

Ministère de l'enseignement supérieur et de la recherche scientifique

وزارة التعليم العالي والبحث العلمي

Université Badji Mokhtar Annaba
Badji Mokhtar University - Annaba



جامعة باجي مختار عنابة

Faculté des Sciences de L'ingéniorat
Département d'informatique

Année : 2014

THÈSE

Présentée en vue de l'obtention du diplôme
de doctorat troisième cycle LMD

Intitulée :

Le paradigme multi-agent de la segmentation d'images de profondeurs

Option

Traitement d'images pour la vision artificielle

Par

KHALDI Amine

Devant le jury

Pr GHOUALMI Nacira	Professeure à l'université d'Annaba	Président
Pr MEROUANI Hayet Farida	Professeure à l'université d'Annaba	Directeur de thèse
Pr SERIDI Hamid	Professeur à l'université de Guelma	Examineur
Pr TLILI Yamina	Professeure à l'université d'Annaba	Examineur
Dr BENBLIDIA Nadjia	Maitre de conférence à l'université de Blida	Examineur

Dédicaces

Afin d'être reconnaissant envers ceux qui m'ont appuyé et encouragé à effectuer ce travail de recherche, je dédie cette thèse :

À ma très chère mère et à ma femme pour leur soutien moral et pour tous les sentiments d'affection et d'amour qui représentent pour moi le pilier de tous mes efforts.

À ma très chère tante décédée récemment qui m'a toujours encourager et cru en moi

À tous les membres de ma famille sans aucune exception.

Je dédie cette thèse à tous ceux qui m'ont soutenu avant et pendant la réalisation de ce travail.

Remerciements

A l'issue de la rédaction de cette recherche, je suis convaincue que la thèse est loin d'être un travail solitaire. En effet, je n'aurais jamais pu réaliser ce travail doctoral sans le soutien d'un grand nombre de personnes dont la générosité, la bonne humeur et l'intérêt manifestés à l'égard de ma recherche m'ont permis de progresser dans cette phase délicate de «l'apprenti chercheur».

Je tiens tout d'abord à remercier le directeur de cette thèse Mme Merouani Farida Hayet professeur à l'université Badji Mokhtar d'Annaba qui m'a encadrée durant ces cinq années. Je lui exprime toute ma gratitude, car elle a su me conseiller et me diriger avec ses remarques pertinentes. Je la remercie pour ses qualités humaines ; sa grande disponibilité, sa patience, sa gentillesse, ses précieux conseils, et de m'avoir fait bénéficier de sa haute compétence professionnelle.

J'aimerais également lui dire à quel point j'ai apprécié sa grande disponibilité et son respect sans faille des délais serrés de relecture des documents que je lui ai adressés. Je ne sais comment exprimer ma gratitude à cette personne autrement qu'en lui promettant d'agir comme elle avec des étudiants dans ma situation, si un jour l'occasion m'en est donnée.

Mes remerciements vont à Mme Ghoulmi Nacira, Professeure à l'université d'Annaba, je la remercie très sincèrement du grand honneur qu'elle m'a fait en présidant mon jury et d'avoir accepté de juger mon travail.

Je tiens aussi à remercier Mr. Seridi Hamid, Professeur à l'université de Guelma ainsi que Mme Benblidia Nadjia Maître de conférence à l'université de Blida qui ont accepté d'examiner mon travail et d'avoir accepté de faire le déplacement jusqu'à Annaba.

Je remercie également Mme Tlili Yamina professeure à l'université d'Annaba pour l'honneur qu'elle m'a fait en acceptant de juger ce travail. Je suis honoré de l'attention et au temps qu'elle a consacré à l'examinations de ce travail

*Mes remerciements son également exprimés à mes chers parents, mes frères et ma femme pour toute l'aide qu'ils m'ont apporté au cours de mes études.
Je remercie aussi tous mes amis*

Résumé

Ce travail de thèse propose une étude approfondie des images de profondeurs ainsi que les méthodes de segmentation les plus utilisées dans le domaine de la segmentation de profondeur. Dans ce travail nous dressons une classification des méthodes de segmentation de profondeur. Cette classification est faite accordement aux critères d'homogénéités auquel obéissent les méthodes. Ce travail comporte une deuxième partie où une application de trois modèles déformables Contour actif classique, contour actif adaptatif ainsi que la méthode des Level Set. Ces trois méthodes sont appliquées pour segmenter des images de profondeurs, les résultats numériques obtenus y sont comparés afin de trouver la meilleure technique de modèles déformables adapté à la segmentation d'images de profondeurs. Pour finir nous proposons une approche multi-agent pour segmenter les images de profondeur.

Mots-clés : *Image de profondeur, Segmentation, Contour actif, Contour adaptatif, Level Set, Système multi-agent.*

ملخص

هذه الأطروحة هي دراسة مفصلة للصور من أعماق وطرق تجزئة الأكثر استخداما في مجال تجزئة من العمق في هذا العمل الذي نقوم بإجراء تصنيف أساليب تجزئة من يتم هذا التصنيف لمعايير ضبط التجانس الأساليب التي تطبع يشمل هذا العمل جزء بالعمق ثان حيث يتم تطبيق نماذج تشوه الصور إلى أعماق شريحة، يتم مقارنة النتائج العددية للعثور على من هو أفضل أسلوب أو طريقة لنماذج تشوه هذ صورة مناسبة لأعماق تجزئة

الكلمات المفتاحية: ضبط التجانس . الصور الرقمية . تشريح حسب العمق

Abstract

This paper considered the use of active contour for an important low-level computer vision problem, namely, range image segmentation. First, we propose a range segmentations method's classification. This classification is done according to the homogeneous criterions of the methods. Then, we presented and compare three methods for edge detection applied to range images based on deformable models, The motivation of our work was to apply two deformable model (Snake) which never been done before in range segmentation. We will apply a classic active contour; Level sets then an adaptive active contour. Finally we propose a multi-agent system approach for segmenting range images.

Keywords: *Range image, Active contour, Level set, Adaptive active contour, Segmentation, Multi-agent System.*

Tables des matières :

<i>Introduction générale</i>	<i>1</i>
Chapitre 1 Analyse et segmentation d'images numériques	6
1.1 Introduction.....	6
1.2 Segmentation d'image basée sur les pixels.....	6
1.2.1 Sélection récursive sur histogrammes.....	6
1.2.2 Détection de vallées	7
1.2.3 Minimisation de la variance	8
1.3 Segmentation basée sur les régions	9
1.3.1 Accroissement de région	9
1.3.2 La méthode du Split and Merge	12
1.3.3 La méthode du CSC (Color Structure Code).....	13
1.4 Segmentation d'image par détection de contour	17
1.4.1 Les méthodes dérivatives du premier ordre.....	18
1.4.2 Les méthodes dérivatives du deuxième ordre	20
1.4.3 Les détecteurs par masquage adapté.....	22
1.4.4 Le filtre de Canny.....	23
1.4.5 Le filtre de Canny - Deriche	24
1.5 Détection de points d'intérêts	25
1.5.1 Le détecteur de Moravec.....	26
1.5.2 Le détecteur de Harris	27
1.5.3 Le détecteur SIFT.....	28
1.6 La poursuite et la fermeture des contours	29
1.6.1 Méthodes de recherche dans des graphes	29
1.6.2 Méthodes de recherche par les automates.....	30
1.7 Conclusion.....	30
Chapitre 2 L'image de profondeur	31
2.1 Introduction.....	31
2.2 Définition.....	31

<i>2.3 Profondeur d'une image</i>	32
<i>2.4 Propriété de cohérence spatiale</i>	35
<i>2.5 Systèmes d'acquisition d'images de profondeur</i>	36
<i>2.6 Types d'images de profondeur</i>	41
<i>2.7 Représentations graphiques d'une image de profondeur</i>	43
<i>2.8 Discontinuités d'une image de profondeur</i>	45
<i>2.8.1 Discontinuité de profondeur (discontinuité d'ordre 0)</i>	46
<i>2.8.2 Discontinuité d'orientation (discontinuité d'ordre 1)</i>	46
<i>2.9 Erreur de mesure d'une image de profondeur</i>	47
<i>2.10 Estimation du bruit dans une image de profondeur</i>	48
<i>2.11 Conclusion</i>	50
<i>Chapitre 3 Segmentation d'images de profondeurs</i>	51
<i>3.1 Introduction</i>	51
<i>3.2 Segmentation d'images de profondeur</i>	51
<i>3.2.1 Méthodes élémentaires de segmentation d'image de profondeur</i>	51
<i>3.2.1.1 Détection de frontière et étiquetage des composantes connexes</i>	52
<i>3.2.1.2 Seuillage et étiquetage des composantes connexes</i>	52
<i>3.2.1.3 Clustering et étiquetage des composantes connexes</i>	52
<i>3.2.1.4 Fusion de régions</i>	53
<i>3.2.1.5 Croissance indépendante de régions</i>	54
<i>3.2.1.6 Division et fusion de régions</i>	54
<i>3.2.2 Méthodes de segmentation d'images de profondeur</i>	55
<i>3.2.2.1 Segmentation selon le type de courbure</i>	55
<i>3.2.2.2 Segmentation en surface plane</i>	56
<i>3.2.2.3 Segmentation en surfaces algébriques</i>	59
<i>3.2.2.4 Segmentation de continuité d'ordre 1</i>	61
<i>3.2.2.5 Autres méthodes de segmentation</i>	63
<i>3.3 Détection des discontinuités</i>	71
<i>3.3.1 Détecteur de point de discontinuité</i>	71
<i>3.3.2 Détecteurs différentiels de point de discontinuité</i>	72

3.3.2.1 Détecteur simple (sans condition d'extremum).....	73
3.3.2.2 Détecteur avec condition d'extremum.....	73
3.3.3 Grandeur différentielle utile à la détection de point de discontinuité....	74
3.3.3.1 Mesures de discontinuité.....	74
3.3.3.2 Estimation des grandeurs différentielles	76
3.3.4 Revue des détecteurs de point de discontinuité.....	79
3.3.4.1 Détecteurs sans condition d'extremum	79
3.3.4.2 Détecteurs avec condition d'extremum	79
3.3.5 Détection conjointe des discontinuités (C_0, C_1) à l'aide d'un unique détecteur	81
3.3.5.1 Mesures de discontinuité pour une IP cartésienne orthonormée	81
3.3.5.2 Extension aux autres types d'IP	83
3.4 Conclusion.....	84
Chapitre 4 Les contours actifs.....	85
4.1 Introduction.....	85
4.2 Principe et définitions de la méthode contours actifs.....	85
4.2.1 L'énergie interne du contour (énergie propre).....	87
4.2.2 L'énergie image (énergie potentielle).....	87
4.2.3 L'énergie de contrainte du contour	88
4.2.4 Convergence du contour actif.....	88
4.2.5 Déformation du contour actif.....	88
4.2.6 Limites et améliorations proposées.....	90
4.3 Type de contour actif.....	90
4.3.1 Le modèle ballon	90
4.3.2 Contour actif par la méthode de gradient vector flow (GVF).....	92
4.3.3 Le contour actif ASM (Actif Shape Model).....	94
4.3.4 Les G-snake.....	95
4.3.5 Modèles géométriques.....	96
4.3.6 Méthode d'ensembles de niveau zéro (Level Sets).....	97
4.3.7 Modèles géodésiques.....	99

4.3.8	<i>Le contour actif Dynamique</i>	100
4.3.9	<i>Contours actifs par approche région</i>	101
4.3.10	<i>Modèle particulière</i>	102
4.4	<i>Conclusion</i>	103
Chapitre 5 <i>Les systèmes multi-agents</i>		105
5.1	<i>Introduction</i>	105
5.2	<i>Définitions</i>	105
5.3	<i>Les entités d'un système multi-agent</i>	107
5.4	<i>Définition d'un agent</i>	108
5.5	<i>Les types d'agents</i>	109
5.5.1	<i>Les agents cognitifs</i>	109
5.5.2	<i>Les agents réactifs</i>	110
5.5.3	<i>Les agents hybrides</i>	111
5.6	<i>Définition de l'environnement</i>	111
5.7	<i>Interactions entre agents</i>	112
5.8	<i>Comportement du system</i>	113
5.8.1	<i>Le concept d'émergence</i>	113
5.8.2	<i>L'auto-organisation</i>	114
5.8.3	<i>Le concept de Stigmergie</i>	115
5.9	<i>Segmentation d'image par utilisation des SMA</i>	116
5.9.1	<i>Approches points</i>	117
5.9.2	<i>Approches contours</i>	118
5.9.3	<i>Approches régions</i>	120
5.10	<i>Conclusion</i>	125
Chapitre 6 <i>Classification des méthodes de segmentation d'images de profondeur</i>		126
6.1	<i>Introduction</i>	126
6.2	<i>Classification des méthodes de segmentation</i>	126
6.2.1	<i>Segmentation selon le type de courbure</i>	126
6.2.2	<i>Segmentation sur surface algébrique</i>	127
6.2.3	<i>Segmentation sur surface plane</i>	128

6.2.4 Segmentation de continuité d'ordre 1	128
6.3 Conclusion.....	131
Chapitre 7 Segmentation d'images de profondeur par contour actif.....	132
7.1 Introduction.....	132
7.2 Prétraitement.....	132
7.2.1 Filtre de Floyd & Steinberg	133
7.2.2 Résultats du filtrage	134
7.3 Base d'images utilisé.....	136
7.4 Langage de programmation utilisé.....	136
7.5 Framework de comparaison	136
7.6 Algorithmes utilisés.....	138
7.6.1 Le contour actif classique (Snake).....	138
7.6.2 Le contour actif adaptatif.....	144
7.6.3 Les Level Sets	149
7.6.4 Comparaison et analyse numérique des résultats obtenus.....	153
7.7 Conclusion.....	156
Chapitre 8 Proposition d'une approche multi-agent pour la segmentation d'images de profondeur	157
8.1 Introduction.....	157
8.2 approche multi-agent pour la segmentation de profondeur.....	157
8.2.1 Processus d'initialisation.....	158
8.2.2 Processus de segmentation	158
8.2.3 Processus de communication.....	158
8.3 Conclusion.....	161
Conclusion générale.....	162
Bibliographie.....	165

Liste des figures :

Chapitre 1 : Analyse et segmentation d'images numériques

- Figure 1.1 – Illustration du principe de l'algorithme de sélection récursive sur histogrammes
- Figure 1.2 – Résultat de la segmentation par détection de vallées
- Figure 1.3 – Résultat de la segmentation par minimisation de la variance
- Figure 1.4 – Accroissement de région : A gauche, seuil=6. A droite, seuil=7
- Figure 1.5 – Deux schémas de regroupement pour des pixels a, b et c
- Figure 1.6 – Découpage par quadtree d'une image 8x8. Traitement avec seuil=100%
- Figure 1.7 – Exemple d'un Région Adjacence Graph
- Figure 1.8 – Exemple d'utilisation de l'algorithme du Split and Merge. A droite, la segmentation avec Split and Merge. A gauche l'image originale.
- Figure 1.9 – Exemple d'îlot de niveau 0 basé sur la topologie hexagonale
- Figure 1.10 – Topologie de l'algorithme CSC en structure hexagonale hiérarchique
- Figure 1.11 – Phase d'initialisation de l'algorithme CSC
- Figure 1.12 – Phase de regroupement de l'algorithme CSC
- Figure 1.13 – Phase de découpage de l'algorithme CSC
- Figure 1.14 – Exemple de segmentation par l'algorithme CSC
- Figure 1.15 – Différents types de contours : marche, toit et pointe
- Figure 1.16 – Fonction d'intensité au voisinage d'un contour, ses dérivées première et seconde
- Figure 1.17 – Les lignes bleues représentent le gradient de couleur du plus clair vers le plus foncé
- Figure 1.18 – L'image originale. L'image traitée par le masque de Sobel, puis par le Gradient
- Figure 1.19 Image originale, Utilisation du filtre de Sobel, Utilisation du filtre de Prewitt.
- Figure 1.20 – Image en niveau de gris, première dérivée de l'image et deuxième dérivée
- Figure 1.21 – Image originale, maxima du gradient puis zéros du Laplacien
- Figure 1.22 – Les fenêtre de Kirsch
- Figure 1.23 – Les fenêtre de Babu et Nevatia
- Figure 1.24 – Résultat du filtre de Canny (image à droite)
- Figure 1.25 – A droite, l'image filtrée avec l'algorithme de Canny-Deriche
- Figure 1.26 – Les différentes situations considérées par le détecteur de Moravec
- Figure 1.27 – Résultats de détection de contour avec Harris puis Sift.

Chapitre 2 : L'image de profondeur

- Figure 2.1 – A gauche une image de profondeur, à droite une image d'intensité
- Figure 2.2 – A gauche une image d'intensité, à droite une image convertie par ondelettes bidimensionnelles
- Figure 2.3 – A gauche une image 2D, à droite une image convertie par carte HOS

- Figure 2.4 – A gauche une image 2D, à droite une image convertie par segmentation hybride
- Figure 2.5 – Illustration de la cohérence spatiale
- Figure 2.6 – Système expérimental d'acquisition d'IP (Université de Neuchâtel)
- Figure 2.7 – Système d'acquisition d'images de profondeur de type Time of Flight
- Figure 2.8 – Système FLRS embarqué dans un ballon
- Figure 2.9 – Image de profondeur acquise par le système FLRS
- Figure 2.10 – Image de profondeur acquise par une stéréo camera
- Figure 2.11 – Système d'acquisition d'images de profondeur Cam Cube, à gauche l'image réelle, à droite l'image de profondeur acquise par Cam Cube
- Figure 2.12 – Acquisition d'images de profondeur. A gauche la scène réelle, à droite l'image de profondeur acquise par le TCSPC
- Figure 2.13 – Image de profondeur brute
- Figure 2.14 – Image de profondeur cartésienne
- Figure 2.15 – Image de profondeur cartésienne orthographique
- Figure 2.16 – Image de profondeur cartésienne orthonormée
- Figure 2.17 – Représentations d'une image de profondeur (a) points 3D (b) fil de fer (c) profils (d) surface neutre.
- Figure 2.18 – Représentation d'une IP (a) rendu réaliste. (b) courbes de niveaux. (c) niveaux de gris (d) pseudo-ombrée.
- Figure 2.19 – Domaine de définition d'une IP et discontinuités de profondeur, (a) vue en coupe de la scène, (b) plan de l'IP, blanc: domaine de définition D_P , traits: courbes $\overline{C^0}$
- Figure 2.20 – Modèle d'objet à facettes, point singulier le long d'une courbe $\overline{C^1}$.
- Figure 2.21 – Différentes interprétations de l'erreur de mesure.

Chapitre 3 : Segmentation d'images de profondeurs

- Figure 3.1 – Segmentation par seuillage de Kasvand, à gauche l'image de profondeur, à droite le résultat de segmentation
- Figure 3.2 – Détecteur de discontinuités de Yokoya et Levine. A gauche l'image de profondeur. A droite l'image segmentée.
- Figure 3.3 – Résultat de segmentation, à gauche l'image de profondeur, à droite l'image segmenté.
- Figure 3.4 – Résultat de segmentation par division-fusion de Taylor, à gauche l'image de profondeur, à droite l'image segmenté.
- Figure 3.5 – Résultat de segmentation. A gauche l'image de profondeur. A droite l'image segmentée.
- Figure 3.6 – Résultat de segmentation, à gauche l'image de profondeur, à droite l'image segmenté.
- Figure 3.7 – Segmentation par supers quadriques de Gupta, à gauche l'image de profondeur, à droite l'image segmentée.
- Figure 3.8 – Segmentation par croissance de régions de Jiang, à gauche l'image de profondeur, à droite l'image segmentée.
- Figure 3.9 – Résultat de segmentation, à gauche l'image de profondeur, à droite l'image segmenté.
- Figure 3.10 – Résultat de segmentation, à gauche l'image de profondeur, à droite l'image segmenté.

- Figure 3.11 – Résultat de segmentation, à gauche l'image de profondeur, à droite l'image segmenté.
- Figure 3.12 – Processus de segmentation automatique de Merchant
- Figure 3.13 – Résultat de segmentation. (a) L'image de profondeur. (b) L'image segmenté.
- Figure 3.14 – Résultat de segmentation. (a) L'image de profondeur. (b) L'image segmenté.
- Figure 3.15 – Résultat de segmentation, à gauche l'image de profondeur, à droite l'image segmenté.
- Figure 3.16 – Résultat de segmentation obtenu, à gauche une image de profondeur, à droite le résultat de la segmentation
- Figure 3.17 – Résultat de segmentation, à gauche l'image de profondeur, à droite l'image segmenté.
- Figure 3.18 – Exemple de discontinuité de profondeur. (a) Image de profondeur. (b) Agrandissement de la région carrée. (c) Représentation orthonormée du profil.
- Figure 3.19 Exemples de discontinuités de profondeur et de discontinuités d'orientation. (a) Codage par niveaux de gris. Trois profils sont sélectionnés. (b) Représentation orthonormée des profils sélectionnés.
- Figure 3.20 – Signal et modèle de discontinuité de profondeur associé.
- (a) Le signal, (b) Le modèle.
- Figure 3.21 – Détection et condition d'extremum. (a) signal, (b) modèle de discontinuité, (c) résultat de détection sans condition d'extremum (points de discontinuité en noir), (d) résultat de détection avec condition d'extremum.
- Figure 3.22 – Signal auquel on associe un modèle de discontinuité de profondeur.
- Figure 3.23 – Signaux auxquels on associe un modèle de discontinuité d'orientation.
- Figure 3.24 – Détection et condition d'extremum. (a) scène originale, (b) détection de discontinuité, (c) Résultat de segmentation
- Figure 3.25 – Image Disparition de connexité lors de l'amincissement d'une discontinuité de profondeur par condition d'extremum du gradient.
- Figure 3.26 – Détection conjointe des discontinuités d'ordre 0 et 1. (a) Mesure de discontinuité codée par des niveaux de gris. (b) Image des points de discontinuité obtenue par seuillage sur la mesure de discontinuité.
- Figure 3.27 – Détection conjointe des discontinuités, (a) discontinuités d'ordre 0, (b) discontinuités d'ordre 1

Chapitre 4 : Les contours actifs

- Figure 4.1 – Forces régissant l'évolution du contour actif
- Figure 4.2 – Forces régissant l'évolution du contour actif
- Figure 4.3 – Limites de convergence de contour avec la méthode classique
- Figure 4.4 – Evolution du contour « ballon » sur une main
- Figure 4.5 – Courbe représentant le nombre de polygone utilisés pour chaque itération
- Figure 4.6 – Exemple d'évolution du contour actif avec la méthode de GVF
- Figure 4.7 – Algorithme du Modèle Actif de forme « Actif Shape Model »
- Figure 4.8 – Détection de contour par l'ASM avec un marquage de 4, 8 puis 20 points
- Figure 4.9 – Courbe se déformant selon sa courbure
- Figure 4.10 – Exemple d'évolution du contour actif Géométrique

- Figure 4.11– Segmentation d’image par la méthode d’ensembles de niveau zéro
- Figure 4.12 – Exemple d’évolution du contour actif Géodésique
- Figure 4.13 – Exemple d’application du contour actif dynamique
- Figure 4.14 – Détection de la pince avec un contour actif basé région
- Figure 4.15 – Exemple d’application du modèle particulière

Chapitre 5 : Les systèmes multi-agents

- Figure 5.1 – Relation unissant l’environnement et l’ensemble des agents.
- Figure 5.2 – Principe de la réaction d’un agent réactif sur un environnement à partir de son cycle de vie.
- Figure 5.3 – Le concept d’émergence.
- Figure 5.4 – Illustration du concept d’auto-organisation.
- Figure 5.5 – Schématisation du concept la stigmergie.
- Figure 5.6 – Segmentation extraite de Haroun image originale et (b) image segmentée
- Figure 5.7 – Segmentation de Scherrer (a) image originale, (b) image segmentée décomposée en blocs et (c) l’histogramme de la représentation des tissus dans le bloc rouge
- Figure 5.8 – Segmentation de Germond (a) image originale et (b) image segmentée
- Figure 5.9 – Détection de contour proposée par Mazouzi, à gauche l’image original, à droite l’image segmentée
- Figure 5.10 – Segmentation de Huang (a) image originale et (b) image segmentée
- Figure 5.11 – Segmentation de Bourjot (a) image originale, (b) image segmentée en 4 régions.
- Figure 5.12 – Segmentation de Richard. (a) image originale décomposée en blocs, (b) image segmentée décomposée en blocs, (c) l’histogramme de l’image, (d) l’histogramme du bloc noir de l’image originale et (e) l’histogramme du bloc noir de l’image segmentée.
- Figure 5.13 – Architecture du système multi-agent proposée par Kazar
- Figure 5.14 – Segmentation de Huang (a) image originale et (b) image segmentée

Chapitre 6 : Classification des méthodes de segmentation d’images de profondeur

- Figure 6.1–Méthodes de segmentation d’images de profondeur par type de courbure
- Figure 6.2–Méthodes de segmentation d’images de profondeur sur surface algébrique
- Figure 6.3–Méthodes de segmentation d’images de profondeur sur surface plane
- Figure 6.4–Méthodes de segmentation d’images de profondeur suivant la continuité d’ordre 1
- Figure 6.5–Classification des méthodes de segmentation d’images de profondeur

Chapitre 7 : Segmentation d’images de profondeur par contour actif

- Figure 7.1 – Algorithme de Floyd-Steinberg
- Figure 7.2 – Algorithme de Floyd-Steinberg
- Figure 7.3 – Filtrage de l’image par Floyd & Steinberg. A gauche l’image de profondeur, à droite l’image filtrée.

- Figure 7.4 – Résultats de segmentation (a) image segmentée sans prétraitements. (b) image segmentée avec utilisation du filtre de Floyd & Steinberg.
- Figure 7.5 – Framework de comparaison de points de contour
- Figure 7.6 – Algorithme du Framework de comparaison
- Figure 7.7 – Calcul d'énergie du contour actif classique
- Figure 7.8 – Algorithme du contour actif classique
- Figure 7.9 – Déformation du contour actif classique, (a) contour initial, (b) contour après 30 itérations, (c) contour après 50 itérations, (d) contour après 100 itérations.
- Figure 7.10– Segmentation d'images de profondeur simple par contour actif classique: (a) Image de profondeur (b) Segmentation obtenue par le snake classique
- Figure 7.11 – Segmentation d'images de profondeur avec cavités par contour actif classique: (a) L'image de profondeur (b) Segmentation obtenue par le snake classique
- Figure 7.12 – Calcul d'amplitude de gradient dépassant un seuil déterminé
- Figure 7.13 – Algorithme du contour actif adaptatif
- Figure 7.14 – Déformation du contour actif adaptatif, (a) contour initial, (b) contour après 30 itérations, (c) contour après 50 itérations, (d) contour après 100 itérations.
- Figure 7.15 – Segmentation d'images de profondeur simple par contour actif adaptatif: (a) Image de profondeur à segmenter (b) Segmentation obtenue par le snake adaptatif (c)
- Figure 7.16 – Segmentation d'images de profondeur avec cavités par contour adaptatif classique: (a) L'image de profondeur (b) Segmentation obtenue par le snake adaptatif
- Figure 7.17 – Algorithme des Level sets
- Figure 7.18 – Calcul de la vitesse F de l'algorithme des Level Set
- Figure 7.19 – Segmentation d'images de profondeur simple par Level set: (a) Image de profondeur à segmenter (b) Segmentation obtenue par les Level set (c)
- Figure 7.20 – Segmentation d'images de profondeur avec cavités par Level set
- L'image de profondeur (b) Segmentation obtenue par les Level set
- Figure 7.21 – Taux de bonne détection pour les trois algorithmes
- Figure 7.22 – Taux de bonne détection pour les images simples
- Figure 7.23 – Taux de bonne détection pour les images avec cavités
- Figure 7.24 – Taux de bonne détection pour les images sans cavités

Chapitre 8 : Proposition d'une approche multi-agent pour la segmentation d'images de profondeur

- Figure 8.1 – Approche multi agents pour la segmentation d'images de profondeur par accroissement de régions

Liste des tableaux :

- Tableau 2.1 – Types d'images de profondeur et interprétation des données.
- Tableau 7.1 – Taux de bonne segmentation obtenus par l'algorithme de snake classique
- Tableau 7.2 – Taux de bonne segmentation obtenus par l'algorithme de snake adaptatif
- Tableau 7.3 – Taux de bonne segmentation obtenus par l'algorithme des Level Set
- Tableau 7.5 – Taux de bonne segmentation avec un écart supérieur à 3%

Liste des formules :

Formule 1.1 Calcul du seuil de l'histogramme

Formule 1.2 Calcul de la variance

Formule 1.3 Construction de la variance

Formule 1.4 Calcul du gradient

Formule 1.5 Calcul de la direction du gradient

Formule 1.6 Filtre de Prewitt

Formule 1.7 Calcul du Laplacien

Formule 1.8 Calcul du masque adapté

Formule 1.9 Rapport signal/bruit

Formule 1.10 Minimisation de la variance

Formule 1.11 Filtre de Canny

Formule 1.12 Détecteur de Moravec

Formule 1.13 Détecteur de Harris

Formule 1.14 Détecteur de Harris et Stephen

Formule 2.1 Coordonnées d'une image de profondeur cartésienne orthographique

Formule 2.2 Calcul des discontinuités d'une image de profondeur

Formule 3.1 Détecteur de point de discontinuité

Formule 3.2 Estimation des dérivées partielles (IP scalaire)

Formule 3.3 Estimation des dérivées dans les 4 directions discrètes (IP scalaire)

Formule 3.4 Estimation des dérivées de deuxième ordre

Formule 3.5 Calcul du gradient et des dérivées deuxièmes minimale et maximale

Formule 3.6 Calcul de la normale et des courbures

Formule 4.1 Equation de Lagrange

Formule 4.2 Calcul du Lagrangien

Formule 4.3 Energie cinétique du contour

Formule 4.4 Equation d'Euler Lagrange

Formule 4.5 Formule pour le recalcul du Lagrangien

Formule 4.6 Calcul de la courbure

Formule 4.7 Calcul de l'énergie

Formule 4.8 Calcul flux du vecteur du gradient

Formule 4.9 Calcul de l'énergie du GVF

Formule 4.10 Calcul de l'énergie du G-Snake

Formule 4.11 Calcul de la vitesse du Snake géométrique

Formule 4.12 Calcul de l'énergie du Snake géométrique

Formule 4.13 Calcul d'énergie des Level set

Formule 4.14 Evolution des Level set

Formule 4.15 Calcul de l'énergie du Snake géodésique

Formule 4.16 Calcul de l'énergie du contour actif dynamique

Formule 4.17 Calcul de l'énergie de frottement du contour actif dynamique

Formule 4.18 Calcul de l'énergie globale du contour actif dynamique

Formule 4.19 Calcul de l'énergie de région

Formule 4.20 Mesure du degré de similitude entre deux régions

Formule 4.21 Calcul de la particule

Formule 7.1 Calcul de l'énergie interne du contour actif classique

Formule 7.2 Calcul de l'énergie externe du contour actif classique

Formule 7.3 Calcul de l'énergie interne du contour actif adaptatif

Formule 7.4 Equation aux dérivées partielles

Formule 7.5 Calcul de la vitesse F des Levels set

Introduction générale

Pour un ordinateur, une image est une matrice dont les valeurs appelées pixels représentent des intensités lumineuses. Une image est donc vue comme un “tout” par un ordinateur et non pas comme une composition d'éléments. Par exemple, une image représentant un clavier d'ordinateur posé sur une table est perçue par un ordinateur comme étant une matrice de pixels. Si on veut mener une analyse sur la partie de l'image associée au clavier, on doit pouvoir séparer les pixels qui représentent le clavier de ceux représentant la table. C'est un problème de traitement d'image connu sous le nom de segmentation d'images. Très tôt, des moyens techniques et informatiques ont permis l'acquisition et le stockage d'images sous forme numérique. Il ne s'agit ni plus, ni moins que de la manipulation d'une matrice, dont les valeurs traduisent des intensités lumineuses. Les possibilités de traitement automatique de ces images allaient pourtant s'avérer délicates, car des capacités aussi banales pour l'œil humain que la reconnaissance d'un objet ou d'un visage allaient poser de réelles difficultés pour l'outil informatique. La recherche en vision par ordinateur a donc rapidement cherché à diminuer le volume de données à traiter, afin de concentrer le maximum d'information présente dans l'image dans un minimum de données. L'ambition à était de pouvoir réaliser des traitements plus poussés sur des données restreintes. Le principe des détecteurs de contours et de régions sont apparus en réponse à ce besoin. Plutôt que de manipuler la matrice de l'image dans sa totalité, il suffit de conserver les points de l'image où il y a les plus forts changements de luminosité (les contours), et de regrouper sous une même étiquette les points qui ont une luminosité proche (les régions).

I. Problématique

Le processus de segmentation est peut-être l'étape la plus importante de l'analyse d'image sachant que sa performance affecte directement les étapes de traitement ultérieures de l'image et détermine de manière significative l'interprétation de l'image résultante. Les deux principales difficultés du problème de segmentation de profondeur sont sa nature et l'absence de définition de la segmentation "correcte". Peut-être en raison de ces défauts, une pléthore d'algorithmes de segmentation a été proposée dans la littérature. Ces algorithmes vont de régimes ad hoc simples aux plus sophistiquées utilisant des modèles d'objets et d'images.

La segmentation de profondeur souffre généralement de l'absence de résultats et des méthodes comparatives. L'image de profondeur contient non seulement l'information liée à la couleur et l'intensité d'un pixel comme l'image d'intensité, mais en plus, elle contient un vecteur associé à la profondeur de ce pixel « appelé RANGEL ». Elle fournit des informations géométriques sur l'objet indépendamment de la position, l'orientation et l'intensité des sources de lumière éclairant la scène, ou des propriétés de réflectance de cet objet. La segmentation d'images de profondeur a été l'objet de beaucoup de recherche dans le domaine de vision par ordinateur depuis près de 30 ans. Au cours de cette période, plusieurs résultats partiels ont trouvé leur place dans de nombreuses applications industrielles telles que l'inspection géométrique, l'ingénierie inverse ou les systèmes de navigation autonomes. Cependant, de la même manière que la segmentation d'image spectrale, la segmentation d'image de profondeur est encore loin d'être satisfaisante.

II. Objectifs de ce travail

Il existe plusieurs méthodes de segmentation d'une image de profondeur en régions, ce qui, par la suite, peuvent être analysés, en fonction de leurs formes, tailles, positions relatives et d'autres caractéristiques, et il existe plusieurs classifications possibles de techniques de segmentation d'images de profondeur. Notre objectif premier est de proposer une nouvelle classification des techniques de segmentation d'images de profondeur. La classification proposée dans notre travail scindera les méthodes de segmentation en trois ou quatre points de vue philosophiques différents en se basant sur le critère d'homogénéité (critère de surface) auquel obéit la méthode de segmentation.

Notre second objectif sera d'appliquer des modèles déformables pour segmenter des images de profondeur, nous utiliserons pour cela trois contours actifs, un contour actif de base classique, un contour actif classique amélioré qu'on appellera « contour actif adaptatif » ainsi que la méthode des Level Set « ensemble de lignes de niveaux ». Nous voulons à travers cette application voir le comportement des contours actifs sur des images de profondeurs, nous essaierons par la suite d'analyser les résultats de segmentation obtenus pour voir quel est l'algorithme le plus adapté à ce type d'images.

Notre troisième et dernier objectif est de proposer une approche multi-agent pour effectuer une segmentation par région d'une image de profondeur, nous aimerions profiter de la

souplesse et de l'autonomie qu'offrent les systèmes multi-agent par leurs répartitions sur l'image en entités indépendantes.

III. Organisation de la thèse

Outre l'introduction générale et la conclusion générale, le travail de cette thèse sera scindé en deux parties, une première partie destinée à dresser un état de l'art et une deuxième partie qui englobe les travaux réalisés, notre travail sera donc organisé comme suit :

Première partie : Etat de l'art

Chapitre 1 : Analyse et segmentation d'images numériques

Dans le premier chapitre, nous allons dresser un état de l'art des méthodes de segmentation utilisée en imagerie numérique, nous présenterons ainsi une multitude d'algorithmes de segmentations selon différentes méthodes d'approches tel la segmentation basée pixels, la segmentation basée régions, la segmentation par détection de contour et nous finirons ce chapitre par la présentation de quelques détecteurs de points d'intérêts ainsi que la poursuite et fermeture des contours.

Chapitre 2 : L'image de profondeur

Dans le deuxième chapitre, nous allons présenter l'image de profondeur ainsi que ces diverses propriétés telles la profondeur et la cohérence spatiale d'une image de profondeur. Nous présenterons aussi dans ce chapitre les types d'images de profondeurs existants ainsi que les diverses représentations graphiques possible d'une image de profondeur. Puis, nous nous intéresserons aux différents systèmes d'acquisition d'images de profondeur qui n'ont cessé d'évoluer durant ces dix dernières années. Pour finir, nous allons décrire les différentes techniques d'estimation du bruit dans une image de profondeur.

Chapitre 3 : Segmentation d'images de profondeurs

Dans le troisième chapitre nous allons présenter les différentes approches de segmentation d'images de profondeur ainsi que plusieurs détecteurs de discontinuités. Nous verrons dans ce chapitre à quelle point la segmentation d'images de profondeur se rapproche des techniques désormais classiques de segmentation d'images 2D. Dans ce chapitre nous passerons en revue toutes les méthodes élémentaires de la segmentation d'images de profondeur, puis, nous verrons par la suite comment ces méthodes élémentaires sont combinées pour élaborer de nouvelles approches de segmentation de profondeur

Chapitre 4 : Les contours actifs

Dans le quatrième chapitre, nous allons présenter le principe des contours actifs en posant quelques définitions telles les notions d'énergies (interne et externe) et la convergence d'un contour actif, puis nous présenterons les différentes approches pour définir un critère de segmentation pour les contours actifs, de l'approche purement "contours" aux approches plus récentes qui intègrent des termes de régions dans les fonctionnelles à minimiser.

Chapitre 5 : Les systèmes multi-agents

Dans le cinquième chapitre; nous allons donner un aperçu général sur les systèmes multi-agents, nous présenterons donc les éléments fondamentaux qui composent un système multi-agents ainsi que les types d'agent existants (cognitif, réactif et hybride). Nous expliquerons aussi les différents modes de coopérations possibles entre agents. Puis, nous allons procéder à l'analyse des systèmes multi-agents et leur apport dans la vision par ordinateur, nous verrons ainsi les différentes approches multi agent proposées en segmentation d'images.

Deuxième partie : Travaux réalisés

Chapitre 6 : Classification des méthodes de segmentation d'images de profondeurs

Après avoir passer en revue les méthodes de segmentation d'image de profondeur au troisième chapitre, nous allons dans ce chapitre proposer une classification des méthodes de segmentation appliquées aux images de profondeur, bien que le critère d'homogénéité de la segmentation n'est pas toujours explicite, nous allons tout de même essayer de classer toutes les méthodes suivant les quatre critères d'homogénéité (type de courbure, surface plane, surface algébrique, continuité C1).

Chapitre 7 : Segmentation d'images de profondeur par contour actifs

Dans ce chapitre, nous allons appliquer à des images de profondeurs trois types de modèle déformables différents, les contours actifs classiques, les contours actifs adaptatifs et les Level set. Les résultats obtenus à la fin de cette série de segmentation vont êtres comparés pour ainsi essayer de trouver quel type de modèle déformables est le plus approprié à une segmentation d'image de profondeur.

Chapitre 8 : Proposition d'une approche multi-agent pour la segmentation d'images de profondeur

Dans ce chapitre nous allons proposer une méthode de segmentation d'image de profondeur en adoptant une approche multi-agent. Nous présenterons dans ce chapitre une méthode d'accroissement de régions où chaque agent représentera une région de l'image, nous verrons comment l'accroissement sera accompli par ces agents ainsi que les différents processus de communication mis à leur disposition pour communiquer entre eux et ainsi synchroniser leurs travaux.

Chapitre 1

Analyse et segmentation d'images numériques

1.1 Introduction

La segmentation d'images est l'un des problèmes phares du traitement d'images, elle consiste à partitionner l'image en un ensemble de régions connexes. L'intérêt de ces régions est de pouvoir être manipulées ensuite via des traitements de haut niveau pour extraire des caractéristiques de forme, de position, de taille, nous aborderons dans ce document une multitude d'algorithmes de segmentations selon différentes méthodes d'approche. Nous allons dans ce chapitre dresser un état de l'art de ces méthodes de segmentation.

1.2 Segmentation d'image basée sur les pixels

Cette première catégorie travaille sur des histogrammes de l'image. L'algorithme construit des classes de couleurs qui sont ensuite projetées sur l'image. On regroupe ainsi des endroits de l'image par couleur, mais cette segmentation n'est pas toujours optimale car certaines régions de l'image sont disjointes même si elles appartiennent à un même groupe de couleur.

1.2.1 Sélection récursive sur histogrammes

La technique de Sélection récursive sur histogrammes [OHLANDER & al, 1978] est une méthode classique de segmentation d'images qui opère essentiellement dans l'espace tonal (l'histogramme) le principe est de calculer l'histogramme de toutes les composantes R, G et B de l'image. Puis, un lissage des histogrammes résultants avec une convolution à une dimension est effectué. Le pic le plus significatif sur l'ensemble des histogrammes H_x est sélectionné. L'image binaire X dont les valeurs sont comprises entre x_1 et x_2 est calculée (x_1 et x_2 étant l'intervalle représentant la largeur du pic H_x). Ensuite, l'ensemble de pixels précédemment formés est retiré du support de l'image. Si le nombre de pixels restants est suffisant, on ré-exécute l'algorithme sur ce nouvel ensemble jusqu'à ce que le nombre de pixels restants soit jugé insuffisant pour une autre exécution.

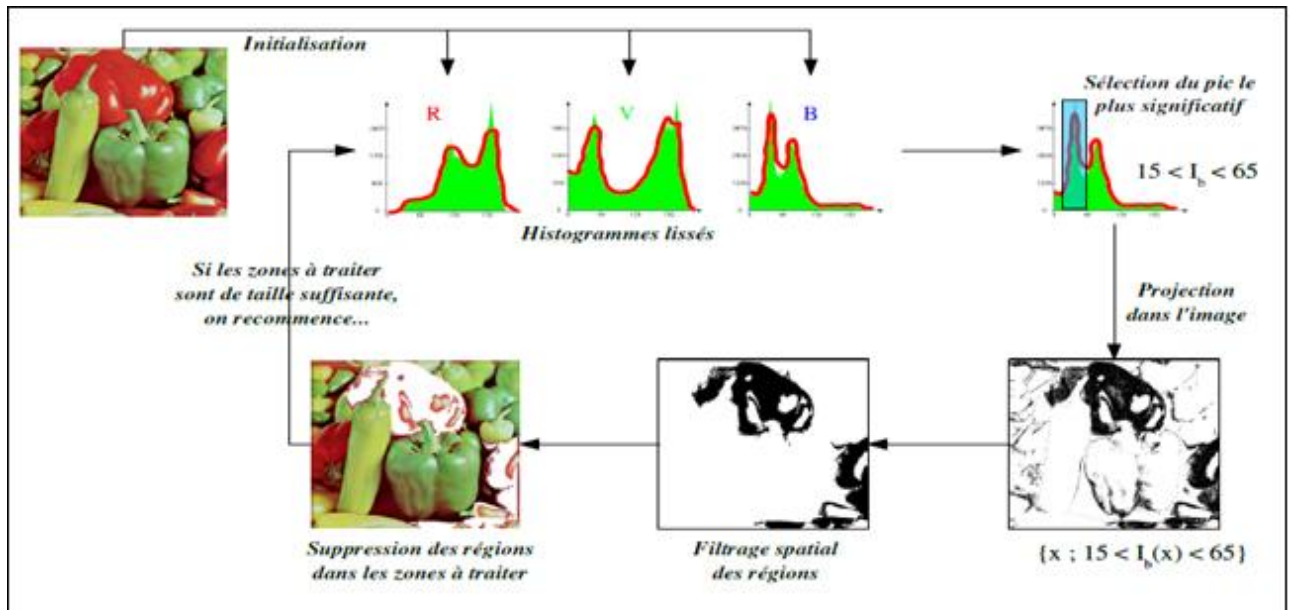


Figure 1.1 – Illustration du principe de l’algorithme de sélection récursive sur histogrammes

Cet algorithme a l’avantage d’être rapide à calculer et bénéficie d’une faible sensibilité aux bruits. Cependant, c’est une méthode globale et signifie que l’appartenance d’un pixel à une région dépend de l’image entière et non des pixels voisins.

1.2.2 Détection de vallées

Cette technique est une technique très intuitive car on suppose que chaque classe correspond à une gamme distincte de niveaux de gris. L’histogramme est alors m-modal. La position des minima de l’histogramme H permet de fixer les $(m-1)$ seuils nécessaires pour séparer les m classes, en termes mathématiques. Les seuils s_i sont obtenus par la formule :

$$H(S_i) = \text{Min} [H(k)], \text{ pour } k \text{ in }]m_i, m_{i+1}[\quad (1.1)$$

Où m_i et m_{i+1} sont les valeurs moyennes de l’intensité lumineuse dans les classes C_i et C_{i+1} . Malgré le développement de techniques robustes visant à faciliter la détection des vallées, cette méthode, bien que simple, est très peu appliquée car les histogrammes traités sont le plus souvent bruités et uni modaux.

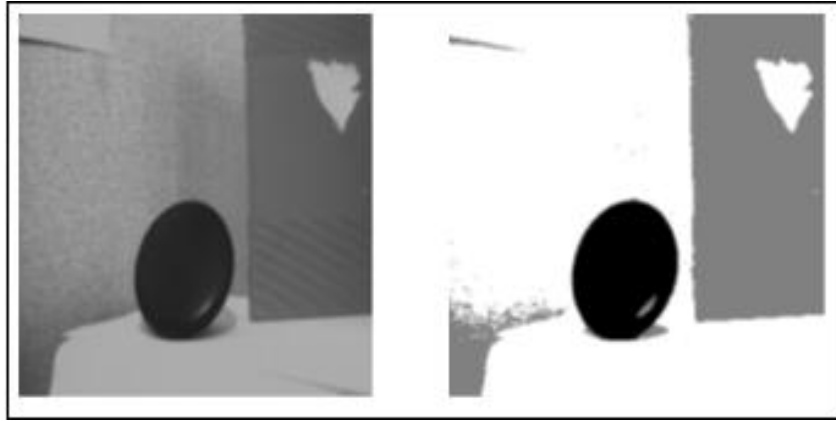


Figure 1.2 – Résultat de la segmentation par détection de vallées

Si certains objets sont bien détectés, il aurait fallu utiliser plus de seuils dans les niveaux de gris clairs pour mieux mettre en évidence les différentes parties de l'image « *Figure 1.2* ». C'est le principal inconvénient de beaucoup d'approche du seuillage où l'on doit définir a priori combien de classes doivent être séparées. Néanmoins cette technique exploite au mieux les deux grands avantages du seuillage qui sont la simplicité et la rapidité.

1.2.3 Minimisation de la variance

La répartition des pixels en N classes est un problème classique de classification. Le choix des seuils S_i permet de détecter m classes auxquels on peut associer taille (T_i), moyenne (m_i) et variance V_i par :

$$\begin{aligned}
 T_i &= \sum_{D_j} \{H(j)\} \\
 m_i &= \sum_{D_j} \{j \cdot H(j) / t_i\} \\
 V_i &= \sum_{D_j} \{(j - m_i)^2 \cdot H(j) / t_i\}
 \end{aligned}
 \tag{1.2}$$

Où H est l'histogramme normalisé (son intégrale est ramenée à l'unité) et $D_j = [s_{i-1}, s_i[$ est la gamme de niveaux de gris correspondant à la classe C_i .

A partir de ces indicateurs statistiques, on peut construire la variance intra classe totale W par :

$$W = \sum (m_i) \{t_i \cdot V_i\}
 \tag{1.3}$$

Le meilleur seuillage dans cette approche correspond à une minimisation de la variance intra classe mais cette technique est difficilement applicable lorsque le nombre de classes est élevé, de plus, il faut que chaque classe ait une taille significative en nombre de niveaux de gris pour

que les indicateurs statistiques aient un sens. Dans le cas de où $N=2$, cette méthode est performante. Sur l'exemple « *Figure 1.3* » un seuil a été déterminé par l'algorithme optimal de Fisher. Le seuil trouvé est de 127 et compte tenu du fait que l'on n'utilise qu'un seul seuil, toutes les composantes de l'image ne peuvent être séparées.

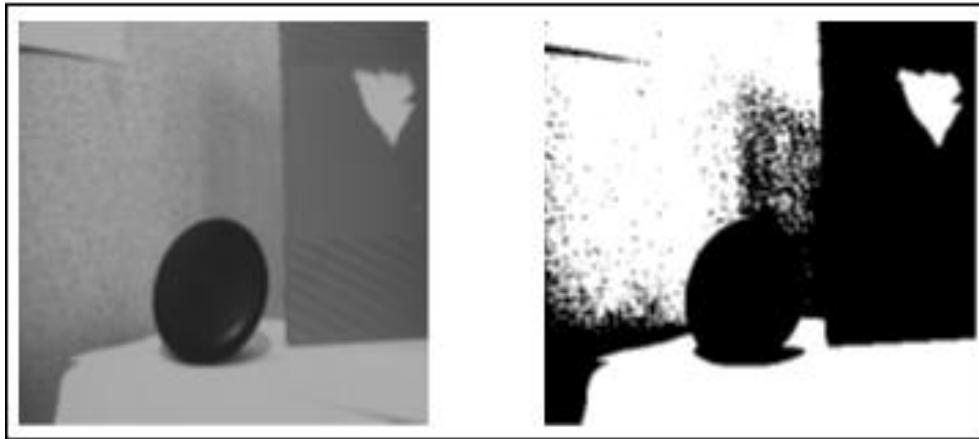


Figure 1.3 – Résultat de la segmentation par minimisation de la variance

1.3 Segmentation basée sur les régions

Cette catégorie peut correspondre aux algorithmes d'accroissement de régions. Ce type de segmentation consiste à prendre un ensemble de petites régions uniformes dans l'image et à regrouper les régions adjacentes de même couleur jusqu'à ce qu'aucun regroupement ne soit plus possible. Les méthodes d'accroissement de régions sont les méthodes de segmentation les plus simples. Cette catégorie peut aussi correspondre à des algorithmes de découpage de régions. On part de l'image entière que l'on va subdiviser en plus petites régions tant que ces régions ne sont pas suffisamment homogènes. Les algorithmes dits « split and merge » sont un mélange de ces deux techniques.

1.3.1 Accroissement de région

Cette méthode [Baillie, 2003] consiste à choisir une région R (Initialement, $R = 1$ pixel) et à étendre la région R en incluant les pixels situés sur la frontière et dont la couleur est proche de celle de R (la variation de couleur est inférieure à un seuil) en répétant cette procédure jusqu'à ce qu'il n'y a plus de pixels de couleur assez proche sur la frontière, on obtient une région de couleur homogène maximale autour du pixel de départ. On associe à chaque pixel de l'image

un index qui est un nombre entier, le but de l'algorithme va être de donner à chaque pixel une valeur d'index qui corresponde à un numéro de région. Pour connaître ensuite l'étendu de la région n , il suffira d'extraire tous les pixels dont l'index vaut n (l'index de chaque pixel est initialement fixé à -1 , valeur indiquant que l'index n'a pas encore été attribué) et on parcourt l'image de haut en bas et de gauche à droite et pour chaque pixel C examiné, on considère les pixels adjacents $c_1; c_2; c_3$ qui ont déjà été examinés. On va ensuite choisir parmi $c_1; c_2; c_3$ l'ensemble des pixels dont la couleur est identique à celle de C , à un facteur de tolérance G près. Si cet ensemble est vide, on crée un nouvel index que l'on attribue au pixel courant c (il existe un compteur d'index qui permet de savoir quel est le prochain nouveau numéro d'index libre). Si cet ensemble n'est pas vide et que chaque pixel le constituant a le même index k , il suffit d'attribuer l'index k au pixel courant C , dans le cas contraire, la situation est un peu plus complexe. Supposons que c_1 et c ont la même couleur (à G près) et l'index de c_1 est 4. De même supposons que c_2 et c ont la même couleur et que l'index de c_2 est 7. Une possibilité est de parcourir l'image déjà traitée et de changer tous les index de valeur 7 en index de valeur 4 et d'attribuer la valeur 4 à l'index de c . Cette solution n'est pas satisfaisante du point de vue de la complexité (algorithme en N^2 ; N étant la taille de l'image). Le plus efficace est de tenir à jour un chaînage qui indique les équivalences entre indexes et le résultat final sera une image indexée, chaque index valant un numéro de région, la complexité globale est de l'ordre de N , la taille de l'image « *Figure 1.4* ».

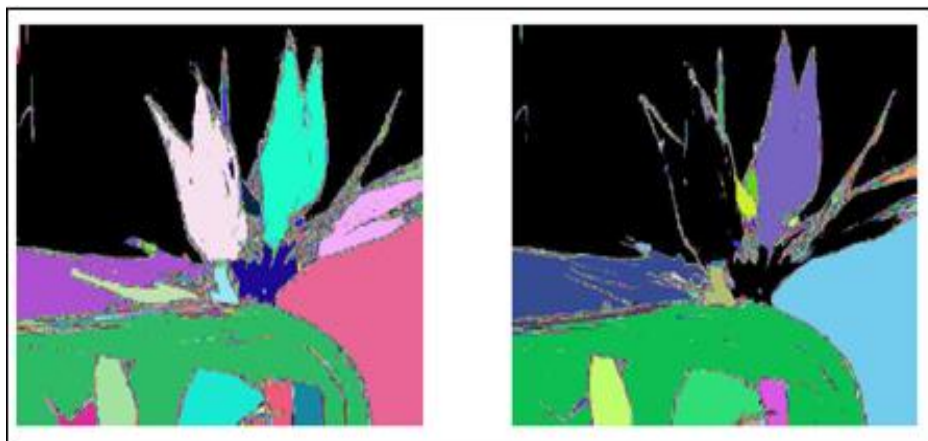


Figure 1.4 – Accroissement de région : A gauche, seuil=6. A droite, seuil=7

On constate que les fleurs sont correctement extraites mais la base en bas se confond avec le fond. Ce problème est caractéristique des algorithmes de segmentation dit « locaux » qui n'ont pas de vue d'ensemble ou de description de plus haut niveau que le pixel. Dans le cas de

l'accroissement de région, entre deux points de deux régions connexes, il existe presque toujours une chaîne de pixels connexes pour laquelle l'écart de couleur entre deux pixels adjacents est inférieur au seuil. Cela se traduit par des effets de « ponts » entre zones, particulièrement sensibles dans les zones de dégradé. La conséquence est que les régions ont tendance à se mélanger au delà d'un certain seuil, et ceci même si les régions en question sont très différentes du point de vue de la couleur.

A droite, la même image est segmentée avec un seuil de 7. On constate que la fleur du milieu est confondue avec le fond (il existe visiblement un pont de pixels pour le seuil 7 entre cette région et le fond). Ceci met en lumière l'un des principaux problèmes liés à l'accroissement de région qui est la sensibilité au seuil mais il n'existe pas de seuil qui fonctionne correctement pour toute une classe d'images, ce qui rend l'automatisation des algorithmes très difficile, à titre d'exemple, la figure 6 montre la segmentation de l'image précédente avec un seuil de 3 (à gauche) et de 4 (à droite) et un filtre pour supprimer les régions de taille inférieure à 100.

Dans cette méthode le résultat final est très sensible à la valeur du seuil et on remarque aussi dans l'exemple de la *Figure 1.5* que les régions obtenues dépendent fortement des pixels d'amorçage choisis et de l'ordre dans lequel les pixels de la frontière sont examinés.

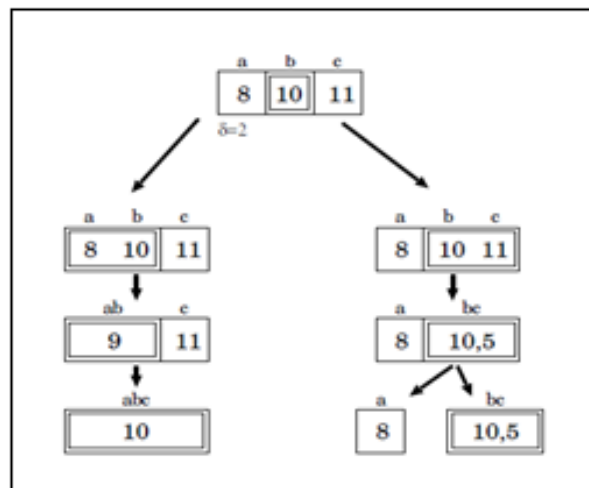


Figure 1.5 – Deux schémas de regroupement pour des pixels a, b et c

1.3.2 La méthode du Split and Merge

L'idée des algorithmes de type Split & Merge [Chaudhuri & al, 2010] est de produire automatiquement une partition initiale en petites régions, qui vont ensuite croître en se regroupant (Merge), la partition initiale (Split) est réalisée en divisant récursivement l'image en régions de tailles identiques comme illustré dans la *Figure 1.6*.

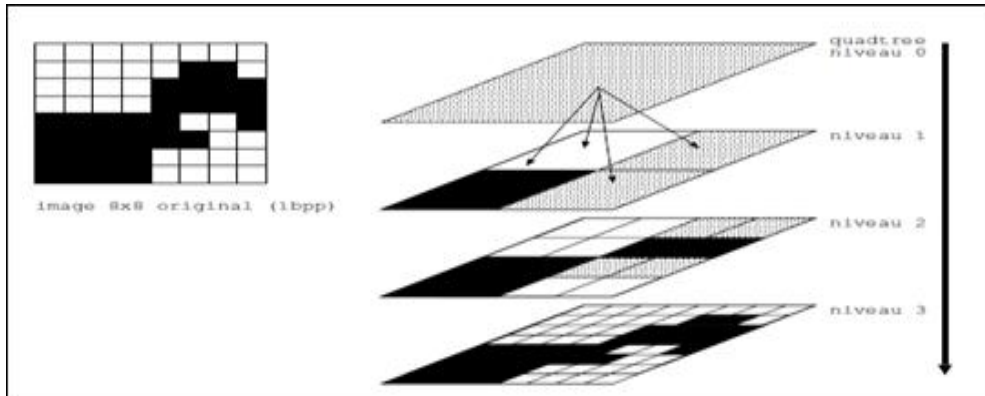


Figure 1.6 – Découpage par quadtree d'une image 8x8. Traitement avec seuil=100%

Le graphe d'adjacence « *Figure 1.7* », ou Région Adjacence Graph (RAG) est créé et à chaque région est associé un sommet du graphe, et des arêtes relient les sommets correspondants à deux régions qui se touchent. La phase de regroupement (Merge) utilise le RAG pour modifier la partition initiale où pour chaque sommet R du RAG, on cherche s'il existe un sommet R' voisin dans le RAG et de valeur suffisamment proche, et si c'est le cas, on les fusionne.

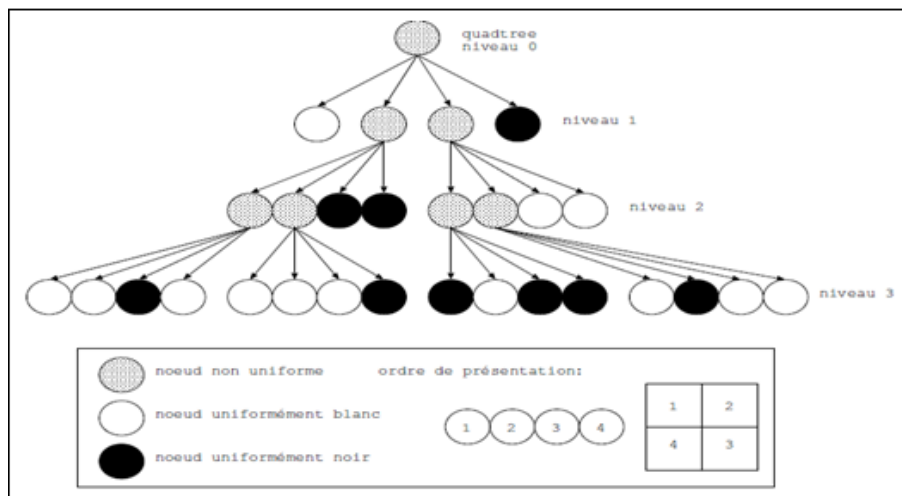


Figure 1.7 – Exemple d'un Région Adjacence Graph

Comme illustré dans la *Figure 1.8*, l'algorithme Split and Merge a été appliqué sur une image mais le seuil utilisé n'est pas précisé et on constate que le nombre de régions extraites est trop élevé car il y a une multitude de petites régions isolées.

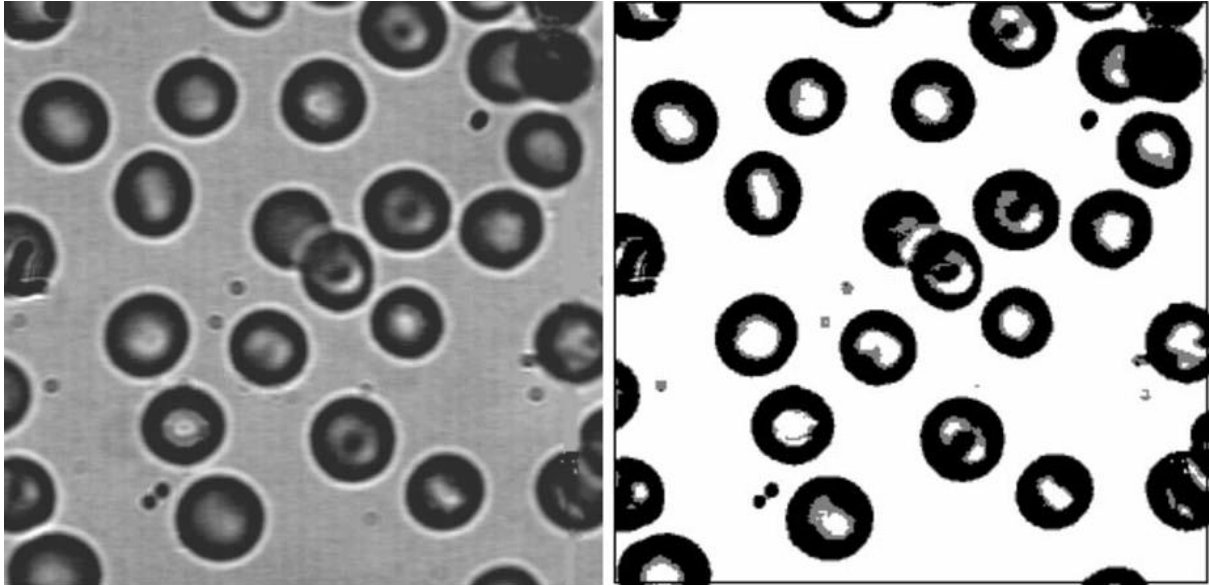


Figure 1.8 – Exemple d'utilisation de l'algorithme du Split and Merge. A droite, la segmentation avec Split and Merge. A gauche l'image originale.

On peut améliorer ce résultat en augmentant le seuil mais quoi qu'il en soit, le résultat final présentera toujours une forme morcelée en escaliers, caractéristique du prétraitement en quadtree, on touche ici au principal désavantage de la méthode Split and Merge qui a beaucoup de difficultés à restituer des régions aux contours naturels et les régions ont toujours cette apparence carrée, de plus, cette méthode souffre du problème général des algorithmes d'accroissement de région qui est que le résultat final dépend de l'ordre dans lequel sont examinées les régions adjacentes lors du regroupement.

1.3.3 La méthode du CSC (Color Structure Code)

L'algorithme CSC (Color Structure Code) a été présenté en 1993 par Rehrmann et Priesse [Baillie, 2001] et il repose sur l'utilisation d'une structure hexagonale hiérarchique de codage de l'image. Il s'agit d'un algorithme de Split and Merge particulièrement efficace et robuste où l'une des propriétés de cet algorithme est qu'il est complètement parallélisable et qu'il ne dépend pas d'un ordre de traitement de l'image comme les algorithmes d'accroissement de région. L'algorithme repose sur l'utilisation d'une topologie hiérarchique particulière des

pixels et la topologie hiérarchique est constituée d'îlots hexagonaux disposés à différents niveaux comme illustré dans la *Figure 1.9*.

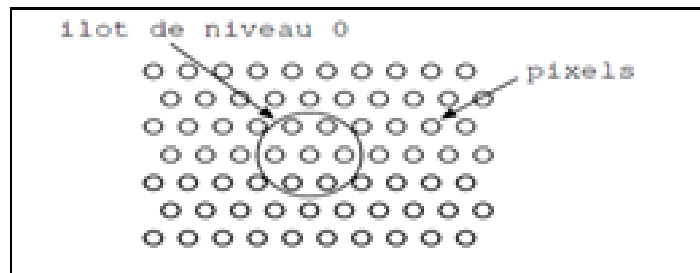


Figure 1.9 – Exemple d'îlot de niveau 0 basé sur la topologie hexagonale

Cette topologie est récursive comme le montre la *Figure 1.10*, c'est à dire qu'un îlot de niveau $n+1$ est constitué de sept îlots de niveau n , disposés hexagonalement de la même manière que les pixels constituant un îlot de niveau 0.

