



**HAL**  
open science

# Etude des courbes discrètes : applications en analyse d'images

Thanh Phuong Nguyen

► **To cite this version:**

Thanh Phuong Nguyen. Etude des courbes discrètes : applications en analyse d'images. Autre [cs.OH].  
Université Henri Poincaré - Nancy 1, 2010. Français. NNT : 2010NAN10095 . tel-01746364

**HAL Id: tel-01746364**

**<https://hal.univ-lorraine.fr/tel-01746364>**

Submitted on 29 Mar 2018

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

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



## AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : [ddoc-theses-contact@univ-lorraine.fr](mailto:ddoc-theses-contact@univ-lorraine.fr)

## LIENS

Code de la Propriété Intellectuelle. articles L 122. 4

Code de la Propriété Intellectuelle. articles L 335.2- L 335.10

[http://www.cfcopies.com/V2/leg/leg\\_droi.php](http://www.cfcopies.com/V2/leg/leg_droi.php)

<http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm>

# Etude des courbes discrètes : applications en analyse d'images

## THÈSE

présentée et soutenue publiquement le

pour l'obtention du

Doctorat de l'université Henri Poincaré – Nancy 1  
(spécialité informatique)

par

Thanh Phuong NGUYEN

### Composition du jury

*Rapporteurs :* Jean Pierre REVEILLÈS, LAIC, Professeur émérite, Université Clermont-Ferrand 1  
Eric ANDRÈS, SIC, Professeur, Université de Poitiers

*Examineurs :* Philippe EVEN, LORIA, Professeur, Université Nancy I  
Laurent WENDLING, LIPADE, Professeur, Université Paris V  
David COEURJOLLY, LIRIS, Chargé de recherche (HDR), CNRS

*Directrice de thèse :* Isabelle DEBLED-RENNESON, LORIA, Maître de conférences (HDR), Université Nancy I

Mis en page avec la classe thloria.

## Remerciements

Les remerciements...



*A mes parents, ma femme et mon fils.*





# Table des matières

<b>Introduction</b>	<b>1</b>
<b>Chapitre 1 Notions de base</b>	<b>5</b>
1.1 Introduction . . . . .	5
1.2 Espace discret . . . . .	5
1.3 Connexité . . . . .	6
1.4 Codage . . . . .	7
1.5 Discrétisation . . . . .	9
1.6 Les primitives discrètes . . . . .	9
1.6.1 Droite discrète 2D . . . . .	9
1.6.2 Droite discrète 3D . . . . .	14
1.6.3 Segment flou . . . . .	15
1.6.4 Etude de l'algorithme de reconnaissance d'un segment flou . . . . .	19
1.6.5 Segment flou 3D . . . . .	23
1.6.6 Cercle discret . . . . .	26
1.6.7 Plan discret . . . . .	29
1.6.8 Morceau flou de plan discret . . . . .	30
1.6.9 Conclusion . . . . .	31
<b>Chapitre 2 Estimation des propriétés locales</b>	<b>33</b>
2.1 Introduction . . . . .	33
2.2 Etat de l'art . . . . .	34
2.2.1 Estimation de la courbure . . . . .	34
2.2.2 Estimation de la torsion . . . . .	40
2.3 Evolution d'un segment flou d'épaisseur $\nu$ . . . . .	42
2.3.1 Maintenance de l'enveloppe convexe . . . . .	42
2.3.2 Détermination de l'épaisseur de l'enveloppe convexe . . . . .	44
2.4 Segment flou maximal d'épaisseur $\nu$ . . . . .	45
2.4.1 Notion de segment flou maximal . . . . .	45

2.4.2	Une propriété importante de la suite des segments flous maximaux d'une courbe . . . . .	45
2.4.3	Détermination de la suite des segments flous maximaux d'une courbe . . .	45
2.5	Courbure discrète d'épaisseur $\nu$ . . . . .	46
2.5.1	Définition . . . . .	46
2.5.2	Estimation de la courbure d'épaisseur $\nu$ en chaque point $C_k$ de $C$ . . . . .	47
2.6	Torsion discrète d'épaisseur $\nu$ . . . . .	49
2.6.1	Définitions . . . . .	49
2.6.2	Torsion discrète . . . . .	50
2.7	Expérimentations et comparaisons . . . . .	51
2.7.1	Expérimentations . . . . .	51
2.7.2	Comparaisons avec les méthodes existantes . . . . .	57
2.8	Conclusions et perspectives . . . . .	58
<b>Chapitre 3 Détection de points dominants et représentation polygonale</b>		<b>63</b>
3.1	Introduction . . . . .	63
3.2	Etat de l'art . . . . .	64
3.2.1	Les méthodes de détection de points dominants . . . . .	64
3.2.2	Les approches multi-échelle, multi-épaisseur . . . . .	68
3.3	Critères d'évaluation . . . . .	68
3.3.1	Les critères existants . . . . .	68
3.3.2	Un nouveau critère alternatif d'évaluation pour les courbes bruitées . . . . .	69
3.4	Détection de points dominants avec un paramètre d'épaisseur $\nu$ . . . . .	72
3.4.1	Points dominants et région de support (ROS) . . . . .	72
3.4.2	Relation entre les points dominants et la séquence de segments flous maximaux . . . . .	73
3.4.3	Algorithme proposé . . . . .	74
3.4.4	Expérimentations . . . . .	76
3.4.5	Discussion sur le nouveau critère d'évaluation . . . . .	86
3.5	Approches multi-épaisseurs . . . . .	89
3.5.1	Une méthode multi-épaisseur pour la représentation polygonale des courbes bruitées . . . . .	89
3.5.2	Méthode adaptative multi-échelle reposant sur une stratégie de découpage-et-fusion . . . . .	90
3.5.3	Expérimentations . . . . .	93
3.6	Conclusions . . . . .	98

---

<b>Chapitre 4 Cercle discret et applications</b>	<b>103</b>
4.1 Introduction . . . . .	104
4.2 Etat de l'art . . . . .	104
4.2.1 Reconnaissance des cercles discrets . . . . .	105
4.2.2 Mesure de circularité . . . . .	106
4.2.3 Décomposition d'une courbe en arcs et segments de droite . . . . .	107
4.3 Reconnaissance des cercles discrets . . . . .	107
4.3.1 Représentation des arcs dans l'espace des tangentes . . . . .	108
4.3.2 Propriétés d'un arc dans la représentation modifiée de l'espace des tangentes	108
4.3.3 Détection des arcs/cercles discrets . . . . .	111
4.3.4 Approximation de l'erreur . . . . .	112
4.3.5 Segmentation en arcs discrets . . . . .	117
4.3.6 Expérimentations . . . . .	118
4.4 Segmentation d'une courbe bruitée en arcs discrets . . . . .	120
4.4.1 Détection du bruit non-supervisée . . . . .	120
4.4.2 Détection des arcs dans une courbe bruitée . . . . .	127
4.4.3 Expérimentations . . . . .	129
4.5 Décomposition d'une courbe en arcs et segments de droite . . . . .	133
4.5.1 Idée principale . . . . .	133
4.5.2 Analyse des configurations . . . . .	133
4.5.3 Algorithme proposé . . . . .	136
4.5.4 Discussion . . . . .	136
4.5.5 Expérimentations . . . . .	137
4.6 Mesure de la circularité . . . . .	139
4.6.1 Similarité des courbes polygonales . . . . .	142
4.6.2 La représentation d'un cercle réel dans l'espace des tangentes . . . . .	143
4.6.3 Mesure de circularité et notre algorithme . . . . .	144
4.6.4 Expérimentations . . . . .	146
4.7 Conclusion . . . . .	154
<b>Conclusions et Perspectives</b>	<b>159</b>
<b>Liste des publications</b>	<b>165</b>
<b>Bibliographie</b>	<b>167</b>



# Liste des tableaux

2.1	Estimation d'erreur sur les résultats de courbure, $N^0$ : nombre des points, le temps de calcul donné en ms. . . . .	53
2.2	Mesure de l'erreur sur les résultats d'estimation de la torsion en utilisant notre méthode sans (avec) un seuil. $N^0$ - nombre des points, $N^1$ - borne, temps est calculé en ms. Dans la seconde ligne, sans : sans seuil, avec : avec un seuil = 0.0005. . . . .	57
2.3	Estimation d'erreur sur les résultats d'estimation de la torsion en utilisant la méthode de Lewiner [LaDGJLC05] sur la courbe de Viviani sans (avec) un seuil. $N^0$ - nombre des points, $N^1$ - borne, temps est calculé en ms. Dans la seconde ligne, sans : sans seuil, avec : avec un seuil = 0.0005. . . . .	57
2.4	Estimation d'erreur sur les résultats d'estimation de la torsion en utilisant la méthode de Mokhtarian [Mok97]. $N^0$ - nombre des points, $N^1$ - borne, temps est calculé en ms. L'implémentation est développée et effectuée sur Matlab 7.5.0. . . . .	57
3.1	Comparaison en utilisant le critère de Sarkar [Sar93]. . . . .	78
3.2	Comparaison en utilisant le critère de Rosin [Ros97] sur la courbe de demi-cercle. . . . .	79
3.3	Erreur d'approximation de notre méthode sur les courbes des figures 3.13, 3.31, 3.15. . . . .	81
3.4	Résultats de la vectorisation sur la figure 3.20. Le temps est calculé en ms, $N_C^0$ - nombre de courbes, $N_P^0$ - nombre de points, $N_{DP}^0$ - nombre de points dominants, CR - ratio de compression. . . . .	86
3.5	Comparaison entre le profil de notre critère $ND - FOM$ et celui de Marji [MS04] $M - FOM$ en augmentant l'épaisseur de travail sur les courbes a, c et h dans la figure 3.22 la courbe de feuille dans la figure 3.31. . . . .	87
3.6	Résultats de la détection de points dominants sur la courbe (c) de la figure 3.22 à l'épaisseur 4 et 5. . . . .	88
3.7	Résultats obtenus dans la figure 3.28. . . . .	94
3.8	Résultats obtenus dans la figure 3.30. . . . .	96
3.9	Comparaisons sur les courbes des figures 3.33, 3.32, 3.34 avec Debled et al. [DRTW04], Sarkar [Sar93], Ansari et al. [AwH91] et Marji et al. [MS04]. . . . .	98
4.1	Mesures de circularité des figures 4.36, 4.37. . . . .	148



# Liste des Algorithmes

1	Segmentation des courbes 8-connexes en segments flous. . . . .	18
2	Reconnaissance incrémentale d'un segment flou à l'épaisseur $\nu$ . . . . .	20
3	Calcul de la hauteur à chaque étape. . . . .	24
4	Calcul de la largeur à chaque étape. . . . .	25
5	Construction de la droite englobante optimale pour un segment flou. . . . .	26
6	Algorithme de décomposition d'une courbe 2D $C$ en sa suite de segments flous maximaux d'épaisseur $\nu$ . . . . .	46
7	Algorithme de décomposition d'une courbe 3D $C$ en sa suite de segments flous maximaux d'épaisseur $\nu$ . . . . .	47
8	Estimation de la courbure d'épaisseur $\nu$ en chaque point d'une courbe 2D ou 3D. . . . .	48
9	Estimation de la torsion d'épaisseur $\nu$ en chaque point d'une courbe 3D $\zeta$ . . . . .	52
10	Détection de points dominants. . . . .	75
11	Méthode sans paramètre pour la représentation polygonale. . . . .	90
12	Procédure <i>SPLIT</i> . . . . .	92
13	Approche multi-échelle pour décomposer une courbe discrète en parties significatives. . . . .	93
14	<i>RECONNAISSANCE_ARC</i> : Reconnaissance des arcs/cercles discrets . . . . .	112
15	<i>RECONNAISSANCE_ARC</i> : Reconnaissance des arcs/cercles discrets . . . . .	117
16	Segmentation d'une courbe en arcs discrets reposant sur l'algorithme 14 . . . . .	118
17	Polygonalisation basée sur la détection non-supervisée du bruit. . . . .	128
18	Segmentation d'une courbe discrète bruitée en arcs. . . . .	129
19	Décomposition d'une courbe en arcs et segments de droite . . . . .	136
20	Mesure de la circularité des courbes fermées. . . . .	146





# Introduction

De nos jours, l'image numérique apparaît beaucoup dans la vie quotidienne. Elle est un objet de recherche depuis plusieurs années. Plusieurs applications qui viennent de ces travaux de recherche jouent un rôle très important dans plusieurs aspects de l'économie et de la société tels que : vision robotique, vidéo-surveillance, imagerie médicale, imagerie aérienne et satellitaire, indexation d'images, . . . Une image numérique est souvent le résultat d'un processus d'acquisition (photographie numérique, imagerie médicale, image satellite, . . .). Elle se compose d'un ensemble de données discrètes qui sont des *pixels* en dimension 2 et *voxels* en dimension 3.

L'une des théories possibles pour l'exploitation des images numériques est la *géométrie discrète* qui s'est largement développée depuis quelques décennies. Cette branche de recherche se propose de définir et de traiter des ensembles de points à coordonnées entières qui formeront des objets géométriques tels que des droites discrètes ou bien des cercles discrets. L'objectif de ce domaine est de fournir des outils mathématiques pour le traitement d'images, l'analyse des images numériques qui sont représentées en ordinateur sous forme des données discrètes. Ce domaine fait partie de la communauté de reconnaissance des formes qui est soutenu par l'IAPR à travers le TC18 (comité technique 18 "Géométrie Discrète", <http://www.tc18.org>).

La géométrie discrète fait partie d'une famille de méthodes dédiées à l'analyse automatique des images numériques. Elle intervient fortement en *analyse d'images*. Une région d'une image peut être considérée comme un objet discret, un contour d'une image peut être considéré comme une courbe discrète. L'étude géométrique des objets discrets et des courbes discrètes mène directement à des applications en analyse d'images. Donc, la géométrie discrète se place naturellement comme un outil efficace pour la reconnaissance des formes, spécifiquement pour l'analyse d'images.

## Présentation du sujet de thèse

Ma thèse se situe dans le cadre du projet de l'ANR-GEODIB (Géométrie Discrète pour des Objets Discrets Bruités, <http://liris.cnrs.fr/geodib>). L'objectif de ce projet est d'étudier des objets discrets qui contiennent du bruit. Ils sont la conséquence de l'influence de la phase d'acquisition. L'étude des propriétés de ces objets donne des applications directes en traitement d'images. Nous utilisons la définition de la primitive de droite discrète proposée par Reveillès [Rev91]. Cette définition permet de décrire et de caractériser simplement et efficacement des objets discrets modélisés avec un bruit. Plusieurs objets flous sont construits à partir de la primitive *segment flou* qui a été introduite en 2003 [DRRRD03] et puis optimisée en 2005 [DRFRD06].

Cette thèse se trouve entre la *géométrie discrète* et l'*analyse d'images*. Nous nous intéressons aux objets discrets. L'objectif est de construire un cadre théorique pour transposer dans  $\mathbb{Z}^n$  des notions de base de la géométrie euclidienne. Spécifiquement, dans le cadre de cette thèse,

nous nous intéressons à l'étude des courbes discrètes. Puisque les contours des objets dans les images peuvent être représentés sous forme d'une liste de courbes discrètes, l'étude des courbes discrètes mène directement aux applications en analyse d'images. Ces deux aspects théoriques et pratiques apparaissent dans mes travaux de thèse. Le premier objectif de ma thèse est de définir des paramètres géométriques permettant de décrire et caractériser les courbes bruitées. Nous nous intéressons plus spécifiquement au paramètre de courbure discrète. Nous souhaitons proposer une définition de courbure adaptée aux courbes bruitées et construire les estimateurs de courbure ainsi que les optimiser. Le deuxième objectif est d'extraire des points d'intérêt d'une courbe discrète qui sont souvent utilisés pour la représentation des formes. Nous souhaitons proposer des méthodes de détection de points dominants et la représentation polygonales des courbes éventuellement bruitées. Le troisième objectif de la thèse est la reconnaissance des primitives discrètes non-linéaires telles que les arcs discrets, les cercles discrets,... Les arcs et les cercles sont des objets de base en géométrie discrète, ils apparaissent souvent aussi dans les images numériques. Cependant, pour lesquels il y a eu moins de travaux que pour les segments de droites dans la littérature. Nous souhaitons introduire des nouvelles méthodes pour la reconnaissance de ces primitives ainsi que la représentation des courbes discrètes reposant sur celles ci.

## Présentation des travaux

Dans un premier temps, nous étudions des propriétés locales des courbes discrètes : courbure pour les courbes 2D, courbure et torsion pour les courbes 3D. Nous proposons une amélioration de l'estimation de courbure en chaque point de la courbe reposant sur la décomposition de la courbe en une séquence de segments flous maximaux. Cette idée a été élargie afin d'extraire des propriétés locales des courbes 3D : courbure et torsion.

Nous nous intéressons ensuite aux applications de la courbure discrète. Le premier sujet d'application est la détection de points dominants et la représentation polygonale d'une courbe discrète. Un point dominant est un point dont la valeur de courbure est localement maximale. Les points dominants dans une courbe discrète jouent un rôle très important. Ce sont des points caractéristiques de la courbe qui contiennent des informations riches de celle ci. Ils servent souvent à l'indexation, la reconnaissance ou bien la reconstruction des objets. Nous avons développé une méthode efficace pour détecter les points dominants d'une courbe discrète en utilisant les propriétés de la suite des segments flous maximaux de la courbe discrète. Grâce à cette méthode, nous développons aussi des méthodes sans paramètres pour obtenir une représentation polygonale d'une courbe bruitée. Ces méthodes se basent sur une approche multi-épaisseur.

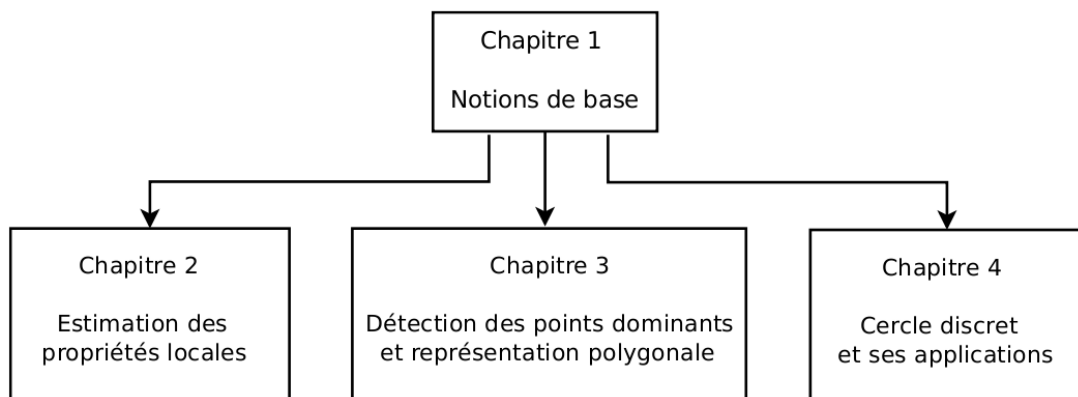
Les cercles et les arcs sont des objets dont la valeur de courbure est constante. C'est pourquoi, ces objets nous intéressent aussi. De plus, la reconnaissance de ces objets est aussi un problème fondamental en géométrie discrète. Ce problème est intéressant non seulement dans le cadre de la géométrie discrète mais aussi dans les autres domaines tels que l'analyse d'images ou bien la reconnaissance des formes car la détection des cercles dans les images est une étape importante dans des systèmes de détection et de reconnaissance d'objets. Dans cette thèse, nous proposons une approche originale et optimale pour la reconnaissance des arcs, cercles dans les courbes discrètes. Nous introduisons des propriétés importantes des arcs, cercles dans l'espace des tangentes introduit par Arkin [ACH<sup>+</sup>91] et Latecki [LL00]. Ces propriétés nous permettent d'obtenir une méthode linéaire de segmentation d'une courbe en arcs. Nous développons aussi une méthode permettant de décomposer une courbe en segments de droite et arcs. Cette décomposition nous permet de construire un descripteur de la courbes reposant sur les primitives arc et segment. Ceci ouvre une nouvelle piste pour la mise en correspondance des objets reposant sur ce descrip-

---

teur. Par ailleurs, nous proposons aussi dans cette thèse une méthode linéaire pour mesurer la circularité des objets discrets.

## Organisation du mémoire

Cette thèse est organisée comme suit. Le chapitre 1 présente des notions de base de la géométrie discrète. Dans le chapitre 2, nous proposons des estimateurs des propriétés locales des courbes 2D et 3D. Le chapitre 3 aborde des nouvelles méthodes de détection de points dominants et de représentation polygonale. Le dernier chapitre introduit les problèmes concernant les cercles discrets : la segmentation en arcs, la décomposition d'une courbe en arcs et segments de droite et la mesure de circularité des courbes fermée. Cette organisation est illustrée par la figure ci-dessous.





# Chapitre 1

## Notions de base

### Sommaire

---

<b>1.1</b>	<b>Introduction</b>	<b>5</b>
<b>1.2</b>	<b>Espace discret</b>	<b>5</b>
<b>1.3</b>	<b>Connexité</b>	<b>6</b>
<b>1.4</b>	<b>Codage</b>	<b>7</b>
<b>1.5</b>	<b>Discrétisation</b>	<b>9</b>
<b>1.6</b>	<b>Les primitives discrètes</b>	<b>9</b>
1.6.1	Droite discrète 2D	9
1.6.2	Droite discrète 3D	14
1.6.3	Segment flou	15
1.6.4	Etude de l'algorithme de reconnaissance d'un segment flou	19
1.6.5	Segment flou 3D	23
1.6.6	Cercle discret	26
1.6.7	Plan discret	29
1.6.8	Morceau flou de plan discret	30
1.6.9	Conclusion	31

---

### 1.1 Introduction

Dans ce chapitre, nous présentons les notions de base relatives à la géométrie discrète. Plusieurs définitions présentées ici sont extraites des livres [CM91, CMC07]. A la fin de ce chapitre, nous présentons les primitives discrètes importantes en géométrie discrète et utilisées dans ce mémoire.

### 1.2 Espace discret

La notion d'espace discret est construite sur la notion de pavage. Un *pavage* est un ensemble de cellules qui partitionnent le plan euclidien. La figure 1.1 illustre quelques pavages qui existent dans la vie quotidienne.

Un *espace discret* est considéré comme un pavage régulier du plan en dimension  $d$ . Un point discret est le centre de gravité de chaque cellule dans ce pavage. Le pavage régulier est construit à partir des polygones réguliers. Nous considérons souvent 3 manières principales de paver le plan euclidien reposant sur un seul polygone régulier.



(a) Une image de mosaïque

(b) Pavage d'une place

(c) Une image de mosaïque

FIGURE 1.1 – Exemples de pavages de la vie quotidienne.

- pavage par carrés
- pavage par triangles
- pavage par hexagones

La figure 1.2 présente différents types de pavages réguliers.

Le dual du pavage est le *maillage* qui est obtenu en reliant les points discrets qui représentent les cellules voisines. On peut considérer le maillage comme un graphe dont les sommets sont les points discrets, et dont les arrêtes présentent les adjacences des cellules du pavage. Nous considérons souvent en géométrie discrète les espaces discrets qui sont construits par des carrés pour le plan en dimension 2 ou bien des cubes multi-dimensionnels pour les dimensions supérieures. Ces espaces sont notés  $\mathbb{Z}^n$  pour la dimension  $n$ .

### 1.3 Connexité

Dans l'espace discret, la notion de *connexité* permet de construire des graphes d'adjacence. Nous considérons un espace discret 2D construit par des carrés. Il y a deux types de connexité. Le *4-voisinage* est défini par la relation d'adjacence par arrête, le *8-voisinage* est défini par la relation d'adjacence par sommet. Formellement, deux points  $p$  et  $q$  sont :

$$\begin{cases} 4\text{-voisins si } |x_p - x_q| + |y_p - y_q| = 1 \\ 8\text{-voisins si } \max(|x_p - x_q|, |y_p - y_q|) = 1 \end{cases}$$

La figure 1.3 présente deux types de voisinage : *4-voisinage* et *8-voisinage* en dimension 2.

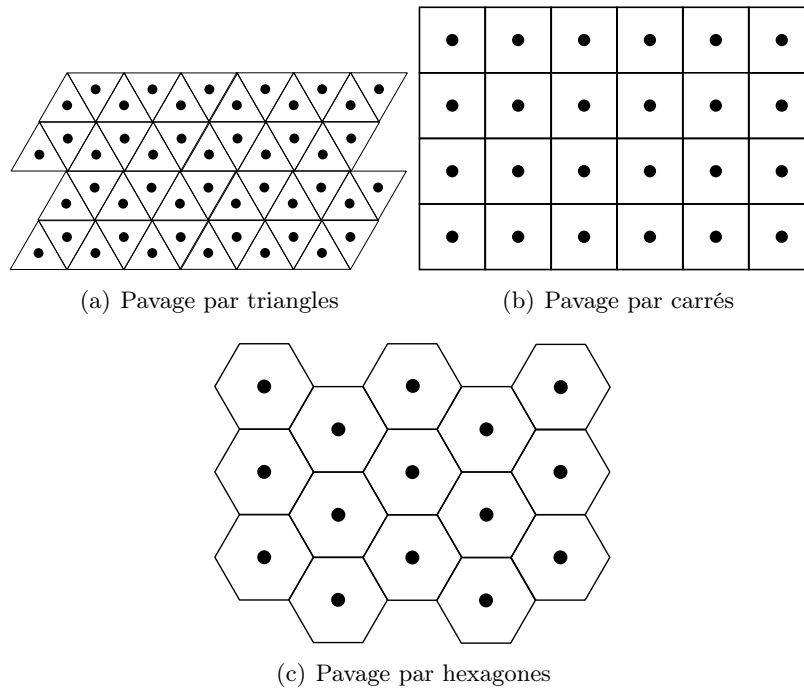


FIGURE 1.2 – Pavage par des polygones réguliers.

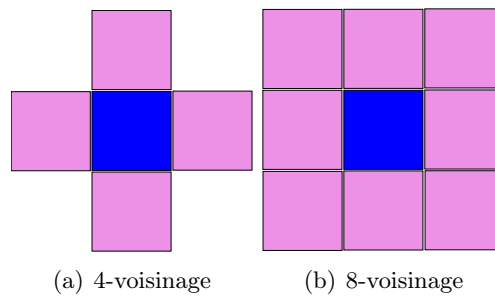


FIGURE 1.3 – Les voisinages en dimension 2.

De même, en dimension 3, nous avons 3 types de connexité : *6-voisinage*, *18-voisinage*, et *26-voisinage*. Ces voisinages sont définis comme suit. Deux points  $p$  et  $q$  sont :

$$\begin{cases} 6\text{-voisins} & \text{si } |x_p - x_q| + |y_p - y_q| + |z_p - z_q| = 1 \\ 18\text{-voisins} & \text{si } \max(|x_p - x_q|, |y_p - y_q|, |z_p - z_q|) = 1 \text{ et } |x_p - x_q| + |y_p - y_q| + |z_p - z_q| \leq 2 \\ 26\text{-voisins} & \text{si } \max(|x_p - x_q|, |y_p - y_q|, |z_p - z_q|) = 1 \end{cases}$$

La figure 1.4 présente 3 types de voisinage : *6-voisinage*, *18-voisinage* et *26-voisinage* en dimension 3.

## 1.4 Codage

Le codage est utilisé pour représenter des courbes dans un espace discret. En général, une courbe représentée doit avoir une propriété de connexité pour deux points consécutifs dans la

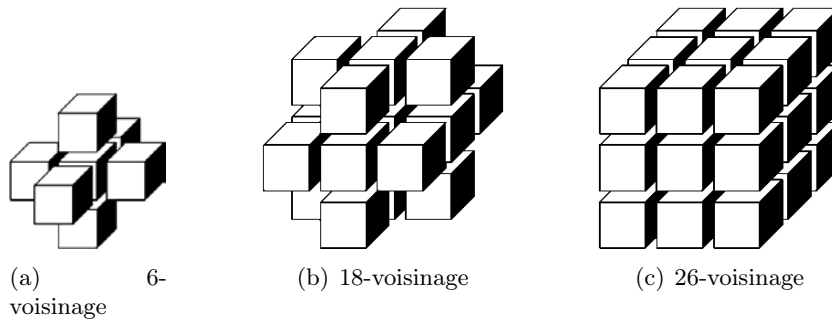


FIGURE 1.4 – Les voisinages en dimension 3.

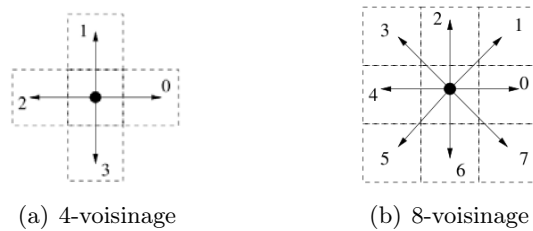


FIGURE 1.5 – Codage de Freeman [Fre61] dans un pavage par carrés.

courbe. Le codage usuel pour représenter des courbes discrètes est celui de Freeman [Fre61]. Ce codage utilise  $k$  labels pour coder des courbes qui satisfont la condition de connexité :  $k$ -voisinage. Ces courbes sont appelées des  $k$ -courbes. Dans un pavage par carré, il existe 2 types de connexité : 4-connexes et 8-connexes. C'est pourquoi, dans cet espace, il existe 2 codages de Freeman correspondant à ces types de connexité (voir la figure 1.5). La figure 1.6 montre plusieurs exemples de codage de Freeman définis pour deux courbes discrètes dans le pavage par carrés.

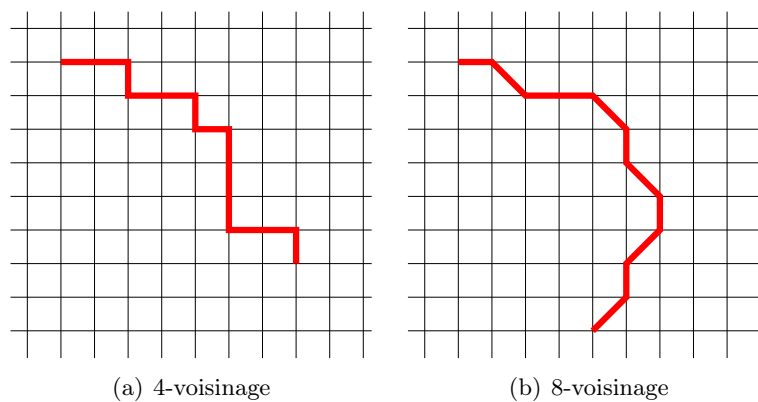


FIGURE 1.6 – Représentation des courbes discrètes par le codage de Freeman [Fre61] dans un pavage par carrés. a : 0030030333003 ; b : 07007676565.



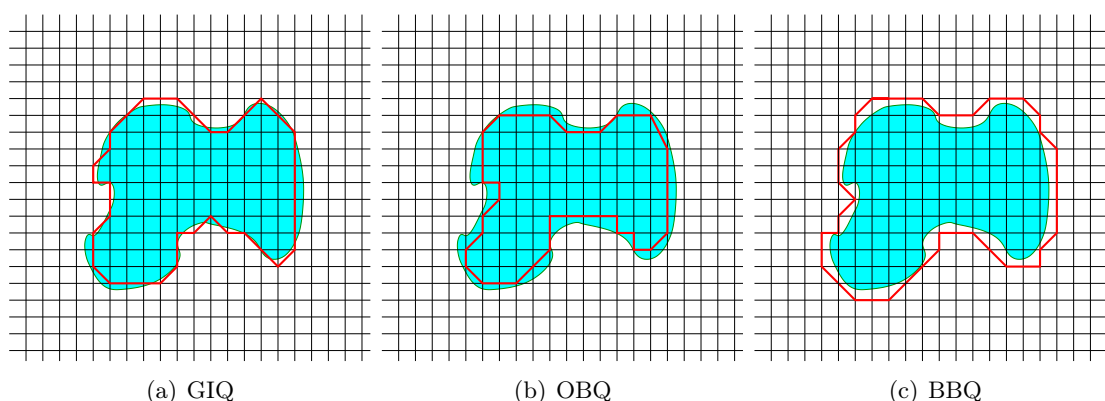


FIGURE 1.7 – Processus de discrétisation.

## 1.5 Discrétisation

Il existe plusieurs techniques pour transposer un objet réel vers l'espace discret. L'approche traditionnelle est de choisir le point discret le plus proche lorsque le contour croise une arête du maillage discret. Ce processus s'appelle *Grid Intersect Quantization* (GIQ). Ce processus est illustré dans la figure 1.7.a.

Si on considère la discrétisation d'un objet, nous pouvons définir les discrétisations interne et externe de l'objet. Dans ce cas, nous choisissons respectivement des points discrets à l'intérieur ou à l'extérieur de l'objet. Il y a deux processus qui correspondent à ces stratégies. Le premier processus s'appelle *Object Boundary Quantization* (OBQ), il correspond au choix des points discrets à l'intérieur de l'objet (voir la figure 1.7.b). Le deuxième processus s'appelle *Background Boundary Quantization* (BBQ) et correspond au choix des points discrets à l'extérieur de l'objet (voir la figure 1.7.c).

## 1.6 Les primitives discrètes

Dans cette section, nous présentons les objets de base de la géométrie discrète. Dans le cadre de notre travail, nous nous intéressons spécifiquement aux objets dans  $\mathbb{Z}^2$ .

Dans la première partie, nous rappelons la notion de droite discrète. Nous allons étudier de façon plus détaillée la définition de Reveillès [Rev91] que nous utiliserons fréquemment par la suite. Cette définition, qui est basée sur l'approche arithmétique, est un point de départ pour définir les notions de segments flous et segments flous maximaux qui seront présentées ultérieurement dans les paragraphes 1.6.3, 1.6.1. La notion de segments flous est proposée pour prendre en compte le bruit présent dans les contours d'objets discrets 2D. Un segment flou maximal est un segment flou qui ne peut pas être élargi ni à gauche ni à droite sur une courbe.

La partie 1.6.6 présentera la notion de cercle discret.

### 1.6.1 Droite discrète 2D

La notion de droite discrète est une notion essentielle en géométrie discrète. Elle constitue un objet de base pour de nombreux modèles discrets. Un algorithme de segmentation en segments de droite est considéré comme un outil de base pour estimer des mesures géométriques sur des objets discrets. Nous rappelons les définitions équivalentes pour caractériser une droite discrète.

- Les trois propriétés sur la chaîne de code de la courbe (Freeman)
- La propriété de la corde (Rosenfeld)
- L'absence de segments irréguliers (Hung)
- La double inégalité diophantienne pour les droites naïves (Reveillès)

### Les premières définitions de droite discrète 2D

*Les trois propriétés sur le code d'une courbe (Freeman) :* H. Freeman a proposé une méthode générale [Fre61] pour coder des courbes 8-connexes dans  $\mathbb{Z}^2$  (voir la figure 1.6).

En 1974, Freeman est le premier à avoir proposé une définition pour caractériser une droite discrète en se basant sur ce codage. Elle est présentée dans la définition ci-dessous (voir la figure 1.8).

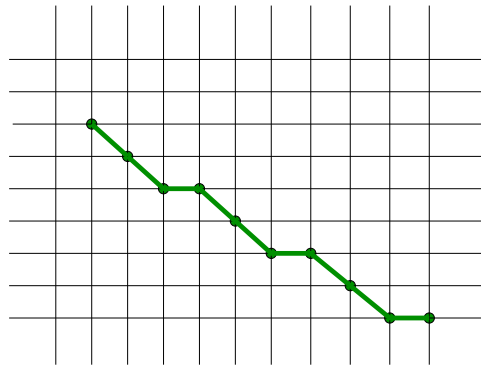


FIGURE 1.8 – Les 3 propriétés de Freeman. Le codage de la courbe est 770770770. Les codes "0" apparaissent isolés et uniformément répartis.

**Définition 1.6.1** *Une courbe discrète est un segment de droite si 3 propriétés ci-dessous sont satisfaites :*

- Son codage ne contient que deux codes différents et ceux-ci diffèrent d'un modulo 8.
- Une des deux valeurs apparaît toujours seule.
- Cet élément isolé est réparti aussi uniformément que possible

**La propriété de la corde (Rosenfeld) :** En 1974, Rosenfeld [Ros74b] a introduit la propriété de la corde permettant de caractériser un segment de droite. Cette propriété de la corde est présentée par la définition suivante (voir la figure 1.9).

**Définition 1.6.2** *Un ensemble des points  $\zeta$  vérifie la propriété de la corde si pour tout couple  $P$  et  $Q$  de  $\zeta$ , et pour tout point  $R(x, y)$  du segment  $PQ$ , il existe toujours un point  $M(i, j)$  de  $\zeta$  tel que*

$$\max(|i - x|, |j - y|) < 1$$

**L'absence de segments irréguliers (Hung) :** En 1984, Hung et al. [SHYH84] ont proposé une autre définition pour caractériser les segments de droite. Celle-ci est appelée l'absence des segments irréguliers. Les segments irréguliers sont présentés par la définition ci-dessous (voir la figure 1.10).

**Définition 1.6.3** *Deux segments d'égales longueurs dans la chaîne de code sont dits irréguliers quand les sommes des éléments de chacun de deux segments diffèrent de plus de 1 modulo 8.*

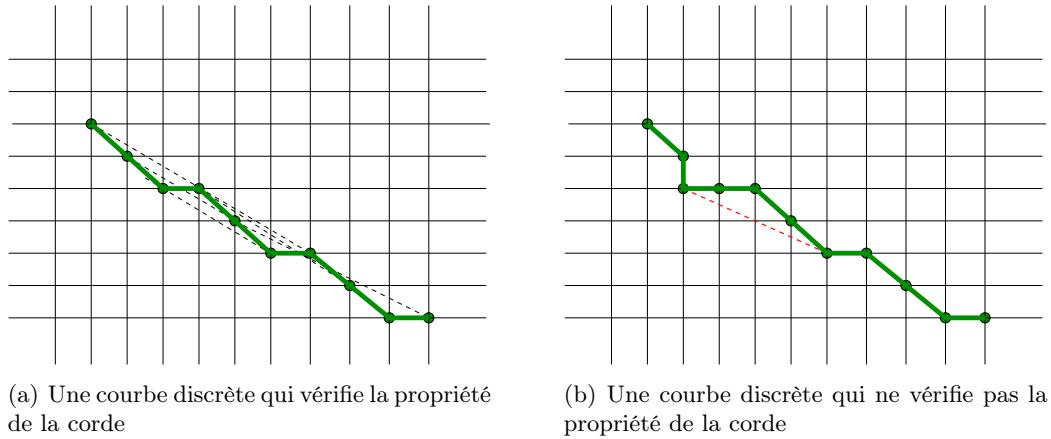


FIGURE 1.9 – Propriété de corde de Rosenfeld [Ros74b].

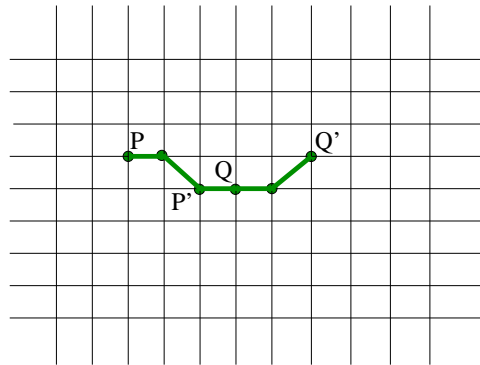


FIGURE 1.10 – Un exemple de segments irréguliers. Le code de Freman de cette courbe est 07001. Les deux segments PQ et P'Q' ont la même longueur. La somme des éléments de PQ est  $0+7+0=7$ , la somme des éléments de P'Q' est  $0+0+1=1$ . Celles-ci diffèrent de 2 modulo 8.

### L'approche arithmétique de Reveillès

En 1989, Reveillès [Rev91] a donné un nouveau point de vue en ce qui concerne la caractérisation d'un segment de droite. Il a proposé une caractérisation arithmétique qui est présentée par la définition suivante.

**Définition 1.6.4** Une droite discrète de pente  $\frac{a}{b}$ , de borne inférieure  $\mu$  et de d'épaisseur  $\omega$  notée par  $D(a, b, \mu, \omega)$ , avec  $a, b, \mu, \omega \in \mathbb{Z}$ , est composée des points  $(x, y) \in \mathbb{Z}^2$  vérifiant cette double inéquation :

$$\mu \leq ax - by < \mu + \omega$$

D'un point de vue géométrique, une droite discrète est un ensemble de points dans la grille qui se trouvent dans une bande bornée par 2 droites réelles.

$$\begin{cases} ax - by = \mu \\ ax - by = \mu + \omega - 1 \end{cases}$$

Reveillès a introduit également un théorème qui détermine la connexité des droites discrètes.

**Théorème 1.6.1** Supposons que  $D(a, b, \mu, \omega)$  est une droite discrète. Alors :

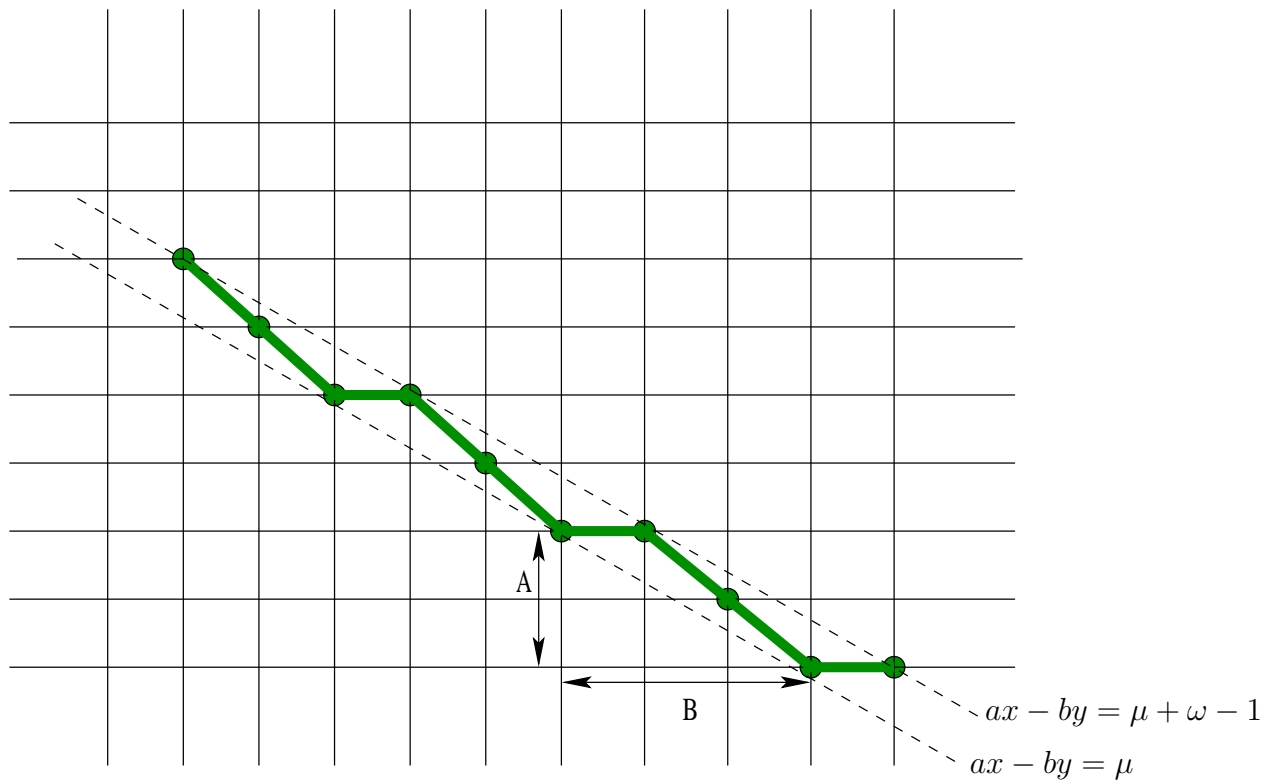


FIGURE 1.11 – Droite discrète arithmétique de Reveillès [Rev91].

- si  $\omega < \max(|a|, |b|)$  la droite est déconnectée.
- si  $\omega = \max(|a|, |b|)$  la droite est 8-connecte, on parle alors de droite naïve.
- si  $\max(|a|, |b|) < \omega < |a| + |b|$  la droite est \*-connecte, elle présente des 4-voisinages et 8-voisinages.
- si  $\omega = |a| + |b|$  la droite est strictement 4-connecte, on parle alors de droite standard.
- si  $|a| + |b| < \omega$ , on parle de droite épaisse.

Une droite naïve est une droite discrète 8-connecte dont l'épaisseur est la plus petite pour laquelle la droite est 8-connecte. Similairement, si  $\omega = |a| + |b|$ , la droite a l'épaisseur la plus petite parmi les droites 4-connectes. La figure 1.12 montre les différentes connexités des droites discrètes qui ont une même pente.

### Reconnaissance des segments de droite discrète naïve

Reposant sur la théorie de Reveillès, Debled-Rennesson [DR95] a proposé une méthode linéaire incrémentale pour reconnaître des segments de droite discrète naïve. Cette méthode se base sur l'approche arithmétique. L'ensemble des points entiers du premier octant appartenant à la droite discrète naïve  $D(a, b, \mu)$  avec  $0 \leq a < b$  et d'abscisses appartenant à l'intervalle  $[0, l]$  est appelé *segment de droite discrète naïve*, noté par  $\Sigma(a, b, \mu, l)$ . Les droites réelles  $ax - by = \mu$  et  $ax - by = \mu + b - 1$  sont appelées les *droites d'appui*. Un point entier de la droite  $ax - by = \mu$  (resp.  $ax - by = \mu + b - 1$ ) sera nommé *point d'appui supérieur* (resp. *inférieur*). Le *point d'appui principal inférieur* (resp. *supérieur*) de  $\Sigma$  est le point d'appui inférieur (resp. supérieur) possédant la plus petite abscisse.

Considérons un segment de droite discrète naïve  $\Sigma(a, b, \mu, l)$ , elle a étudié les conséquences

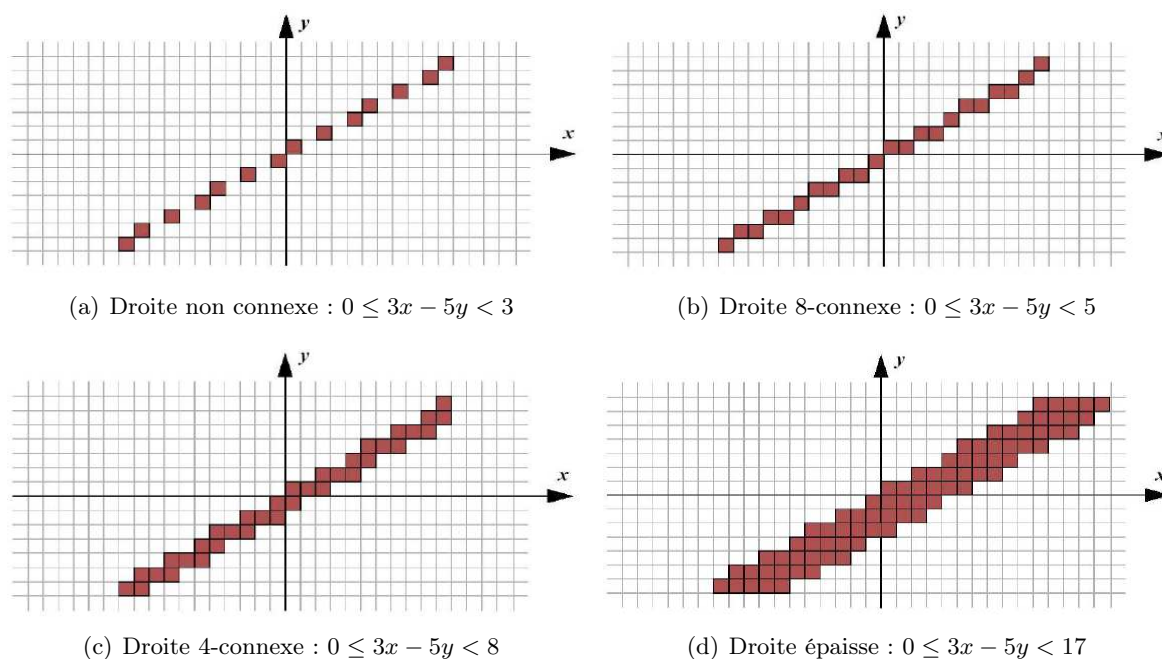


FIGURE 1.12 – Les droites ont des connexités différentes.

de l'ajout d'un point  $M$  au segment  $\Sigma$  et à quelle condition  $\Sigma \cup M$  est un segment de droite discrète. La méthode est introduite en se basant sur le théorème important ci-dessous.

**Théorème 1.6.2** *Considérons  $r(M)$ , le reste au point  $M(x,y)$  en fonction de  $D$  défini par :*

$$r(M) = ax - by$$

- ① Si  $\mu \leq r(M) < \mu + b$ , alors  $M \in D(a, b, \mu)$
- ② Si  $r(M) = \mu - 1$ ,  $\Sigma \cup M$  est un segment reconnu de la droite de pente donnée par le vecteur  $S_0M$  avec  $S_0$  point d'appui principal supérieur de  $D$
- ③ Si  $r(M) = \mu + b$ ,  $\Sigma \cup M$  est un segment reconnu de la droite de pente donnée par le vecteur  $I_0M$  avec  $I_0$  point d'appui principal inférieur de  $D$
- ④ Si  $r(M) < \mu - 1$  ou  $r(M) > \mu + b$ , alors  $\Sigma \cup M$  n'est pas un segment de droite discrète.

Cette approche a été élargie aussi pour la reconnaissance des segments de droite d'épaisseur quelconque.

### Segment maximal

La notion de segment maximal a été introduite par Feschet et Tougne [FT99]. Dans ce travail, ils se sont intéressés à la construction de l'ensemble des tangentes. Chaque tangente est définie comme une suite de points qui ne peut pas être élargie ni à gauche ni à droite et de telle façon que la nouvelle tangente soit encore un segment de droite. De notre point de vue, on peut considérer leur définition de tangente comme un segment maximal qui ne peut pas être élargi ni à gauche ni à droite sur la courbe étudiée. Dans leur travail, ils utilisent l'algorithme de reconnaissance d'une droite discrète naïve [DRR95] pour tester si une suite de points est encore un segment de droite. Nous rappelons ci-dessous leur définition d'un segment maximal (voir aussi la figure 1.13).

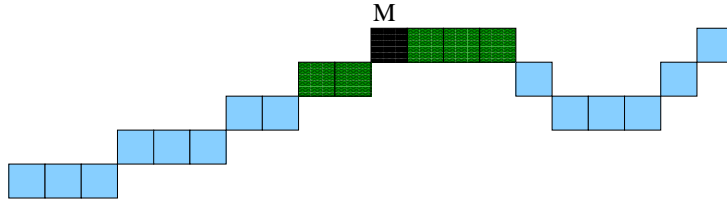


FIGURE 1.13 – Une tangente de Feschet et al. [FT99] en un point M. Elle est considérée comme un segment maximal de notre point de vue.

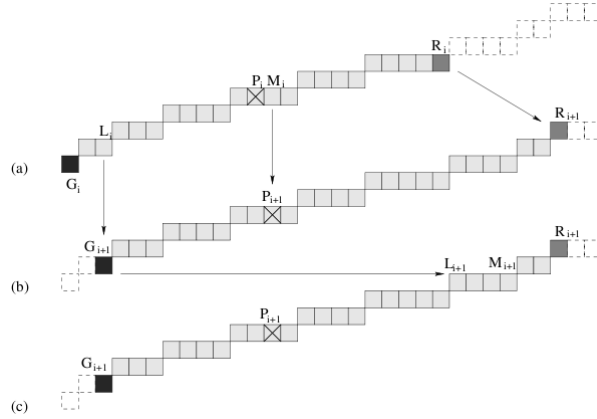


FIGURE 1.14 – Feschet et al. [FT99] : 2 étapes consécutives de calcul du segment maximal, le segment maximal  $L_{i+1}R_{i+1}$  est obtenu à partir du segment maximal  $L_i R_i$ .

**Définition 1.6.5** Soit  $\zeta = \{c_i\}_{i=0}^n$  une courbe discrète. Une suite de points est un segment de droite si il existe une droite discrète naïve  $D(a, b, \mu, \omega)$ , où  $\omega = \max(|a|, |b|)$  qui contient cet ensemble de points. Alors, une tangente en chaque point  $P = c_i$  est un segment  $\{c_{i-k}, \dots, c_i, \dots, c_{i+k+p}\}$  (resp.  $\{c_{i-k-q}, \dots, c_i, \dots, c_{i+k}\}$ ) est un segment tel que  $\{c_{i-k-1}, \dots, c_i, \dots, c_{i+k+1}\}$  et  $\{c_{i-k}, \dots, c_i, \dots, c_{i+k+p+1}\}$  (resp.  $\{c_{i-k-q-1}, \dots, c_i, \dots, c_{i+k}\}$ ) ne sont pas des segments ( $k, p \in \mathbb{Z}$ ).

Dans ce travail, Feschet et al. [FT99] ont proposé aussi une stratégie astucieuse qui permet de construire l'ensemble des segments maximaux d'une courbe discrète en temps linéaire. L'idée est de construire incrémentalement cet ensemble de segments maximaux. Cette stratégie se base sur la méthode de reconnaissance d'un segment de droite [DR95] en enlevant des points au début et ajoutant des point à la fin du segment de droite. C'est pourquoi, cette stratégie permet de réduire la complexité de construction de l'ensemble de segments maximaux en temps  $O(n)$  au lieu de  $O(n^2)$  si on utilise directement la méthode de Debled-Rennesson [DR95]. La figure 1.14 illustre cette stratégie en montrant comment construire le segment maximal suivant à partir d'un segment maximal.

Ce travail a été appliqué à une estimation optimale de la courbure en temps linéaire [FT99].

## 1.6.2 Droite discrète 3D

### Définition

Reposant sur la théorie de Reveillès, Debled [DR95] a introduit la définition de droite discrète 3D comme suit.

**Définition 1.6.6** Une droite discrète 3D, notée  $D$ , de vecteur directeur  $V(a, b, c)$ , avec  $(a, b, c) \in$

$\mathbb{Z}^3, a \leq b \leq c$ , est définie par l'ensemble des points  $(x, y, z)$  de  $\mathbb{Z}^3$  vérifiant les inéquations diophantiennes suivantes :

$$D = \begin{cases} \mu \leq cx - az < \mu + \omega \\ \mu' \leq bx - ay < \mu' + \omega' \end{cases} \quad (1.1)$$

avec  $\mu, \mu', \omega, \omega' \in \mathbb{Z}$ .

On appelle  $\mu, \mu'$  comme les bornes inférieures de  $D$ , de plus,  $\omega, \omega'$  sont appelées les épaisseurs arithmétiques de  $D$ . Une telle droite discrète est notée :  $D(a, b, c, \mu, \mu', \omega, \omega')$

De plus, Feschet et al. ont proposé aussi une autre approche pour définir une droite discrète  $nD$ . Les lecteurs peuvent consulter [FR06] pour avoir plus d'informations détaillées.

### Connexité d'une droite discrète

Similairement à la droite en 2D, le type d'une droite discrète en 3D dépend de la connexité des points de la droite déduite de son épaisseur arithmétique. Soit  $D(a, b, c, \mu, \mu', \omega, \omega')$  avec  $a > b > c$ . Alors,

- $\omega \geq a + c$  et  $\omega' \geq a + b$  :  $D$  est 6-connexe
- $\omega \geq a + c$  et  $a \leq \omega' \leq a + b$ ; ou  $\omega' \geq a + b$  et  $a \leq \omega \leq a + c$  :  $D$  est 18-connexe
- $a \leq \omega < a + c$  et  $a \leq \omega' < a + b$  :  $D$  est 26-connexe
- $\omega < a$  ou  $\omega' < a$  :  $D$  est déconnectée.

### 1.6.3 Segment flou

Dans le cadre de la géométrie discrète, plusieurs outils traditionnels [DR95, FT99] pour analyser les courbes discrètes ne s'adaptent pas au bruit présent dans ces courbes. Une courbe bruitée est considérée comme une courbe fortement courbure. A cause du bruit, l'algorithme de reconnaissance de segments de droites naïves produit des petits segments. Ceux-ci mènent à des résultats insatisfaisants lorsqu'on travaille avec des données bruitées (voir la figure 1.15).

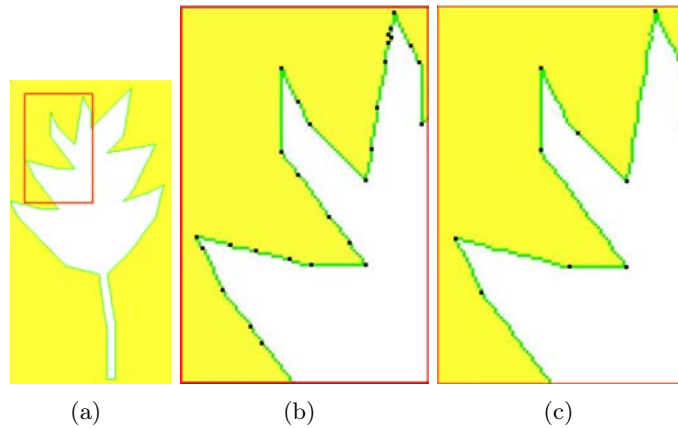


FIGURE 1.15 – Debled-Rennesson [DR07] : (a) Un exemple de courbe discrète obtenue dans un contour de l'image d'une feuille. (b) Un agrandissement sur une partie du contour et segmentation obtenue avec un algorithme de reconnaissance de segments de droites naïves. (c) La segmentation souhaitée qui présente mieux la forme géométrique sur la même partie de la courbe.

C'est pourquoi, Debled-Rennesson a introduit un nouvel objet discret : le segment flou [DRRRD03, DRFRD06] (voir aussi la figure 1.16). Cette notion repose sur la définition d'une

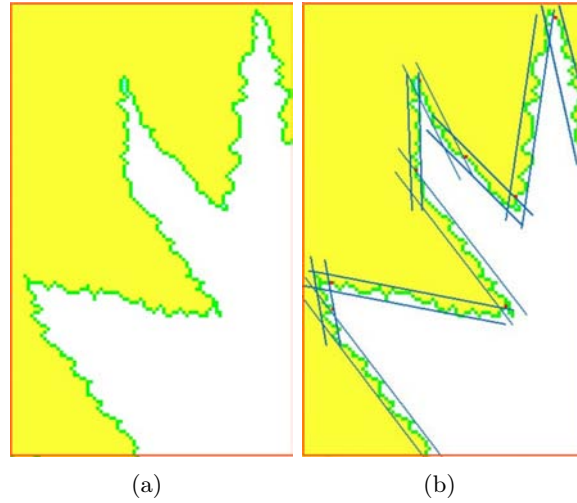


FIGURE 1.16 – Debled-Rennesson [DR07] : (a) Un exemple de courbe discrète fortement bruitée et (b) une segmentation souhaitée.

droite discrète d'épaisseur variable. Dans ces travaux, le segment flou est défini comme un ensemble de points qui appartient à une droite discrète dont l'épaisseur est fixée. Au début, cet ensemble est considéré comme une suite de points 8-voisins. Par la suite, elle a fait évoluer cette définition en supprimant la condition de connexité. Formellement, un segment flou arithmétique est défini par la définition ci-dessous.

**Définition 1.6.7** Soit  $S_f$  une suite de points de  $\mathbb{Z}^2$ . Une droite discrète  $D(a, b, \mu, \omega)$  est dite englobante pour  $S_f$  si tous les points de  $S_f$  appartiennent à  $D(a, b, \mu, \omega)$ .

Afin de segmenter une courbe discrète en segments flous, Debled-Rennesson a proposé 2 approches différentes : l'approche arithmétique [DRRRD03] et l'approche géométrique [DRFRD06].

### Segment flou d'ordre $d$

La première approche s'inspire de l'idée issue de la reconnaissance des segments de droites naïves [DR95]. Elle utilise la même notion de point d'appui. Un point d'appui est un point qui se trouve sur les droites réelles  $ax - by = \mu$  ou bien  $ax - by = \mu + \omega - 1$ . Grâce à cette notion, elle a proposé les définitions suivantes :

**Définition 1.6.8** Soit  $S_f$  une suite de points du premier octant définie dans l'intervalle des abscisses  $[0, l-1]$  et  $D(a, b, \mu, \omega)$  une droite discrète englobante.  $D$  est dite strictement englobante pour  $S_f$  si  $D$  contient au moins 3 points d'appui dans l'intervalle  $[0, l-1]$  et si  $S_f$  contient au moins un point d'appui inférieur et un point d'appui supérieur de  $D$ .

**Définition 1.6.9**  $S_f$  est un segment flou d'ordre  $d$  si et seulement si il existe une droite discrète  $D(a, b, \mu, \omega)$  strictement englobante pour  $S_f$  telle que  $\frac{\omega}{\max(|a|, |b|)} \leq d$

Debled-Rennesson et al. ont proposé la notion de segments flous d'ordre  $d$  [DRRRD03]. Un algorithme de reconnaissance d'un segment flou est proposé aussi reposant sur le calcul dans une configuration incrémentale. Supposons que  $\Sigma$  est un segment flou de la droite discrète avec des caractéristiques  $a, b, \mu, \omega$ . Si on ajoute un point  $M$  à ce segment flou :  $\Sigma' = \Sigma \cup M$ . Alors,



est ce que  $\Sigma'$  est un segment de même droite discrète support que  $\Sigma$  ou non, sinon est ce qu'il existe un segment de droite avec de nouvelles caractéristiques  $a', b', \mu', \omega'$ ? Il existe un théorème [DRRRD03] qui permet de déterminer, à partir d'un segment flou  $S_f$  et d'un point connecté à  $S_f$ , si  $S_f \cup M$  est un segment flou. Ce théorème travaille avec des segments flous d'ordre  $d$ .

**Théorème 1.6.3** *Considérons un segment flou  $S_f$  du premier octant dans l'intervalle des abscisses  $[0, l-1]$  et  $D(a, b, \mu, \omega)$ , une droite strictement englobante. Dans ce cas, l'ordre de  $S_f$  est  $\frac{w}{b}$ . Soit  $M(x_M, y_M)$  un point entier connecté à  $S_f$  ayant pour abscisse  $l$  ou  $l-1$ . Nous définissons le reste au point  $M$ , en fonction de  $D$ , noté  $r(M)$  par la valeur  $r(M) = ax_M - by_M$ .*

- ① Si  $\mu \leq r(M) < \mu + \omega$ ,  $M \in D$   
 $S_f \cup M$  est un segment flou d'ordre  $\frac{w}{b}$  avec  $D$  comme droite englobante.
- ② Si  $r(M) \leq \mu - 1$ ,  $M$  est extérieur à  $D$ ;  
 $S_f \cup M$  est un segment flou d'ordre  $\frac{w'}{b'}$  et la droite  $D'(a', b', \mu', \omega')$  est strictement englobante, avec
- $b'$  et  $a'$  les coordonnées du vecteur  $\overrightarrow{P_{r(M)+1}M}$ ,  $P_{r(M)+1}$  étant le point dont le reste en fonction de  $D$  est  $r(M) + 1$  et  $x_{P_{r(M)+1}} \in [0, b - 1]$ ,
  - $\mu' = a'x_M - b'y_M$
  - $\omega' = a'x_{L_L} - b'y_{L_L} - \mu' + 1$ , avec  $L_L(x_{L_L}, y_{L_L})$  le dernier point d'appui inférieur de la droite  $D$  présent dans  $S_f$
- De plus, on peut aussi calculer les vecteurs de déplacement de  $D'$  :

$$\overrightarrow{V'_{-1}} = (b - b', a' - a) - \left[\frac{b}{b'}\right](b', a') \quad \overrightarrow{V'_{+1}} = (b, a) + \left[\frac{b}{b'}\right](b', a')$$

- ③ Si  $r(M) \geq \mu + \omega$ ,  $M$  est extérieur à  $D$ ;  
 $S_f \cup M$  est un segment flou d'ordre  $\frac{w'}{b'}$  et la droite  $D'(a', b', \mu', \omega')$  est strictement englobante, avec
- $b'$  et  $a'$  les coordonnées du vecteur  $\overrightarrow{P_{r(M)-1}M}$ ,  $P_{r(M)-1}$  étant le point dont le reste en fonction de  $D$  est  $r(M) - 1$  et  $x_{P_{r(M)-1}} \in [0, b - 1]$ ,
  - $\mu' = a'x_{U_L} - b'y_{U_L} - \mu' + 1$ , avec  $U_L(x_{U_L}, y_{U_L})$  le dernier point d'appui supérieur de la droite  $D$  présent dans  $S_f$ ,
  - $\omega' = a'x_M - b'y_M - \mu' + 1$
- De plus, on peut aussi calculer les vecteurs de déplacement de  $D'$  :

$$\overrightarrow{V'_{-1}} = (b, a) - \left[\frac{b}{b'}\right](b', a') \quad \overrightarrow{V'_{+1}} = (b - b', a' - a) + \left[\frac{b}{b'}\right](b', a')$$

Ce théorème est la base de l'algorithme 1, donné par Debled-Rennesson et al. dans [DRRRD03]

Cette méthode incrémentale est utilisée pour construire la droite strictement englobante qui sert à segmenter une courbe discrète en segments flous pour un ordre donné. Elle est assez rapide grâce à sa complexité linéaire. Cependant, la construction n'assure pas que l'épaisseur de la droite créée soit minimale. La raison est que le taux minimal  $\frac{\omega}{\max(|a|, |b|)}$  n'est pas obtenu dans quelque cas (voir la section 2.3 du papier [DRFRD05]). En conséquence, la segmentation de la courbe discrète donne quelquefois trop de segments flous. D'autre part, cette méthode ne travaille qu'avec des courbes discrètes connectées.

**Algorithme 1** : Segmentation des courbes 8-connexes en segments flous.

**Entrée** :  $C$  une suite 8-connexes de points et  $d$  épaisseur pour les segments flous  
**Sortie** : La liste  $L$  des segments flous, chacun d'eux étant défini par son nombre de points  $nbPoint$  et les caractéristiques  $a, b, \mu, \omega$  d'une droite strictement englobante.  
**Initialisation** :  $a = 1; b = 0; w = b; nbPoint = 1; M_c = (0, 0); isSegment = true; end = false; isSameOctant = true; M =$  le premier point de  $C$ .

```

début
  pour  $!end$  faire
    pour  $isSegment$  et  $isSameOctant$  et  $!end$  faire
       $M_{last} = M$  ; testOctant(M) ;  $M_c =$  image de M dans le premier octant ;
       $a_{last} = a; b_{last} = b; \mu_{last} = a; \omega_{last} = a$  ; si  $isSameOctant$  alors
        addPointSf( $a, b, \mu, \omega, M_c$ ) ;  $isSegment = \frac{w}{b} \leq d$ 
      si  $isSegment$  alors  $nbPoint++$  ;
    fin
  fin
fin

```

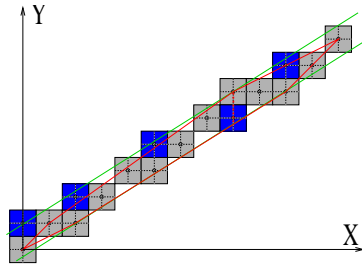


FIGURE 1.17 – Exemple de segment flou d'épaisseur 1 qui est englobé par la droite discrète  $D(5, 8, -8, 11)$ .

### Segment flou d'épaisseur $\nu$

La deuxième approche [DRFRD06] est une approche géométrique qui repose sur l'enveloppe convexe de l'ensemble des points considérés. En étudiant l'épaisseur verticale d'une droite discrète et la distance verticale de l'enveloppe convexe d'un ensemble des points, Debled-Rennesson a défini la notion de droite englobante optimale.

**Définition 1.6.10** Une droite englobante  $D(a, b, \mu, \omega)$  pour une suite de points  $S_f$  est dite optimale si son épaisseur verticale est égale à la distance verticale de l'enveloppe convexe de  $S_f$ .

Cette définition a conduit à la nouvelle notion de segment flou. Dans ce travail [DRFRD06], Debled-Rennesson a proposé la définition de segment flou d'épaisseur  $\nu$  qui est présentée comme suit.

**Définition 1.6.11** Un ensemble  $S_f$  de points entiers est un segment flou d'épaisseur  $\nu$  [DRFRD06] si il existe une droite discrète  $D(a, b, \mu, \omega)$  telle que tous les points de  $S_f$  appartiennent à  $D$  et  $\frac{\omega-1}{\max(|a|, |b|)} \leq \nu$

Un algorithme linéaire et incrémental a été proposé dans [DRFRD06] pour reconnaître si une suite de points est un segment flou d'épaisseur fixée. De plus, cet algorithme fournit les

caractéristiques de la droite englobante optimale de cette suite de points. D'une part, il repose sur le calcul incrémental et linéaire de l'enveloppe convexe du segment parcouru. D'autre part, il utilise les propriétés arithmétiques et géométriques des droites discrètes. La figure 1.18 représente l'ensemble de points reconnu comme un segment flou d'épaisseur 2 et  $\mathcal{D}(2, 7, -8, 11)$  est sa droite englobante optimale.

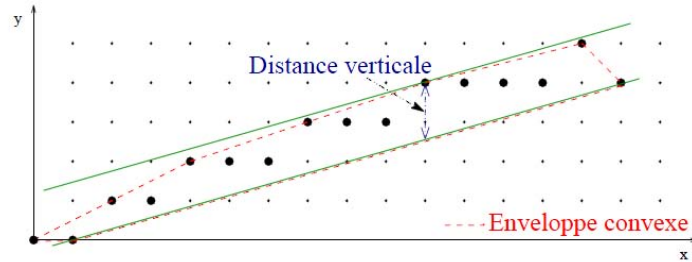


FIGURE 1.18 –  $\mathcal{D}(2, 7, -8, 11)$ , la droite englobante optimale de la suite de points (distance verticale =  $\frac{10}{7} = 1,42$ ).

Dans son travail, Debled-Rennesson suppose que la courbe discrète se trouve dans le premier octant et satisfait la condition x-croissant (l'abscisse du point inséré est incrémentée). Elle a prouvé que dans ce contexte la droite strictement englobante optimale pour un ensemble défini de points entiers  $S_f$  est la droite dont l'épaisseur correspond à la hauteur de l'enveloppe convexe de  $S_f$ . Alors, le problème est transféré vers le problème de détermination de la hauteur de l'enveloppe convexe d'un l'ensemble des points.

On note  $f(x)$  et  $g(x)$  les fonctions déterminées par la frontière en haut et en bas de l'enveloppe convexe de l'ensemble des points  $S_f$  à l'abscisse  $x$ . Donc, la fonction  $hauteur(x) = f(x) - g(x)$  représente la distance verticale de l'enveloppe convexe de  $S_f$  à l'abscisse  $x$ . Grâce à la propriété de concavité de la fonction  $hauteur(x)$ , on peut constater que la valeur maximale de  $hauteur(x)$  est obtenue dans 3 cas : sommet-sommet, sommet-arête, arête-arête. Ils sont appelés **les paires antipodales**. Il est évident que la hauteur de l'enveloppe convexe est obtenue sur une paire antipodale. Alors, pour un algorithme incrémental de construction d'une droite discrète, il faut montrer comment on construit la nouvelle paire antipodale à partir de l'ancienne paire antipodale lorsqu'un nouveau point est inséré. Dans le contexte de configuration incrémentale, la nouvelle paire antipodale est déterminée à partir de l'ancienne paire antipodale grâce à l'information de l'enveloppe convexe obtenue à chaque étape. Normalement, pour les courbes simples, on peut appliquer l'algorithme incrémental de Melkman [Mel87] pour déterminer l'enveloppe convexe à chaque étape. Ces idées de base sont contenus dans l'algorithme 2 proposé par Debled-Rennesson et al. dans [DRFRD06]. La complexité de cet algorithme est linéaire.

#### 1.6.4 Etude de l'algorithme de reconnaissance d'un segment flou

##### Introduction

Nous présentons dans cette section une étude de l'algorithme de reconnaissance d'un segment flou. Ce travail a été fait dans le cadre de mon stage de master de recherche [Ngu05].

Dans la section précédente, nous avons résumé les résultats principaux de la segmentation d'une courbe discrète en segments flous avec une épaisseur donnée. Le résultat le plus important est l'algorithme 2 proposé par Debled-Rennesson et al. [DRFRD06]. *Cet algorithme est optimal.*

---

**Algorithme 2** : Reconnaissance incrémentale d'un segment flou à l'épaisseur  $v$ .

---

**Entrée** :  $S$ , une séquence des points entiers 8-connexes,  $v$ , un nombre réel  
**Sortie** :  $isSegment$ , une variable booléenne;  $a, b, \mu, \omega$ , les nombres entiers  
**Initialisation** :  $isSegment = TRUE$ ;  $a = 0$ ;  $\omega = b = 1$ ;  $\mu = 0$ ;  $M$ =premier point de  $S$ ;  
**début**  
**pour**  $isSegment$  et  $S$  n'est pas balayé complètement **faire**  
     $M$ =point suivant de  $S$ ;  
    Mise à jour l'enveloppe convexe en utilisant l'algorithme de Melkman;  
     $r = ax_M - by_M$   
    **si**  $r = \mu$  **alors**  $U_L = M$ ;  
    **si**  $r = \mu + \omega - 1$  **alors**  $L_L = M$ ;  
    **si**  $r \leq \mu - 1$  **alors**  
         $U_L = M$ ;  
         $N$  le point précédent de  $M$  sur l'enveloppe convexe;  
         $a_0 = y_M - y_N$ ;  $b_0 = x_M - x_N$ ;  
         $a = \frac{a_0}{gcd(a_0, b_0)}$ ;  $b = \frac{b_0}{gcd(a_0, b_0)}$ ;  $\mu = ax_M - by_M$ ;  
        Trouver le premier point  $C$  sur l'enveloppe convexe en bas à partir de  $L_L$  tel  
        que :  $pen\acute{t}e[C, C_{next}] > \frac{a}{b}$ ;  
         $L_L = C$ ;  
    **sinon**  
        **si**  $r \geq \mu + \omega - 1$  **alors**  
            | Cas symétrique;  
        **fin**  
    **fin**  
     $isSegment = \frac{\omega - 1}{b} \leq v$ ;  
**fin**  
**fin**

---

C'est à dire l'épaisseur de la droite discrète englobante construite est minimale, alors les segments flous reconnus sont les plus longs possibles. L'objectif de notre travail est d'étudier l'algorithme 2 en 2D et de l'appliquer pour calculer la courbure et la tangente en chaque point d'une courbe représentant une séquence d'ADN. Cet algorithme suppose que la courbe discrète donnée en entrée se trouve dans *le premier octant* et satisfait la *condition x-croissant* (les abscisses des sommets de la courbe sont incrémentées). Puis, pour le cas général, chaque partie de la courbe est transformée pour satisfaire cette condition. Dans notre travail, nous développons cet algorithme en travaillant directement sans utiliser ces hypothèses. Le point positif de notre algorithme est qu'il peut fonctionner incrémentalement sur *des courbes simples non-connexes* en calculant la hauteur et la largeur de l'enveloppe convexe du segment flou. Il est présenté dans les parties suivantes.

### Analyse du problème

Dans [DRFRD06], Debled et al. ont montré que l'épaisseur verticale d'un segment flou peut être déterminée par une paire antipodale qui se compose d'un sommet et d'une arête. Alors, la reconnaissance incrémentale du segment flou se base sur la construction incrémentale de la paire antipodale lorsqu'on ajoute un point dont l'abscisse est incrémentée. Dans cette partie,

nous considérons la construction des paires antipodales à la fois verticales et horizontales. Nous avons la propriété ci-dessous.

