Ministère de l'enseignement Supérieur et de la recherche Scientifique
وزارة التعليم العالي والبحث العلمي

Badji Mokhtar Annaba University
Université Badji Mokhtar – Annaba
Faculté de Technologie

جامعة باجي مختار – عنابـــــــــة
كلية التكنولوجيا

Département : Informatique
قســـــــم: الإعلام الآلي

Thèse

Présentée pour obtenir le diplôme de

Doctorat En-Sciences

Spécialité : Informatique

Par :
CHERIBET MOHAMED

Thème :

Approches cognitives en vision robotique

Thèse soutenue le 04/06/2026 devant le jury composé de :

N°	Nom et prénom	Grade	Etablissement	Qualité
01	Farah Nadir	Prof	Université Badji Mokhtar - Annaba	Président
02	Mazouzi Smaine	Prof	Université 20 Août 1955 - Skikda	Rapporteur
03	Bounour Nora	Prof	Université Badji Mokhtar - Annaba	Examinateur
04	Laimeche Lakhdar	Prof	Université Larbi Tebessi - Tebessa	Examinateur
05	Khedairia Soufiane	MCA	Université M.C Messaadia - Souk Ahras	Examinateur
06	Benmachiche Abdelmadjid	MCA	Université Chadli Bendjdid – El Taref	Examinateur

Dédicace

À la mémoire de mon père,

Qui, par ses valeurs, sa sagesse et son exemple, continue d'éclairer mon chemin.

Tu restes à jamais dans mon cœur et dans chacun de mes accomplissements.

À ma mère,

Pour son amour inlassable, sa force et ses prières silencieuses.

À ma famille,

Pour leur soutien constant, leur patience et leur foi en moi.

À mes enseignants et encadrants,

Pour leur accompagnement, leurs conseils, et l'inspiration qu'ils m'ont donnée.

À toutes celles et ceux qui ont contribué, de près ou de loin, à la réalisation de cette thèse,

Je vous dédie ce travail avec gratitude.

Remerciements

Au terme de ce travail de recherche, je tiens à exprimer ma profonde gratitude à toutes celles et ceux qui ont contribué, de près ou de loin, à la réalisation de cette thèse.

Je remercie tout d'abord mon directeur de thèse, Pr. Mazouzi Smaine, pour sa confiance, sa disponibilité, ses conseils éclairés et son accompagnement tout au long de ces années. Son exigence scientifique et sa bienveillance ont été des sources constantes de motivation.

Je suis également reconnaissant aux membres du jury pour l'honneur qu'ils me font en acceptant d'évaluer ce travail et de participer au jury de soutenance.

Mes remerciements vont aussi à mes collègues, enseignants et chercheurs, pour les échanges stimulants, l'ambiance de travail agréable et leur soutien.

Je n'oublie pas mes amis et camarades de doctorat, avec qui j'ai partagé des moments de doute, de fatigue, mais aussi de joie et de solidarité.

À ma famille, je veux adresser mes remerciements les plus sincères.
À ma mère, pour son amour, sa patience et sa force.
À mon père, dont la mémoire m'accompagne chaque jour.
À tous les membres de la famille, pour leur soutien indéfectible.

Enfin, merci à toutes les personnes qui ont, d'une manière ou d'une autre, contribué à cette aventure humaine, intellectuelle et scientifique.

Résumé

Peu de travaux de recherche ont opté pour la détection de contours dans les images de profondeur, comme méthodes préalable à la segmentation d'images. Il a été noté dans la littérature que la détection de contours dans ce type d'images est fortement problématique. Ceci est dû à deux raisons principales : Le haut niveau de bruit et de déformations dans les images de profondeur ; et l'Inclinaison des surfaces géométriques par rapport à l'angle de vue. Cependant, la détection de contours demeure avantageuse, notamment pour sa performance en temps de calcul, comparée aux autres approches de segmentation, basées régions.

Dans le cadre de cette thèse, et dans l'objectif de contribuer aux méthodes et techniques en vision robotique, nous avons proposé une nouvelle formulation de détection de contours dans les images de profondeur, support d'excellence en vision robotique. Nous avons aussi proposé une méthode de régularisation bayésienne des contours détectés. La combinaison des deux méthodes permet d'améliorer significativement la détection de contours et par conséquent la segmentation des images en surfaces composant les objets géométriques qui figurent dans ces images.

Ainsi, un nouveau détecteur de Canny, adapté aux images de profondeur, est proposé. Il est basé sur une nouvelle formulation du calcul du gradient de l'image, utilisant les vecteurs normaux aux surfaces, au lieu de la profondeur brute de l'image. Les contours détectés sont ensuite régularisés par une formulation markovienne bayésienne, impliquant une définition bien appropriée des cliques et des potentiels, utilisés dans le Framework MAP-MRF.

Les résultats de détection obtenus, ont pu montrer le fort potentiel des méthodes proposées pour une extraction fiable des cartes de contours dans les images de profondeur.

Mots-clés : Images de profondeur, Segmentation d'images, Détecteur de Canny, Régularisation bayésienne, MAP-MRF.

Abstract

Few research studies have opted for edge detection in range images as a preliminary method for image segmentation. It has been noted in the literature that edge detection in this type of image is highly problematic. This is due to two main reasons: the high level of noise and distortion in range images; and the inclination of geometric surfaces that depends on the viewing angle. However, edge detection remains advantageous, particularly for its computational performance, compared to other region-based segmentation approaches.

As part of this thesis, and with the aim of contributing to methods and techniques in robotic vision, we proposed a new formulation for edge detection in range images, a tool of excellence in robotic vision. We also proposed a bayesian regularization method for the detected edges. The combination of the two methods significantly improves edge detection and, consequently, the segmentation of images into surfaces composing the geometric objects contained in these images. Thus, a new Canny detector, suitable for range images, is proposed. It is based on a new formulation for calculating the image gradient, using surface normal vectors instead of the raw image depth. The detected edges are then regularized using a Markovian-Bayesian formulation, involving a well-suited definition of cliques and potentials used in the MAP-MRF framework.

The obtained detection results demonstrated the strong potential of the proposed methods for reliable extraction of edge maps in range images.

Keywords: Range Images, Image Segmentation, Canny Detector, Bayesian Regularization, MAP-MRF.

ملخص

اختارت دراسات بحثية قليلة كشف الحواف في صور العمق كطريقة أولية لتجزئة الصورة. وقد لوحظ في الأدبيات أن كشف الحواف في هذا النوع من الصور يمثل مشكلة كبيرة. ويرجع ذلك إلى سببين رئيسيين: ارتفاع مستوى الضوضاء والتشويه في صور العمق؛ وميل الأسطح الهندسية الذي يعتمد على زاوية الرؤية. ومع ذلك، يظل كشف الحواف مفيدًا، لا سيما لأدائه الحسابي، مقارنةً بأساليب التجزئة الأخرى القائمة على المنطقة.

وكجزء من هذه الرسالة، وبهدف المساهمة في أساليب وتقنيات الرؤية الروبوتية، اقترحنا صيغة جديدة لكشف الحواف في صور العمق، وهي أداة متميزة في الرؤية الروبوتية. كما اقترحنا طريقة تنظيم بايزية للحواف المكتشفة. ويؤدي الجمع بين الطريقتين إلى تحسين كشف الحواف بشكل كبير، وبالتالي تجزئة الصور إلى أسطح تُشكل الأجسام الهندسية الموجودة في هذه الصور. وبالتالي، يُقترح كاشف كاني جديد، مناسب لصور العمق. يعتمد هذا النظام على صيغة جديدة لحساب تدرج الصورة، باستخدام الاشعة العمودية على الأسطح بدلاً من عمق الصورة الخام. ثم تُنظّم الحواف المكتشفة باستخدام صيغة ماركوفية-بايزيانية، تتضمن تعريفًا مناسبًا للمجموعات والجهود المستخدمة في إطار عمل MAP-MRF.

أظهرت نتائج الكشف المُحصلة الإمكانات القوية للطرق المقترحة لاستخراج خرائط الحواف في صور العمق بدقة.

كلمات مفتاحية: صور العمق، تجزئة الصورة، كاشف كاني، التنظيم البايزي، MAP-MRF.

Table des matières

Liste des figures

Liste des tableaux

Liste des abréviations

IP	Image de Profondeur
LiDAR	Light Detection And Ranging
2D	Deux Dimensions
2.5D	2.5 Dimensions
3D	Trois Dimensions
RGB	Red Green Blue
RGB-D	RGB + Depth
U-Net	Architecture de réseau en U
IoU	Intersection over Union
mIoU	mean Intersection over Union
ToF	Time-of-Flight
IR	Infrarouge
NLSPN	Non-Local Spatial Propagation Network
MAE	Mean Absolute Error
SVM	Support Vector Machine
CNN	Convolutional Neural Network
R-CNN	Region-based Convolutional Neural Network
GPU	Graphics Processing Unit
MRF	Markov Random Field
EM	Expectation-Maximization
MLS	Moving Least Squares
IRM	Imagerie par Résonance Magnétique
IA	Intelligence Artificielle

CAD	Computer-Aided Design
MAP	Maximum A Posteriori
GRF	Gibbs Random Field
MLL	Multi-Level Logistic
FRAME	Filter, Random Fields And Maximum Entropy
MRMRF	Multi-Resolution Markov Random Field
MRR-MRF	Multi-Region Resolution Markov Random Field
DHMRF	Dynamic Hybrid Markov Random Field
HMRF	Hidden Markov Random Fields
GGMRF	Generalized Gauss Markov Random Field
SAR	Synthetic Aperture Radar
FCM	Fuzzy C-Means
k-NN	k-Nearest Neighbor
ML	Maximum Likelihood
M-HSEG	Marker-based Hierarchical SEGmentation

Chapitre 1
Introduction générale

1.1 Contexte et problématique

En quasi-totalité des applications, l'interprétation des images demeurent problématique et les chercheurs du domaine de vision continuent de produire des méthodes et des modèles dont l'objectif est de produire des interprétations fiables des contenus des images. Depuis plus de quatre décennies, un nombre considérable de travaux ont été publiés, traitant différentes problématiques en traitement automatique d'images [113]. Le principal objectif en traitement des images est de comprendre leur contenu, et ce par la reconnaissance des différents objets qui y figurent ou la délimitation des régions d'intérêts, dont l'interprétation en dépend. Par ailleurs, et vu les différents modèles de vision par ordinateur qui ont été mis en œuvre et publiés dans la littérature [114], la segmentation occupe la pierre angulaire de tout modèle ou méthode de détection ou de reconnaissance d'objets [115]. Cependant, la segmentation d'image a commencé et a demeuré un problème difficile, considéré carrément comme un problème mal posé par différents auteurs [116]. L'origine à ce problème est multiple : D'une part l'image est de nature discrète impliquant une discrétisation par échantillonnage du signal analogique continu, ce qui en résulte en une perte de l'information, et la production d'image entachées d'imprécisions. D'autre part, les organes électroniques et optiques d'acquisition, ainsi que les conditions de prise engendrent des erreurs sous formes de bruit et de déformation, engendrant de leur part des données image incertaines.

Certains types d'images sont plus susceptibles pour en contenir plus de bruit et de déformations. Outres les images médicales, dont l'information visuelle est obtenu indirectement, et dans des conditions difficiles à contrôler [117], les images de profondeurs sont entachées de niveau très élevés de bruit et de déformation [118]. En effet, les images de profondeur sont acquises généralement selon des techniques complexes impliquant des scanners laser, et dont l'estimation des profondeurs des points scannés engendre des imprécisions importantes, vu que le temps de vol du laser, ou les angles calculés lors de la

triangulation sont difficiles à mesurer avec une grande précision [4, 14]. Malgré les avancées incessantes en termes d'acquisition d'images de profondeur, les images qui en résultent demeurent hautement bruitées pour que des méthodes classiques de segmentation ou de reconnaissance puissent être directement appliquées.

Pour de telles images, hautement bruitées et déformées, les techniques classiques d'amélioration d'images, telles que le filtrage de bruit par lissage, s'avèrent inefficaces. En effet, les filtres classiques d'atténuation de bruit, s'ils sont appliqués d'une manière soft sur des images hautement bruitées, le bruit résiduel entrave tout post-traitement, dont la segmentation. Par contre si les filtres sont appliqués d'une manière intense, le bruit sera effectivement réduit, mais de vrais contours de l'image seront effacés, et ce qui aboutit à de mauvais résultats de segmentation.

1.2 Objectifs et contributions

Dans le cadre de cette thèse, nous nous sommes attaqués à la problématique citée en haut, à savoir la segmentation des images de profondeur, considérées hautement bruitées. Pour ce faire, nous avons opté pour la détection de contours comme approche de segmentation, en proposant une nouvelle formulation adaptée, aux images de profondeur, du célèbre filtre de détection de Canny, proposé et formulé initialement pour les images à niveau de gris. Nous commençons par montrer les spécificités des données dans une image de profondeur, d'où l'inadéquation du filtre de Canny originel pour ce type d'images. Nous proposons alors une modélisation géométrique des données brutes afin de produire une nouvelle image dont le contenu permet d'appliquer les étapes du filtre de Canny et ainsi produire la carte des contours.

Contrairement à l'utilisation conventionnelle de Canny, où le vecteur de gradient à chaque pixel de l'image est calculé sur la base des données d'image brutes (couleur ou niveau de gris), nous utilisons des angles formés par des vecteurs normaux adjacents à la surface pour calculer à la fois l'amplitude et la direction du gradient. Un tel calcul du gradient permet d'avoir des magnitudes élevées sur les pixels qui se trouvent sur les bords séparant les régions. Cependant, les amplitudes du gradient sont faibles sur les pixels dans les régions homogènes. Dans les images de profondeur, les pixels adjacents qui appartiennent à des surfaces très inclinées ont des valeurs de profondeur éloignées qui ne permettent pas d'utiliser la distance brute comme donnée pour calculer le gradient. Néanmoins, les

orientations des vecteurs normaux sont proches dans ce type de surfaces, ce qui permet de les prendre en considération pour la détection de contours ou l'extraction de régions.

L'utilisation du modèle proposé et l'adaptation de l'algorithme de Canny ont permis d'avoir des résultats meilleurs, comparés aux différents détecteurs appliqués sur les données brutes, dont Canny lui-même. Cependant, les résultats demeurent insatisfaisants, à cause des déformations de contours obtenus, dues à la base au bruit résiduel dans les images lissées. Pour améliorer les résultats de détection, nous avons opté selon une approche régularisation, basée minimisation d'énergie et utilisant des modèles de Markov. A cette fin, nous avons modélisé les contours obtenus par un modèle aléatoire de Markov (MRF Markov Random Field), en définissant les cliques et les potentiels correspondants, dont la minimisation d'énergie, ainsi formulé, permet d'obtenir des contours lisses, dépourvus des irrégularités dues aux bruit dans l'image de profondeur. Les résultats obtenus, après régularisation ont montré l'intérêt de cette dernière, et les contours obtenus ont été significativement améliorés, de sort que les traitements ultérieurs de détection et de reconnaissance peuvent être appliqués.

Il a été constaté, à notre connaissance, qu'aucun travail publié n'a abordé la détection de contours dans les images de profondeur. Nous avons constaté au début de nos travaux qu'une méthode basée contour ne pouvait être appliquée directement aux images de profondeur, vu le niveau élevé du bruit et de déformation dans de telles images. Néanmoins, avec une modélisation adéquate des données images et une adaptation d'un filtre classique de détection, à savoir Canny, ainsi qu'un modèle markovien basé régularisation, nous avons pu développer un détecteur de contours propre aux images de profondeur. Les résultats expérimentaux en utilisant une base d'image de profondeur dédiée à l'évaluation de performance des méthodes de segmentation de ce type d'images, ont montré que la segmentation ainsi opérée est fiable et utile pour la suite du processus d'interprétation.

1.3 Organisation du manuscrit

Le chapitre 2 est intitulé « **Segmentation d'images de profondeur** », dans lequel nous commençons par la définition d'une image de profondeur, ses spécificités et ses techniques d'acquisition. Ensuite, nous entamons la segmentation des images de profondeur par la présentation des différentes approches de segmentation en général, à savoir les méthodes basées contours, les méthodes basées régions et les méthodes par classification. Nous

consacrons le reste du chapitre à la littérature de la segmentation des images de profondeur, et ce, en introduisant les principaux travaux ayant traité du sujet.

Quant au chapitre de la littérature, il est intitulé « **Régularisation en segmentation d'images** ». Nous présentons dans ce chapitre les différentes techniques de régularisation, notamment celles basées sur la minimisation d'énergie. A l'occasion de ceci, nous présentons les différents modèles markoviens utilisés en traitement d'images. Certains travaux, ayant traité de la régularisation par les modèles markoviens sont également présentés.

Le chapitre 4 présente notre première contribution, à savoir, une nouvelle méthode de détection de contours dans les images de profondeur. Le chapitre est intitulé « **Adaptation du filtre de Canny pour les images de profondeur** », où nous présentons le nouveau détecteur adapté. Son principe est inspiré du détecteur Canny, ainsi les caractéristiques inhérentes aux images de profondeur sont considérées. Habituellement, le détecteur de Canny est utilisé avec des images en niveau de gris ou en couleur. Cependant, son application directe avec des images de profondeurs ne donne pas de résultats satisfaisants. Nous montrons qu'à partir de l'image brute, contenant les profondeurs mesurées, une image de relief constituée d'une image de vecteurs normaux aux surfaces locales est calculée. Ainsi, les angles entre les vecteurs voisins sont utilisés pour calculer un gradient basé sur l'angle. Ce dernier est intégré dans l'algorithme de Canny, permettant de produire une carte des contours dans les images de profondeur. Nos expérimentations en utilisant des images réelles de la base de données ABW montent que le nouveau détecteur proposé a surpassé celui de Canny d'origine et d'autres détecteurs classiques.

Le dernier chapitre du manuscrit présente notre seconde contribution, qui consiste à l'amélioration des résultats de détection par régularisation des contours obtenus. Il est intitulé « **Amélioration basée régularisation des contours dans les images de profondeur** ». Nous proposons dans ce chapitre une formulation markovienne du *smootness* des contours, en adoptant les cliques et les potentiels correspondants, bien appropriés. Nous montrons lors de l'expérimentation que la régularisation des contours par la méthode proposée, a permis d'améliorer significativement les contours détectés.

Nous terminons le manuscrit par une **conclusion générale**, dans laquelle nous revenons sur nos contributions et nos résultats, et nous soulignons à la fin, de potentielles perspectives à nos travaux.

Chapitre 2

Segmentation d’images de profondeur

2.1 Introduction

2.1.1 Contexte et enjeux des images de profondeur

Les images de profondeur, représentant la distance des objets à un capteur, ont révolutionné la perception 3D dans des domaines critiques :

- **Robotique** : Navigation autonome et manipulation d’objets grâce à des capteurs LiDAR et Kinect [1].
- **Santé** : Reconstruction 3D d’organes pour la chirurgie assistée (ex : *Da Vinci Surgical System*) [2].
- **Réalité augmentée** : Fusion réaliste d’objets virtuels et réels via des caméras RGB-D [3].

Cependant, leur utilité dépend fortement de leur qualité d’acquisition et de leur segmentation précise, étapes souvent compromises par des défis techniques majeurs.

2.1.2 Problématique : Défis de segmentation

La segmentation des images de profondeur se heurte à trois obstacles principaux :

1. **Bruit et artefacts** :
    - Les capteurs grand public (ex : Kinect) génèrent des erreurs de mesure dues aux réflexions multiples ou aux surfaces non-Lambertiennes [4].
    - Exemple : Jusqu’à 15 % de pixels erronés dans les scènes intérieures complexes [NYU Depth V2 Benchmark, 2012].
2. **Discontinuités géométriques** :
    - Les occultations et les transitions abruptes entre surfaces (ex : bords d’objets) compliquent la détection de contours cohérents [5].

3. **Complexité des scènes** :
    - Les environnements non structurés (ex : entrepôts logistiques) exigent des méthodes robustes aux variations d'échelle et d'éclairage [6].

2.1.3 Objectifs du chapitre

Ce chapitre vise à :

1. **Synthétiser l'état de l'art** :
    - Analyser les méthodes classiques (ex : *split and merge* [7]) et modernes (ex : réseaux *U-Net 3D* [8]).
    - Comparer leurs performances via des benchmarks reconnus (ex : IoU sur ScanNet [9]).
2. **Identifier les lacunes** :
    - Souligner la dépendance aux données annotées des approches par apprentissage profond [10].
    - Révéler le manque de généralisation des méthodes géométriques sur des scènes dynamiques [11].
3. **Préparer le terrain pour des contributions innovantes** :
    - Explorer des pistes hybrides combinant apprentissage profond et contraintes topologiques [12].

2.1.4 Structure du chapitre

Ce chapitre s'articule comme suit :

- **Sections 2–3** : Fondements théoriques et techniques d'acquisition.
- **Sections 4–5** : Défis techniques et revue critique des méthodes de segmentation.
- **Sections 6–7** : Post-traitement/régularisation et conclusion.

2.2 Concepts Fondamentaux

2.2.1 Image de profondeur : Définition et structure

Une **image de profondeur** (ou *depth map*) est une représentation 2D régulière où chaque pixel (i, j) encode la distance $Z(i, j)$ entre le capteur et la surface observée (Figure 2.1).

Formalisation mathématique :

$P = \{p_{i,j}\}$ avec $p_{i,j} \in \mathrm{R3} \cup \{NIL\}$

- $p_{i,j} = (x, y, Z)$: Coordonnées 3D du point projeté sur le plan image.
- NIL : Valeur invalide (absence de mesure, ex : occultations).

La cohérence spatiale est une propriété importante où les pixels voisins dans l'image correspondent à des points 3D voisins dans l'espace réel, sous réserve que le faisceau de mesure soit quasi-parallèle [4]. (Figure 2.2)

Exemple visuel :

Figure 2.1 - Image de profondeur : À gauche, une image de profondeur (niveaux de gris). À droite, l'image RGB correspondante [5].

2.2.2 Comparaison avec les nuages de points

Tableau 2.1 - Image de profondeur vs Nuage de points : Avantages des représentations structurées en vision 3D [13].

Critère	**Image de profondeur**	**Nuage de points**
Structure	Grille 2D régulière	Ensemble 3D non structuré
Cohérence spatiale	Préservée (capteur parallèle)	Dépend de l'acquisition
Traitement algorithmique	Compatible avec les méthodes 2D	Requiert des techniques 3D spécifiques (ex : voxélisation)
Applications	Segmentation rapide, robotique temps réel	Modélisation précise, CAD

2.2.3 Image RGB-D : Fusion de modalités

Une image RGB-D combine :

- RGB : Informations chromatiques (3 canaux).
- Depth : Profondeur (1 canal).

Formats standard :

- NYU Depth V2 : Résolution 640×480, annotations sémantiques [5].
- ScanNet : Scènes intérieures annotées, utilisé pour l'entraînement de réseaux de neurones [9].

Applications typiques :

- Reconstruction 3D (ex : *BundleFusion* [9]).
- Segmentation sémantique pour la robotique mobile [11].

2.2.4 Notation et conventions

- **Système de coordonnées** :
  - Origine au capteur.
  - Axe Z aligné avec l'axe optique.
- **Échantillonnage** :
  - $x_i = \Delta x \cdot i,\ y_j = \Delta y \cdot j$ pour les images cartésiennes orthonormées.

2.2.5 Profondeur d'une image : Définition physique

La profondeur Z est calculée via :

- Capteurs actifs (ex : LiDAR) : $Z = (c\ \Delta t)/2$
  Où c est la vitesse de la lumière.
- Triangulation (ex : Kinect) : $Z = (B\ f)/d$
  avec B = base stéréo, f = focale, d = disparité.

Limites des capteurs :

- Kinect v2 : Précision ±2% à 2 m [4].
- LiDAR : Précision sub-centimétrique, mais coût élevé.

2.2.6 Cohérence spatiale : Aspects intrinsèques et extrinsèques

La cohérence spatiale est en effet en premier lieu une propriété extrinsèque à l'IP, liée aux conditions de son acquisition [41]. Elle est seulement en deuxième lieu une propriété intrinsèque aux données que constitue l'IP, conséquence des conditions d'acquisition (Figure 2.2).

Aspect intrinsèque :

Au strict niveau des données de l'image de profondeur, la propriété de cohérence spatiale signifie que le rangel qui est le plus proche d'un autre dans l'espace 3D (distance euclidienne) est un de ses voisins immédiats dans le tableau et que le rangel voisin d'un autre dans l'image de profondeur est parmi les 9 plus proches dans l'espace 3D.

Aspect extrinsèque :

A ce niveau pour satisfaire la propriété de cohérence spatiale, le faisceau de droites fixé par le système de mesure doit être quasi-parallèle l'une a l'autres et deux droites du faisceau qui sont voisines dans l'espace 3D doivent correspondre à des positions voisines dans le tableau et vice-versa.

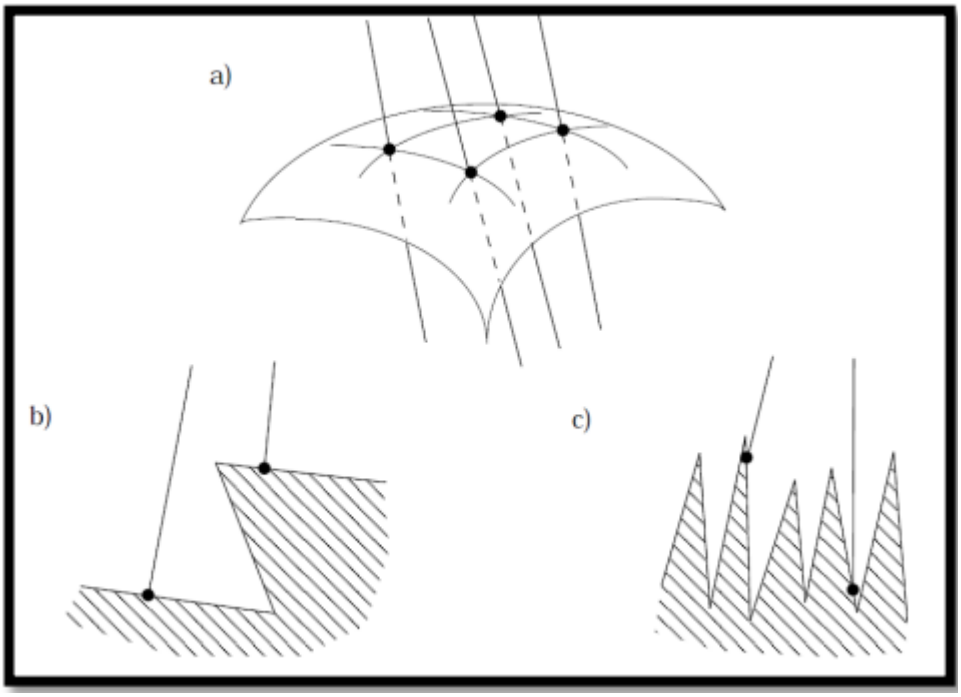


Figure 2.2 - Cohérence spatiale. (a) localement vrai, (b) et (c) localement faux.

2.3 Techniques d'acquisition des images de profondeur

2.3.1 Importance des méthodes d'acquisition

La qualité et la précision d'une image de profondeur dépendent directement de la technologie d'acquisition utilisée. Les méthodes se divisent en deux catégories : actives (nécessitant une source d'énergie externe) et passives (s'appuyant sur l'environnement lumineux existant).

2.3.2 Méthodes actives

2.3.2.1 Temps de vol (Time-of-Flight, ToF)

- **Principe** :
  Mesure du délai entre l'émission d'un signal lumineux (laser ou LED) et sa réception après réflexion sur une surface (Figure 2.3).

$$Z = \frac{c\ \Delta t}{2} \tag{2.1}$$

  Où c est la vitesse de la lumière et Δt le temps de parcours.

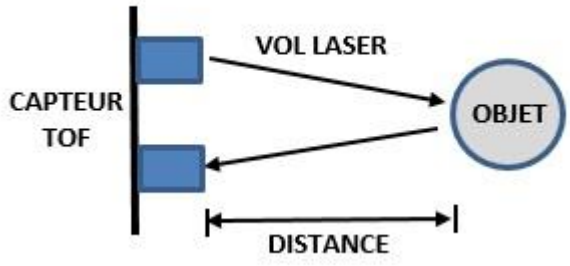


Figure 2.3 - Principe du temps de vol

- **Exemples** :
  - LiDAR : Utilisé en véhicules autonomes (ex : Velodyne HDL-64).
  - Capteurs ToF grand public : Microsoft Azure Kinect, Sony DepthSense.
- **Avantages** :
  - Précision millimétrique, fonctionnement en extérieur.
- **Limites** :
  - Coût élevé, résolution limitée, sensibilité aux réflexions multiples.

2.3.2.2 Triangulation laser

- **Principe** :

  Projection d'un motif infrarouge structuré (grille, points) et calcul du décalage entre le motif projeté et celui capturé par une caméra (Figure 2.4).

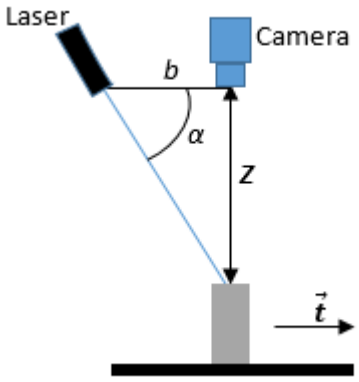


Figure 2.4 - Configuration générale d'un système de triangulation laser.

- **Exemples** :
  - Microsoft Kinect v1 : Projecteur infrarouge + caméra monochrome.
  - Structured Light 3D Scanners : Pour l'industrie (ex : Artec Eva).
- **Avantage** :
  - Coût modéré, adapté aux environnements intérieurs.
- **Limite** :
  - Sensible à la lumière ambiante et aux surfaces réfléchissantes.

2.3.3 Méthodes passives

2.3.3.1 Stéréovision

- **Principe** :

  Utilisation de deux caméras jumelées pour calculer la profondeur par corrélation de disparité (Figure 2.5).

$$Z = \frac{B\,f}{d} \tag{2.2}$$

Où f est la focale, B la base stéréo, et d la disparité.

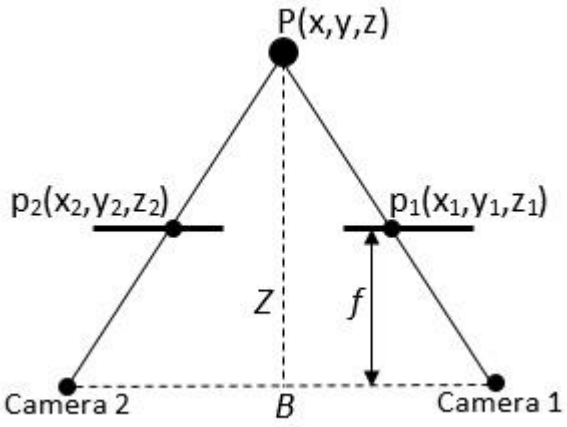


Figure 2.5 - Configuration de stéréovision ; deux caméras capturent un point de la scène

- **Exemples** :
  - Intel RealSense D415 : Capteur infrarouge passif.
  - Caméras binoculaires : ZED 2 de Stereolabs.
- **Avantage** :
  - Aucune émission lumineuse, adapté aux petits espaces.
- **Limites** :
  - Performances médiocres en faible lumière ou sur surfaces non texturées.

2.3.3.2 Shape from X

- **Principe** :

  Inférence de la profondeur à partir de caractéristiques visuelles :
  - Shape from Shading : Analyse des gradients d'ombres.
  - Shape from Motion : Utilisation de séquences vidéo pour reconstruire la 3D.
- **Avantage** :
  - Aucun matériel spécialisé requis.
- **Limites** :
  - Précision limitée, dépendante des conditions d'éclairage.

2.3.4 Comparaison des méthodes

Tableau 2.2 - Comparaison des méthodes d'acquisition.