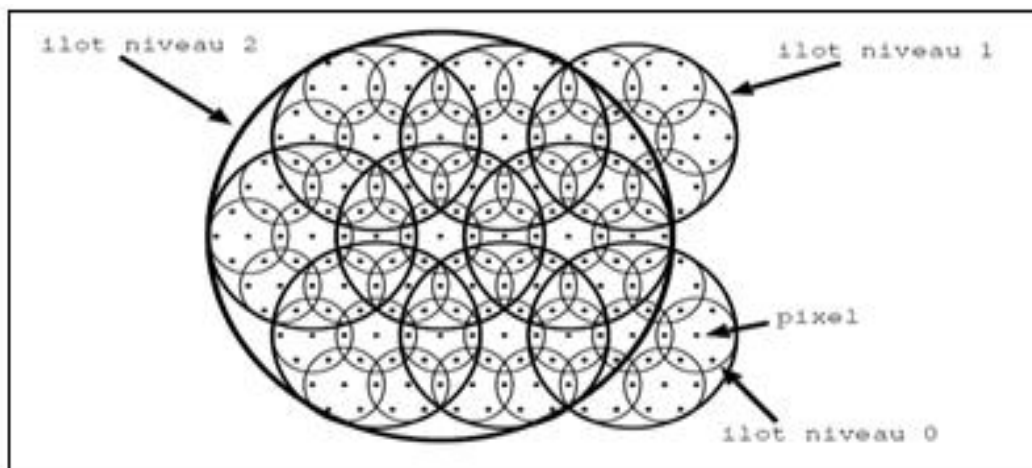


Figure 1.10 – Topologie de l'algorithme CSC en structure hexagonale hiérarchique

La première phase (Initialisation) de l'algorithme CSC consiste à appliquer un simple algorithme d'accroissement de région « *Figure 1.11* » localement dans chaque îlot de niveau 0 nous et appellerons les régions résultantes des éléments (L'étape de construction de ces éléments de niveau 0 peut s'exécuter en parallèle sur plusieurs processeurs puisque le traitement effectué dans un îlot ignore complètement celui de l'îlot voisin, bien qu'ils partagent un pixel un commun).

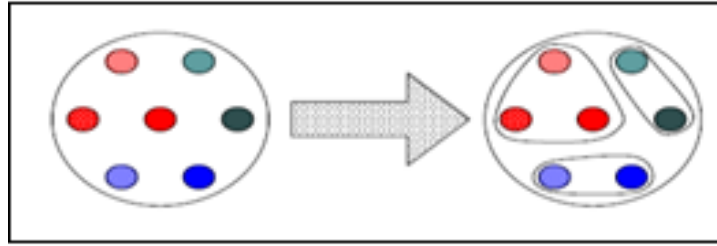


Figure 1.11 – Phase d’initialisation de l’algorithme CSC

Dans la deuxième phase (Regroupement), des éléments de niveau N sont regroupés en éléments de niveau $N + 1$, cette opération s'effectue à l'intérieur d'un îlot de niveau $N + 1$ et les éléments de niveau N sont regroupés si ils sont connectés et de couleur proche « *Figure 1.12* », le test de connectivité est particulièrement efficace (Deux éléments de niveau N sont connectés s'ils ont un élément de niveau $N - 1$ en commun). L'opération est répétée de niveau en niveau jusqu'au niveau recouvrant toute l'image et le résultat sera un arbre de K éléments de niveau N , lorsqu'un élément de niveau N ne trouve aucun autre élément auquel se lier, il devient la racine d'un arbre correspondant à une région terminée de l'image et peut être stocké.

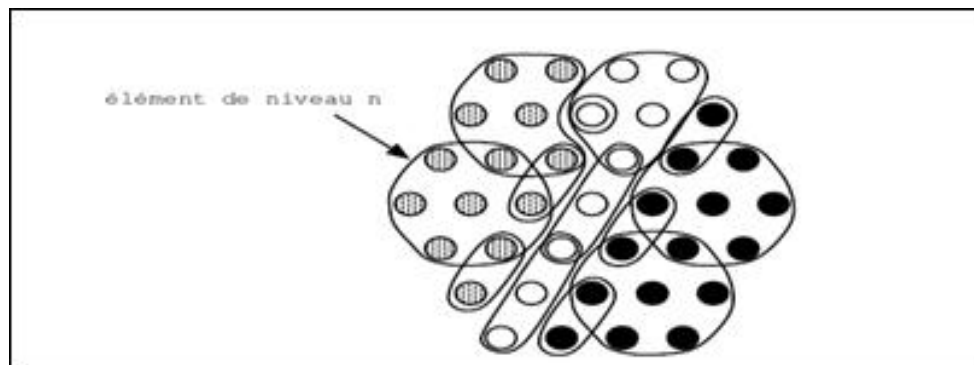


Figure 1.12 – Phase de regroupement de l’algorithme CSC

Dans la troisième et dernière phase (Découpage) si les éléments de niveau n sont connectés et de couleur similaire ils seront regroupés dans un élément de niveau $n+1$. Dans le cas où ils sont connectés mais de couleur incompatible ils ne vont pas être regroupés, bien que tous les éléments de niveau $N-1$ les constituant soient de couleurs homogènes et proches « *Figure 1.13* ».

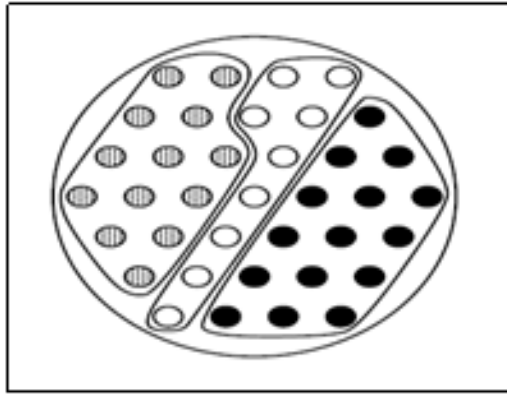


Figure 1.13 – Phase de découpage de l’algorithme CSC

Sur l’exemple de la *Figure 1.14* nous reprenons la même image utilisée pour expérimenter le Split and Merge et il lui est appliqué l’algorithme CSC. L’erreur caractéristique des algorithmes d’accroissement de région est de connecter deux régions de couleurs différentes parce qu’il existe un pont ou chaîne de pixels de la première région vers la deuxième, le long de cette chaîne, chaque pixel a une couleur proche de celle de son voisin, ce qui fait que l’algorithme ne voit pas la dérive qui mène d’une couleur à l’autre. Ces algorithmes utilisent uniquement de l’information locale et ne sont pas capables de détecter des contours au contraste trop faible. L’algorithme CSC utilise à la fois des informations locales (phase de regroupement) et globales (phase de découpage).



Figure 1.14 – Exemple de segmentation par l’algorithme CSC

1.4 Segmentation d'image par détection de contour

La recherche des contours dans une image numérique est un des problèmes les plus étudiés depuis l'origine des travaux sur l'imagerie numérique [Catanzano, 2001]. Ceci est en grande partie dû à la nature très intuitive du contour qui apparaît très naturellement comme l'indice visuel idéal dans la plus grande partie des situations, les contours « *Figure 1.15* » constituent en effet des indices riches, au même titre que les points d'intérêts, pour toute interprétation ultérieure de l'image. Très schématiquement, les contours sont les lieux de variations significatives de l'information niveaux de gris, la notion de contour étant liée à celle de variation, il est évident qu'une telle définition nous amène tout naturellement vers une évaluation de la variation en chaque pixel.



Figure 1.15 – Différents types de contours : marche, toit et pointe

Le principe de la détection de contours repose donc sur l'étude des dérivées de la fonction d'intensité « *Figure 1.16* » dans l'image, les extrema locaux du gradient de la fonction d'intensité et les passages par zéro du Laplacien, la difficulté réside dans la présence de bruit dans les images.

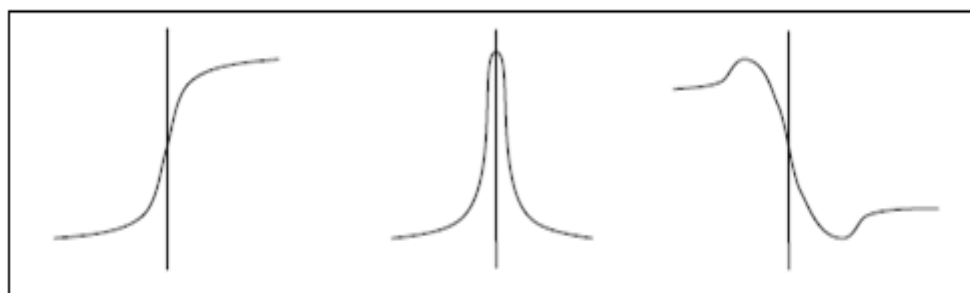


Figure 1.16 – Fonction d'intensité au voisinage d'un contour, ses dérivées première et seconde

Une variation existera si le gradient est localement maximum ou si la dérivée seconde (à définir dans un espace bidimensionnel) présente un passage par zéro. Les principaux algorithmes connus (Sobel, Prewitt, Kirsh, Canny, Deriche) se focalisent sur ce premier aspect du contour.

1.4.1 Les méthodes dérivatives du premier ordre

L'image de gradient [Fauqueur, 2001] d'une image en niveaux de gris est aussi une image en niveaux de gris qui illustre les différences locales de luminosité des pixels. C'est une approximation de la norme de la dérivée première (selon les axes X et Y) de la luminosité en chaque point. Le gradient d'une image est caractérisé par son module et sa direction

- Le module du vecteur Gradient représente la pente de la surface image au point de calcul, la présence locale d'un module élevé traduit une forte variation du niveau de gris autour de ce point.
- La direction du vecteur gradient « *Figure 1.17* » donne la direction de cette pente dans le sens croissant.

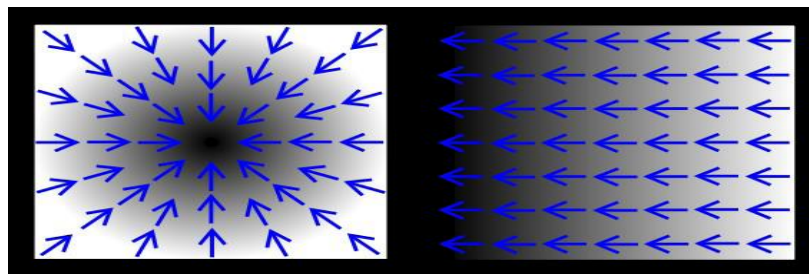


Figure 1.17 – Les lignes bleues représentent le gradient de couleur du plus clair vers le plus foncé

La valeur du gradient selon l'axe X est G_x et la valeur du gradient selon l'axe Y est G_y , la norme du gradient $G(x,y)$ calculé pour le pixel est

$$G(x,y) = \sqrt{G_x^2 + G_y^2} \quad (1.4)$$

G_x et G_y correspondent respectivement aux différences de luminosité selon les axes X et Y de l'image calculées en (x,y) .

La direction du gradient est donnée par:

$$d(x,y) = \text{Arctg}(G_y(x,y) / G_x(x,y)) \quad (1.5)$$

Une version plus évoluée et plus précise du gradient repose sur le même calcul, mais avec des masques plus larges qui est le masque de Sobel où les résultats de détection de contour sont meilleurs « *Figure 1.18* ».

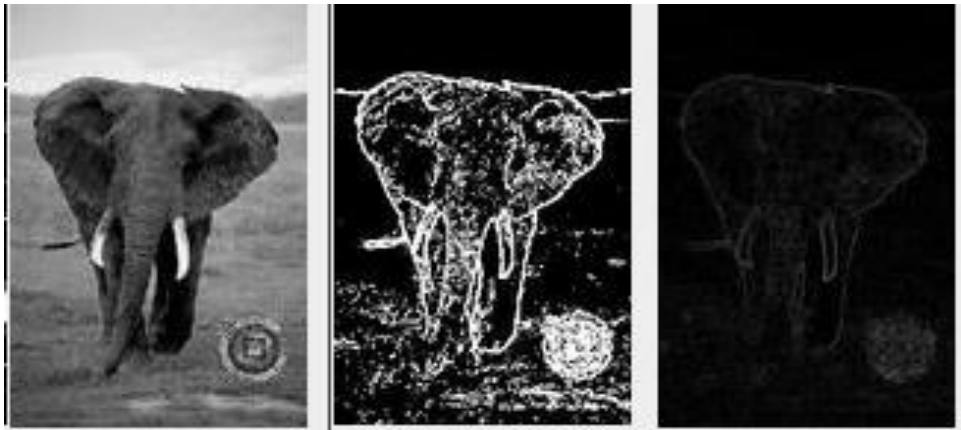


Figure 1.18 – L'image originale. L'image traitée par le masque de Sobel, puis par le Gradient

Un autre filtre a été élaboré qui est le filtre de Prewitt « *Figure 1.19* » sur une matrice de 3*3 est qui est :

$$\begin{pmatrix} -1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & 1 & 1 \end{pmatrix} \quad (1.6)$$



Figure 1.19 Image originale, Utilisation du filtre de Sobel, Utilisation du filtre de Prewitt.

Plus la valeur de $G(x,y)$ sera élevée, plus le point (x,y) sera susceptible d'être un point de contour. Pour que les $G(x,y)$ soient dans $[0;255]$, plutôt que de normaliser les valeurs de $G(x,y)$ comme il est conseillé de faire dans le cas simple du filtrage, pour la détection de contours, il est plus efficace de tronquer à 255 les $G(x,y)$ supérieurs à 255.

1.4.2 Les méthodes dérivatives du deuxième ordre

Ces méthodes ont été proposées en 1976, l'image du Laplacien [Maître, 2003] d'une image en niveaux de gris est aussi une image en niveaux de gris qui illustre les différences locales de luminosité des pixels « Figure 1.20 », mais de manière plus fine que le gradient, puisque le Laplacien est une approximation de la norme de la dérivée seconde (selon les axes X et Y) de la luminosité en chaque point.. Elles utilisent le fait que le passage par zéro du Laplacien permet de bien mettre en évidence les extrémums de la dérivée. Il a été constaté qu'en l'absence de forte courbure, le passage par zéro du Laplacien correspond en effet bien au maximum du gradient dans la direction du gradient. Ces méthodes tirent en outre profit du fait que les zéros de la dérivée seconde constituent un réseau de lignes fermées (évitant donc, en principe, les étapes de poursuite et de fermeture). Mais l'estimation de la dérivée seconde étant très sensible aux bruits, il convient de filtrer très fortement l'image avant d'en mesurer le Laplacien.

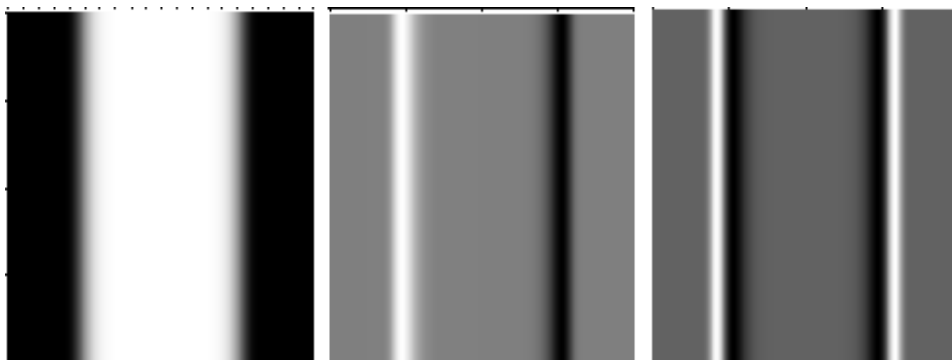


Figure 1.20 – Image en niveau de gris, première dérivée de l'image et deuxième dérivée

Plusieurs approximations discrètes du Laplacien existent et l'image du Laplacien s'obtient par convolution avec l'un des masques suivants:

$$M = \begin{matrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{matrix} \quad \text{ou} \quad M = \begin{matrix} 1 & \frac{1}{8} & \frac{1}{8} \\ 1 & \frac{8}{8} & \frac{1}{8} \\ 1 & \frac{1}{8} & \frac{1}{8} \end{matrix} \quad (1.7)$$

Le Laplacien est un bon détecteur de contours, mais sensible au bruit. Il est donc nécessaire d'effectuer un lissage de l'image avant de calculer l'image du Laplacien, les points de contour sont déterminés par les points (x,y) tels que le Laplacien $L(x,y)$ est "faible" ($|L(x,y)|$ inférieur à L_0) et le gradient $G(x,y)$ "pas trop petit" ($|G(x,y)|$ supérieur à G_0). Selon les valeurs du gradient et du Laplacien en chaque point, il faut donc décider s'il on a affaire à un point de

contour ou non. Vous aurez à déterminer expérimentalement les seuils $L0$ et $G0$ qui permettent de juger si le Laplacien est faible et si le gradient n'est pas trop petit. Ces deux méthodes (maxima du gradient, zéros du Laplacien) reposent sur le même principe qui est de rechercher les maxima locaux de la dérivée première. Pourtant, elles ne sont rigoureusement équivalentes que dans le cas où les contours à détecter ont une faible courbure. La *Figure 1.21* illustre des images où se situent des contours pointus, l'image a été traitée avec le logiciel Khoros, en utilisant le filtre bidimensionnel *GEF edge extract* qui recherche les maxima locaux du gradient (dérivée première).

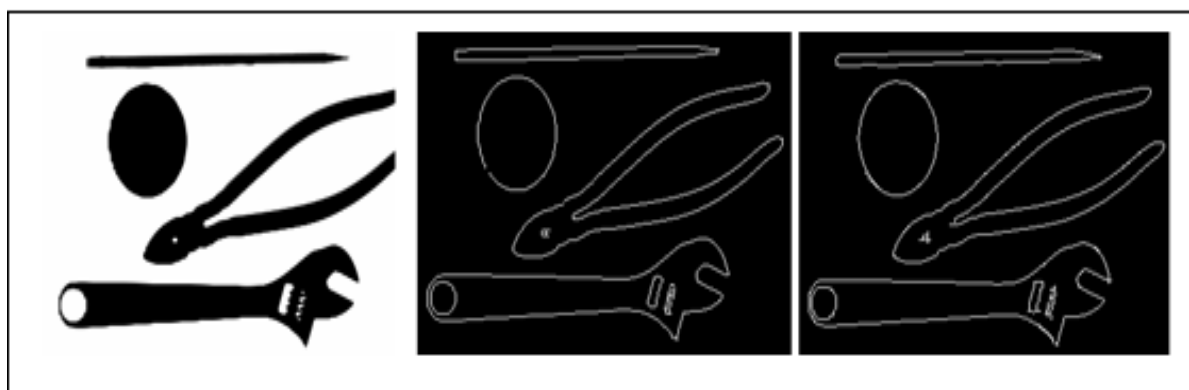


Figure 1.21 – Image originale, maxima du gradient puis zéros du Laplacien

Les deux méthodes donnent de très bons résultats, pourtant nous pouvons observer des différences remarquables en les analysant plus en détail. Il s'agit pour cela de s'intéresser aux parties des contours qui présentent les courbures les plus fortes, si nous nous intéressons au cercle, dont la courbure est constante, les deux méthodes donnent des résultats satisfaisants, ce qui prouve que l'approximation faite était légitime, même pour une courbure relativement importante. Par ailleurs, la méthode du gradient pose le problème de la fermeture du contour du cercle, alors que celle du Laplacien évite ce problème, on peut aussi remarquer (cela se voit en particulier au niveau du crayon) que les contours sont un peu déplacés d'une image à l'autre. La méthode du Laplacien détecte des contours plus à l'extérieur que ceux du gradient, enfin, on remarque que les angles droits du contour gauche du crayon sont respectés par la méthode du gradient, alors que la méthode du Laplacien a tendance à les arrondir, c'est-à-dire, à en réduire la courbure.

1.4.3 Les détecteurs par masquage adapté

Dans ces techniques [Maître, 2003], on recherche en chaque point de l'image la présence d'une configuration conforme à un gabarit appartenant à un dictionnaire de contours. Pour cela, on définit une distance entre un contour type et une fenêtre de l'image, si l'image est dénotée $f(x,y)$ et le $k^{\text{ième}}$ gabarit $G_k(x,y)$ on calcule la valeur R.

$$R_k(x,y) = \sqrt{\frac{(\sum_{x,y \in \text{lafenetre}} f^2(x,y) * \sum_{x,y \in \text{lafenetre}} G_k^2(x,y))}{\sum_{x,y \in \text{lafenetre}} f(x,y)G_k^2(x,y)}} \quad (1.8)$$

On ne conserve que la meilleure des valeurs et seulement si elle est suffisamment faible. L'approche par masques adaptés (*template matching*), permet de connaître aisément la direction du contour, elle permet également de soumettre, en parallèle, ces calculs à des architectures adaptées. Mais la qualité des contours, ainsi que les post- traitements qu'ils requièrent sont très semblables à ceux que l'on a vus pour les filtres de maximum du gradient et les filtres adaptés les plus utilisées sont le filtre de Kirsch, le filtre de Nevatia et Babu.

- **Masques de Kirsch :** Le masque de Kirsch est constitué de deux fenêtre (matrices de 3*3) est avec des rotations de $\pi/4$ pour chaque fenêtre « *Figure 1.22* » il en résultera 8 filtres.

5	5	5	-3	5	5
-3	0	-3	-3	0	5
-3	-3	-3	-3	-3	-3

Figure 1.22 – Les fenêtre de Kirsch

- **Masque de Babu et Nevatia :** Le masque de Babu et Nevatia est constitué de deux fenêtre (matrices de 5*5) est avec des rotations de $\pi/2$ pour la première fenêtre et une rotation de $\pi/4$ pour la deuxième fenêtre « *Figure 1.23* » il en résultera 12 filtres.

-100	-100	0	100	100	-100	32	100	100	100
-100	-100	0	100	100	-100	-78	92	100	100
-100	-100	0	100	100	-100	-100	0	100	100
-100	-100	0	100	100	-100	-100	-92	78	100
-100	-100	0	100	100	-100	-100	-100	32	100

Figure 1.23 – Les fenêtré de Babu et Nevatia

1.4.4 Le filtre de Canny

Canny a proposé un filtre calculé à partir de trois critères non antagonistes [Bourennane & al, 2002], en exprimant ces trois critères sous la forme de l'optimisation conjointe de trois fonctionnelles, et il a été démontré qu'il était linéaire et optimal pour la détection d'une marche d'escalier sous l'hypothèse d'un bruit additif indépendant du signal. Les critères de Canny consistent en :

- Garantir une bonne détection de contour, même à des niveaux faibles, la probabilité de détecter un contour vrai doit être maximale et celle de détecter un contour faux dû au bruit doit être minimale. Cela revient à maximiser le rapport signal sur bruit, donc à maximiser le critère de bonne détection :

$$f_1 = \frac{\int_0^{\infty} \theta(x) dx}{\sqrt{\int_{-\infty}^{\infty} \theta^2(x) dx}} \quad (1.9)$$

- Garantir une bonne localisation des contours, en présence de bruit, le maximum de la réponse du filtre est déplacé d'une quantité aléatoire. Pour que la localisation du contour soit précise, il faut que la variance de ce déplacement soit la plus petite possible, cela revient à maximiser le critère :

$$f_2 = \frac{\theta'(0)}{\sqrt{\int_{-\infty}^{\infty} \theta^2(x) dx}} \quad (1.10)$$

- Assurer une et une seule détection par contour, la réponse du filtre ne doit présenter qu'un seul maximum dans le voisinage d'un contour. Cependant, le bruit crée des maximums non désirés distants en moyenne d'une quantité f_3

Donc pour résoudre ce problème on doit maximiser f_1 et f_2 sous la contrainte f_3 et θ étant la réponse impulsionnelle du filtre 1D, l'application en 2D « *Figure 1.24* » est effectuée par action de deux filtres croisés en x et y .

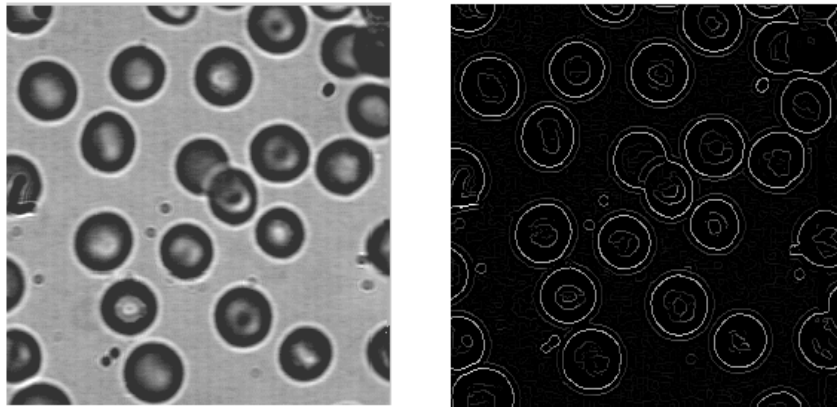


Figure 1.24 – Résultat du filtre de Canny (image à droite)

1.4.5 Le filtre de Canny - Deriche

Ce filtre [Laurent HAAS & all, 1997] permet de détecter les contours d'une image bruitée et floue en tenant compte de paramètres liés au rapport signal sur bruit et à la localisation des contours, ce filtre fonctionne à partir d'un modèle de contour $C(x)$ qui a pour équation :

$$C(x) = \begin{cases} 1 - \frac{e^{-sx}}{2} & x \geq 0 \\ \frac{e^{sx}}{2} & x \leq 0 \end{cases} \quad (1.11)$$

On cherche donc à maximiser les mêmes critères que ceux de Canny, le rapport signal sur bruit (RSB), la localisation (L) et la suppression des réponses multiples (MRC). Ce filtre récursif est donc d'ordre 3 et nécessite 4 lectures de l'image (de gauche à droite, de droite à gauche, de haut en bas et de bas en haut), à chaque lecture une fonction du filtre est calculée et c'est la somme de ces 4 fonctions qui donne le contour final, il faut encore faire subir à l'image du contour un processus identique mais de lissage, qui se calcule de la même façon

que le gradient, pour une détection optimale des contours d'images bruitées, les résultats obtenus «*Figure 1.25*» sont comparables ou meilleurs que ceux du filtre de Canny.

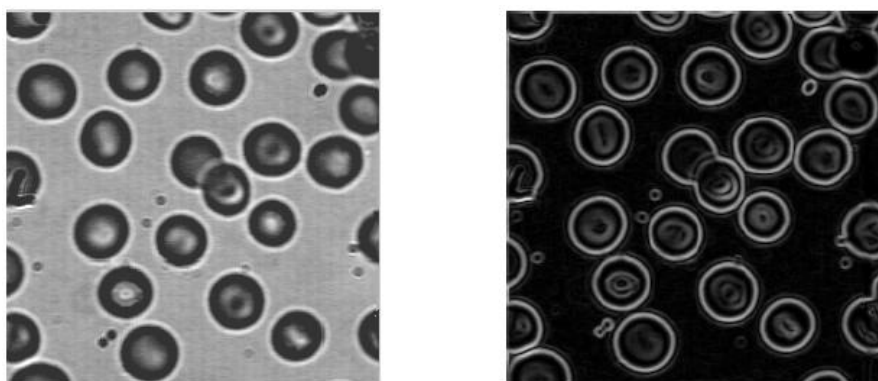


Figure 1.25 – A droite, l'image filtrée avec l'algorithme de Canny-Deriche

1.5 Détection de points d'intérêts

La détection de points d'intérêts [Edmond BOYER, 2004] est, au même titre que la détection de contours, une étape préliminaire à de nombreux processus de vision par ordinateur, les points d'intérêts, dans une image, correspondent à des doubles discontinuités de la fonction d'intensités et celles-ci peuvent être provoquées, comme pour les contours, par des discontinuités de la fonction de réflectance ou des discontinuités de profondeur, ce sont par exemple les coins, les jonctions en T ou les points de fortes variations de texture. De nombreuses méthodes ont été proposées pour détecter des points d'intérêts, elles peuvent être classées grossièrement suivant trois catégories :

- **Approches contours** : l'idée est de détecter les contours dans une image dans un premier temps. Les points d'intérêts sont ensuite extraits le long des contours en considérant les points de courbures maximales ainsi que les intersections de contours.
- **Approches intensité** : l'idée est cette fois-ci de regarder directement la fonction d'intensité dans les images pour en extraire directement les points de discontinuités.

- **Approches à base de modèles** : les points d'intérêts sont identifiés dans l'image par mise en correspondance de la fonction d'intensité avec un modèle théorique de cette fonction des points d'intérêts considérés.

Les approches d'intensité sont celles utilisées généralement ceci est dû à l'indépendance vis à vis de la détection de contours (stabilité) et l'indépendance vis à vis du type de points d'intérêts (méthodes plus générales).

1.5.1 Le détecteur de Moravec

L'idée du détecteur de Moravec est de considérer le voisinage d'un pixel (une fenêtre) et de déterminer les changements moyens de l'intensité dans le voisinage considéré lorsque la fenêtre se déplace dans diverses directions. Plus précisément on considère la fonction :

$$E(x,y) = \sum w(u,v) |I(x+u, y+v) - I(u,v)|^2 \quad (1.12)$$

Où w spécifie la fenêtre (voisinage) considérée, $I(u,v)$ est l'intensité au pixel (u,v) et $E(x,y)$ représente la moyenne du changement d'intensité lorsque la fenêtre est déplacée de (x,y) .

En appliquant cette fonction dans les trois situations principales suivantes « *Figure 1.26* » on obtient les situations suivantes:

- L'intensité est approximativement constante dans la zone image W considérée et la fonction E prendra alors de faibles valeurs dans toutes les directions (x,y) .
- La zone image considérée contient un contour rectiligne, la fonction E prendra alors de faibles valeurs pour des déplacements (x,y) le long du contour et de fortes valeurs pour des déplacements perpendiculaires au contour.
- La zone image considérée contient un coin ou un point isolé et la fonction E prendra de fortes valeurs dans toutes les directions.

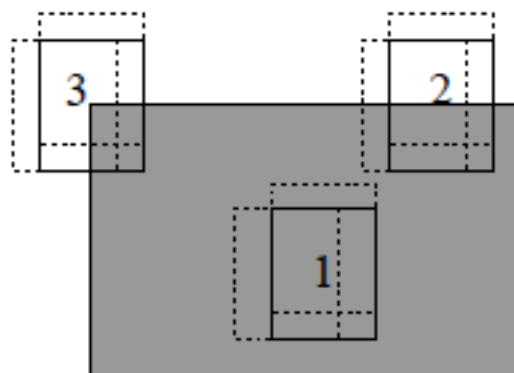


Figure 1.26 – Les différentes situations considérées par le détecteur de Moravec

En conséquence, le principe du détecteur de Moravec est donc de rechercher les maxima locaux de la valeur minimale de E en chaque pixel (au dessus d'un certain seuil).

1.5.2 Le détecteur de Harris

Le détecteur de Moravec fonctionne dans un contexte limité mais souffre en effet de nombreuses limitations, Harris et Stephen [Harris & al, 1988] ont identifié certaines limitations et en les corrigeant, en ont déduit un détecteur de coins très populaire. La réponse du détecteur de Moravec est bruitée en raison du voisinage considéré, le filtre w utilisé est en effet binaire (valeur 0 ou 1) et est appliqué sur un voisinage rectangulaire, pour améliorer cela, Harris et Stephen propose d'utiliser un filtre Gaussien. Le détecteur de Moravec répond de manière trop forte aux contours en raison du fait que seul le minimum de E est pris en compte en chaque pixel, pour prendre en compte le comportement général de la fonction E localement E devient :

$$E(x, y) = (x, y) \cdot M \cdot (x, y)^t \quad (1.13)$$

Où M est une matrice caractérise le comportement local de la fonction E , les valeurs propres de cette matrice correspondent en effet aux courbures principales associées à E

- Si les deux courbures sont de faibles valeurs, alors la région considérée a une intensité approximativement constante.
- Si une des courbures est de forte valeur alors que l'autre est de faible valeur alors la région contient un contour.

- Si les deux courbures sont de fortes valeurs alors l'intensité varie fortement dans toutes les directions, ce qui caractérise un coin.

Par voie de conséquence, Harris et Stephen propose l'opérateur suivant pour détecter les coins dans une image :

$$Det(M) = AB - C^2 \quad \text{et} \quad Trace(M) = A + B \quad \text{et} \quad R = Det(M) - kTrace(M)^2 \quad (1.14)$$

Les valeurs de R sont positives au voisinage d'un coin, négatives au voisinage d'un contour et faibles dans une région d'intensité constante.

1.5.3 Le détecteur SIFT

L'algorithme SIFT (Scale Invariant Feature Transform) a été proposé par David Lowe, université de British Columbia [Lowe, 2004] pour détecter et décrire des zones d'intérêts (local features) dans une image, il s'agit ici non seulement de détecter mais aussi de caractériser, par des valeurs, pour pouvoir reconnaître (mettre en correspondance) par la suite ces zones ou points d'intérêts « *Figure 1.27* » dans d'autres images de la même scène. L'idée générale de SIFT est de trouver des features qui sont invariants à plusieurs transformations (rotation, échelle, illumination et changements mineurs du point de vue), le principe de la détection est donc de trouver les extrema dans l'espace-échelle Gaussien (au lieu de regarder la fonction LoG qui est coûteuse à calculer, on approxime avec la fonction DoG), les extrema sont alors les pixels qui présentent une intensité maximum, ou minimum, par rapport à leurs voisins directs ainsi qu'à ceux dans l'espace-échelle, (les extrema de l'espace-échelle ainsi obtenus sont nombreux, pour les filtrer les candidats avec peu de contraste sont éliminés et les réponses correspondants à des contours sont éliminés)



Figure 1.27 – Résultats de détection de contour avec Harris puis Sift.

Les principaux avantages des points d'intérêts sont :

- Sources d'informations plus fiables que les contours car plus de contraintes sur la fonction d'intensité.
- Robuste aux occultations (soit occulté complètement, soit visible).
- Pas d'opérations de chaînage et Présents dans une grande majorité d'images (Contrairement aux contours).

1.6 La poursuite et la fermeture des contours

Si l'on excepte les contours actifs et les passages par zéro des laplaciens, la plupart des détecteurs de contours fournissent des contours ouverts, c'est-à-dire qu'ils ne séparent pas les composantes de l'image dans des objets topologiquement distincts, cela rend souvent plus difficile l'étape de reconnaissance des formes qui doit suivre, pour cela, on a proposé plusieurs méthodes permettant d'obtenir des contours clos, c'est ce que l'on appelle la fermeture de contours. De nombreuses solutions ont été proposé pour cette étape fondamentale, reposant sur des principes très différents et la qualité des résultats obtenus est généralement directement liée au coût informatique consenti, de très bonnes solutions existant, mais à des coûts prohibitifs pour la plupart des applications.

1.6.1 Méthodes de recherche dans des graphes

Ce sont les méthodes qui reposent sur les bases les plus solides et susceptibles de donner les meilleurs résultats, mais leurs coûts sont souvent très élevés, car elles se proposent d'explorer l'espace de toutes les solutions et de retenir la meilleure en fonction d'un critère que l'opérateur s'est donné, l'écueil de la recherche combinatoire exhaustive peut être évité par des recherches polynomiales assez complexe, surtout fondé sur la programmation dynamique. L'approche de type Viterbi [Chatterjee & al, 2011] considère l'évolution pas à pas d'un contour, à chaque pas le coût du contour est évalué pour toutes les provenances possibles du contour et n'est retenu pour tout point du parcours que le seul chemin qui minimise le coût pour arriver à ce point, on garde alors en chaque point ce coût minimum ainsi que l'adresse du père qui a permis d'y aboutir, un nouveau pas est alors fait et toutes les destinations sont explorées que l'on peut atteindre en un pas à partir des points déjà atteints.

1.6.2 Méthodes de recherche par les automates

Ces solutions, au contraire des précédentes, ne proposent que des solutions très sous-optimales, fondées sur de seuls critères locaux (il n'y a pas d'optimum global comme par programmation dynamique), elles sont particulièrement rapides de mise en œuvre, mais peuvent parfois faire diverger la combinatoire si l'on gère mal la prolifération des chemins par ramification, les résultats d'une détection par masquage adaptés sont les candidats idéaux à ces techniques de fermeture puisqu'ils fournissent des directions de recherche. Les diverses méthodes se distinguent par le choix des successeurs d'un point (parmi des configurations prédéfinies) par le mode de balayage de l'image l'un horizontal l'autre vertical il est également important de définir des fonctions de mérite pour valider les chaînes ainsi trouvées, car les automates, comme la programmation dynamique, trouvent toujours un contour, mais ne garantissent pas toujours qu'il corresponde véritablement au contenu de l'image.

1.7 Conclusion

La segmentation est une des étapes critiques de l'analyse d'images qui conditionne la qualité des mesures effectuées ultérieurement. Elle permet d'isoler dans l'image les objets sur lesquels doit porter l'analyse, de séparer, dans une image, les régions d'intérêt du fond. Comme nous l'avons vu dans ce chapitre il existe plusieurs techniques de segmentations, cependant, on ne sait jamais dire quelle est la segmentation idéale. L'idée est bien sûr que la région se rapproche de la notion d'objet, au sens courant du terme. Néanmoins, on peut dégager des propriétés plus raisonnables qu'on cherche à obtenir dans un algorithme de segmentation, tel la stabilité (la segmentation obtenue ne doit pas varier beaucoup lorsque les conditions d'acquisition varie légèrement) et la régularité des régions obtenues qui doivent être simples à manipuler. Dans le chapitre suivant nous allons présenter l'image de profondeur ainsi que les différents types d'images de profondeur existantes, nous allons aussi présenter les différentes techniques et équipements utilisés pour acquérir une image de profondeur.

Chapitre 2

L'image de profondeur

2.1 Introduction

Durant cette dernière décennie, l'image de profondeur a été largement utilisée afin d'obtenir des modèles 3D d'objets du monde réel. Au cours des dernières années, la technologie à base d'images de profondeur a ouvert quelques voies de recherche dans la vision par ordinateur et les champs d'infographie [Yamauchi & al, 2010], principalement axé sur le traitement de la géométrie et topologie des images de profondeur, par exemple dans la réduction de bruit, l'analyse, la segmentation, la classification et la reconstruction d'un modèle 3D. L'un des plus grands défis des dispositifs modernes d'imagerie de profondeur a été l'acquisition d'images de qualité en temps réel. Le résultat d'une segmentation d'une image de profondeur dépend en grande partie de sa qualité d'acquisition et une image fortement bruitée peut sérieusement affecter le résultat de segmentation obtenu. Dans ce chapitre, nous allons présenter l'image de profondeur ainsi que ces diverses propriétés, nous aborderons aussi les types d'image de profondeurs et leurs représentations graphiques, puis, nous nous intéresserons aux différents systèmes d'acquisition d'images de profondeur modernes. Pour finir nous allons présenter les différentes techniques d'estimation de bruit utilisé en imagerie de profondeur.

2.2 Définition

Une image de profondeur (IP) est un tableau bidimensionnel de positions 3D [Natonek, 1996] vérifiant la propriété de cohérence spatiale, chaque composante de cette matrice représente la distance entre un point de référence(en générale le capteur lui-même) et un point dans le champ de vision du capteur (Appareil photo, Camera), elle est l'équivalent d'une image vidéo [Chaperon, 2002] dans laquelle le niveau de gris de chaque pixel (x,y) serait remplacé par une altitude z (Le terme *2.5D* provient du fait que cette situation est un intermédiaire entre les cas *2D* et *3D* et ne concerne pas les dimensions fractales). La particularité de ce type de données réside donc essentiellement dans sa structure de grille (x,y) et dans sa possibilité de décrire la scène comme un graphe de fonction $z = f(x,y)$. Le terme anglais de *range image element* désigne un élément du tableau. Les images de profondeur « *Figure 2.1* » peuvent être vues comme des nuages de points 3D [Tarel, 1996] et ont une représentation

régulière [Chevalier & al, 2005] et sont considérées comme organisées dans le sens où des points voisins dans l'image de profondeur sont aussi voisins dans l'espace.

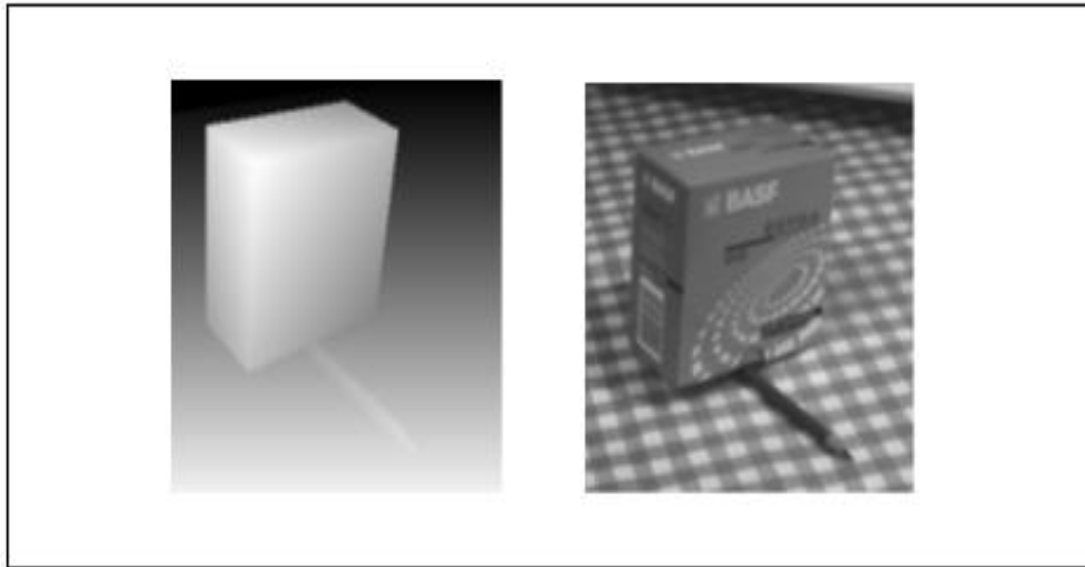


Figure 2.1 – A gauche une image de profondeur, à droite une image d'intensité

Dans les travaux de [Jorge Hernández & al, 2008] une image d'accumulation des nuages de points est associée à l'image de profondeur. En utilisant une caméra virtuelle positionnée sur le plan XY, l'image de profondeur enregistre la distance maximale des points 3D projetés sur le plan de la caméra, et l'image d'accumulation compte le nombre de points (3D) qui sont projetés sur le même pixel de l'image de caméra. Les images de profondeur offrent une correspondance précise grâce aux dernières techniques de balayage laser [Liu, 2009], de ce fait l'image de profondeur est utilisée dans de nombreux domaines comme la reconnaissance d'image, l'extraction d'images, la transmission d'images, la modélisation d'objet, la localisation simultanée et la construction d'une carte.

2.3 Profondeur d'une image

La profondeur est la distance entre la surface visible d'un objet dans une scène et le détecteur de la caméra, elle est une indication utile pour le calcul des coordonnées des points appartenant à cette surface dans un espace tridimensionnel de référence. Avec la demande croissante de contenus 3D, la conversion de nombreuses données 2D existantes en données 3D a gagné un grand intérêt dans le traitement de l'image 3D. De nombreuses méthodes ont été développées [Bicking & al, 2007] pour obtenir les coordonnées 3D des objets en utilisant

des images bidimensionnelles, toutes exploitent les variations des paramètres d'acquisition du système de prise de vue. Ces paramètres d'acquisition du système ou de l'environnement lumineux apportent l'information essentielle pour établir un rapport entre l'image et la scène réelle. D'autres recherches permettent l'estimation de la profondeur d'une image en convertissant des images bidimensionnelles en données tridimensionnelles. Cette conversion « *Figure 2.2* » se fait grâce à une analyse en ondelettes bidimensionnelles [Guo & al, 2008].



Figure 2.2 – A gauche une image d'intensité, à droite une image convertie par ondelettes bidimensionnelles

[KO & al, 2007] Propose une méthode de conversion automatique qui estime les informations de profondeur d'une image mono-vue en fonction du degré de concentration des régions segmentées, pour ensuite générer une image stéréoscopique. L'image est d'abord partitionnée en régions homogènes. Ensuite une carte de statistiques d'ordre supérieur (HOS) est créée, cette carte représente la répartition spatiale de haute fréquence constituant l'image d'entrée. La carte HOS est connue pour être bien adaptée pour résoudre les problèmes de détection et de classification, car elle peut supprimer le bruit gaussien tout en préservant une partie de l'information non-gaussienne. Enfin, une image stéréoscopique est générée « *Figure 2.3* » par calcul des valeurs de parallaxe de chaque région en utilisant la carte de profondeur générée et l'image d'entrée.



Figure 2.3 – A gauche une image 2D, à droite une image convertie par carte HOS

[Kamencay & al, 2012] Propose une méthode d'estimation de la profondeur en effectuant tout d'abord une segmentation de l'image basée sur la couleur, le but de cette segmentation est de simplifier la représentation de l'image à traitée et de faciliter l'analyse. Enfin, pour définir la profondeur de chaque pixel, une méthode d'estimation de disparité est appliquée à chaque pixel des images segmentées dans l'étape précédente. Cet algorithme a été testé sur une vraie scène « *Figure 2.4* » avec des conditions d'éclairage médiocres.

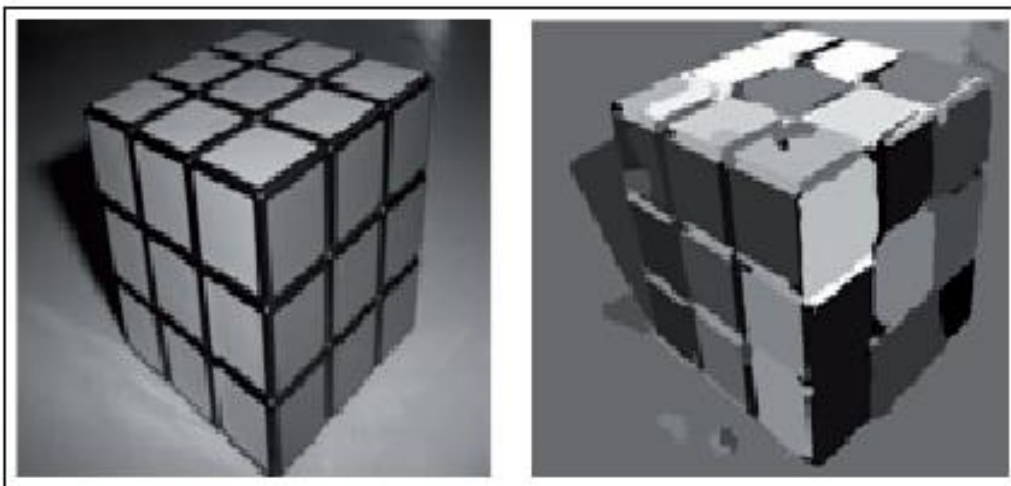


Figure 2.4 – A gauche une image 2D, à droite une image convertie par segmentation hybride

2.4 Propriété de cohérence spatiale

La cohérence spatiale est en premier lieu une propriété extrinsèque à l'IP, liée aux conditions de son acquisition [Villiger & al, 2010]. La cohérence spatiale est seulement en deuxième lieu une propriété intrinsèque aux données que constitue l'image de profondeur, conséquence des conditions d'acquisition. Le système de mesure fixe un faisceau de droites dans R^3 , une droite par position du tableau, les rangels sont interprétés comme l'intersection de ces droites avec une surface continue qui est l'objet de la mesure « Figure 2.5 ».

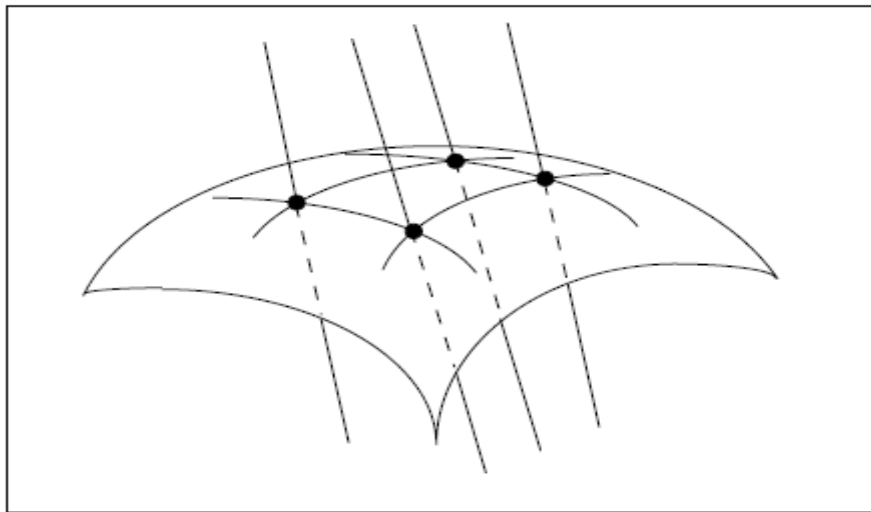


Figure 2.5 – Illustration de la cohérence spatiale

- Aspect intrinsèque : Au strict niveau des données de l'image de profondeur, la propriété de cohérence spatiale signifie que le rangel qui est le plus proche d'un autre dans l'espace $3D$ (distance euclidienne) est un de ses voisins immédiats dans le tableau et que le rangel voisin d'un autre dans l'image de profondeur est parmi les 9 plus proches dans l'espace $3D$.
- Aspect extrinsèque : A ce niveau pour satisfaire la propriété de cohérence spatiale, le faisceau de droites fixé par le système de mesure doit être quasi-parallèle l'une à l'autres et deux droites du faisceau qui sont voisines dans l'espace $3D$ doivent correspondre à des positions voisines dans le tableau et vice-versa.

La cohérence spatiale est une propriété importante de l'image de profondeur [Saastamoinen & al, 2013] sur laquelle reposent plusieurs procédés appliqués aux images de profondeurs telles :

- L'interpolation: On construit une surface représentative de la surface mesurée à partir de l'IP simplement en "remplissant" les mailles du treillis dont les nœuds sont les rangels. Ce procédé n'est pas possible pour une liste quelconque de points, dans ce cas il faut trouver les points les plus proches entre eux dans l'espace 3D pour construire un maillage. Ce problème peut être résolu, par exemple, par une triangulation de Delaunay.
- Représentation par une surface d'approximation: Un ensemble de rangels voisins sont représentés par une surface algébrique d'approximation.
- Estimation de l'erreur de mesure: La partie du signal non compatible avec la cohérence spatiale est assimilée à l'erreur de mesure

2.5 Systèmes d'acquisition d'images de profondeur

Au cours des 30 dernières années, le développement de systèmes d'acquisition d'images de profondeur a fortement progressé. Ces systèmes de mesure sont apparus dans les laboratoires dans les années 80 et s'implantent de nos jours de plus en plus rapidement et de plus en plus massivement dans de nombreux secteurs allant de l'industrie lourde (métallurgie) à la conservation du patrimoine en passant par la biologie. Les capteurs d'image de profondeur permettent de collecter les coordonnées 3D de surfaces réfléchissantes d'objets dans une scène, ces capteurs peuvent être classés comme étant soit active ou passive [Kohlhepp & al, 2009]. Les capteurs actifs projettent un signal sur une scène et mesure la partie du signal qui est réfléchi, radar, sonar et laser sont des exemples de capteurs actifs. Les capteurs passifs fonctionnent en utilisant des conditions environnementales existantes (un système d'imagerie stéréo aérien est un bon exemple). Il existe une grande variété de systèmes d'acquisition d'images de profondeur, ils se distinguent par le procédé de mesure, la géométrie du dispositif, la précision et la rapidité. Ces cameras sont très populaires en raison de leur grande précision [Bhavsar & al, 2010], mais, le processus d'acquisition était souvent très lent ce qui pouvait être accéléré par la capture des données éparées, mais, les capteurs d'image de profondeur évoluent en permanence [Haijiao & al, 2013], ce qui permet de nos jours la numérisation d'une scène pleine de haute résolution en peu de temps, donc, les algorithmes et les techniques mis au point pour traiter des images de profondeur ont également évoluer [Lin & al, 2012] avec le même dynamisme pour faire assurer le traitement de l'information tout rapidement.

- Un autre système [Besl, 1988] expérimental de laboratoire « Figure 2.6 », a été réalisé avec du matériel non spécialisé à l'université de Neuchâtel. Un faisceau laser en forme de plan illumine la scène. Le faisceau dessine une trace visible sur la surface des objets et la trace est repérée dans l'image d'une caméra et amincie à une largeur d'un pixel. Pour chacun des pixels de la trace dans l'image, le point 3D, défini par l'intersection du plan de lumière et de la ligne de vue correspondant au pixel, est calculé et ce point est retenu comme rangel. En traitant de la sorte une image de la caméra, on mesure un profil de la surface des objets correspondant à une section plane et on génère une ligne de l'IP. L'IP est construite profil par profil en translatant l'objet entre deux mesures de profil

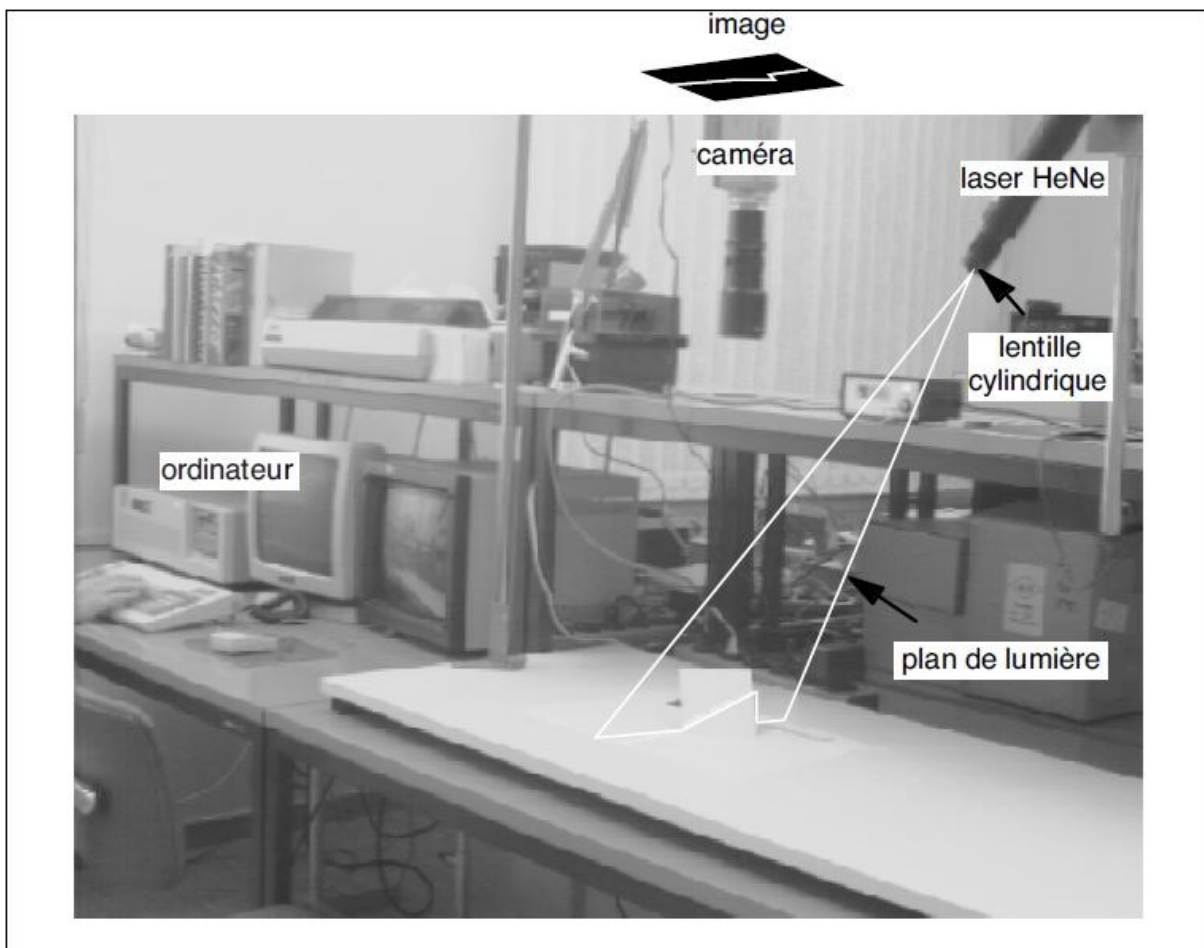


Figure 2.6 – Système expérimental d'acquisition d'IP (Université de Neuchâtel)

- Le Time of Flight « temps de vol » est un système fonctionnant sur le principe de temps de vol « Figure 2.7 ». Un faisceau laser modulé en amplitude est projeté dans une direction [Lange & al, 1999] et un capteur photosensible mesure la lumière

revenant à la source. Grâce à la modulation en amplitude le décalage temporel entre signal lumineux émis et signal lumineux reçu peut être déterminé (à condition qu'il ne dépasse pas la valeur d'une période de modulation). Le décalage temporel est une mesure directe de la distance entre le système d'acquisition et la surface rencontrée par le faisceau laser. La mesure de distance dans une direction permet de définir un point de l'IP, ainsi, l'IP est construite par un balayage du faisceau laser. Ce genre de capteurs est normalement équipé d'un mécanisme de balayage, et l'émetteur et le récepteur sont souvent coaxial, ce qui élimine le problème des données manquantes communs dans les approches de triangulation. La portée maximale de ce capteur est de l'ordre de 300 mètres pour une vitesse d'acquisition maximale d'environ 50 000 points par seconde.

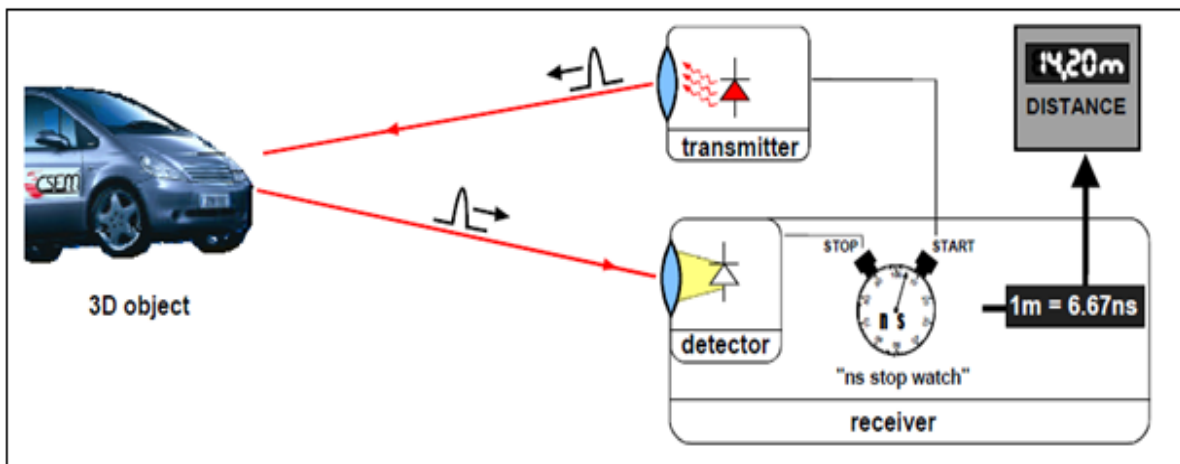


Figure 2.7 Système d'acquisition d'images de profondeur de type Time of Flight

- Le FLRS a été développé [Ikeuchi & al, 2008] pour mesurer des objets à grande échelle en altitude en utilisant un ballon « *Figure 2.8* » de base plutôt que de construire des échafaudages.



Figure 2.8 – Système FLRS embarqué dans un ballon

En ce qui concerne le principe de la mesure, la méthode stéréoscopique passive pourrait capturer des images « *Figure 2.9* » sans l'influence du mouvement du ballon. Le procédé de radar à laser est adapté à la mesure extérieure des objets de grande taille.



Figure 2.9 – Image de profondeur acquise par le système FLRS

- Une caméra stéréo [Somanath & al, 2010] est un type de caméra avec deux ou plusieurs lentilles ayant un capteur d'image séparée pour chaque lentille, cette technique permet à l'appareil de simuler la vision binoculaire humaine et donne la possibilité de capturer des images en trois dimensions (ce procédé qui est connu comme la photographie stéréo). Les caméras stéréo peuvent être utilisées pour faire des photos 3D ou pour acquérir une image de profondeur« *Figure 2.10* ». Les caméras stéréo sont parfois montées dans les voitures pour détecter la largeur de la voie et de la proximité d'un objet sur la route.



Figure 2.10 – Image de profondeur acquise par une stéréo camera

- Un autre système d'acquisition d'image de profondeur « Cam Cube » fournis en temps réel des observations liée a la distance de l'objet et peut capter jusqu'à 50 images par seconde [Kohoutek & al, 2010], il permet d'acquérir une image d'amplitude et une image de profondeur, la luminosité locale ainsi que les distances de chaque pixel sont codés en niveaux de gris « Figure 2.11 ». Pour chaque pixel, la distance est mesurée directement par le calcul de la différence entre la phase d'émission et la phase de réception du signal modulé réfléchi. La carte de phase et une carte complète de la distance peuvent être acquises par la détection du retard de phase entre les deux signaux.



Figure 2.11 – Système d'acquisition d'images de profondeur Cam Cube, à gauche l'image réelle, à droite l'image de profondeur acquise par Cam Cube

- Une nouvelle technologie [Kell & al, 2011] de temps de vol le TCSPC (Time Correlated Single Photon Counting) exploite la détection de photon à l'aide d'une méthode d'échantillonnage statistique [Buller & al, 2007]. Cette technique offre plusieurs avantages en comparaison à la méthode temps de vol classique comme une très grande précision temporelle (et donc en distance) ainsi qu'une très grande sensibilité de détection « Figure 2.12 » et la possibilité de travailler sur des objets d'échelles très différentes (de quelques centimètres à plusieurs mètres). Cette

technologie est pour le moment très peu développée et n'est restée qu'à l'état de prototype de laboratoire (Université d'Heriot-Watt, Ecosse).

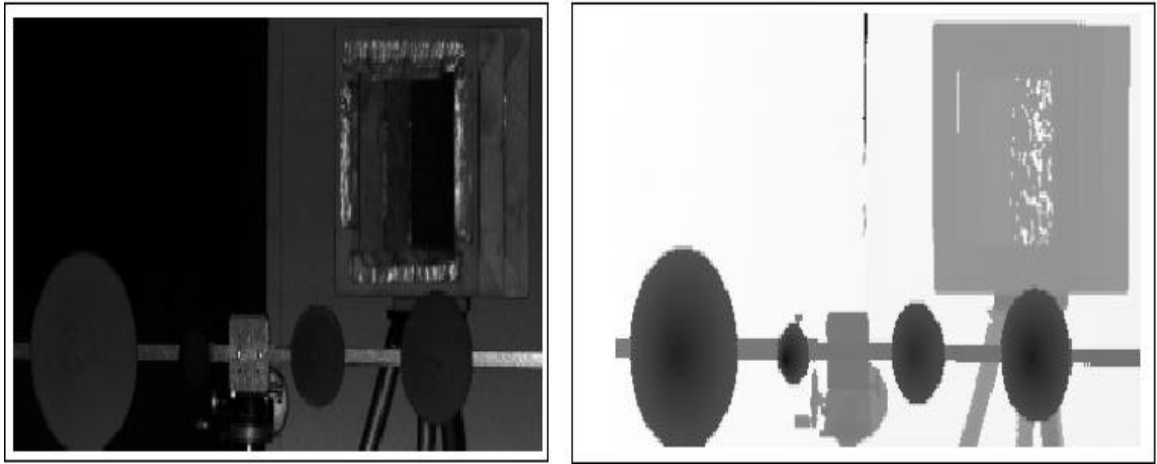


Figure 2.12 – Acquisition d'images de profondeur. A gauche la scène réelle, à droite l'image de profondeur acquise par le TCSPC

2.6 Types d'images de profondeur

Parmi les images de profondeur couramment utilisées dans les milieux de la recherche en vision artificielle, on retient les quatre types suivants [Bowyer & al, 2006] :

- **L'image de profondeur brute $R [i,j]$** : Les droites correspondant aux positions du tableau sont organisées en un faisceau régulier « *Figure 2.13* » convergeant en un point, les rangels sont des valeurs de distance prise de ce point de référence, en connaissant la position du point de référence et la direction des droites, on pourrait reconstruire la position 3D des points mesurés et C'est à ce type d'IP que le terme général est le plus approprié, puisqu'il s'agit littéralement d'un tableau de profondeurs.

