HAL
archives-ouvertes.fr

Interprétation de nuages de points : application à la modélisaion d'environnements 3D en robotique mobile

Nicolas Loménie

▶ **To cite this version:**

Nicolas Loménie. Interprétation de nuages de points : application à la modélisaion d'environnements 3D en robotique mobile. Interface homme-machine [cs.HC]. Université René Descartes - Paris V, 2001. Français. <tel-00136113>

HAL Id: tel-00136113

https://tel.archives-ouvertes.fr/tel-00136113

Submitted on 12 Mar 2007

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

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

UNIVERSITE RENE DESCARTES - PARIS 5
Centre universitaire des Saints-Pères
UFR DE MATHEMATIQUES ET INFORMATIQUES

Thèse présentée en vue de l'obtention du grade de Docteur
de l'Université RENE DESCARTES - PARIS V

Discipline : Sciences de la Vie et de la Matière
Spécialité : Informatique

Par M. Nicolas Loménie

Sujet de la thèse :

Interprétation de nuages de points : application à la modélisation d'environnements 3D en robotique mobile

Soutenue le 10 décembre 2001 devant le jury composé de :

SERRA Jean	Président
DORIZZI Bernadette	Rapporteur
SEQUEIRA Jean	Rapporteur
STAMON Georges	Directeur
LACROIX Simon	
GOULETTE François	
CAMBOU Nicole	
CLOPPET-OLIVA Florence	

Remerciements

♢

Je n’ai pas
besoin de le dire ni
de l’écrire, car tous ceux
qui m’ont porté attention, affection
et intérêt tout au long de cette aventure hu-
maine, intellectuelle et personnelle le savent déjà.
Mais je les remercie ici à nouveau tous, du fond du coeur.
Fred, Jérôme, Florence et tous les membres d’une équipe SIP d’ex-
ception ! Fred et les autres thésards, Sylvie, Françoise, Bernard du pavillon
d’en face de l’UFR de Mathématiques et Informatique ! Véronique, Maurine, Laeti-
tia, Philippe, et toute la troupe des fous de théâtre. Claire, Romain, Saeïd et Samy toujours
proches. Ma mère, ma soeur, François et toute ma famille. Nicole, Simon, Anthony,
François, Thomas qui de Toulouse au LAAS ou de Paris au CAOR et à l’Aé-
rospatiale ont collaboré à plusieurs reprises à ces travaux. Je remer-
cie chaleureusement Bernadette Dorizzi et Jean Sequeira
d’avoir accepté d’être rapporteurs de cette thèse,
et Jean Serra président du jury. Georges
et George, chacun à leur ma-
nière. Et tous ceux qui sont
passés et ne sont
pas restés...

♢

A ma mère

Résumé

Cette thèse traite de l'analyse de nuages de points 3D désorganisés dans le cadre de l'interprétation de scènes 3D. Nos travaux s'appuient sur deux outils : un algorithme de partitionnement efficace inspiré des C-moyennes floues [GG89] d'une part, et des outils de filtrage morphologique de représentation à base de triangulation de Delaunay pour reprendre les termes de N. Amenta dans [ABK98]. Le cadre applicatif essentiel est la navigation autonome en robotique mobile en environnement inconnu, c'est-à-dire sans modèle. Mais la méthodologie générique développée d'analyse de nuages de points a été appliquée à d'autres types d'environnements, notamment plus structurés.

Mots-Clés : regroupement et segmentation, méthodologie, reconstruction 3D, interprétation de scènes et de nuages de points, connaissances heuristiques, analyse d'images.

8

Table des matières

1 Introduction **15**

2 Obtenir des données 3D **19**

2.1 Les systèmes d'acquisition 19

2.1.1 Les acquisitions volumiques 20

2.1.2 Les acquisitions surfaciques 20

2.1.2.1 Les approches passives 20

2.1.2.2 Les approches actives 25

2.1.3 Conclusion 26

2.2 Les méthodes de visualisation/représentation 27

2.2.1 Visualisation ponctuelle 28

2.2.2 Visualisation volumique 29

2.2.2.1 Avec modèle 29

2.2.2.2 Sans modèle 29

2.2.3 Visualisation surfacique 31

2.2.3.1 Modèles discrets par facettes 31

2.2.3.2 Modèles continus par paramètres 32

2.2.4 Conclusion 32

3 Interpréter des données 3D **35**

3.1 Introduction 35

3.2 Reconstruction à partir de la primitive région 37

3.2.1 Modèles plans 37

3.2.1.1 Segmentation en régions pilotée par l'appariement stéréoscopique 38

3.2.1.2 Reconstruction 3D de scènes à partir de la paire d'images stéréoscopiques segmentée 39

3.2.1.3 Expression de le transformation affine 41
3.3 Reconstruction à partir de la primitive contour 44
3.4 Reconstruction à partir de la primitive point 46
3.4.1 Segmentation d'un nuage de points 3D 46
3.4.2 Segmentation d'un nuage de points 3D en primitives géométriques 46
3.4.2.1 Quelques rappels de géométrie différentielle . . 46
3.4.2.2 Quelques outils de visualisation 47
3.4.2.3 La recherche des primitives géométriques . . . 48
3.4.3 Segmentation d'un nuage de points 3D en objets 53
3.4.4 Techniques à base de graphes explicites 56
3.5 Reconstruction 3D de nuage de points 56
3.6 Conclusion 59

4 Regroupement **61**
4.1 Problématique du regroupement 61
4.2 Mesure de proximité 65
4.2.1 Fonction de dissimilarité entre deux éléments 65
4.2.2 Fonction de dissimilarité entre un élément et un ensemble 66
4.2.3 Fonction de dissimilarité entre deux ensembles 67
4.3 Algorithmes 67
4.3.1 Algorithmes séquentiels 68
4.3.2 Algorithmes hiérarchiques 68
4.3.3 Algorithmes itératifs 69
4.4 La théorie du flou 72
4.4.1 Son utilisation en Reconnaissance des Formes 72
4.4.2 Le regroupement flou 73
4.5 Les méthodes itératives de regroupement flou 74
4.5.1 Fondements historiques et algorithmiques 75
4.5.2 Convergence et terminaison 76
4.5.3 Validité et interprétation des résultats 77
4.5.4 Variantes et illustrations 79
4.5.4.1 Distance Euclidienne 79
4.5.4.2 Distance de Mahalanobis 82
4.5.4.3 Distance à une droite 83

4.5.4.4 Distance “exponentielle” 83
4.5.4.5 Récapitulatif 85
4.5.4.6 Variations en fonction du nombre de classes . . 86
4.6 Conclusion 90

5 α-Opérations Morphologiques 91
5.1 Introduction 91
5.2 Définitions 92
5.2.1 Diagramme de Voronoï 92
5.2.2 Complexe de Delaunay 93
5.3 Caractéristiques géométriques et algorithmiques 94
5.3.1 Propriétés 94
5.3.2 Schémas algorithmiques 94
5.3.3 Forme et graphe de Delaunay 95
5.4 α-Objets 97
5.5 Opérateurs morphologiques 100
5.5.1 Erosion 101
5.5.2 Dilatation 101
5.5.3 Ouverture 101
5.5.4 Complexité 102
5.6 Illustrations 102
5.7 Liens avec la morphologie mathématique classique 107
5.8 Conclusion 110

6 Stratégie générique d’analyse 113
6.1 Postulat 114
6.2 Analyse en regroupement 116
6.2.1 Heuristique d’ambiguïté géométrique 120
6.2.2 Heuristique d’ambiguïté géométrique complète 125
6.2.3 Heuristique d’ambiguïté topologique 125
6.2.4 Heuristique d’ambiguïté planaire 131
6.2.5 Récapitulatif des heuristiques utilisées 131
6.3 Quelques éléments de classification des objets 131
6.4 Reconstruction 3D 133
6.5 Organigramme de la stratégie de reconstruction 3D en mode statique 135

6.5.1 Analyse de la scène en objets . . . 135
6.5.2 Reconstruction des objets . . . 137
6.5.3 Organigramme . . . 137
6.6 Organigramme de la stratégie de reconstruction 3D en mode dynamique . . . 140
6.6.1 Reconstruction de l'environnement . . . 140
6.6.2 Reconstruction des objets . . . 143
6.6.3 Organigramme . . . 144
6.7 Complexité algorithmique . . . 144
6.8 Conclusion . . . 146

7 Résultats **149**
7.1 Introduction . . . 149
7.2 Description d'une scène non structurée . . . 150
7.3 Description d'une scène structurée . . . 159
7.3.1 Base 2 . . . 159
7.3.2 Base 3 . . . 162
7.3.2.1 Comparaison avec d'autres approches . . . 162
7.4 Description d'un univers d'obstacles . . . 174
7.5 Conclusion . . . 174

8 Conclusion **177**

Bibliographie **190**

A Algorithme de triangulation de Delaunay 2D **191**

B Algorithme CMFE des C-moyennes floues exponentielles **197**

Table des Figures **207**

“Les sciences n’essaient pas d’expliquer ; c’est tout juste si elles tentent d’interpréter ; elles font essentiellement des modèles. Par modèle, on entend une construction mathématique qui, à l’aide de certaines interprétations verbales, décrit les phénomènes observés. La justification d’une telle construction mathématique réside uniquement et précisément dans le fait qu’elle est censée fonctionner.”

John Von Neumann

Chapitre 1

Introduction

Quand on cherche à intégrer un système de vision sur une machine, on doit accepter de raisonner en trois dimensions voire en quatre. Or, la plupart des travaux de vision font de notre machine un cyclope assis sur une chaise. En effet, physiologiquement, la vision humaine est capable de traiter un environnement 3D non seulement grâce à sa paire d'yeux - possédant ainsi un système stéréoscopique de vision embarqué - mais aussi par la base de connaissances - et notamment de modèles - accumulées avec l'expérience et l'apprentissage. Ainsi, si vous marchez dans la rue et croisez une voiture, ce que vous verrez dans votre représentation mentale de la scène sera moins une vue tronquée de l'objet 3D voiture que le modèle 3D complet que vous vous faites de l'objet 3D voiture. Pour vous en convaincre, il suffira de faire le test simple suivant : cachez un de vos yeux avec une main et observez la scène ; vous n'êtes plus censé percevoir l'information de distance et pourtant vous percevez en 3D l'information de la scène car votre cerveau reconstruit ce qu'il connaît déjà grâce aux modèles stockés en profusion. Nous concevons aisément que l'utilisation de ces modèles 3D en grande quantité pour détecter, reconnaître, se représenter et donc reconstruire en temps réel l'ensemble des objets qu'un homme marchant dans la rue est susceptible de rencontrer est un exploit encore au-dessus des capacités technologiques actuelles de traitement informatique. Par ailleurs, nous en sommes encore réduits à raisonner de façon discrète sur les surfaces d'objets 3D qui nous entourent alors que le système visuel humain est vraisemblablement capable d'avoir une représentation 3D directement continue de telles surfaces en fonction de leurs projections 2D sur chacune des rétines. Au niveau industriel, de nombreux produits ont été déve-

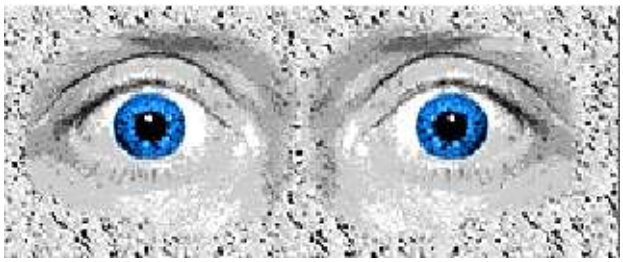

FIG. 1.1 – Système visuel stéréoscopique humain

loppés pour la numérisation surfacique des objets dans les systèmes CAO, où le problème consiste à passer de la perception physique d'un objet réel à son modèle géométrique exploitable informatiquement. Ces techniques sont du plus grand intérêt économique pour le prototypage industriel rapide. Il existe donc d'ores et déjà de nombreuses techniques de numérisation d'objet réel sous forme de nuage de points de $\Re^3$ échantillonnés à leur surface. La plupart du temps, on obtient un ensemble de points désorganisés, c'est-à-dire que les points ne sont pas ordonnés en grille régulière comme nous y avait habitué l'analyse d'image 2D, et la densité n'est pas constante. De plus, l'objet est ordinairement déjà isolé.

A partir de ce cadre maintenant bien établi, nous nous intéressons à l'analyse des nuages de points généraux et décrivant tous types de scènes. L'accent n'est alors plus à mettre sur la technique de reconstruction mais plutôt sur la faculté d'analyse, de dénombrement et de représentation d'un nuage de points contenant plusieurs objets, utilisant à l'occasion des techniques déjà expérimentées de reconstuction.

Par ailleurs, il est vrai qu'historiquement nos travaux se situent dans un cadre de robotique mobile, dans lequel on espère pouvoir doter un robot d'une certaine autonomie en navigation par l'octroi d'une capacité de construire peu à peu un modèle de l'environnement exploré, inconnu *a priori*. Nous avons l'objectif de fournir une description relativement détaillée des obstacles qu'il a eu ou qu'il aura à éviter. Comme le rappelle Anthony Mallet [Mal01] dans l'introduction de sa thèse dédiée à la robotique mobile autonome, le problème de partitionnement d'une scène en objets ou éléments simples reste un problème ouvert en environnement naturel, où l'absence de structure géométrique rend difficile à concevoir des algorithmes de modélisation.

Cette problématique a été abordée en collaboration avec le service Vision de

l'Aérospatiale à Châtillon et le laboratoire LAAS-CNRS à Toulouse, notamment pour la collecte de série de données.

Dans cette perspective, l'originalité de notre travail réside entre autres dans la définition d'une méthodologie générique d'analyse de scène représentée par des nuages de points 3D, indépendamment de l'environnement. Elle tient également à l'application au domaine de la vision par ordinateur d'un algorithme spécifique de partitionnement peu utilisé, et surtout dans le champ restreint de l'Analyse de Données pure. Enfin, ses travaux de recherche nous ont amené à imaginer des opérateurs de la morphologie classique applicables à des nuages de points 2D désorganisés.

Après avoir rappelé dans le deuxième chapitre les principales techniques utilisées pour résoudre la problématique de la reconstruction d'objets ou d'environnements en trois dimensions, nous donnerons dans le chapitre trois un aperçu des méthodes conçues à ce jour pour interpréter des nuages de points 3D, puis nous détaillerons dans la quatrième partie de ce mémoire, les techniques de segmentation d'un nuage de points 3D issues de l'Analyse de Données en explicitant notre choix à partir d'exemples concrets en deux dimensions.

Le cinquième chapitre de ce mémoire est consacré à la description de formes données sous forme de nuage de points 2D désorganisés, avec l'introduction de la notion de forme-α. L'une des originalités de notre travail se situe notamment au niveau de la définition d'opérateurs de filtrage morphologique de formes décrites par de tels nuages.

La sixième partie montre comment l'utilisation conjointe de ces deux techniques de segmentation et de filtrage morphologique à partir de graphe de Delaunay permet d'aboutir, par le réglage de quelques paramètres garants du niveau de détail exigé, à une description de la scène en trois dimensions exploitable pour la navigation autonome d'un robot en milieu inconnu par exemple.

Enfin, le septième chapitre de ce mémoire illustre les résulats obtenus par l'application de notre méthodologie à différents types d'environnements et nous permettra de conclure sur la généricité et la robustesse de l'approche proposée. Nous avons testé nos algorithmes sur des nuages de points issus de différents capteurs et représentant des scènes possédant des niveaux de structuration géométrique très variés :

– de capteurs stéréoscopiques installés sur un robot appelé LAMA évoluant

en pleine nature ;

- de capteurs lasers de la société MENSI utilisés en milieu industriel ;
- de données académiques en univers plan à titre de comparaison ....

Chapitre 2

Obtenir des données 3D

Avant toute tentative d'analyse "intelligente" d'un monde perçu en trois dimensions, il convient de se pencher sur les avancées technologiques qui ont précisément permis ces dernières années de faire percevoir aux machines le monde qui les entourait en trois dimensions. De l'imitation du système visuel humain à l'utilisation de techniques actives comme le laser, les avancées technologiques remarquables réalisées par les ingénieurs et les chercheurs pour appréhender au mieux cette troisième dimension qui échappait encore aux systèmes de vision imaginés méritent qu'on s'y attarde. Au cours de ce chapitre, obtenir des données 3D évoquera à la fois la capacité de les acquérir et celle de les visualiser.

2.1 Les systèmes d'acquisition

Des plus coûteux aux plus volumineux, les caractéristiques des systèmes d'acquisiton de données en trois dimensions sont d'une diversité liée en partie :

- à l'approche envisagée : les cognitivistes ont étudié le système de vision humain pour reproduire la perception de la profondeur par l'usage de plusieurs capteurs plans, pendant qu'une partie des besoins industriels exigeait des techniques plus directes et précises comme la télémétrie laser par exemple ;
- aux avancées technologiques :
- aux impératifs de coûts : un système de vision stéréoscopique équipé de deux caméras CCD est beaucoup moins coûteux qu'un système à base de laser ;
- aux scènes et objets à numériser : des technique passives ont encore du mal

à s'affranchir d'objets peu texturés par exemple.

De façon concomitante, cette diversité est reflétée par la variété des caractéristiques des nuages de points 3D récupérés en sortie des systèmes d'acquisiton : entre un ensemble de points structuré en grille régulière et un nuage de points désorganisé (cf. figure 2.7), les facilités de traitement diffèrent évidemment grandement.

De plus, on fera une ultime distinction de nature entre les techniques de surface et de volume.

2.1.1 Les acquisitions volumiques

C'est dans le domaine médical que l'on retrouve des capteurs capables de scanner un volume. Par exemple, les méthodes dites IRM mesurent l'activité magnétique de tissus organiques en réponse à différentes formes d'excitation, et ceci de manière non invasive. Dans un premier temps, les données à traiter sont alors représentées sous la forme de "voxel" encore appelé élément de volume, équivalent 3D des "pixels" 2D, ou élément d'image.

2.1.2 Les acquisitions surfaciques

Parmi l'ensemble des techniques qui permettent de scanner la surface des objets à saisir en trois dimensions, on distingue généralement les techniques passives qui n'utilisent pas de source d'énergie extérieure au processus de formation de l'image et les techniques actives qui utilisent une source d'énergie propre appliquée à l'objet ou à la scène à numériser.

2.1.2.1 Les approches passives

On citera dans cette catégorie, trois approches passives :

- à un niveau marginal, les techniques utilisant des capteurs mécaniques sensibles ;
- la méthode encore au stade expérimental de *"reconnaissance du relief à partir de l'éclairement"* dite du *"shape from shading"* [DD99] ;
- et enfin la plus répandue et ancienne, la stéréovision, qui nous intéresse particulièrement.

A la manière d'un aveugle, le capteur peut être exclusivement mécanique. On peut alors tâter l'ensemble de la surface d'un objet à l'aide d'un stylet tactile associé à un capteur particulier, dit de Polhemus. Un dispositif électronique mesure alors les déplacements effectués par rapport à une position de référence et calcule les coordonnées du point désigné, ceci sur toute la surface de l'objet. A l'évidence, c'est une technique relativement longue et pénible à mettre en oeuvre.

Quand le capteur est optique, il faut d'abord comprendre le processus de formation de l'image. Un observateur (oeil, appareil photographique ou caméra) observe une scène éclairée par un ensemble de sources lumineuses. Le processus de formation de l'image au niveau du capteur correspond à une projection perspective de chacun des points $P(x, y, z)$ de l'espace physique 3D vers un point $m(u, v)$ de la surface 2D du capteur, selon les lois classiques de l'Optique. Dans cette transformation, on perd l'information de profondeur.

Plus formellement, les caractéristiques optiques du dispositif comme la focale constituent une matrice T appelée matrice perspective de la caméra, qu'il s'agit d'estimer notamment par des techniques de calibration et d'étalonnage. Une fois cette matrice perspective connue, la relation en géométrie projective reliant les coordonnées de $P(x, y, z)$ dans l'espace 3D à celle de son pixel projeté $m(u, v)$ dans un espace 2D est entièrement déterminée en coordonnées homogènes par :

$$s \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = T \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{2.1}$$

où s est un facteur d'échelle.

Pour simplifier, c'est ce facteur d'échelle s indéterminé qui ne permet pas, malgré la connaissance de la matrice T, de retrouver inversement et explicitement l'ensemble des coordonnées 3D du point $P(x, y, z)$ associé à une projection sur une seule image. C'est la raison pour laquelle on distinguera les techniques optiques par le nombre de vues de la scène disponible. Une façon simple de s'en persuader physiquement consiste à tracer la droite qui relie le centre optique du système au point projeté à la surface du capteur. La connaissance d'une seule vue ne nous permet pas de déterminer de quel point sur la droite est issue la projection observée (voir figure 2.1). En revanche, si l'on a un deuxième système d'acquisition, le point réel est déterminé par l'intersection des deux droites reliant chacun

des deux centres optiques aux projections respectives du point réel sur les capteurs 2D respectifs.

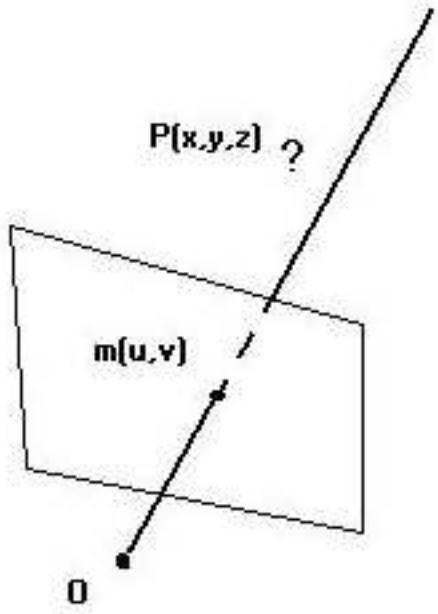

FIG. 2.1 – Illustration de l'indétermination de la position 3D du point projeté le long de la droite liant le centre optique du dispositif optique et le point projeté

En conséquence, si l'on dispose par choix ou par nécessité d'une seule vue 2D de la scène, on parlera alors de techniques dites du *"Shape from shading"*, qui exploitent les caractéristiques d'éclairement du relief à reconstituer comme caractéristique de l'orientation de la surface observée. Ces techniques sont très spécifiques et assez peu génériques, même si elles s'appuient sur des observations physiologiques de la capacité humaine à estimer la profondeur. Dans le prolongement de ce type d'approches, on citera également la technique dite du *"shape from texture"* qui s'attache à utiliser l'information de déformation du motif de texture observée à la surface d'un objet pour en estimer le relief.

Au contraire, si l'on peut disposer de plusieurs vues de la même scène (par un dispositif spécifique ou simplement par le mouvement du capteur[AM90]), on peut *a priori* récupérer "sans ambiguïté" l'information de profondeur du point 3D comme illustré géométriquement sur la figure 2.2.

En effet, dans la plupart des systèmes de vision, le monde en trois dimensions est projeté sur des surfaces réceptrices à deux dimensions. Ces surfaces peuvent

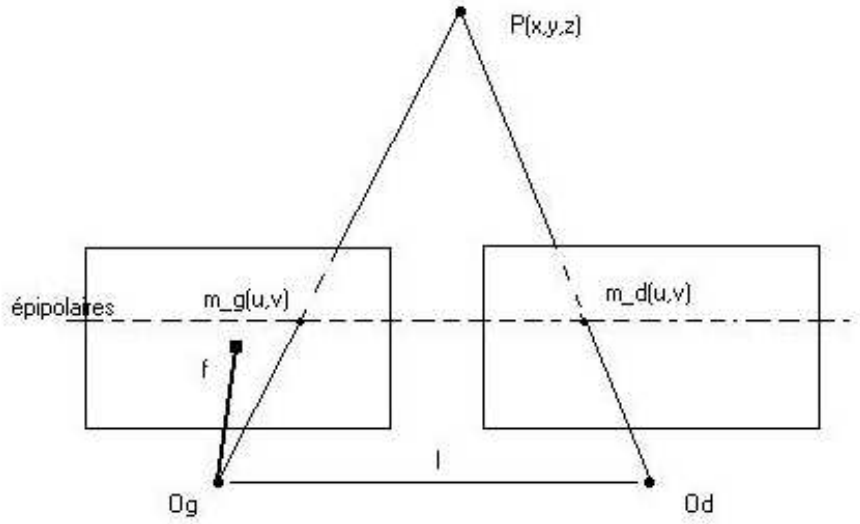

FIG. 2.2 – Modèle de système stéréoscopique de caméras en géométrie rectifiée

être les deux rétines d'un système de vision binoculaire vivant ou les caméras d'un système de vision artificiel. Au cours de ce processus, l'information de profondeur est perdue mais peut être récupérée à partir des *disparités* entre des parties d'image mises en correspondance. En effet, dans un système stéréoscopique, les yeux ou les caméras ont un point de vue légèrement différent. En conséquence, il existe de petites variations entre les images projetées appelées *disparités* comme illustré en figure 2.3, qui sont évaluées spatialement pour obtenir une information de profondeur. En pratique, étant donnés les points image projetés correspondant aux projections optiques dans les deux images d'un point $P(x, y, z)$ d'une surface du monde 3D, identifier la position du point dans le monde 3D devient un simple problème de trigonométrie. Ce procédé est appelé *triangulation*. La partie difficile du problème est de déterminer la correspondance entre les points projetés dans les deux images. Or, pour un pixel $m_g(u, v)$ dans l'image gauche, le pixel $m_d(u, v)$ qui lui correspond dans l'image droite se trouve forcément sur une droite que l'on appelle la droite épipolaire associée à m_g. Pour plus de détails on pourra se référer à [Aya89]. Par ailleurs, techniquement, on peut négliger les disparités verticales en supposant une géométrie de caméras de fronts de visée strictement parallèles (ou se ramener à une telle situation en passant en géométrie rectifiée). Dans ce cas, il est suffisant d'analyser les correspondances entre les images ligne par ligne puisque les lignes *épipolaires* sont dans ces conditions horizontales comme illustré en figure 2.2. La recherche de la troisième dimension est ainsi réduite à un

problème à une dimension, laissant la voie libre à l'utilisation de filtres spatiaux classiques pour interpréter la disparité comme le résultat d'une convolution. Une fois la disparité δ estimée, on récupère approximativement les coordonnées du point $P(x, y, z)$ par les relations :

$$\begin{pmatrix} x = \frac{lu}{\delta} \\ y = \frac{lv}{\delta} \\ z = \frac{lf}{\delta} \end{pmatrix} \tag{2.2}$$

où l désigne la distance entre les deux centres optiques et f la focale du système optique considéré.

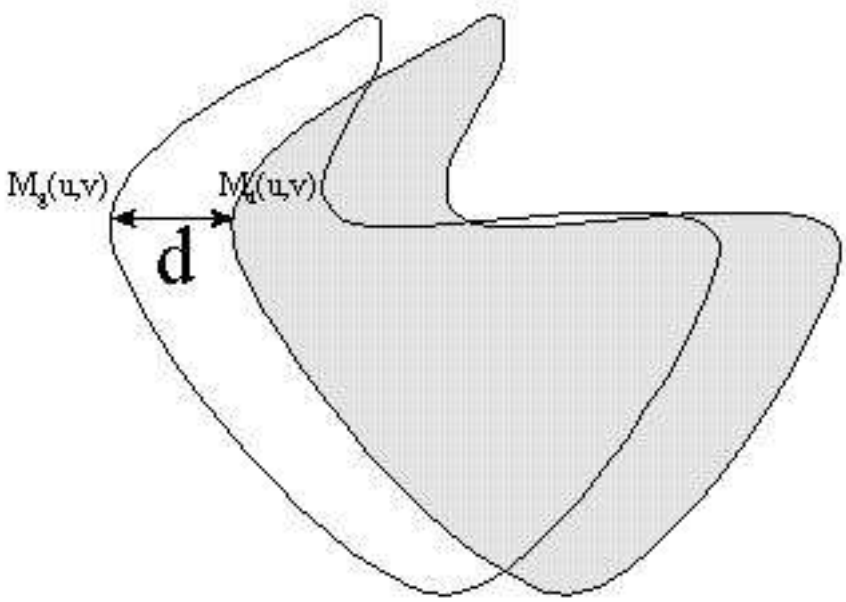

FIG. 2.3 – Illustration de la notion de disparité et de la mise en correspondance entre deux images

On montre en figure 2.4 le robot LAMA du laboratoire LAAS-CNRS à Toulouse, qui a servi à obtenir la grande majorité des séries de scènes d'extérieur présentées dans cette thèse sous la forme de nuages de points 3D. Sa relative autonomie motrice nous a également permis d'explorer tout un environnement, *a priori* inconnu.

On constate donc que ces techniques demandent une puissance de calcul assez importante. Mais, avec ce robot par exemple, on est capable d'atteindre des fré-

FIG. 2.4 – Banc stéréo monté sur le robot LAMA

quences d'acquisition de scènes 3D de l'ordre du Hertz pour des images 192x256 par exemple.

2.1.2.2 Les approches actives

Parallèlement, et pour des impératifs industriels essentiellement, on a beaucoup utilisé les capteurs actifs, qui font intervenir une source d'énergie extérieure propre au système, comme les faisceaux lasers. Le principal inconvénient de cette technique est son prix (de l'ordre du million de francs).

Une première méthode repose sur le principe de triangulation qui permet de récupérer l'information de profondeur de tout point éclairé par la source laser grâce aux caractéristiques géométriques d'un dispositif composé de l'émetteur laser et du récepteur photographique . On donne à titre d'illustration en figure 2.5 une photographie du dispositif utilisé par la société MENSI pour l'acquisition de données en environnement dangereux par le robot SOISIC qui utilise cette technique.

Une deuxième méthode exploite le temps de retour d'un faisceau lumineux, indépendamment de tout principe géométrique de formation de l'image. Au prix

FIG. 2.5 – Capteur laser SOISIC de la société MENSI

d'un dispositif mécanique de balayage très coûteux, on obtient ainsi des nuages de points denses et précis. De façon similaire au laser, on pourra utiliser des trains d'impulsions ultrasonores, ce qui augmente sensiblement simplement le temps d'acquisition, mais surtout l'ambiguïté en raison de la largeur du faisceau.

Enfin, l'usage de lumière structurée permet de réduire les coûts du laser (un système pour 10kF) tout en s'affranchissant des problèmes d'absence de texture, puisque on utilise un motif constant projeté sur la scène à modéliser. Puis, en étudiant les déformations de ce motif on peut reconstruire la scène en 3D. Précis, rapide mais nécessitant une phase de calibration similaire à celle de la stéréovision, cette technologie pourrait constituer un compromis entre les méthodes passives de type stéréovision et les méthodes actives coûteuses, essentiellement pour des objets isolés.

2.1.3 Conclusion

En dernière analyse, on utilisera majoritairement deux méthodes maintenant bien éprouvées :

la stéréovision / photogrammétrie, technique passive : – fournissant des nuages de points 3D de densité variable et relativement imprécis (30 cm à 10 m à peu près), avec une dégradation forte de la qualité du nuage avec la distance de l'objet au capteur (les imprécisions croissent quadratiquement

avec la distance) ;
- peu coûteuse, ancienne, bien maîtrisée et facile de conception ;
- de plus en plus performante ;
- mais sensible aux effets de texture, dans la mesure où les surfaces non texturées sont difficilement appréciables par cette technique et, qu'en conséquence, donne des résultats médiocres si elle est utilisée de façon brute en environnement intérieur : on préferera alors faire de la stéréovision sur des primitives plus évoluées que le point, comme les segments par exemple ;
- et nécessitant une grosse puissance de calcul (rapidité de prise de vue de l'ordre du Hertz actuellement) ;

la triangulation / télémétrie laser, technologie active et concurrente : – fournissant des nuages de points 3D peu bruités, précis (jusqu'à 1 millimètre à 5 mètres pour les plus précis chez MENSI ou UKR, +/- 3cm pour un télémètre moyen) et denses ;
- récente et prometteuse ;
- mais coûteuse (de 30 kF à plus de 1 MF) ce qui interdit toute application de robotique à large diffusion dans le public par exemple ;
- difficile à mettre en place techniquement et relativement lente ;
- sensible aux propriétés spéculaires des objets, nécessitant parfois de traiter la surface à reconstruire par une couche de peinture.

2.2 Les méthodes de visualisation/représentation

Une fois les données acquises, il faut trouver une structure de visualisation ou de représentation plus ou moins significative et adaptée à des traitements informatiques ultérieurs. La primitive point peut suffire à percevoir la forme d'un nuage de points, surtout lorsque le nuage est dense. En revanche, il est intéressant de pouvoir dériver facilement une structure géométrique d'accueil de ces données ponctuelles de plus haut niveau comme des régions ou des contours, sous la forme de maillage par exemple. Toutes ces techniques intéressent au plus au point de nouvelles disciplines comme la création d'images de synthèse ou la reconstruction d'environnement industriel “Tel Que Construit” en CAO pure.

2.2.1 Visualisation ponctuelle

Les dispositifs décrits précédemment permettent d'obtenir des nuages de points géométriques.

Au mieux, on aura également en sortie du capteur des cartes de profondeurs ou *range image* comme illustré en figure 2.6 qui permettent de conserver la structure de voisinage auquel nous a habitué le traitement d'images d'intensité classique. Sur ces cartes, chaque pixel est affecté d'une intensité traduisant l'information de profondeur par l'intermédiaire d'une valeur de disparité. Plus la disparité est grande, plus l'objet est proche et la valeur d'intensité claire. Les zones blanches correspondent aux pixels non évalués, c'est-à-dire non appariés. Ces images offrent donc un espace privilégié d'analyse de l'information 3D de la scène, car on bénéficie alors d'un ensemble de points structuré en grille régulière qui conserve toute les informations de voisinage indispensables aux techniques habituelles de traitement d'images (voir figure 2.7). Pour autant, on peut néanmoins observer que d'une part cette structure de voisinage est contaminée par les nombreux pixels non appariés et d'autre part cette présentation des données est relativement contrainte. En effet, la plupart du temps, on ne dispose pas de cette correspondance pixel 2D / point 3D reconstruit.

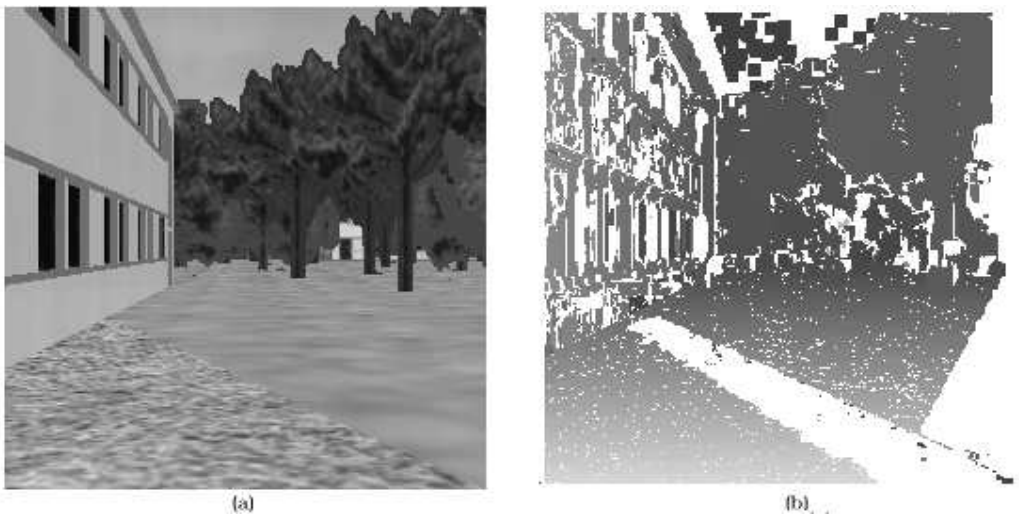

FIG. 2.6 – (a) vue gauche d'une scène artificielle (b) carte de profondeur calculée

En effet, on fusionnera souvent différents points de vue de l'objet étudié pour en avoir une vision complète sous tous les angles. Pour cette raison entre autres, la scène à analyser n'est souvent disponible que sous la forme d'un nuage de

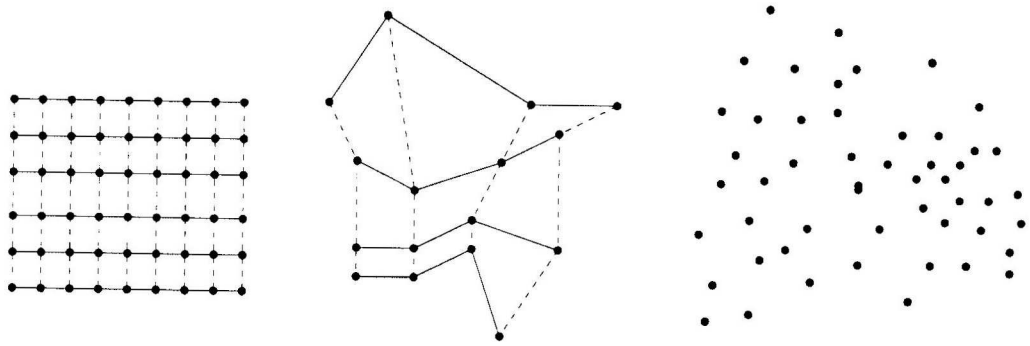

FIG. 2.7 – Les différents types d'organisation de nuage de points

points 3D désorganisé, ce qui ne nous permet plus alors de nous appuyer sur une structure de voisinage déjà établie. Eventuellement, il s'agit alors d'être capable de la construire. Le plus fréquemment, on essayera de trouver un autre mode de représentation plus générique.

2.2.2 Visualisation volumique

2.2.2.1 Avec modèle

Lorsque l'objet à reconstruire est un objet manufacturé par exemple, on peut disposer d'un ensemble de primitives solides de base qui permettent de décrire les objets traités. La plus connue de ces modélisations s'appelle la *Constructive Solid Geometry* ou modèle par arbre CSG et constitue l'un des principaux modèles utilisés en CAO [Eve89][Aub96][Bon94][Mon96]. Un objet 3-D y est construit en assemblant par union, intersection ou différence des formes élémentaires comme des sphères, des cylindres ou des parallépipèdes dont on peut paramétriser les dimensions : les feuilles de l'arbre sont ces éléments géométriques simples et les noeuds les opérations ensemblistes. Le modèle CSG est particulièrement adapté aux objets manufacturés, en permettant la définition d'une grammaire de description d'un objet. Comme toute méthode *structurelle*, la difficulté consiste souvent à segmenter ces primitives élémentaires.

2.2.2.2 Sans modèle

Lorqu'on ne dispose pas de modèle des objets à reconstruire, il s'agit de décomposer l'objet en éléments de volume homogène, de géométrie identique indé-

pendante de l'objet traité. Par exemple, on décomposera l'espace en cubes élémentaires appelés "voxels" de taille fixe, désignés par les coordonnées de leur centre. Un objet est alors représenté par l'ensemble des voxels dont il occupe l'espace, à la manière des pixels en imagerie 2D.

On peut également décomposer l'espace dans une structure de type *octree* (voir figure 2.8) qui permet de rassembler un ensemble de voxels dans une seule cellule cubique homogène du point de vue de l'intensité, à la manière des *quadtree* en imagerie 2D. C'est-à-dire qu'on opère pratiquement une décomposition hiérarchique itérative par divisions succesives d'une cellule cubique de départ en fonction d'un critère d'homogénéité lumineuse. Cette structure réduit ainsi l'espace mémoire nécesssaire au stockage de l'image.

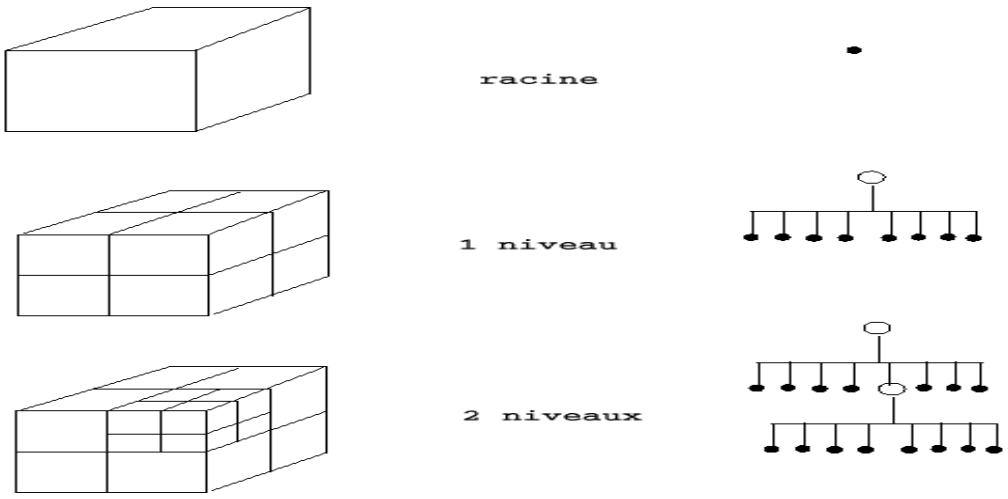

FIG. 2.8 – Visualisation par octree de données volumiques

Enfin, on peut décomposer l'objet en tétraèdres selon des méthodes inspirées des travaux de J-D. Boissonat [Boi84] qui consiste à *sculpter* selon la technique dite du *ray tracing* un bloc de tétraèdres auparavant obtenu par triangulation de Delaunay de l'ensemble des sites du nuage de points 3D à visualiser. Les algorithmes sont relativement complexes notamment au niveau des structures de données pour accueillir les relations de voisinage entre tétraèdres et chacune de ses faces. On donne une illustration d'une telle représentation en figure 2.9.

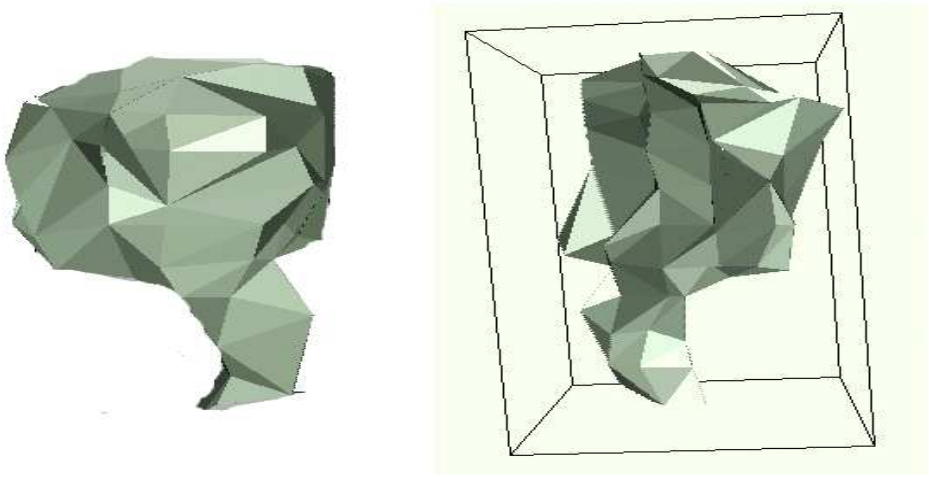

FIG. 2.9 – Structure tétraédrique de Delaunay en 3 dimensions

2.2.3 Visualisation surfacique

2.2.3.1 Modèles discrets par facettes

La triangulation permet également de représenter la surface d'un objet par un assemblage de facettes triangulaires, dans la continuité de l'approche tétraédrique précédente. La décomposition volumique permet d'obtenir une orientation cohérente de l'ensemble des facettes qui en composent l'enveloppe surfacique. La triangulation au sens large a fait l'objet de nombreux travaux.

Si l'on dispose d'une représentation 3D initiale par voxels, la technique dite des *Marching cubes* remplace chaque voxel par une face triangulaire optimale choisie parmi un ensemble fini, ce qui permet d'obtenir *in fine* un maillage triangulaire de la surface à visualiser. Cette méthode demeure très utilisée.

Les surfaces déformables sont des maillages réguliers que l'on déforme en les faisant converger vers une surface ou un objet en leur appliquant un certain nombre de forces. Généralisation des *snakes*, elles sont très utilisées en imagerie médicale.

Le modèle par arbre B-rep (pour *Boundary Representation*) utilise une segmentation en régions de la scène et l'information de frontières qui s'en déduisent sur les surfaces.

Enfin, un autre type de représentation discrète beaucoup plus orienté reconnaissance de formes repose sur la projection de l'objet et de son aspect selon une

vue particulière de l'objet 3D, dans la mesure où l'oeil humain ne percevra jamais d'un seul coup l'objet en entier. En effet, les objets nous apparaissant par projection à travers des vues 2D, il n'est notablement pas nécessaire de connaître toute la géométrie d'un objet pour le reconnaître. Les objets sont alors décrits par des graphes d'aspects.

2.2.3.2 Modèles continus par paramètres

Les surfaces dites démoulables se prêtent quant à elles assez bien à une représentation sous forme paramétrique, notamment en CAO, où les modèles les plus utilisés sont les courbes et surfaces splines, B-splines comme les courbes de Bézier, qui sont définies comme une somme pondérée de fonctions polynomiales régies par un ensemble de points de contrôle et par le degré désiré pour représenter la courbe [Leo91][dC92]. Les "*Non Uniform Rational Basic Splines*" [PT95] (voir figure 2.10) sont des généralisations des courbes de B-splines classiques permettant de représenter plus facilement des formes libres. Les *NURBS* constituent un ensemble de techniques pour l'interpolation et l'approximation des courbes et des surfaces, dans la continuité desquelles on trouvera également des produits tensoriels de deux courbes définissant des carreaux (de Coons par exemple [dC92]). Avec ces types de modèle en général, la solution au problème de détermination de l'ensemble des facteurs correspond à la résolution d'un système linéaire associé à une énergie de déformation. Ce sont des méthodes assez lourdes au niveau calculatoire et risquées sur des données quelconques.

On trouvera plus précisément dans le cas d'objet isolé dont la topologie est *a priori* connue, des méthodes de calcul itératif de la meilleure hyperquadrique ou superquadrique s'adaptant au nuage de points 3D [HGB93][SLM94].

2.2.4 Conclusion

Actuellement, on ne peut prétendre pouvoir visualiser automatiquement qu'uniquement des objets simples, et difficilement des scènes complexes. Plus précisément, au niveau de la modélisation, il existe deux types d'environnement principaux :

– les environnements conçus par l'homme comme des usines, centrales, stations orbitales etc., donc avec modèles *a priori* ;

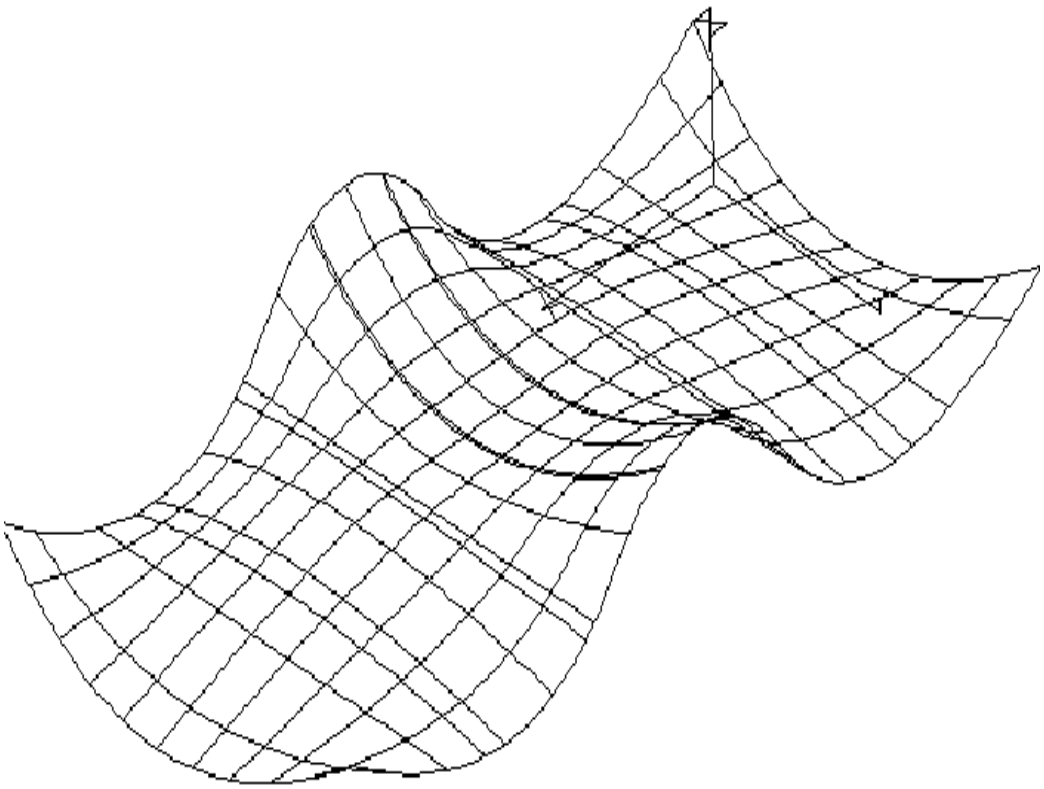

FIG. 2.10 – Illustration d'une visualisation par surface NURBS

– les environnements inconnus ou partiellement connus, comme des sites urbains, des planètes telles Mars etc., donc sans modèle *a priori*.

A ces deux classes d'environnement, et en se restreignant à des objets isolés, correspondent deux modes de visualisation :

– la visualisation de formes libres, concernant tous types de surface pour lesquelles on utilisera alors la modélisation géométrique par facettes de type triangulation, ou des modèles mathématiquement plus élaborés comme les surfaces splines ou de Bézier dans les cas les plus simples ;
– la visualisation de formes géométriques, concernant des objets ou scènes modélisables par un assemblage de primitives géométriques simples (plan, sphère, tore, cône, sphère), dont le modèle CSG est le plus courant des représentants.

Un certains nombre d'états de l'art sur ces techniques ont été publiés dans [Jar83][Jar93][Bes89][ANV90].

Au delà de la simple visualisation d'objet, les ambitions actuelles, s'appuyant sur les avancées technologiques, s'orientent vers des systèmes de modélisation automatique de scènes. A terme, on espère pouvoir obtenir une visualisation de haut niveau d'une scène complexe, non seulement en termes de facettes, mais surtout d'objets.

Notre objectif dans cette thèse consiste à interpréter des données d'entrée représentant un monde en trois dimensions quelconque, et donc d'obtenir un modèle de scène plus qu'un modèle d'objet. Le chapitre suivant décrit la plupart des travaux existants à ce jour dans le domaine de la reconstruction - au sens de l'interprétation - de scène en trois dimensions.

Chapitre 3

Interpréter des données 3D

3.1 Introduction

La reconstruction - au sens de l'interprétation - de scène en trois dimensions est un enjeu majeur de la vision artificielle et notamment en robotique mobile. L'intérêt que les chercheurs lui ont porté a permis de développer des outils spécifiques issus entre autres de la géométrie algorithmique ou d'adapter des méthodes plus classiques de traitement d'image 2D au problème spécifique de la reconstruction. Dans notre application principale, nous disposons d'un nuage de points 3D relativement dense reconstruit par des méthodes stéréoscopiques à partir des deux vues de la scène issues d'un système de capteurs 2D. Nous avons donc à notre disposition essentiellement des nuages de points très mal conditionnés en terme de densité, d'homogénéité et de précision (cf. figure 3.1). Les techniques décrites tentent pour l'essentiel d'interpréter de tels nuages de points. Nous supposons leur applicabilité à des nuages mieux formés issus de télémètres lasers par exemple.

Le panorama des recherches révèle alors deux tendances :

1. soit travailler sur les images 2D de la paire stéréoscopique pour y apparier des primitives de haut niveau comme les régions ou les contours ;
2. soit travailler sur le nuage de points 3D directement et le segmenter en objets sans référence explicite à la paire stéréoscopique.

Cette première distinction établie, on distinguera également les approches utilisant un modèle précis mathématique ou sémantique des objets à détecter de celles n'en disposant pas.

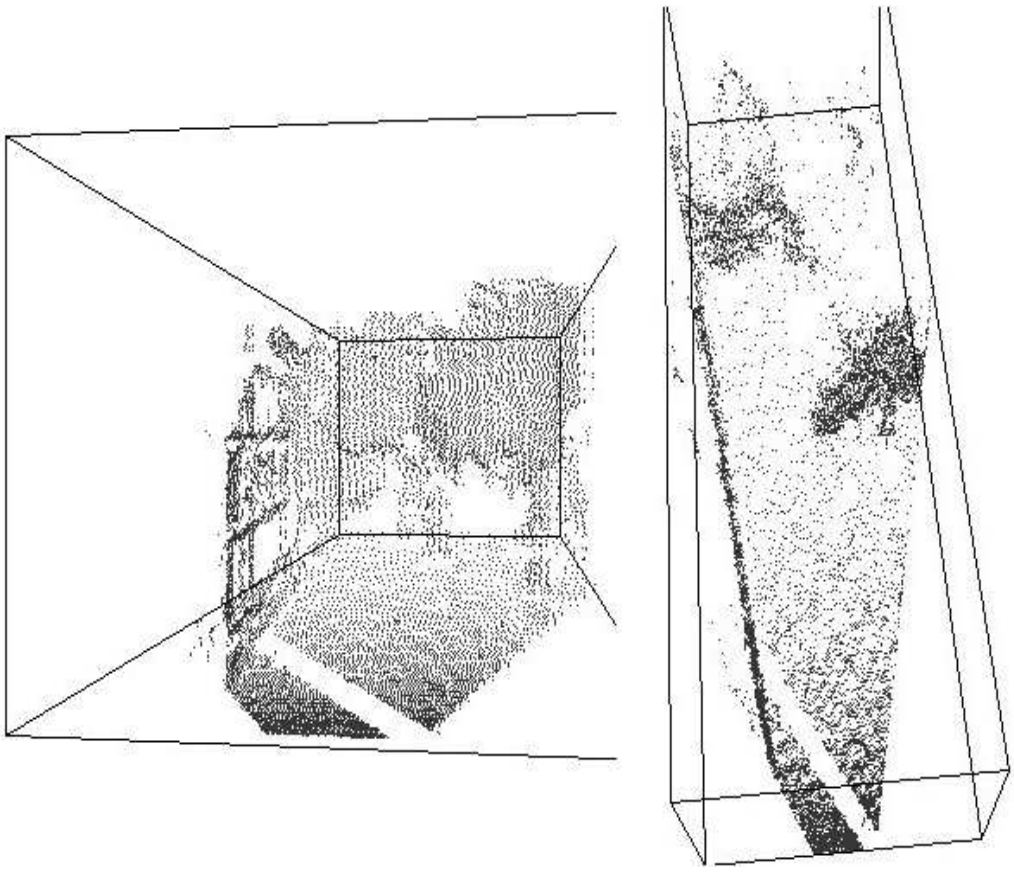

FIG. 3.1 – Un exemple de nuage de points 3D désorganisé et la scène observée par le capteur

3.2 Reconstruction à partir de la primitive région

En stéréo-reconstruction, l'approche utilisant des modèles mathématiques comme les quadriques a très largement été utilisée pour obtenir des cartes de disparité dense par exemple. Sur une carte de profondeur comportant de très nombreux points non appariés, on arrivait alors par interpolation à *remplir les trous*. De plus, la primitive géométrique la plus simple pour faire de la photométrie est la facette 3D. Aussi malgré l'utilisation effective d'autres primitives pour la reconstruction comme les points, les segments et les courbes, une large littérature s'est attachée à tenter d'extraire de la primitive région l'information 3D convoitée. Dans un premier temps, l'environnement était modélisé par un assemblage de surfaces planes 3D dont les projections sur la rétine de la caméra constituaient les régions de l'image 2D.

3.2.1 Modèles plans

Une idée simple, défendue par l'équipe du projet SYNTIM à l'INRIA, envisage l'environnement 3D observé sous la forme d'une juxtaposition de morceaux de plans. Dans cet univers plan , on travaille principalement sur les images 2D, en incorporant des considérations géométriques sur la forme des régions projetées ou photométriques sur la distribution d'intensité de ces régions, supposées être la projection d'une même facette d'un objet de l'espace 3D. Les régions doivent alors être appariées. De façon avantageuse, ces méthodes de segmentation en régions réalisent simultanément une partie du travail de reconstruction puisqu'on espère par ces considérations retrouver l'équation du morceau de plan 3D correspondant.

C'est pourquoi elles furent abondamment étudiées dans les années 90, et plus précisément pour des problèmes de reconstruction 3D d'environnements très structurés comme l'intérieur de bâtiments par exemple. Cette idée dégage deux objectifs légèrement distincts : l'un compte piloter la segmentation en régions par l'utilisation de contraintes stéréoscopiques [Ran92][Lut93], pendant que l'autre suppose la segmentation et la mise en correspondance des régions dans la paire stéréoscopique acquises et cherche à trouver les relations mathématiques inter-régions en géométrie projective permettant de reconstruire l'univers plan 3D sous-jacent[Véz95][Tar96a][Tar96b].

3.2.1.1 Segmentation en régions pilotée par l'appariement stéréoscopique

La segmentation d'une image en régions est un processus de description d'une image par des primitives de plus haut niveau que la primitive pixel. Mais la difficulté de définir de telles primitives, notamment en fonction de leur usage dans la suite du traitement, rend ce processus extrêmement complexe et soulève le paradoxe suivant : pour bien segmenter, il faut savoir ce que l'on cherche, mais pour l'obtenir il faut l'avoir bien segmenté. Partant de cette constatation, l'idée de coupler le processus de segmentation à un processus en amont vient naturellement à l'esprit pour gagner en robustesse. Or, la plupart des algorithmes de segmentation sont guidés par des modèles (de textures par exemple), et ceci de façon exclusive, puisque le but ultime est de classifier l'image en différentes zones. Dans le cas de la stéréoscopie, le but à atteindre est moins un but de classification (objectif intermédiaire) que l'appariement des zones segmentées *en vue de la reconstruction 3D*. On a donc le moyen de piloter la segmentation en régions non seulement par le modèle image mais également par le processus d'appariement en amont. C'est la démarche entreprise dans [Ran92] et [Lut93].

