

**République Algérienne Démocratique et Populaire**  
**Ministère de l'enseignement supérieur et de la recherche scientifique**

**Université 20 Aout 1955 - Skikda**

**Faculté des sciences**

**Département d'informatique**

**Mémoire de fin d'étude en vue d'obtention du diplôme**  
**de Master académique - Option - : Réseaux et systèmes distribués**

**THEME**

# **Segmentation des images de profondeur**

**Réalisé par :**

- **Azzouz Mohamed Sami**
- **Bouchache Faycal**

**Encadré par :**

**Pr Mazouzi Smaine**

**Co-encadré par:**  
**Dr Azzouz Daouia**

**Session : Juin 2023**

# Dédicace

Je dédie ce modeste travail, tout d'abord à dieu le tout puissant,  
qui m'a permis d'accéder à ce niveau du savoir.

Aux deux cœurs qui m'ont toujours porté avec amour et  
tendresse et ils sont toujours là pour moi,

A mes chers parents :

A mes trois sœurs.

A tous les membres de la famille de près et de loin.

Enfin, à tous mes proches, pour l'amour et le soutien qu'ils  
m'ont témoigné lors de la rédaction de ce mémoire.

**AZZOUZ.SAMI**

# Dédicace

Je dédie ce modeste travail, tout d'abord à dieu le tout puissant, qui m'a permis d'accéder à ce niveau du savoir.

## **A mes chers parents :**

Pour leur amour inconditionnel, leur soutien indéfectible et leurs sacrifices incommensurables. Votre encouragement constant a été ma force motrice tout au long de ce parcours. Je vous suis profondément reconnaissant pour votre patience, vos conseils avisés et votre confiance en moi.

## **À mes amis :**

Pour leur présence joyeuse, leurs encouragements sans faille et leurs précieux moments de détente qui ont équilibré mon existence pendant cette période intense. Votre amitié a illuminé mes jours et rendu cette aventure mémorable.

## **À mes professeurs et mentors :**

Pour leur expertise, leur enseignement inspirant et leur passion pour la connaissance. Vous m'avez guidé avec bienveillance et m'avez poussé à repousser mes limites intellectuelles. Vos conseils ont été inestimables et ont contribué à façonner ma pensée critique.

## **À mes collègues et camarades de classe :**

Pour leur collaboration, leurs échanges stimulants et leurs discussions qui ont alimenté mon apprentissage. Votre camaraderie a rendu cette expérience académique encore plus enrichissante.

Enfin, à toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce mémoire, je vous adresse ma sincère reconnaissance. Vos encouragements, vos idées et votre soutien ont été essentiels dans la réalisation de ce projet. Merci du fond du cœur."

*F.Bouchache*

## Remerciements :

*Nos remerciements vont tout d'abord à **Dieu tout puissant** de nous avoir donné le courage, la force, la santé et la volonté pour bien terminer ce modeste travail dans les meilleures conditions.*

*Tous nos remerciements à ceux qui nous ont aidé à terminer ce travail, et en particulier :*

*Nous exprimons notre profonde reconnaissance à **Dr Mazouzi Smaine** de nous avoir encadrés dans ce travail.*

*Nous le remercions pour ses qualités humaines et professionnelles; sa grande disponibilité, son aide, sa patience, sa gentillesse, ses précieux conseils et orientation, et de nous avoir bénéficié de sa haute compétence professionnelle.*

*Nous remercions également **les membres de jury** pour l'honneur qu'ils nous ont fait en acceptant avec gentillesse de lire notre mémoire, de commenter et de juger notre travail.*

*Nos remerciements son également exprimés à **nos chers parents, nos frère et nos sœurs** pour l'aide qu'ils nous ont apporté au cours de nos études.*

*Et nous disons un grand merci à toutes les personnes qui nous ont soutenus et qui nous ont contribués de près ou de loin à l'aboutissement de ce travail.*

*Merci à tous nos amis.*

# **Segmentation des images de profondeur**

# Table des matières

## Dédicaces

## Remerciement

## Table des matières

## Table des figures

## Introduction générale.....01

### Chapitre 1: L'image de profondeur et les techniques d'acquisition 3D

1) Introduction.....	04
2) Image de profondeur.....	05
3) Les techniques d'acquisition 3D.....	05
3.1. Vision passive.....	09
3.1.1. Las techniques « Shape from X ».....	10
3.1.2. Les techniques de stéréovision.....	11
3.2. Vision active.....	13
3.2.1. Techniques radar .....	14
3.2.2. Techniques de triangulation.....	15
3.3. Vision monoculaire .....	18
3.4. Vision multioculaire .....	19
4) Conclusion .....	20

### Chapitre 2 : Segmentation d'images

1) Introduction.....	21
2) Méthodes de segmentation.....	22
2.1. Les approches basées classification de pixels.....	22

2.1.1.	Classification bayésienne.....	23
2.1.2.	Champs de Markov.....	23
2.2.	Les approches basées contour.....	25
2.2.1.	Techniques de détection du contour.....	25
2.2.2.	Méthodes des contours actifs.....	26
2.3.	Les approches basées région.....	27
2.3.1.	Technique de seuillage.....	27
2.3.2.	Techniques de fusion et de partage.....	28
2.3.3.	Segmentation d'images par croissance de région.....	28
3)	Méthodes de segmentation d'images 3D.....	28
3.1.	Les approches fondées sur l'extraction de contours.....	29
3.2.	Les approches fondées sur l'extraction de régions.....	30
3.3.	Les approches Pyramidales.....	31
4)	Conclusion.....	33

### **Chapitre 3 : De nouvelles approches pour la segmentation d'images de profondeur**

1)	Introduction.....	34
2)	Représentation des images de profondeur.....	35
2.1.	Représentation en image de relief.....	35
2.1.1.	La représentation par produit des dérivés.....	35
2.1.2.	La représentation par régression multiple.....	36
3)	Prétraitement.....	41
3.1.	Le lissage adaptatif.....	41
4)	Les méthodes de segmentations.....	41
4.1.	Segmentation adaptative de l'image de profondeur..... ;.....	42
4.2.	Segmentation adaptative de l'image de relief.....	45
5)	Conclusion.....	48

### **Chapitre 4 : Mise en œuvre et expérimentation**

1)	Introduction.....	49
2)	Outils de développement.....	49

2.1. La plate-forme du développement Delphi7.....	49
2.2. Le langage de programmation Delphi .....	51
3) Eléments du code source.....	53
3.1. Structure de données.....	53
3.2. Procédures de traitement.....	54
4) Résultats expérimentaux .....	57
a) Image ABW. Test.8.....	57
b) Image ABW. Test.3.....	59
5) Conclusion .....	63
<b><u>Conclusion générale</u></b> .....	64
<b><u>Liste des références</u></b> .....	67

# Table des figures

<u>N°Figure</u>	<u>Page:</u>
<u>1.1</u> : Taxonomie des techniques optiques d'acquisition de formes 3D.....	07
<u>1.2</u> : Estimation des profondeurs par la technique de la stéréoscopie.....	12
<u>1.3</u> : Problème de régions d'ombre.....	15
<u>1.4</u> : La technique de triangulation.....	16
<u>3.1</u> : image de profondeur.....	42
<u>3.2</u> : Organigramme de la segmentation adaptative de l'image de profondeur.....	44
<u>3-3</u> : Représentation de l'angle $\theta$ compromis entre le vecteur normal et le vecteur de la source lumineuse.....	47
<u>4.1</u> : Masque de Gauss 5*5 pour le lissage de l'image de profondeur.....	53
<u>4.2</u> : Chargement et affichage d'une image de profondeur brute.....	54
<u>4.3</u> : Lissage de l'image par un filtre Gaussien 5*5.....	55
<u>4.4</u> : Détection de contours par la courbure à la surface .....	56
<u>4.5</u> : Détection de contours par gradient des angles .....	56
<u>4.6</u> : Image niveau de gris de l'image de profondeur abw.test.8.....	57
<u>4.7</u> : Résultat de segmentation de l'image abw.test.8 avec un détecteur basé courbure.....	58
<u>4.8</u> : Etat des surfaces de l'image abw.test.8.....	59
<u>4.9</u> : Etat des surfaces de l'image abw.test.8.....	59
<u>4.10</u> : Image niveau de gris de l'image de profondeur abw.test.3.....	60
<u>4.11</u> : Résultat de segmentation de l'image abw.test.3 avec un détecteur basé courbure ....	61
<u>4.12</u> : Etat des surfaces de l'image abw.test.3.....	61
<u>4.13</u> : Etat des surfaces de l'image abw.test.3.....	62

## Résumé

Dans notre mémoire de Master, nous avons exploré la problématique de la segmentation des images de profondeur des objets polyédriques. Nous avons réalisé des travaux de recherche sur les prétraitements nécessaires pour améliorer la qualité des images de profondeur et nous avons proposé deux approches spécifiques pour la segmentation de ces images.

Notre étude décrit deux systèmes différents de segmentation d'images de profondeur. Le premier système utilise une approche simple avec une segmentation adaptative basée sur les données brutes de profondeur et la courbure de la surface avec un lissage des valeurs de profondeur. Le deuxième système se concentre sur une segmentation automatique en utilisant une représentation en image de relief avec calcul du gradient des vecteurs normaux aux surfaces.

Dans notre travail, des tests ont été réalisés avec des images réelles de la base de données ABW, et les résultats montrent que ces approches peuvent être efficaces et fiables pour interpréter les images de profondeur. Les résultats expérimentaux ont montré que la deuxième approche proposée pour la segmentation d'images de profondeur est plus efficace et plus précise, bien que la tâche reste complexe et non résolue définitivement. Il est justifié et ambitieux d'élargir cette approche pour traiter des surfaces plus complexes et aborder des problèmes plus avancés en vision par ordinateur, afin d'améliorer l'efficacité et la performance de la segmentation et d'ouvrir la voie à des applications plus avancées dans différents domaines.

## **Abstract:**

In our Master's thesis, we explored the issue of segmenting of range images of polyhedral objects. We conducted research on the necessary pre-processing steps to improve the quality of range images and proposed two specific approaches for segmenting these images.

Our study describes two different systems for segmenting depth images. The first system uses raw depth data after smoothing of depth values and calculating surface curvature. The second system focuses on automatic segmentation using relief image after adaptive smoothing to reduce noise.

In our work, tests were conducted with real images from the ABW database, and the results show that these approaches can be effective and reliable for interpreting depth images. Experimental results have shown that the second approach proposed for segmenting depth images is effective and accurate, although the task remains complex and not definitively resolved. It is justified and ambitious to extend this approach to deal with more complex surfaces and address more advanced problems in computer vision, in order to improve the efficiency and performance of segmentation and open the way to more advanced applications in different fields.

## ملخص

في رسالة ماجستير، استكشفنا مشكلة تقسيم صور عمق المتعددة الأضلاع. قمنا بعمل أبحاث عن المعالجات المسبقة اللازمة لتحسين جودة صور العمق ولقد اقترحنا طريقتين محددتين لتجزئة هذه الصور.

دراستنا تصف نظامين مختلفين لتصنيف صور العمق. يستخدم النظام الأول نهجًا بسيطًا باستخدام تصنيف مكيف يستند إلى العمق. يتركز النظام الثاني على تصنيف تلقائي باستخدام منتج صورة التضاريس بعد تقديمها لتلميع مكيف لتقليل الضوضاء.

في عملنا، تم إجراء اختبارات باستخدام صور حقيقية من قاعدة بيانات ABW، وأظهرت النتائج أن هذه النهج يمكن أن يكون فعالاً وموثوقاً في تفسير صور العمق. أظهرت النتائج التجريبية أن النهج الثاني المقترح لتشكيل صور العمق فعال ودقيق، على الرغم من أن المهمة لا تزال معقدة وغير محلولة بشكل نهائي. من الميرر والطموح توسيع هذا النهج لمعالجة أسطح أكثر تعقيداً ومواجهة مشكلات أكثر تقدماً في رؤية الحاسوب، بهدف تحسين كفاءة وأداء التشكيل وفتح المجال لتطبيقات أكثر تقدماً في مجالات مختلفة.

# Introduction générale

Actuellement, la segmentation d'images est un domaine de recherche riche et complexe, malgré la simplicité de sa définition. De nombreux travaux ont été réalisés dans ce domaine, mais il reste encore beaucoup de défis à relever. La diversité des applications et des approches mathématiques utilisées, combinée à l'évolution constante de la puissance informatique, ouvre de nouvelles possibilités pour explorer des approches autrefois trop coûteuses en termes de temps de calcul. Ainsi, la segmentation d'images reste un domaine de recherche actif et en évolution, où de nouvelles méthodes et techniques continuent d'émerger.

En vision tridimensionnelle, la segmentation d'images de profondeur devient encore plus complexe. En raison des méthodes d'acquisition d'images de profondeur complexes et peu précises, ainsi que des orientations variables des surfaces des objets, ces images sont limitées comme étant fortement bruitées par rapport aux images bidimensionnelles.

La plupart des approches proposées pour l'analyse de ce type d'images n'ont pas réussi à produire une segmentation fiable et efficace, en particulier lorsqu'elles sont appliquées à des images réelles. La segmentation d'images est essentielle pour fournir une description concise et précise du contenu des images, ce qui est crucial pour les traitements ultérieurs.

La segmentation d'image consiste à attribuer des pixels à des régions homogènes et disjointes, formant ainsi une partition de l'image. Les pixels appartenant à une même région partagent une propriété commune, qui est le critère d'homogénéité de la région.

Les méthodes de segmentation des images en profondeur se divisent généralement en deux catégories distinctes : les méthodes de segmentation par contours et les méthodes de segmentation par régions.

Dans la première catégorie, les pixels correspondant aux discontinuités de profondeur (contours de marche) ou d'orientation de surface (contours de toit) sont sélectionnés et reliés pour délimiter les régions de l'image. Les méthodes appliquées sur les contours sont connues pour être peu coûteuses en termes de calcul, mais elles sont sensibles au bruit et reçoivent des complexes de post-traitement pour le suivi et la fermeture des contours.

Les méthodes de segmentation orientées vers les régions exploitent les propriétés de surface pour regrouper les pixels qui partagent les mêmes caractéristiques en parties connexes et disjointes [Kan93, Li03, Din05, Ba06].

En comparant ces deux approches de segmentation, on constate que les méthodes orientées contour sont généralement plus stables et moins sensibles au bruit que les méthodes orientées région. Cependant, leur efficacité dépend grandement de la sélection initiale des régions d'intérêt, également appelées graines.

Dans les différentes méthodes de segmentation d'images, la plupart nécessitent des étapes de prétraitement, telles que le lissage de l'image ou le filtrage du bruit. Cependant, dans le cas d'images fortement bruitées, comme les images de profondeur, un lissage intensif du bruit peut entraîner la suppression des contours, en particulier les contours en toit et les contours lissés. Ces contours représentent des discontinuités de courbure de surface et leur détection demeure un défi majeur en segmentation d'image. Par contre, si le bruit est sous-estimé, les distorsions résiduelles dans l'image peuvent conduire à des résultats inexacts ou erronés.

Dans le cadre de notre mémoire de master, nous proposons une approche basée sur la détection de contours dans les images de profondeur. Notre étude présentera deux systèmes que nous avons développés pour atteindre cet objectif.

Le premier système que nous proposons repose sur une approche simple de segmentation adaptative de l'image de profondeur. Cette segmentation est réalisée à partir de données l'image, d'un lissage des valeurs de profondeur.

Dans le deuxième système, nous nous concentrons sur une segmentation automatique de l'image de relief. Cette segmentation est réalisée en utilisant un gradient des angles des vecteurs normaux aux surfaces de l'image après qu'elle a été soumise à un lissage adaptatif pour réduire les bruits.

Dans les deux méthodes précédentes les résultats et les traitements ont été rapides mais l'inconvénient de ces systèmes réside dans le fait que certaines régions mal lissées n'ont pas été détectées et la qualité de la segmentation dépend fortement des paramètres de segmentation.

Il est à noter que la représentation des données de profondeur par des plans locaux à chaque pixel permet d'obtenir une visualisation précise des images de profondeur en appliquant un algorithme inspiré du Ray-Tracing.

Les deux méthodes de segmentation sont discutées et comparées dans le chapitre dédié à la mise en œuvre et à l'expérimentation. Ces expérimentations ont été réalisées en utilisant des images réelles provenant de la base de données standard ABW [Hoo96]. Les résultats expérimentaux obtenus démontrent le potentiel des approches proposées pour une interprétation efficace et fiable des images de profondeur.

Le contenu de ce mémoire par chapitre est le suivant :

Dans le chapitre 1 nous présentons l'image de profondeur et nous expliquons les différentes techniques d'acquisition 3D.

Le chapitre 2 est consacré à un état de l'art sur la segmentation d'image 2D et 3D ainsi que les techniques existantes avec leur fonctionnement général, et ce pour mieux situer les méthodes sur lesquelles nous avons travaillé.

Dans le chapitre 3 nous avons expliqué et détaillé les deux méthodes utilisées pour la segmentation d'images et leur application aux images de profondeur.

Le chapitre 4 commence par la présentation des outils de réalisation et quelques éléments d'implémentation. Ensuite, ils sont présentés les résultats des expérimentations effectuées dans le cadre de ce mémoire.

Finalement, nous achevons ce mémoire par une conclusion générale dans laquelle nous résumons nos contributions et nous traçons certaines lignes directrices pour de futurs travaux.

# Chapitre 1

## L'image de profondeur et les techniques d'acquisition 3D

### 1. Introduction

Effectivement, au cours des dernières années, les avancées technologiques dans le domaine de la mesure tridimensionnelle sans contact ont ouvert de nouvelles perspectives et possibilités dans divers domaines d'application. Voici quelques exemples concrets de l'utilisation de ces avancées :

- **Conservation du patrimoine** : La numérisation 3D permet d'acquérir et de modéliser des objets culturels et historiques précieux, tels que des pièces de musée ou des sites archéologiques. Cela permet de les préserver virtuellement, de les étudier plus facilement et de les reproduire à l'identique si nécessaire.
- **Réalité virtuelle et augmentée** : L'acquisition et la modélisation 3D d'environnements réels permettent de créer des expériences immersives en réalité virtuelle ou augmentée. Par exemple, il est possible de se déplacer virtuellement dans une usine, un musée ou un site touristique sans être physiquement présent.
- **Navigation autonome** : Les données tridimensionnelles permettent de modéliser et de caractériser des terrains, ce qui est essentiel pour la navigation autonome terrestre ou la cartographie sous-marine. Les obstacles peuvent être détectés, localisés et évités grâce à ces informations précises sur l'environnement.
- **Contrôle qualité** : La mesure tridimensionnelle sans contact est utilisée pour inspecter et caractériser les défauts de surface dans diverses industries, telles que l'automobile, l'aérospatiale et l'industrie manufacturière. Cela permet d'assurer la qualité des produits et d'optimiser les processus de fabrication.

