THESE

présentée par

Edouard Thiel

pour obtenir le titre de
DOCTEUR DE L'UNIVERSITÉ JOSEPH FOURIER - GRENOBLE I
(Arrêtés ministériels du 5.7.1984 et du 30.3.1992)

Spécialité : Mathématiques Appliquées

LES DISTANCES DE CHANFREIN EN ANALYSE D'IMAGES : FONDEMENTS ET APPLICATIONS

Date de soutenance : 21 **Septembre** 1994
Composition du Jury :

Pierre-Jean Laurent	President
Jean-Pierre Reveilles	Rapporteur
Gabriella Sanniti di Baja	Rapporteur
Bertrand Zavidovique	Rapporteur
Jean-Marc Chassery	Examinateur
Serge Miguet	Examinateur
Annick Montanvert	Examinateur

Thèse préparée au sein du Laboratoire **TIMC - Institut IMAG**

à Sylvie

Remerciements

Ce travail a été effectué au sein du laboratoire TIMC (Techniques de l'Imagerie, de la Modélisation et de la Cognition), de l'institut IMAG (Informatique et Mathématiques Appliquées de Grenoble).

Je remercie particulièrement Monsieur Pierre-Jean Laurent, professeur à l'Université Joseph Fourier de Grenoble, pour l'honneur qu'il m'a fait en acceptant de présider mon jury de thèse.

J'adresse mes plus vifs remerciements à Monsieur Bertrand Zavidovique, professeur à Orsay, qui a accepté la rude charge de rapporteur.

Je tiens à remercier Monsieur Jean-Pierre Reveillès, Maître de conférences à l'Université Louis Pasteur de Strasbourg, pour m'avoir donné le goût à la recherche, et m'avoir définitivement "discrétisé".

Un grand merci à Madame Gabriella Sanniti di Baja, Directeur de recherche au CNR de Naples, pour la richesse de notre collaboration et ses nombreuses qualités humaines.

Je veux exprimer ici ma profonde gratitude à Madame Annick Montanvert, professeur à l'École Normale Supérieure de Lyon et directrice de cette thèse, pour la confiance qu'elle m'a accordée, son dynamisme, et la disponibilité dont elle a toujours fait preuve pour me guider.

J'exprime toute ma reconnaissance à Monsieur Jean-Marc Chassery, Directeur de recherche au CNRS, et à Monsieur Serge Miguet, Maître de conférences à l'École Normale Supérieure de Lyon, qui ont accepté de participer au jury de cette thèse.

Je tiens enfin à remercier tous les membres du laboratoire pour l'aide qu'ils m'ont toujours prodiguée. Je réserve toutefois une attention particulière à Guy Bourrel, sans qui rien ne fonctionnerait, et à Etienne Bertin, pour ses conseils avisés.

Résumé

Les distances de chanfrein sont définies dans l'espace discret ; elles reposent sur la définition et l'application de masques de pondération, et permettent de bonnes approximations de la distance euclidienne réelle. Elles sont couramment utilisées en analyse d'images, pour quantifier ou décrire des régions dans une image. Elles permettent en particulier le calcul de squelettes pondérés, avec des algorithmes efficaces.

Notre propos est de compléter les connaissances sur ces distances à tous les niveaux, et de généraliser les notions et algorithmes.

Après quelques rappels de base, nous étudions les propriétés arithmétiques et géométriques des boules de chanfrein, de manière à déterminer les contraintes exactes pour qu'elles induisent bien une distance. Ces propriétés sont de plus à l'origine de formules de calcul direct. L'optimisation des masques est ensuite accomplie. Le but est de minimiser l'erreur commise par rapport à la distance euclidienne. Notre méthode est validée par l'obtention de nouveaux masques optimaux. Nous donnons un algorithme universel de calcul de l'axe médian, qui génère des tables de correspondance de façon très rapide. Nous proposons une méthode unifiée pour extraire le squelette pondéré d'une image de distance, calculée avec les distances discrètes les plus courantes. En dernier lieu nous présentons une méthode de description de formes, par la polygonalisation du squelette, qui ramène une forme discrète à une représentation vectorielle, conservant un certain degré de réversibilité.

MOTS CLÉS : analyse d'images, chanfrein, distance discrète, axe médian, ligne médiane, squelette, description de formes.

Abstract

Chamfer distances are defined in the discrete space; they rely on the definition and the application of weighted masks, and they provide good approximations of the real Euclidean distance. They are often used in image analysis, for quantification and to describe the regions constituting the shape. In particular they allow the computation of weighted skeletons, with efficient algorithms.

Our goal is to complete the knowledge above each level of these distances, and to generalise notions and algorithms.

After some recalls, we study arithmetical and geometrical properties of chamfer disks, in such a way that we are able to determine the exact constraints, so as to induce a distance. These properties give also direct computation formulas. The mask optimization is then accomplished. The aim is to minimize the error relative to the Euclidean distance. Our method is validated with new optimal masks obtaining. We provide a universal medial axis computation algorithm, which generates look-up tables in a very fast way. A unified distance-driven scheme is proposed to extract the weighted skeleton of a digital pattern, which runs whichever distance is selected among the most common ones. Finally we present a shape description method, where the skeleton is suitably decomposed by a polygonal approximation. Any digital shape is reduced to a vectorial representation, which keeps some degree of reversibility.

KEY WORDS : image analysis, chamfer, discrete distance, medial axis, medial line, skeleton, shape description.

Glossaire

$\mathbb{N}$, $\mathbb{Z}$, $\mathbb{R}$	Entiers naturels, entiers relatifs, réels
$\mathbb{E}$	Espace de travail
d_4, d_8	Distances classiques
d_C, $d_{3,4}$, $d_{5,7,11}$	Distances de chanfrein
d_E	Distance euclidienne
d_{oct}	Distance octogonale
X_4, C_8	Nombres de connexité
F_n	Suite de Farey
a, b, c	Points visibles
$\lozenge$	Point non affecté
(x_i, y_i, w_i)	Coordonnées et poids
Δ_p^q	Déterminant
dx, dy	Déplacements élémentaires
$\widehat{w_i, w_{i+1}}$	Cône
ε	Facteur d'échelle
τ_ε	Taux d'erreur
ε_{opt}, τ_{opt}	Facteur et taux optimaux
DM	Image de distances
AM	Axe médian
LM	Ligne médiane
S	Squelette
χ	Conducteur
$\dot{p}$	Représentant inférieur d'une classe d'équivalence
LUT	Table de correspondance
D_r, D_r^{-1}	Disque et disque inverse
$n_i(p)$	Un voisin de p

Table des matières

Table des figures

Chapitre 1

Introduction générale

CHANFREIN : nom masculin (ancien français *chant*, côté, et *fraindre*, briser). Surface oblique obtenue lorsque l'on abat l'arête d'une pierre, d'une pièce de bois ou de métal.

1.1 Analyse d'images

Depuis les années 1950, l'association de l'image et de l'ordinateur connaît un essor considérable, tant en ce qui concerne le domaine de l'analyse que celui de la synthèse d'images. L'analyse d'images et la vision par ordinateur s'attachent à construire une description explicite du contenu de l'image, tandis que la synthèse d'images part de la modélisation d'une scène pour construire une image cohérente.

L'imagerie couvre des champs d'application très variés, pour lesquels les capteurs, les objectifs de traitement et les conditions d'exécution peuvent être de natures très diverses. On peut citer les techniques de transmission de l'image (compression), l'imagerie biologique ou médicale, la robotique (détection, guidage, contrôle de qualité) qui impose généralement le temps réel et l'autonomie, l'imagerie satellitaire, qui est caractérisée par l'important volume de données à traiter, etc.

L'élément commun à tous ces domaines est l'image, qui intervient aussi bien du point de vue du stockage et de la visualisation de l'information que des traitements à réaliser. Ces aspects sont fortement liés à la donnée informatique, tant au niveau logiciel qu'au niveau des capteurs, des matériels d'affichage et de l'architecture des unités de traitement. Ces contraintes ont conduit à représenter l'image dans l'espace discret, puis à raisonner en restant dans cet espace : la géométrie discrète était née.

La géométrie euclidienne a une très grande influence sur notre intuition. Par exemple l'intersection de deux droites continues non parallèles est un point dans $\mathbb{R}^2$, alors que dans $\mathbb{Z}^2$ l'intersection de deux droites discrètes peut être vide, ou composée de plusieurs points, parfois répartis en sous-ensembles déconnectés [Rev91]. La synthèse d'images, ainsi que l'analyse d'images, utilisent encore aujourd'hui ces modèles continus, alors même qu'ils ne sont pas adaptés à la nature discrète des images (ni des processeurs).

La géométrie discrète a donc été développée ex nihilo, indépendamment de l'arsenal euclidien existant. Elle existe pour l'imagerie et pour elle-même. Elle fait partie des thèmes de nombreuses conférences internationales. Les travaux cités dans [Cha91], et les sujets qui sont exposés au colloque *Discrete Geometry for Computer Image*[1], attestent de sa grande richesse et de ses potentialités.

[1] Ce colloque annuel a lieu alternativement à Strasbourg et à Grenoble ; il est organisé par J.M. Chassery, J. Françon, A. Montanvert et J.P. Reveillès.

1.2 Distance de chanfrein

Les informations présentes dans une image sont au départ purement locales ; on dispose d'une collection de points, dont l'organisation globale nous échappe. Une des premières techniques visant à structurer l'information dans une image a été de doter les points d'une information de profondeur dans la forme. Ce concept est à la base de l'*image de distance*.

Malheureusement, la notion de 'distance' au sens où nous l'entendons habituellement, qui est la distance euclidienne $\sqrt{x^2+y^2}$, provient du monde continu, et n'est plus du tout adaptée à l'espace discret. Il est mathématiquement possible de construire d'autres fonctions de distance, qui donnent des résultats entiers, mais relativement proches de la distance euclidienne. Ce domaine est un véritable volet de la géométrie discrète, que l'on peut appeler la *géométrie des distances*.

Les familles de distances discrètes imaginables sont très variées, et il faut bien prendre en compte les contraintes liées au calcul de la carte de distance, puis de son utilisation. On a tout intérêt à se focaliser sur des distances qui se calculent localement dans l'image, qui demandent le moins d'arithmétique possible au niveau du processeur, qui ont des propriétés mathématiques intéressantes et qui permettent une approximation de qualité variable de la distance euclidienne.

Nous avons été totalement séduit par les *distances de chanfrein*. Il s'agit de distances basées sur un masque de pondérations, dont l'application locale permet le calcul de l'image de distance. Les schémas algorithmiques de base sont dus à Rosenfeld [Ros66], et leur grande efficacité est à l'origine des développements des distances de chanfrein, popularisées par Borgefors [Bor84, Bor86a].

Les applications développées autour des distances de chanfrein sont nombreuses et variées, en atteste la richesse de la littérature spécialisée. Quand il s'agit de faire de la mise en correspondance, de la morphométrie, de calculer des axes médians, lignes médianes et squelettes, d'en extraire des caractéristiques géométriques ou topologiques, l'option distance de chanfrein se révèle très performante.

Le sujet est situé à la rencontre de plusieurs domaines, dont l'arithmétique, l'optimisation, l'analyse et la description de formes. Il fait pleinement partie de la problématique de la géométrie discrète, et est de plus intéressant en soi.

1.3 Objectifs

Les distances de chanfrein ont, à n'en pas douter, de fortes potentialités encore inexploitées ; un état des lieux s'impose. Les connaissances mathématiques relatives aux distances de chanfrein sont très parcellaires. Il s'agit de formules au cas par cas, destinées à exprimer des propriétés constatées pour des petits masques, mais jamais généralisées. Il est impossible de montrer dans ces conditions si un masque induit une distance par exemple. Pour ce qui est de l'optimisation, c'est le phénomène inverse qui prédomine : il y a beaucoup de techniques publiées, mais aucun travail ne fait la part des choses. Enfin dans la partie des squelettes, c'est une approche globale qui fait défaut.

Notre objectif est d'extraire le plus possible de propriétés mathématiques, pour constituer une 'théorie des chanfreins'. Grâce à elle, nous tenterons d'adopter une démarche unificatrice, de clarifier les problèmes et d'apporter des solutions générales. Nous voulons mieux connaître les images de distance calculées avec les distances de chanfrein, pour appuyer les développements ultérieurs, théoriques et applicatifs.

1.4 Plan

Le document est organisé de la façon suivante.

Nous rentrons dans le vif du sujet au chapitre 2, où nous rappelons les éléments de la géométrie discrète qui seront utiles dans notre espace de travail : les images binaires 2D et le maillage carré. Cette étude bibliographique est ensuite centrée sur le domaine des distances discrètes. Le choix des distances de chanfrein est justifié par rapport aux autres distances classiques au niveau conceptuel, puis par l'efficacité des schémas algorithmiques relatifs au calcul des images de distances. Ces dernières jouent un rôle fondamental en analyse d'image, et nous en rappelons les principales applications.

La littérature est très pauvre en ce qui concerne les résultats théoriques pour les distances de chanfrein ; elle se contente de quelques formules extraites au cas par cas. Nous présentons au chapitre 3 notre contribution mathématique au domaine. La structure des masques et les propriétés relatives aux pondérations sont reliées à des théorèmes arithmétiques bien connus. De là on peut exprimer les conditions de distance et de norme en faisant ressortir des propriétés géométriques. L'intérêt des distances locales est le calcul à grande échelle sur toute une image ; mais certaines applications demandent aussi des formules de calcul direct de distance entre points, qui sont développées à la fin de ce chapitre, grâce aux nouveaux théorèmes.

Les distances de chanfrein ont pour but avoué d'approximer la distance euclidienne. Quelques auteurs ont développé leur propre approche, que nous analysons au chapitre 4. Les techniques employées vont du cas par cas au plus général, le choix du critère d'optimisation est mis en exergue, et les résultats contradictoires abondent. Notre approche est double ; il s'agit dans un premier temps de départager ou de relier les méthodes existantes, en mettant en avant les formules de passage. La seconde étape est de profiter de cette expérience pour développer une méthode plus simple et plus générale. De nouveaux masques optimaux sont ainsi proposés, et peuvent être comparés aux masques des autres chercheurs, grâce aux ponts établis dans la première étape.

A la suite de ces questions théoriques, est étudiée dans le chapitre 5 l'extraction de l'axe médian. Il s'agit d'une application fondamentale des images de distance, dont certaines propriétés permettent d'extraire un recouvrement de la forme par des tests uniquement locaux. Les méthodes existantes sont limitées à quelques distances, et font appel à des batteries de méthodes, très spécifiques et difficilement généralisables, sauf l'une d'entre elles qui est fondée sur des tables de correspondance. Nous proposons un algorithme très efficace de calcul de ces tables, qui fonctionne pour tous les masques de chanfrein ; il utilise une propriété liant les images de distance et les images inverses. La caractérisation de l'axe médian est indispensable au calcul du squelette pondéré.

Les squelettes sont une représentation de forme très employée en analyse d'images. Nous ne revenons pas au chapitre 6 sur les squelettes binaires, mais sur le calcul des squelettes pondérés, mené avec une distance de chanfrein. Les algorithmes existants font florès, mais sont généralement conçus pour une distance précise. En effet, les cartes de distance ont de nombreuses propriétés, soit générales aux distances, soit particulières, dues à des phénomènes arithmétiques. Or il est difficile de faire le tri entre ces propriétés si l'on n'adopte pas une démarche globale. C'est ainsi que nous proposons un algorithme multi-distance, qui est présenté pour les distances de chanfrein les plus courantes.

Le squelette pondéré a pour vocation la description de formes. On développe au chapitre 7 une représentation s'appuyant sur l'approximation polygonale du squelette pondéré. La forme est ramenée à un codage vectoriel, quasi-réversible, qui permet de s'affranchir de l'espace de l'image, pour raisonner en terme de la géométrie des régions liées aux segments de la polygonalisation. Une étude est faite pour simplifier la représentation, en annihilant ou en fusionnant certaines régions élémentaires. L'objectif est d'aboutir à une représentation plus proche de l'intuition humaine, pour faciliter la description de la forme.

L'analyse d'images est un domaine où théorie et expérimentation font bon ménage. Nous décrivons au chapitre 8 l'environnement logiciel et les applications développées durant la thèse. L'interface de traitement d'images *Image Package Software* a été développée et diffusée pour faciliter l'interaction entre les disciplines liées à l'imagerie. Des programmes de calcul de squelette ont été écrits sur différentes plates-formes. Le suivi des différentes applications a révélé l'efficacité des algorithmes, et élargi encore le champ d'application des squelettes.

Une conclusion générale clôt ce travail au chapitre 9.

Chapitre 2

Géométrie discrète

Nous présentons dans ce chapitre les distances de chanfrein dans le contexte de la géométrie discrète.

Parmi les espaces de travail habituels, nous orientons notre discours vers le plan et le maillage carré, dont les propriétés topologiques de voisinages et connexités sont rappelées, ainsi que le théorème de Jordan discret.

Les distances discrètes et les images de distance sont définies, puis nous revenons sur la bibliographie et les performances de chacune des principales distances discrètes de l'analyse d'images.

Le choix des distances de chanfrein est justifié par un bon compromis entre l'approximation de la distance euclidienne, la facilité d'utilisation, et l'efficacité des schémas algorithmiques.

2.1 Espace de travail

L'analyse d'images par ordinateur doit travailler sur des données discrètes, de support fini. La représentation de base est l'*image discrète*, dont le support est associé à un *maillage*, qui précise l'arrangement des points entre eux.

L'échantillonnage de la scène est fourni par un capteur, dont les cellules sont le plus souvent disposées régulièrement. On a donc tout intérêt à se placer dans l'espace de l'acquisition, où le maillage est régulier.

Les trois maillages réguliers dans le plan [Cha91] sont le maillage carré, hexagonal et triangulaire. Les deux premiers sont représentés figure 1 ; quant au dernier, il est inusité en pratique.

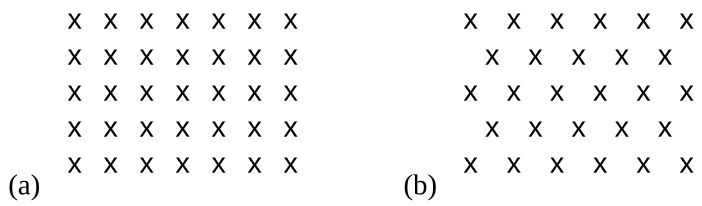


FIG. 1 - Maillage carré (a) et hexagonal (b)

Le **maillage carré** est bien sûr le plus commun, les capteurs ainsi que les écrans d'ordinateur l'utilisent. Le support est le réseau fondamental de $\mathbb{Z}^2$, qui est le plus aisé à manipuler. Il faut cependant tenir compte des types de connexités, avec 4 ou 8 voisins, directs ou indirects, et d'une dualité dans le théorème de Jordan (§2.2).

On parle de **maillage rectangulaire** pour certains capteurs qui n'ont pas la même résolution ligne/colonne (typiquement 4/3) ; plutôt que de procéder à un rééchantillonnage carré, on conserve parfois inchangée cette représentation. En fait le support de l'image est encore $\mathbb{Z}^2$, mais on tient compte du rapport de résolution (par exemple en morphométrie). Des problèmes théoriques en découlent. On étudie les distances de chanfrein en maillage rectangulaire au §3.6.

Les écrans de télévision ont utilisé le **maillage hexagonal** pour améliorer le rendu de l'image (entrelacement, antialiasing, densité des points). Les capteurs hexagonaux sont cependant très rares. Les images sont donc souvent rééchantillonnées depuis le maillage carré. De même les écrans d'ordinateurs sont très rarement hexagonaux, et le rendu est obtenu en simulant une grille hexagonale sur l'écran. Malgré ces artifices, l'intérêt pour le maillage hexagonal n'a jamais décru. En effet, la grille hexagonale est une meilleure approximation des récepteurs de la rétine que la grille carrée. Au niveau morphologique, il peut être intéressant de travailler avec six voisins directs, et la même connectivité pour le fond et la forme. Enfin, les architectures massivement parallèles sont plus facilement implémentables [Ste85]. Mais les résultats et notions peuvent passer facilement d'une représentation à l'autre.

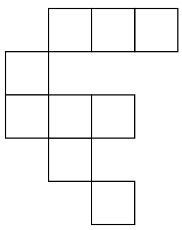
En **trois dimensions**, des applications importantes apparaissent par exemple avec les capteurs qui procèdent par coupes. On distingue l'acquisition destructive, puis empilage des données, inférence [Mon93] ou mise en correspondance [Rol91], de l'acquisition non destructive, par exemple avec le microscope confocal [Paw90, Par93, Uss94]. Le volume de données à traiter consécutivement est gigantesque, va souvent de pair avec des voxels parallélépipédiques, et est malcommode à visualiser [Tro87]. La question des réseaux 3D généralisant la maille hexagonale est traitée dans [Gra93].

Nous notons $\mathbb{E}$ notre espace de travail. Dans tout cet ouvrage, $\mathbb{E}$ est le **plan** $\mathbb{Z}^2$ et le maillage **carré**. Nous nous attachons dans la suite à y développer les questions théoriques et les applications. Leur passage aux autres maillages et dimensions est souvent possible.

2.2 Voisinages et connexité

Nous rappelons dans cette partie les notions de topologie discrète qui sont utiles dans la suite.

Une *image binaire* de taille $N \times M$ à valeurs dans $[0, 1]$ est codée par une matrice d'entiers de même taille. Elle contient des *objets*, qui sont des ensembles connexes de points étiquetés à 1, le fond étant à 0 (figure 2).



$$\longrightarrow \quad \begin{matrix} 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{matrix}$$

FIG. 2 - Représentation d'une image discrète

Un point p de l'image est défini par ses coordonnées cartésiennes (abscisse, ordonnée). Le point p a 4 voisins directs et 4 voisins indirects (figure 3).

FIG. 3 - Voisins directs (a) et indirects (b)

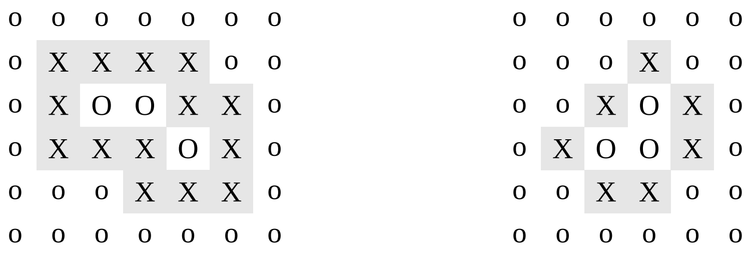
On définit donc le *4-voisinage* et le *8-voisinage*, en considérant respectivement les voisins directs, ou directs et indirects. Ces notions sont si intimement liées aux distances d_4 et d_8 (§2.3.1) qu'on ne sait jamais dans quel ordre les présenter !

Un *chemin* de p_0 à p_k est une suite de points p_0, p_1, ..., p_k telle que p_i est voisin de p_{i-1} pour $1 \leq i \leq k$. On dit que le chemin est *n-connexe* ($n = 4$ ou 8) selon le type de voisinage, ou $*$*-connexe* si le type n'est pas établi.

Un *arc* est un chemin tel que chaque point a exactement 2 voisins, sauf les extrémités qui n'en ont qu'un. Enfin une *courbe* est un arc fermé.

Pour compléter le vocabulaire, deux points voisins sont dits *adjacents*, et deux points reliés par un chemin sont *connectés*.

Les définitions d'objet, d'arc et de courbe sont toutes à prendre pour un type de connexité fixé.



Courbe 4-connexe et intérieur 8-connexe

Courbe 8-connexe et intérieur 4-connexe

FIG. 4 - Dualité des connexités

Dans la figure 4, on constate que le théorème de Jordan n'est pas vérifié si l'on considère la même connexité pour le fond et l'objet. Ces problèmes sont simplement résolus [Kon89] si le complémentaire d'un objet n-connexe ($n = 4, 8$) est analysé en $(12 - n)$ connexité. Le théorème de Jordan se reformule alors :

Théorème 2.1 (Jordan discret) *Le complémentaire de toute courbe discrète 4-connexe (respectivement 8-connexe) est formé de deux composantes 8-connexes (respectivement 4-connexes) : l'intérieur et l'extérieur de la courbe.*

Les *nombres de connexité* suivants donnent en un point p le nombre de composantes connexes formées par son 8-voisinage en 'tournant' autour du point. Ils sont donc compris entre 0 et 4. Soit $B = \{b_i \in [0,1]\,,\ i = 0..7\}$ l'ensemble des points constituant le 8-voisinage de p, numérotés dans l'ordre

$$\begin{array}{ccc} b_3 & b_2 & b_1 \\ b_4 & p & b_0 \\ b_5 & b_6 & b_7 \end{array}$$

et soit $\overline{b}_i = 1 - b_i$. Les indices sont à prendre modulo 8.

Le '*crossing number*' X_4 est le nombre de composantes 4-connexes [Rut66]

$$X_4(B) = \frac{1}{2} \sum_{k=0}^{7} |b_{k+1} - b_k|$$

Le '*connectivity number*' C_8 est le nombre de composantes 8-connexes [Yok75]

$$C_8(B) = b_0 b_2 b_4 b_6 + \sum_{k=0}^{3} \left(\overline{b}_{2k} - \overline{b}_{2k} \overline{b}_{2k+1} \overline{b}_{2k+2} \right)$$

(si le premier terme $b_0 b_2 b_4 b_6$ vaut 1, alors p est *4-interne*, et le second terme vaut 0 ; on néglige parfois le calcul du premier terme).

Ces nombres X_4 et C_8 trouvent leur utilité dans la conservation de la connexité pour le calcul de squelettes et l'extraction des points selle (chapitre 6).

2.3 Distances discrètes

La notion de distance est utile pour quantifier et décrire les objets présents dans une image. La volonté de rester dans l'espace de l'image impose l'utilisation de distances spécifiques, qui fournissent des résultats exclusivement entiers. Après les définitions d'usage, nous faisons un inventaire et un comparatif des principales distances discrètes utilisées en analyse d'images.

2.3.1 Définitions

Définition 2.1 (Distance discrète) *On appelle distance discrète sur un espace* $\mathbb{E}$ *une application* $d : \mathbb{E} \times \mathbb{E} \to \mathbb{N}$ *vérifiant :* $\forall\ A,\ B,\ C \in \mathbb{E}$

1. $d(A, B) \geq 0$; $d(A, B) = 0 \ \Leftrightarrow\ A = B$ — *définie positive*
2. $d(A, B) = d(B, A)$ — *symétrie*
3. $d(A, B) \leq d(A, C) + d(C, B)$ — *inégalité triangulaire*

Les normes usuelles pour $(\xi_1, \ldots, \xi_n)$ dans $\mathbb{R}^n$

$$d_1 = \sum_{i=1}^{n} |\xi_i| \qquad d_2 = \sqrt{\sum_{i=1}^{n} |\xi_i|^2} \qquad d_\infty = \sup_i |\xi_i|$$

vont devenir respectivement $d_4 \geq d_E \geq d_8$ dans $\mathbb{Z}^2$, en l'occurrence

- $d_4(A, B) = |x_b - x_a| + |y_b - y_a|$ — *city block, Manhattan, diamond*
- $d_8(A, B) = \max\left(|x_b - x_a|\,,\,|y_b - y_a|\right)$ — *chessboard, square, échiquier*
- $d_E(A, B) = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2}$ — *distance euclidienne*

pour deux points $A(x_a, y_a)$ et $B(x_b, y_b)$.

Les distances discrètes d_4 et d_8 sont les premières à avoir été employées dans les images de distance, car elles sont simplissimes à calculer, tandis que d_E n'est pas une distance discrète (§2.3.3).

Les distances d_4 et d_8 tirent leur notation de leur disque unité (figure 5).

$$\begin{array}{ccc} & 1 & \\ 1 & 0 & 1 \\ & 1 & \end{array} \qquad\qquad \begin{array}{ccc} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{array}$$

(a) (b)

FIG. 5 - Disques unité de (a) d_4 et (b) d_8

Définition 2.2 (Image de distance) *Etant donné un ensemble* X *dans un espace métrique* $(\mathbb{E},\ d)$*, on appelle image (ou carte) de distance l'image notée* DM_X *telle que la valeur attribuée en tout point* p *est égale à la distance de* p *au complémentaire de* X *:*

$$DM_X \left(\begin{array}{ccl} \mathbb{E} & \longrightarrow & \mathbb{N} \\ p & \longmapsto & d(p, \overline{X}) = \inf\{d(p, q)\,,\ q \in \overline{X}\} \end{array} \right.$$

Les cartes de distance de d_4 et d_8 présentent des courbes de niveaux caractéristiques (figure 6). En effet ces distances sont relatives à la connexité entre points d'une image.

			1	1	1			1	1	1	1	1
	1	1	2	2	2	1	1	2	2	2	2	1
1	2	2	3	3	3	2	2	3	3	3	2	1
1	2	3	4	4	4	3	3	4	4	3	2	1
1	2	3	4	5	5	4	4	5	4	3	2	1
1	2	3	4	5	4	4	3	4	4	3	2	1
1	2	3	4	4	3	3	2	3	3	2	1	
1	2	3	3	3	2	2	1	2	2	2	2	1
	1	2	2	2	1	1		1	1	1	2	1
		1	1	1							1	1

(a)

			1	1	1			1	1	1	1	1
	1	1	1	2	1	1	1	1	2	2	2	1
1	1	2	2	2	2	2	2	2	2	3	2	1
1	2	2	3	3	3	3	3	3	3	3	2	1
1	2	3	3	4	4	4	4	4	3	2	2	1
1	2	3	3	3	3	3	3	3	3	2	1	1
1	2	2	3	3	2	2	2	2	2	2	1	
1	1	2	2	2	2	1	1	1	2	2	1	1
	1	1	2	1	1	1		1	1	1	1	1
		1	1	1							1	1

(b)

FIG. 6 - DM pour (a) d_4 et (b) d_8

Les boules de d_4 et d_8 n'étant pas circulaires (§2.6), les valeurs obtenues sont sensiblement différentes (chapitre 4) de celles associées à d_E, d'où une anisotropie, et une non-robustesse à la rotation de l'image des traitements fondés sur d_4 et d_8.

Les images de distance sont utilisées dans de nombreuses applications. En général, les résultats seront d'autant meilleurs que la distance employée approximera correctement d_E. Mais une 'meilleure' distance requiert souvent plus de calculs ou de mémoire. Il est donc intéressant de disposer de distances variées.

L'idée centrale de la plupart des distances discrètes consiste à approcher d_E par propagation de *distances locales*[1] [Ros66]. La raison en est l'efficacité de calcul, car pour étiqueter un point, seul un petit voisinage doit être consulté.

Un historique complet des distances digitales est réalisé dans [Mel91]. Les alternatives à d_4 et d_8 que l'on rencontre le plus souvent en analyse d'images sont les distances octogonales (§2.3.2), la distance euclidienne adaptée au discret (§2.3.3) et les distances de chanfrein (§2.3.4). D'autres possibilités existent dans la littérature, voir par exemple [Yam84b, Das89, Das90c, Cha92] mais elles sont plus anecdotiques.

2.3.2 Distance octogonale

Le concept de *distance octogonale* fut introduit par Rosenfeld et Pfaltz dans [Ros68]. Ils montrèrent que l'emploi alterné de d_4 et d_8 sur chaque courbe de niveaux

[1] Coût associé à un déplacement.

définit une nouvelle distance, notée d_{oct}, telle que $d_4 \geq d_{oct} \geq d_8$. La boule est un octogone, de côtés perpendiculaires aux axes et aux bissectrices.

Les distances octogonales sont *généralisées* par Das dans [Das90b], à des cycles de longueur arbitraire alternant d_4 et d_8. La séquence (d_4, d_8) correspond à la distance octogonale primitive, que l'on note encore d_{oct} (1/2, 1/2). La séquence (d_4, d_8, d_4) définit d_{oct} (2/3, 1/3), qui est proposée dans [Das90b]. On compare figure 7 les cartes de distance obtenues. On trouve encore d'autres résultats dans [Das90a, Das91].

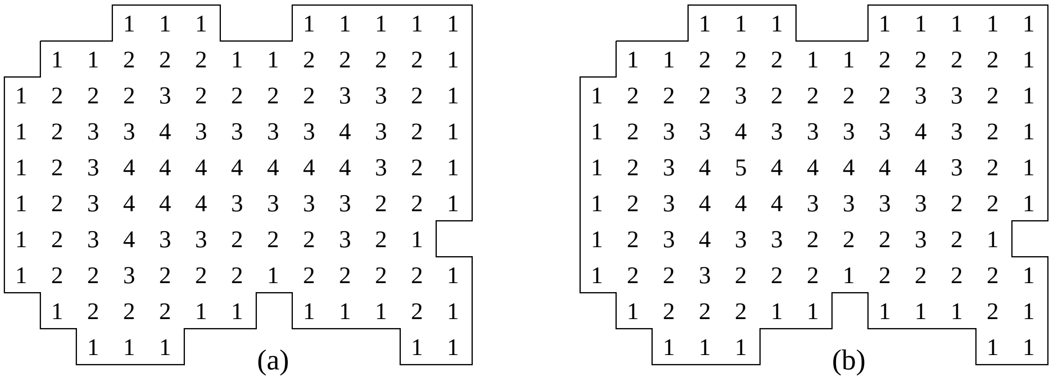


FIG. 7 - *DM* pour (a) d_{oct} (1/2, 1/2) et (b) d_{oct} (2/3, 1/3)

Les distances *hyperoctogonales* sont la généralisation naturelle à la dimension 3 [Dan93a]. Ce sont des cycles alternés de d_6, d_{18} et d_{26}. Danielsson propose ainsi d_{oct} (3/5, 1/5, 1/5).

Pour toutes ces distances, Ragnemalm donne un algorithme de calcul de la distance au fond, en 3 passages séquentiels sur l'image en 2D, et en 4 passages pour le 3D [Rag92]. Cependant un compteur de modulo doit être stocké pour chaque point, et le nombre de passages est plus grand que pour l'algorithme de Rosenfeld (§2.4).

Selon Danielsson, les distances octogonales présentent un intérêt pour les opérations morphologiques sur les images binaires, et permettent d'approximer correctement la distance euclidienne.

On peut minimiser très finement l'aire signée, comprise entre la boule et le disque euclidien, par le choix de la séquence. Mais dans le plan, les boules sont encore des octogones, et le taux d'erreur maximale par rapport à d_E reste élevé. Autrement dit, les d_{oct} généralisées apportent un progrès *figé* par rapport aux distances de base, et ne sont ni simples à manier ni à étudier.

Finalement, les distances octogonales restent marginales en analyse d'images.

2.3.3 Distance euclidienne

Utiliser la distance euclidienne serait bien entendu la solution idéale, mais comme il était déjà souligné dans [Ros68], ni $(d_E)^2$, $\mathrm{int}(d_E)$, ou $\mathrm{trunc}(d_E)$ ne sont des distances. Ces fonctions ne respectent pas l'inégalité triangulaire, en particulier pour les petites valeurs. Cela n'est pas sans conséquences dans les applications. Par exemple, Forchhammer montre dans [For89] que la reconstruction de squelettes pondérés donne de faux résultats, contrairement aux fonctions de distance purement discrètes, vues dans cette section 2.3.

On est donc contraint de changer de système de représentation, en stockant des vecteurs au lieu de distances. Ces vecteurs indiquent le point le plus proche du fond, et peuvent être signés, ou stockés en valeurs absolues. On désigne par *image euclidienne* le couple d'images de coordonnées. La figure 8 montre un exemple de couple d'images de coordonnées (non signées).

			1	0	0			1	0	0	0	0
	1	0	1	0	1	0	0	1	0	0	0	1
1	1	0	1	2	1	0	0	1	2	0	2	1
1	2	2	1	2	1	0	0	1	2	3	2	1
1	2	3	4	2	1	0	0	1	3	2	2	1
1	2	3	4	1	2	1	0	1	3	2	1	0
1	2	2	3	1	2	1	0	1	2	2	1	
1	1	2	2	1	0	1	0	1	0	0	1	0
	1	1	0	1	0	0		1	0	0	1	1
		1	0	0							1	0

(a)

			0	1	1			0	1	1	1	1
	0	1	1	2	1	1	1	1	2	2	2	0
0	1	2	2	2	2	2	2	2	2	3	0	0
0	0	2	3	3	3	3	3	3	3	0	0	0
0	0	0	0	4	4	4	4	4	2	2	0	0
0	0	0	0	4	3	3	3	3	1	1	1	1
0	0	2	2	3	2	2	2	2	2	0	0	
0	1	1	2	2	2	1	1	1	2	2	1	1
	0	1	2	1	1	1		0	1	1	1	0
		0	1	1							0	1

(b)

FIG. 8 - DM pour (a) $d_E : x$ et (b) $d_E : y$

Différents algorithmes de calcul de l'image des distances au fond ont été publiés. Celui de Danielsson [Dan80] opère en 4 passages séquentiels sur l'image, mais produit des erreurs dans certaines configurations, et n'est pas très rapide. Une version modifiée par Ye [Ye88] permet le calcul de l'image signée. De son côté Yamada [Yam84a] a donné un algorithme parallèle, qui dépend donc de la taille des objets dans l'image et est assez lent. Le résultat est quasi-exact, les erreurs étant très rares [For89]. Forchhammer présente un calcul de l'image euclidienne à partir d'une image de chanfrein avec une table de correspondance [For89], qui est exact pour des petites valeurs (inférieures à $\sqrt{17}$ à partir de chanfrein 3,4 et $\sqrt{104}$ pour chanfrein 19,27,42). La transformation de distance euclidienne est approfondie par Ragnemalm dans [Rag89, Rag90b, Rag90a]. Il propose un algorithme séquentiel en 3 passes sur l'image, avec un masque éclaté en 3 parties. Le procédé est exact et rapide. La méthode la plus rapide consiste à faire évoluer une chaîne du contour vers l'intérieur des objets [Vin91] ; le résultat est exact, mais requiert une grosse structure de données.

L'isotropie des propriétés et la fiabilité des mesures sont importantes en analyse d'images. Cela explique le volume de travaux autour de la distance euclidienne. Mais son emploi présente un certain nombre de désagréments, dont l'absence de définition *locale* est une cause. Que l'on stocke l'information de distance sous la forme vectorielle (x, y), par le carré x^2+y^2, voire même en réel $\sqrt{x^2+y^2}$, les transformations de distance que l'on a passées en revue sont complexes et relativement coûteuses. On souligne de plus le problème de l'utilisation de telles cartes de distance, en citant la difficulté à discriminer les centres des boules maximales (§5.4.2).

Il peut donc être judicieux de se placer dans l'espace discret, et de construire des distances qui réalisent un bon compromis entre efficacité et maniabilité.

2.3.4 Distances de chanfrein

La définition de d_4 et d_8 revient à associer un coût de 1 aux déplacements directs ou diagonaux (figure 9.a,b). Il semble naturel de donner des poids plus significatifs du déplacement euclidien réalisé, comme à la figure 9.c avec $(1, \sqrt{2})$. Mais cela n'est pas suffisant, car si l'erreur est annulée suivant les directions multiples de 45°, elle prend des valeurs importantes ailleurs.

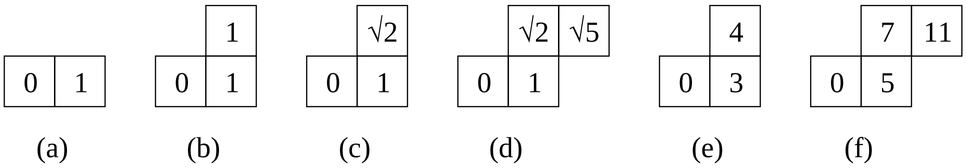


FIG. 9 - Pondérations locales

Montanari a ainsi eu l'idée de pondérer d'autres déplacements dans des voisinages plus grands [Mon68], avec par exemple $(1, \sqrt{2}, \sqrt{5})$ figure 9.d. La distance *rationnelle* qu'il obtient en prenant le chemin de longueur minimale d'un point à un autre, formé des déplacements autorisés, peut en théorie approcher d_E autant qu'on le souhaite. Comme ces distances ne sont pas entières, Hilditch a le premier utilisé les poids (2, 3) dans [Hil69], ce qui revient à approcher $(1, \sqrt{2})$ par $(1, 3/2)$. On trouve une apparition ultérieure de (2, 3) dans [Bar77], pour de la mise en correspondance.

C'est Borgefors qui popularise ces *distances de chanfrein* dans deux papiers célèbres [Bor84] et [Bor86a]. Elle préconise et justifie l'approximation de $(1, \sqrt{2})$ par $(1, 4/3)$ et celle de $(1, \sqrt{2}, \sqrt{5})$ par $(1, 7/5, 11/5)$ (figure 9.e,f). Les distances résultantes sont le chanfrein (3,4) et le chanfrein (5,7,11), notés $d_{3,4}$ et $d_{5,7,11}$. Des cartes de distance les utilisant sont montrées figure 10.

			3	3	3			3	3	3	3	3
	3	3	4	6	4	3	3	4	6	6	6	3
3	4	6	7	8	7	6	6	7	8	9	6	3
3	6	8	10	11	10	9	9	10	11	9	6	3
3	6	9	12	14	13	12	12	13	11	8	6	3
3	6	9	12	12	11	10	9	10	10	7	4	3
3	6	8	11	10	8	7	6	7	8	6	3	
3	4	7	8	7	6	4	3	4	6	6	4	3
	3	4	6	4	3	3		3	3	3	4	3
		3	3	3							3	3

(a)

			5	5	5			5	5	5	5	5
	5	5	7	10	7	5	5	7	10	10	10	5
5	7	10	11	14	11	10	10	11	14	15	10	5
5	10	14	16	18	16	15	15	16	18	15	10	5
5	10	15	20	22	21	20	20	21	18	14	10	5
5	10	15	20	21	18	16	15	16	16	11	7	5
5	10	14	18	16	14	11	10	11	14	10	5	
5	7	11	14	11	10	7	5	7	10	10	7	5
	5	7	10	7	5	5		5	5	5	7	5
		5	5	5							5	5

(b)

FIG. 10 - DM pour (a) $d_{3,4}$ et (b) $d_{5,7,11}$

Le principe des distances de chanfrein, notées d_C, est de pondérer les déplacements dans un voisinage donné avec des entiers, puis de fixer la distance entre tout couple de points, au coût du chemin de coût minimal les rejoignant, formé des déplacements autorisés. L'ensemble des pondérations affectées aux déplacements autorisés est appelé le *masque*.

A ce niveau, aucune définition explicite n'est donnée, mais cela n'est pas gênant dans la mesure où les distances de chanfrein sont conçues pour le calcul global d'images de distances par propagation de distances locales. Des formules explicites seront données au §3.5.2, suite aux développements théoriques du chapitre 3.

Plus généralement, on se donne la possibilité de travailler sur le maillage carré avec des masques entiers de taille $m \times m$ quelconque, symétriques par rapport à l'origine. Ces masques sont définis au §3.2. Pour les dimensions supérieures, voir [Bor84, Bor86a], et pour le maillage hexagonal, voir [Bor88].

Le choix des pondérations et de leur emplacement est primordial. Nous définissons au chapitre 3 les conditions strictes pour que ces masques induisent bien des métriques discrètes. Un choix judicieux permet alors de bonnes approximations de la distance euclidienne (chapitre 4).

Ces distances sont conçues dès le départ dans l'espace discret, et jouissent de propriétés élégantes, que nous ferons apparaître dans ce travail. Mais la caractéristique la plus importante, qui justifie grandement notre intérêt pour les distances de chanfrein, réside dans l'efficacité des algorithmes de transformation de distance.

2.4 Schémas algorithmiques

Une transformation de distance convertit une image binaire en carte de distance DM. Le calcul des distances est en principe une tâche globale. La méthode exhaustive, qui pour chaque point teste toute l'image, est excessivement coûteuse. Heureusement il est possible de restreindre les tests à de petits voisinages, puisque les distances de chanfrein sont basées sur la propagation de distances locales. Les schémas algorithmiques de balayage de l'image peuvent être parallèle ou séquentiel.

Le calcul séquentiel de DM fut publié en 1966 [Ros66] et le parallèle en 1968 [Ros68]. L'équivalence entre les deux méthodes y est montrée. Mais au chapitre 3 on verra un contre-exemple, qui indique que l'équivalence est effective uniquement lorsque le masque de chanfrein induit une distance.

2.4.1 Calcul parallèle

La première approche est mise en œuvre par une propagation sur les contours de l'objet. On initialise le bord de l'objet à 1, puis à chaque étape on étiquette le contour de l'objet formé par les points non traités (figure 11). Cet algorithme s'apparente à un calcul par pelage pour d_4 et d_8, et s'écrit très facilement.

$$
\begin{array}{cccccc}
1 & 1 & 1 & 1 & 1 & 1 \\
1 & * & * & * & * & 1 \\
1 & * & * & * & * & 1 \\
1 & 1 & * & * & 1 & \\
 & & 1 & 1 & 1 &
\end{array}
\longrightarrow
\begin{array}{cccccc}
1 & 1 & 1 & 1 & 1 & 1 \\
1 & 2 & 2 & 2 & 2 & 1 \\
1 & 2 & * & * & 2 & 1 \\
1 & 1 & 2 & 2 & 1 & \\
 & & 1 & 1 & 1 &
\end{array}
\longrightarrow
\begin{array}{cccccc}
1 & 1 & 1 & 1 & 1 & 1 \\
1 & 2 & 2 & 2 & 2 & 1 \\
1 & 2 & 3 & 3 & 2 & 1 \\
1 & 1 & 2 & 2 & 1 & \\
 & & 1 & 1 & 1 &
\end{array}
$$

FIG. 11 - Parcours parallèle avec d_4

Dans le cas plus général des chanfreins, on procède avec des masques, par exemple ceux de la figure 12. On place le centre du masque sur chaque pixel, et la distance locale de chaque point du masque est ajoutée à la valeur du point de l'image correspondant. La nouvelle valeur du pixel central est le minimum de toutes ces sommes.