Dans [Ran92], Sabine Randriamasy développe un processus de segmentation descendante coopérative en régions d'une paire d'images stéréoscopiques. L'intérêt de la méthode réside dans sa propension à alterner étape de segmentation et processus d'appariement : pour cela il est nécessaire de conserver deux hiérarchies de graphes d'adjacence afin de :

- trouver des correspondants à travers plusieurs niveaux de segmentation ;
- choisir les meilleurs appariements possibles en revenant sur une décision d'appariement.

Pour l'appariement de régions similaires dans les deux hiérarchies de segmentations, on utilise des caractéristiques de régions de diverses natures : photométriques (maximum et minimum d'intensités, moyenne et variance de l'intensité), géométriques (surfaces, centre de gravité, moments d'inertie), topologiques (frontières de régions, longueur et contraste moyen le long de cette frontière), les relations d'adjacences et hiérarchiques entre régions (père et fils), les descripteurs de forme (frontière, périmètre, compacité) qui permettent de définir une mesure de dissimilarité entre régions de la paire stéréoscopique. On cherchera à minimiser la dissimilarité entre une région de l'image droite et une région de l'image gauche à apparier sous certaines contraintes notamment de régularité de frontières

au niveau des contours.

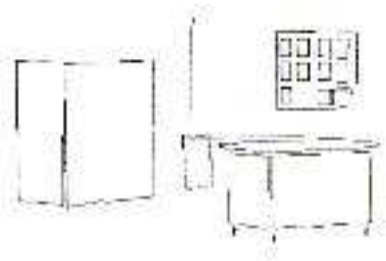

FIG. 3.2 – Reconstruction 3D d'une scène d'intérieur par appariement stéréoscopique de régions planes

La méthode de segmentation pilotée par l'appariement exposée précédemment est tout à fait séduisante dans un univers plan . Cependant, les algorithmes de segmentation sont peu fiables en environnement extérieur : à cause du caractère fortement texturé (feuillage, nappe de sol...) et de la profondeur (gradient d'intensité constant et non nul) des objets en présence. Heureusement, le choix de l'algorithme de segmentation est indépendant de la méthode employée. C'est-à-dire qu'à chaque étape, la nouvelle segmentation fournie au processus d'appariement hiérarchique peut être obtenue par un autre algorithme.

Partant d'une paire d'images segmentées en régions appariées et de la connaissance de tous les paramètres de caméra et de prise de vue, Evelyne Lutton propose dans [Lut93] d'améliorer cette segmentation en régions supposées planes en prenant en compte des contraintes de cohérences 3D, notamment au niveau de la rétro-projection correcte du plan 3D reconstruit dans la paire stéréoscopique segmentée. Elle utilise une modélisation par champs de Markov et la souplesse de définition des cliques pour définir un voisinage étendu aux pixels appariés dans les images de la paire stéroscopique. C'est une méthode plus complexe mais qui semble donner de bons résultats en reconstruction 3D d'environnement intérieur (voir figure 3.2).

3.2.1.2 Reconstruction 3D de scènes à partir de la paire d'images stéréoscopiques segmentée

Dans les méthodes précédentes, on utilise la primitive région 2D pour reconstruire la scène en trois dimensions. Une étude plus formelle de cette hypothèse de

travail fut par la suite menée. Supposons l'appariement de régions acquis et recherchons les appariements correspondant à des surfaces planes dans l'espace de la scène (typiquement des murs ou des portions planes d'objets naturels). Les méthodes varient en fonction de l'espace dans lequel on travaille (espace de l'image [Véz95], espace des disparités [Tar96a], carte des disparités [Luo91]) et des informations que l'on extrait de la primitive région (géométriques, photométriques, duales avec les contours...). On se placera toujours dans le cadre de la géométrie rectifiée où les équations se simplifient.

Jean-Marc Vezien [Véz95] a exploré une approche de type "appariement de primitives" sous contrainte planaire au niveau des stéréo-régions segmentées et appariées. En fait, on considère que chaque région d'une image 2D correspond à la projection d'une surface plane ou assimilable P de la scène observée. Il s'agit alors de retrouver l'équation du plan P dans $\Re^3$. Si nous nous limitons au cas de la géométrie rectifiée, il existe une fonction de correspondance pixel à pixel (entre l'image de gauche et l'image de droite) qui se linéarise automatiquement et permet de passer à un modèle affine pour lequel les invariants sont très riches (en particulier les invariants utilisant les moments d'inertie de régions planes). Une fois obtenue l'expression des transformations affines permettant de passer d'un point de l'image de gauche à un point de l'image de droite, on dérive les équations de mise en correspondance d'attributs géométriques globaux entre deux régions appariées, ce qui fournit une solution analytique au problème d'estimation des paramètres définissant le plan P à reconstruire.

Posons quelques notations.

$M^g = (X^g, Y^g, Z^g)$ et $M^d = (X^d, Y^d, Z^d)$ sont les coordonnées cartésiennes du point M exprimées dans les repères liés aux caméras gauche et droite, respectivement. $m^g = (x^g, y^g, 1)$ et $m_d = (x^d, y^d, 1)$ sont les coordonnées des projections de M dans les plans focaux gauche et droit, respectivement. L'image $m(x, y)$ formée à la distance focale f (prise égale à l'unité) du centre optique O, d'un point $M(X, Y, Z)$ est donnée par :

$$\begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{Z} \begin{pmatrix} X \\ Y \end{pmatrix} \tag{3.1}$$

P est le plan passant par le point M d'équation $Z^g = p^g X^g + q^g Y^g + c^g$ et $Z^d = p^d X^d + q^d Y^d + c^d$ dans les repères liés aux caméras gauche et droite, respectivement.

l est la distance inter-caméras, et on pose $P = p^d/c^d$, $P' = 1 - P * l$, $Q = q^d/c^d$ et $Q = Q * l$.

On note $\mathbf{F}$ la transformation pixel droit/pixel gauche, soit :

$$\begin{pmatrix} x^g \\ y^g \end{pmatrix} = \mathbf{F} \begin{pmatrix} x^d \\ y^d \end{pmatrix} \tag{3.2}$$

3.2.1.3 Expression de le transformation affine

Dans le cas de la géométrie rectifiée, la transformation $\mathbf{F}$ est affine et

$$\begin{pmatrix} x^g \\ y^g \end{pmatrix} = \begin{pmatrix} 1 - \frac{p^d}{c^d} l & -\frac{q^d}{c^d} l \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x^d \\ y^d \end{pmatrix} + \begin{pmatrix} \frac{l}{c^d} \\ 0 \end{pmatrix} = \mathbf{F} \begin{pmatrix} x^d \\ y^d \end{pmatrix} + K \tag{3.3}$$

On retrouve l'expression classique de la disparité en géométrie rectifiée :

$$d(x^g, x^d) = x^g - x^d = \frac{l}{Z} \tag{3.4}$$

A ce stade, il est possible d'appliquer ces formules soit à un ensemble de couples de points $(m^g, m^d)_i$ mis en correspondance, afin de trouver le meilleur triplet (p, q, c) respectant la série d'équations $m_i^g = \mathbf{F}_{pqc}(m_i^d)$ soit aux couples de régions appariées $(\mathbf{R}^g, \mathbf{R}^d)$ dans leur ensemble en les décrivant par leurs moments géométriques. On montre alors qu'on peut exprimer les relations liant les moments géométriques C_{ij} d'ordre $n(n <= 2)$ de deux régions appariées dans un couple stéréoscopique, au moyen des fonctions $\mathbf{F}(p, q, c)$ obtenues précédemment. La donnée des C_{ij} sert alors d'appariement global et fournit le plan $P(p, q, c)$, solution de façon analytique.

Moment d'ordre 0 : les surfaces S^g et S^d des surfaces gauche et droite sont liées par la relation :

$$S^g = |Jac(F)| S^d = |P'| S^d \tag{3.5}$$

Moments d'ordre 1 : par triangulation des centres de gravité des deux régions appariées, on obtient un point référence M_o estimé appartenir au plan solution ce qui donne c en fonction du vecteur normal au plan (p, q) :

$$c^d = Z_o^d - p^d X_o^d - q^d Y_o^d \tag{3.6}$$

Moments d'ordre 2 : les moments d'inertie sont représentés sous forme d'une matrice symétrique I :

$$I = \begin{pmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{pmatrix} \tag{3.7}$$

où les matrices d'inerties sont liées par la relation :

$$I^g = F I^d F^T \tag{3.8}$$

ce qui fournit le système d'équations suivant :

$$\begin{pmatrix} \left(I^d_{xx} - \frac{(I_x y^d)^2}{I^d_{yy}}\right)(P')^2 = I^g_{xx} - \frac{(I_x y^g)^2}{I^g_{yy}} P' > 0 \\ Q' = -\frac{I x y^g}{I^g_{yy}} + \frac{I x y^d}{I^d_{yy}} P' \end{pmatrix} \tag{3.9}$$

Une telle méthode de reconstruction géométrique utilisant la forme des régions droite et gauche est malheureusement sensible aux occultations de régions d'une vue à l'autre, ce qui n'est pas le cas de l'information de gradient d'intensité contenue dans les régions.

C'est pourquoi, l'auteur utilise également des informations photométriques globales contenues dans les régions pour les mettre en relation : les paramètres du profil photométrique (lié au gradient d'intensité par exemple) ou les cartes d'auto-corrélation des intensités. En effet, dans le cas de surfaces stéréo-lambertiennes, l'hypothèse de planarité se traduit au niveau de l'intensité par une distribution linéaire de l'intensité décrite par la formule $I(x, y) = \alpha x + \beta y + \gamma = \vec{k}.\vec{m}$. Le lien entre $\vec{k}^d$et$\vec{k}^g$ s'écrit $\vec{k}^d = F^T \vec{k}^g$. De même, si $C(\vec{h})$ désigne l'auto-corrélation de I par le vecteur $\vec{h}$, on a $C^d(\vec{h}^d) = C^g(F\vec{h}^g)$, et on estime le plan (p, q, c) par minimisation d'une énergie :

$$\epsilon = \sum C^g(F\vec{h}^g) - C^d(\vec{h}^d) \tag{3.10}$$

Jean-Philippe Tarel [Tar96a][Tar96b] poursuit cette démarche d'estimation d'une transformation affine entre régions appariées en géométrie rectifiée, mais se place d'emblée dans l'espace des disparités (x^d, x^g, y) dans lequel les équations qui lient l'espace 3D aux projections 2D sont plus simples que celles décrites dans l'espace réel. Cet espace possède en outre une interprétation géométrique intéressante facilitant parfois le raisonnement. Il est en effet possible d'interpréter la géométrie rectifiée de deux façons. Elle peut être vue comme une projection perspective sur deux images dans un même plan, ou bien comme une projection orthographique sur chaque image, si elles sont placées perpendiculairement l'une

à l'autre. Dans le deuxième cas, l'espace entre les images est celui des disparités. Cette double interprétation est la traduction géométrique de l'isomorphisme entre l'espace réel des points visibles de la scène et l'espace des disparités. Dans ces travaux, on dérive des relations sensiblement équivalentes mais effectivement plus simples pour déterminer le plan reconstruit dans l'espace des disparités. L'auteur sélectionne ensuite les facettes convenablement reconstruites par la méthode géométrique. Les régions occultées et rejetées par la méthode géométrique sont alors reconstruites en utilisant l'information photométrique de gradient d'intensité dans les régions. Si I^d et I^g sont les distributions d'intensités des régions gauche et droite, on suppose que la distribution est planaire. Les surfaces sont donc lambertiennes et les sources lumineuses éloignées, on a alors :

$$I^g = \alpha^g x^g + \beta^g y^g + \gamma^g I^d = \alpha^d x^d + \beta^d y^d + \gamma^d \tag{3.11}$$

En faisant l'hypothèse habituelle de la reconstruction photométrique selon laquelle l'intensité d'un point est indépendante du point de vue, on obtient l'équation du plan solution dans l'espace des disparités en égalant simplement I^d et I^g :

$$\alpha^g x^g - \alpha^d x^d + (\beta^g - \beta^d) y + \gamma^g - \gamma^d = 0 \tag{3.12}$$

Wei Luo [Luo91] a lui aussi étudié l'utilisation de modèles de surface pour la reconstruction tridimensionnelle en travaillant directement sur la carte de disparité. L'image 2D est segmentée en n régions ou objets. On désigne par $d_k(i,p)$ la fonction de disparité, où k désigne l'objet, i le pixel et p les paramètres de la fonction de disparité, le problème est maintenant de déterminer les paramètres p afin de reconstruire complètement les objets. On a en outre une carte de disparité $d(i)$ définie pour chaque pixel i d'un sous-ensemble J de l'ensemble des pixels I de l'image (les mesures sont souvent incomplètes et la carte de disparité n'est pas toujours dense). On désigne par J_k la restriction dans J des pixels de l'objet k et par $\delta_k(i)$ la réduction de la carte de disparité à l'objet J_k. A la condition que J_k soit suffisamment grand, c'est-à-dire qu'il y ait assez de points de disparités connus, il est possible de déterminer les paramètres p en minimisant la fonction suivante :

$$\Lambda = \sum_{i \in J_k} [\delta_k(i) - d_k(i,p)]^2 \tag{3.13}$$

Soit p_o les valeurs de p minimisant Λ. A partir des valeurs Λ et p_o , nous pouvons savoir si le modèle est adapté à la surface d'objet, et prendre la décision de le garder ou de le rejeter. Dans le cas de la surface plane définie ci-dessus, on a :

$$d(x^g, x^d) = \frac{l(f - p^g x^g - q^g x^d)}{c} \tag{3.14}$$

En effet, la disparité d'un plan dans l'espace tridimensionnel est une fonction linéaire dans le repère image. L'auteur ajoute également une limite physique sur le gradient de disparité. Le module du gradient est $G_d = \| \vec{\nabla} d \| = \frac{l}{c}\sqrt{p^2 + q^2} = S.\tan(\theta)$ où S est le rapport entre la distance inter-caméras et la distance du plan à la caméra de gauche et q l'angle entre la normale au plan et l'axe Z. En pratique, et dans un univers isotrope, ce gradient est inférieur à 2 avec une forte probabilité. On constate aussi que le laplacien de la fonction de disparité s'annule sur cette région. Une analyse fine de ces régions peut donner des indices très intéressants sur des modèles de surface à approximer. On peut alors voir le comportement de ces régions pour des modèles de surface plus complexes type superquadriques.

3.3 Reconstruction à partir de la primitive contour

Au lieu d'apparier les régions, on peut classiquement tenter l'approche duale : apparier les discontinuités des images qu'on aura au préalable extraites [Her91]. Ces méthodes se prêtent particulièrement bien en environnement d'intérieur assez structuré où la stéréo-corrélation point à point risque d'échouer la plupart du temps sur les surfaces peu texturées. Dans la thèse de Bruno Serra [Ser96], on utilise la programmation dynamique avec une précision subpixellique pour mettre en correspondance des contours dans deux images d'une paire stéréoscopique, en prenant notamment en compte les contraintes épipolaires appliquées sous forme de zone épipolaire associée à chaque segment. Un résultat de reconstruction 3D est présenté en environnement d'extérieur structuré sur la figure 3.3.

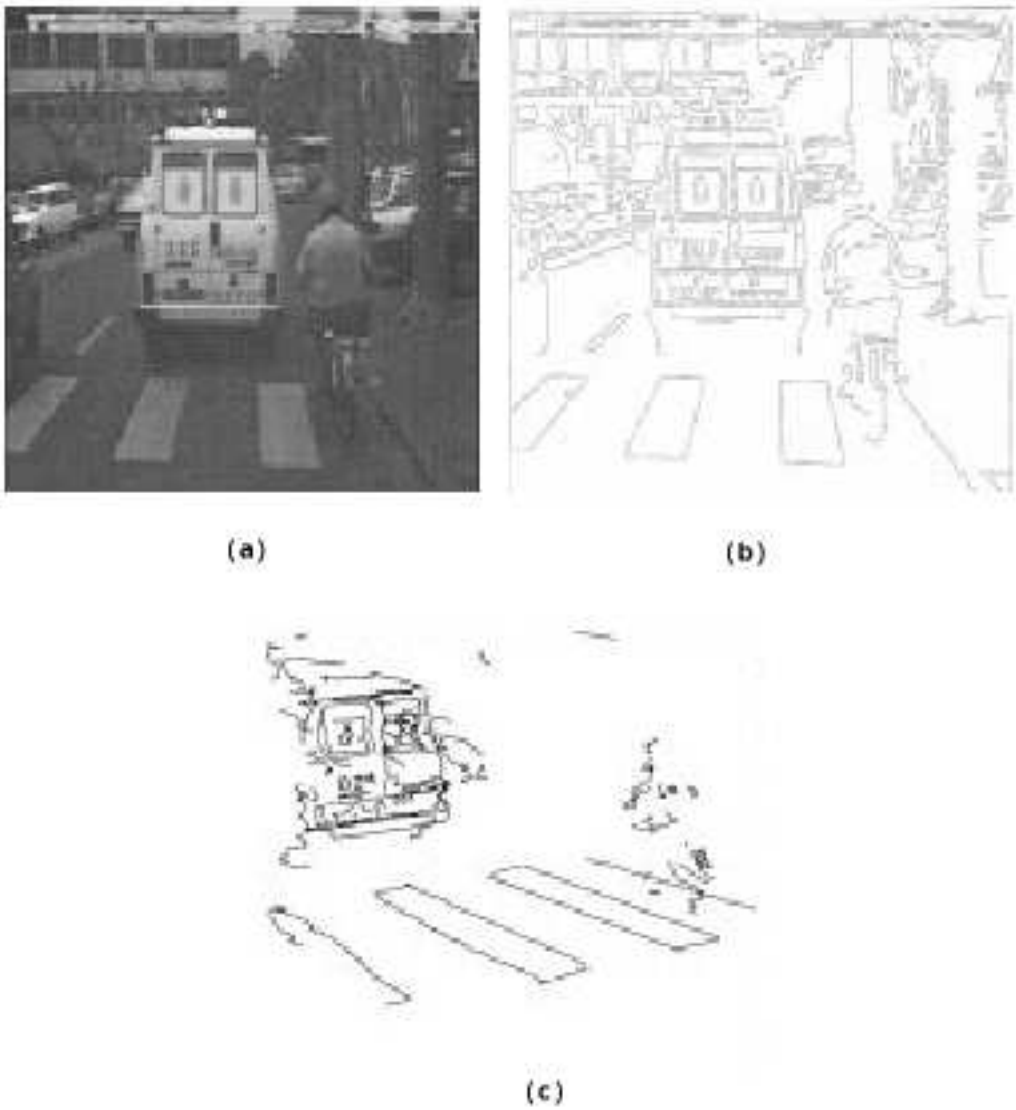

FIG. 3.3 – (a) Image d'intensité 3D d'une scène routière image (b) contours 2D extraits (c) vue 3D de la scène reconstruite

3.4 Reconstruction à partir de la primitive point

3.4.1 Segmentation d'un nuage de points 3D

On peut encore s'écarter totalement de la paire d'images stéréoscopiques en ne manipulant que le nuage de points 3D reconstruit. Cette démarche s'appuie habituellement comme dans [BJ88] sur l'utilisation de l'outil mathématique de géométrie différentielle . Il apparaît en effet que les primitives de description de surfaces 3D invariantes par excellence soient les normales et les courbures en chaque point de la surface. Ce moyen de description du nuage de points 3D permet de passer à un autre espace de représentation où les primitives géométriques choisies sont mises en relief comme dans l'espace des courbures par exemple ou encore sur la sphère gaussienne .

3.4.2 Segmentation d'un nuage de points 3D en primitives géométriques

3.4.2.1 Quelques rappels de géométrie différentielle

On considère une surface C_2 par morceaux. La géométrie différentielle permet d'obtenir des résultats indépendants du repère local et de la paramétrisation de la surface. Nous savons que toute courbe 1D sur la surface au point étudié possède une courbure qui peut se décomposer en partie tangentielle et partie normale à la surface K_n. On peut montrer que cette composante normale de la courbure (qu'on appelle par simplification courbure) ne dépend que de la direction considérée sur la courbe 1D dans le plan tangent ; qu'elle peut prendre toutes les valeurs entre une valeur minimale et une valeur maximale ; enfin, que les directions pour lesquelles elle prend une valeur minimale et une valeur maximale, appelées courbures principales K_{min} et K_{max}, sont perpendiculaires. Ces directions sont notées v_{min} et v_{max}.

On définit le repère de Darboux (v_{min}, v_{max}, n) (cf. figure 3.4) .

Sur des courbes on peut définir des centres de courbure, dans le cas des surfaces, on considérera le centre de courbure Cc associé à la courbure maximale, défini par :

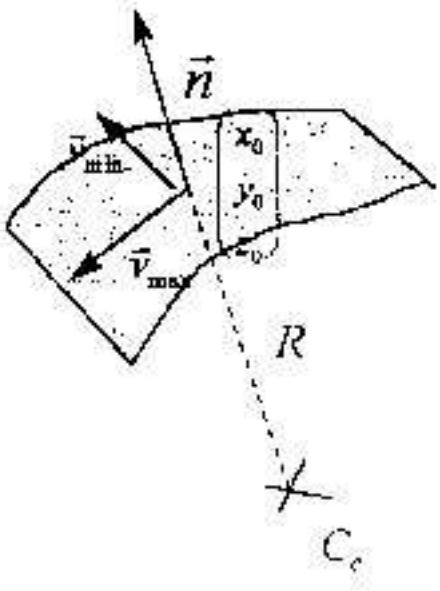

FIG. 3.4 – Repère de Darboux

$$\vec{C}c = \vec{C}c_{max} = \vec{X}_o - \frac{1}{K_{max}}\vec{n} \tag{3.15}$$

La plupart du temps on connaît les surfaces par leur équation implicite $F(x, y, z) = 0$ ou explicite $z = f(x, y)$. Cette dernière est la plus courante en imagerie $3D$. Dans le cas d'une surface $z = f(x, y)$, et en notant f_x, f_y, f_{xx}, f_{xy} et f_{yy} respectivement les dérivées premières par rapport à x et y, et secondes par rapport à xx, xy, et yy, on obtient les courbures K_{min} et K_{max} en un point (x, y) comme les racines d'un polynôme du second degré.

3.4.2.2 Quelques outils de visualisation

La visualisation de ces résultats est une étape importante pour mieux comprendre les phénomènes. Pour visualiser les résultats de dérivation à l'ordre 1 (les normales) on utilisera la "Sphère Gaussienne", pour visualiser les résultats de dérivations à l'ordre 2, on utilisera le "Graphe Global de Courbure".

La "Sphère Gaussienne" est la représentation, sur la sphère unité, du lieu des points de toutes les normales à la surface étudiée. L'intérêt de la sphère Gaussienne est particulièrement prononcé quand il s'agit de représenter des surfaces pas ou peu courbées, comme c'est le cas pour un ensemble de plans. En effet, toutes les normales à un même plan sont identiques, au sens près, ce qui fait qu'un plan est

représenté sur une sphère Gaussienne par deux points diamétralement opposés. Cette dernière propriété justifie l'utilisation de la sphère Gaussienne pour repérer la présence de plans dans une scène et en estimer le nombre. En outre, lorsque les surfaces ont une courbure régulière et simple, comme pour un cylindre, la représentation sur la sphère Gaussienne est également intéressante.

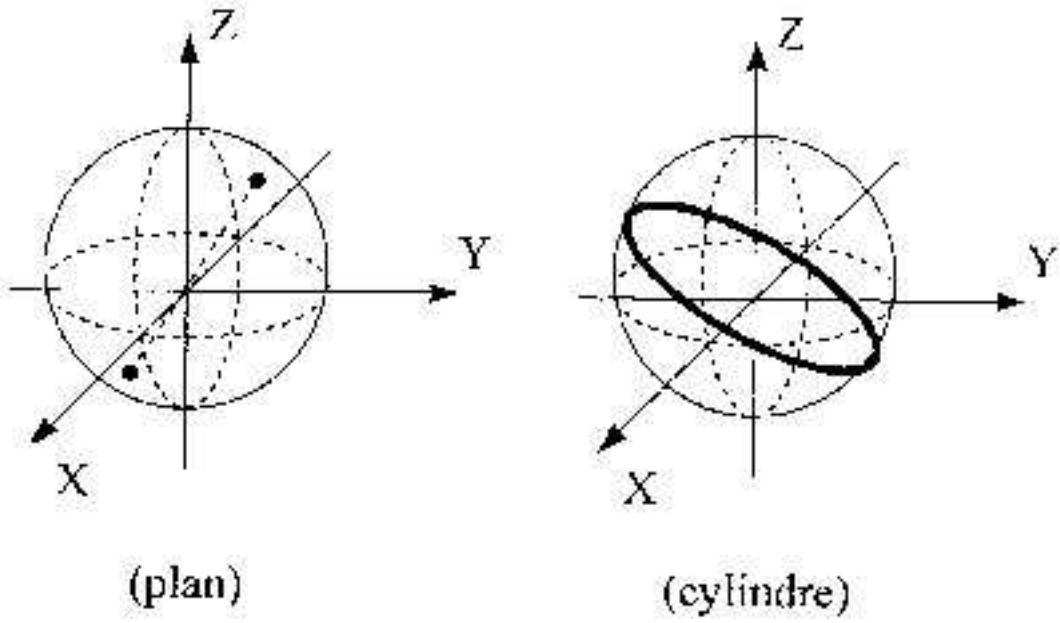

FIG. 3.5 – Sphère gaussienne et primitive géométrique

Le "Graphe Global de Courbures" - défini dans [Gou97] - a pour but de représenter de façon condensée tous les résultats de dérivations à l'ordre 2 en opérant une séparation entre grandeurs dimensionnelles (quantitatives) et adimensionnelles (qualitatives). Il consiste à tracer sur un même graphe tous les couples de points (X, Y) issus du calcul de courbures sur une image, et définis par :

$$\begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} K_{max} \\ K_{ratio} = \frac{K_{min}}{K_{max}} \end{pmatrix} \tag{3.16}$$

K_{max} est une valeur dimensionnelle qui indique la taille des courbures, et par définition K_{ratio} est compris entre -1 et 1, c'est une valeur adimensionnelle qui traduit l'apparence globale de la surface.

3.4.2.3 La recherche des primitives géométriques

Considérons les cinq primitives géométriques décrites en figure 3.6. L'usage de ce graphe global est un bon outil pour repérer des primitives géométriques

dans une image. Les valeurs des courbures maximales et minimales sur les cinq primitives précédentes ont une formulation très simple, puisque l'une au moins des deux est constante dans chaque cas. Il est alors facile de représenter sur le Graphe Global de Courbures les cinq primitives comme illustré en figure 3.6.

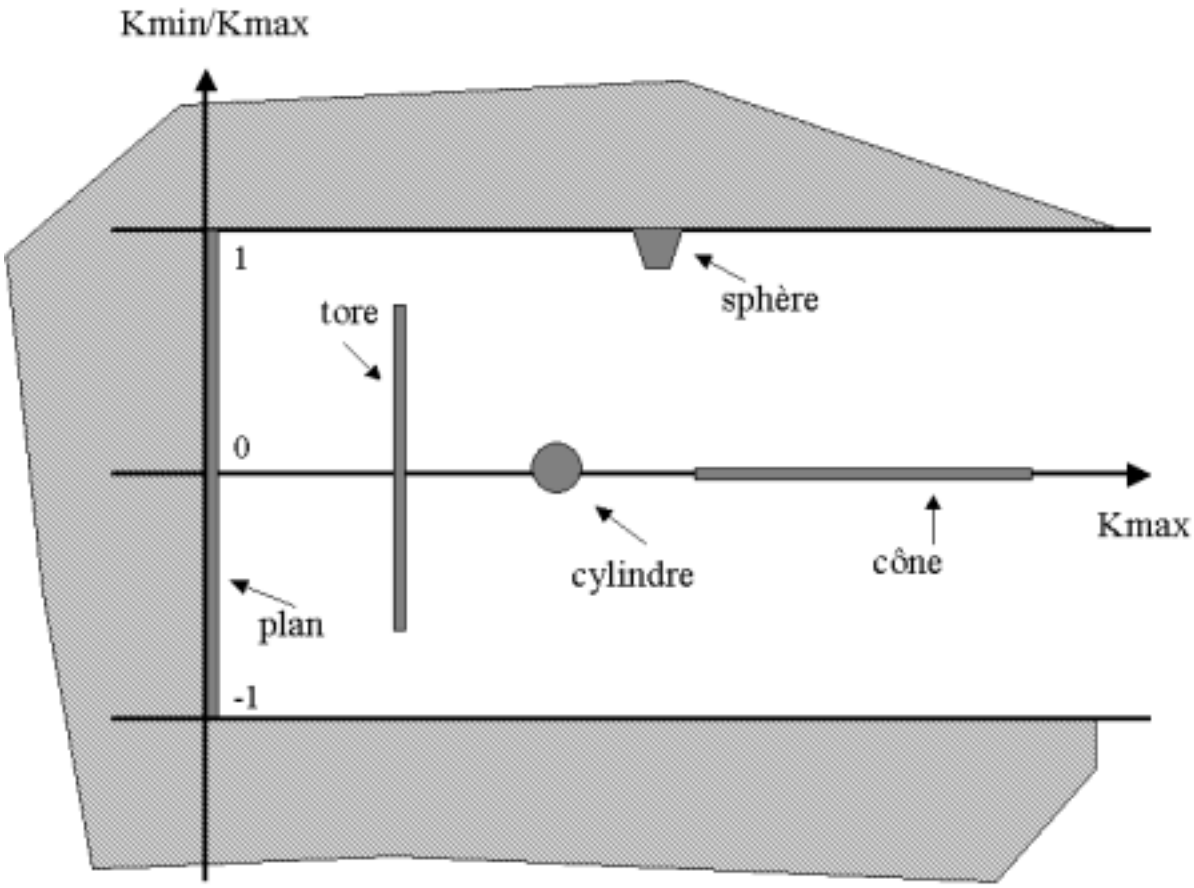

FIG. 3.6 – Graphe global de Courbures et Primitives géométriques

La représentation du plan est particulière : les deux valeurs principales de courbure sont nulles et le rapport K_{ratio} devient numériquement indéterminé tandis que K_{max} est proche de 0. La représentation d'un plan sur le Graphe Global de Courbures est alors un segment de droite vertical situé à $K_{max} = 0, K_{ratio} \in [-1, 1]$.

Dans la même veine, et dans le cas d'objet composé d'une seule surface, dans [CB00], on cherche dans un premier temps à structurer le nuage de points 3D fourni en entrée. Pour cela, on est obligé de l'immerger dans un système de voisinage. Deux méthodes sont employées : ou bien, considérer autour de chaque point une boule de rayon adaptatif qui emprisonnent un certain nombre de voisins les plus proches ; ou bien, utiliser une structure de données plus formelle type trian-

gulation de Delaunay. Au vu des résultats, l'auteur donnera une préférence à la deuxième façon de procéder. A partir de ce système de voisinage, on estime une approximation analytique de la surface locale selon un modèle de quadrique par exemple, avant d'en dériver les informations différentielles comme la courbure moyenne H ou la courbure gaussienne K. L'auteur utilise la classification décrite en figure 3.7.

	K = 0	K > 0 (elliptique)	K < 0 (hyperbolique)
H = 0	plan	indéterminé	surface minimale
H > 0	cylindre concave	vallée	col concave
H < 0	cylindre convexe	pic	col convexe

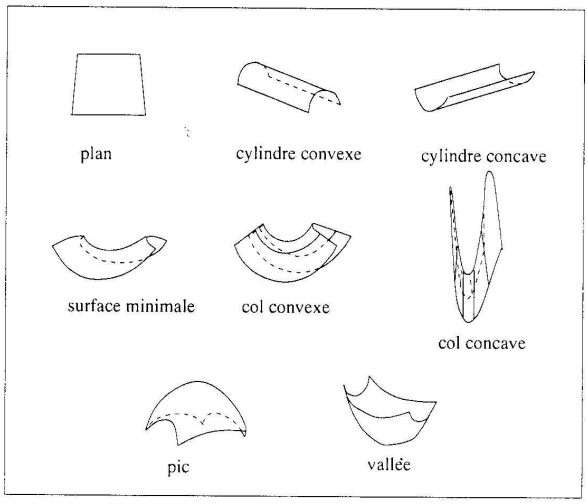

FIG. 3.7 – Classification des éléments de surface en fonction des valeurs de courbure moyenne H et gaussienne K

Une fois obtenue une telle description locale de la surface, on peut alors caractériser les irrégularités de la surface échantillonée et l'absence de satisfaction d'un critère dit d'homogénéité locale entre voisins. Enfin, le problème de la répartition des points du nuage en parties lisses bordées par les non-homogénéités locales de la surface qu'ils représentent est modélisé sous forme markovienne . L'énergie de ce système est définie à partir de cliques appartenant à deux structures de graphe particulières : les arbres d'escarpement minimal pour les parties lisses et maximal pour les irrégularités. Le modèle d'évolution proposée est une simplification des opérations de mise à jour des étiquettes dans ces arbres par propagation anisotrope

le long de leurs arêtes. En résumé, le graphe Γ modélise les relations de voisinage entre les points d'une surface, $V(s,r)$ désigne le coût assigné à chacune des arêtes de ce graphe. Une fonction de coût utilisée dans la définition de l'énergie globale peut correspondre par exemple à une discontinuité locale du plan tangent ou à la présence d'une zone de forte courbure entre deux points voisins sur la surface. Si l'on désigne par $\vec{N_s}$ et $\vec{N_r}$ les vecteurs normaux aux plans tangents aux points n et s respectivement de l'arbre, la fonction de coût s'exprime par la relation :

$$cot(P_s, P_r) = \frac{(|\vec{N_s}.\vec{N_r}| - \epsilon)}{1 - \epsilon} \tag{3.17}$$

où le paramètre ϵ intervient sur le caractère plus ou moins strict du critère d'homogénéité local choisi. Le problème de la minimisation ou de la maximisation de l'énergie U se ramène alors à la minimisation ou à la maximisation du coût total du graphe Γ ainsi valué selon l'illustration en figure 3.8.

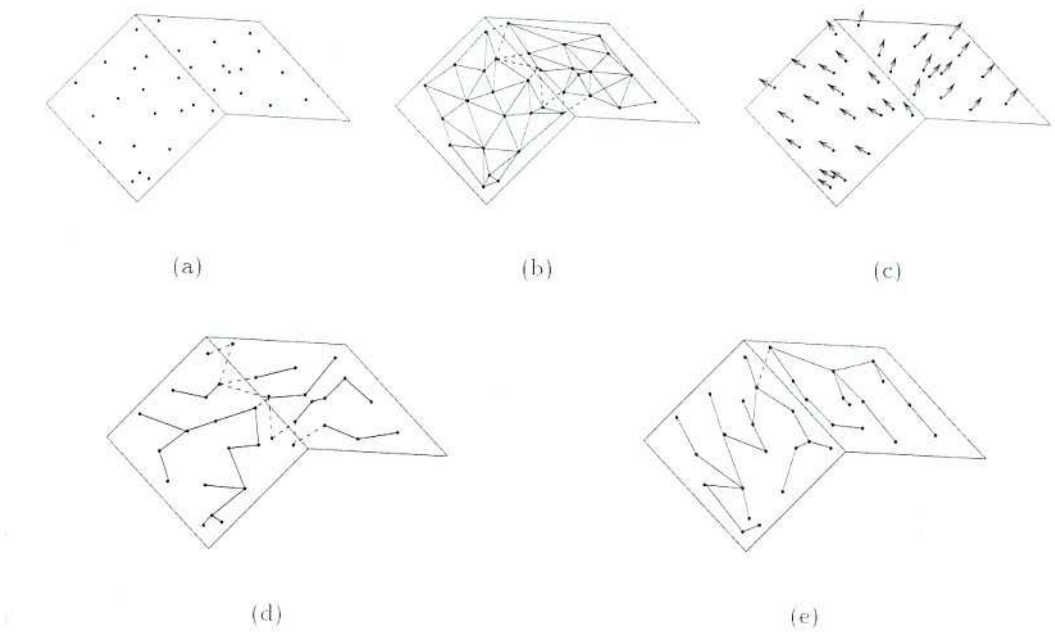

FIG. 3.8 – Passage d'un ensemble de points non organisé de points (a) aux arbres d'escarpement extrémaux associés (d) et (e) en passant par la construction d'un graphe de voisinage Γ (b) et l'inférence des propriétés locales permettant l'association d'une énergie de déformation à chaque arête de Γ (c)

Nous n'exposons pas le détail des étapes de parcours successifs de ces arbres d'escarpement extrémaux et reportons le lecteur à la thèse de Raphaëlle Chaine

[CB00]. En revanche, nous illustrons une partie des résultats obtenus dans ses travaux et qui montrent l'intérêt de son approche. Notamment en figure 3.10, l'auteur présente les résulats de sa méthode de segmentation sur une des images de profondeur de la base rendue disponible par l'université de South Florida (images de profondeur d'objets polyédriques). Il est intéressant de noter que cette initiative est encore rare au niveau de l'imagerie 3D et il faut lui rendre hommage. La mise en place de cette base a permis un programme de comparaison des méthodes de segmentation plane lancé pour *Iinternational Conference on Computer Vision* et *Pattern Analysis and Machine Intelligence* et poursuivi lors de *International Conference on Pattern Recognition 2000*. Retenons simplement que les méthodes présentées exploitent toutes largement le modèle polyédrique de la scène et le système de voisinage en 4-connexité fourni par l'image de profondeur, sauf pour la méthode de Raphaëlle Chaine.

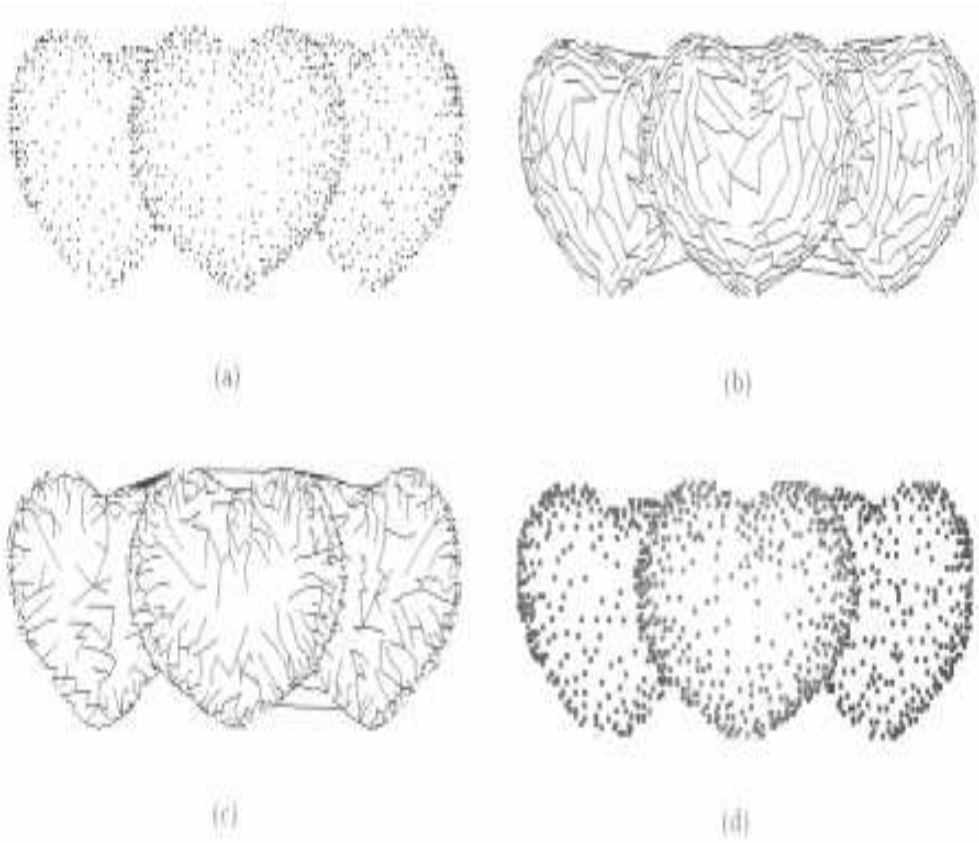

FIG. 3.9 – (a) Ensemble non organisé de 1059 points saisis à la surface d'un objet réel (Technodigit) (b) Arbre d'escarpement minimal (c) Arbre d'escarpement maximal (d) Résultat de la segmentation en utilisant l'information des normales

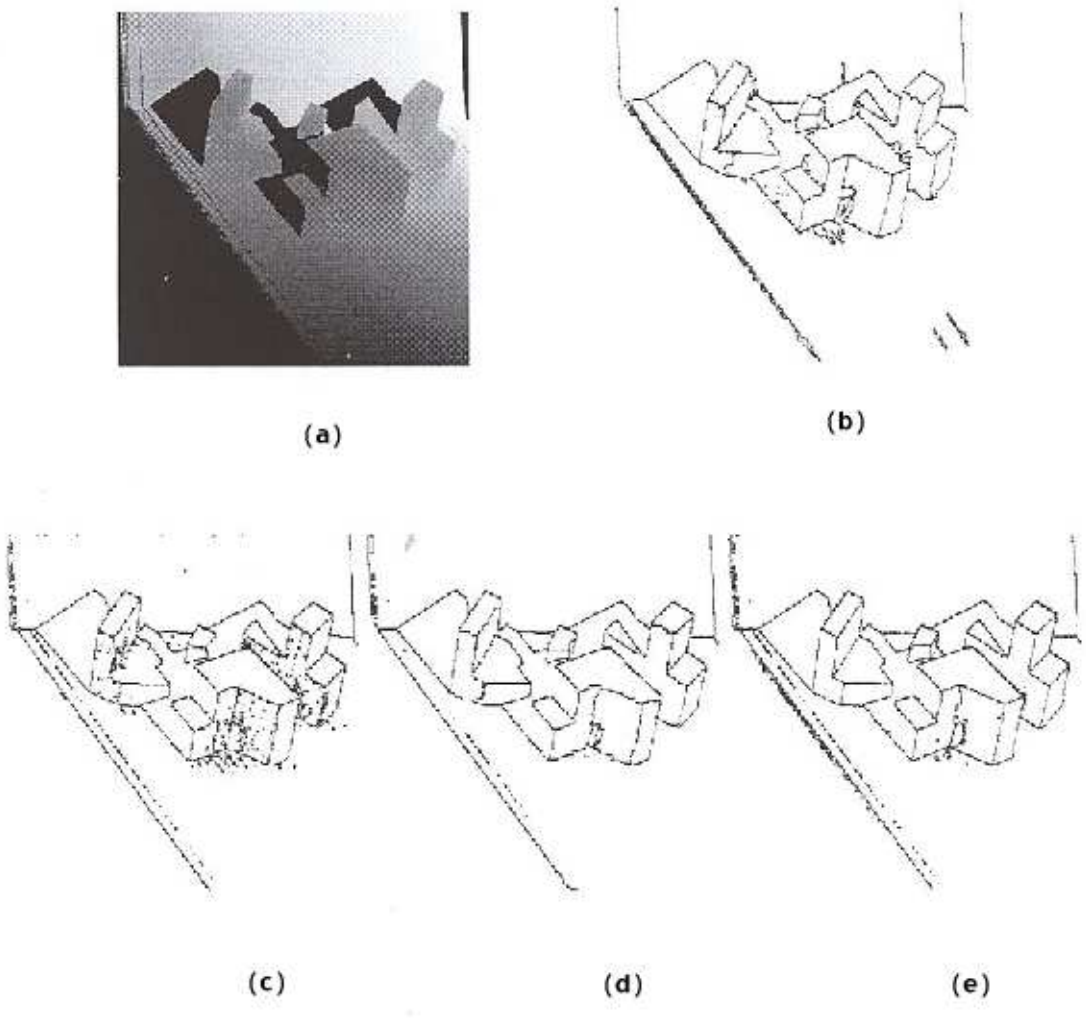

FIG. 3.10 – (a) Image de profondeur 'abw.train.1'. Segmentation de (a) par (b) l'université de Lyon 1 et la méthode de Raphaëlle Chaine (c) l'université de South Florida (d) l'université de Berne (e) l'université d'Edinbourg

3.4.3 Segmentation d'un nuage de points 3D en objets

Ces méthodes s'appuient toutes à un moment ou à un autre sur l'estimation en chaque point de la normale à la surface locale d'approximation optimale prenant en compte les points voisins immédiats dans le nuage de points.

Les travaux de P. Fua [Fua95] inspirés de R. Szelisky [ST92] offrent un outil de représentation à la fois plus souple que les triangulations de Delaunay pour la gestion de topologies multiples et plus aptes à remettre à jour son modèle de

la scène à mesure que d'autres vues stéréoscopiques sont disponibles. Un autre avantage est que la segmentation de la scène se fait en même temps que la reconstruction. Par contre, le résulat final s'apparente plus à un outil de visualisation sophistiqué qu'à une véritable méthode de reconstruction puis qu'on ne récupère en sortie aucune structure globale de représentation. Il s'agit plus d'un outil de segmentation d'un nuage de points 3D en objets agissant localement.

En combinant une représentation à base de particules , un processus robuste d'estimation et de correspondance, et l'optimisation d'une fonction basée sur l'image, ils ont réussi à reconstruire des surfaces sans aucune connaissance *a priori* sur leur topologie et malgré le caractère bruité des données de reconstruction.

La méthode comporte trois étapes :

1. l'initialisation d'un ensemble de particules à partir des données 3D ;
2. l'optimisation de leur localisation ;
3. le regroupement en surfaces globales.

En testant cette méthode sur de nombreuses scènes complexes contenant différents objets, ils ont démontré sa capacité à fusionner les informations venant de nouvelles vues stéréoscopiques à mesure que le robot progresse dans son environnement.

Par ailleurs, plusieurs types de difficultés sont rencontrés. D'une part, les scènes du monde réel contiennent plusieurs objets de topologies inconnues à l'avance : des surfaces sont mieux modélisées par des plans tandis que d'autres sont plus proches de la topologie d'une sphère ou contiennent des trous. D'autre part, les points 3D issus de la carte de disparité forment un échantillonnage irrégulier et bruité des mesures de surfaces. Ce sont ces difficultés essentielles que la méthode des particules parvient assez bien à surmonter. Elle repose sur une représentation centrée sur les objets qui peut s'accommoder de surfaces de complexité arbitraire, et raffine cette description en minimisant une fonction objective qui combine des termes mesurant le caractère lisse de la surface et les corrélations des projections des particules sur la série d'images stéréoscopiques. Par ces dernières mesures, on retourne à l'image originale. Finalement, on impose une métrique sur l'ensemble des particules qui nous permet de les rassembler en surfaces globales significatives.

L'étape d'initialisation construit un maillage régulier 3D de l'espace de la

scène qui va permettre de générer un ensemble de particules régulièrement espacées à partir du nuage de points 3D irréguliers et bruités. Cette étape permet de stocker les points 3D dans les cases appropriées. En associant une surface locale (plans et quadriques) à chaque case contenant suffisamment de points, on génère une particule dont le centre est la projection du centre de la case sur la surface approximante, et dont l'orientation est donnée par la normale à cette surface en ce point.

L'étape de regroupement des particules isolées en entités plus globales est construite sur une règle de regroupement $\mathcal{R}$ entre deux particules. La distance utilisée pénalise plus lourdement la distance du centre d'une particule au plan tangent de l'autre particule plutôt que le long de ce plan tangent. Un seuil sur la distance ainsi définie limite la courbure de la surface globale sous-jacente à laquelle les particules peuvent appartenir. L'ensemble des données munies de cette relation $\mathcal{R}$ peut être à présent vu comme un graphe dont les composants connectés sont les surfaces que nous recherchons.

L'étape de raffinement fait un retour sur les images originales et atteste de la qualité de chaque particule. Pour chaque particule circulaire, on définit un terme stéréo, en projetant le disque 3D sur des éléments elliptiques 2D dans chaque image et en mesurant comment ces éléments sont corrélés. On autorise alors les particules à interagir entre elles et se réarranger pour minimiser un terme d'énergie qui est la somme d'un terme de corrélation d'intensité multi-image et d'un terme d'énergie de déformation, qui tend à renforcer la consistance entre particules voisines. Au cours de l'optimisation de ce terme d'énergie, les particules qui correspondent à une même surface globale sous-jacente vont "se coller" ensemble, et celles qui n'appartiennent pas à une même surface vont se déplacer dans des directions séparées.

La défaillance de l'algorithme semble se situer au niveau du mécanisme de regroupement qui emploie un seuil fixe et global : il faudrait le rendre adaptatif localement pour réaliser une segmentation complète. Il conviendrait donc de développer des techniques de regroupement plus "informées" et d'introduire des heuristiques plus sophistiquées pour remplir les trous dans l'image de disparité originale (par création de particules par exemple).

On citera encore les tenseurs développés par Gérard Medioni [MLT00].

L'ensemble de ces techniques supposent l'obtention d'une surface approxi-

mante locale pour le calcul des quantités différentielles du premier ordre (normale) ou du deuxième ordre (courbures). Par conséquent, cela suppose la construction périlleuse d'une structure de voisinage sur un ensemble de points désorganisé sans compter les imprécisions, qui rendent la détermination des courbures hasardeuses. Dans ces conditions, les limites entre objets risquent de ne plus être respectées puisque la segmentation n'a pas encore été effectuée. Nous tenterons dans notre approche de contourner ce problème.

3.4.4 Techniques à base de graphes explicites

Par exemple les travaux récents de R. Chaine et S. Bouakaz [CB00] utilisent une approche à base de graphes et de modélisation markovienne par relaxation pour réaliser l'analyse surfacique de données 3-D non structurées comme décrit précédemment.

Les diagrammes de Voronoï sont des structures de données très utiles, car elles permettent de représenter des relations de distance entre objets et des phénomènes de croissance [CJMS00]. Leurs propriétés mathématiques sont en outre très nombreuses et intéressantes [JDY95]. Mais les travaux d'interprétation de données à partir de ces graphes sont très peu nombreux. Ces structures donnent toutefois une structure de voisinage indispensable à un nuage de points 3D désorganisé comme en témoigne les travaux de R. Chaine [CB00]. Ils sont utilisés essentiellement à des fins de représentations géométiques de structures 3D sous forme de maillage lorsque les données brutes apparaissent sous forme de nuages de points.

3.5 Reconstruction 3D de nuage de points

Un des aspects fondamentaux de l'interprétation de scène 3D est le choix de la technique de modélisation 3D des structures observées, *une fois isolées*.

Lorsqu'il s'agit de robotique, on utilise généralement des cartes d'élevation 3D ou modèles numériques de terrains (MNT) construits à partir de données régulièrment réparties sur une grille figurant l'altitude mesurée en chaque point du sol. On fait alors appel à des techniques de représentation 3D par maillage relativement bien maîtrisées à l'heure actuelle. D'autant plus que l'on se contente de travailler en 2D puisque l'on considère que le terrain peut être représentée par une fonction explicite donnant l'altitude $z = f(x, y)$ en fonction des coordonnées

2D au sol. L'inconvénient de cette représentation réside entre autre dans son incapacité à représenter des murs verticaux par exemple, ou bien des structures plus complexes avec des avancées par exemple.

Une problématique essentielle en reconstruction de formes tridimensionnelles a été formulée de façon abstraite dans [HDD+92] : étant donné un ensemble de points dans $\Re^3$ désorganisé, comment obtenir un maillage polygonal représentant la forme de ce nuage de points sans struture ? Dans la pratique, des problèmes de reconstruction de cette sorte interviennnent dans divers contextes. Les données produites par les scanners laser représentent des grilles rectangulaires de distances du capteur à l'objet scanné. Si le scanner et l'objet sont fixes, seuls les objets peu complexes peuvent être entièrement digitalisés. Des systèmes plus sophistiqués sont capables de digitaliser des objets cylindriques en faisant tourner l'objet ou le capteur. En revanche, scanner une tasse de café avec une anse par exemple est encore un défi technologique. Pour reconstruire correctement ce type de surface complexe plusieurs numérisations 3D sont nécessaires selon différents points de vue, qu'il s'agit ensuite de fusionner sous la forme d'un nuage de points 3D désorganisé. On trouve un autre exemple en imagerie médicale. On obtient une description 3D de structures biologiques sous la forme d'empilement de contours 2D. Bien que ce problème ait été largement étudié, plusieurs limitations demeurent et notamment le traitement des structures contenant des embranchements [MSS91].

Le but de la reconstruction de surfaces consiste à déterminer une surface S' qui approxime une surface inconnue S en utilisant un échantillon X donné sous la forme d'un nuage de points 3D géométriques. Les techniques existantes peuvent être classifiées en fonction du choix de la représentation de surface reconstruite.

Les méthodes de reconstruction implicites essayent de trouver une fonction continue $f : \Re^3 -> \Re$ telle que l'ensemble de points échantillons X soit proche du noyau $Z(f)$ de cette fonction. Elles diffèrent en fonction de la forme de f et de la mesure de proximité [Pra87] [Mur91].

Parallèlement, les techniques de reconstuction paramétrique reposent sur une immersion topologique $f(\Lambda)$ d'un domaine paramétrique bidimensionnel Λ dans l'espace tridimensionnel. Les travaux existants n'ont utilisé que des domaines spatiaux de topologie simple comme le plan [VMA86] ou la sphère [Bri85]. Dans [SP91], on décrit une méthode hybride implicite/paramétrique pour faire correspondre une sphère déformée à un ensemble de points en utilisant une superqua-

drique déformée. On citera à ce propos les travaux de Delingette qui utilise un maillage déformable de topologie sphérique [Del94]. Enfin, les travaux de R. Keriven [FK98] tentent de combiner la dérivation de l'information 3D et la reconstruction des surfaces présentes dans la paire stéréoscopique, en utilisant les techniques à base d'Equations aux Dérivées Partielles et en se plaçant dans $\Re^4$.

Toutes ces méthodes souffrent à ce jour d'un déficit de robustesse, qui provient notamment du fait que le passage de problèmes définis en 2D aux mêmes problèmes définis en 3D soulève souvent non seulement des problèmes calculatoires mais aussi théoriques mathématiques et algorithmiques [ABK98]. Par ailleurs, elles utilisent la plupart du temps des propriétés spécifiques des nuages de points en entrée. H. Hoppes [HDD+92] contribue pour sa part à la compréhension globale du problème abstrait posé précédemment. Sa méthode s'applique à tout nuage de points désorganisé sans information supplémentaire de normales par exemple et sans spécifications topologiques. Plus récemment, [CL96] ont fourni un algorithme robuste et rapide dans le cas de données laser.

N. Amenta [ABK98] complète la formulation génrale du problème faite par H. Hoppes tentant de donner des définitions formelles des notions intuitives de "bon échantillonage" et de "reconstruction correcte" en établissant un lien formel entre ces deux qualités qualifiant l'entrée et la sortie du processus. Son algorithme utilise le diagramme de Voronoï 3D et la triangulation de Delaunay associée. Il produit un ensemble de triangles appelé le "crust" du nuage de points. Tous les sommets de ces triangles sont des points du nuage et la représentation finale apparaît comme un sous-graphe de la triangulation de Delaunay. En 2D, l'algorithme consiste à calculer le diagramme de Voronoï de l'ensemble de points S. Soit V l'ensemble des sommets de ce diagramme. On calcule alors la triangulation de Delaunay de l'ensemble $S \cup V$. Le "crust" est constitué de toutes les arêtes du graphe de Delaunay reliant deux points de S. Le passage en trois dimensions de cet algorithme bien formulé en 2D pose cependant encore des problèmes notamment au niveau des parties à forte courbure et en cas de bruit dans les données.

Ces techniques sont à rapprocher des tentatives de description ou de formalisation de la notion de forme à partir de la triangulation de Delaunay d'un nuage de points, que ce soit par la technique des α-formes [EK83] ou bien des techniques sculpturales [Boi84]. Nous y reviendrons dans le chapitre 5 consacré aux α-opérateurs morphologiques .

3.6 Conclusion

On s'aperçoit que la construction entièrement automatique d'un modèle d'un objet ou d'une scène complexe est encore impossible en l'état. Si l'on désire être générique, il faut être capable de subdiviser le travail de représentation au niveau de sous-ensembles significatifs du nuage de points. C'est ce que nous appelons l'étape de segmentation. Raphaelle Chaine étudie toutes les méthodes de segmentation utilisées en analyse d'image 2D en niveau d'intensité pour en appliquer une à des ensembles de points 3D désorganisés, mais ne retient que les techniques de relaxation de type modélisation par champ de Markov. Quant à nous, en faisant l'économie de cette analyse, dont les techniques obligent de toute façon à définir de façon relativement abstraite un système de voisinage *a priori* sur les points, nous allons nous attacher directement à étudier le nuage de points 3D du point de vue de l'Analyse de Données.

Chapitre 4

Regroupement

4.1 Problématique du regroupement

Une activité essentielle dans l'analyse de phénomènes, de signaux physiques et de données en général consiste à rassembler les entités étudiées en groupes cohérents, signifiants et relativement indépendants en l'absence de tout apprentissage supervisé. Cette activité est le plus souvent désignée dans la communauté scientifique par le terme anglais de "clustering" , et que nous désignerons dorénavant sous le terme d'activité de regroupement. Le problème se pose par exemple en marketing. Comment regrouper au mieux des consommateurs pour révéler des liens entre leurs différentes habitudes de consommation et éventuellement adapter la politique de l'entreprise pour un produit en particulier (cas du pack de bières et des couches pour bébé : on regroupe les consommateurs par genre féminin ou masculin et on extrait dans la population masculine un lien entre les deux achats) ? En fait, il s'agit de "révéler" une organisation des entités étudiées en groupes judicieux pour permettre de découvrir des similarités ou des différences parmi ces entités et d'en tirer des conclusions utiles. A l'inverse, dans l'acceptation habituelle du terme de regroupement utilisé en Reconnaissance de Formes, il s'agira plutôt de mettre en relation des entités selon une mesure de similarité pour faire émerger une organisation des entités étudiées en groupes judicieux pour l'application envisagée.