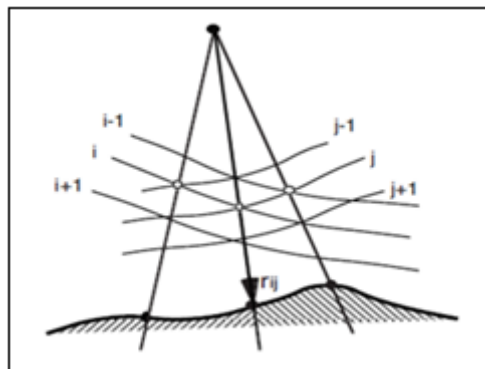


Figure 2.13 – Image de profondeur brute

- **L'image de profondeur cartésienne $X [i, j]$** : Un point de mesure est représenté par un vecteur x de coordonnées cartésiennes « *Figure 2.14* » et le fait que les éléments du tableau soient des vecteurs de troisième dimension et non des scalaires constitue la particularité de ce type d'image.

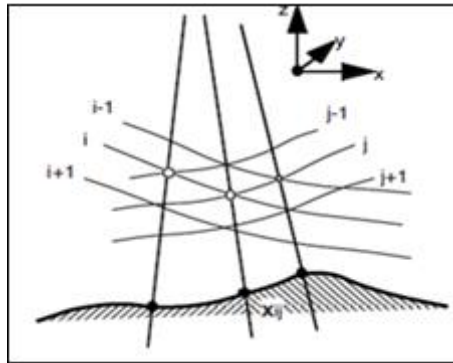


Figure 2.14 Image de profondeur cartésienne

- **L'image de profondeur cartésienne orthographique ($Z [i, j], \Delta x, \Delta y$)** : Dans ce type d'image « *Figure 2.15* » les positions 3D sont des coordonnées cartésiennes et le tableau ne contient cependant que la coordonnée en z , les coordonnées en x et en y sont proportionnelles aux indices du tableau i et j , respectivement:

$$X [i, j] = \Delta x \cdot i \quad \text{ET} \quad Y [i, j] = \Delta y \cdot j \quad (2.1)$$

Les facteurs de proportionnalité Δx et Δy sont les intervalles d'échantillonnage et doivent être associés au tableau $Z [i, j]$ pour que l'information de position soit complète, d'où la notation utilisée.

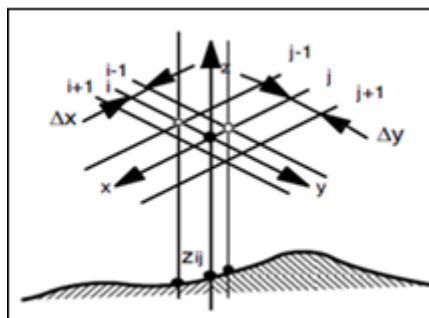


Figure 2.15 Image de profondeur cartésienne orthographique

- **L'image de profondeur cartésienne orthonormée $Z [i, j]$** : les images de profondeur de ce type « *Figure 2.16* » forment un sous-ensemble des images de profondeur

cartésiennes orthographiques, leur particularité étant que les intervalles d'échantillonnage en x et en y sont égaux et valent 1 (le nombre est exempt d'unité, l'unité physique est associée au repère cartésien où $\Delta x = \Delta y = 1$).

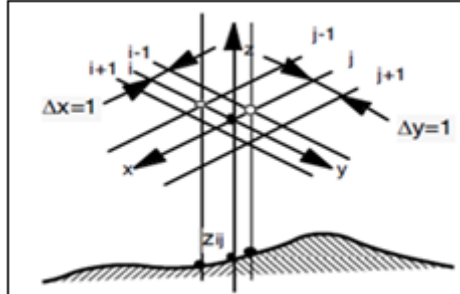


Figure 2.16 Image de profondeur cartésienne orthonormée

On remarque dans « *Tableau 2.1* » que les images de profondeur peuvent être classées en deux catégories selon la dimension de leurs rangels :

- Les images de profondeur vectorielles, notées IP 3D et qui représente les images de profondeur cartésiennes.
- Les images de profondeur scalaires, notées IP 2.5D qui englobent tous les autres types d'images de profondeur.

Tableau 2.1 – Types d'images de profondeur et interprétation des données.

Type d'IP	Dimension des rangels	Transformation des données de l'image en coordonnées cartésienne 3D
IP brute	1 (Scalaire)	<i>Nécessite la connaissance de la géométrie du système d'acquisition utilisé</i>
IP cartésienne	3 (Vecteur)	$X_{i,j} = X_{i,j,1}$ $Y_{i,j} = Y_{i,j,2}$ $Z_{i,j} = Z_{i,j,3}$
IP cartésienne orthographique	1 (Scalaire)	$X_{i,j} = \Delta X i$ $Y_{i,j} = \Delta Y j$ $Z_{i,j} = Z_{i,j}$
IP cartésienne orthonormée	1 (Scalaire)	$X_{i,j} = i$ $Y_{i,j} = j$ $Z_{i,j} = Z_{i,j}$

2.7 Représentations graphiques d'une image de profondeur

Il existe de nombreuses manières de représenter graphiquement une image de profondeur [Levoy & al, 1996], on cite les principales formes de représentation accompagnée d'exemples d'illustration où l'image de profondeur représentée est de type cartésien orthographique (ces images proviennent de la base d'image du Conseil National de Recherches du Canada) :

- Représentation en points 3D (projection des points) « *Figure 2.17.a* »
- Représentation en fil de fer (projection du maillage) « *Figure 2.17.b* »
- Représentation en profils (projection des profils) « *Figure 2.17.c* »
- Représentation en surface neutre (projection du maillage visible) « *Figure 2.17.d* »
- Représentation en rendu réaliste (projection des facettes visibles avec couleur et Ombrage) « *Figure 2.18.a* »
- Représentation en image des courbes de niveaux « *Figure 2.18.b* »
- Représentation en image des profondeurs: niveaux de gris suivant une fonction monotone des profondeurs « *Figure 2.18.c* »
- Représentation en image pseudo-ombrée: niveaux de gris suivant une fonction « *Figure 2.18.d* »

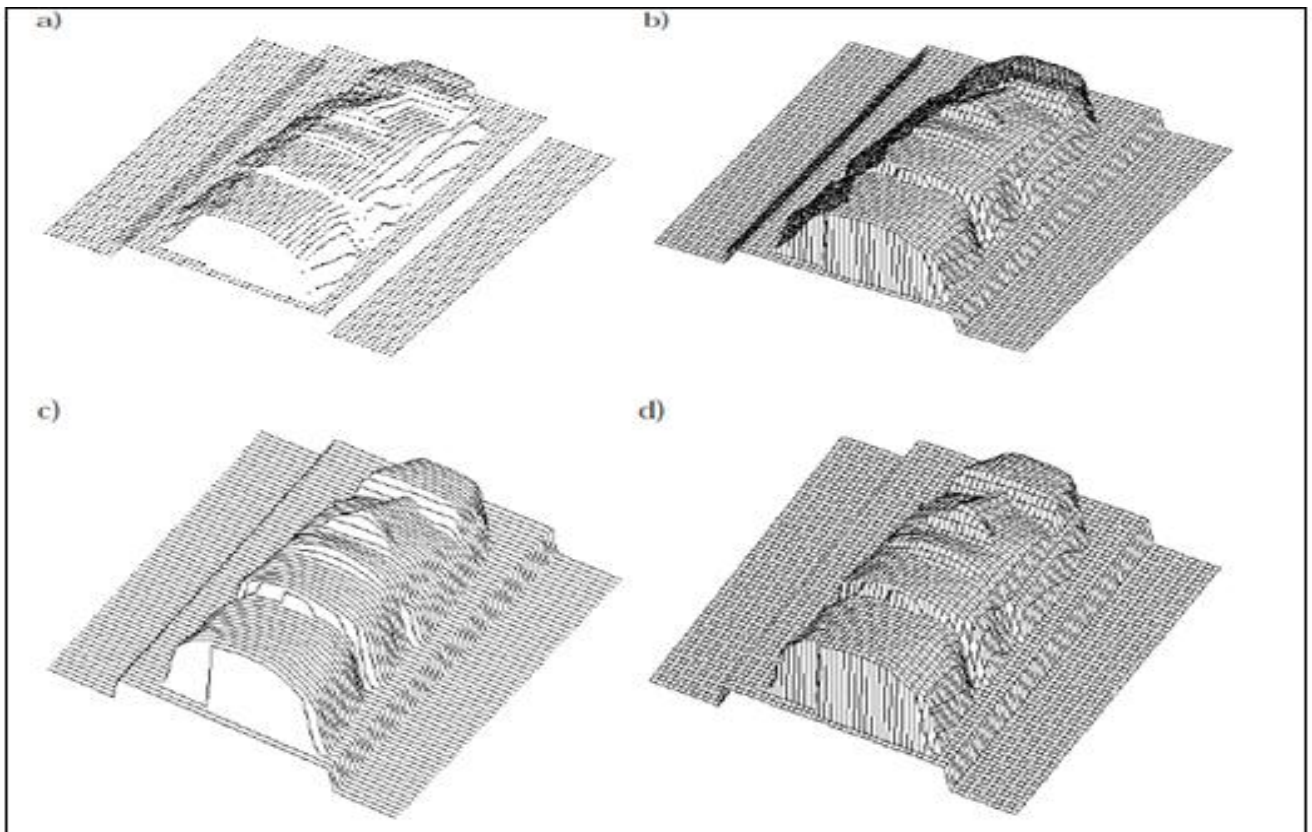


Figure 2.17 – Représentations d'une image de profondeur (a) points 3D (b) fil de fer (c) profils (d) surface neutre.

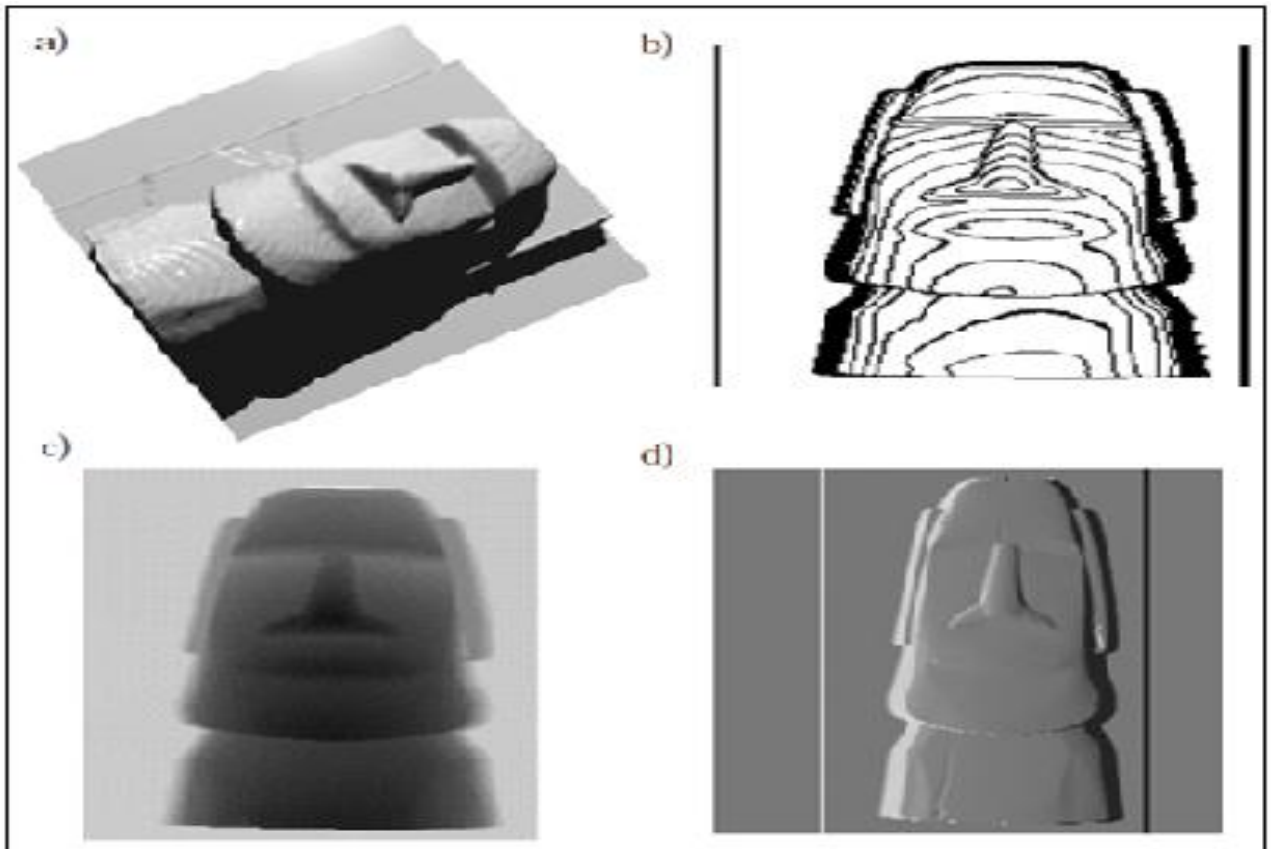


Figure 2.18 – Représentation d'une IP (a) rendu réaliste. (b) courbes de niveaux. (c) niveaux de gris (d) pseudo-ombrée.

2.8 Discontinuités d'une image de profondeur

Les discontinuités d'une image de profondeur [Angel & al, 2004] sont définies par rapport à une surface qui lui est associée en tant que modèle. Pour une définition formelle, on introduit les éléments suivants:

- Le système de coordonnées polaires locales (r, φ) dont l'origine est le point (u, v) .
- La fonction $P(u, v)$, définie sur le domaine Dp du plan des coordonnées d'image (u, v) et représentant la surface.
- La fonction $N(u, v)$ représentant la normale (unitaire) à la surface et définie sur Dp par :

$$N = \frac{P_u * P_v}{\|P_u * P_v\|} \quad (2.2)$$

Où P_u et P_v désignent les dérivées partielles de P par rapport à u et à v .

2.8.1 Discontinuité de profondeur (discontinuité d'ordre 0)

Le point $P(u, v)$ de Dp est un point de discontinuité de profondeur si la fonction P est discontinue (d'ordre 0) au point (u, v) , c'est-à-dire s'il existe des directions φ_1 et φ_2 pour lesquelles les limites de P lorsque r tend vers 0 sont différentes. La propriété de discontinuité de profondeur est notée \bar{C}^0 et $D_{\bar{C}^0}$ désigne le domaine des points de discontinuité de profondeur. Le modèle de point \bar{C}^0 est une conséquence de la géométrie d'acquisition de l'IP et du modèle de surface mesurée continue « Figure 2.19 », les points \bar{C}^0 forment des courbes et les courbes sont généralement fermées, mais pas toujours.

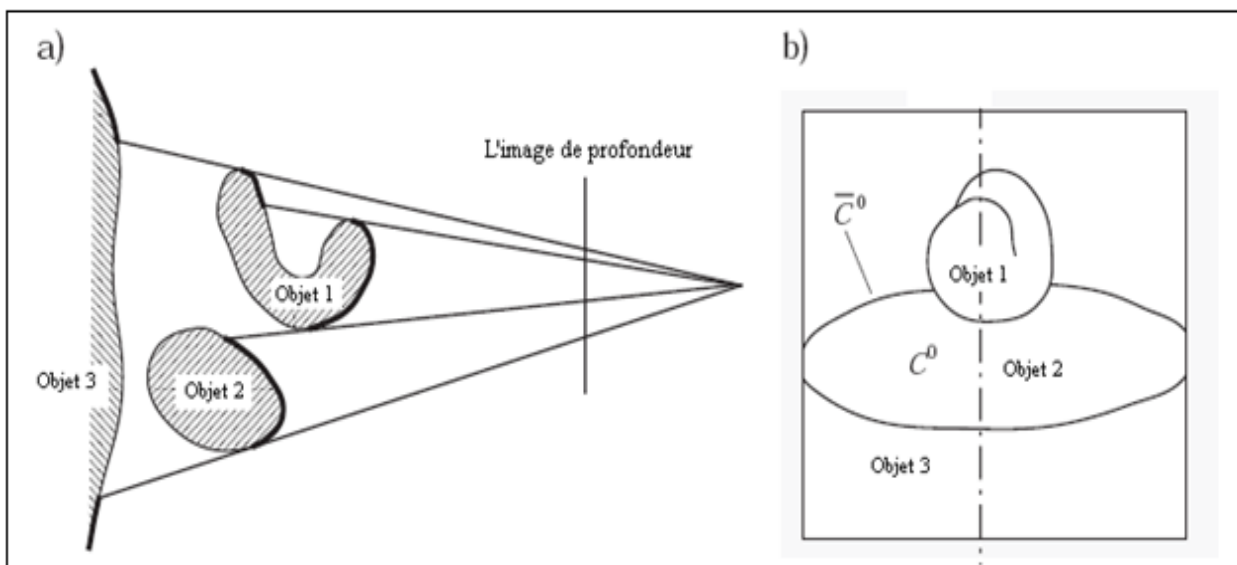


Figure 2.19 – Domaine de définition d'une IP et discontinuités de profondeur, (a) vue en coupe de la scène, (b) plan de l'IP, blanc: domaine de définition $D P$, traits: courbes \bar{C}^0

2.8.2 Discontinuité d'orientation (discontinuité d'ordre 1)

Le point $P(u, v)$ du domaine $(Dp - D_{\bar{C}^0})$ est un point de discontinuité d'orientation si la fonction N est discontinue au point (u, v) , c'est-à-dire s'il existe deux directions φ_1 et φ_2 pour lesquelles les limites de N lorsque r tend vers 0 sont différentes. La discontinuité d'orientation est une discontinuité d'ordre 1 par rapport à la fonction $P(u, v)$, et est notée \bar{C}^1 . Le modèle de point \bar{C}^1 repose sur la géométrie particulière de certains objets répondant au modèle de facettes, Le cas ordinaire de la jonction de deux facettes [Ballard & al, 1982] définit (à la surface de l'objet) une courbe \bar{C}^1 , pour des facettes planes, c'est toujours le cas, cependant, il existe des jonctions de facettes pour lesquelles les points de la courbe ne sont pas tous \bar{C}^1 « Figure 2.20 » il existe aussi des cas où aucun point de la courbe n'est \bar{C}^1 . Aux points pour

lesquels il n'y a pas de discontinuité d'ordre 0 ou 1, la surface $P(u,v)$ est continue d'ordre 1. Cette propriété est notée C^1 .

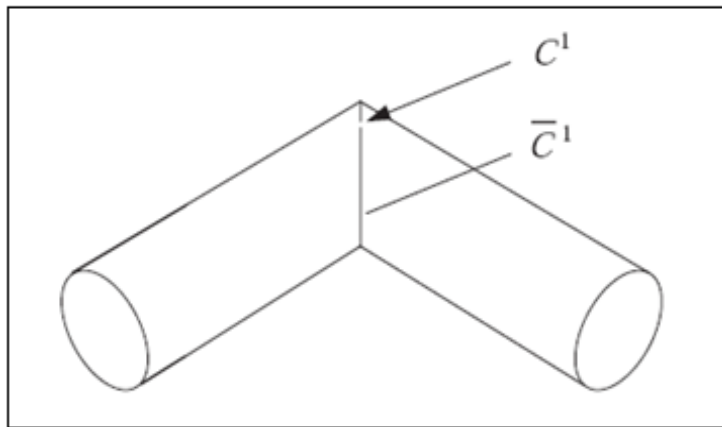


Figure 2.20– Modèle d'objet à facettes, point singulier le long d'une courbe $\overline{C^1}$.

2.9 Erreur de mesure d'une image de profondeur

Une image de profondeur est, comme toute mesure, entachée d'erreur. Il existe différentes interprétations de l'erreur de mesure. Vu l'échantillonnage de mesures dense qui caractérise une image de profondeur, l'intérêt de la mesure d'un point se trouve plus dans le positionnement du point relativement à la surface que dans son positionnement en absolu « Figure 2.21 ». L'erreur significative est donc l'écart du point de mesure $P_{i,j}$ par rapport à la surface échantillonnée. Cette grandeur est appelée erreur apparente. L'erreur apparente peut être interprétée de trois manières différentes

- L'erreur apparente normale qui est la distance (orthogonale) signée qui sépare le point P de mesure de la surface mesurée.
- L'erreur apparente de distance qui est la distance signée qui sépare le point P de la surface mesurée, le long de la droite définie par la direction de mesure.
- L'erreur apparente dans la direction est la distance signée qui sépare P du point de la surface ayant mêmes coordonnées.

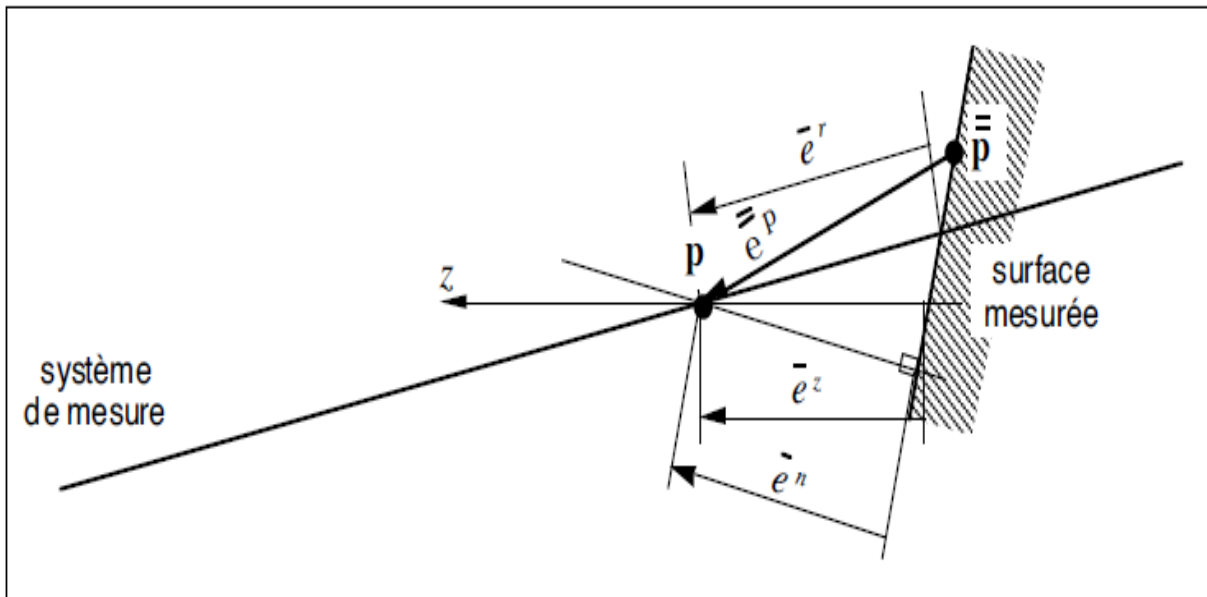


Figure 2.21: Différentes interprétations de l'erreur de mesure.

2.10 Estimation du bruit dans une image de profondeur

Dans la terminologie du traitement du signal, l'erreur de mesure d'une IP est un bruit: composante du signal non issue de la source, alors que la valeur exacte des rangels constitue le signal de source. Nous utilisons donc le terme de bruit comme synonyme d'erreur de mesure. Déterminer l'erreur de mesure d'une IP requiert l'utilisation d'un autre système de mesure plus précis pour indiquer la valeur de mesure exacte. Cependant, quand aucune donnée extérieure au signal mesuré n'est disponible, il faut estimer le bruit sur la base de l'IP seule. L'estimation du bruit repose alors sur des hypothèses concernant le bruit et le signal de source. Nous rapportons trois méthodes d'estimation du bruit tirées de la littérature. Les trois méthodes sont appliquées à des IP de type scalaire. Elles fournissent une estimée de l'erreur de mesure additive.

- Pour Besl [Besl, 1988], c'est aussi l'erreur d'approximation locale par un plan qui sert de base à l'estimation de l'erreur de mesure. A côté de l'erreur quadratique moyenne (EQM) d'approximation, sont retenues ici, à chaque position, les grandeurs déterminant la pente du plan. En éliminant les positions correspondant à un plan de forte pente, l'estimée globale est déterminée soit par la racine carrée de la moyenne des EQM locales, soit par la moyenne des racines carrées des EQM locales.

- Pour l'estimation du bruit dans une image de profondeur *Ade* [*Ade & al, 1990*] utilise l'opérateur 3x3 de moyenne sur les 5 plus proches voisins pour établir une estimation de la valeur exacte du rangel. Le bruit est estimé par la différence entre l'IP originale (valeur mesurée) et l'IP filtrée (estimée de la valeur exacte). La valeur efficace du signal de bruit est retenue comme estimée globale du bruit.
- Jiang et Bunke [*Jian & al, 1994*] estiment localement la surface mesurée par le plan de meilleure approximation. Le niveau local de bruit est estimé par la racine carrée de l'erreur quadratique moyenne (EQM) d'approximation. Pour chaque position de l'IP sont retenues, à côté de l'EQM, les valeurs extrêmes (signées) des erreurs d'approximation. L'estimée globale du bruit est calculée à partir d'une moyenne (sur l'IP) des EQM locales, en écartant les positions pour lesquelles la différence entre valeurs d'erreurs extrêmes est au dessus d'un seuil prédéfini. Plus précisément, l'estimée globale est la racine carrée de la moyenne des EQM locales.

Dans les méthodes d'estimation du bruit par approximation de plan qui viennent d'être décrites, l'erreur d'approximation est définie comme la distance séparant le rangel du point du plan ayant mêmes coordonnées (u,v) . Une alternative consiste à définir l'erreur d'approximation comme la distance du point au plan. Dans ce cas, c'est l'erreur apparente normale qui est estimée. Cette manière de faire, plus coûteuse en calcul, a l'avantage d'être applicable aussi aux IP de type vectoriel. Pour l'essentiel, les méthodes se ressemblent. Leur principe commun s'exprime en trois points:

- Estimation globale par moyennée sur l'image.
- Elimination/atténuation de certaines valeurs locales de grande amplitude.
- Estimation locale par mesure de la variation du signal relativement à une surface localement plane, l'estimée s'annule lorsque l'image de profondeur correspond à un plan échantillonné (non quantifié).

Autrement dit, toutes les méthodes sont basées sur un modèle (hypothèse) de surface mesurée plane et de bruit stationnaire et elles possèdent un moyen d'éliminer l'influence de certains écarts par rapport à ce modèle.

2.11 Conclusion

Ce chapitre a été essentiellement consacré à la définition des notions de base de l'image de profondeur, nous avons fait un inventaire des différentes propriétés de surface pouvant servir de critère d'homogénéité à toute éventuelle segmentation. A côté de cela, Nous avons procéder à la présentation des différents mécanismes de capture d'images de profondeur, nous avons constaté a quel point ces scanners ont évolué au fil des années, cela démontre tout l'intérêt qui est porter à ce types d'images « les images de profondeur ». Nous allons dans le chapitre suivant procéder à la présentation et l'analyse des différentes approches de segmentation d'images de profondeur les plus référenciés dans la littérature.

Chapitre 3

Segmentation d'images de profondeurs

3.1 Introduction

La segmentation est une importante étape pour la compréhension et la perception d'images de profondeur, le problème principal de la segmentation d'image de profondeur réside dans le fait de découper l'image en surfaces disjointes permettant la représentation individuelle de chaque objet figurant dans l'image, du fait de la structure en grille, la segmentation d'images de profondeur se rapproche des techniques désormais classiques de segmentation d'images 2D, les auteurs distinguent en général deux classes de méthodes, les approches par contours (*edge-based*) et les approches par régions (*region-based*) et éventuellement les méthodes hybrides utilisant les deux approches précédentes. Dans ce chapitre nous allons présenter les différentes approches de segmentation d'images de profondeur ainsi que plusieurs détecteurs de discontinuités.

3.2 Segmentation d'images de profondeur

L'étude des méthodes de segmentation est réalisée en deux temps. D'abord la notion de méthode de segmentation est prise dans un sens strict: procédé qui à partir d'une image au sens général et d'un critère d'homogénéité définit des segments. Les méthodes correspondant à cette définition sont appelées élémentaires. Les méthodes élémentaires utilisées pour la segmentation des images de profondeur (IP) font l'objet d'une présentation brève et générale.

Dans une deuxième partie, le concept de méthode de segmentation d'IP est pris dans un sens plus large. Une revue de la littérature révèle l'existence de prétraitements et de post-traitements associés aux méthodes de segmentation. Parfois les méthodes élémentaires sont combinées.

3.2.1 Méthodes élémentaires de segmentation d'image de profondeur

Les méthodes élémentaires sont groupées en deux catégories, les méthodes directes et les méthodes basées sur la mise à jour de régions. Dans les méthodes de la première catégorie, les régions sont formées en fin de procédé par étiquetage des composantes connexes, dans les

méthodes de la deuxième catégorie, les régions sont formées en début de procédé et sont mises à jour petit à petit jusqu'à ce qu'un critère de fin soit satisfait.

3.2.1.1 Détection de frontière et étiquetage des composantes connexes

Cette méthode se base sur une mesure locale de la dissimilitude entre éléments de l'image, si cette dissimilitude est plus élevée qu'un certain seuil, on décide qu'il y a, à cet endroit, un point de frontière, le résultat du test est inscrit dans une image binaire (l'image des points de frontière), dans le cas d'une image de profondeur les points de frontière représentent les points de discontinuité de type $\overline{C^0}$ et $\overline{C^1}$. Pour obtenir les segments de $V(i,j)$. On procède à un étiquetage des composantes connexes de l'image des régions $B(i,j)$ et chaque ensemble d'éléments (i,j) de même étiquette forme un segment. Entre les deux parties de la méthode que sont la détection de points de frontière et l'étiquetage des composantes connexes, on peut intercaler différents traitements. Leur objet est d'obtenir à partir des points de frontière des courbes, respectivement des courbes fermées. Dans le cas d'image d'intensité ces traitements sont groupés sous le thème de la détection de contours.

3.2.1.2 Seuillage et étiquetage des composantes connexes

Dans cette méthode, un ou plusieurs seuillages sont effectués sur une ou plusieurs des composantes de l'image, en découpant l'espace en domaines disjoints D_i , où chaque élément $V(i,j)$ appartient donc à un et un seul domaine D_i , on lui attribue l'étiquette correspondante i . Les segments sont obtenus par étiquetage des composantes connexes de l'image formée par les étiquettes i . Cette méthode est utilisée couramment en segmentation de profondeur pour faire une segmentation selon le type de courbure.

3.2.1.3 Clustering et étiquetage des composantes connexes

Le procédé cherche à regrouper des éléments $V(i,j)$ ayant même valeur, idéalement, c'est-à-dire si la valeur est strictement la même, ces éléments sont représentés par un point unique dans l'espace R (dans un cas pratique il faut laisser une marge de tolérance), la méthode consiste donc à rechercher des points de V formant un groupement rapproché ("cluster"), on attribue à chaque élément $V(i,j)$ l'étiquette m du cluster correspondant et les segments sont obtenus par étiquetage des composantes connexes de l'image formée par les étiquettes M . Cette méthode est essentiellement utilisée pour obtenir des segments correspondant à des

morceaux de surface (approximativement) plane. Dans ce cas, elle est appliquée à l'image des vecteurs normaux ou, éventuellement, à une image de paramètres de plans d'approximation locale.

3.2.1.4 Fusion de régions

A l'initialisation, la méthode définit une partition d'image $P(0)$ constituée de petites régions homogènes, la méthode fonctionne de manière itérative, à chaque itération n , deux régions voisines R_1 et R_2 dont la réunion forme une région homogène sont fusionnées et la partition est remise à jour, le processus s'arrête lorsqu'il n'y a plus aucune paire de régions voisines pouvant être fusionnées. Pour cette méthode il faut évaluer quel serait le résultat de la fusion (réunion des deux ensembles) pour savoir si elle doit être exécutée. Ceci représente une dépense lourde en calcul qu'il faut faire aussi lorsque la fusion n'est pas effectuée. La méthode de fusion n'a d'intérêt pratique que s'il existe un moyen de déterminer la mesure d'erreur accumulée. L'ordre dans lequel ont lieu les fusions joue un rôle important sur la qualité des résultats. La solution qui paraît la meilleure consiste à fusionner à chaque fois les deux régions dont les caractéristiques sont les plus semblables parmi toutes les paires de régions voisines dans l'image. Cependant cette manière de faire est gourmande en calcul. En pratique, on utilise donc souvent une solution de compromis entre résultats de qualité maximale et temps de calcul minimal. Pour l'application de cette méthode sur des images de profondeur il existe deux formes d'application de cette technique :

- **Segmentation de l'image de profondeur** : Le vecteur de description associé à la région est constitué par les paramètres d'une surface algébrique approximant l'ensemble des rangels, la mesure sur laquelle se base le critère d'homogénéité est l'erreur globale avec laquelle la surface algébrique approche les rangels faisant partie de la région.
- **Segmentation d'une image dérivée de l'IP** : Ce peut être l'image des vecteurs normaux, l'image des courbures principales ou encore une image constituée par les paramètres d'une surface algébrique d'approximation locale, le vecteur de description associé à la région est alors la valeur moyenne des éléments de la région et la mesure d'homogénéité en est l'écart-type.

3.2.1.5 Croissance indépendante de régions

Plutôt qu'une méthode, il s'agit d'une catégorie regroupant plusieurs méthodes de segmentation qui se distinguent de celles présentées précédemment par le fait qu'elles procèdent par croissance indépendante des régions, cette manière de faire conduisant généralement à un recouvrement non souhaité entre régions. Pour appliquer cette technique sur une image de profondeur, le vecteur de description \mathbf{g} doit être associé à la région et doit être constitué de paramètres d'une surface algébrique approximant l'ensemble des rangels (Les différentes mesures sont à interpréter en termes d'erreur d'approximation). La méthode comporte en fait deux phases:

- La croissance d'une région par ajout d'éléments compatibles : Ce processus itératif fait croître une région à partir d'un noyau où les éléments $V(i,j)$ candidats à faire croître la région R au pas d'itération n doivent avoir des valeurs compatibles avec celles de la région décrites par le vecteur de description \mathbf{g} , ensuite, la région est testée quant à son homogénéité (compatibilité globale des éléments avec le vecteur de description), si la région est homogène, la croissance continue, sinon, elle s'arrête à moins que le procédé soit appliqué avec différents degrés (ordres) de description. Si c'est le cas et qu'il existe une description d'ordre plus élevé que celle qui est présentement utilisée, l'ordre est augmenté (en augmentant l'ordre, on augmente la taille et avec elle les capacités de description du vecteur \mathbf{g}). Le nouveau vecteur \mathbf{g} est alors calculé et un nouveau test d'homogénéité est effectué.
- Élimination des recouvrements entre régions : ce processus consiste à effectuer un choix de région, lorsque deux ou plusieurs régions se recouvrent fortement, l'élimination du recouvrement passe par une élimination de région(s), les régions dont le vecteur de description \mathbf{g} représente le moins bien les données sont éliminées (une mesure de qualité de description est définie).

3.2.1.6 Division et fusion de régions

Cette méthode commence par diviser l'image, tout d'abord l'image de profondeur est partagée en plusieurs (en général 4) régions de même taille et de même forme (carrée), puis, chacune des régions, pour laquelle le critère d'homogénéité n'est pas rempli, est à son tour divisée en

plusieurs sous-régions et ainsi de suite. A cause du découpage régulier, le processus de division sépare généralement une région homogène en plusieurs morceaux c'est pour cela que la technique de fusion décrite ci-dessus est appliquée, une fois la division terminée, pour regrouper les morceaux.

3.2.2 Méthodes de segmentation d'images de profondeur

D'après les documents trouvés dans la littérature, on peut classer les méthodes de segmentation en quatre classes suivant le critère d'homogénéité auquel obéit la segmentation on peut ainsi trouver la segmentation selon le type de courbure, surface plane, surface algébrique, continuité C^1 et une cinquième classe regroupant les autres méthodes.

3.2.2.1 Segmentation selon le type de courbure

L'estimation des courbures pose problème car elle est perturbée à la fois par la présence du bruit (erreurs de mesure) et par la présence des discontinuités. Pour que l'estimation des courbures soit suffisamment bonne, il faut que le bruit soit faible, d'autre part, il faut éliminer les perturbations dues aux discontinuités par un traitement adéquat. Pour ce type de segmentation la méthode seuillage et étiquetage des composantes connexes est privilégiée.

- Kasvand [Kasvand, 1988] procède à une érosion de 1 à 2 pixels sur l'image de type de courbure pour supprimer l'effet des discontinuités, ensuite, à lieu l'étiquetage des composantes connexes de l'image de type de courbure et Finalement, une croissance des régions est effectuée sous contrôle de l'approximation de la région par une surface algébrique. La figure 3.1 représente une segmentation en fonction des seuils sur les courbures gaussienne. Sont représentés uniquement les deux types de courbure correspondant aux surfaces mesurées (plan, cylindre). Pour chaque degré de lissage, il existe une valeur minimale pour les seuils en dessous de laquelle on est sensible au bruit. Cette valeur dépend évidemment de la qualité de l'image originale. Pour pouvoir faire la distinction entre une surface légèrement courbe et une surface plane, il faut que le lissage soit suffisant.

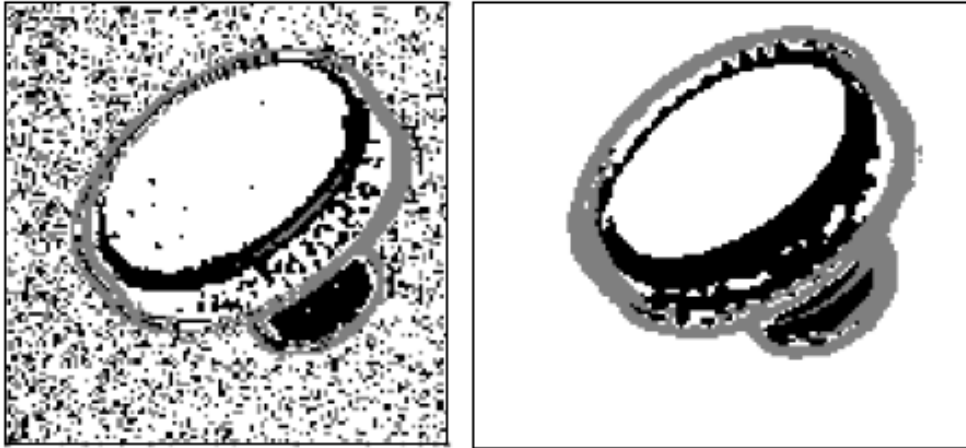


Figure 3.1 – Segmentation par seuillage de Kasvand, à gauche l'image de profondeur, à droite le résultat de segmentation

- Yokoya et Levine [Yokoya& al, 1989] combinent la détection des discontinuités « *Figure 3.2* » avec la segmentation en type de courbure, l'image des discontinuités est utilisée comme masque lors de l'étiquetage des composants connexes de l'image de type de courbure, (les discontinuités restreignent la connexité). Ensuite, une croissance de région par simple dilatation aboutit à un partage d'image qui est une partition.

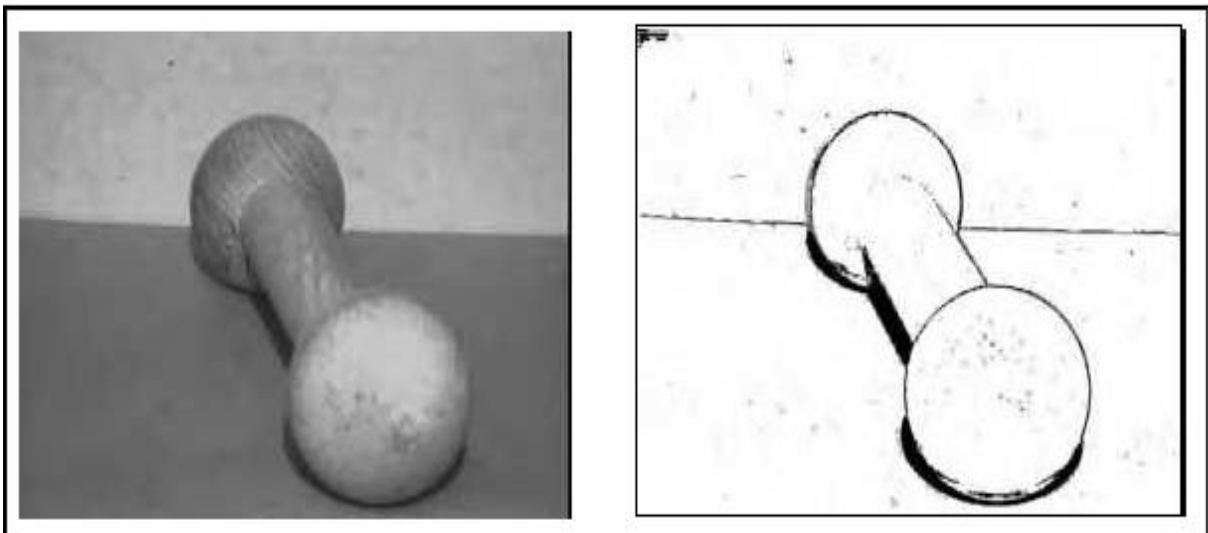


Figure 3.2 – Détecteur de discontinuités de Yokoya et Levine. A gauche l'image de profondeur. A droite l'image segmentée.

3.2.2.2 Segmentation en surface plane

Deux critères d'homogénéité définissent une segmentation en surfaces planes "même orientation" et "même équation de plan", les méthodes basées sur le premier critère doivent

prendre en compte les discontinuités de profondeur pour éviter une fusion entre deux plans parallèles mais non colinéaires (par exemple une face supérieure d'un cube et une surface sur laquelle il est posé). La prise en compte des discontinuités d'ordre 0 est souvent faite de manière implicite, on estime l'orientation au travers des discontinuités, on crée une zone de forte inclinaison faisant office de séparation, l'inconvénient majeur est l'apparition de régions (segments) parasites.

- Parvin et Medioni [*Pavini & al, 1986*] utilisent une méthode de division-fusion basée sur la normale à la surface ou, plus précisément, sur le gradient de l'image de profondeur. Le gradient est estimé par la méthode conventionnelle utilisant un voisinage 3x3, la méthode de division-fusion est ensuite appliquée à l'image de gradients « *Figure 3.3* ». La mesure δ définit le critère d'homogénéité pour la division (l'écart-type du gradient calculé). L'homogénéité pour la fusion est définie sur la base de la valeur de l'angle entre vecteurs normaux associés aux régions, étant donné le procédé d'estimation du gradient utilisé, cette méthode est soumise au même phénomène de régions parasites que celle de Maître et Hügli [*Maître & al, 1989*].

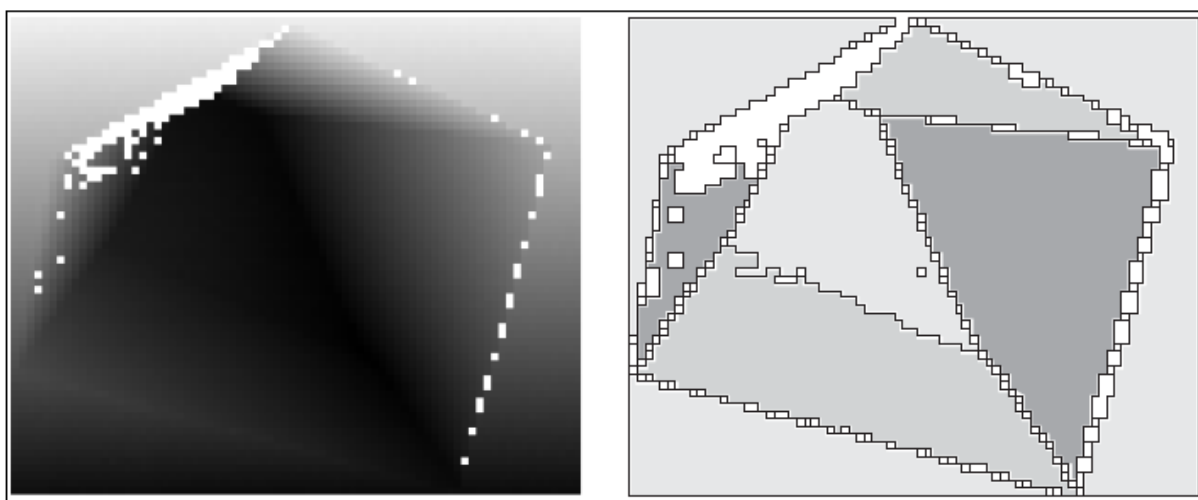


Figure 3.3 – Résultat de segmentation, à gauche l'image de profondeur, à droite l'image segmenté.

- Taylor [Taylor & al, 1989] applique la méthode de division-fusion à une image de normales (gradient), l'estimation du gradient se fait sur un voisinage adapté, ainsi, le problème des régions parasites est éliminé « *Figure3.4* », lors de la division et de la fusion, un test de profondeur doit par contre être effectué lorsque les tests

d'homogénéité sur les normales sont positifs, ceci permet d'éviter que des plans parallèles séparés par une discontinuité de profondeur forment un seul segment.

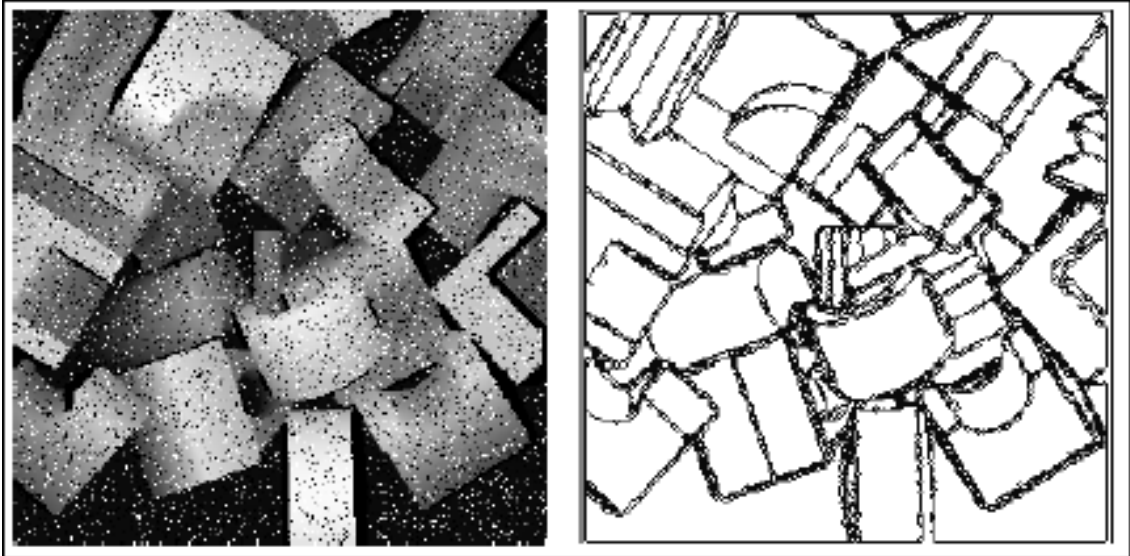


Figure 3.4 – Résultat de segmentation par division-fusion de Taylor, à gauche l'image de profondeur, à droite l'image segmenté.

- Bhavsar [*Bhavsar & al, 2010*] propose une méthode de fusion basée sur la normale (gradient) à la surface, les normales sont calculées sur des facettes de taille minimale (carré de quatre rangels voisins), ces facettes servent de partition initiale, puis, la fusion est effectuée de manière séquentielle en fusionnant à chaque fois, parmi toute les paires de régions voisines « *Figure 3.5* », celles dont l'angle entre les normales est le plus petit.

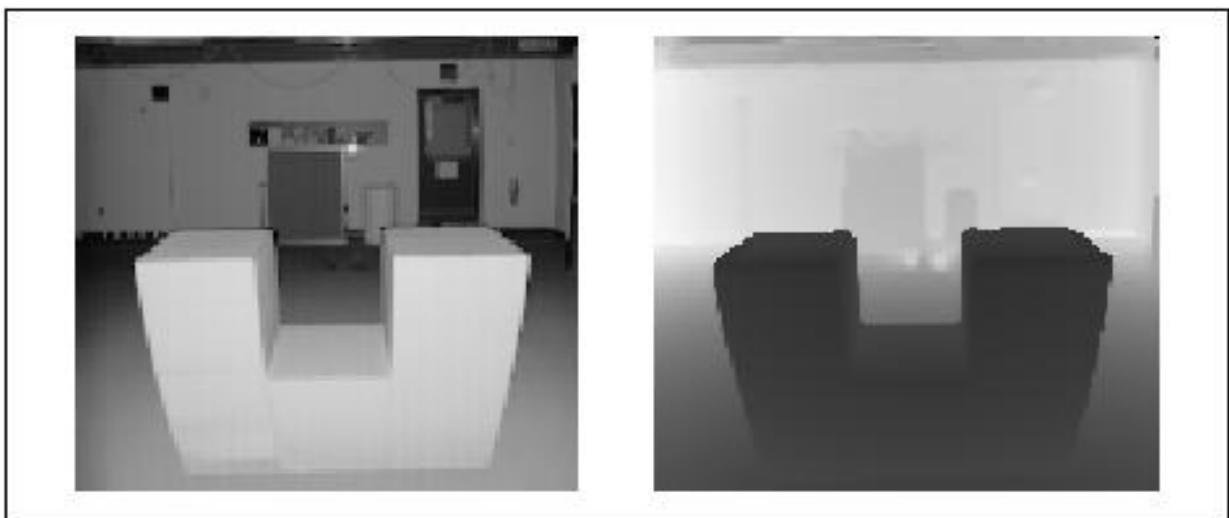


Figure 3.5 – Résultat de segmentation. A gauche l'image de profondeur. A droite l'image segmentée.

3.2.2.3 Segmentation en surfaces algébriques

La segmentation en surfaces algébriques (non strictement planes) est divisée en deux catégories suivant la nature des surfaces $2.5D$ et $3D$, les surfaces $2.5D$ correspondent toutes à des fonctions polynomiales de deux variables et s'appliquent évidemment uniquement aux images de type scalaire, quand aux surfaces $3D$ elles sont des quadriques ou des super quadriques et leur traitement est d'une complexité supérieure à celui des surfaces $2.5D$.

- Leonardi [Leonardi, 1987] présente une méthode de division-fusion liée à un modèle de surface $2.5D$ de type polynomial. Le degré du polynôme peut varier de 0 à 3, mais il est fixe pour une segmentation donnée. La méthode est appliquée à des images d'intensité. Bien que certaines modifications pourraient éventuellement en améliorer les performances, elle peut néanmoins être utilisée telle quelle pour des images de profondeur de type scalaire. Le critère d'homogénéité pour la division et pour la fusion est basé sur l'erreur d'approximation des valeurs d'image par le polynôme de meilleure approximation. Lors de la fusion, erreur d'approximation et coefficients du polynôme sont mis à jour à partir de ceux des régions fusionnées. Il n'est pas nécessaire de procéder à un calcul d'approximation sur la nouvelle région. La figure 3.6 illustre un résultat de segmentation par division-fusion de Leonardi sur image de profondeur. Il est à noter que Leonardi a jugé utile, voire nécessaire de diviser l'image en premier lieu suivant les contours (discontinuités de profondeur d'une image d'intensité).

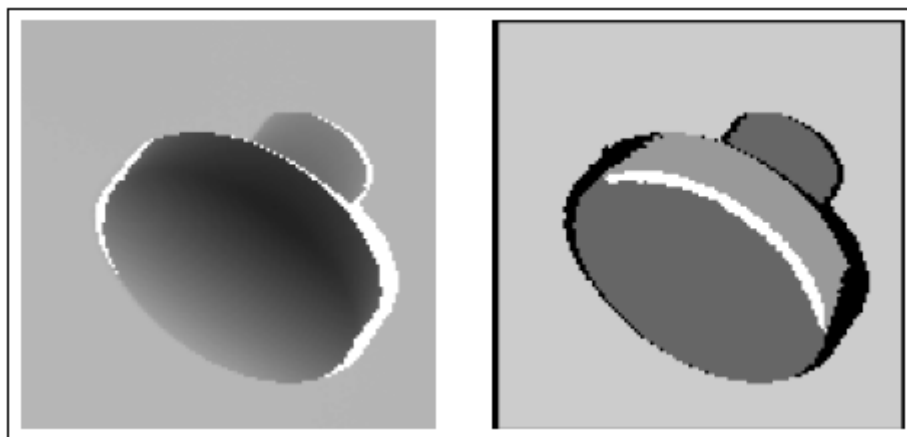


Figure 3.6 – Résultat de segmentation, à gauche l'image de profondeur, à droite l'image segmenté.

- Gupta et Bajcsy [*Gupta & al, 1992, Gupta & al, 1993*] présentent une méthode de segmentation dont le résultat est la description de l'image de profondeur par des supers quadriques (plus précisément des super ellipsoïdes). Pour aboutir au résultat, l'approche suivie par Gupta effectue un raisonnement géométrique sur la base de deux autres segmentations, une segmentation préliminaire en surfaces 2.5D de type polynomial et une segmentation courante sur les résidus de l'approximation par le super ellipsoïde « *Figure 3.7* », il s'agit donc d'un procédé qui dépasse le cadre des traitements bas-niveau d'une image de profondeur.

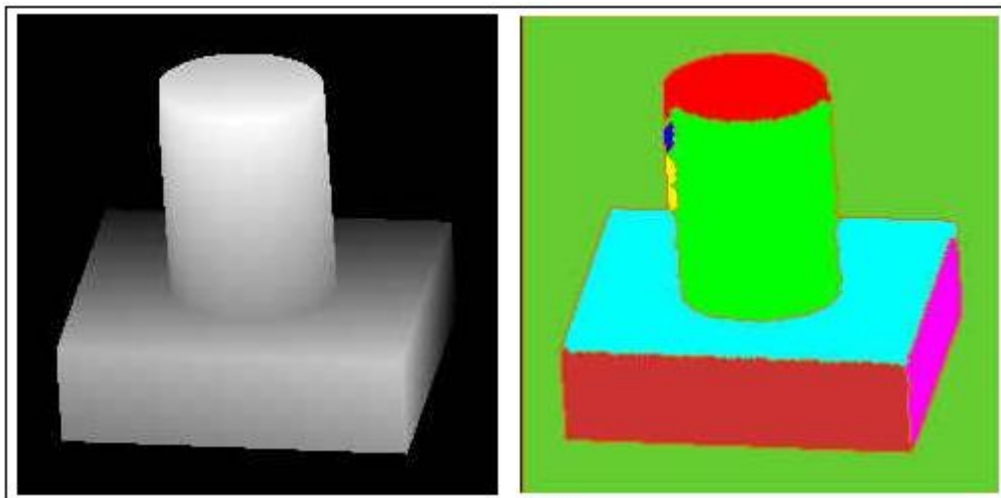


Figure 3.7 – Segmentation par supers quadriques de Gupta, à gauche l'image de profondeur, à droite l'image segmentée.

- Jiang et Bunke [*Jiang & al, 1994*] procèdent à une croissance de régions sous contrainte d'approximation par un plan (2.5D), la méthode de croissance est particulière car elle est basée sur des segments de droite, les régions sont des listes de segments et la croissance est effectuée segment par segment et les segments de droite sont obtenus par division d'un profil (ligne ou colonne) de l'image de profondeur. La croissance des régions se fait de manière séquentielle, c'est-à-dire une région après l'autre. La graine avec laquelle est initialisée la croissance est constituée de trois segments de droite voisins issus de trois profils consécutifs, la croissance commence avec la graine qui satisfait au mieux un test de parallélisme entre segments de droite. La figure 3.8 illustre la segmentation effectuée par Jiang et Bunke.

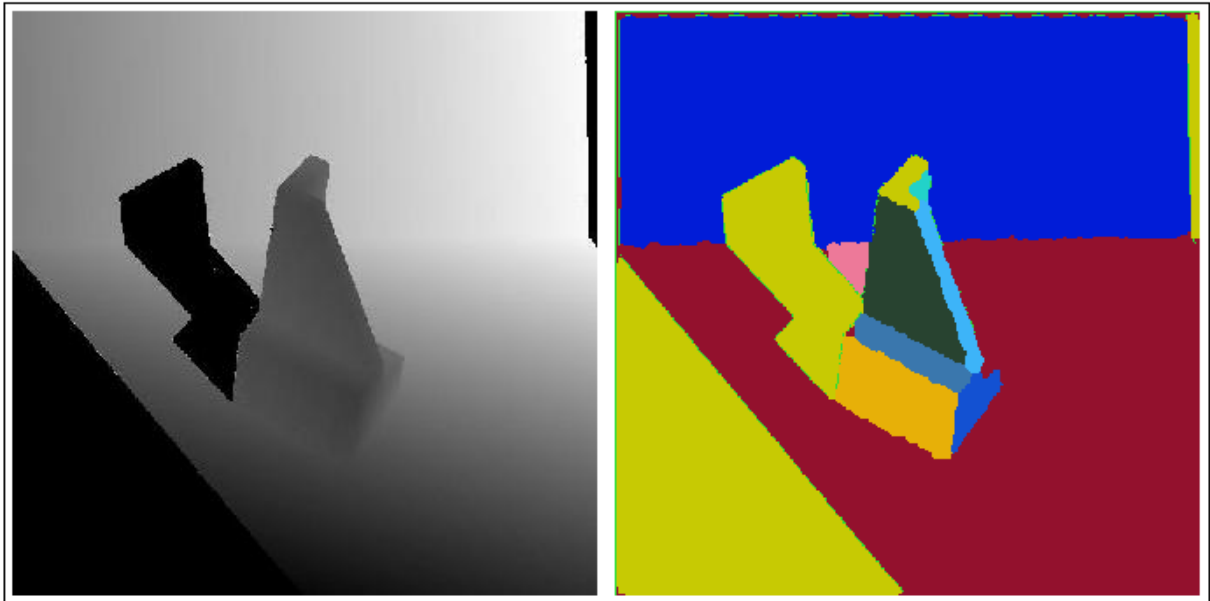


Figure 3.8 – Segmentation par croissance de régions de Jiang, à gauche l'image de profondeur, à droite l'image segmentée.

3.2.2.4 Segmentation de continuité d'ordre 1

Plusieurs méthodes conduisent à une segmentation dont le critère d'homogénéité de surface peut être défini comme la continuité $C1$, deux principes sont appliqués :

- Fusion de segments issus d'une segmentation plus contrainte : La méthode de Besl [Besl & al, 1988] commence par une segmentation selon les types de courbure, les régions qui en sont issues sont ensuite érodées, à partir des graines ainsi obtenues, la méthode procède par croissance, comme décrit précédemment, finalement a lieu une fusion des régions sur la base de l'existence ou non de discontinuité entre les régions. L'inconvénient majeur est qu'il suffit qu'une partie seulement de la frontière entre segments corresponde à une discontinuité pour que la fusion n'ait pas lieu, ce procédé a donc un effet de contrainte de fermeture systématique des frontières associées aux discontinuités « *Figure 3.9* ».

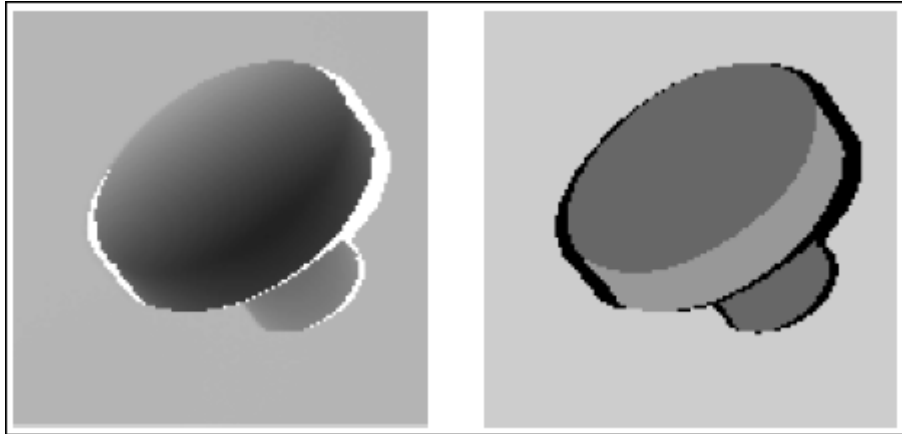


Figure 3.9 – Résultat de segmentation, à gauche l'image de profondeur, à droite l'image segmenté.

- Détection des discontinuités et étiquetage des composantes Connexes : Il s'agit du principe de détection de frontière décrit précédemment, les points de frontière sont les points de discontinuité $\overline{C^0}$ et $\overline{C^1}$. Dans l'approche de Davignon [Davignon, 1992], la croissance du domaine des points de frontière détectés est effectuée de sorte à former toujours des frontières fermées, la croissance se fait en choisissant à chaque pas le pixel voisin dont la mesure de discontinuité est la plus proche (en dessous) d'un seuil, une fois les frontières fermées, les régions connexes sont approchées par une surface algébrique (de type polynomial) de complexité croissante, jusqu'à ce que l'erreur d'approximation soit suffisamment petite, la position des discontinuités est ensuite réajustée à partir de cette représentation de surface « *Figure 3.10* ».

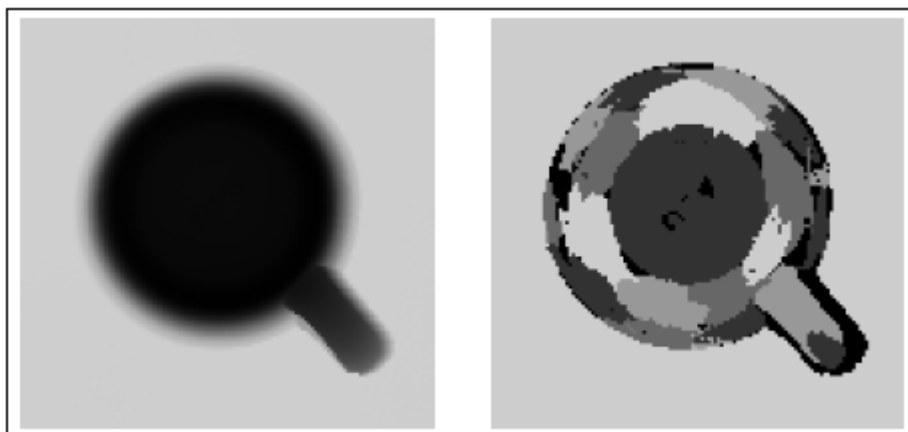


Figure 3.10 – Résultat de segmentation, à gauche l'image de profondeur, à droite l'image segmenté.

Pour certaines des méthodes que nous venons de décrire, le critère d'homogénéité comporte une condition supplémentaire à la pure continuité C1. Cette condition provient de la fermeture systématique des courbes de discontinuité, alors que celles-ci ne sont pas forcément fermées. La fermeture systématique des discontinuités existe soit dans les méthodes procédant par détection des discontinuités, soit dans celles qui procèdent par fusion de segments obéissant à un critère d'homogénéité plus contraignant. Dans certaines méthodes de détection des discontinuités [Pulli & al, 1993], on peut admettre que l'accroissement du domaine des points de discontinuité est destiné à remédier à une détection incomplète des discontinuités. Par contre, dans certaines méthodes, la fermeture des discontinuités est systématique et rajoute ainsi une contrainte à la continuité. Dans d'autres méthodes de fusion de segments [Attila & al, 2006], il suffit qu'une partie seulement de la frontière entre segments corresponde à une discontinuité pour que la fusion n'ait pas lieu. Ce procédé a donc un effet de contrainte de fermeture systématique des frontières associées aux discontinuités. Par conséquent, le critère d'homogénéité de la segmentation finale n'est pas uniquement la continuité C1.

3.2.2.5 Autres méthodes de segmentation

Les méthodes décrites ci-après appliquent le même principe de fusion de segments plus contraints que la méthode de *Besl* [Besl & al, 1988] néanmoins, la fusion repose sur des conditions plus restrictives que l'absence de discontinuités entre segments, la segmentation finale obéit donc à un critère d'homogénéité plus contraignant que la continuité C1.

- Hoffman et Jain [Hoffman & al, 1987] ont une méthode de segmentation constituée de deux étapes, Clustering et fusion. Le "Clustering" a lieu dans un espace à six dimensions (Aux composantes du vecteur normal sont ajoutées les coordonnées cartésiennes du rangel) les coordonnées x et y imposent la condition de connexité et la coordonnée z contraint à une séparation des segments en présence de discontinuités de profondeur, l'algorithme de "Clustering" procède par minimisation des distances intra-"cluster", par conséquent, il existe un paramètre définissant une borne supérieure pour le nombre de clusters, sinon le résultat serait une segmentation en régions mono-élément. Les segments sont classés en régions planes, convexes et concaves et sont finalement fusionnées « *Figure 3.11* » des régions adjacentes appartenant à la même classe pour autant qu'elles ne soient pas séparées par des discontinuités.

