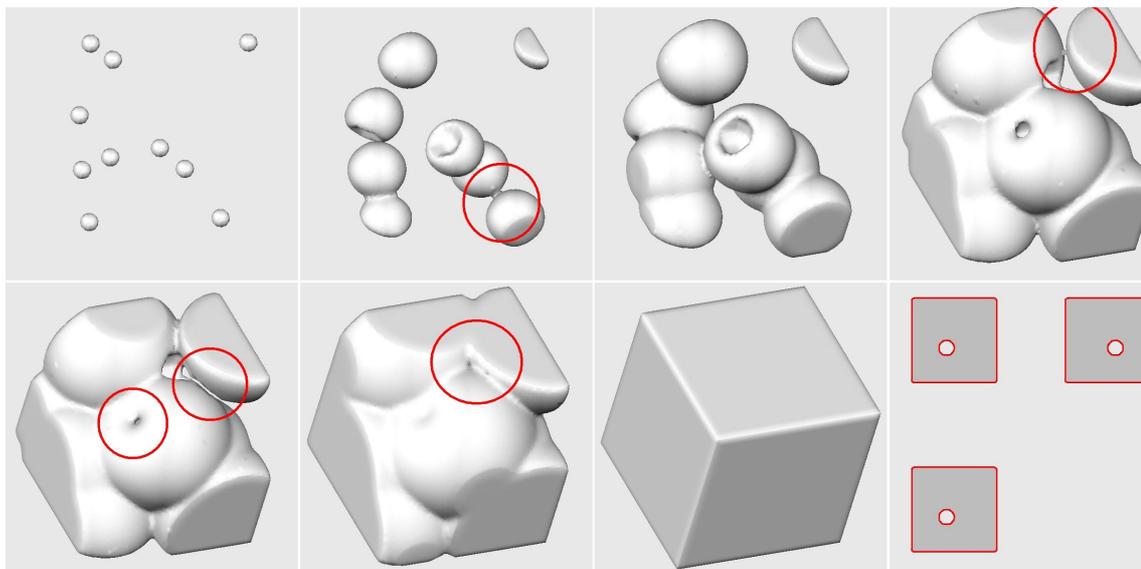


Figure 4.8: Segmentation of a cube containing three cavities with the genus preserving level set method, starting from 10 random seed points. The bottom right image shows one of the reconstructed cavities in some slices of the object along the three axes.

4.4.2 Real data

Experiment 3: Segmentation of the cortical surface

To illustrate the strength of our approach, we apply it to the difficult problem of cortical surface extraction from MRI. Excluding pathological cases, the cerebral cortex, which is a highly-folded thin sheet of gray matter, has the topology of a sphere. The extraction of accurate and topologically correct cortical representations is still an active research area [51, 63, 82].

Some slices of the input anatomical MR image used in this experiment are displayed in Figure 4.9. The initial surface is composed of 55 spherical components, automatically selected in a greedy manner, such that every point is located at a minimal distance of $10mm$ from the previous ones. Topology preserving level sets could not handle such an initialization, since the number of components would remain constant throughout the evolution. As a consequence, only one initial seed could be used, leading to a slower

segmentation process and potentially to topological deadlocks. As regards standard level sets, in this experiment they yield a final surface with 18 handles. In contrast, with our approach, illustrated in Figure 4.10, the components progressively merge and enclose cavities, without generating handles. The final surface has 6 spherical components: the cortical surface and 5 small cavities. In this experiment, these are very small cavities due to image noise. In some other cases, some larger cavities could be caused by pathological diseases such as tumors.

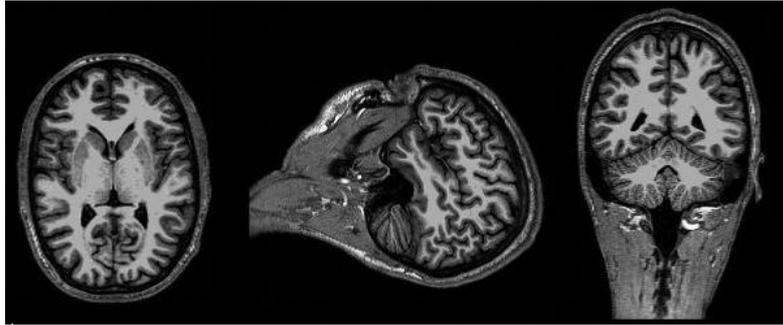


Figure 4.9: Some slices of the input anatomical MR image used in Experiment 3.

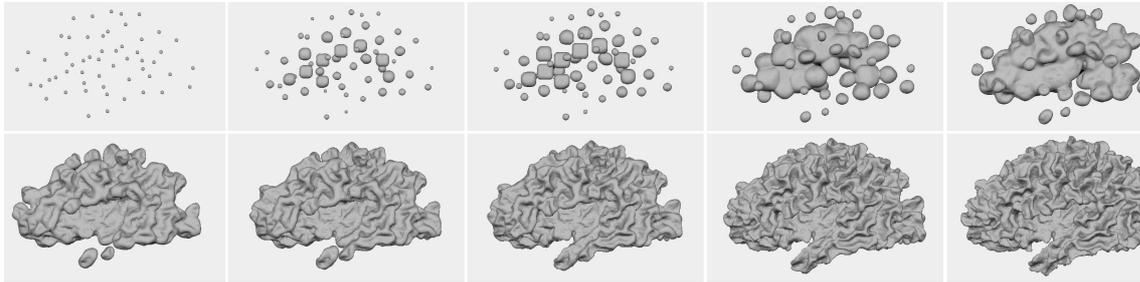


Figure 4.10: Segmentation of the cortical surface from MRI, starting from 55 seed points.

Experiment 4: Segmentation of blood vessels

Finally, we apply our method to the segmentation of blood vessels from MRA (Magnetic Resonance Angiography). In our study image, the expected topology of the vessels is the one of several components with no handles. Figure 4.11 shows the surface evolution with our approach, starting from two very different initializations.

In a first run (top row), the initial surface is composed of 20 seed points, automatically selected at the brightest locations in the image. The evolution iteratively merges most of the components, while preserving the total genus. The final surface has 3 spherical components.

In a second run (bottom row), the surface is initialized with the bounding box of the image. During the evolution, 9 mergings and 99 splittings occur. The final surface has 91 spherical components, 53 of which are isolated voxels due to image noise.

Note that despite very different initializations, the final representation is consistent in both cases. In contrast, topology preserving level sets would require to choose *a priori* the number of components, which is very difficult in this experiment. Moreover, when using an initialization with a bounding box, topology preserving level sets yield a final surface with many geometrical inconsistencies due to topological barriers, displayed in Figure 4.12-right.

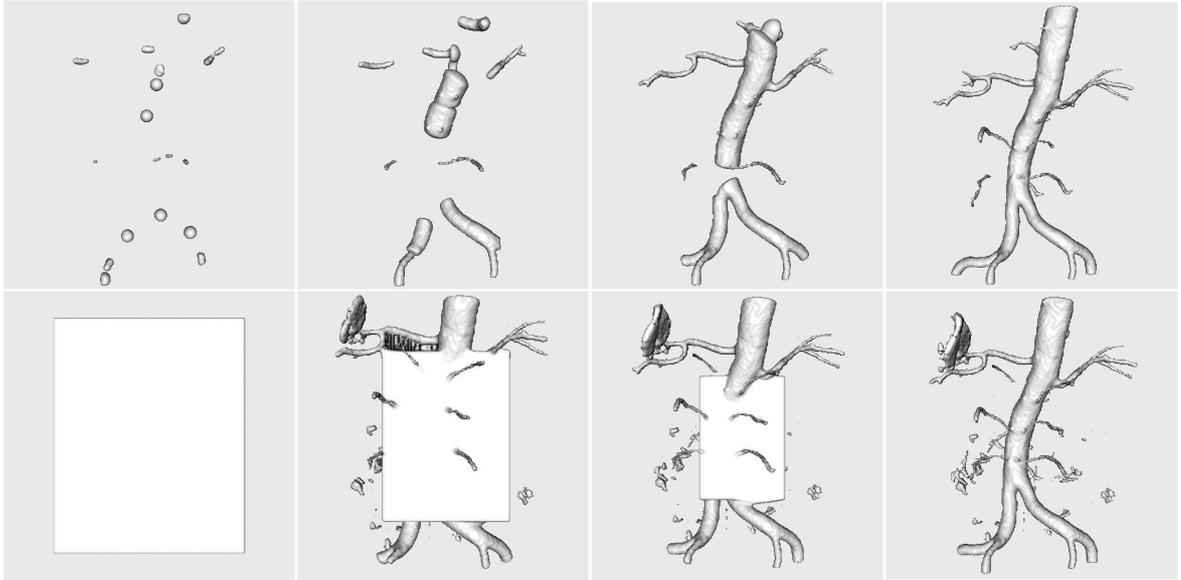


Figure 4.11: Segmentation of blood vessels from MRA, starting from two different initializations: 20 seed points (top row) and a bounding box (bottom row).

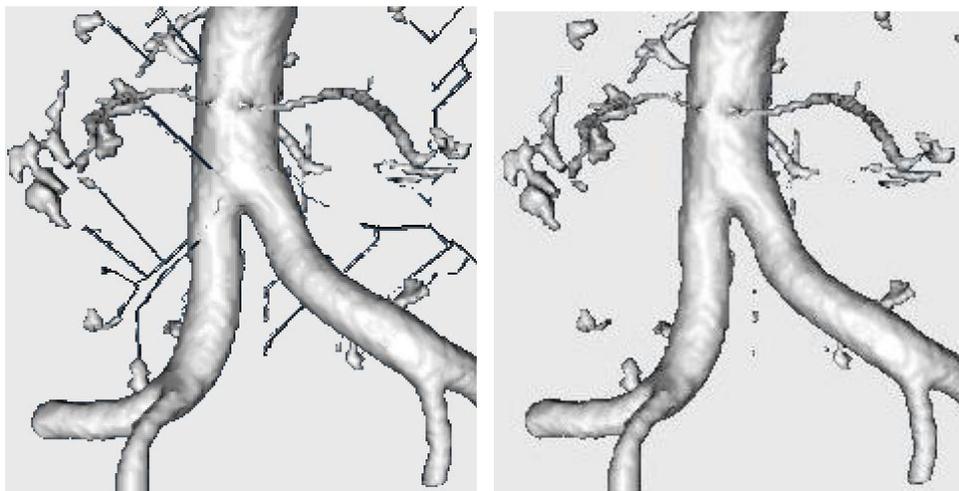


Figure 4.12: Comparison of the results of the topology preserving level set method of [64] (left) and of the genus preserving level set method (right) for segmenting blood vessels from MRA starting from a bounding box.

4.5 Contributions of this chapter

In this chapter, we have presented two novel methods to exert a control on the topology during a level set evolution. Our work significantly improves the topology preserving level set method of [64]. Our first extension allows to concurrently evolve several nested interfaces in the level set framework while preventing topology changes as well as mutual intersections. Our method outputs several mutual intersection free meshes with a certified topology.

Our second extension realizes a trade-off in between standard level sets and topology preserving level sets. Compared to the former, the formation of new handles and the closing of existing handles are prevented. Compared to the latter, the ability to change topology under certain conditions greatly alleviates the sensitivity to initial conditions.

Below, we list the contributions of this chapter:

- We have introduced a new criterion for the preservation of the topology and the prevention of mutual intersections of several nested digital objects.
- We have described an algorithm to evolve several nested level sets, based on the above criterion, and we have checked its consistency with the topology-wise marching cubes algorithm.
- We have introduced a new concept of digital topology called multisimple point, and we have characterized multisimple points with the topological numbers and some extended topological numbers. When dealing only with spherical components, the most common situation in practice, the multisimple point criterion can be checked efficiently.
- We have described an algorithm for genus preserving level sets, based on the multisimple point criterion.
- In some numerical experiments on synthetic data and on real MR images, we have demonstrated the advantages of the genus preserving level set method over the standard level set method and the topology preserving level set method.

The content of this chapter is a joint work with Florent Ségonne, from the CSAIL (Computer Science and Artificial Intelligence Laboratory) at the MIT (Massachusetts Institute of Technology) and the Athinoula Martinos Center for Biomedical Imaging, at the MGH (Massachusetts General Hospital), Boston, USA.

The extension of the topology preserving level set method to several nested interfaces is by Jean-Philippe Pons. As regards the genus preserving level set method, the concept of multisimple point was initially proposed by Florent Ségonne, with the disadvantage of being asymmetric with respect to foreground and background. The symmetric version presented in this thesis was suggested by Jean-Philippe Pons. The level set aspects of

the implementation are by Jean-Philippe Pons, while the digital topology aspects are by Florent Ségonne.

Part of this work has previously appeared in a technical report from the CSAIL [138], and was presented at a workshop of *The International Conference on Computer Vision* [139].

Chapter 5

Improving the Robustness to Local Minima with Spatially Coherent Minimizing Flows

This chapter tackles an important aspect of variational deformable models, which has been largely overlooked so far: the optimization by gradient flows. Classically, the definition of a gradient depends directly on the choice of an inner product structure. This consideration is largely absent from the deformable models literature. Most authors, overtly or covertly, assume that the space of admissible deformations is ruled by the canonical L^2 inner product. The classical geometric gradient flows reported in the literature are relative to this particular choice.

In this chapter, we investigate the relevance of using other inner products, yielding other gradient descents, and some other geometric minimizing flows not deriving from any inner product. In particular, we show how to induce different degrees of spatial coherence into the minimizing flow, in order to decrease the probability of getting trapped into irrelevant local minima. We show with some numerical experiments that the sensitivity of the deformable models framework to initial conditions, which seriously limits its applicability and its efficiency, is alleviated by our application-specific spatially coherent minimizing flows.

5.1 Motivation

In the following we note Γ a deformable model in \mathbb{R}^n and $E(\Gamma)$ the energy functional to be minimized. In order to define the gradient of the energy functional, the first step is to compute its Gâteaux derivatives in all directions, i.e. for all admissible velocity fields v :

$$\delta E(\Gamma, v) \stackrel{def}{=} \lim_{\epsilon \rightarrow 0} \frac{E(\Gamma + \epsilon v) - E(\Gamma)}{\epsilon}. \quad (5.1)$$

Then, we would like to pick the gradient as the direction of steepest descent of the energy. However, it is not yet possible at this stage: to be able to assess the steepness of

the energy, the deformation space needs additional structure, namely an inner product introducing the geometrical notions of angles and lengths. This consideration is largely absent from the deformable models literature: most authors, overtly or covertly, assume that the deformation space is ruled by the canonical L^2 inner product on Γ :

$$\langle u, v \rangle_{L^2} = \int_{\Gamma} u(\mathbf{x}) \cdot v(\mathbf{x}) d\mathbf{x} ,$$

where $d\mathbf{x}$ is the infinitesimal area element (the length in 2D, the area in 3D and so on).

Here, for sake of generality, we model the space of admissible deformations as an inner product space $(F, \langle \cdot, \cdot \rangle_F)$. If there exists a vector $u \in F$ such that

$$\forall v \in F, \delta E(\Gamma, v) = \langle u, v \rangle_F ,$$

then u is unique, we call it the gradient of E relative to the inner product $\langle \cdot, \cdot \rangle_F$, and we note $u = \nabla_F E(\Gamma)$. Clearly, each choice of inner product yields its own gradient. This is often neglected and most authors improperly refer to *the* gradient of the energy. Thus, the classical geometric gradient flows reported in the literature (mean curvature flow [54], geodesic active contours [24, 58, 150], etc.) are relative to the L^2 inner product.

A slightly different definition of the gradient, based on a representation of the space of admissible shapes as a differential manifold, is proposed in [150]. However, this definition requires the shapes and the deformations to be smooth, the energy functional to be differentiable, and the deformation space to be a separable Hilbert space. Our definition is more general since it only demands the existence of the directional derivatives.

The gradient descent method consists in deforming an initial shape Γ_0 in the opposite direction of the gradient.

$$\begin{cases} \Gamma(0) = \Gamma_0 \\ \frac{d\Gamma}{dt} = -\nabla_F E(\Gamma) \end{cases} \quad (5.2)$$

The problem of the existence and the uniqueness of this minimizing flow is out of the scope of this article. Indeed, it is highly dependent on the properties of each particular energy functional. If this evolution exists, it decreases the energy:

$$\frac{dE(\Gamma)}{dt} = -\|\nabla_F E(\Gamma)\|_F^2 \leq 0 .$$

The standard choice for F is the Hilbert space of square integrable velocity fields $L^2(\Gamma, \mathbb{R}^n)$ equipped with its canonical inner product. Very few authors in the deformable models area have considered using other inner products, whereas this is an established technique in image registration [163]. Very recently, in the context of shape representation and analysis, [102, 180] have shown that slightly modifying the L^2 inner product allows to build well-behaved metrics in the space of curves.

Minimizing flows not deriving from any inner product, that is to say evolutions that decrease the energy, without any gradient interpretation, have also been overlooked so far. Note that any evolution fulfilling the condition

$$\frac{dE(\Gamma)}{dt} = \left\langle \nabla_F E(\Gamma), \frac{d\Gamma}{dt} \right\rangle_F \leq 0 \quad (5.3)$$

is a candidate to solve the minimization problem. This idea, proposed in [150], is applied by the same authors to the alignment of curves in images in [113]: a complicated term in the gradient is safely neglected after checking that the evolution still decreases the energy.

The spirit of our work is different. We do not focus either on a specific inner product or on a particular energy functional. We rather explore general procedures to build some new inner products and to compute the associated gradients. We also address the design of non-gradient minimizing flows.

Our motivation is also different. Our primary concern in this work is the sensitivity of the deformable models framework to initial conditions. There are essentially two ways of dealing with this problem: positioning the initial shape very close to the expected final configuration, or using a multiresolution coarse-to-fine strategy, in other words running the optimization on a series of smoothed and subsampled shapes and input data. In this chapter, we pioneer a third way to tackle the problem of unwanted local minima: the careful design of the minimizing flow.

We do not modify the energy, hence the relief of the energy landscape and in particular the “number” of local minima remains unchanged. But by using an evolution that favors certain types of directions, we expect to decrease the probability of falling into unwanted energy basins.

Typically, in many applications, spatially coherent motions are to be preferred over erratic evolutions. For example, in the tracking problem, the object of interest is likely to have similar shapes in consecutive frames. So if we init the contour with the result of the previous frame, it makes sense to encourage the motions which preserve its overall appearance. This way, it may be easier to dodge unexpected local low-energy configurations. A traditional L^2 gradient descent definitely does not have this desirable property since the L^2 inner product completely disregards the spatial coherence of the velocity field.

The rest of this chapter is organized as follows. In Section 5.2, we carry out an abstract study of gradient and non-gradient minimizing flows. In Section 5.3, we propose some particular flows that yield different degrees of spatial coherence. Finally, in Section 5.4, we show in some numerical experiments that the robustness of the deformable models framework to local minima is improved by our application-specific spatially coherent minimizing flows.

5.2 Abstract study

5.2.1 Designing new inner products

We suppose that the space F of admissible deformations is initially equipped with the inner product \langle, \rangle_F . Then, for any symmetric positive definite linear operator $L : F \rightarrow F$, a new inner product can be defined by

$$\langle u, v \rangle_L = \langle Lu, v \rangle_F . \tag{5.4}$$

Here, for simplicity, we assume that the domain and the range of L are equal to F . A similar study is possible if they are strictly smaller than F , under certain conditions, using the Friedrichs extension of L [4]. But these technical details are out of the scope of this work.

The following observation is central to our work: if $\nabla_F E(\Gamma)$ exists and if L is invertible, then $\nabla_L E(\Gamma)$ also exists and we have

$$\nabla_L E(\Gamma) = L^{-1} \nabla_F E(\Gamma) . \quad (5.5)$$

Indeed:

$$\begin{aligned} \forall v \in F, \delta E(\Gamma, v) &= \langle \nabla_F E(\Gamma), v \rangle_F \\ &= \langle LL^{-1} \nabla_F E(\Gamma), v \rangle_F \\ &= \langle L^{-1} \nabla_F E(\Gamma), v \rangle_L . \end{aligned}$$

The above procedure is of great practical interest because it allows to upgrade any existing L^2 gradient flow. However, it is not completely general in the sense that all inner products cannot be expressed in this form. This construction is illustrated in Subsections 5.3.1 and 5.3.2 by some particular inner products yielding spatially coherent gradient flows.

This said, if F is a separable Hilbert space (i.e. a separable inner product space complete with respect to the norm $\|\cdot\|_F$), the Riesz representation theorem tells us that any inner product $\langle \cdot, \cdot \rangle_L$ such that

$$\exists C > 0, \forall u \in F, \|u\|_L < C \|u\|_F$$

can be written in the form of (5.4). This suggests that our procedure accounts for a wide range of inner products.

5.2.2 Designing new minimizing flows

In this subsection, we follow the inverse approach. Instead of working on the inner product, we apply a linear operator $L : F \rightarrow F$ to the gradient, and we study the properties of the resulting flow:

$$\frac{d\Gamma}{dt} = -L \nabla_F E(\Gamma) . \quad (5.6)$$

This naturally sets up a hierarchy among different types of operators:

- if L is positive, the energy is non-increasing along the flow (5.6). Indeed,

$$\frac{dE(\Gamma)}{dt} = - \langle \nabla_F E(\Gamma), L \nabla_F E(\Gamma) \rangle_F \leq 0 .$$

- if L is positive definite, the energy strictly decreases along the flow (5.6) until a critical point of the original gradient flow (5.2) is reached.
- if L is symmetric positive definite and invertible, the flow (5.6) coincides with a gradient descent relative to the inner product $\langle \cdot, \cdot \rangle_{L^{-1}}$, as defined in 5.4).

The third case is contained in Subsection 5.2.1. The second case is illustrated in Subsection 5.3.3 by a Gaussian smoothing of the gradient along the deformable model, in order to generate a smoother minimizing flow.

5.3 Spatially coherent minimizing flows

In this section, we propose some minimizing flows yielding different degrees of spatial coherence. We insist on the fact that this spatial coherence has nothing to do with an eventual regularity term in the energy functional. We do not modify the energy, so the regularity constraint on the deformable model remains unchanged. We modify the trajectory of the minimizing flow, by favoring spatially coherent motions, but this does not condition the regularity of the final shape.

In the following, we make an intense use of differential geometry. We refer the reader to [43] for the basic notions.

5.3.1 The H^1 gradient flow

A first way to introduce a notion of spatial coherence is to use an inner product that penalizes not only the length of the velocity field, but also its variations along the deformable model. To this end, we consider the canonical inner product of the Sobolev space $H^1(\Gamma, \mathbb{R}^n)$ of square integrable velocity fields with square integrable derivatives:

$$\langle u, v \rangle_{H^1} = \int_{\Gamma} u(\mathbf{x}) \cdot v(\mathbf{x}) d\mathbf{x} + \int_{\Gamma} \nabla_{\Gamma} u(\mathbf{x}) \cdot \nabla_{\Gamma} v(\mathbf{x}) d\mathbf{x},$$

where ∇_{Γ} denotes the intrinsic gradient on Γ . The H^1 inner product is related to the L^2 inner product by (5.4) with $L(u) = u - \Delta_{\Gamma} u$, where Δ_{Γ} denotes the intrinsic Laplacian operator on Γ , often called the Laplace-Beltrami operator.

As a result, $\nabla_{H^1} E$ is a smoothed version of $\nabla_{L^2} E$ which can be obtained either by solving an intrinsic heat equation with a data attachment term:

$$\Delta_{\Gamma} u = u - \nabla_{L^2} E, \tag{5.7}$$

or by finding the optimum of:

$$\arg \min_u \int_{\Gamma} |\nabla_{\Gamma} u(\mathbf{x})|^2 d\mathbf{x} + \int_{\Gamma} |u(\mathbf{x}) - \nabla_{L^2} E(\mathbf{x})|^2 d\mathbf{x}.$$

To sum up, using the H^1 inner product instead of the L^2 inner product leads to a smoother gradient flow.

5.3.2 Motion decomposition

Another simple and useful procedure to design new inner products yielding spatially coherent flows, is to decompose the deformation space into a sum of several mutually orthogonal linear subspaces, and to apply different penalty factors to the different types of motions.

Typically, the subspaces are chosen according to an application-specific hierarchy of the motions. For example, rigid/non-rigid, affine/non-affine, etc.

We suppose that such an orthogonal (with respect to \langle, \rangle_F) decomposition of the deformation space F into N closed linear subspaces is available:

$$F = F_1 \perp \cdots \perp F_N .$$

Then a new inner product is derived from \langle, \rangle_F by applying the procedure of Subsection 5.2.1 with

$$L = \bigoplus_{i=1}^N \lambda_i \text{Id}_{F_i} ,$$

where $\forall i, \lambda_i > 0$. The lower is λ_i , the shorter is the length of the velocity fields of F_i , and the stronger will be this type of motion in the new gradient flow.

Obviously, L is symmetric positive definite and invertible. If $\nabla_F E$ exists, so does $\nabla_L E$ and

$$\nabla_L E = \sum_{i=1}^N \frac{1}{\lambda_i} \Pi_{F_i} \nabla_F E , \quad (5.8)$$

where Π_{F_i} denotes the orthogonal projection on the i^{th} subspace. Of course, if all λ_i are equal to 1, $\nabla_L E$ coincides with $\nabla_F E$.

We apply this idea to two useful cases. In the first case, we decompose the velocity field into a translation, an instantaneous rotation, a rescaling motion and a non-rigid residual. In the second case, we isolate the instantaneous affine motion.

In the following, we note $M(\Gamma) = \int_{\Gamma} d\mathbf{x}$ the mass of the deformable model (its length in 2D, its area in 3D and so on) and $\bar{f} = (\int_{\Gamma} f(\mathbf{x}) d\mathbf{x}) / M(\Gamma)$ the average of a scalar or vector quantity on Γ . With this notation in hand, the centroid of the deformable model writes $\bar{\mathbf{x}}$.