Parallèlement, la façon dont on prendra la décision de mettre en relation les différentes entités modifiera le résultat du regroupement. On perçoit donc comment différents facteurs peuvent influencer le résultat final du "clustering" d'un

même ensemble de données. Par ailleurs, on évoquera le concept de regroupement sous diverses appellations : apprentissage non supervisé ou sans professeur en Reconnaissance de Formes, taxonomie numérique en Biologie et Ecologie, typologie en Sciences Sociales et partition dans la Théorie des Graphes. On voit donc que derrière ce concept bien simple de regroupement se cache une diversité de méthodes, de formulations et bref de résultats pour un ensemble d'entités donné. Apparaît ici le caractère subjectif inhérent aux méthodes non supervisées : il n'est pas toujours évident de savoir quel regroupement de données est le plus judicieux. Dans un contexte scientifique, il s'agira donc également de s'affranchir de cette subjectivité pour arriver à des résultats stables, robustes et cohérents avec notre application.

Techniquement, beaucoup d'approches ont été développées pour résoudre les problèmes soulevés par ce besoin de regroupement. Les méthodes statistiques génériques sont certainement les plus nombreuses. Pourtant, des solutions plus "intelligentes" existent, conçues pour résoudre des problèmes spécifiques. Plaçons-nous du point de vue plus global des techniques exploratoires multidimensionnelles qui ont pour vocation d'explorer des tableaux de données le plus souvent issues d'observations statistiques. Deux grandes familles de méthodes correspondent à cette recherche :

- les méthodes factorielles, fondées sur des recherches d'axes principaux (l'analyse en composante principale ou ACP par exemple) dans un *cadre mathématique strict* ;
- les méthodes de classification qui produisent des groupements en classes d'objets (ou en familles de classes hirarchisées), obtenus à la suite de *calculs algorithmiques*.

Conçue pour la première fois par Karl Pearson en 1901, intégrée à la statistique mathématique par Harold Hotelling en 1933, l'Analyse en Composantes Principales n'est vraiment utilisée que depuis l'avènement et la diffusion des moyens de calculs actuels. Formellement, il s'agit de la recherche des axes principaux de l'ellipsoïde indicateur d'une distribution normale multidimensionnelle, ces axes étant estimés à partir d'un échantillon. Pratiquement, il s'agit d'une technique de représentation des données, ayant un caractère optimal selon certains critères algébriques et géométriques et que l'on utilise en général sans référence à des hypothèses de nature statistique ni à un modèle en particulier. Ainsi, une façon

simple de rendre compte de la forme d'un nuage est de le projeter sur des droites, ou sur des plans, en minimisant les déformations que la projection implique. Pour cela, on peut chercher le sous-espace à une dimension H qui maximise la somme des carrés des distances entre les projections sur H de tous les couples de points (k, k') :

$$Max_{(H)}[\sum_{k}\sum_{k'} d^2(k, k')] \tag{4.1}$$

Dans le cadre des méthodes factorielles, l'Analyse en Composante Principale s'applique particulièrement bien aux tableaux de données dans lesquels les p colonnes figurent les variables à valeurs numériques continues et les n lignes représentent les individus ou observations sur lesquels ces variables sont mesurées. Quelquefois, ces lignes pourront être considérées comme des réalisations indépendantes de vecteurs aléatoires, dont les composantes correspondent aux différentes variables. Cette représentation permet de visualiser les proximités entre les individus d'une part et entre les variables d'autre part. C'est pourquoi on visualisera chaque fois deux nuages. Le nuage qui nous intéressera particulièrement pour notre application dans l'espace euclidien réel à trois dimensions sera le nuage des individus dans l'espace des variables, dans lequel nous voudrons ajuster le nuage de n points par un sous-espace à une, puis deux dimensions, de façon à obtenir sur un graphique une représentation visuelle la plus fidèle possible des proximités existant entre les n individus vis-à-vis des p variables. Cet objectif revient à diagonaliser la matrice de corrélation des n individus et à projeter les données sur l'hyperplan porté par les vecteurs propres corrrespondant aux plus grandes valeurs propres appelés axes factoriels. Les coordonnées des données projetées dans ce nouveau repère sont appelées composantes principales ou coordonnées factorielles. En résumé, la similarité entre individus s'interprète en termes de similitudes de comportement vis-à-vis des variables en étudiant la forme des nuages et les proximités entre variables en termes de corrélations en étudiant la position par rapport à l'origine.

Dans notre problématique, nous nous intéresserons plus particulièrement à l'étude de la forme des nuages de points.

De façon concomitante, la classification est une branche de l'analyse de données prolifique dans de nombreux domaines d'application. Contrairement à la famille des méthodes factorielles, on ne se satisfait pas ici d'une visualisation plane

et continue des associations statistiques mais on désire mettre en évidence des classes d'individus. Les représentations synthétiques se manifestent alors soit sous la forme de partitions des ensembles étudiés, soit sous la forme de hiérarchie de partitions, quelquefois il pourra s'agir d'arbres au sens de la théorie des graphes. Comme nous l'avons déjà évoqué, à une même famille de résulats correspond parfois des démarches et des interprétations différentes. Il peut s'agir de découvrir une partition ayant une existence réelle (conjecturée ou révélée) ou au contraire d'utiliser les partitions produites comme des outils ou des intermédiaires de calculs permettant une exploration des données. Dans tous les cas, les techniques de classification font appel *à une démarche algorithmique* et non aux calculs formels usuels. L'une de ces démarches algorithmiques les plus performantes expérimentalement pour traiter un très grand nombre de données et faisant pourtant appel à un formalisme limité consiste à effectuer une *classification autour de centres mobiles* dont on pourra attribuer la paternité tantôt à Forgy en 1965, tantôt à Ball et Hall en 1967, et plus généralement aux travaux de généralisation de Diday en 1971 connus sous le nom de méthode des *nuées dynamiques* . L'idée essentielle étant de trouver une partition d'un ensemble de points minimisant la variance intraclasses et maximisant la variance interclasses.

De façon générale, pour faire du regroupement, il faut considérer quatre étapes :

- description des entités, consistant à construire une abstraction d'un monde réel dans lequel les entités à regrouper sont décrites selon certains schémas ;
- couplage des entités, consistant à définir dans quelles conditions nous considérons qu'une paire d'entités devrait être regroupée pour former un ensemble cohérent ;
- application d'un algorithme de regroupement à la description abstraite de ces entités ;
- interprétation des résultats.

Il faut noter que ces algorithmes ne découvrent pas de structures cachées ou inconnues dans un "système" mais imposent plutôt une structure sur l'ensemble d'entités qu'on lui fournit. Ils décident plus ou moins arbitrairement d'ignorer des liens et d'en favoriser d'autres. Cette décision se fonde sur les mesures de similarité utilisées et sur l'algorithme lui-même. Les structures ainsi imposées par les différents algorithmes ont des qualités différentes et par conséquent un intérêt différent en fonction de l'application envisagée. Pouvons-nous affirmer dans ces

conditions que tous les algorithmes et mesures de similarité donnent de façon générale des structures signifiantes ? Sans essayer de répondre pour l'instant à cette question, la plupart des auteurs estiment que le choix d'une mesure de similarité adéquate est plus importante pour l'exploitation du résultat que le choix de l'algorithme de regroupement. Pour finir, les algorithmes de regroupement doivent s'affranchir de trois écueils principaux :

- (a) initialisation ;
- (b) difficulté de détermination du nombre de groupements ;
- (c) sensibilité au bruit et aux données aberrantes.

4.2 Mesure de proximité

Nous considérons dans la suite le cas d'éléments à valeurs réelles continues.

4.2.1 Fonction de dissimilarité entre deux éléments

Une question intéressante, notamment pour des considérations théoriques sur les propriétés des algorithmes de regroupement, concerne la définition mathématique de la notion de mesure de proximité. Dans cette partie, on considère que nos données sont représentées par des vecteurs de caractéristiques dans $\Re^p$.

Par définition, une mesure de dissimilarité entre deux vecteurs est une application $d : \Re^p \times \Re^p \rightarrow \Re$ vérifiant :

$$\exists d_o \in \Re : -\infty < d_0 \leq d(x, y) < +\infty \; \forall x, y \in \Re^p; \tag{4.2}$$

$$d(x, x) = d_0 \; \forall x \in \Re^p; \tag{4.3}$$

$$d(x, y) = d(y, x) \; \forall x, y \in \Re^p; \tag{4.4}$$

La mesure d devient une métrique de mesure de dissimilarité si :

$$d(x, y) = d_0 \iff x = y; \tag{4.5}$$

$$d(x, z) \leq d(x, y) + d(y, z) \forall x, y, z \in \Re^p; \tag{4.6}$$

Parfois dans la littérature, on appellera d une distance au lieu de mesure de dissimilarité. Cela constitue un abus de langage d'un point de vue mathématique.

Par exemple, la distance euclidienne est une mesure de dissimilarité avec d_0=0 autant qu'une métrique de mesure de dissimilarité.

Dans le cadre flou que nous évoquerons plus loin, soient deux vecteurs $\mathbf{x}, \mathbf{y}$ dont les composantes appartiennent à l'intervalle $[\mathbf{0}, \mathbf{1}]$ dans $\Re^p$. Plus la composante $\mathbf{x_i}$ est proche de $\mathbf{1}$ plus $\mathbf{x}$ possède vraisemblablement le caractère $\mathbf{i}$. Plus $\mathbf{x_i}$ s'approche de $\mathbf{1/2}$ moins on devient certain du rapport entre $\mathbf{x}$ et le $\mathbf{i^{me}}$ caractère. Cette modélisation est une généralsiation de la logique binaire quand $\mathbf{x_i}$ ne peut prendre que la valeur $\mathbf{0}$ ou $\mathbf{1}$ ($\mathbf{x}$ possède le caractère $\mathbf{i}$ ou non), l'idée de la logique floue étant que rien n'arrive ou n'arrive pas avec une absolue certitude. Dans cette voie, le degré de similarité entre deux valeurs réelles $\mathbf{x_i}$ et $\mathbf{y_i}$ dans $[\mathbf{0}, \mathbf{1}]$ est défini par :

$$s(x_i, y_i) = max(min(1 - x_i, 1 - y_i), min(x_i, y_i)) \tag{4.7}$$

et conséquemment une mesure de similarité entre deux vecteurs $\mathbf{x}$ et $\mathbf{y}$ est définie par :

$$s_F^q(x, y) = \left(\sum_{i=1}^{p} s(x_i, y_i)^q \right)^{1/q} \tag{4.8}$$

où $\mathbf{q} \geq \mathbf{1}$ désigne le degré de flou.

4.2.2 Fonction de dissimilarité entre un élément et un ensemble

Dans de nombreux algorithmes de regroupement, on affecte un vecteur x à un groupement C en prenant en compte la proximité $\wp(x, C)$ entre le point x et le groupement C. Il existe alors deux directions générales pour la définition de $\wp(x, C)$:

1. Tous les points de C contribuent à $\wp(x, C)$. On a alors par exemple :
   - $\wp_{max}(x, C) = max_{y \in C}\{\wp(x, y)\}$;
   - $\wp_{min}(x, C) = min_{y \in C}\{\wp(x, y)\}$;
   - $\wp_{moy}(x, C) = \frac{1}{card(C)} \sum_{y \in C} \wp(x, y)$;

   avec $\wp(x, y)$ une mesure de proximité quelconque entre deux points.
2. C est représenté par un prototype et on mesure la proximité entre x et le représentant de C
   - C est représenté par un point si le groupement est compact ;

- C est représenté par un hyperplan si le groupement est de forme linéaire ;
- C est représenté par une hypersphère si le groupement est de forme sphérique.

4.2.3 Fonction de dissimilarité entre deux ensembles

Quelques uns des algorithmes de regroupement utilisent l'information de proximité entre ensembles $\mathbf{C_i}$ et $\mathbf{C_j}$. Construite à partir d'une mesure de proximité $\wp$ entre deux éléments, on trouvra :

- $\wp_{max}(C_i, C_j) = max_{x \in C_i, y \in C_j}\{\wp(x, y)\}$;
- $\wp_{min}(D_i, D_j) = min_{x \in C_i, y \in C_j}\{\wp(x, y)\}$;
- $\wp_{moy}(C_i, C_j) = \frac{1}{card(C_i)card(C_j)} \sum_{x \in C_i} \sum_{y \in C_j} \wp(x, y)$.

avec $\wp(x, y)$ une mesure de proximité quelconque entre deux points.

Intuitivement, on peut comprendre que différents choix de fonctions de proximité entre deux ensembles peuvent conduire à des resultats de regroupement totalement différents. De plus, si on utilise des mesures de proximité différentes simultanément, la même fonction de proximité entre deux ensembles conduira également en général à des résultats de regroupement différents.

On voit là encore toute la difficulté de définir un algorithme de regroupement. Pour reprendre l'analyse de Theodoris et Koutroumbas, "la seule façon de concevoir un algorithme adéquate de regroupement de données se fait par essais et erreurs et, bien sûr, en prenant en compte l'opinion d'un expert du domaine d'application.".

4.3 Algorithmes

Pour analyser un ensemble de N éléments, il faudrait être capable de réaliser toutes les partitions possibles en m regroupements pour $1 \leq m \leq N$ et retenir la meilleure d'entre elles selon un critère de qualité de partitionnement propre à une application spécifique. Or, le nombre de regroupements possibles d'un ensemble de N éléments en m groupements suit la loi de Stirling et vaut :

$$S(N, m) = \frac{1}{m!} \sum_{i=0}^{m} (-1)^{m-i} \binom{m}{i} i^N$$

Ainsi, le nombre de regroupements possibles pour trouver cinq groupements dans un ensemble de cent éléments est $S(100, 5) = 10^{68}$. Ce qui n'est raisonnablement

pas accessible avec les moyens actuels de calcul. Il faut donc trouver une parade à cette limitation technologique voire théorique.

En ce sens, la définition d'un algorithme de regroupement correspondra à la définition de schémas algorithmiques fournissant les regroupements judicieux ne considérant qu'une petite fraction de l'ensemble contenant toutes les partitions possibles de X, de façon plus ou moins heuristique. C'est dans cette perspective qu'un algorithme de regroupement pourrait être qualifié de procédure d'apprentissage - au sens des réseaux neuronaux informatiques - qui essaierait d'identifier les caractéristiques spécifiques des groupements sous-jacents à l'ensemble de données. N'oublions pas enfin que des contraintes devront être ajoutées (par exemple sur le nombre de groupements maximum à détecter) toujours en raison des limitations en ressources.

De façon pratique, il existe trois grandes catégories d'algorithmes de regroupement :

- séquentiels ;
- hiérarchiques ;
- itératifs.

4.3.1 Algorithmes séquentiels

Initiés en 1967 par D. Hall et G. Ball [BH67], ces algorithmes créent au fur et à mesure des groupements nouveaux. On présente un à un chaque point et en fonction d'un seuil Θ on l'affecte à un groupement déjà existant ou bien on crée un nouveau groupement. Cette méthode de regroupement est à mettre en relation avec l'architecture neuronale de type ART2 développée dans le cadre de la théorie de la Résonance Adaptative.

4.3.2 Algorithmes hiérarchiques

Les algorithmes hiérarchiques quant à eux produisent une hiérarchie de regroupements et non pas un seul regroupement commes les algorithmes séquentiels. Ils sont en conséquence surtout utilisés dans les domaines des Sciences de la Vie pour déterminer des taxonomies en Biologie par exemple. On visualise généralement la séquence de regroupements produit par un algorithme hiérarchique de type agglomératif à l'aide d'un dendrogramme. Puis en coupant le dendrogramme

à un niveau spécifique on obtient un regroupement particulier à un niveau de détail donné. On rappellera les trois modes algorithmiques majeurs employés :

- par la théorie des matrices ;
- par la théorie des graphes ;
- par les arbres de recouvrement minimal.

Les problèmes précédents d'initialisation et de sensibilité au bruit ne se posent pas aux algorithmes de type hiérarchique. Le problème de détermination du nombre de groupements optimum peut se faire par l'analyse de la "durée de vie" d'un groupement dans la construction de la hiérarchie de regroupement. Mais toutes ces méthodes demeurent fondées sur l'utilisation d'heuristiques. Par contre, ces algorithmes présentent deux autres inconvénients majeurs :

- ils sont locaux donc on ne peut pas incorporer de connaissance *a priori* sur la forme globale ou la taille des groupements ;
- ils sont statiques ne permettant pas de faire passer une entité d'un groupement à l'autre une fois affectée.

4.3.3 Algorithmes itératifs

Initiés par R. Duda et P. Hart en 1973 [DH73] puis repris principalement par J. Bezdek à partir des années 80 [BCGW81], ces schémas sont fondés sur l'utilisation de techniques de calcul différentiel. Le nombre de groupements C est connu. Ils reposent sur la minimisation d'une fonction J, fonction de l'ensemble X à partionner et d'un vecteur de paramètres inconnus Θ, dépendant fortement de la forme des groupements à détecter. Pour des groupements compacts, $\Theta = [m_1^T, m_2^T, ..., m_C^T]^T$ où m_i est un point de $\Re^p$, généralement le barycentre du nuage de points correspondant au groupement i. Pour des groupements en forme d'anneau, $\Theta = [c_1^T, r_1, c_2^T, r_2, ..., c_C^T, r_C]^T$ où c_i est le centre du cercle correspondant au prototype représentant le groupement i et r_i son rayon.

On répertorie trois grandes approches de minimisation :

1. bayésienne par décomposition de mélanges ;
2. floue ;
3. possibiliste ;

Pour les méthodes bayésiennes, on doit faire face au besoin classique d'hypothèses assez fortes sur la distribution des groupements, le plus traditionnellement

suivant une loi de Gauss pour des groupements compacts. Par ailleurs, elles sont très coûteuse en temps de calcul.

Jean-Paul Benzécri écrivait en 1965 dans son cours à la Sorbonne sur “l’Analyse des données et la reconnaissance des formes” : “Il est impensable d’utiliser des méthodes conçues avant l’avènement de l’ordinateur, il faut complètement réécrire la statistique” tout en préconisant que “le modèle doit suivre les données et non l’inverse”. Or à défaut d’être repensée, la statistique s’est certes considérablement enrichie mais demeure encore trop liée à des hypothèse fortes de modèles de distributions de données.

En revanche, les algorithmes itératifs empruntant la voie d’une modélisation floue sont dynamiques, autorisant une compétition entre les groupements. En effet, on peut incorporer de la connaissance de façon assez souple sur la forme et la taille des groupements en utilisant des prototypes et des mesures de distance appropriés aux applications envisagées. Malheureusement, ces algorithmes utilisent des techniques d’optimisation dont la nature itérative les rendent sensibles à l’initialisation et partant, à l’attraction vers les minima locaux. En résumé, ils sont sensibles aux trois types de problèmes rencontrés quand on désire faire du regroupement.

Dans cette approche, contrairement au regroupement dur, un vecteur peut appartenir simultanément à plus d’un groupement, c’est-à-dire qu’on définit un coefficient d’appartenance u_{ij} à valeurs dans l’intervalle $[0, 1]$ traduisant le degré d’appartenance du vecteur i au groupement j selon la théorie des ensembles flous tandis que dans le cadre d’une modélisation classique dite dure le coefficient u_{ij} prend ses valeurs dans l’ensemble $\{0; 1\}$. La forme de la fonction à optimiser s’écrit de façon générale :

$$J_q(\Theta, U) = \sum_{i=1}^{N} \sum_{j=1}^{m} u_{ij}^q d(x_i, \Theta_j) \tag{4.9}$$

sous la contrainte

$$0 < \sum_{j=1}^{m} u_{ij} = 1 \ \ \forall i; \tag{4.10}$$

où Θ est le vecteur de paramètres représentant les groupements et U désigne la matrice des coefficients u_{ij} d’appartenance de l’élément i au groupement j.

Théoriquement, si $q = 1$, il n’y a pas de regroupement flou meilleur que le meilleur des regroupements durs (au sens de J). Par contre si $q > 1$, il existe des

cas pour lesquels le regroupement flou mène à de plus petites valeurs de $J_q(\Theta, U)$ que le meilleur des regroupements durs.

On rencontrera trois grands cas :

1. Dans le cas de groupements compacts, on représentera un groupement par un point. La mesure de dissimilarité peut être toute distance entre deux points. Par exemple,
   - $d(x_i, \Theta_j) = (x_i - \Theta_j)^T A(x_i - \Theta_j)$ où A est une matrice définie positive et symétrique. Dans ce cas, on a affaire à l'algorithme des C-moyennes floues à proprement dit qui se résoud linéairement.
   - la distance de Minkowski $d(x_i, \Theta_j) = (\sum_{k=1}^{p} |x_{ik} - \Theta_{jk}|^q)^{\frac{1}{q}}$. On a alors affaire à l'algorithme des q C-moyennes floues qui se résoud par des techniques plus complexes itératives de Gauss-Newton ou Levenberg-Marquardt.
2. On représente un groupement par une surface quadrique le plus souvent ellipsoïdale. Dans ce cas, il faut tout d'abord définir la distance entre un point et une quadrique : à savoir algébrique, perpendiculaire ou radiale.
3. On assimile les groupements à des hyperplans. Soit on considère la distance d'un point à un plan dans $\Re^3$ par exemple soit on utilise la distance de Mahalanobis normalisée :

$$d^2(x, \Theta_j) = |\sum_j|^{\frac{1}{l}} (x - c_j)^T \sum_j^{-1} (x - c_j) \tag{4.11}$$

   L'un des algorithmes classiquement utilisé est alors celui de Gustafson-Kessel [Gus79].

Enfin, l'approche possibiliste relâche une contrainte en demandant seulement que :

$$0 < \sum_{i=1}^{N} u_{ij} \leq N \quad \forall j; \tag{4.12}$$

$$max_{j=1\ldots C}(u_{ij}) > 0 \quad \forall i. \tag{4.13}$$

Ici, u_{ij} peut-être interprété comme le degré de compatibilité de x_i avec le représentant du j^{me} groupement, ou encore la possibilité que x_i appartienne au j^{me} groupement, qui est indépendante des possibilités de x_i d'appartenir à un autre groupement.

4.4 La théorie du flou

Les ensembles flous sont nés des travaux de Lotfi Zadeh en 1965 [Zad65] comme un moyen de traiter de l'imprécision et de l'incertitude dans la vie quotidienne. On doit les voir comme un moyen d'étendre la théorie traditionnelle des ensembles, pour permettre une interprétation floue des structures de données et offrir une formulation naturelle et intuitivement plausible de divers problèmes traités par la Reconnaissance des Formes notamment. Lorsque l'imprécision en question est d'ordre non statistique, comme dans l'expression "Georges doit avoir autour de trente ans", on parlera d'une incertitude de type flou. A partir de là, on comprendra que l'un des premiers intérêts de cette nouvelle théorie s'exprime dans le domaine du maniement linguistique d'informations, et notamment du contrôle de processus industriels. Dans la théorie des ensembles classiques, un élément e appartient ou n'appartient pas à un ensemble F. En 1965, Lotfi Zadeh propose de représenter cet ensemble F au moyen d'une fonction d'appartenance m_F prenant ses valeurs dans l'intervalle [0,1]. La valeur $m_F(e)$ est appelée coefficient d'appartenance de e à F. Ce faisant un élément peut appartenir avec plus ou moins de cohésion à un ensemble ou à plusieurs ensembles. En ce sens, les fonctions d'appartenance donnent une certaine élasticité à la description des faits. En revanche, une des difficultés majeures de cette formulation réside dans la construction de ces fonctions d'appartenance, qui n'ont rien de statistique comme en théorie des probabilités, mais qui demeurent le plus souvent subjectives.

4.4.1 Son utilisation en Reconnaissance des Formes

Nous pourrions définir de façon assez consensuelle la discipline de Reconnaissance des Formes comme la recherche de structures dans des données et la catégoriser par conséquent parmi les sciences inexactes ou expérimentales. C'est la raison pour laquelle cette discipline propose plusieurs approches parfois complémentaires parfois compétitives pour approcher la solution à un problème. Au sens où J-C Simon pouvait l'expliquer [Sim84], on a affaire à une discipline constructiviste qui définit ses méthodes en fonction des résultats concrets et expérimentaux qu'elle fournit. En clair, la Reconnaissance des Formes s'occupe de décrire les phénomènes physiques en adaptant sa modélisation en fonction de l'application envisagée, à l'image des différents modèles utilisés dans les Sciences Physiques

pour décrire une même réalité physique.

La nécessité de prise en compte de la réalité floue des éléments à traiter en Reconnaissance des Formes se fait ressentir dans les deux domaines essentiels de la Reconnaissance des Formes : l'analyse de primitives et l'analyse des regroupements. La nature floue des primitives permettant de décrire les éléments à reconnaître se retrouve dans l'imprécision caractérisant toute description d'un phénomène par un être humain. De même, quand on parvient à trouver des regroupements parmi les éléments étudiés, les frontières entre ces regroupements ne sont jamais nettes et très souvent deux regroupements peuvent se superposer pour une partie de leurs éléments. Enfin, donner une décision finale sur l'appartenance d'un élément à une classe ne se conçoit pas sans entâcher cette décision d'une mesure de confiance, voire de possibilité ou de nécessité. Les Systèmes Experts ont su en leur temps exploiter cette grande flexibilité offerte par la théorie des Ensembles Flous, relayée par la théorie des Possibilités et de l'Evidence.

Parallèlement, en Vision par Ordinateur, cette nécessité de prendre en compte l'imprécision dans les systèmes a donné lieu aux développements de nombreux algorithmes spécifiques de traitement d'images. Par exemple, si l'on considère le problème de l'extraction d'objets dans une scène, la question est " Comment peut-on définir exactement la région de l'objet dans une scène quand ses frontières sont mal définies ?". En l'occurence, tout seuillage direct réalisé pour extraire l'objet propagera l'incertitude associée aux étapes de traitement ultérieures et ceci pourrait affecter à leur tour les étapes d'analyse de primitives et de reconnaissance.

4.4.2 Le regroupement flou

La segmentation d'images se propose de regrouper des pixels pour former des régions correspondant à des objets siginificatifs de la scène. En ce sens, c'est un algorithme de regoupement utilisant une mesure de proximité entre pixels. En imagerie médicale essentiellement, où les structures que l'on cherche à segmenter ont des frontières mouvantes et pas toujours très marquées sur les images disponibles, on désire faire des mesures de volume par exemple qui doivent s'affranchir de ce problème d'incertitude et d'imprécision sur les contour des régions à mesurer. On a développé des techniques de segmentation floue [HB93] à partir de la théorie des ensembles flous propres à gérer ce genre de situation. On voit donc que dans de nombreux domaines la prise en compte d'une incertitude d'ordre non

statistique s'impose et permet de négocier des situations ingérables dans un cadre plus classique.

L'apport du flou dans les problèmes de regroupement de données en général est du même ordre. En insufflant plus d'élasticité dans les modèles employés, on atteint des résultats jugés le plus souvent de meilleure qualité et beaucoup plus réalistes, notamment parmi les techniques de segmentation d'images [BHC93] [US96] [PPDX97] [PP99] [XPP97] [BBKF94] [Les99]. Les algorithmes de regroupement flou en constituent une preuve expérimentale flagrante. Ils ont été abondamment utilisés pour détecter des lignes, des plans, des ellipses, des courbes et des surfaces [Bez81][Dav89][DB92][KFN95].

4.5 Les méthodes itératives de regroupement flou

Parmi les algorithmes les plus abondamment développés pour obtenir une classification directe d'un ensemble de données figurent les algorithmes de regroupement de type ISODATA, acronyme pour *"iterative, self-organizing data analysis techniques A"*. Ainsi, historiquement, ce domaine de la Reconnaissance Statistique des Formes se trouve intimement lié à une des branches les plus prolifiques de la modélisation informatique neuronale appelée le regroupement par cartes auto-organisatrices de Kohonen, même si les connexions mathématiques entre ces deux techniques d'inspiration commune restent encore à établir. Dans cette optique commune, le problème de regroupement est également souvent évoqué sous l'appellation d'*apprentissage non supervisé*, le terme d'*apprentissage* faisant référence ici à l'apprentissage des labels corrects pour les "bons" sous-groupes.

Sur le modèle développé initialement par Ball[BH67], ces algorithmes itératifs, dans la lignée des algorithmes de regroupement en général, voient leurs performances influencées à des degrés divers par le choix du nombre de groupements C, des prototypes initiaux, de la mesure de proximité utilisée, et par les propriétés géométriques des données. En pratique, les performances de tels instruments de regroupement dépendent en grande partie de "l'intelligence" et de l'expertise de son concepteur ou des utilisateurs à l'instar d'ailleurs de nombreuses techniques utilisées dans les domaines de la Reconnaissance des Formes et du Traitement d'Images.

4.5.1 Fondements historiques et algorithmiques

La plupart des modèles flous de regroupement et de classification couramment utilisés repose sur l'idée développée par Ruspini en 1969 [Rus69]. Il est le premier à introduire la notion de C-partitions floues non dégénérées d'un ensemble X de n vecteurs caractéristiques de R^p. Soit c un entier tel que $1 < c < n$ et soit $X = \{x_1, x_2, ..., x_n\}$ un ensemble de n vecteurs caractéristiques de R^p. Nous dirons que c sous-ensembles flous $u_i : X \rightarrow [0, 1]$ constituent une c-partition floue de X si les $c \times n$ valeurs $\{u_{ik} = u_i(x_k), 1 \leq k \leq n, 1 \leq i \leq c\}$ satisfont les trois conditions suivantes :

$$0 \leq u_{ik} \leq 1 \ \ \forall i, k; \tag{4.14}$$

$$\sum u_{ik} = 1 \ \ \forall k; \tag{4.15}$$

$$0 < \sum u_{ik} < n \ \ \forall i. \tag{4.16}$$

Chaque ensemble de $c \times n$ valeurs satisfaisant les conditions précédentes peut être visualisé dans une matice $U_{c \times n} = [u_{ik}]$. L'ensemble de telles matrices est appelé l'ensemble des C-partitions floues non dégénérées de X :

$$M_{fcn} = \{U \ dans \ \Re^{cn} | u_{ik} \ satisfasse \ (4.14), (4.15), (4.16) \ \forall i, k\}.$$

La raison pour laquelle ces matrices sont appelées des partitions vient de l'interprétation des u_{ik} comme l'appartenance de x_k au i^{eme} sous-ensemble de partitionnement de X. Incidemment, tout algorithme de regroupement génère des solutions au problème de regroupement dans X qui sont des matrices dans M_{fcn}. En conséquence, regrouper dans X consiste dans cette formulation simplificatrice à identifier une partition "optimale" U de X dans M_{fcn} - c'est-à-dire une partition qui regroupe ensemble les vecteurs de données d'objets (et par conséquent les objets qu'ils représentent) partageant quelques similarités mathématiques bien définies. Cette approche suppose de façon intuitive qu'un regroupement mathématiquement optimal représentera une description pertinente des regroupements naturels du point de vue du processus physique dont dérivent les objets. Puis, Ruspini définit et analyse le premier algorithme utilisant une fonction d'objectivité floue pour générer les c-partitions floues d'un ensemble fini de données non labélisées. Dans cette définition, le nombre de groupes c était supposé connu. Dans le cas contraire, cette valeur devient un paramètre du problème de regroupement. Cette facette du

problème est souvent appelée la question de la *validité du regroupement*. La plupart des travaux à cette date reconnaît l'importance de la question et utilise des méthodes de nature heuristique pour la résoudre. Pour finir, force est de constater que de nombreux algorithmes statistiques de regroupement (comme par exemple l'apprentissage non supervisé par le maximum de vraisemblance) produisent habituellement des solutions dans M_{fcn}. On voit que Ruspini aura ainsi posé les bases simples et fertiles d'une très grande littérature sur la nature algébrique et géométrique de ce type de partitionnement. J.C. Dunn [Dun73] puis J. Bezdek [BCGW81] généralisent la fonction objective utilisée pour le regroupement à une famille infinie de fonctionnelles associées au problème des C-moyennes floues :

$$J_m(U, v; X) = \sum_i \sum_k u_{ik}^m (||x_k - v_i||_A)^2$$

où $m \in [1, \infty)$ est un poids en exposant sur chaque coefficient d'appartenance, U est une c-partition floue de X, $v = (v_1, v_2, ..., v_c)$ sont les centres des groupements dans R^p, A est une matrice définie positive $n \times n$, et $||x_k - v_i||_A = (x_k - v_i)^T A(x_k - v_i)$ est la distance selon la norme A de x_k à v_i. L'idée de base de l'algorithme des C-moyennes floues consiste à minimiser J_m en fonction de la paire de variables U et v, et selon l'hypothèse que les matrices U faisant partie d'une paire optimale pour J_m identifient de "bonnes" partitions des données. De nombreuses études ont été menées pour résoudre ce problème dont le plus populaire ensemble d'algorithmes est connu sous le nom d'algorithmes des C-moyennes floues ou encore ISODATA flous. Ces derniers sont fondés sur l'utilisation simple d'itérations de Picard sous les conditions nécessaires émergeant d'un Lagrangien du gradient de J_m s'annulant.

4.5.2 Convergence et terminaison

Quelques travaux s'intéressent aux problèmes de convergence numérique des algorithmes itératifs de regroupement de type ISODATA. La preuve (erronée finalement) de J. Bezdec [Bez80] utilisant le théorème de Zangwill montrait que chaque séquence itérée $\{U_t, v_t\}$ d'un algorithme ISODATA converge (ou possède une sous-séquence qui converge) vers un extremum local de J_m, à partir de n'importe quelle initialisation dans $M_{fcn} \times (\Re^p)^c$. W.T. Tucker [Tuc87] trouvera plus tard un contre-exemple à cette affirmation, et aboutira au nouveau résultat de convergence vers un minimum local ou un point de rebroussement de J_m. En

pratique, la terminaison des algorithmes de type ISODATA (ce qui est différent à la fois de la notion de convergence numérique - avec le nombre d'itérations - et de la notion de convergence asymptotique - avec le nombre d'éléments -) n'a jamais posé de problème. Toutefois, on peut affirmer que les résultats théoriques valables ne sont pas très nombreux.

4.5.3 Validité et interprétation des résultats

Les méthodes d'analyse de données produisent plus que des représentations. Elles dévoilent des traits structuraux et permettent d'éprouver la cohérence des données. Devant les résultats d'une classification, on est naturellement amené à se poser des questions sur la qualité de la représentation :

- Observe-t-on vraiment quelque chose ?
- Les données exhibent-elles réellement une structure ou existe-t-il des classes ?
- A-t-on découvert des classes préexistantes ?
- Est-ce que les configurations obtenues sont stables ?
- Quel nombre de classes retenir, le plus souvent en vue d'une utilisation particulière ?

Autant de questions simples amenant à des tentatives de solutions complexes. En effet, la méthodologie de validation est souvent pragmatique et orientée selon les applications.

D'une façon plus générale, dans le cadre des méthodes de classification en C classes C_k, on peut par exemple associer une densité $f_k(x)$. Dans le cas où les densités $f_k(x)$ sont celles de lois normales sphériques de même matrice de covariances $\sigma^2 I$ et de moyennes μ_k, la partition qui réalise le maximum de vraisemblance est celle qui minimise le critère :

$$cr(C) = \sum_{k=1}^{C} \sum_{i \in C_k} d^2(i, v_k) \tag{4.17}$$

où v_k est le centre de gravité de la classe C_k. On reconnaît le critère utilisé dans l'agrégation autour de centres mobiles. La partition optimale exacte est actuellement impossible à déterminer, mais la méthode des centres mobiles, on l'a vu, conduit rapidement à un optimum local. Ce critère permet donc, dans le cadre fourni par ce modèle, d'évaluer la qualité d'une partition.

Mais la réalité des critères de validité de partition est autrement plus variée et souvent empiriquement facilitée par l'usage de critères externes.

Parmi tous les indices de partitionnement définis pour juger de la qualité d'une partition, on trouve le *Coefficient de Partition* ou *CP*, défini par :

$$PC = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{m} u_{ij}^2 \tag{4.18}$$

où u_{ij} sont les valeurs obtenues après la convergence de l'algorithme de regroupement flou.

On trouve un autre index dans cette catégorie le *Coefficient d'Entropie de Partition* ou *EP*, défini par :

$$PE = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{m}(u_{ij} log_a u_{ij}) \tag{4.19}$$

On définit également un index dit de Xie-Beni appelé fonction de compacité ou de séparation, défini par :

$$XB = \frac{\sigma_2}{N * d_{min}} \tag{4.20}$$

où d_{min} est la distance minimum entre tous les centres de classes et σ_2 mesure la variation dans les classes par :

$$\sigma_2 = \sum_{j=1}^{m} \sigma_j^2 \tag{4.21}$$

avec

$$\sigma_j^2 = \sum_{i=1}^{N} u_{ij}^2 \|x_i - v_j\|^2 \tag{4.22}$$

Enfin, en définissant le matrice de covariance floue Σ_j par :

$$\Sigma_j = \frac{\sum_{i=1}^{N} u_{ij}^2 (x_i - v_j)(x_i - v_j)^T}{\sum_{i=1}^{N} u_{ij}^2} \tag{4.23}$$

on définit également le volume flou des groupements à partir du déterminant de cette matrice :

$$V_j = |\Sigma_j|^{1/2} \tag{4.24}$$

et le volume flou total :

$$VF = \sum_{j=1}^{m} V_j \tag{4.25}$$

A partir de toutes ces définitions, soit $X_j = \{x \in X : (x-v_j)^T \sum_j^{-1}(x-v_j) < 1\}$ qui contient tous les vecteurs à l'intérieur d'une boule de rayon 1 autour de v_j, soit $S_j = \sum_{x_i \in X_j} u_{ij}$ la somme des coefficients d'appartenance des éléments appartenant à cette région, alors la *Densité Moyenne de Partition* ou *DMP* est définie par :

$$DMP = \frac{1}{m} \sum_{j=1}^{m} \frac{S_j}{V_j} \tag{4.26}$$

Enfin, on pourra utiliser une autre mesure appelée *Densité de Partition* ou *DP* et définie par :

$$DP = \frac{S}{VF} \tag{4.27}$$

où $S = \sum_{j=1}^{m} S_j$.

En réalité, en fonction de la nature des nuages de points à traiter et de l'application envisagée, la stratégie de détermination du nombre optimal de classes n'est pas liée à une démarche méthodologique très rigoureuse jusqu'à présent. Il est à espérer qu'une stratégie puisse répondre à de nombreux problèmes de ce type dans des contextes variés.

4.5.4 Variantes et illustrations

Maintenant que le schéma algorithmique adopté est précisé, il est intéressant de visualiser les différents résultats expérimentaux en fonction du choix de la "distance" employée, du nombre de regroupement spécifié en entrée et bien sûr de la géométrie des données à regrouper. Le jeu de données testé est constitué de quatre ensembles de points dans $\Re^2$ comme illustré en figure 4.1.

4.5.4.1 Distance Euclidienne

Cette distance est particulièrement appropriée à la détection de groupements relativement compacts de topologie plutôt hyperellipsoïdale. En figure 4.2, nous illustrons le comportement de l'algorithme des C-Moyennes floues utilisant une simple distance euclidienne pour la détection de deux classes dans chacun des ensembles tests de la figure 4.1.

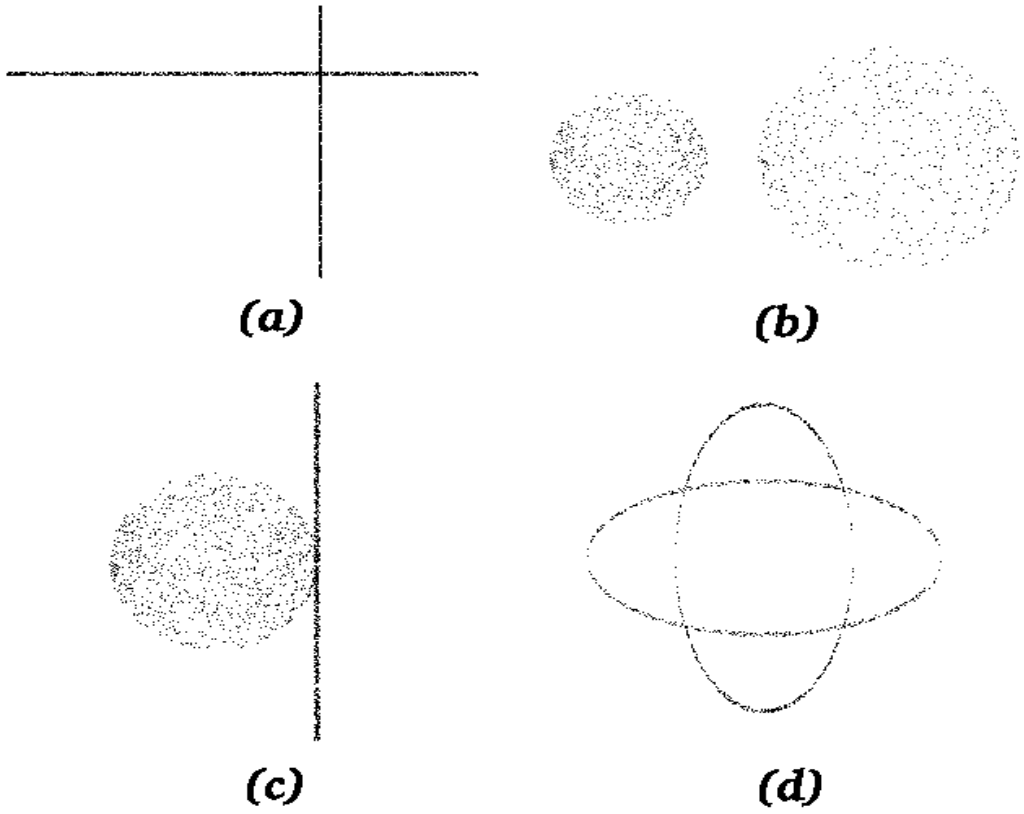

FIG. 4.1 – Jeu de données test (a) groupements de nature linéaire (b) groupements de type compacts mais de densités en points différentes d'un facteur d'environ triple (c) groupements de types mixtes (d) groupements de contours d'ellipses

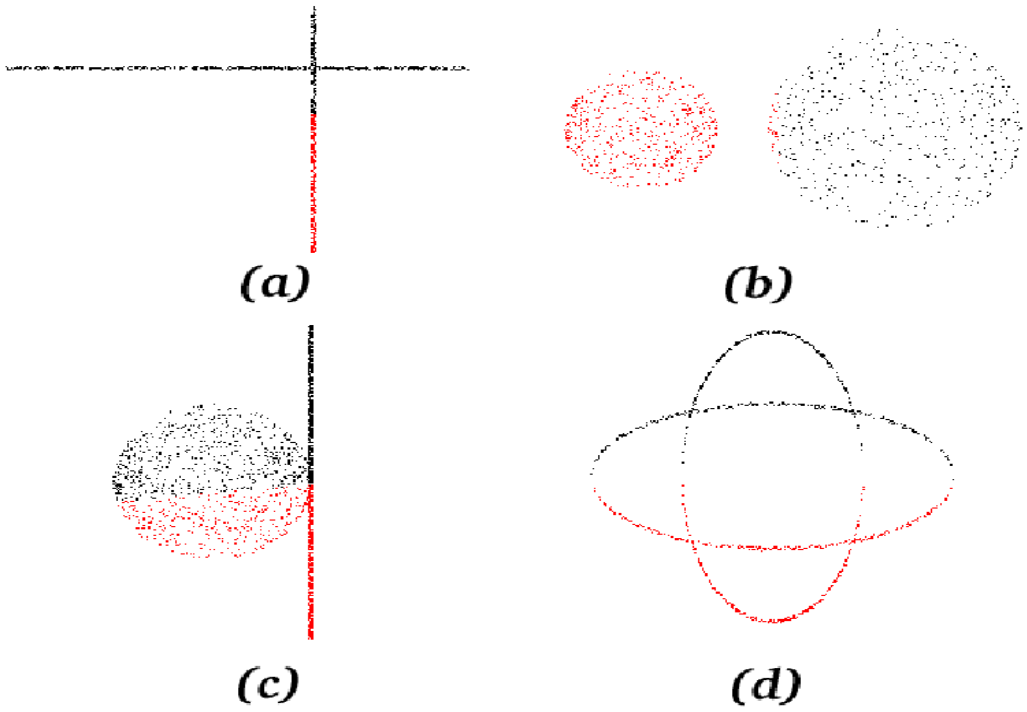

FIG. 4.2 – Regroupement en deux classes par la méthode des C-Moyennes floues utilisant la distance euclidienne sur les (a) groupements de nature linéaire (b) groupements de type compacts de densité en points différentes (c) groupements de types mixtes (d) groupements de contours d'ellipses

4.5.4.2 Distance de Mahalanobis

La distance de Mahalanobis normalisée est définie par la formule 4.28.

$$d(X,Y) = \sqrt{(det(A))XAY^t} \tag{4.28}$$

où A est une matrice symétrique réelle représentant la matrice de covariance du nuage de points.

Elle est censée permettre la prise en compte de la forme des groupements en fonction des directions privilégiées du nuage correspondant aux éléments propres de la matrice de corrélation du nuage. En figure 4.3, nous illustrons le comportement de l'algorithme des C-Moyennes floues utilisant une telle distance pour la détection de deux classes dans chacun des ensembles tests de la figure 4.1.

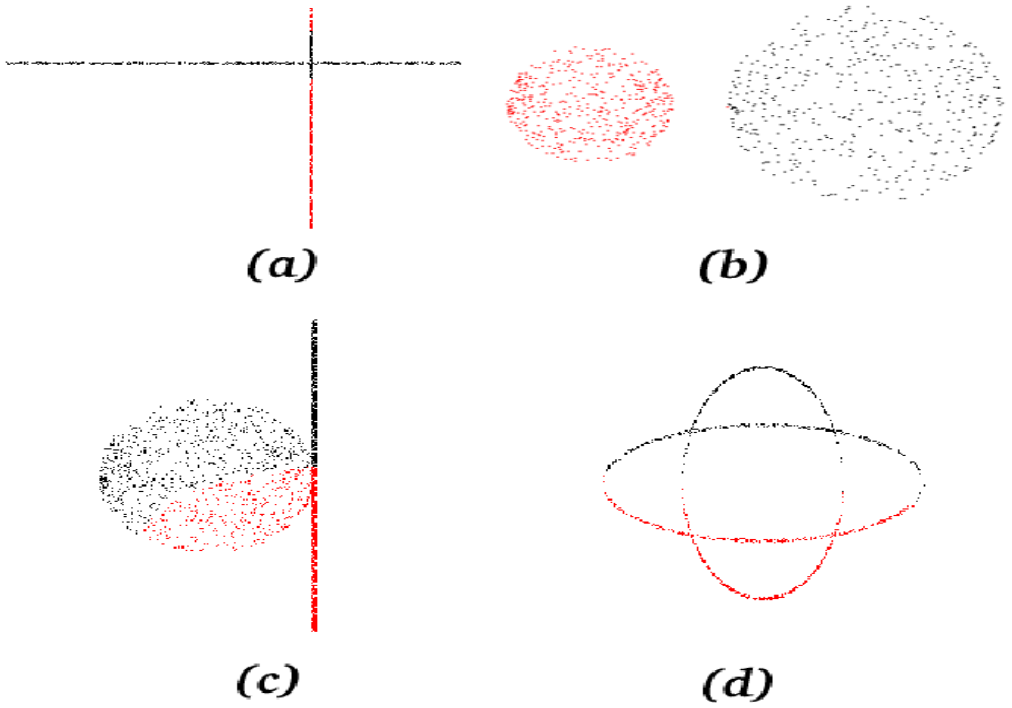

FIG. 4.3 – Regroupement en deux classes par la méthode des C-Moyennes floues utilisant la distance de Mahalanobis normalisée sur les (a) groupements de nature linéaire (b) groupements de type compacts de densité en points différentes (c) groupements de types mixtes (d) groupements de contours d'ellipses

4.5.4.3 Distance à une droite

Dans le cas de groupements linéaires, on peut faire le choix de représenter chaque classe par une droite d'approximation optimale en 2D. En figure 4.4, nous illustrons le comportement de l'algorithme des C-Moyennes floues utilisant une telle distance pour la détection de deux classes dans chacun des ensembles tests de la figure 4.1.

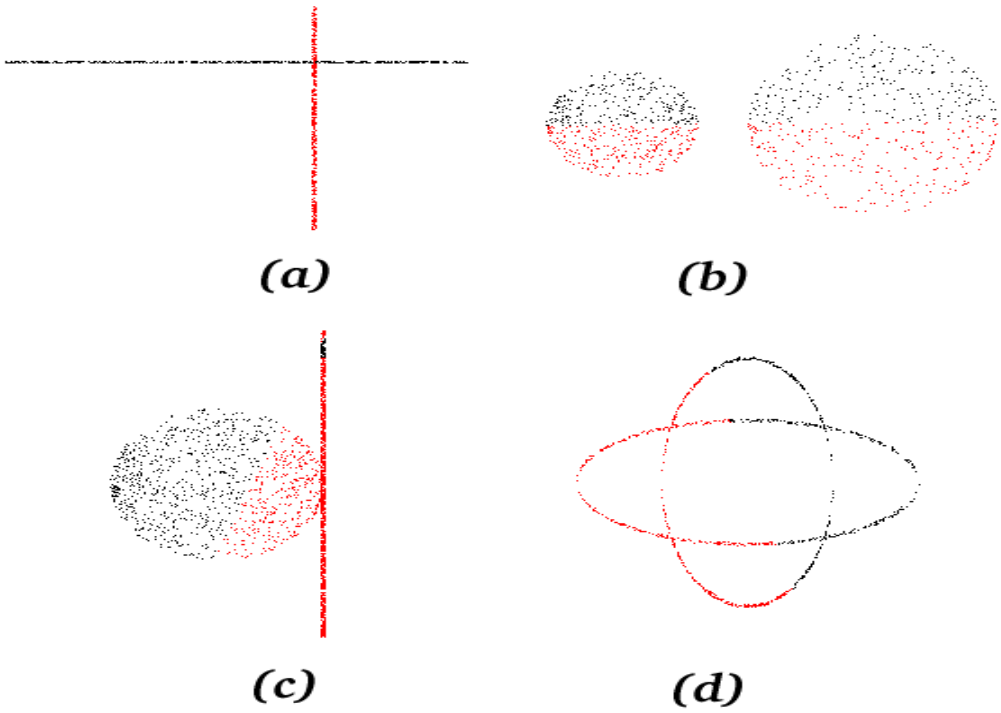

FIG. 4.4 – Regroupement en deux classes par la méthode des C-Moyennes floues utilisant la distance point - droite prototype sur les (a) groupements de nature linéaire (b) groupements de type compacts de densité en points différentes (c) groupements de types mixtes (d) groupements de contours d'ellipses

4.5.4.4 Distance "exponentielle"

La distance "exponentielle" est définie dans [GG89] par la formule suivante :

$$d_e^2(X_j, V_i) = \frac{[det(F_i)]^{1/2}}{P_i} exp[(X_j - V_i)^T F_i (X_j - V_i)/2] \tag{4.29}$$

où

$$P_i = \frac{1}{N} \sum_{j=1}^{N} u_{ij} \tag{4.30}$$

$$F_i = \frac{\sum_{j=1}^{N} u_{ij} (X_j - V_i)(X_j - V_i)^T}{\sum_{j=1}^{N} u_{ij}} \tag{4.31}$$

F_i étant la matrice de covariance floue du i^{me} groupement et P_i la probabilité *a priori* de sélectionner le i^{me} groupement.

Elle est censée permettre la prise en compte de la forme des groupements en fonction des directions privilégiées du sous-nuage. Ces directions correspondent aux éléments propres de la matrice d'autocorrélation du sous-nuage. En figure 4.5, nous illustrons le comportement de l'algorithme des C-Moyennes floues utilisant une telle distance pour la détection de deux classes dans chacun des ensembles tests de la figure 4.1. L'initialisation des centroïdes et des coefficients d'apartenance est réalisée par un paritionnement ISODATA classique utilisant la distance euclidienne. On appellera dorénavant cet algorithme de regroupement associé à la distance "exponentielle" méthode des *C-Moyennes Floues Exponentielles* ou algorithme *CMFE*[1] (cf. annexe B).

En fait, l'algorithme *CMFE* fournit notamment de bonnes performances dans les situatons délicates suivantes :

- les groupements ont des formes très différentes ; cette variabilité est contrôlée par la prise en compte de la matrice de covariance du groupement F_i à la manière de la distance de Mahalanobis ;
- les groupements ont des densités de points différentes ; cette variabilité est contrôlée par le rapport de la probabilité *a priori* P_i sur le volume flou $\sqrt{(det(F_i))}$;
- les groupements ont des tailles différentes ; ce paramètre est pris en compte par la probabilité *a priori* P_i.

[1]Dans l'article originel l'algorithme porte l'acronyme UFP-ONC pour "unsupervised fuzzy partition - optimal number of classes" et il intègre une procédure de détermination du nombre de classes optimal

La distance “exponentielle” agit comme un estimateur robuste, discréditant dans le calcul de la distance les points “trop éloignés”. En fait, l’expression de cette distance est très proche de l’expression de la densité de probabilité d’une distribution de type gaussien (voir équation 4.32, à l’inversion près entre autres.

$$f(X) = [(2\pi)^n \det(\Sigma)]^{-1/2} exp[-\frac{1}{2} X^t \Sigma^{-1} X] \tag{4.32}$$

où X est un vecteur de $\Re^n$.

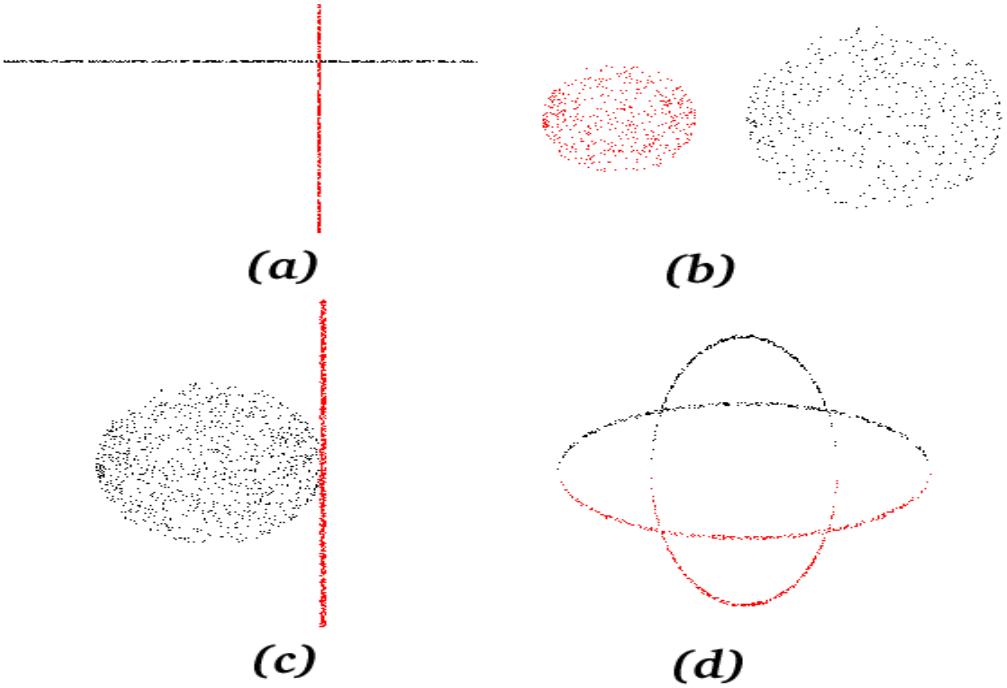

FIG. 4.5 – Regroupement en deux classes par la méthode des C-Moyennes floues utilisant la distance exponentielle sur les (a) groupements de nature linéaire (b) groupements de type compacts de densité en points différentes (c) groupements de types mixtes (d) groupements de contours d’ellipses

4.5.4.5 Récapitulatif

Nous avons exposé toute la difficulté pour un algorithme de partitionnement de traiter la variabilité des groupements à exhiber. Lorsque les objets sont com-

pacts et bien séparés, l'usage de la distance euclidienne est suffisante. L'usage de prototype droite et donc d'une distance point-droite permet de séparer des groupements linéaires. Lorsque les groupements ont des formes ellipsoïdales mais que la forme des groupements est malgré tout très variable, l'usage de la distance de Mahalanobis s'impose. Mais parmi tous ces choix possibles, seul l'algorithme *CMFE* permet de traiter dans son ensemble la variabilité évoquée au détriment toutefois de résultats théoriques de convergence vers une solution optimale et des temps de calcul plus longs. En revanche, expérimentalement, cette méthode semble surclasser toutes les autres du point de vue de la généricité. Finalement très peu utilisée, et lorsque c'est le cas, exclusivement dans le strict domaine de l'Analyse de Données, cette approche va montrer sa grande utilité dans les préoccupations de Vision par Ordinateur auxquelles nous nous intéressons plus particulièrement dans cette thèse.

Nous récapitulons dans un seul schéma 4.6 les différentes actions des différents algorithmes de regoupement flou testés sur nos données test, en faisant apparaître successivement une seul classe obtenue après paritionnement, l'autre classe se déduisant en complémentant à l'ensemble de départ repris en figure 4.6(a). Cette figure illustre le comportement optimum de l'algorithme *CMFE* sur ce type de données.

4.5.4.6 Variations en fonction du nombre de classes

En figure 4.7, nous illustrons le comportement de l'algorithme des C-Moyennes floues utilisant la *distance exponentielle* pour la détection d'un nombre de classes variable dans un ensemble test contenant à la fois des classes de topologie sphérique et linéaire.

Nous avons regroupé dans deux graphes en figure 4.8 et 4.9 le comportement des différentes mesures de validité de partition décrites en 4.5.3

On observe sur cet exemple, de façon tout à fait empirique, que la mesure de partitionnement la plus performante est la *Densité Moyenne de Partition (DMP)* qui fournit une partition optimale en cinq regroupements. Expérimentalement, cette mesure s'est révélée effectivement très performante et robuste dans l'ensemble de nos expériences.

