

# Introduction à la Morphologie Mathématique, Théorie et applications

Édité par Hugues TALBOT

version 2.0

20 octobre 2010



## Table des matières

<b>Table des matières</b> . . . . .	9
<b>Chapitre 1. Introduction à la morphologie mathématique</b> . . . . .	13
Laurent NAJMAN et Hugues TALBOT	
1.1. Premiers pas en morphologie : la dilatation et l'érosion . . . . .	14
1.1.1. La notion de treillis complet . . . . .	14
1.1.1.1. Exemples de treillis . . . . .	14
1.1.1.2. Treillis et ordre . . . . .	15
1.1.2. Opérateurs élémentaires . . . . .	15
1.1.2.1. Eléments structurants . . . . .	16
1.1.2.2. Eléments structurants plans . . . . .	18
1.1.3. Transformations en tout ou rien . . . . .	18
1.2. Le filtrage morphologique . . . . .	19
1.2.1. Ouvertures et fermetures par éléments structurants . . . . .	19
1.2.2. Géodésie et reconstruction . . . . .	20
1.2.2.1. Ouverture et fermeture par reconstruction . . . . .	20
1.2.2.2. Structure de l'espace, voisinage . . . . .	21
1.2.2.3. Chemins et connexité . . . . .	23
1.2.3. Filtrage connexe et nivellements . . . . .	24
1.2.4. Ouverture/fermeture par aire . . . . .	24
1.2.5. Filtres algébriques . . . . .	25
1.2.6. Famille granulométrique . . . . .	27
1.2.7. Filtres alternés séquentiels . . . . .	27
1.3. Résidus . . . . .	28
1.3.1. Gradients . . . . .	29
1.3.2. Les transformations chapeaux haut-de-forme . . . . .	29
1.4. Fonction distance, squelette et courbe granulométrique . . . . .	29
1.4.1. Boules maximales et squelette . . . . .	30
1.4.2. Courbe granulométrique . . . . .	32
1.4.2.1. Applications . . . . .	33
1.4.2.2. Granulométries par érosions et dilatation . . . . .	33
1.5. Ligne de partage des eaux . . . . .	33
1.6. En guise de conclusion . . . . .	35

<b>Chapitre 2. Introduction à la mesure en analyse d'image</b> . . . . .	37
Hugues TALBOT, Jean SERRA et Laurent NAJMAN	
2.1. Introduction . . . . .	37
2.2. Principes généraux . . . . .	38
2.3. Anneau convexe et fonctionnelles de Minkowski . . . . .	39
2.3.1. La caractéristique d'Euler -Poincaré. . . . .	41
2.3.2. Interprétation de la caractéristique d'Euler-Poincaré . . . . .	42
2.3.3. La caractéristique d'Euler-Poincaré dans l'espace discret . . . . .	42
2.4. Stéréologie et fonctionnelles de Minkowski . . . . .	44
2.4.1. Fonctionnelles de Minkowski . . . . .	44
2.4.1.1. En dimension 1 . . . . .	46
2.4.1.2. En dimension 2 . . . . .	46
2.4.1.3. En dimension 3 . . . . .	46
2.5. Changement d'échelle et stationarité . . . . .	47
2.6. Individus et granulométries . . . . .	47
2.6.1. Comptages sans biais . . . . .	48
2.6.2. Granulométries en nombre et en mesure . . . . .	49
2.6.2.1. Correction de Miles-Lantuejoul . . . . .	49
2.6.3. Granulométries linéaires . . . . .	50
2.7. Extension aux niveaux de gris . . . . .	52
2.7.1. Aires et volumes . . . . .	53
2.7.2. Gradient et périmètre . . . . .	53
2.7.3. Caractéristique d'Euler-Poincaré . . . . .	53
2.7.4. Un contre-exemple : la longueur d'une courbe . . . . .	53
2.8. Conclusion . . . . .	54
<b>Chapitre 3. Distances, granulométries, squelette</b> . . . . .	55
Michel COUPRIE et Hugues TALBOT	
3.1. Squelettes . . . . .	55
3.1.1. Boules maximales . . . . .	55
3.1.2. Fronts de feu . . . . .	56
3.1.3. Propriétés du squelette dans le continu . . . . .	57
3.2. Squelettes dans les espaces discrets . . . . .	58
3.3. Familles granulométriques et squelettes . . . . .	59
3.3.1. Famille granulométrique . . . . .	59
3.3.2. Application des granulométries . . . . .	60
3.3.3. Formule des érodés ultimes . . . . .	60
3.3.4. Formule de Lantuéjoul . . . . .	61
3.4. Distances discrètes . . . . .	63
3.5. Fonction bissectrice . . . . .	67
3.6. Transformations homotopiques . . . . .	68
3.6.1. Voisinages, connexité . . . . .	70
3.6.2. Nombres de connexité et points simples . . . . .	71
3.6.3. Amincissement homotopique . . . . .	72
3.6.4. Algorithmes séquentiels et parallèles d'amincissement . . . . .	72
3.6.5. Squelette basé sur la distance euclidienne . . . . .	73
3.7. Conclusion . . . . .	74

<b>Chapitre 4. Morphologie et algorithmes</b> . . . . .	77
Thierry GÉRAUD, Hugues TALBOT et Marc VAN DROOGENBROECK	
4.1. Introduction . . . . .	77
4.2. Traduction définition / algorithme . . . . .	78
4.2.1. Structures de données . . . . .	78
4.2.2. Forme et étendue du domaine . . . . .	79
4.2.3. Structure d'ensembles de points . . . . .	80
4.2.4. Raccourcis d'écriture . . . . .	80
4.2.5. De la définition à la réalisation . . . . .	80
4.3. Taxonomie des algorithmes . . . . .	81
4.3.1. Critères de taxonomie . . . . .	82
4.3.2. Compromis . . . . .	84
4.3.3. Classes d'algorithmes et canevas . . . . .	85
4.4. Exemple de la reconstruction géodésique . . . . .	86
4.4.1. La version mathématique : algorithme parallèle . . . . .	86
4.4.1.1. Algorithmes similaires . . . . .	87
4.4.1.2. Algorithme séquentiel . . . . .	88
4.4.1.3. Algorithmes similaires . . . . .	88
4.4.1.4. Algorithme à base de file d'attente . . . . .	88
4.4.1.5. Remarques . . . . .	90
4.4.1.6. Algorithme hybride . . . . .	91
4.4.1.7. Remarques . . . . .	91
4.4.1.8. Algorithme par Union-Find . . . . .	91
4.4.1.9. Idée générale de l'algorithme . . . . .	91
4.4.1.10. Détails . . . . .	92
4.4.1.11. Comparaison avec les méthodes précédentes . . . . .	93
4.4.2. Comparaison de ces algorithmes . . . . .	93
4.5. Perspectives historiques et notes bibliographiques . . . . .	94
4.5.1. Pré- et péri-morphologie . . . . .	94
4.5.1.1. Algorithmes à base de graphes . . . . .	94
4.5.1.2. Algorithmes de géométrie et de topologie discrète . . . . .	94
4.5.1.3. Algorithmes inspirés du continu . . . . .	95
4.5.1.4. Optimisation discrète et continue pour la segmentation . . . . .	95
4.5.1.5. Algorithmes d'analyse linéaire . . . . .	96
4.5.2. Historique des développements algorithmiques en morphologie mathématique . . . . .	96
4.5.2.1. Algorithmes parallèles . . . . .	96
4.5.2.2. Algorithmes séquentiels . . . . .	97
4.5.2.3. Algorithmes en largeur d'abord . . . . .	97
4.5.2.4. Algorithmes inspirés des graphes . . . . .	97
4.5.2.5. Algorithmes topologiques . . . . .	97
4.5.2.6. Filtrage . . . . .	98
4.6. Conclusion . . . . .	99
<b>Table des figures</b> . . . . .	101
<b>Bibliographie</b> . . . . .	103



## Chapitre 1

# Introduction à la morphologie mathématique

Dans ce chapitre, nous allons tenter d'introduire rapidement ce qu'est la morphologie mathématique, ou ce qui rentre dans son champ d'étude. Cette question est difficile, non pas tant au niveau technique qu'au niveau du point de départ. Historiquement, la morphologie a commencé par l'étude des ensembles aléatoires pour des applications à l'industrie minière. Elle a rapidement été appliquée au traitement et à l'analyse des images dans un cadre déterministe, d'abord dans un cadre d'images binaires, puis en niveaux de gris, puis encore plus tard sur des images en couleur, multispectrales, et en dimensions supérieures à 2. Entrent dans le cadre morphologique de nombreuses disciplines mathématiques, par exemple des considérations de théorie des ensembles généralisée au cadre des treillis, des ensembles aléatoires et de la théorie de la mesure, de topologie, de géométrie discrète et continue ; ainsi que des considérations algorithmiques, et des applications.

D'un point de vue très large, le principe de l'analyse morphologique est d'extraire de la connaissance à partir des réponses fournies par différentes transformations en général non-linéaires.

La pensée morphologique [MAT 75, SER 82, SER 88b] – voir également [HEI 94, SCH 94, SOI 03] – sort du cadre habituellement enseigné aujourd'hui à l'école et même à l'université. Dans les mathématiques « classiques », une fonction associe à un point de l'espace un autre point. *A contrario*, en morphologie, une fonction associe un ensemble à un autre ensemble. Les conséquences sont importantes. Par exemple, si un point est de mesure nulle, ce n'est pas généralement le cas pour un ensemble. Ce simple fait permet de poser des questions qui n'auraient pas de sens autrement : par exemple la probabilité de présence d'un point est toujours nulle, ce n'est pas généralement le cas de celle d'un ensemble.

D'autre part, on peut comparer la morphologie aux autres disciplines du traitement des images. Par exemple, en théorie linéaire, on suppose qu'une image n'est qu'un signal qui a plus d'une dimension. On suppose aussi que les signaux se superposent et s'additionnent. La structure de base est l'espace vectoriel, et les opérateurs sont ceux qui préservent la structure et commutent avec les lois de bases, c'est-à-dire l'addition et la multiplication par une constante. On en dérive naturellement les opérateurs de convolution. Dans ce cadre, il est donc naturel d'étudier les transformations de Fourier ou en ondelettes, ou encore de regarder les décompositions par projection sur des vecteurs de bases. Cette façon de procéder est extrêmement utile, mais elle est incomplète.

---

Chapitre rédigé par Laurent NAJMAN et Hugues TALBOT.

En effet, très souvent une image par exemple en 2D n'est pas seulement un signal, mais une projections d'une « réalité » plus large en 3D. Deux objets qui se superposent du fait de la projection ne s'additionnent pas, mais s'occulent. L'addition n'est alors pas l'opérateur le plus naturel ; on raisonnera plutôt en termes d'ensembles, et donc en termes d'union, d'intersection, etc. En morphologie, on caractérise ce qu'on observe par des transformées géométriques en tirant parti de certaines formes, de la connexité, de l'orientation, de la taille, etc. La structure de base la plus adaptée n'est donc pas celle des espaces vectoriels, mais celle de la généralisation de la théorie des ensembles sous la forme des treillis complets [BIR 95].

**1.1. Premiers pas en morphologie : la dilatation et l'érosion**

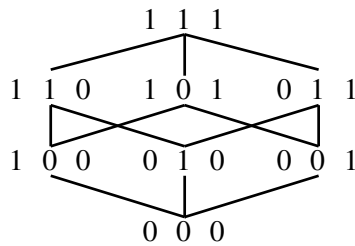
Pour pouvoir définir les opérateurs de la morphologie mathématique, on passe par la notion abstraite de treillis complet. On sera ensuite capable de « faire » de la morphologie sur tout treillis.

**1.1.1. La notion de treillis complet**

Un treillis [BIR 95]  $(E, \leq)$  est la donnée d'un ensemble  $E$  (l'espace) muni d'une relation d'ordre  $\leq$  réflexive ( $x \leq x$ ), antisymétrique ( $x \leq y$  et  $y \leq x \Rightarrow x = y$ ) et transitive ( $x \leq y$  et  $y \leq z \Rightarrow x \leq z$ ), telle que pour deux éléments  $x$  et  $y$  on puisse définir un plus grand élément  $x \vee y$  et un plus petit élément  $x \wedge y$ . Un treillis est dit *complet* si toute partie  $P$  de  $E$  admet une borne supérieure  $\vee P$  (*supremum*, le plus petit des majorants de  $P$ ) et une borne inférieure  $\wedge P$  (*infimum*, le plus grand des minorants de  $P$ ). Dans un treillis, le *supremum* et l'*infimum* jouent des rôles symétriques ; en particulier, si l'on se place dans le treillis  $\mathcal{P}(E)$  des parties de l'ensemble  $E$ , deux opérateurs  $\psi$  et  $\psi^*$  sont duaux si pour tout  $X$ ,  $\psi(X^c) = [\psi^*(X)]^c$ , où  $X^c = E \setminus X$  est le complément de  $X$  dans  $E$ .

1.1.1.1. Exemples de treillis

Un exemple de treillis est donné en figure 1.1. Il s'agit d'un treillis simple mais informatif : le treillis des couleurs primaires additives (rouge, vert et bleu). 0 représente la couleur absente et 1 la couleur présente. La couleur noire est  $[0, 0, 0]$ , le blanc  $[1, 1, 1]$ , le rouge pur  $[1, 0, 0]$ , le vert pur  $[0, 1, 0]$ , etc. Le violet (magenta) est  $[1, 0, 1]$ . Dans ce treillis, on ne peut pas comparer rouge et vert, ou encore violet et jaune, mais le blanc est supérieur (plus brillant) à toutes les autres, et le noir inférieur (plus sombre aux autres). Quel que soit l'ensemble des couleurs choisies, par exemple jaune et bleu, on peut définir un *supremum* et un *infimum*, bien que l'ordre des couleurs ne soit pas total.



**Figure 1.1.** Un exemple de treillis : le treillis des couleurs primaires additives

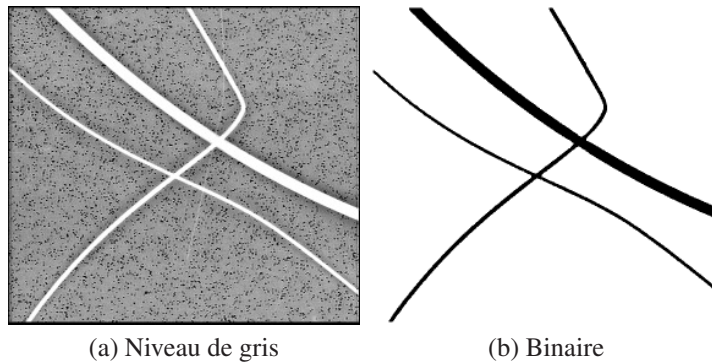
Un autre treillis est celui des nombres réels : tout intervalle dans  $\mathbb{R}$  possède un *supremum* et un *infimum*, au sens usuel. La notion de treillis n'est pas fondamentalement complexe.



### 1.1.1.2. Treillis et ordre

De nombreux opérateurs morphologiques préservent la structure d'ordre, c'est-à-dire que ce sont des opérateurs croissants ( $x \leq y \Rightarrow \Phi(x) \leq \Phi(y)$ ). Les opérateurs les plus simples que l'on peut introduire sur un treillis sont ceux qui commutent avec le *supremum* ou avec l'*infimum*. Plus précisément, une dilatation est un opérateur croissant anti-extensif ( $\gamma(X) \leq X$ ) qui commute avec le *supremum*  $\vee$ , une érosion est un opérateur croissant extensif ( $\gamma(X) \geq X$ ) qui commute avec l'*infimum*  $\wedge$ . Ces deux opérateurs sont duaux l'un par rapport à l'autre.