**Propriété 1.6.1** *Soit  $S_f$ , un segment flou avec  $D(a, b, \mu, \omega)$  une droite discrète englobante optimale verticale (resp. horizontale) pour  $S_f$ .  $S'_f = S_f \cup \{M\}$ .  $C'$  est l'enveloppe convexe de  $S'_f$ . Supposons que  $M$  n'appartienne pas à  $D(a, b, \mu, \omega)$ . Soit  $M_1, M_2$  2 points voisins de  $M$  dans  $C'$ . Alors, la ligne qui contient la hauteur (resp. largeur) de  $S'_f$  va couper la droite  $[MM_1]$  ou  $[MM_2]$ .*

**Preuve 1.6.1** *La distance verticale de l'enveloppe convexe est une fonction concave. La valeur maximale est obtenue dans 2 cas : sommet-sommet ou arête-sommet (le cas arête-arête est un cas particulier de ces 2 cas). Donc, la hauteur de l'enveloppe convexe doit passer par un sommet  $A$ . L'autre extrémité de la hauteur est appelée le sommet  $B$ . Pour prouver cet énoncé, nous montrons que  $M_1$  et  $M_2$  ne se trouvent pas dans un même demi-plan dont le bord est la droite de la hauteur. Donc, si cet événement est prouvé,  $M_1$  et  $M_2$  doivent se trouver dans 2 demi-plans différents dont le bord est la droite de la hauteur ou bien un de deux points  $M_1$  et  $M_2$  se trouve sur la droite de la hauteur. Dans le premier cas, il existe un point  $M_1$  ou  $M_2$  qui ne se trouve pas dans même demi plan dont le bord est la droite de la hauteur avec le point  $M$  (grâce au théorème de Dirichlet : 3 lapins avec 2 cages). Supposons que c'est  $M_1$ , donc  $[MM_1]$  va couper la droite de la hauteur. Dans le deuxième cas, supposons que  $M_1$  se trouve sur la droite de la hauteur. Dans ce cas là,  $[MM_1]$  et la droite de la hauteur ont un même point commun qui est  $M_1$ , donc le segment  $[MM_1]$  va couper la droite de la hauteur.*

*Donc, pour obtenir ce résultat, je suppose que  $M_1$  et  $M_2$  se trouvent dans un même demi-plan dont le bord est la droite de la hauteur. Cet événement va mener à un contradiction sur l'hypothèse de la position de  $M$  et le rôle de  $M_1$  et  $M_2$  dans la nouvelle enveloppe convexe*

*Si  $M_1, M_2$  et  $M$  se trouvent sur le même demi-plan dont le bord est la hauteur de l'enveloppe convexe pour  $S_f$ , il est facile de déduire que  $A, M_1, M_2, M$  ne se trouvent pas sur  $C'$ . Si  $M_1, M_2$  se trouvent sur le même demi-plan qui ne contient pas  $M$ , il est évident que  $B$  ne se trouve pas dans l'angle  $\angle M_1MM_2$  (voir figure 1.19). Dans l'autre cas,  $M_1$  et  $M_2$  se trouve sur 2 demi-plans. Alors, la ligne qui contient la hauteur de l'enveloppe convexe pour  $S_f$  coupe  $[MM_1]$  ou  $[MM_2]$ . C'est pourquoi, la nouvelle hauteur pour  $C'$  va couper  $[MM_1]$  ou  $[MM_2]$  grâce la propriété de concavité de la fonction de la distance verticale.*

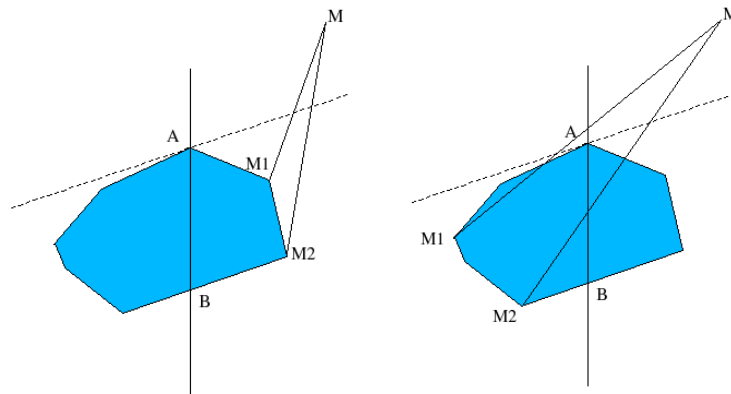
*Le cas horizontal est similaire.*

Nous voulons construire un algorithme qui travaille avec un segment flou dans le cas général. Notre idée de base est de construire les droites englobantes optimales verticale et horizontale à chaque étape incrémentale. Grâce à l'énoncé 1.6.1, nous pouvons proposer ci-dessous une stratégie incrémentale pour construire des droites englobantes optimales horizontale et verticale.

- *Construction de la droite englobante optimale horizontale*

Supposons que nous connaissons la largeur de l'enveloppe convexe de  $S_f$ , cette largeur est obtenue dans deux cas possibles : sommet-sommet (Z-X), sommet-arête(Z-XY) (cas arête-arête peut être considéré un cas spécial de ces cas). Ces cas sont illustrés dans la figure 3.3

On peut considérer le premier cas comme un cas particulier du deuxième cas. Le problème introduit est de déterminer la largeur de l'enveloppe convexe de  $S'_f$  si on ajoute  $M$  à  $S_f$ , ( $S'_f = S_f \cup M$ ). Supposons que les coefficients d'arête XY sont  $a, b$  ( $a > 0$ ),  $r(x, y) = ax - by$ ,  $r(T) = \max(r(X), r(Z))$ ,  $r(S) = \min(r(X), r(Z))$ . Nous constatons que si  $r(M) \leq r(T)$  et  $r(M) \geq r(S)$ , la largeur de l'enveloppe convexe de  $S'_f$  ne change pas. Grâce à la symétrie, nous ne considérons pas le cas  $r(M) > r(T)$ . Appelons  $M_h, M_l$  les points voisins précédent et suivant de  $M$  dans l'enveloppe convexe de  $S'_f$  ( $M_h.Y > M_l.Y$ ). Il est évident que si  $M.Y > Z.Y$ , la nouvelle largeur

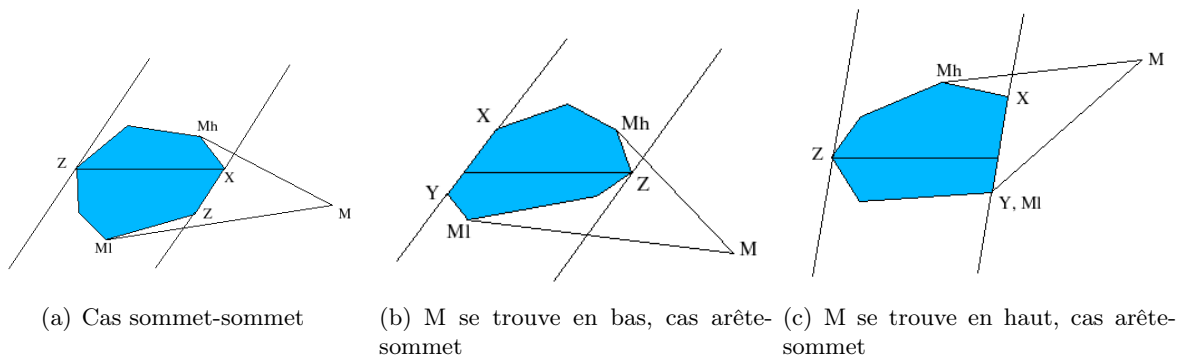


(a)  $M, M_1$  et  $M_2$  se trouvent sur le même demi-plan bord AB (b)  $M, M_1$  et  $M_2$  ne se trouvent pas sur le même demi-plan bord AB

FIGURE 1.19 – On considère deux cas où les points voisins de  $M$  ( $M_1$  et  $M_2$ ) sur l’enveloppe convexe se trouvent sur le même demi-plan dont le bord est la hauteur de l’enveloppe convexe pour  $S_f$ .

va couper l’arête  $MM_l$ , sinon c’est l’arête  $MM_h$ . L’autre point d’appui est déterminé à partir de  $C$  ( $C$  est le point d’appui dans la précédente étape, il est  $X$  si  $r(X) < r(Z)$ , sinon il est  $Z$ ) après quelques tests.

- Construction de la droite englobante optimale verticale



(a) Cas sommet-sommet

(b)  $M$  se trouve en bas, cas arête-sommet

(c)  $M$  se trouve en haut, cas arête-sommet

FIGURE 1.20 – Calcul de la largeur.

Similairement à la construction de la droite englobante optimale horizontale, les cas possibles sont illustrés dans la figure 1.21. Supposons que  $r(M) > r(X)$ , et  $r(M) > r(Z)$  (les autres cas sont considérés similairement). Appelons  $M_g, M_d$  les points précédent et suivant de  $M$  dans l’enveloppe convexe de  $S_f'$  ( $M_d.X > M_g.X$ ). Il est évident que si  $M.X > Z.X$ , la nouvelle largeur va couper l’arête  $MM_g$ , sinon c’est l’arête  $MM_d$ . L’autre point d’appui est déterminé à partir de  $C$  ( $C$  est le point d’appui dans la précédente étape, il est  $X$  si  $r(X) < r(Z)$ , sinon il est  $Z$ ) après quelques tests.

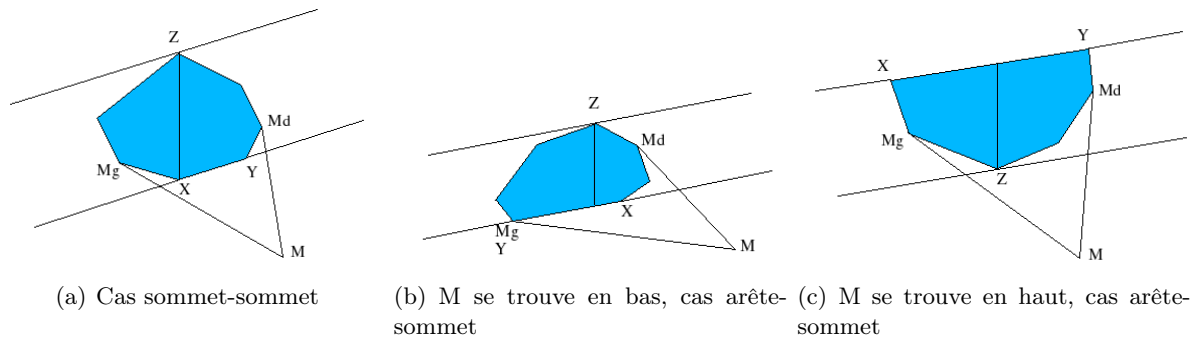


FIGURE 1.21 – Calcul de la hauteur.

## Conception des algorithmes

### Algorithmes

Nous proposons ci-après deux algorithmes (algorithme 4, algorithme 3) pour calculer la largeur et la hauteur de l'enveloppe convexe à chaque étape.

Grâce aux algorithmes 3 et 4, nous proposons l'algorithme 5 pour construire la droite englobante optimale pour un segment flou. Ce segment flou se trouve dans le premier octant et satisfait la condition de x-croissant. L'idée de base est qu'on calcule parallèlement la largeur et la hauteur du segment flou à chaque étape. Si  $d_V > d_H$ , on change la direction principale de la droite englobante.

### Complexité

Pour déterminer l'enveloppe convexe à chaque étape, nous utilisons la version incrémentale de l'algorithme de Melkman [Mel87]. A chaque étape, le coût du recalcul de la largeur et de la hauteur est constant. Donc, la complexité de l'algorithme de reconnaissance d'un segment flou optimal est linéaire. Donc, si la courbe d'entrée est simple, en appliquant cet algorithme, nous pouvons reconnaître des segments flous de la courbe discrète avec l'épaisseur  $\nu$  en temps linéaire.

## Extension de segment flou dans l'espace réel

Notre algorithme de reconnaissance d'un segment flou présenté ci-dessous n'utilise pas l'hypothèse de la connexité d'une courbe discrète. De plus, la reconnaissance repose sur le calcul de l'enveloppe convexe d'un ensemble de points qui est faisable dans l'espace réel. Par conséquent, nous pouvons appliquer cet algorithme pour la reconnaissance d'un segment flou qui est étendu dans l'espace réel. Puisque la notion de droite discrète n'existe plus dans cet espace, nous introduisons une définition de segment flou reposant sur l'hauteur de son enveloppe convexe.

**Définition 1.6.12** *Un ensemble  $S_f$  de points réels est d'un segment flou d'épaisseur  $\nu$  si l'hauteur de l'enveloppe convexe de cet ensemble de points est inférieure ou égale à  $\nu$*

### 1.6.5 Segment flou 3D

Déduite de ce travail [DRFRD06], nous pouvons avoir une notion de segment flou 3D (voir la figure 1.22) comme suit.

**Définition 1.6.13** *Un ensemble  $S_{f_{3D}}$  de points entiers est un segment flou 3D d'épaisseur  $\nu$  avec le vecteur principal  $(a, b, c)$  vérifiant  $(a, b, c) \in \mathbb{Z}^3$  et  $a \geq b \geq c$  si il existe une droite discrète 3D  $D(a, b, \mu, \mu', e, e')$  telle que*

**Algorithme 3** : Calcul de la hauteur à chaque étape.

---

**Entrée** :  $S_f$ - un segment flou,  $M$ - un point ajouté,  $v$  -une épaisseur donnée  
**Sortie** : Les caractéristiques de la droite englobante verticale pour le nouveau segment flou  $S'_f$

**Initialisation** :  $S'_f = S_f \cup M$

**début**

Déterminer  $M_G, M_D$  - les deux points voisins de  $M$  dans l'enveloppe convexe de  $S'_f$  ;

**si** arête-sommet **alors**

| Déterminer les coefficients  $a, b$  ( $a > 0$ ) de l'arête  $XY$  ;  $r(x, y) = ax - by$  ;

**sinon**

| **si**  $slope(XX_{precedent}) > slope(ZZ_{precedent})$  **alors**

| |  $a, b$  ( $a > 0$ ) correspondent à  $XX_{precedent}$  ;  $r(x, y) = ax - by$  ;

| **sinon**

| |  $a, b$  ( $a > 0$ ) correspondent à  $ZZ_{precedent}$  ;  $r(x, y) = ax - by$  ;

| **fin**

**fin**

**si**  $(r(M) - r(X)) * (r(M) - r(Z)) < 0$  **alors**

| Largeur ne change pas ; **return** ;

**fin**

**si**  $(r(M) < r(X))$  et  $(r(M) < r(Z))$  **alors**

| **si**  $M_X \geq Z_X$  **alors**

| | **si**  $r(X) < r(Z)$  **alors**

| | |  $C = Z$  ;

| | **sinon**

| | |  $C = X$  ;

| | **fin**

| | **pour**  $slope(CC_{suivant}) < slope(M_G M)$  **faire**

| | |  $C' = C$  ;  $C = C_{suivant}$  ;

| | **fin**

| | **si**  $C_Y < M_Y$  **alors**

| | |  $M_G M, C$  : arête-sommet de la hauteur pour  $S'_f$  ;

| | **sinon**

| | | **si**  $C_Y > M_Y$  **alors**

| | | |  $C' C, M$  : arête-sommet de la hauteur pour  $S'_f$  ;

| | | **sinon**

| | | |  $C, M$  : arête-arête de la hauteur pour  $S'_f$  ;

| | | **fin**

| | **fin**

| **sinon**

| | Symétrie du précédent cas ;

| **fin**

**sinon**

| Similarité du précédent cas ;

**fin**

Déterminer la hauteur entre sommet-sommet ou sommet-arête ; **return** ;

**fin**

---



---

**Algorithme 4** : Calcul de la largeur à chaque étape.
 

---

**Entrée** :  $S_f$ - un segment flou,  $M$ - un point ajouté,  $v$  -une épaisseur donnée  
**Sortie** : Les caractéristiques de la droite englobante horizontale pour le nouveau segment flou  $S'_f$   
**Initialisation** :  $S'_f = S_f \cup M$   
**début**

Déterminer  $M_H, M_L$  - les deux points voisins de  $M$  dans l'enveloppe convexe de  $S'_f$  ;  
**si** *arête-sommet* **alors**  
 | Déterminer les coefficients  $a, b$  ( $a > 0$ ) de l'arête  $XY$  ;  $r(x, y) = ax - by$  ;  
**fin**  
**sinon**  
 | **si**  $slope(XX_{precedent}) > slope(ZZ_{precedent})$  **alors**  
 | |  $a, b$  correspondent à  $XX_{precedent}$  ;  $r(x, y) = ax - by$  ;  
 | **fin**  
 | **sinon**  
 | |  $a, b$  correspondent à  $ZZ_{precedent}$  ;  $r(x, y) = ax - by$  ;  
 | **fin**  
**fin**  
**si**  $(r(M) - r(X)) * (r(M) - r(Z)) < 0$  **alors** Largeur ne change pas ; **return** ;  
**si**  $(r(M) > r(X))$  et  $(r(M) > r(Z))$  **alors**  
 | **si**  $M_Y \geq Z_Y$  **alors**  
 | | **si**  $r(X) < r(Z)$  **alors**  $C = X$  ; **sinon**  $C = Z$  ;  
 | | **pour**  $slope(CC_{precedent}) > slope(MM_L)$  **faire**  
 | | |  $C' = C$  ;  $C = C_{precedent}$  ;  
 | | **fin**  
 | | **si**  $C_Y < M_Y$  **alors**  
 | | |  $M_L M, C$  : arête-sommet de la largeur pour  $S'_f$  ;  
 | | **fin**  
 | | **sinon**  
 | | | **si**  $C_Y > M_Y$  **alors**  
 | | | |  $C' C, M$  : arête-sommet de la largeur pour  $S'_f$  ;  
 | | | **fin**  
 | | | **sinon**  
 | | | |  $C, M$  : arête-arête de la largeur pour  $S'_f$  ;  
 | | | **fin**  
 | | **fin**  
 | **fin**  
 | **sinon**  
 | | Symétrie du précédent cas ;  
 | **fin**  
**fin**  
**sinon**  
 | Similarité du précédent cas ;  
**fin**  
 Calculer la largeur entre sommet-sommet ou sommet-arête obtenu ; **return** ;  
**fin**

---

---

**Algorithme 5** : Construction de la droite englobante optimale pour un segment flou.

---

**Input** : S : une courbe discrète  
**Output** :  $D(a, b, \mu, \omega)$  la droite englobante optimale construite  
**Initialization** :  $isSegment = TRUE$  ;  
**début**  
  **pour** S n'est pas encore complètement balayé et  $isSegment$  **faire**  
    M : point suivant de S ; Ajouter M à  $S_f$  ;  $S_f' = S_f \cup M$  ;  
    Déterminer l'enveloppe convexe ;  
    Calculer la hauteur de  $S_f'$  ;  
    Calculer la largeur de  $S_f'$  ;  
    **si**  $d_H \geq d_V$  **alors**  
      |  $D(a, b, \mu, \omega)$  est la droite qui correspond à la hauteur pour  $S_f'$  ;  
    **sinon**  
      |  $D(a, b, \mu, \omega)$  est la droite qui correspond à la largeur pour  $S_f'$  ;  
    **fin**  
    **si**  $\frac{\omega-1}{\max(|a|, |b|)} > d$  **alors**  
      |  $isSegment = FALSE$  ;  
    **fin**  
  **fin**  
**fin**

---

- $D(a, b, \mu', e')$  est la droite optimale pour la séquence des points de projection de  $Sf_{3D}$  sur le plan  $(0, x, y)$  et  $\frac{e'-1}{\max(|a|, |b|)} \leq \nu$
- $D(a, c, \mu, e)$  est la droite optimale pour la séquence des points de projection de  $Sf_{3D}$  sur le plan  $(0, x, z)$  et  $\frac{e-1}{\max(|a|, |c|)} \leq \nu$

Similairement au cas 2D, une notion étendue du segment flou 3D peut être proposée dans l'espace réel.

### 1.6.6 Cercle discret

Comme les droites discrètes, les cercles discrets sont aussi des objets de base de la géométrie discrète. Il y a deux notions qui sont très proches. Un disque discret correspond à une région intérieure d'un cercle, un cercle discret correspond au contour d'un cercle. Dans cette section, nous présentons les définitions existantes dans la littérature en ce qui concerne deux types d'objet : disque discret et cercle discret. Il y a une dualité entre deux notions.

#### Disque discret

**Disque discret de Kim [Kim84a]** : Kim a proposé une définition du disque discret (voir la figure 1.23). Considérons S - un ensemble des points dans  $\mathbb{Z}^2$ ,  $\bar{S}$  est le complément de S dans  $\mathbb{Z}^2$ . Kim a introduit les définitions ci-dessous.

**Définition 1.6.14** Une région discrète est un ensemble fini de points 4-connexes dans  $\mathbb{Z}^2$ .

**Définition 1.6.15** Image discrète d'une région : considérons une région P. Un ensemble S est une image de P si

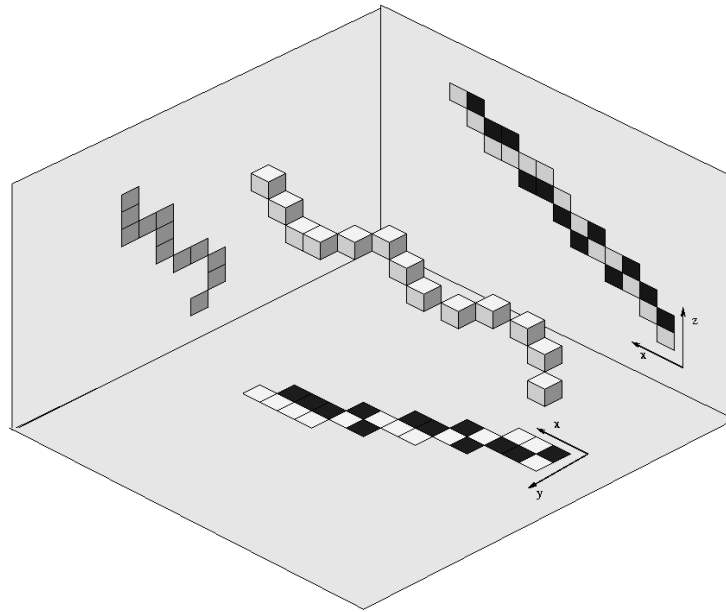


FIGURE 1.22 – Debled-Rennesson [DR95] : Un segment flou 3D.

- $S \subseteq P$  et,
- $\overline{S} \cap P = \phi$

**Définition 1.6.16** Une région discrète  $Q$  est un **disque discret** si il existe un disque  $q$  dont l'image discrète est  $Q$ .

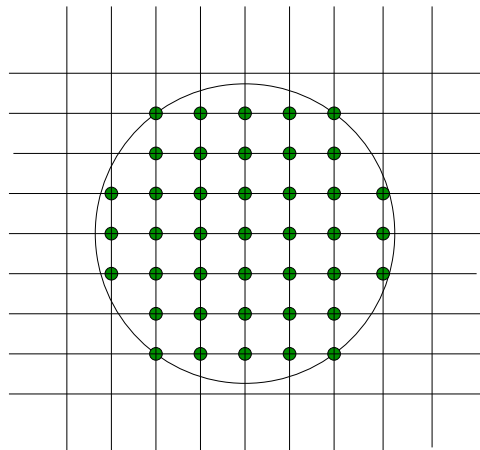


FIGURE 1.23 – Un disque discret de Kim [Kim84a].

### Cercle discret

*Cercle discret de Nakamura* : De même que Kim, Nakamura [NA84] a proposé une autre définition de disque discret qui est en fait la discrétisation de type GIQ d'un cercle discret. Dans cette discrétisation, la discrétisation d'un cercle repose sur les points entiers qui sont les plus proches du cercle réel (voir la figure 1.24).

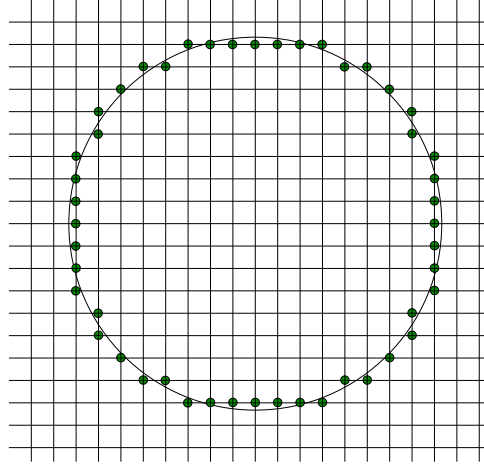


FIGURE 1.24 – Cercle discret de Nakamura basé sur la discrétisation de type GIQ.

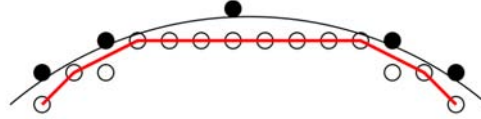


FIGURE 1.25 – Polygone quasi circulaire de Coeurjolly [CGRT04].

*Polygone quasi-circulaire de Coeurjolly* : Coeurjolly a proposé une définition de courbes quasi-circulaires [CGRT04] reposant sur le problème de séparation des points par des arcs circulaires (voir la figure 1.25).

**Définition 1.6.17** Une polyligne convexe  $\Gamma = \{v_i\}_{i=1,\dots,n}$  est quasi-circulaire si on peut séparer par un cercle tous les points de la grille qui se trouvent dans l'enveloppe convexe de  $\Gamma$  et tous les points extérieurs de l'enveloppe convexe.

*Cercle analytique discret de Andres* : En s'inspirant de la définition des droites discrètes de Reveillès, Andres a proposé une définition des cercles analytiques discrets [And94a] (voir aussi la figure 1.26). Cette définition permet de vérifier si un point discret appartient à un cercle discret ou non. Un cercle est considéré comme un anneau qui dépend du paramètre  $\omega \in \mathbb{R}^+$  qui est appelé l'épaisseur arithmétique. Nous présentons ci-dessous cette notion par la définition suivante.

**Définition 1.6.18** Le cercle analytique discret  $\mathbb{C}(M, R, \omega)$  de centre  $M = (x_0, y_0) \in \mathbb{R}^2$ , de rayon  $R \in \mathbb{R}^+$ , et d'épaisseur arithmétique  $\omega \in \mathbb{R}^+$  est un ensemble de points dans  $\mathbb{Z}^2$  qui est défini comme suit.

$$\mathbb{C}(M_0, R, \omega) = \{(i, j) \in \mathbb{Z}^2 \mid (R - \frac{\omega}{2})^2 \leq (i - x_0)^2 + (j - y_0)^2 < (R + \frac{\omega}{2})^2\}$$

**Définition 1.6.19** Supposons que  $\mathbb{C}(M, R, \omega)$  est un cercle analytique discret. Si  $\omega = 1$ ,  $\mathbb{C}(M, R, \omega)$  est un cercle régulier.

Les cercles réguliers correspondent aux droites naïves dans la définition de Reveillès.

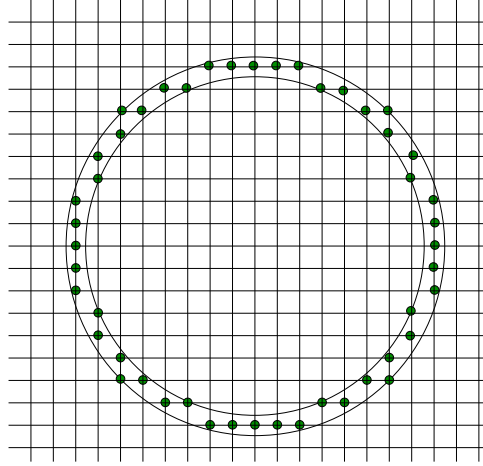


FIGURE 1.26 – Cercle analytique discret de Andres [And94a].

### 1.6.7 Plan discret

Dans la littérature, il existe plusieurs définitions du plan discret. Généralement, ces définitions viennent du processus de discrétisation du plan euclidien. Dans cette section, nous présentons les différentes définitions existantes.

#### Propriété du triangle corde

S'inspirant de la propriété de la corde de Rosenfeld [Ros74b] pour le cas 2D, Kim et Rosenfeld [KR82] ont proposé une notion similaire pour le cas 3D. Nous présentons ci-dessous la propriété du triangle corde.

**Définition 1.6.20** *Un ensemble de voxels  $E$  vérifie la propriété du triangle corde si pour tout triangle  $T$  formé de  $u$ ,  $v$  et  $w$  dans  $E$ , pour tout point  $(x, y, z)$  de  $T$ , il existe un point  $p$  de  $E$  tel que :  $\max\{|p_x - x|, |p_y - y|, |p_z - z|\} < 1$*

Kim [Kim84b] a montré que cette propriété permet de caractériser un plan discret infini et sans bord. Pourtant, celle-ci n'est pas appropriée dans le cas d'un morceau de plan discret. Dans le théorème ci-dessous, il a proposé aussi un lien qui est présenté entre les morceaux de plan discret et des propriétés sur leur enveloppe convexe.

**Théorème 1.6.4** *Un ensemble des voxels  $E$  est un morceau de plan si et seulement si il existe un plan  $P$  s'appuyant sur une face de l'enveloppe convexe de  $E$  tel que :*

$$d_H(E, P) < 1$$

avec  $d_H$  la distance de Hausdorff [Fed96] adaptée à la grille :

$$d_H(E, P) = \min\{\max_{v \in E}(d_x(v, P)), \max_{v \in E}(d_y(v, P)), \max_{v \in E}(d_z(v, P))\}$$

où  $d_x = x - x'$ ,  $d_y = y - y'$ ,  $d_z = z - z'$  avec  $(x', y, z)$ ,  $(x, y', z)$ ,  $(x, y, z')$  sont des points réels sur  $P$

Pourtant, ce théorème n'est pas complet. Debled a montré dans sa thèse [DR95] un contre-exemple de ce théorème.

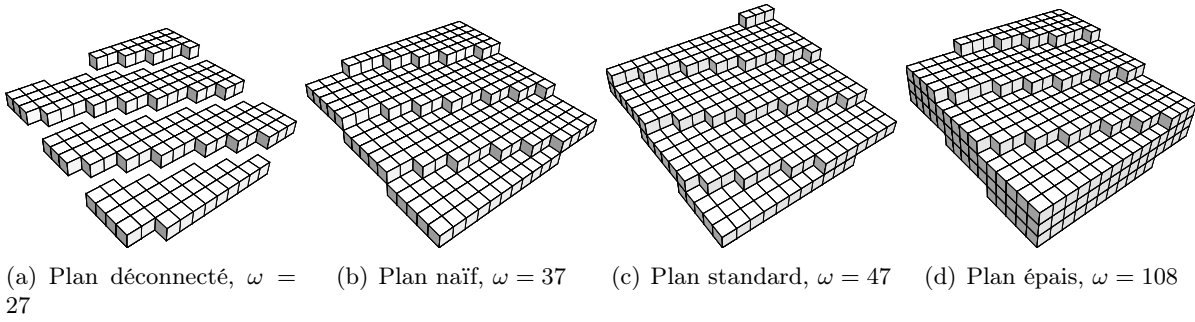


FIGURE 1.27 – Provot [Pro09] : Les plans discrets dont le vecteur normal est  $(3,7,37)$  à travers des épaisseurs différentes  $\omega$ .

### Propriété de régularité des plans discrets

Reposant sur le travail de Hung [SHYH84] dans le cas 2D, Veelaert [Vee93] a proposé une définition similaire dans le cas 3D qui est présentée par la définition ci-dessous :

**Définition 1.6.21** *Un ensemble voxels  $E$  est dit régulier si et seulement si :*

- la projection de  $E$  sur le plan  $Oxy$  est bijective
- pour tout quadruplet  $(u,v,w,t)$  de voxels de  $E$  tel que  $(\overrightarrow{vu_x}, \overrightarrow{vu_y}) = (t\overrightarrow{w_x}, t\overrightarrow{w_y})$ , on a :  $|\overrightarrow{vu_z} - t\overrightarrow{w_z}| < 1$

Veelaert [Vee93] a montré aussi que cette propriété est suffisante pour caractériser tous les morceaux de plan dont la projection sur  $Oxy$  est un rectangle.

### L'approche arithmétique

Andrès [And94b] a développé l'approche de Reveillès [Rev91] pour définir un plan discret arithmétique comme l'ensemble des points qui se trouvent entre deux plans parallèles. La définition ci-dessous présente la définition du plan discret arithmétique.

**Définition 1.6.22** *Un ensemble de voxels  $E$  appartient à un plan discret arithmétique de vecteur normal  $(a,b,c)^T$ , de borne inférieure  $\mu$  et d'épaisseur  $\omega$  ( $a,b,c \in \mathbb{Z}, \text{pgcd}(a,b,c) = 1$ ) est un ensemble de voxels  $(x,y,z)$  qui vérifie cette condition :  $\mu \leq ax + by + cz < \mu + \omega$*

Similairement au cas 2D, l'épaisseur arithmétique  $\omega$  va contrôler l'épaisseur du plan discret. Lorsque la valeur de  $\omega$  est changée, l'épaisseur de ces plans est changée aussi. Donc, nous obtenons des structures discrètes différentes.

- Quand  $\omega = \max(|a|, |b|, |c|)$ , le plan est appelé *naïf*.
- Quand  $\omega = |a| + |b| + |c|$ , le plan est appelé *standard*.
- Quand  $\omega > |a| + |b| + |c|$ , le plan est appelé *épais*.

La figure 1.27 illustre les structures différentes des plans discrets lorsque la valeur de l'épaisseur arithmétique  $\omega$  change.

### 1.6.8 Morceau flou de plan discret

#### Définitions

Provot et al. [PBDR06, PDR08] ont proposé la notion de *morceau flou de plan discret* qui est inspirée de la notion *segment flou* dans le cas 2D. Nous avons les définitions ci-dessous.

**Définition 1.6.23** Soit  $\mathcal{E}_b$  un ensemble de points de  $\mathbb{Z}^3$ . On dit que le plan discret  $\mathcal{P}(a, b, c, \mu, \omega)$  est un **plan englobant** de  $\mathcal{E}_b$  si tous les points de  $\mathcal{E}_b$  appartiennent à  $\mathcal{P}$ . On appelle hauteur du plan  $\mathcal{P}(a, b, c, \mu, \omega)$ , la valeur  $\frac{\omega-1}{N(a,b,c)}$ .

La hauteur d'un plan discret peut être interprétée de la façon suivante :

- ① si  $N = \|\cdot\|_2$ , alors la hauteur  $\frac{\omega-1}{N(a,b,c)}$  représente la distance euclidienne entre les deux plans d'appui du plan  $\mathcal{P}(a, b, c, \mu, \omega)$ . En effet, soient  $P_1 : ax + by + cz = \mu$  et  $P_2 : ax + by + cz = \mu + \omega - 1$  les deux plans d'appui de  $\mathcal{P}$ . Alors comme  $P_1$  and  $P_2$  sont parallèles, la distance entre  $P_1$  et  $P_2$  est égale à  $\frac{|(\mu+\omega-1)-\mu|}{\sqrt{a^2+b^2+c^2}}$ , i.e.  $\frac{\omega-1}{\|(a,b,c)\|_2}$  puisque  $\omega > 0$ .
- ② si  $N = \|\cdot\|_\infty$ , alors la hauteur  $\frac{\omega-1}{N(a,b,c)}$  représente la distance selon la direction principale du vecteur  $(a, b, c)$ . En effet, et sans perte de généralité, nous pouvons supposer que  $\max(|a|, |b|, |c|) = |c|$ , ce qui signifie que la direction principale est colinéaire à l'axe  $Oz$ . Soient  $M_1(x_1, y_1, z_1) \in P_1$  et  $M_2(x_2, y_2, z_2) \in P_2$  tels que  $x_1 = x_2$  et  $y_1 = y_2$ . La distance entre  $P_1$  et  $P_2$  est égale à  $|z_1 - z_2| = \frac{|c(z_1 - z_2)|}{|c|} = \frac{|a(x_1 - y_2) + b(y_1 - y_2) + c(z_1 - z_2)|}{|c|} = \frac{|ax_1 + by_1 + cz_1 - (ax_2 + by_2 + cz_2)|}{|c|} = \frac{|\mu - (\mu + \omega - 1)|}{|c|}$  car  $M_1 \in P_1$  et  $M_2 \in P_2$ , i.e.  $\frac{\omega-1}{\|(a,b,c)\|_\infty}$  puisque  $\omega > 0$ .

**Définition 1.6.24** Soit  $\mathcal{E}_b$  un ensemble de points de  $\mathbb{Z}^3$ . Un plan englobant de  $\mathcal{E}_b$  est dit **optimal** si sa hauteur est minimale.

**Définition 1.6.25** Un ensemble  $\mathcal{E}_b$  de points de  $\mathbb{Z}^3$  est un **morceau flou de plan discret d'épaisseur  $\nu$**  si et seulement si la hauteur d'un plan englobant optimal de  $\mathcal{E}_b$  est inférieure ou égale à  $\nu$ .

La figure 1.28 présente quelques exemples de morceaux flous de plans discrets.

### Reconnaissance des morceaux flou de plan

Provot et al. ont proposé différents algorithmes de reconnaissance relativement aux deux normes considérées ci-dessus. La première méthode [PBDR06] se base sur une approche de programmation linéaire. La deuxième méthode [PDR08] se base sur une approche géométrique.

*Approche par programmation linéaire :* Cette méthode résout le problème en considérant la norme infinie  $\|\cdot\|_\infty$ . Provot rappelle tout d'abord la formulation générale d'un problème de programmation linéaire, ainsi qu'une méthode de résolution : l'algorithme du simplexe. Il voit ensuite comment modéliser le problème de reconnaissance de morceaux flous de plan en ces termes pour pouvoir lui appliquer l'algorithme du simplexe et il propose une interprétation géométrique des opérations du problème dual.

*Approche géométrique :* Cette méthode résout le problème en considérant la norme  $\|\cdot\|_2$ . L'approche exploite en fait la géométrie de l'enveloppe convexe de l'ensemble de points  $S_b$ , en énumérant l'ensemble des paires de plans euclidiens parallèles « intéressants », afin de conserver celles pour lesquelles la distance euclidienne entre les deux plans de la paire est la plus petite.

### 1.6.9 Conclusion

Nous avons brièvement présenté quelques notions de base en géométrie discrète. Après avoir présenté la notion de segment flou, nous avons décrit une étude de l'algorithme de reconnaissance d'un segment flou. Cet algorithme n'utilise pas l'hypothèse sur la condition x-croissant. De plus, il peut fonctionner incrémentalement sur des courbes simples non-connexes. Nous utiliserons cet

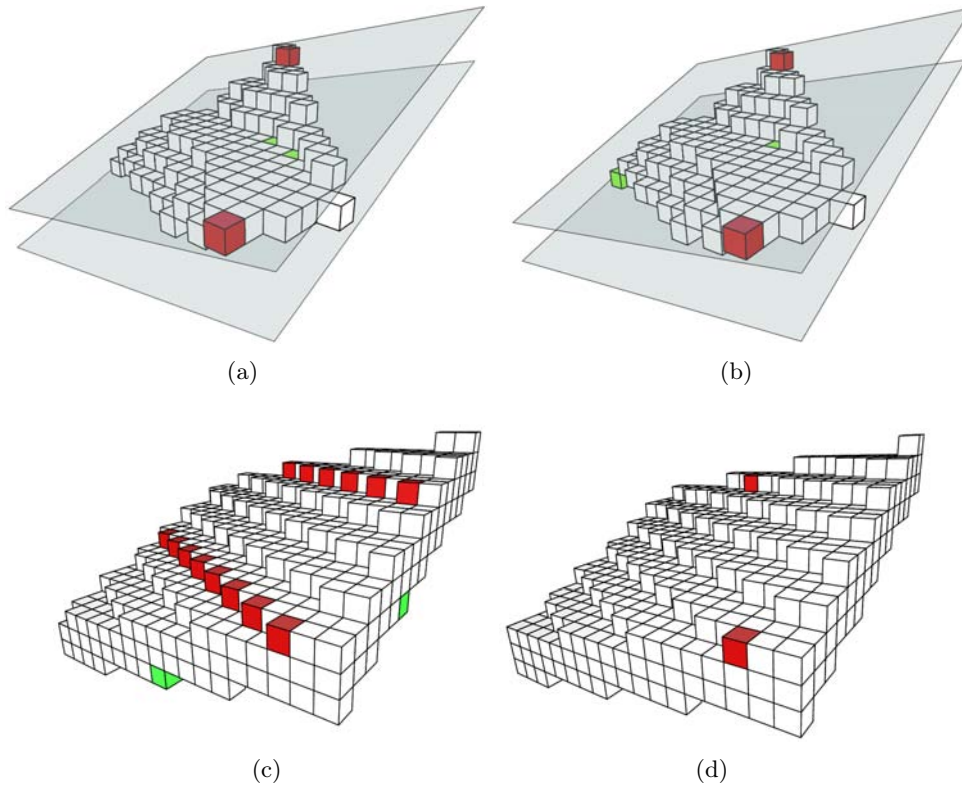


FIGURE 1.28 – Provot [Pro09] : (a) et (b) Un morceau flou de plan discret d'épaisseur 3 et un plan englobant optimal pour (c) la norme euclidienne :  $\mathcal{P}_2(4, 8, 19, -80, 49)$  et la hauteur de  $\mathcal{P}_2 = 2.28$  (d) la norme infinie :  $\mathcal{P}_\infty(31, 65, 157, -680, 397)$  et la hauteur de  $\mathcal{P}_\infty = 2.52$ . Les plans d'appui et les points d'appui correspondants de  $\mathcal{P}_2$  et  $\mathcal{P}_\infty$  sont respectivement représentés sur (a), (b) et (c), (d).

algorithme qui permet de travailler naturellement avec des nombres réels dans les chapitres 2 et 4 pour la reconnaissance des segments flous dans l'espace réel. Nous avons présenté aussi les différentes définitions de cercle discret. Nous nous servirons de ces notions dans les chapitres suivants.



# Chapitre 2

## Estimation des propriétés locales

### Sommaire

---

<b>2.1</b>	<b>Introduction</b>	<b>33</b>
<b>2.2</b>	<b>Etat de l'art</b>	<b>34</b>
2.2.1	Estimation de la courbure	34
2.2.2	Estimation de la torsion	40
<b>2.3</b>	<b>Evolution d'un segment flou d'épaisseur <math>\nu</math></b>	<b>42</b>
2.3.1	Maintenance de l'enveloppe convexe	42
2.3.2	Détermination de l'épaisseur de l'enveloppe convexe	44
<b>2.4</b>	<b>Segment flou maximal d'épaisseur <math>\nu</math></b>	<b>45</b>
2.4.1	Notion de segment flou maximal	45
2.4.2	Une propriété importante de la suite des segments flous maximaux d'une courbe	45
2.4.3	Détermination de la suite des segments flous maximaux d'une courbe	45
<b>2.5</b>	<b>Courbure discrète d'épaisseur <math>\nu</math></b>	<b>46</b>
2.5.1	Définition	46
2.5.2	Estimation de la courbure d'épaisseur $\nu$ en chaque point $C_k$ de $C$	47
<b>2.6</b>	<b>Torsion discrète d'épaisseur <math>\nu</math></b>	<b>49</b>
2.6.1	Définitions	49
2.6.2	Torsion discrète	50
<b>2.7</b>	<b>Expérimentations et comparaisons</b>	<b>51</b>
2.7.1	Expérimentations	51
2.7.2	Comparaisons avec les méthodes existantes	57
<b>2.8</b>	<b>Conclusions et perspectives</b>	<b>58</b>

---

### 2.1 Introduction

Les propriétés géométriques des courbes discrètes sont importantes dans de nombreuses applications en vision par ordinateur, synthèse d'images et reconnaissance des formes. C'est pourquoi, la courbure pour les courbes 2D ainsi que la courbure et la torsion pour les courbes 3D sont des sujets intéressants pour caractériser les courbes discrètes.

En ce qui concerne les courbes planaires, de nombreuses applications dans ces domaines tels que l'approximation des courbes [SDRW06], la compression d'images vectorielles [LaDGJLC05], et en particulier la détection des points dominants après le travail pionnier d'Attneave [Att54],

sont basés sur la propriété de courbure. L'estimation de la courbure est un problème important dans plusieurs applications de traitement d'images qui ont besoin de mesures géométriques sur des objets discrets.

Dans l'espace 3D, la torsion et la courbure sont les propriétés importantes qui permettent de décrire comment une courbe 3D vire. De nombreuses applications ont utilisé les estimations de ces propriétés. Ruben Medina et al. [MWOS04] utilisent l'estimation de la torsion pour construire la description des artères en imagerie médicale. Lewiner et al. [LaDGJLC05] utilisent l'estimation de la torsion et de la courbure dans une application de compression de courbe 3D.

Dans le cadre de la géométrie discrète, des estimateurs de paramètres géométriques ont été proposés mais ils reposent sur la reconnaissance des droites discrètes qui est très sensible au bruit existant dans les courbes étudiées [FT99, CMT01, LVdV05]. Le contour de ces objets discrets est souvent bruité à cause du processus d'acquisition. Par conséquent, la notion de segment flou [DRFRD06] (voir la section 1.6.3) a été introduite, elle permet la segmentation souple des courbes discrètes, en tenant compte du bruit en utilisant la définition arithmétique des droites discrètes [Rev91]. De plus, cette notion permet de manipuler des segments de droite dont les points sont déconnectés.

Nous proposons dans ce chapitre une nouvelle méthode reposant sur la définition des segments flous pour l'estimation des paramètres géométriques locaux des courbes 2D et 3D. Cette méthode utilise une approche géométrique et se base sur les résultats de la géométrie discrète sur la décomposition d'une courbe discrète en la suite de segments flous maximaux [FT99, DRFRD06, NDR07]. Nous présentons dans ce chapitre les résultats obtenus pour un estimateur de courbure en 2D [NDR07] et une extension au cas 3D de ces résultats [NDR08a, NDR08b]. Nous présentons aussi une nouvelle approche pour l'estimation de la torsion discrète.

Ce chapitre est organisé comme suit. La section 2 présente un état de l'art des méthodes existantes dans la littérature en ce qui concerne l'estimation de la courbure et de la torsion. Dans la section 3, nous étudions le problème d'évolution d'un segment flou d'épaisseur  $\nu$  lorsqu'on ajoute (resp. retient) un point au (resp. du) segment flou. Ensuite, nous proposons une extension de la notion de segment flou maximal pour les courbes discrètes générales. Un algorithme permettant de déterminer la suite des segments flous maximaux sera présenté dans la section 4. Dans la section 5, après avoir rappelé la définition de la courbure adaptées aux courbes bruitées, nous proposons un nouvel algorithme d'estimation de la courbure en chaque point d'une courbe. Dans cette section, nous proposons aussi une idée originale afin d'étendre cet algorithme dans l'espace 3D. La section suivante présente les estimateurs de courbure et de torsion pour les courbes 3D. Les expérimentations et les comparaisons sont introduites dans la dernière section.

## 2.2 Etat de l'art

### 2.2.1 Estimation de la courbure

Dans la littérature, les méthodes existantes sont proposées reposant sur des définitions différentes de courbure dans l'espace discret. Ces définitions sont déduites à partir de leurs homologues du cas continu. Dans l'espace continu, celles-ci sont équivalentes, pourtant ce n'est pas pareil dans l'espace discret. Donc, nous rappelons ci-dessous les définitions équivalentes dans l'espace continu qui sont les points de départ pour construire les définitions de courbure discrète.

## Définitions

Dans la géométrie différentielle, il y a plusieurs définitions équivalentes de courbure. Il y a trois définitions qui sont souvent utilisées pour définir la notion de courbure dans l'espace discret. Ces définitions sont équivalentes en géométrie classique, pourtant l'application de celles-ci mène à des définitions différentes en géométrie discrète. Ces définitions sont présentées de la façon suivante.

- La courbure est définie par la dérivée seconde de la courbe.
- La courbure est définie par la dérivée de l'orientation de tangente.
- La courbure est définie par l'inverse de rayon du cercle osculateur.

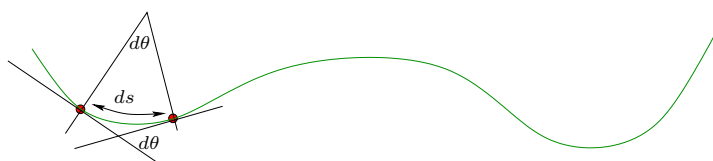


FIGURE 2.1 – Courbe continue dans l'espace 2D.

**Définition 2.2.1** Considérons  $\alpha : I \rightarrow \mathbb{R}^2$  une courbe paramétrée par la longueur d'un arc  $s \in I$ . Alors,

- ①  $\alpha'(s)$  est un vecteur tangent de  $\alpha$  en  $s$ .
- ②  $|\alpha''(s)| = \kappa(s)$  est appelé la courbure de  $\alpha$  en  $s$ .
- ③ Si  $s$  est paramétrée par un paramètre  $t$  ( $s = (x(t), y(t))$ ),  $\kappa(t) = \frac{x'(t)y''(t) - y'(t)x''(t)}{((x'(t))^2 + (y'(t))^2)^{\frac{3}{2}}}$  est la courbure de  $\alpha(s(t))$  en  $s(t)$ .

Nous avons le corollaire ci-dessous.

**Définition 2.2.2** Considérons  $\alpha : I \rightarrow \mathbb{R}^2$  une courbe paramétrée par la longueur d'un arc  $s \in I$ .  $t'(s)$  est un vecteur tangent de  $\alpha$  en  $s$ . Alors,  $|t'(s)|$  est la courbure de  $\alpha$  en  $s$ .

La troisième définition de courbure est basée sur le rayon du cercle osculateur. Un cercle osculateur (voir la figure 2.2) en un point P de la courbe  $\zeta$  est un cercle tangent en P dont le centre se trouve du côté concave de la courbe. Donc, nous avons la définition suivante.

**Définition 2.2.3** Considérons  $\alpha : I \rightarrow \mathbb{R}^2$  une courbe paramétrée par la longueur d'un arc  $s \in I$ .  $R(s)$  est le rayon du cercle osculateur en  $s$ . Alors, la courbure en  $s$  peut être estimée comme suit.

$$\kappa = \begin{cases} +\frac{1}{R(s)} & \text{si la courbe est localement convexe en ce point} \\ -\frac{1}{R(s)} & \text{sinon} \end{cases}$$

Dans l'espace continu, ces trois définitions sont équivalentes. Pourtant, elles sont le point de départ pour concevoir des algorithmes différents pour l'estimation de la courbure dans l'espace discret.

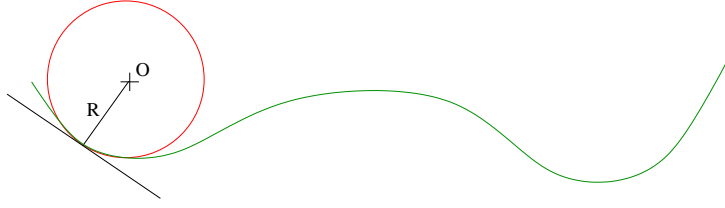


FIGURE 2.2 – La courbure est estimée comme l'inverse du rayon du cercle osculateur.

### Les méthodes existantes

Nous présentons dans cette section certaines des méthodes existantes d'estimation de la courbure discrète tirées de la littérature et reposant sur ces définitions. Considérons une discrétisation  $X$  d'un objet  $O$  dans une grille discrète. Supposons que  $c(X) = \zeta$  est un contour de cet objet discrétisé. Normalement, on peut considérer  $\zeta$  comme une courbe des points 8-connexe. Donc, le problème d'estimation de la courbure discrète est introduit comme suit. Considérons un contour discret  $\zeta = \{C_i\}_0^n$ . Comment pouvons-nous estimer la courbure en chaque point de la courbe ?

#### A. Méthodes qui utilisent la dérivée seconde

**Mokhtarian et al. [MM92]** : Mokhtarian a utilisé la définition 2.2.1 pour construire une méthode d'estimation de la courbure. Considérons une courbe plane  $\Gamma$  paramétrée par la longueur d'un arc  $u : \Gamma = (x(u), y(u))$ . En utilisant la convolution gaussienne, une version convoluée  $\Gamma_\sigma$  est calculée :  $\Gamma_\sigma = (X(u, \sigma), Y(u, \sigma)) = (x(u) \otimes g(u, \sigma), y(u) \otimes g(u, \sigma))$ , où  $g(u, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{u^2}{2\sigma^2}}$ . Donc, la courbure en chaque point peut être calculée sur la courbe évoluée  $\Gamma_\sigma$  comme suit.

$$\kappa(u, \sigma) = \frac{X_u(u, \sigma)Y_{uu}(u, \sigma) - X_{uu}(u, \sigma)Y_u(u, \sigma)}{\sqrt{(X_u(u, \sigma)^2 + Y_u(u, \sigma)^2)^3}}$$

où  $X_u(u, \sigma) = x(u) \otimes g_u(u, \sigma)$ ,  $X_{uu}(u, \sigma) = x(u) \otimes g_{uu}(u, \sigma)$ ,  $Y_u(u, \sigma) = y(u) \otimes g_u(u, \sigma)$ ,  $Y_{uu}(u, \sigma) = y(u) \otimes g_{uu}(u, \sigma)$ ,  $g_u(u, \sigma) = \frac{\partial g(u, \sigma)}{\partial u}$ ,  $g_{uu}(u, \sigma) = \frac{\partial^2 g(u, \sigma)}{\partial u^2}$ .

De la même façon pour les courbes 3D, une version convoluée de la courbe est définie par  $\Gamma_\sigma = (X(u, \sigma), Y(u, \sigma), Z(u, \sigma))$ . La courbure est estimée par la formule suivante.

$$\kappa(u, \sigma) = \frac{\sqrt{(X_u Y_{uu} - X_{uu} Y_u)^2 + (Y_u Z_{uu} - Y_{uu} Z_u)^2 + (Z_u X_{uu} - Z_{uu} X_u)^2}}{\sqrt{(X_u^2 + Y_u^2 + Z_u^2)^3}}$$

**Lewiner et al. [LaDGJLC04]** : Lewiner a aussi proposé une technique d'estimation de la courbure reposant sur la dérivée seconde. Sa méthode se base sur la technique des moindres carrés et sur l'approximation locale de la longueur d'un arc. Considérons une suite de points  $\{p_i\}$  dans une courbe  $r$ . Pour chaque point  $p_j$ , il considère une fenêtre de  $2q + 1$  points autour de  $p_j = r(j)$  pour estimer les dérivées de  $r(s)$ . Supposons que  $s_i$  soit la longueur d'un arc correspondant au point  $p_i$  en partant de  $p_0$ . Cette longueur d'un arc  $l_i$  est estimée comme  $l_i = \sum_{k=0}^{i-1} \text{longueur}(p_k p_{k+1})$ . En considérant que  $p_0 = r(0)$  est l'origine, il a utilisé une approximation de second degré  $r(s) = r'(0)s + \frac{1}{2}r''(0)s^2 + g_1(s)s^3$  avec  $\lim_{s \rightarrow 0} g_1(s) = 0$ . L'estimation des dérivées est effectuée par la technique des moindres carrés. Son idée est d'approximer localement la courbe

par une courbe paramétrique  $(\hat{x}(s), \hat{y}(s))$ . Une fonction de coordonnée est une fonction quadrique de la longueur d'un arc (par exemple :  $\hat{x}(s) = x_0 + x'_0 \cdot s + \frac{1}{2}x''_0 \cdot s^2$ ). Les dérivées  $x'_0$  et  $x''_0$  peuvent être estimées en minimisant  $E_x(x'_0, x''_0) = \sum_{i=-q}^q \omega_i (x_i - x'_0 l_i - \frac{1}{2}x''_0 (l_i)^2)^2$ . Le poids  $\omega_i$  du point  $p_i$  peut être considéré simplement  $\omega_i = 1$ . L'estimation de  $y'_0$  et  $y''_0$  effectuée repose sur les propriétés des vecteurs de tangente et des vecteurs normaux.

$$\begin{cases} (x'_0)^2 + (y'_0)^2 = 1 \\ x'_0 x''_0 + y'_0 y''_0 = 0 \end{cases}$$

Puis, la valeur de courbure en chaque point de la courbe est déterminée reposant sur la formule suivante.

$$\kappa(t) = \frac{r' \times r''}{\|r'\|^3}$$

Pour les courbes 3D, une approche similaire peut être appliquée, et la courbure est estimée en utilisant la formule ci-dessous.

$$\kappa(t, q, \sigma) = \frac{r' \times r''}{\|r'''\|}$$

### B. Méthodes qui utilisent un processus de calcul de variation d'angle des tangentes

**Worring et al. [WS93a]** : Worring a premièrement proposé une méthode pour l'estimation de la courbure reposant sur le calcul de variation d'angle des tangentes. Il a introduit l'estimateur suivant :

$$\kappa(p_i) = \frac{\theta(p_i) * G'_\sigma}{1.107}$$

où  $\theta(p_i) = \angle(\text{LineFitting}(p_i, m), 0x)$  et  $G'_\sigma$  est un filtre Gaussien d'écart type  $\sigma$ . Son idée est de déterminer une tangente en chaque point  $p_i$  (procédure  $\text{LineFitting}(p_i, m)$ ) reposant sur la technique des moindres carrés sur les points  $\{p_j\}_{j=i-m}^{i+m}$ . La fonction  $\theta(p_i)$  correspond à l'angle entre cette tangente et l'axe  $Ox$ . La convolution de cette fonction avec un filtre Gaussien dérivatif permet d'estimer la dérivée de  $\theta$ . Le coefficient 1.107 est la distance moyenne entre 2 pixels sur une distribution aléatoire de droites.

**Vialard [Via96]** : Vialard a développé l'approche de Worring et al. en remplaçant la technique des moindres carrés par une notion de tangente discrète. La formule est la même par rapport à celle de Worring. Une tangente discrète est construite en reconnaissant un segment de droite discrète naïve [DR95]. Un segment de droite est un ensemble fini de points 8-connexes qui appartient à une droite discrète naïve. Une tangente discrète en chaque point est considérée comme un segment de droite qui entoure ce point. **Feschet et Tougne [FT99]** : Feschet et al. [FT99] ont optimisé la méthode de Vialard en temps linéaire. Ils ont considéré les tangentes qui sont définies par Vialard comme des segments maximaux. Ils ont construit une bijection entre les tangentes en chaque point de la courbe et l'ensemble de segments maximaux de cette courbe. En introduisant un algorithme optimal pour la décomposition d'une courbe en la suite de segments maximaux, l'estimation de la courbure peut être effectuée en temps linéaire.

**Kerautret et Lachaud [KL09a]** : Kerautret et Lachaud [KL09a] ont proposé une nouvelle méthode d'estimation de la courbure reposant sur la dérivée d'angle des tangentes et sur une

optimisation globale. Cette méthode exploite les propriétés locales des courbes discrètes en utilisant les bornes sur les orientations des tangentes qui sont définies par des segments maximaux de droite. Cet estimateur s'adapte aux courbes bruitées en remplaçant les segments par des segments flous maximaux. **C. Méthodes basées sur l'estimation de cercle osculateur**

**Kovalevsky [Kov90]** : Kovalevsky a proposé une solution géométrique afin de calculer les arcs réels possibles qui séparent 2 ensembles de points discrets reposant sur l'algorithme de Voronoï. Considérons la figure 2.3, un objet discret est présenté sur l'image (a). Puis, deux composants connectés sont construits à partir de cet objet (voir l'image(b)). En considérant les deux bords de ces composants, il obtient dans l'image (c) deux ensembles des points qui sont donnés en entrée de la méthode de séparation par arcs reposant sur l'algorithme de Voronoï.

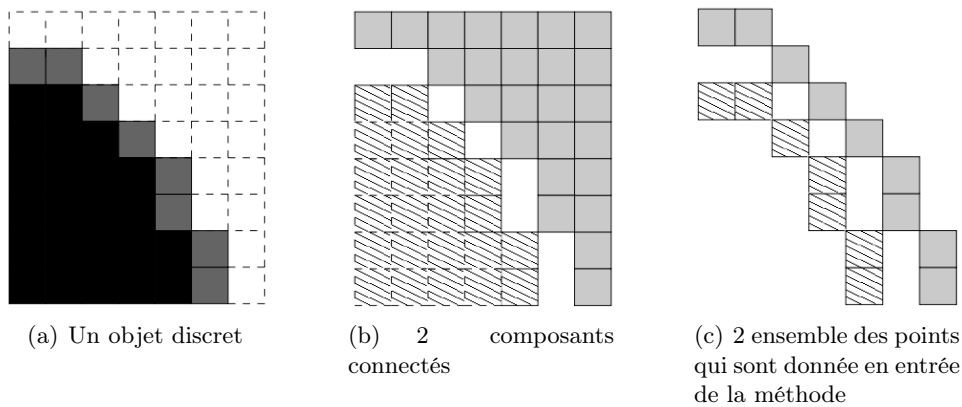


FIGURE 2.3 – Méthode de Kovalevsky [Kov90].

**Worring et al. [WS93a]** : Worring et al. ont proposé aussi une méthode reposant sur l'estimation du cercle osculateur. Ils utilisent la formule ci-dessous

$$k^*(p_i) = \frac{1}{r^*(p_i)}$$

avec

$$(c^*, r^*) = ArcFitting(p_i, m)$$

L'estimation du centre  $c^*$  et du rayon  $r^*$  du cercle osculateur est calculé par la technique des moindres carrés sur une fenêtre de  $m$  pixels. La valeur de courbure en un point est donnée par l'inverse du rayon estimé.

**Coeurjolly et al. [CMT01]** : Coeurjolly et al. ont proposé une méthode [CMT01] reposant sur la représentation des cercles de Pitteway [M.L74]. Dans cette représentation, un cercle discret arithmétique de centre  $(0,0)$  et de rayon  $R$  peut être défini par le faisceau de paraboles suivants :

$$\left\{ y = \left[ \sqrt{x(1+2k) - k^2 + 1/4} \right] x = R - k0 \leq k \leq R \right.$$

Pour une hauteur  $l$ , les cercles arithmétiques dont la première parabole a une hauteur  $l$  ont un rayon entier qui est entre  $r_{inf}$  et  $r_{sup}$ . Ces rayons sont définis comme suit.

$$\left\{ r_{inf} = \lceil \left( l - \frac{1}{2} \right)^2 - \frac{1}{4} \rceil r_{sup} = \lfloor \left( l + \frac{1}{2} \right)^2 - \frac{1}{4} \rfloor \right.$$

Le rayon entier peut être calculé comme la valeur moyenne entre ces 2 rayons  $r_{inf}$  et  $r_{sup}$ . Il a aussi montré qu'on peut estimer  $l$  comme une demi-tangente qui est déterminée en appliquant l'algorithme de Feschet et Tougne [FT99]. Donc, la courbure en chaque point peut être estimée comme suit.

$$\kappa^* = \frac{2}{r_{inf} + r_{sup}}$$

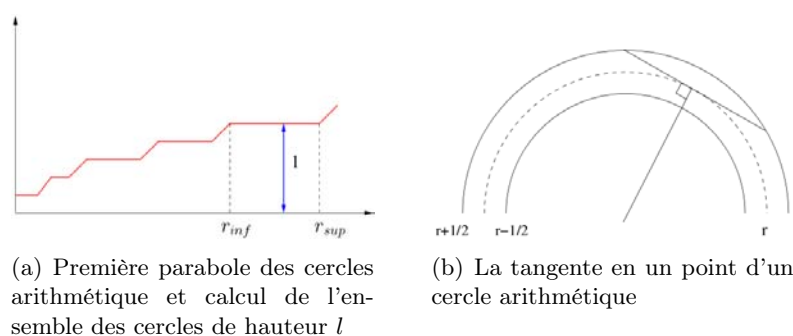


FIGURE 2.4 – Méthode de Coeurjolly et al [CMT01].

**Coeurjolly et al. [CS03]** : Coeurjolly et al. ont aussi proposé une autre solution [CS03] pour estimer la courbure discrète. Cet estimateur repose sur la *courbure normalisée d'ordre  $m$* . L'idée est de construire en chaque point  $p_i$  de la courbe les extrémités du segment de droite discrète le plus long à gauche  $P_G$  et à droite  $P_D$ . L'estimation de la courbure au point  $p_i$  de la courbe est donnée par l'inverse du rayon du cercle circonscrit du triangle  $P_D p_i P_G$ . On note les longueurs des segments de droite de ce triangle comme  $a$ ,  $b$ , et  $c$ . Alors, le rayon du cercle circonscrit est déterminé par :

$$R_c = \frac{abc}{\sqrt{(a+b+c)(a+b-c)(a+c-b)(b+c-a)}}$$

Alors, la courbure en ce point est estimée comme suit.

$$\kappa = \frac{1}{R_c} = \frac{\sqrt{(a+b+c)(a+b-c)(a+c-b)(b+c-a)}}{abc}$$

**Debled-Rennesson [DR04]** : Debled-Rennesson [DR04] a fait évoluer l'approche de Coeur-

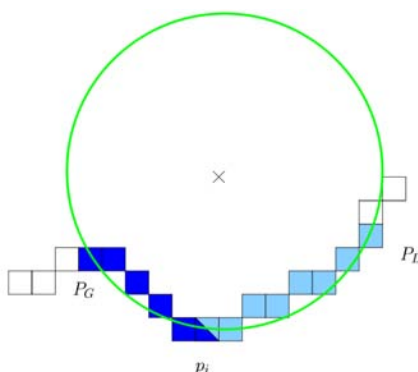


FIGURE 2.5 – Estimateur de courbure de Coeurjolly et al. [CS03] basé sur les demi-tangentes à gauche et à droite.

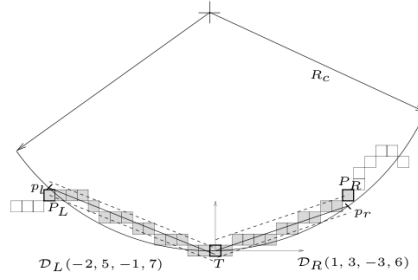


FIGURE 2.6 – Courbure discrète adaptée au bruit.

jolly et al. [CS03] en introduisant la notion de courbure qui s'adapte au bruit. Reposant sur un algorithme de reconnaissance d'un segment flou d'ordre  $d$  [DRRRD03], les demi-tangentes en chaque point sont des segments flous d'ordre  $d$  à gauche et à droite les plus longs à partir de ce point. Une version améliorée a été proposée dans [SDRW06]. Dans ce travail, la reconnaissance d'un segment flou d'épaisseur  $\nu$  a été remplacée afin de reconnaître 2 demi-tangentes en chaque point. En bref, les méthodes existantes dans la littérature estiment la courbure reposant sur une des définitions de courbure dans l'espace discret. Il existe trois approches principales qui correspondent à 3 définitions de courbure dans l'espace discret.

## 2.2.2 Estimation de la torsion

La torsion est une notion complémentaire à la courbure pour les courbes en 3D. Normalement, la courbure n'est pas suffisante pour caractériser les propriétés locales des courbes 3D. Cette notion mesure seulement la rapidité du changement de direction de la courbe. C'est suffisant dans le cas des courbes planaires car ses plans osculateurs ne changent pas. Ce n'est pas correct en ce qui concerne des courbes 3D avec lesquelles le plan osculateur en chaque point change souvent. Donc, la torsion est proposée pour mesurer la rapidité du changement de direction du plan osculateur de la courbe. Pour une meilleure compréhension, nous rappelons ci-dessous quelques définitions dans la géométrie classique (voir [adC76] pour avoir des informations plus détaillées).

### A. Torsion en géométrie classique

Nous présentons dans cette section les définitions de courbure et de torsion en géométrie discrète.

**Définition 2.2.4** *Supposons que  $r : I \rightarrow \mathbb{R}^3$  est une courbe paramétrisée par  $t$ . Alors,*

- $T(t)$  est un vecteur unitaire dans la direction  $r'(t)$ .  $T(t)$  est appelé un vecteur unitaire de tangente.
- $N(t)$  est un vecteur unitaire dans la direction  $r''(t)$ .  $N(t)$  est appelé un vecteur normal pour  $T(t)$ .
- $\kappa(t) = |T'(t)|$  est appelé la courbure de  $r$  à  $t$ .
- Le plan qui est déterminé par le vecteur normal ( $N(t)$ ) et le vecteur tangent ( $T(t)$ ) est dit plan osculateur à  $t$ . Le vecteur unitaire  $B(t) = T(t) \wedge N(t)$
- $\tau(r) = |B'(t)|$  est la torsion de la courbe en  $t$

**Définition 2.2.5** *Supposons que  $r : I \rightarrow \mathbb{R}^3$  est une courbe 3D paramétrée par  $t$*

- La courbure de  $r$  à  $t \in I$  :  $\kappa(t) = \frac{|r' \wedge r''|}{|r'|^3}$



$$- \text{ La torsion de } r \text{ à } t \in I : \tau(t) = \frac{(r' \wedge r'').r'''}{|r' \wedge r''|^2}$$

## B. Les méthodes existantes d'estimation de la torsion discrète.

Plusieurs méthodes [LaDGJLC05, Mok97, MWOS04, KdF88] ont été proposées pour l'estimation de la torsion. Mokhtarian [Mok97] a utilisé directement la formule de la torsion en utilisant un lissage gaussien. Lewiner et al. [LaDGJLC05] ont utilisé la méthode des moindres carrés pour estimer la courbure et la torsion d'une courbe discrète 3D. Ruben Medina et al. [MWOS04] ont proposé deux méthodes pour estimer les valeurs de la torsion et de la courbure en chaque point de la courbe. La première utilise la transformée de Fourier, la deuxième est basée sur la technique des moindres carrés. Nous présentons ci-dessous quelques méthodes connues dans la littérature.

**Mokhtarian et al. [Mok97]** : Mokhtarian a utilisé un filtre gaussien pour estimer la torsion en chaque point de la courbe. Par exemple, pour une courbe 3D qui est représentée paramétriquement comme suit  $r(u) = (x(u), y(u), z(u))$ , il a considéré une convolution de la courbe par un noyau de Gaussien  $g(u, \sigma) : \Gamma(u, \sigma) = r(u) \otimes g(u, \sigma)$  où  $\otimes$  est un opérateur de convolution. On sait que :

$$\dot{\Gamma}(u, \sigma) = \frac{\partial \Gamma(u, \sigma)}{\partial u} = r(u) \otimes \frac{\partial g(u, \sigma)}{\partial u}$$

$$\ddot{\Gamma}(u, \sigma) = \frac{\partial^2 \Gamma(u, \sigma)}{\partial u^2} = r(u) \otimes \frac{\partial^2 g(u, \sigma)}{\partial u^2}$$

et

$$\dddot{\Gamma}(u, \sigma) = \frac{\partial^3 \Gamma(u, \sigma)}{\partial u^3} = r(u) \otimes \frac{\partial^3 g(u, \sigma)}{\partial u^3}$$

$$\text{Supposons que } \begin{cases} X(u, \sigma) = x(u) \otimes g(u, \sigma) \\ Y(u, \sigma) = y(u) \otimes g(u, \sigma) \\ Z(u, \sigma) = z(u) \otimes g(u, \sigma) \end{cases}$$

Grâce aux propriétés de convolution ci-dessus, la torsion en chaque point peut être estimée comme suit.

$$\tau(u, \sigma) = \frac{\dot{X}(\ddot{Y}\ddot{Z} - \ddot{Z}\ddot{Y}) + \dot{Y}(\ddot{Z}\ddot{X} - \ddot{X}\ddot{Z}) + \dot{Z}(\ddot{X}\ddot{Y} - \ddot{Y}\ddot{X})}{(\dot{X}\ddot{Y} - \ddot{X}\dot{Y})^2 + (\dot{Y}\ddot{Z} - \ddot{Y}\dot{Z})^2 + (\dot{Z}\ddot{X} - \ddot{Z}\dot{X})^2}$$

**Lewinner et al. [LaDGJLC05]** : En suivant le même principe pour la courbure, Lewinner et al. ont proposé un estimateur de torsion en chaque point d'une courbe paramétrique reposant sur la technique des moindres carrés. Considérons une suite de points  $\{p_i\}$  sur la courbe  $r$ . Pour chaque point  $p_j$ , il considère une fenêtre de  $2p + 1$  points autour de  $p_j$  afin d'estimer les dérivées de  $r(s)$ . Supposons que  $s_i$  est la longueur d'un arc qui correspond à  $p_i$  en partant de  $p_0$ . Elle

peut être estimée comme  $l_i = \sum_{k=0}^{i-1} p_k p_{k+1}$ . Sans perte de généralité, il suppose que  $p_0 = r(0)$  est

l'origine. Pour estimer la torsion en chaque point, il utilise l'approximation d'ordre 3 de la série de Taylor.  $r(s) = r'(0)s + \frac{1}{2}r''(0)s^2 + \frac{1}{6}r'''(0)s^3 + g_2(s)s_i^4 +$  avec  $\lim_{s \rightarrow 0} g_2(s) = 0$ . Concernant les courbes 3D, l'estimateur de torsion utilise une approximation des points par une courbe cubique paramétrique. Alors,  $x'_0$ ,  $x''_0$  et  $x'''_0$  sont déterminés en minimisant :

$$E_x(x'_0, x''_0, x'''_0) = \sum_{i=-q}^q \omega_i (x_i - (x'_0 s_i + \frac{1}{2} x''_0 s_i^2 + \frac{1}{6} x'''_0 s_i^3))^2$$

Similairement, on peut estimer les valeurs ci-dessous :  $y_0', y_0'', y_0''', z_0', z_0'', z_0'''$ . Donc, la torsion est déterminée comme suit.

$$\tau(t, q, \sigma) = \frac{(r' \times r'').r'''}{\|r' \times r''\|^2}$$

Dans les sections suivantes, nous allons présenter une nouvelle méthode reposant sur une approche de géométrie discrète pour estimer la courbure en chaque point d'une courbe discrète 2D, la courbure et la torsion en chaque point d'une courbe discrète 3D.

Dans les sections suivantes, nous allons présenter les méthodes d'estimation de la courbure discrète et de la torsion discrète en chaque point de la courbe reposant sur l'étude de la suite de segments flous maximaux.

## 2.3 Evolution d'un segment flou d'épaisseur $\nu$

Dans [DRFRD06], Debled et al. ont introduit un algorithme optimal pour segmenter un segment flou en temps linéaire. Cet algorithme est incrémental, il se base sur le calcul de l'épaisseur de l'enveloppe convexe du segment flou en ajoutant incrémentalement des points dans une direction. Ces travaux ont permis d'affirmer que la reconnaissance d'un segment flou d'épaisseur  $\nu$  correspond au calcul de la hauteur et de la largeur de l'enveloppe convexe de ce segment flou. Pourtant, cette méthode ne fonctionne que dans le cas incrémental. Elle ne fonctionne pas dans le cas dynamique où un point peut être ajouté au segment flou ou être enlevé de ce segment flou. Dans ce paragraphe, nous présentons une nouvelle stratégie pour maintenir un segment flou d'épaisseur  $\nu$  lorsqu'on ajoute (resp. supprime) un point à (resp. d') un segment flou.

La nouvelle stratégie repose sur la maintenance de l'enveloppe convexe [OvL81] et sur l'estimation de la hauteur de l'enveloppe convexe [Buz05] en temps  $O(\log^2 n)$ . Nous présentons brièvement ces résultats dans la section suivante.

### 2.3.1 Maintenance de l'enveloppe convexe

Le problème de maintenance de l'enveloppe convexe introduit par Overmars et al. [OvL81] est défini comme suit :

**Problème 1** *Soit un ensemble vide initial  $S$  et une liste de  $n$  points  $(p_1, p_2, \dots, p_n)$  dans laquelle chaque point correspond à une insertion ou suppression à partir de  $S$  (un point peut être supprimé s'il existe dans  $S$ ), comment peut-on maintenir l'enveloppe convexe de  $S$  ?*

Overmars et Van Leeuwen ont considéré une enveloppe convexe comme l'union de 2 parties : l'enveloppe convexe haute ( $U_{hull}$ ) et l'enveloppe convexe basse ( $L_{hull}$ ) (voir aussi la figure 2.7). Ces parties de l'enveloppe convexe sont représentées par une structure de données *Concatenable Queue* qui permet les opérations de recherche, insertion, suppression, découpage et fusion en temps  $O(\log n)$  [AHU74]. Ils calculent  $U_{hull}$  et  $L_{hull}$  après chaque opération (insertion ou suppression) en utilisant une stratégie de découpage et fusion. Cette stratégie est basée sur la détermination d'un pont entre 2 enveloppes convexes en temps  $O(\log n)$  (voir la figure 2.9). Puis, la nouvelle enveloppe convexe est déterminée en unissant  $U_{hull}$  et  $L_{hull}$ . Grâce à leur méthode, ils ont proposé et prouvé le théorème ci-dessous :

**Théorème 2.3.1** *Les enveloppes convexes  $U$ -hull et  $L$ -hull de l'ensemble  $S$  de  $n$  points peuvent être maintenues dynamiquement dans le pire cas en temps  $O(\log^2 n)$  par une opération d'insertion ou de suppression.*

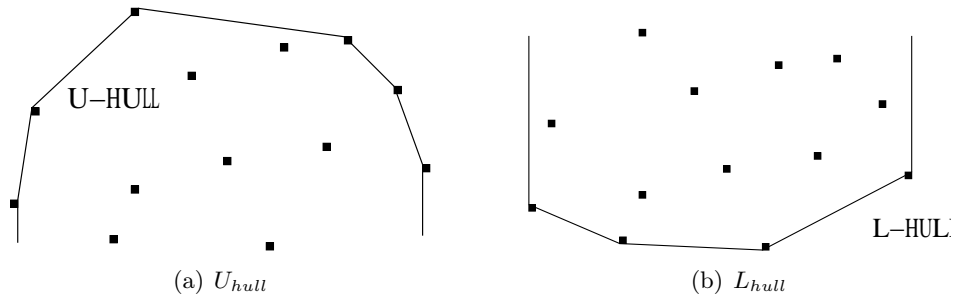


FIGURE 2.7 – Représentation d'une enveloppe convexe par l'union de 2 parties : l'enveloppe convexe supérieure et l'enveloppe convexe inférieure.