Ces exemples illustrent la diversité des applications rendues possibles grâce aux avancées dans le domaine de la mesure tridimensionnelle sans contact et du traitement des données 3D. Cette technologie continue d'évoluer et de trouver de nouvelles applications, ouvrant la voie à des innovations encore plus importantes dans un large éventail de domaines.

## 2. Image de profondeur

Les images de profondeur que vous mentionnez sont des représentations qui fournissent des informations tridimensionnelles sur les objets observés. Elles sont composées de coordonnées 3D  $(x, y, z)$  pour chaque point de la surface de l'objet. Cela signifie qu'en chaque point  $(x, y)$  dans le plan de l'image, nous avons une valeur de profondeur  $Z(x, y)$  qui indique la distance entre l'objet et le capteur.

En plus des informations de profondeur, il est également possible d'associer des informations de luminance et de chrominance à chaque point de l'image. Cela signifie que pour chaque point de coordonnées  $(x, y, z)$ , on peut également enregistrer des informations sur la couleur et la luminosité de cet emplacement.

Cette combinaison d'informations tridimensionnelles  $(x, y, z)$  avec des informations de couleur (luminance et chrominance) permet une représentation plus complète et précise des objets observés. Ces images de profondeur sont utilisées dans divers domaines, tels que la vision par ordinateur, la robotique, la réalité virtuelle, l'analyse de scènes, et bien d'autres, pour des applications telles que la segmentation d'objets, la reconstruction 3D, la détection d'obstacles, etc.

## 3. Les techniques d'acquisitions 3D

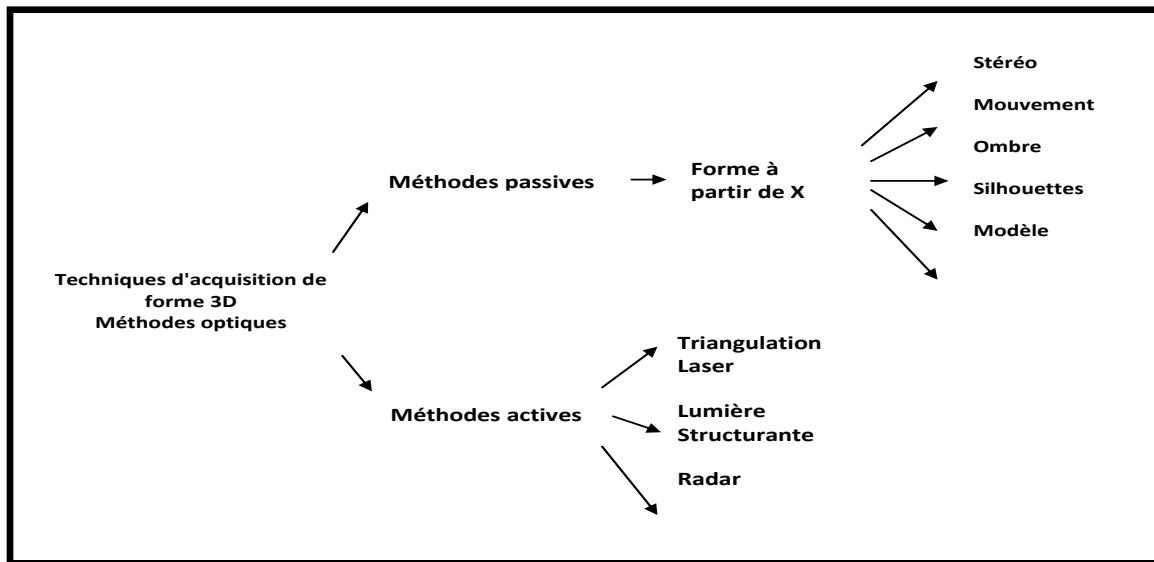
En effet, il existe différentes méthodes pour acquérir des informations tridimensionnelles sur la forme des objets. Ces méthodes sont utilisées dans divers domaines et applications, tels que la rétroprojection, le contrôle de qualité, la médecine, la réalité virtuelle, la robotique et l'archéologie.

On peut classer ces méthodes en deux catégories principales : les méthodes passives et les méthodes actives.

- **Méthodes passives** : Les méthodes passives exploitent les informations présentes dans l'environnement sans nécessiter de source de lumière ou de projection spécifique. Elles utilisent souvent des caméras ou des capteurs qui enregistrent la lumière réfléchiée par les objets pour reconstruire leur forme en 3D. Parmi les méthodes passives couramment utilisées, on retrouve la stéréovision, la photogrammétrie, la tomographie par rayons X, la tomographie par imagerie par résonance magnétique (IRM) et la numérisation laser.
- **Méthodes actives** : Les méthodes actives nécessitent l'émission d'un signal ou d'une source de lumière spécifique pour obtenir des informations tridimensionnelles. Ces méthodes impliquent souvent la mesure du temps de vol (Time-of-Flight) ou la triangulation laser. Par exemple, la triangulation laser utilise un laser pour projeter un faisceau sur l'objet, et en mesurant la déviation du faisceau réfléchi, on peut déterminer la distance entre l'objet et le capteur. Les scanners laser 3D et la Kinect de Microsoft sont des exemples de méthodes actives largement utilisées.

Chaque méthode a ses avantages et ses limites en termes de précision, de résolution, de vitesse d'acquisition, de coût et de complexité. Le choix de la méthode dépendra du contexte d'utilisation spécifique et des besoins de l'application.

La figure 1 présente cette catégorisation des approches d'acquisition de formes 3D.



**Figure 1.1 : Taxonomie des techniques optiques d'acquisition des formes 3D**

Effectivement, l'utilisation d'une seule caméra ne permet pas d'obtenir directement des informations de profondeur, sauf dans certains cas particuliers[Pam90]. Pour reconstruire des informations de profondeur à partir d'une seule caméra, il est souvent nécessaire d'utiliser des techniques de stéréovision, de structure à mouvement[Her91], ou de vision par lumière structurée.

Cependant, dans de nombreux cas, l'utilisation d'autres capteurs en complément de la caméra est nécessaire pour obtenir des informations de profondeur précises et fiables. Certains capteurs couramment utilisés pour acquérir des informations de profondeur sont :

- 1. Capteurs de profondeur :** Il s'agit de capteurs dédiés qui mesurent directement la distance entre le capteur et les objets dans la scène. Des exemples courants de capteurs de profondeur sont les caméras ToF (Time-of-Flight), les caméras à lumière structurée (comme la Kinect de Microsoft), les capteurs LiDAR (Light Detection and Ranging) et les capteurs à ultrasons. Ces capteurs sont conçus spécifiquement pour mesurer la distance et fournir des informations de profondeur.
- 2. Capteurs inertiels :** Les capteurs inertiels, tels que les accéléromètres et les gyroscopes, peuvent être utilisés pour estimer les mouvements et les changements

## Chapitre 1 :

### L'image de profondeur et les techniques d'acquisition 3D

de position de la caméra. En combinant ces informations avec des algorithmes de traitement d'image, il est possible d'obtenir des estimations de profondeur.

- 3. Capteurs de mouvement :** Certains systèmes de capture de mouvement utilisent des marqueurs ou des capteurs placés sur les objets à suivre. En analysant les changements de position et d'orientation de ces marqueurs dans l'espace, il est possible de reconstruire la structure 3D de la scène.

L'utilisation de ces capteurs en complément de la caméra permet d'améliorer la précision et la fiabilité des informations de profondeur obtenues. Cela ouvre la voie à de nombreuses applications dans des domaines tels que la réalité virtuelle, la robotique, la vision par ordinateur et bien d'autres.

Effectivement, la stéréovision binoculaire ou trinoculaire est une méthode couramment utilisée pour obtenir des informations de profondeur à partir de deux ou plusieurs caméras. Cette technique repose sur le principe de la triangulation, où la disparité entre les vues stéréoscopiques est utilisée pour estimer la distance des objets dans la scène [Aya83, Vin86, Cou91].

Les données obtenues à partir de la stéréovision sont généralement regroupées en segments ou en régions en utilisant des critères d'homogénéité tels que l'intensité, la couleur ou la texture des pixels. Ces segments ou régions représentent les variations d'éclairage, de réflectance et de texture présentes dans la scène.

Cependant, il est important de noter que l'extraction d'informations géométriques stables et précises à partir de ces données peut être difficile. Les variations d'éclairage, les ombres, les surfaces réfléchissantes ou les textures complexes peuvent affecter la précision des estimations de profondeur.

En ce qui concerne la classification des capteurs de profondeur, il est courant de les regrouper en fonction de leurs propriétés. On distingue généralement entre capteurs actifs et passifs, selon que le capteur illumine activement la scène ou non. Les capteurs actifs, tels que les caméras à lumière structurée ou les capteurs LiDAR, émettent un signal pour

## Chapitre 1 :

### L'image de profondeur et les techniques d'acquisition 3D

mesurer la distance. Les capteurs passifs, tels que les caméras stéréo ou les caméras monoculaires utilisant des techniques de traitement d'image, se basent uniquement sur les informations visuelles disponibles.

De plus, on peut également classer les systèmes de capture de profondeur comme monoculaires ou multioculaires, en fonction du nombre d'images produites par le système. Les systèmes monoculaires produisent une seule image, tandis que les systèmes multioculaires produisent plusieurs images simultanément pour permettre une meilleure estimation de la profondeur.

En somme, l'utilisation de la stéréovision et d'autres techniques de capture de profondeur permet d'obtenir des informations de profondeur à partir de données visuelles, mais il peut être difficile d'obtenir des informations géométriques précises dans des conditions complexes.

#### 3.1 Vision passive :

La vision par ordinateur vise à interpréter l'environnement en utilisant des images d'intensité capturées par une ou plusieurs caméras. Les capteurs passifs utilisés dans la vision par ordinateur mesurent l'énergie ambiante réfléchiée par la scène, ce qui signifie qu'ils dépendent de la lumière présente dans l'environnement[Fla99].

Si les conditions d'éclairage ne sont pas maîtrisées, cela peut entraîner plusieurs problèmes dans l'interprétation des images et dans la compréhension de la scène. Les variations d'éclairage peuvent affecter les niveaux d'intensité des pixels, entraînant des différences de contraste et de luminosité entre les différentes parties de l'image. Cela peut rendre plus difficile la détection des objets, la segmentation des régions et la reconnaissance des formes.

De plus, dans des situations où l'éclairage est inadéquat ou non uniforme, il peut y avoir des ombres, des reflets ou des zones surexposées qui peuvent altérer les informations visuelles et rendre la tâche d'interprétation plus complexe.

Pour remédier à ces problèmes, il est souvent nécessaire de prendre des mesures pour contrôler et stabiliser les conditions d'éclairage lors de la capture des images. Cela peut inclure l'utilisation de sources de lumière artificielle, l'ajustement des paramètres de la caméra, la calibration des capteurs, ou l'utilisation de techniques de traitement d'image avancées pour compenser les variations d'éclairage.

En résumé, les capteurs passifs utilisés dans la vision par ordinateur sont sensibles aux conditions d'éclairage, et si celles-ci ne sont pas maîtrisées, cela peut entraîner des difficultés dans l'interprétation des images et dans la compréhension de la scène. Il est donc important de prendre en compte ces facteurs lors de la conception des systèmes de vision par ordinateur et d'utiliser des techniques appropriées pour compenser les variations d'éclairage et obtenir des résultats plus fiables.

Les systèmes passifs utilisés en vision par ordinateur peuvent être classés en deux catégories principales :

#### 3.1.1 Les techniques « Shape from X »

Les techniques "Shape from X" font référence à des méthodes qui exploitent les variations de textures, d'ombres ou de contours dans une image pour obtenir des informations sur la profondeur des objets présents dans la scène. Celles-ci peuvent être obtenues, également, de la variation des formes lors des mouvements "Shape from motion" [Maz96]. Ces techniques peuvent être utilisées pour estimer la forme 3D des objets à partir d'une image 2D.

- **Shape from texture (estimation de la forme à partir de la texture) :** Cette méthode consiste à analyser les variations de texture dans une image pour estimer la profondeur et l'orientation des surfaces des objets. Elle repose sur le calcul du gradient de texture, qui permet de détecter les changements de texture et d'inférer des informations sur la structure 3D des objets. Cependant, cette méthode peut être sensible aux variations d'éclairage et aux occlusions, ce qui peut affecter sa précision.

- **Shape from shading (estimation de la forme à partir de l'ombre)** : Cette méthode exploite les informations d'ombre présentes dans une image pour reconstruire la forme 3D des objets. Elle repose sur le principe que les variations d'ombre sont causées par les variations de relief des surfaces. Pour utiliser cette méthode, il est nécessaire de connaître à priori les positions des sources de lumière ainsi que les propriétés photométriques des surfaces. Cependant, cette méthode peut être sensible aux conditions d'éclairage et aux réflexions spéculaires, ce qui peut limiter sa précision.

Ces méthodes "Shape from X" ont été largement étudiées dans le domaine de la vision par ordinateur, mais elles sont souvent considérées comme grossières et qualitatives dans leurs résultats. En raison de leur imprécision et de leurs limitations, elles sont moins utilisées dans le domaine de la vision artificielle par rapport à d'autres techniques plus avancées telles que la stéréovision ou la capture de profondeur par capteurs spécifiques.

### 3.1.2 Les techniques de stéréovision

Les techniques de stéréovision sont des méthodes passives qui utilisent des contraintes entre deux ou plusieurs caméras pour calculer les informations de distance et de profondeur d'une scène. Le principe de la stéréovision repose sur l'observation de la même scène à partir de différents points de vue, ce qui permet de trianguler les positions 3D des points dans l'espace.

Lorsqu'une scène est observée par deux caméras, chaque point de la scène est projeté sur les plans images des deux caméras. En trouvant les correspondances entre les points dans les deux images, on peut calculer les coordonnées 3D de ces points en utilisant la relation géométrique entre les plans images des deux caméras. Les caractéristiques avec de forts gradients, tels que les bords, sont souvent utilisées comme points de correspondance potentiels, et des algorithmes de recherche heuristiques sont utilisés pour trouver les correspondances appropriées.

## Chapitre 1 :

### L'image de profondeur et les techniques d'acquisition 3D

La stéréovision peut être classifiée en deux catégories selon la provenance des images : la stéréoscopie statique et la stéréoscopie dynamique.

- **La stéréoscopie statique** utilise plusieurs vues prises simultanément depuis des positions différentes. Les caméras sont fixées et les images sont acquises au même instant. Cela permet d'obtenir des informations de profondeur pour une scène statique.
- **La stéréoscopie dynamique** utilise plusieurs vues prises à des instants différents. Pour une scène fixe, la caméra est déplacée pour acquérir différentes perspectives de la scène. Pour un capteur fixe, l'objet lui-même est déplacé dans la scène. Cette approche permet de capturer des informations de profondeur pour des scènes en mouvement.

La stéréovision est largement utilisée dans de nombreux domaines, tels que la robotique, la vision par ordinateur, la réalité virtuelle, la cartographie 3D, la navigation autonome, etc. Elle permet d'obtenir des informations de profondeur précises et détaillées, ce qui est essentiel pour de nombreuses applications nécessitant une compréhension tridimensionnelle de la scène.

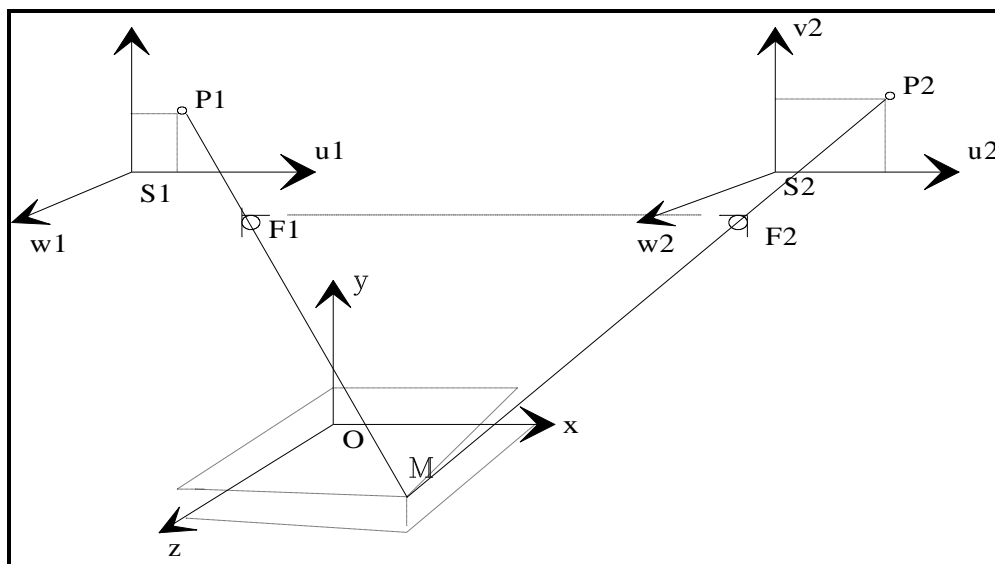


Figure 1.2: Estimation des profondeurs par la technique de la stéréoscopie

#### 3.2. Vision active

Ce qui en fait une technologie précieuse pour la numérisation tridimensionnelle. Les capteurs actifs dans le domaine de la vision utilisent une émission contrôlée d'énergie, qui peut être sous forme de lumière, de radiofréquences (radar) ou d'autres types de rayonnement, pour mesurer l'interaction de cette énergie avec la surface des objets dans la scène. En connaissant les caractéristiques de l'émetteur et du récepteur, ainsi que l'interprétation du rayonnement réfléchi, il est possible d'obtenir des informations tridimensionnelles.

Les capteurs actifs optiques, par exemple, peuvent contrôler les conditions d'éclairage de la scène et interpréter non seulement l'intensité du rayonnement réfléchi, mais également le temps de vol d'une impulsion lumineuse ou la phase de l'onde réfléchie[Fla99]. Ces informations permettent de reconstruire la géométrie tridimensionnelle de la scène.