Critère	**ToF**	**Triangulation**	**Stéréovision**	**Shape from X**
Précision	Élevée (mm)	Moyenne (cm)	Moyenne (cm)	Faible (dm)
Coût	Élevé	Modéré	Modéré	Faible
Environnement	Intérieur/Extérieur	Intérieur	Intérieur	Variable
Défis	Réflexions multiples	Lumière ambiante	Surfaces non texturées	Éclairage inconstant

2.3.5 Défis communs

- **Bruit** :
    - Artéfacts de mesure (ex : *flying pixels* dans les zones occultées).
- **Synchronisation** :
    - Alignement temporel et spatial entre capteurs RGB et profondeur (pour les RGBD).
- **Calibration** :
    - Correction des distorsions optiques (ex : modèle de Brown-Conrady).

2.4 Défis Techniques et Prétraitement

Cette section explore les principaux défis associés aux images de profondeur et les techniques de prétraitement pour améliorer leur qualité.

2.4.1 Sources de Bruit et Leurs Impacts

2.4.1.1 Bruit Aléatoire

- **Définition** : Variations imprévisibles des mesures de profondeur, souvent dues aux limitations matérielles.

- **Exemples** :
    - *Capteurs à temps de vol (ToF)* : Bruit thermique générant des fluctuations de ±1 cm à 10 m [14].
    - *Capteurs à triangulation (ex : Kinect)* : Erreurs de ±2 % à 2 m [4].
- **Impact** : Points aberrants (*outliers*) dans les nuages de points, réduisant la précision de la segmentation.

2.4.1.2 Artefacts Systématiques

- **Multi-trajets** :
    - Cause : Réflexions multiples sur des surfaces réfléchissantes (verre, métal).
    - Effet : Mesures de profondeur erronées ou fantômes.
- **Interférences Lumineuses** :
    - Cause : Lumière ambiante intense perturbant les capteurs à lumière structurée.
    - Solution : Filtrage spectral ou utilisation de lasers modulés [15].
- **Exemple** :

  La figure 2.6 montre des exemples de bruits et erreurs systématiques.

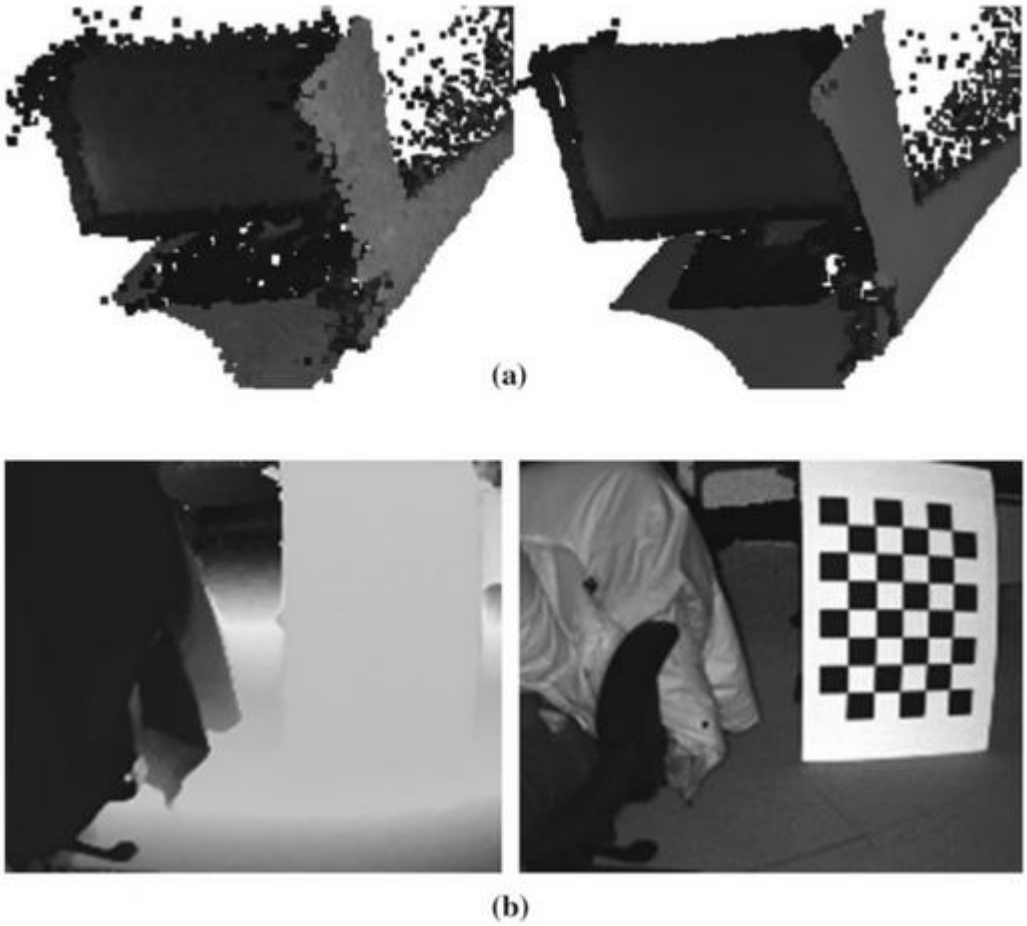


Figure 2.6 - Exemple de bruit et erreur systématiques (Source : [15]).

Ces erreurs proviennent du principe fondamental de la mesure de profondeur : **(a) Integration time error** : Un temps d'intégration plus long (à droite) montre une précision de profondeur plus élevée qu'un temps d'intégration court (à gauche). **(b) IR amplitude error** : Des points 3D à la même profondeur (damier à gauche) présentent des amplitudes IR différentes (damier à droite) selon la couleur de l'objet cible.

2.4.2 Déformations Géométriques

2.4.2.1 Distorsions de Lentille

- **Types** :
    - *Radiales* : Déformation en « barillet » ou « coussinet » (caméras grand-angle).
    - *Tangentielles* : Désalignement des lentilles.

- **Correction** :
    - Modèle de calibration de Brown-Conrady [16] :

$$x_{corrigé} = x\,.(1 + k_1 r^2 + k_2 r^4) + 2\,p_1 xy + p_2\,(r^2 + 2x^2) \quad (2.3)$$

Où $r = \sqrt{x^2 + y^2}$; k_1, k_2 : Coefficients de distorsion radiale ; p_1, p_2 : Coefficients tangentiels.

2.4.2.2 Occultations

- **Cause** : Objets bloquant partiellement la vue du capteur.
- **Solutions** :
    - Acquisition multi-vues : Fusion de scans depuis différents angles [9].
    - Complétion par IA : Algorithmes comme *NLSPN* pour prédire les zones manquantes [17].

2.4.3 Techniques de Prétraitement

2.4.3.1 Filtrage du Bruit

1. **Filtre Médian** :
    - Remplace chaque pixel par la médiane de son voisinage.
    - Avantage : Supprime efficacement le bruit *salt-and-pepper*.
    - Limite : Floute les contours fins.
2. **Filtre Bilatéral** :
    - Combine un filtre spatial et un filtre d'intensité :

$$I_{filtré}(x) = \frac{1}{W} \sum_{y \in \Omega} I(y)\,.\; e^{-\frac{\|x-y\|^2}{2\sigma_s^2}}\;.\; e^{-\frac{(I(x)-I(y))^2}{2\sigma_r^2}} \quad (2.4)$$

Où σ_s contrôle le voisinage spatial, σ_r le voisinage en intensité.

- Avantage : Préserve les contours tout en lissant les zones homogènes.

2.4.3.2 Complétion de Profondeur

- **Méthodes Classiques** :
    - *Interpolation linéaire* : Simple mais inefficace sur les bords.

- *Inpainting géométrique* : Utilise la cohérence spatiale pour remplir les trous [18].

- **Méthodes Modernes** :
  - NLSPN (*Non-Local Spatial Propagation Network*) :
    - Utilise un réseau de neurones pour propager les informations de profondeur [17].
    - Performance : Réduction de l'erreur absolue moyenne (MAE) sur NYU Depth V2.

Les défis techniques (bruit, distorsions, occultations) et les méthodes de prétraitement (filtrage, complétion, calibration) sont critiques pour la qualité des données. Un prétraitement adapté permet d'améliorer significativement les performances des algorithmes de segmentation.

2.5 Segmentation d'images de profondeur

La segmentation est une importante étape pour la compréhension et la perception d'images de profondeur. Le problème principal de la segmentation d'image de profondeur réside dans le fait de découper l'image en surfaces disjointes, permettant la représentation individuelle de chaque objet figurant dans l'image, du fait de la structure en grille car l'image de profondeur est une image pleine de bruit liée aux méthodes d'acquisition. La segmentation d'images de profondeur se rapproche des techniques désormais classiques de segmentation d'images 2D. Les auteurs distinguent en général deux classes de méthodes, les approches par contours (edge-based) et les approches par régions (region-based) et éventuellement les méthodes hybrides utilisant les deux approches précédentes.

Cette section détaille les trois principales familles de méthodes de segmentation appliquées aux images de profondeur : basées sur les contours, basées sur les régions, et par classification. Chaque approche est analysée en fonction de ses principes, algorithmes, avantages, limites et applications pratiques.

2.5.1 Méthodes basées sur les contours

Principe : Identifier les discontinuités géométriques (changements brusques de profondeur ou d'orientation de surface).

Algorithmes :

1. **Détection de gradients** :
    - **Sobel** :

      Calcul des gradients horizontal G_x et vertical $G_y$via les masques de convolution :

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}, \qquad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix}$$

      La magnitude du gradient $\sqrt{G_x^2 + G_y^2}$, est utilisée pour repérer les contours.

    - **Canny ;** :

      Chaînage des contours par seuillage hystérésis et suppression des non-maxima locaux. Adapté aux images de profondeur via des filtres adaptatifs [19].

2. **Détection de coins** :

   Les coins sont des points dans une image où les variations d'intensité sont significatives dans toutes les directions. Ils sont cruciaux pour des tâches comme l'appariement de caractéristiques, la reconstruction 3D ou le suivi d'objets. Deux méthodes classiques pour les détecter sont *Harris* [20] et *Shi-Tomasi* [21].

    - *Harris* : Mesure de la variation d'intensité locale via la matrice de structure.
    - *Shi-Tomasi* : Amélioration pour une meilleure détection des coins faibles en utilisant la plus petite valeur propre.

Avantages :

- Rapidité : Adapté aux applications en temps réel (ex : robotique mobile [22]).
- Simplicité : Faible besoin en ressources calculatoires.

Limites :

- Sensibilité au bruit : Requiert un prétraitement (filtrage médian/gaussien) pour atténuer les artefacts.

- Contours incomplets : Nécessite un post-traitement (fermeture de contours via morphologie mathématique).

Exemple d'application :
Détection des bords d'un meuble dans une scène intérieure, utilisée en robotique pour l'évitement d'obstacles.

2.5.2 Méthodes basées sur les régions

Principe : Regrouper les pixels homogènes selon un critère géométrique (distance, orientation).

Algorithmes :

1. **Split and Merge** :
    - *Split* : Division récursive de l'image en sous-régions carrées jusqu'à homogénéité (variance de profondeur < seuil).
    - *Merge* : Fusion des régions adjacentes si leur variance combinée reste sous le seuil.
2. **Croissance de régions** :
    - Expansion à partir de germes initiaux (ex : pixels de profondeur minimale).
    - Critère d'arrêt : Variance locale ou nombre maximal d'itérations [23].

Avantages :

- Robustesse au bruit : Moins affecté par les artefacts ponctuels grâce aux critères statistiques.
- Segmentation cohérente : Production de régions connexes, idéales pour la modélisation 3D.

Limites :

- Complexité : Temps de calcul élevé pour les images haute résolution (ex : nuages de points LiDAR).
- Sur-segmentation : Division excessive en petites régions nécessitant une fusion manuelle.

Exemple d'application :

Segmentation d'un sol plat en intérieur pour la navigation robotique, permettant une cartographie précise.

2.5.3 Méthodes par classification

Principe : Catégoriser les pixels en classes prédéfinies (ex : sol, mur, objet) via des modèles statistiques ou d'apprentissage.

Approches :

1. **Machine Learning classique** :
    - *SVM* : Classification basée sur des caractéristiques extraites (profondeur moyenne, gradient, histogrammes) [24].
    - *Random Forests* : Arbres de décision entraînés sur des descripteurs géométriques locaux [25].
2. **Apprentissage profond** :
    - *U-Net 3D* : Extension de l'architecture U-Net pour traiter des volumes 3D, intégrant des couches de *pooling* 3D [26].
    - *Mask R-CNN RGB-D* : Segmentation d'instances avec entrée multi-modale (RGB + profondeur) et masques de profondeur [27].
    - *PointNet++* : Traitement direct des nuages de points via des caractéristiques hiérarchiques [28].

Avantages :

- Précision élevée : Performances supérieures sur des scènes complexes (ex : scènes domestiques encombrées).
- Adaptabilité : Apprentissage de caractéristiques hiérarchiques sans extraction manuelle.

Limites :

- Besoin de données annotées : Exige des jeux de données volumineux (ex : *NYU Depth V2*, *ScanNet*).
- Complexité calculatoire : Entraînement coûteux en ressources (GPU haute performance).

Exemple d'application :

Segmentation sémantique d'une cuisine pour un robot domestique, identifiant plans de travail, électroménagers et obstacles.

2.5.4 Comparaison des méthodes

Tableau 2.3 - Comparaison des méthodes de segmentation.

Critère	**Contours**	**Régions**	**Classification**
Vitesse	Rapide	Moyenne	Lente (entraînement)
Robustesse au bruit	Faible	Élevée	Élevée (avec filtrage)
Précision	Limité (contours)	Bonne (surfaces)	Excellente (sémantique)
Cas d'usage	Détection d'objets	Navigation robotique	Scènes complexes

2.5.5 Revue des travaux de la littérature

La segmentation des images de profondeur dépend fortement des objets pouvant apparaître dans les images. Selon ce constat, les méthodes de segmentation d'images de profondeur peuvent être divisées en quatre familles, selon le critère d'homogénéité utilisé. Il s'agit des méthodes : basées sur des surfaces planes, basées sur des courbures, basées sur des surfaces algébriques et basées sur la continuité C1.

Dans les méthodes basées sur des surfaces planes, des critères géométriques adaptés sont utilisés, tels que l'équation du plan et son orientation. Ces critères sont ensuite exploités pour extraire des régions ou éventuellement détecter des contours, selon la méthode adoptée. Il est également nécessaire de distinguer des surfaces ayant la même orientation mais appartenant à des plans parallèles différents. La méthode de Parvin [29] appartient à cette famille, où l'auteur a utilisé une technique de *division-fusion* avec le vecteur normal de la surface comme caractéristique de l'image. L'homogénéité repose sur les angles entre les vecteurs normaux et est utilisée pour la fusion des régions en post-traitement. Une technique de division-fusion basée sur le gradient a également été employée dans les travaux de Taylor [30] et Bhavsar [31].

Dans les méthodes basées sur des courbures, les auteurs utilisent des techniques de seuillage sur les courbures moyenne et gaussienne. Cependant, l'estimation des courbures pose problème dans cette catégorie de méthodes. En effet, le bruit dû aux erreurs de mesure et aux discontinuités de profondeur ne permet pas une bonne estimation. Ainsi, pour obtenir une estimation correcte des courbures, il est nécessaire de traiter le bruit et d'éliminer les perturbations liées aux discontinuités via un traitement approprié. Besl et Jain [32] ont utilisé cette méthode pour définir des graines initiales pour la croissance de régions. La détection des discontinuités pour une segmentation basée sur les courbures a été proposée par Yokoya et Levine [33]. Dans un autre travail, Kasvand [34] a prétraité les pixels appartenant à un voisinage local afin d'atténuer les effets des discontinuités.

La segmentation en surfaces algébriques, qui ne sont pas strictement planes, concerne deux catégories : les surfaces 2.5D et 3D. Les surfaces 2.5D, correspondant à des fonctions polynomiales à deux variables, ne s'appliquent qu'aux images scalaires. Pour les surfaces 3D, qui correspondent à des quadriques ou des superquadriques, leur traitement est plus complexe que celui des surfaces 2.5D. Gupta et Bajcsy [35,36] ont présenté une méthode de segmentation décrivant l'image de profondeur par des superquadriques, comme des ellipsoïdes. Jiang et Bunke [37] ont utilisé une méthode de croissance de régions sous contraintes 2.5D d'approximation plane. Cette croissance particulière repose sur des segments de droite, où les régions sont formées de listes de segments. Ainsi, la croissance d'une région s'effectue segment par segment, les segments linéaires étant délimités par division de profil (par colonne ou par ligne) de l'image traitée.

Dans les méthodes basées sur la continuité C1, la segmentation repose sur un critère défini comme l'homogénéité d'une surface, appelée continuité C1. Deux principes se distinguent :

1. Fusion de segments issus d'une segmentation plus contrainte [32].
2. Croissance des points de frontière détectés pour former des frontières fermées [38].

Récemment, les réseaux neuronaux profonds ont été largement utilisés pour la classification d'images, la détection et la reconnaissance d'objets. Cependant, dans plusieurs cas, un post-traitement basé sur les contours extraits est nécessaire pour finaliser la reconnaissance, comme dans le travail de Y. Wong et al. [39]. Dans cette étude, après avoir reconnu les *Racing Bib Numbers* (RBNs) via YOLOv3, les auteurs ont appliqué une suppression des *non-maxima* pour prédire une seule boîte englobante par objet cible.

Après avoir examiné les différentes méthodes, nous pouvons conclure que la plupart s'appuient sur des implémentations antérieures utilisées pour des images en couleur ou en niveaux de gris.

Les méthodes de segmentation ont évolué d'une approche géométrique rigide vers des modèles *data-driven* flexibles. Cependant, les travaux historiques comme ceux de Gupta & Bajcsy restent pertinents, soit comme base théorique, soit via leur intégration dans des architectures hybrides. Les défis actuels appellent à une synergie entre interprétabilité géométrique et puissance de l'IA.

2.6 Post-traitement et régularisation

Cette section explore les techniques utilisées pour améliorer les résultats de segmentation en corrigeant les artefacts, en lissant les contours et en garantissant une cohérence spatiale. Ces étapes sont importantes pour transformer une segmentation brute en une interprétation exploitable dans des applications pratiques (ex : robotique, imagerie médicale).

2.6.1 Nécessité du post-traitement

Les algorithmes de segmentation (par contours, régions ou apprentissage) génèrent souvent des résultats imparfaits en raison :

- Du bruit résiduel : Pixels mal classés ou artefacts de mesure.
- Des contours fragmentés : Discontinuités non fermées.
- De la sur-segmentation : Régions trop fines ou non pertinentes.

Exemple :
Une segmentation par division-fusion peut produire des centaines de petites régions sur un mur texturé, nécessitant une fusion ultérieure.

2.6.2 Techniques de régularisation

2.6.2.1 Champs de Markov (MRF)

- **Principe** :
  Modélisation des dépendances spatiales entre pixels voisins via une fonction d'énergie :

$$E(X) = \sum_{i} \phi(x_i) + \lambda \sum_{(i,j)\in N} \psi(x_i, x_j) \qquad (2.5)$$

Où :

- $\phi(x_i)$: Coût de données (adéquation aux observations).
- $\psi(x_i, x_j)$: Pénalise les différences entre pixels voisins.
- λ : Poids de la régularisation.

- **Application** :
  Lissage des régions segmentées dans des images LiDAR (ex : travaux de Shotton et al., [40]).

2.6.2.2 Régularisation bayésienne

- **Principe** :
  Intégration de connaissances a priori (ex : forme des objets) via le théorème de Bayes :

$$P(\theta|X) \propto P(X|\theta) \,.\, P(\theta) \tag{2.6}$$

  Où :

  - $P(\theta)$: Distribution a priori (ex : régularité géométrique).
  - $P(X|\theta)$: Vraisemblance des données.

- **Mise en œuvre** :
  - Réaffectation des pixels ambigus via l'algorithme EM.
  - Exemple : Correction des contours d'un organe dans un scan 3D médical.

2.6.2.3 Méthodes variationnelles

- **Principe** :
  Minimisation d'une fonctionnelle d'énergie combinant fidélité aux données et contraintes de lissage :

$$E(u) = \int_{\Omega} ((Z-u)^2 + \alpha\, |\nabla u|^2) dx\, dy \tag{2.7}$$

  Où α contrôle le lissage.

- **Application** :
  Unification de régions fragmentées dans des images de profondeur Kinect.

2.6.3 Techniques de post-traitement spécifiques

2.6.3.1 Fermeture de contours

- **Algorithme** :
  - Détection des contours brisés via analyse de connectivité.
  - Connexion des extrémités avec des lignes droites ou des courbes spline.
- **Outils** :
  - Morphologie mathématique (dilatation/érosion).
  - Algorithmes de comblement de lacunes.

2.6.3.2 Suppression des petites régions

- **Méthode** :
  - Calcul de la taille des régions (en pixels ou en volume 3D).
  - Fusion ou suppression des régions sous un seuil critique.
- **Exemple** :
  Élimination des feuilles volantes dans un nuage de points LiDAR.

2.6.3.3 Lissage des surfaces

- **Algorithmes** :
  - Lissage Laplacien : Ajustement itératif des positions des points pour réduire le bruit.
  - MLS (*Moving Least Squares*) : Approximation locale des surfaces par des polynômes.

2.6.4 Comparaison des méthodes

Tableau 2.4 - Comparaison des méthodes de post-traitement.

Méthode	**Avantages**	**Limites**	**Complexité**
MRF	Cohérence spatiale garantie	Calcul coûteux	Élevée
Bayésienne	Intègre des connaissances a priori	Dépend de la qualité du modèle	Moyenne
Variationnelle	Lissage efficace	Perte de détails fins	Faible
Fermeture de contours	Rapide et simple	Peut créer des artefacts	Faible

2.6.5 Applications pratiques

- **Robotique mobile** :
  Lissage des cartes de profondeur pour éviter les collisions (ex : drones autonomes).
- **Imagerie médicale** :
  Régularisation des contours de tumeurs dans des IRM 3D.
- **Reconstruction 3D** :
  Suppression du bruit dans les scans architecturaux (ex : monuments historiques).

Le post-traitement et la régularisation transforment une segmentation rudimentaire en une représentation exploitable, en harmonisant précision et robustesse. Alors que les méthodes classiques (MRF, bayésiennes) restent pertinentes, l'intégration de l'apprentissage profond ouvre la voie à des approches adaptatives capables de généraliser à des environnements dynamiques et complexes.

2.7 Conclusion

Ce chapitre a exploré la segmentation des images de profondeur, une étape essentielle pour interpréter la géométrie 3D d’une scène. Nous avons vu que ces images, qui encodent des distances plutôt que des couleurs, posent des défis uniques : bruit des capteurs, discontinuités géométriques et complexité des objets.

Les méthodes classiques (contours, régions) offrent des solutions rapides mais sensibles aux imperfections des données. Les approches modernes, comme l’apprentissage profond (*U-Net*, *Mask R-CNN*), permettent une segmentation précise des scènes complexes, au prix d’un besoin important en données annotées.

Malgré les progrès, des défis persistent :

- Robustesse face aux capteurs variés (Kinect, LiDAR).
- Temps réel pour des applications embarquées (robotique, réalité augmentée).

Les perspectives incluent l’hybridation de méthodes géométriques et de réseaux de neurones, ainsi que l’optimisation pour des usages pratiques (imagerie médicale, véhicules autonomes).

Chapitre 3

Régularisation en segmentation d'images

3.1 Introduction

La segmentation d'images est considérée comme un problème mal posé, dont la solution est fortement indéterminée ou fortement mal conditionnée. Les approches classiques pour calculer une solution d'un problème mal posé nécessitent des informations supplémentaires pour renforcer l'unicité et la stabilité. A cette fin, les méthodes de régularisation sont largement utilisées. Dans ce cas, la solution est généralement obtenu en minimisant une fonctionnelle énergétique *E* contenant un terme de fidélité $\mathcal{F}$ qui mesure la cohérence de la segmentation candidate avec l'image observée, et un terme de régularisation $\mathcal{R}$ qui favorise les solutions avec des propriétés appropriées :

$$(I^*, u^*) \coloneqq \arg\min_{(I,u)} E(I, u\,;\, I_0) = \arg\min_{(I,u)} (\mathcal{F}(I, u\,;\, I_0) + \lambda\, \mathrm{R}(I, u)) \tag{3.1}$$

Où $\lambda > 0$ est un paramètre de régularisation qui nécessite généralement un réglage minutieux pour équilibrer convenablement $\mathcal{F}$ et $\mathcal{R}$ (voir, par exemple, [42] et les références qu'il contient).

Le problème de minimisation (3.1) peut être résolu par les équations d'Euler-Lagrange, qui peuvent être dérivées en intégrant par parties la fonctionnelle de l'énergie et en utilisant le théorème de Gauss avec le lemme fondamental du calcul des variations. Ensuite, une solution numérique peut être calculée en appliquant une approche de descente de gradient, où la direction de descente est paramétrée par un temps artificiel, et une discrétisation par différences finies. Une alternative largement utilisée et efficace consiste à discrétiser le problème (3.1) puis à le résoudre par une méthode d'optimisation numérique.

Récemment, des techniques d'apprentissage automatique ont été appliquées avec succès à des problèmes de segmentation. L'idée maîtresse consiste à ajuster un modèle générique à une solution spécifique en apprenant par rapport à des exemples de données (données d'apprentissage). La phase d'apprentissage extrait les informations préalables à intégrer

dans le terme de régularisation à partir d'un grand ensemble de données contenant des paires de type (image, ground-truth label) [43]. Des approches d'apprentissage automatique utilisant uniquement des données d'image comme ensemble de données d'apprentissage sont également utilisées.

Dans la suite du chapitre, nous allons présenter les différentes méthodes et modèles de régularisation utilisés dans la segmentation d'images.

3.2 Modèles régularisables de segmentation

3.2.1 En détection de contours

Les modèles basés contours visent à trouver $u^* = \cup_i \partial\ \Omega_i$ en résolvant le problème de minimisation par rapport à la courbe u (dans ce cas, I et I^* ne sont pas explicitement pris en compte). Parmi ces modèles, les Contours actifs [44], ou Snakes, sont très commun. Ici, les termes de fidélité et de régularisation agissent respectivement comme une force interne et une force externe, qui déplacent la courbe dans l'image pour trouver les bornes des ensembles Ω_i. Plus précisément, la fonctionnelle énergétique prend la forme :

$$E_{AC}(u) = \underbrace{\int_0^1 g\ (|\nabla\ I_0(u(s))|)^2 ds}_{\mathcal{F}} + \ \lambda\ \underbrace{\int_0^1 |u'(s)|^2\ ds}_{\mathcal{R}} \tag{3.2}$$

Où I_0 est l'image observée, g est une fonction de détection de contours et la courbe u est paramétrée par $s \in [0, 1]$. Le premier terme attire la courbe vers les contours, tandis que le second contrôle sa douceur (Smoothness), et par conséquent la courbe u change de forme comme un serpent.

La courbe évolutive est pilotée par les propriétés de surface, telles que la courbure et la direction normale, et par les caractéristiques de l'image, telles que les niveaux de gris, la teinte ou la saturation dans les images couleur, et l'intensité du gradient dans les images 2D ou le changement de pente dans les images 3D. Par exemple, la courbure moyenne peut être utilisée et dans ce cas la fonction de détection de contours est également responsable de l'arrêt de la courbe sur les contours. La fonction g peut être définie comme suit :

$$g(|\nabla I_0| = \frac{1}{1 + \ |\nabla(G_\sigma * \ I_0)|^2} \tag{3.3}$$

Où g est une fonction positive et décroissante, G_σ est le noyau gaussien d'écart type σ, et $*$ désigne l'opérateur de convolution.

Dans une approche lagrangienne, une courbe initiale est développée par

$$\frac{\partial u}{\partial t} + \mathcal{L}(u) = 0 \tag{3.4}$$

Où $\mathcal{L}$ est un opérateur différentiel. L'évolution la plus simple est donnée par

$\mathcal{L}(u) = F*N$, où N est la normale à la courbe et F est une constante qui détermine la vitesse d'évolution. Plus généralement, l'évolution est portée par une force extérieure. Par exemple, dans l'évolution de la courbure moyenne, $\mathcal{L}(u) = \kappa N$, où κ est la courbure euclidienne de u [45].

Lorsque u a une représentation explicite, il n'est pas facile de gérer les changements topologiques comme la fusion et la division, et un nouveau paramétrage de la courbe peut être nécessaire. Par conséquent, l'évolution de la courbe u est communément décrite par les méthodes level-set [46], grâce à leur capacité à suivre les changements de topologie, les points de rebroussement et les coins. Dans une approche level set, la courbe u est implicitement représentée par l'ensemble de niveau zero-level d'une fonction $\phi(t, x)$, c'est-à-dire $u = \{ x \in \Omega : \phi(t, x) = 0\}$.

Les formulations level set de l'évolution la plus simple et de la courbure moyenne se lisent respectivement :

$$\frac{\partial \phi}{\partial t} = F\left| \nabla_\phi \right| \ et \ \frac{\partial \phi}{\partial t} = \text{div}\left(\frac{\nabla_\phi}{|\nabla_\phi|}\right)|\nabla_\phi| \tag{3.5}$$

3.2.2 En extraction de régions

Les modèles basés régions fournissent directement la segmentation au moyen de la partition d'image $\{\Omega_i,\ i = 1, \ldots, m\}$.

Les modèles de croissance de région sont parmi les modèles les plus simples appartenant à cette classe [47]. Étant donné que les critères d'agrégation basés uniquement sur les mesures en niveau de gris peuvent ne pas être suffisantes pour obtenir des segmentations précises, les méthodes de croissance de région ont été fusionnées avec des approches variationnelles où l'évolution évolue selon la minimisation d'une fonctionnelle de l'énergie, y compris les termes basés région [48].

L'un des modèles les plus populaires a été proposé par Mumford et Shah [49]. Dans ce cas, la fonctionnelle *E* dans (3.2) prend la forme :

$$E_{MS}(I,u) = \underbrace{\int_{\Omega} (I - I_0)^2 \, dx}_{\mathcal{F}} + \underbrace{\lambda \int_{\Omega - u} |\nabla I|^2 dx + \mu \, len(u)}_{\mathcal{R}} \tag{3.6}$$

Où $len(u)$ désigne la longueur de u et $\lambda, \mu > 0$ sont des paramètres de régularisation. Le terme de fidélité tente d'atteindre la distance minimale entre I_0 et l'approximation optimale lisse par morceaux I, et le terme de régularisation tente de réduire la variation de I dans chaque ensemble Ω_i tout en gardant la courbe u aussi courte que possible.

La minimisation (3.6) dans un espace convenable fournit un couple optimal

(I^*, u^*) représentant une description simplifiée de I_0 au moyen d'une fonction à variation bornée et d'un ensemble de contours [50]. Enfin, dans [49] le modèle de Mumford et Shah est formulé comme un raffinement déterministe d'un modèle probabiliste pour la restauration d'images.

Une version simplifiée du modèle de Mumford-Shah est sa restriction aux fonctions constantes par morceaux. Le modèle de Chan-Vese [51] est un cas particulier de cette version simplifiée, visant à obtenir une segmentation en deux phases où la fonction constante par morceaux ne prend que deux valeurs. Sa fonctionnelle E prend la forme suivante :

$$E_{CV}(I, C_{in}, C_{out}) = \underbrace{(\int_{\Omega} H(I)\,(C_{in} - I_0)^2 dx + \int_{\Omega} (1 - H(I))(C_{out} - I_0)^2 dx)}_{\mathcal{F}} + \lambda \underbrace{\int_{\Omega} |\nabla H(I)| dx}_{\mathcal{R}} \tag{3.7}$$

Où H est la fonction de Heaviside et C_{in} et C_{out} sont les valeurs moyennes de l'intensité au premier plan et arrière-plan de l'image, respectivement. La solution I^* est la meilleure approximation de I_0 parmi toutes les fonctions qui ne prend que deux valeurs.

La minimisation (3.7) est un problème non convexe, donc les méthodes de résolution peuvent rester bloquées dans les minima locaux et entraîner des segmentations insatisfaisantes. Afin de pallier cet inconvénient, certaines stratégies ont été proposées, notamment la convexification de la fonctionnelle en tirant parti de ses propriétés

géométriques. Un exemple est donné par le modèle de partitionnement en deux phases introduit par Chan, Esedoglu et Nikolova [52] :

$$E_{CEN}(I, C_{in}, C_{out}) = \underbrace{\int_{\Omega}((C_{in} - I_0)^2\, I + \ (C_{out} - I_0)^2(1 - I)\, dx}_{\mathcal{F}} + \lambda \underbrace{\int_{\Omega} |\nabla I| dx}_{\mathcal{R}} \qquad (3.8)$$

Avec $0 \leq I \leq 1$ et C_{in} , $C_{out} > 0$.