On peut se demander pourquoi, dans cette introduction, présenter un cadre aussi abstrait que celui des treillis. Pour répondre à cette interrogation légitime, il est peut-être temps de préciser ce qu'est une image. Observons celle de la figure 1.2.a, qui est en niveaux de gris. Cette image a un contenu technique intéressant, il s'agit de fibres de verre observées en microscopie électronique, mais ce contenu n'a pas d'importance ici. La première constatation que nous pouvons faire est que cette image est une fonction numérique  $F : E \rightarrow T$ , où  $E$  est l'espace des points et  $T$  est l'ensemble des valeurs de  $F$ , c'est-à-dire dans notre cas l'ensemble des niveaux de gris. L'espace  $E$  des points peut être vu comme continu (par exemple  $E = \mathbb{R}^n$ ) ou discret (par exemple  $E = \mathbb{Z}^n$ ). L'espace  $T$  est un sous-ensemble de  $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, +\infty\}$ . Nous noterons par  $T^E$  l'ensemble des fonctions de  $E$  dans  $T$ . Un des premiers intérêts des treillis est de permettre de définir les opérateurs agissant sur les images sans spécifier davantage l'espace de définition de ces images. Une présentation beaucoup plus complète des treillis et de l'algèbre morphologique est donnée dans le chapitre ??.



**Figure 1.2.** Une image en niveaux de gris (a) et une image binaire (b), seuil de (a)

### 1.1.2. Opérateurs élémentaires

On peut définir les opérateurs de base de la morphologie de plusieurs façons. Il est utile de considérer le cas le plus simple, celui des images à deux niveaux : strictement blanc et strictement noir. Ce cadre est loin d'être le seul sur lequel la morphologie s'exprime, mais il a plusieurs avantages : il est à la fois simple et relativement intuitif, mais également suffisamment flexible pour qu'on puisse généraliser les opérations qu'on y définit à des cadres plus complexes.

Une des opérations les plus simples que l'on puisse réaliser sur une image en niveaux de gris  $F$  est de la *seuiller*. Le seuil de  $F$  pour l'altitude  $t$  est l'ensemble  $X_t(F)$  défini par :

$$X_t(F) = \{p \in E, F(p) \leq t\} \quad (1.1)$$

Un seuil de l'image 1.2a est l'image 1.2b. L'image 1.2a est dite *en niveaux de gris*, et l'image 1.2b est dite *binnaire*. On peut considérer une image binaire soit comme un ensemble, soit comme une fonction numérique prenant ses valeurs dans  $\{0, 1\}$ . Encore une fois, le cadre des treillis permet de se poser la question le plus tard possible.

Si nous considérons les images binaires comme des sous-ensembles d'un espace  $E$ , la structure adaptée est le treillis  $\mathcal{P}(E)$  des parties de  $E$ . C'est d'ailleurs ce treillis ensembliste qui est le plus utilisé dans la pratique. Dans ce cas, la structure d'ordre  $\leq$  est donné par l'inclusion  $\subseteq$ , le *supremum*  $\vee$  par l'union  $\cup$ , et l'*infimum*  $\wedge$  par l'intersection  $\cap$ . Il est à noter que ce treillis n'est pas le seul possible, si par exemple on spécialise le type d'ensembles sur lequel on travaille. Si on souhaite traiter uniquement le cas des ensembles convexes, alors il est adapté de choisir le treillis des ensembles convexes, munis de l'inclusion habituelle pour l'*infimum*, et de l'enveloppe convexe de l'union pour le *supremum*.

### 1.1.2.1. Eléments structurants

Dans la pratique de la morphologie mathématique, on étudie souvent les images (binaires ou niveaux de gris) à l'aide d'un ensemble  $B$  que l'on connaît *a priori*, et que l'on adapte à nos besoins. Cet ensemble  $B$  est appelé un élément structurant. Soit  $X$  une image binaire, c'est-à-dire un sous-ensemble de  $E$ . Le *translaté* de  $X$  par  $p \in E$  est l'ensemble  $X_p = \{x + p, x \in X\}$ . La dilatation de  $X$  par  $B$  est définie par :

$$\delta_B(X) = X \oplus B = \bigcup_{b \in B} X_b \quad (1.2)$$

$$= \bigcup_{x \in X} B_x \quad (1.3)$$

$$= \{x + b, x \in X, b \in B\} \quad (1.4)$$

La dilatation est le lieu géométrique des points  $z$  tels que  $B_z$  intersecte  $X$  :  $\delta_B(X) = \{p, B_p \cap X \neq \emptyset\}$ . Par conséquent, la dilatation de  $X$  par  $B$  « grossit »  $X$ . Les ensembles  $X$  et  $B$  jouent un rôle symétrique dans la formule.

L'érosion de  $X$  par  $B$  est définie par :

$$\varepsilon_B(X) = X \ominus B = \bigcap_{b \in B} X_{-b} \quad (1.5)$$

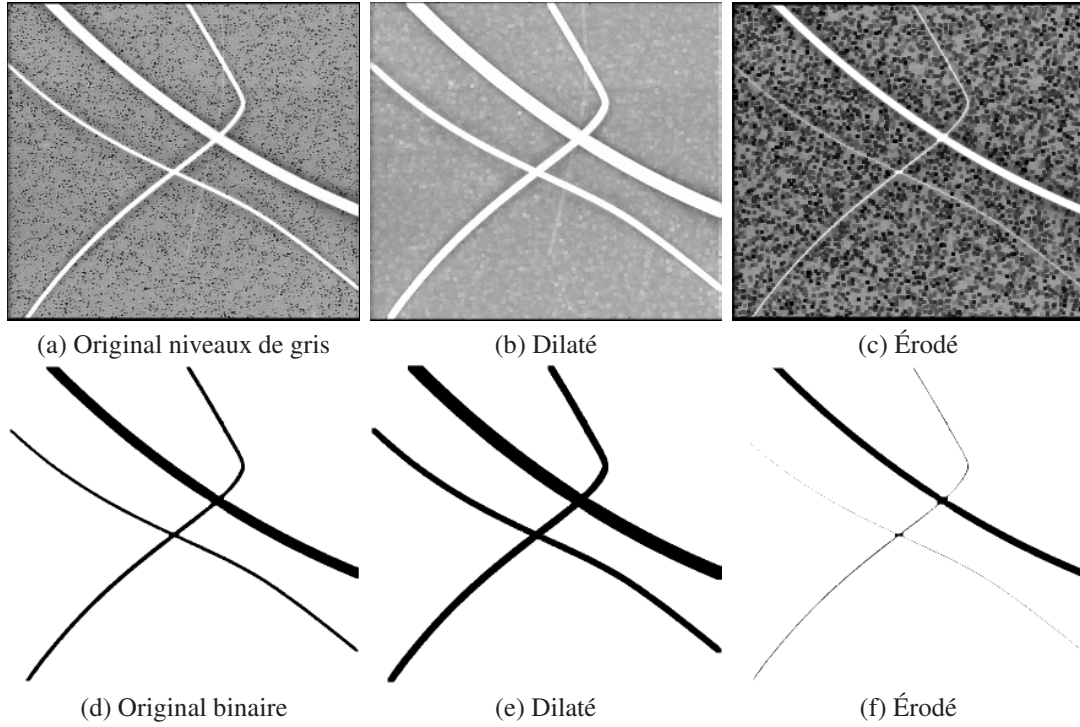
$$= \{p \in E, B_p \subseteq X\} \quad (1.6)$$

L'érosion est le lieu géométrique des points  $p$  tels que  $B_p$  est inclus dans  $X$ . Une érosion « réduit » l'ensemble  $X$ .

L'érosion et la dilatation sont *duales par complémentation* : dilater un ensemble revient à éroder son complémentaire avec l'élément structurant symétrique :

$$(X \oplus B)^c = X^c \ominus \check{B}, \text{ et } (X \ominus B)^c = X^c \oplus \check{B}$$

où  $\check{B} = \{-b, b \in B\}$ .



**Figure 1.3.** Dilatations et érosions en niveaux de gris et en binaire des images de la figure 1.2 par un élément structurant carré symétrique de taille  $5 \times 5$

Afin d'étendre ces opérateurs au cas des niveaux de gris, une approche consiste à prendre comme ensemble le sous-graphe  $SG(F)$  d'une fonction  $F$  :

$$SG(F) = \{(x, t) \in E \times T, t \leq F(x)\}$$

Avec cette approche, dilater (éroder) une fonction à niveaux de gris revient simplement à dilater (éroder) tous ses seuils.

Une approche équivalente consiste à se placer dans le treillis des fonctions, où la structure d'ordre est celle induite par l'ordre sur  $T$  ; en particulier, pour deux fonctions  $F, G \in T^E$ , nous avons :

$$F \leq G \iff \forall x \in E, F(x) \leq G(x).$$

Ainsi, les formules 1.2 et 1.5 se traduisent alors de la manière suivante :

$$\delta_G(F) = SG(F) \oplus SG(G) = \sup_{y \in E} \{F(y) + G(y - x)\} \quad (1.7)$$

et :

$$\varepsilon_G(F) = SG(F) \ominus SG(G) = \inf_{y \in E} \{F(y) - G(y - x)\} \quad (1.8)$$

Dans ces équations, la fonction  $G$  tient le rôle d'*élément structurant fonctionnel*. Cette fonction peut être arbitraire, par exemple on utilise parfois des fonctions paraboliques pour réaliser certaines opérations [BOO 96], comme la fonction distance euclidienne [MEH 99].

### 1.1.2.2. *Éléments structurants plans*

Dans la pratique, les éléments structurants fonctionnels les plus utilisés sont les éléments structurants plans, qui sont des fonctions structurantes nulles sur un support compact  $K$  et prenant pour valeur  $\wedge T$  ailleurs. Dans ce cas, les formules 1.7 et 1.8 se réduisent à :

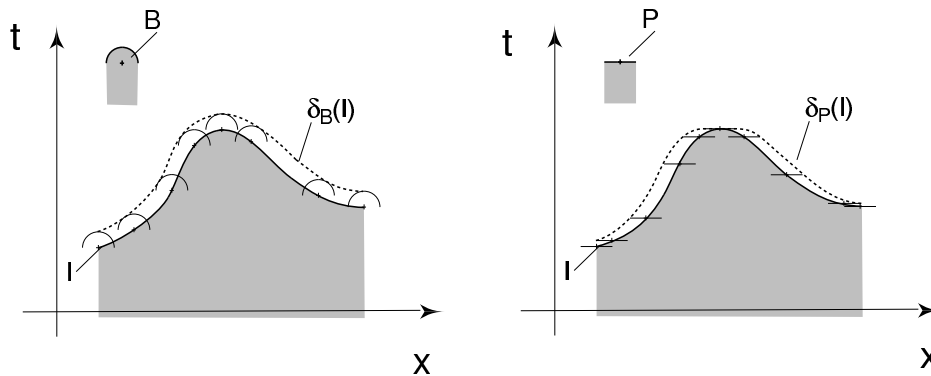
$$\varepsilon_K(F)(x) = \inf_{y \in E, y-x \in K} F(y) = \inf_{y \in K_x} F(y) \quad (1.9)$$

$$\delta_K(F)(x) = \sup_{y \in K_x} F(y) \quad (1.10)$$

Dans ce cas, la vision alternative suivante est particulièrement éclairante : appliquer un opérateur morphologique plan sur une fonction  $F$  revient à appliquer un opérateur morphologique sur chacun des seuils  $X_t(F)$  de la fonction  $F$  ; par exemple, pour la dilatation par un élément structurant plan  $K$ , nous obtenons :

$$\delta_K(F) = \vee \{t \in T, p \in \delta_K(X_t(F))\} \quad (1.11)$$

La figure 1.4 présente un exemple de dilatation d'un signal mono-dimensionnel par une fonction structurante. Les figures 1.3(a,b) présentent le cas bi-dimensionnel.



**Figure 1.4.** Dilatation d'un signal par un élément structurant non plan (fonction structurante) et un élément structurant plan. Dilater par un ES plan revient à prendre en chaque point le maximum de la fonction sur la fenêtre définie par l'ES

On voit donc que les opérateurs morphologiques peuvent s'étendre facilement du cas binaire au cas niveaux de gris. Il est souvent plus facile d'avoir une intuition de ce que fait un opérateur dans le cas binaire, mais lorsque l'on travaille sur une image en niveaux de gris, il est préférable de rester dans ce mode afin de repousser tout seuillage, qui nécessite un paramètre, le plus longtemps possible.

### 1.1.3. *Transformations en tout ou rien*

Les opérateurs de dilatation et d'érosion sont déjà utiles lorsqu'ils sont employés tout seuls, par exemple pour supprimer certains types de bruits, mais ils gagnent en puissance lorsqu'on les combine. Par exemple, on

peut considérer des transformations qui tiennent compte à la fois des points qui appartiennent à l'ensemble et des points qui n'y appartiennent pas. On a alors besoin de deux éléments structurants, l'un noté  $T_1$  s'applique à l'ensemble et l'autre  $T_2$  s'applique à son complémentaire. On note :

$$X \star T = (X \ominus T_1) \cap (X^c \ominus T_2) \quad (1.12)$$

On appelle ces opérations *transformations en tout ou rien* ou *HMT (hit or miss transforms)*. Les transformations en tout ou rien peuvent servir à faire de la reconnaissance de formes :  $X \star T$  est le lieux des points où  $T_1$  s'ajuste à la forme et où  $T_2$  s'ajuste au fond. De nombreuses procédures classiques de simplification de forme, telle la squelettisation, s'appuient sur de telles transformations. Le chapitre sur la squelettisation du volume 2 donne de plus amples informations.

Plus généralement, la composition des opérations morphologiques comme la dilatation et l'érosion mène au filtrage morphologique.

## 1.2. Le filtrage morphologique

En traitement du signal classique, le terme de « filtre » désigne à peu près tous les types de traitement. En morphologie mathématique, ce terme recouvre une signification plus précise, puisqu'il s'agit d'opérateurs croissants et idempotents. Dans ce cadre, les deux principaux opérateurs sont l'ouverture et la fermeture. Une ouverture  $\gamma$  est un opérateur croissant, anti-extensif et idempotent (c'est-à-dire que pour toute image  $I$ ,  $\gamma(\gamma(I)) = \gamma(I)$ ), et une fermeture est un opérateur croissant extensif et idempotent. Cela signifie qu'une ouverture ou une fermeture convergent en une itération, et qu'il ne sert à rien de les utiliser plusieurs fois de suite, du moins avec les mêmes paramètres.