Le processus est répété jusqu'à ce qu'il n'y ait plus de pixel qui soit modifié. Cet algorithme est massivement parallélisable.

(a)

	1	
1	0	1
	1	

(b)

1	1	1
1	0	1
1	1	1

(c)

4	3	4
3	0	3
4	3	4

(d)

	11		11	
11	7	5	7	11
	5	0	5	
11	7	5	7	11
	11		11	

FIG. 12 - Masques parallèles de (a) d_4 (b) d_8 (c) $d_{3,4}$ (d) $d_{5,7,11}$

Dans ces deux méthodes, le nombre d'itérations dépend de l'épaisseur de l'objet, et la complexité est en $O(N^3)$ pour une image $N \times N$. On préfèrera cependant des algorithmes où le nombre de passages sur l'image est fixe, car ceux-ci sont plus performants sur machine séquentielle.

2.4.2 Calcul séquentiel

L'algorithme séquentiel qui suit est dit de 'Rosenfeld'. On décompose le masque parallèle en deux demi-masques symétriques par rapport à 0 (figure 14).

(a)

4	3	4
3	0	

(b)

	0	3
4	3	4

(c)

	11		11	
11	7	5	7	11
	5	0		

(d)

		0	5	
11	7	5	7	11
	11		11	

FIG. 13 - Masques séquentiels de $d_{3,4}$ et $d_{5,7,11}$, avant (a,c) et arrière (b,d)

Le demi-masque *avant* contient les points antérieurs au milieu 0 dans le *sens du balayage*. Ses n points ont les coordonnées $(x_i,\ y_i)$ et le poids w_i. Le demi-masque *arrière* est donc constitué des n points $(-x_i,\ -y_i)$ de poids w_i. Aucun des deux demi-masques ne contient le point 0.

Parcours avant : de haut en bas et de gauche à droite

$$\mathcal{A}[x,y] = \min \{ \mathcal{A}[x + x_i,\ y + y_i] + w_i \}$$

Parcours arrière : de bas en haut et de droite à gauche

$$\mathcal{A}[x,y] = \min \{ \mathcal{A}[x,y]\ ,\ \mathcal{A}[x - x_i,\ y - y_i] + w_i \}$$

Cet algorithme est très efficace en $O(N^2)$, puisqu'il ne nécessite que deux consultations par point de l'image $\mathcal{A}$, et ce sur un voisinage restreint du point, indépendamment de la taille des objets à traiter (figure 14).

3	3	3	3	3	3
3	6	6	6	6	4
3	6	9	9	8	4
3	6	9	12	8	
		3	6	4	

$\longrightarrow$

3	3	3	3	3	3
3	6	6	6	6	3
3	6	7	7	4	3
3	3	4	6	3	
		3	3	3	

FIG. 14 - DM pour $d_{3,4}$ en 2 balayages séquentiels

2.4.3 Transformation inverse

La transformation inverse part d'une image de poids, et génère le disque associé à chacun de ces poids ; elle permet par conséquent la génération de formes. Le type de l'algorithme est le même que celui du §2.4.2, et utilise les mêmes demi-masques.

Parcours avant : de haut en bas et de gauche à droite

$$\mathcal{A}[x,y] = \max\{\, \mathcal{A}[x,y]\,,\ \mathcal{A}[x+x_i,\ y+y_i] - w_i \,\}$$

Parcours arrière : de bas en haut et de droite à gauche

$$\mathcal{A}[x,y] = \max\{\, \mathcal{A}[x,y]\,,\ \mathcal{A}[x-x_i,\ y-y_i] - w_i \,\}$$

L'axe médian est défini au chapitre 5. Il est extrait de DM et permet de retrouver la forme initiale par une transformation inverse de distance (figure 15).

	4	6	6	3		
3	2	3	7	8	5	2
		3	4	5	4	1
			1	2	1	

$\longrightarrow$

	2	3	3	2	1	
1	4	6	6	5	4	1
3	2	4	7	8	5	2
		3	4	5	4	1
			1	2	1	

FIG. 15 - DM^{-1} pour $d_{3,4}$ en 2 balayages séquentiels

Dans [Bor91b] est donné pour d_E un algorithme de transformation inverse en 3 passes, qui est l'extension de [Rag90b]. Cet algorithme ne fournit pas une vraie image de distance, car chaque vecteur pointe vers un germe au lieu d'un point du fond. Mais c'est aussi le cas pour les distances de chanfrein.

2.4.4 Parallélisation

Les multiples schémas algorithmiques présentés dans cet ouvrage sont souvent parallélisables. Lorsqu'une transformation est définie localement, et indépendamment du devenir du voisinage, alors l'algorithme est dit parallèle, et la parallélisation est triviale. On peut citer par exemple l'extraction de l'axe médian et des points selle.

Par contre, lorsque la transformation est définie séquentiellement, le calcul d'un point ne peut être fait qu'après celui de multiples autres, en respectant la chronologie. Il faut alors repenser ou adapter les algorithmes. C'est ce qui a été fait par Miguet et Robert dans [Mig91], pour l'algorithme de Rosenfeld, la machine cible étant un anneau de processeurs (typiquement des transputers). Leur algorithme est paramétré par la granularité des calculs élémentaires et la stratégie d'allocation des données. Dans la figure 16, on calcule le début d'un bloc à la même étape que la fin du bloc précédent, ici de taille 3×4 pixels.

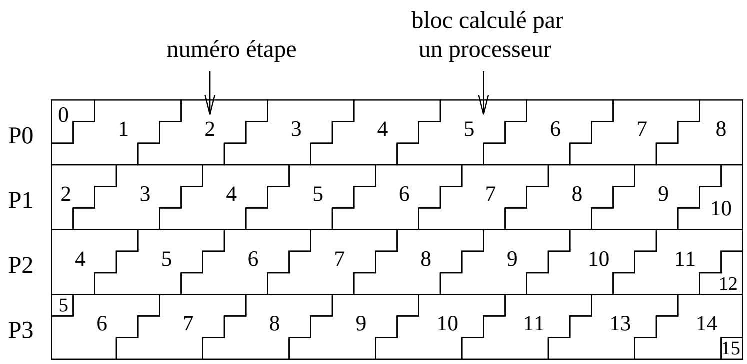


FIG. 16 - Algorithme sur machine parallèle [Mig91]

Toute une étude théorique est menée pour trouver un équilibre entre la charge de répartition des calculs entre processeurs, avec un délai d'initialisation minimal, et le volume de communications, pour ne pas dégrader les performances. Les résultats sont éloquents : sur un anneau de 32 transputeurs, le facteur d'accélération est de 26 par rapport à l'algorithme sur machine séquentielle.

Toujours sur des machines MIMD, Embrechts propose dans [Emb94] des algorithmes de type 'divide and conquer' pour les distances d_4, $d_{3,4}$ et d_E.

Cette approche demande une transformation de distance partielle dans chaque nœud, suivie par des échanges de données entre nœuds, grâce auxquelles une image de distance cohérente peut être produite. Plusieurs algorithmes sont donnés.

On mentionne enfin les travaux de Paglieroni [Pag92b] qui propose une architecture spécialisée, et ceux de Borgefors [Bor89], dont l'algorithme est présenté conjointement avec une méthode de stockage dans des pyramides.

2.5 Applications

Dans ce document, les images de distance forment le substrat pour le calcul de l'axe médian, de la ligne médiane et du squelette pondéré. Les images de distance permettent de nombreuses autres applications ; parmi elles, la mise en correspondance [Bar77, Bor86b], le lissage, la fusion, l'interpolation [Pag92a], la morphologie [Nac94]. Des variantes de transformation de distance sont également possibles :

2.5.1 Image avec contraintes

Les images de distance avec contraintes sont un cas spécial, où les images sources consistent non seulement en pixels objet et pixels fond, mais aussi en pixels obstacle. Une valeur dans une carte de distance avec contraintes donne la distance au fond non 'à vol d'oiseau', mais par un chemin qui évite tous les obstacles.

Une application typique est le déplacement en robotique [Dor86b], où le plus court chemin entre deux points est recherché. Des approfondissements théoriques sur les distances géodésiques sont cités dans [Mel91].

Deux classes d'algorithmes peuvent être appliqués. La première opère sur de simples images où les obstacles restent dans le même espace discret. L'adaptation de l'algorithme de Rosenfeld pour les distances de chanfrein est faite en mettant les pixels obstacle à une valeur 'infinie', et en interdisant leur mise à jour. Mais cette méthode est rarement suffisante [Dan93b]. Les algorithmes itératifs par les contours sont ici supérieurs [Rag90a, Ver89]. La deuxième classe agit sur une description paramétrique des obstacles, typiquement vectorielle. Ragnemalm décrit dans [Rag93a] un tel algorithme pour la distance euclidienne.

2.5.2 Diagramme de Voronoï

Le calcul du diagramme de Voronoï peut être effectué de façon exacte par la géométrie algorithmique [Ber94], ou de façon approchée en temps constant avec une image de distance [Bor86a].

On étiquette chaque germe avec un label unique dans une image $\mathcal{A}$, et dans $\mathcal{B}$ on met les germes en fond et le complémentaire en objet. On applique une transformation de distance sur $\mathcal{B}$, tout en conservant la trace des provenances de l'information de distance dans $\mathcal{A}$. A l'arrivée, chaque région de Voronoï est étiquetée à la valeur de son germe dans $\mathcal{A}$.

Les résultats sont d'autant meilleurs que la distance est plus euclidienne. Melkemi souligne dans [Mel92] que cet algorithme fonctionne également pour le calcul du Voronoï généralisé, et que l'approche discrète, par un échantillonnage, peut aider la réalisation du résultat continu.

2.6 Boules classiques

Alors que la boule de la distance euclidienne est un cercle, et que les boules des distances octogonales sont toujours des octogones, les distances de chanfrein peuvent avoir des boules d'aspect varié.

On peut obtenir une boule de rayon R pour un masque $(a, b, \ldots)$ donné en étiquetant chaque point à sa distance à l'origine. La boule est alors l'ensemble des points de valeur inférieure ou égale à $Ra + r$, où $r \in [0\,,\, a-1]$. Pour un rayon R fixé il y a donc une *famille* de a boules, en faisant varier r (figure 17). Le fait qu'il y ait plusieurs boules par rayon laisse supposer des phénomènes arithmétiques très intéressants.

			9			
	8	7	6	7	8	
	7	4	3	4	7	
9	6	3	0	3	6	9
	7	4	3	4	7	
	8	7	6	7	8	
			9			

r = 0

		10	9	10		
	8	7	6	7	8	
10	7	4	3	4	7	10
9	6	3	0	3	6	9
10	7	4	3	4	7	10
	8	7	6	7	8	
		10	9	10		

r = 1

	11	10	9	10	11	
11	8	7	6	7	8	11
10	7	4	3	4	7	10
9	6	3	0	3	6	9
10	7	4	3	4	7	10
11	8	7	6	7	8	11
	11	10	9	10	11	

r = 2

FIG. 17 - Boules de $d_{3,4}$ avec $R = 3$

L'allure générale de ces boules (figure 18) est un losange pour d_4, un carré pour d_8, un octogone pour $d_{2,3}$ et $d_{3,4}$, un hexadécagone pour $d_{5,7,11}$, etc. A mesure que l'on enrichit le masque, le nombre de côtés de ces polygones augmente. Les propriétés arithmétiques et géométriques sont étudiées au chapitre 3.

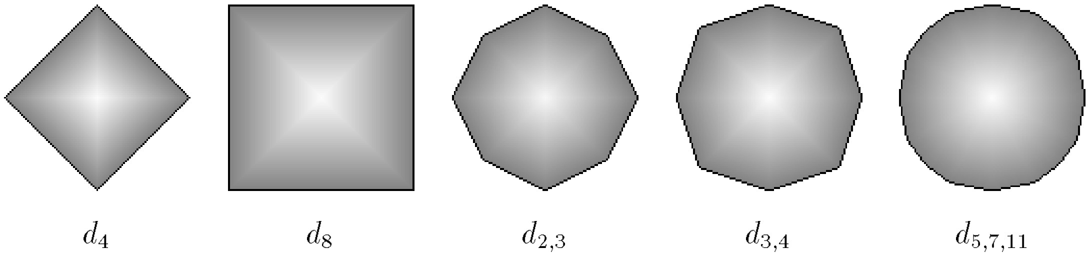


FIG. 18 - Boules classiques

Les taux d'erreur relativement à la distance euclidienne vont de 41% pour d_4 à 2% pour $d_{5,7,11}$. En pondérant des masques plus grands il est possible d'approximer plus finement encore la distance euclidienne (chapitre 4).

2.7 Conclusion

Des concepts de géométrie discrète ont été introduits. La présentation est faite en 2D, et les notions passent souvent très bien à la dimension supérieure, à l'exception de certains points fondamentaux tels que le théorème de Jordan 3D [Mal94], qui demandent une nouvelle étude.

Nous avons présenté les principales distances discrètes existantes. Par leurs qualités intrinsèques et leurs potentialités, les distances de chanfrein forment une alternative très enrichissante.

De plus les distances de chanfrein ont des algorithmes de transformation de distance séquentiels très performants. Leur passage au 3D est immédiat. La parallélisation de ces transformations l'est moins, mais des stratégies efficaces existent.

Les distances de chanfrein sont bien assises dans la littérature, sauf au niveau théorique, où des points obscurs subsistent ; notre objectif est de les éclaircir.

Par exemple, Borgefors montre qu'un masque à deux coefficients induit une distance si on peut l'écrire comme combinaison linéaire positive de d_4 et d_8. Ce raisonnement est juste pour les masques 3×3, mais par contre il devient insuffisant pour les masques plus grands ; il manque une condition de *convexité* sur la boule, comme nous le verrons au chapitre 3.

Chapitre 3

Arithmétique et distances

Paradoxalement, on ne trouve quasiment rien dans la littérature sur les propriétés arithmétiques et géométriques des boules de chanfrein. De même, personne jusqu'alors n'a répondu de façon globale à la question majeure : dans quelles conditions un masque de chanfrein induit-il une distance ?

La théorie des chanfreins que nous allons exposer dans ce chapitre, a démarré dans [Thi92c], puis a continué dans [Thi92a]. Suite à une étude complémentaire, nous présentons ici de tous nouveaux résultats.

Un rappel sur les suites de Farey et les points visibles est nécessaire pour définir le masque de chanfrein et les notations. La géométrie de la boule laisse apparaître les structures de cône d'influence et de déplacement élémentaire. Traduite de la convexité et de la symétrie, la condition de distance est donnée en termes de contraintes sur les pondérations. Leur liste par voisinage est établie, et illustrée par des contre-exemples. La condition de norme est enfin liée à la régularité du masque choisi. En complément nous donnons une collection de formules analytiques, et une extension au maillage rectangulaire.

3.1 Points visibles et suites de Farey

Avant de nous plonger dans l'étude des masques de chanfrein, nous présentons dans cette section les éléments arithmétiques indispensables.

Nous rappelons d'abord certaines propriétés des rationnels positifs, ou fractions 'naïves', telles que $\frac{1}{2}$ ou $\frac{7}{11}$. On trouve la démonstration de ces théorèmes, ainsi que d'autres propriétés intéressantes, au chapitre 3 de [Har78].

Définition 3.1 (Suites de Farey) *Les suites de Farey F_n d'ordre n sont les séries croissantes de fractions irréductibles entre 0 et 1, dont les dénominateurs n'excèdent pas n. Donc $\frac{h}{k} \in F_n$ si $0 \leq h \leq k \leq n$ et $\mathrm{pgcd}(h,k) = 1$.*

Théorème 3.1 *Si $\frac{h}{k} < \frac{h'}{k'}$ sont deux termes successifs de F_n, alors*

$$k\,h' - h\,k' = 1\,. \tag{1}$$

Théorème 3.2 *Si $\frac{h}{k} < \frac{h''}{k''} < \frac{h'}{k'}$ sont trois termes successifs de F_n, alors*

$$\frac{h''}{k''} = \frac{h+h'}{k+k'}\,. \tag{2}$$

La définition 3.1 donne la première série F_1. Selon (2), on obtient F_n connaissant F_{n-1} : entre chaque couple successif $\frac{h}{k} < \frac{h'}{k'}$ de F_{n-1}, on intercale le nouveau terme $\frac{h+h'}{k+k'}$ si $k + k' \leq n$.

F_1 $\frac{0}{1} < \frac{1}{1}$
F_2 $\frac{0}{1} < \frac{1}{2} < \frac{1}{1}$
F_3 $\frac{0}{1} < \frac{1}{3} < \frac{1}{2} < \frac{2}{3} < \frac{1}{1}$
F_4 $\frac{0}{1} < \frac{1}{4} < \frac{1}{3} < \frac{1}{2} < \frac{2}{3} < \frac{3}{4} < \frac{1}{1}$
F_5 $\frac{0}{1} < \frac{1}{5} < \frac{1}{4} < \frac{1}{3} < \frac{2}{5} < \frac{1}{2} < \frac{3}{5} < \frac{2}{3} < \frac{3}{4} < \frac{4}{5} < \frac{1}{1}$
F_6 $\frac{0}{1} < \frac{1}{6} < \frac{1}{5} < \frac{1}{4} < \frac{1}{3} < \frac{2}{5} < \frac{1}{2} < \frac{3}{5} < \frac{2}{3} < \frac{3}{4} < \frac{4}{5} < \frac{5}{6} < \frac{1}{1}$

De telles fractions $\frac{h}{k}$ peuvent être regardées comme la relation entre deux entiers positifs premiers entre eux, coordonnées du point (k,h) sur le réseau fondamental de $\mathbb{Z}^2$.

Définition 3.2 (Point visible) *Un point $P(x,y)$ est dit visible (i.e visible depuis l'origine) s'il n'y a aucun point du réseau sur (OP) qui soit situé entre O et P. Une condition nécessaire et suffisante est $\mathrm{pgcd}(x,y) = 1$.*

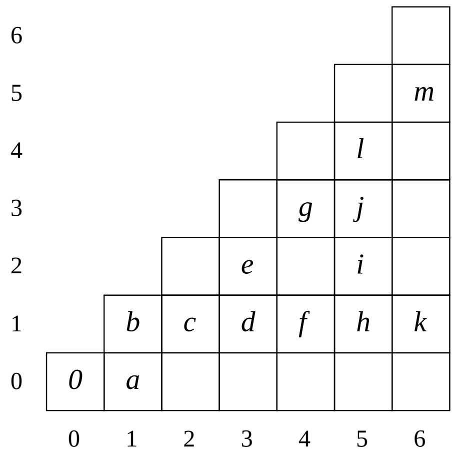


FIG. 19 - Noms des Points visibles (pour $x \leq 6$)

On note a, b, c, ... les points visibles dans le premier octant, dans l'ordre croissant de leur distance euclidienne à l'origine (figure 19). Cette notation est utilisée dans tout le document.

Regardons ce qui se passe lorsque les points visibles sont triés angulairement dans le sens trigonométrique par rapport à l'origine. Pour la figure 19 il vient :

$$a/k/h/f/d/i/c/j/e/g/l/m/b \tag{3}$$

Soit deux points visibles $P(x_p, y_p)$ et $Q(x_q, y_q)$, et soit $\Delta_p^q = \det \begin{vmatrix} x_p & x_q \\ y_p & y_q \end{vmatrix}$ l'aire du parallélogramme δ défini par (OP) et (OQ). L'ordre angulaire se traduit par

$$\angle(Ox, OP) < \angle(Ox, OQ) \ \Leftrightarrow\ \Delta_p^q > 0\,. \tag{4}$$

Théorème 3.3 *Si $\Delta_p^q = 1$ alors δ ne contient aucun point sur le réseau ; si $\Delta_p^q > 1$ alors δ contient au moins un point.*

Le théorème suivant relie les suites de Farey et les points visibles. L'équivalence entre l'ordre angulaire et l'ordre de Farey est immédiate [Har78]. Pour s'en convaincre, comparer (3) et F_6.

Théorème 3.4 *Soit V_n l'ensemble des points visibles $M_i(x_i, y_i)$ classés dans l'ordre angulaire et tels que $x_i \leq n$. Alors les y_i/x_i forment exactement la suite croissante de Farey F_n d'ordre n.*

3.2 Définition du masque de chanfrein

Un masque de chanfrein consiste en un voisinage centré en O, de taille $m \times m$, dans lequel on pondère certains déplacements. Le poids $w > 0$ affecté à un point (x, y) du masque est nommé *pondération locale*. Ce w engendre par translation les *périodes* $2w$, $3w$, etc (figure 20.a). Pour des raisons évidentes d'efficacité liée aux algorithmes du §2.4, un masque ne sera donc constitué que de points visibles.

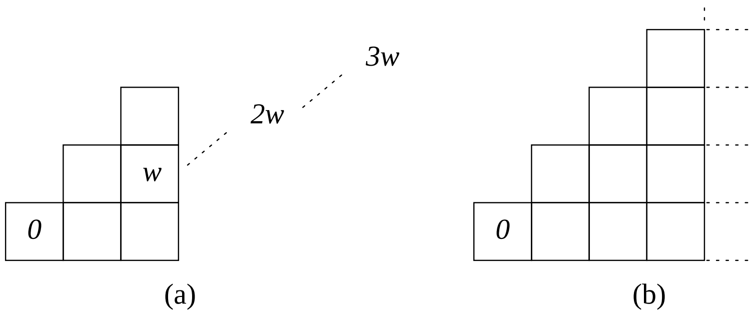


FIG. 20 - (a) Point visible et périodes ; (b) Premier octant.

Notre but est de trouver les conditions exactes pour qu'un tel masque induise bien une distance. Nous verrons à la section 3.4 que le théorème 3.8 impose la symétrie du masque par rapport à l'origine.

Le cas général pour lequel cette étude est poursuivie est le maillage carré. Il implique la symétrie par rapport aux axes et aux bissectrices, appelée *8-symétrie*. La section 3.6 est consacrée aux particularités de la maille rectangulaire, qui n'implique que la symétrie aux axes, appelée *4-symétrie*.

On appelle *générateur* la partie d'un masque de laquelle sont déduites, par la symétrie adéquate, tous les autres poids : il s'agit ici du premier octant (figure 20.b).

On continue à employer la notation statique a, b, c ... de la figure 19 pour les points visibles. On emploie également dans la suite la notation $M_i(x_i, y_i, w_i)$, des points visibles (x_i, y_i) affectés, de poids w_i, triés dans l'ordre angulaire avec (4). On note enfin $|(x, y)|$ le poids en (x, y) et $\|(x, y)\| = \sqrt{x^2 + y^2}$.

Définition 3.3 (Masque de chanfrein) *Un masque de chanfrein est valide si son générateur possède au moins les points a et b, ne comporte que des points visibles, et si ses pondérations locales vérifient certaines inégalités.*

On retrouve d_8 avec $a = 1$, $b = 1$, et d_4 avec $a = 1$, $b = 2$.

Le point fondamental est de fixer les bonnes inégalités, dont le rôle est de faire respecter l'inégalité triangulaire, la séparabilité, etc, de telle sorte qu'un masque de chanfrein induise bien une métrique discrète. Par exemple, dans un masque de taille 3×3, les contraintes sont

$$a \leq b \leq 2a$$

et pour un masque 5×5 on doit avoir

$$2a \leq c \ , \ 3b \leq 2c \ , \ c \leq a + b$$

comme nous le montrons par la suite. Nous établissons les conditions nécessaires et suffisantes pour tous les voisinages au théorème 3.10 de la section 3.4.

On note 'chanfrein $a, b, c, \ldots$' un tel masque, en remplaçant $a, b, c, \ldots$ par leur valeur, ou par '$\Diamond$' si le point correspondant n'est pas affecté.

En effet, on est libre de ne pas pondérer certains points visibles parmi $c, d, e, \ldots$ tout en conservant la validité de la définition du chanfrein. Ces choix éventuels vont conférer certaines propriétés à la distance discrète obtenue, et influer sur la géométrie de sa boule. Un très grand nombre de métriques sont donc possibles.

3.3 Géométrie de la boule

L'allure générale des boules classiques, présentées à la section 2.6, va du carré à l'hexadécagone. Les boules de chanfrein semblent toutes être des polygones, dont le nombre de côtés augmente au fur et à mesure que l'on enrichit le masque. Nous allons caractériser ce phénomène.

3.3.1 Cône d'influence

On étudie la structure de la carte des distances à l'origine, obtenue avec l'algorithme du §2.4, qui donne à un point le *minimum* des valeurs affectées aux points alentour augmentées des pondérations locales. Dans l'exemple de la figure 21, présenté pour $d_{5,7,11}$, nous avons noté dans le haut de chaque case la ou les provenances possibles parmi $5, 7, 11$ de ces minimums.

Définition 3.4 (Cône d'influence) *On appelle cône d'influence de M_i, M_{i+1}, le cône délimité par les droites (O, M_i) et (O, M_{i+1}). On l'appelle aussi : cône $\widehat{w_i, w_{i+1}}$.*

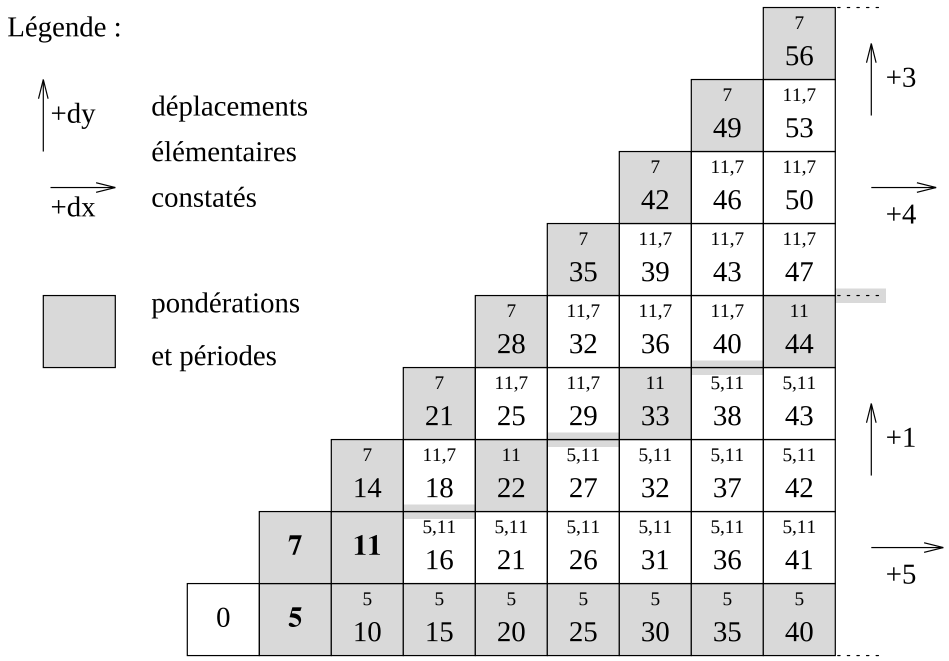


FIG. 21 - Cônes d'influence de chanfrein $5, 7, 11$

Définition 3.5 (Déplacements élémentaires) *Les déplacements élémentaires dx et dy à partir d'un point (x, y) correspondent au coût d'un déplacement d'une case ; ainsi $dx = |(x+1, y)| - |(x, y)|$ et $dy = |(x, y+1)| - |(x, y)|$.*

Dans l'exemple de la figure 21 avec $d_{5,7,11}$, les cônes $\widehat{5, 11}$ et $\widehat{11, 7}$ sont délimités par les zones grisées. Voici une propriété essentielle des cônes :

Théorème 3.5 *Dans un cône $\widehat{w_i, w_{i+1}}$, les seuls points du masque à intervenir dans la recherche du minimum pour le calcul des distances à O, sont les distances locales w_i et w_{i+1} :*

- *w_i et w_{i+1} engendrent par translations leurs périodes respectives ;*
- *tous les points strictement compris entre les droites (O, M_i) et (O, M_{i+1}) et qui sont au delà du parallélogramme (OM_i, OM_{i+1}), sont obtenus par w_i ou par w_{i+1}.*

Démonstration : Les pondérations locales w_j du masque engendrent chacune leur période par définition. Pour montrer le $2^{ème}$ point du théorème dans le cône $\widehat{w_i, w_{i+1}}$, on commence par considérer le parallélogramme discret (OM_i, OM_{i+1}), de sommets $O, M_i, M_{i+1}, M_i + M_{i+1}$, éventuellement d'intérieur vide, comme illustré figure 22.

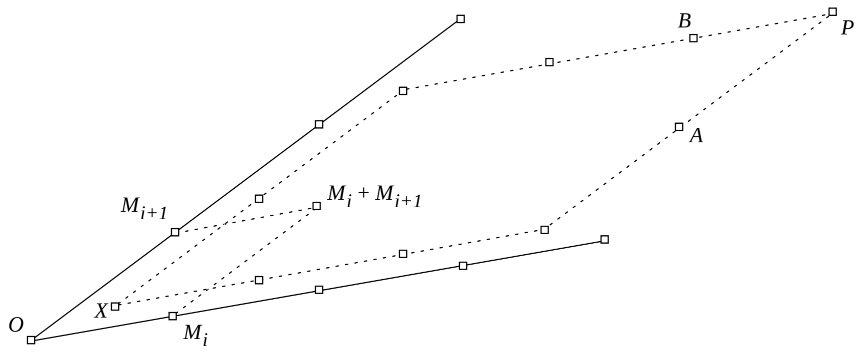


FIG. 22 - Le parallélogramme discret (OM_i, OM_{i+1})

Les points intérieurs à ce parallélogramme, s'il y en a, sont déterminés par certains points du masque, mais sans être atteints par M_i ou M_{i+1} ; dans tous les cas, le point $M_i + M_{i+1}$ a bien la valeur $w_i + w_{i+1}$ comme minimum.

A partir de là, on translate le parallélogramme et les valeurs de son adhérence (l'intérieur plus les points de bord, qui sont ici les sommets) dans tout le cône M_i, M_{i+1} : pour calculer le poids en un point P, il suffit de chercher à quel point X du premier parallélogramme il correspond, et de lui ajouter le nombre de déplacements M_i et M_{i+1} nécessaires pour l'atteindre ; donc ce sont bien les distances locales w_i et w_{i+1} qui interviennent dans le cône.

En notant $A = P - M_{i+1}$ et $B = P - M_i$ (figure 22), on a bien A et B sur le grand parallélogramme qui mène à X ; A et B sont deux points du cône d'influence, et la valeur P est donc bien obtenue par M_i ou M_{i+1} (de façon similaire) dans la recherche du minimum local, depuis A comme depuis B. □

3.3.2 Déplacements élémentaires

Lorsque les points M_i et M_{i+1} sont successifs au sens de Farey, on dit encore que ces points sont *consécutifs*, et que le cône $\widehat{w_i, w_{i+1}}$ est *régulier*. Un masque est dit *régulier* si tous ses points affectés sont consécutifs.

Théorème 3.6 *Si le cône $\widehat{w_i, w_{i+1}}$ est régulier, alors les déplacements élémentaires dx_i et dy_i sont constants dans tout l'intérieur du cône, et leurs poids sont donnés par :*

$$\begin{cases} dx_i = & y_{i+1}\, w_i - y_i\, w_{i+1} \\ dy_i = & x_i\, w_{i+1} - x_{i+1}\, w_i \end{cases} \tag{5}$$

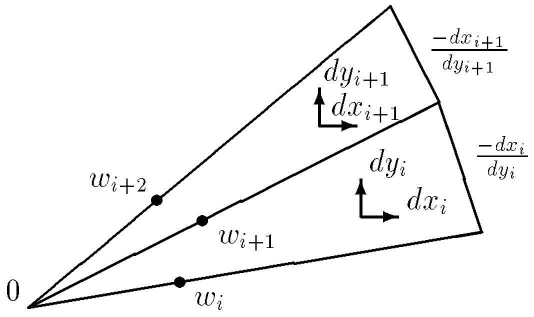


FIG. 23 - Notations des cônes

Démonstration : Avec les notations de la figure 23, nous montrons d'abord (i) que les déplacements élémentaires sont constants, puis (ii) nous les calculons.

(i) D'après le théorème 3.5, toutes les valeurs P à l'intérieur du cône M_i, M_{i+1}, privé du parallélogramme (OM_i, OM_{i+1}), sont égales à des combinaisons linéaires de M_i et M_{i+1}, plus un point X du parallélogramme (figure 22) ; c'est à dire $P = X + u\, M_i + v\, M_{i+1}$.

Or ici M_i et M_{i+1} sont consécutifs, donc d'après le théorème 3.3, le parallélogramme est d'intérieur vide. Donc X est égal à un des sommets du parallélogramme, d'où $P = u'\, M_i + v'\, M_{i+1}$.

Autrement dit, tous les points de l'intérieur du cône sont atteints par une combinaison linéaire de M_i et M_{i+1}, et donc les différences entre deux points l'un au dessus de l'autre, ou l'un à côté de l'autre, sont *indépendantes de leur position*, ce qui prouve (i).

(ii) Pour calculer dx_i et dy_i il faut trouver des couples (s, t) et (s', t') tels que

$$\begin{cases} s\, x_i + t\, x_{i+1} & = 1 \\ s'\, y_i + t'\, y_{i+1} & = 1 \end{cases}$$

pour des déplacements horizontaux ou verticaux de 1 case.

Or ici M_i et M_{i+1} sont consécutifs[1], donc (th. 3.1) $\Delta_i^{i+1} = x_i\, y_{i+1} - y_i\, x_{i+1} = 1$

$$\text{d'où} \begin{cases} y_{i+1}\, x_i - y_i\, x_{i+1} & = 1 \\ -x_{i+1}\, y_i + x_i\, y_{i+1} & = 1 \end{cases} \quad \text{et} \begin{cases} s = y_{i+1} & t = -y_i \\ s' = -x_{i+1} & t' = x_i \end{cases}$$

$$\text{finalement} \begin{cases} dx_i & = s\, w_i - t\, w_{i+1} & = y_{i+1}\, w_i - y_i\, w_{i+1} \\ dy_i & = s'\, w_i - t'\, w_{i+1} & = -x_{i+1}\, w_i + x_i\, w_{i+1} \end{cases}$$

[1] Dans le cas contraire, les déplacements élémentaires sont Δ_i^{i+1}-périodiques.

qui est bien l'expression (5). □

On retrouve de la sorte les déplacements élémentaires figure 21 avec $d_{5,7,11}$:

$$\widehat{5,11}\begin{cases} dx &= 1\times 5 - 0\times 11 &= 5 \\ dy &= 1\times 11 - 2\times 5 &= 1 \end{cases} \qquad \widehat{11,7}\begin{cases} dx &= 1\times 11 - 1\times 7 &= 4 \\ dy &= 2\times 7 - 1\times 11 &= 3 \end{cases}$$

Comme autre exemple on prend les points visibles consécutifs $c(2,1)$ et $j(5,3)$ dans la figure 24. On a bien $3\times 2 - 1\times 5 = 1$, d'où $dx = 3c - 1j$ et $dy = 2j - 5c$.

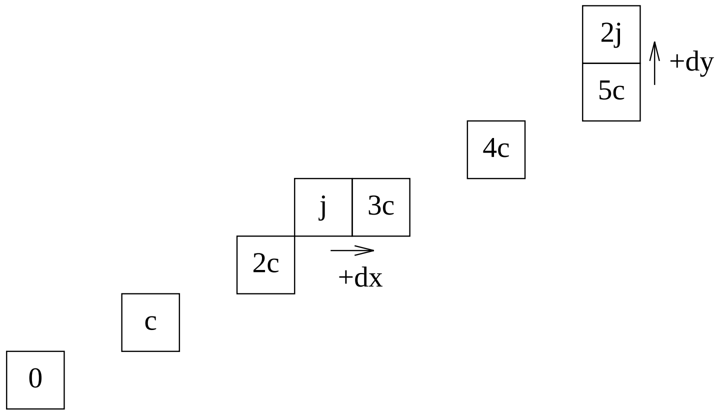


FIG. 24 - Déplacements élémentaires dans le cône $\widehat{c,j}$

Corollaire 3.1 *Dans le cône régulier* $\widehat{w_i, w_{i+1}}$, *les* intervalles de niveaux $]n-a, n]$ *sont des droites* $*$-connexes.

Il suffit de remarquer que $dx_i \leq a$ et $dy_i \leq a$ selon le principe même des chanfreins. Les droites discrètes $*$-connexes ont des propriétés très riches [Rev91], mais ne sont pas étudiées ici. Les courbes de niveaux dans les images de distance sont elles aussi $*$-connexes.

Corollaire 3.2 *Dans le cône régulier* $\widehat{w_i, w_{i+1}}$, *la pente du côté associé est* $\frac{-dx_i}{dy_i}$.

On peut maintenant expliquer la caractéristique déjà remarquée à propos des boules de chanfrein : ce sont des polygones 8-symétriques, dont chaque côté est porté par un cône d'influence. Pour un masque de chanfrein dont le générateur a n points, la boule est un polygone à $8\times(n-1)$ côtés au plus, car 2 pentes successives peuvent être identiques (cas de d_4 et d_8 par exemple).

3.3.3 Lien avec Montanari

Les distances de chanfrein ont pour ancêtre les distances de Montanari [Mon68]. Elles consistent à pondérer les déplacements dans un réseau donné, avec leur valeur euclidienne réelle.

Nous rappelons la longueur d'un chemin minimal sur un réseau (théorème 1, formule (6) dans [Mon68]) puis nous la généralisons au cas discret, en retrouvant le théorème 3.6.

Définition 3.6 (Réseau) *Un réseau de type n ($n \neq 0$) est constitué des points de $\mathbb{Z}^2$, où chaque point P est connecté par une ligne droite à chaque point Q appartenant au carré centré en P et de côté $(2n+1)$.*

Soit $P(x,y)$ un point du premier octant, et $t = \frac{y}{x}$ la pente de (OP), avec $0 \leq t \leq 1$. On note $T(O,P)$ la longueur d'un chemin minimal, joignant O à P, sur un réseau de type n.

Théorème 3.7 (Chemin minimal de Montanari) *Supposons que le rationnel t n'appartienne pas à la suite de Farey F_n. Soit t_i et t_j deux termes successifs de F_n encadrant t, alors*

$$T(O,P) = x\left[\frac{t-t_i}{t_j-t_i}\sqrt{1+t_j^2} + \frac{t_j-t}{t_j-t_i}\sqrt{1+t_i^2}\right], \quad t_i < t < t_j\,. \qquad (6)$$

En écrivant $t_i = \frac{y_i}{x_i}$ et $t_j = \frac{y_j}{x_j}$ sous leur forme irréductible, il vient $\sqrt{1+t_i^2} = \frac{w_i}{x_i}$ (idem pour j) avec $w_i = \sqrt{x_i^2+y_i^2}$ le poids du déplacement, d'où

$$T(O,P) = x\left[\frac{t-t_i}{t_j-t_i}\frac{w_j}{x_j} + \frac{t_j-t}{t_j-t_i}\frac{w_i}{x_i}\right]$$

or t_i et t_j sont successifs, donc $y_jx_i - y_ix_j = 1$, et $\frac{1}{t_j-t_i} = \frac{x_ix_j}{y_jx_i-y_ix_j} = x_ix_j$. En revenant aux coordonnées, (6) devient dans le cône $\widehat{w_i, w_j}$

$$T(O,P) = x\,(y_jw_i - y_iw_j) + y\,(x_iw_j - x_jw_i) \qquad (7)$$

et on retrouve exactement les déplacements élémentaires du théorème 3.6.

La validité de (7) est établie lorsque les poids w correspondent aux déplacements euclidiens. Elle est conservée pour un système de poids w entiers, sous réserve de bien définir une distance.

3.4 Condition de distance

La section 3.3 nous ayant renseignés sur la géométrie de la boule, il nous paraît élégant de formuler la condition de distance sous un aspect géométrique.

Théorème 3.8 *n est une norme ssi sa boule est convexe et symétrique.*

Démonstration : Soit X un espace vectoriel euclidien. On appelle *jauge* sur X toute $f : X \to \mathbb{R}$ convexe et telle que $f(\lambda x) = \lambda f(x) \quad \forall x \; \forall \lambda \geq 0$. Berger [Ber78] montre que la boule $C(f) = \{x \in X : f(x) \leq 1\}$ est un convexe de X, et que toute norme sur X est une jauge. Réciproquement, soit C un convexe compact contenant O ; il en dérive une jauge $f_C(x) = \inf\{\lambda : \lambda > 0\,, \, x \in \lambda C\}$, qui est une norme uniquement lorsque C est symétrique. □

3.4.1 Contraintes exactes

Le théorème 3.8 nous fit définir dès la section 3.2, le masque de chanfrein comme étant symétrique par rapport à l'origine. Il en va donc de même pour la boule. Connaissant la pente de chaque côté, il est possible de rendre la boule convexe :

Théorème 3.9 (Condition de distance) *Soit M un masque de chanfrein défini par les n points (x_i, y_i, w_i). Soit dx_i, dy_i les déplacements élémentaires donnés par le théorème 3.6. Alors M induit une distance ssi pour tout i on a*

$$0 \leq \frac{dy_i}{dx_i} \leq \frac{dy_{i+1}}{dx_{i+1}} \leq 1 \quad \textit{et} \quad dx_i > 0\,. \tag{8}$$

Si $n < 3$ (un seul cône), (8) s'écrit $0 \leq \frac{dy_1}{dx_1} \leq 1$ et $dx_1 > 0$.

Démonstration : La boule est convexe si dans le premier octant, la pente de chaque côté est inférieure à celle de son prochain, et est comprise entre la verticale et la diagonale :

$$\begin{array}{rrcccccc} & -\infty & \leq & \frac{-dx_i}{dy_i} & \leq & \frac{-dx_{i+1}}{dy_{i+1}} & \leq & -1 \\ \Leftrightarrow & \infty & \geq & \frac{dx_i}{dy_i} & \geq & \frac{dx_{i+1}}{dy_{i+1}} & \geq & 1 \\ \Leftrightarrow & 0 & \leq & \frac{dy_i}{dx_i} & \leq & \frac{dy_{i+1}}{dx_{i+1}} & \leq & 1 \end{array}$$

Les pentes sont définies lorsque $dx_i > 0$. □

On obtient les contraintes exactes sur les poids en appliquant le théorème 3.6 :

Théorème 3.10 (Contraintes sur les pondérations) *Soit M un masque de chanfrein défini par les n points (x_i, y_i, w_i). Alors M induit une distance ssi pour tout i on a*

$$\frac{x_{i+1}}{x_i}\, w_i \leq w_{i+1} \leq \frac{x_{i+1}+y_{i+1}}{x_i+y_i}\, w_i \quad et \quad \Delta_i^{i+2}\, w_{i+1} \leq \Delta_{i+1}^{i+2}\, w_i + \Delta_i^{i+1}\, w_{i+2} \qquad (9)$$

avec $\Delta_p^q = x_p y_q - x_q y_p$.

Démonstration : Les n points du masque sont triés par angles et appartiennent au 1[er] octant. On résume ces hypothèses par

$$0 \leq y_i \leq x_i\ ,\ 0 < x_i\ ,\ 0 < w_i\ ,\ \frac{y_i}{x_i} < \frac{y_{i+1}}{x_{i+1}}\ ,\ \ \forall i.$$

On décompose (8) en 4 étapes :

$$0 < dx_i\ ,\ 0 \leq dy_i\ ,\ dy_i \leq dx_i\ ,\ \frac{dy_i}{dx_i} < \frac{dy_{i+1}}{dx_{i+1}}\ ,\ \ \forall i.$$

1. De la condition $0 < dx_i$, il vient $0 < y_{i+1}w_i - y_i w_{i+1}$. Si $y_i = 0$ alors la condition revient à $0 < w_i$, qui est déjà en hypothèse, sinon à $w_{i+1} < \frac{y_{i+1}}{y_i}\, w_i$. Or on a toujours $\frac{x_{i+1}+y_{i+1}}{x_i+y_i} < \frac{y_{i+1}}{y_i}$ puisque $\frac{y_i}{x_i} < \frac{y_{i+1}}{x_{i+1}}$. Donc $0 < dx_i$ est impliquée par la 3[ème] condition $dy_i \leq dx_i$, et n'apparaît pas dans le théorème.

2. $0 \leq dy_i$ donne $\frac{x_{i+1}}{x_i}\, w_i \leq w_{i+1}$.

3. Le terme $dy_i \leq dx_i$ se traduit par $w_{i+1} \leq \frac{x_{i+1}+y_{i+1}}{x_i+y_i}\, w_i$. Son seul rôle est de contenir la pente des côtés en dessous de la diagonale, pour permettre la convexité après 8-symétrie.