Les systèmes actifs peuvent être classés en deux catégories principales :

- 1. Les techniques basées sur le temps de vol :** Ces capteurs, tels que les capteurs radar, mesurent le temps qu'il faut à un signal émis pour atteindre la surface de l'objet et revenir au capteur. En utilisant cette mesure de temps de vol, on peut estimer la distance entre le capteur et l'objet, ce qui permet de reconstruire la forme tridimensionnelle de la scène. Les applications des capteurs basés sur le temps de vol sont variées, allant de la robotique industrielle à la navigation des robots mobiles en extérieur.
- 2. Les techniques basées sur la triangulation :** Ces capteurs utilisent la triangulation pour estimer la distance entre le capteur et les points de la surface de l'objet. Ils émettent une lumière structurée, telle qu'un faisceau laser ou un motif de lumière structurée, sur la scène. En mesurant la déformation de ce motif sur la surface de l'objet, on peut calculer la distance et reconstruire la forme tridimensionnelle. Les capteurs basés sur la triangulation sont généralement moins coûteux, mais leur champ d'application peut être plus limité par rapport aux capteurs basés sur le temps de vol.

Les systèmes de vision active sont utilisés dans de nombreux domaines, tels que la robotique, la navigation autonome, la cartographie 3D, la réalité augmentée, la surveillance, etc. Ils offrent des capacités de mesure précises et rapides de la géométrie tridimensionnelle, ce qui les rend adaptés à de nombreuses applications nécessitant une vision en 3D.

#### 3.2.1. Techniques radar

Les techniques de radar sont utilisées dans les capteurs actifs pour obtenir des informations de profondeur. Dans ces techniques, l'objet est illuminé par une source lumineuse, généralement un laser, et l'information de profondeur est calculée à partir de l'énergie réfléchie.

La méthode la plus couramment utilisée dans les capteurs radar est la mesure du temps de vol. Étant donné que les rayons électromagnétiques se déplacent à une vitesse constante, la distance jusqu'à un point sur la surface de l'objet peut être déterminée en envoyant une impulsion lumineuse laser vers ce point et en mesurant le temps nécessaire pour que l'impulsion réfléchie parvienne au capteur. Cette méthode est particulièrement adaptée à la numérisation à longue distance.

Une autre approche consiste à illuminer continuellement le point sur la surface avec un signal laser modulé en amplitude. La différence de phase entre le signal laser incident et le signal réfléchi est alors proportionnelle à la valeur de profondeur recherchée[Fla99]. Cette méthode permet également d'obtenir des informations de profondeur.

Les capteurs radar offrent une grande précision et sont utilisés dans de nombreux domaines, tels que la télédétection, la navigation, la surveillance, etc. Ils permettent d'obtenir des informations de profondeur précises sur des objets à différentes distances.

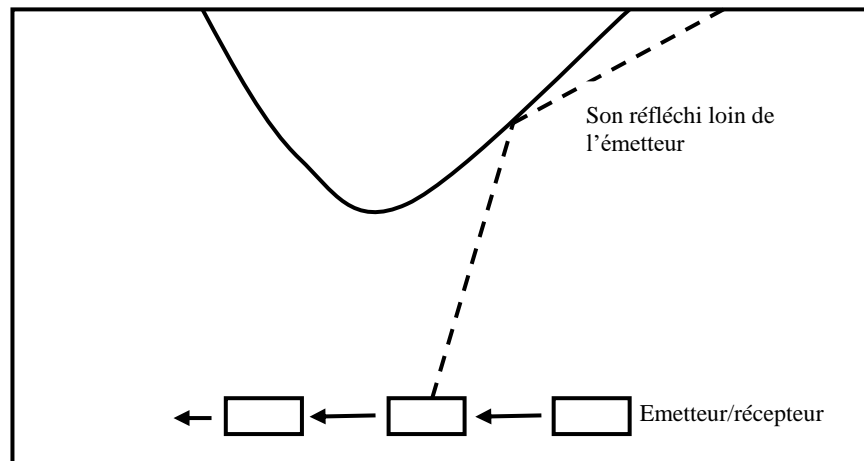


Figure 1.3 : Problème de régions d'ombre

### 3.2.2. Techniques de triangulation

Les capteurs basés sur la triangulation sont moins coûteux à construire que les capteurs radar, car ils nécessitent simplement une caméra et un projecteur de lumière. Cependant, le logiciel utilisé pour obtenir les informations de distance peut être plus complexe.

Le principe de la triangulation est le suivant : un faisceau lumineux ponctuel est projeté sur l'objet, puis on étudie l'intersection du faisceau avec l'objet pour calculer les coordonnées des points. En connaissant la position de la source lumineuse par rapport à l'observateur (la caméra), il est facile de déterminer la distance du point considéré. Souvent, un processus de calibration est utilisé pour obtenir la relation géométrique précise entre les différentes composantes du système.

Un capteur de triangulation typique comprend une caméra qui observe la scène et un projecteur de lumière qui émet un faisceau lumineux ponctuel sur l'objet. En mesurant l'emplacement du point d'intersection du faisceau lumineux sur l'objet dans l'image capturée par la caméra, la distance du point par rapport au capteur peut être déduite.

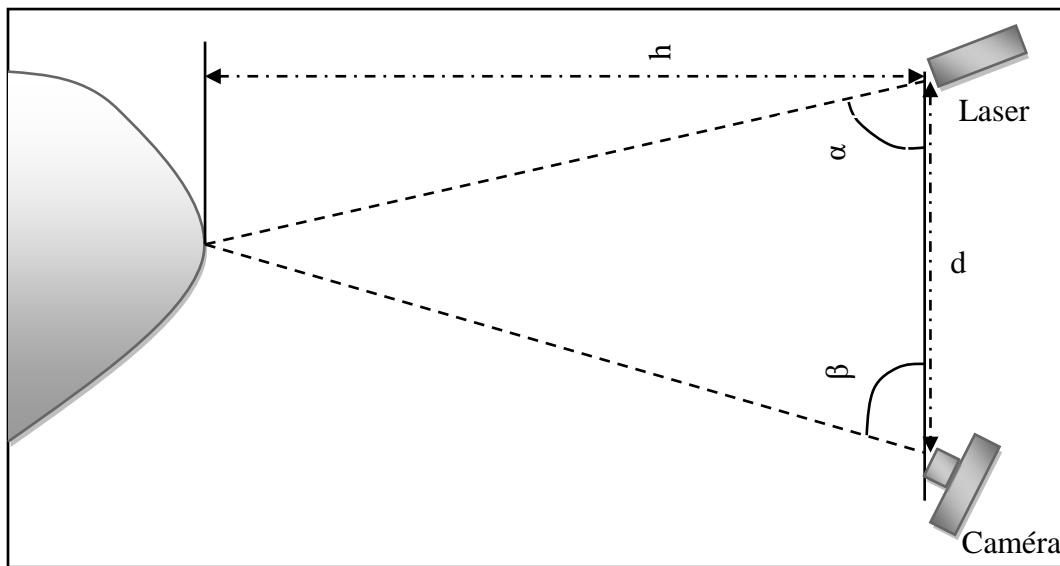
Cette technique de triangulation permet d'obtenir des informations de distance précises pour les objets dans la scène. Elle est couramment utilisée dans des applications

## Chapitre 1 :

### L'image de profondeur et les techniques d'acquisition 3D

telles que la mesure dimensionnelle, le contrôle de qualité, la vision industrielle, et d'autres domaines où une précision élevée est requise.

Il convient de noter que bien que les capteurs basés sur la triangulation soient moins coûteux que les capteurs radar, ils ont une portée limitée et sont plus sensibles aux conditions d'éclairage et aux variations de texture de la surface de l'objet. Un diagramme simple d'un capteur de triangulation apparaît sur la figure [Fla99].



**Figure 1.4: La technique de triangulation**

La distance  $h$ , d'un point de l'objet, par rapport au plan de balayage est déterminé comme suit :

$$h = \frac{d}{C \tan(\alpha) + C \tan(\beta)}$$

Les capteurs par triangulation peuvent être classés en fonction du type de lumière projetée [Fla99]. Voici deux catégories courantes :

**1. Projection d'un point :** Dans cette configuration, un faisceau lumineux, généralement émis par un laser, est utilisé pour illuminer un point spécifique sur la surface de l'objet. Un détecteur monodimensionnel, tel qu'une barrette de photodiodes, est suffisant

## Chapitre 1 :

### L'image de profondeur et les techniques d'acquisition 3D

pour capturer le signal réfléchi. À chaque acquisition, un point unique est numérisé sur la surface de l'objet. Cette méthode est simple et efficace pour mesurer la distance d'un point spécifique.

**2. Projection d'un trait :** Au lieu d'un point, un trait lumineux est projeté sur la surface de l'objet. Ce type de capteur est potentiellement plus rapide que les capteurs à projection de point, car lors de chaque acquisition, l'ensemble du trait lumineux est numérisé. Cela permet de couvrir une zone plus large de l'objet en une seule acquisition. Pour cette configuration, un détecteur bidimensionnel est nécessaire, généralement une matrice CCD, pour capturer l'image du trait lumineux réfléchi. Cette méthode peut être utilisée pour obtenir une cartographie plus rapide de la surface de l'objet.

Chaque type de projection (point ou trait) présente ses propres avantages et inconvénients en termes de précision, de vitesse et de coût. Le choix entre ces deux configurations dépend des exigences spécifiques de l'application et des contraintes du système.

Effectivement, une autre catégorie de capteurs par triangulation est la projection d'un patron de lumière structurée. Dans cette configuration, un motif lumineux spécifique, tel qu'un nuage de points, des traits parallèles ou des grilles, est projeté sur la surface de l'objet[Har90]. Lorsque ce motif est capturé par une caméra, il permet d'échantillonner immédiatement toute la surface de l'objet.

Cette méthode offre l'avantage de permettre une acquisition rapide de l'ensemble de la surface, ce qui peut être particulièrement utile dans des applications telles que la numérisation 3D et la reconnaissance de formes. Cependant, elle présente des défis en termes d'interprétation des données et peut être intrinsèquement moins précise que les autres méthodes de triangulation.

L'interprétation des données consiste à associer chaque point du motif lumineux projeté à un point correspondant sur la surface de l'objet. Cela peut être réalisé en utilisant des techniques de correspondance et de reconstruction 3D. Cependant, les variations d'éclairage, les ombres et les réflexions peuvent rendre cette tâche complexe.

## Chapitre 1 :

### L'image de profondeur et les techniques d'acquisition 3D

En termes de précision, la projection d'un patron de lumière structurée peut être moins précise que les méthodes de projection d'un point ou d'un trait. Cela est dû à des facteurs tels que la résolution de la caméra, la calibration du système et les erreurs liées à l'interprétation des données.

Malgré ces limitations, la projection d'un patron de lumière structurée reste une méthode populaire pour la numérisation 3D et d'autres applications où une acquisition rapide de la surface de l'objet est nécessaire.

### 3.3. Vision monoculaire :

La vision monoculaire est une technique qui utilise un seul capteur pour capturer une seule image de la scène. L'objectif est d'interpréter cette image afin d'extraire des informations de profondeur sur les objets de la scène[Fla99].

Cependant, la vision monoculaire présente certaines limitations par rapport aux autres techniques d'acquisition 3D. Étant donné qu'une seule image est utilisée, il est souvent nécessaire d'avoir des points de vue complémentaires pour reconstruire l'intégralité de l'objet. Cela peut être réalisé en utilisant des méthodes de stéréovision ou en analysant une séquence d'images.

Les méthodes de vision monoculaire, regroupées sous le terme de "forme à partir de X", ne sont généralement pas aussi précises que les autres méthodes d'acquisition 3D. Elles sont davantage utilisées pour des applications telles que la reconnaissance des formes et d'autres tâches où une description qualitative ou approximative de la scène est suffisante.

L'une des principales raisons de la limitation de précision de la vision monoculaire est le manque d'informations directes sur la profondeur dans une seule image. L'estimation de la profondeur à partir d'une image seule est une tâche complexe qui nécessite des hypothèses et des techniques sophistiquées. Cela peut entraîner des résultats moins précis par rapport aux méthodes de stéréovision ou aux capteurs actifs.

Malgré ces limitations, la vision monoculaire reste une technique couramment utilisée dans de nombreux domaines, notamment la vision par ordinateur, la reconnaissance des formes et l'analyse d'images. Des progrès continus sont réalisés pour améliorer la précision de ces méthodes et leur permettre de fournir des informations de profondeur plus précises.

#### 3.4. Vision multioculaire

La vision multioculaire est une approche qui utilise deux ou plusieurs images pour calculer les informations de profondeur de la scène. Les systèmes binoculaires, par exemple, traitent deux images pour obtenir des informations de profondeur. Certains systèmes utilisant des capteurs radar ou laser peuvent également produire des images combinant la profondeur et l'intensité des objets.

L'utilisation d'images multiples permet d'obtenir des informations complémentaires, ce qui peut améliorer la précision de l'estimation de la profondeur. Cependant, l'un des défis principaux de la vision multioculaire réside dans l'appariement des informations visuelles entre les différentes images. Il est nécessaire d'établir des correspondances entre les points homologues des différentes vues afin de calculer la profondeur.

La stéréovision est une technique couramment utilisée dans la vision multioculaire. Elle a atteint une précision suffisante pour être utilisée dans des applications telles que la photogrammétrie (la mesure des distances et des positions à partir d'images) ou la reconstruction du relief à partir d'images aériennes ou satellitaires[Fla99]. Grâce à la combinaison des informations provenant de différentes images, la stéréovision peut fournir des estimations de profondeur plus précises que la vision monoculaire.

En résumé, la vision multioculaire utilise deux ou plusieurs images pour calculer les informations de profondeur de la scène. Les systèmes binoculaires, la stéréovision et d'autres techniques de mise en correspondance des images sont utilisés pour extraire ces informations. Cela permet d'obtenir des résultats plus précis et est utilisé dans diverses applications telles que la photogrammétrie et la reconstruction 3D.

#### 4. Conclusion

Dans ce chapitre, nous avons examiné les différentes techniques d'acquisition d'images tridimensionnelles. Nous avons présenté les principales méthodes relevant des deux approches d'acquisition : les techniques passives et les techniques actives. Nous nous sommes concentrés particulièrement sur les techniques actives, car ce sont les images produites par ces dispositifs que nous utiliserons dans notre travail pour expérimenter les approches de segmentation.

Dans le chapitre suivant, nous allons présenter un état de l'art des méthodes de segmentation d'images les plus répandues. La segmentation d'images consiste à diviser une image en régions ou en objets significatifs, afin de faciliter leur analyse et leur compréhension. Nous examinerons différentes approches de segmentation, telles que les méthodes basées sur les seuils, les méthodes de regroupement (clustering), les méthodes de détection de contours, ainsi que les méthodes plus avancées utilisant des techniques d'apprentissage automatique et de vision par ordinateur.

L'objectif de ce chapitre sera de fournir un aperçu des techniques de segmentation d'images les plus couramment utilisées, en mettant en évidence leurs avantages, leurs limites et leurs domaines d'application. Cela nous permettra de choisir la méthode la plus appropriée pour notre travail de segmentation d'images tridimensionnelles basées sur les données acquises à l'aide des techniques actives.

En résumé, le chapitre précédent nous a permis de comprendre les différentes techniques d'acquisition d'images tridimensionnelles, en mettant l'accent sur les techniques actives. Dans le prochain chapitre, nous aborderons l'état de l'art des méthodes de segmentation d'images, afin de sélectionner la méthode la plus adaptée à notre travail de segmentation d'images tridimensionnelles.

# Chapitre 2

## Segmentation d'image

### 1. Introduction

La segmentation d'images revêt une grande importance dans l'analyse et le traitement des images. Il s'agit d'une opération de base qui consiste à diviser une image en régions  $R$ , qui sont des ensembles de pixels appartenant à une même structure telle que des objets ou une scène.

La segmentation est un sujet fascinant dans le domaine du traitement d'images. C'est une étape cruciale dans la plupart des systèmes de vision par ordinateur. Selon la définition de [Coc95], la segmentation est un processus de bas niveau qui consiste à diviser l'image  $A$  en sous-ensembles  $R_i$ , appelés régions. Chaque région est constituée de pixels connexes partageant des caractéristiques communes qui les distinguent des pixels des régions voisines. Cette partition doit satisfaire les critères suivants : aucune région ne doit être vide, l'intersection entre deux régions doit être vide et l'ensemble des régions doit couvrir l'intégralité de l'image.

Malgré les nombreuses méthodes proposées, la quête d'une méthode optimale pour la segmentation d'images reste un défi dans le domaine de l'analyse et du traitement d'images.

Dans ce chapitre, nous présentons un bref survol des techniques existantes en donnant leur fonctionnement général, afin de mieux situer dans l'état de l'art actuel les méthodes sur lesquelles nous avons travaillé et il est composé de deux parties :

- 1- Méthodes existantes en segmentation d'images,
- 2- Méthodes existantes en segmentation d'images 3D.

### 2. Méthodes de segmentation

Donnons d'abord une définition formelle de la segmentation d'image :

Les régions de l'image segmentée doivent satisfaire chacune des règles suivantes[Nsal17]:

- $\forall i, R_i$  est connexe
- $\forall i, j, i \neq j, R_i \cap R_j = \emptyset$
- $\cup R_i = I$  ( $I$  : image)
- $\forall i, p(R_i) = \text{vrai}$   $p$  : prédicat d'homogénéité
- $\forall i, j, i \neq j, R_i$  et  $R_j$  sont connexes, et  $P(R_i \cup R_j) = \text{faux}$

La segmentation d'image sert à :

- Fournir des régions homogènes (selon un critère donné) :
- Réduire le bruit
- Localiser de manière précise les contours des régions

Il existe beaucoup d'approches de la segmentation d'images, telle que les techniques de seuillage, les techniques de détection de contour, les approches de fusion et partage de régions, techniques basées croissance de régions, les techniques basées contours actifs, les approches basées graphes théorétiques, les techniques basées multi-résolution, le clustering, et les méthodes basées sur la couleur et la texture et ses techniques peuvent être regroupées en quatre catégories principales :

#### 2.1. Les approches basées classification de pixels

La classification d'images joue un rôle essentiel dans de nombreuses applications, notamment la quantification d'images et la segmentation d'images. Son objectif est de définir une classification des pixels en fonction de leur luminance, en utilisant les niveaux de gris présents dans l'image.