Translation, rotation and scaling

In this paragraph, we focus on the two-dimensional and three-dimensional cases. The expressions below are for the 3D case, but can easily be adapted to 2D.

We note T , R and S the subspaces of the translations, the instantaneous rotations around the centroid, and the scaling motions centered on the centroid, respectively:

$$\begin{aligned} T &= \{u : \mathbf{x} \mapsto \mathbf{t} \mid \mathbf{t} \in \mathbb{R}^3\} , \\ R &= \{u : \mathbf{x} \mapsto (\mathbf{x} - \bar{\mathbf{x}}) \wedge \omega \mid \omega \in \mathbb{R}^3\} , \\ S &= \{u : \mathbf{x} \mapsto s(\mathbf{x} - \bar{\mathbf{x}}) \mid s \in \mathbb{R}\} . \end{aligned}$$

These subspaces are mutually orthogonal for the L^2 inner product. We suppose that they are included in the space of admissible deformations F , and that the latter is ruled by the L^2 inner product. We note G the orthogonal complement of these subspaces:

$F = T \perp R \perp S \perp G$. The orthogonal projection of a velocity field u on T , R and S writes:

$$\begin{aligned}\Pi_T u(\mathbf{x}) &= \bar{u}, \\ \Pi_R u(\mathbf{x}) &= (\mathbf{x} - \bar{\mathbf{x}}) \wedge \frac{\int_{\Gamma} u(\mathbf{x}) \wedge (\mathbf{x} - \bar{\mathbf{x}})}{\int_{\Gamma} \|\mathbf{x} - \bar{\mathbf{x}}\|^2}, \\ \Pi_S u(\mathbf{x}) &= \frac{\int_{\Gamma} u(\mathbf{x}) \cdot (\mathbf{x} - \bar{\mathbf{x}})}{\int_{\Gamma} \|\mathbf{x} - \bar{\mathbf{x}}\|^2} (\mathbf{x} - \bar{\mathbf{x}}).\end{aligned}$$

The new gradient is deduced from the L^2 gradient by (5.5) with

$$L^{-1} = \text{Id} + \left(\frac{1}{\lambda_T} - 1\right) \Pi_T + \left(\frac{1}{\lambda_R} - 1\right) \Pi_R + \left(\frac{1}{\lambda_S} - 1\right) \Pi_S.$$

The weights λ_T , λ_R and λ_S are adapted to the user's needs in each particular application. For example:

- Boost rigid+scaling motions: $\lambda_T, \lambda_R, \lambda_S \ll 1$,
- Boost rigid motions: $\lambda_T, \lambda_R \ll 1$, $\lambda_S = 1$,
- Boost translations: $\lambda_T \ll 1$, $\lambda_R = \lambda_S = 1$.

Affine motion

We can apply this same idea to the subspace A of instantaneous affine motions:

$$A = \{u : \mathbf{x} \mapsto \mathbf{A}\mathbf{x} + \mathbf{b} \mid \mathbf{A} \in \mathbb{R}^{n \times n}, \mathbf{b} \in \mathbb{R}^n\}.$$

The L^2 orthogonal projection on this subspace writes:

$$\Pi_A u(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b},$$

where

$$\begin{aligned}\mathbf{A} &= \left[\int_{\Gamma} u(\mathbf{x})(\mathbf{x} - \bar{\mathbf{x}})^T \right] \left[\int_{\Gamma} (\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^T \right]^{-1}, \\ \mathbf{b} &= \bar{u} - \mathbf{A}\bar{\mathbf{x}}.\end{aligned}$$

5.3.3 Intrinsic Gaussian smoothing

We apply the procedure of Subsection 5.2.2 to design a smoothed version of the L^2 gradient flow. To some extent, it resembles the H^1 gradient flow of Subsection 5.3.1. However, here, we apply an *ad hoc* procedure to the L^2 gradient, a Gaussian smoothing along the deformable model, without resorting to an inner product.

We define a linear intrinsic smoothing operator which may be seen as the counterpart on Γ of Gaussian smoothing in \mathbb{R}^{n-1} , by considering the solution \tilde{u} of the intrinsic heat equation on Γ with initial condition u :

$$\begin{cases} \tilde{u}(\cdot, 0) = u \\ \frac{\partial \tilde{u}}{\partial \tau} = \Delta_{\Gamma} \tilde{u} \end{cases} . \quad (5.9)$$

We then note $L_\tau u$ its solution $\tilde{u}(\cdot, \tau)$ at time $\tau \geq 0$. We use a flow (5.6) based on L_τ to drive the deformable model. In other words, to get the new flow, we diffuse the L^2 gradient isotropically along Γ during a time τ . The larger is τ , the smoother is the flow.

On the one hand, L_τ is symmetric positive. In particular, the new flow decreases the energy.

- L_τ is symmetric:

$$\begin{aligned} \langle L_0(u), v \rangle_{L^2} &= \langle u, L_0(v) \rangle_{L^2} = \langle u, v \rangle_{L^2} , \\ \frac{\partial}{\partial \tau} \langle L_\tau(u), v \rangle_{L^2} &= \frac{\partial}{\partial \tau} \langle u, L_\tau(v) \rangle_{L^2} = - \langle \nabla_\Gamma u, \nabla_\Gamma v \rangle_{L^2} . \end{aligned}$$

- L_τ is positive:

$$\langle L_\tau(u), u \rangle_{L^2} = \langle L_{\tau/2} L_{\tau/2}(u), u \rangle_{L^2} = \|L_{\tau/2}(u)\|_{L^2}^2 \geq 0 .$$

But the other hand, the inversion of L_τ for $\tau > 0$ is an ill-posed anti-diffusive process. So a gradient interpretation is not available.

5.4 Experimental results

Our approach can be applied to virtually any deformable model evolution. Below, we show some particular applications which demonstrate its interest.

The content of this work is not specific to a particular geometric representation of the deformable model. In our experiments, we use the level set method (*cf* Subsection 2.1.2), motivated by its numerical stability and its ability to handle topology changes automatically. The implicit representation also offers an elegant expression of the Laplace-Beltrami operator [13] (*cf* Appendix B.3), and of the average of a quantity along the deformable model [118] (*cf* Appendix A.2).

The additional computational cost of our approach depends on the type of minimizing flow we consider. The extra time is barely noticeable for the rigid plus scaling and affine flows of paragraphs 5.3.2 and 5.3.2. The latter only require to compute a handful of integrals on the deformable model. The smooth minimizing flows of Subsections 5.3.1 and 5.3.3 are more demanding. In 2D, the implicit diffusion equations (5.7) and (5.9) are equivalent to some convolutions with respect to the curvilinear coordinate on Γ . In 3D and more, they must be solved with some iterative methods, for each time step.

5.4.1 Shape warping

We illustrate our approach in the problem of shape warping. In this context, the energy functional to be minimized is a measure of dissimilarity between the evolving shape and a target shape. The study of shape metrics is still an active research area [183, 181, 26, 180], and there are many candidates for the dissimilarity measure. In our experiments, we use a differentiable approximation of the well-known Hausdorff distance, as proposed in [26], to warp the contours of two different hands.

Figure 5.1 compares the evolution of the deformable model when using the L^2 gradient descent (top row) and a modified gradient descent favoring rigid plus scaling motions (bottom row) as in paragraph 5.3.2. Both evolutions achieve a perfect warping. However, despite the similarity of the two input shapes, the L^2 gradient flow goes through some states of completely different appearances. The trajectory followed by this flow looks particularly inefficient and unnatural, because the notion of length contained in the L^2 inner product is very far from our intuition. In contrast, the behavior of our gradient flow is natural and visually pleasing.

In Figure 5.2, we show a three-dimensional warping example from a Teddy bear to Hayao Miyazaki’s character Totoro. We use here the $W^{1,2}$ -norm of the difference of the distance functions as proposed in [26]. Once again, a modified gradient descent favoring rigid plus scaling motions yields better results than the L^2 gradient descent.

This suggests that our approach can infer relevant correspondences between the two shapes, as a byproduct of the warping process. This point-to-point matching is obtained by tracking the points along the evolution. It does not make much sense with a L^2 gradient flow, because the latter yields a strictly normal velocity field. But when using our approach, the velocity field has a meaningful tangential part. Maintaining the point correspondence during the evolution is straightforward with an explicit geometric representation. It is also feasible with a level set representation, with the extension proposed in Chapter 3.

5.4.2 Tracking

We now illustrate the better robustness of spatially coherent minimizing flows to local minima, in the problem of tracking an object in a monocular video sequence. We have used the boundary-based energy of the geodesic active contours method [24], to track a moving hand. Note that a region-based segmentation method, as the ones described in Subsection 2.3.1, could give better results on our particular test sequence. However, our concern here are not the results themselves but rather the improvements brought by our approach.

Figure 5.3 compares the evolution of the deformable model when using the L^2 gradient descent (top row) and a modified gradient descent favoring affine motions (bottom row) as in paragraph 5.3.2. Due to large displacements between consecutive frames, the L^2 gradient flow fails and the deformable model finally locks between two fingers, whereas our gradient flow manages to dodge this unwanted low-energy configuration.

5.5 Contributions of this chapter

The contributions of this chapter span three different levels:

- At the conceptual level, we have highlighted the impact of the inner product structure of the deformation space on the behavior of the deformable models framework.

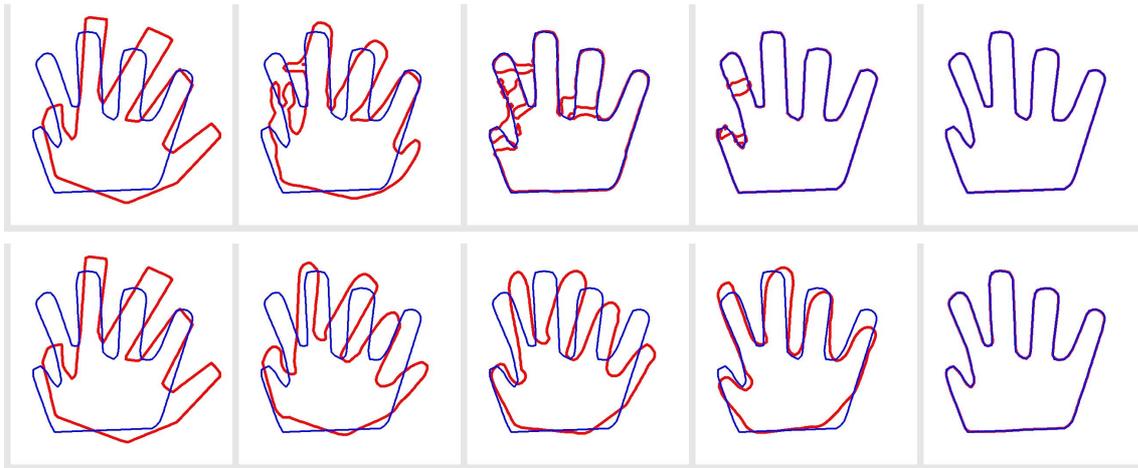


Figure 5.1: Shape warping with the L^2 gradient descent (top) and with a modified gradient descent favoring rigid plus scaling motions (bottom): $\lambda_T = \lambda_R = \lambda_S = 0.025$.

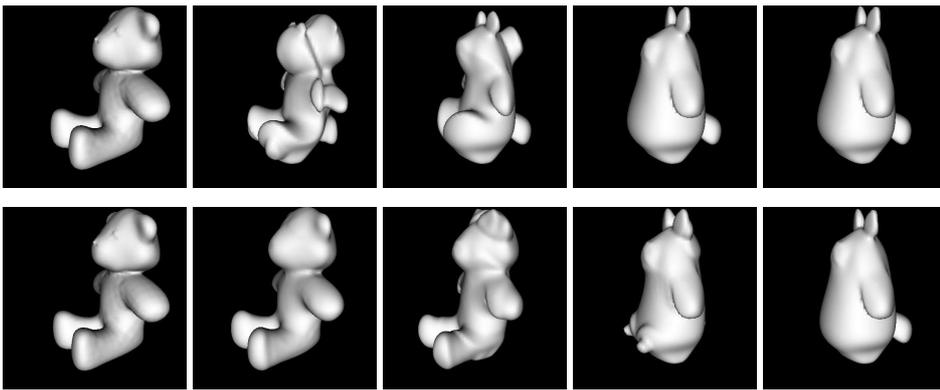


Figure 5.2: 3D shape warping with the L^2 gradient descent (top) and with a modified gradient descent favoring rigid plus scaling motions (bottom): $\lambda_T = \lambda_R = \lambda_S = 0.025$.

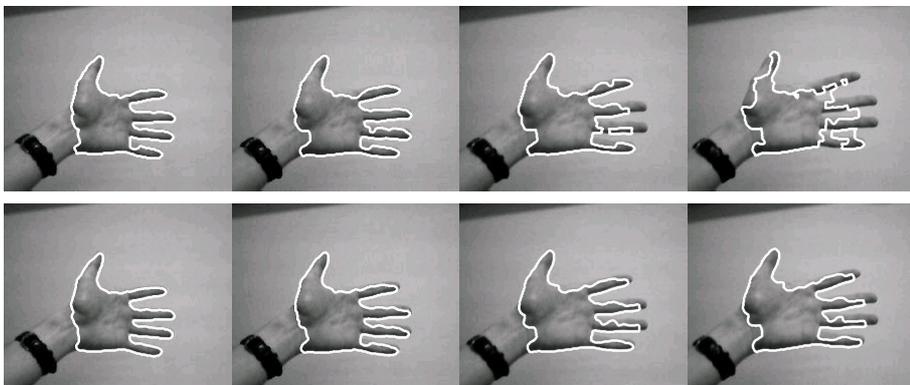


Figure 5.3: Tracking a hand in a video sequence with the L^2 gradient descent (top) and with a modified gradient descent favoring affine motions (bottom): $\lambda_A = 0.025$.

- At the methodological level, we have proposed some general procedures to build several families of inner products as well as some minimizing flows not deriving from any inner product.
- At the application level, we have proposed some particular minimizing flows which introduce different degrees of spatial coherence in the evolution. We have shown in some numerical experiments that these evolutions, as they better fit our intuitive notion of deformation cost, and as they mimic the behavior of the objects of interest, are at the same time more pleasing visually and more robust to local minima.

The content of this chapter is a joint work with Guillaume Charpiat, from the ENS (École Normale Supérieure), Paris, France. The initial idea and the abstract study presented in this thesis were done concurrently by Guillaume Charpiat and Jean-Philippe Pons. The motion decomposition approach is by Jean-Philippe Pons. The implementation and experiments are by Guillaume Charpiat and the redaction is by Jean-Philippe Pons. This work has been presented at the *International Conference on Computer Vision* [27].

Part II

Applications

Chapter 6

Area Preserving Cortex Unfolding

In this chapter, we propose a novel deformable models method to generate unfolded area preserving representations of the cerebral cortex. The cortical surface is evolved with an application-specific normal motion, and an adequate tangential motion is constructed in order to ensure an exact area preservation throughout the evolution. We describe the continuous formulation of our method as well as its numerical implementation with triangulated surfaces and level sets. A straightforward level set implementation of cortex unfolding is not feasible due to the absence of parameterization. The use of the level set representation for this application is made possible by the contribution of Chapter 3 for maintaining a point correspondence during the evolution. We show the applicability of our approach by computing inflated representations of the cortex from real brain data.

6.1 Motivation

Building unfolded representations of the cerebral cortex has become an important area of research in medical imaging. On a simplified geometry, it becomes easier to visualize and analyze functional or structural properties of the cortex, particularly in previously hidden sulcal regions. Moreover, if some metric properties of the cortical surface can be preserved while eliminating folds, it makes sense to map the data from different subjects in a canonical space for building brain atlases [141].

Three types of unfolded representations have been proposed in the literature: inflated, that is to say a smoothed version of the cortex retaining its overall shape, spherical and flattened (see [52] and references therein). Three metric properties have been considered: geodesic distances, angles and areas. Unfortunately, preserving distances exactly is impossible because the cortical surface and its simplified version will have different Gaussian curvature [43]. Moreover, it is not possible to preserve angles and areas simultaneously.

As a consequence, early methods for cortex unfolding have settled for variational approaches, leading to a variety of local forces encouraging an approximate preservation of area and angle while smoothing the surface [36, 22]. In [52], the authors point out that the result of such methods are not optimal with respect to any metric property, and work

out a method which focuses on distances only and minimizes their distortion.

A lot of work has focused on building conformal representations of the cortical surface, i.e. one-to-one, onto, and angle preserving mappings between the cortex and a target surface, often a sphere or a plane (see [61, 7] and references therein). These approaches use a well known fact from Riemannian geometry that a surface without any handles, holes or self-intersections can be mapped conformally onto the sphere, and any local portion thereof onto a disk.

In our work we focus on area preservation, motivated by a general result which ensures the existence of an area preserving mapping between two diffeomorphic surfaces of the same total area [107]. A method for building such an area preserving mapping for the visualization of medical structures has been proposed in [189, 8]. In addition, this method allows to pick, among all existing area preserving mappings, a local minimum of metric distortion. This minimum is in some sense the nearest area preserving map to a conformal one. This mathematical formulation is very promising but few numerical experiments are presented in the paper. Moreover, the numerical implementation of their method is feasible only for planar or spherical maps.

We propose a method to evolve the cortical surface while preserving local area, and we use it to build area preserving inflated representations of the cortex. Our method relates both to [189, 8] and [36, 22, 106]. We evolve a surface in time as in [36, 22, 106], whereas the method of [189, 8] is static and generates a single mapping between the cortex and a sphere. We achieve an exact area preservation as in [189, 8], whereas in [36, 22, 106] area preservation is only encouraged by tangential forces. Furthermore, the latter approaches only have a discrete formulation and are specific to one type of deformable model, whereas ours is continuous and can be applied numerically both to triangulated surfaces and level sets.

To achieve this, we treat the normal and the tangential components of the motion differently. On the one hand, the normal motion controls the geometry of the surface. It is application-specific and is chosen by the user. For example, it can be a mean curvature motion in order to obtain a multiresolution representation of the cortex, or it can be a force pulling the surface towards a target shape, typically a sphere. On the other hand, given the selected normal motion, an adequate tangential motion is constructed in order to ensure an exact area preservation throughout the evolution.

The rest of the chapter is organized as follows. In Section 6.2, we present a general method for building an area preserving motion for a codimension one manifold in any number of dimensions. Given a desired normal motion, we compute a non-trivial tangential motion such that the evolution is area preserving. In Section 6.3, we demonstrate the applicability of our method by computing area preserving inflated representations of the cortex from real brain data.

6.2 Area preserving surface motion

In this section we first explicitly state the mathematical condition for the preservation of the local area of an evolving codimension one manifold in any number of dimensions. This condition ensures that the total area, as well as the area of any patch, is preserved. When the total area needs not to be preserved, we can nonetheless preserve the relative local area, i.e. the ratio between the area of any patch and the total area. We then derive a procedure to build an area preserving or relative area preserving tangential motion. Finally we describe the numerical implementation of our method with two major types of deformable models: triangulated surfaces and level sets. In the following, we make an intensive use of differential geometry. We refer the reader to [43, 38] for the basic theory. Note that, contrary to conformal approaches, our formulation is not limited to a genus zero surface and applies to any topology.

6.2.1 The total, local and relative area preserving conditions

Let us consider an hypersurface Γ in \mathbb{R}^n evolving with a velocity field \mathbf{v} . We define the local areal factor J at a point of Γ as the ratio between the initial area and the current area of an infinitesimal patch around this point. So an increasing J indicates a local shrinkage and a decreasing one indicates a local expansion. The preservation of total area, local area, and relative area respectively write

$$\bar{J} = 1 \quad (6.1a) \quad ; \quad J = 1 \quad (6.1b) \quad ; \quad J = \bar{J} \quad (6.1c) \quad , \quad (6.1)$$

where $\bar{\cdot}$ is the average of a quantity along Γ . \bar{J} is related to the variation of the total area through $A_0 = A \bar{J}$. The local areal factor J of a material point evolves according to

$$\frac{DJ}{Dt} + J \operatorname{div}_\Gamma \mathbf{v} = 0 \quad , \quad (6.2)$$

where D/Dt denotes the material derivative, and $\operatorname{div}_\Gamma$ denotes the intrinsic divergence operator on Γ . As a consequence, the condition to be verified by the velocity field for the preservation of total area, local area and relative area are respectively

$$\overline{\operatorname{div}_\Gamma \mathbf{v}} = 0 \quad (6.3a) \quad ; \quad \operatorname{div}_\Gamma \mathbf{v} = 0 \quad (6.3b) \quad ; \quad \operatorname{div}_\Gamma \mathbf{v} = \overline{\operatorname{div}_\Gamma \mathbf{v}} \quad (6.3c) \quad . \quad (6.3)$$

Note that the right-hand side of (6.3c) is spatially constant but is time-varying. Also, the preservation of local area is the combination of the preservation of total and relative area, so in the sequel we focus on (6.3a,c) only.

If we decompose \mathbf{v} into its outward normal component v_N and its tangential part \mathbf{v}_T , (6.3a,c) become

$$\overline{H v_N} = 0 \quad (6.4a) \quad ; \quad \operatorname{div}_\Gamma \mathbf{v}_T + (n-1) H v_N = (n-1) \overline{H v_N} \quad (6.4b) \quad , \quad (6.4)$$

where \mathbf{N} is the outward normal and H is the mean curvature. In the left-hand side of (6.4b), we now better see the two different sources of local expansion/shrinkage: one is tangential motion, the other is the association of normal motion and curvature.

For a plane curve C embedded in \mathbb{R}^2 deforming with velocity $\mathbf{v} = \alpha\mathbf{T} + \beta\mathbf{N}$, the relative area preservation condition boils down to $\frac{\partial\alpha}{\partial s} + \kappa\beta = \overline{\kappa\beta}$, where κ denotes the curvature of C . This particular case has been used (with the inward normal convention) in [103] in the Lagrangian curve evolution framework: the authors build a tangential velocity that achieves a uniform redistribution of grid points along the curve, in order to overcome numerical instabilities caused by merging of grid points or by formation of the so-called swallow tails.

In [106], the authors design tangential forces which tend to equalize the surface of the faces of an evolving 3D simplex mesh, in order to maintain its geometric quality. Their approach is successful but, in contrast with our work, it only has a discrete formulation and cannot be generalized to other types of deformable models.

6.2.2 Designing a relative area preserving tangential motion

We now outline our method to build a relative area preserving velocity field. We are given a normal speed v_N . Let us consider the solution η of the following intrinsic Poisson equation on Γ :

$$\Delta_\Gamma \eta = (n-1) (H v_N - \overline{H v_N}) \quad , \quad (6.5)$$

where Δ_Γ denotes the Laplace-Beltrami operator on Γ . Finding a solution of (6.5) is possible because the right-hand side is of zero average [128]. Moreover, the solution η is unique up to a constant. Then one can check that

$$\mathbf{v} = v_N \mathbf{N} - \nabla_\Gamma \eta \quad (6.6)$$

verifies (6.4). Note that the normal motion is not altered since $\nabla_\Gamma \eta$, the intrinsic gradient of η , is purely tangential, and that the resulting velocity field is non local: it depends on the whole shape and motion of the interface.

For a given normal speed, there are in general an infinity of admissible area preserving tangential velocity fields. The particular solution derived from (6.5) is a reasonable choice since our method outputs a null tangential motion if the normal motion is already area preserving.

In general, the given normal speed v_N does not preserve total area, and our method can only enforce a relative area preservation. If a strict area preservation is required, we can either apply an adequate rescaling at a post processing step, or integrate in the normal motion a rescaling motion $-\overline{H v_N} (\mathbf{x} \cdot \mathbf{N})\mathbf{N}$ so that the total area is preserved.

6.2.3 The area relaxation term

Numerical implementations of the above method may yield deceiving results. Indeed, in practice, (6.5) can only be solved up to a certain tolerance, and the surface evolution is subject to discretization errors. As a consequence, the area preserving condition cannot be fulfilled exactly and the area of interface patches may drift slowly from its expected values. Thus, area preservation is likely to be broken when the number of iterations increases.

We augment our method with an area relaxation term which encourages a uniform area redistribution whenever area preservation is lost. This feature is crucial to counteract local drifts in area due to numerical errors. We insist on the fact that this additional term is not a numerical heuristic: it is fully consistent with the mathematical formulation given above. Let us now seek the solution of

$$\Delta_{\Gamma} \eta = (n - 1) (H v_N - \overline{H v_N}) + \lambda(1 - \frac{J}{\bar{J}}) , \quad (6.7)$$

where λ is a weighting parameter. We build the velocity field from (6.6) as previously. Using (6.2) we get that

$$\frac{D(\bar{J}/J)}{Dt} = \lambda(1 - \bar{J}/J) . \quad (6.8)$$

Hence the ratio \bar{J}/J relaxes exponentially towards unity with time constant $1/\lambda$. In our problem we have $J = \bar{J} = 1$ for $t = 0$ so that the solution of (6.8) is $J = \bar{J}$ for all t as desired. This was already the case without the relaxation term, so at first sight this term is of no use. But in practice, numerical errors can make \bar{J}/J incidently deviate from unity. Thanks to the area relaxation term, it is now possible to recover from this numerical drift.

6.2.4 Numerical methods

For each time step we have to solve the intrinsic Poisson equation (6.7). Apparently it represents a huge computational cost. Hopefully, only the very first iteration is expensive. Indeed, for subsequent iterations, we use the solution η of time $t - 1$ as the initial guess for time t . If the shape and the normal motion change slowly relatively to the chosen time step, the guess is likely to be very close to the actual solution, and solving (6.7) is computationally cheap.

