



**UNIVERSITE SULTAN MOULAY SLIMANE**  
**Faculté des Sciences et Techniques**  
**Béni-Mellal**



*Centre d'Études Doctorales : Sciences et Techniques*  
*Formation Doctorale : Mathématiques et Physique Appliquées*

## **THÈSE**

Présentée par

**Ilhame AGNAOU**

Pour l'obtention du grade de

**DOCTEUR**

*Spécialité : Informatique*

---

---

## **Représentation d'Objets 3D et Extraction des Primitives pour un Système de Reconnaissance**

---

---

**Soutenue le 12/12/2020, devant la commission d'examen composée de :**

Président	: Pr. Said SAFI	Faculté Polydisciplinaire, Béni Mellal
Rapporteur	: Pr. Ali RACHIDI	Faculté des Sciences Appliquées, Ait Melloul
Examineur	: Pr. Mohamed BASLAM	Faculté des Sciences et Techniques, Béni Mellal
Co-directeur	: Pr. Belaid BOUIKHALENE	Faculté Polydisciplinaire, Béni Mellal
Directeur de thèse	: Pr. Ahmed ZEGHAL	Faculté des Sciences et Techniques, Tanger



---

*À mes parents qui se sont beaucoup sacrifiés  
pour mes études*

---

---

Cette thèse a été réalisée au sein du Laboratoire de Mathématiques et Applications (**LMA**) de la Faculté des Sciences et Techniques Beni Mellal (**FSTBM**) avec le Laboratoire d'Innovation en Mathématiques, Applications & Technologies de l'Information (**LIMATI**) de la Faculté Polydisciplinaire Beni Mellal (**FPBM**) sous la direction du professeur Ahmed Zeghal et l'encadrement du professeur Belaid Bouikhalene.

---

## **Remerciements**

---

---

Ce travail a été effectué au sein du Laboratoire de mathématiques et applications (LMA) de la Faculté des Sciences et Techniques Beni Mellal (FSTBM) avec le Laboratoire d'Innovation en Mathématiques, Applications & Technologies de l'Information (LIMATI) de la Faculté Polydisciplinaire Beni Mellal (FPBM), sous la direction du professeur **Ahmed ZEGHAL**.

**Je** tiens à lui écrouler ici toute ma reconnaissance pour sa disponibilité et ses précieux conseils.

**J'**assume très sincèrement de ma gratitude, le professeur **Belaid BOUIKHALENE**, qui m'a dirigé au long de mon travail et m'a longuement encouragé et m'a motivé. Je suis sensible à l'honneur qu'il me fait de participer au jury.

**Monsieur** le Professeur **Said SAFI** d'avoir accepté d'être président du jury. Je lui adresse mes sincères remerciements.

**Je** remercie le Professeur **Ali RACHIDI** pour avoir accepté de rapporter ma thèse, et pour ses conseils et ses remarques pertinentes.

**Je** remercie également le Professeur **Mohamed BASLAM** pour l'intérêt qu'il a porté à mon travail et d'avoir accepté de participer au jury en tant qu'examineur.

**Je** remercie tous les membres du Laboratoire de mathématiques et applications et toute l'équipe d'informatique.

**Merci** beaucoup à tous mes amis qui, de près ou de loin m'ont aidé et encouragé aux moments opportuns. Je les remercie pour tout le temps précieux que nous avons passé ensemble.

**Et** enfin, cette aventure n'aurait pas été possible sans le soutien de **Mes parents** et de **Mes frères**. Je vous remercie de tout mon cœur pour votre patience et vos encouragements car lorsque ma motivation était chancelante, vous avez su insuffler un vent d'énergie m'aidant à avancer.

## Résumé

---

---

*Le travail présenté dans ce document s'inscrit dans le cadre de la reconnaissance, et plus précisément, l'identification, la description et la représentation des objets 3D en vue 2D, cette thèse s'est divisée en trois axes :*

*Dans le premier axe, nous avons élaboré et développé une approche de l'algorithme d'amincissement « Thinning » appliquée à des objets 3D en vue 2D. Le but de cette nouvelle méthode c'est d'extraire des squelettes à partir de ces objets et de les comparer avec des objets 3D voxels pour bien décrire la morphologie des objets utilisés et faciliter la reconnaissance des objets 3D en vue 2D à partir de leurs squelettes.*

*Dans le deuxième axe de ce travail, nous avons effectué dans le cadre de la reconnaissance des objets, une détection des objets 3D et leurs formes libres par les descripteurs et les détecteurs locaux des points d'intérêt afin de les identifier. Dans ce contexte, notre objectif c'est de représenter et extraire un objet 3D en vue 2D à partir de la détection de leurs points d'intérêt et notifier les caractéristiques de détection et de description de chaque objet pour faciliter sa recherche dans les bases de données.*

*Pour cette raison, nous avons proposé un nouveau détecteur robuste au bruit, performant et qui englobe des critères pour extraire des points caractéristiques des objets 3D et qui sont les points d'intérêt en précisant leurs formes libres, ce détecteur est basé sur le détecteur de SUSAN tout en utilisant des mesures différentielles et une étude comparative pour l'évaluer et savoir sa performance.*

*La reconnaissance d'objets tridimensionnels est un problème très difficile. En utilisant une base de données de vues 2D d'objets de références, de nombreux problèmes se posent dans ce domaine. En particulier, lorsque le nombre d'objets augmente, la taille de la base de données augmente et il devient de plus en plus difficile de reconnaître correctement les objets puisque les informations ne permettent plus de séparer correctement les classes des objets.*

*Pour éviter ces problèmes en utilisant d'autres sources d'information que les vues et les images 2D des objets 3D. Ces informations supplémentaires viennent de la structure de l'objet, ou de son contexte habituel, c'est pour cela l'idée établie dans le troisième axe c'était de faire une extraction des maillages 3D des objets en vue 2D en se basant sur les surfaces de Bezier pour obtenir un bon maillage extrait et exacte de l'objet concerné afin de faciliter les tâches de la reconnaissance des objets à partir de leurs maillages, la reconstruction de leurs modèles 3D ou leur recherche dans une base de données...etc.*

### Mots-clés

**« Thinning » ; Objet 3D/2D ; Squelette ; Reconnaissance, détection, détecteur, point d'intérêt ; surface Bezier ; Maillage ; extraction ; forme libre ;**

## **Abstract**

---

---

*The work presented in this document is part of the recognition, and more specifically, the identification, description and representation of 3D objects in 2D view, this thesis is divided into three axes:*

*In the first axis, we have developed and developed an approach to the thinning algorithm "Thinning" applied to 3D objects in 2D view. The goal of this new method is to extract skeletons from these objects and compare them with 3D voxel objects to properly describe the morphology of the objects used and facilitate the recognition of 3D objects in 2D view from their skeletons.*

*In the second axis of this work, we carried out in the framework of object recognition, a detection of 3D objects and their free forms by local descriptors and detectors of interest points in order to identify them. In this context, our objective is to represent and extract a 3D object in 2D view from the detection of their interest points and to notify the detection and description characteristics of each object to facilitate its search in databases.*

*For this reason, we proposed a new high performance and noise resistant detector that includes criteria for extracting characteristic points from 3D objects and which are the interest points by specifying their free forms. This detector is based on the SUSAN detector while using differential measurements and a comparative study to evaluate it and know its performance.*

*The recognition of three-dimensional objects is one of the problems that is very difficult. When using a database of 2D views of reference objects, many problems arise in this field. In particular, as the number of objects increases, the size of the database increases and it becomes more and more difficult to recognize objects correctly since the information no longer allows the classes of objects to be separated correctly.*

*To avoid these problems by using sources of information other than 2D views and images of 3D objects. This additional information comes from the structure of the object, or from its usual context, this is why the established idea in the third axis was to make an extraction of 3D meshes of objects in 2D view based on Bezier surfaces to obtain a good extracted and exact mesh of the object concerned in order to facilitate the tasks of recognizing objects from their meshes, reconstructing their 3D models or their search in a database... etc.*

### **Keywords**

***"Thinning"; 3D/2D Object; Skeleton; Recognition, detection, detector, point of interest; Bezier surface; Meshing; extraction; free shape;***

العمل المقدم في هذا المستند هو جزء من التشخيص، وبشكل أكثر تحديدًا، تحديد ووصف وتمثيل الكائنات ثلاثية الأبعاد في عرض ثنائي الأبعاد، وتنقسم هذه الرسالة إلى ثلاث محاور:

في المحور الأول، قمنا بتصميم وتطوير نهج جديد لترقق خوارزمية " النحت والترقيق " تطبيقها على كائنات ثلاثية الأبعاد في عرض ثنائي الأبعاد والغرض من هذا الأسلوب الجديد لاستخراج الهياكل العظمية من هذه الكائنات ومقارنتها مع الأجسام الحجمية ثلاثية الأبعاد لوصف صحيح التشكل من الأشياء المستخدمة وتسهيل التشخيص الثلاثي الأبعاد للكائنات من هياكلها.

في المحور الثاني من هذا العمل، ونحن في إطار التعرف على الأشياء وعلى وجه الخصوص الكشف عن أجسام ثلاثية الأبعاد وأشكالها للتعرف عليها. ومع ذلك، فإنه لا يزال العديد من المشاكل في هذه الجزء التي ترتبط بكمية كبيرة من المعلومات. في هذا السياق، وهدفنا هو التعرف على وجوه ثلاثية الأبعاد من الكشف عن نقاط اهتمامهم وخصائص الكشف ووصف كل كائن لتسهيل بحثه في قواعد البيانات.

لهذا السبب، فإننا سوف نقترح كاشف قوي جديد مقاوم للضجيج، ويشمل عدة معايير لاستخراج نقاط التميز الثلاثية الأبعاد للكائنات التي هي النقاط التي تهم وتفيد أشكالها الحرة، ويستند كاشف على سوزان أثناء استخدامه عدة قياسات تفاضلية ودراسة مقارنة لتقييم المعرفة والأداء.

تشخيص الأجسام ثلاثية الأبعاد مشكلة صعبة للغاية. باستخدام وجهات النظر ثنائية الأبعاد ذات المراجع في قاعدة البيانات إلى كائنات. والعديد من المشاكل في هذا المجال. على وجه الخصوص، عندما يزيد عدد الكائنات، ويزيد حجم قاعدة البيانات ويصبح من الصعب على نحو متزايد التعرف على الأشياء بشكل صحيح لأن المعلومات لم تعد قادرة على استيعاب ماهية الكائنات.

لتجنب هذه المشاكل باستخدام مصادر المعلومات بخلاف طرق العرض ثنائية الأبعاد وصور الكائنات ثلاثية الأبعاد. تأتي هذه المعلومات الإضافية من بنية الكائن، أو من سياقه المعتاد، وهذا هو السبب في أن الفكرة المعمول بها في المحور الثالث كانت إجراء استخراج شبكات ثلاثية الأبعاد من الكائنات في عرض ثنائي الأبعاد استنادًا إلى أسطح ومساحات بيديه للحصول على شبكة جيدة خاصة بكل كائن لتسهيل استخراج مهام التشخيص، وإعادة بناء نماذج كائنات ثلاثية الأبعاد أو البحث عن كائنات في قاعدة البيانات عن طريق البحث عن شبكاتهم ... الخ

## الكلمات الدالة

" التنعيف والترقيق ". الكائن ثنائي الأبعاد / ثلاثي الأبعاد، الهيكل العظمي. الاعتراف، كشف، كشف عن نقطة اهتمام. مساحة بيديه، شبكة، استخراج، الشكل الحر للكائن.



# Table de matières

---

---

Remerciements .....	5
Résumé .....	6
Abstract.....	7
ملخص.....	8
Table de matières .....	9
Table des figures.....	14
Liste des tableaux .....	17
Liste des abréviations.....	18
Liste des publications.....	19
<b>1 Introduction générale .....</b>	<b>21</b>
1.1 <i>Problématique et Difficulté</i> .....	21
1.2 <i>Organisation du manuscrit</i> .....	23
<b>2 Chapitre 1 : Généralités.....</b>	<b>26</b>
2.1 <i>Introduction</i> .....	26
2.2 <i>Quelques exemples de travaux de reconnaissance d'objets (Contexte Historique)</i> .....	27
2.2.1 Approches basées sur l'étude et l'extraction des caractéristiques .....	27
2.2.1.1 Reconnaissance d'objets 3D par points d'intérêt.....	27
2.2.1.2 Reconnaissance d'objets 3D par leurs caractéristiques clés .....	28
2.2.1.3 Reconnaissance active d'objets 3D de forme libre par Vision en Robotique .....	28
2.2.1.4 Reconnaissance active d'objets représentés par la Couleur.....	28
2.2.1.5 Reconnaissance et récupération d'objets par la correspondance du squelette graphique .....	29
2.2.2 Approches basées sur la géométrie .....	29
2.2.2.1 Reconnaissance d'objets 3D à partir des images 2D en utilisant le hachage géométrique.....	29
2.2.2.2 Reconnaissance d'objets à l'ère géométrique .....	29
2.3 <i>Architecture générale d'un système de Reconnaissance</i> .....	30
2.3.1 Introduction et Procédure .....	30
2.3.1.1 Représentation d'objet ou de modèle .....	31
2.3.1.2 Extraction de caractéristiques .....	31
2.3.1.3 Correspondance caractéristiques-modèle .....	32
2.3.1.4 Formation d'hypothèses .....	32
2.3.1.5 Vérification d'objet.....	32
2.3.1.6 Complexité de la reconnaissance d'objet .....	32
2.3.2 Aperçu général sur les phases d'un système de reconnaissance .....	34
2.3.2.1 Préparation des données .....	34
2.3.2.1.1 Numérisation.....	34

2.3.2.1.2	Prétraitement.....	34
2.3.2.1.3	Calcul des représentations ou extraction des caractéristiques.....	35
2.3.2.2	Apprentissage.....	35
2.3.2.3	Classification.....	35
2.3.2.4	Post-traitement.....	35
2.3.3	Conclusion.....	36
<b>2.4</b>	<b><i>Prétraitement et Représentation des caractéristiques</i></b> .....	<b>36</b>
2.4.1	A propos du Prétraitement.....	36
2.4.2	Prétraitement, Description et Extraction des primitives visuelles.....	37
2.4.2.1	Description des images.....	37
2.4.2.1.1	Image 2D.....	37
2.4.2.1.2	Image 2.5D.....	38
2.4.2.1.3	Image 3D.....	39
2.4.2.2	Représentation des objets.....	40
2.4.2.2.1	Représentations centrées sur l'observateur.....	41
2.4.2.2.2	Représentations centrées sur les objets.....	41
2.4.2.3	Traitement ponctuel.....	41
2.4.2.3.1	Techniques de seuillage.....	42
2.4.2.3.1.1	Seuillage global.....	42
2.4.2.3.1.2	Seuillage local.....	43
2.4.2.3.2	Traitement d'histogramme.....	44
2.4.2.4	Réduction du bruit.....	45
2.4.2.5	Segmentation.....	46
2.4.2.5.1	Segmentation par régions.....	47
2.4.2.5.2	Segmentation par contour.....	48
2.4.2.5.3	Segmentation par les points d'intérêt (Etat de l'art).....	51
2.4.2.5.3.1	Différentes approches.....	52
2.4.2.5.3.2	Détecteurs 2D.....	53
2.4.2.5.3.2.1	Détecteur de coins.....	53
2.4.2.5.3.2.2	Détecteur de régions.....	55
2.4.2.5.3.2.3	Autres détecteurs.....	57
2.4.2.5.3.3	Détecteurs 3D.....	57
2.4.2.6	Opérations morphologiques.....	60
2.4.2.6.1	Érosion et Dilatation.....	61
2.4.2.6.2	Ouverture et Fermeture.....	61
2.4.2.6.3	Amincissement et squelettisation (Etat de l'art).....	62
2.4.2.6.3.1	Squelettisation 2D.....	63
2.4.2.6.3.1.1	Algorithme d'amincissement « Thinning 2D ».....	63
2.4.2.6.3.1.2	Simulation de la propagation d'une onde « feu de prairie ».....	63

2.4.2.6.3.1.3	Squelette à partir de la carte des distances .....	64
2.4.2.6.3.1.4	Squelette à partir du diagramme de Voronoï.....	64
2.4.2.6.3.2	Squelettisation 3D.....	65
2.4.2.6.3.2.1	Amincissement et propagation en bords « Thinning 3D » .....	65
2.4.2.6.3.2.2	Utilisation d'un champ de distance .....	66
2.4.2.6.3.2.3	Méthodes géométriques.....	67
2.4.2.6.3.2.4	Fonctions générales sur le terrain .....	68
2.4.3	Généralité sur la représentation des objets par Maillage .....	68
2.4.3.1	Introduction .....	68
2.4.3.2	Générateurs de maillage en 2D et 3D.....	69
2.4.3.2.1	Méthode de « transport par projection » .....	69
2.4.3.2.2	Méthode « frontale 2D/3D » .....	70
2.4.3.2.3	Méthode du type « Voronoi 2D/3D » .....	70
2.4.3.2.4	Méthode basée sur la notion de « quadtree » .....	71
2.4.3.3	Générateurs de maillage en 3D.....	71
2.4.3.3.1	Méthode de type « différence finie » .....	71
2.4.3.3.2	Méthode de « translation-empilement ».....	72
2.4.3.3.3	Méthode par blocs avec « transport projection » .....	72
2.4.3.3.4	Méthode basée sur la notion « d'octree » .....	73
2.4.3.3.5	Mailleur de surface dans l'espace.....	73
2.4.3.4	Conclusion.....	73
<b>2.5</b>	<b><i>Description et Extraction des caractéristiques.....</i></b>	<b>73</b>
2.5.1.1	Caractéristiques globales .....	74
2.5.1.2	Caractéristiques locales .....	74
2.5.1.3	Description statistique .....	75
2.5.1.3.1	Moments .....	75
2.5.1.3.2	Histogrammes .....	75
2.5.1.3.3	Transformée de Fourier.....	76
2.5.1.3.4	SIFT & SURF .....	76
2.5.1.4	Description structurelle .....	77
2.5.1.4.1	Extraction de structures topologiques .....	77
2.5.1.4.2	Codage .....	78
2.5.1.4.3	Représentation par les graphes.....	78
<b>2.6</b>	<b><i>Conclusion .....</i></b>	<b>78</b>
<b>3</b>	<b>Chapitre 2 : Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes.....</b>	<b>81</b>
<b>3.1</b>	<b><i>Notions de Base.....</i></b>	<b>81</b>
3.1.1	Introduction à La squelettisation / L'amincissement.....	81
3.1.2	Définitions et propriétés .....	82

3.1.2.1	Définitions .....	82
3.1.2.2	Propriétés.....	83
3.1.3	Amincissement vs Axe médiane .....	85
<b>3.2</b>	<b><i>Approches de l'axe médian vs l'algorithme Thinning.....</i></b>	<b>86</b>
3.2.1	Classes des algorithmes de squelettisation .....	86
3.2.1.1	Algorithmes basés sur des masques.....	87
3.2.1.2	Algorithmes basés sur les frontières .....	87
3.2.2	Principe de l'Axe médiane .....	88
3.2.3	Elagage du squelette.....	89
3.2.4	Principe de l'algorithme Thinning « Amincissement » .....	90
3.2.4.1	Amincissement et la propagation des bords .....	90
3.2.4.2	Thinning sur les objets voxels .....	92
3.2.4.2.1	Principe .....	92
3.2.4.2.2	Implémentations et Résultats d'algorithme « Thinning » sur les objets 3D.....	92
3.2.4.2.3	Interprétations et observations .....	94
3.2.5	Approche de l'algorithme Thinning développée sur les objets 3D en vue 2D .....	95
3.2.5.1	Démarche de notre algorithme .....	96
3.2.6	Résultats et Discussion.....	97
3.2.6.1	Complexité temporelle .....	101
<b>3.3</b>	<b><i>Conclusion .....</i></b>	<b>103</b>
<b>4</b>	<b>Chapitre 3 : Représentation des formes libres des objets 2D/3D en vue 2D par l'extraction des P.Is .....</b>	<b>143</b>
<b>4.1</b>	<b><i>Introduction .....</i></b>	<b>143</b>
<b>4.2</b>	<b><i>Notions de base sur les points d'intérêt (PIs).....</i></b>	<b>144</b>
4.2.1	Définition et différents types de points d'intérêt.....	145
4.2.2	Détection des Coins.....	145
<b>4.3</b>	<b><i>Détecteur SUSAN de Smith .....</i></b>	<b>146</b>
4.3.1	Calcul de l'USAN.....	147
4.3.2	Comment trouver les pixels de bordure ?.....	149
<b>4.4</b>	<b><i>Nouvelle approche développée.....</i></b>	<b>150</b>
4.4.1	Choix de la nouvelle approche du détecteur SUSAN ?.....	150
4.4.1.1	Méthodologie de la nouvelle approche.....	151
4.4.1.2	Nouveau détecteur robuste au bruit .....	152
4.4.1.3	Organigramme de l'approche proposée.....	153
4.4.1.4	Modèle 2D/3D Utilisé .....	153
<b>4.5</b>	<b><i>Résultats et Discussion .....</i></b>	<b>154</b>
4.5.1	Objets 3D de la base de Benchmark.....	154
4.5.2	Objets 3D du logiciel « Sweet Home 3D » .....	156

4.5.3	Objets 3D Bruités en vue 2D.....	158
<b>4.6</b>	<b><i>Interprétation et évaluation</i></b> .....	<b>160</b>
4.6.1	Interprétation et complexité .....	160
<b>4.7</b>	<b><i>Conclusion</i></b> .....	<b>162</b>
<b>5</b>	<b>Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier</b> .....	<b>163</b>
<b>5.1</b>	<b><i>Introduction</i></b> .....	<b>163</b>
<b>5.2</b>	<b><i>Notions de Base &amp; Création Du Maillages 3D Adapté à L'objet</i></b> .....	<b>164</b>
5.2.1	Définition d'un maillage en 3D.....	164
5.2.2	Représentation des maillages par les surfaces de Bezier.....	165
5.2.2.1	Courbes de Bezier.....	165
5.2.2.2	Surfaces de Bezier .....	167
5.2.2.3	Subdivision triangulaire des surfaces de Bezier guide à un maillage fine et lisse : .....	170
<b>5.3</b>	<b><i>Pré-traitement</i></b> .....	<b>172</b>
5.3.1	Centrage .....	173
5.3.2	Mise à l'échelle .....	174
5.3.3	Alignement.....	174
<b>5.4</b>	<b><i>Approche proposée</i></b> .....	<b>175</b>
5.4.1	Présentation de la méthode proposée et implémentation.....	175
5.4.2	Organigramme de l'algorithme proposé.....	178
5.4.3	Résultats et discussions .....	178
<b>5.5</b>	<b><i>Conclusion</i></b> .....	<b>181</b>
<b>6</b>	<b>Conclusion générale et perspectives</b> .....	<b>182</b>
<b>6.1</b>	<b><i>Conclusion générale</i></b> .....	<b>182</b>
<b>6.2</b>	<b><i>Perspectives</i></b> .....	<b>183</b>
	<b>Références</b> .....	<b>184</b>

## Table des figures

---

Figure 2-1: Image (a) : Niveau de gris ; Image (b) : 3D PIs détectés .....	28
Figure 2-2: Différents composants d'un système de reconnaissance d'objets.....	30
Figure 2-3: Exemple d'une image 2D.....	38
Figure 2-4: Exemple d'une projection 3D à un plan 2D pour dessiner une image de 2.5D.....	38
Figure 2-5: 2D vers 2.5D .....	39
Figure 2-6: Exemple d'une image 3D sous l'effet de l'éclairage et de la texture dans un espace 3D (Logiciel « Sweet Home 3D ») .....	40
Figure 2-7: Méthode d'Otsu [15] .....	43
Figure 2-8: Méthode de seuillage par Hystérésis[20].....	44
Figure 2-9: Exemple de traitement histogramme (égalisation d'histogramme).....	45
Figure 2-10: Réduction du bruit sous les filtres Moyenneur & Gaussien dans un espace obscuré .....	46
Figure 2-11: Réduction du bruit sous les filtres Moyenneur & Gaussien dans un espace lumineux.....	46
Figure 2-12: Résultat d'un algorithme de segmentation en régions.....	47
Figure 2-13: Distributions des types de régions en fonction des deux plus grandes valeurs propres.....	53
Figure 2-14: Exemple de masques d'analyse du détecteur SUSAN (haut), permettant le calcul des zones USAN (bas).....	54
Figure 2-15: Détecteur de SUSAN - Illustration du principe du détecteur de SUSAN.....	55
Figure 2-16: Extraction de PIs par le détecteur-descripteur SIFT .....	56
Figure 2-17: Exemple de quelques algorithmes de détection des points d'intérêt sur une image 2D .....	57
Figure 2-18: Image (a) : le descripteur SIFT 2D normal. Image (b) : le descripteur SIFT 2D. Image (c) : le descripteur SIFT 3D avec ses sous-volumes 3D. [57].....	58
Figure 2-19: Points d'intérêt détectés par points saillants(a),3D-Harris(b), 3D-SIFT (c), SD-corners (d), et HKS (e) [63].....	60
Figure 2-20: Exemple d'érosion et dilatation d'un objet 3d en vue 2d en niveau de gris .....	61
Figure 2-21: Exemple d'ouverture et de fermeture d'un objet 3d en vue 2d en niveau de gris .....	62
Figure 2-22: Partition 2D et construction 3D associée.....	72
Figure 2-23: Exemple de détection par les descripteurs locaux SIFT et SURF .....	76
Figure 3-1: Exemple de squelette [129] .....	82
Figure 3-2: Différence entre la squelettisation et l'algorithme Thinning [80] .....	85
Figure 3-3: Objet et leur squelette selon les types d'anneau.....	86
Figure 3-4: (a et c) Axe médian en 2D;(b) Surface médiane en 3D avec un disque inscrits (2D) et une boule (3D). .....	88
Figure 3-5: Exemples de courbes squelettes de différentes objets 3D [131].....	89
Figure 3-6: Généralisation de l'algorithme Thinning.....	92
Figure 3-7: Méthode Thinning appliqué au voxel-temps d'exécution 152.508498 seconds.....	93
Figure 3-8: Méthode répulsive potentiel Field appliqué au voxel[102] .....	93

Figure 3-9: Méthode Distance Field appliqué au voxel [93].....	94
Figure 3-10: Masque 1 .....	95
Figure 3-11: Masque 2 .....	95
Figure 3-12: Démarche de notre Algorithme .....	97
Figure 3-13: Obtention d'un squelette d'un cheval après le passage par quelques itérations par le masque 1 et 2 .....	97
Figure 3-14: Obtention des squelettes à partir des objets 3D en appliquant notre nouvelle approche .....	98
Figure 3-15: Obtention des squelettes à partir des objets 3D voxels.....	99
Figure 3-16: Comparaison entre le nouveau thinning(approche proposée) et l'ancien thinning(original thinning- fonction Matlab-).....	100
Figure 3-17: Comparaison au niveau du temps d'exécution .....	101
Figure 4-1: Différents types de points d'intérêt : coin simple, jonction en 'V', jonction en 'T', jonction en 'L', jonction en 'damier'[33].....	145
Figure 4-2: Etapes qui constituent l'algorithme de détection de contour SUSAN.....	147
Figure 4-3: Zone USAN des masques circulaires de la figure 4-2 est de couleur blanche.....	148
Figure 4-4: Différence entre un noyau sur un bord et pas sur un bord .....	148
Figure 4-5: Carte des bords résultante avec l'image d'entrée correspondante [136].....	149
Figure 4-6: Organigramme du nouveau détecteur de SUSAN (NDS) .....	153
Figure 4-7: Différence entre la détection d'objet 3D par l'ancien détecteur de SUSAN (DS) et le nouveau détecteur SUSAN (NDS).....	155
Figure 4-8: Différence entre la détection d'objet 3D-BD de Benchmark- par l'ancien détecteur de SUSAN (DS) et le nouveau détecteur SUSAN (NDS).....	155
Figure 4-9: Différence entre la détection des objets 3D - BD de Sweet Home 3D- par l'ancien (DS) et le nouveau détecteur de SUSAN(NDS). « Sweet Home 3D » .....	156
Figure 4-10: Différence entre la détection des objets 3D par Le détecteur SUSAN (DS) et le nouveau détecteur SUSAN (NDS) « Sweet Home 3D » .....	157
Figure 4-11: Différence entre la détection des objets 3D bruités par l'ancien (DS) et le nouveau détecteur de SUSAN(NDS) .....	158
Figure 4-12: Différence entre la détection d'objets 3D bruités par (DS) et le nouveau détecteur SUSAN (NDS) .....	159
Figure 4-13: Comparaison de temps d'exécution entre DS et NDS en fonction de Taille d'image testée .....	161
Figure 5-1: Courbe de Bézier avec 5 points de contrôle .....	165
Figure 5-2: Construction d'une courbe Bezier à 3 points pour $t=2/3$ .....	166
Figure 5-3: Exemples de courbes Bezier.....	166
Figure 5-4 : Construction d'une courbe Bezier à 5 points pour $t=1/3$ (« Construction de Casteljau »).....	166
Figure 5-5: Surface de Bezier .....	167
Figure 5-6: Surface de Bezier : construction par l'algorithme de de Casteljau (a) et (b).....	168
Figure 5-7: A gauche un polygone $P_i$ de contrôle et à droite un polyèdre de contrôle $P_{ij}$ .....	168
Figure 5-8: Illustration de l'algorithme pour évaluer $B(u;v)$ .....	169

Figure 5-9 : Produit tensoriel de deux courbes de degré 3 (ordre 4) .....	170
Figure 5-10: "Gumbo" modèle formé de patch bicubiques et crée par Edwin Catmull [144].....	170
Figure 5-12: Polyèdre $\{P_{i,j,k}\}$ et patch correspondant.....	171
Figure 5-11: Etapes de la subdivision triangulaire de Bezier [145] .....	171
Figure 5-13: Exemple de Bezier triangulaire [145].....	172
Figure 5-14: Exemple d'objets déformées : objet d'origine (a), objet translaté (b). Objet ayant subi un changement d'échelle (c), objet ayant subi une rotation (d). [146] .....	173
Figure 5-15: Exemple d'un objet mis à l'échelle, objet d'origine (a) et objet intégré (b) [146] .....	174
Figure 5-16: Etapes de la subdivision de maillage selon notre implémentation .....	176
Figure 5-17: Exemples de maillages extraits de notre algorithme .....	177
Figure 5-18: Organigramme de l'algorithme proposé.....	178
Figure 5-19: Maillages extraits des objets 3D en vue 2D selon notre implémentation .....	179
Figure 5-20: Comparaison entre les maillages des objets extraits au niveau des données et de temps d'exécution .....	180
Figure 5-21: Capture plus proche des sommets et des faces d'un maillage extrait d'après notre algorithme.....	181



## Liste des tableaux

---

---

Tableau 2-1: Quelques exemples de détection de contours et leurs propriétés .....	48
Tableau 2-2: Comparaison entre les images 2D et 2.5D en appliquant la segmentation par contour actif .....	50
Tableau 3-1: Comparaison au niveau du temps d'exécution.....	101
Tableau 3-2: Comparaison entre deux grandeurs du temps nécessaire à l'exécution d'un algorithme d'un type de complexité [133].....	102
Tableau 4-1: Comparaison de temps d'exécution entre DS et NDS en fonction de Taille d'image testée.....	160
Tableau 5-1: Comparaison entre les maillages des objets extraits au niveau des données et de temps d'exécution .....	180

## Liste des abréviations

---

---

<b>SRO</b>	<b>Système de Reconnaissance d'Objets</b>
<b>CAO</b>	<b>Conception Assistée par Ordinateur</b>
<b>SVM</b>	<b>Machine à Support de Vecteurs</b>
<b>KPPV</b>	<b>K Plus Proches Voisins</b>
<b>3D</b>	<b>Tridimensionnel</b>
<b>2.5D</b>	<b>2D et demi ou pseudo 3D</b>
<b>2D</b>	<b>Bidimensionnel</b>
<b>PIs</b>	<b>Points d'Intérêts</b>
<b>PTZ</b>	<b>Prêt à Taux Zéro (type de cam)</b>
<b>SUSAN</b>	<b>Smallest Univalve Segment Assimilating Nucleus</b>
<b>SIFT</b>	<b>Scale Invariant Feature Detector</b>
<b>SURF</b>	<b>Speeded Up Robust Feature Detector</b>
<b>GLOH</b>	<b>Gradient Location-Orientation Histogram</b>
<b>PDF</b>	<b>Fonction de Densité de Probabilité</b>
<b>CSS</b>	<b>Curvature Scale Space</b>
<b>DOG</b>	<b>Fonction de Différence de Gaussiennes</b>
<b>ACP</b>	<b>Analyse en Composantes Principales</b>
<b>DFs</b>	<b>Descripteurs de Fourier</b>
<b>NDG</b>	<b>Niveau de Gris</b>
<b>DS</b>	<b>Détecteur SUSAN</b>
<b>NDS</b>	<b>Nouveau Détecteur SUSAN</b>

## Liste des publications

---

---

### Contribution

Pour ce présent travail, nous avons élaboré trois approches qui sont incluent dans les premières phases d'un système de reconnaissance d'objet et plus précisément dans la phase du prétraitement et représentation des primitives visuelles. Les résultats de notre travail ont donné lieu aux publication et communications suivantes :

#### ❖ Publications :

- Agnaou, I. & Bouikhalene, B. (2017). New Approach for 3D Object Forms Detection Using a New Algorithm of SUSAN Descriptor. *The International Arab Journal of Information Technology (IAJIT)*, Vol. 14, No. 4A, Special Issue,614-623.
- Agnaou, I., Maarir, A. & Bouikhalene, B. (2014). Extraction of curves skeletons making a comparative study between 3D voxel objects and simple 3D objects using 2D view by applying a new approach of Thinning algorithm. *International Journal of Innovation and Applied Studies (IJIAS)*, ISSN 2028-9324, Vol. 7 No 2, 464-471.

#### ❖ Article soumis pour publication :

- Agnaou, I & Bouikhalene, B. (06/2020) Extraction of Free Shapes from Three-Dimensional Objects in 2D View Based on Bezier's Surfaces. *The Visual Computer Journal, Springer - Editorial Manager (TV CJ)*

#### ❖ Communications :

- Agnaou, I. & Bouikhalene, B. (2016). New approach for 3d object forms detection using a new algorithm of SUSAN descriptor. *The 17th International Arab Conference on Information Technology (ACIT'16)*. FST Béni Mellal, Maroc.
- Agnaou, I., Maarir, A. & Bouikhalene, B. (2014). Extraction of skeleton curves by a new approach of Thinning algorithm applied on 3D object via 2D view. *Journées Doctorales en Technologies de l'information et de la Communication (JDTIC'14)*, ENSIAS, Rabat, Maroc.

- Agnaou, I., Maarir, A., Abou-Bakre, A. & Bouikhalene, B. (2014). Application of the Laplacian Pyramid on the 3D objects via 2D view from the RGB space to HSI space. *1<sup>er</sup> Journées Doctorales sous le thème « Recherche Scientifique : Innovation et développement »*. FST Béni Mellal, Morocco.
  
- Agnaou, I., Zeghal, A. & Bouikhalene, B. (2014). Segmentation of 3D objects using 2D view by applying the active contour. In the *First International Conference on Business Intelligence (CBI'14)*. FST Béni Mellal, Morocco.

# 1 Introduction générale

---

---

## 1.1 Problématique et Difficulté

---

La reconnaissance d'objets est l'une des capacités les plus fascinantes que les humains possèdent facilement depuis leur enfance. D'un simple regard sur un objet, l'être humain est capable de dire son identité ou sa catégorie malgré la variation d'apparence, de plus, en raison d'un changement de pose, d'éclairage, de texture, de déformation et d'occlusion, l'être humain peut facilement de donner de l'observation d'un ensemble d'objets à la reconnaissance d'objets qui n'ont jamais été vus auparavant. Par exemple, les enfants sont capables de généraliser le concept de "chaise" ou de "tasse" après avoir vu seulement quelques exemples.

Néanmoins, il s'agit d'une tâche considérable de développer des systèmes de vision qui correspondent aux capacités cognitives des êtres humains, ou des systèmes qui sont à la fois capable de dire l'identité spécifique d'un objet observé. Les principales raisons peuvent être attribuées aux facteurs suivants : la pose relative d'un objet par rapport à une caméra, les variations d'éclairage et la difficulté de généraliser à travers les objets à partir d'un ensemble d'images exemplaires.

La régularité des images est le cœur des systèmes de reconnaissance d'objets, prises sous différentes conditions d'éclairage et de pose, sont extraites et reconnues. En d'autres termes, tous les algorithmes étudiés adoptent certaines représentations ou modèles pour capturer ces caractéristiques, facilitant ainsi les procédures pour déterminer leur identité. De plus, les représentations peuvent être des modèles géométriques 2D ou 3D. Le processus de reconnaissance, qu'il soit génératif ou discriminatoire, s'effectue ensuite par comparaison de l'image de test avec les représentations ou modèles d'objets stockés.

C'est pour cela, la reconnaissance d'objets 2D/3D est l'un des principaux objectifs des applications de vision par ordinateur destinées à fonctionner dans des environnements réels comme le transport intelligent, à la surveillance de la sécurité, l'interaction..., etc. Cependant, les méthodes de reconnaissance d'objets sont généralement complexes parce qu'elles impliquent la segmentation, la représentation d'objets, l'extraction de caractéristiques et l'appariement d'une grande quantité de données. Par conséquent, la plupart d'entre eux travaillent hors ligne ou traitent avec une base de données limitée et déjà apprise.

## Introduction générale

Les objets 3D peuvent être reconnus par l'étude d'un ensemble des images 2D correspondant à l'objet à partir de différents points de vue, et bien que de nombreux algorithmes de reconnaissance d'objets aient été proposés, la plupart d'entre eux sont conçus pour la reconnaissance d'objets 2D. Ainsi, les chercheurs en vision par ordinateur accordent plus d'attention à la reconnaissance d'objets 3D, mais ils étendent directement les algorithmes de reconnaissance d'objets 2D à la reconnaissance d'objets 3D sans tenir compte des caractéristiques des objets 3D, c'est pour cela la reconnaissance d'objet 3D reste un sujet difficile et non totalement résolu, c'est le but dont on va utiliser dans les prochains chapitres de représentation et d'extraction des caractéristiques des objets les images 2D et demi ou pseudo 3D «2.5D» qui sont entre 2D et 3D c'est-à-dire que nous allons travailler avec des objets 3D en vue 2D pour faciliter la tâche de la représentation des objets et la reconnaissance de leurs formes, et automatiquement on simplifie un peu la tâche de la reconnaissance des objets ou des formes d'objets en exploitant notre approches résultantes pour compléter les autres étapes d'un SRO. Ainsi, pour alléger cette tâche de reconnaissance d'objet qui est décomposé aux plusieurs phases, on va se concentrer dans cette thèse sur la première et la deuxième phase d'un système de reconnaissance qui est fondé sur le prétraitement, la représentation et l'extraction des primitives soient visuelles ou locales. Généralement ,le problème de la reconnaissance des objets est étroitement lié au problème de la segmentation, cette dernière fait partie du processus de prétraitement et d'extraction de l'information, qui est une condition préalable à toute reconnaissance, puisque la segmentation par les points d'intérêts est une part parmi plusieurs approches de segmentation ,on va s'intéresser à cette dernière pour qu'on puisse décrire et représenter un objet par sa forme tout en présentant d'une nouvelle approche dont le but est d'isoler les objets 3D en vue 2D de leur arrière-plan pour les segmenter en appliquant d'autre opérations de prétraitement afin de préparer les données à l'autre phase suivante. Les résultats de la segmentation peuvent entraîner la suppression de certains pixels qui produisent des trous dans certaines parties des images ; Cela pourrait être vu dans des objets dedans les images 2.5D ayant certains trous causés par la suppression de quelques pixels qui peut provoquer des ruptures et des déformations de l'objet traité ,ainsi , on peut trouver des impuretés sans utilité dans l'objet segmenté et c'est le cas du traditionnel algorithme d'amincissement « Thinning » qui résulte de genre de squelette avec ces inconvénients ,c'est pour ça ,nous allons faire une amélioration à cet algorithme afin d'obtenir et extraire un squelette bien défini et bien structuré sans impuretés ou extensions inutiles des objets 3D en vue 2D et cette étape s'intègre dans les opérations morphologiques dont le but est de bien extraire et filtrer le squelette et la forme de l'objet à traité. Du fait que nous avons travaillé dans un milieu continu dans toute ces opérations précédentes, on peut le discrétiser spatialement pour créer une

modélisation géométrique et construire un modèle virtuel d'un objet qui est le maillage (soit 2D ou 3D) dont le but est d'effectuer une simplification d'un système par un modèle représentant ce système et la forme de l'objet extrait en général, ainsi, pour cette raison nous y allons proposer une nouvelle méthode dont l'objectif est détecter l'objet sélectionné puis créer et extraire son maillage en utilisant les surfaces de Beziers triangulaires afin d'obtenir des représentations fines et exactes ,ce qui rend la reconnaissance des formes d'objets et leurs recherches aussi dans une base de données par ces maillages plus facile.

Dans notre travail nous allons utiliser une image 2D améliorée et une image 3D simulée c'est-à-dire que nous allons travailler par des images 2.5D (dimension entre 2D et 3D, appelé aussi : pseudo 3D comme le mentionne l'article anglais de Wikipédia [1]) afin d'utiliser des objets 3D en vue 2D et réduire le temps de calcul ,ainsi, qu'en profitant des propriétés et des caractéristiques des images 2D (bidimensionnelles) par rapport au tridimensionnelles qui ralentissent le temps de calcul en utilisant autres caractéristiques tridimensionnelles plus complexes ,aussi ,pour donner l'impression au visionneur qu'on a travaillé sur un plan 3D.

## 1.2 Organisation du manuscrit

---

La suite du manuscrit est organisée selon deux parties :

- La première partie présente deux choses : un état d'art sur les méthodes utilisées dans quelques systèmes de reconnaissance d'objet et l'architecture générale d'un système de reconnaissance d'objet. Cette partie est constituée par un seul chapitre qui contient les 4 sections suivantes :
  - **Chapitre 1**
    - **Section 2.2 et 2.3** : la section 2.2 décrit les techniques utilisées dans un SRO en présentant des extraits du contexte historiques sur ces méthodes, puis, dans la section 2.3 on a abordé un aperçu général sur le principe, les aspects et l'architecture d'un système de reconnaissance d'objets, ainsi que les différentes phases d'un SRO pour qu'on puisse se concentrer sur les premières phases qui sont :le prétraitement, la description et la représentation des caractéristiques dans les sections suivantes 2.4 et 2.5.
    - **Section 2.4 et 2.5** : Dans la première section 2.4, nous avons donné une telle description et représentation des images et des objets, puis nous avons spécifier toutes les étapes concernant le prétraitement dont on va l'appliquer sur les images

2.5D en donnant un aperçu sur l'état de l'art de chaque approche à améliorer ou à créer dans les chapitres (3-4-5). Ainsi de suite, dans la section 2.5, nous avons présenté une description statique et structurelle sur les méthodes d'extraction des caractéristiques et qui sont lié aux méthodes décrites dans l'état de l'art dans la section 2.4 et les chapitres (3-4-5) ci-dessous.

- La deuxième partie présente notre contribution et qui est constituée par trois chapitres :
  - o **Chapitre 3** : Ce chapitre présente une représentation des formes d'objets 3D en vue 2D par l'extraction de leurs squelettes et décrit les expériences réalisées.
  - o **Chapitre 4** : Ce chapitre est consacré à la représentation des formes libres des objets 2D/3D en vue 2D par l'extraction des points d'intérêts (PIs) tout en faisant une implémentation et comparaison entre les deux pour qu'on puisse conclure le degré de performance de l'approche proposée.
  - o **Chapitre 5** : Ce chapitre est basé sur la création et la représentation des maillages des objets 3D en vue 2D en utilisant les surfaces de Beziers pour les exploiter dans les systèmes de reconnaissances des formes d'objets ou à la recherche des objets par leurs maillages dans une base de données (comme perspective), etc...

Dans la conclusion de cette thèse, nous présentons un bref aperçu sur nos contributions tout en proposant de multiples perspectives de recherche concernant la reconnaissance des objets étudiés tout en profitant des méthodes proposées afin de faciliter les tâches d'un SRO qui suivent les tâches étudiées dans ce papier en faisant des tests expérimentaux, ainsi, pour obtenir un bon et fiable système de reconnaissance et faciliter la recherche et la modélisation des objets.



---

PREMIÈRE PARTIE :  
ÉTUDES BIBLIOGRAPHIQUES et  
PRELIMINAIRES

---

## 2 Chapitre 1 : Généralités

---

---

### 2.1 Introduction

---

La reconnaissance d'objets est un domaine d'application de base en vision par ordinateur. Depuis de nombreuses années, il est considéré comme un domaine de recherche approfondie, en particulier dans le domaine de la 2D et la 3D.

La reconnaissance d'objets peut être définie comme une tâche d'identifier des objets dans le monde réel à partir d'une image ou d'une séquence vidéo, en utilisant des modèles d'objets qui sont connus a priori. C'est à dire que le problème de reconnaissance d'objet peut être défini comme un problème d'étiquetage à partir de modèles d'objets connus.

Etant donné une image contenant un ou plusieurs objets d'intérêt et un ensemble d'étiquettes correspondant à un ensemble de modèles connus du système, le système devrait attribuer des étiquettes correctes aux régions, ou à un ensemble de régions de l'image afin de faciliter la reconnaissance de chaque objet trouvé dans l'image ou dans la scène en général.

Ainsi, les systèmes de reconnaissance d'objets bidimensionnels ou tridimensionnels donnent une définition de haut niveau d'un objet à partir des données détectées qui y sont représentées.

A partir de là, le système peut prendre de diverses décisions telles que l'identification de l'objet à partir d'une base de données, l'estimation de la pose de l'objet et l'apprentissage de nouveaux objets à partir des données image...etc., toutes ces tâches doivent passer par des phases successives pour bien créer un fiable système de reconnaissance.

Dans ce chapitre, nous abordons les aspects fondamentaux de la reconnaissance d'objets, nous présentons quelques anciens exemples de systèmes de reconnaissance (**section 2.2**) et nous allons donner un petit rappel sur l'architecture et les principaux composants d'un système de reconnaissance d'objets en discutant leurs rôles et en détaillant toutes les phases qui le constituent (**section 2.3**).