### 1.2.1. Ouvertures et fermetures par éléments structurants

On peut en particulier définir des filtres en composant les opérateurs de dilatation et d'érosion par élément structurant. Ainsi, l'ouverture de  $X$  par  $B$  est définie par :

$$\gamma_B(X) = X \circ B = (X \ominus B) \oplus B \quad (1.13)$$

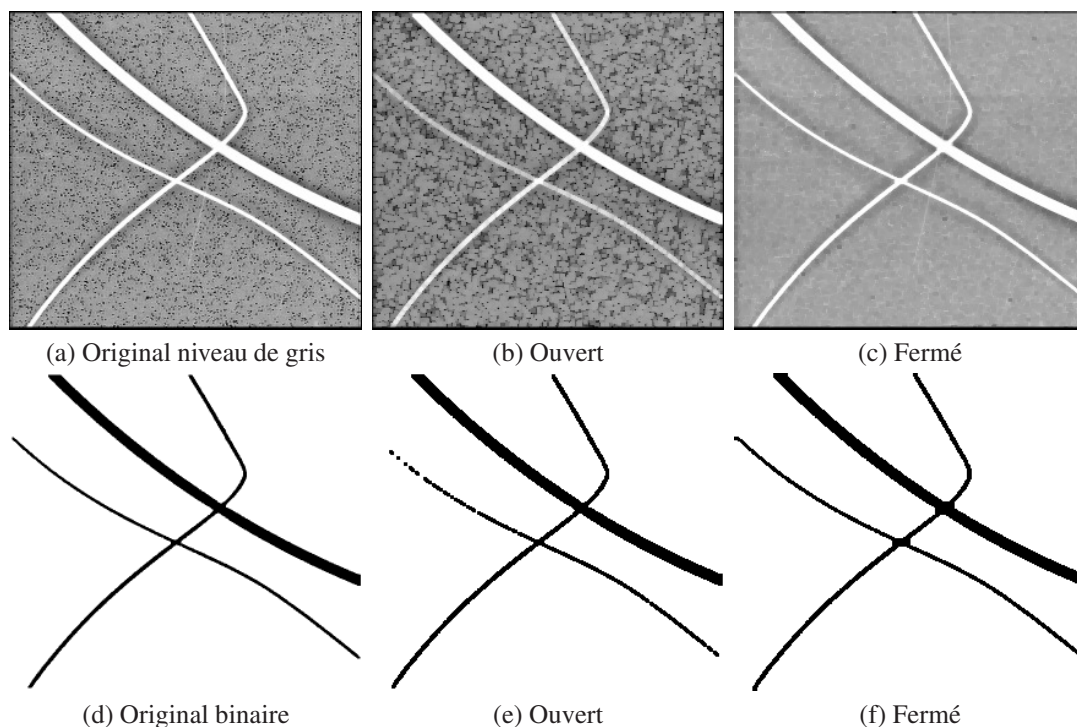
$$= \bigcup \{B_p, p \in E \text{ et } B_p \subseteq X\} \quad (1.14)$$

et la fermeture de  $X$  par  $B$  est définie par :

$$\varphi_B(X) = X \bullet B = (X \oplus B) \ominus B. \quad (1.15)$$

Les formules sont similaires dans le cas niveaux de gris. L'ouverture par élément structurant détruit les parties « étroites » des objets, alors que la fermeture bouche les trous « étroits ». Etroit est vu au sens de l'élément structurant. Des exemples d'ouvertures et de fermetures sont présents dans la figure 1.5.(c,d,e,f).

On qualifie ces ouvertures et fermetures à base d'éléments structurants de *morphologiques*. Cependant on ne peut pas combiner n'importe quelle érosion suivie d'une dilatation et appeler cette opération une ouverture. Les deux opérations qui forment une ouverture sont appelées *adjointes*, par référence à la dualité très particulière qui relie dilatation et érosion, dualité qui ne coïncide pas avec le passage au complémentaire (voir chapitre ??).



**Figure 1.5.** Ouvertures et fermetures en niveaux de gris et en binaire des images de la figure 1.2 par un élément structurant carré de taille  $5 \times 5$

### 1.2.2. Géodésie et reconstruction

Introduisons à présent la dilatation d'un ensemble  $X$  par un élément structurant  $B$  conditionnellement à un ensemble  $R$  généralement appelé *masque* :

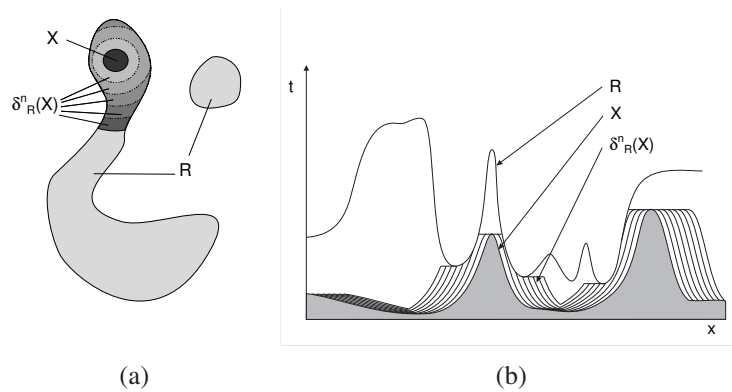
$$\delta_{R,B}^{(1)}(X) = (X \oplus B) \cap R \quad (1.16)$$

Le résultat est toujours compris dans le masque  $R$ . Les dilatations successives sont obtenues par itérations de la dilatation par un petit élément structurant, généralement l'ES fondamental de la trame (voir section 1.2.2.2).

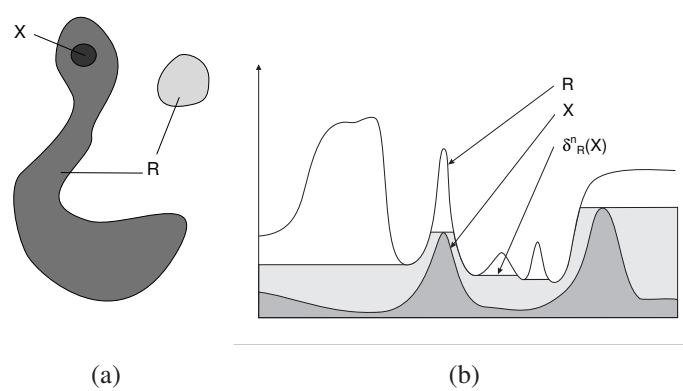
$$\delta_{R,B}^{(n+1)}(X) = \delta_{R,B}^{(n)}(X) \quad (1.17)$$

#### 1.2.2.1. Ouverture et fermeture par reconstruction

Un des premiers intérêts de la dilatation géodésique est de permettre l'opération de *reconstruction*. On appelle *reconstruction* de  $X$  dans  $R$  par  $B$  l'ensemble  $\delta_{R,B}^{\infty}(X)$ . A partir de marqueurs désignant les parties que l'on souhaite en quelque sorte « garder », une dilatation géodésique de taille infinie (en pratique jusqu'à idempotence) permettra de retrouver les formes originelles de ces parties. On voit sur la figure 1.7 un tel exemple de transformation. Pour un marqueur donné, une reconstruction géodésique par dilatation est une ouverture.



**Figure 1.6.** Dilatation géodésique. a : cas binaire, b : cas numérique (élément structurant plan)



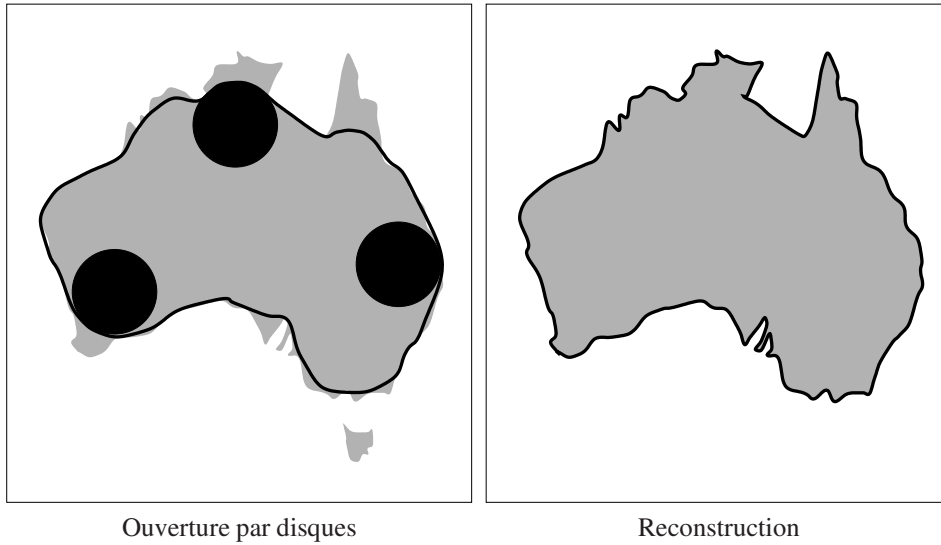
**Figure 1.7.** Reconstruction. On retrouve la forme des ensembles initiaux en a. En b on ne reconstruit pas tout car on a utilisé un ES plan

Comme son nom l'indique, la reconstruction permet de retrouver l'apparence des objets après filtrage. Cet opérateur est illustré en figure 1.8. La composition d'une érosion suivie d'une reconstruction par dilatation est un exemple simple d'ouverture algébrique. Ce type d'ouverture possède toutes les propriétés d'une ouverture, mais ne peut s'exprimer sous forme de la composition d'une érosion suivie par une dilatation. C'est aussi un filtre connexe. Le chapitre ?? donne de plus amples informations à ce sujet.

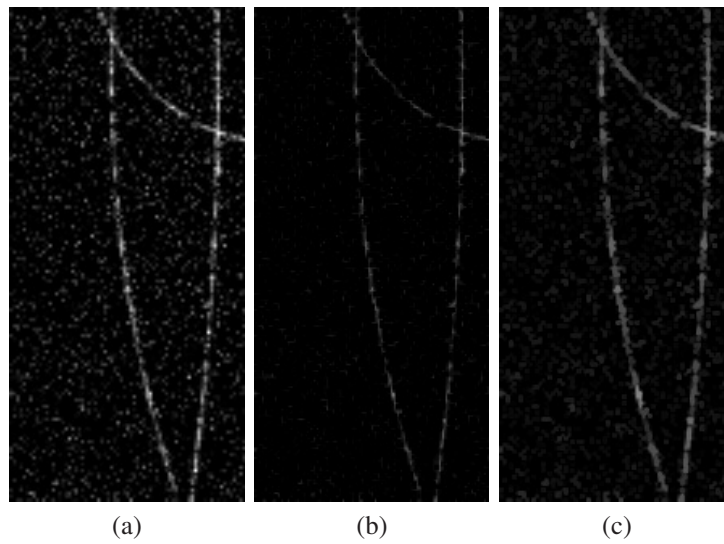
Par complément, on peut aussi définir tout aussi simplement la reconstruction géodésique par érosion, qui va former une fermeture. Bien entendu, ces opérateurs s'appliquent au cas des images en niveaux de gris, comme illustré en figure 1.9.

#### 1.2.2.2. Structure de l'espace, voisinage

Jusqu'à présent, nous n'avons pas parlé de la structure de l'espace  $E$ . Les opérateurs que nous avons introduits n'en sont pas réellement dépendants. Cependant, le cas de la dilatation conditionnelle illustre particulièrement bien le fait que la donnée d'un élément structurant pour une dilatation est équivalent à la donnée d'une



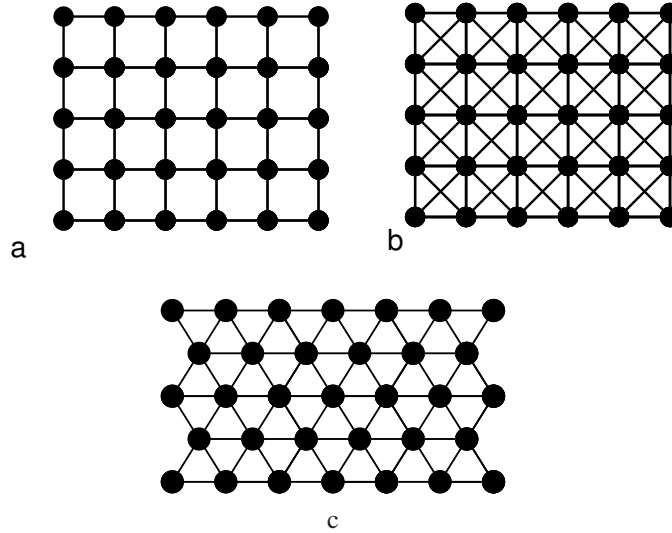
**Figure 1.8.** Ouverture par reconstruction d'une forme (ici une carte de l'Australie). L'érosion fait disparaître l'île de Tasmanie, mais la reconstruction permet de retrouver la forme du continent principal



**Figure 1.9.** Ouverture par reconstruction en niveaux de gris : en a, image originale de traces de particules en chambre à bulle, en b érosion par un ES de taille  $5 \times 5$ , en c, reconstruction par dilatation. On a éliminé la majorité du bruit de scintillement tout en gardant la forme générale des traces



connexité. Nous allons expliciter cela pour le cas de la géométrie discrète sur  $\mathbb{Z}^n$  (mais on pourrait procéder de manière similaire dans le cas continu).



**Figure 1.10.** Portions de trames. Pour le cas carré, on peut choisir d'affecter à chaque point 4 voisins (trame a) ou 8 voisins (trame b). Dans le cas de la trame hexagonale, chaque point a 6 voisins (trame c)

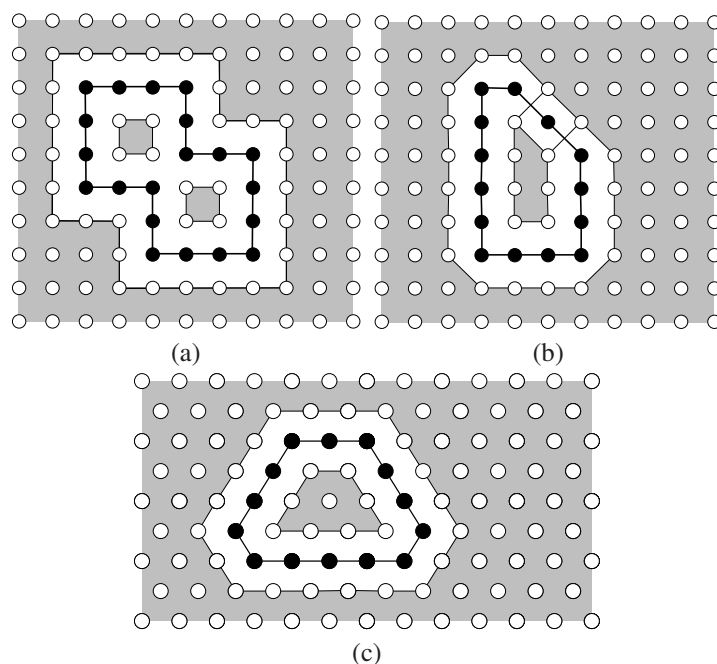
Commençons par introduire la notion de voisinage  $\Gamma$  sur l'espace  $E$ . Dans le cas discret,  $\Gamma$  est une relation binaire sur  $E$  qui est réflexive  $((x, y) \in \Gamma)$  et symétrique  $(x, y \in \Gamma \leftrightarrow (y, x) \in \Gamma)$ . On dit que  $(E, \Gamma)$  est un graphe. Nous noterons également par  $\Gamma$  l'application de  $E$  dans  $2^E$  qui à tout point  $x$  de  $E$  associe  $\Gamma(x) = \{y \in E, (x, y) \in \Gamma\}$ , ensemble des voisins de  $x$ . Si  $y \in \Gamma(x)$ , on dit que  $x$  et  $y$  sont *adjacents*. En traitement d'images, les relations les plus classiques sont définies sur un sous-ensemble  $E \subset \mathbb{Z}^2$ . Par exemple, en 4-connexité, pour tout  $x = (x_1, x_2) \in E$ ,  $\Gamma(x) = \{(x_1, x_2), (x_1+1, x_2), (x_1-1, x_2), (x_1, x_2+1), (x_1, x_2-1)\} \cap E$ . On peut définir de la même manière la 8- ou la 6-connexité (voir figure 1.10). L'application  $\Gamma$  est une dilatation, et inversement, à partir de toute dilatation sur un espace discret, on peut définir un graphe.

### 1.2.2.3. Chemins et connexité

Dans la grille carrée, qui est celle utilisée le plus souvent en pratique en 2D, on ne peut se contenter d'utiliser une seule notion de voisinage. En effet, on s'attend à retrouver dans la trame discrète la propriété de Jordan du plan euclidien, qui dit que toute courbe fermée simple – qui ne se recoupe pas – divise le plan en deux régions connexes, l'une d'extension finie et l'autre infinie. Or dans le cas discret, on ne retrouve pas cette propriété par défaut.