Triangulated surfaces

Solving a Poisson equation on a triangulated surface can be done with a finite element technique as proposed in [7]. Equation (6.7) then translates into a linear system with a sparse symmetric positive semi-definite matrix and can be solved efficiently with numerical iterative methods such as the conjugate gradient method [11].

Level sets

At first sight, a level set implementation of cortex unfolding is not feasible. Indeed, as discussed in Chapter 3, a level set representation conveys a purely geometric description of the interface. The tangential part of the velocity vanishes in the level set evolution equation. As a consequence, it is impossible to maintain a point correspondence or to handle interfacial data with the straightforward level set method. We use the LSPC method (Level Sets with a Point Correspondence) of Subsection 3.2.4 to make the level set method applicable to cortex unfolding.

In order to solve the Poisson equation in the implicit representation, we use a finite-difference discretization of the Laplace-Beltrami operator proposed in [13], and detailed in Appendix B.3. Equation (6.7) then translates into a linear system with a sparse symmetric indefinite matrix suited for a minimum residual method [11]. To compute the average of a quantity along the interface, we use a smoothed version of the Dirac function on the Cartesian grid as in [118] (*cf* Appendix A.2).

6.3 Experimental results

In this section, we focus on a level set representation of the deformable model and we demonstrate the strength of our approach in some numerical experiments on synthetic and real data.

6.3.1 Synthetic data

In Figure 6.1 we study the test case of a shrinking (left) or expanding (right) square with a null tangential velocity (top) and with a relative area preserving tangential velocity computed with our method (bottom). The graphical conventions are similar to the experiments of Section 3.4: the initial curve and the final curve are plotted with a dotted line and a dashed line respectively, and the correspondence from the final curve to the initial one is represented with arrows.

As discussed in Chapter 3, when using a null tangential velocity (top), a discontinuity in the correspondence develops in the presence of a shock and a many-to-one correspondence appears in the presence of a rarefaction (*cf* Section 3.4, Experiments 2&3 respectively). In contrast, when using a relative area preserving tangential velocity (bottom), the data are uniformly redistributed along the curve as expected.

In Figure 6.2 we study the case of a textured free-hand folded figure evolving under the averaged mean curvature flow, with (bottom) and without (top) an area preserving tangential velocity. This latter case illustrates that interfacial data may be considerably altered without an adequate tangential velocity. Indeed, note how the large interior part with high curvature of the initial curve (first column) turns into a much smaller patch on the final curve (columns 2&3, top row). The corresponding data are unacceptably distorted. We insist on the fact that this phenomenon is generated by the evolution itself. It is not an artefact of our method for maintaining the point correspondence in the level set framework.

We finish with a synthetic cortex unfolding example in Figure 6.3. We simulate a tumor on the left temporal gyrus of a subsampled human brain (left). The area of the tumor is considerably underestimated if the cortex is unfolded with a regular mean curvature flow (middle). This does not occur if we use a relative area preserving tangential velocity (right).

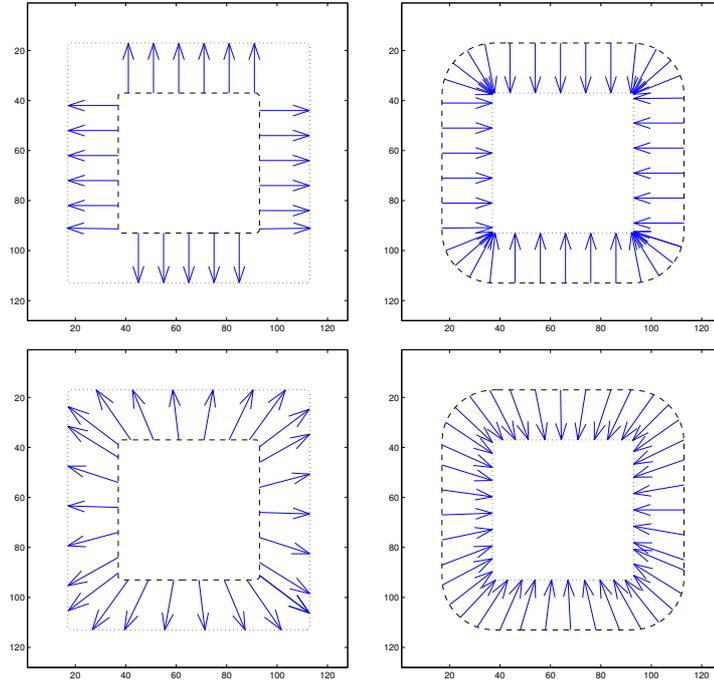


Figure 6.1: A shrinking (left) or expanding (right) square with (bottom) and without (top) a relative area preserving tangential velocity.

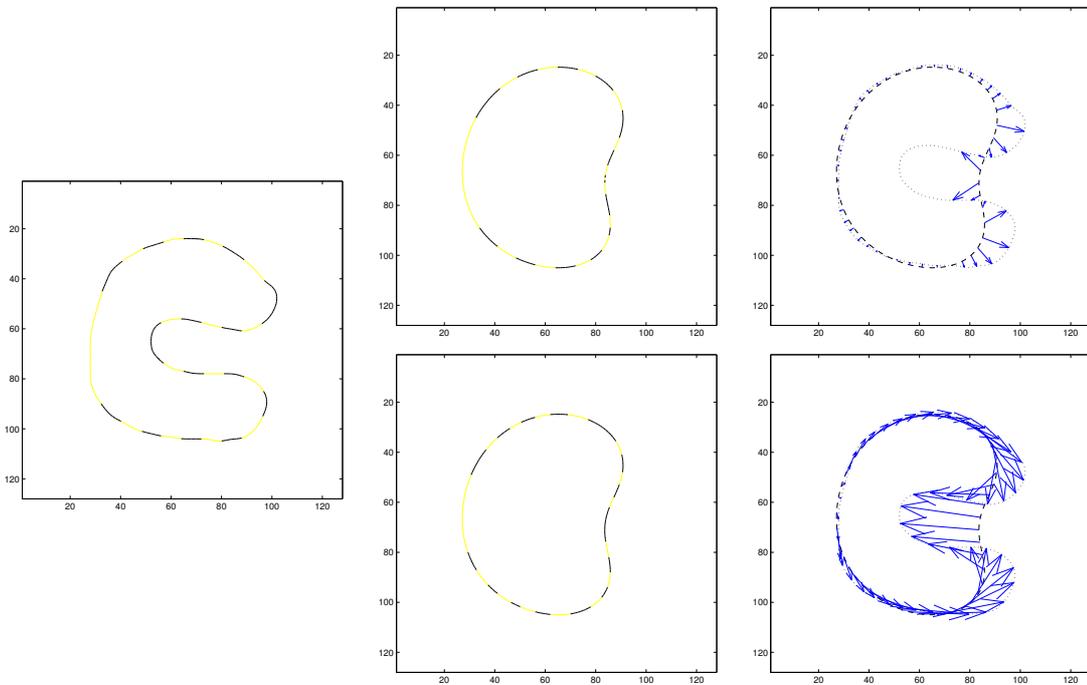


Figure 6.2: 2D unfolding with (bottom) and without (top) a relative area preserving tangential velocity.

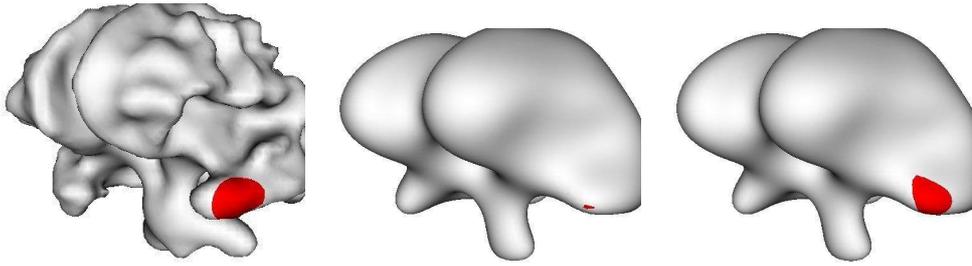


Figure 6.3: Unfolding a synthetic cortex with (right) or without (middle) a relative area preserving tangential velocity.

6.3.2 Real data

We now compute inflated representations of the cortex from real brain data. Our method is quite flexible through the choice of the normal motion, and we could obtain a sphere or any target shape by designing a force pulling the surface towards the target. In order to obtain smoothed versions of the cortex, we use the well-known and easy-to-implement motion by mean curvature $v_N = -H$ or a variant with an additional rescaling motion $v_N = -H + \overline{H^2}(\mathbf{x} \cdot \mathbf{N})$ in order to preserve the total area.

In the following experiments, the input of our algorithm is a $128 \times 128 \times 128$ level set of the pial surface of a human brain extracted from a $256 \times 256 \times 256$ anatomical T_1 -weighted MR image with the segmentation algorithm presented in Chapter 7. To give an idea of the resolution of the surface, the marching cubes algorithm [89] produces roughly 100000 triangles from this level set.

In Figure 6.4, in order to show the benefits of area preservation, we compare three inflated representations. The first one (row 2) was obtained with the nice method of Fischl et al. [52]. This method minimizes metric distortion but does not consider area. The second inflated cortex (row 3) was obtained with a standard mean curvature motion with a null tangential motion. The third one (row 4) was obtained with a mean curvature motion plus an area preserving tangential motion computed with our method. As expected, the geometries of last two representations are identical since the normal motion is the same in both cases. We display the histograms of the normalized areal factor J/\bar{J} in each case (column d). Not surprisingly, area distortion is far smaller when using our method. In this example, it is less than 5 percent almost everywhere. More interestingly, the overall aspect of all the representations is similar, which suggests that our method does not induce a blow-up in metric distortion. Moreover, as shown in Table 6.1, the amount of area distortion decreases when the area relaxation parameter λ increases. On the other hand, higher values of λ require to use smaller time steps and hence to perform a larger number of iterations.

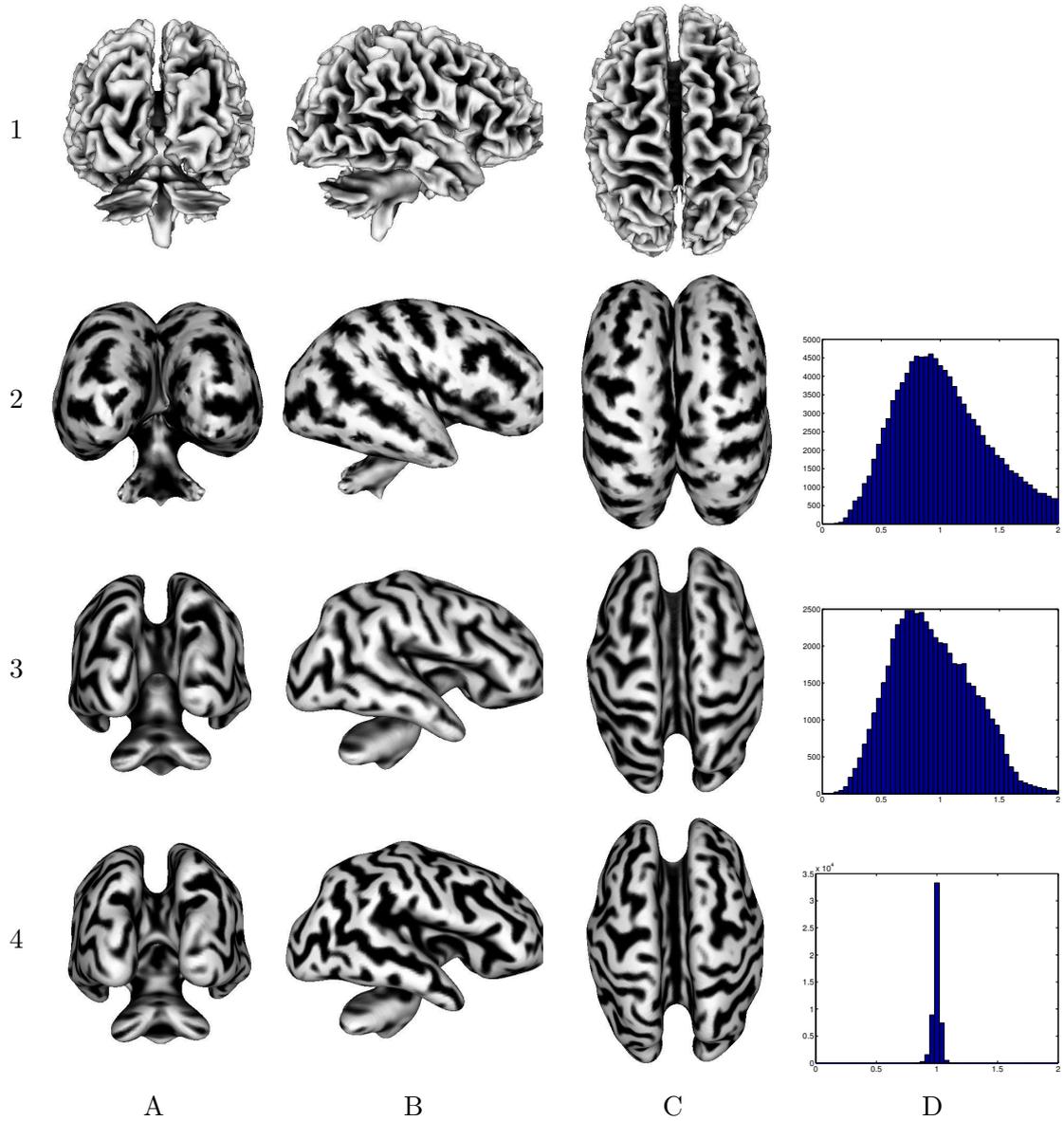


Figure 6.4: Several views (columns A-D) of the initial surface (row 1) and of the inflated representations obtained with the popular method of Fischl et al. (row 2), with a standard mean curvature motion (row 3) and with our method (row 4). Histograms of the normalized areal factor J/\bar{J} in each case (column D).

λ	1	3	5	10
$\sigma(J/\bar{J})$	0.060	0.041	0.035	0.027

Table 6.1: Standard deviation of J/\bar{J} against λ when using our method.

6.4 Contributions of this chapter

In this chapter, we have presented a novel method to generate unfolded area preserving representations of the cerebral cortex: depending on the normal motion driving the geometry, an adequate tangential motion is computed to ensure an exact area preservation. The contributions of this chapter are the following:

- We have stated the mathematical condition for the preservation of the local area of an evolving codimension one manifold in any number of dimensions.
- We have proposed a procedure to build an area preserving tangential velocity field from a given normal speed. We have described an implementation of this procedure with triangulated surfaces and with the level set method.
- We have applied this procedure to cortex unfolding with a preservation of the relative area throughout the evolution. We have demonstrated the efficiency of the approach on real brain data and we have compared it with a popular existing method.

This work has been published in the *International Conference on Medical Imaging and Computer-Assisted Intervention* [122].

Chapter 7

Head Segmentation from MRI under Topological Constraints

We have designed a method to automatically and accurately extract surface meshes of several head tissues from anatomical MR images. The emphasis of our method is on guaranteeing some topological properties of the output meshes, such as spherical topology, absence of self-intersections and mutual intersections. Our method is a successful combination of hidden Markov random field classification [185], of a region-based deformable models method, and of the topology preserving nested level set method proposed in Chapter 4. The latter allows to prevent mutual intersections between the inner and outer interfaces of the cerebral cortex. We demonstrate each step of our approach on real brain data.

7.1 Motivation

7.1.1 Problem statement

In brain imaging, the extraction of geometrical models of the head and of brain structures, in particular of the cortical surface, is often a first step in the study of brain anatomy and function. The morphometric study of sulcal and gyral patterns of an individual subject or of a population is useful for the clinical diagnosis of pathological diseases or for the generation of brain atlases. Moreover, geometric modeling is necessary for the reconstruction, visualization, analysis and interpretation of the brain activity obtained from functional imaging. For example, mapping the measurements obtained from positron emission tomography (PET) or functional magnetic resonance imaging (fMRI) to the cortical surface permits a better understanding of brain functional organization [36]. Also, the reconstruction of brain electrical activity from the measurements of electro-encephalography (EEG) or magneto-encephalography (MEG) outside the head requires to model the spatial distribution and the electrical conductivity of the different head tissues [48]. As a manual slice by slice labeling of the different tissues in MR images is tedious and inaccurate, an automatic, reliable and efficient 3D segmentation method is of particular importance.

This task is generally made difficult by the high complexity of anatomical structures and by various artefacts in the imaging process such as noise, partial volume effects, and intensity inhomogeneities related to the non-uniformity of the RF field. The segmentation of the cortical surface is particularly challenging due to its convoluted nature. The cerebral cortex is a highly folded sheet of gray matter (GM) at the surface of the brain. It is bounded by the cerebrospinal fluid (CSF) on the outside, and by the white matter (WM) on the inside. The human cortex has been reported to be 1.5 to 5.0 mm thick [191]. The reconstruction of the cortex from MRI has received much interest in the literature. The deformable models framework is particularly relevant to address this problem because it provides a consistent geometrical representation suitable for a surface-based analysis, contrarily to the voxel-wise labelings output by statistical classification methods [173, 66, 185].

7.1.2 Previous work

Before reviewing the previous work on cortex segmentation with deformable models, we invite the reader to go back to Subsection 2.3.1 for an overview of image segmentation with deformable models. The reader may also refer to [174, 97] for some surveys on deformable models in medical image segmentation.

An early work on cortex segmentation with deformable models appears in [37]. In this work, the center of the gray matter layer is reconstructed by evolving a parameterized deformable surface. The external forces driving the deformation are based on a binary indicator function of the gray matter. This binary mask is obtained by preprocessing the image with morphological operations and with a statistical classification method.

The idea of using two coupled surfaces to simultaneously extract the inner and outer interfaces of the cortex was concurrently introduced in [184, 91]. This coupled approach has several advantages over an approach with a single surface. First, when needed, it allows to compute directly the cortical thickness. Second, it improves the robustness to a low GM/WM contrast and to the partial volume effect. For instance, it helps the outer surface to enter into deep narrow sulci. It also prevents the inner surface from collapsing into CSF and the outer surface from penetrating non-brain tissue.

In [184], the two surfaces are represented by two level set functions, and their mutual distance is constrained to stay in a certain range. The evolution laws of the two level sets are based on boundary-based information, in the spirit of the geometric active contours method [23, 29, 94]. The intensity of each tissue is modeled by a Gaussian distribution, and an edge indicator function is computed as a measure of the likelihood of the voxels lying on the boundary between two tissues.

In [91], the formulation is variational and the geometric representation consists in two polyhedral surfaces. Self-intersections and mutual intersections are prevented by monitoring the distance between non-adjacent polygons, which makes this approach very expensive computationally. A term in the energy functional encourages a preferred distance between the two surfaces, which is likely to bias a posterior measurement of the cortical thick-

ness. The image term is the distance to the nearest edge of a precomputed statistical classification of the image.

In [60], a method similar to [184] is proposed. The major difference is a procedure to maintain the signed distance property of the two level set functions along the evolution. The evolution laws of the level sets rely on two intensity thresholds, as in (4.10), and should be regarded as a special case of the region-based image segmentation methods reviewed in Subsection 2.3.1.

A reformulation in a variational form of the methods proposed in [184, 60], in the spirit of the geodesic active contours method [24], has been proposed in [56]. Unfortunately, as discussed in Subsection 2.2, this work makes an abuse in the computation of the variation of the energy with respect to a variation of the level set function.

In [175], the reconstruction of the medial surface of the cortex is performed in several successive stages. First, the images are preprocessed to remove skin, bone, fat and other non-brain tissues. This step is often referred as *skull stripping* (see [137] and references therein). Second, a fuzzy c-means classification is computed, yielding a fuzzy membership function for each tissue. Third, an explicit deformable surface is initialized with an isosurface of the WM membership function. Finally, the deformable surface is evolved towards the maximum values of the GM membership function. In [64], the above method is upgraded with a topology preserving level set representation of the deformable surface, as described in Subsection 4.1.3, in order to obtain a final mesh with a certified spherical topology.

Recently, in [62], an extension of the methods of [175, 64] to the joint extraction of the inner, the central and the outer surfaces of the cortex has been proposed. This approach includes a procedure to guarantee the nesting of the different interfaces. The level set function of the central surface is prevented from becoming larger at any point than the level set function of the GM/WM boundary. Similarly, the level set function of the GM/CSF boundary is prevented from becoming larger at any point than the level set function of the central surface. However, contrarily to the topology preserving nested level set method proposed in Chapter 4, this procedure does not strictly prevent mutual intersections between the final meshes, because it is not consistent with the isocontour extraction step.

More precisely, we can easily find some level set functions $\phi_1, \phi_2 : \mathbb{R}^3 \rightarrow \mathbb{R}$ such that

$$\forall \mathbf{x} \in \Omega, \phi_1(\mathbf{x}) > \phi_2(\mathbf{x})$$

and such that there is still an intersection between the two triangulated surfaces output by the topology-consistent marching cubes algorithm described in Subsection 4.1.3.

For example, we consider the level set function ϕ_1 of the cortical surface displayed in Figure 6.4. The corresponding triangulation obtained with a (26,6) connectivity pair has 133 584 triangles and is of spherical topology. The triangulation obtained from $\phi_2 = \phi_1 - 0.5$ with the same connectivity pair has 133 768 triangles and is mostly outside of the first surface. But actually, the two triangulations intersect.

7.1.3 Goals of our approach

In this chapter, we propose a new method to reconstruct the boundary of several head tissues from MR images. The input of our algorithm is a T_1 -weighted anatomical MR image and the approximate intensities of the main head tissues: air, skin, cerebrospinal fluid, gray matter and white matter. It robustly generates triangulated surfaces of the outer skin interface, of the brain contour and of the inner and outer interfaces of the cortex. In the future, we plan to extend it to the skull and to some subcortical structures of interest. The emphasis of our method is on guaranteeing some topological properties of the output meshes, such as spherical topology, absence of self-intersections and mutual intersections. These properties are crucial in some applications such as cortex unfolding/flattening (*cf* Chapter 6). They are also necessary for building volume meshes of the head from the extracted surface meshes. Such volume meshes can be used in the source reconstruction problem from MEG/EEG [48].

Our method alleviates several limitations of existing approaches. First, unlike [62], we strictly guarantee that the final triangulations do not intersect. Second, in most works, the evolution of the deformable models is driven by a precomputed voxel-wise classification, and not directly by the MR intensities. This precludes a subvoxel accuracy. Also, a clear link between the deformation and the image formation process cannot be established. In our approach, a statistical classification method is used to provide a good initialization of the deformable models, to estimate the tissue parameters and to correct the effects of the bias field. Then, the evolution is driven by the intensity of the bias-corrected MR image, according to a Bayesian region-based formulation [190, 165, 114, 130, 71, 115].

7.2 Methods

7.2.1 Hidden Markov random field classification

The first step of our approach is a statistical classification of the voxels into a small number N of tissue classes chosen *a priori*. As proposed in [185], we use a hidden Markov random field classification, coupled with an automatic estimation of the tissue parameters and of the bias field with the expectation-maximization (EM) algorithm.

The tissue distribution is modeled by a Markov random field (MRF) [55] L encouraging neighboring voxels to have the same class labels, while the observed intensity of each tissue class $i \in \{1, \dots, N\}$ is modeled by a Gaussian distribution with mean μ_i and with standard deviation σ_i :

$$p(I(\mathbf{x}) = y \mid L(\mathbf{x}) = i) = \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(-\frac{(y - \mu_i)^2}{2\sigma_i^2}\right). \quad (7.1)$$

The labels of the voxels are estimated from the observed intensities with a *maximum a posteriori* (MAP) criterion:

$$\hat{L} = \arg \max_L P(L|I) = \arg \max_L P(I|L) P(L). \quad (7.2)$$

Moreover, the conditional independence of the intensities at the different voxels is assumed, yielding

$$P(I|L) = \prod_{\mathbf{x} \in \Omega} p(I(\mathbf{x})|L(\mathbf{x})), \quad (7.3)$$

while $P(L)$ is the Gibbs distribution associated to the MRF.

The MAP estimation then translates into the minimization of an energy functional. An exact minimization is computationally unfeasible due to the huge number of unknowns. As a consequence, a greedy strategy yielding a suboptimal solution is adopted: the deterministic iterated conditional modes (ICM) algorithm [17].

The parameters of the statistical model are the mean and the standard deviation of each tissue class, and a bias field accounting for the inhomogeneities in the RF field. In our method, this bias is taken as affine against intensities and smooth and non-parametric over space. In other words:

$$\forall \mathbf{x} \in \Omega, I(\mathbf{x}) = \alpha(\mathbf{x}) I^*(\mathbf{x}) + \beta(\mathbf{x}), \quad (7.4)$$

where I and I^* respectively denote the observed and ideal intensities, and α, β are two smooth functions.

An initial estimate of the tissue parameters is provided by the user. Then, iteratively, class labels are estimated by MAP, then the tissue parameters and the bias field are updated with the EM algorithm. The different outputs of this step are:

- a labeling L of image voxels,
- the mean μ_i and the standard deviation σ_i of each tissue class $i \in \{1, \dots, N\}$,
- a bias-corrected image, which is an estimation of the ideal image I^* .

7.2.2 Topology preserving nested level sets

The HMRF classification is powerful as regards automatic parameter estimation but it is not sub-voxel accurate and disregards topology. Extracting an isosurface of the labeling L would give a very coarse reconstruction with a lot of topological defects. Hence we feed the output of the previous step into a deformable models segmentation task, implemented with the topology preserving nested level set method proposed in Section 4.2.

This way, we can benefit from the advantages of both methods while discarding their respective drawbacks. The labeled image is used to position the initial deformable surfaces, thereby avoiding the problem of robustness to distant initializations. A set of topology preserving nested level sets are fit to the labeling L , before starting the deformable models evolution itself, in order to obtain a close and topologically consistent initialization. Later in the algorithm, the labels are no more taken into account, and the surfaces are evolved according to the intensities of the bias-corrected image.