4. On a $\frac{dy_i}{dx_i} \leq \frac{dy_{i+1}}{dx_{i+1}} \Leftrightarrow \frac{x_i w_{i+1} - x_{i+1} w_i}{y_{i+1} w_i - y_i w_{i+1}} \leq \frac{x_{i+1} w_{i+2} - x_{i+2} w_{i+1}}{y_{i+2} w_{i+1} - y_{i+1} w_{i+2}}$, où $x_{i+1} y_{i+1} w_i w_{i+2}$ s'annule, puis en divisant tout par w_{i+1}, $\Delta_i^{i+2}\, w_{i+1} \leq \Delta_{i+1}^{i+2}\, w_i + \Delta_i^{i+1}\, w_{i+2}$. Cette condition exprime l'inégalité triangulaire, et ne s'applique que lorsque $n \geq 3$. Tout raisonnement inductif est empêché par le terme $\Delta_p^q = x_p y_q - x_q y_p$, qui vaut 1 ssi w_p et w_q sont successifs (théorèmes 3.1 et 3.3). □

3.4.2 Liste par voisinage

Dans cette partie nous allons appliquer le théorème 3.10 à différents voisinages, pour illustrer davantage sa signification, mais aussi dans un but pratique.

w_i	a	k	h	f	d	i	c	j	e	g	l	m	b
x_i	1	6	5	4	3	5	2	5	3	4	5	6	1
y_i	0	1	1	1	1	2	1	3	2	3	4	5	1

FIG. 25 - Coordonnées des points visibles

Masque	$\star w_i \leq w_{i+1} \leq \star w_i$	$\star w_{i+1} \leq w_i + w_{i+2}$
3×3	$a \leq b \leq 2\,a$	
5×5	$2\,a \leq c \leq 3\,a$	$c \leq a + b$
	$\frac{1}{2}\,c \leq b \leq \frac{2}{3}\,c$	
7×7	$3\,a \leq d \leq 4\,a$	$d \leq a + c$
	$\frac{2}{3}\,d \leq c \leq \frac{3}{4}\,d$	$3\,c \leq d + e$
	$\frac{3}{2}\,c \leq e \leq \frac{5}{3}\,c$	$e \leq c + b$
	$\frac{1}{3}\,e \leq b \leq \frac{2}{5}\,e$	
13×13	$6\,a \leq k \leq 7\,a$	$k \leq a + h$
	$\frac{5}{6}\,k \leq h \leq \frac{6}{7}\,k$	$2\,h \leq k + f$
	$\frac{4}{5}\,h \leq f \leq \frac{5}{6}\,h$	$2\,f \leq h + d$
	$\frac{3}{4}\,f \leq d \leq \frac{4}{5}\,f$	$3\,d \leq f + i$
	$\frac{5}{3}\,d \leq i \leq \frac{7}{4}\,d$	$i \leq d + c$
	$\frac{2}{5}\,i \leq c \leq \frac{3}{7}\,i$	$5\,c \leq i + j$
	$\frac{5}{2}\,c \leq j \leq \frac{8}{3}\,c$	$j \leq c + e$
	$\frac{3}{5}\,j \leq e \leq \frac{5}{8}\,j$	$3\,e \leq j + g$
	$\frac{4}{3}\,e \leq g \leq \frac{7}{5}\,e$	$2\,g \leq e + l$
	$\frac{5}{4}\,g \leq l \leq \frac{9}{7}\,g$	$2\,l \leq g + m$
	$\frac{6}{5}\,l \leq m \leq \frac{11}{9}\,l$	$m \leq l + b$
	$\frac{1}{6}\,m \leq b \leq \frac{2}{11}\,m$	

FIG. 26 - Contraintes par masques

On classe les points visibles de la figure 19 dans l'ordre angulaire, et on obtient la figure 25. D'après le théorème 3.10, un masque de chanfrein défini dans l'un des voisinages de la figure 26 n'induira une distance que lorsque les inégalités correspondantes seront respectées.

Le calcul d'un masque de chanfrein se fait comme suit : on choisit un voisinage de points visibles $M_i\,(x_i, y_i)$ et on les pondère avec l'une des méthodes décrites au chapitre 4. Le modèle classique consiste à fixer une valeur pour a, puis à affecter les autres coefficients, avec $w_i = \mathrm{round}(a\sqrt{x_i^2 + y_i^2})$ ou $w_i = \mathrm{trunc}(a\sqrt{x_i^2 + y_i^2})$. Il est aisé de rechercher les valeurs de $a \in [1..255]$ qui sont *interdites* par le théorème 3.10 en fonction de la taille du masque (figure 27).

masque	a interdits pour l'arrondi
3×3	
5×5	2
7×7	$1..4, 6, 7, 9, 15$
9×9	$1..4, 6, 7, 9..11, 15, 16, 19, 23$
masque	**a interdits pour la partie entière**
3×3	
5×5	
7×7	$2, 4, 7, 12$
9×9	$1..4, 6..9, 11..14, 16, 18, 19, 21, 23, 24, 26, 28, 31, 33, 36, 41$

FIG. 27 - Listes de a interdits

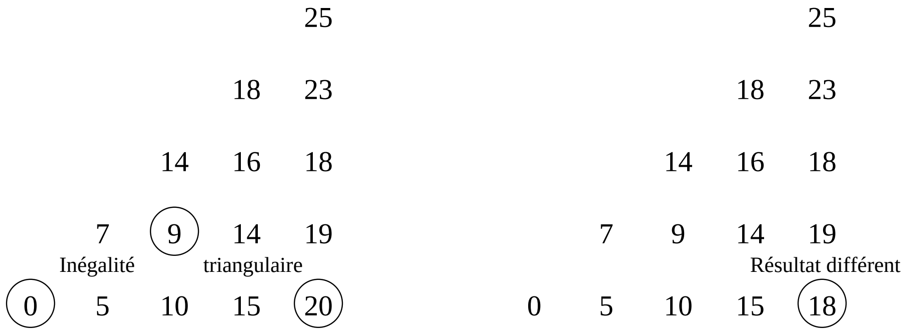
3.4.3 Contre-exemples

Dans les exemples de non-distances qui suivent, on se propose de montrer les répercussions négatives du non-respect du théorème 3.10.

Masque $5, 7, 9$

Ce masque 5×5 n'étant pas construit avec un modèle classique, au lieu de simplement vérifier si $a = 5$ est permis (figure 27), il faut tester toutes les inégalités (figure 26), à savoir $2\,a \leq c \leq 3\,a$, $\frac{1}{2}\,c \leq b \leq \frac{2}{3}\,c$, $c \leq a + b$. Or ni $2\,a \leq c$ ni $b \leq \frac{2}{3}\,c$ ne sont respectées, donc $d_{5,7,9}$ n'est pas une distance.

Cela se traduit par le non respect de l'inégalité triangulaire, dans l'image des distances figure 28.a. Les calculs séquentiel et parallèle (§2.4) ne donnent pas le même résultat, ce qui forme un contre-exemple à l'assertion de l'équivalence de ces deux méthodes.



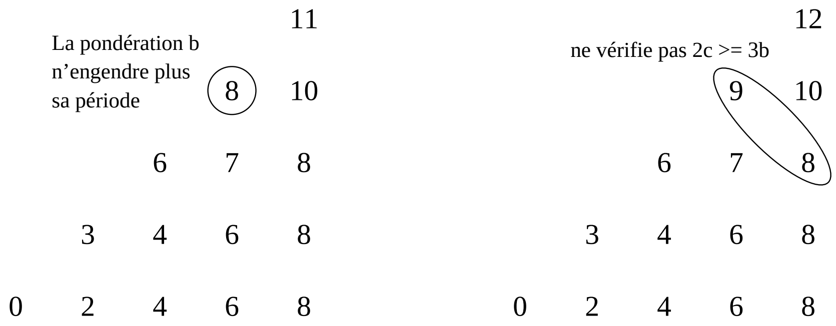
(a) Séquentiel (b) Parallèle

FIG. 28 - Séquentiel $\neq$ parallèle

Masque $2,3,4$

Ce masque quant à lui a bien été construit avec le modèle classique (arrondi), mais avec une valeur de a interdite pour 5×5 ; donc $d_{2,3,4}$ n'est pas une distance.

Sur l'image des distances figure 29.a, la pondération b n'engendre plus sa période, et l'algorithme de Rosenfeld est mis en défaut. Cette image est ainsi différente de l'image théorique (figure 29.b), qui correspond au modèle du §3.3.



(a) Séquentiel (b) Théorique

FIG. 29 - Séquentiel $\neq$ théorique

Nous retiendrons de ces contre-exemples que la validité de l'algorithme de Rosenfeld (§2.4.2) par rapport au modèle théorique, et son équivalence au schéma parallèle, sont toutes étroitement liées au fait que le masque induise une distance.

3.5 Compléments sur les distances

3.5.1 Norme

Une distance est une norme si elle respecte *l'homogénéité* : $n(\lambda x, \lambda y) = |\lambda|\, n(x, y)$. Le théorème suivant précise le rapport entre une norme et les notions de régularité pour le masque et le cône, qui sont définies au §3.3.2.

Théorème 3.11 *Une distance de chanfrein dérive d'une norme si son masque est régulier.*

Démonstration : L'homogénéité au niveau d'un cône revient à dire que les déplacements élémentaires sont constants. Or d'après le théorème 3.6, c'est le cas dès lors que le cône est régulier.

Réciproquement, dans un cône irrégulier M_i, M_{i+1}, on a vu dans la démonstration du théorème 3.6 que le premier parallélogramme (OM_i, OM_{i+1}) n'est pas vide, mais contient au moins un point X, qui ne provient ni de M_i ni de M_{i+1}, mais de pondérations extérieures ; cela engendre des perturbations dans les déplacements élémentaires, qui contreviennent à l'homogénéité du cône, et interdit au masque de dériver d'une norme. □

Un masque de voisinage complet est toujours régulier.

Par exemple (a, b, c, d, e) dans le voisinage 5×5 est régulier. L'ordre angulaire est $a/d/c/e/b$; or a/c et c/b sont chacun consécutifs, donc les masques $(a, b, c, d, \Diamond)$ et $(a, b, c, \Diamond, e)$ sont réguliers.

Par contre, b/d ne sont pas consécutifs, donc $(a, b, \Diamond, d, e)$ n'est pas régulier.

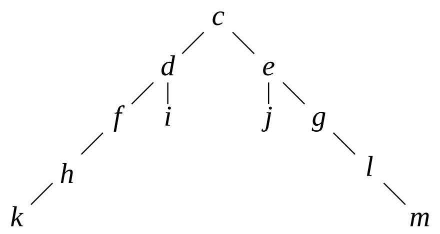


Fig. 30 - Arbre des masques réguliers obtenus avec la règle suivante : toute feuille est supprimable ; un nœud est supprimé avec tous ses fils.

Au chapitre 4, on sera amené à optimiser le masque en répartissant équitablement les angles des cônes, ce qui revient à supprimer certaines pondérations. Il est possible de construire des masques incomplets qui restent réguliers (figure 30). On conserve alors les propriétés édictées dans ce chapitre.

Mais *quid* des masques irréguliers? Les trop nombreux cas de figure envisageables nous contraignent au cas par cas. L'expérience nous apprend que dans les cônes irréguliers, les déplacements élémentaires ne sont plus constants mais cycliques, et que les intervalles de niveaux $]n-a, n]$ sont encore des droites $*$-connexes. On garde la distance, mais on perd la norme (et l'axe médian, cf chapitre 5).

3.5.2 Formules directes

Le calcul d'une carte de distance se fait très facilement grâce à l'algorithme de Rosenfeld, dont la rapidité a donné leur nom aux distances de chanfrein. Il est bien utile également de pouvoir calculer directement la distance entre deux points. Voici donc regroupées dans cette section, des formules analytiques pour les distances de chanfrein. On expose trois méthodes générales, avec leur application à 2 et 3 pondérations, puis leur expression pour les distances classiques. On développe enfin le calcul avec 5 pondérations.

Soit $P(x_P, y_P)$ et $Q(x_Q, y_Q)$ deux points, et $x = |x_Q - x_P|$, $y = |y_Q - y_P|$.

• La première méthode consiste à écrire $d_C(P,Q) = x\,dx + y\,dy$, où dx et dy sont les déplacements élémentaires dans le cône basé en P et contenant Q. Il faut donc trouver le cône, puis calculer les dx et dy correspondants avec le théorème 3.6.

Dans le cône $\widehat{a,b}$, on a : $dx = a, dy = b-a$; dans $\widehat{a,c}$: $dx = a, dy = c-2a$; et dans $\widehat{c,b}$: $dx = c-b, dy = 2b-c$. Donc $d_{a,b} = a\,x + (b-a)\,y$, et $d_{a,b,c} = a\,x + (c-2a)\,y$ ou $(c-b)\,x + (2b-c)\,y$, suivant le cône $\widehat{a,c}$ ou $\widehat{c,b}$.

Pour les distances classiques, en supposant que Q est dans le premier octant de P, on a $d_4 = d_{1,2} = x+y$, $d_8 = d_{1,1} = x$, $d_{3,4} = 3x+y$, $d_{5,7,11} = 5x+y$ ou $4x+3y$.

• La deuxième méthode consiste à écrire d_C dans chaque cône comme combinaison linéaire de d_4 et d_8. Comme $d_4 = x+y$ et $d_8 = \max(x,y)$, on trouve que $d_C = dy\,d_4 + (dx-dy)\,d_8$. Donc $d_{a,b} = (b-a)\,d_4 + (2a-b)\,d_8$ et $d_{a,b,c} = (c-2a)\,d_4 + (3a-c)\,d_8$ ou $(2b-c)\,d_4 + (2c-3b)\,d_8$.

Ainsi on retrouve $d_{3,4} = d_4 + 2\,d_8$ [Arc88]. Enfin $d_{5,7,11} = d_4 + 4\,d_8$ ou $3\,d_4 + d_8$.

• La troisième et dernière méthode consiste à écrire la distance de P à Q en fonction de déplacements horizontaux et diagonaux H et D. Il est facile de voir que $H = 2\,d_8 - d_4$ et $D = d_4 - d_8$. De là, $d_{a,b} = a\,H + b\,D$ [Arc88] et $d_{a,b,c} = a\,H + b\,D + (c-a-b)\,\min(H,D)$ [Thi94b]. Cette dernière est calculable directement, sans avoir à chercher le cône ni à se ramener au premier octant par des symétries.

Par conséquent, $d_4 = H + 2D$, $d_8 = H + D$, $d_{3,4} = 3H + 4D$ et $d_{5,7,11} = 5H + 7D - \min(H,D)$ (formule nettement plus condensée que dans [Arc92]).

En résumé, les expressions que l'on utilisera pour un calcul efficace au chapitre 6 sont $d_{2,3} = d_4 + d_8$, $d_{3,4} = d_4 + 2\,d_8$, et $d_{5,7,11} = 5H + 7D - \min(H,D)$.

On s'inspire de ces méthodes pour calculer $d_{a,b,c,d,e}$, dont aucune expression ne figure dans la littérature. En appliquant le théorème 3.6 aux coordonnées des 5 points a, d, c, e, b (figure 25), on peut écrire dans chaque cône :

$$\begin{array}{lcl}
d_{cône} &=& dx\,x + dy\,y &=& dy\,d_4 + (dx - dy)\,d_8 \\
d_{\widehat{ad}} &=& a\,x + (d-3a)\,y &=& (d-3a)\,d_4 + (4a-d)\,d_8 \\
d_{\widehat{dc}} &=& (d-c)\,x + (3c-2d)\,y &=& (3c-2d)\,d_4 + (3d-4c)\,d_8 \\
d_{\widehat{ce}} &=& (2c-e)\,x + (2e-3c)\,y &=& (2e-3c)\,d_4 + (5c-3e)\,d_8 \\
d_{\widehat{eb}} &=& (e-2b)\,x + (3b-e)\,y &=& (3b-e)\,d_4 + (2e-5b)\,d_8
\end{array}$$

Comme nous l'avons dit plus haut, il est plus intéressant de s'affranchir de la détermination des cônes, en utilisant les déplacements H et D.

$$\begin{array}{lcll}
d_{cône} &=& dx\,H + (dy + dx)\,D & \\
d_{\widehat{ad}} &=& a\,H + (d-2a)\,D & 2D \le H \\
d_{\widehat{dc}} &=& (d-c)\,H + (2c-d)\,D & D \le H \le 2D \\
d_{\widehat{ce}} &=& (2c-e)\,H + (e-c)\,D & H \le D \le 2H \\
d_{\widehat{eb}} &=& (e-2b)\,H + b\,D & 2H \le D
\end{array}$$

En posant un système d'équations on trouve finalement

$$\begin{aligned}
d_{a,b,c,d,e} &= a\,H + b\,D + (3c-d-e)\,\min(H,D) + \\
&\quad (e-c-b)\,\min(2H,D) + (d-c-a)\,\min(H,2D) \qquad (10)
\end{aligned}$$

Les termes en facteur devant les 'min' correspondent visiblement à

$$(\Delta_i^{i+2}\,w_{i+1} - w_i - w_{i+2}).$$

Cette indication nous met sur la voie d'une formulation générale des distances de chanfrein avec H et D.

3.6 Maille rectangulaire

Certains systèmes de vision industrielle, utilisant une caméra matricielle, numérisent les images avec un pas de discrétisation différent en abscisse et en ordonnée. Dans ce contexte, il peut être intéressant de réétudier les opérateurs de distance pour tenir compte de la forme rectangulaire des pixels [Bol92]. Cette adaptation permet d'éviter les problèmes de rééchantillonnage.

Plutôt que de se placer sur un réseau rectangulaire, donc à coordonnées non entières, nous allons conserver le réseau fondamental de $\mathbb{Z}^2$ comme espace de travail, et reporter la rectangularité sur l'allure ovale des boules.

Les masques de chanfrein à employer ici sont donc 4-symétriques. Leur générateur est le premier quartant, pour lequel on adopte les notations de la figure 31. On étend également la notation angulaire $M_i(x_i, y_i, w_i)$ à tout le quartant.

4		*F*		*G*	
3		*D*	*E*		*g*
2		*C*		*e*	
1	*A*	*b*	*c*	*d*	*f*
0	*0*	*a*			
	0	1	2	3	4

FIG. 31 - Noms des points en maille rectangulaire

Les propriétés des cônes et déplacements élémentaires du §3.3 sont intégralement conservées. Par contre, les conditions de distance vont différer.

Une boule 4-symétrique est convexe si, dans le premier quartant, la pente de chaque côté est inférieure à celle de son prochain, et est comprise entre la verticale et l'horizontale. Le passage de la 8- à la 4-symétrie revient simplement à remplacer 'diagonale' par 'horizontale', i.e $\frac{dy_i}{dx_i} \leq 1$ par $\frac{dy_i}{dx_i} \leq +\infty$ dans (8). La condition de distance se réécrit en remplaçant (8) dans le théorème 3.9 par :

$$0 \leq \frac{dy_i}{dx_i} \leq \frac{dy_{i+1}}{dx_{i+1}} \quad \text{et} \quad \begin{cases} dx_i > 0 & \text{dans le premier octant} \\ dy_i > 0 & \text{dans le second octant} \end{cases} \tag{11}$$

Pour obtenir les contraintes sur les pondérations, on supprime simplement la condition $dy_i \leq dx_i$ dans la démonstration du théorème 3.10. Dès lors, il faut remettre les conditions $0 < dx_i$ ou $0 < dy_i$ suivant l'octant. Les contraintes (9) sur les pondérations sont donc remplacées par

$$\frac{x_{i+1}}{x_i}\, w_i \leq w_{i+1} \leq \frac{y_{i+1}}{y_i}\, w_i \quad \text{et} \quad \Delta_i^{i+2}\, w_{i+1} \leq \Delta_{i+1}^{i+2}\, w_i + \Delta_i^{i+1}\, w_{i+2} \tag{12}$$

avec $w_{i+1} < \frac{y_{i+1}}{y_i}\, w_i$ dans le premier octant, et $\frac{x_{i+1}}{x_i}\, w_i < w_{i+1}$ dans le second octant.

Le tableau 26 est remplacé par la figure 32, où sont réécrites avec (12) les inégalités pour les voisinages 3×3 et 5×5.

Masque	$\star\, w_i \leq w_{i+1} \leq \star\, w_i$		$\star\, w_{i+1} \leq w_i + w_{i+2}$	
3×3	$a \leq b$	$A \leq b$	$b \leq a + A$	
5×5	$2\,a \leq c$	$2\,A \leq C$	$c \leq a + b$	$C \leq A + b$
	$b < c \leq 2\,b$	$b < C \leq 2\,b$	$b \leq c + C$	

FIG. 32 - Contraintes par masques en maille rectangulaire

Le calcul d'un masque est effectué en tenant compte du rapport L de rectangularité, par $w_i = \text{round}(a\sqrt{x_i^2 + (L\, y_i)^2})$ ou $w_i = \text{trunc}(a\sqrt{x_i^2 + (L\, y_i)^2})$. Il n'est donc pas possible d'établir une liste de a interdits indépendamment de L.

La condition de norme du §3.5.1 est inchangée, le masque devant être régulier sur le quartant. Il faut par contre recalculer les formules analytiques du §3.5.2, car les poids ne sont plus les mêmes dans les deux octants.

La validité de nos développements autour des masques de chanfrein est donc globalement conservée en maillage rectangulaire. Il en va de même pour les schémas algorithmiques du §2.4.

3.7 Conclusion

Quelques éléments arithmétiques liés aux points visibles et aux suites de Farey, et la notation dans l'ordre angulaire des points du masque, nous ont permis de dégager les structures générales de cônes et de déplacements élémentaires.

Ces concepts ont fait apparaître la géométrie de la boule. Nous avons alors traduit, d'une condition de convexité, les contraintes exactes sur les pondérations pour qu'un masque induise bien une distance. Des contre-exemples ont montré la nécessité d'avoir une vraie distance, pour la validité de l'algorithme de Rosenfeld. La régularité du masque a été reliée aux normes, et de nouvelles formules de calcul direct ont été établies.

Nous avons montré que la théorie pouvait s'adapter sans problème au maillage rectangulaire. Celui-ci peut être employé pour éviter un rééchantillonnage. Mais au niveau de l'image, l'emploi d'une telle distance est moins isotrope que dans le maillage traditionnel carré [Coq93].

Ces travaux sont naturellement extensibles en trame hexagonale d'une part, et dans l'espace 3D d'autre part. Il suffit de reprendre les notations et de dérouler les raisonnements.

Dans le calcul des pondérations, la possibilité de prendre un facteur d'échelle différent de a est exploitée au chapitre 4. On peut enfin imaginer des variantes aux masques de chanfrein, par exemple en travaillant en sous-résolution, ou en distinguant des pondérations proches, de pondérations plus éloignées, qui 'rattraperaient' l'erreur par rapport à la distance euclidienne.

Chapitre 4

Optimisation

Ce chapitre est consacré à l'optimisation des masques de chanfrein. Le principe général de calcul des pondérations dans un voisinage donné est de minimiser l'erreur commise par rapport à la distance euclidienne.

Le sujet a porté à polémique, chaque nouvel article publié répondant aux précédents, et apportant de nouveaux critères, dans le but d'améliorer l'isotropie ou de minimiser plus finement. De multiples masques ont été proposés, ainsi que des taux d'erreur, voire des approches philosophiques.

Nous ajoutons notre pierre à l'édifice, avec la définition d'une erreur relative, une méthode de calcul et un algorithme d'optimisation, suffisamment souples pour s'appliquer à des masques de taille quelconque.

Les performances des masques sont systématiquement comparées, et les rapports théoriques entre notre méthode et la bibliographie sont analysés.

4.1 Introduction

Une raison importante de l'attrait des distances de chanfrein est la possibilité d'approximer autant que souhaité la distance euclidienne, en jouant sur la taille du masque et la valeur des pondérations.

Les applications nécessitent souvent un compromis spécifique entre la qualité de l'approximation et la charge de calcul. On a donc intérêt à disposer d'une variété de masques, chacun étant optimisé pour une taille précise.

Les *desiderata* à concilier sont multiples :

- On travaille principalement sur des images de taille 512×512 codées sur 16 bits (0..65535). Un objet dans cette image ne pourra avoir une épaisseur supérieure à $\frac{512}{2} = 256$, et donc la première pondération a qui joue le rôle de facteur d'échelle devra être impérativement inférieure à $\frac{65536}{256} = 256$. Les valeurs de a que l'on s'autorise sont donc comprises entre 1 et 255, en privilégiant les plus petites valeurs de a, qui diminuent les *coûts* de stockage et de calcul.

- On souhaite que le masque induise une distance (chapitre 3). Si l'on utilise la méthode classique de calcul des pondérations à partir de a (§3.4.2), il suffit de se cantonner aux *a permis* (figure 27).

- L'optimisation sous-entend la minimisation d'un critère. Se posent alors les problèmes du choix du critère, et de la méthode d'optimisation, bien souvent liée à la nature du critère.

- D'autres éléments sont à prendre en compte, par exemple pour permettre le calcul de l'axe médian (chapitre 5). Ainsi, le choix de coefficients premiers entre eux par cône, permet de stocker des tables de correspondance de taille finie, et un a petit limite considérablement la taille de ces tables.

- Respecter la condition de norme peut se révéler indispensable (par exemple, pour le calcul de l'axe médian encore une fois). La norme dépend non des poids, mais de la régularité du masque. La figure 30 du §3.5.1 montre comment obtenir de tels masques. Bien entendu tous les masques complets sont réguliers.

- Enfin, les méthodes que nous allons voir sont plus ou moins complexes à mettre en œuvre. Pour le calcul de grands masques, la simplicité et la souplesse sont les bienvenues !

Les méthodes existantes sont étudiées au §4.2 ; les concepts y sont introduits au fur et à mesure. Nous développons notre approche au §4.3, puis nous proposons au §4.4 de nouveaux masques de chanfrein.

4.2 Méthodes existantes

4.2.1 Approches locales

L'approche locale consiste à approximer les déplacements euclidiens $\sqrt{2}$, $\sqrt{5}$, ... par $\frac{b}{a}$, $\frac{c}{a}$, ... (cf figure 19 pour les notations a, b, c, ...). Pour des masques de chanfrein à deux points a et b, l'idée la plus intuitive consiste à utiliser les *fractions continues*

$$1+\frac{1}{1+\frac{1}{1+\frac{1}{1+\cdots}}}. \tag{13}$$

En initialisant par $a = 1$ et $b = 2$ et en bouclant sur $b' = 2a + b$, $a' = a + b$ on obtient la série

$$\frac{2}{1},\ \frac{4}{3},\ \frac{10}{7},\ \frac{24}{17},\ \frac{58}{41},\ \frac{140}{99},\ \frac{338}{239},\ \frac{816}{577} \tag{14}$$

qui converge très vite vers $\sqrt{2}$, mais dont les termes deviennent aussi très grands ; or la borne supérieure des a que l'on fixe au §4.1 est 255, ce qui est dépassé dès le $8^{ème}$ terme. L'autre inconvénient des fractions continues est qu'elles ne permettent pas l'approximation *simultanée* des déplacements euclidiens.

C'est pourquoi nous développâmes dans [Thi92c] la méthode locale suivante.

Fixons un voisinage de pondération $(x_i,\ y_i,\ w_i)$. Pour un a donné, les autres pondérations prennent classiquement la valeur $w_i = \mathrm{round}(a\sqrt{x_i{}^2 + y_i{}^2})$ (§3.4.2). Il faut donc trouver un a tel que la partie tronquée $w_i - a\sqrt{x_i{}^2 + y_i{}^2}$ soit la plus petite possible pour chaque coefficient w_i du masque. On note

$$E_a = \max_i \left(w_i - a\sqrt{x_i{}^2 + y_i{}^2} \right) \tag{15}$$

le maximum des erreurs des w_i pour ce a. Pour chaque $a \in [1..255]$ on calcule E_a, puis on les classe par ordre croissant, de telle sorte que les a intéressants soient en tête de liste.

Cette méthode est très simple, et permet de calculer rapidement de grands masques [Thi92b]. Mais en même temps, elle ne s'intéresse qu'à une approximation locale de d_E. Elle ne permet pas l'évaluation de l'erreur sur les points éloignés, ou à l'intérieur des cônes par exemple.

4.2.2 La méthode de Borgefors

Borgefors propose une approche qui consiste à minimiser le maximum de l'erreur commise par rapport à d_E sur une droite verticale, parce que le support de l'image est rectangulaire [Bor86a].

Les masques 3×3, 5×5 et 7×7 sont analysés. La partie mathématique de l'optimisation est développée dans l'espace continu.

Dans chaque cône délimité par les droites $y = \alpha\, x$ et $y = \beta\, x$, la distance de chanfrein est ramenée à une expression du type

$$d_C = y\, S + x\, T \tag{16}$$

où les bornes du cônes et les valeurs de S et T sont calculées au cas par cas (sans la généralisation du chapitre 3).

On considère la droite verticale $x = M$ sur l'image des distances à l'origine, et le cône de pentes α, β, dans lequel sont déterminés S et T. La différence entre d_C et d_E sur la droite est

$$\mathrm{Diff}(y) = y\, S + M\, T - \sqrt{M^2 + y^2}\,, \quad \alpha\, M \leq y \leq \beta\, M\,. \tag{17}$$

Le maximum absolu de (17) est la mesure de Borgefors de l'optimalité de la distance dans ce cône. Son but est de déterminer les pondérations qui minimisent ce maximum. Le maximum est atteint lorsque la dérivée s'annule, ou aux bornes de l'intervalle $y = \alpha\, M$ et $y = \beta\, M$. La dérivée de $\mathrm{Diff}(y)$ est

$$\mathrm{Diff}'(y) = S - \frac{y}{\sqrt{M^2 + y^2}}\,. \tag{18}$$

L'extremum est atteint lorsque $\mathrm{Diff}'(y) = 0$, i.e pour

$$y_0 = \frac{M\, S}{\sqrt{1 - S^2}} \tag{19}$$

en s'arrangeant pour que $|S| < 1$, et si $\alpha\, M \leq y_0 \leq \beta\, M$.

Le maximum de la différence (17) dans l'intervalle $\alpha\, M \leq y \leq \beta\, M$ est

$$\mathrm{Diff}_2 = \mathrm{Diff}(y_0) = \left(T - \sqrt{1 - S^2}\right) M \tag{20}$$

et les valeurs aux bornes sont

$$\mathrm{Diff}_1 = \mathrm{Diff}(\alpha\, M) = \left(\alpha\, S + T - \sqrt{1 + \alpha^2}\right) M\,, \;\; \mathrm{Diff}_3 = \mathrm{Diff}(\beta\, M)\,. \tag{21}$$

Le taux maximal d'erreur dans le cône est donc

$$\text{maxdiff} = \max\left(|\text{Diff}_1|,\ |\text{Diff}_2|,\ |\text{Diff}_3|\right) \tag{22}$$

et il faut minimiser globalement l'erreur sur tous les cônes du masque.

Comme les expressions dans (20) et (21) sont facteurs de M, Borgefors optimise de fait l'expression

$$\frac{d_C - d_E}{M} \tag{23}$$

ce qui a son importance dans les sections suivantes.

Nous ne revenons pas sur le détail de la détermination des pondérations, abondamment détaillée dans [Bor86a]. En résumé, sont d'abord calculés des coefficients optimaux *réels*, en fixant un certain degré de liberté, ou en intersectant des courbes. Puis elle cherche une approximation entière des coefficients optimaux, en les multipliant par un facteur d'échelle.

Par exemple dans le voisinage 3×3, en fixant $a = 1$ elle trouve

$$b_{opt} = \frac{1}{\sqrt{2}} + \sqrt{\sqrt{2} - 1} \approx 1.35070\,. \tag{24}$$

qui est correctement approché par $\frac{4}{3}$. Elle justifie de cette façon le masque $(3, 4)$ dans le voisinage 3×3, puis $(5, 7, 11)$ dans le voisinage 5×5. Elle ne recommande pas de masque précis dans le voisinage 7×7, ni leur emploi.

La méthode s'effectue au cas par cas, et devient vite très lourde à gérer pour de plus grands voisinages. Le passage au discret est abrupt.

Certains auteurs ont critiqué le choix du critère (23) à optimiser, ainsi que le support de test, une droite verticale, qui n'est peut-être pas le meilleur gage d'isotropie.

4.2.3 Améliorer l'isotropie

Borgefors suggère dans [Bor86a] l'utilisation d'un autre critère pour l'optimisation, qui consiste à minimiser la *différence moyenne* avec la distance euclidienne. C'est Vossepoel qui se charge de cette étude [Vos88]. Il reprend l'analyse de Borgefors dans les voisinages 3×3 et 5×5, et tente d'améliorer l'isotropie.

En effet dans l'équation (17), la variable y n'est pas idéale, car dans le calcul de l'intégrale requise pour cette optimisation, le même incrément de y engendrera des incréments variables de l'angle φ. Vossepoel opère donc la substitution $y = M\ \tan\varphi$

dans (17). Il obtient une fonction de φ

$$f(\varphi) = \frac{\mathrm{Diff}(M\ \tan\varphi)}{M} = T + S\ \tan\varphi - \frac{1}{\cos\varphi} \tag{25}$$

puis minimise la moyenne de la variance

$$\overline{f^2(\varphi)} = \frac{1}{\varphi_2 - \varphi_1}\int_{\varphi_1}^{\varphi_2} f^2(\varphi)\,d\varphi \tag{26}$$

dans chaque cône d'angles φ_1, φ_2.

La méthode de Vossepoel suit une démarche similaire en tous points à celle de Borgefors, seul le critère ayant changé. Après avoir déterminé des coefficients optimaux, différents de ceux de Borgefors, il revient à des fractionnels, mettant en avant les masques $(2,3)$, $(7,10)$ et $(16,23)$ dans le voisinage 3×3, et confortant $(5,7,11)$ dans le voisinage 5×5.

En fait, Vossepoel minimise encore une fois l'expression (23), simplement par une autre méthode. Ce n'est pas un progrès décisif pour l'isotropie.

Le calcul des coefficients optimaux effectué par Vossepoel est commenté par Beckers et Smeulders dans [Bec89]. Ces derniers proposent de remplacer la droite verticale de test par des 'droites aléatoires de distribution isotrope'. Leurs coefficients optimaux restent assez proches de ceux de Vossepoel, dont les masques ne sont pas remis en cause.

Verwer propose de porter l'optimisation sur un cercle euclidien [Ver91]. Il minimise successivement les critères de l'erreur maximale et de la variance. Sa démarche mathématique est radicalement différente des précédentes. Elle possède l'avantage indéniable de fonctionner en dimension n et dans des voisinages de côté m quelconque.

Il commence par analyser le critère d'isotropie, sur lequel nous reviendrons au §4.3.1. Il justifie de la sorte l'emploi du cercle pour normaliser l'erreur. Son étude est effectuée dans le continu, ayant bien argumenté le passage entre distance de chanfrein discrète, et distance de chanfrein continue. Cela lui permet de restreindre ses calculs d'un cercle discret de rayon infini au cercle euclidien unité.

Pour le calcul de l'erreur maximale absolue, il cherche les valeurs minimales et maximales de l'erreur. Les minima de l'erreur sont situés sur les vecteurs pondérés, et il suffit donc de chercher leur minimum global. Les maxima sont atteints dans les cônes. Il calcule leurs valeurs par la méthode des multiplicateurs de Lagrange. Le maximum de l'erreur est le maximum global. Lorsque les coefficients optimaux sont déterminés, des approximations entières sont recherchées.

Verwer utilise le support du cercle pour optimiser et calculer l'erreur. C'est encore le cas dans [Bec92] et [Coq93]. Ces auteurs jouent également sur un autre élément, le facteur d'échelle, qui fait l'objet de la section 4.2.4.

4.2.4 Facteur d'échelle

L'emploi d'un facteur d'échelle ε est indispensable pour comparer une distance de chanfrein $\frac{d_C}{\varepsilon}$ à la distance euclidienne d_E.

En général on fixe $\varepsilon = a$, car ce principe permet une définition cohérente des intervalles de niveaux sur une image de distance. Sinon, les alternatives possibles sont un facteur d'échelle entier $\varepsilon \neq a$ [Coq93], rationnel [Bec92] ou réel [Ver91].

Une fonction d'erreur possède un minimum global et un maximum global. La valeur absolue de ces grandeurs est le *taux* d'erreur, leur différence est l'*amplitude*. En jouant sur le facteur d'échelle, on peut ramener le taux d'erreur à la moitié de l'amplitude.

Vossepoel est le premier à exploiter ce concept [Vos88], et suggère un processus itératif pour trouver ε. En réponse, Borgefors calcule analytiquement les ε optimaux pour les masques 3×3 et 5×5 [Bor91a], en complétant son approche originelle (§4.2.2).

Les trois papiers [Ver91, Bec92, Coq93] ont en commun le support de calcul de l'erreur, un arc de cercle, mais exposent des méthodes pour le critère de l'erreur maximale qui sont très différentes, et développées indépendamment les unes des autres.

Verwer donne les taux optimaux théoriques en 2D et 3D [Ver91] pour des voisinages de côté 3 à 25. En 2D, il propose une formulation compacte du taux d'erreur, après avoir constaté que dans un voisinage $m \times m$, l'erreur est la plus grande dans le cône de plus grand angle ψ. Ce cône est toujours celui délimité par les vecteurs $(1, 0)$ et $(\frac{1}{2}(m-1), 1)$. Le taux d'erreur optimal théorique τ est une fonction de ψ :

$$\tau = \frac{1 - \cos\frac{1}{2}\psi}{1 + \cos\frac{1}{2}\psi} \qquad \text{avec } \psi = \operatorname{atan}\frac{2}{m-1}\,. \tag{27}$$

Par exemple pour $m = 3$, $\tau = 3.95\%$ et pour $m = 5$, $\tau = 1.35\%$. Verwer montre que ces taux sont atteints avec les coefficients optimaux $1, \sqrt{2}$ et $1, \sqrt{2}, \sqrt{5}$ pour des facteurs d'échelle de 1.0412 et 1.0137. Les valeurs des coefficients optimaux correspondent aux déplacements euclidiens, ce qui n'est pas paradoxal par rapport aux coefficients de Borgefors [Bor86a], car les facteurs d'échelle ont varié.

Au niveau des approximations entières, dans le voisinage 3×3, ce sont les masques (2, 3), (5, 7) et (12, 17) qui sont préférés. Dans le voisinage 5×5, le masque ($a = 5$, $b = 7$, $c = 11$, $\varepsilon = 5.0092$, $\tau = 1.79\%$) est dépassé par ($a = 17$, $b = 24$, $c = 38$, $\varepsilon = 17.2174$, $\tau = 1.43\%$), mais il représente toujours un bon rapport qualité / coût de stockage. Enfin dans le voisinage 7×7, le masque ($a = 12$, $b = 17$, $c = 27$, $d = 38$, $\varepsilon = 12.1033$, $\tau = 0.85\%$) est très intéressant, car il est régulier et son taux est la moitié de celui de (5, 7, 11), avec seulement un poids de plus.

La méthode de Becker est simple et élégante. Elle s'appuie sur un modèle continu, dans lequel s'inscrit le cas discret [Bec92]. Il montre que toute l'étude peut se faire dans le cadre classique des distances de jauge (cf §3.4).

Le support de test est à nouveau le cercle unité, qui garantit l'isotropie du raisonnement. Becker revient explicitement aux cônes et utilise les suites de Farey, par extension de [Mon68]. Il illustre toute sa démarche dans le voisinage 7×7.

La distance de Becker est une fonction continue affine par morceaux facile à manipuler. Il montre comme Verwer que le maximum de l'erreur se situe toujours au niveau du premier cône. Il propose donc de diviser l'erreur par 2 en la répartissant de manière symétrique dans ce cône. Au lieu de changer directement le facteur d'échelle, il translate juste une partie de la fonction d'erreur en y ajoutant une fonction affine, ce qui revient à une déformation géométrique de la boule convexe.

Le taux optimal prévu par Becker dans le voisinage 7×7 est de 0.66%, comme Verwer. Après quelques ajustements numériques pour obtenir une solution entière, Becker retient le masque ($a = 47$, $b = 67$, $c = 106$, $d = 149$, $e = 170$, $\varepsilon = \frac{142}{3}$, $\tau = 0.72\%$), qui est légèrement meilleur que la proposition de Verwer, et avec un facteur d'échelle rationnel, mais avec un poids supplémentaire et un coût de stockage plus grand.

En dernier lieu, Coquin et Bolon présentent l'adaptation littérale de la méthode de Borgefors à une trajectoire circulaire [Coq93], et optimisent un masque 5×5.

Les calculs sont effectués en maillage rectangulaire (voir aussi [Bol92]), dont le maillage carré est un cas particulier. Le nombre de cas à traiter se dédouble puisque le masque n'est plus symétrique dans ce cas général.

Les coefficients optimaux déduits par Coquin et Bolon dans la maille carrée sont très proches de ceux de Borgefors. L'erreur n'est plus proportionnelle à M (sur la droite $x = M$) mais à R (sur le cercle de rayon R). Le critère optimisé en fin de compte est

$$\frac{d_C - R}{R} \tag{28}$$

et les taux optimaux d'erreur sont les mêmes que ceux donnés par Verwer, calculables directement par (27).

L'approximation entière est effectuée avec le choix d'un facteur d'échelle. Le masque proposé[1] est ($a = 73$, $b = 104$, $c = 163$, $\varepsilon = 74$, $\tau = 1.49\%$), ce qui le situe entre les deux masques 5×5 cités plus haut pour Verwer. Le facteur d'échelle est cette fois entier, mais le coût de stockage est élevé.

L'expérience nous a montré que lorsqu'on optimise le taux d'erreur avec $\varepsilon = a$, les petites valeurs de a sont à l'honneur, alors que pour l'optimisation de l'amplitude, ce sont les valeurs plus élevées de a qui prennent le pas (cf §4.4.4). En effet, dès que a croît, les coefficients tendent vers les déplacements euclidiens, ce qui profite d'abord à l'amplitude de l'erreur.

Le facteur d'échelle permet de ramener l'erreur à la moitié de l'amplitude ; dans ces conditions, c'est l'amplitude qui est vraiment optimisée, et si le facteur d'échelle est libre, les pondérations finalement intéressantes sont automatiquement portées à être plus grandes. Comme il est plus difficile de trouver un facteur $\varepsilon \neq a$ entier qui centre l'erreur que de calculer directement un ε réel, ce phénomène est accru.

4.3 Méthode proposée

4.3.1 La bonne erreur relative

Dans [Bor86a] et [Vos88], les critères du maximum de l'erreur et de la variance ont été optimisés. Ni l'un ni l'autre n'ont été normalisés par la longueur euclidienne, mais par M dans l'équation (23). Ce sont donc des mesures absolues, non relatives de l'erreur par rapport à la distance euclidienne.

Ces deux méthodes exploitent la droite verticale $x = M$. Comme la distance aux points de cette droite à 45° est supérieure de celle à 0°, l'erreur commise dans la

[1] Ceci est un *erratum* de l'article original.

direction diagonale a une influence exagérée par rapport à l'erreur commise dans d'autres directions. *Pour préserver l'isotropie, il faut donc normaliser l'erreur par la distance euclidienne.*

Notre analyse coïncide exactement avec celle de Verwer [Ver91], qui a choisi de normaliser l'erreur (28) en se plaçant sur le cercle euclidien. Nous allons normaliser l'erreur sur une droite verticale par (29), en reprenant les calculs de Borgefors rappelés au §4.2.2, puis nous verrons quels sont les avantages de notre approche.

Le point de départ est (16), dans le cône α, β, où nous cherchons à minimiser

$$G(y) = \frac{\frac{1}{\varepsilon}\, d_C - d_E}{d_E} \tag{29}$$

avec $d_E = \sqrt{M^2 + y^2}$ et ε le facteur d'échelle. La dérivée de $G(y)$ est

$$G'(y) = \frac{1}{\varepsilon}\, \frac{\sqrt{M^2+y^2}\, S - \frac{y}{\sqrt{M^2+y^2}}\, (y\, S + M\, T)}{M^2 + y^2} \tag{30}$$

qui atteint son extremum lorsque $G'(y) = 0$, i.e pour

$$y_0 = M\, \frac{S}{T} \tag{31}$$

qui est indépendant de ε. Le maximum de (29) dans l'intervalle $\alpha\, M \leq y \leq \beta\, M$ est

$$G_2 = G(y_0) = \frac{1}{\varepsilon}\, \sqrt{S^2 + T^2} - 1 \tag{32}$$

et les valeurs aux bornes sont

$$G_1 = G(\alpha\, M) = \frac{1}{\varepsilon}\, \frac{\alpha\, S + T}{\sqrt{1+\alpha^2}} - 1\,, \quad G_3 = G(\beta\, M)\,. \tag{33}$$

On remarque immédiatement que (32) et (33) sont de la forme

$$G(\xi\, M) = \frac{1}{\varepsilon}\, H(S, T, \xi) - 1 \tag{34}$$

et de valeurs indépendantes de M, donc de la droite. La conclusion s'impose :

Lemme 4.1 *L'erreur relative sur une droite verticale coupant le cône est strictement équivalente à l'erreur relative sur un arc de cercle*[2]*, et plus généralement sur toute trajectoire intersectant le cône.*

[2] L'erreur est automatiquement relative sur un cercle (O, R) puisque $d_E = R$.