L'analyse d'images a pour but de décrire de façon quantitative ou qualitative les objets présents dans les images, ce qui nous donne une transformation de l'information de bas niveau issue de la phase d'acquisition en une information de haut niveau où les formes et les structures sont décrites de façon synthétique. Les grandes étapes de ce processus sont le prétraitement, la segmentation, la description et l'interprétation (**section 2.4 et 2.5**). C'est pour cela, on va se focaliser dans le reste du chapitre sur la phase du prétraitement pour bien préparer et traiter les données d'un système SRO afin de réaliser leur

reconnaissance ; en disant c'est la tâche principale qui engendre notre travail et plus précisément nous nous sommes intéressés à la représentation et d'analyse des objets 3D en vue 2D et de leur description et extraction des caractéristiques, afin de les préparer aux autres phases en rappelons d'un état de l'art sur les approches développées dans notre thèse , et qui s'intègrent exactement dans la phase du prétraitement et de la représentation des formes tout en se basant sur :la détection par les points d'intérêt, la squelettisation et la création des maillages et toutes ces tâches seront représentées dans une vue 2D en utilisant des images 2.5D ,puis on va finir par une vue générale de l'état de l'art sur les méthodes d'extraction des caractéristiques selon notre sujet.

## **2.2 Quelques exemples de travaux de reconnaissance d'objets (Contexte Historique)**

---

Dans la section suivante, nous allons présenter deux différents types d'approches de reconnaissance qui sont des extraits des algorithmes et des méthodes créés et qui sont inclus dans le même principe de reconnaissance des objets afin de savoir les approches proposées pour un SRO. Alors, pour en savoir plus, voir ci-dessous :

### **2.2.1 Approches basées sur l'étude et l'extraction des caractéristiques**

#### **2.2.1.1 Reconnaissance d'objets 3D par points d'intérêt**

Dans le contexte de reconnaître un objet 3D isolé donné dans une vue requête à partir d'une base d'apprentissage contenant quelques vues de cet objet. L'objectif de Ayet Shaiek [2] est de formuler une méthodologie locale qui combine des aspects d'approches existantes et apporte une amélioration sur la performance de la reconnaissance. Il a opté pour une méthode par points d'intérêt (PIs) fondée sur des mesures de la variation locale de la forme et la sélection de points saillants qui est basée sur la combinaison de deux espaces de classification de surfaces : l'espace SC (indice de forme-intensité de courbure), et l'espace HK (courbure moyenne courbure gaussienne). La figure 2-1 illustre la sélection des PIs sur objet 3D.

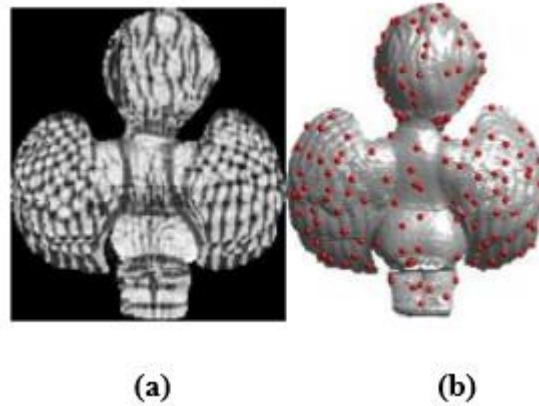


Figure 2-1: Image (a) : Niveau de gris ; Image (b) : 3D PIs détectés

### 2.2.1.2 Reconnaissance d'objets 3D par leurs caractéristiques clés

Laurent Desmecht et Marcel Rémon [3] ont fait un développement d'un nouveau modèle mathématique pour les objets tridimensionnels qui est décrit sous la notion des caractéristiques clés et en pratique, cette structure peut être facilement obtenue à partir d'une représentation en fils de fer et les objets obtenus sont alors considérés comme un ensemble de petites faces et les caractéristiques clés en sont des sous-ensembles. Cette approche a de nombreux avantages : la taille de la base de données peut être réduite de manière considérable puisque ce système permet de générer, à la demande, tous les vues possibles sous un grand nombre de formes (contours, région, squelette...) ce qui permettent d'effectuer la discrimination et la reconnaissance d'objets.

### 2.2.1.3 Reconnaissance active d'objets 3D de forme libre par Vision en Robotique

D'une manière générale, pour reconnaître un objet, les auteurs utilisent une seule image, et lorsqu'ils n'utilisent que les caractéristiques 2D des objets 3D, ils perdent une dimension, qui peut générer des erreurs de reconnaissance entre deux objets dont les caractéristiques 2D sont similaires mais qui sont très différentes en 3D. C'est pourquoi Felipe de Jésus Trujillo [4] a essayé de traiter ce problème en prenant les caractéristiques géométriques et photométriques des objets afin d'obtenir un système de reconnaissance plus robuste en exploitant ces caractéristiques 2D de l'apparence des objets comme propriétés 3D des mêmes objets.

### 2.2.1.4 Reconnaissance active d'objets représentés par la Couleur

Pour faire un système de reconnaissance, on doit exploiter des traitements pour l'extraction des caractéristiques sur chaque image, et on peut considérer parmi ces caractéristiques : la couleur, la forme, la taille, la texture, etc., ces caractéristiques sont enregistrées dans une base d'apprentissage et

sont aussi comparées avec celles déjà enregistrées pendant la phase de reconnaissance, l'idée du Trujillo [4] est de se baser seulement sur la « couleur » pour caractériser les objets à reconnaître en utilisant un capteur actif comme une caméra PTZ qui permet d'extraire des informations discriminantes pour bien reconnaître l'objet.

#### **2.2.1.5 Reconnaissance et récupération d'objets par la correspondance du squelette graphique**

L'idée d'abord est de reconnaître l'objet préalablement extrait à partir d'un ensemble de modèles à l'aide d'une méthode d'appariement des graphes du squelette dont le but est de représenter un squelette extrait par la méthode de contour déformable puis de le comparer à un ensemble de modèles de squelettes d'objets pour sélectionner un modèle candidat afin d'avoir une reconnaissance précise des objets. La sortie du système est un objet reconnu et bien segmenté. [5]

### **2.2.2 Approches basées sur la géométrie**

#### **2.2.2.1 Reconnaissance d'objets 3D à partir des images 2D en utilisant le hachage géométrique**

C'est une technique générale de reconnaissance basée sur un modèle appelé le « Hachage Géométrique » [6] dont le but est l'identification d'un objet dans la scène, ainsi que sa position et son orientation. Cette technique est basée sur une étape intensive de prétraitement, réalisée hors ligne, où les caractéristiques invariantes de transformation des modèles sont indexées dans une table de hachage. Cela rend la reconnaissance effective et particulièrement trop efficace pour représenter un modèle 3D par ses projections 2D, et l'approche étudiée se distingue par son parallélisme inhérent élevé et sa capacité à traiter des scènes occultées. [6]

#### **2.2.2.2 Reconnaissance d'objets à l'ère géométrique**

Les premières tentatives de reconnaissance d'objets se sont concentrées sur l'utilisation de modèles géométriques d'objets pour tenir compte de leur variation d'apparence due au changement de point de vue et d'illumination. L'idée principale est que la description géométrique d'un objet 3D permet de prédire avec précision la forme projetée dans une image 2D sous une projection projective, facilitant ainsi le processus de reconnaissance à l'aide d'informations sur les bords ou les limites (ce qui est invariant pour certains changements d'éclairage). Aussi, une grande attention a été accordée à l'extraction des primitives géométriques (p. ex., lignes, cercles, etc.) qui sont invariants au changement de point de vue [7]. Néanmoins, il a été démontré que de telles primitives ne peuvent être fiables que si elles sont extraites dans des conditions limitées (variation contrôlée de l'éclairage et du point de vue

avec une certaine occlusion). Un excellent examen de la recherche de Mundy [8] sur la reconnaissance d'objets basée sur la géométrie peut également être trouvé dans [8].

## 2.3 Architecture générale d'un système de Reconnaissance

### 2.3.1 Introduction et Procédure

Un système de reconnaissance est décomposé en trois étapes essentielles (figure 2-2) : Prétraitement, extraction des caractéristiques et la classification.

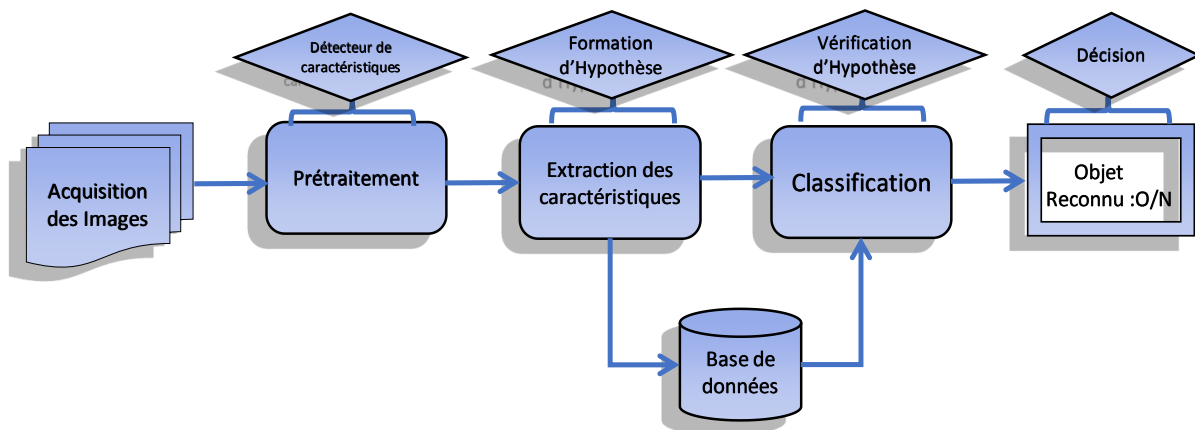


Figure 2-2: Différents composants d'un système de reconnaissance d'objets

Ce bloc-diagramme montre les interactions et le flux d'informations entre les différentes composantes du système.

La base de données des modèles contient tous les modèles connus du système, et l'information dans la base de données du modèle dépend de l'approche utilisée pour la reconnaissance. Elle peut varier d'une description qualitative ou fonctionnelle à une information géométrique de surface précise. Dans de nombreux cas, les modèles d'objets sont des vecteurs de caractéristiques abstraits. Une caractéristique est un attribut de l'objet qui est considéré très important pour décrire et reconnaître l'objet par rapport aux autres objets et parmi les caractéristiques les plus couramment utilisées : la taille, la couleur et la forme, la texture...etc.

Le détecteur de caractéristiques applique des opérateurs aux images et identifie les emplacements des caractéristiques qui aident à former des hypothèses sur les objets. Les caractéristiques utilisées par un système dépendent des types d'objets à reconnaître et de l'organisation de la base de données modèle. En utilisant les caractéristiques détectées dans l'image, l'hypothétique assigne des probabilités aux objets présents dans la scène. Cette étape est utilisée pour réduire l'espace

de recherche de la reconnaissance en utilisant certaines caractéristiques. La base de données de modèles est organisée à l'aide d'un certain type de schéma d'indexation pour faciliter l'élimination des objets candidats improbables d'un éventuel examen. Le vérificateur utilise ensuite des modèles d'objets pour vérifier les hypothèses et affiner la probabilité des objets. Le système sélectionne ensuite l'objet ayant la probabilité la plus élevée en fonction de tous les éléments de preuve, comme étant l'objet correct.

Tous les systèmes de reconnaissance d'objets utilisent des modèles de manière explicite ou implicite et utilisent des détecteurs de caractéristiques basés sur ces modèles d'objet. L'importance des composantes de formation et de vérification des hypothèses varie selon les approches de la reconnaissance d'objet. Certains systèmes n'utilisent que la formation d'hypothèses, puis sélectionnent l'objet avec la plus grande probabilité en tant qu'objet correct. Les approches de classification de modèle sont un bon exemple de cette approche. En revanche, de nombreux systèmes d'intelligence artificielle reposent peu sur la formation d'hypothèses et font plus de travail dans les phases de vérification. En fait, l'une des approches classiques, l'appariement des modèles, contourne entièrement l'étape de formation des hypothèses.[9]

En général, un système de reconnaissance d'objets doit choisir les outils et techniques appropriés pour les étapes décrites ci-dessus. De nombreux facteurs doivent être pris en compte dans le choix des méthodes appropriées pour une application particulière. Les questions centrales qui devraient être prises en compte lors de la conception d'un système de reconnaissance d'objets sont les suivantes :

### **2.3.1.1 Représentation d'objet ou de modèle**

Pour certains objets, des descriptions géométriques peuvent être disponibles et également efficaces, tandis que pour des autres, il peut être nécessaire de recourir à des caractéristiques génériques ou fonctionnelles. La représentation d'un objet doit capturer toutes les informations pertinentes sans aucune redondance et doit organiser ces informations sous une forme permettant un accès facile aux différents composants du système de reconnaissance d'objet. [9]

### **2.3.1.2 Extraction de caractéristiques**

La plupart des caractéristiques peuvent être calculées dans des images bidimensionnelles, mais elles sont liées aux caractéristiques tridimensionnelles des objets. En raison de la nature du processus de formation de l'image, certaines caractéristiques sont faciles à calculer de façon fiable tandis que

d'autres sont très difficiles. Les questions de détection des caractéristiques ont été abordées dans la partie de détection de caractéristiques locales et globales de cette thèse. [9]

### **2.3.1.3 Correspondance caractéristiques-modèle**

Dans la plupart des tâches de reconnaissance d'objets, il existe de nombreuses caractéristiques et de nombreux objets. Une approche d'appariement exhaustive résoudra le problème de reconnaissance, mais elle peut être trop lente à se concrétiser. Alors, l'efficacité des caractéristiques et l'efficacité d'une technique d'appariement doivent être prises en compte dans l'élaboration d'une approche d'appariement. [9]

### **2.3.1.4 Formation d'hypothèses**

L'étape de formation de l'hypothèse est essentiellement une heuristique visant à réduire la taille de l'espace de recherche. Cette étape utilise la connaissance du domaine d'application pour attribuer une sorte de mesure de probabilité ou de confiance à différents objets du domaine. Cette mesure reflète la probabilité de la présence d'objets en fonction des caractéristiques détectées. [9]

### **2.3.1.5 Vérification d'objet**

La présence de chaque objet probable peut être vérifiée à l'aide de leurs modèles. Il faut examiner chaque hypothèse plausible pour vérifier la présence de l'objet ou l'ignorer. Si les modèles sont géométriques, il est facile de vérifier avec précision les objets à l'aide de l'emplacement de la caméra et d'autres paramètres de la scène. Dans d'autres cas, il peut être impossible de vérifier une hypothèse. [9]

### **2.3.1.6 Complexité de la reconnaissance d'objet**

Les images de scènes dépendent de l'éclairage, des paramètres de la caméra et de son emplacement. Comme un objet doit être reconnu à partir d'images d'une scène contenant plusieurs entités, la complexité de la reconnaissance des objets dépend de plusieurs facteurs. Et pour tester la complexité de la tâche de la reconnaissance d'objets, on peut tenir compte aux facteurs suivants [9]:

- **Constance de la scène**
- **Espaces de modèles d'image**
- **Nombre d'objets dans la base de données des modèles**
- **Nombre d'objets dans une image et possibilité d'occlusion**



La tâche de reconnaissance d'objets est affectée par plusieurs facteurs. C'est pour ça nous allons catégoriser le problème de reconnaissance d'objets dans les classes suivantes :

- **Bidimensionnel -2D-**

Dans de nombreuses applications, les images sont acquises à une distance suffisante pour considérer la projection comme orthographique. Si les objets sont toujours dans une position stable dans la scène, ils peuvent être considérés comme bidimensionnels, ce qui permet d'utiliser une base de modèle bidimensionnelle. Il y a deux cas possibles :

- Les objets ne seront pas occlus, comme dans la télédétection et de nombreuses applications industrielles.
- Les objets peuvent être obstrués par d'autres objets d'intérêt ou être partiellement visibles, comme dans le cas d'un problème de pièces.

Dans certains cas, bien que les objets puissent être éloignés, ils peuvent apparaître dans des positions différentes, ce qui donne des vues stables multiples.

Dans de tels cas également, le problème peut être considéré comme une reconnaissance bidimensionnelle des objets.

- **Tridimensionnel -3D-**

Si les images d'objets peuvent être obtenues à partir de points de vue arbitraires, un objet peut apparaître très différent dans ses deux vues. Pour la reconnaissance d'objet à l'aide de modèles tridimensionnels, il convient de prendre en compte l'effet de perspective et le point de vue de l'image. Le fait que les modèles soient tridimensionnels et que les images ne contiennent que des informations bidimensionnelles affecte les approches de reconnaissance d'objet. De plus, le rôle de ces deux facteurs c'est de savoir si les objets sont séparés des autres objets ou non. Donc, pour ça nous allons profiter de travailler dans les chapitres suivantes dont on va représenter nos approches par les images de 2,5 dimension qui décrivent l'apparence visuelle des objets de différents points de vue pour donner l'impression au observateur de visualiser une image ou une scène en 3D tout en utilisant le moins possible de la réelle 3D afin de réduire les temps de calcul et accélérer les tâches de l'extraction ,la détection et la reconnaissance en bénéficiant même de leurs caractéristiques bidimensionnelles.

Pour les cas tridimensionnels, il convient de considérer les informations utilisées dans la tâche de reconnaissance d'objet et on a ici deux cas différents :

- Intensité : Il n'y a pas d'information de surface disponible explicitement dans les images d'intensité. En utilisant les valeurs d'intensité, les caractéristiques correspondant à l'intensité de la structure tridimensionnelle des objets devraient être reconnues.

- Images en 2,5 dimensions : Dans de nombreuses applications (jeux, imprimante, applications Android...), les représentations de surface avec des coordonnées centrées sur le spectateur sont disponibles ou peuvent être calculées à partir d'images. Ces informations peuvent être utilisées pour la reconnaissance d'objets et les images de distance qui sont également en 2,5 dimensions. Ces images donnent la distance aux différents points d'une image d'un point de vue particulier.

- **Segmentation**

Segmenter des images c'est de séparer les objets de l'arrière-plan, et les problèmes de reconnaissance d'objets et de segmentation sont étroitement liés dans la plupart des cas.

Dans certaines applications, il est possible de segmenter facilement un objet. Dans les cas où les objets n'ont pas été segmentés, le problème de reconnaissance est étroitement lié au problème de segmentation. [9]

## **2.3.2 Aperçu général sur les phases d'un système de reconnaissance**

### **2.3.2.1 Préparation des données**

#### **2.3.2.1.1 Numérisation**

Nous pouvons construire une représentation des données qui peuvent être directement manipulées par la machine à partir des informations du monde physique. Des capteurs (microphone, caméra, instruments de mesure) transforment les données reçues du monde réel en une représentation numérique discrète. L'espace qui en résulte, appelé espace de représentation qui a une très grande dimension  $x$  qui lui permet d'avoir autant d'informations que possible sur les images, les formes, les objets et les données numérisées en général.[10]

#### **2.3.2.1.2 Prétraitement**

L'importance de l'étape de prétraitement d'un système de reconnaissance réside dans sa capacité à remédier à certains problèmes qui peuvent se produire lors de l'étape d'acquisition. Ainsi, l'utilisation de ces techniques peut améliorer la qualité de l'image en la préparant pour l'étape suivante. Ce prétraitement consiste à sélectionner dans l'espace de représentation les informations nécessaires

pour le domaine d'application. Cette sélection implique souvent l'élimination du bruit, la normalisation des données, la segmentation, les opérations morphologiques, ainsi que l'élimination de la redondance et il existe un certain nombre de méthodes pour cette phase.

Ainsi, le nouvel espace de représentation a une dimension  $x'$  très inférieure à  $x$  mais il reste un espace de grande dimension et contient des informations encore assez primitives. [10]

### **2.3.2.1.3 Calcul des représentations ou extraction des caractéristiques**

Il s'agit de la phase finale de la préparation des données. Elle fournit un certain nombre de caractéristiques ou paramètres plus représentatives (les fameux attributs) en utilisant des algorithmes de sélection et/ou d'extraction d'attributs. Les attributs étant limités en nombre, l'espace des paramètres ainsi obtenu est de dimension  $x''$  très petite par rapport à  $x'$ . [10]

### **2.3.2.2 Apprentissage**

Les systèmes actuels de reconnaissance d'objets sont généralement limités à la reconnaissance d'objets présents dans leur base de données, et toute déviation par rapport à ces objets rend l'objet de la scène non reconnaissable. Les systèmes de reconnaissance doivent avoir la possibilité de connaître les nouveaux objets rencontrés dans des environnements non structurés et de les ajouter à la base de données. C'est pour ça l'apprentissage ou entraînement est devenu une partie importante du système de reconnaissance. Le classificateur étant généralement une fonction paramétrique, et l'apprentissage va permettre d'optimiser les paramètres du classificateur pour le problème à résoudre, en utilisant des données d'entraînement. Et lorsque les données d'entraînement sont préalablement classées, l'apprentissage est dit supervisé, sinon il est non supervisé. [10] [11]

### **2.3.2.3 Classification**

L'idée de base de la classification est de reconnaître les objets en fonction de leurs caractéristiques. Cette phase est le noyau des systèmes de reconnaissances. En utilisant les modèles (paramètres) obtenus lors de l'apprentissage, le classificateur assigne à chaque objet inconnu son ou ses objets les plus probables soit en travaillant en 2D ou 3D. [10] [11]

### **2.3.2.4 Post-traitement**

Le but de cette phase c'est de corriger les résultats de la classification en utilisant des outils spécifiques au domaine d'application. Par exemple pour un système de reconnaissance de textes manuscrits, le classificateur se charge de classer chaque caractère séparément, alors que le post

traitement applique un correcteur orthographique sur tout le texte pour le valider et éventuellement corriger le résultat de la classification. Cette phase aide à améliorer considérablement la qualité de la reconnaissance.[10]

### **2.3.3 Conclusion**

Comment on a vu dans ces deux sections, nous avons donné brièvement un aperçu sur l'état de l'art et la démarche d'un système de reconnaissance d'objet pour avoir une vue générale sur la procédure des SRO, puis, nous avons cité toutes les phases d'un SRO et les expliquer un peu pour bien éclaircir chaque tâche et comment marche-t-elle. Ainsi, et d'après ce qu'on a vu, on ne peut pas faire reconnaître un objet ou une forme, etc...sans passer de l'essentiel tâche du prétraitement. Par suite, on va se concentrer dans cette thèse sur cette tâche basique en présentant nos travaux qui s'intègrent dans le même contexte.

## **2.4 Prétraitement et Représentation des caractéristiques**

---

### **2.4.1 A propos du Prétraitement**

L'importance de l'étape de prétraitement d'un système de reconnaissance d'objet, réside dans sa capacité à remédier à certains problèmes qui peuvent se produire lors de l'étape d'acquisition. Ainsi, l'utilisation de ces techniques peut améliorer la qualité de l'image en la préparant pour l'étape suivante. D'une manière générale, les méthodes de prétraitement [12] utilisent des images soient en deux dimensions ou en trois dimensions, soient en couleur, ou au niveau de gris ou noir et blanc. De plus, de telles images peuvent également contenir un arrière-plan rendant difficile l'extraction des objets et de leurs caractéristiques quelle que soit la dimension de l'image à traiter.

Dans certaines applications, des images peuvent être obtenues de telle sorte que les objets en trois dimensions peuvent être considérés en deux dimensions. Les modèles dans de tels cas peuvent être représentés à l'aide de modèles bidimensionnels, et si les modèles sont tridimensionnels et que les effets de perspective ne peuvent être ignorés, la situation devient alors plus complexe. Donc, dans ce cas, les caractéristiques sont détectées dans un espace d'image bidimensionnel, alors que les modèles d'objets peuvent être dans un espace tridimensionnel. Ainsi, la même caractéristique tridimensionnelle peut apparaître comme une caractéristique différente dans une image.

Pour cela nous allons proposer dans cette thèse des images de dimension 2.5 c'est-à-dire que nous allons prendre des images bidimensionnelles et qui contiennent des objets tridimensionnels afin de voir des objets 3D en vue 2D, ce qui donne l'impression à l'observateur que nous avons travaillé dans un espace tridimensionnel et réel afin d'alléger le temps de calcul pendant le traitement, et nous verrons tout ça dans les chapitres suivantes 3 et 4.

Le résultat souhaité du prétraitement est une image contenant uniquement l'objet à détecter. Pour y parvenir, plusieurs techniques sont mises à notre disposition. Certaines techniques servent à améliorer l'image en supprimant le bruit et / ou corriger le contraste. D'autres, utilisent le seuillage pour éliminer le bruit de l'arrière-plan. Ensuite, il y a des techniques de segmentation qui permettent de séparer les objets les uns des autres, plus quelques opérations morphologiques comme l'amincissement et la squelettisation pour améliorer l'extraction d'une ou des caractéristiques de l'image obtenue sans perdre de ses propriétés topologiques.

Dans la suite, nous présentons quelques techniques utilisées au cours de l'étape du prétraitement en incluant un état de l'art des méthodes utilisées dans ce papier pour prendre une idée générale à propos des chapitres suivants.

## **2.4.2 Prétraitement, Description et Extraction des primitives visuelles**

### **2.4.2.1 Description des images**

#### **2.4.2.1.1 Image 2D**

Une image 2D est un tableau rectangulaire de ce que l'on appelle les pixels et les valeurs de pixel représentent l'information d'intensité ou de couleur de l'image numérique correspondante à divers endroits, indiquées par des numéros de rangées et de colonnes. Une image 2D peut également être appelée de manière informelle en tant que tranche d'image ou matrice (figure 2-3) de valeurs de pixels.

Les images 2D sont représentées en hauteur et en largeur et il n'y a aucune caractéristique de profondeur car elles sont dessinées sur des espaces bidimensionnels et selon deux axes x et y.

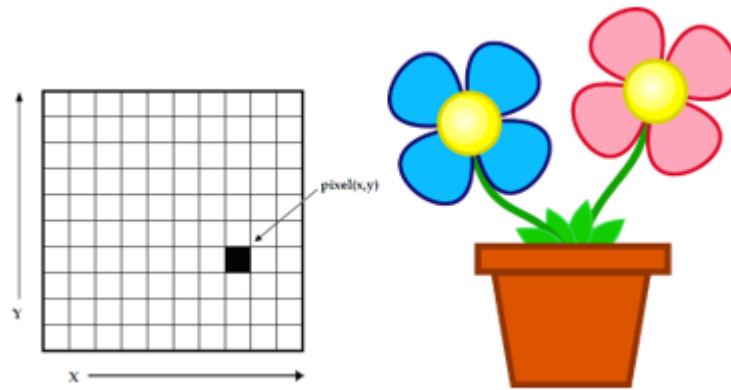


Figure 2-3: Exemple d'une image 2D

### 2.4.2.1.2 Image 2.5D

Une autre façon de déterminer la pose 3D d'un objet consiste à estimer la projection de la position de l'objet dans un espace 3D sur une image de caméra 2D. De nombreux procédés utilisent des images dites de distance ou des cartes de profondeur, où des informations sur la direction  $z$  (par exemple, la distance  $z$  par rapport au capteur) sont stockées en fonction de la position  $[x, y]$  dans le plan image (figure 2-4). Une telle représentation de données n'est pas encore complète en 3D et est donc souvent appelée 2.5D. [1]

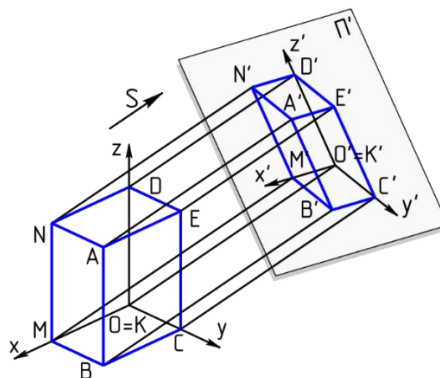


Figure 2-4: Exemple d'une projection 3D à un plan 2D pour dessiner une image de 2.5D

L'effet 2.5D peut être rendu en utilisant des ombres appliquées à un objet 2D. Ceci peut être réalisé en dessinant l'objet projetant une ombre derrière lui sur le fond. Un exemple simple serait de dessiner n'importe quel objet sur une feuille de papier blanc c'est-à-dire une image 2D, puis d'ajouter des ombres qui s'étendent derrière lui dans la distance, une dimension qui n'existe pas réellement pour cet objet, contrairement à la dimension 3D, mais nous pouvons impliquer et remarquer l'effet des ombres en étirant l'image à courte distance, et la même image à distance apparaîtra améliorée.

Notre perspective sur l'objet à partir de la 2D ressemble maintenant à la 3D. De plus, il existe des applications et des programmes qui permettent d'utiliser des outils pour générer des effets 2.5D dans des animations 2D et 3D en créant la fausse perspective, puis d'animer cette perspective en changeant pour un effet 2.5D pour donner l'impression 3D en utilisant des ombres derrière les images 2D. (figure 2-5)

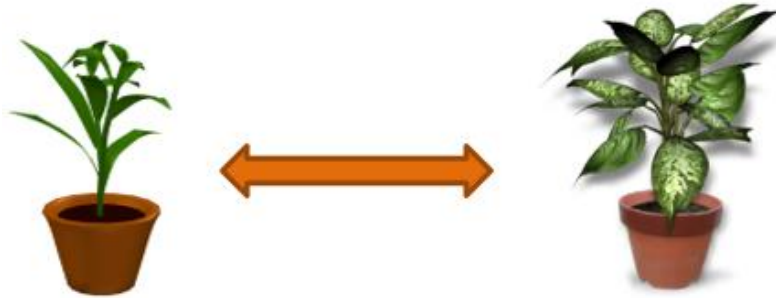


Figure 2-5: 2D vers 2.5D

Ce type d'image sera notre base de travail dans cette thèse pour qu'on puisse les traiter et savoir la différence entre les images 2D et 2.5D au niveau d'implémentation tout en utilisant des prétraitements bidimensionnels.

#### 2.4.2.1.3 Image 3D

Une image 3D est généralement vue comme une séquence ordonnée de tranches d'image 2D disposées dans la direction z et l'arrangement 3D de valeurs de voxels est appelé un matroïde de valeurs de voxel. La plupart des opérations de traitement des images 3D sont similaires à celles du traitement des images 2D. Les images 3D sont traitées à l'aide de fenêtres de numérisation 3D, tandis que les images 2D sont traitées à l'aide de fenêtres de numérisation 2D.

La principale différence entre la 2D et la 3D est le nombre d'axes. Alors que la 2D est dessinée sur deux axes, la 3D est dessinée sur un papier différent avec trois axes c'est-à-dire dans un espace tridimensionnel.

Le succès dans le domaine de l'animation se définit par la création des images 3D distinctes et complètes. Cette technique est utilisée dans l'industrie cinématographique pour actualiser des scènes. Les images améliorées combinées à un éclairage et une texture spéciale qui sont appliquées aux objets créés qui semblent solides et qui nous donnent une impression réelle.

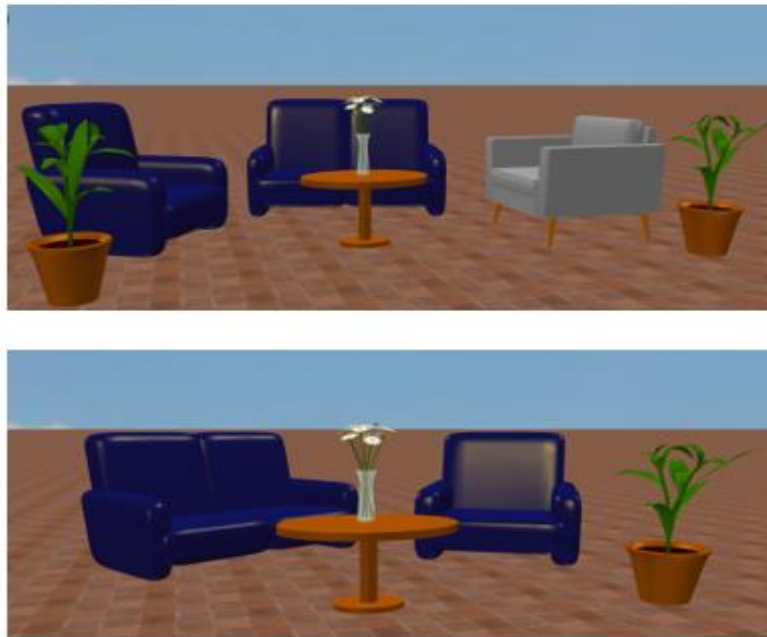


Figure 2-6: Exemple d'une image 3D sous l'effet de l'éclairage et de la texture dans un espace 3D (Logiciel « Sweet Home 3D »)

#### 2.4.2.2 Représentation des objets

Les images représentent une scène du point de vue d'une caméra. Il semble naturel pour représenter des objets dans un système de coordonnées centré sur la caméra ou sur le spectateur. Mais il y a une autre possibilité est de représenter des objets dans un système de coordonnées centré sur l'objet, et puisqu'il est facile de passer d'un système de coordonnées à un autre en utilisant leurs positions relatives, la question centrale dans le choix du système de coordonnées approprié pour représenter les objets est la facilité de représentation pour permettre la représentation la plus efficace pour la détection des caractéristiques et les processus successifs.

Les représentations pour la reconnaissance d'objets ne font pas exception car les concepteurs doivent tenir compte des paramètres dans leurs problèmes de conception afin de choisir la meilleure représentation pour la tâche.

Parmi les représentations les plus couramment utilisées dans la reconnaissance d'objets, on a deux types :

- Les représentations centrées sur l'observateur
- Les représentations centrées sur les objets



#### 2.4.2.2.1 Représentations centrées sur l'observateur

Si les objets apparaissent généralement dans un nombre relativement restreint de positions stables par rapport à la caméra, ils peuvent être représentés efficacement dans un système de coordonnées centré sur l'observateur.

Si une caméra est située à une position fixe et que les objets se déplacent de telle sorte qu'ils ne présentent que certains aspects à la caméra, alors on peut représenter des objets basés uniquement sur ces vues. Si la caméra est éloignée des objets, la tridimensionnalité des objets peut être ignorée. Dans de tels cas, les objets ne peuvent être représentés que par un ensemble limité de vues, en fait, une seule vue dans la plupart des cas. Enfin, si les objets dans un domaine d'applications sont significativement différents les uns des autres, alors des représentations centrées sur l'observateur peuvent être suffisantes. [13]

#### 2.4.2.2.2 Représentations centrées sur les objets

Une représentation centrée sur l'objet utilise la description des objets dans un système de coordonnées attaché aux objets. Cette description est généralement basée sur des caractéristiques tridimensionnelles ou sur la description d'objets.

Les représentations centrées sur l'objet sont indépendantes des paramètres et de l'emplacement de la caméra. Ainsi, pour les rendre utiles à la reconnaissance d'objets, la représentation doit avoir suffisamment d'informations pour produire des images d'objets ou des caractéristiques d'objets dans des images pour une caméra et un point de vue connus afin de produire des image 2.5 D et qui contiennent des objets 3D pour faciliter leur reconnaissance.[13]

#### 2.4.2.3 Traitement ponctuel

Le traitement ponctuel modifie les valeurs des pixels de l'image originale pour créer les valeurs des pixels correspondants dans l'image améliorée, ce qui est exprimé dans l'équation (2.1).

$$f(x, y) = T[I(x, y)] \quad (2-1)$$

Tel que,  $I(x, y)$  est l'image d'origine (d'entrée),  $f(x, y)$  est l'image améliorée et T décrit la transformation entre les deux images. Parmi les techniques du traitement ponctuel les plus utilisés : les techniques de seuillage.

### 2.4.2.3.1 Techniques de seuillage

Le seuillage est la méthode de segmentation la plus simple dont les pixels sont partitionnés en fonction de leur valeur d'intensité c'est-à-dire c'est un processus consistant à séparer les objets (informations) d'une image de son arrière-plan. Il est généralement appliqué aux images numérisées au niveau de gris ou en couleur. Le seuillage peut être classé en deux catégories principales : global et local. Les méthodes de seuillage global choisissent une valeur de seuil pour l'image entière du document, qui est souvent basée sur l'estimation du niveau de fond à partir de l'histogramme d'intensité de l'image. D'autre part, le seuillage local utilise des valeurs différentes pour chaque pixel en fonction de l'information de la zone locale.

#### 2.4.2.3.1.1 Seuillage global

Dans la littérature, de nombreuses méthodes de seuillage ont été publiées, par exemple Saho et al. [14] ont comparé les performances de plus de 20 algorithmes de seuillage global en utilisant des mesures uniformes. La comparaison a montré que la méthode de séparabilité par classe Otsu [15] [16] donnait les meilleures performances. D'autre part, dans une évaluation de la détection des changements par Rosin et Ioannidis [17], on a conclu que l'algorithme d'Otsu était très peu performant comparé à d'autres méthodes globales. Trier et Jain [18] a examiné quatre techniques globales qui montrent que la méthode d'Otsu a surpassé les autres méthodes étudiées. En outre, Fischer [19] a comparé 15 méthodes globales et a confirmé que la méthode Otsu est préférée dans le traitement d'images. La méthode Otsu est l'une des techniques largement utilisées pour convertir une image au niveau de gris en une image binaire puis on peut calculer le seuil optimal séparant ces deux classes afin que leur propagation combinée soit minimale.

Le seuillage global utilise en général un seuil approprié  $T$  :

$$f(x, y) = \begin{cases} 1, & \text{si } f(x, y) > T \\ 0, & \text{si } f(x, y) < T \end{cases} \quad (2-2)$$

Pour fixer le seuil, il faut tâtonner à partir de l'image en noir et blanc résultante censée mettre en évidence les contours. Le seuil correspond à une valeur réelle dans l'intervalle  $[0,1]$ . Au préalable, il est nécessaire de normaliser l'image du module du gradient afin que tous les pixels se trouvent également dans l'intervalle  $[0, 1]$ . Mais cette méthode est très simple et reste peu efficace. C'est pour cela la méthode d'Otsu vise à trouver la valeur optimale pour le seuillage global et effectuer

un seuillage automatique à partir de la forme de l'histogramme de l'image pour rendre la segmentation plus robuste et le seuil doit être automatiquement sélectionné par le système.

Pour choisir un seuil automatique on doit savoir : Les caractéristiques d'intensité des objets, les tailles des objets, les fractions d'une image occupée par les objets et le nombre d'objets différents apparaissant dans une image. En appliquant un seuillage global sur nos données, nous pouvons voir son effet sur l'image et l'objet et voir la différence entre l'image originale et l'image résultante.

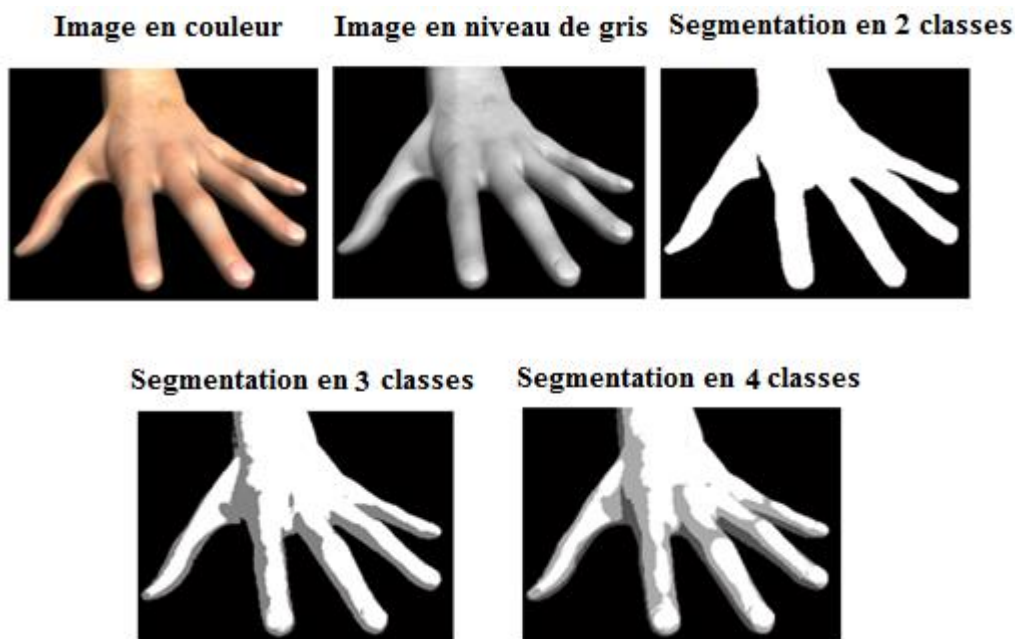


Figure 2-7: Méthode d'Otsu [15]

#### 2.4.2.3.1.2 Seuillage local

Les techniques de seuillage locaux sont utilisées avec des images ayant des arrière-plans complexes. Dans la littérature, les techniques de seuillage locales sont développées principalement pour des applications spécifiques et la plupart du temps elles ne fonctionnent pas bien dans différentes applications. Les résultats peuvent être supérieurs au seuillage ou sous seuillage en fonction du contraste et de la luminosité comme la méthode du seuillage à Hystérésis [20], qui est une méthode un peu plus perfectionnée et qui permet d'améliorer la détection des contours des zones contrastées, tout en évitant de détecter des zones du fond. Son principe est d'utiliser deux seuils : un seuil « haut T1 » et un seuil « bas T2 ». Le seuil haut va servir à sélectionner les contours les plus significatifs dans l'image. Ces contours sont contenus dans l'image résultante en noir et blanc. Le seuil bas permet de mettre en évidence des contours moins forts de l'image. Ces contours sont conservés dans l'image résultante

seulement s'ils sont situés dans le voisinage des contours les plus significatifs mis en évidence par le seuillage. En résumé, les étapes de l'algorithme sont les suivantes :

- on effectue un calcul de gradient sur l'image à niveau de gris
- tous les pixels en dessous de T1 sont mis à zéro
- tous les pixels au-dessus de T2 sont conservés
- on ne garde les pixels dont le niveau est entre T1 et T2 que s'ils sont connexes à au moins un pixel de niveau supérieur à T2.

Ainsi le niveau bas se débarrasse du bruit indésirable. Le seuillage haut ne garde que les composantes importantes mais incomplètes. La troisième étape permet de compléter ces contours.[21]



Figure 2-8: Méthode de seuillage par Hystérésis[20]

#### 2.4.2.3.2 Traitement d'histogramme

Le traitement d'histogramme est utilisé dans l'amélioration, la compression et la segmentation de l'image. Un histogramme trace simplement la fréquence à laquelle chaque niveau de gris se produit de 0 (noir) à 255 (blanc). Les images numérisées ou capturées peuvent avoir une gamme limitée de couleurs ou ne pas avoir de contraste (détails). L'amélioration de l'image par traitement d'histogramme peut permettre d'améliorer les détails, mais peut également aider d'autres opérations à vision par ordinateur, telles que la segmentation. C'est une étape initiale du prétraitement. L'égalisation d'histogramme (amélioration du contraste) [22] et la spécification d'histogramme[23] (correspondance) sont deux méthodes largement utilisées pour modifier l'histogramme d'une image pour produire une image bien meilleure.

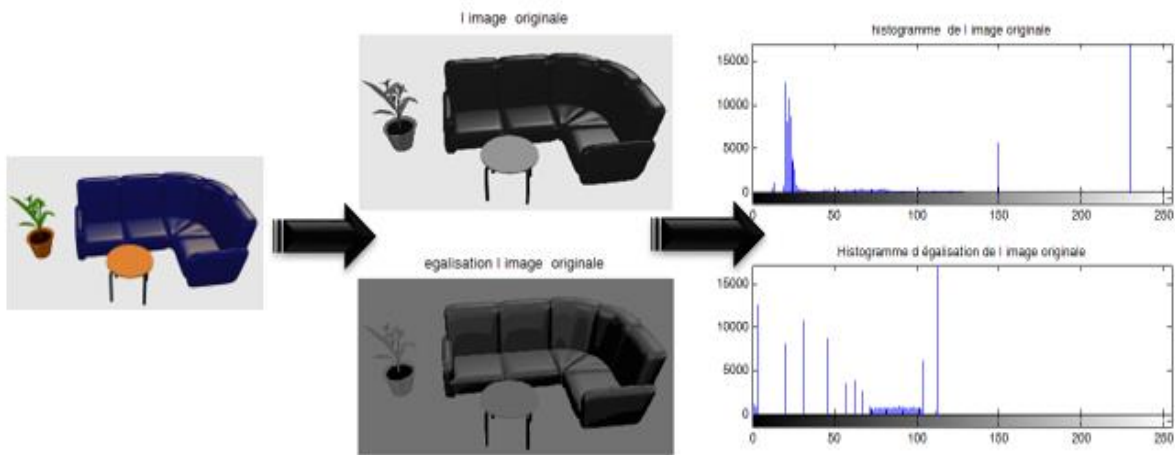


Figure 2-9: Exemple de traitement histogramme (égalisation d'histogramme)

#### 2.4.2.4 Réduction du bruit

Malgré le développement technologique des outils d'acquisition d'images, il existe encore des sources de bruit qui ne peuvent être éliminées. Il s'agit des variations stochastiques par opposition aux distorsions déterministes, telles que l'ombrage ou le manque de focalisation. La plupart du bruit peut être éliminé par les capteurs. Les systèmes de reconnaissance des objets bénéficient de la réduction du bruit en phase du prétraitement ce qui peut apporter à une amélioration substantielle de la fiabilité et de la robustesse des étapes d'extraction des caractéristiques et de classification du système SRO. Une manifestation commune du bruit dans les images binaires prend la forme de pixels isolés et le traitement de l'élimination de ce type de bruit est appelé remplissage [24]. Dans les images en gris, les filtres médians et les filtres passe-bas (Comme : Gaussien, Moyenneur) ont permis d'éliminer le bruit de pixel isolé. Le flou gaussien et les filtres moyens sont un meilleur choix pour donner une texture lisse à l'image. En général, Le filtrage est une des méthodes les plus courantes de réduction de bruit dans une image dont son but est de diminuer l'amplitude des perturbations liées au bruit, en préservant les zones de transitions. [25]

Et nous pouvons voir comme exemple l'effet du filtrage sur l'image bruité par le filtre gaussien et le filtre moyenneur :

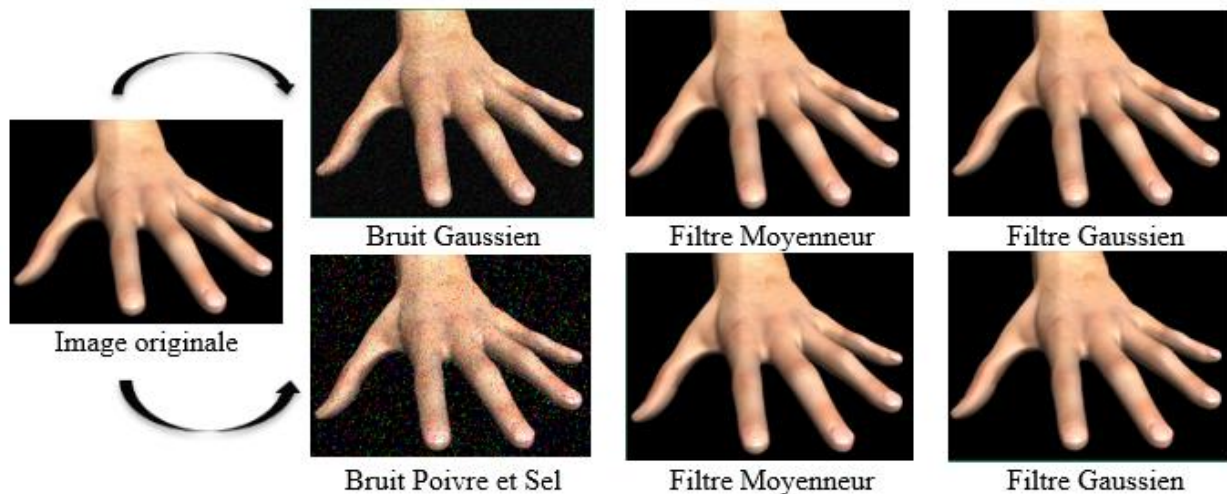


Figure 2-10: Réduction du bruit sous les filtres Moyenneur & Gaussien dans un espace obscuré

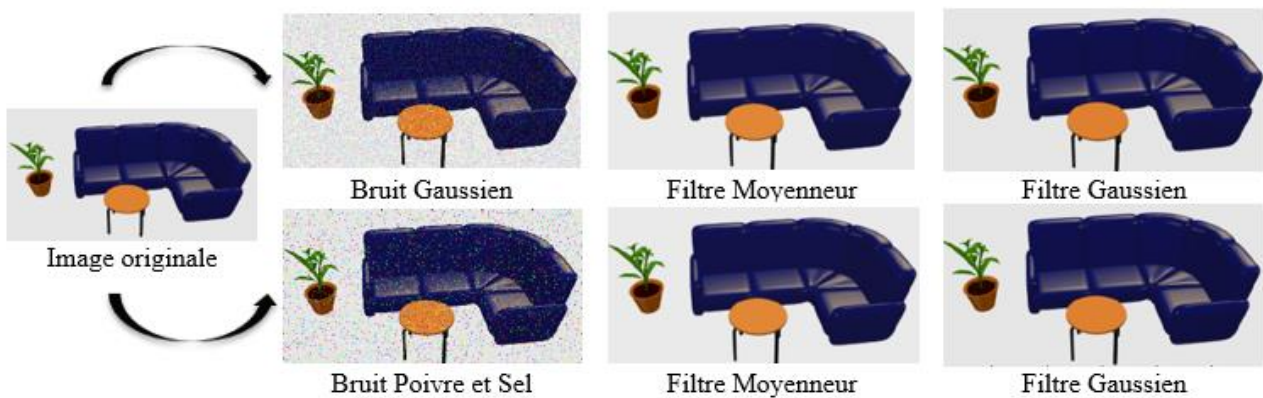


Figure 2-11: Réduction du bruit sous les filtres Moyenneur & Gaussien dans un espace lumineux

### 2.4.2.5 Segmentation

La segmentation est un processus très abordé par les chercheurs dans le domaine de la vision par ordinateur et elle reste une étape essentielle en traitement d'images qui vient après les tâches du prétraitement. En général, c'est une opération qui consiste à subdiviser une image en régions significatives comme les primitives de surfaces et des contours et qui sert à extraire les caractéristiques de l'objet et l'isoler de son arrière-plan, ce qui permet une distinction entre l'objet et le fond. En particulier, c'est la première procédure à réaliser pour effectuer une reconnaissance de formes ou d'objets 2D ou 3D car elle aide à localiser et à délimiter les entités présentes dans l'image par le découpage de cette dernière en un nombre fini de régions (ou segments) bien définies qui correspondent à des objets, des parties d'objets ou des groupes d'objets qui apparaissent dans une image.