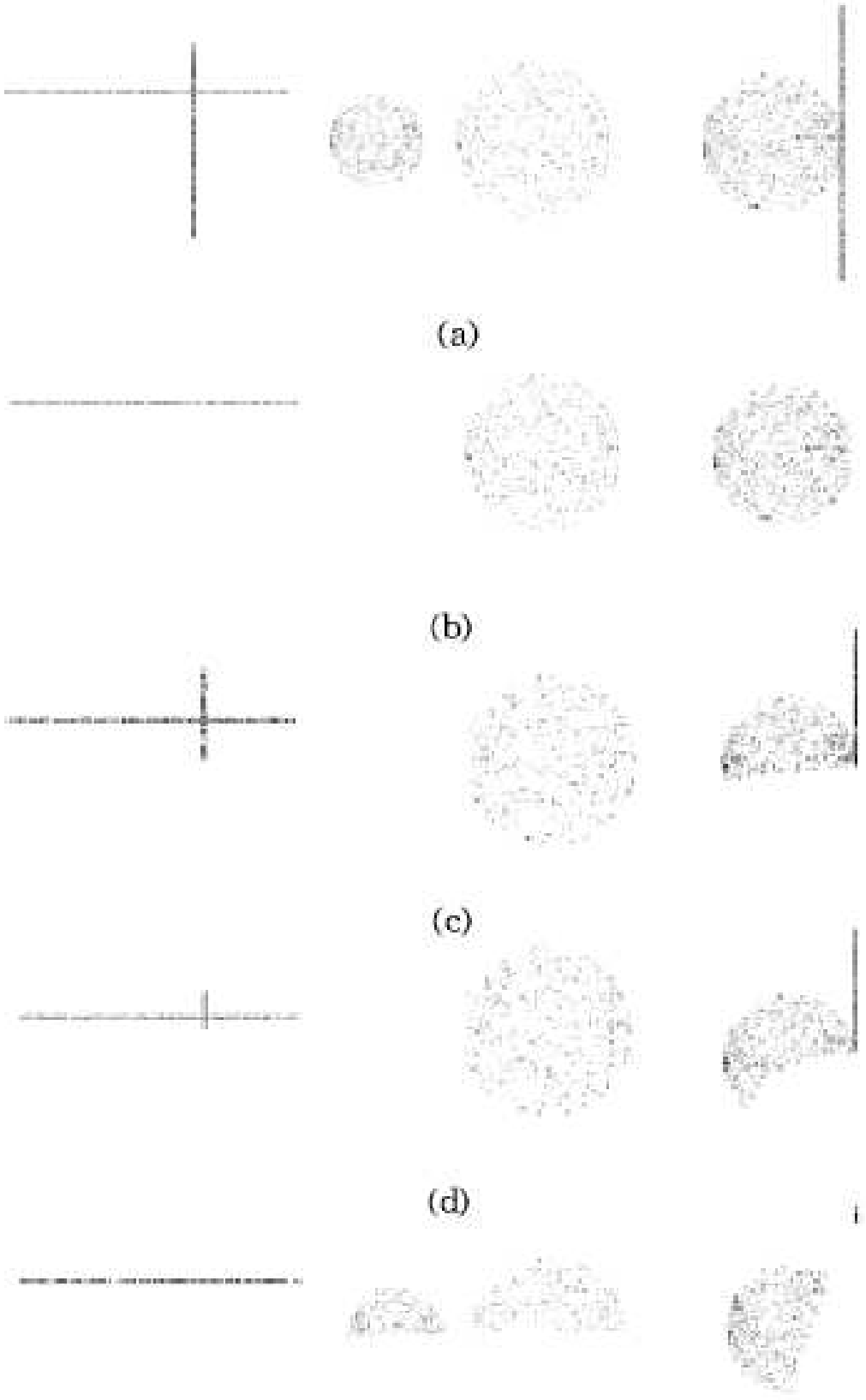

FIG. 4.6 – (a) Ensembles tests puis regroupement en 2 classes par la méthode des C-Moyennes floues et ne montrant qu'une seule des dexu classes (b) en utilisant

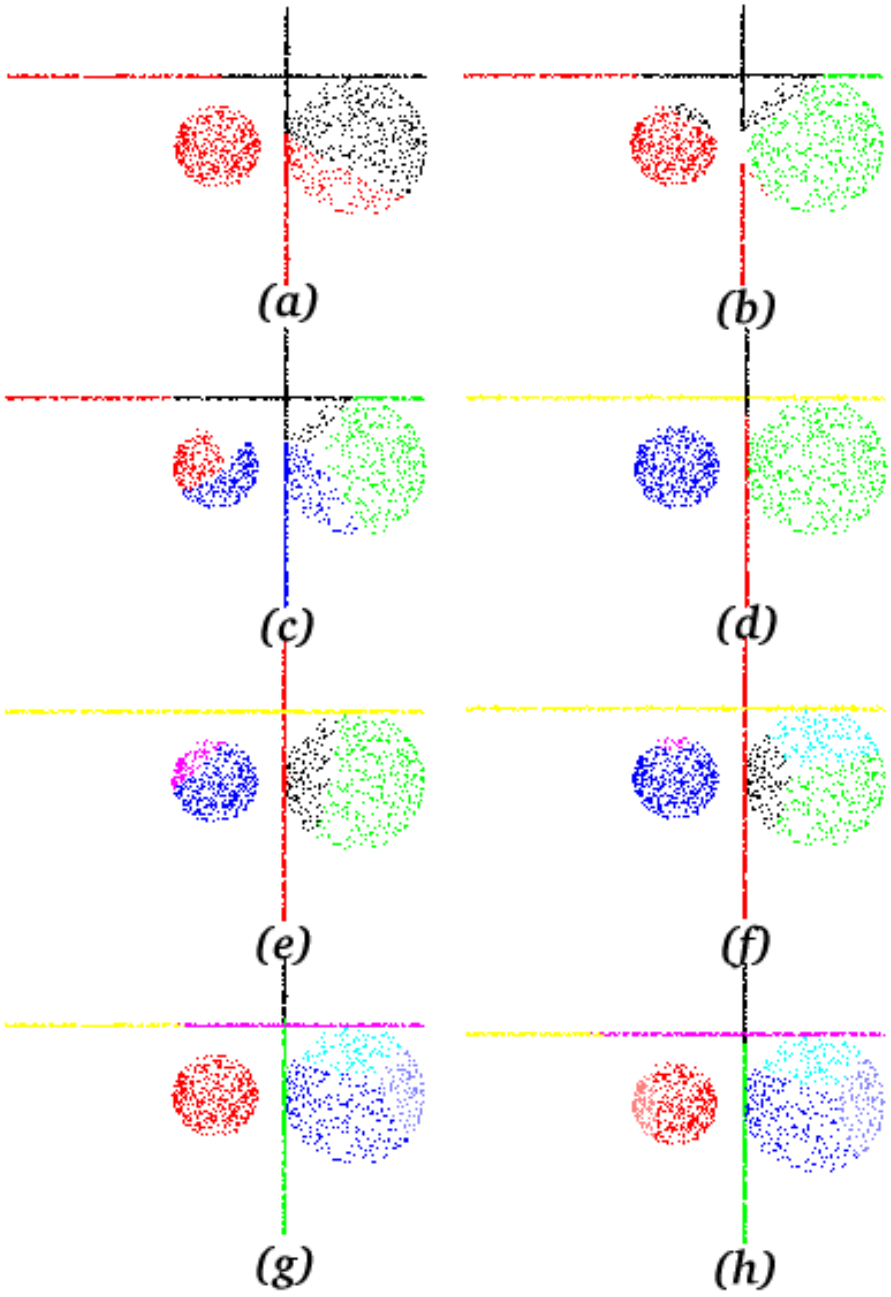

FIG. 4.7 – Regroupement par la méthode des C-Moyennes floues utilisant la distance exponentielle en (a) 2 classes (b) 3 classes (c) 4 classes (d) 5 classes (e) 6 classes (f) 7 classes (g) 8 classes (h) 9 classes

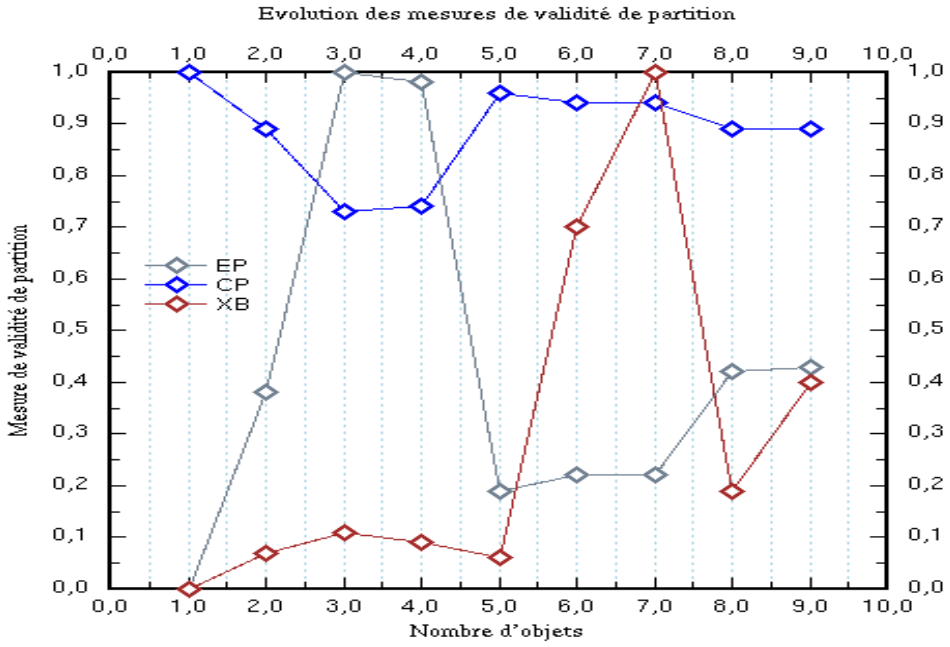

FIG. 4.8 – Evolution des critères CP, EP et XB

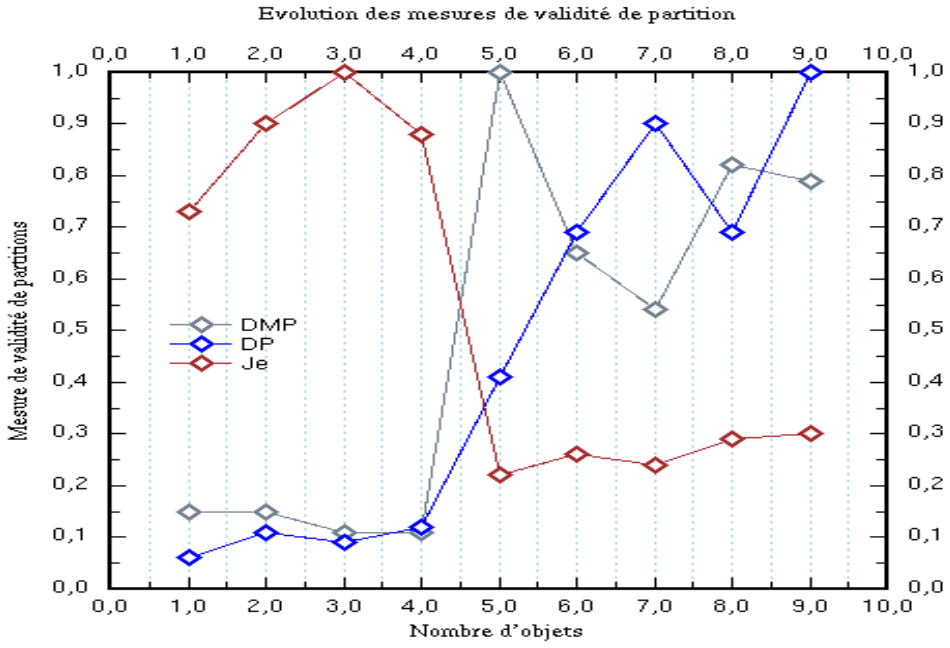

FIG. 4.9 – Evolution des critères DMP, DP et Je

4.6 Conclusion

Nous avons essayé dans cette partie de formaliser autant que faire se peut une discipline essentiellement expérimentale et algorithmique appelée regroupement de données par classification non supervisée. Après avoir dépeint grossièrement les choix s'offrant aux concepteurs de tels algorithmes, on a mis l'accent sur une technique spécifique développée il y une dizaine d'années par deux chercheurs I. Gath et A.B. Geva [GG89] qui s'avère expérimentalement la plus complète. Cette technique, en alliant des propriétés statistiques - par l'usage d'une distance s'assimilant à une mesure de maximum de vraisemblance (voir paragraphe 4.5.4.4) - à des propriétés algorithmiques efficaces - issues des méthodes de type ISODATA (voir paragraphe 4.5.1)- , constituera le fondement de notre système de vision artificielle 3D.

Chapitre 5

α-Opérations Morphologiques

5.1 Introduction

Les applications informatiques traitent souvent des données se présentant sous la forme abstraite d'un nuage de points en deux ou trois dimensions, et il est quelquefois utile de pouvoir calculer ce qu'on pourrait appeler la "forme d'un tel ensemble". C'est notamment le cas dans des problèmes de reconstruction stéréoscopique où les scènes sont disponibles sous la forme de nuages de points 3D qu'il s'agit d'analyser en termes d'obstacles et de zones navigables puis de reconstruire objet par objet sous la forme de structures géométriques de type maillage de Delaunay.

Le problème de la segmentation de tels nuages de points désorganisés a déjà été abordé dans la partie précédente où l'on utilise des techniques de regroupement flou s'inspirant des nuées dynamiques [BCGW81][Gus79] et améliorées dans [GG89] pour prendre en compte la forme des ensembles en cours de constitution. Cette démarche permet d'obtenir des ensembles de points censés représenter les objets de la scène. La description de la forme de ces objets devrait alors constituer le dernier traitement à opérer pour décrire la scène.

L'utilisation des α-formes 2D ou 3D introduites au départ par H. Edelsbrunner [EK83] donne une définition formelle de la forme d'un ensemble de points et peuvent être utilisées à escient pour reconstruire des objets 3D décrits par un nuage de points 3D [LGCS99]. Ces structures construisent les formes en "sculptant" les triangulations de Delaunay associées mais de façon plus formelle que les techniques préconisées dans [Boi84] [BG92]. Or, le manque de précision dans la

reconstruction des points 3D, imposé à la fois par des préoccupations de temps-réel en vision par ordinateur et par les limites technologiques actuelles des calculateurs ne permet pas de prétendre reconstruire très précisément la forme 3D des ensembles de points segmentés. Etant données ces contraites, il est suffisant de reconstruire une approximation de la silhouette 2D des objets par projection des nuages de points sur les plans d'approximation optimaux au sens des moindres carrés médians par exemple. C'est la raison pour laquelle nous nous limitons dans cet exposé à la description d'opérateurs de forme sur des ensembles de points 2D désorganisés.

L'intuition de filtrer des nuages de points désorganisés par le biais des graphes de Voronoï commence à se préciser dans la littérature dédiée à la visualisation graphique. N. Amenta emploie dans [ABK98] [AB98] le terme de filtrage de Voronoï pour retrouver la forme d'un nuage de points quelconque en dimension 2, puis plus difficilement en dimension 3. Luc Vincent introduit lui aussi la notion de filtrage morphologique de graphes [Vin89] [Vin92], tout comme Jean Serra [SCS95]. Dans cette veine, nous commençerons par rappeler les définitions de diagramme de Voronoï et de triangulation de Delaunay. Puis, après avoir expliqué ce qu'est une α-forme et comment on l'obtient, on précisera notre contribution à la définition d'opérateurs d'α-érosion, α-dilatation et α-ouverture agissant sur des représentations à base de graphe (et notamment des structures de nuages de points désorganisés décrites par leur triangulation de Delaunay). L'illustration de leurs comportements fera aparaître des résultats similaires aux opérateurs de morphologie mathématique classique appliqués à des images d'intensité structurées régulièrement. De façon similaire, nos opérateurs permettent de filtrer la forme des données spécifiques traitées. En définitive, la définition de tels opérateurs morphologiques sur des nuages de points désorganisés constitue l'un des apports originaux de nos recherches.

5.2 Définitions

5.2.1 Diagramme de Voronoï

Définition 5.1 *Soit un ensemble* M *de* n *points de* $\mathbf{E}^d$, $M_1, \ldots, M_n$, *qu'on appellera sites dans la suite pour les différencier des autres points de* $\mathbf{E}^d$. *On associe à*

chaque site M_i la région $V(M_i)$ de E^d constituée des points plus proches de M_i que des autres sites :

$$V(M_i) = \{X \in \mathbf{E}^d, \delta(X, M_i) \leq \delta(X, M_j)\ \forall j \neq i\} \tag{5.1}$$

où δ désigne la distance euclidienne en général, mais peut désigner n'importe quelle autre distance de même que les sites ne sont pas forcément ponctuels. Le maillage de l'espace ainsi obtenu est appelé diagramme de Voronoï.

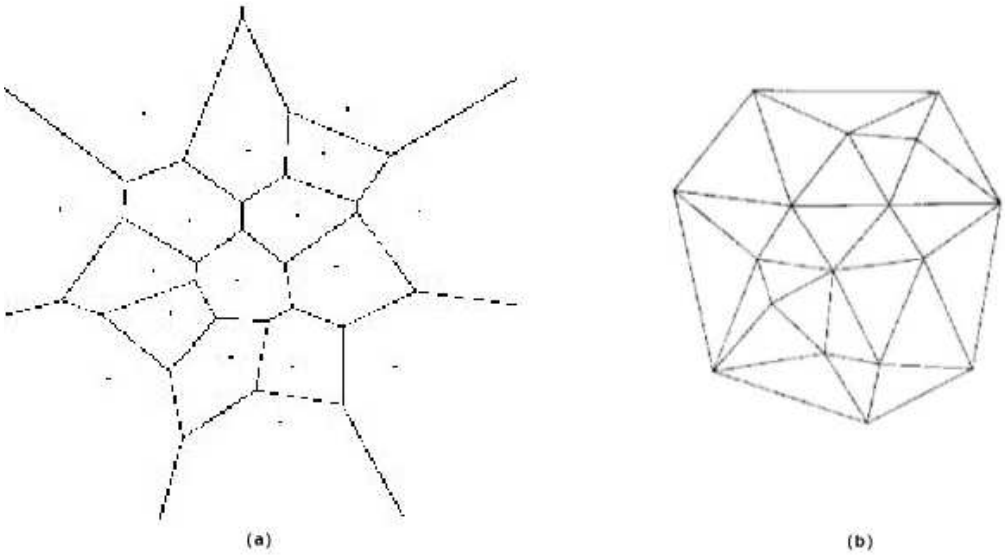

FIG. 5.1 – (a) Diagramme de Voronoï et (b) Triangulation de Delaunay d'un même nuage de points

La région $V(M_i)$ peut également être vue comme l'intersection d'un nombre fini de demi-espaces (limités par des hyperplans médiateurs des segments $MiMj$, $j = 1, \ldots, n, j \neq i$) ; c'est donc un polytope convexe non borné. Les $V(M_i)$ constituent avec leurs faces un complexe cellulaire appelé diagramme de Voronoï, dont un exemple est représenté en figure 5.1(a). On peut également interpréter la région $V(M_i)$ comme l'ensemble des centres des sphères passant par M_i dont l'intérieur ne contient aucun des $M_j, j \neq i$.

5.2.2 Complexe de Delaunay

Le complexe de Delaunay de n points $M_1, M_2, \ldots, M_n$ de $\mathbf{E}^d$ est un complexe dual du diagramme de Voronoï dont les arêtes relient deux sites voisins au sens

du diagramme de Voronoï, c'est-à-dire dont les cellules associées partagent une arête. Un exemple de tel complexe est illustré en figure 5.1(b).

Algorithmiquement, calculer le diagramme de Voronoï d'un ensemble de points ou la triangulation de Delaunay de cet ensemble est donc équivalent en terme de complexité. Il est à noter que l'enveloppe complexe d'un ensemble de points peut elle aussi être dérivée du diagramme de Voronoï.

5.3 Caractéristiques géométriques et algorithmiques

5.3.1 Propriétés

En dimension n, chacune des n-faces du complexe de Delaunay (c'est-à-dire des tétraèdres en dimension 3 et des triangles en dimension 2) de $\mathbf{M}$, ensemble des n sites, admet une sphère circonscrite passant par tous ses sommets ; l'intérieur de chacune de ces sphères ne contient aucun des M_i.

Propriété 5.1 *Parmi toutes les triangulations $T(\mathbf{M})$ de $\mathbf{M}$, les triangulations de Delaunay sont optimales du point de vue de la finesse et du grain.*

Proposition 5.1 *Deux points M_i et M_j déterminent un côté de la triangulation de Delaunay si et seulement si il existe une sphère passant par ces deux points qui ne contient aucun autre point de $\mathbf{M}$ dans son intérieur.*

Avec ces quelques résultats de base, on peut construire le diagramme de Voronoï, la triangulation de Delaunay, ou l'enveloppe convexe d'un ensemble de sites en une complexité au pire en $O(nlogn + n\lceil d/2 \rceil)$ où n est le nombre de sites et d la dimension de l'espace [JDY95].

5.3.2 Schémas algorithmiques

Il existe plusieurs schémas algorithmiques différents pour construire de tels graphes : incrémental, diviser et régner, stochastique. Dans ce type d'algorithmes géométriques, l'essentiel réside dans la structure de données utilisée pour stocker les résultats intermédiaires et finaux, comme la liste des facettes créées, validées et invalidées par exemple (cf. Annexe A).

Globalement, pour chaque algorithme, partant d'une triangulation quelconque, des processus d'inversions de n-faces (arêtes en dimension 2, face triangulaire

en dimension 3) permettent de contraindre celle-ci selon un critère d'optimalité locale $a(T)$ (où T est une 2-face de la triangulation (autrement dit un triangle) en dimension 2). On montre que la triangulation de Delaunay est optimale parmi les triangulations possibles d'un ensemble de points du point de vue du critère de l'angle max-min où $a(T) = 1/$(l'angle maximum de T).

5.3.3 Forme et graphe de Delaunay

La notion de forme d'un objet peut être approximée par un sous-graphe de la triangulation de Delaunay d'un ensemble de points échantillonés sur la surface [Boi84].

Si l'on dispose d'une triangulation de Delaunay de l'ensemble des points $\mathbf{M}$, on peut également définir un ensemble de descripteurs de forme appelé spectre des α-formes qui constitue également une famille de sous-graphes de la triangulation de Delaunay [EK83]. Ces formes correspondent à une extension de la notion classique d'enveloppe convexe. On donne ci-après les définitions données par H. Edelsbrunner dans [EK83].

Définition 5.2 *Soit α un réel positif. L'α-enveloppe de $\mathbf{M}$ peut être définie comme l'intersection de tous les disques fermés de rayon $1/\alpha$ contenant tous les points de $\mathbf{M}$.*
Soit α un réel négatif. L'α-enveloppe complexe de $\mathbf{M}$ est définie comme l'intersection de tous les complémentaires des disques fermés de rayon $-1/\alpha$ contenant tous les points de $\mathbf{M}$.
Si $\alpha = 0$, on définit la 0-enveloppe comme l'enveloppe convexe traditionnelle, c'est-à-dire que nos disques se transforment en demi-plans fermés.

Il définit ensuite les disques généralisés de rayon r. Si r est positif ou nul, ils correspondent aux disques habituels. Si r est négatif, ils correspondent au complémentaire des disques de rayon $-r$ associé.

Définition 5.3 *Un point M_i dans un ensemble est α-extrême dans $\mathbf{M}$ s'il existe un disque fermé généralisé de rayon $1/\alpha$, tel que M_i est sur sa frontière et qu'il contient tous les points de $\mathbf{M}$. Si pour deux α-extrêmes points P et Q il existe un disque fermé généralisé de rayon $1/\alpha$ contenant ces deux points sur sa frontière et tous les autres points , alors P et Q sont α-voisins.*

Définition 5.4 *Etant donné* $\mathbf{M}$ *un ensemble de points et un réel* α*, l'*α*-forme de* $\mathbf{M}$ *est le graphe de segments dont les sommets sont les points* α*-extrêmes et dont les côtés relient ces points* α*-extrêmes respectifs.*

Spécifiquement, toute α-forme négative est un sous-graphe de la triangulation de Delaunay décrite précédemment. M.Melkemi [Mel97] propose un algorithme simple de calcul de ces α-formes à partir d'une triangulation de Delaunay.

H. Edelsbrunner [EK83] a en effet montré que pour chaque côtés e de la triangulation de Delaunay il existe deux réels α_{min} et α_{max} tels que e est un côté de l'α-forme de $\mathbf{M}$ si et seulement si $\alpha_{min} < \alpha < \alpha_{max}$.

L'identification de ces côtés de la triangulation de Delaunay de $\mathbf{M}$ se fait alors comme suit dans le cas des α-forme négatives.

Soit $r = -1/\alpha$ $(\alpha < 0)$ un réel. Pour un côté P_iP_j, on calcule deux réels r_{min} et r_{max}. Un côté P_iP_j appartient à l'α-forme si $r_{min} < r < r_{max}$. On calcule r_{min} et r_{max} ainsi :

- Cas 1 : P_iP_j est commun à deux disques de Delaunay c_1 et c_2 ;
  alors $r_{min} = \frac{1}{2d_e(P_i,P_j)}$, $r_{max} = max(d_1, d_2)$ où $d_1 = d_e(c_1, P_i)$ et $d_2 = d_e(c_1, P_j)$;

- Cas 2 : P_iP_j est associé à un seul cercle de centre c passant par P_i, P_j et P_k ; alors $r_{max} = +\infty$ et si P_k et c tombent du même côté de P_iP_j alors $r_{min} = \frac{1}{2d_e(P_i,P_j)}$ sinon $r_{min} = d_e(P_i, c)$.

Comme le rappele 15 ans plus tard Nina Amenta [ABK98], les α-formes sont des constructions paramétriques qui associent une forme polyédrique à un ensemble non organisé de points. Pour clarifier la définition de ces structures, on a peu à peu inversé les paramètres α et r ce qui fait qu'à présent le paramètre α est associé aux rayons des sphères circonscrites aux simplexes. Un simplexe (arête, triangle au tétrahèdre) est inclus dans l'α-forme s'il possède une sphère circonscrite dont l'intérieur ne contient pas de points de $\mathbf{M}$ et de rayon au moins α. Ces objets ont encore été utilisés comme étape préliminaire à une chaîne de reconstruction [BBX95].

Parallèlement *en dimension 2*, différents résultats théoriques sur diverses approches de reconstruction de courbes continues à partir de triangulation de Delaunay ont été publiés [Att97] [BB97] [ABE99]. On voit donc que ces structures

offrent encore un terrain encore fertile de recherches.

L'une des améliorations de l'élégante technique introduite par N. Amenta [ABK98] pour la reconstruction de surface réside dans sa nature intrinsèquement 2D et par le fait qu'elle n'utilise pas de paramètre global de contrôle de la forme désirée comme le paramètre α des α-formes, mais le calcule localement automatiquement. Toutefois, ceci ne nous est pas apparu comme un inconvénient dans nos expérimentations. Pour avoir une "bonne description", des objets rencontrés dans notre application, le cadre des α-objets convient tout à fait. En effet, nous n'avons pas à faire à des objets manufacturés ou dont le niveau de détail désiré soit très précis. D'autre part, nous travaillerons sur des projections 2D des nuages de points associés aux objets rencontrés (cf. partie 6). En l'occurence, en utilisant l'algorithme décrit en Annexe A pour le calcul de la triangulation de Delaunay et en conservant les points auxilaires créés "à l'infini", une valeur de α égalant trois fois la valeur médiane de l'ensemble des valeurs des rayons des cercles circonscrits aux triangles du maillage global (incluant les points "à l'infini") donne une "bonne description" de la forme des obstacles rencontrés. Aussi, dans la mouvance des filtrages utilisant les stuctures de Voronoï évoquée, nous espérons apporter une contribution au filtrage de forme sur la base des α-objets.

5.4 α-Objets

Nous commençons par expliquer plus en détails ce que sont une α-forme et ses dérivés et comment toutes ces structures sont obtenues [EM94].

D'abord introduite par H. Edelsbrunner [EK83], la notion d'α-forme donne une définition formelle de ce que peut être la forme d'un nuage de points. Plus précisément, elle définit une famille discrète de formes dont le niveau de détail est géré par le paramètre α qui contrôle la courbure maximale autorisée dans la description de la forme. Nous nous concentrons sur la structure dérivée dite α-complexe d'un ensemble de points S qui peut être vue comme une triangulation de l'intérieur de l'α-forme correspondante et qui peut être également définie comme un sous-graphe de la triangulation de Delaunay Del de S. Intuitivement, une fois la triangulation de Delaunay obtenue [dB97][CJMS00], l'α-complexe agit comme une gomme sphérique effaçant les triangles de Del capable de recevoir une boule ouverte B_α de rayon α ne contenant aucun point de S. Très liée à

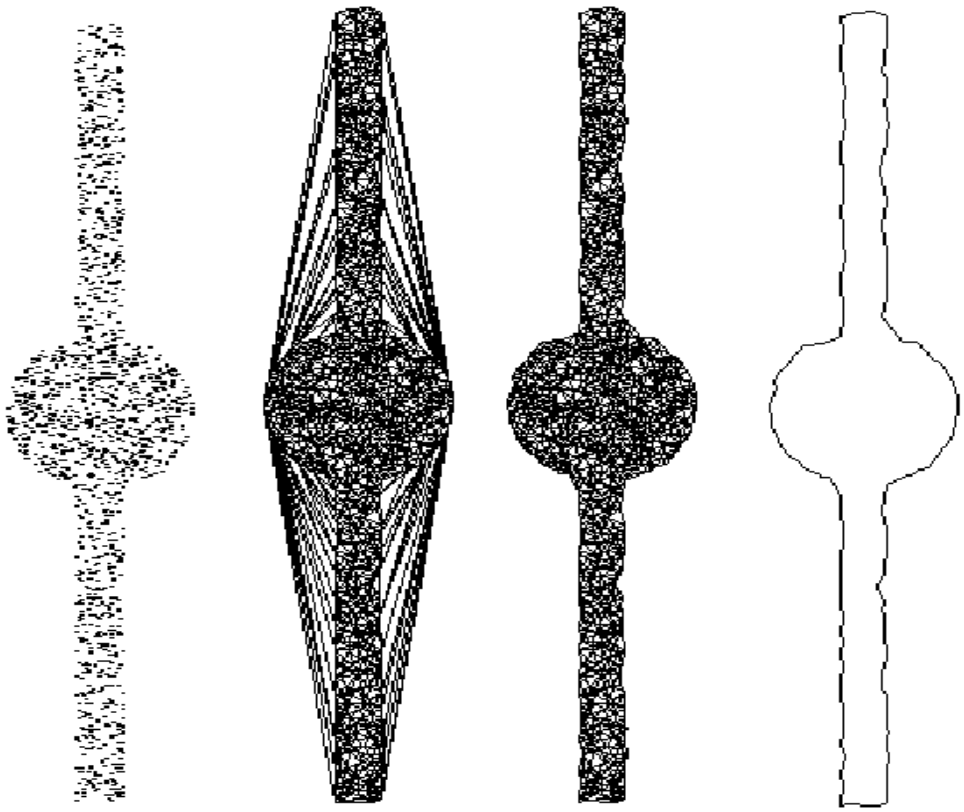

FIG. 5.2 – De la gauche vers la droite : Ensemble de points initial ; Triangulation de Delaunay ; 0.1-complexe ; 0.1-forme (intérieur) ; 0.1-enveloppe (frontière)

la notion d'α-forme et d'α-complexe, l'α-enveloppe d'un ensemble de points est une généralisation de l'enveloppe convexe d'un ensemble de points. La figure 5.2 résume toutes ces structures pour un nuage de points 2D désorganisé et synthétique.

Nous prenons la précaution de mentionner que les définitions des α-objets ne sont pas encore figées. Nous avons choisi d'expliciter celles données dans les articles cités en référence.

Définissons une boule vide α-boule comme une boule de rayon α ne contenant aucun point de S. Alors, l'α-forme de S est définie comme le complémentaire de la réunion de toutes les α-boules. Or, la morphologie mathématique est bien connue pour ses relations ensemblistes : "$B \subset S$", "$B \cap S = \emptyset$" où S est l'ensemble à analyser et B est l'élément structurant dont la forme dépend des besoins de l'analyse. Ces relations constituent la base des opérateurs de morphologie mathématique classique tels que l'érosion et la dilatation. Mais les définitions sont assez différentes même si les α-complexes semblent effectivement éroder l'∞-complexe correspondant à l'enveloppe convexe (fig.1). Définissons les k-simplexes $\sigma_T = conv(T)$ (c'est-à-dire l'enveloppe convexe de T), avec $T \subseteq S$ et $|T| = k+1$ pour $0 \leq k \leq 2$. Définissons ρ_T comme le rayon de la sphère circonscrite à σ_T. Pour chaque simplexe $\sigma_T \in Del$ il existe un unique intervalle tel que σ_T soit une face de l'α-forme F_α si et seulement si α appartient à cet intervalle. Soit $up(\sigma_T)$ l'ensemble de toutes les faces incidentes à σ_T dont la dimension est supérieure de un à celle de σ_T, c'est-à-dire $up(\sigma_T) = \{\sigma_{T'} \in Del \ |T \subset T' \ et \ |T'| = |T|+1\}$. Alors, pour chaque σ_T, on dérive deux valeurs λ_T et μ_T :

$$\begin{cases} si\ |T| = 3, & \lambda_T = \mu_T = \rho_T \\ sinon & \begin{cases} \lambda_T = min\left(\{\lambda_{T'} \,|\sigma_{T'} \in up(\sigma_T)\}\right) \\ et \\ \mu_T = max\left(\{\mu_{T'} \,|\sigma_{T'} \in up(\sigma_T)\}\right)\} \end{cases} \end{cases}$$

Enfin, un simplexe est dit :

$$\begin{cases} Intérieur & si\,\sigma_T \notin \partial F_\alpha \\ Régulier & si\,\sigma_T \in \partial F_\alpha\, en\, bornant\, un\, simplexe \\ & de\, dimension\, supérieure\, dans\, C_\alpha \\ Singulier & si\,\sigma_T \notin \partial F_\alpha\, sans\, borner\, un\, simplexe \\ & de\, dimension\, supérieure\, dans\, C_\alpha \end{cases}$$

où ∂F_α désigne la frontière de la forme-α F_α.

Alors, le tableau 5.4 explique comment construire les α-objets.

σ_T *est...*		*Singulier*	*Régulier*	*Intérieur*
Triangle				$(\rho_T, \infty[$
Arête,	$\notin \partial conv(S)$	(ρ_T, λ_T)	(λ_T, μ_T)	$(\mu_T, \infty[$
	$\in \partial conv(S)$	(ρ_T, λ_T)	$(\lambda_T, \infty[$	
Sommet,	$\notin \partial conv(S)$	$[0, \lambda_T)$	(λ_T, μ_T)	$(\mu_T, \infty[$
	$\in \partial conv(S)$	$[0, \lambda_T)$	$(\lambda_T, \infty[$	

TAB. 5.1 – Obtention d'une α-forme

Il faut remarquer que chaque arête appartenant à $\partial conv(S)$ est le côté d'un triangle dont le rayon de la sphère circonscrite est infini, avec un des sommets à l'infini. Théoriquement, l'α-complexe C_α est constitué de tous les simplexes intérieurs, réguliers, et singuliers pour un α donné, mais pour des raisons de simplicité, on l'assimilera à l'intérieur de l'α-forme F_α qui est maillé seulement par les *triangles* intérieurs (on met de côté les arêtes et les points intérieurs). La frontière de l'α-forme est formée par l'ensemble des arêtes régulières et de leurs sommets.

5.5 Opérateurs morphologiques

Une fois un α-complexe optimal de S obtenu, par exemple celui de volume minimal et contenant tous les points de S ou bien celui correspondant au choix explicité plus haut, notre but est de définir des opérateurs morphologiques agissant sur des nuages de points désorganisés pour filtrer la forme ainsi décrite. A partir de maintenant, l'α-complexe C_α est assimilé à une triangulation de l'intérieur de l'α-forme, c'est-à-dire à une sous-triangulation T_α obtenue à partir de Del.

5.5.1 Erosion

L'α-érosion est définie comme un sous-graphe de Del obtenu par propagation des valeurs μ_T aux triangles voisins. Ainsi, à chaque triangle $conv(T)$ de C_α est associé des valeurs e_T^k définies par :

$$\begin{cases} e_T^k = max\left\{e_{T'}^{k-1} \,|T' \in voisin(T)\right\} \\ et \\ e_T^0 = \mu_T = \rho_T \end{cases} \tag{5.2}$$

où $voisin(T)$ est l'ensemble de tous les triangles T de Del partageant au moins un sommet avec le triangle T, c'est-à-dire :

$$voisin(T) = \{T' \in Del \,|T' \cap T = \emptyset \, et \, |T'| = |T| = 3\} \tag{5.3}$$

L'α-érodé d'ordre k est défini comme la réunion de tous les triangles de C_α pour lesquels e_T^k est inférieur à α, c'est-à-dire :

$$\alpha^k - érodé = \left\{T' \in Del \,\middle|\, e_{T'}^k < \alpha \, et \, |T'| = 3\right\} \tag{5.4}$$

A partir de maintenant, toute α-transformation d'ordre 1 ou α^1-transformation est appelée α-transformation. Il faut remarquer que dans ces conditions l'α^k-érodé de Del est identique à l'α-érodé de l'α^{k-1}-érodé, en remplaçant e_T^0 par e_T^{k-1}. Si bien que l'α^k-érosion est équivalente à la réalisation de k α-érosions successives en propageant les valeurs e_T^k.

5.5.2 Dilatation

La dilatation de toute sous-triangulation T_α de Del est définie comme la réunion de tous les triangles de Del partageant au moins un sommet avec T_α. La dilatation de Del donne l'espace 2D entier.

5.5.3 Ouverture

L'α-ouverture d'ordre k (α^k-ouverture) est définie comme le dilaté de l'α-érodé d'ordre k de C_α. Il faut noter que réaliser une α^k-ouverture ne revient pas cette fois-ci à réaliser k α-ouvertures successives en propageant les valeurs e_T^k, mais à réaliser k α-érosions successives suivies d'une α-dilatation dans le but de filtrer le contour de la forme obtenue (cf. figure 5.3).

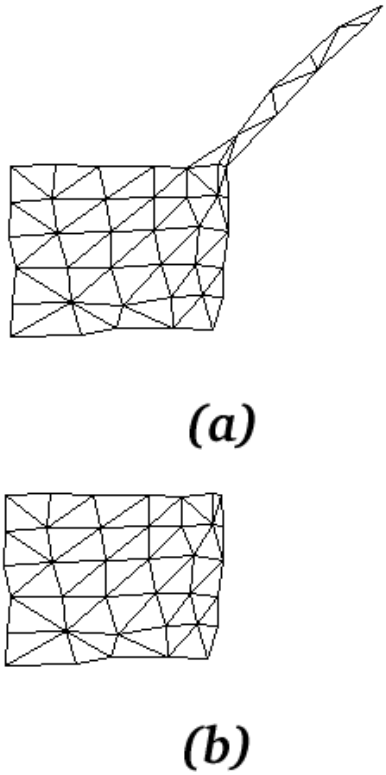

FIG. 5.3 – Filtrage de la forme (a) par ouverture d'ordre 1 (b)

5.5.4 Complexité

La complexité de calcul de tous ces opérateurs est la même que celle de la triangulation de Delaunay c'est-à-dire au pire en *O(Nlog(N))* [JDY95] pour un nuage de N points.

5.6 Illustrations

Les opérations d'ouverture morphologique sur un ensemble sont utilisées pour filtrer géométriquement les contours de l'objet qu'il représente. Par exemple, de petites ouvertures étroites et de petites excroissances peuvent être réduites par des opérations d'ouverture, ce qui pourrait être intéressant dans nos applications robotiques en vision stéréoscopique. En effet, le processus de segmentation de nuages de points 3D en sous-ensembles peut créer des formes non régulières à cause des points aberrants ou mal affectés.

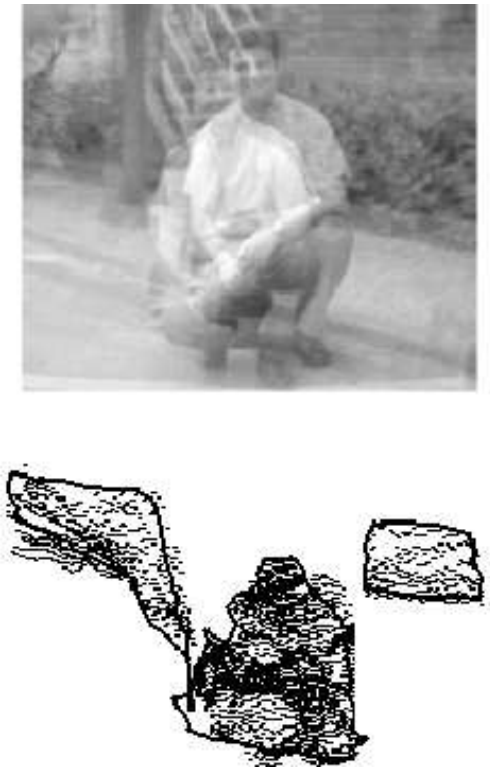

FIG. 5.4 – De haut en bas : Scène observée ; Résultat de la segmentation du nuage de points 3D représentatif de la scène à moins de 7 mètres et les α-formes associées.

Par exemple, la figure 5.4 illustre le résultat en segmentation de la scène représentant un homme accroupi et la délimitation de la forme des deux ensembles de points extraits par des α-formes optimales, l'un pour le sol derrière et l'autre pour l'homme. On précise d'une part que l'image de la scène fait apparaître sur deux planches couleurs l'image droite et l'image gauche du système de stéréoscopie (d'où l'effet de flou), et que d'autre part on ne retient que les points reconstruits situés à moins de 7 mètres de distance à cause de la faible précision en reconstruction. On constate encore que l'α-forme optimale représentant le sol sépare correctement les deux parties du sol. Le paramètre α vaut 0.4 pour cette dernière α-forme. Dans nos expériences, la valeur de α optimale a été fixée correspondant au triple de la valeur médiane des rayons de centres circonscrites à chaque triangle du maillage de Delaunay associé. L'application d'une 0.4-ouverture sur l'ensemble de points représentant le sol permet d'éliminer l'excroissance du bas comme illustrée en figure 5.5.

La figure 5.6 montre plus précisément comment la petite excroissance en bas de la forme est ainsi effacée en faisant apparaître superposés le 0.4-complexe, le 0.4-érodé, et le 0.4-ouvert sachant que :

$$\alpha - erode \subset \alpha - ouvert \subset \alpha - complexe$$

FIG. 5.5 – De gauche à droite : Ensemble de points initial ; 0.1-complex ; 0.1-ouverture

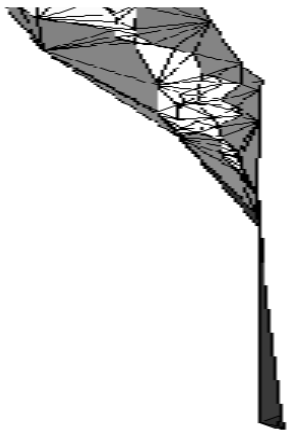

FIG. 5.6 – Détails superposés de la 0.1-ouverture : en blanc le 0.1-érodé, en gris le 0.1-ouvert, et en noir le 0.1-complexe.

Il peut être parfois utile de faire des mesures sur des parties constitutives d'un objet. Dans ce but, ce dernier doit être segmenté en ses différentes parties constitutives. La morphologie mathématique peut alors utiliser des érosions successives pour obtenir des germes pour chaque partie et réaliser alors une dilatation géodésique de ces derniers (il suffit de penser à la séparation de poumons en imagerie

médicale). Le même type de transformations peut être réalisé sur les deux cercles liés de la figure 5.7. En appliquant une α-ouverture d'ordre 7, et en dilatant ensuite géodésiquement parallèlement les deux germes obtenus à l'intérieur de l'α-complexe et en empêchant qu'un triangle n'appartienne à deux germes dilatés différents, on obtient les deux cercles segmentés en figure 5.7.

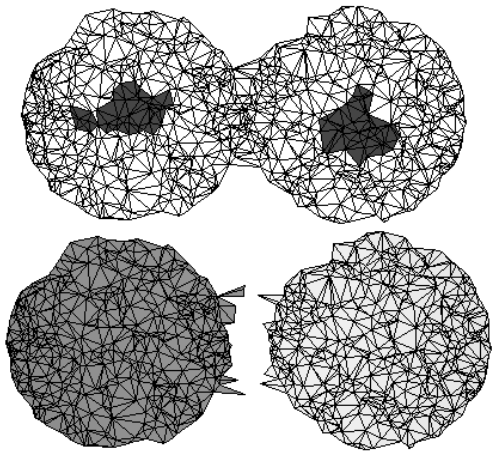

FIG. 5.7 – De haut en bas : 0.8-ouverture d'ordre 7 ; Dilatation géodésique des deux germes

Lorsqu'on a des nuages de points disséminés, ces opérations morphologiques définies sur des nuages de points désorganisés permettent de faire du regroupement par filtrage de forme. La figure 5.8 illustre cet usage en montrant en plus l'effet d'α-érosions successives.

La notion d'érosion successive peut également être appliquée à la structure d'α-enveloppe, et pas seulement au maillage de la structure d'α-complexe comme précédemment. La propagation des valeurs e_T^k sur les faces est alors remplacée par la propagation de valeurs e_A^k associées aux arêtes définies par :

$$e_A^k = max\{e_T^k, e_{T'}^k\} \tag{5.5}$$

où T et T' sont les deux triangles adjacents à l'arête. Cette version est illustrée en figure 5.9.

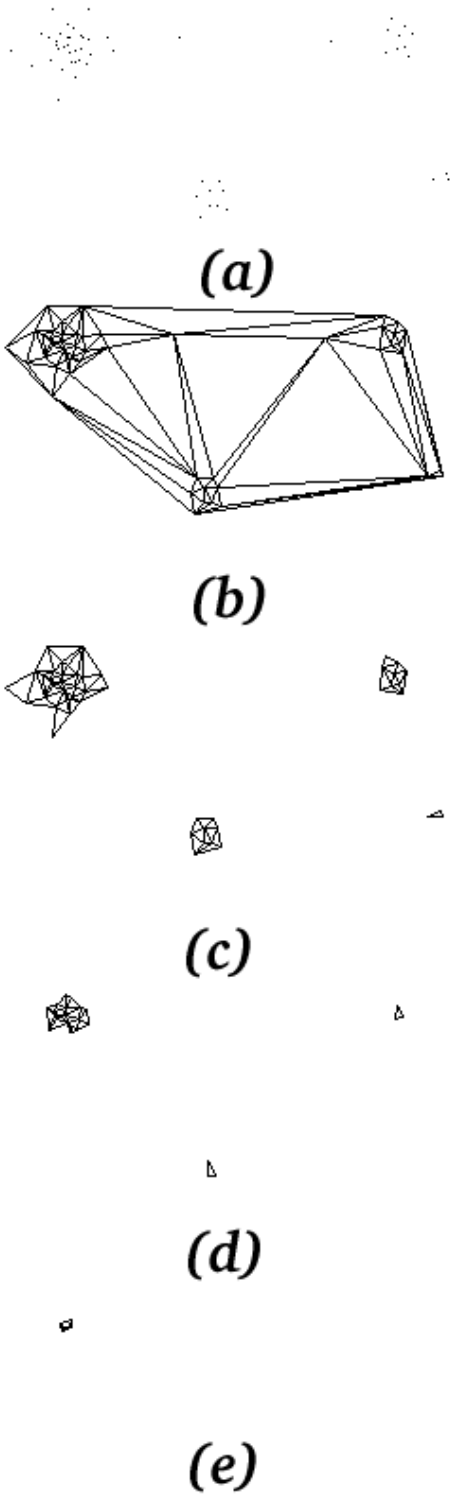

FIG. 5.8 – Regroupement par erosions successives

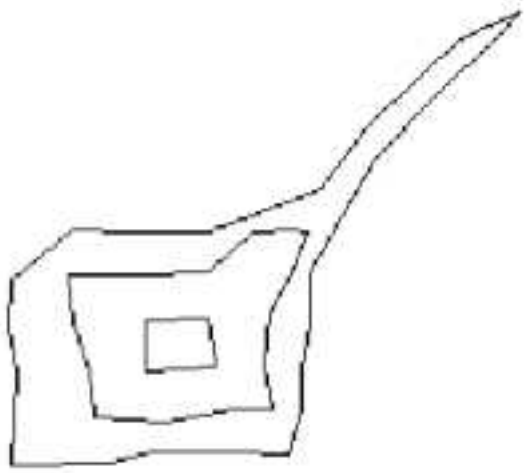

FIG. 5.9 – Erosions successives d'une α-enveloppe

5.7 Liens avec la morphologie mathématique classique

En imagerie binaire, ce sont souvent des informations quantitatives que l'on cherche à extraire. La morphologie mathématique classique possède un fondement ensembliste. Elle caractérise la forme dans le cadre de la logique booléenne et le contexte des treillis par l'utilisation de relations ensemblistes et d'ensembles géométriques arbitraires appelés éléments structurants [Ser87][SM88] . On définira ainsi les érosions ou dilatations morphologiques de formes binaires en 2D par réunion successives d'éléments structurants intersectant la forme ou s'incluant dans la forme à analyser.

Définition 5.5 *L'érosion morphologique de X par l'élément structurant B est définie par :*
$E = \{x/B_x \subset X\}$
où $B_x = \{b + x/b \in B, x \in X\}$ (B_x est l'ensemble des translatés de B par x)

Définition 5.6 *La dilatation morphologique de X par l'élément structurant B est définie par :*
$D = \{x/B_x \cap X \neq \emptyset\}$

De ces fondements ensemblistes s'ensuivent toute une série de propriétés telles que la croissance, l'anti-extensivité, l'idempotence. Par ailleurs, les fondements

topologiques, fonctionnels et probabilistes de cette discipline en ont fait également son succès sur les images en niveaux de gris.

Parallèlement, on peut donner une interprétation fonctionnelle des opérateurs morphologiques développés dans la théorie initiale. Si l'on considère une image comme une fonctionnelle dont les valeurs prises sur le treillis de l'image discrétisée sont les valeurs d'intensité de chaque pixel, les définitions de l'érosion et de la dilatation peuvent s'énoncer différemment. Que l'image soit binaire ou en niveau de gris, ces définitions, qui s'appuient sur la structure de sous-graphes que l'on peut associer à toute fonction, fournissent une autre procédure d'obtention des formes érodées et dilatées.

Définition 5.7 *L'érosion morphologique de la forme X définie par la fonction $f(x, y)$ définie sur le pavage discret de l'image, par l'élément structurant B est définie par :*
$E(x, y) = inf_{voisinage\ defini\ par\ B}\{f(x, y)\}$

Définition 5.8 *La dilatation morphologique de la forme X définie par la fonction $f(x, y)$ définie sur le pavage discret de l'image, par l'élément structurant B est définie par :*
$D(x, y) = sup_{voisinage\ defini\ par\ B}\{f(x, y)\}$

Le lecteur se persuadera que cette vision fonctionnelle concorde avec la vision ensembliste dans le cadre de la théorie des ensembles classique si la fonction $f(x, y)$ prend ses valeurs dans $0, 1$.

Revenons à notre description de la forme d'un nuage de points par les structures géométriques à base de graphes que sont les formes-α . Comme on l'a vu, ces structures s'obtiennent identiquement par l'intermédiaire de relations ensemblistes et topologiques, interprétables à un niveau fonctionnel.

Nous allons expliciter le parallèle proposé.

Donnons-nous une image en niveau de gris définie par la fonctionnelle $f(x, y)$ définie sur le pavage de l'image discrétisée, autrement dit sur un système de voisinage régulier d'où les notions de 4-connexité et 8-connexité découlent automatiquement comme illustré en figure 5.10. Parallèlement, considérons un nuage de point 2D désorganisé contenant un ensemble de quatre points constituant un rectangle englobant du nuage comme illustré en figure 5.11. On va maintenant tenter de donner une interprétation équivalente pour notre image d'intensité et pour notre

image de points géométriques. Dans un premier temps, la triangulation de Delaunay de ce nuage de points définit un système de voisinage. Dans un second temps, on assimilera l'inverse des rayons des cercles circonscrits aux triangles de la triangulation de Delaunay aux valeurs d'intensité des pixels dans l'image en niveaux de gris. Les sites sont donc dans le cas de l'image en niveaux de gris les pixels et dans le cas de l'image de points géométriques, les triangles. Les niveaux des sites sont dans un cas la valeur d'intensité et dans l'autre cas la valeur de l'inverse du rayon de la sphère circonscrite au site géométrique.

FIG. 5.10 – Image en niveaux de gris

La première étape d'un système de traitement d'images pourrait être la binarisation de l'image d'entrée. La figure 5.12 décrit cette étape dans les deux cas pour un seuil pris égal à deux fois la valeur médiane des niveaux des sites. La "binarisation" de l'image géométrique revient à extraire la forme-α où α égale deux fois la médiane des valeurs des niveaux des sites de l'image géométrique.

Une fois obtenues ces images "binarisées", on peut appliquer des opérateurs de filtrage morphologique classiques aux formes "binaires" obtenues. La forme de l'élément structurant utilisé dans le cas classique est explicitée en figure 5.13. Les résultats d'une érosion réalisée sur l'image d'intensité 5.12(a) avec un élément structurant rectangulaire de taille 20 et sur l'image de sites géométriques 5.12(b) est décrit en figure 5.14. De la même façon, on illustre le résultat de l'ouverture en figure 5.15.

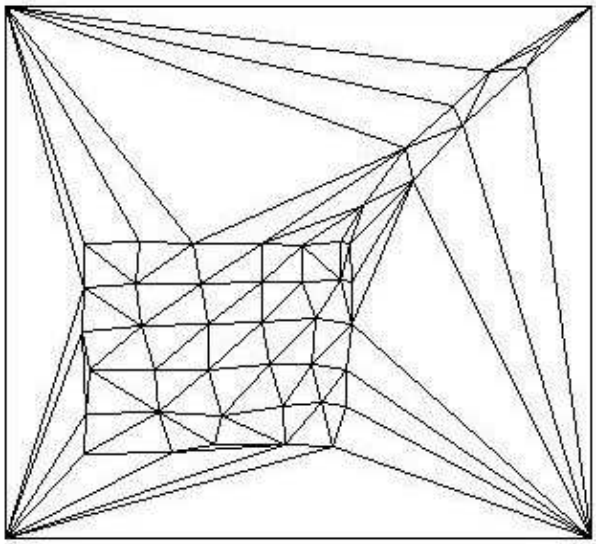

FIG. 5.11 – Image de points géométriques et système de voisinage défini par la triangulation de Delaunay

5.8 Conclusion

Les résultats présentés montrent que les opérateurs conçus pour les triangulations de Delaunay réalisent le même type de transformation que ceux utilisés en morphologie mathématique classique [LGCS00a][LGCS00b]. Même si les définitions sont différentes, ils semblent partager les même propriétés. Par définition, une α-ouverture obéit aux lois de croissance et d'idempotence. Contrairement à la morphologie mathématique classique, nous n'avons pas à notre disposition de trame régulière et les ensembles de points ne sont pas structurés, si bien que la notion de voisinage change. Il n'y a plus de 4-connexité ni de 8-connexité, et ceci oblige à considérer ici le voisinage défini implicitement par la structure de diagramme de Voronoï sous-jacente à la triangulation de Delaunay, c'est-à-dire les triangles de Del en tant qu'éléments structurants. Ceci empêche d'adapter la forme de l'élément structurant alors que c'est un point essentiel en morphologie

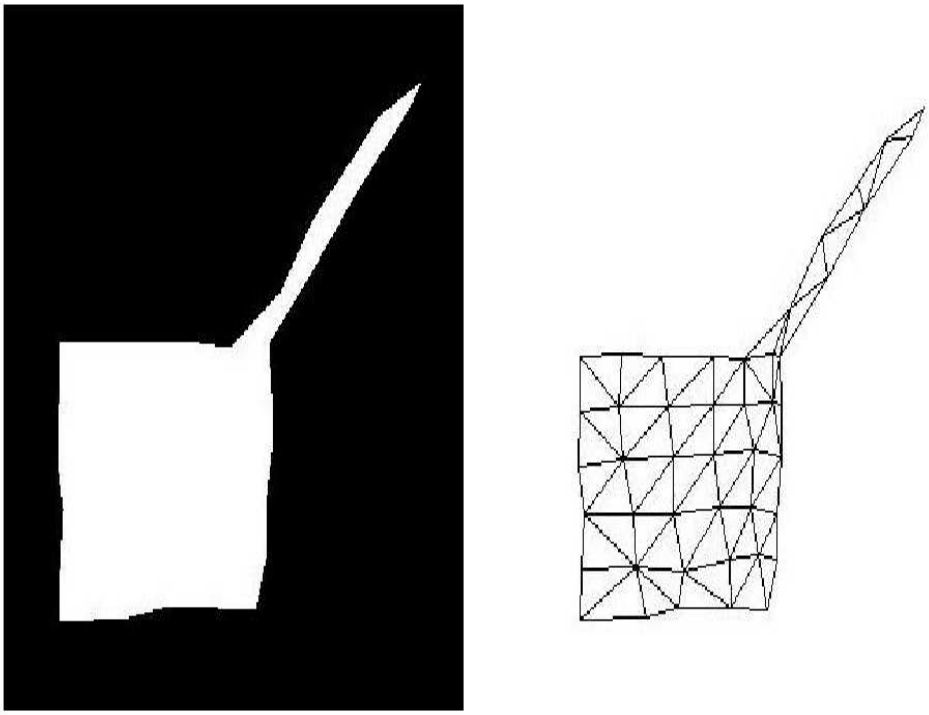

FIG. 5.12 – (a) Image d'intensité binarisée (b) Image de points géométriques "binarisée"

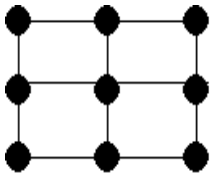

FIG. 5.13 – Elément structurant

mathématique. Nous aimerions maintenant étendre ces opérateurs à des triangulations de Delaunay en dimension trois, et tenter de pousser plus avant la comparaison avec les opérateurs de morphologie mathématique classique, notamment en rendant anisotropes les opérateurs définis.