On calcule globalement à partir des G_1, G_2 et G_3 de chaque cône, l'erreur relative minimale τ_{min} et maximale τ_{max}. Le taux d'erreur est $\tau = \max(|\tau_{min}|, |\tau_{max}|)$ et l'amplitude est $\delta = |\tau_{max} - \tau_{min}|$.

Le lemme 4.1 rend ces quatre grandeurs indépendantes de la trajectoire de calcul, dès lors qu'elle intersecte bien tous les cônes.

Notre calcul de l'erreur relative fournit donc les mêmes τ (pour le même ε) que dans [Ver91, Bec92, Coq93], mais des valeurs différentes de l'erreur absolue dans [Bor86a]. Par exemple, Borgefors donne une erreur de 8.09% pour $d_{3,4}$ (avec $\varepsilon = 3$). Notre calcul avec $S = b - a = 1$, $T = a = 3$, $\varepsilon = a$ donne $G_1 = 0$, $G_3 \approx -5.72\%$ et $G_2 \approx 5.41\%$. Le maximum relatif de l'erreur est donc 5.72%, et se situe au niveau des bissectrices. C'est à cet endroit que l'absence de normalisation est la plus importante dans le calcul de Borgefors, avec un facteur de $\sqrt{2}$, et en effet on retrouve bien $5.72\sqrt{2} \approx 8.09$.

4.3.2 Changer de facteur d'échelle

Supposons connu, pour un masque donné et un facteur d'échelle ε, le taux d'erreur τ_ε. Ce dernier provient de τ_{min} ou de τ_{max}, et par (34) s'écrit $\tau_\varepsilon = \frac{1}{\varepsilon} H(S, T, \xi) - 1$. Prenons maintenant un autre facteur d'échelle ε'. Si le nouveau taux d'erreur $\tau_{\varepsilon'}$ a la même provenance parmi τ_{min} ou τ_{max}, cela ce traduit par $\tau_{\varepsilon'} = \frac{1}{\varepsilon'} H(S, T, \xi) - 1$, et on peut écrire $(\tau_\varepsilon + 1)\,\varepsilon = (\tau_{\varepsilon'} + 1)\,\varepsilon'$, d'où le théorème :

Théorème 4.1 *Le taux $\tau_{\varepsilon'}$ est déduit de τ_ε par la relation*

$$\tau_{\varepsilon'} = \frac{\varepsilon}{\varepsilon'}\,(\tau_\varepsilon + 1) - 1 \qquad (35)$$

si ε' est tel que ce soit le même extremum qui fournisse ces taux.

Le facteur d'échelle qui permet de ramener l'erreur à la moitié de l'amplitude est noté ε_{opt}. Il est défini par la relation

$$-\,\tau_{opt}\ (\text{déduit de } \tau_{min}) \ = \tau_{opt}\ (\text{déduit de } \tau_{max}) \qquad (36)$$

qui par (35) devient

$$-\,\frac{\varepsilon}{\varepsilon_{opt}}\,(\tau_{min} + 1) + 1 = \frac{\varepsilon}{\varepsilon_{opt}}\,(\tau_{max} + 1) - 1 \qquad (37)$$

et de là :

Théorème 4.2 *On calcule directement le facteur d'échelle optimal avec*

$$\varepsilon_{opt} = \varepsilon \left(\frac{\tau_{min} + \tau_{max}}{2} + 1 \right) \tag{38}$$

connaissant les extrema τ_{min} et τ_{max} du masque.

Comme exemple, pour $d_{5,7,11}$ on trouve $\tau_{min} = -1.61\%$ et $\tau_{max} = 1.98\%$. On se ramène aux chiffres de Verwer avec $\varepsilon_{opt} = 5\left(\frac{-0.0161+0.0198}{2} + 1\right) \approx 5.0092$, et $\tau_{opt} = \frac{5}{5.0092}(0.0198 + 1) - 1 \approx 1.79\%$.

Le facteur d'échelle optimal ε_{opt} permet aussi de clarifier le théorème 4.1 : le passage de ε à ε' est valide s'ils sont tous les deux plus grands ou plus petits que ε_{opt}. Comme il est toujours possible de calculer ε_{opt} par (38), puis τ_{opt} par (35), alors le passage $\varepsilon \to \varepsilon_{opt} \to \varepsilon'$ l'est aussi, avec pour la dernière étape

$$\tau_{\varepsilon'} = \begin{cases} \frac{\varepsilon_{opt}}{\varepsilon'}(\tau_{opt} + 1) - 1 & \text{si} \quad \varepsilon' < \varepsilon_{opt} , \\ 1 - \frac{\varepsilon_{opt}}{\varepsilon'}(1 - \tau_{opt}) & \text{si} \quad \varepsilon' > \varepsilon_{opt} . \end{cases} \tag{39}$$

4.3.3 Calcul de l'erreur

Etant donné un masque de taille *quelconque*, on cherche à calculer son erreur relative à la distance euclidienne, par un moyen simple et efficace. Les deux méthodes présentées ci-après vont dans ce sens. Lorsque les extrema sont connus pour un facteur d'échelle donné, on peut se ramener à tout autre facteur d'échelle avec la méthode exposée au §4.3.2.

• Dans chaque cône $\widehat{w_i, w_{i+1}}$, on calcule $S = dy_i$ et $T = dx_i$ par le théorème 3.6, puis on évalue (32) et (33) avec $\varepsilon = a$. De là sont déterminés les extrema τ_{min} et τ_{max}, le taux d'erreur τ et l'amplitude δ.

Cette méthode fonctionne uniquement pour les masques réguliers, mais permet de localiser précisément les extrema et donne les valeurs exactes.

• On génère le premier octant des distances à l'origine jusqu'à une droite verticale $x = M$. Sur chaque point $(M, y)_{(y=0..M)}$ de cette droite, on calcule l'erreur relative par (29), avec $\varepsilon = a$. On en extrait les grandeurs τ_{min}, τ_{max}, τ et δ.

Cette démarche est valable pour tous les masques induisant une distance, et permet en plus de tracer la courbe de l'erreur. Les résultats seront d'autant plus précis que M sera grand. Typiquement, on prend $M = 6! = 120$.

Les courbes d'erreur sont toujours données sous la même forme que figure 33, avec en ordonnée le taux d'erreur en %, et en abscisse le nom des pondérations délimitant les cônes sur le chemin du calcul.

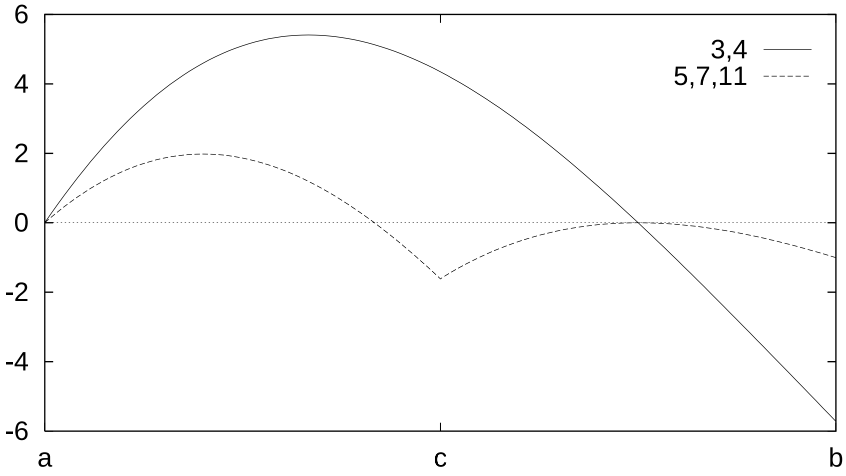


FIG. 33 - Courbes d'erreur de $d_{3,4}$ et $d_{5,7,11}$

On insiste sur l'indépendance théorique des résultats par rapport à la trajectoire, droite ou cercle, contrairement à ce qui fut dit dans [Coq93]. Dans leur article, ils calculèrent l'erreur sur un cercle 8-connexe, et annoncèrent pour le masque ($a = 73$, $b = 103$, $c = 163$, $\varepsilon = 74$) un taux $\tau = 1.42\%$. Or il se trouve que l'erreur est maximale avec 1.58% sur la bissectrice, et que leur cercle n'intersecte pas exactement celle-ci, d'où cette imprécision (en fait $b = 104$ et $\tau = 1.49\%$).

Dans la pratique discrète, il est plus précis (et facile) de tester la droite verticale que le cercle.

4.4 Nouveaux chanfreins

4.4.1 Recherche des poids

Les auteurs cités dans la section 4.2 cherchent d'abord des coefficients optimaux, puis se ramènent à des poids entiers par 'ajustement numérique', c'est-à-dire en multipliant chaque coefficient optimal par de multiples facteurs d'échelle, jusqu'à trouver une solution satisfaisante.

Nous avons vu au §4.3.3 comment calculer très simplement (et rapidement) l'erreur relative pour tout masque. Dans la même veine, nous proposons de supprimer l'étape de recherche des coefficients optimaux, et d'appliquer l'algorithme suivant :

1. Choisir un ensemble de points visibles à pondérer.
2. Pour $a = 1 \ldots 255$ faire
   (a) Pondérer avec $w_i = \text{round}(a\sqrt{x_i^2 + y_i^2})$ ou $w_i = \text{trunc}(a\sqrt{x_i^2 + y_i^2})$ les points choisis.
   (b) Vérifier que ce masque induit une distance par le théorème 3.10, sinon le rejeter.
   (c) Calculer l'erreur relative du masque (§4.3.3).
3. Trier puis afficher les masques par intérêt décroissant.

Cet algorithme est exhaustif et se base sur les erreurs *effectives*, plutôt que de chercher à approcher des coefficients optimaux. Il suppose le choix préalable d'un critère d'optimisation, et d'un ensemble de points à pondérer.

Pour le critère d'optimisation on a l'embarras du choix : (Opt1) le taux d'erreur τ_a, (Opt2) l'amplitude δ ou le taux optimal τ_{opt}, (Opt3) le taux $\tau_{\varepsilon'}$ avec $\varepsilon' = \text{round}(\varepsilon_{opt})$, suivant que l'on cherche un masque avec un facteur d'échelle $\varepsilon = a$, $\varepsilon \neq a$ entier ou réel. On prend en compte également le 'coût de stockage' du masque dont il était question au §4.1, en privilégiant les a petits.

Quant au choix des points à affecter, il dépend de la taille du voisinage. Si le voisinage est petit on retient tous les points visibles ; si le voisinage est plus grand, on peut économiser sur le coût du masque, en choisissant les points à affecter. On peut même jouer sur la géométrie de la boule, en essayant de bien répartir les angles des différents cônes, pour approximer harmonieusement le cercle euclidien. On montre au §3.5.1 les conditions pour que de tels masques soient réguliers et définissent une norme.

4.4.2 Masques 5×5 et 7×7

Coquin et Bolon ont proposé un masque 5×5 ($a = 73$, $b = 104$, $c = 163$, $\varepsilon = 74$, $\tau = 1.492\%$) avec un facteur d'échelle différent de a mais entier [Coq93]. En nous plaçant sur le même terrain, notre algorithme du §4.4.1 avec le critère (Opt3) a donné le masque ($a = 72$, $b = 102$, $c = 161$, $\varepsilon = 73$, $\tau = 1.369\%$), qui est encore meilleur, pour un coût semblable. La figure 34 montre les deux courbes d'erreur.

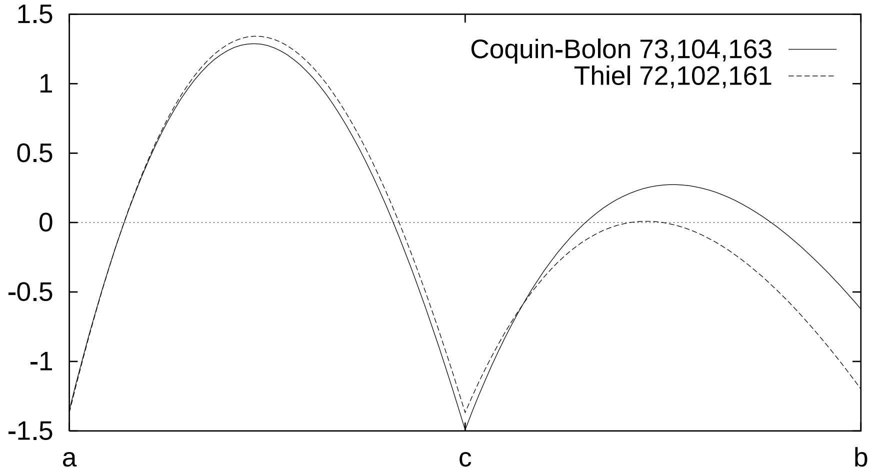


FIG. 34 - Alternative au masque de Coquin-Bolon

On constate dans la figure 35 que les deux masques sont tous les deux proches des coefficients optimaux calculés par Coquin et Bolon. Ces coefficients ne sont pas à remettre en cause. Ils sont d'ailleurs presque identiques à ceux de Verwer dans [Ver91] (mais pour le critère de la minimisation de la variance). Simplement il est difficile de déterminer *a priori*, en ne se basant que sur les coefficients optimaux, lequel des masques est vraiment le meilleur. Cela justifie notre méthode du §4.4.1, où l'étape du calcul des coefficients optimaux est supprimée.

poids	Masque Coquin-Bolon	Coeff. opt. Coquin-Bolon	Coeff. opt. Verwer	Masque Thiel
a	$\frac{73}{74} \approx 0.9865$	0.9864	0.9801	$\frac{72}{73} \approx 0.9863$
b	$\frac{104}{74} \approx 1.4054$	1.4079	1.4060	$\frac{102}{73} \approx 1.3973$
c	$\frac{163}{74} \approx 2.2027$	2.2057	2.2044	$\frac{161}{73} \approx 2.2055$

FIG. 35 - Comparaison aux coefficients optimaux de Coquin-Bolon

Becker a calculé le masque 7×7 ($a = 47$, $b = 67$, $c = 106$, $d = 149$, $e = 170$, $\varepsilon = \frac{142}{3}$, $\tau = 0.724\%$), avec un facteur d'échelle réel [Bec92]. Nous proposons le masque ($a = 62$, $b = 88$, $c = 139$, $d = 196$, $e = 224$, $\varepsilon = 62.3908$, $\tau = 0.657\%$), optimisé avec le critère (Opt2). La figure 36 montre les deux courbes d'erreur ; tout se joue dans le premier octant.

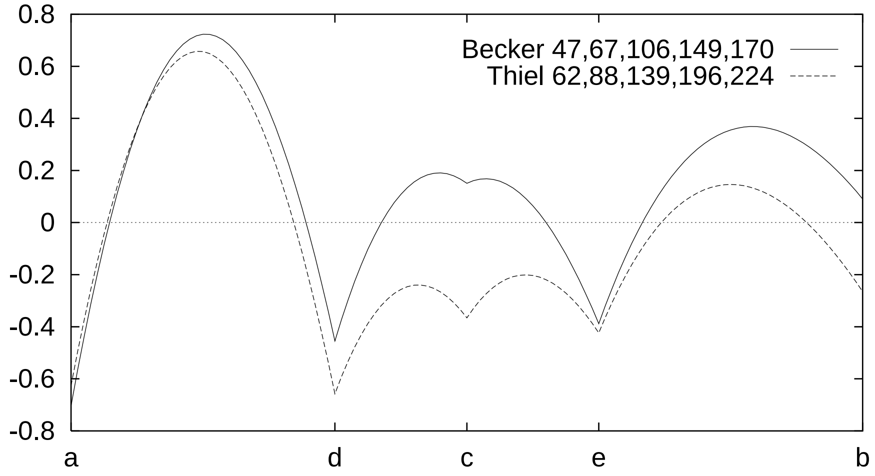


FIG. 36 - Alternative au masque de Becker

4.4.3 Grands masques

L'optimisation de grands masques ne pose aucun problème avec notre méthode. Nous donnons en exemple les résultats obtenus pour un voisinage 13×13, qui contient les points visibles de a à m (cf figure 19 au §3.1).

La première optimisation est effectuée pour le critère (Opt1), et donne le masque ($a = 68$, $b = 96$, $c = 152$, $d = 215$, $e = 245$, $f = 280$, $g = 340$, $h = 346$, $i = 366$, $j = 396$, $k = 413$, $l = 435$, $m = 531$, $\varepsilon = 68$, $\tau = 0.269\%$). La seconde optimisation est faite avec le critère (Opt2), et le résultat est ($a = 233$, $b = 330$, $c = 521$, $d = 737$, $e = 840$, $f = 961$, $g = 1165$, $h = 1188$, $i = 1255$, $j = 1359$, $k = 1417$, $l = 1492$, $m = 1820$, $\varepsilon = 233.3632$, $\tau = 0.175\%$). La figure 37 montre leurs courbes respectives.

Le taux théorique optimal pour un tel voisinage est de 0.1707% d'après [Ver91]. Le deuxième masque que l'on a donné est de performance très voisine avec $\tau = 0.175\%$.

Mais les masques de cette taille sont coûteux en temps de calcul. L'étude des courbes d'erreur laisse supposer qu'il est possible de garder les mêmes taux en supprimant certaines pondérations. Il suffit que chaque courbe d'erreur reste dans le *même intervalle*. Comme nous tenons à ce que les masques restent réguliers (§4.1), nous utilisons l'arbre des masques réguliers de la figure 30.

Les points immédiatement supprimables sont i et j. Ensuite, on a le choix entre k et m pour la prochaine suppression. Or l'erreur sur un masque complet centré est toujours la plus grande dans le premier cône (§4.2.4), donc on ne va pas toucher à k, mais supprimer m puis l. On vérifie que les deux masques proposés, privés des points i, j, m, l conservent le même taux d'erreur. Les points effectivement pondérés sont $a..h$, k, et les courbes d'erreur sont tracées figure 38.

Le gain en temps de calcul est de 33%, puisque le nombre total de points affectés dans le voisinage 13×13 passe de 96 à 64. Ce nombre est de 16 pour les masques 5×5. Pour un temps de calcul multiplié par 4, le taux d'erreur est divisé par 10.

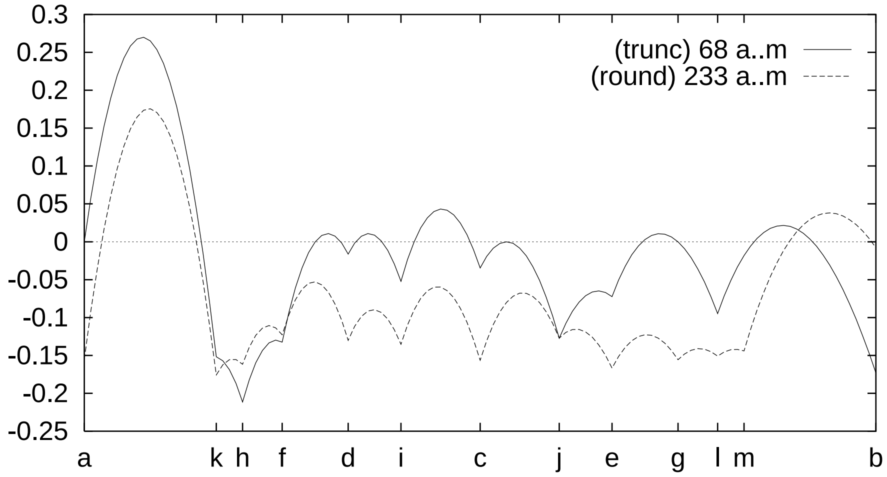


FIG. 37 - Masques optimaux complets 13×13

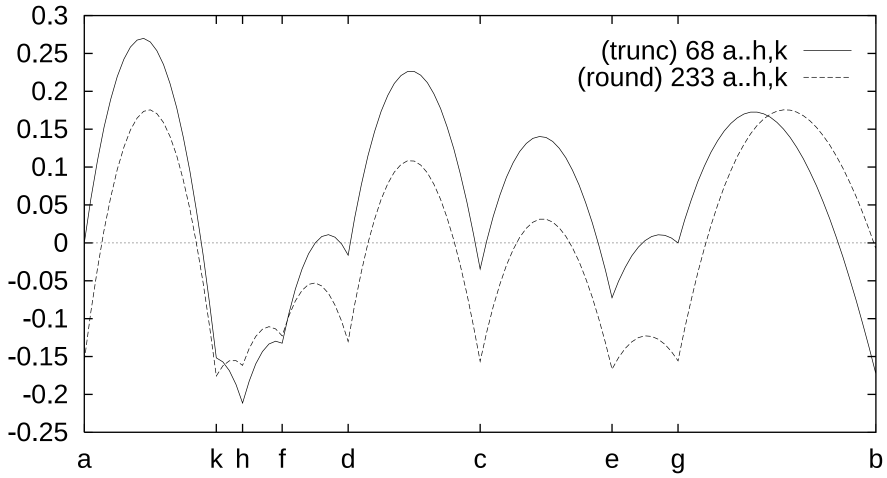


FIG. 38 - Masques optimaux 13×13, allégés mais réguliers

4.4.4 Compléments

La figure 39 vient illustrer la discussion de la fin du §4.2.4, à propos de la tendance des a à être petits ou grands, suivant le critère d'optimisation.

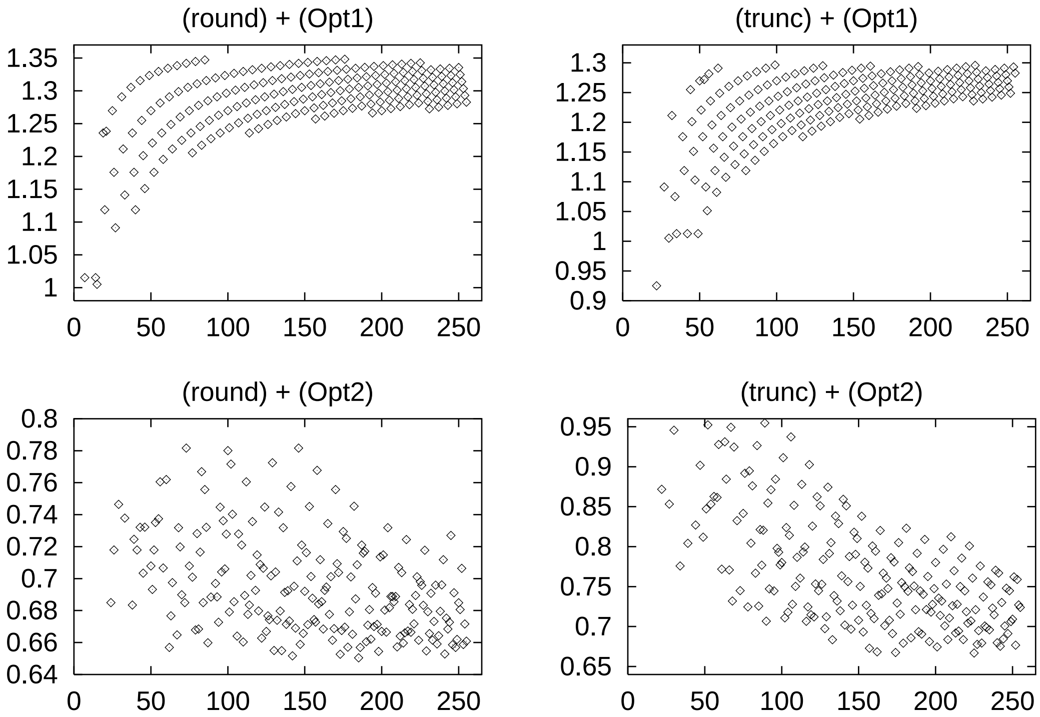


FIG. 39 - Distribution des taux d'erreur dans un masque 5×5 en fonction de a

Les taux optimaux théoriques sont donnés par la formule (27) de Verwer, et sont quasiment atteints par les masques que nous avons proposés. Nous traçons figure 40 la courbe de l'équation (27) en fonction de m ; elle se comporte comme $\frac{1}{m}$.

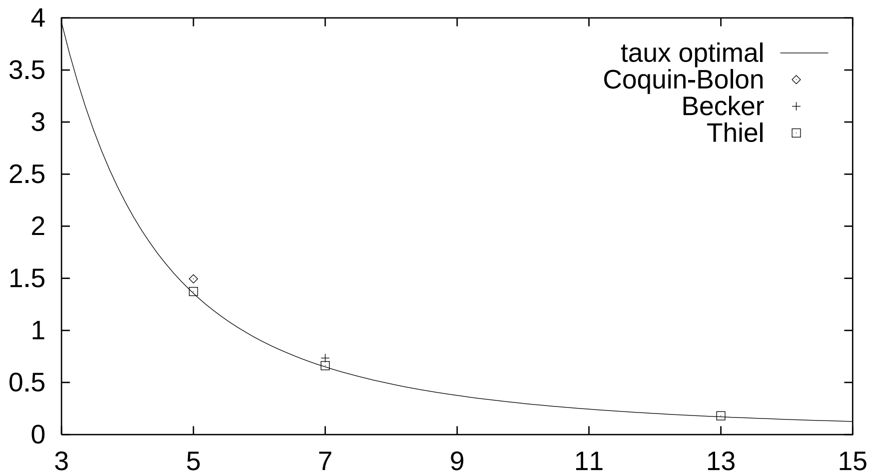


FIG. 40 - Courbe des taux optimaux théoriques

4.5 Conclusion

Les méthodes existantes de la bibliographie ont été développées parce que les approches purement locales ne sont pas satisfaisantes. La méthode de Borgefors a inauguré le domaine de l'optimisation des distances de chanfrein, avec un test sur une droite verticale. D'autres auteurs ont proposé de changer le critère d'optimisation dans le but d'améliorer l'isotropie, en prenant la variance plutôt que le maximum de l'erreur, ou en changeant le support de test, par des droites obliques ou des cercles. Ils ont aussi joué sur le facteur d'échelle pour améliorer les résultats.

Par notre approche, nous avons montré que l'erreur relative est un meilleur critère que l'erreur absolue, et que pour cette première les supports de test et les méthodes de calculs sont équivalents. Lorsque le taux d'erreur est connu pour un facteur d'échelle, on peut se ramener à toute autre échelle, y compris le facteur optimal, qui rend l'erreur optimale en centrant la courbe de l'erreur.

Notre algorithme d'optimisation est basé sur le calcul du taux *effectif* de l'erreur ; il évite le calcul de coefficients optimaux. Il génère des masques optimaux, dont la comparaison avec les masques de Becker et Coquin-Bolon valide notre démarche. De grands masques sont également calculés, pour illustrer la souplesse de la méthode, et montrer l'intérêt de la notion de masques allégés.

La distribution de l'erreur montre l'influence du critère employé sur l'ordre de grandeur des pondérations. La courbe des taux optimaux théoriques montre que l'on peut approximer très finement la distance euclidienne ; le terme de distance *quasi-euclidienne* pour les distances de chanfrein est parfaitement justifié.

L'optimisation en 3D est étudiée par les mêmes auteurs, principalement par Borgefors et Verwer. Le passage de notre approche au 3D est triviale, en particulier celui de l'algorithme ; il suffit de générer un simplexe de distances et de tester un plan vertical. Dans le maillage rectangulaire, l'optimisation est réalisée en 2D dans [Bol92, Coq93], et en 3D dans [Coq94] pour de petits voisinages.

Chapitre 5

Axe médian

L'axe médian est un concept faisant intervenir le recouvrement d'une forme par des boules, de manière à accéder à une représentation minimale de l'image. Cette notion est indispensable au calcul de la ligne médiane et du squelette pondéré. Ce chapitre est donc consacré au calcul de l'axe médian.

Les méthodes existantes sont spécifiques à chaque distance, et utilisent des critères différents. Nous proposons pour la première fois un algorithme universel et efficace de calcul de l'axe médian pour les distances de chanfrein.

Nous recensons enfin les propriétés de l'axe médian et ses applications.

5.1 Introduction

La description de formes nécessite des méthodes pour extraire les caractéristiques importantes des objets présents dans une image. Un objet binaire est un ensemble de points qui, pris séparément, ne donnent pas d'information pertinente. C'est pourquoi les objets sont fréquemment représentés par leur contour. Une alternative intéressante consiste à rechercher un 'axe de symétrie généralisé' dans la forme. Cet ancêtre du squelette fut introduit dans l'espace continu en 1964 par Blum, qui le baptisa 'axe médian' [Blu67]. Cette terminologie a ensuite évolué.

Toute forme peut être ramenée à une union de sous-ensembles de ses points, eux-mêmes codés par une position et un paramètre de taille. Ce codage permet de retrouver la forme. Pfaltz et Rosenfeld ont défini l'axe médian en termes de 'voisinages maximaux' dans l'espace discret [Pfa67]. Nous redonnons leur définition.

Etant donné une *famille de boules*, que nous préciserons plus loin, et une forme, nous considérons les boules strictement incluses dans la forme.

Définition 5.1 (Boule maximale) *Une boule est dite maximale dans la forme si elle n'est incluse dans aucune autre boule.*

Une telle boule peut cependant être incluse dans l'union de plusieurs autres (§5.6).

Définition 5.2 (Axe médian) *L'axe médian, noté AM, est le lieu des centres des boules maximales dans la forme.*

L'allure de l'axe médian et son mode de calcul dépendent de la géométrie des boules choisies, celles-ci pouvant être de toute nature. Par exemple, voici schématisé figure 41, l'axe médian recouvrant une forme par des cercles.

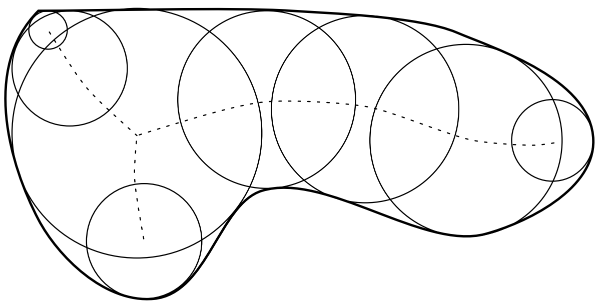


FIG. 41 - Axe médian avec des cercles

Comme autre exemple, dans [Wu86, Jen92] on cherche à recouvrir un objet discret par des rectangles de taille variable. Pour ce faire, des algorithmes spécialisés, séquentiels et parallèles, ont été développés. A partir de cette représentation, les auteurs sont en mesure de calculer aire et périmètre, et aussi de réaliser des opérations ensemblistes.

L'image de distance DM est une des premières approches pour donner une structure à une image binaire, et pour révéler ses propriétés. C'est dans ce contexte que l'axe médian prend tout son sens. En effet, les boules considérées pour une distance donnée sont les disques de différents rayons, et nous verrons comment il est possible de caractériser *localement* les centres de boules maximales sur une image de distance.

L'axe médian AM est mémorisé au niveau de la position des centres, et du rayon des boules maximales. Il est suffisant pour retrouver la forme, car il correspond à un recouvrement. L'axe médian joue donc aussi un rôle pour la compression de données [Ros82]. Le retour à la forme initiale à partir de AM est aisé, grâce à l'algorithme de transformation inverse de distance du §2.4.3. Celui-ci a été conçu pour que chaque point de l'axe médian génère le disque associé à son poids, en deux passages séquentiels globaux sur l'image.

Ce chapitre est consacré à l'extraction de l'axe médian suivant les différentes distances de chanfrein.

Rosenfeld et Pfaltz ont montré dans [Ros66] que les *maxima locaux* définis au §5.2, qui sont présents dans DM, sont nécessaires pour identifier AM. Mais si leur critère fonctionne bien pour les distances d_4 et d_8, il en va autrement pour les autres distances pondérées. Borgefors considère que les algorithmes de calcul de l'axe médian sont spécifiques à chaque distance. Elles demandent ainsi la recherche de *disques équivalents* (§5.3) pour les distances $d_{a,b}$. Dans le cas de $d_{5,7,11}$, une autre stratégie doit être adoptée, avec l'emploi de *tables de correspondance* (§5.4).

La figure 54 au chapitre 6 montre l'axe médian d'une forme calculé suivant les distances d_4, d_8, $d_{3,4}$ et $d_{5,7,11}$.

Il n'existe actuellement pas d'algorithme universel de calcul de l'axe médian, valable pour toute distance de chanfrein. Nous répondons à cette attente au §5.4.3, puis nous explorons certains points au §5.5. Le chapitre est clos par un énoncé des propriétés plus classiques au §5.6, suivi de quelques applications au §5.7.

5.2 Maxima locaux

Considérons un masque de chanfrein (x_i, y_i, w_i). Dans la carte de distance, chaque point p reçoit l'information de distance de l'un au moins de ses voisins $n_i(p)$ (sur le masque). Tout voisin de ce type est placé sur un chemin minimal, constitué uniquement de déplacements permis du fond à p. De même, le point p peut propager l'information de distance à certains voisins, qui sont plus internes dans l'objet que p. Lorsqu'une telle propagation a lieu de p à un voisin $n_i(p)$, alors $n_i(p) = p + w_i$.

Définition 5.3 (Maximum local) *On appelle maximum local un point p qui ne propage l'information de distance à aucun de ses voisins, c'est-à-dire*

$$n_i(p) < p + w_i \quad \forall i\,. \tag{40}$$

Un maximum local p peut avoir des voisins plus grands que lui, mais ils reçoivent leur poids depuis d'autres points, plus petits que p. Aux origines de cette appellation, seules d_4 et d_8 étaient considérées, et un tel point était un vrai extremum. Ce n'est plus le cas lorsque $a \neq 1$ dans le masque, mais l'appellation est conservée.

Pour d_4 et d_8, les maxima locaux coïncident exactement avec l'axe médian [Ros66]. Une illustration est donnée figure 42. De ce fait on peut définir un Critère de Centre Maximal noté CCM, qui sélectionne tous les maxima locaux de DM.

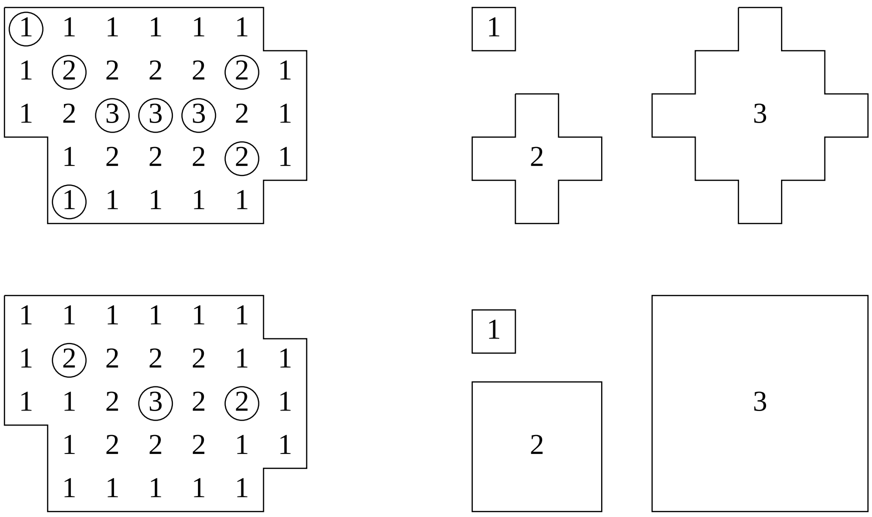


FIG. 42 - DM et AM (cerclé) pour d_4 (en haut) et d_8 (en bas) ; allure des boules.

Le CCM n'est plus exact dès que $a \neq 1$ dans le masque, car il peut sélectionner en plus de AM, des centres de boules non maximales. Ces points erronés sont encore appelés *points redondants*, ou *pseudo AM*. Un exemple est donné pour $d_{3,4}$ figure 43, dans lequel le CCM fournit de nombreux points redondants ; toutes les boules de rayon 3, et certaines boules de rayon 6, sont complètement recouvertes par des boules voisines.

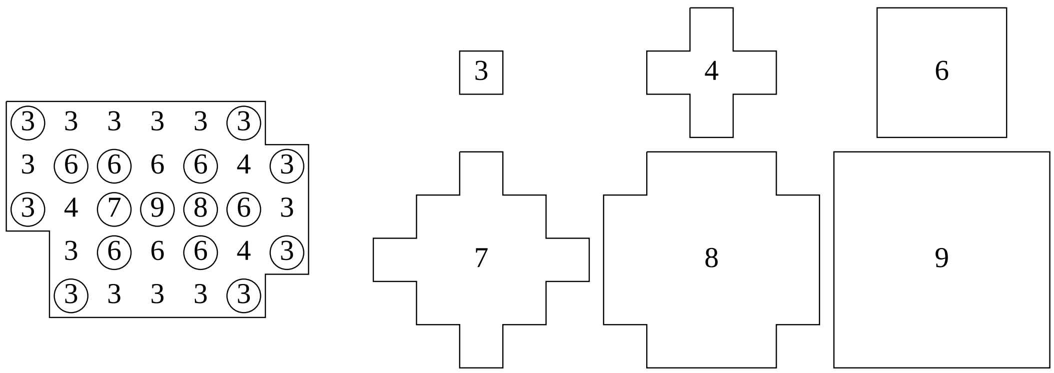


FIG. 43 - DM et AM inexact (cerclé) pour $d_{3,4}$; allure des boules.

La présence des points redondants est très gênante, car elle est susceptible de nuire irrémédiablement à la propagation de la ligne médiane.

5.3 Disques équivalents

Il est important d'analyser les valeurs possibles et impossibles dans une image de distance, à partir de l'ensemble des pondérations du masque. Ce problème est cité dans la littérature sous le nom de *problème de Frobenius* [Huj87, Ser93].

Nous consacrons cette section au cas d'un masque (a, b). Le problème de Frobenius à deux variables est résolu dans [Syl84].

Théorème 5.1 (Sylvester, 1884) *Soit $0 < a < b$ deux nombres premiers entre eux et posons $\chi = (a-1)(b-1)$, que l'on appelle* conducteur *de a et b. On considère l'équation*

$$ax + by = t, \quad (x, y) \in \mathbb{N}^2.$$

1. *Si $t \geq \chi$, cette équation admet toujours au moins une solution ;*
2. *si $t = \chi - 1$, cette équation n'a pas de solution ;*
3. *il y a exactement $\frac{1}{2}\chi$ valeurs de t pour lesquelles il n'y a pas de solution.*

Par exemple avec (3, 4), on a $\chi = 6$; on peut atteindre $\{3, 4\} \cup [6, +\infty[$, mais pas les valeurs 1, 2 et 5, qui sont dites *non représentables*.

On constate que ce sont certaines valeurs inférieures au conducteur qui mettent à mal le CCM, plus exactement les labels précédés de valeurs non représentables. Leur existence est liée à la notion de disque équivalent, introduite dans [Arc87]. On dit que deux disques sont équivalents si ils recouvrent les mêmes pixels, indépendamment des valeurs de ces pixels.

Théorème 5.2 (Disques équivalents) *Soit $0 < a < b$ deux entiers, et $q < p$ deux termes successifs de la suite des entiers représentables par a et b. Alors la classe d'équivalence du disque associé à p est l'intervalle $[q+1, p]$. On note $\dot{p} = q+1$ le plus petit représentant de la classe.*

Démonstration : On obtient le disque $B(p)$ associé à p avec l'algorithme de transformation inverse du §2.4.3 appliqué à p. Les valeurs présentes dans $B(p)$ sont de la forme $p - (ax + by) > 0$. Par construction on a $B(p-1) \subseteq B(p)$. On a aussi

$$B(p-1) \subsetneq B(p) \iff \exists\,(x, y) \in \mathbb{N}^2 \;/\; p - (ax + by) = 1$$

puisque les points à 1 dans $B(p)$ sont à 0 pour $p - 1$. On peut donc écrire

$$\begin{aligned} p - 1 \text{ est représentable} &\iff \exists\,(x', y') \in \mathbb{N}^2 \;/\; p - 1 = ax' + by' \\ &\iff \exists\,(x', y') \in \mathbb{N}^2 \;/\; p - (ax' + by') = 1 \\ &\iff B(p-1) \subsetneq B(p) \end{aligned}$$

d'où à l'inverse, $B(p-1) \equiv B(p)$ si $p - 1$ est non représentable. □

Dans les exemples suivants on écrit les entiers de 1 à $\chi(a, b)$, on souligne les entiers représentables, et on place entre crochets les classes d'équivalences.

• Dans le cas de $\chi(3, 4) = 6$, la suite $[1, 2, \underline{3}]\ [\underline{4}]\ [5, \underline{6}]$ correspond bien à la figure 44.

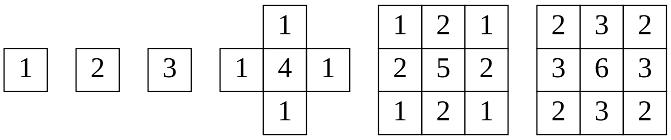


FIG. 44 - Disques équivalents pour $d_{3,4}$

• Notre second exemple est $\chi(5,7) = 24$, pour lequel la suite

$$[1,2,3,4,\underline{5}]\ [6,\underline{7}]\ [8,9,\underline{10}]\ [11,\underline{12}]\ [13,\underline{14}]\ [\underline{15}]\ [16,\underline{17}]\ [18,\underline{19}]\ [\underline{20}]\ [\underline{21}]\ [\underline{22}]\ [23,\underline{24}]$$

est conforme à la figure 45, dans laquelle nous montrons seulement les octants.

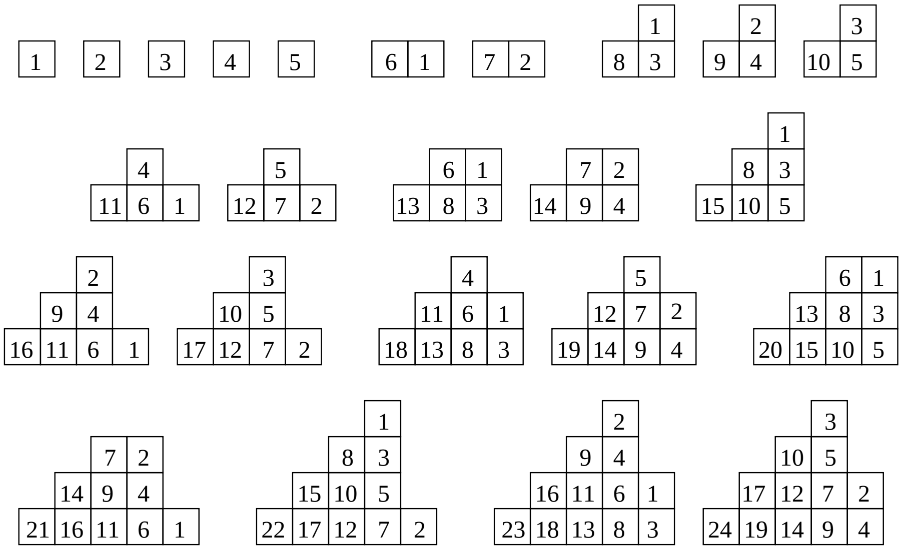


FIG. 45 - Octants équivalents pour $d_{5,7}$

Arcelli et Sanniti di Baja ont montré que dans le cas d'une distance $d_{a,b}$, il suffit d'abaisser chaque label p à la valeur correspondante $\dot{p}$ sur l'image de distance ; on obtient une nouvelle image, sur laquelle le CCM est exact [Arc87, Arc88].

1	1	1	1	1	1	
1	5	5	5	5	4	1
1	4	7	9	8	5	1
	1	5	5	5	4	1
	1	1	1	1	1	

FIG. 46 - Image de disques équivalents et axe médian exact (cerclé) pour $d_{3,4}$

Pour d_4 et d_8, $a = 1$ donc $\chi = 0$ et aucun label n'est changé. Dans le cas de $d_{3,4}$, on effectue $\dot{6} = 5$ et $\dot{3} = 1$ [Arc86], et on obtient l'axe médian exact figure 46. De même, pour $d_{5,7}$ on a $\dot{5} = 1$, $\dot{7} = 6$, $\dot{10} = 8$, $\dot{12} = 11$, $\dot{14} = 13$, $\dot{17} = 16$, $\dot{19} = 18$, $\dot{24} = 23$. Etant donné que $\forall p > \chi(a,b)$, $\dot{p} = p$, on a intérêt à choisir des pondérations a et b petites et premières entre elles, de telle sorte que $\chi(a,b)$ soit fini et le plus petit possible, pour limiter la quantité de labels à abaisser.

5.4 Tables de correspondance

5.4.1 Cheminement

Dans l'ensemble des masques de chanfrein, les masques 3×3 forment un cas particulier, en vertu du fait qu'ils ne définissent qu'un seul cône dans le premier octant. Cette propriété biaise souvent les généralisations que l'on pourrait attendre de l'étude de $d_{a,b}$; il en va ainsi pour la méthode des disques équivalents.

Prenons le cas de $d_{5,7,11}$, qui va bien illustrer les problèmes de la généralisation. Cette distance possède les cônes $\widehat{5,11}$ et $\widehat{11,7}$, pour lesquels on a $\chi(5,11) = 40$ et $\chi(11,7) = 60$. En appliquant le théorème 5.2 non plus aux disques mais à chaque cône, dans $\widehat{5,11}$ on trouve $\dot{5} = 1$, $\dot{10} = 6$, $\dot{15} = 12$, $\dot{20} = 17$, $\dot{25} = 23$, $\dot{30} = 28$, $\dot{35} = 34$ et $\dot{40} = 39$, tandis que dans $\widehat{11,7}$ les équivalences sont $\dot{7} = 1$, $\dot{11} = 8$, $\dot{14} = 12$, $\dot{18} = 15$, $\dot{21} = 19$, $\dot{25} = 23$, $\dot{28} = 26$, $\dot{32} = 30$, $\dot{35} = 34$, $\dot{39} = 37$, $\dot{42} = 41$, $\dot{46} = 45$, $\dot{49} = 48$, $\dot{53} = 52$ et $\dot{60} = 59$.