3.2.3 Modèles statistiques

Les modèles statistiques fournissent généralement une probabilité conditionnelle,

$P(S|I_0)$, d'une segmentation $S \in \Sigma$ compte tenu de l'image I_0, puis sélectionnent la segmentation avec la probabilité la plus élevée. La segmentation par l'approche du Maximum a Posteriori (MAP) est donnée par :

$$S^* = \arg\max_{S \in \Sigma} P(S|I_0) \qquad (3.9)$$

Où la probabilité a posteriori $P(S|I_0)$ peut être exprimée par le théorème de Bayes comme :

$$P(S|I_0) \ \propto P(I_0|S)\, P(S) \qquad (3.10)$$

$P(S)$ Est la probabilité a priori mesurant à quel point S satisfait certaines propriétés de l'image donnée, et la probabilité conditionnelle $P(I_0|S)$ mesure la probabilité de I_0 étant donné S.

Les modèles de champ aléatoire de Markov (MRF) offrent un cadre pour définir l'a priori et la vraisemblance en capturant les propriétés de l'image comme la texture, la couleur, etc. [53]. La segmentation est formulée dans un cadre d'étiquetage d'image, c'est-à-dire, $S = \phi\,(I(x))$, où le problème est réduit pour trouver l'étiquetage qui maximise la probabilité a posteriori. Les dépendances d'étiquettes sont modélisées par un MRF ; puis, en utilisant le théorème de Hammersley-Clifford, on obtient la distribution de Gibbs :

$$P(S) = \frac{1}{Z}\, exp^{(-U(S))} \qquad (3.11)$$

Où la fonction d'énergie *U* prend la forme :

$$U(S) = \sum_{c \in C} V_c(S_c) \tag{3.12}$$

C'est l'ensemble des cliques de S, $V_c(S_c)$ est le potentiel de la clique $c \in C$ ayant la configuration d'étiquette S_c et Z est une constante de normalisation. La probabilité conditionnelle $P(I_0|S)$ peut être modélisée par une distribution gaussienne. Ensuite, l'estimation MAP d'origine est équivalente au problème de minimisation d'énergie suivant :

$$S^* = \arg\max_S P(S|I_0) = \arg\min_S U(S) \tag{3.13}$$

3.2.4 Modèles basés apprentissage automatique

Récemment, des approches d'apprentissage automatique, et en particulier d'apprentissage profond (deep learning) , sont utilisées pour résoudre des problèmes de segmentation d'images, surpassant les approches précédentes. D'une manière générale, les approches d'apprentissage automatique ne bénéficient pas d'informations préalables sur la solution, mais « apprennent » la segmentation à partir de grands ensembles de données d'apprentissage. Plus en détail, le but d'une approche d'apprentissage automatique est de définir un modèle de segmentation

$f_\theta : \mathcal{I} \to \Sigma$ de sorte que la segmentation de I_0 peut être obtenue par $I^*=f_\theta(I_0)$. La fonction f_θ est généralement non linéaire et θ est un grand vecteur de paramètres. La phase d'apprentissage sélectionne θ afin de minimiser une fonctionnelle de perte (ou de coût) qui mesure la précision de la segmentation prédite $f_\theta(I_0)$.

En apprentissage automatique supervisé, les données d'apprentissage sont disponibles à partir de bases de données de segmentations annotées, qui fournissent un grand nombre de paires ($I_0, I^*) \in X \times Y \subset \mathcal{I} \times \Sigma$ ($X \times Y$ est nommé ensemble d'apprentissage). Ainsi, θ est obtenu par minimiser une fonction de perte qui prend souvent la forme d'une erreur quadratique moyenne plus un terme de régularisation :

$$\theta^* = \arg\min_\theta \mathcal{L}(X, Y, \theta) = \arg\min_\theta \sum \| f_\theta(I_0) - I^* \|^2 + \mathcal{R}_\theta\ (f_\theta(I_0)) \quad (\tag{3.14}$$

Dans l'apprentissage automatique non supervisé, l'ensemble d'apprentissage n'est pas disponible et l'objectif est d'entraîner f_θ à reconnaître des modèles spécifiques ou des caractéristiques d'image dans les données. Cette approche est parfois appelée apprentissage auto-supervisé [54], parce que l'information est extraite des données elles-mêmes plutôt

que d'un ensemble de « prédictions » (c'est-à-dire segmentations). Alors le terme de fidélité dans (3.14) prend la forme :

$$\sum_{\mathfrak{T}} \| f_0(I_0) - \phi(f_0(I_0)) \|^2 \tag{3.15}$$

Où ϕ est l'opérateur d'étiquetage.

Afin d'extraire progressivement des caractéristiques de plus haut niveau des données, les modèles d'apprentissage automatique utilisent une structure multicouche appelée réseau de neurones, constituée de compositions de fonctions successives. Le nombre de couches est la profondeur du modèle, d'où la terminologie d'apprentissage en profondeur. Un réseau de neurones à L couches est une fonction :

$$f_\theta : \mathcal{I} \text{ x } (H_1 \, x \, \dots x \, H_L) \to \Sigma, \quad f_0(I) = (f_L \circ f_{L-1} \circ \dots \circ f_L)(I) \tag{3.16}$$

Où f_i : $\mathrm{R}^{d_{i-1}} \times H_i \to \mathrm{R}^{d_i}$: $\mathrm{R}^{d_{i-1}} \times H_i \to \mathrm{R}^{d_i}$ sont les fonctions de couche, chacune dépendant d'une composante θ_i de θ, $d_0 = d$, et $d_L = n$ avec n égal au nombre de caractéristiques. Parmi les architectures de réseaux de neurones profonds bien connues utilisées avec succès dans la segmentation d'images, nous mentionnons AlexNet [55], GoogLeNet [56], VGGNet [57] et Fully CNN [58].

3.3 Modèles de segmentation markoviens

3.3.1 Champ de Markov aléatoire

Au début du 20ème siècle, largement inspiré du modèle d'Ising [59], un nouveau type de processus stochastique est apparu dans la théorie des probabilités, appelé champ de Markov aléatoire (MRF). Les MRF sont rapidement devenus un outil largement utilisé dans une variété de problèmes, pas seulement en mécanique statistique. L'utilisation des MRF dans le traitement d'images est devenue populaire avec l'article fondateur de S. Geman et D. Geman [50] en 1984, mais sa première utilisation dans le domaine date du début des années 70 [60, 61]. Il est utilisé dans l'étiquetage visuel pour établir des distributions probabilistes d'étiquettes en interaction.

Cette section présente les notations et les résultats relatifs à la vision.

3.3.1.1 Système de voisinage et cliques

Les sites de S sont liés les uns aux autres via un système de voisinage. Un système de voisinage pour S est défini comme :

$$\mathcal{N} = \{\mathcal{N}_i | \forall i \in S\} \tag{3.17}$$

Où $\mathcal{N}_i$ est l'ensemble des sites voisins de i. La relation de voisinage a les propriétés suivantes :

(1) Un site n'est pas voisin de lui-même : $i \notin \mathcal{N}_i$;

(2) La relation de voisinage est mutuelle : $i \in \mathcal{N}_{i'} \Leftrightarrow i' \in \mathcal{N}_i$.

Pour un treillis régulier S, l'ensemble des voisins de *i* est défini comme l'ensemble des sites dans un rayon $\sqrt{r}$ de i.

$$\mathcal{N}_i = \{i' \in S \mid [dist(pixel_i ; pixel_j)]^2 \leq r \; ; \; i' \neq i\} \tag{3.18}$$

Où $dist(A ; B)$ désigne la distance euclidienne entre $A \; et \; B \; et \; r$ prend une valeur entière. Notez que les sites aux limites ou à proximité des bords ont moins de voisins.

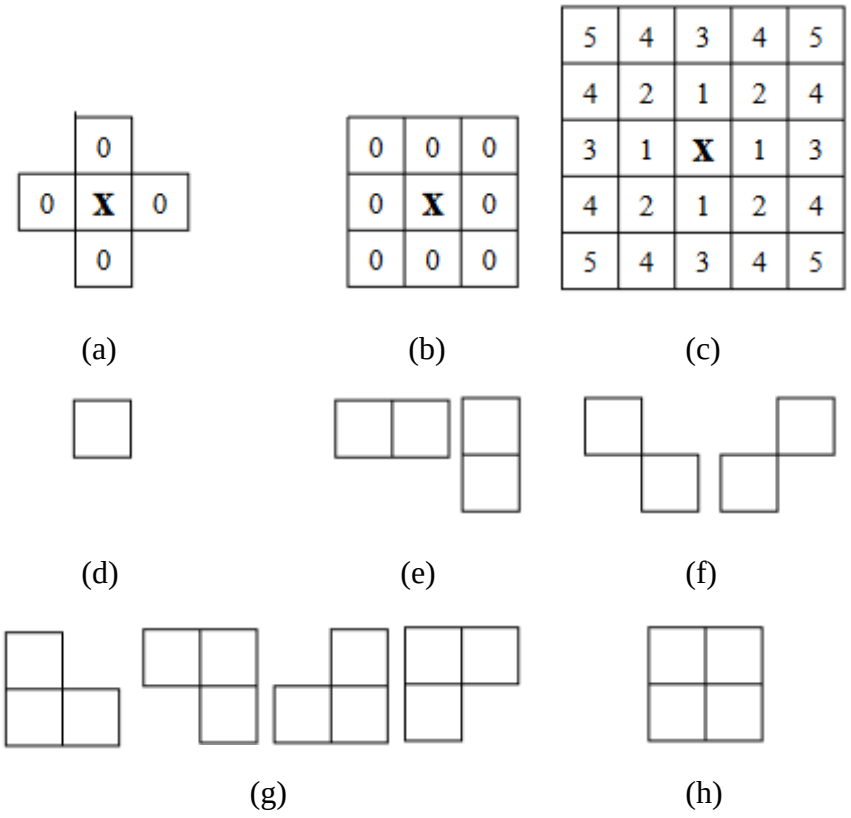
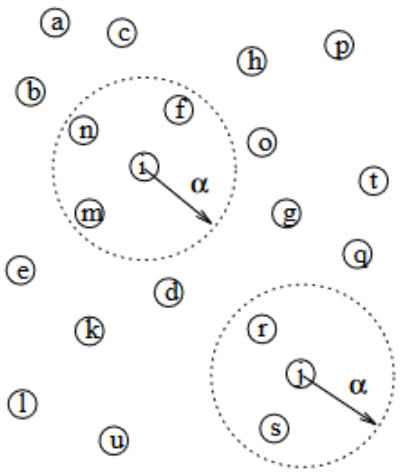


Figure 3.1 - Voisinage et cliques.

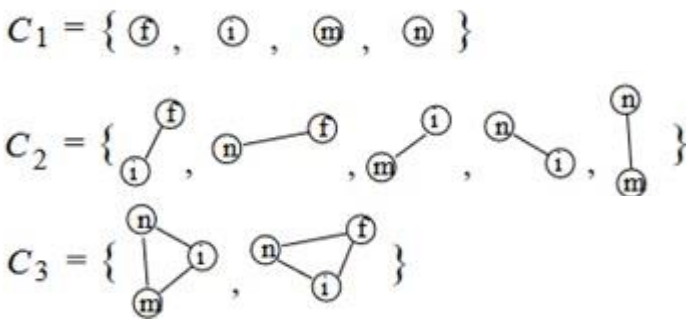
Dans le système de voisinage de premier ordre, également appelé système à 4 voisinages, chaque site (intérieur) a quatre voisins, comme le montre la Figure 3.1(a) où x désigne le site considéré. Dans le système de voisinage de second ordre, également appelé système 8-voisinages, il y a huit voisins pour chaque site (intérieur), comme le montre la Figure 3.1(b).

Les nombres $n = 1\,;::::\,;5$ illustrés à la Figure 3.1(c) indiquent les sites voisins les plus éloignés dans le système de voisinage du n-ième ordre. La forme d'un ensemble de voisins peut être décrite comme la coque renfermant tous les sites de l'ensemble.

Lorsque l'ordre des éléments de S est spécifié, l'ensemble voisin peut être déterminé plus explicitement. Par exemple, lorsque $S = \{1\,;::::\,;m\}$ est un ensemble ordonné de sites et ses éléments indexent les pixels d'une image 1D, un site intérieur $i \in \{2\,;::::\,;m-1\}$ a deux plus proches voisins, $\mathcal{N}_i = \{i-1\,, i+1\}$, et un site aux frontières (les deux extrémités) a un voisin chacun, $\mathcal{N}_1 = \{2\}$ et $\mathcal{N}_m = \{m-1\}$. Lorsque les sites d'un réseau rectangulaire régulier $S = \{(i\,;j) \mid 1 \le i\,;j \le n\}$ correspondent aux pixels d'une image $n \times n$ dans le plan 2D, un site interne $(i\,;j)$ a quatre plus proches voisins car $\mathcal{N}_{i\,;j} = \{(i-1\,,j),\ (i+1,j),(i,j-1),(i,j+1)\}$, un site à une frontière a trois voisins et un site aux coins en a deux. Pour un ensemble *S* irrégulier, le voisin ou ensemble $\mathcal{N}_i$ de i est défini de la même manière que (3.18) pour comprendre les sites proches dans le rayon $\sqrt{r}$.



(a)



(b)

Figure 3.2 - Voisinage et cliques sur un ensemble de sites irréguliers.

$$\mathcal{N}_i = \{i' \in S \mid [dist(feature_{i'}, feature_i)]^2 \leq r,\ i' \neq i\} \tag{3.19}$$

En général, les ensembles voisins $\mathcal{N}_i$ pour un S irrégulier ont des formes et des tailles variables. Les sites irréguliers et leurs voisinages sont illustrés à la Figure 3.2(a). Les zones de voisinage pour les sites i et j sont marquées par les cercles en pointillés.

Le couple $(S\,;\,\mathcal{N}) = G$ constitue un graphe au sens usuel ; S contient les nœuds et $\mathcal{N}$ détermine les liens entre les nœuds selon la relation de voisinage.

Une clique C pour $(S\,;\,\mathcal{N})$ est définie comme un sous-ensemble de sites dans S. Il est constitué soit d'un seul site $c = \{i\}$, soit d'une paire de sites voisins $C = \{i\,;\,i'\}$, soit d'un triplé de sites voisins $C = \{i\,;\,i'\,;\,i''\}$, et ainsi de suite. Les collections de cliques à un seul site, à deux sites et à trois sites seront notées C_1, C_2 et C_3, respectivement, où :

$$C_1 = \{i \mid i \in S\} \tag{3.20}$$

$$C_2 = \{\{i, i'\} \mid i' \in \mathcal{N}_i, i \in S\} \tag{3.21}$$

$$C_3 = \{\{i, i', i''\} \mid i, i', i'' \in S \text{ sont voisins }\} \tag{3.22}$$

Les sites d'une clique sont ordonnés, et $\{i, i'\}$ n'est pas la même clique que $\{i', i\}$, et ainsi de suite. La collection de toutes les cliques pour $(S\,;\,\mathcal{N})$ est :

$$C = C_1 \cup C_2 \cup C_3 \ldots \tag{3.23}$$

Le type d'une clique pour $(S\,;\,\mathcal{N})$ d'un réseau régulier est déterminé par sa taille, forme et orientation. Les Figures 3.1(d)-(h) montrent les types de cliques pour les systèmes de voisinage de premier et second ordre pour un réseau. Les cliques à un seul site et à paires horizontales et verticales dans (d) et (c) sont toutes celles du système de voisinage de premier ordre (a). Les types de cliques pour le système de voisinage de second ordre (b) incluent non seulement ceux de (d) et (c) mais aussi les cliques diagonales à deux sites (f) et les cliques à trois sites (g) et à quatre sites (h). Comme l'ordre de système de voisinage augmente, le nombre de cliques croît rapidement et implique des dépenses en calcul.

Les cliques pour les sites irréguliers n'ont pas de formes fixes comme celles d'un réseau régulier. Par conséquent, leurs types sont essentiellement représentés par le nombre de sites impliqués. Considérant les quatre sites f, i, m et n dans le cercle de la Figure 2.2(a) dans lequel m et n sont supposés être voisins l'un de l'autre, tout comme n et f. Ensuite, les cliques à un seul site, à deux sites et à trois sites associées à cet ensemble de sites sont présentées à la Figure 3.2(b). L'ensemble {m ; i ; f} ne forme pas une clique car f et m ne sont pas voisins.

3.3.1.2 Champs de Markov aléatoires (Définition)

Soit $F = \{F_1 ; : : : ; F_m\}$ une famille de variables aléatoires définies sur l'ensemble S, dans laquelle chaque variable aléatoire F_i prend une valeur f_i dans L. La famille F est appelée champ aléatoire. On utilise la notation $F_i = f_i$ pour désigner l'événement où F_i prend la valeur f_i et la notation $(F_1 = f_1 ; : : : ; F_m = f_m)$ pour désigner l'événement conjoint. Pour plus de simplicité, un événement conjoint est abrégé en $F = f$ où $f = \{f_1 ; : : : ; f_m\}$ est une configuration de F, correspondant à une réalisation du champ. Pour un ensemble discret d'étiquettes L, la probabilité que la variable aléatoire F_i prenne la valeur $f_i$est notée $P(F_i = f_i)$, abrégée $P(f_i)$ sauf s'il est nécessaire d'élaborer les expressions, et la probabilité jointe est notée $P(F = f) = P(F_1 = f_1 ; : : : ; F_m = f_m)$ et abrégé $P(f)$. Pour un L continu, nous avons des fonctions de densité de probabilité, $p(F_i = f_i)$ et $p(F = f)$.

On dit que F est un champ aléatoire de Markov sur S par rapport à un système de voisinage $\mathcal{N}$ si et seulement si les deux conditions suivantes sont satisfaites :

$$P(f) > 0, \forall f \in F \quad (Positivité) \tag{3.24}$$

$$P(f_i \mid f_{s-\{i\}}) = P(f_i \mid f_{\mathcal{N}_i}) \quad (Markovianité) \tag{3.25}$$

Où $S - \{i\}$ est la différence d'ensemble, $f_{s-\{i\}}$ désigne l'ensemble d'étiquettes aux sites dans $S - \{i\}$ et

$$f_{\mathcal{N}_i} = \{f_{i'} \mid i' \in \mathcal{N}_i\} \tag{3.26}$$

Représente l'ensemble des labels aux sites voisins de i. La positivité est supposée pour certaines raisons techniques et peut généralement être satisfait dans la pratique. Par exemple, lorsque la condition de positivité est satisfaite, la probabilité conjointe $P(f)$ de tout champ aléatoire est uniquement déterminée par ses probabilités conditionnelles locales [60]. La Markovianité décrit les caractéristiques locales de F. Dans les MRF, seules les étiquettes voisines ont des interactions directes les unes avec les autres. Si nous choisissons

le plus grand système de voisinage dans lequel les voisins de n'importe quel site incluent tous les autres sites, alors tout F est un MRF par rapport à un tel système de voisinage.

3.3.1.3 Champs de Gibbs aléatoires

Un ensemble de variables aléatoires F est dit un champ aléatoire de Gibbs (GRF) sur S par rapport à $\mathcal{N}$ si et seulement si ses configurations obéissent à une distribution de Gibbs. Une distribution de Gibbs prend la forme suivante :

$$P(f) = Z^{-1} \text{ x } e^{-\frac{1}{T}U(f)} \tag{3.27}$$

Où

$$Z = \sum_{f \in \mathbb{F}} e^{-\frac{1}{T}U(f)} \tag{3.28}$$

Est une constante de normalisation appelée fonction de partition, T est une constante appelée la température qui doit être supposée égale à 1 sauf indication contraire, et $U(f)$ est la fonction d'énergie.

$$U(f) = \sum_{c \in C} V_c\,(f) \tag{3.29}$$

L'énergie est une somme des potentiels de clique $V_c(f)$ sur toutes les cliques possibles C. La valeur de $V_c(f)$ dépend de la configuration locale de la clique c. La distribution gaussienne est un membre spécial de cette famille de distribution de Gibbs. Un GRF est dit homogène si $V_c(f)$ est indépendant de la position relative de la clique c dans S. Il est dit isotrope si V_c est indépendant de l'orientation de c.

Pour calculer une distribution de Gibbs, il faut évaluer la fonction de partition Z qui est la somme sur toutes les configurations possibles dans F. Donc, c'est un nombre combinatoire d'éléments dans F pour un L discret. L'évaluation est prohibitive même pour des problèmes de tailles modérées. Plusieurs méthodes d'approximation existent pour résoudre ce problème.

$P(f)$ Mesure la probabilité d'occurrence d'une configuration particulière f. Les configurations les plus probables sont celles avec des énergies plus faibles. La température T contrôle la netteté de la distribution. Lorsque la température est élevée, toutes les configurations ont tendance à être également réparties. Près de la température zéro, la distribution se concentre autour des minima de l'énergie globale. Étant donné T et $U(f)$, nous pouvons générer une classe de motifs (patterns) par échantillonnage de l'espace de

configuration F selon $P(f)$. Pour les problèmes d'étiquetage discrets, un potentiel de clique $V_c(f)$ peut être spécifié par un certain nombre de paramètres. Par exemple, soit $f_c = (f_i\,; f_{i'}\,; f_{i''})$ la configuration locale sur une triple-clique $c = \{i\,;\, i'\,;\, i''\}$, f_c prend un nombre fini d'états et donc $V_c(f)$ prend un nombre fini de valeurs. Pour les problèmes d'étiquetage continu, f_c peut varier de façon continue. Dans ce cas, $V_c(f)$ est une fonction continue (éventuellement par morceaux) de f_c.

3.3.1.4 Markov-Gibbs Equivalence

Un MRF est caractérisé par sa propriété locale (la Markovianité) alors qu'un GRF est caractérisé par sa propriété globale (la distribution de Gibbs). Le théorème de Hammersley-Cliford [62] établit l'équivalence de ces deux types de propriétés. Le théorème énonce que F est un MRF sur S par rapport à $\mathcal{N}$ si et seulement si F est un GRF sur S par rapport à $\mathcal{N}$. De nombreuses preuves du théorème existent, par ex. dans [60 ; 63 ; 64].

La valeur pratique du théorème est qu'il fournit un moyen simple pour spécifier la probabilité conjointe. On peut spécifier la probabilité conjointe $P(F = f)$ en spécifiant les fonctions de potentiel de clique $V_c(f)$ et en choisissant les fonctions de potentiel appropriées pour le comportement souhaité du système. De cette manière, on encode la connaissance a priori ou la préférence sur les interactions entre étiquettes. Le choix des formes et des paramètres des fonctions de potentiel pour un encodage correct des contraintes est un sujet majeur dans la modélisation MRF. Les formes des fonctions de potentiel déterminent la forme de la distribution de Gibbs. Lorsque tous les paramètres impliqués dans les fonctions de potentiel sont spécifiés, la distribution de Gibbs est complètement définie.

3.3.2 Framework MAP/MRF en segmentation d'image

Les modèles MRF en vision par ordinateur sont devenus populaires et le domaine s'est développé rapidement en s'adressant à une variété de tâches de bas niveau.

Passons maintenant à la formulation mathématique d'un modèle d'image MRF. Soit $\mathcal{R} = \{r_1, r_2, \dots, r_m\}$ un ensemble de sites et $\mathcal{F} = \{F_r : r \in \mathcal{R}\}$ un ensemble de données images (ou observations) sur ces sites. L'ensemble de toutes les observations possibles $f = (f_{r_1}, f_{r_2}, \dots, f_{r_m})$ est noté ϕ. De plus, on nous donne un autre ensemble de sites

$S = \{s_1, s_2, \dots, s_n\}$, chacun de ces sites peut prendre une étiquette parmi $\Lambda = \{0, 1, \dots, L-1\}$. L'espace de configuration Ω est l'ensemble de tous les étiquetages discrets globaux

$\omega = (\omega_{s_1}, \ldots, \omega_{s_n})$, $\omega_s \in \Lambda$. Les deux ensembles de sites $\mathcal{R}$ et S ne sont pas nécessairement disjoints, ils peuvent avoir des parties communes ou se référer à un ensemble commun de sites. Notre objectif est de modéliser les étiquettes et les observations avec un champ aléatoire conjoint $(\mathcal{X}, \mathcal{F}) \in \Omega \times \phi$. Le champ $\mathcal{X} = \{X_s\}_{s \in S}$ est appelé l'étiquette et $\mathcal{F} = \{F_r\}_{r \in \mathcal{R}}$ est appelé le champ d'observation.

3.3.2.1 Estimation bayésienne

Tout d'abord, nous construisons un estimateur bayésien pour le champ d'étiquette. Les probabilités conjointes et conditionnelles peuvent être définies en termes de distributions a priori et a posteriori :

$$P_{\mathcal{X},\mathcal{F}}(\omega, f) = P_{\mathcal{F}|\mathcal{X}}(f \mid \omega) P_{\mathcal{X}}(\omega) \tag{3.30}$$

$$P_{\mathcal{X}|\mathcal{F}}(\omega \mid f) = \frac{P_{\mathcal{X},\mathcal{F}}(\omega, f)}{P_{\mathcal{F}}(f)} = \frac{P_{\mathcal{F}|\mathcal{X}}(f \mid \omega) P_{\mathcal{X}}(\omega)}{P_{\mathcal{F}}(f)} \tag{3.31}$$

La réalisation du champ d'observation étant connue, *P* (*f*) est constant et on peut écrire :

$$P_{\mathcal{X}|\mathcal{F}}(\omega \mid f) \propto P_{\mathcal{F}|\mathcal{X}}(f \mid \omega) P_{\mathcal{X}}(\omega) \tag{3.32}$$

L'estimateur peut être formulé comme la fonction de décision suivante δ [24] :

$$\delta : \phi \to \Omega \tag{3.33}$$

$$f \mapsto \delta(f) = \hat{\omega} \tag{3.34}$$

Et le risque de Bayes correspondant [24] est donné par :

$$r(P_X, \delta) = E\{R(\omega, \delta(f))\} \tag{3.35}$$

Où $R(\omega, \delta(f))$ est une fonction de coût définie ultérieurement. Selon la règle de décision bayésienne [65], notre estimateur doit correspondre au risque bayésien minimal :

$$\hat{\omega} = \arg\min_{\omega' \in \Omega} \sum_{\omega \in \Omega} R(\omega, \omega') P_{\mathcal{X}|\mathcal{F}}(\omega \mid f) \tag{3.36}$$

3.3.2.2 Maximum A Posteriori (MAP)

L'estimateur MAP est l'estimateur le plus utilisé en traitement d'images. Sa fonction de coût est définie par :

$$R(\omega, \omega') = 1 - \delta(\omega', \omega) \tag{3.37}$$

Où $\delta(\omega', \omega)$ est le delta de Kronecker. Clairement, cette fonction a le même coût pour toutes les configurations différentes de ω'. À partir des équations (3.36) et (3.37), l'estimateur MAP du champ d'étiquette est donné par :

$$\widehat{\omega}^{MAP} = \arg\max_{\omega \in \Omega} P_{\mathcal{X}|\mathcal{F}}(\omega \mid f) \tag{3.38}$$

Cet estimateur fournit pour une observation donnée f, les modes de distribution postérieure, c'est-à-dire les étiquetages les plus probables compte tenu de l'observation f. L'équation (3.38) est un problème d'optimisation combinatoire qui nécessite des algorithmes spéciaux tels que le recuit simulé.

3.3.3 Exemples de modèles de Markov

3.3.3.1 Auto-Modèles

Les contraintes contextuelles sur deux étiquettes sont les contraintes d'ordre le plus bas pour transmettre des informations contextuelles. Ils sont largement utilisés en raison de leur forme simple et de leur faible coût de calcul. Ils sont codés dans l'énergie de Gibbs sous forme de potentiels de clique de paires de sites. Avec des potentiels de clique allant jusqu'à deux sites, l'énergie prend la forme :

$$U(f) = \sum V_1(f_i) + \sum\sum V_2\,(f_i, f_{i'}) \tag{3.39}$$

Un GRF ou un MRF spécifique peut être spécifié par une sélection appropriée des V_1 et V_2. Certains modèles GRF importants sont décrits ensuite.

Lorsque $V_1(f_i) = f_i G_i(f_i)$ et, $V_2(f_i, f_{i'}) = \beta_{i,i'}\, f_i\, f_{i'}$ où $G_i(.)$ sont des fonctions arbitraires et i ; i' sont des constantes reflétant l'interaction *pair-site* entre i et i' , l'énergie est :

$$U(f) = \sum f_i G_i\,(f_i) + \sum \beta_{i,i'}\; f_i\, f_{i'} \tag{3.40}$$

Ce qui précède est appelé auto-modèles [60]. Les auto-modèles peuvent ensuite être classés en fonction des hypothèses faites sur les f_i individuels. Un auto-modèle est dit auto-logistique, si les f_i prennent des valeurs dans l'ensemble d'étiquettes discrètes $L = \{0\ ;\ 1\}$ (ou $L = \{-1, +1\}$). L'énergie correspondante est de la forme suivante :

$$U(f) = \sum_{\{i\} \in C_1} \alpha_i f_i + \sum_{\{i,i'\} \in C_2} \beta_{i,i'}\, f_i\, f_{i'} \tag{3.41}$$

Un auto-modèle est dit auto-binomial si les f_i prennent des valeurs dans

$\{0\ ,1,\ldots\ ,M-1\}$ et chaque f_i a une distribution conditionnellement binomiale de M essais et de probabilité de succès q.

Un auto-modèle est dit auto-normal, aussi appelé MRF gaussien [66], si l'ensemble d'étiquettes L est la ligne réelle et la distribution conjointe est normale multivariée.

Un modèle connexe mais différent est le modèle d'auto régression simultanée (SAR). Contrairement au modèle auto-normal qui est défini par les m fonctions de densité de probabilité conditionnelle, ce modèle est défini par un ensemble de m équations.

3.3.3.2 Modèle logistique multi-niveaux

Le modèle auto-logistique peut être généralisé au modèle logistique multi-niveau (MLL) [67 ; 68 ; 69], également appelé processus de Strauss [70] et modèle d'Ising généralisé [50]. Il y a $M\ (>\ 2)$ étiquettes discrètes dans l'ensemble d'étiquettes, $L\ =\ \{1\ ;:::;M\}$. Dans ce type de modèles, un potentiel de clique dépend du type c (lié à la taille, la forme et éventuellement l'orientation) de la clique et de la configuration locale $f_c\ \triangleq \{f_i \mid i\ \in\ c\}$. Pour les cliques contenant plus d'un site $(\#\ c\ >\ 1)$, les potentiels de clique MLL sont définis par :

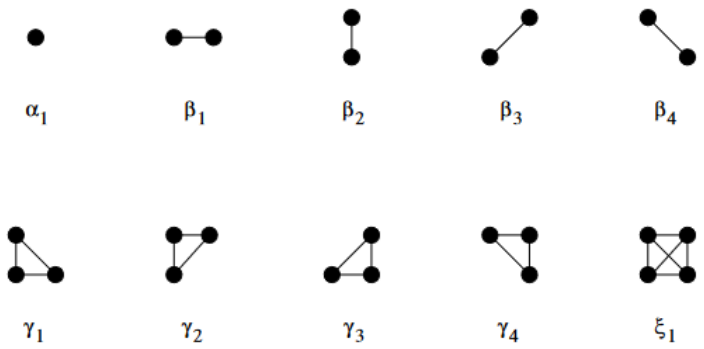


Figure 3.3 - Types de cliques et paramètres potentiels associés pour le système de voisinage de second ordre.

$$V_c(f) = \begin{cases} \varsigma_c & Si\ tous\ les\ sites\ sur\ c\ ont\ le\ même\ label \\ -\varsigma_c & Autrement \end{cases} \qquad (3.42)$$

Un modèle de Gibbs hiérarchique à deux niveaux a été proposé pour représenter à la fois les images contaminées par le bruit et les images texturées [68 ; 69]. La distribution de Gibbs de niveau supérieur utilise un champ aléatoire isotrope, par ex. MLL, pour

caractériser le processus de formation de la région de type blob. Une distribution de Gibbs de niveau inférieur décrit le remplissage dans chaque région. Le remplissage peut être un bruit indépendant ou un type de texture, les deux pouvant être caractérisés par des distributions de Gibbs. Cela fournit une approche pratique pour la modélisation MAP-MRF. En segmentation d'image bruitée et texturée [68 ; 69 ; 71 ; 72 ; 73], par exemple, le niveau supérieur détermine l'a priori de f pour le processus de région tandis que le niveau inférieur de Gibbs contribue à la probabilité conditionnelle des données sachant f. Les différents niveaux de MRF dans la hiérarchie peuvent avoir différents systèmes de voisinage.