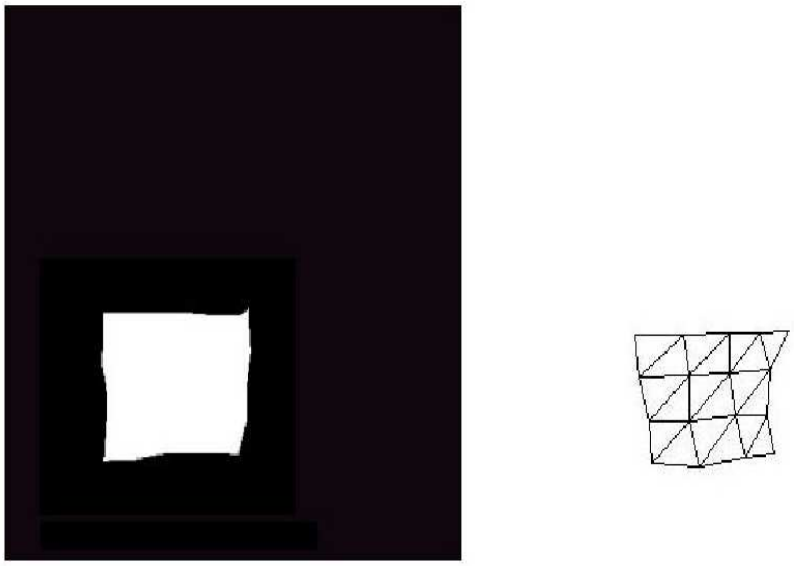

FIG. 5.14 – Erosion appliquée à (a) l'image d'intensité (b) à l'image de points géométriques

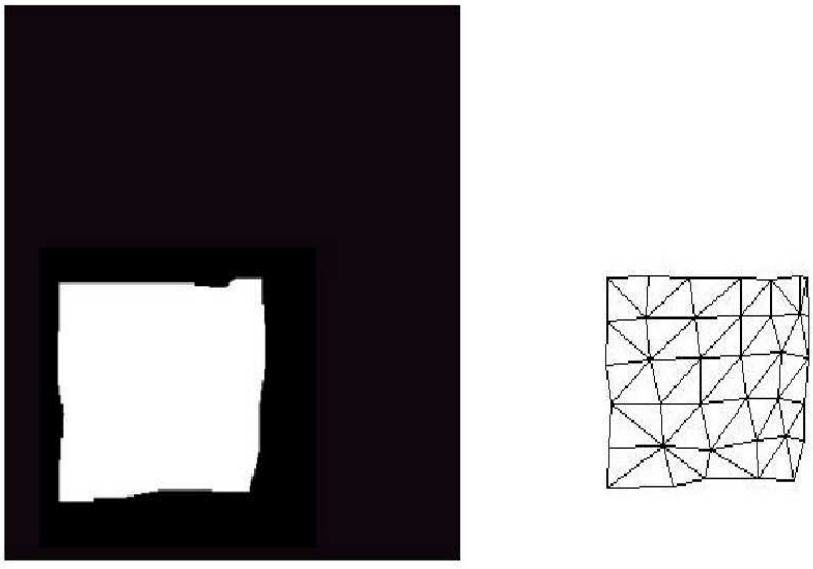

FIG. 5.15 – Ouverture appliquée à (a) à l'image de points géométriques (b) à l'image de points géométriques

Chapitre 6

Stratégie générique d'analyse

Les chapitres précédents ont mis en place les outils qui vont nous permettre d'analyser puis de reconstruire sous forme d'objets isolés des scènes décrites par des nuages de points 3D.

D'une part, la méthode de partitionnement *CMFE* de nuages de points 3D inspirée des nuées dynamiques permet d'extraire dans un même formalisme les structures planaires et les structures de topologie plus sphérique (cf. chapitre 4). D'autre part, les opérateurs morphologiques définis sur des graphes de Voronoï permettent une analyse plus fine des structures segmentées par filtrage de la forme représentée par ces graphes (cf. chapitre 5).

Plus précisément, l'algorithme proposé dans [GG89] s'appuyant sur la formulation des C-Moyennes floues classiques mais utilisant une distance dite "exponentielle" permet de prendre en compte la variabilité :

- de la forme des groupements en cours de formation, ce qui autorise la séparation des groupements de topologie plane des groupements de structures hyperellipsoïdales ;
- de la densité en points des groupements, prenant ainsi en considération une des caractéristiques essentielles des nuages de points 3D obtenus par stéréoscopie, où la densité diminue avec la profondeur de scène ;
- du nombre de points dans chaque groupement, permettant notamment de traiter dans le même formalisme la détection d'une large zone ou d'un petit objet.

Par ailleurs, les opérateurs morphologiques [LGCS00a] agissant sur des maillages quelconques permettent d'affiner l'analyse du nuage de points à segmenter et à

reconstruire. Par filtrage et analyse de l'information de forme, on doit parvenir à améliorer le caractère non supervisé du partitionnement pour atteindre un point d'équilibre idéal en terme de temps de calcul et de niveau de détails désiré.

En fait, il s'agit essentiellement de trouver le bon nombre de classes. Cette problématique essentielle a été abordée par quelques auteurs [GG89] [BJ81] [Bez74] [Bez75] [Gun78] [Win81] [Win82] [LR83] [WBW89] [FS89] [XB91] [PB95] [PB97] [RzJR98] mais sans parvenir à un outil véritablement générique, notamment par emprunt de considérations exclusivement numériques et souvent extrêmement calculatoires [OZ00] [Ba96] [FK99] [Ste95] [DK97]. A présent, nous allons tenter de définir une stratégie globale de reconstruction 3D combinant les bonnes propriétés de ces outils, et permettant notamment de trouver ce nombre optimal de classes.

La figure 6.1 illustre le processus global d'interprétation de nuages de points. Dans un premier temps, nous allons détailler la méthodologie de partitionnement adoptée. Puis, nous donnerons quelques éléments de classification des objets détectés, notamment la façon de définir un obstacle sans la connaissance de modèle. Enfin, la méthode de reconstruction 3D des objets sera expliquée. Nous expliciterons notre stratégie globale d'analyse et de reconstruction de scènes 3D décrites par un nuage de points dans une optique précise de navigation autonome pour un robot. En dernier lieu, une approche dynamique de la stratégie sera proposée.

6.1 Postulat

On partira d'un postulat vérifié expérimentalement sur l'ensemble des scènes qui ont servi à la validation de nos résultats :

Postulat 6.1 *Pour tout nuage de points, il existe une K-partition obtenue par l'algorithme* CMFE *exhibant une structuration en zones et objets, en adéquation avec des objectifs de navigation autonome d'un robot.*

Cette observation est ponctuellement illustrée sur la figure 6.3, où l'on peut voir que toute K-partition pour K variant de 2 à 10 présente une interprétation cohérente de la scène de la figure 6.2. La 2-partition de la scène exhibe un partitionnement en avant-scène et arrière-plan, ce dernier étant défini notamment par une diminution de la densité de points avec la profondeur de scène, caractéristique de l'acquisition stéréoscopique. La 4-partition répartit de façon homogène

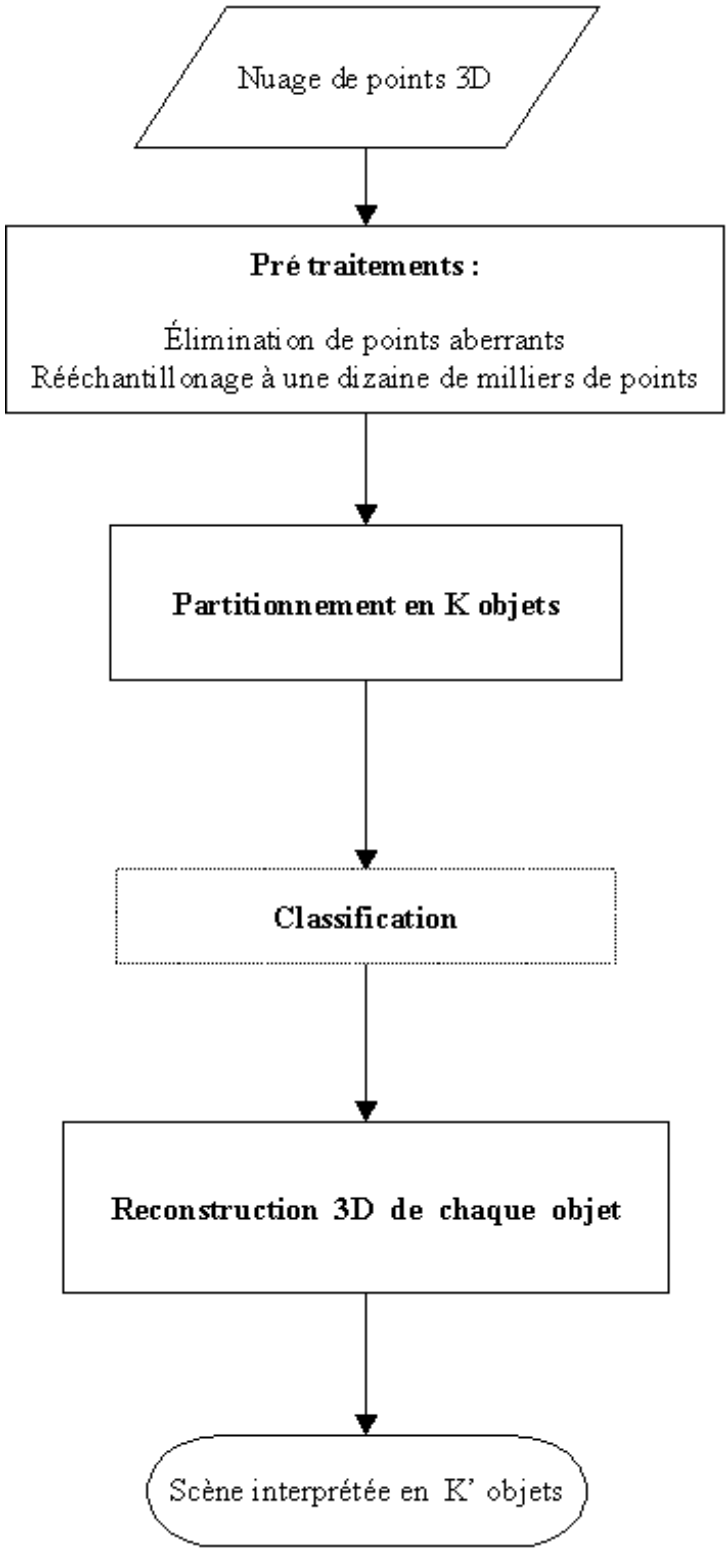

FIG. 6.1 – Organigramme global de la stratégie de reconstruction d'un nuage de points 3D

la scène selon quatre zones cohérentes dont l'une au fond à droite correspondant au groupement rouge rassemble la poubelle renversée et la partie montante du sol. La 5-partition isole pour la première fois la zone de rochers au fond à gauche dans le groupement noir. La 6-partition rééquilibre la scène en mettant dans un groupement noir la zone de rochers et dans un groupement rouge les deux obstacles poubelle et pente de sol déjà rassemblés dans la 4-partition. La 8-partition n'apporte pas de changement particulier. Quant à la 10-partition, elle montre enfin tous les obstacles identifiés dans la scène correctement isolés des zones navigables : la poubelle renversée est représentée par le groupement noir, la zone de sol montante par le groupement jaune et enfin la zone de rochers par le groupement vert clair.

Cette suite de K-partitions illustre le comportement "intelligent" de l'algorithme de partitionnement *CMFE* utilisé, intégrant une distance dite "exponentielle". En effet, quel que soit le nombre de groupements K spécifié en entrée, on obtient une interprétation cohérente et exploitable de la scène en ces K groupements. Ce qui n'aurait pas été le cas avec la distance euclidienne seule.

La 10-partition de la scène présentée à nouveau en figure 6.4 avec le rendu 2D du partitionnement 3D obtenu corrobore le postulat précédent : ce partitionnement, optimal selon notre objectif, exhibe parfaitement l'ensemble des zones et objets de la scène permettant à un robot de distinguer les zones navigables, c'est-à-dire plus ou moins planes et dans la direction du sol, de l'ensemble des zones d'obstacles. Remarquons que les objets du fond de la scène ne sont pas pris en compte dans le nuage 3D reconstruit en raison de la trop forte imprécision de reconstruction au-delà d'une certaine distance au capteur.

6.2 Analyse en regroupement

Nous venons donc d'admettre que l'outil de partitionnement utilisé fournissait, quel que soit le paramètre K, une interprétation d'intérêt de la scène. En revanche, il ne présente pas pour le moment la possibilité d'exhiber la K-partition la plus intéressante en fonction d'un objectif déterminé. Toute la stratégie d'interprétation de scènes va donc à présent reposer sur la détermination de ce paramètre K optimal, c'est-à-dire sur la détermination du nombre optimal de groupements à effectuer sur le nuage de points 3D pour décrire au mieux la scène, dans un objectif d'évitement d'obstacles notamment. La stratégie comporte deux phases :

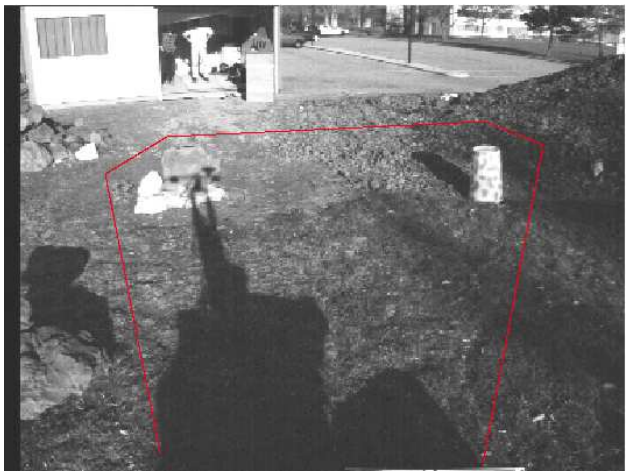

FIG. 6.2 – Image d'une paire stéréoscopique de scène d'extérieur : on y voit apparaître une zone de rochers sur la gauche, une poubelle renversée sur la droite et un peu plus loin au fond une zone de sol montante

1. une phase de partitionnement préliminaire contrôlé par un critère purement numérique ;
2. une phase de partitionnement incrémental contrôlé par une batterie extensible de critères heuristiques géométriques, topologiques, voire morphologiques.

Nous avons abondamment évoqué les mesures numériques de validité de partitions dans le chapitre 4. La plus robuste d'entre elles correspond à la mesure de Densité Moyenne de Partition (*DMP*) introduite dans [GG89]. C'est en conséquence sur cette mesure que nous appuierons la phase de partitionnement préliminaire de la scène en zones globalement structurantes.

Or, l'utilisation seule de cette mesure ne nous garantit pas une interprétation optimale de la scène dans l'objectif de navigation autonome. Elle fournit en général une K-partition cohérente mais insuffisante en vue de l'application spécifique. En réalité, on peut montrer expérimentalement que si l'on parcourt la courbe $DMP(K)$ et que l'on visualise la K-partition obtenue pour chaque maximum relatif de la courbe, on observe à chaque fois un partitionnement significatif de la scène.

Seulement, jusqu'à quel maximum relatif de la courbe $DMP(K)$ faut-il pousser la segmentation pour obtenir le niveau de détail exigé ? La réponse est arbitraire : pour certaines scènes un arrêt au premier maximum relatif est suffisant

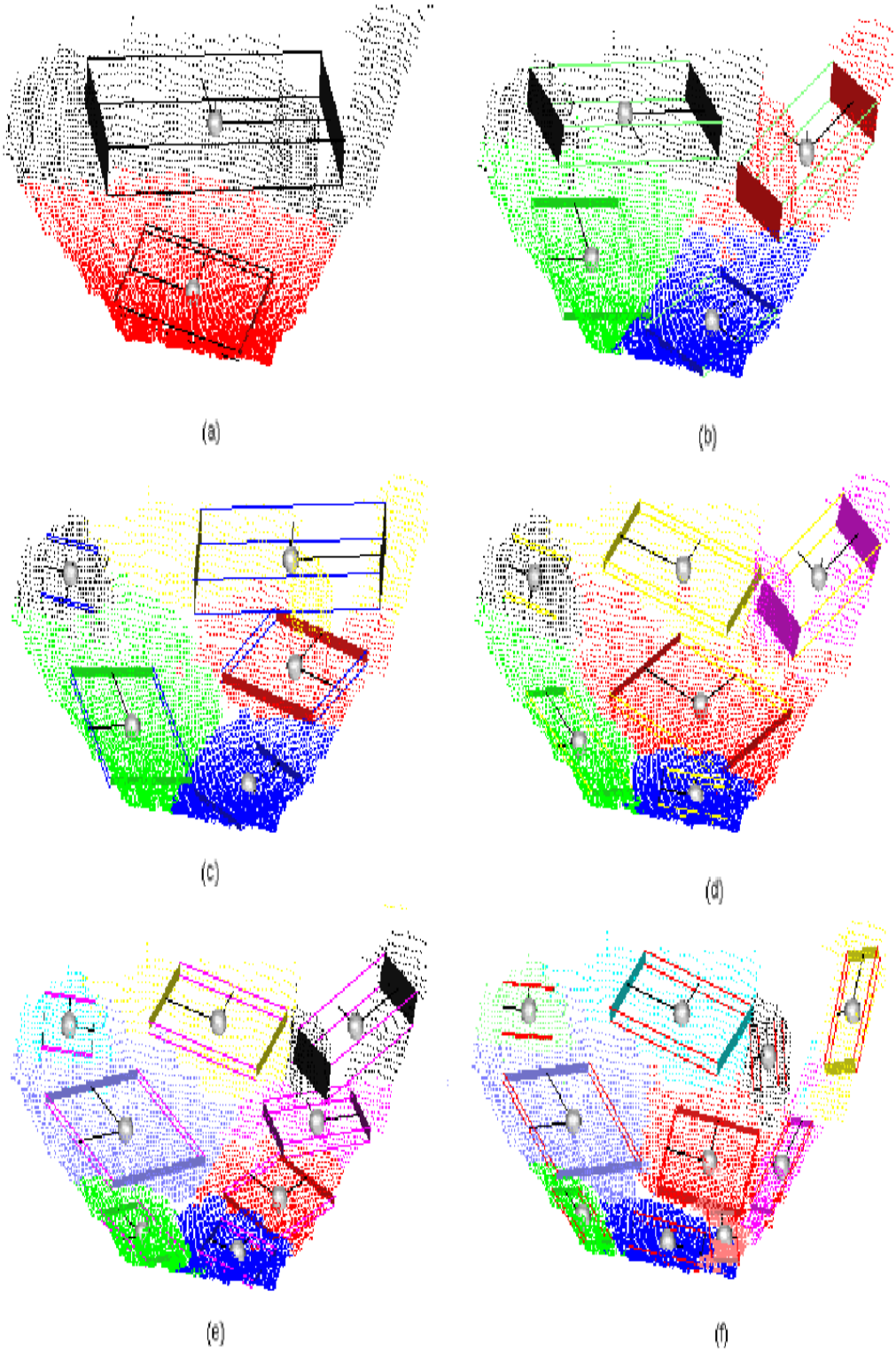

FIG. 6.3 – Différentes K-partitions de la scène précédente à partir du nuage de points 3D reconstruit par stéréoscopie (a) 2-partition (b) 4-partition (c) 5-partition (d) 6-partition (e) 8-partition (f) 10-partition

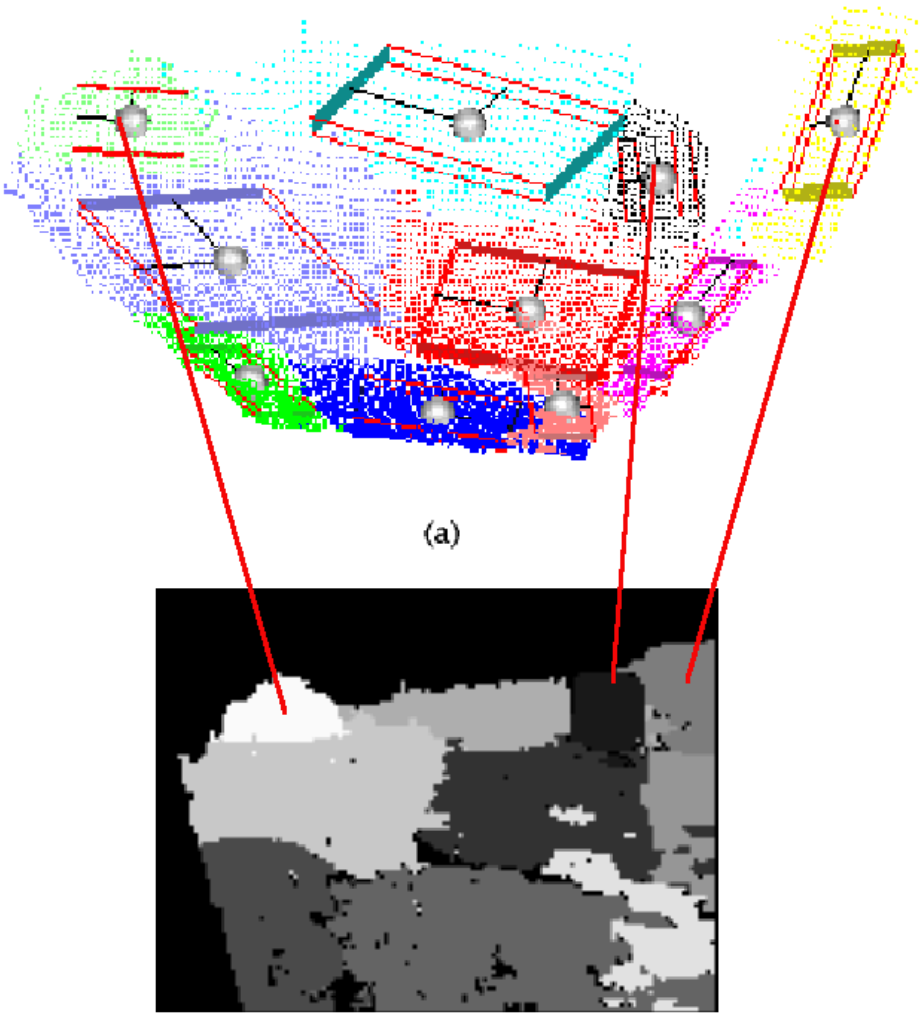

FIG. 6.4 – (a) 10-partition du nuage de points 3D représentant la scène précédente ; elle donne une description de la scène faisant apparaître tous les obstacles séparés : l"amas de rochers sur la gauche, la poubelle renversée sur la droite et un peu plus loin au fond la zone de sol montante (b) segmentation 2D obtenue par rétroprojection des points 3D sur l'image du capteur

alors que pour d'autres plus complexes il aurait fallu continuer jusqu'au troisième maximum relatif. Par ailleurs, les capacités de calcul ne permettent pas forcément d'explorer toutes les K-partitions possibles en parallèles pour choisir finalement la meilleure, d'autant que plus on augmente le nombre K d'objets plus le processus de partitionnement est lent (cf. paragraphe 6.7). Pour toutes ces raisons, pour affiner la détermination du nombre optimal K' de groupements à partir d'une première approximation K correspondant à un maximum relatif de la courbe $DMP(K)$, nous allons utiliser des critères d'ordre heuristique combinant une interprétation géométrique, topologique voire morphologique des formes obtenues pour les groupements segmentés.

Ces critères heuristiques d'affinement du paramètre K s'appuient sur l'utilisation des opérations morphologiques agissant sur des représentations à base de diagrammes de Voronoï définies au chapitre 5.

En effet, et cela constitue un point essentiel, la procédure de segmentation du nuage de points 3D par la méthode *CMFE* génère plusieurs comportements biaisés en fonction des données traitées et de l'application envisagée, comportements qu'il va s'agir de corriger.

Par la suite, on qualifie d'*ambigü* un groupement susceptible d'être à nouveau partitionné pour accéder à une description plus détaillée et plus fidèle de la forme qu'il décrit. En outre, par ce procédé itératif de partitionnement des objets *ambigüs*, on peut considérablement réduire le temps de calcul puisqu'on peut alors se focaliser sur un objet de la scène contenant beaucoup moins de points que la scène entière. Pour les problèmes de complexité numérique, on se reportera au paragraphe 6.7.

6.2.1 Heuristique d'ambiguïté géométrique

Un des comportements biaisés observés pour l'algorithme *CMFE* sur les nuages de points de scène d'extérieur non structurée dont nous disposions est illustré en figure 6.5 ainsi qu'une schématisation 2D en figure 6.6. Les obstacles sont constitués de trois petits rochers répartis à la surface du sol. Les trois rochers sont rassemblés dans un même groupement de couleur noire. En fait, constituant trois entités hyperellipoïdales émergeant d'une zone plane, ils sont plus *proches* selon la *distance exponentielle* entre eux que du sol, plus proches du point de vue de la forme. En clair, la forme de chacun de ces groupements joue un rôle attractif ou

répulsif.

Cette première 4-partition correspond au second maximum relatif de la fonction $DMP(K)$ pour K variant de 1 à 8. On constate que la partition optimale obtenue par application simple d'un critère numérique du type DMP dont on extrairait un maximum d'ordre fixe sur une suite de K-partitions ne fournit pas une description suffisamment *détaillée* de la scène en terme d'obstacles isolés. Le *niveau de détail* n'est pas à la hauteur de nos attentes même si la partition proposée est cohérente et correspond à une réalité descriptive de la scène. Sur la figure 6.8, on montre les trois partitions correspondant aux trois maxima relatifs de la courbes $DMP(K)$ apparaissant sur la figure 6.7. La 2-partition montre une séparation en avant-scène dense et arrière-scène moins dense. La 4-partition montre les trois rochers rassemblés dans un seul groupement. La 7-partition exhibe les deux rochers les plus proches - donc les plus denses - séparés. On voit donc que l'heuristique devra permettre de relancer la segmentation à partir de n'importe quel maximum relatif pour atteindre le niveau de détail exigé par l'application.

En l'occurence, l'analyse de la forme générée par un nuage de points va nous permettre d'élever ce niveau de détail - paramétrisé finalement par la valeur de K - à la hauteur de nos exigences. C'est la morphologie mathématique appliquée aux diagrammes de Voronoï qui devrait nous permettre d'exhiber une heuristique simple de détection des partitionnements insuffisants. En pratique, on a vu dans le chapitre 5 que les α-formes constituent une manière de décrire la forme générée par un nuage de points désorganisé. Par ailleurs, nous avons montré qu'une α-ouverture réalisée sur le diagramme de Voronoï représentant ce nuage de points permet de lisser la forme en question en éliminant ses excroissances et impuretés.

Pour résoudre ce comportement biaisé, nous avons ainsi défini une heuristique appelée *heuristique d'Ambiguïté Géométrique* ou *heuristique hAG* qui consiste, pour chaque groupement détecté, à :

- projeter orthogonalement le nuage de points 3D sur son plan d'approximation principale donné par la direction de la plus petite valeur propre de la matrice de corrélation du nuage comme décrit dans le chapitre 4 ;
- extraire l'α-ouvert optimal du nuage projeté ainsi obtenu ;
- déterminer alors le nombre de composantes connexes significatives, c'est-à-dire de surface suffisante, présentes dans cet α-ouvert ;
- si ce nombre est plus grand que l'unité, on dira que le groupement présente

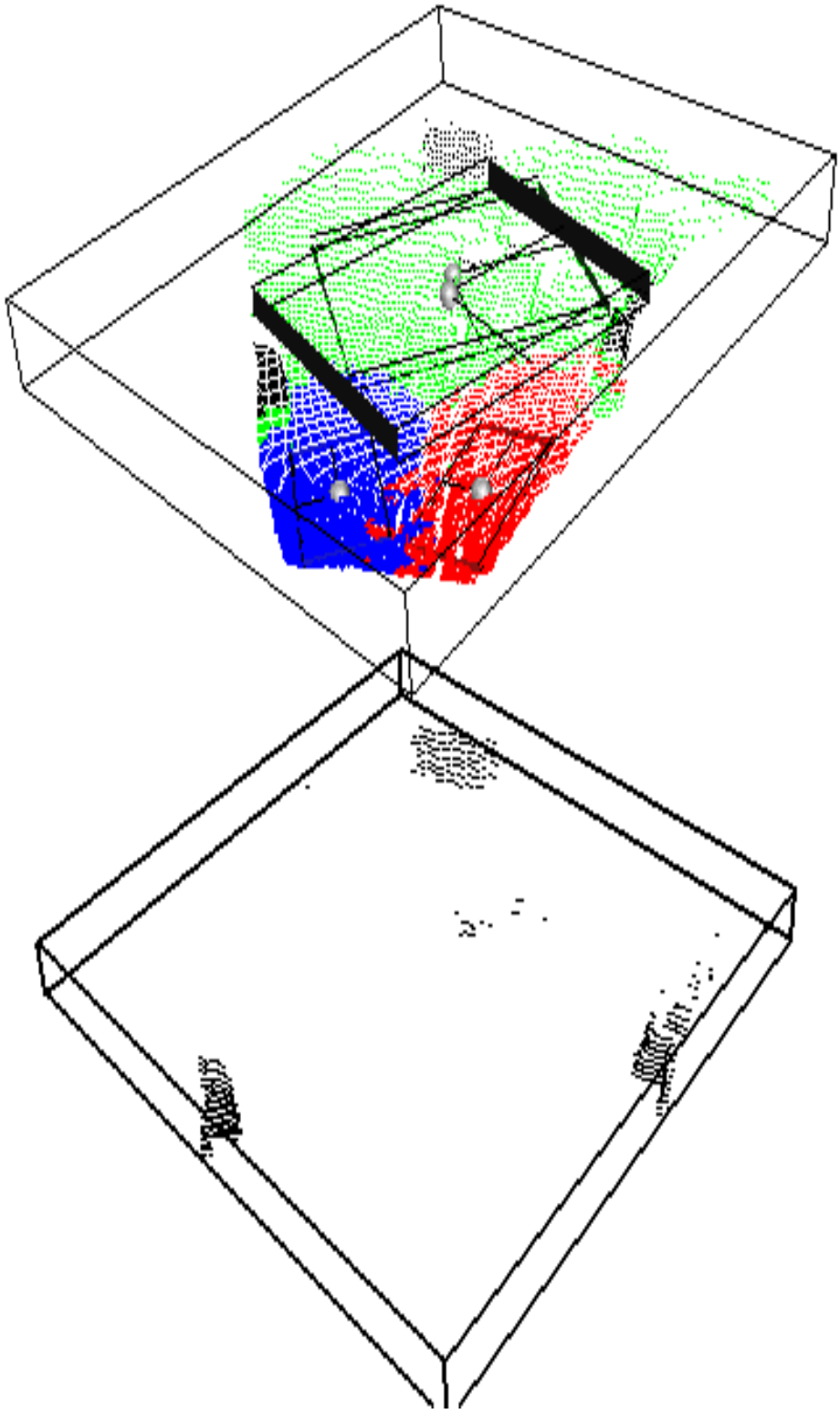

FIG. 6.5 – Au-dessus 4-partition dans laquelle trois rochers de structure ellipsoïdale émergent dans un groupement de couleur noire d'une large zone planaire de couleur verte et sont rassemblés dans un seul groupement ; en-dessous l'objet posant problème isolé

FIG. 6.6 – Schématisation 2D d'un des comportements biaisés de l'algorithme CMFE : plusieurs obstacles émergeant d'une large zone de sol sont rassemblés dans le même groupement

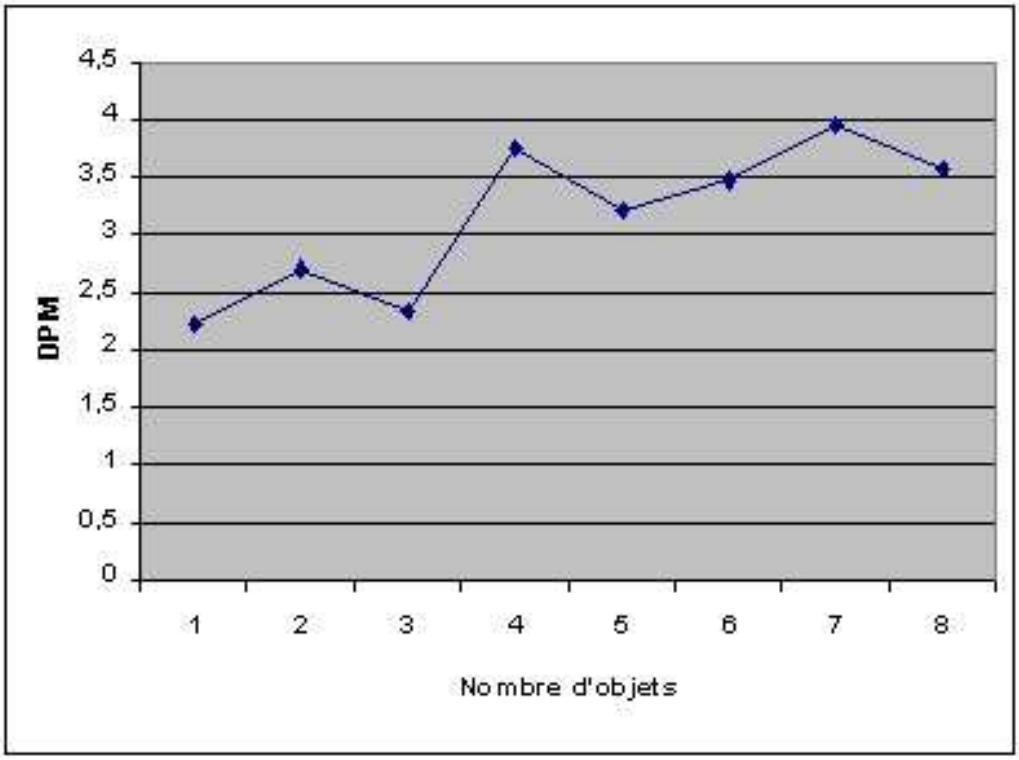

FIG. 6.7 – Courbe décrivant le comportement de la mesure de qualité de partition DMP(K) en fonction de K

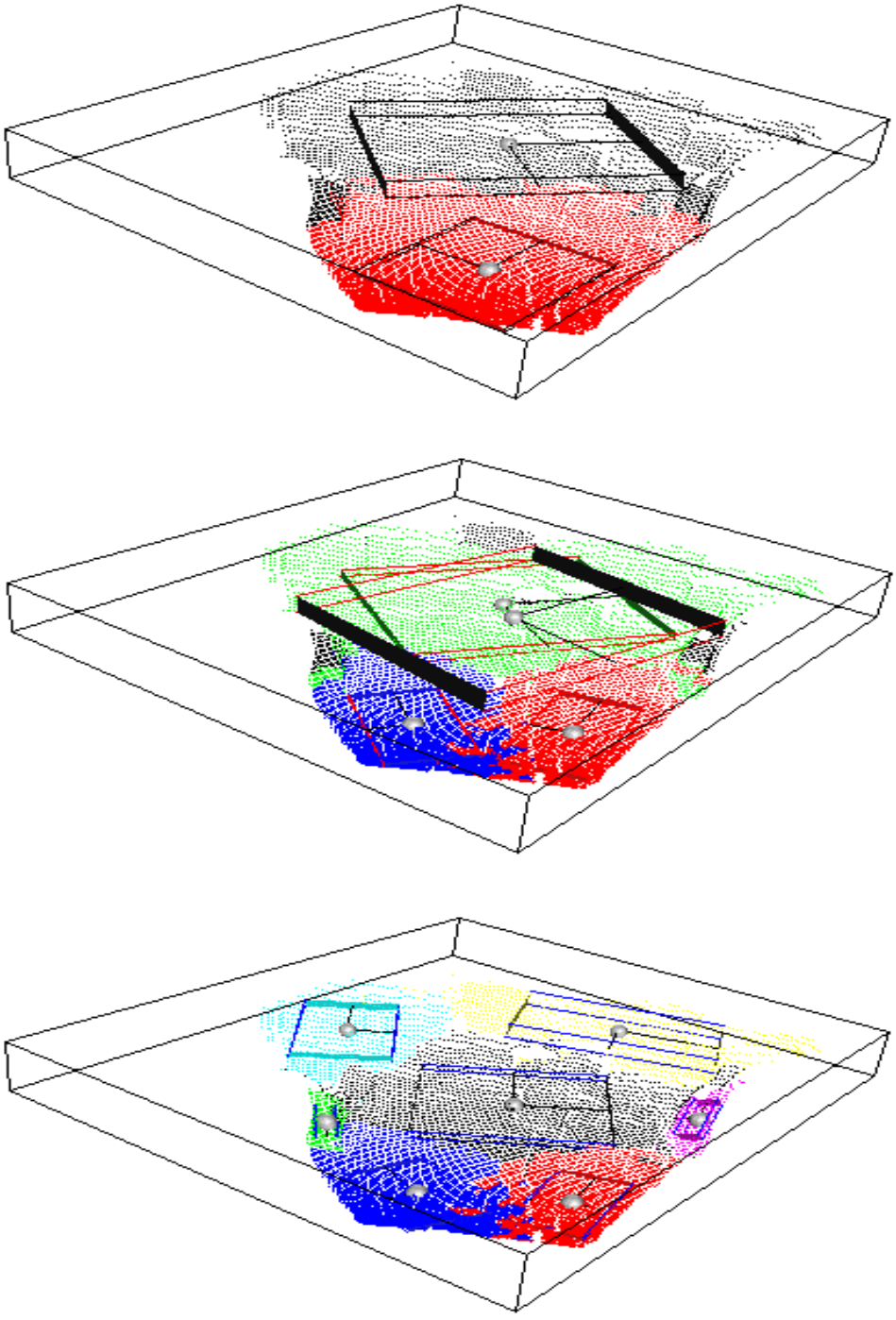

FIG. 6.8 – Différentes partitions correspondant à des maxima relatif de la courbe DMP(K)

une ambiguïté d'ordre géométrique et on décidera de poursuivre le partitionnement en augmentant le niveau de détail ; sinon, on valide la partition obtenue pour ce nombre de groupements K courant.

Ainsi, le groupement de couleur noire présente une ambiguïté d'ordre géométrique faisant apparaître trois composantes connexes de surface voisine comme illustré en figure 6.9. La statégie de reconstruction 3D développée décide donc de poursuivre le partitionnement. Si on applique l'heuristique *hAG* à la 4-partition obtenue en figure 6.5, on obtient alors une 6-partition donnant finalement le partitionnement optimal souhaité comme illustré en figure 6.10. Dans cette partition, tous les obstacles apparaissent correctement segmentés en vue de la reconstruction 3D. On notera que pour relancer le partitionnement, on a effectué un nouveau partitionnement non supervisé centré sur l'objet incriminé.

6.2.2 Heuristique d'ambiguïté géométrique complète

Cette heuristique est une extension de la précédente. Elle réalise la même analyse que précédemment dans les trois directions principales du nuage de points. C'est-à-dire qu'on réalise l'analyse morphologique des projections du nuage selon les trois plans principaux. On notera cette heuristique *hAGC* pour *heuristique d'Ambiguïté Géométrique Complète*. Cette heuristique sera surtout utile dans le cas de nuages peu denses et très complexes sans information de prise de vue comme ceux correspondant à des intérieurs d'usine dans notre base de test (Base 2 du chapitre 7).

6.2.3 Heuristique d'ambiguïté topologique

L'autre comportement biaisé de l'algorithme *CMFE* est illustré en figure 6.11 avec une schématisation 2D en figure 6.12. La scène est constituée d'un seul obstacle situé au-delà de la zone d'avant-sol très dense. Mais l'obstacle, peu dense, est en quelque sorte avalé par la zone de sol très étendue du fond de la scène. L'algorithme de partitionnement utilisé considère l'excroissance formée par l'obstacle sur le groupement plat conmme un gonflement parasite ne justifiant pas la création d'un nouveau groupement. Comme précédemment, cette première segmentation correspond au premier maximum relatif de la fonction $DMP(K)$ pour K variant de 1 à 10. Encore une fois, on voit que cette validation de partition n'offre pas un

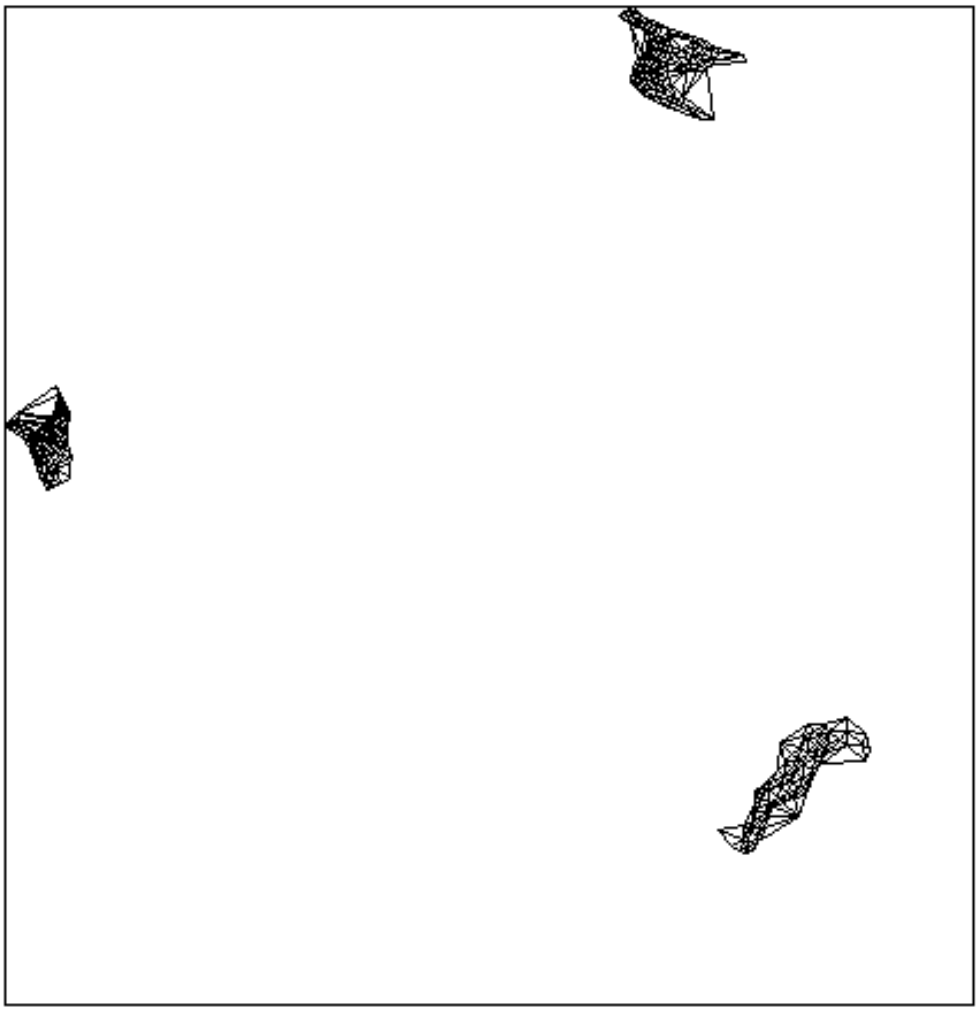

FIG. 6.9 – Ambiguïté géométrique au niveau de l'objet de couleur bleue

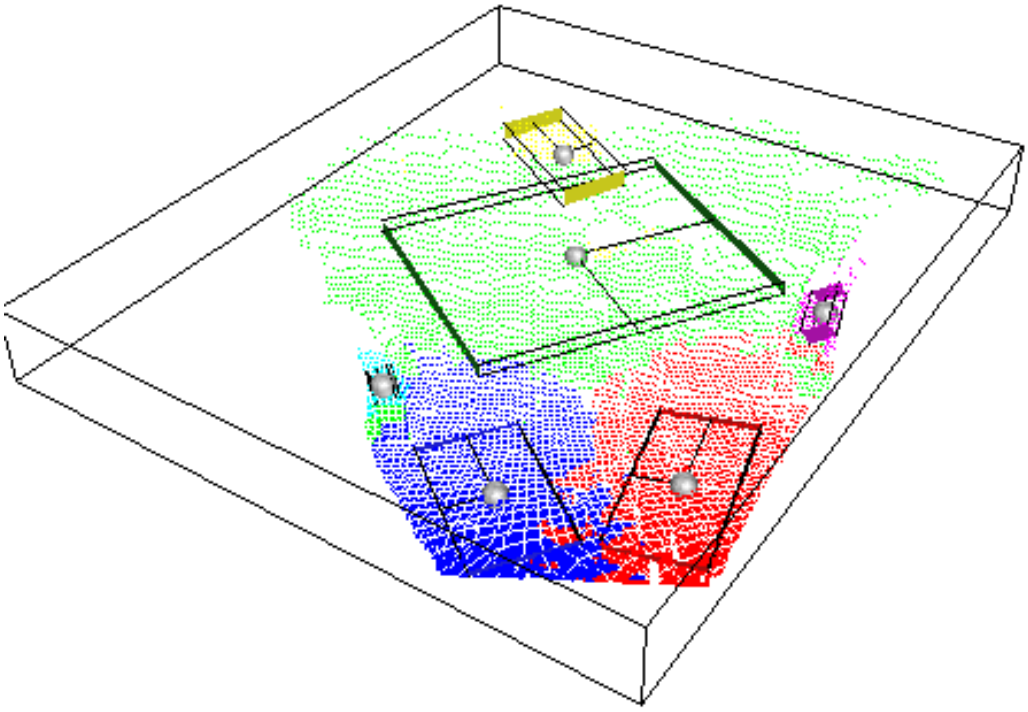

FIG. 6.10 – 6-partition optimale après application de l'heuristique hAG et segmentation incrémentale à partir de la 5-partition précédente

niveau de détail suffisant pour notre objectif d'évitement d'obstacles. En fait, l'algorithme s'est contenté de diviser le nuage de points en une avant-scène très dense et un fond de scène beaucoup moins dense globalement. Notons que si l'obstacle se trouvait plus près du capteur, il serait plus dense et serait certainement correctement segmenté par l'algorithme *CMFE*.

Pour résoudre ce problème, il faut dans un premier temps parvenir à estimer la direction du sol. Pour cela, on récupère la zone plane dense la plus proche de la caméra qui dans la plupart des cas correspond à la direction du sol pour cette prise de vue.

Puis, chaque objet segmenté est jugé à l'aulne de cette direction privilégiée, indiquant l'horizontalité. Parallèlement, on récupère ce qu'on appelle la *direction principale de chaque objet* : pour cela, on garde les deux valeurs propres maximales de la matrice de covariance du nuage de points décrivant l'objet. Le plan formé par ces deux directions correspond à *la direction de l'objet* (cf. paragraphe 6.3).

Un obstacle de topologie sphérique (codé 2 dans nos conventions) est alors dit ambigü si sa direction se confond dans un intervalle fixé avec la direction du sol estimée. On rencontre ce cas assez fréquemment lorsque un obstacle est avalé par une large zone de sol comme illustré en figure 6.13 *(a)(b)*. Cette large zone de sol impose sa direction. Physiquement, cette observation est justifiée par le fait que les obstacles obéissent aux lois de la gravité et se définissent en général par une certaine *perpendicularité au sol*. Sur cette scène, qui contient *a posteriori* un seul obstacle en fond de scène, la zone plane de couleur rouge en avant-plan donne l'horizontale ou la direction du sol et le groupement de couleur jaune (de topologie non plane) est qualifié d'ambigü par application de l'heuristique précedente. En effet, il est de topologie sphérique mais dans la direction du sol. En conséquence, la scène est à nouveau partitionnée, incrémentalement, et on obtient la nouvelle partition de la scène illustrée en figure 6.13*(c)* qui exhibe l'ensemble des éléments d'intérêt de la scène relatif à notre application spécifique.

On désignera cette heuristique par la notation *hAT* pour *heuristique d'Ambiguïté Topologique*. L'usage de cette heuristique implique la détermination de deux seuils : la valeur seuil de λ_{min} - correspondant à la plus petite valeur propre de la matrice de corrélation du nuage (cf. paragraphe 6.3) - au-delà de laquelle un objet n'est plus considéré plan et l'angle β - que fait la direction d'un objet

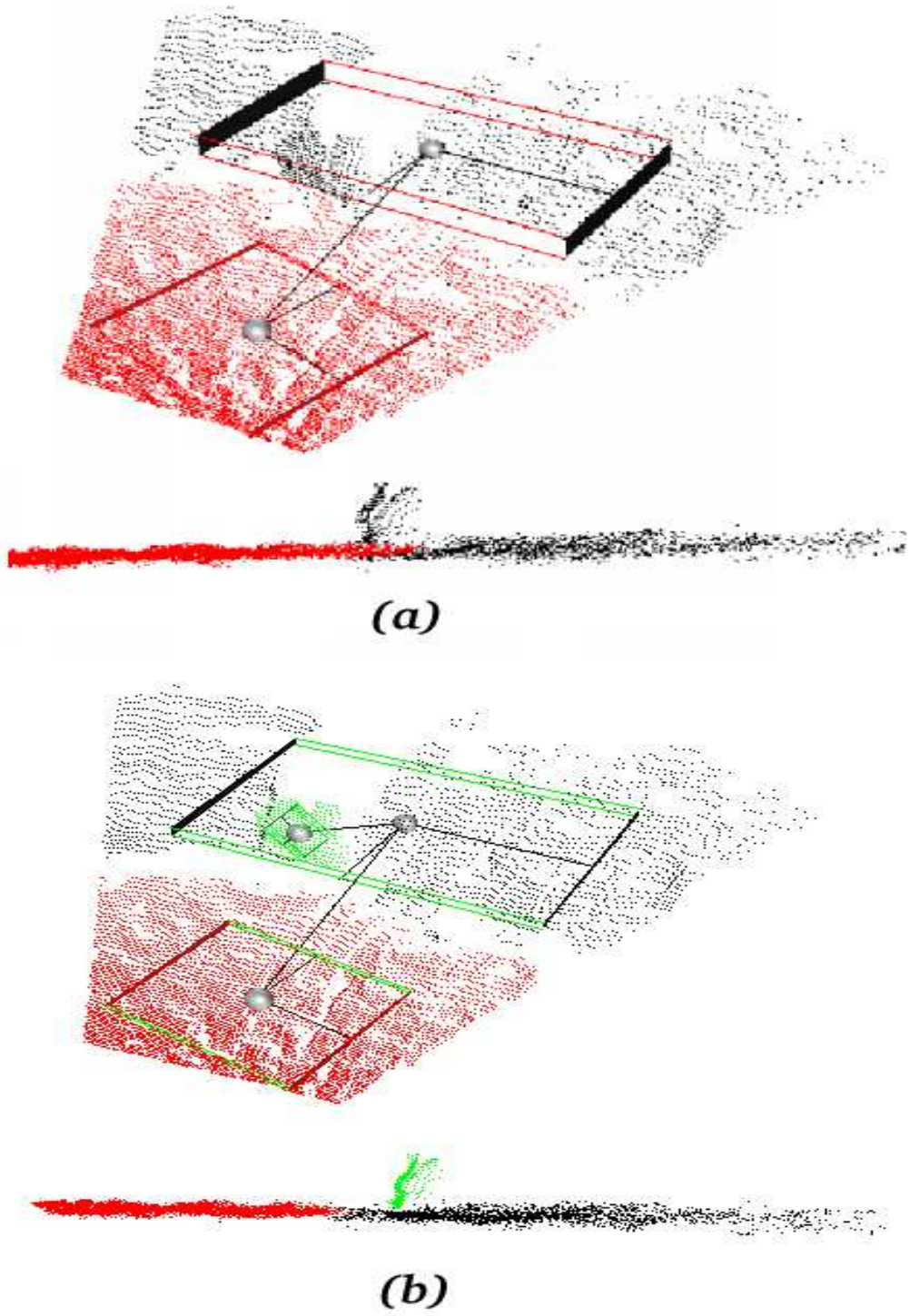

FIG. 6.11 – (a) 2-partition dans laquelle un obstacle est rassemblé malencontreusement avec la zone de sol étendu qui le supporte dans un seul groupement de couleur noire (b) 3-partition donnant une description de la scène faisant apparaître cet obstacle correctement séparé des zones de sol

FIG. 6.12 – Schématisation 2D d'un des comportements biaisés de l'algorithme CMFE : un obstacle petit ou peu dense est associé à une large zone de sol

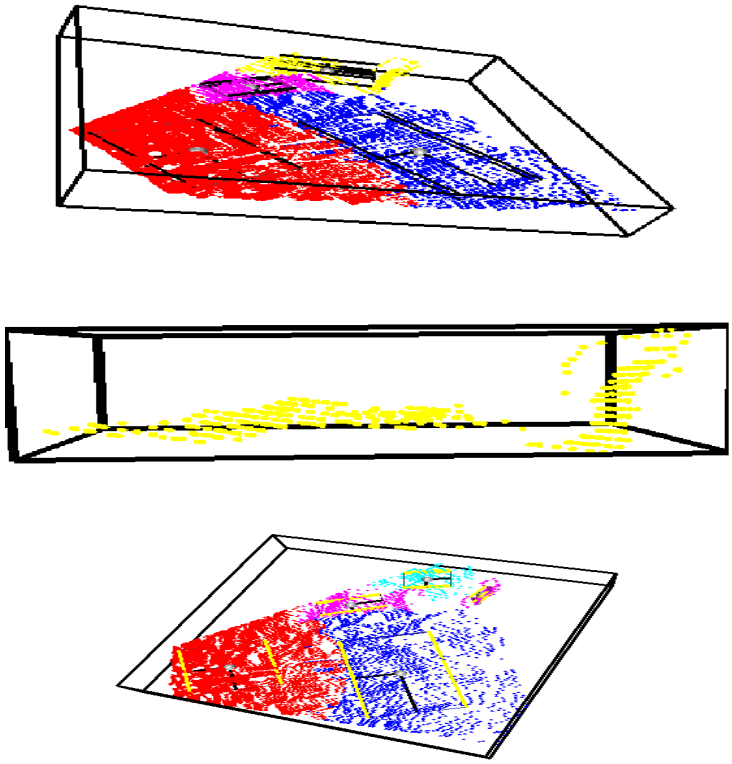

FIG. 6.13 – Illustration du comportement de l'heuristique hAT : (a) la 3-partition préliminaire (b) l'objet de la scène du haut ne respectant pas l'heuristique hAM (c) la 4-partition obtenue après application de la stratégie

avec la direction du sol - au-delà duquel un objet n'est plus considéré localement horizontal.

6.2.4 Heuristique d'ambiguïté planaire

Dans un monde plan, on peut avoir besoin d'une heuristique permettant de tester la planarité d'un groupement pour éventuellement le partitionner à nouveau s'il ne satisfait pas cette propriété. Cette heuristique est notée *hAP* pour *heuristique d'Ambiguïté de Planarité*. Elle fait intervenir le même seuil λ_{min} précédent.

6.2.5 Récapitulatif des heuristiques utilisées

En fonction de l'application, on choisira une ou plusieurs heuristiques et on déterminera les seuils optimaux à utiliser. On récapitule dans le tableau ci-après l'ensemble des heuristiques proposées jusqu'à présent en précisant à chaque fois les seuils à fixer. Dans la pratique, ces seuils sont très aisés à déterminer en fonction des données à traiter notamment lorsqu'on a une information métrique réelle, c'est-à-dire les dimensions réelles des objets traités.

- hAT : seuil sur λ_{min} et l'angle β en-decà duquel la direction de l'objet est assimilée à celle du sol ;
- hAG et hAGC : seuil sur α (facilement automatisable) pour la détermination de l'α-ouvert optimal et la taille Σ significative des composantes connexes détectées ;
- hAP : seuil sur λ_{min} ;

6.3 Quelques éléments de classification des objets

Même si on évolue dans un monde sans modèle, pour obtenir un résultat interprétable dans notre application de navigation autonome, on doit être capable de décider si un objet constitue un obstacle ou une zone plane navigable horizontale. Pour cela, on est obligé de définir de façon la plus basique possible la notion d'obstacle et de zones de sol.

En ce qui concerne les zones navigables, il est utile de récupérer quand cela est possible la direction du sol. Dans notre application de navigation robotique en milieu inconnu, on considère que la direction du sol est donnée par l'objet plan le

plus proche de la caméra et contenant suffisamment de points puisque l'algorithme détecte presque toujours la partie d'avant-sol comme un objet séparé très dense et plat. Une fois cette direction $\vec{sol}$ évaluée (qui correspond en réalité au vecteur normal au plan du sol), on peut classifier les objets segmentés en trois catégories. Une stratégie plus robuste d'estimation du sol serait évidemment nécessaire en navigation réelle, mais celle-ci fonctionne si les images sont bien conditionnées, c'est-à-dire si le robot ne se trouve pas au pied d'un obstacle.

En ce qui concerne les obstacles, on décide de définir un obstacle comme un objet dont la direction principale (c'est-à-dire le plan perpendiculaire à la direction correspondant à la plus petite valeur propre du nuage de points λ_{min}) fait avec la direction du sol estimée un angle inférieur à un seuil β. On voit ici que ces choix peuvent s'avérer relativement arbitraires en navigation réelle. En pratique, ils ont permis d'obtenir une bonne interprétation sur l'ensemble de nos images tests qui constitue une base d'une centaine d'images.

Ainsi, à chaque objet segmenté, on attribue un code devant décrire la nature de l'objet à reconstruire :

- Si la plus petite valeur propre λ_{min} correspondant à son nuage descriptif est *inférieure* à un seuil σ(fixé à $5cm$ dans notre application), l'objet est considéré plan ; l'affectation des labels se fait alors comme suit :

  Code 0 : si $\vec{v_{min}}$ le vecteur propre correspondant à λ_{min} fait un angle inférieur à β (fixé à $\frac{\pi}{8}$ par exemple) avec $\vec{sol}$, l'objet est validé comme zone navigable en tant que plan dans la direction du sol ;

  Code 1 : sinon, l'objet est validé comme un obstacle-plan dans une direction oblique au plan du sol estimé ;

- Sinon, l'objet est considéré non plan ; l'affectation des labels se fait alors comme suit :

  Code 2 : l'objet est validé comme un obstacle de topologie hyperellipsoïdale ;

En fonction de la finesse de ces seuils, on obtiendra des cartes d'obstacles plus ou moins complètes, plus ou moins sensibles aux petits obstacles, et formées d'objets plus ou moins plans. Une des extensions possibles du système pourrait être l'ajout d'un véritable module de classification par reconnaissances des formes. C'est la raison pour laquelle la boîte correspondant à la classification dans l'or-

ganigramme global est en pointillée : le module présent tente de répondre à une question complexe de classification obstacles-zones navigables sans aucun modèle.