On peut imaginer d'appliquer les équivalences suivant le déplacement testé, mais que dire du cas de 11, qui intervient dans les deux cônes ? On peut encore essayer de déterminer de quel cône un point p a le plus de chance d'avoir des centres dont les boules recouvrent son disque, puis d'appliquer l'équivalence correspondante. Une dernière idée consiste à chercher les disques équivalents à partir des classes dans les deux cônes. On obtient $\dot{5} = 1$, $\dot{7} = 6$, $\dot{10} = 8$, $\dot{14} = 12$, $\dot{18} = 17$, $\dot{20} = 19$, $\dot{25} = 23$ et $\dot{35} = 34$.

On trouve toujours des contre-exemples à ce genre de méthodes, qui n'apportent au mieux qu'une solution approchée ; il faut donc explorer des voies différentes.

Arcelli et Frucci utilisent dans[Arc92] un axe médian avec $d_{5,7,11}$; mais ils ne donnent aucun détail sur la façon dont ils le calculent.

La voie choisie par Nacken [Nac94] est issue de la morphologie mathématique. Son raisonnement le conduit à distinguer les boules ouvertes des boules fermées, les images de distance 'externes' des 'internes', puis les classes d'entiers appartenant à $5\mathbb{N} + 11\mathbb{N}$ ou à $11\mathbb{N} + 7\mathbb{N}$. Son algorithme final se fait en 4 passages : 2 pour l'image de distance, 1 pour abaisser les labels en image de distance 'interne', puis 1 passage d'extraction de l'axe médian, avec des tests spécifiques à chaque type de voisin. La démarche pour aboutir à cet algorithme est assez complexe, et ne laisse pas prévoir une extension immédiate à des masques plus grands.

Nous retenons la solution la plus générale et la plus simple, qui est la méthode des *tables de correspondance.*

5.4.2 Principe

Considérons un masque de chanfrein (x_i, y_i, w_i) définissant une distance d_C, et une image de distance DM calculée sur une forme X. On peut déterminer si un point $p \in X$ est un centre maximal, en recherchant dans son voisinage de pondération $n_i(p)$ si il existe un point $q \in X$, dont le disque recouvre complètement celui de p. Dans ce cas, la présence de q *interdit* à p d'appartenir à l'axe médian.

Supposons que pour toute valeur de p, on connaisse les valeurs *minimales* de ces voisins q, que l'on stocke chacune dans $LUT_i(p)$. Alors

$$p \in AM \iff \forall i,\ n_i(p) < LUT_i(p)\,. \tag{41}$$

On appelle ces LUT_i les *tables de correspondance* de la distance d_C. Leur application par (41) est on ne peut plus simple et rapide ; le problème réside dans le précalcul des tables, que l'on aborde dans la suite.

La première apparition notable des LUT est due à Borgefors, Ragnemalm et Sanniti di Baja dans [Bor91a] pour la distance euclidienne. L'approche par les tables de correspondance semble être la seule opérationnelle pour d_E. Deux LUT indexées sur ${d_E}^2 \in \mathbb{N}$ sont employées, pour les voisins directs et indirects. Les tables sont élaborées par des tests systématiques ; la combinatoire est énorme, mais les calculs sont effectués une fois pour toutes. Les auteurs donnent ainsi les LUT pour des boules de rayon inférieur à $\sqrt{80}$.

Borgefors propose dans [Bor93] une caractérisation de l'axe médian pour la distance $d_{5,7,11}$. Trois LUT sont données pour les voisins de type a, b et c. Les entrées dans les tables sont limitées aux valeurs possibles dans une DM, et bornées par 60, qui est le plus grand des deux conducteurs $\chi(5, 11) = 40$ et $\chi(11, 7) = 60$. Au delà, le CCM est valide puisque chaque disque est unique, tandis qu'en deçà, l'existence de labels impossibles implique des configurations particulières de recouvrement, qui justifient l'emploi des LUT.

La taille des tables est faible, tout en assurant le calcul de AM pour des objets d'épaisseur quelconque. Mais Borgefors ne dit pas comment retrouver ses tables, ni comment procéder avec d'autres masques.

Indépendamment de Borgefors, nous avons proposé dans [Thi94d, Thi94b], des tables de correspondance totalement identiques pour $d_{5,7,11}$; nous les redonnons figure 47. Les cases vides et les entrées non représentées dans les tables correspondent au cas où le CCM est exact.

p	a	b	c
5	7	10	14
7	11		
10	14	15	20
14	18	20	
16		22	
18	22		28
20		26	30
21		27	

p	a	b	c
25	28	30	35
27		33	
29	33		
31		37	
32		38	
35	39	41	45
38		44	
39		45	

p	a	b
40	44	
42		48
46		52
49		55
53		59
60		66

FIG. 47 - Table de correspondance de $d_{5,7,11}$

L'application de ces tables à l'extraction de l'axe médian est illustrée figure 48. La méthode de calcul de ces tables est exposée au §5.4.3 ; elle a été conçue pour fonctionner avec des masques de toutes tailles.

					5	5	5		
5	5			5	7	10	7	5	
5	7	5	5	7	11	14	11	7	5
5	7	10	10	11	14	18	14	10	5
	5	5	5	7	11	14	11	7	5
				5	10	10	7	5	
				5	5	5	5		

FIG. 48 - Axe médian exact (cerclé) pour $d_{5,7,11}$

5.4.3 Algorithme

Le calcul d'une entrée dans la table de correspondance pour une boule de rayon r dans la direction n_i consiste à rechercher le plus petit disque de rayon r' centré en n_i qui le recouvre complètement. Un disque de rayon r est obtenu par la transformation inverse du §2.4.3 sur un point étiqueté à r. On note ce disque D_r^{-1}.

Il est possible de déterminer r' en testant différentes valeurs, ce qui demande à chaque fois une transformation inverse. Au niveau de la *LUT* entière, cette méthode exhaustive est extrêmement coûteuse.

Notons $D_r(p) = \{q \,|\, d_C(p,q) \leq r\}$. Comme $D_r^{-1}(p) = \{q \,|\, r - d_C(p,q) > 0\}$ on a immédiatement le lemme suivant :

Lemme 5.1 $D_r = D_{r+1}^{-1}$.

Cette simple constatation est à la base de tout l'algorithme qui va suivre : en effet il suffit de calculer l'octant des distances à l'origine *une seule fois* au départ de l'algorithme ; le lemme 5.1 permet alors de tester les recouvrements de boules sur cet octant. Le calcul de l'octant est effectué en un passage séquentiel.

La seconde astuce que nous employons est de mettre toutes les *LUT* à jour au fur et à mesure en *un seul parcours* de l'octant des distances. La troisième idée est d'économiser la mémoire en passant par un index des valeurs possibles.

L'algorithme est présenté en 8 phases. Toutes les variables sont des entiers ou des tableaux d'entiers. L'image originale contenant *DT* est accédée par `image1[x,y]`. On calcule d'abord les *LUT* dans `image2`, qui contient ensuite l'axe médian. Les images sont de taille `xsiz * ysiz`. On gère un `bord` vide autour des images, qui vaut au moins le rayon du masque. Le masque $(x_i,\ y_i,\ w_i)$ est stocké dans les tableaux `mx[i]`, `my[i]`, `mw[i]`, de telle sorte que les `mn1` premiers points soient ceux du premier octant, et que le nombre total de points (déduits par symétries) soit `mn8`.

1) Initialisations

L'entier `rayon` est le rayon du plus grand disque (par excès) que l'on puisse trouver dans une image de cette taille. Le poids a est assigné à `mA`.

```
rayon     = min (xsiz, ysiz) / 2 + 2;
max_index = rayon * mA;
max_mini  = (rayon + bord) * mA;
index     = allouer_memoire (max_index);
for (i = 0; i < max_index; i++)
    index[i] = FAUX;
```

2) Construction de l'octant des distances

L'octant des distances inférieures à `max_mini` est calculé (a), et l'index des valeurs possibles `index` est rempli (b). Le point origine de l'octant est `[bord,bord]`.

```
for (y = 0; y < ysiz; y++) for (x = 0; x < xsiz; x++)
    image2[x,y] = INFINI;
image2[bord,bord] = 0;

for (x = bord+1; x < bord + rayon + bord; x++)
for (y = bord; y <= x; y++)
{
    mini = image2[x-mx[0],y-my[0]] + mw[0];
    for (i = 1; i < mn1; i++)
    {
        v = image2[x-mx[i],y-my[i]] + mw[i];
        if (v < mini) mini = v;
    }
    if (mini > max_mini) break;       /* sort de ‘for y’ */
    image2[x,y] = mini;                          /* 2.a) */
    if (mini < max_index) index[mini] = VRAI;    /* 2.b) */
}
```

3) Disques équivalents

On remplit l'index par les disques équivalents. Cette phase est nécessaire car on va tester `image2 + 1`, ce qui nous fait sortir des valeurs possibles. Au niveau du résultat, on peut retrouver si `i` est une valeur possible avec `index[i] < index[i+1]`. Durant cette phase, on calcule aussi la taille `max_lut` des *LUT*.

```
index[0] = VRAI; max_lut = 0;
for (i = 0; i < max_index; i++)
    if (index[i])
         index[i] = max_lut++;
    else index[i] = max_lut;
max_lut++;
```

4) Créer les *LUT*

On alloue la mémoire pour les *LUT*, puis on les initialise à 0. Lorsque `mA` est grand, de nombreuses valeurs sont impossibles dans l'octant, et `max_lut` est très inférieur à `max_index` (d'où les économies importantes de mémoire).

```
for (i = 0; i < mn1; i++)
{
    lut[i] = allouer_memoire (max_lut);
    for (t = 0; t < max_lut; t++) lut[i][t] = 0;
}
```

5) Remplir les ***LUT***

On parcourt l'octant colonne à colonne, et on met à jour les *LUT*. Le but est que D^{-1}(`lut[i][index[r]]`) soit le plus petit disque contenant D^{-1}(`r`) dans la direction `i`. Le lemme 5.1 nous amène à tester `image2 + 1`.

```
for (x = bord; x < bord + rayon; x++)
for (y = bord; y <= x; y++)
{
    t2 = image2[x,y] + 1;
    if (t2 < max_index)
    {
        it2 = index[t2];
        for (i = 0; i < mn1; i++) lut[i][it2] =
            max (lut[i][it2], image2[x+mx[i],y+my[i]] + 1);
    }
    else break;  /* sort de 'for y' */
}
```

6) Corriger les inclusions

Le test de la phase (5) détecte les recouvrements maximaux en chaque point du bord des disques. Pour passer du niveau local au niveau global, il faut encore corriger certaines de ces inclusions, ce qui est fait simplement en rétablissant l'ordre croissant dans les *LUT*.

```
for (i = 0; i < mn1; i++)
{
    v = 0;
    for (t = 1; t < max_lut; t++)
        if (lut[i][t] > v)
             v = lut[i][t];
        else lut[i][t] = v;
}
```

7) Extraction de l'axe médian

Les tables de correspondance sont maintenant complètes, et on peut calculer l'axe médian dans `image2`. Les *LUT* sont stockées pour le premier octant, et on se ramène par 8-symétries à tout le masque. Le tableau `mg[i]` donne le numéro de *LUT* inférieur à `mn1` qu'il faut appliquer dans la direction `i`.

```
for (y = 0; y < ysiz; y++) for (x = 0; x < xsiz; x++)
    if (image1[x,y] == 0 or image1[x,y] >= max_index)
        image2[x,y] = image1[x,y];
    else {
        test = VRAI; it1 = index[image1[x,y]];
        for (i = 0; i < mn8; i++)
            if (image1[x+mx[i],y+my[i]] >= lut[mg[i]][it1])
             { test = FAUX; break; }  /* sort de 'for i' */
        image2[x,y] = (test) ? image1[x,y] : 0;
    }
```

8) Destruction des *LUT*

```
for (i = 0; i < mn1; i++) liberer_memoire (lut[i]);
liberer_memoire (index);
```

5.4.4 Commentaires

L'octant de distance est construit par un passage séquentiel sur l'image, avec un huitième de masque. Le calcul des *LUT* est effectué en un seul passage sur l'octant, auquel il faut ajouter quelques parcours linéaires des *LUT*. Cet algorithme est donc particulièrement efficace.

La méthode employée ne suppose aucune condition sur la distance de chanfrein employée, et est donc valable quel que soit le masque. L'algorithme choisit la taille des *LUT* en fonction de la dimension des images, et non en fonction de conducteurs (cf §5.3). Les tables ne sont pas stockées mais recalculées à chaque fois. Il n'est donc point besoin de rechercher des masques dont les pondérations sont premières entre elles par cône.

Les valeurs données par les *LUT* sont les plus petits rayons des disques assurant le recouvrement, autrement dit les valeurs minimales dans les disques équivalents. Pour revenir à des tables ne recensant que les valeurs possibles dans une image de distance, il faut remplacer les bornes inférieures des classes par leurs bornes supérieures (c'est ce qu'on a fait pour produire la figure 47). Cette manipulation est nécessaire pour déterminer quels sont les points qui diffèrent du CCM (§5.2), mais n'est d'aucune utilité dans l'algorithme.

La figure 49 montre les premières valeurs des *LUT* pour quelques distances.

p	a	b
3	4	5
4	7	8
6	8	9
7	10	11
8	11	12
9	12	13
10	13	14
11	14	15
12	15	16
13	16	17

p	a	b	c
5	6	8	12
7	11	12	17
10	12	15	19
11	16	17	22
14	17	19	23
15	19	22	26
16	21	22	27
18	22	23	28
20	23	26	30
21	26	27	32

p	a	b	c	d
12	13	18	28	39
17	25	28	39	51
24	28	35	45	55
27	37	39	51	63
34	39	45	55	66
36	45	52	62	72
38	49	52	63	75
44	51	55	66	77
48	55	62	72	82
50	61	63	75	87

FIG. 49 - Début des *LUT* de $d_{3,4}$, $d_{5,7,11}$ et $d_{12,17,27,38}$

5.5 Conditions de calculabilité

L'algorithme décrit au §5.4.3 donne les *LUT* exactes. On peut donc maintenant explorer le point suivant. Par l'emploi de la méthode des tables de correspondance, on a postulé que l'axe médian est détectable localement, c'est-à-dire par des tests dans le voisinage de pondération. C'est vrai pour les distances classiques d_4, d_8, $d_{3,4}$ et $d_{5,7,11}$. Mais est-ce bien le cas pour tous les masques de chanfrein, définis dans un voisinage plus grand?

Une condition nécessaire à la détectabilité locale est que le masque doit définir une norme, autrement dit que le masque doit être régulier (§3.5.1). En effet dans le cas contraire, il existe un cône irrégulier, dans lequel les déplacements élémentaires sont non constants, ce qui fausse complètement le test local.

Mais cette condition n'est pas suffisante : nous avons trouvé des contre-exemples (très rares) pour certains masques réguliers, où un disque n'était pas recouvert localement, mais par un grand disque situé beaucoup plus loin. Ces contre-exemples ont été trouvés par des test systématiques, et sont assez surprenants.

Prenons l'exemple de $d_{14,20,31,44}$. Dans l'image de distances calculée sur D^{-1}_{351}, le point (4,2) vaut 291, et se situe en dehors du masque de pondération. Ce point est détecté comme centre maximal, car aucune boule de son voisinage ne recouvre son disque D^{-1}_{291}, qui pourtant est recouvert par D^{-1}_{351}.

Ce problème est fondamental, mais la question reste ouverte.

5.6 Propriétés de l'axe médian

L'axe médian que l'on a caractérisé pour les distances de chanfrein possède un certain nombre de propriétés, classiques ou moins connues, que nous énumérons ici.

Réversibilité

L'axe médian correspond à un recouvrement de la forme par des boules de la métrique choisie. Il suffit donc de régénérer les boules, connaissant leur position et leur rayon, pour retrouver la forme initiale. Les algorithmes du §2.4.3 présentent l'intérêt majeur d'y parvenir globalement, en un nombre de passages fixe sur l'image : 2 pour les distances de chanfrein, et 3 pour la distance euclidienne. Le type de ces algorithmes est le même que ceux permettant le calcul d'une image de distance.

Codage

La simplicité du stockage et du décodage de l'axe médian en font une propriété appréciable dans le maniement des images binaires. L'utilisation de cette forme de codage est facilitée par l'indépendance de l'axe médian par rapport à la position de la forme dans l'image, contrairement au codage en quadtree par exemple [Cha91].

Non-minimalité

L'axe médian est composé des boules maximales, mais le principe de cette définition n'assure pas forcément que l'ensemble des points retenus pour assurer la réversibilité est minimal. En effet une boule maximale n'est incluse dans aucune autre, mais peut être incluse dans l'union de plusieurs autres, comme le montre la figure 50. La détection de ces points est abordée au §5.7.

1	1	1	1	1
1	2	*2*	2	1
1	1	1	1	1

	3			
3	4	3	3	3
	3	*4*	6	3
		3	3	3

FIG. 50 - Non-minimalité (italique) de AM (cerclé) pour d_8 et $d_{3,4}$

Stabilité

L'application de la transformation de distance inverse à un axe médian génère une forme, où les points sont étiquetés souvent différemment de l'image de distance de départ, car les labels proviennent des poids de AM, et non plus de points du fond. Les problèmes de caractérisation de l'axe médian en découlent. De toute image porteuse d'une information de distance on peut générer une forme ; mais l'axe médian de cette dernière sera sans doute différent de l'image de départ (figure 51).

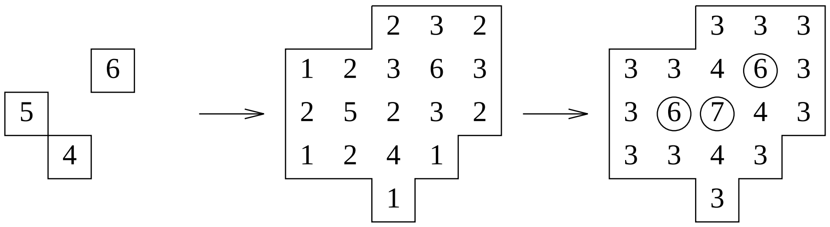


FIG. 51 - Image de poids, image reconstruite, DM et AM pour $d_{3,4}$

Déconnexion

Une propriété essentielle de l'axe médian est sa déconnexion. Elle distingue l'axe médian discret des squelettes continus définis par Blum [Blu67], qui sont simultanément connexes et réversibles. Cette déconnexion limite l'exploitation de l'axe médian, qui ne respecte pas l'homotopie de la forme. C'est pourquoi on s'emploie, au chapitre 6, à construire la ligne médiane qui est un sur-ensemble connecté de AM.

Epaisseur

On remarque également que l'axe médian peut comporter localement une certaine épaisseur, due par exemple à un diamètre pair de la forme. Voila une différence supplémentaire avec les squelettes, qui sont 'filiformes' dans l'espace continu, et 'd'épaisseur 1' dans l'espace discret (chapitre 6). Cette épaisseur est fort heureusement limitée, l'axe médian ne formant quasiment jamais de bloc 3×3.

Trous

La dernière propriété que l'on cite est la présence possible de trous dans l'axe médian. Cette éventualité est rare, mais il faut en tenir compte dans le calcul ultérieur de la ligne médiane. Dans l'exemple figure 52, la forme est obtenue par transformation inverse sur l'ensemble de gauche avec $d_{5,7,11}$. La partie droite montre une portion de l'image de distance dans laquelle l'axe médian en grisé forme un trou de deux pixels.

	51	51	
54	55	55	54
	60	60	

$\longrightarrow$

36	38	40	41	41	40	38	36
40	43	44	46	46	44	43	40
43	47	49	51	51	49	47	43
44	49	54	55	55	54	49	44
45	50	55	60	60	55	50	45
44	49	53	55	55	53	49	44
42	46	49	50	50	49	46	42

FIG. 52 - Trou dans l'axe médian avec $d_{5,7,11}$

5.7 Applications

La première application à laquelle on pense est la compression de données. Les performances peuvent encore être améliorées en cherchant à obtenir un sous-ensemble réversible minimal de l'axe médian. Les opérations pour y parvenir sont délicates et coûteuses. Davies propose dans [Dav80] un algorithme itératif pour la distance d_6 dans la maille hexagonale, mais qui n'assure ni l'unicité ni la minimalité.

Des opérations ensemblistes et géométriques sont possibles à partir de l'axe médian [Cha91], mais elles sont relativement limitées.

Une idée intéressante consiste à améliorer le rendu d'une rotation discrète sur un objet binaire. En effet les rotations dans l'espace discret [And92] ont la fâcheuse tendance de perturber le contour des objets. Le résultat est nettement meilleur si on extrait l'axe médian, que l'on rote, avant de régénérer la forme, de préférence avec une distance quasi-euclidienne, par exemple $d_{5,7,11}$.

L'exploitation de l'axe médian peut se révéler suffisante pour la granulométrie, ou le décomptage de cellules, si les objets en présence sont de taille régulières et assez ronds, et que l'on veut privilégier la vitesse sur la précision. Le procédé classique consiste à rechercher le plus grand centre maximal, à supprimer tous les autres centres situés dans son disque, puis à boucler avec un seuil d'épaisseur.

Les applications de calcul de périmètre et d'aire citées dans [Wu86, Cor89] utilisent un axe médian calculé avec d_8, qui est rendu homotope par le rajout de points, et affiné ; il s'agit en réalité d'un squelette.

De fait, on a presque toujours intérêt à travailler avec la forme reconnectée de l'axe médian, qui autorise des raisonnements globaux sur les objets, et rend les applications plus robustes. En ce sens, l'axe médian n'est qu'une étape du calcul du squelette pondéré, et non un but en soi. Mais cette étape est indispensable.

5.8 Conclusion

Diverses caractérisations de l'axe médian ont été proposées dans la littérature. Elles sont toutes destinées à une distance spécifique. La méthode la plus simple utilise des tables de correspondance ; c'est alors la table qui est dépendante de la distance, non plus la méthode.

Nous avons proposé un algorithme de calcul des tables de correspondance en un seul passage sur l'octant des distances. Il est très rapide et fonctionne pour tous les masques. Le passage de cet algorithme au 3D est de plus immédiat.

L'exploration des propriétés de masques plus grands que les habituels $d_{3,4}$ et $d_{5,7,11}$ a contredit pour certains masques un postulat communément admis, selon lequel un centre maximal peut être détecté dans le voisinage de pondération. La nécessité d'avoir une norme, c'est-à-dire un masque régulier, est acquise ; la détermination d'une loi précise, qui assure la détectabilité locale, est le prochain défi à relever.

Chapitre 6

Squelette pondéré

Nous proposons une méthode pour extraire le squelette pondéré d'une image de distance, calculée avec les distances discrètes les plus courantes.

Le squelette obtenu est quasi réversible car il inclut presque tous les points de l'axe médian. Il est robuste en rotation si l'on adopte une distance qui approxime correctement la distance euclidienne. L'algorithme comporte une étape d'élagage et d'embellissement, qui permet de simplifier la structure du squelette selon les besoins et d'en améliorer l'esthétique. Le temps de calcul est très limité et est indépendant de l'épaisseur des objets à squelettiser.

6.1 Introduction

La description de formes est une étape importante en analyse d'images. Si l'objet a une forme simple, i.e une silhouette compacte ou une frontière convexe, sa description peut être donnée en termes géométriques comme l'aire, le périmètre, le facteur de forme, les moments, etc. Ces traits sont parfois suffisants pour *classifier*, mais non pour *décrire* de façon appropriée une forme complexe, de frontière non convexe et pouvant être perçue comme l'union de régions simples. Dans ce cas, il vaut mieux suivre une approche structurelle.

La notion de *squelette* est apparue dans [Blu67]. La définition donnée par Blum est intuitive ; il s'agit de modéliser, dans l'espace *continu*, l'intersection d'un front d'onde partant du bord d'un objet, de façon à obtenir une représentation filiforme de l'objet.

Les propriétés majeures que l'on attend du squelette sont d'être *centré d'épaisseur 1*, *homotope* et *réversible*. L'homotopie signifie que l'image et son squelette ont le même nombre de composantes connexes, et pour chacune d'entre elles, le même nombre de trous. Centré dans la forme, le squelette fournit une représentation équivalente à l'objet et unidimensionnelle. La description de l'objet peut se faire entièrement à travers celle du squelette, qui possède toute l'information synthétisée sous forme de valeurs de distances.

Le calcul du squelette est une opération difficile. Quasiment impossible dans l'espace continu, elle a divergé en trois familles de méthodes.

- A partir d'une polygonalisation du contour, une formulation analytique en segments de droites et en arcs de paraboles en produite dans [Mar87].

- Une autre méthode semi-continue prend en entrée un échantillonnage du contour. On calcule le diagramme de Voronoï [Ber94], duquel on extrait un squelette sous forme de graphe [Att93, Att94b, Att94a]. La comparaison de cette approche avec les squelettes pondérés discrets est effectuée dans [Thi93a].

- Quant à la famille de méthodes dédiées à l'espace discret, on distingue les squelettes *binaires*, résultats d'un calcul itératif, des squelettes *pondérés*, construits sur une image de distance.

Les méthodes de calcul de squelettes binaires sont fondées sur la suppression itérative de points du contour. Les points à supprimer sont détectés par analyse de leur voisinage, en faisant intervenir la plupart du temps des masques de détection [Cha91]. Une très abondante littérature traite de ce sujet ; on peut entre autres trouver 140 références dans [Lam92].

Le squelette pondéré est généralement considéré comme une bonne représentation de formes binaires lorsque leur épaisseur varie ou est non négligeable, et est utilisé comme un puissant outil pour la décomposition et la description de formes [Cor84, Mon86, Thi92e]. En effet, les poids des points du squelette nous informent sur l'épaisseur locale, et ainsi élargissent le nombre de classes de figures analysables par leur squelette.

Les algorithmes de squelettisation menée en distance produisent justement des squelettes dont les points sont correctement étiquetés à leur distance au fond [Cha91]. Différentes cartes de distance peuvent guider la squelettisation, avec les distances d_4 [Arc89], d_8 [Mon87a], $d_{3,4}$ [San94], $d_{5,7,11}$ [Arc92] et aussi d_E [Arc93], avec des algorithmes de différents types.

L'idée de trouver une méthode multi-distances semble être naturelle, mais les algorithmes existants sont généralement pensés pour un cas spécifique. Une exception peut être trouvée dans [Dor86a], qui est la première approche multi-distances (pour d_4, d_8, $d_{2,3}$ et $d_{5,7,11}$). On y retrouve les grandes étapes que nous décrirons dans la suite, mais avec des définitions *ad hoc* de l'axe médian et des points selles. Les résultats obtenus ne sont pas à proprement parler des squelettes, car ils ne sont pas d'épaisseur 1 et ne passent pas forcément par le vrai axe médian.

Dans ce chapitre, nous proposons un algorithme de squelettisation, valable pour toute distance de chanfrein à 2 ou 3 poids dans un voisinage 5×5 (en particulier d_4, d_8, $d_{3,4}$ et $d_{5,7,11}$) [Thi94d, Thi94b].

6.2 Trame de la squelettisation

Une image binarisée sur le maillage carré contient des *objets* (ensembles de points connexes non nuls) et le *fond* à 0. Pour éviter un paradoxe topologique, nous considérons la 8-connexité pour les objets et la 4-connexité pour le fond (théorème de Jordan §2.2). Aucune limitation n'est demandée sur le nombre de trous du fond, mais nous supposons qu'une phase préliminaire de nettoyage nous débarrasse des trous nuisibles (par exemple, un filtrage par aire, qui respecte bien les détails, ou bien une ouverture / fermeture, plus brutale). En effet, les boucles en correspondance avec de tels trous pourraient irréparablement affecter la structure du squelette. A l'inverse, les branches du squelette provenant de renflements non significatifs seront supprimées pendant l'élagage du squelette.

La carte de distance notée DM est une copie de l'image où chaque point est étiqueté à la valeur de sa distance au fond (§2.4). L'algorithme que nous proposons extrait le squelette de DM, calculé selon l'une des distances d_4, d_8, $d_{3,4}$ et $d_{5,7,11}$.

Le squelette S d'un objet en est un sous-ensemble, donc S est 8-connexe (et son complémentaire est 4-connexe) quelle que soit la distance pour DM.

Définition 6.1 (Squelette pondéré) *S est caractérisé par les propriétés :*

1. *S a le même nombre de 8-composantes que l'objet, et chaque composante de S a le même nombre de 4-trous que sa partie correspondante dans le fond (homotopie).*
2. *S est centré dans l'objet.*
3. *S est l'union d'épaisseur 1, de 8-arcs et de 8-courbes [Ros73].*
4. *Les points de S sont étiquetés à leur distance au fond.*
5. *S inclut presque tous les points de l'axe médian (l'inclusion complète est incompatible avec le $3^{ème}$ point).*

En général, le voisinage $N(p)$ d'un point p inclut les voisins $n_i(p)$ qui peuvent être atteints par un déplacement simple depuis p. Ainsi $N(p)$ devrait inclure de 4 points pour d_4 à 16 points pour $d_{5,7,11}$. Dans ce chapitre, $N(p)$ inclut toujours les 16 voisins de p (figure 53). Dans la suite, p indique indifféremment le point ou son poids.

L'algorithme que nous proposons comprend trois grandes étapes :

1. Calcul de la carte de distance et extraction de l'axe médian ;
2. Propagation de la ligne médiane ;
3. Extraction du squelette.

	9		10	
8	1	2	3	11
	0	p	4	
15	7	6	5	12
	14		13	

(a)

	c		c	
c	b	a	b	c
	a	p	a	
c	b	a	b	c
	c		c	

(b)

FIG. 53 - Notations utilisées dans tout le chapitre pour les 16 points (figure a) du voisinage $N(p)$ de p. Nous attribuons les valeurs adéquates aux poids a, b, c (aussi notés w_i) selon la distance employée. Les voisins $n_i(p)$ sont encore appelés a-voisins $(i = 0, 2, 4, 6)$, b-voisins $(i = 1, 3, 5, 7)$ ou c-voisins $(i = 8..15)$ (figure b).

La construction de la ligne médiane nécessite le calcul des points selle, une phase de propagation et le remplissage de faux trous. L'extraction du squelette inclut la réduction de la ligne médiane à l'épaisseur unité, un élagage et un embellissement.

L'algorithme est implémenté sur machine séquentielle. Pour des raisons de clarté, nous préférons décrire les étapes de la squelettisation comme si elles étaient accomplies individuellement. Le nombre d'inspections est en réalité faible, et indépendant à la fois de la taille des objets dans l'image et de la distance choisie.

6.3 Première étape : DM et AM

La première étape de l'algorithme est le calcul de la carte de distance et la détection de l'axe médian.

Comme l'information de distance peut être propagée d'un point à tous ses voisins, des opérations locales sont suffisantes pour construire la carte de distance DM pour $d_{a,b,c}$. Au §2.4 on donne l'algorithme de Rosenfeld qui, en deux passages séquentiels sur l'image, $f_1(p)$ pour le passage avant, et $f_2(p)$ pour le passage arrière, étiquette chaque point p des objets à sa distance au complémentaire.

$$
\begin{aligned}
f_1(p) &= \min\{\, n_0(p) + a,\ n_1(p) + b,\ n_2(p) + a,\ n_3(p) + b,\ n_{8..11}(p) + c \,\} \\
f_2(p) &= \min\{\, p,\ n_4(p) + a,\ n_5(p) + b,\ n_6(p) + a,\ n_7(p) + b,\ n_{12..15}(p) + c \,\}
\end{aligned}
$$

En donnant la valeur ∞ aux poids correspondant à des déplacements interdits pour une distance $d_{a,b,c}$ donnée, il est possible de traiter de façon homogène cette transformation de distance. On fixe ainsi $a = 1$, $b = c = \infty$ pour d_4 ; $a = b = 1$, $c = \infty$ pour d_8 ; $a = 3$, $b = 4$, $c = \infty$ pour $d_{3,4}$; $a = 5$, $b = 7$, $c = 11$ pour $d_{5,7,11}$.

L'axe médian AM est le lieu des centres des boules maximales dans la forme, i.e qui ne sont incluses dans aucune autre. Il correspond à un recouvrement de la forme, et est donc réversible : il suffit d'appliquer sur AM la transformation de distance inverse du §2.4.3.

L'inclusion des points de AM dans la ligne médiane LM garantit une localisation correcte par ses extrémités des renflements de la forme. La boule maximale située sur une telle extrémité est le cercle osculateur du renflement correspondant ; lorsqu'on déplace cette boule maximale sur sa branche, elle touche le bord de l'objet en de multiples endroits, centrant au mieux LM dans sa forme.

Il n'est pas suffisant d'inclure AM pour garantir la stabilité du squelette en rotation. En effet, les boules des métriques discrètes sont des polygones dont les côtés ont des orientations fixes (§3.3). Ainsi le nombre et la localisation des disques nécessaires pour recouvrir une figure dépendent de sa géométrie et de l'orientation de ses contours. La stabilité en rotation croît avec le nombre de côtés caractérisant le disque. De ce fait les disques construits avec $d_{5,7,11}$ et $d_{3,4}$ doivent être préférés aux disques construits avec d_4 ou d_8.

Le chapitre 5 est consacré au calcul de l'axe médian. On extrait l'axe médian en un seul passage sur l'image de distance avec des opérations locales, en employant de petites tables de correspondance précalculées.

Nous illustrons notre algorithme de squelettisation en montrant à chaque étape les résultats pour les 4 distances sur le même exemple synthétique. La figure 54 montre l'image originale et l'axe médian.

6.4 Propagation de la ligne médiane

La deuxième étape de l'algorithme consiste à calculer la ligne médiane. Elle nécessite la détection de points selle, une propagation avec un suivi de gradients, et une détection de faux trous.

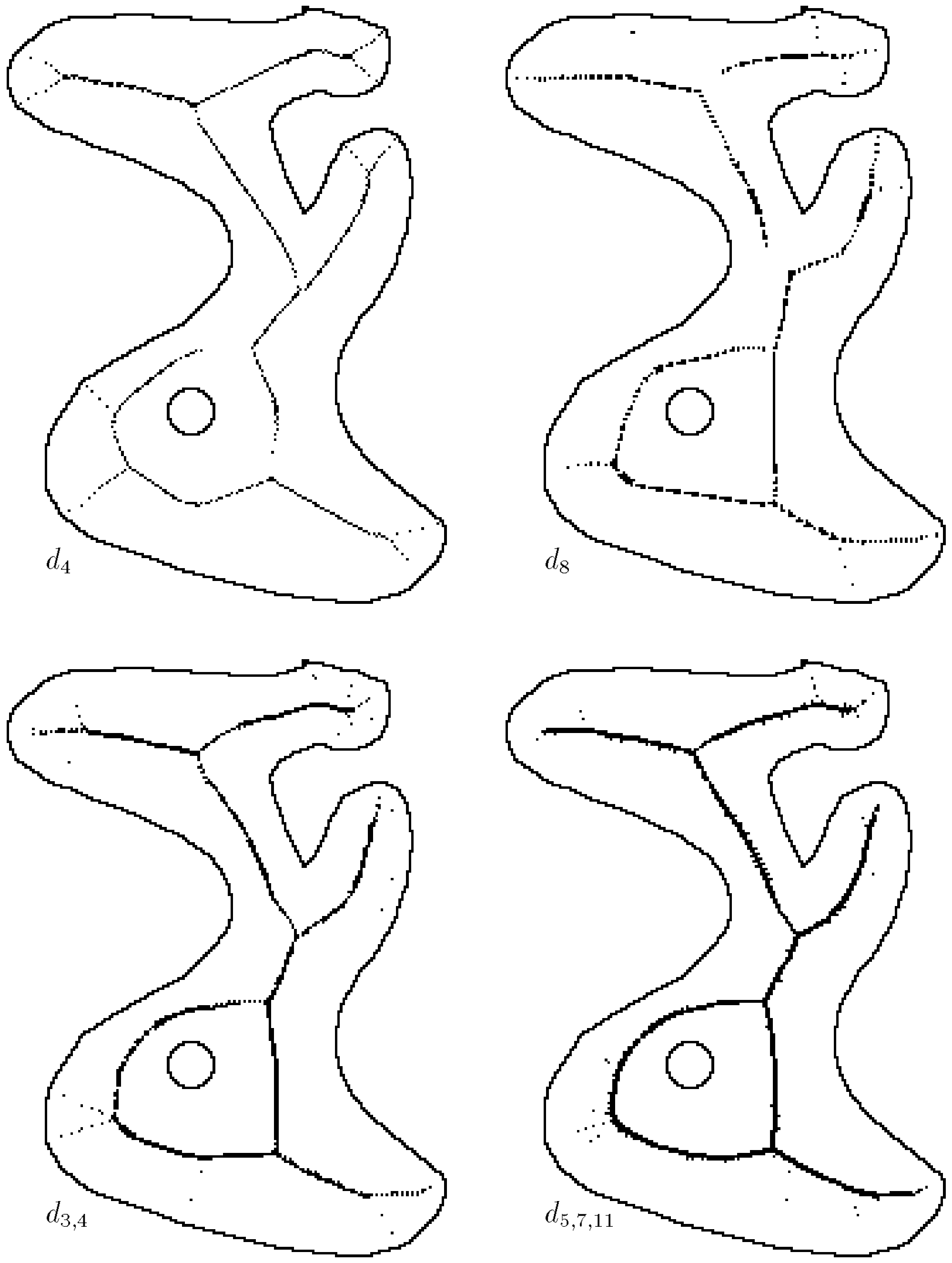


FIG. 54 - Axe médian et bord externe de l'objet initial

6.4.1 Détection des points selle

Les points selle constituent une crête connectant deux parties de la carte de distance comportant des poids plus grands. La plupart des points selle sont des AM, et leur identification est donc déjà assurée. Cependant, les extrémités de ces crêtes (voire la crête entière, lorsque sa taille est d'au plus 2 points), sont susceptibles de ne pas être des AM [Arc93].

Un p non AM est un point selle si il remplit l'une au moins des Conditions de Point Selle notée CPS (cf figure 53 pour la notation des voisinages, et figure 55 en exemple) :

1. Le 8-voisinage de p comprend plus d'une composante 4-connexe de points plus petits que p.

2. Le 8-voisinage de p comprend plus d'une composante 8-connexe de points plus grands que p.

3. Les poids de l'un des triplets suivants sont égaux à p : $(n_0(p), n_1(p), n_2(p))$, $(n_2(p), n_3(p), n_4(p))$, $(n_4(p), n_5(p), n_6(p))$, $(n_6(p), n_7(p), n_0(p))$.

1	1	1	1	1	1							1
1	2	2	2	2	1					1	1	1
1	2	3	3	2	1	1	1	1	1	1	2	1
1	2	2	3	2	2	2	2	2	2	2	2	1
1	1	2	2	2	3	3	3	3	2	2	1	1
	1	1	1	2	3	3	2	2	2	1	1	
			1	2	2	2	2	1	1	1		
			1	1	1	1	1	1				

FIG. 55 - Carte de distance d_8, axe médian (cercles) et points selle (losanges) de types 2, 3, 1 de gauche à droite

Les nombres de composantes 4 ou 8-connexes dans les conditions 1. et 2. peuvent être facilement calculés avec le 'crossing number' X_4 et le 'connectivity number' C_8 (cf §2.2) respectivement : on doit avoir $X_4(n_i < p) > 1$ ou $C_8(n_i > p) > 1$. La condition 3. est testée pour tous les points si $a = 1$, seulement pour les points à a sinon.

6.4.2 Propagation des chemins

En général, l'ensemble des points AM et des points selle n'est pas connecté. Aussi des points supplémentaires sont nécessaires. Dans [Arc85], les points de reconnection sont détectés en deux passages sur l'image pour d_8, en utilisant des propriétés spécifiques à cette distance. La méthode habituellement retenue pour les distances de chanfrein, consiste à propager des chemins suivant les gradients *positifs* ; la structure de la carte de distance garantit la bonne reconnection du résultat.

La propagation ne doit en principe démarrer que dans les configurations selle au niveau des extrémités des crêtes. En effet seuls ces points dans la crête ont des voisins de poids supérieur. Un seul chemin est tracé pour chacun d'entre eux, excepté lorsque la longueur de cette crête est de 1 (ses extrémités sont confondues), auquel cas 2 chemins proviendront du même point selle.

Nous n'opérons aucun test pour distinguer les extrémités d'une crête des autres points selle, ou pour mesurer la taille des crêtes. Les points selle pouvant faire partie de AM, le suivi des chemins doit être tenté pour tout point AM ou selle. Soit p un tel point, $N(p)$ contient au plus 2 composantes 8-connexes de points plus grands que p, desquels un chemin ascendant peut être commencé. Pour chaque point de la composante courante, le gradient est calculé. Le voisin $n_i(p)$ qui maximise le gradient dans sa composante est marqué comme premier point du chemin. De là, les voisins $n_k(n_i(p))$ sont inspectés pour trouver le prochain point (plus grand que $n_i(p)$), nécessairement unique, qui continuera le chemin. Le tracé du chemin se poursuit sur le plus grand gradient, tant que des points possédant un gradient positif sont trouvés.

Certains problèmes doivent être traités, qui découlent de ce que les chemins 8-connexes sont tracés, alors que la carte DM où le gradient est calculé n'a pas été nécessairement obtenue avec la distance d_8. De plus, sélectionner indifféremment tous les voisins maximisant le gradient peut épaissir excessivement les chemins, voire former un delta. Pour éviter cela et pour obtenir des chemins 8-connexes, nous assignons des valeurs spéciales aux poids $v_i = a'$, b', c' utilisés pour calculer le gradient. (voir figure 53 pour la notation des voisinages).

Pour chaque voisin $n_i(p) > p$, le gradient est $grad_i = [n_i(p) - p]/v_i$ avec : $a' = 2$, $b' = 3$, $c' = -1$ pour d_4 et d_8 ; $a' = 3$, $b' = 4$, $c' = -1$ pour $d_{3,4}$; $a' = 5$, $b' = 7$, $c' = 11$ pour $d_{5,7,11}$.

Dans le cas de $d_{3,4}$, au plus 2 points adjacents $n_i(p)$ ($i = 0..7$) maximisent le gradient dans la même composante. Lorsque c'est le cas, seul le voisin direct (i pair) est propagé pour éviter l'épaississement du chemin.

Dans le cas de $d_{5,7,11}$, 3 points peuvent maximiser simultanément le gradient (un a-, un b-, et un c-voisin). Pour éviter l'épaississement, et suivre le gradient correct (le maximisant du a-voisin de p n'est pas toujours le c-voisin de p), on ne doit accepter que le c-voisin pour la propagation. A partir du moment où un c-voisin de p est choisi, on rajoute simplement le a- ou le b-voisin intermédiaire pour respecter la 8-connexité du chemin (de préférence celui des deux qui a le plus fort gradient).

6.4.3 Remplissage des faux trous

Les fausses boucles dans la ligne médiane sont des sous-ensembles de points multi-connectés de LM, qui n'entourent aucun trou de l'objet. Ces faux trous doivent être remplis (en les marquant comme points de LM) pour obtenir une LM topologiquement correcte.

Les faux trous sont provoqués par l'axe médian (figure 52), ou sont obtenus lorsque des chemins propagés (presque) parallèles se rencontrent (phénomène typiquement discret). Dans les cas de d_4, d_8 et $d_{3,4}$, la seule possibilité est la rencontre de deux chemins orientés diagonalement. Ces faux trous ont une taille de 1 pixel. Pour $d_{5,7,11}$, puisque les chemins peuvent suivre une direction du cavalier, il est de plus possible de voir des faux trous de 2 pixels.

Un point p appartient à un faux trou si l'une des conditions suivantes est remplie :

1. $p \notin LM$, $\{n_0(p), n_2(p), n_4(p), n_6(p)\} \subset LM$;
2. $p \notin LM$, $\{n_0(p), n_2(p), n_3(p), n_5(p), n_6(p), n_4(n_4(p))\} \subset LM$;
3. $p \notin LM$, $\{n_0(p), n_2(p), n_4(p), n_5(p), n_7(p), n_6(n_6(p))\} \subset LM$.

Ce test doit être effectué à la fin de la propagation de la ligne médiane, et peut se simplifier (on teste d'abord p, $n_0(p)$, $n_2(p)$, etc). Le remplissage ne provoque pas d'épaississement excessif de LM, car les trous sont en général extrêmement clairsemés dans l'image. A cet endroit du procédé, la ligne médiane obtenue est 8-connexe, homotope, comprend tous les points de l'axe médian et est donc réversible, mais n'est pas d'épaisseur 1 (figure 56).