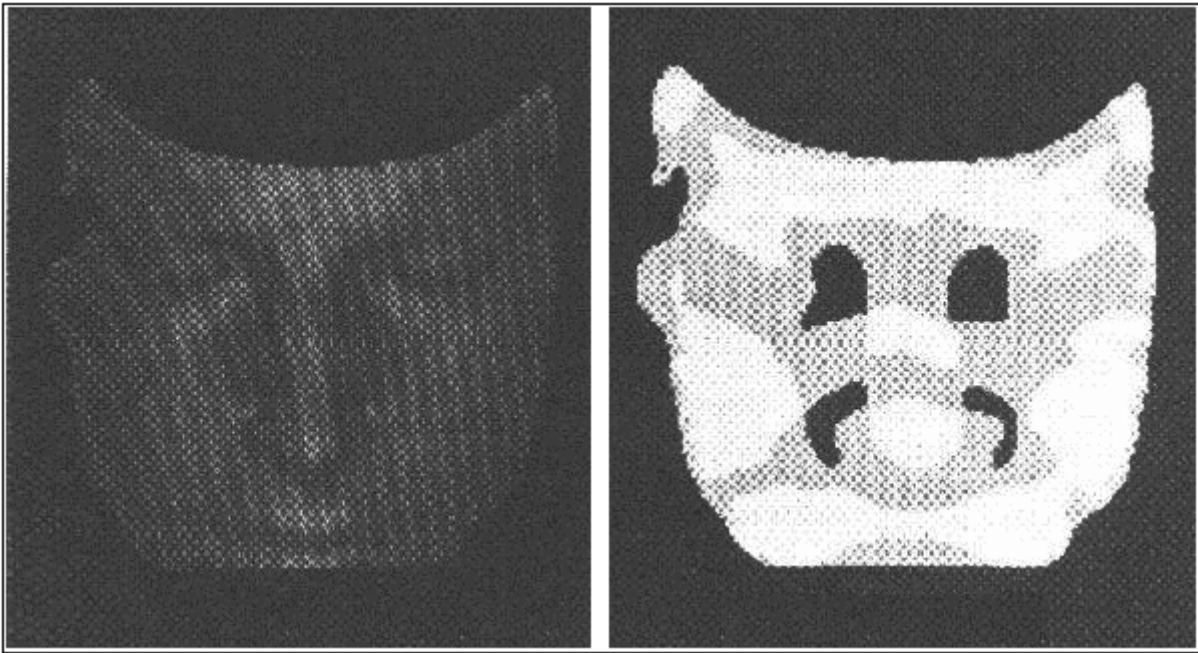


Figure 3.11 – Résultat de segmentation, à gauche l'image de profondeur, à droite l'image segmenté.

- Ade et Ylä-Jääski proposent une méthode [Ade & al, 1990] qui comporte plusieurs étapes. La première étape consiste en un "Clustering" des vecteurs normaux et à un étiquetage des composantes connexes, les régions de petites tailles (étroites) sont éliminées à la fin de cette première étape. La deuxième étape effectue une croissance des régions conservées, cette croissance comporte trois pas, un pas de croissance avec une contrainte d'approximation par un plan, un pas avec une contrainte d'approximation par une surface 2.5D polynomiale de second degré, puis, une dilatation jusqu'à ce que les régions se touchent, finalement, a lieu une étape de fusion où deux régions adjacentes sont fusionnées si leurs normales et leurs courbures principales ont des valeurs proches et si elles ont une frontière commune suffisamment longue.
- Merchán [Merchán & al, 2002] propose une méthode de segmentation automatique d'une image de profondeur comportant des scènes complexes (comprend plusieurs objets, des formes irrégulières). Contrairement à la plupart des méthodes existantes qui se déroulent avec un ensemble de images obtenues à partir des points de vue différents, pour cette méthode une vue unique est utilisé « Figure 3.12(a) » et un processus de segmentation 3D est développé pour séparer les éléments constitutifs d'une scène complexe. Tout d'abord une projection orthogonale sur 2D (dans une

multi pixel matrix) est effectuer « *Figure 3.12(b)* » avant d'entamer de segmentation, « *Figure 3.12(c)* » puis, on effectue une transformation inverse (*projection en 3D*) en réassignant chaque segment au segment qui lui correspond dans la scène, Les parties segmentées sont retirés de la scène « *Figure 3.12(d)* » pour l'itération suivante et une nouvelle scène est à nouveau explorée, le processus et répéter « *Figure 3.12(e)* » jusqu'à ce qu'il n'y est plus aucun objet sur la scène. Parfois le segment extrait correspond à plus d'un objet de la scène, cette situation est envisagée par la méthode et l'algorithme détecte automatiquement que ces circonstances ont eu lieu et effectue une nouvelle segmentation au cours de ces segments à partir de la scène d'exploration. L'algorithme a été testé sur un ensemble d'images de données réelles, les scènes sont composées de plusieurs objets qui peuvent avoir des formes irrégulières qui ont été peints avec la même couleur et l'algorithme a apporté de bon résultat et a prouvé qu'il pouvait être utilisé avec succès pour réaliser la segmentation d'une scène 3D.

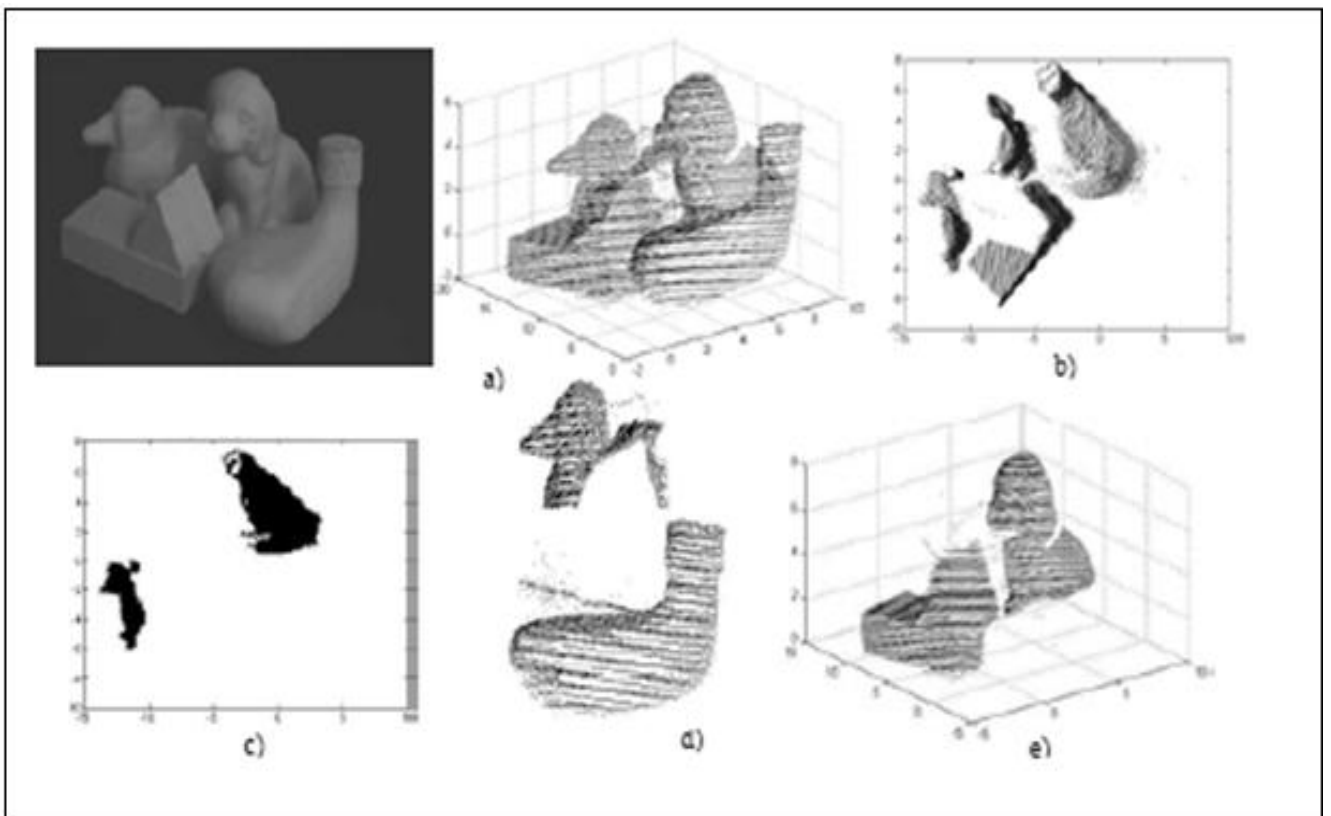


Figure 3.12 – Processus de segmentation automatique de Merchan

- Paulo Fabiano dans [Fabiano & al, 2004] propose un algorithme de segmentation d'images de profondeur combinant des techniques basés région et des techniques basés contour et applicable aux surfaces planes et courbées en améliorant un estimateur robuste de détection et d'extraction de surface appelé RANSAC (Random Sample Consensus). Comme prétraitement de l'image, un filtre médian 3*3 et une analyse de composants principales est appliqué pour estimer les vecteurs de surfaces, ensuite, pour chaque pixel, la variation angulaire dans le sens normal est calculée en passant du pixel P à un pixel de proximité Q, après l'utilisation d'un seuillage bas et haut, chaque point est alors marqué comme appartenant a une surface plane, courbé, ou indéterminé (sinon), ce prétraitement permet donc la classification des pixels à utiliser pour identifier les régions planes et courbés. Le RANSAC a été spécifiquement conçu pour améliorer la détection de surface et d'extraction de données télémétriques présentant de multiples structures (surfaces) et d'erreurs de détection (dégradation), dans cette approche l'estimateur utilise un nouvel algorithme génétique afin d'accélérer le processus d'optimisation de montage de surface solide, tout en évitant une convergence prématurée. La performance de détections indiquent la nécessité d'une amélioration dans la détection « Figure 3.13 » des limites de surface, aussi, le test de planéité de cette méthode doit être amélioré afin de mieux détecter les surfaces à faible courbure, cet algorithme peut être étendu pour traiter des nuages de points résultant d'enregistrement d'image de profondeur RIR (Range Image Registration).

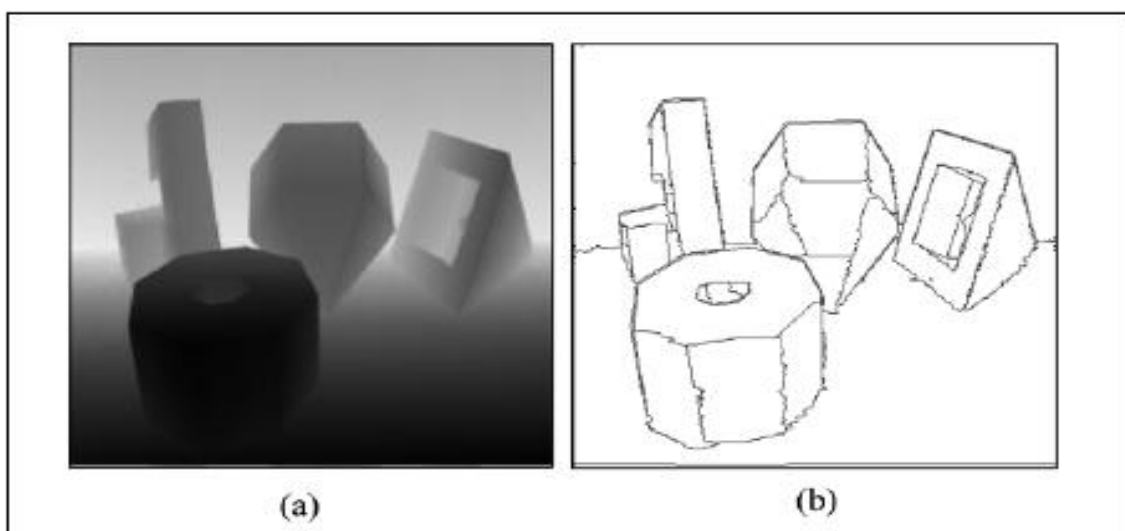


Figure 3.13 – Résultat de segmentation. (a) L'image de profondeur. (b) L'image segmenté.

- Wand et Suter ont développé le MDPE (*Maximum Density Power Estimator*) qui est un estimateur non paramétrique de la densité et incluant les techniques d'estimation de gradient [Wang & al, 2004] qui est similaire à de nombreux estimateurs à échantillonnage aléatoire. L'algorithme utilisé consiste à choisir une fenêtre de recherche et un p sous-ensemble au hasard, puis, à l'application de l'algorithme Mean Shift sur tous les points de la fenêtre, ensuite, la densité est calculée sur tous les points de données de la fenêtre choisie pour établir la densité de puissance. L'essentiel de la méthode réside en l'application de plusieurs procédures pour trouver la densité locale maximale de ces résidus (les points de données ayant moins de résidus devraient être aussi nombreux que possible, et les résidus doit être aussi faible que possible). Le MDPE peut tolérer plus de 85% des valeurs aberrantes (différente de la normale) et a régulièrement été observés pour bien fonctionner avec encore plus de 90% des valeurs aberrantes, cependant, lorsque deux surfaces se coupent « *Figure 3.14* », les points autour de la ligne d'intersection peut éventuellement être attribué à la première surface soit à la deuxième surface (ceci étant du au fait que la ligne d'intersection se trouve sur les deux surfaces et les points de données sont les deux surfaces), des informations complémentaires (tels que la normale à la surface à chaque pixel) doit être utilisé pour traiter les données à proximité de la ligne d'intersection afin d'améliorer les résultats. Quand toutes les surfaces sont estimées, certaines valeurs isolées (à cause du bruit introduit par la caméra de l'image de profondeur, peuvent rester) et une procédure de post-traitement devrait être élaboré pour éliminer les valeurs aberrantes isolé.

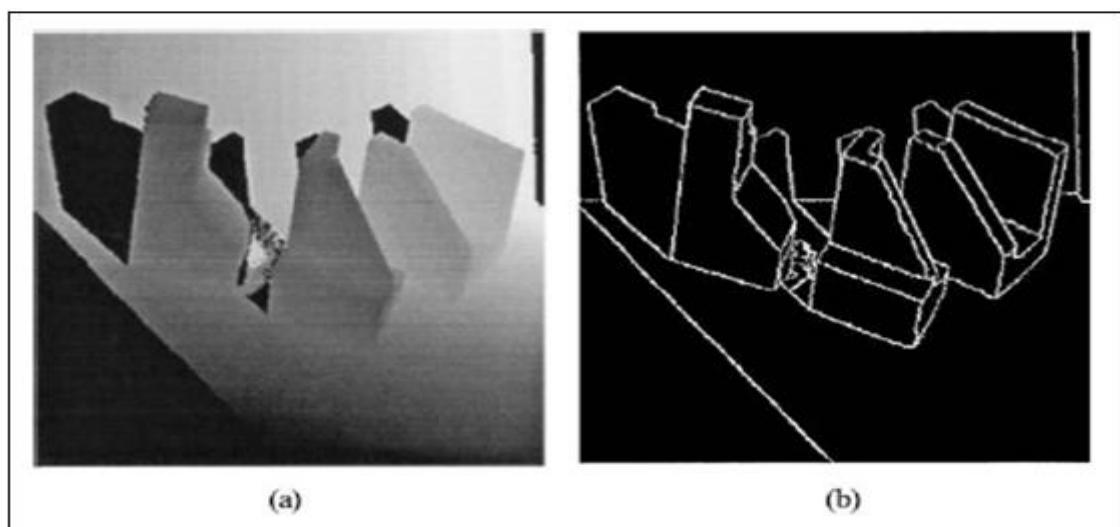


Figure 3.14 – Résultat de segmentation. (a) L'image de profondeur. (b) L'image segmenté.

- Bab-Hadiashar et Gheissari Proposent une méthode fiable [Bab-Hadiashar & al, 2004] segmenter une image de profondeur comportant des surfaces planes et courbées, ils proposent une surface de sélection (modèle) et un critère basé sur la minimisation de la somme de flexion et de torsion des énergies de toutes les surfaces possibles pour détecter et identifier un modèle représentant des surfaces d'objets courbés. L'algorithme commence par éliminer les pixels dont les voisins n'ont pas de valeur de profondeur valables en raison de la limitation du télémètre utilisé pour mesurer la profondeur (principalement en raison de mauvaise texture, ces points sont habituellement marqués par le télémètre avec un nombre hors-série) puis, chercher un groupe de données pour lesquelles tous les pixels appartiennent à une surface plane (Même s'il n'y a pas de surface plane dans l'image, ils peuvent être approximative à une très petite zone locale comme une surface plane), ensuite, comparer le modèle le plus élevé dans la bibliothèque (des modèle possible) à tous les groupes de données acceptées et trouver le résiduel pour chaque point, pour appliquer une méthode de sélection du modèle pour la région élargie (en adaptant et en comparant tous les modèles dans la bibliothèque de modèle pour la région élargie) et de trouver le modèle approprié et y évaluer l'ampleur du bruit et rejeter les points dont les carrés résiduelle sont supérieure à seuil de l'échelle du bruit, enfin, éliminer la partie segmentée de l'image et réitérer jusqu'à ce qu'il n'y es plus de partie a retiré « *Figure 3.15* ».

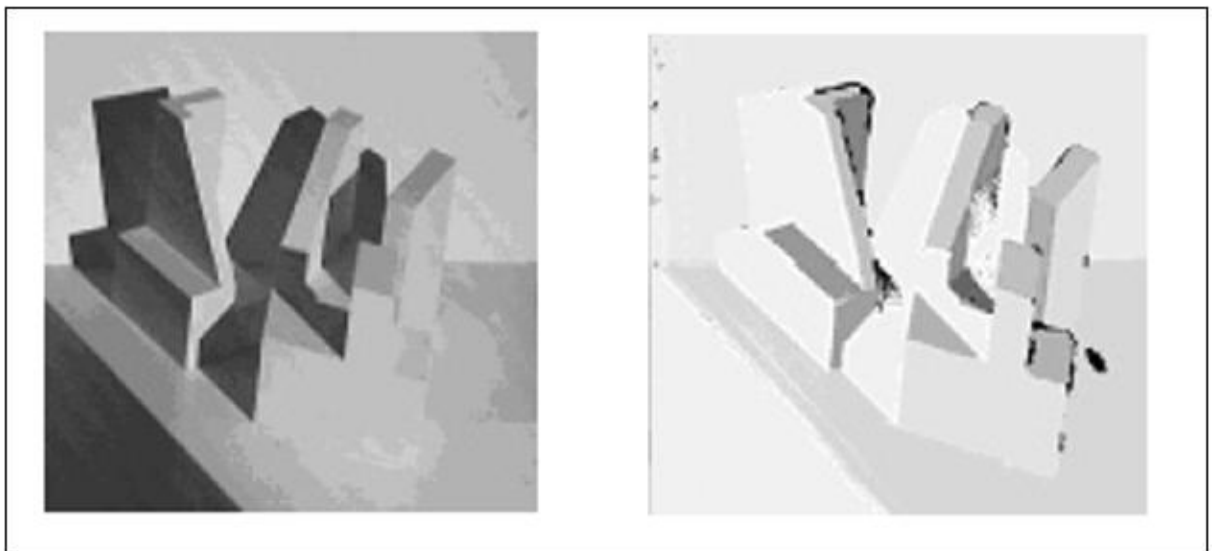


Figure 3.15 – Résultat de segmentation, à gauche l'image de profondeur, à droite l'image segmenté.

L'algorithme proposé a été en mesure d'identifier correctement le modèle sous-jacent pour chaque surface à l'aide de la bibliothèque de modèles et du critère de sélection de modèles, cette méthode permet selon ce critère de développer une gamme de modèles robustes à base d'algorithmes de segmentation capable de distinguer entre différents types de surfaces tout en segmentant les objets .

- Han dans [Han & al, 2004] présente une méthode de segmentation efficace utilisant le Jump-Diffusion sur des images de profondeur dans une formulation bayésienne, l'algorithme fonctionne sur des scènes complexes du monde réel qui consistent en un nombre inconnu d'objets (ou surfaces) de différentes tailles et types, tels que les avions, les coniques, les surfaces lisses, et les objets encombrés (comme des buissons). Formulé dans le cadre bayésien, la probabilité à posteriori est répartie sur un espace de solution avec un nombre dénombrable de sous-espaces de dimensions variables (probabilité d'avoir une surface plane conique ou encombrée).

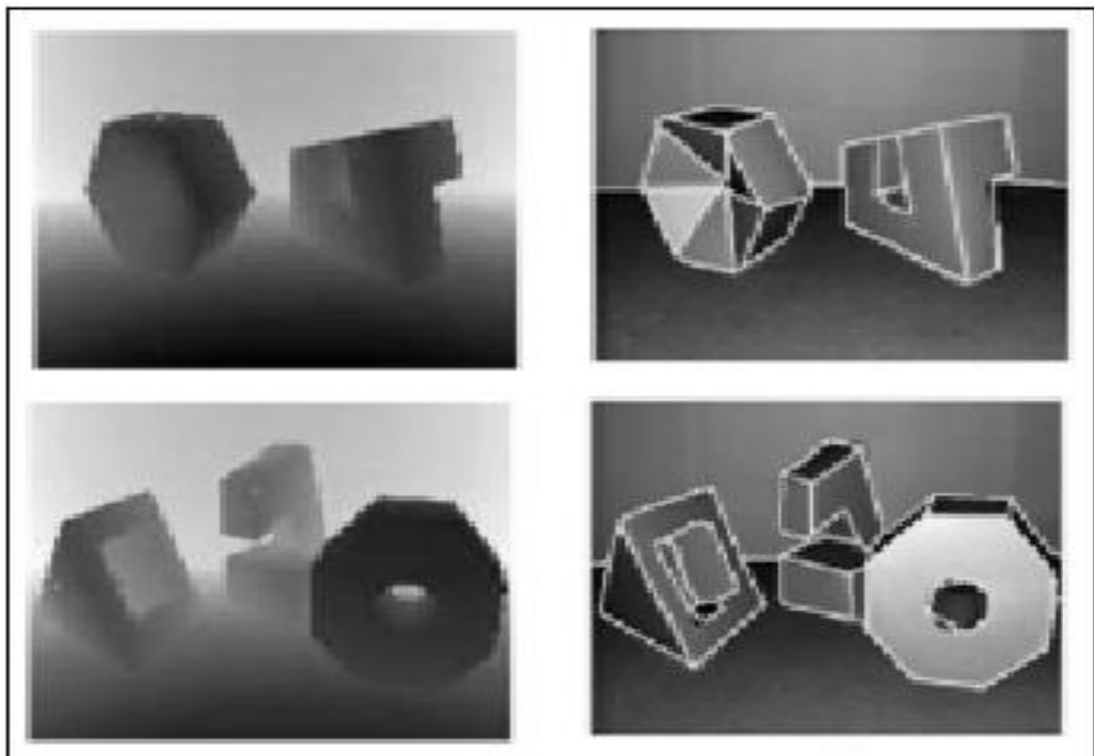


Figure 3.12– Résultat de segmentation obtenu, à gauche une image de profondeur, à droite le résultat de la segmentation

L'algorithme simule les chaînes de Markov à deux sauts réversibles (Les sauts réversibles réalisées entre les sous-espaces de dimensions différentes« *Figure 3.16*»,

telles que le changement des modèles de surface) et effectue une diffusion aléatoire à parcourir l'espace des solutions. L'algorithme utilise un processus de Jump-Diffusion comme outil général pour la conception de chaînes de Markov traversant l'espace des solutions complexes et donc la solution dépend plus de l'initialisation, mais, l'algorithme se révèle gourmand en temps de calcul et mets environ 1 heure sur un PC PentiumIV à segmenter une scène d'environ 300*300 pixels avec des conditions initiales arbitraires ce qui constitue un énorme inconvénient.

- Germann propose une méthode[Germann & al, 2010] de segmentation d'image de profondeur entièrement automatique, rapide, robuste à l'occlusion des scènes et encombré, l'algorithme effectue un filtre médian 3*3 pour la réduction de bruit, puis, une heuristique simple est appliquée pour détecter les contours dans l'image en comparant les valeurs de pixels voisins« Figure 3.17», si la différence de distance dépasse un seuil prédéfini un pixel est marqué comme appartenant à un contour, à chaque itération de l'algorithme, la valeur de la distance de chaque pixel est comparé à la distance du plus proche pixels marqué comme appartenant à un contour, la valeur de la distance et les coordonnées de ce pixel sont mises à jour si la distance est moindre, Le nombre d'itérations correspond au maximum de distance d'un pixel à son plus proche voisin appartenant à un contour. Cet algorithme apporte de bons résultats mais de petites modifications en raison des restrictions mécaniques pourraient être utiles, par exemple, de rejeter les objets qu'un bras robot réel ne peut pas atteindre, ou de prendre en compte le fait que la vitesse du bras favorise les objets à proximité de sa position actuelle.

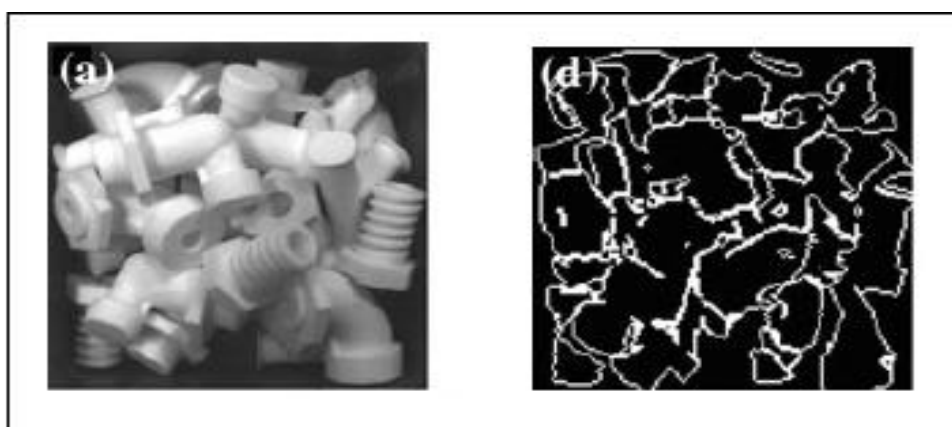


Figure 3.17 – Résultat de segmentation, à gauche l'image de profondeur, à droite l'image segmenté.

3.3 Détection des discontinuités

Les discontinuités (d'ordre 0 et 1) d'une image de profondeur sont définies par rapport à une surface que l'on associe à l'image de profondeur en tant que modèle. La détection des discontinuités consiste à décider où introduire des discontinuités dans le modèle de surface. Cette décision se base sur les données de l'image de profondeur. La détection des discontinuités d'une image de profondeur a souvent pour but d'extraire des primitives *ID* liées aux courbes que forment les points de discontinuité. Le niveau de localité/globalité auquel est effectuée l'analyse du signal pour la détection constitue un critère essentiel de distinction entre les méthodes.

3.3.1 Détecteur de point de discontinuité

Soit une image de profondeur P , un détecteur de point de discontinuité est une opération de voisinage sur P dont le résultat est la décision concernant la présence d'un point de discontinuité à la position de référence. Formellement, pour la position (i, j) , la décision $b(i, j)$ résulte de l'opération Ω sur le voisinage $W(i, j)$.

$$b(i, j) = \Omega[(k, l), p(k, l) : (k, l) \in W(i, j)] \quad (3.1)$$

Un détecteur de point de discontinuité est discret et local :

- Discret : La détection se limite à un ensemble fini de points, ce sont soit les positions (i, j) de l'IP, soit les positions intermédiaires, $(i + 1/2, j)$ et $(i, j + 1/2)$, utilisées fréquemment pour décrire les frontières des régions. (A noter que cette deuxième solution permet d'obtenir une segmentation ayant la propriété de partition.)
- Local : La décision est prise de manière indépendante pour chaque voisinage. (On peut par contre imaginer une méthode où la décision serait prise de manière coopérative.)

La *Figure 3.18* représente un exemple de discontinuité de profondeur, deux régions sont sélectionnées, un domaine carré et un profil, l'image de profondeur de taille (128x128x8bits) provient d'un système d'acquisition de type "temps de vol" développé par l'ERIM (Environmental Research Institute of Michigan), il s'agit d'un capteur relativement ancien; depuis, la qualité de mesure des capteurs a été améliorée.

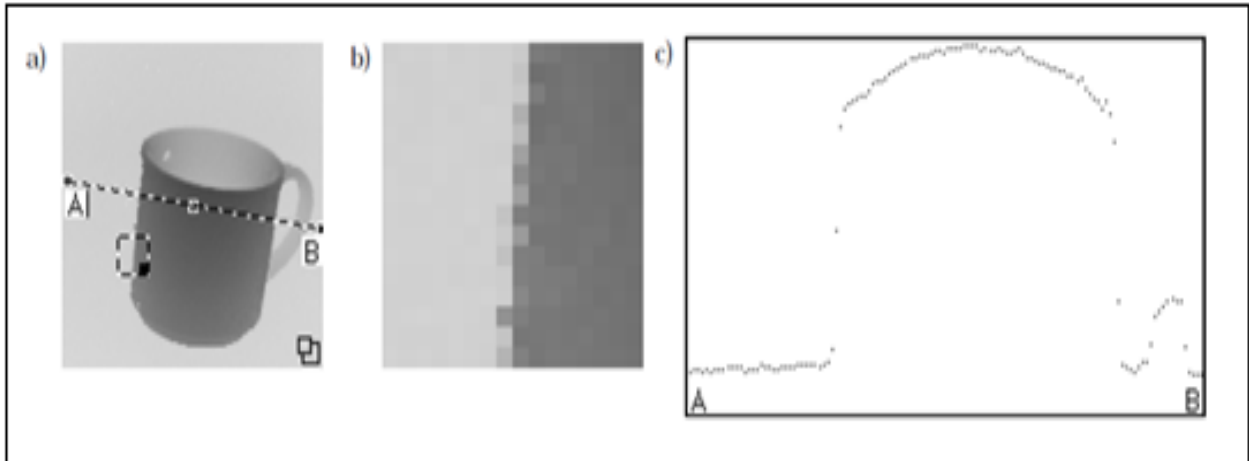


Figure 3.18 – Exemple de discontinuité de profondeur. (a) Image de profondeur. (b) Agrandissement de la région carrée. (c) Représentation orthonormée du profil.

3.3.2 Détecteurs différentiels de point de discontinuité

Il s'agit de détecteurs de point de discontinuité particuliers basés sur l'évaluation de grandeurs différentielles de l'image de profondeur. Dans ce paragraphe, le cas simple d'une discontinuité de profondeur et d'un signal discret unidimensionnel $[P_i]$ est présenté « Figure 3.19 ».

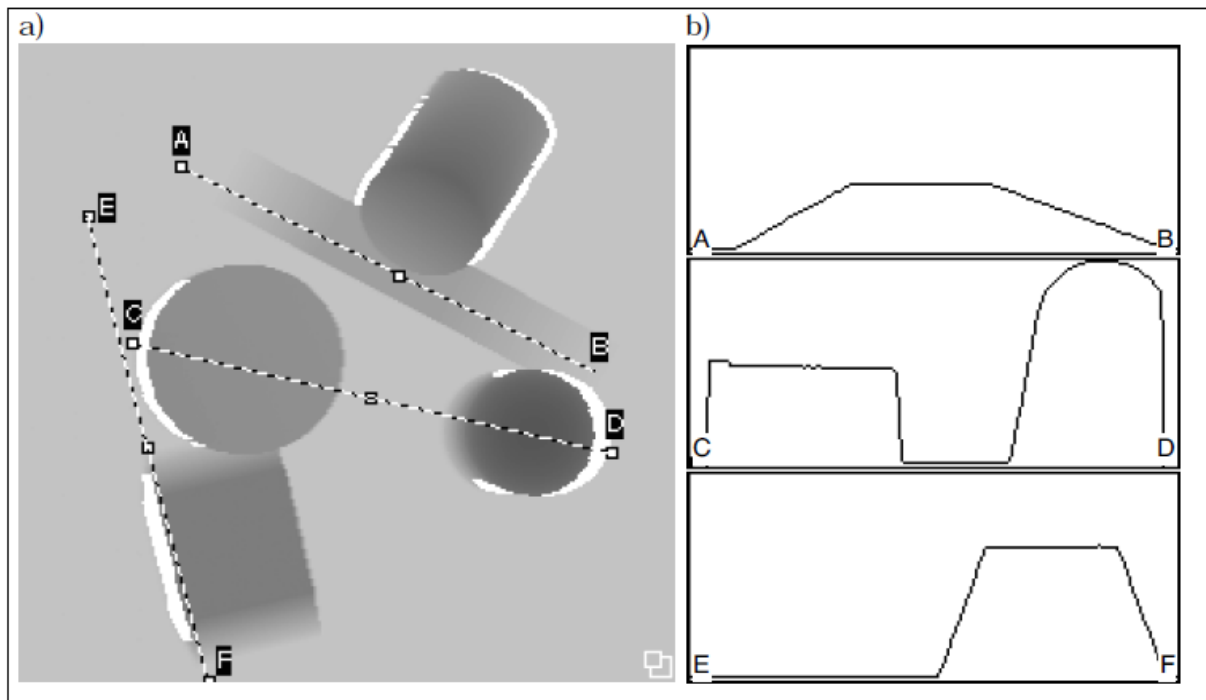


Figure 3.19 Exemples de discontinuités de profondeur et de discontinuités d'orientation. (a) Codage par niveaux de gris. Trois profils sont sélectionnés. (b) Représentation orthonormée des profils sélectionnés.

3.3.2.1 Détecteur simple (sans condition d'extremum)

La Figure 3.20 présente un exemple de signal auquel on associe le modèle de discontinuité, de manière évidente, les positions proches de la discontinuité se distinguent des autres par une valeur plus élevée de la pente du signal.

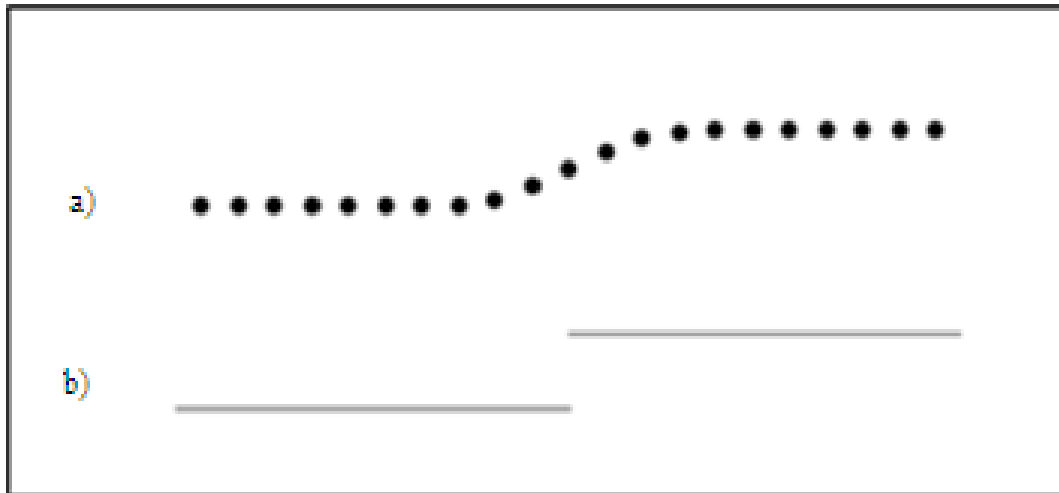


Figure 3.20 – Signal et modèle de discontinuité de profondeur associé.
(a) Le signal, (b) Le modèle.

Un seuillage sur la valeur de la pente permet de mettre en évidence les positions proches de la discontinuité par rapport aux autres et réalise donc un détecteur simple de discontinuité. Formellement, on attribue au signal discret les propriétés d'un signal continu dont on estime la dérivée. La détection des discontinuités se fait par seuillage sur la valeur absolue de la dérivée et nous disons qu'une pente élevée est l'indice d'une discontinuité et que la valeur absolue de la dérivée est une mesure de discontinuité notée Δ .

3.3.2.2 Détecteur avec condition d'extremum

En général, le détecteur différentiel simple de discontinuité marque plusieurs positions contiguës pour une seule discontinuité « *Figure 3.21* ». L'utilisation de la condition d'extremum est exprimée ici par le passage à zéro de la deuxième dérivée et permet donc d'obtenir une seule position marquée comme discontinuité (amincissement à 1 pixel). Pour une même grandeur différentielle (ici la dérivée), il existe plusieurs détecteurs de discontinuité qui se distinguent dans la manière avec laquelle la grandeur différentielle est estimée.

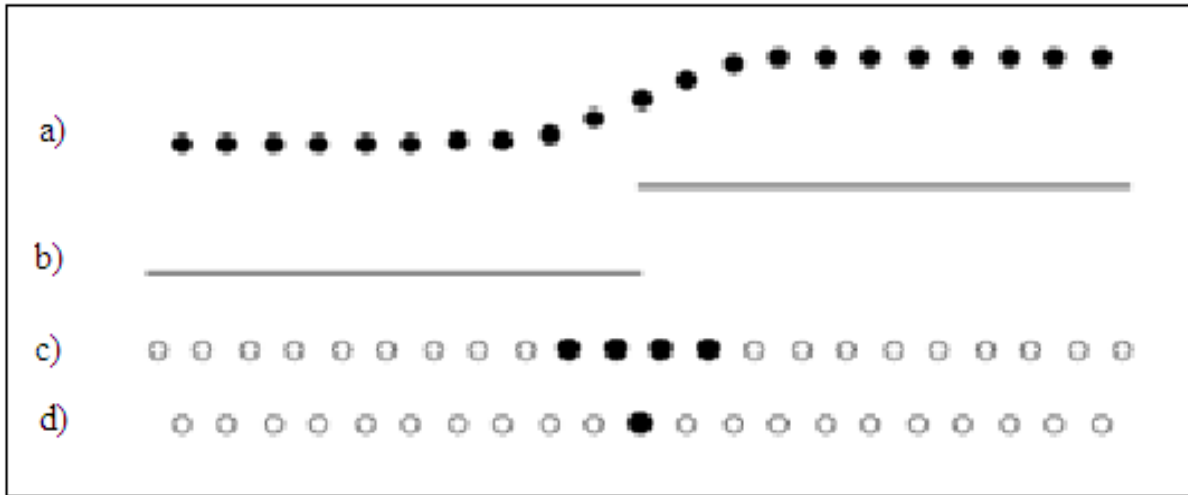


Figure 3.21 – Détection et condition d'extremum. (a) signal, (b) modèle de discontinuité, (c) résultat de détection sans condition d'extremum (points de discontinuité en noir), (d) résultat de détection avec condition d'extremum.

3.3.3 Grandeurs différentielles utiles à la détection des points de discontinuité

Nous présentons ici les grandeurs différentielles appropriées à constituer une mesure de discontinuité (ordre 0 ou 1) et traitons de leur estimation.

3.3.3.1 Mesures de discontinuité

Nous attribuons à une image de profondeur les propriétés d'une surface continûment dérivable $P(u, v)$ et on peut distinguer les mesures de discontinuité suivante :

- **Discontinuité de profondeur** : Une forte inclinaison de la surface $P(u, v)$ par rapport à la direction de mesure est l'indice d'une discontinuité de profondeur. Dans le cas d'une IP scalaire, ceci correspond à une forte pente de la surface « *Figure 3.22* ». Deux mesures de discontinuité de profondeur s'imposent, d'une part, le produit scalaire de la normale à la surface avec la direction de mesure et, d'autre part la norme ou le carré de la norme du gradient. La première exige de connaître la direction de mesure et la deuxième s'applique seulement à une IP scalaire.

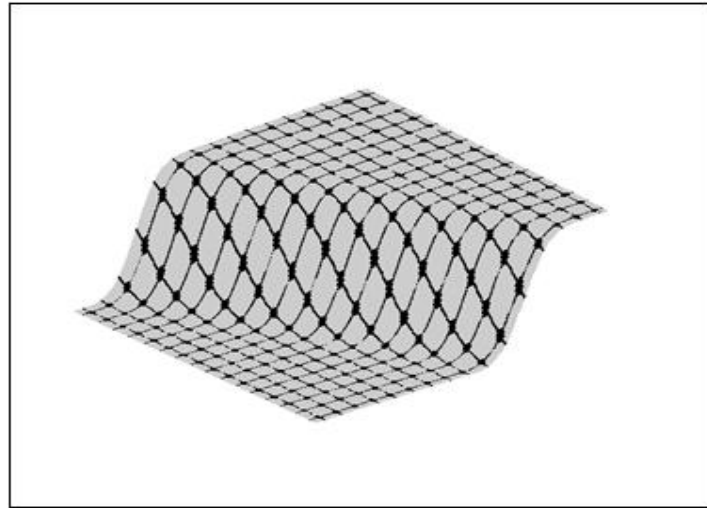


Figure 3.22 – Signal auquel on associe un modèle de discontinuité de profondeur.

- Discontinuité d'orientation** : Une valeur élevée de la courbure de la surface $p(u,v)$ dans au moins une direction est l'indice d'une discontinuité d'orientation « *Figure 3.23* » les mesures de discontinuité appropriées sont donc la courbure extrême en valeur absolue (maximum des valeurs absolues de la courbure minimale K_m et de la courbure maximale K_m) et la somme des carrés des courbures minimale K_m et maximale K_m Exprimée à l'aide de la courbure moyenne H et de la courbure gaussienne K . Cependant, pour une IP scalaire, une valeur élevée de la deuxième dérivée dans au moins une direction constitue aussi un indice pour une discontinuité d'orientation, dans ce cas, il y a donc deux mesures de discontinuité supplémentaires, la deuxième dérivée extrême en valeur absolue et la somme des carrés des deuxièmes dérivées minimale et maximale qui est égale à la variation quadratique.

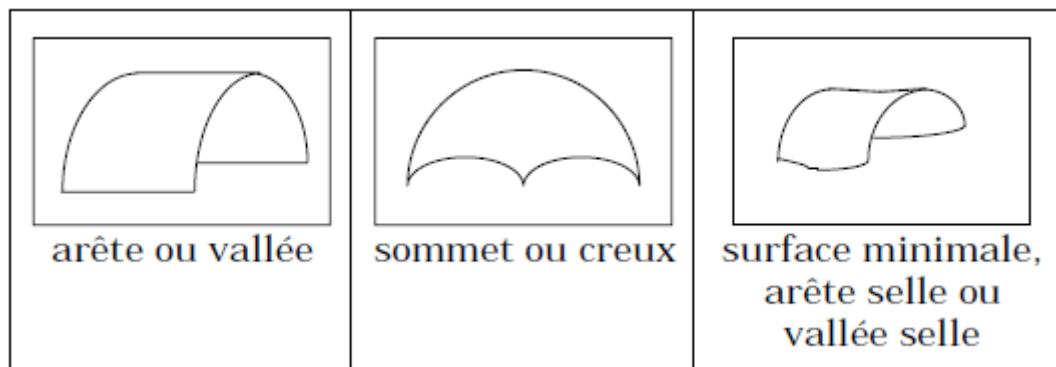


Figure 3.23 – Signaux auxquels on associe un modèle de discontinuité d'orientation.

- **Discontinuités de profils** : Pour une IP scalaire, les directions des axes de coordonnées (u,v) ainsi que les deux directions diagonales définissent en chaque position quatre profils plans de la surface $p(u,v)$. Les grandeurs différentielles de ces profils (dérivées d'ordre 1 et 2, courbure) constituent aussi des mesures de discontinuité appropriées pour la surface.

3.3.3.2 Estimation des grandeurs différentielles

Les grandeurs différentielles d'une image de profondeur sont estimées par des opérations de voisinage. Les opérateurs différentiels sont sensibles au bruit. La robustesse vis-à-vis du bruit s'obtient par augmentation de la taille du voisinage. Celle-ci peut être effectuée soit de manière directe, soit de manière indirecte par une opération de lissage préalable. L'augmentation de la taille du voisinage a peut causer une perte de résolution.

- **Estimation des dérivées partielles (IP scalaire)** : L'estimation de chacune des dérivées partielles d'une IP est réalisée par corrélation avec un masque approprié. Les masques conventionnels pour l'estimation des dérivées partielles d'ordre 1 d'une IP sont soit l'opérateur de Prewitt ou l'opérateur de Sobel. Pour les dérivées partielles de deuxième ordre P_u et P_v , on utilise les masques :

$$\mathbf{D}_{uu}^A = \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ -2 & \boxed{-2} & -2 \\ 1 & 1 & 1 \end{bmatrix}, \quad \mathbf{D}_{vv}^A = \frac{1}{3} \begin{bmatrix} 1 & -2 & 1 \\ 1 & \boxed{-2} & 1 \\ 1 & -2 & 1 \end{bmatrix} \quad (3.2)$$

- **Estimation des dérivées dans les 4 directions discrètes (IP scalaire)** : Au lieu d'être calculées à partir des dérivées partielles en utilisant les formules pour les dérivées directionnelles, les dérivées dans les quatre directions discrètes peuvent être estimées simplement par corrélation avec les masques appropriés. Les masques symétriques de taille minimale pour les dérivées d'ordre 1 sont :

$$\begin{aligned}
 \mathbf{D}_0^1 &= \frac{1}{2} \begin{bmatrix} -1 \\ \square \\ 1 \end{bmatrix} & \mathbf{D}_1^1 &= \frac{1}{2\sqrt{2}} \begin{bmatrix} -1 & & \\ & \square & \\ & & 1 \end{bmatrix} \\
 \mathbf{D}_2^1 &= \frac{1}{2} [-1 \ \square \ 1] & \mathbf{D}_3^1 &= \frac{1}{2\sqrt{2}} \begin{bmatrix} & & \\ & \square & \\ -1 & & \end{bmatrix}
 \end{aligned} \tag{3.3}$$

Pour les dérivées de deuxième ordre, les masques de taille minimale sont :

$$\begin{aligned}
 \mathbf{D}_0^2 &= \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} & \mathbf{D}_1^2 &= \frac{1}{2} \begin{bmatrix} 1 & & \\ & \square & \\ & & 1 \end{bmatrix} \\
 \mathbf{D}_2^2 &= [1 \ \square \ 1] & \mathbf{D}_3^2 &= \frac{1}{2} \begin{bmatrix} & & \\ & \square & \\ 1 & & \end{bmatrix}
 \end{aligned} \tag{3.4}$$

- **Calcul du gradient et des dérivées deuxièmes minimale et maximale (IP scalaire) :** Le calcul du gradient à partir des dérivées partielles P_u et P_v est immédiat. Sa norme vaut :

$$\| \text{Grad } P \| = \sqrt{P_u^2 + P_v^2} \tag{3.5}$$

- **Calcul de la normale et des courbures (IP cartésienne orthographique) :** L'estimation de la normale et des courbures pour une IP cartésienne orthographique peut être réalisée par calcul à partir des dérivées partielles. La courbure des profils correspondant aux directions horizontale, verticale et diagonales du tableau d'IP se calcule à partir des dérivées directionnelles P_{n1} et P_{n2} :

$$K_n = \frac{z_n''}{(1 + z_n'^2)^{3/2}} \tag{3.6}$$

Où :

$$\begin{aligned}
 z_0'^2 &= p_0' / \Delta x^2 & z_0'' &= p_0'' / \Delta x^2 \\
 z_1'^2 &= p_1' / (\Delta x^2 + \Delta y^2) & z_1'' &= p_1'' / (\Delta x^2 + \Delta y^2) \\
 z_2'^2 &= p_2' / \Delta y^2 & z_2'' &= p_2'' / \Delta y^2 \\
 z_3'^2 &= p_3' / (\Delta x^2 + \Delta y^2) & z_3'' &= p_3'' / (\Delta x^2 + \Delta y^2)
 \end{aligned}$$

- **Estimation de la normale et des courbures (IP vectorielle) :** L'estimation de la normale pour une IP vectorielle est réalisée par une méthode d'approximation de plan sur un voisinage de la position considérée. La valeur de la normale est déterminée par le plan qui approche au mieux les rangels du voisinage, au sens des moindres carrés de la distance des points au plan. Du point de vue du calcul, la normale est donnée par le vecteur propre correspondant à la plus petite valeur propre d'une matrice 3×3 . La matrice est construite à partir des valeurs des rangels. Pour l'estimation des courbures, trois méthodes sont possibles :
 - La première méthode est basée sur une triangulation de la surface. La courbure gaussienne et d'autres caractéristiques intrinsèques de la surface en un point sont estimées par calcul à partir de la longueur des côtés des triangles ayant ce point pour sommet. Cette méthode n'utilise que les voisins immédiats de la position considérée.
 - La deuxième méthode est basée sur une estimation de la courbure le long d'une courbe définie à partir de deux points \mathbf{p} et \mathbf{q} . La valeur de courbure est estimée par le quotient dont elle est la limite: l'angle entre les normales à la surface en ces points, divisé par la distance entre les points. La valeur de l'angle entre les normales est approchée par la norme de la différence des normales. Un voisinage de $n+1$ points définit n valeurs de courbure en un point. Les valeurs extrêmes sont une estimation des courbures minimale et maximale en ce point. On peut en tirer une estimation des courbures moyenne et gaussienne en appliquant les formules correspondantes. L'éloignement des points tend à surestimer la courbure puisque la distance euclidienne $[\mathbf{p} - \mathbf{q}]$ sous-estime la distance entre les deux points le long de la surface. Pour cette raison, le voisinage doit rester de taille restreinte.
 - La troisième méthode utilise un parabolôïde d'approximation dans un système de coordonnées locales. L'estimation de la normale fixe le vecteur \mathbf{E} du système de coordonnées locales. Les coordonnées des rangels faisant partie d'un voisinage choisi sont calculées dans ce nouveau système. L'évaluation des paramètres du parabolôïde de meilleure approximation permet de déterminer les courbures minimale et maximale et leur direction respective dans le repère local. Ces dernières sont finalement transformées dans le repère global.

3.3.4 Revue des détecteurs de point de discontinuité

Les détecteurs de point de discontinuité décrits dans la littérature sont de type différentiel. Nous les divisons en deux catégories, suivant s'ils appliquent ou non la condition d'extremum.

3.3.4.1 Détecteurs sans condition d'extremum

Chen et Kak [Park & al, 2008] détectent les discontinuités de profondeur sur la base de la distance euclidienne entre rangels voisins. La mesure servant à la détection des discontinuités d'orientation est une estimation de la courbure de la surface le long d'une courbe reliant deux rangels voisins. La courbure est estimée par le quotient dont elle est la limite: variation d'angle de la normale à la surface divisée par le chemin parcouru. Pour l'estimation des vecteurs normaux, une technique de voisinage adapté est utilisée: la normale est celle du plan de meilleure approximation parmi les approximations sur les N^2 voisinages de taille $N * N$ auxquels appartient la position (i, j) . La taille de voisinage utilisée est de $5 * 5$.

3.3.4.2 Détecteurs avec condition d'extremum

- Fan et al. [Fan & al, 1987] traitent des images de profondeur de type scalaire. Deux méthodes de détection des discontinuités sont proposées. Toutes deux utilisent la courbure :
 - Dans la première méthode, la détection des discontinuités se fait sur la base de la courbure extrême K_e (c'est-à-dire celle d'entre les courbures minimale et maximale qui est la plus grande en valeur absolue). Les points correspondant à un passage par zéro de la courbure extrême sont retenus comme points de discontinuité de profondeur. Il s'agit d'une condition similaire à celle qui impose un extremum du gradient. Elle est utile à l'amincissement. Cependant, il manque une autre condition (similaire au seuillage sur la norme du gradient) permettant d'éliminer les passages par zéro non significatifs. La valeur absolue de la courbure extrême sert de mesure pour les discontinuités d'orientation. La condition d'extremum est utilisée pour réaliser l'amincissement.
 - La deuxième méthode est similaire à la première. Mais la courbure extrême est remplacée par une combinaison des courbures de profils dans les quatre

directions discrètes correspondant aux axes et diagonales du tableau. Les points de discontinuité de profondeur sont déterminés par le passage à zéro de la courbure de l'un des profils. Les points de discontinuité d'orientation sont déterminés par le passage à une valeur extrême de la courbure de l'un des profils, à condition que cette valeur soit suffisamment élevée.

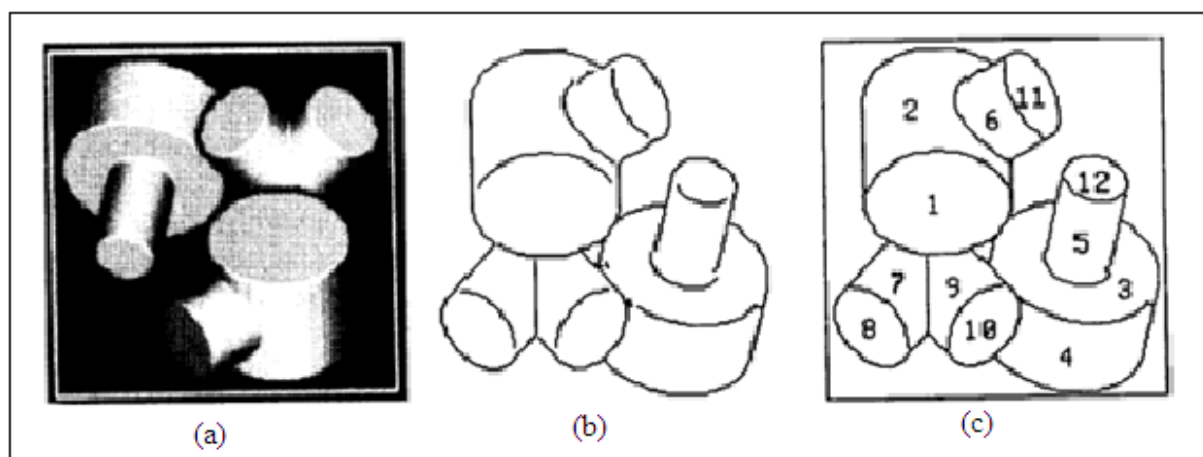


Figure 3.24 – Détection et condition d'extremum. (a) scène originale, (b) détection de discontinuité, (c) Résultat de segmentation

Les détecteurs avec condition d'extremum réalisent un amincissement qui n'est pas nécessaire. De plus, la condition d'extremum génère des interruptions de frontières. La Figure 3.24 montre cet effet dans le cas de la détection des discontinuités de profondeur avec un amincissement par condition d'extremum du gradient: il peut arriver que la connexité des courbes soit perdue en présence d'une jonction en T (Il s'agit du même problème qui est rencontré dans la détection des contours d'une image d'intensité par passage à zéro du Laplacien).

- Davignon [Davignon, 1992] travaille avec des IP de type scalaire. Pour la détection des discontinuités de profondeur « Figure3.25 », la norme du gradient sert de mesure de discontinuité. L'amincissement de la région de discontinuité se fait en cherchant le maximum dans la direction du gradient. Les dérivées deuxièmes minimale et maximale sont utilisées comme mesure de discontinuité d'ordre 1. La condition d'extremum des dérivées minimale et maximale dans leur direction respective sert à l'amincissement.

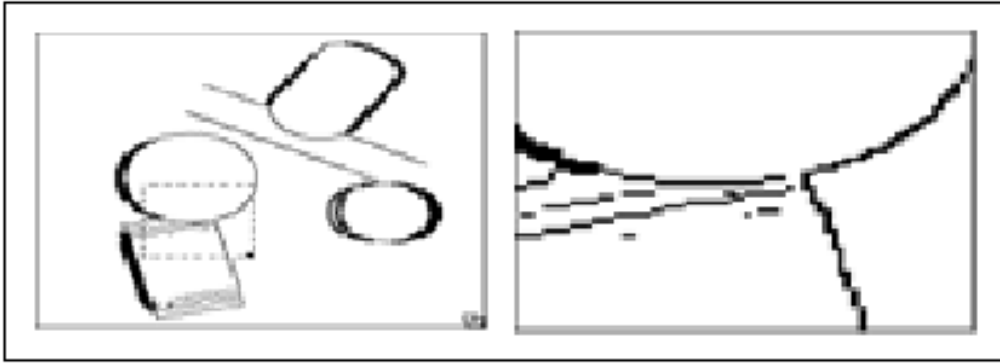


Figure 3.25 – Image Disparition de connexité lors de l'amincissement d'une discontinuité de profondeur par condition d'extremum du gradient.

3.3.5 Détection conjointe des discontinuités (C_0, C_1) à l'aide d'un unique détecteur

La continuité d'ordre 0 est une condition nécessaire à la continuité d'ordre 1. Par conséquent, les mesures de discontinuité d'orientation devraient aussi s'appliquer aux discontinuités de profondeur. On peut alors utiliser des mesures de discontinuité d'orientation pour détecter les deux types de discontinuité à la fois (C_0 et C_1).

3.3.5.1 Mesures de discontinuité pour une IP cartésienne orthonormée

Ce sont les mesures de discontinuité d'orientation inventoriées précédemment (discontinuité de profondeur, discontinuité de profil, discontinuité d'orientation) qui sont estimées sur un voisinage de taille minimale (3×3). La *Figure 3.26* présente les mesures de discontinuité d'une image profondeur (Conventionnellement les points de discontinuité sont représentés en noir), puis, deux résultats de seuillage sur ces mesures (seuillage élevé puis seuillage bas). Il est à constater les faits suivants:

- Les mesures de discontinuité d'orientation basées sur la courbure donnent de mauvais résultats de détection des discontinuités de profondeur « *Figure 3.26* », la raison en est une mauvaise estimation de la courbure par les opérateurs utilisés.

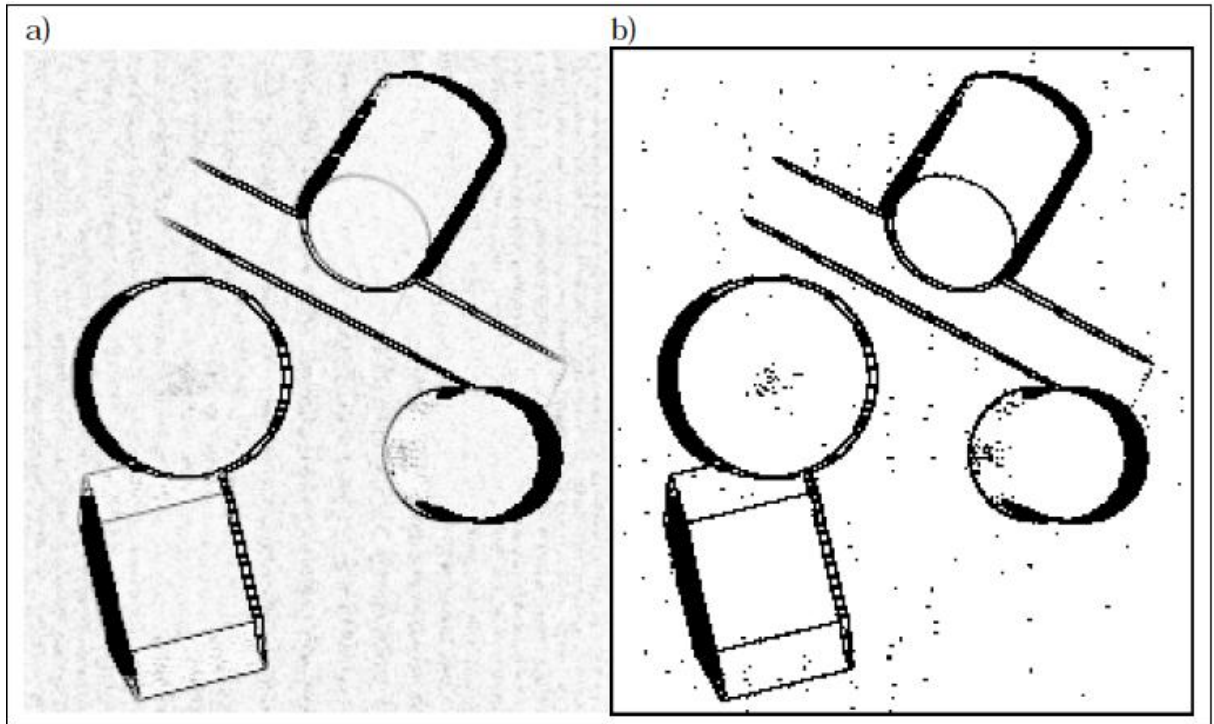


Figure 3.26 – Détection conjointe des discontinuités d'ordre 0 et 1. (a) Mesure de discontinuité codée par des niveaux de gris. (b) Image des points de discontinuité obtenue par seuillage sur la mesure de discontinuité.

- Les mesures de discontinuité basées sur la deuxième dérivée réagissent aux deux types de discontinuité. Cependant, elles sont plus sensibles aux surfaces courbes fortement inclinées. Ce phénomène n'est pas lié à l'échantillonnage, la deuxième dérivée (extrême) d'une surface courbe inclinée est effectivement élevée, ceci a comme inconvénient que de telles surfaces obtiennent l'étiquette de discontinuité suite à la décision par seuillage.
- Toutes les mesures de discontinuité sont fortement sensibles au bruit de l'image de profondeur« *Figure 3.27* », il s'en suit que certaines discontinuités ne peuvent pas être détectées parce qu'on ne peut pas abaisser suffisamment le seuil de détection.

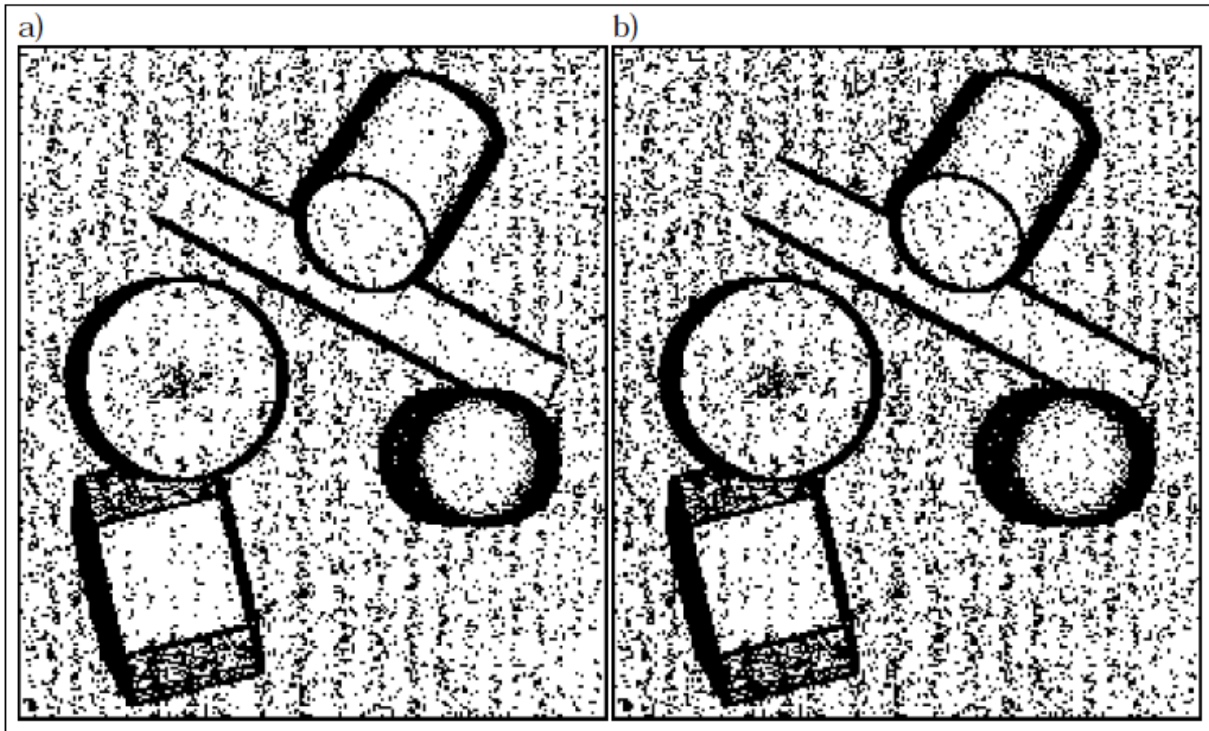


Figure 3.27 – Détection conjointe des discontinuités, (a) discontinuités d'ordre 0, (b) discontinuités d'ordre 1

3.3.5.2 Extension aux autres types d'IP

Les résultats présentés jusqu'à maintenant sont liés à un type d'image de profondeur particulier (cartésienne orthonormée). Dans ce qui va suivre nous montrerons comment les mesures de discontinuité peuvent être étendues aux autres types d'IP.

- **IP scalaires** : Formellement, toutes les mesures de discontinuité peuvent être appliquées à n'importe quelle IP scalaire. A ce moment-là, c'est la surface $P(u,v)$ plutôt que $z(x,y)$ qui est analysée. Qualitativement, les résultats devraient être similaires à ceux obtenus en transformant la surface dans le repère cartésien.
- **IP vectorielles** : La mesure de discontinuité peut être appliquée aux IP de type vectoriel, il suffit d'estimer les courbures avec les méthodes appropriées, La mesure de discontinuité de profil peut aussi être appliquée à une image de type vectoriel. Le profil en question se trouve dans le plan passant par le rang de position (i, j) et ses deux voisins dans la direction discrète de référence, la courbure est estimée à partir de la taille des segments consécutifs définis par les trois points et de l'angle que forment ces segments.