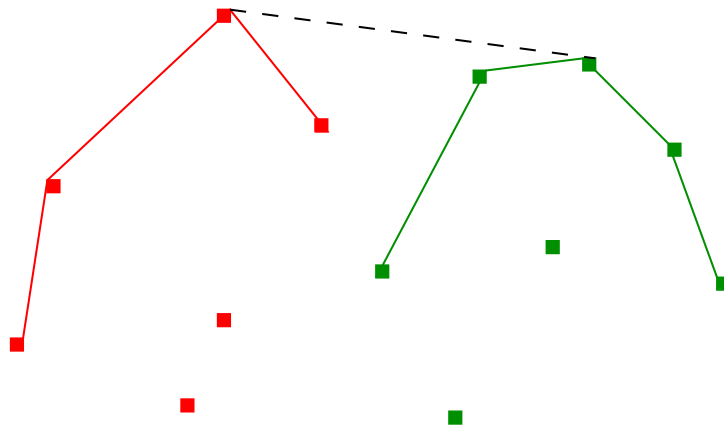


FIGURE 2.8 – Détermination du pont entre 2 enveloppes convexes de deux ensembles de points.

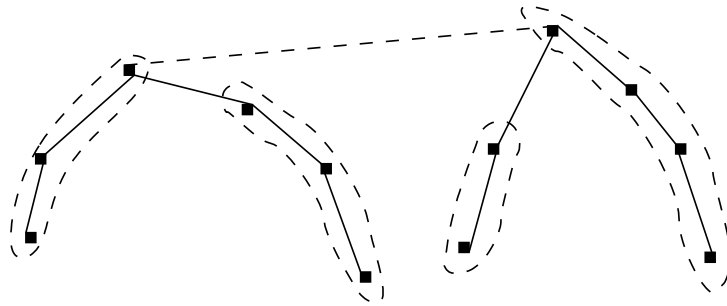


FIGURE 2.9 – Evolution de l’enveloppe convexe en haute.

La figure 2.9 illustre la stratégie de découpage et fusion des enveloppes convexes supérieures reposant sur la détermination du pont entre 2 enveloppes convexes.

### 2.3.2 Détermination de l’épaisseur de l’enveloppe convexe

Nous utilisons la technique de recherche binaire [Buz05] pour déterminer l’épaisseur verticale et l’épaisseur horizontale de l’enveloppe convexe. L’enveloppe convexe est considérée comme une union de 2 parties  $U_{hull}$  et  $L_{hull}$ . Cette technique est basée sur la propriété de concavité de la fonction  $height(x) = U_{hull}(x) - L_{hull}(x)$ . De plus, l’épaisseur en chaque point sur l’enveloppe convexe est déterminée en temps  $O(\log n)$  en utilisant la technique de recherche binaire afin de déterminer l’arrête qui se trouve en face de ce point. C’est pourquoi, l’épaisseur de l’enveloppe convexe peut être déterminée en temps  $O(n \log^2 n)$ . La figure 2.10 illustre cette stratégie de détermination de l’épaisseur de l’enveloppe convexe.

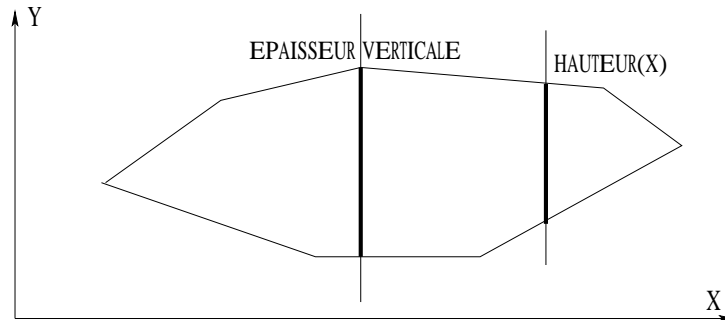


FIGURE 2.10 – Calcul de l’épaisseur verticale.

Grâce aux 2 résultats ci-dessus, nous pouvons déterminer les paramètres d’un nouveau segment flou lorsqu’on ajoute (resp. supprime) un point à (resp. d’) un segment flou. Cette méthode peut travailler avec des courbes générales ou bien déconnectées.

Cette idée nous permet de déduire une procédure de maintenance de segment flou d’épaisseur  $\nu$  nommé *SegmentMaintain*. L’utilisation de cette procédure est comme suit :

- $SegmentMaintain(S_f, M, INSERT)$  permet de déterminer un nouveau segment flou en ajoutant le point  $M$  à  $S_f$
- $SegmentMaintain(S_f, M, DELETE)$  permet de déterminer un nouveau segment flou en supprimant le point  $M$  de  $S_f$

## 2.4 Segment flou maximal d'épaisseur $\nu$

### 2.4.1 Notion de segment flou maximal

Dans cette partie, nous introduisons quelques notions sur les segments flous maximaux et les tangentes en utilisant des notations présentées dans [LVdV05]. Il faut rappeler que les segments flous et les tangentes utilisées ici sont des segments flous d'épaisseur  $\nu$ , proposés dans [DRFRD06]. Considérons  $C_{i,j}$ , un segment flou d'épaisseur  $\nu$ . Il est noté par  $S(i, j, \nu)$ . Le premier indice  $j$ ,  $i \leq j$ , tel que  $S(i, j, \nu)$  et  $\neg S(i, j+1, \nu)$ , est appelé *front* de  $i$ . Le front de  $i$  est noté par  $F(i)$ . De même façon, le première indice  $i$ ,  $i \leq j$ , tel que  $S(i, j, \nu)$  et  $\neg S(i-1, j, \nu)$ , est appelé *queue* de  $j$  et noté par  $Q(j)$ .

**Définition 2.4.1** Un  $S(i, j, \nu)$  est dit segment flou maximal d'épaisseur  $\nu$  noté par  $MS(i, j, \nu)$  ssi  $S(i, j, \nu)$  et  $\neg S(i, j+1, \nu)$  et  $\neg S(i-1, j, \nu)$ .

Il est évident que pour  $MS(i, j, \nu)$  on a :  $F(i) = j$  et  $Q(j) = i$ . Du côté du segment flou maximal d'épaisseur  $\nu$ , nous proposons aussi la notion de demi-segment flou maximal d'épaisseur  $\nu$

**Définition 2.4.2** Un segment flou d'épaisseur  $\nu$   $S(i, j, \nu)$  est dit demi-segment flou maximal d'épaisseur  $\nu$  noté par  $DMS(i, j, \nu)$  ssi il satisfait un des deux critères suivants :

- $F(i) = j$
- $Q(j) = i$

Quand  $F(i) = j$ , on parle de demi-segment flou maximal à droite. Quand  $Q(j) = i$ , on parle de demi-segment flou maximal à gauche.

### 2.4.2 Une propriété importante de la suite des segments flous maximaux d'une courbe

**Proposition 2.4.1** Soit  $C$  - une courbe discrète,  $MS_\nu(C)$  - la suite de segments flous maximaux d'épaisseur  $\nu$  de la courbe  $C$ .  $MS_\nu(C) = \{S(B_1, E_1, \nu), S(B_2, E_2, \nu), \dots, S(B_m, E_m, \nu)\}$  satisfait  $B_1 < B_2 < \dots < B_m$ . Alors, on a :  $E_1 < E_2 < \dots < E_m$

**Preuve 2.4.1** Considérons la reconnaissance de 2 segments flous maximaux consécutifs :  $S(B_i, E_i, \nu)$  et  $S(B_{i+1}, E_{i+1}, \nu)$ . Grâce à l'hypothèse, nous avons  $B_i < B_{i+1}$ . Supposons que  $E_i \geq E_{i+1}$ , donc  $S(B_{i+1}, E_{i+1}, \nu)$  fait partie de  $S(B_i, E_i, \nu)$ . C'est pourquoi,  $S(B_{i+1}, E_{i+1}, \nu)$  n'est pas un segment flou maximal. Cette contradiction prouve la proposition 2.4.1.

### 2.4.3 Détermination de la suite des segments flous maximaux d'une courbe

Dans cette partie nous proposons une méthode pour déterminer les segments flous maximaux d'épaisseur  $\nu$  d'une courbe discrète  $C$ .

#### Cas 2D

##### Complexité :

Chaque point de la courbe est considéré au maximum 2 fois dans cet algorithme. La détermination de la nouvelle droite discrète englobante optimale lorsqu'on ajoute (resp. supprime) un point à (resp. d') un segment flou est effectuée en temps  $O(\log^2 n)$ . Alors, la complexité de cet algorithme est en  $O(n \log^2 n)$ .

---

**Algorithme 6** : Algorithme de décomposition d'une courbe 2D  $C$  en sa suite de segments flous maximaux d'épaisseur  $\nu$ .

---

**Données** :  $C$  - La courbe discrète de  $n$  points,  $\nu$  - l'épaisseur de segmentation

**Résultat** :  $MBS_\nu$  - la suite des segments flous maximaux d'épaisseur  $\nu$

**début**

$k=0$ ;  $S_b = \{C_0\}$ ;  $MBS_\nu = \emptyset$ ;  $a = 0$ ;  $b = 1$ ;  $\omega = b$ ,  $\mu = 0$ ;

**pour**  $\frac{\omega-1}{\max(|a|,|b|)} \leq \nu$  **faire**

$k++$ ;  $S_b = S_b \cup C_k$ ; Déterminer  $D(a, b, \mu, \omega)$  of  $S_b$ ;

$bSegment=0$ ;  $eSegment=k-1$  ;

$MBS_\nu = MBS_\nu \cup MBS(bSegment, eSegment, \nu)$ ;

**pour**  $k < n - 1$  **faire**

**pour**  $\frac{\omega-1}{\max(|a|,|b|)} > \nu$  **faire**

$S_b = S_b \setminus C_{bSegment}$ ;  $bSegment++$  ;

    Déterminer  $D(a, b, \mu, \omega)$ , la droite englobante optimale pour  $S_b$ ;

**pour**  $\frac{\omega-1}{\max(|a|,|b|)} \leq \nu$  **faire**

$k++$ ;  $S_b = S_b \cup C_k$ ;

    Déterminer  $D(a, b, \mu, \omega)$ , la droite englobante optimale pour  $S_b$ ;

$eSegment=k-1$ ;  $MBS_\nu = MBS_\nu \cup MBS(bSegment, eSegment, \nu)$ ;

**fin**

---

## Cas 3D

L'algorithme 7 nous permet d'obtenir une décomposition d'une courbe discrète 3D  $C$  en sa suite des segments flous maximaux 3D d'épaisseur  $\nu$  en temps  $O(n \log^2 n)$ . Il utilise l'algorithme 6 pour déterminer les segments flous maximaux 2D à partir des projections de la courbe sur les plans de coordonnées.

### Complexité :

Pour déterminer la droite discrète englobante optimale pour le segment flou 3D  $S_b$  (l'étape marquée par (\*) dans l'algorithme 7), nous considérons les caractéristiques de deux segments flous 2D qui sont obtenus dans les plans de projection et nous les combinons afin d'obtenir les caractéristiques de la droite englobante optimale 3D pour  $S_b$ . Donc, la décomposition d'une courbe 3D en sa suite de segments flous maximaux d'épaisseur  $\nu$  peut être faite en temps  $O(n \log^2 n)$ .

## 2.5 Courbure discrète d'épaisseur $\nu$

### 2.5.1 Définition

Nous rappelons dans cette section l'estimateur de courbure qui est adapté aux courbes bruitées [DR04]. Il est déduit directement à partir de l'estimateur de Coeurjolly [CMT01] pour des courbes 2D non bruitées. Cette technique peut être considérée comme une généralisation de la courbure normalisée d'ordre  $m$  [HS98].

Nous considérons  $C$ , une courbe discrète en 2D ou en 3D, et  $C_k$  un point de la courbe. Nous considérons des points  $C_l$  et  $C_r$  qui satisfont :  $l < k < r$ ,  $BS(l, k, \nu)$  et  $\neg BS(l - 1, k, \nu)$ ,  $BS(k, r, \nu)$  et  $\neg BS(k, r + 1, \nu)$ .

L'estimation de la **courbure d'épaisseur  $\nu$  en point**  $C_k$  est déterminée comme l'inverse du rayon du cercle circonscrit qui passe par trois points  $C_l$ ,  $C_k$  et  $C_r$ . Afin de déterminer le rayon

---

**Algorithme 7** : Algorithme de décomposition d'une courbe 3D  $C$  en sa suite de segments fous maximaux d'épaisseur  $\nu$ .

---

**Données** :  $C$  - courbe discrète de  $n$  points,  $\nu$  - l'épaisseur de segmentation

**Résultat** :  $MBS_\nu$  - la suite de segments fous maximaux d'épaisseur  $\nu$  de  $C$

**début**

$k=0$ ;  $S_b = \{C_0\}$ ;  $MBS_\nu = \emptyset$ ;  $a = 0$ ;  $b = 1$ ;  $\omega = b$ ,  $\mu = 0$ ;

**pour** les épaisseur de 2 segments fous qui sont obtenus sur les projections des points de  $S_b$  sur les plans de coordonnée sont  $\leq \nu$  **faire**

$k++$ ;  $S_b = S_b \cup C_k$  ;

    Déterminer  $\mathcal{D}_{3D}(a, b, c, \mu, \mu', e, e')$ , la droite discrète englobante optimale pour  $S_b$  ;  
    (\*)

$bSegment=0$  ;  $eSegment=k-1$  ;

$MBS_\nu = MBS_\nu \cup MBS(bSegment, eSegment, \nu)$ ;

**while**  $k < n - 1$  **do**

**while** les épaisseurs de 2 segments fous qui sont obtenus sur les projections des points de  $S_b$  sur les plans de coordonnées sont  $> \nu$  **do**

$S_b = S_b \setminus C_{bSegment}$  ;  $bSegment++$  ;

        Déterminer  $\mathcal{D}_{3D}(a, b, c, \mu, \mu', e, e')$ , la droite discrète englobante optimale pour  $S_b$  ; (\*)

**while** les épaisseurs de 2 segments fous qui sont obtenus sur les projections des points de  $S_b$  sur les plans de coordonnées sont  $\leq \nu$  **do**

$k++$ ;  $S_b = S_b \cup C_k$ ;

        Déterminer  $\mathcal{D}_{3D}(a, b, c, \mu, \mu', e, e')$ , la droite discrète englobante optimale pour  $S_b$  ; (\*)

$eSegment=k-1$  ;  $MBS_\nu = MBS_\nu \cup MBS(bSegment, eSegment, \nu)$ ;

**fin**

---

$R_\nu(C_k)$  du cercle circonscrit du triangle  $[C_l, C_k, C_r]$ , nous utilisons la formule suivante [HS98] (voir aussi la figure 2.11.a et la figure 2.12).

Supposons que  $s_1 = \|\overrightarrow{C_k C_r}\|$ ,  $s_2 = \|\overrightarrow{C_k C_l}\|$  et  $s_3 = \|\overrightarrow{C_l C_r}\|$ , alors

$$R_\nu(C_k) = \frac{s_1 s_2 s_3}{\sqrt{(s_1 + s_2 + s_3)(s_1 - s_2 + s_3)(s_1 + s_2 - s_3)(s_2 + s_3 - s_1)}}$$

Ainsi, la courbure d'épaisseur  $\nu$  en chaque point  $C_k$  est déterminée par :  $\mathcal{C}_\nu(C_k) = \frac{s}{R_\nu(C_k)}$  avec  $s = \text{sign}(\det(\overrightarrow{C_k C_r}, \overrightarrow{C_k C_l}))$  (il indique la concavité et la convexité de la courbe).

La valeur de la courbure est de zéro dans le cas les demi-tangentes sont colinéaires. Ce cas peut être testé indépendamment.

### 2.5.2 Estimation de la courbure d'épaisseur $\nu$ en chaque point $C_k$ de $C$

Dans cette section, nous proposons un nouvel algorithme pour déterminer *la courbure d'épaisseur*  $\nu$  en chaque point d'une courbe  $C$  de  $n$  points en 2D ou en 3D. La complexité de cette méthode est meilleure que l'algorithme existant [DR04] qui fonctionne en temps  $O(n^2)$ . Ces méthodes sont basées sur la détermination du cercle circonscrit qui passe par 3 points :  $C_k$  et 2 extrémités des segments fous les plus longs à gauche et à droite qui sont construits à partir de ce point.

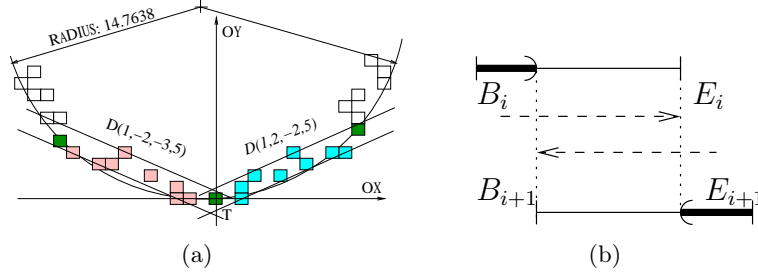


FIGURE 2.11 – (a) Estimation de la courbure 2D en point  $C_k$  à l'épaisseur 2; (b)  $E_i$  ( $B_{i+1}$ ) est le front (queue) du point dans le première (seconde) droite épaisse.

### Description de l'algorithme (voir les figures 2.11 et 2.12)

Supposons que  $MBS_R(k, r, \nu)$  et  $MBS_L(l, k, \nu)$  soient les segments flous les plus longs à gauche et à droite construits à partir du point  $C_k$ . Alors, il existe  $r' \leq k$  et  $l' \geq k$  tel que  $MBS_R(k, r, \nu) \subset MBS(r', r, \nu)$  et  $MBS_L(l, k, \nu) \subset MBS(l, l', \nu)$ .

Considérons la décomposition de la courbe  $C$  en une suite de segments flous d'épaisseur  $\nu$  :  $MBS_\nu(C) = \{MBS(B_1, E_1, \nu), MBS(B_2, E_2, \nu), \dots, MBS(B_m, E_m, \nu)\}$  avec  $B_1 < B_2 < \dots < B_m$  et  $E_1 < E_2 < \dots < E_m$ . Nous recherchons des indices  $i$  et  $j$  tels que  $i$  est le premier indice vérifiant  $E_i \geq k$  et que  $j$  est le dernier indice vérifiant  $B_j \leq k$ . Alors, il est évident que  $l = B_i$ ,  $r = E_j$ . De plus, la courbure d'épaisseur  $\nu$  en chaque point  $C_k$  est l'inverse du rayon du cercle circonscrit du triangle  $[C_l, C_k, C_r]$ . Nous avons le résultat ci-dessous.

**Proposition 2.5.1** *Considérons la suite de segments flous maximaux d'épaisseur  $\nu$  d'une courbe discrète  $C$  :  $MBS_\nu(C) = \{MBS(B_1, E_1, \nu), MBS(B_2, E_2, \nu), \dots, MBS(B_m, E_m, \nu)\}$ , alors nous avons :*

- $\forall k$  tel que  $E_{i-1} < k \leq E_i$ , alors  $F(k) = B_i$
- $\forall k$  tel que  $B_i \leq k < B_{i+1}$ , alors  $Q(k) = E_i$

Cette proposition est utilisée dans l'algorithme 8 (voir aussi la figure 2.11.b)

---

**Algorithme 8** : Estimation de la courbure d'épaisseur  $\nu$  en chaque point d'une courbe 2D ou 3D.

---

**Données** :  $\mathcal{C}$  Courbe discrète de  $n$  points,  $\nu$  l'épaisseur de segmentation

**Résultat** :  $\{\mathcal{C}_\nu(C_k)\}_{k=0..n-1}$  - Courbure d'épaisseur  $\nu$  en chaque point de la courbe  $\mathcal{C}$

**début**

Construire  $MBS_\nu = \{MBS_i(B_i, E_i, \nu)\}_{i=0 \text{ to } m-1}$  (voir l'algorithme 6 pour le cas 2D ou l'algorithme 7 pour le cas 3D);  $m = |MBS_\nu|$ ;  $E_{-1} = -1$ ;  $B_m = n$ ;

**pour**  $i = 0$  **to**  $m - 1$  **faire**

**pour**  $k = E_{i-1} + 1$  **to**  $E_i$  **faire**  $L(k) = B_i$ ;

**pour**  $k = B_i$  **to**  $B_{i+1} - 1$  **faire**  $R(k) = E_i$ ;

**pour**  $i = 0$  **to**  $n - 1$  <sup>1</sup> **faire**

$R_\nu(C_i) =$  Rayon du cercle circonscrit qui passe par 3 points  $[C_{L(i)}, C_i, C_{R(i)}]$ ;

$\mathcal{C}_\nu(C_i) = \frac{\text{sign}(\det(\overrightarrow{C_i C_{R(i)}}, \overrightarrow{C_i C_{L(i)}}))}{R_\nu(C_i)}$  ;

**fin**

---

1. La borne dans l'algorithme 8 est vraie pour une courbe fermée. Dans le cas d'une courbe non-fermée,



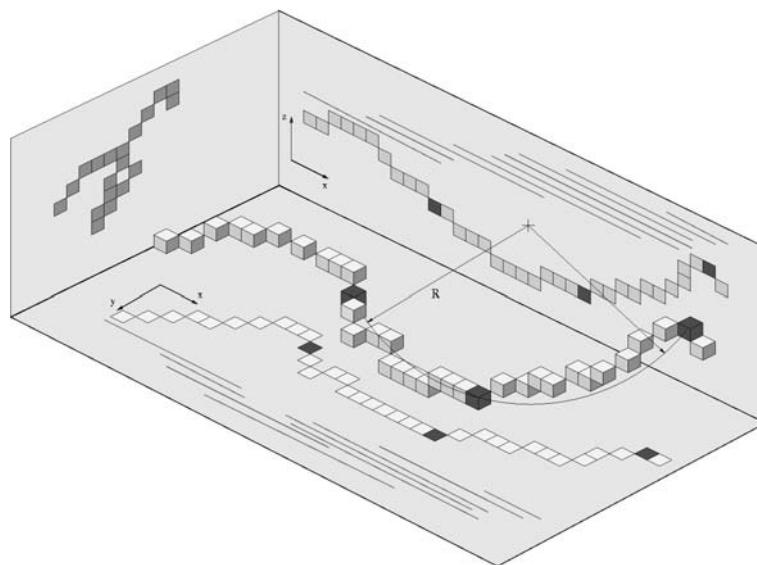


FIGURE 2.12 – Considérons une courbe 3D. Tout d’abord, la suite des segments flous maximaux est déterminée. Afin de déterminer la valeur de courbure en deuxième point en noir, son extrémité à gauche (resp. droite) est localisée comme l’extrémité à gauche (resp. droite) du segment flou maximal correspondant. Puis, la courbure est déterminée comme l’inverse du rayon du cercle circonscrit. L’épaisseur de travail est 2.

### Complexité

Les 2 étapes de labélisation et d’estimation de la courbure sont effectuées en un temps linéaire. Par contre, la détermination des segments flous maximaux est effectuée en un temps  $O(n \log^2 n)$ . Donc, la complexité de notre méthode est  $O(n \log^2 n)$ . Puisque la complexité de la reconnaissance d’un segment flou est en  $O(n)$  pour les courbes simples, et  $O(n \log n)$  pour les courbes générales [Buz05], la méthode existante [SDRW06] d’estimation de la courbure fonctionne en temps  $O(n^2)$  pour les courbes simples et  $O(n^2 \log^2 n)$  pour les courbes générales.

## 2.6 Torsion discrète d’épaisseur $\nu$

### 2.6.1 Définitions

La courbure 3D ne suffit pas pour découvrir toutes les propriétés d’une courbe 3D. Ce paramètre mesure seulement la vitesse de changement de direction de la courbe. Au niveau des courbes 2D, le plan osculateur ne change pas. Pour les courbes 3D, la torsion est un paramètre qui mesure la vitesse de variation de changement du plan osculateur. Pour clarifier cette notion, nous rappelons ci-dessous quelques définitions de géométrie différentielle (voir [Opr07] pour plus de détails).

**Définition 2.6.1** *Supposons que  $r : I = (a, b) \rightarrow \mathbb{R}^3$  est une courbe 3D régulière unitaire paramétrée par  $t$ .*

*[i]  $T(t)$  (resp.  $N(t)$ ) est un vecteur unitaire dans la direction  $r'(t)$  (resp.  $r''(t)$ ). Donc,  $N(t)$  est un vecteur normal à  $T(t)$ .  $T(t)$  (resp.  $N(t)$ ) est appelé le **vecteur tangent** (resp. **vecteur normal**) en  $t$ .*

---

l’instruction devient *Pour  $i = l$  to  $n - 1 - l$ ,  $l$  est une valeur constante.*

[ii]  $k(t) = |T'(t)|$  est appelé la **courbure** de  $r$  en  $t$ .

[iii] Le plan qui est construit par le vecteur tangent et le vecteur normal ( $T(t)$  et  $N(t)$ ), est appelé le **plan osculateur** en  $t$ . Le vecteur unitaire  $B(t) = T(t) \wedge N(t)$  est le vecteur normal du plan osculateur et il est appelé le **vecteur binormal** en  $t$ .

[iv]  $\tau(t) = |B'(t)|$  est appelé la **torsion** de la courbe en  $t$ .

**Définition 2.6.2** Supposons que  $r : I \rightarrow \mathbb{R}^3$  est une courbe 3D paramétrée par  $t$ .

i La courbure de  $r$  en  $t \in I : k(t) = \frac{|r' \wedge r''|}{|r'|^3}$

ii La torsion de  $r$  en  $t \in I : \tau(t) = \frac{(r' \wedge r'') \cdot r'''}{|r' \wedge r''|^2}$

Grâce à la définition 2.6.2, la valeur de la torsion en chaque point est 0 si la valeur de courbure en ce point est 0.

## 2.6.2 Torsion discrète

La torsion discrète est étudiée dans [LaDGJLC05, Mok97, MWOS04, KdF88].

Dans cette section, nous proposons une nouvelle approche géométrique [NDR08a, NDR08b] pour le problème de l'estimation de la torsion. Cette méthode utilise les définitions et les résultats qui sont présentés dans la section précédente.

### Définitions

Supposons que  $\zeta$  est une courbe discrète 3D,  $C_k$  est le  $k^{\text{ième}}$  point de la courbe. Nous considérons les points  $C_r$  et  $C_l$  de  $\zeta$  tels que :  $l < k < r$ ,  $BS(l, k, \nu) \& \neg BS(l - 1, k, \nu)$  et  $BS(k, r, \nu) \& \neg BS(k, r + 1, \nu)$ . Nous rappelons que la valeur de la courbure d'épaisseur  $\nu$  en  $C_k$  est déterminée par le cercle circonscrit qui passe par 3 points  $C_k, C_l, C_r$ . Si  $\overrightarrow{C_l C_k}$  et  $\overrightarrow{C_l C_r}$  sont colinéaires, la valeur de courbure au point  $C_k$  est 0, alors la valeur de la torsion à ce point est aussi 0. Donc, sans perte de généralité, nous supposons que ces vecteurs ne sont pas colinéaires. D'autre part, le plan qui est défini par  $\overrightarrow{C_l C_k}$  et  $\overrightarrow{C_l C_r}$ , noté  $(C_k, C_l, C_r)$ , contient non seulement le vecteur tangent mais aussi le vecteur normal de la courbe au point  $C_k$ . Donc, nous avons la définition ci-dessous.

**Définition 2.6.3** Le **plan osculateur d'épaisseur  $\nu$  au point  $C_k$**  est le plan  $(C_k, C_l, C_r)$

Le plan osculateur  $(C_k, C_l, C_r)$  a 2 vecteurs unitaires tangents  $\vec{t1} = \frac{\overrightarrow{C_k C_l}}{|\overrightarrow{C_k C_l}|}$  et  $\vec{t2} = \frac{\overrightarrow{C_k C_r}}{|\overrightarrow{C_k C_r}|}$ . Donc, le vecteur binormal de la courbe à ce point est défini comme  $\vec{b}_k = \vec{t1} \wedge \vec{t2}$ . Alors, nous donnons ci-dessous la définition de la torsion d'épaisseur  $\nu$ .

**Définition 2.6.4** La **torsion discrète d'épaisseur  $\nu$  au point  $C_k$**  est la dérivée de  $\vec{b}_k$

### ESTIMATEUR DE TORSION

Notre méthode proposée pour l'estimation de la torsion est basée sur la définition 2.6.4. Nous rappelons que la suite des vecteurs binormaux  $\{\vec{b}_k\}_{k=0}^{n-1}$  peut être construite à partir de la suite des segments flous maximaux en temps  $O(n \log^2 n)$ . C'est pourquoi, nous allons obtenir une estimation de la valeur de la torsion en chaque point de la courbe en calculant la dérivée en chaque position de la suite  $\{\vec{b}_k\}_{k=0}^{n-1}$ . La méthode traditionnelle pour l'estimation de la dérivée

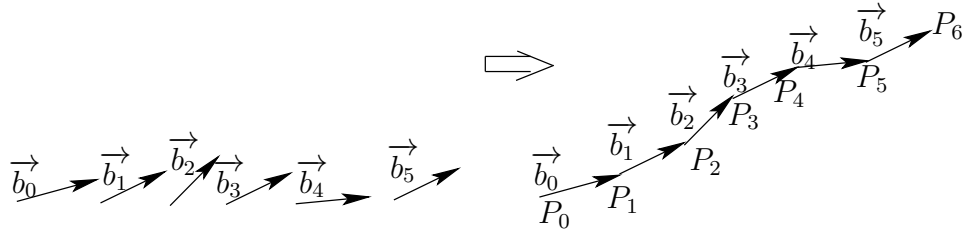


FIGURE 2.13 – La courbe  $\zeta_1$  est construite à partir de la suite des vecteurs binormaux.

d'une suite discrète est d'utiliser un noyau gaussien [WS93b]. Nous proposons ci-dessous une approche géométrique pour ce problème.

Nous considérons la courbe  $\zeta_1 = \{P\}_{i=0}^n$  qui est construite par cette règle :  $\overrightarrow{P_i P_{i+1}} = \vec{b}_i$ ,  $i = 0, \dots, n-1$  (voir la figure 2.13).

**Proposition 2.6.1** *Le vecteur tangent en chaque point  $P_i$  de la courbe  $\zeta_1$  est  $\vec{b}_i$  ( $i = 0, \dots, n-1$ )*

**Preuve 2.6.1** *En géométrie différentielle, le vecteur tangent de la courbe  $r(t)$  au point  $P_{t_0} = r(t_0)$  est défini comme suit.*

$$t(t_0) = r'(t_0) = \lim_{h \rightarrow 0} \frac{r(t_0 + h) - r(t_0)}{h} = \lim_{h \rightarrow 0} \frac{\overrightarrow{P_{t_0} P}}{h}$$

Donc, dans l'espace discrète, le vecteur tangent au point  $P_i = \alpha(i)$  peut être estimé comme  $t(i) = \frac{r(i+1) - r(i)}{1} = \frac{\overrightarrow{P_i P_{i+1}}}{1} = \overrightarrow{P_i P_{i+1}} = \vec{b}_i$

**Proposition 2.6.2** *La valeur de la torsion discrète d'épaisseur  $\nu$  en chaque point de la courbe  $C$  est estimée comme la valeur de courbure en chaque point de la courbe  $\zeta_1$*

**Preuve 2.6.2** *La torsion discrète au point  $C_k$  de la courbe  $\zeta$  est la dérivée de  $\vec{b}_k$ . D'autre part,  $\vec{b}_k$  est le vecteur tangent au  $k^{i\text{ème}}$  point de la courbe  $\zeta_1$ . Alors, cette valeur est aussi la valeur de courbure au  $k^{i\text{ème}}$  point de la courbe  $\zeta_1$*

Alors, grâce aux 2 propositions ci-dessus, nous pouvons estimer la valeur de torsion discrète d'épaisseur  $\nu$  en chaque point de la courbe  $\zeta$  en déterminant la valeur de courbure correspondant au point de la courbe  $\zeta_1$ . La méthode que nous proposons pour l'estimation de la torsion est présentée de façon plus détaillée dans l'algorithme 9.

## 2.7 Expérimentations et comparaisons

### 2.7.1 Expérimentations

Nous avons expérimenté ces méthodes sur la configuration d'ordinateur suivante : CPU Pentium 4 de vitesse 3.2GHz, 1G de mémoire RAM, noyau du système d'exploitation linux 2.6.22-14. Concernant les courbes non-fermées, les résultats ne sont pas fiables pour les parties du début et de fin (voir les bornes qui sont présentées dans les algorithmes 8 et 9). Nous utilisons aussi un paramètre de borne dans la phase d'estimation d'erreur pour éliminer cet effet.

2. La borne dans cet algorithme 9 est identique à celle de l'algorithme 8.

---

**Algorithme 9** : Estimation de la torsion d'épaisseur  $\nu$  en chaque point d'une courbe 3D  $\zeta$ .

---

**Données** :  $C$  Courbe discrète 3D de  $n$  points,  $\nu$  l'épaisseur de segmentation

**Résultat** :  $\{\mathcal{T}_\nu(C_k)\}_{k=0..n-1}$  - Torsion d'épaisseur  $\nu$  en chaque point de  $C$

**début**

Construire  $MBS_\nu = \{MBS_i(B_i, E_i, \nu)\}_{i=0 \text{ to } m-1}$  ;

$m = |MBS_\nu|$  ;  $E_{-1} = -1$  ;  $B_m = n$  ;

**pour**  $i = 0$  à  $m - 1$  **faire**

**pour**  $k = E_{i-1} + 1$  à  $E_i$  **faire**  $L(k) = B_i$  ;

**pour**  $k = B_i$  à  $B_{i+1} - 1$  **faire**  $R(k) = E_i$  ;

**pour**  $i = 0$  à  $n - 1$  <sup>2</sup> **faire**

$\vec{t1} = \frac{C_i C_{L(i)} \vec{C}_i}{|C_i C_{L(i)}|}$  ;  $\vec{t2} = \frac{C_i C_{R(i)} \vec{C}_i}{|C_i C_{R(i)}|}$  ;  $\vec{b}_i = \vec{t1} \wedge \vec{t2}$  ;

    Construire  $\zeta_1 = \{P_k\}_{k=0}^n$ , avec  $\overrightarrow{P_k P_{k+1}} = \vec{b}_k$  ;

    Estimer la valeur de la torsion discrète d'épaisseur  $\nu$  en chaque point de la courbe  $C$  en appliquant l'algorithme 8 sur la courbe  $\zeta_1$  ;

**fin**

---

## Mesure d'erreur

Nous utilisons 3 critères pour mesurer l'erreur : erreur relative moyenne (meanRE), erreur relative maximale (maxER) et erreur relative quadrique (ERQ). Nous considérons 2 suites : les résultats attendus  $\{IR_i\}_{i=1}^n$  et les résultats obtenus  $\{RR_i\}_{i=1}^n$  en chaque position de la courbe. Ensuite, nous définissons :

$$meanRE = \frac{1}{n} \sum_{i=1}^n \frac{|RR_i - IR_i|}{IR_i} \quad (2.1)$$

$$maxER = \max \left\{ \frac{|RR_i - IR_i|}{IR_i} \right\}, i = 1, \dots, n \quad (2.2)$$

$$ERQ = \sqrt{\frac{1}{n} \sum_{i=1}^n \left( \frac{|RR_i - IR_i|}{IR_i} \right)^2} \quad (2.3)$$

Ces critères sont modifiés à partir des critères classiques d'erreur. Ils nous permettent de mesurer comment le résultat obtenu respecte le profil du résultat idéal.

## Estimation de la courbure

Nous présentons dans la figure 2.14 quelques expérimentations de notre estimateur de courbure pour des courbes planaires. Deux courbes discrètes (voir les figures 2.14.a et .c) sont présentées avec leurs profils de courbure déterminés par notre méthode ((voir les figures 2.14.b et .d)). Les points de la courbe 2.14.c correspondent aux pics (carrés noirs) du profil de courbure 2.14.d indiqués par des pixels en noir. Nous trouvons que ces pixels noirs sont bien localisés aux coins de la courbe 2.14.c. Par conséquent, une application de détection des points dominants peut être déduite de notre méthode d'estimateur de courbure.

Pour les courbes 3D, la figure 2.17 présente quelques expérimentations effectuées avec notre estimateur de courbure 3D. Ces tests sont effectués sur les courbes discrètes non bruitées : hélice,

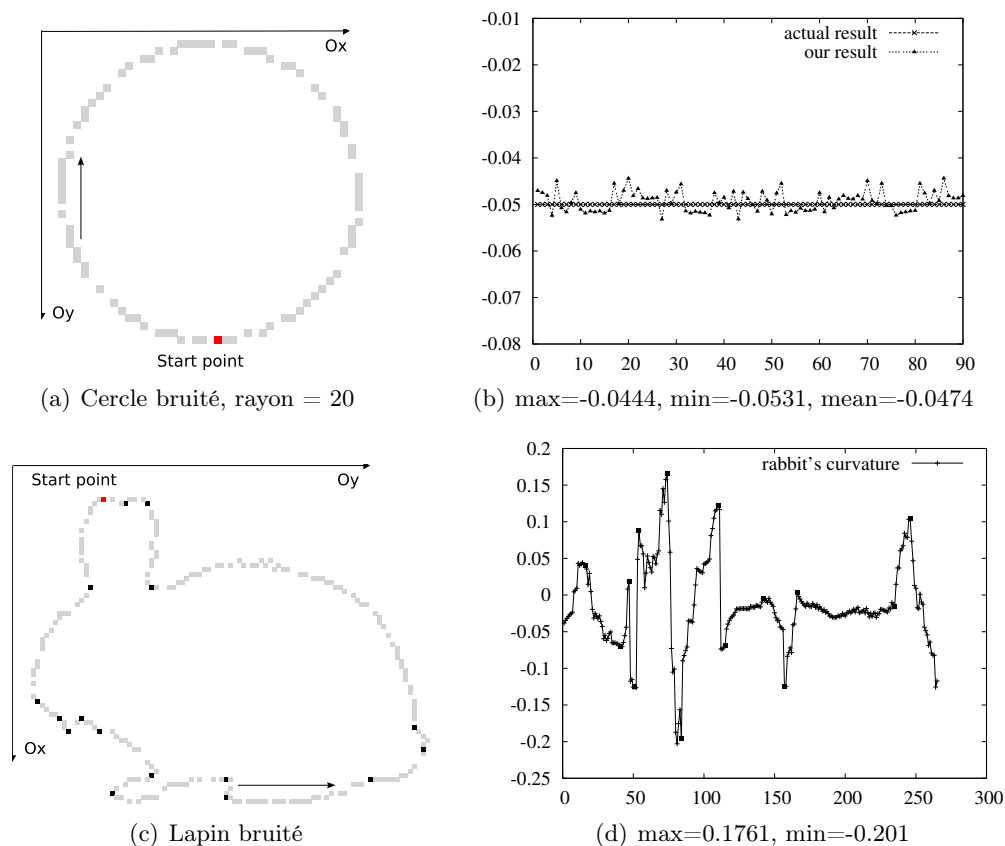


FIGURE 2.14 – Exemples d'extraction de la courbure 2D d'épaisseur 2 avec notre méthode : (a) Un cercle discret (rayon=20) - (b) le profil correspondant de courbure (c) Une courbe de lapin (d) Son profil de courbure.

Viviani, ...<sup>3</sup> avec leurs profils idéaux de courbure (la figure 2.17.b et 2.17.e). Les figures 2.17.c et 2.17.f sont des profils de courbure qui sont obtenus par notre estimateur.

En ce qui concerne l'erreur d'approximation, le tableau 2.1 présente les erreurs de nos méthodes par rapport aux résultats idéaux obtenus lors de tests effectués sur ces courbes ci-dessus.

<i>Courbes</i>	$N^0$	<i>Bornes</i>	<i>meanRE</i>	<i>maxER</i>	<i>ERQ</i>	<i>Temps</i>
Cercle	90	6	0.0186	0.1118	0.0316	100
Hélice	760	20	0.0074	0.0519	0.0132	610
Viviani	274	20	0.1185	0.7248	0.2198	200

TABLE 2.1 – Estimation d'erreur sur les résultats de courbure,  $N^0$  : nombre des points, le temps de calcul donné en ms.

### Estimation de la torsion

Nous présentons quelques expérimentations de notre méthode sur les courbes idéales 3D : hélice, Viviani, horoptère, et hyper hélice. Les tests sont effectués après avoir discrétisé les courbes

3. Math curves, <http://www.mathcurve.com/courbes3d/courbes3d.shtml>

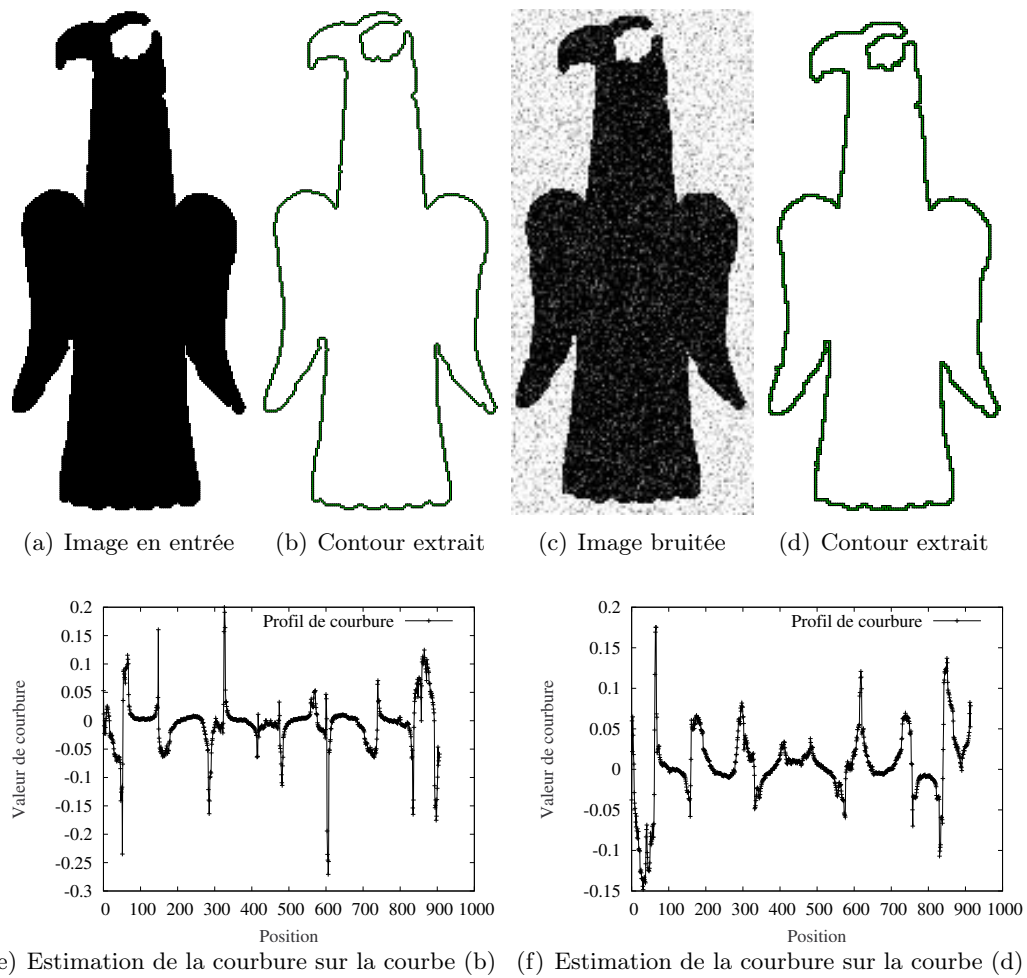
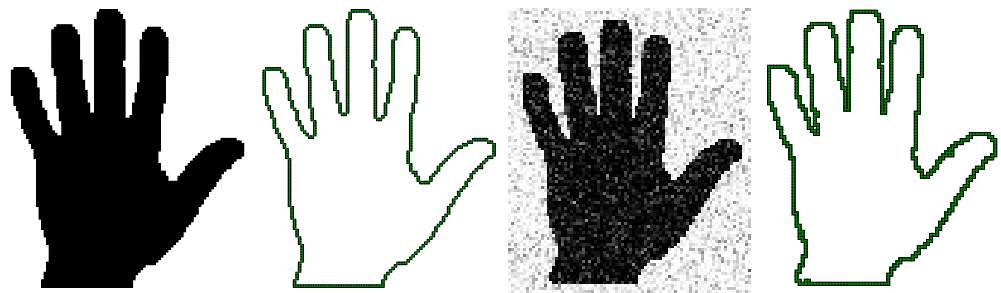
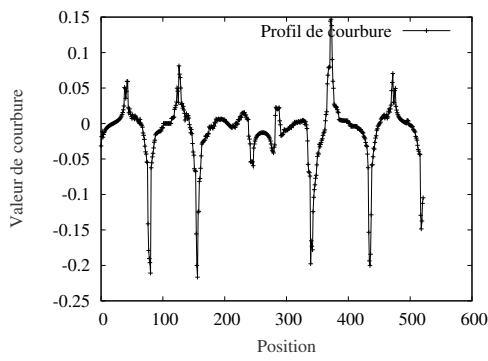


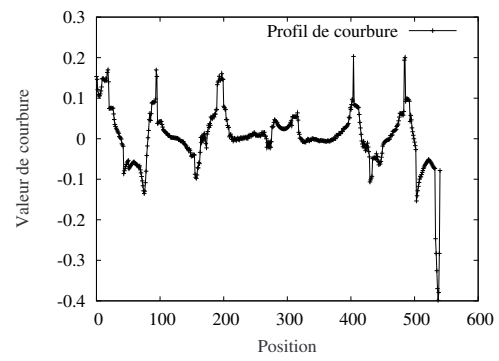
FIGURE 2.15 – Expérimentation sur les courbes bruitées. L'épaisseur de travail est 2.



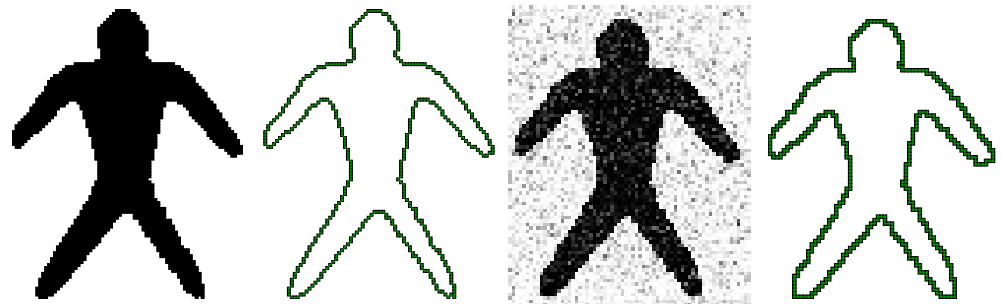
(a) Image en entrée (b) Contour extrait (c) Image bruitée (d) Contour extrait



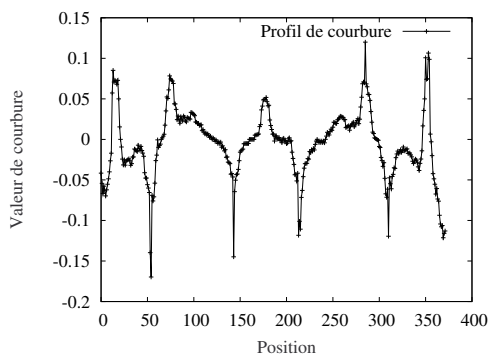
(e) Estimation de courbure sur la courbe (h)



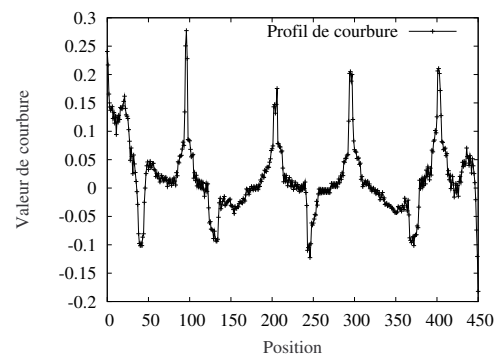
(f) Estimation de courbure sur la courbe (k)



(g) Image en entrée (h) Contour extrait (i) Image bruitée (j) Contour extrait



(k) Estimation de courbure sur la courbe (b)



(l) Estimation de courbure sur la courbe (d)

FIGURE 2.16 – Expérimentation sur les courbes bruitées. L'épaisseur de travail est 2.

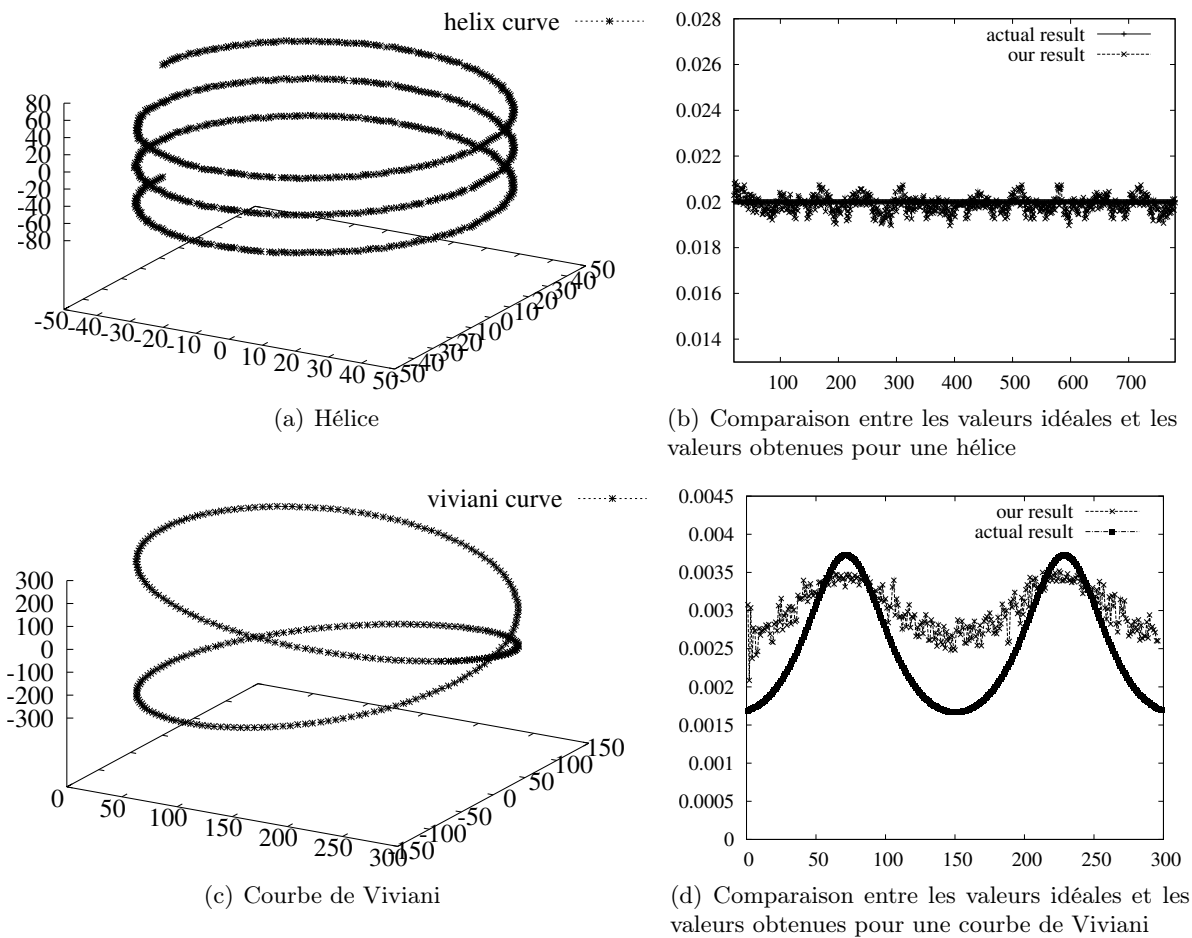


FIGURE 2.17 – Expérimentation sur l’estimation de la courbure, l’épaisseur de travail  $\nu = 2$ .

3D (voir figure 2.18).

Dans la plupart des cas qui sont présentés dans le tableau 2.2 et la figure 2.18, l’erreur relative moyenne ne dépasse pas 0.15, et l’erreur relative quadrique ne dépasse pas 0.015. En chaque point, si la valeur de torsion idéale de la courbe d’entrée s’approche de 0, le résultat de l’estimation n’est pas bon. Considérons le cas de la courbe de Viviani dans le tableau 2.2 (sans seuil) et dans la figure 2.20.a. Dans ce cas, l’erreur relative maximale est haute (15.6036) tandis que l’erreur relative moyenne est acceptable (0.628899). Dans un cas particulier, si la plus grande partie de la courbe en entrée a une valeur de torsion idéale qui s’approche de 0, le résultat de l’estimation est beaucoup moins précis (voir figure 2.19, figure 2.20.v).

Comme les autres méthodes d’approximation, notre méthode ne fonctionne pas très bien lorsque la valeur de torsion idéale s’approche de 0. La raison vient de la formule d’erreur relative. La divergence entre la valeur idéale et la valeur d’estimation est réduite lentement lorsque la valeur idéale s’approche de 0. Alors, l’erreur relative est plus haute dans cette situation. Considérons le cas de la courbe hyper-hélice (voir la figure 2.19). Le problème est que l’estimation de la torsion n’est pas bonne pour les valeurs qui s’approche de 0. Malgré que la valeur d’estimation s’approche aussi de 0, pourtant l’erreur relative est très haute. Dans la figure 2.20.b, nous montrons la relation entre la torsion estimée et la torsion idéale d’une courbe d’hyper-hélice entre les indices 15 et 250. Dans cet intervalle d’indices, la torsion idéale s’approche de 0. C’est pourquoi,



<i>Courbes</i>	$N^0$	$N^1$	<i>meanRE</i>		<i>maxER</i>		<i>ERQ</i>		<i>Time</i>
			Sans	Avec	Sans	Avec	Sans	Avec	
Sphérique	255	30	0.1317	0.1317	0.3428	0.3428	0.1642	0.1642	280
Horoptère	239	30	0.0827	0.0827	0.1862	0.1862	0.0979	0.0979	300
Hélice	760	30	0.0481	0.0481	0.5145	0.5145	0.0814	0.0814	920
Viviani	274	30	0.6289	0.4570	15.6036	3.5128	1.6228	0.6156	290
Hyper hélice	740	30	8551.24	1.1527	154378	3.8019	24704.2	1.6210	720

TABLE 2.2 – Mesure de l’erreur sur les résultats d’estimation de la torsion en utilisant notre méthode sans (avec) un seuil.  $N^0$  - nombre des points,  $N^1$  - borne, temps est calculé en ms. Dans la seconde ligne, sans : sans seuil, avec : avec un seuil = 0.0005.

<i>Estimateur</i>	$N^0$	$N^1$	<i>meanRE</i>		<i>maxER</i>		<i>ERQ</i>		<i>Temps</i>
			Sans	Avec	Sans	Avec	Sans	Avec	
Courbure	274	30	0.5191	0.5010	1.0618	1.0085	0.7355	0.7098	NA
Torsion	274	30	0.5367	0.4987	1.2160	1.0036	0.7615	0.7066	NA

TABLE 2.3 – Estimation d’erreur sur les résultats d’estimation de la torsion en utilisant la méthode de Lewiner [LaDGJLC05] sur la courbe de Viviani sans (avec) un seuil.  $N^0$  - nombre des points,  $N^1$  - borne, temps est calculé en ms. Dans la seconde ligne, sans : sans seuil, avec : avec un seuil = 0.0005.

l’erreur relative entre la torsion estimée et la torsion idéale est très importante, malgré le fait que la valeur estimée de la torsion ne dépasse pas à 0.006. Alors, nous proposons de considérer seulement les points dont la valeur de torsion idéale est supérieure à un seuil. Dans les questions 1, 2 et 3,  $n$  est remplacé par le nombre de points dont la valeur de torsion idéale est supérieure à un seuil. La tableau 2.2 montre aussi l’erreur d’approximation avec un seuil de 0.0005.

## 2.7.2 Comparaisons avec les méthodes existantes

Nous comparons nos estimateurs avec les méthodes connues de la littérature.

Les figures 2.21, 2.22 et les tableaux 2.3, 2.4 présentent des comparaisons avec les méthodes de Lewiner [LaDGJLC05] et de Mokhtarian [Mok97]. On peut voir que le résultat de la méthode de Mokhtarian dépend beaucoup de la valeur du paramètre  $\sigma$ . Lorsque la valeur de  $\sigma$  est plus petite, le résultat est très bruité. Pourtant, une grande valeur de  $\sigma$  mène à des résultats inexacts

<i>Courbes</i>	$N^0$	<i>Borne</i>	<i>Sigma</i>	<i>meanRE</i>	<i>maxER</i>	<i>ERQ</i>	<i>Temps</i>
Hyper hélice	740	30	4	0.451744	5.39524	0.796259	200
Sphérique	255	30	9	0.32234	0.852819	0.47207	140
-	-	-	10	0.315752	0.843035	0.461791	220
-	-	-	12	0.302302	0.962883	0.442564	290
Viviani	274	30	7	0.466501	0.999349	0.664052	100
-	-	-	9	0.457997	0.996809	0.653339	140
-	-	-	12	0.451581	0.994291	0.643084	300

TABLE 2.4 – Estimation d’erreur sur les résultats d’estimation de la torsion en utilisant la méthode de Mokhtarian [Mok97].  $N^0$  - nombre des points,  $N^1$  - borne, temps est calculé en ms. L’implémentation est développée et effectuée sur Matlab 7.5.0.

et l'effet de marge dans lequel les résultats sont totalement différents par rapport aux résultats idéaux. Un avantage de notre méthode, comme celle de Mokhtarian, est qu'elle n'utilise qu'un seul paramètre d'épaisseur, la méthode de Mokhtaria utilise aussi 1 paramètre  $\sigma$ . La méthode de Lewiner utilise 2 paramètres différents ( $\sigma$  et  $q$ ). Le résultat d'estimation de la torsion semble plus bruité que celles des autres. En ce qui concerne la qualité des résultats d'estimation, notre méthode est meilleure que les 2 autres méthodes. Pourtant, parmi 3 ces méthodes, la méthode de Mokhtarian est la plus performante grâce à la complexité linéaire.

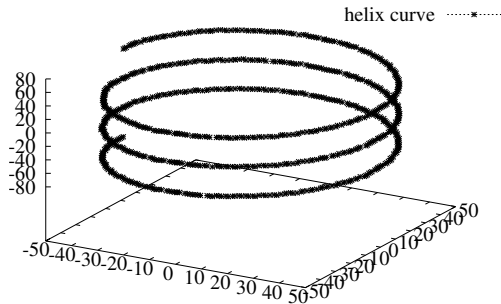
## 2.8 Conclusions et perspectives

Nous avons présenté dans ce chapitre une méthode d'estimation de la courbure pour les courbes 2D et pour les courbes 3D. Nous avons proposé aussi une méthode d'estimation de la torsion pour les courbes 3D. Ces méthodes exploitent les avantages de l'utilisation des segments flous maximaux [NDR07] et sont plus efficaces que les méthodes existantes [DR04], [SDRW06] qui utilisent simplement les méthodes [DRFRD06, Buz05] de reconnaissance de segment flou. Ces résultats ont été publiés dans les publications suivantes [NDR07, NDR08a, NDR08b].

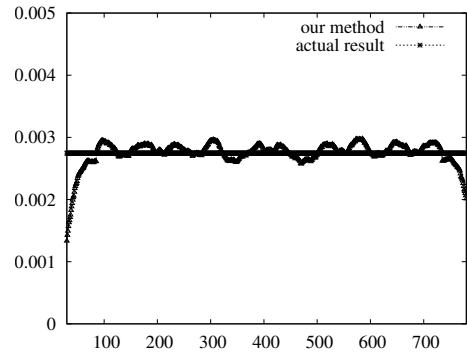
Ces estimateurs nous permettent d'extraire des propriétés locales des courbes 2D et 3D. Nous espérons utiliser ces méthodes pour identifier et classifier des objets 2D, 3D. Nos travaux peuvent être utilisés pour une application aux fibres dans les images 3D de papier [CS03]. Les autres applications peuvent être la compression géométrique [LaDGJLC05] ou bien l'image médicale [MWOS04]. Dans le futur, nous essayerons d'appliquer ces estimateurs pour la mise en correspondance des courbes d'ADN.

## Remerciements

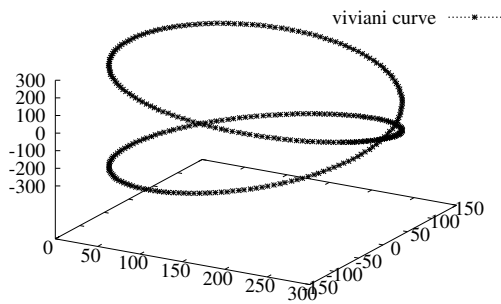
Nous tenons à remercier Thomas Lewiner pour m'avoir fourni son programme d'estimation de la courbure et de torsion qui nous a permis de faire des comparaison entre les deux méthodes.



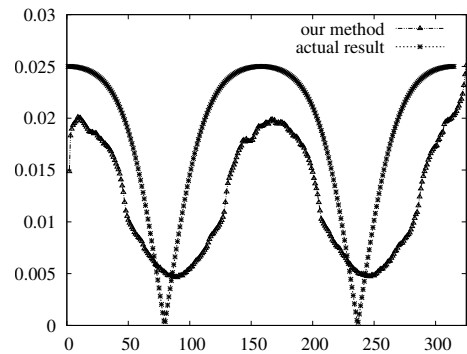
(a) Hélice



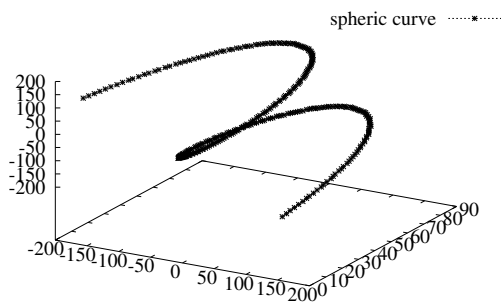
(b) Comparaison entre la valeur idéale et la valeur obtenue pour l'hélice



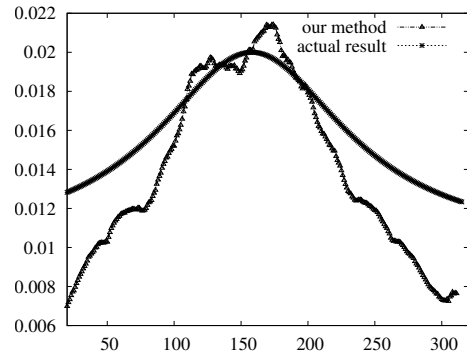
(c) Courbe de Viviani



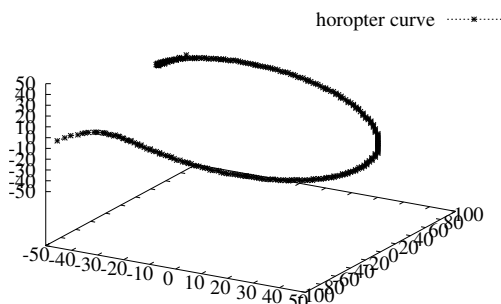
(d) Comparaison entre la valeur idéale et la valeur obtenue pour la courbe Viviani



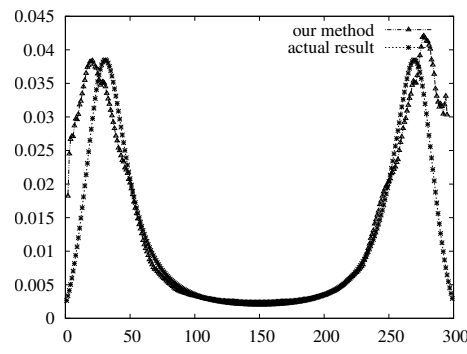
(e) Courbe sphérique



(f) Comparaison entre la valeur idéale et la valeur obtenue pour la courbe sphérique



(g) Horopter curve



(h) Comparaison entre la valeur idéale et la valeur obtenue pour la courbe horoptère

FIGURE 2.18 – Expérimentation sur l'estimation de la torsion, l'épaisseur de travail  $\nu = 2$ .

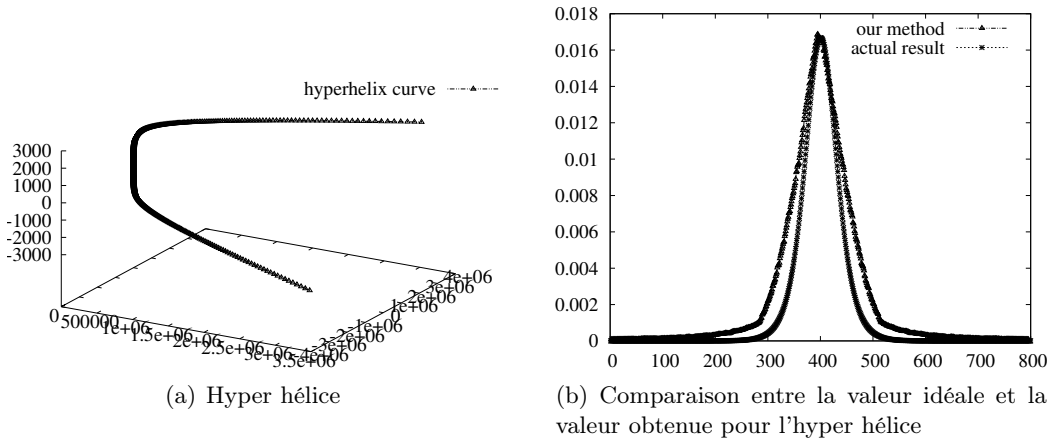


FIGURE 2.19 – Le pire cas d'estimation de la torsion : la majorité des valeurs de la courbe a la valeur de torsion idéale qui s'approche de 0..

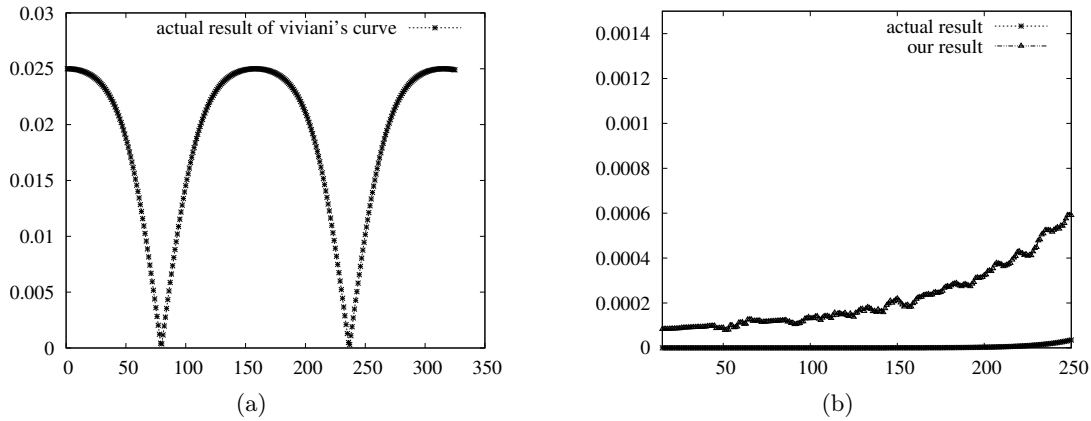


FIGURE 2.20 – a :Torsion de la courbe de Viviani; b :La torsion idéale et l'estimation de la torsion par notre méthode sur une courbe d'hyper hélice. Les indices sont entre 15 et 250.

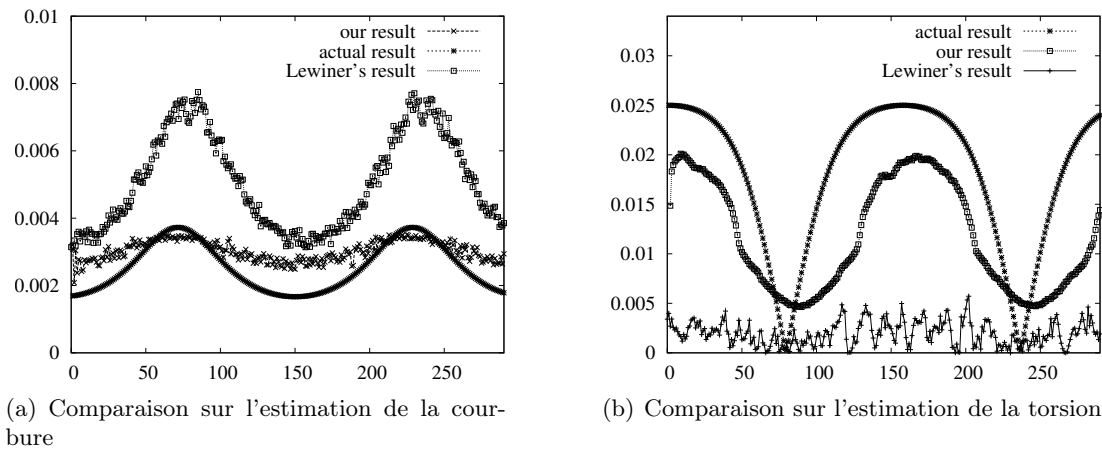


FIGURE 2.21 – Comparaisons entre nos estimateurs de courbure, de torsion et ceux de méthode de Lewiner sur la courbe de Viviani. Les résultats de Lewiner sont fournis par l'auteur.

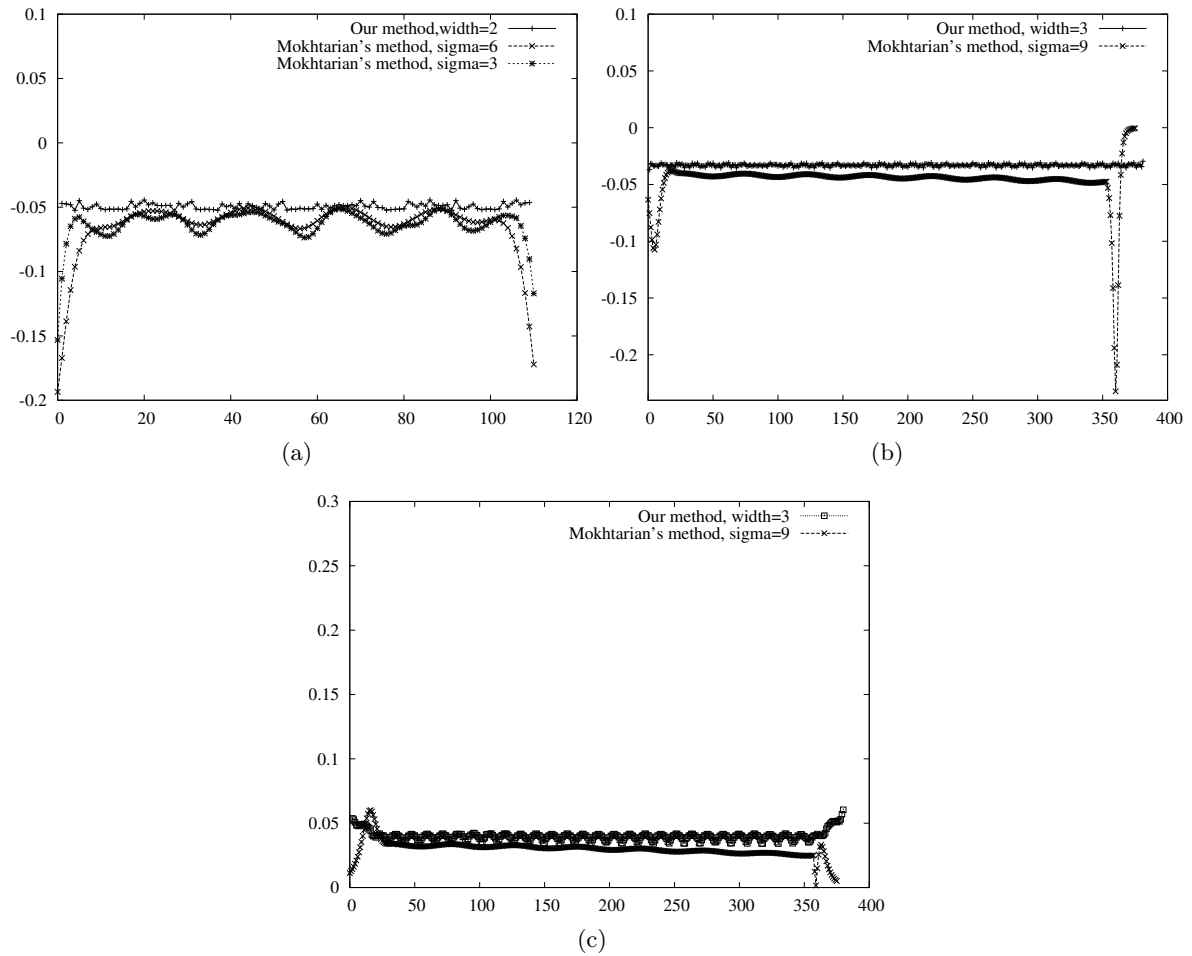


FIGURE 2.22 – a. Comparaison sur l'estimation de la courbure 2D entre notre méthode et la méthode de Mokhtarian. Le test est effectué sur un cercle discret dont le rayon est 20. ; b (resp. c) : Comparaison sur l'estimation de la courbure 3D (resp. torsion) entre notre méthode et la méthode de Mokhtarian. Le test est effectué sur la courbe d'hélice.



## Chapitre 3

# Détection de points dominants et représentation polygonale

### Sommaire

---

<b>3.1</b>	<b>Introduction</b>	<b>63</b>
<b>3.2</b>	<b>Etat de l'art</b>	<b>64</b>
3.2.1	Les méthodes de détection de points dominants	64
3.2.2	Les approches multi-échelle, multi-épaisseur	68
<b>3.3</b>	<b>Critères d'évaluation</b>	<b>68</b>
3.3.1	Les critères existants	68
3.3.2	Un nouveau critère alternatif d'évaluation pour les courbes bruitées	69
<b>3.4</b>	<b>Détection de points dominants avec un paramètre d'épaisseur <math>\nu</math></b>	<b>72</b>
3.4.1	Points dominants et région de support (ROS)	72
3.4.2	Relation entre les points dominants et la séquence de segments flous maximaux	73
3.4.3	Algorithme proposé	74
3.4.4	Expérimentations	76
3.4.5	Discussion sur le nouveau critère d'évaluation	86
<b>3.5</b>	<b>Approches multi-épaisseurs</b>	<b>89</b>
3.5.1	Une méthode multi-épaisseur pour la représentation polygonale des courbes bruitées	89
3.5.2	Méthode adaptative multi-échelle reposant sur une stratégie de découpage-et-fusion	90
3.5.3	Expérimentations	93
<b>3.6</b>	<b>Conclusions</b>	<b>98</b>

---

### 3.1 Introduction

La notion de point dominant (DP) est bien connue dans le domaine de la reconnaissance de formes en particulier dans la représentation des courbes. Le rôle le plus important des points dominants est qu'ils sont des points clés pour la décomposition d'une courbe en parties significatives. C'est une étape très importante dans l'analyse de forme. Dans ce chapitre, nous allons présenter de nouvelles méthodes pour la détection de points dominants en se basant sur les résultats récents du domaine de la géométrie discrète.