6.4 Reconstruction 3D

A présent, il s'agit d'obtenir une structure géométrique 3D de représentation des objets et notamment des obstacles. A cette fin, nous allons mettre à profit le filtrage de formes réalisé par les opérateurs morphologiques définis dans le chapitre 5. Ainsi, si l'objet est validé comme un obstacle, on procède à sa reconstruction à l'aide des α-formes et des filtrages de formes permettant d'épurer la reconstruction 3D de ces obstacles par le biais de son α-ouvert optimal.

Les techniques de maillage purement 3D sont encore victimes de nombreuses difficultés. Par exemple, lorsque l'on passe d'un algorithme de triangulation de Delaunay conçu dans le cas 2D au même algorihtme dans le cas 3D, on soulève de nouvelles difficultés mathématiques et algorithmiques. La procédure d'échange d'arêtes (cf. annexe A) maîtrisée en 2D n'est plus aussi robuste dans le cadre d'échange de faces dans une procédure 3D. Par ailleurs, le temps de calcul s'accroit évidemment de façon importante. Pour ces raisons de robustesse essentiellement, nous avons décidé de travailler en 2D par projection de vue. Une autre justification à ce choix repose sur le fait que nous ne cherchons pas à reconstruire très précisément les objets rencontrés comme des scanners 3D chargés de reproduire un produit manufacturé en 3D. Nous désirons simplement avoir une bonne approximation de la forme réelle de l'obstacle 3D et de son occupation de l'espace. Ce qui nous intéresse, c'est que le robot soit capable de passer sous un arbre. Si on utilise seulement la boîte englobante comme indication d'une zone d'obstacle, tout l'espace contenu sous le feuillage au niveau du tronc est interdit. Si l'on récupère une description plus fidèle de la forme de l'arbre dans l'espace 3D, on libère l'espace ainsi interdit à la navigation. Par ailleurs, pour d'éventuelles applications de reconnaissance d'obstacle, cette information de forme est évidemment d'une importance capitale.

Ainsi, on commence par récupérer la projection des points du nuage 3D sur un plan 2D. A partir de cette projection 2D, on peut reconstruire l'α-ouvert optimal du nuage projeté avant de rétroprojeter le maillage ainsi obtenu dans l'espace

3D. Le plan de projection n'est pas quelconque et dépendra des connaissances de prises de vue. Si l'on connaît l'axe de visée, le plan de reconstruction peut être le plan image. Sinon, il suffit d'estimer le plan principal du nuage de points 3D à partir de sa matrice de covariance et de projeter orthogonalement le nuage sur ce plan comme illustré sur l'obstacle de la figure 6.14. Sur cette figure, le maillage 2D colorié en gris clair correspond à l'α-ouvert de la silhouette de l'objet utilisé pour la reconstruction 3D. Notons enfin que ce choix d'utilisation de la *silhouette 2D* obtenue par projection est justifiée par le fait que l'on possède à un instant t un seul point de vue de l'objet à reconstruire et pas une connaissance de l'objet 3D en entier. Cette limite de représentation 3D sera abordée plus en détail dans le cadre dynamique .

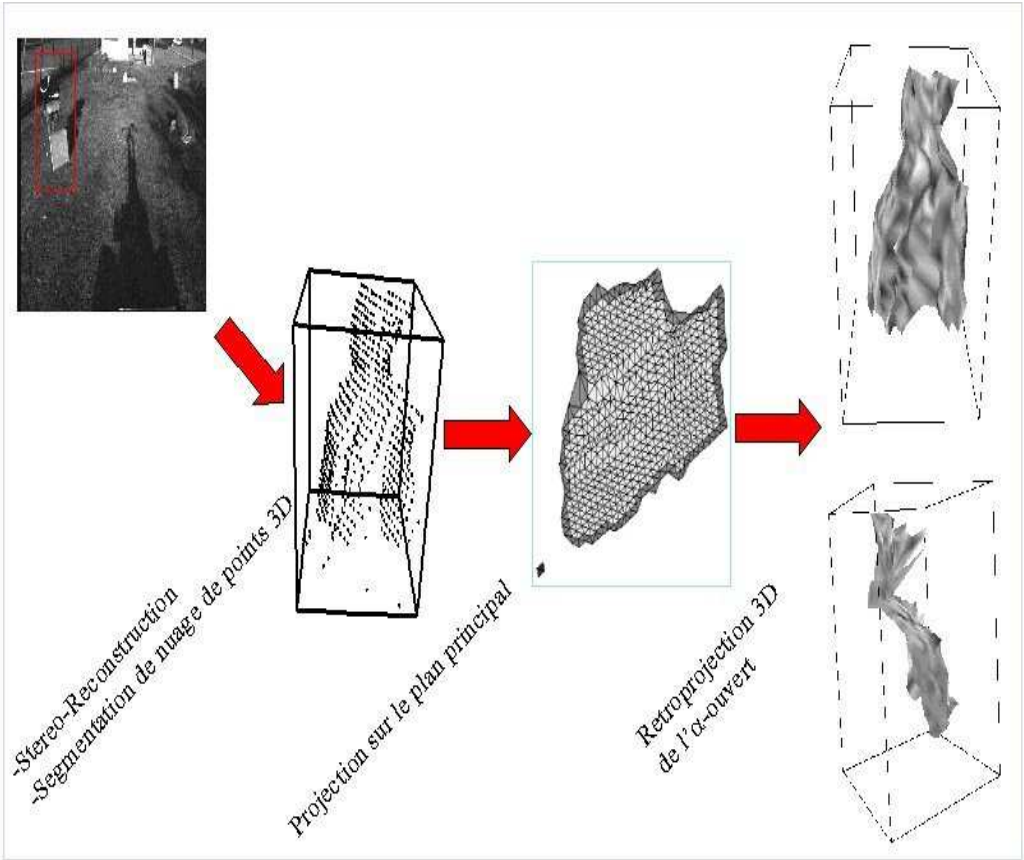

FIG. 6.14 – Processus de reconstruction sous forme d'un maillage 3D d'un nuage de points correspondant à un obstacle isolé

6.5 Organigramme de la stratégie de reconstruction 3D en mode statique

6.5.1 Analyse de la scène en objets

Nous avons donc observé deux types de comportements biaisés lors de nos expériences de partionnement des nuages de points représentant des scènes d'extérieur non structurées en utilisant le seul critère $DMP(K)$. Pour les corriger, nous avons défini les deux heuristiques *hAG* et *hAT* qui permettent de relancer la segmentation du nuage de points en incrémentant le nombre K de groupements jugé nécessaire à une *bonne* représentation de la scène reconstruite en 3 dimensions, selon les objectifs d'une application spécifique.

Pour des questions à la fois de robustesse et de temps de calcul, si nous avons obtenu une K-partition nécessitant d'être poursuivie par application d'une des deux heuristiques *hAG* et/ou *hAT* en une $K + p$-partition, on ne part pas d'une initialisation aléatoire. Mais, on utilise la C-partition floue M_{fkn} précédemment obtenue, et on ajoute p centroïdes en guise d'initialisation de la $K + p$-partition à effectuer.

Le nuage de points 3D est dans un premier temps prétraité. Il est d'une part débarrassé des points abérants pour le réduire à un cube de données mimimal et d'autre part on le rééchantillonne selon un facteur d pour avoir un nuage d'une dizaine de milliers de points. Cette restriction permet de réduire le temps de calcul et n'entame pas la représentativité du nuage de points dans le mesure où l'on n'a pas affaire à des objets manufacturés de grande précision. Enfin, la précision stéréoscopique décroissant en $\frac{1}{Z^2}$ avec la distance Z à la caméra, on ne retiendra que les points 3D de la scène situés à une distance inférieure à un seuil D fixé en fonction des paramètres de caméra (sur le banc stéréoscopique du robot LAMA, à $7m$ de distance, on a une erreur en reconstruction de l'ordre du décimètre).

Puis, un premier partitionnement non supervisé dit *préliminaire* permet de dégager les zones d'intérêt principales de la scène. Cette étape utilise l'algorithme de partitionnement *CMFE* et permet d'obtenir une première description de la scène en K objets. Celle-ci correspond au premier maximum relatif de la courbe $DMP(k)$. A partir de là, on applique la batterie d'heuristiques conçue pour améliorer la description de la scène en fonction des objectifs de l'application

et du comportement observé de l'algorithme de partitionnement sur le type de données en entrée. Nous avons en l'occurence défini deux heuristiques pour notre application principale robotique. Tant qu'une de ces heuristiques n'est pas respectée, on relance le partitionnement incrémentalement à partir du partitionnement précédent. La stratégie présente alors une alternative :

- soit on repart du partitionnement précédent, on initialise avec l'ensemble des groupements trouvés et les données associées, puis on ajoute un centroïde *aléatoirement* avant de relancer l'algorithme pour déterminer la meilleure (K+1) - partition ; dans ce cas, on est sûr de respecter au mieux les frontières entre objets puisqu'on laisse la possibilité à des points classés dans un groupement non ambigü de changer de groupement, mais ceci au prix d'un temps de calcul plus élevé ;
- soit on conserve la partition précédente en l'état, on extrait l'objet incriminé et on relance l'algorithme de partitionnement sur ce seul objet ; dans ce cas, on risque de respecter moins bien les frontières des groupements, mais on accélère considérablement le temps de calcul. A partir de là, une nouvelle alternative est possible :
  - on choisit de relancer une segmentation non supervisée sur ce nuage de points isolé ;
  - on contraint le partitionnement sur un intervalle d'objets : par exemple, on contraint à une 2-partition pour éviter l'inflation de petits groupements qu'il s'agirait ensuite de rassembler, ou bien on contraint à ne pas dépasser le nombre de composantes connexes fournies par l'heuristique hAG par exemple ;

Qu'on ne se méprenne pas, la stratégie est unifiée. On a seulement imaginé quelques variantes pour optimiser le résultat en fonction des données à traiter sans modifier le schéma global. Ces modifications sont extêmement modulaires. Cette stratégie est résumée dans les organigrammes 6.15 et 6.16. A la fin de la stratégie de partitionnement, on récupère une description de la scène en K' objets représentatifs des obstacles et des zones navigables dans la scène. Cependant, même si nous avons rapporté l'explication de cette stratégie à notre problème concret de navigation d'un robot en environnement inconnu, cette démarche s'avère générique notamment par l'extensibilité de la batterie d'heuristiques. Cette propriété sera vérifiée expérimentalement dans le chapitre suivant.

6.5.2 Reconstruction des objets

Une fois un obstacle détecté (codé 1 ou 2 dans notre formalisme), il s'agit de le reconstruire dans l'espace 3D pour préserver au mieux sa forme et éventuellement pouvoir le reconnaître par la suite. De plus, pour la navigation, la mise à disposition d'une simple boîte englobante ou de la forme réelle de l'obstacle ne représente pas à l'évidence la même qualité d'information à des fins d'interprétation. Par exemple, s'il s'agit d'un arbre, il est utile de savoir que l'objet est constitué vers le bas d'une partie plus fine correspondant au tronc et à côté duquel le robot peut passer.

En bref, dans le cas statique, nous disposons d'une seule vue de l'objet. En récupérant la projection des points selon un plan, on reconstruit l'α-ouvert optimal du nuage à partir de cette *silhouette 2D* projetée avant de rétroprojeter le maillage ainsi obtenu dans l'espace 3D. Si l'on connaît l'axe de visée, le plan de reconstruction peut être le plan image. Sinon, il suffit d'estimer le plan principal du nuage de points 3D à partir de sa matrice de covariance et de projeter orthogonalement le nuage sur ce plan comme illustré sur l'obstacle de la figure 6.14.

6.5.3 Organigramme

Le processus est résumé dans l'organigramme de la figure 6.15.

Une variante au niveau du "re-partitionnement" permet d'obtenir des résultats plus rapidement en relançant pour chaque objet ne vérifiant pas une des heuristiques un nouveau partitionnement non supervisé comme illustré sur l'organigramme de la figure 6.16. Par contre, en se restreignant à une sous-partie du nuage de départ, on peut craindre de créer des détails (sur-segmentation) qui ne seraient pas apparus au niveau de structure initial de la scène entière. On peut également craindre de moins bien définir les frontières puisqu'on retire une partie des points originels et donc des objets originels vers lesquels auraient pu migrer certains points des nouveaux objets segmentés (selon le principe des C-moyennes floues). En fait, retirer un objet risque de détruire la structure globale du nuage, voire la structure d'un autre groupement. En pratique, cet effet n'est ni prépondérant ni important et peut donc être ignoré en première approximation, mais il subsiste.

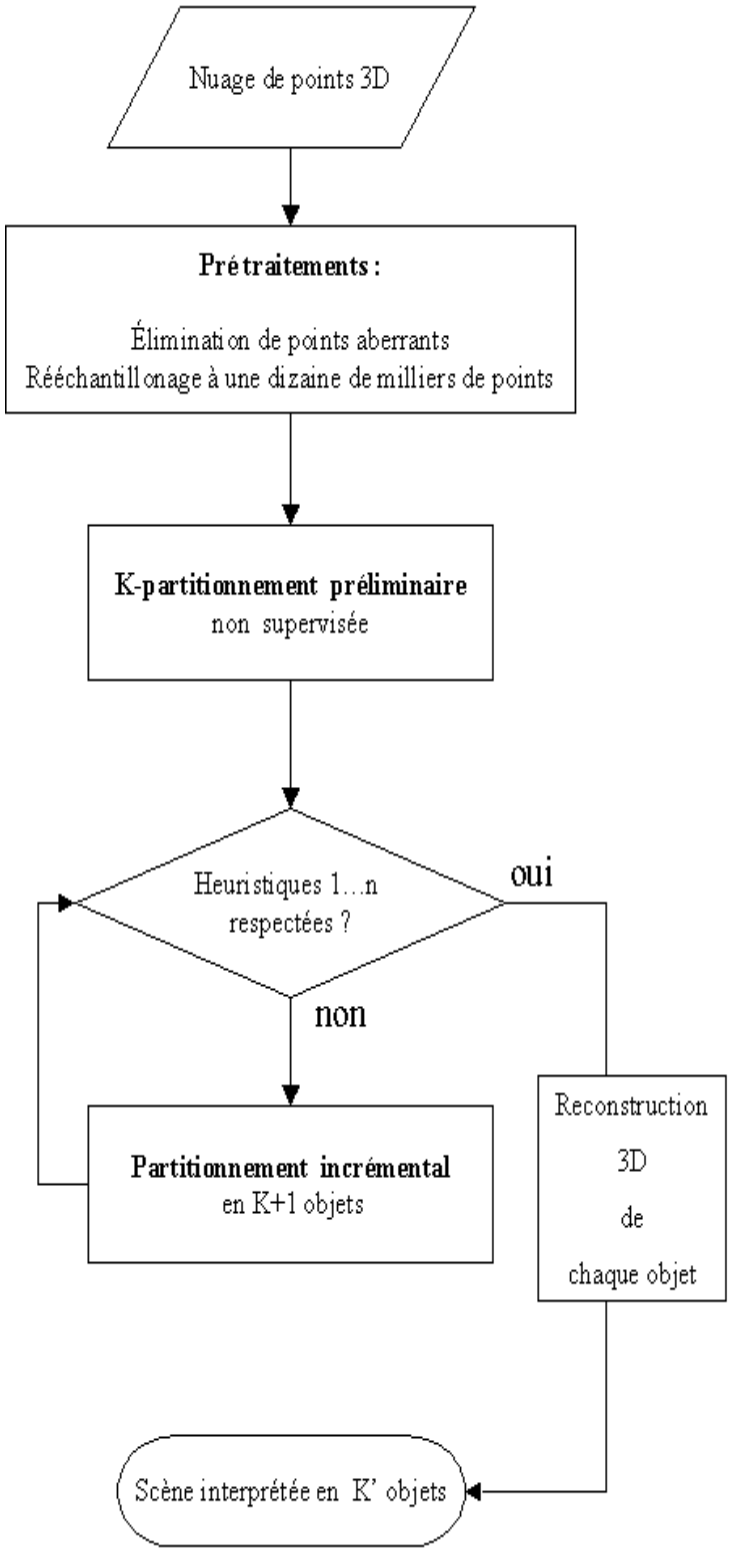

FIG. 6.15 – Organigramme de la stratégie de reconstruction d'un nuage de points 3D en mode statique : schéma lent

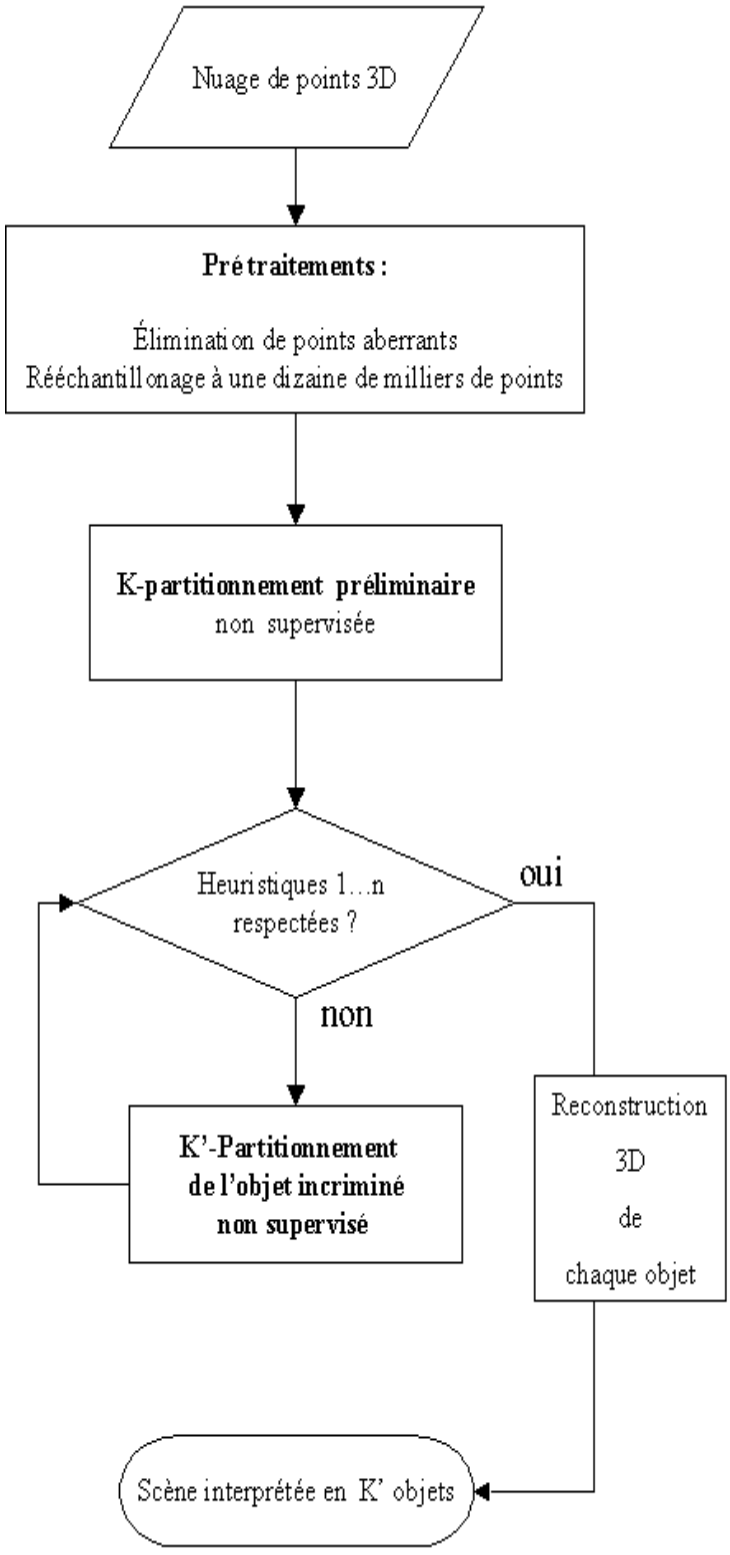

FIG. 6.16 – Organigramme d'une variante de la stratégie de reconstruction d'un nuage de points 3D en mode statique : schéma rapide

6.6 Organigramme de la stratégie de reconstruction 3D en mode dynamique

6.6.1 Reconstruction de l'environnement

En mode dynamique ou exploratoire, c'est-à-dire quand on cherche à reconstruire un modèle d'environnement mis à jour à chaque nouvelle prise de vue quand le robot avance, on doit être capable de gérer deux situations nouvelles :

1. l'utilisation d'informations redondantes provenant d'une analyse précédente, notamment en guise d'initialisation ;
2. la mise à jour du modèle 3D des objets rencontrés, soulevant tous les problèmes de fusion de données ;

En clair, on doit s'appuyer sur l'information de partition à un temps t précédent la prise de vue courante. Typiquement, dans un premier temps le robot avance de 1 mètre à chaque fois dans la même direction avant d'acquérir une nouvelle scène. Disposant d'un positionnement par GPS par exemple, on peut alors facilement voir que les deux scènes se recouvrent et en même temps éliminer les objets dont le centroïde sort du champ de vision du robot. A partir de cette mise à jour des centroïdes présents dans la scène, on peut relancer la segmentation sur le nouveau nuage de points 3D en partant des objets précédents. Cette heuristique de mise à jour appelée *hMAJ* permet d'utiliser une information redondante de scène en scène pour rendre la reconstruction de l'environnement plus robuste et plus rapide. Il faut noter cependant que cette heuristique, si elle permet d'accélérer le temps de traitement, déforme parfois les frontières des objets à détecter. En effet, en récupérant les informations de forme et de partitionnement d'une segmentation correspondant à un nuage de points différent, on impose une structure qui peut aller à l'encontre de celle que l'algorithme aurait découvert si on l'avait initialisé normalement. Donc, pour être plus précis sur la frontière des objets, on déconseillera l'utilisation de cette heuristique si les contraintes de temps de calcul l'autorisent. En fait, théoriquement l'initialisation devrait orienter le résultat final puisqu'on converge en théorie vers un extremum local ou un point de rebroussement de la fonction objective. Or, en pratique, on s'aperçoit qu'une initialisation normale par un algorithme classique de regroupement flou utilisant la distance euclidienne comme effectuée dans l'algorithme *CMFE* (cf. annexe B) est *a priori*

suffisante et stable.

Dans la foulée, le robot dispose d'une vue panoramique. Dans la plupart des séries acquises par le robot LAMA , le système d'acquisition scrute l'environnement autour de lui selon des pas rotatoires prédéfinis. De façon générique, pour chaque scène on peut estimer la zone trapèzoïdale décrite par le champ de vision *dans un repère global*, la comparer à celle des scènes précédentes avant de garder une scène qui intersecte notablement le champ de vision courant et relancer la segmentation à partir de l'initialisation précédente en ayant toujours pris soin d'éliminer les objets sortant du nouveau champ de vision (cf. figure 6.17).

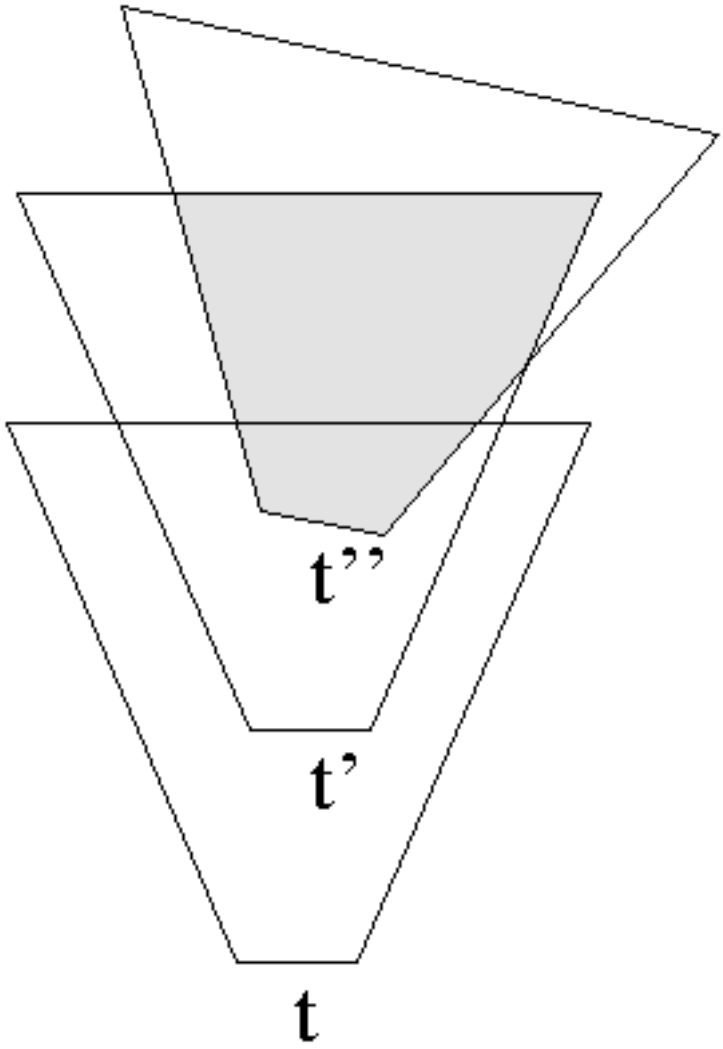

FIG. 6.17 – Evolution du champ de vision du robot au fur et à mesure qu'il avance. En gris, on a représenté la zone commune aux prises de vue aux instants t' et t"

Petit à petit, le robot construit un modèle d'environnement, principalement en termes d'obstacles émergeant du sol pour obtenir *in fine* une carte des obstacles reconstruits par maillages de Delaunay filtrés.

Parallèlement, pour chaque objet reconnu comme un obstacle à reconstruire, sa boîte englobante sert de référence pour éventuellement mettre à jour l'aspect de l'objet grâce à une autre vue. L'ensemble des obstacles rencontrés est stocké dans une liste doublement chaînée dont les éléments sont constitués des champs suivants :

- double Maxx,Maxy,Maxz,Minx,Miny,Minz ; (correspondant à la boîte englobante non orientée de l'obstacle)
- int Numero ; (correspondant au numero de l'obstacle)
- int NbPoints[nbvues] ; (correspondant aux nombres de points dans chacune des *nbvues* vues)
- Point vue[nbvues] ; (correspondant aux points de visée de chacune des *nbvues* vues)
- Point centre[nbvues] ; (correspondant aux centroïdes de l'obstacle estimé pour chacune des *nbvues* vues)
- int Current ; (correspondant à l'indice de la vue précédemment traitée)
- struct objet *Next,*Prev ;

En effet, si un obstacle est détecté à l'instant t, on compare sa boîte englobante avec les boîtes englobantes de tous les autres obstacles déjà reconstruits et stockés, puis on compare la distance aux centroïdes. En fonction de ces mesures d'intersection et de proximité, on décide si l'obstacle nouvellement détecté est une autre vue d'un obstacle déjà reconstruit ou s'il s'agit d'un nouvel obstacle. Cette procédure de mise à jour est délicate à mettre en oeuvre car elle suppose une interprétation fiable des objets rencontrés, voire un module de reconnaissance d'obstacles pour être totalement robuste. Malgré tout, elle est simple et donne la plupart du temps une carte d'obstacles fiable (sur les scènes de la base 1, cf. chapitre 7).

Les problèmes d'exploration de scènes ont surtout été étudiés dans le cadre d'environnements très structurés et autorisant une connaissance *a priori* sur cette structure. La plupart du temps, l'objectif est une exploration complète et optimale de la scène par asservissement visuel. L'ensemble de ces études s'inscrit dans le concept de vision active [Baj88][TTA95][TL95][ECR92][Alo90][Bal91][MC99].

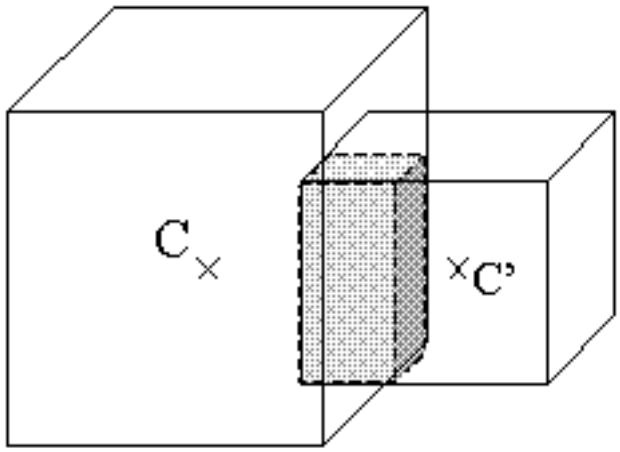

FIG. 6.18 – Configuration de boîtes englobantes de deux obstacles détectés : la distance CC' et la zone d'intersection entre les deux boîtes représentée en grisée permettent d'associer les obstacles au cours de l'exploration

Cette problématique en environnement quelconque et *sans modèle* n'a jamais été à notre connaissance réellement abordée.

6.6.2 Reconstruction des objets

En stéréoscopie, le recalage de vues est un problème ardu. Dans la thèse d'Anthony Mallet [Mal01], on tente entre autres de recaler directement deux nuages de points 3D. Dans la pratique, et jusqu'à récemment, le robot du LAAS utilise également l'odométrie et le positionnement GPS pour avoir en permanence la position du robot dans un référentiel global terrestre. Malheureusement, en raison de la relative imprécision de ces moyens de recalage, et surtout de la faible précision de la méthode stéréoscopique, les données recalées par ces méthodes donnent toujours un décalage, souvent systématique, des différents nuages de points correspondant à un même objet, et ceci même quand l'axe de visée est le même. Espérer donner un modèle complet et unifié de l'objet dans ces conditions dépasse les objectifs de cette thèse. En effet, si on tente de fusionner deux tels nuages de points 3D, on obtient au mieux un effet d'épis commme illustré en figure 6.19.

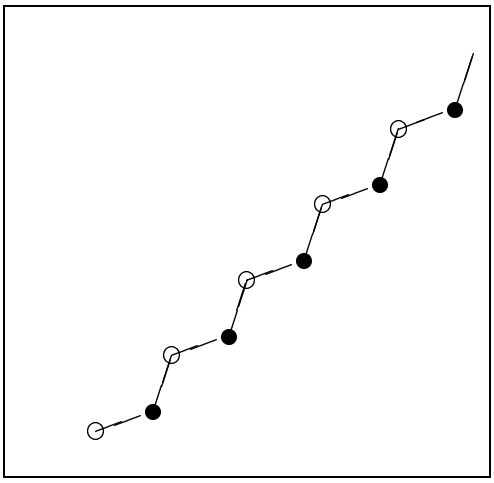

FIG. 6.19 – Effet d'épis sur la reconstruction d'un contour après fusion de nuages de points 2D légèrement décalés

C'est la raison pour laquelle on a préféré gérer la description d'objets décrits par plusieurs vues, en restant à ce niveau fragmenté. Un objet sera représenté par un nombre prédéfini de vues (par exemple 4) au mieux réparties sur la sphère des vues. Si une nouvelle vue d'un objet faisant un angle suffisamment distant avec une vue dejà acquise est disponible, on complète la description de l'objet 3D par cette vue. Si cette nouvelle acquisition est suffisamment proche d'une vue précédemment stockée pour la description de l'objet, on met à jour la description pour cette vue de l'objet si le nombre de points nouvellement acquis pour cette vue est supérieur au précédent.

6.6.3 Organigramme

Le processus est résumé dans l'organigramme de la figure 6.20. La gestion au niveau de la reconstruction 3D multivues d'un obstacle n'est pas explicitée dans la mesure où elle demeure encore largement insuffisante.

6.7 Complexité algorithmique

La méthododologie développée est assez gourmande en temps de calcul. Supposons que l'ensemble de départ soit constitué de N points, qu'il y ait au plus C objets dans la scène et que nous bornons à K le nombre d'itérations pour que l'al-

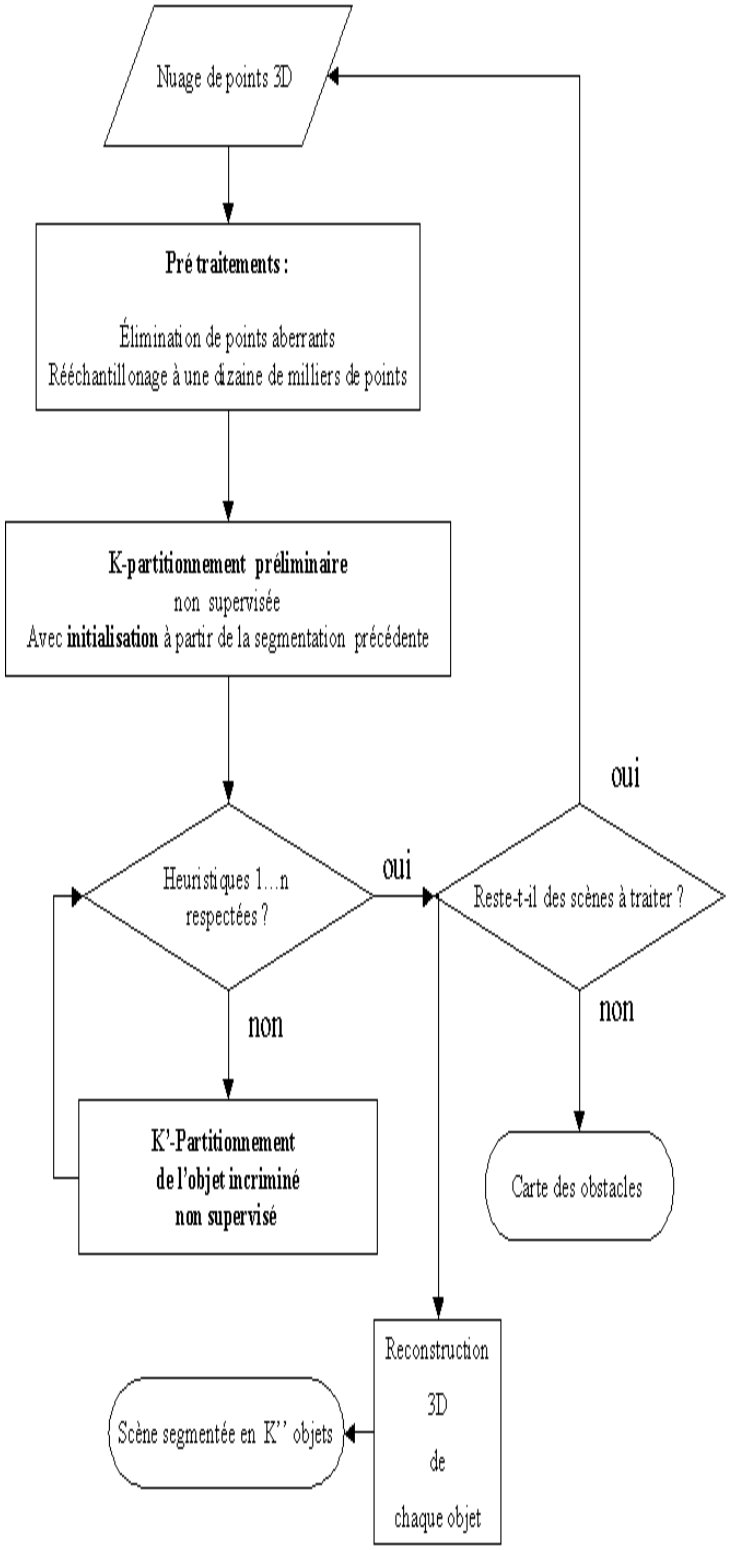

FIG. 6.20 – Organigramme de la stratégie de reconstruction d'un nuage de points 3D en mode exploratoire

gorithme de regroupement *CMFE* termine. Au pire, la phase de partitionnement initial a une complexité en *O(KNC)*.

Il existe par ailleurs des algorithmes pour le calcul du diagramme de Voronoï ou du complexe de Delaunay de complexité en *O(NlogN)* pour le cas 2D (en $O(NlogN + N^{d/2})$ où d est la dimension de l'espace dans le cas général [JDY95]). Cette complexité est la même pour le calcul des α-ouverts utilisés dans l'heuristique *hAG* et pour la reconstruction 3D. Donc à chaque application de l'heuristique *hAG*, la complexité au pire est en *0(NlogN)*.

On a finalement une complexité au pire en *0(HN(KC+log(N))* où H est le nombre de fois où on relance le partitionnement. En pratique, si l'on adopte le schéma rapide, on diminue considérablement le nombre de points sur lequel effectuer un nouveau regroupement de type *CMFE*. Par ailleurs, le calcul des triangulations de Delaunay se fait sur chaque objet et non pas sur l'ensemble de départ ce qui réduit également considérablement la complexité algorithmique au fur et à mesure que l'on structure le nuage de points.

De plus, les différents éléments de la méthodologie sont parallélisables. On peut calculer différents K-partitionnements en parallèle dans la phase d'initialisation. Les tests heuristiques sur les groupements et notamment les calculs d'α-ouverts peuvent également être effectués en parallèle.

Enfin, dans nos expérimentations, les nuages de points ont pu être rééchantillonés à loisir sans modification notable du résultat de la segmentation. Nous travaillions en général sur des nuages de quelques dizaines de milliers de points. Sans oublier qu'en mode exploratoire notamment, nous pouvons disposer d'une initialisation proche de la structure finale à détecter en utilisant une analyse précédente, ce qui améliore aussi considérablement le temps de traitement pour l'algorithme de recherche itérative d'extremum *CMFE*.

6.8 Conclusion

Nous avons donc décrit une méthodologie globale d'interprétation de scènes décrites par un nuage de points 3D *sans modèle* explicite des objets qu'elles contiennent. Nous avons illustré notre propos à travers la problématique de navigation automome d'un robot et la détection d'obstacles. Les stratégies décrites précédemment dans le cas statique ou dynamique demeurent unifiées et leur ca-

ractère générique permet de les appliquer à tous types de données. En effet, par l'intermédiaire du jeu d'heuristiques aisément modifiable à l'intérieur du système puisqu'il ne lui est pas intrinsèquement lié, on peut parvenir au niveau de détail désiré en segmentation. Nous allons montrer dans le chapitre suivant à travers les résultats obtenus que la généricité de cette méthodologie permet de traiter des données aussi différentes que des scènes d'environnements naturels non structurés aussi bien que des scènes très structurées en adaptant simplement les heuristiques utilisées et bien entendu les seuils métriques mis en jeu.

Chapitre 7

Résultats

7.1 Introduction

Comme pour toute méthodologie, seule la mise à l'épreuve sur une batterie d'images variées permet sa validation. La première difficulté dans nos disciplines consiste à se créer une base d'images réelles suffisamment diversifiée pouvant illustrer un échantillon large de cas réels. Quelques efforts apparaissent dans ce sens dans la communauté pour rendre disponible des images de profondeur. Nos sources sont quant à elles assez hétérogènes.

La nature de nos images sous forme de nuage de points 3D est de trois ordres :

- une base de scènes en environnement extérieur peu structuré ; ces images ont été acquises grâce au robot LAMA du laboratoire LAAS-CNRS à Toulouse ; elles ont été acquises en séquence sur un terrain jouxtant un parking et un petit parc dans lequel on trouve des rochers, des obstacles divers déposés par les manipulateurs, des arbres, des murets, des dépressions de terrain ou des pentes etc. ;
- une base de scènes en environnement intérieur très structuré ; ces images ont été acquises par le robot SOISIC de la société MENSI par un capteur laser dans une usine nucléaire puis confiées au laboratoire CAOR de l'Ecole des Mines à Paris, et sont constituées d'une part de plans correspondant au sol et aux murs et d'autre part de tuyauterie et containairs divers.
- une base de scènes très structurées de type univers-plan ; cette base a été initiée par l'équipe de Hoover [HJBJ+96] accompagnée d'une procédure de quantification de la qualité de la segmentation obtenue. Plusieurs concours

ont été lancés dans les conférences internationales utilisant ce cadre formel. Cette base est disponible sur l'internet à l'adresse suivante `http://marathon.csee.usf.edu/range/seg-comp/SegComp.html`.

L'avantage de la première base d'images réside dans sa présentation sous forme de série d'images acquises à mesure que le robot avance et observe la scène grâce à une vision périscopique. On peut ainsi recréer un cas de navigation autonome réel d'un robot dans un environnement inconnu, dans lequel on peut essayer de reconstruire peu à peu un modèle de l'univers exploré en terme d'objets-obstacles et de zones navigables. Dans la plupart des expériences menées, le robot avance d'un mètre avant de prendre une série de prises stéréoscopiques à des intervalles d'angles réguliers par rapport à sa direction d'avancée (typiquement de -120 deg à +120 deg tous les 40 deg ce qui fournit sept vues différentes à un point d'arrêt donné).

Nous présentons en figure 7.1 un panorama de la base 1 des objets rencontrés dans les séries de scènes acquises par le robot LAMA et en figure 7.2 un panorama des objets de la base 3. Les images d'intensité de la base 2 ne sont pas disponibles dans la série d'acquisition.

7.2 Description d'une scène non structurée

Nous allons commencer par présenter des résultats sur des nuages de points représentant des scènes d'extérieur non structurées acquises par le robot LAMA. De l'ensemble des scènes observées par ce robot lors de son exploration, nous avons voulu extraire les plus significatives par la variété de leur contenu. Dans cette pespective, nous illustrons sur chacune des scènes suivantes l'image de la caméra gauche de la scène observée en (a), la segmentation obtenue en 2D en (b) et la segmentation obtenue sur le nuage de points 3D selon deux vues en (c) et (d). Les heuristiques utilisées dans la méthodologie de description optimale sont les deux heuristiques d'ambiguïté géométrique et topologique telle que décrites dans la partie 6. Les seuils utilisés sont décrits dans le tableau 7.2.

La méthodologie adoptée prévilégie la rapidité de temps de calcul et utilise donc le schéma rapide décrit dans la partie 6.

Rappelons enfin que les objets du fond de chaque scène ne sont pas pris en compte dans le nuage 3D reconstruit en raison de la faible précision en recons-

FIG. 7.1 – Illustration des séries de scènes acquises par le robot LAMA

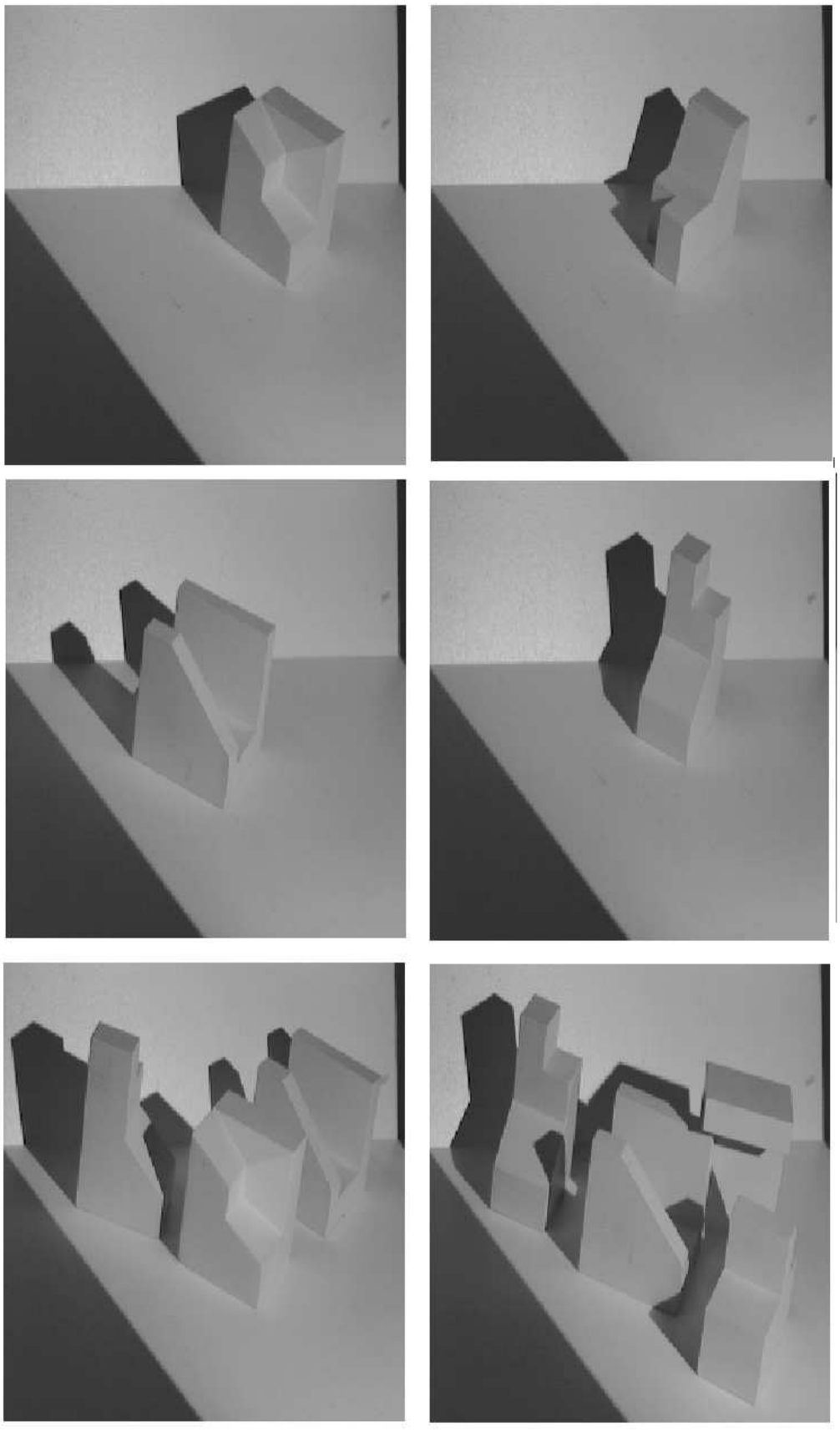

FIG. 7.2 – Illustration des scènes de type univers-plan

Heuristique	Seuil	Valeur
hAT	λ_{min}	5 cm
hAT	β	$\pi/12$
hAG	α	$3 * med_{Del}\rho$
hAG	Σ	50 points

TAB. 7.1 – Valeur des seuils utilisés pour traiter la base 1

truction 3D au delà d'une certaine distance au capteur.

Les figures 7.3,7.4,7.5,7.6 mettent en scène un obstacle humain. Les petits rochers présents autour de l'homme son également détectés. L'arbre en arrière-plan dans la figure 7.5 est lui aussi correctement segmenté.

Dans la figure 7.7, on voit comment l'algorithme segmente des obstacles de types rochers répartis sur le sol et de tailles différentes : deux grand blocs au premier et en arrière plan et un plus petit dans un plan intermédiaire.

La figure 7.8, on peut voir un obstacle construit pour l'occasion à l'aide d'une chaise recouverte de dalles de pierre et de vêtements.

Dans les figures 7.9 et 7.10, on voit l'habileté de l'algorithme à détecter des structures plus ou moins filiformes émergeant du sol : un lampadaire dans la première et un tronc d'arbre dans la seconde. Dans cette dernière figure 7.10, l'algorithme extrait également le pan de muret visible derrière l'arbre, et mieux encore, dans la figure suivante 7.11, le muret est subdivisé selon ses deux plans visibles formant une partie du parallépipède qui lui correspondrait dans une reconstruction de type CAO, cependant que le feuillage haut de l'arbre devant le muret est correctement segmenté.

Dans la figure 7.12, on voit la capacité de l'algorithme à séparer des objets de nature semblables mais très rapprochés comme des files de voiture garées : deux sur la droite et deux sur la gauche du champ de vision.

Enfin, la dernière figure 7.13 illustre la possibilité de détecter de petites surélevations de type bords de trottoirs. Par ailleurs, dans cette image, l'arbre est décomposé en deux parties de troncs et son feuillage.

L'évaluation quantitative des résultats sur cette base est difficile et fastidieuse à mettre en oeuvre. Nous espérons valider les résultats obtenus en intégrant le module développé sur une plateforme robotique dans un objectif de navigation auto-