3.4 Conclusion

Les méthodes de segmentation d'image de profondeur ont été passées en revue, d'abord les méthodes élémentaires, puis les méthodes formant l'état de l'art du domaine. Pour les cinq méthodes élémentaires recensées, nous avons relevé l'existence d'un lien étroit entre le critère d'homogénéité et la méthode. Si deux des cinq méthodes s'appliquent à plusieurs critères d'homogénéité (modèle de surface), les autres sont spécifiques à seul un critère d'homogénéité. Nous avons aussi présenté les détecteur de points de discontinuité, et nous constatons que tous les auteurs effectuent une détection séparée des discontinuités C^0 et C^1 , ce qui n'apporte pas de bons résultats. D'autre part, ce sont souvent les détecteurs avec condition d'extremum qui sont utilisés alors qu'ils fonctionnent mal. Dans le chapitre suivant nous allons présenter le principe des contours actifs en posant quelques définitions, puis, nous décrirons les différents types de contours actifs utilisés en segmentation d'image.

Chapitre 4

Les contours actifs

4.1 Introduction

Les contours actifs constituent le premier exemple notable des modèles déformables, historiquement, ils ont été introduits par Michael Kass, Andrew Witkin et Demetri Terzopoulos en 1988 et font figure de référence en la matière, le mot snake (serpent en français) se justifie par la déformation du contour lors de sa convergence, le critère selon lequel le contour se déforme en faisant évoluer le contour initial associé à la recherche d'une minimisation d'une fonctionnelle interprétable en terme de contraintes physiques et la convergence du contour vers la frontière recherchée est considérée comme une condition de stabilité. Les contours actifs sont largement utilisés en segmentation d'images pour leur capacité à intégrer les processus de détection et de chaînage des contours en un seul processus de minimisation d'énergie [Khalifa Nawres & al, 2005], toutefois l'estimation des paramètres et les problèmes d'initialisation font des contours actifs une méthode difficile à calibrer. Dans la littérature, deux approches de contour actif sont proposées, l'approche par contour et l'approche par région. Les domaines d'utilisation sont nombreux tant en 2D qu'en 3D tels la reconnaissance de formes, la simulation, le suivi de scènes, la segmentation d'images. Dans ce chapitre, nous allons voir tout d'abord le principe des contours actifs en posant quelques définitions puis nous verrons quelques modèles de contours actifs.

4.2 Principe et définitions de la méthode contours actifs

Les contours actifs sont définis par une courbe continue, fermée ou non, à extrémités fixes ou non [Chen & al, 2006]. Ils se déforment à partir d'une position d'initialisation située près de l'objet d'intérêt. Le modèle est soumis à des forces qui le déforment et le déplacent dans l'image. L'énergie du modèle « Figure 4.1 » comprend un terme d'énergie interne de régularisation ou de lissage qui correspond à la fonctionnelle de Tikhonov et un terme d'énergie externe ou

d'adéquation aux données, auquel des contraintes en terme d'énergie peuvent être ajoutées.

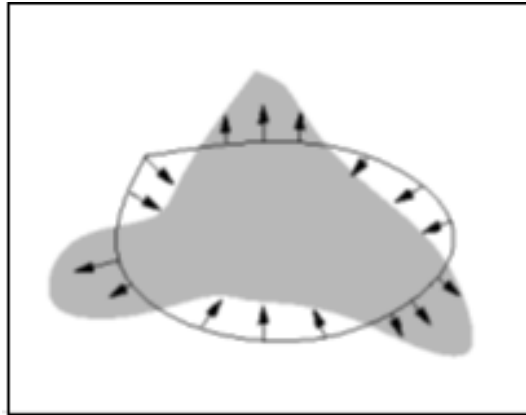


Figure 4.1 – Forces régissant l'évolution du contour actif

Un contour peut approximativement être défini comme une zone de l'image où l'intensité des pixels changent brusquement. Cette discontinuité dans l'image est le passage d'un niveau de gris à un autre, de manière plus ou moins rapide, donnant lieu à des interprétations d'échelons, de rampe, de toit ou de ligne. La détection de contours peut être précédée par une phase de réduction du bruit. Ce prétraitement risque, par contre, d'adoucir, voire de faire disparaître certains bords. La détection de contours peut aussi être précédée d'une phase de renforcement de contours. Ce prétraitement peut également accentuer le bruit et faire apparaître des artefacts inutiles. La méthode de détection des contours actifs nécessite l'initialisation d'une ligne de contour en entourant la zone des objets à détecter et en faisant évoluer cette courbe « *Figure 4.2* » de façon à minimiser son énergie E .

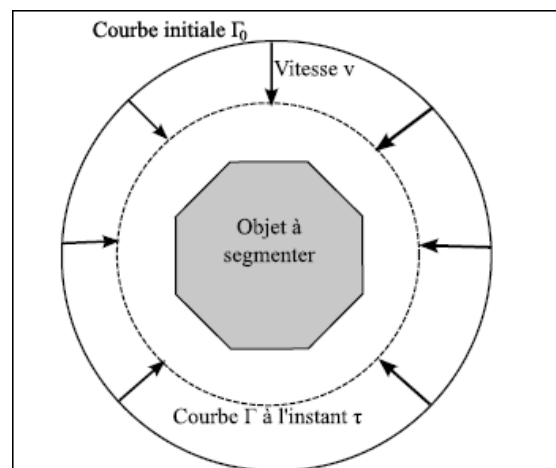


Figure 4.2 – Forces régissant l'évolution du contour actif

Les développements théoriques sont étudiés dans le domaine continu puis discrétisés pour la programmation. Dans le domaine continu, les approches dérivatives sont les plus intuitives pour détecter les variations d'amplitude. Une dérivation du premier ordre présentera un maximum pour le contour ; une dérivée seconde présentera un passage par zéro. L'identification d'une transition peut être obtenue par seuillage de la norme du gradient. Les contours actifs tirent leur origine des modèles élastiques, mais la communauté s'accorde à les attribuer à l'équipe Kass, Witkin et Terzopoulos [Kass *et al.* 1988] qui introduisirent les snakes ou courbes minimisantes. Les snakes tiennent leur nom de leurs aptitudes à se déformer comme des serpents. Depuis la publication de cette équipe, les modèles déformables sont devenus un sujet très important pour la communauté du traitement d'images. De très nombreuses équipes s'y sont intéressées de manière plus ou moins approfondie.

4.2.1 L'énergie interne du contour (énergie propre)

Une énergie propre [Flore Faille & all, 1999] est due uniquement à la forme du contour, dite énergie interne (E_{int}) qui va dépendre uniquement de la forme du snake, c'est une contrainte de régularité sur la courbe et s'écrit :

$$E_{int} = (a(s)) \cdot || Vs(s) || + b(s) \cdot || Vss(s) || \quad (4.1)$$

Où a et b sont les poids respectifs accordés à la dérivée première Vs et à la dérivée seconde Vss , a et b seront ajustés pour trouver un intermédiaire entre un contour souple mais pas forcément très régulier (et qui pourra alors se caler sur des coins et des angles aigus) et un contour très régulier qui suivra bien le contour sans s'accrocher sur des petites aspérités.

4.2.2 L'énergie image (énergie potentielle)

Une énergie potentielle imposée par l'image caractérise les éléments vers lesquels on veut attirer le snake sur l'image [Li & al, 2007], pour notre problème, puisqu'il s'agit de détecter des contours, on va s'intéresser aux lignes de fort gradient:

$$E_{image} = -gradient(i) \quad (4.2)$$

Pour certains problèmes, on peut choisir d'attirer le snake vers des zones présentant d'autres caractéristiques, vers les zones les plus sombres par exemple, un minimum d'énergie sera

atteint si la courbe passe par les points de gradient maximal de l'image lisse par un filtre gaussien (Le signe négatif indique que les gradients les plus grands minimiseront cette énergie et attireront le snake).

4.2.3 L'énergie de contrainte du contour

Une énergie E_{Contour} qui exprime certaines contraintes supplémentaires qui peuvent être imposées par l'utilisateur vu le snake qu'il veut obtenir, il peut par exemple imposer une distance minimale ou maximale entre deux points consécutifs du contour actif, elle va s'exprimer en fonction de $V(s)$ uniquement. La segmentation est donc réalisée à travers un processus de minimisation d'une énergie notée E exprimée par :

$$E(\text{Contour}) = E_{\text{interne}}(\text{Contour}) + E_{\text{image}}(\text{Contour}) + E_{\text{contrainte}}(\text{Contour}) \quad (4.3)$$

4.2.4 Convergence du contour actif

L'algorithme du snake s'arrête lorsqu'il atteint un état stationnaire dans lequel aucun point du snake ne change de position, afin d'éviter que la recherche ne se poursuive indéfiniment dans le cas où les points continuent à changer leurs positions, un nombre d'itérations maximum est déterminé, si ce nombre est atteint, la recherche prend fin et le résultat de l'itération actuelle est proposé comme solution finale.

4.2.5 Déformation du contour actif

Pour déformer le contour actif [Bresson & al, 2007], on pose $v(x, y)$ comme étant la coordonnée généralisée pour la réduction des équations du mouvement donné par l'équation de Lagrange, d'où la minimisation de la fonction J suivante :

$$J = \int L(v(s, t)) ds \quad (4.4)$$

Où $L(v(s, t))$ est le Lagrangien du contour et Le Lagrangien L est défini par :

$$L(v) = K(v) - U(v) \quad (4.5)$$

$K(v)$ étant l'énergie cinétique du contour due à son mouvement et donnée par la formule :

$$\text{Courbe } S(i) = \text{mesure - angle} \frac{(S_{i-1}S_i, S_{i-1}S_i)}{(|S_{i-1}S_i| - |S_{i-1}S_i|)} K(v) = \frac{1}{2} \int \mu(s) v_i^2 ds \quad (4.6)$$

$\mu(s)$: étant la densité linéique (quantité de charge sur une ligne).

$U(v)$ Energie potentielle du contour due à sa position, elle dépend de l'énergie interne du modèle physique (élasticité et rigidité) et de l'énergie potentielle externe associée à l'image :

$$U(v) = E \text{ snake} = \int E(v(s)) ds = \int [Interne(v(s)) + Externe(v(s))] ds \quad (4.7)$$

La solution du problème de minimisation de l'énergie se résume à trouver un chemin qui possède un état stationnaire, le minimum doit toujours satisfaire l'équation d'Euler Lagrange :

$$\frac{d}{dt} \left(\frac{\partial L}{\partial V_t} \right) = \frac{\partial L}{\partial V} \quad (4.8)$$

Où $\partial(L)$ étant la dérivée partielle de L (dérivée par rapport à l'une de ses variables, les autres étant gardées constantes).

En remplaçant les expressions de K et U le Lagrangien s'exprimera comme suit :

$$L = \frac{1}{2} \int \mu(s) |v_i^2| - a(s) |v_x(s)|^2 - b(s) |v_y(s)|^2 - E_{externe}(v(s)) ds \quad (4.9)$$

Nous rajoutons les conditions aux limites et les conditions initiales, l'évolution du contour est basée sur le déplacement du contour de façon itérative de telle sorte que les forces internes et les forces externes soit équilibré, l'état final est obtenu lorsque la dérivée de $V(s,t)$ approche de zéro, plusieurs méthodes ont été utilisées pour la résolution de cette équation.

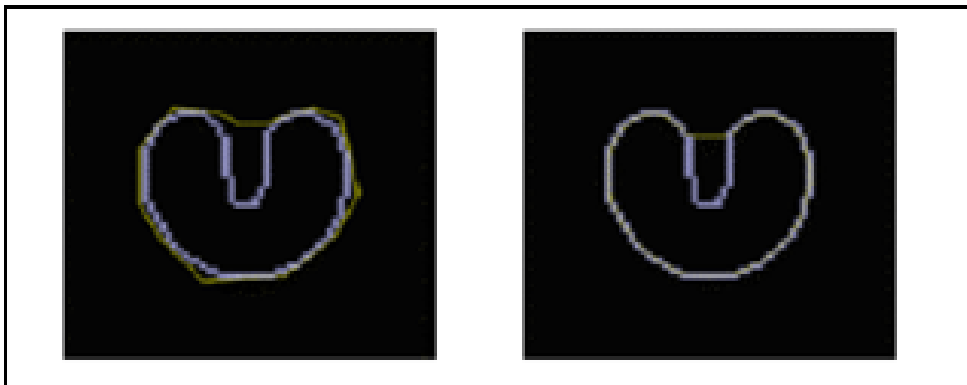


Figure 4.3 – Limites de convergence de contour avec la méthode classique

4.2.6 Limites et améliorations proposées

On remarque la difficulté du choix des divers paramètres, ainsi que l'influence prépondérante de l'initialisation sur la convergence du contour actif, plus généralement, ce modèle peut être classé parmi les approches contour en ce sens que l'information utilisée est exclusivement une information contour, il est donc bien adapté à une certaine classe de problèmes mais peut être mis en échec en présence de fort bruit, de plus, il est très sensible à l'initialisation. Récemment, de nombreuses améliorations ont été proposées afin de dépasser ces limites, par exemple l'ajout d'une force de pression pour permettre au contour de se comporter comme un ballon ou une nouvelle force externe, appelée gradient vecteur flow (GVF), est proposée, ces méthodes seront présentées d'avantages dans ce document.

4.3 Type de contour actif

Il existe de nombreux types de contours actifs et de nombreuses modifications et améliorations ont été proposées, dans ce qui suit nous nous intéressons au modèles de base les plus connus et les plus utilisés en segmentation par modèles déformables.

4.3.1 Le modèle ballon

Le modèle classique ayant pour caractéristique de nécessiter une initialisation et que le bruit risquer de provoquer des minima locaux d'énergie avant d'atteindre le contour cherché, David Laurent Cohen a introduit la notion de "force de ballon" et cette force, selon son sens [Tahvilian & al, 2012], permet au snake de se gonfler ou de se rétracter. Dans cette approche le contour est identifié par une suite de points discrets représentant les sommets du contour associé à une ligne polygonale, le principe d'évolution consiste à construire une suite de contours déformés selon un critère de mesure d'énergie locale, un tel modèle s'associe à l'évolution d'une bulle, cette évolution qui se matérialise par une croissance de la bulle à laquelle on associe une fonction d'énergie :

$$E(\text{Contour}) = E_{\text{interne}}(\text{Contour}) + E_{\text{image}}(\text{Contour}) + E_{\text{contrainte}}(\text{Contour}) \quad (4.10)$$

La condition d'arrêt correspond à une rupture du modèle et se voit par un éclatement de la bulle où le contour peut se propager librement au-delà de la frontière désirée, le contour est

associé à un polygone qui est donnée sous forme d'une liste chaînée de sommets $S(i)$ où chaque sommet est identifié par ses coordonnées (x_i, y_i) , pendant les itérations, chaque sommet $S(i)$ peu évoluer vers un nouvel emplacement situé à une distance fixe de dernier emplacement et selon une direction orthogonale à la direction donnée par le segment $S(i-1)$ et $S(i+1)$, à chaque sommet on associe la mesure de distance au sommet précédent, la mesure de distance au sommet suivant et une mesure de courbure locale donnée par l'équation suivante :

$$\text{Courbe}S(i) = \text{mesure - angle} \frac{(S_{i+1}S_i, S_{i-1}S_i)}{(|S_{i+1}S_i| - |S_{i-1}S_i|)} \quad (4.11)$$

La fonction d'énergie se décompose en une somme d'une fonction d'énergie interne et d'une fonction d'énergie externe, la valeur globale d'énergie en chaque point intègre les deux mesures précédentes pondérées par le paramètre λ qui précise l'importance donnée à la courbure vis-à-vis de l'information de gradient image, ceci selon l'équation suivante :

$$E(c) = E \text{ interne}(c) + \lambda E \text{ externe}(c) \quad (4.12)$$

L'initialisation de snake se fait par le positionnement manuel d'un polygone à l'intérieur de la région à segmenter, le passage d'une itération à une autre consiste à rechercher le sommet d'énergie minimale et à l'évoluer vers sa nouvelle position, la nouvelle position de sommet est à rapprocher de la notion de raideur dans les modèles de snakes impliquant le paramètre α , la distance entre l'ancienne position du sommet et la nouvelle position est majorée par un seuil qui traduit l'élasticité associée au paramètre β (ce paramètre s'associe à une notion de vitesse de propagation), l'utilisateur peut intervenir sur le paramètre de vitesse de propagation, le paramètre de raideur et le paramètre d'élasticité selon le type d'images qu'on veut segmenter.

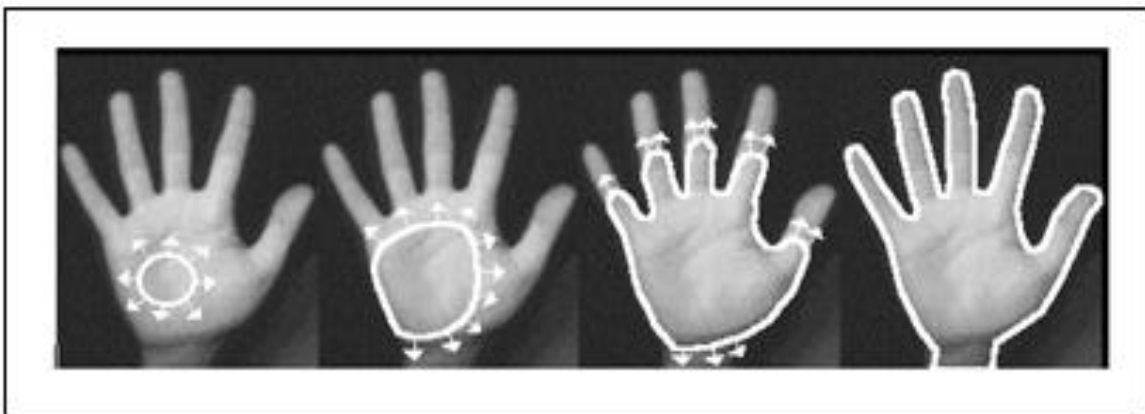


Figure 4.4 – Evolution du contour « ballon » sur une main

La méthode de ballon a été utilisée « *Figure 5.4* » pour détecter les frontières d'un objet dans une image et des tests ont été effectués pour détecter les frontières de la main dans plusieurs positions et la Figure 5.5 illustre le nombre de polygone utilisés a chaque itération.

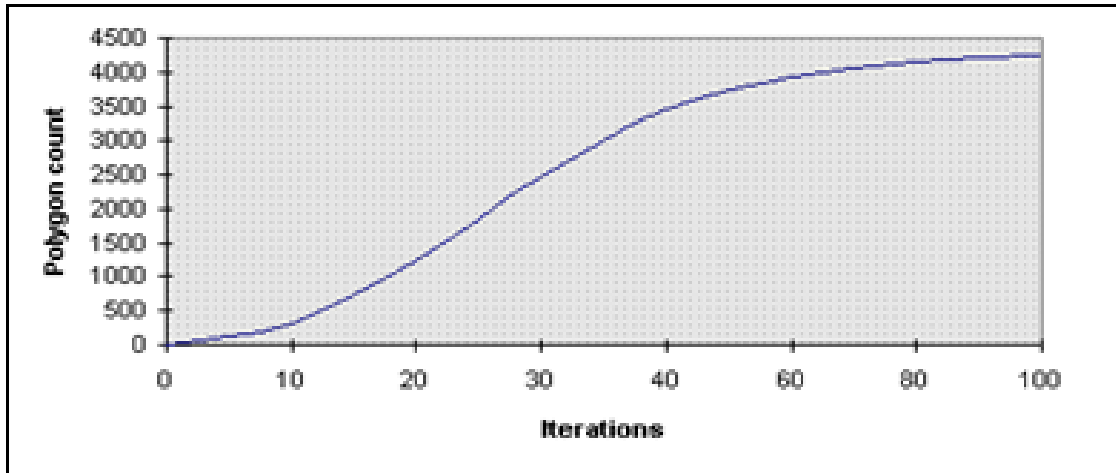


Figure 4.5 – Courbe représentant le nombre de polygone utilisés pour chaque itération

4.3.2 Contour actif par la méthode de gradient vector flow (GVF)

Cette méthode (gradient vector flow) a été proposée en 1997 par Chenyang Xu et Jurry L. Prince, la méthode a été développée dans le cadre d'une approche plus générale en traitement d'image reposant sur le concept de contour actif [Ikhlef *BECHAR*, 1997], dans le but de ramener une solution plus ou moins satisfaisante à certaines limitations de l'approche classique telles que l'initialisation du snake et sa convergence vers les régions concaves. L'approche contour par la méthode de GVF se divise en deux étapes :

- Calcul de la carte de contours (*EdgeMap*) : La carte de contours permet de caractériser les zones d'intérêt dans l'image.
- Calcul de flux du vecteur du gradient : Le flux de vecteur du gradient (GVF) est défini comme le champ de vecteurs $v(x, y)$ qui minimise l'équation d'énergie suivante :

$$E = \iint \mu(ux^2 + uy^2 + vx^2 + vy^2 + |f_x|^2 + |v - f_x|^2) dx dy \quad (4.13)$$

Pour cela il faudra que V satisfasse les deux équations d'Euler suivantes :

$$\mu \nabla u^2 - (u - f_x)(f_x^2 + f_y^2) = 0 \quad \text{Et} \quad \mu \nabla v^2 - (v - f_x)(f_x^2 + f_y^2) = 0 \quad (4.14)$$

Où f_x et f_y sont les dérivées de f par rapport à x et à y , μ paramètre de régularisation, ∇ opérateur de Laplacien.

Il faut noter que lorsque v tend à prendre la valeur du gradient f là où ce dernier est suffisamment grand, et tend à se lisser là où ce dernier est petit, et entre ces deux situations, v réalise un compromis entre les deux termes de l'énergie à travers le coefficient μ (le paramètre de régularisation).

Une fois le champ GVF v est calculé [Cheng & al, 2006], il est introduit dans l'équation d'Euler associée à la fonctionnelle d'énergie optimale d'un snake comme suit :

$$\frac{\partial x(s,t)}{\partial t} = \alpha x'(s,t) - \beta x''(s,t) + v(s,t) \quad (4.15)$$

La *Figure 4.6* illustre l'utilisation du contour actif GVF et qu'on remarque que le snake converge correctement vers les concavités.

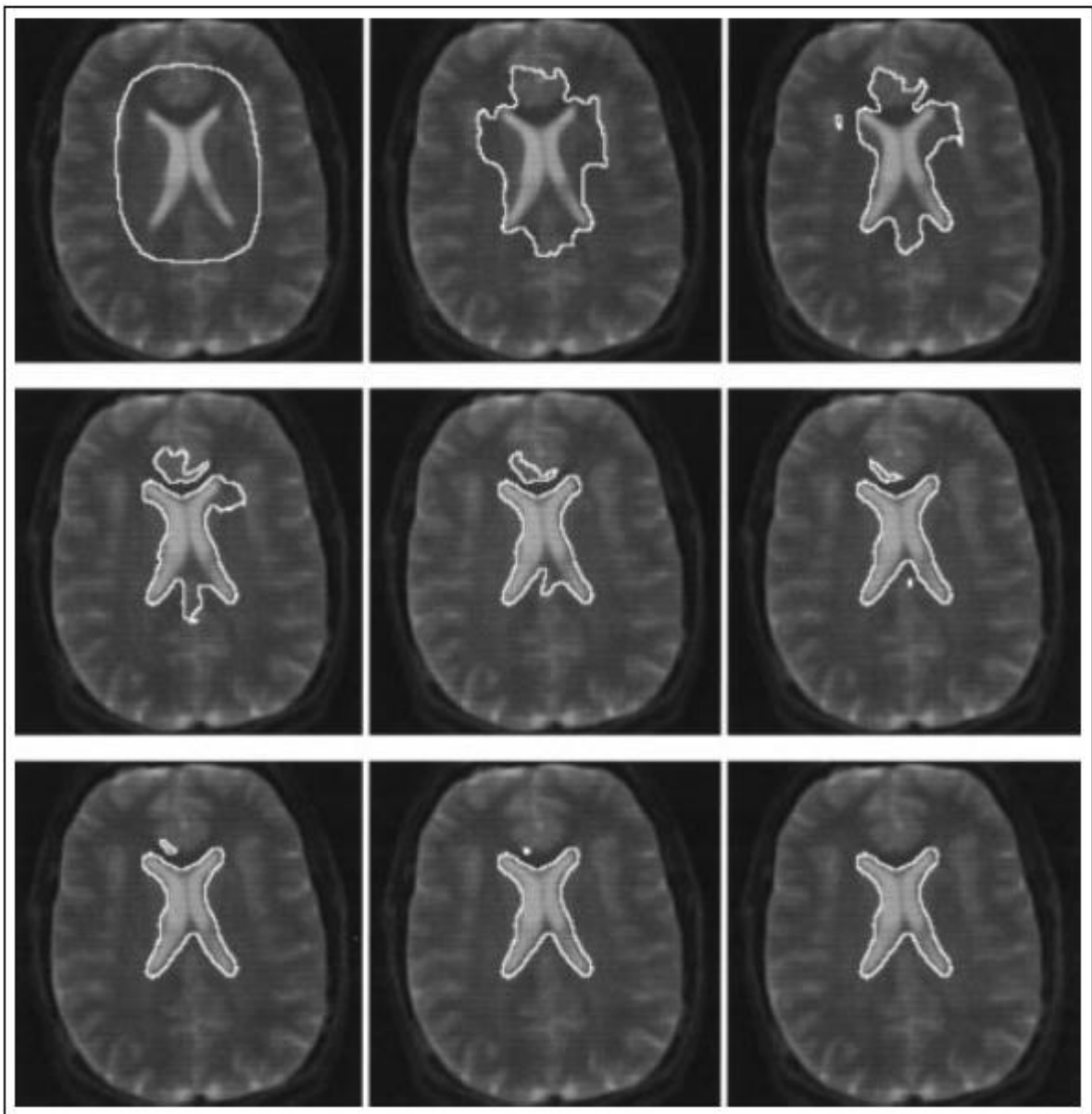


Figure 4.6 – Exemple d'évolution du contour actif avec la méthode de GVF

4.3.3 Le contour actif ASM (Actif Shape Model)

Le Modèle Actif de forme « ASM » peut être considéré comme une spécialisation du modèle snake, dans le sens où si on considère que l'ASM [Ginneken & al, 2002] est un snake dans lequel on n'utilise pas la même forme pour identifier n'importe quel objet mais on incorpore a priori une information ou une connaissance sur l'objet recherché dans l'image, de ce fait l'ASM est aussi appelé « smart snake, l'ASM de par son fondement statistique nécessite l'utilisation d'un échantillon qui doit être représentatif le plus possible de la population concerné par l'étude. D'autant plus que la méthode ASM «Figure 4.7 » se base sur un modèle qui guide la recherche des objets dans les nouvelles images et n'accepte aucune forme qui ne soit pas représentée dans l'ensemble des formes défini par ce dit modèle, cette contrainte constitue donc une première difficulté de la méthode car d'un côté pour que la méthode accepte un nombre important de formes dans son modèle il faut que l'échantillon utilisé dans la phase d'apprentissage soit assez grand ce qui implique un investissement en temps important notamment dans l'étape de marquage, de l'autre côté, si on opte pour la minimisation du temps d'exécution, il va falloir minimiser la taille de l'échantillon ce qui donne un modèle qui accepte un nombre restreint de formes et cela influe énormément dans la capacité de trouver les objets dans de nouvelles images.

- Calculer un profil pour chaque point de référence de la forme moyenne pendant la phase d'apprentissage.
- Poser la forme moyenne au plus près des formes recherchées.
- Répéter Jusqu'à satisfaction de la condition de convergence
 - Rechercher, le long de chaque normale de la forme précédemment élaborée, le profil correspondant au mieux à celui calculé pour la forme moyenne, les nouveaux points de marquage sont les points centraux des profils ainsi trouvés.
 - Rechercher le modèle de forme s'adaptant le mieux aux points trouvés à l'étape précédente et celui-ci constituera la forme de départ pour l'itération suivante.
- Génération de la forme finale du contour

Figure 4.7 – Algorithme du Modèle Actif de forme « Actif Shape Model »

Les profils sont des vecteurs calculés à partir de la texture environnante le long de la normale au contour en chaque point de référence de la forme moyenne, la comparaison entre les profils est réalisée à l'aide de la distance de Mahalanobis qui est basée sur la corrélation entre des variables par lesquelles différents modèles peuvent être identifiés et analysés, elle permet de déterminer la similarité entre deux jeux de données, la condition d'arrêt choisie consiste à laisser l'algorithme itérer jusqu'à ce que seulement un faible pourcentage de points de référence continue à se déplacer, dans le cas où la recherche continue indéfiniment, un nombre maximum d'itérations est fixé pour stopper la recherche et le dernier résultat est donné comme solution finale. Le principal inconvénient de la méthode des contours actifs concerne l'initialisation qui doit être très proche de l'objet recherché, la méthode ASM permet de marquer manuellement les quatre coins de la vertèbre illustré dans la *Figure 4.8* et les points intermédiaires seront marqués automatiquement, le contour initial ainsi formé sera proche du contour de la vertèbre dans le cas où cette dernière ne présente pas de zones concaves très prononcées.

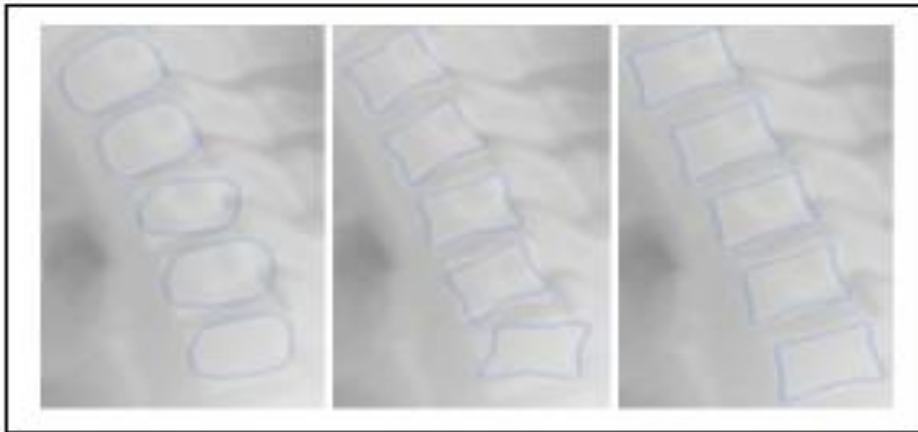


Figure 4.8– Détection de contour par l'ASM avec un marquage de 4, 8 puis 20 points

4.3.4 Les G-snake

Les G-snakes se base sur le modèle du snake tout en apportant des petites modifications, la plus importante est le changement de l'équation de base de l'énergie totale du snake en :

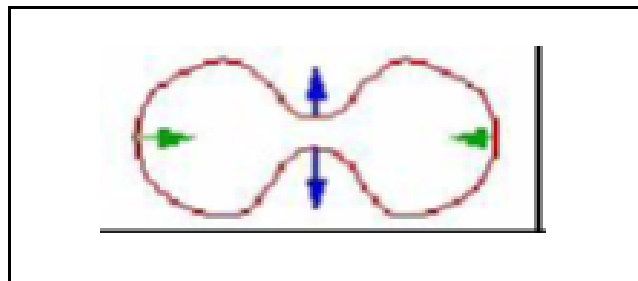
$$E = \arg \min \sum_{i=1}^n \lambda_i E_{int}(v_i) + (1 - \lambda_i) E_{ext}(v_i) \quad (4.16)$$

Où λ est le paramètre de régularisation et est compris entre 0 et 1.

Mettre un $\lambda \gg (1 - \lambda)$ augmente la régularisation et rend le modèle plus robuste au bruit de l'image, les petites valeurs de λ permettent au snake de mieux capturer les discontinuités, une autre modification concerne le changement de support sur lequel évolue le snake, en général, l'énergie est calculée sur la gaussienne et non par sur l'image originale, le G-snake, évolue sur deux images en même temps, la gaussienne ainsi que la carte de direction des contours, cette méthode a été essentiellement consacrée pour la détection de la silhouette dans une séquence vidéo.

4.3.5 Modèles géométriques

La nature des modèles déformables géométriques et géodésiques [PETITJEAN,2000] diffère radicalement de l'approche variationnelle des contours actifs, ainsi, l'approche par équations aux dérivées partielles consiste à faire évoluer un contour fermé orthogonalement à lui-même et avec une vitesse dépendant de sa courbure locale et d'informations externes. Cette méthode met en œuvre une courbe se déformant dans le temps et l'espace pour atteindre les frontières d'un objet à détecter dans un image $I(x,y)$, la courbe se déforme comme indiqué dans la *Figure 4.9*, selon sa normale, à une vitesse proportionnelle à sa courbure.



Figure– 4.9 Courbe se déformant selon sa courbure

Pour faire converger la courbe C vers les frontières d'objets [He & al, 2008], la vitesse dépend également d'une fonction G qui permettra l'arrêt sur ces frontières « *Figure 4.10* ». G est une fonction décroissante du gradient de l'image tendant vers 0 en l'infini et est souvent définie par :

$$g(I) = \frac{1}{1 + |\nabla I|^p} \quad (4.17)$$

I étant l'image régularisée par un opérateur gaussien et $p = 1$ ou 2 , les valeurs de $g(I)$ sont proches de 0 dans les régions où le gradient est élevé et proche de 1 dans les régions

d'intensité relativement constante, d'autre part, on introduit une constante positive v chargé de forcer l'évolution du contour vers les frontières de l'objet, on peut alors écrire :

$$\frac{\partial C}{\partial t} = g(I)(v+k)N^{\rho} \quad (4.18)$$

Où N étant la normale intérieure à la courbe k .



Figure 4.10 – Exemple d'évolution du contour actif Géométrique

4.3.6 Méthode d'ensembles de niveau zéro (Level Sets)

La méthode des Ensembles de Niveau Zéro [Cremers & al, 2007] est une méthode de simulation numérique utilisée pour l'évolution des courbes et des surfaces dans les domaines discrets, cette méthode permet de faire évoluer une courbe paramétrique fermée $C(p)$ suivant une équation du type :

$$\frac{\partial C}{\partial t} = FN \quad (4.19)$$

Où t est le temps, F est une fonction scalaire de vitesse (dépendant des propriétés locales de la courbe, les paramètres externes liés au gradient d'image et les termes additionnels de propagation de la courbe) et N est la normale unitaire à la courbe.

Chaque point de la courbe C évolue suivant la direction normale à la courbe avec une vitesse F . Ici, le front 2D est considéré comme le niveau $\psi = 0$ d'une fonction 3D ψ , l'idée de ces courbes de niveaux (Level set) est alors de déduire la propagation de $C(t)$ à partir de la

propagation de ψ , la courbe $C(t)$ peut alors changer de topologie sans problème de paramétrage contrairement aux cas des contours actifs classiques cela implique donc une paramétrage intrinsèque à la courbe et C sera l'intersection de $\psi(t)$ avec le plan défini par $\psi = 0$, la forme de cette intersection pouvant être quelconque. La fonction ψ est initialement la distance signée au contour C et est définie ainsi :

$$\psi(x, t = 0) = \pm d(x, C(t = 0)) \quad (4.20)$$

Où x est un point de R^2 , appartenant au plan de C

Le front $C(t)$ est récupéré par la détection de la courbe de niveau zéro de ψ on aura donc :

$$\frac{\partial \psi}{\partial t} = g(I)(v+k)|\nabla \psi| \quad (4.21)$$

Sachant que la courbe k s'écrit sous la forme de :

$$k = \frac{\nabla \psi}{|\nabla \psi|} \quad (4.22)$$

On aura alors :

$$E = \arg \min \sum_{i=1}^n \lambda_i E_{int}(v_i) + (1 - \lambda_i) E_{ext}(v_i) \quad (4.23)$$

L'avantage principal de cette équation est la possibilité de gérer automatiquement le changement de topologie de la courbe en évolution, la courbe C peut être divisée en deux ou trois courbes, inversement plusieurs courbes peuvent fusionner et devenir une seule courbe, ce que ne peuvent faire les snakes classiques, un autre point positif est que cette méthode ne dépend pas de l'initialisation, la *Figure 4.11* illustre un exemple d'utilisation de Level Set sur une image médicale (crâne).

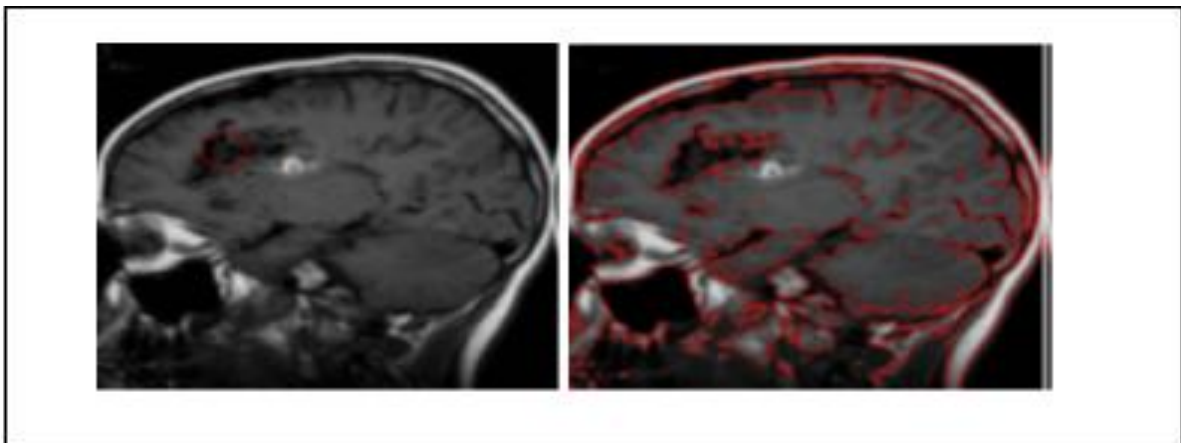


Figure – 4.11 Segmentation d'image par la méthode d'ensembles de niveau zéro

4.3.7 Modèles géodésiques

Les contours actifs géodésiques [Thomas & al, 2004] sont utilisés pour la détection de contours ou de surfaces, mais servent également à la détection de mouvement, la segmentation en texture et dans plusieurs domaines tels que l'imagerie médicale ou satellitaire. Dans la méthode Level Set la courbe s'arrête quand $g = 0$, ce qui correspond à un contour (au sens frontière) idéal, or, au cas où il y a différentes valeurs de gradients le long du contour, ce qui arrive souvent dans les images réelles, g prend différentes valeurs selon différents points sur la frontière, il faut donc réduire les valeurs de g possibles (à la frontière) pour stopper l'évolution de la courbe, le nouveau terme est $\nabla g(I) \cdot \nabla \psi$ et permet de résoudre les problèmes précédents tout en éliminant le choix d'une constante, il fournit une attraction plus grande ; en pratique, le géodésique ne s'arrête pas aux ombres (faux contours) on aura alors :

$$\frac{\partial \psi}{\partial t} = g(I) \left(v + \frac{\nabla \psi}{|\nabla \psi|} \right) |\nabla \psi| + (\nabla g(I) \cdot \nabla \psi) \quad (4.24)$$

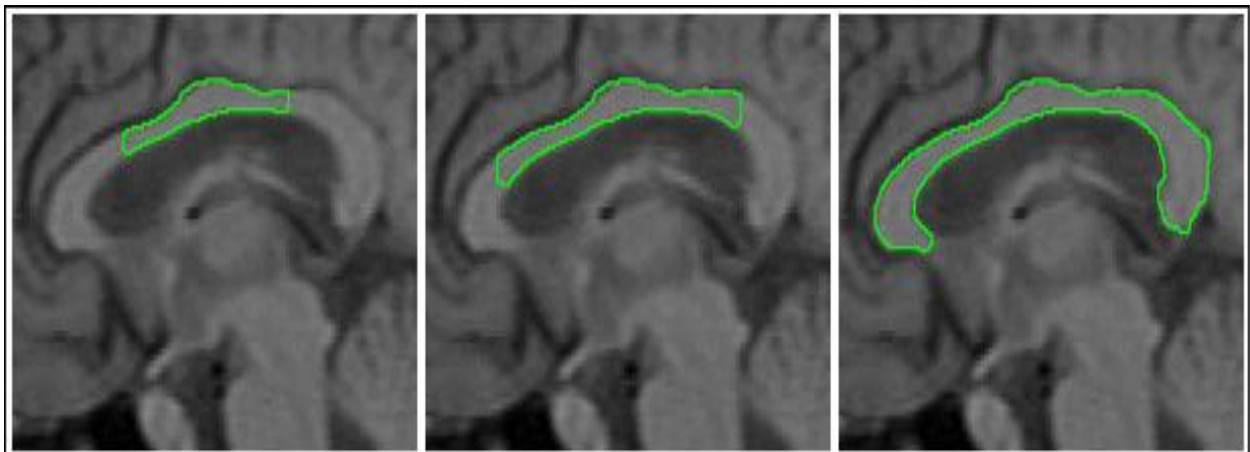


Figure 4.12 – Exemple d'évolution du contour actif Géodésique

Les contours actifs géodésiques offrent des avantages tel la détection simultanée des frontières extérieures et intérieures de plusieurs objets, le contour final ne dépend pas de l'initialisation, il existe toujours une solution unique et stable. Mais les contours actifs géodésiques présentent des inconvénients, le processus est sensible au bruit et l'image devra donc être filtrée au préalable, la vitesse de convergence dépend beaucoup des paramètres, La qualité de la convergence dépend de la fonction d'arrêt.

4.3.8 Le contour actif Dynamique

Le modèle du snake tel qu'il est décrit jusqu'ici peut être utilisé aussi bien pour localiser précisément un contour dans une image fixe [Niethammer & al, 2006] que pour suivre un mouvement non rigide du contour, pour trouver la position de la cible du suivi dans une nouvelle image, le snake est initialisé sur sa position dans l'image précédente et le processus de minimisation est relancé, cependant, si entre les deux images le déplacement de la cible est trop grand, le snake la perd et se stabilise sur un minimum local de l'image la plus proche, c'est pourquoi le modèle de base des snakes a été adapté à l'objectif particulier du suivi d'objet, est d'associer une masse au snake afin que son énergie interne intègre une énergie dynamique minimale lorsque le snake se déplace en ligne droite à vitesse constante, mathématiquement, le temps est introduit comme paramètre de la configuration géométrique du snake qui est alors représenté par $V(s,t)$. La masse le long du snake est modélisée par la fonction $\mu(s)$ et l'énergie dynamique peut être définie par :

$$E_{\text{Dynamique}}(v(s,t)) = \frac{1}{2} \mu(s) \left| \frac{\partial^2 v(s,t)}{\partial t^2} \right|^2 \quad (4.25)$$

L'énergie dynamique est égale à zéro lorsque la variation de vitesse (l'accélération) est nulle, ce terme permet de suivre des contours plus rapidement à condition que ceux-ci ne soient pas sujets à décélérations trop brusques ou que la fréquence d'échantillonnage de l'image soit assez élevée, l'énergie dynamique est minimale lorsque le snake est exactement sur la position prédite, l'introduction de ce terme a cependant l'inconvénient de faire osciller le snake autour de sa position de stabilisation, un frottement visqueux est ajouté au modèle afin de dissiper l'énergie cinétique lorsque le snake est proche de sa position d'équilibre. L'énergie due au frottement se traduit par :

$$E_{\text{Frottement}}(v(s,t)) = \frac{1}{2} \gamma(s) \left| \frac{\partial v(s,t)}{\partial t} \right|^2 \quad (4.26)$$

Où γ est le facteur de viscosité le long du snake (Résistance).

Le comportement du snake est toujours simulé par minimisation de son énergie globale qui prend la forme suivante :

$$E_{\text{Globale}}(v,t) = \int_0^1 (E_{\text{Frottement}}(v(s,t)) + E_{\text{Dynamique}}(v(s,t)) + E_{\text{Interne}}(v(s,t)) + E_{\text{Externe}}(v(s,t))) ds \quad (4.27)$$

Les résultats sont résumés dans la *Figure 4.13* on indique à chaque fois sur l'image d'origine le contour initial (en vert) et le contour final (en rouge).

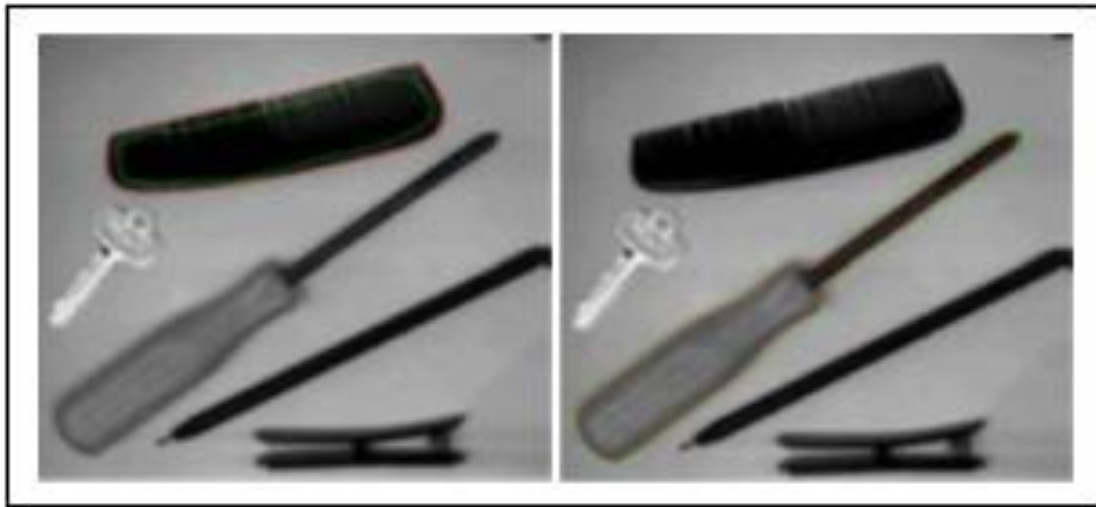


Figure – 4.13 Exemple d'application du contour actif dynamique

4.3.9 Contours actifs par approche région

Les contours actifs définis par les approches régions [Lankton & al, 2008] s'intéressent à l'intérieur et à l'extérieur de la région, les approches régions peuvent être divisées en deux classes, dans la première les paramètres statistiques de l'objet et du fond sont estimés localement autour du contour, alors que dans la deuxième, toute l'image est prise en compte. Le contour d'un objet est défini comme l'ensemble de points de l'objet ayant des voisins n'appartenant pas à l'objet, il délimite donc, deux régions différentes, l'idée consiste à appliquer à chaque point du contour dont le voisinage appartient au fond une force dirigée vers l'intérieur du contour, et inversement, chaque point du contour possédant un voisinage appartenant à l'objet sera soumis à une force en direction de l'extérieur, toutes ces forces sont définies avec une direction normale au contour, l'intensité de pixels d'une région k est approchée par une combinaison linéaire des fonctions $K_i(x,y)$ et l'énergie de la région k est définie comme suit :

$$W(\text{Région } k) = \iint |I(x,y) - m_k| * |I(x,y) - m_k| dx dy \quad (4.28)$$

Où $I(x,y)$ est l'intensité des pixels et m_k une intensité moyenne.

Cette définition de l'énergie permet de définir celle associée au contour, elle est choisie comme étant la Ward-distance entre les deux régions $R(in)$ et $R(out)$, elle permet de mesurer le degré de similitude entre deux régions, l'énergie sera alors :

$$W(\text{contour}) = D[R(\text{in}), R(\text{out})] = W[R(\text{in}) + R(\text{out})] - W(R(\text{in})) - W(R(\text{out})) \quad (4.29)$$

La *Figure 4.14* illustre l'utilisation de cette méthode, une image avec un fond gris a été prise afin d'éviter les contours parasites, contenant une pince à détecter, afin de détecter cette forme, une forme initiale (un rectangle ici) au voisinage des contours a été placée, nous observons ci-dessus le résultat obtenu de la méthode des contours actifs à la fin des itérations successives nécessaires.

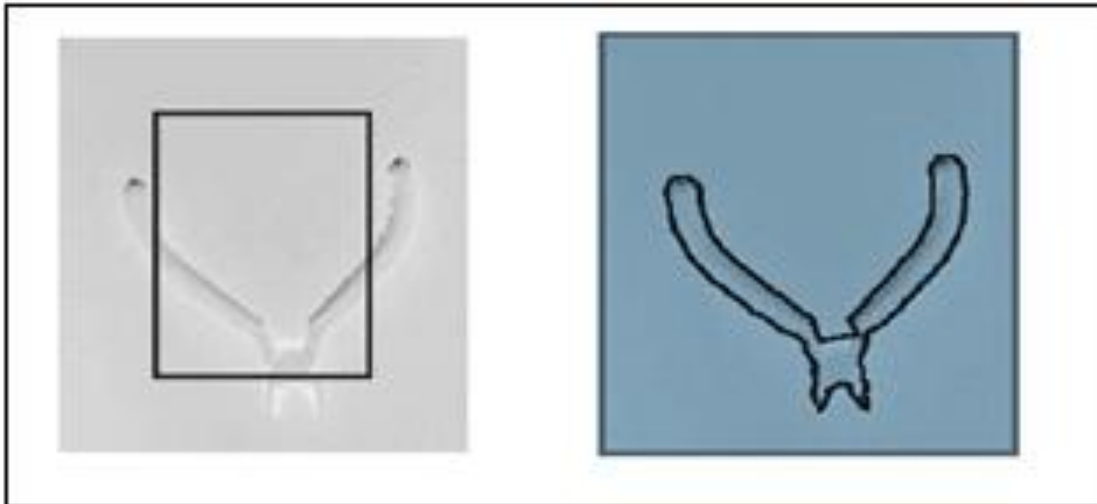


Figure 4.14 – Détection de la pince avec un contour actif basé région

4.3.10 Modèle particulière

Dans ce modèle récent et particulier, Angella [Angella 2001] considère les snaxels comme des entités indépendantes, appelées particules, virtuellement liées entre elles par des contraintes de voisinage imposant la topologie du système. Les contours actifs peuvent être fermés ou ouverts à extrémités fixes ou non. Il introduit la notion de « contour actif étendu » pour un contour actif à topologie complexe. Il fixe la topologie en imposant des contraintes de régularisation entre snaxels dans la matrice de lissage. En supposant que le contour actif étendu ne puisse pas se décomposer en plusieurs contours actifs indépendants, il le qualifie de connexe. Ce contour connexe se caractérise par la présence de snaxels ayant plus de deux voisins. Les coefficients α et β sont globaux et communs à toutes les branches qui composent le contour. La matrice de régularisation maintient la cohésion interne du modèle mais ne représente aucun lien explicite entre les points d'échantillonnage. L'initialisation, comme dans le modèle classique, doit s'effectuer à proximité du résultat cherché. Le modèle particulière est une prolongation naturelle du contour actif étendu puisqu'il modélise les forces d'interactions. Un système particulière se définit comme un ensemble fini S dont les éléments

sont des particules notées P_i avec N le nombre d'éléments de l'auteur [Angella 2001] parle parfois de nœud par analogie aux graphes ou d'agent lorsque l'accent est porté sur leur caractère comportemental. Les propriétés de P_i peuvent varier avec le temps. A chaque P_i , il associe une masse m constante, un vecteur de position $x(t)$ et une vectrice vitesse $v(t)$. Il note $f(t)$ la somme des forces appliquées à l'instant t sur P_i . La particule est donc caractérisée par :

$$P(t) = \{m, x(t), v(t), f(t)\} \quad (4.30)$$

Un tel système trouve son domaine d'application dans la recherche d'arborescences dans des images. Par exemple, des vaisseaux dans l'imagerie médicale, la squelettisation de caractères manuscrits ou d'imprimerie, enfin la recherche de réseaux routiers en imagerie aérienne.

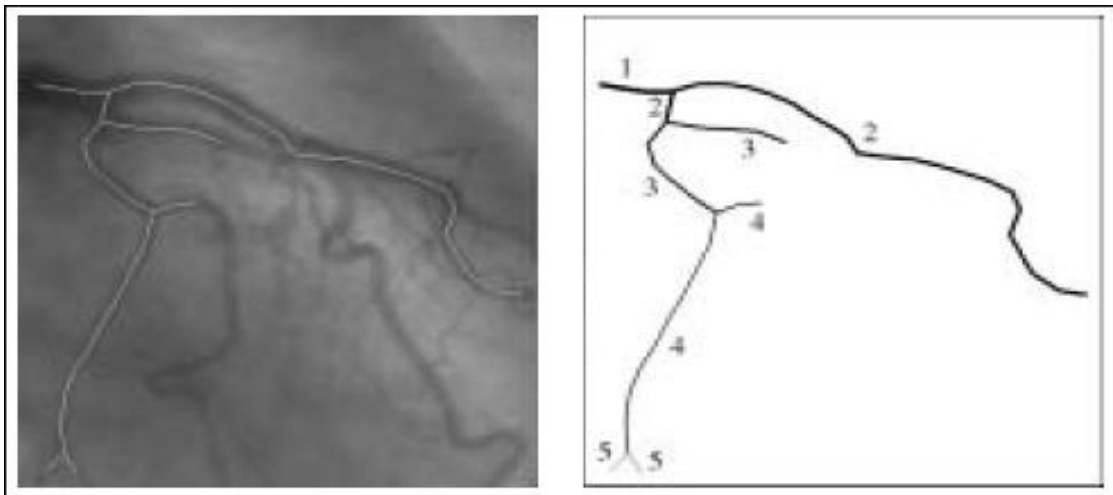


Figure 4.15 - Exemple d'application du modèle particulaire

4.4 Conclusion

Nous avons présenté les différentes approches pour définir un critère de segmentation pour les contours actifs, de l'approche purement "contours" aux approches plus récentes qui intègrent des termes de régions dans les fonctionnelles à minimiser. Les approches contour ne prenant en compte que de l'information locale sont assez sensibles au bruit et lors de la minimisation de la fonctionnelle, il n'est pas rare de tomber dans un minimum local. Quant aux approches régions, la plupart des méthodes sont efficaces dans des cas simples où l'objet à segmenter est facilement séparable du fond, de par sa couleur par exemple, mais la segmentation est plus difficile lorsque les objets à segmenter ne respectent pas les hypothèses faites sur leurs

distributions. Cependant, les contours actifs apparaissent comme un outil pertinent pour parvenir à implémenter le problème de segmentation d'image de profondeur, puisqu'ils permettent de faire évoluer un contour vers l'objet d'intérêt. De plus, il est possible d'ajouter un a priori géométrique sur l'objet à segmenter en imposant des contraintes de régularisation sur le contour. Dans le chapitre suivant, nous allons nous intéresser au mécanisme des systèmes multi-agent et présenter les systèmes plus référencés dans la littérature et leurs apports en segmentation d'images.

Chapitre 5

Les systèmes multi-agents

5.1 Introduction

La thématique des Systèmes Multi-Agents (SMA) est actuellement un champ de recherche très actif pour de nombreux types d'applications. Sa particularité provient de sa connexion avec plusieurs domaines de recherche comme l'Intelligence Artificielle Distribuée (IAD) et la Vie Artificielle (VA). C'est une discipline qui s'intéresse aux comportements collectifs produits par les interactions de plusieurs entités autonomes appelées agents. Dans ce chapitre, nous allons exposer l'essentiel des concepts utilisés dans le domaine du SMA. Tout d'abord, nous décrirons le principe général des SMA tout en exposant les différentes parties qui les composent en partant des notions d'agents, en passant par la définition d'un environnement jusqu'à l'exploration des caractéristiques d'un SMA. Puis nous présenterons les méta-heuristiques couramment utilisées et nous terminerons par un aperçu des différentes approches SMA appliquées en imagerie.

5.2 Définitions

Les systèmes multi-agent sont apparus au carrefour des recherches traitant l'IAD et la VA. L'une des sources d'inspiration pour ces systèmes est le comportement des sociétés d'animaux. Celles généralement étudiées sont les familles d'insectes telles que les abeilles, les fourmis ou les termites [Boissier et al, 2004]. D'autres sources d'inspiration se trouvent également dans la sociologie, la psychologie sociale, les sciences cognitives. Ces systèmes ont repris les modes de communication et de concertation (organisation, négociation, combinaison) entre les agents provenant de l'IAD et les idées d'autonomie et d'émergence de solutions à partir des interactions individuelles de la VA. D'une part, les SMA se placent au sein des sciences cognitives, des sciences sociales et des sciences naturelles pour modéliser, expliquer et simuler des phénomènes naturels, et susciter des modèles d'auto-organisation. D'autre part, ils se présentent comme une pratique et une technique dont l'objectif est de réaliser des systèmes complexes à partir du concept Agent et de leurs interactions par l'intermédiaire des processus de communication, de coopération et de coordination d'actions.

Le principe d'un modèle IAD réside dans le fait de découper le problème à résoudre en une multitude de sous-problèmes qui vont être à leur tour résolus par des agents afin de fournir des solutions partielles. Ces solutions partielles vont ensuite être regroupées pour construire la solution globale du problème. De plus, les systèmes SMA définissent les agents comme des entités ayant leur propre autonomie et flexibilité. L'objectif de la modélisation d'un SMA est alors de déterminer comment les agents vont se coordonner pour échanger leurs connaissances, leurs buts et leurs stratégies pour agir et résoudre des problèmes complexes. Généralement, les types d'application des SMA sont décomposés en trois grandes classes.

- La modélisation de phénomènes du monde réel afin de les simuler pour mieux comprendre leurs comportements. Ceci est réalisé par l'observation, la compréhension et l'explication de leur comportement et de leur évolution. Ce sont par exemple, des applications de simulation de phénomènes sociaux, environnementaux, éthologiques, etc.
- La conception de programmes dans lesquels des agents qualifiés d'assistants jouent le rôle des êtres humains. La notion d'agent simplifie la conception de ces programmes et amène de nouvelles problématiques centrées utilisateurs telles que la communication, la sécurité. On trouve un exemple de telles approches dans les systèmes de ventes aux enchères dans lesquels les agents jouent les rôles de commissaires priseurs et d'acheteurs.
- La résolution distribuée de problèmes il s'agit de concevoir des systèmes artificiels intégrant un ensemble d'agents et une dynamique organisationnelle permettant l'accomplissement collectif d'une tâche. L'hypothèse de base est que des agents, de par leurs comportements et leurs interactions locales, peuvent donner lieu, à un niveau global, à un comportement d'ensemble cohérent et fonctionnel. Ce sont par exemple les fourmis qui partent de leur fourmilière à la recherche de la nourriture en créant le plus court chemin.

À l'heure actuelle, la programmation de robots, de jeux ou encore la simulation de processus biologiques sont autant d'applications des SMA qui montrent que ce domaine est très actif aussi bien en recherche fondamentale qu'en développement. Nous allons maintenant décrire les concepts des catégories d'agents, de notion d'environnement, de modes d'interaction, d'auto-organisation et d'émergence.

5.3 Les entités d'un système multi-agent

À première vue, un SMA peut être simplement considéré comme un ensemble d'agents partageant un environnement commun. Toutefois, il est difficile de trouver une définition unique du concept d'Agent [Khosla & al, 2004], d'où il s'avère difficile de caractériser un SMA. Ferber a exposé quelques aspects fondamentaux qui permettent de réaliser cette caractérisation à partir des systèmes rencontrés dans la nature:

- Un environnement E, disposant en général d'une métrique
- Un ensemble d'objets O, auxquels on peut associer une position dans E à un moment donné. Ces objets sont passifs : les agents peuvent les percevoir, les créer, les détruire et les modifier
- Un ensemble d'agents A, lesquels représentent les entités actives du système.
- Un ensemble de relations R, qui unissent les objets entre eux ainsi que les agents;
- Un ensemble d'opérateurs Op permettant aux agents de A de percevoir, produire, consommer, transformer et manipuler des objets de O ;
- Des opérateurs chargés de représenter l'application de ces opérations et la réaction du monde à cette tentative de modification, que l'on appellera les lois de l'univers.

D'après cette description, nous pouvons définir un SMA comme un système qui s'intéresse aux comportements collectifs produits par les interactions de plusieurs agents autonomes et flexibles et que ces interactions sont définies comme la coopération et la compétition entre ces agents qui peuvent être hétérogènes « *Figure 5.1* ». Dans ce qui suit, nous allons définir la notion d'agent ainsi que les catégories d'agents pouvant être représentées dans un SMA.

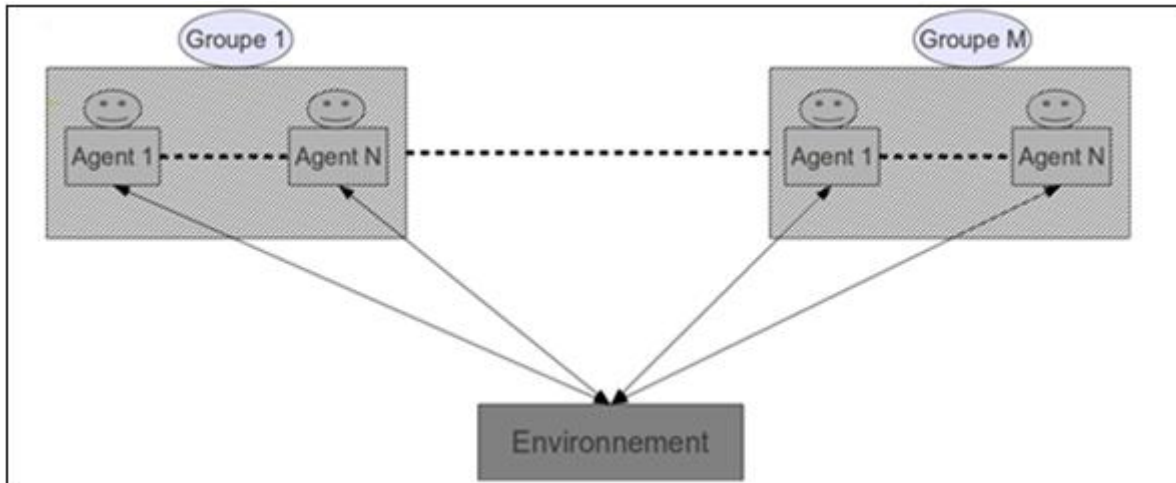


Figure 5.1 Relation unissant l'environnement et l'ensemble des agents.

5.4 Définition d'un agent

Dans la littérature, nous trouvons plusieurs définitions du concept d'Agent. Elles présentent certaines similitudes et dépendent du type d'application pour laquelle est conçu l'agent. Les définitions respectives de Ferber et de Jennings [*Jennings & al, 1998*]. Sont généralement les plus communément utilisées au sein de la communauté Multi-Agents.

- **Définition 1 :** Un agent est une entité autonome, réelle ou abstraite, qui est capable d'agir sur elle-même et sur son environnement, qui dans un univers Multi-Agents, peut communiquer avec d'autres agents, et dont le comportement est la conséquence de ses observations, de ses connaissances et des interactions avec les autres agents.
- **Définition 2 :** Un agent est un système informatique, situé dans un environnement, et qui agit d'une façon autonome et flexible pour atteindre les objectifs pour lesquels il a été conçu.

Les définitions exposées ci-dessus présentent plusieurs propriétés clés tels que l'autonomie, l'action, la communication et la flexibilité. L'autonomie est un élément du comportement de l'agent qui est fortement liée aux notions d'apprentissage et d'adaptation. Si ceci est déterminé a priori, l'agent manque alors d'autonomie. Les agents sont considérés comme autonomes s'ils ne sont pas guidés par des commandes venant de l'utilisateur ou d'un autre agent, mais par un ensemble de tendances qui peuvent prendre la forme de buts individuels à

satisfaire ou de fonctions de satisfaction ou de survie que l'agent cherche à optimiser. Maes définit les agents autonomes comme suit :

- **Définition 3 :** Les agents autonomes sont des systèmes de calcul qui se trouvent dans un environnement complexe et dynamique, perçoivent et agissent d'une manière autonome dans cet environnement, et ce faisant réalisent les objectifs et les tâches pour lesquelles ils ont été conçus.

L'action est un concept fondamental pour les SMA reposant sur le fait que les agents accomplissent des actions qui vont modifier leur environnement et donc leurs prises de décisions futures. Les interactions entre agents dépendent du mode de communication. Une communication entre 2 agents est considérée comme une action pour celui qui la transmet et comme une perception pour celui qui la reçoit.

5.5 Les types d'agents

Nous distinguons deux grandes familles d'agents [Georgeff & al, 1992] les agents réactifs et les agents cognitifs. Cette distinction tient essentiellement au processus décisionnel chez l'agent et à la représentation de l'environnement dont il dispose. Si l'agent est doté d'une représentation symbolique de l'environnement à partir duquel il est capable de formuler des raisonnements, nous disons qu'il est cognitif tandis que s'il ne dispose que d'une représentation limitée à ses perceptions, nous disons qu'il est réactif. Qu'il soit réactif ou cognitif, un agent évolue toujours selon un cycle en trois étapes :

- Perception : ses capteurs lui fournissent une vision locale de son environnement.
- Décision : suivant ses intentions, son état interne et sa perception, l'agent choisit une action à effectuer.
- Action : il modifie l'environnement par son action.