Le problème de Jordan est illustré dans la figure 1.11.

Dans la figure 1.11a, si on suppose la grille 4-connexe, le sous-ensemble du plan à l'intérieur de la courbe n'est pas simplement connexe. En (b), la courbe ne sépare pas l'intérieur de l'extérieur. En revanche, dans la grille hexagonale, on peut démontrer qu'on n'a pas ce type de problème.



**Figure 1.11.** *Propriété de Jordan digitale. La trame carrée ne vérifie pas cette propriété. En a le lacet simple non dégénéré sépare le plan en trois composantes connexes. En b il ne sépare rien du tout. En c la propriété de Jordan est vérifiée, comme toujours en trame hexagonale*

Pour résoudre ce problème dans la trame carrée de manière pragmatique, on considère le plus souvent deux types de connectivité [ROS 73, ROS 75] : une pour les objets (à l'intérieur des courbes) et une pour le fond (à l'extérieur). Une solution mathématiquement plus satisfaisante est de doter la grille discrète d'une vraie topologie, par exemple en suivant Khalimski [KHA 90].

### 1.2.3. Filtrage connexe et nivellements

La combinaison d'ouvertures et de fermetures par reconstruction permet de définir à son tour de nouveaux opérateurs de filtrage, qui vont avoir tendance à étendre les zones plates d'une image. Ces combinaisons sont appelées *nivellements* (voir chapitre ??). D'une manière générale, les nivellements font partie d'une classe plus grande appelée filtrage connexe.

Une représentation permettant un traitement efficace des images pour le filtrage connexe est *l'arbre des coupes*, dont il sera question au chapitre ?. Un cas particulier d'opérateur connexe est l'ouverture par aire (encore appelée ouverture surfacique).

### 1.2.4. Ouverture/fermeture par aire

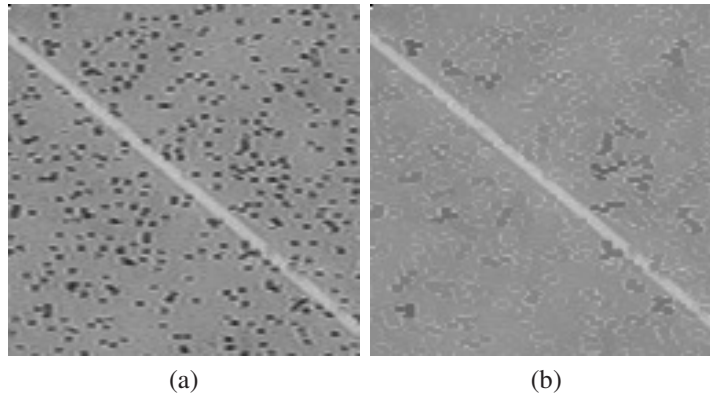
L'ouverture et la fermeture par un élément structurant déforment l'objet. Par exemple, les coins sont arrondis si on utilise une boule. L'ouverture ou la fermeture par aire n'ont pas ce défaut.

Soient  $X \subseteq E$ , et  $x_0, x_n \in X$ . Un *chemin* de  $x_0$  à  $x_n$  dans  $X$  est une suite  $\pi = (x_0, x_1, \dots, x_n)$  de points de  $X$  telle que  $x_{i+1} \in \Gamma(x_i)$ , et dans ce cas,  $n$  est la *longueur* du chemin  $\pi$ . On dit que  $X$  est *connexe* si pour tout  $x$  et  $y$  dans  $X$  il existe un chemin de  $x$  à  $y$  dans  $X$ . On dit que  $Y \subseteq E$  est une *composante connexe* de  $X$  si  $Y \subseteq X$ ,  $Y$  est connexe et  $Y$  est maximale pour cette propriété (c'est-à-dire  $Y = Z$  dès que  $Y \subseteq Z \subseteq X$  et  $Z$  connexe).

De manière informelle, une ouverture par aire élimine les « pâtés » : on peut définir un pâté comme une composante connexe de petite taille. La fermeture par aire comble les trous de manière similaire. La taille est le nombre de points de la composante connexe. Une ouverture par aire enlève les composantes connexes dont l'aire est inférieure à un seuil donné  $\lambda$ .

On peut vérifier facilement qu'une ouverture possède bien les trois propriétés d'une ouverture algébrique : elle est anti-extensive (elle élimine des composantes connexes), croissante et idempotente (les composantes connexes éliminées restent éliminées). De même pour la fermeture par aire.

Une fermeture par aire est illustrée par la figure 1.12.



**Figure 1.12.** Fermeture surfacique de taille 20 (pixels carrés). Les minima de petite taille ont été comblés de façon adaptative. Les pics de l'image n'ont pas été affectés (par exemple, la fibre)

### 1.2.5. Filtres algébriques

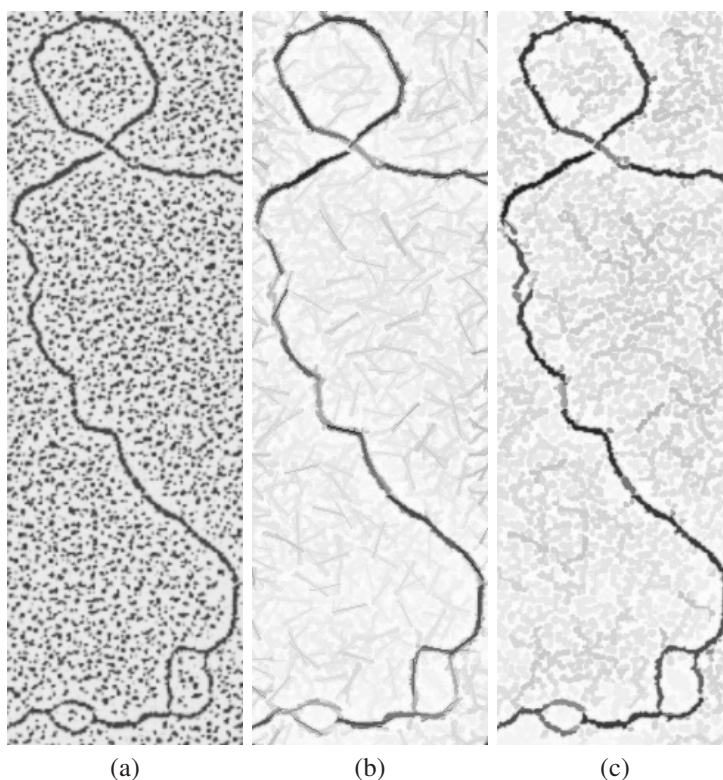
Le filtrage par aire peut s'exprimer de manière différente. Prenons le cas de l'ouverture sur une image binaire. Une composante connexe  $C$  d'aire  $A$  sera préservée par une ouverture par aire avec un paramètre  $\lambda \leq A$ . Clairement il existe une ouverture morphologique par un élément structurant d'aire  $\lambda$  qui préserve  $A$ , par exemple l'ouverture qui utilise  $A$  lui-même (ou un de ses sous-ensemble d'aire  $\lambda$ ) comme élément structurant.

Sachant que l'on doit préserver toutes les composantes connexes possibles d'aire au moins  $\lambda$ , on en déduit qu'on peut considérer, au moins conceptuellement, toutes les ouvertures possibles avec cette famille d'éléments structurants d'aire  $\lambda$ . Il est facile de montrer que le *supremum* de ces ouvertures, c'est-à-dire l'opérateur qui en chaque point préserve le maximum de toutes les ouvertures de la famille, est lui-même une ouverture, et qu'il préserve toutes les composantes connexes d'aire  $\lambda$ . Il s'agit donc précisément de l'ouverture par aire de paramètre  $\lambda$ .

On peut donc mettre en œuvre l'ouverture par aire en calculant toutes les ouvertures par éléments structurants d'aire  $\lambda$ , mais cela serait très inefficace. En effet, la taille de la famille croît exponentiellement avec le paramètre  $\lambda$ . Cependant, la représentation d'une ouverture (ou d'une fermeture) par une famille d'ouverture (respectivement fermeture) morphologique est extrêmement utile. Il existe un théorème dû à Matheron [MAT 75] qui montre l'existence d'une décomposition morphologique pour toute ouverture ou fermeture (voir le chapitre ??, section ??).

D'une manière plus pratique, on retiendra que toute composition par *supremum* d'ouvertures est elle-même une ouverture. Respectivement, une composition par *infimum* de fermetures est elle-même une fermeture. Ces filtres sont appelés ouvertures (respectivement fermetures) *algébriques*.

Dans le volume 2, au chapitre 6, nous verrons comment mettre en œuvre certains filtres algébriques. En forme d'illustration, la figure 1.13 propose une application de filtrage d'un objet fin : la composition de fermetures par des familles d'éléments structurants adaptés, ici soit des segments de droites, soit des chemins [HEI 05], ce qui permet de préserver l'objet d'intérêt tout en filtrant le fond. Comme dans ce cas l'objet d'intérêt n'est pas suffisamment localement rectiligne, les chemins sont mieux adaptés.



**Figure 1.13.** Fermetures algébriques par composition. En a, une image d'ADN en microscopie électronique. En b le résultat de l'infimum des fermetures par une série de segments orientés. En c le résultat de l'infimum des fermetures par une série de chemins

### 1.2.6. Famille granulométrique

L'idée de granulométrie est inspirée du tamisage du sable. En faisant passer du sable au travers d'un tamis, on récolte dans le tamis les particules de plus grande taille que celle du tamis, et au travers les particules plus petites. En utilisant une famille de tamis de tailles variables, on peut ainsi trier le contenu d'un tas de sable.

De la même manière qu'on doit utiliser une famille de tamis compatibles pour obtenir des résultats de tamisage prévisibles, en morphologie on doit utiliser une famille granulométrique d'ouvertures/fermetures de taille croissante. Ces dernières sont indexées par un paramètre  $\lambda$ , souvent entier, telles que :

$$\lambda \geq \mu \Rightarrow \gamma_\lambda \leq \gamma_\mu ; \varphi_\lambda \geq \varphi_\mu$$

Cette propriété s'appelle la propriété d'*absorption*. On impose aussi souvent que  $\gamma_0 = \varphi_0 = \text{Id}$ .

On peut par exemple prendre pour famille granulométrique une famille d'éléments structurants, par exemple en 8-connexité, dans la trame carrée, la famille des carrés  $B_n$  de taille  $(2n + 1) \times (2n + 1)$ . La famille des ouvertures  $\gamma_{B_n}$ , indexée par  $n$ , vérifie la propriété d'absorption. On peut également prendre la famille des ouvertures par reconstruction correspondante, ou encore une famille d'ouverture ou de fermeture par aire, de paramètre croissant.

Nous utiliserons des familles granulométriques pour estimer la taille des objets dans la section 1.4.2.

### 1.2.7. Filtres alternés séquentiels

Ouvertures et fermetures se ressemblent par deux points : elles sont croissantes et idempotentes ; elles diffèrent en ce qui concernent l'extensivité. Cela nous pousse à étudier la classe des opérations qui vérifient les deux premiers points. On les nomme *filtres morphologiques*. Par exemple, le produit d'une ouverture par une fermeture, ou l'inverse, est un filtre, dit alterné.

La théorie du filtrage morphologique permet de réaliser des compositions d'opérateurs de manière efficace. En particulier, on peut introduire ce qu'on appelle les *filtres alternés séquentiels* (FAS) qui sont, comme leur nom l'indique, des compositions d'ouvertures et de fermetures de tailles croissantes, c'est-à-dire formant une granulométrie (ouvertures) ou une anti-granulométrie (fermetures).

Par exemple le filtre alterné séquentiel « blanc » (commençant par une ouverture) peut s'écrire comme :

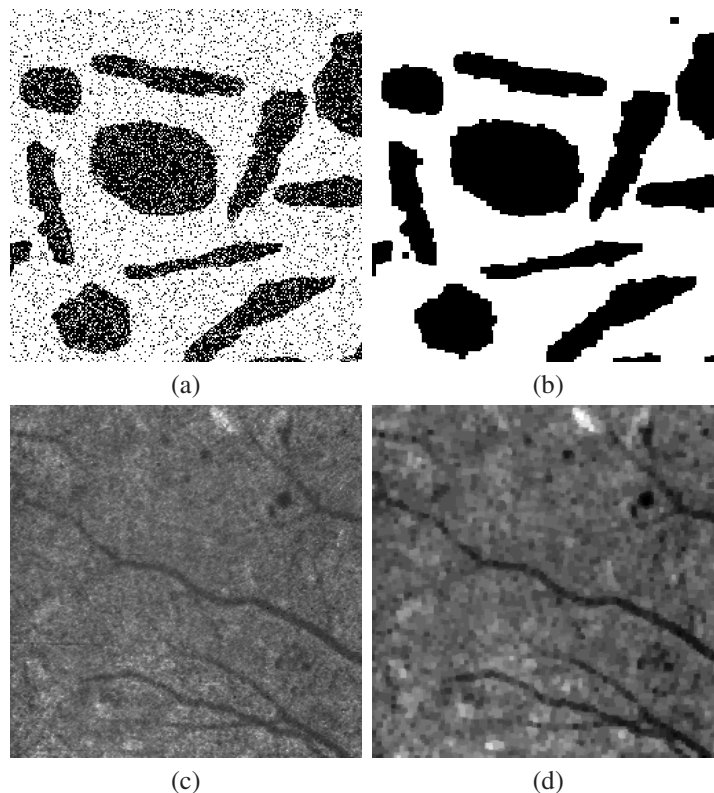
$$\Phi_n(x_i) = \phi_n \gamma_n \phi_{n-1} \gamma_{n-1} \dots \phi_1 \gamma_1 \quad (1.18)$$

et le FAS « noir » (commençant par une fermeture) par :

$$\Psi_n(x_i) = \gamma_n \phi_n \gamma_{n-1} \phi_{n-1} \dots \gamma_1 \phi_1 \quad (1.19)$$

La théorie des filtres morphologiques est assez complexe et ne saurait tenir dans ces quelques pages. Par conséquent nous nous contentons de montrer quelques exemples d'applications élémentaires (figure 1.14).

Les filtres alternés séquentiels peuvent servir à diminuer l'importance du bruit d'origine aléatoire dans les images aussi bien binaires qu'à niveaux de gris, facilitant ainsi une étape ultérieure de segmentation. D'autre part, les filtres alternés séquentiels étant des compositions d'opérations à base d'ES, on peut, en jouant sur



**Figure 1.14.** Utilisation des filtres alternés séquentiels. (a) Image binaire, (b) FAS blanc taille 2, (c) Fond d'œil, (d) FAS noir taille 1

ces ES, amplifier, préserver ou faire disparaître les formes que l'on souhaite, dans les limites du choix des ES utilisables. Généralement, on ne module pas les opérations que l'on souhaite opérer sur ces images en fonction d'une forme statistique du bruit, mais bien précisément en fonction du contenu sémantique de l'image. Les filtres morphologiques sont donc la plupart du temps fabriqués « sur mesure » en fonction du *contenu* de l'image.

La théorie du filtrage morphologique sera développée dans les chapitres ??, ?? et ??.

### 1.3. Résidus

Toutes les opérations que nous venons de voir sont croissantes, c'est-à-dire qu'elles préservent l'ordre. Celles que nous présentons à partir de maintenant ne le sont plus.

En morphologie mathématique, contrairement à d'autres approches, on ne cherche pas à préserver l'information présente dans une image. Au contraire, les opérateurs de base étant non-inversibles, on s'attend précisément à réduire l'information disponible, opération par opération. Une des clés du traitement morphologique est de réaliser ces étapes de filtrage d'une manière contrôlée, afin de préserver au mieux l'information utile et de faire disparaître autant que possible le bruit. Les figures précédentes, par exemple 1.12 ou 1.13 sont des illustrations directes de cette philosophie.