3.3.3.3 Modèle GRF hiérarchique

Différents modèles de Gibbs hiérarchiques résultent du modèle de Gibbs hiérarchique à deux niveaux en fonction de ce qui est choisi pour les régions et pour les remplissages, respectivement. Par exemple, chaque région peut être remplie par une texture auto-normale [74 ; 73] ou une texture auto-binomiale [72] ; le modèle MLL pour la formation de région peut être remplacé par un autre modèle MRF approprié.

Un inconvénient du modèle hiérarchique est que la probabilité conditionnelle $P\ (d_i \mid f_i = I)$ pour les régions données par $\{i \in S \mid f_i = I\}$ ne peut pas toujours s'écrire exactement. Par exemple, lorsque le MRF de niveau inférieur est une texture modélisée comme un champ auto-normal, sa distribution conjointe sur une région de forme irrégulière n'est pas connue. Cette difficulté peut être surmontée en utilisant des schémas approximatifs tels que la pseudo-vraisemblance ou en utilisant la méthode d'analyse propre [75].

3.3.3.4 Le modèle FRAME

Le modèle FRAME (Filter, Random Fields And Maximum Entropy), proposé par [76 ; 77 ; 78], est un modèle MRF généralisé qui fusionne l'essence de la théorie du filtrage et de la modélisation MRF à travers le principe d'entropie maximale. Il est généralisé dans les deux aspects suivants : (1) Le modèle FRAME est défini en termes de statistiques, c'est-à-dire de fonctions potentielles, calculées à partir de la sortie d'un banc de filtres par lequel l'image est filtrée, au lieu des potentiels de cliques de l'image elle-même. Etant donné une image (une réalisation d'un MRF), l'image est filtrée par un banc de filtres, donnant un ensemble d'images de sortie. Certaines statistiques sont ensuite calculées à partir des images de sortie. (2) Le modèle FRAME permet d'apprendre les paramètres du modèle à

partir d'un ensemble d'échantillons (exemples d'images) représentatifs du MRF à modéliser. En outre, il donne également un algorithme pour la sélection des filtres.

Le modèle FRAME fournit un moyen de modéliser des motifs complexes d'ordre élevé d'une manière traitable. Dans le modèle MRF traditionnel, le voisinage est généralement petit pour que le modèle reste traitable, et il est donc difficile de modéliser des motifs dans lesquels l'interaction dans un grand voisinage est nécessaire. En revanche, le modèle FRAME est capable de modéliser des motifs plus compliqués en incorporant un voisinage plus large et des fonctions de potentielles de cliques d'ordre supérieur implicitement déterminées par les fenêtres de filtre ; de plus, il utilise une procédure d'apprentissage d'accompagnement pour estimer les fonctions de potentielles d'ordre élevé à partir des sorties du filtre. Cela rend le modèle d'ordre élevé facile à formuler bien qu'il est coûteux en calcul. Il y a deux choses à apprendre dans le modèle FRAME : (1) les fonctions de potentielles $\theta_l^{(k)}$ et (2) les types de filtres $G^{(k)}$ à utiliser.

3.3.3.5 Modélisation MRF multi-résolution

La motivation de la modélisation MRF multi-résolution (MRMRF) est similaire à celle de FRAME : modéliser des motifs complexes et macro en incorporant des interactions dans un grand voisinage. L'approche utilisée dans la modélisation MRMRF consiste à construire un modèle MRF basé sur les sorties de filtres multi-résolutions.

Le modèle MRMRF tente d'incorporer des informations dans un large voisinage en fusionnant la théorie du filtrage et les modèles MRF, ce qui est similaire au modèle FRAME [78]. Comparé au modèle MRF traditionnel, le modèle MRMRF peut révéler plus d'informations contenues dans les textures puisque les images originales sont décomposées en sous-bandes d'échelles et de directions différentes et sous-échantillonnées. Pour cette raison, le modèle MRMRF est plus puissant que le modèle MRF traditionnel ; cependant, il semble moins puissant que le modèle FRAME puisque seules les interactions jusqu'au *pair-site* sont prises en compte dans le modèle MRMRF. D'un point de vue calcul, il se situe également entre le MRF traditionnel et le FRAME, le FRAME étant très coûteux en calcul.

3.4 Revue des travaux de la littérature

Nous discutons dans cette section certaines méthodes utilisant les champs de markov aléatoires (MRF) ainsi que de leur contribution.

Zheng & al. (2012) ont proposé un nouveau modèle de segmentation d'image en incorporant la résolution multi-régions et le modèle de champ de Markov aléatoire. La transformation en ondelettes est une méthode basée sur les pixels et largement utilisée pour les approches de segmentation multi-résolution, mais elle souffre du manque de modélisation du motif de macro-texture d'une image donnée et cette méthode peut améliorer la précision de la segmentation par rapport à la méthode multi-résolution basée sur le niveau de pixel.

En le combinant avec le modèle MRF, un nouveau modèle MRR-MRF est conçu pour la segmentation d'images. Par rapport aux techniques à résolution multiple fixe, la technique à résolution multi-régions utilise les régions sursegmentées comme unités de base pour réaliser la décomposition. Par conséquent, la représentation à résolution multi-régions peut décrire les macro-motifs et motifs complexes de texture d'une image donnée. Elle peut faire en sorte que le MRR-MRF s'oppose au bruit et modélise plus d'informations de texture [79].

Zhou & al. (2013) ont présenté un champ de Markov aléatoire hybride actif (DHMRF), qui capture clairement la forme de l'objet de niveau intermédiaire et l'aspect visuel de bas niveau (par exemple, la texture et la couleur) pour l'étiquetage de l'image. Chaque nœud dans DHMRF est décrit soit par un modèle déformable, soit par un modèle d'apparence en tant que prototype visuel. D'autre part, les contours encodent deux types d'intersections : la co-occurrence et le contexte spatial en couches, par rapport aux étiquettes et aux prototypes des nœuds connectés. Pour apprendre le modèle DHMRF, un algorithme itératif est conçu pour sélectionner automatiquement les caractéristiques les plus informatives et estimer les paramètres du modèle. L'algorithme atteint une efficacité de calcul élevée puisque un schéma branch-and-bound est introduit pour estimer les paramètres du modèle et enfin présenter un modèle DHMRF pour relever les défis de l'incorporation de modèles de forme explicites ainsi que de modèles d'apparence dans un cadre probabiliste de segmentation d'images multi-classes. Au moment des expérimentations ces méthodes ont été testées sur deux jeux de données : MSRC 21-class et ensemble de données de classe LHI 15 et les résultats sont évalués en termes de précision de segmentation par pixel.

Malgré une précision de pointe, des résultats encore meilleurs peuvent être obtenus en tenant compte conjointement des indices d'apparence et les caractéristiques de forme pour modéliser des objets structurés et également l'utilisation d'un modèle de forme hiérarchique basé sur les pièces, pour que les objets structurés puissent faire face aux inter- ou auto-occlusion, tout en conservant une efficacité et une précision élevées [80].

Zhu & al. (2013) ont présenté une nouvelle caractéristique visuelle psychologique basée sur le seuil différentiel et l'appliquent dans un cadre de champ de Markov aléatoire supervisé. La détection fiable de la saillance peut aider dans de nombreux traitements utiles sans connaissance préalable de la scène, tels que la compression d'image à contenu sensible, la segmentation, etc. Même si de nombreux efforts ont été consacrés à ce sujet, l'expression des caractéristiques et la construction du modèle sont plus loin d'être parfaites. Les cartes de saillance obtenues ne sont donc pas suffisamment satisfaisantes. Afin de surmonter ces défis et à la fin de la présentation, une méthode supervisée pour la détection de la saillance a été présentée. La méthode proposée incorpore principalement la seule caractéristique de saillance purement inspirée, la caractéristique de couleur basée sur un seuil différentiel, pour que chaque pixel soit saillant grâce à l'apprentissage MRF. Ses performances ont été évaluées sur deux ensembles de données d'images publiques et la méthode proposée surpasse les 14 autres méthodes de pointe pour la détection de saillance en termes d'efficacité et de robustesse. Une application de la sculpture sur couture est également impliquée, ce qui illustre intuitivement l'utilité de la méthode proposée [81].

Ghamisi & al. (2013) ont proposé un nouveau framework automatique pour la classification des images hyper spectrales. La nouvelle méthode est basée sur la combinaison d'une segmentation basée sur un champ de Markov aléatoire caché avec un classificateur de machine à vecteurs de support (SVM). Afin de conserver des limites dans la carte de classement finale, un pas de pente est pris en compte.

La technologie de télédétection hyper spectrale permet d'obtenir une séquence de centaines d'images spectrales contiguës allant de l'ultraviolet à l'infrarouge. Les classificateurs spectraux carrés traitent les images hyper spectrales comme une liste de mesures spectrales et ne tiennent pas compte des dépendances spatiales, ce qui conduit à une diminution théâtrale des précisions de classification et enfin un framework entièrement automatisé qui prend en compte à la fois les informations spectrales et les informations spatiales a été introduit pour la classification d'images hyper spectrales. Dans le framework, SVM est

utilisé pour l'extraction d'informations spectrales. En parallèle, HMRF-M est utilisé pour la suppression des informations spatiales.

L'efficacité de la méthode proposée est testée dans les deux situations avec et sans prise en compte du pas de gradient. La méthode proposée est évaluée sur deux jeux de données (Indian Pines et Salinas). Dans les deux cas, la nouvelle approche surpasse les autres méthodes étudiées. Il est à noter que le concept de HMRF est utilisé pour la première fois dans le domaine de la télédétection dans cet article, et l'efficacité de celui-ci pour la segmentation d'images hyper spectrales est démontrée. Enfin, il est montré dans cet article que la méthode est performante en termes de précision par rapport à l'état de l'art. De plus, l'approche proposée est entièrement automatique et facile à utiliser par rapport à la plupart des méthodes [82].

Gong & al. (2013) ont présenté une nouvelle approche pour la détection des changements dans les images radar à synthèse d'ouverture (SAR). L'approche classe les régions modifiées et les régions inchangées par clustering flou de c-means (FCM) avec une nouvelle fonction d'énergie de champ de Markov aléatoire (MRF). Afin de réduire l'effet de bruit de défiguration, une nouvelle forme de fonction d'énergie MRF avec un terme supplémentaire est établie pour modifier l'appartenance de chaque pixel et le degré de modification est déterminé par la relation des pixels voisins. La forme spécifique du terme supplémentaire dépend de différentes situations et est finalement établie en utilisant la méthode des moindres carrés ; leurs contributions résident dans deux aspects.

Premièrement, afin de réduire l'effet du bruit de déblais, l'approche proposée se concentre sur la modification de l'appartenance au lieu de modifier la fonction objective. Ce qui simplifie le calcul dans toutes les étapes impliquées. Sa fonction objective peut simplement revenir à la forme originale de FCM, ce qui entraîne évidemment une consommation de temps inférieure à celle de certains algorithmes FCM récemment améliorés. Deuxièmement, l'approche proposée modifie l'appartenance de chaque pixel selon une nouvelle forme de fonction d'énergie MRF à travers laquelle les voisins de chaque pixel ainsi que leur relation sont concernés. L'analyse théorique et les résultats expérimentaux sur de vrais ensembles de données SAR montrent que l'approche proposée peut détecter les changements réels et atténuer l'effet des bruits de chatoiement [83].

Xu & al. (2012) ont proposé une méthode de classification sémantique non supervisée efficace pour les images satellites à haute résolution et ajoutent un coût d'étiquette, qui peut

structurer une solution basée sur un ensemble d'étiquettes déterminées par optimisation de l'énergie, dans les champs aléatoires sur des sujets cachés, et un algorithme itératif est ainsi proposé pour que le nombre de classes soit finalement converti à un niveau approprié. Par rapport aux autres algorithmes de classification mentionnés, cette méthode peut non seulement obtenir des résultats de segmentation sémantique précis par des structures à plus grande échelle, mais peut également attribuer automatiquement le nombre de segments et, à la portée à l'extrémité de destination, découvrir qu'un algorithme de classification sémantique non vérifiée a été proposé pour les images satellites à haute résolution. L'algorithme itératif basé sur le coût de l'étiquette et le BIC peut déterminer automatiquement le nombre de classes dans la classification. Il peut également conserver la cohérence de la sémantique. L'évaluation sur quatre scènes a montré que la méthode proposée permet d'obtenir de meilleures performances de classification [84].

Eches & al. (2012) ont proposé les régions des sites MRF. Ces régions ont construit un filtre de zone auto-complémentaire qui découle de la théorie morphologique. Ce type de filtre divise l'image originale en zones plates où les pixels de base ont les mêmes valeurs spectrales. Une fois le MRF clairement établi, un algorithme bayésien hiérarchique est proposé pour estimer les abondances, les étiquettes de classe, la variance du bruit et les hyper paramètres correspondants. Un échantillonneur de Gibbs hybride est construit pour générer des échantillons en fonction de la distribution a posteriori correspondante des paramètres inconnus et hyper paramètres. Des travaux récents ont montré que l'exploitation des dépendances spatiales entre les pixels de l'image peut améliorer le démixage spectral. Les champs aléatoires de Markov (MRF) sont classiquement utilisés pour modéliser ces corrélations spatiales et partitionner l'image en plusieurs classes avec des abondances homogènes. Enfin, une approche entièrement bayésienne estimant conjointement la matrice des membres finaux, les régions de similarité et les autres paramètres d'intérêt mériterait d'être étudiée [85].

Hochbaum, D. S. (2001) a proposé un algorithme qui résout le problème en temps polynomial lorsque la fonction de déviation est convexe et la fonction de séparation est linéaire ; et en temps fortement polynomial lorsque la fonction de coût d'écart est linéaire, quadratique ou convexe et linéaire par morceaux avec peu de morceaux (où « peu » signifie un nombre exponentiel dans une fonction polynomiale du nombre de variables et de contraintes).

Parmi les problèmes de champ aléatoire de Markov, il existe également une relation par paires entre les objets. L'objectif dans le problème de champ aléatoire de Markov est de minimiser la somme de la fonction de coût d'écart et d'une fonction de pénalité qui croît avec la distance entre les valeurs de la fonction de séparation des paires liées [86].

Kwon & al. (2000) ont proposé un classificateur de page Web basé sur une édition de l'approche k-Nearest Neighbor (k-NN). Pour améliorer les performances de l'approche k-NN, on complète l'approche k-NN avec une méthode de sélection de caractéristiques et un schéma de sélection de termes utilisant des balises, et on réforme les mesures de similarité des documents utilisées dans le modèle d'espace vectoriel et on découvre que la mesure de similarité a permis d'améliorer l'efficacité de l'approche k-NN. De plus, l'utilisation de la sélection de caractéristiques dans l'approche k-NN a amélioré les performances dans le test de validation [87].

Zhang & al. (2009) ont développé une nouvelle technique de détection de contours utilisant le modèle de Markov caché 3-D et son modèle proposé peut non seulement capturer la relation des coefficients d'ondelettes inter-échelle, mais prend aussi en considération la dépendance intra-échelle. Un algorithme d'estimation du maximum de vraisemblance (ML) efficace sur le plan de calcul est utilisé pour calculer les paramètres, et l'état caché de chaque coefficient est découvert par une estimation maximum a posteriori (MAP) et son modèle a le potentiel d'être un outil de modélisation arithmétique multi-échelle efficace pour d'autres tâches de traitement d'images ou de vidéos [88].

Tu & al. (2009) ont proposé un algorithme d'apprentissage, auto-contexte. Étant donné un ensemble d'images d'entraînement et leurs cartes d'étiquettes correspondantes, ils apprennent d'abord un classificateur sur des patchs d'image locaux. Les cartes de confiance de classification créées par le classificateur appris sont ensuite utilisées comme informations de contexte, en plus des patchs d'image d'origine, pour former un nouveau classificateur. L'algorithme itère ensuite jusqu'à convergence. L'auto-contexte intègre des informations de bas niveau et de contexte en fusionnant un grand nombre de fonctionnalités d'apparence de bas niveau avec des informations de contexte et de forme non exprimées. L'algorithme discriminant qui en résulte est général et facile à mettre en œuvre. Sous presque les mêmes réglages de paramètres lors de l'apprentissage, ils appliquent l'algorithme à trois applications de vision difficiles : la ségrégation d'arrière-plan, l'estimation de la configuration du corps humain et l'étiquetage des régions de la scène. De plus, le contexte joue également un rôle très important dans les images médicales/cérébrales

où les structures anatomiques sont pour la plupart contraintes à des positions relativement fixes [89].

Nascimento & al. (2011) ont introduit une nouvelle méthode de démixage hyper spectral non vérifié conçue pour des ensembles de données hyper spectraux linéaires mais hautement mélangés, dans laquelle le simplexe de volume minimum, généralement estimé par des algorithmes purement géométriques, est très éloigné du vrai simplexe lié aux membres terminaux. La méthode proposée, prolongement de leurs études antérieures, fait appel au Framework numérique. La fraction de richesse a priori est un mélange de densités de Dirichlet, appliquant ainsi automatiquement les contraintes sur les fractions d'abondance imposées par le processus d'acquisition, à savoir la non négativité et la somme à un. Un algorithme de minimisation cyclique est développé où les éléments suivants sont observés : 1) Le nombre de modes de Dirichlet est déduit sur la base du principe de longueur de description minimale ; 2) un algorithme de maximisation de probabilité généralisé est dérivé pour supposer les paramètres du modèle ; et 3) une séquence d'optimisations basées sur le lagrangien augmenté est utilisée pour calculer les signatures des membres finaux [90].

Yang & al. (2011) ont donné le concept de modèle en couches pour la détection d'objets et la segmentation d'images et décrivent un modèle probabiliste génératif qui compose la sortie d'une banque de détecteurs d'objets afin de définir des masques de forme et expliquer l'apparence, l'ordre de profondeur et les étiquettes de tous les pixels d'une image. En particulier, ce système estime à la fois les étiquettes de classe et les étiquettes d'instance d'objet. S'appuyer sur les critères de référence précédents pour la détection d'objets et la segmentation d'images et définir un nouveau score qui évalue à la fois la segmentation des classes et des instances [91].

Khalifa & al. (2011) ont proposé un nouveau Framework automatique pour analyser l'épaisseur de paroi et la fonction d'épaississement de ces images qui se compose de trois étapes principales. Tout d'abord, les bordures des parois internes et externes sont segmentées de leurs tissus environnants avec un modèle déformable arithmétique guidé par une relation de vitesse stochastique spéciale. Ce dernier prend en compte des modèles de forme et d'apparence de Markov-Gibbs pour l'objet d'intérêt et son arrière-plan. Dans la deuxième étape, les correspondances point à point entre les frontières intérieure et extérieure sont trouvées en résolvant l'équation de Laplace et fournissent des estimations initiales de l'épaisseur de paroi locale et de l'indice de la fonction d'épaississement.

Enfin, les effets de l'erreur de segmentation sont réduits et une analyse continue de l'épaississement de la paroi est effectuée par minimisation itérative de l'énergie à l'aide d'un modèle d'image de champ aléatoire de Gauss Markov généralisé (GGMRF) [92].

Coletta al. (2011) ont donné une certaine extension des deux algorithmes de clustering basés sur FCM. Ces algorithmes utilisés pour regrouper les données distribuées en arrivant à des moyens positifs pour déterminer les paramètres importants des algorithmes (y compris le nombre de clusters), et en formant un ensemble de directives structurées systématiquement comme la sélection de l'algorithme spécifique, en fonction de la nature de l'environnement de données et des hypothèses faites sur le nombre de clusters. Une analyse approfondie de la complexité, y compris les aspects spatiaux, temporels et de communication, est rapportée [93].

Tarabalka & al. (2011) ont proposé et étudié l'utilisation de marqueurs sélectionnés automatiquement et une nouvelle méthode HSEG basée sur des marqueurs (M-HSEG) pour la classification spectrale-spatiale d'images hyperspectrales est proposée . Deux approches basées sur la classification pour la sélection automatique des marqueurs sont adaptées et comparées à cette fin. Ensuite, un nouvel algorithme HSEG contrôlé basé sur des marqueurs est appliqué, résultant en une carte de classification spectrale-spatiale. Trois implémentations différentes de la méthode M-HSEG sont proposées et leurs performances en termes de précision de classification sont comparées en conclusion. L'approche de segmentation HSEG est l'un des rares algorithmes de segmentation de pointe qui exploite naturellement les informations spectrales et spatiales et un ensemble hiérarchique de segmentations d'images. De nombreux domaines d'application peuvent grandement bénéficier de méthodes capables d'analyser automatiquement les hiérarchies de segmentation et de sélectionner un seul niveau de segmentation optimal [94].

3.5 Conclusion

Dans ce chapitre, nous avons abordé les modèles régularisables en segmentation d'image. Notre objectif de ce chapitre de littérature est de montrer les types différents modèles régularisables, en considérant à la fois l'approche de détection de primitives d'intérêt (Contours, Région), et aussi le fondement théorique de chacun des types de modèles. Malgré, la popularité des modèles basés apprentissage, notamment ces dernières années avec la disponibilité croissante des données d'apprentissage étiquetées, les modèles

markoviens restent largement utilisables, et ont un avantage incontournable, qui est l'utilisabilité de ces modèles même en absence de toute donnée d'apprentissage.

Chapitre 4

Adaptation du filtre de Canny pour les images de profondeur

4.1 Introduction

Une étape antérieure à la régularisation des résultats de segmentation d'image, est celle de la détection, que ce soit selon l'approche contours, régions, ou autres approches hybrides. Dans ce chapitre, nous présentons notre méthode pour la détection de contours dans les images de profondeur en introduisant les algorithmes de base nécessaires pour la mettre en œuvre. Dans un travail antérieur [95], nous avons présenté un aperçu du détecteur proposé. Ainsi, dans le présent travail, nous prolongeons le travail en introduisant la méthode complète et son expérimentation approfondie, ainsi qu'une comparaison des résultats avec ceux d'autres détecteurs dédiés à l'image 2D. Les résultats de détections feront l'objet de régularisation en utilisant des méthodes de minimisation d'énergie, et utilisant le formalisme markovien. Cette partie sera présentée au chapitre suivant.

4.2 Etapes du filtre de Canny

Le filtre de Canny [96] est un détecteur de contours qui utilise un algorithme à plusieurs étapes pour détecter une large gamme de contours dans des images 2D. La détection de contour basée filtre consiste à localiser les réponses impulsionnelles élevées du filtre utilisé. L'approche utilisée par Canny est basée sur la qualité des critères requis pour un détecteur de contour optimal. L'algorithme de Canny, qui utilise un opérateur de calcul de gradient, tel que Sobel, est conçu pour être optimal selon trois critères :

- Bonne détection : l'algorithme doit marquer autant que possible tous les contours réels de l'image.

- Bonne localisation : les contours marqués doivent être aussi proches que possible des contours réels.

- Réponse minimale : un contour donné dans l'image ne doit être marqué qu'une seule fois et, dans la mesure du possible, le bruit de l'image ne doit pas créer de faux contours.

Pour satisfaire ces exigences, Canny a utilisé le calcul des variations, une technique qui trouve la fonction qui optimise une fonctionnelle donnée. La fonction optimale dans le détecteur Canny est décrite par la somme de quatre termes exponentiels, mais peut être approchée par la première dérivée d'une gaussienne.

Le détecteur Canny permet de détecter les contours qui correspondent à des variations importantes et rapides des données de l'image. Cependant, il peut ignorer les variations lentes entraînant la perte de contours dans la carte de contours finale. En effet, d'une part, si le seuil utilisé de la norme de gradient est faible, la carte de contours résultante contient tous les contours réels, mais avec une quantité de bruit élevée. En revanche, si le seuil est élevé, le bruit est faible, mais plusieurs contours réels pourraient être ignorés.

Avant de présenter le détecteur proposé pour les images de profondeur, nous passons en revue les étapes suivies par le détecteur Canny original. Ceci, peut se résumer dans les six étapes suivantes :

1. Réduction du bruit : La première étape consiste à réduire le bruit avant de détecter les contours. Le filtre 2D utilise une gaussienne [97] qui s'exprime comme suit :

$$G_\sigma(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-x_0)^2+(y-y_0)^2}{2\sigma^2}} \tag{4.1}$$

2. Calcul du gradient : Après la réduction du bruit, l'étape suivante consiste à appliquer un calcul de gradient, qui renvoie les intensités des contours. L'opérateur gradient, (Roberts, Prewitt, Sobel) [98] par exemple, retourne une estimation de la première dérivée dans la direction horizontale (G_x) et la direction verticale (G_y). La norme du gradient N et sa direction θ sont alors calculées en chaque pixel de l'image

$$N(x,y) = \sqrt{G_x(x,y)^2 + G_y(x,y)^2} \tag{4.2}$$

$$\theta = tan^{-1}\left(\frac{G_y(x,y)}{G_x(x,y)}\right) \tag{4.3}$$

L'angle de direction du contour est arrondi à l'un des quatre angles, représentant l'horizontale, la verticale et les deux diagonales (0, 90, 45 et 135 degrés) (Figure 4.1).

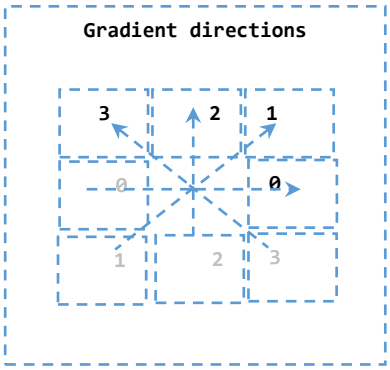


Figure 4.1 - Directions du gradient

Dans un premier temps et en utilisant la norme du gradient N, un seuillage est effectué et visant à ne conserver que les vrais contours candidats. Une telle opération aboutit à une première carte de contours, représentée par un tableau 2D *EdgeMap* :

$$EdgeMap(x,y) = \begin{cases} 1 & if N(x,y) > threshold \\ 0 & else \end{cases} \quad (4.4)$$

3. Non-maximum suppression : La carte de gradient enregistre une intensité à chaque pixel de l'image. Une intensité élevée indique une probabilité élevée qu'un contour soit présent au niveau de ce pixel. Cependant, cette intensité n'est pas suffisante pour décider si un pixel correspond ou non à un contour. Seuls les pixels correspondant aux maxima locaux sont considérés comme des pixels de contours et sont conservés pour l'étape suivante de l'algorithme. Un maximum local est situé au pixel où la dérivée de la norme du gradient est nulle.

4. Double seuillage : Deux seuils différents sont utilisés et deux cartes de contours sont alors calculées, l'une avec un seuil bas et l'autre avec un seuil haut. Le seuil bas permet d'identifier les pixels non significatifs (intensité inférieure au seuil bas). Ainsi, il peut détecter la majorité des contours et même du bruit. Le seuil haut permet d'identifier les pixels significatifs (intensité supérieure au seuil haut). Ainsi, il ne trouve que de vrais contours et sans bruit. Cependant, il peut manquer de vrais contours. De plus, la suppression

des non-maxima doit être appliquée aux deux cartes de contours. La réduction des contours à une épaisseur d'un pixel, en utilisant le processus de suppression des non-maxima, peut être exprimé par l'algorithme simplifié suivant :

```
Algorithm nonMaximaSuppression
Inputs
Map :Edgemap
Gradient :N
Outputs
Edges :Edgemap
Begin
 Edges ← Map
For each pixel P(I,j) of the image Do
  D← direction of the gradient in P
  For each of P's two neighbors in the direction D Do
    If gradient norm in the neighbor> gradient norm in P Then
      Edges(I,j)← 0 // delete this edge point
    EndIf
   EndFor
EndFor
End.
```

5. Seuillage par hystérésis : La différenciation des contours sur la carte générée se fait par seuillage. Cela nécessite deux seuils ; un seuil haut et un seuil bas, qui sera comparé à l'intensité du gradient de chaque pixel. Si l'intensité du gradient du pixel est inférieure au seuil bas, le pixel est rejeté. S'il est supérieur au seuil haut, le pixel est accepté comme faisant partie d'un contour. S'il est compris entre le seuil bas et le seuil haut, le pixel est accepté s'il est connecté à un pixel déjà accepté. Par conséquent, les deux cartes de contours obtenues sont fusionnées là où la seconde, obtenue avec le seuil haut, est préférée, si nécessaire. Donc, nous commençons par les contours de la deuxième carte, qui sont pour la plupart sans bruit, puis complétons avec les contours de la première carte.

Le pseudo-code ci-dessous montre une version récursive du seuillage par hystérésis :

```
Algorithm Hysteresis Thresholding
Inputs
Map1 : Edges map 1 (with Lower threshold )
Map2 : Edges map 2 (with higher threshold)
Outputs
Map2 : Final edges map
Begin
 NeighboringStack←Empty Stack
For each edge point Edges(I,j) of Map2 Do
  NeighboringStack.Push(Edges(I,j))
  While NeighboringStack is not empty Do
    V←NeighboringStack.pop
    For each point P of the 8 Neighbors of V Do
      If P is an edge-point in Map1 but not in Map2 Then
           Add P to Map2
           NeighboringStack.Push(P)
      EndIf
    EndFor
  EndWhile
EndFor
End.
```

4.3 Un détecteur de contour pour les images de profondeur

Les images de profondeur représentent les distances aux points de surface à partir d'un observateur, généralement situé à la caméra de profondeur. Par conséquent, le calcul du vecteur gradient par la première dérivée de la fonction image brute n'est pas adapté aux images de profondeur. Cela est dû aux discontinuités qui existent entre les différents objets ou entre les surfaces d'un même objet.

Contrairement aux images 2D, où les valeurs d'intensité des pixels voisins sont suffisantes pour calculer le gradient, dans les images de profondeur, les pixels d'un voisinage 2D local ne représentent pas nécessairement le véritable voisinage 3D, ce qui entraîne ce qu'on appelle un biais 3D (Figure 4.2).

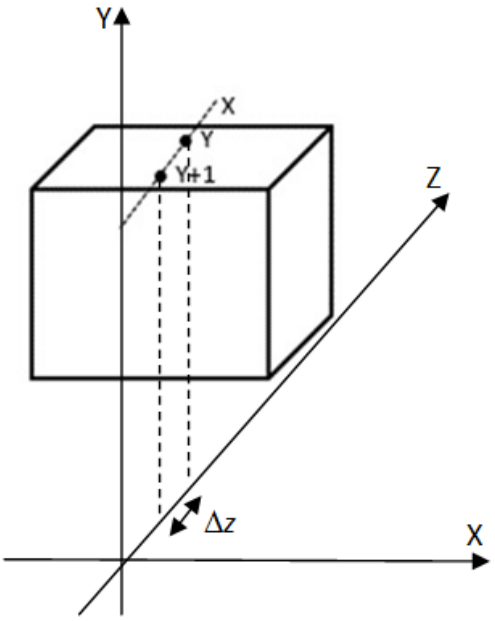


Figure 4.2 - 3D bias dans les images de profondeur

Dans la figure ci-dessus les pixels $(x, y) et\ (x, y + 1)$sont proches en 2D, mais distants selon Z $(\Delta z >> 1)$.

La discontinuité des données constitue une caractéristique des images de profondeur par rapport aux images en couleur ou en niveaux de gris. L'application du filtre Canny original aux images de profondeur entraîne à la fois une grande quantité de valeurs aberrantes et des contours perdus. Dans l'image de la Figure 4.2, la distance $\Delta z = |I\ (y + 1, x) - I\ (y, x)|$ est élevé lorsque la surface sur laquelle sont situés les deux points est orientée étroitement orthogonal à la direction d'observation. En effet, les pixels $(x, y) et\ (x, y + 1)$ sont voisins

dans le plan *(X, Y)*, mais ils sont éloignés l'un de l'autre dans l'espace. Dans les images 2D en couleur ou en niveaux de gris, cette situation ne se produit pas, car les différences sont uniformes quelle que soit l'orientation de la surface.