5.5.1 Les agents cognitifs

Les agents cognitifs [Parunak & al, 1997] disposent d'une base de connaissances comprenant l'ensemble des informations disponibles et un raisonnement nécessaire à la réalisation de leur tâche ainsi qu'à la gestion des interactions avec les autres agents et avec l'environnement. De plus, ils possèdent des buts et des plans pour décider de leurs actions. Les groupes d'agents

peuvent coordonner leur activité et négocier entre eux pour résoudre leurs conflits. Ces comportements peuvent être assimilés à des comportements sociaux et les recherches dans ce domaine s'appuient sur les travaux de sociologie des organisations des groupes d'individus. L'une des architectures cognitives les plus connues est l'architecture BDI : Belief (Croyance), Desire (Désir), Intention (Intention). Les croyances correspondent aux informations dont dispose l'agent sur son environnement. Les désirs correspondent aux états de l'environnement que l'agent souhaiterait voir réalisés. Les intentions correspondent aux projets de l'agent pour satisfaire ses désirs.

5.5.2 Les agents réactifs

Contrairement aux agents cognitifs, les agents réactifs ne sont pas nécessairement assez intelligents individuellement pour que le système ait un comportement global que l'on puisse qualifier d'intelligent. Des mécanismes de réactions aux événements et aux différentes perceptions, ne tenant pas compte ni d'une explication des buts, ni des mécanismes de planification, peut alors découler la résolution de problèmes complexes: le comportement complexe du système émerge alors de la coexistence et de la coopération des comportements simples de chaque agent « *Figure 5.2* ».

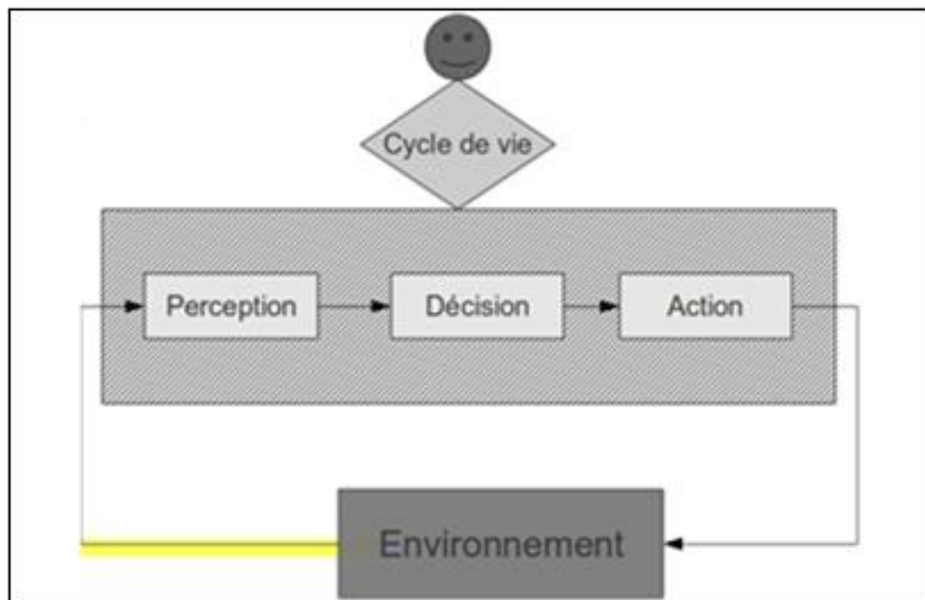


Figure 5.2 Principe de la réaction d'un agent réactif sur un environnement à partir de son cycle de vie.

L'organisation des fourmilières ou des termitières est généralement citée en exemple de tels systèmes qualifiés d'émergents. Dans ce cas, les buts sont la recherche de la nourriture, les soins apportés aux œufs ou encore la construction des nids. Les SMA réactifs présentent des intérêts en termes de fiabilité assurée par le grand nombre d'agents du système et leur simplicité ainsi que le passage à l'échelle par l'adaptation à l'augmentation du nombre d'agents.

5.5.3 Les agents hybrides

Les agents hybrides sont conçus pour combiner des capacités réactives à des capacités cognitives, ce qui leur permet d'adapter leur comportement en temps réel à l'évolution de l'environnement. Dans le modèle hybride, un agent est composé de plusieurs couches, rangées selon une hiérarchie en trois couches. La couche purement réactive, la couche intermédiaire et la couche sociale. Dans ces trois couches, on va de la simple capture d'informations à partir de senseurs jusqu'à la gestion de la coopération entre agents en passant par la construction d'abstraction à partir des connaissances.

5.6 Définition de l'environnement

L'environnement [Weyns & al, 2007] est l'endroit d'immersion dans lequel évoluent les agents. On distingue le plus souvent deux sortes : l'environnement social qui est composé des autres agents du système et l'environnement physique au sein duquel les agents évoluent. Un agent peut se placer dans l'un et/ou dans l'autre. L'environnement physique dépend largement de l'application traitée alors que l'environnement social dépend souvent des choix de conception des SMA. L'environnement est une première classe d'abstraction fournissant les conditions environnementales aux agents pour exister et sert d'intermédiaire pour les interactions entre les agents ainsi que pour l'accès aux ressources. Les propriétés essentielles de l'environnement sont :

- **Accessible/inaccessible** : le système peut obtenir une information complète, exacte et à jour sur l'état de son environnement. Dans un environnement inaccessible, une information partielle est uniquement disponible.

- **Déterministe/non déterministe** : une action a un effet unique et certain. Si le système agit dans son environnement, il n'y a aucune incertitude sur l'effet de son action sur l'état de l'environnement. L'état suivant de l'environnement est complètement déterminé par l'état courant. Dans un environnement non déterministe, une action n'a pas un effet unique garanti.
- **Dynamique/statique** : l'état de l'environnement dynamique dépend des actions du système qui se trouve dans cet environnement, mais aussi des actions d'autres processus. Ainsi, les changements ne peuvent pas être prédits par le système. Un environnement statique ne peut changer sans que le système agisse.
- **Continu/discret** : le nombre d'actions et de perceptions possibles dans cet environnement est infini et indénombrable. Dans le cas discret, l'environnement est défini par un nombre d'états fini et selon un découpage régulier tel qu'une grille. Par exemple, dans le jeu de la poursuite (qualifié comme un problème de proies et de prédateurs) dont le but est de parvenir à ce que les prédateurs capturent les proies en les entourant, les agents proies et prédateurs se déplacent dans une grille et ont chacun des coordonnées/positions à l'intérieur de chaque case).

L'environnement est défini à la fois comme lieu d'inscription permettant de partager le résultat des activités de chaque agent et comme ensemble de contraintes sur la dynamique des SMA. L'environnement peut être très élémentaire et se réduire à l'ensemble des canaux de communication entre les agents. Toutefois, dans les SMA à base d'inspiration biologique ou physique, l'environnement qui possède généralement une métrique, est plus présent et a un rôle primordial.

5.7 Interactions entre agents

L'interaction entre agents dans un SMA [Bourjot & al, 2003] peut être définie comme la mise en relation dynamique de deux ou plusieurs agents par le biais d'un ensemble d'actions. Ces interactions prennent deux formes, directes et indirectes. Les interactions directes sont en général des caractéristiques des agents cognitifs qui, travaillant pour un but précis, sont capables de communiquer intentionnellement. Ce type d'interaction se rapproche d'un acte de langage. Tandis que les interactions indirectes sont les caractéristiques des agents réactifs qui

répondent à des stimuli de l'environnement par le dépôt d'information relative à des critères environnementaux. Dans le cas des agents réactifs, Les interactions entre les agents sont modélisées comme suit :

- **Coopération** : Travaillé ensemble dans un but commun, ceci est fait au travers de l'environnement sans aucune communication directe entre les agents.
- **Coordination** : Organiser la solution d'un problème de telle sorte que les interactions nuisibles soient évitées ou que les interactions bénéfiques soient exploitées.

5.8 Comportement du system

Dans les SMA, on parle souvent de phénomènes émergents ou d'auto-organisation des agents. L'émergence et l'auto-organisation soulignent des caractéristiques différentes du comportement d'un système. Les deux phénomènes [WOLF & al, 2004] peuvent exister séparément, comme ils peuvent coexister dans un système dynamique. Le but de cette section est la différenciation des caractéristiques importantes des deux concepts au travers d'une vue d'ensemble de l'historique de l'utilisation de chaque concept.

5.8.1 Le concept d'émergence

Un système exhibe de l'émergence quand il y a des émergents cohérents au niveau macro qui apparaissent dynamiquement à partir des interactions entre les parties au niveau micro. De tels émergents sont nouveaux par rapport aux parties du système prises individuellement en compte. Dans cette définition, les émergents désignent ici le résultat du processus d'émergence : propriétés, comportements, structures, modèles, etc. Le niveau macro considère le système dans son intégralité tandis que le niveau micro considère le système du point de vue des différentes entités qui le composent. J. Odell a identifié comme éléments essentiels des phénomènes émergents [Odell & al, 2007], le contrôle décentralisé, le lien bidirectionnel entre les deux niveaux micro et macro « *Figure 5.3* », la robustesse du système et la flexibilité dues à l'insensibilité aux perturbations locales et le fait que le comportement global soit nouveau par rapport aux comportements individuels du niveau macroscopique.

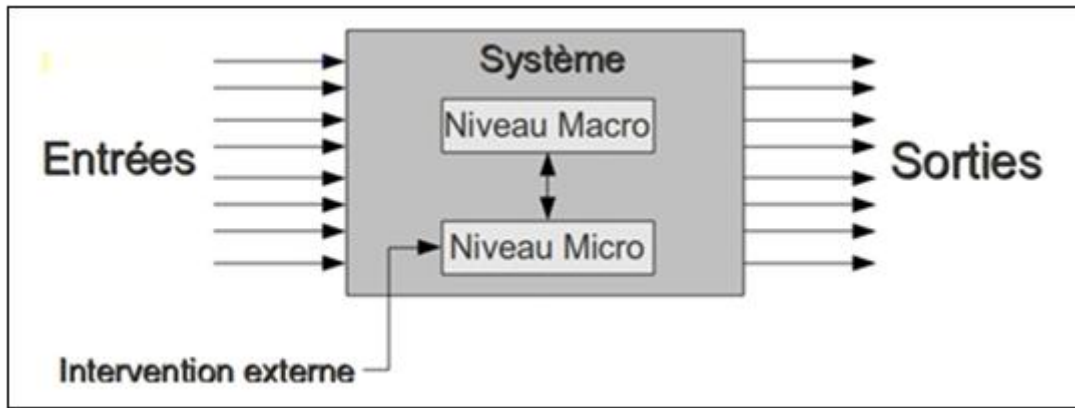


Figure 5.3 Le concept d'émergence.

De nombreux cas de systèmes émergents ont été étudiés en biologie et en écologie. En écologie, de nombreux systèmes naturels composés d'individus autonomes exhibent des aptitudes à effectuer des tâches qualifiées de complexes sans contrôle global. C'est le cas des colonies d'insectes sociaux tels que les termites ou les fourmis. En biologie, le système immunitaire est représentatif du fonctionnement d'un système complexe composé d'un ensemble d'agents autonomes. L'immunité du système étant le phénomène émergent obtenu.

5.8.2 L'auto-organisation

Le concept d'auto-organisation « *Figure 5.4* », est utilisé depuis les dernières décennies en biologie pour expliquer la diversité organisée des motifs chez les vivants, par exemple les rayures et d'autres motifs chez les animaux ou les structures géométriques de coquillages. Toutefois, c'est un cas particulier d'auto-organisation qui ne fait pas apparaître tout l'aspect dynamique de l'auto-organisation [Heylighen & al, 1999]. En effet, l'auto-organisation ne tend pas forcément vers un état d'équilibre stable. A l'heure actuelle, l'idée d'auto-organisation est maintenant employée dans beaucoup de disciplines et d'applications telles que les travaux liés à la simulation, les réseaux et la robotique.

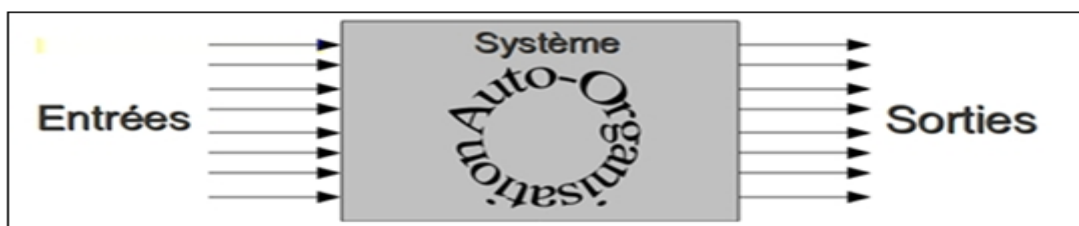


Figure 5.4 Illustration du concept d'auto-organisation.

L'auto-organisation est un processus dynamique et adaptatif où les systèmes acquièrent et maintiennent une structure sans contrôle externe. Les propriétés les plus importantes de l'auto-organisation sont :

- **Mise en ordre croissante** : On peut définir l'organisation comme une augmentation de l'ordre dans le comportement d'un système qui permet à ce dernier d'acquérir une structure spatiale, temporelle ou fonctionnelle. Cependant, il faut signaler que tout système exhibant une mise en ordre croissante ne peut être dit auto-organisé que s'il est autonome.
- **Autonomie** : Toute augmentation d'ordre ne peut être appelée auto-organisation. La deuxième caractéristique importante de l'auto-organisation est l'absence de contrôle externe, d'où provient le terme "auto". Un système doit alors s'organiser sans interférence avec l'extérieur.
- **Adaptation et robustesse face aux changements** : Dans les systèmes auto-organisés, le terme robustesse est employé dans le sens de l'adaptabilité aux perturbations et aux changements. Un système auto-organisé doit pouvoir faire face à ces changements et maintenir son organisation de manière autonome.
- **Dynamique** : Les changements et les perturbations affectent les structures organisées. Pour répondre à ces changements, une structure auto-organisée devra savoir s'adapter. On parlera alors d'organisation dynamique. Les SMA sont actuellement un des outils les plus utilisés pour modéliser les systèmes auto-organisés.

5.8.3 Le concept de Stigmergie

On considère généralement que la combinaison de l'émergence et de l'auto-organisation constitue une approche prometteuse pour proposer des solutions aux problèmes complexes et fortement dynamiques « *Figure 5.5* ». Cette combinaison a suscité notre intérêt pour traiter la segmentation d'images, activité qualifiée de complexe et pour laquelle aucune méthodologie générique n'a pas encore été élaborée. La combinaison des concepts entre l'émergence et l'auto-organisation a été définie par Grassé [Rupert & al, 2008]. Il définit le terme de stigmergie qui provient des mots grecs *stigma* "marque, signe" et *ergon* "travail, action", exprimant la notion que les actions d'un agent laissent des signes dans l'environnement, signes perçus par lui-même et par les autres agents et qui déterminent leurs prochaines actions. Grassé a étudié le comportement des termites et a décrit les modifications de l'état de

l'environnement par l'intermédiaire des gradients de phéromones déposés, les termites arrivent, sans régulation centrale, à communiquer et à coordonner leur action. La structuration de l'environnement a un effet sur le comportement des agents qui agissent à leur tour sur la structure de l'environnement, et ainsi de suite [Denis & al, 2003]. Nous assistons alors à un phénomène de couplage entre le système et son environnement, générant l'auto-organisation, à un niveau macro, de l'ensemble. Ce couplage permet au système de s'autoproduire comme un collectif auto-organisé. En d'autres termes, la stigmergie fournit un moyen de coordination entre les agents, émanant du système lui-même. C'est donc une méthode de communication indirecte dans un environnement émergent auto-organisé, où les individus communiquent entre eux en modifiant leur environnement. Plusieurs travaux concernent l'exploitation de l'environnement pour la résolution et l'optimisation de problèmes à travers des mécanismes stigmergiques, des phéromones, des toiles, des champs de forces, etc. Parmi les approches d'optimisation, nous citons l'algorithme d'optimisation par des colonies de fourmis qui miment la capacité de véritables fourmis à résoudre des problèmes complexes. Par exemple, partant de leur nid à la recherche de nourriture, les fourmis déposent des phéromones sur leur chemin afin de guider les autres vers un chemin qui finira par être le plus court chemin pour accéder à la nourriture. La perturbation des phéromones gérée par le phénomène de l'évaporation induit une auto-organisation des fourmis sur le chemin le plus utilisé avec une émergence de la solution optimale.

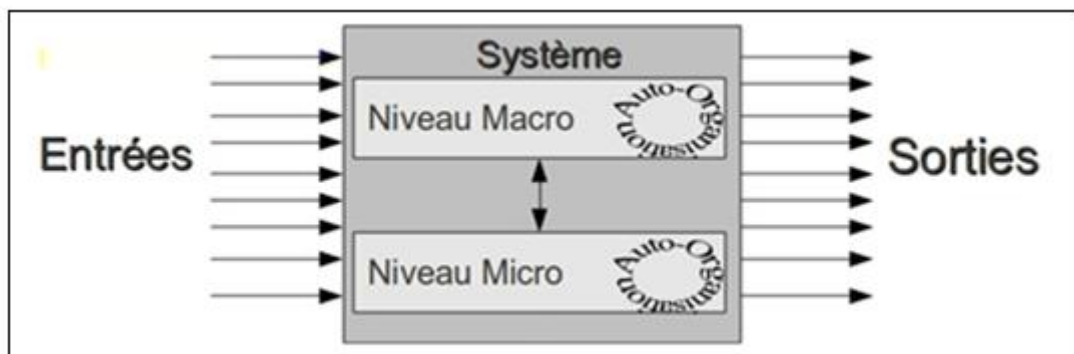


Figure 5.5 Schématisation du concept la stigmergie.

5.9 Segmentation d'image par utilisation des SMA

Dans ce qui suit, nous allons exposer brièvement quelques travaux utilisant les SMA dans le domaine du traitement d'images et plus particulièrement dans le but de segmenter ces images. Il est à noter que la plupart de ces travaux sont effectués dans le cas de la 2D.

5.9.1 Approches points

- Haroun [*Haroun & al, 2005*] à proposé une approche de segmentation des principaux tissus d'une image IRM du cerveau. Les auteurs ont présenté une approche coopérative mettant en œuvre deux algorithmes de segmentation d'image : la classification FCM pour gérer l'incertitude et l'imprécision, et la croissance de régions pour agir localement sur l'image. Un système Multi-Agents est introduit dans la phase de croissance de régions afin d'améliorer la qualité de la segmentation. Il existe deux types d'agents, un agent de type "Contrôleur" et plusieurs agents de type "Croissance" déployés sur l'image. Le processus est répété plusieurs fois jusqu'à ce qu'il ne reste aucun pixel susceptible d'être membre d'une région. Cette méthode utilise un processus de classification floue pour séparer les 3 principaux tissus du reste du cerveau afin de les segmenter « *Figure 5.6* ». Malgré le fait que le nombre de régions est petit, leur système n'arrive pas à détecter tous les pixels de l'image.

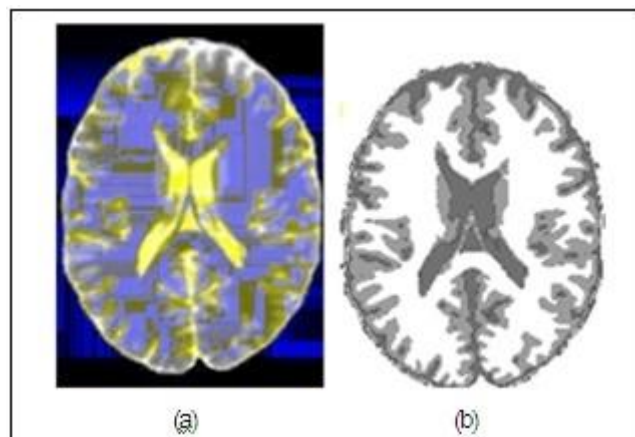


Figure 5.6 Segmentation extraite de Haroun image originale et (b) image segmentée

- Scherrer [*Scherrer & al, 2007*] a considéré la segmentation des tissus et des structures (la matière grise, la matière blanche et le liquide céphalo-rachidien) par les champs de Markov sur des images 3D de scanner comme deux procédures locales et coopératives immergées dans un cadre Multi-Agents. La segmentation des tissus est effectuée en partitionnant le volume en sous-volumes où les agents estiment les modèles locaux des champs de Markov en coopération avec leurs voisins pour assurer la cohérence des modèles locaux. Ces modèles reflètent mieux la distribution d'intensité locale. La segmentation des structures est réalisée via des agents dynamiquement localisés qui

intègrent des contraintes anatomiques et spatiales fournies par une description floue à priori de l'anatomie du cerveau « *Figure 5.7* ». Cette segmentation ne se réduit pas à une étape de post-traitement : les agents des structures coopèrent avec les agents des tissus pour rendre les modèles progressivement plus précis. Le processus s'arrête lorsque le système converge. L'inconvénient de cette approche est la taille de la fenêtre du voisinage qui diffère d'une image à l'autre pour estimer l'appartenance de chaque voxel à chaque classe de structure.

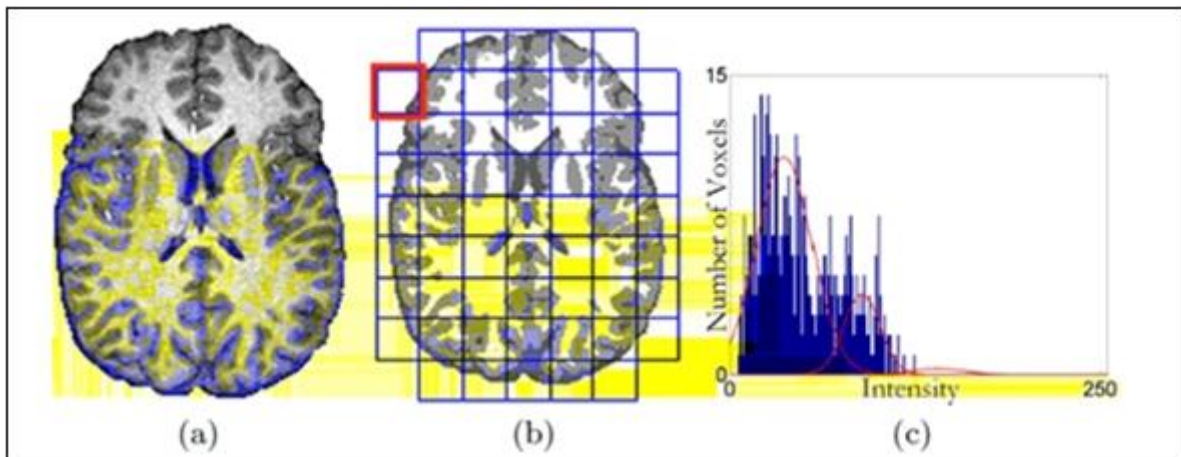


Figure 5.7 Segmentation de Scherrer (a) image originale, (b) image segmentée décomposée en blocs et (c) l'histogramme de la représentation des tissus dans le bloc rouge

5.9.2 Approches contours

- Pour segmenter des images 2D d'IRM du cerveau, Germond [Germond & al, 2000] a proposé une méthode de segmentation coopérative basée sur la segmentation en régions de la MG et de la MB. Cette segmentation est effectuée par des agents spécialisés à partir d'un modèle déformable du cerveau construit au préalable. À l'issue de la segmentation en régions de ces tissus, le contour du cerveau plus précis est obtenu par des agents contours qui utilisent la méthode des contours actifs « *Figure 5.8* ». A la fin de l'algorithme, un retour complet sur le processus de segmentation est considéré, initialisé par le contour reconstruit lors de la phase précédente. Ce processus est répété plusieurs fois jusqu'à satisfaction d'un critère d'arrêt. L'inconvénient de cette approche est la propagation d'erreur induite par le passage d'information d'une coupe à l'autre.

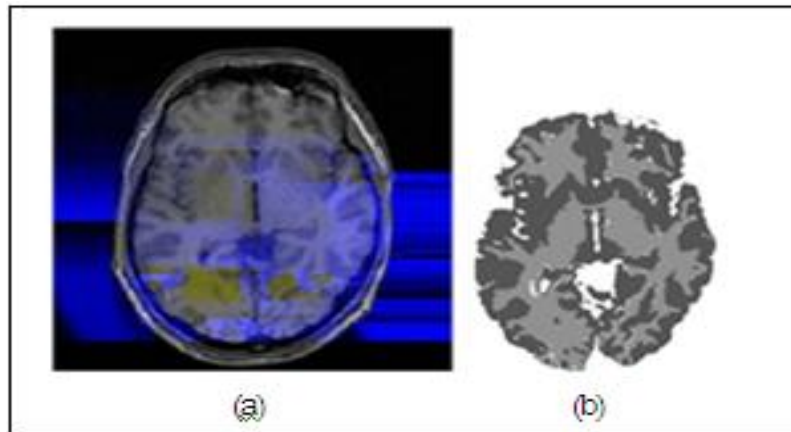


Figure 5.8 Segmentation de Germond (a) image originale et (b) image segmentée

- Mazouzi propose une approche multi-agent pour la segmentation d'images de profondeur [Mazouzi & al, 2008]. L'approche consiste en l'utilisation d'une population d'agents autonomes pour la segmentation d'une image de profondeur en ses différentes régions planes. Dans cette méthode les agents s'adaptent aux régions sur lesquelles ils se déplacent, puis effectuent des actions coopératives et compétitives produisant une segmentation collective de l'image. Les interactions indirectes entre les agents autonomes qui se déplacent dans l'image permettent à la fois, une élimination efficace de bruit, et une extraction fiable de régions. La compétition entre les agents qui s'auto-organisent autour des bordures des régions permettent de faire émerger les lignes de contours « *Figure 5.9* » dans l'image. Ces contours, dont aucune détection explicite n'est codée dans les agents, émergent des interactions et des actions collectives au sein de la population d'agents. Cette approche s'apprête à être généralisée à la segmentation d'images contenant des surfaces courbées. Pour ce faire, il est nécessaire de modéliser les surfaces dans ces images, et d'adapter le comportement des agents en fonction de la nature de ces surfaces.

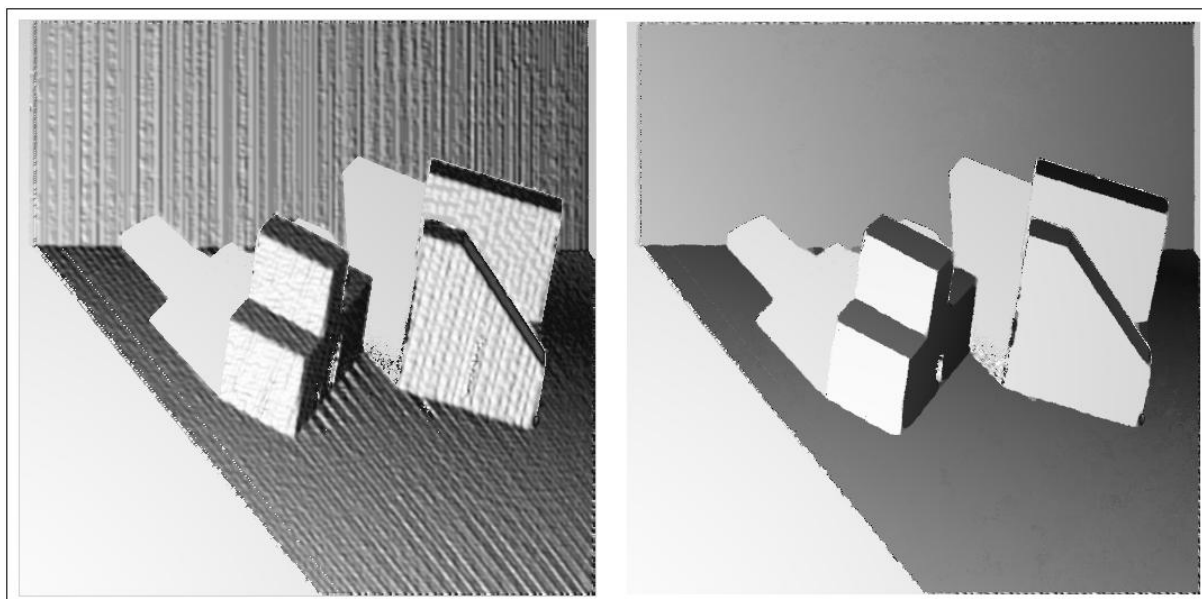


Figure 5.9 Détection de contour proposée par Mazouzi, à gauche l'image original, à droite l'image segmentée

5.9.3 Approches régions

- Pour segmenter des images scanner CT du sein, Duchesnay [Duchesnay & al, 2001] a réalisé une plateforme Multi-Agents en plusieurs étapes. Comme pré-classification, Duchesnay et al. Utilisent la décomposition en «division/fusion» avec des agents de type "région", ils obtiennent alors une segmentation qui ne donne que les zones homogènes dans l'image. Ils utilisent ensuite des agents de type "contour", sans initialisation dont l'objectif est de détecter les hautes fréquences. Chacun de ces deux types d'agents partagent via l'environnement certain caractéristique lié à l'image. Le système interne est modélisé par un réseau de Pétri et implémenté par sept règles. Un agent est ensuite placé dans chaque zone homogène et le processus de la segmentation basé sur les SMA commence. Les pyramides irrégulières sont adoptées comme élément organisationnel de la population d'agents ; la progression du processus se faisant de la base de la pyramide jusqu'au sommet. Ce processus s'arrête après un certain nombre d'itérations lorsqu'il y a convergence. Dans le même contexte, Benamrane et Nassane ont utilisé la plateforme JADE pour effectuer la segmentation d'images IRM de cerveau « Figure 5.10 » en changeant le système interne par la méthode de croissance de régions. L'inconvénient de cette méthode repose sur le choix de l'initialisation des régions initiales. et aussi la complexité des règles d'interaction qui dépendent de l'image à segmenter.

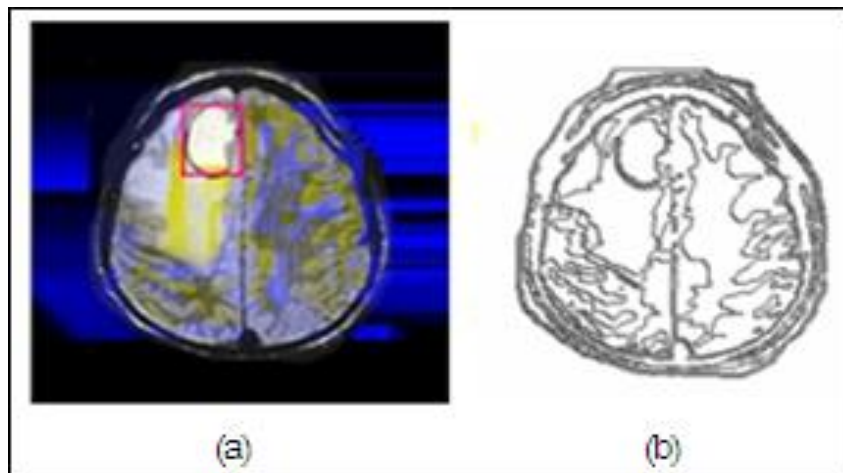


Figure 5.10 Segmentation de Huang (a) image originale et (b) image segmentée

- Bourjot [Bourjot & al, 2003] a développé un SMA basé sur les araignées sociales. Il a séparé la segmentation en deux parties : la segmentation d'une seule région et la segmentation de plusieurs régions. À l'étape initiale, les agents sont distribués aléatoirement sur l'image. Chaque agent commence à tisser des toiles selon une fonction de transition et l'objectif de la segmentation. Après un nombre d'itérations fixé par l'utilisateur, une analyse des toiles est faite pour extraire les régions de l'image « *Figure 5.11* ». L'inconvénient de cette méthode est la sélection empirique des informations couleurs de chaque colonie.

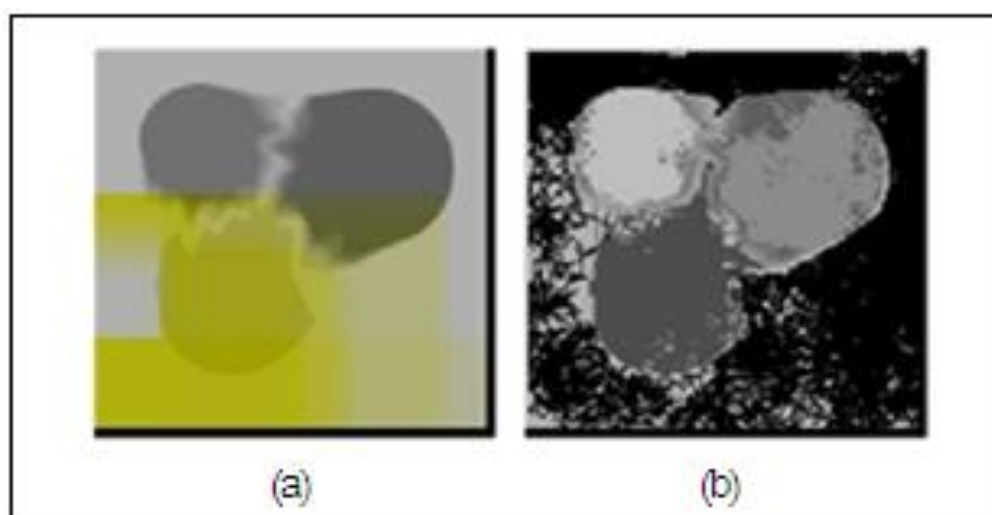


Figure 5.11 Segmentation de Bourjot (a) image originale, (b) image segmentée en 4 régions.

- Pour segmenter des images IRM de cerveau, Richard [Richard & al, 2004] a proposé une architecture hiérarchique d'agents situés et coopératifs. Trois types d'agents ont été utilisés. Un agent de contrôle global, un agent de contrôle local, et un agent dédié au tissu. Le rôle de l'agent de contrôle global est de partitionner le volume de données en territoires adjacents et d'affecter à chaque territoire un agent de contrôle local. Le rôle de ce dernier est de créer les agents dédiés au tissu, dont l'objectif est d'effectuer une croissance de régions à l'intérieur du volume local selon des critères topologiques et colorimétriques. Les paramètres de distribution des données sont mis à jour par coopération entre les agents voisins. Par l'utilisation de différents types d'agents, le système proposé a pris en compte le traitement d'images de bas niveau, ainsi que le contrôle sur les tâches de segmentation « Figure 5.12 ». Cependant, il résulte des structures complexes d'interaction dont la gestion est coûteuse. Dans le même contexte, ils proposent d'utiliser les champs de Markov cachés comme un critère de fusion. Cet algorithme présente l'inconvénient de dépendre largement du nombre de partitions qu'une image peut avoir. Ainsi, un mauvais choix à priori de ce nombre conduit inévitablement à un mauvais résultat.

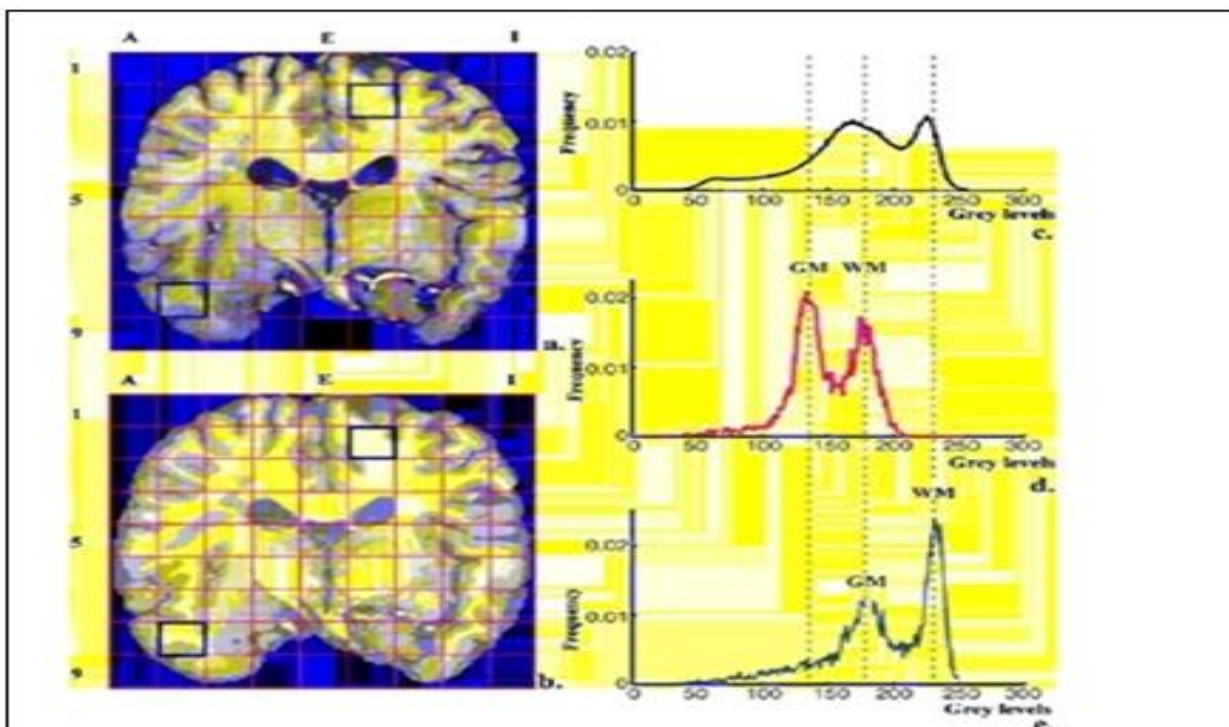


Figure 5.12 : Segmentation de Richard. (a) image originale décomposée en blocs, (b) image segmentée décomposée en blocs, (c) l'histogramme de l'image, (d) l'histogramme du bloc noir de l'image originale et (e) l'histogramme du bloc noir de l'image segmentée.

Kazar [Kazar & al, 2010] proposer une nouvelle approche de la segmentation d'images par une coopération entre agents pour la fusion et la division de régions. Dans ce travail il essaye de détecter toutes les régions dans une image, par l'utilisation des informations de recouvrement de toiles pour la différenciation et la distinction des régions. Pour cela il propose « *Figure 5.13* » deux catégories d'agent :

- Un agent de gestion du système (agent moniteur), contient les informations nécessaires au fonctionnement du système multi-agents. C'est cet agent qui est le centre de contrôle du système. C'est lui qui crée les agents de segmentation (agents araignée et agents toile) et les initialise en utilisant les informations obtenus lors de l'étape de prétraitement, il est aussi responsable de l'ordonnancement et du lancement des tâches de ses agents (agents de segmentation), de plus, il décide de la suppression des agents devenus inactifs.
- Des agents nommés agents de segmentation (agents araignées, agents toiles), chargés de l'amélioration de la pré-segmentation initiale pour obtenir des régions homogènes. Les agents araignée parcourent l'image à la recherche des régions homogène en construisant des toiles sur les régions pré segmentées qui les considère comme homogènes, en simulant le comportement des araignées sociales. L'agent toile sert à l'interprétation des toiles construites par les agents araignées à des régions finales.

L'approche proposée dans ce travail utilise des systèmes multi-agents pour l'inspiration du phénomène collectif de la construction de toile chez les araignées sociales. Chaque agent araignée explore en construisant sa toile. Il peut dépasser les limites de sa région s'il trouve qu'il y a une homogénéité par rapport aux régions voisines. La gestion de la coopération entre agents et la dynamique du système est assurée par le principe de la stigmergie, coordination par stigmergie qui est modélisée implicitement dans le comportement (déplacement influencé par la soie, plus il y a de soie vers une position, plus celle-ci a de chance d'être choisie). Enfin, les agents toiles sont créés pour l'interprétation des toiles construites en régions, par une coopération entre ces agents pour la fusion ou la division de leurs régions.

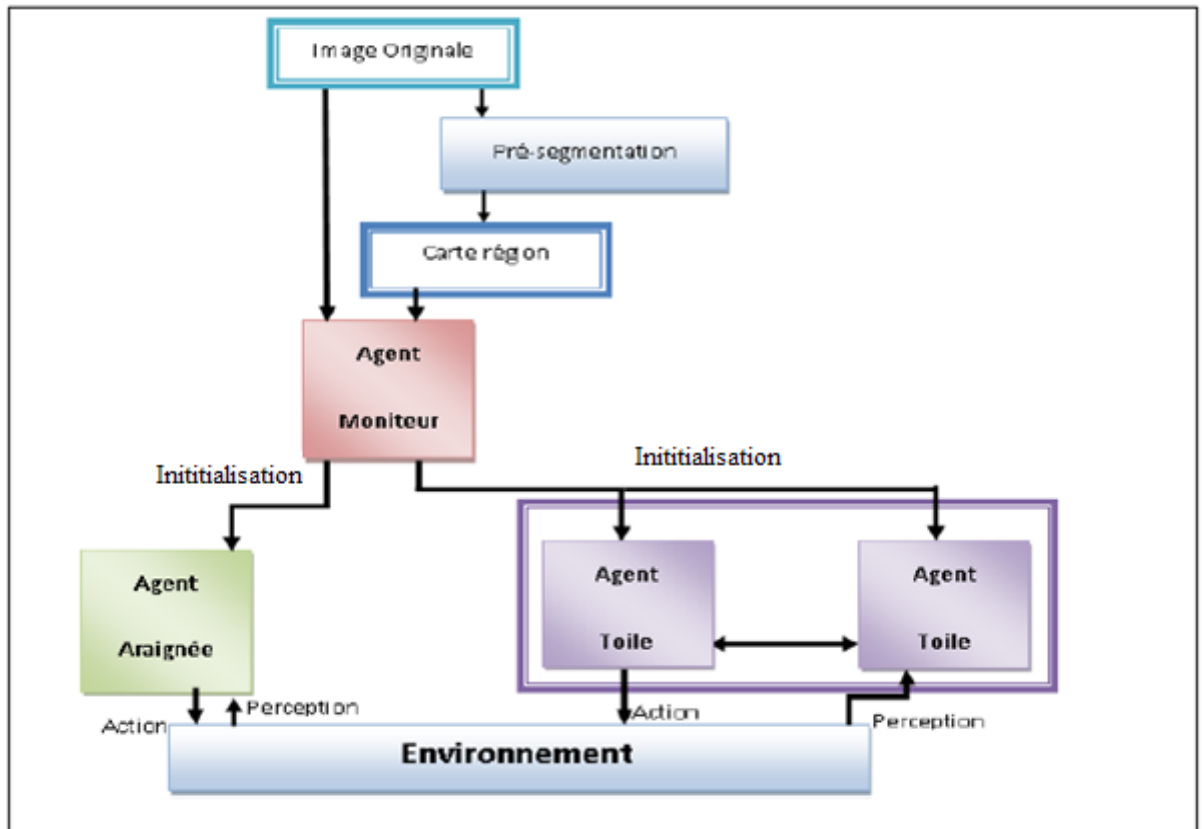


Figure 5.13 – Architecture du système multi-agent proposée par Kazar

- Pour segmenter des images IRM de cerveau, Huang [Huang & al, 2008] a développé une approche de segmentation basée sur les colonies de fourmis. Son approche considère que chaque fourmi a la capacité de mémoriser un objet de référence qui sera mis à jour quand elle trouve une nouvelle cible. Une mesure floue de la connexité est adoptée pour évaluer la similarité entre la cible et l'objet de référence. Le comportement des fourmis est affecté par le voisinage direct et la coopération entre les fourmis est effectuée de façon empirique par le partage d'informations par le biais de la mise à jour de phéromones « Figure 5.14 ». L'algorithme s'arrête après un nombre d'itérations fixé à l'avance par l'utilisateur. L'inconvénient de cette méthode est le nombre important de paramètres à déterminer (7 paramètres).

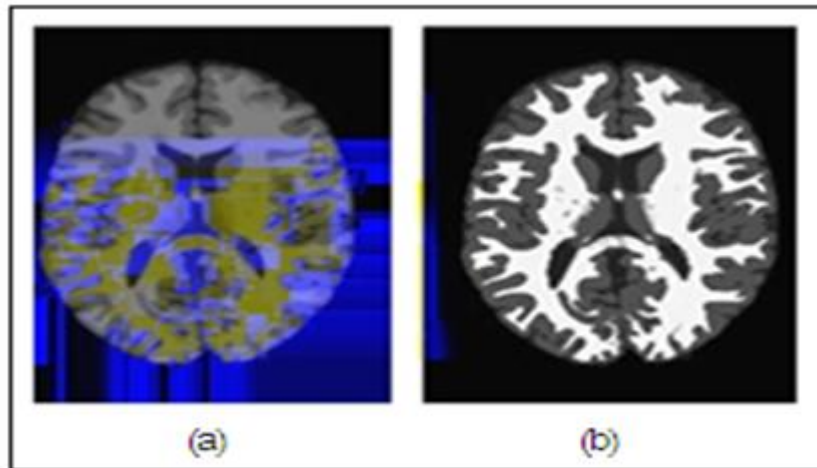


Figure 5.14 Segmentation de Huang (a) image originale et (b) image segmentée

5.10 Conclusion

Ce chapitre a présenté quelques éléments fondamentaux de la définition des Systèmes Multi-Agents. Ils permettent la modélisation distribuée des traitements et des connaissances, différents modes de coopération. Nous avons vu qu'il existait deux types d'agents, les agents réactifs et les agents cognitifs. Nous avons aussi présenté différentes approches et algorithmes pour modéliser l'activité d'un Systèmes Multi-Agents. Dans le chapitre qui suit nous allons proposer une classification des méthodes de segmentations utilisées en segmentation de profondeur.

Chapitre 6

Classification des méthodes de segmentation d'images de profondeur

6.1 Introduction

La revue de littérature que nous avons présentée au troisième chapitre a permis de montrer comment les méthodes élémentaires étaient utilisées en pratique pour segmenter des images de profondeur, formant parfois des méthodes combinées. Nous avons dans ce chapitre procéder à une classification de ces méthodes.

6.2 Classification des méthodes de segmentation

Notre choix sur le critère de classification s'est porté sur le critère d'homogénéité auquel obéissent les méthodes de segmentation. Nous avons ainsi pu scinder ces méthodes en quatre catégories distinctes [Khaldi & al, 2012(a)].

6.2.1 Segmentation selon le type de courbure

Ces méthodes sont confrontées au problème d'estimation de la courbure, celle-ci est perturbée à la fois par le bruit et les discontinuités, la segmentation n'est représentative de la surface mesurée que si le bruit est faible et si les perturbations dues aux discontinuités sont éliminées par un traitement adéquat. Comme nous l'avons précédemment vu Besl et Jain utilisent cette segmentation uniquement pour déterminer les graines initiales pour une croissance indépendante de régions où les images traitées sont de relativement mauvaise qualité. Nous pouvons rajouter à cette catégorie la méthode de segmentation de Kasvand qui procède à une érosion de pixels sur l'image de type de courbure pour supprimer l'effet des discontinuités. Ainsi que la méthode d'étiquetage des composants connexes sur l'image de type de courbure de Yokoya et Levine « *Figure 6.1* ».

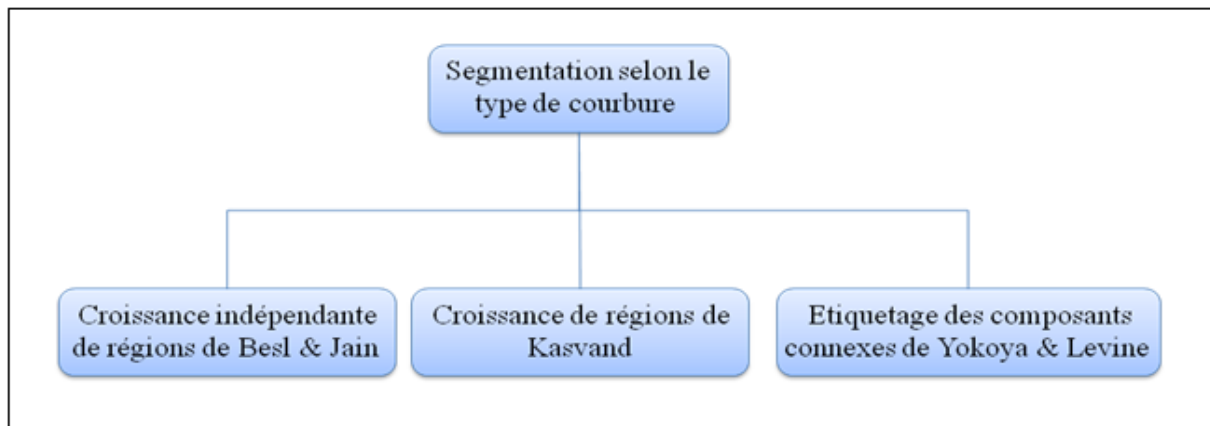


Figure 6.1–Méthodes de segmentation d’images de profondeur par type de courbure

6.2.2 Segmentation sur surface algébrique

Segmentation sur surface algébrique : Concernant les surfaces algébriques il y a deux types de méthodes, division/fusion et croissance indépendante de région. Les méthodes de division/fusion comme celle présentée par Leonardi qui est une méthode de division-fusion liée à un modèle de surface 2.5D de type polynomial. Cette méthode est mal adaptée à la segmentation des images de profondeur suivant un modèle de surface algébrique notamment à cause des discontinuités. En ce qui concerne les méthodes de croissance indépendante de régions, on trouve deux approches qui se distinguent au niveau de leur principe fondamental. L'approche initiée par Jiang et Bunke qui cherche une représentation approximative minimale de l'image de profondeur et l'approche de Gupta et Bajcsy fait une recherche exhaustive de la meilleure localisation d'un ou de plusieurs modèles de surface dans l'image « *Figure 6.2* ».

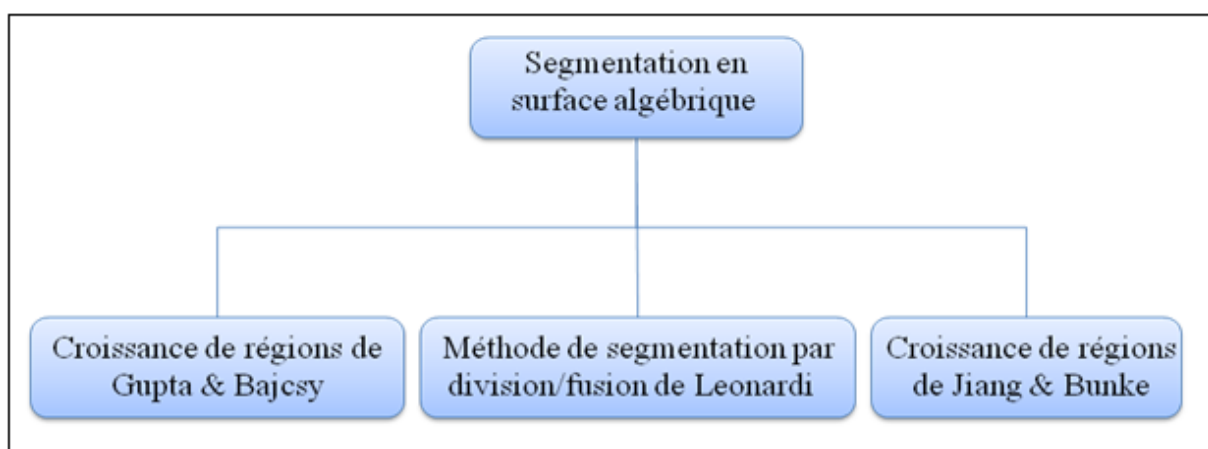


Figure 6.2–Méthodes de segmentation d’images de profondeur sur surface algébrique

6.2.3 Segmentation sur surface plane

On trouve dans cette catégorie les trois méthodes de division / fusion basées sur la normale (gradient) à la surface de Bhavsar [bhavsar& al, 2010], Parvin & Medioni [Pavini& al, 1986] ainsi que Taylor [Taylor & al, 1989]. Ces trois méthodes « *Figure 6.3* » prennent en compte les discontinuités de profondeur pour éviter une fusion entre deux plans parallèles mais non colinéaires ce qui est fréquent en segmentation sur surface plane (par exemple une face supérieure d'un cube et une surface sur laquelle il est posé). Les méthodes sont nombreuses, mais la plupart utilisent des propriétés particulières à une surface plane. Nous ajoutons à cette classe la méthode de détection de contour par système multi-agent de Mazouzi. La segmentation en surfaces planes devrait être traitée de la même manière que le cas plus général des surfaces algébriques.

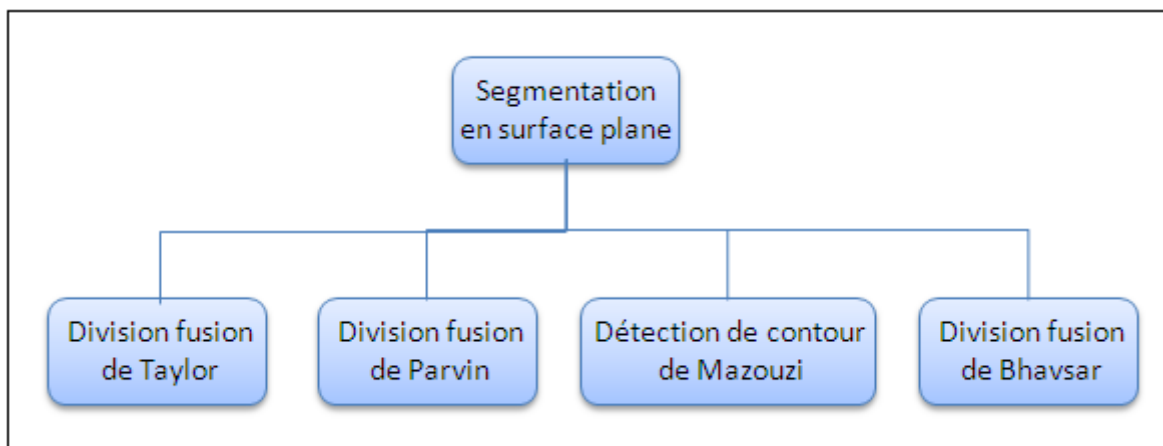


Figure 6.3–Méthodes de segmentation d’images de profondeur sur surface plane

6.2.4 Segmentation de continuité d’ordre1

Segmentation de continuité d’ordre1 : A côté de la méthode simple de segmentation suivant C1, il existe des méthodes complexes, tout l’effort investi par ces méthodes est toutefois consacré à forcer la fermeture des frontières définies par les discontinuités, une fermeture systématique des frontières correspond à une modification du critère d’homogénéité le rendant plus contraignant que la pure continuité C1. Comme nous pouvons le constater, l’algorithme proposé par Besl [Besl& al, 1988] il suffit qu’une partie seulement de la frontière entre segments corresponde à une discontinuité pour que la fusion n’ait pas lieu et ce procédé aura donc un effet de contrainte de fermeture systématique des frontières associées aux

discontinuités. Nous pouvons rajouter à cette catégorie « *Figure 6.4* » la méthode de détection des discontinuités et d'étiquetage des composantes Connexes proposée par Davignon [Davignon, 1992] où la détection de discontinuités d'ordre 0 et 1 est effectuée. Il est à noter qu'elle a (la segmentation en Continuité C1) l'avantage de fournir, contrairement aux autres méthodes, une représentation simplifiée du segment.

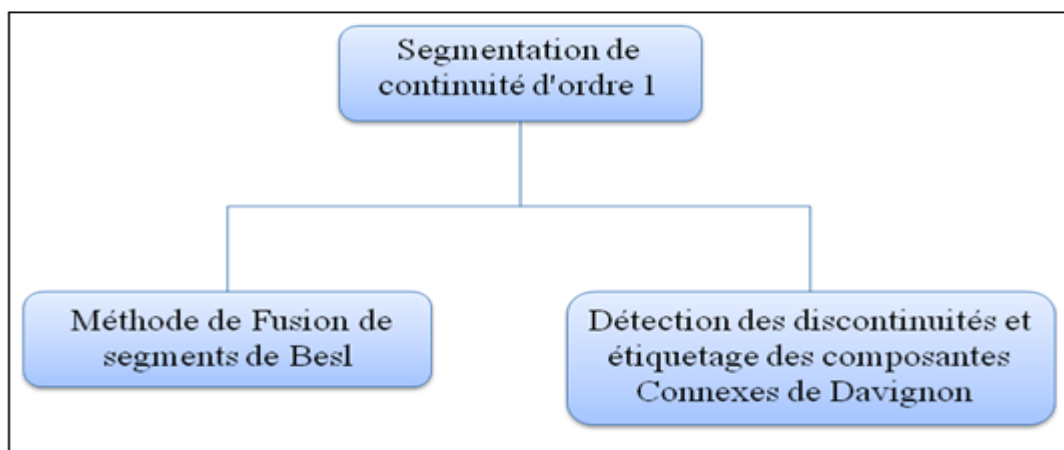


Figure 6.4—Méthodes de segmentation d'images de profondeur suivant la continuité d'ordre 1

Nous avons dans la classification proposée « *Figure 6.5* » affecté les trois méthodes de division fusion de Bhavsar, Parvin et Taylor à la catégorie de segmentation en surface plane où comme nous l'avons vu ces trois techniques prennent en compte les discontinuités de profondeur pour éviter une fusion entre deux plans parallèles mais non colinéaires (par exemple une face supérieure d'un cube et une surface sur laquelle il est posé).

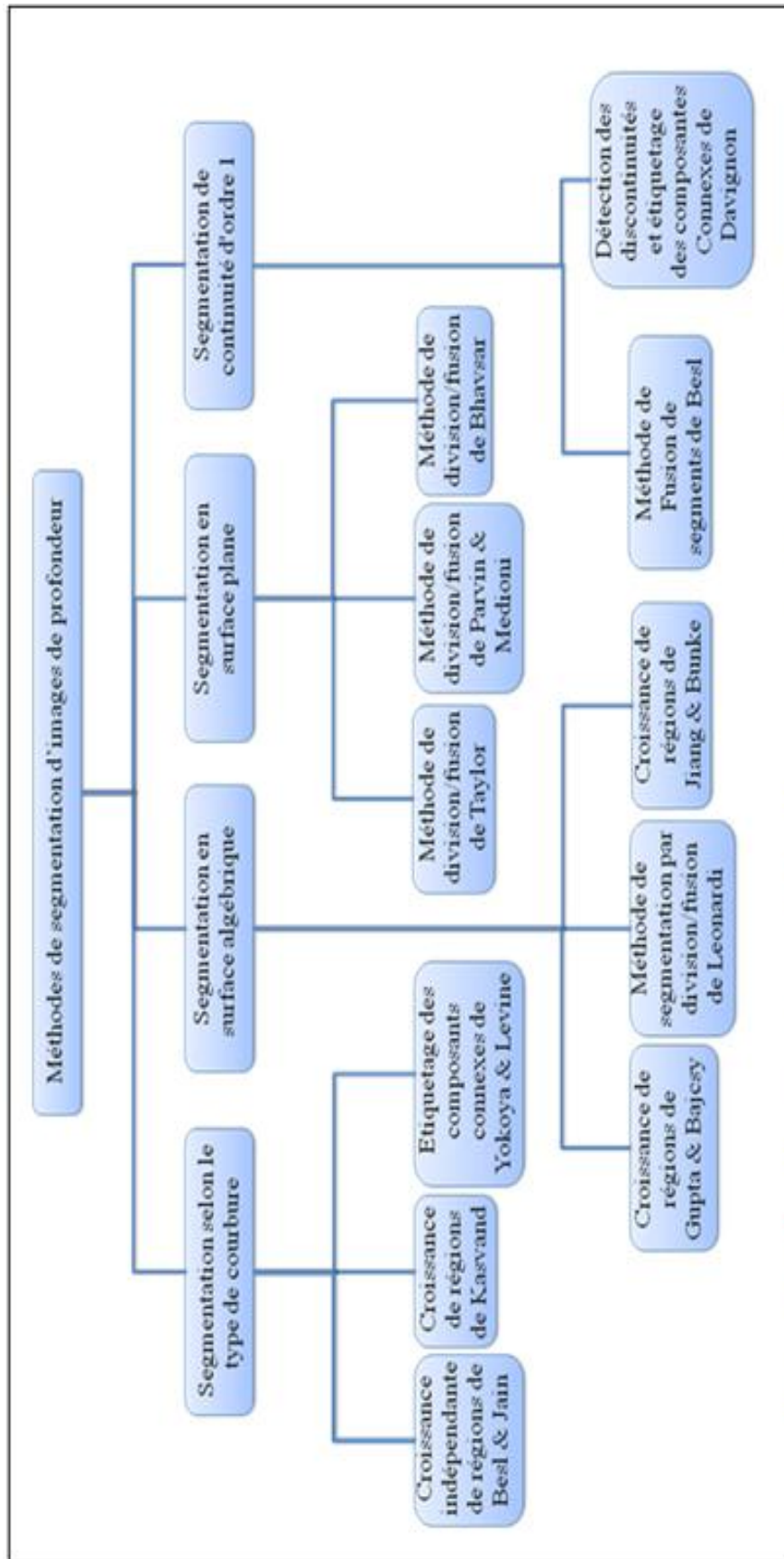


Figure 6.5—Classification des méthodes de segmentation d'images de profondeur

6.3 Conclusion

Bien que le critère d'homogénéité de la segmentation résultante ne soit pas toujours explicite, nous avons tout de même pu classer pratiquement toutes les méthodes suivant les quatre critères d'homogénéité (type de courbure, surface plane, surface algébrique, continuité C1). Nous avons ainsi regroupé les méthodes de Besl et de Kasvand qui sont utilisées en segmentation sur type de courbure où l'estimation de la courbure pose problème et est perturbée par le bruit et les discontinuités. Nous allons dans le chapitre qui suit mener une expérimentation où nous appliquerons trois modèles déformables (contour actif classique, contour actif adaptatif et Level set) pour segmenter des images de profondeur.

Chapitre 7

Segmentation d'images de profondeur par contour actif

7.1 Introduction

Durant notre étude bibliographique nous avons abordé différentes approches de segmentation en imagerie de profondeur et nous avons constaté la similarité des méthodes utilisées avec celle utilisées pour la segmentation d'images 2D classique, cependant, nous avons remarqué qu'aucun auteur n'a expérimenté ni présenté une méthode de segmentation d'images de profondeur en utilisant les contours actifs. Nous allons donc dans ce chapitre appliquer trois algorithmes de modèles déformables (Adaptive snake, Classique snake et Level sets) à des images de profondeurs. Cependant, les images de profondeur sont réputées pour être énormément bruitées, bruit qui est dû aux conditions d'acquisition de ces images, pour toute éventuelle segmentation sur ce type d'images un prétraitement est nécessaire pour améliorer la perception globale de l'image de profondeur. Comme les contours actifs sont très sensibles aux petites variations d'intensité, il est donc nécessaire d'atténuer le bruit dans les images. Nous allons donc commencer par présenter l'algorithme de Floyd & Steinberg qui est une technique de traitement du signal utilisé en traitement d'image pour atténuer le bruit et donner une meilleure perception de l'image de profondeur. Puis, nous allons présenter et appliquer trois algorithmes de modèles déformables (Adaptive snake, Classique snake et Level sets) à des images de profondeurs. Après avoir appliqué les trois modèles déformables précédemment décrits nous finirons ce chapitre en effectuant une comparaison des résultats de segmentation obtenus par les trois algorithmes pour essayer de trouver le modèle déformable le plus adapté à la segmentation d'images de profondeur.

7.2 Prétraitement

Les images de profondeur sont connues pour être fortement bruitées et cela rend leur segmentation un peu délicate, les rangels bruyants conduisent souvent à de mauvais résultats de segmentation et il nous faut donc, avant de les segmenter, prétraiter ces images pour améliorer leur qualité (rapport signal sur bruit et homogénéité des zones), pour atténuer les

effets de tout bruit présent, l'image est filtrée par le filtre Floyd & Steinberg qui améliore la perception globale d'une image et évite l'exclusion des zones de faible intensité.

7.2.1 Filtre de Floyd & Steinberg

L'algorithme de Floyd-Steinberg [Hocevar & al, 2008] est un exemple de la technique de diffusion d'erreur, l'objectif est d'utiliser de simples seuils de filtrage sur chaque rangel mais il faut prendre en compte avec précision les erreurs de luminosité qu'elle peut induire. Si on considère une image de 50% de gris (une image avec chaque rangel exactement à mi-chemin de luminosité entre le noir et blanc). La solution sera de prendre en compte le résultat du filtrage d'un rangel « Figure 7.1 » lorsqu'on appliquera ce filtre à ses voisins (l'erreur à la suite d'un filtrage d'un rangel est décalée à ses voisins et il influencera donc leur filtrage aussi).