The topology preserving nested level set method requires the choice of a connectivity pair for the different interfaces. In our method, this choice is motivated by the typical

aspect of sulcal and gyral patterns, illustrated in a close up of an anatomical MR image in Figure 7.1. This image reveals that the white matter and the cerebrospinal fluid feature

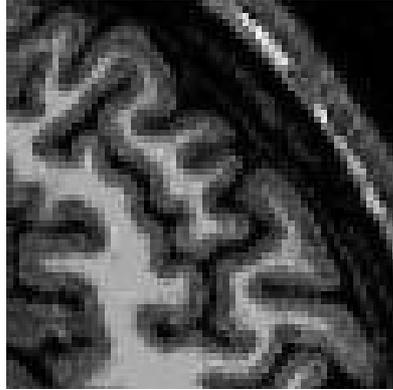


Figure 7.1: Close up of sulcal and gyral patterns in an anatomical MR image.

many sheet-like parts. In typical anatomical MR images, the thickness of these parts is of the order of one voxel while the thickness of the gray matter exceeds two or three voxels almost everywhere.

In order to correctly recover the connectivity of these sheet-like parts, the highest possible digital connectivity must be chosen for WM and CSF. Finally, the compatible connectivity pairs used for the different brain tissues are:

- WM: (26,6),
- GM: (6,26),
- CSF: (26,6).

Note that the alternation of the connectivity pairs, needed by the topology preserving nested level set method, is fulfilled.

7.2.3 Bayesian region-based deformable models evolution

Since the image inhomogeneities have been removed, the boundaries between the different tissues can be found robustly with a Bayesian region-based method, as proposed in [190, 165, 114, 130, 71, 115].

More precisely, let S_1, S_2, \dots, S_{N-1} be the surfaces separating the different tissues. We note X_i the region of space inside S_i . The surfaces are nested in the sense that

$$X_1 \subset X_2 \subset \dots \subset X_{N-1} . \quad (7.5)$$

The different tissues are located in the regions $\Omega_1, \Omega_2, \dots, \Omega_N$ comprised between two consecutive surfaces.

$$\begin{cases} \Omega_1 & = X_1 , \\ \Omega_i & = X_i \setminus X_{i-1} , \quad \forall 1 < i < N , \\ \Omega_N & = \Omega \setminus X_{N-1} . \end{cases} \quad (7.6)$$

Given the bias-corrected image, which we note I in the following, we want to estimate the partition $\mathcal{P}(\Omega) = \{\Omega_1, \dots, \Omega_N\}$. Again, we use a MAP estimation: we try to maximize

$$P(\mathcal{P}(\Omega)|I) \propto P(I|\mathcal{P}(\Omega)) P(\mathcal{P}(\Omega)) \quad (7.7)$$

with respect to the partition. The independence of the different regions and of the intensities at the different voxels is assumed, yielding

$$P(I|\mathcal{P}(\Omega)) = \prod_{\mathbf{x} \in \Omega_1} p(I(\mathbf{x})|\Omega_1) \prod_{\mathbf{x} \in \Omega_2} p(I(\mathbf{x})|\Omega_2) \dots \prod_{\mathbf{x} \in \Omega_N} p(I(\mathbf{x})|\Omega_N), \quad (7.8)$$

while $P(\mathcal{P}(\Omega))$ favors partitions with boundaries of small areas:

$$P(\mathcal{P}(\Omega)) \propto \exp \left(- \sum_{1 \leq i < N} \lambda_i \text{area}(S_i) \right). \quad (7.9)$$

The MAP estimation then translates into the minimization of the following energy functional:

$$E(\{\Omega_1, \dots, \Omega_N\}) = \sum_{1 \leq i < N} - \int_{\Omega_i} \log p(I(\mathbf{x})|\Omega_i) d\mathbf{x} + \lambda_i \text{area}(S_i). \quad (7.10)$$

The intensity statistics of the different regions are the Gaussian distributions estimated during the HMRF classification. Finally, the geometric gradient flows driving the deformation of the different surfaces are given by

$$\frac{\partial S_i}{\partial t} = \left(\frac{(I - \mu_{i+1})^2}{2\sigma_{i+1}^2} - \frac{(I - \mu_i)^2}{2\sigma_i^2} - 2 \log \frac{\sigma_{i+1}}{\sigma_i} - \lambda_i H_i \right) \mathbf{N}_i, \quad (7.11)$$

where \mathbf{N}_i is the outward normal and H_i is the mean curvature of S_i .

7.3 Experimental results

In our experiments, the input of our algorithm is a $256 \times 256 \times 256$ anatomical T_1 -weighted MR image. Some slices of this image are displayed in Figure 7.2.

In Figure 7.3, we show some slices of the labeling of brain tissues output by the HRMF classification step.

Figure 7.4 shows several views of the inner and outer boundaries of the cortex reconstructed with our method. The output meshes for the GM/WM and GM/CSF interfaces have 712244 and 947960 triangles, respectively. As expected, these meshes have a spherical topology (i.e. one connected component and an Euler number of 2), and do not have self-intersections or mutual intersections. Note how well the sheet-like parts of the GM/WM interface at the extremities of the gyri are recovered, including very thin parts in the cerebellum.

In order to evaluate the accuracy of the reconstruction, we display our results in some slices of the original MR data in Figure 7.5. Finally, in Figure 7.6, we show in a same 3D view the two interfaces of the cortex and the skin surface.

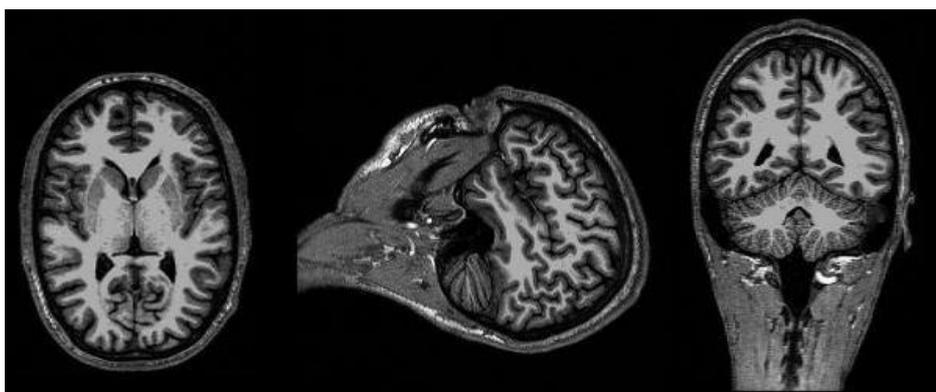


Figure 7.2: An axial (left), sagittal (middle) and coronal (right) slice of the input anatomical MR image used in our experiments.

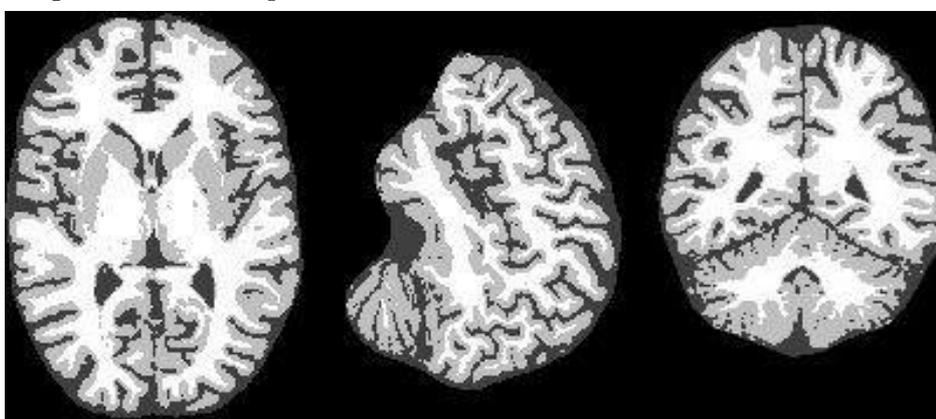


Figure 7.3: Slices of the labeling output by the HMRF classification step.

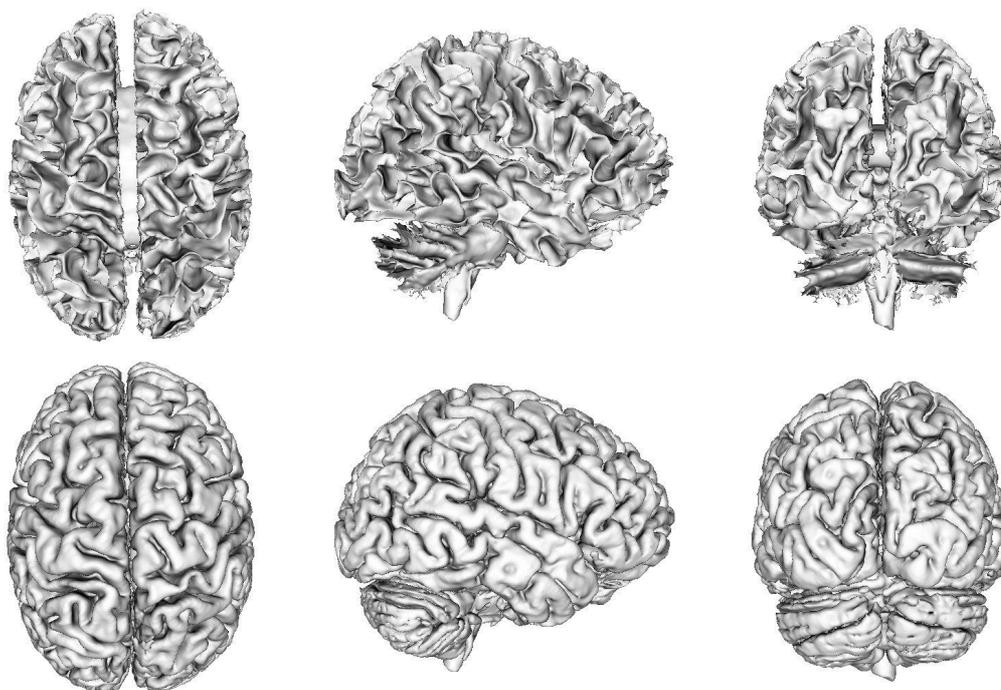


Figure 7.4: Several views of the GM/WM (top) and CSF/GM (bottom) interfaces.

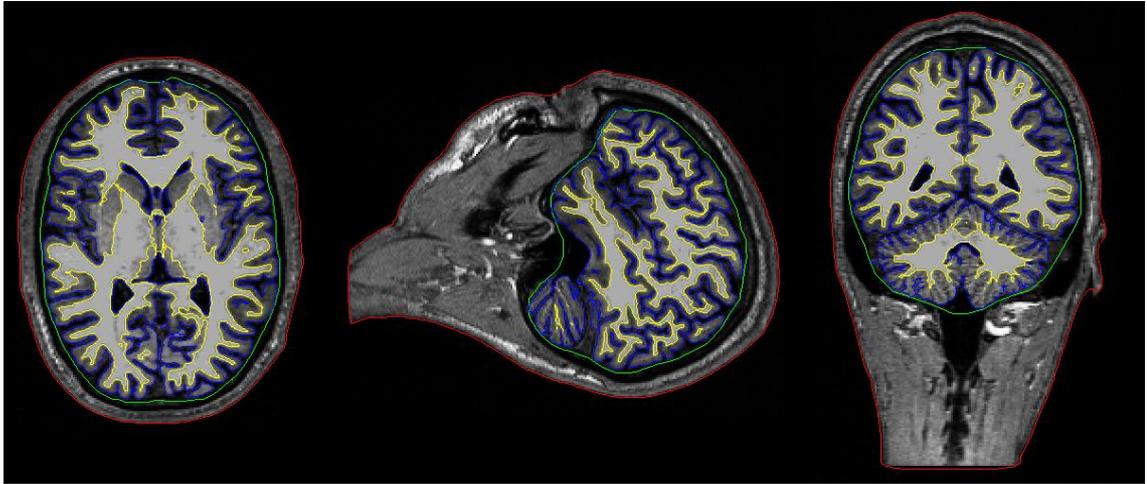


Figure 7.5: Some cuts of the different output interfaces, surimposed on some slices of the original MR image.



Figure 7.6: Combined view of the different interfaces.

7.4 Contributions of this chapter

In this chapter, we have presented a novel method to extract several head tissues from anatomical MR images. The contributions of our approach with respect to the existing literature are the following:

- The intersections between the different output meshes are strictly prevented, thanks to the topology preserving nested level set method proposed in Chapter 4.
- The deformable surfaces are evolved directly according to the bias-corrected MR intensities, thanks to a Bayesian region-based formulation.
- The thin sheet-like parts of WM and CSF bounding the cortex are recovered, thanks to an adequate choice of the digital connectivity pairs of the different tissues.

Our software is used routinely in the Odyssee Laboratory to obtain the surfaces needed for the analysis of fMRI signals and for the reconstruction of brain activity from MEG/EEG measurements. It has also been deployed at the Neurophysiology Section of the Katholieke Universiteit Leuven, Belgium, to segment the cortical surface of monkeys, in the context of the European MAPAWAMO Project: “Mapping visual cortex regions in awake, behaving monkey using functional MRI”. An example of the results obtained with our algorithm is displayed in Figure 7.7.

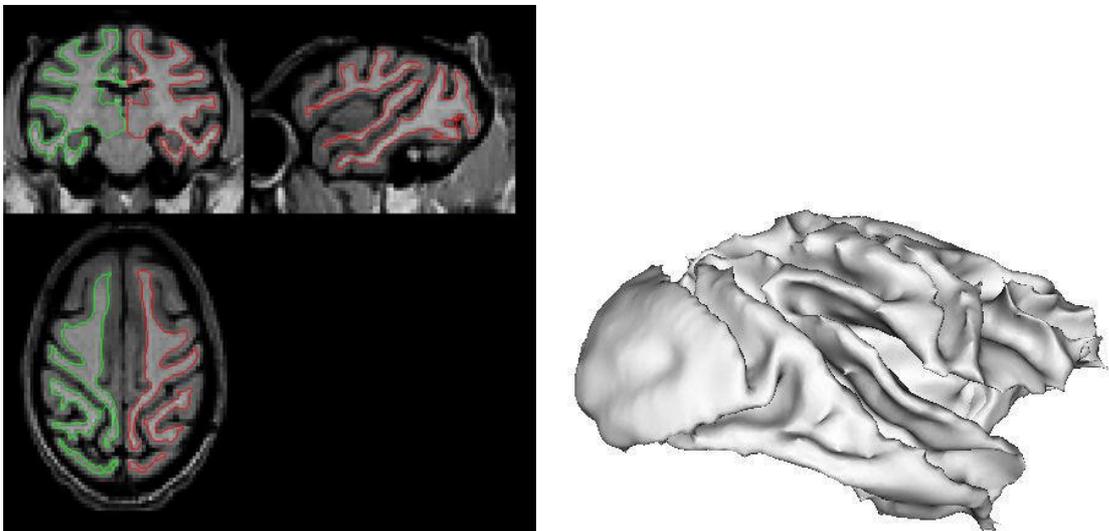


Figure 7.7: Segmentation of the cortical surface of a monkey with our method.

The content of this chapter is a joint work with Florent Ségonne, from the CSAIL (Computer Science and Artificial Intelligence Laboratory) at the MIT (Massachusetts Institute of Technology) and the Athinoula Martinos Center for Biomedical Imaging, at the MGH (Massachusetts General Hospital), Boston, USA. This work has appeared in a special issue of *NeuroImage* [48]

Chapter 8

Multi-View Stereo Reconstruction and Scene Flow Estimation with a Global Image-Based Matching Score

In this chapter, we present a new variational method for multi-view stereovision and non-rigid three-dimensional motion estimation from multiple video sequences. Our method minimizes the prediction error of the shape and motion estimates. Both problems then translate into a generic image registration task. The latter is entrusted to a global measure of image similarity, chosen depending on imaging conditions and scene properties. Contrarily to existing deformable models methods, which integrate a matching measure computed independently at each surface point, our approach computes a global image-based matching score between the input images and the predicted images. The matching process fully handles projective distortion and partial occlusions. Neighborhood as well as global intensity information can be exploited to improve the robustness to appearance changes due to non-Lambertian materials and illumination changes, without any approximation of shape, motion or visibility. Moreover, our approach results in a simpler, more flexible, and more efficient implementation than in existing methods. The computation time on large datasets does not exceed thirty minutes on a standard workstation. Finally, our method is compliant with a hardware implementation with graphics processor units. Our stereovision algorithm yields very good results on a variety of datasets including specularities and translucency. We have successfully tested our motion estimation algorithm on a very challenging multi-view video sequence of a non-rigid scene.

8.1 Introduction

8.1.1 Problem statement

Recovering the geometry of a scene from several images taken from different viewpoints, namely *stereovision*, is one of the oldest problems in computer vision. More recently, some authors have considered estimating the dense non-rigid three-dimensional motion field of a scene, often called *scene flow*¹ [167], from multiple video sequences. In this case, the input data are a two-dimensional array of images, in which each row is a multi-view stereovision dataset for a given time instant, and each column is a video sequence captured by a given camera. Combining stereovision and scene flow allows to build a spatio-temporal model of a dynamic scene. Once such a model is available, some novel virtual views of the scene can be generated by interpolation across space and time [166].

Stereovision and scene flow estimation both require to match different images of the same scene, in other words to find points in different cameras and in different frames corresponding to a same physical point. Once the correspondence problem is solved, the shape and the three-dimensional motion of the scene can be recovered easily by triangulation. Unfortunately, the correspondence problem is a very difficult task in computer vision because a scene patch generally has different shapes and appearances when seen from different points of view and at different times. To overcome this difficulty, most existing stereovision and scene flow algorithms rely on unrealistic simplifying assumptions that disregard either/both shape/appearance changes.

8.1.2 Common photometric and geometric assumptions used for shape and motion estimation

The oldest and most naive assumption about the photometric properties of a scene is *brightness constancy*: corresponding pixels are assumed to have the same color. This only applies to strictly Lambertian objects and requires a precise photometric calibration of the cameras. Yet this assumption is still popular in the stereovision literature. It motivates the multi-view photo-consistency measure used in voxel coloring [140], space carving [84], and in some deformable models methods [45, 88]. Similarly, the variational formulation of [154] relies on square intensity differences. In a later paper [153], the same authors model the intensity deviations from brightness constancy by a multivariate Gaussian. However, this does not remove any of the severe limitations of this simplistic assumption.

This assumption is also present in many methods for scene flow estimation, through the use of the spatio-temporal derivatives of the input images [186, 21, 109]. Due to the brightness constancy assumption and to the local relevance of spatio-temporal derivatives, these differential methods apply mainly to slowly-moving scenes under constant illumination.

For a better robustness to noise and to realistic imaging conditions, matching mea-

¹The scene flow should not be confused with the optical flow, which is the two-dimensional motion field of points in an image. The optical flow is the projection of the scene flow in the image plane of a camera.

asures embedded in stereovision and scene flow algorithms have to aggregate neighborhood intensity information. In return, they are confronted with geometric distortion between the different views and the different time instants. Some stereovision methods disregard this difficulty and use fixed matching windows. The underlying assumption is called the *fronto parallel hypothesis*: the retinal planes of the cameras are identical and the scene is an assembly of planes parallel to them. This assumption can still be found in recent work [88, 69]. To minimize the impact of projective distortion, these authors compute the stereo discrepancy of a scene patch with its most front-facing cameras only. However, this approximation is questionable in most camera setups.

Some methods go beyond this hypothesis by taking into account the tangent plane to the object [50, 75, 45, 57], or by using adaptive matching windows [76, 133]. More generally, most techniques trade robustness to realistic photometric conditions for an approximation of shape and motion in the computation of the matching measure. As a result, the robustness of the matching process is uncertain in the parts of the scene that do not verify these approximations. For example, using fixed matching windows for stereo correspondence leads to an oversmoothing of depth discontinuities. Similarly, using a tangent plane approximation to compute the matching measure as in [50, 75, 45, 57], even if the tangent plane at nearby points does not have to be the same, is not relevant in the regions of high curvature of the objects.

8.1.3 Previous work on multi-view complete stereovision

Doing a complete review of the stereovision area is out of the scope of this thesis. We limit ourselves to the methods that allow to obtain a complete reconstruction of a scene from a high number of input views. The methods in which the geometry is represented by one or several depth maps or disparity maps are not of interest here, because they only yield partial models of the scene. Several such models can be fused at post-processing, but anyway these methods cannot handle visibility globally and consistently during the estimation. For sake of completeness, let us mention two recent important works in this category: the graph cuts method of [80] and the PDE-based method of [154]. The interested reader can also refer to [134] for a good taxonomy of dense two-frame rectified stereo correspondence algorithms.

Thus, in the following, we focus on multi-view complete stereovision methods. These methods fall into two categories: the *space carving* framework and the deformable models framework.

Space carving

In the space carving framework [84], the scene is represented by a three-dimensional array of voxels. Each voxel can be labeled empty or occupied. When the algorithm starts, all voxels are occupied. Then the volume is traversed in an adequate order. If a voxel is not consistent with all the input images, it is relabeled empty. The order of the traversal is

important because the visibility of the voxels is taken into account in the consistency test. In an earlier method called voxel coloring [140], there was a constraint on the placement of the cameras, and the algorithm required only a single pass. Space carving handles arbitrary camera configurations but is a little more expensive computationally.

The space carving framework suffers from several important limitations. First, it makes hard decisions. Once a voxel is carved away, it cannot be recovered. And if one voxel is removed in error, further voxels can be erroneously removed in a cascade effect. This limitation is partially alleviated by the probabilistic space carving method [19]. Second, in the original space carving algorithm, the photo-consistency test derives from a brightness constancy constraint, and the choice of the global threshold on the color variance is often problematic. Recently, there have been some attempts to relax these photometric constraints [162, 178]. The robustness to calibration errors is also addressed in [83]. Third, the voxel-based representation disregards the continuity of shape, which makes it very hard to enforce any kind of spatial coherence. As a result, space carving is very sensitive to noise and outliers, and typically yields very noisy reconstructions.

Deformable models

Here, contrarily to the space carving framework, the formulation is continuous and has a geometric interpretation. The unknown scene is modelled by a two-dimensional surface, and scene reconstruction is stated in terms of an energy minimization. In Subsection 2.3.2, we have already mentioned the major multi-view stereovision deformable models methods [50, 75, 45, 88, 69, 57]. These approaches share several limitations. First, in all these methods, the matching measure is computed independently at each surface point, then these quantities are integrated on the surface. The matching measure at a point relies on a local approximation of the surface, either by a fronto-parallel plane [88, 69] or at best by the tangent plane [50, 75, 45, 57]. Moreover, the visibility of the whole neighborhood is assumed to be the same as the reference point. For example, in [50], the cross correlation between two slanted matching windows is computed without taking into account the eventual partial occlusions of the windows. The primary purpose of these assumptions is a simplification of the modelling and of the resulting computations. They are clearly not valid in real-world scenes, which typically include many occlusions, depth discontinuities and sharp angles. Thus, these simplifying assumptions make the robustness of the matching process on real data very uncertain.

Second, all these methods follow a minimal surface approach. One drawback of this approach is that data fidelity and regularization are mixed. As a result, it is difficult to tune the regularizing behavior. A good discussion of this topic can be found in [148]. The authors show in some numerical experiments that the results of [75] can be further improved by integrating the matching measure on the images rather than on the surface.

Third, they lack flexibility in the choice of the matching criterion. Photo-consistency [45, 88], the normalized cross correlation [50, 57, 69] and lastly the radiance tensor [75] have been considered. These matching measures are hard-wired in their respective method

and cannot be upgraded to cope with different imaging conditions.

Finally, the dependency of the matching measure on the surface normal leads to a complex implementation. It requires to handle matching windows of different shapes or a tessellation of the tangent plane, at each surface point. It also results in a very complex minimizing flow involving second-order derivatives of the matching score [58, 150]. More precisely, the energy has the following form:

$$E(S) = \int_S g(\mathbf{x}, \mathbf{N}) d\mathbf{x} , \quad (8.1)$$

where S denotes the surface and g the matching measure. The gradient writes

$$\nabla E(S) = [\nabla g \cdot \mathbf{N} + 2gH - \operatorname{div}_S(g\mathbf{N})] \mathbf{N} , \quad (8.2)$$

where \mathbf{N} is the normal, H is the mean curvature of the surface, $g_{\mathbf{N}}$ is the derivative of the matching measure with respect to the orientation of the tangent plane and div_S is the intrinsic divergence operator on the surface.

The computation of the last term of (8.2) is tricky, time-consuming and unstable, and, to our knowledge, all authors have resigned to ignore it.

8.1.4 Previous work on scene flow estimation

Three-dimensional motion estimation from multiple video sequences has long been limited to rigid or piecewise-rigid scenes [155, 44, 172] or parametric models [186, 93]. The problem of computing a dense non-rigid three-dimensional motion field from multiple video sequences has been addressed only recently. Two types of methods prevail in the scene flow literature.

The first family of methods [186, 21, 109] relies on the spatio-temporal derivatives of the input images. As pointed out in [167], estimating the scene flow from these derivatives without regularization is an ill-posed problem. Indeed, the associated normal flow equations only constrain the scene flow vector to lie on a line parallel to the iso-brightness contour on the object. This is nothing but a 3D version of the aperture problem for optical flow [12]. In [21, 109], several samples of the spatio-temporal derivatives are combined in order to overconstrain the scene flow, whereas in [186], the aperture problem is solved by complementing the normal flow constraint with a Tikhonov smoothness term. However, due to the underlying brightness constancy assumption, and to the local relevance of spatio-temporal derivatives, these differential methods apply mainly to slowly-moving Lambertian scenes under constant illumination.