Par conséquent, un ensemble bien approprié de caractéristiques doit être synthétisé puis appliqué pour le calcul du gradient dans les images de profondeur. Pour notre détecteur proposé, nous calculons une nouvelle image basée surface, en ajustant l'équation du plan à chaque pixel de l'image brute, puis nous considérons les paramètres de l'équation obtenue comme l'ensemble des caractéristiques pour le calcul du vecteur gradient. Nous pouvons donc résumer notre contribution en deux points :

1) Génération d'une nouvelle image basée surface qui permet de quantifier les variations des vecteurs normaux aux surfaces. Ces variations constituent la base pour le calcul du nouveau vecteur gradient.

2) Adaptation du filtre Sobel pour utiliser les angles de l'image de surface plutôt que les données brutes de l'image de profondeur (comme dans les images 2D).

Le calcul de l'équation du plan en un pixel donné est effectué soit par produit vectoriel en considérant deux couples de vecteurs, définis par trois pixels adjacents, soit par une technique de multi-régression, comme dans plusieurs travaux de la littérature [99].

Néanmoins, il a été remarqué que la plupart des méthodes examinées calculent des approximations du plan ajusté pour une surface entière, ce qui rend le calcul de l'équation moins précis. Pour surmonter ce problème, et comme nous nous sommes concentrés uniquement sur la façon dont le contour peut être détecté localement, nous nous sommes intéressés à la façon de représenter la surface localement et de pouvoir calculer un vecteur de gradient approprié permettant la détection des contours selon les étapes du détecteur Canny. Ainsi, nous sommes contentés d'un voisinage limité autour de chaque pixel, visant à vérifier s'il s'agit d'un pixel de contour ou non.

De plus, les images de profondeur sont très bruitées par nature et nécessitent une opération de lissage pour réduire le bruit sans effacer les contours. Cela devrait être fait dans l'étape de prétraitement. De plus, le calcul du vecteur gradient implique l'utilisation des caractéristiques obtenues à partir d'un ensemble de pixels appartenant à un voisinage local. Ainsi, une erreur survenant au niveau d'un pixel donné peut être récupérée en prenant en compte les caractéristiques des pixels voisins.

Selon notre adaptation, les étapes de Canny pour les images de profondeur ne sont modifiées que pour la méthode de calcul du vecteur gradient et pour synthétiser les caractéristiques adaptées qui doivent être utilisées. Les autres étapes restent inchangées (Figure 4.3). En effet, ces dernières étapes ne dépendent pas de la façon où ces fonctionnalités sont utilisées.

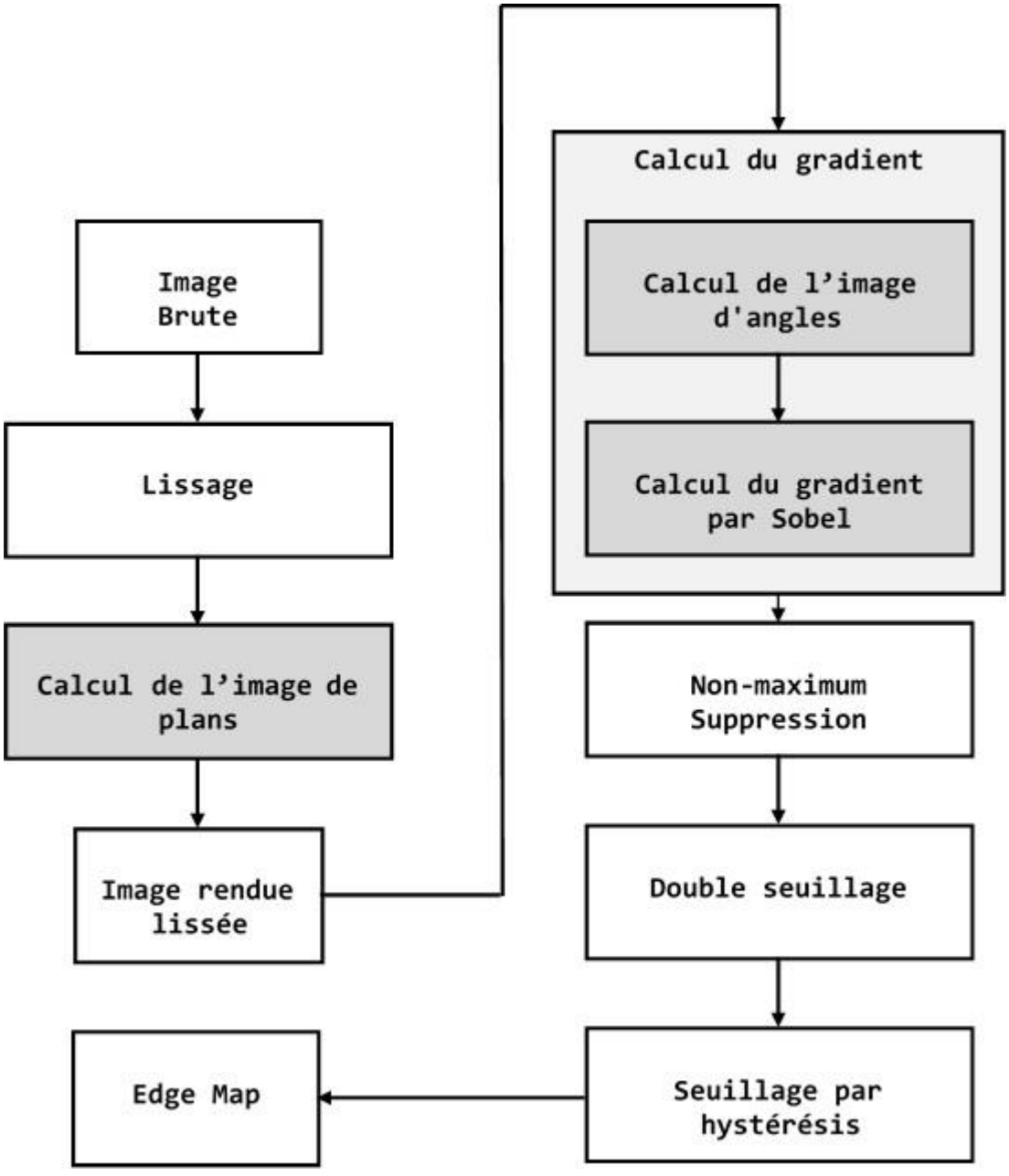


Figure 4.3 - Organigramme de l'ensemble du détecteur proposé.

Le calcul de l'image lissée est déplacé en dehors de l'algorithme de Canny, de sorte que l'image basée surface peut être calculée. Les étapes en gris sont celles où l'adaptation est effectuée.

Dans la partie suivante, nous présenterons notre nouvelle méthode proposée pour le calcul du gradient. Cette méthode a l'avantage d'être rapide et de rester bien adaptée à la particularité des images de profondeur qui a été introduite plus haut.

4.4 Détection de contours basée angle

Notre méthode proposée consiste à calculer un vecteur gradient, qui mesure le niveau de variation de l'orientation de la surface. Ainsi, l'estimation du vecteur gradient en un pixel donné (x, y), nécessite le calcul d'équations de plans selon les directions définies par les pixels voisins. La norme et la direction du gradient sont calculées en utilisant l'angle entre les vecteurs normaux obtenus aux pixels du voisinage. Cependant, pour un pixel donné, on calcule les équations des plans définis selon les huit sous-ensembles possibles de 4 pixels dans le voisinage, comme le montre la Figure 4.4.

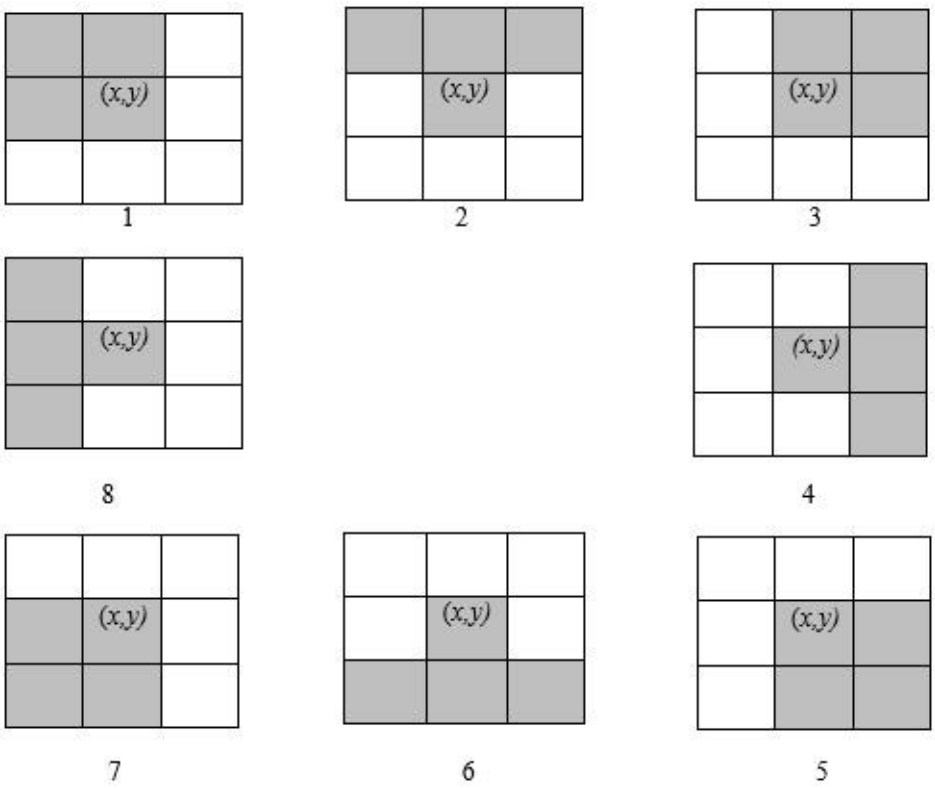


Figure 4.4 - Sous-ensembles de pixels utilisés pour ajuster les équations du plan.

Dans la figure ci-dessus, pour ajuster les équations du plan pour un pixel (x, y). Toutes les combinaisons du pixel central avec ses trois voisins sont considérées.

Pour faire l'approximation de la norme du gradient qui permet de tester si un pixel donné peut être un pixel de contour ou non, nous calculons les équations des plans dans le voisinage. Pour chaque plan, nous utilisons le pixel central et trois de ses voisins :

Étant donné un pixel (x, y), on utilise l'ensemble des pixels voisins $\{(x + \Delta x, y + \Delta y)\ ;\ \Delta x, \Delta y = -1..1\}$ pour définir huit plans. Le tableau 1 ci-dessous présente la liste des pixels impliqués dans le calcul de chaque équation.

Tableau 4.1 - Ensembles de pixels impliqués dans le calcul des huit équations des plans

.

1	$(x, y)\ ;\ (x-1, y)\ ;\ (x-1, y-1)\ ;\ (x, y-1)$
2	$(x, y)\ ;\ (x-1, y-1)\ ;\ (x, y-1)\ ;\ (x+1, y-1)$
3	$(x, y)\ ;\ (x, y-1)\ ;\ (x+1, y-1)\ ;\ (x+1, y)$
4	$(x, y)\ ;\ (x+1, y-1)\ ;\ (x+1, y)\ ;\ (x+1, y+1)$
5	$(x, y)\ ;\ (x+1, y)\ ;\ (x+1, y+1)\ ;\ (x, y+1)$
6	$(x, y)\ ;\ (x+1, y+1)\ ;\ (x, y+1)\ ;\ (x-1, y+1)$
7	$(x, y)\ ;\ (x, y+1)\ ;\ (x-1, y+1)\ ;\ (x-1, y)$
8	$(x, y)\ ;\ (x-1, y+1)\ ;\ (x-1, y)\ ;\ (x-1, y-1)$

En considérant l'ensemble S des pixels concernés, l'équation du plan $a\,x + b\,y + c\,z + d = 0$ est calculée par multi-régression linéaire. Comme toutes les surfaces sont visibles et orientées vers l'observateur, le paramètre *c* est nécessairement non nul. Par conséquent, l'équation du plan peut être exprimée par $z = \alpha\,x + \beta\,y + \gamma$, où $\alpha, \beta\ et\ \gamma$ sont obtenus après minimisation de la fonction objective $\Phi(\alpha, \beta, \gamma)$ qui exprime les moindres carrés comme suit [100] :

$$\Phi(\alpha,\beta,\gamma) = \sum_{p \in S} (\alpha x_p + \beta y_p + \gamma - \mathrm{I}\,(x_p, y_p))^{\,2} \qquad (4.5)$$

Dans l'étape suivante, on calcule l'angle $\phi_{x,y}$ formé par les deux vecteurs normaux pour chaque paire de pixels $\{(x, y)\ ;\ (x + \Delta x, y + \Delta y)\}$. Ensuite, nous calculons le vecteur gradient en introduisant un opérateur de Sobel modifié, comme suit :

D'après l'équation du plan $z = \alpha\, x + \beta\, y + \gamma$, le vecteur normal est $\vec{V}_1(\alpha, \beta, -1)$. L'angle $\phi_{x,y}$ défini entre le vecteur normal et le plan vertical (XY), exprimé par le vecteur $\vec{V}_2(0,0,1)$, est calculé comme suit :

$$\phi_{x,y} = Acos\left(\frac{\vec{V}_1 \,.\, \vec{V}_2}{\left|\left|\vec{V}_1\right|\right| \mathrm{X} \left|\left|\vec{V}_2\right|\right|}\right) \tag{4.6}$$

L'ensemble des angles $\{\phi_{x,y}\}$ formés par les vecteurs normaux 3D aux surfaces et le plan vertical (XY) sont pris en compte pour calculer le vecteur gradient, au lieu des valeurs de pixels $I(x, y)$ (Figure 4.5).

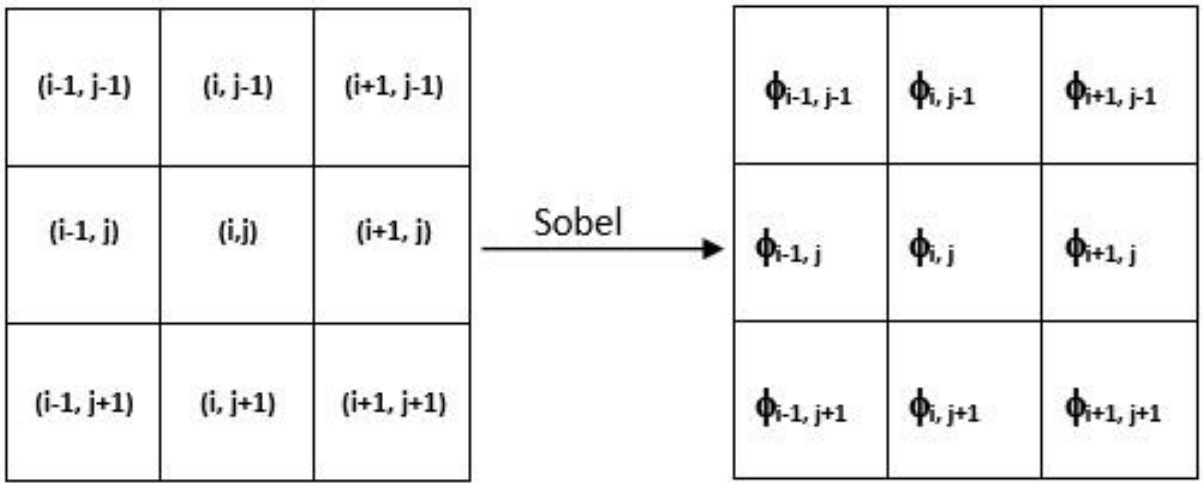


Figure 4.5 - Nouvelles fonctionnalités pour le calcul du gradient à l'aide d'un détecteur Sobel modifié basé angle. Les raw ranges (Valeurs brutes) sont transformées en angles.

Les nouvelles composantes du vecteur gradient (G_x^{ϕ}, G_y^{ϕ}) sont obtenues par les opérateurs horizontaux et verticaux de Sobel. L'angle $\theta = Atan\ (\frac{G_y^{\phi}}{G_x^{\phi}})$ définit la direction du vecteur gradient.

Les normes et les angles obtenus du vecteur gradient, ainsi calculés à chaque pixel de l'image de profondeur, sont utilisés comme entrées pour les étapes suivantes de l'algorithme de Canny, qui restent inchangées pour notre détecteur proposé.

4.5 Evaluation du détecteur proposé

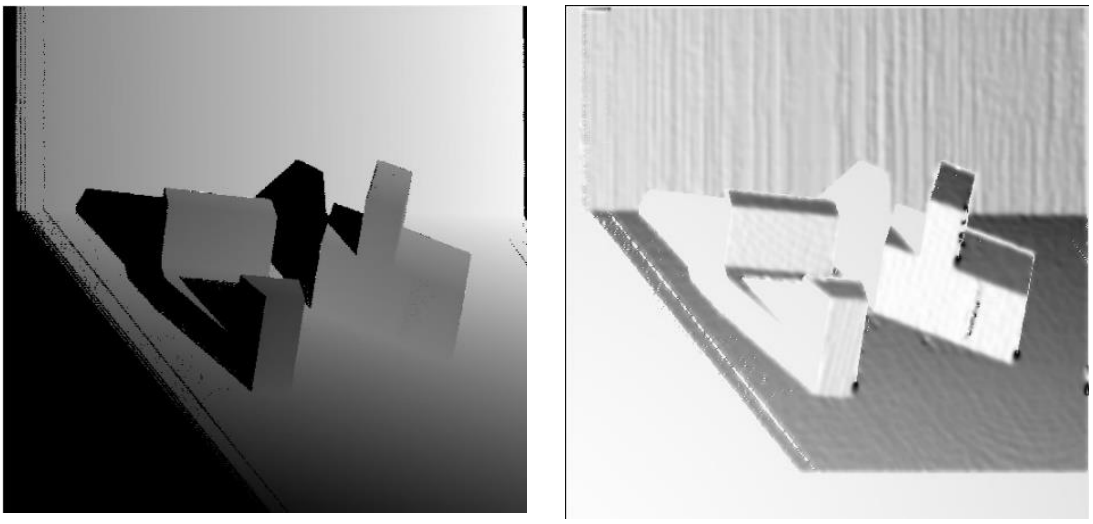
Pour l'évaluation de notre méthode, nous avons utilisé la base de données ABW dédiée à la segmentation d'images de profondeur. Comme pour le détecteur Canny original, un

lissage gaussien est effectué afin de réduire le bruit dans l'image. Après plusieurs tests en faisant varier le paramètre σ du filtre gaussien dans l'intervalle [0,1... 1,0], nous avons fixé σ à 0,8, pour lequel les contours des images testées ont été correctement détectés. L'utilisation du lissage gaussien au lieu d'autres techniques de filtrage du bruit est due à la nature du bruit dans les images de profondeur. En effet, dans de telles images, le bruit est principalement dû à des distorsions dans les données d'image plutôt qu'à des valeurs aberrantes impulsives, où les filtres médians pourraient être bien adaptés.

4.5.1 Evaluation Qualitative

Comme dans nos travaux antérieurs, publiés dans [95], nous commençons par introduire quelques échantillons d'images de profondeur de la base de données ABW, avec leurs résultats de détection de contours en utilisant le détecteur Canny original et celui proposé. Une telle présentation visuelle des résultats permet au lecteur d'avoir une idée de la qualité visuelle avancée de la détection des contours en utilisant notre détecteur proposé. La Figure 4.6.a montre une image de profondeur de la base de données ABW. L'image est affichée en fonction de ses données de profondeur brutes. Les régions noires sont les ombres où le rayon laser n'a pas atteint la surface de l'objet. A chaque pixel, la profondeur est représentée par un niveau de gris allant de 0 à 255. Sur la Figure 4.6.b, qui représente une image rendue des données brutes, à l'aide d'un algorithme de rendu simple avec simulation d'une source lumineuse, on peut remarquer un niveau de bruit élevé sous forme de distorsions des surfaces planes, dues à des mesures de distance moins précises.

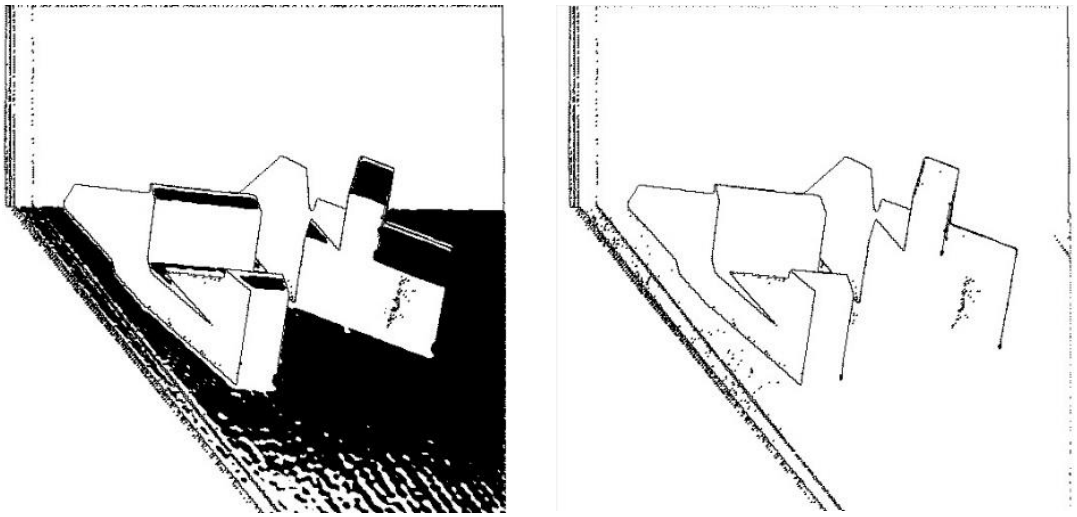
Le résultat de l'application du détecteur Canny original, en utilisant les données d'image brutes, est illustré à la Figure 4.7.a pour le seuil haut et à la Figure 4.7.b pour le seuil bas. La première image (Figure 4.7.a) montre clairement une sous-détection des contours, où l'on peut remarquer que plusieurs contours n'ont pas été détectés, en particulier ceux qui forment les frontières entre les surfaces d'un même objet. Pour de tels contours, il existe un continuum dans les données de profondeur, de sorte que le vecteur de gradient sera continu à ces points. En revanche, on remarque clairement une sur-détection des contours dans la seconde image (Figure 4.7.b).



(a) (b)

Figure 4.6 - Un exemple d'image de la base de données ABW, (a) image de profondeur, (b) image rendue montrant la nature du bruit dans de telles images.

En effet, de nombreux pixels à l'intérieur des différentes surfaces planes ont été détectés comme pixels de contours, alors qu'ils ne le sont pas. Sinon, la plupart des contours définissant les frontières entre les surfaces ont été détectées. Cependant, les pixels de certaines surfaces fortement inclinées ont tous été détectés comme pixels de contours. Les dernières étapes de l'algorithme de Canny ; à savoir la non maximum suppression et le seuillage par hystérésis, ne peuvent pas améliorer de tels résultats dans la première image et génèrent beaucoup de valeurs aberrantes dans la seconde.



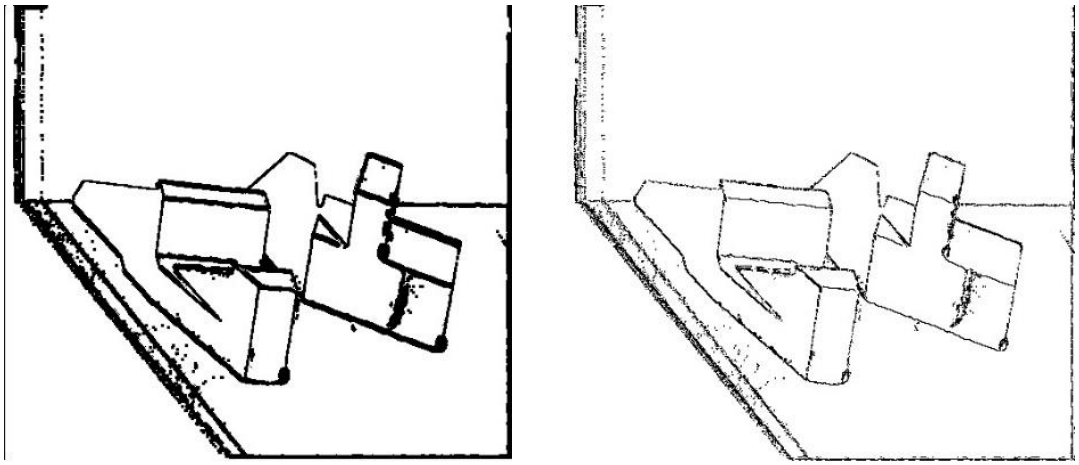
(a) (b)

Figure 4.7 - Détection des contours dans l'image abw.test.3 avec le détecteur Canny original qui utilise les données brutes : (a) Contours détectés avec un seuil de gradient supérieur (0,2), (b) Contours détectés avec un seuil de gradient inférieur (0,12). Une sous-détection est constatée en (a), alors qu'une sur-détection est constatée en (b).

Après avoir appliqué notre détecteur proposé, nous avons pu obtenir les résultats présentés sur la Figure 4.8. Les résultats de détection, obtenus avant les étapes d'élimination des non-maxima et de seuillage par hystérésis, sont présentés sur la Figure 4.8.a. On peut remarquer, contrairement au détecteur Canny original, que notre détecteur adapté qui utilise un gradient modifié, basé sur la variation des angles normaux, permet de produire une carte des contours, utilisable pour les étapes ultérieures de Canny où un post-traitement adéquat peut être effectué.

Le résultat final de la détection est illustré à la Figure 4.8.b. Ce résultat est obtenu après avoir appliqué la suppression des non-maxima et le seuillage par hystérésis sur le gradient d'image de la Figure 4.8.a. La meilleure valeur pour le seuil inférieur selon le détecteur proposé était de 0,12 radian (6,86°) et celle du seuil supérieur était de 0,20 radian (11,46°). De telles valeurs ont été obtenues en faisant varier les deux seuils dans l'intervalle [0,05... 0,25], par pas de 0,05. Nous avons considéré les valeurs pour lesquelles le résultat du détecteur était le meilleur pour l'ensemble des images d'entraînement ABW. Nous concluons que les résultats visuels de la détection des contours étaient satisfaisants pour l'ensemble des images de test. On peut aussi dire que les contours ont été correctement

détectés, y compris ceux appartenant aux frontières intra-objet. Ces derniers sont difficiles à détecter comme cela a été rapporté dans tous les travaux ayant traité de la segmentation des images de profondeur.



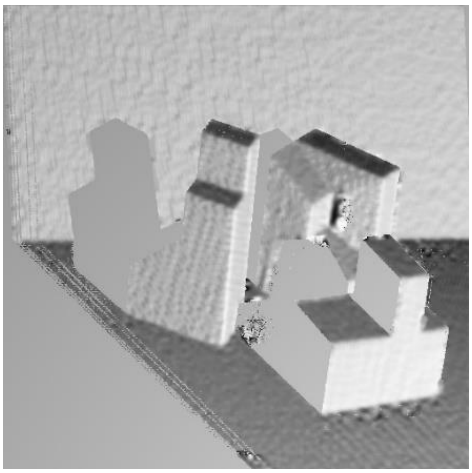
(a) (b)

Figure 4.8 - Détection des contours dans l'image abw.test.3 avec le détecteur proposé : (a) Contours détectés avant suppression des non-maxima et seuillage par hystérésis, (b) Contours détectés après suppression des non-maxima et seuillage par hystérésis (résultat final).

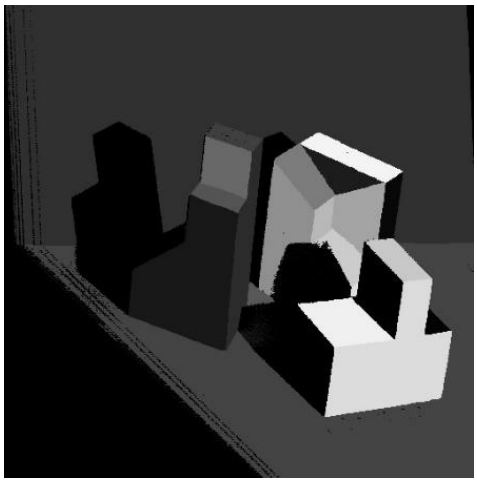
4.5.2 Evaluation Quantitative

Pour autant que nous le sachions, il n'existe pas de méthode native pour la détection des contours dans les images de profondeur ; toutes les méthodes proposées que nous avons trouvées dans la littérature sont basées région ou basées sur l'apprentissage automatique [101]. De plus, les images de profondeur pures telles que celles d'ABW, ou celles d'autres ensembles de données RGBD qui fournissent une modalité de profondeur, telles que OSD (Object Segmentation Database) [102], ne fournissent pas la réalité terrain (ground-truth) de la détection de contours. Par conséquent, il sera difficile d'évaluer quantitativement les méthodes basées sur la détection de contours pour les images de profondeur.

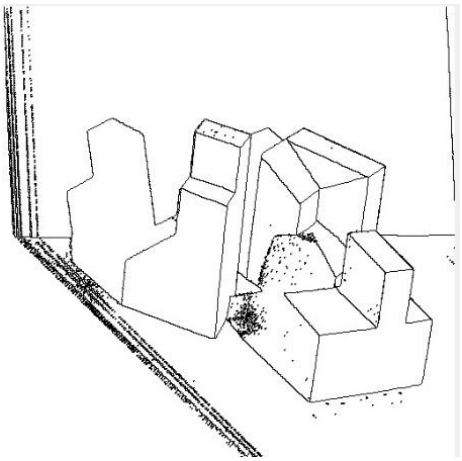
Néanmoins, nous avons pu générer la réalité terrain des cartes de contours à partir de la réalité terrain de la segmentation basée région. La Figure 3.9 montre un exemple d'image de l'ensemble de données ABW ; à savoir abw.test.8, où nous pouvons voir sur la Figure 4.9.b et la Figure 4.9.c la réalité terrain des régions et les contours que nous en avons générées.



(a)



(b)



©

Figure 4.9 - (a) Un exemple d'image de l'ensemble de données ABW, (b) Ground Truth des régions, (c) Ground Truth générée des contours.

Après avoir extrait la carte des contours Ground truth (réalité terrain) de la segmentation basée régions, fournie par le jeu de données ABW, nous considérons l'indice Dice comme une métrique pour évaluer la qualité de la détection des contours et pour comparer les résultats obtenus par le détecteur proposé avec ceux du détecteur Canny original.

L'indice Dice [103] permet d'exprimer l'écart entre une carte de contours, produite par notre détecteur et la carte de contours de la réalité terrain correspondante. Cet indice est calculé sur la base des éléments suivants :

True Positives (TP) : Nombre de pixels de contours correctement détectés.

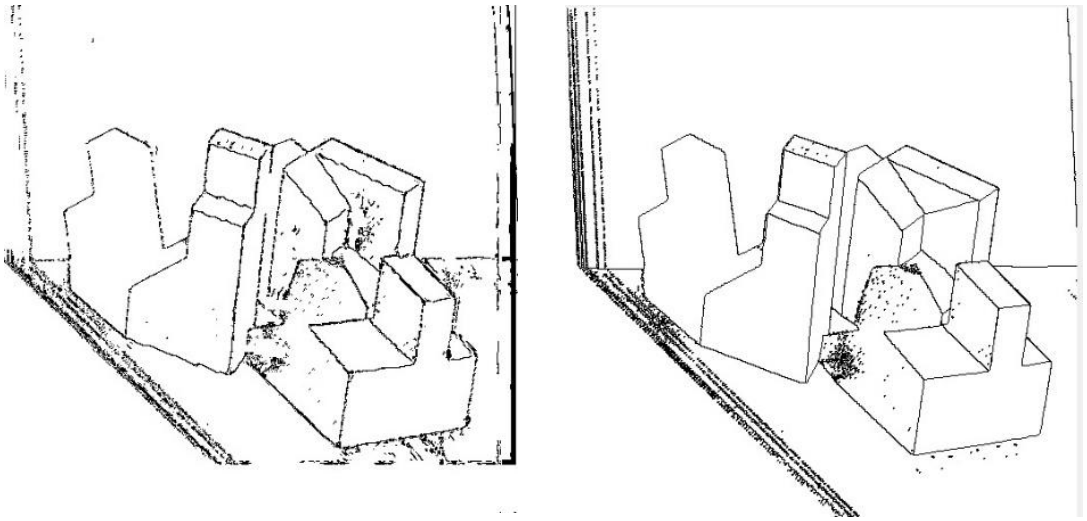
False Positives (FP) : nombre de pixels détectés à tort comme pixels de contours (absents dans la réalité terrain).