D'une manière générale, on appelle les transformations qui font appel à des opérations morphologiques combinées avec des différences : des *résidus*. Les chapeaux haut-de-forme, les gradients [RIV 93] et quelques autres transformations sont des résidus.

Du fait des propriétés globales des opérateurs de la morphologie, on connaît par avance le comportement de ces résidus. Par exemple, du fait de l'anti-extensivité de l'ouverture, la différence entre l'image originale et une ouverture de cette même image sera toujours positive.

### 1.3.1. Gradients

Le gradient d'une image est sa dérivée. Pour une image 2D et plus, c'est un vecteur en chaque point. En morphologie on considère le plus souvent uniquement la magnitude du gradient. Les gradients permettent de mettre en évidence les zones de forte variation d'intensité dans l'image. Ce sont classiquement les zones de contours des objets.

A partir de la dilatation et de l'érosion on peut définir le *gradient morphologique* :

$$\text{Grad}(F) = \delta_B(F) - \epsilon_B(F) \quad (1.20)$$

et les deux variantes :

$$\text{Grad}^+(F) = \delta_B(F) - F \quad (1.21)$$

$$\text{Grad}^-(F) = F - \epsilon_B(F) \quad (1.22)$$

appelés *gradient externe* et *gradient interne*. En général,  $B$  est la boule unité de la trame considérée. On verra un exemple d'utilisation du gradient à propos de la ligne de partage des eaux.

### 1.3.2. Les transformations chapeaux haut-de-forme

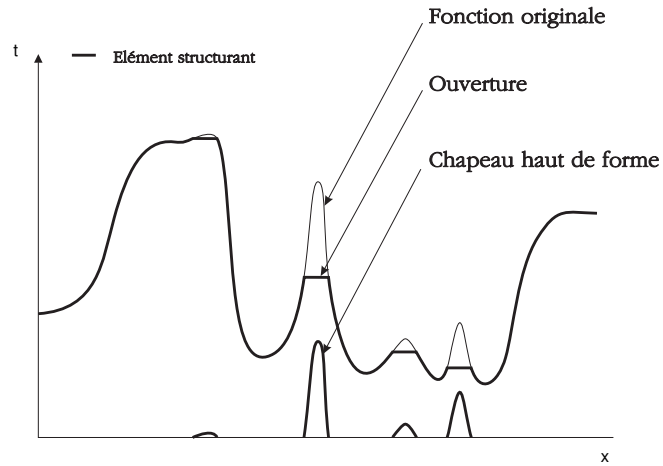
On appelle *transformation chapeau haut-de-forme* ou *top hat* la différence entre l'image originale et l'ouvert de cette image, ou entre le fermé de l'image et l'image originale. Le premier type de chapeau haut-de-forme est appelé *chapeau haut-de-forme blanc* car il permet de détecter ce que l'ouverture a fait disparaître, c'est-à-dire les pics ou parties claires de l'image originale. Le second type de chapeau haut-de-forme, dual du premier, est appelé *chapeau haut-de-forme noir*, car il détecte les vallées, ou parties sombres d'une image. La figure 1.15 illustre le cas des chapeaux haut-de-forme blancs.

Il existe autant de façons d'obtenir des transformations chapeaux haut-de-forme que de types d'ouvertures ou de fermetures.

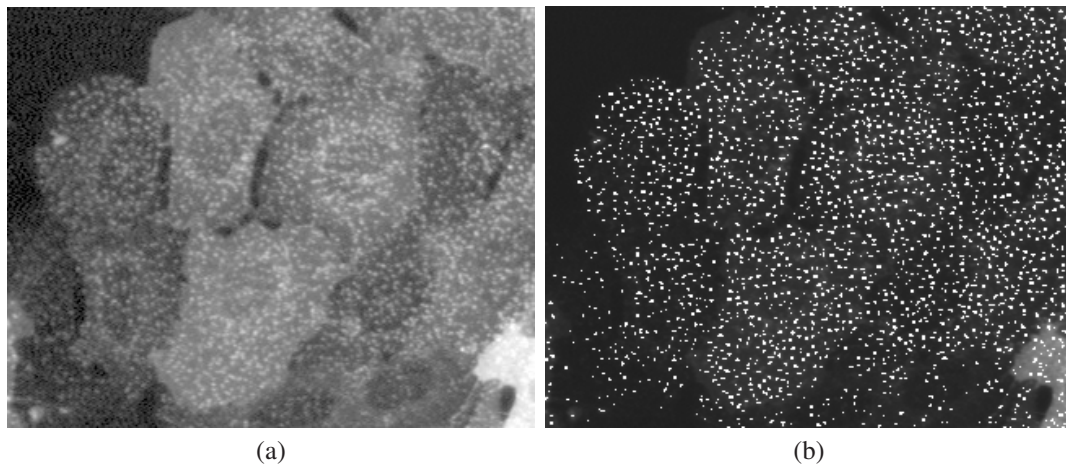
## 1.4. Fonction distance, squelette et courbe granulométrique

Soit  $X \subseteq E$ . La distance de  $x$  à  $y$  dans  $X$  est soit la longueur du plus petit chemin de  $x$  à  $y$  dans  $X$ , soit  $+\infty$  si il n'existe pas de chemin de  $x$  à  $y$  dans  $X$ .

La notion de distance permet d'introduire celle de *fonction distance*, qui à chaque point  $x$  d'un ensemble  $X$  fait correspondre la distance de ce point au plus proche point du complémentaire de  $X$  (voir figure 1.17).



**Figure 1.15.** Top-hat sur un signal 1-D

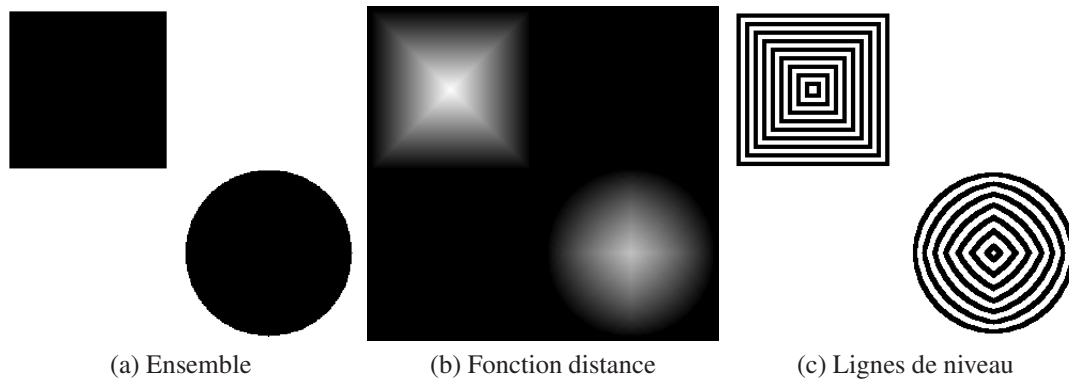


**Figure 1.16.** Top-hat sur une image de cellules. En a l'image originale, les points brillants sont des vésicules rendues fluorescentes par marquage bio-moléculaire. En b le résultat du seuillage du top-hat surimposé en blanc dans l'image de départ

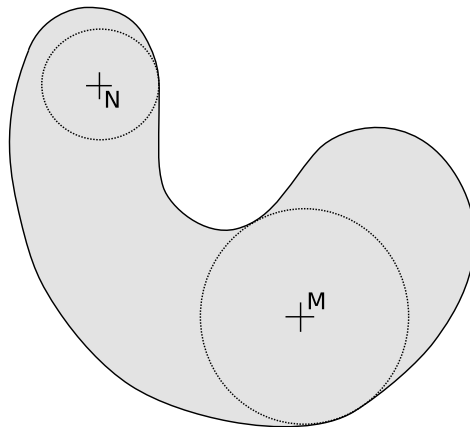
#### 1.4.1. Boules maximales et squelette

Une boule de rayon  $R$  est l'ensemble des points situés à une distance inférieure ou égale à  $R$  d'un point central  $p$ . Il est clair que cette notion dépend de la distance utilisée. Par exemple en 8-connexité, la boule de rayon 3 est un carré de taille  $7 \times 7$ . Lorsque l'on utilise une fonction distance euclidienne, la boule est un disque discret.



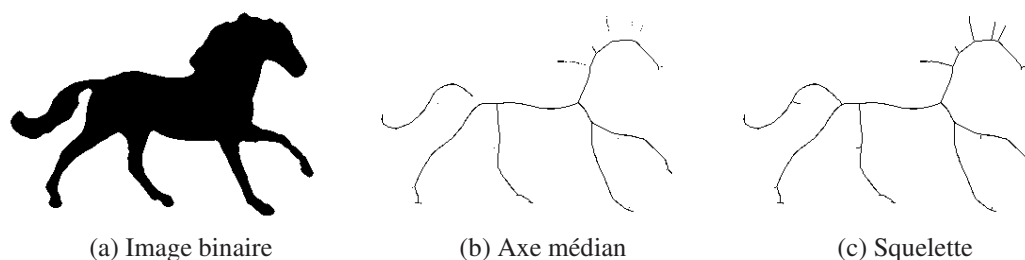


**Figure 1.17.** La fonction distance d'un ensemble est l'application qui en chaque point de l'ensemble associe sa distance au complémentaire. Fonction distance ici en 8-connexité



**Figure 1.18.** Deux centres de boules incluses dans un ensemble binaire. La boule centrée en  $M$  est maximale car aucune autre boule de la même famille (ici des disques) ne peut la contenir et être simultanément incluse dans l'ensemble. La boule centrée en  $N$  n'est pas maximale

Une boule maximale  $B$  relative à un ensemble  $E$  est une boule telle qu'il n'existe pas de boule  $B'$  telle que  $B \subset B' \subset E$  (voir la figure 1.18). Cette notion d'apparence simple permet de définir des résidus intéressants, en particulier le squelette. Cet objet est connu depuis au moins les années 1930 [DUR 30, DUR 31, BOU 32]. Il a été popularisé en traitement des images dans les années 1960 [BLU 61, CAL 68], sous le nom d'*axe médian* de  $E$ , qui peut être défini comme le lieu des centres des boules maximales de cet ensemble.



**Figure 1.19.** Exemple de squelette. (a) une image binaire, (b) axe médian (filtré) de (a), (c) squelette (filtré) de (a) contenant l'axe médian (b)

Dans l'espace euclidien, l'axe médian est appelé le squelette. Le squelette d'un ensemble  $O$  connexe, ouvert, non-vide et borné défini de cette façon possède des propriétés très intéressantes du point de vue géométrique et topologique : il est connexe, homotopique à  $O$  et négligeable au sens de la mesure de Lebesgue<sup>1</sup> [RIV 87].

Dans le cas discret, les centres de boules maximales de  $O$  ne sont pas forcément situés sur la grille, et l'axe médian n'est alors pas connexe. On définit alors informellement le squelette discret  $S(O)$  comme un objet connexe, contenu dans  $O$ , homotopique à  $O$  et d'épaisseur la plus fine possible. Cette notion est disjointe de celle de l'axe médian, mais on peut contraindre un squelette discret à contenir ce dernier. En pratique, les squelettes et axes médians d'images discrètes sont bruités, et il faut appliquer des techniques de filtrage pour pouvoir les utiliser dans les applications. Dans le volume 2, au chapitre 4, on trouvera des définitions précises de ces notions, ainsi que des algorithmes pour les calculer et les filtrer. Un exemple d'axe médian filtré et de squelette filtré contenant cet axe est donné dans la figure 1.19.

#### 1.4.2. Courbe granulométrique

La morphologie mathématique propose plusieurs façons d'obtenir des informations sur la taille des objets dans une image. L'une d'entre elles repose sur la notion de *granulométrie*, qui est dérivée directement de la notion de filtre morphologique (voir la section 1.2.6 de ce chapitre, et aussi le chapitre 4 du volume 2).

La courbe granulométrique d'une image est une représentation de la distribution des tailles dans une image. Les résidus intermédiaires d'une famille granulométrique  $\gamma_n$ , indexée par  $n$ , sont caractéristiques des tailles dans une image. Formellement, la courbe granulométrique est la fonction  $G_f(\lambda)$  définie sur l'intervalle des  $\lambda$ , telle que :

$$G_I(\lambda) = \sum I - \sum \gamma_\lambda(I) \quad \lambda \in [0, 1, 2, \dots, R], \quad (1.23)$$

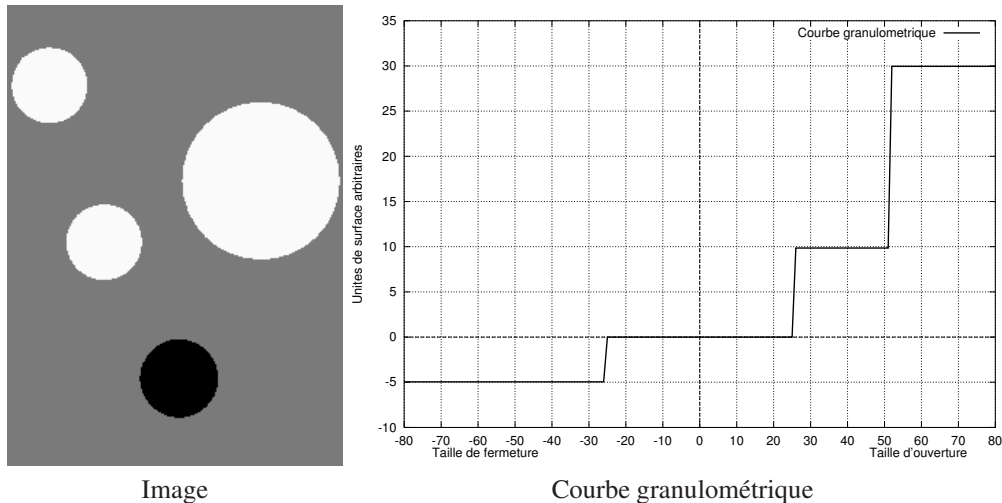
où  $I$  est l'image,  $\sum I$  la somme (ou l'intégrale) des pixels de l'image, et  $R$  la valeur de  $\lambda$  au-delà de laquelle plus rien ne change (car l'image est devenue constante). Pour les fermetures, le signe de la soustraction est échangé (car toute fermeture est extensive, et donc  $\varphi(I) \geq I$ ).

Les courbes granulométriques par ouvertures permettent d'estimer les tailles des pics des images, alors que les courbes granulométriques par fermeture permettent d'estimer les tailles des vallées.

On peut réaliser une courbe granulométrique à la fois par ouverture et par fermeture. Afin d'obtenir une seule courbe continue, par convention la taille du paramètre des fermetures est donnée sur l'intervalle négatif

1. Bien qu'il puisse être partout dense dans  $O$ .

en abscisse, et on a gardé le même sens de soustraction que pour l'ouverture (et donc plus la fermeture est grande, plus le résultat est négatif). Une telle courbe granulométrique est donnée en figure 1.20.



**Figure 1.20.** Une courbe granulométrique par ouverture et fermeture en utilisant une famille de disques euclidiens comme éléments structurants, indexés par leur rayons, sur une image artificielle. Les ouvertures croissantes sont vers les abscisses positives, et les fermetures croissantes dans le sens des abscisses négatives

Sur cet exemple, on illustre le fait que la courbe granulométrique enregistre le volume d'image qui disparaît au-delà d'une certaine taille d'ouverture/fermeture.