Généralement, cette approche repose sur l'utilisation d'algorithmes de classification des données afin de classer les pixels d'une image en différentes classes, tout en cherchant à optimiser un critère d'homogénéité spécifique.

Les méthodes de la classification d'images sont divisées en deux grandes catégories : les méthodes non-supervisées et les méthodes supervisées.

Dans le premier scénario, la classification est automatisée à l'aide d'algorithmes tels que k-means ou fuzzy c-means. Dans le second scénario, la segmentation est guidée par l'utilisateur, ce qui nécessite la définition de certains paramètres en amont, tels que Minimum-Distance-to-Means, Likelihood et Parallelopiped [NSal17].

#### 2.1.1. Classification bayésienne

Cette approche exploite la règle de Bayes en combinaison avec des modèles généraux décrivant les distributions des niveaux de gris des régions recherchées. Pour chaque pixel de l'image, sa probabilité d'appartenir à une classe spécifique est calculée [Lau99, Wei96]. La probabilité d'appartenir à la classe "i" étant donné un individu "x" est formulée par l'équation suivante [Lau99]:

$$p(w_i/x) = \frac{p(x/w_i) p(w_i)}{P(x)}$$

La fonction de décision bayésienne consiste à associer à chaque individu la classe à laquelle il appartient avec la probabilité la plus forte. Elle est décrite par : [Lau99]

$$\begin{cases} E \rightarrow \Omega \\ X \rightarrow w_i, p(w_i/x) \geq p(w_j/x) \quad \forall j = 1 \dots C \end{cases}$$

Où : **E** : Espace des caractéristiques

**Ω** : ensemble des classes

**C** : nombre de classes

**X** : ensemble des individus à classifier

**N** : nombre d'individus à classifier

#### 2.1.2. Champs de Markov

Cette approche permet d'envisager la recherche de modèles de distributions locaux selon un principe équivalent localement à celui de la classification bayésienne.

La théorie des champs de Markov permet la minimisation d'une fonction d'énergie définie sur l'image à segmenter en introduisant la notion de voisinage entre pixels, Une image de taille NxN est définie par l'ensemble de ses pixels Pij et notée : [Gat01, Lau99]

$$I = \{p_{ij} = (i,j), 1 \leq i, j \leq N\}$$

Pour chaque pixel on lui associe une variable aléatoire  $A_i$  ayant des valeurs appartenant à un ensemble  $\Omega$  donné. L'image est représentée par un vecteur aléatoire  $A = (A_i, i \in I)$  et  $a=(a_i, i \in I)$  est une réalisation de  $A$ . Un système de voisinage  $V$  est aussi défini sur cette image de la façon suivante : tout voisinage «  $v$  » vérifie deux propriétés qui sont :

- 1) Un pixel  $n$ 'appartient pas à son voisinage
- 2) La relation de voisinage est symétrique : si un pixel «  $s$  » appartient au voisinage  $V_t$  d'un pixel «  $t$  » alors  $t$  appartient au voisinage  $V_s$  de  $s$ .

La définition d'un champ markovien associe un champ aléatoire  $A$  à un système de voisinage  $V$  à la condition suivante : la condition de positivité qui est nécessaire pour que le champ soit un champ de **Gibbs** :

$$\begin{cases} P(\mathbf{a}) > 0 \\ P(a_i / a_j, j \in I \setminus \{i\}) = P(a_i / a_j, v_i) \end{cases}$$

L'influence d'un pixel est donc limitée à un champ local de l'image, située autour de ce pixel. La définition des cliques est nécessaire pour la formalisation. Une clique est un ensemble de pixels tel que deux pixels quelconques de l'ensemble sont voisins au sens de  $V$ . Un champ aléatoire  $A$  associé à un système de voisinage  $V$  est un champ de Markov si et seulement si sa distribution de probabilité  $P(A = a)$  est une mesure de Gibbs : [Lau99]

$$\forall A \in \Omega^{N \times N}, P(a) = \frac{\exp(-U(a))}{\sum_{a \in \Omega^{N \times N}} \exp(-U(a))}$$

Où  $U$  est une fonction d'énergie correspondant à la somme des potentiels de chaque clique de l'image. En utilisant la théorie markovienne la segmentation est considérée comme l'estimation d'une image classifiée à partir de l'image observée  $A$ , c'est-à-dire la modélisation de  $P(\Lambda = \lambda / A = a)$  [Gem84]. L'approche classique consiste à utiliser une approche bayésienne qui permet, à partir d'une modélisation prédéfinie de la loi sur  $\Lambda$  de retrouver la probabilité  $P(\Lambda = \lambda / A = a)$

En général,  $\Lambda$  est une loi paramétrée et les paramètres optimaux sont obtenus par maximisation ultérieure de la probabilité. L'approche markovienne de la segmentation est ainsi consistée à définir une énergie sur une image puis à trouver une classification qui

minimise cette énergie. Ce type d'approche est à mettre en relation avec les approches de type contours actifs qui utilisent aussi des principes de minimisation énergétique pour la recherche de contours [Lau99].

#### 2.2. Les approches basées contour

L'approche basée sur les contours vise à identifier les contours des différents objets présents dans l'image. Dans cette méthode, la détection des contours consiste à explorer une zone d'intérêt. Pour chaque position de contour (à partir des pixels susceptibles de constituer un contour), on évalue s'il existe une transition significative au niveau de l'attribut sélectionné (couleur, niveau de gris, etc.). Ensuite, il est nécessaire de déterminer des contours fermés. Cependant, l'approche basée sur les contours est sensible au bruit et nécessite souvent l'application de techniques de lissage avant d'entreprendre la segmentation. Il est important de souligner que l'approche basée sur les contours ne conduit pas directement à une segmentation, car les contours détectés ne sont pas toujours continus, en raison des discontinuités présentes dans l'image résultant de variations d'éclairage (par exemple, l'effet d'ombre). Cependant, il existe des techniques permettant d'obtenir des contours fermés en exploitant les informations relatives aux régions.

##### 2.2.1 Techniques de détection du contour

La détection des contours dans une image a été l'un des problèmes les plus étudiés depuis les débuts de la recherche en imagerie. Cela s'explique en grande partie par la nature hautement intuitive du contour, qui se présente comme l'indice visuel idéal dans la plupart des situations.

La détection de contours dans le domaine du traitement d'images revêt une importance primordiale, car elle permet une réduction significative de la quantité d'informations liées à une image, tout en préservant des éléments structurels tels que les contours et les frontières. [Jia00]

Ces techniques se concentrent sur la détection des frontières des objets résultant des discontinuités. Elles commencent par détecter des contours initiaux (bords) dans une image, puis tentent de reconstruire des contours fermés à partir de ces bords. Les bords correspondent généralement à des discontinuités locales. En d'autres termes, un bord se produit chaque fois qu'il y a un saut d'une valeur constante locale d'un côté à une valeur constante locale

différente de l'autre côté. Les techniques de détection de contours supposent qu'il existe des discontinuités claires le long de la majeure partie des véritables frontières. Cependant, ces approches rencontrent des difficultés particulières avec les frontières caractérisées par de faibles discontinuités de contraste et des changements d'intensité à angles extérieurs [NSal].

Les détecteurs de contours sont utilisés sur une image afin de différencier les pixels de contour des pixels qui ne sont pas des contours. Le résultat obtenu est une carte de contours, fournissant les informations nécessaires pour tracer les contours exacts des régions d'intérêt dans une image. Des renforçateurs de contours sont souvent utilisés pour accentuer les contours, mais ils ne font pas de distinction entre les pixels de contour et les pixels qui ne le sont pas. Des techniques telles que le gradient, le laplacien et les opérateurs gaussiens sont des exemples de renforçateurs de contours, tandis que le seuillage et l'opérateur de Canny sont des exemples de détecteurs de contours.

Les approches gradient et Laplacien reposent sur le fait que les contours correspondent à des discontinuités d'ordre 0 de la fonction d'intensité. L'approche gradient détermine les extrêmes locaux dans la direction du gradient et l'approche Laplacien détermine les passages par zéro du Laplacien.

#### a) Le Gradient

En utilisant les dérivés partiels, le gradient d'une image donne les taux de changement de niveau de gris par unité de distance dans les directions des axes de coordonnées [NSal].

#### b) Le Laplacien

Ressemble à l'opérateur gradient, l'opérateur Laplacien est un autre renforceur de bordure basé sur les dérivés de l'image. L'opérateur du gradient emploie la première dérivée spatiale de l'image, tandis que l'opérateur Laplacien est basé sur la dérivée seconde de l'image. Cependant, le Laplacien est caractérisé par sa sensibilité au bruit accrue par rapport au gradient [NSal].

### 2.2.2. Méthodes des contours actifs

La segmentation d'images, qui consiste à extraire les contours d'un objet, est l'une des tâches cruciales en traitement d'images. Les contours actifs représentent une approche distincte par rapport aux méthodes traditionnelles de segmentation d'images. Leur principe

consiste à faire évoluer un contour initial fermé vers une position d'équilibre, en se rapprochant des bords de l'objet à détecter [Coh93, Hat03].

### 2.3. Les approches basées région

L'approche basée sur la segmentation par région consiste à fusionner directement les pixels et les régions voisines partageant une certaine propriété commune. L'ensemble final des régions fusionnées de pixels forme l'image segmentée. Les méthodes les plus populaires de segmentation par région sont celles qui utilisent soit la croissance de régions, soit la division/fusion de régions [NSal17].

#### 2.3.1. Technique de seuillage

Le seuillage est l'une des techniques de segmentation les plus anciennes, qui essaient de distinguer l'objet d'intérêt du fond de l'image par un seuil d'intensité. Les techniques de seuillage les plus anciennes essaient de trouver un seuil simple pour l'image entière ; Cependant, un seuil simple est sensible au changement d'intensité dans les différentes parties de l'image. Les techniques ultérieures sont plus adaptatives pour trouver différents seuils pour différentes sections de données. Le seuillage est souvent employé dans l'analyse d'images médicale, parfois comme étape essentielle de prétraitement ou d'initialisation pour d'autres techniques [Sah98].

Le seuillage est une technique simple pour la détection de contour et la segmentation d'images. Cette technique, emploie une valeur seuil ( $T$ ) choisie dans l'intervalle des niveaux de gris de l'image, puis elle marque différemment les pixels ayant un niveau de gris inférieur et supérieur de cette valeur. Le seuillage peut être défini comme suit :

$$I(x,y) > T \Rightarrow IT(x,y) = 1$$

$$I(x,y) \leq T \Rightarrow IT(x,y) = 0$$

Où  $I(x, y)$  est l'image après le seuillage, par conséquent, tous les pixels avec des intensités supérieures au *seuil* sont étiquetés, par exemple par 1, et le reste sera zéro, le résultat du seuillage est donc une image binaire.

Le seuillage peut être à niveau unique, à deux niveaux ou à plusieurs niveaux. Dans le seuillage à un seul niveau, une seule valeur du seuil est choisie, comme décrit précédemment. Dans la méthode à deux niveaux, deux valeurs du seuil sont choisies, basses ( $T_{bas}$ ) et haut ( $T_{haut}$ ).

Les Pixels ayant des intensités situées dans cet intervalle sont considérées comme points du contour et sont étiquetés par une certaine valeur, par exemple 1, et le reste sera 0...etc. La technique de seuillage est cependant inadaptée au cas des images contenant plusieurs objets de même couleur. La segmentation dans ce cas peut fournir une autre image.

#### 2.3.2. Techniques de fusion et de partage

Les techniques de partage et de fusion de régions divisent l'image en petites régions et fusionnent itérativement les régions voisines qui satisfont un certain critère d'uniformité. La fusion la plus simple vérifie essentiellement les contraintes de ressemblances d'intensités moyennes. Des méthodes plus complexes sont les méthodes basées modèle, où les régions sont seulement fusionnées si elles sont du même modèle, et si les paramètres du modèle sont convenables. Les résultats du processus de fusion donnent de plus en plus grandes régions qui sont homogènes en tenant en considération un certain critère d'uniformité [HMai].

#### 2.3.3. Segmentation d'images par croissance de région

La segmentation par croissance de région vise à fusionner les pixels les plus proches de l'image qui sont uniformes en terme de leurs attributs (la couleur, l'intensité...) en prenant en considération une certaine mesure d'uniformité [Jia03]. Cette approche par croissance de région doit choisir un ensemble de pixels initiaux considérés comme des germes à partir desquels la croissance des régions commencera. Le critère d'uniformité utilisé peut par exemple être la variance d'une région ou la somme des erreurs quadratiques... qui doivent ne pas dépasser un seuil prédéterminé. La croissance des régions s'effectue par agrégation de pixels aux germes initiaux, elle consiste à fusionner successivement aux régions les pixels voisins qui sont similaires, en ce sens que les régions continuent à vérifier le critère d'uniformité après chaque agrégation d'un pixel. Les régions adjacentes qui ensemble vérifient le critère d'uniformité peuvent être fusionnées et de nouveaux germes peuvent être insérés dans les régions ne pouvant pas être fusionnées avec les régions existantes [Bou98].

### 3. Méthodes de segmentation d'images 3D

La segmentation est une étape essentielle visant à extraire un ensemble limité d'informations géométriques pertinentes (sommets, courbes, surfaces, etc.) afin de réaliser une tâche spécifique dans le module de perception. Cependant, l'image de profondeur est souvent

sujette à des artefacts dus aux méthodes d'acquisition, ce qui rend difficile la recherche d'une méthode véritablement efficace pour la segmentation.

Les procédures de segmentations trouvées dans la littérature sont très nombreuses, et ils cherchent tous à trouver deux caractéristiques qui sont :

- Les discontinuités entre surfaces ;
- Les surfaces.

On peut englober les algorithmes de segmentation en deux principales catégories qui seront présentées dans ce paragraphe.

### 3.1. Les approches fondées sur l'extraction de contours

La première idée est d'utiliser les méthodes propres aux traitements d'images de luminance : les extracteurs de contours.

Le problème est de construire des masques propres aux images de profondeur car les images de luminance auront un masque plus simple parce qu'on cherche la discontinuité de luminance ou de couleur mais dans le cas de l'image de profondeur on cherche deux discontinuités :

- la discontinuité de profondeur ;
- la discontinuité dans la dégradation entre les profondeurs.

Monga et Deriche [Mon88], [Mon89] ont généralisé leur détecteur de contours 2D, à la dimension  $n$ . Pour cela ils utilisent des filtres à réponse impulsionnelle séparables selon les différentes directions, qui leur permettent de ramener le problème à la recherche de filtres de lissage monodimensionnels et des filtres de dérivation correspondants. Les filtres utilisés sont les filtres de Deriche et Shen.

A partir des contours après une extraction, certains auteurs ont proposé des techniques pour extraire des informations sur la nature des surfaces adjacentes aux contours. Par exemple, dans [Bol79], un ensemble de points 3D admettant pour envelopper deux segments, comme par exemple une ellipse et une demi-ellipse seront interprétés comme étant la surface d'un cylindre.

### 3.2. Les approches fondées sur l'extraction de régions

Du fait de la difficulté d'extraire des contours d'une image de profondeur acquise par une camera laser, la plupart des méthodes existants se reposent sur le principe de « Split and Merge » (séparation-fusion) ; rappelons que le problème général en segmentation par régions est de trouver une partition de l'ensemble des données perceptuelles en régions connexes.

Le critère d'uniformité  $H$  doit être vérifié par toutes les données internes à une région :

Ce critère doit aussi permettre de discriminer une région de toutes les régions qui lui sont adjacentes. Ces deux relations peuvent s'exprimer comme suit :

- Pour tout région  $R(i)$  : - la région doit être uniforme  $\Leftrightarrow H(R(i))$  est vrai ;
- Pour toute région  $R(j)$  adjacente à  $R(i)$ ,  $H(R(i) + R(j))$  est faux.

Tout algorithme de « Split and Merge » comporte 3 phases :

- **La phase d'initialisation**

On définit les régions de départ « region seed » qui peuvent ne pas satisfaire les deux dernières règles, ces régions peuvent être choisies arbitrairement. Dans le cas d'une image (organisation matricielle des données perceptuelles), la structuration en «quadtree» (arbre quaternaire ou chaque feuille est une partie de l'image) est souvent utilisée pour faciliter la gestion des fusions et séparation de régions, mais cette structure impose des contraintes sévères sur la forme des régions.

- **La phase de fusion (Merge)**

Le critère  $H$  (critère d'uniformité) sert à réunir les régions adjacentes qui ne satisfont pas la seconde règle.

- **La phase de séparation (Split)**

Le critère  $H$  sert cette fois à couper les régions qui ne satisfont pas la première règle. En cas générale la séparation (division) d'une région sert à maximiser le critère  $H$  de chaque région résultante.

Deux cas spéciaux de cette méthode peuvent être soulignés :

- À l'initialisation on utilise tout l'ensemble comme région source et puis on commence par le « split » jusqu'à trouver un résultat optimal.
- Au contraire cette fois on utilise chaque point d'un ensemble complet comme région à elle-même et dans ce cas on commence par le « merge » jusqu'à trouver des régions infusibles, dans ce cas on parle d'un algorithme de « croissance de régions ».

Il faut bien noter que cette méthode (« Split and Merge ») dépend fortement du choix du critère d'uniformité  $H$

On trouve dans la littérature des travaux dans ce type de méthodes tel que Colly en [Col91] et Mazzouzi en [Maz08] et [Maz07].

Colly [Col91] a proposé sur la méthode de grossissement de régions qui comporte deux phases :

- Le découpage de l'image de point en petites régions : (les patches) ;
- Le grossissement de ces régions élémentaires par adjonction de points

Les patches sont des régions élémentaires, et le but de ce système est de regrouper les points qui ont les mêmes caractéristiques.

### 3.3. Les approches Pyramidales

On trouve dans la littérature cette approche spécialisée au début pour la segmentation des images IRM (images 3D) mais Checchin et Al en [Che97] ont proposé une segmentation des images de profondeur avec une approche pyramidale pour les objets uniquement polyédriques cet algorithme repose sur deux principes :

- Estimation de la normale en chaque une des pixels de l'image.
- Le choix des germes de départ pour la croissance des régions.