Une erreur dans la segmentation de l'objet à reconnaître augmente forcément le risque d'une mauvaise reconnaissance. En restant sur le même thème, on va découvrir quelques classes de segmentation utilisées dans la phase du prétraitement comme :

- La segmentation fondée sur les régions
- La segmentation fondée sur les contours
- La segmentation fondée sur les points d'intérêt

#### 2.4.2.5.1 Segmentation par régions

Le but principal de cette segmentation est de diviser une image en régions. Certaines méthodes de segmentation, comme le seuillage, permettent d'atteindre cet objectif en recherchant les limites entre les régions en fonction des discontinuités dans les propriétés des niveaux de gris ou des couleurs.

La segmentation par région est une technique permettant de déterminer directement les régions, soit elles partent d'une première partition de l'image, qui est ensuite modifiée en divisant ou regroupant des régions, et on parle alors de méthodes de décomposition/fusion ; soit elles partent de quelques régions, qui sont amenées à croître par incorporation de pixels jusqu'à ce que toute l'image soit couverte, et on parle alors de méthodes par croissance de régions [26].

Les algorithmes de décomposition/fusion exploitent les caractéristiques propres de chaque région (surface, intensité lumineuse, colorimétrie, texture, etc.). Dans l'algorithmes de fusion les régions voisines sont comparées et fusionnées si elles sont suffisamment proches dans certaines propriétés , tant que ,dans l' algorithmes de décomposition les grandes régions non uniformes sont divisées en plus petites zones qui peuvent être uniformes, et les algorithmes de croissance de région partent d'un premier ensemble de régions, qui peuvent être calculées automatiquement, ou fournies par un utilisateur de manière interactive dont les régions seront grandes par incorporation des pixels les plus similaires suivant un critère donné. [27], [28].

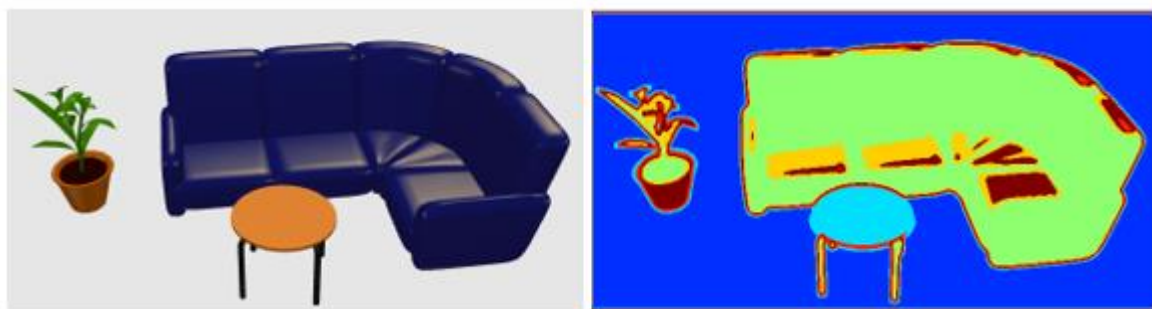


Figure 2-12: Résultat d'un algorithme de segmentation en régions



### 2.4.2.5.2 Segmentation par contour

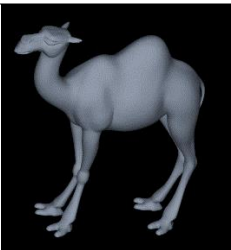
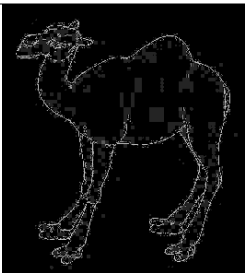
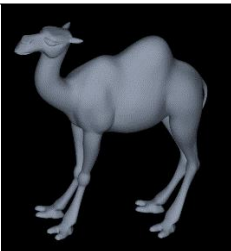
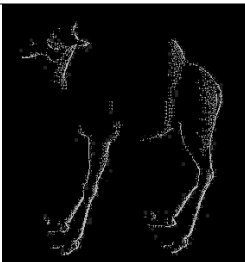
Le principe de cette approche cherche à exploiter le fait qu'il existe une transition détectable entre deux régions connexes. Plusieurs problèmes pratiques de traitement d'image nécessitent une analyse préalable où il faut découper l'image en régions homogènes séparées par des contours.

En général les régions homogènes sont des régions où l'intensité lumineuse varie lentement en fonction des coordonnées spatiales. Les contours qui séparent ces régions sont des portions filiformes de l'image c'est-à-dire ayant une faible largeur et une épaisseur d'un pixel où les variations d'intensité sont importantes.

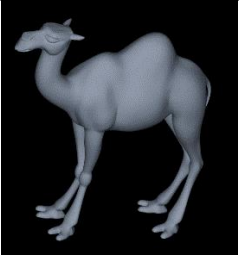
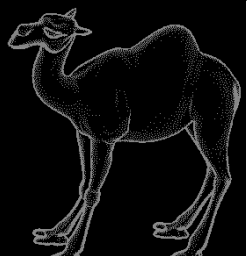
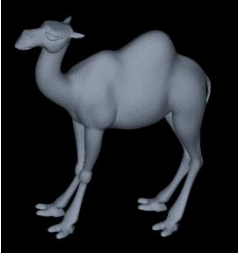
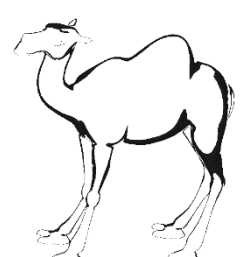
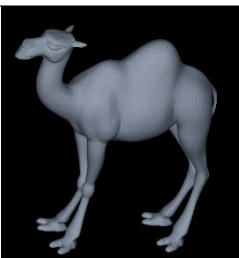

La détection des contours est une opération fondamentale en traitement d'images où les contours extraits et leurs caractéristiques peuvent servir l'entrée à un système de reconnaissance d'objets, et il existe de nombreuses méthodes pour détecter les contours parmi eux la méthode par gradient qui est basée sur un nombre d'opérateur comme Sobel, Prewitt, Robert, Kirsch...[29][25], et une autre méthode par un laplacien.

Ces méthodes ont été proposées pour détecter des transitions dans les images et leur intérêt principal c'est de minimiser le nombre d'opérations nécessaires en cas d'itération du processus sur un ensemble d'images. Et pour les découvrir un peu, on va illustrer quelques exemples de détection de contour dans le tableau suivant (tableau 2-1) :

Tableau 2-1: Quelques exemples de détection de contours et leurs propriétés

Image originale	Contour	Image résultante	Propriétés du contour
	Laplacien		<ul style="list-style-type: none"> <li>- Avoir des contours très fins</li> <li>- Amélioration et filtrage des images en accentuant l'effet de contour</li> <li>- Sensibilité au bruit accrue par rapport au gradient</li> </ul>
	Kirsch		<ul style="list-style-type: none"> <li>- Orientation du gradient de module maximum</li> </ul>



	<p><b>Robert</b></p>		<ul style="list-style-type: none"> <li>- Forte sensibilité au bruit en raison de la taille des masques, alors il paraît nécessaire d'ajouter un filtrage</li> <li>-extraction d'un contour exacte de l'objet détecté</li> </ul>
	<p><b>Prewitt</b></p>		<ul style="list-style-type: none"> <li>-Invariance par rotation</li> <li>- il combine à la fois un filtrage et une dérivée</li> <li>- sensible au bruit</li> </ul>
	<p><b>Sobel</b></p>		<ul style="list-style-type: none"> <li>- Plus développé et très populaire.</li> <li>-il combine deux opérateurs : lissage et dérivation comme Prewitt</li> <li>- c'est moins sensible au bruit</li> </ul>

En résumant tout ça, on peut dire que pour une image sans bruit avec des contours réguliers, on peut utiliser le filtre de Robert parce qu'il définit bien les contours, alors que les filtres de Prewitt et de Sobel ont tendance à les épaissir.

Pour une image un peu bruitée et contenant beaucoup de détails, le filtre de Robert conserve ses détails et les filtres de Prewitt, de Sobel et de Kirsch donnent un assez bon aperçu de tous les détails de l'image de manière assez équivalente tout en épaississant toujours les contours.

Enfin, pour une image possédant des contours francs mais bruitée, les filtres de Roberts donnent des contours assez fins mais en oublient quelques-uns, les filtres de Prewitt et de Sobel épaississent toujours les contours et débruitent légèrement l'image.

















De plus, on ne peut pas oublier de citer la segmentation par contour actif [30], [31], qui est une méthode robuste dont ce dernier est une courbe dynamique qui évolue au sein d'un domaine d'image pour atteindre une position optimale ou la forme d'un objet en minimisant une énergie fonctionnelle. Cette courbe sera déformée à partir d'une position située à proximité de l'initialisation de l'objet

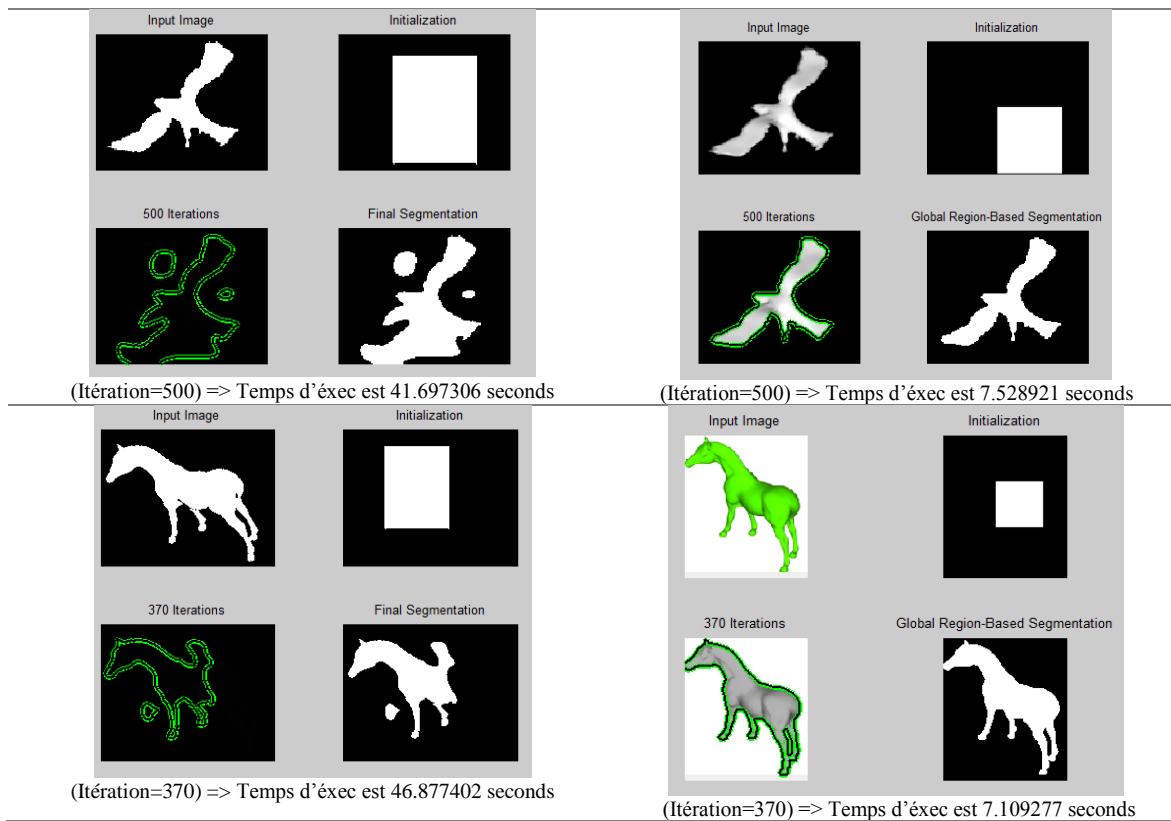
d'intérêt, puis elle sera soumise à des forces qui se déforment et se déplacent dans l'image pour bien englober l'objet à détecter.

Au début de l'algorithme, le contour est placé uniformément autour de l'objet à détecter, puis il se rétractera pour en épouser le plus possible à sa forme, puis à chaque itération, l'algorithme tentera de trouver un meilleur positionnement du contour pour minimiser les dérives par rapport aux contraintes utilisées. Ainsi, L'algorithme s'arrêtera lorsqu'il ne sera plus possible d'améliorer le positionnement ou simplement lorsque le nombre maximum d'itérations aura été atteint. Les concepts d'énergies internes et externes sont utilisés pour caractériser respectivement la forme du contour et son positionnement sur l'image en tenant compte des lignes de gradient.

Pour évaluer cette méthode, nous avons la testé sur des images 2D et des image 2.5D pour faire une telle comparaison au niveau de la rapidité de la tâche, le temps de calcul, le nombre d'itération et l'extraction complète de la forme d'objet détecté (tableau 2-2) :

Tableau 2-2: Comparaison entre les images 2D et 2.5D en appliquant la segmentation par contour actif

Application aux images 2D		Application aux images 2.5 D	
 <p>Input Image</p>	 <p>Initialization</p>	 <p>Input Image</p>	 <p>Initialization</p>
 <p>250 Iterations</p>	 <p>Final Segmentation</p>	 <p>250 Iterations</p>	 <p>Global Region-Based Segmentation</p>
(Itération=250) => Temps d'exéc est 67.785702 seconds		(Itération=250) => Temps d'exéc est 4.614399 seconds	
 <p>Input Image</p>	 <p>Initialization</p>	 <p>Input Image</p>	 <p>Initialization</p>
 <p>3580 Iterations</p>	 <p>Final Segmentation</p>	 <p>3580 Iterations</p>	 <p>Global Region-Based Segmentation</p>
(Itération=3580) => Temps d'exéc est 1026.557159 seconds		(Itération=3580) => Temps d'exéc est 164.293899 seconds	



D'après les résultats obtenus dans le tableau 2-2, on peut déduire que la segmentation des objets 3D en vue 2D par le contour actif est complète parce que le contour actif englobe toute la forme de l'objet détecté et sous le même nombre d'itération pour les deux tests, on peut dire aussi qu'elle est très rapide, par contre, la segmentation faite par les images bidimensionnelles qui contiennent des objets 2D n'est pas satisfaisante pour compléter la détection du contour malgré qu'on a respecté le même nombre d'itérations à chaque type d'objet, mais elle est retardé trop par rapport à l'autre, ce qui montre la bonne performance du contour actif aux images 2.5D relativement aux images 2D

### 2.4.2.5.3 Segmentation par les points d'intérêt (Etat de l'art)

Une segmentation en points d'intérêt est également une segmentation statique et une bonne segmentation résulte un bon système de reconnaissance d'objet. Pour cela, on doit analyser nos images et extraire d'eux des primitives et caractéristiques significatives (contours, fonctions d'intensité ou modèles morphologiques). Ces caractéristiques peuvent être une région d'intérêt qui représente une propriété intéressante d'une partie de l'image, et le point d'intérêt est un cas particulier des régions d'intérêt. De nombreuses méthodes s'appuient sur l'utilisation des points d'intérêts qui caractérisent un coin dans une image ou un objet dedans une image soit 2D ou 3D. Et afin de mettre en correspondance un ensemble de points d'une image à une autre, une description locale est utilisée et

permet d'extraire l'information du voisinage de chaque point (valeurs des pixels, des intensités lumineuses, des gradients). [32]

La détection et la description des points d'intérêts a connu une longue tradition au sein de traitement et d'analyse d'images et le problème essentiel en détection c'est de trouver des points d'intérêt d'une manière automatique et suffisamment robustes et fiable. Pour cette raison, on va mettre l'accent sur les points d'intérêt et les différents types de détecteurs des PIs soient 2D ou 3D. De même, pour les découvrir et comprendre leurs rôles, nous allons juger de doter cette partie d'un état de l'art aux différentes approches de détection de points d'intérêt dédiés aux images 2D et aux objets 3D.

#### 2.4.2.5.3.1 Différentes approches

Plusieurs méthodes ont été proposées pour détecter des points d'intérêts et elles peuvent être classées grossièrement suivant trois catégories :

- **Approches contours** : l'idée est de détecter les contours dans une image dans un premier temps. Les points d'intérêts sont ensuite extraits le long des contours en considérant les points de courbures maximales ainsi que les intersections de contours.
- **Approches à base de modèles** : les points d'intérêts sont identifiés dans l'image par mise en correspondance de la fonction d'intensité avec un modèle théorique de cette fonction des points d'intérêts considérés. Ces méthodes ne sont utilisables que pour des types précis de points d'intérêt comme les coins.
- **Approches intensité** : l'idée est cette fois-ci de regarder directement la fonction d'intensité dans les images pour en extraire directement les points de discontinuités. Ces méthodes n'utilisent ni les contours ni le modèle théorique du signal. La présence ou pas d'un point d'intérêt est indiquée par une mesure calculée directement à partir du signal. [33][34]

Les approches de la troisième catégorie sont celles utilisées généralement et plusieurs détecteurs s'affilient sous cette catégorie et on va parler brièvement sur quelques opérateurs de détection des PIs qui ont été proposés dans la littérature des plus simples aux plus complexes en citant le principe de chacun d'eux.

### 2.4.2.5.3.2 Détecteurs 2D

#### 2.4.2.5.3.2.1 Détecteur de coins

Ces détecteurs localisent les points et les régions d'intérêt qui ont une propriété importante en l'image et ne conviennent pas à la détection de régions homogènes. Ce type de détecteurs ont été utilisés dans plusieurs travaux pour obtenir une telle robustesse et stabilité pour la reconnaissance d'objets.[2]

#### I. Moravec

Les travaux les plus anciens concernant la détection iconique des points d'intérêt sont ceux de Moravec [35]. Il a défini un détecteur qui introduit pour la première fois le concept d'autocorrélation. Le principe était de calculer les variations d'intensité lumineuse dans quatre directions parallèles aux lignes et aux colonnes de l'image, puis de mesurer les différences entre une fenêtre rectangulaire autour d'un pixel et quatre fenêtres voisines. Selon Moravec, un point ayant des variations d'intensité dans toutes les directions est un point d'intérêt. [33]

#### II. Harris

- **Détecteur basé sur la matrice de Harris**

Il faut préciser qu'en 1988, Harris et Stephens [36] voulaient obtenir des bons résultats fournis par le détecteur précédant, ils ont repris les idées de Moravec et les améliorer afin de le rendre plus stable et plus performant. Comme on a dit ce détecteur est défini par Harris et Stephens [36] en se basant sur le calcul de la fonction d'autocorrélation, Harris et Stephens se ramenaient à l'étude de valeurs propres de la matrice de Harris (matrice de moments de second ordre). Sachant que les valeurs propres de cette matrice de Harris représentent les courbures principales de la fonction d'autocorrélation, nous avons alors trois cas qui se présentent (figure 2-13) :

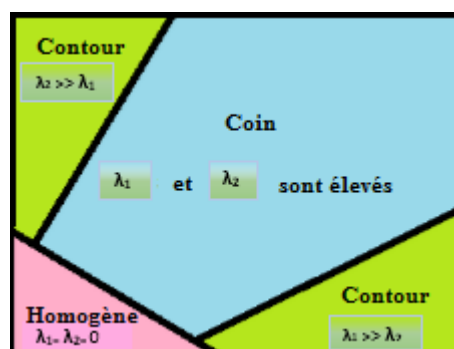


Figure 2-13: Distributions des types de régions en fonction des deux plus grandes valeurs propres de la matrice de Harris[36]

- Une région homogène, si les deux valeurs propres sont faibles.
- Un contour (transition) si l'une des valeurs propres est très grande par rapport à l'autre.
- Un coin (point d'intérêt), si les deux valeurs propres sont élevées.

Cet opérateur permet de ne garder que les structures en coins, c'est à dire de courbure suffisamment grande. Malgré leur taux élevé de répétabilité, ces détecteurs restent sensibles au bruit et au changement d'échelle. Par suite, Dufournaud et al [37] ont proposé une version multi-échelle de l'opérateur de Harris dans laquelle les points d'intérêt correspondent aux maxima locaux du détecteur de Harris appliqué à plusieurs échelles.[2]

- **Détecteur basé sur la matrice de Harris-Laplacien**

Le détecteur de Harris-Laplace [38] détecte les points d'intérêt en utilisant la fonction de Harris adaptée à l'échelle. Par la suite, les points maxima du Laplacien de gaussien sont choisis à une échelle caractéristique. Le Laplacien correspond à la trace de la matrice de Harris. Ces détecteurs sont invariants à l'échelle et aux transformations affines.

### III. SUSAN

Ce détecteur nommé SUSAN (Smallest Univalued Segment Assimilating Nucleus) est proposé par Smith et Brady [39], il se base sur une analyse locale circulaire. Cette approche n'utilise pas les dérivées des intensités de l'image mais elle mesure, dans une fenêtre, le nombre de pixels ayant une intensité voisine de celle du pixel central (figure 2-14), et une forte variation des intensités signifie l'existence d'un point d'intérêt [33]. Ce détecteur permet de détecter les contours, les coins (figure 2-15) et aussi les jonctions et on détermine les points d'intérêt de la façon suivante :

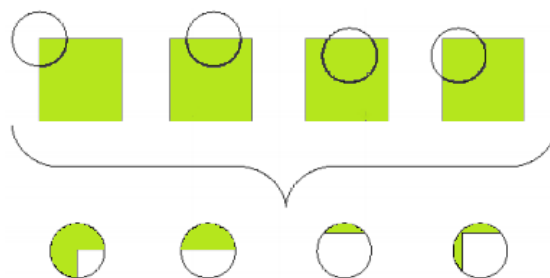


Figure 2-14: Exemple de masques d'analyse du détecteur SUSAN (haut), permettant le calcul des zones USAN (bas)

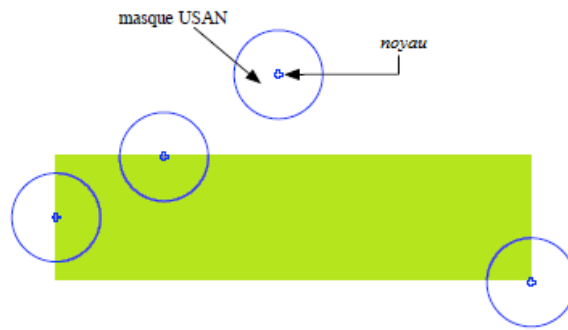


Figure 2-15: Détecteur de SUSAN - Illustration du principe du détecteur de SUSAN

- Si à l'intérieur de ce disque, tous les pixels ont la même intensité que le noyau, alors on est dans une zone homogène ;
- Si dans ce disque, la moitié des pixels ont la même intensité que le noyau alors le pixel étudié est sur un contour ;
- Si seulement peu de pixels dans ce disque ont la même intensité que le noyau, alors on est en présence d'un coin. [40]

Pour savoir de plus sur ce détecteur, on va le détailler dans le chapitre 4 de l'extraction des PIs à partir d'un nouveau détecteur de SUSAN.

#### 2.4.2.5.3.2 Détecteur de régions

Ces détecteurs s'attaquent aux défauts des détecteurs des coins dans les régions uniformes et les régions lissées. En effet, ces détecteurs extraient les régions de l'image qui sont des zones homogènes en termes d'intensité.

### I. SIFT

L'algorithme de SIFT (Scale Invariant Feature Transform) est présenté dans [41] par D.Lowe, son but était de localiser des points d'intérêt avec un vecteur descripteur afin de pouvoir caractériser un objet et être capable de le reconnaître en comparant les caractéristiques des points trouvés à une base de données, c'est-à-dire qu'il s'agit ici non seulement de détecter mais aussi de caractériser par des valeurs, pour pouvoir reconnaître par la suite ces points d'intérêts dans d'autres images de la même scène. Un autre objectif est également de résoudre le problème du changement d'échelle qui pose généralement des difficultés aux autres détecteurs.

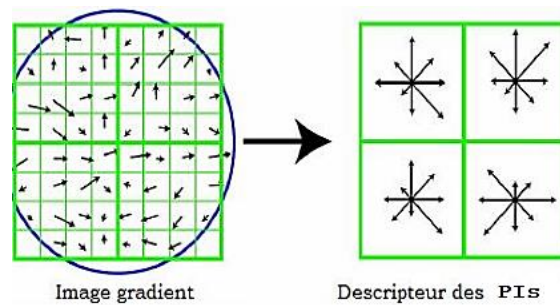


Figure 2-16: Extraction de PIs par le détecteur-descripteur SIFT

En 2005, Mikolajczyk et Schmid [42] et les auteurs Dalal et Triggs [43] ont présenté une extension à SIFT : GLOH (Gradient Location-Orientation Histogram) afin d'augmenter la performance et la robustesse du descripteur SIFT. [40]

## II. Kadir

Kadir et Brady [44] introduisent le détecteur de régions saillantes qui s'appuie sur la fonction de densité de probabilité (PDF) des valeurs d'intensité. La PDF de l'entropie est estimée par les valeurs de l'histogramme des niveaux de gris dans un patch circulaire à une échelle et une position précise. Ce détecteur est invariant au changement d'échelle et à la rotation.

Un autre travail de Kadir [45] propose une extension de cet algorithme pour assurer l'invariance affine. Il considère une recherche exhaustive sur les déformations elliptiques du patch qui reste très coûteuse en temps de calcul.[2]

## III. Beaudet

Le détecteur de Beaudet [46] est l'un des premiers détecteurs de points d'intérêt qui est basé sur la matrice Hessienne et il définit un opérateur à partir des secondes dérivées du signal pour juger si un point est un point d'intérêt ou non. Son point faible est qu'il est basé sur des dérivés secondes ce qui le rend plus sensible au bruit [33] [47].

## IV. SURF

SURF (Speeded Up Robust Features) est un descripteur robuste d'image, qui fut d'abord présenté par Herbert Bay et al. [48] en 2006, et qui peut être utilisé pour la reconnaissance d'objet ou la reconstruction 3D. Il est partiellement inspiré de SIFT. Sa version standard est bien plus rapide que SIFT et ses auteurs affirment qu'il est plus robuste aux différentes transformations d'images car il est



basé sur des ondelettes 2D appelées les ondelettes de Haar qui sont appliquées à des images intégrales ce qui permet de diminuer les temps de calculs d'une façon significative [49][50]

### 2.4.2.5.3.2.3 Autres détecteurs

Il existe d'autres détecteurs dont certains utilisent le passage par zéro du laplacien [29], d'autres retournent les points où la variance locale [51] est élevée, d'autres encore utilisent le contraste [52]. le détecteur de Paler, [53] fait la différence entre l'image d'origine et l'image filtrée pour retrouver les coins. Le détecteur de Itti et al [54] s'inspirent de la première phase pour proposer un détecteur de régions saillantes qui est basé sur des pyramides de cartes d'intensité, de couleurs et d'orientations du gradient de l'image. Les cartes sont ensuite normalisées puis combinées afin de sélectionner les points les plus saillants. Ainsi, il y en a beaucoup de détecteurs comme Kitchen-Rosenfeld, Noble, Harris électrostatique, MSER (régions) CSS, Achard, DOG, etc. [2]

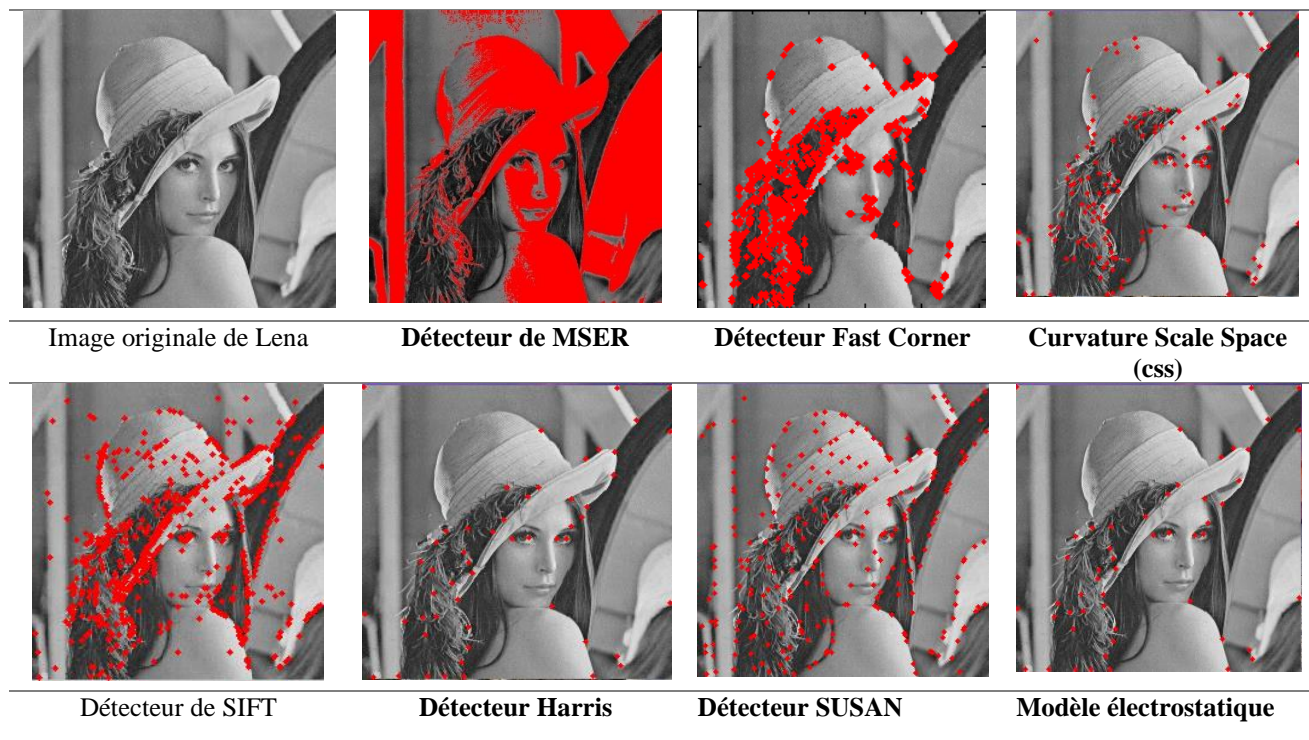


Figure 2-17: Exemple de quelques algorithmes de détection des points d'intérêt sur une image 2D

### 2.4.2.5.3.3 Détecteurs 3D

#### I. Harris 3D

Sipiran, et al.[55], ont proposé un algorithme qui est une extension de la méthode de Harris[36] de détection de coins pour des maillages 3D. pour sélectionner les PIs ,il s'est passé par deux façons: soit

en prenant un nombre fixe de sommets ayant les plus grandes réponses, soit par une approche de regroupement (clustering) pour avoir des PIs bien distribués sur toute la surface de l'objet.[2]

## II. 3D-SURF

Ce détecteur SURF 3D est introduit par Knopp et al.[56] , c'est une extension de la méthode SURF à la 3D qui définit une matrice Hessienne 3D pour des primitives spatio-temporelles. La saillance maximale est extraite sur plusieurs octaves obtenues par l'application de gaussiennes aux données voxelisées. [2]

## III. 3D-SIFT

Scovanner et al. [57] introduisent un descripteur SIFT 3D pour des vidéos avec le temps comme troisième dimension. Similairement au 2D, le gradient spatio-temporel ( $L_x, L_y, L_t$ ) est calculé sur les points 3D  $M_{3D}(x, y, t)$  qui définissent la magnitude et l'orientation du gradient.

$$M_{3D}(x, y, t) = \sqrt{L_x^2 + L_y^2 + L_t^2}; \theta(x, y, t) = \tan^{-1}\left(\frac{L_y}{L_x}\right);$$

$$\phi(x, y, t) = \tan^{-1}\left(\frac{L_t}{\sqrt{L_x^2 + L_y^2}}\right)$$
(2.3)

Chaque point a assigné un couple d'angles ( $\Theta, \phi$ ) qui renseigne sur l'orientation du gradient en 3D. Un histogramme 2D qui cumule la distribution des orientations de ces angles est fourni comme signature. Les points d'intérêt sont choisis aléatoirement pour permettre une utilisation en temps réel. [2]

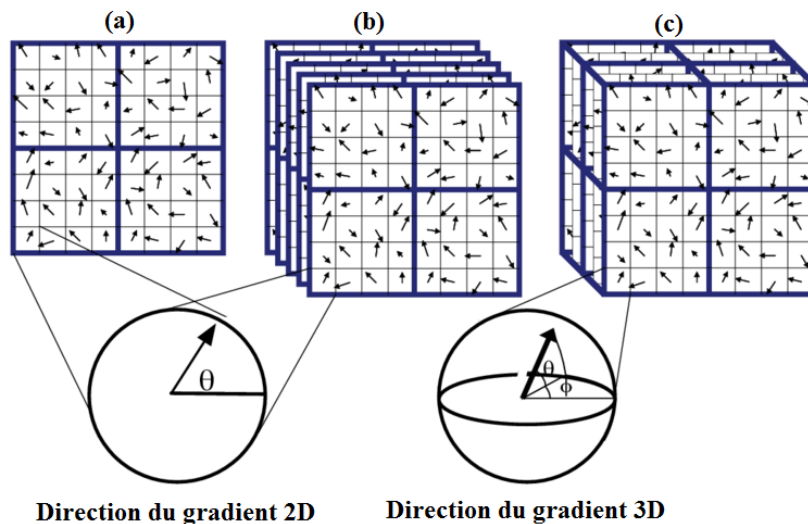


Figure 2-18: Image (a) : le descripteur SIFT 2D normal. Image (b) : le descripteur SIFT 2D. Image (c) : le descripteur SIFT 3D avec ses sous-volumes 3D. [57]

#### **IV. Heat Kernel Signature (HKS)**

Sun et al [58] utilisent comme mesure de qualité le noyau de chaleur calculé sur la maille: la résolution de l'équation de chaleur sur l'espace et le temps permet de construire un équivalent des espaces d'échelle. Les maximas du noyau sont alors choisis comme PIs. [2]

#### **V. Points saillants pondérés**

Tangelder et Veltkamp [59] proposent une approche locale fondée sur des représentations par points saillants pondérés permettant la prise en compte de la position spatiale relative de l'objet polyèdre. Ils commencent par appliquer une ACP (Analyse en composantes principales) pour normaliser la position de l'objet et le décompose ensuite en une grille 3D.

Pour l'opération de détection et de pondération des points d'intérêt, ils comparent trois méthodes qui utilisent les sommets et les facettes qui lui sont adjacentes : la Méthode de courbures gaussiennes [2], la Méthode de variation des normales [2] et la Méthode du mi-point. [2]

#### **VI. Paires de points saillants**

Li et al. [60] utilisent la détection des points saillants avec une projection linéaire sur une surface lissée et une seule version d'échelle des caractéristiques multi-échelles Li, et al. [61]. En effet, l'approche de sélection des PIs se base sur une représentation en espace d'échelle d'une surface c'est-à-dire qu'ils font une projection des points 3D et de leur normale sur une version lissée de la surface. [2]

#### **VII. Coins SD (SD-corners)**

Novatnack et Nishino [62] ont également construit une représentation espace-échelle du modèle ; et ils définissent deux types de coins géométriques : les points qui ont une forte courbure isotrope, et les points qui ont une forte courbure dans au moins deux directions. Ainsi, ils calculent le Gram matrice du premier ordre de dérivés partielles de la carte normale à chaque point. Si la valeur propre maximale de la matrice de Gram est élevée à un point alors le point est considéré comme ayant une forte réponse d'angle. Certains de ces coins candidats peuvent résider sur les bords plutôt que sur les coins, de sorte qu'ils sont éliminés en utilisant deuxième dérivées d'ordre de la carte normale.

Les coins sont détectés séparément à chaque échelle, puis des coins à différentes échelles sont rassemblés dans un seul ensemble. Enfin, ces coins détectés dans le domaine 2D et ils sont mappés en arrière de la surface du modèle 3D. [63][2]

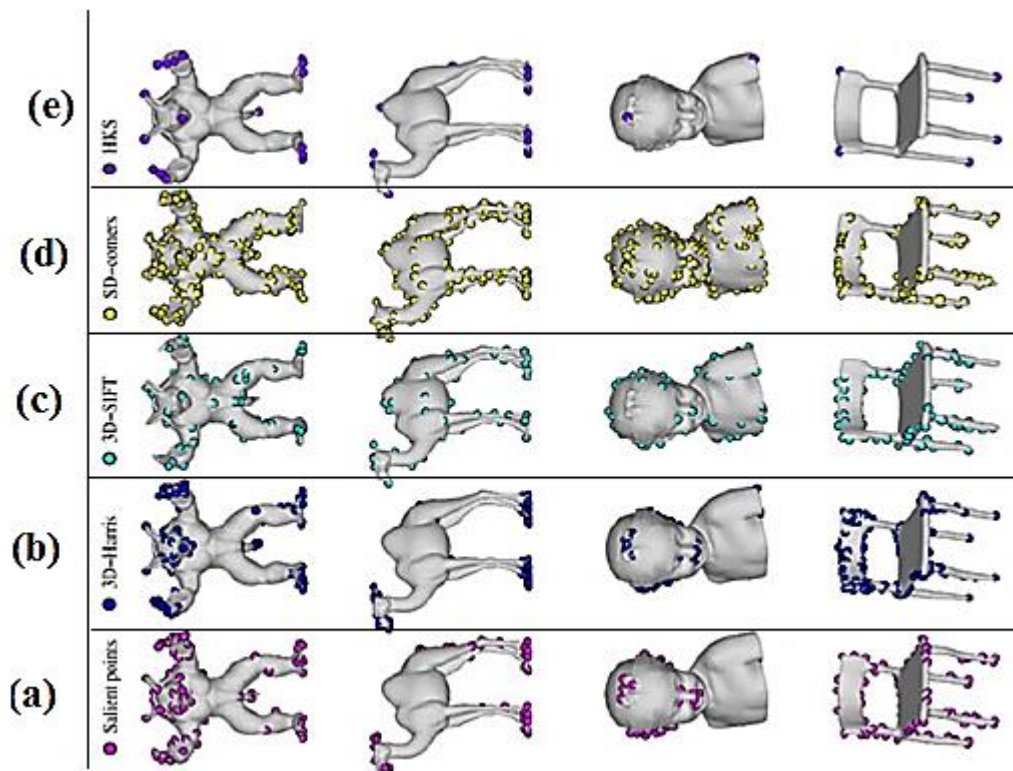


Figure 2-19: Points d'intérêt détectés par points saillants(a), 3D-Harris(b), 3D-SIFT (c), SD-corners (d), et HKS (e) [63]

#### 2.4.2.6 Opérations morphologiques

Les résultats de la segmentation peuvent entraîner la suppression de certains pixels qui produisent des trous dans certaines parties des images ; Cela pourrait être vu à partir des objets dedans des images ayant certains trous causés par la suppression de quelques pixels pendant le seuillage et après leur binarisation. Des trous plus grands peuvent provoquer des ruptures et des déformations à l'objet traité. D'autre part, la segmentation rend plus difficile la séparation des objets, et ces objets qui ressemblent à des gouttes et seront difficiles à interpréter. La solution à ces problèmes est le filtrage morphologique.

Le principe de la morphologie mathématique consiste à comparer l'image à analyser à un ensemble géométrique connu appelé élément structurant qui est déplacé de façon à ce que son origine passe par toutes les positions de l'image, pour mettre en évidence certaines caractéristiques de l'image [64]. En suivant ce principe, on peut alors se baser sur les techniques [65] suivantes : l'érosion et la dilatation, l'ouverture et la fermeture, l'amincissement et la squelettisation. Ces techniques ne fonctionnent que sur des images binaires (La morphologie mathématique ensembliste [64]) ou au niveau de gris (La morphologie mathématique fonctionnelle [64]).

### 2.4.2.6.1 Érosion et Dilatation

Les opérations de dilatation et érosion sont des opérations morphologiques qui augmentent ou diminuent les objets en taille et peuvent être très utiles pendant le prétraitement. L'érosion rend un objet plus petit en supprimant les pixels sur ses bords ; cependant, la dilatation rend un objet plus grand en ajoutant des pixels autour de ses bords. Il existe deux techniques générales d'érosion et de dilatation : le seuillage et les techniques de masquage. La technique de seuillage regarde les voisins d'un pixel et change son état si le nombre de pixels voisins différents dépasse un seuil. Fondamentalement, si le nombre des pixels zéro au voisinage d'un pixel dépasse un paramètre de seuil alors le pixel est mis à zéro. La technique de masquage utilise un tableau  $n \times n$  des uns et zéros sur une image d'entrée et érode ou dilate l'entrée, la direction de l'érosion ou de la dilatation peut être contrôlée [66].

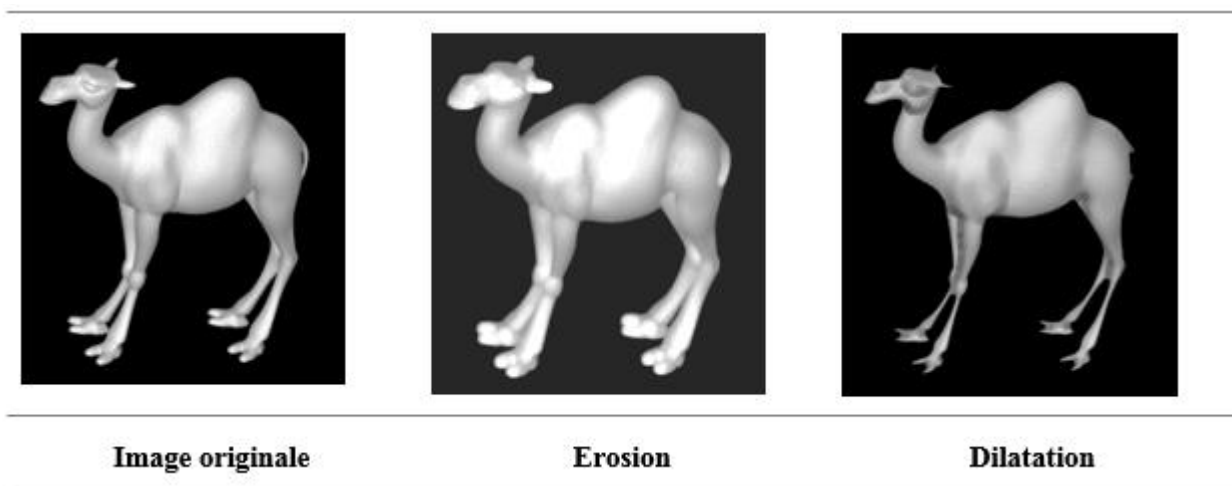


Figure 2-20: Exemple d'érosion et dilatation d'un objet 3d en vue 2d en niveau de gris

### 2.4.2.6.2 Ouverture et Fermeture

Les opérations « ouverture » et « fermeture », désignent les opérateurs morphologiques qui sont dérivés des opérations fondamentales de l'érosion et de la dilatation. Ils sont normalement appliqués aux images binaires. L'effet de base d'une ouverture/fermeture est similaire à celui de l'érosion/dilatation dans la mesure où il tend à supprimer/ajouter certains des pixels du premier plan sur les bords. Cependant, il est moins destructeur que l'érosion/dilatation en général.

Les opérateurs d'ouverture et de fermeture fonctionnent bien, mais produisent parfois des résultats indésirables où la fermeture peut fusionner des objets qui ne doivent pas être fusionnés et l'ouverture peut élargir les trous et provoquer la rupture d'un objet [65], [66].