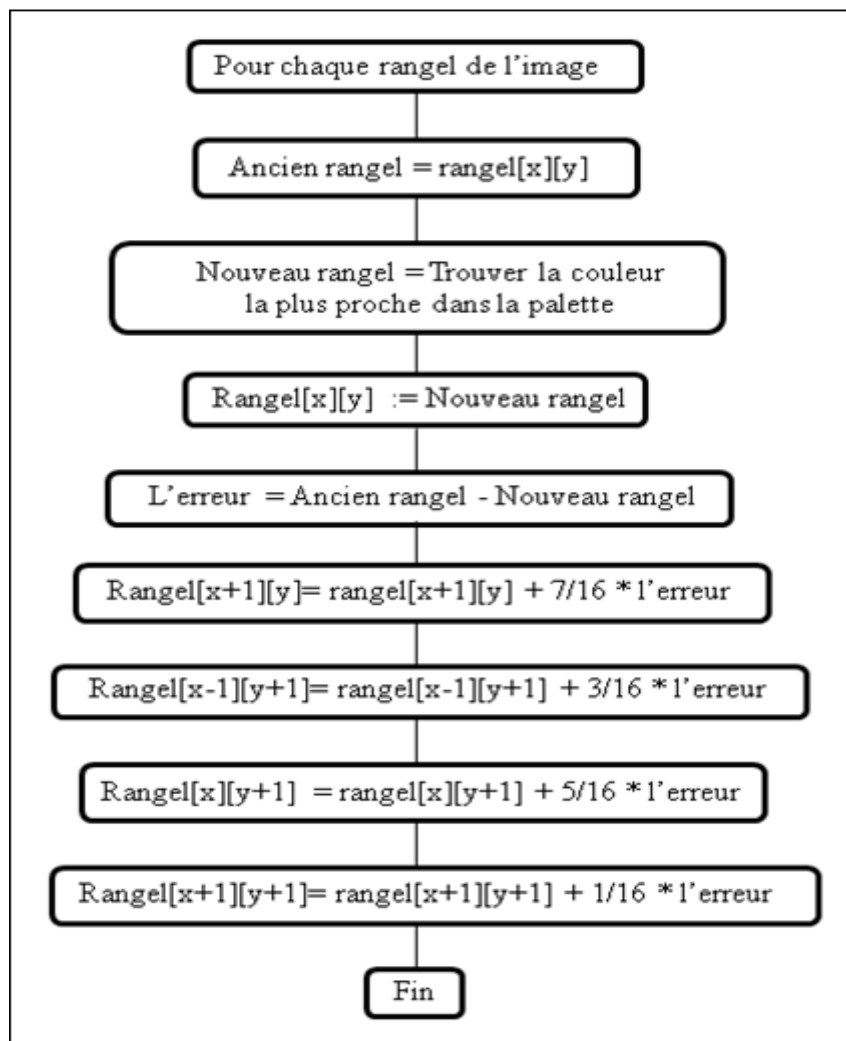


Figure 7.1 – Algorithme de Floyd-Steinberg

Le filtrage est réalisé par propagation de l'erreur de quantification. Lorsqu'on veut afficher un point, on choisit parmi la palette des couleurs disponibles celle qui est la plus proche de la couleur exacte du pixel. L'erreur induite par cette approximation est ensuite reportée sur les pixels voisins du pixel traité. En pratique, seule une partie des voisins est utilisée « *Figure 7.2* ». Ceci permet de traiter l'image en une seule passe. L'erreur est ensuite divisée entre chaque voisin et ajoutée à leur valeur « C » et cette valeur actualisée sera utilisée quand viendra leur tour d'être approximés. Pour tenir compte de leurs positions respectives, les corrections apportées à chaque voisin sont pondérées différemment.

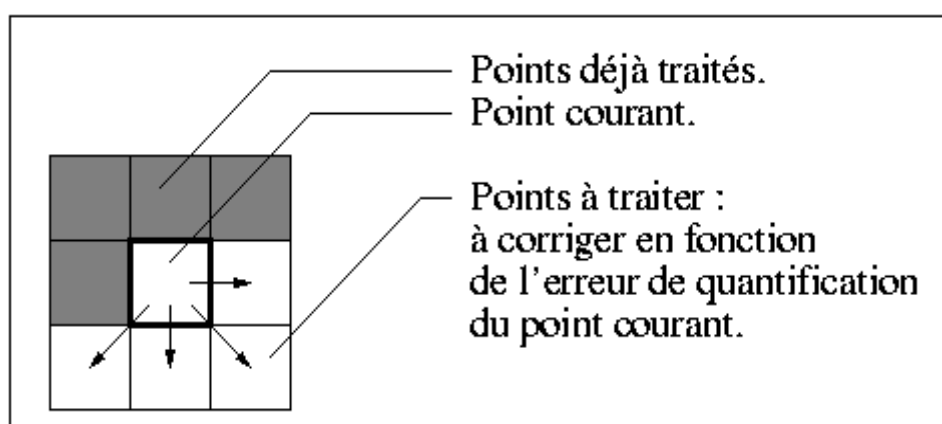


Figure 7.2 – Algorithme de Floyd-Steinberg

7.2.2 Résultats du filtrage

Ce tramage matriciel 3x3 a un bon respect des nuances (10 niveaux). En contrepartie, il introduit un flou sur le bord des formes et on voit apparaître un quadrillage peu esthétique « *Figure 7.3* ». Il faut tenir compte du fait que les pixels de la dernière colonne de l'image n'ont pas de voisins de droite et que ceux de la dernière ligne n'ont pas de voisins de dessous. Ils sont néanmoins modifiés, notamment par le traitement de la colonne ou de la ligne précédente. Cette méthode est souvent utilisée par les navigateurs web pour afficher les photos en couleurs quand on ne dispose que de 256 couleurs par exemple.

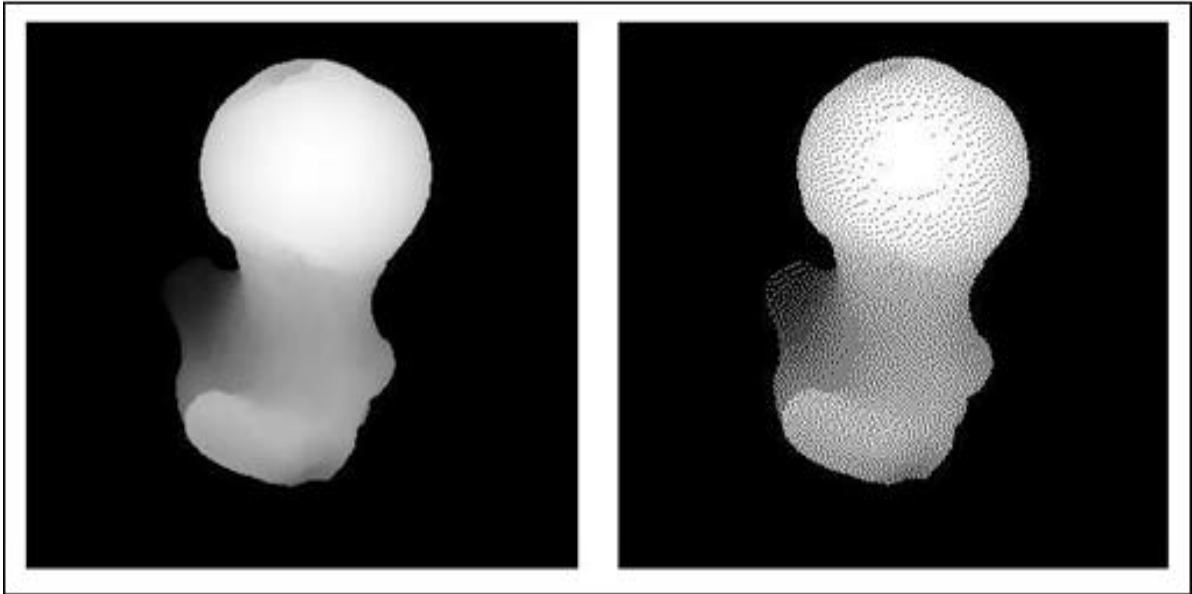


Figure 7.3 – Filtrage de l’image par Floyd & Steinberg. A gauche l’image de profondeur, à droite l’image filtrée.

Avant d’avoir eu recours à un prétraitement de l’image avant segmentation, nous avons essayé de segmenter l’image de profondeur par contour actif classique et comme nous pouvons le constater dans la *Figure 7.4*, les zones de faible intensités sont négligées lors de la segmentation. C’est cette négligence qui nous a poussé à prétraiter l’image de profondeur avant toute segmentation.

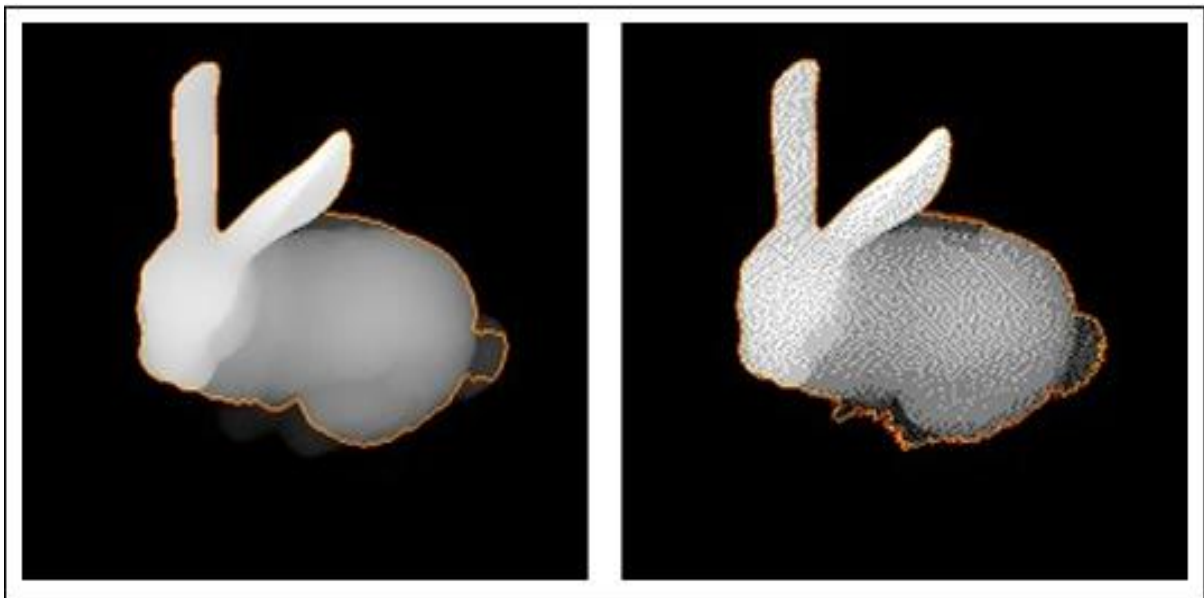


Figure 7.4 – Résultats de segmentation (a) image segmentée sans prétraitements. (b) image segmentée avec utilisation du filtre de Floyd & Steinberg.

7.3 Base d'images utilisé

Les images utilisées dans notre travail proviennent de la base d'images de l'université de Stuttgart, cette base contient une collection d'images de profondeur provenant de modèles polygonaux de haute résolution disponibles sur le web (<http://range.informatik.uni-stuttgart.de/htdocs/html/>), six de ces modèles ont été reconstruites à partir de scanners de profondeur dans le laboratoire de Stuttgart. Nos expérimentations ont été conduites sur un PC standard doté d'un processeur Intel(R) 2.20GHz et disposant de 1GB de RAM.

7.4 Langage de programmation utilisé

Les trois modèles déformables que nous allons utiliser et évaluer dans ce chapitre sont tout trois conçu en langage Java. Le Java est un langage de programmation à usage général, évolué et orienté objet dont la syntaxe est proche du C. Ses caractéristiques ainsi que la richesse de son écosystème et de sa communauté lui ont permis d'être très largement utilisé pour le développement d'applications de types très disparates. Java est notamment largement utilisée pour le développement d'applications de traitement d'images ainsi que des applications d'entreprises et mobiles.

7.5 Framework de comparaison

Afin de procéder à une évaluation numérique des résultats obtenus par chaque algorithme, nous avons procédé à une segmentation manuelle de notre base d'image, cette segmentation sera utilisée comme segmentation de référence pour évalué les images de segmentations résultantes. Puis, nous avons conçu une application en C++ Builder « *Figure 7.5* » qui est un un logiciel de développement rapide d'applications conçu par Borland qui reprend les mêmes concepts, la même interface et la même bibliothèque que Delphi en utilisant le langage C++. Il permet de créer rapidement des applications Win32 ainsi qu'une interface graphique avec son éditeur de ressources.

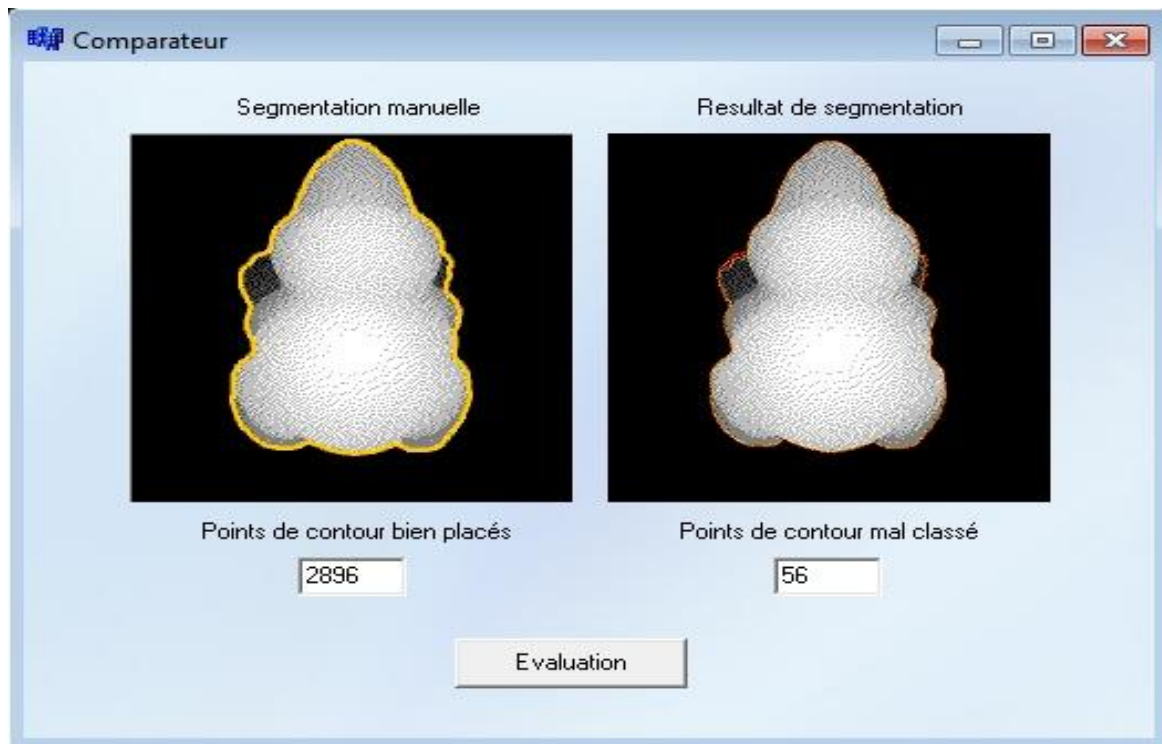


Figure 7.5 – Framework de comparaison de points de contour

Ce Framework aura comme entrée l'image de profondeur segmentée manuellement et l'image de profondeur segmentée par un des modèles déformables, le Framework va extraire tout les points de contour des deux images et comparera la position des points de contour des deux images «*Figure 7.6*». Si la position d'un point de contour de l'image segmentée automatiquement correspond au point de contour de l'image segmentée manuellement on pourra alors dire qu'il est bien placé, le cas échéant le point serait alors mal placé. Après avoir traité tout les points de contour, le nombre de points bien placés par rapport au nombre de points de contour de l'image manuellement segmentée nous permettra de calculer le pourcentage de pixels bien placés et ainsi évaluer les résultats de segmentation obtenues.

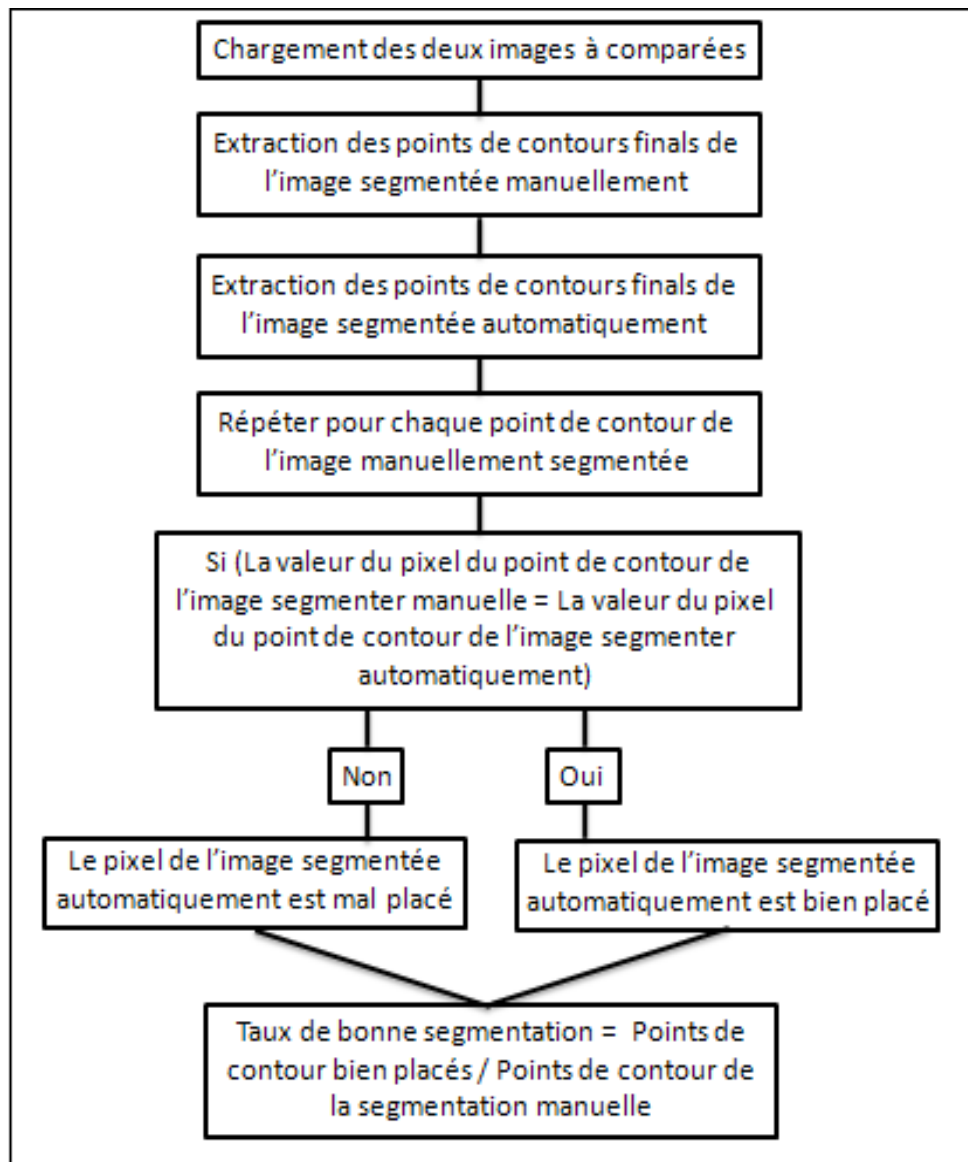


Figure 7.6 – Algorithme du Framework de comparaison

7.6 Algorithmes utilisés

Parmi les différents modèles déformables cité dans le chapitre cinq, nous avons retenu pour notre travail trois modèles déformable pour segmenter les images de profondeur.

7.6.1 Le contour actif classique (Snake)

Le snake classique que nous utilisons est une courbe paramétrée C qui sera attirée par les zones de forts gradients, c'est à dire où la norme du gradient d'un point sera élevée. Le principe consiste à placer dans l'image au voisinage de la forme à détecter un contour

initial qui sera ensuite déformé sous l'action de plusieurs forces. Cette manœuvre demande une certaine énergie « *Figure 7.7* » constituée d'une énergie interne et d'une énergie externe :

- Une énergie interne E_{int} , L'énergie interne va dépendre des dérivées du premier et du second ordre de la courbe paramétrée représentant le snake. Cette énergie correspond à la morphologie et aux caractéristiques de la courbe telles que la courbure, la longueur, cette énergie est définie par la fonction suivante :

$$E_{int} = \alpha(s) (dv/ds)^2 + \beta(s) (d^2v/ds^2)^2 \quad (7.1)$$

Où α est le facteur d'élasticité β le facteur de rigidité du contour permettant ainsi d'obtenir des courbes plus ou moins lisses.

- L'énergie externe concerne l'image et ses propriétés et caractéristiques telles que la présence de bords ou encore de bruit. Elle permet de s'assurer que le snake se trouve sur les bords l'image en maximisant la quantité de la norme gradient tout au long de la courbe et en minimisant donc son opposé, cette énergie est définie par la fonction suivante :

$$E_{ext} = -\lambda(s) \left| \Delta I(v(s)) \right|^2 \quad (7.2)$$

Où le facteur λ dépend de l'image I initiale et Δ est l'opérateur gradient. On peut faire précéder le gradient d'un filtrage passe-bas de l'image permettant d'obtenir des contours moins bruités et d'augmenter leur zone d'influence.

```

for(int dy=-1;dy<=1;dy++) {
    for(int dx=-1;dx<=1;dx++) {
        p.setLocation(cur.x+dx, cur.y+dy);
        e_unif[1+dx][1+dy] = f_unif(prev,next,p);
        e_curv[1+dx][1+dy] = f_curv(prev,p,next);
        e_fl[1+dx][1+dy] = f_fl(cur,p);
        e_ine[1+dx][1+dy] = f_ine(cur,p);
    }
}

```

Figure 7.7 – Calcul d'énergie du contour actif classique

Le contour initial est tout d'abord placé au centre de l'image en forme de cercle, le déplacement et l'évolution de ce modèle se fait par l'itération de l'algorithme « *Figure 7.8* » afin de minimiser l'énergie totale. Pour chaque point P du contour actif, la fonction d'énergie E est calculée pour tous les points n appartenant au voisinage de P . Le point P_0 caractérisé par l'énergie minimale E_0 est alors choisi pour remplacer P si $E > E_0$. Dans le cas contraire, le point de contour P n'est pas modifié. Ce mécanisme est répété « *Figure 7.9* » jusqu'à convergence (lorsque le contour obtenu à l'itération t est identique à celui obtenu à l'itération $t+1$).

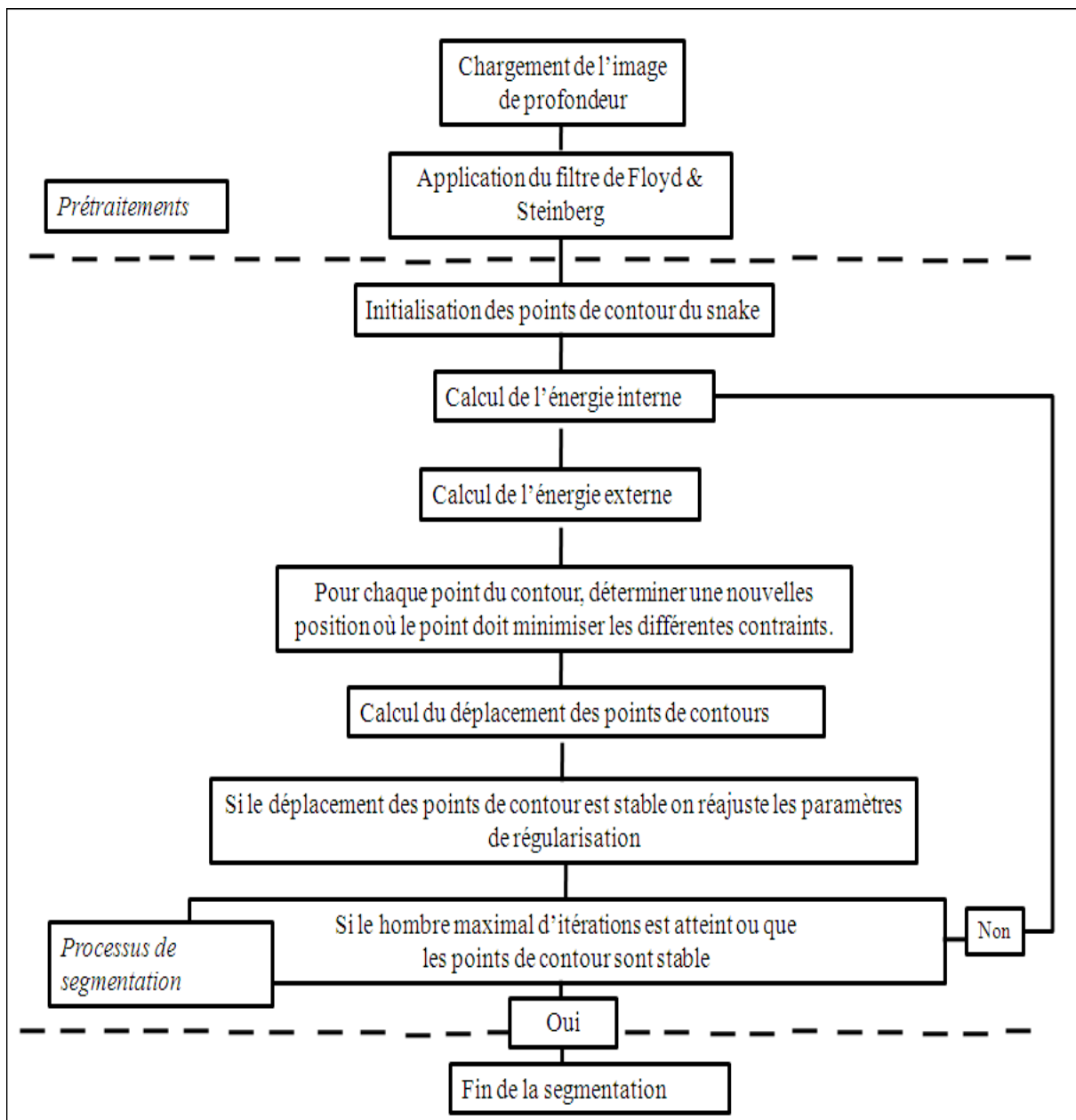


Figure 7.8 – Algorithme du contour actif classique

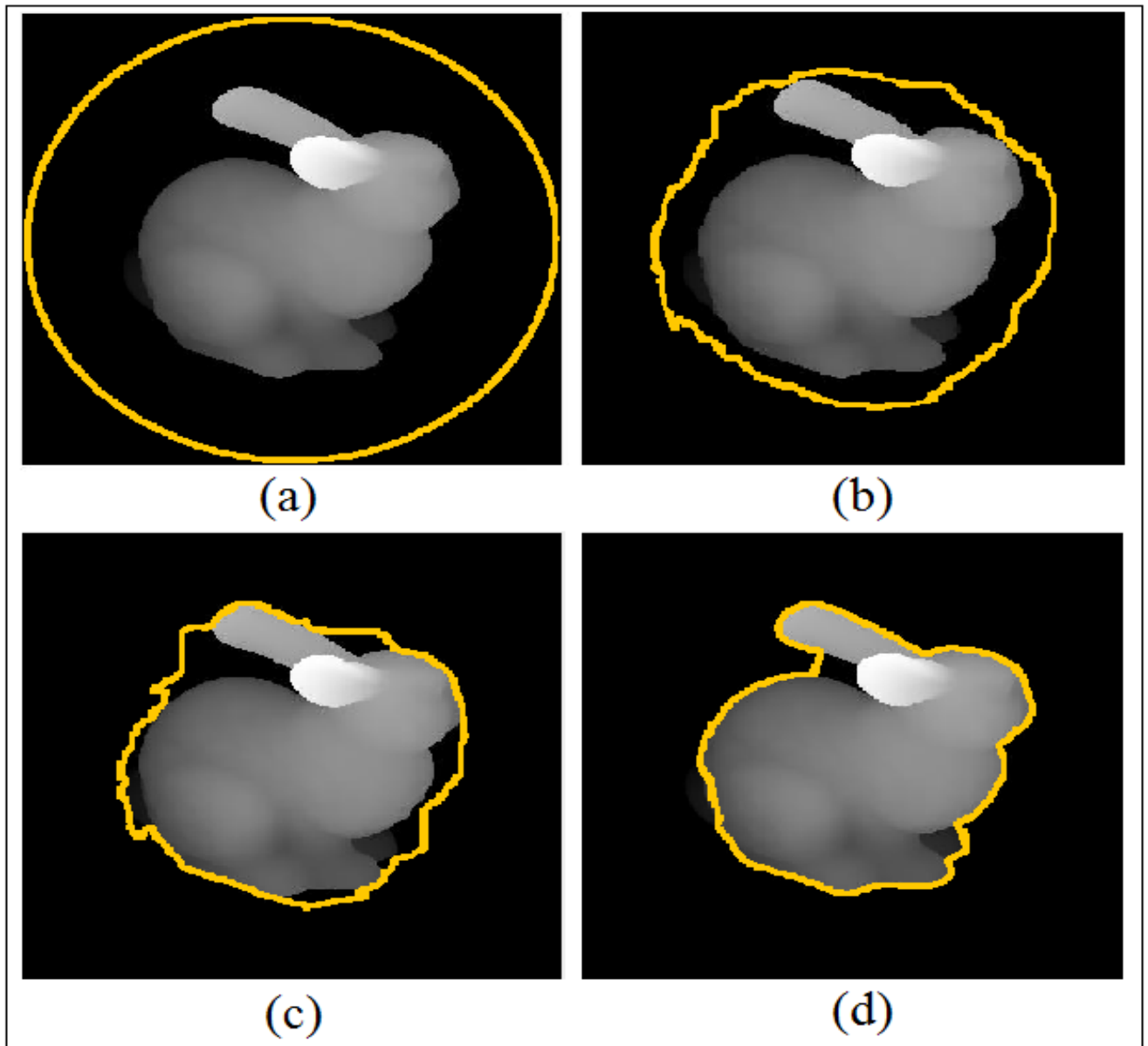
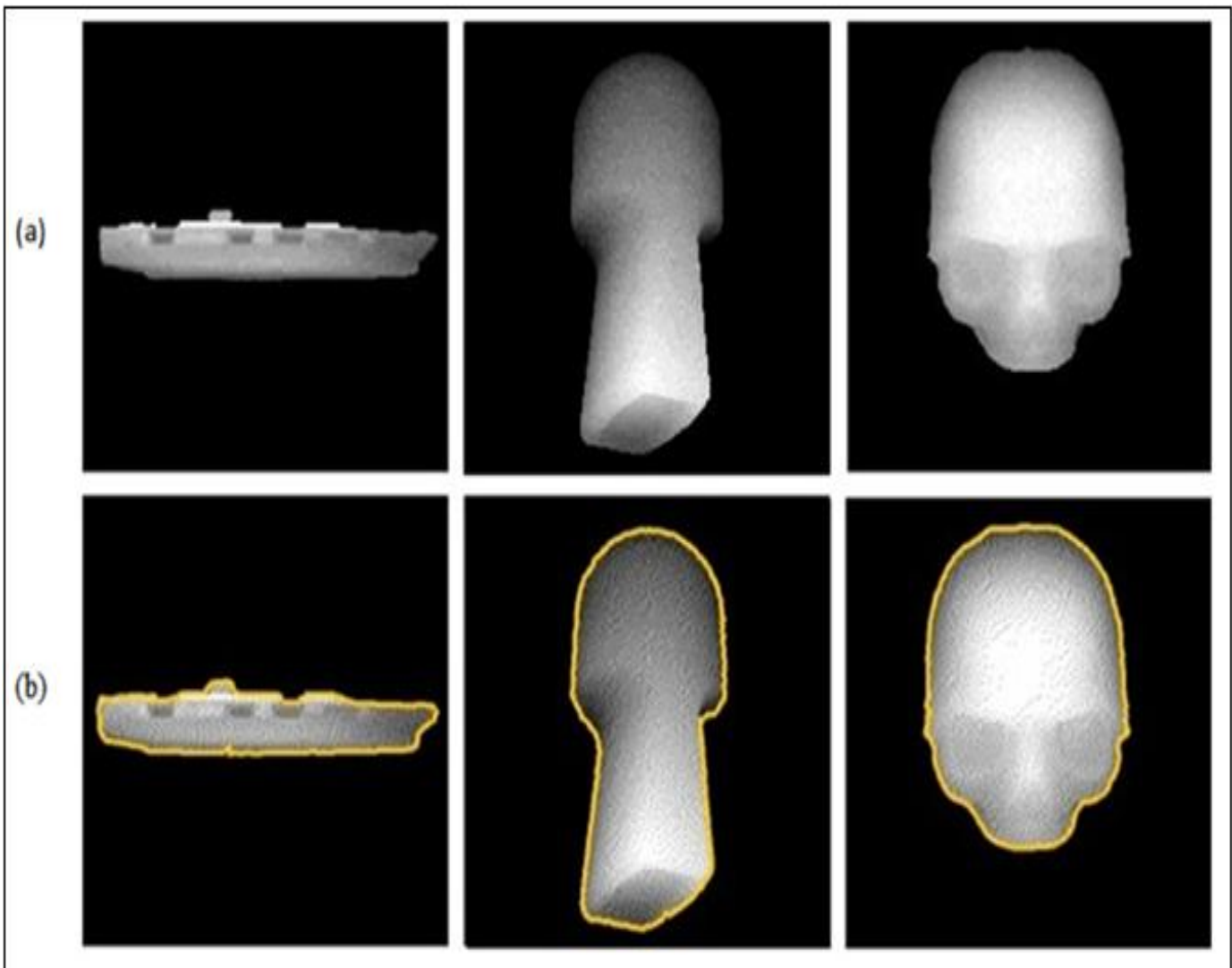


Figure 7.9 – Déformation du contour actif classique, (a) contour initial, (b) contour après 30 itérations, (c) contour après 50 itérations, (d) contour après 100 itérations.

Nous avons choisie 82 images que nous scindons en deux catégories, des images simples et des images comportant des cavités. Avant toute expérimentation, toutes ces images sont préalablement filtrées par l'algorithme de Floyd & Steinberg. Puis nous appliquons le modèle déformables (snake classique) aux 82 images de profondeur, comme on peut le voir dans la « *Figure 7.10* » une évaluation visuelle permet de constaté que les résultats de segmentation obtenus par cette méthode offrent une segmentation parfaite pour les images simples qui ne comporte pas de fortes cavités.



**Figure 7.10– Segmentation d’images de profondeur simple par contour actif classique:
 (a) Image de profondeur (b) Segmentation obtenue par le snake classique**

Cependant, les résultats des segmentations obtenues pour les images de profondeur comportant des cavités « *Figure 7.11* » ne sont pas complets et cela montre les limites du contour classique. Le snake traditionnel ne converge pas correctement vers les concavités car la force verticale qui permet de passer le snake vers la concavité est nulle à cause du champ de forces externes qui a été défini. Le modèle a convergé avant que certains éléments du contour actif n’aient pu atteindre certains bords de la forme.

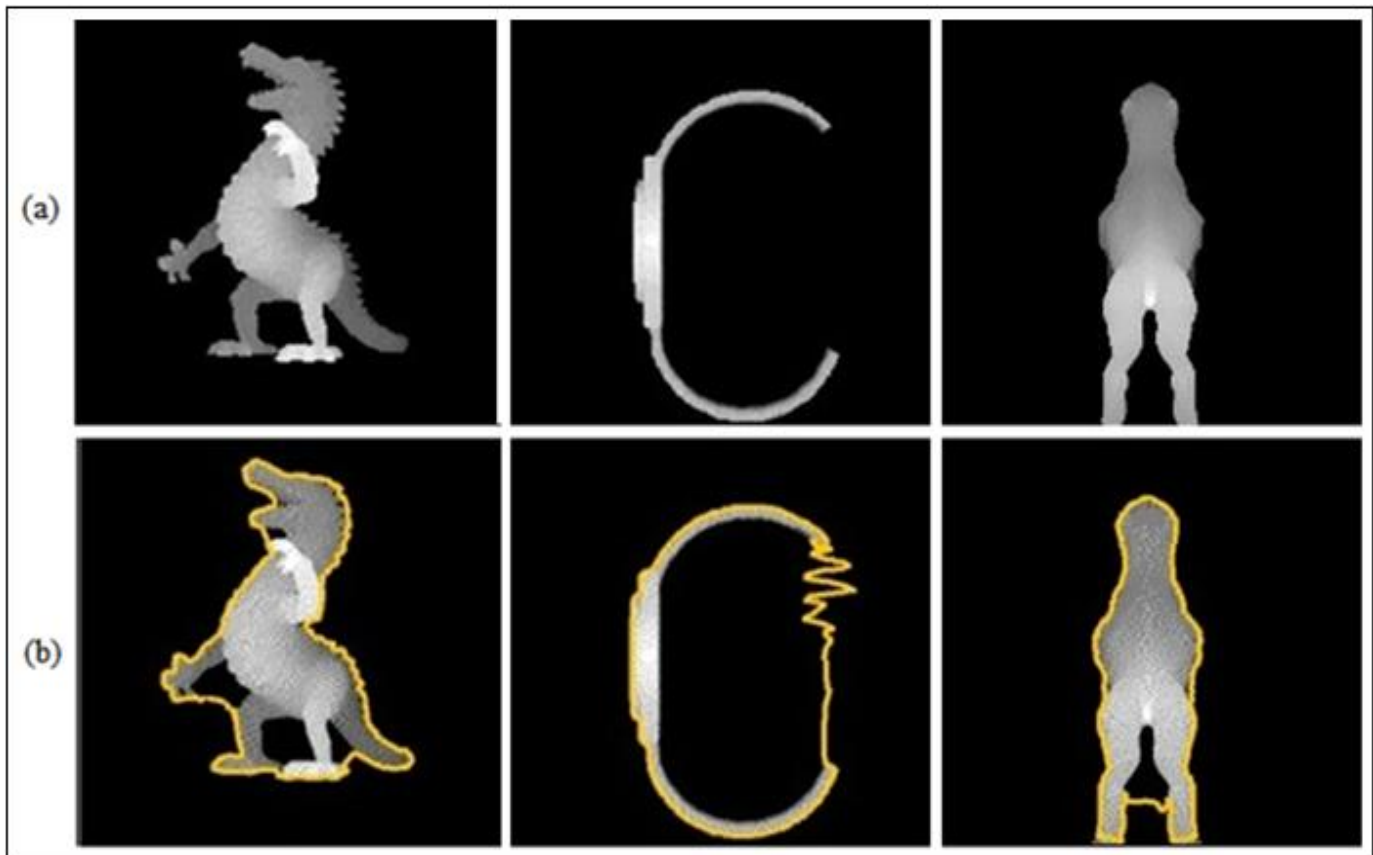


Figure 7.11 – Segmentation d’images de profondeur avec cavités par contour actif classique: (a) L’image de profondeur (b) Segmentation obtenue par le snake classique

Nous avons ensuite comparer les 82 images segmentées par le contour actif classique avec les images segmentées manuellement en utilisant le Framework de comparaison vu précédemment, les résultats sont présentés dans la Tableau 7.1.

Tableau 7.1 – Taux de bonne segmentation obtenus par l’algorithme de snake classique

Image de profondeur	Nombre de points de Contour généré par la segmentation manuelle	Nombre de points de contour mal placés	Taux de bonne segmentation
Dinopet2	9053	1806	75.07 %
Copter2	5526	1060	76.26 %
Dinopet3	8498	1618	76.48 %
Dinopet5	8172	1378	79.71 %
Dinopet1	5185	812	81.43 %
Dinopet6	7547	844	87.40 %
Dinopet4	8432	847	88.83 %
Copter1	5737	573	88.90 %
Agfa3	4526	36	99.19 %
Baljoint2	5127	34	99.33 %
Carrier2	4104	19	99.53 %
Entel	6589	26	99.60 %

7.6.2 Le contour actif adaptatif

Ce modèle [Khaldi & al, 2012(b)] combine deux approches classiques, les courbes déformables et les contours actifs classiques. Dans la formulation des courbes déformables, seule la composante normale du vecteur de déformation agit sur la forme de la courbe, ce modèle est basé sur la déformation le long des normales et rend le snake attiré par les bords les plus proches trouvé perpendiculairement au modèle (Détecter les contours perpendiculairement au modèle est nécessaire pour entrer dans les cavités), comme dans l'approche classique du snake, la régularisation du modèle est assurée par la minimisation de une énergie interne calculée comme suit :

$$E_{int} = \int_s (\alpha(s)|v'(s)|^2 + \beta(s)|v''(s)|^2)ds \quad (7.3)$$

Où $v(s)$ est la courbe et α, β sont des paramètres de régularisation qui varient le long de la courbe. β est arbitrairement fixé à zéro pour augmenter la vitesse de calcul.

Initialement, le contour est un grand cercle contournant tout les formes de la scène. Puis à chaque itération, chaque point P est attiré vers le bord le plus proche le long de la normale de ce point (Les bords sont détectés en utilisant l'opérateur de Canny-Deriche). A partir de la position d'un point de P_1 , on recherche le long de la normale de ce rangel le rangel le plus proche « Figure 7.12 » ayant une amplitude de gradient dépassant un seuil déterminé, dans une plage de distance spécifique dans les deux directions.

```
public void compute_lambdas() {
    double force;
    double maxforce = 0.0;
    double minr = config.getRegMin();
    double maxr = config.getRegMax();

    for (int i = 0; i < NPT; i++) {
        force = Math.sqrt(deplace[i].x * deplace[i].x + deplace[i].y * deplace[i].y);
        if (force > maxforce) {
            maxforce = force;
        }
    }

    for (int i = 0; i < NPT; i++) {
        force = Math.sqrt(deplace[i].x * deplace[i].x + deplace[i].y * deplace[i].y);
        lambda[i] = maxr / (1.0 + ((maxr - minr) / minr) * (force / maxforce));
    }
}
```

Figure 7.12 – Calcul d'amplitude de gradient dépassant un seuil déterminé

Initialement, les paramètres de régularisation sont élevés pour garantir au modèle finesse et capture de la forme globale de l'objet. A chaque itération « *Figure 7.13* », le déplacement moyen D de chaque point de contour est calculé une augmentation de cette valeur d'une itération à l'autre indique que le snake se poursuit « *Figure 7.14* » vers les bords et aucun paramètre ne sera changer, si la distance D diminue, les paramètres de régularisation sont diminués pour l'itération suivante.

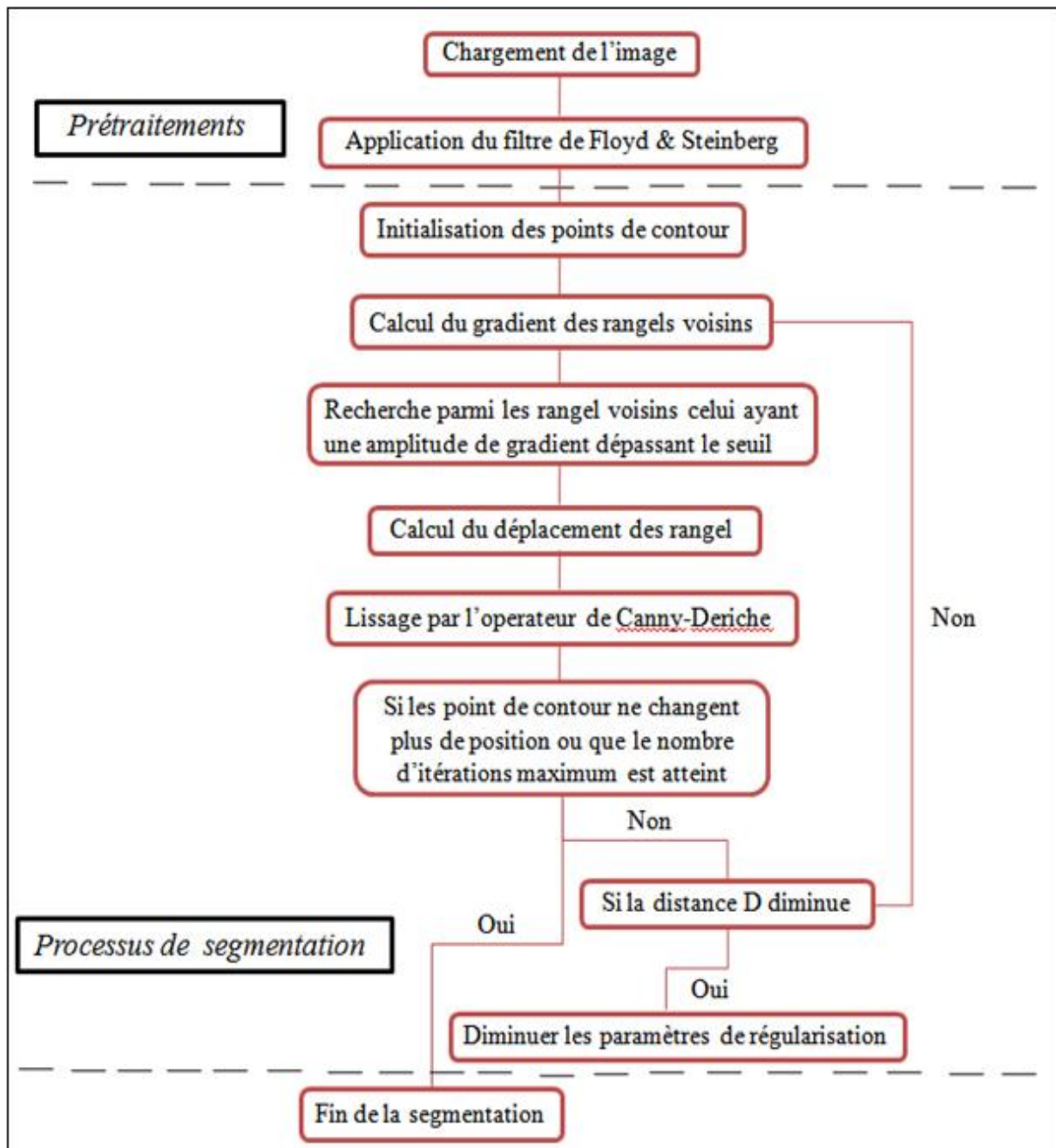


Figure 7.13 – Algorithme du contour actif adaptatif

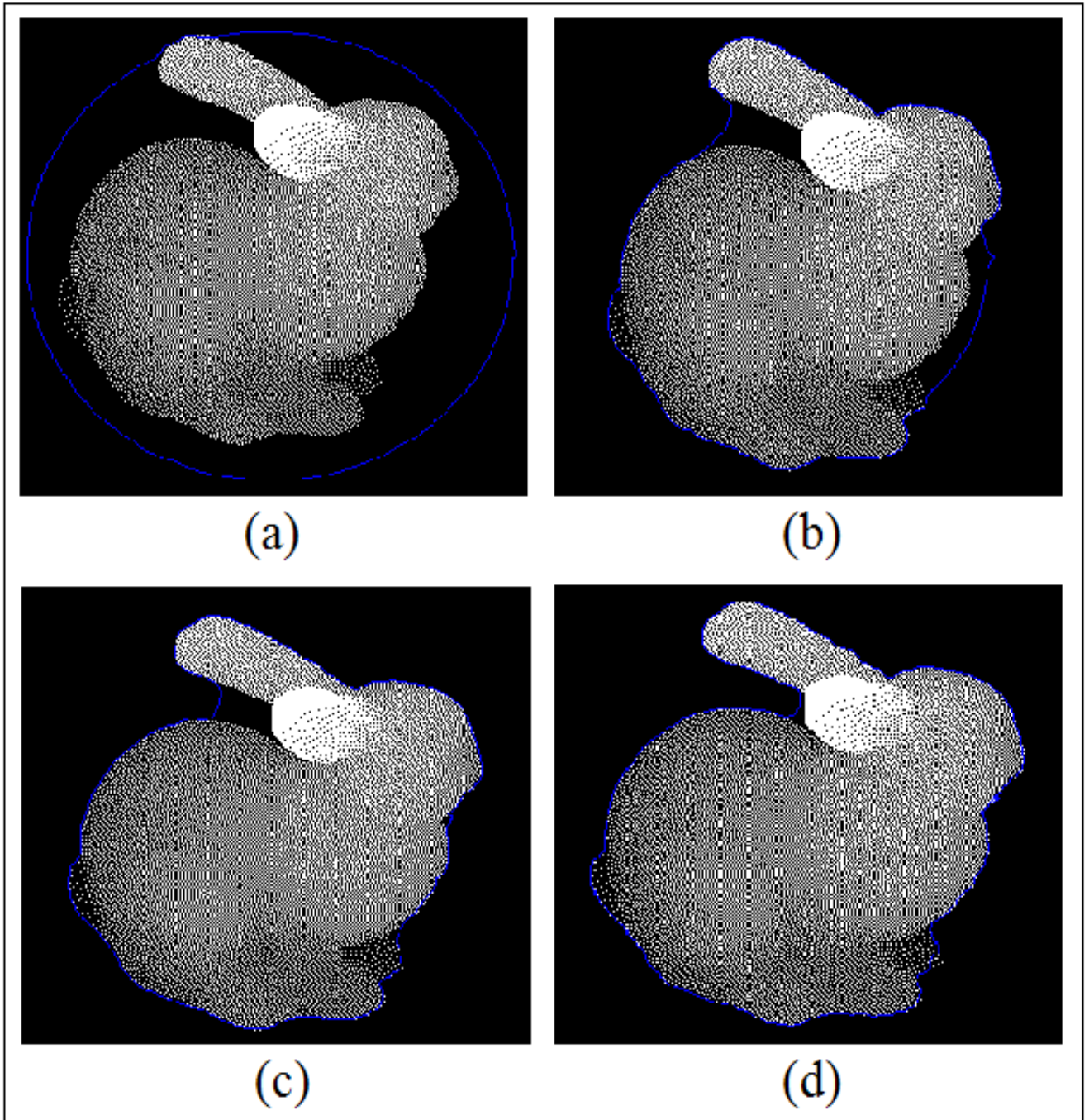


Figure 7.14 – Déformation du contour actif adaptatif, (a) contour initial, (b) contour après 30 itérations, (c) contour après 50 itérations, (d) contour après 100 itérations.

Ce mécanisme permet au snake d'entrer progressivement dans les cavités. En parallèle, le paramètre de lissage contrôlé par l'opérateur de Canny-Deriche est également adapté selon l'évolution du snake. Le lissage est important pendant les premières itérations pour éviter la sensibilité au bruit, puis y est réduit et une nouvelle image de gradient est calculée à chaque itération. Le déplacement final résulte d'un compromis entre le déplacement complet vers le

contour le plus proche le long de la normale et la satisfaction de contraintes géométriques incorporées dans l'énergie interne. L'algorithme est stopper dans le cas où aucun point du contour ne change de position ou si le nombre d'itérations maximum est atteint. Pour continuer notre expérimentation, nous appliquons le modèle déformables (snake adaptatif) aux 82 images de profondeur tout comme pour le contour actif classique nous exécuterons l'algorithme pour des images sans cavités et des images comportant des cavités. Les résultats obtenus par cette méthode sont quasiment les mêmes que ceux obtenus par le snake classique « *Figure 7.15* » pour les images simple sans cavités.

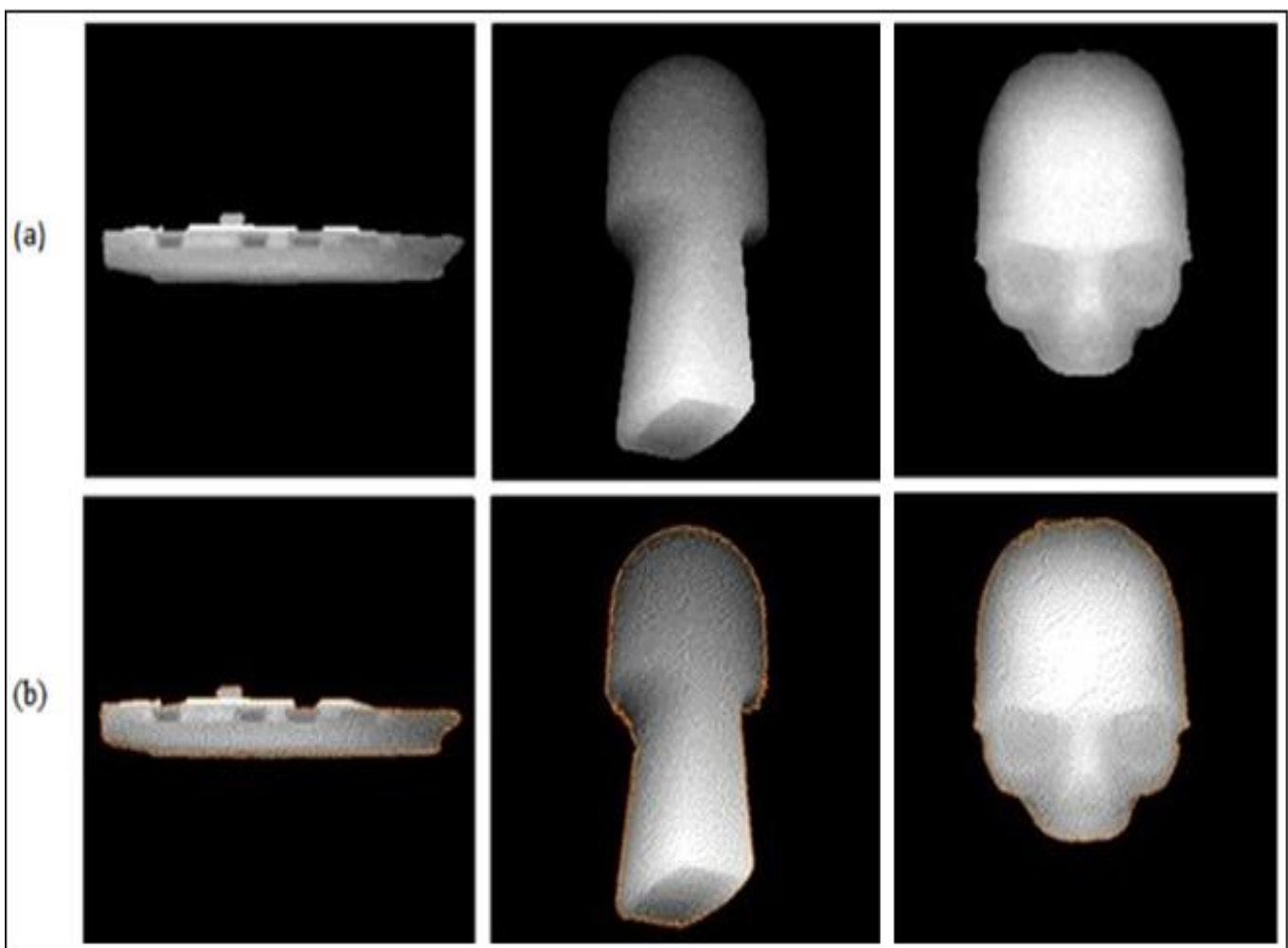


Figure 7.15 – Segmentation d’images de profondeur simple par contour actif adaptatif: (a) Image de profondeur à segmenté (b) Segmentation obtenue par le snake adaptatif (c)

Par contre les résultats obtenus pour les images de profondeur comportant des cavités sont nettement meilleurs pour le snake adaptatif, on constate « *Figure 7.16* » que les points de contour ont parfaitement épousés les bords de la forme à segmentée.

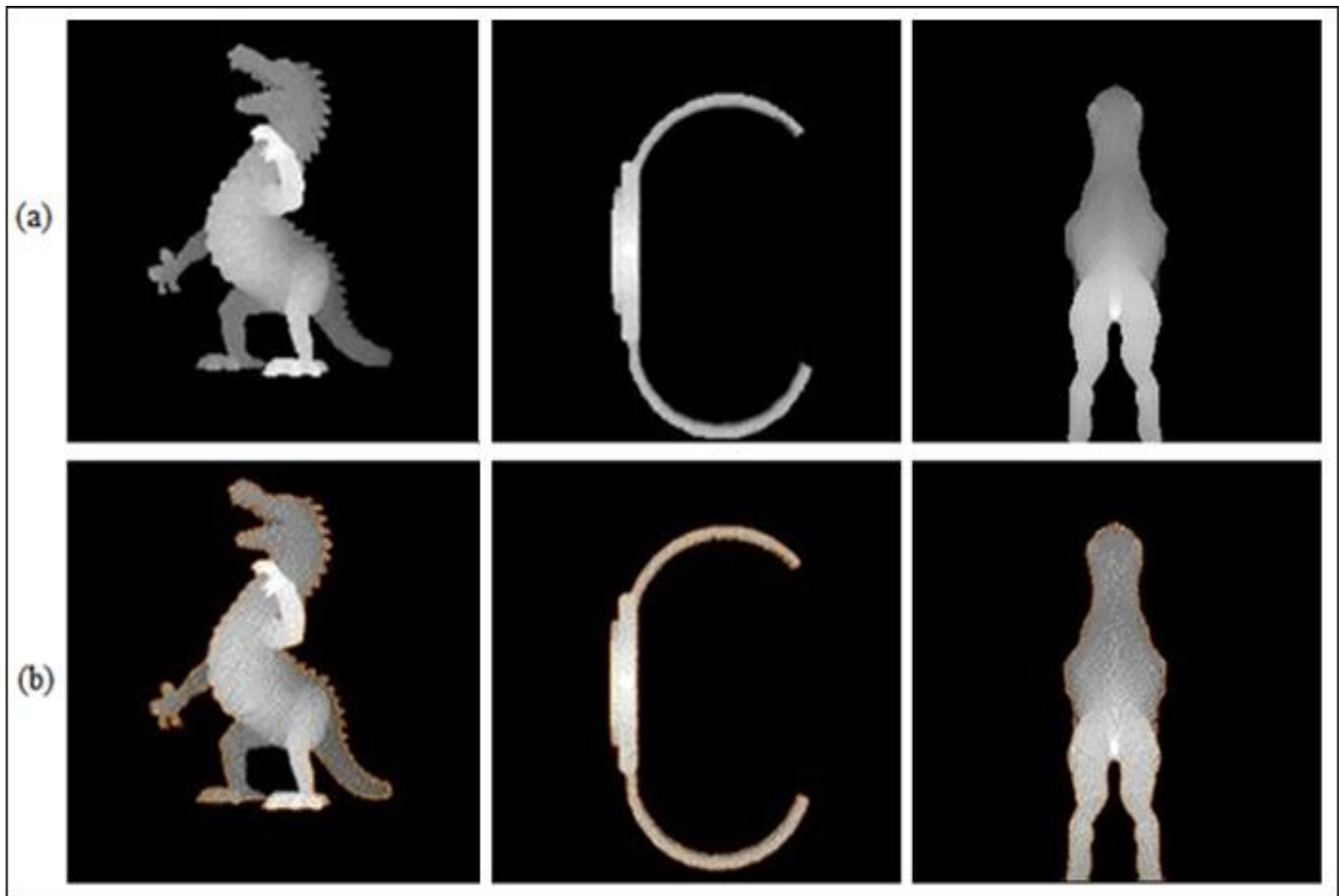


Figure 7.16 – Segmentation d’images de profondeur avec cavités par contour adaptatif classique: (a) L’image de profondeur (b) Segmentation obtenue par le snake adaptatif

Comme pour le contour actif classique, nous avons utilisé le Framework de comparaison pour évaluer les résultats de segmentation obtenus par l’application du contour actif adaptatif avec ceux obtenus par la segmentation manuelle, les résultats sont présentés dans la Tableau 7.2.

Tableau 7.2 – Taux de bonne segmentation obtenus par l’algorithme de snake adaptatif

Image de profondeur	Nombre de points de Contour généré par la segmentation manuelle	Nombre de points de contour mal placés	Taux de bonne segmentation
Dinopet5	6794	175	84.74 %
Ente1	6563	155	84.83 %
Copter2	4466	88	85.45 %
Dinopet1	4373	106	85.92 %
Bunny2	5029	13	98.46 %
Ente3	4652	12	98.52 %
Deoflach2	4769	9	98.97 %
Ente4	5828	4	99.61 %
Baljoint1	5359	97	99.65 %
Ente2	6598	2	99.79 %

7.6.3 Les Level Sets

La technique de Level Set est une catégorie importante de techniques de segmentation d'images modernes basés sur les équations aux dérivées partielles PDE, qui est une évaluation progressive des différences entre les pixels voisins pour trouver les contours des objets, idéalement, l'algorithme converge à la limite de l'objet où les différences sont le plus élevé, la caractéristique remarquable de ces méthodes est que les contours peuvent fractionner ou fusionner selon les changements de la fonction de réglage. Cette méthode permet aussi de détecter plus d'une frontière simultanément en plaçant plusieurs contours initiaux, mais la vitesse de convergence sera plus lente que les autres méthodes de segmentation. Un défi majeur dans les contours actif est le bon positionnement du contour initial. Etant donné que le contour se déplace soit vers l'intérieur ou vers l'extérieur, la segmentation obtenue dépend donc fortement de l'emplacement initial des points de contour. Au départ « *Figure 7.17* », on commence avec une courbe fermée et lui permettre de se déplacer perpendiculairement à elle-même à une vitesse prescrite, nous décrivons cette courbe en utilisant une forme paramétrique explicite (comme pour les snake), mais cette forme paramétrique provoque des difficultés lorsque les courbes doivent subir le fractionnement ou de fusion, pour remédier à cette difficulté, lors de l'évolution de la forme souhaitée le contour actif implicite prend l'interface originale et l'incorpore dans la fonction scalaire de plus grande dimension définie sur toute l'image. La fonction de réglage de niveau est une fonction de distance signé, à partir de l'ensemble de niveau zéro, ceci donne une courbe fermée C_0 , si le pixel se trouve sur la courbe la fonction est égale à zéro, sinon, la fonction renvoi la distance minimale entre le pixel et la courbe (la distance est négative pour les pixels à l'intérieur de C_0 et positive pour les pixels à l'extérieur C_0). Cette fonction varie avec le temps et l'espace et évolue en utilisant une équation aux dérivées partielles (PDE)

$$\Psi_{t+1} + F|\nabla\Psi_t| = 0 \quad (1) \quad (7.4)$$

Où F est la vitesse en direction de la normale N à la courbe, nous ne considérons l'évolution de la fonction dans une direction perpendiculaire à elle-même

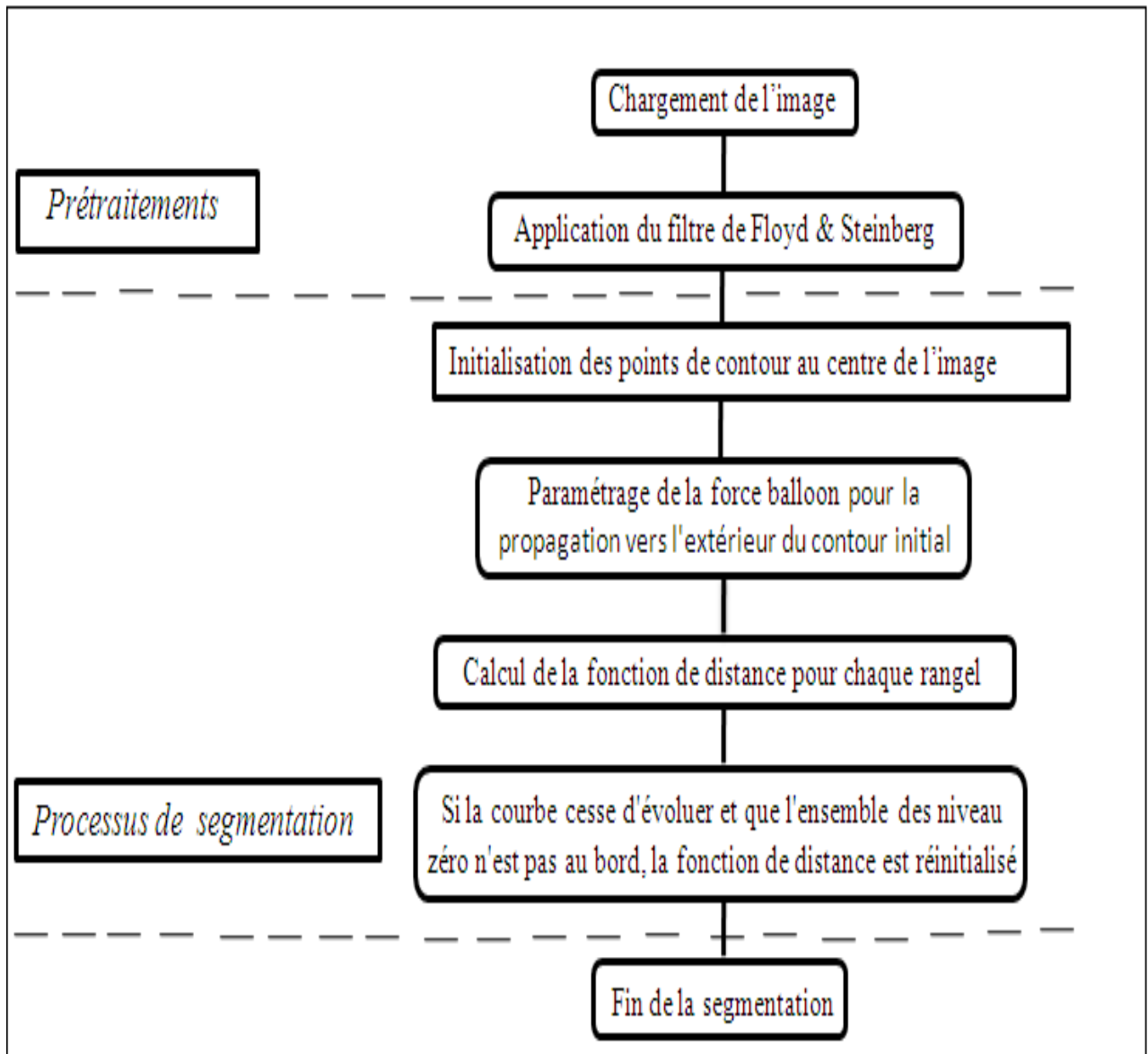


Figure 7.18 – Algorithme des Level sets

Cette vitesse F « Figure 7.18 » est composé de trois termes, un terme constant (similaire à la force de l'inflation utilisé dans les modèles déformables), un terme dépendant de la courbure locale à chaque point et un terme dépendant de l'image (dans notre cas, les bords de l'image)

$$\psi_{n+1} = \psi_n - dt * k_I(x, y) * (U_n - \varepsilon K) * |\nabla \psi_n| \quad \varepsilon \in [0, 1] \quad (7.5)$$

U_n étant la fonction définissant la zone ou l'objet à rechercher, K la courbure locale à chaque instant t et K_1 le critère d'arrêt.

```

if(e.getSource().equals(changeSpeedFunction))
{
    NarrowBandLevelSet ls = getSelectedLevelSet();

    if(ls == null)
        return;

    try{
        SpeedFunction s = askForSpeedFunction(ls.getPhi());
        Solver solver = new NormalDirectionSolver(s);
        ls.getNarrowBand().setSolver(solver);

    }catch(Exception ex){}
}

```

Figure 7.18 – Calcul de la vitesse F de l’algorithme des Level Set

Pour finir notre expérimentation, nous appliquons le modèle déformables (Level set) aux 82 images de profondeur, comme on peut le voir dans les figures « Figure 7.19 » et « Figure 7.20 ». Les résultats visuels sont à peu près visuellement les mêmes que pour ceux de l’algorithme des contours actifs classique et de l’algorithme des contours actifs adaptatifs, cette méthode offre donc une segmentation parfaite des images simples sans cavités.