La première méthode est proposée pour la détection de points dominants à une échelle précise. Elle a besoin d'un seul paramètre qui est l'épaisseur des segments flous. Ce paramètre correspond à l'échelle à laquelle on travaille. Notre méthode peut naturellement s'appliquer aux courbes générales qui peuvent être bruitées ou bien déconnectées. Cette méthode compte sur l'étude de la structure géométrique de la courbe grâce à la décomposition de la courbe en segments flous maximaux d'épaisseur fixée qui a été présentée dans le chapitre précédent. Un processus de balayage est appliqué pour déterminer les zones communes des segments flous dont les pentes sont croissantes ou décroissantes. Chaque zone contient un point dominant qui sera détecté comme le point central de cette zone. Grâce aux avantages du segment flou, notre méthode peut également s'appliquer à des échelles différentes de la courbe.

La deuxième approche présentera une méthode multi-épaisseur afin de déterminer l'épaisseur de travail la plus appropriée en proposant un nouveau critère d'évaluation pour comparer des méthodes de détection de points dominants ou bien des méthodes de polygonalisation. Dans un cadre multi-épaisseur, nous considérons les résultats qui sont obtenus par la méthode précédente à travers plusieurs épaisseurs de travail. L'épaisseur la plus appropriée sera déterminée par un critère d'évaluation.

Enfin, nous proposerons une troisième méthode multi-échelle adaptative reposant sur des critères d'évaluation pour décomposer une courbe en parties significatives. Reposant sur la première méthode, une stratégie de découpage-et-fusion est utilisée afin de décomposer une courbe en parties significatives avec les points dominants correspondants. Pour chaque partie, nous allons déterminer l'échelle la plus appropriée pour la représenter.

Le reste de ce chapitre est organisé comme suit. La prochaine section présentera un état de l'art concernant le problème de détection de points dominants. La section 3.3 présentera des critères d'évaluation pour évaluer des méthodes de détection de points dominants. Les sections 3.4, 3.5 présenteront 3 nouvelles méthodes concernant le problème de détection de points dominants et la représentation polygonale des courbes discrètes. A la fin de chaque session, nous présenterons les expérimentations de ces 3 méthodes.

## 3.2 Etat de l'art

### 3.2.1 Les méthodes de détection de points dominants

Le travail concernant la détection de points dominants a commencé depuis la découverte de Attneave [Att54] qui a montré que les points de la courbe dont les valeurs de courbure sont localement maximales contiennent beaucoup d'informations utiles sur cette courbe. Ils sont alors suffisants pour la caractériser : le polygone, construit à partir de ces points, donne une forme d'approximation de cette courbe. Une méthode de détection de points dominants peut mener à une bonne représentation d'une courbe planaire. De plus, une représentation de forme basée sur la détection de points dominants a plusieurs avantages. Tout d'abord, il y a une réduction de données lorsqu'on sauvegarde cette représentation au lieu d'utiliser la courbe originale. Deuxièmement, cette représentation se concentre sur les caractéristiques principales de la forme de cette courbe, elle est donc utile à utiliser dans la reconnaissance des formes, l'extraction des caractéristiques ou bien la décomposition d'une courbe en parties significatives. Par conséquent, ces points dominants jouent un rôle crucial dans l'approximation de courbes, la reconnaissance des formes et l'indexation d'images [ZHG06]. Ils ont aussi des applications dans les autres domaines de la vision par ordinateur tels que la compression des données de vecteur [Kol07]. A partir du travail de Attneave, plusieurs méthodes qui ont été proposées sur la détection de points dominants. En ce qui concerne ce sujet, quelques problèmes se sont présentés : temps de calcul, nombre des



paramètres, sélection du point de départ, résultats mauvais avec des courbes bruitées...

En général, nous pouvons classer les méthodes existantes en 2 groupes. Le premier contient des *méthodes directes* qui déterminent directement des points dominants tels que les points de valeur maximale de courbure. Ces méthodes utilisent des mesures de courbure [KLN08], ou bien des mesures alternatives de courbure telles que le *k-cosinus* [RJ73], la région de support [CR89]. Les *méthodes indirectes* sont souvent basées sur l'approximation polygonale de la courbe, les points dominants sont déduits après cette étape. Dans ces méthodes les points dominants sont détectés comme les sommets du polygone approximé. De plus, nous pouvons diviser les méthodes de l'approximation polygonale en 3 approches principales : l'approche séquentielle, l'approche de découpage-et-fusion, l'approche de recherche heuristique. Dans les parties suivantes, nous rappelons quelques méthodes populaires. Un panorama complet sur les méthodes de détection de points dominants peut être consulté dans la thèse de Marji [Mar03].

### Les méthodes directes de détection de points dominants

**Rosenfeld et Johnston [RJ73]** : Rosenfeld et al. ont utilisé le cosinus de l'angle entre 2 arcs de longueur  $k$  de chaque côté d'un point d'une courbe comme une mesure alternative de courbure. Cette mesure est appelée *k-cosinus*. Les extrémités à gauche et à droite de chaque point  $p_i$  sont déterminées en vérifiant l'angle  $\angle p_{i-(m-j)}p_i p_{i+(m-j)}$  où  $m = \frac{N}{10}$ ,  $N$  est le nombre des points de la courbe,  $j = \{0, \dots, h\}$ . Cette méthode est simple. Pourtant, en fixant  $m = \frac{N}{10}$  cette méthode n'est pas une approche naturelle car les extrémités à gauche et à droite sont considérées dans une fenêtre de taille fixe et on a besoin d'une hypothèse de symétrie de ces points. De plus, le résultat dépendra de la forme de la courbe.

**Teh et Chin [CR89]** : Teh et al ont proposé une méthode non-paramétrique pour la détection de points dominants. Ils ont utilisé la région de support (ROS) comme une mesure significative de courbure qui est déterminée en chaque point  $p_i$  grâce à ses propriétés locales (la corde  $p_{i-k}p_{i+k}$  et la distance perpendiculaire à partir de  $p_i$  vers cette corde). Enfin, les points dominants sont détectés par un processus qui détecte des pics maximaux. Ils ont aussi conclu dans ce travail que la performance d'un algorithme de détection de points dominants dépend non seulement de l'exactitude de la mesure significative mais aussi de la détermination précise de la ROS. Cette méthode est le premier travail qui utilise la région de support comme une mesure significative de courbure pour la détection de point dominant. La méthode est simple et assez rapide. Pourtant, l'utilisation de l'hypothèse associée à la symétrie d'une région de support ne donne pas un calcul exact et naturel de cette dernière.

**Ansari et Huang [AwH91]** : Ansari et al. ont proposé une méthode non-paramétrique pour la détection de points dominants. Ils ont utilisé l'algorithme de Teh [CR89] pour estimer la région de support de chaque point de contour. Puis, chaque point est lissé par un filtre Gaussien dont l'épaisseur est proportionnelle à la région de support de ce point. Le *k-cosinus* est utilisé comme mesure et les points dominants sont obtenus par un processus reposant sur la suppression de points qui ne sont pas maximums. Ansari a montré aussi que les détecteurs de points dominants ne peuvent pas être comparés simplement en se basant sur l'erreur d'approximation. L'avantage de cette méthode est qu'elle ne demande aucun paramètre pour détecter des points dominants.

**Arcelli et Ramella [AR93]** : Arcelli et al. ont proposé une procédure itérative afin de localiser les points dominants. L'ensemble initial  $V_0 = \{v_1, v_2, \dots\}$  est construit comme l'ensemble des points dont le code différentiel (la différence de code de deux points voisins) n'est pas zéro. Si le point de contour  $p_x = v_{x\pm 1}$  a le code différentiel plus grand que celui de  $v_x$ , alors  $p_x$  remplace  $v_x$  dans l'ensemble  $V_0$ . La mesure de chaque sommet dans l'ensemble  $V_0$  est déterminée par la géométrie (la superficie, l'angle et la hauteur) du triangle qui est formée de ce sommet et des deux

sommets voisins dans cet ensemble. Afin d'obtenir l'ensemble  $V_k$  à l'itération  $k$ , chaque sommet dans l'ensemble  $V_{k-1}$  est évalué dans le contexte de la forme générale en consultant la valeur moyenne de la superficie et le cosinus de l'angle correspondant au triangle à chaque sommet dans l'ensemble  $V_{k-1}$ . Dans cette méthode, le nombre de points choisis diminue la taille de l'itération d'itération, et le processus de sélection converge vers un ensemble de points dominants espérés. Le polygone obtenu en reliant les points successifs se rapproche du contour de façon intuitive.

**Sarkar [Sar93]** : Sarkar a proposé une méthode simple pour la détection de points dominants à partir d'un traitement du code de Freeman [Fre61] d'un contour. A partir du premier point de la courbe, cette méthode reposant sur un traitement des codes de Freeman examine consécutivement le code différentiel des points et détecte les points dominants. Si le code différentiel  $c_i$  du point  $p_i$  est 0,  $p_i$  n'est pas un point dominant. Si  $c_i \in \{2, 3, 5, 6\}$ ,  $p_i$  est un point dominant. Si  $c_i \in \{1, 4, 7\}$ , quelques étapes supplémentaires seront appliquées pour déterminer l'importance du point  $p_i$ . L'avantage de cette méthode est qu'elle n'utilise aucun paramètre en entrée. Sarkar a proposé aussi dans ce travail un critère d'évaluation pour considérer des méthodes de détection de points dominants. Ce critère introduit repose sur un compromis entre l'erreur d'approximation et le taux de compression. C'est le premier critère dans la littérature en ce qui concerne l'évaluation des méthodes de détection de points dominants.

**Cronin [Cro99]** : Cronin a assigné un code spécial à chaque point basé sur le code Freeman de ses deux voisins proches. L'ensemble initial de points dominants est construit comme l'ensemble des points dont les codes de chaînes différentielles ne sont pas zéro. Un processus d'élimination est appliqué ensuite pour éliminer les candidats les moins susceptibles d'être des points dominants à partir de cet ensemble. Ces candidats correspondent à des séquences pré-définies. L'algorithme se termine lorsque le contour est examiné pour rechercher toutes les séquences pré-définies. L'inconvénient de cette méthode est qu'elle ne travaille qu'avec des courbes fermées.

**Marji et Siy [MS03]** : Marji et al. ont déterminé la région de support (ROS) en utilisant l'intégrale de l'erreur quadratique. Une région de support en chaque point se compose de 2 bras de support à gauche et à droite qui sont des segments de droite construite à partir de ce point. Ils ont sélectionné des points de l'extrémité du bras de support comme points dominants reposant sur la fréquence de leur sélection. Cette méthode est non-paramétrique, elle n'a besoin aucun paramètre en entrée. De plus, le polygone approximé maintient la symétrie de la forme.

## Les méthodes indirectes de détection de points dominants

### Approche séquentielle

**Ray et Ray [RR93]** : Ray et Ray ont proposé une technique pour déterminer le polygone optimal d'approximation d'une courbe. Ils ont déterminé les segments de droite les plus longs possible avec une erreur minimale. La norme  $L_1$  est utilisée pour mesurer la proximité entre un polygone et la courbe en entrée. Cet algorithme est effectué en temps linéaire.

**Yohisuke Kurozumi et Wayne A. Davis [KD82]** : Kurozumi et Davis ont proposé une méthode d'approximation qui minimise la distance maximale entre la courbe donnée en entrée et les lignes approximées. Cette méthode s'exécute en temps  $O(n \log n)$ . Le polygone satisfait la condition où le nombre de segments est minimal et l'erreur d'approximation est inférieur à un seuil donné.

**Aoyama [AK91]** : a utilisé un balayage linéaire pour évaluer des conditions d'erreur. Si les conditions ne sont pas satisfaisantes, une nouvelle recherche de nouveaux segments est appliquée. Le problème de cette méthode est que parfois les sommets ne correspondent pas aux coins de la courbe puisqu'un nouveau vecteur est défini seulement quand les conditions sont violées.

La méthode comporte quatre avantages :

- Elle peut efficacement approximer des figures dans lesquelles coexistent des portions de courbes à la fois grandes et petites.
- Les coins peuvent être conservés.
- Les zones plates sont conservées.
- Les droites dans l'image initiale sont directement adoptées comme des segments de droite dans le polygone approximé.

### Approche découpage-et-fusion

Dans cette approche, les lignes de droite sont adaptées à une segmentation initiale, et ensuite le moindre carré de l'erreur est calculé. Ensuite, ces méthodes utilisent une procédure itérative qui coupe une ligne si l'erreur est trop grande et fusionne 2 points si l'erreur est suffisamment petite. Les résultats d'approximation de cette approche peuvent être loin du résultat optimal si une segmentation initiale mauvaise est utilisée.

**Ramer [U.R72]** : Ramer a présenté une méthode récursive qui commence avec une segmentation initiale du contour. A chaque itération, le segment est divisé à partir du point dont la distance entre cette partie de la courbe et le segment est maximale si l'approximation d'erreur est supérieure à un seuil pré-défini. Ramer a montré que la complexité de la méthode est augmenté beaucoup si on essaie de réduire le nombre des points dominants. Donc, étant donné un seuil pré-défini d'erreur, le nombre des points dominants détectés n'est pas minimum.

**Sarfraz et al. [SAM04]** : Sarfraz et al. ont proposé un algorithme récursif dans lequel le segment de droite le plus long est déterminé avec un seuil d'erreur d'approximation pré-défini. Les points communs sont marqués comme les points dominants, et la partie de la courbe est divisée en sous-segment à partir de chaque point dominant.

**Held et al. [HAA94]** : Held et al. ont proposé une technique de découpage-et-fusion dans laquelle la différence de pente est utilisée pour diviser les segments et ceux-ci sont fusionnés en se basant sur un critère de mesure perceptive. L'algorithme n'a besoin d'aucun paramètre et fonctionne bien sur une grande variété de formes, et n'est pas sensible au changement d'échelle ou à la rotation.

### Approche heuristique

Ce type d'approche est utilisée pour éviter la complexité lourde de la polygonalisation optimale. L'algorithme génétique [Yin98], la recherche tabou [Yin00], l'algorithme de colonies de fourmis [Yin03], l'argument flou [LC99] sont quelques techniques connues dans cette catégorie. Les algorithmes heuristiques sont souvent coûteux. De plus, il n'est pas assuré que ces algorithmes donnent un résultat optimal.

Parmi ces 3 approches, l'approche successive est simple et rapide, mais la qualité de ses résultats dépend du point de départ. Récemment, Masood [Mas08] a proposé une nouvelle méthode qui ne tombe pas sur ces 3 approches. Cette technique simple et efficace est basée sur l'extraction des points de repos. Un point de repos est un point dont le code de Freeman est différent par rapport à celui du point précédent dans la courbe. L'ensemble des points de repos est considéré comme l'ensemble initial des points dominants. Chaque point dominant  $DP_j$  est assigné par un AVE (la valeur de l'erreur associée) correspondant qui est la distance perpendiculaire maximale des points qui se trouvent entre  $DP_{j-1}$  et  $DP_{j+1}$  vers  $DP_{j-1}DP_{j+1}$ . Ensuite, un processus itératif d'élimination est appliqué pour supprimer le point de repos dont l'AVE est le plus petit.

Ce processus s'arrête lorsque l'erreur d'approximation passe à un seuil qui est défini au début. Cette idée est similaire à l'approche de Latecki [LL99].

Tous les méthodes ci-dessus sont proposées pour traiter des courbes non-bruitées. Nous nous intéressons aussi aux courbes bruitées. C'est pourquoi dans ce chapitre, nous allons présenter une nouvelle méthode de détection de points dominants. Cette méthode peut accepter des bruits existants dans la courbe en utilisant la notion de segment flou (voir le chapitre 1).

### 3.2.2 Les approches multi-échelle, multi-épaisseur

Concernant les méthodes de détection de points dominants, il y a toujours deux objectifs qui sont contradictoires. Le premier est la simplification de la représentation de la courbe, le deuxième est la qualité de détection des caractéristiques principales de la courbe. Ces caractéristiques peuvent être des segments de droite, des arcs de cercle, ou bien des points de coin, . . . Afin de faire un compromis entre ces deux objectifs, il existe dans la littérature plusieurs méthodes qui utilisent une approche multi-échelle ou multi-résolution pour régler ce problème.

En général, l'approche multi-échelle considère la représentation de forme à travers différentes échelles pour obtenir une bonne description de la courbe. Witkin [Wit83] a introduit la notion de représentation d'espace des échelles. Il considère une famille de représentations de forme en changeant la valeur d'un paramètre qui joue le rôle d'échelle dans cette représentation. Mokhtarian a proposé une technique dans l'espace d'échelle de courbure [MS98]. Zhang et al. [ZLY<sup>+</sup>07] calculent la courbure de la courbe avec le filtre Gaussien avec différentes échelles. Leur méthode est basée sur la technique de l'espace d'échelle de courbure dans laquelle les points dominants sont détectés comme des extrêmes locaux du produit de courbure [ZLY<sup>+</sup>07] dont la valeur dépasse à un seuil.

Arrelbola [AH05] a traité une structure de multi-résolution qui contient les basses résolutions successives d'un même objet en utilisant une approche pyramidale. Il a adapté l'algorithme reliant des pixels de multi-résolution pour le traitement de la courbe qui est représenté par une chaîne de Freeman.

Un autre cadre multi-ordre au sens de l'épaisseur a été proposé dans [DRTW04] afin d'analyser une courbe discrète à différents niveaux d'épaisseur. Cette approche utilise des segments flou d'ordre  $d$  pour la polygonalisation d'une courbe discrète.

Nous allons proposer dans les sections suivantes des méthodes multi-épaisseur pour la représentation des courbes bruitées sans paramètres. L'une est une méthode qui détermine l'épaisseur de travail la plus appropriée. L'autre essaie de décomposer une courbe en parties significatives.

## 3.3 Critères d'évaluation

### 3.3.1 Les critères existants

Dans la littérature, plusieurs méthodes de détection de points dominants et d'approximation polygonale ont été proposées. Alors, une mesure d'évaluation permettant de comparer les méthodes existantes est essentielle. Il existe deux critères d'évaluation [Sar93, Ros97] qui sont utilisés majoritairement pour l'évaluation des méthodes de détection de points dominants ainsi que les méthodes de polygonalisation. Le critère de Sarkar [Sar93] repose sur l'erreur d'approximation et le taux de compression. L'erreur d'approximation peut être l'erreur quadratique ( $ISE$ ) ou bien la distance maximale ( $L_\infty$ ) entre les points de la courbe donnée en entrée et le polygone approximé. Le taux de compression est la capacité de réduction des données entre la courbe

donnée en entrée et le polygone approximé :

$$CR = \frac{N}{N_{DP}}$$

avec  $N$  est le nombre de points de la courbe,  $N_{DP}$  est le nombre de points dominants extraits.

Un grand taux de compression apporte beaucoup d'erreurs d'approximation. Au contraire, le maintien d'une petite erreur d'approximation cause un petit taux de compression. Alors, Sarkar [Sar93] a introduit le critère FOM (figure de mérite) qui est une division entre le taux de compression et l'erreur d'approximation comme un compromis entre l'erreur basse d'approximation et le bénéfice de la réduction de données élevées.

$$FOM = \frac{CR}{ISE}$$

Cette mesure est utilisée afin de faire un compromis entre un grand taux de compression et une petite erreur d'approximation qui sont obtenus avec une méthode de détection de points dominants.

Rosin [Ros97] divise l'évaluation en deux composantes : efficacité et fidélité. L'idée principale est de comparer le polygone approximé qui correspond à la méthode de détection de points dominants à un polygone optimal qui est construit dans la même condition. L'efficacité mesure la capacité de compression du polygone approximé dans les mêmes conditions d'erreur d'approximation. La fidélité mesure l'erreur d'approximation du polygone approximé dans les mêmes conditions que le nombre de segments du polygone.

$$Efficacité = \frac{N_{opt}}{N_{approx}}$$

$$Fidélité = \frac{E_{opt}}{E_{approx}}$$

où  $N_{opt}$  (resp.  $N_{approx}$ ) est le nombre de sommets dans le polygone optimal (resp. approximé)  $E_{opt}$  (resp.  $E_{approx}$ ) est l'intégrale de l'erreur quadratique du polygone optimal (resp. approximé). Une mesure combinée entre 2 composants est définie comme une moyenne géométrique de ces mesures :

$$Merit = \sqrt{Fidélité \times Efficacité}$$

### 3.3.2 Un nouveau critère alternatif d'évaluation pour les courbes bruitées

En ce qui concerne les critères d'évaluation, la mesure de mérite permet de comparer parmi les résultats obtenus par des méthodes de détection de points dominants quand le nombre de points dominants est différent. Pourtant, comme la complexité d'un algorithme optimal est lourde ( $O(n^2)$  [PCV94] pour le problème de min- $\varepsilon$ ,  $O(Mn^2)$  [PD95] pour le problème de min-#), l'utilisation du critère de Rosin mène à une méthode inefficace pour déterminer le paramètre le plus approprié à travers une approche multi-échelle. Donc, nous proposons d'utiliser une approche reposant sur le critère de Sarkar pour déterminer le paramètre approprié d'épaisseur.

Un certain nombre d'auteurs [MS04, AH05] montrent que le critère FOM n'est pas suffisant pour faire un compromis entre l'intégrale de l'erreur quadratique (ISE) et le taux de compression (CR). La raison est que l'ISE change plus rapidement par rapport CR pour plusieurs courbes testes. Marji et al. [MS04] ont proposé une version modifiée de Sarkar :  $FOM_n = \frac{CR^n}{ISE}$ . En réalité, ils utilisent  $n = 2$ . Nous proposons dans cette section un nouveau critère d'évaluation afin de

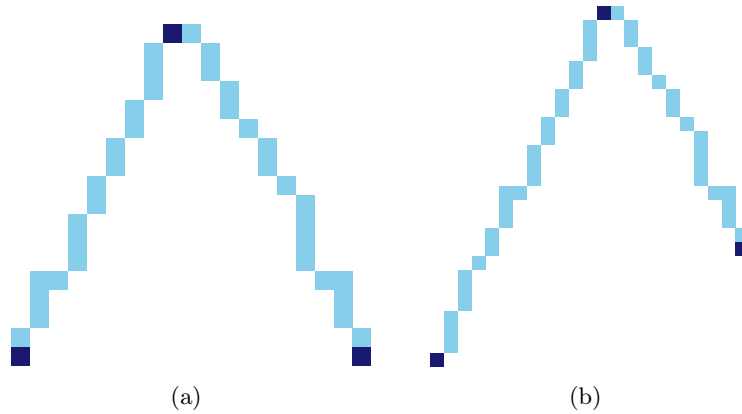


FIGURE 3.1 – a,b -Deux courbes ont la même valeur  $L_\infty$  par rapport aux polygones construits à partir de leurs points dominants mais en ce qui concerne l'ISE, la deuxième courbe a une ISE plus grande que celle de la première courbe.

s'adapter aux courbes bruitées. Ce critère est introduit en reposant sur une observation où la décision humaine d'un point considéré comme un coin dépend aussi de l'approximation locale. D'autre part, l'ISE qui définit l'erreur d'approximation au niveau global,  $L_\infty$  mesure l'erreur d'approximation au niveau local. Nous trouvons que la décision humaine à propos d'un point dominant dépend aussi de l'approximation locale. De plus, à cause de la propriété additionnelle du critère d'approximation *ISE*, dans plusieurs cas, le nombre de coins qui sont observés par les hommes et  $L_\infty$  pour une région de support d'un point dominant ne changent pas tandis que l'ISE augmente considérablement. Considérons deux courbes dans la figure 3.1 a, b. Les deux courbes ont le même nombre de points dominants et la même  $L_\infty$  par rapport aux polygones construits à partir de ceux-ci. Pourtant la courbe dans la figure 3.1.b a l'ISE qui est plus grande que celle de la courbe dans la figure 3.1.a. Au contraire, la propriété dominante du second point dans la figure 3.1.b est plus grande que celle du point correspondant dans la figure 3.1.a grâce à la longueur du segment flou à gauche dans sa région de support. Comme  $L_\infty$  contrôle la distance de variation maximale entre chaque point de la courbe et le polygone approximé, il est plus approprié que l'ISE pour considérer les segments flous à gauche et à droite de la région de support d'un point dominant.

Donc, le maintien d'un ISE bas n'est pas suffisant afin d'obtenir une bonne représentation polygonale. Nous devons considérer aussi  $L_\infty$  malgré qu'il corresponde aussi à l'erreur d'approximation. Alors, nous proposons une version modifiée du critère d'évaluation de Sarkar en utilisant la remarque de Marji. Notre proposition s'est basée sur l'observation que l'ISE change plus rapidement que  $CR$  et  $L_\infty$ . Nous supposons que l'ISE a une relation quadratique par rapport à  $L_\infty$ . Cette hypothèse vient de la propriété quadratique dans la formule de l'ISE. Nous montrons dans la figure 3.2 le changement du taux  $CR/L_\infty$  à travers des épaisseurs différentes. Cette figure confirme cette hypothèse concernant la relation linéaire entre  $CR$  et  $L_\infty$ . Donc, nous proposons le **nouveau critère d'évaluation** suivant :

$$ND\_FOM = \frac{CR^2}{ISE} \times \frac{CR}{L_\infty} = \frac{CR^3}{ISE \times L_\infty}$$

Ce critère sera utilisé dans les deux méthodes non-paramétriques qui seront présentées dans les deux sections 5 et 6 de ce chapitre et nous montrons alors son intérêt sur de nombreux exemples.

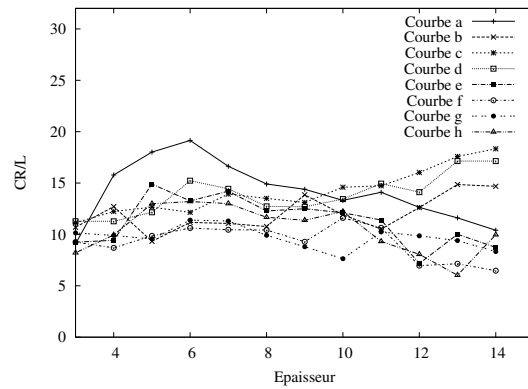


FIGURE 3.2 – Profil de CR/L sur 6 courbes présentées dans la figure 3.22.

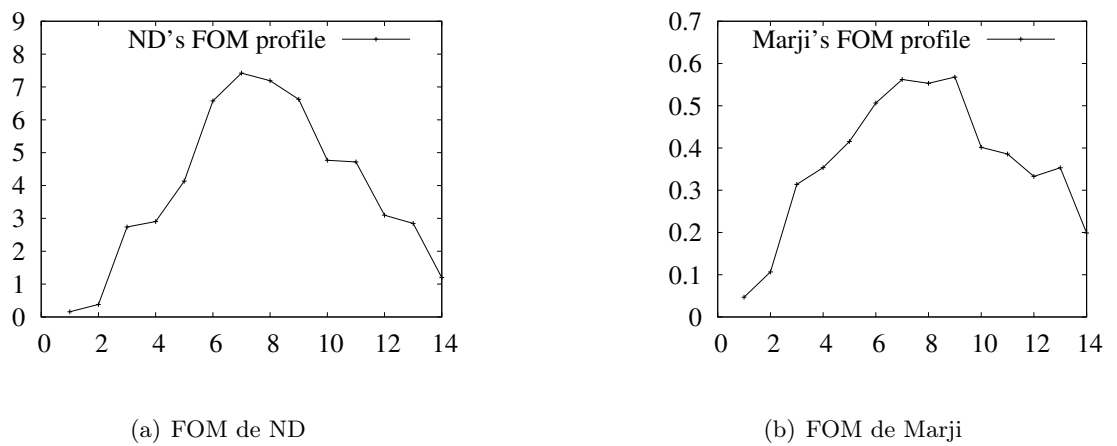


FIGURE 3.3 – Comparaison entre les profils du critère de Marji et de notre critère sur la courbe de la figure 3.22.h.

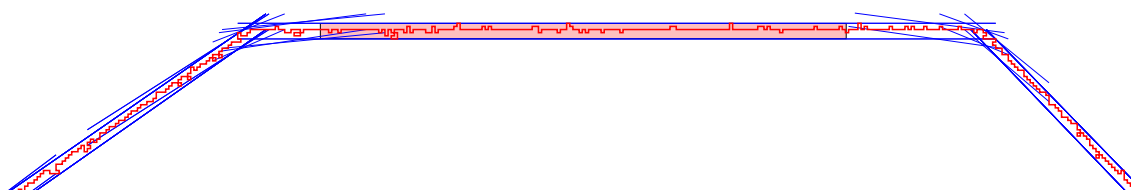
### 3.4 Détection de points dominants avec un paramètre d'épaisseur

$\nu$

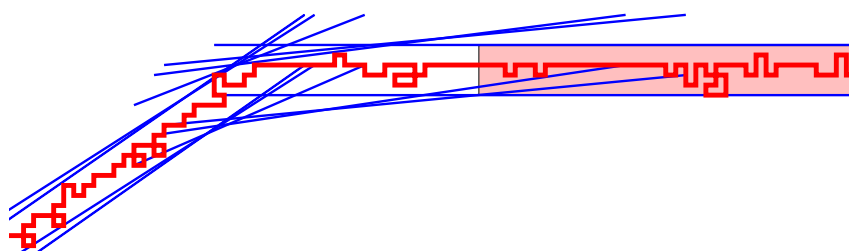
Plusieurs méthodes ont été proposées pour la détection de points dominants en utilisant des définitions différentes de points dominants. Nous rappelons ci-dessous la définition d'Attneave [Att54] dans ce premier travail qui concerne les points dominants.

**Définition 3.4.1** *Un point dominant (coin) sur une courbe discrète est un point dont la valeur de courbure est localement maximale.*

Nous pouvons détecter directement les points dominants d'une courbe discrète comme les maximums locaux dans son profil de courbure (voir le chapitre précédent). Le plus grand problème est qu'il y aura plusieurs points faiblement dominants qui correspondent aux petits pics du profil de courbure de la courbe. Dans cette section, nous présentons une nouvelle méthode pour la détection de points dominants reposant sur des résultats théoriques de la géométrie discrète : la décomposition d'une courbe en sa séquence de segments flous maximaux (voir la section 3 du chapitre 2). Cette décomposition nous permet d'obtenir des informations importantes concernant la structure géométrique de la courbe. L'épaisseur des segments flous maximaux permet de travailler à travers des échelles différentes, et de tenir compte du bruit qui apparaît dans la courbe. Cette épaisseur est le seul paramètre dans notre méthode.



(a) La suite des segments flous maximaux



(b) Zoom de (a)

FIGURE 3.4 – La zone rouge n'est pas une zone commune de segments flous maximaux.

#### 3.4.1 Points dominants et région de support (ROS)

En nous inspirant de [CR89], nous proposons dans cette section la notion de région de support (ROS) qui s'adapte à la notion de segment flou.

**Définition 3.4.2** *La région de support d'épaisseur  $\nu$  au point  $M$  d'une courbe discrète est définie par les segments flous d'épaisseur  $\nu$  les plus longs qui sont construits à partir de ce point vers la gauche et vers la droite de la courbe. L'angle de ROS à ce point est l'angle entre les pentes des droites d'appui des segments flous. (voir la figure 3.5).*



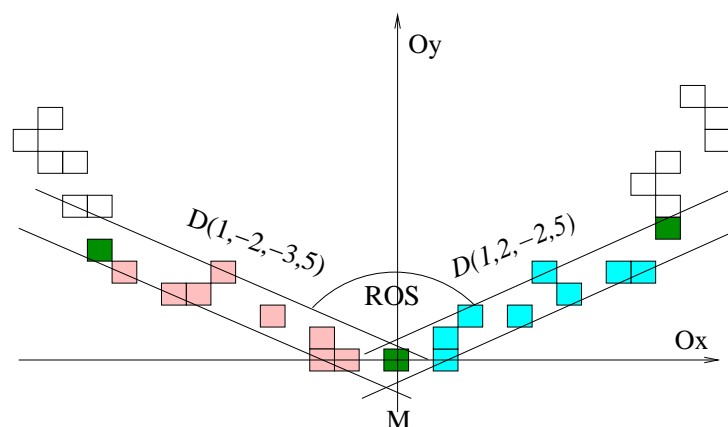


FIGURE 3.5 – Les segments flous à gauche et à droite du point M constitue la région de support de ce point.

**Remarque 1** Plus l'angle de ROS en un point est petit, plus le caractère dominant de ce point est élevé.

Cette remarque vient du travail décrit dans le chapitre 2 où la courbure au point  $C_k$  est estimée comme l'inverse du rayon de cercle circonscrit qui passe par  $C_k$  et les 2 extrémités à gauche et à droite des segments flous maximaux à gauche et à droite qui sont construits à partir de ce point. Par conséquent, si l'angle de ROS en ce point est  $\pi$ , ce point ne peut pas être un point dominant.

### 3.4.2 Relation entre les points dominants et la séquence de segments flous maximaux

Dans cette section, nous étudions la relation entre les positions des points dominants et les segments flous maximaux. Nous avons besoin des propriétés 2.4.1, 2.5.1 qui ont été présentées en détail dans la sections 2.4, 2.5 du chapitre 2.

**Proposition 3.4.1** Si un point n'appartient qu'à un seul segment flou maximal, son angle de ROS est  $\pi$

**Preuve 3.4.1** Considérons les points de la zone rose (voir la figure 3.4) qui ne se trouvent dans aucune zone commune de segments flous maximaux. Alors, ils n'appartiennent qu'à un seul segment flou maximal. En appliquant la propriété 2.5.1, les extrémités à gauche et à droite de ces points sont aussi 2 extrémités à gauche et à droite d'un même segment flou maximal. Par conséquent, les angles de ROS de ces points sont  $\pi$ .

Considérons maintenant une zone commune de quelques segments flous maximaux consécutifs d'une courbe  $C$ .

**Proposition 3.4.2** La zone commune la plus petite de segments flous maximaux successifs dont les pentes sont croissantes ou décroissantes contient un ou plusieurs points dont l'angle de ROS est localement minimal.

**Preuve 3.4.2** Considérons  $k$  segments flous maximaux consécutifs qui ont une zone commune. Supposons que ces  $k$  segments flous maximaux ont des extrémités correspondantes :  $(B_1, E_1)$ ,

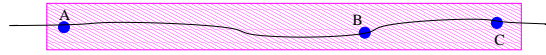


FIGURE 3.6 – Si un segment flou maximal contient plus de 2 candidats de points dominants, les candidats au milieu sont des candidats faiblement dominants.

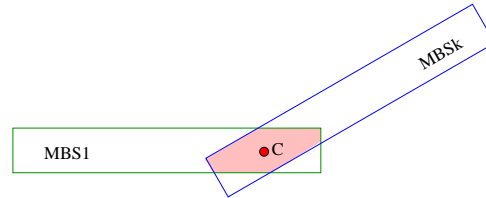


FIGURE 3.7 – Un point dominant est espéré de se trouver au point central de la zone la plus petite.

$(B_2, E_2), \dots, (B_k, E_k)$ . Leur pentes satisfont  $slope_1 < slope_2 < \dots < slope_k$  (similairement pour le cas décroissant). Grâce à la propriété 2.4.1, nous devons avoir :  $B_1 < B_2 < \dots < B_k$ ;  $E_1 < E_2 < \dots < E_k$ . Puisque ces segments flous maximaux partagent la zone commune, nous avons  $B_k < E_1$ . C'est pourquoi la zone commune est  $[B_k, E_1]$ . Grâce à la propriété 2.5.1, les extrémités à gauche et à droite des points dans  $k$  zones communes partielles  $[B_1, B_2[$ ,  $[B_2, B_3[$ ,  $\dots [B_k, E_1[$  sont respectivement  $(B_1, E_1)$ ,  $(B_1, E_2), \dots (B_1, E_k)$ . Les pentes des segments flous à gauche pour les points qui se trouvent dans ces zones communes partielles sont toujours égales  $slope_1$ . Au contraire, les pentes des segments flous à droite pour ces points dans ces zones communes partielles sont respectivement  $slope_1, slope_2, \dots, slope_k$ . Similairement, nous déduisons que les pentes des segments flous à droite pour les points qui se trouvent dans les zones communes  $]E_1, E_2], \dots, ]E_{k-1}, E_k]$  sont égale à  $slope_k$  et les pentes des segments flous à gauche pour ces points sont respectivement  $slope_2, \dots, slope_k$ . L'angle ROS des points qui se trouvent dans la zone  $[B_k, E_1)$  est égal à  $(slope_1, slope_k)$  et cette valeur due à l'hypothèse de la croissance des pentes des segments flous maximaux est minimale pour tous les points dont l'indice est de  $B_1$  à  $E_k$ . C'est pourquoi, cette zone qui est la plus petite contient un candidat point dominant.

Cette zone qui contient des points dont l'angle de ROS est minimal correspond à une zone localement la plus courbée de la courbe. Intuitivement, elle contient un point dominant de la courbe. Nous utilisons une stratégie heuristique ci-dessous pour localiser la position de chaque point dominant dans chaque zone commune.

**Stratégie heuristique :** Dans chaque zone commune des segments flous maximaux dont les pentes sont croissantes ou décroissantes, un candidat point dominant est localisé comme le point central de cette zone.

Afin d'éliminer des candidats faiblement dominants, nous utilisons la propriété naturelle ci-dessous due à la forme de segment de droite d'un segment flou maximal et due à la propriété de coin d'un point dominant (voir la figure 3.6).

**Remarque 2** Un segment flou contient au maximum 2 points dominants.

### 3.4.3 Algorithme proposé

#### Algorithme

Nous proposons ci-dessous un algorithme pour la détection des points dominants.

Considérons la zone qui satisfait cette condition. Cette zone contient un candidat point dominant (cf. proposition 2.5.1). Grâce à la propriété 2.4.1, cette zone doit être l'intersection entre le premier et le dernier segment flou maximal de la suite de segments maximaux consécutifs (voir la figure 3.7). Nous rappelons que chaque point dans cette zone a la même région de support. Puis, nous proposons de localiser le candidat point dominant comme le point central qui possède des propriétés géométriques proches de celles d'un coin. C'est pourquoi le candidat est détecté comme le **point central** de la partie de la courbe qui correspond à cette zone.

En se basant sur le cadre théorique de la section 3.4.1 et en utilisant cette stratégie heuristique, nous présentons ci-dessous un algorithme pour la détection de points dominants. Il contient deux parties :

- le balayage des zones communes des segments flous maximaux consécutifs.
- la détection de points dominants dans chaque zone commune dont les pentes sont croissantes ou décroissantes.

---

**Algorithme 10** : Détection de points dominants.

---

**Données** :  $C$  courbe discrète de  $n$  points,  $\nu$  épaisseur de segmentation

**Résultat** :  $D$  l'ensemble des points dominants

**début**

	Construire $MBS_\nu = \{MBS(B_i, E_i, \nu)\}_{i=1}^m, \{slope_i\}_{i=1}^m$ ; $m =  MBS_\nu $ ; $p = 1$ ; $q = 1$ ; $D = \emptyset$ ;
Balayage	<b>pour</b> $p \leq m$ <b>faire</b> <b>pour</b> $E_q > B_p$ <b>faire</b> $p++$ ; Ajouter $(q, p - 1)$ à une <i>pile</i> ; $q = p - 1$ ;
Détection	<b>pour</b> <i>pile</i> $\neq \emptyset$ <b>faire</b> Prendre $(q, p)$ à partir de <i>pile</i> ; Décomposer $\{slope_q, slope_{q+1}, \dots, slope_p\}$ en séquences monotones; Déterminer la dernière séquence monotone $\{slope_r, \dots, slope_p\}$ ; Déterminer $DP$ - le point central de la zone commune des segments flous maximaux $MBS(B_r, E_r, \nu), \dots, MBS(B_p, E_p, \nu)$ ; $D = \{D \cup DP\}$ ;
	<b>fin</b>

---

### Complexité

La complexité de cette méthode dépend de la décomposition d'une courbe en séquence de segments flous maximaux. Nous pouvons utiliser notre technique de décomposition d'une courbe en suite de segments flous maximaux en temps  $O(n \log^2 n)$  (voir l'algorithme 6 dans le chapitre 2). Récemment, Faure [FF08] a proposé une amélioration de cette décomposition qui fonctionne en temps  $O(n \log n)$ . L'estimation des pentes des segments flous maximaux est effectuée en temps linéaire. D'autre part, chaque segment flou maximal est visité au maximum 2 fois lorsque la courbe est décomposée en zones communes de segments flous maximaux dont les pentes sont croissantes ou décroissantes. Donc, dans cette phase, les points dominants sont détectés en temps linéaire. Pour résumer, la complexité de cette méthode est  $O(n \log n)$  grâce à l'amélioration de Faure concernant la construction de la séquence de segments flous maximaux.

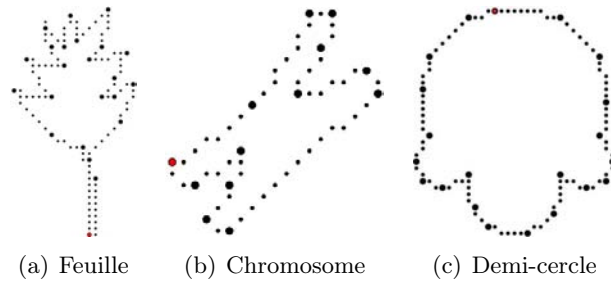


FIGURE 3.8 – Les points dominants détectés à l'épaisseur 0.9

### 3.4.4 Expérimentations

#### Expérimentation et comparaison avec les autres méthodes

Nous présentons dans la figure 3.8 les résultats qui sont obtenus par la première méthode sur les 3 courbes connues : chromosome, feuille et demi-cercle. Les figures 3.9, 3.10, 3.11 et 3.12 présentent les comparaisons avec d'autres méthodes de la littérature sur ces courbes.

Nous avons comparé notre méthode avec ces autres méthodes selon les critères de Sarkar [Sar93] (voir le tableau 3.1) ainsi que les critères de Rosin [Ros97] (voir le tableau 3.2).

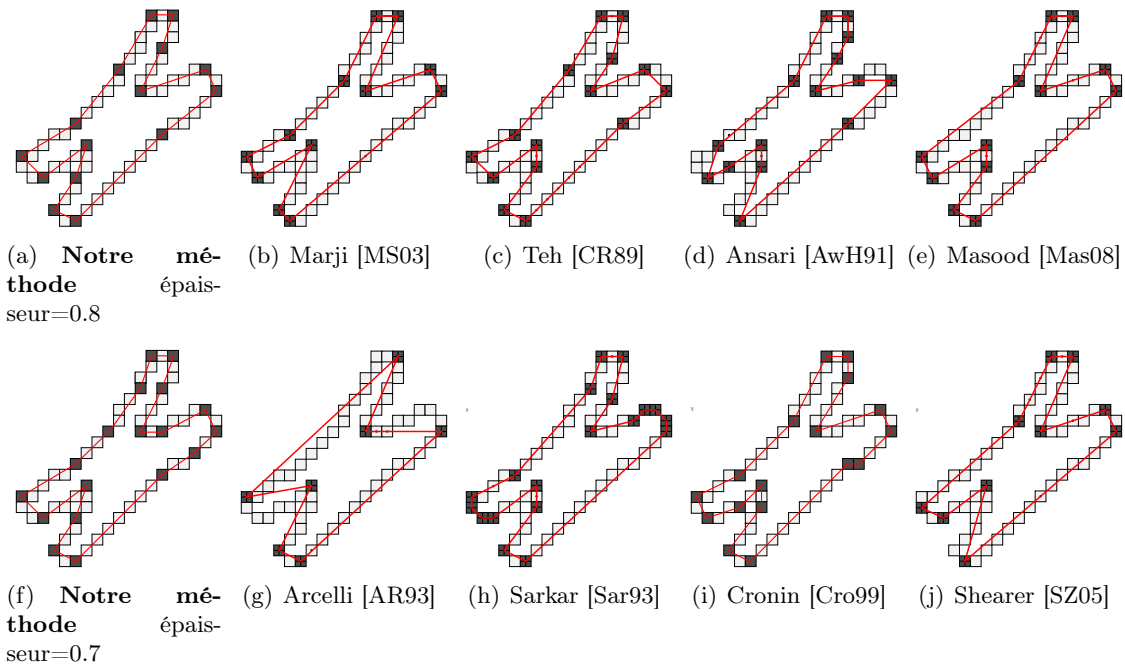


FIGURE 3.9 – Les différentes méthodes de détection de points dominants sur la courbe du chromosome

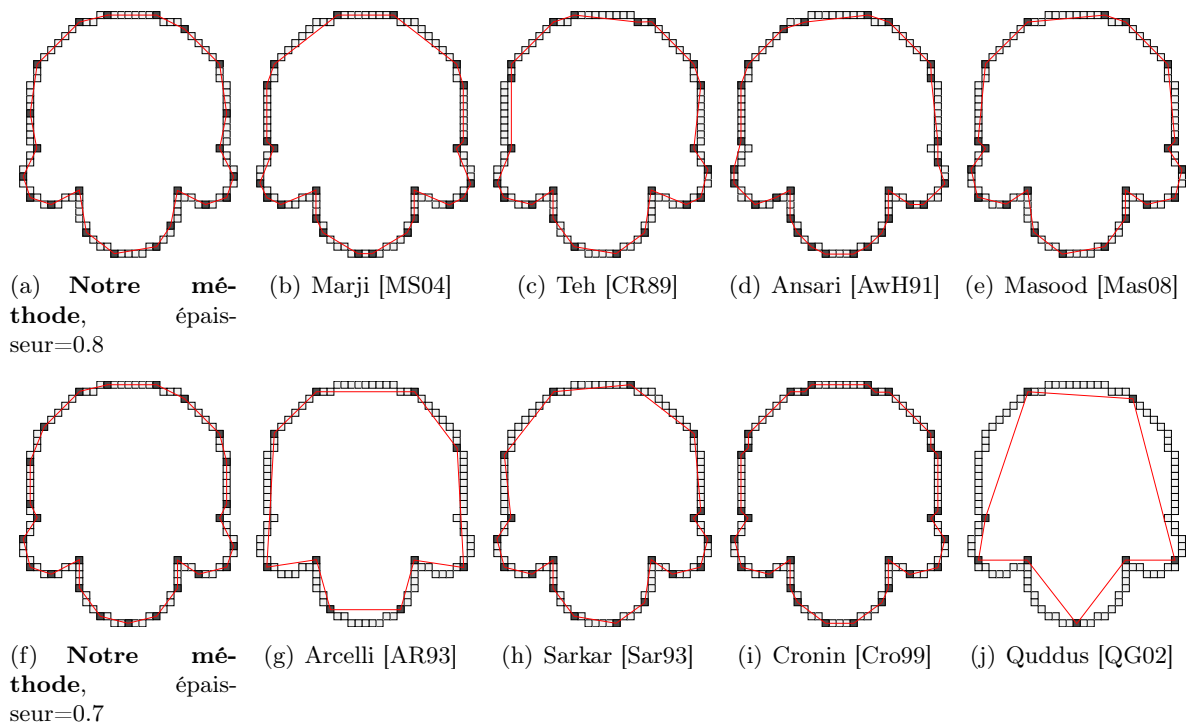


FIGURE 3.10 – Les différentes méthodes de détection de points dominants sur la courbe de demi-cercle.

<i>Courbe</i>	<i>Algorithme</i>	<i>nDP</i>	<i>CR</i>	$L_\infty$	<i>ISE</i>	<i>FOM</i>
Chromosome	<b>Notre méthode</b> , width=0.7	18	3.3333	0.63246	4.05964	0.821091
	Sharkar [Sar93]	19	3.16	0.55	3.857	0.819
	Shearer [SZ05]	10	6		6.086	0.744
	Wu [Wu03]	17	3.53	0.64	5.01	0.704
	<b>Notre méthode</b> , épais- seur=0.8	15	4	0.685994	5.69133	0.702824
	Ray [RR92b]	18	3.33	0.65	4.81	0.693
	Masood [Mas08]	12	5	0.88	7.76	0.65
	Ray [RR92a]	18	3.33	0.707	5.566	0.599
	Teh [CR89]	15	4.00	0.74	7.2	0.556
	Marji [MS03]	11	5.45	0.895	9.96	0.548
	Ansari-Huang [AwH91]	16	3.75	2	20.25	0.185
Feuille	<b>Notre méthode</b> , épais- seur=0.7	33	3.63636	0.632456	5.56036	0.65398
	<b>Notre méthode</b> , épais- seur=0.8	24	5	0.727607	9.98827	0.500587
	Masood [Mas08]	23	5.217	0.74	10.61	0.49
	Shearer [SZ05]	22	5.46		13.06	0.418
	Marji [MS03]	22	5.45	0.78	13.21	0.413
	Sarkar [Sar93]	23	5.22	0.784	13.17	0.396
	Teh [CR89]	29	4.14	0.99	14.96	0.277
	Ray [RR92b]	32	3.75	0.99	14.18	0.264
	Ray [RR92a]	32	3.75	0.996	14.718	0.255
	Wu [Wu03]	23	5.22	1	20.34	0.256
Ansari-Huang [AwH91]	30	4.00	2.13	25.57	0.156	
Demi-cercle	<b>Notre méthode</b> (épais- seur=0.7)	25	4.12	0.557086	5.42258	0.759786
	Masood [Mas08]	22	4.64	0.72	8.61	0.54
	<b>Notre méthode</b> (épais- seur=0.8)	22	4.681	0.784	10.31	0.454
	Wu [Wu03]	27	3.78	0.88	9.19	0.411
	Ray [RR92b]	27	3.78	0.88	11.5	0.329
	Shearer [SZ05]	13	7.85		22.97	0.342
	Sarkar [Sar93]	19	5.37	1.474	17.37	0.309
	Ray [RR92a]	29	3.52	0.833	11.818	0.298
	Ansari-Huang [AwH91]	28	3.64	1.26	17.83	0.24
	Marji [MS03]	18	5.67	1	24.2	0.234
	Masood [Mas08]	19	5.37	1.00	23.9	0.23
Teh [CR89]	22	4.64	1	20.61	0.225	

TABLE 3.1 – Comparaison en utilisant le critère de Sarkar [Sar93].

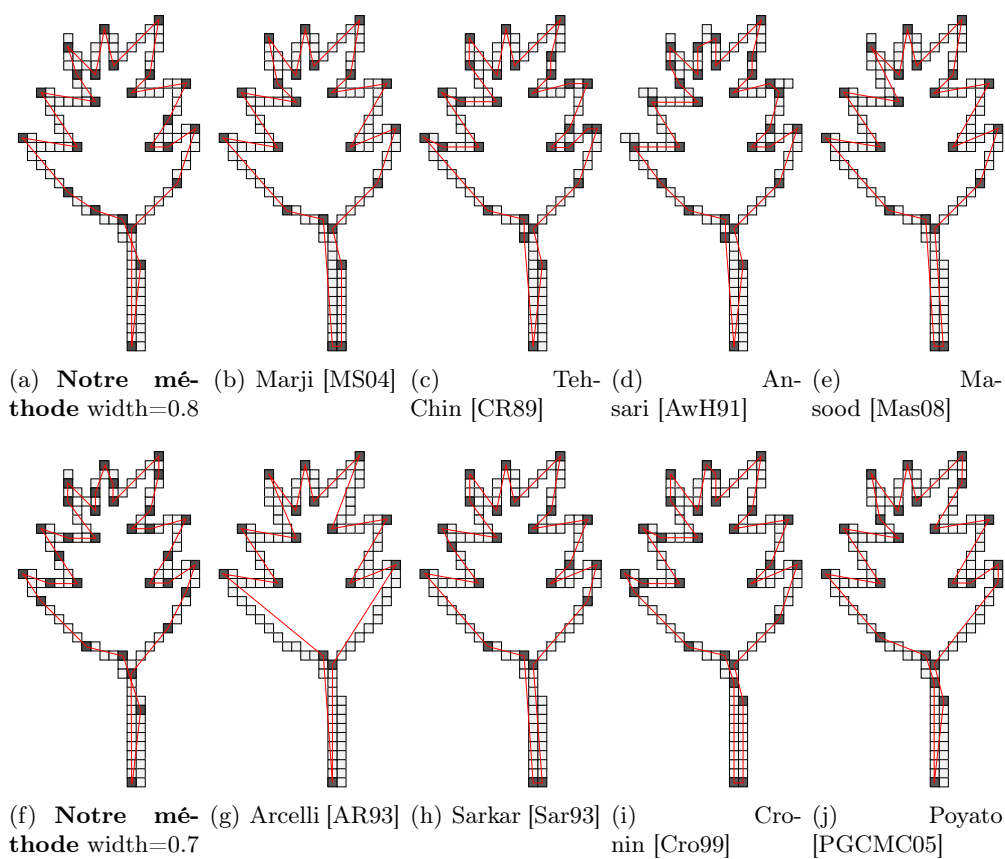
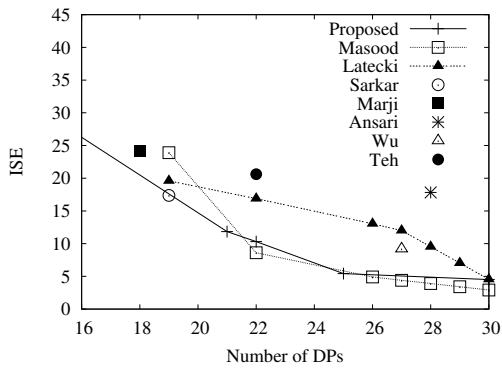


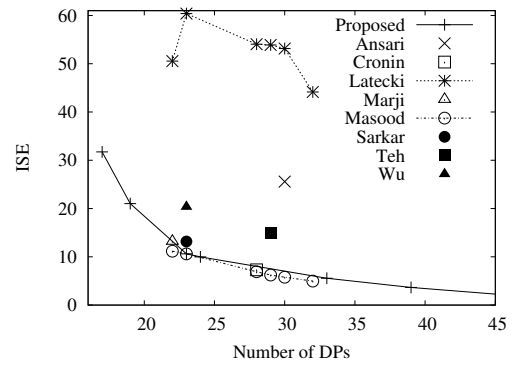
FIGURE 3.11 – Les différentes méthodes de détection de points dominants sur la courbe de la feuille.

Méthode	nDP	ISE	Fidélité	Efficacité	Merit	Rosin's rank
<b>Notre méthode</b> (épaisseur=0.7)	25	5.422	86.4	96	91.08	-
Massod [Mas08]	21	9.82	81.7	95.8	88.5	-
Ramer [U.R72]	26	5.27	76.9	92.6	84.4	2
<b>Notre méthode</b> (épaisseur=0.8)	22	10.31	67.99	86.36	76.6	-
Sarkar II [Sar93]	20	13.65	66	78.9	72.2	4
Marji [MS04]	15	22.7	61.67	84.0	71.9	-
Sarkar I [Sar93]	19	17.38	57.8	73.7	65.3	6
Arcelli-Ramella [AR93]	10	75.10	51.8	80.3	64.5	7
Wu [Wu03]	27	9.01	41.09	74.07	55.17	-
Teh [CR89]	22	20.61	34.0	59.2	42.4	17
Ansari-Huang [AwH91]	28	17.83	18.8	49.1	28.8	26
Ray [RR92a]	29	11.82	25.4	60.0	39.0	23

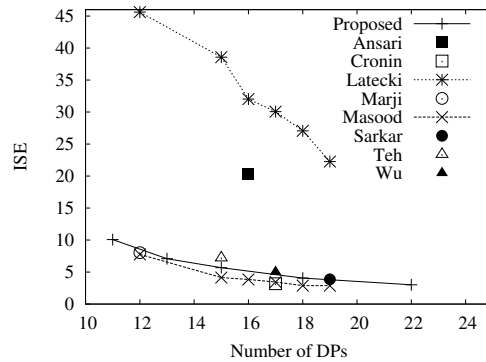
TABLE 3.2 – Comparaison en utilisant le critère de Rosin [Ros97] sur la courbe de demi-cercle.



(a) Courbe de demi-cercle



(b) Courbe de feuille



(c) Courbe de chromosome

FIGURE 3.12 – Relation entre le nombre des points dominants et l'erreur d'approximation (ISE) parmi les méthodes différentes : notre méthode, Massood [Mas08], Latecki [LL99], Shearer [SZ05], Wu [Wu03], Marji [MS03], Teh [CR89], Ansari [AwH91], Sarkar [Sar93], Cronin [Cro99]



<i>Courbe</i>	$N_0$	<i>Epaisseur</i>	$N_d$	$CR$	$L_\infty$	$ISE$
Poisson	595	3	22	27.045	3.347	591.019
		4	17	35	3.51	1030.59
Poisson bruité	644	3	26	24.77	3.417	845.93
		4	20	32.2	3.96	1611.35
		5	18	35.778	4.466	2432.92
Feuille réelle	338	2	26	13	1.946	205.406
		3	18	18.778	3	472.1
Feuille réelle bruité	370	2	32	11.562	2.045	265.684
		3	22	16.82	3.394	509.62
		4	17	21.765	4.274	934.85
Frontière française	3492	4	107	32.635	5.300	8045.97
		7	56	62.357	10	29799.4
		18	18	194	17.435	156015

TABLE 3.3 – Erreur d'approximation de notre méthode sur les courbes des figures 3.13, 3.31, 3.15.

Le tableau 3.1 montre une comparaison entre notre méthode et les autres méthodes reposant sur le critère de Sarkar. De plus, le tableau 3.2 présente une comparaison en utilisant le critère de Rosin. Notre méthode est meilleur que 27 méthodes dans la liste de Rosin [Ros97]. Cette liste contient un classement de 30 méthodes existantes dans la littérature selon les critères de Rosin.

D'autre part, en changeant la valeur du paramètre d'épaisseur, nous obtenons des nombres différents de points dominants avec l'erreur d'approximation correspondante pour chaque courbe. La figure 3.12 présente la relation entre le nombre de points dominants et l'erreur d'approximation parmi les différentes méthodes lorsqu'elles sont appliquées sur 3 courbes : feuille, chromosome et demi-cercle. Grâce à ce graphe, nous constatons que notre méthode est presque aussi efficace que celle de Masood (voir aussi la section 3.2, pages 65-66 pour avoir plus de détails sur cette méthode). Cette méthode qui utilise une procédure itérative est plus coûteuse que notre méthode. De plus, elle est meilleure que les autres méthodes.

### Travail avec des courbes bruitées ou déconnectées

Notre méthode peut travailler naturellement avec les courbes non-bruitées à travers des échelles différentes en utilisant le paramètre d'épaisseur. De plus, grâce à la notion de segment flou, cette méthode peut travailler aussi avec des courbes bruitées ou bien des courbes déconnectées. Les figures 3.13.b, 3.13.c, 3.31.b, 3.31.c présentent des résultats sur les contours qui sont extraits directement. D'autre part, les figures 3.13.d, 3.13.e, 3.13.f, 3.31.d, 3.31.e, 3.31.f, 3.16 présentent les résultats sur les courbes bruitées qui sont dégradées par le modèle Kanungo [KHB<sup>+</sup>94]. La figure 3.17 montre que la méthode proposée peut travailler bien avec des courbes déconnectées.

### Application à la vectorisation

Nous appliquons cette méthode pour la vectorisation. La figure 3.18 présente une petite base de données qui se compose de 4 images différentes dont les contours sont extraits dans la figure 3.19 en utilisant le filtre de Canny [Can86]. Chaque image de contour est considérée comme un ensemble des courbes. La méthode est appliquée sur ces courbes afin d'obtenir des vectorisations.

La figure 3.20 et le tableau 3.4 présentent les résultats obtenus par notre méthode sur ces images de contour. L'expérimentation est effectuée sur une configuration d'ordinateur comme

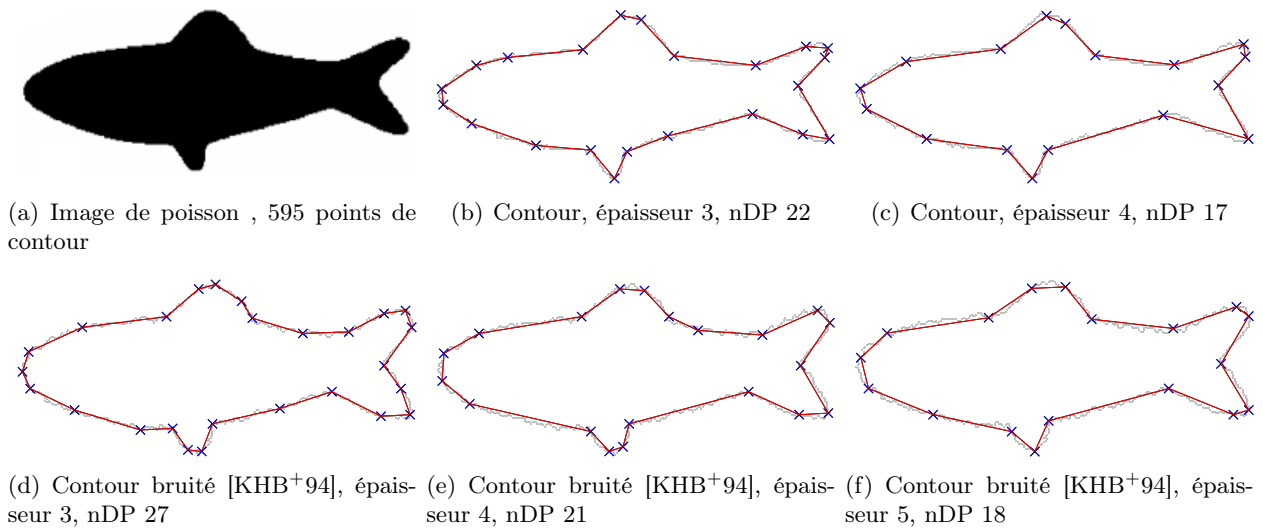


FIGURE 3.13 – Travail sur les données brutes, les images bruitées sont dégradées en utilisant le modèle de Kanungo [KHB<sup>+</sup>94]

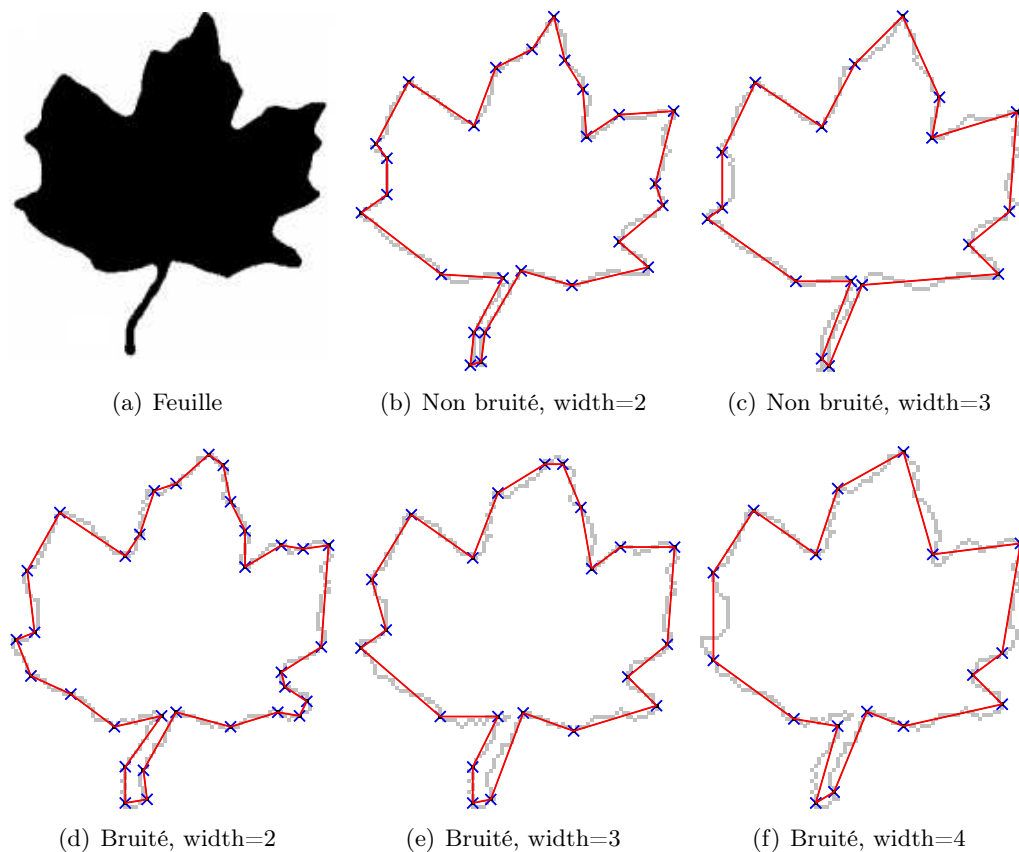


FIGURE 3.14 – Travail sur le contour d'une feuille en utilisant des épaisseurs différentes. Les courbes bruitées sont dégradées en utilisant le modèle de Kanungo [KHB<sup>+</sup>94].

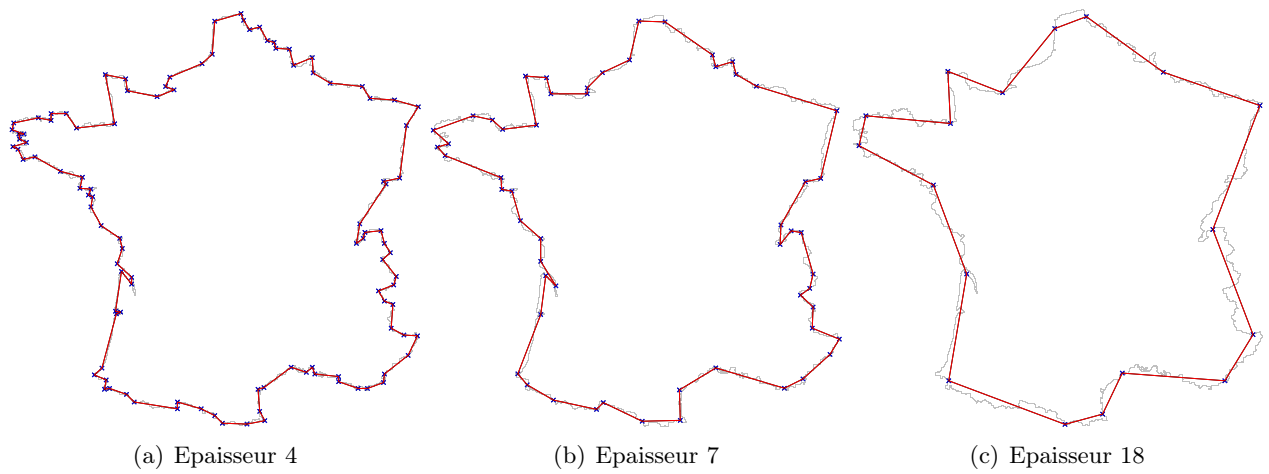


FIGURE 3.15 – Travail sur la frontière française.

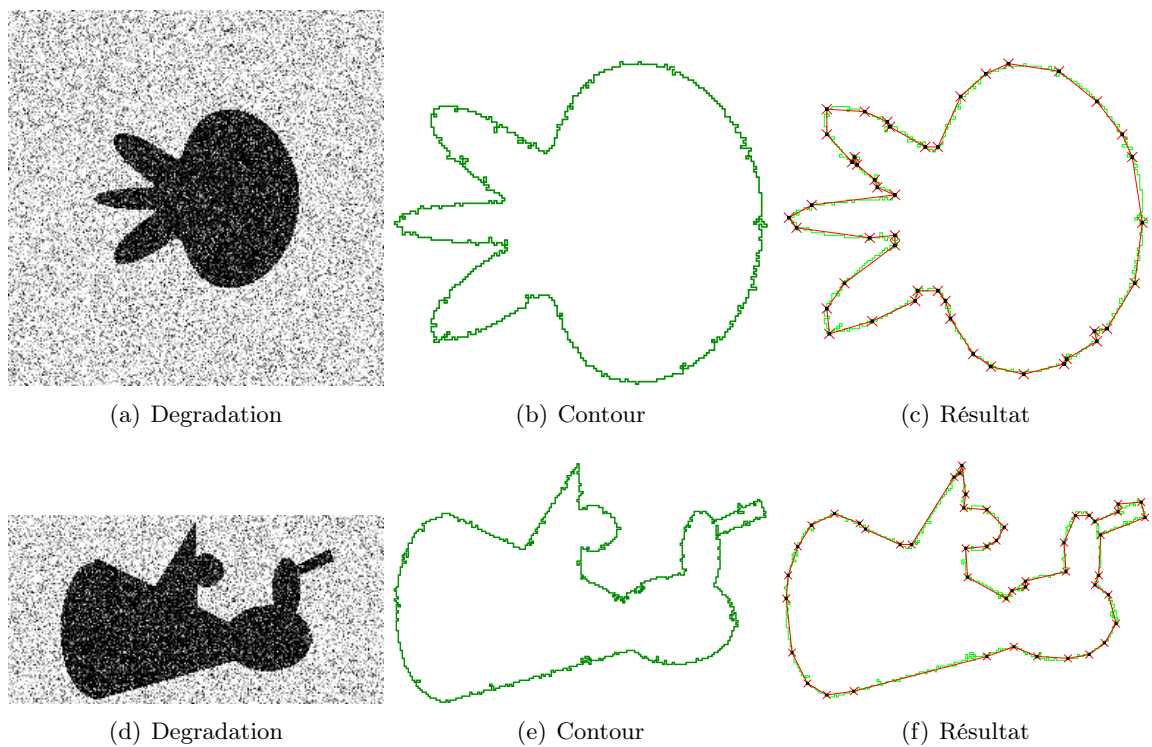


FIGURE 3.16 – Expérimentation sur des courbes bruitées en utilisant le modèle de Kanungo [KHB<sup>+</sup>94]. L'épaisseur de travail est de 2.

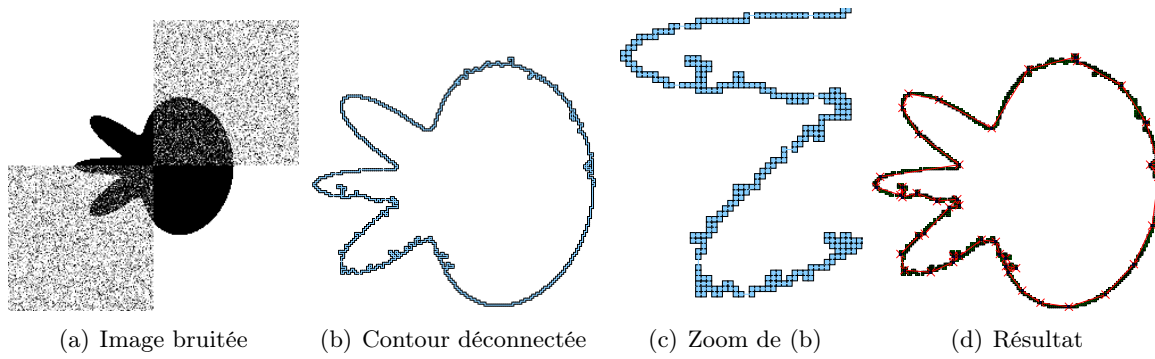


FIGURE 3.17 – Expérimentation sur une courbe bruitée déconnectée. L'épaisseur de travail est 2.

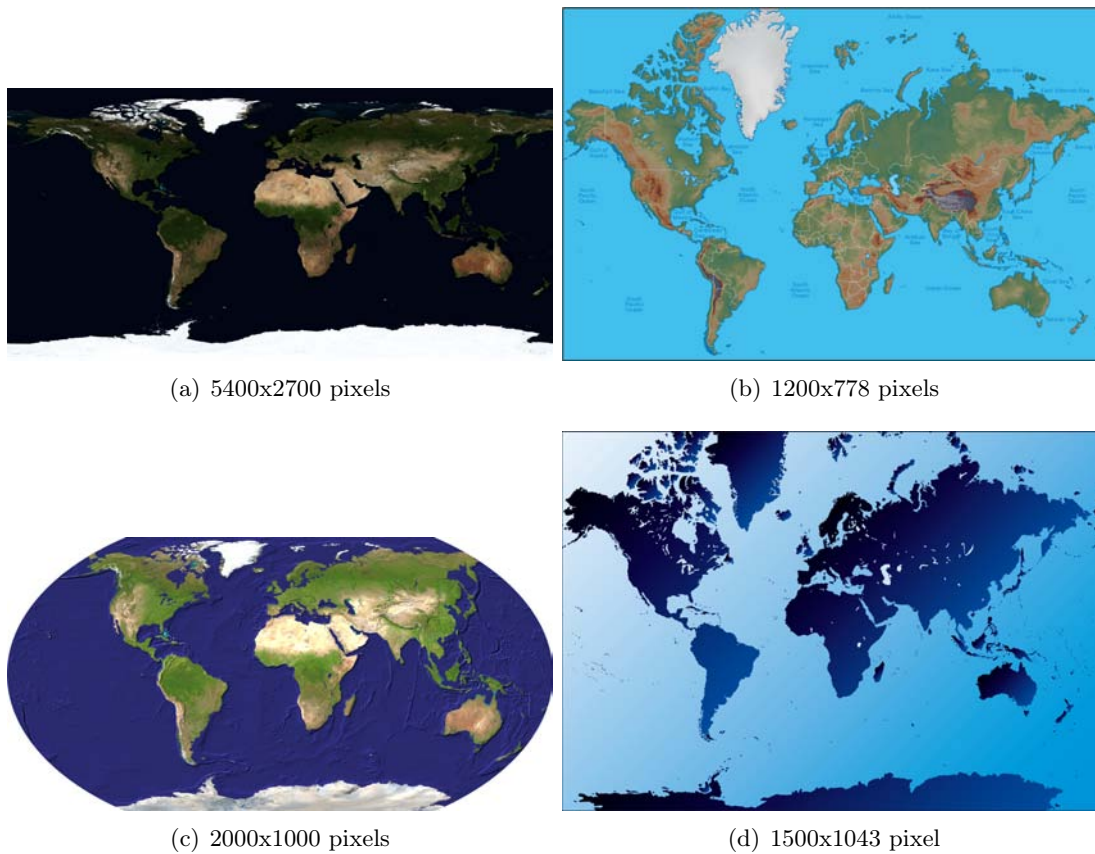


FIGURE 3.18 – Base de données des images.

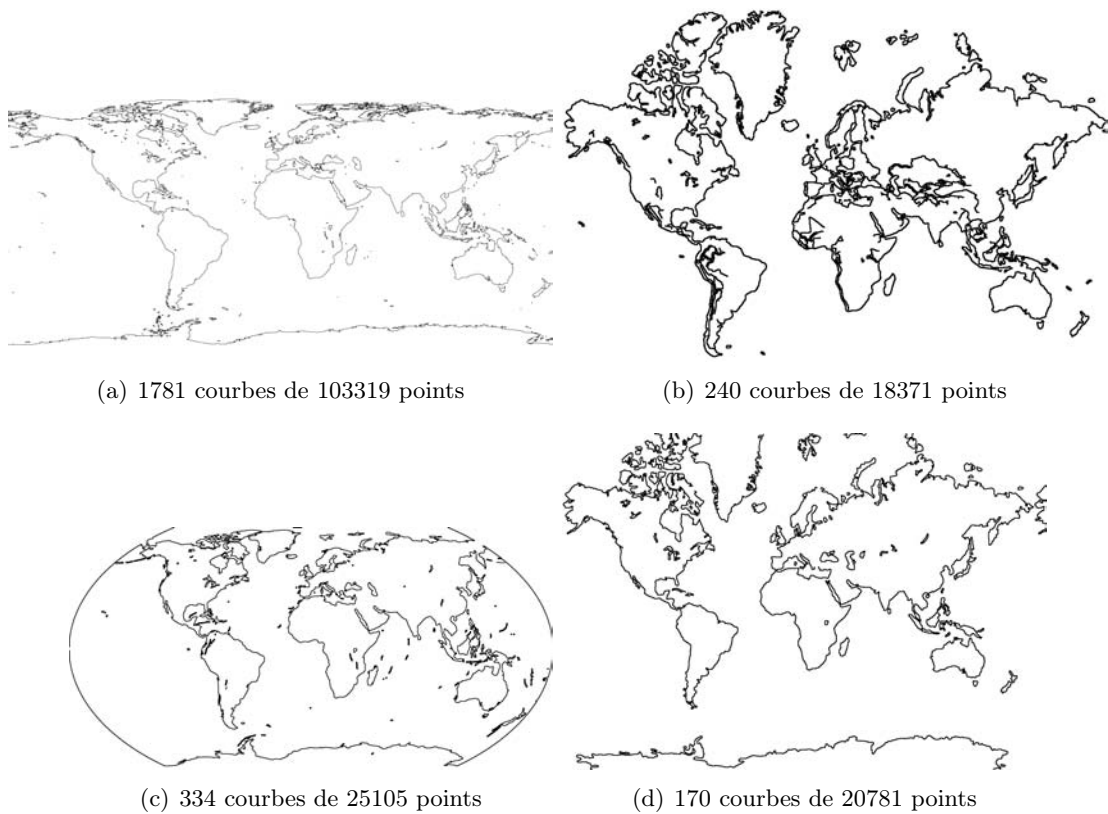


FIGURE 3.19 – Les images de contours extraites à partir de la figure 3.18. <sup>a</sup>

---

*a.* Pour la meilleure représentation, ces images sont érodées dans ces figures.