Le principe général de cette méthode est le suivant :

1) La distinction de chaque pixel par un vecteur normal avec comme suit :

Le vecteur normal  $N$  du point  $P$  de coordonnées  $x, y$  tel que  $z=f(x, y)$  est :

$$\begin{aligned}
 N &= \frac{\partial OP}{\partial x} \wedge \frac{\partial OP}{\partial y} \\
 &= \begin{vmatrix} 1 & 0 \\ 0 & z_x = f'_x(x, y) \end{vmatrix} \wedge \begin{vmatrix} 0 & 1 \\ z_y = f'_y(x, y) & 0 \end{vmatrix}
 \end{aligned}$$

Puis on passe par l'analyse unidimensionnelle définis par les axes X, Y séparément, et on remarque bien que les points dans le même segment vertical ou horizontal ont le même vecteur alors on peut distinguer un segment selon les points horizontal (d'équation  $Z = Ax$ ) ou vertical ( $Z = By$ ).

2) Sélectionné les germes de départ qui ne seraient que les pixels qui appartiennent aux segments horizontaux ou verticaux et puis affectés tout les pixels qui n'appartiennent pas aux segments aux régions qui satisfont le plus le critère d'homogénéité, et tant que l'image obtenue est largement sur-segmentée alors il faut passer à l'étape de fusion. Les régions extraites sont représentées par un graphe d'adjacence et initialisent une pyramide de graphes.

3) La pyramide des graphes est une pyramide irrégulière caractérisée par :

- C'est un empilement de graphes d'adjacence, chacun d'eux correspondant à un niveau de la pyramide ;
- Le niveau  $k \in \mathbb{N}$  de la pyramide est construit à partir du niveau  $k-1$  ;
- Chaque région de l'image, accompagnée de ses attributs, est modélisée par un sommet du graphe d'adjacence.
- La réduction du nombre de régions du niveau  $k$  au niveau  $k+1$  s'exprime dans cette approche orientée « graphes » par la réduction du nombre de sommets. Sur le graphe de similarité, les deux règles suivantes doivent être respectées :

**Règle1 :** Deux sommets adjacents au niveau  $k$  ne peuvent survivre tous les deux au niveau  $k+1$ .

**Règle2 :** Au niveau  $k$ , tout sommet non-survivant doit posséder au moins un sommet de son voisinage qui survit au niveau  $k+1$ .

### 4. Conclusion

Dans ce chapitre, nous avons revisité la définition de la segmentation d'une image et nous avons brièvement exploré différentes familles de méthodes utilisées pour la segmentation d'images 2D et 3D. Ensuite, nous avons fourni un aperçu de l'état de l'art des méthodes les plus couramment utilisées pour la segmentation d'images.

Au chapitre suivant, nous entamerons la présentation de notre travail en introduisant les nouvelles opérations que nous avons définies. Ces opérations permettent une segmentation efficace d'image de profondeur.

# Chapitre 3

## De nouvelles approches pour la segmentation d'image de profondeur

### 1. Introduction

La segmentation d'images joue un rôle fondamental et crucial dans de nombreuses applications de vision par ordinateur. Malgré le développement de nombreuses méthodes de segmentation, il n'existe pas encore de résultats entièrement satisfaisants. L'analyse des images de profondeur constitue un problème largement étudié en traitement d'images, où les méthodes de segmentation sont couramment utilisées. Ces méthodes visent à regrouper les pixels qui présentent des propriétés similaires, dans le but de segmenter efficacement l'image.

En réalité, lorsque l'image est fortement affectée par le bruit, l'application d'un lissage intensif peut entraîner la suppression des contours des objets [Pul97] et/ou la création de régions qui n'existent pas réellement dans l'image. Cependant, si l'image est insuffisamment lissée, le bruit résiduel compromet l'efficacité de la détection des contours et entraîne des erreurs.

Nous proposons dans ce chapitre 2 algorithmes de segmentation d'images de profondeur. Nous montrons que le dernier, notamment dans le chapitre suivant, représente la solution la plus adéquate pour une segmentation d'image de profondeur. Ces algorithmes sont :

- Un algorithme de segmentation adaptative de l'image de profondeur avec les données de profondeur brutes.
- Un algorithme de segmentation adaptative de l'image de relief travaillant sur une image de relief exprimée par les produits des dérivées image.

## 2. Représentation des images de profondeur

Une image de profondeur est une représentation où chaque pixel enregistre la distance entre le point correspondant de la scène et le plan du capteur télémétrique. Les images de profondeur sont généralement représentées sous forme de matrices, avec des valeurs comprises dans un intervalle déterminé par les capacités du capteur. Ces valeurs sont généralement de type réel.

Chaque image représente une scène qui contient des objets polyédriques qui se chevauchent mutuellement. Chaque pixel  $(x, y)$  de l'image correspond à la distance  $Z(x, y)$  entre le plan du capteur télémétrique et le point correspondant dans la scène. Dans cette représentation, les régions de l'image correspondent aux parties visibles des surfaces des différents objets présents dans la scène [Maz09].

On trouve dans l'état de l'art beaucoup de méthodes de représentation chacune est dépendante de l'application du système conçu.

### 2.1. Représentation en image de relief

L'image de relief est dérivée de l'image de profondeur, et elle est représentée par une matrice de relief où chaque point  $(i, j)$  contient les composantes du vecteur normal à la surface en ce point.

Dans cette méthode, chaque pixel est associé à deux composantes : la profondeur du pixel et le vecteur normal à la surface en ce pixel.

On peut obtenir l'image de relief par deux méthodes :

#### 2.1.1. La représentation par produit des dérivés

On a calculé le vecteur normal  $(\vec{N})$  par le produit vectoriel entre le vecteur de dérivé  $\vec{V1}$  par rapport à  $x$  et le vecteur de dérivé  $\vec{V2}$  par rapport à  $y$  comme suit :

$$\begin{aligned}\vec{N} &= \vec{V1} \wedge \vec{V2} \\ &= \begin{pmatrix} 1 \\ 0 \\ I'_x(I(i,j)) \end{pmatrix} \wedge \begin{pmatrix} 0 \\ 1 \\ I'_y(I(i,j)) \end{pmatrix} \\ &= \begin{pmatrix} -I'_x(I(i,j)) \\ I'_y(I(i,j)) \\ 1 \end{pmatrix}\end{aligned}$$

Où :

$I'_x(I(i,j))$  est la dérivée première de l'image I selon l'axe x tel que :

$$I'_x(i,j) = \frac{\partial I(i,j)}{\partial x}$$

$I'_y(I(i,j))$  est la dérivée première de l'image I selon l'axe y tel que :

$$I'_y(i,j) = \frac{\partial I(i,j)}{\partial y}$$

Le vecteur de dérivé  $\vec{V1}$  est dirigé vers la gauche par contre le vecteur de dérivé  $\vec{V2}$  est dirigé vers le haut et ils sont parallèle à la surface qui contient le point.

Le calcul du produit vectoriel entre  $\vec{V1}$  et  $\vec{V2}$  nous donne un vecteur normal de ces deux vecteurs c.-à-d. un vecteur normal à la surface au point (i, j).

#### 2.1.2. La représentation par la méthode de régression multiple

La méthode de régression multiple est un outil statistique permettant d'estimer le meilleur plan à la surface à un point donné. Le voisinage  $X(x, y)$  d'un pixel (x, y) est constitué des pixels (x', y') situés dans une fenêtre de taille 3×3 centrée en (x, y), et dont les profondeurs  $Z(x', y')$  sont à un seuil donné (Ebs) de profondeur proches.

Dans la méthode de régression multiple on calcule l'équation du plan dans un référentiel cartésien 3D qui passe par le point (x, y) en utilisant les pixels appartenant à son voisinage et qui peut être exprimée comme suit :

$$ax + by + c = z$$

## Chapitre 3 :

### De nouvelles approche pour la segmentation d'image de profondeur

Avec cette équation du plan déduit les composantes du vecteur normal  $\vec{v}$  à la surface qui contient le point (i, j) qui sont (a, b, -1)<sup>T</sup> comme suit :

En premier, les coefficients de la surface au point (x, y) sont obtenus par la minimisation de la fonction, donnée par :

$$\Phi(a, b, c) = \sum_{i=1}^n (ax_i + by_i + c - z_i)^2$$

En second terme on cherche la dérivé de  $\Phi$  par apport a chaque composante ; a, b et c comme suit :

$$\frac{\partial \Phi}{\partial a} = a \sum_{i=1}^n X_i^2 + b \sum_{i=1}^n X_i Y_i + c \sum_{i=1}^n X_i = \sum_{i=1}^n X_i Z_i$$

$$\frac{\partial \Phi}{\partial b} = a \sum_{i=1}^n X_i Y_i + b \sum_{i=1}^n Y_i^2 + c \sum_{i=1}^n Y_i = \sum_{i=1}^n Y_i Z_i$$

$$\frac{\partial \Phi}{\partial c} = a \sum_{i=1}^n X_i + b \sum_{i=1}^n Y_i + nc = \sum_{i=1}^n Z_i$$

Puis on calcule les deltas suivants :

$$\text{Delta} = \begin{bmatrix} \sum_{i=1}^n X_i^2 & \sum_{i=1}^n X_i Y_i & \sum_{i=1}^n X_i \\ \sum_{i=1}^n X_i Y_i & \sum_{i=1}^n Y_i^2 & \sum_{i=1}^n Y_i \\ \sum_{i=1}^n X_i & \sum_{i=1}^n Y_i & n \end{bmatrix}$$

$$\Delta a = \begin{bmatrix} \sum_{i=1}^n X_i Z_i & \sum_{i=1}^n X_i Y_i & \sum_{i=1}^n X_i \\ \sum_{i=1}^n Y_i Z_i & \sum_{i=1}^n Y_i^2 & \sum_{i=1}^n Y_i \\ \sum_{i=1}^n Z_i & \sum_{i=1}^n Y_i & n \end{bmatrix}$$

$$\Delta b = \begin{bmatrix} \sum_{i=1}^n X_i^2 & \sum_{i=1}^n X_i Z_i & \sum_{i=1}^n X_i \\ \sum_{i=1}^n X_i Y_i & \sum_{i=1}^n Y_i Z_i & \sum_{i=1}^n Y_i \\ \sum_{i=1}^n X_i & \sum_{i=1}^n Z_i & n \end{bmatrix}$$

$$\Delta c = \begin{bmatrix} \sum_{i=1}^n X_i^2 & \sum_{i=1}^n X_i Y_i & \sum_{i=1}^n X_i Z_i \\ \sum_{i=1}^n X_i Y_i & \sum_{i=1}^n Y_i^2 & \sum_{i=1}^n Y_i Z_i \\ \sum_{i=1}^n X_i & \sum_{i=1}^n Y_i & \sum_{i=1}^n Z_i \end{bmatrix}$$

Avec les deltas on peut avoir les composantes a, b et c comme suit :

$$\mathbf{a} = \frac{\mathbf{\Delta a}}{\mathbf{\Delta}}$$

$$\mathbf{b} = \frac{\mathbf{\Delta b}}{\mathbf{\Delta}}$$

$$\mathbf{c} = \frac{\mathbf{\Delta c}}{\mathbf{\Delta}}$$

## Chapitre 3 :

### De nouvelles approche pour la segmentation d'image de profondeur

Afin de normaliser les vecteurs normaux à la surface, nous procédons comme suit :

- a. Calcul de la norme du vecteur :

$$\text{Norme} = \sqrt{a^2 + b^2 + 1}$$

- b. Normalisation :

$$a = \frac{a}{\text{Norme}}$$

$$b = \frac{b}{\text{Norme}}$$

$$c = \frac{-1}{\text{Norme}}$$

Pour l'image entière, nous parcourons la matrice correspondante, en convertissons les positions horizontales et verticales (i, j) en coordonnées réelles (x, y) comme suit :

$$X = \frac{(i - \frac{size}{2})}{ampl}$$

$$Y = \frac{(\frac{size}{2} - j)}{ampl}$$

$$Z = Im(i, j)$$

Où : **Ampl** : L'échelle ou l'amplitude des valeurs de l'image (donnée par le capteur).

Pour chaque pixel, nous avons calculé le vecteur normal en utilisant un voisinage de points. Ce voisinage est défini par une fenêtre de taille 3x3 qui entoure le pixel en question. L'algorithme suivant fournit des détails supplémentaires sur le calcul du vecteur normal pour chaque pixel.

#### Algorithme de Régression Multiple

##### Début

**pour** i'  $\leftarrow$  1 à size-1 **faire**

**pour** j'  $\leftarrow$  1 à size-1 **faire**

Inisialisation ;

n  $\leftarrow$  9;

**pour** i  $\leftarrow$  1 à size-1 **faire**

**pour** j  $\leftarrow$  1 à size-1 **faire**

x  $\leftarrow$  (i-size/2)/Echelle;

y  $\leftarrow$  (size/2-j)/Echelle;

z  $\leftarrow$  matP(i,j);

sx  $\leftarrow$  sx + x;

sx2  $\leftarrow$  sx2 + x \* x;

sy  $\leftarrow$  sy + y;

sy2  $\leftarrow$  sy2 + y \* y;

sz  $\leftarrow$  sz + z;

sxy  $\leftarrow$  sxy + x \* y;

sxz  $\leftarrow$  sxz + x \* z;

syz  $\leftarrow$  syz + y \* z;

##### **Finpour**

##### **Finpour**

delta  $\leftarrow$  sx2 \* (n \* sy2 - sy \* sy) - sxy \* (sxy \* n - sx \* sy) + sx \* (sxy \* sy - sx \* sy2);

deltaA  $\leftarrow$  sxz (sy2 \* n - sy \* sy) - sxy \* (syz \* n - sz \* sy) + sx \* (syz \* sy - sz \* sy2);

deltaB  $\leftarrow$  sx2 \* (syz \* n - sz \* sy) - sxz \* (sxy \* n - sx \* sy) + sx \* (sxy \* sz - sx \* syz);

deltaC  $\leftarrow$  sx2\*(sy2 \* sz - sy \* syz) - sxy\*(sxy \* sz - sx \* syz)+sxz \*(sxy\*sy - sx\*sy2);

a  $\leftarrow$  deltaA / delta;

b  $\leftarrow$  deltaB / delta;

c  $\leftarrow$  -1;

Norme  $\leftarrow$   $\sqrt{a^2 + b^2 + 1}$ ;

a  $\leftarrow$  a / Norme ;

b  $\leftarrow$  b / Norme ;

c  $\leftarrow$  -1 / Norme ;

matR(i',j').a  $\leftarrow$  a;

matR(i',j').b  $\leftarrow$  b;

matR(i',j').c  $\leftarrow$  c;

##### **Finpour**

##### **Finpour**

##### **Fin**

Où : size \* size est la taille de l'image,

matP est la matrice de profondeur, et

matR est la matrice de relief où chaque éléments (i, j) contient les 3 composants (a, b,

c) du vecteur normal à la surface en ce point.

### 3. Prétraitement

#### 3.1. Le lissage adaptatif

Nos méthodes utilisent souvent des techniques de prétraitement, telles que le lissage, qui est une opération mathématique visant à réduire le bruit avant d'effectuer la segmentation. Cela permet d'éliminer le bruit présent dans ces images. Cependant, il faut faire attention car un lissage excessif ou plusieurs itérations de lissage peuvent entraîner l'effacement des contours des objets et la déformation des objets dans l'image. Bien que le lissage successif puisse éliminer les régions bruitées, il est important de trouver un équilibre pour préserver les contours et la forme des objets dans l'image.

En comparaison avec les méthodes de segmentation classiques, les images de profondeur sont souvent sujettes à un fort bruit. Cela pose un défi lors de l'application d'un lissage simple, en particulier pour les points situés entre deux objets.

Le lissage simple crée des points qui n'appartiennent à aucune région. Alors nous avons eu besoin d'appliqué sur ces images un lissage adaptatif où il prend en considération les points qui sont proche du point centre de lissage c.-à-d. les points du même objet.

Les images utilisées représentent des scènes contenant des objets polyédriques. La technique du lissage se fait lors du parcours de l'image.

### 4. Les méthodes de segmentations

Les méthodes conventionnelles de segmentation d'images ne produisent pas de résultats précis pour les images de profondeur en raison du niveau élevé de bruit qui les affecte. Il a également été observé que l'application d'un lissage intensif du bruit entraîne l'effacement des contours. Cependant, lorsque le bruit est insuffisamment filtré, la détection des contours aboutit à de nombreuses erreurs. Pour remédier à ce problème, notre travail a exploré trois méthodes différentes : deux approches de segmentation adaptative et une autre basée sur l'accroissement de région aléatoire.

## Chapitre 3 :

### De nouvelles approche pour la segmentation d'image de profondeur

#### 4.1. Segmentation adaptative de l'image de profondeur

Vu la forte dégradation des données de profondeur, nous avons opté pour l'opérateur de la courbure pour déterminer si un point de l'image appartient à un bord d'une surface (point A) ou appartient à l'intérieur d'une surface (point B).

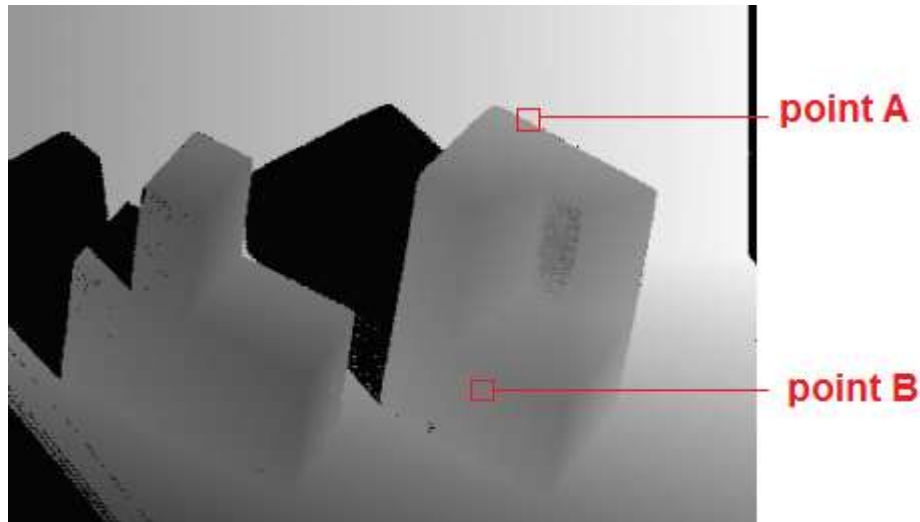


Figure 3.1 : points dans une image de profondeur

Expression de la courbure :

$$\gamma = \frac{(1 + f_y'^2) f_{xx}'' + (1 + f_x'^2) f_{yy}'' - 2 f_x' f_y' f_{xy}''}{2(1 + f_x'^2 + f_y'^2)^{3/2}}$$

f désigne la fonction image :  $f(x,y) = I[x,y]$ .