#### 1.4.2.1. Applications

La courbe granulométrique permet de résumer la distribution des tailles d'objets dans une image, sans passer par une étape de segmentation. Par conséquent, la notion d'objet n'est pas bien définie dans ce contexte. On peut seulement parler de volumes de niveaux de gris. D'autre part, comme il s'agit d'une simple courbe, le contenu en est forcément synthétique, et n'est donc pas toujours d'interprétation facile. On peut souvent s'en servir comme d'une indication de texture, voir par exemple une étude du vieillissement de conduites de vapeur pour génération d'électricité [BRE 00], illustrée en figure 1.21.

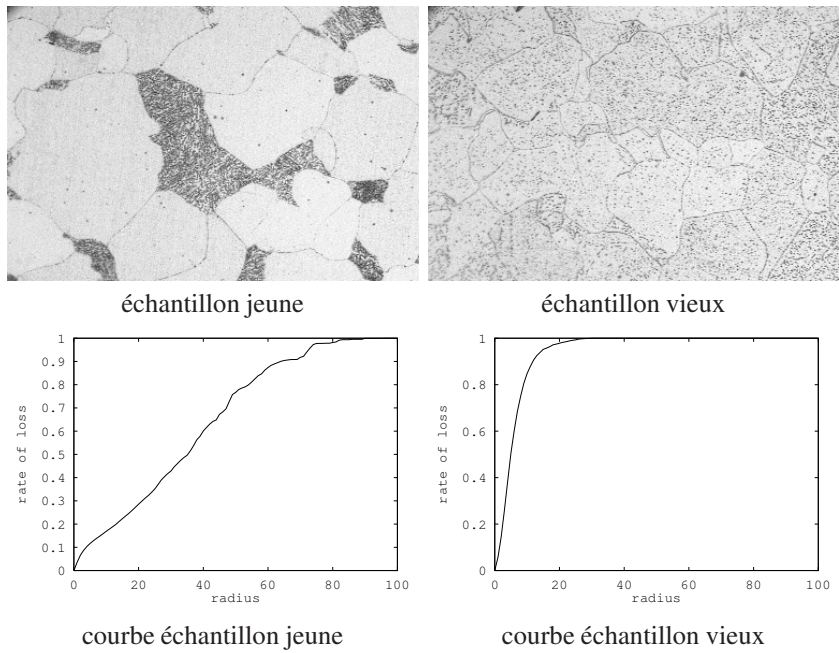
De nombreuses applications utilisent les granulométries pour estimer des paramètres d'opérateurs liés à la taille, par exemple [COM 07].

#### 1.4.2.2. Granulométries par érosions et dilatation

On peut également réaliser des granulométries en n'utilisant que la notion d'érosion et/ou de dilatation. Il y a alors un lien fort avec le squelette [MAT 92].

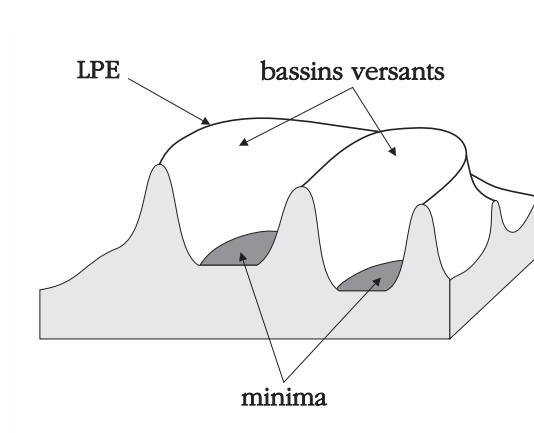
### 1.5. Ligne de partage des eaux

En reprenant l'analogie des images à niveaux de gris avec un relief topographique, on peut définir une transformation topographique intéressante qui est la *ligne de partage des eaux* (LPE). Imaginons une goutte d'eau



**Figure 1.21.** Application des granulométries : courbes granulométriques d'images de microscopie de surface d'un conduit de vapeur en fonte, pour un échantillon d'un conduit jeune, et d'un conduit vieux. L'âge détériore les joints de grains, ce qui apparaît clairement dans les deux courbes granulométriques

tombant sur le relief créé par l'image. Celle-ci va s'écouler jusqu'à ce qu'elle atteigne un minimum régional. A chaque *minimum* régional  $M$  on peut associer l'ensemble des points  $p$  tels qu'une goutte d'eau tombant en  $p$  rejoigne  $M$ . On appelle cet ensemble un *bassin versant*. Les points situés à la frontière de deux bassins constituent un ensemble de contours fermés qu'on appelle la *ligne de partage des eaux* (voir figure 1.22).



**Figure 1.22.** La ligne de partage des eaux

Une autre vision de la ligne de partage des eaux consiste non pas à partir des endroits où tombent des gouttes d'eau, mais d'« inonder » l'image en partant des *minima* de celle-ci. La ligne de partage des eaux est alors l'ensemble des points où au moins deux nappes d'eau se rencontrent. Ces deux visions ne sont en général pas compatibles [NAJ 94, NAJ 05, COU 07b].

Si la littérature sur les algorithmes de LPE est très fournie [BEU 79b, VIN 91c, MEY 94, ROE 01, COU 05], les études des propriétés de ces algorithmes sont relativement récentes [NAJ 94, BER 05, NAJ 05, COU 07b]. La formalisation des différentes visions de LPE dans le cadre discret, ainsi que les propriétés mathématiques des objets ainsi obtenus, sont présentées dans le chapitre ??.

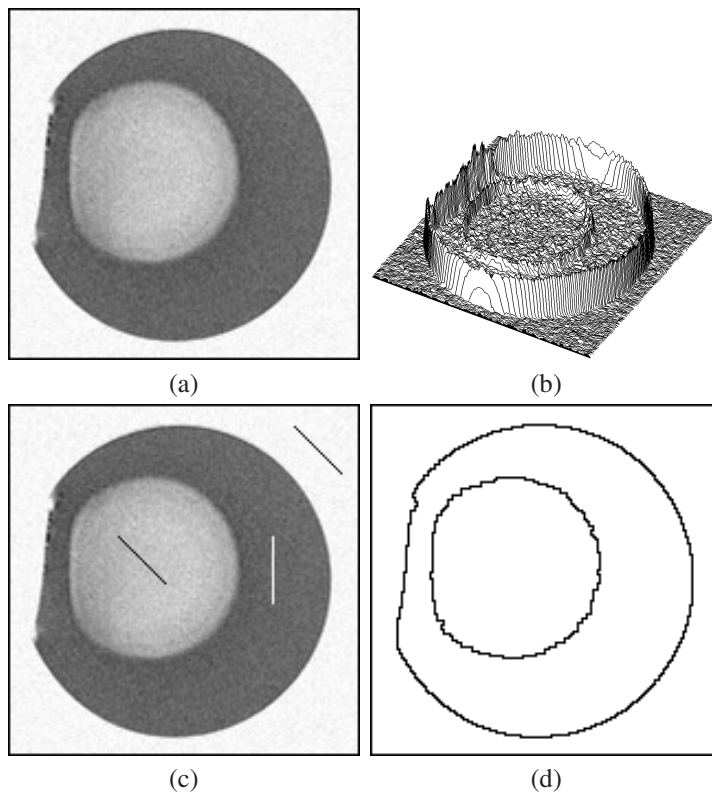
La ligne de partage des eaux est à la base d'une méthodologie puissante et souple introduite au centre de morphologie mathématique dans les années 1970 [BEU 79b] et développée au début des années 1990 [MEY 90b, VIN 91c]. Par la suite, cette méthodologie a été unifiée dans les approches hiérarchiques [BEU 94, NAJ 96]. L'idée générale est la suivante : on commence par définir ce qu'on recherche en exhibant des *marqueurs intérieurs* à l'objet, c'est-à-dire un ensemble binaire éventuellement labellisé (de niveau de gris différent pour chaque particule connexe) entièrement inclus dans les objets que l'on recherche. La forme de ces marqueurs n'est pas importante, seules leur localisation et leur extension le sont. On cherche de la même manière un ou des *marqueurs extérieurs* à l'objet. On considère le *gradient* de l'image. On reconstruit ensuite le gradient en imposant comme *minima* ces marqueurs (et en éliminant les *minima* « naturels » du gradient). Une et une seule ligne de partage des eaux est alors présente entre chaque marqueur, et elle a tendance à se situer sur le contour des objets à segmenter, qui ont déjà été pré-détectés par le gradient.

Illustrons à l'aide d'un exemple, la segmentation d'un œuf en RMN [GRA 93a], visible sur la figure 1.23. Sur cet exemple les marqueurs du jaune, du blanc et de l'extérieur de l'œuf sont donnés à la main. Il est possible [GRA 93a] de rechercher automatiquement des marqueurs de ces parties. Contrairement à l'apparence très simple de l'image, un seuillage ne donne pas de bons résultats, alors que la segmentation par LPE donne un résultat pratiquement parfait.

La méthodologie de segmentation morphologique sera développée dans le chapitre ??.

## 1.6. En guise de conclusion

Ce chapitre, par essence introductif, ne contient qu'une petite partie des opérateurs de la morphologie mathématique. Nous espérons que ce texte donnera au lecteur l'envie d'aller plus loin dans la théorie. Un tel lecteur trouvera alors dans ce volume et dans le suivant de quoi combler ses attentes. Il pourra par exemple dans le volume 2 s'intéresser aux approches stochastiques et floues, ou encore chercher à comprendre comment étendre la morphologie aux espaces pour lesquels il n'existe pas de structure d'ordre immédiatement évidente.



**Figure 1.23.** En a image originale, en b le gradient (vu en relief), en c les marqueurs (ici imposés à la main) et en d la segmentation obtenue. (Image originale : image d'un œuf en RMN, source Neils Roberts, University of Liverpool)

## Chapitre 2

# Introduction à la mesure en analyse d'image

### 2.1. Introduction

On fait parfois la distinction entre :

- le *traitement d'image*, qui consiste à transformer des images en de nouvelles images (avec peut-être moins de bruit par exemple) ;
- l'*analyse d'image*, qui consiste à produire des informations numériques à partir d'images : des nombres, des diamètres, des longueurs, etc ;
- la *synthèse d'image*, qui consiste à utiliser des nombres pour produire des images artificielles.

L'ouvrage présent est plutôt orienté vers le traitement d'image utilisant des méthodes à base de morphologie mathématique, mais ces méthodes peuvent aussi être utilisées avec profit dans le contexte des deux autres disciplines. L'analyse d'image, en particulier, est le sujet de ce chapitre.

En général, l'analyse est une étape qui suit le filtrage et la segmentation, afin de répondre à des questions scientifiques ou techniques provenant d'informations d'origine visuelle. Les détails de l'analyse proprement dite peuvent varier suivant le domaine d'application, qui incluent l'imagerie médicale et biomédicale, la science des matériaux, la biométrie, la surveillance vidéo, la vision par ordinateur, et les applications grand public (reconnaissance de visage, détections de sourires dans les appareils de photos numérique par exemple). Une question qui mérite d'être posée est celle de l'existence ou non d'un dénominateur commun à tous ces domaines.

On peut également se poser la question de savoir si le traitement d'image nécessite une ou des étapes de mesure. En premier lieu ça ne semble pas être le cas, mais en réalité des mesures sont effectuées constamment, soit de manière explicite – comme but final dans l'analyse d'image, soit de manière implicite, comme par exemple dans la vision par ordinateur : comment reconnaître qu'une photographie représente un paysage comprenant une vue sur la mer ? Une façon de procéder est de construire un arbre de décision comprenant une grande quantité de mesures. Un autre exemple est le codage vidéo utilisant la norme MPEG, où des mesures de similarité entre portion d'images sont effectuées constamment afin d'éviter les répétitions dans le flux transmis.

L'éventail des situations possibles est en fait très grand. Le contexte sémantique est souvent plus limité en analyse d'image qu'en vision par ordinateur par exemple, car l'analyste sait en général à quoi il a affaire : des bactéries, des organes, des os ou des cristaux par exemple. En revanche, les mesures effectuées sur ces différents objets peuvent être complexes, délicats, et obtenus par des moyens limités et indirects. Par exemple on peut vouloir souhaiter obtenir des informations de volume à partir de coupes sériées : par exemple estimer la progression d'une ostéopathie requiert de mesurer la taille de pores en 3D dans des os. Un emphysème réduit la surface de contact des poumons entre l'air et le flux sanguin : peut-on estimer cette surface de contact ? Certains matériaux composites développent des dendrites, souhaitables ou non. Peut-on effectuer des mesures significatives sur ces mêmes dendrites, sans risquer de confondre un grand nombre de petites dendrites avec un plus petit nombre de grandes dendrites ?

Le but de ce chapitre est de clarifier ces questions. La suite est divisée comme suit : tout d'abord nous allons spécifier quelles sont les caractéristiques désirables des mesures que nous pouvons effectuer dans des images. Quelles mesures se « comportent bien » et lesquelles non ? Quelles mesures sont elles fiables et lesquelles sont facilement entachées d'erreurs ? Nous formaliserons ceci en définissant un espace d'objets sur lesquels des mesures fiables sont réalisables, et nous préciserons leurs propriétés. Nous spécifierons comment construire ces mesures et nous introduirons la *stéréologie*. Ensuite nous étudierons les propriétés des mesures en cas de changement d'échelle, ainsi que la différence entre mesurer des individus et des populations. Finalement nous étendrons ces considérations aux images à niveaux de gris (images numériques), permettant dans certains cas d'obtenir des mesures sans segmentation préalable.

## 2.2. Principes généraux

Afin de pouvoir associer des nombres aux images étudiées, on doit s'appuyer sur trois ordres de préoccupations.

1) On cherche en général à s'affranchir de contingences de la position dans le champ et du grossissement : si l'on modifie le cadre d'une photographie, on ne s'attend pas à ce que la personne photographiée change de taille. Pour certains objets par exemple des globules rouges, qui sont naturellement isotropes) la mesure ne doit pas non plus dépendre de l'orientation, alors que pour d'autres, c'est le contraire (dans le cas des caractères imprimés sur une page, par exemple. On s'attend à ce qu'un logiciel de reconnaissance de caractère fonctionne mieux si la page est bien alignée). Enfin, si on change le grossissement d'une image, par exemple au moyen d'un zoom optique, il est primordial que la variation de la mesure en fonction du grossissement ne dépende que de ce dernier. Par exemple, si on souhaite mesurer l'aire d'un objet, et qu'on augmente le facteur de zoom par deux, on s'attend à ce que l'aire mesurée en pixels carrés soit multipliée par quatre, et non pas n'importe quel autre facteur. Si on mesure un périmètre, celui-ci doit varier linéairement avec le facteur de zoom. On souhaite aussi le plus souvent réaliser des mesures homogènes au sens physique, c'est-à-dire de dimension donnée : l'aire d'un cercle est une mesure homogène, son périmètre aussi, mais pas la somme « aire + périmètre ».

2) Lorsque l'objet n'est pas vu en totalité mais seulement au travers d'un champ limité de prise de vue, se posent alors plusieurs problèmes supplémentaires : quelle est l'influence du champ sur les objets qui sont à l'intérieur ? En particulier (mais pas seulement) que faire avec les objets qui ne sont que partiellement visibles ? D'autre part, comment inférer statistiquement ce que l'on observe à l'échantillon tout entier ?

3) Souvent réaliser des mesures sur un objet nécessite une étape de reconnaissance et d'individualisation. Cette opération s'appelle une *segmentation* (voir chapitre ?? dans le volume « Morphologie Mathématique 1 », dédié à cette question). Dans la plupart des problèmes d'analyse d'image, il existe plus d'un objet d'intérêt dans chaque champ de mesure, et plus d'une image d'un échantillon est acquise. Cependant, segmenter et mesurer un échantillon tout entier est long et difficile, voire impossible (par exemple dans le cas des sections histologiques). La question qu'on peut se poser est alors celle de la représentativité de ce qui aura été effectivement



échantillonné et segmenté par rapport à la « réalité ». Cette question est liée à celle des distributions de taille. Nous apporterons quelques éléments de réponse en section 2.6.3.