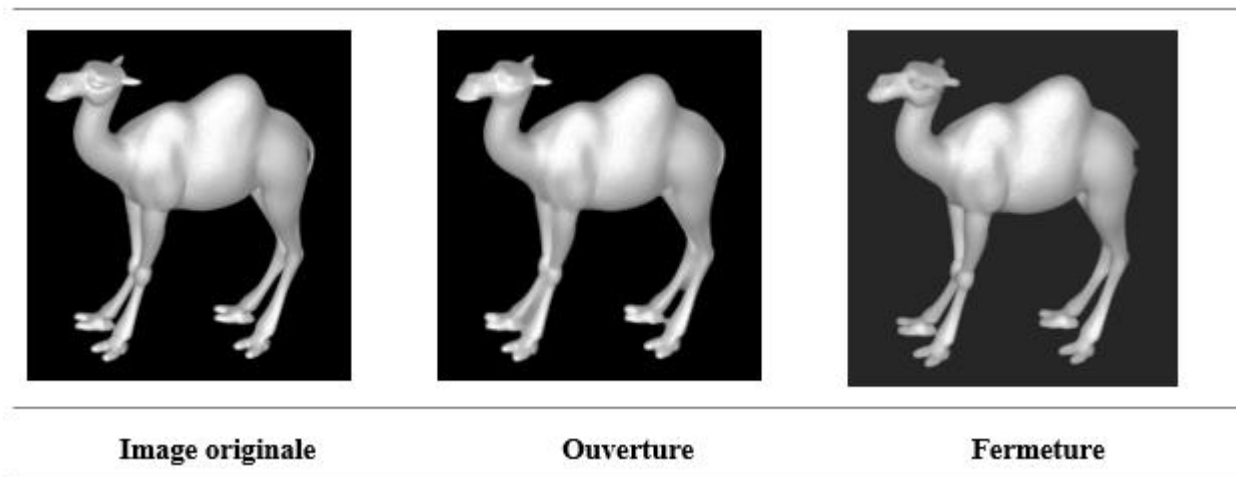


Figure 2-21: Exemple d'ouverture et de fermeture d'un objet 3d en vue 2d en niveau de gris

#### 2.4.2.6.3 Amincissement et squelettisation (Etat de l'art)

La squelettisation est un processus qui permet de produire une version réduite d'une image binaire avec une épaisseur d'un seul pixel sans perdre les propriétés topologiques, de connectivité de l'image originale. Pendant le processus, la plupart des pixels d'avant-plan de l'image originale sont éliminés. Les pixels non éliminés sont ceux qui appartiennent au squelette de l'avant-plan de l'image.

Il existe deux techniques de base pour produire le squelette d'un objet : l'amincissement de base et les transformations de l'axe médian. C'est une opération morphologique qui est utilisée pour éliminer les pixels sélectionnés du premier plan des images binaires, un peu comme l'érosion ou l'ouverture. L'amincissement est un processus de réduction de données qui érode un objet jusqu'à ce qu'il soit d'un pixel de largeur, produisant un squelette de l'objet.

Cette technique de base fonctionne bien, mais il est impossible de recréer l'objet original à partir du résultat de l'amincissement. La recréation de l'objet original nécessite la transformation de l'axe médian. Ainsi, cette dernière se compose de tous les points d'un objet qui sont à une distance minimale de plus d'un bord de l'objet. Chaque pixel au bas était la distance la plus courte aux deux bords de l'objet, ce qui forment des lignes vers le centre [15]. Elle est semblable à la mesure de la distance euclidienne de n'importe quel pixel au bord de l'objet [65], et le pixel qui minimise cette distance, appartient à l'axe médian de l'objet.

Étant donné une forme, un squelette est une structure mince centrée qui décrit conjointement la topologie et la géométrie de la forme, c'est pour cela plusieurs types de squelettes peuvent être définis, chacun ayant son propre ses propres propriétés, avantages et inconvénients. De même, il existe un

grand nombre de méthodes pour calculer un type de squelette donné ayant ses propres exigences, avantages et limites. Alors, nous présenterons ici des algorithmes de processus de squelettisation dans l'espace 2D avant de donner leurs extensions existantes dans l'espace 3D.

#### **2.4.2.6.3.1 Squelettisation 2D**

##### **2.4.2.6.3.1.1 Algorithme d'amincissement « Thinning 2D »**

L'amincissement consiste à éroder peu à peu le contour des objets jusqu'à obtenir une figure mince et centrée. A un instant donné, un point du contour est supprimé s'il vérifie les deux conditions suivantes d'une part, sa suppression ne modifie pas l'homotopie (il ne doit pas apparaître de trous dans l'objet, et l'objet ne peut pas être coupé). D'autre part, ce point n'est pas une extrémité de l'ensemble courant (il faut préserver les branches qui apparaissent). Finalement, les points que l'on peut supprimer sont appelés des points simples non terminaux.

Les méthodes d'amincissement effectuent plusieurs passages sur l'image. Durant un passage, tous les points de l'image sont examinés séquentiellement ou parallèlement. Les points simples non terminaux sont alors supprimés. Pour déterminer si un point peut être supprimé ou non, il suffit d'examiner son voisinage immédiat. C'est un traitement purement local.

Il existe plusieurs références sur le sujet qu'il est impossible de récapituler ici. Dans [67] plus d'une centaine de références sont passées en revue.

Ces méthodes s'étendent facilement l'espace 3D [68]–[71]. Le principe reste le même. Il s'agit de détecter les points simples et de les supprimer. Ce faisant, il faut veiller à préserver le nombre de composantes connexes, de cavités et de trous.

##### **2.4.2.6.3.1.2 Simulation de la propagation d'une onde « feu de prairie »**

Les méthodes présentées dans cette partie s'inspirent directement d'une propriété des squelettes continus. Cette propriété est que le squelette peut s'interpréter comme le lieu d'extinction d'un front d'onde parti du contour et se propageant dans la forme à vitesse constante et de façon isotrope. A l'endroit où plusieurs fronts d'onde se rencontrent, leurs propagations s'arrêtent et donnent la position du squelette. [72] Pour calculer le squelette dans une image, il suffit donc de simuler la propagation d'une onde. Généralement, l'onde est modélisée par un contour discret qui se déplace selon des techniques de suivi de contour ou de lacets [73]. Dans [74], la propagation utilise le principe de Huygens. A chaque étape, un nouveau front d'onde est généré à partir du front d'onde courant. Leymarie

propose une méthode originale[75]. Le squelette est le résultat de la stabilisation d'un contour actif (et non plus discret) dans une fonction potentielle.

Cette fonction potentielle est l'inverse de la carte de distance. Le fait de simuler un contour discret rend très difficile toute extension 3D. Un algorithme de squelettisation par suivi de surfaces a été proposé par Marion-poty et Miguet [76].

#### **2.4.2.6.3.1.3 Squelette à partir de la carte des distances**

Les méthodes présentées dans cette section calculent le squelette à partir de l'axe médian après une phase de reconnexion et d'amincissement. Elles sont fondées sur le calcul de cartes de distances. Construire la carte de distance d'une image binaire revient à étiqueter chaque point par sa distance au fond. Cette étape, qui permet d'accéder à une information globale sur l'image, s'effectue uniquement par une série de mises à jour locales, et les points de l'axe médian sont les maxima locaux de la carte de distances. Pour y parvenir, différentes distances ont été introduites et comparées [77], [78] .

Le squelette s'obtient après une phase de reconnexion de l'axe médian, en propageant des chemins le long des lignes de crêtes de la carte de distances [79], [80]. Le résultat est un squelette pondéré, où chaque point est étiqueté par sa plus proche distance au bord.

Ces méthodes sont difficiles à étendre à l'espace 3D, en raison de la phase de reconnexion. Le calcul de cartes de distances 3D et d'axes médians 3D a davantage été abordé jusqu'à présent [81]. En effet, dans la littérature, il existe plusieurs méthodes qui sont fondées sur le même algorithme, avec certaines variations sur ces trois aspects : le choix de la métrique pour le calcul de distance, la détection des maxima locaux et les méthodes d'assemblage des arêtes du squelette.

#### **2.4.2.6.3.1.4 Squelette à partir du diagramme de Voronoï**

De nouvelles méthodes d'extraction du squelette reposant sur le calcul du graphe de Voronoï [82]–[84] Ces méthodes, appelées par la suite méthodes continues, et les données qu'elles utilisent sont un ensemble de points, aux coordonnées réelles, échantillonnant la frontière de l'objet à squelettiser. Le calcul du squelette passe alors par le calcul du graphe de Voronoï de ces points, et il est calculé par une construction incrémentale : « Si la densité de ces points va à l'infini, le diagramme de Voronoï correspondant converge vers le squelette »

Certains auteurs avaient souligné en particulier que la carte de distance est un équivalent discret appauvri du graphe de Voronoï. L'émergence des méthodes continues a été rendue possible pour deux



raisons : d'une part grâce aux efforts menés en géométrie algorithmique pour améliorer le calcul du graphe de Voronoi, d'autre part grâce à la constante amélioration de la puissance des ordinateurs.

Les méthodes continues possèdent de nombreux avantages :

- Le squelette a une structure de graphe qu'il est facile de parcourir.
- Contrairement aux méthodes discrètes (les trois méthodes précédentes), les points du squelette connaissent les points du bord de l'objet qui leur sont les plus proches
- Les méthodes continues sont plus faciles mettre en œuvre que les méthodes calculant le squelette exact d'un polygone.
- Les méthodes continues s'étendent très facilement à l'espace 3D, ce qui n'est pas le cas des autres approches.

La squelettisation basée sur les diagrammes de Voronoï peut répondre aux deux exigences (topologique et géométrique), mais il s'agit d'un processus coûteux, en particulier pour les objets volumineux et complexes.

Pour toutes ces raisons, nous avons choisi de nous intéresser exclusivement à la méthode discrète d'amincissement « Thinning » pour trouver un "bon" équivalent du squelette continu dans le monde discret à partir de la définition de leurs propriétés en faisant une telle amélioration à cet algorithme et pour plus de détails vous pouvez consulter le chapitre 3 suivant.

#### **2.4.2.6.3.2 Squelettisation 3D**

##### **2.4.2.6.3.2.1 Amincissement et propagation en bords « Thinning 3D »**

Comme nous avons déjà dit, Les méthodes d'amincissement tentent de produire une courbe squelette en supprimant itérativement des points simples de la limite de l'objet. Un simple point [85] est un point objet qui peut être supprimé sans changer la topologie de l'objet.

Le processus commence à partir de la limite de l'objet et se poursuit vers l'intérieur jusqu'à ce qu'aucun autre point simple ne puisse être supprimé. A chaque itération, chaque voxel ou objet 3d de frontière est testé par rapport à un ensemble de conditions préservant la topologie et éventuellement supprimé. Les conditions sont généralement implémentées sous forme de modèles (ou de masques), de taille 3x3x3 ou supérieure. Des conditions supplémentaires sont utilisées pour empêcher l'enlèvement de points d'extrémité de surface ou de courbe afin de maintenir les propriétés géométriques de l'objet.

Les méthodes d'amincissement directionnelle n'éliminent les voxels ou les objets 3D que d'une direction particulière dans chaque passage (par exemple : Nord, Sud, Haut, Bas) en utilisant des différents nombres de directions et de conditions pour identifier les points d'extrémité [85] [71] [86] [87] [70]. Pour cette raison, ces méthodes sont sensibles à l'ordre dans lequel les différentes directions sont traitées et les squelettes résultants peuvent ne pas être centrés dans l'objet.

Les méthodes d'amincissement entièrement parallèle [88][89] et non directionnelle [90][91] n'ont pas cet inconvénient. Certaines méthodes d'amincissement produisent un squelette de surface dans la première étape et continuent à amincir jusqu'à l'obtention d'un squelette large d'un voxel [90]; d'autres produisent directement une courbe squelette, et la plupart des algorithmes d'amincissement sont conçus et prouvés corrects pour une connectivité spécifique.

#### **2.4.2.6.3.2 Utilisation d'un champ de distance**

La transformation de distance ou champ de distance est défini pour chaque point intérieur d'un objet 3D comme étant la plus petite distance entre ce point et la limite de l'objet, ainsi, il existe des différentes fonctions de distance qui peuvent être utilisées telles que la distance euclidienne et la méthode de marche rapide [92].

Les crêtes de ce champ de distance correspondent à des voxels qui sont localement centrés dans l'objet. La plupart des méthodes de cette classe tentent de trouver ces voxels. Ceux-ci agissent comme candidats potentiels (à partir du plus grand nombre d'objets voxels) pour les points de la courbe squelette. Les candidats doivent alors être en quelque sorte "taillés" ou aminci pour produire un squelette 1D.

Les valeurs résultantes sont ensuite connectées à l'aide d'une connexion de chemin ou d'une approche d'arbre de recouvrement minimum [93] .Par conséquent, la plupart des algorithmes comportent trois étapes : 1. trouver les points de crêtes (maxima locaux, selles), 2. tailler et 3. connecter ou relier. D'autres méthodes maintiennent explicitement la connectivité en combinant les étapes 2 et 3. et après avoir obtenu les valeurs candidates du volume d'origine, ces points sont regroupés et connectés. Pour la connectivité, la plupart utilisent des arbres de recouvrement minimaux et les chemins les plus courts [94]ou d'autres algorithmes de graphes.

La méthode du champ de distance peut extraire avec précision la surface médiale ; Cependant, ils ne peuvent pas extraire un squelette de courbe d'objets arbitraires sans recourir à des techniques supplémentaires pour élaguer la surface médiale.

Le principal avantage de ces méthodes est que le calcul du champ de distance est très rapide et qu'il est généralement nécessaire à l'application. De plus, pour les objets tubulaires, l'approche par champ de distance fonctionne très bien.

#### 2.4.2.6.3.2.3 Méthodes géométriques

Les méthodes géométriques s'appliquent généralement aux objets représentés par des mailles polygonales ou des ensembles de points dispersés dans un espace continu. Une approche populaire consiste à utiliser le diagramme de Voronoi généré par les sommets de la représentation polygonale 3D ou directement par un ensemble de points non organisés. Le diagramme de Voronoi représente une subdivision de l'espace en régions qui sont plus proches d'un élément générateur (un sommet de maillage dans le cas d'un modèle 3D) que de tout autre élément de ce type. Les arêtes et les faces internes du diagramme de Voronoi peuvent être utilisés pour extraire le squelette de la forme.

Les noyaux et les répétitions M [95][96] sont également des approches de l'axe médian/surface. Un noyau est un locus dans un espace dont les coordonnées sont la position, le rayon et les orientations associées. L'emplacement du noyau représente le milieu de la figure et l'étalement du noyau représente la largeur de la figure. Les répétitions M sont une généralisation du concept noyau. Le M-rep modélise la surface médiane à l'aide d'une " maille " d'atomes connectés. Chaque atome décrit la position, la largeur, le cadre figuratif local qui implique les directions figuratives, et un angle objet entre des positions opposées, correspondantes sur la frontière implicite.

Une structure similaire est l'échafaudage de choc, qui repose sur le concept de sphères de contact [97] et représente l'axe/surface médiale par un ensemble de courbes de choc, définies comme l'intersection de feuilles de surface médiane (et non le squelette courbe).

Les méthodes décrites ci-dessus peuvent être étiquetées en fonction de l'axe médian/surface. Le principal inconvénient des méthodes géométriques basées sur l'axe médian est leur sensibilité au bruit. Par exemple, la forme puissante d'Amenta [98] contient un grand nombre de branches indésirables qui doivent être taillées pour extraire un simple squelette.

De plus, ces méthodes sont plus intensives en calcul que les méthodes basées sur l'amincissement et la distance sur le terrain, il existe d'autres méthodes géométriques qui évitent complètement l'axe médian. Une approche est basée sur le calcul d'ensembles de niveaux d'un graphe géodésique et sur l'extraction d'un squelette en reliant les centres des niveaux adjacents [99]. Le

squelette résultant n'est pas unique pour un objet donné, car il dépend de l'emplacement du point source utilisé pour construire le graphique géodésique.

Li et al. [100] construit un squelette de segment de ligne en réduisant les arêtes dans l'ordre des longueurs (le plus court en premier). Cette méthode est sensible au pavage de maille. Katz et Tal [101] décomposent d'abord une surface de maillage en segments à l'aide du groupement, puis utilisent cette segmentation pour construire un squelette (chaque groupement peut être représenté par un sommet centré). Mais si le maillage d'entrée n'est pas une variété-2, les résultats peuvent être imprévisibles.

#### **2.4.2.6.3.2.4 Fonctions générales sur le terrain**

Cette classe comprend la fonction de champ potentiel [102] [103] où le potentiel en un point à l'intérieur de l'objet est déterminé comme la somme des potentiels générés par les charges ponctuelles à la limite de l'objet ; la fonction de champ électrostatique [104] ; la fonction de force répulsive visible [105] ; la fonction de base radiale [106]. Les points du squelette sont trouvés en déterminant les "puits" du champ et en les reliant à l'aide d'un algorithme de suivi des forces [107] ou en minimisant l'énergie d'un contour actif [106].

Le principal avantage de ces fonctions sur le champ de distance est le suivant qu'ils peuvent produire de jolies courbes sur des feuilles médianes où le champ de distance est constant. En effet, ils prennent en compte des zones plus larges et non seulement la distance au point le plus proche de la frontière. Cela crée également un effet de moyenne qui rend ces algorithmes moins sensibles au bruit limite. Cependant, ils sont beaucoup plus coûteux à calculer.

### **2.4.3 Généralité sur la représentation des objets par Maillage**

#### **2.4.3.1 Introduction**

Avoir une représentation d'une surface ou d'une forme de haute qualité d'un objet est crucial dans de nombreux domaines tels que la mécanique de calcul, la conception assistée par ordinateur et l'infographie.

Lorsque la surface d'un objet est mal représentée, il est difficile de modéliser et de calculer avec précision des caractéristiques géométriques telles que le volume et la courbure afin d'évaluer un système de reconnaissance qui est basé sur ce genre de représentation. En effet, le moyen le plus largement utilisé pour représenter la surface ou la forme d'un objet (2d ou 3d) consiste à utiliser un

maillage triangulaire, et c'est notre idée principale dans le dernier chapitre 5 tout en utilisant les courbes et les surfaces de [108] Beziars.

Généralement, les maillages permettent de partitionner un espace plan en un ensemble de triangles, ce qui permet alors de passer d'un espace continu à un espace discret pour représenter et manipuler des objets tridimensionnels ou bidimensionnels. Alors, on peut également utiliser un maillage d'un objet pour approximer une fonction dont on ne peut pas connaître une forme analytique et qui nécessite dans une première étape, la construction d'un maillage du domaine de calcul.

En restant dans le même thème de la représentation et l'extraction des primitives visuelles des objets, on a s'orienter vers un nouvel axe qui est la représentation des objets par des maillages afin de faciliter le problème épineux de la reconnaissance. Alors, pour représenter et extraire la forme d'un objet 3d, on se base sur des descripteurs de formes en analysant les composantes principales de l'objet par les moments géométriques, les moments de Zernike ou la transformée de Fourier (la section (2.5) pour plus de détails) tout en intégrant une fonction sur le volume de l'objet.

Dans la littérature, quelques catégories de générateurs sont couramment utilisées, les maillages résultants peuvent ensuite être manipulés et regroupés pour créer de nouveaux maillages d'une complexité plus élevée soit en 2D ou en 3D.

### **2.4.3.2 Générateurs de maillage en 2D et 3D**

#### **2.4.3.2.1 Méthode de « transport par projection »**

À l'aide de cette méthode [109] il sera possible de créer le maillage de n'importe quel domaine de topologie simple (un segment, un triangle ou un quadrangle déformé). La subdivision du domaine sera liée à la position et au nombre de points sur son contour et les données de ce maillage sont donc une discrétisation du contour du domaine traité.

Cette approche comprend trois phases :

- Un transport sur les côtés de l'élément unité de même nature que le domaine des points du contour de ce dernier en conservant les distances relatives entre ces points.
- Une connexion des points de maillage d'un côté à l'autre du domaine unité ce qui résulte un maillage canonique.
- Un transport de ce maillage de référence sur le domaine réel.

Ces transformations respectent les sommets et les côtés du domaine à mailler et le maillage résultant sera composé d'éléments de même nature que la zone couverte par l'étude d'un point de vue topologique.[110]

#### 2.4.3.2.2 Méthode « frontale 2D/3D »

Cette approche [110] est adaptée aux géométries a priori quelconques, ce type de maillage [111] va construire le recouvrement du domaine en **triangles/tétraèdres**, à partir **des points/des faces** de son contour. Il utilise donc ce contour comme une donnée, et plus précisément une approximation **polygonale/polyédrique** de ce dernier sous la forme d'une liste de segments.

Le processus de cette méthode est répétitif : à partir d'un contour donné, une analyse des propriétés (angles et tailles) du contour et de leur voisinage permet de sélectionner une zone de départ, ce qui amène à un nouveau contour qui sera formé en éliminant du contour actuel **les arêtes/ faces** du ou des éléments construits communes avec le contour initial et en insérant à leur place la ou **les arêtes/ faces** construites non communes à deux **triangles/tétraèdres**. Ce nouveau contour est alors traité par la même méthode, et quand le contour est vide, le maillage final est obtenu ; puis on peut le normaliser pour obtenir des **triangles/tétraèdres** de meilleure qualité. [110]

#### 2.4.3.2.3 Méthode du type « Voronoi 2D/3D »

Cette méthode [112] [113] est pour absolument toutes les géométries, elle permet de construire un recouvrement de **triangles/tétraèdres**, à partir d'**une partition polygonale/des faces** décrivant du contour du domaine, c'est-à-dire qu'elle est basée sur le même principe que la méthode précédente.

Cette technique est divisée en plusieurs phases [110] :

- Création du nuage de points associé aux données, c'est-à-dire **les points /les faces** du contour du domaine.
- Calcul de la position de **4 points /8 points** supplémentaires de telle sorte que le **quadrilatère/hexaèdre** formé par eux comprend tous les points du nuage.
- Maillage de ce quadrilatère à l'aide de **2 triangles/5 tétraèdres**.
- Insertion, un à un, des points du nuage pour construire un maillage comprenant comme sommets d'éléments ces points.
- Reconnaissance **des arêtes/des faces** données dans ce maillage.
- Élimination des éléments externes à l'objet traité.

- Création de points internes dans ce dernier maillage.
- Régularisation possible du maillage résultant.

#### **2.4.3.2.4 Méthode basée sur la notion de « quadtree »**

Ce type de mailleur [114] [115], va construire le recouvrement du domaine considéré à partir des points de son contour. Un premier rectangle est formé qui contient tous les points du contour. La méthode consiste à découper, à partir de ce rectangle initial, chaque rectangle (père) en 4 rectangles (fils), de façon récursive, jusqu'à obtention d'une partition du rectangle initial en éléments quadrangulaires tels que chacun contienne au plus un seul point du contour. Puis cette partition sera équilibrée par définition de fils de telle sorte qu'au plus un point intermédiaire existe sur une arête d'une maille [110].

Ensuite, le processus passe par une analyse générée des mailles obtenus :

- Si la maille extérieure ne contenant aucun point : une telle maille est détruite,
- Si la maille intérieure ne contenant aucun point : une telle maille produit un quadrangle (qui peut être découpé en triangles) si ses arêtes ne possèdent pas de points intermédiaires ou est découpé en triangles dans le cas inverse
- Si la maille contenant un « morceau » du contour : les points intersections du contour et des arêtes de la maille sont créés, on définit de la sorte une partition de la maille dont on ne garde que la partie interne au domaine (le contour final du maillage est donc formé à ce stade).

Enfin ,ce processus termine par une régularisation effectuée aux points internes du maillage c'est à dire aux sommets hors contour créés pendant la partition (père-fils) [110].

#### **2.4.3.3 Générateurs de maillage en 3D**

##### **2.4.3.3.1 Méthode de type « différence finie »**

Cette approche est parfaitement adaptée aux géométries de type cubique, cette méthode permettra de construire le recouvrement du domaine en hexaèdres (qui pourra être recoupés ultérieurement) à partir des données des points dans 3 directions.

En étant situé dans le plan défini par les deux premières directions, un maillage est construit. Il est formé de quadrangles dont les sommets sont déduits par "convolution" des deux ensembles de points associés aux directions. Chaque élément de cette maille est ensuite utilisé pour construire un

empilement d'hexaèdres par "convolution" avec les points de la troisième direction (figure 2-22). En effet, cette méthode résout déjà le problème du maillage de domaines assez complexes relevant de cette topologie particulière [110].

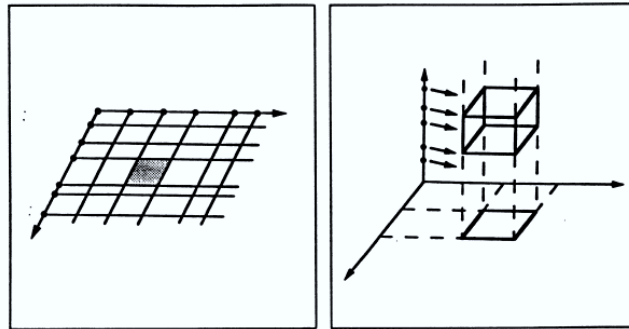


Figure 2-22: Partition 2D et construction 3D associée

#### 2.4.3.3.2 Méthode de « translation-empilement »

Par cette méthode [116] le maillage de tout domaine de la topologie cylindrique peut être créé à partir de la connaissance d'un maillage de référence 2D, y compris les différentes couches d'éléments 3D seront déduites. Les données de ce maillage sont donc le maillage 2D et la définition des sections le long du générateur du "cylindre" ; cette méthode peut donc être considérée comme une généralisation de la précédente.

Chaque segment du maillage 2D est associé à une série de quadrangles dans l'espace, aux triangles sont associés des pentaèdres, et le quadrangle des hexaèdres.

Afin de permettre de traiter des domaines incluant l'axe du cylindre associé, il est nécessaire de considérer les cas de dégénérescence possible, un quadrangle donnant lieu à la création de pentaèdres et non plus hexaèdres [110].

#### 2.4.3.3.3 Méthode par blocs avec « transport projection »

Cette méthode [109] permet de construire le recouvrement par éléments finis à partir de la découpe structurée d'un maillage grossier du domaine. Ce dernier recouvrement est composé de blocs de géométrie simple (lignes, triangles, quadrangles, tétraèdres, pentaèdres, hexaèdres) [110]. et la découpe est guidée par la position relative des points sur les bords de la maille grossière. En fait, cette approche est identique à celle proposée dans le cadre 2D dans le cas d'un seul bloc, et on peut la résumer comme suit :



- Maillage des faces des éléments grossiers [110].
- Maillage des blocs.
- Énumération des éléments issus du découpage pour obtenir les éléments composant le maillage du bloc
- Recollement les maillages des différents blocs.

#### **2.4.3.3.4 Méthode basée sur la notion « d'octree ».**

La méthode présentée en dimension 2 peut être généralisée à la dimension 3 [114] [117] la notion de quadtree est alors remplacée par celle d'octree : un père est un hexaèdre qui permet de définir 0 ou 8 fils. La découpe de la partition composée d'hexaèdres est alors une opération plus complexe qu'en dimension 2, notamment pour obtenir un maillage conforme, les éléments sont découpés en tétraèdres [110].

#### **2.4.3.3.5 Mailleur de surface dans l'espace.**

A partir d'un maillage 2D et d'une fonction de projection spatiale, le maillage de surface correspondant est généré. A partir d'une définition paramétrée ou sous forme de carreaux de la surface, le maillage de la surface est déduit. Les principales méthodes utilisées sont les techniques de "projection" pour les surfaces décrites en blocs de topologie simple, ou du type "Voronoi" pour tous les contours ; ce sont les approches vues dans le cadre 2D généralisé à l'espace de la surface [110].

#### **2.4.3.4 Conclusion**

Ces générateurs de maillages restent la base de plusieurs implémentations de création et reconstruction des objets, de plus, il convient de mentionner d'autres outils de création de mailles comme (treillis, poutres, ferrures, charnières, etc.). Ce qui nous avons amené à l'idée du dernier chapitre (chapitre 5) dont on va créer des maillages 3D des objets 2D détectés dans des images 2.5D tout en utilisant les surfaces de Beziers.

## **2.5 Description et Extraction des caractéristiques**

---

L'extraction des caractéristiques ou des primitives consiste à transformer une image en un vecteur de primitives de taille fixe, de même, on peut la définir aussi comme un processus d'élimination des informations redondantes à partir des données brutes ; ce qui permet de garder que les informations les plus représentatives.

De nombreux types de caractéristiques sont utilisés pour la reconnaissance d'objets. La plupart d'eux sont basées sur les régions ou les limites d'une image. On suppose qu'une région ou une limite fermée correspond à une entité qui est soit un objet ou une partie d'un objet. Certaines des caractéristiques les plus couramment utilisées sont les suivantes :

### **2.5.1.1 Caractéristiques globales**

Les caractéristiques globales sont généralement certaines caractéristiques des régions dans les images telles que la surface (taille), le périmètre, largeur, hauteur, les descripteurs de Fourier et les moments. Les caractéristiques globales peuvent être obtenues soit pour une région en considérant tous les points à l'intérieur d'une région, soit seulement pour les points situés à la limite d'une région. Dans chaque cas, notre objectif est de trouver des descripteurs qui sont obtenus en considérant tous les points, leur emplacement, leurs caractéristiques d'intensité et leurs relations spatiales.

### **2.5.1.2 Caractéristiques locales**

Les caractéristiques locales se trouvent habituellement à la limite d'un objet ou représentent une petite zone distincte d'une région. La courbure et les propriétés connexes sont couramment utilisées comme caractéristiques locales. La courbure peut être la courbure d'une frontière ou peut être calculée sur une surface. La surface peut être une surface d'intensité ou une surface en 2,5 dimensions. Les points à forte courbure sont généralement appelés coins ou sommets et jouent un rôle important dans la reconnaissance des objets. Les caractéristiques locales peuvent contenir une forme spécifique d'une petite limite ou un segment ou une arête de surface ou d'un maillage. Certaines caractéristiques locales couramment utilisées sont la courbure, les segments de limites et les coins.

Généralement, il existe deux types de méthodes de description des caractéristiques : une, statistique et une autre structurelles. Les caractéristiques statistiques sont bien fondées mathématiquement car elles décrivent un motif en un ensemble de calculs statistiques, et cette description donne des informations locales sur le contenu de l'image. Les caractéristiques structurelles décrivent les propriétés topologiques et géométriques dans l'image. C'est une description très informative et discriminante, ce qui permet de prendre des décisions rapides dans la phase de reconnaissance avec une complexité de calcul modéré.

### 2.5.1.3 Description statistique

La représentation d'une image par répartition statistique des points prend en charge les variations de style dans certaines mesures. Bien que ce type de représentation ne permette pas la reconstruction de l'image originale, il est utilisé pour réduire la dimension des caractéristiques en fournissant une vitesse élevée et une faible complexité. Un certain nombre de techniques sont utilisées pour l'extraction de caractéristiques, certains d'entre eux sont : les moments, les histogrammes et les transformées de Fourier.

#### 2.5.1.3.1 Moments

En générale, les moments décrivent des quantités numériques à une certaine distance d'un point de référence ou d'un axe. La description basée sur les moments a été initialement proposée pour de la reconnaissance d'objets. Ils étaient utilisés pour travailler sur des images contenant l'objet entier. L'avantage des moments, tels que les moments de Hu et de Zernike [118], est leur invariance aux translations, rotations et changements d'échelle isotrope. L'une des premiers concepts est celui des moments invariants de Hu [118]; Ils représentent les propriétés spatiales de la distribution des pixels dans l'image soit pour décrire une forme d'un objet dedans une image ,une squelette d'un objet, ou au voisinage d'un point d'intérêt et c'est le cas de ce qu'on va voir au chapitre 3 et 4. Ensuite, les moments de Zernike ont été proposés par Chong et al.[119], ils exigent une précision de calcul inférieure pour représenter les images avec la même précision que les moments réguliers. Les résultats proposés dans la littérature montrent que l'utilisation de ce type de moment est meilleure, notamment en termes de redondance de l'information et de possibilité de reconstruction.

#### 2.5.1.3.2 Histogrammes

En général, l'histogramme représente une estimation de la distribution des intensités de l'image. Swain et Ballard [120]utilisent en 1991 des histogrammes ayant pour objectif la reconnaissance d'objets. De nombreuses méthodes ont été proposées et il est possible de les classer en deux catégories, l'une s'appuyant sur des histogrammes d'intensités lumineuses ou de couleur qui consiste à donner un graphique statistique permettant de représenter la distribution des intensités des pixels, ainsi, il est appliqué à l'image entière, pourtant ,il est possible de l'extraire dans le voisinage d'un point d'intérêt [32], [120], [121] et l'autre, ce sont les histogrammes de gradients orientés dont l'idée est la description de l'apparence et de la forme d'un objet dans une image par la répartition de l'intensité du gradient.[32], [43].

### 2.5.1.3.3 Transformée de Fourier

Les descripteurs de Fourier (DFs) sont parmi les descripteurs les plus populaires pour la reconnaissance de formes et recherche d'images. Ils ont souvent été utilisés pour leur simplicité et leur bonne performance en termes de reconnaissance et facilitent l'étape d'appariement. De plus, ils décrivent la forme de l'objet à différents niveaux de détails. Les descripteurs de Fourier sont calculés à partir du contour des objets. Leur principe est de représenter le contour de l'objet par un signal 1D, puis de le décomposer en séries de Fourier [35]. Les DFs sont généralement connus comme une famille de descripteurs parce qu'ils dépendent de la façon dont les objets sont représentés sous forme de signaux.[32], [122] .

### 2.5.1.3.4 SIFT & SURF

Comme nous avons déjà vu dans la segmentation par les PIs, nous avons donné un certain nombre de détecteurs et de méthodes de détection ; et pour valider ces derniers, les auteurs ont proposé de les comparer à ces deux descripteurs locaux (caractéristiques locales) les plus utilisées et les plus performantes. En effet, la littérature [123] [124] [42] désigne les méthodes SIFT et SURF comme celles ayant les meilleurs résultats en termes de taux d'appariement et de robustesse aux différentes transformations de l'image car ils incluent à la fois la détection des zones d'intérêts et le calcul d'un vecteur caractéristique en chacune de ces zones. A ce qui concerne le vecteur caractéristique, les SIFT sont grossièrement un histogramme des orientations du gradient et les SURF consistent aux calculs d'approximation d'ondelettes de Haar.[32], [122]

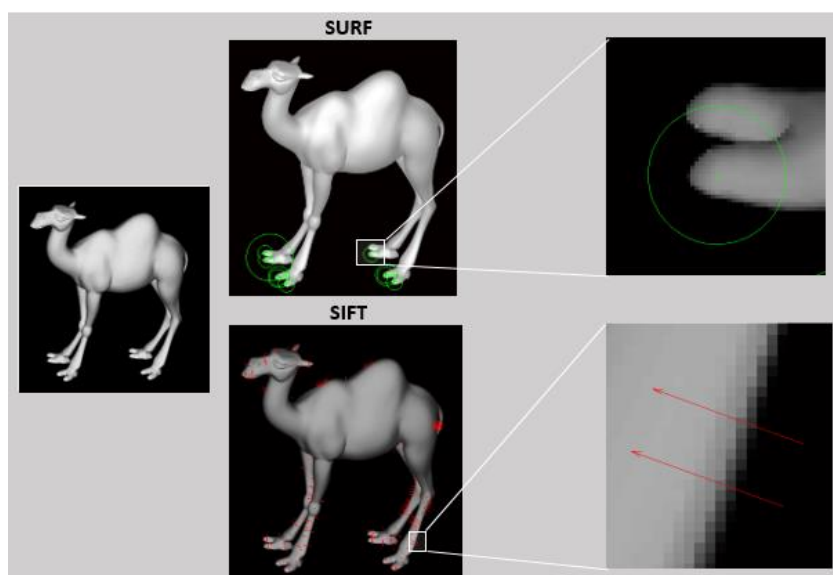


Figure 2-23: Exemple de détection par les descripteurs locaux SIFT et SURF

#### **2.5.1.4 Description structurelle**

Les descriptions structurelles consistent à extraire les propriétés topologiques et géométriques des images avec une résistance élevée aux déformations et aux variations de style. Ce type de représentation peut également encoder certaines connaissances sur la structure de l'objet ou peut fournir quelques connaissances sur les composants constituant cet objet. Diverses représentations topologiques et géométriques peuvent être regroupées en deux catégories : l'extraction de structures topologiques, le codage et la représentation par les graphes.

##### **2.5.1.4.1 Extraction de structures topologiques**

Dans cette catégorie, les formes géométriques simples et leurs attributs (longueur, position, orientation...) sont utilisés comme description de la forme d'objet. Cette description est généralement extraite à partir du squelette ou du contour de l'objet dedans l'image. Parmi ces formes géométriques, les coins qui sont des éléments locaux importants dans les images. En général, ce sont les points qui ont une courbure élevée et qui se trouvent dans la jonction des différentes régions de luminosité des images.

Sans perdre des informations de l'image, l'extraction de coins peut minimiser les données de traitement. Par conséquent, la détection de coin a une valeur pratique et il joue un rôle important dans la représentation de l'image et dans d'autres domaines.

Un grand nombre de détecteurs de coin ont été proposés par les chercheurs. Ces méthodes peuvent être divisées en deux classes principales : une est basé sur le contour et l'autre est basé sur l'intensité. Les méthodes basées sur le contour récupèrent d'abord les contours de l'image, puis recherchent les maximas de courbure ou les points d'inflexion le long de ces contours. Par exemple, Peng et al. [125] ont introduit une méthode de détection d'angle basée sur les contours utilisant la transformée en ondelettes pour sa capacité à détecter des variations brusques. Les méthodes basées sur l'intensité, estiment une mesure qui est destinée à indiquer la présence d'un coin directement à partir des valeurs de niveaux de gris dans une image [126]. Cette catégorie se caractérise par sa rapidité et son indépendance par rapport à d'autres caractéristiques locales. Deux algorithmes représentatifs de cette catégorie : l'algorithme de détection de coins Susan [39] et l'algorithme de détection de coins Harris [36], qui sont les algorithmes de détection de coins les plus utilisés en pratique d'après ce qu'on a déjà vu dans la partie des détecteur de coins, et on verra après une description détaillée de l'algorithme de Susan qui sera abordé dans le chapitre 4.

#### 2.5.1.4.2 Codage

Le codage (chaîne de codes) est basé sur l'idée qu'une représentation d'un contour ou squelette peut être obtenue en stockant la position relative entre les pixels consécutifs dans l'image. Les techniques de codage sont l'une des plus anciennes techniques de vision par ordinateur, initialement introduites par Freeman [127]. Essentiellement, l'ensemble de pixels dans la bordure ou le squelette d'un objet dedans une image est traduite en un ensemble de connexions entre eux. Étant donné un ensemble de points connectés, puis à partir d'un pixel, nous devons être capables de déterminer la direction dans laquelle le pixel suivant doit être trouvé. À savoir, le pixel suivant est l'un des points adjacents dans l'une des directions principales du compas. Ainsi, le code de chaîne est formé en concaténant le nombre qui désigne la direction du pixel suivant. C'est-à-dire, étant donné un pixel, la direction successive d'un pixel au pixel suivant devient un élément dans le code final. Cela est répété pour chaque point jusqu'à ce que le point de départ soit atteint lorsque la forme (fermée) est complètement analysée.

#### 2.5.1.4.3 Représentation par les graphes

Le but est de représenter la forme (squelette ou maillage) par un graphe. Cette représentation est basée sur l'idée que toute forme est capturée par la perception visuelle humaine et non pas comme un ensemble, mais en certaines parties, leurs positions spatiales et leurs relations, jouent un rôle très important dans l'apprentissage et la reconnaissance.

Généralement, un graphe est une représentation mathématique formelle d'un ensemble d'objets et de leurs relations. Chaque objet est appelé un sommet. Les relations entre les objets sont appelées arrêtes. Plus formellement, nous définissons un graphe  $G$  comme une paire ordonnée de  $G = (V, E)$  où  $V$  est un ensemble de sommets,  $E$  est un ensemble d'arrêtes qui définissent la connectivité entre une paire de sommets, et la matrice d'adjacence est une méthode de représentation des graphes.[128]

## 2.6 Conclusion

---

Dans ce chapitre, nous avons présenté, dans un premier temps, des exemples d'approches utilisées dans un système de reconnaissance, de plus, on a abordé les différentes définitions et notions relatives à la chaîne de processus de reconnaissance d'objets. En effet, nous avons décrit les différents éléments et aspects d'un SRO en se basant sur un diagramme comportant les étapes de sa réalisation. Nous avons présenté et expliqué chaque processus du diagramme pour qu'on puisse donner un aperçu général sur les phases d'un SRO.

## **[Vers un fiable (SRO) !]**

Dans un deuxième temps, nous avons concentré notre travail sur la phase du prétraitement et représentation et extraction des caractéristiques des objets détectés dans des images de 2.5 dimensions en expliquant tous ce que nous avons besoin dans notre thèse et en détaillant les différents travaux réalisés dans chaque phase du prétraitement pour qu'on puisse les exploiter dans les trois chapitres suivants.

Choisir une approche pour l'amélioration d'une phase du prétraitement (le cas du chapitre 3 et 4) ou s'orienter vers un nouvel axe qui s'intègre à la représentation et reconnaissance des formes (le cas du chapitre 5) dépend énormément de plusieurs caractéristiques et du temps de calculs requis pour qu'on obtienne des données bien définis et bien structurés pour un fiable SRO. Dans ce travail nous avons opté pour l'utilisation de trois approches de représentation et extraction de forme et des caractéristiques des objets 3D en vue 2D.

---

---

## DEUXIÈME PARTIE : CONTRIBUTION

---

---



## 3 Chapitre 2 : Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes

---

---

### 3.1 Notions de Base

---

#### 3.1.1 Introduction à La squelettisation / L'amincissement

Décrire un objet par une représentation de type "squelette" constitue une étape importante dans un certain nombre d'applications relevant du domaine du traitement des images ou de la reconnaissance de formes. L'avantage du squelette est de préserver en une grande partie la structure topologique des objets ainsi que les informations contenues dans leur contour.

Récemment, d'importants efforts ont été dirigés sur les algorithmes de squelettisation en traitement d'image et reconnaissance des formes. Un certain nombre d'algorithmes ont été proposés pour améliorer l'efficacité de calcul en tentant au même temps, de produire une représentation de l'axe médiane d'une région donnée. Typiquement, ceux-ci sont donc les algorithmes de squelettisation qui suppriment, par des procédés itératifs, les pixels de bord d'une région.

La squelettisation est une méthode qui a pour but de décrire un nouveau descripteur de formes. Elle a été développée à l'origine par Harry Blum dans les années soixante. Cette méthode a gagné l'intérêt de plusieurs chercheurs. Les algorithmes d'amincissement examinent les pixels de l'image binaire soit 2D ou 2.5D et effacent itérativement ceux qui n'appartiennent pas au squelette final.

Les examens des pixels se fait de manière séquentielle ou parallèle.

- **Approche séquentielle** : La suppression d'un point P à la  $n^{ieme}$  itération dépend des opérations faites jusqu'ici mais aussi sur les pixels déjà traités.
- **Approche parallèle** : Les pixels sont examinés de façon indépendante à chaque itération. La suppression de P à la  $n^{ieme}$  itération dépend des opérations réalisées à l'itération précédente.

Le processus d'amincissement peut être considéré comme une érosion des objets de l'image, préservant leurs principales propriétés topologiques, comme la connectivité et l'homotopie. L'application itérative d'opérateurs d'amincissement conduit aux squelettes, qui peuvent être considérées comme des versions compressées des objets. Le processus de squelette est utilisé dans de nombreuses images y compris la compression des données, la reconnaissance des formes, la comparaison des formes, etc. Dans 2-D, ce procédé est bien connu (environ 300 références) et les algorithmes d'amincissement peuvent être divisés en deux catégories ou en deux classes : les

## [Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

algorithmes basés sur les masques et les algorithmes basés sur les contours. La première classe contient des algorithmes dont les opérateurs d'amincissement sont exécutés sur tous les pixels de l'image. De manière différente, les algorithmes appartenant à l'algorithme La deuxième classe n'est exécutée que sur les pixels appartenant aux contours des objets. En ce qui concerne la 3D, il existe plusieurs algorithmes appartenant à la première classe où tous les voxels de la classe sont traités plan par plan, puis ligne par ligne. Mais à notre connaissance, il n'existe aucun algorithme en utilisant la notion de suivi de contour 3D comme en 2D. Dans ce chapitre, nous présentons un algorithme d'amincissement basé sur les images 2.5D tout en utilisant des objets 3D en vue 2D, mais avant on va présenter quelques notions de base en ce qui concerne ce sujet.

### 3.1.2 Définitions et propriétés

#### 3.1.2.1 Définitions

Un des aspects fondamentaux de l'analyse d'images est le passage à une description synthétique des objets. Pour y parvenir, on peut associer à chaque objet son squelette. Ce faisant, On représente les objets par leurs axes de symétrie principaux. Dans le plan, le squelette d'un objet est un ensemble de lignes passant "en son milieu". Dans l'espace, le squelette est un ensemble de surfaces centrées dans la forme. Ainsi, le squelette résume l'aspect général de l'objet, sans tenir compte de son épaisseur.

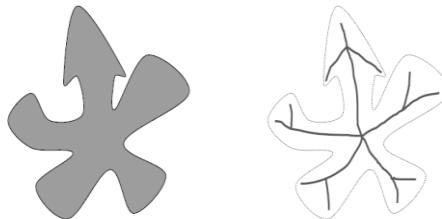


Figure 3-1: Exemple de squelette [129]

Ce concept a été introduit pour la première fois par Blum dans l'espace continu. Sa motivation était de trouver un "bon" descripteur de formes. En particulier, il souhaitait pouvoir décrire de façon pertinente les Objets étudiés en biologie et biomédecine (chromosomes, os, Dès 1978, Blum et Nagel proposent d'utiliser le squelette afin de décomposer les formes, de les classer et d'en mesurer des paramètres. L'engouement pour cette transformation ne s'est pas démenti par la suite. De nombreux travaux ont été et continuent d'être publiés tant sur les propriétés que sur les méthodes de calcul ou les applications du squelette.

## [Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

Il existe plusieurs définitions équivalentes du squelette dans l'espace continu. Par exemple, le squelette peut être vu comme le lieu des points où deux fronts d'ondes en provenance des contours se rencontrent. Nous utilisons par la suite des différentes définitions en ce qui concerne un squelette d'objet :

**Définition 2.1 (Boule maximale) :** Une boule  $B$  incluse dans un objet  $X$  est dite **maximale** s'il n'existe pas d'autre boule  $B'$  incluse dans  $X$  et la contenant strictement.