#### Algorithme de segmentation adaptative de profondeur

##### Début

Lire ebs1 ;

**Pour** i=2 à size-2 faire

**Pour** j=2 à size-2 faire

C=Courbure(x,y)

**Si** (C< SeuilC )

**Alors** point de surface ;

**Sinon** point de contour ;

**Fin si**

**Fin pour**

**Fin pour**

**Fin.**

Malgré la rapidité et la simplicité de cette méthode nous avons trouvé beaucoup d'inconvénients qui peuvent être résumés comme suit :

- L'opération nécessite un lissage qui peut influencer sur les résultats de segmentation.
- L'opération de segmentation dépend complètement du choix de seuil « SeuilC » ce qui rend l'opération semi-automatique.
- Les surfaces fortement bruitées seront détectées comme des contours.
- On ne peut pas détecter les facettes fortement inclinées avec cet algorithme car elles seront considérées comme des surfaces bruitées. La dégradation dans les valeurs est grande et sensible au bruit.

L'organigramme suivant montre le fonctionnement de la méthode de segmentation, basée sur la courbure de l'image de profondeur.

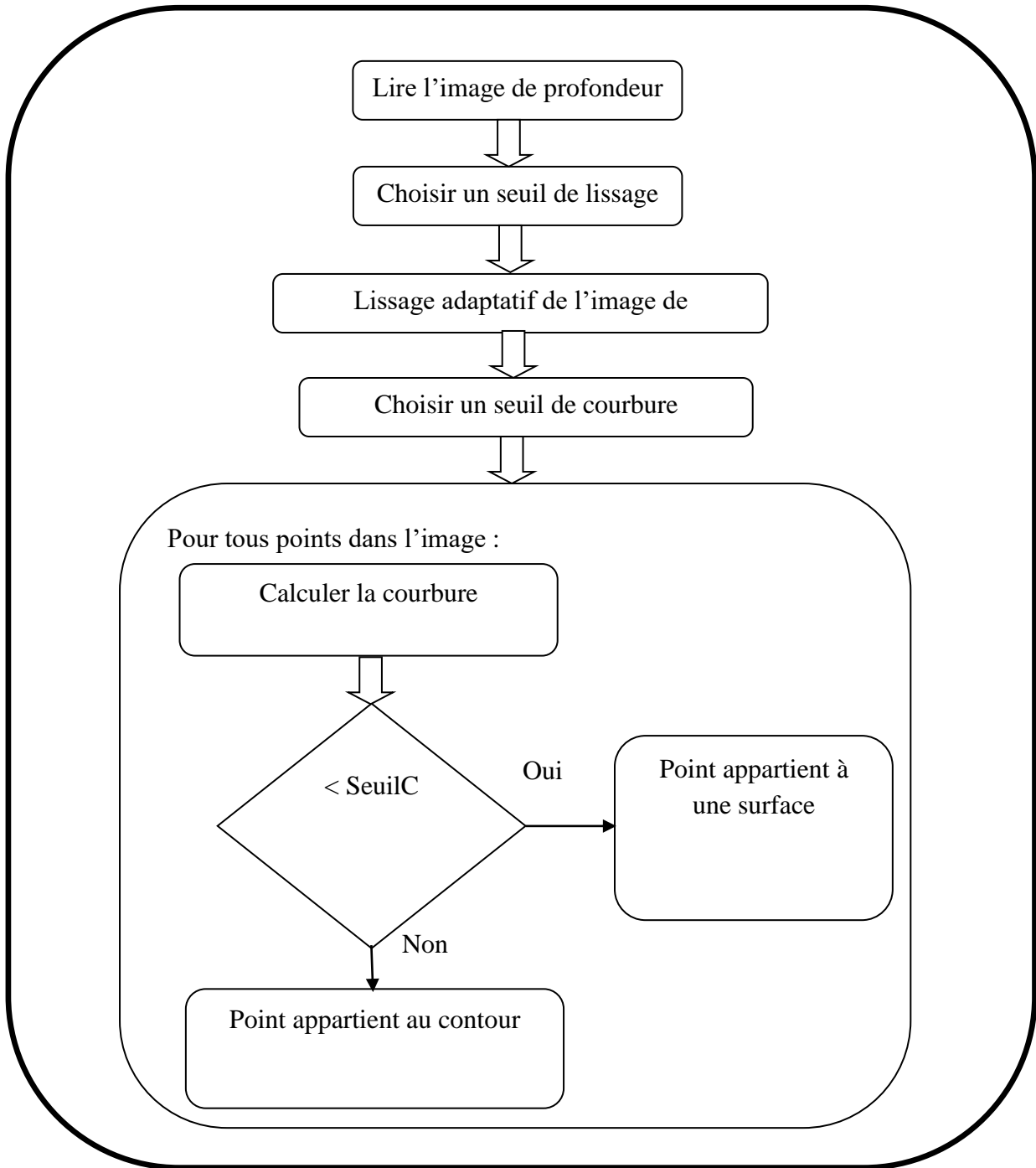


Figure 3.2 : Organigramme de la segmentation basée courbure adaptative de l'image de profondeur.

#### 4.2. Segmentation adaptative de l'image de relief

A cause de la grande quantité de bruit dans les images de profondeur nous étions obligés de les transformer en d'autres formats. Nous avons opté pour l'image de relief, étant donné que pour ce format d'image, c'est la surface qui est représenté et non pas la profondeur.

L'image de relief est une image calculée à partir de l'image de profondeur mais elle contient plus d'informations car en général l'image de profondeur contient deux composantes : La première est la profondeur du point par rapport au capteur, et la deuxième est une composante complexe qui contient les trois composantes du vecteur normale à la surface en ce point.

Après représentation de l'image de relief par le produit des dérivés nous avons effectué un lissage adaptatif sur la même image de relief puis la segmentation est effectuée comme suit :

On calcule le gradient des angles des vecteurs normaux aux surfaces avec m'axe Z.

Si la norme du gradient est supérieure à un certain seuil (SeuiNG) alors le point est considéré comme point de contour, sinon il s'agit d'un point de surface.

L'algorithme suivant montre le principe de la méthode :

#### Algorithme de segmentation adaptative de relief par Gradient des angles

##### Début

Lire SeuilNG;

Pour i=1 à size-1 faire

Pour j=1 à size-1 faire

Gx = Angle(x,y)- Angle(x-1,y)

Gy= Angle(x,y)- Angle(x,y-1)

NG= Sqrt(Gx<sup>2</sup> + Gy<sup>2</sup>)

Si (NG > SeuilNG )

Alors Point de contour;

Sinon Point de surface ;

Fin si

Fin pour

Fin pour

Fin.

Nous avons aussi représenté nos résultats obtenus en utilisant une coloration inspirée de l'algorithme de visualisation Ray-Tracing en utilisant les vecteurs normaux à la surface et en simulant une source lumineuse. Le déroulement du calcul se fait comme suit :

- On calcule le vecteur porteur du segment entre la source lumineuse.
- Le cosinus de l'angle  $\theta$  entre les deux vecteurs (vecteur normal et vecteur de la source lumineuse) est calculé comme suit :

$$\begin{aligned}\vec{V1} * \vec{V2} &= \text{Cos } \theta * \|\vec{V1}\| * \|\vec{V2}\| \\ &= \|\vec{V1} * \vec{V2}\|\end{aligned}$$

Alors :

$$\text{Cos } \theta = \frac{\|\vec{V1} * \vec{V2}\|}{\|\vec{V1}\| * \|\vec{V2}\|}$$

## Chapitre 3 :

### De nouvelles approche pour la segmentation d'image de profondeur

Où :

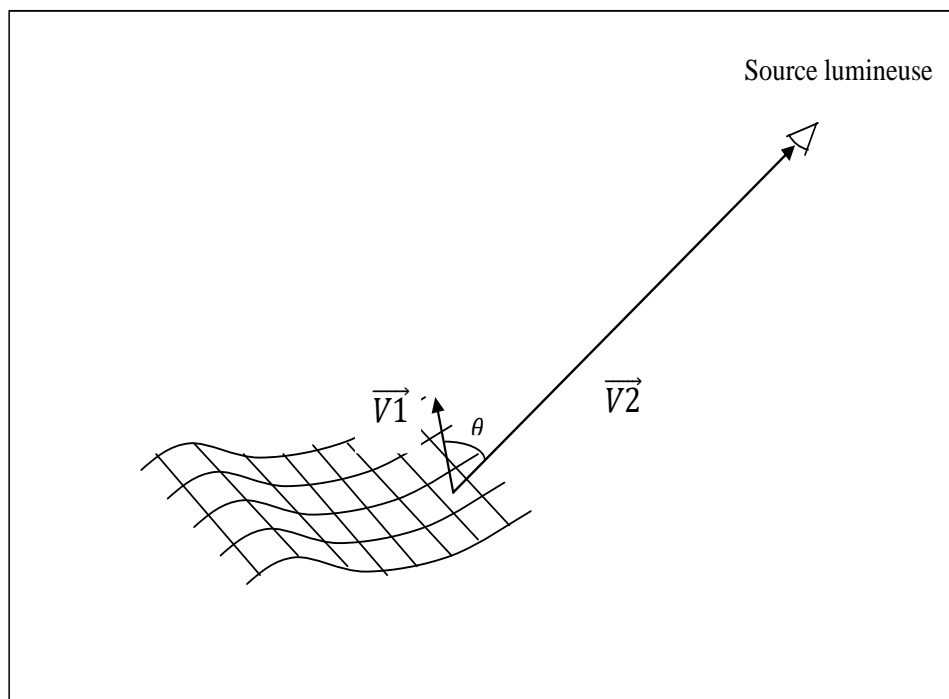
- $\vec{V1}$  est le vecteur normal au point.
- $\vec{V2}$  est le vecteur (point  $\rightarrow$  source lumineuse).
- $\cos \theta$  est l'angle entre ces deux vecteurs où :  $-1 < \cos \theta < +1$

**Si**  $\cos \theta \leq 0$  **alors**

Couleur du pixel = 0

**Si non** Couleur du pixel =  $\cos \theta * 255$

La figure suivant représente l'angle  $\theta$  qui se trouve entre le vecteur normal et le vecteur de la source lumineuse :



**Figure 3.3: Représentation de l'angle  $\theta$  compris entre le vecteur normal et le vecteur de la source lumineuse**

## Chapitre 3 :

### De nouvelles approche pour la segmentation d'image de profondeur

L'algorithme de coloration suivant montre le déroulement de cette méthode de coloration :

**Algorithme de Coloration :**

**Début**  
Inialisation ;  
**pour** i = 0 à size **faire**  
    **pour** j = 0 à size **faire**  
        v(0)  $\leftarrow$  ux - x(i, j);  
        v(1)  $\leftarrow$  uy - y(i, j);  
        v(2)  $\leftarrow$  uz - z(i, j);  
        d1  $\leftarrow$   $\sqrt{v(0)^2 + v(1)^2 + v(2)^2}$ ;  
        d2  $\leftarrow$   $\sqrt{\text{matR}(i, j, 0)^2 + \text{matR}(i, j, 1)^2 + \text{matR}(i, j, 2)^2}$  ;  
        costeta  $\leftarrow$   $\frac{\text{matR}(i, j, 0) * v(0) + \text{matR}(i, j, 1) * v(1) + \text{matR}(i, j, 2) * v(2)}{d1 * d2}$  ;  
        **Si** costeta < 0 **alor**  
            ImR(i, j)  $\leftarrow$  0;  
        **Sinon**  
            ImR(i, j)  $\leftarrow$  costeta \* 250;  
        **FinSi**  
    **Finpour**  
**Finpour**  
Afficher ImR ;  
**Fin**

Tel que : ImR est l'image de relief.

## 6. Conclusion

Dans ce chapitre, nous avons examiné en détail l'adaptation de deux méthodes de segmentation d'images aux images de profondeur, après avoir rappelé la définition de l'image de profondeur. En utilisant des prétraitements tels que le lissage adaptatif, ces méthodes permettent d'éliminer efficacement le bruit, de détecter les contours selon deux représentations différentes.

Les contours ont été correctement détectés, notamment pour la deuxième méthode qui a donné de bons résultats. Nous allons comparer nos deux approches de segmentation d'images de profondeur en utilisant les images réelles de la base ABW. Les résultats de cette comparaison seront présentés dans le prochain chapitre de ce mémoire.

# Chapitre 4

## Mise en œuvre et expérimentation

### 1. Introduction

Dans ce chapitre, nous présentons le contexte dans lequel nous avons mis en œuvre notre segmentation d'images. Pour cela, nous avons utilisé l'outil d'applications Windows et offre un environnement de développement intégré (EDI) convivial appelé Delphi IDE. Nous avons effectué des tests sur plusieurs images réelles provenant d'une base de données appelée ABW, qui comprend 30 images de test [Hoo96].

### 2. Outils de développement

#### 2.1. La plate-forme du développement [Delphi 7]

Delphi est un système de développement visuel rapide sous Windows (Rapid Application Development) qui permet de créer des applications fenêtrées directement exécutables (.EXE) et redistribuables librement sous Windows ou DOS. Il est à noter qu'il existe une version pour Linux nommée KYLIX (compatible avec les composants CLX, il suffit de recompiler). Sa simplicité d'emploi autorise une utilisation immédiate, car il suffit de cliquer-glisser des composants dans une fiche et de gérer quelques événements pour créer des applications simples. Des assistants, modèles et tuteurs interactifs facilitent la prise en main du logiciel.

Delphi utilise le langage Pascal Orienté Objet (il est toutefois possible d'utiliser d'anciennes sources en Pascal standard grâce au compilateur en ligne de commande). Ce langage est facile à apprendre et beaucoup plus simple que le C++ traditionnel. Les objets utilisés ont des propriétés et des méthodes. Les propriétés sont les caractéristiques de l'objet (couleur, taille, ...) tandis que les méthodes sont les procédures (classiques ou événementielles) et fonctions qui y sont rattachées.

Des outils puissants sont rattachés à Delphi. Le canevas facilite les graphismes et évite l'appel aux API de Windows (cette solution reste évidemment possible). Les principales boîtes de dialogue de Windows existent en tant qu'objets et sont facilement réutilisables. L'utilitaire

BDE permet la création et l'accès aisé aux bases de données de tous types. La liaison avec ACCESS se fait directement, grâce à un driver natif intégré.

Il est facile d'écrire des DLL utilisables par n'importe quel autre programme sous Windows. Ces DLL peuvent contenir des fiches Delphi, des procédures, des données ou des Ressources graphiques (bitmaps, icônes, curseurs).

La création d'objets (visuels ou non) réutilisables est un gain de temps appréciable pour le développement. L'intégration des objets Active X peut compléter la bibliothèque existante.

Il existe d'autres systèmes de développement rapide sous Windows. Delphi est particulièrement bien placé :

- moins de lignes de code et rapidité de compilation
- possibilité d'utiliser des procédures événementielles partagées
- notion de modèles réutilisables (fiches, menus, objets)
- orientation objet totale et native
- richesse des composants fournis : pas d'OCX à acheter en complément
- assembleur intégré, compilateur en ligne de commande
- débogage facile au niveau de la source et du processeur
- possibilité d'allocation dynamique de la mémoire en utilisant les pointeurs

Delphi 7 est une version spécifique de l'environnement de développement intégré (EDI) Delphi, développée par Borland et sortie en 2002. C'est l'une des versions les plus populaires de Delphi et elle a été largement utilisée par de nombreux développeurs.

Voici quelques caractéristiques et fonctionnalités clés de Delphi 7 :

- 1. Langage Pascal** : Delphi 7 utilise le langage de programmation Pascal comme base syntaxique. Il est similaire à d'autres versions de Delphi et facile à apprendre si vous avez de l'expérience avec Pascal ou d'autres langages similaires.
- 2. Programmation orientée objet** : Delphi 7 prend en charge la programmation orientée objet (POO) avec des concepts tels que les classes, les objets, l'encapsulation, l'héritage et le polymorphisme. Vous pouvez structurer votre code de manière efficace en utilisant ces principes.
- 3. Bibliothèque de composants visuels (VCL)** : Delphi 7 est livré avec une riche bibliothèque de composants visuels (VCL) qui permet de créer rapidement des

interfaces utilisateur (UI) en faisant glisser-déposer des composants sur les formulaires. Vous pouvez personnaliser ces composants et les étendre pour répondre aux besoins spécifiques de votre application.

4. **Développement pour Windows** : Delphi 7 est principalement axé sur le développement d'applications pour la plate-forme Windows. Vous pouvez créer des applications de bureau, des applications client-serveur, des services Windows, des pilotes de périphériques, etc.
5. **Prise en charge des bases de données** : Delphi 7 offre une prise en charge intégrée pour la connectivité aux bases de données. Vous pouvez accéder à différentes bases de données telles qu'Oracle, MySQL, Microsoft SQL Server, InterBase, etc., en utilisant les composants de base de données fournis.
6. **Compilation native** : Les applications développées avec Delphi 7 sont compilées en code natif, ce qui signifie qu'elles s'exécutent directement sur le système d'exploitation Windows sans nécessiter d'environnement d'exécution supplémentaire.
7. **Éditeur de code et débogage** : Delphi 7 dispose d'un éditeur de code puissant avec des fonctionnalités telles que la coloration syntaxique, l'achèvement automatique du code, la navigation dans le code et le débogage intégré pour faciliter le développement et le dépannage de vos applications.

Il est important de noter que Delphi 7 est une version plus ancienne et n'est plus activement maintenue par Embarcadero Technologies (l'actuel propriétaire de Delphi). Des versions plus récentes de Delphi sont disponibles, telles que Delphi 10.4.2, qui offrent des fonctionnalités améliorées, une prise en charge des plates-formes supplémentaires et des améliorations de performances.

### 2.2 Le langage de programmation Delphi :

Delphi est un langage de programmation orienté objet dérivé du Pascal. Il est largement utilisé pour le développement d'applications Windows et offre un environnement de développement intégré (EDI) convivial appelé Delphi IDE. Delphi a été développé par la société Borland, mais il est actuellement détenu et développé par Embarcadero Technologies.