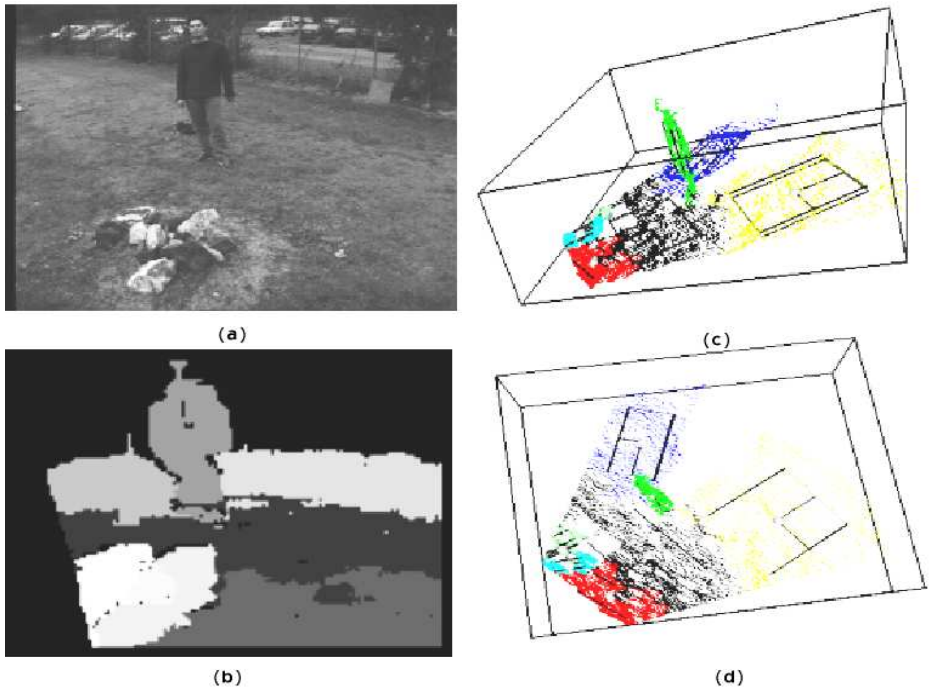

FIG. 7.3 – (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté

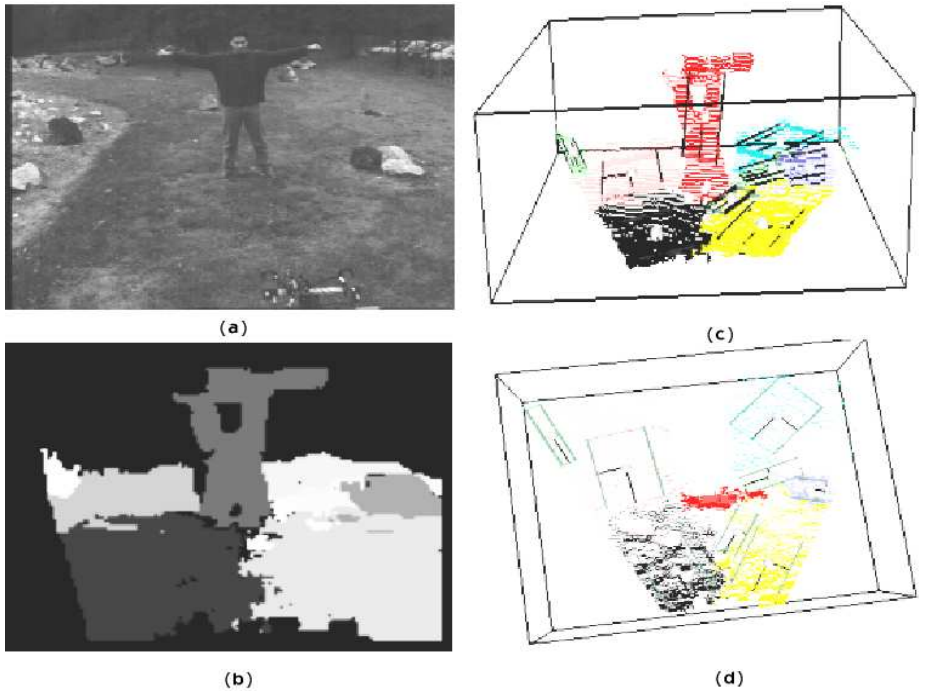

FIG. 7.4 – (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté

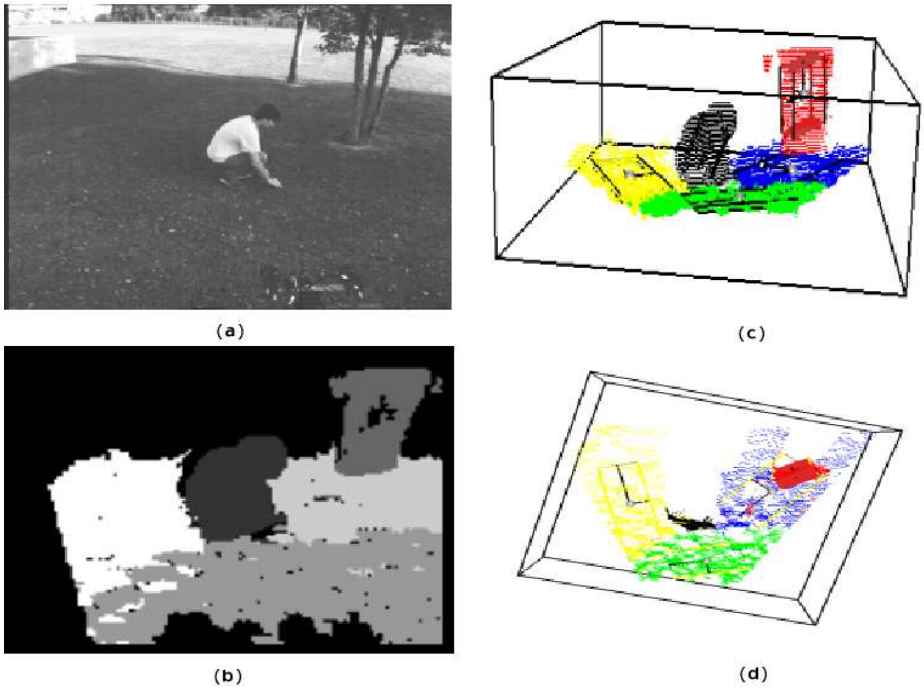

FIG. 7.5 – (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté

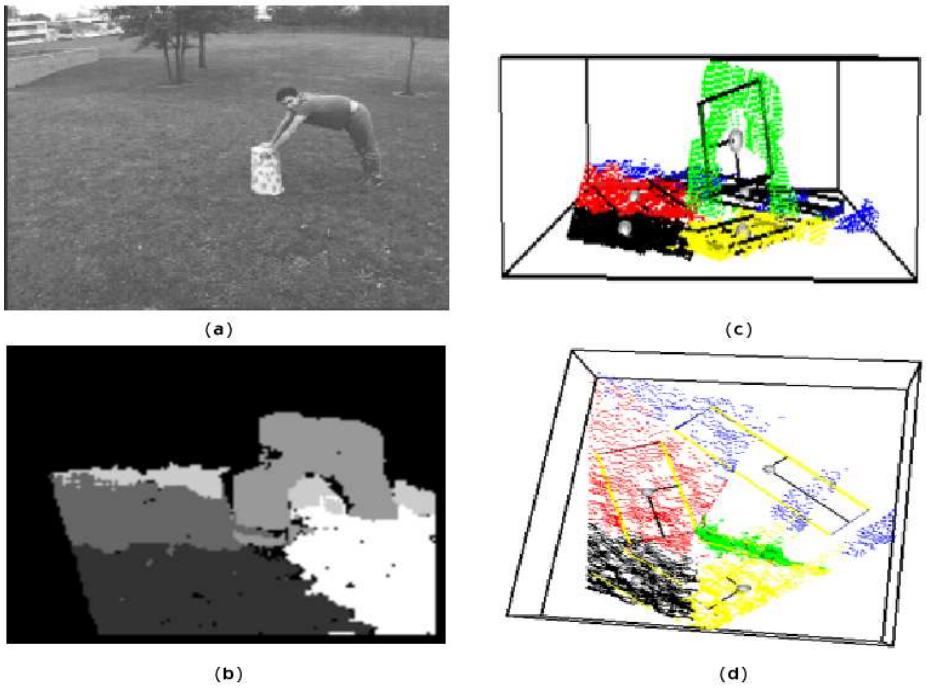

FIG. 7.6 – (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté

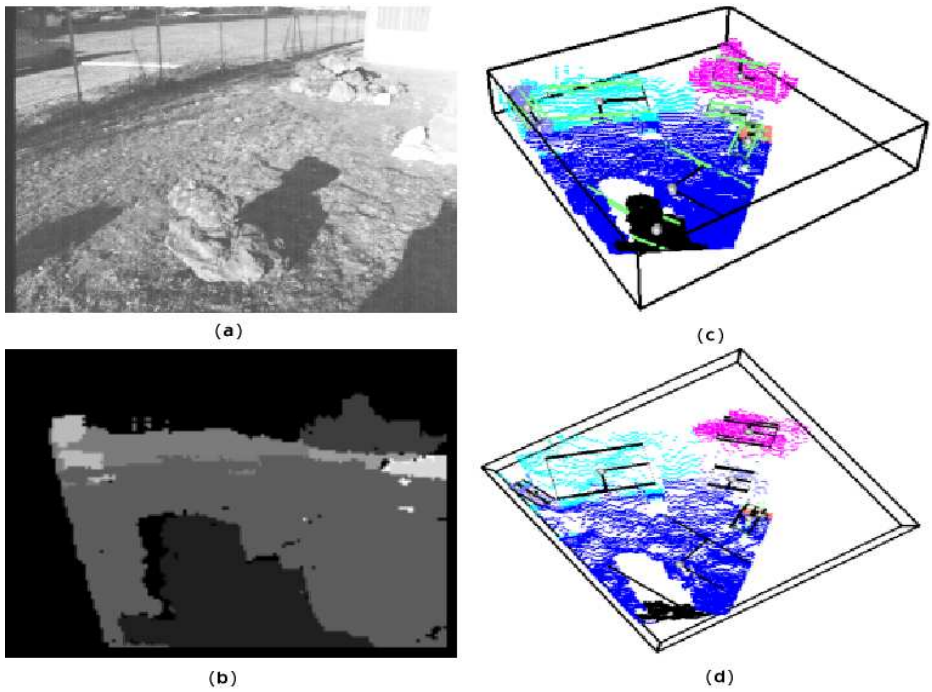

FIG. 7.7 – (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté

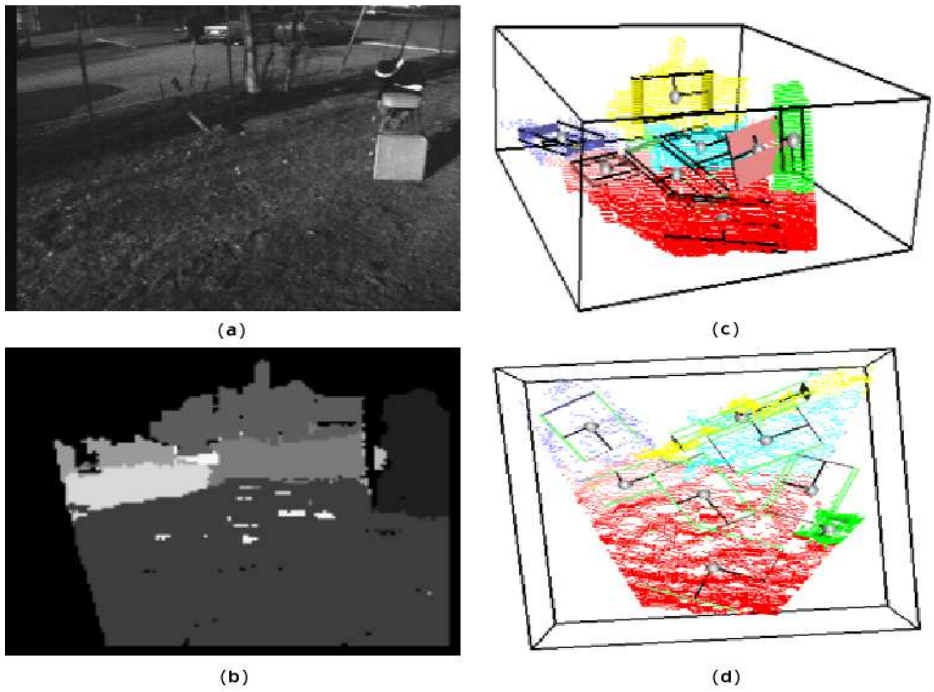

FIG. 7.8 – (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté

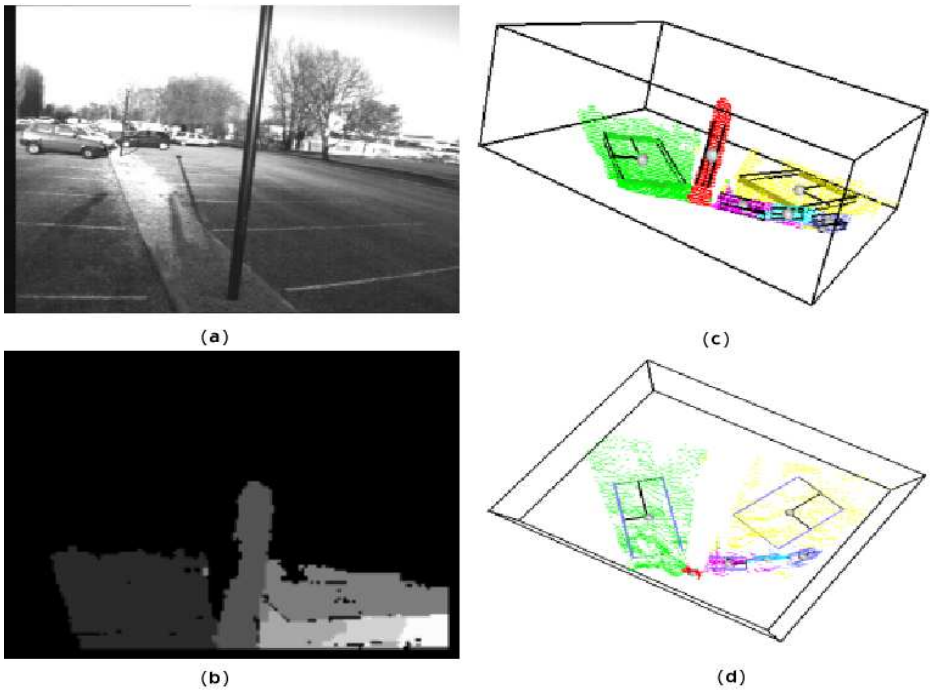

FIG. 7.9 – (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté

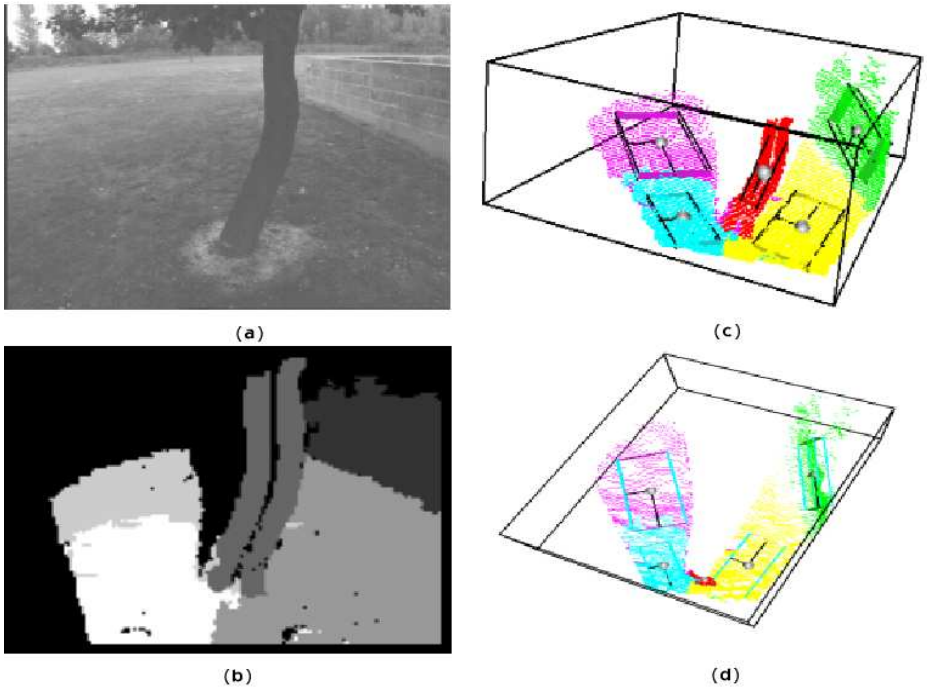

FIG. 7.10 – (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté

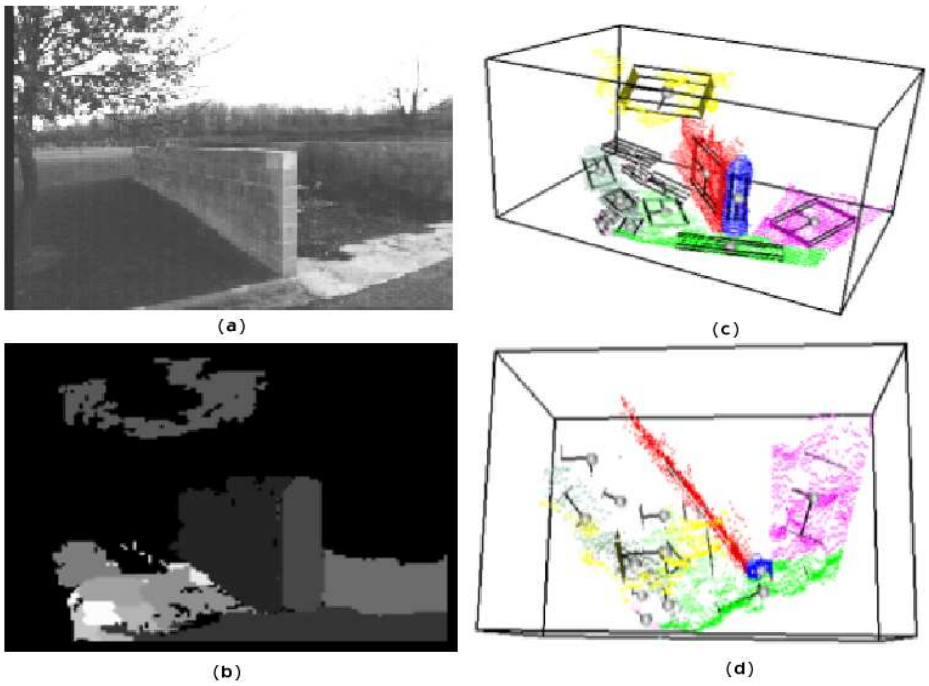

FIG. 7.11 – (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté

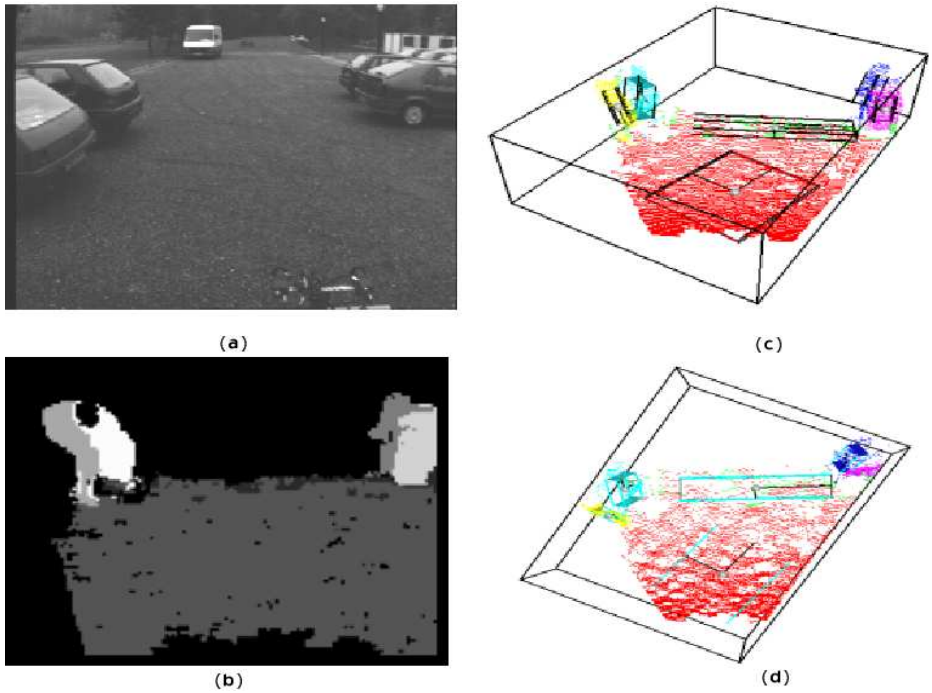

FIG. 7.12 – (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté

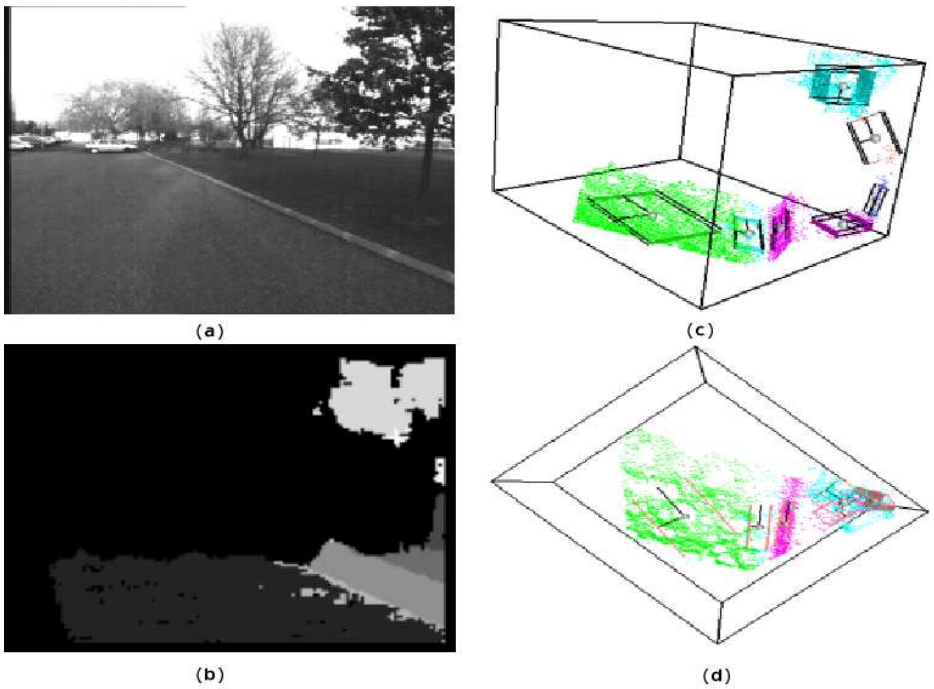

FIG. 7.13 – (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté

nome. Le robot LAMA pourrait par exemple utiliser cette description en termes d'objets à des fins de localisation [Mal01] [BBCM95]. En tout état de cause, qualitativement cette méthode montre sa robustesse et son efficacité en terme d'interprétation sur l'ensemble des images disponibles dans ce type d'environnement (c'est-à-dire sur une centaine d'images dont certaines redondantes en raison du principe de prise de vue tous les mètres).

7.3 Description d'une scène structurée

Nous exposons à présent les résultats que nous pouvons obtenir sur des scènes beaucoup plus structurées. Les scènes de cette base sont d'assez mauvaise qualité visuelle ce qui rend assez difficile la visualisation 2D des résultats.

7.3.1 Base 2

Typiquement, on trouvera dans cette catagorie des intérieurs d'usine nucléaire dans lesquels on trouve de nombreux pans de murs et de sols ou de sursols, des

tuyaux ou bien encore des containers.

Toutes les figures présentées sont constituées de quatre vues du nuage de points 3D segmenté à l'aide de l'heuristique hAGC avec des valeurs de seuils identiques aux précédentes (cf. tableau 7.2).

La première scène présentée en figure 7.14 est l'illustration de la difficulté de reconstruire par des modèles de type CAO un tel enchevêtrement d'objets divers, associés certes à des primitives géométriques simples. Notre algorithme permet de simplifier une telle reconstruction en exhibant les différentes zones d'intérêt et, mieux, la plupart du temps, les objets eux-mêmes. On voit deux zones de sols en rouge et vert, un pilier en T décomposé en trois branches linéaires, un tuyau et son rebord, de nombreux pans de murs, et différents autres ensembles de piliers.

La deuxième scène présentée en figure 7.15, plus simple, très structurée, exhibe un parallépipède correctement éclaté en chacune de ses faces, quelques pans verticaux isolés et le haut de la pyramide à degrés efficacement segmentés.

La troisième scène présentée en figure 7.16, assez complexe également, montre un escalier de fer en bleu, les deux rampes à gauche et à droite isolées, deux pans de mur, une grande bande de sol en rose et plusieurs plans de sursols au-dessus légèrement décalés (certes difficile à apprécier par projections 2D).

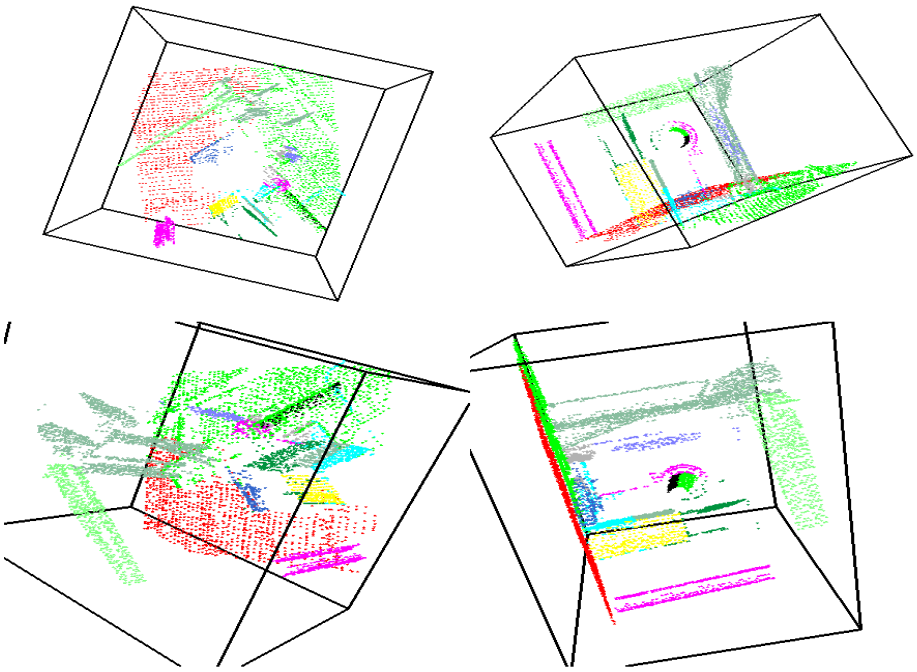

FIG. 7.14 – Vues du nuage de points 3D segmenté

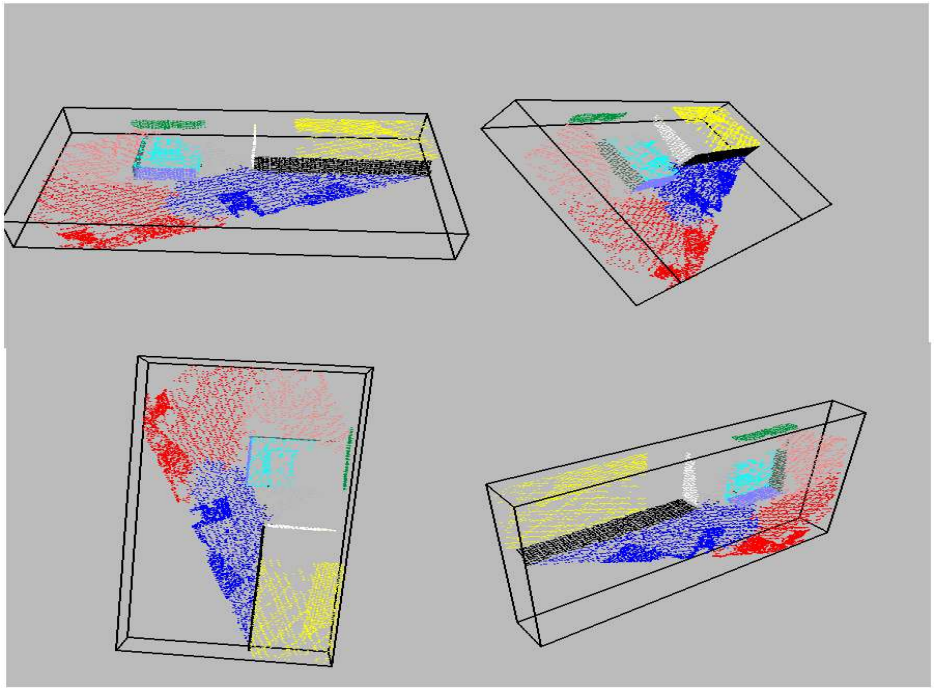

FIG. 7.15 – Vues du nuage de points 3D segmenté

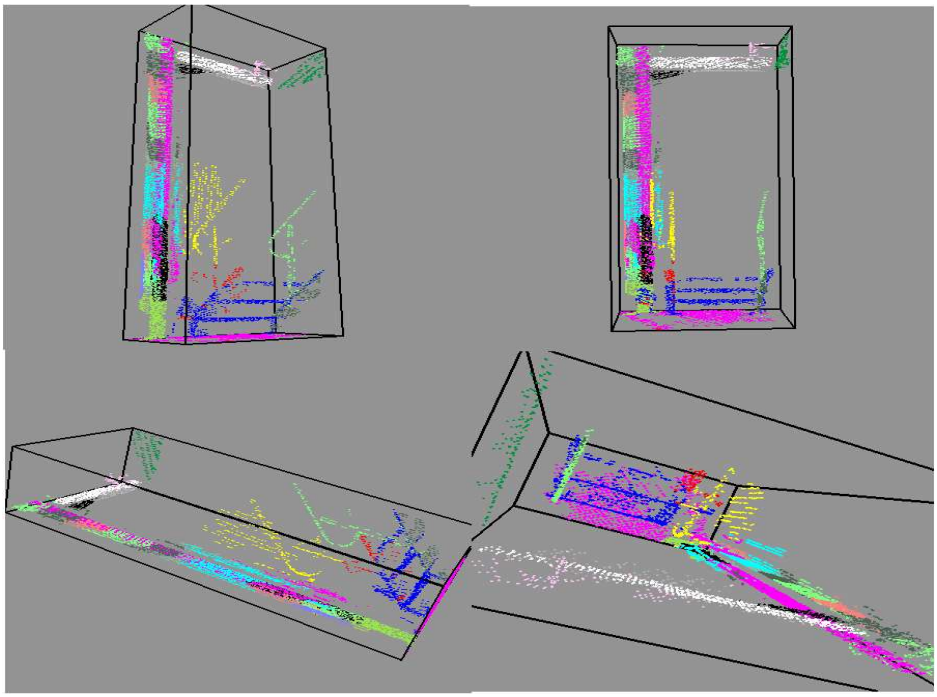

FIG. 7.16 – Vues du nuage de points 3D segmenté

7.3.2 Base 3

Pour chaque scène de la base 3, on donne en (a) l'image d'intensité, en (b) la partition préliminaire obtenue par la simple application du critère *DPM*, en (c) la partition finale obtenue après l'application itérative de l'heuristique *hAP* et enfin en (d) la segmentation finale ; à l'exception de la première scène illustrée en figure 7.17 dans laquelle on a présenté quelques résultats intermédiaires. La segmentation finale est obtenue en rassemblant les zones planes de mêmes orientations. Dans ce type de scènes polyédriques le but est de découvrir toutes les régions planes qui la constituent. On donne dans le tableau suivant 7.3.2 les seuils utilisés. On utilise un seuil supplémentaire sur la surface des régions reconstruites en cours de partitionnement à conserver appelé Σ.

Heuristique	Seuil	Valeur
hAP	λ_{min}	1 cm
non def.	Σ	500 points

TAB. 7.2 – Valeur des seuils utilisés pour traiter la base 3

La méthodologie adoptée prévilégie la rapidité de temps de calcul et utilise donc le schéma rapide décrit dans la partie 6.

On voit encore d'un point de vue purement qualitatif la capacité de la méthode à décrire la scène selon les spécifications indiquées.

7.3.2.1 Comparaison avec d'autres approches

Cette base est née de la volonté de proposer une méthodologie commune de comparaison quantitive de résultats de segmentation d'images de profondeur. Utilisée dans le cadre de différents concours organisés lors de conférences internationales comme *ICCV* ou *ICPR*, plusieurs auteurs commencent à l'utiliser comme référence dans nombre d'articles de revues. Ainsi dans [CB00], l'auteur compare ses résultats à ceux obtenus par les universités de South California, de Berne et d'Edimbourg. A partir de l'image de profondeur, chaque université teste ses algorithmes de segmentation. Avec cette préoccupation d'apporter quelques éléments quantitatifs des performances de notre méthodologie, on commence par présenter en figure 7.25 et 7.26 une comparaison qualitative des résultats obtenus sur les images 7.23 et 7.24.

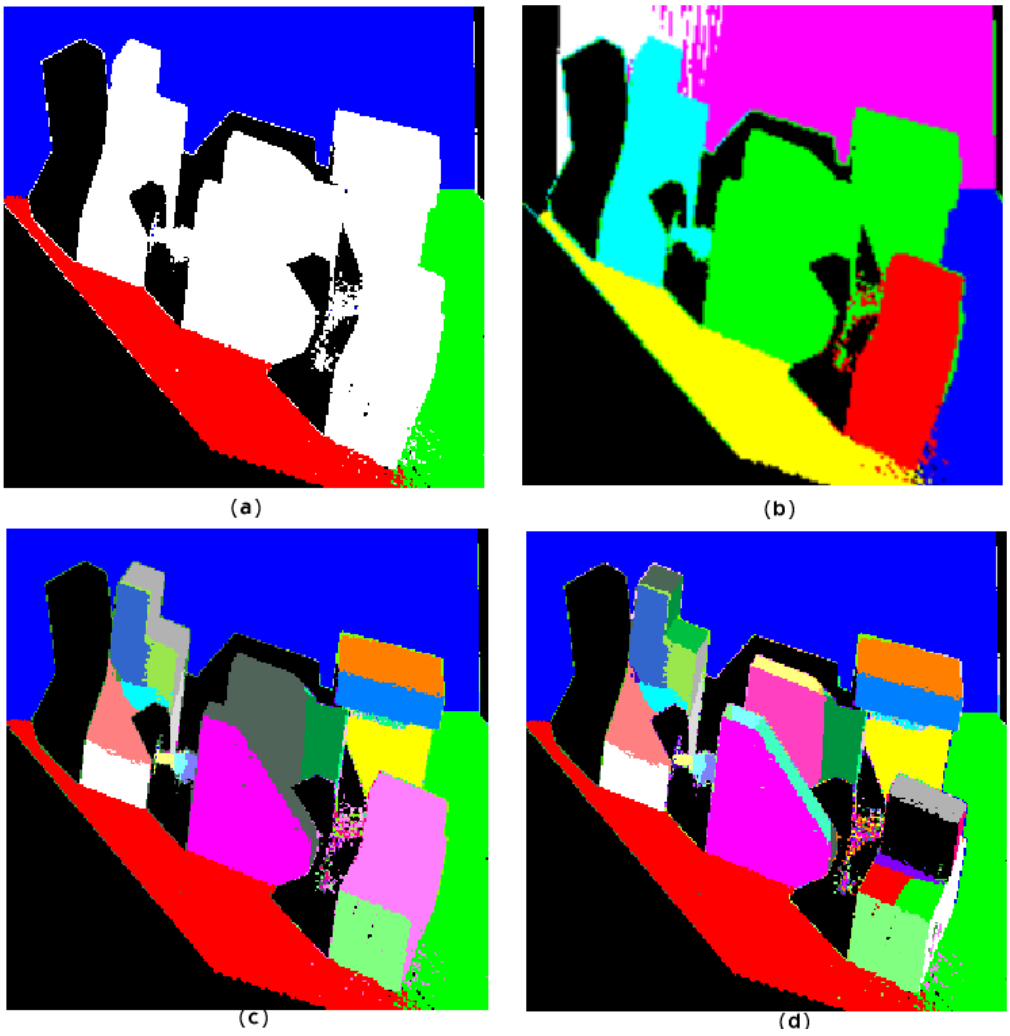

FIG. 7.17 – (a) Partition préliminaire (b)(c) Partitions intermédiaires (d) Partition finale

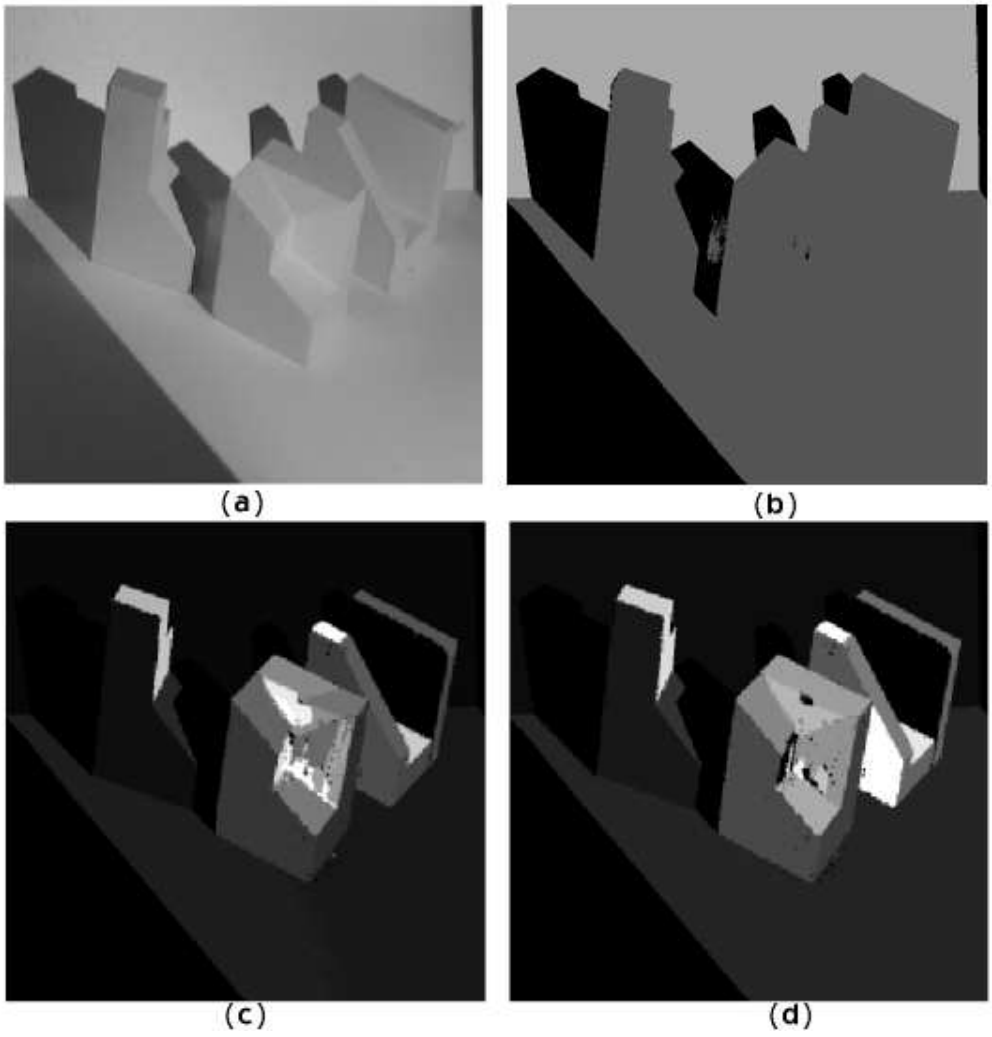

FIG. 7.18 – (a) Image d'intensité (b) Partition préliminaire (c) Partition finale (d) Segmentation finale

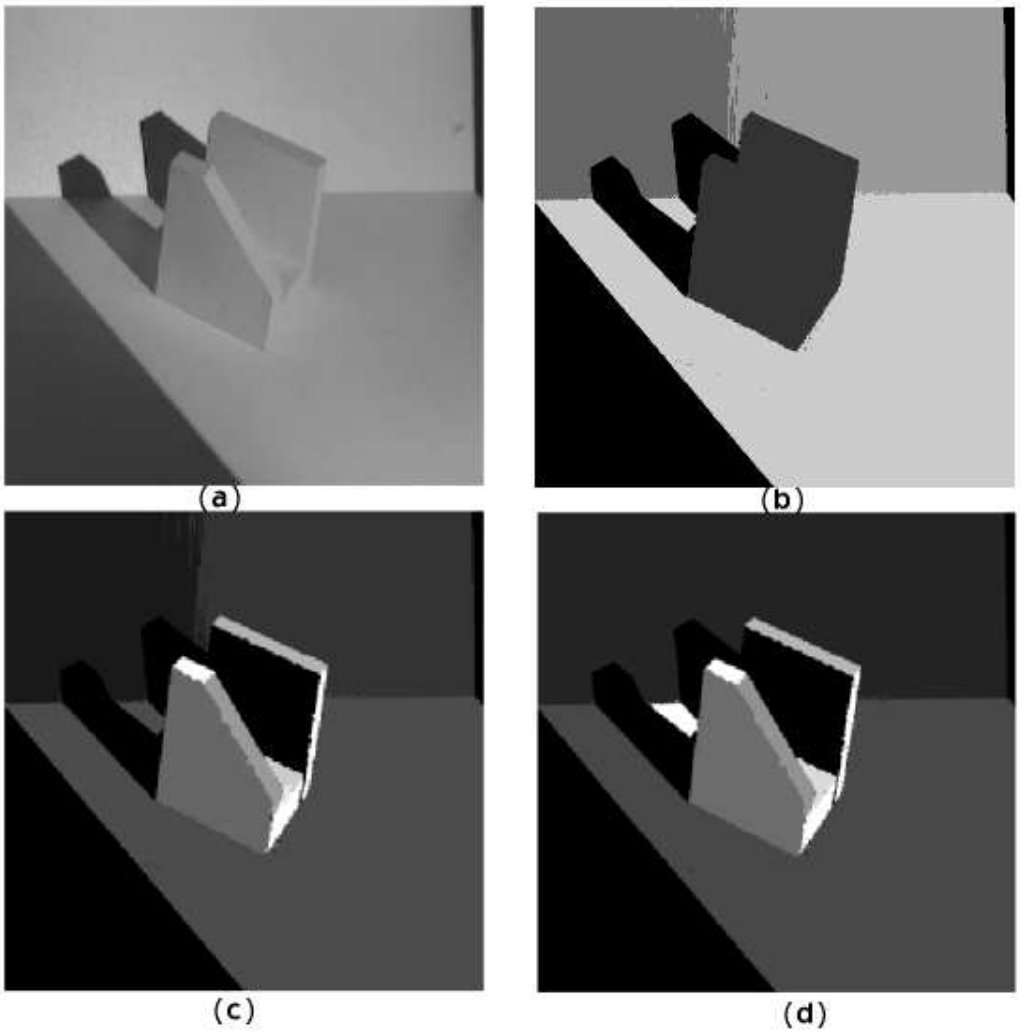

FIG. 7.19 – (a) Image d'intensité (b) Partition préliminaire (c) Partition finale (d) Segmentation finale

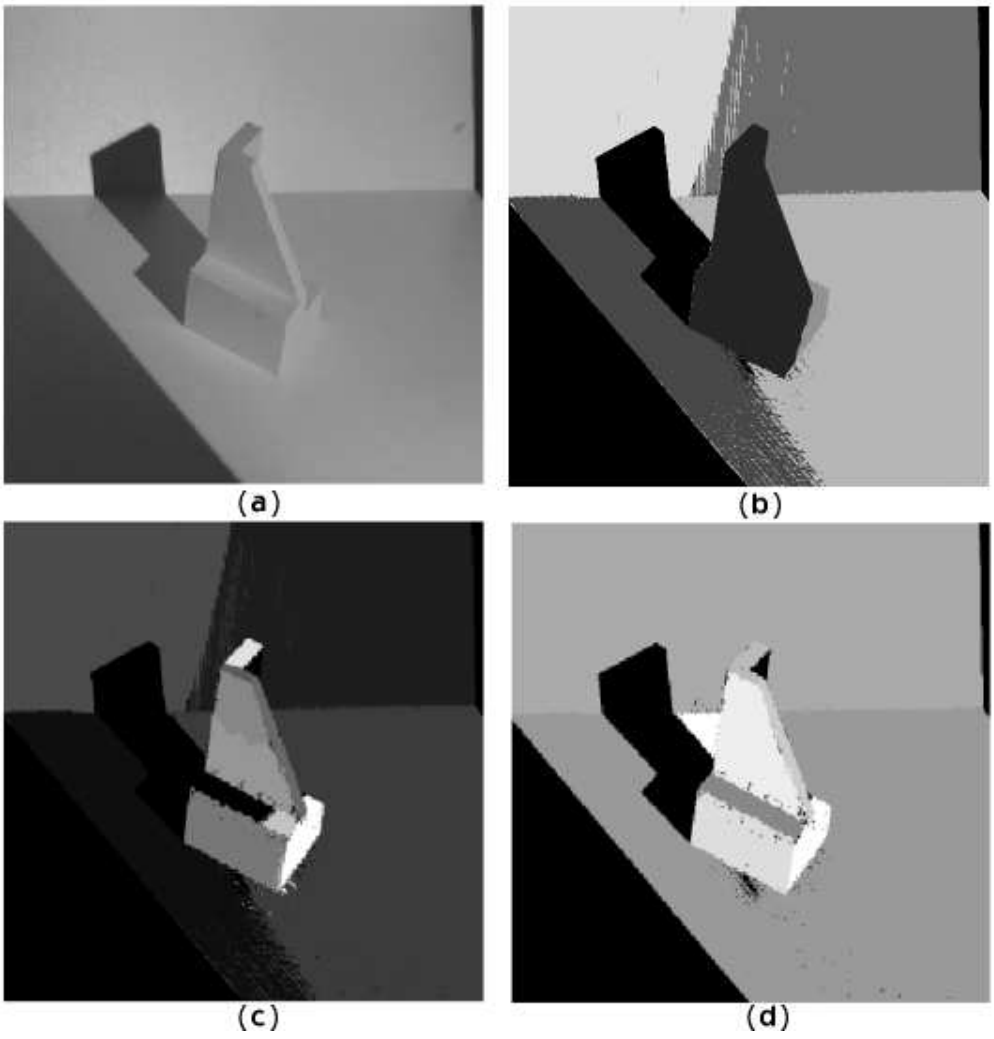

FIG. 7.20 – (a) Image d'intensité (b) Partition préliminaire (c) Partition finale (d) Segmentation finale

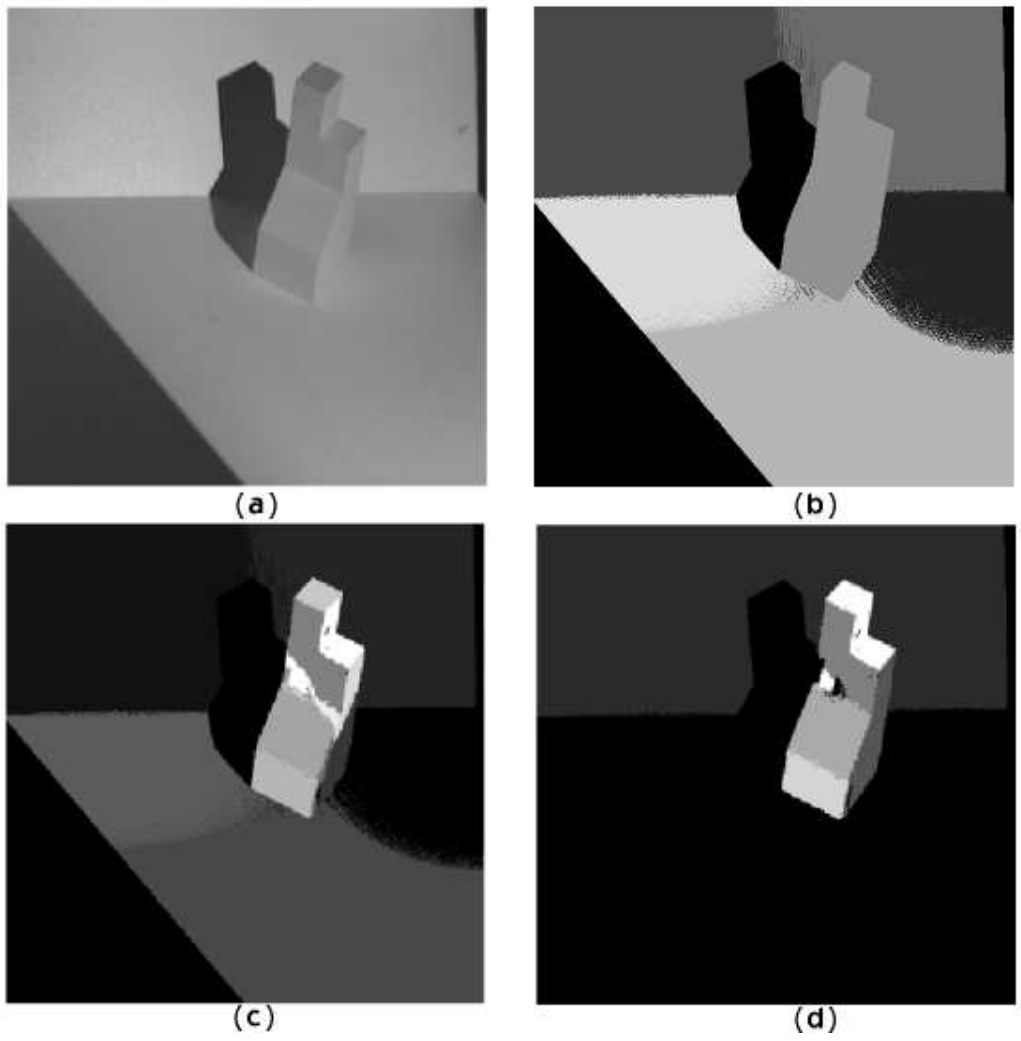

FIG. 7.21 – (a) Image d'intensité (b) Partition préliminaire (c) Partition finale (d) Segmentation finale

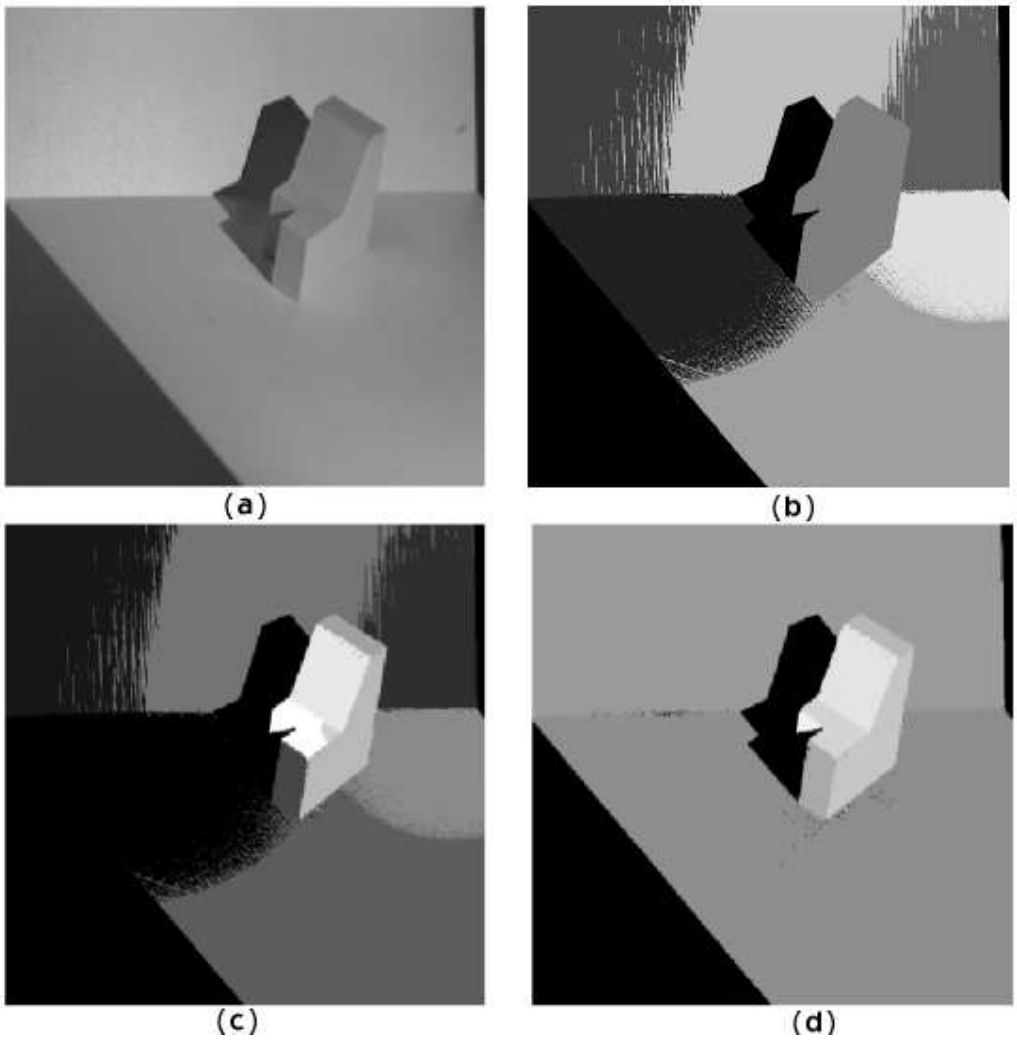

FIG. 7.22 – (a) Image d'intensité (b) Partition préliminaire (c) Partition finale (d) Segmentation finale

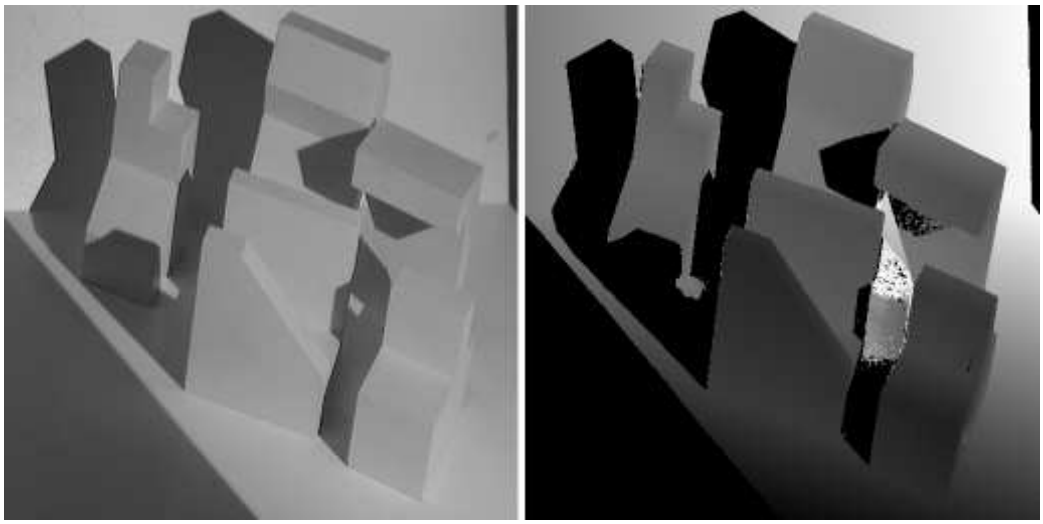

FIG. 7.23 – De gauche à droite : image d'intensités et image de profondeur de abw.train.0

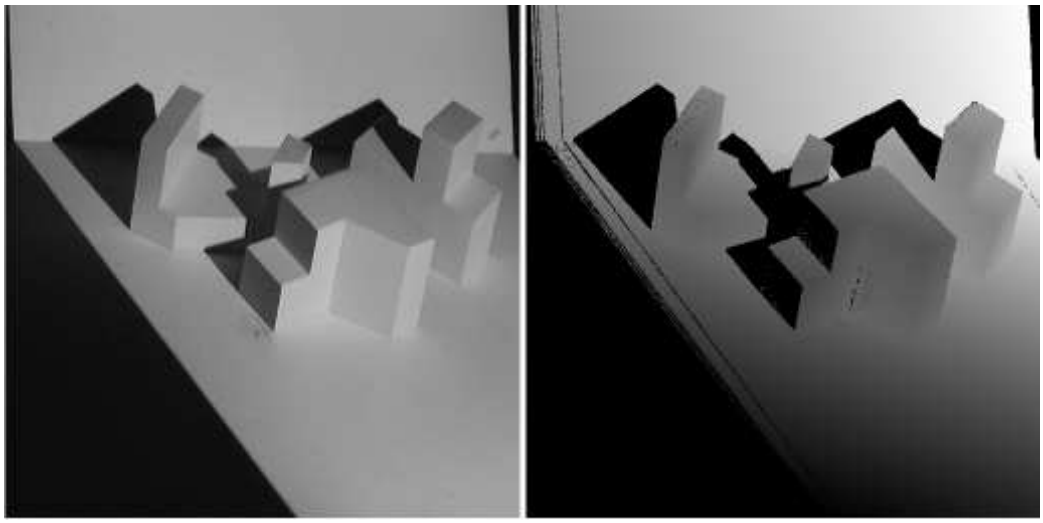

FIG. 7.24 – De gauche à droite : image d'intensités et image de profondeur de abw.train.1

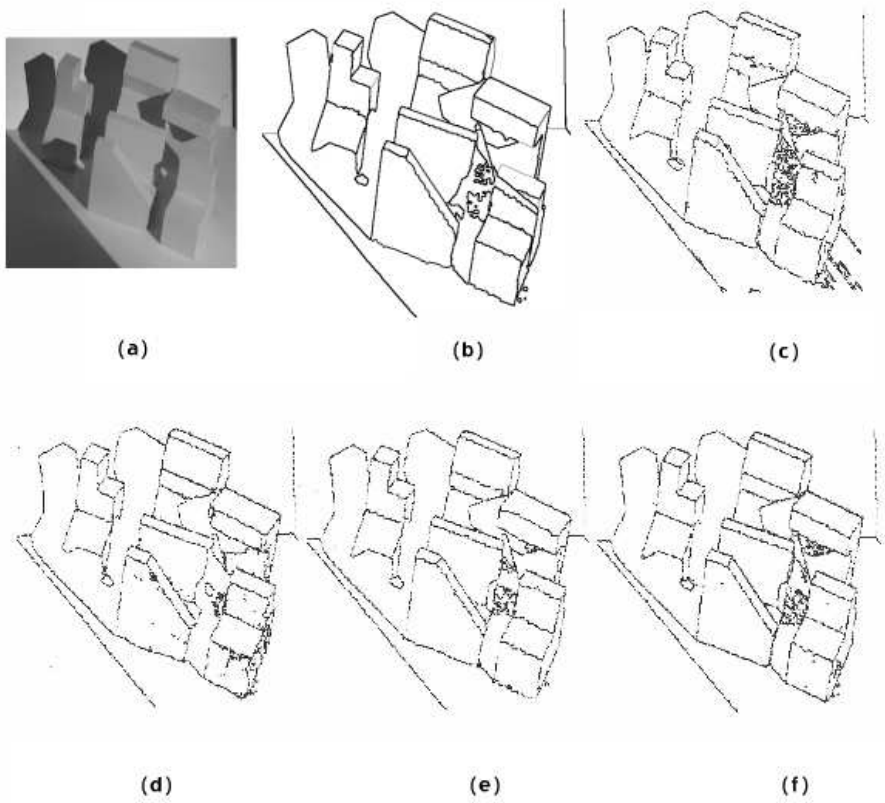

FIG. 7.25 – Segmentation de l'image abw.train.0 (a) par notre méthodologie (b) par l'Université de Lyon 1 (c) par l'université de South Florida (d) par l'université de Berne (e) par l'université d'Edinbourg (f)

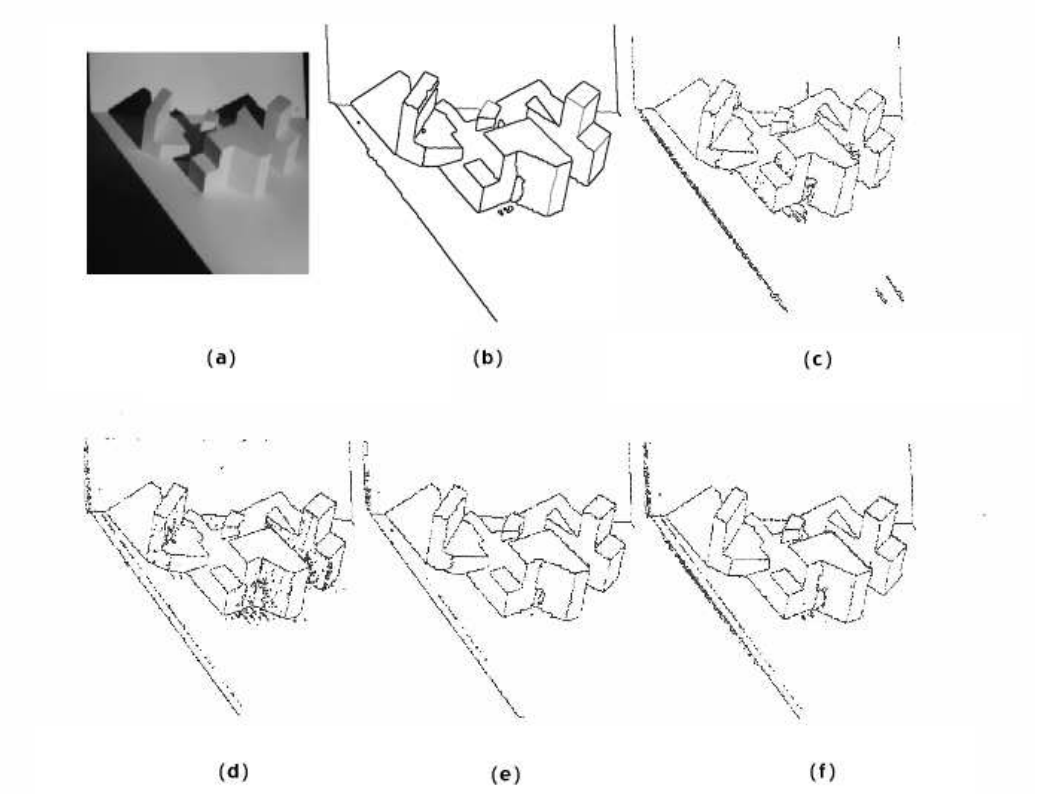

FIG. 7.26 – Segmentation de l'image abw.train.1 (a) par notre méthodologie (b) par l'Université de Lyon 1 (c) par l'université de South Florida (d) par l'université de Berne (e) par l'université d'Edinbourg (f)

Ces résultats n'ont pas vocation à concurrencer les résultats obtenus par des algorithmes qui furent pour la plupart développés spécifiquement pour ce concours mais plutôt à prouver la généricité de la méthodologie développée. En effet, par l'intermédiaire de l'introduction d'une bonne heuristique et de quelques post-traitements (rassemblement des groupements voisins de même orientation essentiellement), on parvient à de bons résultats de segmentation. Il est à noter que la plupart des algorithmes servant à la comparaison utilisent pleinement la structure de grille régulière fournie par l'image de profondeur et cherchent tous à calculer localement l'information de courbure de surface à partir du voisinage inscrit dans la grille régulière. Seule Raphaëlle Chaine tente de s'affranchir de cette structure de grille régulière. En outre, leurs méthodes sembleront difficilement fonctionner sur des nuages de points moins structurés, c'est-à-dire en environnement naturel.

Nous donnons une comparaison quantitative de notre algorithme de partitionnement sur la base de 30 images de la base ABW rassemblée par l'université de South California. Ces images ont été segmentées manuellement en utilisant des modèles CAD des objets pour obtenir des images "Vérité Terrain" (images VT). Un outil développé par Hoover et al. [HJBJ$^+$96] compare "objectivement" la segmentation obtenue par l'algorithme testé (images "Segmentation Machine" ou images SM) avec la "Vérité Terrain" en utilisant un ensemble de métriques de performance comme le nombre de régions correctement segmentées, omises ou correspondant à du bruit, sur ou sous segmentées. Les définitions de classification dépendent d'un taux de tolérance T ($0.5 < T \leq 1$) reflétant la rigueur accordée à la définition. Dans ce formalisme, on définit les métriques suivantes :

1. Détection correcte : une paire de régions R_n dans l'image VT et R_m dans l'image SM est classifiée en détection correcte si au moins T pourcent des pixels de R_m sont des pixels de R_n et inversement ;
2. Sur-segmentation : une région R_n dans l'image VT et un ensemble de régions R_{m1},...,R_{mx} dans l'image SM sont classifiées en sur-segmentation si au moins T pourcent des pixels dans chaque région R_{mi} sont des pixels de R_n et inversement ;
3. Sous-segmentation : un ensemble de régions R_{n1},...,R_{nx} de l'image VT et une région R_m dans l'image SM sont classifiées en sous-segmentation si au moins T pourcent des pixels de la région R_m sont des pixels de la réunion de R_{n1},...,R_{nx} ;

4. Non détectée : une région de l'image VT non classifiée en détection correcte, sous ou sur segmentation est classifiée en région non détectée ;
5. Bruit : une région de l'image SM non classifiée en détection correcte, sous ou sur segmentation est classifiée en région de bruit.

Pour mesurer la précision de la géométrie reconstruite, l'algorithme de segmentation doit fournir les normales aux surfaces des régions détectées. La précision est évaluée en mesurant la différence angulaire (en degrés) entre chaque normale comparée à l'image VT.

Une moyenne de ces mesures a été effectuée sur l'ensemble de ces images. Le tableau 7.3.2.1 donne le nombre de régions "Vérité Terrain" moyen par image ainsi que le nombre moyen de régions correctement détectées par les différents algorithmes, le nombre moyen de régions sur et sous-segmentées, le nombres moyens de régions non détectées et de régions faussement détectées. La mesure géométrique de précision sur les angles inter-régions est également fournie. Les résultats sont comparés à ceux des universités de South Florida, de Washington [HJ91], de Berne [JB94], d'Edinbourg [FA91], de Birmingham [KS00], et de notre méthode élaborée à l'Université Paris 5.

Groupe de rech.	Régions VT	Correct	Diff.ang. (écart)	Sur-sgmt.	Sous-sgmt.	Omis	Bruit
USF	15.2	12.7	1.6(0.8)	0.2	0.1	2.1	1.2
WSU	15.2	9.7	1.6(0.7)	0.5	0.2	4.5	2.2
UB	15.2	12.8	1.3(0.8)	0.5	0.1	1.7	2.1
UE	15.2	13.4	1.6(0.9)	0.4	0.2	1.1	0.8
UBham	15.2	13.4	1.6(0.9)	0.4	0.3	0.8	1.1
UP5	15.2	12.2	1.7(0.9)	0.3	0.1	2.6	2.2

TAB. 7.3 – Comparaison quantitative des résultats en segmentation de la base 3 par différentes universités

A la lecture de ce tableau, notre méthode n'apporte pas une amélioration quantitative mais rivalise avec les autres méthodes. D'un point de vue qualitatif, elle donne également des résultats comparables comme en témoignent les figures 7.25 et 7.26. L'apport majeur de notre méthodologie est sa généricité. Elle peut s'affranchir à un coût minimum de la diversité des environnements à analyser.