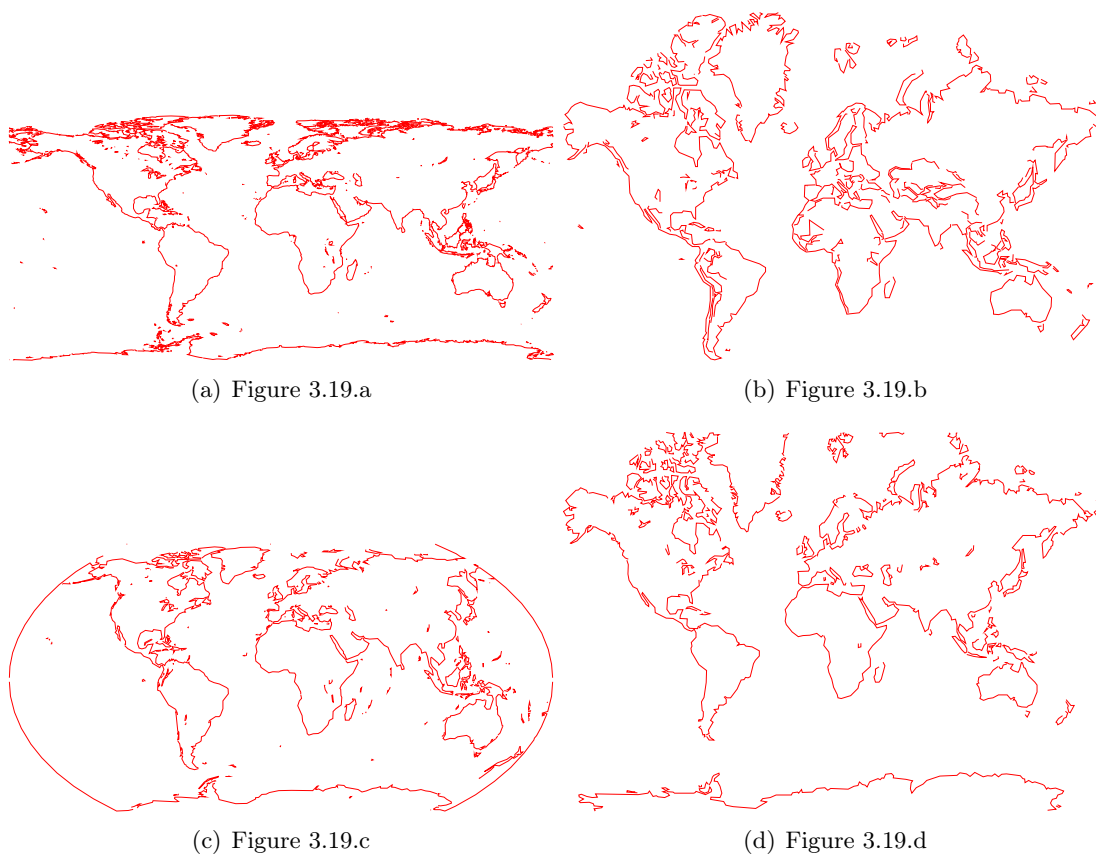


FIGURE 3.20 – Vectorisation de base de donnée d’images dans la figure 3.19. L’épaisseur de travail est 2.

suit : CPU Pentium 4 avec la vitesse de 2.26GHz, 4G de RAM, le noyau de linux 2.6.31-20. La méthode peut travailler avec  $10^5$  points en 6 secondes (voir le tableau 3.4).

Fig.	$N_C^0$	$N_P^0$	$N_{DP}^0$	CR	Time
3.20.a	1781	103319	14473	7.139	6019
3.20.b	240	18371	1818	10.1	1071
3.20.c	334	25105	2094	11.989	1671
3.20.d	170	20781	1949	10.662	1367

TABLE 3.4 – Résultats de la vectorisation sur la figure 3.20. Le temps est calculé en ms,  $N_C^0$  - nombre de courbes,  $N_P^0$  - nombre de points,  $N_{DP}^0$  - nombre de points dominants, CR - ratio de compression.

### 3.4.5 Discussion sur le nouveau critère d’évaluation

Nous présentons dans les figures 3.22, 3.31 et dans le tableau 3.5 les résultats qui sont obtenus par la suite d’épaisseurs successives. Les images en entrée sont présentées dans la figure 3.21 et les courbes dans la figure 3.22 sont extraites à partir de ces images bruitées qui sont dégradées par le modèle Kanungo [KHB<sup>+</sup>94]. La figure 3.23 montre aussi une séquence de résultats sur la courbe  $d$  dans la figure 3.22 en augmentant l’épaisseur de travail. Nous avons déterminé que

l'épaisseur la plus appropriée est 7 en utilisant notre critère d'évaluation.

	Epaisseur :	1	2	3	4	5	6	7	8	9
a curve	ND-FOM	0.147	0.603	5.472	5.705	18.241	29.435	<b>34.438</b>	25.826	21.582
	M-FOM	0.055	0.155	0.586	0.627	1.155	1.633	<b>1.798</b>	1.552	1.447
c curve	ND-FOM	0.203	1.078	3.233	10.996	<b>12.722</b>	10.159	10.124	10.984	9.205
	M-FOM	0.084	0.219	0.430	<b>0.987</b>	0.957	0.803	0.833	0.861	0.681
h curve	ND-FOM	0.158	0.381	2.739	2.904	4.133	6.575	<b>7.421</b>	7.191	6.624
	M-FOM	0.047	0.106	0.314	0.354	0.415	0.506	<b>0.562</b>	0.553	0.567
real leaf	ND-FOM	1.336	2.232	<b>2.405</b>	0.630	0.076	0.175	0.156	0.163	0.183
	M-FOM	0.260	0.395	<b>0.485</b>	0.470	0.478	0.429	0.391	0.373	0.374

TABLE 3.5 – Comparaison entre le profil de notre critère  $ND - FOM$  et celui de Marji [MS04]  $M - FOM$  en augmentant l'épaisseur de travail sur les courbes a, c et h dans la figure 3.22 la courbe de feuille dans la figure 3.31.

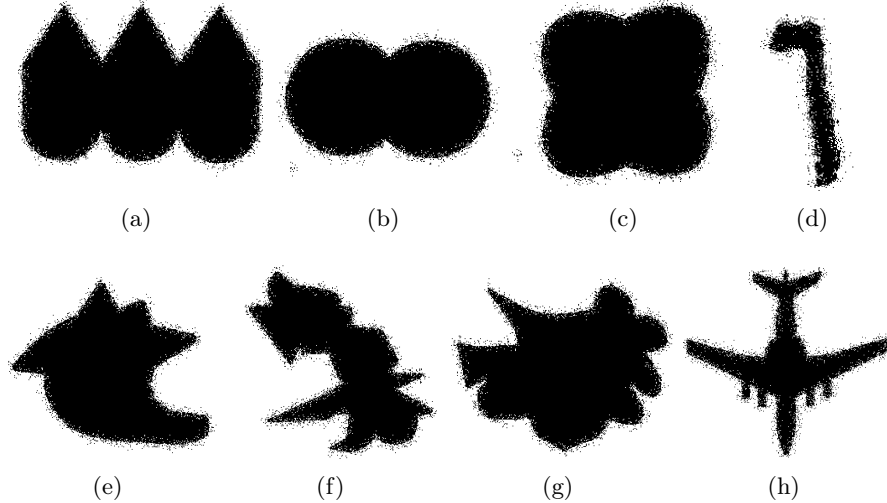


FIGURE 3.21 – Les images bruitées en entrée.

Nous avons testé notre critère et le critère de Marji [MS04] sur plusieurs courbes bruitées. Ils donnent les mêmes résultats dans presque 90% des cas. Nous trouvons que, notre critère donne des pics plus clairs que le critère de Marji dans les profils des critères d'évaluation. Considérons le cas d'une feuille dans le tableau 3.5. Il y a deux profils qui ont le même pic à l'épaisseur 3 dans leur profils. Pourtant, concernant le critère de Marji, il n'est pas facile de décider de l'épaisseur la plus appropriée pour la détection de points dominants parmi les valeurs d'épaisseur 3, 4 et 5. Au contraire, notre critère donne un vrai pic à l'épaisseur 3. C'est similaire en ce qui concerne la courbe h lorsqu'il y a 2 pics dans le profil du critère de Marji aux épaisseurs 7 et 9. Concernant la courbe c, il y a des différents pics entre les 2 profils. Le critère de Marji donne le premier pic à l'épaisseur 4 au lieu de l'épaisseur 5 comme notre critère. Donc, notre critère est bien adapté aux courbes bruitées. Notre profil de critère d'évaluation donne le même pic à l'épaisseur 4 (10.996 par rapport à 11.702 à l'épaisseur 5) (voir le tableau 3.5). Donc, il y a une convergence entre ces deux critères. La figure 3.24, le tableau 3.6 donnent les résultats concernant la courbe c de la figure 3.22 à 2 épaisseurs différentes : 4 et 5. Le nombre de points dominant est réduit considérablement de l'épaisseur 4 à l'épaisseur 5.

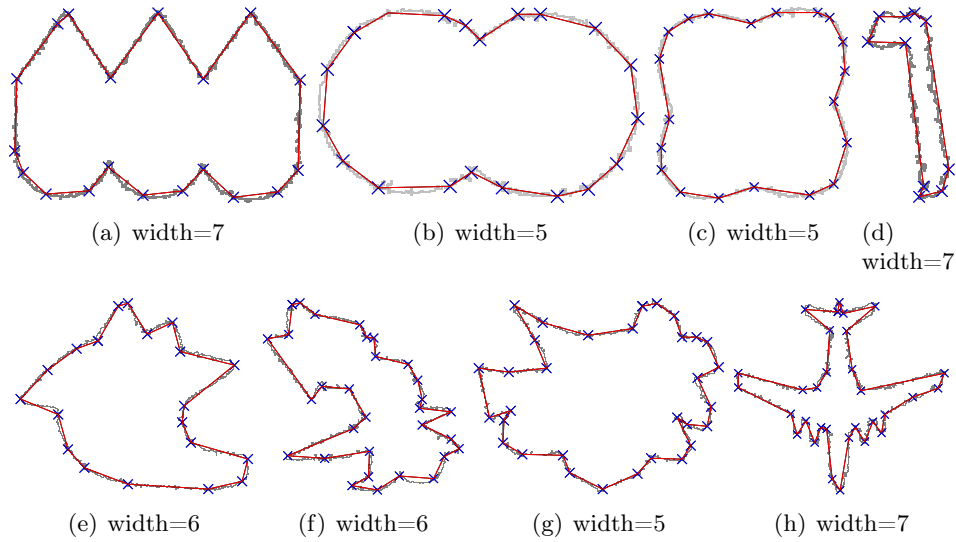


FIGURE 3.22 – Détection automatique de points dominants. L'épaisseur la plus appropriée est déterminée en testant des épaisseurs successives avec le critère ND\_FOM.

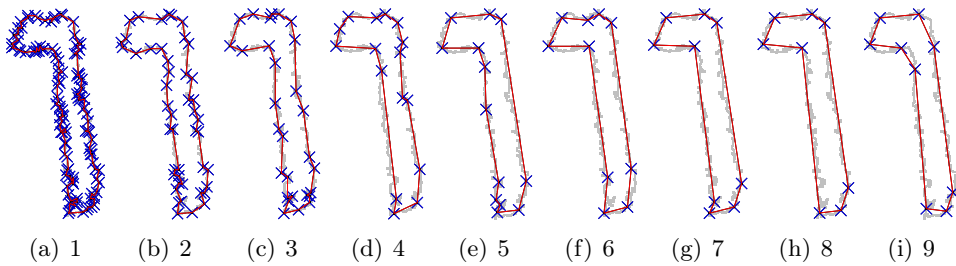


FIGURE 3.23 – Une séquence des résultats qui sont obtenus en augmentant l'épaisseur de travail de 1 à 9. L'épaisseur la plus appropriée est 7 en utilisant notre critère d'évaluation.

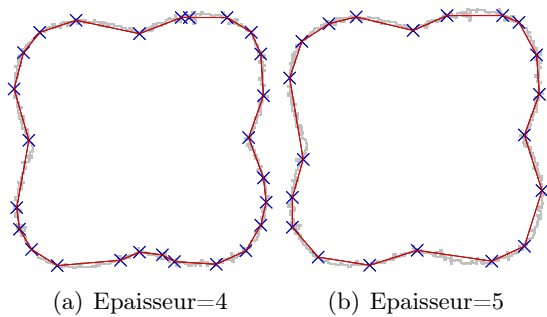


TABLE 3.6 – Résultats de la détection de points dominants sur la courbe (c) de la figure 3.22 à l'épaisseur 4 et 5.

<i>Epaisseur</i>	<i>ISE</i>	$L_\infty$	<i>CR</i>	nDPs
4	3554.38	5.316	59.231	26
5	5698.53	6.297	77	20

FIGURE 3.24 – Détection de points dominants sur la courbe c de la figure 3.22 à l'épaisseur 4 et 5.



## 3.5 Approches multi-épaisseurs

### 3.5.1 Une méthode multi-épaisseur pour la représentation polygonale des courbes bruitées

Dans cette section, une nouvelle méthode est proposée afin de répondre aux questions qui sont posées concernant la méthode précédente de détection de points dominants. Etant donnée une courbe discrète en entrée, quels sont les effets du paramètre d'épaisseur  $\nu$  sur les résultats de détection de points dominants? Comment pouvons nous comparer les différents résultats de cette méthode en changeant la valeur de ce paramètre? Quelle est la valeur de ce paramètre pour obtenir le meilleur résultat correspondant? Le reste de cette section essaiera de répondre à ces questions intéressantes.

#### Problèmes et restrictions de la méthode précédente

Plusieurs méthodes existantes pour la détermination des points dominants ne fonctionnent pas très bien avec des courbes bruitées en extrayant à partir des objets qui sont acquis dans des mauvaises conditions. Ces courbes ont beaucoup de points faiblement dominants. La méthode précédente peut être efficace avec des courbes bruitées en utilisant un paramètre d'épaisseur approprié. Pourtant, cette méthode ne donne pas de bons résultats lorsqu'elle s'applique aux courbes bruitées en utilisant le paramètre par défaut (l'épaisseur 0.8). La figure 3.25.c présente un résultat avec une courbe bruitée en utilisant le paramètre par défaut. C'est pourquoi, la qualité de détection ne correspond pas à notre attente. Une question naturelle se pose : *Quelle est l'épaisseur la plus appropriée pour travailler avec une courbe bruitée?* Dans la partie suivante, nous proposons une approche multi-épaisseur afin de répondre cette question.

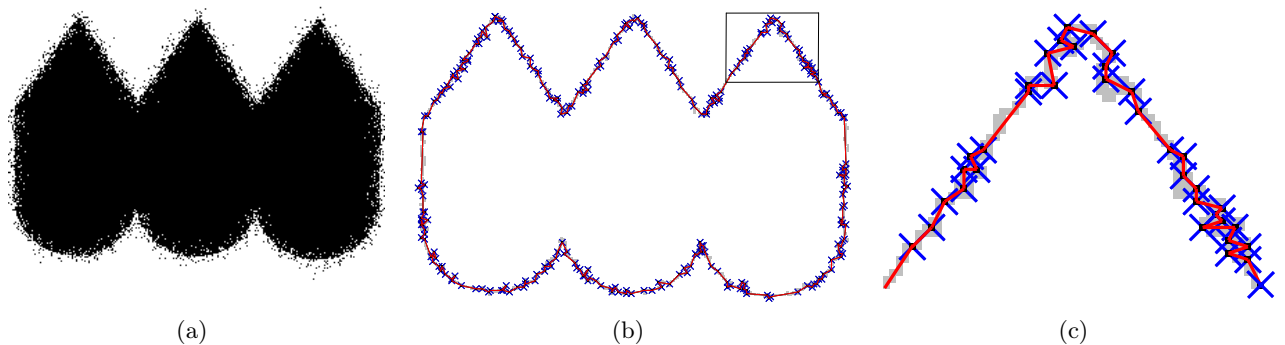


FIGURE 3.25 – a- Image en entrée, b- Courbe extraite avec des points dominants détectés en utilisant la méthode précédente à l'épaisseur 1, c- Une partie de la figure 3.25.b

#### La méthode proposée

Nous présentons une méthode sans paramètre pour la détection de points dominants et la représentation polygonale des courbes bruitées. Reposant sur la méthode précédente, cette méthode propose une approche multi-épaisseur afin de déterminer la meilleure épaisseur de travail pour chaque courbe donnée en entrée. En bref, pour chaque courbe donnée en entrée, nous allons examiner les représentations polygonales consécutives qui sont obtenues en augmentant la valeur du paramètre d'épaisseur de la méthode précédente. L'épaisseur la plus appropriée sera déterminée en utilisant un critère d'évaluation sur les représentations polygonales à chaque niveau.

Nous utiliserons aussi notre critère d'évaluation pour évaluer différentes méthodes de détection de points dominants ainsi que les méthodes de polygonalisation.

---

**Algorithme 11** : Méthode sans paramètre pour la représentation polygonale.

---

**Données** :  $C$  courbe discrète de  $n$  points

**Résultat** :  $optEpaisseur$  - l'épaisseur obtenue,  $P$  - représentation polygonale  
 $width = 1^4$ ;  $isLoop = TRUE$ ;  $maxFOM = 0$ ;

**pour**  $isLoop$  **faire**

Utiliser l'algorithme 10 (voir la page 73) pour obtenir  $P'$  comme la représentation polygonale de la courbe à l'épaisseur  $width$ ;

Calculer  $FOM_{ND}$ ;

**si**  $FOM_{ND} > maxFOM$  **alors**

└  $optEpaisseur = width$ ;  $maxFOM = FOM_{ND}$ ;  $P = P'$ ;

**else**

└  $isLoop = FALSE$ ;

└  $width++$ ;

---

Puis, nous proposons un algorithme sans paramètre pour segmenter une courbe pouvant être bruitée (voir l'algorithme 11). L'idée principale est de déterminer le premier pic dans le profil de notre critère d'évaluation qui est construit par un processus multi-épaisseur.

### Complexité

Supposons que  $M$  est l'épaisseur maximal qu'on travaille. La complexité de cette méthode est déterminée par cette formule :  $O(Mn \log n)$ . Nous considérons maintenant la relation entre  $M$  et  $n$ .

La boucle dans l'algorithme 11 se terminera à l'épaisseur la plus appropriée qui est adaptée au bruit et à la forme principale de la courbe. Pour l'épaisseur suivante, puisque la stabilité de la forme a été obtenue, le nombre des points dominants ne change plus donc le CR ne change plus, mais l'erreur d'approximation augmente considérablement à cause des faibles décalages concernant la position des points dominants qui ne sont pas bien localisés. Alors, le premier pic dans le profil de notre critère d'évaluation correspond localement à l'épaisseur la plus appropriée. Intuitivement, cette valeur correspond au niveau moyen de bruit de la courbe. Elle ne dépend pas de la longueur de la courbe. C'est pourquoi, le nombre des itérations dans cette boucle peut être considéré comme une valeur constante par rapport à la longueur de la courbe en entrée. En pratique, nous considérons  $M = 20$ . Par conséquent, les algorithmes 10 et 11 ont la même complexité en  $O(n \log n)$ .

### 3.5.2 Méthode adaptative multi-échelle reposant sur une stratégie de découpage-et-fusion

Dans cette section, nous présentons une nouvelle solution pour l'approximation polygonale et pour la décomposition d'une courbe en parties significatives. Cette méthode est basée sur une approche multi-épaisseur. Reposant sur la première méthode, une stratégie de découpage-et-fusion est utilisée afin de décomposer une courbe en parties significatives avec les points

---

4. Nous supposons que l'épaisseur de départ est 1 lorsqu'on travaille avec des courbes bruitées.

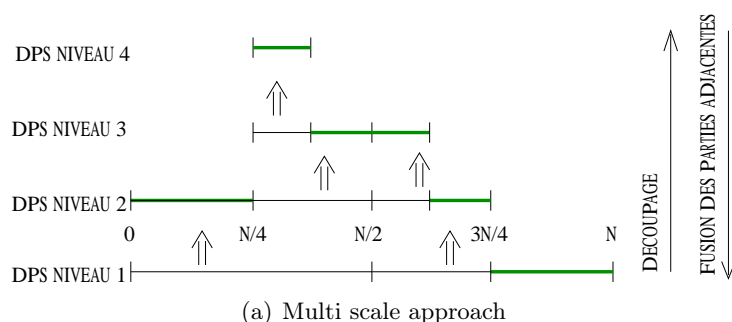


FIGURE 3.26 – Stratégie de division-et-fusion pour partitionner une courbe.

dominants correspondants. Pour chaque partie, nous allons déterminer l'échelle correspondante qui est la plus appropriée pour la représenter.

### Méthode proposée

La première méthode (voir la section 3.4) a proposé une solution pour la représentation polygonale et la détection de points dominants avec un paramètre fixe. Nous avons montré dans la section 3.5.3 qu'elle convient bien aux courbes normales qui ne sont pas bruitées lorsqu'on utilise le paramètre par défaut. De plus, le changement de valeur du paramètre d'épaisseur permet de travailler avec des courbes bruitées grâce à l'avantage de l'utilisation des segments flous. Donc, une question intéressante est proposée. Pour chaque courbe discrète, est ce qu'on peut décomposer la courbe en parties significatives telle que chaque partie correspond à son échelle la plus appropriée? Dans cette section, nous présentons une nouvelle méthode pour l'approche multi-échelle afin de répondre à cette question.

L'idée principale est de décomposer une courbe éventuellement bruitée en parties significatives avec lesquelles chaque partie sera représentée à l'échelle la plus appropriée. Un ensemble initial de points dominants est construit par l'ensemble des points dominants à l'épaisseur 1. Nous appliquons une procédure de division en utilisant un critère d'évaluation comme la condition afin de déterminer l'épaisseur la plus appropriée pour chaque partie de la courbe. En fusionnant les parties adjacentes qui ont la même échelle, la courbe est décomposée en parties significatives avec lesquelles les points dominants sont détectés à l'échelle correspondante la plus appropriée. Enfin, un processus de suppression est appliqué pour éliminer des points dominants intermédiaires dans l'ensemble initial. Ce processus est basé aussi sur l'utilisation d'un critère d'évaluation. Ce critère d'évaluation peut être le critère de Marji [MS04] ou bien notre critère (voir la section 3.3).

Nous présentons dans l'algorithme 12 une procédure récursive nommée SPLIT pour diviser une partie de la courbe si le critère d'évaluation sur la détection de points dominants à l'échelle suivante est mieux que celui à l'échelle courante. La figure 3.26 illustre comment utiliser la stratégie de division-et-fusion pour décomposer une courbe en parties significatives dans cet algorithme. Son idée principale est de comparer les valeurs du critère d'évaluation sur deux résultats de détection à l'échelle actuelle et à l'échelle suivante. Si la valeur obtenue à l'échelle suivante est plus grande que celle à l'échelle actuelle, cette partie de courbe sera divisée en 2 moitiés pour être considérée à l'échelle suivante en appliquant récursivement cette procédure. Sinon, l'épaisseur actuelle est considérée comme l'épaisseur la plus appropriée pour représenter cette partie de la courbe et cette procédure s'arrête à cette échelle.

Nous utilisons la notation  $D_k$  (resp.  $D$ ) pour indiquer l'ensemble des points dominants détectés à l'épaisseur  $k$  (resp. à travers l'approche multi-échelle);  $D_k[m]$  (resp.  $D[m]$ ) correspond au

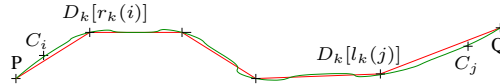


FIGURE 3.27 –  $EV_k(P, Q, i, j)$

$m^{ième}$  point dans l'ensemble correspondant. Les notations  $l_k(i)$  et  $r_k(i)$  (resp.  $l(i)$  et  $r(i)$ ) sont utilisées pour indiquer l'indice associée à  $D_k$  (resp.  $D$ ) le plus proche à gauche et à droite du  $i^{ième}$  point de la courbe en entrée. De plus, la notation  $EV_k(P, Q, i, j)$  (voir aussi la figure 3.27) est utilisée pour indiquer le critère d'évaluation qui est déterminé entre la partie de la courbe de  $P$  à  $Q$  et le polygone qui est construit à partir des 2 points  $P$ ,  $Q$  et l'ensemble des points dominants détectés à l'épaisseur  $k$  qui se trouve entre  $C_i$  et  $C_j$  ( $i$  et  $j$  sont 2 indices de la courbe entre  $P$  et  $Q$ ).

---

**Algorithme 12** : Procédure *SPLIT*.

---

**Données** :  $C$  courbe discrète de  $n$  points,  $D$  l'ensemble en entrée des points dominants,  $b$  (resp.  $e$ ) l'indice débutant (resp. à la fin),  $k$  l'épaisseur des segments flous

**Résultat** :  $D$  - Les points dominants détectés à travers l'approche multi-échelle  
**début**

$m = \lfloor \frac{b+e}{2} \rfloor$ ;

**si**  $EV_k(D[l(b)], D[r(m)], b, m) > EV_{k+1}(D[l(b)], D[r(m)], b, m)$  **alors**

└ Remplacer  $D[r(b)], \dots, D[l(m)]$  par  $D_k[r_k(b)], \dots, D_k[l_k(m)]$ ;

**sinon**

└ **si**  $r_k(b) \leq l_k(m)$  **alors**

└└ *SPLIT*( $C, b, m, k + 1$ );

**si**  $EV_k(D[l(m)], D[r(e)], m, e) > EV_{k+1}(D[l(m)], D[r(e)], m, e)$  **alors**

└ Remplacer  $D[r(m)], \dots, D[l(e)]$  par  $D_k[r_k(m)], \dots, D_k[l_k(e)]$ ;

**sinon**

└ **si**  $r_k(m) \leq l_k(e)$  **alors**

└└ *SPLIT*( $C, m, e, k + 1$ );

**fin**

---

Grâce à l'algorithme 12, nous proposons dans l'algorithme 13 une méthode pour la décomposition d'une courbe en parties significatives avec leurs points dominants correspondants. Premièrement, une séquence de l'ensemble de points dominants de l'épaisseur 1 à  $maxEpaisseur$  est construite en utilisant l'algorithme 10 (voir la page 73). L'ensemble des points dominants à l'épaisseur 1 est considéré comme l'ensemble initial des points dominants. Ensuite, nous appliquons la procédure récursive qui est présentée dans l'algorithme 12 afin de déterminer la décomposition de la courbe en différentes échelles. Puis, un processus de suppression est appliqué pour retirer des points dominants initiaux si possible en comparant le critère d'évaluation obtenu sur l'ensemble des points dominant avec et sans ce point initial. Finalement, en fusionnant les parties adjacentes qui s'arrêtent à une même échelle, nous obtenons la décomposition de la courbe en parties significatives avec des points dominants détectés à l'échelle correspondante.

---

5.  $D$  contient des points dominants avec les informations de position et d'échelle.

---

**Algorithme 13** : Approche multi-échelle pour décomposer une courbe discrète en parties significatives.

---

**Données** :  $C$  courbe discrète de  $n$  points,  $maxEpaisseur$  l'épaisseur maximale de travail

**Résultat** :  $D$  l'ensemble des points dominants

début

**pour**  $i=1$  à  $maxEpaisseur$  **faire**

    Utiliser l'algorithme 10 pour déterminer  $D_k$  - l'ensemble des points dominants  
    détectés à l'épaisseur  $k$  ;

$D = D_1$  ;  $D = SPLIT(C, D, start, 0, n, 1)$ <sup>5</sup>;

  Fusionner les parties adjacentes à une même échelle;

fin

---

## Complexité

Supposons que  $M$  est l'épaisseur maximal qu'on travaille. A chaque épaisseur, la procédure *SPLIT* perd totalement un temps de  $O(n)$ . Donc, la complexité de cette méthode est déterminée par cette formule :  $O(Mn \log n)$ . Nous considérons maintenant la relation entre  $M$  et  $n$ .

Pour chaque partie de la courbe, nous pouvons utiliser les mêmes arguments que dans la section précédente pour conclure que l'épaisseur correspondante peut être considérée comme une constante par rapport au nombre de points de cette partie de la courbe. Donc, l'épaisseur maximale peut être considérée aussi comme une constante par rapport au nombre de points de la courbe. En pratique, nous considérons  $M = 10$ . Donc, la complexité de la méthode qui est présentée dans l'algorithme 13 est aussi  $O(n \log n)$ .

### 3.5.3 Expérimentations

Nous présentons dans cette section l'expérimentation avec les deux méthodes non-paramétriques ci-dessous pour la polygonalisation des courbes bruitées : méthode multi-épaisseur et méthode adaptative.

## Expérimentations

### Expérimentations sur les courbes non-bruitées

Nous avons implémenté et expérimenté les deux méthodes non-paramétriques. La figure 3.28 montre quelques résultats obtenus avec les deux méthodes. Le table 3.7 présente ces résultats qualitativement.

**Expérimentations sur les courbes bruitées** Les deux méthodes sont expérimentées sur les courbes bruitées qui sont dégradées en utilisant le modèle de Kanungo. La figure 3.29 présente les résultats sur une courbe déconnectée. Un autre test est présenté dans la figure 3.31. Nous avons appliqué aussi nos méthodes proposées sur les images bruitées qui sont dégradées à partir de la base d'images SIID<sup>6</sup>. La figure 3.30 et le tableau 3.8 montrent les résultats obtenus avec ces deux méthodes. Nous constatons que la méthode adaptative fonctionne mieux la méthode multi-épaisseur dans la condition de distribution non-uniforme de bruit. Elle fonctionne aussi bien que la méthode multi-épaisseur dans la condition de distribution uniforme de bruit.

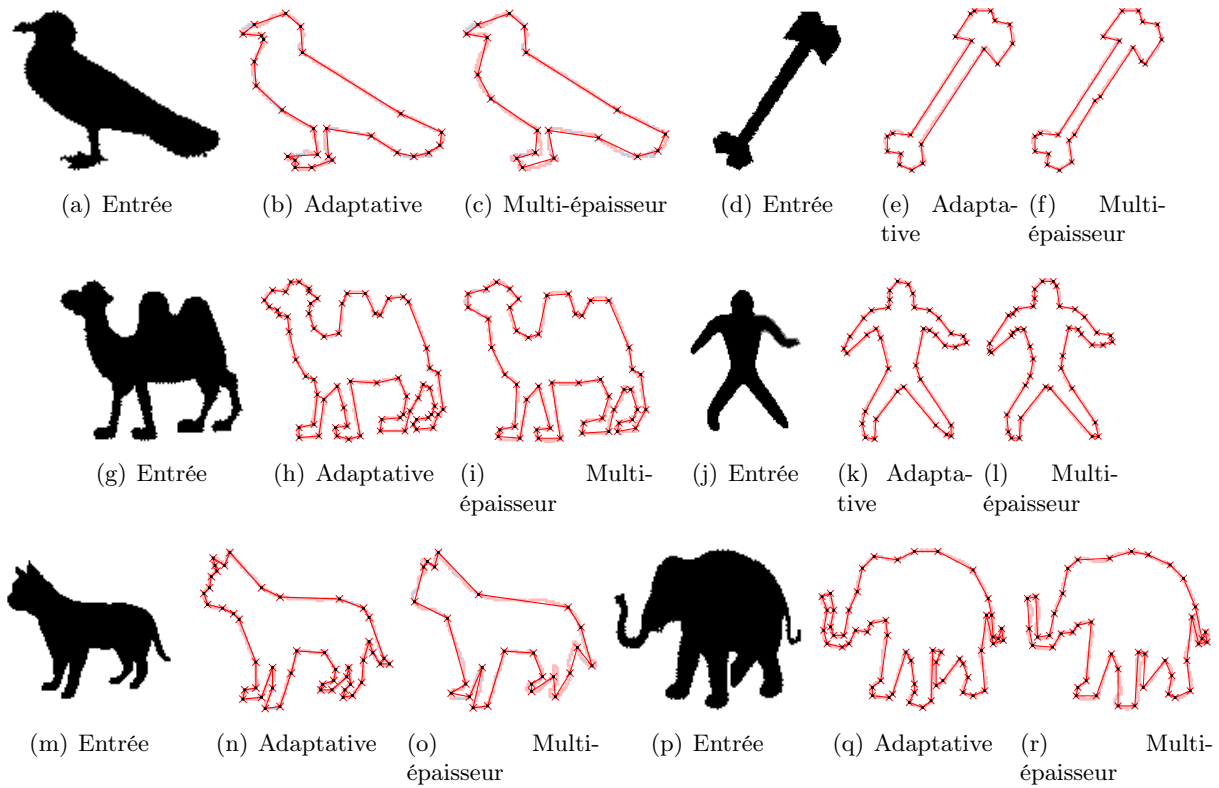


FIGURE 3.28 – Résultats qui sont obtenus par les méthodes proposées.

Figures	n	Méthode	nDP	ISE	$L_\infty$	CR	Marji FOM	ND FOM
3.28.b, 3.28.c	526	Adaptative	26	286.404	2.433	20.23	<b>1.429</b>	<b>11.884</b>
		Multi-épaisseur	19	596.037	3.073	27.684	1.286	11.585
3.28.e, 3.28.f	518	Adaptative	19	391.32	2.378	27.26	<b>1.899</b>	<b>21.78</b>
		Multi-épaisseur	22	308.12	2.378	23.545	1.799	17.818
3.28.k, 3.28.l	866	Adaptative	67	276.496	1.414	12.925	<b>0.604</b>	<b>5.522</b>
		Multi-épaisseur	50	518.643	2	17.32	<b>0.578</b>	<b>5.009</b>
3.28.n, 3.28.o	574	Adaptative	39	189.931	1.455	14.718	<b>1.140</b>	<b>11.535</b>
		Multi-épaisseur	44	165.203	1.368	13.045	1.030	9.82
3.28.q, 3.28.r	578	Adaptative	47	180.959	1.374	12.298	<b>0.836</b>	<b>7.479</b>
		Multi-épaisseur	25	763.467	3.051	23.12	0.700	5.306
3.28.t, 3.28.u	794	Adaptative	56	291.976	1.661	14.179	<b>0.688</b>	<b>5.878</b>
		Multi-épaisseur	43	652.143	2.696	18.465	0.523	3.580

TABLE 3.7 – Résultats obtenus dans la figure 3.28.

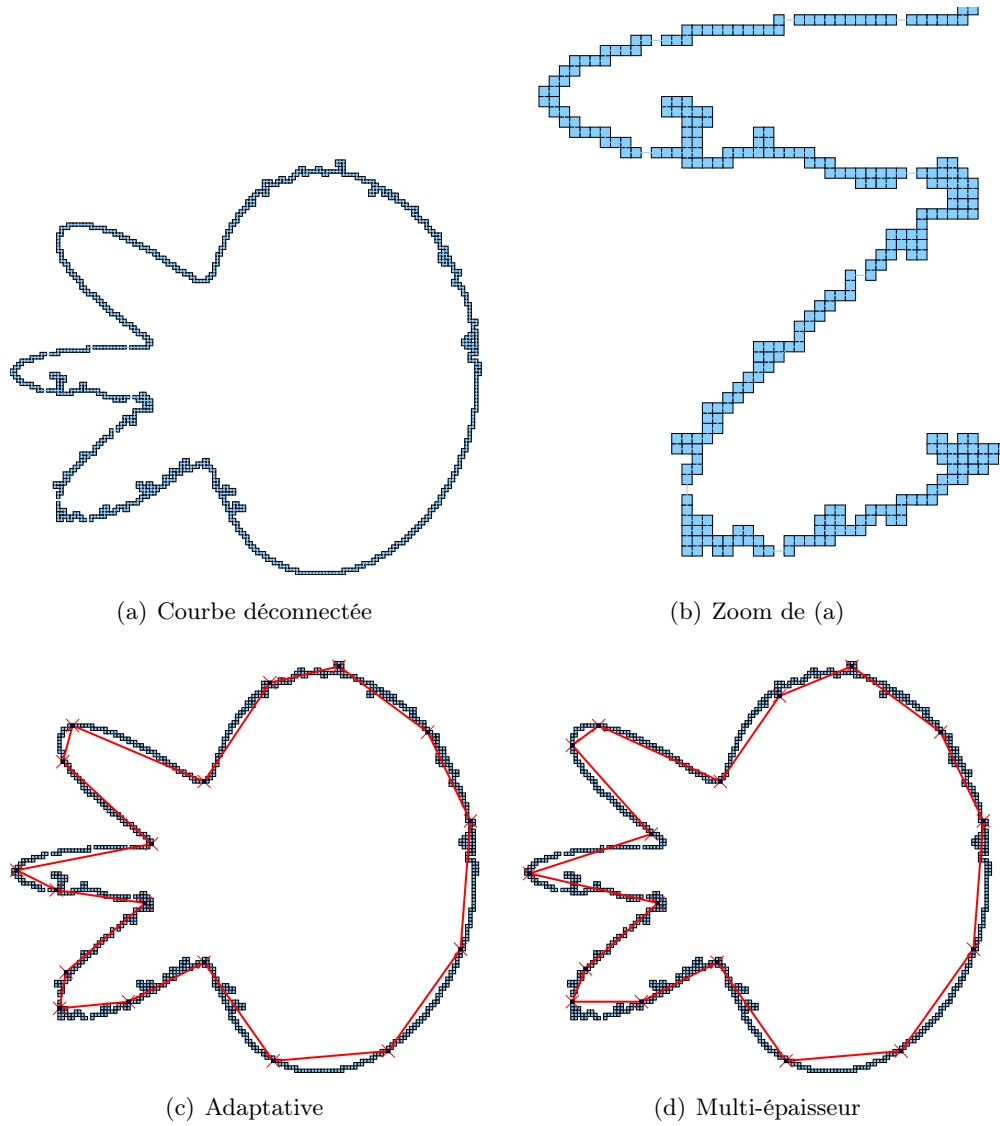


FIGURE 3.29 – Résultat obtenue sur une courbe déconnectée.

Figures	n	Méthode	nDP	ISE	$L_\infty$	CR	Marji's FOM	ND's FOM
3.30.b, 3.30.c	764	Adaptative	24	922.165	3.682	31.833	<b>1.099</b>	<b>9.501</b>
		Multi-épaisseur	23	1185.96	5.059	33.217	0.930	6.108
3.30.e, 3.30.f	794	Adaptative	15	662.467	2.828	48.267	<b>3.517</b>	<b>60.011</b>
		Multi-épaisseur	13	1033.61	3.191	55.692	3.001	52.376
3.30.h, 3.30.i	1106	Adaptative	36	2274.1	4.391	30.722	<b>0.415</b>	<b>2.904</b>
		Multi-épaisseur	33	3051.99	4.391	33.515	0.368	2.809
3.30.k, 3.30.l	996	Adaptative	30	1777.93	3.527	33.2	0.62	<b>5.836</b>
		Multi-épaisseur	31	1584.62	4.427	32.129	<b>0.651</b>	4.727
3.30.n, 3.30.o	684	Adaptative	30	558.447	2.977	22.8	0.931	7.129
		Multi-épaisseur	29	569.945	2.977	23.586	<b>0.976</b>	<b>7.733</b>
3.30.q, 3.30.r	658	Adaptative	10	1674.66	6.41	65.8	<b>2.585</b>	<b>26.54</b>
		Multi-épaisseur	11	1687.87	6.207	59.818	2.12	20.43

TABLE 3.8 – Résultats obtenus dans la figure 3.30.

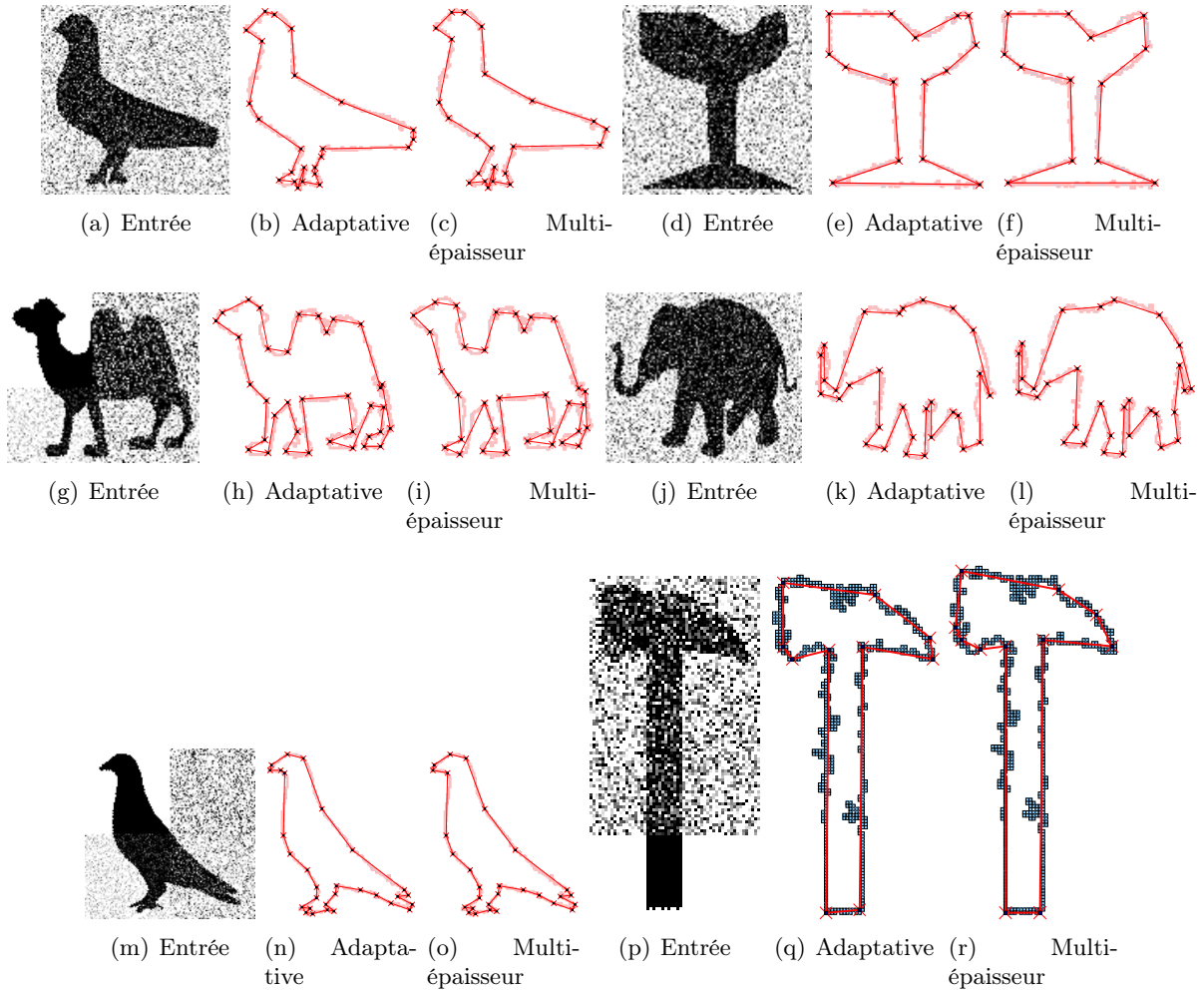


FIGURE 3.30 – Travail avec des courbes bruitées qui sont dégradées en utilisant le modèle de Kanungo [KHB<sup>+</sup>94].



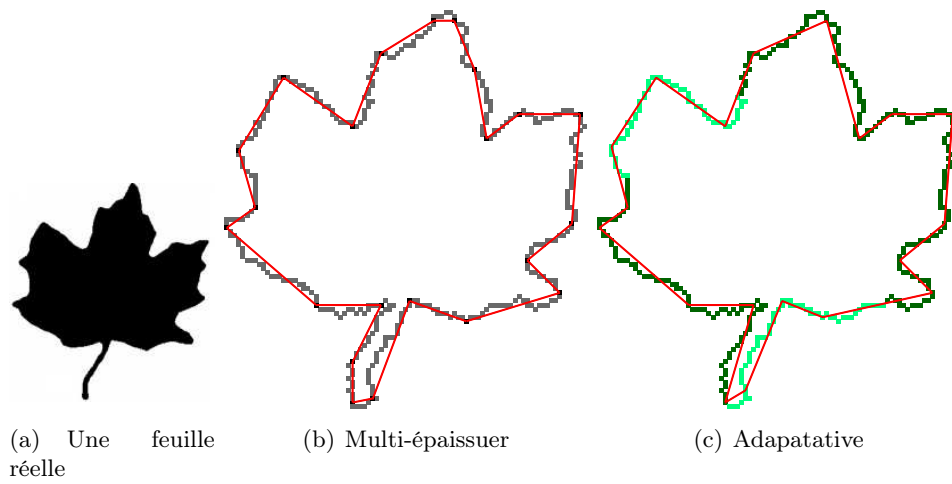


FIGURE 3.31 – Résultats obtenus sur une image réelle, le bruit est généré en utilisant le modèle Kanungo.

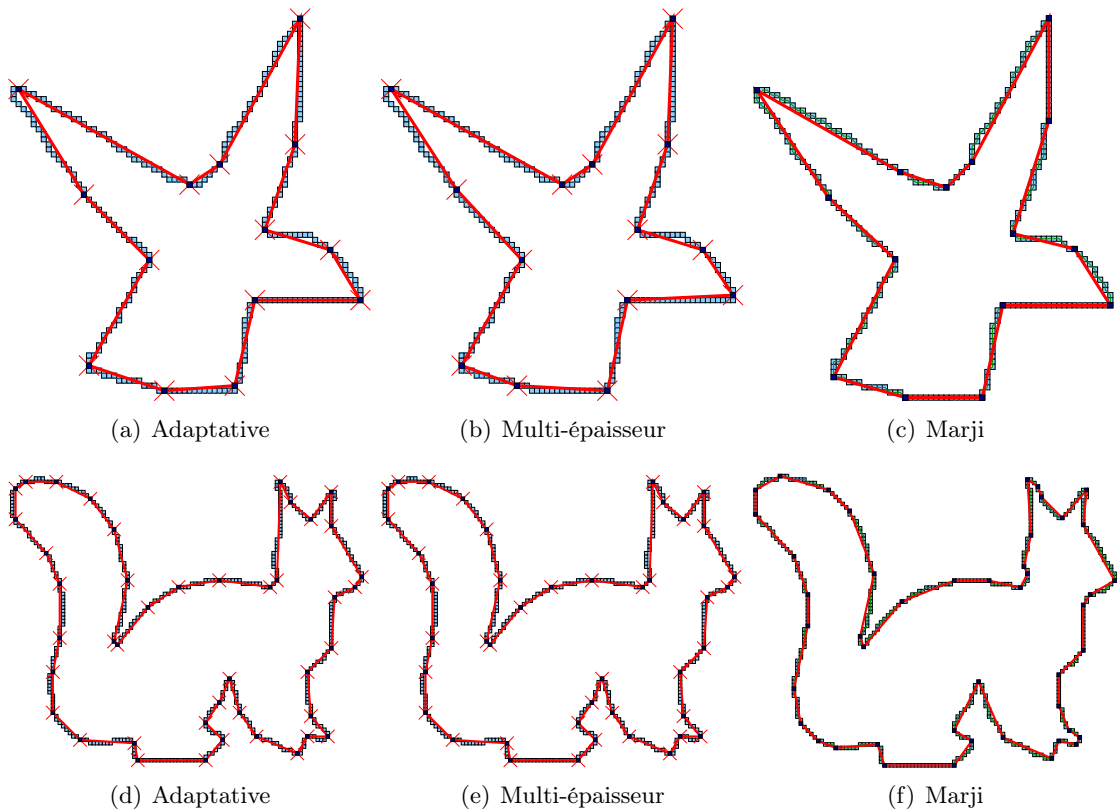


FIGURE 3.32 – Comparaison avec la méthode de Marji [MS04].

Fig.	$n$	Method	$nDP$	$ISE$	$L_\infty$	$CR$	FOM	Marji's FOM	ND's FOM
3.32.a-c	261	Adaptive	14	68.311	1.691	18.643	<b>0.273</b>	<b>5.088</b>	<b>56.092</b>
		Multi-width	43	74.116	1.691	18.643	0.252	4.689	51.699
		Marji	15	64.283	1.479	17.4	0.271	4.709	55.386
3.32.d-f	427	Adaptive	43	110.751	1	9.930	0.90	0.890	<b>8.842</b>
		Multi-width	43	110.751	1	9.930	0.90	0.890	<b>8.842</b>
		Marji	46	85.307	1.410	9.283	<b>0.11</b>	<b>1.010</b>	6.649
3.33.a-c	525	Adaptive	36	150.386	1.455	14.583	0.097	<b>1.414</b>	<b>14.172</b>
		Multi-width	21	490.795	2.654	25	0.051	1.273	11.995
		Debled	18	2664.41	7.555	29.167	0.011	0.319	1.233
3.33.d-f	339	Adaptive	25	153.769	1.671	13.56	<b>0.088</b>	<b>1.196</b>	<b>9.702</b>
		Multi-width	14	769.598	3.372	24.214	0.031	0.762	5.471
		Debled	17	1339.77	5.129	20	0.015	0.298	1.164
3.34.a-d	103	Adaptive	21	12.126	0.848	4.905	<b>0.404</b>	<b>1.984</b>	<b>11.468</b>
		Multi-width	21	12.126	0.848	4.905	<b>0.404</b>	<b>1.984</b>	<b>11.468</b>
		Sarkar	20	16.143	1.052	5.15	0.319	1.643	8.039
		Ansari	29	12.865	1.213	3.552	0.276	0.981	2.872

TABLE 3.9 – Comparaisons sur les courbes des figures 3.33, 3.32, 3.34 avec Debled et al. [DRTW04], Sarkar [Sar93], Ansari et al. [AwH91] et Marji et al. [MS04].

### Comparaisons

Nous présentons dans les figures 3.32, 3.33 et le tableau 3.9 les comparaisons entre les méthodes proposées et les méthodes non-paramétriques : celle de Marji [MS04], celle de Sarkar [Sar93], celle de Ansari et al. [AwH91] et la méthode de multi-ordre [DRTW04]. Ces résultats sont considérés en utilisant plusieurs de critères d'évaluation. Le tableau 3.9 prouve que les méthodes proposées donnent des meilleurs résultats que ceux de la méthode multi-ordre. Pourtant, la méthode multi-ordre fonctionne plus rapidement que nos méthodes grâce à sa complexité linéaire. Nos méthodes donnent aussi des meilleurs résultats que ceux de Sarkar [Sar93] et de Ansari [AwH91]. Nos méthodes donnent des résultats comparables par rapport aux ceux de la méthode de Marji dans la condition non-bruitée. De plus, la méthode de Marji ne fonctionne pas bien dans des conditions de bruit. Nos méthodes donnent des bon résultats avec les courbes non-bruitées et de plus elles peuvent travailler bien dans la condition de bruit dans laquelle les autres méthodes peuvent être inefficaces.

### Application à la vectorisation des courbes bruitées

Nous proposons d'utiliser ces méthodes pour une application à la vectorisation des courbes bruitées qui sont extraites à partir d'images bruitées. La figure 3.35 présente quelques exemples sur des images bruitées.

## 3.6 Conclusions

La détection de points dominants dans les courbes discrète est un sujet intéressant dans la représentation des formes. Elle donne directement des applications en analyse d'images, vectorisation, et mise en correspondance des formes.

6. <http://www.lems.brown.edu/vision/researchAreas/SIID>

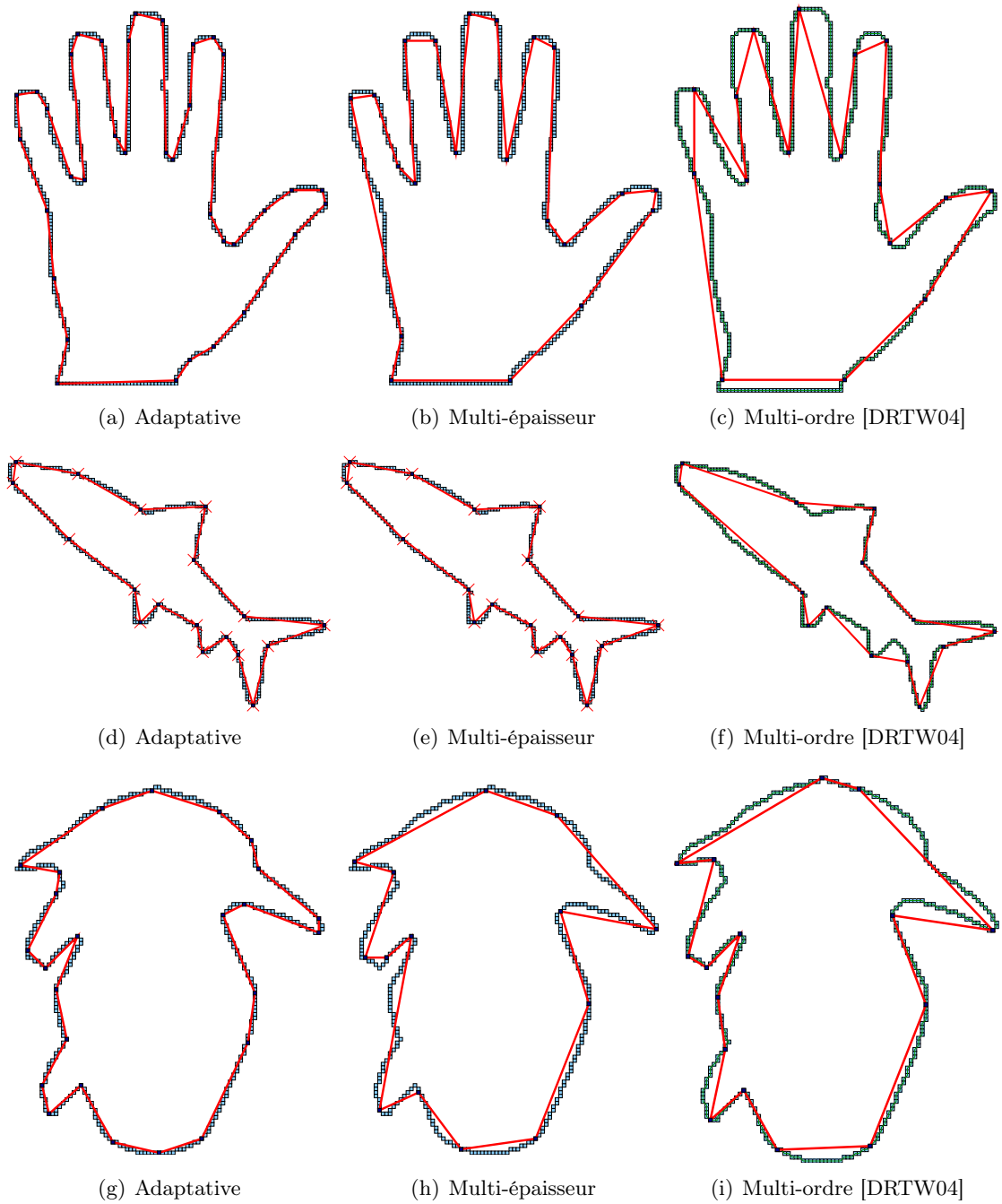


FIGURE 3.33 – Comparaison avec la méthode multi-ordre [DRTW04].

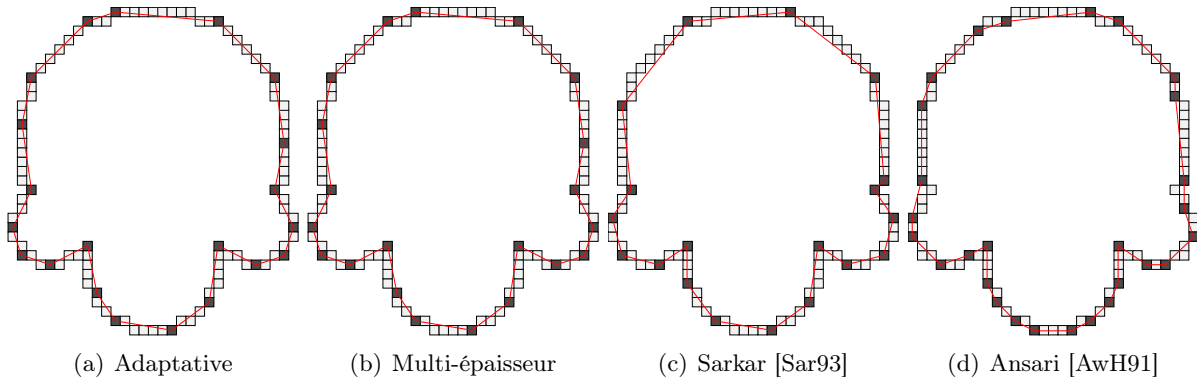


FIGURE 3.34 – Comparaison avec la méthode de Sarkar [Sar93] et celle de Ansari [AwH91].

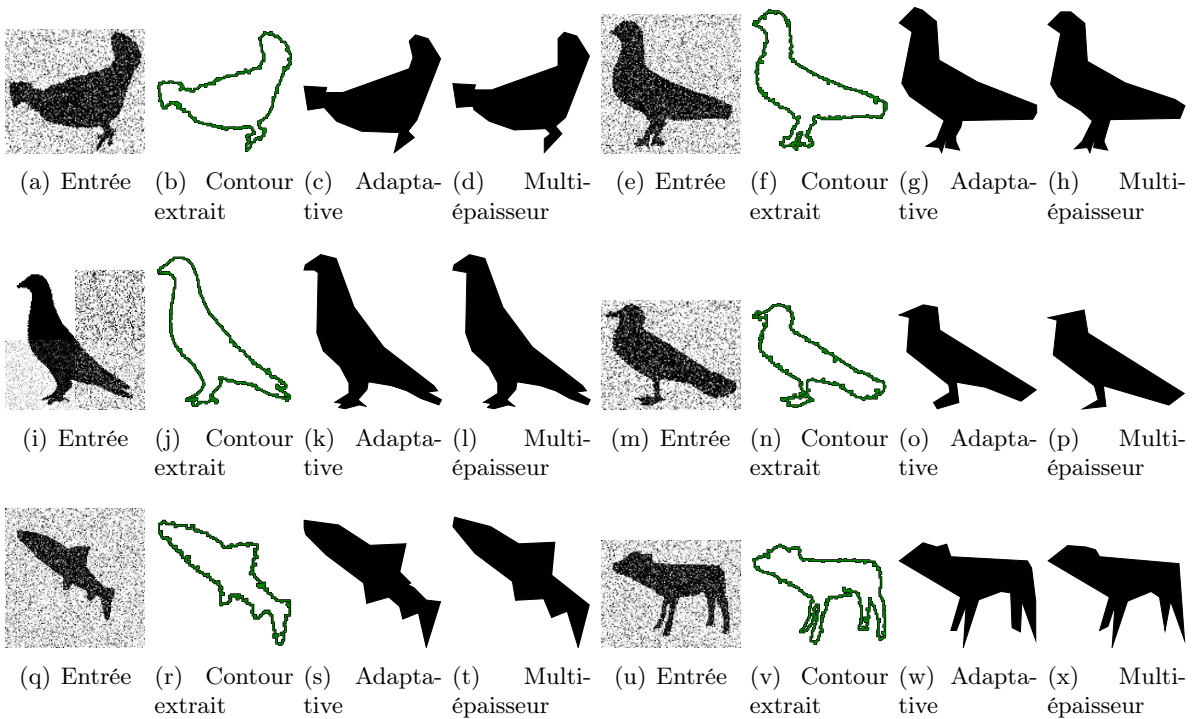


FIGURE 3.35 – Vectorisation en utilisant les méthodes proposées.

Dans ce chapitre, nous avons présenté une nouvelle méthode pour la détection de points dominants reposant sur des résultats récents de la géométrie discrète. Cette méthode utilise la notion de segment flou donc elle peut travailler naturellement avec des courbes bruitées. Notre méthode a été comparée avec plusieurs méthodes de la littérature. Elle peut donner de meilleurs résultats par rapport à plusieurs méthodes en utilisant le paramètre par défaut (l'épaisseur 0.9). De plus, cette méthode peut aussi s'utiliser à travers des échelles différentes en changeant la valeur du paramètre d'épaisseur. Cette méthode a été publiée dans la conférence ICIP'09 [NDR09a]. Une version étendue a été acceptée pour publication dans le journal Pattern Recognition [NDR10c]. Cette méthode efficace a été mise en ligne par le travail d'un ingénieur. Les utilisateurs peuvent la tester directement sur cette adresse suivante : <http://kerrecherche.iutsd.uhp-nancy.fr/~ines/DominantPoint.html>.

Nous avons aussi présenté un cadre de travail sans paramètre pour s'adapter aux courbes bruitées reposant sur une approche multi-épaisseur. Nous avons introduit un nouveau *critère d'évaluation* pour évaluer les résultats obtenus avec des méthodes différentes de détection de points dominants ainsi que des méthodes de polygonalisation. Cette proposition a été introduite par l'étude des critères existants dans la littérature [Sar93, Ros97, MS04]. Une méthode multi-épaisseur a été proposée aussi en utilisant ce critère d'évaluation pour s'adapter aux courbes bruitées. Elle a été publiée dans la conférence IWCIA'09 [NDR09b].

Nous avons aussi présenté une solution pour la décomposition d'une courbe en parties significatives avec lesquelles les points dominants sont détectés à l'échelle correspondante. Chaque partie significative correspond à son échelle naturelle. C'est pourquoi, le contour d'objet peut être mieux représenté par les points dominants au travers du cadre multi-échelle. Cette méthode adaptative a été publiée dans la conférence ICPR'10 [NDR10e].

Dans le futur, nous nous intéresserons à la recherche d'applications pertinentes pour appliquer ces méthodes à la vision par ordinateur. Spécifiquement, nous nous intéresserons à une application de mise en correspondance reposant sur nos méthodes de polygonalisation.



# Chapitre 4

## Cercle discret et applications

### Sommaire

---

<b>4.1</b>	<b>Introduction</b>	<b>104</b>
<b>4.2</b>	<b>Etat de l'art</b>	<b>104</b>
4.2.1	Reconnaissance des cercles discrets	105
4.2.2	Mesure de circularité	106
4.2.3	Décomposition d'une courbe en arcs et segments de droite	107
<b>4.3</b>	<b>Reconnaissance des cercles discrets</b>	<b>107</b>
4.3.1	Représentation des arcs dans l'espace des tangentes	108
4.3.2	Propriétés d'un arc dans la représentation modifiée de l'espace des tangentes	108
4.3.3	Détection des arcs/cercles discrets	111
4.3.4	Approximation de l'erreur	112
4.3.5	Segmentation en arcs discrets	117
4.3.6	Expérimentations	118
<b>4.4</b>	<b>Segmentation d'une courbe bruitée en arcs discrets</b>	<b>120</b>
4.4.1	Détection du bruit non-supervisée	120
4.4.2	Détection des arcs dans une courbe bruitée	127
4.4.3	Expérimentations	129
<b>4.5</b>	<b>Décomposition d'une courbe en arcs et segments de droite</b>	<b>133</b>
4.5.1	Idée principale	133
4.5.2	Analyse des configurations	133
4.5.3	Algorithme proposé	136
4.5.4	Discussion	136
4.5.5	Expérimentations	137
<b>4.6</b>	<b>Mesure de la circularité</b>	<b>139</b>
4.6.1	Similarité des courbes polygonales	142
4.6.2	La représentation d'un cercle réel dans l'espace des tangentes	143
4.6.3	Mesure de circularité et notre algorithme	144
4.6.4	Expérimentations	146
<b>4.7</b>	<b>Conclusion</b>	<b>154</b>

---

## 4.1 Introduction

Les cercles et les arcs discrets sont des objets de base qui apparaissent souvent dans les images. Donc, l'étude de ces primitives est un sujet important en géométrie discrète et aussi en analyse d'images. Dans ce chapitre, nous allons présenter de nouveaux résultats concernant l'étude de ces primitives intéressantes.

Tout d'abord, nous proposons un algorithme linéaire reposant sur une approche de géométrie discrète pour segmenter une courbe en arcs et cercles discrets. Cette méthode utilise une représentation originale des arcs et des cercles discrets. En utilisant cette représentation, nous transformons le problème de reconnaissance des arcs discrets en un problème de reconnaissance de droites discrètes et nous en déduisons un algorithme de segmentation d'une courbe discrète en arcs discrets.

Ensuite, nous présentons dans cette section une extension de cette méthode pour travailler avec des courbes bruitées. La méthode précédente utilise un paramètre en entrée pour la polygonalisation de la courbe discrète. Ce paramètre est utilisé pour déterminer l'approximation polygonale. Il permet de tenir compte du bruit présent dans la courbe. Nous appliquons un résultat récent [KL09a] à propos de la détection de bruit pour déterminer l'épaisseur appropriée de travail. Plusieurs approches seront proposées en considérant deux hypothèses différentes de distribution de bruit. Nous appliquons aussi cette méthode pour la détection des arcs dans des images.

Par la suite, nous proposons une nouvelle solution pour la décomposition d'une courbe discrète en arcs et en segments de droite en temps  $O(n \log n)$ . C'est une solution reposant sur la reconnaissance des cercles, arcs discrets et la détection des points dominants présentée dans le chapitre précédent. La détection des points dominants est utilisée afin de renforcer la qualité des résultats de la décomposition.

Nous introduisons aussi dans ce chapitre une nouvelle mesure de circularité reposant sur l'outil de Arkin et al. [ACH<sup>+</sup>91], Latecki et al. [LL00] concernant la mesure de similarité des formes qui est construite sur la représentation dans un espace des tangentes. Nous proposons ensuite un algorithme linéaire qui utilise cette mesure pour mesurer la circularité des courbes fermées. Cette méthode peut être aussi considérée comme une méthode pour la reconnaissance des objets circulaires.

L'organisation de ce chapitre est comme suit. La section 4.2 présente un état de l'art sur la reconnaissance des arcs discrets, la segmentation en arcs discrets, la décomposition d'une courbe en arcs et segments, et la mesure de circularité. La section 4.3 présente une nouvelle méthode de reconnaissance des arcs discrets et la segmentation d'une courbe en arcs discrets. La section 4.4 propose une extension de ce travail aux courbes bruitées. Ensuite, la section 4.5 introduit une solution pour décomposer une courbe en arcs et segments. Puis, nous présentons dans la section 4.6 une nouvelle méthode pour mesurer la circularité d'une courbe discrète.

## 4.2 Etat de l'art

En ce qui concerne l'étude des cercles et des arcs discrets, nous pouvons recenser trois problèmes classiques. Le premier problème est de déterminer si un ensemble de points est une discrétisation d'un cercle réel. Le deuxième problème estime la ressemblance entre la courbe étudiée et une discrétisation d'un cercle réel. Le troisième problème est la décomposition d'une courbe discrète en primitives arc et segment de droite. Nous allons présenter les méthodes issues de la littérature en ce qui concerne ces problèmes dans les sections ci-dessous.



### 4.2.1 Reconnaissance des cercles discrets

La reconnaissance de primitives est un sujet important en analyse d'images et en géométrie discrète. C'est la première étape pour la décomposition d'un contour d'images en primitives telles que les droites, les cercles, ... De très nombreux auteurs se sont intéressés au problème de la reconnaissance de segments de droites discrètes en utilisant différentes approches (cf. [KR04] pour un état de l'art des différents algorithmes).

La reconnaissance des arcs de cercles et des cercles discrets est aussi un sujet intéressant en analyse d'images et en géométrie discrète mais pour lequel il y a eu moins de travaux que pour les segments de droites. Nous rappelons brièvement ci-dessous les méthodes existantes dans la littérature pour la reconnaissance de cercles discrets (voir aussi [Coe02] pour un état de l'art de plusieurs méthodes existantes).

Généralement, un ensemble de points appartient à un cercle discret s'il existe un processus de discrétisation qui détermine la discrétisation d'un cercle réel contenant l'ensemble de point. En ce qui concerne le sujet de la reconnaissance de cercle discret, nous pouvons le diviser en deux problèmes distincts :

- Test de circularité : ce problème répond à la question suivante : "Cet ensemble de points correspond il à un cercle réel?"
- Reconnaissance de cercle : l'objectif de ce problème est de construire l'ensemble de cercle réels solution.

**Nakamura et Aizawa [NA84]** : Nakamura et al. ont proposé un algorithme de reconnaissance de cercle discret reposant sur certaines définitions de cercle discret. Dans ce travail, ils définissent un cercle discret comme l'ensemble des points discrets qui se trouvent les plus proches d'un cercle réel. Cette méthode cherche le centre du cercle après avoir effectué un nombre fini d'étapes. Cependant, cet algorithme a une complexité exponentielle dans le pire cas. Dans la conclusion, les auteurs proposent quelques pistes pour réduire la complexité en temps polynomial.

**Kim [Kim84a]** : Kim a proposé une première approche de la reconnaissance de disques discrets. Cette méthode [Kim84a] détecte si un ensemble de points dans une image  $N \times N$  est un disque discret en temps  $O(n^3)$ . La méthode repose sur l'étude des propriétés géométriques d'un disque discret.

**Kim et Anderson [KA84]** : Kim et Anderson ont proposé une deuxième stratégie pour la reconnaissance de disques discrets. Cette nouvelle méthode optimise le calcul et permet d'obtenir une complexité de  $O(n^2)$ .

**Fisk [Fis86]** : En se basant sur le problème classique de séparation des arcs de la géométrie algorithmique, Fisk [Fis86] a aussi proposé un algorithme pour la reconnaissance d'un disque discret en temps  $O(n^2)$ .

**Kovalevsky [Kov90]** : En se basant sur la séparation de deux ensembles de points par un cercle, Kovalevsky [Kov90] a proposé une méthode pour la reconnaissance d'un cercle discret en temps  $O(n^2 \log n)$ . Cette complexité n'est pas donnée par les auteurs dans cet article. Puisque la méthode se ramène à un calcul d'intersection de  $n^2$  inéquations en dimension 2, Coeurjolly [Coe02] a obtenu cette complexité en utilisant l'algorithme de Preparata et Shamos [PS85].

**Sauer [Sau93]** : Sauer a présenté une méthode linéaire pour décider si une courbe discrète est un cercle discret. Il a montré que ce problème est équivalent à la résolution d'un système de  $n$  inégalités en dimension 3 de complexité linéaire en utilisant l'algorithme de Megiddo [Meg84]. Cependant, sa méthode ne permet pas de travailler avec des arcs.

**Damaschke [Dam95]** : Damaschke a redéfini le problème du test de circularité en un système d'inéquations linéaires de dimension 3. Ensuite, il a présenté un algorithme linéaire pour reconnaître un arc discret en se basant sur le théorème de Megiddo.

**Worring [WS95]** : a proposé une méthode de segmentation d'un cercle discret en utilisant un processus de fenêtre de taille fixe. Ils propose un calcul de l'ensemble des cercles réels pour des morceaux de courbes de taille  $r$  et utilise un estimateur statistique pour évaluer le rayon d'un cercle solution pour la courbe totale donnée en entrée. Evidemment, la qualité des résultats dépend beaucoup du paramètre  $r$ .

**Coeurjolly [CGRT04]** : a transformé le problème de la reconnaissance d'un cercle en un problème de recherche d'un point 2D qui appartient à l'intersection de  $n^2$  demi-plans. Il a utilisé une étape de pré-traitement en utilisant des propriétés arithmétiques des courbes discrètes afin de réduire la complexité à  $O(n^{4/3} \log n)$ .

**Roussillon [RScT09]** : a proposé un algorithme linéaire pour la reconnaissance des arcs discrets qui respectent une des 3 conditions suivantes : les rayons sont fixés, ou les contours touchent un point fixe, ou bien les centres appartiennent à une droite fixe. Cependant, ces restrictions ne permettent pas à cette méthode de travailler efficacement dans le cas général. De plus, il a proposé une autre méthode qui n'est pas linéaire ( $O(n^{4/3})$ ) mais qui fonctionne dans le cas général.

### 4.2.2 Mesure de circularité

A côté du problème de reconnaissance et segmentation en arcs, la mesure de la circularité est aussi un problème intéressant. Par rapport à cette problématique, on peut retrouver dans la littérature les différentes terminologies suivantes : compacité [Ros74a, KA84], rondeur [Che02], circularité [Har74, RST08]. Similairement à Roussillon, nous préférons utiliser la terminologie "circularité" car elle évoque le degré de circularité de la courbe ou de la région étudiée.

**Rosenfeld [Ros74a]** : Ce problème a été proposé initialement dans le travail de Rosenfeld [Ros74a]. Il a proposé une division entre le carré du périmètre d'une figure et son aire ( $P^2/A$ ) comme une mesure de non-compacité. Cette mesure vient de la mesure classique correspondante pour mesurer la non-compacité d'un objet dans la géométrie Euclidienne. Dans la géométrie classique, cette mesure est toujours supérieure ou égale à  $4\pi$ , elle prend la valeur minimale si et seulement si l'objet est un disque. En géométrie discrète, le périmètre  $P$  d'une région  $F$  peut être déterminé de deux façons. La première façon dépend du nombre de points du contour de  $F$ . La seconde façon suit le contour de  $F$  et compte 1 pour chaque déplacement vertical ou bien horizontal,  $\sqrt{2}$  pour chaque déplacement diagonal. Une mesure de circularité de  $F$  peut être déduite comme  $4\pi A/P^2$ . Rousillon [RST10] a indiqué que cette mesure classique montre des désavantages en pratique. De plus, nous trouvons que cette mesure n'est pas une bonne mesure de circularité en géométrie discrète. Premièrement, elle dépend de comment le périmètre est défini. Deuxièmement, elle prend la valeur minimale pour un octogone ou diamant discret, et elle n'est pas minimale pour un cercle discret.

**Haralick [Har74]** : Haralick a introduit une autre mesure de circularité en considérant la variable aléatoire  $R$  de la distance entre le centre et le bord du périmètre. Supposons que  $\mu_R$  (resp.  $\sigma_R$ ) est la moyenne (resp. la variance) de ces distances. Il a montré que  $\mu_R/\sigma_R$  est une mesure possible de circularité d'une région discrète. Haralick a proposé aussi 4 propriétés principales pour construire une bonne mesure de circularité des courbes fermées.

**Kim et Anderson [KA84]** : Kim et al. ont proposé une nouvelle mesure de circularité pour des régions discrètes convexes. Cette mesure est construite à partir de son algorithme de reconnaissance de disque. Supposons que  $P$  est une région convexe,  $P_0$  correspond au plus petit disque qui contient  $P$ . Il appelle  $A_P$  (resp.  $A_{P_0}$ ) est la superficie de  $P$  (resp.  $P_0$ ). Il a proposé  $m_p = \frac{A_P}{A_{P_0}}$

comme une mesure de circularité.

**Roussillon et al. [RST08]** : Roussillon et al. ont proposé une nouvelle mesure de circularité pour des courbes discrètes fermées reposant sur le problème de séparation des cercles. A partir d'une courbe discrète 4-connexe, il a construit 2 ensembles finis des points S et T en regardant les points intérieurs de la courbe. Afin de vérifier si S et T sont séparé par un cercle, il les a transformés vers  $S'$  et  $T'$  dans l'espace dual. La hauteur minimale entre 2 enveloppes convexes  $CH(S')$  et  $CH(T')$  est proposée comme une mesure de circularité. Il a proposé aussi un algorithme pour mesurer la circularité en se basant sur cette mesure en temps  $O(n \log n)$ .

### 4.2.3 Décomposition d'une courbe en arcs et segments de droite

Un problème important dans la vision par ordinateur est l'extraction de caractéristiques significatives à partir du contour de l'image. C'est une première étape pour construire un descripteur de l'image de haut niveau. Plusieurs méthodes existantes utilisent les points dominants ou bien les segments de droite comme des caractéristiques significatives. Les arcs et les segments de droite sont des objets de base qui apparaissent souvent dans l'image spécialement dans les images de dessin technique. Une combinaison des arcs et des segments de droite est donc une bonne solution pour construire un descripteur de l'image. Celui-ci évite le problème où un arc est représenté par de nombreux segments de droite ou bien par de nombreux points dominants.