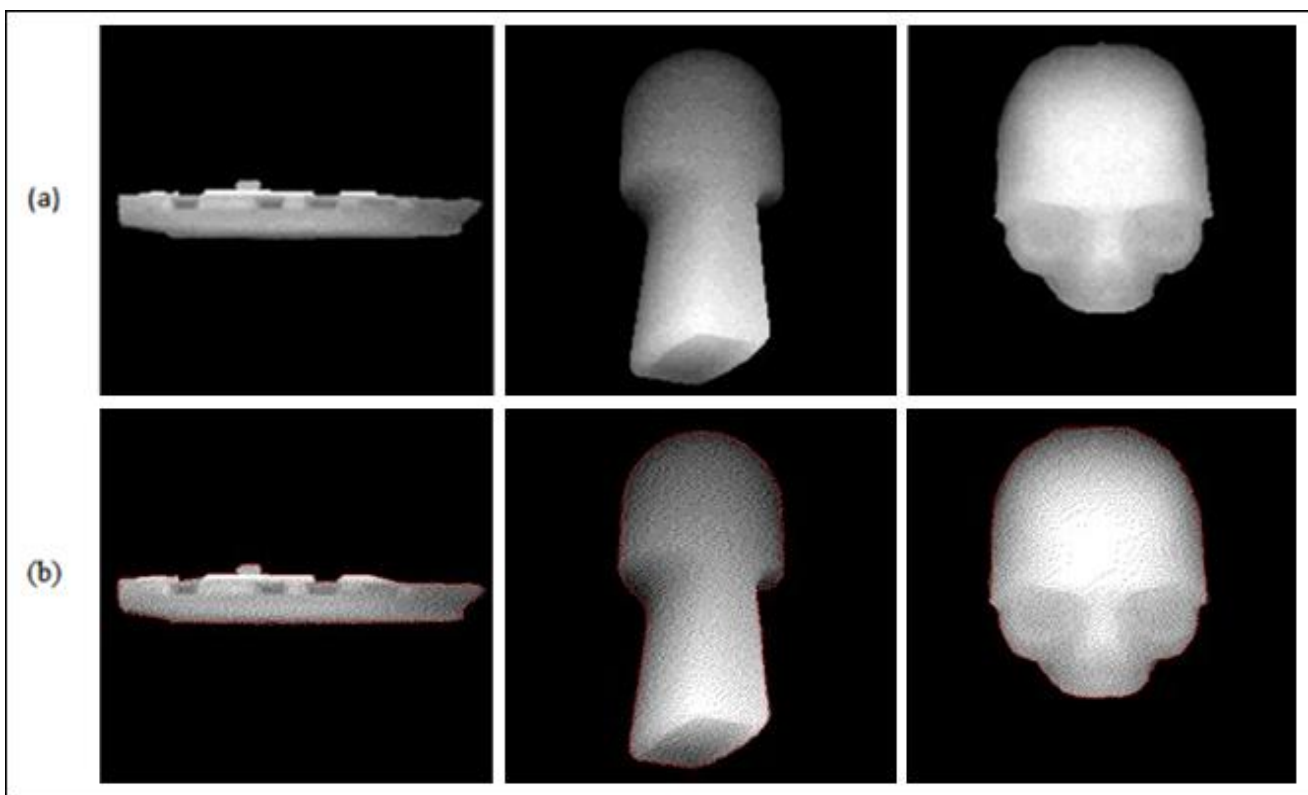


Figure 7.19 – Segmentation d’images de profondeur simple par Level set: (a) Image de profondeur à segmenté (b) Segmentation obtenue par les Level set (c)

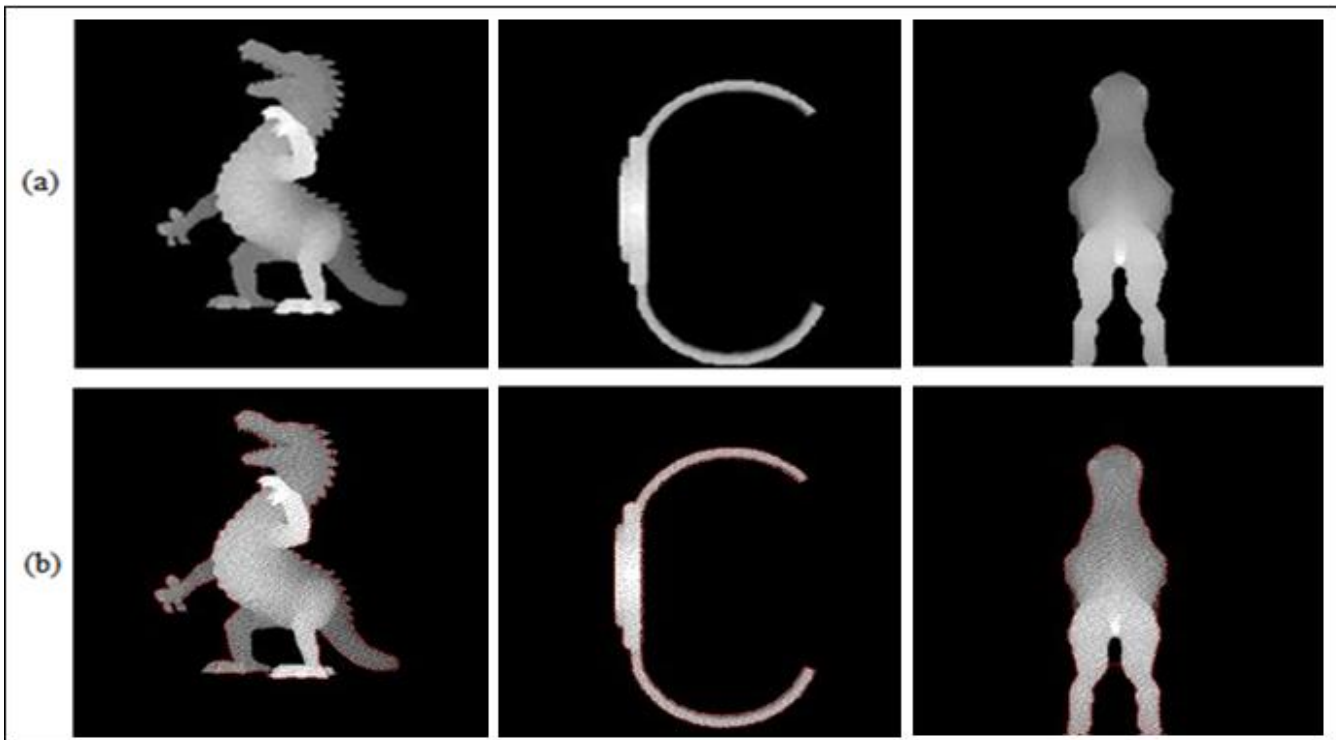


Figure 7.20 – Segmentation d’images de profondeur avec cavités par Level set
(a) L’image de profondeur (b) Segmentation obtenue par les Level set

Comme pour le contour actif classique et adaptatif, nous avons utilisé le Framework de comparaison pour évaluer les résultats de segmentation obtenus par l’application des Level Set avec ceux obtenus par la segmentation manuelle, les résultats sont présentés dans la Tableau 7.3.

Tableau 7.3 – Taux de bonne segmentation obtenus par l’algorithme des Level Set

Image de profondeur	Nombre de points de Contour généré par la segmentation manuelle	Nombre de points de contour mal placés	Taux de bonne segmentation
Bunny1	5468	360	83.67 %
Dinopet2	9363	513	92.92 %
Fighter3	6215	396	93.13 %
Fighter4	6873	381	93.89 %
Dinopet5	8347	357	94.74 %
Fighter1	9122	447	94.78 %
Carrier1	4194	87	97.91 %
Agfa2	4935	91	98.09 %
Ente4	5908	65	98.84 %
Baljoint3	4121	43	98.89 %
Baljoint1	5456	54	98.99 %
Baljoint2	5130	23	99.54 %

7.6.4 Comparaison et analyse numérique des résultats obtenus

Les taux de bonne segmentation [Khaldi & al, 2014] ont été calculés pour les 82 images segmentées et ceux par les trois techniques (Level set, Adaptive Snake et Classique snake). Le résultat global (en moyenne) est de 96,01% de bonne détection pour les images segmenté via l'algorithme des Level sets, suivie par l'algorithme de contour actif adaptatif « Figure 7.21 » avec un taux de bonne segmentation de 95,17%, pour finir l'algorithme du contour actif classique offre 82,60% de bonne détection(ce qui est beaucoup moins que les résultats obtenus par les deux autres approches).

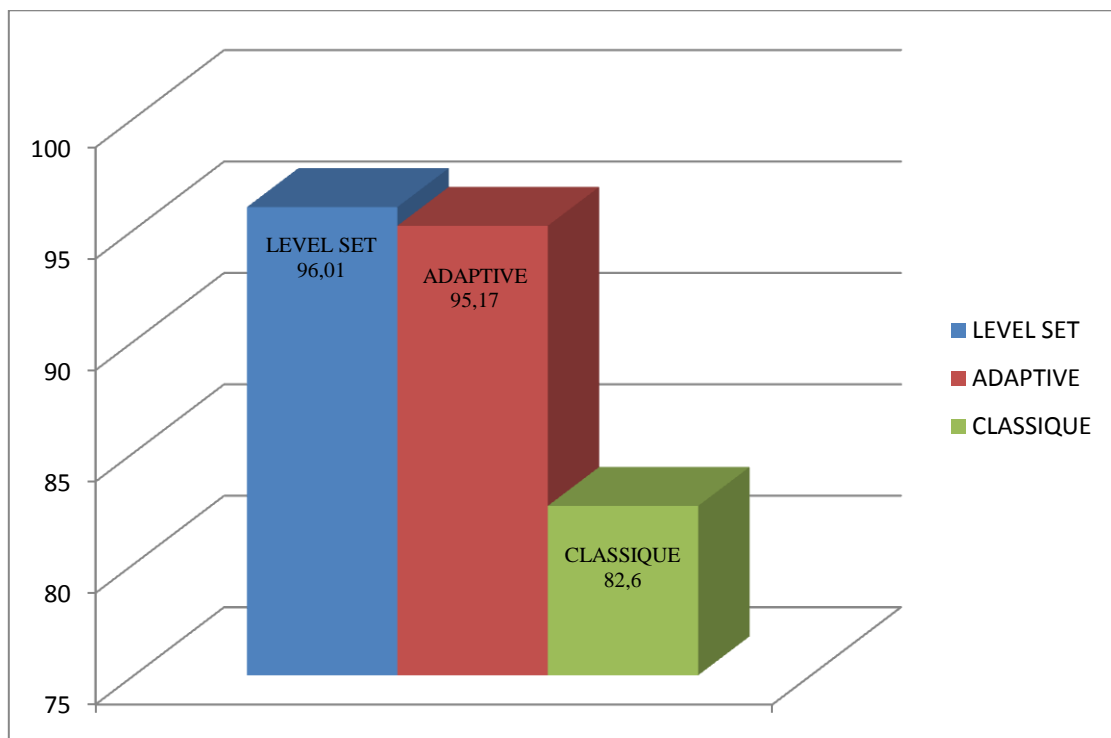


Figure 7.21 – Taux de bonne détection pour les trois algorithmes

Nous avons analysé séparément les taux de bonne segmentation pour les images de profondeurs simples et ceux des images de profondeurs comportant des cavités et ceux pour les trois algorithmes appliquer « Figure 7.22 ».

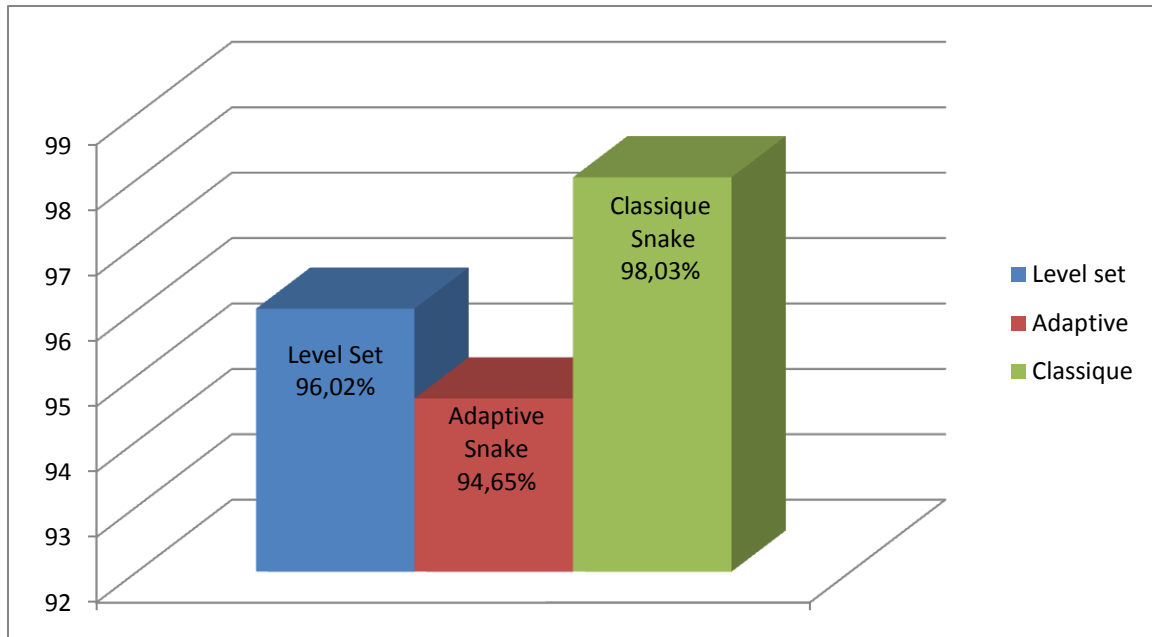


Figure 7.22 – Taux de bonne détection pour les images simples

Nous remarquons que pour les images simples le contour actif classique apporte de très bon résultat avec 98% de bonne segmentation en moyenne bien plus que les contours actifs adaptatifs et Level set à respectivement 94% et 96%. Cependant pour les images de profondeur avec cavités « *Figure 7.23* » nous remarquons que les Level set et Les snake adaptatif offre de bon résultats de segmentation (respectivement 96% et 95.69%) contrairement au contour actif classique qui ne totalise que 67% de bonne segmentation.

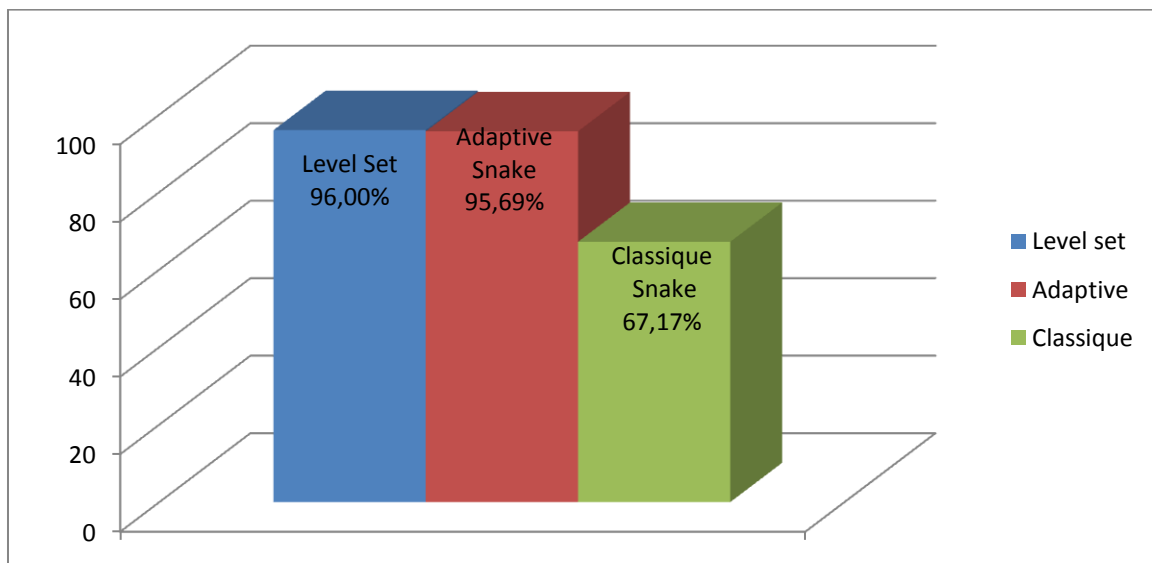


Figure 7.23 – Taux de bonne détection pour les images avec cavités

Nous avons analysé les taux de bonne segmentation obtenus et retenu le meilleur algorithme de segmentation pour chaque image de profondeur de notre base, nous constatons que l'algorithme des Level sets segmente aussi bien les images simples que les images complexe comportant des cavités « *Figure 7.24* » il à été retenu comme meilleur algorithme pour 37 images de profondeur sur les 82 images de profondeur utilisées. Par contre, en analysant les résultats des contours actifs classiques et adaptatifs, ils offrent certes de bons résultats de segmentation pour les images simple mais pour le résultat global ils n'ont été retenus que pour 23 images pour l'algorithme du contour actif adaptatif et 22 images pour le contour actif classique et ce pour les 82 images de profondeur utilisées

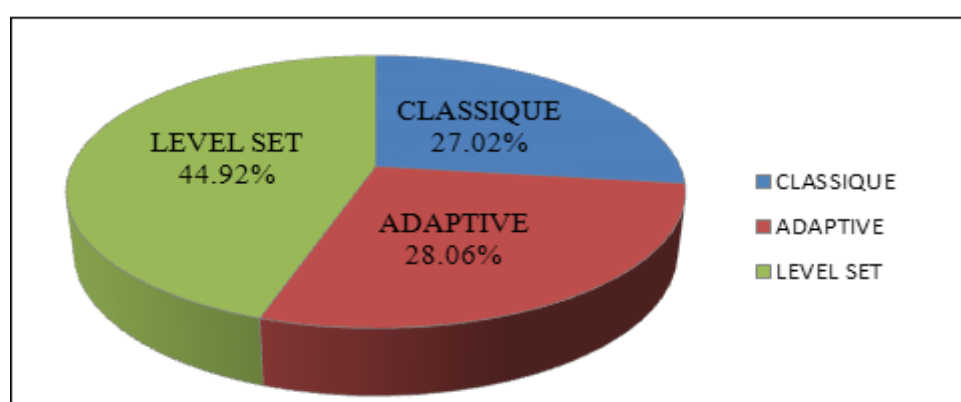


Figure 7.24 – Taux de bonne détection pour les images avec cavités

Nous avons choisi huit de ces 82 résultats où l'écart est supérieur à 3% (comparé au meilleur taux de segmentation) et comme nous pouvons voir les résultats dans la « *Tableau 7.4* », cela confirme notre comparaison visuelle des images avec cavités. Le plus mauvais résultat obtenu par l'algorithme des Level sets est l'image «Pitt4» avec 92,17% de bonne détection. Contrairement au snake classique qui donne de très mauvais résultats pour les images avec cavité comme on peut le voir dans « Dinopet3 » avec un résultat de 76,48% et « Watch1 » 54,55%, ce qui est un écart important par rapport aux autres résultats obtenus par les Level sets (respectivement 96,16% et 97,05%) et le snake adaptatif (respectivement 94,67% et 85,21%).

Tableau 7.4 – Taux de bonne segmentation avec un écart supérieur à 3%

	Fighter5	Agfa3	Pitbull3	Manta2	Dinopet3	Watch1	Pitt4	Kroete2
Snake	98.57 %	99.19 %	80.74 %	89.96 %	76.48 %	54.55 %	76.16 %	85.18 %
Level set	95.12 %	95.18 %	98.07 %	97.44 %	96.16 %	97.05 %	92.17 %	95.01 %
Adaptive	94.24 %	98.70 %	92.36 %	96.05 %	94.67 %	85.21 %	81.21 %	97.89 %

7.7 Conclusion

Dans ce chapitre nous avons proposé une segmentation des images de profondeurs (images simples et images complexe) en utilisant trois techniques de modèles déformables. Bien que les calculs numériques soient rapides, certains problèmes liés aux méthodes du contour actif classique et adaptatif peuvent apparaître selon le contexte dans lequel elle est utilisée, si un objet se trouve sur une zone de l'image très hétérogène, il sera difficile pour le contour actif de détecter les contours de ce premier. Les contours actifs s'avèrent très pertinents pour isoler des formes convexes régulières et idéales. Pour l'extraction de formes comportant beaucoup de concavités franches, il faudra diminuer l'impact du non respect de cette contrainte pour privilégier l'écartement entre les points ou même l'énergie externe. Un autre aspect non négligeable mais non directement lié à la méthode des snakes réside dans l'image à analyser. En effet, celle-ci doit être la plus pure possible pour effacer au maximum le bruit qui pourrait amener à un pic de gradient sur lequel le snake pourrait rester accroché. Concernant le Level sets, ce modèle est très polyvalent car il peut soit reproduire le comportement classique d'un snake avec des valeurs élevées de régularisation, mais aussi de segmenter des structures très fines et complexes, pendant les premières itérations, le modèle est assez lisse, mais n'entre pas dans les cavités, après quelques itérations, le modèle est moins lisse mais entre parfaitement dans les cavités.

Chapitre 8

Proposition d'une approche multi-agent pour la segmentation d'images de profondeur

8.1 Introduction

Les méthodes traditionnelles de segmentation d'image utilisent une approche séquentielle qui consiste à traiter l'image rangel après rangel. Cependant il existe d'autres approches comme les systèmes multi-agents qui permettent de traité l'image d'une manière non séquentielle. Un système multi-agent consiste en un certain nombre d'agents qui interagissent entre eux. Un agent est capable de percevoir son environnement, d'agir de manière autonome, d'interagir avec les autres agents, d'apprendre de ses expériences et de s'adapter à son environnement. Dans notre proposition nous aimerions utiliser un système multi agent pour segmenter des images de profondeur en effectuant une croissance de régions.

8.2 approche multi-agent pour la segmentation de profondeur

La méthode que nous proposons dans ce qui suit est une extension de la méthode d'accroissement de région classique où l'on introduit une architecture multi agent pour effectuer la segmentation. Dans cette architecture nous aurons deux types d'agents :

- **Un agent contrôleur** : On peut définir l'agent contrôleur comme étant un super agent, qui se chargera de la création et du lancement d'exécution des autres agents.
- **Les agents de croissance** : Qui seront des agents dont la tâche principale est l'exécution de la croissance de région. Leur création est demandée par le contrôleur.

Un agent possède une vie limitée, avec un début et une fin, il sera créé au début par l'agent contrôleur et pourra être détruit également à tout moment. Durant son cycle de vie un agent aura trois statuts possibles, il sera « Actif » quand il exécutera les traitements dont il aura la charge, l'agent pourra passer en statut « Suspendu » lorsqu'il sera en attente d'informations qui seront produites par un autre agent. Lorsque l'agent est stoppé dans son traitement (la croissance de régions) par la condition d'arrêt, il se métra dans l'état « Inactif ».

8.2.1 Processus d'initialisation

Tout d'abord, il y aura une phase d'initialisation des agents et leurs dispositions dans l'image de profondeur. Cette étape de sélection sera donc cruciale, car c'est de l'ensemble des rangels sélectionnés que dépend le potentiel de choix de l'algorithme et plus généralement ce que l'on pourrait appeler le « champ de vision » de la méthode de construction. Cette initialisation se fera d'une manière automatique et sera réalisée sur les minima locaux de l'image du gradient tout en maintenant une certaine distance entre deux agents pour ne pas avoir d'agents trop voisins.

8.2.2 Processus de segmentation

Durant la deuxième phase « la phase d'exécution de la segmentation », chaque agent se déplace dans l'image du point courant à celui de son voisinage. Chaque agent possède un ensemble de points (voisinage) représentant les prochains à explorer. De même chaque agent met aussi à jour des données statistiques sur les rangels déjà agrégés lui permettant de calculer à tout moment la moyenne et l'écart type des niveaux de gris des rangels qui lui appartiennent. Le critère choisi est celui qui privilégie les rangels dont l'intensité est la plus proche de la moyenne. Tout rangel agrégé est marqué comme appartenant à l'agent qui le marque et celui-ci deviendra donc son propriétaire.

8.2.3 Processus de communication

Un conflit de marquage surviendra lorsqu'un agent se rendra compte qu'il veut marquer un rangel qui est déjà pris par un autre agent. Dans ce cas, les deux agents doivent négocier. Nous aurons deux modes de communication, une communication indirecte qui est une communication par signaux via l'environnement, et une communication directe qui procède à un échange de messages entre les agents. Le premier mode sera assuré par le marquage des rangels permettant à un agent de se rendre compte qu'un rangel donné est déjà pris. La communication directe serait utilisée dans la phase de négociation d'un rangel litigieux. Nous aurons trois messages :

- Un message de type « Discussion » : Il sera préparé et envoyé par tout agent qui veut marquer un rangel. Tout agent qui veut marquer un rangel doit s'assurer que celui-ci

est libre. Sinon il doit le négocier avec son propriétaire courant. Il a alors la responsabilité d'initier la discussion. Si un agent A veut envoyer un message de discussion à un agent B qui est le propriétaire du rangel R qui fait l'objet de la réclamation, il doit s'assurer que $(A-R) < (B-R)$. A étant la valeur accordée à l'agent A et B la valeur accordée à l'agent B. Chaque message de ce type comporte comme information un point représentant le rangel sur lequel porte la discussion, la valeur donnée des deux agents, la référence de l'agent expéditeur de la requête ainsi que la référence de l'agent destinataire du message.

- Message de type «RangelDonné» : Suite à la réception d'un message de discussion, un agent doit d'abord vérifier qu'il est le propriétaire du rangel sur lequel porte la discussion. Si tel est le cas, il réévalue sa valeur et la compare avec celle contenue dans le message. Il doit céder le rangel dans le cas où sa valeur réévaluée est inférieure à celle reçue dans le message. Dans ce cas, il doit aussi effectuer des mises à jour visant à supprimer ce rangel de ses rangels agrégés. Il a alors la responsabilité de modifier le marquage du rangel considéré en positionnant l'autre agent comme nouveau propriétaire. Enfin, pour informer l'autre agent qu'il est le nouveau propriétaire du rangel, il doit préparer et envoyer un message de type «RangelDonné » afin que ce dernier puisse considérer ce rangel comme étant le sien et continuer son exploration en tenant compte de ce fait. Tout message de ce type comporte comme information le rangel discuté et les références de l'émetteur et du récepteur.
- Message de type « Libre » : Ce type de message est préparé et envoyé par tout agent qui reçoit un message de type discussion et qui se rend compte qu'il n'est plus le propriétaire du rangel sur lequel porte la discussion. Cette situation peut survenir par exemple dans le cas où deux agents A et B envoient un message de discussion à un troisième autre agent sur un même rangel. Si ce troisième agent traite en premier le message de A et lui lègue le rangel, au moment de traiter la requête de B il se rendra compte qu'il n'est plus le propriétaire du rangel. Comme le message de type «RangelDonné», tout message de type « Libre » comporte comme information le rangel qui fait l'objet du message et les références de l'émetteur et du récepteur.

Avant tout intégration de rangel, l'agent doit vérifier que celle-ci est possible. Le cas échéant, les données statistique sont mises à jour, la valeur du rangel est évaluée et le rangel est

marqué comme appartenant à l'agent. Un seuil sera alors fixé, la valeur de ce seuil permettra de limiter la propagation de la croissance de région et permettra aussi de s'assurer que les rangels agrégés en fin de croissance ne soient pas mauvais. Généralement, les rangels conquis dans les derniers moments du processus de croissance de région parmi ceux qui contribuent le plus à augmenter l'écart type des rangels de la région. Un agent s'arrête après cinq rangels cédés consécutivement ce qui voudrait dire que la région qu'il représente ne peut plus s'accroître « *Figure 8.1* ». Lorsque tous les agents s'arrêteront, aucun accroissement ne sera possible et nous aurons la segmentation finale.

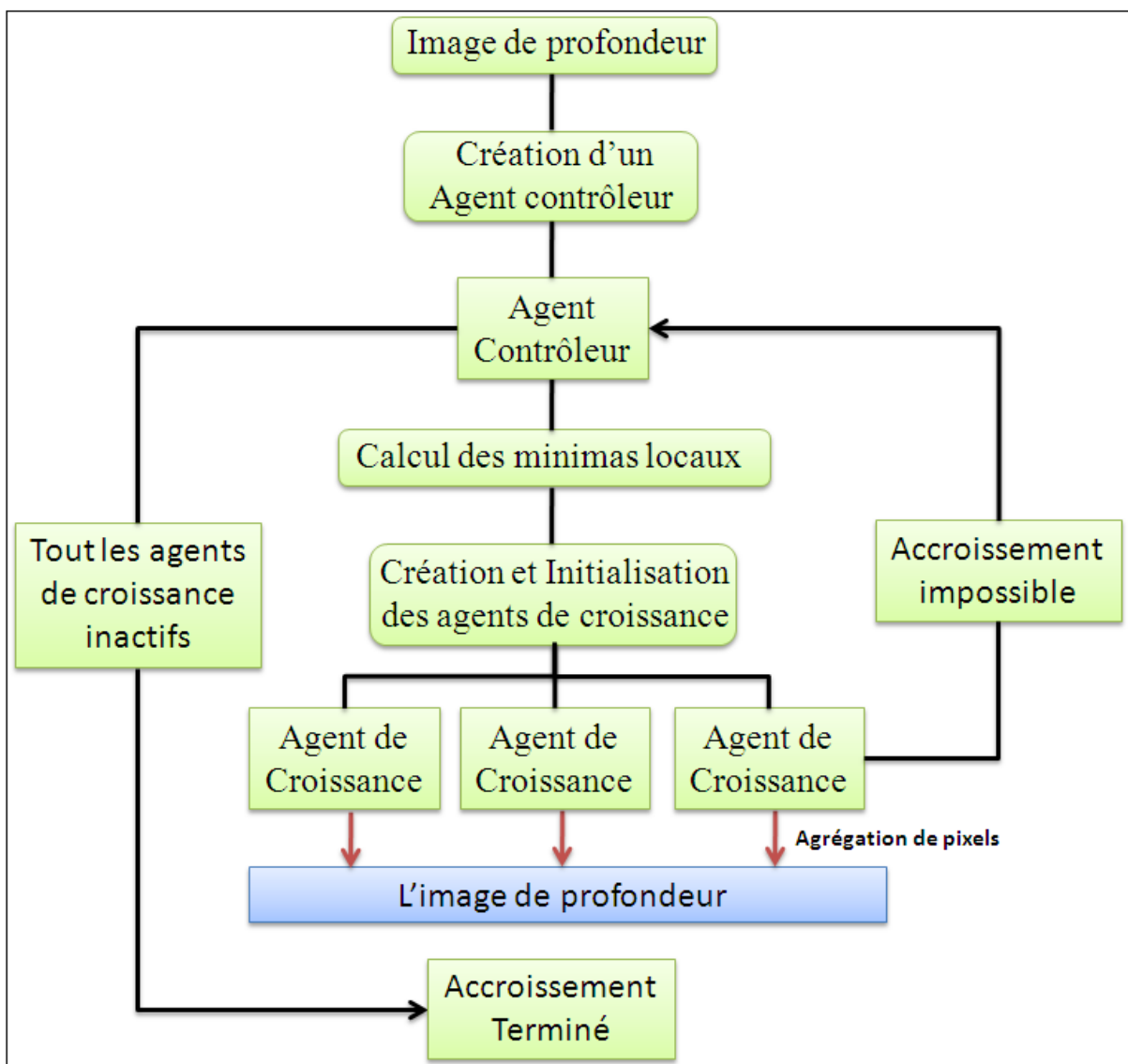


Figure 8.1 – Approche multi agents pour la segmentation d'images de profondeur par accroissement de régions

8.3 Conclusion

Nous avons dans ce chapitre proposé une architecture multi-agent pour segmenter une image de profondeur en utilisant le principe de la méthode d'accroissement de régions. De part leur définition, l'utilisation des agents offre une souplesse dans la segmentation par rapport aux approches séquentielles. En effet, répartir les agents sur la surface d'une image peut présenter des avantages. Cependant, un agent en tant que composant logiciel agit de façon autonome selon l'implémentation de son comportement.

Conclusion générale

Ce travail de thèse a été consacré au problème de la segmentation des images de profondeur par contours actifs. Nous avons d'abord dans le premier chapitre présenté l'image de profondeur ainsi que ces diverses propriétés, telle la profondeur et la cohérence spatiale d'une image de profondeur. Nous avons ensuite abordé les types d'images de profondeurs existants et leurs représentations graphiques possibles, puis, nous avons procédé à la présentation des différents mécanismes de capture d'images de profondeur et nous avons constaté à quel point ces scanners ont évolué au fil des années.

Nous avons ensuite passer en revue, les méthodes élémentaires, puis les méthodes formant l'état de l'art de la segmentation d'images de profondeur. Les éléments traités ne sont pas fondamentalement nouveaux mais, exposés sous un éclairage différent et peut-être plus approfondi par rapport à ce qu'on trouve ailleurs dans la littérature, ils sont d'un intérêt certain. Nous en récapitulons l'essentiel. Tout d'abord, nous avons mis en évidence le fait que la segmentation des images de profondeur comporte deux aspects essentiels, le critère d'homogénéité (modèle de surface), d'une part, et la méthode réalisant la segmentation, d'autre part. Dans les méthodes rencontrées dans la littérature, le critère d'homogénéité n'est pas toujours explicite. Nous avons cependant pu classer la quasi-totalité des méthodes suivant quatre critères d'homogénéité: type de courbure, surface plane, surface algébrique, et continuité d'ordre 1. Nous avons présenté les différents détecteurs de discontinuité, avec leurs caractéristiques propres. Notre attention s'est portée sur les détecteurs de taille minimale servant à la détection conjointe des discontinuités d'ordre 0 et de celles d'ordre 1.

Nous avons ensuite présenté les différentes méthodes de contours actif existantes et les différentes approches pour définir un critère de segmentation pour les contours actifs, de l'approche purement "contours" aux approches plus récentes qui intègrent des termes de régions dans les fonctionnelles à minimiser. Notre travail considéré l'utilisation de contour actif pour un problème important de la vision par ordinateur, à savoir , la segmentation d'image de profondeur , nous avons appliqué trois méthodes de détection de contours (contour actif classique, adaptatif et Level set) basé sur les modèles déformables à des images de profondeurs , comme mentionné dans l'introduction, la motivation de notre travail a été

d'appliquer le modèle déformable " Snake " qui n'a jamais été fait auparavant dans la segmentation d'images de profondeur .

Les contours actifs classiques ont l'avantage de réduire considérablement le nombre de points de contours (comparé aux autres méthodes de contour actifs) ainsi que le temps d'exécution de cet algorithme qui est rapide. Après nos expérimentations nous avons remarqué que le contour actif apporter de bons résultats de segmentation sur des images de profondeurs simples. Cependant cette méthode a quelques limitations et ne peut pas détecter des discontinuités dans les cavités, les contours actifs sont très utiles pour isoler des formes convexes régulières, cependant, ils ont quelques inconvénients. Tout d'abord, la fonction n'est pas intrinsèque, elle dépend du paramétrage du contour (Le choix des constantes est difficile), le contour actif classique fournir une localisation précise des bords que si le contour initial est donné suffisamment près des bords, car il n'utilise que la information locale le long du contour (estimation d'une bonne position de contours initiaux sans connaissance préalable est un problème difficile). De plus la géométrie de l'objet à détecter n'est pas prise en compte. La contrainte de régularité est forte et ne permet pas de détecter les zones concaves du contour. Elle interdit également les changements de topologie : un seul objet peut être segmenté. Une telle régularité pousse le contour à rétrécir, voire à disparaître s'il ne rencontre pas de zones de gradient suffisamment fort. Ce sera le cas, par exemple, si le contour initial est à l'intérieur de l'objet à segmenter, et trop éloigné des frontières de l'objet.

Par contre, l'application des Level sets et du contour actif adaptatifs ont donné de bons résultats de segmentation, aussi bien pour les images de profondeur simples que pour les images complexes. Les Level sets comme les contours actifs adaptatifs nous ont permis de segmenter une panoplie de formes très complexes comportant de grandes variations de courbures. Leurs capacités à changer de topologie les a rendues très intéressants. Mais, contrairement au contour actif adaptatif l'algorithme du Level set souffre d'un temps de calcul beaucoup plus important et le nombre de points de contour générés et considérablement plus important.

Dans cette thèse nous avons également proposé une approche se basant sur le paradigme multi-agent qui sera adopté à la méthode de croissance de régions. L'utilisation des agents nous offrira une certaine souplesse dans la segmentation par rapport aux approches séquentielles. En effet, répartir les agents sur la surface d'une image peut présenter les avantages d'une étude répartie tout en ayant un contrôle global grâce à la communication et

de la coopération entre les agents. L'interaction entre les agents fournit une solution alternative pour la segmentation d'images, par rapport aux méthodes de segmentation basées sur des modèles complexes et coûteux en temps de calcul.

Perspectives

Dans nos futurs travaux plusieurs idées peuvent se concrétiser.

- Le prolongement immédiat de ce travail c'est la réalisation de la méthode de croissance de régions et ceux en adoptant une approche multi-agents que nous avons proposés dans le chapitre 8. Nous croyons fortement que l'interaction entre les agents fournit une solution alternative pour la segmentation d'images, par rapport aux méthodes de segmentation basées sur des modèles complexes et coûteux en temps de calcul.
- Un deuxième besoin s'y ressent, c'est l'utilisation des différents types de contour pour la segmentation des images de profondeur mais en intégrant le paradigme SMA. Dans ce futur travail, chaque agent sera doté d'une méthode soit celle des Level sets, celle du snake adaptatif, et celle du snake classique. Ces trois agents seront en compétition et celui qui fournira la meilleure segmentation sera retenu.

Bibliographie

[Ade & al, 1990] F. Ade, A. Ylä-Jääski, Segmentation and symbolic description of range images, Tech. Rep. BIWI-TR-113 Inst. for Comm. Techn., Swiss Federal Institute for Technology, Zürich, April 1990.

[Angel & al, 2004] Miguel Angel Garcia, Angel Domingo Sappa, Efficient Generation of Discontinuity-Preserving Adaptive Triangulations from Range Images, IEEE Transactions On Systems, Man, And Cybernetics—Part B: Cybernetics, Vol. 34, No. 5, October 2004

[Attila & al, 2006] Kuba Attila, Nyúl László, Palágyi Kálmán, Discrete Geometry for Computer Imagery, Proceedings Series: Lecture Notes in Computer Science, Vol. 4245, Szeged, Hungary, 2006.

[Bab-Hadiashar & al, 2004] Alireza Bab-Hadiashar, Niloofar Gheissari, Model Selection for Range Segmentation of Curved Objects, School of Engineering & Science, Swinburne University of Technology, Melbourne, Australia, 2004

[Ballard & al, 1982] D. H. Ballard, C. M. Brown, Representation of three-dimensional structures, Computer vision Prentice-Hall, Englewood Cliffs, New Jersey, 1982

[BELLET, 1998] Une approche incrémentale, coopérative et adaptative pour la segmentation des images en niveaux de gris, Thèse de doctorat, Institut National Polytechnique de Grenoble, 1998.

[Besl & al, 1988] P. J. Besl, R. C. Jain, Segmentation through variable-order surface fitting, IEEE Trans. Pattern Anal. Machine Intell, vol. PAMI-10, no. 2, pp. 167-192, March 1988.

[Besl, 1988] Paul Besl, Active Optical Range Imaging Sensors, General Motors Research Laboratories, Michigan USA, 1988

[Bhavsar & al, 2010] V. Bhavsar, A. N. Rajagopalan, Inpainting Large Missing Regions in Range Images, International Conference on Pattern Recognition, pp. 3464-3467, 2010.

[Bicking & al, 2007] Frédérique Bicking, Christophe Simon, Thierry Simon, Estimation de profondeur à partir d'images floues par approche génétique, université de Nancy, 2007

[Boissier et al, 2004] O. Boissier, S. Gitton, and P. Glize. Caractéristiques des systèmes et des applications. Systèmes multi-agents, vol 29, pp 25-54. 2004.

[Boucher & al, 1997] Boucher, Doisy, Ronot, Garbay, A society of goal-oriented agents for the analysis of living cells, publié dans Elsevier Science Inc. ISBN:3-540-62709-X, New York, USA, 1997

[Bourennane & al, 2002] E. Bourennane, P. Gouton, M. Paindavoine, F. Truchetet, generalization of Canny-Deriche filter for detection of noisy exponential edge, Signal Processing, Vol 82, No 10, pp 1317-1328, 2002

[Bourjot & al, 2003] C. Bourjot, V. Chevrier, and V. Thomas. A new swarm mechanism based on social spiders colonies from web weaving to region detection. *Web Intelligence and Agent Systems*, 1:47-64, 2003.

[Bourjot & al, 2003] C. Bourjot, V. Chevrier, and V. Thomas. A new swarm mechanism based on social spiders colonies from web weaving to region detection. *Web Intelligence and Agent Systems*, 1:47-64, 2003.

[Bowyer & al, 2006] Kevin W. Bowyer , Kyong Chang , Patrick Flynn , A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition , *Computer vision and Image Understanding*, Vol 101, Issue 1, pp 1-15, January 2006

[Bresson& al, 2007] Xavier Bresson, Selim Esedolu, Pierre Vandergheynst, Jean-Philippe Thiran, Fast Global Minimization of the Active Contour/Snake Model, *Stanley Osher Journal of Mathematical Imaging and Vision* , Vol 28, Issue 2, pp 151-167, 2007

[Buller & al, 2007] Gerald S. Buller, Andrew M. Wallace, Ranging and Three-Dimensional Imaging Using Time-Correlated Single-Photon Counting and Point-by-Point Acquisition, *IEEE journal of selected topics in quantum electronics*, Vol. 13, No. 4, August 2007

[Chaperon, 2002], Thoma Chaperon, segmentation de nuage de points 3d pour la modélisation automatique d'environnements industriels numérisés, école des mines de paris, 2002

[Chatterjee & al, 2011] Shaunak Chatterjee, Stuart J. Russell: A temporally abstracted Viterbi algorithm. *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, pp 96-104, Spain, 2011

[Chaudhuri & al, 2010] D. Chaudhuri, A. Agrawal, Split-and-merge Procedure for Image Segmentation using Bimodality Detection Approach, *Defence Science Journal*, Vol 60, No 3, May 2010

[Chen & al, 2006] Chen Sagiv, Nir A. Sochen, Yehoshua Y. Zeevi. Integrated active contours for texture segmentation. *IEEE Transactions on Image Processing*, Vol 15, Issue 6, pp1633-1646, 2006.

[Cheng & al, 2006] J. Cheng and S.W. Foo, Dynamic directional gradient vector flow for snakes, *IEEE Transactions on Image Processing*, vol. 15, Issue 6, pp 1563-1571, 2006.

[Chevalier & al, 2005] Laurent Chevalier, Fabrice Jaillet, Atilla Baskurt, Segmentation et modélisation 3D par un ensemble de Super ellipsoïdes, Université de Lyon, 2005

[Cremers& al, 2007] Daniel Cremers, Mikael Rousson, Rachid Deriche. A review of statistical approaches to level set segmentation: Integrating color, texture, motion and shape. *International Journal of Computer Vision*, Vol 72, Issue 2, pp195-215, 2007

[Davignon, 1992] A. Davignon, Contribution of edges and regions to range image segmentation, *Applications of Artificial Intelligence X: Machine Vision and Robotics* K. W. Bowyer, Editor *Proc.SPIE 1708*, pp.228-239, 1992.

[Denis & al, 2003] B. Denis De Senneville, P. Desbarats, B. Quesson, and C. Moonen. Real-time artefact corrections for quantitative mr temperature mapping. In Journal of WSCG, vol 11, No.1, pp 87-94. 2003.

[DRIEU, 2001] Benjamin DRIEU, L'intelligence artificielle distribuée appliquée aux jeux d'équipe situés dans un milieu dynamique, Université Paris 8, 2001

[Duchesnay & al, 2001] E. Duchesnay, J. . Montois, Y. Jacquelet, and A. Kinie. An agent-based implementation of irregular pyramid for distributed image segmentation. In Emerging Technologies and Factory Automation, Proceedings of the 8th IEEE International Conference on, volume 1, pp 409-415, 2001.

[Duchesnay& al, 2003] Edouard Duchesnay , Jean-Jacques Montois, Yann Jacquelet, Cooperative agents society organized as an irregular pyramid: a mammography segmentation application, publié dans Elsevier Science Inc.ISSN:0167-8655, New York, USA, 2003

[ELAET, 2004]Fabienne ELAET, Une architecture multi-agent pour la détection, la reconnaissance et l'identification de cibles, Thèse de doctorat, Université Joseph Fourier, Grenoble, 2004

[Fabiano& al, 2004]Paulo Fabiano Urnau Gotardo, Olga Regina Pereira Bellon, Kim Boyer, Luciano Silva, Range Image Segmentation Into Planar and Quadric Surfaces Using an improved Robust Estimator and Genetic Algorithm, IEEE transactions on systems, man, and cybernetics, vol. 34, no. 6, december 2004

[Georgeff & al, 1992] A. Rao and M. Georgeff. An abstract architecture for rational agents. In C. Rich, W. Swartout, and B. Nebel, editors, Proceedings of Knowledge Representation and Reasoning, pages 439-442, 1992.

[Germann& al, 2010] Marcel Germann, In Kyu Park, Michael D. Breitenstein, Hanspeter Pfister, Fast and automatic object pose estimation for range images on the GPU, Machine Vision and Applications 21:749-766 Springer-Verlag , 2010

[Germond & al, 2000] L. Germond, M. Dojat, C. Taylor, and C. Garbay. A cooperative framework for segmentation of mri brain scans. Artificial Intelligence in Medicine, Vol 20, pp 77-93, 2000.

[Ginneken& al , 2002] Bram van Ginneken, Alejandro F. Frangi, Joes J. Staal, Bart M. ter Haar Romeny, and Max A. Viergever, Active Shape Model Segmentation with Optimal Features, IEEE Transactions on medical imaging, VOL. 21, Issue. 8, 2002

[Guo & al, 2008] Ge Guo, Nan Zhang, Longshe Huo, Wen Gao, 2D To 3D Conversion Based On Edge Defocus And Segmentation, IEEE International Conference On Acoustics, Speech and Signal Processing, Las Vegas, USA, 2008

[Gupta & al, 1992] A. Gupta, R. R. Bajcsy, Integrated approach for surface and volumetric segmentation of range images using biquadrics and super quadrics, Applications of Artificial Intelligence X: Machine Vision and Robotics K. W. Bowyer, Editor Proc. SPIE 1708, pp.210-227, 1992.

[Gupta & al, 1993] A. Gupta, R. R. Bajcsy, Recognition of superquadric models in dense range data, Three-Dimensional Object Recognition Systems A. K. Jain, P. J. Flynn, Editors Elsevier, pp.285-310, 1993.

[Haijiao & al, 2013] Theoretical distribution of range data obtained by laser radar and its applications Jiang Haijiao, Lai Jiancheng, Yan Wei, Wang Chunyong, Li Zhenhua, Optics & Laser Technology, vol 45, pp 278-284, 2013

[Haroun & al, 2005] R. Haroun, F. Boumghar, S. Hassas, and L. Hamami. A massive multi-agent system for brain MRI segmentation, volume 3446. LNAI, 2005.

[Harris & al, 1988] C. Harris, and M. Stephens, A Combined Corner and Edge Detector. Proceedings of the 4th Alvey Vision Conference, pp 147—151, 1988

[He & al, 2008] Ning He, Peng Zhang, Proceedings of the 4th international conference on Intelligent Computing: Advanced Intelligent Computing Theories and Applications - with Aspects of Theoretical and Methodological Issues, pp 1141 - 1148, 2008

[Heylighen & al, 1999] F. Heylighen. The science of self-organization and adaptivity. In: Knowledge Management, Organizational Intelligence and Learning, and Complexity, in: The Encyclopedia of Life Support Systems, EOLSS, pages 253-280. Publishers Co. Ltd, 1999.

[Hocevar & al, 2008] Sam Hocevar, and Gary Niger, Reinstating Floyd-Steinberg: Improved Metrics for Quality Assessment of Error Diffusion Algorithms, Lecture Notes in Computer Science, Vol 5099, pp 38-45, 2008

[Hoffman & al, 1987] R. Hoffman, A. K. Jain, Segmentation and classification of range images, IEEE Trans. Pattern Anal. Machine Intell. vol. PAMI-9, no. 5, pp. 608-620, September 1987.

[Huang & al, 2008] H. Cao, P. Huang, and S. Luo. A novel image segmentation algorithm based on artificial ant colonies, volume 4987. Lecture note in computer science, 2008.

[Ikeuchi & al, 2008] Katsushi Ikeuchi, Daisuke Miyazaki, Shape rectification of 3d data obtained by a moving range sensor by using image sequences, Digitally archiving cultural objects, Springer Book, 2008

[Jennings & al, 1998] N. Jennings, K. Sycara, and M. Wooldridge. A roadmap of agent research and development. Autonomous Agents and Multi-Agent Systems, Vol1, pp7-38, 1998.

[Jiang & al, 1994] X. Y. Jiang, H. Bunke, Fast segmentation of range images into planar regions by scan line grouping, 1994.

[Jorge Hernández & al, 2008] Jorge Hernández, Beatriz Marcotegui, Segmentation et Interprétation des Nuages de Points pour la Modélisation d'Environnements Urbains, Centre de morphologie mathématique, Mines ParisTech, 2008

[Kamencay & al, 2012] P. Kamencay, M. Breznan, R. Jarina, P. Lukac, M. Zachariasova, Improved Depth Map Estimation from Stereo Images Based on Hybrid Method, Journal of Radio engineering, Vol. 21 Issue 1, p70, 2012

[Kasvand, 1988] T. Kasvand, The k1k2 space in range image analysis, Proc.9th Int. Conf. on Pattern Recognition pp.923-926, Italy, 1988.

[Kazar & al, 2010] Kazar Okba, Sahar Sana, Une Approche Basée Agent pour la Détection de Régions, Actes du Septième Colloque sur l'Optimisation et les Systèmes d'Information (COSI'2010) , Algerie, 2010

[Kell& al, 2011] Gerald Kell, Andreas Bültner, Michael Wahl , SPAD: a new red sensitive single-photon counting module, Proceedings of SPIE, the International Society for Optical Engineering, Volume 8033, 2011

[Khaldi & al, 2012(a)] Khaldi Amine, Merouani Hayet Farida, Range Image Segmentation Classification, International Journal of Computer Science Issues, Volume 9, Issue 4, Juillet 2012.

[Khaldi & al, 2012(b)] Khaldi Amine, Merouani Hayet Farida, An Active Contour for Range Image Segmentation, Signal and image processing an international journal, Volume 3, Issue 3, Juin 2012.

[Khaldi & al, 2014] Khaldi Amine, Merouani Hayet Farida, Deformable models approach for range image segmentation, The International Journal of Imaging and Robotics, Volume 12, Issue 1, Janvier 2014.

[Khosla & al, 2004] Khosla Rajiv, Ichalkaranje Nikhil, Design of Intelligent Multi-Agent Systems, Studies in Fuzziness and Soft Computing, Springer Book, Vol. 162, 2004

[Klein, 2009] Francois KLEIN, Contrôle d'un système multi-agents réactif par modélisation et apprentissage de sa dynamique globale, Thèse de doctorat, Université Nancy 2, 2009

[KO & al, 2007] Jaeseung Ko, Manbae Kim, Changick Kim, 2D-To-3D Stereoscopic Conversion: Depth-Map Estimation in a 2D Single-View Image, Proceedings of SPIE N°6696, Applications of Digital Image Processing, San Diego, 2007

[Kohlhepp& al, 2009]Peter Kohlhepp, Daniel Fischer, Ekkehard Hoffmann, Intrinsic line features and contour metric for locating 3-D objects in sparse, segmented range images, Image and Vision Computing, Vol 17, pp. 403-417, 2009

[Kohoutek& al, 2010] Tobias K. Kohoutek, Rainer Mautz, Andreas Donaubaue, Real-time Indoor Positioning Using Range Imaging Sensors, Proceedings of SPIE, the International Society for Optical Engineering, Volume 7724, Brussels , BELGIQUE, 2010

[Lange & al, 1999]Lange, Seitz, Biber, Schwarte, Time-of-flight range imaging with a custom solid-state image sensor, Laser Metrology and Inspection, Vol. 3823, Munich, 1999

[Lankton & al, 2008] Shawn Lankton, Allen Tannenbaum, Localizing Region-Based Active Contours, IEEE Transaction on Image Processing, Vol 17, Issue 11, pp 2029-2039, 2008

[Leonardi, 1987] R. Leonardi, Segmentation adaptative pour le codage d'images, Thèse de doctorat, Département d'Electricité, Ecole Polytechnique Fédérale de Lausanne, 1987.

[Levoy & al, 1996] Marc Levoy, Brian Curless, A volumetric method for building complex models from range images, Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, pp 303-312, New York, USA, 1996

[Li & al, 2007] Chunming Li, Chiu-Yen Kao, Zhaohua Ding, John C. Gore, Implicit Active Contours Driven by Local Binary Fitting Energy, IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 2007

[Lin & al, 2012] Jiaxuan Lin, Zhensen Wu, Xiang Su, Jiaji Wu, Biao Wang, Yunhua Cao, GPU-accelerated Computation of 3D laser radar range imaging of arbitrary coarse targets, IEEE 18th International Conference on Parallel and Distributed Systems, 2012

[Liu & al, 1999] Jiming Liu, Yuan Y. Tang, Adaptive Image Segmentation with Distributed Behavior-Based Agents, Hong Kong Baptist University, 1999

[Liu, 2009] Yonghuai Liu, Replicator Dynamics in the Iterative Process for Accurate Range Image Matching, International Journal of Computer Vision, Volume 83, Number 1, Pages 30-56, 2009

[Lowe, 2004] David Lowe, Distinctive Image Features from Scale-Invariant Keypoints, International Journal of Computer Vision, Vol 60 Issue 2, November 2004

[Maître & al, 1989] G. Maître, H. Hügli, Extraction de surfaces par croissance de régions sur un modèle d'objet 2.5D Rapport 259EC05/89 Institut de Microtechnique, Université de Neuchâtel, 1989.

[Merchán & al, 2002] P. Merchán, A. Adán, S. Salamanca, C. Cerrada, 3D Complex Scenes Segmentation from a Single Range Image Using Virtual Exploration, Lecture Notes in Computer Science, Volume 2527/2002, 2002

[Mkwelo & AL, 2007] Simphiwe Mkwelo, Frederick Nicolls, Gerhard De Jager, lecture notes in computer science, volume 4756, progress in pattern recognition, image analysis and applications, pages 340-349, 2007

[Müller, 2002] Jean-Pierre Müller, Des systèmes autonomes aux systèmes multi-agents, Rapport présenté pour l'obtention de l'Habilitation à Diriger les Recherches en Informatique, Université Montpellier II, 2002

[Natonek, 1996] Emerico Natonek, Système de reconnaissance d'objets par vision basé modèles 3-D appliqué aux mondes virtuels robotisés, Ecole fédérale polytechnique de Lausanne, 1996

[Niethammer& al, 2006] Marc Niethammer, Allen Tannenbaum, Sigurd Angenent Dynamic Active Contours for Visual Tracking, IEEE Transaction on Automat Control. Vol 51, Issue 4, pp 562-579, 2006

[Odell & al, 2007] James Odell, Danny Weyns, Andrea Omicini, Environment as a first class abstraction in multiagent systems, Autonomous Agents and Multi-Agent Systems, Vol 14, Issue 1, 2007

[OHLANDER & al, 1978] Ron OHLANDER, Keith PRICE, Raj REDDY, Picture segmentation using a recursive region splitting method, Computer Graphics and Image Processing, Vol8, pp:313-333, 1978

[Park & al, 2008] Johnny Park, Avinash Kak, 3D Modeling of Optically Challenging Objects, IEEE transactions on visualization and computer graphics, vol 14, Issue 2, pp. 246-262, 2008

[Parunak & al, 1997] H. V. Parunak. "Go to the ant": Engineering principles from natural multi-agent systems. Annals of Operations Research, 75(1):69-101, 1997.

[Pavini& al, 1986] B. Parvin, G. Medioni, Segmentation of range images into planar surfaces by split and merge, Computer Vision Pattern Recognition, pp. 415-417, 1986.

[PAVÓN, 2006] Juan PAVÓN, Développement Dirigé par Modèles des Systèmes Multi-Agents, Rapport présenté pour l'obtention de l'Habilitation à Diriger les Recherches en Informatique, Université Pierre et Marie Curie, 2006

[Pulli& al, 1993] Kari Pulli and Matti Pietikäinen, Range Image Segmentation Based on Decomposition of Surface Normals, Scandinavian Conference on Image Analysis (SCIA), Norway, 1993.

[Richard & al, 2004] N. Richard, M. Dojat, and C. Garbay. Automated segmentation of human brain mri images using a multi-agent approach. Artificial Intelligence in Medicine, Vol 30, Issue 2, pp153-175, 2004.

[Richard, 2004] Nathalie richard, Une approche située, coopérative et décentralisée pour l'interprétation d'images cérébrales par RMN, Thèse de doctorat, Institut National Polytechnique de GRENOBLE, 2004

[Rupert & al, 2008] Maya Rupert Carlos Li Salima Hassas, An Organisational Multi-agent Systems Approach for Designing Collaborative Tagging Systems, Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 2008

[Saastamoinen & al, 2013] Kimmo Saastamoinen, Jani Tervo, Jari Turunen, Pasi Vahimaa, and Ari T. Friberg, Spatial coherence measurement of polychromatic light with modified Young's interferometer, Optics Express, Vol. 21, Issue 4, pp. 4061-4071, 2013

[Sabouret, 2009] Nicolas Sabouret, Interactions sur le fonctionnement dans les systèmes multi-agents ouverts et hétérogènes, Rapport présenté pour l'obtention de l'Habilitation à Diriger les Recherches en Informatique, Université Pierre et Marie Curie, 2009

[Scherrer & al, 2007] B. Scherrer, M. Dojat, F. Forbes, and C. Garbay. Mrf agent based segmentation: Application to mri brain scans. In Proceedings of the 11th conference on Artificial Intelligence in Medicine, pp 13-23, Berlin, Heidelberg, 2007.

[Somanath & al, 2010] Gowri Somanath, M V Rohith, Chandra Kambhamettu, Single Camera Stereo System Using Prism and Mirrors, Lecture Notes in Computer Science Volume 6454, pp 170-181, 2010

[Tahvilian& al, 2012] Homa Tahvilian, Payman Moallem, Amirhassan Monadjemi Balloon energy based on parametric active contour and directional Walsh-Hadamard transform and its application in tracking of texture object in texture background, EURASIP Journal on Advances in Signal Processing, December 2012

[Tarel, 1996], Jean-Philippe Tarel, Recalage géométrique avec plusieurs prototypes, institut national de recherche en informatique et en automatique des Yvelines, 1996

[Taylor & al, 1989] R. W. Taylor, M. Savini, A. P. Reeves, Fast segmentation of range imagery into planar regions, Computer Vision Graphics and Image Processing, vol. 45, pp. 42-60, 1989.

[Thomas & al, 2004] P. Thomas Fletcher, Sarang C. Joshi. Principal geodesic analysis on symmetric spaces: Statistics of diffusion tensors., Lecture Notes in Computer Science Volume 3117, pp 87-98, 2004

[Tranier, 2007] John Tranier, Vers une vision intégrale des systèmes multi-agents, Thèse de doctorat, Université des Sciences et Techniques du Languedoc, 2007

[Vidal, 2010] José Vidal, Fundamentals of Multiagent Systems with NetLogo Examples, Computer Science Department University of Southern California, Juillet 2010

[Villiger& al, 2010] M. Villiger, C. Pache, Rainer A. Leitgeb, Theo Lasser, Coherent transfer functions and extended depth of field, Proceedings of SPIE, the International Society for Optical Engineering, Volume 7554, 2010

[Wang & al, 2004] HANZI WANG , DAVID SUTER, MDPE: A Very Robust Estimator for Model Fitting and Range Image Segmentation, Department of Electrical and Computer Systems Engineering, Monash University, Australie, 2004

[Weyns & al, 2007] D. Weyns and T. Holvoet. Architecture-centric software development of situated multi-agent systems. Proceedings of the 7th international conference on Engineering societies in the agents world, pp 62-85, Berlin, Heidelberg, 2007.

[WOLF & al, 2004] T. De Wolf and T. Holvoet. Emergence and self-organisation: a statement of similarities and differences. Lecture Notes in Artificial Intelligence, pp 96-110. Springer Verlag, 2004.

[Yamauchi & al, 2010] Koichiro Yamauchi, BirBhanu, Hideo Saito, 3D Human Body Modeling Using Range Data, International Conference on Pattern Recognition, 2010

[Yokoya& al, 1989] N. Yokoya, M. D. Levine, Range image segmentation based on differential geometry: a hybrid approach, IEEE Trans. Patt. Anal.Mach. Intell, vol. PAMI-11, no.6, pp.643-649, June 1989.

Travaux scientifiques personnelles

- Publication d'un papier « An Active Contour for Range Image Segmentation » dans un journal scientifique « Signal and image processing an international journal », Volume 3, Numéro 3, Juin 2012. Ce journal est indexé dans les bases suivantes : Inspec IET, google scholar, Scribd, world cat, ulrichs web, cseb, EBSCO, pub get, Pub zone, Proquest, .Docstoc, Citeseerx, The Elektronische Zeitschriften bibliothek EZB. L'ISSN du journal est le 2229 – 3922
- Publication d'un papier « Range Image Segmentation Classification » dans un journal scientifique « International Journal of Computer Science Issues», Volume 9, Numéro 4, Juillet 2012. Ce journal est indexé dans les bases suivantes : SCINS, Arxiv.org, Q.Sensei, Bielefeld Academic Search Engine, .Docstoc, google scholar, Scribd, Citeseerx, scientific Commons, directory of open acces journals, SciRate.com, DBLP computer science bibliography. L'ISSN du journal est le 1694-0814.
- Publication d'un article « Deformable models approach for range image segmentation » dans un journal scientifique « The International Journal of Imaging and Robotics», Volume 12, Numéro 1, Janvier 2014. Ce journal est indexé dans les bases suivantes : SCOPUS (Elsevier Bibliographic Databases), SCImago Journal & Country Rank, Index Copernicus, Zentralblatt MATH of European Mathematical Society, Indian Science Abstracts, The American Statistical Association & Institute of Mathematical Statistics, getCITED, Newjour, JournalSeek, Math-jobs.com Journal Index, Academic keys, Ulrich's Periodicals Directory, International Statistical Institute (ISI, Netherlands) Journal Index. L'ISSN du journal est le 0974-0627.