Voici quelques caractéristiques clés du langage de programmation Delphi :

### 1. Syntaxe Pascal :

Delphi utilise une syntaxe similaire à celle du langage de programmation Pascal, ce qui le rend relativement facile à apprendre pour ceux qui ont une expérience antérieure avec Pascal ou d'autres langages similaires.

### 2. Programmation orientée objet :

Delphi prend en charge la programmation orientée objet (POO) avec des concepts tels que l'encapsulation, l'héritage et le polymorphisme. Les développeurs peuvent créer des classes, des objets et des méthodes pour organiser leur code de manière structurée.

### 3. Composants visuels :

Delphi propose une bibliothèque étendue de composants visuels (VCL) qui permet aux développeurs de créer rapidement des interfaces utilisateur (UI) en faisant glisser-déposer des composants sur les fenêtres de l'application. Les composants peuvent être personnalisés et étendus pour répondre aux besoins spécifiques de l'application.

### 4. Accès aux API Windows :

Delphi permet un accès facile aux API Windows, offrant ainsi une grande flexibilité pour interagir avec le système d'exploitation, accéder à des fonctionnalités spécifiques de Windows et tirer parti des ressources système.

### 5. Compilation native :

Les applications Delphi sont compilées en code natif, ce qui signifie qu'elles s'exécutent directement sur le système d'exploitation sans nécessiter d'environnement d'exécution supplémentaire.

### 6. Plateformes prises en charge :

Outre le développement d'applications Windows, Delphi prend également en charge le développement multiplateforme. Avec Delphi, vous pouvez développer des applications pour Windows, macOS, iOS, Android et Linux.

#### 7. Base de code partagée :

Delphi offre la possibilité de partager une partie du code source entre différentes plateformes, ce qui permet de réduire les efforts de développement et de maintenance pour les applications multiplateformes.

Delphi a été largement utilisé pour le développement d'applications de bureau, d'applications client-serveur, d'applications d'entreprise et d'applications mobiles. Il possède une communauté active de développeurs et une documentation complète pour vous aider à démarrer et à approfondir vos connaissances en Delphi.

### 3. Eléments du code source

Dans le code source, plusieurs éléments sont utilisés pour représenter les données :

#### 3.1. Structure de données :

- **Les images de profondeur (Trange)** : Elles sont de type texte, où chaque entrée représente la profondeur d'un pixel aux coordonnées (x, y). Les images de profondeur sont représentées par une matrice unidimensionnelle qui contient des valeurs réelles dans une certaine plage.
- **Les matrices de relief (Tplanes)** : Elles sont représentées par une matrice tridimensionnelle de taille [res \* res \* 3]. La troisième dimension de cette matrice correspond aux coefficients a, b, c de l'équation du plan (Tplan).

Le masque de lissage de Gauss utilisé est le suivant :

```
const
    Gausse : array[-2..2,-2..2] of integer = ((2,4,5,4,2),
                                              (4,9,12,9,4),
                                              (5,12,15,12,5),
                                              (4,9,12,9,4),
                                              (2,4,5,4,2)
                                              );
    sommeGausse=159;
```

Figure 4.1 : Masque de Gauss 5\*5 pour le lissage de l'image de profondeur

#### 3.2. Procédures de traitement

Dans notre application, nous avons utilisé plusieurs procédures de traitement, notamment :

- **Chargement** : Nous avons mis en place une procédure de chargement des données. Cela implique la création d'un flux de données entre l'application et le fichier texte contenant les valeurs réelles de l'image de profondeur. Nous parcourons ensuite le fichier texte ligne par ligne et remplissons la matrice de profondeur en convertissant les valeurs textuelles en valeurs réelles.

```
Unit1
procedure TForm1.displayRange;
var x,y:integer;
    gl:byte;
begin
    image1.Picture:=nil;
    image1.Width:=res;
    image1.Height:=res;
    for y:=0 to res-1 do
        for x:=0 to res-1 do
            begin
                gl:=range[y,x];
                image1.canvas.pixels[x,y]:=RGB(gl,gl,gl);
            end;
        end;
    end;

procedure TForm1.rangeLoadClick(Sender: TObject);
var ifile:file;
    i:integer;
begin
    res:=512;
    chdir(directory.Text);
    assignfile(ifile,imageName.Text);
    reset(ifile,1);
    blockread(ifile,range,32);
    blockread(ifile,range,512*512);
    closefile(ifile);
    for i:=1 to 2 do smoothing;
    displayRange;
end;
```

**Figure 4.2 : Chargement et affichage d'une image de profondeur brute**

- **Lissage adaptatif** : Nous avons implémenté une procédure de lissage adaptatif sur l'image de profondeur afin de réduire les bruits. Cette technique consiste à parcourir l'image en prenant en compte les pixels environnants du pixel central, à l'aide d'une

matrice 5\*5 par exemple. Nous calculons ensuite la moyenne des valeurs des pixels environnants, mais uniquement pour les pixels qui sont proches, avec un seuil de lissage adaptatif (Ebs). Cela permet de réduire les variations brusques de profondeur et d'obtenir une image lissée.

```
Unit1 |
-----
procedure TForm1.Smoothing;
var x,y,xc,yc:integer;
    produit,sommeGausse2:integer;
begin
    smoothed:=range;
    for y:=2 to res-1-2 do
        for x:=2 to res-1-2 do
            begin
                produit:=0;
                sommeGausse2:=0;
                for yc:=-2 to 2 do
                    for xc:=-2 to 2 do
                        if abs(range[y+yc,x+xc]-range[y,x])<Dtr then
                            begin
                                produit:=produit+range[y+yc,x+xc]*Gausse[yc,xc];
                                sommeGausse2:=sommeGausse2+Gausse[yc,xc];
                            end;
                        smoothed[y,x]:=trunc(produit/sommeGausse2);
                    end;
                range:=smoothed;
            end;
        end;
    end;
```

Figure 4.3 : Lissage de l'image par un filtre Gaussien 5\*5

L'entourage mais seulement pour les points proches et cela avec un seuil Dtr (lissage adaptatif).

- **Segmentation**

Dans notre travail, nous avons développé deux méthodes de segmentation pour traiter les images de profondeur. Les deux méthodes sont basées sur les contours.

Comme ça été montré au chapitre précédent, la première méthode (Figure 4.4) est basée sur la courbure des surfaces, alors que la seconde méthode (Figure 4.5) est basée sur le gradient des angles de surfaces dans l'image de relief.

```

Unit1
procedure TForm1.edgeDetectionClick(Sender: TObject);
var x,y,gx,gy,ng:integer;
    Ixx,Iyy,Ixy,Ix,Iy:integer;
    K,maxK,kx,ky:extended;
begin
    maxK:=-1.0;
    image2.Picture:=nil;
    for y:=1 to res-1-1 do
        for x:=1 to res-1-1 do
            begin
                Ixx:=range[y,x+1]-2*range[y,x]+range[y,x-1];
                Iyy:=range[y+1,x]-2*range[y,x]+range[y-1,x];
                Ixy:=range[y+1,x+1]-range[y,x+1]-range[y+1,x]+range[y,x];
                Ix:=range[y,x+1]-range[y,x-1];
                Iy:=range[y+1,x]-range[y-1,x];
                kx:=Ixx/((1+sqr(Ix))*sqr(1+sqr(Ix)));
                ky:=Iyy/((1+sqr(Iy))*sqr(1+sqr(Iy)));
                K:=sqr(sqr(kx)+sqr(ky));
                if K>maxK then maxK:=K;
                kGausse[y,x]:=K;
            end;
        for y:=1 to res-1-1 do
            for x:=1 to res-1-1 do
                begin
                    kGausse[y,x]:=kGausse[y,x]/maxK;
                    if kGausse[y,x]>0.05 then
                        image2.Canvas.Pixels[x,y]:=RGB(0,0,0);
                    end;
                end;
            end;
end;

```

Figure 4.4 : Détection de contours par la courbure à la surface

```

Unit1
procedure TForm1.AngleEdgeDetectionClick(Sender: TObject);
var x,y,gl:integer;
    a,b,c,a1,b1,c1:extended;
    gx,gy,ng:extended;
begin
    for y:=3 to res-1-3 do
        for x:=3 to res-1-3 do
            begin
                a:=Planes[y,x].a;
                b:=Planes[y,x].b;
                c:=-1;
                a1:=0; b1:=0; c1:=-1;
                Angles[y,x]:=(a*a1+b*b1+c*c1)/(sqr(a*a+b*b+c*c)*sqr(a1*a1+b1*b1+c1*c1));
                gl:=trunc(255*Angles[y,x]);
                image1.Canvas.Pixels[x,y]:=rgb(gl,gl,gl);
            end;
        image2.Picture:=nil;
        for y:=4 to res-1-4 do
            for x:=4 to res-1-4 do
                begin
                    gx:=Angles[y,x+1]-Angles[y,x-1];
                    gy:=Angles[y+1,x]-Angles[y-1,x];
                    ng:=sqr(gx*gx+gy*gy);
                    if abs(ng)>0.050 then
                        image2.Canvas.Pixels[x,y]:=RGB(0,0,255);
                    end;
                end;
            end;
end;

```

Figure 4.5 : Détection de contours par gradient des angles

### 4. Résultats expérimentaux

Dans notre étude, nous avons utilisé une base de données d'images de profondeur appelée ABW, qui comprend 30 images de test [Hoo96]. Nous avons effectué des expériences en utilisant certaines de ces images, en variant la complexité des objets présents dans les scènes représentées.

L'objectif de ces expérimentations était d'évaluer les performances de notre méthode de segmentation et d'amélioration sur des scènes d'images de profondeur avec différentes configurations et niveaux de complexité des objets. En utilisant un échantillon diversifié d'images de la base de données ABW, nous avons pu analyser comment notre approche se comportait face à des situations variées.

Les résultats expérimentaux nous ont permis de mesurer l'efficacité de notre méthode en termes de précision de la segmentation et de l'amélioration des résultats obtenus. Nous avons pu observer la capacité de la méthode basée relief à identifier les vrais contours dans les images segmentées.

Ces résultats expérimentaux nous ont fourni des indications sur les performances et les limites de notre approche, ainsi que des pistes pour d'éventuelles améliorations et ajustements nécessaires pour une segmentation plus précise et une meilleure représentation des scènes d'images de profondeur.

#### a) Image ABW. Test.8

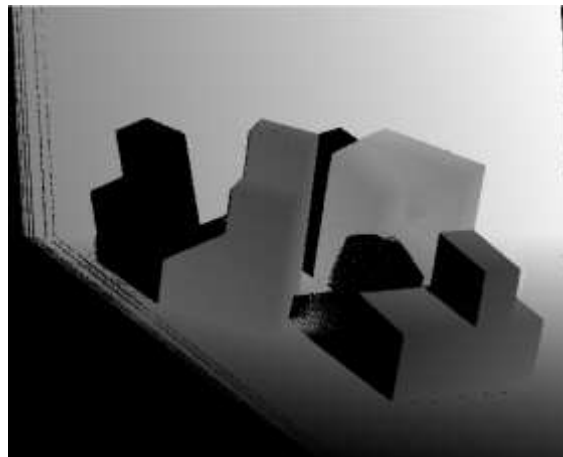


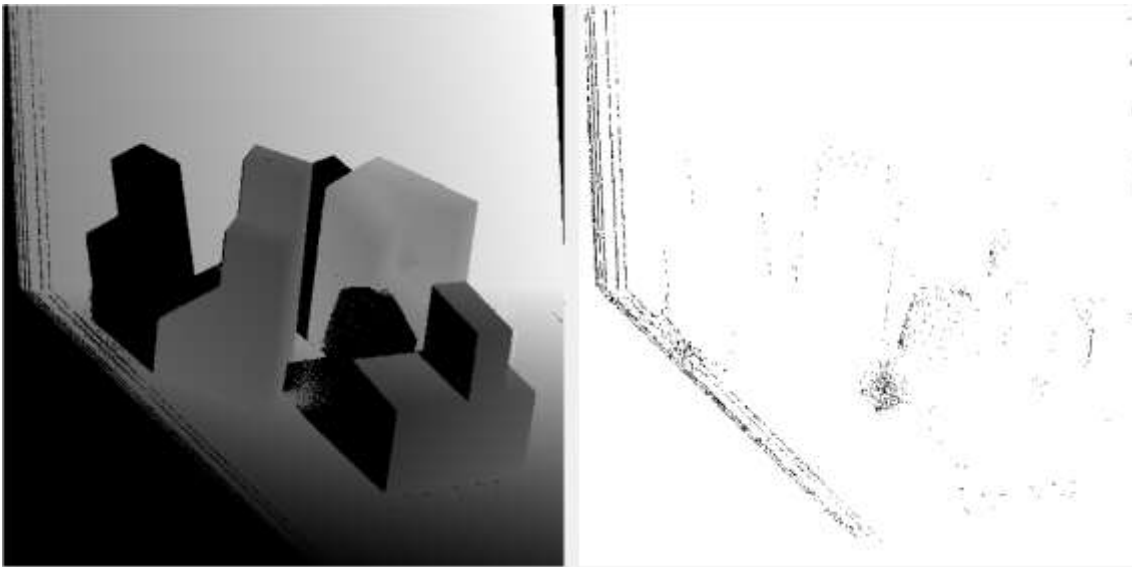
Figure 4.6 : Image niveau de gris de l'image de profondeur abw.test.8

## Chapitre 4 :

### Mise en œuvre et expérimentation

Dans la Figure 4.6, nous avons représenté l'image de profondeur "abw.test.8" en utilisant une image en niveaux de gris. Cette représentation nous a permis de détecter facilement les objets présents dans l'image ainsi que leurs textures. Cependant, il est difficile de détecter les surfaces, même pour l'œil humain, avec cette représentation.

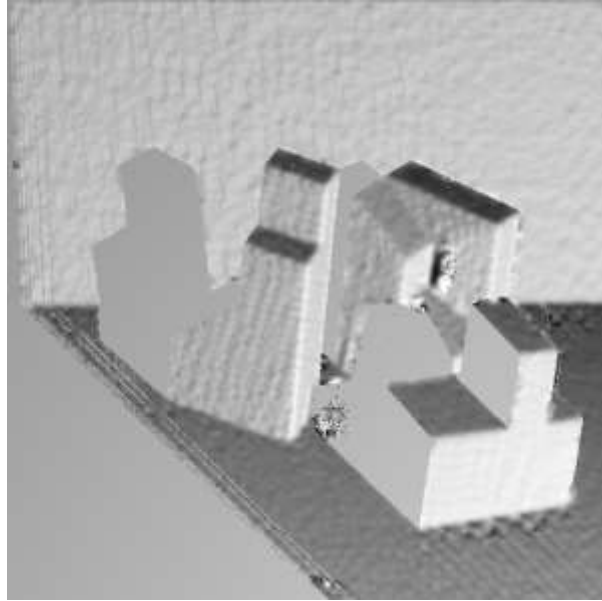
Dans la Figure 4.7, nous avons effectué une segmentation adaptative de l'image de profondeur en utilisant la courbure de l'image.



**Figure 4.7 : Résultat de segmentation de l'image abw.test.8 avec un détecteur basé courbure**

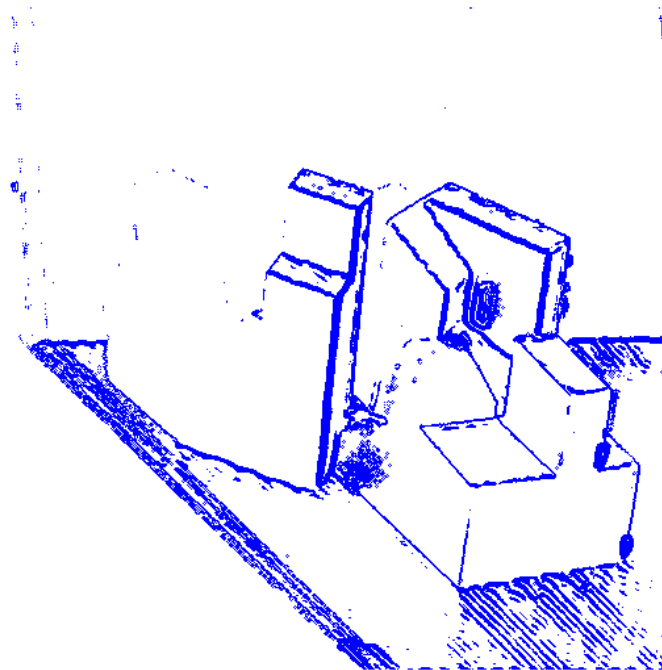
### Discussion :

Malgré la force de l'opérateur de la courbure de la surface, les résultats obtenus ne sont pas satisfaisant. En effet, ceci est dû aux fortes altérations des surfaces des objets lors de l'acquisition des images, comme c'est montré dans la figure 4.8 pour la même image abw.test.8.



**Figure 4.8 : Etat des surfaces de l'image abw.test.8**

Dans la figure 4.9, nous montrons les résultats de détection avec le gradient des angles sur l'image de relief.

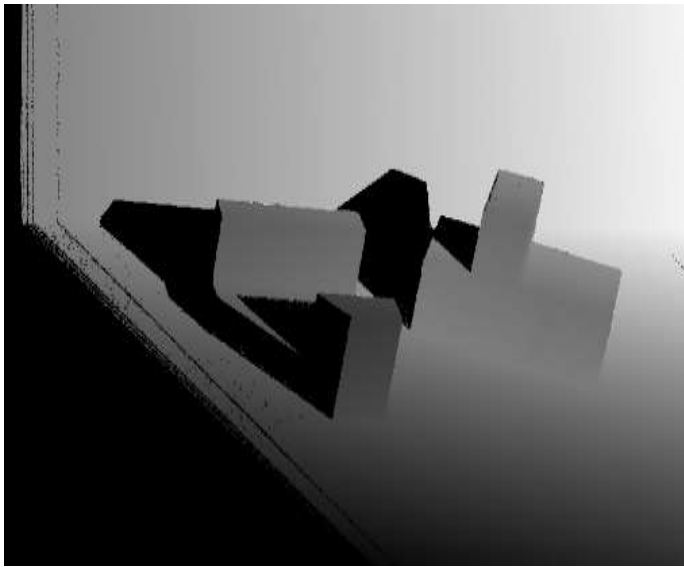


**Figure 4.9 : Etat des surfaces de l'image abw.test.8**

### Discussion :

Nous pouvons remarquer la nette amélioration des contours détectés dans cette image segmentée. Cette amélioration est certainement due à l'utilisation de l'image de relief, qui nous a permis de calculer le gradient des angles. Cette grandeur nous a permis d'améliorer les résultats de détection.