In the second family of methods [167, 186, 144], the optical flow is computed independently in each camera, then these estimations are combined to get the scene flow. This approach is not optimal since it disregards the consistency between the different corresponding optical flows. Moreover, the noise in the different optical flows and the bias introduced by the heuristic spatial smoothness constraints alter the scene flow in an unpredictable manner.

Finally, there has been an attempt to extend the space carving framework to the simultaneous estimation of shape and scene flow [168] by using a plane-sweep algorithm in a 6D space. However, this approach again relies on a brightness constancy assumption, has a very high computational and memory cost, and is unable to enforce the smoothness of the recovered shape and motion.

8.1.5 Motivations of our approach

In this chapter, we propose a common variational framework for multi-view complete stereovision and scene flow estimation which overcomes most of the limitations listed above. The metric used in our framework is the ability to predict the other input views from one input view and the estimated shape or motion. This is related to the methodology proposed in [158] for evaluating the quality of motion estimation and stereo correspondence algorithms. But in our method, the prediction error is used for the estimation itself rather than for evaluation purposes.

Contrarily to existing deformable models approaches, which compute a matching measure independently at each surface point and integrate these quantities on the surface, or on the image domain as in [148], our approach computes a global image-based matching score between the input images and the predicted images. The matching process fully handles projective distortion and partial occlusions. Neighborhood as well as global intensity information can be exploited to improve the robustness to appearance changes, without any approximation of shape, motion or visibility.

Our formulation is completely decoupled from the nature of the image similarity measure used to assess the quality of the prediction. It can be the normalized cross correlation, some statistical measures such as the *correlation ratio* [129], *mutual information* [171, 92], or any other application-specific measure. Through this choice, we can make the estimation robust to camera spectral sensitivity differences, non-Lambertian materials and illumination changes. Also, any user-defined regularity constraint can be used.

Our method computes global matching scores on entire images from which projective distortion and semi-occluded regions have been removed, thereby avoiding the complex machinery usually needed to handle many matching windows of different shapes, or many tessellations of the tangent plane. The pixels used in the computation of the matching score are exactly the ones that are visible, judging from the current position of the surface. Moreover, the minimizing flow is much simpler than in [50, 75, 45, 57], in the sense that it only involves first-order derivatives of the matching score. This results in elegant and efficient algorithms.

Our scene flow method does not fall into the two existing categories. It works directly in 3D object space. It evolves a 3D vector field to register the input images captured at different times.

The rest of this chapter is organized as follows. In Section 8.2, we present our variational formulation of multi-view complete stereovision and non-rigid 3D motion estimation. In Section 8.3, we detail two particular similarity measures that can be used in our frame-

work: normalized cross correlation and mutual information. Section 8.4 describes our implementation with level sets and graphics card hardware acceleration, and presents our experimental results.

8.2 Minimizing the prediction error

Our method consists in maximizing, with respect to shape and motion, the similarity between each input view and the predicted images coming from the other views. We adequately warp the input images to compute the predicted images, which simultaneously removes projective distortion. Numerically, this can be done at a low computational cost using texture-mapping graphics hardware (*cf* Section 8.4). For example, in the case of stereovision, it corresponds to what is classically known as the reprojection error: we back-project the image taken by one camera on the surface estimate, then we project it to the other cameras to predict the appearance of the other views. The closer the shape estimate is to the actual geometry, the more similar the reprojected images will be to the corresponding input images, modulo noise, calibration errors, appearance changes and semi-occluded areas. This is the core principle of our approach. Although the expression “reprojection error” is more common in the stereovision literature, we use “prediction error” in the following, because it has the advantage of being relevant to both shape and motion estimation.

In our framework, both shape and motion estimation are formulated as a generic image registration task. This analogy is widely used in the context of rectified stereovision and optical flow. But it has fewer illustrations in multi-view stereo with arbitrary camera configurations, and it is definitely novel in the context of scene flow estimation. The registration task is entrusted to a global measure of image similarity, chosen depending on imaging conditions and scene properties. This measure is basically a function mapping two images to a scalar value. The more similar the two images are, the lower the value of the measure is. Neighborhood as well as global intensity information can be used in this measure.

We incorporate the similarity measure and a regularization term in an energy functional. The regularization term is required to make the problem well-posed. It is application-specific. For example, it could be designed to preserve shape or motion discontinuities. Here we focus primarily on the design of the matching term and we propose a basic smoothing regularization term.

To minimize our energy functionals, we use a geometric gradient flow, embedded in a multi-resolution coarse-to-fine strategy to decrease the probability of getting trapped in irrelevant local minima. We run the optimization on a series of smoothed and subsampled images.

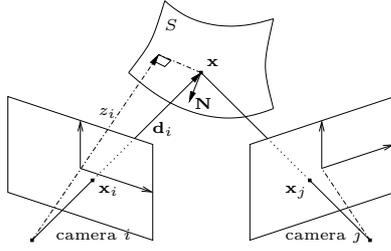


Figure 8.1: The camera setup and our notations.

8.2.1 Stereovision

In the following, let a surface $S \subset \mathbb{R}^3$ model the shape of the scene. We note $I_i : \Omega_i \subset \mathbb{R}^2 \rightarrow \mathbb{R}^d$ the image captured by camera i . In practice $d = 1$ for grayscale images or $d = 3$ for color images. The perspective projection performed by camera i is denoted by $\Pi_i : \mathbb{R}^3 \rightarrow \mathbb{R}^2$. Our method takes into account the visibility of the surface points. In the sequel, we will refer to S_i as the part of S visible in image i . The reprojection from camera i onto the surface is denoted by $\Pi_{i,S}^{-1} : \Pi_i(S) \rightarrow S_i$. With this notation in hand, the reprojection of image j in camera i via the surface writes $I_j \circ \Pi_j \circ \Pi_{i,S}^{-1} : \Pi_i(S_j) \rightarrow \mathbb{R}^d$. We note M a generic measure of similarity between two images.

The matching term \mathcal{M} is the sum of the dissimilarity between each input view and the reprojected images coming from all the other cameras. Thus, for each ordered pair of cameras (i, j) , we compute the similarity between I_i and the reprojection of I_j in camera i via S , on the domain where both are defined, i.e. $\Omega_i \cap \Pi_i(S_j)$, in other words after discarding semi-occluded regions:

$$\mathcal{M}(S) = \sum_i \sum_{j \neq i} \mathcal{M}_{ij}(S), \quad (8.3)$$

$$\mathcal{M}_{ij}(S) = M|_{\Omega_i \cap \Pi_i(S_j)} \left(I_i, I_j \circ \Pi_j \circ \Pi_{i,S}^{-1} \right). \quad (8.4)$$

Following [148], and for the reasons given in Subsection 8.1.3, we depart from the minimal surface approach. Our energy functional is the sum of a matching term computed in the images and of a user-defined regularization term. But our approach goes further than [148], in the sense that the matching process is global and completely image-based. In contrast, in [148], the matching measure was computed independently at each surface point, using an object-based tangent plane approximation, and later integrated on the image domain.

We now compute the variation of the matching term with respect to an infinitesimal vector displacement δS of the surface. Figure 8.1 displays the camera setup and our notations. We neglect the variation related to visibility changes. Indeed, the latter would yield an additional term confined to the horizons of the surface in the different cameras. Hence, this term only has a codimension-two support and its influence would be considerably decreased by the regularity constraint. This technical assumption is commonly used in

the stereovision literature [50, 75, 45, 88]. Using the chain rule, we get

$$\left. \frac{\partial \mathcal{M}_{ij}(S + \epsilon \delta S)}{\partial \epsilon} \right|_{\epsilon=0} = \int_{\Omega_i \cap \Pi_i(S_j)} \underbrace{\frac{\partial_2 M(\mathbf{x}_i)}{1 \times d}}_{1 \times d} \underbrace{DI_j(\mathbf{x}_j)}_{d \times 2} \underbrace{D\Pi_j(\mathbf{x})}_{2 \times 3} \underbrace{\left. \frac{\partial \Pi_{i,S+\epsilon \delta S}^{-1}(\mathbf{x}_i)}{\partial \epsilon} \right|_{\epsilon=0}}_{3 \times 1} d\mathbf{x}_i ,$$

where \mathbf{x}_i is the position in image i and $D \cdot$ denotes the Jacobian matrix of a function. To guide the reader, we have indicated the dimensions of the different matrices appearing in the product.

When the surface moves, the reprojected image changes. Hence the variation of the matching term involves the derivative of the similarity measure with respect to its second argument, denoted by $\partial_2 M$. Its meaning is detailed in Section 8.3. Throughout this section, for sake of conciseness, we have omitted the images for which this derivative is evaluated. But the reader must be aware that the reprojected images, as well as the domains where the similarity measures are computed, change along the minimizing flow.

We then use a relation between the motion of the surface and the displacement of the reprojected surface point $\mathbf{x} = \Pi_{i,S}^{-1}(\mathbf{x}_i)$:

$$\left. \frac{\partial \Pi_{i,S+\epsilon \delta S}^{-1}(\mathbf{x}_i)}{\partial \epsilon} \right|_{\epsilon=0} = \frac{\mathbf{N}^T \delta S(\mathbf{x})}{\mathbf{N}^T \mathbf{d}_i} \mathbf{d}_i ,$$

where \mathbf{d}_i is the vector joining the center of camera i and \mathbf{x} , and \mathbf{N} is the outward surface normal at this point. Finally, we rewrite the integral in the image as an integral on the surface by the change of variable

$$d\mathbf{x}_i = -\mathbf{N}^T \mathbf{d}_i d\mathbf{x} / z_i^3 ,$$

where z_i is the depth of \mathbf{x} in camera i , and we obtain

$$\left. \frac{\partial \mathcal{M}_{ij}(S + \epsilon \delta S)}{\partial \epsilon} \right|_{\epsilon=0} = - \int_{S_i \cap S_j} \left[\partial_2 M(\mathbf{x}_i) DI_j(\mathbf{x}_j) D\Pi_j(\mathbf{x}) \frac{\mathbf{d}_i}{z_i^3} \right] [\mathbf{N}^T \delta S(\mathbf{x})] d\mathbf{x} .$$

In other words, the gradient of the matching term is

$$\nabla \mathcal{M}_{ij}(S)(\mathbf{x}) = -\delta_{S_i \cap S_j}(\mathbf{x}) \left[\partial_2 M(\mathbf{x}_i) DI_j(\mathbf{x}_j) D\Pi_j(\mathbf{x}) \frac{\mathbf{d}_i}{z_i^3} \right] \mathbf{N} , \quad (8.5)$$

where $\delta \cdot$ is the Kronecker symbol. As expected, the gradient cancels in the regions not visible from both cameras. Note that the term between square brackets is a scalar function.

The regularization term is typically the area of the surface, and the associated minimizing flow is a mean curvature motion. The evolution of the surface is then driven by

$$\frac{\partial S}{\partial t} = \left[-\lambda H + \sum_i \sum_{j \neq i} \delta_{S_i \cap S_j} \partial_2 M DI_j D\Pi_j \frac{\mathbf{d}_i}{z_i^3} \right] \mathbf{N} , \quad (8.6)$$

where H denotes the mean curvature of S , and λ is a positive weighting factor.

8.2.2 Scene flow

Let now S^t model the shape of the scene and I_i^t be the image captured by camera i at time t . Let $v^t : S^t \rightarrow \mathbb{R}^3$ be a 3D vector field representing the motion of the scene between t and $t+1$. The matching term \mathcal{F} is the sum over all cameras of the dissimilarity between the images taken at time t and the corresponding images at $t+1$ warped back in time using the scene flow.

$$\mathcal{F}(v^t) = \sum_i \mathcal{F}_i(v^t), \quad (8.7)$$

$$\mathcal{F}_i(v^t) = M \left(I_i^t, I_i^{t+1} \circ \Pi_i \circ (\text{Id} + v^t) \circ \Pi_{i,S^t}^{-1} \right). \quad (8.8)$$

Its gradient writes

$$\begin{aligned} \nabla^T \mathcal{F}_i(v^t)(\mathbf{x}) = \\ - \delta_{S_i^t}(\mathbf{x}) \frac{\mathbf{N}^T \mathbf{d}_i}{z_i^3} \underbrace{\partial_2 M(\mathbf{x}_i)}_{1 \times d} \underbrace{DI_i^{t+1}(\Pi_i(\mathbf{x} + v^t(\mathbf{x})))}_{d \times 2} \underbrace{D\Pi_i(\mathbf{x} + v^t(\mathbf{x}))}_{2 \times 3}. \end{aligned} \quad (8.9)$$

Here, the regularization term is typically the harmonic energy of the flow over the surface, and the corresponding minimizing flow is an intrinsic heat equation based on the intrinsic Laplacian, often called the Laplace-Beltrami operator. The evolution of the scene flow is then driven by

$$\frac{\partial v^t}{\partial \tau} = \mu \Delta_{S^t} v^t + \sum_i \delta_{S_i^t} \frac{\mathbf{N}^T \mathbf{d}_i}{z_i^3} [\partial_2 M DI_i^{t+1} D\Pi_i]^T, \quad (8.10)$$

where τ is the fictitious time of the minimization, Δ_{S^t} denotes the Laplace-Beltrami operator on the surface, and μ is a positive weighting factor.

8.3 Some similarity measures

In this section, we present two similarity measures that can be used in our framework: normalized cross correlation and mutual information [171, 92]. Cross correlation assumes a local affine dependency between the intensities of the two images, whereas mutual information can cope with general statistical dependencies. We have picked these two measures among a broader family of statistical criteria proposed in [67] for multimodal image registration.

Whereas mutual information is an established matching measure in image registration, it has been seldom used for the stereo problem. To our knowledge, apart from our work, it only appears in the context of classical binocular rectified stereovision [79, 47].

In the following, we consider two scalar images $I_1, I_2 : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$. The measures below can be extended to vector (e.g. color) images by summing over the different components.

Note that the shape of Ω can be very complex. In practice, it corresponds to the domain where both an input image and an associated prediction are defined. In other words, semi-occluded regions are discarded from Ω .

The minimizing flows given in Section 8.2 involve the derivative of the similarity measure with respect to the second image, denoted by $\partial_2 M$. The meaning of this derivative is the following: given two images $I_1, I_2 : \Omega \rightarrow \mathbb{R}^d$, we note $\partial_2 M(I_1, I_2)$ the function mapping Ω to the row vectors of \mathbb{R}^d , verifying for any image variation δI :

$$\lim_{\epsilon \rightarrow 0} \frac{M(I_1, I_2 + \epsilon \delta I) - M(I_1, I_2)}{\epsilon} = \int_{\Omega} \partial_2 M(I_1, I_2)(\mathbf{x}) \delta I(\mathbf{x}) d\mathbf{x} . \quad (8.11)$$

8.3.1 Cross correlation

Cross correlation is still the most popular stereovision matching measure. Most methods settle for fixed rectangular correlation windows. In this case, the choice of the window size is a difficult trade-off between match reliability and oversmoothing of depth discontinuities due to projective distortion [134]. Some authors alleviate this problem by using adaptive windows [76, 133]. In our method, we match distortion-free images, so the size of the matching window is not related to a shape approximation. The matter here is in how big a neighborhood the assumption of affine dependency is valid. Typically, non-Lambertian scenes require to reduce the size of the correlation window, making the estimation less robust to noise and outliers. In our implementation, instead of hard windows, we use smooth Gaussian windows. They make the continuous formulation of our problem more elegant and they can be implemented efficiently with fast recursive filtering.

Thus, we gather neighborhood information using convolutions by a Gaussian kernel of standard deviation σ . The local mean, variance, covariance and cross correlation of the two images respectively write

$$\begin{aligned} \mu_i(\mathbf{x}) &= \frac{G_\sigma \star I_i(\mathbf{x})}{\omega(\mathbf{x})} , & v_i(\mathbf{x}) &= \frac{G_\sigma \star I_i^2(\mathbf{x})}{\omega(\mathbf{x})} - \mu_i^2(\mathbf{x}) + \beta^2 , \\ v_{1,2}(\mathbf{x}) &= \frac{G_\sigma \star I_1 I_2(\mathbf{x})}{\omega(\mathbf{x})} - \mu_1(\mathbf{x}) \mu_2(\mathbf{x}) , & cc(\mathbf{x}) &= \frac{v_{1,2}(\mathbf{x})}{\sqrt{v_1(\mathbf{x}) v_2(\mathbf{x})}} , \end{aligned}$$

where ω is a normalization function accounting for the shape of the domain: $\omega(\mathbf{x}) = \int_{\Omega} G_\sigma(\mathbf{x} - \mathbf{y}) d\mathbf{y}$. The β constant prevents the denominator from being zero. Beyond its numerical usefulness, this constant has a rigorous justification, as shown in [67]. It is related to the Parzen Gaussian kernel used to estimate the local joint probability distribution of the two images [117].

We aggregate the opposite of the local cross correlation to get a similarity measure corresponding to our needs:

$$M^{CC}(I_1, I_2) = - \int_{\Omega} cc(\mathbf{x}) d\mathbf{x} . \quad (8.12)$$

The minimizing flow involved by our method includes the derivative of the similarity measure with respect to the second image. In this case, it writes

$$\partial_2 M^{CC}(I_1, I_2)(\mathbf{x}) = \alpha(\mathbf{x}) I_1(\mathbf{x}) + \beta(\mathbf{x}) I_2(\mathbf{x}) + \gamma(\mathbf{x}) , \quad (8.13)$$

where

$$\begin{aligned}\alpha(\mathbf{x}) &= G_\sigma \star \frac{-1}{\omega \sqrt{v_1 v_2}}(\mathbf{x}), & \beta(\mathbf{x}) &= G_\sigma \star \frac{cc}{\omega v_2}(\mathbf{x}), \\ \gamma(\mathbf{x}) &= G_\sigma \star \left(\frac{\mu_1}{\omega \sqrt{v_1 v_2}} - \frac{\mu_2 cc}{\omega v_2} \right)(\mathbf{x}).\end{aligned}$$

In practice, along the minimizing flow, the α, β, γ functions change slowly relative to I_1 and I_2 . So, in our implementation, we update them only every ten iterations to reduce the computational burden.

8.3.2 Mutual information

Mutual information is based on the joint probability distribution of the two images, estimated by the Parzen window method [117] with a Gaussian kernel of standard deviation β :

$$P(i_1, i_2) = \frac{1}{|\Omega|} \int_{\Omega} G_\beta(I_1(\mathbf{x}) - i_1, I_2(\mathbf{x}) - i_2) d\mathbf{x}. \quad (8.14)$$

We note P_1, P_2 the marginals:

$$P_1(i_1) = \int_{\mathbb{R}} P(i_1, i_2) di_2, \quad P_2(i_2) = \int_{\mathbb{R}} P(i_1, i_2) di_1.$$

Our measure is the opposite of the mutual information of the two images:

$$M^{MI}(I_1, I_2) = - \int_{\mathbb{R}^2} P(i_1, i_2) \log \frac{P(i_1, i_2)}{P_1(i_1)P_2(i_2)} di_1 di_2. \quad (8.15)$$

Its derivative with respect to the second image writes [67, 49]:

$$\partial_2 M^{MI}(I_1, I_2)(\mathbf{x}) = \zeta(I_1(\mathbf{x}), I_2(\mathbf{x})), \quad (8.16)$$

where

$$\zeta(i_1, i_2) = \frac{1}{|\Omega|} G_\beta \star \left(\frac{\partial_2 P}{P} - \frac{P'_2}{P_2} \right)(i_1, i_2).$$

In our implementation, the ζ function is updated only every ten iterations.

8.4 Experimental results

We have implemented our method in the level set framework [112], motivated by its numerical stability and its ability to handle topology changes automatically. However, our method is not specific to a particular surface model: an implementation with meshes would be straightforward.

The predicted images can be computed very efficiently thanks to graphics card hardware-accelerated rasterizing capabilities. In our implementation, we determine the visibility of surface points in all cameras using OpenGL depth buffering, we compute the reprojection of an image to another camera via the surface using projective texture mapping, and we discard semi-occluded areas using shadow-mapping [135]. The bottleneck in our current implementation is the computation of the similarity measure. Since it only

involves homogeneous operations on entire images, we could probably resort to a graphics processor unit based implementation with fragment shaders (see <http://www.gpgpu.org>).

The only parameters inherent to our framework are the regularization coefficients λ and μ in equations (8.6) and (8.10). This being said, the similarity measure embedded in our method may have its own parameters: the size of the correlation window for cross correlation, the standard deviation of the Parzen kernel for mutual information, etc.

In all the following experiments, we have used a matching window with a standard deviation of 2 pixels ($\sigma = 2$) for cross correlation, and a Parzen kernel of variance 10 ($\beta^2 = 10$) for both cross correlation and mutual information.

8.4.1 Stereovision

Table 8.1 describes the stereovision datasets used in our experiments. All datasets are color images except “Hervé” which is grayscale. All are real images except “Buddha”. “Cactus” and “Gargoyle” are courtesy of Pr. Kyros Kutulakos (University of Toronto). “Buddha” and “Bust” are publicly available from the OpenLF software (LFM project, Intel).

Name	#Images	Image size	#Pairs	Measure	Level set size	Time (sec.)
Hervé	2	512×512	2	MI	128^3	107
Cactus	30	768×484	60	CC	128^3	1670
Gargoyle	16	719×485	32	MI	128^3	905
Buddha	25	500×500	50	CC	128^3	530
Bust	24	300×600	48	CC	$128 \times 128 \times 256$	1831

Table 8.1: Description of the stereovision datasets used in our experiments.

We have used either cross correlation (CC) or mutual information (MI), with $\sigma = 2$ and $\beta^2 = 10$. Both perform well on these complex scenes. “Buddha” and “Bust” are probably the more challenging datasets: “Buddha” is a synthetic scene simulating a translucent material and “Bust” includes strong specularities.

Using all possible camera pairs is not necessary since, when two cameras are far apart, no or little part of the scene is visible in both views. Consequently, in practice, we only pick pairs of neighboring cameras. The number of camera pairs used in each experiment is given in Table 8.1.

The number of iterations is 600 for all datasets. However, in most of our experiments, the convergence is attained earlier, so the computation time could be reduced using an appropriate stopping criterion. The only exception is the “Hervé” dataset, where the rear part of the face, not visible from any of the two cameras, is only driven by a mean curvature motion, and has not yet converged after 600 iterations, causing a rounded shape behind the face instead of a join of minimal area.

In all our experiments, the regularizer is a mean curvature motion, and the initial surface is an approximate bounding box of the scene. Although this initial guess is very



Figure 8.2: “Hervé” stereo pair and our results.

far from the objects, we manage to converge to the expected shape and to recover its concavities thanks to the coarse-to-fine strategy. We use four levels in the multi-resolution pyramid. So the level set size at the coarser resolution is 16^3 for most datasets and $16 \times 16 \times 32$ for the “Bust” dataset.

We show our results in Figures 8.2, 8.3, 8.4, 8.5 and 8.6. For each dataset, we display some of the input images, the ground truth when available, then some views of the estimated shape, and finally the same views after reprojecting the texture coming from the most front-facing camera. Note that this texture-mapped representation does not aim at photorealism. In particular, it generates artifacts at the places where the source of the texture changes. It is only intended to show the validity of the output of our method for more sophisticated image-based rendering techniques.

In all our experiments, the overall shape of the objects is successfully recovered, and a lot of details are captured: the eyes and the mouth of “Hervé”, the stings of “Cactus”, the ears and the pedestal of “Gargoyle”, the nose and the collar of “Buddha”, the ears and the mustache of “Bust”. A few defects are of course visible. Some of them can be explained. The hole around the stick of “Gargoyle” is not fully recovered. This may be due to the limited number of images (16): some parts of the concavity are visible only in one camera. The depression in the forehead of “Bust” is related to a very strong specularities: intensity is almost saturated in some images. In Figure 8.7, we illustrate the multi-resolution evolution of the surface for the “Bust” dataset, starting from a coarse bounding box.

Finally, in Table 8.2, we compare our results with the non-Lambertian stereovision method of [75] on the “Buddha” and the “Bust” datasets. We adopt the same shape error measure than in their work: the ratio between the volume of the symmetric difference between the estimated shape and the true shape and the volume of the true shape. The errors on the “Buddha” dataset are comparable. Our method performs significantly better than [75] on the “Bust” dataset. Moreover, visually, our reconstructions are slightly more

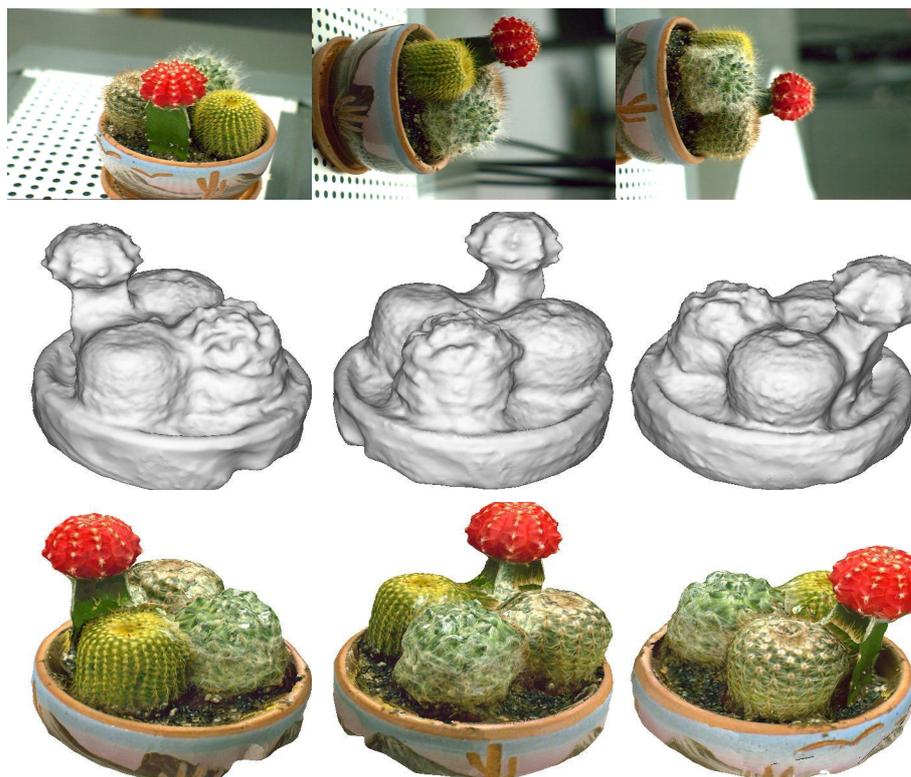


Figure 8.3: Some images from the “Cactus” dataset and our results.

detailed. But above all, our computation time is considerably smaller. It does not exceed thirty minutes on a 2 GHz Pentium IV PC under Linux, versus several hours.