Dans la littérature, plusieurs méthodes ont été proposées pour la segmentation en arcs. Nous pouvons les classer en trois groupes principaux : l'approximation en arc, la transformation circulaire de Hough, et l'extension progressive d'arc.

**L'approximation en arc** [RW89, Ell02] est une approche reposant sur la vectorisation d'une image binaire en une suite de petits segments de droite. Tout d'abord, il collectionne un groupe de vecteurs qui est considéré appartenir à un arc et ensuite il utilise des processus itératifs afin de trouver les paramètres de l'arc approximé. La limitation de cette approche est la sensibilité aux distorsions de vecteur.

**La transformation circulaire de Hough** : Les autres méthodes [PH95, Lea92] ont utilisé la transformation circulaire de Hough afin de détecter des arcs. L'idée principale est d'utiliser l'espace des paramètres des cercles pour détecter les paramètres dont les valeurs cumulatives sont hautes comme candidats arc ou cercle. Ces méthodes donnent de meilleurs résultats que la première approche. Pourtant, elles dépendent d'un seuil et elles ont un coût élevé de traitement.

**L'extension progressive d'arc** : La dernière approche [WD98, SLC04] utilise des contraintes d'arc pour améliorer la segmentation.

Concernant les autres méthodes, Salmon et al. [SDRW06] ont proposé une méthode reposant sur le profil de courbure. Ils utilisent une notion de courbure discrète reposant sur les segments flous. L'idée principale est de construire le profil de courbure et ensuite détecter des points clés associés à ce profil pour la reconstruction.

## 4.3 Reconnaissance des cercles discrets

Nous présenterons dans cette section une méthode linéaire de reconnaissance d'un cercle discret ou d'un arc discret reposant sur une approche de géométrie discrète. Premièrement, une polygonalisation est appliquée en temps linéaire sur la courbe à analyser. Ensuite, nous utilisons une transformation, proposée par Arkin [ACH<sup>+</sup>91] et reprise par Latecki [LL00], pour représenter le polygone obtenu dans un nouvel espace, appelé **espace des tangentes**. Nous démontrerons qu'une suite de cordes d'un cercle correspond à une suite de points colinéaires

dans cet espace. Le problème de la reconnaissance d'un cercle ou d'un arc de cercle peut donc être transformé en un problème de reconnaissance de segment de droite discrète. Comme le processus de discrétisation d'un cercle discret cause des distorsions dans cet espace, nous utilisons l'algorithme de reconnaissance de segment flou [DRFRD06] pour détecter les segments de droites discrètes dans cet espace.

Le reste de cette section est organisée comme suit. Le paragraphe suivant présente comment transformer un polygone dans l'espace des tangentes et propose certaines propriétés d'un arc de cercle dans cette représentation. Le paragraphe 4.3.3 propose une méthode linéaire pour reconnaître un arc/cercle discret. Le paragraphe 4.3.4 considère l'erreur d'approximation effectuée par cette méthode. Enfin l'algorithme de segmentation et des expérimentations sont présentés dans les sections 4.3.5 et 4.3.6.

### 4.3.1 Représentation des arcs dans l'espace des tangentes

Arkin [ACH<sup>+</sup>91], Latecki [LL00] ont proposé une représentation de l'espace des tangentes comme outil de mesure de similarité pour la reconnaissance des formes. En s'inspirant de ce travail, nous proposons une représentation modifiée de l'espace des tangentes afin de représenter un polygone. La différence entre les deux représentations est que dans cette représentation, la longueur du polygone n'est pas normalisée sur l'axe  $Ox$ . Nous allons montrer dans la prochaine section qu'un arc ou un cercle possède des propriétés importantes dans notre représentation modifiée de l'espace des tangentes. Pour une meilleure compréhension, nous rappelons dans cette section les notions relatives à la construction de l'espace des tangentes d'un polygone.

#### Représentation modifiée de l'espace des tangentes

Supposons que  $C = \{C_i\}_{i=0}^n$  est une courbe polygonale avec  $\alpha_i = \angle(\overrightarrow{C_{i-1}C_i}, \overrightarrow{C_iC_{i+1}})$  et  $l_i$  la longueur du segment  $C_iC_{i+1}$ . Nous avons  $\alpha_i > 0$  si  $C_{i+1}$  est à droite de  $\overrightarrow{C_{i-1}C_i}$ ,  $\alpha_i < 0$  sinon.

On considère la transformation qui associe au polygone  $C$  de  $\mathbb{Z}^2$  le polygone  $T$  de  $\mathbb{R}^2$  formé des segments  $T_{i2}T_{(i+1)1}, T_{(i+1)1}T_{(i+1)2}$  pour  $i$  de 0 à  $n - 1$  avec

$$\begin{aligned} T_{02} &= (0, 0), \\ T_{i1} &= (T_{(i-1)2}.x + l_{i-1}, T_{(i-1)2}.y), \text{ pour } i \text{ de } 1 \text{ à } n, \\ T_{i2} &= (T_{i1}.x, T_{i1}.y + \alpha_i), \text{ pour } i \text{ de } 1 \text{ à } n - 1. \end{aligned}$$

Supposons que  $L = \sum_{i=0}^{n-1} l_i$ ,  $l'_i = \frac{l_i}{L}$ ,  $i \in \{0, \dots, n - 1\}$ .

Nous appelons que *l'espace des tangentes est normalisé* dans le cas les points  $T_{i1}$ ,  $T_{i2}$  sont construits de la façon suivante :

$$\begin{aligned} T_{i1} &= (T_{(i-1)2}.x + l'_{i-1}, T_{(i-1)2}.y), \quad i \in \{1, \dots, n\}, \\ T_{i2} &= (T_{i1}.x, T_{i1}.y + \alpha_i), \quad i \in \{1, \dots, n - 1\} \quad (n \text{ si } C \text{ est une courbe fermée, } C_0 \equiv C_n) \end{aligned}$$

La *fonction de tangente* de  $C$ , notée par  $T(C)$ ,  $T(C) : [0, 1] \rightarrow \mathbb{R}^2$ , est la fonction qui représente la courbe qui correspond au polygone  $T$ .

La figure 4.1 montre comment transférer une courbe polygonale vers sa représentation dans l'espace des tangentes.

### 4.3.2 Propriétés d'un arc dans la représentation modifiée de l'espace des tangentes

Les théorèmes ci-dessous permettent d'exhiber les propriétés de la représentation d'un polygone correspondant à un arc de cercle dans l'espace modifié des tangentes.

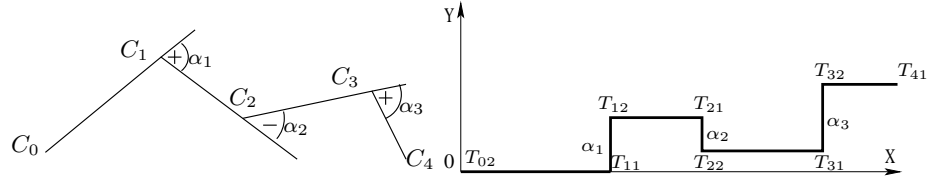


FIGURE 4.1 – De gauche à droite : a. Courbe polygonale donnée en entrée, b. Représentation de cette courbe polygonale dans l'espace des tangentes.

**Théorème 4.3.1** *Supposons que  $C = \{C_i\}_{i=0}^n$  est un polygone,  $\alpha_i = \angle(\overrightarrow{C_{i-1}C_i}, \overrightarrow{C_iC_{i+1}})$  et tel que  $\alpha_i \leq \alpha \leq \frac{\pi}{4}$  pour  $i \in \{1, \dots, n-1\}$ . La longueur de  $C_iC_{i+1}$  est  $l_i$ , pour  $i \in \{1, \dots, n\}$ . Alors,  $C$  est un polygone qui approxime un cercle ou un arc de cercle si et seulement si la séquence d'équations ci-dessous est satisfaite.*

$$\frac{l_i + l_{i+1}}{\alpha_i} = 2R\Omega, 1 \leq i \leq n-1$$

où  $0.9744 < \frac{\sin \alpha/2}{\alpha/2} \leq \Omega < 1$

**Preuve 4.3.1** *Tout d'abord, nous avons deux remarques suivantes. La fonction  $\sin(x)$  est concave sur l'intervalle  $0, \frac{\pi}{4}$ , la fonction  $\frac{\sin(x)}{x}$  est décroissant sur l'intervalle  $0, \frac{\pi}{4}$ .*

*Considérons la figure 4.2.*

- ① **Supposons que  $C$  est un polygone dont les extrémités se trouvent sur un arc de cercle.**

*Soient  $H_0, H_1$  les projections orthogonales du centre  $O$  respectivement sur les segments  $C_0C_1$  et  $C_1C_2$  (cf. Fig. 4.2). Alors, nous avons :  $\alpha_1 = \angle H_0OC_1 + \angle C_1OH_1$ . D'autre part,  $\sin(\angle H_0OC_1) = \frac{l_0}{2R}$ ,  $\sin(\angle C_1OH_1) = \frac{l_1}{2R}$ . Donc,  $\frac{l_0}{2R} + \frac{l_1}{2R} = \sin(\angle H_0OC_1) + \sin(\angle C_1OH_1) \geq 2 \sin \frac{\angle H_0OC_1 + \angle C_1OH_1}{2} = 2 \sin \frac{\alpha_1}{2}$ . Puisque  $\frac{\sin x}{x}$  est décroissant sur l'intervalle  $0, \frac{\pi}{4}$ , donc  $\sin \frac{\alpha_1}{2} \geq \frac{\alpha_1}{2} \frac{\sin \frac{\alpha}{2}}{\frac{\alpha}{2}}$ . Par la suite,  $\frac{l_0+l_1}{2R} \geq \alpha_1 \frac{\sin \frac{\alpha}{2}}{\frac{\alpha}{2}}$ . D'autre part,  $\sin(\angle H_0OC_1) < \angle H_0OC_1$  et  $\sin(\angle C_1OH_1) < \angle C_1OH_1$ , donc  $\frac{l_0+l_1}{2R} < \alpha_1$ . En bref, nous avons  $\frac{l_0+l_1}{\alpha_1} = 2R\Omega$  où  $1 > \Omega > \frac{\sin \frac{\alpha}{2}}{\frac{\alpha}{2}}$ . En considérant les segments successifs de  $C$ , l'implication est prouvée.*

- ② **Si  $\frac{l_i+l_{i+1}}{\alpha_i} = 2\bar{R}\Omega$ , où  $\frac{\sin \alpha/2}{\alpha/2} < \Omega < 1$ , est satisfait pour  $1 \leq i \leq n-1$ .**

*On considère  $O_1$  (respectivement  $O_2$ ) l'intersection entre 2 lignes bissectrices de  $C_0C_1$  et  $C_1C_2$  (respectivement  $C_1C_2$  et  $C_2C_3$ ). En utilisant l'hypothèse ci-dessus, nous avons  $2 * O_1C_2\Omega = \frac{l_0+l_1}{\alpha_1}$ . Similairement,  $2 * O_2C_2\Omega = \frac{l_1+l_2}{\alpha_2}$ . Donc, nous avons  $O_1C_2 \simeq O_2C_2$ . Alors,  $O_1 \equiv O_2$ . Nous en déduisons donc que  $C$  est un polygone dont les sommets sont sur un arc de cercle.*

**Théorème 4.3.2** *Supposons que  $C$  est une courbe polygonale. On considère le polygone  $T(C)$ , correspondant à sa représentation dans l'espace modifié des tangentes, constitué des segments  $T_{i2}T_{(i+1)1}, T_{(i+1)1}T_{(i+1)2}$  pour  $i$  de 0 à  $n-1$ .  $\{M_i\}_{i=1}^{n-1}$  est l'ensemble des points centraux de  $\{T_{i2}T_{(i+1)1}\}_{i=1}^{n-1}$ . Alors  $\frac{l_1+l_2}{\alpha_1} = \frac{l_2+l_3}{\alpha_2} = \dots = \frac{l_{n-1}+l_n}{\alpha_{n-1}}$  si et seulement si  $\{M_i\}_{i=0}^{n-1}$  est un ensemble de points colinéaires.*

**Preuve 4.3.2 Preuve :** *Considérons la figure 4.3. Soit  $I = \{I_i\}_{i=1}^{n-1}$  avec  $I_i$  l'intersection entre  $T_{(i-1)2}T_{(i+1)1}$  et  $T_{i1}T_{i2}$ .*

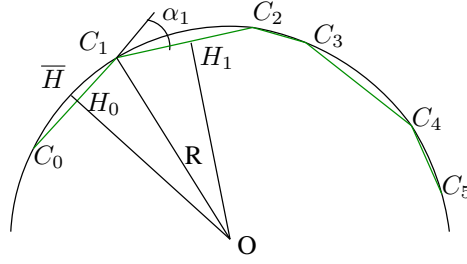


FIGURE 4.2 – Un polygone inscrit dans un cercle réel.

Par le théorème de Thalès nous avons :  $\frac{T_{(i-1)2}T_{i1}}{T_{i2}T_{(i+1)1}} = \frac{T_{i1}I_i}{I_iT_{i2}}$

$$\Leftrightarrow \frac{\frac{1}{2}T_{(i-1)2}T_{i1}}{\frac{1}{2}T_{i2}T_{(i+1)1}} = \frac{T_{i1}I_i}{I_iT_{i2}} \Leftrightarrow \frac{M_{i-1}T_{i1}}{T_{i2}M_i} = \frac{T_{i1}I_i}{I_iT_{i2}}$$

De plus (selon le théorème de Thalès),  $M_{i-1}M_i$  passe par  $I_i$  pour tout  $i$  de 1 à  $n-1$  (\*)

$$D'autre part, nous avons : \frac{T_{(i-1)2}T_{i1}}{T_{i2}T_{(i+1)1}} = \frac{T_{i1}I_i}{I_iT_{i2}} \Leftrightarrow \frac{l_i}{l_{i+1}} = \frac{T_{i1}I_i}{I_iT_{i2}} \Leftrightarrow \frac{l_i+l_{i+1}}{T_{i2}T_{(i+1)1}} = \frac{T_{i1}I_i+I_iT_{i2}}{I_iT_{i2}} \Leftrightarrow \frac{l_i+l_{i+1}}{T_{i2}T_{(i+1)1}} = \frac{\alpha_i}{I_iT_{i2}} \Leftrightarrow \frac{l_i+l_{i+1}}{\alpha_i} = \frac{T_{i2}T_{(i+1)1}}{I_iT_{i2}} \quad \forall i \in \{1, \dots, n-1\} (**)$$

① Si cette condition  $\frac{l_1+l_2}{\alpha_1} = \frac{l_2+l_3}{\alpha_2} = \dots = \frac{l_{n-1}+l_n}{\alpha_{n-1}}$  est satisfaite.

Grâce à (\*\*), nous avons :  $\frac{T_{12}T_{21}}{I_1T_{12}} = \dots = \frac{T_{i2}T_{(i+1)1}}{I_iT_{i2}} = \dots = \frac{T_{(n-1)2}T_{n1}}{I_{(n-1)}T_{(n-1)2}}$  Les triangles rectangles suivants sont donc semblables :  $\triangle T_{12}T_{21}I_1, \dots, \triangle T_{i2}T_{(i+1)1}I_i, \dots, \triangle T_{(n-1)2}T_{n1}I_{(n-1)}$ . Par conséquent :

$$\begin{aligned} - \frac{T_{12}T_{21}}{I_1T_{12}} = \dots = \frac{T_{i2}T_{(i+1)1}}{I_iT_{i2}} = \dots = \frac{T_{(n-1)2}T_{n1}}{I_{(n-1)}T_{(n-1)2}} \\ - \angle T_{12}T_{21}I_1 = \dots = \angle T_{i2}T_{(i+1)1}I_i = \dots = \angle T_{(n-1)2}T_{n1}I_{n-1} \end{aligned}$$

Comme ces triangles ci-dessous sont semblables :  $\triangle T_{12}T_{21}I_1, \dots, \triangle T_{i2}T_{(i+1)1}I_i, \dots, \triangle T_{(n-1)2}T_{n1}I_{n-1}$  et  $M_i$  est le milieu de  $T_{i2}T_{(i+1)1}$ ,  $\forall i \in \{1, \dots, n-1\}$ , on a alors :  $\angle I_1M_1T_{21} = \dots = \angle I_iM_iT_{(i+1)1} = \dots = \angle I_{n-1}M_{n-1}T_{n1}$ . Par conséquent, grâce à (\*), et au théorème de Thalès,  $\{M_i\}_{i=1}^{n-1}$  est un ensemble des points colinéaires.

② Si  $M = \{M_i\}_{i=1}^{n-1}$  est un ensemble de points colinéaires.

Grâce à (\*),  $M = \{M_i\}_{i=1}^{n-1}$  et  $I = \{I_i\}_{i=1}^{n-1}$  est un ensemble de points colinéaires.

$\Rightarrow$  En appliquant le théorème de Thalès, nous avons :  $\angle T_{12}M_1I_1 = \dots = \angle T_{i2}M_iI_i = \dots = \angle T_{(n-1)2}M_{n-1}I_{n-1}$ .

$\Rightarrow$  Ces triangles rectangulaires sont semblables :

$$\triangle T_{12}M_1I_1, \dots, \triangle T_{i2}M_iI_i, \dots, \triangle T_{(n-1)2}M_{n-1}I_{n-1}.$$

$$\Rightarrow \text{Nous avons : } \frac{T_{12}M_1}{T_{12}I_1} = \dots = \frac{T_{i2}M_i}{T_{i2}I_i} = \dots = \frac{T_{(n-1)2}M_{n-1}}{T_{(n-1)2}I_{n-1}}$$

$$\Leftrightarrow \frac{T_{12}T_{21}}{I_1T_{12}} = \dots = \frac{T_{i2}T_{(i+1)1}}{I_iT_{i2}} = \dots = \frac{T_{(n-1)2}T_{n1}}{I_{(n-1)}T_{(n-1)2}}$$

$$\Leftrightarrow \frac{l_1+l_2}{\alpha_1} = \frac{l_2+l_3}{\alpha_2} = \dots = \frac{l_{n-1}+l_n}{\alpha_{n-1}}$$

Grâce aux théorèmes 4.3.1 et 4.3.2, nous en déduisons le théorème 4.3.3 ci-dessous.

**Théorème 4.3.3** Supposons que  $C = \{C_i\}_{i=0}^n$  est un polygone,  $\alpha_i = \angle(\overrightarrow{C_{i-1}C_i}, \overrightarrow{C_iC_{i+1}})$  et tel que  $\alpha_i \leq \alpha \leq \frac{\pi}{4}$  pour  $i \in \{1, \dots, n-1\}$ . On considère le polygone  $T(C)$ , correspondant à sa représentation dans l'espace des tangentes, constitué des segments  $T_{i2}T_{(i+1)1}, T_{(i+1)1}T_{(i+1)2}$  pour  $i$  de 0 à  $n-1$ .  $\{M_i\}_{i=0}^{n-1}$  est l'ensemble de points centraux de  $\{T_{i2}T_{(i+1)1}\}_{i=1}^{n-1}$ . Alors,  $C$  est un polygone qui approxime un cercle ou un arc de cercle si et seulement si  $\{M_i\}_{i=0}^{n-1}$  est un ensemble de points colinéaires.

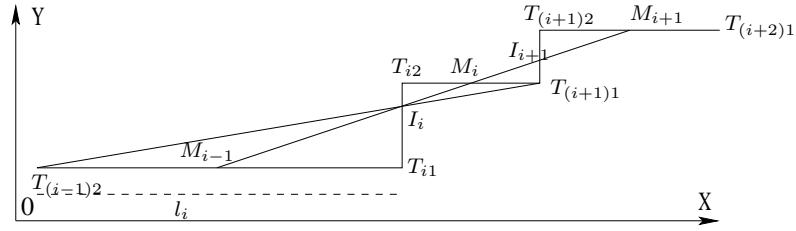


FIGURE 4.3 – Représentation d'un polygone dans l'espace modifié des tangentes (voir le théorème 4.3.3).

Nous utilisons ce théorème afin de reconnaître un cercle ou un arc de cercle discret en appliquant dans l'espace modifié des tangentes l'algorithme de reconnaissance de segments flous. Dans la section 4.3.3, nous proposons une méthode linéaire, issue de ce cadre théorique, pour la reconnaissance d'un cercle ou un arc de cercle discret.

### 4.3.3 Détection des arcs/cercles discrets

#### Algorithme proposé

L'idée principale est de travailler sur la représentation de la courbe discrète à analyser dans l'espace modifié des tangentes. Dans cette représentation, l'ensemble des points centraux  $M = \{M_i\}_{i=1}^{n-1}$  (voir la section 4.3.2) sera construit. Ensuite, nous utilisons une procédure linéaire [DRFRD06] qui est étendue pour travailler avec des nombres réels pour tester la colinéarité de ces points. Si la réponse est positive, la courbe polygonale d'entrée est un arc discret. Afin de pouvoir reconnaître des cercles discrets, nous remarquons que  $\sum_{i=1}^n \alpha_i$  est l'angle au centre de l'arc de cercle qui correspond au polygone approximé. La distinction entre cercles et arcs de cercles se fera en utilisant la somme qui doit être égale à  $2\pi$  ou pas.

En reliant les points d'une courbe discrète, nous obtenons un polygone. Cependant nous ne pouvons pas appliquer directement sur ce polygone les résultats de la section 3 car le changement d'angle d'un point au point suivant est toujours un multiple de  $\frac{\pi}{4}$  et parfois, ce changement ( $\frac{\pi}{2}$  ou  $\frac{3\pi}{4}$ ) est trop grand et les hypothèses du théorème 1 ne sont plus respectées (l'approximation  $\sin x \simeq x$  n'est pas valable). Donc, afin d'éviter cet inconvénient, une étape de pré-traitement est nécessaire avant la transformation vers la représentation dans l'espace des tangentes. D'abord, nous faisons une polygonalisation de la courbe donnée en entrée. Ensuite, en appliquant les résultats de la section 4.3.2, nous allons vérifier si ce polygone est une approximation de l'ensemble des cordes consécutives d'un cercle discret.

Nous proposons dans l'algorithme 14 une méthode linéaire pour la reconnaissance d'un arc ou d'un cercle. Le paramètre  $\alpha$  donné en entrée de l'algorithme permet de gérer l'erreur d'approximation effectuée en considérant que  $\sin \alpha_i \simeq \alpha_i$  pour  $i \in \{1, \dots, n-1\}$  (cf. la section suivante où nous étudions les erreurs d'approximation induites). Le paramètre  $\nu_1$  est utilisé pour la polygonalisation de la courbe en entrée. Le paramètre  $\nu_2$  est utilisé comme l'épaisseur dans l'algorithme de reconnaissance de segment flou [DRFRD06] pour vérifier la colinéarité des points centraux dans la représentation de l'espace des tangentes. Nous remarquons que la représentation d'un arc ou d'un cercle dans l'espace des tangentes se trouve toujours dans une zone dont les ordonnées varient entre 0 et  $2\pi$  (c'est toujours  $2\pi$  pour une courbe fermée). Donc, le paramètre  $\nu_2$  doit être

7.  $\nu_1 = 1$  si la courbe donnée en entrée est non-bruitée.

---

**Algorithme 14** : *RECONNAISSANCE\_ARC* : Reconnaissance des arcs/cercles discrets

---

**Données** :  $C = \{C_i\}_{i=0}^n$  courbe discrète,  $\alpha_{max}$  l'angle maximal admis dans l'espace des tangentes,  $\nu_1$ - l'épaisseur des segments flous pour la polygonalisation,  $\nu_2$ - l'épaisseur de segment flou pour tester la colinéarité.

**Résultat** : ARC si  $C$  est un arc discret, CERCLE si  $C$  est un cercle discret, NON sinon.  
**début**

Utiliser l'algorithme [DRFRD06] pour faire la polygonalisation de  $C$  par des segment flous d'épaisseur  $\nu_1$   $\tau$  :  $P = \{P\}_{i=0}^m$ ;

Représenter  $P$  dans l'espace modifié des tangentes par  $T(P)$  (cf. section 4.3.1);

**si** il existe  $i$  tel que  $T_{i2}.y - T_{i1}.y > \alpha_{max}$  **alors**

└ retourner NON;

Déterminer l'ensemble des points centraux  $I = \{M_i\}_{i=1}^{n-1}$  de  $\{T_{i2}T_{(i+1)1}\}_{i=1}^{n-1}$  ;

Utiliser la méthode [DRFRD06] pour vérifier si  $I$  est un segment flou d'épaisseur  $\nu_2$ ;

**si**  $I$  est un segment flou d'épaisseur  $\nu_2$  **alors**

└ **si**  $T_{m1} \simeq 2 * \pi$  **alors** retourner CERCLE;

└ **sinon** retourner ARC;

**sinon**

└ retourner NON;

**fin**

---

assez petit par rapport à  $2\pi$ . En pratique, nous choisissons  $\nu_2$  entre 0.2 et 0.3.

### Complexité

Les algorithmes de polygonalisation et de reconnaissance de segments flous sont linéaires [DRFRD06]. La complexité pour transformer un polygone vers sa représentation dans l'espace modifié des tangentes est en  $O(m)$  où  $m$  est nombre de sommets du polygone approximé et la construction de l'ensemble des points centraux  $M = \{M_i\}_{i=1}^{m-1}$  est en  $O(m)$ . Puisque  $m \ll n$ , la complexité totale de notre méthode de reconnaissance est donc  $O(n)$  où  $n$  est nombre de points de la courbe donnée en entrée.

#### 4.3.4 Approximation de l'erreur

Supposons que  $\alpha = \max\{\alpha_i\}_{i=1}^n$ .

Deux étapes dans la méthode que nous proposons (cf. algorithme 14) peuvent engendrer des erreurs d'approximation. Nous devons donc déterminer les bornes maximales sur les erreurs effectuées. La première source d'erreur d'approximation vient de l'hypothèse  $\sin \alpha_i \simeq \alpha_i$  pour  $i \in \{1, \dots, n-1\}$  dans le théorème 4.3.3. La deuxième source d'erreur d'approximation provient de l'épaisseur fixée dans l'utilisation de l'algorithme de reconnaissance de segment flou pour extraire un ensemble des points colinéaires dans la représentation dans l'espace des tangentes.

#### Approximation d'un ensemble de cordes par une droite dans l'espace des tangentes

Après la transformation du polygone  $P$  dans l'espace des tangentes, supposons que l'ensemble des points centraux  $\{M_i\}_{i=0}^{n-1}$ , issu de  $T(P)$ , soit un ensemble de points colinéaires. Grâce au théorème 4.3.2, nous avons :  $\frac{l_1+l_2}{\alpha_1} = \frac{l_2+l_3}{\alpha_2} = \dots = \frac{l_{n-1}+l_n}{\alpha_{n-1}} = K$  Posons  $K = 2 * \bar{R}$ , on peut



alors considérer  $\bar{R}$  comme le rayon du cercle approximé. Supposons que  $R_i$  est le rayon du cercle passant par les 3 points  $C_{i-1}, C_i, C_{i+1}$ ;  $\alpha_{i1} = \angle H_{i-1}OC_i$ ,  $\alpha_{i2} = \angle H_iOC_i$  (voir la figure 4.2). Les différences entre  $R_i, i \in \{1, \dots, n-1\}$  et  $\bar{R}$  causent une erreur d'approximation. Nous introduisons alors une borne d'erreur à cette étape.

**Définition 4.3.1** *L'erreur induite lorsque nous utilisons le théorème 4.3.3 pour vérifier si un polygone est un ensemble de cordes d'un cercle est définie comme suit*

$$EB = \max\left\{\left|\frac{R_i - \bar{R}}{\bar{R}}\right|\right\}_{i=1}^n$$

Nous avons :  $l_i + l_{i+1} = 2\bar{R} * \alpha_i = 2\bar{R} * (\alpha_{i1} + \alpha_{i2}) = 2R_i * (\sin \alpha_{i1} + \sin \alpha_{i2})$ . Alors,  $\frac{\bar{R}}{R_i} = \frac{\sin \alpha_{i1} + \sin \alpha_{i2}}{\alpha_i} > \frac{\sin(\alpha_{i1} + \alpha_{i2})}{\alpha_i} = \frac{\sin \alpha_i}{\alpha_i}$ . Grâce à la propriété de décroissance de la fonction  $\frac{\sin x}{x}$ , nous avons  $\frac{\sin \alpha_i}{\alpha_i} \geq \frac{\sin \alpha}{\alpha}$ . Donc,

$$EB < \frac{\alpha}{\sin \alpha} - 1$$

La figure 4.4 présente la majoration de l'erreur quand  $\alpha$  varie entre 0 et  $\frac{\pi}{4}$ . En pratique nous choisissons  $\frac{\pi}{4}$  comme la valeur seuil de  $\alpha$  pour assurer la circularité des arcs et des cercles que nous segmentons avec notre algorithme. Dans cette condition, nous obtenons une majoration de l'erreur qui est inférieure à 0.1 (voir figure 4.4).

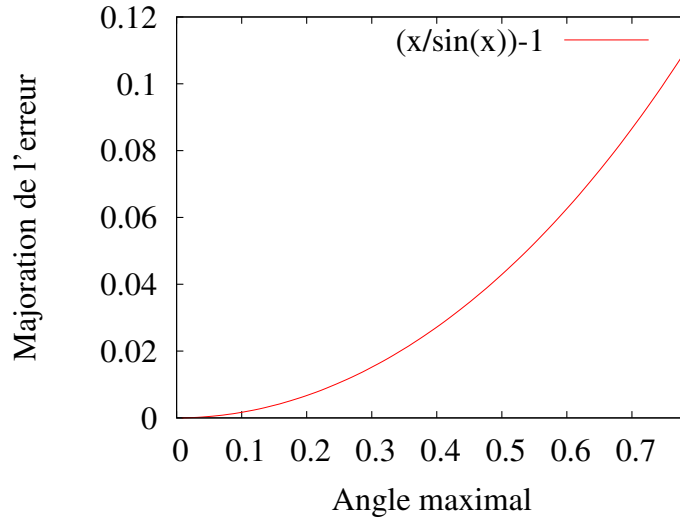


FIGURE 4.4 – Erreur d'approximation quand l'angle maximal varie de 0 à  $\pi/4$

### Test de colinéarité par l'algorithme de reconnaissance de segments flous à une épaisseur $\nu$

Considérons la figure 4.5. L'ensemble des points centraux  $\{M_i\}_{i=0}^{n-1}$  est reconnu comme un segment flou d'épaisseur  $\nu_2$  et sa droite englobante optimale est  $D(a, b, \mu, \omega)$ . Comme l'épaisseur horizontale de ce segment flou  $\epsilon$  est égale à  $\frac{\omega-1}{a}$  et que l'épaisseur verticale est égale à  $\frac{\omega-1}{b}$  et est inférieure ou égale à  $\nu_2$  alors, nous avons la relation  $\epsilon \leq \frac{b}{a}\nu_2$ . Supposons que la droite réelle centrale de ce segment flou contienne l'ensemble des points  $\{M'_i\}_{i=0}^{n-1}$  tel que  $M'_i.y = M_i.y$  (voir

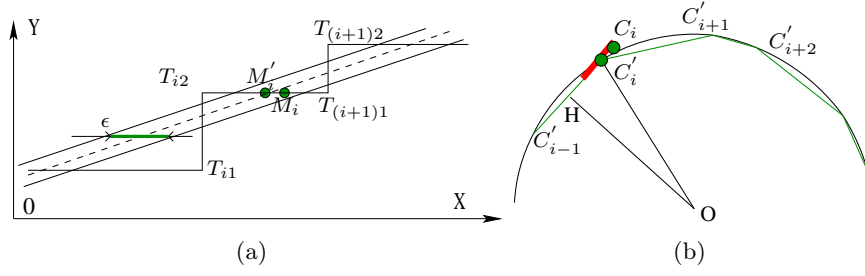


FIGURE 4.5 – Erreur d'approximation lorsqu'on utilise l'algorithme de reconnaissance de segments flous [DRFRD06] pour tester la colinéarité de l'ensemble des points centraux dans l'espace des tangentes.

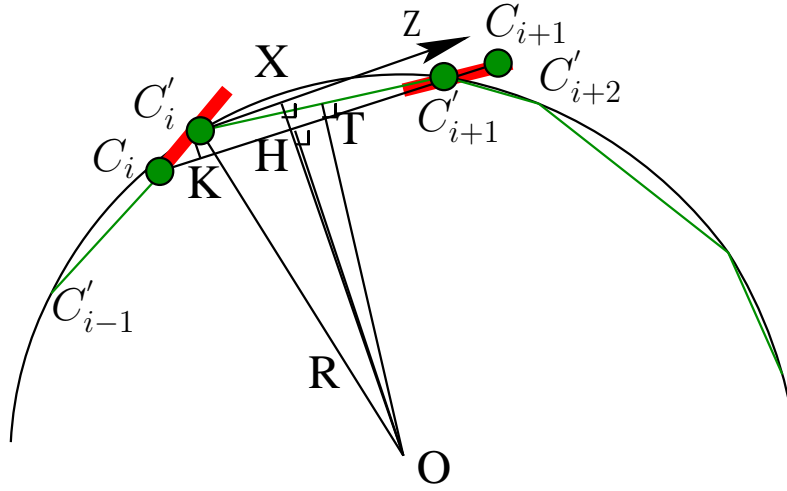


FIGURE 4.6 – Un polygone inscrit dans un cercle réel.

la figure 4.5.a). Cet ensemble correspond à la suite de points  $\{C'_i\}_{i=0}^n$  sur un cercle réel. Grâce à la règle de transformation vers l'espace des tangentes, nous avons  $C'_i C_i = M'_i M_i$  (voir aussi la figure 4.5.a et 4.5.b). Parce que  $M_i, \forall i \in \{0, \dots, n\}$  appartient à ce segment flou dont l'épaisseur horizontale est  $\epsilon$ , donc  $C'_i C_i = M'_i M_i \leq \frac{\epsilon}{2}$ . Donc, nous avons  $OC'_i - C'_i C_i < OC_i < OC'_i + C'_i C_i \Leftrightarrow R - \frac{\epsilon}{2} < OC_i < R + \frac{\epsilon}{2}$ . C'est à dire que l'ensemble des points  $\{C_i\}_{i=0}^n$  appartient à un cercle discret  $\mathbb{C}(O, R, \epsilon)$ .

Nous obtenons donc le résultat suivant.

**Théorème 4.3.4** *Si les points centraux dans l'espace des tangentes forment un segment flou d'épaisseur  $\nu_2$  [DRFRD06], avec  $D(a, b, \nu_2, \omega)$  comme droite discrète englobante alors, l'ensemble des points  $\{C_i\}_{i=0}^n$  se trouve dans un cercle d'Andrès  $\mathbb{C}(O, R, \frac{b}{a}\nu_2)$ .*

Supposons que K, H sont respectivement les projections de O,  $C'_i$  sur  $C_i C_{i+1}$ ; T est la projection de O sur  $C'_i C'_{i+1}$ ; X est la projection de O sur  $C'_i z$  qui est parallèle avec  $C_i C_{i+1}$  (voir la figure 4.6). Nous avons  $|OT| = R \cos \frac{\angle C'_i O C'_{i+1}}{2}$ . Si  $C_i$  n'appartient pas à  $C'_{i-1} C'_i$ , on a  $|OH| > |OT|$ . Dans l'autre cas, nous avons  $|C'_i K| + |OH| = |OX|$ . D'autre part, il est évident que  $|OX| > |OT|$  (voir la figure 4.7). En effet, nous avons  $0 < \angle T C'_i O < \angle X C'_i O < \frac{\pi}{2}$ , donc  $|OT| = R \sin \angle T C'_i O < R \sin \angle X C'_i O = |OX|$

Par conséquent, comme  $|C'_i K| + |OH| = |OX|$  et  $|OX| > |OT|$ , on a  $|OH| > |OT| - |C'_i K|$ .

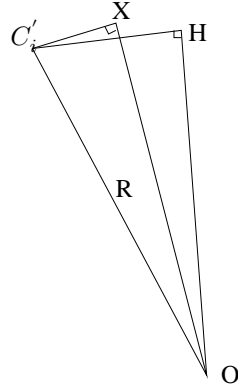
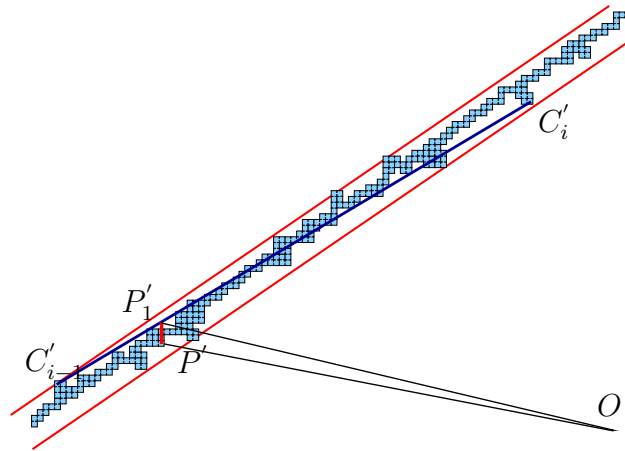


FIGURE 4.7 – Relation entre OX et OH.


 FIGURE 4.8 – Relation entre  $OP'$  et  $OP$ .

Puisque  $|C'_i K| < |C_i C'_i| \leq \frac{\epsilon}{2}$  (si on accepte  $\alpha_{max} \leq \frac{\pi}{4}$ , on peut avoir une approximation  $C'_i K \leq C_i C'_i \sin \alpha_{max} < \frac{1}{\sqrt{2}} C_i C'_i < \frac{1}{2\sqrt{2}} \epsilon$ ). C'est pourquoi, dans tous les cas nous avons  $|OH| > |OT| - |C'_i K| \geq R \cos \frac{\angle C'_i O C'_{i+1}}{2} - \frac{\epsilon}{2}$ . Considérons un point arbitraire  $P$  qui se trouve sur le polygone  $\{C_i\}_{i=0}^m$ , comme  $|OP| \geq |OH|$ , nous avons :

$$\begin{cases} OP < \max\{OC_i\}_{i=0}^m < R + \frac{\epsilon}{2} \\ OP > R \cos \frac{\angle C'_i O C'_{i+1}}{2} - \frac{\epsilon}{2} \end{cases} \quad (4.1)$$

Considérons maintenant un point discret  $P'$  qui appartient à un segment flou d'épaisseur  $\nu_1$  dont les extrémité sont  $C_{i-1}$  et  $C_i$ . Supposons que  $P'_1$  est la projection verticale de  $P'$  sur  $C_{i-1}C_i$ . Alors,  $P'P'_1 \leq \nu_1$  (voir la figure 4.8). Donc,

$$OP - \nu_1 < OP' < OP + \nu_1 \quad (4.2)$$

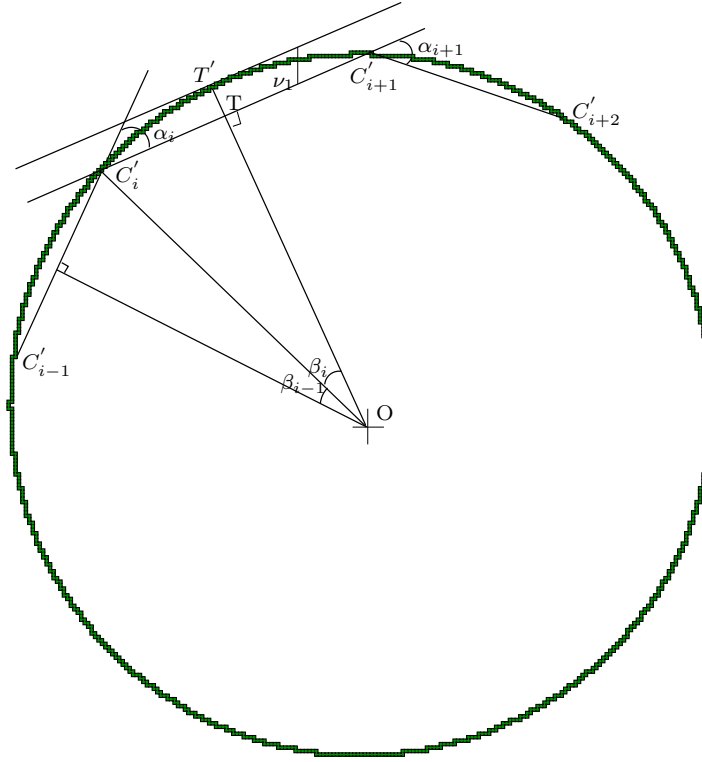


FIGURE 4.9 – Un cercle discret parfait.

Grâce à 4.1 et 4.2, nous avons

$$\begin{cases} OP' < R + \frac{\epsilon}{2} + \nu_1 \\ OP' > R \frac{\cos \angle C'_i OC'_{i+1}}{2} - \frac{\epsilon}{2} - \nu_1 \end{cases} \quad (4.3)$$

Retournons maintenant à un cercle discret parfait de rayon  $R$  qui est polygonalisé par un ensemble de segments flous d'épaisseur  $\nu_1$  avec les extrémités  $\{C'_i\}_{i=0}^m$  (voir aussi la figure 4.9). Nous avons l'estimation suivante :

$$R(1 - \cos \frac{\angle C'_i OC'_{i+1}}{2}) \leq \nu_1 \quad (4.4)$$

C'est pourquoi, grâce aux 4.3 et 4.4, nous avons

$$\begin{cases} OP' < R + \frac{\epsilon}{2} + \nu_1 < R + \frac{\epsilon}{2} + 2\nu_1 \\ OP' > R - \frac{\epsilon}{2} - 2\nu_1 \end{cases} \quad (4.5)$$

Nous obtenons donc le résultat suivant

**Théorème 4.3.5** *Une courbe discrète est approximée par des segments flous d'épaisseur  $\nu_1$  dont les extrémités sont  $\{C'_i\}_{i=0}^n$ . Si les points centraux dans l'espace des tangentes forment un segment flou d'épaisseur  $\nu$  [DRFRD06], avec  $D(a, b, \nu_2, \omega)$  comme droite discrète englobante alors, la courbe donnée en entrée se trouve dans un cercle Andrès  $\mathbb{C}(O, R, \frac{b}{a}\nu_2 + 4\nu_1)$ .*

D'autre part, nous avons  $M'_{i+1}.y - M'_i.y = M_{i+1}.y - M_i.y = \alpha_i, i = \{0, \dots, n-2\}$ . Donc,  $\angle(\overrightarrow{C'_{i-1}C'_i}, \overrightarrow{C'_iC'_{i+1}}) = \angle(\overrightarrow{C_{i-1}C_i}, \overrightarrow{C_iC_{i+1}}) = \alpha_i, i = \{0, \dots, n-2\}$ . Supposons que  $C'_i OC'_{i+1} =$

$2\beta_i, i = \{0, \dots, n-2\}$  (dans la figure 4.9,  $\beta_i = C'_i OT$ ). Alors, il est évident que  $\angle(\overrightarrow{C'_{i-1}C'_i}, \overrightarrow{C'_iC'_{i+1}}) = \beta_{i-1} + \beta_i, i = \{1, \dots, n-2\}$ . Par conséquent, nous avons cette relation :  $\beta_{i-1} + \beta_i = \alpha_i, i = \{1, \dots, n-2\}$ . Donc,  $\alpha_{max} \leq 2 \max\{\beta_i\}_{i=0}^{n-2}$ . Grâce à cet événement et 4.4, nous obtenons

$$R(1 - \cos \frac{\alpha_{max}}{2}) \leq \nu_1 \quad (4.6)$$

Donc, nous obtenons la relation suivante :

$$\alpha_{max} \leq 2 \arccos(1 - \frac{\nu_1}{R}) \quad (4.7)$$

L'inéquation 4.7 nous permet de contrôler la qualité de la détection de cercles reposant sur l'information de rayon du cercle et l'épaisseur des segments flous qui est utilisée dans la phase de polygonalisation. Nous proposons ci-dessous un algorithme amélioré de l'algorithme 14. Nous rappelons que le rayon approximé peut être déduit à partir de la pente du segment flou qui contient l'ensemble des points centraux dans l'espace des tangentes (voir le théorème 4.3.1).

---

**Algorithme 15** : *RECONNAISSANCE\_ARC* : Reconnaissance des arcs/cercles discrets

---

**Données** :  $C = \{C_i\}_{i=0}^n$  courbe discrète,  $\nu_1$ - l'épaisseur des segments flous pour la polygonalisation,  $\nu_2$ - l'épaisseur de segment flou pour tester la colinéarité.

**Résultat** : ARC si  $C$  est un arc discret, CERCLE si  $C$  est un cercle discret, NON sinon.

**début**

Utiliser l'algorithme [DRFRD06] pour faire la polygonalisation de  $C$  par des segment flous d'épaisseur  $\nu_1$ <sup>8</sup> :  $P = \{P\}_{i=0}^m$ ;

Représenter  $P$  dans l'espace modifié des tangentes par  $T(P)$  (cf. section 4.3.1);

Déterminer l'ensemble des points centraux  $I = \{M_i\}_{i=1}^{n-1}$  de  $\{T_{i2}T_{(i+1)1}\}_{i=1}^{n-1}$  ;

Utiliser la méthode [DRFRD06] pour vérifier si  $I$  est un segment flou d'épaisseur  $\nu_2$ ;

**si**  $I$  est un segment flou d'épaisseur  $\nu_2$  **alors**

    R : le rayon du arc, cercle reconnu qui est calculé à partir de la pente de  $I$ ;

$\alpha_{max} = 2 \arccos(1 - \frac{\nu_1}{R})$ ;

**si** il existe  $i$  tel que  $T_{i2}.y - T_{i1}.y > \alpha_{max}$  **alors**

**retourner** NON;

**si**  $T_{m1} \simeq 2 * \pi$  **alors** **retourner** CERCLE;

**sinon** **retourner** ARC;

**sinon**

**retourner** NON;

**fin**

---

### 4.3.5 Segmentation en arcs discrets

#### Algorithme

Nous obtenons une méthode de segmentation en arcs discrets, utilisant le principe de l'algorithme 14 (voir la page 109), en appliquant une segmentation en segments flous d'épaisseur  $\nu$  [DRFRD06] sur la courbe des points centraux dans l'espace des tangentes. Cette méthode est présentée dans l'algorithme 16.

---

8.  $\nu_1 = 1$  si la courbe donnée en entrée est non-bruitée.

9.  $\nu_1 = 1$  si la courbe donnée en entrée est non-bruitée.

---

**Algorithme 16** : Segmentation d'une courbe en arcs discrets reposant sur l'algorithme 14

---

**Données** :  $C = \{C_i\}_{i=0}^n$  courbe discrète,  $\alpha$  l'angle maximal admis dans l'espace des tangentes,  $\nu_1$ - l'épaisseur des segments flous pour la polygonalisation,  $\nu_2$ - l'épaisseur de segment flou pour tester la colinéarité.

**Résultat** : ARC- séquence des arcs

**début**

$ARC = \emptyset$ ;

    Utiliser l'algorithme [DRFRD06] pour faire la polygonalisation de  $C$  en segments flous d'épaisseur  $\nu_1$  <sup>9</sup> :  $P = \{P\}_{i=0}^m$ ;

    Représenter  $P$  dans l'espace modifié des tangentes par  $T(P)$  (cf. section 4.3.1);

    Déterminer l'ensemble des points centraux  $M = \{M_i\}_{i=1}^n$  (cf. section 4.3.1);

    Utiliser l'algorithme [DRFRD06] pour faire la polygonalisation de  $M$  en segments flous d'épaisseur  $\nu_2$  :  $S = \{S\}_{i=0}^k$ ;

**pour**  $i=0$  à  $k-1$  **faire**

$\{M_b, \dots, M_e\}$  : suite de points de  $M$  qui correspond à  $S_i S_{i+1}$ ;

$C'$  : partie de la courbe  $C$  qui correspond à  $S_i S_{i+1}$ ;

$isArc = VRAI$ ;

**pour**  $i=b$  à  $e-1$  **faire**

**si**  $M_{i+1}.y - M_i.y > \alpha$  **alors**

$isArc = FAUX$

**si**  $isArc == VRAI$  **alors**

            Ajouter  $C'$  à  $ARC$

**fin**

---

### Complexité

La polygonalisation en segments flous est effectuée en temps linéaire. La phase de test des arcs est aussi réalisée en temps linéaire car chaque point  $M_i$  de l'ensemble  $M$  est visité une fois en temps constant. La complexité de cette méthode est donc linéaire.

### 4.3.6 Expérimentations

#### Résultats expérimentaux

Nous avons programmé cette méthode linéaire de segmentation en arcs discrets et en cercles discrets. La figure 4.10 présente un exemple de segmentation en arcs. Dans un premier temps, le polygone approximé (voir la figure 4.10.b) est construit à partir de la courbe donnée en entrée à la figure 4.10.a. Puis, nous la transformons pour obtenir sa représentation dans l'espace des tangentes (voir la figure 4.10.d). Puis, la figure 4.10.e présente la courbe des points centraux obtenus dans l'espace des tangentes. La figure 4.10.c montre les arcs extraits en vérifiant la colinéarité des points centraux dans cet espace. Afin d'obtenir une méthode robuste, nous appliquons la méthode de reconnaissance des segments flous [DRFRD06] pour vérifier si l'ensemble des points est colinéaire.

La figure 4.11 montre les arcs que nous détectons en utilisant des paramètres par défaut ( $\nu_1 = 1$ ,  $\nu_2 = 0.25$ ) en comparaison avec la méthode de Coeurjolly. La figure 4.11.d montre le résultat de la segmentation qui est obtenue avec la méthode de Coeurjolly. Notre méthode fournit une qualité de segmentation équivalente à celle obtenue avec la méthode de Coeurjolly.

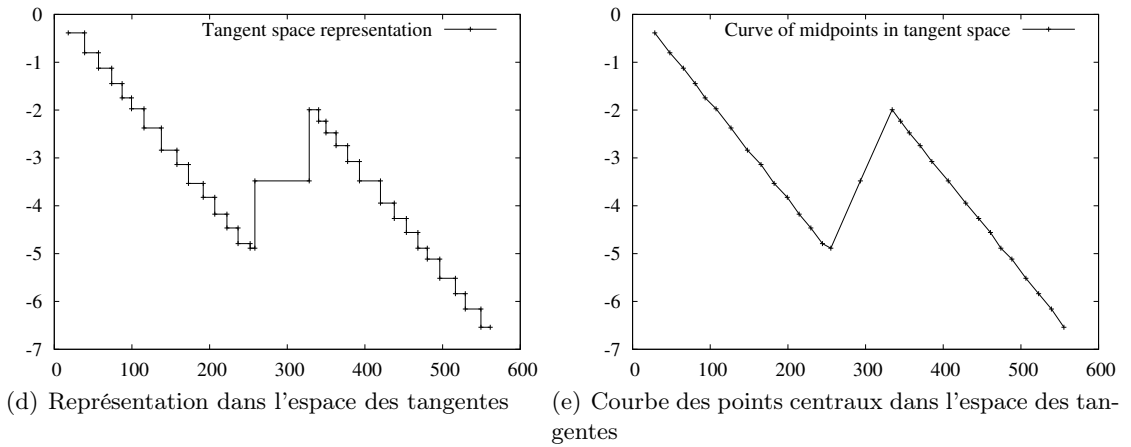
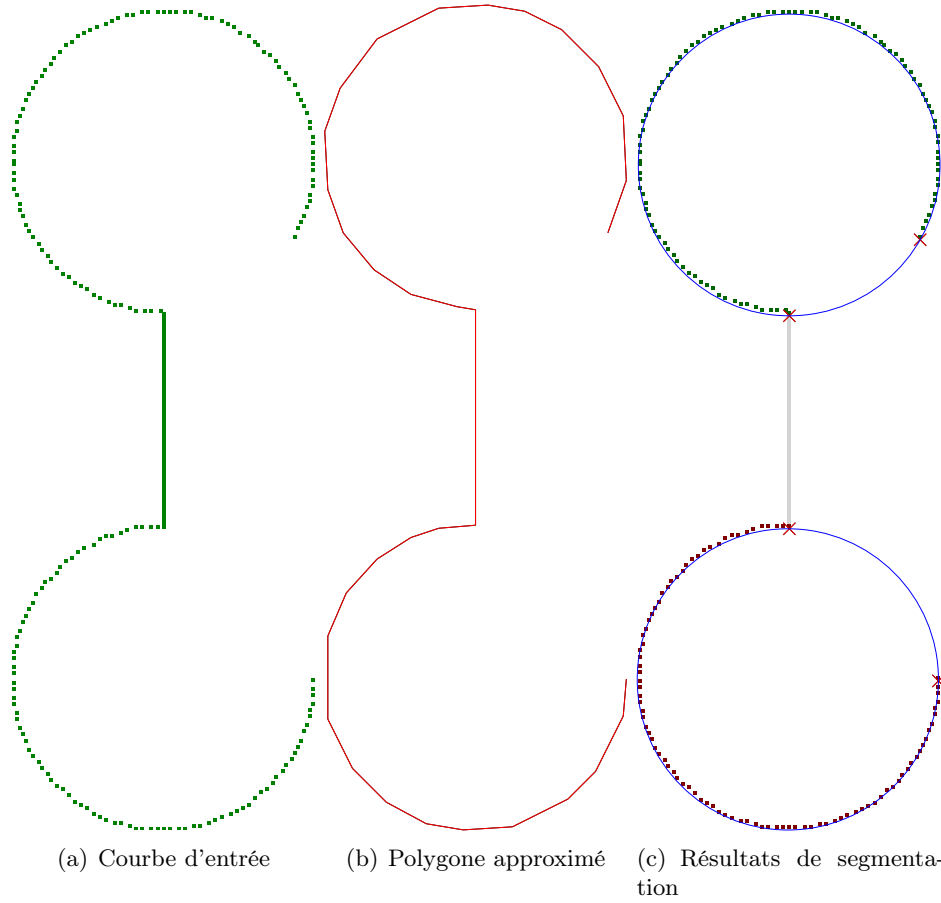


FIGURE 4.10 – Résultats de segmentation en arcs d'une courbe discrète

De plus, la figure 4.12 présente le résultat obtenu par notre méthode sur une partie d'ellipse en comparaison avec le résultat obtenu par celle de Roussillon [RScT09] qui utilise un rayon fixé comme paramètre. Elle montre que notre méthode peut segmenter une courbe discrète en arcs significatifs dont les rayons sont détectés automatiquement au lieu de plusieurs arcs de rayon fixé [RScT09]. Donc, notre méthode est aussi robuste au niveau de la qualité de segmentation.

### Comparaison de la complexité avec d'autres méthodes

Grâce à la complexité linéaire, notre méthode est plus efficace que plusieurs méthodes de reconnaissance d'arcs et de cercles telles que la méthode de Kovalevsky [Kov90] (en temps  $O(n^2 \log n)$ ) ou celle de Coeurjolly [CGRT04] (en temps  $O(n^{4/3} \log n)$ ). Récemment, Roussillon [RScT09] a proposé une méthode linéaire pour la segmentation en cercles dans un contexte de 3 sous-problèmes (voir section 4.2.1). Ces contraintes ne mènent pas à une méthode efficace dans le cas général par rapport à la nôtre. Dans la littérature, il y a seulement 2 méthodes linéaires pour la reconnaissance d'arcs/cercles. Ce sont la méthode de Sauer [Sau93] et celle de Damaschke [Dam95]. Pourtant, ils ont utilisé des outils sophistiqués de programmation linéaire [Meg84] et d'autre part, la méthode de Sauer ne peut pas travailler avec des arcs.

### Application aux documents techniques

Nous avons testé notre méthode sur les contours d'objets qui sont extraits à partir d'images réelles. La figure 4.13 présente quelques image dont les contours sont extraits. Nous appliquons cette méthode sur ces contours pour détecter des arcs. La figure 4.14 présente les arcs extraits qui sont obtenus en utilisant notre méthode. La figure 4.16 présente le résultat sur des images réelles.

La figure 4.15 montre une expérimentation sur des images de document technique qui sont utilisées dans une compétition de segmentation en arcs [GRE09]. Les figures 4.15.a, 4.15.d, 4.15.g sont les images données en entrée. Les figures 4.15.c, 4.15.f, 4.15.i présentent les arcs extraits sur les contours qui sont présentés dans les figures 4.15.b, 4.15.e, 4.15.h. Notre méthode donne des bons résultats sur ce type d'images qui contiennent souvent des primitives arc et cercle.

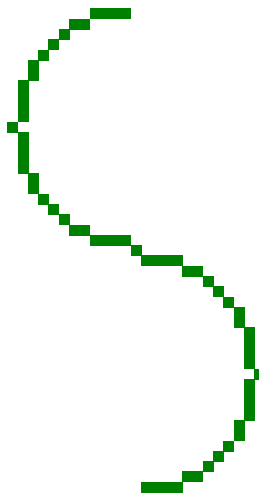
## 4.4 Segmentation d'une courbe bruitée en arcs discrets

Nous présentons dans cette section une extension de la méthode précédente pour travailler avec des courbes bruitées. Ce travail utilise la méthode de détection du bruit proposée par B. Kerautret et J. Lachaud [KL09b] pour construire un polygone approximé qui n'est pas sensible au bruit. Ce polygone est considéré avec la méthode précédente pour extraire les arcs existants. Nous présentons brièvement dans la section 4.4.1 le travail de Kerautret et Lachaud [KL09b] pour la détection du bruit sur un contour discret.

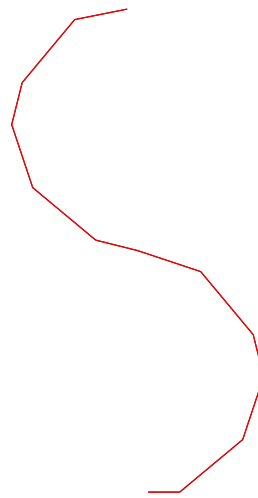
### 4.4.1 Détection du bruit non-supervisée

La technique d'estimation du bruit repose sur une propriété asymptotique de la primitive de *segment maximal*. Nous rappelons qu'une droite discrète standard (DDS) est un ensemble de points 4-connexes  $\{(x, y) \in \mathbb{Z}^2, \mu \leq ax - by < \mu + |a| + |b|\}$ ,  $a, b, \mu \in \mathbb{Z}$ . Un *segment discret de droite* (SDD) est un morceau de DDS. Si nous considérons une courbe discrète 4-connexes  $C$ , un *segment maximal SM* est un sous-ensemble de  $C$  qui est un SSD et n'est plus un SSD

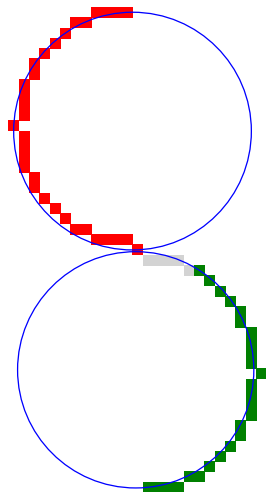




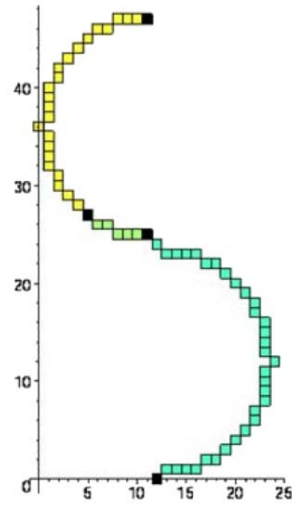
(a) Courbe en entrée



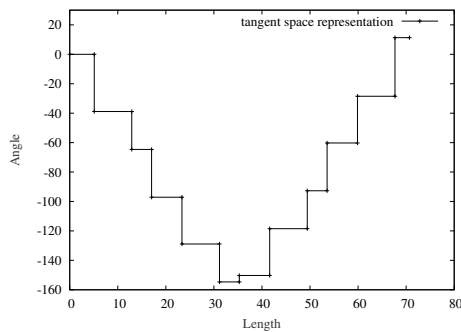
(b) Polygone approximé



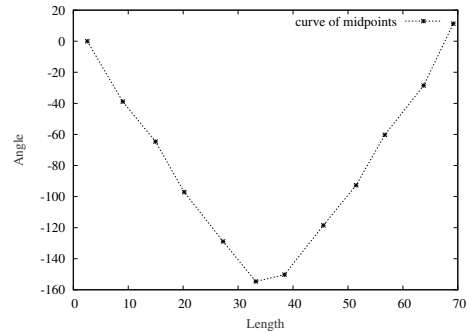
(c) Résultat de la segmentation avec notre méthode



(d) Résultat de la segmentation avec la méthode de Coeurjolly [CGRT04]



(e) Représentation dans l'espace des tangentes



(f) Courbe des points centraux dans l'espace des tangentes

FIGURE 4.11 – Segmentation en arcs discrets d'une courbe issue du travail de Coeurjolly [CGRT04].

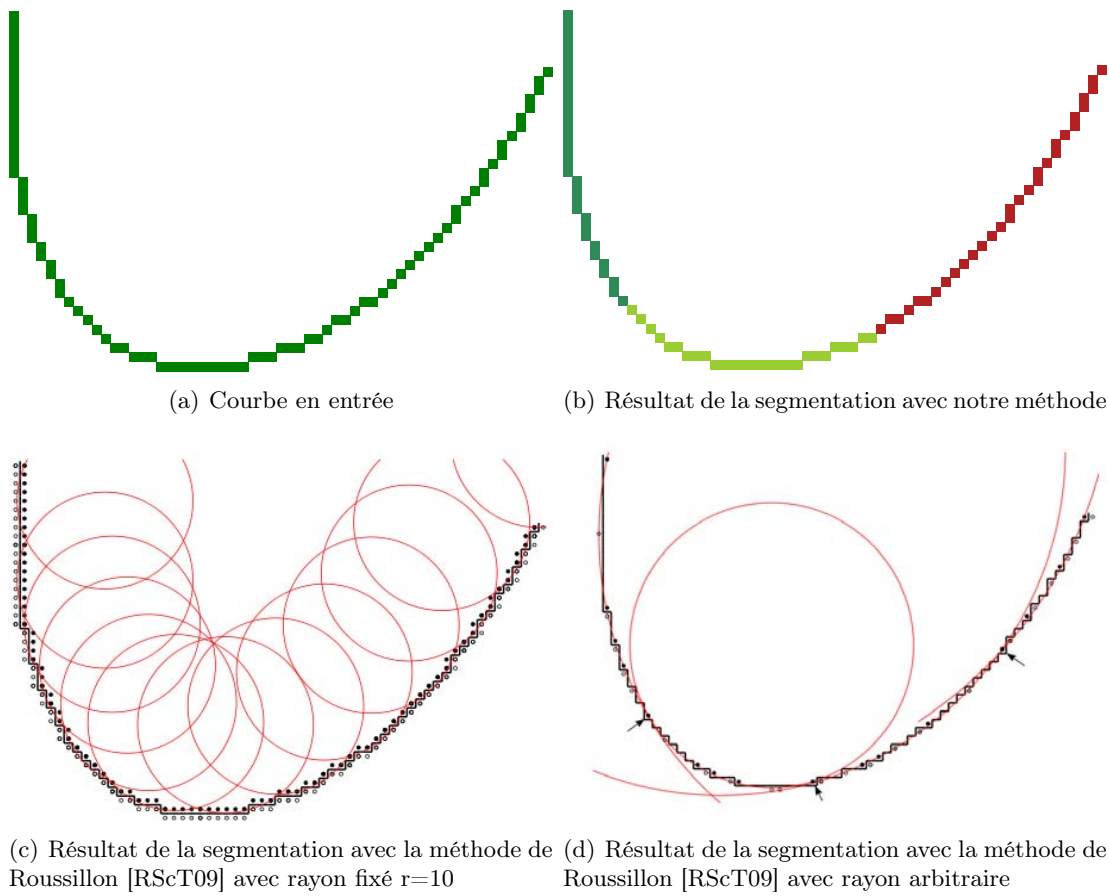


FIGURE 4.12 – Segmentation en arcs discrets sur une partie d’ellipse issue du travail de Roussillon [RScT09].

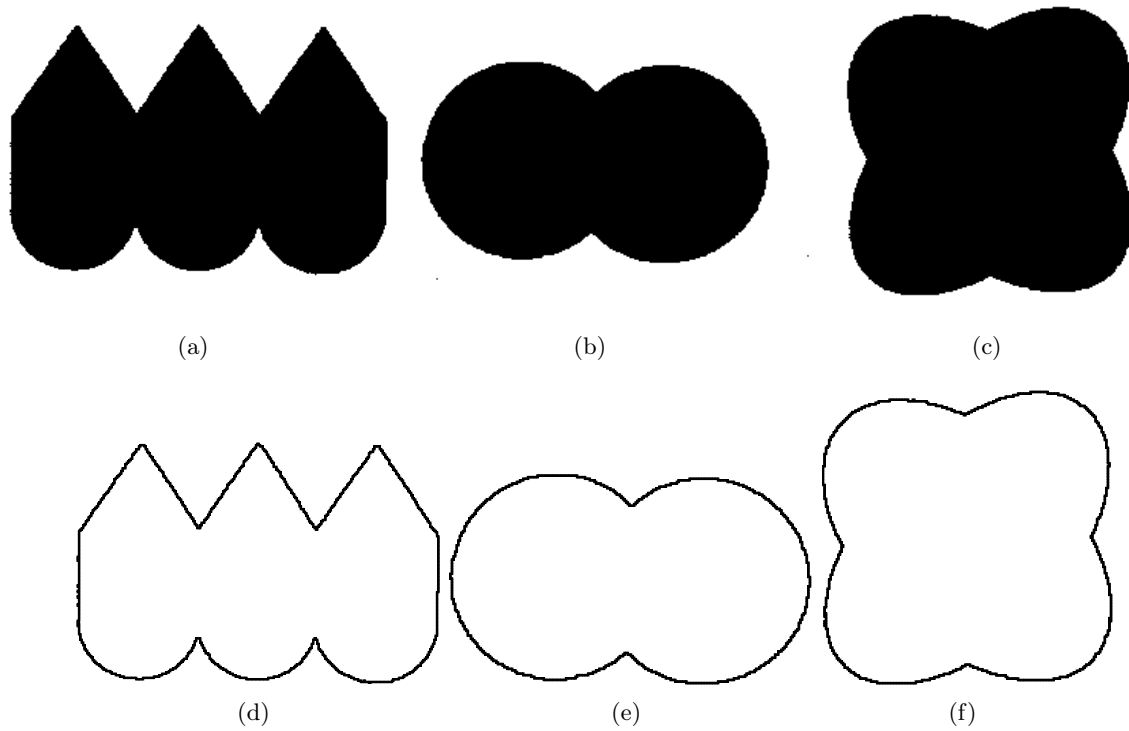


FIGURE 4.13 – a,b,c : Les images d'entrée ; d,e,f : Leur contours extraits

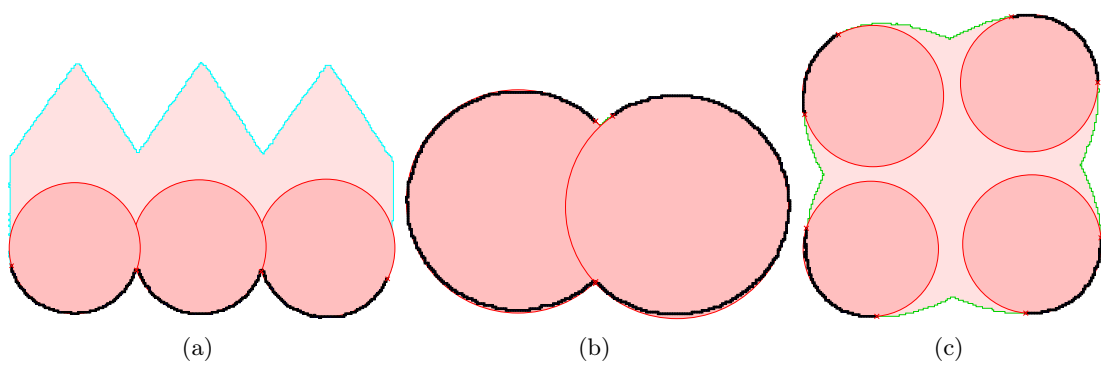


FIGURE 4.14 – Résultats extraits sur les images dans la figure 4.13

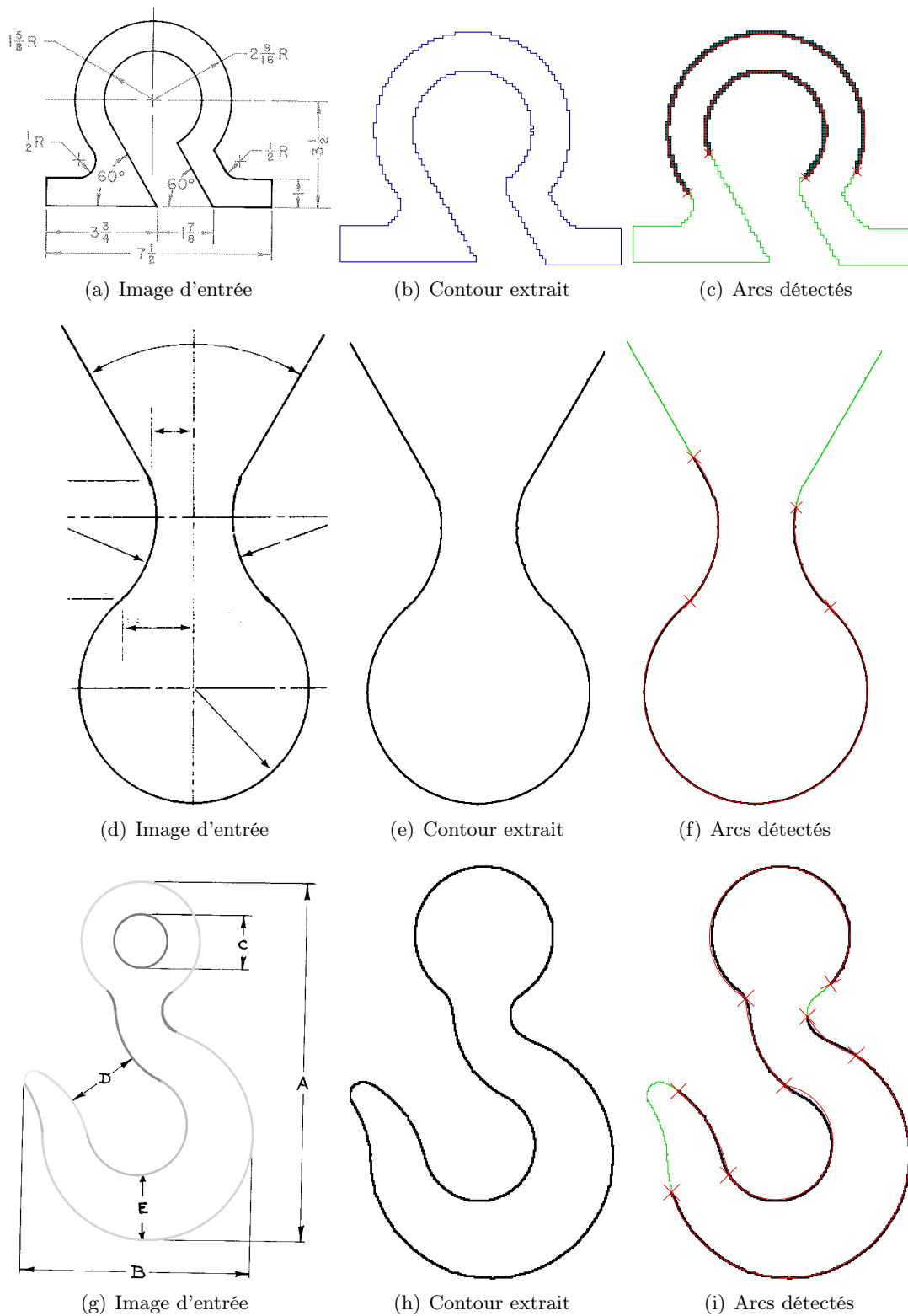
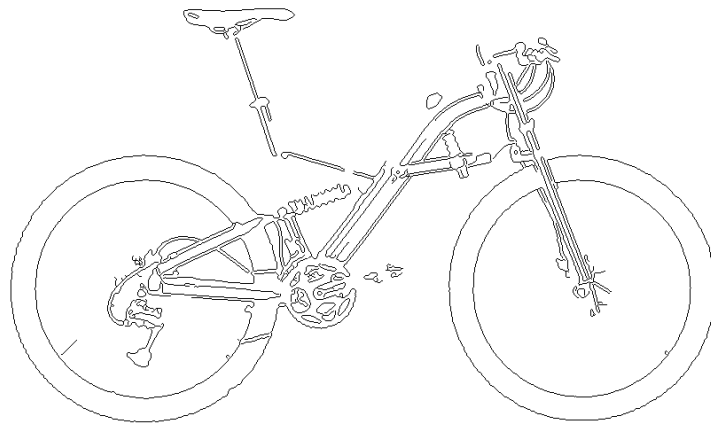


FIGURE 4.15 – Expérimentation sur les images de document technique [GRE09]. Les arcs extraits sont présentés dans des zones épaisses.



(a) Image en entrée



(b) Contour extrait



(c) Détection des arcs

FIGURE 4.16 – Expérimentation sur une image réelle à l'épaisseur 2.

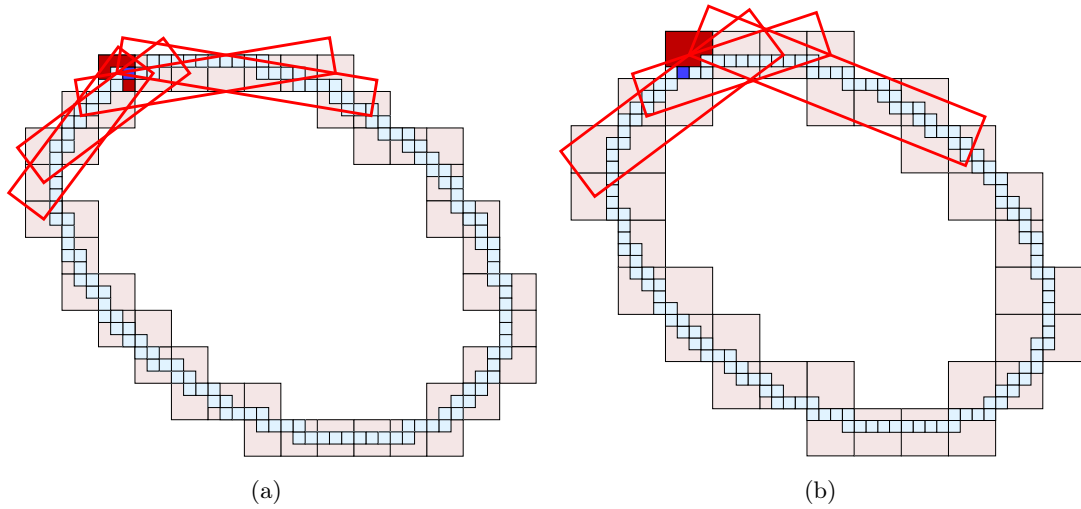


FIGURE 4.17 – Kerautret et Lachaud : Illustration des segments maximaux qui sont obtenus sur les contours de sous-échantillon  $\phi_3(C)$  (a) et  $\phi_3(C)$  (b).

lorsqu'on ajoute un autre point de  $C \setminus SM$ . La figure 4.17.a présente une suite de quatre segments maximaux de la courbe représentée par des pixels gris.