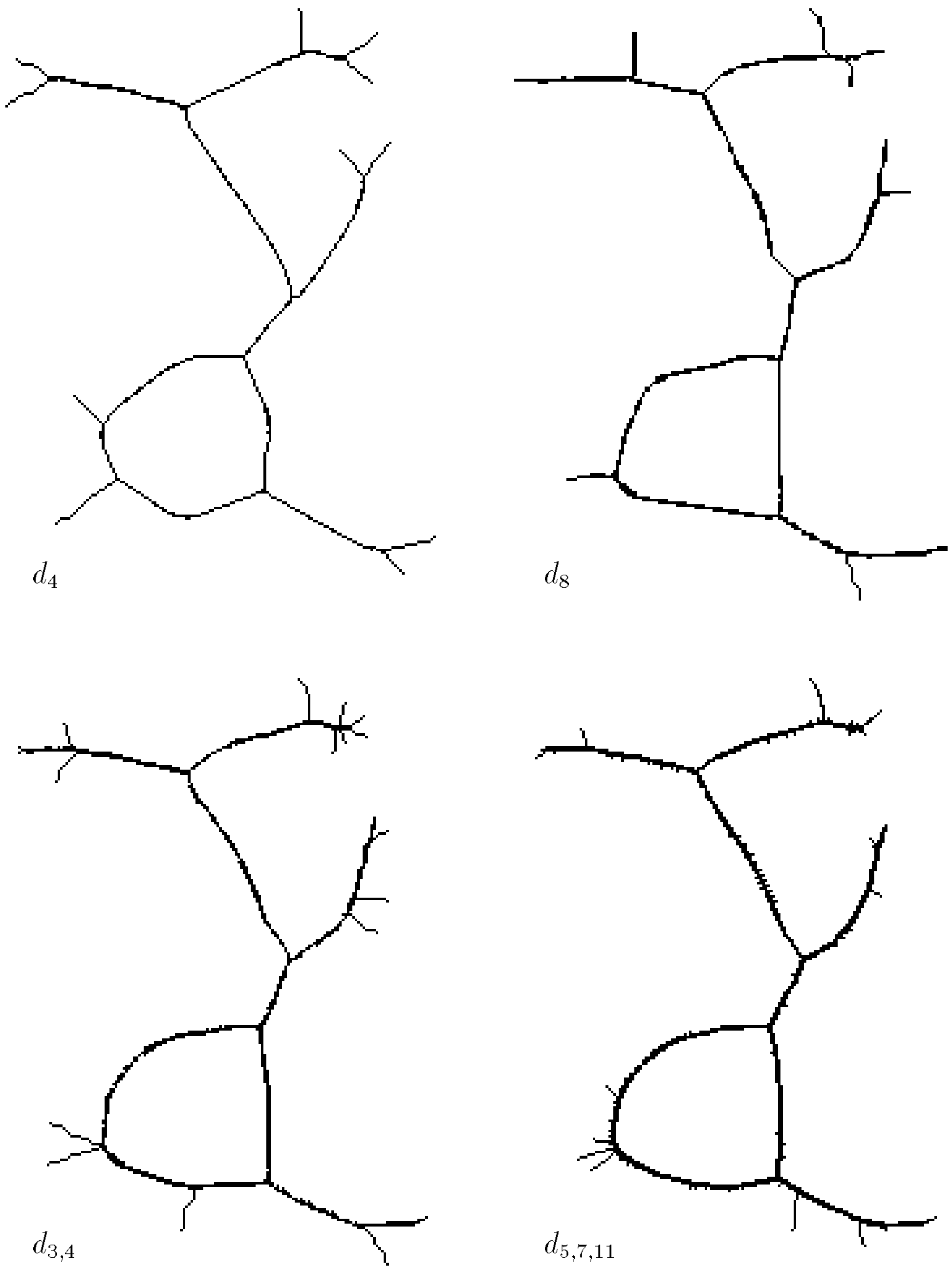


FIG. 56 - Ligne médiane

6.5 Extraction du squelette

6.5.1 Réduction à l'épaisseur 1

La réduction de la ligne médiane à l'épaisseur 1 peut être obtenue en appliquant des opérations séquentielles de suppression qui préservent la topologie, et qui évitent le raccourcissement des branches du squelette. La réduction à l'épaisseur 1 est ici un processus de re-marquage plutôt que de suppression réelle, l'information 'ancien LM' pouvant servir par la suite. On désigne par S l'ensemble des points restant à un moment donné.

Il arrive que la ligne médiane présente une forme irréductible [Eck88], qui donc ne peut être amincie. Dans le cas général, la ligne médiane peut contenir des points 4-internes, voire 8-internes (bien plus rares). Pour favoriser le centrage du squelette dans la ligne médiane, cette étape est réalisée en 2 passes. Au 1^{er} passage, les points 4-internes dans LM ne sont pas supprimables, puis au 2^{nd} passage, tous les points restants sont candidats à la suppression (figure 57).

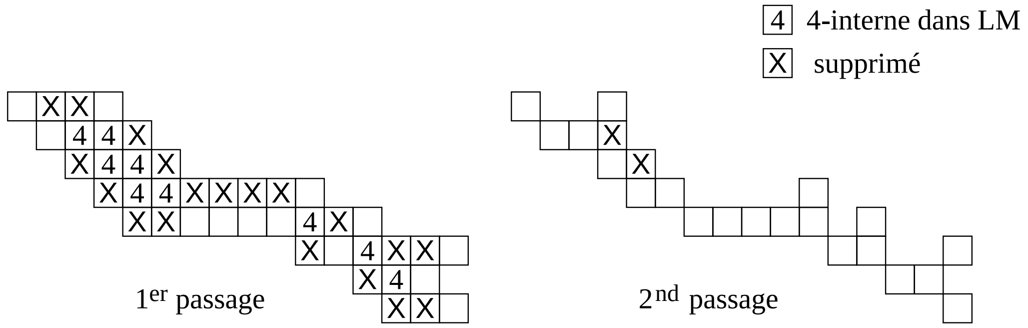


FIG. 57 - Réduction de la ligne médiane à l'épaisseur 1

Un point p est supprimé dans la $1^{ère}$ inspection de S (la 2^{nde}) si il satisfait les conditions C0+C1 (conditions C0+C2). C0 préserve la topologie et évite en même temps tout raccourcissement des branches de S. C1 centre S dans LM, et C2 empêche la création de trous dans S.

C0 Au moins un triplet de voisins $n_i(p)$, $n_{i+2}(p)$, $n_{i+5}(p)$ existe ($i = 0, 2, 4, 6$, addition modulo 8), tel que $n_i(p)$ et $n_{i+2}(p) \in S$ alors que $n_{i+5}(p) \notin S$.

C1 Au moins un 4-voisin de p n'appartient pas à LM (originale).

C2 Au moins un 4-voisin de p n'appartient pas à S (points restants à cet instant).

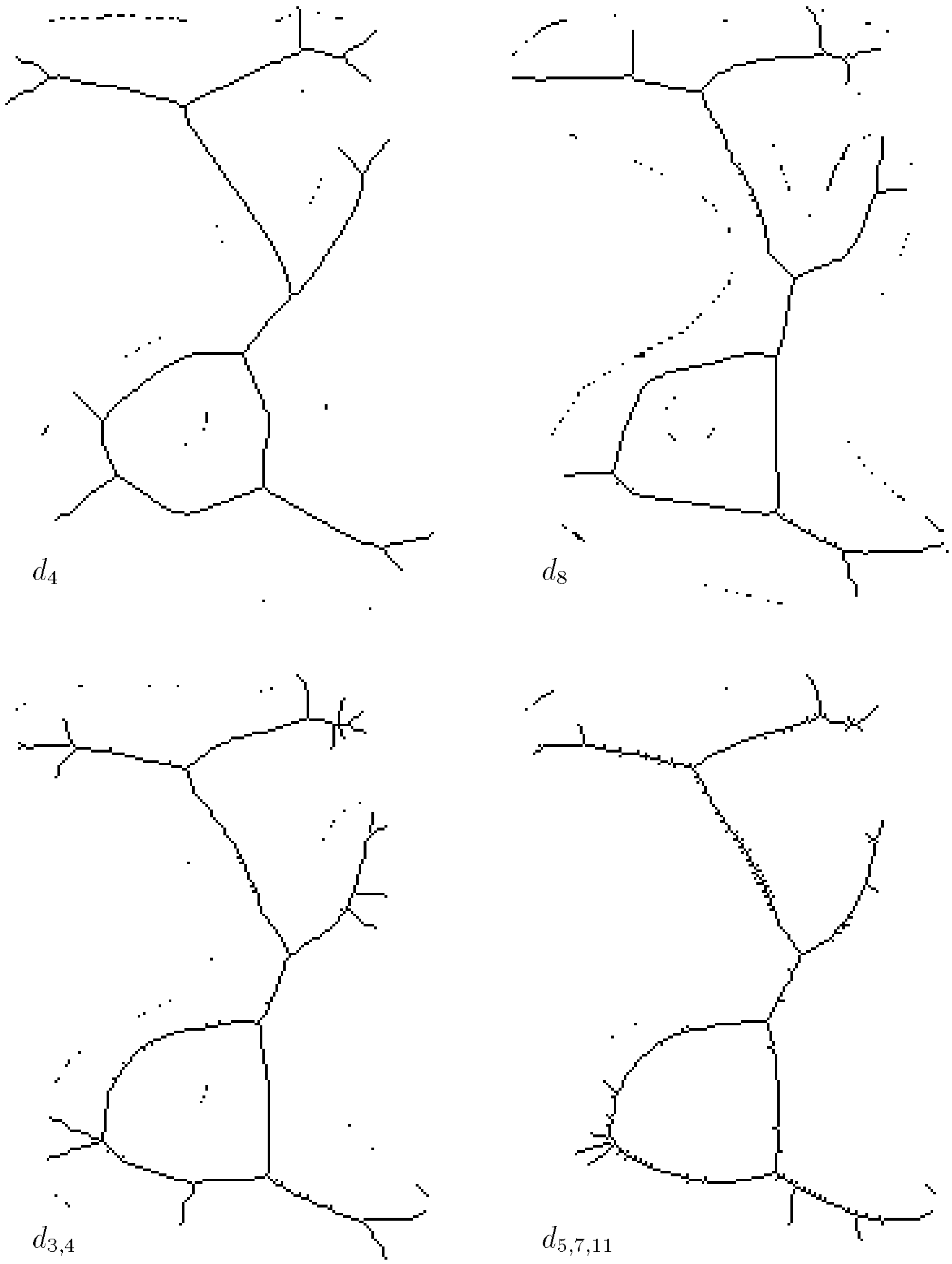


FIG. 58 - Squelette pondéré non élagué et points perdus (sur les bords)

Cette méthode très simple réduit considérablement le cardinal des points du squelette. Comme certains points de l'axe médian peuvent être supprimés, la transformation n'est plus totalement réversible, mais les points perdus (au niveau du bord) sont si épars et insignifiants (leur existence est fortement liée à la digitalisation de départ de l'objet) que le squelette peut nonobstant être considéré comme une représentation fidèle de l'objet (figure 58).

6.5.2 Elagage

L'élagage est effectué pour simplifier la structure du squelette, en supprimant des branches périphériques (i.e branches délimitées par un point terminal) qui ne correspondent pas à des renflements significatifs pour une application donnée. Il s'agit entre autres de branches créées par des bruits du contour, ou artefacts de la réduction de la ligne médiane à l'épaisseur unité. L'élagage est de plus important pour réduire la sensibilité du squelette à la rotation et au changement d'échelle.

Elaguer une branche implique cependant la suppression des points de l'axe médian qu'elle contient. Pour tenir compte de la perte d'information causée par la coupe de la branche, il est souhaitable de se baser sur un critère convenable d'importance relative de la branche. En général, une branche peut être coupée 'en sécurité' si la différence recouvrée avant et après la coupure est négligeable. Cette différence peut être évaluée en terme de *degré de recouvrement* entre les disques centrés sur les deux extrémités de la branche à couper [San94].

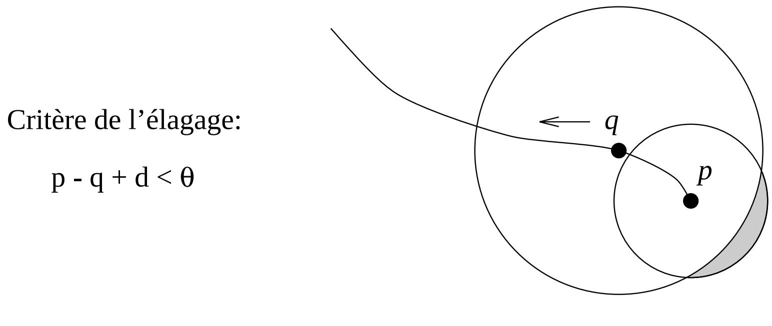


FIG. 59 - Degré de recouvrement

Soit p le point terminal de la branche et $q > p$ un point s'éloignant de p sur la branche. Soit P et Q leurs disques associés. La branche peut être élaguée de p à q

lorsque Q recouvre largement P. Le degré de recouvrement entre P et Q peut se traduire par $(p - q + d(p, q))$, où d est la distance discrète utilisée pour construire la carte de distance. On utilise un seuil de tolérance θ qui dépend de la qualité de réversibilité attendue (figure 59). Le choix du seuil introduit une description hiérarchique des régions, par analogie avec les squelettes semi-continus [Att93].

Nous généralisons l'expression définie en [San94] de telle sorte qu'elle puisse être calculée quelle que soit la distance utilisée. Pour toute distance discrète, la distance entre deux points p et q peut être exprimée en fonction du nombre de déplacements unitaires horizontaux H et diagonaux D, le long d'un chemin 8-connexe de longueur minimale les rejoignant (voir section 3.5.2).

Pour $d_{a,b,c}$, le degré de recouvrement entre les disques P et Q est donc :

$$p - q + a\,H + b\,D + \delta \min(H, D)$$

où $\delta = 0$ si $c = \infty$ (cas de d_4, d_8 et $d_{3,4}$), sinon $\delta = c - b - a$ ($\delta = -1$ pour $d_{5,7,11}$).

Le seuil θ peut être fixé à a fois le nombre maximum de lignes ou colonnes périphériques dont on accepte la perte. On utilise couramment $\theta = 2 \times a$.

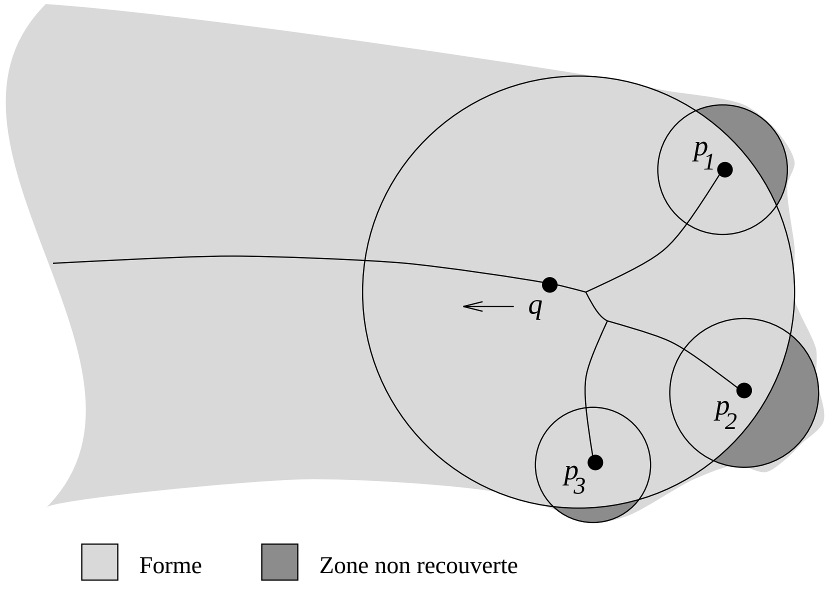


FIG. 60 - Elagage contrôlé

Dans [San94], l'élagage est aussi interrompu lorsque $q < p$, ou lorsqu'un nœud est atteint. En effet un nœud intermédiaire pourrait se transformer durant l'élagage en point terminal et provoquer une sommation des pertes. Mais ici nous ne limitons pas l'élagage aux branches externes du squelette. Nous propageons une étiquette unique pour chaque point terminal sur sa branche, et chaque fois que des branches se rencontrent en nœud, nous ajoutons leurs étiquettes dans des listes circulaires. De là, chaque point q connaît tous ses ancêtres $p_1, p_2..p_n$, et le critère de recouvrement est testé pour chaque couple (q, p_i) ; dès que l'un d'eux dépasse le seuil on stoppe (figure 60).

6.5.3 Embellissement et simplification

Au niveau macroscopique, le squelette comporte des branches périphériques de moindre importance, qui sont élaguées au §6.5.2. Au niveau des pixels, on peut distinguer 3 catégories de points dans le squelette :

- le point *extrémité*, qui a une unique composante 4-connexe de voisins non dans le squelette ;
- le point *normal*, qui a exactement 2 8-voisins ;
- le point de *branchement*, non-extrémité, qui a au moins 3 8-voisins.

On fait apparaître ces catégories de points sur un fragment de squelette figure 61.

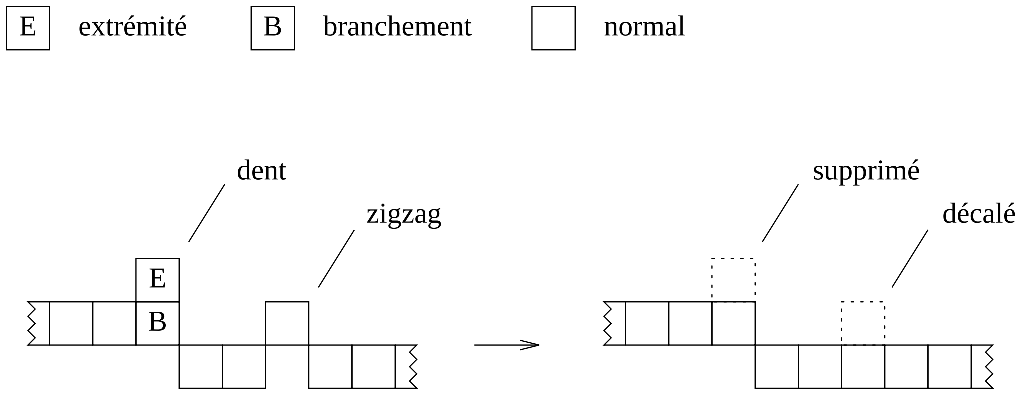


FIG. 61 - Fragment de squelette

L'algorithme d'amincissement du §6.5.1 privilégie la rapidité, mais provoque de nombreux 'zigzags' et de 'dents' (point extrémité voisin d'un point de branchement). Tout l'intérêt consiste à produire le squelette le plus simple structurellement, pour faciliter le parcours ultérieur des arcs du squelette, et le squelette le plus 'lissé', pour la robustesse de la polygonalisation de ces arcs, effectuée au chapitre 7. Pour obtenir de beaux squelettes, on doit donc corriger les zigzags et supprimer les dents.

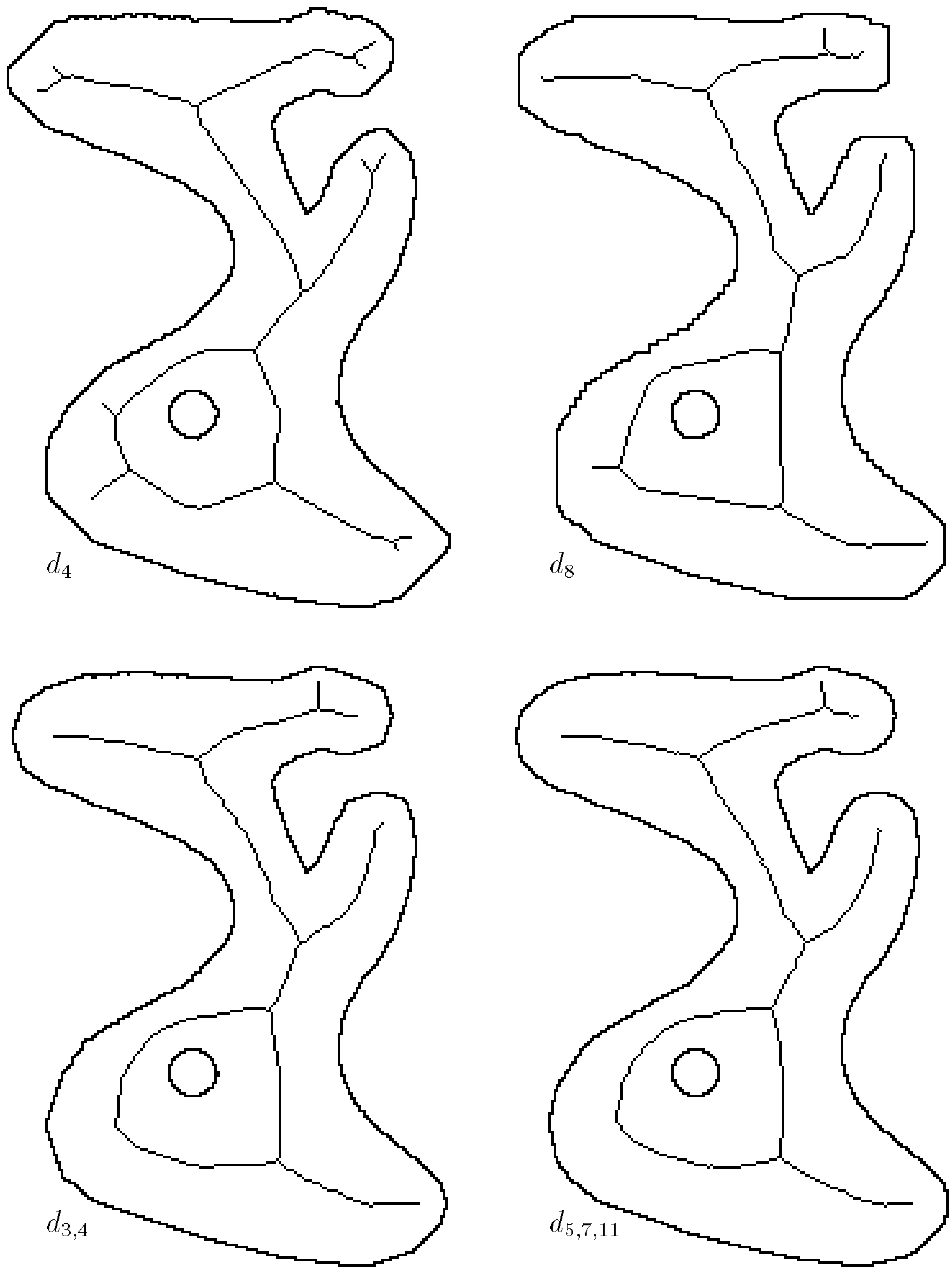


FIG. 62 - Squelette définitif (après élagage $\theta = 2 \times a$ et embellissement) et bord externe de la région codée

La correction des zigzags s'opère en déplaçant le marqueur de certains points vers un voisin. Un point p n'ayant dans S que 2 voisins $n_i(p)$ et $n_{i+2}(p)$ ($i = 1, 3, 5, 7$, addition modulo 8) est supprimé de S, au profit de son voisin $n_{i+1}(p)$, qui est marqué à la place de p dans le même temps. On peut décider de n'accepter $n_{i+1}(p)$ que s'il faisait partie de LM (originale) avant sa réduction à l'épaisseur unité, auquel cas il est un point de squelette de plein droit.

La détection des dents est très simple, vu leur définition. La suppression du point extrémité peut modifier (c'est le but) la catégorie du point de branchement voisin (figure 61).

On doit tenir compte du fait qu'un déplacement ou une suppression peuvent causer un épaississement de S dans le voisinage, qui fera supprimer d'autres points, créant des dents, zigzags, etc.

Nous avons implémenté la phase complète d'élagage / embellissement / correction selon un schéma parallèle. On mémorise au départ tous les points terminaux, puis chaque sous-phase est accomplie et donne lieu éventuellement à un voisinage à traiter. On ne balaie plus l'image, d'où la très grande rapidité du processus. La figure 62 montre le résultat final sur notre exemple.

6.6 Discussion

Dans la méthode de squelettisation que nous avons présentée dans ce chapitre, on peut regrouper certaines phases, et nous ramener à l'algorithme suivant :

1. calculer l'image de distances DM (2 parcours) ;
2. marquer l'axe médian, les points selles et les chemins de reconnection (1 parcours alterné avec une propagation) ;
3. remplir les faux trous et mémoriser les points du squelette (1 parcours) ;
4. réduire à l'épaisseur 1 (les points sont directement adressés) ;
5. élaguer et simplifier (le squelette est parcouru à partir des points extrémité).

On ne peut pas parler ici de complexité de l'algorithme, mais de nombre de passages sur l'image. Le coût de cet algorithme est modeste en temps de calcul car seulement 5 parcours de l'image sont nécessaires, quelle que soit l'épaisseur de la forme à squelettiser.

Cependant le temps de calcul dépend du ratio entre les points objet et fond de l'image, ainsi que de la distance employée. L'algorithme a été testé sur un ensemble d'images 512×512 dans [Thi94b]. La comparaison des résultats montre que le temps de calcul est légèrement plus grand pour $d_{5,7,11}$, puisqu'il nécessite des opérations sur un plus grand nombre d'arguments. La distance d_8 est à peine plus coûteuse que d_4 et $d_{3,4}$, la ligne médiane étant plus épaisse pour d_8.

Le critère d'arrêt pour l'élagage est donné au §6.5.2 en terme de degré de recouvrement, et se traduit par une plus ou moins grande perte en nombre de pixels selon la distance employée. Cela est dû à la géométrie des boules. Le ratio entre les points objet et les points perdus donne une mesure de la réversibilité de la méthode.

Un compromis raisonnable entre le temps de calcul et la réversibilité est $d_{3,4}$. Le squelette produit par cette distance présente aussi une bonne stabilité en rotation. C'est pourquoi $d_{3,4}$ est généralement employée dans les applications.

La distance $d_{5,7,11}$ doit être préférée lorsque une plus grande fidélité est requise, car elle est très proche de la distance euclidienne. Le temps de calcul légèrement supérieur est compensé par la plus grande stabilité en rotation, et la précision utile par exemple en morphométrie.

Le critère d'élagage peut être amélioré. Une estimation de l'aire non recouvrée affinerait la réversibilité, mais peut-être aussi au détriment de la simplicité du squelette.

6.7 Conclusion

Dans ce chapitre, nous avons introduit une méthode unifiée pour extraire le squelette pondéré d'une image de distance, construite avec les distances discrètes les plus courantes. Nous avons utilisé des distances à un poids (d_4), deux poids (d_8 et $d_{3,4}$) et trois poids ($d_{5,7,11}$). L'algorithme peut être facilement étendu à toute distance $d_{a,b}$ ou $d_{a,b,c}$. Pour ce faire, il suffit de recalculer les tables de correspondance pour l'extraction de l'axe médian. L'extension aux distances de plus de trois pondérations est possible, avec quelques modifications supplémentaires : au niveau de la ligne médiane, il faut adapter le suivi de gradient et la recherche des faux trous. Si la démarche est voisine pour la distance euclidienne, on peut difficilement unifier le détail des opérations et des structures.

Employer différentes distances produit naturellement des cartes de distance et des squelettes d'aspect et de propriétés différentes. Cela nous permet de choisir la distance la plus appropriée à une application donnée. Dans la figure 62 on montre la géométrie des squelettes, ainsi que la forme recouverte très caractéristique. Evidemment la distance qui produit les meilleurs résultats est la plus isotrope, $d_{5,7,11}$.

Le temps de calcul est très limité, et les phases de l'algorithme peuvent être regroupées à 5 inspections de l'image. Certains algorithmes de squelettisation sont conçus directement sur machines parallèles [Ube93]. Notre méthode peut être parallélisée, mais avec une stratégie spécifique pour chaque étape. Les cartes de distances sont traitées au §2.4.4. L'extraction des points selle et de l'axe médian est une opération locale. Les phases de propagation et de parcours du squelette demandent une étude particulière.

Chapitre 7

Description de formes

Une forme binaire, pouvant être perçue comme la superposition de régions élongées, est décomposée en régions simples à travers son squelette. Le squelette pondéré est interprété comme une courbe 3D, où les coordonnées de chaque pixel sont les coordonnées planaires plus le poids. La courbe 3D est polygonalisée, chaque segment représentant l'échine d'une région élémentaire, i.e dont l'épaisseur change linéairement et dont l'orientation est fixe. De là les échines sont analysées, pour simplifier la décomposition du squelette et éviter la redondance. Certaines échines sont annihilées tandis que d'autres sont fusionnées.

Les éléments résultants sont utiles pour la représentation et la description de la forme. La décomposition peut être menée à différents niveaux de résolution, en jouant sur la polygonalisation et la fusion.

7.1 Introduction

Partant d'un système de représentation, la description est une étape importante pour la compréhension ou le traitement de formes.

La description est spécifique au type de scène à analyser (images binaires ou en niveaux de gris, volumes, mouvements, robotique). Nous nous intéressons au cas des images binaires 2D. Les deux grandes classes de codage d'un objet binaire sont les représentations par les régions, et les représentations par les contours.

Le contour d'une forme discrète, constitué d'une suite de points, est une information locale. De nombreuses applications peuvent s'en satisfaire, par exemple la décomposition en parties convexes à partir du contour [Ong92]. Les méthodes orientées contour sont généralement plus efficaces lorsqu'elles s'appuient sur une approximation du bord, qui est plus simplifiée, par exemple avec des segments de droite [Pav77], des segments et des arcs de cercles, ou avec des splines [Pav92].

Le codage des régions est une approche alternative et complémentaire au codage par les contours. La forme est dissociée par partitionnement ou recouvrement. L'exemple typique de partitionnement est le quadtree [Sam90]. Un deuxième exemple, qui n'est pas contraint par le maillage, est le pavage de Voronoï [Ber94]. L'axe médian (chapitre 5) et le squelette pondéré (chapitre 6) donnent un recouvrement de la forme ; on y revient dans la suite.

Les représentations axiales allient contours et régions. Elles sont dédiées à une classe de formes appelées *rubans*. Les rubans de Blum [Blu78] sont générés en faisant glisser un disque sur une courbe planaire. Les rubans de Brooks [Bro81] utilisent le même principe, mais avec des segments perpendiculaires à la courbe. Les rubans de Brady [Bra84] sont définis par des symétries locales. Ces trois familles de rubans sont comparées dans [Ros86, Pon88].

Pour évaluer une représentation, il est nécessaire de prendre en compte ses qualités de compression, son degré de réversibilité, les caractéristiques de la structure de données, le coût de calcul de la représentation, ainsi que son potentiel d'adaptation à différents traitements. En ce sens, il n'existe pas de représentation idéale, chacune d'entre elles se révélant mieux adaptée à certains objectifs.

Nos objectifs sont relatifs aux traitements des objets contenus dans une image, en termes d'interprétation et de manipulation.

L'interprétation de formes géométriques peut s'exprimer par un système de représentation fondé sur des formes élémentaires, qui par exemple coïncident localement avec les régions. Un graphe permet alors d'accéder à une description hiérarchisée.

La manipulation peut être la décomposition d'objets complexes en entités plus simples, ou le filtrage par la suppression de parties non significatives.

Nous excluons de ce chapitre les traitements géométriques et ensemblistes, qui ne relèvent pas de la description et de la compréhension de formes.

Un premier moyen d'appréhender le contenu d'une image est l'extraction de paramètres. Ainsi les moments et les projections sur les axes sont directement calculés à partir du squelette pondéré (avec d_8) dans [San90]. On mentionne d'autres paramètres géométriques, tels que l'aire, le périmètre et le centre de gravité, et les paramètres topologiques comme le nombre d'Euler et les nombres de connexité [Cha91], à partir desquels on peut avoir une évaluation de la complexité de la forme, et produire d'autres paramètres, tels que le facteur d'allongement ou le degré de convexité.

Le degré d'appariement est un élément descriptif pour la reconnaissance ; il est lié à une base de formes, nécessairement finie. Il peut être calculé à l'aide d'une image de distance [Bar77, Bor86b]. L'appariement peut également être effectué à partir de la polygonalisation du contour [Ste90].

Dans [Rom93], le contour est approximé par des B-splines quadratiques. Les axes de symétries locales en sont ensuite extraits, suivant l'approche des rubans de Brady [Bra84]. Par nature cette représentation est locale, très sensible aux bruits, et coûteuse à calculer. L'utilisation de B-splines lève ces problèmes. Les axes de symétrie sont ensuite exploités pour la décomposition de formes, fondée sur l'importance relative des régions et leurs relations spatiales. Mais certains phénomènes liés aux B-splines, et la non-homotopie des axes de symétrie, limitent cette méthode.

Disposant d'un squelette pondéré calculé dans le chapitre 6, notre objectif est de mettre à l'épreuve cette représentation, en développant une description souple et utilisable pour la manipulation et l'interprétation de formes [Thi94a, Thi94c].

Dans la bibliographie antérieure, on trouve dans [Sir91], une représentation uniquement fondée sur la polygonalisation d'un squelette binaire. Une méthode plus élaborée [Chi89] approxime un squelette pondéré (avec d_8) par des segments de droites et des arcs de cercles. L'appariement de deux graphes structurels quelconques, provenant de la polygonalisation d'un squelette, est développé dans [Dar94]. Le contexte de ces deux dernières méthodes est la reconnaissance de caractères.

Ces représentations tiennent peu compte de l'épaisseur des objets, et sont trop focalisées sur une même problématique. Nous effectuons au §7.2 l'approximation polygonale sur le squelette pondéré (multi-distance), en tenant compte de l'épaisseur. L'exploitation des propriétés du codage a lieu au §7.3. La simplification et la fusion de la description, dans le but de faciliter l'interprétation, sont présentées au §7.4.

7.2 Polygonalisation du squelette

7.2.1 Préliminaires

Le squelette pondéré est calculé au chapitre 6 sur des objets pouvant être perçus comme l'union de régions élongées (ou de rubans).

La classification des points de squelette en point extrémité, point normal et point de branchement est définie au §6.5.3. Un point extrémité est placé en correspondance avec le bout d'une région élongée de l'objet. Un point de branchement identifie la rencontre d'arcs du squelette, et est positionné au niveau de la superposition des régions élongées.

Les points extrémité et de branchement nous autorisent à interpréter le squelette comme la concaténation de *branches de squelette*. Ce sont des arcs du squelette, dont les points sont tous normaux, sauf les terminaisons. A ce propos, on appelle branche de squelette *périphérique* une branche délimitée par un point extrémité.

Lorsque tous les pixels de S sont des points normaux, alors le squelette est une simple courbe. Cette courbe est traitée comme une unique branche de squelette, dont les extrémités sont choisies arbitrairement parmi 2 points adjacents du squelette. Ces deux pixels sont considérés comme deux points de branchement, pour traiter toutes les branches de squelettes de façon homogène.

Une *région élémentaire* de l'objet B est un ensemble R, obtenu par transformation de distance inverse (§2.4.3) sur une section de branche de squelette, tel que :

- l'épaisseur locale de R change linéairement et monotoniquement le long de la section de branche de squelette ;
- les parties du contour communes à R et B sont des segments.

La section de branche de squelette correspondant à la région élémentaire R est l'*échine* de R. L'échine peut être raisonnablement représentée par les coordonnées et le poids de ses extrémités. En retour, une approximation de R peut être obtenue avec l'enveloppe convexe des deux disques associés aux extrémités de l'échine. Les disques ont des allures différentes suivant la distance adoptée, alors que la portion centrale de l'enveloppe est un trapèze.

Cortopassi a proposé le terme *gyxel* pour une région élémentaire, en s'appuyant sur la distance d_4 [Cor88]. Nous conservons cette appellation pour les distances de chanfrein. Un gyxel est représenté figure 63.

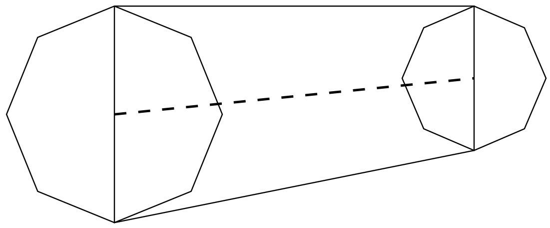


FIG. 63 - Un gyxel pour $d_{3,4}$ (avec l'échine en pointillé)

7.2.2 Approximation polygonale

Le squelette est traité branche par branche. La décomposition effectuée est équivalente à celle que l'on obtiendrait individuellement sur chaque région, résultant de la transformation inverse de distance sur chaque branche. Les branches du squelette sont en principe toutes significatives, suite à la phase d'élagage du §6.5.2.

Une structure de graphe est adoptée, pour mémoriser les terminaisons des branches de squelette et les relations spatiales entre branches.

Chaque branche du squelette est ensuite partitionnée au moyen d'une approximation polygonale, de telle sorte que chaque portion rectiligne du squelette constitue l'échine d'une région élémentaire.

Les points de division doivent être placés aux changements de direction sur le squelette qui reflètent un changement d'orientation au niveau du contour. Des points de division doivent aussi être placés lorsque la variation de l'épaisseur n'est plus linéaire ou monotone. Pour localiser ces deux types de points de division, nous considérons chaque branche de squelette comme un arc dans l'espace 3D, repéré par les coordonnées planaires et la distance normalisée.

La valeur *normalisée* d'un point du squelette étiqueté à p, est le plus petit entier k, tel que $k \geq p/a$, pour une distance $d_{a,b,\ldots}$.

Nous normalisons la distance pour traiter de façon uniforme les trois coordonnées, en autorisant une variation d'une unité dans chacune des trois directions, lorsque l'on passe d'un point du squelette à l'un de ses voisins. Grâce à cela, la branche de squelette est encore un arc connexe dans la représentation 3D.

L'approximation polygonale est effectuée avec l'algorithme récursif de Pavlidis [Pav77]. Sa mise en œuvre est très simple, et le résultat n'est pas influencé par le sens de parcours de l'arc. D'autres propriétés intéressantes liées à cet algorithme viennent conforter son choix au §7.3.

Les terminaisons de la branche courante, notées v_i et v_f, sont acceptées comme points de division, et sont stockées dans la structure de données. De là sont identifiés récursivement de nouveaux sommets (mémorisés dans la structure de données). La distance euclidienne $d_{3D}(p)$ entre chaque pixel p de la branche de squelette et la droite 3D (v_i, v_f) est calculée. Celui des points p qui maximise d_{3D} est pris comme nouveau sommet v, à condition que $d_{3D}(v)$ soit supérieur à un seuil θ, fixé a priori. Le processus est relancé sur les sous-arcs (v_i, v) et (v, v_f) ; il s'arrête lorsque $d_{3D}(v) \leq \theta$.

Si plusieurs pixels de la branche du squelette maximisent $d_{3D}(p)$, les deux points v_1 et v_2 qui sont respectivement les plus proches de v_i et v_f sont acceptés comme sommets, et le processus est relancé sur les 3 sous-arcs (v_i, v_1), (v_1, v_2) et (v_2, v_f). En effet, accepter tous les points qui maximisent $d_{3D}(p)$ pourrait engendrer une approximation polygonale avec trop de sommets, pas tous nécessairement significatifs. A contrario, n'accepter qu'un seul point rendrait la décomposition dépendante de l'ordre dans lequel les pixels sont examinés.

Nous remarquons que le calcul de la racine carrée nécessaire pour obtenir la distance euclidienne $d_{3D}(p)$ peut être évité, puisque le même résultat est obtenu en comparant le carré de la distance et le carré du seuil.

La valeur du seuil θ est fixée selon la tolérance considérée comme acceptable pour une tâche donnée. Le seuil doit être assez petit pour favoriser un recouvrement fidèle des régions élémentaires. La valeur $\theta = 1.5$ s'est révélée expérimentalement correcte ; elle est utilisée figure 64 sur une forme synthétique.

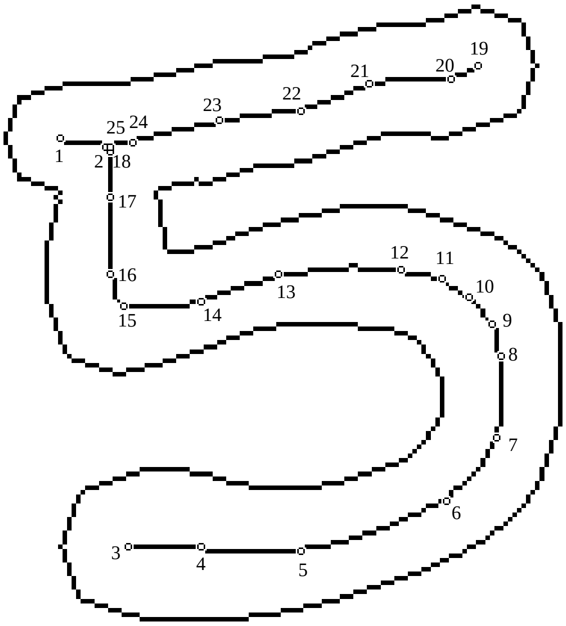


FIG. 64 - Partition du squelette pour $d_{3,4}$ et $\theta = 1.5$. Les sommets sont numérotés.

7.3 Propriétés du partitionnement

7.3.1 Aspect multirésolution

Des algorithmes de calcul d'un squelette multirésolution ont été proposés dans la littérature [Arc81, Dil87]. Ils tenaient compte de la courbure du contour pour détecter les points extrémité, et fournissaient différents squelettes, selon la présence ou la longueur de branches périphériques. De là était possible une hiérarchie entre les branches de squelette, et donc entre les régions élongées.

Dans la méthode que nous proposons, différentes représentations des mêmes branches de squelette sont réalisées à différents degrés, en faisant varier les seuils (croissants) de la polygonalisation [Thi93b]. Le but est d'obtenir une description multirésolution des régions élongées associées aux branches de squelette. Le seuil le plus petit donne le plus haut degré de description.

En fait, l'approximation polygonale n'est faite qu'une seule fois, en prenant la plus petite valeur des seuils que l'on veut tester. Grâce à l'algorithme récursif de découpage de Pavlidis, les sommets candidats au plus bas degré sont déjà des sommets acceptés dans les résolutions supérieures. L'information nécessaire pour identifier les sommets dans les approximations successives est disponible, car dès qu'un sommet

p est mémorisé à une étape donnée dans le processus récursif, nous stockons la distance $d_{3D}(p)$. Pour obtenir une description plus grossière avec un seuil plus grand, il suffit de le comparer à $d_{3D}(p)$.

i	x	y	k	m	t
1	30	40	15	5	b
2	20	38	12	5	e
3	35	128	16	5	e
4	51	128	17	1	n
5	73	129	14	4	n
6	105	118	12	5	n
7	116	104	14	3	n
8	117	86	14	5	n
9	115	79	16	1	n

i	x	y	k	m	t
10	110	73	15	3	n
11	104	69	14	1	n
12	95	67	15	5	n
13	68	68	12	2	n
14	51	74	11	3	n
15	34	75	16	5	n
16	31	68	15	1	n
17	31	51	11	2	n
18	31	41	14	5	b

i	x	y	k	m	t
19	112	22	14	5	e
20	106	25	13	1	n
21	88	26	13	1	n
22	73	32	12	2	n
23	55	34	14	1	n
24	36	39	13	1	n
25	31	40	15	5	b

FIG. 65 - Représentation compacte d'une multirésolution du squelette

Nous proposons une représentation compacte des multiples résolutions effectuées sur une forme [Thi93b]. Nous associons à chaque sommet de numéro i un quintuplet (x, y, k, m, t) où (x, y) sont les coordonnées cartésiennes, k la distance normalisée, m un compteur de présence du point aux différents niveaux de résolution, et enfin t le type du sommet. La représentation compacte est illustrée dans la table de la figure 65, à partir des sommets numérotés de la figure 64. Les degrés d'approximations avec $\theta = 2.5$, $\theta = 4$, $\theta = 6$ et $\theta = 8$ sont directement déduits de l'approximation initiale où $\theta = 1.5$. Les régions associées sont représentées figure 66.

La complexité de l'algorithme de Pavlidis est en $O(n \log n)$, où n est le nombre de points du squelette. La recherche des niveaux de sous-résolution est un simple parcours d'une approximation plus fine. L'utilisation de la distance normalisée nous permet de traiter la polygonalisation indépendamment de la distance de chanfrein ayant permis le calcul du squelette.

Au §7.4 on est amené à distinguer deux niveaux de résolution, avec $\theta_1 = 1.5$ et $\theta_2 = 2$, pour distinguer les sommets qui affinent les détails des sommets 'robustes'.