7.4 Description d'un univers d'obstacles

Lorsque le robot est en phase d'exploration d'un environnement inconnu, le but ultime de notre application est de construire une carte des obstacles pour la zone explorée. C'est-à-dire que dans un référentiel commun, on place l'ensemble des objets étiquetés 1 (obstacle plan) ou 2 (obstacle hyperellipsoïdal) et reconstruits après filtrage morphologique. Nous illustrons en 7.28 la carte des obstacles reconstruits obtenue après l'analyse de la série d'image de la figure 7.27 selon le schéma exploratoire décrit dans la partie 6.

Cet objectif suppose une mise en correspondance des obstacles détectés à chaque nouvelle scène segmentée. Pour l'instant, nous rappelons que nous prenons en compte la distance aux deux centroides et l'intersection des deux boîtes englobantes. Cette stratégie basique donne de bons résultats lorsque les obstacles sont bien séparés et relativement compacts mais échoue lorsqu'on à affaire à un obstacle s'étendant sur plusieurs scènes comme un long mur. Dans ce cas, le mur n'est pas correctement interprété : il est soit fragmenté soit tronqué.

7.5 Conclusion

Ces résultats sont plus qu'encourageants, même si difficilement quantifiables. Sur toutes nos expériences, la méthodologie de segmentation de nuages de points 3D proposée permet d'obtenir un résultat exploitable. Sur notre application essentielle d'évitement d'obstacles, la stratégie d'interprétation et de reconstruction d'un nuage de points 3D est très efficace. Dans l'objectif plus ambitieux de reconstruction globale de l'univers exploré, les résultats sont bons mais mériteraient d'être améliorés : en effet, à mesure que le robot avance, il arrive que des obstacles soient dédoublés dans la carte globale reconstruite parce que la mise en correspondance des obstacles détectés n'est pas assez performante. Ainsi par exemple un obstacle détecté à l'instant t, constituant une partie d'un obstacle précédemment reconstruit à l'instant t' dans la carte globale, n'est pas associé à cet obstacle. Il constitue alors un artéfact dédoublant l'obstacle physique. Une étape de reconnaissance pourrait améliorer les résultats, dans la mesure où les décisions prises pour la mise en correspondance reposent sur des considérations exclusivement métriques et basiques.

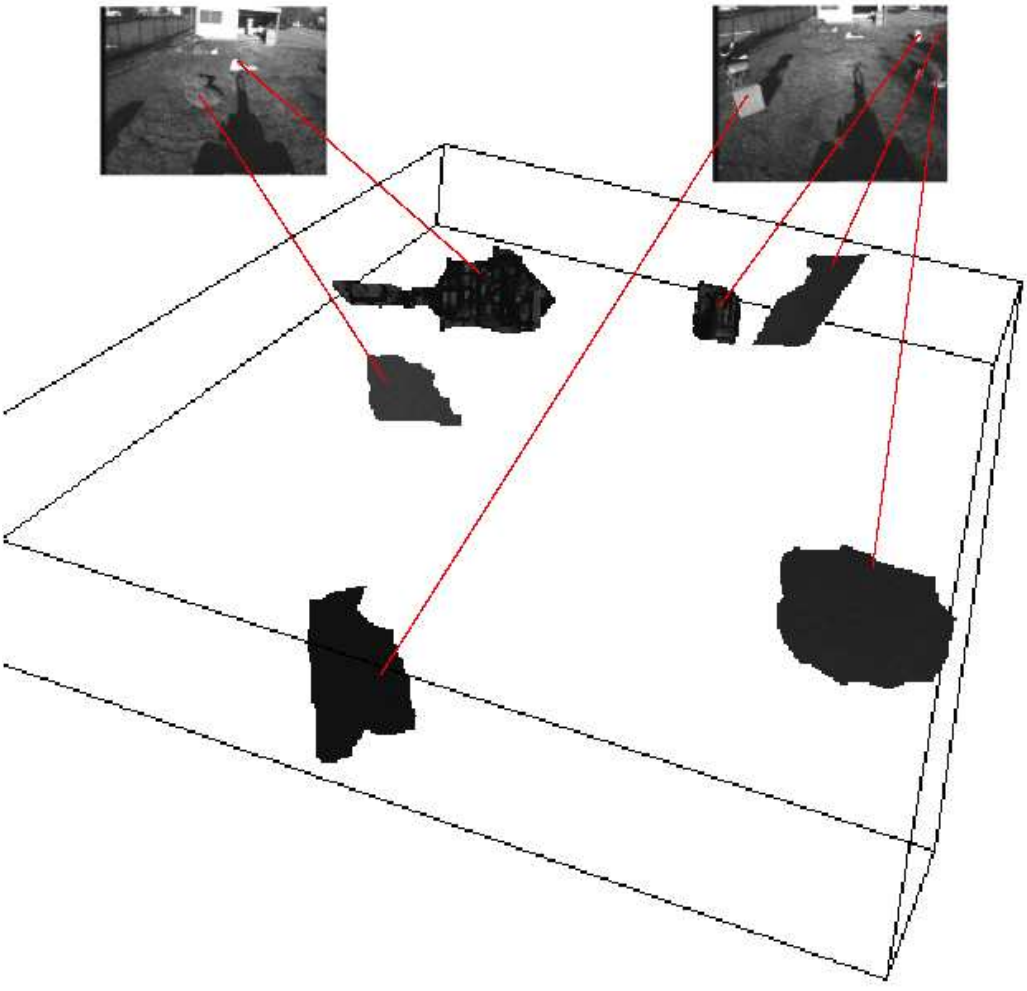

FIG. 7.28 – Carte d'obstacles reconstruits

Chapitre 8

Conclusion

Nous partions d'une problématique très appliquée : l'analyse d'un nuage de points 3D obtenu par les techniques stéréoscopiques en vue de la navigation autonome d'un robot. Dès l'origine toutefois, la voie de recherche s'ouvrait assez largement puisqu'aucune hypothèse d'environnement n'était faite et qu'aucun modèle ne devait être utilisé.

A partir de cette problématique initiale, nous avons établi une distinction de principe entre deux objectifs : l'interprétation d'une scène isolée et l'exploration d'un univers dans laquelle la notion de fusion d'information intervient.

La stratégie d'exploration d'un univers souffre d'un déficit de module de reconnaissance de formes. En terme clair, sans modèle il est difficile de donner une carte d'obstacle robuste dans la mesure où la définition des obstacles rencontrés reposent exclusivement sur une poignée de critères numériques flous et arbitraires comme la pente de l'obstacle et l'espacement minimal entre les obstacles. Une erreur à ce niveau tend alors à se propager de scène en scène. Malgré tout, on peut obtenir une carte tout à fait exploitable dans la mesure où tous les obstacles sont présents, même si parfois quelques artefacts embrouillent la carte finale (par exemple, un morceau d'obstacle obtenu à un instant t n'est pas mis en correspondance avec un obstacle de la carte déjà reconstruit à un instant t', ce qui correspond à un dédoublement d'obstacle).

En revanche, nous pouvons dire que la stratégie proposée pour le premier objectif d'interprétation de scène isolée a été entièrement validée sur l'ensemble des données rassemblées. Partant du postulat que l'algorithme *CMFE* fournit quel que soit le nombre de groupements spécifié une K-partition cohérente, nous l'avons

placé au centre de notre méthodologie.

Notre contribution pourrait être en partie localisée dans le domaine de la conception et le développement d'algorithmes spécifiques de traitement de nuage de points géométriques 2D désorganisé. L'ensemble est situé dans la lignée des opérateurs morphologiques qui s'appliquent traditionnellement à des images d'intensité 2D structurées régulièrement. Les techniques d'α-ouverture ont permis notamment de reconstruire la forme 3D des objets rencontrés de façon épurée. Ces α-opérateurs ont également permis l'introduction d'heuristiques d'analyse des groupements en cours de formation lors du processus de partitionnement. Ces heuristiques ont été conçues pour controler le niveau de description désiré en fonction de l'application finale.

Globalement, nous avons donc défini une méthodologie d'analyse de nuage de points adaptative et générique. En fonction de l'application et de quelques critères qualitatifs sur les objets à détecter fournis au processus de façon indépendante, la méthodologie est capable de traiter tout type de nuages de points. En effet, les scènes qui nous ont permis de tester notre démarche décrivent aussi bien des environnements structurés que non structurés, en milieu extérieur ou intérieur, naturels ou réalisés par l'homme.

La démarche adoptée pourrait être considérée comme déclarative. En effet, dans ce cas, l'algorithme *CMFE* serait le moteur d'inférence et les heuristiques constitueraient les connaissances, qui peuvent être succinctes ou plus élaborées, assimilées à des groupements à effectuer (cf. figure 8.1). Le traitement et la connaissance sont bel et bien indépendants dans notre système. Cette organisation permet de l'envisager comme une structure d'accueil générique à la problématique générale d'interprétation de nuages de points.

Les perspectives de ce travail sont l'intégration d'un module de description de formes 2D. Une tel module pourrait permettre de définir des heuristiques intéressantes s'appuyant plus sur la forme des groupements constitués. Ainsi, l'heuristique topologique *hAT* devrait être avantageusement remplacée par une heuristique capable de reconnaitre un groupement dont la forme d'une de ses silhouettes est décrite par la figure 8.2. Il s'agit effectivement des groupements concernés par cette heuristique : une zone de sol large avalant un obstacle peu dense.

Ce module de description de formes 2D devrait également permettre de développer un module de reconnaissance de formes 2D indépendant lui aussi pour

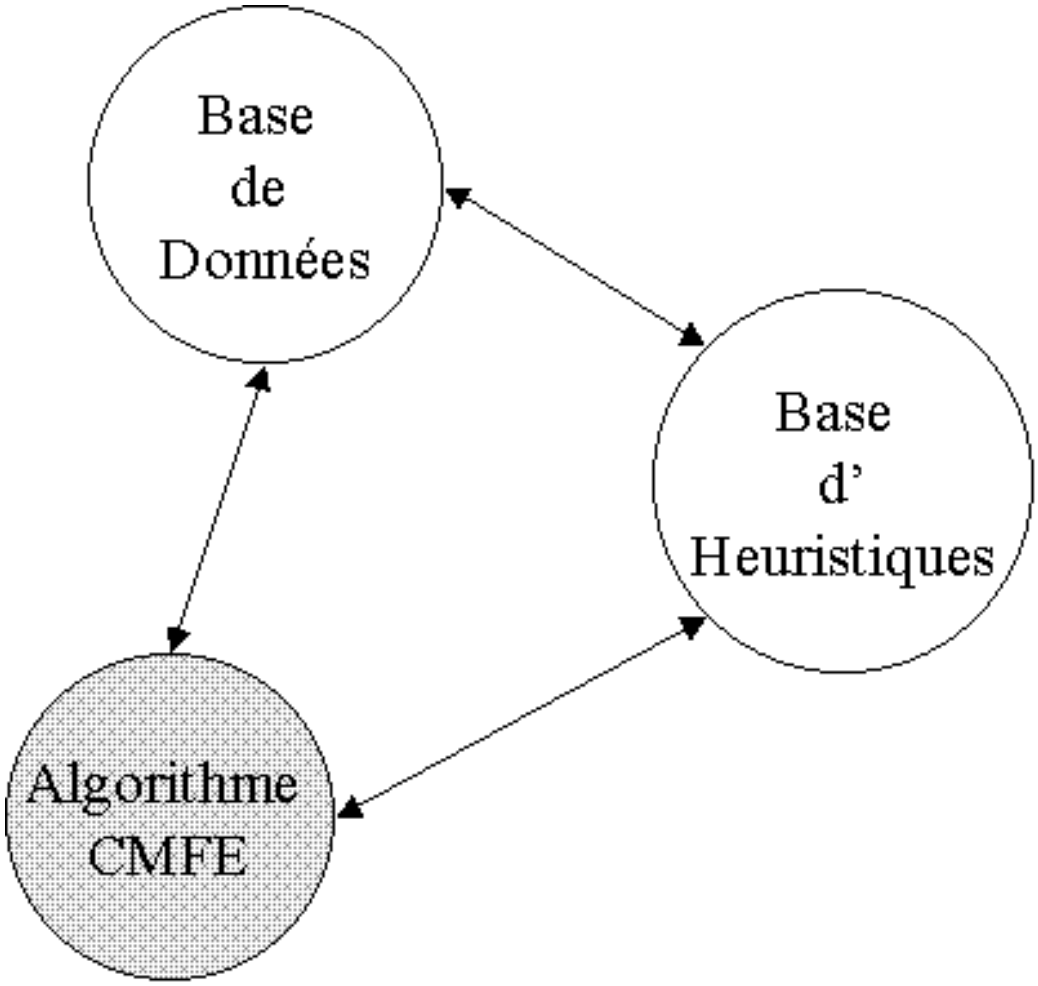

FIG. 8.1 – Structure générique d'analyse de nuages de points

FIG. 8.2 – Forme typique des groupements concernés par l'heuristique hAT

rendre plus robuste encore la stratégie d'interprétation de la scène, dans le cadre de la robotique mobile notamment et de l'exploration d'environnements. A partir de masques binaires extraits d'images bruitées, il s'agirait de comparer la silhouette 2D d'un objet avec un modèle 3D connu.

Nous pensons avoir ainsi apporté quelques éléments de réponse aux objectifs posés au début de ce travail scientifique.

Bibliographie

[AB98] N. Amenta and M. Bern. Surface reconstruction by voronoï filtering. In *14th Symposium on Computation Geometry*, June 1998.

[ABE99] N. Amenta, M. Bern, and D. Eppstein. The crust and the β-skeleton : Combinatorial curve reconstruction. In *Graphical Models and Image Processing*, 1999.

[ABK98] N. Amenta, M. Bern, and M. Kamvysellis. A new voronoï-based surface reconstruction algorithm. In *Proc. SIGGRAPH'98*, 1998.

[Alo90] Y. Aloimonos. Purposive and qualitative active vision. In *International Conference on Pattern Recognition(ICPR'90)*, pages 346–360, 1990.

[AM90] E. Arbogast and R. Mohr. 3d structures inference from images sequences. In *workshop on Syntactical and Structural Pattern Recognition*, pages 21–37, Murray-Hill, 1990.

[ANV90] ANVAR. Méthodes optiques tridimensionnelles pour l'identification, le positionnement et la métrologie. ANVAR, étude réalisée par Bertin Cie, 1990. Série " Technologies et développement", nř 14.

[Att97] D. Attali. r-regular shape reconstruction from unorganized points. In *13th ACM Symposium on Computational Geometry*, pages 248–253, June 1997.

[Aub96] P. Aubry. *Méthodes de vision 2D et 3D pour la robotique*. PhD thesis, Université Paris 6, 1996.

[Aya89] N. Ayache. *Vision stéréoscopique et réception multisensorielle*. InterEditions, Science informatique, 1989.

[Ba96] A.M. Bensaid and al. Validity-guided (re)clustering with applications to image segmentation. *IEEE. Trans. Fuzzy Syst.*, 4 :112–123, 1996.

[Baj88] R. Bajcsy. Active perception. *Proc. IEEE*, 48(1) :996–1005, 1988.

[Bal91] D.H. Ballard. Animate vision. *Artificial Intelligence*, 48(1) :57–86, 1991.

[BB97] F. Bernardini and C. Bajaj. Sampling and reconstructing manifolds using α-shapes. In *9th Canadian Conference on Computational Geometry*, pages 193–198, August 1997.

[BBCM95] S. Betge-Brezetz, R. Chatila, and M.Devy. Object-based modelling and localization in natural environments. In *International CDonference on Robotics and Automation*, pages 2920–2927, 1995.

[BBKF94] M.E. Brandt, T.P. Bohan, L.A. Kramer, and J.M. Fletcher. Estimation of csf, white and gray matter volumes in hydrocephalic children using fuzzy clustering of mr images. *Comp. Med. Imag. and Graph.*, 20 :25–34, 1994.

[BBX95] C. Bajaj, F. Bernardini, and G. Xu. Automatic reconstruction of surfaces and scalar fields from 3d scans. In *SIGGRAPH'95 Proceedings*, pages 109–118, July 1995.

[BCGW81] J.C. Bezdec, C. Coray, R. Gunderson, and J. Watson. Detection and characterization of cluster substructures. *SIAM J. Appl.Math.*, 40 :339–372, 1981.

[Bes89] P. Besl. *Advances in Machine Vision*, chapter chapitre 1. Actival optical range imaging sensors. Springer, 1989.

[Bez74] J.C. Bezdek. Cluster validity with fuzzy sets. *J. Cybernet.*, 3(3) :58–72, 1974.

[Bez75] J.C. Bezdek. Mathematical models for systematic and taxonomy. In *Proc. 8th Internat. Conf. Numerical Taxonomy*, pages 143–166, San Francisco, 1975.

[Bez80] J.C. Bezdek. A convergence theorem for the fuzzy isodata clustering algorithms. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2(1) :1–8, January 1980.

[Bez81] J.C. Bezdek. *Pattern recognition with Fuzzy Objective Function Algorithms*. New York : Plenum Press, 1981.

[BG92] J.D. Boissonat and B. Geiger. Three-dimensionnal reconstruction of complex shapes based on the delaunay triangulation. Rapport de Recherche 1697, INRIA, Sofia Antipolis, Avril 1992.

[BH67] G. Ball and D. Hall. A clustering technique for summarizing multivariate data. *Behavior science*, 12 :153–155, 1967.

[BHC93] J.C. Bezdek, L.O. Hall, and L.P. Clarke. Review of mr image segmentation techniques using pattern recognition. *Med. Phys.*, 20 :1033–1048, 1993.

[BJ81] E. Backer and A.K. Jain. A clustering performance measure based on fuzzy set decomposition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 3(1) :66–75, 1981.

[BJ88] Paul J. Besl and Ramesh C. Jain. Segmentation through variable order surface fitting. *PAMI*, 10(2), march 1988.

[Boi84] J.D. Boissonnat. Geometric structures for three-dimensional shape representation. *ACM Transactions on graphics*, 3(4) :266–286, october 1984.

[Bon94] J. Bonneau. *Mise en oeuvre de techniques de vision artificielle pour l'aide à la modélisation interactive de l'environnement*. PhD thesis, Université Paris 6, 1994.

[Bri85] J.F. Brinkley. Knowledge-driven ultrasonic three-dimensional organ modeling. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 7(4) :431–441, July 1985.

[CB00] R. Chaine and S. Bouakaz. Analyse surfacique de données 3d non structurées : une approche basée sur les graphes. In *RFIA*, pages 37–46, 2000.

[CJMS00] F. Cloppet, J-M.Oliva, and G. Stamon. Angular bissector network, a simplified generalized voronoï diagram : application to processing complex intersections in biomedical images. *PAMI*, 22(1) :120–128, january 2000.

[CL96] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *SIGGRAPH'96 Proceedings*, pages 303–312, July 1996.

[Dav89] R.N. Davé. Use of the adaptative fuzzy clustering algorithm to detect lines in digital images. *Intelligent Robots and Computer Vision VIII*, 1 :600–611, 1989.

[DB92] R.N. Davé and K. Bhaswan. Adaptative fuzzy c-shells clustering and detection of ellipses. *IEEE Trans on Neural Networks*, 3(5) :643–662, may 1992.

[dB97] M. de Berg. *Computationnal geometry : algorithms and applications*. Springer, 1997.

[dC92] B. de Cambray. Modélisation 3d : Etat de l'art. Rapport de recherche 92.71, MASI, octobre 1992.

[DD99] P. Daniel and J.D. Durou. Réalisation d'images réelles vérifiant les hypothèses du shape from shading. In *Actes des journées des jeunes chercheurs francophones en vision par ordinateur ORASIS*, pages 161–170, Aussois, France, 1999.

[Del94] H. Delingette. *Modélisation, déformation et reconnaissance d'objets tridimensionnels à l'aide de maillages simplexes*. PhD thesis, Ecole Centrale de Paris, 1994.

[DH73] R. Duda and P. Hart. *Pattern classification and scene analysis*. New York : Wiley Interscience, 1973.

[DK97] R.N. Davé and R. Krishnapuram. Robust clustering methods : A unified view. *IEEE Trans on Fuzzy Syst.*, 5(2) :270–293, 1997.

[Dun73] J.C. Dunn. A fuzzy relative of the isodata process and in detecting compact well-separated clusters. *Journal of Cybernetics*, 3(3) :32–57, 1973.

[ECR92] B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. *IEEE Transactions on Robotics and Automation*, 8(3) :313–326, 1992.

[EK83] H. Edelsbrunner and D.G. Kirkpatrick. On the shape of a set of points in the plane. *IEEE. Trans. Inform. Theory*, 29 :551–559, 1983.

[EM94] H. Edelsbrunner and E.P. Mücke. Three-dimensionnal alpha-shapes. *ACM Transactions on Graphics*, 13(1) :43–72, 1994.

[Eve89] P. Even. *Modélisation géométrique tridimensionnelle interactive pour la génération de retours synthétiques en téléopération asistée par ordinateur*. PhD thesis, Université de Rennes 1, 1989.

[FA91] P.J. Flynn and A.K.Jain. Bonsai : 3d object recognition using constrained search. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(10) :1066–1075, october 1991.

[FK98] O. Faugeras and R. Keriven. Variational principles, surface reconstruction, pde's, level set methods, and the stereo problem. *IEEE Transactions on Image Processing, Special Issues on Geometry Driven Diffusion and PDEs in Image Processing*, 7(3) :336–344, March 1998.

[FK99] H. Frigui and R. Krishnapuram. A robust competitive clustering algorithm with applications in computer vision. *IEEE Pattern Analysis and Machine Intelligence*, 21(5) :450–465, Mai 1999.

[FS89] Y. Fukuyama and M. Sugeno. A new method of choosing the number of clusters for the fuzzy c-mean method. In *Proc. 5th Fuzzy System Symposium*, pages 247–250, 1989.

[Fua95] P. Fua. Reconstructing complex surfaces from multiple stereo views. In *International Conference on Computer Vision*, Boston, june 1995.

[GG89] I. Gath and A.B. Geva. Unsupervised optimal fuzzy clustering. *PAMI*, 11(7) :773–781, july 1989.

[Gou97] F. Goulette. *Construction algorithmique de modèles CAO à partir d'images télémétriques*. PhD thesis, ENSMP, 1997.

[Gun78] R. Gunderson. Application of fuzzy isodata algorithms to star-tracker pointing systems. In *Proc. 7th Triannual World IFAC Congr.*, pages 1319–1323, Helsinki, 1978.

[Gus79] D.E. Gustafson. Fuzzy clustering with a fuzzy covariance matrix. In *proc. IEEE CDC*, pages 761–766, Janvier 1979.

[HB93] N. Ben Hajel-Boujemaa. *Modélisation floue de l'incertitude pour la segmentation d'images*. PhD thesis, Paris 5, 1993.

[HDD+92] H. Hoppes, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganised points. In *SIGGRAPH'92 Proceedings*, pages 71–78, July 1992.

[Her91] Matthieu Herrb. *Vision en mouvement pour la robotiquue mobile*. PhD thesis, Université Paul Sabatier, Toulouse, 1991.

[HGB93] Song Han, D.B. Goldgof, and K.W. Bowyer. Using hyperquadrics for shape recovery from range data. In *International Conference on Computer Vision(ICCV'93)*, pages 492–496, 1993.

[HJ91] Richard L. Hoffman and Anil K. Jain. Segmentation and classification of range images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 9(5) :608–620, september 1991.

[HJBJ+96] A. Hoover, G. Jean-Baptiste, X. Jiang, P.J. Flynn, H. Bunke, D.B. Goldgof, K. Bowyer, D.W. Eggert, A. Fitzgibon, and R.B. Fisher. An experimental comparison of range image segmentation algorithms. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(7) :673–689, july 1996.

[Jar83] R.A. Jarvis. A perspective on range finding techniques for computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(2) :122–138, 1983.

[Jar93] R.A. Jarvis. *Three-Dimensional Object Recognition Systems*, chapter Range Sensing for computer vision. Elsevier, 1993.

[JB94] X.Y. Jiang and H. Bunke. Fast segmentation of range images into planar regions by scan line grouping. *Machine Vision and Application*, 7(2) :115–122, 1994.

[JDY95] J-D.Boissonat and Mariette Yvinec. *Géométrie algorithmique*. EDISCIENCE international, 1995.

[KFN95] R. Krishnapuram, H. Frigui, and O. Nasraoui. Fuzzy and possibilistic shell clustering algorithms and their application to boundary detection and surface approximation. *IEEE Trans. Fuzzy Syst.*, 3(1) :29–60, february 1995.

[KS00] Klaus Köster and Michael Spann. Mir : An approach to robust clustering - application to range image segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22(5) :430–443, may 2000.

[Leo91] J.C. Leon. *Modélisation et construction de surfaces pour la CFAO*. Hermès, 1991.

[Les99] P. Lescure. *Analyse et traitement d'images biologiques, microscopiques trichromes - Lien entre couleur et épaisseur*. PhD thesis, Université Paris 5, 1999.

[LGCS99] N. Loménie, L. Gallo, N. Cambou, and G. Stamon. Structuration plane d'un nuage de points 3d non structuré et détection des zones d'obstacle. In *Vision Interface*, pages 164–171, 1999.

[LGCS00a] N. Loménie, Laurent Gallo, N. Cambou, and G. Stamon. Filtrage morphologique de formes par représentation par triangulation de delaunay. In *Vision Interface*, pages 122–127, Montréal, Canada, mai 2000.

[LGCS00b] N. Loménie, Laurent Gallo, N. Cambou, and G. Stamon. Morphological operators on representations based on delaunay triangulation. In *Proc. International Conference on Pattern Recognition*, pages 556–559, Barcelone, Espagne, september 2000.

[LR83] G. Libert and M. Roubens. New experimental results in cluster validity of fuzzy clustering algorithms. In : J. Janssen et J.F. Macrotorchino et J.M Proth (Eds.), New Trends in Data Analysis and Applications, North-Holland, Amsterdam, 1983.

[Luo91] W. Luo. *Utilisation de modèles de surfaces en stéréovision*. PhD thesis, ENST, 1991.

[Lut93] Evelyne Lutton. 3d model based stereo reconstruction using coupled markov random fields. Technical Report 1951, INRIA, 1993.

[Mal01] Anthony Mallet. *Localisation d'un robot mobile en environnements naturels : méthodes et intégration*. PhD thesis, Institut National Polytechnique de Toulouse/ LAAS, 2001.

[MC99] E. Marchand and François Chaumette. Active vision for complete scene reconstruction and exploration. *IEEE Pattern Analysis and Machine Intelligence*, 21(1) :65–72, Janvier 1999.

[Mel97] M. Melkemi. A-shapes and their derivatives. In *13th ACM Symposium on Computational Geometry*, pages 367–369, June 1997.

[MLT00] G. Medioni, M.S. Lee, and C.K. Tang. *A computational framework for segmentation and grouping*. Elsevier, 2000.

[Mon96] L. Monchal. *Reconstruction de primitives géométriques tridimensionnelles à l'aide d'une séquence d'images*. PhD thesis, Université Paris 6, 1996.

[MSS91] D. Meyers, S. Skinner, and K. Sloan. Surfaces from contour : the correspondance and branching problems. In *Proceedings of graphic Interface'91*, pages 246–254, June 1991.

[Mur91] S. Muraki. Volumetric shape description of range data using "blobby model". *Computer Graphics(SIGGRAPH'91 Proceedings)*, 25(4) :227–235, July 1991.

[OZ00] S.H. Ong and X. Zhao. On post-clustering evaluation and modification. *Pattern Recognition Letters*, 21 :365–373, 2000.

[PB95] N.R. Pal and J.C. Bezdek. On cluster validity for the fuzzy c-means model. *IEEE Trans. Fuzzy Syst.*, 3(3) :370–379, 1995.

[PB97] N.R. Pal and J.C. Bezdek. Correction to on cluster validity for the fuzzy c-means model. *IEEE Trans. Fuzzy Syst.*, 5(1) :152–153, 1997.

[PP99] D.L. Pham and J.L. Prince. An adapative fuzzy c-means algorithm for image segmentation in the presence of intensity inhomogeneities. *Pattern Recognition Letters*, 20 :57–68, 1999.

[PPDX97] D.L. Pham, J.L. Prince, A.P. Dagher, and C. Xu. An automated technique for statistical characterization of brain tissues in magnetic resonance imaging. *International Journal on Pattern Recognition and Artificial Intelligence*, 11 :1189–1211, 1997.

[Pra87] V. Pratt. Direct least-squares fitting of algebraic surfaces. *Computer Graphics(SIGGRAPH'87 Proceedings)*, 21(4) :145–152, July 1987.

[PT95] L. Piegl and W. Tiller. *The NURBS book*. Springer, 1995.

[Ran92] Sabine Randriamasy. *Segmentation descendante coopérative en régions de paires d'images stéréoscopiques*. PhD thesis, Paris 9, 1992.

[Rus69] E.H. Ruspini. A new approach to clustering. *Information and Control*, 15(1) :22–32, 1969.

[RzJR98] M. Ramze Rezaee and B.P.F. Lelieveldt znd J.H.C. Reiber. A new cluster validity index for the fuzzy c-mean. *Pattern Recognition Letters*, 19 :237–246, 1998.

[SCS95] J. Serra, J. Crespo, and R.W. Schafer. Graph-based morphological filtering and segmentation. In *Proc. VI Spanish Symposium on Pattern Recognition and Image Analysis*, pages 80–87, April 1995.

[Ser87] J. Serra. *Cours de Morphologie Mathématique*. Ecole des Mines, 1987.

[Ser96] Bruno Serra. *Reconnaissance et localisation d'objets cartographiques 3D en vision aérienne dynamique*. PhD thesis, Université de Nice - Sophia-Antipolis, 1996.

[Sim84] J.C. Simon. *La reconnaissance des formes par algorithmes*. Masson, 1984.

[SLM94] F. Solina, A. Leonardis, and A. Macerl. A direct part-level of range images using volumetric models. In *IEEE Conference on Robotics and Automation*, pages 2254–2259, May 1994.

[SM88] J. Serra and G. Matheron. *Image Analysis and Mathematical Morphology : Theorical Advances*, volume II. Academic Press, 1988.

[SP91] S. Sclaroff and A. Pentland. Generalized implicit functions for computer graphics. *Computer Graphics(SIGGRAPH'91 Proceedings)*, 25(4) :247–250, July 1991.

[ST92] R. Szelisky and D. Tonnesen. Surface modeling with oriented particle system. *Computer Graphics*, 26(2) :185–189, july 1992.

[Ste95] C.V. Stewart. Minpran : A new robust estimator for computer vision. *IEEE Pattern Analysis and Machine Intelligence*, 17(10) :925–938, October 1995.

[Tar96a] Jean-Philippe Tarel. *Estimation géométrique et appariement en modélisation automatique*. PhD thesis, Université Paris 9, 1996.

[Tar96b] Jean-Philippe Tarel. Reconstruction globale et robuste de facettes 3d. Technical Report 2813, INRIA, 1996.

[TL95] B. Triggs and C. Laugier. Automatic camera placement for robotic vision. In *Proc IEEE Int'l Conf. Robotics and Automation*, volume 2, pages 1732–1738, Nagoya, Japan, 1995.

[TTA95] K. Tarabanis, R. Tsai, and P.K. Allen. The mvp sensor planning system for robotic vision task. *IEEE Transactions on Robotics and Automation*, 11(1) :72–85, 1995.

[Tuc87] W. T. Tucker. Counterexamples to the convergence theorem for fuzzy isodata clustering algorithms. *Analysis of Fuzzy Information*, 3(7), 1987. J.C. Bezdek.

[US96] J.K. Udupa and S. Samarasekera. Fuzzy connectedness and object definition : Theory, algorithms and applications in image segmentation. *Graph. Models and Image processing*, 58 :246–261, 1996.

[Vin89] L. Vincent. Graphs and mathematical morphology. *Signal Processing*, 16(4) :365–388, April 1989.

[Vin92] L. Vincent. *Mathematical Morphology in Image Processing*, chapter Graph Morphology in Image Analysis, pages 171–203. Marcel-Dekker, September 1992.

[VMA86] B.C. Vemuri, A. Mitiche, and J.K. Aggarwal. Curvature-based representation of objects from range data. *Image and Vision Computing*, 2(4) :227–234, August 1986.

[Véz95] Jean-Marc Vézien. *Techniques de reconstruction globale par analyse de paires d'images stéréoscopiques*. PhD thesis, Paris 7, 1995.

[WBW89] M.P. Windham, H. Bock, and H.F. Walker. Clustering information from convergence rate. In *Proc. 2nd Conf. International Federation Classification Soc.*, page 143, Washington, 1989.

[Win81] M.P. Windham. Cluster validity for fuzzy clustering algorithms. *Fuzzy Sets Syst.*, 5 :177–185, 1981.

[Win82] M.P. Windham. Cluster validity for fuzzy c-means clustering algorithms. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 4(4) :357–363, 1982.

[XB91] X.L. Xie and G.A. Beni. A validity measure for fuzzy clustering. *IEEE. Trans. Pattern Analysis and Machine Intelligence*, 13(8) :841–847, 1991.

[XPP97] C. Xu, D.L. Pham, and J.L. Prince. Finding the brain cortex using fuzzy segmentation, isosurfaces, and deformable surfaces. In *Proceedings of the XVth International Conference on Inform. Processing in Med. Imag.*, pages 399–404, 1997.

[Zad65] L.A. Zadeh. Fuzzy sets. *Information and Control*, 8 :338–352, 1965.

Annexe A

Algorithme de triangulation de Delaunay 2D

Nous donnons l'algorithme itératif de construction du maillage d'un nuage de points 2D sous la forme de sa triangulation de Delaunay. Le coeur de l'algorithme réside dans les structures de données utilisées.

Ainsi, l'algorithme procède par insertions successives de points P. Le point P à insérer doit être associé au triangle de la tiangulation en cours qui le contient. Puis, de nouveaux triangles sont construits ayant pour sommet P et les sommets du triangle associé. Ce dernier triangle doit être invalidé pour la triangulation finale mais conservé dans une structure pour les besoins algorithmiques de recherche du triangle contenant un prochain point à insérer.

En effet, cette recherche procède par parcours d'un arbre D contenant tous les triangles créés. Ainsi, un noeud de l'arbre correspond à un triangle créé et ses fils aux triangles créés à partir de celui-ci lors de l'insertion d'un point contenu dans ce triangle père. La racine de cet arbre est un triangle virtuel englobant l'ensemble de points 2D à trianguler. Ainsi, à la fin de l'algorithme, la triangulation de Delaunay du nuage de point S est l'ensemble des triangles feuilles de l'arbre D. Parallèlement, on doit conserver pour chaque arête créée, l'information permettant de retrouver rapidement les triangles adjacents. Cette information est stockée dans une liste H des arêtes créées. Chaque élément de cette liste doublement chaînée contient deux pointeurs vers les deux triangles, noeuds de D, bordés par l'arête en question. A cette fin, chaque arête possède deux sens, c'est-à-dire qu'elle possède une arête jumelle. Ainsi l'arête AB est liée à sa jumelle l'arête BA. On affecte à

l'arête AB le triangle adjacent situé à droite de l'arête selon le sens de parcours AB et à l'arête BA l'autre triangle adjacent. On oriente ainsi les triangles.

On a donc une structure de liste doublement chaînée H dont les éléments appelés *HalfEdge* sont des structures contenant :

- int Origin ;
- HalfEdge *Twin ;
- Face *Adjacent ;
- HalfEdge *Next,*Last ;

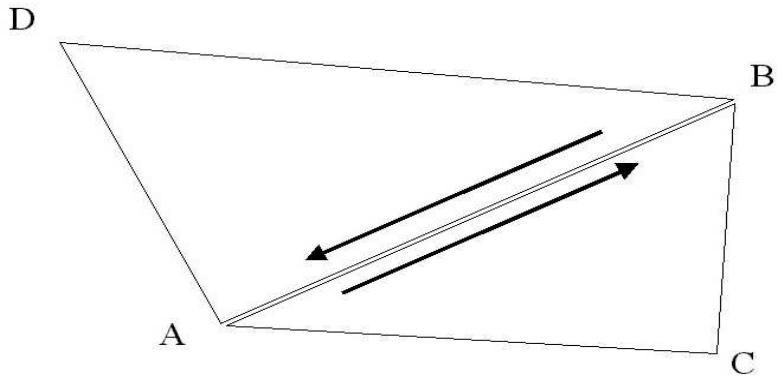

FIG. A.1 – Illustration des éléments de la structure HalfEdge : l'arête AB d'origine A est liée à sa jumelle BA et à la face ABC

On a également en parallèle un arbre D dont les noeuds sont des structures appelées *Face* et contenant :

- HalfEdge *Edges[3] ;
- Face *Fils[3] ;

A chaque création d'une nouvelle face, on vérifie si ce triangle est valide, c'est-à-dire que pour chaque arête AB, BC et CA, on teste si la nouvelle association de triangles PAB - QAB par exemple appartient à la triangulation de Delaunay. Pour cela, on teste si le disque circonscrit à PAB ne contient pas le point Q. Si tel est le cas, on bascule l'arête AB vers l'arête PQ pour créer deux nouveaux triangles qui eux appartiendront à la triangulation de Delaunay finale.

A la fin de l'algorithme, il ne reste plus qu'à supprimer tous les triangles partageant un sommet avec le premier triangle virtuel englobant correspondant à la

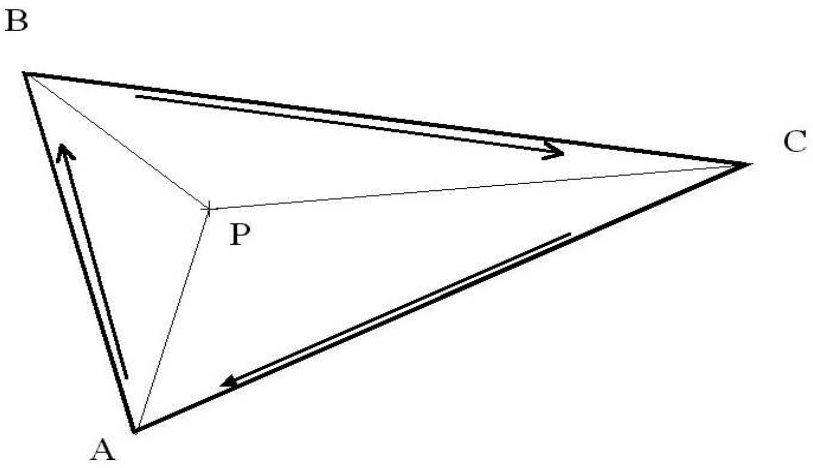

FIG. A.2 – Illustration des éléments de la structure Face : la face de sommets A,B et C est liée à ses trois arêtes AB, BC, et CA et à ses faces Fils si elles existent PAB, PBC, PCA

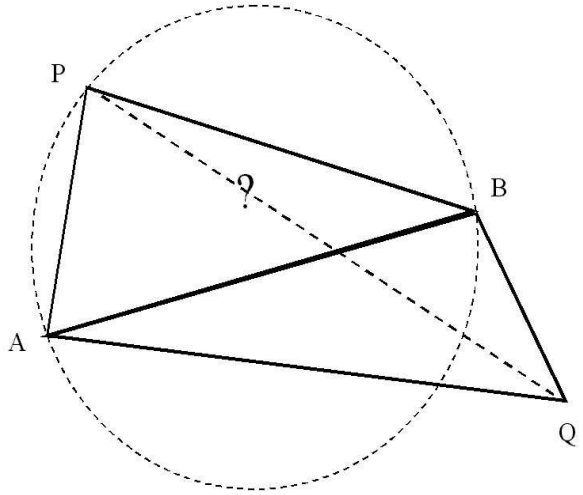

FIG. A.3 – Illustration de la procédure de validation de l'arête AB lors de la création du nouveau triangle PAB : si le cercle circonscrit au triangle PAB ne contient pas le point Q, l'arête est valide ; sinon, on doit basculer l'arête AB vers l'arête PQ

racine de l'arbre D.

Algorithm 1 Algorithme de Triangulation de Delaunay

Require: S est un ensemble de n points P_q pour q de 0 à n-1

Require: Del un arbre

Require: H une liste

Construire le triangle englobant en ajoutant les trois points P_n,P_{n+1} et P_{n+2} de coordonnées $(0, 0, 3M)$, $(0, 3M, 0$ et $(3M, 0, 0)$ où M est la coordonnées maximale de l'ensemble des points de S

Affecter ce triangle à la racine de Del et initialiser la liste H avec les arêtes ainsi créées

for all i tel que $0 \leq i \leq n-1$ **do**

Déterminer le triangle feuille de Del qui contient le point P_i, soit $P_qP_rP_s$

Ajouter à Del les trois triangles $P_iP_qP_r$, $P_iP_rP_s$ et $P_iP_sP_q$ et modifier les structures Del et H en conséquence

LEGALISER(P_qP_r)

LEGALISER(P_rP_s)

LEGALISER(P_sP_q)

end for

Oter les triangles partageant un sommet avec le triangle virtuel englobant

Cet algorithme ne donne pas de façon certaine l'enveloppe correcte de la triangulation de Delaunay, c'est-à-dire une enveloppe convexe. Il convient d'effectuer un traitement supplémentaire qui consiste à suivre les arêtes de l'enveloppe de la triangulation obtenue et de réparer localement la triangulation en ajoutant des triangles de bord si nécessaire. En pratique, si le triangle englobant est choisi suffisamment grand, ce problème ne doit pas se poser.

Algorithm 2 Procédure LEGALISER

Require: Une arête P_qP_r

Récupérer les deux triangles adjacents à P_qP_r, soient P_t et P_i les deux nouveaux sommets

if ILLEGAL(P_qP_r, P_i, P_t) **then**

Remplacer les deux triangles $P_qP_rP_t$ et $P_qP_rP_i$ par les deux triangles $P_tP_qP_i$ et $P_iP_rP_t$ et modifier les structures Del et H en conséquence

LEGALISER(P_qP_t)

LEGALISER(P_rP_t)

end if

Algorithm 3 Fonction ILLEGAL

Require: Une arête P_qP_r et deux sommets P_i et P_t

if P_t est contenu dans le cercle circonscrit à $P_iP_qP_r$ **then**

Retourner 1

else

Retourner 0

end if

Annexe B

Algorithme CMFE des C-moyennes floues exponentielles

L'algorithme d'initialisation utilise la distance euclidienne d.

Algorithm 4 INITIALISATION

Choisir les centroïdes initiaux V_i (prototypes)

Calculer les coefficients d'appartenance de chaque point 3D à chaque groupement :

$$u_{ij} = \frac{|\frac{1}{d^2(X_j,V_i)}|}{\sum_{k=1}^{K}|\frac{1}{d^2(X_j,V_k)}|} \tag{B.1}$$

Calculer les nouveaux centroïdes $\hat{V}_i$: $\hat{V}_i = \frac{\sum_{j=1}^{N}(u_{ij})^2 X_j}{\sum_{j=1}^{N}(u_{ij})^2}$ et mettre à jour les valeurs des coefficients d'appartenance u_{ij} à $\hat{u}_{ij}$ selon l'équation B.1

si $max_{ij}\{|u_{ij} - \hat{u}_{ij}|\} \leq \epsilon$ arrêter, sinon retourner à l'étape précédente où ϵ est un critère de terminaison compris entre 0 et 1

Puis l'algorithme *CMFE* utilise la *distance exponentielle* d_e définie par :

$$d_e^2(X_j, V_i) = \frac{[det(F_i)]^{1/2}}{P_i} exp[(X_j - V_i)^T F_i (X_j - V_i)/2] \tag{B.2}$$

où

$$P_i = \frac{1}{N}\sum_{j=1}^{N} u_{ij} \tag{B.3}$$

et

$$F_i = \frac{\sum_{j=1}^{N} u_{ij}(X_j - V_i)(X_j - V_i)^T}{\sum_{j=1}^{N} u_{ij}} \tag{B.4}$$

F_i étant la matrice de covariance floue du i^{me} groupement et P_i la probabilité *a priori* de sélectionner le i^{me} groupement.

Algorithm 5 CMFE

INITIALISATION des V_i et des u_{ij}

Calculer la matrice F_i et la probabilité P_i initiales pour chaque groupement

Calculer les coefficients d'appartenance de chaque point 3D à chaque groupement :

$$u_{ij} = \frac{|\frac{1}{d_e^2(X_j, V_i)}|}{\sum_{k=1}^{K} |\frac{1}{d_e^2(X_j, V_k)}|} \tag{B.5}$$

Calculer les nouveaux centroïdes $\hat{V}_i$:

$$\hat{V}_i = \frac{\sum_{j=1}^{N} (u_{ij})^2 X_j}{\sum_{j=1}^{N} (u_{ij})^2} \tag{B.6}$$

et mettre à jour les valeurs des coefficients d'appartenance u_{ij} à $\hat{u}_{ij}$ selon l'équation B.5, ainsi que les nouvelles valeurs de F_i et P_i

si $max_{ij}\{|u_{ij} - \hat{u}_{ij}|\} \leq \epsilon$ arrêter, sinon retourner à l'étape précédente où ϵ est un critère de terminaison compris entre 0 et 1

Index

ACP, 60
Appariement
    primitives, 38

C-Moyennes, 111
C-moyennes, 84
Champs de Markov, 48
Classification, 60, 129
Clustering, 59
CMFE, 82, 84, 111, 112, 123, 134, 139, 143, 145
Complexe-α, 95
Complexité, 99, 143
Convergence, 74

Delaunay, 92
    complexe, 91, 92
    triangulation, 89, 92
Disparité, 39
Dissimilarité, 63
DMP, 84, 115, 118, 126, 132

Enveloppe
    α, 93
    convexe, 93
Enveloppe convexe, 92

Filtrage
    Forme-α, 131
    Voronoï, 90
Forme-α, 89, 93, 106, 120
    dilatation, 90, 99
    dérivés, 95
    ouverture, 90, 99, 120, 145
    érosion, 90, 98, 103

Graphe Global de Courbure, 45
Géométrie différentielle, 44

Heuristique, 118, 120, 123, 129

ISODATA, 72

LAMA, 139, 147

Mahalanobis, 69, 79
Minkowski, 69
Morphologie mathématique, 97, 101, 105
    α-ouverture, 99
    α-équation, 99
    α-érosion, 98
    dilatation, 105, 106
    érosion, 105, 106

Nuées dynamiques, 62, 89, 111

Qualité de partitionnement, 75, 76

Reconnaissance des formes, 70

Segmentation, 36
SOISIC, 147
Sphère gaussienne, 44

Stirling, 65
Système de particules, 52

Taxonomie, 66
Théorie du flou, 70, 71
    partition, 73

Univers plan, 35, 37

Voronoï, 90–92
    diagramme, 90, 111

Table des figures

1.1 Système visuel stéréoscopique humain 16

2.1 Illustration de l'indétermination de la position 3D du point projeté le long de la droite liant le centre optique du dispositif optique et le point projeté 22
2.2 Modèle de système stéréoscopique de caméras en géométrie rectifiée 23
2.3 Illustration de la notion de disparité et de la mise en correspondance entre deux images 24
2.4 Banc stéréo monté sur le robot LAMA 25
2.5 Capteur laser SOISIC de la société MENSI 26
2.6 (a) vue gauche d'une scène artificielle (b) carte de profondeur calculée 28
2.7 Les différents types d'organisation de nuage de points 29
2.8 Visualisation par octree de données volumiques 30
2.9 Structure tétraédrique de Delaunay en 3 dimensions 31
2.10 Illustration d'une visualisation par surface NURBS 33

3.1 Un exemple de nuage de points 3D désorganisé et la scène observée par le capteur 36
3.2 Reconstruction 3D d'une scène d'intérieur par appariement stéréoscopique de régions planes 39
3.3 (a) Image d'intensité 3D d'une scène routière image (b) contours 2D extraits (c) vue 3D de la scène reconstruite 45
3.4 Repère de Darboux 47
3.5 Sphère gaussienne et primitive géométrique 48
3.6 Graphe global de Courbures et Primitives géométriques 49

3.7 Classification des éléments de surface en fonction des valeurs de courbure moyenne H et gaussienne K 50
3.8 Passage d'un ensemble de points non organisé de points (a) aux arbres d'escarpement extrémaux associés (d) et (e) en passant par la construction d'un graphe de voisinage Γ (b) et l'inférence des propriétés locales permettant l'association d'une énergie de déformation à chaque arête de Γ (c) 51
3.9 (a) Ensemble non organisé de 1059 points saisis à la surface d'un objet réel (Technodigit) (b) Arbre d'escarpement minimal (c) Arbre d'escarpement maximal (d) Résultat de la segmentation en utilisant l'information des normales 52
3.10 (a) Image de profondeur 'abw.train.1'. Segmentation de (a) par (b) l'université de Lyon 1 et la méthode de Raphaëlle Chaine (c) l'université de South Florida (d) l'université de Berne (e) l'université d'Edinbourg 53

4.1 Jeu de données test (a) groupements de nature linéaire (b) groupements de type compacts mais de densités en points différentes d'un facteur d'environ triple (c) groupements de types mixtes (d) groupements de contours d'ellipses 80
4.2 Regroupement en deux classes par la méthode des C-Moyennes floues utilisant la distance euclidienne sur les (a) groupements de nature linéaire (b) groupements de type compacts de densité en points différentes (c) groupements de types mixtes (d) groupements de contours d'ellipses 81
4.3 Regroupement en deux classes par la méthode des C-Moyennes floues utilisant la distance de Mahalanobis normalisée sur les (a) groupements de nature linéaire (b) groupements de type compacts de densité en points différentes (c) groupements de types mixtes (d) groupements de contours d'ellipses 82
4.4 Regroupement en deux classes par la méthode des C-Moyennes floues utilisant la distance point - droite prototype sur les (a) groupements de nature linéaire (b) groupements de type compacts de densité en points différentes (c) groupements de types mixtes (d) groupements de contours d'ellipses 83

4.5 Regroupement en deux classes par la méthode des C-Moyennes floues utilisant la distance exponentielle sur les (a) groupements de nature linéaire (b) groupements de type compacts de densité en points différentes (c) groupements de types mixtes (d) groupements de contours d'ellipses 85
4.6 (a) Ensembles tests puis regroupement en 2 classes par la méthode des C-Moyennes floues et ne montrant qu'une seule des dexu classes (b) en utilisant la distance exponentielle (c) euclidienne (d) de Mahalanobis normalisée (e) à une droite 87
4.7 Regroupement par la méthode des C-Moyennes floues utilisant la distance exponentielle en (a) 2 classes (b) 3 classes (c) 4 classes (d) 5 classes (e) 6 classes (f) 7 classes (g) 8 classes (h) 9 classes . 88
4.8 Evolution des critères CP, EP et XB 89
4.9 Evolution des critères DMP, DP et Je 89

5.1 (a) Diagramme de Voronoï et (b) Triangulation de Delaunay d'un même nuage de points 93
5.2 De la gauche vers la droite : Ensemble de points initial ; Triangulation de Delaunay ; 0.1-complexe ; 0.1-forme (intérieur) ; 0.1-enveloppe (frontière) 98
5.3 Filtrage de la forme (a) par ouverture d'ordre 1 (b) 102
5.4 De haut en bas : Scène observée ; Résultat de la segmentation du nuage de points 3D représentatif de la scène à moins de 7 mètres et les α-formes associées. 103
5.5 De gauche à droite : Ensemble de points initial ; 0.1-complex ; 0.1-ouverture 104
5.6 Détails superposés de la 0.1-ouverture : en blanc le 0.1-érodé, en gris le 0.1-ouvert, et en noir le 0.1-complexe. 104
5.7 De haut en bas : 0.8-ouverture d'ordre 7 ; Dilatation géodésique des deux germes 105
5.8 Regroupement par erosions successives 106
5.9 Erosions successives d'une α-enveloppe 107
5.10 Image en niveaux de gris 109
5.11 Image de points géométriques et système de voisinage défini par la triangulation de Delaunay 110

5.12 (a) Image d'intensité binarisée (b) Image de points géométriques "binarisée" 111
5.13 Elément structurant 111
5.14 Erosion appliquée à (a) l'image d'intensité (b) à l'image de points géométriques 112
5.15 Ouverture appliquée à (a) à l'image de points géométriques (b) à l'image de points géométriques 112

6.1 Organigramme global de la stratégie de reconstruction d'un nuage de points 3D 115
6.2 Image d'une paire stéréoscopique de scène d'extérieur : on y voit apparaître une zone de rochers sur la gauche, une poubelle renversée sur la droite et un peu plus loin au fond une zone de sol montante 117
6.3 Différentes K-partitions de la scène précédente à partir du nuage de points 3D reconstruit par stéréoscopie (a) 2-partition (b) 4-partition (c) 5-partition (d) 6-partition (e) 8-partition (f) 10-partition 118
6.4 (a) 10-partition du nuage de points 3D représentant la scène précédente ; elle donne une description de la scène faisant apparaître tous les obstacles séparés : l"amas de rochers sur la gauche, la poubelle renversée sur la droite et un peu plus loin au fond la zone de sol montante (b) segmentation 2D obtenue par rétroprojection des points 3D sur l'image du capteur 119
6.5 Au-dessus 4-partition dans laquelle trois rochers de structure ellipsoïdale émergent dans un groupement de couleur noire d'une large zone planaire de couleur verte et sont rassemblés dans un seul groupement ; en-dessous l'objet posant problème isolé 122
6.6 Schématisation 2D d'un des comportements biaisés de l'algorithme CMFE : plusieurs obstacles émergeant d'une large zone de sol sont rassemblés dans le même groupement 123
6.7 Courbe décrivant le comportement de la mesure de qualité de partition DMP(K) en fonction de K 123
6.8 Différentes partitions correspondant à des maxima relatif de la courbe DMP(K) 124
6.9 Ambiguïté géométrique au niveau de l'objet de couleur bleue . . . 126

6.10 6-partition optimale après application de l'heuristique hAG et segmentation incrémentale à partir de la 5-partition précédente 127
6.11 (a) 2-partition dans laquelle un obstacle est rassemblé malencontreusement avec la zone de sol étendu qui le supporte dans un seul groupement de couleur noire (b) 3-partition donnant une description de la scène faisant apparaître cet obstacle correctement séparé des zones de sol 129
6.12 Schématisation 2D d'un des comportements biaisés de l'algorithme CMFE : un obstacle petit ou peu dense est associé à une large zone de sol 130
6.13 Illustration du comportement de l'heuristique hAT : (a) la 3-partition préliminaire (b) l'objet de la scène du haut ne respectant pas l'heuristique hAM (c) la 4-partition obtenue après application de la stratégie 130
6.14 Processus de reconstruction sous forme d'un maillage 3D d'un nuage de points correspondant à un obstacle isolé 134
6.15 Organigramme de la stratégie de reconstruction d'un nuage de points 3D en mode statique : schéma lent 138
6.16 Organigramme d'une variante de la stratégie de reconstruction d'un nuage de points 3D en mode statique : schéma rapide 139
6.17 Evolution du champ de vision du robot au fur et à mesure qu'il avance. En gris, on a représenté la zone commune aux prises de vue aux instants t' et t" 141
6.18 Configuration de boîtes englobantes de deux obstacles détectés : la distance CC' et la zone d'intersection entre les deux boîtes représentée en grisée permettent d'associer les obstacles au cours de l'exploration 143
6.19 Effet d'épis sur la reconstruction d'un contour après fusion de nuages de points 2D légèrement décalés 144
6.20 Organigramme de la stratégie de reconstruction d'un nuage de points 3D en mode exploratoire 145

7.1 Illustration des séries de scènes acquises par le robot LAMA . . . 151
7.2 Illustration des scènes de type univers-plan 152

7.3 (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté 154
7.4 (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté 154
7.5 (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté 155
7.6 (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté 155
7.7 (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté 156
7.8 (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté 156
7.9 (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté 157
7.10 (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté 157
7.11 (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté 158
7.12 (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté 158
7.13 (a) Image originale (b) Image 2D segmentée (c) (d) vues du nuage de points 3D segmenté 159
7.14 Vues du nuage de points 3D segmenté 160
7.15 Vues du nuage de points 3D segmenté 161
7.16 Vues du nuage de points 3D segmenté 161
7.17 (a) Partition préliminaire (b)(c) Partitions intermédiaires (d) Partition finale 163
7.18 (a) Image d'intensité (b) Partition préliminaire (c) Partition finale (d) Segmentation finale 164
7.19 (a) Image d'intensité (b) Partition préliminaire (c) Partition finale (d) Segmentation finale 165
7.20 (a) Image d'intensité (b) Partition préliminaire (c) Partition finale (d) Segmentation finale 166

7.21 (a) Image d'intensité (b) Partition préliminaire (c) Partition finale (d) Segmentation finale 167
7.22 (a) Image d'intensité (b) Partition préliminaire (c) Partition finale (d) Segmentation finale 168
7.23 De gauche à droite : image d'intensités et image de profondeur de abw.train.0 169
7.24 De gauche à droite : image d'intensités et image de profondeur de abw.train.1 169
7.25 Segmentation de l'image abw.train.0 (a) par notre méthodologie (b) par l'Université de Lyon 1 (c) par l'université de South Florida (d) par l'université de Berne (e) par l'université d'Edinbourg (f) . 170
7.26 Segmentation de l'image abw.train.1 (a) par notre méthodologie (b) par l'Université de Lyon 1 (c) par l'université de South Florida (d) par l'université de Berne (e) par l'université d'Edinbourg (f) . 171
7.27 Série de scènes dans une phase exploratoire du robot LAMA . . . 175
7.28 Carte d'obstacles reconstruits 176

8.1 Structure générique d'analyse de nuages de points 179
8.2 Forme typique des groupements concernés par l'heuristique hAT . 179

A.1 Illustration des éléments de la structure HalfEdge : l'arête AB d'origine A est liée à sa jumelle BA et à la face ABC 192
A.2 Illustration des éléments de la structure Face : la face de sommets A,B et C est liée à ses trois arêtes AB, BC, et CA et à ses faces Fils si elles existent PAB, PBC, PCA 193
A.3 Illustration de la procédure de validation de l'arête AB lors de la création du nouveau triangle PAB : si le cercle circonscrit au triangle PAB ne contient pas le point Q, l'arête est valide ; sinon, on doit basculer l'arête AB vers l'arête PQ 193