### Propriétés asymptotiques des segments maximaux

La présence de bruit est liée à la longueur discrète des segments maximaux. Soit  $\text{Dig}_h(X)$  la discrétisation Gaussienne à la taille  $h$  de la grille de la forme  $X$  ; et soit  $P$  un point du contour de  $X$ . Nous appelons  $(L_j^h)$  les longueurs discrètes des segments maximaux qui sont définies sur  $\text{Dig}_h(X)$  et qui couvrent  $p$ . Dans [LVdV07] (lemme 15) et [Lac06] (théorème 5.26), les auteurs ont montré que si  $X$  est une courbe connectée dans  $\mathbb{R}^2$  avec un morceau  $C^3$  de contour  $\partial X$  et  $U$  - un voisin connecté ouvert de  $P$ , alors nous avons

$$\text{si } U \text{ est strictement convexe ou concave, alors } \Omega(1/h^{1/3}) \leq L_j^h \leq O(1/h^{1/2}) \quad (4.8)$$

$$\text{si } U \text{ a une valeur de courbure nulle par tout, alors } \Omega(1/h) \leq L_j^h \leq O(1/h) \quad (4.9)$$

En pratique, il est impossible d'obtenir une discrétisation asymptotique de la forme initiale 0 avec des tailles de grille de plus en plus fines  $h$ , les auteurs considèrent un sous-échantillonnage  $\phi_i(O)$  avec une taille de pixels croissante  $i \times i$  pour  $i = 2, \dots, n$ . Puis, ils considèrent les longueurs discrètes  $(L_j^{h_i})$  des segments maximaux des courbes sous-échantillonnées suivant une séquence de taille de grille  $h_i = ih$ . Pour une taille donnée de sous-échantillon  $i$ , la longueur discrète moyenne de tous les segments maximaux qui contiennent des pixels sous-échantillonnés est notée  $\bar{L}^{h_i}$ .

### Profil multi-échelle pour la détection du bruit

Le profil multi-échelle  $\mathcal{P}_n(P)$  d'un point  $P$  est défini comme une séquence des échantillons  $(X_i, Y_i) = (\log(i), \log(\bar{L}^{h_i}))_{i=1..n}$ .

Quelques profils multi-échelles sont illustrés sur le contour réel de la figure 4.18. Selon l'inégalité (4.8) (resp. (4.9)), si  $P$  est localisé sur une zone courbée (resp. plate) ; la pente de l'approximation affine du profil multi-échelle doit être dans l'intervalle  $[-\frac{1}{2}, -\frac{1}{3}]$  (resp.  $[-1, -\frac{1}{2}]$ ). Pour les zones d'un contour bruité, les propriétés précédentes ne sont pas valable mais elles sont utilisées utilisées

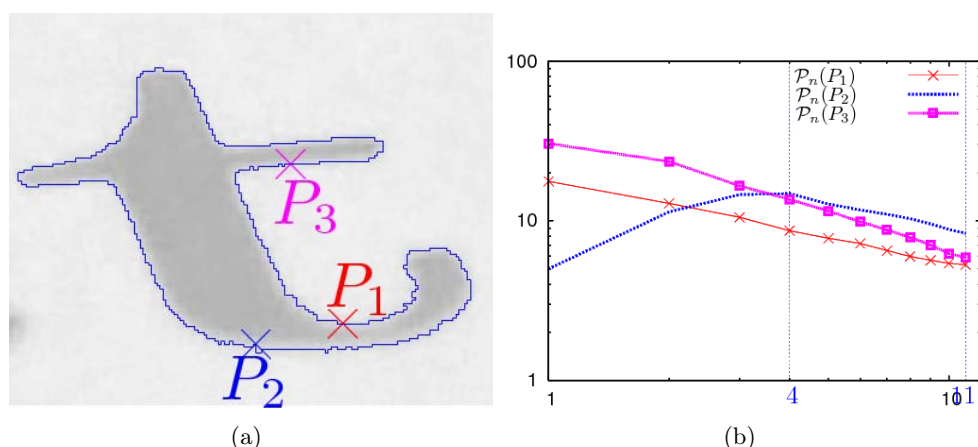
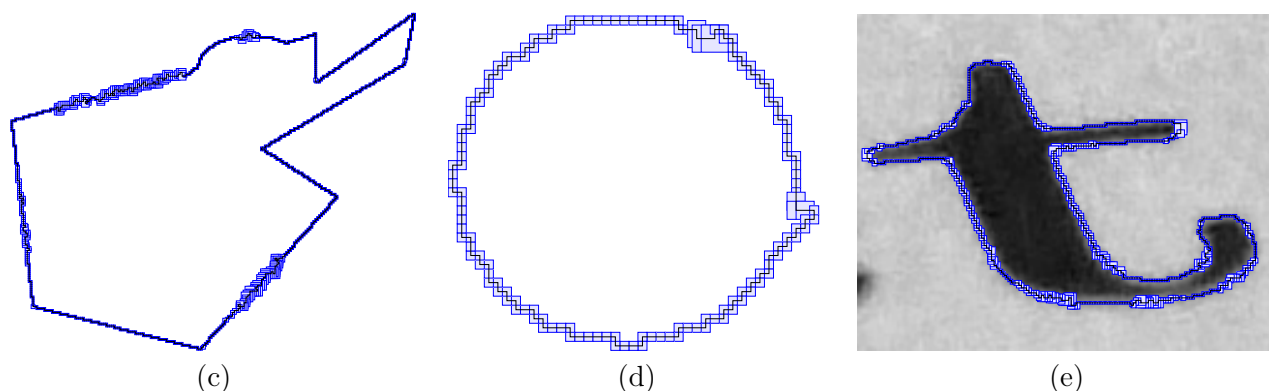


FIGURE 4.18 – Profil multi-échelle.


 FIGURE 4.19 – Kerautret et Lachaud : Détection de bruit sur plusieurs courbes (c-e). Pour chaque point  $P$ , le niveau de bruit  $\eta(P)$  est représenté par une boîte de taille  $\eta(P) + 1$  dont le centre est  $P$ .

pour détecter une telle partie. Un seuil  $t_m$  est donné pour déterminer l'échelle *significative* que est définie comme une paire  $(i_1, i_2)$ ,  $1 \leq i_1 < i_2 \leq n$ , telle que pour tous  $i$ ,  $i_1 \leq i < i_2$ ,  $\frac{Y_{i+1} - Y_i}{X_{i+1} - X_i} \leq t_m$ . Considérons l'exemple de la figure 4.18, les échelles significatives des points  $P_1$  et  $P_2$  sont respectivement égales à  $(1, 11)$  et  $(4, 11)$ .

A partir de l'échelle significative précédente  $(i_1, i_2)$ ,  $i_1 - 1$  définit le niveau de bruit du point  $P$  (noté par  $\eta(P)$ ). De façon expérimentale, la valeur de seuil  $t_m = 0$  donne les meilleurs résultats sur les courbes courbées ou bien plates. La figure 4.19 présente quelques résultats qui sont obtenus sur plusieurs courbes. Le niveau de bruit est bien détecté sur les courbes de résolution basse ou bien sur des contours extraits à partir des images réelles.

#### 4.4.2 Détection des arcs dans une courbe bruitée

Nous avons abordé dans la section 4.3, l'algorithme de détection des arcs qui utilise un paramètre  $\nu_1$  pour tenir compte du bruit de la courbe donnée en entrée dans la phase de polygonalisation. Ce paramètre est modifié manuellement afin d'obtenir le polygone approximé qui permet d'effectuer une bonne reconnaissance des arcs. Dans la section 4.3, nous avons introduit aussi une question intéressante : "Pour chaque courbe bruitée, comment pouvons nous choisir la valeur de  $\nu_1$  pour obtenir le meilleur résultat?". Dans cette section, nous proposons un cadre

de travail afin d'obtenir une méthode sans paramètre de segmentation en arcs. Cette méthode utilise le résultat de détection de bruit [KL09b] qui a été présenté dans la section précédente.

### Polygonalisation des courbes bruitées

Nous proposons deux solutions pour la segmentation en arcs des courbes bruitées. Les deux solutions construisent un polygone approximé en considérant le bruit du contour discret. Nous considérons deux hypothèses différentes par rapport à la distribution du bruit. La première hypothèse suppose que la distribution de bruit est uniforme et la deuxième est non-uniforme.

---

**Algorithme 17** : Polygonalisation basée sur la détection non-supervisée du bruit.

---

**Données** :  $C = \{C_i\}_{i=0}^n$  courbe discrète,  $\eta = \{\eta_i\}_{i=0}^n$  Les informations du bruit, *uniformNoise*- *true* si la distribution du bruit est uniforme, *false* dans l'autre cas

**Résultat** : P-Polygone approximé

**début**

$b \leftarrow 0$ ; Ajouter  $C_b$  to  $P$ ;

**si** *!uniformNoise* **alors**

**pour**  $b < n$  **faire**

            Utiliser [DRFRD06] pour reconnaître  $\{C_b, \dots, C_e\}$  comme un segment flou d'épaisseur  $\eta_b$ ;

$b \leftarrow e$ ; Ajouter  $C_b$  to  $P$ ;

**sinon**

$\bar{\eta} \leftarrow$  valeur moyenne de  $\eta = \{\eta_i\}_{i=0}^n$ ;

**pour**  $b < n$  **faire**

            Utiliser [DRFRD06] pour reconnaître  $\{C_b, \dots, C_e\}$  comme un segment flou d'épaisseur  $\bar{\eta}$ ;

$b \leftarrow e$ ; Ajouter  $C_b$  à  $P$ ;

**fin**

---

Ces solutions reposent sur la détection non-supervisée du bruit qui est présentée dans la section 4.4.1. Cette méthode donne l'information sur le bruit de la courbe donnée en entrée. Cette estimation locale du bruit est utilisée pour construire un polygone approximé qui s'adapte au bruit de la courbe. Les deux processus de polygonalisation sont présentés dans l'algorithme 17. Nous considérons deux options possibles de la distribution de bruit :

**Distribution uniforme du bruit.** Dans ce cas, le polygone approximé est construit en utilisant des segments flous dont la valeur d'épaisseur est constante. Cette valeur est définie comme la moyenne des niveaux de bruit  $\bar{\eta}$  qui sont obtenus sur la courbe donnée en entrée.

**Distribution non-uniforme du bruit.** Dans ce cas, la construction du polygone approximé par les segments flous est obtenue en modifiant l'épaisseur des segments flous à partir de chaque reconnaissance. L'épaisseur est choisie en fonction niveau du bruit  $\eta$  du premier point qui est ajouté au segment flou.

**Indépendance du point de départ.** Les informations concernant les zones plates/courbées nous permet de bien construire le polygone approximé en choisissant le bon point de départ. Nous déterminons le plus grand intervalle  $(f_b, f_e)$   $b, e < n$  du contour qui est associé à une zone plate. Puis, si le contour est fermé, nous appliquons l'algorithme 17 sur le nouveau point de départ qui est déplacé de  $e$  point à partir du point de départ initial. Si le contour n'est pas fermé,



---

**Algorithme 18** : Segmentation d'une courbe discrète bruitée en arcs.

---

**Données** :  $C = \{C_i\}_{i=0}^n$  courbe discrète bruitée**Résultat** : ARC- suite des arcs détectés**début**
 $N \leftarrow \{N_i\}_{i=0}^n$  l'information de bruit qui est déterminée par [KL09a] (voir Section 4.4.1);

 $ARC \leftarrow \emptyset$ ; Utiliser l'algorithme 17 pour obtenir le polygone approximé  $P = \{P\}_{i=0}^m$  de  $C$ ;
Représenter  $P$  dans l'espace des tangentes par  $T(P)$  (voir la section Section 4.3);Déterminer la suite des points centraux  $MpC = \{M_i\}_{i=1}^n$  (voir la section Section 4.3);Utiliser [DRFRD06] pour polygonaliser  $MpC$  comme une suite des segments flous d'épaisseur 0.25 :  $S = \{S\}_{i=0}^k$ ;**pour**  $i$  **de** 0 **à**  $k - 1$  **faire**
 $\{M_j\}_{j=b}^e$  : séquence des points de  $MpC$  qui correspond à  $S_i S_{i+1}$ ;
 $C'$  : une partie de  $C$  qui correspond à  $S_i S_{i+1}$ ; $isArc \leftarrow vrai$ ;**pour**  $i$  **de**  $b$  **à**  $e - 1$  **faire**
**si**  $M_{i+1}.y - M_i.y > \frac{\pi}{4}$  **alors**
 $\quad isArc \leftarrow false$ 
**si**  $isArc$  **alors**
 $\quad$  Ajouter  $C'$  à  $ARC$ 
**fin**


---

 l'algorithme 17 est appelé 2 fois avec les contours partiels  $C'_{e,n}$  et  $C'_{0,b}$ .

### Segmentation d'une courbe bruitée en arcs

L'algorithme 18 présente une approche non-paramétrique pour la segmentation d'une courbe bruitée en arcs. Comme nous avons pu montrer dans la section 4.3, la borne d'erreur d'approximation de la reconnaissance des arcs dépend de l'épaisseur  $\nu_2$  de la reconnaissance des segments flous et de l'angle maximal  $\alpha_{max}$  dans l'espace des tangentes. Pour assurer la qualité de segmentation, nous fixons ces paramètres comme suit :  $\nu_2 = 0.25$  et  $\alpha_{max} = \frac{\pi}{4}$  dans l'algorithme 16 (voir la section 4.3 pour plus information détaillée et pour la configuration de  $\alpha_{max}$ ). Donc, la méthode proposée est pratiquement non-paramétrique lorsque nous travaillons avec des courbes bruitées grâce à la détection non-supervisée de bruit. Elle n'utilise aucun paramètre et détecte automatiquement des arcs existants dans une courbes bruitée.

#### 4.4.3 Expérimentations

La première expérimentation est effectuée sur une courbe synthétique bruitée (voir la figure 4.20). Le bruit Gaussien est ajouté globalement dans les images (a,b) et de manière spécifique dans chaque quadrant (c,d). L'utilisateur choisit seulement la distribution uniforme ou non-uniforme de bruit pour obtenir la détection des arcs plus précis (paramètre booléen *uniformNoise*). L'autre expérimentation est effectuée sur une image réelle de 10 millions pixels dans la figure 4.21. Toutes ces expérimentations ont été effectuées sur un ordinateur *Mac OS X, 2.8GHz, 4GB DDR3 Intel Core Duo*. Les contours sont extraits à partir du filtre de Canny [Can86]. La

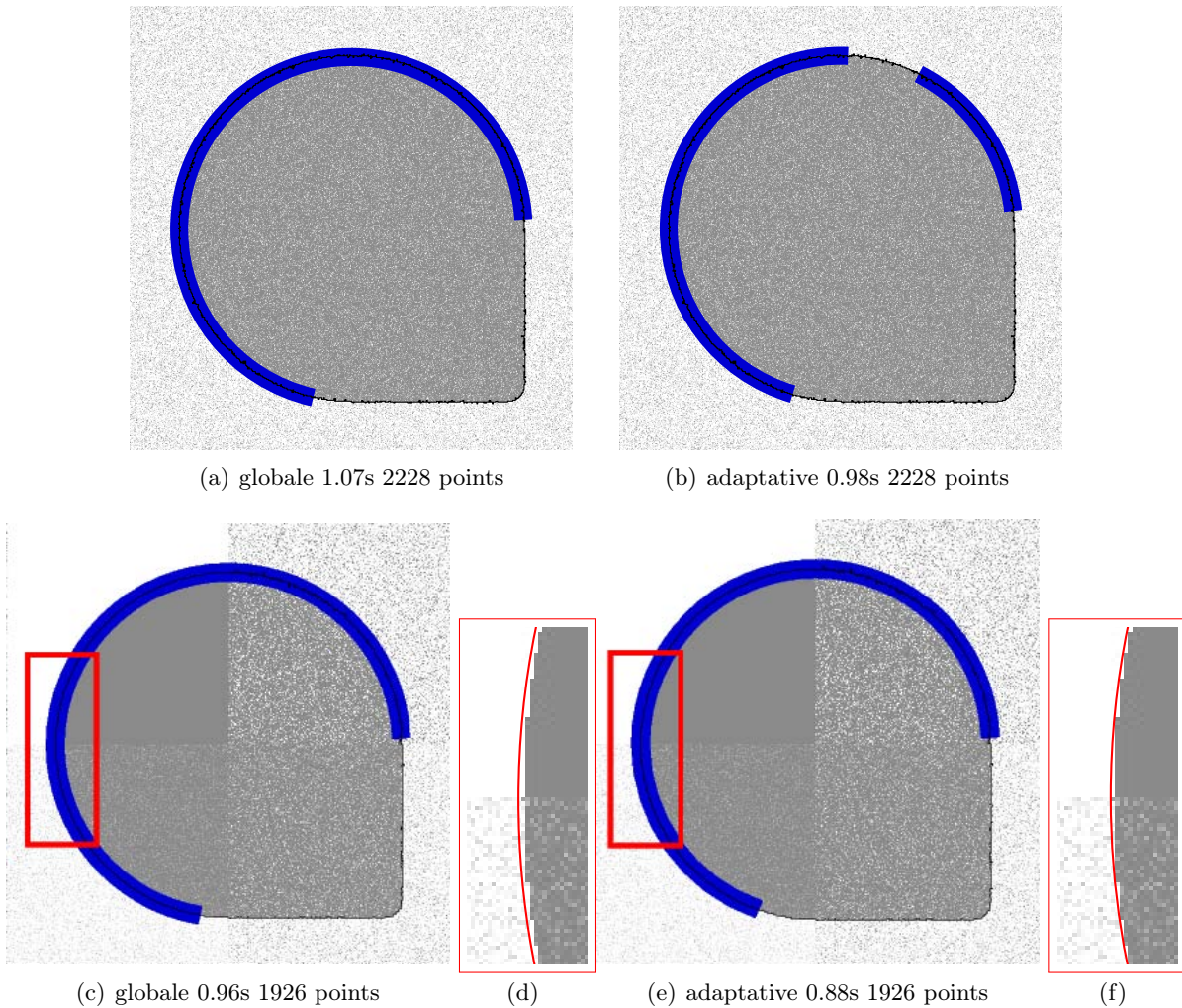
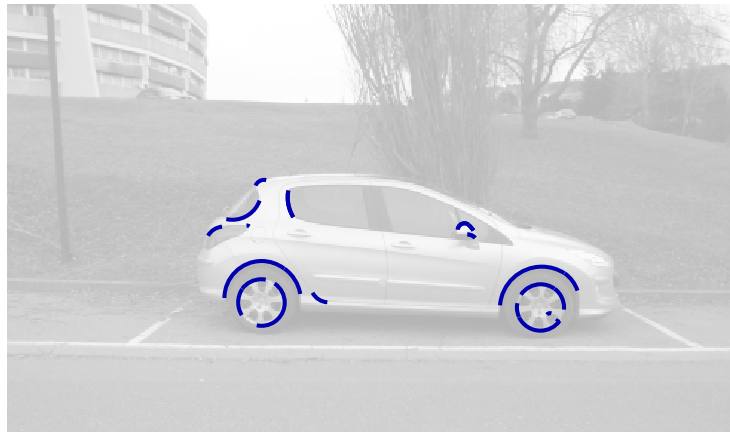


FIGURE 4.20 – Détection des arcs reposant sur une distribution uniforme du bruit (a,d) et non-uniforme du bruit (b,e) (taille d’image 512x512). (d,f) une partie de (c,e).

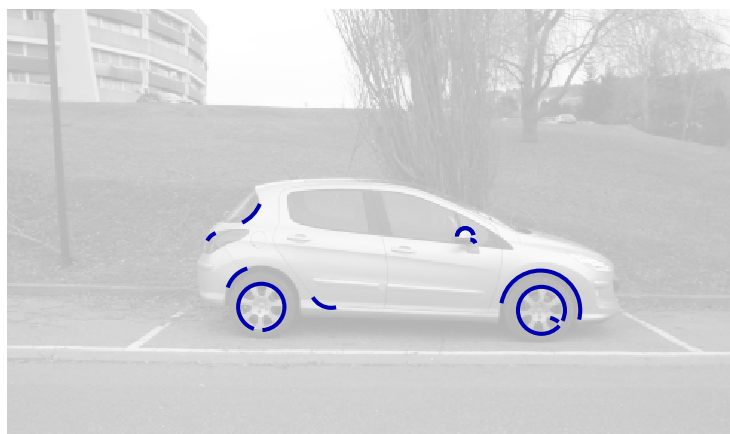
figure 4.22 présente d’autres expérimentations sur une image réelle reposant sur les transformées de Hough : CHT [KBS75], MHT [Dav84] and FHT [RFQ03]. Pour toutes ces expérimentations, l’image source est seuillée par une valeur définie dans l’espace de couleur HSB. De nombreux paramètres  $\mu_C$ ,  $\mu_M$ , et  $\mu_F$  sont associés respectivement à  $(radiusMin, radiusStep, nbCircles)$ ,  $(minRadius, maxRadius)$ ,  $(gradTreshold, radius, radiusVariation)$ . Nous notons que les autres paramètres ne sont pas abordés ici, par exemple pour FHT, et peuvent influencer les résultats ( $\alpha$  et  $\beta$  voir [RFQ03]).



(a) Image source



(b) Méthode globale



(c) Méthode adaptative

FIGURE 4.21 – Détection des arcs en utilisant notre méthode (taille 4000x2672 pixels).

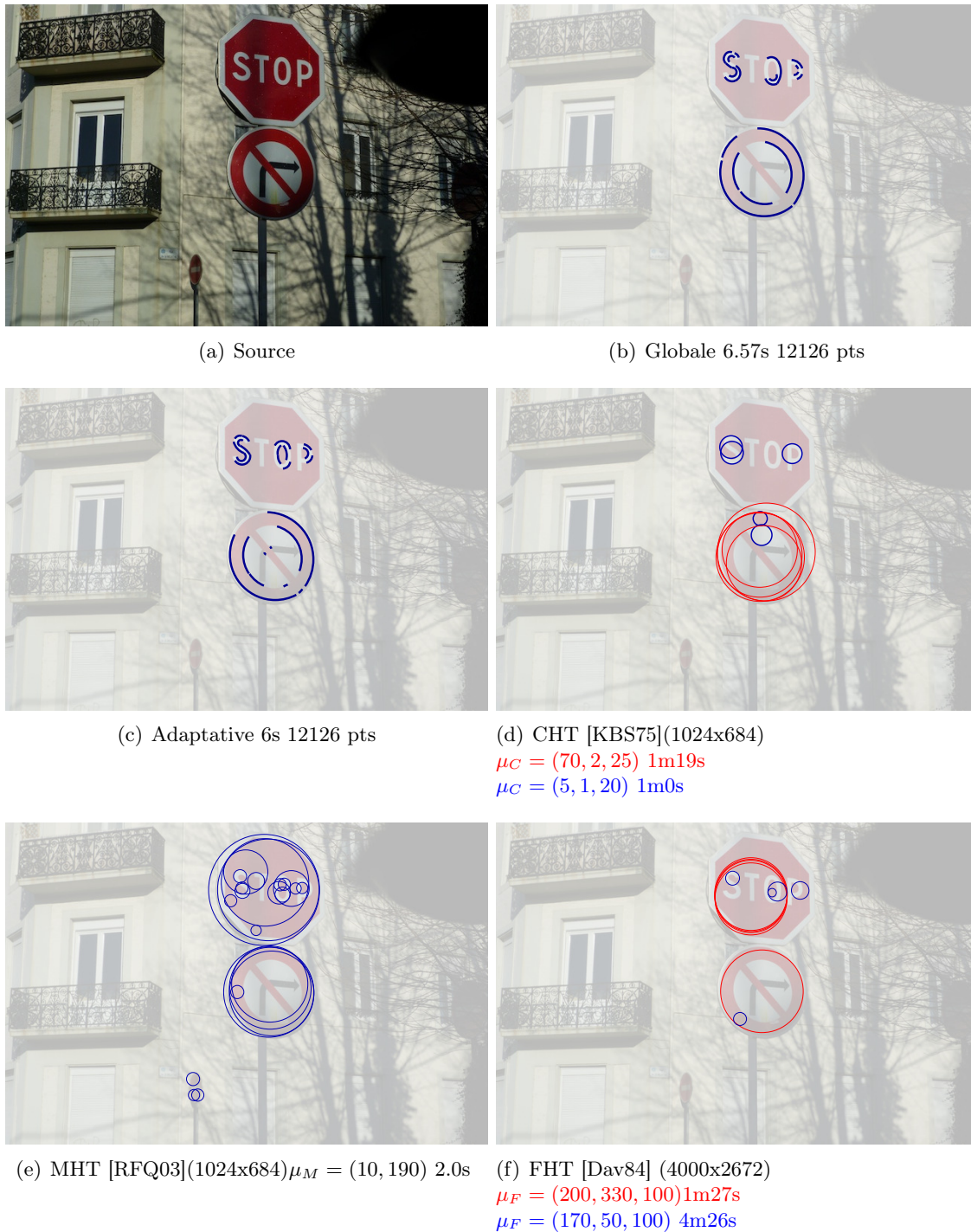


FIGURE 4.22 – Application de notre méthode sur une image réelle (4000x2672 pixels) avec les valeurs possibles de *uniformNoise* (a-c), et comparaison avec les trois méthodes reposant sur la transformée de Hough (d-f).

## 4.5 Décomposition d'une courbe en arcs et segments de droite

Dans cette section, nous présentons une nouvelle méthode pour décomposer une courbe en arcs et en segments de droite. Elle est basée sur une nouvelle méthode de reconnaissance d'arcs et cercles discrets (voir la section 4.3) et sur la détection de points dominants (voir la section 3.4). La détection des points dominants est utilisée comme une étape de pré-traitement pour extraire des points d'intérêt de la courbe. Notre méthode a une complexité de  $O(n \log n)$ . Elle possède donc un coût bas de traitement. De plus, elle est simple et facile à implémenter.

En ce qui concerne la reconnaissance d'un cercle discret, l'algorithme 14 (voir la page 111) permet de reconnaître un cercle discret ou un arc discret en détectant des segments de droite dans l'espace des tangentes. Le paramètre  $\alpha_{max}$  est utilisé afin d'assurer que l'hypothèse du théorème 4.3.1 est valable. Nous proposons la définition ci-dessous.

**Définition 4.5.1** *Dans la courbe des points centraux de l'espace des tangentes, un point isolé est un point central qui satisfait que les différences entre la valeur d'ordonnée de ce point et celles de ces points voisins sont plus grandes que le seuil  $\alpha_{max}$ .*

### 4.5.1 Idée principale

Notre idée principale est d'appliquer un détecteur de points dominants (voir la section 3.4) comme une étape de pré-traitement pour améliorer la qualité de la décomposition. Nous avons constaté que les extrémités entre un arc et un segment de droite ou entre deux arcs, ou bien entre deux segments de droite sont toujours des points dominants. Nous détectons alors des points dominants qui sont candidats pour être des extrémités parmi des primitives dans la décomposition d'une courbe. Ensuite, nous allons grouper les points dominants s'ils constituent un arc. La reconnaissance d'un arc est basée sur le théorème 4.3.1 en utilisant l'algorithme 14. Grâce à cet algorithme, un arc correspond à un segment de droite sur la courbe des points centraux. Ce segment ne contient aucun point isolé. Le processus de décomposition est effectué reposant sur le profil de la courbe des points centraux dans l'espace des tangentes. Ceci utilise une nouvelle notion de point isolé qui est définie au début de cette section.

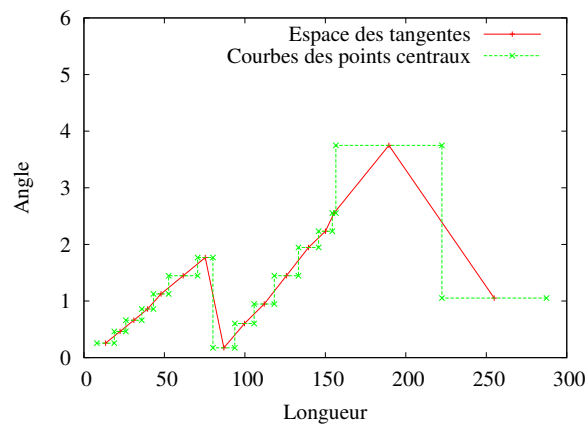
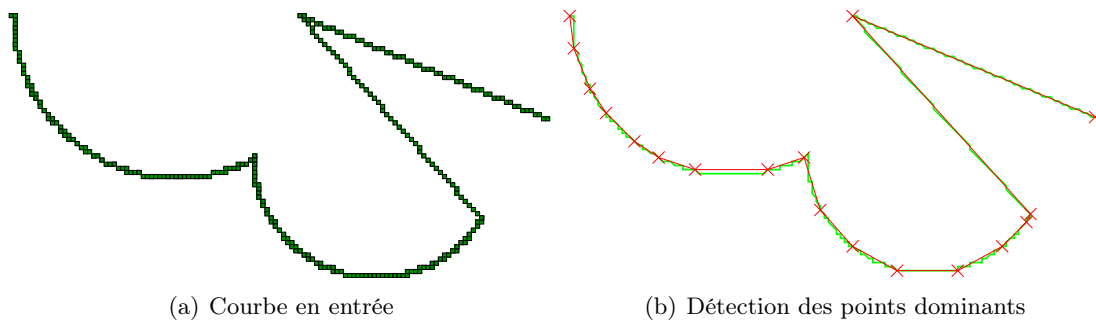
### 4.5.2 Analyse des configurations

Considérons la figure 4.23. Dans cet exemple, il y a toutes les configurations possibles parmi les primitives arc et segment de droite : arc-arc, arc-segment, segment-segment. La figure 4.24 présente la correspondance de ces configurations dans l'espace des tangentes.

En ce qui concerne la courbe des points centraux dans l'espace des tangentes, nous avons les remarques ci-dessous.

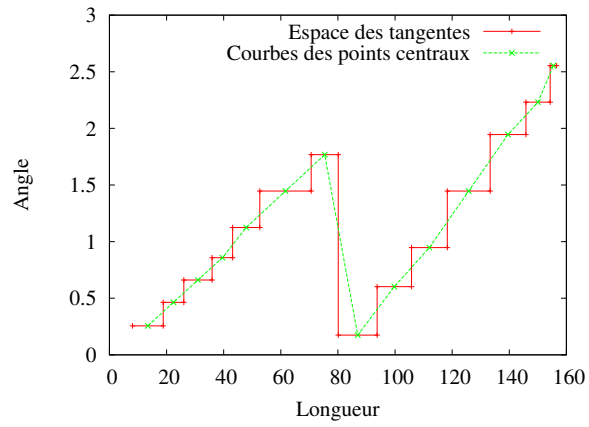
- Un segment de droite de la courbe correspond à un point isolé dans l'espace des tangentes.
- Un point isolé dans l'espace des tangentes peut être colinéaire avec un ensemble de points colinéaires qui correspondent à un arc de la courbe.
- Deux arcs consécutifs de la courbe correspondent à 2 segments de droite dans l'espace des tangentes qui ne contiennent aucun point isolé.

Grâce à la deuxième remarque, nous déduisons qu'une polygonalisation de la courbe des points centraux ne donne pas une meilleure qualité d'extraction des arcs. Donc, nous proposons un traitement reposant sur la détection des points isolés.

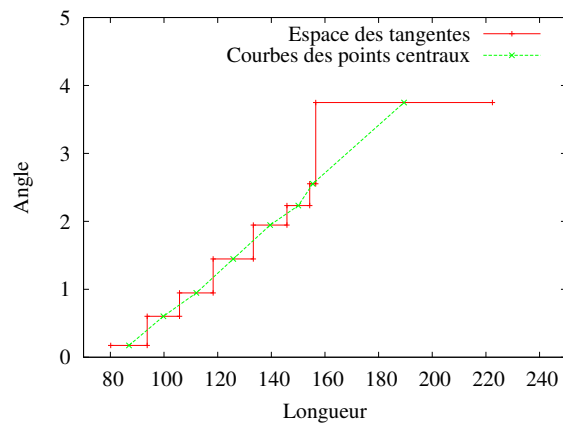


(c) Représentation des points dominants dans l'espace des tangentes et sa courbe des points centraux

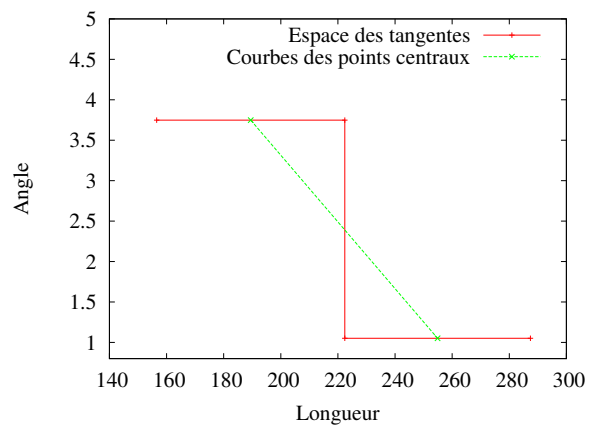
FIGURE 4.23 – Un exemple qui contient toutes les configurations de primitives.



(a) arc-arc



(b) arc-segment



(c) segment-segment

FIGURE 4.24 – Configurations.

### 4.5.3 Algorithme proposé

#### Algorithme

Grâce aux remarques ci-dessus, nous présentons dans cette section un algorithme pour la décomposition d'une courbe en arcs et en segments de droite. L'algorithme incrémentale étendu [DRFRD06] est utilisé pour segmenter la courbe des points centraux. Si un point central est un point isolé, le segment de droite correspondant est déterminé. Sinon, il est ajouté pour reconnaître un segment flou dans l'espace des tangentes. Si le nouvel ensemble de points centraux n'est plus un segment flou, la reconnaissance du segment flou s'arrête à ce point, et un arc correspondant à la courbe donnée en entrée est détecté. Ce point est considéré comme le premier point du nouveau segment flou dans l'espace des tangentes.

---

#### Algorithme 19 : Décomposition d'une courbe en arcs et segments de droite

---

**Données** :  $C = \{C_1, \dots, C_n\}$ -une courbe discrète,  $\alpha_{max}$ - l'angle maximal admis dans l'espace des tangentes,  $\nu$ - l'épaisseur des segments flous

**Résultat** :  $ARCS$ - l'ensemble des arcs,  $LINES$ - l'ensemble des segments de droite

**début**

Utiliser l'algorithme 10 (voir page 73) pour détecter l'ensemble de points dominants à l'épaisseur 1<sup>10</sup> :  $DP = \{D_0, \dots, D_m\}$ ;  $BS = \emptyset$  ;

Transformer le polygone qui est construit à partir de  $DP$  vers l'espace des tangentes et construire la courbe des points centraux  $M = \{M_0, \dots, M_{m-1}\}$ ;

**pour**  $i=0$  **to**  $m-1$  **faire**

$C_{b_i}C_{e_i}$ - partie de  $C$  qui correspond à  $M_i$ ;

**si**  $M_i$  est un point isolé  $\left( (|M_i.y - M_{i-1}.y| > \alpha_{max}) \vee (|M_i.y - M_{i+1}.y| > \alpha_{max}) \right)$

**alors**

            Ajouter  $C_{b_i}C_{e_i}$  à  $LINES$ ;

**sinon**

**si**  $BS \cup M_i$  est un segment flou d'épaisseur  $\nu$  [DRFRD06] **alors**

$BS = BS \cup M_i$ ;

**sinon**

$C' \leftarrow$  partie de  $C$  qui correspond à  $BS$ ;

                Ajouter  $C'$  à  $ARCS$ ;  $BS = \{M_i\}$ ;

**fin**

---

#### Complexité

La détection des points dominant peut être effectuée en temps  $O(n \log n)$ . La transformation dans l'espace des tangentes est linéaire. Le processus pour déterminer la courbe des points centraux est aussi linéaire. Donc, la complexité de notre méthode est donc  $O(n \log n)$ .

#### 4.5.4 Discussion

La méthode proposée pourrait avoir une complexité linéaire si une polygonalisation linéaire est appliquée au lieu de l'utilisation de la détection des points dominants. Cependant, cette

---

10. Lorsqu'on travaille avec des courbes non-bruitées, l'épaisseur de travail est 1.



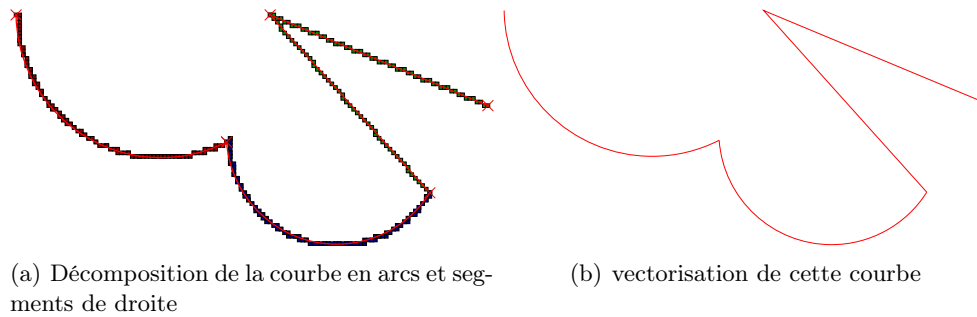


FIGURE 4.25 – Expérimentation sur la courbe de la figure 4.23.

amélioration de performance mène aux décalages des extrémités des primitives. Cet effet est plus claire lorsque la courbe en entrée est une courbe bruitée. Au contraire, l'utilisation de la détection des points dominants nous permet de bien localiser la position de ces extrémités. Nous suggérons d'appliquer un cadre linéaire seulement si la courbe en entrée n'est pas courbe bruitée.

#### 4.5.5 Expérimentations

##### Résultat expérimental

Les figures 4.25.a, 3.35.c, 3.35.g, 3.35.k, 4.27.b, 4.28.c et 4.30 présentent quelques résultats expérimentaux avec notre méthode. Les extrémités entre primitives sont bien localisées grâce la détection des points dominants. Dans ces exemples, il reste encore des petits segments qui font partie d'un arc sont reconnues comme un segments de droite.

##### Comparaison

Salmon et al. [SDRW06] a proposé une méthode avec le même but reposant sur le profil de courbure de la courbe à analyser qui est construit en utilisant la détermination des segments flous à gauche et à droite en chaque point en temps  $O(n^2)$ . Donc, cette méthode est moins efficace que la nôtre. De plus, l'utilisation d'un filtre et la méthode des moindres carrés sur le profil de courbure cause une distorsion dans les résultats de reconstruction. Au contraire, grâce à la détection des points dominants, notre méthode détecte bien les extrémités des primitives. C'est aussi une preuve de robustesse de notre méthode par rapport à celle de Salmon et al. [SDRW06].

Les figures 4.27 et 4.28 présentent quelques expérimentations (décomposition de la courbe en arcs et segments) sur les courbes extraites à partir des images de documents techniques en comparant avec les résultats de Salmon [SDRW06].

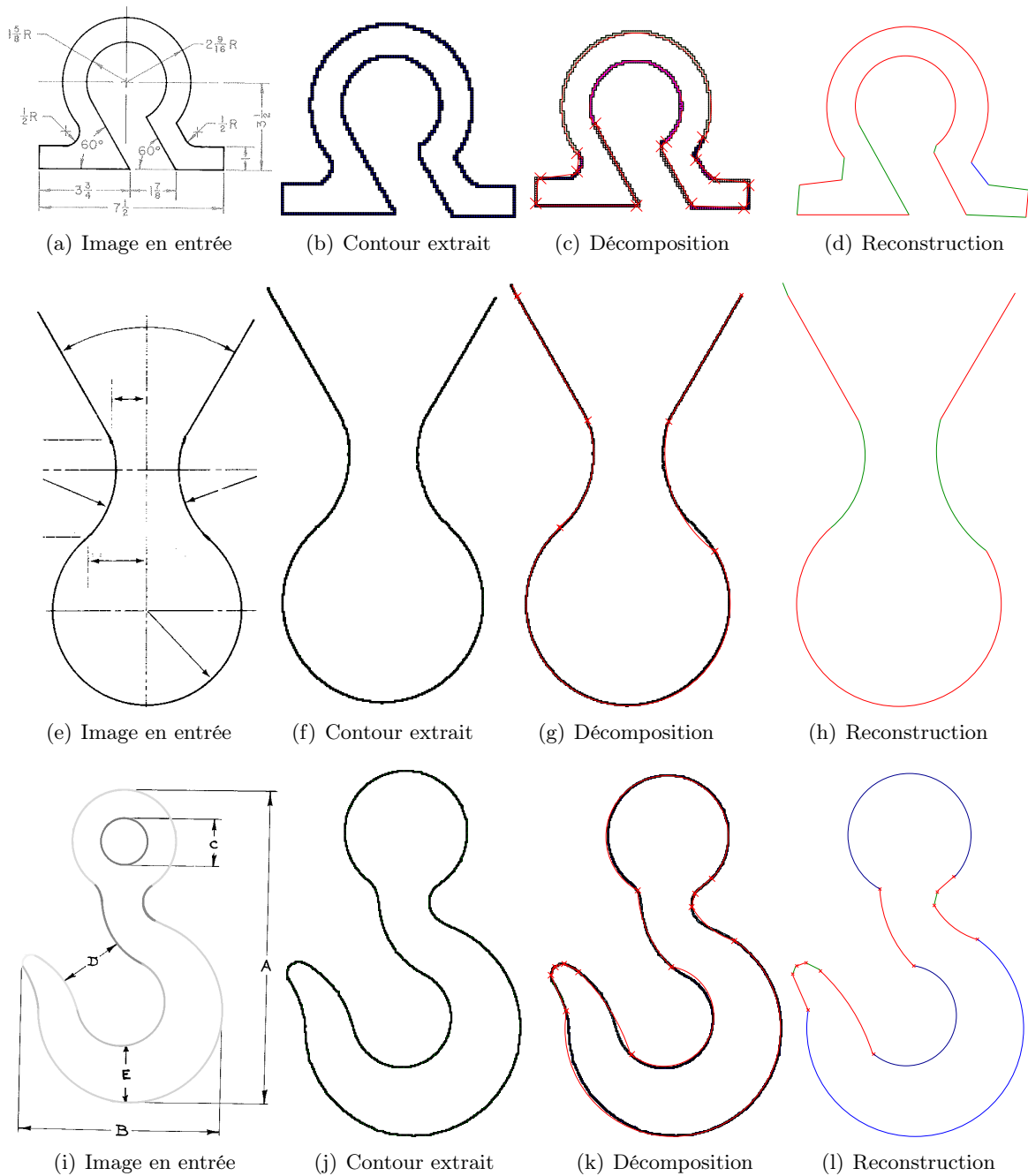


FIGURE 4.26 – Expérimentation sur les images de document technique (suite).

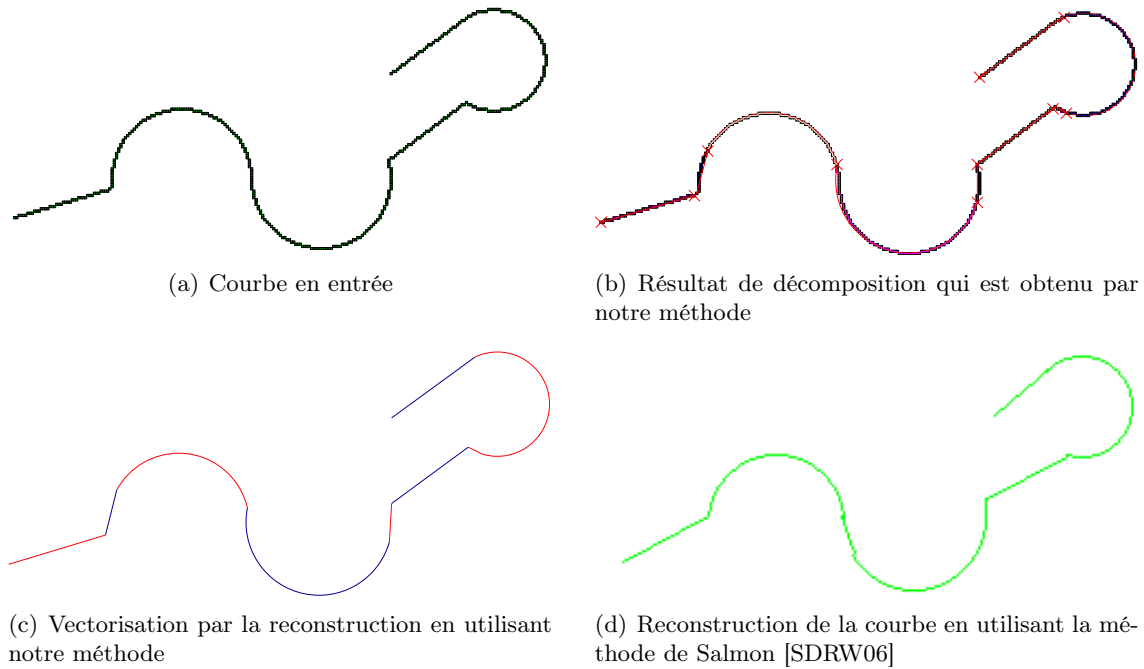


FIGURE 4.27 – Comparaison avec Salmon et al. [SDRW06].

### Application à la vectorisation

Grâce à cette méthode, la figure 4.25.b présente la reconstruction de la courbe reposant sur la décomposition d'une courbe en arcs et segments. C'est aussi l'idée principale pour une application à la vectorisation de la courbe en se basant sur les arcs et segments extraits. Un arc est construit simplement à partir de 2 extrémités et du points central de l'arc. Les figures 3.35.d, 3.35.h, 3.35.l, et 4.31 présentent quelques exemples sur les images de documents techniques ou les symboles de circuit. Les figures 4.30, et 4.32 présentent une comparaison entre la vectorisation de cette méthode et celle de la détection des points dominants sur l'ensemble des courbes données en entrée dans la figure 4.29.

## 4.6 Mesure de la circularité

Dans cette section, nous proposons une nouvelle mesure de circularité pour des courbes discrètes fermées. Nous allons montrer qu'il satisfait les 4 propriétés principales proposées par Haralick pour une mesure de circularité. Nous présentons aussi un algorithme linéaire pour estimer la circularité d'une courbe reposant sur cette mesure.

Nous remarquons que dans cette section, l'espace des tangentes que nous travaillerons est normalisé. C'est pourquoi, nous avons la remarque ci-dessous.

**Remarque 3** Si  $\zeta$  est une courbe convexe fermée (dans le sens des aiguilles d'une montre), la coordonnée de  $T_{n1}$  dans l'espace des tangente est  $(1, 2\pi)$ .

Cette remarque est basée sur la propriété de la somme des angles intérieurs dans un  $n$  - polygone convexe qui est  $(n - 2)\pi$ . Par conséquent, la somme des angles extérieurs de ce polygone est égale à  $n\pi - (n - 2)\pi = 2\pi$ .

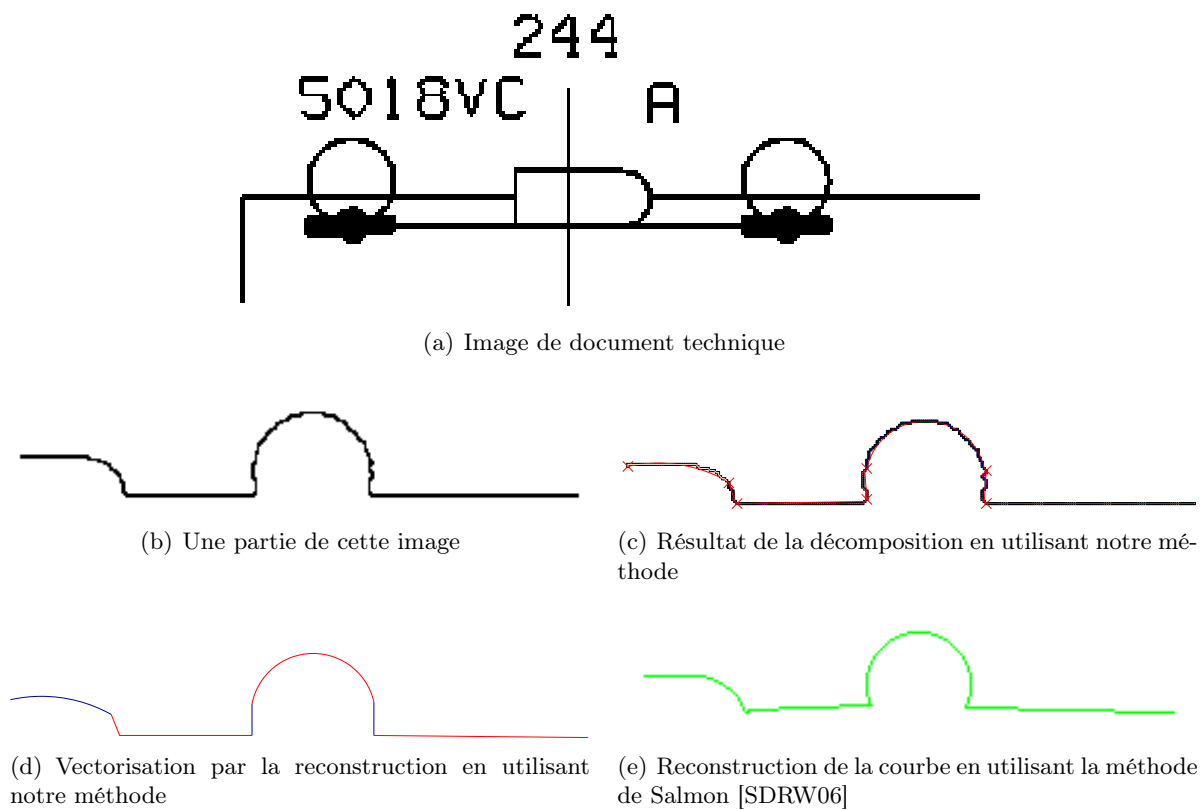


FIGURE 4.28 – Comparaison avec Salmon et al. [SDRW06] sur une image de document technique. Notre méthode localise des coins entre des primitives mieux que celle de Salmon.

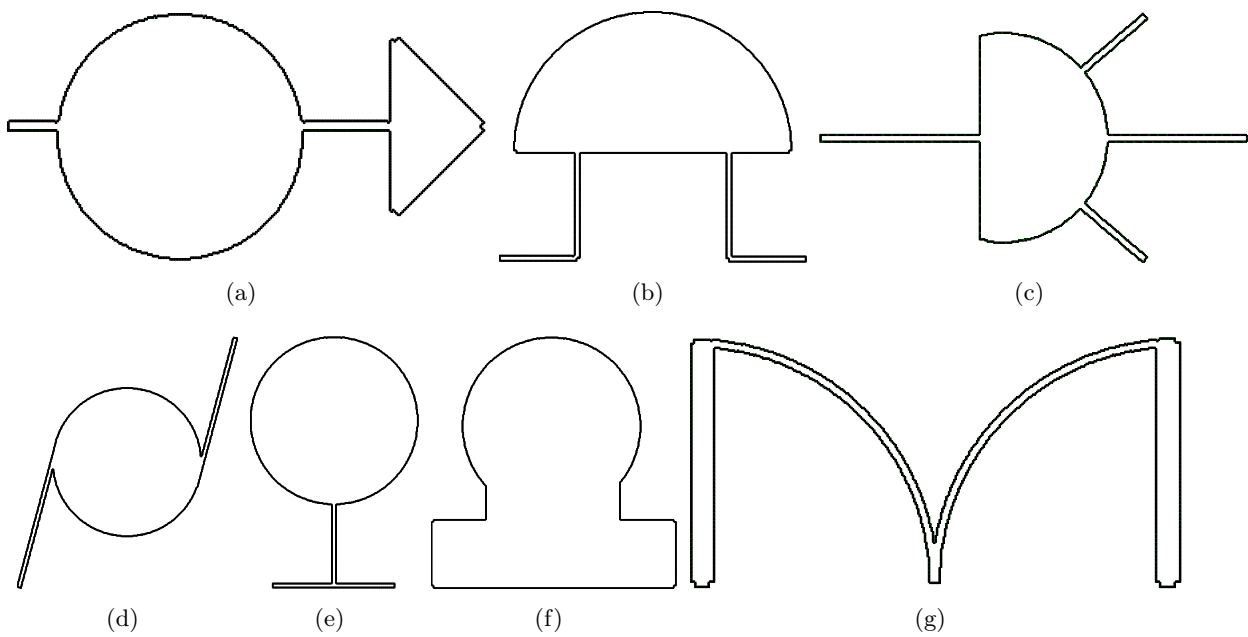


FIGURE 4.29 – Les contours extraits à partir des images de symbole de circuit.

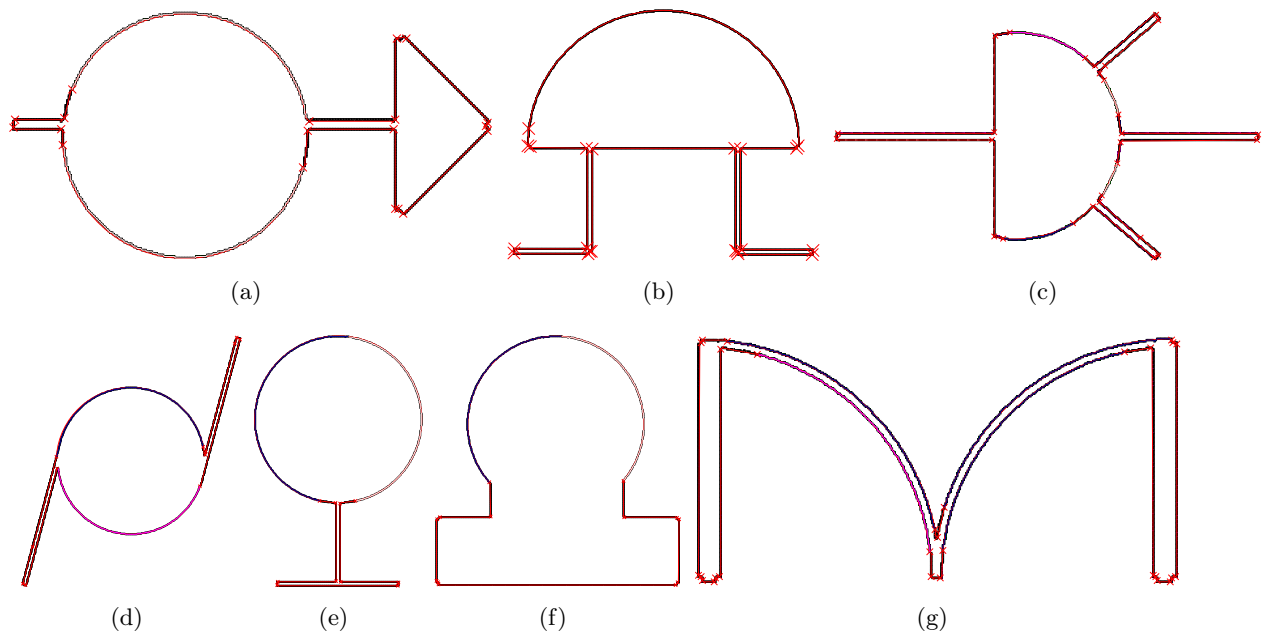


FIGURE 4.30 – Les résultats de segmentation sur les courbes de la figure 4.29.

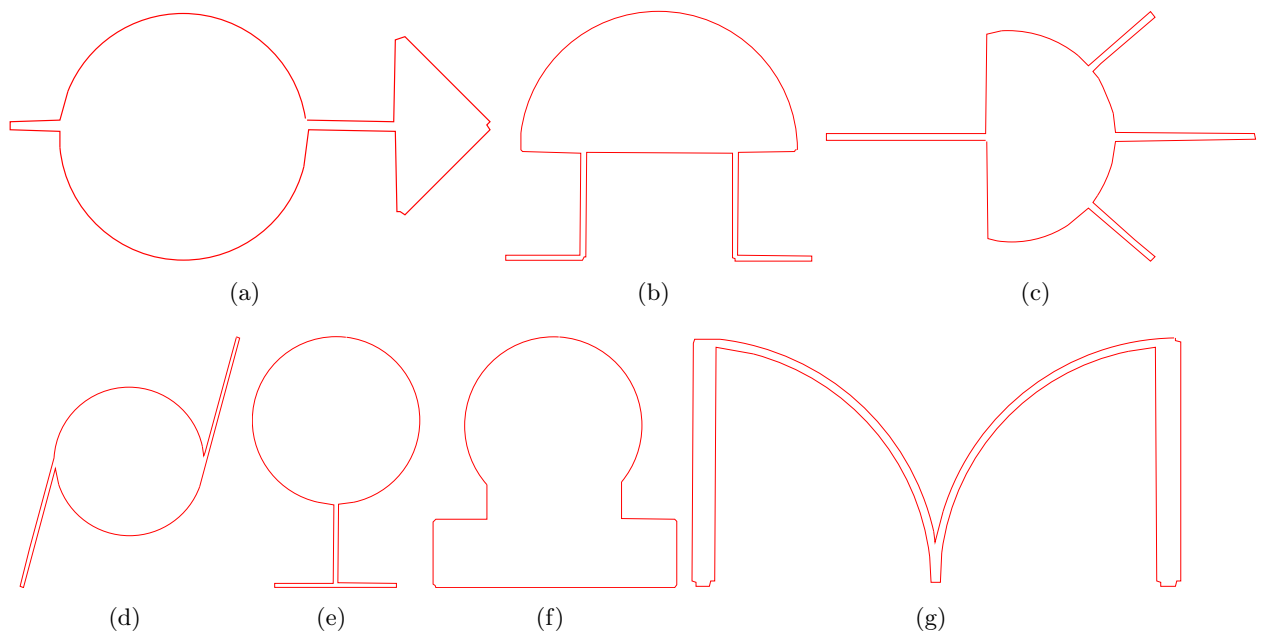


FIGURE 4.31 – Les résultat de vectorisation des courbes de la figure 4.29 reposant sur la décomposition d'une courbe en arcs, segments.

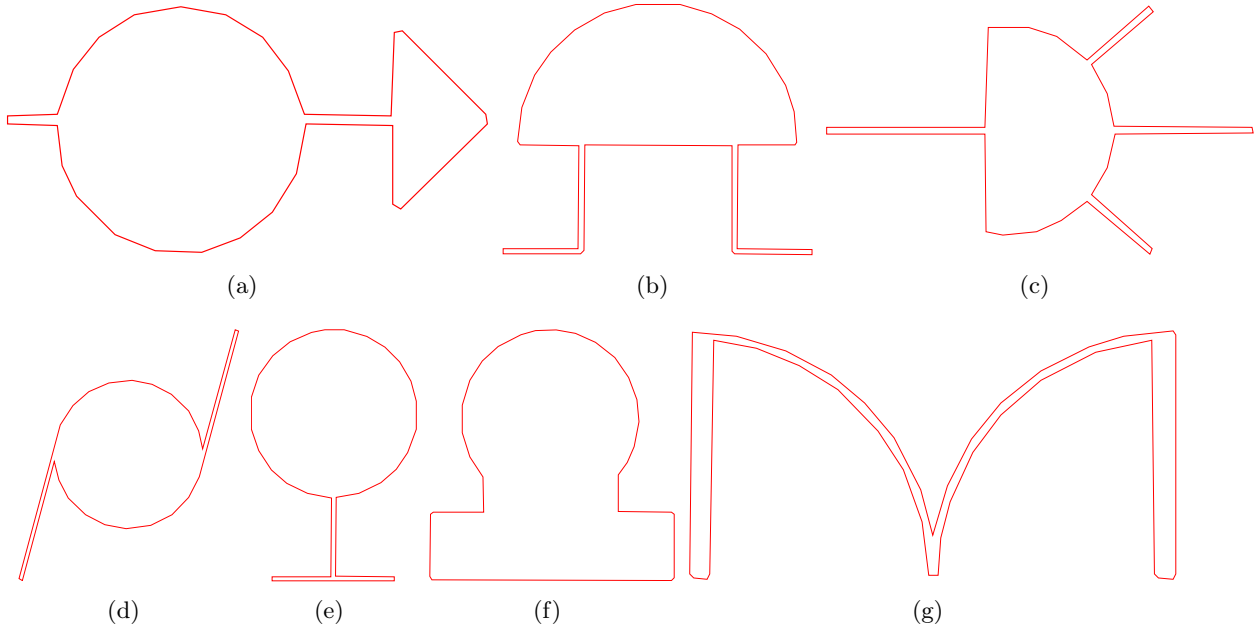


FIGURE 4.32 – Les résultats de vectorisation des courbes dans la figure 4.29 reposant sur la détection des points dominants [NDR09a] (voir la section 3.4). L'épaisseur de travail est 2.

#### 4.6.1 Similarité des courbes polygonales

Supposons que  $c$  et  $d$  sont des polygones ;  $T(c)$ ,  $T(d)$  sont leur fonctions dans l'espace des tangentes qui sont construites à partir de leurs représentations dans cet espace.

Arkin [ACH<sup>+</sup>91] définit une mesure pour estimer la distance entre  $c$  et  $d$  comme suit.

$$d_2(c, d) = \left( \min_{\theta \in \mathbb{R}, t \in [0,1]} \int_0^1 |T_c(s+t) - T_d(s) + \theta|^2 ds \right)^{\frac{1}{2}} \quad (4.10)$$

où  $\theta$  est donné par la formule suivante pour minimiser la valeur de  $\int_0^1 |T_c(s+t) - T_d(s) + \theta|^2 ds$

$$\theta^*(t) = \int_0^1 (T_d(s) - T_c(s+t)) ds = \int_0^1 T_d(s) ds - \int_0^1 T_c(s) ds - 2\pi t \quad (4.11)$$

Arkin a montré que cette mesure est une métrique qui satisfait les propriétés ci-dessous.

- $d_2(a, b) \geq 0$  pour tout  $a$  et  $b$ .
- $d_2(a, a) = 0$  si et seulement si  $a = b$ .
- $d_2(a, b) = d_2(b, a)$  pour tout  $a$ , et  $b$ .
- $d_2(a, b) + d_2(b, c) \geq d_2(a, c)$  pour tout  $a$ ,  $b$  et  $c$ .

Cette métrique compare uniquement les formes polygonales, elle est invariante par translation, rotation et changement d'échelle. De plus, elle correspond bien aux notions intuitives de la ressemblance des formes. Donc, Arkin propose cette métrique comme une mesure de similarité pour la mise en correspondance des formes.

Reposant sur le travail de Arkin, Latecki et al. [LL00, LL02] ont proposé une mesure de similarité entre 2 polygones  $c$  et  $d$  (voir aussi la figure 4.33) qui est présentée comme suit.

$$S_a(c, d) = \left( \int_0^1 (T(c)(s) - T(d)(s) + \theta_0)^2 ds \right) \times \max(l(c), l(d)) \max\left(\frac{l(c)}{l(d)}, \frac{l(d)}{l(c)}\right) \quad (4.12)$$

où  $l(c)$  (resp.  $l(d)$ ) est la longueur de la courbe polygonale  $c$  (resp.  $d$ ). L'intégrale dans l'équation ci-dessus est pondérée avec la longueur d'arc qui est pénalisée par le quotient de longueur des courbes correspondantes. La constante  $\theta_0$  qui est donnée dans le lemme 3 dans [ACH<sup>+</sup>91] est une translation de  $T(d)$  qui minimise l'intégrale  $\int_0^1 (T(c)(s) - T(d)(s) + \theta_0)^2 ds$ . Elle est donnée par le lemme 3 dans [ACH<sup>+</sup>91] comme suit.

$$\theta_0 = \int_0^1 (T(c)(s) - T(d)(s)) ds \tag{4.13}$$

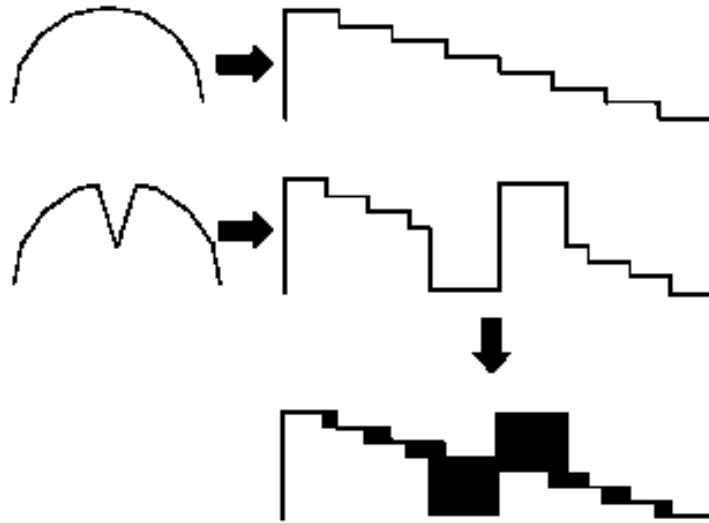
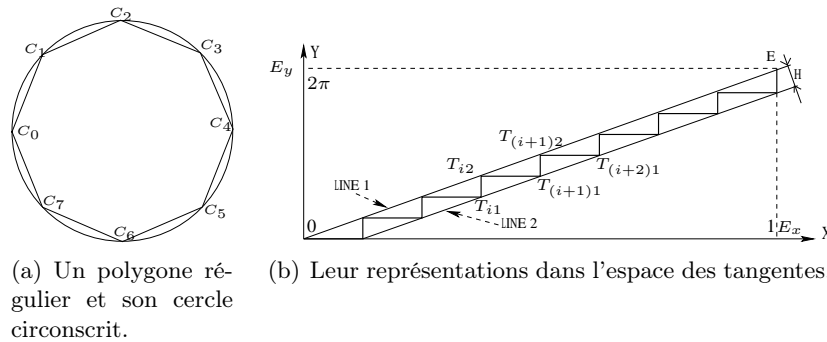


FIGURE 4.33 – Mesure de similarité (Source : Latecki et al. [LL02])

### 4.6.2 La représentation d'un cercle réel dans l'espace des tangentes

Considérons la figure 4.34. La figure 4.34.b correspond à la représentation dans l'espace des tangentes de ce polygone régulier et son cercle circonscrit dans la figure 4.34.a.



(a) Un polygone régulier et son cercle circonscrit.

(b) Leur représentations dans l'espace des tangentes.

FIGURE 4.34 – Un polygone régulier et son cercle circonscrit.

Nous avons le théorème ci-dessous.

**Théorème 4.6.1** *Un cercle réel est représenté dans l'espace des tangentes normalisé par le segment de droite  $OE$  où l'ordonnée et l'abscisse du point  $E$  sont les suivantes :  $E_y = 2 * \pi$  et  $E_x = 1$ .*

**Preuve 4.6.1** *Sans perte de généralité, nous considérons un cercle réel orienté dans le sens des aiguilles d'une montre. Considérons un polygone régulier  $\zeta$  inscrit dans ce cercle. Nous appelons les 2 points extrémités de  $T(\zeta)$  comme suit :  $T_{02} \equiv O(0, 0)$ ,  $T_{n1} \equiv E(1, 2\pi)$ . Grâce à la propriété de régularité, la longueur de chaque côté, et l'angle à chaque sommet de ce polygone sont les mêmes. Donc, dans la représentation correspondante dans l'espace des tangentes de ce polygone (voir la figure 4.34.b), nous avons  $T_{02}T_{11} = \dots = T_{(i-1)2}T_{i1} = \dots = T_{(n-1)2}T_{n1}$ , et  $T_{11}T_{12} = \dots = T_{i1}T_{i2} = \dots = T_{(n-1)1}T_{(n-1)2}$ . Grâce au théorème de Thalès,  $T1 = \{T_{(i+1)1}\}, i = \{1, \dots, n-1\}$  et  $T2 = \{T_{i2}\}, i = \{0, \dots, n-1\}$  sont 2 ensembles des points colinéaires. Considérons la figure 4.34.b, 2 droites réelles  $line1$  et  $line2$  contiennent ces 2 ensembles. Supposons que  $h$  est la distance perpendiculaire entre  $line1$  et  $line2$ . Il est évident que  $0 \leq h < T_{i1}T_{i2} = \frac{2\pi}{n}, i \in \{1, \dots, n-1\}$ . Grâce à  $\lim_{n \rightarrow +\infty} \frac{2\pi}{n} = 0$ , nous obtenons  $\lim_{n \rightarrow +\infty} h = 0$ . De plus,  $line1$  passe toujours le point  $O(0,0)$ , et le point  $E(0,2\pi)$  grâce à la remarque 3 (voir page 142). Par conséquent, quand  $n \rightarrow +\infty$ ,  $line1$  et  $line2$  devient la droite qui passe 2 points :  $O(0,0)$  et  $E(0,2\pi)$ . Puisque  $T(\zeta)$  se trouve entre les 2 droites  $line1$  et  $line2$ , donc quand  $n \rightarrow +\infty$ ,  $T(\zeta) \rightarrow OE$ .*

Par la suite, nous obtenons le corollaire ci-dessous.

**Corollaire 4.6.1** *La fonction de tangente  $T$  d'un cercle réel dans l'espace des tangentes normalisé est  $T(s) = 2\pi s, s \in [0, 1]$*

### 4.6.3 Mesure de circularité et notre algorithme

#### Mesure de circularité

##### Qu'est ce qu'est une bonne mesure de circularité ?

Dans le travail de pionnier [Har74] concernant la mesure de circularité, Haralick a défini certaines propriétés principales d'une bonne mesure de circularité d'une région discrète ou bien d'une courbe discrète fermée. Selon lui, une bonne mesure doit posséder les 4 propriétés ci-dessous.

- ① Plus une région devient circulaire, plus la mesure de sa circularité augmente.
- ② Les valeurs pour des régions discrètes respectent les valeurs pour les régions correspondantes dans l'espace réel.
- ③ Elle est indépendante de l'orientation.
- ④ Elle est indépendante de l'aire.

En fait, les mesures existantes de circularité n'ont pas une même portée de valeur. La valeur de la mesure de Haralick approche  $\infty$  lorsque la région considérée approche à un cercle réel. Dans la même condition, la valeur de mesure de Kim et Roussillon approche 1. Donc, nous changeons la première propriété de Haralick comme "*Plus une région devient circulaire, plus la mesure de sa circularité approche la valeur idéale.*"

#### Notre mesure de circularité

Nous proposons dans cette section une nouvelle mesure de circularité reposant sur la mesure de



similarité des polygones qui est proposée par Latecki et al. [LL00,LL02]. L'idée principale est de déterminer la similarité entre la courbe en entrée  $P$  et un cercle réel. Grâce au corollaire 4.6.1, la fonction de tangente d'un cercle réel est déterminée. Notre mesure de circularité est proposée comme suit.

$$CM_P = \int_0^1 \left( T(P)(s) - 2\pi s + \theta_0 \right)^2 ds \quad (4.14)$$

où  $\theta_0$  est défini par le lemme 3 dans [ACH<sup>+</sup>91] comme suit.

$$\theta_0 = \int_0^1 T(P)(s)ds - \int_0^1 2\pi s ds = \int_0^1 T(P)(s)ds - \pi \quad (4.15)$$

Puisque les fonctions de tangente de tous les cercles réels sont identiques grâce au théorème 4.6.1 et de plus nous voulons mesurer la similarité entre la courbe donnée en entrée et n'importe quel cercle réel, donc la pénalité par la différence de longueur des parties correspondantes [LL00] n'est pas nécessaire.

Cette mesure correspond à la similarité entre une courbe discrète fermée et un cercle réel. Lorsque cette courbe approxime un cercle, la valeur de cette mesure approche 0 qui est la valeur idéale de notre mesure.

### Les propriétés d'invariance

Nous allons montrer que notre mesure satisfait les propriétés proposées par Haralick en ce qui concerne la mesure de circularité.

- ① Lorsque la courbe discrète devient circulaire, grâce au théorème 4.6.1 sa représentation dans l'espace des tangentes de Latecki devient un segment de droite. Alors,  $T(P)(s) \rightarrow 2\pi s$  et  $\theta_0 \rightarrow 0$ . Donc,  $CM_P \rightarrow 0$  C'est pourquoi, *plus une région devient circulaire, plus la mesure de sa circularité s'approche de la valeur idéale.*
- ② Cette mesure est construite à partir de la méthode de Latecki qui sert à mesurer la similarité entre une courbe discrète et un cercle réel. Donc, *les valeurs pour les régions discrètes suivent les valeurs pour les régions correspondantes dans l'espace réel.*
- ③ Si la courbe en entrée est considérée dans le sens direct, le résultat n'est pas modifié en considérant la fonction des cercles réels dans l'espace des tangentes :  $T(s) = -2\pi s$ . Alors, cette mesure est *indépendante de l'orientation.*
- ④ Nous avons montré dans le théorème 4.6.1 que la représentation de tous les cercles réels de rayons différents est la même dans l'espace des tangentes de Latecki. D'autre part, cette mesure est conçue pour mesurer la similarité entre une courbe discrète et un cercle réel. Donc, elle est *indépendante de la superficie.*

De plus, nous savons bien que la représentation dans l'espace des tangente est indépendante de la rotation. C'est pourquoi, notre mesure est *indépendante de la rotation.*

### Algorithme proposé

#### Algorithme

Reposant sur les études théoriques ci-dessus, nous proposons dans l'algorithme 20 une méthode pour mesurer la circularité des courbes discrètes fermées en utilisant notre mesure.

#### Complexité

La transformation de la courbe polygonale vers l'espace des tangentes s'exécute en temps linéaire.

---

**Algorithme 20** : Mesure de la circularité des courbes fermées.

---

**Données** :  $P = \{P_i\}_{i=0}^n$  une courbe fermée (dans le sens des aiguilles d'une montre).

**Résultat** :  $CM_P$  - circularité de la courbe  $P$

**début**

Représenter  $P$  dans l'espace des tangentes par  $T(P)$  (voir la section 4.3.1);

$\theta_0 = -\pi$ ;  $CM_P = 0$ ;

**pour**  $i = 0$  à  $n - 1$  **faire**

$\theta_{0+} = T_{i2}.y(T_{(i+1)2}.x - T_{i2}.x)$ ;

**pour**  $i = 0$  à  $n - 1$  **faire**

$a_i = T_{i2}.y + \theta_0$ ;  $d_i = T_{(i+1)2}.x - T_{i2}.x$ ;  
   $d2_i = T_{(i+1)2}.x^2 - T_{i2}.x^2$ ;  $d3_i = T_{(i+1)2}.x^3 - T_{i2}.x^3$ ;  
   $CM_{P+} = a_i^2 d_i - 2\pi a_i d2_i + 4\pi/3 d3_i$ ;

**return**  $CM_P$  ;

**fin**

---

De plus, il est évident que  $\theta_0$  est déterminé en temps linéaire.

$$\theta_0 + \pi = \sum_{i=0}^{n-1} \int_{T_{i2}.x}^{T_{(i+1)2}.x} (T_{i2}.y) ds = \sum_{i=0}^{n-1} (T_{i2}.y) s \Big|_{T_{i2}.x}^{T_{(i+1)2}.x}$$

Par ailleurs,  $CM_P$  peut être aussi déterminé en temps linéaire grâce à la notion *d'événement critique* présentée dans [ACH<sup>+</sup>91]. Nous supposons que  $T(P)$  est représentée par une suite de segments  $T_{i2}T_{(i+1)1}$ ,  $T_{(i+1)1}T_{(i+1)2}$ . Donc,  $CM_P = \sum_{i=0}^{n-1} \int_{T_{i2}.x}^{T_{(i+1)2}.x} (T_{i2}.y + \theta_0 - 2\pi s)^2 ds =$

$$\sum_{i=0}^{n-1} \left( (a_i^2 s - 2\pi a_i s^2 + 4\pi^2/3 s^3) \Big|_{T_{i2}.x}^{T_{(i+1)2}.x} \right) \text{ où } a_i = T_{i2}.y + \theta_0, i \in \{0, \dots, n-1\}.$$

#### 4.6.4 Expérimentations

##### Résultats expérimentaux

Les figures 4.35, 4.36 et 4.37, le tableau 4.1 présentent des expérimentations sur des images de fruits et des objets non-circulaires.

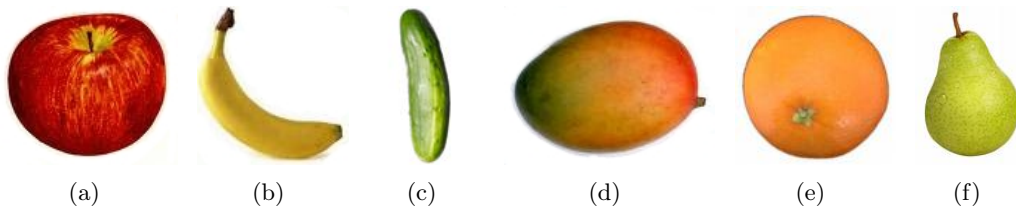


FIGURE 4.35 – Les images de fruit.