8.4.2 Stereovision + scene flow

We have tested our scene flow algorithm on a challenging multi-view video sequence of a non-rigid scene. The “Yiannis” sequence is taken from a collection of datasets that were made available to the community by Dr. Patrick Baker and Dr. Jan Neumann (University of Maryland) for benchmark purposes. This sequence shows a character (Pr. Yiannis Aloimonos) talking while rotating his head. It was captured by 22 cameras at 54 fps plus 8 high-resolution cameras at 6 fps. Here we focus on the 30 synchronized sequences at the lower frame rate to demonstrate that our method can handle large displacements.

We have applied successively our stereovision and scene flow algorithms: once we know the shape S^t , we compute the 3D motion \mathbf{v}^t with our scene flow algorithm. Since $S^t + \mathbf{v}^t$ is a very good estimate of S^{t+1} , we use it as the initial condition in our stereovision algorithm and we perform a handful of iterations to refine it. This is much faster than restarting the optimization from scratch. We also compute the backward motion from $t + 1$ to t for the purpose of time interpolation.

In this experiment, we use cross correlation with the value of the parameters given earlier. The level set size is 128^3 and the number of levels of the multi-resolution pyramid is 4.



Figure 8.4: Some images from the “Gargoyle” dataset and our results.

Figure 8.8 displays the first four frames of one of the input sequence and our estimation of shape and 3D forward motion at corresponding times. We successfully recover the opening and closing of the mouth, followed by the rotation of the head while the mouth opens again. Moreover, we capture displacements of more than twenty pixels.

We use our results to generate time-interpolated 3D sequences of the scene. To synthesize images at intermediate time instants, we can either use the previous shape and texture warped by the forward motion, or the next shape and texture warped by the backward motion. Ideally the two should coincide exactly, but of course this is never the case in practice. As a consequence, we linearly interpolate between forward and backward extrapolated images to guarantee a smooth blending between frames. In return it causes “crossfading” artifacts in some places where forward and backward extrapolation significantly diverge.

We display a short excerpt of such a time-interpolated sequence in Figure 8.9. Note the progressive opening and closing of the mouth. Please see the *Odyssee Laboratory* web page for more results.

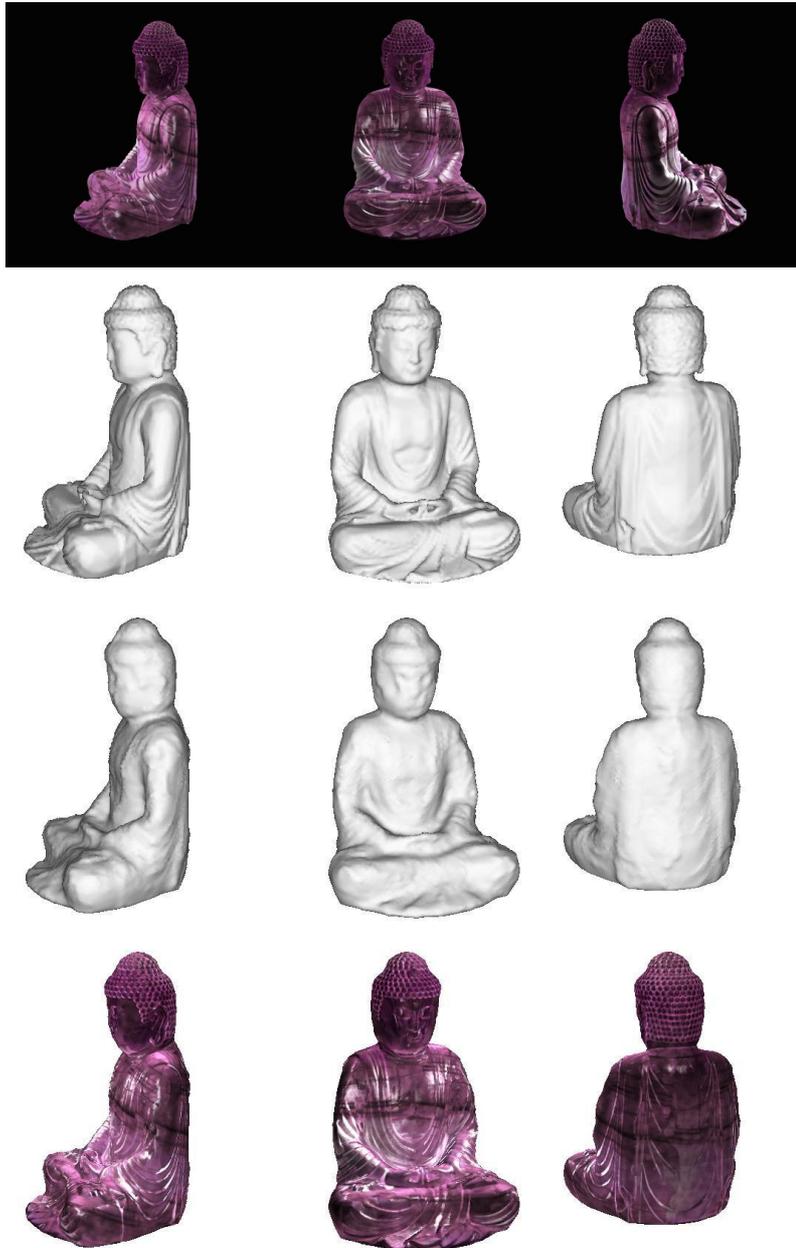


Figure 8.5: Some images from the “Buddha” dataset, ground truth and our results.

Method	Error on “Buddha”	Error on “Bust”
[75]	3.5 %	5.7 %
Our method	4.0 %	3.0 %

Table 8.2: Quantitative comparison between our method and the non-Lambertian stereo-
vision method of [75].

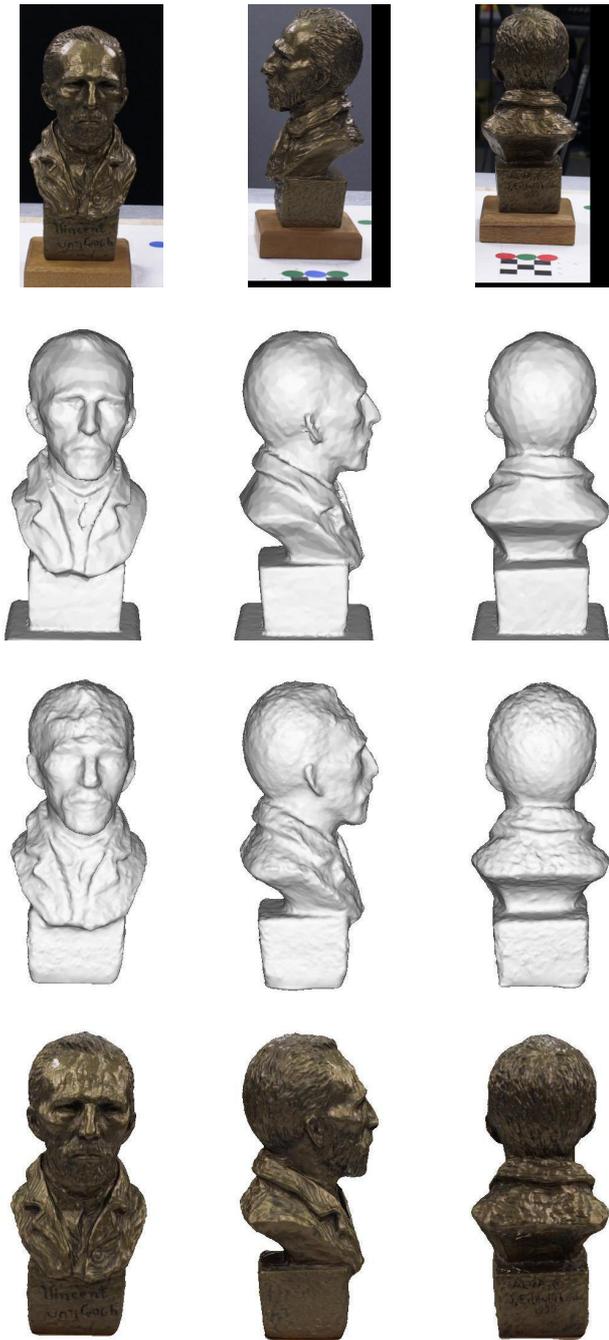


Figure 8.6: Some images from the “Bust” dataset, pseudo ground truth and our results.

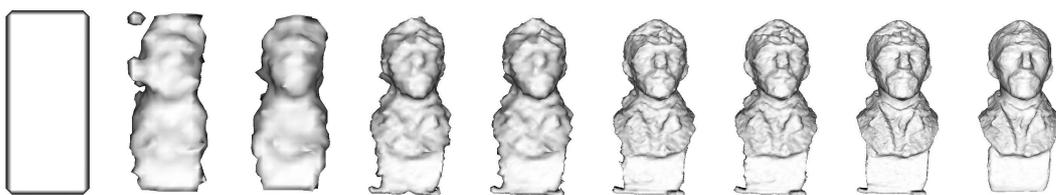


Figure 8.7: Multi-resolution shape evolution for the “Bust” dataset.

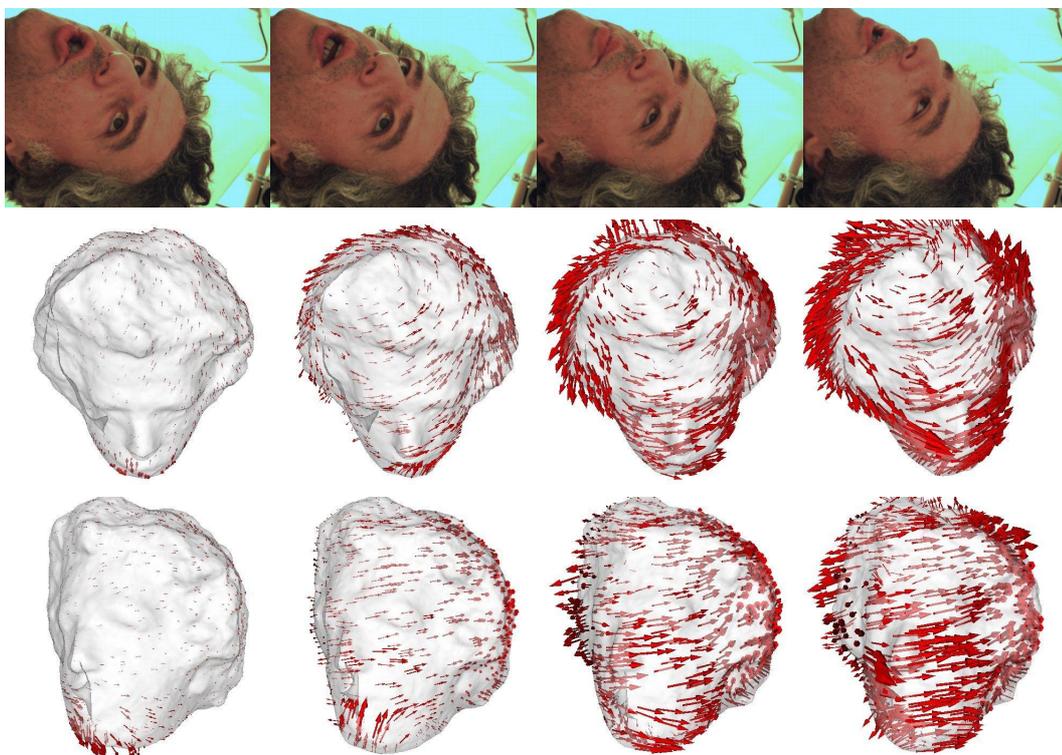


Figure 8.8: First images of one sequence of the “Yiannis” dataset and our results.



Figure 8.9: An excerpt of the time-interpolated 3D sequence for the “Yiannis” dataset.

8.5 Contributions of this chapter

In this chapter, we have presented a novel method for multi-view stereovision and scene flow estimation which minimizes the prediction error using a global image-based matching score. We adequately warp the input views and we register the resulting distortion-free images with a user-defined image similarity measure, which can include neighborhood and global intensity information. No approximation of shape, motion or visibility is made in the matching process.

We have implemented our stereovision method with the level set method and we have obtained results comparing favorably with state-of-the-art methods, even on complex non-Lambertian real-world images including specularities and translucency. Using our algorithm for motion estimation, we have successfully recovered the 3D motion of a non-rigid scene and we have synthesized time-interpolated 3D sequences.

Part of this work has been presented in computer vision conferences [124, 126] and has appeared in a book chapter [125] and a technical report [123]. A journal version has been submitted to *The International Journal of Computer Vision* [127].

Acknowledgements

We would like to thank Pr. Kyros Kutulakos for providing us with the “Cactus” and “Gargoyle” datasets, and Dr. Jan Neumann for his support on the “Yiannis” dataset.

Conclusion

In this thesis, we have made the deformable models framework progress along a number of axes, with the goal of a wider applicability and a greater efficiency. Our contributions span three key elements: the geometric representation, the minimization procedure, and the design of the energy functional. All along this thesis, we have pointed out the shortcomings concealed in several established components of the deformable models framework, such as the level set method, geometric gradient flows and the geodesic active contours approach. Departing from these standard tools have permitted us to consider new applications or to shed a different light on some classical problems:

- We have overcome the loss of the point correspondence with the level set method. As a result, we have been able to apply the level set framework to cortex unfolding.
- We have improved the control of topology changes with the level set method, so that we have been able to segment medical images with a certified topology.
- We have proposed new minimizing flows which are likely to increase the robustness to local minima of many deformable models methods.
- We have revisited multi-view stereovision with deformable models. We have substituted a global image-based matching score to the widely used minimal surface approach, to obtain a more exact and more flexible modeling, and a more efficient implementation.

Throughout these studies, implementation issues have been a major concern, because we are convinced that disregarding numerical aspects can lead the most promising modelings to a failure.

Of course, there is still place for improvement in most parts of this thesis. Although we believe that many problems in computer vision and in medical imaging can potentially benefit from our methodological proposals, a detailed inventory of these problems and a thorough experimental evaluation should be carried out. We are currently exploring other fields of application of the methodological contributions of this thesis. For example, a work on shape statistics of the cortical surface with landmark points is in progress, based on our level set method with a point correspondence.

In the second part of this thesis, our future challenge is certainly an in depth evaluation of our approaches for area preserving cortex unfolding and head segmentation from MRI

on a large real dataset. The validation of our results by experts would also be desirable. Finally, as regards multi-view stereovision and scene flow estimation, our future work includes a hardware implementation with graphics processor units to further reduce the computation time, and the fusion of shape and motion estimations in order to exploit their redundancy.

Conclusion (version française)

Dans cette thèse, nous avons fait progresser la méthode des modèles déformables selon plusieurs axes, avec pour but d'élargir leur champ d'application et d'accroître leur performance. Nos contributions s'étendent à trois éléments clés: la représentation géométrique, la procédure de minimisation et l'élaboration de la fonctionnelle d'énergie. Tout au long de cette thèse, nous avons mis en évidence les insuffisances que renferment plusieurs composantes bien établies de la méthode des modèles déformables, comme la méthode des ensembles de niveau, les flots géométriques de gradient, et la méthode des contours actifs géodésiques. En nous écartant de ces outils standards, nous avons pu envisager de nouvelles applications, ou jeter un regard neuf sur des problèmes classiques :

- Nous sommes venus à bout de la perte de la correspondance ponctuelle avec la méthode des ensembles de niveau. Du coup, nous avons pu appliquer ce cadre au dépliement de cortex.
- Nous avons amélioré le contrôle des changements de topologie avec la méthode des ensembles de niveau, ce qui nous a permis de segmenter des images médicales avec une topologie certifiée.
- Nous avons proposé de nouveaux flots de minimisation susceptibles d'accroître la robustesse aux minima locaux de nombreuses méthodes à base de modèles déformables.
- Nous avons revisité la stéréovision multi-caméras par modèles déformables. Nous avons substitué un critère de mise en correspondance global et basé image à l'approche très prisée par surface minimale, afin d'obtenir une modélisation plus fidèle et plus flexible, et une implémentation plus efficace.

Dans tous ces travaux, les questions d'implémentation ont eu une place majeure, car nous sommes convaincus que négliger les aspects numériques peut faire échouer les modélisations les plus prometteuses.

Bien sûr, des améliorations sont possibles dans de nombreuses parties de cette thèse. Ainsi, nous pensons que nos propositions méthodologiques peuvent bénéficier à de nombreux problèmes en vision par ordinateur et en imagerie médicale, mais un inventaire précis et une évaluation expérimentale complète devraient être menés. Actuellement, nous explorons d'autres champs d'application des contributions méthodologiques de cette thèse. Par exemple, un travail sur des statistiques de forme de la surface corticale au moyen de

points d'amer est en cours, sur la base de notre méthode des ensembles de niveau avec une correspondance ponctuelle.

Dans la deuxième partie de cette thèse, le défi à relever est certainement une évaluation en profondeur, sur un grand nombre de jeux de données réelles, de nos approches de dépliement de cortex avec préservation de l'aire et de segmentation de la tête à partir d'images IRM. La validation de nos résultats par des experts serait également souhaitable. Enfin, en ce qui concerne la stéréovision multi-caméras et l'estimation du flot de scène, nos travaux futurs comprennent une implémentation matérielle sur carte graphique, pour réduire encore davantage le temps de calcul, et la fusion des estimations de la forme et du mouvement, afin d'exploiter leur redondance.

Appendices

Appendix A

Formulas of Geometric and Differential Calculus

Throughout this appendix, we note Γ a closed embedded codimension-one interface in \mathbb{R}^n . We also consider a scalar function $f : \Gamma \rightarrow \mathbb{R}$ and a vector function $\mathbf{v} : \Gamma \rightarrow \mathbb{R}^n$ defined on the interface.

A.1 Some useful identities of intrinsic differential geometry

In this thesis, we use the following identities, taken from [43, 38]:

- The intrinsic Laplacian (Laplace-Beltrami operator) is defined from the intrinsic gradient and the intrinsic divergence:

$$\Delta_{\Gamma} f = \operatorname{div}_{\Gamma} \nabla_{\Gamma} f . \quad (\text{A.1})$$

- The classical formula for the divergence of a product of functions holds with intrinsic differential operators:

$$\operatorname{div}_{\Gamma}(f \mathbf{v}) = f \operatorname{div}_{\Gamma} \mathbf{v} + \nabla_{\Gamma} f \cdot \mathbf{v} . \quad (\text{A.2})$$

- The intrinsic divergence of the field of normals equals the total curvature:

$$\operatorname{div}_{\Gamma} \mathbf{N} = (n - 1) H . \quad (\text{A.3})$$

- The intrinsic divergence of a normal vector field has a simple form:

$$\operatorname{div}_{\Gamma}(f \mathbf{N}) = (n - 1) f H . \quad (\text{A.4})$$

A.2 Some useful expressions for implicit interfaces

In this section, the interface is represented by an embedding level set function $\phi : \mathbb{R}^n \rightarrow \mathbb{R}$ such that

$$\begin{cases} \phi(\mathbf{x}) < 0 & \text{if } \mathbf{x} \text{ is inside } \Gamma, \\ \phi(\mathbf{x}) = 0 & \text{if } \mathbf{x} \in \Gamma, \\ \phi(\mathbf{x}) > 0 & \text{if } \mathbf{x} \text{ is outside } \Gamma. \end{cases}$$

By a slight abuse of notation, f and \mathbf{v} now denote some extensions of the interfacial quantities in the whole space. Then, the common geometric properties and intrinsic differential operators are given by the following formulas:

- Outward unit normal:

$$\mathbf{N} = \frac{\nabla\phi}{\|\nabla\phi\|}. \quad (\text{A.5})$$

- Mean curvature:

$$H = \frac{1}{n-1} \operatorname{div} \left(\frac{\nabla\phi}{\|\nabla\phi\|} \right). \quad (\text{A.6})$$

In 2D, it leads to the following expression:

$$\kappa = \frac{\phi_{xx}\phi_y^2 - 2\phi_x\phi_y\phi_{xy} + \phi_{yy}\phi_x^2}{(\phi_x^2 + \phi_y^2)^{3/2}}. \quad (\text{A.7})$$

In 3D:

$$H = \frac{\left\{ \begin{array}{l} (\phi_{yy} + \phi_{zz})\phi_x^2 + (\phi_{xx} + \phi_{zz})\phi_y^2 + (\phi_{xx} + \phi_{yy})\phi_z^2 \\ -2\phi_x\phi_y\phi_{xy} - 2\phi_x\phi_z\phi_{xz} - 2\phi_y\phi_z\phi_{yz} \end{array} \right\}}{2(\phi_x^2 + \phi_y^2 + \phi_z^2)^2}. \quad (\text{A.8})$$

- Projection of a vector $\mathbf{u} \in \mathbb{R}^n$ on a plane orthogonal to $\mathbf{v} \in \mathbb{R}^n$:

$$P_{\mathbf{v}}\mathbf{u} = \mathbf{u} - \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{v}\|^2} \mathbf{v}. \quad (\text{A.9})$$

- Intrinsic gradient:

$$\nabla_{\Gamma} f = P_{\nabla\phi} \nabla f \quad (\text{A.10})$$

$$= \nabla f - \frac{\nabla f \cdot \nabla\phi}{\|\nabla\phi\|^2} \nabla\phi. \quad (\text{A.11})$$

- Intrinsic Laplacian (Laplace-Beltrami operator):

$$\Delta_{\Gamma} f = \frac{1}{\|\nabla\phi\|} \operatorname{div}(\|\nabla\phi\| P_{\nabla\phi} \nabla f). \quad (\text{A.12})$$

Below, we transform this expression to get an alternative formula:

$$\begin{aligned} \Delta_{\Gamma} f &= \frac{1}{\|\nabla\phi\|} \operatorname{div}(\|\nabla\phi\| \nabla f) - \frac{1}{\|\nabla\phi\|} \operatorname{div} \left((\nabla\phi \cdot \nabla f) \frac{\nabla\phi}{\|\nabla\phi\|} \right) \\ &= \Delta f + \frac{1}{\|\nabla\phi\|} \nabla(\|\nabla\phi\|) \cdot \nabla f - \frac{\nabla(\nabla\phi \cdot \nabla f)}{\|\nabla\phi\|} \cdot \frac{\nabla\phi}{\|\nabla\phi\|} - \frac{\nabla\phi \cdot \nabla f}{\|\nabla\phi\|} \operatorname{div} \left(\frac{\nabla\phi}{\|\nabla\phi\|} \right) \\ &= \Delta f + \frac{D^2\phi \nabla\phi}{\|\nabla\phi\|^2} \cdot \nabla f - \frac{\nabla\phi^T D^2 f \nabla\phi}{\|\nabla\phi\|^2} - \frac{D^2\phi \nabla\phi}{\|\nabla\phi\|^2} \cdot \nabla f - (n-1)H \frac{\nabla\phi \cdot \nabla f}{\|\nabla\phi\|} \\ &= \Delta f - \frac{\nabla\phi^T D^2 f \nabla\phi}{\|\nabla\phi\|^2} - (n-1)H \frac{\nabla\phi \cdot \nabla f}{\|\nabla\phi\|}, \end{aligned} \quad (\text{A.13})$$

where $D^2 f$ denotes the Hessian matrix of f .

- Intrinsic Jacobian:

$$D_{\Gamma} \mathbf{v} = D\mathbf{v} \left(\mathbf{I} - \frac{\nabla \phi \nabla \phi^T}{\|\nabla \phi\|^2} \right), \quad (\text{A.14})$$

where $D\mathbf{v}$ denotes the Jacobian matrix of \mathbf{v} .

- Intrinsic divergence:

$$\text{div}_{\Gamma} \mathbf{v} = \text{tr } D_{\Gamma} \mathbf{v} \quad (\text{A.15})$$

$$= \text{div } \mathbf{v} - \frac{\nabla \phi^T D\mathbf{v} \nabla \phi}{\|\nabla \phi\|^2}. \quad (\text{A.16})$$

- Average along the interface:

$$\bar{f} = \frac{\int_{\mathbb{R}^n} f(\mathbf{x}) \delta_{\epsilon}(\phi(\mathbf{x})) \|\nabla \phi(\mathbf{x})\| d\mathbf{x}}{\int_{\mathbb{R}^n} \delta_{\epsilon}(\phi(\mathbf{x})) \|\nabla \phi(\mathbf{x})\| d\mathbf{x}}, \quad (\text{A.17})$$

where δ_{ϵ} is a smoothed version of the Dirac function:

$$\delta_{\epsilon}(d) = \begin{cases} \frac{1}{2\epsilon} \left(1 + \cos \left(\frac{\pi d}{\epsilon} \right) \right) & \text{if } |d| < \epsilon, \\ 0 & \text{otherwise.} \end{cases} \quad (\text{A.18})$$

Appendix B

Numerical Schemes

B.1 Evolution schemes

In this section, we detail the numerical schemes necessary to implement the evolution of the level set function ϕ , depending on the properties of the velocity field \mathbf{v} . In some cases, the velocity field is a sum of several motions of different nature (e.g. the sum of a normal propagation and of a mean curvature motion). In this case, each part of the velocity field is treated with the adequate scheme and the total variation of the level function is the sum of these different contributions.