**Définition 2.2 (Squelette) :** le squelette ou l'axe médian de l'objet  $S$  est l'ensemble des centres des boules maximales de  $S$ , il sera noté par la suite  $Sk(X)$ .

**Définition 2.3 (Squelettisation)** La squelettisation est la transformation qui, à un objet  $X$ , associe son squelette  $Sk(X)$ .

**Définition 2.4 (Axe médian pondéré) :** l'axe médian pondéré de  $X$  est l'ensemble des couples {centre, rayon} des boules maximales de  $X$ .

Le squelette se définit ainsi dans un espace de dimension quelconque. Néanmoins, le plus souvent, il a été étudié et calculé dans le plan.

### 3.1.2.2 Propriétés

Les squelettes présentent quelques propriétés très intéressantes, qui seront exposées dans ce qui suit [130][131] :

**Invariance par translation et rotation :** Le squelette est invariant par translation et rotation. Etant donné une translation ou une rotation  $g$  et un Objet  $X$ , on a  $Sk(g(X)) = g(Sk(X))$ .

**Réversibilité :** A partir des points du squelette et des rayons des boules maximales, il est possible de reconstruire la forme. Ainsi, la squelettisation est réversible à condition d'avoir mémorisé en chaque point  $p$  du squelette le rayon  $r(p)$  de la boule maximale centrée en  $p$ . La fonction  $r$  est parfois appelée fonction d'étanchéité

**Codage :** La propriété de recouvrement de la forme par les boules maximales est à la base d'algorithmes de codage de formes binaires. Néanmoins, ce codage n'est pas optimal en nombre de boules extraites. En effet, certaines boules ne sont pas indispensables à la reconstruction car elles peuvent être recouvertes par deux ou plusieurs autres boules.

## [Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

**Epaisseur :** Il est généralement admis que le squelette est mince. Dans [130] s'interroge sur la signification mathématique du mot "mince". Il montre en particulier que l'intérieur du squelette est vide mais que l'on peut trouver des exemples de squelettes ayant une adhérence de mesure non nulle. Démontrer des résultats généraux sur l'épaisseur du squelette s'avère délicat.

**Structure de graphe :** Sous certaines hypothèses de régularité, il est possible de montrer que le squelette a une structure de graphe. Ainsi, les techniques issues de la théorie des graphes peuvent être appliquées directement aux objets, par exemple pour des problèmes de mise en Correspondance.

**Homotopie :** Dans le cas le plus général, un Objet connexe n'a pas forcément un squelette connexe. Cependant, nous allons supposer par la suite que les Objets continus que nous étudions ont même type d'homotopie que leur squelette.

La définition du type d'homotopie et les hypothèses permettant de démontrer ce résultat sont rappelées dans l'annexe A. Dans le plan, deux Objets homotopes ont même nombre de composantes connexes et pour chaque composante connexe, même nombre de trous. Cette propriété est fondamentale car elle garantit que l'objet et son squelette ont même aspect et justifie l'utilisation du squelette comme descripteur de formes.

**Semi-continuité :** Il est bien connu que la squelettisation n'est pas une transformation continue. Ceci est son principal point faible et a pour conséquence de rendre le squelette extrêmement sensible aux petites perturbations de l'objet.

**Construction :** Calculer le squelette dans le plan ou l'espace ne va pas sans difficulté. De nombreuses méthodes ont été proposées qui sont classées en trois catégories [130]:

- Les méthodes discrètes : L'objet est mémorisé dans une image binaire. Le squelette est redéfini dans un espace discret. Son calcul utilise des techniques propres à la géométrie discrète.
- Les méthodes continues : L'objet est représenté par un ensemble de points échantillonnant sa frontière, ou par une approximation polygonale. Des travaux récents ont montré la possibilité d'approcher le squelette à l'aide du graphe de Voronoï ponctuel ou généralisé. Ces méthodes font appel aux outils de la géométrie algorithmique.
- Les méthodes exactes. L'objet est connu sous forme implicite, explicite ou paramétrique. On désire trouver son squelette exact (sous forme implicite, explicite ou paramétrique). Il existe peu d'Objets pour lesquels ce problème a été résolu.

### 3.1.3 Amincissement vs Axe médiane

L'amincissement est un processus de réduction des données qui érode un objet encore et encore (sans le casser) jusqu'à ce qu'il ait une largeur d'un pixel et produise un squelette de l'objet. Il est plus facile de reconnaître des objets comme des lettres ou des silhouettes ou les objets 3D en regardant leurs os nus, Il existe deux techniques de base pour produire le squelette d'un objet : l'amincissement de base et les transformations de l'axe médian.

Dans ce contexte, le seul paramètre d'amincissement est un paramètre, de sorte qu'il faudrait n'éroder une image qu'une seule fois. Le réglage unique à zéro provoque l'amincissement à continuer à éroder jusqu'à ce que les objets de l'image aient tous une largeur d'un pixel.

Cette technique d'amincissement de base fonctionne bien, mais il est impossible de la recréer. Dans ce contexte, le seul paramètre de l'éclaircissage est un paramètre, de sorte qu'il faudrait n'éroder une image qu'une seule fois. Le réglage unique à zéro provoque l'amincissement à continuer à éroder jusqu'à ce que les objets de l'image aient tous une largeur d'un pixel.

Cette technique d'amincissement de base fonctionne bien, mais il est impossible de la recréer. La recréation de l'objet original du résultat de l'amincissement nécessite la transformation de l'axe médian.

La transformation de l'axe médian trouve les points d'un objet qui forment des lignes en son centre, c'est-à-dire son axe médian. Il est plus facile de comprendre le médial si on comprend d'abord la mesure de la distance euclidienne (ne comprenez pas vous aimez ces grands termes qui signifient vraiment des choses très simples ?). L'euclidienne La mesure de distance est la distance la plus courte entre un pixel d'un objet et le bord de l'objet. La transformation de l'axe médian se compose de tous les points d'un objet qui sont à une distance minimale de plus d'un bord de l'objet. Son avantage c'est que nous pouvons recréer l'original objet de la transformation.

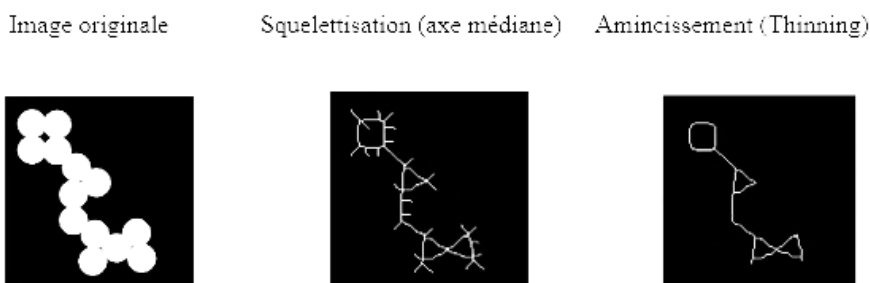


Figure 3-2: Différence entre la squelettisation et l'algorithme Thinning [80]

## 3.2 Approches de l'axe médian vs l'algorithme Thinning

Il existe de nombreux algorithmes de squelettes différents pour 2D et 3D. Bien que certains des algorithmes 2D s'étendraient à la 3D, nous limitons notre discussion à algorithmes explicitement conçus pour la 3D en vue 2D, mais tout d'abord on va présenter dans la suite de ce chapitre les types de ces algorithmes de squelettisation en donnant le principe de chaque type que ça soit les algorithmes appliqués au objets 2D, 3D ou voxels.

La discussion ci-dessous passe en revue le squelette général des courbes en 3D c'est-à-dire la génération d'un algorithme de type courbe 1D sur des objets 3D en vue 2D.

### 3.2.1 Classes des algorithmes de squelettisation

Les squelettes 2D sont des objets d'une épaisseur d'un pixel. En 3D, les squelettes que nous obtenons sont généralement superficiels, mais nous aimerions parfois obtenir des squelettes encore plus réduits. On peut voir sur la figure 3-3 que la surface de l'eau (celui de Gong) peut être réduit en un squelette filaire (celui de Rolland). Cette réduction d'un squelette superficiel à un squelette filaire est obtenue en préservant les caractéristiques de l'objet.

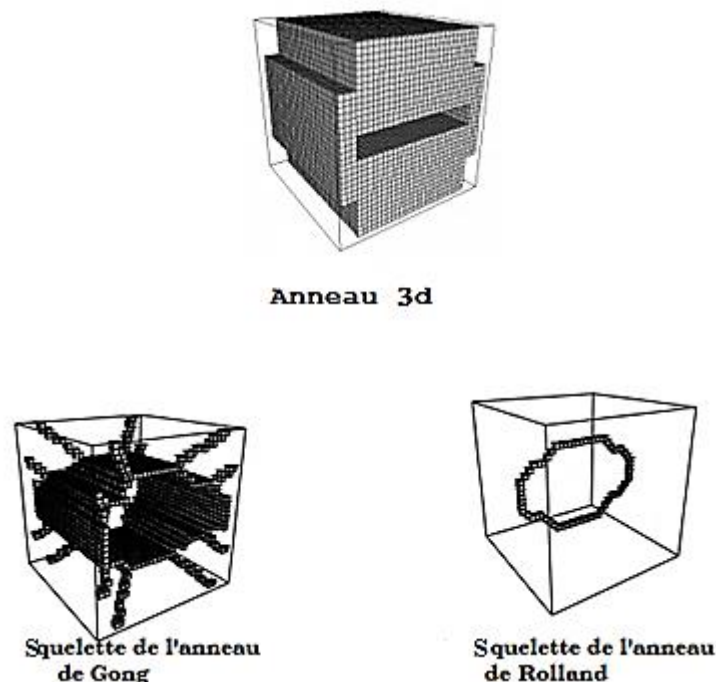


Figure 3-3: Objet et leur squelette selon les types d'anneau

## [Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

Voyons maintenant les différentes classes d'algorithmes qui ont été étudiées dans la littérature. En 2D comme en 3-D, nous pouvons séparer les algorithmes d'éclaircissement en deux classes : les algorithmes basés sur des masques et ceux qui ont basés sur les frontières.

### 3.2.1.1 Algorithmes basés sur des masques

Un opérateur d'amincissement est appliqué aux points de l'image dans l'ordre naturel de la ligne de balayage. Un opérateur d'amincissement peut être vu comme une fonction booléenne ayant les valeurs d'un point et de ses voisins en entrée, et la nouvelle valeur de ce point en sortie. Cet opérateur est appliqué sur tous les points en séquence ou en parallèle. Les opérateurs d'amincissement sont basés sur les critères de connexité :

- Le lien entre les objets et leur complément doit être préservé
- L'opérateur ne doit pas créer ou rejeter un composant
- L'opérateur ne doit pas fusionner certains composants

Ces opérateurs d'amincissement sont généralement basés sur un petit voisinage autour du point donné et c'est le cas de notre algorithme proposé. En 2D, le voisinage 3x3 ou 8-voisinages est habituel dans la littérature, mais nous pouvons étendre ce voisinage à un voisinage de  $K \times K$ . En 3D, un 3x3x3 ou 26-voisinages est aussi habituel.

### 3.2.1.2 Algorithmes basés sur les frontières

Seuls les points appartenant aux frontières des objets sont scannés. Cette classe peut être divisée en deux autres sous-classes :

1. Une sous-classe comprend les algorithmes qui appliquent un des opérateurs d'amincissement décrits ci-dessus, uniquement aux points appartenant aux frontières. Dans cette méthode, les bordures ne sont utilisées que pour réduire le nombre de voxels visités à chaque itération. Tout algorithme basé sur un masque peut être transformé en un algorithme de cette sous-classe puisque les points de fond et les points intérieurs ne sont jamais modifiés par une itération d'un opérateur basé sur un masque.

Les algorithmes de cette classe ont l'avantage de n'utiliser que des calculs locaux, mais leur complexité est souvent difficile à analyser : L'opérateur d'amincissement est appliqué à tous les points de frontière. Seules certaines d'entre elles sont supprimées à une itération donnée. Les autres doivent être reconsidérés lors de la prochaine itération, et éventuellement tout au long du processus s'il s'agit de

### [Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

points appartenant au squelette. Il en résulte des complexités du pire des cas allant jusqu'à  $O(n^3)$  pour  $n \times n$  images et jusqu'à  $O(n^4)$  pour  $n \times n \times n$  uns.

2. L'autre sous-classe comprend les algorithmes qui suivent explicitement les contours des objets, et d'utiliser les contours résultants dans l'opérateur d'amincissement lui-même. De telles approches ont été jusqu'à maintenant utilisé uniquement en 2D, et faire un usage intensif des notions de C-voisins. En 3D bien qu'il existe des algorithmes qui extraient la surface d'un objet, on n'obtient pas la connaissance des travaux antérieurs dans ce sens.

Les algorithmes de la deuxième sous-classe utilisent un suivi de contour explicite, qui est un processus séquentiel et non local. Il s'agit là d'un inconvénient majeur dans l'éventualité d'une mise en œuvre parallèle. Mais ils ont la propriété exceptionnelle de ne visiter chaque point d'objet qu'un nombre constant de fois, de sorte que la complexité de l'algorithme dans le pire des cas peut être majorée par le volume de l'entrée. Comme il s'agit également de la limite inférieure de tout algorithme de squelette, ils sont asymptotiquement optimaux.[131]

#### 3.2.2 Principe de l'Axe médiane

En 2D, l'axe médian d'une forme est défini comme le lieu des centres des disques maximaux inscrits, en d'autres termes, dans le cas continu, il est le lieu des points qui sont équidistants à partir d'au moins deux points sur la frontière de l'objet. L'axe médian d'une forme 2D est constitué d'un ensemble de courbes. En 3D, cependant, le problème devient plus compliqué. Ici, nous parlons d'une surface interne, défini en termes de centres de boules maximales inscrits, qui en plus d'un ensemble de courbes peuvent aussi contenir des éléments de surface. La figure 3-4 suivante montre l'axe médian d'une forme 2D (un rectangle) et la face interne d'une forme en 3D (boîte). A noter que sur la figure 3-4 (b) uniquement le patch horizontal de la surface médiane est ombré, mais en fait, la face interne se compose de neuf tâches différentes déterminées par les lignes tracées en rouge et les bords du bloc. L'axe médian (2D) et la surface interne (3D) sont également connus comme le squelette [131].

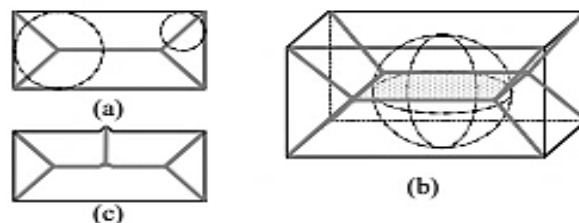


Figure 3-4: (a et c) Axe médian en 2D;(b) Surface médiane en 3D avec un disque inscrits (2D) et une boule (3D).



### [Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

Si la surface interne est augmentée à chaque point le rayon de la boule maximale centrée sur ce point, une reconstruction complète et précise de l'objet d'origine est possible de la surface médiane seul, par des billes de croissance. Cette propriété a une application immédiate de la compression de la forme et de l'animation de volume.

Un inconvénient majeur de la surface médiane (axe) est sa sensibilité intrinsèque de petits changements à la surface de l'objet en raison de la façon dont il est défini. Un exemple illustratif en 2D est représenté sur la figure 3-4 (c) où il peut être observé comment un petit changement dans la forme de l'objet peut générer un grand changement dans l'axe médian.

Dans de nombreuses applications, cependant, une représentation concise des objets 3D avec des arcs de courbes ou de lignes droites est souhaitable en raison de sa simplicité. Par exemple l'animation utilise traditionnellement un IK (cinématique inverse) squelette composé d'un petit nombre de segments de ligne connectés représentant le torse, les bras et les jambes, pour les applications par exemple la navigation virtuelle, exigent également un ensemble de chemins de courbe pour la traversée. Cette représentation de la ligne en forme d'un objet 3D est également connu comme la ligne centrale ou du courbe squelette et est une représentation simplifiée de 1D de son surface interne qui est composé uniquement de courbes [131].



Figure 3-5: Exemples de courbes squelettes de différents objets 3D [131]

#### 3.2.3 Elagage du squelette

L'inconvénient majeur du squelette est sa sensibilité au bruit sur le bord de la forme. Ce bruit est la présence de pixels parasites qui s'ajoutent de façon aléatoire sur le contour de la forme. Une perturbation du bord de l'objet n'entraîne une modification que d'une partie limitée du squelette. Néanmoins, cette modification peut être importante. Les détails du bord qui semblent négligeables visuellement par rapport à l'ensemble de la forme peuvent ainsi générer de longues branches. La longueur des branches n'est pas proportionnelle à la taille de la perturbation dont elle est issue. En d'autres termes, plus une forme est épaisse au niveau de la perturbation et plus la branche issue de cette

## [Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

dernière sera longue pour rejoindre le squelette et prenons la figure 3-4(c) comme exemple, le rectangle dispose d'une perturbation. C'est la raison pour laquelle nombre de chercheurs pensent qu'il est nécessaire d'attribuer plus ou moins d'importance aux branches du squelette dans le cas où celui-ci est utilisé.

Les méthodes de squelettisation sont généralement sensibles au bruit, ce qui crée des branches intéressantes dans le squelette. Certains chercheurs [129] choisissent de leur donner une importance négligeable, d'autres préfèrent supprimer définitivement les branches qu'ils considèrent inutiles afin d'accroître la stabilité au bruit du squelette. L'élagage du squelette vise donc à supprimer les branches causées par le bruit sur le contour de la forme (celles qui ont le moins de signification) tout en respectant la topologie de la forme. C'est pour cela, il faut réaliser un post-traitement sur le squelette en attribuant des valeurs d'importance à chaque branche, c'est-à-dire, il faut amincir le squelette puis l'élaguer branche par branche, c'est ce que nous avons fait dans notre algorithme, on a pris une décision binaire concernant la conservation d'une branche en fixant un seuil minimal d'importance, et toute partie à supprimer ayant une valeur d'importance inférieure au seuil fixé pour ne conserver que les branches importantes et avoir un squelette plus robuste au bruit.[132]

### 3.2.4 Principe de l'algorithme Thinning « Amincissement »

L'amincissement joue un rôle crucial dans les applications d'analyse d'image et de reconnaissance de formes. C'est l'une des méthodes de prétraitement les plus fréquemment utilisées pour analyser différents types d'images (2D,2.5D,3D). L'amincissement consiste essentiellement à réduire un objet numérique épais en un squelette mince. Il existe plusieurs algorithmes d'amincissement pour obtenir un squelette d'une image dans la littérature.

#### 3.2.4.1 Amincissement et la propagation des bords

Les méthodes d'amincissement tentent de produire un squelette courbe en l'élimination itérative des voxels de la limite d'un objet jusqu'à ce que la minceur requise soit obtenue. Tous les amincissements fonctionnent dans l'espace discret et s'appuient sur les algorithmes sur le concept de point simple, Un simple point est un point d'objet (voxel) qui peut être enlevé sans changer la topologie de l'objet.

Une propriété importante des points simples est qu'ils peuvent être caractérisés localement, c'est-à-dire qu'on peut déterminer si un objet (voxel) est simple ou non en inspectant seulement son voisinage local, ce qui rend les algorithmes d'amincissement beaucoup plus efficaces.

## [Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

Le processus d'amincissement commence à partir de la limite de l'objet et se poursuit vers l'intérieur jusqu'à ce qu'aucun autre point simple ne puisse être enlevé. A chaque itération, chaque objet de frontière est testé par rapport à un ensemble de conditions de topologies et éventuellement supprimées. Les conditions sont généralement implémentées sous forme de modèles (ou de masques), de taille 3x3x3 ou supérieure. Le centre d'un masque est placé sur le voxel testé et couvre tout son voisinage local. Chacun des objets (voxels) du masque a une valeur de "0", "1" ou "je m'en fiche". Une valeur de "0" doit correspondre à un voxel de fond, une valeur de "1" doit correspondre à un voxel objet, tandis qu'un " je m'en fiche " peut correspondre à un voxel de fond ou un voxel objet.

Cependant, la suppression de tous les points simples de l'objet entraîne un écartement excessif des branches du squelette courbe. En effet, tous les points d'extrémité des courbes du squelette sont des points simples en soi (c'est-à-dire que leur suppression ne changera pas la topologie de l'objet). Des conditions supplémentaires sont utilisées pour empêcher l'enlèvement de points d'extrémité de surface ou de courbe afin de maintenir les propriétés géométriques de l'objet.

Il existe plusieurs sous-classes de méthodes d'amincissement basées sur la façon dont les points simples sont détectés et pris en compte pour les l'enlèvement.

### Algorithme : Généralisation de la méthode d'amincissement « l'algorithme

#### Thinning »

*Thin* (X: object, C: condition) → X: object

- 1: repeat
- 2: {parallel labelling of points which belong to P}
- 3:  $P \leftarrow \emptyset$
- 4: for each point x in X, in parallel, do
- 5: if C (x, X) then
- 6: put x in P
- 7: end if
- 8: end for
- 9: {parallel deletion of P-simple points}

## [Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

```
10:  $X' \leftarrow X$ 
11: for each point  $x$  in  $P$ , in parallel, do
12: if  $x$  is  $P$ -simple for  $X'$  then
13: delete  $x$  in  $X$ 
14: end if
15: end for
16: until no points are deleted
```

Figure 3-6: Généralisation de l'algorithme Thinning

### 3.2.4.2 Thinning sur les objets voxels

#### 3.2.4.2.1 Principe

Sans perte de généralité, et pour simplifier la discussion dans le reste de ce chapitre, nous supposons qu'un objet 3D se réfère à une discrétisation 3D voxélisée. Les modèles polygonaux peuvent être convertis en objets volumétriques par voxélisation. À l'application de l'algorithme Thinning sur les objets voxels, l'ensemble de données 3D qui en résulte est un champ de vecteurs ou un tableau 3D où chaque voxel contient une valeur de vecteur et une fois nous calculons ce champ de vecteurs, nous utilisons ses caractéristiques topologiques tels que les points critiques etc... pour extraire notre squelette hiérarchie. La méthodologie présentée a un certain nombre d'avantages significatifs par rapport aux méthodes précédentes (la méthode géométrique, Distance Field, Potentiel Field), à savoir : elle produit directement des squelettes connectés sans l'utilisation d'une étape de reconnexion (figures 3-7,3-8,3-9).

#### 3.2.4.2.2 Implémentations et Résultats d'algorithme « Thinning » sur les objets 3D

Ici, nous allons présenter les résultats implémentés sur les objets voxels par l'algorithme « Thinning » et deux autres méthodes pour voir la différence entre leurs squelettes extraits :

[Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

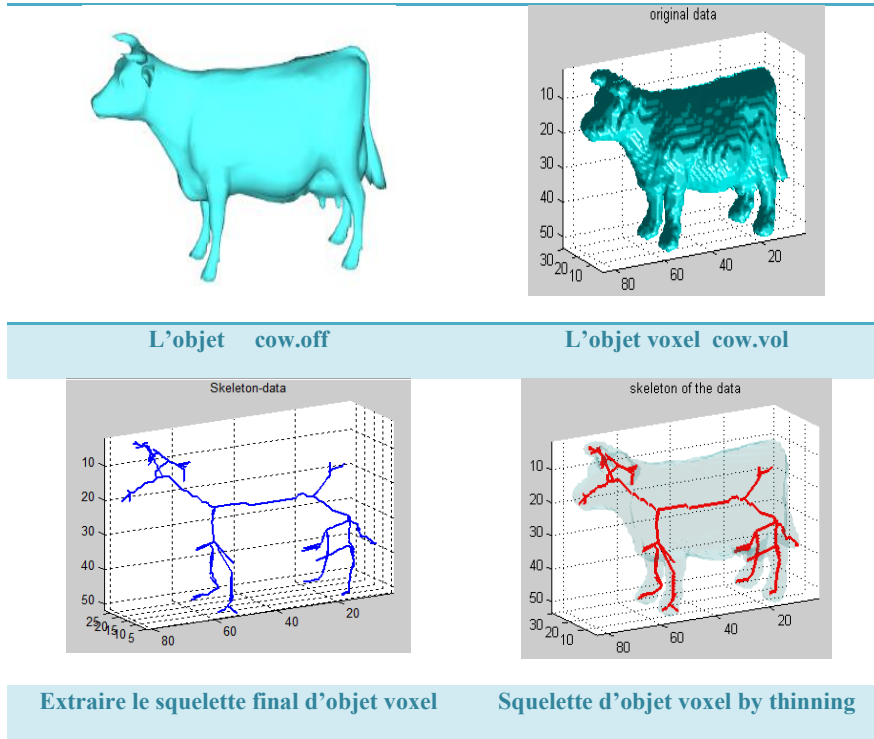


Figure 3-7: Méthode Thinning appliqué au voxel-temps d'exécution 152.508498 seconds

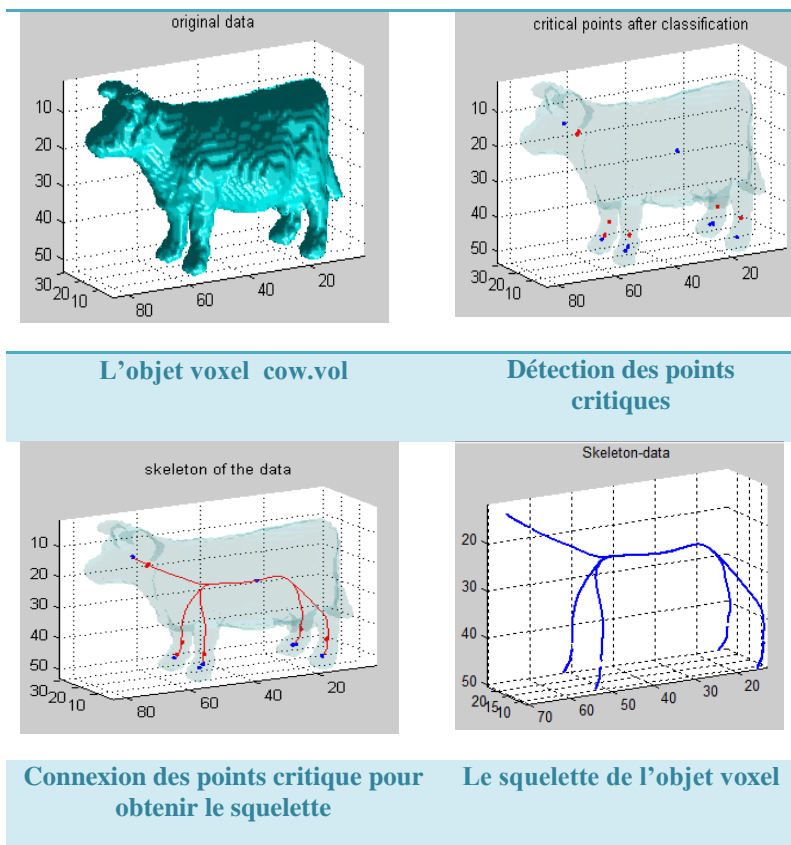


Figure 3-8: Méthode répulsive potentiel Field appliqué au voxel[102]

## [Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

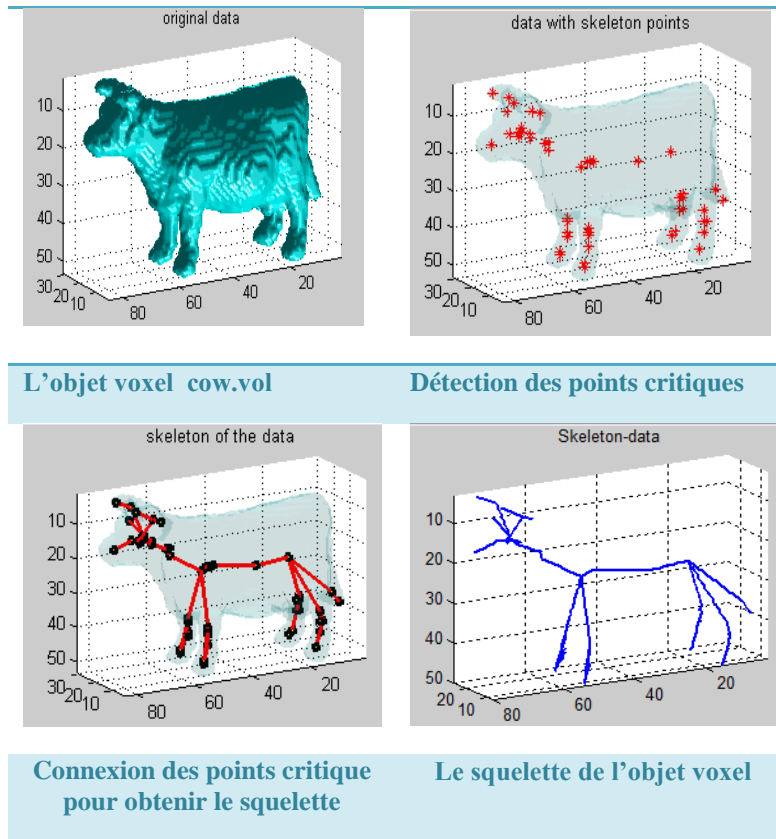


Figure 3-9: Méthode Distance Field appliqué au voxel [93]

### 3.2.4.2.3 Interprétations et observations

Nous remarquons qu'à partir des trois exemples de méthodes d'extraction des squelettes, la seule méthode qui fait pas la connexion des points critiques c'est la méthode Thinning et nous observons qu'elle nous a donné un squelette précis et exact et trop proche au squelette anatomique de l'objet concerné (et c'est la chose qui nous intéresse pour comparer après ce squelette résultant par un objet voxel 3D ou un objet 3D dedans une image 2D ,ce qui amène à obtenir un bon system de reconnaissance), malgré qu'elle n'est pas lisse, mais en tous cas on peut reconnaître rapidement et facilement le type d'objet à partir de son squelette extrait, et d'autre part on voit que les deux dernières méthodes ont passé par l'étape de reconnexion des points critiques ce qui rend que les deux dernières méthodes ne génèrent pas une courbe squelette directement et les points de squelette résultant doivent être élagués puis connectés pour obtenir une courbe squelette bien définie.

## [Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

### 3.2.5 Approche de l'algorithme Thinning développée sur les objets 3D en vue 2D

Pour obtenir un squelette exact et continue, il faut éviter son décalage par rapport à l'axe médian et la suppression de la forme ou des parties de l'objet, gagner le temps d'exécution quel que soit le nombre des itérations pendant l'implémentation tout en préservant la topologie de l'objet à squelettiser.

Dans ce chapitre, nous allons développer l'algorithme d'amincissement « Thinning » des modèles numériques binaires qui est appliqué aux objets 3D en vue 2D.

Cet algorithme produit un squelette qui élimine itérativement les points qui limitent l'objet 3D concerné commençant par les bords de ce dernier sans modification de la topologie de l'objet (voir [85] [76], [131] pour une revue complète de la topologie numérique).

Le processus général de cette méthode commence à partir de la limite de l'objet et continue vers l'intérieur jusqu'à ce que plus un seul point ne puisse être retiré, et à chaque itération, chaque objet dedans l'image est testée par rapport à un ensemble de conditions et le maintien de la topologie peut être enlevé.

Les conditions sont généralement mises en œuvre sous forme de deux phases « phase 1 et phase 2 » ou plutôt « masque 1 et masque 2 » dont la taille est 3x3. (Figure 3-10 et 3-11)

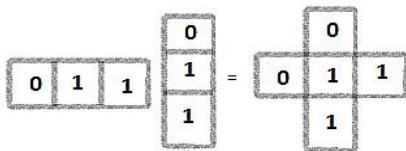


Figure 3-10: Masque 1

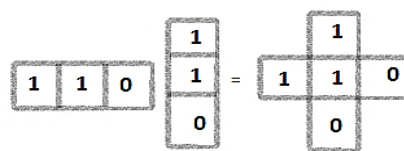


Figure 3-11: Masque 2

La première phase réduit l'épaisseur de l'objet 3D testé trop alternativement et adoucissant en supprimant les points de contour, et la seconde phase fait le réglage de la phase résultante de la première phase pour obtenir un axe centré de l'application qui est le squelette de l'objet 3D et ce processus en général s'intègre dans le principe de la propagation des bords (voir la démarche ci-dessous).

De plus, des conditions supplémentaires sont utilisés pour empêcher le retrait de la surface à la fin de la courbe et de maintenir les propriétés géométriques des points de l'objet. La méthode directionnelle supprime les objets dans une direction particulière à chaque fois (par exemple, du Nord, du Sud, Haut, Bas) en utilisant des nombres différents de directives et conditions pour identifier les critères d'évaluation [71][87].

## [Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

Par conséquent, ces méthodes directionnelles sont sensibles à l'ordre dans lequel les différentes directions sont traitées et les squelettes en résultent ne peuvent être centrés dans l'objet et pas de méthodes d'amincissement directionnelles entièrement parallèles qui n'ont pas de désavantage entre ces méthodes, par contre notre algorithme actuel se base sur une méthode parallèlement non directionnelle testée aux objets 3D en vue 2D et qui finit par un élagage pour obtenir un squelette bien défini et non bruité. Les résultats obtenus montrent que l'algorithme de l'extraction des squelettes produit des squelettes minces et centrés avec moins de distorsion et mieux formés que d'autres et d'une bonne connectivité spécifique et disant ayant presque toutes les propriétés précédentes (3.1.2.2).

### 3.2.5.1 Démarche de notre algorithme

*1. Trouvez un emplacement de pixel (i, j) où les pixels de l'image correspondent à celles de modèle T1. Avec ce modèle tous les pixels sur le haut de l'image sont enlevés déplacent de gauche à droite et de haut en bas.*

*2. Si le pixel central n'est pas un point final, et a le numéro de connectivité = 1, puis marquer ce pixel pour la suppression.*

*Endpoint pixel : un pixel est considéré comme un point de terminaison s'il est connecté à un seul autre pixel. Autrement dit, si un pixel noir a un seul voisin noir des huit voisins possibles.*

*3. Répétez les étapes 1 et 2 pour tous les emplacements de pixels correspondant à T1.*

*4. Répétez les étapes 1-3 pour le reste des modèles : T2, T3 et T4.*

*T2 correspondra pixels sur le côté gauche de l'objet, se déplaçant de bas en haut et de gauche à droite. T3 sélectionner des pixels le long du bas de l'image et déplacer de droite à gauche et de bas en haut. T4 de repérer les pixels sur le côté droit de l'objet, se déplaçant de haut en bas et de droite à gauche.*

*5. Suppression des branches sans importance pour générer un squelette visuel optimal (élagage).*

*6. Réglez à noir les pixels marqués pour suppression ayant une valeur de '0'*

*7. afficher le squelette extrait.*



[Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

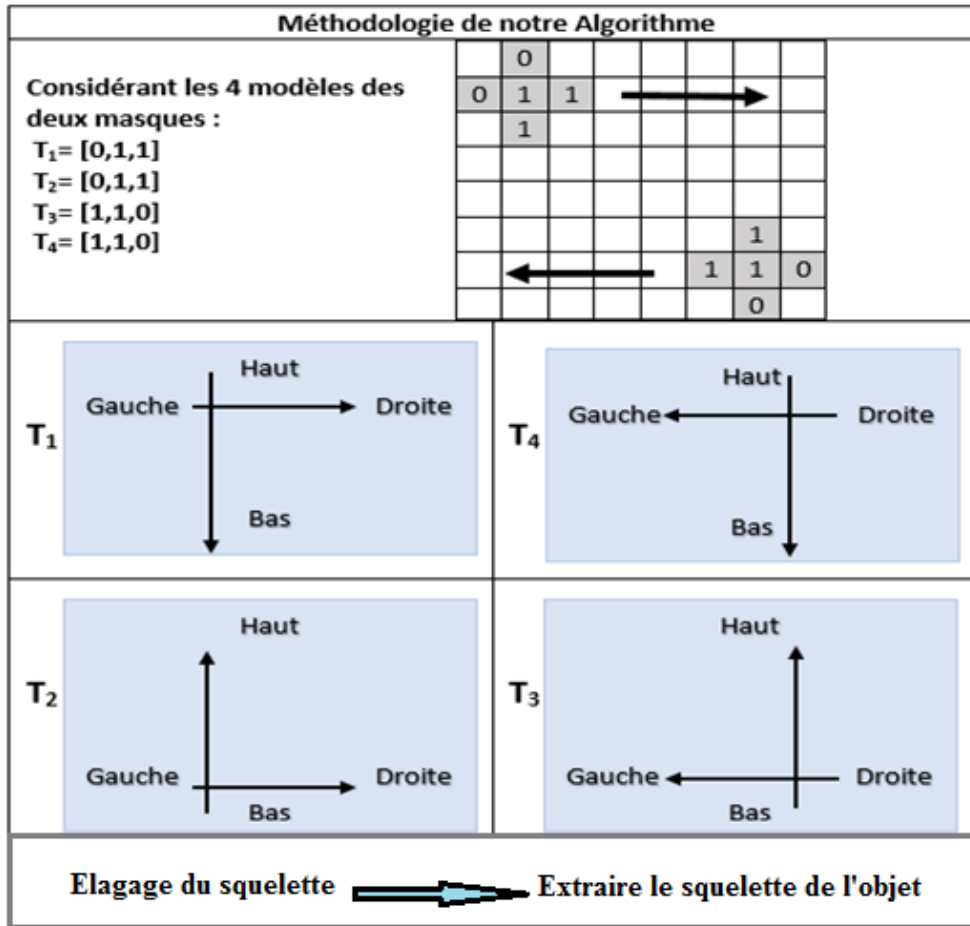


Figure 3-12: Démarche de notre Algorithme

### 3.2.6 Résultats et Discussions

Les résultats suivants montrent que l'algorithme d'amincissement et d'effacement alterné produit des squelettes qui sont un peu adoucis, avec le moindre de distorsion et mieux formées que les autres.

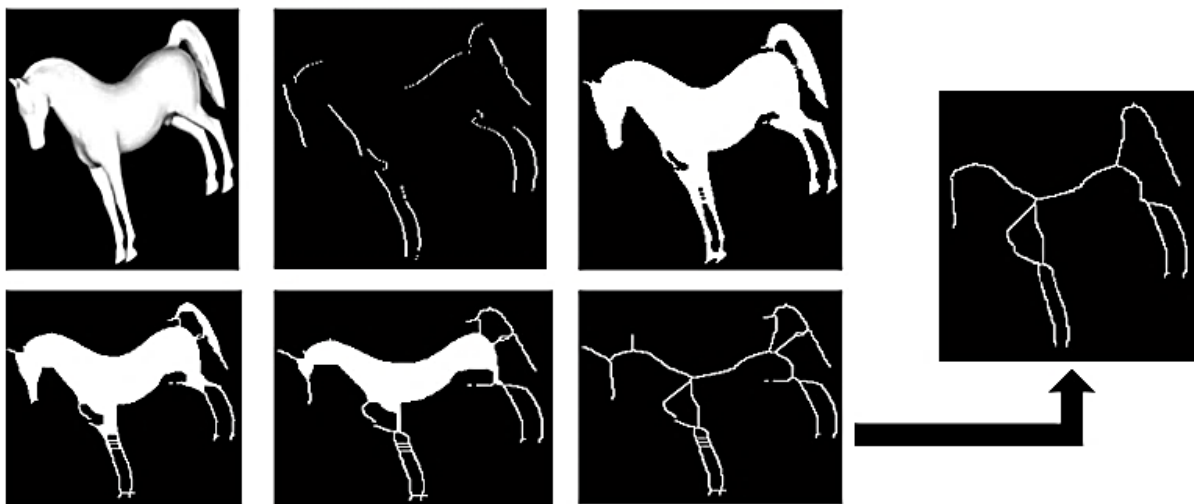


Figure 3-13: Obtention d'un squelette d'un cheval après le passage par quelques itérations par le masque 1 et 2

[Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]







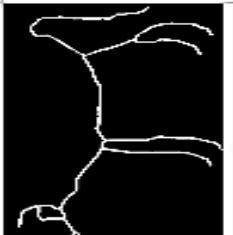
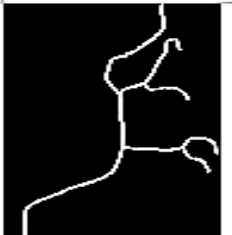
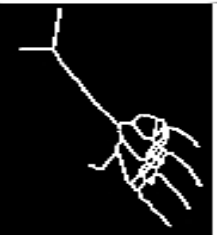



Squelette			
Objet 3D			
Nom d'objet 3D	Dinohol	Chameau	Aigle
Squelette			
Objet 3D			
Nom d'objet 3D	Vache	Dinoflat	Main

Figure 3-14: Obtention des squelettes à partir des objets 3D en appliquant notre nouvelle approche

[Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

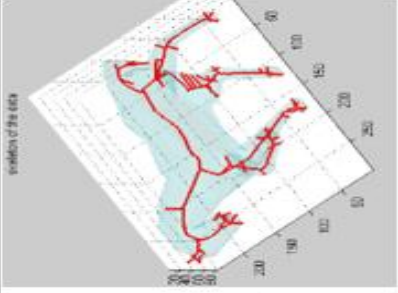
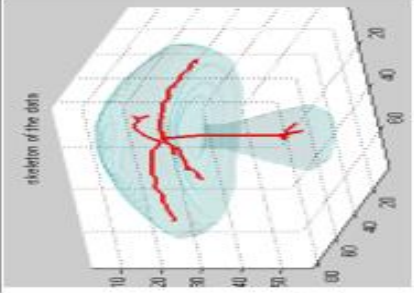
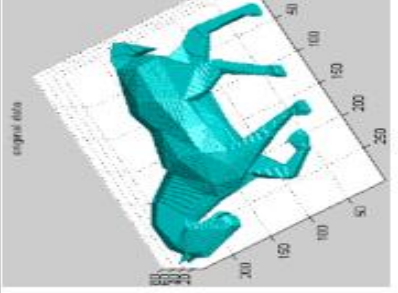
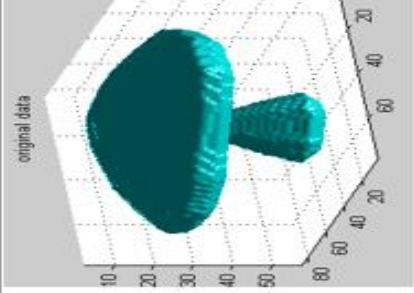
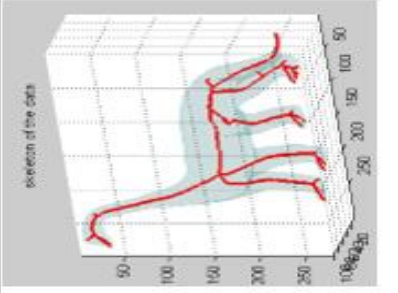
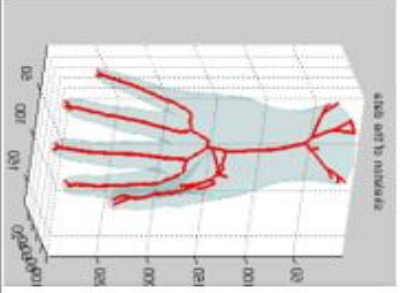
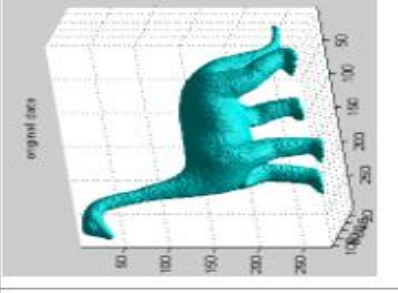
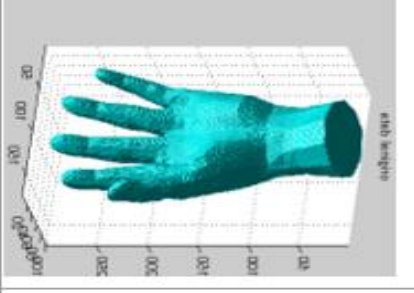
Squelette		
Objet Voxel		
Name of voxel obj	Horse	Mushroom
Squelette		
Objet Voxel		
Name of voxel obj	Dinohol	Hand

Figure 3-15: Obtention des squelettes à partir des objets 3D voxels

[Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

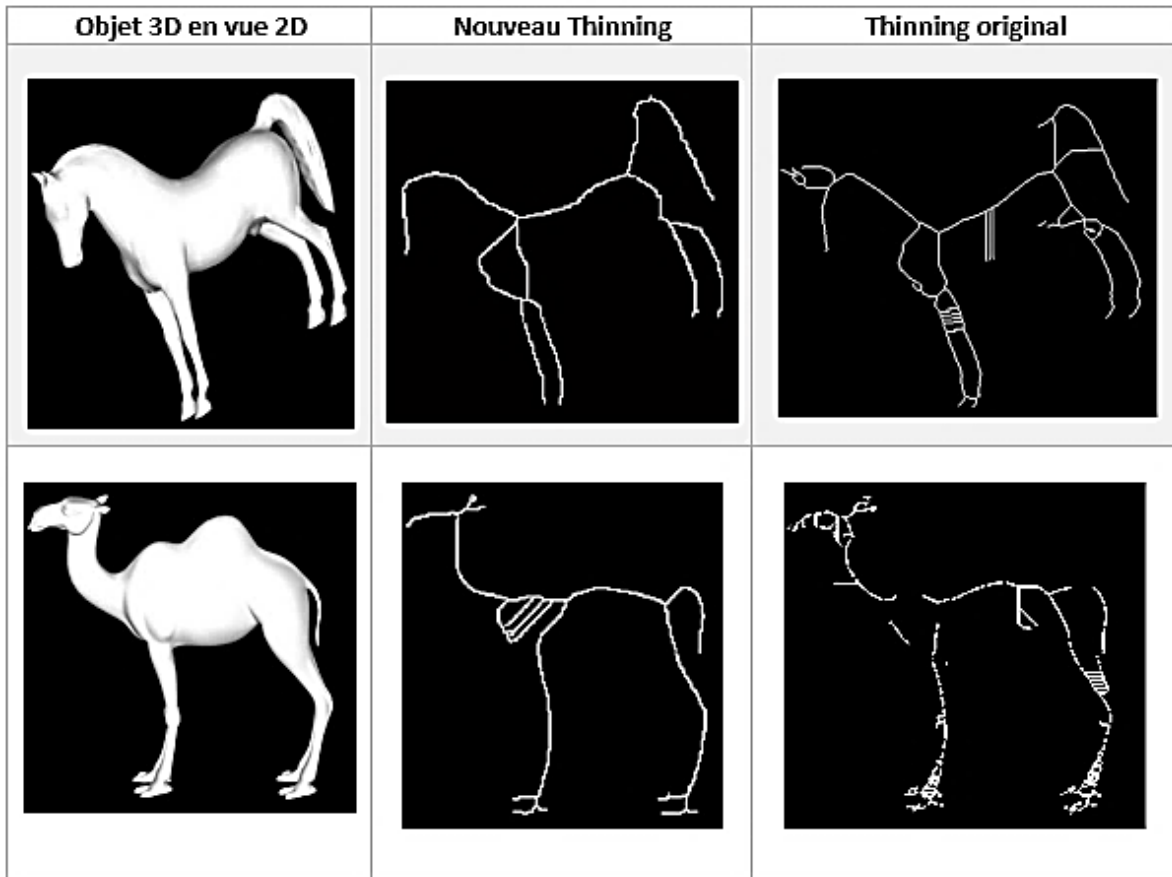


Figure 3-16: Comparaison entre le nouveau thinning(approche proposée) et l'ancien thinning(original thinning-fonction Matlab-)

Selon une étude comparative entre les objets 3D voxels qui sont images 3D comme :la vache, le cheval, le Dinosaur, la main...etc., et les objets 3D en vue 2D, nous voyons que la détection des squelettes de ces derniers est plus rapide par rapport aux objets voxels (Tableau 3-1), plus qu'elle est nette, parfaite et continue, et c'est beaucoup mieux que le squelette extrait des voxels qui sont discontinues en quelques noeux de connectivité ,De plus elle ressemble aussi au squelette anatomique de l'objet concerné. Tant que les objets 3D voxel demandent un temps de calcul important, la taille des caractéristiques sera prise en compte, donc, nous pouvons conclure que le résultat obtenu dépend de la taille des voxels. Et si on dispose seulement des squelettes inconnus (extraits) sans savoir de quel objet s'agit-il ! On peut faire facilement la reconnaissance de l'objet 3D concernant le squelette présenté.