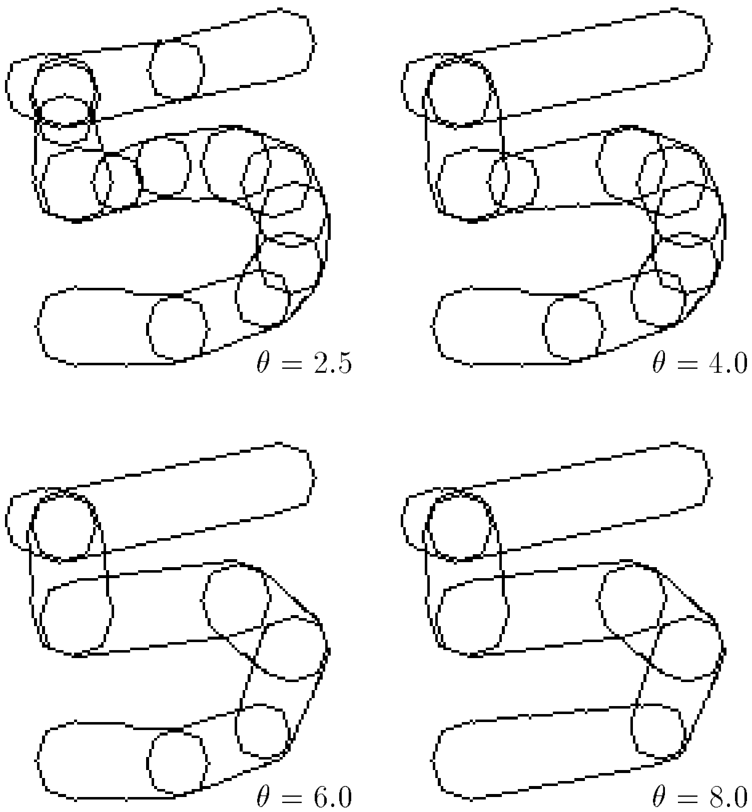


FIG. 66 - Quatre niveaux de décomposition

7.3.2 Inversibilité

Il n'est pas indispensable de regénérer les gyxels, représentés par les échines, pour extraire des propriétés géométriques (telles que l'aire ou le périmètre), ou détecter des propriétés de la forme (comme l'orientation ou la rectangularité). Ces caractères peuvent être facilement déduits à partir des coordonnées 3D des sommets.

Il peut être cependant utile de recréer la forme, pour illustrer des manipulations sur les gyxels, ou simplement avoir une représentation réversible. Deux techniques sont applicables, par la transformation de distance inverse, ou par le tracé des gyxels. La structure de données doit contenir les coordonnées planaires des sommets, ainsi que le poids issu de l'image de distance (le poids normalisé serait moins précis).

Dans la première méthode, on trace chaque échine avec un algorithme de segment discret ad hoc, qui pondère linéairement les pixels tout au long du segment, depuis le poids d'une extrémité jusqu'au poids de la seconde. On redessine ainsi une forme très proche du squelette pondéré de départ (mais ce n'est pas tout à fait un squelette). Il suffit de lui appliquer une transformation inverse de distance (§2.4.3) pour retrouver une forme très proche de l'objet initial.

La seconde méthode consiste à dessiner analytiquement le bord de chaque gyxel. Pour les distances classiques à 2 et 3 coefficients, le calcul des enveloppes convexes est faisable par la géométrie algorithmique, connaissant les pentes de chaque côté (§3.3.2). Pour des masques plus grands, il vaut mieux schématiser le gyxel en employant directement des cercles.

La première méthode reste dans le domaine des images de distances, alors que la seconde se rapproche plus de la géométrie algorithmique. L'une est efficace pour régénérer toute la forme et est simple à mettre en œuvre, tandis que l'autre est plus appropriée pour dessiner séparément les régions élémentaires.

La qualité de la réversibilité dépend directement du seuil de polygonalisation. Le but principal de cette méthode n'est cependant pas d'approximer un objet le plus fidèlement possible (on a déjà perdu un peu lors du calcul du squelette puis de son élagage), mais de fournir une représentation simplifiée de la forme.

En ce sens, la polygonalisation du squelette est bien adaptée, nous ramenant à une expression vectorielle de la forme, très compacte et souple, qui nous autorise à développer de l'algorithmique efficace pour la description de la forme.

7.4 Simplification et fusion

Certaines régions représentées par des échines de squelette sont (presque complètement) recouvertes par des régions adjacentes, en particulier lorsque l'approximation polygonale est réalisée avec un seuil bas. Pour éviter la redondance, les régions recouvertes ne devraient plus être prises en compte, ou alors fusionnées avec des régions adjacentes [Thi93c]. Pour ce faire, on effectue dans leur ordre d'apparition les opérations ci-après, réalisées sur les échines.

7.4.1 Annihilation des échines courtes

Toutes les échines *courtes* (pour fixer les idées, disons les échines de 4 pixels et moins) sont examinées. Les échines courtes, dont les échines contigües ne sont pas courtes, sont appelées échines courtes *isolées*.

Toute échine courte s_i, délimitée par deux points normaux, est annihilée, en déplaçant les deux sommets, partagés avec les échines contigües s_{i-1} et s_{i+1}, vers une position commune dans l'espace 3D. Cette position est soit le barycentre de s_i, soit l'intersection de s_{i-1} et s_{i+1}, suivant l'angle entre s_{i-1} et s_{i+1}. Le pixel commun aux deux nouvelles échines, obtenues en modifiant s_{i-1} et s_{i+1}, n'est pas retenu comme un sommet, si les deux nouvelles échines sont alignées, dans la limite de la tolérance définie lors de l'approximation polygonale. Lorsque c'est le cas, les deux nouvelles échines sont fusionnées en une échine unique, qui représente encore une région élémentaire.

Toute échine courte délimitée par un point extrémité ou par un point de branchement est supprimée. Cependant, lorsque s_i est délimité par un point de branchement, la trace de s_i doit être gardée, pour respecter la position relative des branches de squelette, et par là des régions correspondantes dans la décomposition de la forme. Dans ce cas, l'échine s_i joue le rôle de lien, mais ne possède plus de pouvoir de représentation dans la décomposition.

7.4.2 Annihilation des échines superflues

Des échines qui ne sont pas courtes peuvent aussi correspondre à des régions presque complètement recouvertes par des régions adjacentes. Ces échines sont *superflues* pour la représentation et la description de la forme.

Une échine peut être considérée comme superflue, si l'enveloppe des disques centrés en ses extrémités (le gyxel) ne diffère pas significativement de l'union des deux disques, ce qui peut être le cas si les deux disques se recouvrent partiellement. Nous évaluons le recouvrement de la façon suivante. Soit d la distance de chanfrein entre les deux extrémités de l'échine, et l_1 et l_2 leur poids respectifs. L'échine est jugée superflue si la condition de recouvrement ${l_1}^2 + {l_2}^2 \geq d^2$ est satisfaite.

Si les extrémités d'une échine superflue s sont deux points normaux et si les deux échines contigües sont toutes deux non superflues, alors s est annihilé en suivant la même stratégie que pour les échines courtes du §7.4.1.

Les effets de l'annihilation d'une échine sont illustrés figure 67.

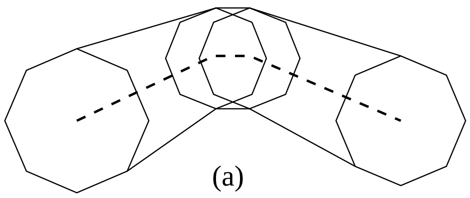


FIG. 67 - Annihilation d'une échine : dans la partition initiale du squelette (a) en trois échines (pointillé), le gyxel central non significatif est supprimé (b).

7.4.3 Croisement de régions

Les échines ayant un point de branchement en commun identifient des régions élémentaires qui se croisent. Les échines sont significatives si les régions élémentaires correspondantes ne se recouvrent pas notablement. Dans le cas contraire, leur présence n'est nécessaire dans le squelette que pour refléter les relations spatiales entre les branches de squelette. Comme auparavant au §7.4.1, ces échines sont ramenées au rang de lien, sans aucun pouvoir de représentation dans la décomposition.

Cependant il peut se produire que considérer l'une de ces échines comme un lien n'est pas suffisant, pour éliminer la redondance dans la décomposition, et pour améliorer la stabilité de la décomposition en rotation. Ainsi des échines successives, localisées sur la même branche de squelette, n'ont parfois aucun pouvoir de représentation. En effet, la branche de squelette peut avoir été excessivement fragmentée durant la phase de polygonalisation, si bien que la portion de la branche de squelette, qui dans la décomposition finale ne devrait avoir qu'un rôle de lien, se retrouve morcelée en de multiples fragments. D'autre part, l'approximation polygonale du squelette, extraite avec un seuil bas, est rarement stable lorsque l'orientation de la forme initiale est changée. De là, une interprétation différente, quant à l'état de lien, pourrait survenir si la forme est rotée.

Les sommets extraits pour le seuil bas habituel $\theta_1 = 1.5$, qui demeurent des sommets pour un seuil plus grand $\theta_2 = 2$, sont davantage portés à être présents dans la décomposition du squelette lorsque la forme est rotée, et à identifier correctement les portions de branches de squelette ne devant jouer qu'un rôle de lien dans la décomposition finale. Ces sommets, dits *stables*, sont identifiés en se référant au quadruplet (x, y, label, d_{3D}) stocké pour chaque sommet durant l'approximation polygonale (§7.2).

Soit b_1, b_2, ..., b_n les branches de squelette partageant le point de branchement v_b, et soit v_i ($i = 1$ à n) le premier sommet localisé sur b_i subsistant dans l'approximation polygonale pour θ_2. La partie de b_i délimitée par v_b et v_i est déclarée sans pouvoir de représentation, si pour un sommet v_j, localisé sur b_j ($j \neq i$), la condition de recouvrement du §7.4.2 est satisfaite pour les disques centrés en v_i et v_j. Si c'est le cas, la partie est maintenue à un rôle de lien. Dans le cas contraire, la condition de recouvrement est testée entre le disque centré en v_b et entre chaque disque de la partie de b_i, délimitée par v_b et v_i, pour identifier le plus grand lien et pour simplifier la décomposition. L'analyse des nœuds est illustrée figure 68.

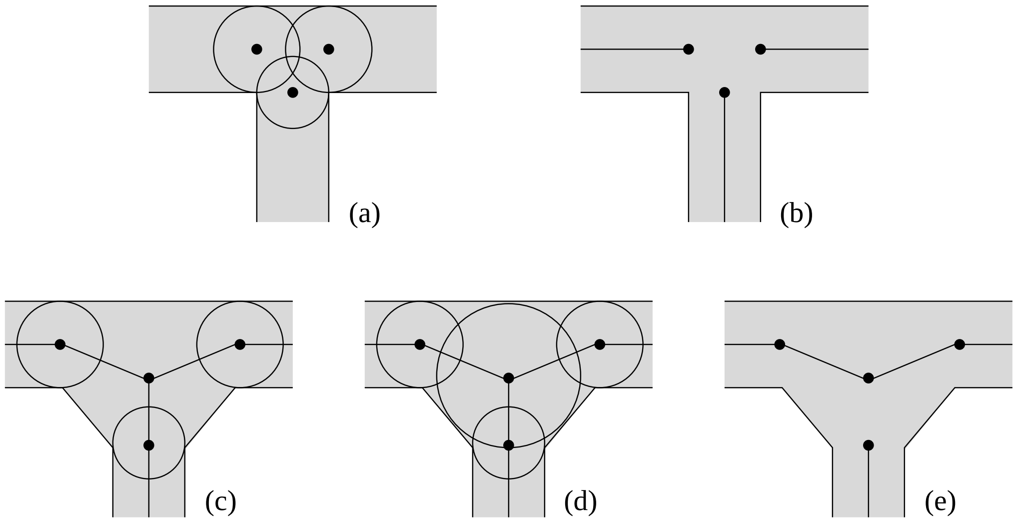


FIG. 68 - Analyse des nœuds : les échines non représentatives sont supprimées en (b) et (e). Les disques extrêmes se recouvrent en (a), mais pas en (c) ; le disque centré sur le point de branchement recouvre l'un des trois disques en (d).

7.4.4 Fusion d'échines

A cette étape du processus, toutes les échines subsistantes contribuent à retrouver la forme. En fusionner certaines pourrait être utile pour réduire le nombre de régions, qui vont constituer les primitives de la description de la forme. La réduction du nombre de régions accroît généralement la stabilité en rotation, et fournit des résultats plus en accord avec l'intuition humaine.

Des régions élémentaires suffisamment proches en épaisseur et en orientation peuvent être fusionnées, en regroupant les échines correspondantes. Une région fusionnée, bien que n'étant plus une région simple, peut encore être décrite à partir des coordonnées 3D des sommets des échines regroupées. Différentes concaténations sont possibles, en faisant varier un degré de tolérance, et elles produiront différentes décompositions de la forme. Les décompositions ont toutes le même pouvoir de représentation, puisque chaque région fusionnée est l'union de ses régions élémentaires. Ce n'était pas le cas lorsque différentes décompositions étaient obtenues par changement de seuil dans la polygonalisation du squelette (cf figure 66).

Le fait de disposer de plusieurs décompositions du même objet est utile pour faciliter la reconnaissance de la forme, et nous permet de sélectionner une décomposition plus appropriée à la solution d'un problème spécifique.

L'orientation et l'épaisseur de deux régions élémentaires contiguës sont fidèlement reflétées par leurs échines. Si les échines, représentées par des segments 3D, sont alignées dans les limites d'une tolérance choisie, alors ces régions sont semblables et devraient être fusionnées.

Chaque paire d'échines successives, appartenant à la même branche de squelette, est examinée. Soit $(v_{i-1},\, v_i)$ et $(v_i,\, v_{i+1})$ les sommets délimitant la paire courante. Soit D_i la distance euclidienne entre v_i et la droite $(v_{i-1},\, v_{i+1})$, et L_i la distance euclidienne entre v_{i-1} et v_{i+1}. Un booléen F est initialisé à 0 ; il est mis à 1 en correspondance avec chaque sommet v_i, tel que D_i/L_i est inférieur à un seuil de fusion φ, fixé a priori.

Soit v_1, v_2, …, v_n une suite de sommets pour lesquels le booléen $F = 1$. De plus, soit v_0 et v_{n+1} les deux sommets précédant immédiatement v_1 et suivant immédiatement v_n.

- Si $n = 1$, alors les deux échines $(v_0,\, v_1)$ et $(v_1,\, v_{n+1})$ sont fusionnées.

- Si $n > 1$, alors pour chaque i de 1 à n, la distance D'_i entre v_i et la droite $(v_0,\, v_{n+1})$ est divisée par la longueur L'_i du segment $[v_0,\, v_{n+1}]$. Si pour chaque sommet on a $D'_i/L'_i < \varphi$, alors toutes les échines sont fusionnées. Dans le cas contraire, on tente de fusionner avec la même stratégie, les suites $v_1 \ldots v_{k-1}$, $v_k \ldots v_j$ et $v_{j+1} \ldots v_n$, en incrémentant k et en décrémentant j de concert.

Les sommets délimitant une suite d'échines fusionnées sont considérés comme les extrémités de l'échine complexe résultante. Soulignons que ces sommets maintiennent leur pouvoir de représentation, car la région associée à l'échine complexe est l'union des régions élémentaires associées aux échines fusionnées.

La valeur du seuil de fusion φ dépend de la tolérance désirée. Expérimentalement nous avons adopté la valeur de défaut $\varphi = 0.25$. Des seuils plus grands peuvent favoriser la fusion.

Nous prenons comme exemple une coupe de joint en caoutchouc, qui est squelettisée figure 69, avec un seuil d'élagage de 3. Les squelettes sont polygonalisés avec le seuil $\theta = 1.5$. Chaque segment code pour une région (un gyxel). L'ensemble de ces régions est montré figure 70 ; l'information est très redondante. Aux figures 71 et 72, 2 valeurs de fusion différentes ont été employées. Notons que contrairement à la figure 66, les régions ne sont pas élémentaires.

La possibilité de fusion entre échines partageant un sommet de branchement devrait aussi être prise en compte, de façon à ce que la décomposition finale de la forme ne soit pas conditionnée par la décomposition préliminaire du squelette en ses branches le constituant. Des travaux complémentaires sont menés en ce sens.

7.5 Conclusion

Dans ce chapitre, nous avons exposé une méthode de décomposition de forme se basant sur la décomposition de son squelette. La méthode est adéquate pour des formes pouvant être perçues comme l'union de régions élongées ; elle pourrait être employée par exemple dans le contexte de l'analyse d'un document, pour classifier les symboles alphanumériques qu'il contient.

L'adéquation de la méthode dépend de la signification des donnés de départ, qui est le squelette pondéré multi-distance développé au chapitre 6. Ce squelette inclut en particulier une phase d'élagage, pour éviter les branches parasites. Cependant des distorsions peuvent apparaître, indépendamment de la technique de squelettisation, qui sont intrinsèquement liées à l'inclusion de l'axe médian dans le squelette (sans parler de la forme des boules de la métrique discrète employée). La polygonalisation élimine les petites distorsions dues aux bruits du contour ; les phases d'annihilation et de fusion d'échines répondent aux distorsions plus larges dues à l'axe médian.

La stabilité de la décomposition en rotation est une présupposition indispensable pour toute application où l'orientation de la forme n'est pas connue a priori. Cette

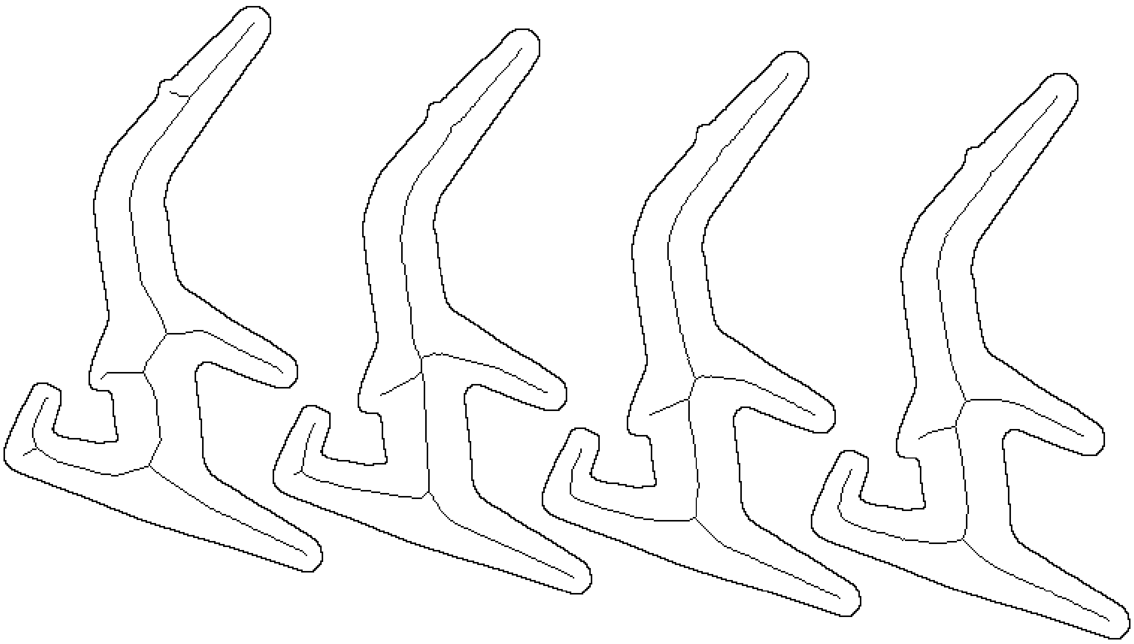


FIG. 69 - Squelettes pour d_4, d_8, $d_{3,4}$ et $d_{5,7,11}$ (de gauche à droite), et bord externe de l'objet initial (un joint en caoutchouc).

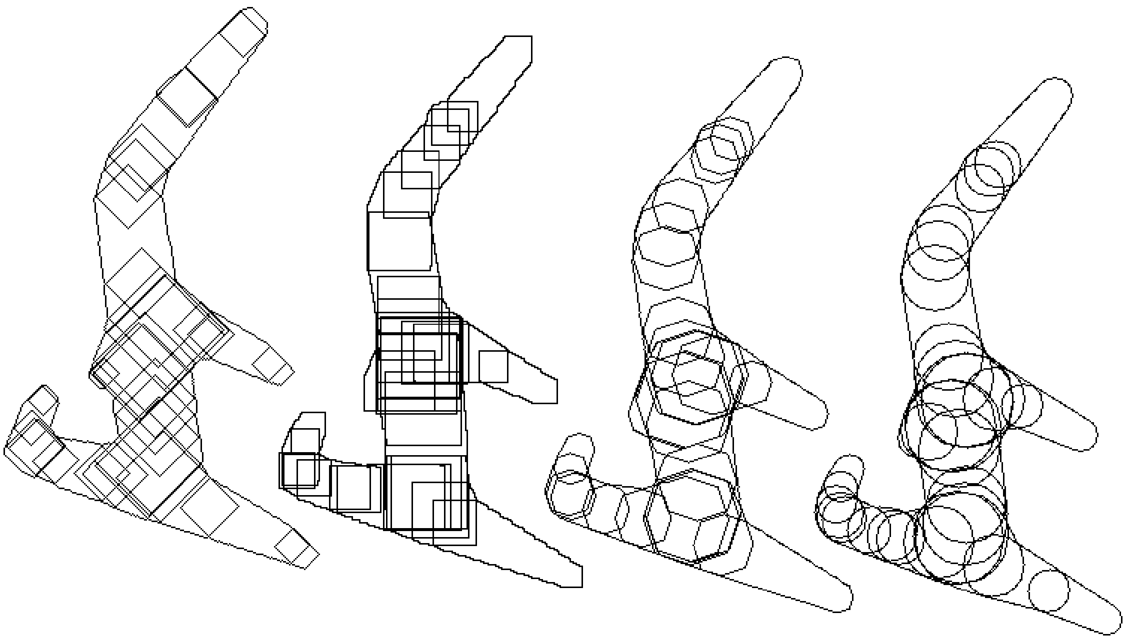


FIG. 70 - Toutes les régions correspondant à la polygonalisation du squelette

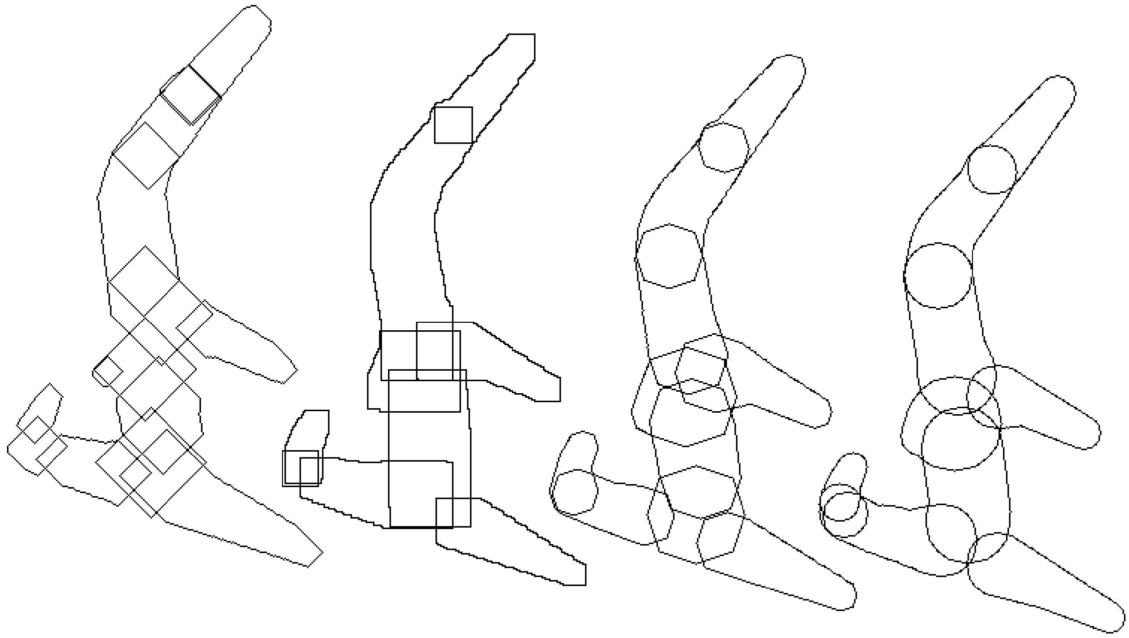


FIG. 71 - Décompositions avec $\varphi = 0.15$

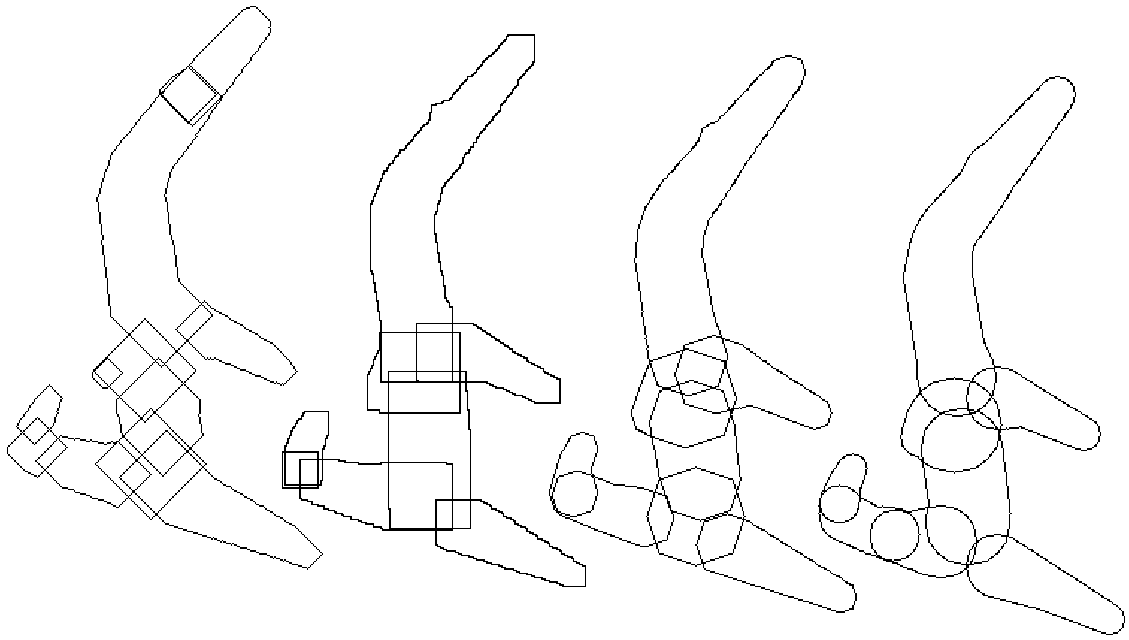


FIG. 72 - Décompositions avec $\varphi = 0.25$

stabilité peut être appréciée par le choix de la distance discrète. Elle peut être encore favorisée par les phases d'annihilation et de fusion, qui réduisent aux plus significatifs le nombre de composants dans la décomposition.

Une caractéristique intéressante de la méthode est la possibilité d'obtenir des décompositions à différents niveaux. Cela peut être accompli en changeant les seuils de l'approximation ou de la fusion. Dans le premier cas, la représentation obtenue n'a pas le même pouvoir de représentation. En effet, les éléments décomposés du squelette sont considérés comme les échines de régions simples, indépendamment du seuil employé. Les représentations variées peuvent être données de façon compacte. Dans le second cas, les décompositions diffèrent les unes des autres par le nombre de leur régions fusionnées, mais ont toutes le même pouvoir de représentation. Une région fusionnée avec un seuil faible est plus simple à décrire, mais la description complète de la forme en termes de ses régions fusionnées est moins maniable.

Le temps de calcul du procédé est très modeste, car tous les calculs sont effectués sur un faible volume de données (les points du squelette, puis après, les sommets de la polygonalisation), qui sont stockées sous forme vectorielle.

Nous sommes bien conscients que notre méthode peut être améliorée, spécialement dans la phase de fusion. Cette phase devrait être un point de départ pour pratiquer de meilleures décompositions, adaptées pour une application spécifique. Il est en effet difficile de concevoir une méthode de décomposition totalement générale.

Pour compléter notre propos, nous mentionnons certains points qui sont à l'étude. Nous essayons de prendre en compte plus d'information, provenant toujours des coordonnées des sommets de la polygonalisation du squelette, qui pourrait provoquer une fusion de régions plus en accordance avec l'intuition humaine. Ces informations concernent la détection de sommets dont le signe de la courbure change, ou dont la valeur de distance est minimale. La détection de tels sommets pourrait nous aider à ne pas fusionner des objets en forme de 'S' ou de clepsydre. De même la phase de décomposition du squelette pourrait être mieux adaptée au traitement de formes arrondies. Par exemple, à la place d'une approximation polygonale, on pourrait employer une technique d'approximation de courbes, telle que les B-splines.

Chapitre 8

Développements

Ce chapitre présente les outils développés et leurs environnements d'exploitation, puis leur utilisation dans le cadre d'applications.

Deux logiciels de traitement d'images sont décrits. Pour chacun d'entre eux, le contexte et les objectifs sont posés, puis le principe du maniement de l'interface est donné. Nous abordons ensuite la partie développement, et nous achevons avec l'intégration des opérateurs.

Des applications en imagerie sont par ailleurs évoquées, traitant la décoalescence d'agrégats, le béton cellulaire et les fibres de papiers.

8.1 Introduction

L'analyse d'images requiert de multiples expérimentations ; des interfaces conviviales et puissantes sont donc nécessaires, pour les manipulations et la programmation des opérateurs de traitement d'images.

L'environnement de développement relatif à ce travail a varié, suivant les laboratoires d'accueil et les applications spécifiques ; certaines ont été programmées pour une carte d'acquisition donnée, ou d'autres dans un langage fixé. Les différentes plate-formes ont été :

- stations Sun (langage C) ;
- PC, ou PC et carte Matrox (C, Pascal, assembleur) ;
- stations Apollo (Pascal).

Les logiciels que nous allons décrire, IPS et GOLT, ont été créés respectivement sur station Sun et sur PC. Ils ont permis de mettre au point les algorithmes qui sont au cœur des applications en imagerie closant ce chapitre.

8.2 Logiciel IPS

8.2.1 Historique

L'équipe INFODIS du laboratoire TIMC-IMAG de Grenoble développe depuis 1985 le logiciel *Image Package Software*. Il regroupe le savoir faire du laboratoire en analyse et traitement d'images. Il est composé autour d'une bibliothèque d'opérateurs et d'une interface multifenêtres chargée de l'activation des opérateurs.

Le but du projet IPS est triple, avec l'illustration des méthodes, leur tests (protocoles, faisabilité) et l'autoformation. Il doit faciliter la recherche et les interactions entre les domaines de compétences, en mettant à la disposition de tous les travaux de chacun.

Le logiciel a suivi les différentes plate-formes matérielles qui ont été utilisées au laboratoire : stations Apollo (1985), stations graphiques IBM 6150 (1987), Macintosh (1990). Nous nous sommes personnellement chargé de l'implémentation d'IPS sur les stations Sun, avec le système UNIX[1] et le gestionnaire OpenWindows[2], ce dernier

[1]Système multi-tâches et multi-utilisateurs.
[2]Il offre aux développeurs une vaste panoplie d'outils, très agréables à programmer.

étant une sur-couche du système graphique X11[3]. Notre écriture a démarré en 1992, avec le langage C.

L'expérience des logiciels de recherche dédiés au traitement d'images nous a conduit à reformuler complètement la logique de l'interface. Le but poursuivi était de rendre le maniement du programme le plus intuitif possible, pour qui est familiarisé avec OpenLook.

IPS est maintenant distribué en deux versions :

- La version globale (G.IPS). Elle regroupe une bibliothèque de quelque 200 opérateurs généraux et spécifiques au laboratoire, qui proviennent essentiellement des plate-formes antérieures. Il s'agit de transformations géométriques, algébriques ou locales, morphologie mathématique, segmentation, représentation par régions ou par contours, Fourier, pyramides, etc.

- La version personnelle (P.IPS). Elle comprend l'interface proprement dite, et un *Builder* qui facilite l'intégration de nouveaux opérateurs. Chaque utilisateur peut donc définir sa version autonome, tout en ayant la possibilité de communiquer avec d'autres P.IPS et avec G.IPS.

8.2.2 Description de l'interface

L'interface de IPS a été conçue comme un ensemble d'objets — image, opérateur, macro, graphe, etc — qui ont chacun leur menu contextuel et leur icone appropriés. Les icones peuvent être sélectionnés, déplacés, superposés, dupliqués, etc. La figure 73 montre l'allure de l'interface.

La barre de commandes contient les menus *File*, *New*, *Oper*, *Macro*, *Extra* et *Help*. Le menu *New* permet de demander la création d'un icone (image, macro, etc). Le menu *Oper* contient la hiérarchie des opérateurs ; le premier niveau du menu est punaisé en bas à gauche de la figure 73.

La fenêtre principale contient les icones. Dans notre exemple on peut voir 9 icones image, 1 icone macro (*Skeleton Comp*) et 2 icones opérateur (*Rotation* et *Area Threshold*).

[3]Système de fenêtrage orienté réseaux, qui introduit l'entité *display*, correspondant à un clavier, une souris et un ou plusieurs écrans ; il permet l'exécution et l'affichage à distance.

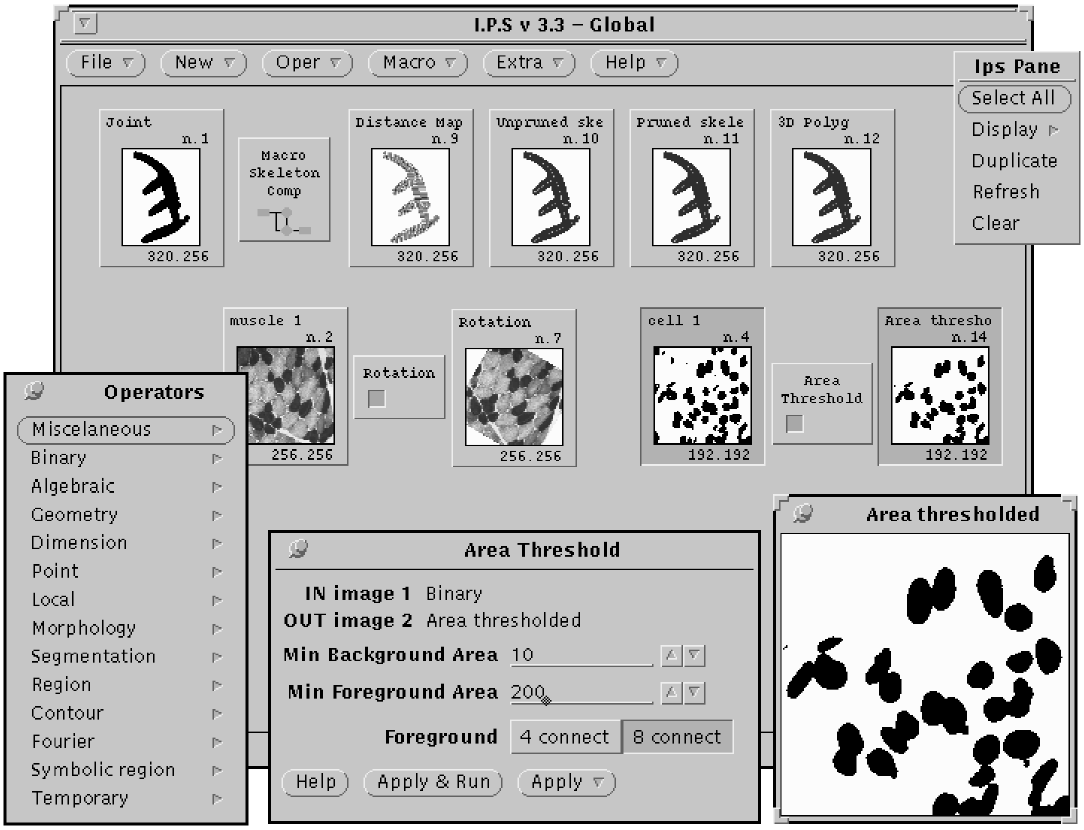


FIG. 73 - L'interface IPS

Le menu contextuel d'un icone image permet d'ouvrir des fenêtres d'édition et de changer les propriétés (taille, palette de couleurs, nom). On peut également éditer l'image en double-cliquant sur l'icone.

L'éditeur inclus dans IPS permet les manipulations classiques et quelques dessins, le zoom et le mode loupe. Dans ce dernier sont superposables les valeurs d'une image avec les couleurs d'une autre, ce qui est utile par exemple pour localiser l'axe médian sur une image de distances.

Chaque opérateur possède une boîte de paramètres. La boîte de l'opérateur *Area Threshold* est punaisée en bas de la figure 73. Elle précise le nombre d'images en entrée et en sortie, et demande les seuils d'aire et la connexité à appliquer.

Pour exécuter un opérateur, il faut sélectionner dans l'ordre les images en entrée (obligatoire) et en sortie (facultatif). Les images non sélectionnées en sortie sont créées (puis ajoutées dans la sélection), sinon elles sont réutilisées, au besoin en changeant leur taille.

Prenons l'exemple de l'opérateur *Area Threshold* figure 73. L'image #4 a été sélectionnée en entrée, puis la demande d'exécution a créé en sortie l'image #14, qui est éditée en bas à droite.

Si maintenant on modifie les paramètres de l'opérateur et qu'on l'exécute à nouveau sans changer la sélection, l'image #4 sera encore lue en entrée, et l'image #14 sera réutilisée. Lorsque l'exécution sera achevée, l'aspect de l'icone ainsi que le contenu de la fenêtre d'édition seront automatiquement mis à jour.

Il est possible d'attacher la sélection à un opérateur par le biais de son menu contextuel. Un double-clic sur l'opérateur lance alors son exécution. Chaque opérateur contient un voyant dans son coin inférieur gauche. Il est vert quand il est au repos, bleu au repos avec un attachement, et rouge pendant l'exécution.

Le même code de couleur est utilisé pour le petit pictogramme qui orne la partie inférieure des icones macro. Ces dernières permettent de mémoriser une séquence d'opérateurs, qui utiliseront leurs images attachées, ou remplaceront dans leur attachement les images sélectionnées en entrée de la macro.

8.2.3 Développement

Le menu décrivant la hiérarchie des opérateurs est contenu dans un fichier ressource, ce qui permet de modifier son organisation sans rien recompiler.

La manipulation des icones est entièrement gérée par le noyau d'IPS. Il est naturellement possible d'ajouter de nouveaux objets en définissant simplement leur icone et leurs méthodes. Un icone *diagramme de Voronoï* est en cours d'implémentation, et des icones *histogramme* et *masque de convolution* sont en projet.

Dans cette interface, tout est dynamique. Un mécanisme interne très important est le dispositif de messagerie qui a été implémenté entre les icones et les fenêtres. Tout icone ou fenêtre peut s'y abonner, et déclarer de qui elle peut recevoir des messages. Le but est le suivant : chaque image peut être éditée un nombre quelconque de fois ; une image peut être superposée à une autre en mode loupe. De plus une fenêtre de propriétés de l'image peut être affichée, de même qu'un repère de l'image pour telle ou telle fenêtre d'édition. Supposons que dans l'une d'elles, nous changions la palette de couleur. Cette fenêtre va simplement envoyer le message 'la palette a changé', qui sera diffusé par IPS à tous les abonnés, qui réagirons en conséquence. Un autre exemple est le cas d'un opérateur qui change la taille d'une image. Il envoie le message 'nouvelle taille', qui est diffusé par IPS, et tous les objets se mettent d'eux-mêmes à jour. Un dernier exemple : lorsqu'on demande à un icone image de se détruire, il envoie le message 'destruction' à la ronde, et chaque objet qui interagit avec cet icone se détruit aussi. L'icone lui-même reçoit le message et c'est là seulement qu'il disparaît.

Nous avons profité de la puissance offerte par X11 pour permettre la communication entre IPS et PIPS : il est parfaitement possible de transmettre des icones de l'un à l'autre, par un simple 'glisser-lâcher' de la souris. Cela évite de reprogrammer tous les opérateurs dans chaque version de PIPS, et permet de travailler avec plusieurs bibliothèques d'opérateurs à la fois.

Ce type de travail n'est jamais terminé, et de nombreux projets sont en attente. Ainsi le portage de l'interface en Motif, plus répandu que OpenWindows, mais pas trop différent car basé sur X11 ; la séparation de l'interface et des opérateurs, qui seraient regroupés en bibliothèques autonomes, selon un principe client/serveur ; la définition d'un langage de macro-commandes complet.

8.2.4 Versions personnelles

La grande force d'IPS est de faciliter l'intégration de nouveaux opérateurs, grâce à un *Builder*[4]. Tout utilisateur désirant développer sa version personnelle PIPS, lance un programme d'installation, qui crée les répertoires, importe les librairies et le Builder. Le Builder permet de connecter très simplement de nouveaux opérateurs, par un jeu de questions et d'étapes successives. Le menu des opérateurs est édité ; on demande un nouveau sous-menu, ou la création d'un opérateur. Il faut rentrer son nom ainsi qu'un nom de fonction pour le langage de macrocommandes, qui sert en même temps d'identificateur. Le nombre d'images et leurs légendes sont ensuite demandés, puis le nombre de paramètres et leurs types respectifs (entier, réel, potentiomètre, menu, chaîne de caractères, case à cocher, etc). Le langage de l'opérateur est enfin précisé, parmi le C, le Pascal, le Fortran et le C++ ; l'appel de l'opérateur et la définition de la boîte de paramètres sont alors générés automatiquement dans une fenêtre d'édition, et l'on peut au besoin compléter l'appel.

Le code généré allant appeler l'opérateur est très lisible ; en effet, le noyau d'IPS a

[4] Le Builder et le rapatriement des opérateurs ont été réalisés par E. Piasentin, stagiaire en M.S.T Expert en Systèmes Informatiques, juin–juillet 1992 et juin–août 1993.

été systématiquement écrit par le biais de fonctions à arguments variables, s'inspirant largement en cela de la programmation d'OpenWindows. Un appel d'opérateur est donné figure 74.

```
CO_BEGIN_CONNECT (AreaThreshold)

    CO_BEGIN_DEF
        IPS_NAME,       "Area|Threshold",
        IPS_IMAGE_IN ,  1, "Binary",
        IPS_IMAGE_OUT,  2, "Area thresholded",
        IPS_PAR_INT,    1, "Min Bg Area", 0, 65535, 10,
        IPS_PAR_INT,    2, "Min Fg Area", 0, 65535, 10,
        IPS_PAR_CHOICE, 3, "Foreground", 1,
                        "4 connect", "8 connect", NULL,
    CO_END_DEF

    CO_BEGIN_EXE
        int min_bg = CO_GET_INT(1);
        int min_fg = CO_GET_INT(2);
        int connex = CO_GET_CHOICE(3);

        thresh_area (p->pix[1], p->pix[2], p->xm, p->ym,
                     min_bg, min_fg, connex);
    CO_END_EXE

CO_END_CONNECT
```

FIG. 74 - Connexion sous IPS de l'opérateur visualisé figure 73

De multiples possibilités sont mises à la disposition des développeurs, comme de créer des boîtes de paramètres qui réagissent à certains choix, par exemple en grisant certains paramètres. De même, il est possible de préciser dynamiquement la taille des images. Par exemple, une transposition doit interchanger les dimensions, et une pyramide doit pouvoir diviser les tailles successivement par 2. Tout est fait pour que l'opérateur n'ait pas accès aux données privées d'IPS, mais passe par des macros en C, qui pourront évoluer de façon transparente. A ce propos, un programme de mise à jour permet, lorsqu'une nouvelle version d'IPS est disponible, de changer les librairies et de recompiler PIPS en accordance.

Il y a actuellement une douzaine de développeurs sous leur version propre de PIPS ; les bibliothèques vont de l'imagerie satellitaire aux *snakes*, en passant par le redressement géométrique, les approximations polygonales, les images sysmiques, les images RMN et les squelettes.

8.3 Logiciel GOLT

8.3.1 Contexte

Un séjour de 6 mois a été réalisé durant la thèse, dans le cadre d'un programme 'EuroDoc', financé par la région Rhône-Alpes. Le laboratoire d'accueil était l'Istituto di Cibernetica (CNR) de Naples, avec G. Sanniti di Baja (Directeur de Recherche).

Le sujet de recherche concernait le calcul de squelettes pondérés et leur utilisation en description de formes, qui font l'objet des chapitres 6 et 7 de ce document. L'environnement informatique du laboratoire était principalement composé de PC.

A cette occasion, nous avons écrit GOLT, un logiciel spécialisé dans les images de distances, qui facilite les nombreuses expérimentations et manipulations d'images demandées par le sujet.

Un certain nombre de contraintes ont guidé le développement de GOLT :

- il fallait que les opérateurs puissent fonctionner sous GOLT et IPS sans réécriture, en langage C ;
- le projet devait être bien adapté à la puissance modeste des PC ;
- la réalisation devait être rapidement achevée.

8.3.2 Interface

L'interface de GOLT se compose d'un écran principal (figure 75) et d'un écran d'édition. GOLT permet de travailler sur 9 images fixes (150×150). La barre du menu principal contient les noms des sous-menus, qui apparaissent à la droite des images. Le choix de la distance est réalisé *via* un menu spécialisé (en haut à droite).

Pour exécuter un opérateur, on commence par cliquer son nom dans le sous-menu correspondant. Cette action ouvre en bas de l'écran une boîte de paramètres spécifique à l'opérateur ; on a alors la possibilité de modifier ces valeurs. On clique enfin sur les images dans l'ordre adéquat ; dès que le nombre d'images est atteint, l'opérateur est exécuté.