All the schemes below are taken from [112, 143, 156, 187]. We give their expression in 3D, but they can easily be adapted to any number of dimensions. We note ϕ_{ijk} the grid values of the level set function. Let $\phi_{ijk}^{\pm x}, \phi_{ijk}^{\pm y}, \phi_{ijk}^{\pm z}$ denote its decentered spatial derivatives. They can be computed either with a simple decentered difference of the form

$$\phi_{ijk}^{+x} = \frac{\phi_{i+1jk} - \phi_{ijk}}{\Delta x},$$

where Δx is the size of a grid step, or with the high-order WENO finite-difference schemes described in Appendix B.2.

B.1.1 Normal propagation

The evolution equation for a normal propagation with speed β is:

$$\frac{\partial \phi}{\partial t} + \beta \|\nabla \phi\| = 0.$$

The corresponding numerical scheme is:

$$\frac{\partial \phi_{ijk}}{\partial t} = - \left[\max(\beta_{ijk}, 0) \Delta_{ijk}^+ + \min(\beta_{ijk}, 0) \Delta_{ijk}^- \right],$$

where

$$\Delta_{ijk}^+ = \left[\max(\phi_{ijk}^{-x}, 0)^2 + \min(\phi_{ijk}^{+x}, 0)^2 + \max(\phi_{ijk}^{-y}, 0)^2 + \min(\phi_{ijk}^{+y}, 0)^2 + \max(\phi_{ijk}^{-z}, 0)^2 + \min(\phi_{ijk}^{+z}, 0)^2 \right]^{1/2},$$

$$\Delta_{ijk}^- = \left[\max(\phi_{ijk}^{+x}, 0)^2 + \min(\phi_{ijk}^{-x}, 0)^2 + \max(\phi_{ijk}^{+y}, 0)^2 + \min(\phi_{ijk}^{-y}, 0)^2 + \max(\phi_{ijk}^{+z}, 0)^2 + \min(\phi_{ijk}^{-z}, 0)^2 \right]^{1/2} .$$

B.1.2 Passive advection

The evolution equation for a passive advection with a velocity field \mathbf{v} is:

$$\frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi = 0 .$$

The corresponding numerical scheme is:

$$\frac{\partial \phi_{ijk}}{\partial t} = - \left[\max(v_{ijk}^x, 0) \phi_{ijk}^{-x} + \min(v_{ijk}^x, 0) \phi_{ijk}^{+x} + \max(v_{ijk}^y, 0) \phi_{ijk}^{-y} + \min(v_{ijk}^y, 0) \phi_{ijk}^{+y} + \max(v_{ijk}^z, 0) \phi_{ijk}^{-z} + \min(v_{ijk}^z, 0) \phi_{ijk}^{+z} \right] .$$

B.1.3 Mean curvature motion

The evolution equation for a mean curvature motion is:

$$\frac{\partial \phi}{\partial t} = \frac{1}{n-1} \operatorname{div} \left(\frac{\nabla \phi}{\|\nabla \phi\|} \right) \|\nabla \phi\| .$$

The corresponding numerical scheme is based on the formulas (A.7) or (A.8), with all the first-order and the second-order spatial derivatives computed with central differences. For example,

$$\phi_{ijk}^{0x} = \frac{\phi_{i+1jk} - \phi_{i-1jk}}{2\Delta x} .$$

B.1.4 Reinitialization

The reinitialization PDE is:

$$\frac{\partial \phi}{\partial t} + \operatorname{sign}(\phi_0) (\|\nabla \phi\| - 1) = 0 .$$

The corresponding numerical scheme derives from the scheme for normal propagation:

$$\frac{\partial \phi_{ijk}}{\partial t} = S_\epsilon - \left[\max(S_\epsilon, 0) \Delta_{ijk}^+ + \min(S_\epsilon, 0) \Delta_{ijk}^- \right] ,$$

where S_ϵ is a smoothed version of $\operatorname{sign}(\phi_0)$:

$$S_\epsilon = \frac{\phi_{0ijk}}{\sqrt{\epsilon^2 + \phi_{0ijk}^2}} .$$

B.1.5 Extension

The extension PDE is:

$$\frac{\partial f}{\partial t} + \text{sign}(\phi) (\nabla f \cdot \nabla \phi) = 0 .$$

The corresponding numerical scheme derives from the scheme for passive advection, with the following velocity field:

$$v_{ijk}^x = S_\epsilon \phi_{ijk}^{0x}, \quad v_{ijk}^y = S_\epsilon \phi_{ijk}^{0y}, \quad v_{ijk}^z = S_\epsilon \phi_{ijk}^{0z} .$$

B.2 High-order WENO finite-difference schemes

In our implementation, for a greater accuracy, the decentered spatial derivatives involved in the evolution schemes of Appendix B.1 are computed with the high-order WENO (weighted essentially non-oscillatory) finite difference schemes derived in [73, 72]. Below, we give the formulas for the derivatives of ϕ in the x direction for the WENO schemes of third-order and fifth-order. These formulas can easily be adapted to the derivatives in the other directions.

B.2.1 Third-order WENO scheme

$$\begin{aligned} \phi_i^{+x} &= \Phi^{WENO3} \left(\frac{\phi_{i+2} - \phi_{i+1}}{\Delta x}, \frac{\phi_{i+1} - \phi_i}{\Delta x}, \frac{\phi_i - \phi_{i-1}}{\Delta x} \right), \\ \phi_i^{-x} &= \Phi^{WENO3} \left(\frac{\phi_{i-1} - \phi_{i-2}}{\Delta x}, \frac{\phi_i - \phi_{i-1}}{\Delta x}, \frac{\phi_{i+1} - \phi_i}{\Delta x} \right), \end{aligned}$$

where

$$\Phi^{WENO3}(a, b, c) = \frac{b + c - \omega(a - 2b + c)}{2}, \quad \omega = \frac{1}{1 + 2r^2}, \quad r = \frac{\epsilon + (a - b)^2}{\epsilon + (b - c)^2} .$$

B.2.2 Fifth-order WENO scheme

$$\begin{aligned} \phi_i^{+x} &= \Phi^{WENO5} \left(\frac{\phi_{i+3} - \phi_{i+2}}{\Delta x}, \frac{\phi_{i+2} - \phi_{i+1}}{\Delta x}, \frac{\phi_{i+1} - \phi_i}{\Delta x}, \frac{\phi_i - \phi_{i-1}}{\Delta x}, \frac{\phi_{i-1} - \phi_{i-2}}{\Delta x} \right), \\ \phi_i^{-x} &= \Phi^{WENO5} \left(\frac{\phi_{i-2} - \phi_{i-3}}{\Delta x}, \frac{\phi_{i-1} - \phi_{i-2}}{\Delta x}, \frac{\phi_i - \phi_{i-1}}{\Delta x}, \frac{\phi_{i+1} - \phi_i}{\Delta x}, \frac{\phi_{i+2} - \phi_{i+1}}{\Delta x} \right), \end{aligned}$$

where

$$\Phi^{WENO5}(a, b, c, d, e) = \frac{\omega_0(2a - 7b + 11c) + \omega_1(-b + 5c + 2d) + \omega_2(2c + 5d - e)}{6(\omega_0 + \omega_1 + \omega_2)},$$

$$\omega_0 = \frac{1}{(\epsilon + r_0)^2}, \quad \omega_1 = \frac{6}{(\epsilon + r_1)^2}, \quad \omega_2 = \frac{3}{(\epsilon + r_2)^2},$$

$$r_0 = \frac{13}{12}(a - 2b + c)^2 + \frac{1}{4}(a - 4b + 3c)^2,$$

$$r_1 = \frac{13}{12}(b - 2c + d)^2 + \frac{1}{4}(b - d)^2,$$

$$r_2 = \frac{13}{12}(c - 2d + e)^2 + \frac{1}{4}(3c - 4d + e)^2.$$

B.3 Finite-difference discretization of the Laplace-Beltrami operator

Following [13], we have computed explicitly a finite-difference discretization of the Laplace-Beltrami operator by averaging complementary schemes which alternate backward and forward derivatives for the computation of the gradient and the divergence. We have obtained the following expression in 3D:

$$\begin{aligned}
\mathbf{N}_{i,j,k} &= \frac{1}{2} (\phi_{i+1,j,k} - \phi_{i-1,j,k}, \phi_{i,j+1,k} - \phi_{i,j-1,k}, \phi_{i,j,k+1} - \phi_{i,j,k-1}) \\
c_{+x} &= (\|\mathbf{N}_{i,j,k}\| - N_{i,j,k}^x / \|\mathbf{N}_{i,j,k}\| + \|\mathbf{N}_{i+1,j,k}\| - N_{i+1,j,k}^x / \|\mathbf{N}_{i+1,j,k}\|) / 2 \\
c_{-x} &= (\|\mathbf{N}_{i,j,k}\| - N_{i,j,k}^x / \|\mathbf{N}_{i,j,k}\| + \|\mathbf{N}_{i-1,j,k}\| - N_{i-1,j,k}^x / \|\mathbf{N}_{i-1,j,k}\|) / 2 \\
c_{+y} &= \left(\|\mathbf{N}_{i,j,k}\| - N_{i,j,k}^y / \|\mathbf{N}_{i,j,k}\| + \|\mathbf{N}_{i,j+1,k}\| - N_{i,j+1,k}^y / \|\mathbf{N}_{i,j+1,k}\| \right) / 2 \\
c_{-y} &= \left(\|\mathbf{N}_{i,j,k}\| - N_{i,j,k}^y / \|\mathbf{N}_{i,j,k}\| + \|\mathbf{N}_{i,j-1,k}\| - N_{i,j-1,k}^y / \|\mathbf{N}_{i,j-1,k}\| \right) / 2 \\
c_{+z} &= (\|\mathbf{N}_{i,j,k}\| - N_{i,j,k}^z / \|\mathbf{N}_{i,j,k}\| + \|\mathbf{N}_{i,j,k+1}\| - N_{i,j,k+1}^z / \|\mathbf{N}_{i,j,k+1}\|) / 2 \\
c_{-z} &= (\|\mathbf{N}_{i,j,k}\| - N_{i,j,k}^z / \|\mathbf{N}_{i,j,k}\| + \|\mathbf{N}_{i,j,k-1}\| - N_{i,j,k-1}^z / \|\mathbf{N}_{i,j,k-1}\|) / 2 \\
\\
c_{+xy} &= N_{i+1,j,k}^x N_{i+1,j,k}^y / \|\mathbf{N}_{i+1,j,k}\| & c_{-xy} &= N_{i-1,j,k}^x N_{i-1,j,k}^y / \|\mathbf{N}_{i-1,j,k}\| \\
c_{+xz} &= N_{i+1,j,k}^x N_{i+1,j,k}^z / \|\mathbf{N}_{i+1,j,k}\| & c_{-xz} &= N_{i-1,j,k}^x N_{i-1,j,k}^z / \|\mathbf{N}_{i-1,j,k}\| \\
c_{+yx} &= N_{i,j+1,k}^y N_{i,j+1,k}^x / \|\mathbf{N}_{i,j+1,k}\| & c_{-yx} &= N_{i,j-1,k}^y N_{i,j-1,k}^x / \|\mathbf{N}_{i,j-1,k}\| \\
c_{+yz} &= N_{i,j+1,k}^y N_{i,j+1,k}^z / \|\mathbf{N}_{i,j+1,k}\| & c_{-yz} &= N_{i,j-1,k}^y N_{i,j-1,k}^z / \|\mathbf{N}_{i,j-1,k}\| \\
c_{+zx} &= N_{i,j,k+1}^z N_{i,j,k+1}^x / \|\mathbf{N}_{i,j,k+1}\| & c_{-zx} &= N_{i,j,k-1}^z N_{i,j,k-1}^x / \|\mathbf{N}_{i,j,k-1}\| \\
c_{+zy} &= N_{i,j,k+1}^z N_{i,j,k+1}^y / \|\mathbf{N}_{i,j,k+1}\| & c_{-zy} &= N_{i,j,k-1}^z N_{i,j,k-1}^y / \|\mathbf{N}_{i,j,k-1}\|
\end{aligned}$$

$$\begin{aligned}
\Delta_{\Gamma} f_{i,j,k} &= -(c_{+x} + c_{-x} + c_{+y} + c_{-y} + c_{+z} + c_{-z}) f_{i,j,k} \\
&\quad + c_{+x} f_{i+1,j,k} + c_{-x} f_{i-1,j,k} + c_{+y} f_{i,j+1,k} \\
&\quad + c_{-y} f_{i,j-1,k} + c_{+z} f_{i,j,k+1} + c_{-z} f_{i,j,k-1} + \\
&\quad [- (c_{+xy} + c_{+yx}) f_{i+1,j+1,k} + (c_{-xy} + c_{+yx}) f_{i-1,j+1,k} \\
&\quad + (c_{+xy} + c_{-yx}) f_{i+1,j-1,k} - (c_{-xy} + c_{-yx}) f_{i-1,j-1,k} \\
&\quad - (c_{+xz} + c_{+zx}) f_{i+1,j,k+1} + (c_{-xz} + c_{+zx}) f_{i-1,j,k+1} \\
&\quad + (c_{+xz} + c_{-zx}) f_{i+1,j,k-1} - (c_{-xz} + c_{-zx}) f_{i-1,j,k-1} \\
&\quad - (c_{+yz} + c_{+zy}) f_{i,j+1,k+1} + (c_{-yz} + c_{+zy}) f_{i,j-1,k+1} \\
&\quad + (c_{+yz} + c_{-zy}) f_{i,j+1,k-1} - (c_{-yz} + c_{-zy}) f_{i,j-1,k-1}] / 4
\end{aligned}$$

The above expression is further simplified if ϕ is a signed distance function.

Appendix C

Publications of the Author

Book chapters

- J.-P. Pons, R. Keriven and O. Faugeras. **Modelling non-rigid dynamic scenes from multi-view image sequences.** In N. Paragios, Y. Chen and O. Faugeras, eds, *Mathematical Models in Computer Vision: The Handbook*, chapter 27, Springer, 2005.

Journal papers

- J.-P. Pons, G. Hermosillo, R. Keriven and O. Faugeras. **Maintaining the point correspondence in the level set framework.** Submitted to *Journal of Computational Physics*, 2005.
- J.-P. Pons, R. Keriven and O. Faugeras. **Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score.** Submitted to *The International Journal of Computer Vision*, 2005.

Conference papers

- G. Charpiat, R. Keriven, J.-P. Pons and O. Faugeras. **Designing spatially coherent minimizing flows for variational problems based on active contours.** In *International Conference on Computer Vision*, volume 2, pages 1403-1408, 2005.
- J.-P. Pons, R. Keriven and O. Faugeras. **Modelling dynamic scenes by registering multi-view image sequences.** In *International Conference on Computer Vision and Pattern Recognition*, volume 2, pages 822-827, 2005.
- J.-P. Pons, R. Keriven and O. Faugeras. **Area preserving cortex unfolding.** In *International Conference on Medical Image Computing and Computer Assisted Intervention*, volume 1, pages 376-383, 2004.

- J.-P. Pons, G. Hermosillo, R. Keriven and O. Faugeras. **How to deal with point correspondences and tangential velocities in the level set framework.** In *International Conference on Computer Vision*, volume 2, pages 894-899, 2003.
- J.-P. Pons, R. Keriven, O. Faugeras and G. Hermosillo. **Variational stereovision and 3D scene flow estimation with statistical similarity measures.** In *International Conference on Computer Vision*, volume 2, pages 597-602, 2003.

Technical reports

- F. Ségonne, J.-P. Pons, E. Grimson and B. Fischl. **A novel active contour framework. Multi-component level set evolution under topology control.** Technical report AIM-2005-020, CSAIL/MIT, 2005.
- J.-P. Pons, R. Keriven and O. Faugeras. **Modelling dynamic scenes by registering multi-view image sequences.** Technical report 5321, INRIA, 2004.
- J.-P. Pons, G. Hermosillo, R. Keriven and O. Faugeras. **How to deal with point correspondences and tangential velocities in the level set framework.** Technical report 4857, INRIA, 2003.

Others

- F. Ségonne, J.-P. Pons, E. Grimson and B. Fischl. **A novel level set framework for the segmentation of medical images under topology control.** To appear in *Workshop on Computer Vision for Biomedical Image Applications: Current Techniques and Future Trends*, 2005.
- N. Wotawa, J.-P. Pons, L. Lopez, R. Deriche and O. Faugeras. **fMRI data smoothing constrained to the cortical surface: a comparison of the level-set and mesh-based approaches.** In *Human Brain Mapping*, 2004.

Bibliography

- [1] D. Adalsteinsson and J.A. Sethian. A fast level set method for propagating interfaces. *Journal of Computational Physics*, 118(2):269–277, 1995.
- [2] D. Adalsteinsson and J.A. Sethian. The fast construction of extension velocities in level set methods. *Journal of Computational Physics*, 148(1):2–22, 1999.
- [3] D. Adalsteinsson and J.A. Sethian. Transport and diffusion of material quantities on propagating interfaces via level set methods. *Journal of Computational Physics*, 185(1):271–288, 2003.
- [4] N.I. Akhiezer and I.M. Glazman. *Theory of Linear Operators in Hilbert Space*. Pitman, 1981.
- [5] L. Ambrosio and H.M. Soner. Level set approach to mean curvature flow in arbitrary codimension. *Journal of Differential Geometry*, 43:693–737, 1996.
- [6] A.A. Amini, T.E. Weymouth, and R. Jain. Using dynamic programming for solving variational problems in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(9):855–867, 1990.
- [7] S. Angenent, S. Haker, A. Tannenbaum, and R. Kikinis. Laplace-Beltrami operator and brain surface flattening. *IEEE Transactions on Medical Imaging*, 18(8):700–711, 1999.
- [8] S. Angenent, S. Haker, A. Tannenbaum, and R. Kikinis. On area preserving mappings of minimal distortion. Preprint, 2002.
- [9] E. Bardinet, L.D. Cohen, and N. Ayache. Tracking and motion analysis of the left ventricle with deformable superquadrics. *Medical Image Analysis*, 1(2):129–149, 1996.
- [10] E. Bardinet, L.D. Cohen, and N. Ayache. A parametric deformable model to fit unstructured 3D data. *Computer Vision and Image Understanding*, 71(1):39–54, 1998.
- [11] R. Barret, M. Berry, T.F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. van der Vorst. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*. SIAM, 1994.

- [12] J.L. Barron, D.J. Fleet, and S.S. Beauchemin. Performance of optical flow techniques. *The International Journal of Computer Vision*, 12(1):43–77, 1994.
- [13] M. Bertalmío, L.T. Cheng, S. Osher, and G. Sapiro. Variational problems and partial differential equations on implicit surfaces. *Journal of Computational Physics*, 174(2):759–780, 2001.
- [14] M. Bertalmío, G. Sapiro, and G. Randall. Region tracking on level-sets methods. *IEEE Transactions on Medical Imaging*, 18(5):448–451, 1999.
- [15] G. Bertrand. Simple points, topological numbers and geodesic neighborhoods in cubic grids. *Pattern Recognition Letters*, 15(10):1003–1011, 1994.
- [16] G. Bertrand. A Boolean characterization of three-dimensional simple points. *Pattern Recognition Letters*, 17(2):115–124, 1996.
- [17] J. Besag. On the statistical analysis of dirty pictures (with discussion). *Journal of Royal Statistical Society*, 48(3):259–302, 1986.
- [18] Y. Boykov and V. Kolmogorov. Computing geodesics and minimal surfaces via graph cuts. In *International Conference on Computer Vision*, volume 1, pages 26–33, 2003.
- [19] A. Broadhurst, T.W. Drummond, and R. Cipolla. A probabilistic framework for space carving. In *International Conference on Computer Vision*, volume 1, pages 388–393, 2001.
- [20] P. Burchard, L.-T. Cheng, B. Merriman, and S. Osher. Motion of curves in three spatial dimensions using a level set approach. *Journal of Computational Physics*, 170(2):720–741, 2001.
- [21] R.L. Carceroni and K.N. Kutulakos. Multi-view scene capture by surfel sampling: From video streams to non-rigid 3D motion, shape and reflectance. *The International Journal of Computer Vision*, 49(2–3):175–214, 2002.
- [22] G.J. Carman, H.A. Drury, and D.C. Van Essen. Computational methods for reconstructing and unfolding the cerebral cortex. *Cerebral Cortex*, 5(6):506–517, 1995.
- [23] V. Caselles, F. Catte, T. Coll, and F. Dibos. A geometric model for active contours. *Numerische Mathematik*, 66:1–31, 1993.
- [24] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *The International Journal of Computer Vision*, 22(1):61–79, 1997.
- [25] T. Chan and L. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, February 2001.
- [26] G. Charpiat, O. Faugeras, and R. Keriven. Approximations of shape metrics and application to shape warping and empirical shape statistics. *Foundations of Computational Mathematics*, 5(1):1–58, February 2005.

- [27] G. Charpiat, R. Keriven, J.-P. Pons, and O. Faugeras. Designing spatially coherent minimizing flows for variational problems based on active contours. In *International Conference on Computer Vision*, 2005. to appear.
- [28] Y. Chen, H. Tagare, S. Thiruvankadam, F. Huang, D. Wilson, K. Gopinath, R. Briggs, and E. Geiser. Using prior shapes in geometric active contours in a variational framework. *The International Journal of Computer Vision*, 50(3):315–328, 2002.
- [29] David L. Chopp. Computing minimal surfaces via level set curvature flow. *Journal of Computational Physics*, 106:77–91, 1993.
- [30] I. Cohen and L.D. Cohen. A hybrid hyperquadric model for 2-D and 3-D data fitting. *Computer Vision and Image Understanding*, 63(3):527–541, 1996.
- [31] L.D. Cohen. On active contour models and balloons. *CVGIP: Image Understanding*, 53:211–218, 1991.
- [32] L.D. Cohen and I. Cohen. Finite-element methods for active contour models and balloons for 2-D and 3-D images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1131–1147, November 1993.
- [33] T. Cootes, C. Taylor, D. Cooper, and J. Graham. Active shape models-their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.
- [34] M.G. Crandall and P.-L. Lions. Viscosity solutions of Hamilton–Jacobi equations. *Trans. AMS*, 277:1–43, 1983.
- [35] D. Cremers, T. Kohlberger, and C. Schnörr. Shape statistics in kernel space for variational image segmentation. *Pattern Recognition*, 36(9):1929–1943, September 2003. Special Issue on Kernel and Subspace Methods in Computer Vision.
- [36] A.M. Dale and M.I. Sereno. Improved localization of cortical activity by combining EEG and MEG with MRI cortical surface reconstruction: A linear approach. *Journal of Cognitive Neuroscience*, 5(2):162–176, 1993.
- [37] C.A. Davatzikos and R.N. Bryan. Using a deformable model to obtain a shape representation of the cortex. *IEEE Transactions on Medical Imaging*, 15(6):785–795, 1996.
- [38] M.C. Delfour and J.-P. Zolésio. *Intrinsic Differential Geometry and Theory of Thin Shells*. Quaderni, Scuola Normale Superiore (Pisa, Italy), 2000.
- [39] H. Delingette. General object reconstruction based on simplex meshes. *The International Journal of Computer Vision*, 32(2):111–146, 1999.
- [40] H. Delingette and J. Montagnat. Shape and topology constraints on parametric active contours. *Computer Vision and Image Understanding*, 83(2):140–171, 2001.

- [41] A. Dervieux and F. Thomasset. A finite element method for the simulation of Rayleigh-Taylor instability. *Lecture Notes in Mathematics*, 771:145–159, 1979.
- [42] A. Dervieux and F. Thomasset. Multifluid incompressible flows by a finite element method. *Lecture Notes in Physics*, 11:158–163, 1981.
- [43] M. P. DoCarmo. *Differential Geometry of Curves and Surfaces*. Prentice-Hall, 1976.
- [44] F. Dornaika and R. Chung. Binocular matching constraints from motion. *Pattern Recognition*, 35(10):2003–2012, 2002.
- [45] Y. Duan, L. Yang, H. Qin, and D. Samaras. Shape reconstruction from 3D and 2D data using PDE-based deformable surfaces. In *European Conference on Computer Vision*, volume 3, pages 238–251, 2004.
- [46] P. Dupuis, U. Grenander, and M. Miller. Variational problems on flows of diffeomorphisms for image matching. *Quarterly of Applied Math.*, 56:587–600, 1998.
- [47] G. Egnal. Mutual information as a stereo correspondence measure. Technical Report MS-CIS-00-20, University of Pennsylvania, 2000.
- [48] O. Faugeras, G. Adde, G. Charpiat, C. Chefd’Hotel, M. Clerc, T. Deneux, R. Deriche, G. Hermosillo, R. Keriven, P. Kornprobst, J. Kybic, C. Lenglet, L. Lopez-Perez, T. Papadopoulo, J.-P. Pons, F. Ségonne, B. Thirion, D. Tschumperlé, T. Viéville, and N. Wotawa. Variational, geometric, and statistical methods for modeling brain anatomy and function. *NeuroImage*, 23S1:S46–S55, 2004. Special issue: Mathematics in Brain Imaging - Edited by P.M. Thompson, M.I. Miller, T. Ratnanather, R.A. Poldrack and T.E. Nichols.
- [49] O. Faugeras and G. Hermosillo. Well-posedness of two non-rigid multimodal image registration methods. *Siam Journal of Applied Mathematics*, 64(5):1550–1587, 2004.
- [50] O. Faugeras and R. Keriven. Variational principles, surface evolution, PDE’s, level set methods and the stereo problem. *IEEE Transactions on Image Processing*, 7(3):336–344, 1998.
- [51] B. Fischl, A. Liu, and A.M. Dale. Automated manifold surgery: Constructing geometrically accurate and topologically correct models of the human cerebral cortex. *IEEE Transactions on Medical Imaging*, 20(1):70–80, 2001.
- [52] B. Fischl, M.I. Sereno, and A.M. Dale. Cortical surface-based analysis II : Inflation, flattening, and a surface-based coordinate system. *Neuroimage*, 9(2):195–207, 1999.
- [53] P. Fua and Y.G. Leclerc. Object-centered surface reconstruction: Combining multi-image stereo and shading. *The International Journal of Computer Vision*, 16(1):35–56, 1995.