##### Images synthétiques

La figure 4.38 présente un profil de mesure de circularité sur une suite de 98 polygones réguliers dont le nombre de côtés varie entre 3 et 60. Ces polygones inscrivent dans un cercle

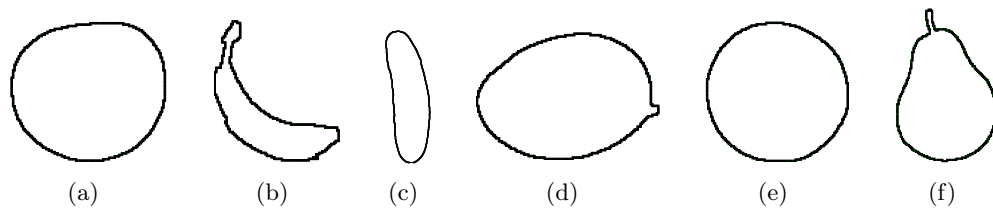


FIGURE 4.36 – Les contours extraits des images dans la figure 4.35.

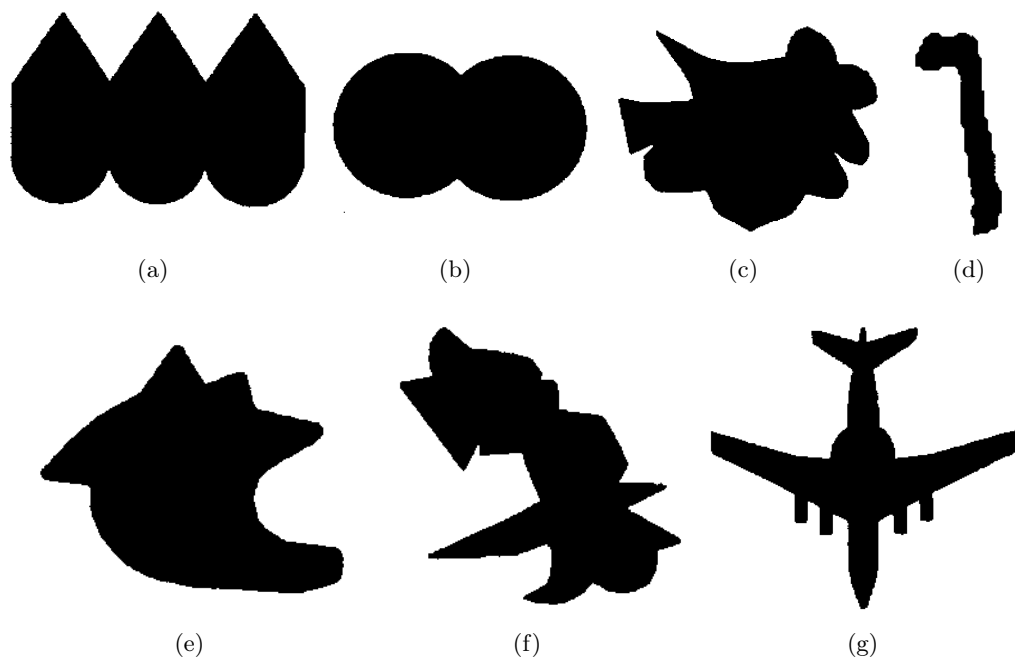


FIGURE 4.37 – Expérimentation sur des objets non-circulaires

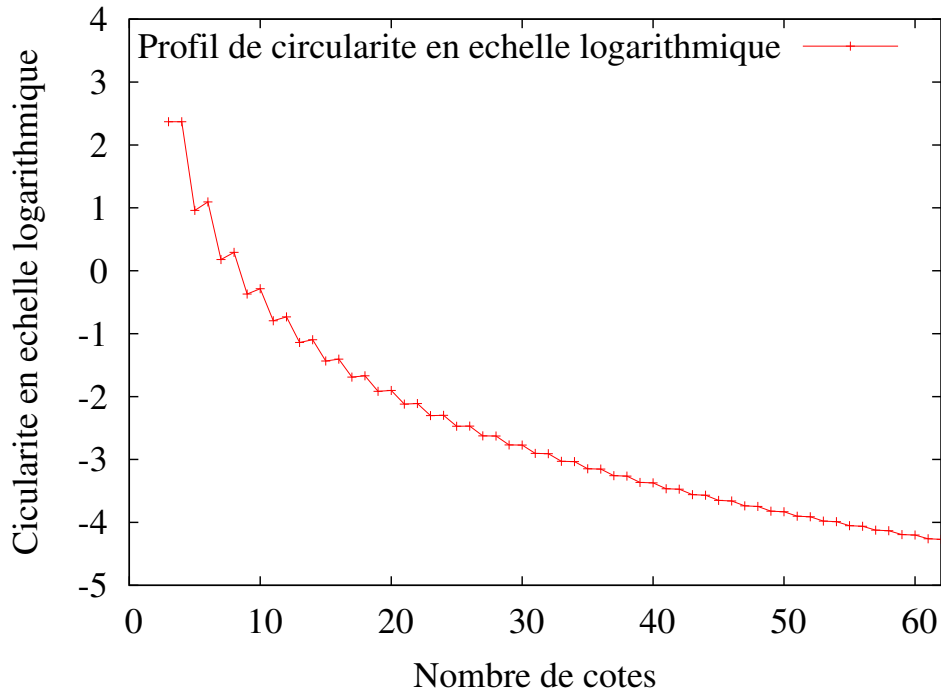


FIGURE 4.38 – Expérimentation sur une suite de polygone réguliers.

dont le rayon est 200. La figure 4.39 présente les 30 premiers polygones réguliers dans cette suite. De plus, nous avons expérimenté aussi sur une suite d'ellipses dont la longueur du grand rayon est fixée à 200, la longueur du petit rayon varie entre 40 et 200. La figure 4.41 présente quelques ellipses dans cette suite aux petits rayons différents. La figure 4.46 présente un profil de mesure de circularité sur une suite des cercles différents dont le rayon varie de 50 à 340.

### Images réelles

Nous appliquons notre méthode sur une étude de cas : un ensemble d'images de pièces de monnaie. Tout d'abord, nous expérimentons sur les monnaies grecques antiques qui ne sont pas vraiment des objets circulaires. Puis, nous appliquons sur des monnaies européennes modernes qui sont des objets circulaires. Les figures 4.42 et 4.44 présentent les images en entrée des pièces grecques antiques et européennes modernes. Les figures 4.43 et 4.45 présentent la segmentation de leurs contours et les mesures correspondantes de circularité qui sont obtenues par notre méthode.

Fig.	a	b	c	d	e	f	g
Fig. 4.36	0.055	15.385	1.591	0.122	0.076	7.422	
Fig. 4.37	1.279	0.212	4.622	4.078	5.92	7.203	10.22

TABLE 4.1 – Mesures de circularité des figures 4.36, 4.37.

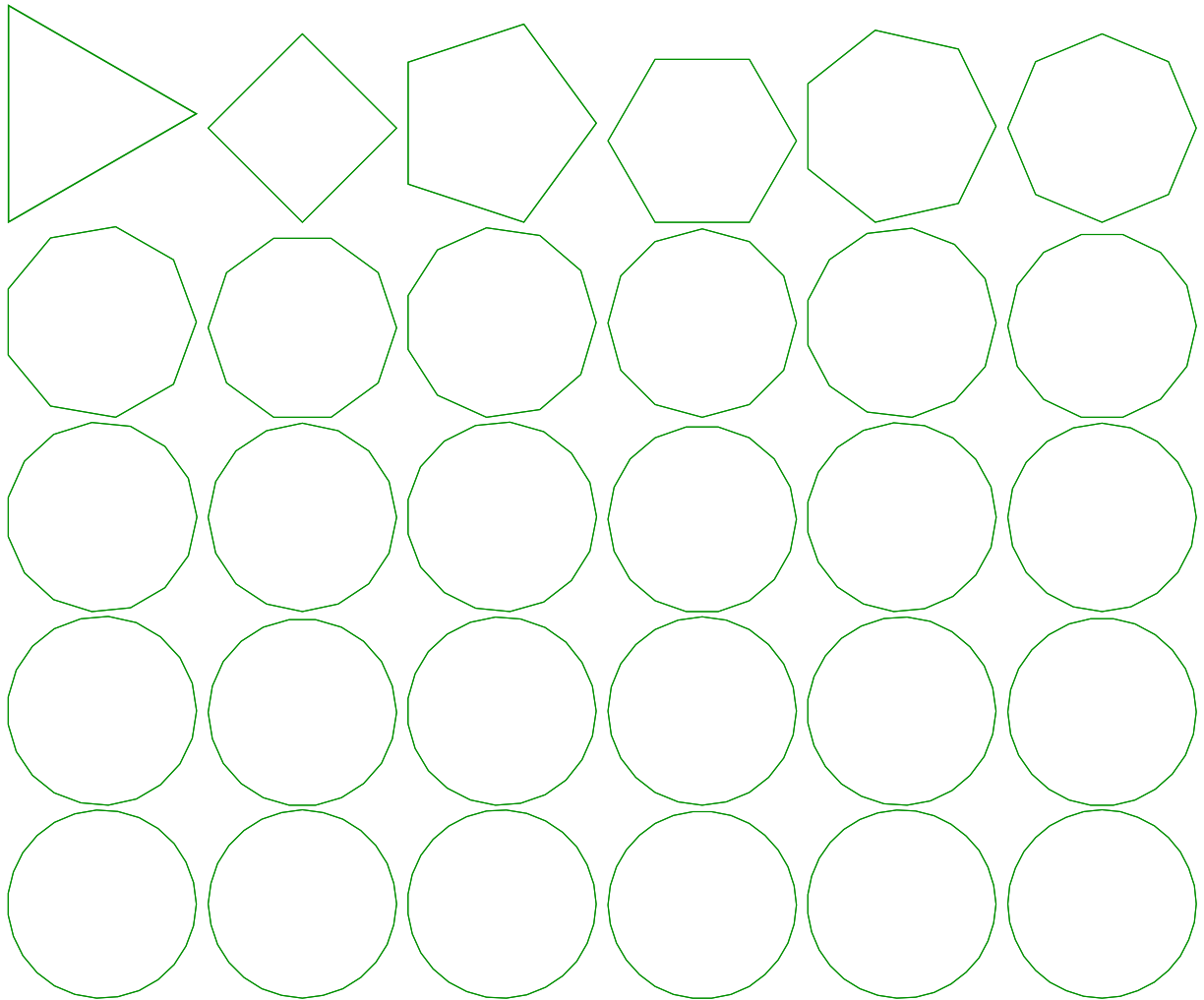


FIGURE 4.39 – Séquence de polygone réguliers dont le nombre de côtés varie entre 3 et 32.

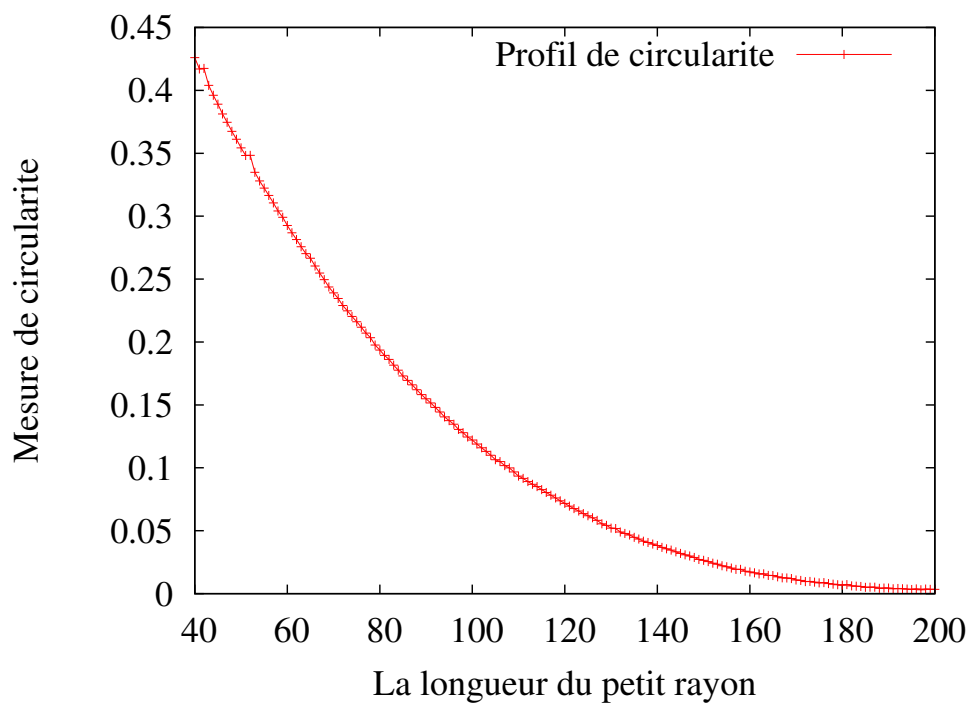


FIGURE 4.40 – Expérimentation sur une suite d’ellipses dont la longueur du grand rayon est fixée à 200, la longueur de petit rayon est variée entre 40 et 200.

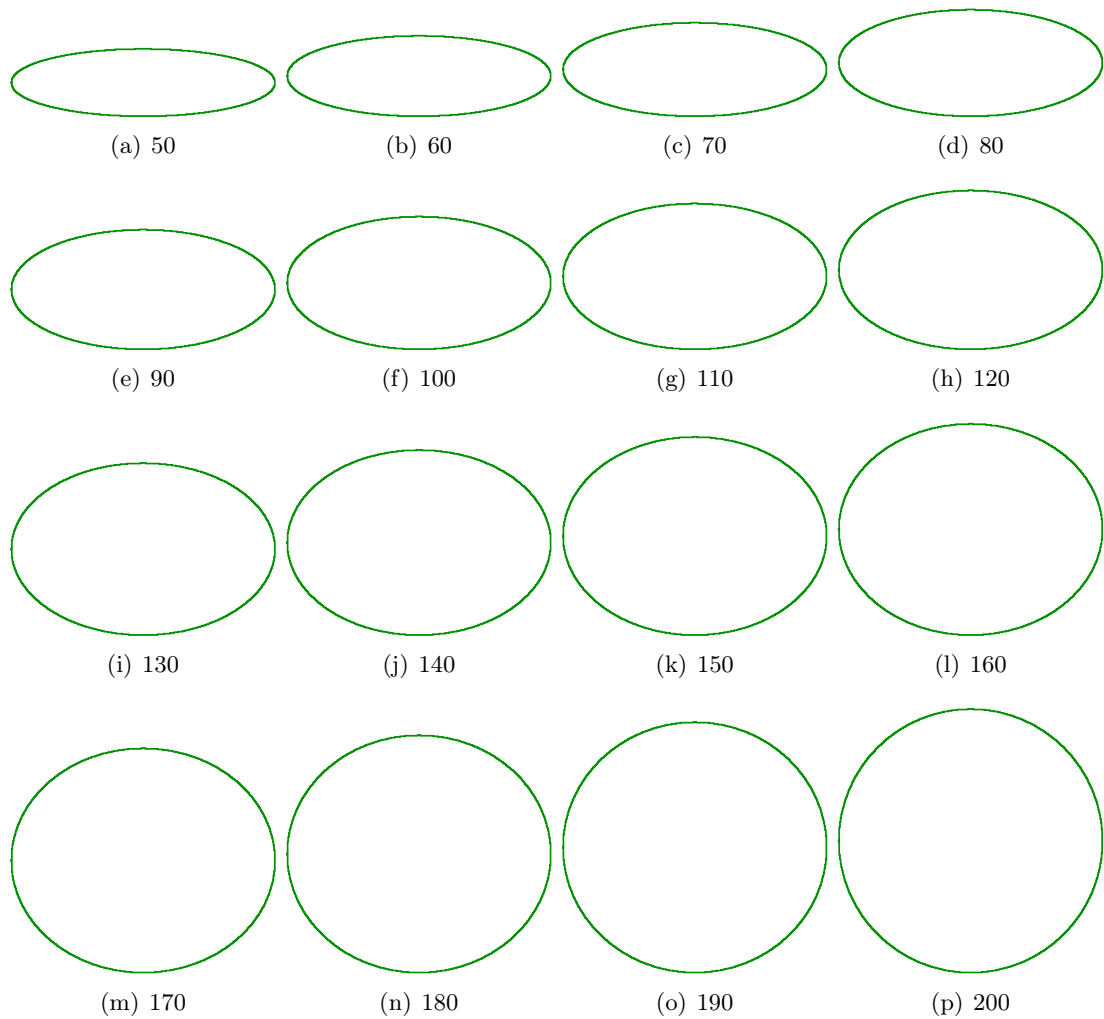


FIGURE 4.41 – Suite d'ellipses dont la longueur du grand rayon est fixée à 200, la longueur du petit rayon varie entre 40 et 200.



FIGURE 4.42 – Images de monnaies grecques antiques.

### Application à la reconnaissance des objets circulaires

Cette méthode peut être utilisée aussi comme une méthode de reconnaissance des objets circulaires si nous utilisons un seuil sur le résultat de circularité. Donc, cette méthode peut reconnaître aussi des objets circulaires en temps linéaire. La figure 4.46 présente les mesures de circularité qui sont obtenues sur plusieurs cercles différents dont le rayon varie de 15 à 95. Grâce au tableau 4.36, à la figure 4.46, nous choisissons 0.08 comme seuil pour la reconnaissance des cercles. La figure 4.45, 4.47 présente quelques cercles qui sont reconnus par cette méthode.

### Comparaison avec les méthodes existantes

Qualitativement, nous présentons ci-dessous quelques comparaisons avec des mesures existantes dans la littérature.

### Comparaison des mesures de circularité

Nous considérons la mesure proposée avec des mesures existantes : Roussillon [RST10], Zunic [ZHR10], Bribiesca [Bri08] sur trois ensembles de données. Le premier est un ensemble de cercles discrets dont les rayons changent de 30 à 200. Le deuxième est un ensemble d'ellipses discrètes dont le demi-diamètre de l'axe majeur est fixé à 200, le demi-diamètre de l'axe mineur est de 40 à 200. Le troisième est un ensemble de polygones réguliers dont le nombre de côtés varie de 3 à 100. De plus, ces polygones sont inscrits dans un cercle dont le rayon est 200. Puisque les mesures ci-dessous ont la même portée de valeur de 0 à 1, nous considérons une mesure modifiée pour avoir une même portée de valeur pour la comparaison comme suit :  $CM' = \frac{1}{1+CM}$  où  $CM$  est la mesure proposée. La figure 4.48 présente une comparaison sur la convergence de la mesure





FIGURE 4.43 – La segmentation de contours des images dans la figure 4.42 et le résultat de mesure de circularité qui est obtenu par notre méthode.



FIGURE 4.44 – Images de monnaie européen.



FIGURE 4.45 – La segmentation de contours des images dans la figure 4.44 et le résultat de mesure de circularité qui est obtenu par notre méthode.

modifiée  $CM'$  avec les mesures existantes. Pour le premier ensemble de données, avec les mesures de Bribiesca et Zunic, notre mesure est meilleure pour mesurer la circularité des cercles discrets car elle s'approche bien à la valeur idéale. Pour le deuxième ensemble, la mesure proposée et celle de Zunic sont meilleures car elle est bien éloignée la valeur idéale pour les courbes non-circulaires. Similairement, pour le troisième, la mesure proposée et celle de Roussillon sont meilleures. La mesure de Bribiesca n'est pas très bien pour 2 derniers ensembles car elle s'approche de 1 pour toutes les courbes dans ces ensembles de données (voir aussi la figure 4.49 pour plus de détails). Ses profils sont quasi horizontaux dans les figures 4.48.b et c. Donc, cette mesure est moins bien pour classifier des objets circulaires.

### Complexité

Cette méthode est simple à implémenter. Sa complexité linéaire est meilleure que les méthodes existantes en ce qui concerne la mesure de circularité telles que méthode de Kim et al [KA84] ( $O(n^3\sqrt{n})$ ), la méthode de Roussillon et al. [RST08] ( $O(n \log n)$ ), Zunic et al [ZHR10] ( $O(n^2)$ ) et Bribiesca [Bri08] ( $O(n^2)$ ). .

## 4.7 Conclusion

Nous avons présenté dans ce chapitre une méthode linéaire de détection d'arcs discrets [NDR10d], et de cercles discrets à partir de laquelle nous avons déduit un algorithme de segmentation de courbes discrètes en arcs et cercles discrets. Cette méthode repose sur une approche de géométrie discrète. Elle est simple, facile et robuste à implémenter. Grâce à sa complexité

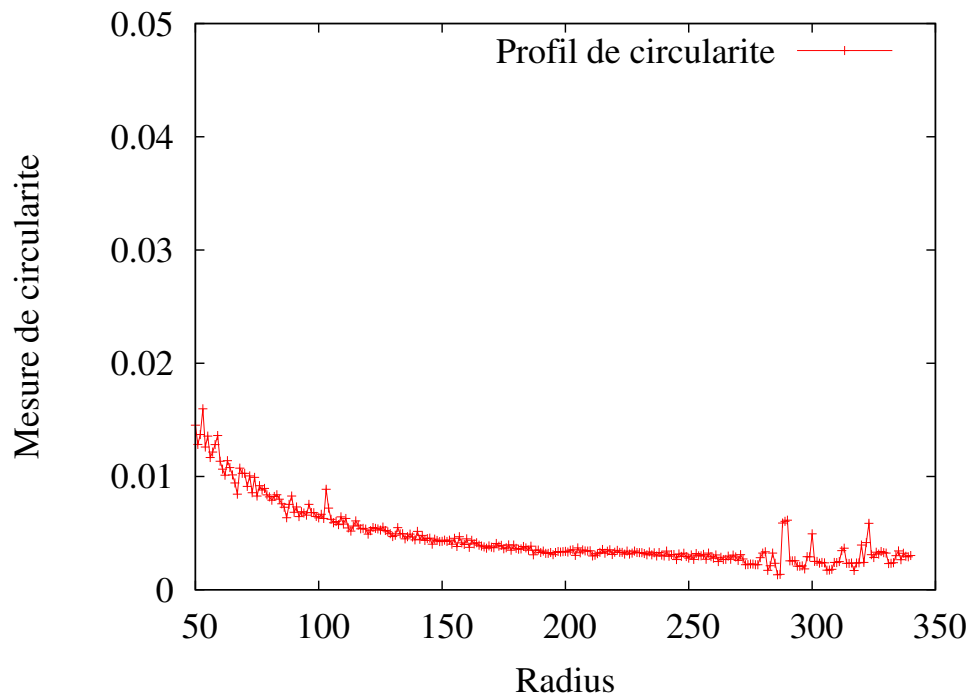


FIGURE 4.46 – Profil de circularité sur des différents cercles dont le rayon varie de 50 à 350.

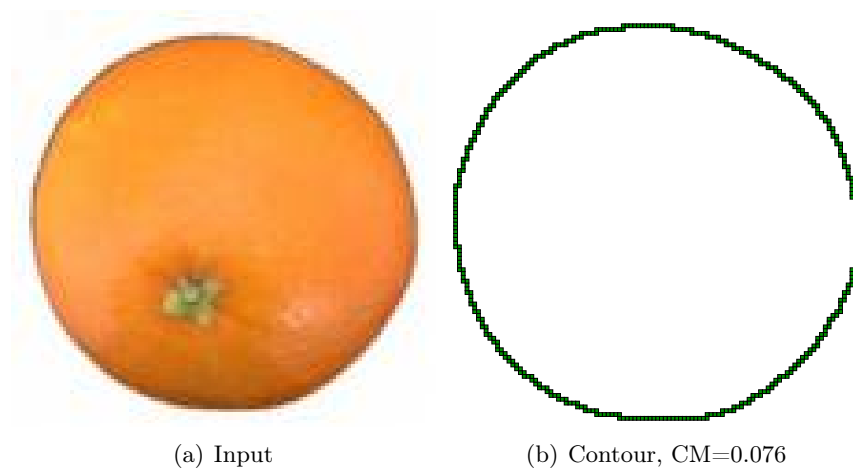
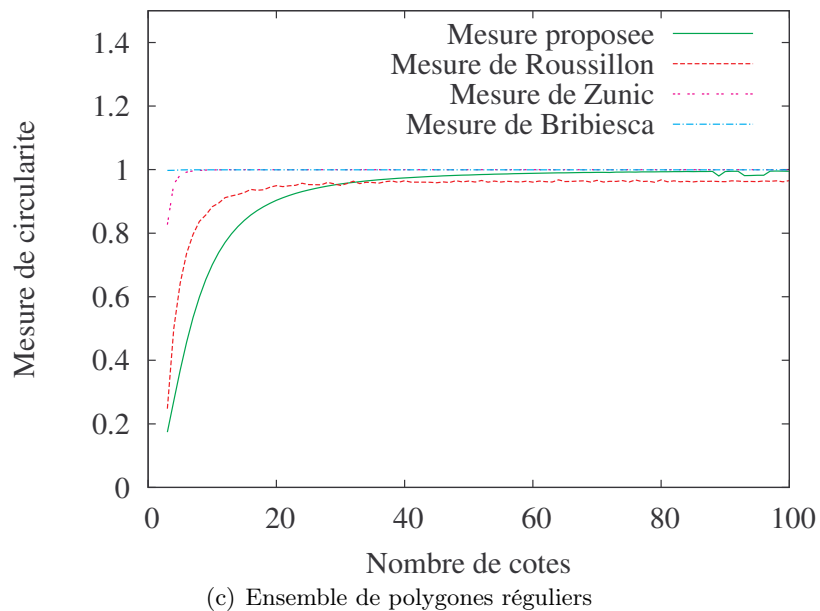
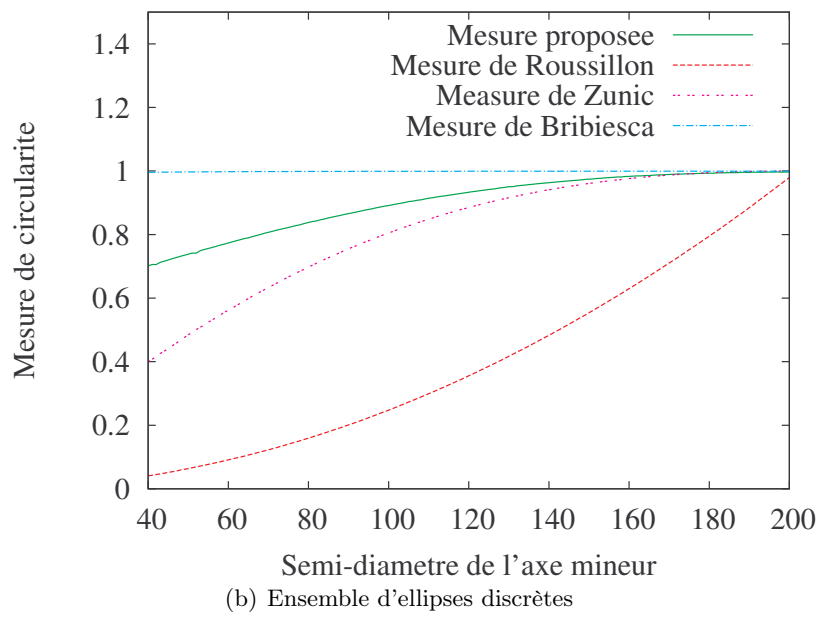
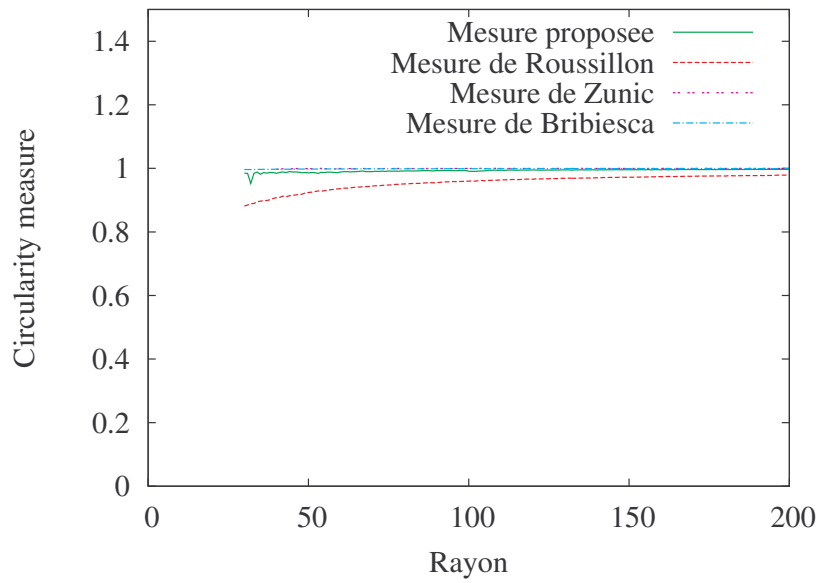


FIGURE 4.47 – Reconnaissance d'un objet circulaire.



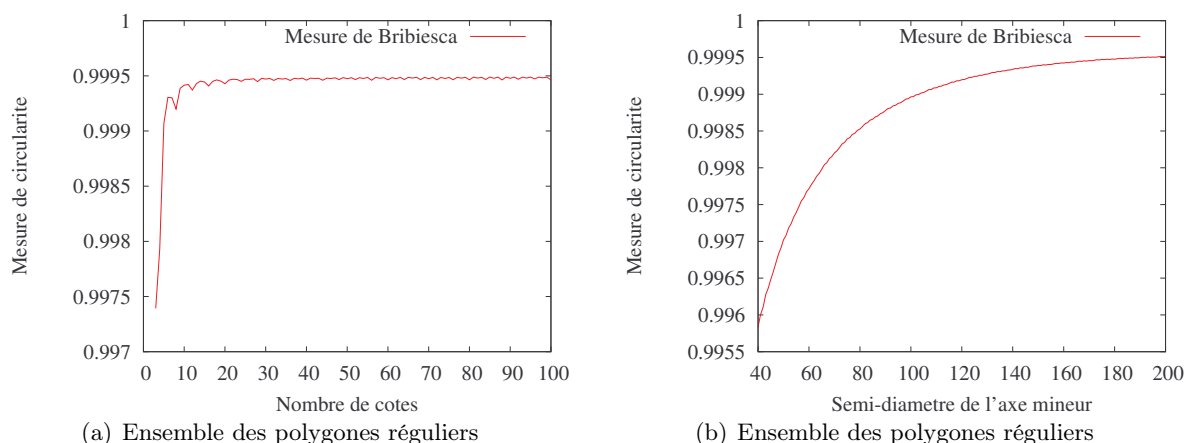


FIGURE 4.49 – Profil de la mesure de Bribiesca.

linéaire, elle est plus efficace que les méthodes récentes [Kov90, CGRT04, RScT09] de segmentation en arcs, et cercles de géométrie discrète. Nous souhaitons utiliser cette méthode pour l'extraction d'arcs de cercles dans des images de documents techniques. Cette méthode a été publiée dans la conférence nationale RFIA'10 [NDR10f]. Elle a été mise en ligne par le travail d'un ingénieur. Les utilisateurs peuvent la tester directement sur cette adresse suivante : <http://kerrecherche.iutsd.uhp-nancy.fr/~ines/ArcSegmentation.html>.

Nous avons proposé aussi une extension de ce travail pour les courbes bruitées. La technique de détection du bruit a été appliquée pour déterminer l'épaisseur des segments flous qui sert à la polygonalisation dans deux hypothèses différentes de distribution du bruit. Cette méthode a été publiée dans la conférence ICCVG'10 [NKDROL10].

Nous avons présenté dans ce chapitre une méthode pour la décomposition d'une courbe en segments et arcs en temps  $O(n \log n)$  [NDR10b]. Elle utilise la détection des points dominants pour renforcer la qualité de la décomposition.

Nous avons présenté aussi dans ce chapitre une méthode pour mesurer la circularité de courbes fermées en proposant une nouvelle mesure de circularité. Cette mesure est basée sur les outils de mesure de similarité de Arkin [ACH<sup>+</sup>91] et de Latecki [LL00]. Cette méthode a été publiée dans la conférence ICPR'10 [NDR10a].

Même si plusieurs problèmes ont été rencontrés et surmontés, il reste encore des défis à relever en ce qui concerne ce thème de recherche. Nous détaillons dans le chapitre de conclusion (voir la page 160) les perspectives de ces travaux dans le futur.

## Remerciements

Nous tenons à remercier Eric Domenjoud pour ses précieux commentaires en ce qui concerne le travail de reconnaissance des cercles discrets [NDR10f].



# Conclusions et Perspectives

Les travaux menés durant cette thèse abordent différents problèmes de la géométrie discrète et de l'analyse d'images. Nous avons présenté des méthodes originales et performantes permettant d'étudier des courbes discrètes en 2D et en 3D. Celles-ci mènent aussi à des applications en analyse d'images. Brièvement, nos contributions sont les suivantes :

- Estimation de propriétés locales :
  - Estimation de la courbure [NDR07] en un point d'une courbe 2D bruitée.
  - Estimation de la courbure et de la torsion [NDR08a] en un point d'une courbe 3D bruitée.
- Détection des points dominants et représentation polygonale :
  - Détection des points dominants [NDR09a, NDR10c] en utilisant un paramètre fixe.
  - Méthode sans paramètres [NDR09b] de représentation polygonale reposant sur une approche multi-épaisseur.
  - Méthode adaptative [NDR10e] de représentation polygonale reposant sur la stratégie de découpage et fusion.
- Cercle discret et applications :
  - Reconnaissance des cercles et arcs et segmentation d'une courbe en arcs discrets [NDR10d], [NDR10f] en temps linéaire.
  - Segmentation d'une courbe bruitée en arcs discrets [NKDROL10].
  - Décomposition d'une courbe en arcs et segments de droite [NDR10b].
  - Mesure de la circularité en temps linéaire [NDR10a].

Nous détaillons ci-dessous notre contribution pour chacun de ces thèmes.

## Estimation de propriétés locales

### Résumé

Après avoir fait un bref historique des très nombreux travaux effectués au sujet de l'estimation de la courbure, nous avons montré l'intérêt de l'utilisation de segments flous ainsi que la représentation d'une courbe 2D par la suite de ses segments flous maximaux. Ceux-ci nous permettent de travailler avec des courbes bruitées en réduisant la complexité de l'estimation de la courbure. Nous proposons une solution pour déterminer une évolution dynamique d'un segment flou d'épaisseur  $\nu$  lorsqu'on ajoute (resp. supprime) un point à (resp. d') un segment flou reposant sur le calcul de l'évolution dynamique d'une enveloppe convexe et la technique de détermination de l'épaisseur de l'enveloppe convexe. Cette solution mène à une construction de la suite des segments flous maximaux d'une courbe discrète en temps  $O(n \log^2 n)$ . Nous utilisons cette représentation efficace pour estimer la courbure discrète en chaque point d'une courbe avec une même complexité.

Par la suite, nous avons fait évoluer ce travail dans le cas 3D pour estimer des propriétés locales d'une courbe discrète 3D : la courbure et la torsion. Pour le faire, une notion étendue

de segment flou 3D a été proposée. Donc, l'estimation de la courbure 3D peut être effectuée similairement au cas 2D. En ce qui concerne l'estimation de la torsion, nous proposons une estimation indirecte reposant sur l'estimation de la courbure de la courbe binormale qui est construite à partir des vecteurs binormaux de la courbe donnée en entrée.

## **Perspectives**

Les méthodes proposées permettent d'extraire des propriétés locales des courbes 2D et 3D. C'est pourquoi, nous espérons utiliser ces méthodes pour identifier et classifier des objets 2D, 3D.

Nous espérons aussi proposer une méthode efficace pour estimer la courbure en chaque point d'une surface discrète. Notre idée est de considérer les courbes discrètes qui passent à chaque point de la surface. En estimant les courbures de ces courbes à ce point, nous pouvons déduire sa valeur de courbure à la surface à ce point.

## **Détection des points dominants et représentation polygonale**

### **Résumé**

Les points dominants jouent un rôle très important dans la représentation et la mise en correspondance des formes. Dans le cadre de notre thèse, nous nous sommes intéressés à une approche géométrique de la détection des points dominants afin de travailler avec des courbes pouvant être bruitées. Après avoir étudié plusieurs méthodes issues de la littérature, nous avons montré une relation entre la représentation d'une courbe par sa suite de segments flous maximaux et la détection des points dominant de la courbe. Un candidat point dominant se trouve au milieu d'une zone commune de segments flous maximaux dont les pentes sont croissantes ou bien décroissantes. C'est pourquoi, une méthode efficace de détection de points dominants a été proposée reposant sur la représentation d'une courbe discrète par sa suite de segments flous maximaux. Grâce à la notion de segment flou d'épaisseur  $\nu$ , cette méthode peut travailler naturellement avec des courbes bruitées en changeant le paramètre d'épaisseur.

Nous nous sommes ensuite intéressés aux méthodes non-paramétriques afin de travailler automatiquement avec des courbes bruitées. Nous avons proposé deux méthodes sans paramètres. La première méthode est une approche multi-épaisseur qui détermine l'épaisseur la plus appropriée pour détecter les points dominants. Ce cadre de travail repose sur un nouveau critère d'évaluation qui permet d'évaluer le résultat de la détection à travers différentes épaisseurs. La deuxième méthode est une solution adaptative qui permet de déterminer les parties significatives de la courbes avec les points dominants correspondants. Elle est basée sur la technique de découpage-et-fusion qui permet de déterminer l'épaisseur appropriée de chaque partie significative de la courbe.

## **Perspectives**

Dans le futur, nous espérons construire un système en ligne qui permet de tester nos méthodes de détection des points dominants et de polygonalisation en les comparant avec les autres méthodes existant dans la littérature.

Nous nous intéressons aussi à une application de mise en correspondance reposant sur nos méthodes de polygonalisation.



---

## Cercle discret et applications

### Résumé

Les arcs et les cercles discrets sont des objets de base en géométrie discrète. Ils apparaissent souvent dans les images. A la fin de la thèse, nous avons étudié ces primitives importantes d'un point de vue théorique. Nous nous intéressons aussi aux applications directes de cette étude en analyse d'images.

Tout d'abord, nous nous intéressons à une méthodes de reconnaissance de cercles et d'arcs discrets. Après avoir étudié les méthodes existantes dans la littérature, nous avons montré l'intérêt de la représentation d'une courbe polygonale dans l'espace des tangentes pour la reconnaissance des cercles discrets. Nous avons introduit une propriété importante concernant la colinéarité de la courbes des points centraux dans l'espace des tangentes lorsqu'on considère la représentation dans cet espace d'un polygone approximé d'un cercle réel. Cette propriété nous permet de convertir le problème de reconnaissance d'un cercle en un problème de reconnaissance d'un segment de droite. Parmi plusieurs outils existants qui nous permet de vérifier la colinéarité d'un ensemble de points, nous avons choisi la reconnaissance des segments flous afin de bien déterminer le cercle d'Andrès qui contient la courbe donnée en entrée. Notre méthode qui est linéaire peut travailler avec des cercles discrets ainsi qu'avec des arcs discrets.

Ensuite, nous avons utilisé cet algorithme de reconnaissance des cercles et arc discrets pour l'analyse des courbes bruitées. Nous appliquons la technique de détection du bruit de Kerautret et Lachaud pour avoir l'information du bruit présent dans la courbe donnée en entrée. Cette information est utilisée pour construire le polygone approximé qui tient compte du bruit de la courbe. Nous utilisons deux hypothèses différentes sur la distribution du bruit. La première hypothèse suppose que la distribution de bruit est uniforme. La deuxième hypothèse correspond à une distribution non-uniforme du bruit. Le polygone approximé est construit en fonction de ces hypothèses. Dans le cas d'une distribution uniforme, nous utilisons la valeur moyenne du bruit comme épaisseur des segments flous du polygone construit. Dans l'autre cas, la valeur du bruit du point début est considérée comme épaisseur à reconnaître du segment flou correspondant. C'est pourquoi, cette méthode travaille avec des courbes bruitées dans les deux conditions de bruit. L'utilisateur doit choisir la distribution du bruit avant d'appliquer notre méthode.

D'autre part, nous nous intéressons aussi à une méthode pour décomposer une courbe discrète en segments de droite et arcs. Une décomposition de courbe reposant sur ces primitives nous permet d'obtenir une bonne description la forme. Cette description qui joue un rôle important dans la reconnaissance des formes, la mise en correspondance des formes a plusieurs avantages par rapport à l'utilisation des primitives linéaires telles que points dominants, segments de droite, etc. Elle réduit le nombre de primitives ainsi que l'erreur d'approximation lorsqu'on travaille avec un arc de cercle. Nous proposons une solution hybride pour augmenter la robustesse de la décomposition. Tout d'abord, la technique de détection des points dominants est appliquée. Ensuite, le polygone qui est construit à partir de cet ensemble de points dominants est représenté dans l'espace des tangentes. Puis, les primitives arcs ou segments de droite sont détectées grâce à la détection des points isolés sur le profil de points centraux dans l'espace des tangentes.

Enfin, nous proposons une méthode linéaire qui permet de mesurer la circularité des courbes fermées en déterminant la similarité entre la courbe donnée en entrée et la discrétisation d'un cercle réel. Nous avons montré que tous les cercles réels ont une même représentation dans l'espace des tangentes. Donc, nous pouvons mesurer la similarité entre la courbe en entrée et un cercle réel reposant sur la métrique de similarité de Latecki qui est calculée sur la représentation dans l'espace des tangentes. Notre mesure de circularité satisfait toutes les demandes de Haralick et

elle peut être déterminée simplement en temps linéaire. Cette méthode peut être considérée aussi comme une méthode de reconnaissance d'objets circulaires.

## Perspectives

Les études des primitives cercles et arcs discrets ont été effectuées à la fin de ma thèse. Plusieurs problèmes intéressants ont été creusés, mais il reste encore des travaux théoriques ainsi que pratiques en ce qui concerne cette thématique. Dans le futur, nous allons compléter les problèmes suivants qui ne sont pas encore réglés.

- *Reconnaissance des cercles discrets* : Nous souhaitons comparer notre méthode avec les autres méthodes de reconnaissance des arcs, des cercles discrets issues de la littérature. Nous nous intéressons aussi à une preuve de convergence des méthodes de reconnaissance des cercles discrets. Notre idée est de convertir des cercles reconnus vers un seul type de cercles discrets : cercle d'Andrès. Cette définition nous permet de considérer un ensemble de cercles avec les différentes épaisseurs.

- *Mesure de circularité 2D* : Nous allons comparer notre méthode avec des autres méthodes dans la littérature : celle de Haralick, celle de Rousillon. Une étude de convergence entre des méthodes de mesure de circularité sera effectuée aussi. Par ailleurs, nous nous intéressons à étudier une extension de la méthode de mesure de circularité pour les courbes non-fermées. Le défi principal pour mesurer la circularité d'une courbe non-fermée est de déterminer l'arc réel le plus approximatif de la courbe donnée en entrée. Si nous utilisons directement la différence entre les ordonnées verticales de sa représentation dans l'espace des tangentes comme l'angle de l'arc réel approximatif, nous n'obtenons pas une méthode stable en réalité. Une piste pour régler ce problème est d'utiliser la technique de régression linéaire pour approximer la représentation dans l'espace des tangentes par une droite réelle dont la pente détermine bien l'arc réel correspondant. Bien évidemment, une méthode de mesure de circularité des courbes non-fermées peut mener à une nouvelle approche incrémentale de décomposition d'une courbe en arcs et en segments de droite. Nous effectuerons aussi une comparaison entre notre méthode de reconnaissance de cercle et celle de reconnaissance des objets circulaires reposants sur la mesure de circularité.

- *Version étendue de mesure de circularité en 3D* : Nous avons proposé une mesure de circularité pour les courbes fermées [NDR10a] et puis une version étendue pour les courbes non-fermées [?]. Dans l'avenir, nous espérons travailler sur des objets 3D en proposant une mesure convenable pour déterminer la différence entre un objet discret 3D et une sphère. Nous nous intéressons à deux approches potentielles. La première approche correspond à la détermination de la mesure 3D en transformant le problème vers la mesure de circularité en 2D. La deuxième approche s'intéresse à une estimation directe de la similarité entre l'objet étudié et une sphère.

- *Détection d'ellipses en images* : Dans [NDR10f, NDR10d], nous avons proposé une méthode simple et linéaire pour détecter des cercles, des arcs discrets. Naturellement, nous pensons à faire évoluer ce travail pour détecter des autres primitives non-linéaires. Tout d'abord, nous travaillerons spécifiquement sur des primitives d'ordre 2 telles que les ellipses, les paraboles. Parmi ces primitives, nous nous intéressons fortement à la détection des ellipses car cette primitive apparaît souvent dans les images. Donc, une telle méthode mène directement à des applications en analyse d'images.

Notre approche se base sur un modèle d'approximation d'une ellipse par 4 arcs de cercle. Etant donnée une courbe discrète fermée, nous essayons de détecter 4 arcs correspondants qui forment une approximation d'une ellipse. Nous avons obtenu des résultats intéressants à partir de cette idée. Un article est en cours de rédaction sur ce sujet. Dans l'avenir, nous essayerons détecter des ellipses et des morceaux d'ellipses dans images.

---

- *Etude de la représentation d'une courbe discrète par la suite de ses arcs maximaux* : Un segment maximal d'une courbe discrète est un segment qui ne peut être élargi ni à gauche ni à droite sur cette courbe. Depuis plusieurs années, la représentation d'une courbe par la suite de ses segments maximaux a fait l'objet de nombreuses études dans la communauté de géométrie discrète. Cette représentation mène à des applications importantes en analyse d'images telles que l'estimation de la courbure [NDR07] en chaque point d'une courbe discrète, sa représentation polygonale [Fes05], la détection de ses points dominants [NDR09b] ou encore la détection du bruit présent dans la courbe [KL09a]. Nous souhaitons élargir cette étude avec des primitives non-linéaires comme des arcs de cercles.

Dans le cadre de cette perspective, nous nous intéressons à la représentation d'une courbe discrète par des primitives maximales non-linéaires. La primitive non-linéaire la plus simple est l'arc de cercle. De la même façon que pour les segments, on peut définir un arc de cercle maximal d'une courbe discrète comme un arc de cercle qui ne peut être élargi ni à gauche ni à droite sur cette courbe. Dans ce travail, nous nous intéressons à l'étude de la représentation d'une courbe par sa suite d'arcs maximaux. L'objectif est de pouvoir déduire de cette représentation des caractéristiques géométriques qui pourront être utilisées dans le cadre d'applications en analyse d'images.



# Liste des publications

- [1] Nguyen, T.P., Debled-Rennesson, I. : Curvature estimation in noisy curves. In : CAIP. Volume 4673 of LNCS. (2007) 474–481
- [2] Nguyen, T.P., Debled-Rennesson, I. : Curvature and torsion estimators for 3d curves. In : ISVC (1). Volume 5358 of LNCS., Springer (2008) 688–699
- [3] Nguyen, T.P., Debled-Rennesson, I. : On the local properties of digital curves. International Journal of Shape Modeling **14** (2008) 105–125
- [4] Nguyen, T.P., Debled-Rennesson, I. : A linear method for segmentation of digital arcs (2010) Rapport de recherche, <http://www.loria.fr/~nguyentp/pubs/ArcSegmentation.pdf>.
- [5] Nguyen, T.P., Debled-Rennesson, I. : Fast and robust dominant point detection on digital curves. In : ICIP, Cairo, Egypt (2009) 953–956
- [6] Nguyen, T.P., Debled-Rennesson, I. : A discrete geometry approach for dominant point detection. Pattern Recognition (2010) to appear.
- [7] Nguyen, T.P., Debled-Rennesson, I. : Free-parameter method for polygonal representation of noisy curves. In : IWCIA, Cancun, Mexico (2009)
- [8] Nguyen, T.P., Debled-Rennesson, I. : A multi-scale approach to decompose a digital curves into meaningful parts. In : ICPR. (2010) to appear.
- [9] Nguyen, T.P., Debled-Rennesson, I. : Segmentation en arcs discrets en temps linéaire. In : RFIA. (2010)
- [10] Nguyen, T.P., Kerautret, B., Debled-Rennesson, I., Oliver-Lachaud, J. : Unsupervised, fast and precise recognition of digital arcs in noisy images. In : ICCVG. LNCS (2010) to appear.
- [11] Nguyen, T.P., Debled-Rennesson, I. : Decomposition of a curve into arcs and straight lines (2010) Rapport de recherche.
- [12] Nguyen, T.P., Debled-Rennesson, I. : Circularity measuring in linear time. In : ICPR. (2010) to appear.



# Bibliographie

- [ACH<sup>+</sup>91] Esther M. ARKIN, L. Paul CHEW, Daniel P. HUTTENLOCHER, Klara KEDEM et Joseph S. B. MITCHELL : An efficiently computable metric for comparing polygonal shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13(3):209–216, 1991.
- [adC76] Manfredo Perdigão do CARMO : *Differential geometry of curves and surfaces*. 1976.
- [AH05] Fabián ARREBOLA et Francisco Sandoval HERNÁNDEZ : Corner detection and curve segmentation by multiresolution chain-code linking. *Pattern Recognition*, 38(10):1596–1614, 2005.
- [AHU74] A.V. AHO, J.E. HOPCROFT et J.D. ULLMAN : *The Design and Analysis of Computer Algorithms*. 1974.
- [AK91] Hiroshi AOYAMA et Masahiro KAWAGOE : A piecewise linear approximation method preserving visual feature points of original figures. *CVGIP : Graph. Models Image Process.*, 53(5):435–446, 1991.
- [And94a] Eric ANDRÈS : Discrete circles, rings and spheres. *Computers & Graphics*, 18(5):695–706, 1994.
- [And94b] Eric ANDRÈS : Le plan discret. In *Colloque de géométrie discrète en imagerie : fondements et applications*, 1994.
- [AR93] Carlo ARCELLI et Giuliana RAMELLA : Finding contour-based abstractions of planar patterns. *Pattern Recognition*, 26(10):1563–1577, 1993.
- [Att54] E. ATTNEAVE : Some informational aspects of visual perception. *Psychol. Rev.*, 61(3):183–193, 1954.
- [AwH91] Nirwan ANSARI et Kuo wei HUANG : Non-parametric dominant point detection. *Pattern Recognition*, 24(9):849–862, 1991.
- [Bri08] Ernesto BRIBIESCA : An easy measure of compactness for 2d and 3d shapes. *Pattern Recognition*, 41:543–554, 2008.
- [Buz05] L. BUZER : An elementary algorithm for digital line recognition in the general case. In *DGCI*, volume 3429 de *LNCS*, pages 299–310, 2005.
- [Can86] J CANNY : A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, 1986.
- [CGRT04] David COEURJOLLY, Yan GÉRARD, Jean-Pierre REVEILLÈS et Laure TOUGNE : An elementary algorithm for digital arc segmentation. *Discrete Applied Mathematics*, 139(1-3):31–50, 2004.
- [Che02] Mu-Chen CHEN : Roundness measurements for discontinuous perimeters via machine visions. *Comput. Ind.*, 47(2):185–197, 2002.

- [CM91] Jean-Marc CHASSERY et Annick MONTANVERT : *Géométrie discrète en analyse d'images*. Hermès, 1991.
- [CMC07] David COEURJOLLY, Annick MONTANVERT et Jean-Marc CHASSERY : *Géométrie discrète et images numériques*. Traité IC2. Hermès, 2007.
- [CMT01] D. COEURJOLLY, S. MIGUET et L. TOUGNE : Discrete curvature based on osculating circle estimation. *In IWVF*, volume 2059 de *LNCS*, pages 303–312, 2001.
- [Coe02] David COEURJOLLY : *Algorithmique et géométrie discrète pour la caractérisation des courbes et des surfaces*. Thèse de doctorat, Université Lumière Lyon 2, 2002.
- [CR89] C.H.TEH et R.T.CHIN : On the detection of dominant points on the digital curves. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2:859–872, 1989.
- [Cro99] Terence M. CRONIN : A boundary concavity code to support dominant point detection. *Pattern Recognition Letters*, 20(6):617–634, 1999.
- [CS03] David COEURJOLLY et Stina SVENSSON : Estimation of curvature along curves with application to fibres in 3d images of paper. *In SCIA*, volume 2749 de *LNCS*, pages 247–254, 2003.
- [Dam95] Peter DAMASCHKE : The linear time recognition of digital arcs. *Pattern Recognition Letters*, 16(5):543–548, 1995.
- [Dav84] E.R DAVIES : A modified hough scheme for general circle location. *Pattern Recognition Letters*, 7:37–43, 1984.
- [DR95] I. DEBLED-RENNESON : *Reconnaissance des droites et plans discrets*. Thèse de doctorat, Université Louis Pasteur, 1995.
- [DR04] I. DEBLED-RENNESON : Estimation of tangents to a noisy discrete curve. *In Vision Geometry XII, SPIE*, volume 5300, pages 117–126, 2004.
- [DR07] I. DEBLED-RENNESON : *Éléments de géométrie discrète vers une étude des structures discrètes bruitées*, 2007. Habilitation à digérer des recherches, Université Nancy 1.
- [DRFRD05] Isabelle DEBLED-RENNESON, Fabien FESCHET et Jocelyne ROUYER-DEGLI : Optimal blurred segments decomposition in linear time. *In DGCI*, pages 371–382, 2005.
- [DRFRD06] Isabelle DEBLED-RENNESON, Fabien FESCHET et Jocelyne ROUYER-DEGLI : Optimal blurred segments decomposition of noisy shapes in linear time. *Computers & Graphics*, 30(1):30–36, 2006.
- [DRR95] Isabelle DEBLED-RENNESON et Jean-Pierre REVEILLÈS : A linear algorithm for segmentation of digital curves. *IJPRAI*, 9(4):635–662, 1995.
- [DRRRD03] Isabelle DEBLED-RENNESON, Jean-Luc REMY et Jocelyne ROUYER-DEGLI : Segmentation of discrete curves into fuzzy segments. *Electronic Notes in Discrete Mathematics*, 12:372–383, 2003.
- [DRTW04] Isabelle DEBLED-RENNESON, Salvatore TABBONE et Laurent WENDLING : Fast polygonal approximation of digital curves. *In ICPR*, volume 1, pages 465–468, 2004.
- [Ell02] Dave ELLIMAN : Tif2vec, an algorithm for arc segmentation in engineering drawings. *In GREC*, volume 2390 de *LNCS*, pages 350–358, London, UK, 2002. Springer-Verlag.



- 
- [Fed96] H. FEDERER : *Geometric Measure Theory*, Springer-Verlag. 1996.
- [Fes05] Fabien FESCHET : Fast guaranteed polygonal approximations of closed digital curves. In *SCIA*, volume 3540 de *LNCS*, pages 910–919, 2005.
- [FF08] Alexandre FAURE et Fabien FESCHET : Tangential cover for thick digital curves. In *DGCI*, volume 4992 de *LNCS*, pages 358–369, 2008.
- [Fis86] Steve FISK : Separating point sets by circles, and the recognition of digital disks. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-8(4): 554–556, July 1986.
- [FR06] Fabien FESCHET et Jean-Pierre REVEILLÈS : A generic approach for n-dimensional digital lines. In *DGCI*, volume 4245 de *Lecture Notes in Computer Science*, pages 29–40, 2006.
- [Fre61] H. FREEMAN : On the encoding of arbitrary geometric configurations. *IRE Trans. Electron. Comput.*, 10:260–268, 1961.
- [FT99] F. FESCHET et L. TOUGNE : Optimal time computation of the tangent of a discrete curve : Application to the curvature. In *DGCI*, volume 1568 de *LNCS*, pages 31–40, 1999.
- [GRE09] GREC09 : Arc segmentation contest, 2009. <http://www.cs.usm.my/arcseg2009/>.
- [HAA94] A. HELD, K. ABE et C. ARCELLI : Towards a hierarchical contour description via dominant point detection. *Systems, Man and Cybernetics, IEEE Transactions on*, 24(6):942–949, Jun 1994.
- [Har74] R.M. HARALICK : A measure for circularity of digital figures. *IEEE Trans. Systems, Man, and Cybernetics*, 4, 1974.
- [HS98] J.W. HARRIS et H. STOCKER : *Handbook of mathematics and computational science*, Springer-Verlag. 1998.
- [KA84] Chul E. KIM et Timothy A. ANDERSON : Digital disks and a digital compactness measure. In *STOC '84 : Proceedings of the sixteenth annual ACM symposium on Theory of computing*, pages 117–124, New York, NY, USA, 1984. ACM.
- [KBS75] Carolyn KIMME, Dana BALLARD et Jack SKLANSKY : Finding circles by an array of accumulators. *Short Communications Graphics and Image Processing*, 18(2):120–122, 1975.
- [KD82] Y. KUROZUMI et W.A. DAVIS : Polygonal approximation by the minimax method. *Computer Graphics and Image Processing*, 19(3):248–264, July 1982.
- [KdF88] Nasser D. KEHTARNAVAZ et Rui J. P. de FIGUEIREDO : A 3-d contour segmentation scheme based on curvature and torsion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 10(5):707–713, 1988.
- [KHB<sup>+</sup>94] Tapas KANUNGO, Robert M. HARALICK, Henry S. BAIRD, Werner STUETZLE et David MADIGAN : Document degradation models : Parameter estimation and model validation. In *MVA*, pages 552–557, 1994.
- [Kim84a] Chul E. KIM : Digital disks. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-6(3):372–374, May 1984.
- [Kim84b] Chul E. KIM : Three-dimensional digital planes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-6:639–645, 1984.

- [KL09a] Bertrand KERAUTRET et Jacques-Olivier LACHAUD : Curvature estimation along noisy digital contours by approximate global optimization. *Pattern Recognition*, 42(10):2265–2278, 2009.
- [KL09b] Bertrand KERAUTRET et Jacques-Olivier LACHAUD : Multi-scale analysis of discrete contours for unsupervised noise detection. In *IWCIA*, volume 5852 de *Lecture Notes in Computer Science*, pages 187–200, 2009.
- [KLN08] Bertrand KERAUTRET, Jacques-Olivier LACHAUD et Benoît NAEGEL : Comparison of discrete curvature estimators and application to corner detection. In *ISVC (1)*, volume 5358 de *LNCS*, pages 710–719, 2008.
- [Kol07] Alexander KOLESNIKOV : Optimal algorithm for lossy vector data compression. In *ICIAR*, volume 4633 de *LNCS*, pages 761–771, 2007.
- [Kov90] V.A. KOVALEVSKY : New definition and fast recognition of digital straight segments and arcs. In *ICPR*, volume 2, pages 31–34, Jun 1990.
- [KR82] Chul E. KIM et A ROSENFELD : Convex digital solids. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 4(6):612–618, 1982.
- [KR04] Reinhard KLETTE et Azriel ROSENFELD : Digital straightness - a review. *Discrete Applied Mathematics*, 139(1-3):197–230, 2004.
- [Lac06] J.-O. LACHAUD : *Espaces non-euclidiens et analyse d'image : modèles déformables riemanniens et discrets, topologie et géométrie discrète*. Habilitation à diriger des recherches, Université Bordeaux 1, Talence, France, 2006. (en français).
- [LaDGJLC04] Thomas LEWINER, Jo ao D. GOMES JR., Hélio LOPES et Marcos CRAIZER : Arc-length based curvature estimator. *Computer Graphics and Image Processing, Brazilian Symposium on*, 0:250–257, 2004.
- [LaDGJLC05] Thomas LEWINER, Jo ao D. GOMES JR., Hélio LOPES et Marcos CRAIZER : Curvature and torsion estimators based on parametric curve fitting. *Computers & Graphics*, 29(5):641–655, 2005.
- [LC99] Liyuan LI et Weinan CHEN : Corner detection and interpretation on planar curves using fuzzy reasoning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11):1204–1210, 1999.
- [Lea92] V. F. LEAVERS : The dynamic generalized hough transform : its relationship to the probabilistic hough transforms and an application to the concurrent detection of circles and ellipses. *CVGIP : Image Underst.*, 56(3):381–398, 1992.
- [LL99] Longin Jan LATECKI et Rolf LAKÄMPER : Polygon evolution by vertex deletion. In *Scale-Space*, pages 398–409, 1999.
- [LL00] L.J. LATECKI et R. LAKAMPER : Shape similarity measure based on correspondence of visual parts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(10):1185–1190, Oct 2000.
- [LL02] Longin Jan LATECKI et Rolf LAKÄMPER : Application of planar shape comparison to object retrieval in image databases. *Pattern Recognition*, 35(1):15–29, 2002.
- [LVdV05] J.-O. LACHAUD, A. VIALARD et F. de VIEILLEVILLE : Analysis and comparative evaluation of discrete tangent estimators. In *DGCI*, volume 3429 de *LNCS*, pages 240–251, 2005.
- [LVdV07] J.-O. LACHAUD, A. VIALARD et F. de VIEILLEVILLE : Fast, accurate and convergent tangent estimation on digital contours. *Image Vision Comput.*, 25(10):1572–1587, octobre 2007.

- 
- [Mar03] Majed MARJI : *On the detection of dominant points on digital planar curves*. Thèse de doctorat, Wayne State University, 2003.
- [Mas08] Asif MASOOD : Dominant point detection by reverse polygonization of digital curves. *Image Vision Comput.*, 26(5):702–715, 2008.
- [Meg84] Nimrod MEGIDDO : Linear programming in linear time when the dimension is fixed. *J. ACM*, 31(1):114–127, 1984.
- [Mel87] Avraham A. MELKMAN : On-line construction of the convex hull of a simple polyline. *Inf. Process. Lett.*, 25(1):11–12, 1987.
- [M.L74] Pitteway M.L.V. : Integer circles, etc. - some further thoughts. *Computer Graphics and Image Processing*, pages 262–265, 1974.
- [MM92] Farzin MOKHTARIAN et Alan K. MACKWORTH : A theory of multiscale, curvature-based shape representation for planar curves. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(8):789–805, 1992.
- [Mok97] Farzin MOKHTARIAN : A theory of multiscale, torsion-based shape representation for space curves. *Computer Vision and Image Understanding*, 68(1):1–17, 1997.
- [MS98] F. MOKHTARIAN et R. SOUMELA : Robust image corner detection through curvature scale space. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20:1376–1381, 1998.
- [MS03] Majed MARJI et Pepe SIY : A new algorithm for dominant points detection and polygonization of digital curves. *Pattern Recognition*, 36(10):2239–2251, 2003.
- [MS04] Majed MARJI et Pepe SIY : Polygonal representation of digital planar curves through dominant point detection - a nonparametric algorithm. *Pattern Recognition*, 37(11):2113–2130, 2004.
- [MWOS04] Ruben MEDINA, Andreas WAHLE, Mark E. OLSZEWSKI et Milan SONKA : Curvature and torsion estimation for coronary-artery motion analysis. In *SPIE Medical Imaging 2004 : Physiology, Function, and Structure from Medical Images*, volume 5369, pages 504–515, 2004.
- [NA84] Akira NAKAMURA et Kunio AIZAWA : Digital circles. *Computer Vision, Graphics, and Image Processing*, 26(2):242–255, 1984.
- [NDR07] Thanh Phuong NGUYEN et Isabelle DEBLED-RENNESON : Curvature estimation in noisy curves. In *CAIP*, volume 4673 de *LNCS*, pages 474–481, 2007.
- [NDR08a] Thanh Phuong NGUYEN et Isabelle DEBLED-RENNESON : Curvature and torsion estimators for 3d curves. In *ISVC (1)*, volume 5358 de *LNCS*, pages 688–699. Springer, 2008.
- [NDR08b] Thanh Phuong NGUYEN et Isabelle DEBLED-RENNESON : On the local properties of digital curves. *International Journal of Shape Modeling*, 14:105–125, 2008.
- [NDR09a] Thanh Phuong NGUYEN et Isabelle DEBLED-RENNESON : Fast and robust dominant point detection on digital curves. In *ICIP*, pages 953–956, Cairo, Egypt, November 2009.
- [NDR09b] Thanh Phuong NGUYEN et Isabelle DEBLED-RENNESON : Free-parameter method for polygonal representation of noisy curves. In *IWCIA*, Cancun, Mexico, November 2009.
- [NDR10a] Thanh Phuong NGUYEN et Isabelle DEBLED-RENNESON : Circularity measuring in linear time. In *ICPR*, 2010. to appear.

- [NDR10b] Thanh Phuong NGUYEN et Isabelle DEBLED-RENNESON : Decomposition of a curve into arcs and straight lines, 2010. Rapport de recherche.
- [NDR10c] Thanh Phuong NGUYEN et Isabelle DEBLED-RENNESON : A discrete geometry approach for dominant point detection. *Pattern Recognition*, 2010. to appear.
- [NDR10d] Thanh Phuong NGUYEN et Isabelle DEBLED-RENNESON : A linear method for segmentation of digital arcs, 2010. Rapport de recherche, <http://www.loria.fr/~nguyentp/pubs/ArcSegmentation.pdf>.
- [NDR10e] Thanh Phuong NGUYEN et Isabelle DEBLED-RENNESON : A multi-scale approach to decompose a digital curves into meaningful parts. *In ICPR*, 2010. to appear.
- [NDR10f] Thanh Phuong NGUYEN et Isabelle DEBLED-RENNESON : Segmentation en arcs discrets en temps linéaire. *In RFIA*, 2010.
- [Ngu05] Thanh Phuong NGUYEN : Courbure discrète et ses applications. Mémoire de D.E.A., IFI, 2005.
- [NKDROL10] Thanh Phuong NGUYEN, Bertrand KERAUTRET, Isabelle DEBLED-RENNESON et Jaques OLIVER-LACHAUD : Unsupervised, fast and precise recognition of digital arcs in noisy images. *In ICCVG, LNCS*, 2010. to appear.
- [Opr07] John OPREA : *Differential geometry and its applications*. 2007.
- [OvL81] M.H. OVERMARS et J. van LEEUWEN : Maintenance of configurations in the plane. *J. Comput. and Syst. Sci.*, 23:166–204, 1981.
- [PBDR06] Laurent PROVOT, Lilian BUZER et Isabelle DEBLED-RENNESON : Recognition of blurred pieces of discrete planes. *In DGCI*, volume 4245 de *Lecture Notes in Computer Science*, pages 65–76. Springer, 2006.
- [PCV94] Juan Carlos PÉREZ-CORTES et Enrique VIDAL : Optimum polygonal approximation of digitized curves. *Pattern Recognition Letters*, 15(8):743–750, 1994.
- [PD95] Arie PIKAZ et Its'hak DINSTEIN : Optimal polygonal approximation of digital curves. *Pattern Recognition*, 28(3):373–379, 1995.
- [PDR08] Laurent PROVOT et Isabelle DEBLED-RENNESON : Segmentation of noisy discrete surfaces. *In IWCIA*, volume 4958 de *Lecture Notes in Computer Science*, pages 160–171. Springer, 2008.
- [PGCMC05] Ángel Carmona POYATO, Nicolás Luis Fernández GARCÍA, Rafael Medina CARNICER et Francisco José MADRID-CUEVAS : Dominant point detection : A new proposal. *Image Vision Comput.*, 23(13):1226–1236, 2005.
- [PH95] Soo-Chang PEI et Ji-Hwei HORNG : Circular arc detection based on hough transform. *Pattern Recognition Letters*, 16(6):615–625, 1995.
- [Pro09] Laurent PROVOT : *Vers une polyérisation des objets discrets bruités 3D*. Thèse de doctorat, Université Nancy 1, 2009.
- [PS85] Franco P. PREPARATA et Michael Ian SHAMOS : *Computational Geometry - An Introduction*. Springer, 1985.
- [QG02] Azhar QUDDUS et Moncef GABBOUJ : Wavelet-based corner detection technique using optimal scale. *Pattern Recognition Letters*, 23(1-3):215–220, 2002.
- [Rev91] Jean-Pierre REVEILLÈS : Géométrie discrète, calculs en nombre entiers et algorithmique, 1991. Thèse d'état. Université Louis Pasteur, Strasbourg.

- 
- [RFQ03] A. A. RAD, K. FAEZ et N. QARAGOZLOU : Fast circle detection using gradient pair vectors. *In Digital Image Comp. : Techniques and Applications*, pages 879–887, 2003.
- [RJ73] A. ROSENFELD et E. JOHNSTON : Angle detection on digital curves. *IEEE Trans. Comput.*, 22:940–941, 1973.
- [Ros74a] A ROSENFELD : Compact figures in digital pictures. *IEEE Transactions on Systems, Man and Cybernetics*, SMC4:221–223, 1974.
- [Ros74b] A. ROSENFELD : Digital straight line segments. *Computers, IEEE Transactions on*, C-23(12):1264–1269, Dec. 1974.
- [Ros97] Paul L. ROSIN : Techniques for assessing polygonal approximations of curves. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(6):659–666, 1997.
- [RR92a] Bimal Kr. RAY et Kumar S. RAY : Detection of significant points and polygonal approximation of digitized curves. *Pattern Recognition Letters*, 13(6):443–452, 1992.
- [RR92b] Bimal Kumar RAY et Kumar S. RAY : An algorithm for polygonal approximation of digitized curves. *Pattern Recognition Letters*, 13(7):489–496, 1992.
- [RR93] Bimal Kumar RAY et Kumar S. RAY : Determination of optimal polygon from digital curve using  $L_1$  norm. *Pattern Recognition*, 26(4):505–509, 1993.
- [RScT09] Tristan ROUSSILLON, Isabelle SIVIGNON et Laure cu TOUGNE : On-line recognition of digital circular arcs. *In DGCI*, volume 5810 de LNCS. Springer, sep 2009.
- [RST08] Tristan ROUSSILLON, Isabelle SIVIGNON et Laure TOUGNE : Test and measure of circularity for digital curves. *In The 2008 International Conference on Image Processing, Computer Vision, and Pattern Recognition (ICCV'08 )*, juillet 2008.
- [RST10] Tristan ROUSSILLON, Isabelle SIVIGNON et Laure TOUGNE : Measure of circularity for parts of digital boundaries and its fast computation. *Pattern Recognition*, 43(1):37–46, 2010.
- [RW89] Paul L. ROSIN et Geoff A. W. WEST : Segmentation of edges into lines and arcs. *Image Vision Comput.*, 7(2):109–114, 1989.
- [SAM04] M. SARFRAZ, M.R. ASIM et A. MASOOD : Piecewise polygonal approximation of digital curves. *Information Visualisation, 2004. IV 2004. Proceedings. Eighth International Conference on*, pages 991–996, July 2004.
- [Sar93] Debranjian SARKAR : A simple algorithm for detection of significant vertices for polygonal approximation of chain-coded curves. *Pattern Recognition Letters*, 14(12):959–964, 1993.
- [Sau93] Peter SAUER : On the recognition of digital circles in linear time. *Comput. Geom. Theory Appl.*, 2(5):287–302, 1993.
- [SDRW06] J.-P. SALMON, Isabelle DEBLED-RENNESON et Laurent WENDLING : A new method to detect arcs and segments from curvature profiles. *In ICPR*, volume 3, pages 387–390. IEEE Computer Society, 2006.
- [SHYH84] T. Kasvand S.-H.-Y. HUNG : On the chord property and its equivalences. *In International Conference on Pattern Recognition*, pages 116–119, 1984.

- [SLC04] Jiqiang SONG, M.R. LYU et Shijie CAI : Effective multiresolution arc segmentation : algorithms and performance evaluation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(11):1491–1506, Nov. 2004.
- [SZ05] Mathew SHEARER et Ju Jia ZOU : Detection of dominant points based on noise suppression and error minimisation. In *ICITA '05 : Proceedings of the Third International Conference on Information Technology and Applications (ICITA'05) Volume 2*, pages 772–775, Washington, DC, USA, 2005. IEEE Computer Society.
- [U.R72] U.RAMER : An iterative procedure for the polygonal approximation of planar curves. *Computer, Graphic and Image processing*, 1:224–256, 1972.
- [Vee93] P. VEELAERT : On the flatness of digital hyperplanes. *Journal of Mathematical Imaging and Vision*, 3:205 – 221, 1993.
- [Via96] Anne VIALARD : Geometrical parameters extraction from discrete paths. In *DGCI*, volume 1176 de *LNCS*, pages 24–35, 1996.
- [WD98] L. WENYIN et D. DORI : Incremental arc segmentation algorithm and its evaluation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(4):424–431, Apr 1998.
- [Wit83] Andrew P. WITKIN : Scale-space filtering. In *IJCAI*, pages 1019–1022, 1983.
- [WS93a] Marcel WORRING et Arnold W. M. SMEULDERS : Digital curvature estimation. *CVGIP : Image Understanding*, 58:366 – 382, 1993.
- [WS93b] Marcel WORRING et Arnold W. M. SMEULDERS : Digital curvature estimation. *Computer Vision Graphics Image Processing : CVIU*, 58(3):366–382, 1993.
- [WS95] M. WORRING et A.W.M. SMEULDERS : Digitized circular arcs : characterization and parameter estimation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17(6):587–598, Jun 1995.
- [Wu03] Wen-Yen WU : An adaptive method for detecting dominant points. *Pattern Recognition*, 36(10):2231–2237, 2003.
- [Yin98] Peng-Yeng YIN : A new method for polygonal approximation using genetic algorithms. *Pattern Recogn. Lett.*, 19(11):1017–1026, 1998.
- [Yin00] Peng-Yeng YIN : A tabu search approach to polygonal approximation of digital curves. *IJPRAI*, 14(2):243–, 2000.
- [Yin03] Peng-Yeng YIN : Ant colony search algorithms for optimal polygonal approximation of plane curves. *Pattern Recognition*, 36(8):1783–1797, 2003.
- [ZHG06] Feng ZHAO, Qingming HUANG et Wen GAO : Image matching by multiscale oriented corner correlation. In *ACCV (1)*, volume 3851 de *LNCS*, pages 928–937, 2006.
- [ZHR10] J. ZUNIC, K. HIROTA et P.L. ROSIN : A Hu moment invariant as a shape circularity measure. *Pattern Recognition*, 43(1):47–57, January 2010.
- [ZLY+07] Xiaohong ZHANG, Ming LEI, Dan YANG, Yuzhu WANG et Litao MA : Multi-scale curvature product for robust image corner detection in curvature scale space. *Pattern Recognition Letters*, 28(5):545–554, 2007.

---

## Résumé

Dans cette thèse, nous nous intéressons à l'étude des courbes discrètes et ses applications en analyse d'images. Nous avons proposé une amélioration de l'estimation de courbure reposant sur le cercle circonscrit. Celle-ci repose sur la notion de segment flou maximal d'épaisseur  $\nu$  et sur la décomposition d'une courbe discrète en sa séquence de segments flous maximaux. Par la suite, nous avons appliqué cette idée en 3D afin d'estimer la courbure et la torsion discrète en chaque point d'une courbe 3D. Au niveau de l'application, nous avons développé une méthode rapide et fiable pour détecter les points dominants dans une courbe 2D. Un point dominant est un point dont la courbure est localement maximale. Les points dominants jouent un rôle très important dans la reconnaissance de formes. Notre méthode utilise un paramètre qui est l'épaisseur des segments flous maximaux. Reposant sur cette nouvelle méthode de détection des points dominants, nous avons développé des méthodes sans paramètres de détection des points dominants. Celles-ci se basent sur une approche multi-épaisseur. D'autre part, nous nous intéressons particulièrement au cercles et arcs discrets. Une méthode linéaire a été développée pour reconnaître des cercles et arcs discrets. Puis nous avons fait évoluer cette méthode afin de travailler avec des courbes bruitées en utilisant une méthode de détection du bruit. Nous proposons aussi une mesure de circularité. Une méthode linéaire qui utilise cette mesure a été aussi développée pour mesurer la circularité des courbes fermées. Par ailleurs, nous avons proposé une méthode rapide pour décomposer des courbes discrètes en arcs et en segments de droite.

**Mots-clés:** courbure discrète, torsion discrète, point dominant, cercle discret, circularité



## Abstract

In this thesis, we are interested in the study of discrete curves and its applications in image analysis. We have proposed an amelioration of curvature estimation based on circumcircle. This method is based on the notion of blurred segment of width  $\nu$  and on the decomposition of a curve into the sequence of maximal blurred segment of width  $\nu$ . Afterwards, we have applied this idea in 3D to estimate the discrete curvature and torsion at each point of a 3D curve. Concerning the applications, we have developed a rapid et reliable method to detect dominant points of a 2D curve. A dominant point is a point whose the curvature value is locally maximum. The dominant points play an important role in pattern recognition. Our method uses a parameter : the width of maximal blurred segments. Based on this novel method of dominant point detection, we proposed free-parameter methods for polygonal representation. They are based on a multi-width approach. Otherwise, we are interested in discrete arcs and circles. A linear method has been proposed for the recognition of arcs and circles. We then develop a new method for segmentation of noisy curves into arcs based on a method of noise detection. We also proposed a linear method to measure the circularity of closed curves. In addition, we have proposed a robust method to decompose a curve into arcs and line segments.

**Keywords:** discrete curvature, discrete torsion, corner point, discrete circle, circularity