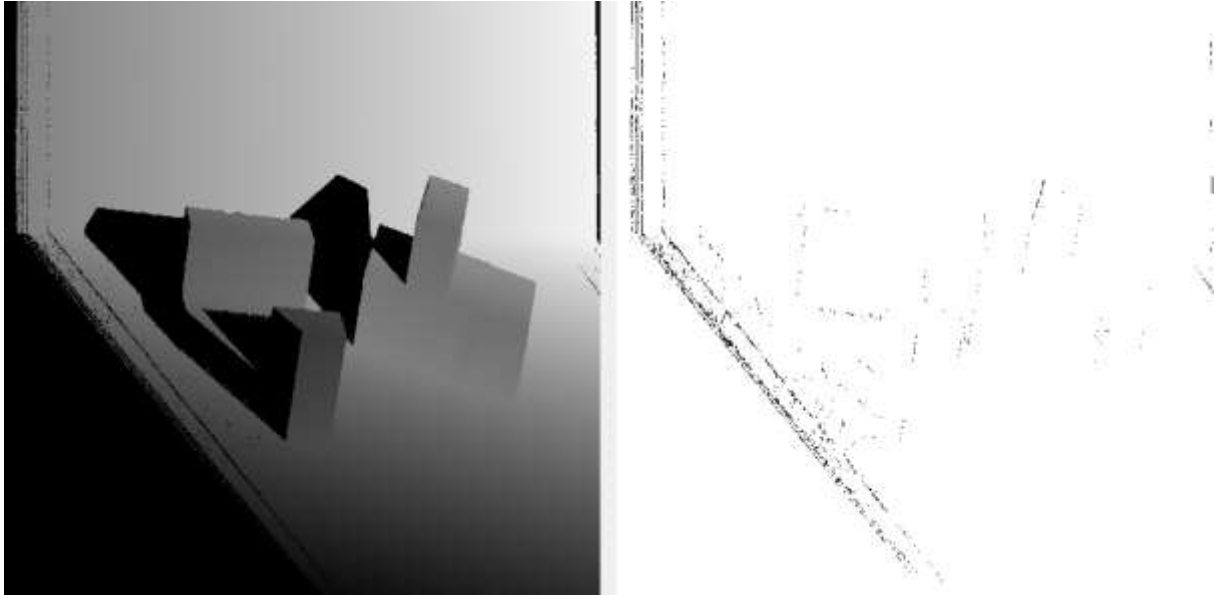
#### **b) Image ABW. Test.3**



**Figure 4.10 : Image niveau de gris de l'image de profondeur abw.test.3**

Dans la Figure 4.10, nous avons représenté l'image de profondeur "abw.test.3" en utilisant une image en niveaux de gris. Cette représentation nous a permis de détecter facilement les objets présents dans l'image ainsi que leurs textures. Cependant, il est difficile de détecter les surfaces, même pour l'œil humain, avec cette représentation.

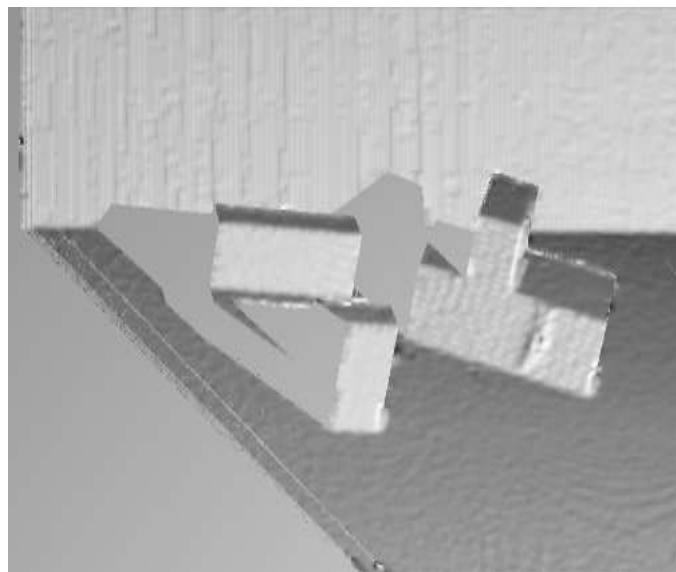
Dans la Figure 4.11, nous avons effectué une segmentation adaptative de l'image de profondeur en utilisant la courbure de l'image.



**Figure 4.11 : Résultat de segmentation de l'image abw.test.3 avec un détecteur basé courbure**

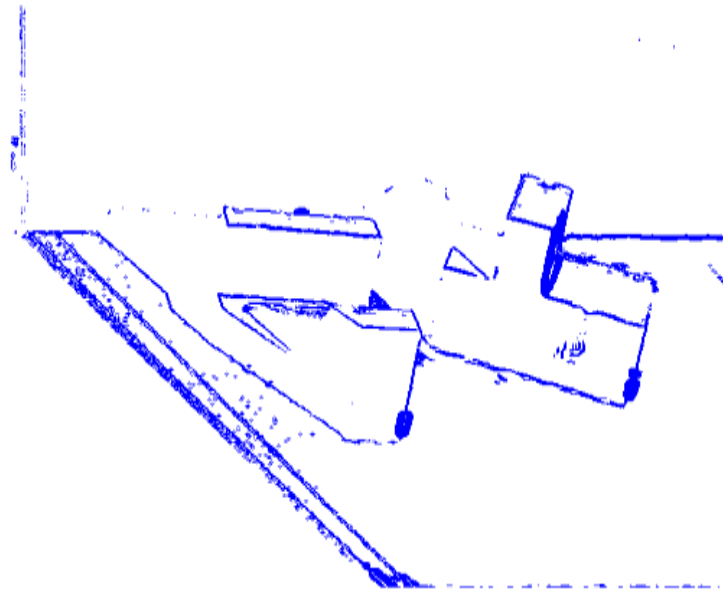
### Discussion :

Les résultats obtenus ne sont pas satisfaisants, malgré la puissance de l'opérateur de la courbure de la surface. Ce manque de satisfaction s'explique par les altérations importantes des surfaces des objets lors de l'acquisition des images, comme on peut le constater dans la figure 4.12 pour la même image abw.test.3.



**Figure 4.12 : Etat des surfaces de l'image abw.test.3**

Dans la figure 4.13, nous montrons les résultats de détection avec le gradient des angles sur l'image de relief.



**Figure 4.13 : Etat des surfaces de l'image abw.test.3**

### **Discussion :**

Il est évident que les contours détectés dans cette image segmentée ont considérablement été améliorés. Cette amélioration peut être attribuée à l'exploitation de l'image de relief, qui a permis le calcul du gradient des angles. L'utilisation de cette mesure a contribué à une amélioration significative des résultats de détection.

### 5. Conclusion

Dans ce dernier chapitre de notre mémoire, nous avons montré comment nos méthodes de détection de contours dans les images de profondeur ont été implémentées et testées. Nous avons commencé le chapitre par introduire l'environnement de développement, à savoir Delphi, puis nous avons montré quelques éléments du code en langage Pascal sous Delphi, avant de présenter les résultats des deux méthodes.

Il a été ressorti des résultats que la méthode basée sur le gradient des angles dans l'image de relief est largement supérieure à celle basée sur la courbure des surfaces dans l'image de profondeur brute.

Cependant, les résultats, même avec la méthode des gradients des angles sont peu satisfaisants, ce qui nous amène à chercher comment améliorer cette dernière.

# Conclusion générale

Dans notre mémoire de Master, nous avons exploré la problématique de la segmentation des images de profondeur des objets polyédriques. Nous avons réalisé des travaux de recherche sur les prétraitements nécessaires pour améliorer la qualité des images de profondeur et nous avons proposé deux approches spécifiques pour la segmentation de ces images.

Ces approches ont été développées dans le but de résoudre les défis liés à la segmentation des objets polyédriques à partir des images de profondeur. Nous avons étudié différents prétraitements pour traiter les données de profondeur, tels que le lissage et la normalisation des paramètres des équations de surface.

Nous avons également mis en place des méthodes de segmentation basées sur des techniques de contour en utilisant des approches mathématiques telles que la régression multiple. Nous avons évalué et comparé ces méthodes en termes de résultats de segmentation, de précision et de qualité visuelle.

En résumé, notre travail de recherche a porté sur la segmentation des images de profondeur des objets polyédriques en proposant des prétraitements adaptés et des approches de segmentation spécifiques. Ces travaux visent à améliorer la compréhension et la représentation des objets polyédriques à partir des données de profondeur, ouvrant ainsi des perspectives pour des applications dans des domaines tels que la vision par ordinateur et la robotique.

Dans la première approche de segmentation que nous avons développée, nous avons utilisé une représentation en niveau de gris des images de profondeur. Cette représentation permet de simplifier les données de profondeur en les convertissant en niveaux de gris, facilitant ainsi leur traitement ultérieur.

Ensuite, nous avons appliqué une segmentation adaptative sur l'image de profondeur prétraitée, après avoir effectué un lissage adaptatif. Le lissage adaptatif vise à réduire le bruit

et les variations indésirables présentes dans l'image de profondeur, afin d'obtenir une meilleure qualité pour la segmentation.

Une fois que nous avons obtenu l'image segmentée, nous avons procédé à une détection des contours. Cette étape permet de mettre en évidence les limites entre les différentes régions de l'image, ce qui est essentiel pour une segmentation précise des objets polyédriques.

La détection des contours peut être réalisée à l'aide de différentes techniques, telles que la dérivée de l'image ou des opérations de filtrage. L'objectif est d'identifier les variations abruptes dans les niveaux de gris de l'image et de les représenter comme des contours.

Cette première approche combine donc plusieurs étapes, l'utilisation de la profondeur pour la segmentation adaptative en passant par le lissage adaptatif, pour obtenir une image segmentée avec une détection des contours. Ces étapes visent à améliorer la qualité et la précision de la segmentation des objets polyédriques dans les images de profondeur.

Dans la deuxième approche de segmentation que nous avons développée, nous avons utilisé une représentation en image de relief. Cette représentation permet d'appliquer le gradient sur les angles des vecteurs normaux, ce qui permettra une détection plus fiables des contours dans les images.

Une fois que nous avons obtenu cette représentation en image de relief, nous avons appliqué une segmentation adaptative basée sur les contours. Cette méthode de segmentation vise à détecter les contours des objets en se basant sur les variations du gradient des angles dans l'image. Elle permet de distinguer les régions avec des contours nets des régions plus lisses.

En résumé, la deuxième approche de segmentation repose sur la représentation par image de relief, suivie d'une segmentation adaptative basée sur les contours en utilisant le gradient des angles. Cette approche vise à améliorer la précision de la segmentation en mettant l'accent sur les contours des objets polyédriques.

La réalisation d'une étude expérimentale pour comparer les deux méthodes de segmentation que nous avons proposées est une approche essentielle pour évaluer leur performance et leur efficacité. En utilisant un jeu d'images réelles provenant de la

bibliothèque ABW, vous avez pu évaluer les résultats de chaque méthode et comparer leur performance.

L'utilisation d'un jeu d'images réelles provenant de la bibliothèque ABW a renforcé la validité de notre étude expérimentale en utilisant des données du monde réel, ce qui a permis de mieux évaluer les performances des méthodes de segmentation.

En conclusion, il est tout à fait justifié et ambitieux de prévoir des extensions et des généralisations de notre travail actuel pour traiter des surfaces plus complexes et aborder des problèmes plus avancés en vision par ordinateur. Ces efforts permettront d'améliorer l'efficacité et la performance de la segmentation d'images de profondeur, tout en ouvrant la voie à des applications plus avancées dans différents domaines.

## Liste des références

**[Aya83]** : N.Ayache. Un système de vision bidimensionnelle en robotique industrielle. Thèse de docteur Ingénieur Université de Paris-Sud, 1983.

**[Bol79]**: R.C.Bolles. Robust Feature Matching Through Maximal Cliques. In Proceeding of the SPIE technical symposium on imaging applications for automated industrial inspection and assembly, volume 182, pages 14-149, 1979.

**[Bou98]**: A. BOUCHER, C. GARBAY," Des agents spécialisés pour la compréhension de séquences d'images ", RFIA'98, vol. II – p.275-284 (1998)

**[Che97]** : P.Checchin, L.Trassoudaine et J.Alizon, Segmentation d'images de profondeur : une approche pyramidale, Université de Technologie de Compiègne, BP 20529 – 60205 Compiègne cedex, FRANCE, 1997.

**[Coc95]** : J.-P. Cocquerez, S. Philipp, *Analyse d'images : filtrage et segmentation*, Paris, Masson, 1995.

**[Coh93]**: L. D. Cohen, I.Cohen, "Finite-element methods for active contour models and balloons for 2-D and 3-D images. » IEEE Trans. On PAMI 15(11) pp. 1131{1147, November 1993.

**[Col91]** : J.Colly. Reconnaissance d'Objets Articules Tridimensionnels A Partir d Informations Multi-sensorielles Pour Des Applications Robotiques, Thèse Doctorat de l Université Paul Sabatier de Toulouse, 1991.

**[Cou91]** : V.Coutance. La couleur en vision par ordinateur-Application à la robotique- Thèse de l'université Paul Sabatier, Toulouse (France), Laboratoire d'automatique et d'analyse des systèmes (C.N.R.S), Janvier 1991.

**[Fla99]**: Flavio Prieto. Méthodologie assistés par ordinateur : Apport des capteurs 3D sans contacte, thèse de l'institut des sciences appliquées de LYON, 13 Décembre 1999.

**[Gat01]** : H.Benoit-Cattin, T.Zouagui and C. Odet, " une vision fonctionnelle de la segmentation d'images." In : Congrès Francophone de Vision par Ordinateur, ORASIS, 5-8 juin, Cahors, France. 2001, pp. 425-433.

**[Har90]:** Harrison, D.D. and Weir, M.P. High-speed triangulation-based 3D imaging with orthonormal data projections and error detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, April 1990, Vol.12, N°4, p.409-416.

**[Hat03]:** J.Hatamzadeh-Tabrizi, "Using Active Contours for Segmentation of Middle-Ear-Images", master thesis, 2003.

**[Her91]:** M.Herrb. Vision en mouvement pour la robotique mobile. Thèse de l'Université Paul Sabatier, Toulouse(France) 91048, Laboratoire d'Automatique et d'analyse des systèmes (C.N.R.S), Février 1991.

**[HMai]:** H.Maître, "La segmentation par régions", cours <http://www.tsi.enst.fr/~bloch>

**[Hoo96]:** Hoover, A., Jean-Baptiste, G., Jiang, X., Flynn, P. J., Bunke, H., Goldgof, D. B., Bowyer, K. W., Eggert, D. W., Fitzgibbon, A. W. et Fisher, R. B. (1996). An experimental comparison of range image segmentation algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7):673–689.

**[Jia00]:** X. Jiang, "An adaptive contour closure algorithm and its experimental evaluation." *IEEE Trans. on PAMI* 22(11) pp. 1252{1265, November 2000.

**[Jia03]:** Y. Lu, T. Jiang, and Y. Zang, "Region growing method for the analysis of functional MRI data," *NeuroImage* 20(1), pp. 455–465, 2003.

**[Lau99] :** G.Laurence. « Trois principes de coopération pour la segmentation en imagerie de résonance magnétique cérébrale". PH Thespis / IMAG, UNIVERSITE JOSEPHFOURIER - GRENOBLE I. TIMC (10 Octobre 1999).

**[Maz96]:** S. Mazouzi. Reconnaissance d'objets 3D complexes à partir d'images denses. Thèse de magister, Université de Constantine, (1996).

**[Maz07] :** S.Mazouzi, et M.Batouche, (2007b). Range image segmentation by randomized region growing and bayesian edge regularization. *Journal of Computer Science*, 3(5): 410–417.

**[Maz08] :** S.Mazouzi, Reconnaissance De Formes Par Les Systèmes Auto-Organisés, Thèse Doctorat de l Université Mentouri de Constantine, 2008.

**[Maz09]** : Dr. Mazzouzi Smail, article de TSI 2009 « Une approche multi-agents pour la segmentation d'images de profondeur à base d'objets polyédriques »

**[Mon88]**: O.Monga & R.Deriche. 3D Edge Detection Using Recursive Filtering: Application to Scanner Images. Rapport technique 390, Institut National de Recherche en Informatique et en Automatique, Domaine de voluceau Rocquencourt B.P.105 78153 Le Chesnay France, Novembre 1988.

**[Mon89]**: O.Monga, R.Deriche, G.Malandrin & J.P.Cocquerez. Recursive Filtering and Edge Closing: Two Primary Tools for 3D Edge Detection. Rapport technique 1103, Institut National de Recherche en Informatique et en Automatique, Domaine de voluceau Rocquencourt B.P.105 78153 Le Chesnay France, Aout 1989.

**[NSal17]** : Salima Nebti. Optimisation par Ecosystemes Artificiels : Application à la segmentation d'images, thèse de magistère de l'université mentouri de Constantine faculté des sciences de l'ingénieur département d'informatique.

**[Pam90]**: L.H.Pampagnin. Reconnaissance d'objets tridimensionnels en perception monoculaire et multisensorielle- Application à la robotique spatiale-. Thèse de l'Université Paul Sabatier, Toulouse (France) 779, Laboratoire d'Automatique et d'Analyse des Systèmes (C.N.R.S.), Octobre 1990.

**[Pul97]**: K. Pulli, M. PietikÄäinen, "Surface Modeling and Display from Range and Color Data", ICIAP'97, Florence 1997, pp. 385-397.

**[Sah98]**: P. K. Sahoo, S.Soltani, Wong, A. K. C., and Chen, Y. C."A survey of thresholding Techniques." CVGIP 41(2) pp. 233{260, February 1998.

**[Vin86]** : A. Robert De Saint Vincent. Perception et modélisation de l'environnement d'un robot mobile : une approche par stéréovision. Thèse de l'université Paul Sabatier, Toulouse (France), Laboratoire d'Automatique et d'Analyse des Systèmes (C.N.R.S.), Toulouse (France), Novembre 1986.

**[Wel96]**: W.M. Wells, L. Grison, R. Bikinis, and F.A. Jobless. "Adaptative Segmentation of MRI data", *IEEE Transactions on Medical Imaging*, 15(4), pp. 429–442, 1996.