Ces contraintes de nature physique sont en partie des choix, sans nécessité universelle, et en partie des contraintes imposées par le monde physique. Il arrive que les objets d'étude s'incluent facilement dans le champ d'analyse (par exemple en cytologie), ou que l'on s'abstienne de segmenter (en utilisant des granulométries par exemple). Cependant, il faut que nos mesures soient valables même en dehors de ces contextes favorables.

Dans la suite de ce chapitre, nous commençons par traiter le cas des ensembles binaires, correspondant aux segmentations, puis nous étendons les résultats aux fonctions numériques en section 2.7. Nous avons également choisi de présenter les choses d'abord dans l'espace euclidien, et d'en faire dériver les mesures discrètes dans un deuxième temps.

### 2.3. Anneau convexe et fonctionnelles de Minkowski

Cette section est dévolue au cas des objets entièrement inclus dans le champ d'analyse. Nous modélisons ces objets par des ensembles euclidiens  $X \subset \mathcal{P}(\mathbb{R}^d)$ . En général la classe des parties de l'espace euclidien  $\mathcal{P}(\mathbb{R}^d)$  est trop vaste et trop complexe pour modéliser le monde réel. En effet, certains de ces éléments n'ont même pas de mesure de Lebesgue<sup>1</sup>. L'ensemble des compacts de  $\mathbb{R}^d$  est encore trop général : en effet, alors qu'un ensemble compact aura toujours une mesure de Lebesgue (aire en 2D, volume en 3D), certains objets, dits fractals, peuvent avoir un périmètre (en 2D) ou une surface (en 3D) infini, tel le flocon de Koch [MAN 83]. Nous allons nous restreindre un peu plus en suivant H. Hadwiger [HAD 57], en considérant la classe des unions finies d'ensembles convexes. On appelle cette classe l'*anneau convexe*, et on la dénote par  $\mathcal{CR} = \mathcal{CR}(\mathbb{R}^d)$ . Cette classe est fermée par translation, rotation, union finie et par intersection. Elle est aussi compatible avec la notion d'images discrètes, qui ne sont que des unions finies de pixels ou de voxels convexes. Nous verrons que cette classe est réceptive à la notion de « bonne » mesure, et qu'elle nous permet de définir le concept clé de caractéristique d'Euler-Poincaré.

**DÉFINITION 2.1 (Fonctionnelle de mesure).**— On appelle fonctionnelle (de mesure) sur l'anneau convexe toute fonction numérique  $W$  à valeur réelles :  $W : \mathcal{CR} \rightarrow \mathbb{R}$ .

Une fonctionnelle peut satisfaire plusieurs propriétés utiles.

**Homogénéité** La fonctionnelle  $W$  sur  $\mathcal{CR}(\mathbb{R}^d)$  est homogène quand :

$$W(kX) = k^p W(X)$$

On voit que quand  $X$  varie par homothétie (grossissement),  $W(kX)$  ne dépend que de  $W(X)$  et de  $k$ , comme nous l'avions souhaité.

**C-Additivité** Un autre problème se pose : les fonctionnelles sont des mesures *globales* sur les ensembles. Comment leur associer des caractéristiques locales, c'est-à-dire sur des petits voisinages autour de chaque point qui constituent ceux-ci ? La réponse est donnée par la condition de C-additivité, c'est-à-dire :

$$W(X \cup Y) = W(X) + W(Y) - W(X \cap Y) \quad X, Y \in \mathcal{P}(\mathbb{R}^d)$$

---

1. Par exemple les ensembles de Vitali, qui peuvent être construits en utilisant l'axiome du choix.

De proche en proche, elle permet de passer de l'ensemble initial  $X \cup Y$  à des morceaux de plus en plus petits. En pratique, elle nous conduit aux éléments structurants élémentaires (dans  $\mathbb{Z}^2$ , d'un, de deux, ou de quatre pixels). La  $C$ -additivité permet aussi de combiner des informations issues d'échantillons épars à travers l'espace.

**invariance par déplacements** Comme décrit plus haut, on s'attend à ce que les mesures réalisées sur un ensemble binaire ne dépendent pas de sa position dans l'image, et pour certaines, ni de son orientation. On parle d'invariance par *déplacement* et non par isométrie, en séparant les parties translation et rotation.

Si  $b$  est un vecteur de  $\mathbb{R}^d$ ,  $X_b$  est l'ensemble  $X$  translaté du vecteur  $b$ ,  $\alpha$  est un angle et  $X^\alpha$  est l'ensemble  $X$  soumis à une rotation d'angle  $\alpha$ . On a les deux invariances suivantes :

$$W(X_b) = W(X) \quad (\text{Invariance par translation}) \quad (2.1)$$

$$W(X^\alpha) = W(X) \quad (\text{Invariance par rotation}). \quad (2.2)$$

L'association des deux invariances est aussi appelée invariance par *isométrie* ou encore invariance par transformation rigide, voir la section ?? au chapitre ??.

**régularité** Dans la mesure du possible, on souhaite réaliser des mesures qui ne varient pas grandement en fonction du bruit ou de la procédure de discrétisation. Ceci s'exprime par une condition de *continuité*, mais celle-ci est difficile à exprimer simplement, et d'autre part une contrainte trop forte de régularité obligerait à s'affranchir de mesures utiles. On utilise le souvent une notion de *continuité conditionnelle*, qui est la suivante.

**DÉFINITION 2.2 (Continuité conditionnelle).**– Une fonctionnelle  $W$  de  $\mathcal{CR}(\mathbb{R}^d)$  est continue conditionnellement si et seulement si elle est continue sur la sous-classe des ensembles convexes compacts de  $\mathbb{R}^d$  pour la métrique de Hausdorff. On définit la monotonie conditionnelle de manière similaire.

La combinaison des quatre propriétés précédentes a été proposée par H. Minkowski à propos des ensembles convexes. Sa définition peut être transposée au cas de l'anneau convexe par la définition suivante.

**DÉFINITION 2.3 (Fonctionnelle de Minkowski).**– Une fonctionnelle  $W$  de  $\mathcal{CR}(\mathbb{R}^d)$  qui est définie, positive bornée, homogène,  $C$ -additive, invariante par déplacements, conditionnellement continue et conditionnellement monotone est appelée une fonctionnelle de Minkowski.

Il est utile de prendre quelques exemples et contre-exemples de fonctionnelles de Minkowski.

– L'aire d'un objet en 2D. Soient deux ensembles compacts convexes  $X$  et  $Y$  dans l'espace  $\mathbb{R}^2$ , et  $A$  la mesure de Lebesgue. On voit que  $A$  est une fonctionnelle de Minkowski sur ces objets : elle est clairement  $C$ -additive, on a  $A(kX) = k^2A(X)$ ,  $A$  est invariante par translation et rotation, et  $A$  est continue et croissante.

– Le périmètre d'un objet est également une fonctionnelle de Minkowski, bien qu'elle ne soit que *convexe-continue* (c'est-à-dire continue uniquement dans le cadre des convexes).

– La fonctionnelle de comptage. Soit  $n$  un entier fini, considérons  $n$  ensembles  $X_i$  connexes de  $\mathbb{R}^2$  et la fonctionnelle  $\mathcal{L}(X_i)$  qui à chaque ensemble connexe associe un unique entier  $1 \leq \mathcal{L}(X_i) \leq n$  différent pour chaque  $X_i$ . Cet entier est le *label* de  $X_i$ . Cette fonctionnelle n'est pas de Minkowski car elle n'est pas  $C$ -additive.

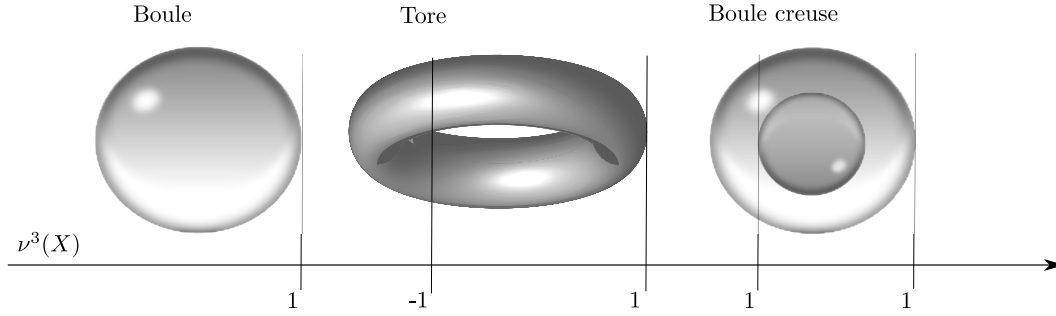


Figure 2.1. Caractéristique d'Euler-Poincaré pour certaines figures en 3D

### 2.3.1. La caractéristique d'Euler -Poincaré.

Dans l'anneau convexe, toutes les fonctionnelles de Minkowski proviennent, par intégration, d'une même origine de degré zéro, qu'on appelle la caractéristique d'Euler-Poincaré (CEP), ou encore le nombre de connexité.

THÉORÈME 2.4 (Hadwiger).– L'unique fonctionnelle définie sur l'anneau convexe  $\mathcal{K}(\mathbb{R}^d)$ , qui est  $C$ -additive et constante pour les compacts convexes est la caractéristique d'Euler-Poincaré  $\nu^d$ .

La constance pour les convexes entraîne que la CEP est de dimension 0, donc invariante par homothétie et par déplacements. Grâce à la  $C$ -additivité, son calcul ne met en jeu que des éléments structurants infinitésimaux. Elle se définit dans  $\mathbb{R}^d$  récursivement en partant de 0. Cette construction est décrite en détails dans [HAD 57], nous ne reprenons ici que les points importants :

- en dimension 0, l'espace est réduit à un point et  $\nu^0(X)$  vaut 1 si  $X$  est ce point, et 0 si  $X$  est l'ensemble vide ;
- en dimension 1,  $\nu^1(X)$  compte le nombre de segments disjoints dont est composé  $X$  ;
- en dimension 2,  $\nu^2(X)$  correspond au nombre de grains de  $X$  (c'est-à-dire le nombre d'objet disjoints composant  $X$ ) moins son nombre de pores (c'est-à-dire le nombre de trous disjoints dans les grains de  $X$ ) ;
- en dimension 3,  $\nu^3(X)$  correspond à la somme du nombre de minima et de maxima dans une direction d'échantillonnage, diminué du nombre de points selle. Ainsi, la CEP d'une sphère vaut 1, celle d'un tore 0 et celle d'une couronne sphérique 2 (voir figure 2.1). Elle est lié au *genre* de  $X$ .

En topologie algébrique on définit le *genre* d'une surface 2D plongée dans  $\mathbb{R}^3$  par le nombre de boucle que l'on peut tracer sur la surface sans la déconnecter. Ainsi le genre de la surface d'une sphère est 0, de la surface d'un tore 1 et celle d'un tore avec une poignée 2 (et ainsi de suite). On montre que la CEP d'un ensemble  $X$  en 3D est déduit de l'équation suivante :

$$\nu^3(X) = \sum [1 - G(\partial X_i)], \quad (2.3)$$

où  $G(A)$  est le genre de la surface  $A$  et  $\partial X_i$  une composante connexe de la frontière de  $X$ . Dans le cas de la sphère, du tore et de la couronne sphérique, on retrouve bien les valeurs attendues illustrées à la figure 2.1.

La CEP se définit également récursivement en dimension arbitraire supérieure à 3, mais nous nous arrêtons ici, comme l'espace 3D correspond à l'espace physique classique.

### 2.3.2. Interprétation de la caractéristique d'Euler-Poincaré

Comme indiqué par l'équation 2.3, il y a un lien important entre la CEP et l'étude algébrique et topologique des surfaces, datant du début du XX<sup>e</sup> siècle avec Poincaré. Pour rappel, les variétés 2D plongées dans un espace  $\mathbb{R}^d$  sont classifiées en *orientables* (telles la surface du tore, le cylindre, la sphère) ou *non-orientables* (telles la bouteille de Klein ou le ruban de Moebius). Le bord unique d'un ruban de Moebius peut-être déformé continûment en un cercle, formant une surface qui dans  $\mathbb{R}^3$  s'auto-intersecte, et qui porte plusieurs noms, par exemple surface d'Apéry ou en anglais *crosscap*. Le théorème de classification des surfaces fermées est le suivant.

**THÉORÈME 2.5 (Classification des surfaces fermée).**– *Toute surface fermée est homéomorphe à une sphère à laquelle on a ajouté (1) un nombre  $p$  de poignées, obtenues en ôtant  $2p$  disques à la sphère et en y connectant un cylindre à chaque paire de trous ; et (2) un nombre  $q$  de surfaces d'Apéry, obtenues en remplaçant  $q$  disques sur la sphère par de telles surfaces.*

En référence à cette classification, la CEP d'une variété 2D quelconque est obtenue par l'équation suivante :

$$\nu(S) = 2 - 2p - q - r, \quad (2.4)$$

où  $p$  est le nombre de poignées,  $q$  le nombre de surfaces d'Apéry et  $r$  le nombre de trous (ou de contours libres). On retrouve ainsi la CEP d'une sphère (2), d'un tore (0), etc.

Plus simplement, la CEP étant  $C$ -additive, et celle des objets simples connue (1 pour une boule, etc), on peut la calculer localement par addition en considérant des unions d'objets. Un autre point important est que la CEP d'une surface associée à son caractère orientable ou non-orientable permet de caractériser de manière unique cette surface du point de vue homotopique. Elle est donc indépendante du caractère continu ou discret de la surface en question, et donc d'une importance capitale en analyse d'image.

### 2.3.3. La caractéristique d'Euler-Poincaré dans l'espace discret

La référence à Euler dans la CEP vient de sa célèbre formule liant le nombre de faces, lignes et points dans les polyèdres convexes :

$$F - E + V = 2, \quad (2.5)$$

où  $F$  est le nombre de faces,  $E$  le nombre de lignes etc  $V$  le nombre de points. Or la CEP d'un polyèdre convexe est précisément égale à 2, et en général, pour tout polyèdre  $X$ , on trouve l'expression de sa CEP par l'équation suivante :

$$\nu(X) = F - E + V. \quad (2.6)$$

Cette équation est purement discrète. Or tout objet discrétisé dans la grille régulière carrée (en 2D) ou cubique (en 3D) se réduit respectivement à une surface ou un volume dont la frontière est un polygone ou un polyèdre respectivement. Cette observation permet de calculer la CEP de tout objet discrétisé en comptant certains points particuliers.

#### En dimension 1

Sur une grille régulière en 1 dimension, si  $X$  est un ensemble borné sur  $\mathbb{Z}$ , en reprenant la terminologie des graphes, alors

$$\nu^1(X) = N(\text{sommets}) - N(\text{arêtes}) = N(\bullet) - N(-) \quad (2.7)$$

On retrouve bien le fait que la CEP d'un intervalle fermé borné est 1, par exemple (voir figure 2.2).

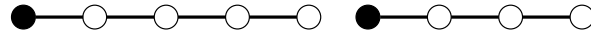


Figure 2.2. CEP en 1D d'un ensemble discret. Chaque segment a pour CEP=1.