Plus que ça ,dans la figure 3-16 ,nous pouvons clairement voir les résultats obtenus d'original Thinning 2D et le nouveau Thinning, ce qui nous montre la grande différence entre les deux et le rôle de la taille de l'image 2D qui contient un objet 3D dans le traitement par rapport au objet 3D testé directement ,ce qui gaspille beaucoup de temps de calcul (en 3 dimensions) pendant l'exécution de

**[Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]**

l'algorithme ,et en effet , nous obtenons des squelettes discontinues et qui ne présentent pas bien la structure de l'objet testé.

**3.2.6.1 Complexité temporelle**

Tableau 3-1: Comparaison au niveau du temps d'exécution

Objet3D/Temps d'exécution Nom d'objet 3D	Objets 3D via vue 2D	Objets 3D voxels
	Temps d'exécution T(s)	Temps d'exécution T(s)
Cheval	59.35	92.24
Main	60.21	228.9
Dinosaure	50.3	70.8
Vache	50.8	152.5
Moyenne	55.165	136.11
Complexité d'algorithme	$O(n^2)$	$O(n^3)$

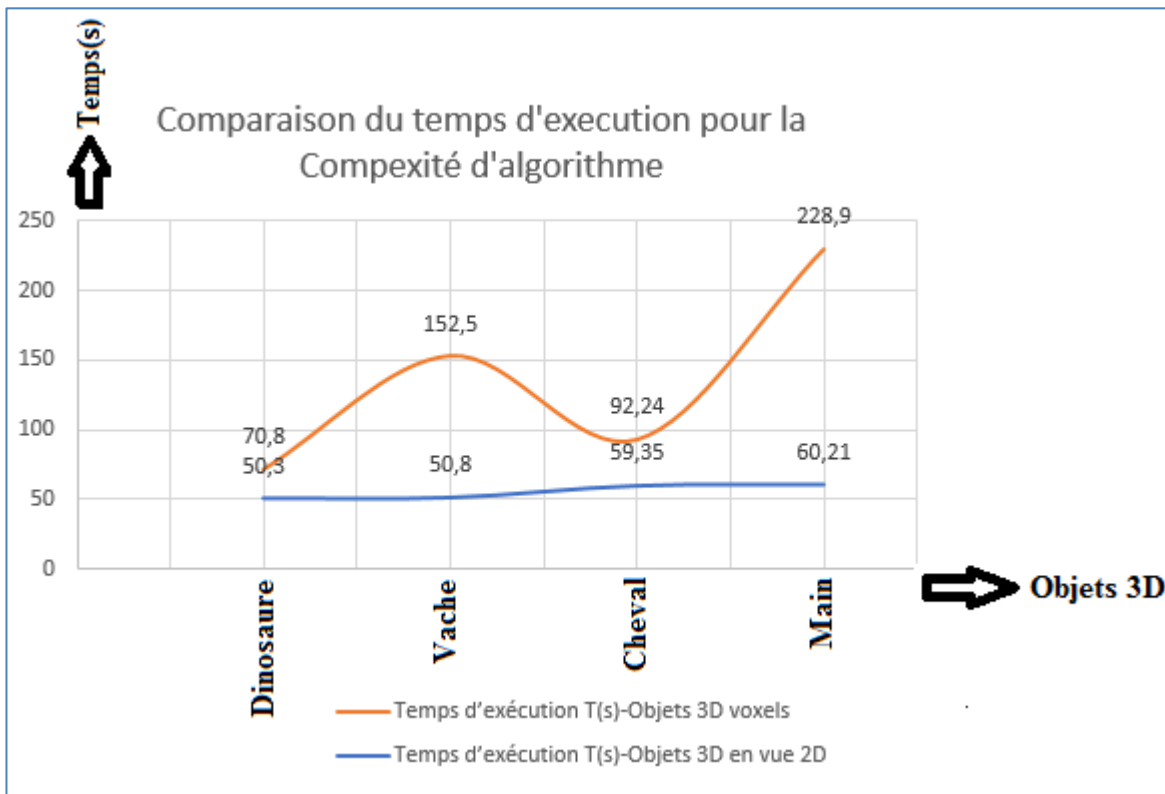


Figure 3-17: Comparaison au niveau du temps d'exécution

## [Représentation des formes d'objets 3D en vue 2D par l'extraction des squelettes]

Dans le tableau 3-1, nous avons calculé la complexité de chaque algorithme et bien étudié leur efficacité en comparant les deux algorithmes d'amincissement appliqués aux objets voxels au niveau du temps d'exécution en raison de la complexité qui conduit à une classification des problèmes liés à la performance des algorithmes les plus connus que la résolution. A priori, nous ne pouvons pas toujours mesurer le temps de calcul sur toutes les entrées possibles d'un algorithme ce qui nous amène à calculer la complexité temporelle de notre approche et de l'autre approche comparative en exécutant cet algorithme sur un ordinateur d'un processeur Intel Core i5-4200M, CPU @2.50HZ, d'une mémoire vive (RAM) de 4Go et dont le système d'exploitation est Windows 7 64bits.

Pour l'original algorithme de Thinning contient une seule boucle, ce qui nous donne une complexité linéaire  $O(n)$  qui ne gaspille pas trop de temps où le  $n$  est le nombre de pixel total de l'image pendant l'implémentation, mais cet algorithme ne donne pas des bons résultats pour que nous puissions les comparer à notre algorithme proposé.

Notre but ici, c'est de comparer la complexité du temps de l'algorithme proposée et testé sur les objets 3D en vue 2D et de l'algorithme appliqué aux objets 3D voxels. Notre algorithme dispose d'une complexité quadratique  $O(n^2)$  car il contient deux boucles imbriquées l'une dans l'autre. Cette complexité est plus petite que la complexité cubique  $O(n^3)$  de l'autre algorithme testé sur les objets 3D voxels, et qui contient trois boucles imbriquées l'une dans l'autre, et deux autres boucles imbriquées aussi l'une dans l'autre et répété deux fois et trois boucles simples, ce qui nous montre que notre méthode est plus rapide et efficace pour donner de bons résultats que l'autre méthode comparable tout en utilisant des objets 3D en vue 2D à la place des objets 3D directement, et voir ci-dessous un tableau comparatif entre les deux complexités étudiées en général:

Tableau 3-2: Comparaison entre deux grandeurs du temps nécessaire à l'exécution d'un algorithme d'un type de complexité [133]

time time for n=x	$O(n^2)$	$O(n^3)$
n=5	250 ns	1.25 $\mu$ s
n=10	1 $\mu$ s	10 $\mu$ s
n=20	4 $\mu$ s	80 $\mu$ s
n=50	25 $\mu$ s	1.25 ms
n=250	625 $\mu$ s	156 ms
n=1000	10 ms	10 s
n=10 000	1 s	2.7 hour
n=1 000 000	2.8 hour	316 years

### 3.3 Conclusion

---

Le présent chapitre a permis d'avoir une idée sur la façon dont les algorithmes d'amincissement ont été utilisés pour extraire les squelettes des formes et des objets 3D en vue 2D. En effet, Il y eu de nombreux algorithmes de différents courbe-squelettisation proposés dans la littérature. Dans ce chapitre, notre travail concerne l'étude, le calcul, l'extraction et la simplification des squelettes d'objets 2D et 3D qui sont utiles pour la description et la reconnaissance de formes et des objets 3D, la quantification et la mise en correspondance, etc. Dans un premier temps, nous recensons les différentes techniques de calcul du squelette qui sont testés sur des images binaires avec des outils de la géométrie discrète. Dernièrement, une nouvelle famille de méthodes, appelées méthodes continues a vu le jour. Le squelette est approché à l'aide du graphe de Voronoï d'un échantillonnage de la frontière, et se calcule par des moyens propres à la géométrie algorithmique. Notre intérêt s'est porté sur une nouvelle tentative pour améliorer l'algorithme Thinning appliqué sur des objets 3D en vue 2D en comparant les résultats d'un autre algorithme d'amincissement appliqué sur les objets 3D voxels (image 3D). En effet, pour finir, nous avons réalisé une application pratique et qui valide notre approche.

## 4 Chapitre 3 : Représentation des formes libres des objets 2D/3D en vue 2D par l'extraction des P.Is

---

---

### 4.1 Introduction

---

Aujourd'hui, les méthodes de vision par ordinateur sont utilisées dans de nombreuses applications telles que la vidéo-surveillance, la conduite ou la reconstruction 3D. Ces différentes applications s'appuient généralement sur des procédés de reconnaissance de formes ou de suivi. Pour ce faire, l'image est analysée afin d'en extraire des amers ou des primitives (contours, fonctions d'intensité ou modèles morphologiques). Les méthodes les plus courantes s'appuient sur l'utilisation des points d'intérêts qui caractérisent un coin dans une image ou d'un objet soit 2D ou 3D. Et afin de mettre en correspondance un ensemble de points d'une image à une autre, une description locale est utilisée qui permet d'extraire l'information du voisinage de chaque point (valeurs des pixels, des intensités lumineuses, des gradients). [32]. Cette description et extraction des points d'intérêt a connu une longue tradition au sein de traitement et d'analyse d'images.

Dans le domaine 2D, l'extraction d'indices dans une image est une étape indispensable en vision par ordinateur, c'est pour cela la détection des points d'intérêt directement à partir du signal semble la technique la plus puissante. D'autre part, la détection des points d'intérêt s'avère fiable quelles que soient les images traitées, car ils donnent des résultats comparables sur des images très texturées comme sur des images contenant moins de textures. Plusieurs travaux [42] montre l'utilité des points d'intérêt parce qu'ils sont locaux ce qui montre leur robustesse en présence de transformations d'images comme l'occultation, la rotation, le changement de point de vue... etc. [33].

Dans le domaine de la transformation du modèle 3D, la détection de points d'intérêt est essentielle pour de nombreuses applications, telles que l'enregistrement, simplification de maillages, segmentation de maillages, la sélection de point de vue d'objet correspondant et la récupération.

L'utilisation de points d'intérêt pour correspondre à des formes en 3D a l'avantage de fournir des caractéristiques locales qui sont sémantiquement significative et aussi invariant à la rotation, mise à l'échelle, le bruit, la déformation, et l'articulation.



Ces points d'intérêt qui sont également appropriée pour la gamme 3D de la reconnaissance des images et des objets sont aussi appelés les points caractéristiques, les points saillants ou des points clefs, ce sont les points qui se distinguent dans leur localité, et qui sont présents et stable à toutes les instances d'un objet, ou de sa catégorie d'objets.[63]

Dans ce chapitre, nous sommes particulièrement concernés par l'analyse des points d'intérêt sur les objets 3D, et on va concentrer cette analyse sur des algorithmes de détection de points d'intérêt automatique 2D et 3D.

Pour cet objectif, plusieurs combinaisons de détecteurs et descripteurs de points d'intérêt 2D et 3D sont testés et les expériences sont basées sur des objets 3D avec des différentes propriétés, et les résultats obtenus indiquent que la méthode spécifique pour extraire et décrire les points d'intérêt dans les données 3D doit être choisie avec soin parce que dans de nombreux cas l'exactitude souffre à partir d'une trop forte réduction des points disponibles.

Comme on a dit, cette partie est inclut dans le domaine de la reconnaissance d'objets 3D en utilisant un échantillon d'images 2D, et en particulier à la recherche d'un nouveau détecteur des points d'intérêt pour extraire ou détecter les formes libres des objets 3D dedans des images 2D par les PIs afin de faciliter leurs taches de reconnaissance et leur recherche dans une base de données. C'est pour cela nous allons développer un nouveau détecteur des points d'intérêt basé sur le détecteur de SUSAN [39] tout en donnant une telle définition sur ce dernier et on verra les manipulations faites dans ce qui suit en faisant une telle comparaison sur la détection des objets 3D en vue 2D qui sont déjà expérimentés.

## **4.2 Notions de base sur les points d'intérêt (PIs)**

---

Les descriptions globales s'avèrent inadéquates quand il s'agit de rechercher un objet d'une image requête dans une base d'images ou de rechercher juste une partie de l'image requête. De ce fait, l'utilisation de descriptions locales constitue une étape clé de la recherche d'objet dans les bases d'images. Pour réaliser cette description il faut extraire des primitives de l'image appelées points d'intérêt découvrant les zones qui présente un contenu informatif important.

#### 4.2.1 Définition et différents types de points d'intérêt

La notion de points d'intérêt a été introduite pour la première fois par Moravec [35]. Pour lui, les points d'intérêts correspondent à un changement bidimensionnel du signal comme par exemple les coins, les jonctions en T, en Y, en L...etc. Toutefois, un point d'intérêt est plus général qu'un coin, pourtant ils sont utilisés dans la littérature comme équivalents. [134]

La détection de points d'intérêt, toute comme la détection de contours, est une étape préliminaire à de nombreux processus dans le domaine de la vision par ordinateur. Les points ainsi extraits peuvent être utilisés dans les procédés de reconstruction 3D, de suivi d'objets, de reconnaissance des objets soient 2D ou 3D, de détection de personnes ou encore de reconnaissance de gestes.

Les points d'intérêt correspondent généralement à une discontinuité des niveaux de gris comme le montre la figure ci-dessous Ils peuvent également apparaître lors d'une modification de la structure, la déformation d'un objet, une courbature, de la texture ou de la géométrie de l'image. [33]

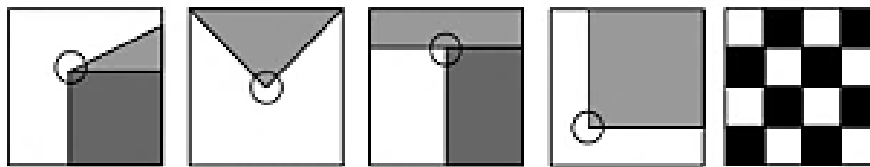


Figure 4-1: Différents types de points d'intérêt : coin simple, jonction en 'V', jonction en 'T', jonction en 'L', jonction en 'damier'[33]

#### 4.2.2 Détection des Coins

La détection des coins joue un rôle important dans la vision par ordinateur, ainsi que dans la reconstruction 3D de la scène, c'est pour cela l'efficacité des points d'intérêt a été prouvée pour la reconnaissance d'objets dans les deux domaines 2D et 3D.

A l'aide des coins, nous pouvons déterminer les points les plus caractéristiques d'un objet ainsi de le reconstruire et aussi ils sont utiles en cas de reconnaissance de formes d'objets. Dans cet article,

on va introduire un nouvel algorithme de détection des coins qui est fondée sur la base de l'algorithme de SUSAN.

### 4.3 Détecteur SUSAN de Smith

---

Après avoir d'abord utilisé le célèbre algorithme de détection des contours Canny [135] pour trouver les contours, nous avons découvert un nouvel outil de détection des contours, SUSAN [39]. Le principal avantage de cet algorithme est que les bords qu'il produit sont plus solides et mieux connectés, c'est-à-dire qu'il manque moins de pièces dans les bords. Cela facilite l'extraction de formes entières des objets. De plus, grâce à une implémentation relativement simple, le passage de Canny [135] à SUSAN [39] s'est fait facilement. En effet, SUSAN est un algorithme plus rapide que Canny, ce qui est important dans la création de cette étape, qui est la plus coûteuse du système.

La différence d'approche entre Canny et SUSAN est que SUSAN ne se rapproche pas des dérivées dans l'espace pixel de l'image, une technique utilisée depuis longtemps pour trouver des bords dans une image. Il détermine plutôt par pixel s'il se trouve sur un bord, en le comparant aux pixels d'une zone environnante. Nous pouvons voir la version abstraite de l'algorithme SUSAN est présentée à la figure ci-dessous.

Nous n'utiliserons cependant pas l'algorithme SUSAN complet, mais seulement les trois premières étapes pour obtenir les pixels de bord, ceci sera suivi d'une approche plus spécifique visant à extraire les formes de l'image. En se rappelant maintenant du fonctionnement des étapes utilisées de l'algorithme qui sont déjà détaillés dans la partie de l'état de l'art.

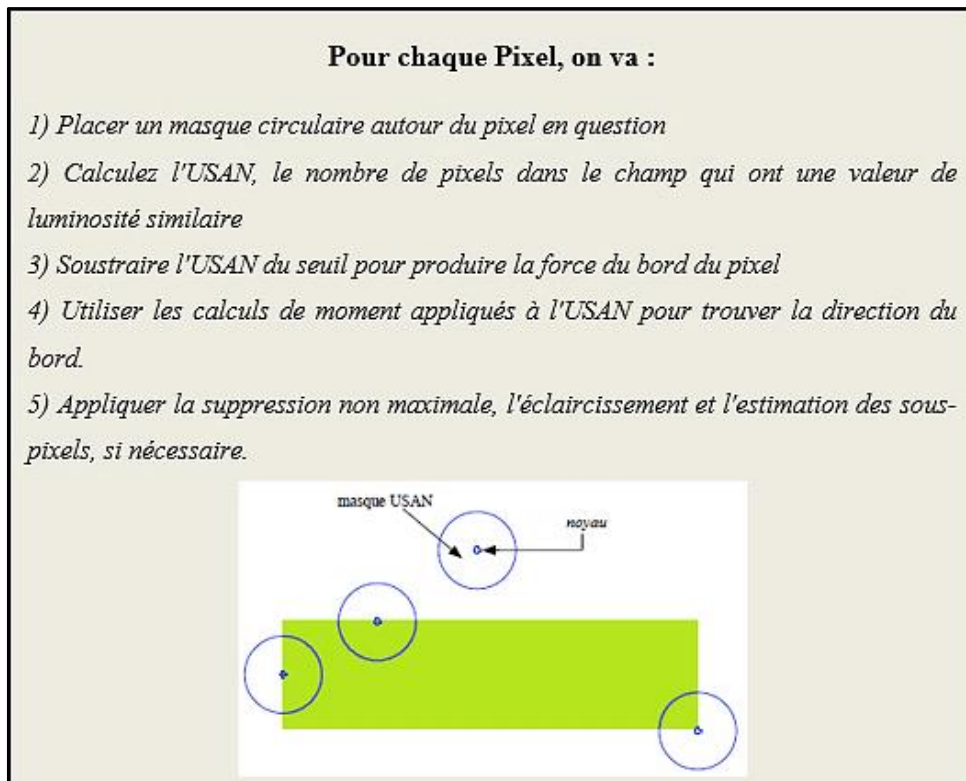


Figure 4-2: Etapes qui constituent l'algorithme de détection de contour SUSAN

### 4.3.1 Calcul de l'USAN

USAN signifie "Univalue Segment Assimilating Nucleus". Il s'agit d'une zone partielle à l'intérieur d'une plus grande zone circulaire de pixels qui ont approximativement la même valeur que le nucléide du pixel central de la zone, du fait, plus le nombre de pixels est similaire au noyau, plus la zone USAN est grande, atteignant un maximum lorsque tous les pixels de la zone ont la même valeur. Dans l'algorithme SUSAN, l'USAN est calculé pour chaque pixel en faisant le noyau (ou centre) d'un masque circulaire qui passe sur l'image. On peut facilement constater que plus le USAN devient petit, plus la probabilité qu'il se trouve sur un bord devient grande. Dans la figure 4-2, on montre des exemples de masques circulaires sont répartis sur un dessin d'un rectangle, accompagné de la figure 4-3 où la zone USAN de ces masques circulaires est représentée en blanc.

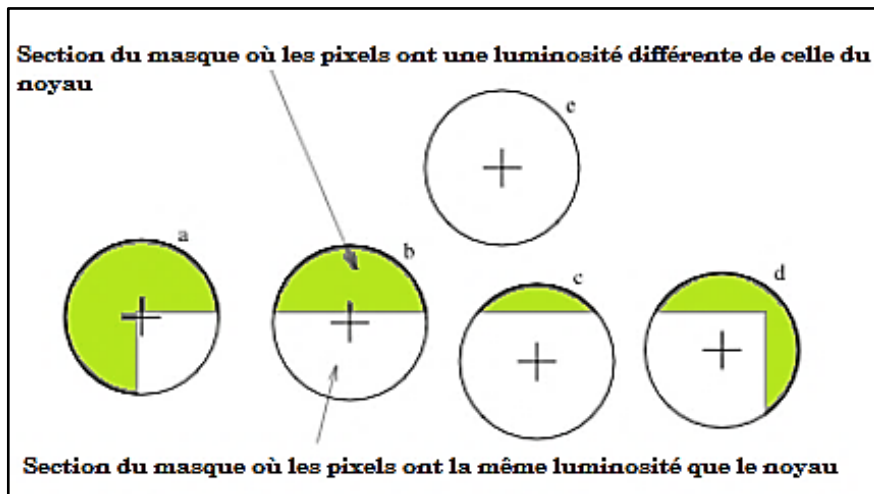


Figure 4-3: Zone USAN des masques circulaires de la figure 4-2 est de couleur blanche.

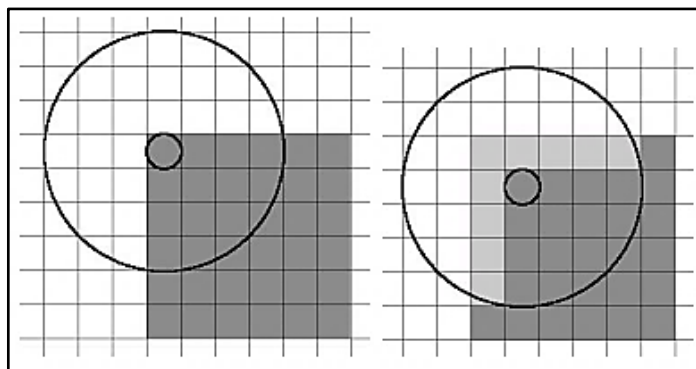


Figure 4-4: Différence entre un noyau sur un bord et pas sur un bord

Pour avoir une idée encore meilleure de la façon dont la région d'USAN change lorsqu'elle rencontre un bord, nous avons inclus la figure 4-4 qui montre clairement la différence entre la zone USAN qui est sur un bord et la zone qui est juste au large du même bord. La propriété d'un cercle étant la plus large au milieu est ici intelligemment exploitée pour créer cet effet. Tout déplacement autour du centre, la place du noyau, a un effet beaucoup plus important que sur les bords extérieurs du cercle.

La comparaison d'un noyau à d'autres pixels peut se faire en utilisant un seuil absolu pour la différence permise. Cependant, il a été démontré qu'il est plus performant à l'aide d'une équation qui permet une comparaison plus aisée

[Chapitre 3 : Représentation des formes libres des objets 2D/3D en vue 2D par l'extraction des P.Is]

$$C(r, r_0) = e^{-\left[\frac{|B(r)-B(r_0)|}{t_{sim}}\right]^6} \quad [136] \quad (4-1)$$

Ici  $B(r)$  est la luminosité du pixel,  $t_{sim}$  est le seuil de similarité et  $r_0$  est le noyau.

À l'aide de cette mesure de comparaison, la valeur de l'USAN entier peut être calculée en effectuant la comparaison dans toute la zone, en additionnant toutes les comparaisons :

$$n(r_0) = \sum_r C(r, r_0) \quad [136] \quad (4-2)$$

### 4.3.2 Comment trouver les pixels de bordure ?

Après avoir appliqué l'étape précédente pour chaque pixel, un paysage 3D de valeurs USAN s'est formé avec une grande quantité de minima locaux comme dans la figure 4-5 (voir l'échelle inversée). Dans l'étape suivante de l'algorithme SUSAN, ces minima locaux sont filtrés en les comparant à un seuil.

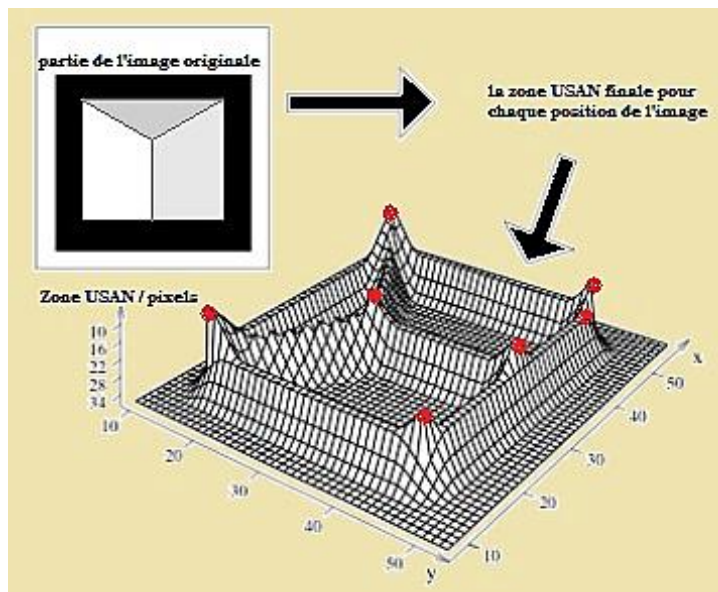


Figure 4-5: Carte des bords résultante avec l'image d'entrée correspondante [136]

En général, le seuil n'est pas une constante fixe, mais une variable basée sur les données recueillies à l'étape précédente. Le seuil est fixé au 3/4 du USAN maximum trouvé pour un coin et

le seuil est fixé au  $\frac{1}{2}$  pour un bord. Ceci conduit à un seuil plus bas dans les images qui montrent plus de variations, contrairement par exemple aux images en noir et blanc simples, où de grandes parties de l'image ont la même valeur de luminosité. La réponse du bord est maintenant formellement approchée comme dans l'équation (4-3) ;

$$R(r_0) = \begin{cases} t_{circ} - n(r_0), & \text{si } n(r_0) < t_{circ} \\ 0, & \text{sinon} \end{cases} \quad [136] \quad (4-3)$$

Ici  $t_{circ}$  est le seuil qui décide de la taille du USAN pour être considéré comme un pixel de bord. Puisque dans notre approche aucune distinction n'est faite entre les pixels de bord de valeurs  $R(r)$  différentes, nous réduirons tous les pixels en pixels de bord (ayant un  $R(r)$  supérieur à zéro) ou non (ces pixels avec une valeur nulle pour  $R(r)$ ).

## 4.4 Nouvelle approche développée

---

### 4.4.1 Choix de la nouvelle approche du détecteur SUSAN ?

Les approches 3D locales avec les points d'intérêt s'inspire des techniques 2D pour définir un détecteur ou un descripteur 3D afin de faciliter la reconnaissance des objets 3D/3D ou 2D/3D.

Rappelons qu'une région d'intérêt, est une partie de l'image qui représente une propriété intéressante. Le point est un cas particulier des régions d'intérêt. Héritant du caractère local, les points d'intérêts permettent essentiellement de remédier aux occultations de l'image. Parti du fait que dans l'image plusieurs points ont des caractéristiques plus significatives que d'autres, et c'était l'idée de H. Moravec [35] qui consiste à utiliser la fonction d'autocorrélation afin de déterminer la meilleure position du point saillant, de façon à ce que toute position voisine contienne moins d'informations.

La problématique essentielle en détection est de trouver des points d'intérêts d'une façon automatisée et suffisamment robustes pour l'appariement et qui suit une forme libre d'un objet pour le détecter facilement. Pour ce faire, un critère primordial à respecter qui est la répétitivité des points sous plusieurs conditions de prise de vue en notant bien que la dimension du descripteur est importante d'autant son influence sur le temps de calcul.

Dans ce qui suit, nous allons voir la nouvelle approche d'un détecteur s'inspirant du détecteur SUSAN, utilisé pour le cas 2D/3D qui va nous faciliter l'extraction des formes libres des objets 3D soit en vue 2D en se basant sur l'algorithme de SUSAN 2D.

#### 4.4.1.1 Méthodologie de la nouvelle approche

Cette approche n'utilise pas les dérivées des intensités de l'image mais elle mesure dans une fenêtre 2D le nombre de pixels ayant une intensité voisine de celle du pixel central de l'objet 3D, et voir qu'une forte variation des intensités signifie l'existence d'un point d'intérêt. Ce détecteur permet de visualiser les coins et les arêtes des objets 3D en faisant une détection locale des objets grâce à l'application d'un nouveau masque circulaire pareil au masque circulaire de SUSAN, mais la différence entre ces deux masques c'est qu'on fait une soustraction d'un élément 1 de chaque pixel des éléments détectés, pour que les points d'intérêt détectent seulement les objets trouvés dans le terrain. Ensuite on fait une comparaison de tous les pixels à l'intérieur du masque avec le NDG du nucleus à l'aide de l'équation de comparaison C utilisée dans l'algorithme basique de SUSAN, puis on calcule le nombre de pixels dedans le nouveau masque qui ont le même NDG que le nucleus et plus précisément on définit la nouvelle zone USAN des objets testés faisant une soustraction de sa taille avec un seuil géométrique choisi qui correspond au minimum de contraste de détermination des contours.

$$C(r, r_0) = e^{-\left[\frac{B(r)-B(r_0)}{t}\right]^6} \quad [136] \quad (4-4)$$

Finalement, on termine l'opération de la détection par la suppression des fausses réponses positives en utilisant le centre de gravité du USAN suivi d'une suppression des non-maximums pour trouver les points d'intérêt qui détecte la forme libre des objets 3D testés.

Ce nouveau détecteur ne tient pas compte de la forme de la structure de l'image locale autour d'un point bien localisé tant qu'il s'intéresse aux objets 3D à détecter. Au contraire, il analyse plusieurs régions séparément en utilisant des mesures locales directes en déterminant ainsi les frontières de régions individuelles qui ont de fortes courbures.



#### 4.4.1.2 Nouveau détecteur robuste au bruit

Les points d'intérêt sont définis comme étant des points qui possèdent des caractéristiques qui permettent de les distinguer des autres points de l'image ou de l'objet, comme par exemple un fort contraste. Historiquement, on a cherché à extraire les coins des objets dans une image, les détecteurs ne trouvaient pas seulement les coins, mais également d'autres points dans des zones texturées ou en présence de bruit. Par l'application du générateur du masque et malgré le bruit détecté sur les objets soit un bruit gaussien ou poivre et sel, l'approche a bien détecté les objets 3D en vue 2D tant que l'algorithme original de SUSAN, détecte la plupart des points du bruit de l'image ou des objets malgré sa robustesse contre le bruit. Mais en tous cas, le fait que ce détecteur ne nécessite pas de dérivés, fait que les performances en présence de bruit sont bonnes. L'effet intégrateur ainsi que la réponse non-linéaire du principe de SUSAN permettent une forte rejection du bruit.

#### 4.4.1.3 Organigramme de l'approche proposée

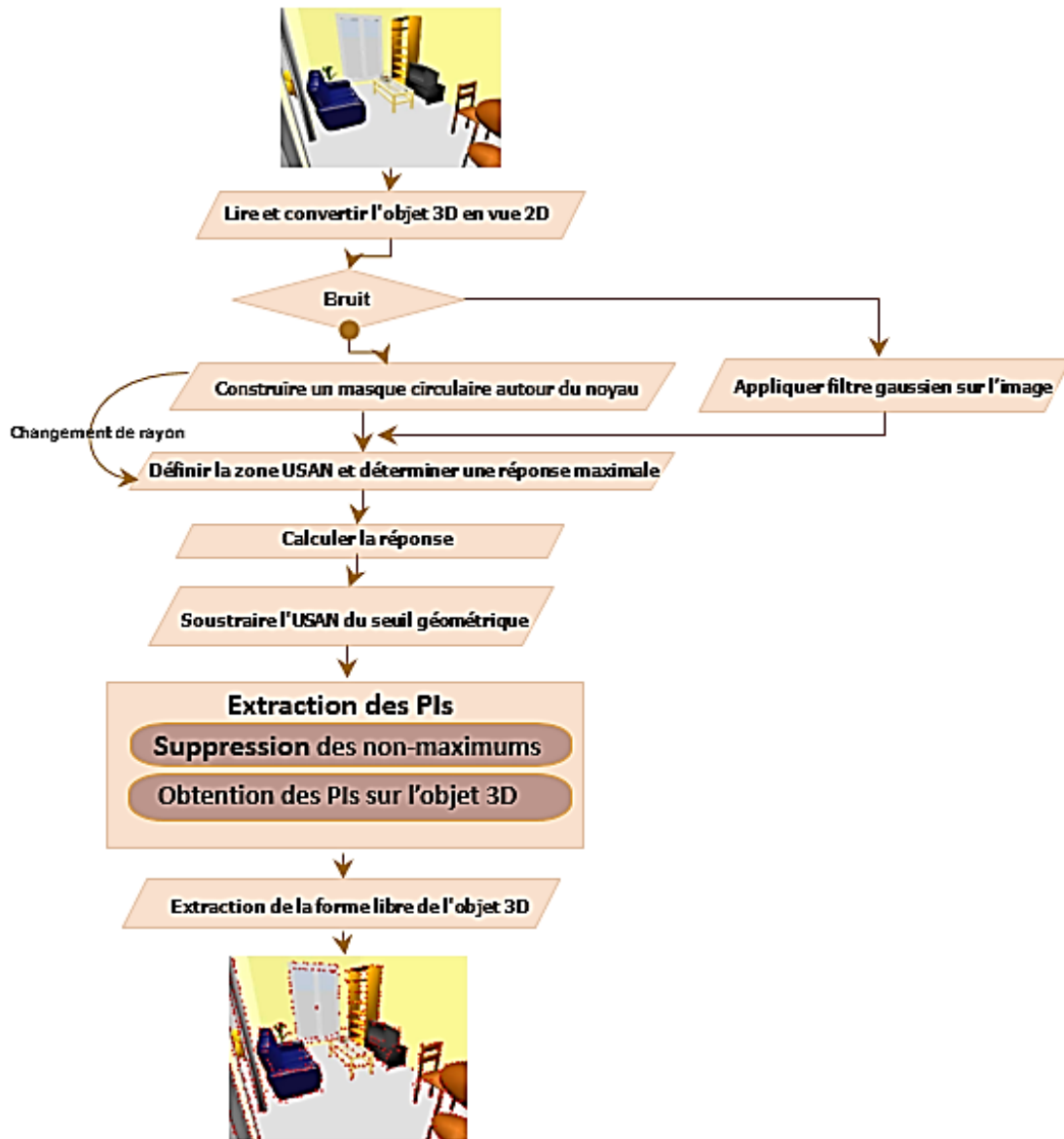


Figure 4-6: Organigramme du nouveau détecteur de SUSAN (NDS)

#### 4.4.1.4 Modèle 2D/3D Utilisé


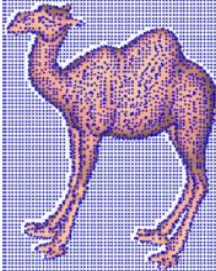






Certains modèles sont des modèles standards qui sont largement utilisés dans la recherche de la forme 3D, comme Human\_Hand, Smooth, Armadillo, la tête de David, Teapot, Horse, etc....ceux sont les objets de la base de données de Benchmark. Mais pour les modèles 2D/3D qu'on a testé dans nos expériences nous avons les créés à partir du logiciel "Sweet Home 3D". [137] et ce logiciel

qu'on a exploité dans notre travail contient un ensemble de données 3D qu'on peut les télécharger depuis notre site de référence.

## 4.5 Résultats et Discussion

Comme nous avons mentionné dans les deux derniers paragraphes, nous avons recueilli des données via notre application, et appliquer l'ancien et le nouveau détecteur de SUSAN comme nous avons décrit avant. Nous avons effectué l'analyse sur deux ensembles de données, la base de données des objets 3D de benchmark et une petite base de données des objets créés d'après le logiciel "Sweet Home 3D".

### 4.5.1 Objets 3D de la base de Benchmark

3D Object	SUSAN Detector	The new Detector SUSAN (NDS)	
	Radius=2 ; alpha=50 ; r =2	Radius=2 ; alpha=50 ; r =2	Radius =20 ; alpha=50 ; r =2
			
<b>Camel.off</b>	Interest points number = 5120 Elapsed time is 2.41 seconds.	Interest points number = 779 Elapsed time is 1.46 seconds.	Interest points number = 2487 Elapsed time is 2.29 seconds.
			
<b>Horse.off</b>	Interest points number =2622 Elapsed time is 1.30 seconds.	Interest points number = 386 Elapsed time is 1.20 seconds.	Interest points number =1110 Elapsed time is 1.28 seconds.

[Chapitre 3 : Représentation des formes libres des objets 2D/3D en vue 2D par l'extraction des P.Is]

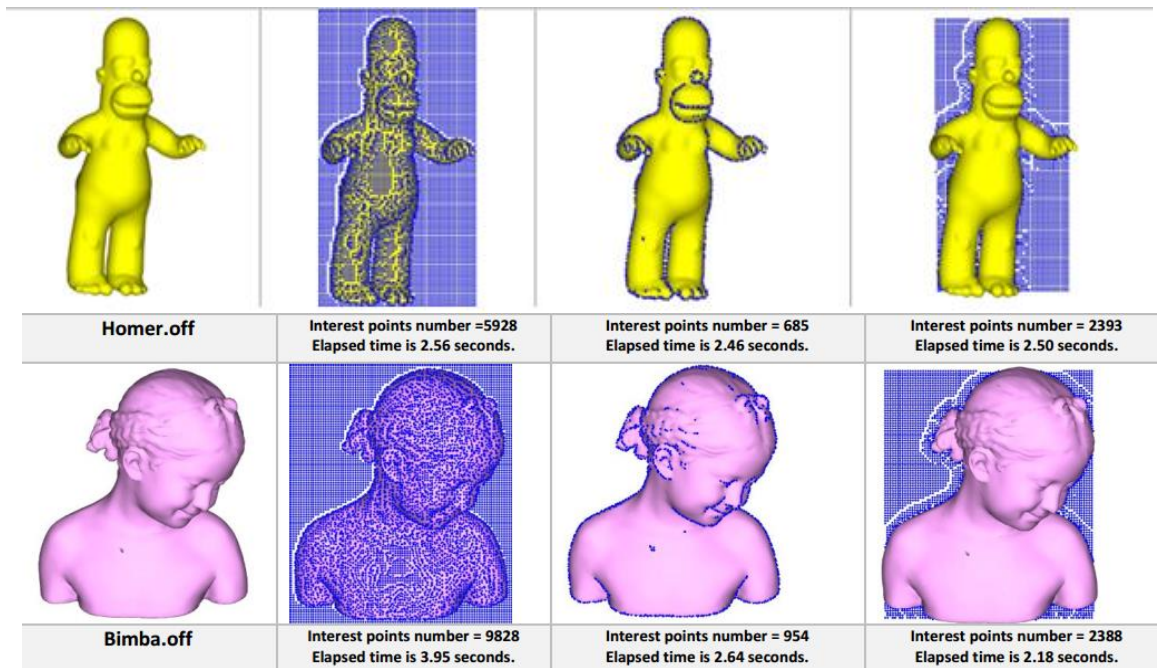


Figure 4-7: Différence entre la détection d'objet 3D par l'ancien détecteur de SUSAN (DS) et le nouveau détecteur SUSAN (NDS).

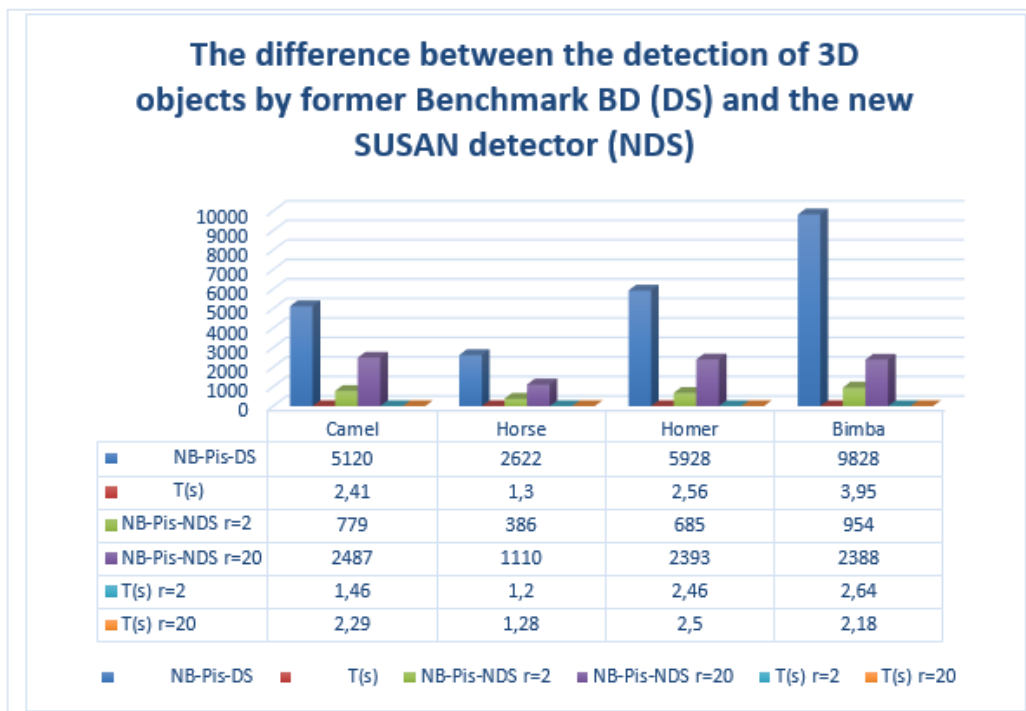


Figure 4-8: Différence entre la détection d'objet 3D-BD de Benchmark- par l'ancien détecteur de SUSAN (DS) et le nouveau détecteur SUSAN (NDS).

➤ Voir que dans le masque utilisé, on a traité trois cas :



[Chapitre 3 : Représentation des formes libres des objets 2D/3D en vue 2D par l'extraction des P.Is]

- Le cas du détecteur SUSAN → il y a une détection totale des points d'intérêt sur l'espace et l'objet 3D.
- Le cas du nouveau détecteur SUSAN → on a vu deux cas :
  - Le cas du rayon du masque =2 → il y a une détection des points d'intérêt autour de l'objet 3D testé pour préciser la détection de sa forme libre.
  - Le cas du rayon du masque =20 → il y a une détection des points d'intérêt dans l'espace derrière l'objet 3D testé ce qui nous donne une apparence de l'objet 3D détecté.

4.5.2 Objets 3D du logiciel « Sweet Home 3D »








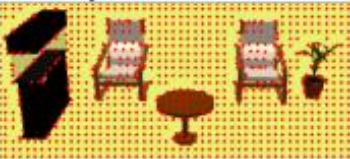


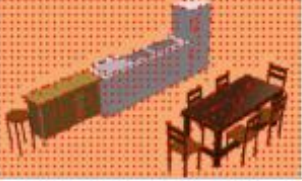

3D Object	SUSAN Detector (DS) Radius=2 ; alpha=80 ; r=5	The new Detector SUSAN (NDS) Radius =2 ; alpha=80 ; r=5
 Room 1	 Interest points number = 1691 Elapsed time is 4.18 seconds.	 Interest points number = 680 Elapsed time is 3.35 seconds.
 Room 2	 Interest points number = 1428 Elapsed time is 3.63 seconds.	 Interest points number = 426 Elapsed time is 2.83 seconds.
 Room 3	 Interest points number = 1029 Elapsed time is 2.92 seconds.	 Interest points number = 347 Elapsed time is 2.43 seconds.
 Room 4	 Interest points number = 1118 Elapsed time is 4.82 seconds.	 Interest points number = 393 Elapsed time is 4.51 seconds.

Figure 4-9: Différence entre la détection des objets 3D - BD de Sweet Home 3D- par l'ancien (DS) et le nouveau détecteur de SUSAN(NDS). « Sweet Home 3D »

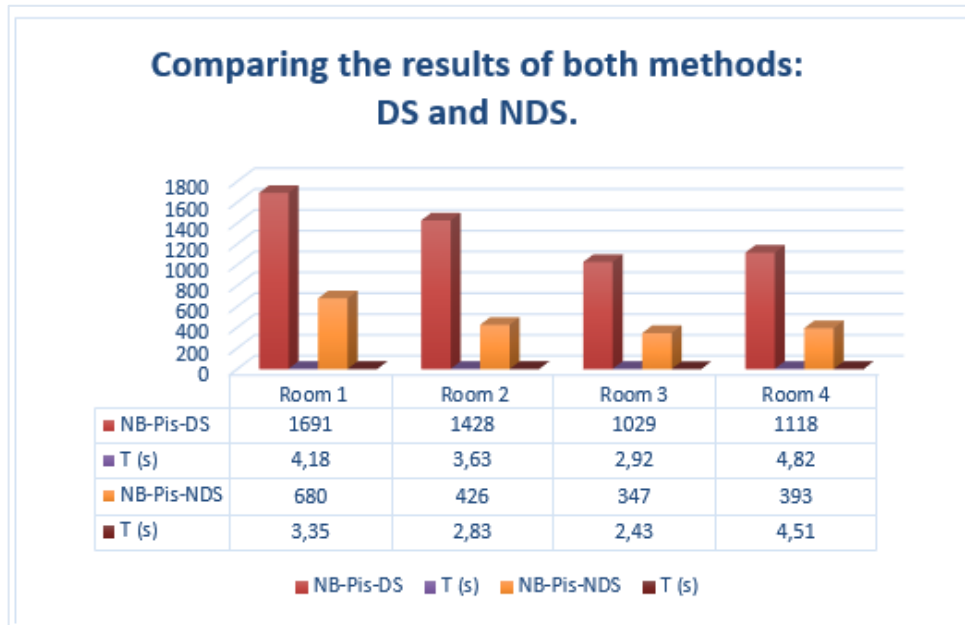


Figure 4-10: Différence entre la détection des objets 3D par Le détecteur SUSAN (DS) et le nouveau détecteur SUSAN (NDS)  
« Sweet Home 3D »

- On a ici deux cas à traiter le premier de SUSAN qui détecte les objets 3D et leur espace et le deuxième algorithme qui nous montre une extraction de la forme libre des objets 3D testés.
- Par le nouveau détecteur NDS, on obtient toujours moins de points d'intérêt et dans un temps de calcul réduit par contre au détecteur SUSAN, plus que le nombre de points d'intérêt augmente, plus que le temps de calcul agrandit, et les points d'intérêt qui détectent la forme libre de ces objets sont bien localisés par rapport aux autres du détecteur SUSAN.