False Negatives (FN) : Nombre de vrais pixels de contours qui n'ont pas été détectés.

En fonction de TP, FP et FN, l'indice Dice s'exprime comme suit :

$$\kappa = \frac{2 \times \mathrm{TP}}{2 \times \mathrm{TP} + \mathrm{FP} + \mathrm{FN}} \tag{4.7}$$

L'indice de Dice pour la carte de contours produite à la Figure 4.10.a concernant la carte de contours de la réalité terrain de la Figure 4.10.b est de 0,8642, où les trois paramètres étaient les suivants : TP = 12353 pixels, FP = 3214 et FN = 666.



(a) (b)

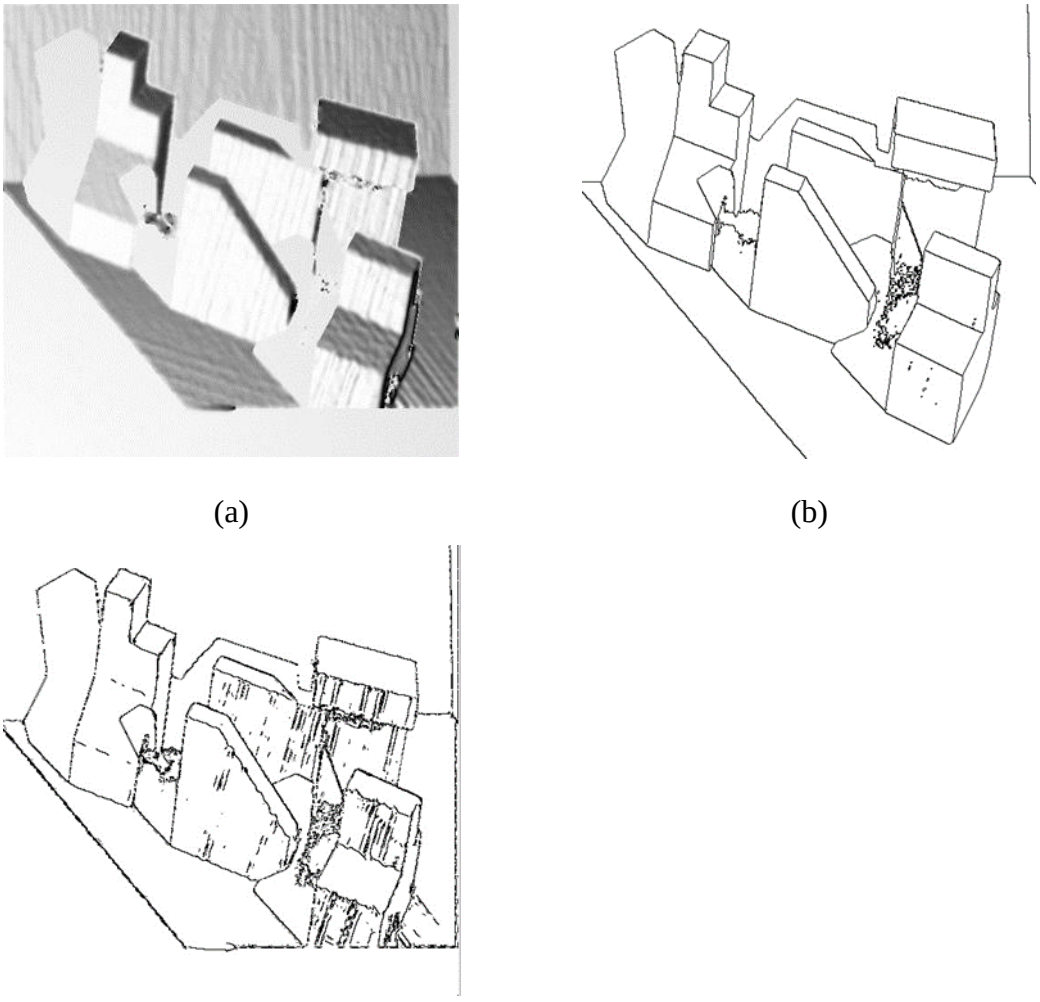
Figure 4.10 - Comparaison entre les contours détectés et la réalité terrain : (a) Contours détectés par le détecteur proposé, (b) Contours réalité terrain.

Pour l'ensemble des 30 images de la base de données ABW, les résultats selon l'indice Dice sont présentés dans le tableau 4.2.

Tableau 4.2 - Moyenne, max. et min. Indice Dice pour l'ensemble de données ABW.

Mean Dice	Standard-deviation	Max. Dice	Min. Dice
0.8238	0.0335	0.8642 Obtenu avec abw.test.8	0.7629 Obtenu avec abw.test.0

La Figure 4.11 montre les résultats de la détection visuelle pour l'image ayant le score le plus bas ; à savoir, abw.test.0.

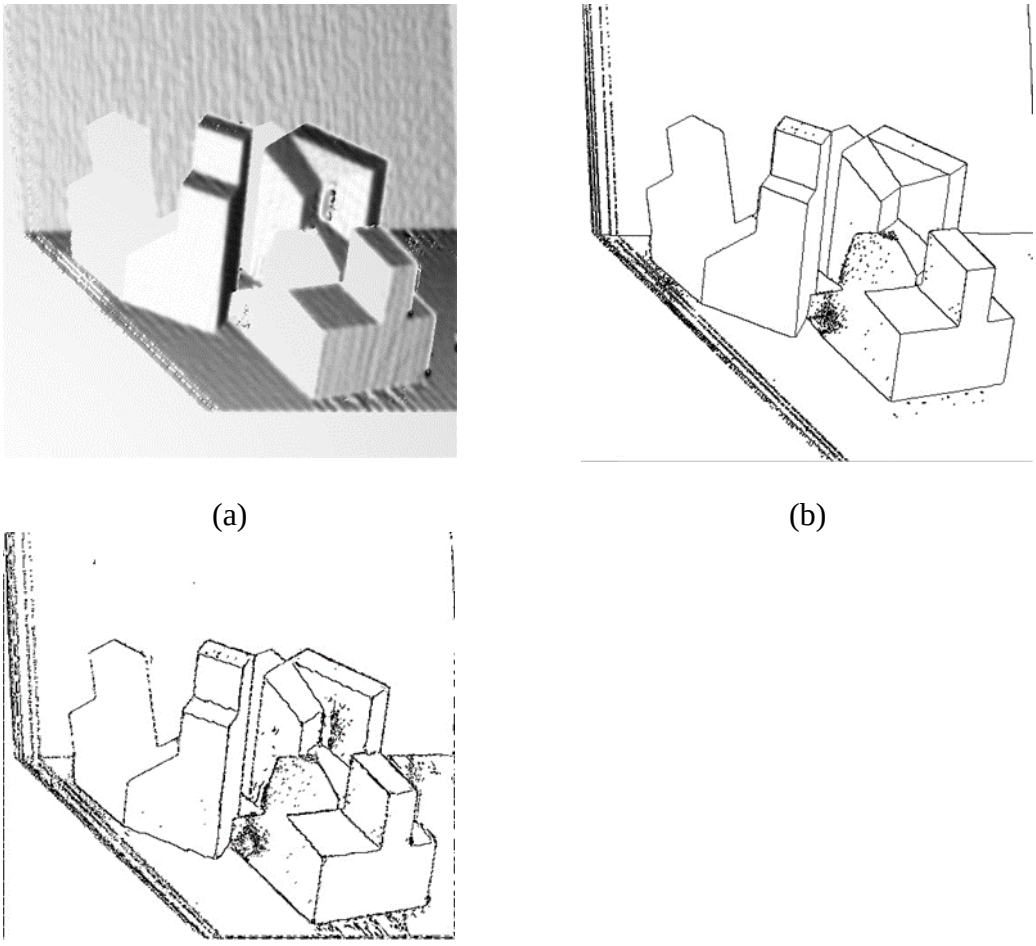


(a) (b)

(c)

Figure 4.11 - Résultats de la détection de contours correspondant à l'indice Dice le plus bas (abw.test.0) (a) rendu lissé abw.test.0, (b) contours détectés en réalité terrain, (c) Contours détectés par le détecteur proposé,

La Figure 4.12 montre les résultats de la détection visuelle pour l'image ayant le score le plus élevé ; à savoir, abw.test.8.



(a) (b)

(c)

Figure 4.12 - Résultats de la détection de contours correspondant à l'indice le plus élevé (abw.text.8), (a) Rendu lissé abw.test.8, (b) Contours détectés en réalité terrain et (c) Contours par le détecteur proposé.

D'après les résultats présentés dans le Tableau 4.3, nous pouvons affirmer que le détecteur proposé pour les images de profondeur a considérablement amélioré la détection des contours dans les images de profondeur. Pour la valeur moyenne de l'indice Dice, l'amélioration est d'environ 18%, ce qui sera très utile pour le post-traitement des images de profondeur. De plus, il est plus stable, étant donné que sa déviation standard (écart type) est nettement inférieure à celle du détecteur Canny original.

Tableau 4.3 - Comparaison des résultats de détection selon l'indice Dice, impliquant le détecteur proposé et le détecteur Canny d'origine.

	Mean Dice	Standard-Deviation	Max. Dice	Min. Dice
Détecteur proposé sur l'image basée surface	0.8238	0.0335	0.8642	0.7629
Canny original sur l'image brute	0.6972	0.0539	0.7813	0.5998

4.6 Conclusion

Dans ce chapitre, nous avons introduit un détecteur de contours dédié aux images de profondeur, et ce, en proposant une adaptation du filtre de Canny, et en considérant les spécificités géométriques des données de profondeur. En effet, contrairement au détecteur classique qui se base sur la valeur de l'intensité dans les images visuelles, nous avons considéré l'angle du vecteur normal à la surface comme donnée élémentaire pour le calcul du gradient. Ce choix nous a permis de capturer et de représenter les variations du relief des surfaces figurant dans une image de profondeur. L'évaluation qualitative et quantitative du détecteur proposé a montré sa haute performance comparée avec celle du détecteur classique, quand ce dernier est appliqué aux images de profondeur. La détection opérée avec ce nouveau détecteur est alors sujette à une régularisation markovienne et bayésienne, dont les détails seront présentés dans le chapitre suivant.

Chapitre 5

Amélioration basée régularisation des contours dans les images de profondeur

5.1 Introduction

La segmentation d'images est une étape fondamentale en vision par ordinateur, avec des applications variées allant de la reconnaissance d'objets à la reconstruction 3D, en passant par l'analyse médicale et la robotique. Parmi les différentes étapes de la segmentation, la détection de contours joue un rôle très important, car elle permet d'identifier les limites des objets dans une image. Cependant, les contours détectés par des algorithmes classiques, tels que le filtre de Canny, sont souvent bruités, incomplets ou irréguliers, en particulier dans les images de profondeur. Ces images, qui représentent la distance des objets par rapport à la caméra, sont fréquemment affectées par des artefacts, des variations d'intensité, ou des erreurs de mesure, ce qui complique la détection précise des contours.

Dans notre travail précédent [104], nous avons proposé une adaptation du filtre de Canny pour les images de profondeur, en tenant compte des spécificités de ces images, telles que les variations de profondeur et les discontinuités. Cette adaptation a permis d'améliorer la détection des contours dans les images de profondeur, mais les résultats restent souvent affectés par du bruit et des irrégularités. Par conséquent, une étape de régularisation de contours est nécessaire pour affiner les contours détectés et les rendre plus conformes à la structure réelle des objets dans l'image.

Pour pallier ces limitations, des techniques de régularisation de contours sont couramment utilisées. Ces techniques visent à améliorer la qualité des contours détectés en les rendant plus réguliers et plus conformes à la structure réelle des objets dans l'image. Dans le cas des images de profondeur, où les objets sont souvent de nature polyédrique (composés de surfaces planes et d'arêtes droites), un a priori important peut être exploité : la droiture des segments de contours. En effet, dans ces images, les contours sont supposés être localement droits, ce qui constitue une propriété importante pour guider la régularisation.

En considérant cet a priori et en travaillant sur les résultats obtenus par le détecteur de Canny adapté aux images de profondeur, nous proposons dans ce chapitre une méthode des champs aléatoires de Markov (MRF : Markov Random Field), basée sur la minimisation d'énergie pour la régularisation des contours obtenus. Cette approche repose sur une modélisation probabiliste qui permet de capturer les dépendances spatiales entre les pixels, tout en intégrant l'a priori de linéarité des contours. En formulant le problème de régularisation comme une minimisation d'énergie, nous cherchons à trouver un compromis optimal entre la fidélité aux données observées et le respect de l'a priori.

L'objectif de ce chapitre est de présenter une méthode efficace pour améliorer les résultats de segmentation dans les images de profondeur, en exploitant l'a priori de linéarité des contours. Nous détaillons ici la modélisation des MRF, la formulation du problème de minimisation d'énergie, ainsi que les algorithmes proposés pour résoudre ce problème. Enfin, nous validons notre approche à travers des expérimentations sur des images de profondeur issues de la base de données ABW, en comparant les résultats avant et après régularisation.

Ce chapitre est organisé comme suit :

- **Section 2** : Contexte et motivation – Nous expliquons pourquoi la régularisation de contours est importante et quelles sont les applications potentielles.
- **Section 3** : Problématique – Nous détaillons les défis spécifiques liés à la régularisation des contours dans les images de profondeur.
- **Section 4** : Contributions – Nous présentons les solutions proposées et en quoi elles sont innovantes.
- **Section 5** : Méthodologie – Nous décrivons en détail la méthode de régularisation basée sur les MRF et la minimisation d'énergie.
- **Section 6** : Expérimentations et résultats – Nous validons notre méthode sur des images de profondeur et discutons des résultats obtenus.
- **Section 7** : Conclusion – Nous résumons les contributions et ouvrons des perspectives pour des travaux futurs.

5.2 Contexte et motivation

La détection et la régularisation de contours sont des étapes importantes dans de nombreuses applications de vision par ordinateur. Ces étapes jouent un rôle important dans

des domaines variés tels que la reconstruction 3D, la navigation autonome, la robotique, et l'analyse médicale. Dans ces applications, la précision des contours est essentielle pour assurer la qualité des résultats finaux. Par exemple, dans les systèmes de conduite autonome, des contours précis sont nécessaires pour détecter les bords des routes, les obstacles et les autres véhicules. De même, en reconstruction 3D, des contours régularisés permettent de générer des modèles 3D plus précis et plus réalistes.

5.2.1 Importance des contours dans les images de profondeur

Les images de profondeur, qui représentent la distance des objets par rapport à la caméra, sont de plus en plus utilisées dans des applications telles que la réalité augmentée, la robotique, et la surveillance. Contrairement aux images RGB classiques, les images de profondeur fournissent des informations géométriques directes sur la scène, ce qui les rend particulièrement utiles pour des tâches telles que la détection d'obstacles, la navigation, et la reconstruction 3D. Cependant, ces images présentent également des défis spécifiques, tels que des artefacts, des variations d'intensité, et des erreurs de mesure, qui peuvent compliquer la détection précise des contours.

Dans notre travail précédent [104], nous avons proposé une adaptation du filtre de Canny pour les images de profondeur, en tenant compte des spécificités de ces images, telles que les variations de profondeur et les discontinuités. Cette adaptation a permis d'améliorer la détection des contours, mais les résultats restent souvent affectés par du bruit et des irrégularités. Par conséquent, une étape de régularisation de contours est nécessaire pour affiner les contours détectés et les rendre plus conformes à la structure réelle des objets dans l'image.

5.2.2 Applications de la régularisation de contours

La régularisation de contours est particulièrement importante dans les applications où la précision des contours est essentielle. On peut citer quelques exemples d'applications où cette technique joue un rôle important :

- **Reconstruction 3D** : Dans la reconstruction 3D à partir d'images de profondeur, des contours précis sont essentiels pour générer des modèles 3D fidèles à la réalité. Une régularisation efficace permet de réduire les artefacts et d'améliorer la qualité des modèles [105].

- **Navigation autonome** : Les systèmes de conduite autonome reposent sur la détection précise des contours pour identifier les bords des routes, les obstacles, et les autres véhicules. Une régularisation des contours permet d'améliorer la fiabilité de ces systèmes [106].
- **Robotique** : En robotique, la détection de contours est utilisée pour la localisation, la cartographie, et la manipulation d'objets. Une régularisation des contours permet d'améliorer la précision de ces tâches [107].
- **Analyse médicale** : Dans l'imagerie médicale, la détection de contours est utilisée pour segmenter des structures anatomiques, comme les os ou les organes. Une régularisation des contours permet d'obtenir des résultats plus précis et plus fiables.

5.2.3 Défis actuels dans la régularisation de contours

Malgré son importance, la régularisation de contours dans les images de profondeur pose plusieurs défis majeurs :

1. **Définition d'un a priori adapté** : Pour régulariser les contours, il est nécessaire de définir un a priori sur la structure des objets dans l'image. Dans le cas des objets polyédriques, cet a priori est la linéarité des contours, mais pour des objets plus complexes (comme des objets organiques ou courbés), un a priori différent serait nécessaire [108].
2. **Compromis entre fidélité aux données et respect de l'a priori** : Une régularisation trop forte peut effacer des détails importants, tandis qu'une régularisation trop faible peut laisser subsister du bruit ou des artefacts. Trouver un compromis optimal entre la fidélité aux données observées et le respect de l'a priori est un défi complexe. Une solution est d'utiliser des méthodes de régularisation basées sur la minimisation d'énergie pour résoudre ce problème, mais ces méthodes peuvent être coûteuses en termes de calcul [109].
3. **Efficacité computationnelle** : Les méthodes de régularisation doivent être suffisamment efficaces pour être applicables à des images de grande taille et en temps réel, ce qui est essentiel pour des applications pratiques comme la robotique ou la conduite autonome [110].

5.2.4 Motivation pour une nouvelle approche

Compte tenu de ces défis, il est nécessaire de développer des méthodes de régularisation de contours qui soient à la fois précises et efficaces. C'est dans ce contexte que nous proposons une nouvelle approche basée sur les champs aléatoires de Markov (MRF) et la minimisation d'énergie. Cette approche permet de capturer les dépendances spatiales entre les pixels tout en intégrant un a priori de linéarité des contours, ce qui la rend particulièrement adaptée aux images de profondeur d'objets polyédriques.

Après avoir présenté le contexte et la motivation de notre travail, nous détaillons dans la section suivante les défis spécifiques liés à la régularisation des contours dans les images de profondeur, ainsi que les problématiques que notre méthode cherche à résoudre.

5.3 Problématique

La régularisation de contours dans les images de profondeur est un problème complexe qui pose plusieurs défis majeurs. Ces défis sont liés à la nature des images de profondeur, à la définition d'un a priori adapté, et à la nécessité de trouver un compromis entre fidélité aux données et respect de l'a priori. Dans cette section, nous détaillons ces défis et expliquons pourquoi ils rendent la régularisation de contours dans les images de profondeur particulièrement difficile.

5.3.1 Nature des images de profondeur

Les **images de profondeur** présentent des caractéristiques spécifiques qui les distinguent des images RGB classiques. Ces caractéristiques incluent :

- **Artefacts et bruit** : Les images de profondeur sont souvent affectées par des artefacts et du bruit, dus à des erreurs de mesure ou à des limitations des capteurs de profondeur. Par exemple, les capteurs Kinect, largement utilisés pour acquérir des images de profondeur, sont connus pour produire des artefacts tels que des trous ou des distorsions dans les zones de faible réflexivité [111].
- **Discontinuités de profondeur** : Les images de profondeur présentent souvent des discontinuités abruptes, correspondant aux bords des objets. Ces discontinuités peuvent être difficiles à détecter avec précision, en particulier dans les zones où les variations de profondeur sont faibles ou les objets se chevauchent.
- **Variations d'intensité** : Contrairement aux images RGB, où les contours sont souvent associés à des variations de couleur ou de texture, les contours dans les

images de profondeur sont principalement définis par des variations de distance. Cela rend la détection de contours plus sensible au bruit et aux erreurs de mesure.

Ces caractéristiques rendent la détection et la régularisation de contours dans les images de profondeur particulièrement difficiles, nécessitant des approches spécifiques pour gérer le bruit et les artefacts tout en préservant les discontinuités importantes.

5.3.2 Définition d'un a priori adapté

Un des principaux défis de la régularisation de contours est la définition d'un a priori adapté à la nature des objets dans l'image. Dans le cas des images de profondeur contenant des objets polyédriques, l'a priori retenu est la linéarité des segments de contours. Cependant, pour des objets plus complexes (comme des objets organiques ou courbés), un a priori différent serait nécessaire.

- **Linéarité des contours** : Pour les objets polyédriques, les contours sont supposés être localement droits. Cet a priori est utile pour guider la régularisation, mais il peut ne pas être applicable à des objets présentant des courbures ou des formes irrégulières. Il est difficile de définir un a priori adapté pour des objets de formes variées, en particulier dans des scènes complexes où plusieurs types d'objets coexistent [108].
- **A priori pour des objets complexes** : Pour des objets non polyédriques, comme des objets organiques ou des surfaces courbées, l'a priori de linéarité n'est plus valable. Dans ce cas, il est nécessaire de définir un a priori plus flexible, capable de capturer des contours courbés ou irrégulier. Une solution est d'utiliser des méthodes de régularisation basées sur des modèles de contours flexibles, mais ces méthodes peuvent être coûteuses en termes de calcul et difficiles à paramétrer [109].

5.3.3 Compromis entre fidélité aux données et respect de l'a priori

Un autre défi majeur est de trouver un compromis optimal entre la fidélité aux données observées (c'est-à-dire la proximité des contours régularisés aux contours détectés) et le respect de l'a priori (c'est-à-dire la linéarité des contours). Ce compromis est difficile à atteindre pour plusieurs raisons :

- **Régularisation trop forte** : Une régularisation trop forte peut effacer des détails importants, comme des contours fins ou des structures complexes. Par exemple,

dans les images de profondeur, une régularisation excessive peut conduire à la perte de contours correspondant à des petits objets ou à des détails fins.

- **Régularisation trop faible** : À l'inverse, une régularisation trop faible peut laisser subsister du bruit ou des artefacts, ce qui réduit la qualité des contours régularisés. Par exemple, dans les zones où le bruit est important, une régularisation insuffisante peut conduire à des contours irréguliers ou fragmentés.
- **Optimisation de l'énergie** : Trouver un compromis optimal nécessite de formuler le problème de régularisation comme une minimisation d'énergie, où l'énergie totale est composée d'une énergie externe (fidélité aux données) et d'une énergie interne (respect de l'a priori). Cependant, la minimisation de cette énergie peut être complexe, en particulier pour des images de grande taille ou des scènes complexes [110].

5.3.4 Efficacité computationnelle

Enfin, les méthodes de régularisation doivent être suffisamment efficaces pour être applicables à des images de grande taille et en temps réel. Ceci est particulièrement important pour des applications pratiques comme la robotique ou la conduite autonome, où les algorithmes doivent fonctionner en temps réel sur des flux vidéo ou des images de haute résolution.

- **Complexité algorithmique** : Les méthodes de régularisation basées sur les champs aléatoires de Markov (MRF) ou la minimisation d'énergie peuvent être coûteuses en termes de calcul, en particulier pour des images de grande taille ou des scènes complexes. Le développement des algorithmes efficaces pour la minimisation d'énergie dans le cadre des MRF est nécessaire, en particulier pour des applications en temps réel [112].
- **Optimisation en temps réel** : Pour des applications en temps réel, il est essentiel de développer des algorithmes de régularisation qui soient à la fois précis et rapides [105].

Après avoir détaillé les défis spécifiques liés à la régularisation des contours dans les images de profondeur, nous présentons dans la section suivante les contributions de notre travail, en expliquant comment notre méthode propose de résoudre ces problèmes.

5.4 Contributions

Dans ce chapitre, nous proposons une méthode de régularisation de contours basée sur les champs aléatoires de Markov (MRF) et la minimisation d'énergie, spécifiquement adaptée aux images de profondeur contenant des objets polyédriques. Nos contributions principales sont les suivantes :

5.4.1 Modélisation de l'a priori de linéarité

La première contribution de notre travail est la modélisation de l'a priori de linéarité des contours dans les images de profondeur. Contrairement aux approches existantes qui utilisent des a priori génériques ou peu adaptés aux objets polyédriques, nous proposons un modèle spécifique qui exploite la propriété de droiture des segments de contours.

- **Modèle MRF pour la linéarité des contours** : Nous définissons un modèle MRF qui capture les dépendances spatiales entre les pixels de l'image, en intégrant un a priori de linéarité des contours. Ce modèle repose sur une distribution de Gibbs, où l'énergie interne est calculée en fonction de l'alignement des pixels de contour dans des cliques de 3×3 pixels. Cette approche permet de pénaliser les configurations de pixels qui ne respectent pas l'a priori de linéarité, tout en favorisant les configurations où les contours sont alignés.
- **Potentiels de cliques** : Pour chaque clique de 3×3 pixels, nous définissons des potentiels qui mesurent le degré d'alignement des pixels de contour. Ces potentiels sont définis de manière à favoriser les configurations où les contours sont linéaires, tout en pénalisant les configurations où les contours sont courbés ou fragmentés. Par exemple, une clique où les pixels de contour sont parfaitement alignés aura un potentiel de 0, tandis qu'une clique où les pixels de contour sont partiellement alignés aura un potentiel de 0.5, et une clique où les pixels de contour ne sont pas alignés aura un potentiel de 1.

5.4.2 Formulation du problème de régularisation

La deuxième contribution de notre travail est la formulation du problème de régularisation comme une minimisation d'énergie. Cette formulation permet de trouver un compromis optimal entre la fidélité aux données observées (c'est-à-dire les contours détectés par le filtre de Canny) et le respect de l'a priori de linéarité.

- **Énergie externe** : L'énergie externe mesure la similarité entre les contours régularisés et les contours détectés par le filtre de Canny. Elle est calculée en fonction de la correspondance entre les pixels de contour dans les deux images. Une grande ressemblance donne une faible énergie, tandis qu'une faible ressemblance donne une grande énergie.
- **Énergie interne** : L'énergie interne impose la linéarité des contours en fonction de l'a priori. Elle est calculée en fonction des potentiels des cliques, comme décrit précédemment. Plus les contours sont linéaires, plus l'énergie interne est faible.
- **Énergie totale** : L'énergie totale à minimiser est la somme de l'énergie externe et de l'énergie interne. En minimisant cette énergie, nous cherchons à trouver un compromis optimal entre la fidélité aux données et le respect de l'a priori.

5.4.3 Algorithme de minimisation d'énergie

La troisième contribution de notre travail est la proposition d'un algorithme de minimisation d'énergie efficace pour résoudre le problème de régularisation. Cet algorithme est conçu pour être applicable à des images de grande taille et en temps réel, ce qui est essentiel pour des applications pratiques comme la robotique ou la conduite autonome.

- **Minimisation par programmation dynamique** : Nous proposons un algorithme de minimisation basé sur la programmation dynamique, où le problème global est décomposé en sous-problèmes plus simples. Chaque sous-problème correspond à la minimisation de l'énergie d'une clique de 3×3 pixels. En résolvant ces sous-problèmes de manière itérative, nous obtenons une solution globale qui minimise l'énergie totale.
- **Itérations et convergence** : L'algorithme est conçu pour fonctionner de manière itérative, en mettant à jour les contours régularisés à chaque itération jusqu'à convergence. La convergence est atteinte lorsque l'énergie globale ne diminue plus de manière significative. Cette approche permet d'améliorer progressivement la qualité des contours régularisés.

5.4.4 Validation expérimentale

La quatrième contribution de notre travail est la validation expérimentale de notre méthode sur des images de profondeur issues de la base de données ABW. Cette validation permet de démontrer l'efficacité de notre méthode par rapport aux approches existantes.

- **Base de données ABW** : Nous utilisons des images de profondeur de la base de données ABW, qui contient des scènes intérieures et industrielles avec des objets polyédriques. Ces images présentent des défis typiques, tels que du bruit, des artefacts, et des discontinuités de profondeur.
- **Comparaison avec le filtre de Canny** : Nous comparons les résultats de notre méthode avec ceux obtenus par le filtre de Canny adapté aux images de profondeur. Les résultats montrent une amélioration significative de la qualité des contours, avec une meilleure conformité à la structure réelle des objets dans l'image.
- **Évaluation quantitative** : Nous évaluons quantitativement les résultats en utilisant des métriques telles que la précision, le rappel, et le F1-score. Ces métriques permettent de mesurer la qualité des contours régularisés par rapport à une vérité terrain.

Après avoir présenté les contributions de notre travail, nous détaillons dans la section suivante la méthodologie proposée, en expliquant en détail la modélisation des MRF, la formulation du problème de minimisation d'énergie, et les algorithmes utilisés pour résoudre ce problème.

5.5 Méthodologie

Dans cette section, nous détaillons la méthode proposée pour la régularisation des contours dans les images de profondeur. Notre approche repose sur une modélisation probabiliste basée sur les champs aléatoires de Markov (MRF) et une minimisation d'énergie pour trouver les contours idéaux. Nous décrivons ici les étapes importantes de notre méthode, en commençant par la modélisation des MRF, puis en expliquant la formulation du problème de minimisation d'énergie, et enfin en présentant les algorithmes utilisés pour résoudre ce problème.

5.5.1 Modélisation des champs aléatoires de Markov (MRF)

Les champs aléatoires de Markov (MRF) sont un outil puissant pour modéliser les dépendances spatiales entre les pixels d'une image. Dans notre cas, nous utilisons un

modèle MRF pour capturer l'a priori de linéarité des contours dans les images de profondeur.

- **Définition du modèle MRF** : Un MRF est défini sur un graphe où chaque nœud représente un pixel de l'image, et les arêtes représentent les relations de voisinage entre les pixels. Dans notre modèle, nous considérons un voisinage de 3×3 pixels pour chaque clique, ce qui permet de capturer les dépendances locales entre les pixels.
- **Distribution de Gibbs** : Le modèle MRF est associé à une distribution de Gibbs, qui définit la probabilité d'une configuration de pixels en fonction de son énergie. L'énergie d'une configuration est calculée comme la somme des potentiels des cliques, où chaque potentiel mesure le degré d'alignement des pixels de contour dans la clique.
- **Potentiels de cliques** : Pour chaque clique de 3×3 pixels, nous définissons des potentiels qui mesurent le degré d'alignement des pixels de contour. Ces potentiels sont définis comme suit :
    - Potentiel = 0 : Si le centre de la clique est un point de contour et que deux autres pixels de contour sont parfaitement alignés avec lui.
    - Potentiel = 0.5 : Si le centre de la clique est un point de contour et que deux autres pixels de contour sont partiellement alignés avec lui.
    - Potentiel = 1 : Dans tous les autres cas (par exemple, si les pixels de contour ne sont pas alignés).

Cette modélisation permet de favoriser les configurations où les contours sont linéaires, tout en pénalisant les configurations où les contours sont courbés ou fragmentés.

5.5.2 Régularisation de contours basée MRF

Considérant que les contours E_c obtenus par le détecteur Canny sont la version altérée des contours idéaux E_i, à estimer, à partir des données image et d'un a priori donné. Dans notre cas, et parce que nos images représentent des objets polyédriques, les contours sont linéaires par morceaux. Nous montrons ensuite comment cet a priori est modélisé et utilisé pour calculer l'énergie du modèle qui sera minimisée après son ajout à l'énergie des données.

Selon le formalisme bayésien, en particulier le critère du maximum a postériori (MAP) pour l'estimation du modèle, le problème de régularisation des contours, peut être exprimé comme la maximisation de la probabilité a posteriori exprimée comme suit :

$$P(E_i|I_g) = \frac{P(I_g|E_i).P(E_i)}{P(I_g)} \tag{5.1}$$

Où : $P(I_g|E_i)$ est la vraisemblance et décrit la probabilité de trouver l'image du gradient I_g connaissant la carte de contours (idéale) E_i. Elle est généralement modélisée par une distribution connue (comme la distribution normale).