Un éditeur est inclus dans GOLT, qui regroupe essentiellement les fonctionnalités offertes par IPS (dessins, zoom, loupe). L'écran d'édition est statique ; les traditionnels ascenseurs autour de l'image sont remplacés par une vignette dans un coin

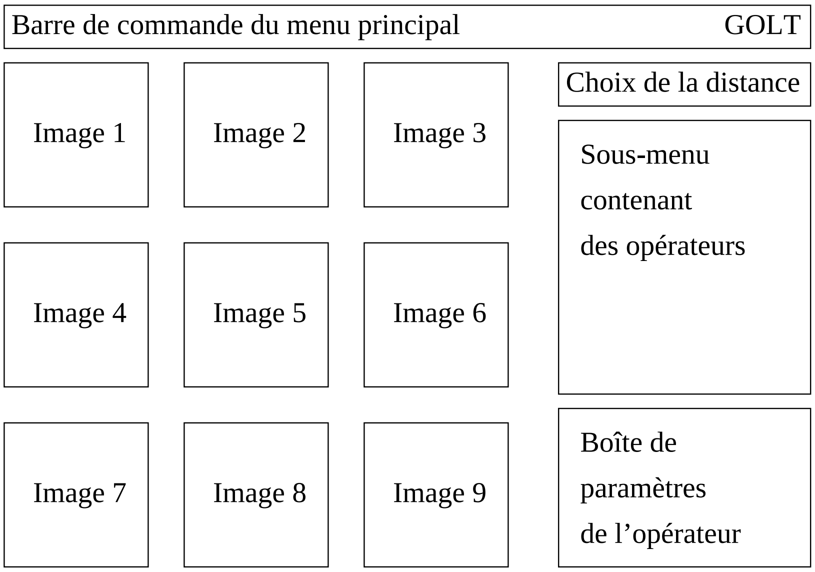


FIG. 75 - L'écran principal de GOLT

de l'écran, qui montre l'image globale et la portion de l'image éditée. On peut se déplacer dans l'image à partir de la zone d'édition comme de la vignette.

De l'aide est disponible à tout moment ; il suffit de cliquer avec le deuxième bouton de la souris sur la partie intéressée. Cette aide concerne le maniement de l'interface comme l'utilisation des opérateurs.

La philosophie de GOLT se distingue de celle d'IPS à de multiple égards. Les images sont statiques, en nombre restreint, et sont sélectionnées après l'opérateur. Une image peut être sélectionnée plusieurs fois pour un même opérateur. Les opérateurs ne sont pas organisés en icones ; on ne peut donc pas travailler avec plusieurs exemplaires d'un même opérateur dotés de paramètres différents, ni attacher des images à un opérateur.

Les ambitions de GOLT sont ainsi plus modestes, mais la prise en main est immédiate, les manipulations sont simples et rapides, et l'architecture interne de GOLT est d'une grande souplesse.

8.3.3 Développement

Nous avons perfectionné un gestionnaire de fenêtre existant, écrit sur PC en Pascal pendant le DEA.

Cette couche se situe immédiatement au dessus de DOS, et gère l'interruption souris, le clavier, les fenêtres et l'affichage. Les éléments graphiques implémentés sont les boutons, menus, cases à cocher, boutons radio, zones de saisie, zones graphique. Le gestionnaire est conçu de telle sorte que les fenêtres et leurs décorations puissent être entièrement décrites dans un fichier ressource ; la maintenance en est grandement facilitée. Le déplacement des fenêtres n'est pas géré, et le mécanisme de ventilation des événements est primitif. Notre gestionnaire est simpliste mais efficace, peu gourmand et bien adapté au projet de GOLT.

Un module écrit en assembleur réorganise la mémoire du PC à la demande, et permet d'appeler un programme fils, ce qui n'est pas prévu à l'origine dans le couple DOS / Turbo Pascal. Grâce à cela, les opérateurs d'analyse d'images ont pu être écrits en langage C, de même que des macro-commandes regroupant plusieurs opérateurs étaient écrites en *batch*[5].

Cette séparation très nette entre l'interface et les opérateurs est à l'origine de l'acronyme de GOLT, pour *Graphical On-Line Tool*.

Les images sont stockées dans des fichiers tampons ; GOLT communique simplement le nom des images selectionnées à l'opérateur, puis remet à jour l'affichage des images modifiées. La segmentation de la mémoire propre à DOS nous a contraint à limiter la taille des images à 150×150 (plus un bord). Mais cette taille est déjà intéressante, et bien adaptée à la puissance de la machine comme à la capacité de l'écran.

8.3.4 Appel des opérateurs

Les opérateurs sont exécutables depuis la ligne de commande du DOS. Le nom des images et les paramètres leur sont passés en argument.

Un fichier ressource décrit les sous-menus d'opérateurs et leurs caractéristiques : pour chaque opérateur est donné un petit script, qui déclare le nombre d'images en entrée et en sortie, le nombre de paramètres et la description de la boîte de paramètres, ainsi que la boîte d'aide. Lorsque l'opérateur est sélectionné, les paramètres fixés et les images cliquées, alors GOLT interprète le script et compose une ligne de commande, avec le nom de l'exécutable, le nom des images, les valeurs des paramètres, selon le format demandé.

[5] Le langage de commandes du DOS.

Cette organisation est très souple, et permet de rajouter rapidement des opérateurs, des macros, ou toute autre chose — par exemple, un programme de décompression qui charge des images — sans rien recompiler. Un autre avantage indéniable, est que durant l'exécution d'un programme fils, la sortie standard de l'affichage et des erreurs peut être redirigée vers un fichier trace, qui permet de faciliter la mise au point en cas de plantage ; de même, les images qui sont stockées dans des fichiers, sont intégralement conservées. Enfin, grâce à l'utilitaire Smartdrive, qui est un cache-disque, les chargements d'images et de programmes sont optimisés de fait.

Les opérateurs programmés sont relatifs au calcul de l'axe médian, de la ligne médiane, du squelette et de leur décomposition, dont les algorithmes ont été décrits tout au long de ce document :

- Divers : binarisation, inversion, rotation pythagoricienne, symétries, opérations booléennes, opérations locales.
- Transformations de distance : carte de distance, image inverse, axe médian, ligne médiane, boule de chanfrein, tests divers.
- Squelette : réduction de l'épaisseur, classification des points, élagage, qualité de l'approximation, polygonalisation, son inverse, tracé des gyxels, annihilation, analyse des nœuds, fusion, tracé des régions fusionnées.
- Macro-commandes : enchaînement des traitements, calcul du squelette en 4 distances, hiérarchie de la polygonalisation, tests intensifs, etc.

8.4 Applications en imagerie

Nous décrivons brièvement les applications développées au fil de cette thèse en imagerie, en les illustrant principalement par des figures. La décoalescence d'agrégats est utile en biologie, par exemple pour le décomptage de cellules. Les propriétés physiques du béton cellulaire autoclavé sont modélisées par analyse d'images. Enfin les fibres de papier sont étudiées à partir de la polygonalisation du squelette.

8.4.1 Décoalescence d'agrégats

Un des premiers problèmes qui a été résolu [Thi91a] avec ces outils consiste à détecter des rétrécissements dans une forme (figure 76) [Thi92e]. Nous disposons d'une ligne médiane qui contient l'information de distance au fond. Une zone de rétrécissement détectable, c'est-à-dire un col sur DM, se traduit sur LM par un minimum local (qui n'est pas une extrémité). Les chemins de séparation sont définis en suivant la ligne de plus grande pente de part et d'autre du col sur l'image des distances.

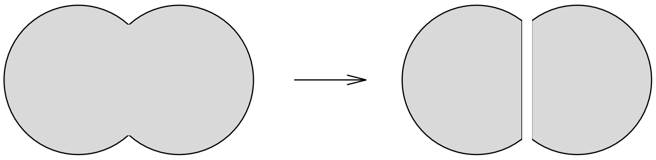


FIG. 76 - Séparation de deux agrégats

La détermination du col sur la ligne médiane peut être compliquée par l'épaisseur de la ligne médiane. Un progrès considérable est obtenu en la remplaçant par le squelette pondéré. Mais le col peut être aussi étendu sur une large zone ; l'utilisation de la polygonalisation du squelette (§7.2) résout efficacement ce problème.

Le choix de la distance est important, car un rétrécissement détectable pour une distance ne l'est pas forcément pour une autre. Par exemple dans la figure 77, le rétrécissement est détectable pour $d_{3,4}$ mais non pour d_8.

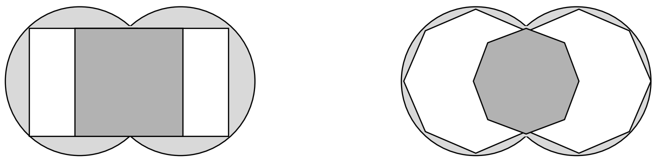


FIG. 77 - Rétrécissement détectable pour $d_{3,4}$ mais non pour d_8

On a tout intérêt à utiliser une distance quasi-euclidienne ; ainsi la figure 78 montre le plus grand rétrécissement détectable pour $d_{5,7,11}$ mais non pour $d_{3,4}$, qui soit appréciable.

8.4.2 Béton cellulaire autoclavé

Une application concernant la modélisation des propriétés dans le béton poreux fut développée pendant le DEA à l'Institut de Mécanique de Grenoble, avec J.P. Laurent et C. Frendo Rosso [Thi91a, Lau92]. Elle était basée sur la décoalescence à partir de la ligne médiane et $d_{3,4}$. L'utilisation de $d_{5,7,11}$ et de la polygonalisation du squelette devrait sensiblement améliorer la robustesse de la méthode.

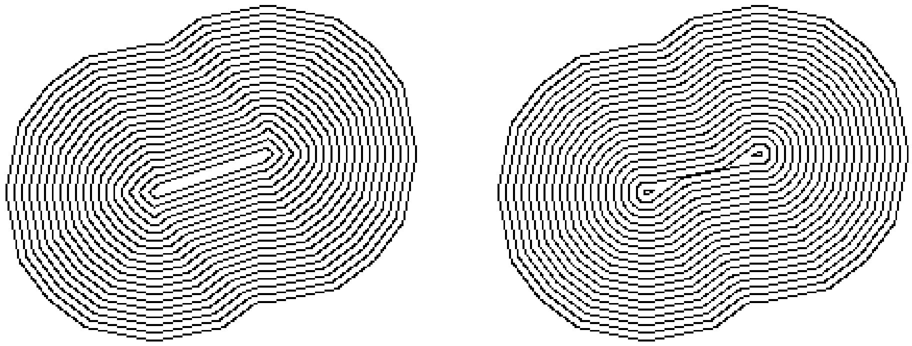


FIG. 78 - Plus grand rétrécissement détectable pour $d_{5,7,11}$ (à droite) mais non pour $d_{3,4}$ (à gauche)

Le béton cellulaire autoclavé est un béton très riche en pores (15% de matière, 85% de cavités), que l'on peut produire avec des contraintes quelconques. C'est un matériau très intéressant, dont on voudrait pouvoir modéliser certaines propriétés à partir de coupes. Il est conducteur au niveau de la matière, et isolant au niveau des cavités. Pour modéliser la conductivité thermique, l'idée est de considérer cette matière comme un réseau de résistances et de calculer par des méthodes relevant de l'électricité l'impédance équivalente. Etant principalement constitué de vide, le circuit n'est pas connecté en de nombreux endroits ; c'est pour cela qu'il est nécessaire de considérer la coupe de béton en 2D$\frac{1}{2}$, c'est-à-dire de rajouter les liens qui passent dans les couches supérieures ou inférieures entre les pores.

C'est là que l'analyse d'images entre en jeu : on détecte les rétrécissements entre pores, on rajoute de la matière au niveau des séparations, puis on calcule l'exosquelette, qui sert de base au circuit électrique.

La méthode est très performante, et fait économiser beaucoup de temps sur des méthodes d'érodé ultime. De plus elle permet d'accéder à un paramètre supplémentaire, qui est le taux de 'tortuosité'.

8.4.3 Fibres

Un logiciel de caractérisation de fibres dans les images de papier par utilisation des transformées en distances a été développé pour A. Tessadro, de l'Ecole Française de Papeterie de Grenoble [Tes94]. Il s'agissait de calculer le squelette et son approximation polygonale, qui sert de base pour étudier les fibres, au niveau de la courbure, de la longueur, et de générer par exemple un diagramme de leurs largeurs.

Le cahier des charges était difficile : il fallait contenir les algorithmes sur une carte Matrox, avec 2 plans mémoire 512×512 de 8 bits chacun. Les algorithmes sont complètement réécrits et optimisés en fonction de l'accès ligne à ligne de la carte. La phase d'élagage est simplifiée (il faudrait une troisième image pour stocker les étiquettes), et la polygonalisation est réalisée en 2D. Le premier plan image est dévolu à l'image de distance. Comme ce plan est de 8 bits, le choix de $d_{3,4}$ s'impose pour traiter toutes sortes de fibres. Le second plan sert d'image de travail.

FIG. 79 - Pâte mécanique d'épicéa

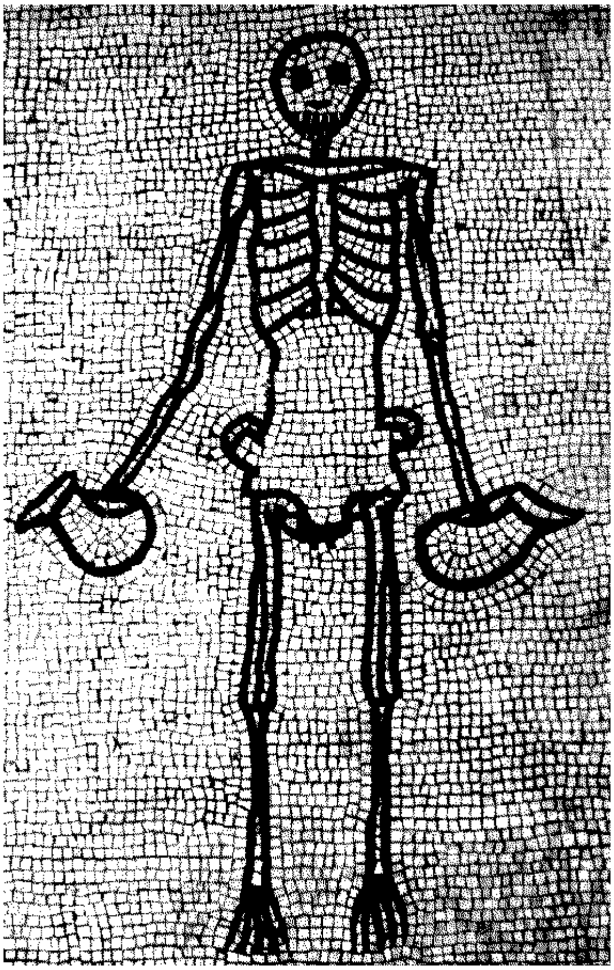
Les images de fibres sont très difficiles et de nature assez variée. A la figure 79 est représentée de la pâte mécanique d'épicéa. Le traitement complet sur une image 512×512 est de 45 secondes, contre 5 minutes pour Visilog, dont l'algorithme de squelettisation (binaire) plus frustre, ne polygonalise pas le squelette. Les algorithmes ont été mis au point dans IPS, avec un module simulant la carte Matrox.

8.5 Conclusion

Dans ce chapitre, nous avons décrit des logiciels de traitement d'images, réalisées en complément du travail théorique et du développement des outils, ainsi que des applications utilisant les squelettes.

Chapitre 9

Conclusion générale



Les squelettes discrets au temps des romains

Mosaïque, Musée Archéologique National de Naples.

La géométrie discrète est un domaine très vaste où il reste beaucoup à faire. Dans le contexte de l'analyse d'images, les images de distances ont leur rôle à jouer. L'étude réalisée dans ce document, a montré combien les distances de chanfrein sont une alternative remarquable à la distance euclidienne.

Nous avons développé une théorie des distances de chanfrein, nous permettant de définir les conditions exactes de distance et de norme, cette dernière étant liée à la régularité du masque. Nous avons pu établir de nouvelles formules de calcul direct, et montrer l'impérativité de bien avoir une distance pour la validité des algorithmes de calcul de l'image de distance.

Nous avons fait un état de l'art en ce qui concerne l'optimisation des masques, dans le but d'approximer au meilleur coût la distance euclidienne. Une méthode plus simple et plus efficace a été décrite, qui par le biais de l'exhaustivité, et en se basant sur l'erreur effective, assure de l'optimalité des masques. De plus nous avons établi les formules de changement de facteur d'échelle, qui permettent de comparer les différentes méthodes de la littérature.

Le chapitre concernant l'axe médian contient un nouvel algorithme remarquablement efficace de calcul de tables de correspondance pour caractériser les centres de boules maximales. Il nous a donné la possibilité de contredire le postulat selon lequel une boule recouverte par une autre, l'était obligatoirement par une boule centrée dans le voisinage de pondération. La condition de détectabilité est ouverte, et promet d'intéressantes recherches.

Un algorithme de squelettisation multi-distance a été proposé, et la présentation a été effectuée avec les quatre distances de chanfrein les plus courantes ; mais la méthode est adaptable aux masques plus grands, les deux seuls problèmes auxquels répondre étant le suivi de gradiant dans de tels voisinages, et la nécessité d'une formule de calcul direct pour la phase d'élagage. L'algorithme unique permet en particulier de comparer les mérites des différentes distances. Une étude complémentaire de la rotation du squelette serait à ce titre intéressante.

La méthode de description de formes, faisant appel à la polygonalisation du squelette, est très prometteuse. Il reste de nombreux points à compléter, au niveau des points de branchement et de la fusion de régions ; une technique d'approximation fondée par exemple sur les B-splines serait à étudier. Une autre perspective serait l'utilisation du squelette pour la mise en correspondance.

Les squelettes ont été construits dans plusieurs interfaces, élaborées en plus de la thèse ; l'application de ces méthodes à des problèmes concrets s'est révélée tout à fait satisfaisante.

Les progrès techniques au niveau des ordinateurs et des capteurs permettent de plus en plus de travailler en 3D. En particulier en robotique, il est indispensable de se placer dans cet espace. C'est pourquoi la prochaine grande étape de ce travail doit être la généralisation des distances de chanfrein au 3D, tant au niveau de la théorie qu'au niveau des algorithmes. Certains points sont immédiats, mais d'autres font apparaître des problèmes spécifiques à la troisième dimension. Nous espérons que ce travail trouvera son utilité pour aider à cet aboutissement.

Bibliographie

[And92] E. Andrès. *Cercles discrets et rotations discrètes*. PhD thesis, ULP, Strasbourg, 1992.

[Arc81] C. Arcelli. Pattern thinning by contour tracing. *Computer Graphics and Image Processing*, 17:130–144, 1981.

[Arc82] C. Arcelli and G. Sanniti di Baja. Shape splitting using maximal neighborhoods. In *6^{th} ICPR*, pages 1106–1108, Munich, 1982.

[Arc84] C. Arcelli and G. Sanniti di Baja. An approach to figure decomposition using width information. *Computer Vision, Graphics and Image Processing*, 26:61–72, 1984.

[Arc85] C. Arcelli and G. Sanniti di Baja. A width-independent fast thinning algorithm. *IEEE Trans. on PAMI*, 7(4):463–474, 1985.

[Arc86] C. Arcelli, M. Del Sordo, and G. Sanniti di Baja. Maximal neighborhoods in (3,4)-chamfer distance transforms. In *8^{th} Int. Symp. MECO*, pages 108–110, Taormina, Italy, 1986.

[Arc87] C. Arcelli and G. Sanniti di Baja. Weighted distance transforms : a characterization. In *4^{th} Conf. on IAPR*, pages 205–212, Cefalu (Italy), 1987.

[Arc88] C. Arcelli and G. Sanniti di Baja. Finding local maxima in a pseudo-euclidean distance transform. *Computer Vision, Graphics and Image Processing*, 43:361–367, 1988.

[Arc89] C. Arcelli and G. Sanniti di Baja. A one-pass two-operation process to detect the skeletal pixels on the 4-distance transform. *IEEE Trans. on PAMI*, 11(4):411–414, 1989.

[Arc90] C. Arcelli, R. Colucci, and G. Sanniti di Baja. On the description of digital strips. In *Int. Conf. on Artificial Intelligence Applications and Neural Networks*, pages 193–196, Zurich, 1990.

[Arc92] C. Arcelli and M. Frucci. Reversible skeletonization by (5-7-11)-erosion. In C. Arcelli et al., editors, *Visual Form Analysis and Recognition*, pages 21–28. Plenum, New York, 1992.

[Arc93] C. Arcelli and G. Sanniti di Baja. Euclidean skeleton via centre-of-maximal-disc extraction. *Image and Vision Computing*, 11(3):163–173, April 1993.

[Att93] D. Attali and A. Montanvert. Squelettes et diagrammes de Voronoï. Technical Report 922, IMAG, Grenoble, Juillet 1993.

[Att94a] D. Attali, P. Bertolino, and A. Montanvert. Using polyballs to approximate shapes and skeletons. In *12^{th} ICPR*, volume 1, pages 626–628, Jerusalem, Oct 1994.

[Att94b] D. Attali and A. Montanvert. Semicontinuous skeletons of 2D and 3D shapes. In C. Arcelli et al., editors, *Aspects of Visual Form Processing*, pages 32–41. World Scientific, Singapore, 1994.

[Bar77] H.G. Barrow, J.M. Tenenbaum, R.C. Bolles, and H.C. Wolf. Parametric correspondence and chamfer matching: two new techniques for image matching. In *proc. 5^{th} Int. Joint Conf. on Artif. Intell.*, pages 659–663, 1977.

[Bec89] A.L.D. Beckers and A.W.M. Smeulders. A comment on “A note on ‘Distance transformations in digital images’ ”. *Computer Vision, Graphics and Image Processing*, 47:89–91, 1989.

[Bec92] J.M. Becker. Approximations de la distance euclidienne sur trame : extension de la méthode Montanari-Borgefors. In *Distancia'92, congrès international sur l'analyse en distance*, pages 263–266, Rennes, Juin 1992.

[Ber78] M. Berger. *Convexes et polytopes, polyèdres réguliers, aires et volumes*, volume 3. Cedic / Fernand Nathan, $2^{ème}$ édition, 1978. §11.8.12.

[Ber94] E. Bertin. *Diagrammes de Voronoï 2D et 3D: Applications en analyse d'images.* PhD thesis, U. Joseph Fourier, Grenoble 1, 1994.

[Blu67] H. Blum. A transformation for extracting new descriptors of shape. In W. Wathendunn, editor, *Models for the Perception of Speech and Visual Form*, pages 362–380, Cambridge, 1967. MIT Press.

[Blu78] H. Blum and R.N. Nagel. Shape description using weighted symmetric axis features. *Pattern Recognition*, 10:167–180, 1978.

[Bol92] P. Bolon, J.L. Vila, and T. Auzepy. Opérateur local de distance en maillage rectangulaire. In *2^{nd} Discrete Geometry for Computer Image*, pages 45–56, Grenoble, Sept 1992.

[Bor84] G. Borgefors. Distance transformations in arbitrary dimensions. *Computer Vision, Graphics and Image Processing*, 27:321–345, 1984.

[Bor86a] G. Borgefors. Distance transformations in digital images. *Computer Vision, Graphics and Image Processing*, 34:344–371, 1986.

[Bor86b] G. Borgefors. *On hierarchical edge matching in digital images using distance transformations.* PhD thesis, Royal Institute of Technology, Stockholm, 1986.

[Bor88] G. Borgefors. Distance transformations in hexagonally digitized images. Technical Report C 30497-3.3, FOA, Linköping, Sweden, June 1988.

[Bor89] G. Borgefors. Time and memory efficient distance transformations for parallel and pyramid machines. Technical Report C 30531-3.4, FOA, Linköping, Sweden, May 1989.

[Bor90] G. Borgefors. Using resolution pyramids to efficiently store distance transforms of arbitrary size. In Cantoni et al., editors, *Progress in Image Analysis and Processing*, pages 691–697. World Scientific, 1990.

[Bor91a] G. Borgefors. Another comment on 'a note on 'distance transformation in digital images". *Computer Vision, Graphics and Image Processing*, 54:301–306, 1991.

[Bor91b] G. Borgefors, I. Ragnemalm, and G. Sanniti di Baja. The Euclidean Distance Transform : finding the local maxima and reconstructing the shape. In *7th Scandinavian Conf. on Image Analysis*, volume 2, pages 974–981, Aalborg, Denmark, 1991.

[Bor93] G. Borgefors. Centres of maximal disks in the 5-7-11 distance transform. In *8th Scandinavian Conf. on Image Analysis*, pages 105–111, Tromsø, Norway, 1993.

[Bra84] J.M. Brady and H. Asada. Smoothed local symmetries and their implementation. Memo 757, MIT Artificial Intelligence Laboratory, Feb 1984.

[Bro81] R.A. Brooks. Symbolic reasoning among 3D models and 2D images. *Artificial Intelligence*, 17:285–348, 1981.

[Cha79] J.M. Chassery. Connectivity and consecutivity in digital pictures. *Computer Graphics and Image Processing*, 9:294–300, 1979.

[Cha80] J.M. Chassery and M.I. Chenin. Topologies on discrete spaces. In Simon and Haralick, editors, *Digital Image Processing*, pages 59–66. Reidel, 1980.

[Cha91] J.M. Chassery and A. Montanvert. *Geométrie discrète en analyse d'images*. Hermès, 1991.

[Cha92] D. Chaudhuri, C.A. Murthy, and B.B. Chaudhuri. A modified metric to compute distance. *Pattern Recognition*, 25(7):667–677, 1992.

[Chi89] A. Chianese, L.P. Cordella, M. De Santo, and M. Vento. Decomposition of ribbon-like shapes. In *6th Scandinavian Conf. on Image Analysis*, pages 416–423, Oulu (Finland), 1989.

[Coq93] D. Coquin and P. Bolon. Comparaison d'opérateurs locaux de distance. In *3rd Discrete Geometry for Computer Image*, pages 182–191, Strasbourg, Sept 1993.

[Coq94] D. Coquin, Y. Chehadeh, and P. Bolon. Opérateur local de distance 3D, en maillage parallélépipédique. In *4th Discrete Geometry for Computer Image*, pages 147–156, Grenoble, Sept 1994.

[Cor84] L.P. Cordella and G. Sanniti di Baja. An approach to the decomposition of complex figures. *Digital Image Analysis*, pages 155–162, 1984.

[Cor88] P.P. Cortopassi and T.C. Rearick. A computationally efficient algorithm for shape decomposition. In 2^{nd} *Int. Conf. on Computer Vision*, pages 597–601, 1988.

[Cor89] L.P. Cordella and G. Sanniti di Baja. Geometric properties of the union of maximal neighborhoods. *IEEE Trans. on PAMI*, 11(2):214–217, 1989.

[Dan80] P.E. Danielsson. Euclidean distance mapping. *Computer Graphics and Image Processing*, 14:227–248, 1980.

[Dan93a] P.E. Danielsson. 3D octogonal metrics. In 8^{th} *Scandinavian Conf. on Image Analysis*, pages 727–736, 1993.

[Dan93b] P.E. Danielsson and S. Tanimoto. Time complexity for serial and parallel propagation in images. In *Architecture and Algorithms for Digital Image Processing*, volume 435, pages 60–67. SPIE, 1993.

[Dar94] P. Dargenton, N. Vincent, and H. Emptoz. Appariement de deux graphes structurels quelconques pour la reconnaissance de lettres manuscrites. In $9^{ème}$ *congrès Reconnaissance des Formes et I.A*, volume 1, pages 461–471, Paris, Janv 1994. AFCET.

[Das89] P.P. Das. More on path generated digital metrics. *Pattern Recognition Letters*, 10:25–31, 1989.

[Das90a] P.P. Das. Lattice of octagonal distances in digital geometry. *Pattern Recognition Letters*, 11:663–667, 1990.

[Das90b] P.P. Das and B.N. Chatterji. Octogonal distances for digital pictures. *Information Sciences*, 50:123–150, 1990.

[Das90c] P.P. Das and J. Mukherjee. Metricity of super-knight's distance in digital geometry. *Pattern Recognition Letters*, 11:601–604, 1990.

[Das91] P.P. Das. Counting minimal paths in digital geometry. *Pattern Recognition Letters*, 12:595–603, 1991.

[Dav80] E.R. Davies and A.P.N. Plummer. A new method for the compression of binary picture data. In *5^{th} ICPR*, pages 1150–1152, Miami, 1980.

[Dil87] A.R. Dill, M.D. Levine, and P.B. Noble. Multiple resolution skeletons. *IEEE Trans. on PAMI*, 9:495–504, 1987.

[Dor86a] L. Dorst. Pseudo-euclidean skeletons. In *8^{th} ICPR*, pages 286–288, Paris, 1986.

[Dor86b] L. Dorst and P.W. Verbeek. The constrained distance transformation: A pseudo-Euclidean recursive implementation of the Lee-algorithm. In Young et al., editors, *Signal Processing III: Theories and Applications*, pages 917–920. Elsevier Science, 1986.

[Eck88] U. Eckhardt and G. Maderlechner. The structure of irreducible digital sets obtained by thinning algorithms. In *9^{th} ICPR*, pages 727–729, Rome, 1988.

[Emb94] H. Embrechts. *MIMD divide-and-conquer algorithms for geometric operations on binary images*. PhD thesis, Katholieke Universiteit, Leuven, 1994.

[For87] S. Forchhammer. Skew and distance of digitized grid points in the plane. In *5^{th} Scandinavian Conf. on Image Analysis*, pages 451–458, Stockholm, 1987.

[For89] S. Forchhammer. Euclidean distances from chamfer distances for limited distances. In *6^{th} Scandinavian Conf. on Image Analysis*, pages 393–400, Oulu (Finland), 1989.

[Gon92] W. Gong. *Nouveaux algorithmes et squelettisation en 2 et 3 dimensions*. PhD thesis, CNAM, 1992.

[Gra93] C. Gratin. *De la représentation des images au traitement morphologique d'images tridimensionnelles*. PhD thesis, ENSM, Paris, 1993.

[Har78] G.H. Hardy and E.M. Wright. *An introduction to the theory of numbers*. Oxford University Press, fifth edition, October 1978. §3.1.

[Hil69] J. Hilditch and D. Rutovitz. Chromosome recognition. *Ann. New York Acad. Sci.*, 157:339–364, 1969.

[Huj87] M. Hujter and B. Vizvari. The exact solutions to the Frobenius problem with three variables. *Ramanujan Math. Soc.*, 2(2):117–143, 1987.

[Jen92] J.F. Jenq and S. Sahni. Serial and parallel algorithms for the Medial Axis Transform. *IEEE Trans. on PAMI*, 14:1218–1224, 1992.

[Kon89] T.Y. Kong and A. Rosenfeld. Digital topology: introduction and survey. *Computer Vision, Graphics and Image Processing*, 48:357–393, 1989.

[Kov89] V.A. Kovalesky. Finite topology as applied to image analysis. *Computer Vision, Graphics and Image Processing*, 46:141–161, 1989.

[Lam92] L. Lam, S.W. Lee, and C.Y. Suen. Thinning methodologies - A comprehensive survey. *IEEE Trans. on PAMI*, 14(9):869–885, Sept 1992.

[Lau92] JP. Laurent and C. Frendo Rosso. Application of image analysis to the estimation of AAC thermal conductivity. In *3^{rd} RILEM Int. Symp. on Autoclaved Aerated Concrete*, pages 14–16, Zürich, Switzerland, 1992.

[Mal94] R. Malgouyres. *Une caractérisation des surfaces de $\mathbb{Z}^3$*. PhD thesis, LLAIC1, Clermont, 1994.

[Mar87] M.P. Martinez-Perez, J. Jimenez, and J.L. Navalon. A thinning algorithm based on contours. *Computer Vision, Graphics and Image Processing*, 39:186–201, 1987.

[Mel84] R.A. Melter and I. Tomescu. Metric bases in digital geometry. *Computer Vision, Graphics and Image Processing*, 25:113–121, 1984.

[Mel91] R.A. Melter. A survey of digital metrics. *Contemporary Mathematics*, 119:95–106, 1991.

[Mel92] M. Melkemi. *Approches géométriques par modèles de Voronoï en segmentation d'images*. PhD thesis, UJF, Grenoble, 1992.

[Mey89] F. Meyer. Skeletons and perceptual graphs. *Signal Processing*, 16:335–363, 1989.

[Mig91] S. Miguet and Y. Robert. Parallélisation d'algorithmes de balayage d'image sur un anneau de processeurs. *Technique et Science Informatiques*, 10(4):287–295, 1991.

[Mon68] U. Montanari. A method for obtaining skeletons using a quasi-euclidean distance. *Journal of ACM*, 15:600–624, 1968.

[Mon86] A. Montanvert. Graph representation and shape description. In *8^{th} ICPR*, pages 430–432, Paris, 1986.

[Mon87a] A. Montanvert. *Contribution au traitement de formes discrètes : squelettes et codage par graphe de la ligne médiane.* PhD thesis, USTMG and INPG, Grenoble, 1987.

[Mon87b] A. Montanvert. Graph environment from medial axis for shape manipulation. In *4^{th} Conf. on IAPR*, pages 197–204, Cefalu (Italy), 1987.

[Mon90] A. Montanvert and D. Adelh. Tools for shape complexity evaluation and associated decomposition processes. In *Progress in Image Analysis and Processing*, pages 348–355, 1990.

[Mon93] A. Montanvert and Y. Usson. Discrete distances applied to 2D granulometry and 3D reconstruction. In *8^{th} Scandinavian Conf. on Image Analysis*, pages 1153–1160, 1993.

[Mul93] J.C. Mullikin. *Discrete and continuous methods for three-dimensional image analysis.* PhD thesis, TU, Delft, 1993.

[Nac93] P.F.M. Nacken. Efficient algorithms for mathematical morphology based on chamfer distances. In *W. on Math. Morpho. and its Appl. to Signal Proc.*, Barcelona, May 1993.

[Nac94] P.F.M. Nacken. *Image analysis methods based on hierarchies of graphs and multi-scale mathematical morphology.* PhD thesis, Amsterdam, 1994.

[Neh92] Ph. Nehlig. *Applications affines discrètes et antialiassage.* PhD thesis, ULP, Strasbourg, 1992.

[Ong92] S.H. Ong, Jayasooriah, H.H. Yeow, and R. Sinniah. Decomposition of digital clumps into convex parts by contour tracing and labelling. *Pattern Recognition Letters*, 13:789–795, 1992.

[Pag92a] D.W. Paglieroni. Distance transforms: properties and machine vision applications. *Computer Vision, Graphics and Image Processing*, 54(1):56–74, 1992.

[Pag92b] D.W. Paglieroni. A unified distance transform algorithm and architecture. *Machine Vision and Applications*, 5:47–55, 1992.

[Par93] F. Parazza, C. Humbert, and Y. Usson. Method for 3D volumetric analysis of intranuclear fluorescence distribution in confocal microscopy. *Computerized Medical Imaging and Graphics*, 17(3):189–200, 1993.

[Pav77] T. Pavlidis. *Structural pattern recognition.* Springer Verlag, New York, 1977.

[Pav92] T. Pavlidis. Applications of splines to shape description. In C. Arcelli et al., editors, *Visual Form Analysis and Recognition*, pages 431–441. Plenum, New York, 1992.

[Paw90] J.B. Pawley. *Handbook of biological confocal microscopy.* Plenum press, New York, 1990.

[Pet93] J.C. Pettier and J. Camillerapp. Segmentation et représentation d'images de traits. Technical Report 2066, INRIA, Oct 1993.

[Pfa67] J.L. Pfaltz and A. Rosenfeld. Computer representation of planar regions by their skeletons. *Comm. of ACM*, 10:119–125, feb 1967.

[Pon88] J. Ponce. Ribbons, symmetries and skewed symmetries. In *Image Understanding Workshop*, pages 1074–1079, Cambridge, Mass., 1988.

[Rag89] I. Ragnemalm. The euclidean distance transform and its implementation on SIMD architectures. In *6^{th} Scandinavian Conf. on Image Analysis*, pages 379–384, Oulu (Finland), 1989.

[Rag90a] I. Ragnemalm. Contour processing distance transforms. In Cantoni et al., editors, *Progress in Image Analysis and Processing*, pages 204–212. World Scientific, 1990.

[Rag90b] I. Ragnemalm. *Generation of Euclidean distance maps*. PhD thesis, Dep. Electrical Eng., Linköping University, Sweden, 1990.

[Rag92] I. Ragnemalm. The Euclidean distance transform in arbitrary dimensions. In *Int. Conf. on Image Processing*, pages 290–293, Maastricht, 1992.

[Rag93a] I. Ragnemalm. The constrained Euclidean distance transform. Chapter 3 of [Rag93b], pages 121–167, 1993.

[Rag93b] I. Ragnemalm. *The Euclidean Distance Transform*. PhD dissertation, Dep. Electrical Eng., Linköping University, Sweden, 1993.

[Rev91] J.P. Reveillès. *Géométrie Discrète, Calcul en Nombres Entiers et Algorithmique*. PhD thesis, ULP, Strasbourg, 1991.

[Rho90] F. Rhodes. Some characterization of the chessboard metric and the city block metric. *Pattern Recognition Letters*, 11:669–675, 1990.

[Rho92] F. Rhodes. Discrete euclidean metrics. *Pattern Recognition Letters*, 13:623–628, 1992.

[Rol91] F. Rolland. *Représentation tridimensionnelle et reconstruction 3D à partir de coupes 2D*. PhD thesis, UJF, Grenoble, 1991.

[Rom93] H. Rom and G. Medioni. Hierarchical decomposition and axial shape description. *IEEE Trans. on PAMI*, 15(10):973–981, 1993.

[Ros66] A. Rosenfeld and J.L. Pfaltz. Sequential operations in digital picture processing. *Journal of ACM*, 13(4):471–494, 1966.

[Ros68] A. Rosenfeld and J.L. Pfaltz. Distance functions on digital pictures. *Pattern Recognition*, 1:33–61, 1968.

[Ros73] A. Rosenfeld. Arcs and curves in digital pictures. *Journal of ACM*, 20:81–87, 1973.

[Ros82] A. Rosenfeld and A.C. Kak. *Digital image processing.* Academic Press, New-York, 1982.

[Ros86] A. Rosenfeld. Axial representation of shape. *Computer Vision, Graphics and Image Processing*, 33:156–173, 1986.

[Rut66] D. Rutovitz. Pattern recognition. *Journal of Royal Statis.*, 129:504–530, 1966.

[Sam90] H. Samet. *The design and analysis of spatial data structures.* Addison Wesley, 1990.

[San90] G. Sanniti di Baja. O(N) computation of projections and moments from the labeled skeleton. *Computer Vision, Graphics and Image Processing*, 49:369–378, 1990.

[San94] G. Sanniti di Baja. Well-shaped, stable and reversible skeletons from the (3,4)-distance transform. *Journal of Visual Communication and Image Representation*, 5:107–115, 1994.

[Sch94] M. Schmitt and J. Mattioli. *Morphologie mathématique.* Masson, Collection Logique Mathématiques Informatique, 1994.

[Ser93] R. Seroul. Le problème des timbres poste ou le problème de Frobenius. Personnal communication, 1993.

[Shi90] F.Y. Shih and C.C. Pu. Medial axis transformation with single-pixel and connectivity preservation using euclidean distance computation. In *10^{th} ICPR*, pages 723–725, Atlantic City, 1990.

[Sir91] A. Sirjani and G.R. Cross. On representation of a shape's skeleton. *Pattern Recognition Letters*, 12:149–154, 1991.

[Ste85] S.R. Sternberg. An overview of image algebra and related architectures. In S. Levialdi, editor, *Integrated Technology for Parallel Image Processing*, pages 79–100, London, 1985. Academic Press.

[Ste90] F. Stein and G. Medioni. Efficient two dimensional object recognition. In *10^{th} ICPR*, pages 13–17, Atlantic City, 1990.

[Syl84] J. Sylvester. Mathematicals questions with their solutions. *Educational Times*, 41:21, 1884.

[Tes94] A. Tessadro. *Caractérisation morphologique des fibres par analyse d'images*. PhD thesis, Ecole Française de Papeterie, Grenoble, 1994.

[Thi91a] E. Thiel. *Amélioration des distances du chanfrein ; construction de la ligne médiane et décomposition de formes*. DEA en Mathématiques Appliquées, UJF, Grenoble 1, 1991.

[Thi91b] E. Thiel and A. Montanvert. Approximation discrète de la distance euclidienne pour l'analyse d'images : étude et amélioration des distances du chanfrein. In *8ème congrès Reconnaissance des Formes et I.A*, volume 2, pages 829–839, Lyon, Nov 1991. AFCET.

[Thi91c] E. Thiel and A. Montanvert. Approximations discrètes de la distance euclidienne. In *1st Discrete Geometry for Computer Image*, Strasbourg, Sept 1991.

[Thi92a] E. Thiel. Distances discrètes. In *2nd Discrete Geometry for Computer Image*, pages 69–78, Grenoble, Sept 1992.

[Thi92b] E. Thiel and A. Montanvert. Chamfer masks : discrete distance functions, geometrical properties and optimization. In *11th ICPR*, volume 3, pages 244–247, The Hague, The Netherlands, Sept 1992.

[Thi92c] E. Thiel and A. Montanvert. Etude et amélioration des distances du chanfrein pour l'analyse d'images. *Technique et Science Informatiques*, 11(4):9–41, 1992.

[Thi92d] E. Thiel and A. Montanvert. Les distances du chanfrein en géométrie discrète. In *Distancia'92, congrès international sur l'analyse en distance*, pages 295–298, Rennes, Juin 1992.

[Thi92e] E. Thiel and A. Montanvert. Shape splitting from medial lines using the 3-4 chamfer distance. In C. Arcelli et al., editors, *Visual Form Analysis and Recognition*, pages 537–546. Plenum, New York, 1992.

[Thi93a] D. Attali and E. Thiel. Du squelette discret ou continu. In *3^{rd} Discrete Geometry for Computer Image*, pages 236–244, Strasbourg, Sept 1993.

[Thi93b] G. Sanniti di Baja and E. Thiel. A multiresolution shape description algorithm. In D. Chetverikov et al., editors, *Lectures Notes in Computer Science*, volume 719, pages 208–215. Springer Verlag, 1993.

[Thi93c] G. Sanniti di Baja and E. Thiel. Shape description via weighted skeleton partition. In *7^{th} ICIAP*, pages 87–94, 1993.

[Thi94a] G. Sanniti di Baja and E. Thiel. (3,4)-weighted skeleton decomposition for pattern representation and description. *Pattern Recognition*, 27:1039–1049, 1994.

[Thi94b] G. Sanniti di Baja and E. Thiel. Computing and comparing distance-driven skeletons. In C. Arcelli et al., editors, *Aspects of Visual Form Processing*, pages 465–486. World Scientific, Singapore, 1994.

[Thi94c] G. Sanniti di Baja and E. Thiel. The path based distance skeleton : a flexible tool to analyse silhouette shape. In *12^{th} ICPR*, volume 2, pages 570–572, Jerusalem, Oct 1994.

[Thi94d] E. Thiel. Unification de la squelettisation menée en distance. In *$9^{ème}$ congrès Reconnaissance des Formes et I.A*, volume 1, pages 349–358, Paris, Janv 1994. AFCET.

[Tro87] Y. Trousset. *Visualisation d'objets médicaux tridimensionnels*. PhD thesis, ENST, Paris, 1987.

[Ube93] S. Ubeda. *Algorithmes d'amincissement d'images sur machines parallèles*. PhD thesis, ENS, Lyon, 1993.

[Uss94] Y. Usson, F. Parazza, P.S. Jouk, and G. Michalowicz. Method for the study of the three-dimensional orientation of the nuclei of myocardial cells in fetal human heart by means of confocal scanning laser microscopy. *Journal of Microscopy*, 174:101–110, 1994.

[Ver89] J.H. Verwer, P.W. Verbeek, and S.T. Dekker. An efficient uniform cost algorithm applied to distance transforms. *IEEE Trans. on PAMI*, 11:425–429, 1989.

[Ver91] J.H. Verwer. Local distance for distance transformations in two and three dimensions. *Pattern Recognition Letters*, 12:671–682, 1991.

[Vin91] L. Vincent. Exact Euclidean distance function by chain propagations. In *Computer Vision and Pattern Recognition*, pages 520–525, Hawaï, June 1991.

[Vos88] A.M. Vossepoel. A note on "Distance transformations in digital images". *Computer Vision, Graphics and Image Processing*, 43:88–97, 1988.

[Wu86] A.Y. Wu, S.K. Bhaskar, and A. Rosenfeld. Computation of geometric properties from the Medial Axis Transform in $O(n \log n)$ time. *Computer Vision, Graphics and Image Processing*, 34:76–92, 1986.

[Xia89] Y. Xia. Skeletonization via the realization of the fire front propagation and extinction in digital binary shapes. *IEEE Trans. on PAMI*, 11:1076–1086, 1989.

[Yam84a] H. Yamada. Complete euclidean distance transformation by parallel operation. In *7^{th} ICPR*, pages 69–71, Montreal, 1984.

[Yam84b] M. Yamashita and N. Honda. Distance functions defined by variable neighborhood sequences. *Pattern Recognition*, 17(5):509–513, 1984.

[Ye88] Qin-Zhong Ye. The signed Euclidean distance transform and its applications. In *9^{th} ICPR*, pages 495–499, Rome, 1988.

[Yok75] S. Yokoi, J. Toriwaki, and T. Fukumura. An analysis of topological properties of digitized binary pictures using local features. *Computer Graphics and Image Processing*, 4:63–73, 1975.