4.5.3 Objets 3D Bruités en vue 2D



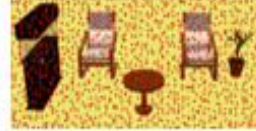










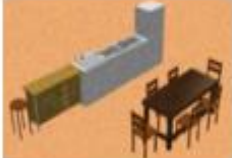
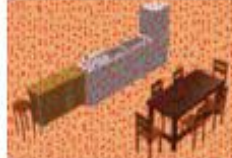
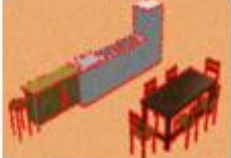
<i>Chambre 1</i>	<i>Chambre bruitée 1</i>	<i>Détecteur SUSAN (DS)</i>	<i>Le nouveau détecteur de SUSAN (NDS)</i>
			
		Interest points number = 1078 Elapsed time is 3.20 seconds.	Interest points number = 906 Elapsed time is 3.13 seconds.
3D Object : <del>Maxplanck.off</del>	<del>Maxplanck</del> bruité	<i>Détecteur SUSAN (DS)</i>	<i>The new SUSAN Detector (NDS)</i>
			
		Interest points number = 468 Elapsed time is 1.15 seconds.	Interest points number = 288 Elapsed time is 1.10 seconds.
3D Object : <del>Hand.off</del>	<del>Main</del> bruitée	<i>Détecteur SUSAN (DS)</i>	<i>Le nouveau détecteur de SUSAN (NDS)</i>
			
		Interest points number = 525 Elapsed time is 3.16 seconds.	Interest points number = 324 Elapsed time is 1.60 seconds.
<i>Chambre 2</i>	<i>Chambre bruitée 2</i>	<i>Détecteur SUSAN (DS)</i>	<i>Le nouveau détecteur de SUSAN (NDS)</i>
			
		Interest points number = 1092 Elapsed time is 2.18 seconds.	Interest points number = 917 Elapsed time is 1.73 seconds.

Figure 4-11: Différence entre la détection des objets 3D bruités par l'ancien (DS) et le nouveau détecteur de SUSAN(NDS)

- Ici, nous avons deux cas : Le premier est le détecteur SUSAN qui détecte l'objet 3D et son espace et le second algorithme fait l'extraction de la forme libre des objets 3D testés dedans des images bruitées.
- Grâce au nouveau détecteur SUSAN, nous obtenons moins de points d'intérêt et rapidement. Ces points détectent la forme libre des objets 3D et ils sont bien situés par rapport à un point d'intérêt d'un autre détecteur qui est répandu.

[Chapitre 3 : Représentation des formes libres des objets 2D/3D en vue 2D par l'extraction des P.Is]

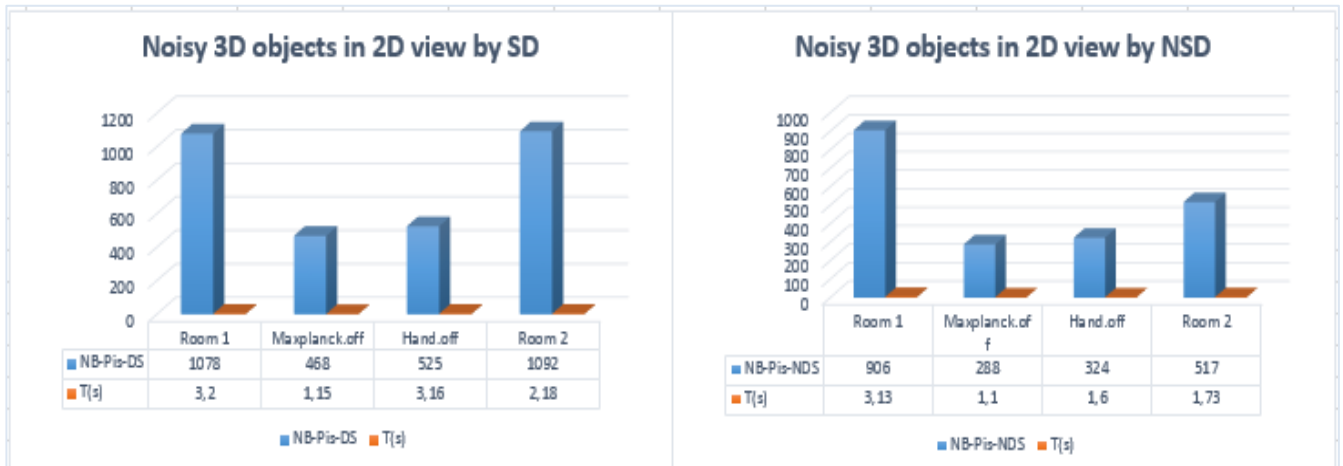


Figure 4-12: Différence entre la détection d'objets 3D bruités par (DS) et le nouveau détecteur SUSAN (NDS)

➤ Ici on ajoute du bruit aux objets 3D, on les teste par les deux détecteurs de SUSAN, puis on constate qu'on obtient deux types de résultats :

- Pour le détecteur de SUSAN, on obtient une propagation aléatoire de plusieurs de points d'intérêt et pas de détection de l'objet 3D ce qui prend toujours plus de temps d'exécution par rapport au nouveau détecteur.
- Pour le nouveau détecteur de SUSAN, on obtient moins de points d'intérêt dans un peu de temps par rapport au premier détecteur, plus la détection de la forme libre des objets 3D est bonne et les points d'intérêt sont bien localisés par rapport aux points d'intérêt du premier détecteur de SUSAN. Ce qui nous montre la vraie robustesse du nouveau détecteur contre le bruit.

➔ En reconnaissance des objets, un choix exact des points d'intérêt permet de minimiser le temps de calcul et éviter l'ambiguïté dans le calcul des descripteurs et de la mise en correspondance et voir la vraie résistance et robustesse des détecteurs contre le bruit.



## 4.6 Interprétation et évaluation

### 4.6.1 Interprétation et complexité

Selon une étude comparative des résultats obtenus dans les figures 4-7 et 4-9 et le tableau 4-1 ci-dessus, on constate que la détection d'objet par le nouveau détecteur SUSAN est plus rapide par rapport à l'ancien détecteur, ou bien on met le même rayon du masque est là l'échange, toujours on a des résultats rapides par rapport aux autres.

Et c'est la même chose pour les objets bruités dans le dernier tableau de résultats, une détection parfaite et claire est visible sur le nouveau détecteur SUSAN par rapport à l'ancien détecteur, ce qui nous montre l'efficacité du nouveau détecteur pour les objets bruités ou sans bruit nécessitant une durée de calcul considérable. De plus, nous voyons une détection de plusieurs points d'intérêt par détecteur SUSAN qui sont dispersés partout sans spécifier la forme libre de l'objet, ce qui va valoriser ce nouveau détecteur.

En effet, certaines caractéristiques qui facilitent l'extraction et la détection de formes libres d'objets 3D dans les images 2D sont incluses dans les résultats, ce qui facilite la reconnaissance de ces objets étudiés. Selon les résultats ci-dessus, nous allons calculer la moyenne de temps d'exécution de chaque tableau des résultats et pour les deux algorithmes afin d'étudier leur efficacité et performance au niveau de temps d'exécution tout en comparant les deux détecteurs SUSAN par rapport à la taille de l'image (Longueur x Largeur px) entrée en mégapixels ((Longueur\*Largeur)/10<sup>6</sup> Mpx) (Tableau pour savoir la relation proportionnelle Taille/Temps ou n/temps où n c'est la taille de donnée, ce qui va nous amène à vérifier la complexité des deux détecteurs.

Tableau 4-1: Comparaison de temps d'exécution entre DS et NDS en fonction de Taille d'image testée

	Taille de l'image testée en (Mégapixels)	Temps(s)-NDS	Temps(s)-DS
Resultats1	0,27	1,94	2,55
Resultats2	0,2	3,28	3,88
Resultats3	0,05	1,89	2,42

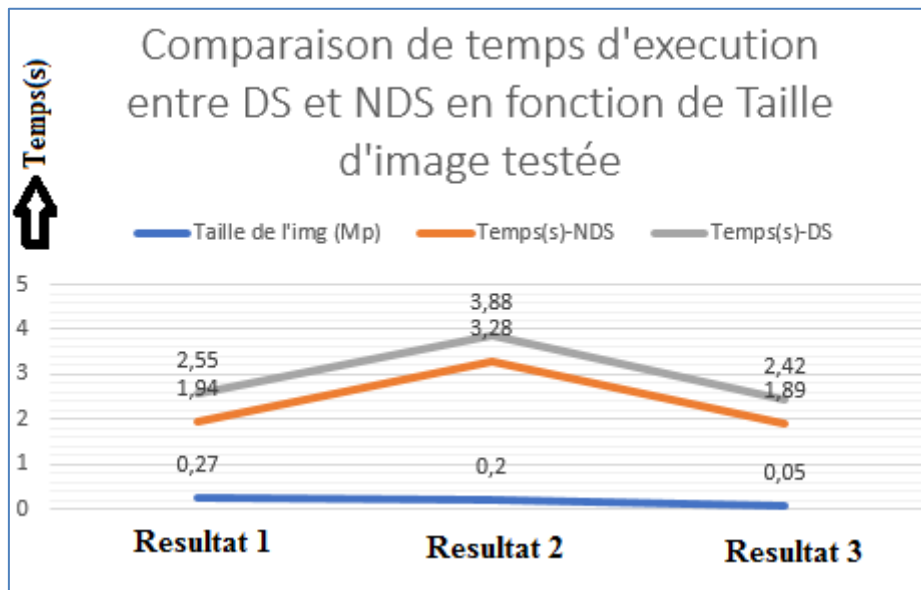


Figure 4-13: Comparaison de temps d'execution entre DS et NDS en fonction de Taille d'image testée

Selon la figure 4-12 , nous remarquons que le temps d'execution du détecteur SUSAN est élevé par rapport au temps d'execution du nouveau détecteur SUSAN malgré qu'on a entré des images de la même taille ,soient bruitées ou non, donc nous pouvons dire que le nouveau détecteur SUSAN ne gaspille pas plus de temps d'implémentation par rapport à l'autre et sa détection des points d'intérêt est bien localisé sur les bords et les coins de chaque objets 3D dedans une image 2D,en fait ,on peut calculer la complexité des deux détecteurs pour savoir la différence entre les deux au niveau de leur performance .

A priori, on ne peut pas toujours mesurer le temps de calcul sur toutes les entrées possibles d'un algorithme, ce qui nous amène à calculer la complexité temporelle des deux détecteurs et à savoir s'ils sont robustes au bruit ou pas. Après avoir calculer la complexité de notre nouveau détecteur, nous avons trouvé une complexité quadratique  $O(n^2)$  qui est inférieure à la complexité cubique  $O(n^3)$  de l'ancien détecteur SUSAN ce qui montre que notre détecteur est plus rapide que l'autre comparable. L'étude de la complexité peut conduire à une classification des problèmes liés à la performance des algorithmes les plus connus que cette résolution.

## 4.7 Conclusion

---

Dans ce chapitre, nous avons présenté notre contribution concernant la détection des formes libres des objets 3D en vue 2D en utilisant des caractéristiques pour modéliser un objet 3D et sa forme libre et qui sont appelés « les points d'intérêt » et qui sont toujours encore présentes dans les objets de notre vie commune. En plus, l'utilisation de descripteurs/détecteurs globaux et locaux en même temps, et l'approche hiérarchique qui essaie d'abord la reconnaissance du modèle complet avant de reconnaître uniquement des sous-modèles de plus en plus petits, rendent possibles de détecter dans une scène, un objet même s'il est partiellement caché (le cas des occultations).

L'algorithme que nous avons développé pour faciliter la détection de la forme libre d'un objet 3D, devait exploiter à la fois des caractéristiques géométriques et cela afin d'obtenir un système de reconnaissance plus robuste et stable.

Et d'après les résultats qu'on a obtenus soit avec le bruit ou sans, cette nouvelle approche nous donne une bonne performance des résultats obtenus qui sont relativement satisfaisantes, dans la comparaison avec ceux qui sont obtenus par d'autres algorithmes. Et on peut aussi appliquer cette méthode dans les différents contextes de traitement d'images, et dans la vision par ordinateur, pour la détection des contours, ainsi que dans la segmentation de l'image et des objets 3D en vue 2D.

## 5 Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier

---

---

### 5.1 Introduction

---

Dans le but d'améliorer la qualité du traitement et de minimiser le temps de réponse global d'un système de vision et faciliter le problème épineux de la reconnaissance d'objets au niveau de la hiérarchie des tâches de vision, le système de vision doit être capable de combiner ses capacités de représentation interne dans le but de réussir des décisions.

Une grande partie des méthodes de reconnaissance d'objets 3D calcule les descripteurs de formes en intégrant une fonction sur le volume de l'objet. Pour cela, l'objet est dans un premier temps discrétisé sur une grille de voxels et l'intégration est faite dans le domaine discret. Ces types d'objets qui sont complexes, ont une certaine structure. Par exemple, une théière est formée d'une anse, d'un récipient, etc. L'idée est que, pendant la reconnaissance, la détection d'un objet passe par la détection de ses sous-objets appelés caractéristiques clés. L'ensemble de ces caractéristiques clés définissent l'objet. En pratique, cette structure peut être facilement obtenue à partir d'une représentation en fils de fer c'est-à-dire un maillage ; les objets sont alors considérés comme un ensemble de petites faces et les caractéristiques clés en sont des sous-ensembles.

Les maillages sont principalement utilisés pour représenter les modèles 3D car ils s'intègrent bien au matériel de rendu graphique. Des caractéristiques telles que le volume, les moments et les coefficients de transformation de Fourier doivent être calculés efficacement à partir de la représentation du maillage.

Les maillages et, en particulier, les maillages triangulaires sont probablement aujourd'hui les représentations les plus courantes des formes. Plutôt que de spécifier pour chaque point 3D s'il se trouve à l'intérieur ou à l'extérieur de la forme, ou simplement de répertorier un ensemble de points sur la surface, un maillage représente la forme par une approximation de sa surface, qui est souvent modélisée comme une variété bidimensionnelle. Concrètement, dans un espace 3D, un maillage est un ensemble de sommets (points 3D) et de faces (ensembles de sommets coplanaires).

## [Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier]

Les maillages 2D permettent de partitionner un espace plan en un ensemble de triangles, ce qui permet alors de passer d'un espace continu à un espace discret. Ceci pourra être utilisé par exemple en informatique graphique et en conception assistée par ordinateur pour représenter et manipuler des objets tridimensionnels. On peut également utiliser un maillage pour approximer une fonction dont on ne peut pas connaître une forme analytique : c'est notamment le cas pour la résolution de certaines équations différentielles.

C'est pour cela, nous allons établir une représentation interne d'objets liée avec une extraction de sa forme libre et de sa position sous forme d'un maillage 3D. Ce qui facilite la tâche des algorithmes de traitement d'images et essentiellement ceux de l'apprentissage et de reconnaissance en se basant sur l'utilisation des techniques d'alignement ce qui consistent à transformer le repère des modèles en celui des données (image) en se profitant de l'algorithme des surfaces de Bézier pour bien extraire le maillage 3D du modèle donné, ce qui simplifie après les tâches de la reconnaissance, de la recherche et de l'indexation des objets dans une base de données. Dans un premier temps, nous allons présenter un maillage 3D, qu'est-ce que c'est une courbe et surface de Bézier ? tout en donnant des petits détails sur eux, puis nous présenterons notre travail effectué sur les objets 3D en vue 2D, et enfin nous afficherons les résultats des générations de maillages 3D à partir des espaces 2D tout en utilisant les courbes et les surfaces de Bézier pour qu'on puisse passer d'un maillage 2D à un autre 3D.

### **5.2 Notions de Base & Création Du Maillages 3D Adapté à L'objet**

---

#### **5.2.1 Définition d'un maillage en 3D**

Avant de poursuivre plus en avant, il est nécessaire de définir ce qu'est un maillage ! Un maillage est constitué d'un ensemble d'éléments géométriques formés en 3D, de sommets, d'arêtes et de faces, qui partitionne un domaine délimitant les données à analyser. Il est défini par sa métrique et sa topologie. La métrique est l'ensemble des coordonnées des sommets de ses éléments, auxquels sont associées, le cas échéant, une ou plusieurs valeurs physiques. La topologie correspond à la définition de la connectivité entre les différents sommets, ainsi qu'aux relations de voisinage entre les éléments constitutifs du maillage. [138]

## [Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier]

On dit qu'un maillage est conforme lorsque l'ensemble de ses éléments constitutifs recouvre le domaine, et que toute intersection entre deux éléments du maillage est réduite soit à l'ensemble vide, soit à un sommet, soit à une arête, soit à une face, pour un maillage en 3D.

### 5.2.2 Représentation des maillages par les surfaces de Bézier

Afin d'obtenir un bon maillage d'objet, on doit passer par trois étapes : dessin des courbes de Bézier, dessin des surfaces de Bézier, et finalement une subdivision triangulaire de ces surfaces.

#### 5.2.2.1 Courbes de Bézier

➤ **Définition** Une courbe de Bézier est une courbe polynomiale de degré  $n$  définie par  $n+1$  points de contrôle  $(P_0, \dots, P_n)$ , et décrite sur  $t \in [0; 1]$  par :

$$P(t) = \sum_{i=0}^n B_i^n(t) P_i \quad (5-1)$$

Où 
$$B_i^n(t) = C_n^i (1-t)^{n-i} t^i \quad (5-2)$$

avec 
$$C_n^i = \frac{n!}{i!(n-i)!} \quad (5-3)$$

➔ Les  $B_i^n(t)$  sont appelés polynômes de Bernstein. [139]

➔ L'ensemble des points  $(P_0, \dots, P_n)$ , pris dans cet ordre, est appelé polygone de contrôle.

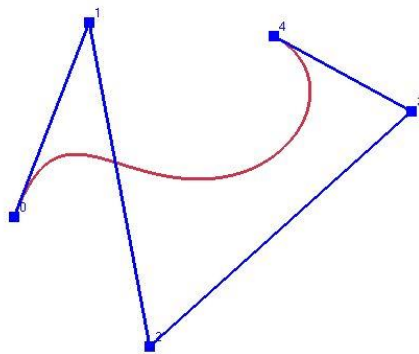


Figure 5-1: Courbe de Bézier avec 5 points de contrôle

[Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier]

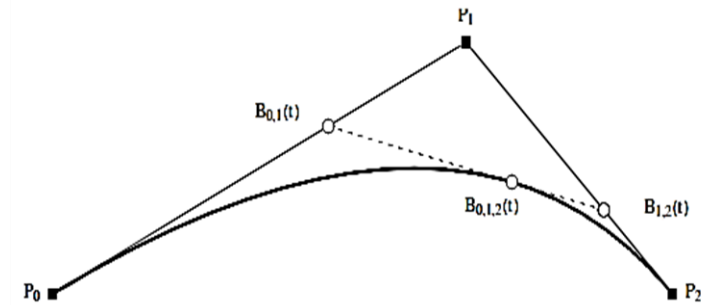


Figure 5-2: Construction d'une courbe Bézier à 3 points pour  $t=2/3$

On notera que cette courbe est définie même si  $t$  n'appartient pas à  $[0, 1]$ , mais elle n'est pas contrôlée à l'extérieur de cet intervalle. Plusieurs exemples de courbes sont présentés dans la figure suivante [140] :

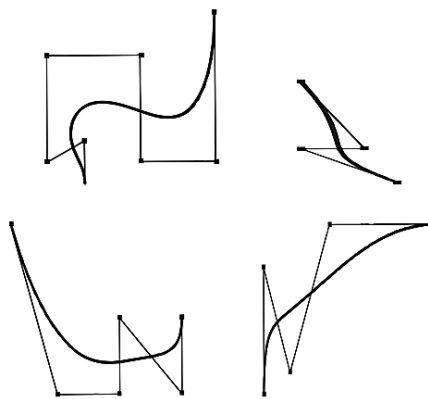


Figure 5-3: Exemples de courbes Bézier

L'algorithme de construction de Casteljau [139] [141] se généralise facilement à un nombre de points quelconques comme le montre l'exemple suivant. Le polygone engendré par l'ensemble des points  $(P_0, \dots, P_n)$  est appelé polygone de Bézier (ou polygone de contrôle).[140]

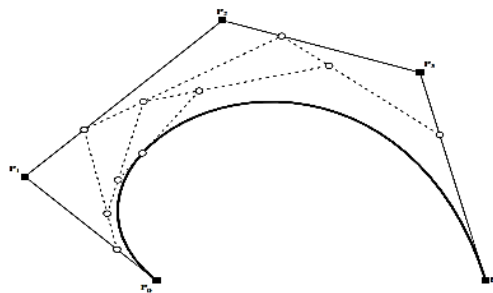


Figure 5-4 : Construction d'une courbe Bézier à 5 points pour  $t=1/3$  (« Construction de Casteljau »)

➤ *Propriétés des courbes de Bezier*

- La courbe est de degré  $n$ .
- Invariantes par les transformations affines c'est-à-dire on a besoin que de transformer les points de contrôle.
- Invariantes par les transformations affines du paramètre de parcours  $t$ .
- La courbe est tangente au polygone de contrôle en  $P_0$  et  $P_n$ .
- On peut rajouter des poids associés à chaque point de contrôle.
- La courbe est exprimée comme une combinaison linéaire des points en chaque  $t$  :  $P(t)$  est barycentre et les coefficients barycentriques des points  $P_i$  sont les  $B_i^n(t)$ .
- Donne des formes naturelles.
- Contrôle aisé et « assez » intuitif (position des points de contrôle).
- Très utilisées en informatique graphique.
- Très nombreuses propriétés (enveloppe convexe, intersection, ...). [139]

5.2.2.2 Surfaces de Bezier

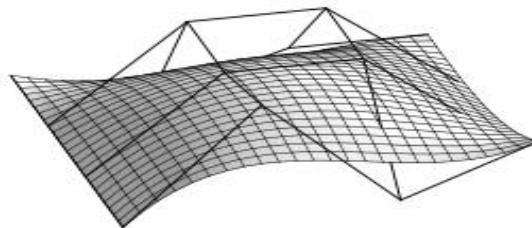


Figure 5-5: Surface de Bezier

➤ *Approche intuitive des surfaces de Bezier :*

Les surfaces de Bezier se construisent à partir de la déformation d'une grille de points. Si cette grille est de taille  $n \times p$  de points, on note ces points  $P$  avec  $i \in \{0, \dots, n-1\}$  et  $j \in \{0, \dots, p-1\}$  On appelle carreau de la grille, le "quadrilatère" Constitué de quatre points adjacents sur la grille :  $\{P_{ij}, P_{i+1,j}, P_{i,j+1}, P_{i+1,j+1}\}$ . [140]

• *Construction par l'algorithme de de Casteljau :*

La figure 5-6 montre l'adaptation de l'algorithme de *Casteljau* [141] présenté pour les courbes de Bezier pour la construction de surfaces de Bezier.

Si on considère un ensemble de points de contrôle constitué de  $n \times p$  points, alors le point de la surface de Bezier associé aux paramètres  $(u, v)$  se construit de la façon suivante. Pour tous les carreaux de la surface, on calcule l'interpolation bilinéaire correspondant au point  $(u, v)$  ; ceci



[Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier]

permet d'obtenir un ensemble de nouveaux points qui définissent à leurs tours un ensemble de carreaux (à la première étape :  $n-1 \times p-1$ ). On recommence alors itérativement le processus jusqu'à ne plus obtenir qu'un seul point. [140]

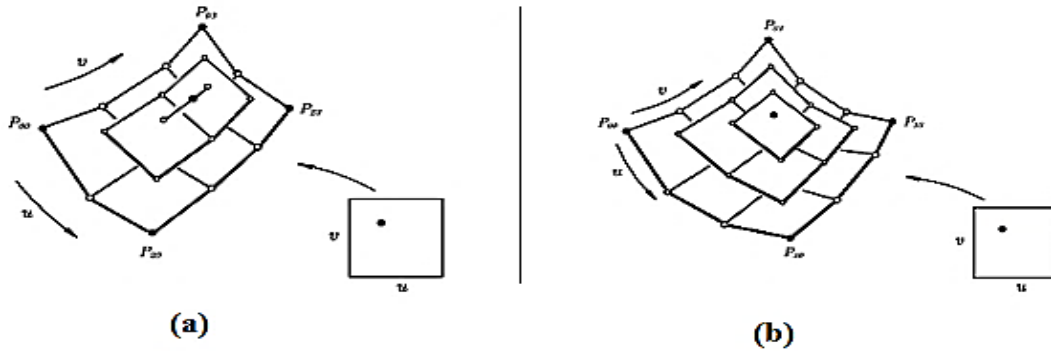


Figure 5-6: Surface de Bézier : construction par l'algorithme de de Casteljau (a) et (b)

• *L'approche « produit tensoriel » :*

On peut définir une surface à partir de deux courbes, comme par exemple un tore à partir d'un cercle et d'une droite. On parle alors de surface fibrée, ou surface produit tensoriel. Cela revient à décorrélérer l'influence de chaque paramètre de la surface.

On peut également utiliser deux familles de courbes, c'est ce qu'on fait avec les surfaces de Bézier.

→ *Formulation*

L'idée est de réutiliser au maximum les courbes de Bézier. Plutôt que de considérer un polygone de contrôle ( $\mathbf{P}_i$ ), on considère un polyèdre de contrôle ( $\mathbf{P}_{ij}$ ) composé de  $m \times n$  points de contrôle agencés dans une grille. [142]

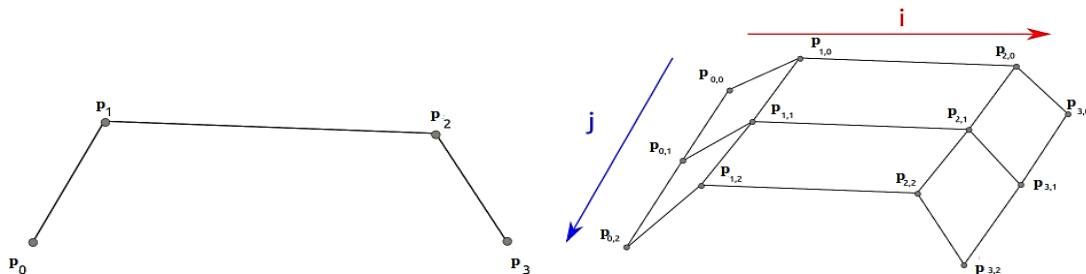


Figure 5-7: A gauche un polygone  $\mathbf{P}_i$  de contrôle et à droite un polyèdre de contrôle  $\mathbf{P}_{ij}$

**[Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier]**

Les surfaces de Bezier sont décrites par le produit tensoriel ci-dessous.

$$B(u, v) = \sum_{i=0}^m \sum_{j=0}^n P_{ij} B_i^m(u) B_j^n(v) \tag{5-4}$$

Où m est le degré en u et n est le degré en v et  $B_i^k$  les polynômes de Bernstein, et  $(u; v) \in [0; 1]^2$ .

**→ Evaluation de surface de Bezier**

On a deux algorithmes géométriques pour évaluer cette surface de Bezier, soit On fixe d'abord j et on fait varier i ou fixer i et faire varier j. et voilà ce qu'on va faire quand on fixe j et varier i :

On a :

$$B(u, v) = \sum_{j=0}^n B_j^n(v) \underbrace{\left[ \sum_{i=0}^m P_{ij} B_i^m(u) \right]}_{=P_j(u)} \tag{5-5}$$

$P_j(u)$  défini une courbe de Bezier en  $u$  que l'on peut évaluer par le Casteljau.

L'équation 5-5 définit une courbe de Bezier en v dont les points de contrôle  $P_j(u)$  dépendent de  $u$ . Au final il y aura [142]:

- { n + 1 évaluation de De Casteljau pour le degré m
- { 1 évaluation de De Casteljau pour le degré n

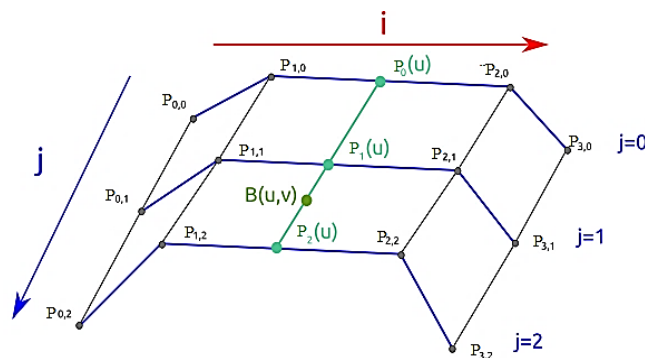


Figure 5-8: Illustration de l'algorithme pour évaluer B(u,v)

[Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier]

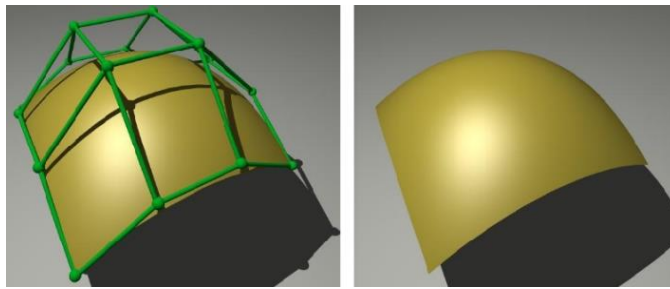


Figure 5-9 : Produit tensoriel de deux courbes de degré 3 (ordre 4)

➤ *Propriétés des surfaces de Bézier :*

- Le degré de la surface dans chaque direction polynomiale est inférieur d'une unité au nombre de sommets polygonaux définis dans cette direction.
- La continuité de la surface dans chaque direction paramétrique est inférieure de deux fois au nombre de sommets de polygones définis.
- La surface suit généralement la forme du filet polygonal défini
- La surface est contenue dans la coque convexe du filet polygonal.
- La surface est invariante sous les transformations affines
- Chacune des courbes limites dans ce cas est une courbe de Bézier [143]

➤ *Utilisation*

Généralement les surfaces de Bézier sont utilisées sous la forme de réseaux de **patch** bicubique (C'est à dire  $m = n = 3$ ) ou biquadratiques (C'est à dire  $m = n = 2$ ).

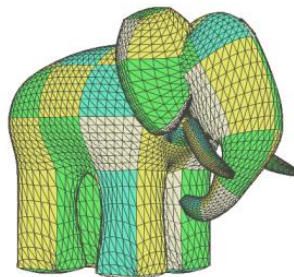


Figure 5-10: "Gumbo" modèle formé de patch bicubiques et créé par Edwin Catmull [144]

**5.2.2.3 Subdivision triangulaire des surfaces de Bézier guide à un maillage fine et lisse :**

Le patch triangulaire de Bézier est défini par les équations suivantes :

[Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier]

$$Surface S = \left\{ P(u,v,w) = \sum_{\substack{i,j,k \geq 0 \\ i+j+k=n}} P_{i,j,k} B_{i,j,k}^n(u,v,w), u,v,w \geq 0, u+v+w=1 \right\} \quad (5-6)$$

fonction de base :  $B_{i,j,k}^n(u,v,w) = \frac{n!}{i!j!k!} u^i v^j w^k$

Po ints de contrôle :  $\{P_{i,j,k}\}$

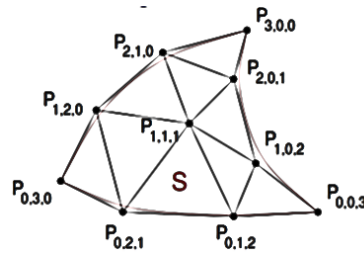


Figure 5-11: Polyèdre  $\{P_{i,j,k}\}$  et patch correspondant

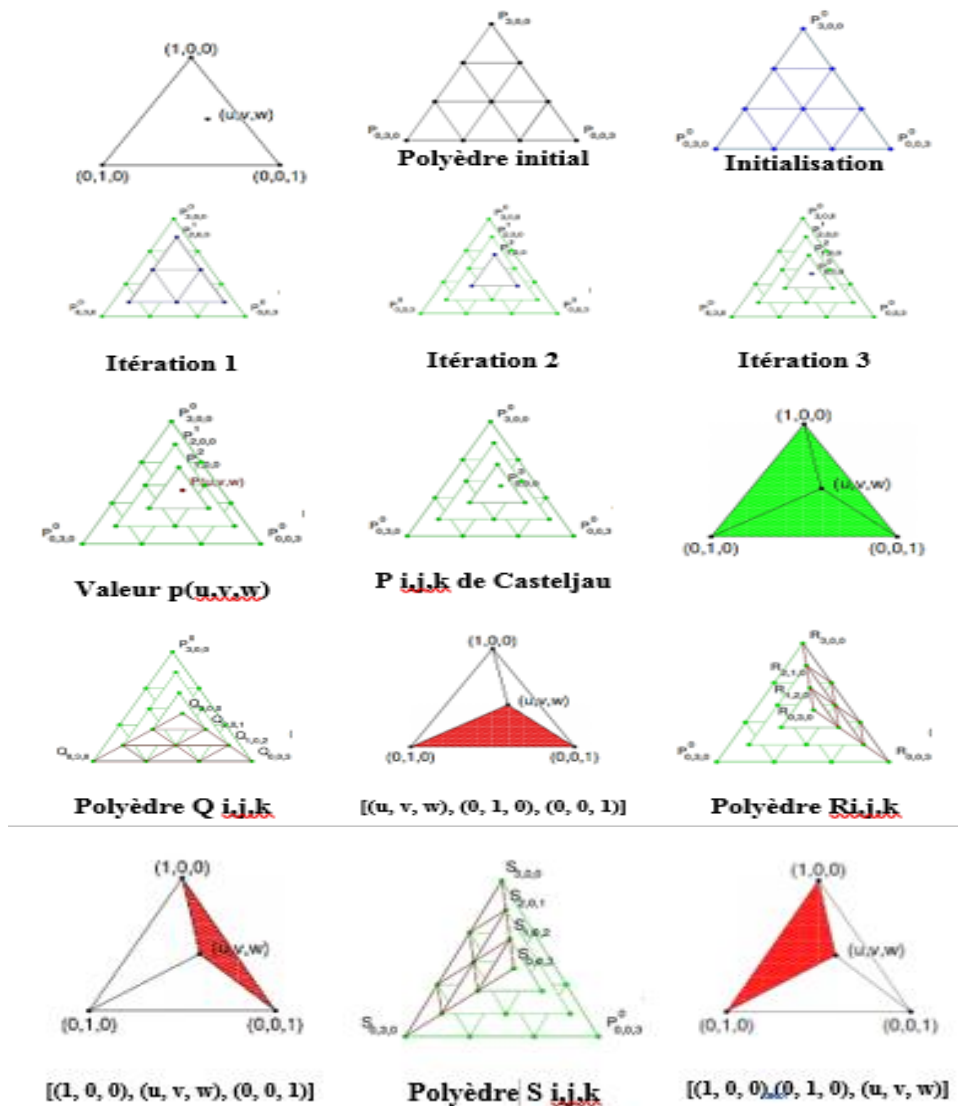


Figure 5-12: Etapes de la subdivision triangulaire de Bézier [145]

## [Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier]

En ce qui concerne les surfaces de produit tensoriel, l'algorithme de Casteljau peut également être utilisé pour subdiviser une surface de Bézier triangulaire en trois patchs splines se touchant en un point donné  $(\mathbf{u}_0, \mathbf{v}_0, \mathbf{w}_0)$ . La figure 5-12 montrent les limites des trois nouveaux réseaux dans le cas  $n = 3$ . Les points de Bézier des trois nouvelles plaques de surface sont à nouveau des points limites du schéma de Casteljau (que nous organisons maintenant comme un tétraèdre). Le patch I a des points de Bézier, le patch II a des points de Bézier et le patch III a des points de Bézier où  $r + j + k = n$ . En combinant plusieurs étapes de l'algorithme de Casteljau, nous pouvons subdiviser un patch de Bézier en un nombre arbitraire de sous-triangles de degré  $n$ . La séquence de surfaces linéaires par morceaux interpolant les réseaux de Bézier converge vers la surface de Bézier.[145]

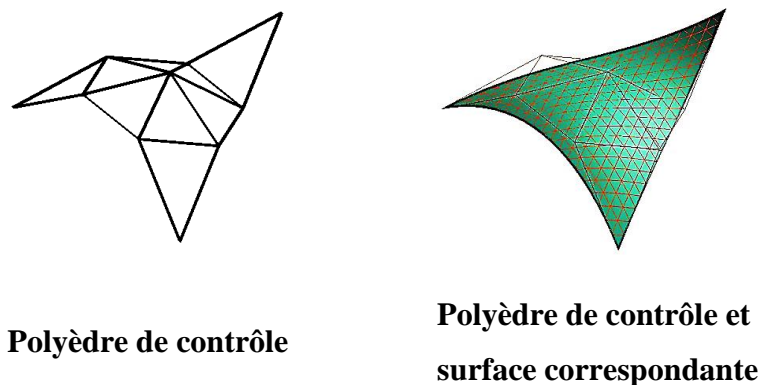


Figure 5-13: Exemple de Bézier triangulaire [145]

### 5.3 Pré-traitement

---

La plupart des approches de détection de reconnaissance d'objets que nous avons déjà annoncées au chapitre 2 sont robustes aux plus fortes déformations usuelles, car l'étape du pré-traitement est très intéressante puisqu'elle normalise les objets. Il faut distinguer trois sortes de déformations qui peuvent distordre un objet 3D [146]:

- La translation, qui fait déplacer un objet relativement au centre du repère ;
- Le changement d'échelle, qui modifie la taille de l'objet ;
- La rotation, qui fait pivoter l'objet.

La figure 5-14 présente un exemple d'un objet ayant passé par ces trois déformations par rapport au repère dedans l'objet.

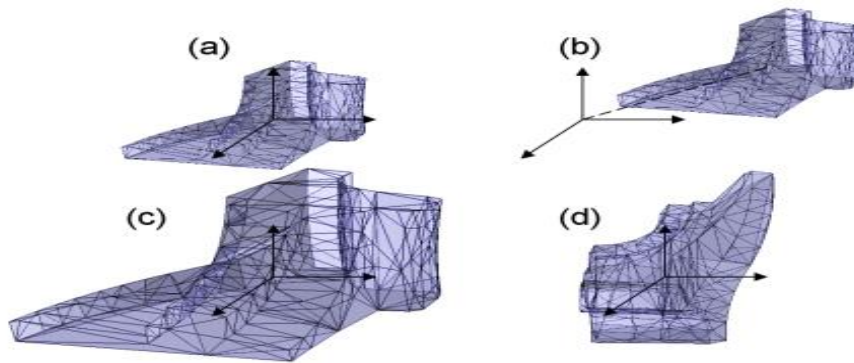


Figure 5-14: Exemple d'objets déformées : objet d'origine (a), objet translaté (b).  
Objet ayant subi un changement d'échelle (c), objet ayant subi une rotation (d). [146]

Il existe des méthodes qui sont normalement adaptées à ce type de distorsion (le descripteur ne fait pas de changement). Nous nous référons à des techniques qui sont invariantes aux rotations, aux changements d'échelle et aux translations. En revanche, si ces méthodes ont un pré-traitement qui normalise les objets avant qu'ils ne soient reconnus, on dit qu'elles sont résistantes à ces déformations.

La normalisation comprend trois points, [146]qui sont :

- Le centrage, qui place le centre de gravité des objets au centre du repère pour renforcer la robustesse des méthodes aux translations.
- La mise à l'échelle, qui consiste à modifier la taille de l'objet pour assurer la robustesse des méthodes aux changements d'échelle.
- L'alignement, qui permet d'aligner les objets par rapport aux autres axes afin de renforcer la robustesse des méthodes aux rotations.

### 5.3.1 Centrage

Le centrage est le fait d'aligner le centre de gravité de l'objet avec le centre du repère. Le calcul du centre de gravité de l'objet,  $G = (p_x, p_y, p_z)$ , peut se faire en additionnant les différentes composantes de chacun des points du maillage de l'objet, selon la formule :

## [Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier]

$$G = \left( \frac{\sum_{i=0}^N a_i}{n}; \frac{\sum_{i=0}^N b_i}{n}; \frac{\sum_{i=0}^N c_i}{n} \right) \quad (5-7)$$

Où  $n$  est le nombre de points formant le maillage de l'objet. [146]

### 5.3.2 Mise à l'échelle

La normalisation a pour but de rendre tous les objets résistants aux changements d'échelle et de même taille. Pour ce faire, les objets sont intégrés dans la sphère unité et au centre du repère avec un rayon de valeur 1 et le point le plus éloigné du centre du repère doit être retrouvé pour tous les points de l'objet centré :

$$N = \max_{i=1..N} d(O, (A_i, B_i, C_i)) \quad (5-8)$$

Où  $d$  est la fonction qui calcule la distance entre les points  $i$  et le centre du repère, et pratiquer une normalisation par  $N$ , pour regrouper tous les points à l'intérieur de la sphère unité selon la formule [146] :

$$(A_i, B_i, C_i) = \left( \frac{a_i}{N}; \frac{b_i}{N}; \frac{c_i}{N} \right) \quad (5-9)$$

La figure 5-15 illustre un exemple de ce processus de mise à l'échelle. [146]

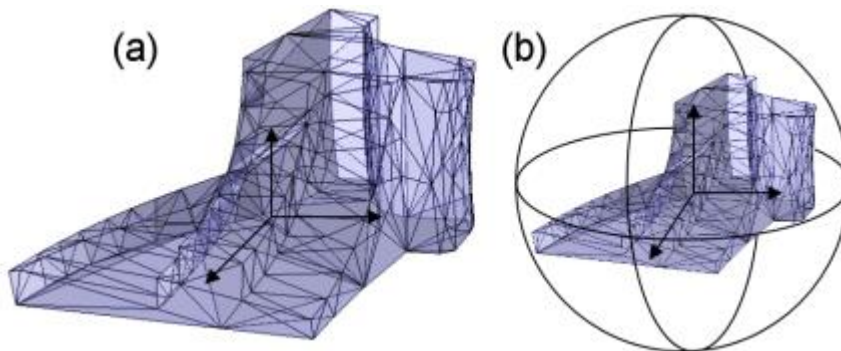


Figure 5-15: Exemple d'un objet mis à l'échelle, objet d'origine (a) et objet intégré (b) [146]

### 5.3.3 Alignement

Grâce à l'alignement, les méthodes de détection et de reconnaissance sont adaptées aux rotations 3D. Le concept d'alignement permet de calculer les principaux axes des objets et de les

## [Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier]

aligner avec ceux du repère. Le calcul des axes principaux d'un objet peut être réalisé en retirant les vecteurs propres de la matrice comprenant les moments géométriques du second-ordre [146] :

$$M = \begin{pmatrix} m_{200} & m_{110} & m_{101} \\ m_{110} & m_{020} & m_{011} \\ m_{101} & m_{011} & m_{002} \end{pmatrix} \quad (5-10)$$

Les principaux axes sont obtenus en effectuant le calcul des vecteurs propres de la matrice M. C'est la méthode dite de l'analyse en composantes principales (ACP). Le calcul des valeurs propres et des vecteurs de la matrice peut être réalisé par la méthode de Jacobi [147]. Il faut alors simplement aligner les vecteurs propres sur les axes du repère en pivotant tous les points de l'objet par [146] :

$$(A_i, B_i, C_i) = \begin{pmatrix} V_{11} & V_{12} & V_{13} \\ V_{21} & V_{22} & V_{23} \\ V_{31} & V_{32} & V_{33} \end{pmatrix} \times \begin{pmatrix} a_i \\ b_i \\ c_i \end{pmatrix} \quad (5-11)$$

Où  $V_i$  sont les vecteurs propres. L'ordre des vecteurs propres est en fonction des valeurs propres. Le vecteur propre dont la valeur propre est la plus grande doit être aligné sur le premier axe de l'axe a, le deuxième sur l'axe b et le troisième sur l'axe c.[146]

**Remarque** : il est ainsi possible de calculer les paramètres de normalisation, qui ne sont plus sur la maille de l'objet, mais sur le volume de l'objet. Il en résulte une normalisation plus robuste, qui nécessite moins de remaillage.

### 5.4 Approche proposée

---

#### 5.4.1 Présentation de la méthode proposée et implémentation

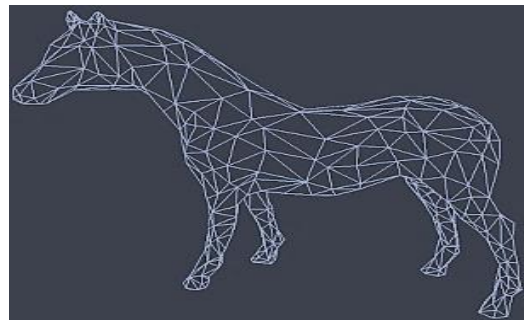
Il existe divers articles qui parlent des extractions des formes libres des objets 3D par plusieurs méthodes. Et comme nous avons déjà mentionné dans l'introduction, nous allons faire une simulation sur les objets tridimensionnels dedans des images 2D en faisant extraire leurs formes à partir de leurs maillages en se basant sur les surfaces de Bézier suivant la position de l'objet concerné, pour avoir un maillage presque exact de l'objet étudié. Ce qui facilite après les tâches de la reconnaissance, la reconstruction et la création d'un objet 3D à partir de son image, et aussi sa



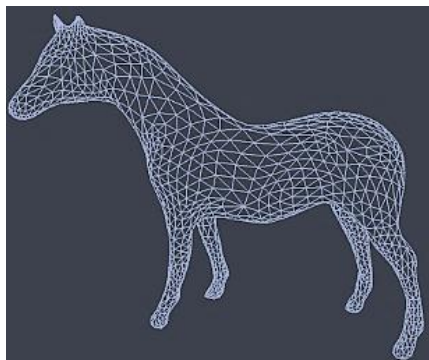
## [Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier]

recherche dans une base de données, ce qui est devenu plus convivial dans la majorité des logiciels et des matériels informatiques qui font de nombreuses recherches.