$P(E_i)$: est l'a priori et il décrit la probabilité de notre modèle (car il existe d'autres modèles alternatifs), il peut être modélisé par une distribution connue. Cependant, lorsqu'elle ne peut pas être définie par une distribution spécifique, elle est modélisée par une distribution de Gibbs, considérant que les données suivent un modèle markovien [112]. Dans notre cas, il sera donc défini en proposant un nouveau modèle MRF.

$P(I_g)$: est l'évidence et décrit la probabilité de trouver les données. Elle peut facilement être calculée par apprentissage statistique ou modélisée par une distribution connue (mais elle est généralement ignorée pour le calcul du maximum à postériori MAP, comme nous le verrons plus loin).

L'Ensemble idéal de contours E_i est celui qui donne la probabilité la plus élevée $P(E_i|I_g)$.

Notons que le MAP est une alternative parmi d'autres, et il est utilisé lorsqu'on peut définir la distribution pour le modèle $P(E_i)$ Cependant, il est possible d'utiliser d'autres alternatives telles que ML (Maximum Likelihood) lorsque $P(E_i)$ ne peut pas être estimé.

En considérant que $P(I_g)$ est constant, maximiser $P(E_i|I_g)$ revient à maximiser $P(I_g|E_i) * P(E_i)$, donc l'ensemble de contours recherché (idéal) E_i s'obtient comme suit :

$$E_i = \underset{E_i}{Argmax}(\, P(I_g|E_i) * P(E_i)) \tag{5.2}$$

Régularisation de contours par MAP-MRF

En considérant que $P(E_i)$ et $P(I_g|E_i)$ suivent chacun une distribution de Gibbs, ils peuvent être exprimés comme suit :

$$P(E_i) = \frac{1}{Z1} * \ e^{-IEnergy(E_i)} \tag{5.3}$$

$$P(I_g|E_i) = \frac{1}{Z2} * \ e^{-EEnergy(I_g|E_i)} \tag{5.4}$$

Où :

$Z1$ et $Z2$ sont des constantes de normalisation.

$IEnergy(E_i)$ Est l'énergie interne, où son calcul est basé sur l'a priori linéaire des contours, exprimé par un modèle markovien que nous introduisons ci-dessous.

$EEnergy(I_g|E_i)$ est l'énergie externe qui exprime l'adéquation du modèle E_i aux données I_g.

En d'autres termes, l'énergie externe décrit la force d'attachement des données au modèle. Plus l'énergie externe est faible, plus le modèle est fidèle aux données observées.

Ainsi, maximiser le MAP, $P(I_g|E_i) * P(E_i)$ équivaut à minimiser l'énergie totale $TEnergy$, exprimée par :

$TEnergy(E_i|I_g) = EEnergy(I_g|E_i) + IEnergy(E_i)$.

Pour trouver le minimum de la fonction objective ci-dessus, il est nécessaire de trouver un compromis idéal entre la fidélité des données au modèle et celle de l'a priori (linéarité des contours).

5.5.3 Formulation du problème de minimisation d'énergie

Le problème de régularisation des contours est formulé comme une minimisation d'énergie, où l'énergie totale est composée de deux termes : une énergie externe (fidélité aux données) et une énergie interne (respect de l'a priori).

5.5.3.1 Energie interne

Pour régulariser les contours nous considérons que les images de profondeur polyédriques ont des bordures linéaires sans courbes. C'est pourquoi nous privilégierons les contours linéaires dans notre modèle.

Pour une modélisation simpliste, nous divisons l'image en un ensemble de cliques de sites 3X3, et chaque clique aura un potentiel qui déterminera l'énergie interne comme le montre la figure ci-dessous.

Selon le modèle MRF, qui définit l'énergie comme la somme des potentiels d'un ensemble de cliques, on définit pour chaque clique C dans le voisinage 3x3 une énergie en fonction de la configuration des pixels dans ce voisinage :

0 : Si son centre est un point de contour et que seuls deux autres pixels sont des pixels de contour et qu'ils sont alignés avec lui.

0,5 : Si son centre est un point de contour et que seuls deux autres pixels sont des pixels de contour et qu'ils sont partiellement alignés avec lui.

1 : Dans tous les autres cas.

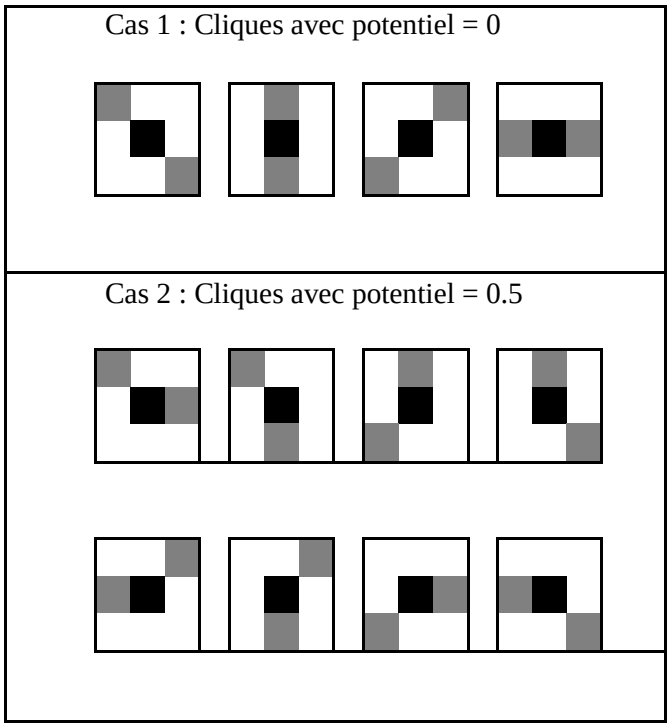


Figure 5.1 - Potentiel de cliques.

Pour calculer l'énergie interne d'une clique *C*, on utilise l'algorithme suivant :

```
Potentiel=1
Si le voisinage est du cas 1 Alors
Potentiel=0
Sinon
Si voisinage est du cas 2 Alors
Potentiel=0.5
Fsi
Fsi
Energie interne = Potentiel
```

5.5.3.2 Energie externe

Le modèle de contours doit être fidèle aux contours réels de l'image. Dans notre cas nous avons une sorte de données exprimant les contours détectées (avant régularisation) :

1. la carte des contours préliminaires obtenue par le filtre Canny.

2. L'image de gradient brute basée sur l'angle qui représente les variations qui ont été utilisées pour la détection des contours.

Ainsi l'énergie externe $EEnergy\ (I_g|E_i)$ s'exprime comme suit :

$$EEnergy = ECanny + EGradient.$$

5.5.3.3 Energie externe de Canny

Cette énergie mesure la similarité entre les contours obtenus selon le modèle (E_i) et ceux obtenus avec le filtre de Canny.

Une grande ressemblance donne une faible énergie et une faible ressemblance donne une grande énergie.

La similarité est calculée pour chaque clique comme suit :

C : La matrice 3X3 de la clique avec les contours obtenus par le filtre de Canny.

M : La matrice 3X3 de la clique avec les contours générés par notre modèle.

$$ECanny(i,j) = -\sum_{\substack{x=i-1\dots i+1 \\ y=j-1\dots j+1}} Sc(x,y) \qquad (5.5)$$

$$Sc(x,y) = \begin{cases} 1, & Si\ C(x,y)\ et\ M(x,y)\ sont\ des\ points\ contours \\ 0, & Sinon \end{cases}$$

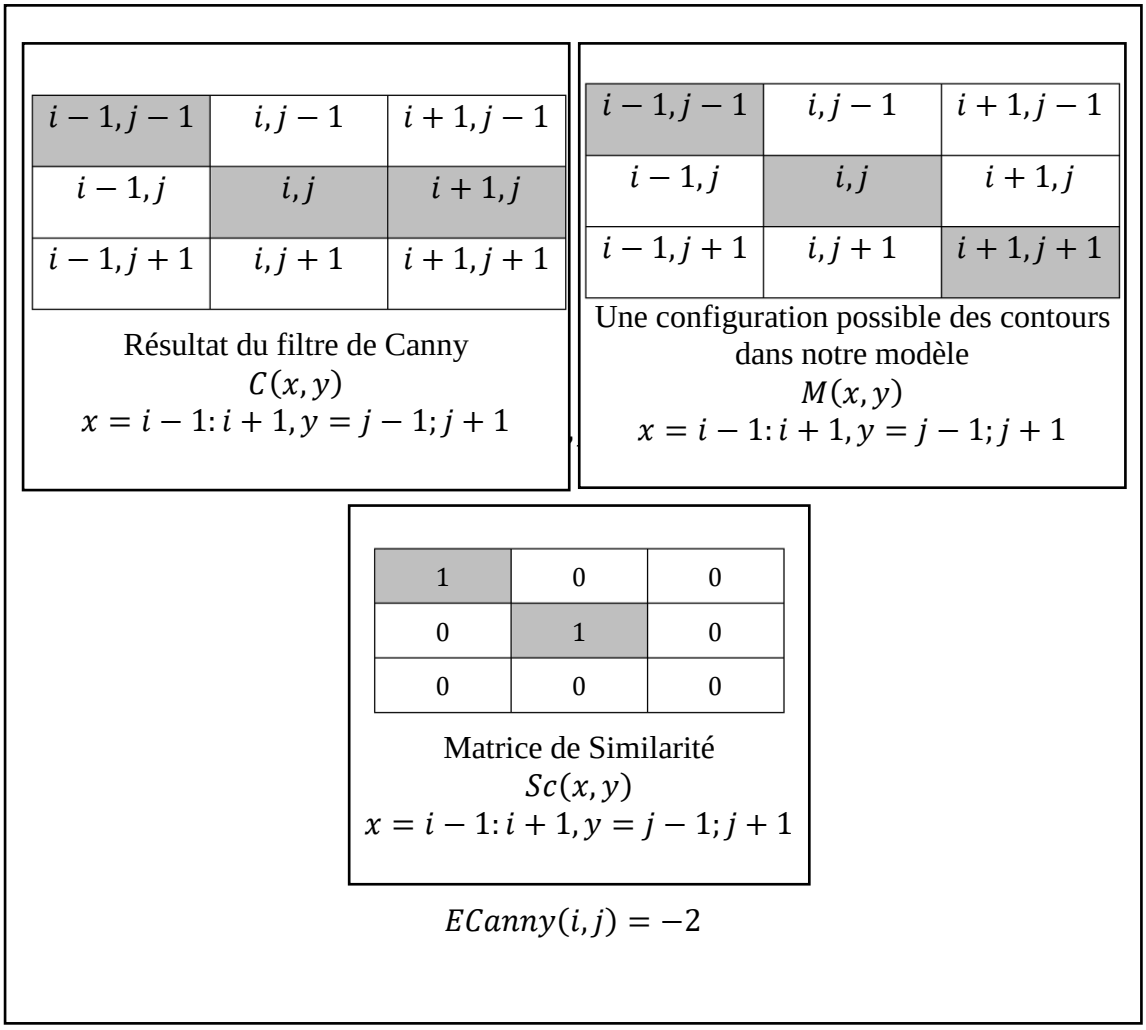


Figure 5.2 - Calcul de l'énergie externe de Canny.

On peut remarquer facilement que l'énergie est minimale lorsque la correspondance entre les contours de Canny et notre modèle est maximale.

5.5.3.4 Energie externe du gradient

Cette énergie donne plus de valeur aux points avec une forte norme du gradient indépendamment des résultats obtenus avec le filtre de Canny. Cela consiste à calculer la somme des valeurs de la norme du gradient pour les pixels choisis par notre modèle, puis à multiplier la somme par la valeur -1. Sa formulation formelle est la suivante :

G : la matrice 3X3 de la clique avec les normes du gradient.

M : la matrice 3X3 de la clique avec les contours générés par notre modèle.

$$EGradient(i,j) = -\sum_{\substack{x=i-1\dots i+1 \\ y=j-1\dots j+1}} Sg(x,y) \tag{5.6}$$

$$Sg(x,y) = \begin{cases} G(x,y), & Si\ M(x,y)\ est\ un\ point\ contour \\ 0, & Sinon \end{cases}$$

Ainsi l'énergie externe aura la formule :

$EExtern = ECanny + EGradient.$

$$EExtern = -\sum_{\substack{x=i-1\dots i+1 \\ y=j-1\dots j+1}} Sc(x,y) - \sum_{\substack{x=i-1\dots i+1 \\ y=j-1\dots j+1}} Sg(x,y) \tag{5.7}$$

5.5.4 Algorithme de minimisation d'énergie

Pour minimiser l'énergie globale de l'image entière, nous proposons de minimiser individuellement l'énergie de chaque clique et d'attribuer la valeur de la somme des énergies locales des cliques à l'énergie globale.

Le principe adopté est issu de la programmation dynamique où la solution d'un problème peut être obtenue en le décomposant en sous-problèmes plus faciles à résoudre.

Alors :

$$GlobalEnergy = \sum_{\substack{c \in cliques \\ de\ l'image}} Energy(c) \tag{5.8}$$

L'énergie d'une clique est donnée par :

$Energy(c) = Intern\ energy(c) + Externe\ energie(c)$

Afin d'obtenir l'énergie minimale d'une clique, il y a 12 configurations possibles (chacune a son propre potentiel et donc son énergie interne est inférieure ou égale à 0,5).

L'énergie totale de la clique est calculée pour toutes ces configurations et ne sera retenu que celle avec la plus petite valeur. L'addition de toutes les valeurs obtenues nous donne l'énergie globale de l'image.

Néanmoins, une seule itération semble insuffisante pour bien régulariser les contours de l'image. Une deuxième itération, utilisant les contours obtenus plutôt que ceux du filtre de Canny original, semble alors être obligatoire pour améliorer de plus en plus les contours jusqu'à la convergence.

L'algorithme est synthétisé comme suit :

```
Inputs :   Array :  CANNY (Carte de contours de Canny)
           Array :  GRADIENT (Toutes les norms de gradient)
           Max :    Nombre maximal d'itérations
           Epsilon
Outputs :  Array :  RESULT  (Carte finale de contours)

Iterations_Number = 0
Ancient_GlobalEnergy = nombre_tres_grand
While  Iterations_Number < Max Do

      GlabalEnergy = 0

      For each Clique Do
            CliqueEnergy = minimum(InternEnergy + ExternEnergy)
                  {On examine les 12 possibilités}
            GlobalEnergy = GlobalEnergy + CliqueEnergy
      EndFor

      // Nous modifions les contours RESULTAT selon GlobalEnergy
      If | Ancient_GlobalEnergy – GlobalEnergy | < epsilon Then
            Stop (L'algorithme a convergé)
      EndIf
      Iterations_Number = Iterations_Number +1
      Ancien_GlobalEnergy = GlobalEnergy
EndWhile
```

A noter que d'une itération à la suivante les cliques changent, ce qui donne un sens à la boucle externe des itérations. Lorsque au fil des itérations l'énergie globale se stagne à une valeur donnée, cette dernière est la minimale, et la configuration obtenues des cliques correspondra à l'image régularisée.

5.5.5 Complexité et efficacité

L'efficacité de notre algorithme est un aspect important de notre méthode, en particulier pour des applications en temps réel. Nous discutons ici de la complexité de l'algorithme et des optimisations possibles.

- **Complexité algorithmique** : La complexité de l'algorithme dépend du nombre de cliques dans l'image et du nombre d'itérations nécessaires pour atteindre la convergence. Pour une image de taille N×N, le nombre de cliques est de l'ordre de $O(N^2)$, et chaque clique est traitée en temps constant.

- **Optimisations** : Pour améliorer l'efficacité de l'algorithme, nous utilisons des techniques de programmation dynamique et de mémoïsation pour éviter de recalculer les énergies des cliques à chaque itération. De plus, nous exploitons la structure locale des MRF pour paralléliser le traitement des cliques, ce qui permet d'accélérer l'exécution de l'algorithme.

Ainsi nous avons arrivé au terme de la présentation de notre approche de régularisation de contours basée sur une modélisation bayésienne et une optimisation sous forme de minimisation d'énergie ainsi que les algorithmes utilisés pour trouver l'optimum.

Après avoir détaillé la méthodologie proposée, nous présentons dans la section suivante les expérimentations et résultats obtenus avec notre méthode. Nous décrivons les données utilisées, les métriques d'évaluation, et les résultats obtenus par rapport aux approches existantes.

5.6 Expérimentations et résultats

Dans cette section, nous présentons les expérimentations menées pour valider notre méthode de régularisation de contours basée sur les champs aléatoires de Markov (MRF) et la minimisation d'énergie. La régularisation consiste à améliorer les contours obtenus avec une étape de détection (dans notre cas, avec le nouveau filtre de Canny). Elle pourra être schématisée par la figure suivante (Figure 5.3) :

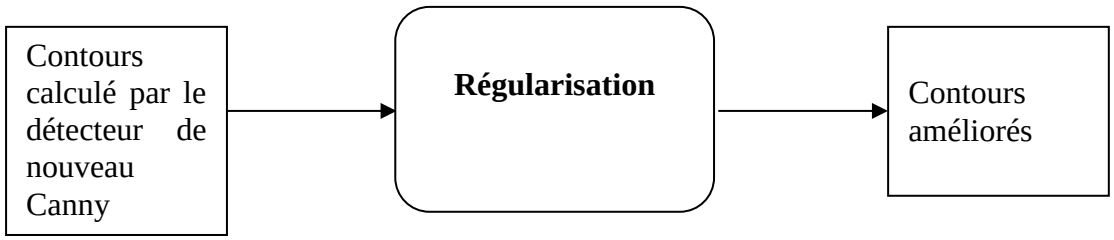


Figure 5.3 - Place de la régularisation dans le processus de segmentation.

Dans la suite de la section nous décrivons les données utilisées, les métriques d'évaluation, et les résultats obtenus, en comparant notre méthode avec d'autres approches existantes.

5.6.1 Données utilisées

Pour valider notre méthode, nous avons utilisé des images de profondeur issues de la base de données ABW, qui contient des scènes intérieures et industrielles avec des objets polyédriques. Ces images présentent des défis typiques, tels que du bruit, des artefacts, et des discontinuités de profondeur, ce qui les rend adaptées pour tester notre méthode de régularisation.

- **Base de données ABW** : La base de données ABW est largement utilisée dans la communauté de la vision par ordinateur pour évaluer les algorithmes de détection de contours et de segmentation. Elle contient des images de profondeur avec des annotations de contours de référence (vérité terrain), ce qui permet une évaluation quantitative des résultats.
- **Sélection des images** : Nous avons sélectionné un sous-ensemble d'images représentatives de la base de données ABW, en nous concentrant sur des scènes contenant des objets polyédriques (comme des meubles, des machines, ou des bâtiments). Ces images ont été choisies pour leur complexité et leur représentativité des défis rencontrés dans les applications réelles.

5.6.2 Métriques d'évaluation

Pour évaluer la performance de notre méthode, nous avons utilisé plusieurs métriques couramment utilisées dans la littérature pour évaluer la qualité des contours détectés. Ces métriques permettent de comparer les contours régularisés avec la vérité terrain (contours de référence).

1. **Précision (Precision)** : La précision mesure la proportion de contours détectés qui correspondent réellement à des contours dans la vérité terrain. Une précision élevée indique que peu de faux positifs sont présents dans les résultats.

$$\text{Précision} = \frac{\text{Vrais positifs (VP)}}{\text{Vrais positifs (VP)} + \text{Faux positifs (FP)}}$$

2. **Rappel (Recall)** : Le rappel mesure la proportion de contours de référence qui ont été correctement détectés. Un rappel élevé indique que peu de contours de référence ont été manqués.

$$\text{Rappel} = \frac{\text{Vrais positifs (VP)}}{\text{Vrais positifs (VP)} + \text{Faux négatifs (FN)}}$$

3. **F1-score** : Le F1-score est une métrique combinée qui prend en compte à la fois la précision et le rappel. Il est particulièrement utile pour évaluer la performance globale d'un algorithme.

$$\text{F1-score} = 2 \cdot \frac{\text{Précision} \cdot \text{Rappel}}{\text{Précision} + \text{Rappel}}$$

4. **Temps d'exécution** : Nous avons également mesuré le temps d'exécution de notre algorithme pour évaluer son efficacité computationnelle. Ceci est particulièrement important pour des applications en temps réel, comme la robotique ou la conduite autonome.

5.6.3 Résultats obtenus

Nous avons comparé les résultats de notre méthode avec ceux obtenus par le filtre de Canny adapté aux images de profondeur (Figure 5.3 et Figure 5.4).

- **Comparaison avec le filtre de Canny** : Les résultats montrent que notre méthode améliore significativement la qualité des contours détectés par rapport au filtre de Canny. En particulier, les contours régularisés sont plus linéaires et moins bruités, tout en préservant les détails importants.
- **Temps d'exécution** : Le temps d'exécution de notre algorithme est comparable à celui des autres méthodes, malgré la complexité supplémentaire introduite par la modélisation des MRF. Ceci est dû à l'utilisation de techniques d'optimisation telles que la mémoïsation et la programmation dynamique.

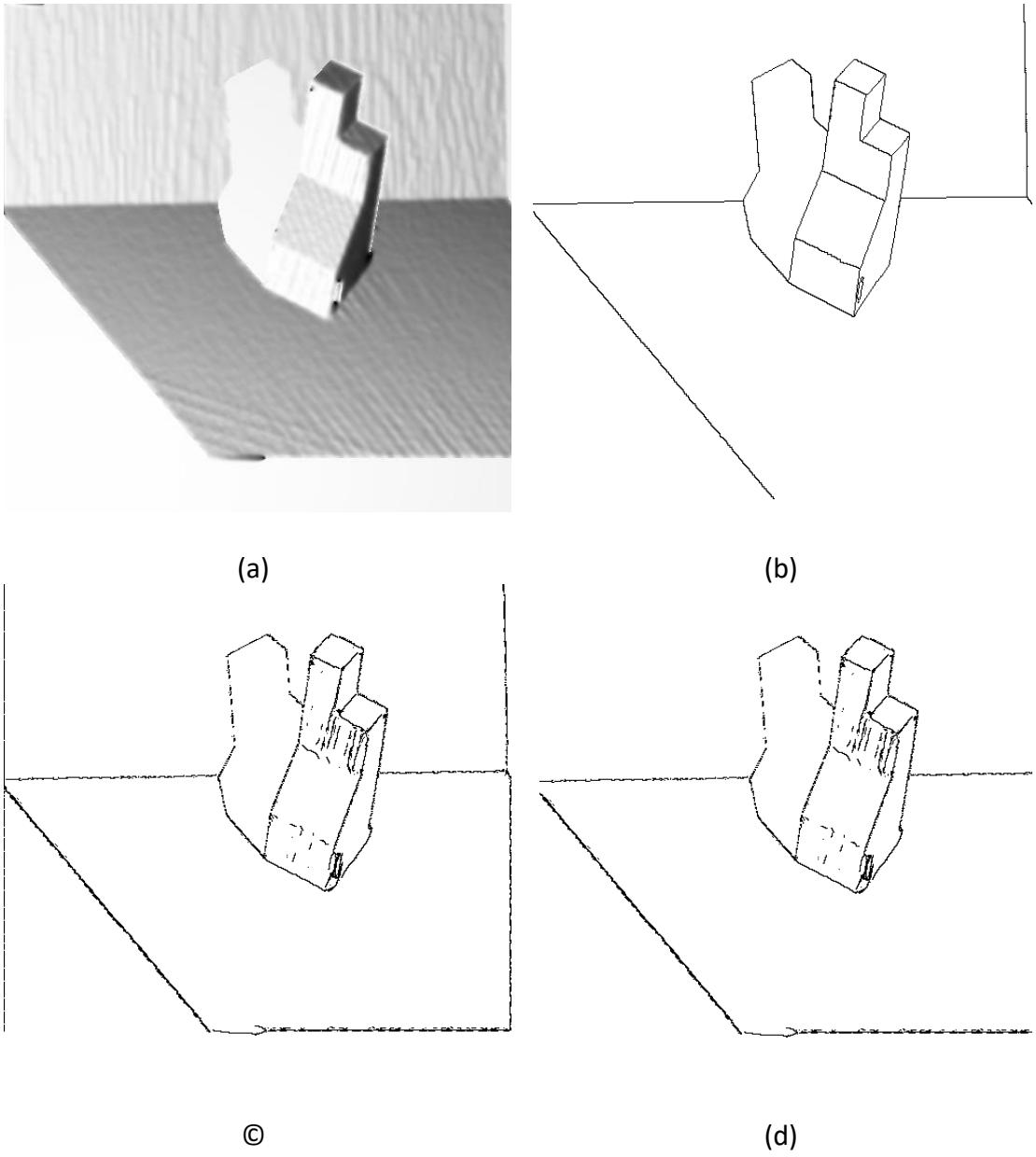


Figure 5.4 - Résultat de la segmentation de l'image de profondeur ABW.test.27 : (a) : image de relief, (b) : Réalité terrain des contours, (c) : Résultat de détection selon le nouveau filtre de Canny, (d) résultat après régularisation.

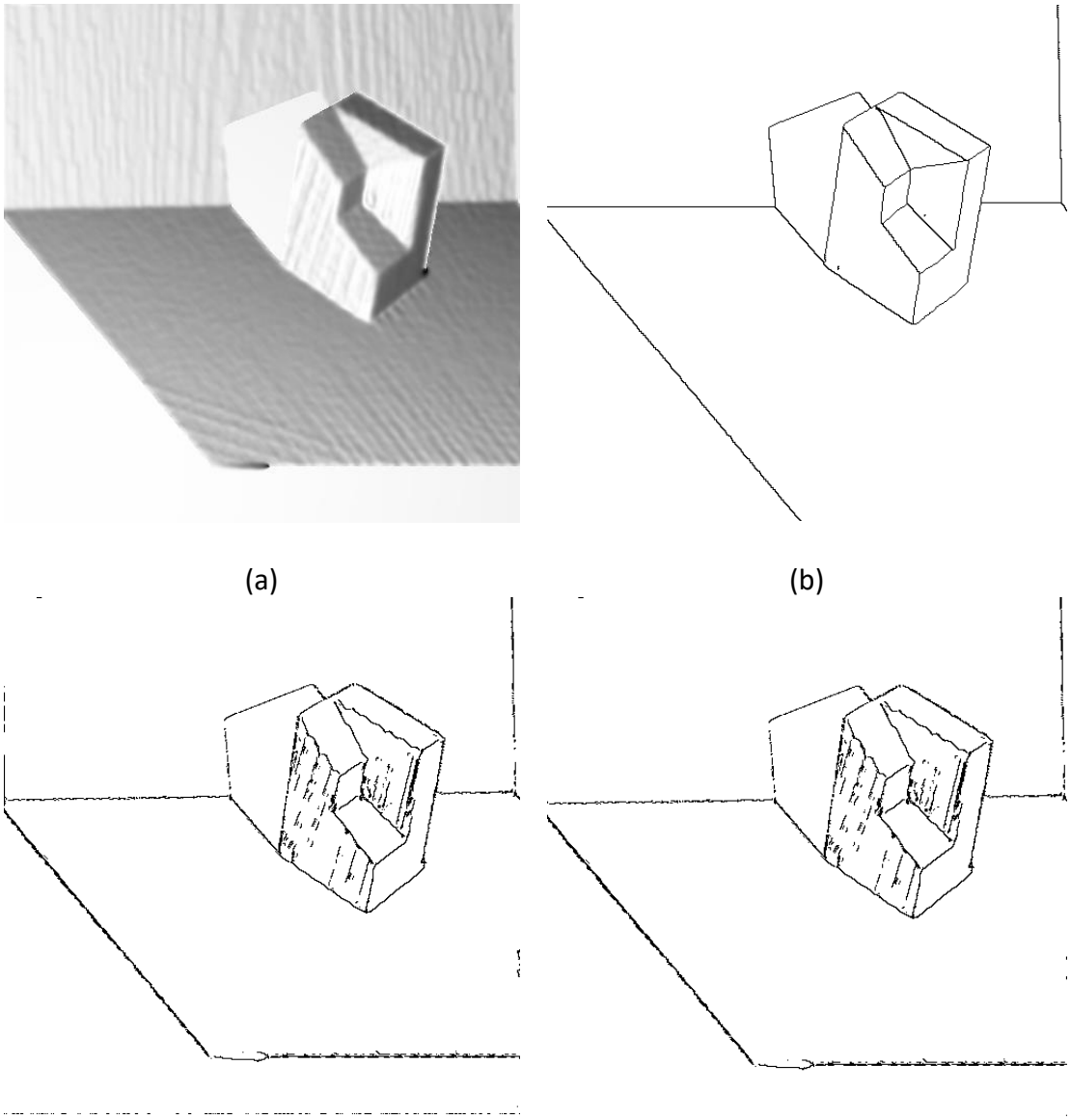


Figure 5.5 - Un autre exemple de détection de contours avec régularisation (image ABW.test.29)

Quantitativement parlant, le Tableau 5.1 montre les résultats des 3 métriques utilisées pour le détecteur de Canny adapté aux images de profondeur avant et après la régularisation des contours. Nous avons reporté les résultats des trois métriques pour l'ensemble des 30 images de la base ABW. Pour un nombre élevé d'images, les contours après régularisation ont été clairement améliorés, comparé à la réalité terrain. Cependant, et en considérant la moyenne des métriques pour les 30 images de test de la base ABW, nous constatons une légère amélioration. Cependant cette amélioration est jugée non proportionnelle par rapport à l'effort de calcul déployé pour exécuter la tache de régularisation, et qui est de nature itérative, car impliquant une optimisation de la fonction d'énergie globale. Néanmoins, la

réflexion sur d'autres a priori sur la nature des objets qui y figurent dans les images et les surfaces qui les composent peuvent améliorer significativement les résultats de régularisation.

Tableau 5.1 - Comparaison des métriques (Qualité, Précision, Rappel et F1) avant et après régularisation bayésienne des contours.

Image N°	Qualité (Q)		Précision		Rappel		F Mesure (F1)	
	Canny	+ Regul	Canny	+ Regul	Canny	+Regul	Canny	+ Regul
0	0,551282	0,550894	0,218484	0,218403	0,462032	0,462032	0,296676	0,296602
1	0,777294	0,777411	0,306481	0,306671	0,487888	0,487888	0,376471	0,376614
2	0,804817	0,804753	0,324775	0,325242	0,483267	0,483166	0,388478	0,388779
3	0,77069	0,771235	0,300051	0,300441	0,488829	0,488829	0,371853	0,372152
4	0,730281	0,730565	0,287584	0,287909	0,483521	0,483546	0,360659	0,360921
5	0,764084	0,764189	0,299323	0,299688	0,486724	0,486724	0,370685	0,370964
6	0,686151	0,685923	0,26873	0,269221	0,473564	0,473564	0,342886	0,343284
7	0,771191	0,771112	0,29382	0,294435	0,480732	0,480732	0,364723	0,365196
8	0,788062	0,788394	0,310532	0,310687	0,483665	0,483665	0,378227	0,378342
9	0,642834	0,642766	0,255859	0,256186	0,463298	0,463298	0,329661	0,329933
10	0,716355	0,716784	0,278223	0,278853	0,468771	0,468771	0,349194	0,34969
11	0,706228	0,70604	0,291194	0,291386	0,467404	0,467404	0,358833	0,35898
12	0,79494	0,795053	0,31969	0,32015	0,480121	0,480121	0,383815	0,384146
13	0,765354	0,765769	0,296522	0,296958	0,480005	0,480005	0,366586	0,366919
14	0,729188	0,729003	0,304731	0,304881	0,469393	0,469393	0,369549	0,36966
15	0,755616	0,756324	0,300279	0,300614	0,476874	0,476874	0,368512	0,368765
16	0,803858	0,803958	0,327968	0,328606	0,474003	0,474003	0,387689	0,388135
17	0,673586	0,673768	0,278073	0,278541	0,460148	0,460148	0,346657	0,347021
18	0,745029	0,745091	0,293972	0,294001	0,488288	0,488288	0,366995	0,367018
19	0,074769	0,074774	0,051472	0,051549	0,117279	0,117279	0,071545	0,071618
20	0,754735	0,754878	0,288183	0,288559	0,48195	0,48195	0,360691	0,360985
21	0,509814	0,510351	0,197891	0,198445	0,453256	0,453256	0,2755	0,276035
22	0,670276	0,670152	0,264113	0,264544	0,466921	0,466875	0,337385	0,337724
23	0,662827	0,662766	0,256894	0,257144	0,472301	0,472301	0,332781	0,332991
24	0,684497	0,68443	0,264102	0,264548	0,473953	0,473953	0,339194	0,339562
25	0,483211	0,483732	0,194268	0,194732	0,444206	0,444206	0,270316	0,270765
26	0,670123	0,670276	0,266894	0,266857	0,467414	0,467414	0,339775	0,339746
27	0,051019	0,051188	0,029742	0,029897	0,100221	0,100469	0,045871	0,046082
28	0,417632	0,41817	0,163024	0,163421	0,434695	0,434695	0,237121	0,23754
29	0,409957	0,410596	0,166627	0,167023	0,418337	0,418337	0,238327	0,238732
Moyenne	0,645523333	0,645678167	0,256650033	0,2569864	0,446302	0,4463062	0,324221833	0,3244967

5.6.4 Discussion des résultats

Les résultats obtenus démontrent l'efficacité de notre méthode pour la régularisation des contours dans les images de profondeur. En particulier, notre méthode permet d'améliorer la linéarité des contours tout en préservant les détails importants, ce qui est fondamental pour des applications telles que la reconstruction 3D ou la navigation autonome.