- [54] M. Gage and R.S. Hamilton. The heat equation shrinking convex plane curves. *J. of Differential Geometry*, 23:69–96, 1986.
- [55] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.
- [56] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky. Cortex segmentation: a fast variational geometric approach. *IEEE Transactions on Medical Imaging*, 21(12):1544–1551, 2002.
- [57] B. Goldlücke and M. Magnor. Space-time isosurface evolution for temporally coherent 3D reconstruction. In *International Conference on Computer Vision and Pattern Recognition*, volume 1, pages 350–355, 2004.
- [58] B. Goldlücke and M. Magnor. Weighted minimal hypersurfaces and their applications in computer vision. In *European Conference on Computer Vision*, volume 2, pages 366–378, 2004.
- [59] J. Gomes. *Implicit representations of evolving manifolds in computer vision*. PhD thesis, Université de Nice Sophia-Antipolis, 2001.
- [60] J. Gomes and O. Faugeras. Reconciling distance functions and level sets. *Journal of Visual Communication and Image Representation*, 11(2):209–223, 2000.
- [61] X. Gu, Y. Wang, T.F. Chan, P.M. Thompson, and S.-T. Yau. Genus zero surface conformal mapping and its application to brain surface mapping. *IEEE Transactions on Medical Imaging*, 23(7):949–958, 2004.
- [62] X. Han, D.L. Pham, D. Tosun, M.E. Rettmann, C. Xu, and J.L. Prince. CRUISE: Cortical reconstruction using implicit surface evolution. *NeuroImage*, 23(3):997–1012, 2004.
- [63] X. Han, C. Xu, U. Braga-Neto, and J.L. Prince. Topology correction in brain cortex segmentation using a multiscale, graph-based algorithm. *IEEE Transactions on Medical Imaging*, 21(2):109–121, 2002.
- [64] X. Han, C. Xu, and J.L. Prince. A topology preserving level set method for geometric deformable models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(6):755–768, 2003.
- [65] A. Hatcher. *Algebraic Topology*. Cambridge University Press, 2002.
- [66] K. Held, E.R. Kops, B.J. Krause, W.M. Wells, and R. Kikinis. Markov random field segmentation of brain MR images. *IEEE Transactions on Medical Imaging*, 16(6):878–886, 1997.

- [67] G. Hermosillo, C. Chéfd'hotel, and O. Faugeras. Variational methods for multimodal image matching. *The International Journal of Computer Vision*, 50(3):329–343, 2002.
- [68] G. Hermosillo, O. Faugeras, and J. Gomes. Unfolding the cerebral cortex using level set methods. In *Scale-Space Theories in Computer Vision*, pages 58–69, 1999.
- [69] C. Hernández Esteban and F. Schmitt. Silhouette and stereo fusion for 3D object modeling. *Computer Vision and Image Understanding*, 96(3):367–392, 2004.
- [70] X. Huang, D. Metaxas, and T. Chen. Metamorphs: Deformable shape and texture models. In *International Conference on Computer Vision and Pattern Recognition*, volume 1, pages 496–503, 2004.
- [71] S. Jehan-Besson, M. Barlaud, and G. Aubert. DREAM2S: Deformable regions driven by an Eulerian accurate minimization method for image and video segmentation. *The International Journal of Computer Vision*, 53(1):45–70, 2003.
- [72] G.-S. Jiang and D. Peng. Weighted ENO schemes for Hamilton-Jacobi equations. *SIAM Journal of Scientific Computing*, 21(6):2126–2143, 2000.
- [73] G.-S. Jiang and C.-W. Shu. Efficient implementation of weighted ENO schemes. *Journal of Computational Physics*, 126(1):202–228, 1996.
- [74] H. Jin, D. Cremers, A.J. Yezzi, and S. Soatto. Shedding light on stereoscopic segmentation. In *International Conference on Computer Vision and Pattern Recognition*, volume 1, pages 36–42, 2004.
- [75] H. Jin, S. Soatto, and A. Yezzi. Multi-view stereo reconstruction of dense shape and complex appearance. *The International Journal of Computer Vision*, 63(3):175–189, 2005.
- [76] T. Kanade and M. Okutomi. A stereo matching algorithm with an adaptive window: Theory and experiment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9):920–932, September 1994.
- [77] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *The International Journal of Computer Vision*, 1(4):321–331, 1987.
- [78] S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and A. Yezzi. Gradient flows and geometric active contour models. In *Proceedings of the 5th International Conference on Computer Vision*, pages 810–815, Boston, MA, June 1995. IEEE Computer Society Press.
- [79] J. Kim, V. Kolmogorov, and R. Zabih. Visual correspondence using energy minimization and mutual information. In *International Conference on Computer Vision*, volume 2, pages 1033–1040, 2003.

- [80] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. In *European Conference on Computer Vision*, volume 3, pages 82–96, 2002.
- [81] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):147–159, 2004.
- [82] N. Kriegeskorte and R. Goebel. An efficient algorithm for topologically correct segmentation of the cortical sheet in anatomical MR volumes. *NeuroImage*, 14(2):329–346, 2001.
- [83] K.N. Kutulakos. Approximate N-view stereo. In *European Conference on Computer Vision*, volume 1, pages 67–83, 2000.
- [84] K.N. Kutulakos and S.M. Seitz. A theory of shape by space carving. *The International Journal of Computer Vision*, 38(3):199–218, 2000.
- [85] J.-O. Lachaud and A. Montanvert. Deformable meshes with automated topology changes for coarse-to-fine 3D surface extraction. *Medical Image Analysis*, 3(2):187–207, 1999.
- [86] M. Leventon, E. Grimson, and O. Faugeras. Statistical Shape Influence in Geodesic Active Contours. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 316–323, Hilton Head Island, South Carolina, June 2000. IEEE Computer Society.
- [87] R. LeVeque. High-resolution conservative algorithms for advection in incompressible flow. *SIAM Journal of Numerical Analysis*, 33(2):627–665, 1996.
- [88] M. Lhuillier and L. Quan. Surface reconstruction by integrating 3D and 2D data of multiple views. In *International Conference on Computer Vision*, volume 2, pages 1313–1320, 2003.
- [89] W.E. Lorensen and H.E. Cline. Marching cubes: A high-resolution 3D surface reconstruction algorithm. *ACM Computer Graphics*, 21(4):163–170, 1987.
- [90] L.M. Lorigo, O.D. Faugeras, W.E.L. Grimson, R. Keriven, R. Kikinis, A. Nabavi, and C.-F. Westin. CURVES: Curve evolution for vessel segmentation. *Medical Image Analysis*, 5(3):195–206, 2001.
- [91] D. MacDonald, N. Kabani, D. Avis, and A.C. Evans. Automated 3-D extraction of inner and outer surfaces of cerebral cortex from MRI. *NeuroImage*, 12(3):340–356, 2000.
- [92] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens. Multimodality image registration by maximization of mutual information. *IEEE Transactions on Medical Imaging*, 16(2):187–198, 1997.

- [93] S. Malassiotis and M.G. Strintzis. Model based joint motion and structure estimation from stereo images. *Computer Vision and Image Understanding*, 65(1):79–94, 1997.
- [94] R. Malladi, J.A. Sethian, and B.C. Vemuri. Shape modeling with front propagation: A level set approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2):158–175, February 1995.
- [95] J.-F. Mangin, V. Frouin, I. Bloch, J. Regis, and J. Lopez-Krahe. From 3D magnetic resonance imaging to structural representations of the cortex topography using topology preserving deformations. *Journal of Mathematical Imaging and Vision*, 5:297–318, 1995.
- [96] T. McInerney and D. Terzopoulos. A dynamic finite element surface model for segmentation and tracking in multidimensional medical images with application to cardiac 4D image analysis. *Computerized Medical Imaging and Graphics*, 19(1):69–83, 1995.
- [97] T. McInerney and D. Terzopoulos. Deformable models in medical image analysis: a survey. *Medical Image Analysis*, 1(2):91–108, 1996.
- [98] T. McInerney and D. Terzopoulos. Topology adaptive deformable surfaces for medical image volume segmentation. *IEEE Transactions on Medical Imaging*, 18(10):840–850, 1999.
- [99] T. McInerney and D. Terzopoulos. T-snakes: Topology adaptive snakes. *Medical Image Analysis*, 4(2):73–91, 2000.
- [100] F. Mémoli, G. Sapiro, and S. Osher. Solving variational problems and partial differential equations mapping into general target manifolds. *Journal of Computational Physics*, 195(1):263–292, 2004.
- [101] D.N. Metaxas and D. Terzopoulos. Shape and nonrigid motion estimation through physics-based synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(6):580–591, 1993.
- [102] P.W. Michor and D. Mumford. Riemannian geometries of space of plane curves. Preprint, 2005.
- [103] K. Mikula and D. Sevcovic. Evolution of plane curves driven by a nonlinear function of curvature and anisotropy. *SIAM Journal of Applied Mathematics*, 61(5):1473–1501, 2001.
- [104] J.V. Miller, D.E. Breen, W.E. Lorensen, R.M. O’Bara, and M.J. Wozny. Geometrically deformed models: a method for extracting closed geometric models from volume data. In *SIGGRAPH*, pages 217–226, 1991.

- [105] J. Montagnat, H. Delingette, and N. Ayache. A review of deformable surfaces: Topology, geometry and deformation. *Computer Vision and Image Understanding*, 19(14):1023–1040, 2001.
- [106] J. Montagnat, H. Delingette, N. Scapel, and N. Ayache. Representation, shape, topology and evolution of deformable surfaces. Application to 3D medical image segmentation. Technical Report 3954, INRIA, 2000.
- [107] J. Moser. On the volume elements on a manifold. *AMS Transactions*, 120(2):286–294, 1965.
- [108] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42:577–684, 1989.
- [109] J. Neumann and Y. Aloimonos. Spatio-temporal stereo using multi-resolution subdivision surfaces. *The International Journal of Computer Vision*, 47:181–193, 2002.
- [110] S. Osher and R. Fedkiw. *The Level Set Method and Dynamic Implicit Surfaces*. Springer-Verlag, 2002.
- [111] S. Osher and N. Paragios, editors. *Geometric Level Set Methods in Imaging, Vision and Graphics*. Springer Verlag, 2003.
- [112] S. Osher and J.A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations. *Journal of Computational Physics*, 79(1):12–49, 1988.
- [113] N.C. Overgaard and J.E. Solem. An analysis of variational alignment of curves in images. In *International Conference on Scale Space and PDE Methods in Computer Vision*, pages 480–491, 2005.
- [114] N. Paragios and R. Deriche. Geodesic active regions and level set methods for supervised texture segmentation. *The International Journal of Computer Vision*, 46(3):223–247, 2002.
- [115] N. Paragios and R. Deriche. Geodesic active regions and level set methods for motion estimation and tracking. *Computer Vision and Image Understanding*, 97(3):259–282, 2005.
- [116] J.-Y. Park, T. McInerney, D. Terzopoulos, and M.-H. Kim. A non-self-intersecting adaptive deformable surface for complex boundary extraction from volumetric images. *Computer and Graphics*, 25(3):412–440, 2001.
- [117] E. Parzen. On estimation of a probability density function and mode. *Annals Mathematical Statistics*, 33:1065–1076, 1962.

- [118] D. Peng, B. Merriman, S. Osher, H.-K. Zhao, and M. Kang. A PDE-based fast local level set method. *Journal of Computational Physics*, 155(2):410–438, 1999.
- [119] J.-P. Pons, G. Hermosillo, R. Keriven, and O. Faugeras. How to deal with point correspondences and tangential velocities in the level set framework. In *International Conference on Computer Vision*, volume 2, pages 894–899, 2003.
- [120] J.-P. Pons, G. Hermosillo, R. Keriven, and O. Faugeras. How to deal with point correspondences and tangential velocities in the level set framework. Technical Report 4857, INRIA, 2003.
- [121] J.-P. Pons, G. Hermosillo, R. Keriven, and O. Faugeras. Maintaining the point correspondence in the level set framework. *Journal of Computational Physics*, 2005. submitted.
- [122] J.-P. Pons, R. Keriven, and O. Faugeras. Area preserving cortex unfolding. In *International Conference on Medical Image Computing and Computer Assisted Intervention*, volume 1, pages 376–383, 2004.
- [123] J.-P. Pons, R. Keriven, and O. Faugeras. Modelling dynamic scenes by registering multi-view image sequences. Technical Report 5321, INRIA, 2004.
- [124] J.-P. Pons, R. Keriven, and O. Faugeras. Modelling dynamic scenes by registering multi-view image sequences. In *International Conference on Computer Vision and Pattern Recognition*, volume 2, pages 822–827, 2005.
- [125] J.-P. Pons, R. Keriven, and O. Faugeras. Modelling non-rigid dynamic scenes from multi-view image sequences. In N. Paragios, Y. Chen, and O. Faugeras, editors, *The Handbook of Mathematical Models in Computer Vision*, chapter 27. Springer, 2005.
- [126] J.-P. Pons, R. Keriven, O. Faugeras, and G. Hermosillo. Variational stereovision and 3D scene flow estimation with statistical similarity measures. In *International Conference on Computer Vision*, volume 2, pages 597–602, 2003.
- [127] JP. Pons, R. Keriven, and O. Faugeras. *Mathematical Models in Computer Vision: The Handbook*, chapter Modelling Non-Rigid Dynamic Scenes from Multi-View Image Sequences. Springer, 2005.
- [128] J. Rauch. *Partial Differential Equations*. Springer-Verlag, New York, 1991.
- [129] A. Roche, G. Malandain, X. Pennec, and N. Ayache. The correlation ratio as new similarity metric for multimodal image registration. In W.M. Wells, A. Colchester, and S. Delp, editors, *Medical Image Computing and Computer-Assisted Intervention-MICCAI'98*, number 1496 in Lecture Notes in Computer Science, pages 1115–1124, Cambridge, MA, USA, October 1998. Springer.

- [130] Mikael Rousson and Rachid Deriche. Adaptive segmentation of vector valued images. In S. Osher and N. Paragios, editors, *Geometric Level Set Methods in Imaging Vision and Graphics*. Springer, 2003.
- [131] Mikael Rousson and Nikos Paragios. Shape priors for level set representations. In A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, editors, *Proceedings of the 7th European Conference on Computer Vision*, volume 2, pages 78–92, Copenhagen, Denmark, May 2002. Springer-Verlag.
- [132] P. Saint-Marc, H. Rom, and G.G. Medioni. B-spline contour representation and symmetry detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1191–1197, 1993.
- [133] D. Scharstein and R. Szeliski. Stereo matching with nonlinear diffusion. *The International Journal of Computer Vision*, 28(2):155–174, 1998.
- [134] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *The International Journal of Computer Vision*, 47(1):7–42, 2002.
- [135] M. Segal, C. Korobkin, R. van Widenfelt, J. Foran, and P. Haeberli. Fast shadows and lighting effects using texture mapping. *Computer Graphics*, 26(2):249–252, 1992.
- [136] F. Ségonne. *Segmentation of Medical Images under Topological Constraints*. PhD thesis, CSAIL, MIT, 2005. to appear.
- [137] F. Ségonne, A.M. Dale, E. Busa, M. Glessner, D. Salat, H.K. Hahn, and B. Fischl. A hybrid approach to the skull stripping problem in MRI. *NeuroImage*, 22(3):1060–1075, 2004.
- [138] F. Ségonne, J.-P. Pons, F. Fischl, and E. Grimson. A novel active contour framework: Multi-component level set evolution under topology control. AI memos AIM-2005-020, CSAIL, MIT, 2005.
- [139] F. Ségonne, J.-P. Pons, E. Grimson, and B. Fischl. A novel level set framework for the segmentation of medical images under topology control. In *Workshop on Computer Vision for Biomedical Image Applications: Current Techniques and Future Trends*, 2005. to appear.
- [140] S.M. Seitz and C.R. Dyer. Photorealistic scene reconstruction by voxel coloring. *The International Journal of Computer Vision*, 35(2):151–173, 1999.
- [141] M.I. Sereno, A.M. Dale, A. Liu, and R.B.H. Tootell. A surface-based coordinate system for a canonical cortex. *NeuroImage*, 3(3):S252, 1996.
- [142] J.A. Sethian. A fast marching level set method for monotonically advancing fronts. In *Proceedings of the National Academy of Sciences*, volume 93, pages 1591–1694, 1996.

- [143] J.A. Sethian. *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Sciences*. Cambridge Monograph on Applied and Computational Mathematics. Cambridge University Press, 1999.
- [144] Y.Q. Shi, C.Q. Shu, and J.N. Pan. Unified optical flow field approach to motion analysis from a sequence of stereo images. *Pattern Recognition*, 27(12):1577–1590, 1994.
- [145] C.-W. Shu. Total-variation-diminishing time discretizations. *SIAM Journal of Scientific and Statistical Computing*, 9(6):1073–1084, 1988.
- [146] K. Siddiqi, Y. Lauzière, A. Tannenbaum, and S. Zucker. Area and length minimizing flows for shape segmentation. *IEEE Transactions on Image Processing*, 7(3):433–443, 1998.
- [147] G.G. Slabaugh and G.B. Unal. Active polyhedron: Surface evolution theory applied to deformable meshes. In *International Conference on Computer Vision and Pattern Recognition*, volume 2, pages 84–91, 2005.
- [148] S. Soatto, A.J. Yezzi, and H. Jin. Tales of shape and radiance in multi-view stereo. In *International Conference on Computer Vision*, volume 2, pages 974–981, 2003.
- [149] J.E. Solem and A. Heyden. Reconstructing open surfaces from unorganized data points. In *International Conference on Computer Vision and Pattern Recognition*, volume 2, pages 653–660, 2004.
- [150] J.E. Solem and N.C. Overgaard. A geometric formulation of gradient descent for variational problems with moving surfaces. In *International Conference on Scale Space and PDE Methods in Computer Vision*, pages 419–430, 2005.
- [151] L.H. Staib and J.S. Duncan. Boundary finding with parametrically deformable models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(11):1061–1075, 1992.
- [152] G. Storvik. A Bayesian approach to dynamic contours through stochastic sampling and simulated annealing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(10):976–986, 1994.
- [153] C. Strecha, R. Fransens, and L. Van Gool. Wide-baseline stereo from multiple views: a probabilistic account. In *International Conference on Computer Vision and Pattern Recognition*, volume 2, pages 552–559, 2004.
- [154] C. Strecha, T. Tuytelaars, and L. Van Gool. Dense matching of multiple wide-baseline views. In *International Conference on Computer Vision*, volume 2, pages 1194–1201, 2003.

- [155] C. Strecha and L. Van Gool. Motion-stereo integration for depth estimation. In *European Conference on Computer Vision*, volume 2, pages 170–185, 2002.
- [156] M. Sussman, P. Smereka, and S. Osher. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics*, 114(1):146–159, 1994.
- [157] R. Szeliski. Bayesian modeling of uncertainty in low-level vision. *The International Journal of Computer Vision*, 5(3):271–301, 1990.
- [158] R. Szeliski. Prediction error as a quality metric for motion and stereo. In *International Conference on Computer Vision*, volume 2, pages 781–788, 1999.
- [159] R. Szeliski and D. Tonnesen. Surface modeling with oriented particle systems. In *SIGGRAPH*, pages 185–194, 1992.
- [160] G. Taubin, F. Cukierman, S. Sullivan, J. Ponce, and D.J. Kriegman. Parameterized families of polynomials for bounded algebraic curve and surface fitting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(3):287–303, March 1994.
- [161] D. Terzopoulos and D.N. Metaxas. Dynamic 3D models with local and global deformations: Deformable superquadrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(7):703–714, 1991.
- [162] A. Treuille, A. Hertzmann, and S.M. Seitz. Example-based stereo with general BRDFs. In *European Conference on Computer Vision*, volume 2, pages 457–469, 2004.
- [163] A. Trouvé. Diffeomorphisms groups and pattern matching in image analysis. *The International Journal of Computer Vision*, 28(3):213–21, 1998.
- [164] A. Tsai, A.J. Yezzi, W. Wells, C. Tempany, D. Tucker, A. Fan, W.E. Grimson, and A. Willsky. A shape-based approach to the segmentation of medical imagery using level sets. *IEEE Transactions on Medical Imaging*, 22(2):137–154, 2003.
- [165] A. Tsai, A.J. Yezzi, and A.S. Willsky. Curve evolution implementation of the Mumford-Shah functional for image segmentation, denoising, interpolation, and magnification. *IEEE Transactions on Image Processing*, 10(8):1169–1186, 2001.
- [166] S. Vedula, S. Baker, and T. Kanade. Spatio-temporal view interpolation. In *ACM Eurographics Workshop on Rendering*, pages 65–76, 2002.
- [167] S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade. Three-dimensional scene flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):475–480, 2005.

- [168] S. Vedula, S. Baker, S. Seitz, and T. Kanade. Shape and motion carving in 6D. In *International Conference on Computer Vision and Pattern Recognition*, volume 2, pages 592–598, 2000.
- [169] B.C. Vemuri, J. Ye, Y. Chen, and Leonard C.M. A level-set based approach to image registration. In *IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, pages 86–93, 2000.
- [170] L.A. Vese and T. Chan. A multiphase level set framework for image segmentation using the Mumford and Shah model. *The International Journal of Computer Vision*, 50(3):271–293, 2002.
- [171] P. Viola and W. M. Wells III. Alignment by maximization of mutual information. *The International Journal of Computer Vision*, 24(2):137–154, 1997.
- [172] W. Wang and J.H. Duncan. Recovering the three-dimensional motion and structure of multiple moving objects from binocular image flows. *Computer Vision and Image Understanding*, 63(3):430–446, 1996.
- [173] W.M. Wells, E.L. Grimson, R. Kikinis, and F.A. Jolesz. Adaptive segmentation of MRI data. *IEEE Transactions on Medical Imaging*, 15(4):429–442, 1996.
- [174] C. Xu, D.L. Pham, and J.L. Prince. Medical image segmentation using deformable models. In J.M. Fitzpatrick and M. Sonka, editors, *Handbook of Medical Imaging*, volume 1, chapter 3, pages 129–174. SPIE Press, 2000.
- [175] C. Xu, D.L. Pham, M.E. Rettmann, D.N. Yu, and J.L. Prince. Reconstruction of the human cerebral cortex from magnetic resonance images. *IEEE Transactions on Medical Imaging*, 18(6):467–480, 1999.
- [176] C. Xu and J.L. Prince. Snakes, shapes, and gradient vector flow. *IEEE Transactions on Image Processing*, 7(3):359–369, 1998.
- [177] J.-J. Xu and H.-K. Zhao. An Eulerian formulation for solving partial differential equations along a moving interface. *Journal of Scientific Computing*, 19:573–594, 2003.
- [178] R. Yang, M. Pollefeys, and G. Welch. Dealing with textureless regions and specular highlights: A progressive space carving scheme using a novel photo-consistency measure. In *International Conference on Computer Vision*, volume 1, pages 576–584, 2003.
- [179] A. Yezzi, S. Kichenassamy, A. Kumar, P. Olver, and A. Tannenbaum. A geometric snake model for segmentation of medical imagery. *IEEE Transactions on Medical Imaging*, 16(2):199–209, April 1997.
- [180] A.J. Yezzi and A.C.G. Mennucci. Metrics in the space of curves. Preprint, 2005.

- [181] A.J. Yezzi and S. Soatto. Deformation: Deforming motion, shape average and the joint registration and approximation of structures in images. *The International Journal of Computer Vision*, 53(2):153–167, 2003.
- [182] A.J. Yezzi and S. Soatto. Stereoscopic segmentation. *The International Journal of Computer Vision*, 53(1):31–43, 2003.
- [183] L. Younes. Computable elastic distances between shapes. *SIAM Journal of Applied Mathematics*, 58(2):565–586, 1998.
- [184] X. Zeng, L.H. Staib, R.T. Schultz, and J.S. Duncan. Segmentation and measurement of the cortex from 3D MR images using coupled surfaces propagation. *IEEE Transactions on Medical Imaging*, 18(10):100–111, 1999.
- [185] Y. Zhang, M. Brady, and S. Smith. Segmentation of brain MR images through a hidden Markov random field model and the expectation-maximization algorithm. *IEEE Transactions on Medical Imaging*, 20(1), 2001.
- [186] Y. Zhang and C. Kambhamettu. On 3-D scene flow and structure recovery from multiview image sequences. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 33(4):592–606, 2003.
- [187] H.-K. Zhao, T. Chan, B. Merriman, and S. Osher. A variational level set approach to multiphase motion. *Journal of Computational Physics*, 127(1):179–195, 1996.
- [188] H.-K. Zhao, S. Osher, B. Merriman, and M. Kang. Implicit and non-parametric shape reconstruction from unorganized points using a variational level set method. *Computer Vision and Image Understanding*, 80(3):295–314, 2000.
- [189] L. Zhu, S. Haker, and A. Tannenbaum. Area-preserving mappings for the visualization of medical structures. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, volume 2, pages 277–284, 2003.
- [190] S. Zhu and A. Yuille. Region competition: unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(9):884–900, September 1996.
- [191] K. Zilles. Cortex. In G. Paxinos, editor, *The Human Nervous System*, pages 757–802. Academic Press, 1990.