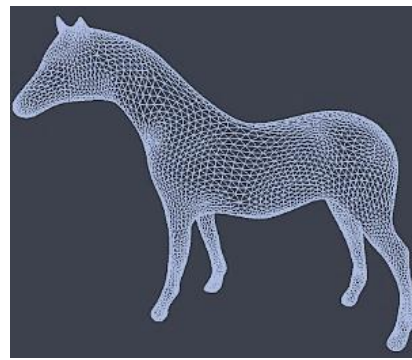
Pour réaliser cette approche, nous devons identifier et bien localiser l'objet qui se trouve sur une image. Tout d'abord on va balayer cette dernière avec une fenêtre en définissant la zone d'intérêt, à chaque position en appliquant un opérateur sur les pixels de la fenêtre afin d'estimer s'il y a une transition significative au niveau de l'attribut choisi, et après le prétraitement cité ci-dessus ,on applique l'algorithme des surfaces de Bezier subdivisé par des polygones triangulaires de Casteljau [141] liés par des points ce qui fait des liaisons entre ces petits triangles en donnant exactement le maillage de l'objet extrait.



Maillage initial



Maillage  
Après un pas  
de subdivision



Maillage  
Après deux pas  
de subdivision

Figure 5-16: Etapes de la subdivision de maillage selon notre implémentation

**[Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier]**

Voici une série d'implémentation des maillages extraits selon notre programme :

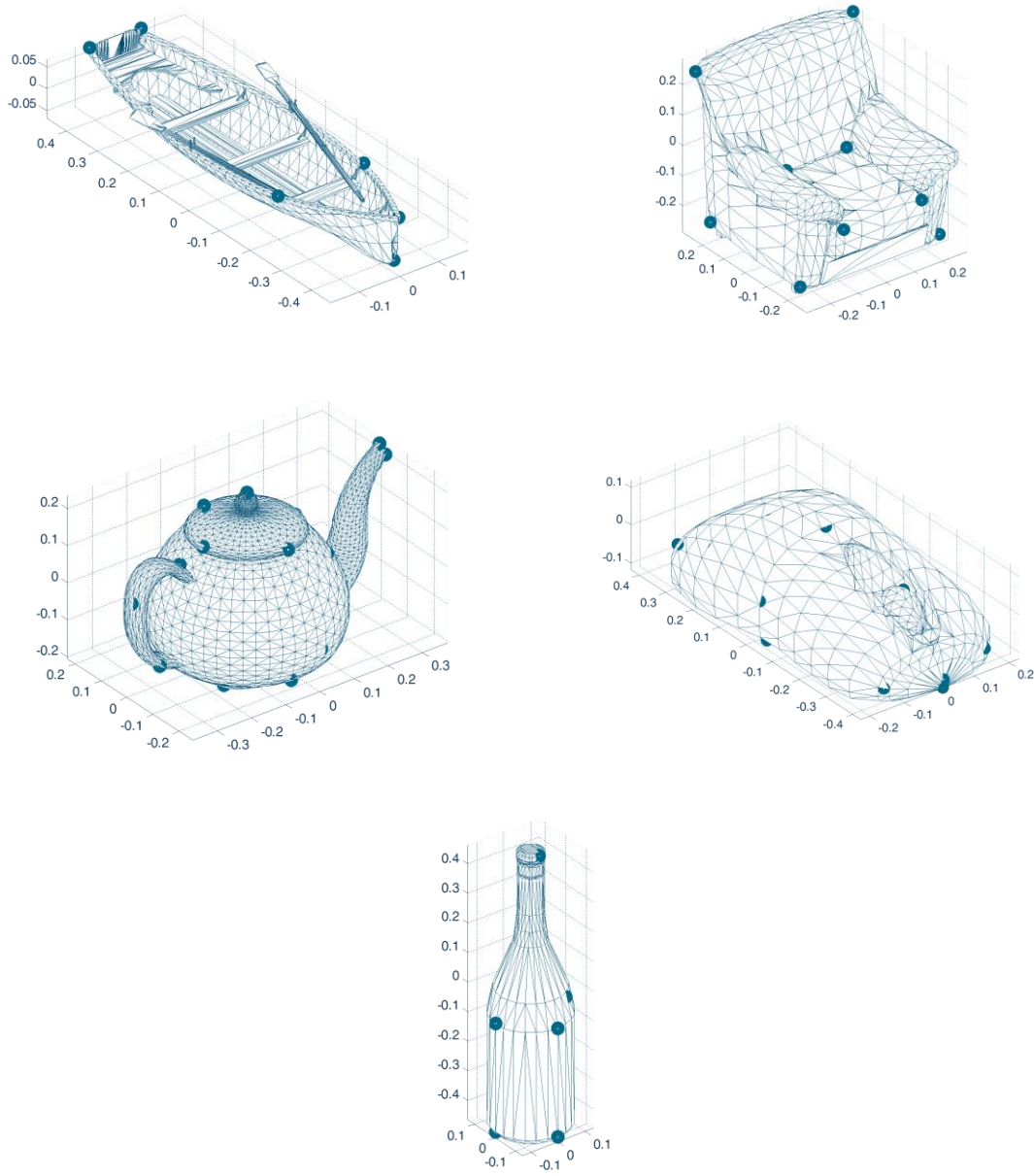


Figure 5-17: Exemples de maillages extraits de notre algorithme

### 5.4.2 Organigramme de l'algorithme proposé

En se basant sur notre approche nous allons suivre l'organigramme suivant pour avoir des maillages presque exacts aux objets étudiés en utilisant toutes les équations précédentes au niveau algorithmique :

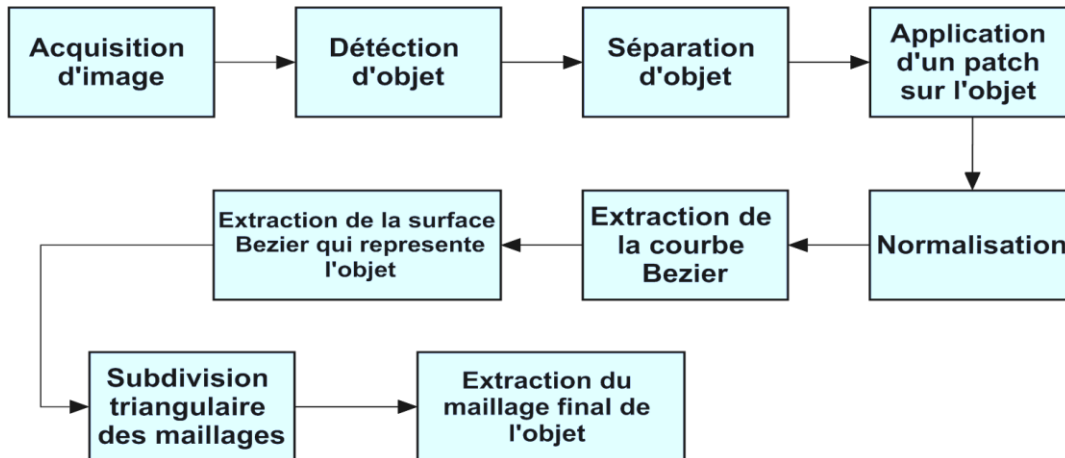


Figure 5-18: Organigramme de l'algorithme proposé

### 5.4.3 Résultats et discussions



[Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier]

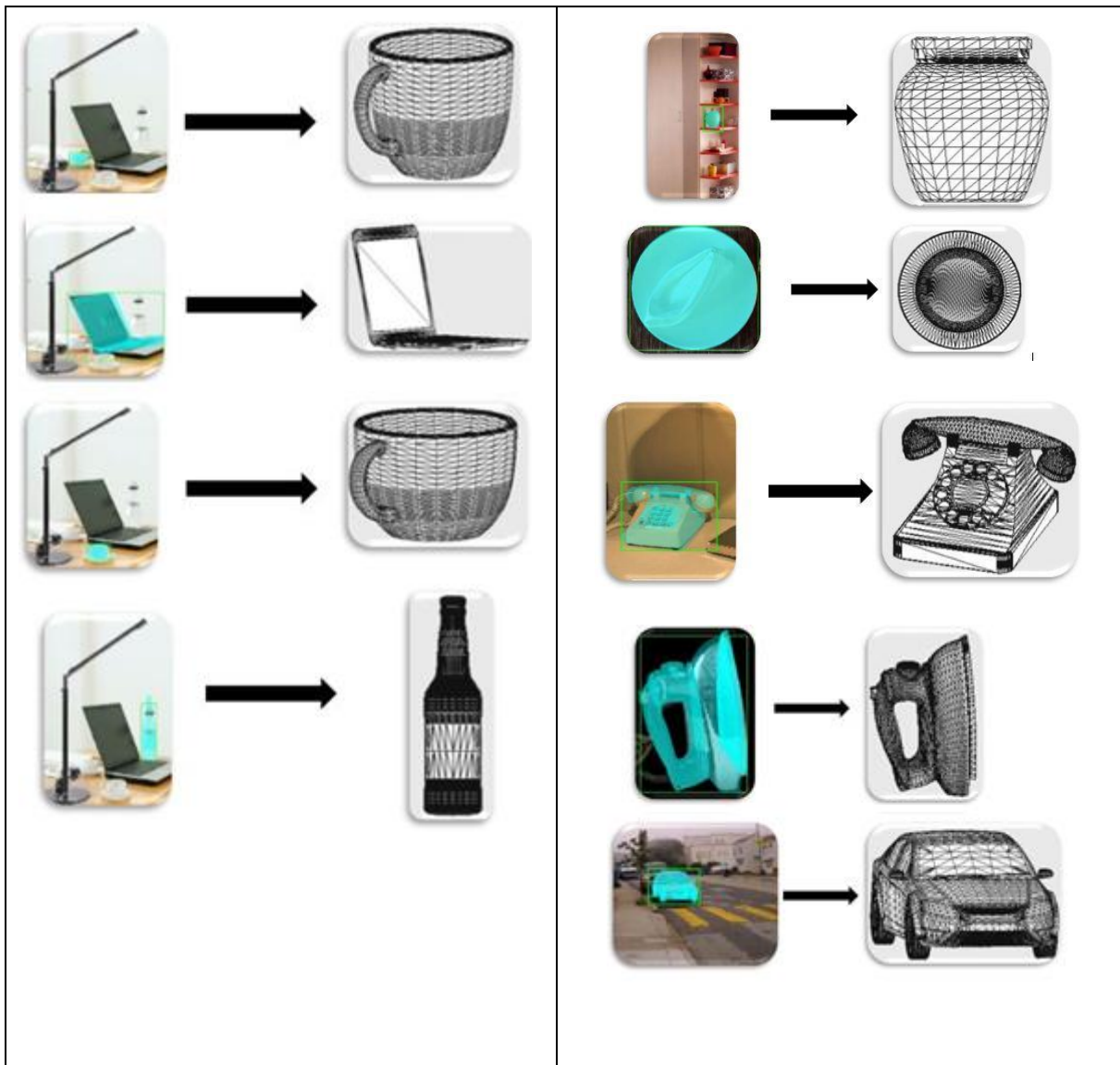


Figure 5-19: Maillages extraits des objets 3D en vue 2D selon notre implémentation

En se basant sur la base de données extraite de [148], nous avons testé plus que 70 images sur un ordinateur d'un processeur Intel Corei5-4200M, CPU @2.50HZ ,4Go (RAM), dont le système d'exploitation est Windows 7 64bits ; pour qu'on puisse obtenir des résultats plus ou moins convaincants.

Afin de bien valoriser ce travail, plus que l'application d'algorithme sur les images 2D on a appliqué le même programme sur les objets 3D pour qu'on puisse voir le rôle de la subdivision triangulaire et la différence entre les maillages des objets 2D et 3D au niveau du nombre des faces

**[Chapitre 4 : Représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier]**

et des sommets et aussi au niveau du temps d'exécutions pendant le traitement et l'extractions, et pour faire une toute petite comparaison, on va résumer tout ça dans le tableau 5-1 ci-dessous :

Tableau 5-1: Comparaison entre les maillages des objets extraits au niveau des données et de temps d'exécution

	Objet 3D en vue 2D (Maillage extrait)			Objet 3D (Maillage original)		
	Faces	Sommets	Temps d'exécution(s)	Faces	Sommets	Temps d'exécution(s)
<b>Avion</b>	30216	27392	0.951	30198	27392	2.329
<b>Voiture</b>	28937	28200	1.305	28929	28200	3.091
<b>Bouteille</b>	2226	600	2.663	1114	600	13.530
<b>Souris</b>	22545	21088	0.271	22539	21088	2.240
<b>Lunettes</b>	16990	8194	1.477	15490	8194	1.991
<b>Chaise</b>	1020	960	1.333	1016	960	3.530
<b>Canapé</b>	7003	2649	0.922	5113	2649	2.029
....	.....	.....	.....	.....	.....	.....

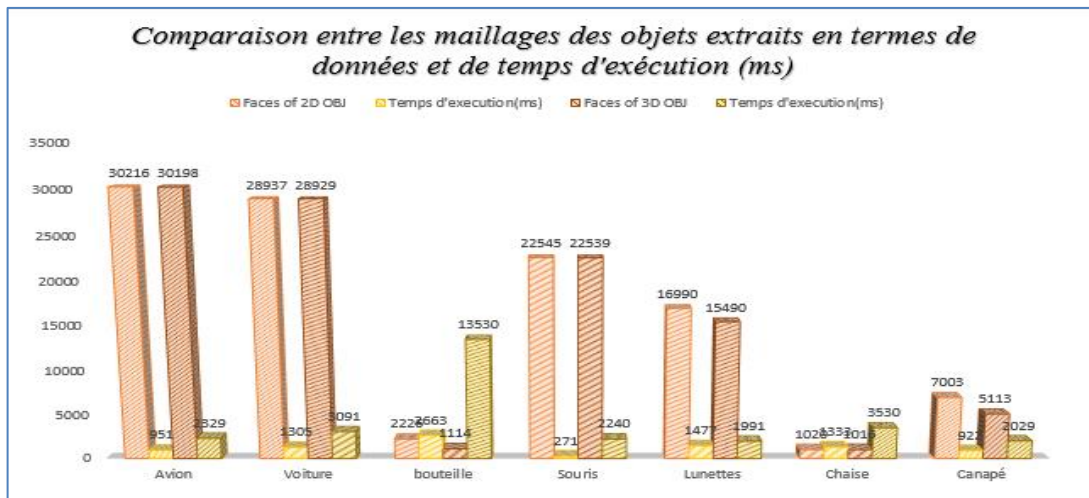


Figure 5-20: Comparaison entre les maillages des objets extraits au niveau des données et de temps d'exécution

D'après ces résultats on voit bien le rôle des subdivisions triangulaires de Bézier qui nous donnent plusieurs faces triangulaires pour les maillages des objets 3D en vue 2D par rapport aux objets 3D qui ce sont des maillages originaux ,et tout ça influence sur la rapidité du temps d'exécution pendant le traitement et la création d'un maillage plus proche au objet concerné , et plus le nombre des faces est élevé ,plus qu'on obtient des maillages plus exacts et lisses et qui représentent une forme plus précise au objet testé.



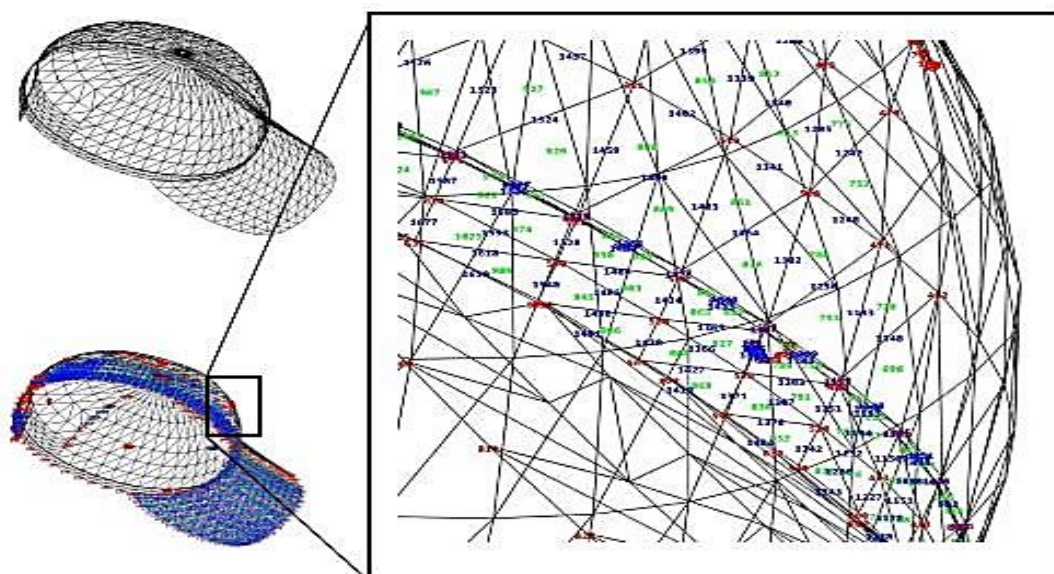


Figure 5-21: Capture plus proche des sommets et des faces d'un maillage extrait d'après notre algorithme

## 5.5 Conclusion

---

Dans ce chapitre, nous avons présenté notre contribution concernant la représentation des objets 3D en vue 2D par l'extraction des maillages via les surfaces de Bézier car la représentation et la catégorisation des objets dans un système de vision permet de faciliter la tâche de la reconnaissance d'objets. Et comme d'habitude, nous avons utilisé une forme géométrique (modélisation en trois dimensions) des prototypes dans le traitement d'images...etc. C'est pour cela nous avons tenté d'utiliser d'autres aspects aussi significatifs comme l'extraction des maillages des objets 3D en vue 2D pour simplifier les tâches de la reconstruction des objets 3D à partir de leurs maillage et aussi leur reconnaissance, et d'après ce qu'on a vu, les résultats obtenus sont encourageants et les maillages résultants nous a montrer l'efficacité et la performance de l'algorithme implémenté soit au niveau du maillage aperçus après l'implémentation et qui ressemble à un grand degré au objet identifié dans l'image, soit au niveau de l'exécution rapide.

## 6 Conclusion générale et perspectives

---

---

### 6.1 Conclusion générale

---

Dans ce manuscrit, nous avons présenté notre travail de recherche que nous avons mené sur la représentation des objets 3D en vue 2D soient par leurs squelettes ou la détection des formes libres par les points d'intérêts ou l'extraction des maillages 3D. Il s'agit de l'élaboration d'outils de prétraitement et plus précisément la phase de la représentation des caractéristiques, la description et l'extraction des caractéristiques d'objets, ce qui permet à la fin à la réalisation d'un système de reconnaissance fiable, car sans la tâche du prétraitement on ne peut pas faire la reconnaissance ou la recherche des objets 2D ou 3D dans une base de données quelconque.

En ce qui concerne la représentation des objets 3D en vue 2D par leurs squelettes, nous avons dans un premier temps, élaboré un algorithme d'amincissement qui permet d'extraire un squelette d'un objet (3D en vue 2D ou voxel) bien défini, bien connecté, lisse, mince, centré avec moins de distorsion et rapidement formé que d'autres squelettes à partir de l'application de deux masques de l'algorithme « Thinning » tout en testant cet algorithme sur deux catégories d'objets, les objets voxels et les objets 3D en vue 2D pour qu'on puisse profiter de l'espace bidimensionnel de l'image à squelettiser et afin de simplifier les tâches de la reconnaissance d'objets par leurs squelettes [72] ou leur recherche dans des bases de données tout en faisant cette recherche seulement par leurs squelettes et pas l'objet entier.

Passant à l'autre chapitre concernant la représentation et l'extraction des formes libres des objets 3D en vue 2D par un détecteur des points d'intérêt « détecteur de SUSAN », nous avons amélioré l'algorithme originale en changeant quelques critères et en ajoutant quelques autres pour obtenir des résultats satisfaisantes à propos de la détection de la forme libre de l'objet testé en faisant une étude comparative entre les objets de la base de données Benchmark et la base de données de Sweet Home 3D et les objets bruités tout en travaillant avec des objets 3D dans un espace 2D et en testant les deux détecteurs de SUSAN soit l'ancien ou le nouveau détecteur amélioré et dans un temps d'exécution trop court par rapport au ancien détecteur avec une très bonne extraction de forme libre des objets résultants ce qui facilite encore la tâche de la reconnaissance des objets et leurs formes par les points d'intérêts [2] .

Dans le dernier chapitre qui est à propos de la représentation des objets par leurs maillages, et afin d'élaborer et améliorer un système de modélisation, de reconnaissance ou de recherche d'objets, nous avons réalisé une application qui fait l'extraction des maillages en fil fer des objets 3D dedans des images 2D tout en passant par des étapes successives. Tout d'abord on a commencé par l'extraction des courbes de Beziere, puis les rassembler dans des surfaces de Beziere en faisant une subdivision triangulaire pour obtenir des maillages bien définis et lisses ce qui nous a donné des maillages qui ressemblent fortement aux objets originaux. Autrement dit, cette approche va aider ceux qui vont créer des systèmes de reconnaissances ou de recherches ou de modélisation, ça veut dire qu'on peut faire une reconnaissance ou recherche d'un objet par son maillage ou encore on peut faire une modélisation d'un objet à partir de son maillage pour le rendre un objet 3D bien défini, c'est-à-dire une reconstruction de l'objet à partir de son maillage.

## 6.2 Perspectives

---

Bien que les résultats obtenus au cours de ce travail soient prometteurs, nous pouvons poursuivre cette thèse selon les orientations suivantes :

- **Création et développement des systèmes de reconnaissances en profitant des trois contributions étudiées dans cette thèse :**
  - *La reconnaissance des objets 3D par leurs squelettes*
  - *La reconnaissance des formes des objets 3D par les points d'intérêt*
  - *La reconnaissance des objets 3D en utilisant leurs maillages*
- **Recherche des objets 3D dans une base de données par :**
  - *Les squelettes*
  - *Les formes extraites par les points d'intérêts*
  - *Les maillages.*
- **Modélisation des objets 3D à l'aide de :**
  - *Des maillages extraits (Reconstruction d'objets)*
  - *L'impression 3D des objets à partir de leurs squelettes et maillages*
- **Détection des mouvements d'objets 3D à l'aide de la détection du mouvement des squelettes**



## Références

---

- [1] Wikipédia, “Wikipédia.” [Online]. Available: <https://fr.wikipedia.org/wiki/2,5D>.
- [2] A. Shaiek, “Reconnaissance d ’objets 3D par points d ’ interet,” Ecole Nationale Supérieure des Mines de Paris, 2013.
- [3] L. Desmecht and M. Rémon, “Reconnaissance d ’ objets tridimensionnels par leurs caractéristiques clés” 2004.
- [4] F. de J. T. Romero, “Modélisation Et Reconnaissance Active D’Objets 3D De Forme Libre Par Vision En Robotique,” l’ Institut National Polytechnique de Toulouse.
- [5] L. He, C. Y. Han, B. Everding, and W. G. Wee, “Graph matching for object recognition and recovery,” *Pattern Recognit.*, vol. 37, no. 7, pp. 1557–1560, 2004.
- [6] D. G. Lowe, “3D Object Recognition from Single 2D Images ,” *Artif. Intell.*, vol. 31, no. April, pp. 355–395, 1987.
- [7] J. L. M. and A. Zisserman, “Geometric Invariance in Computer Vision,” *MIT Press*, p. 560.
- [8] J. P. J. Mundy, M. Hebert, C. Schmid, and A. Z. University, “Toward Category-Level Object Recognition,” *Springer-Verlag Berlin Heidelb.*, no. 3–28, p. 620, 2006.
- [9] R. Jain, R. Kasturi, and B. G. Schunck, “Chapter 15 Object Recognition,” in *Machine Vision*, pp. 459–486.
- [10] B. F. DJABEUR DJEZZAR Mohammed Rafik, “Mise au Point d’une Application de Reconnaissance de Formes,” Tlemcen, 2017.
- [11] K. K. Sergios Theodoridis, *Pattern Recognition*. Elsevier Science & Technology Books, 2003.
- [12] S. L. E. Rafael C. Gonzalez, Richard E. Woods, *Digital Image Processing Using MATLAB 3rd edition*. Gatesmark, 2020.
- [13] G. Mamic and M. Bennamoun, “Representation and recognition of 3D free-form objects,” *Digit. Signal Process. A Rev. J.*, vol. 12, no. 1, pp. 47–76, 2002.
- [14] S. Soltani, “SURVEY A Survey of Thresholding Techniques,” *Comput. VISION, Graph. IMAGE Process. 41*, vol. 41, pp. 233–260, 1988.
- [15] A. K. Chaubey, “Comparison of The Local and Global Thresholding Methods in Image Segmentation,” *World J. Res. Rev.*, vol. 2, no. 1, pp. 1–4, 2016.
- [16] P. Smith, D. B. Reid, C. Environment, L. Palo, P. Alto, and P. L. Smith, “A Tlreshold Selection Method from Gray-Level Histograms,” *IEEE Trans. SYSTREMS, MAN, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, 1995.
- [17] P. R. and E. Ioannidis, “Evaluation of Global Image Thresholding for Change Detection”,” vol. 24, pp. 2345–2356, 2003.

- [18] O. T. and A. Jain, “Goal-Directed Evaluation of Binarization Methods,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 12, pp. 1191–1201.
- [19] S. FISCHER, “Digital Image Processing: Skewing and Thresholding,” 2000.
- [20] L. Lara, M. Gonzalez, S. Vera, and P. Frederic, “A Threshold with Hysteresis,” pp. 1–4.
- [21] “Effets du seuillage à hystérésis sur la détection de contours.”
- [22] K. S. R. Vaka Murali Mohan, R. Kanaka Durga, Swathi Devathi, “Image Processing Representation Using Binary Image; Grayscale, Color Image, and Histogram,” in *Proceedings of the Second International Conference on Computer and Communication Technologies*, 2016, pp. 353–361.
- [23] F. B. N. Russ, J. C., *The image processing handbook*. 2016.
- [24] M. S. De Lawrence O’Gorman, Michael J. Sammon, *Practical Algorithms for Image Analysis*. 2008.
- [25] “Option de traitement d’images Mémento pour la séance N o 6,” 2007.
- [26] S. Kamdi and R. K. Krishna, “Image Segmentation and Region Growing Algorithm,” *Int. J. Comput. Technol. Electron. Eng.*, vol. 2, no. 1, pp. 2249–6343, 2012.
- [27] “Segmentation d’image.” [Online]. Available: [https://fr.wikipedia.org/wiki/Segmentation\\_d%27image#Segmentation\\_par\\_approche\\_frontières](https://fr.wikipedia.org/wiki/Segmentation_d%27image#Segmentation_par_approche_frontières).
- [28] “Partie 3 : Introduction aux Différents Types de Segmentation.” [Online]. Available: <https://patrick-bonnin.developpez.com/cours/vision/apprendre-bases-traitement-image/partie-3-introduction-differents-types-segmentation/#LVI>.
- [29] UFRIMA, “Détection de contours,” *Cours d’analyse d’images*, pp. 1–17, 2007.
- [30] L. D. Cohen, “On Active Contour Models,” *Act. Percept. Robot Vis.*, vol. 331, pp. 599–613, 2011.
- [31] P. A. Yushkevich *et al.*, “User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability,” *Neuroimage*, vol. 31, no. 3, pp. 1116–1128, 2006.
- [32] M. Grand-Brochier, “Descripteurs 2D et 2D + t de points d’intérêt et pour des appariements robustes,” 2011.
- [33] LAKEHAL ELKHAMSSA, “Classification et Reconnaissance d’objets en utilisant les points d’intérêt Mémoire,” 2009.
- [34] “Détection de points d’intérêts - Mise en correspondance.”
- [35] H. Moravec, “Towards automatic visual obstacle avoidance,” in *IJCAI’77 Proceedings of the 5th international joint conference on Artificial intelligence*, 1977, vol. 2, pp. 584–584.

- [36] M. S. C. Harris, "A Combined Corner and Edge Detector," *Proceeding Forth Alvey Vis. Conf*, pp. 147–151, 1988.
- [37] S. C. et H. R. Dufournaud Y., "Matching images with different resolutions," *Comput. Vis. Pattern Recognit.*, pp. 612–618, 2000.
- [38] C. S. K. Mikolajczyk, "Affine invariant interest point detectors," *International Journal of Computer Vision (IJCV)*, vol. 60, pp. 63–86, 2004.
- [39] M. S. and J. Brady, "Susan - a new approach to low level image processing," *Int. J. Comput. Vis.*, vol. 23, pp. 45–78, 1997.
- [40] G. G. Ales, U. P. Sabatier, D. Narbonne, and T. Oulouse, "Détection de points d ' intérêt pour la mise en correspondance par propagation Détecteurs de points d ' intérêt Étapes de détection," 2015.
- [41] D. G. Lowe, "Object Recognition from Local Scale-Invariant Features," in *International Conference on Computer Vision (ICCV)*, 1999, pp. 1–8.
- [42] C. S. K. Mikolajczyk, "A Performance Evaluation of Local Descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1615–1630, 2005.
- [43] B. T. N. Dalal, "Histograms of oriented gradients for human detection," *IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 886–893, 2005.
- [44] T. K. et M. Brady, "Saliency, Scale and Image Description," *Int. J. Comput. Vis.*, pp. 83–105, 2001.
- [45] M. brad. T.kadir, "An Affine Invariant Salient Region Detector," in *Conference: Computer Vision - ECCV 2004, 8th European Conference on Computer Vision*.
- [46] P. Beaudet, "Rotationally invariant image operators," *Int. J. Curr. Pharm. Res.*, pp. 579–586, 1978.
- [47] I. John Wiley & Sons, *2-D and 3-D Image Registration*. 2005.
- [48] L. van G. Herbert Bay, Tinne Tuytelaars, "SURF:Speeded UpRobust Features," in *European Conference on Computer vision*, 2006, pp. 404–417.
- [49] Herbert Bay, Andreas Ess, Tinne Tuytelaars et Luc Van Gool, « SURF: Speeded Up Robust Features », *Computer Vision and Image Understanding*, vol. 110, n o 3, p. 346 359, 2008
- [50] M. AUGÉ and L.-S. EL HACHEM, "Reconnaissance d ' Objets 3D," p. 11.
- [51] B. YEGNANARAYANA, "Artificial Neural Networks," 2009.
- [52] "Détecteur de contraste." [Online]. Available: <https://www.tecnoland.fr/detecteurs-de-contraste-kts-core-sick/>.
- [53] J. I. et J. K. K. Paler, J. F"oglein, "Local ordered grey levels as an aid to corner detection," *J. Pattern Recognit. Soc.*, vol. 17(5), pp. 535–543, 1984.
- [54] C. K. et E. N. L. Itti, "A Model of Saliency-Based Visual Attention for Rapid Scene Analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20(11), pp. 1254–1259, 1998.

- [55] S. I. et B. B., “A robust 3D interest points detector based on Harris operator,” *Eurographics Work. 3D Object Retr.*, pp. 7–14, 2010.
- [56] T. R. et V. G. L. Knopp J., Prasad M., Willems G., “Hough transform and 3D SURF for robust three dimensional classification,” *Eur. Conf. Comput. Vis.*, pp. 589–602, 2010.
- [57] A. S. et S. M. Scovanner P., “A 3-dimensional sift descriptor and its application to action recognition,” in *Proceedings of the 15th international conference on Multimedia.*, 2007, p. 360.
- [58] O. M. et G. L. Sun J., “A concise and provably informative multi-scale signature based on heat diffusion,” *Eurographics Symp. Geom. Process.*, vol. 28, no. 5, pp. 197–206, 2009.
- [59] T. J. et V. RC, “Polyhedral model retrieval using weighted point sets,” *Int. J. Image Graph.*, vol. 3, no. 1, pp. 119–129, 2003.
- [60] L. X. et G. I., “3D object recognition from range images using pyramid matching,” *IEEE 11th Int. Conf. Comput. Vis.*, pp. 1–6, 2007.
- [61] L. X. et G. I., “Multi-scale features for approximate alignment of point-based surfaces,” in *Proceedings of the third Eurographics symposium on Geometry processing*, 2005, p. 217.
- [62] K. Novatnack, J., Nishino, “Scale-dependent 3D geometric features,” *ICCV*, pp. 1–8, 2007.
- [63] H. Dutagaci, C. P. Cheung, and A. Godil, “Evaluation of 3D interest point detection techniques via human-generated ground truth,” *Vis. Comput.*, vol. 28, no. 9, pp. 901–917, 2012.
- [64] C. Petitjean, “Techniques en traitement et analyse d’images,” 2007. [Online]. Available: [http://urfist.chartes.psl.eu/anciensite/image\\_numerique/analyse.htm](http://urfist.chartes.psl.eu/anciensite/image_numerique/analyse.htm).
- [65] P. Abbott, “Image Processing in C,” *Assem. Autom.*, vol. 18, no. 3, pp. 126–142, 1998.
- [66] M. Sinecen, “Digital Image Processing with MATLAB,” *Appl. from Eng. with MATLAB Concepts*.
- [67] L. Lam and S.-W. Lee, “Thinning Methodologie - A Comparative Survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 9, pp. 869–885, 1992.
- [68] S. Lobregt, P. W. Verbeek, and F. C. A. Groen, “Three-Dimensional Skeletonization: Principle and Algorithm,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-2, no. 1, pp. 75–77, 1980.
- [69] K. J. H. and K. Preston, “Three-Dimensional skeletonization of elongated solids,” *Comput. Vision, Graphics Image Process.*, vol. 27, pp. 78–91, 1984.
- [70] K. S. F. YF. Tsao, “A parallel thinning algorithm for 3D pictures,” *Comput. Graph. Image Process.*, vol. 17, pp. 315–331, 1981.
- [71] W. G. and G. Bertrand, “A simple parallel 3D thinning algorithm,” *10th Int. Conf. Pattern Recognit.*, pp. 188–190, 1990.

- [72] D. Merad and R. Reconnnaissance, “Reconnnaissance 2D/2D et 2D/3D d’objets à partir de leurs squelettes,” 2004.
- [73] C.Arcelli, “Pattern thinning by contour tracing,” *Comput. Graph. Image Process.*, vol. 17, no. 2, pp. 130–144, 1981.
- [74] Y.Xia, “Skeletonization via the realization of the fire front’s propagation and extinction in digital binary shapes,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 10, pp. 1076–1086, 1989.
- [75] F. L. ; M. D. Levine, “Simulating the grassfire transform using an active contour model,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 1, pp. 56–75.
- [76] V. Marion-poty, “A new 2 -D and 3 -D thinning algorithm based on successive border generations generations,” *Esprit*, no. 94, 1994.
- [77] G.Borgefors, “Distance transformations in digital images,” *Comput. Graph. Image Process.*, vol. 34, pp. 344–371, 1986.
- [78] P.E.Danielsson, “Euclidean distance mapping,” *Comput. Graph. Image Process.*, vol. 7, pp. 227–248, 1980.
- [79] P. B. G. and D. W. C. C.W.Niblack, “Generating skeletons and centerlines from the distance transform,” *CVGIP Graphical Model. Image Process.*, vol. 54(5), pp. 420–437, 1992.
- [80] J. and J. L. N. M.P.Martines-Perez, “A thinning algorithm based on contours,” *Comput. Graph. Image Process.*, vol. 39, pp. 186–201, 1987.
- [81] A. M. and J.-M. C. F.Rolland, “3D medial axis and 3D skeletons,” in *the 7th Scandinavian Conf on Image Analysis*, 1991, pp. 395–402.
- [82] D. A. and A.Montanvart, “Squelettes et diagrammes de Voronoi,” *Tech. Rep. RR922,IMAG,Grenoble*, 1993.
- [83] J. W. B. and V.R.Algazi, “Continuous skeleton computation by Voronoi diagram,” *CVGIP Image Underst.*, vol. 55, no. 3, pp. 329–337, 1992.
- [84] J. D. B. and B.Geiger, “Three dimensional reconstruction of complex shapes based on the Delaunay triangulation,” *Tech. Rep. No 1697,INRIA*.
- [85] G. B. and Z. Aktouf., “A three-dimensional thinning algorithm using subfields,” *Vis. Geom. III*, pp. 113–124, 1994.
- [86] T. L. and R. L. Kashyap, “Building skeleton models via 3-d medial surface/axis thinning algorithms,” *CVGIP Graph. Model. Image Process.*, vol. 56, no. 6, pp. 462–478.
- [87] C. L. and G. Bertrand, “A 3D 12-subiteration thinning algorithm based on P-simple points,” *Discret. Appl. Math. , Elsevier*, vol. 139, pp. 171–195, 2004.
- [88] G. M. U. Eckhardt, “Invariant Thinning, Pattern Recognition and Artificial Intellicence,” vol. 7, pp. 1115–1144, 1993.

- [89] C. M. M. and M. Sonka., “A fully parallel 3d thinning algorithm and its applications,” *Comput. Vis. Image Underst.*, vol. 64, no. 3, pp. 420–433, 1996.
- [90] G. S. D. B. G. Borgefors, I. Nyström, “Computing skeletons in three dimensions,” *Pattern Recognit.*, vol. 32, no. 7, 1999.
- [91] K. Suresh., “Automating the CAD/CAE Dimensional Reduction Process, ACM Symp,” *Solid Model. Appl.*, 2003.
- [92] J. A. Sethian, “Fast Marching Methods,” *SIAM Rev.*, vol. 41(2), pp. 199–235, 1999.
- [93] A. W. T. Y. Zhou, A. Kaufman, “Three-dimensional Skeleton and Centerline Generation Based on an Approximate Minimum Distance Field,” *Vis. Comput.*, vol. 14, pp. 303–314, 1998.
- [94] R. E. P. L. Wade, “Automated generation of control skeletons for use in animation,” *Vis. Comput.*, vol. 18(2), pp. 97–110, 2002.
- [95] S. M. P. C.A. Burbeck, “Object representation by cores: Identifying and representing primitive spatial regions,” *Vision Res.*, vol. 35(13), pp. 1917–1930, 1995.
- [96] A. L. T. S.M. Pizer, “M-reps: A new object representation for graphics,” *Tech. Rep. TR99-030, Univ. North Carolina, Chapel Hill, NC*, vol. 17, 1999.
- [97] B. B. K. F.F. Leymarie, “Computation of the Shock Scaffold for Unorganized Point Clouds in 3D,” in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2003.
- [98] R. K. N. Amenta, S. Choi, “The Power Crust,” in *Proceedings of 6th ACM Symposium on Solid Modeling*, 2001, pp. 249–260.
- [99] F. L. A. Verroust, “Extracting skeletal curves from 3D scattered data,” *Vis. Comput.*, vol. 16, pp. 15–25, 2000.
- [100] Z. H. X. Li, T.W. Woon, “Decomposing polygon meshes for interactive applications,” in *Proc. ACM Symp. On Interactive 3D Graphics*, pp. 35–42, 2001
- [101] A. T. S. Katz, “Hierarchical mesh decomposition using fuzzy clustering and cuts,” in *Proc. ACM SIGGRAPH*, 2003.
- [102] J. C. N. Ahuja, “Shape Representation Using a Generalized Potential Field Model,” *IEEE PAMI*, vol. 19(2), pp. 169–176, 1997.
- [103] M.-C. K. J. Chuang, C. Tsai, “Skeletonization of ThreeDimensional Object Using Generalized Potential Field,” *IEEE PAMI*, vol. 22(11), pp. 1241–1251, 2000.
- [104] Y. H. Y. T. Grigorishin, “Skeletonization: An Electrostatic FieldBased Approach,” *Pattern Anal. App.*, vol. 1, pp. 163–177, 1998.
- [105] M. O. F. Wu, W. Ma, P. Liou, R. Liang, “Skeleton Extraction of 3D Objects with Visible Repulsive Force,” in *Eurographics Symp. On Geometry Processing*, 2003.
- [106] M. O. W. Ma, F. Wu, “Skeleton Extraction of 3D Objects with Radial Basis Functions,” *IEEE SMA*, 2003.

- [107] R. B. N. Cornea, D. Silver, X. Yuan, “Computing Hierarchical Curve-Skeletons of 3D Objects,” *Vis. Comput.*, vol. 21(11), pp. 945–955.
- [108] “Représentation des courbes et des surfaces.”2009.
- [109] G. P. L., “Mailleur 3D par découpage structuré d’éléments grossiers,” INRIA, 1989.
- [110] P. L. George, P. L. G. Génération, and E. Finis, “Génération des maillages pour la simulation par Eléments Finis de problèmes physiques,” 1990.
- [111] G. A., “Mailleur tridimensionnel automatique pour des géométries complexes,” INRIA, 1989.
- [112] C. J. L., “Maillages 2D et 3D. Expérimentation de la triangulation de Delaunay,” in *Conf. on Automated mesh generation and adaptation, Grenoble*, 1987.
- [113] W. D. F., “Computing the n-dimensional Delaunay tessellation with applications to Voronoï polytopes,” *Comput.*, vol. 24, no. 2, 1981.
- [114] S. M. S. YERRI M. A., “Automatic 3D mesh generation by the modified-octree technique,” *Int. J. Num. Meth. Eng.*, vol. 20, 1984.
- [115] H. A. F. CHENG J. H., FINNIGAN P. M. and S. W. J. KELA A., “Quadtrees/octree meshing with adaptive analysis,” *Numer. grid Gener. Comput. fluid Mech.*, 1988.
- [116] G. A. GEORGE P. L., “Mailleur 3D en topologie « cylindrique »,” INRIA, 1988.
- [117] F. J. E. HEPHARD M. S., GUERINONI F. and B. P. L. LUDWIG R. A., “Finite octree mesh generation for automated adaptive 3D flow analysis,” *Numer. grid Gener. Comput. fluid Mech.*, 1988.
- [118] M. Hu, “Visual pattern recognition by moment invariants,” *IEEE Trans. Inf. Theory*, vol. 8, pp. 179–187, 1962.
- [119] and R. M. C-W. Chong, P. Raveendran, “A comparative analysis of algorithms for fast computation of zernike moment,” *Pattern Recognit.*, vol. 36, pp. 731–742, 2003.
- [120] M. S. and D. Ballard, “Color indexing,” *Int. J. Comput. Vis.*, vol. 7, pp. 11–32, 1991.
- [121] B. S. and A. Waibel., “Gaze tracking based on face-color,” *Int. Work. Autom. Face Gesture Recognit.*, pp. 344–349, 1995.
- [122] B. A. Y. Kamel, “Un système basé sur la vision par ordinateur pour la prévention des dangers des chameaux dans les routes sahariennes,” 2016.
- [123] J. B. Gabriela Csurka, Martin Kraus, Leonid Mestetskiy, Paul Richard, “Etude comparative de descripteur invariants d’objets,” *ORASIS*, 2005.
- [124] L. J. and O. Gwun, “A comparison of sift, pca-sift and surf,” *Int. J. Image Process.*, vol. 3(4), p. 143–152, 2009.
- [125] M. D. X. Peng, C. Zhou, “Corner detection method based on wavelet transform,” *Proc. SPIE*, vol. 4550, pp. 319–323, 2001.

- [126] R. L. R. Elias, “Cones: a new approach towards corner detection,” *Proc Can. Conf. Electr. Comput. Eng.*, vol. 2, pp. 912–916, 2002.
- [127] H. Freeman, “On the encoding of arbitrary geometric configurations,” *IRE Trans. Electron. Comput. EC*, vol. 10, pp. 260–268, 1961.
- [128] B. M. Y. Ouadid, M. Fakir, “Reconnaissance des caractères Tifinagh Imprimés par une Description Structurelle,” *7ème Ed. la Conférence Int. sur Les Technol. d’Information Commun. pour l’Amazighe, TICAM’16*, 2017.
- [129] A. Leborgne, “Appariement de formes basé sur une squelettisation hiérarchique,” Université de Lyon, 2016.
- [130] G. Matheron, “Examples of topological properties of skeletons,” *Image Anal. Math. Morphol.*, vol. 2, no. chap 11, pp. 217–238.
- [131] N. D. Cornea, D. Silver, and P. Min, “Curve-skeleton properties, applications, and algorithms,” *IEEE Trans. Vis. Comput. Graph.*, vol. 13, no. 3, pp. 530–548, 2007.
- [132] et G. S. di B. Gunilla Borgefors, Giuliana Ramella, “Hierarchical decomposition of multiscale skeletons,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23(11), pp. 1296–1312, 2001.
- [133] Alexander C. Zorach et Robert E. Ulanowicz, « Quantifying the complexity of flow networks: How many roles are there? », *Complexity*, vol. 8, n° 3, 2003, p. 68–76 ,ISSN 1099-0526, DOI [10.1002/cplx.10075](https://doi.org/10.1002/cplx.10075)
- [134] C. Schmid, “ppariement D’images par Invariants Locaux de Niveaux de Gris Application à L’indexation d’une Base D’objets,” Institut National Polytechnique de Grenoble, 1996.
- [135] C. Leverger, “Détection des contours d’une image : le filtre de Canny,” 2017.
- [136] J. De Vries, “Object Recognition : A Shape-Based Approach using Artificial Neural Networks,” University of Utrecht, 2006.
- [137] “sweet-home-3D.” [Online]. Available: <http://www.sweethome3d.com/fr/>.
- [138] M. C. Mongenet, M. M. Granet, M. D. Trystram, and M. T. Priol, “Conception d’une méthode de maillage 3D parallèle pour la construction d’un modèle de Terre réaliste par la tomographie sismique,” 2006.
- [139] M. Synthèse and F. Aubert, “Chapitre 5 : Courbes et surfaces,” 2015, pp. 1–43.
- [140] U. PASCAL MIGNOT and DE REIMS, “Modélisation des courbes et surfaces,” pp.71-99.
- [141] “Algorithme\_de\_Casteljau.” [Online]. Available: <http://www.bibmath.net/dico/index.php?action=affiche&quoi=./d/decasteljau.html>
- [142] U. F. R. Im, “TP6 : Surface produit tensoriel - Surface de Bézier,” 2015.
- [143] A. M. L. Cad, “Bezier Surfaces Recall how we constructed a Bezier Curve.”
- [144] E. Catmull, “Edwin Catmull.” [Online]. Available: [https://fr.wikipedia.org/wiki/Edwin\\_Catmull](https://fr.wikipedia.org/wiki/Edwin_Catmull).



- [145] N. Szafran, “Courbes & Surfaces de subdivision Surface de subdivision,” 2008.
- [146] J. Ricard, “Indexation et recherche d ’ objets 3D a partir de requêtes 2D et 3D,” pp. 1–234, 2005.
- [147] F. J. M. , J. von NeumannGoldstine, H. H., “The Jacobi Method for Real Symmetric Matrices,” *J. ACM (JACM)*, vol. 6, no. 1, pp. 59–96.
- [148] “Base de donnée : ImageNet.” [Online]. Available: <http://image-net.org/>.