- **Amélioration de la linéarité** : Les contours régularisés par notre méthode sont plus linéaires et moins fragmentés que ceux obtenus par le filtre de Canny ou d'autres méthodes de régularisation. Ceci est dû à l'a priori de linéarité intégré dans notre modèle MRF.
- **Réduction du bruit** : Notre méthode permet de réduire significativement le bruit et les artefacts présents dans les contours détectés, tout en préservant les contours réels. Ceci est particulièrement important pour des applications où la précision des contours est nécessaire.
- **Efficacité computationnelle** : Grâce à l'utilisation de techniques d'optimisation telles que la mémoïsation et la programmation dynamique, notre algorithme est suffisamment efficace pour être applicable à des images de grande taille et en temps réel.

En considérant la carte de contours après régularisation (figure 4.4.e), on remarque l'amélioration de contours obtenus par la méthode de régularisation proposée. Ceci pourra être expliqué par le fait que certaines cliques ont changé de configuration dans le sens ou des pixels de contours se sont vus déplacés localement. En effet, un pixel détecté comme pixel de contour peut se déplacer localement, si ce déplacement n'altère pas considérablement l'énergie externe, liant les contours obtenus aux données réelles de l'image. Dans le cas d'un tel déplacement sans forte contrainte de données, améliore l'énergie interne dans le sens ou la configuration de pixels qui en résulte respecte l'a priori considéré, à savoir la linéarité des contours.

Après avoir présenté les expérimentations et les résultats obtenus, nous concluons ce chapitre en résumant les contributions principales de notre travail et en soulignant son importance dans le domaine de la vision par ordinateur.

5.7 Conclusion et perspectives

Dans ce chapitre, nous avons présenté une méthode de régularisation de contours basée sur les champs aléatoires de Markov (MRF) et la minimisation d'énergie, spécifiquement adaptée aux images de profondeur contenant des objets polyédriques. Notre méthode repose sur un a priori de linéarité des contours, qui permet d'améliorer la qualité des contours détectés tout en respectant la structure réelle des objets dans l'image. Les résultats obtenus démontrent l'efficacité de notre méthode pour la régularisation des contours, avec une amélioration significative par rapport aux approches existantes.

5.7.1 Contributions principales

Les contributions principales de ce travail sont les suivantes :

1. **Modélisation de l'a priori de linéarité** : Nous avons proposé un modèle MRF pour capturer la linéarité des contours dans les images de profondeur d'objets polyédriques. Ce modèle repose sur une distribution de Gibbs qui pénalise les écarts par rapport à l'a priori de droiture des contours.
2. **Formulation du problème de régularisation** : Nous avons formulé le problème de régularisation comme une minimisation d'énergie, où l'énergie totale est composée d'une énergie externe (fidélité aux données) et d'une énergie interne (respect de l'a priori). Cette formulation permet de trouver un compromis optimal entre la fidélité aux données observées et le respect de l'a priori.
3. **Algorithme de minimisation d'énergie** : Nous avons proposé un algorithme de minimisation basé sur la programmation dynamique, qui permet de résoudre le problème de régularisation de manière efficace. Cet algorithme est conçu pour être applicable à des images de grande taille et en temps réel.
4. **Validation expérimentale** : Nous avons validé notre méthode sur des images de profondeur issues de la base de données ABW, en comparant les résultats avant et après régularisation. Les résultats montrent une amélioration significative de la qualité des contours, avec une meilleure conformité à la structure réelle des objets dans l'image.

5.7.2 Importance de notre travail

Notre travail contribue à l'avancement des techniques de régularisation de contours dans les images de profondeur, en particulier pour des applications où la précision des contours est importante, comme la reconstruction 3D, la navigation autonome, et la robotique. En exploitant un a priori de linéarité des contours, notre méthode permet d'obtenir des résultats plus précis et plus réguliers que les approches existantes, tout en étant suffisamment efficace pour des applications en temps réel.

- **Amélioration de la qualité des contours** : Notre méthode permet de réduire le bruit et les artefacts dans les contours détectés, tout en préservant les détails importants. Ceci est particulièrement utile pour des applications où la précision des contours est essentielle, comme la reconstruction 3D ou la détection d'obstacles.
- **Efficacité computationnelle** : Grâce à l'utilisation de techniques d'optimisation telles que la mémoïsation et la programmation dynamique, notre algorithme est suffisamment efficace pour être applicable à des images de grande taille et en temps réel.

5.7.3 Perspectives

Bien que notre méthode ait démontré des résultats prometteurs, plusieurs pistes de recherche peuvent être explorées pour améliorer encore sa performance et son applicabilité :

1. **Extension à des objets non polyédriques** : Pour étendre notre méthode à des objets plus complexes, tels que des objets organiques ou des surfaces courbées, il serait nécessaire de définir un a priori plus flexible, capable de capturer des contours courbés ou irréguliers.
2. **Intégration de techniques d'apprentissage profond** : L'utilisation de méthodes d'apprentissage profond (deep learning) pour la détection initiale des contours pourrait améliorer la qualité des contours initiaux et rendre notre méthode plus robuste au bruit.
3. **Validation sur des données réelles** : Il serait intéressant d'évaluer notre méthode sur des données réelles, provenant par exemple de capteurs de profondeur utilisés dans des applications industrielles ou de robotique. Cela permettrait de valider la robustesse de notre méthode dans des conditions réelles.

4. **Intégration dans des applications pratiques** : Enfin, il serait intéressant d'intégrer notre méthode dans des applications pratiques, telles que des systèmes de reconstruction 3D, de navigation autonome, ou de robotique. Cela permettrait de démontrer l'utilité de notre méthode dans des contextes réels et d'identifier d'éventuelles améliorations nécessaires.

5.7.4 Conclusion

En conclusion, notre méthode de régularisation de contours basée sur les MRF et la minimisation d'énergie représente une contribution dans le domaine de la vision par ordinateur, en particulier pour les images de profondeur contenant des objets polyédriques. Les résultats obtenus démontrent que notre méthode permet d'améliorer significativement la qualité des contours, tout en étant suffisamment efficace pour des applications en temps réel. Nous espérons que ce travail ouvrira la voie à de nouvelles recherches dans ce domaine et contribuera au développement de techniques de régularisation encore plus performantes.

Chapitre 6

Conclusion générale

La segmentation d'images, comme problème mal posé, demeure une problématique dans le domaine du traitement d'image et de la vision par ordinateur. L'essence de la difficulté de la segmentation réside d'une part du manque des modèles d'objets qu'une image acquise peut en contenir, et d'autre part de l'imprécision et de l'incertitude des mesures des indices visuels (intensité, couleur, profondeur, … etc.).

Il ressort de la littérature du domaine de la segmentation d'images qu'une méthode universelle de segmentation est inexistante, et il faut spécialiser les méthodes aux différents types d'images traitées. Le bon exemple est la segmentation d'images de profondeur, où il a été démontré à travers des dizaines de travaux de littérature que les méthodes classiques de segmentation, utilisant l'intensité ou la couleur, sont inefficaces avec ce type d'image. En effet, l'image de profondeur ne contient pas d'indices visuels, mais des valeurs de profondeur exprimant dans leur ensemble la géométrie des objets qui ont été capturés dans l'image.

Partant de cette problématique, il a été question dans ce travail de thèse de proposer une méthode de détection bien appropriée aux images de profondeur, et de pouvoir utiliser les résultats de détection pour améliorer la segmentation en procédant selon une approche de régularisation. Etant donné cet objectif, nous nous sommes orientés vers la détection de contours dans les images de profondeur. Cependant, et dès les premiers tests d'opérateurs classiques de détection dont celui de Canny, nous avons constaté la difficulté de la détection de contours dans les images de profondeurs. En effet, pour les profondeurs, un contour peut exister en un point donné de l'image malgré la continuité et la dérivabilité des données de profondeur, ce qui n'est pas le cas pour les images visuelles en intensité du gris ou en couleurs.

Cette particularité des données des images de profondeur, nous a poussé à chercher une nouvelle représentation des données brutes afin qu'une détection puisse être opérée en

utilisant des filtres de détection classiques de contours, tels que le filtre de Canny. Ceci a consisté en notre première contribution, qui est l'adaptation du filtre de Canny pour les images de profondeur, et ce par le calcul d'une image formée de vecteurs normaux aux surfaces, et d'utiliser les angles entre ces vecteurs comme données pour le calcul du gradient de l'image. Ce dernier est utilisé par la suite selon l'algorithme de Canny pour en extraire les vrais points de contours, qui représente les arrêtes séparant les surfaces d'objets capturés dans les images de profondeur.

Le nouveau détecteur a montré sa performance, comparé aux détecteurs classiques, utilisant les données brutes de l'image. En effet, malgré le niveau élevé de bruit et de déformation dans les images de profondeur, le détecteur proposé a pu extraire efficacement les lignes de contours séparant les surfaces dans les images, issues de la base d'image de profondeur de référence ABW.

Cependant, les résultats obtenus sont sujets d'amélioration. Pour cette fin, nous avons opté pour la régularisation des contours, en considérant l'apriori de linéarité des contours dans les images de profondeur à objets polyédriques. Partant de l'apriori de la linéarité et en faisant recours au Framework MAP-MRF, célèbre en traitement statistiques d'images, nous avons défini l'ensemble de cliques et de leurs potentiels afin que la minimisation d'énergie selon ce Framework, résulte en une segmentation améliorée. Les résultats de régularisation sur les images de la base ABW ont montré l'intérêt de la régularisation de contours pour l'amélioration des résultats de segmentation.

Perspectives

Tout le long du travail de thèse, nous nous sommes restreints qu'aux images de profondeurs à objets polyédriques, où les contours sont des arêtes droites séparant les différentes surfaces ou définissant les horizons des objets (avec continuité ou discontinuité des données de profondeur). Cependant, nous savons très bien que les objets dans la nature ou dans les environnements artificiels peuvent se présenter selon des géométries complexes, impliquant des surfaces et des arêtes courbées. Dans nos futurs travaux, nous nous attaquons à ce type d'objets à surfaces complexes, où il est nécessaire d'adapter d'une part d'autres détecteurs de contours, et d'autre part formuler des aprioris permettant de faire recours à des méthodes de régularisation pour l'amélioration des résultats.

Bibliographie

[1] Newcombe, R. A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A. J., ... & Fitzgibbon, A. (2011, October). Kinectfusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE international symposium on mixed and augmented reality* (pp. 127-136). Ieee.

[2] Taylor, R. H., Menciassi, A., Fichtinger, G., Fiorini, P., & Dario, P. (2016). Medical robotics and computer-integrated surgery. Springer handbook of robotics, 1657-1684.

[3] Izadi, S., Kim, D., Hilliges, O., Molyneaux, D., Newcombe, R., Kohli, P., ... & Fitzgibbon, A. (2011, October). Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In Proceedings of the 24th annual ACM symposium on User interface software and technology (pp. 559-568).

[4] Khoshelham, K., & Elberink, S. O. (2012). Accuracy and resolution of kinect depth data for indoor mapping applications. sensors, 12(2), 1437-1454.

[5] Silberman, N., Hoiem, D., Kohli, P., & Fergus, R. (2012). Indoor segmentation and support inference from rgbd images. In Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part V 12 (pp. 746-760). Springer Berlin Heidelberg.

[6] Zhao, H., Jiang, L., Jia, J., Torr, P. H., & Koltun, V. (2021). Point transformer. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 16259-16268).

[7] Pavlidis, T., & Horowitz, S. L. (1974). Segmentation of plane curves. IEEE transactions on Computers, 100(8), 860-870.

[8] Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., & Ronneberger, O. (2016). 3D U-Net: learning dense volumetric segmentation from sparse annotation. In Medical Image Computing and Computer-Assisted Intervention–MICCAI 2016: 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part II 19 (pp. 424-432). Springer International Publishing.

[9] Dai, A., Chang, A. X., Savva, M., Halber, M., Funkhouser, T., & Nießner, M. (2017). Scannet: Richly-annotated 3d reconstructions of indoor scenes. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 5828-5839).

[10] Hazirbas, C., Ma, L., Domokos, C., & Cremers, D. (2016, November). Fusenet: Incorporating depth into semantic segmentation via fusion-based cnn architecture. In Asian conference on computer vision (pp. 213-228). Cham: Springer International Publishing.

[11] Gupta, S., Arbelaez, P., & Malik, J. (2013). Perceptual organization and recognition of indoor scenes from RGB-D images. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 564-571).

[12] Choe, J., Park, C., Rameau, F., Park, J., & Kweon, I. S. (2022, October). Pointmixer: Mlp-mixer for point cloud understanding. In European conference on computer vision (pp. 620-640). Cham: Springer Nature Switzerland.

[13] Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., & Bennamoun, M. (2020). Deep learning for 3d point clouds: A survey. IEEE transactions on pattern analysis and machine intelligence, 43(12), 4338-4364.

[14] Glennie, C. L., Kusari, A., & Facchin, A. (2016). Calibration and stability analysis of the VLP-16 laser scanner. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 40, 55-60.

[15] Hansard, M., Lee, S., Choi, O., & Horaud, R. P. (2012). Time-of-flight cameras: principles, methods and applications. Springer Science & Business Media.

[16] Zhang, Z. (2002). A flexible new technique for camera calibration. IEEE Transactions on pattern analysis and machine intelligence, 22(11), 1330-1334.

[17] Park, J., Joo, K., Hu, Z., Liu, C. K., & So Kweon, I. (2020). Non-local spatial propagation network for depth completion. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16 (pp. 120-136). Springer International Publishing.

[18] Shen, J., & Cheung, S. C. S. (2013). Layer depth denoising and completion for structured-light rgb-d cameras. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1187-1194).

[19] Canny, J. (1986). A computational approach to edge detection. IEEE Transactions on pattern analysis and machine intelligence, (6), 679-698.

[20] Harris, C., & Stephens, M. (1988, August). A combined corner and edge detector. In Alvey vision conference (Vol. 15, No. 50, pp. 10-5244).

[21] Shi, J. (1994, June). Good features to track. In 1994 Proceedings of IEEE conference on computer vision and pattern recognition (pp. 593-600). IEEE.

[22] Jung, B., & Sukhatme, G. S. (2010). Real-time motion tracking from a mobile robot. International Journal of Social Robotics, 2, 63-78.

[23] Adams, R., & Bischof, L. (1994). Seeded region growing. IEEE Transactions on pattern analysis and machine intelligence, 16(6), 641-647.

[24] Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine learning, 20, 273-297.

[25] Breiman, L. (2001). Random forests. Machine learning, 45, 5-32.

[26] Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., & Ronneberger, O. (2016). 3D U-Net: learning dense volumetric segmentation from sparse annotation. In Medical Image Computing and Computer-Assisted Intervention–MICCAI 2016: 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part II 19 (pp. 424-432). Springer International Publishing.

[27] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 2961-2969).

[28] Qi, C. R., Yi, L., Su, H., & Guibas, L. J. (2017). Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems, 30.

[29] Parvin, B. (1986). Segmentation of range images into planar surfaces by split and merge. In CVPR (pp. 415-417).

[30] Taylor, R. W., Savini, M., & Reeves, A. P. (1989). Fast segmentation of range imagery into planar regions. Computer vision, Graphics, and image Processing, 45(1), 42-60.

[31] Bhavsar, A. V., & Rajagopalan, A. N. (2010, August). Inpainting large missing regions in range images. In 2010 20th International Conference on Pattern Recognition (pp. 3464-3467). IEEE.

[32] Besl, P. J., & Jain, R. C. (1988). Segmentation through variable-order surface fitting. IEEE Transactions on pattern analysis and machine intelligence, 10(2), 167-192.

[33] Yokoya, N., & Levine, M. D. (1989). Range image segmentation based on differential geometry: A hybrid approach. IEEE Transactions on Pattern Analysis and Machine Intelligence, 11(6), 643-649.

[34] Kasvand, T. (1988, January). The k1k2 space in range image analysis. In 9th International Conference on Pattern Recognition (pp. 923-924). IEEE Computer Society.

[35] Gupta, A., & Bajcsy, R. K. (1992, March). Integrated approach for surface and volumetric segmentation of range images using biquadrics and superquadrics. In Applications of artificial intelligence X: Machine Vision and Robotics (Vol. 1708, pp. 210-227). SPIE.

[36] Gupta, A., & Bajcsy, R. (1993). Volumetric segmentation of range images of 3D objects using superquadric models. CVGIP: Image understanding, 58(3), 302-326.

[37] Jiang, X., & Bunke, H. (1994). Fast segmentation of range images into planar regions by scan line grouping. Machine vision and applications, 7, 115-122.

[38] Davignon, A. (1992, March). Contribution of edges and regions to range image segmentation. In Applications of Artificial Intelligence X: Machine Vision and Robotics (Vol. 1708, pp. 228-239). SPIE.

[39] Wong, Y. C., Choi, L. J., Singh, R. S. S., & Zhang, H. (2019). Deep learning-based racing bib number detection and recognition. Jordanian Journal of Computers and Information Technology, 5(3).

[40] Shotton, J., Winn, J., Rother, C., & Criminisi, A. (2009). Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. International journal of computer vision, 81, 2-23.

[41] Villiger, M., Pache, C., Leitgeb, R. A., & Lasser, T. (2010, February). Coherent transfer functions and extended depth of field. In Optical Coherence Tomography And Coherence Domain Optical Methods In Biomedicine XIV (Vol. 7554, pp. 129-133). SPIE.

[42] Antonelli, L., De Simone, V., & di Serafino, D. (2020). Spatially adaptive regularization in image segmentation. Algorithms, 13(9), 226.

[43] Lucas, A., Iliadis, M., Molina, R., & Katsaggelos, A. K. (2018). Using deep neural networks for inverse problems in imaging: beyond analytical methods. IEEE Signal Processing Magazine, 35(1), 20-36.

[44] Kass, M., Witkin, A., & Terzopoulos, D. (1988). Snakes: Active contour models. International journal of computer vision, 1(4), 321-331.

[45] Alvarez, L., & Morel, J. M. (1994). Formalization and computational aspects of image analysis. Acta numerica, 3, 1-59.

[46] Osher, S., & Sethian, J. A. (1988). Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. Journal of computational physics, 79(1), 12-49.

[47] Pratt, W. K. (2007). JEA Jr. Digital Image Processing, 4th ed. Hoboken, NJ, USA: John Wiley & Sons, Inc.

[48] Revol-Muller, C., Grenier, T., Rose, J. L., Pacureanu, A., Peyrin, F., & Odet, C. (2013). Region growing: When simplicity meets theory–region growing revisited in feature space and variational framework. In Computer Vision, Imaging and Computer Graphics. Theory and Application (pp. 426-444). Springer, Berlin, Heidelberg.

[49] Mumford, D. B., & Shah, J. (1989). Optimal approximations by piecewise smooth functions and associated variational problems. Communications on pure and applied mathematics.

[50] Geman, S., & Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. IEEE Transactions on pattern analysis and machine intelligence, (6), 721-741.

[51] Chan, T. F., & Vese, L. A. (2001). Active contours without edges. IEEE Transactions on image processing, 10(2), 266-277.

[52] Chan, T. F., Esedoglu, S., & Nikolova, M. (2006). Algorithms for finding global minimizers of image segmentation and denoising models. SIAM journal on applied mathematics, 66(5), 1632-1648.

[53] Kato, Z., & Pong, T. C. (2006). A Markov random field image segmentation model for color textured images. Image and Vision Computing, 24(10), 1103-1114.

[54] Doersch, C., Gupta, A., & Efros, A. A. (2015). Unsupervised visual representation learning by context prediction. In Proceedings of the IEEE international conference on computer vision (pp. 1422-1430).

[55] rizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. Communications of the ACM, 60(6), 84-90.

[56] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2014). Going Deeper with Convolutions. arXiv e-prints, page. arXiv preprint arXiv:1409.4842.

[57] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[58] Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3431-3440).

[59] Parisi, G., & Shankar, R. (1988). Statistical field theory.

[60] Besag, J. (1974). Spatial interaction and the statistical analysis of lattice systems. Journal of the Royal Statistical Society: Series B (Methodological), 36(2), 192-225.

[61] Woods, J. (1972). Two-dimensional discrete Markovian fields. IEEE Transactions on Information Theory, 18(2), 232-240.

[62] Clifford, P., & Hammersley, J. M. (1971). Markov fields on finite graphs and lattices.

[63] Moussouris, J. (1974). Gibbs and Markov random systems with constraints. Journal of statistical physics, 10(1), 11-33.

[64] Kindermann, R., & Snell, J. L. (1982). Markov Random Fields and Their Application, Providence, RI: Amer. Math. Soc.

[65] Ferguson. T. S. (1967). Mathematical Statistics: A Decision Theoretic Approach. Probability and Mathematical Statistics, New York, NY: Academic Press.

[66] Chellapa, R. (1985). Two-dimensional discrete Gaussian Markov random field models for image processing. Progress in Pattern Recognition, 1985, 2, 79-112.

[67] Elliott, H., Derin, H., Cristi, R., & Geman, D. (1984, March). Application of the Gibbs distribution to image segmentation. In ICASSP'84. IEEE International Conference on Acoustics, Speech, and Signal Processing (Vol. 9, pp. 678-681). IEEE.

[68] Derin, H., & Cole, W. S. (1986). Segmentation of textured images using Gibbs random fields. Computer Vision, Graphics, and Image Processing, 35(1), 72-98.

[69] Derin, H., & Elliott, H. (1987). Modeling and segmentation of noisy and textured images using Gibbs random fields. IEEE Transactions on pattern analysis and machine intelligence, (1), 39-55.

[70] Strauss, D. J. (1977). Clustering on coloured lattices. Journal of Applied Probability, 14(1), 135-143.

[71] Lakshmanan, S., & Derin, H. (1989). Simultaneous parameter estimation and segmentation of Gibbs random fields using simulated annealing. IEEE Transactions on Pattern Analysis and Machine Intelligence, 11(8), 799-813.

[72] Hu, R., & Fahmy, M. M. (1992). Texture segmentation based on a hierarchical Markov random field model. Signal processing, 26(3), 285-305.

[73] Won, C. S., & Derin, H. (1992). Unsupervised segmentation of noisy and textured images using Markov random fields. CVGIP: Graphical models and image processing, 54(4), 308-328.

[74] Simchony, T., & Chellappa, R. (1988, January). Stochastic and deterministic algorithms for MAP texture segmentation. In ICASSP-88., International Conference on Acoustics, Speech, and Signal Processing (pp. 1120-1121). IEEE Computer Society.

[75] Wu, Z., & Leahy, R. (1993). An approximate method of evaluating the joint likelihood for first-order GMRFs. IEEE transactions on image processing, 2(4), 520-523.

[76] Zhu, S. C., Wu, Y. N., & Mumford, D. (1997). Minimax entropy principle and its application to texture modeling. Neural computation, 9(8), 1627-1660.

[77] Zhu, S. C., & Mumford, D. (1997). Prior learning and Gibbs reaction-diffusion. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(11), 1236-1250.

[78] Zhu, S. C., Wu, Y., & Mumford, D. (1998). Filters, random fields and maximum entropy (FRAME): Towards a unified theory for texture modeling. International Journal of Computer Vision, 27(2), 107-126.

[79] Zheng, C., Wang, L., Chen, R., & Chen, X. (2012). Image segmentation using multiregion-resolution MRF model. IEEE geoscience and remote sensing letters, 10(4), 816-820.

[80] Zhou, Q., Zhu, J., & Liu, W. (2013). Learning dynamic hybrid Markov random field for image labeling. IEEE Transactions on Image Processing, 22(6), 2219-2232.

[81] Zhu, G., Wang, Q., Yuan, Y., & Yan, P. (2013). Learning saliency by MRF and differential threshold. IEEE Transactions on Cybernetics, 43(6), 2032-2043.

[82] Ghamisi, P., Benediktsson, J. A., & Ulfarsson, M. O. (2013). Spectral–spatial classification of hyperspectral images based on hidden Markov random fields. IEEE Transactions on Geoscience and Remote Sensing, 52(5), 2565-2574.

[83] Gong, M., Su, L., Jia, M., & Chen, W. (2013). Fuzzy clustering with a modified MRF energy function for change detection in synthetic aperture radar images. IEEE Transactions on Fuzzy Systems, 22(1), 98-109.

[84] Xu, K., Yang, W., Liu, G., & Sun, H. (2012). Unsupervised satellite image classification using Markov field topic model. IEEE Geoscience and Remote Sensing Letters, 10(1), 130-134.

[85] Eches, O., Benediktsson, J. A., Dobigeon, N., & Tourneret, J. Y. (2012). Adaptive Markov random fields for joint unmixing and segmentation of hyperspectral images. IEEE Transactions on Image Processing, 22(1), 5-16.

[86] Hochbaum, D. S. (2001). An efficient algorithm for image segmentation, Markov random fields and related problems. Journal of the ACM (JACM), 48(4), 686-701.

[87] Kwon, O. W., & Lee, J. H. (2000, November). Web page classification based on k-nearest neighbor approach. In Proceedings of the fifth international workshop on on Information retrieval with Asian languages (pp. 9-15).

[88] Zhang, R., Ouyang, W., & Cham, W. K. (2009, November). Image multi-scale edge detection using 3-D hidden Markov model based on the non-decimated wavelet. In 2009 16th IEEE International Conference on Image Processing (ICIP) (pp. 2173-2176). IEEE.

[89] Tu, Z., & Bai, X. (2009). Auto-context and its application to high-level vision tasks and 3D brain image segmentation. IEEE transactions on pattern analysis and machine intelligence, 32(10), 1744-1757.

[90] Nascimento, J. M., & Bioucas-Dias, J. M. (2011). Hyperspectral unmixing based on mixtures of Dirichlet components. IEEE Transactions on Geoscience and Remote Sensing, 50(3), 863-878.

[91] Yang, Y., Hallman, S., Ramanan, D., & Fowlkes, C. C. (2011). Layered object models for image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 34(9), 1731-1743.

[92] Khalifa, F., Beache, G. M., Gimelrfarb, G., Giridharan, G. A., & El-Baz, A. (2011). Accurate automatic analysis of cardiac cine images. IEEE Transactions on Biomedical Engineering, 59(2), 445-455.

[93] Coletta, L. F., Vendramin, L., Hruschka, E. R., Campello, R. J., & Pedrycz, W. (2011). Collaborative fuzzy clustering algorithms: Some refinements and design guidelines. IEEE Transactions on Fuzzy Systems, 20(3), 444-462.

[94] Tarabalka, Y., Tilton, J. C., Benediktsson, J. A., & Chanussot, J. (2011). A marker-based approach for the automated selection of a single segmentation from a hierarchical set of image segmentations. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 5(1), 262-272.

[95] Mohamed, C., & Smaine, M. (2019, December). Edge detection in range images using a modified Canny filter. In 2019 International Conference on Theoretical and Applicative Aspects of Computer Science (ICTAACS) (Vol. 1, pp. 1-7). IEEE.

[96] Canny, J. (1986). A computational approach to edge detection. IEEE Transactions on pattern analysis and machine intelligence, (6), 679-698.

[97] D'Haeyer, J. P. (1989). Gaussian filtering of images: A regularization approach. Signal processing, 18(2), 169-181.

[98] Bhardwaj, S., & Mittal, A. (2012). A survey on various edge detector techniques. Procedia Technology, 4, 220-226.

[99] Hoover, A., Jean-Baptiste, G., Jiang, X., Flynn, P. J., Bunke, H., Goldgof, D. B., ... & Fisher, R. B. (1996). An experimental comparison of range image segmentation algorithms. IEEE transactions on pattern analysis and machine intelligence, 18(7), 673-689.

[100] Olive, D. J., & Olive, D. J. (2017). Multiple linear regression (pp. 17-83). Springer International Publishing.

[101] Mazouzi, S., & Guessoum, Z. (2021). A fast and fully distributed method for region-based image segmentation: Fast distributed region-based image segmentation. Journal of Real-Time Image Processing, 18(3), 793-806.

[102] Richtsfeld, A., Mörwald, T., Prankl, J., Zillich, M., & Vincze, M. (2012, October). Segmentation of unknown objects in indoor environments. In 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (pp. 4791-4796). IEEE.

[103] Dice, L. R. (1945). Measures of the amount of ecologic association between species. Ecology, 26(3), 297-302.

[104] Cheribet, M., & Mazouzi, S. (2021). A New Adapted Canny Filter for Edge Detection in Range Images. Jordanian Journal of Computers and Information Technology, 7(3).

[105] Newcombe, R. A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A. J., ... & Fitzgibbon, A. (2011, October). Kinectfusion: Real-time dense surface mapping and tracking. In 2011 10th IEEE international symposium on mixed and augmented reality (pp. 127-136). IEEE.

[106] Geiger, A., Lenz, P., & Urtasun, R. (2012). Are we ready for autonomous driving? The KITTI vision benchmark suite. In Conference on Computer Vision and Pattern Recognition (CVPR), 2012 IEEE (pp. 3354-3361). IEEE.

[107] Thrun, S., Burgard, W., & Fox, D. (2005). Probabilistic robotics cambridge. MA: MIT Press [Google Scholar].

[108] Felzenszwalb, P. F., & Huttenlocher, D. P. (2004). Efficient graph-based image segmentation. International journal of computer vision, 59, 167-181.

[109] Blake, A., & Zisserman, A. (1987). Visual reconstruction. MIT press.

[110] Boykov, Y., Veksler, O., & Zabih, R. (2001). Fast approximate energy minimization via graph cuts. IEEE Transactions on pattern analysis and machine intelligence, 23(11), 1222-1239.

[111] Khoshelham, K., & Elberink, S. O. (2012). Accuracy and resolution of Kinect depth data for indoor mapping applications. Sensors, 12(2), 1437-1454.

[112] Li, S. Z. (2001). Markov random field modeling in image analysis. Springer.

[113] Bebis, G., Boyle, R. I. C. H. A. R. D., Parvin, B. A. H. R. A. M., Koracin, D., & Ushizima, D. (2019). Advances in visual computing. Springer International Publishing.

[114] Marr, D. (2010). Vision: A computational investigation into the human representation and processing of visual information. MIT press.

[115] Kato, Z., & Zerubia, J. (2012). Markov random fields in image segmentation. Foundations and Trends® in Signal Processing, 5(1–2), 1-155.

[116] Antonelli, L., De Simone, V., & di Serafino, D. (2022). A view of computational models for image segmentation. ANNALI DELL'UNIVERSITA'DI FERRARA, 68(2), 277-294.

[117] Ramesh, K. K. D., Kumar, G. K., Swapna, K., Datta, D., & Rajest, S. S. (2021). A review of medical image segmentation algorithms. EAI Endorsed Transactions on Pervasive Health & Technology, 7(27).

[118] Bennai, M. T., Guessoum, Z., Mazouzi, S., Cormier, S., & Mezghiche, M. (2020). A stochastic multi-agent approach for medical-image segmentation: application to tumor segmentation in brain MR images. Artificial Intelligence in Medicine, 110, 101980.