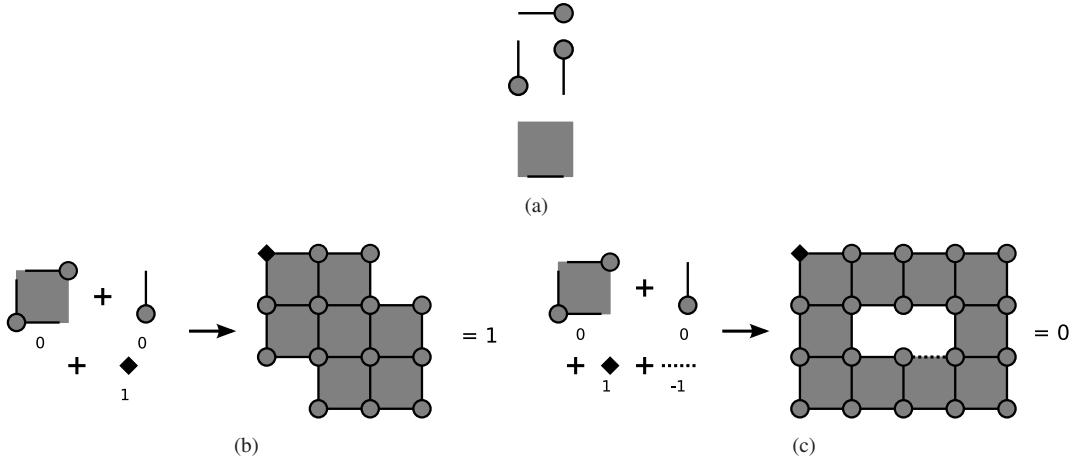


Figure 2.3. Calcul de la CEP pour un ensemble 2D discret. En (a), on montre des éléments de CEP=0 qui permettent la construction par union de la plus grande partie des ensembles discrets en (b) et (c). Pour la figure (b), il faut ajouter en plus un point unique (diamant noir en haut a gauche), de CEP=1, l'ensemble total ayant une CEP=1. En (c) nous avons un ensemble avec un grain unique et un pore. Pour le constituer, il faut en plus une arête unique (en pointillé), de CEP=-1. La CEP totale de cet ensemble vaut donc 1-1 = 0.

En dimension 2

Sur une grille carrée régulière, si  $X$  est un ensemble borné dans  $\mathbb{Z}^2$ , alors on retombe sur la formule d'Euler :

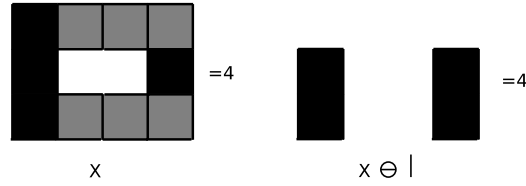
$$\begin{aligned} \nu^2(X) &= N(\text{sommets}) - N(\text{arêtes}) + N(\text{faces}) \\ &= N(\bullet) - N(-) - N(|) + N(\square) \end{aligned} \tag{2.8}$$

En 2D cette CEP se calcule relativement facilement, par dénombrement. Voir par exemple la figure 2.3. Dans cette figure, la CEP des ensembles discrets est rigoureusement la même que dans le continu, et correspond à la règle établie précédemment, à savoir  $\nu^2(X) = \text{nombre de grains moins nombre de pores}$ . Néanmoins, on doit considérer chaque pixel comme un complexe (muni de points, arêtes et d'une face), ce qui n'est pas souhaitable.

Par analogie avec la définition par récurrence dans le continu, nous pouvons également calculer la CEP discrète en 2D de la manière suivante. Appelons  $\overline{\nu^1}(A)$  la somme des CEP en 1D des sections horizontale de tout ensemble  $A$ , et notons  $A \ominus |$  l'érosion de Minkowski de  $A$  par un segment vertical unitaire, alors on a l'équation suivante :

$$\nu^2(X) = \overline{\nu^1}(X) - \overline{\nu^1}(X \ominus |), \tag{2.9}$$

Cette équation est facile à mettre en œuvre dans la pratique. Elle permet d'obtenir la CEP à partir des pixels plutôt que des complexes autour de ces mêmes entités. La figure 2.4 explique son fonctionnement. On constate que la formule 2.9 accumule 1 par grain et -1 par pore, ce qui correspond bien à la CEP attendue provenant du continu.



**Figure 2.4.** Calcul de la CEP 2D discrète par érosion de la même figure que celle de 2.3(c). On a représenté en noir les pixels qui sont des débuts de section horizontale, dans la figure originale à gauche et dans la figure érodée à droite. La différence vaut 0, ce qui est bien la CEP de la figure initiale

### En dimension 3

De même qu'en 2D, on peut définir la CEP d'un ensemble  $X$  borné dans  $\mathbb{Z}^3$  :

$$\nu^3(X) = N(\text{sommets}) - N(\text{arêtes}) + N(\text{faces}) - N(\text{volumes}) \quad (2.10)$$

En grille cubique, il y a 3 types d'arêtes et de faces. Pour éviter de faire le calcul dans les complexes, et de même qu'en 2D on peut également l'exprimer récursivement en utilisant  $\nu^2$  :

$$\nu^3(X) = \overline{\nu^2}(X) - \overline{\nu^2}(X \ominus |). \quad (2.11)$$

Dans cette équation,  $\overline{\nu^2}(A)$  est la somme des CEP dimensionnelles des sections horizontales de  $A$  (calculées sur les pixels), et  $X \ominus |$  est l'érosion de  $X$  par un segment vertical unitaire.

De manière pratique les équations 2.9 et 2.11 sont faciles à mettre en œuvre, car il s'agit de simples comptages de configurations locales.

## 2.4. Stéréologie et fonctionnelles de Minkowski

La stéréologie au sens strict est l'étude des relations entre les mesures sur un objet physique complet (donc en 3 dimensions) et celles sur des échantillonnages de dimensions inférieures, tels des sections par des plans, des droites ou des points. Ce nom a été donné à cette discipline dans les années 1960 par DeHoff [DEH 62], Weibel [WEI 63], Haug [HAU 63] et Underwood [UND 70]. Cependant certaines des résultats et des techniques formalisées par la stéréologie existent depuis longtemps, certaines depuis au moins le XVI<sup>e</sup> siècle [MAL 89] et plus récemment depuis les débuts de la microscopie quantitative [DEL 48]. Par extension, rentrent dans le champs de la stéréologie toute les techniques de mesures physiques réalisée manuellement ou automatiquement, par échantillonnage sans biais.

### 2.4.1. Fonctionnelles de Minkowski

Nous allons maintenant nous intéresser aux moyennes de la CEP selon les translation et les rotations dans les sous-espaces de  $\mathbb{R}^d$ . Par exemple, nous pouvons exprimer le calcul de  $\nu^0$  sur tous les points de l'espace  $\mathbb{R}^2$  ou  $\mathbb{Z}^2$ . Cet exemple est particulièrement parlant, puisque dans le discret comme dans le continu,  $\nu^0(X \in \mathcal{K}(\mathbb{R}^2))$  vaut en chaque point  $p$  1 si  $p \in X$  et 0 sinon. En faisant la somme de ces fonctionnelles, on en obtient une autre

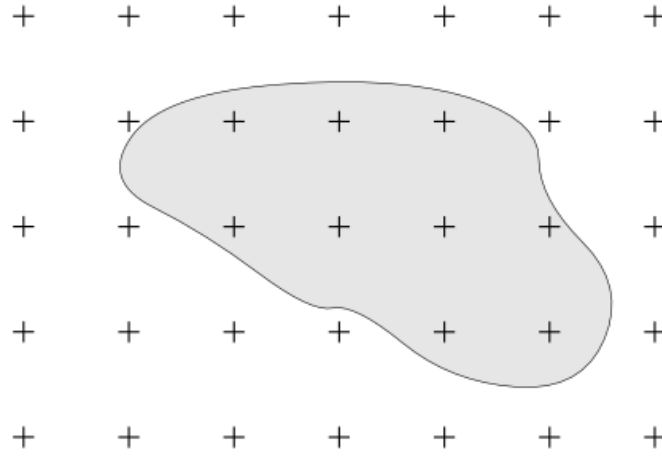


Figure 2.5. Estimation de l'aire par  $\nu^0$ .

également intéressante, qui n'est autre que l'aire de  $X$  (voir figure 2.5)! Notons que dans  $\mathbb{R}^2$ , l'aire de  $X$  est clairement de degré 2.

D'une manière générale, considérons un ensemble  $X$  de l'anneau convexe  $\mathcal{R}(\mathbb{R}^d)$ , ainsi que la famille de ses sections  $X \cap \Pi_k$  par les hyper-plans  $\Pi_{k,\omega}$  de dimension  $k$  ( $0 \leq k \leq n$ ), et de directions  $\omega$ . Prenons l'intégrale des CEP quand à  $\omega$  fixé,  $\Pi_k$  varie par translations, puis la moyenne par rotations de cette somme. On obtient une fonctionnelle de dimension  $k$ , que sa construction rend invariante par déplacements, homogène de degré  $n - k$ ,  $c$ -additive, et continue sur les compact convexes. Lorsque  $k$  varie de 0 à  $n$ , on arrive donc  $n + 1$  fonctionnelles qui vérifient les contraintes que nous recherchons. Mieux encore, ces contraintes les caractérisent, dans le sens du théorème suivant [HAD 57].

THÉORÈME 2.6.— Toute fonctionnelle sur l'anneau convexe de  $\mathbb{R}^d$ , invariante par déplacements,  $C$ -additive, et continue pour les compacts convexes est combinaison linéaire des  $n + 1$  fonctionnelles de Minkowski de  $\mathbb{R}^d$ .

Lorsque on impose la monotonie conditionnelle, les coefficients de la combinaison linéaire sont tous positifs. Quand on impose en plus l'homogénéité, on retrouve à un multiple près les fonctionnelles de Minkowski. En particulier, toujours dans  $\mathbb{R}^d$  :

- la première fonctionnelle (de degré  $n$ ) est l'intégrale de la mesure de Lebesgue sur  $X$ . Elle est croissante et semi-continue supérieurement ;
- la seconde (de degré  $n - 1$ ) est l'intégrale de la mesure de superficie de la frontière  $\partial X$  de  $X$ . En 2D c'est le périmètre ;
- la pénultième (de degré 1) est appelée *norme* ou *diamètre moyen*. Elle commute avec l'addition de Minkowski (la dilatation) :

$$M(\lambda X \oplus \mu Y) = \lambda M(X) + \mu M(Y)$$

- la dernière a été déjà décrite en détail, c'est la caractéristique d'Euler-Poincaré.

Pour les cas pratiques où  $n \in \{1, 2, 3\}$ , les fonctionnelles de Minkowski ont des interprétations géométriques qui en font les bases de la mesure en analyse d'images.

2.4.1.1. *En dimension 1*

Dans  $\mathbb{R}^1$ , la première fonctionnelle se réduit à la longueur  $L(X)$  de l'ensemble  $X$ . La seconde et dernière est le nombre  $\nu^0(X)$  de segments qui forment  $X$ .

2.4.1.2. *En dimension 2*

Dans  $\mathbb{R}^2$ , les trois fonctionnelles sont l'aire  $A$ , le périmètre  $U$  et la CEP  $\nu^2$ , avec la relation :

$$U(X) = \int_{\pi} d\alpha \int_{\mathbb{R}} \nu^1[X \cap \Delta(x, a)] dx,$$

Cette dernière relation interprète le périmètre comme une somme d'intercepts pris dans toutes les directions.

De plus, lorsque  $X$  est compact et convexe, le périmètre est relié aux projections de  $X$  sur des lignes droites  $\Delta_{\alpha}$  par la relation suivante :

$$U(X) = \int_{\pi} L[X | \Delta_{\alpha}] d\alpha.$$

2.4.1.3. *En dimension 3*

Dans  $\mathbb{R}^3$ , les quatre fonctionnelles sont le volume  $V$ , l'aire surfacique  $S$ , le diamètre moyen ou *norme*  $M$  et la CEP  $\nu^3$ . Nous avons les trois expressions suivantes :

$$\text{volume} \quad v(X) = \int_{\mathbb{R}^3} \nu^0(X \cap \{x\}) dx \quad (2.12)$$

$$\text{aire surfacique} \quad \frac{1}{4}s(X) = \frac{1}{4\pi} \int_{4\pi} d\omega \int_{\mathbb{R}^2} \nu^1[X \cap \Delta(x, \omega)] dx \quad (2.13)$$

$$\text{diamètre moyen} \quad d(X) = \frac{1}{4\pi} \int_{4\pi} d\omega \int_{\mathbb{R}} \nu^2[X \cap \Pi(x, \omega)] dx \quad (2.14)$$

Les trois relations des équation (2.12) à (2.14) permettent de donner une interprétation euclidienne à des données discrètes. Par exemple, l'équation (2.12) devient, en discrétisant :

$$v^*(X) = (\text{Nombre de voxels de } X) \times v_0$$

où  $v_0 = a^3$  (pour la grille cubique) or  $a^{3/4}$  (pour le réseau FCC) or  $a^{3/2}$  (réseau CC). De même, l'équation (2.13) s'écrit :

$$s^*(X) = (\text{nombre moyen d'intercepts}) \times 2a^2\sqrt{2}$$

La moyenne est prise sur les six directions de l'équation (2.13) dans la grille cubique. L'estimation  $s^*(X)$  correspond à une surface euclidienne  $s(X)$ , et donc diffère de l'aire qu'on obtiendrait en sommant l'aire des faces de l'ensemble discret  $X$ . Par exemple, une face d'épaisseur nulle compte double.

On note également le point suivant : lorsque la surface 3D  $\partial X$  admet toujours deux courbures  $C_1$  and  $C_2$ , la norme correspond à l'intégrale de la courbure moyenne, c'est-à-dire :

$$2M(X) = \int_{\partial X} (C_1 + C_2) ds.$$



En particulier, lorsque  $X$  est convexe, son aire et sa norme peuvent être interprétés en termes de projections  $X | \Pi_\omega$  et  $X | \Delta_\omega$  sur les plans  $\Pi_\omega$  et les lignes  $\Delta_\omega$

$$\pi S(X) = 4\pi \int A(X | \Pi_\omega) d\omega$$

$$2M(X) = 4\pi \int L(X | \Delta_\omega) d\omega = (2/\pi) \int_{4\pi} U(X | \Pi_\omega) d\omega$$

## 2.5. Changement d'échelle et stationarité

Nous avons jusqu'ici considéré le cas des objets isolés et entièrement contenus dans le champ de mesure. Intéressons nous maintenant à une structure qui s'étend dans un domaine spatial  $Y$  compact, certes, mais bien plus vaste que ce que notre petite fenêtre d'étude  $Z$  permet de voir, sans qu'il soit même matériellement possible de recouvrir  $Y$  par une famille de masques de mesure. Le formalisme auquel nous sommes conduits est alors celui développé par G.Matheron dans [MAT 89] sous le nom de *représentations glissantes*. Elle consiste à engendrer la fonction aléatoire :

$$X(h) = Y \cap Z_0(\underline{x} + h) \quad h \in Z_0$$

où  $Z_0$  désigne le champ de mesure centré à l'origine, et où  $\underline{x}$  est le point aléatoire obtenu en munissant  $Y \oplus Z$  de la loi de probabilité uniforme. Nous avons alors le théorème suivant ([MAT 89], p.122) :

THÉORÈME 2.7.– *l'ensemble fermé aléatoire  $X$  est stationnaire dans  $Z_0$ .*

Cette stationarité ne porte, bien sûr, que sur les moments dont le support est inclus dans  $Z_0$ . Ainsi, par le truchement de la probabilité uniforme, l'invariance selon les translations (mais pas les rotations...) se mue en stationarité, ce qui permet de récupérer les fonctionnelles de Minkowski dans ce nouveau cadre. Les stéréologistes notent  $V_V(X)$  le volume spécifique de  $X$  (c'est-à-dire par unité de volume de l'espace), par  $S_V(X)$  la surface spécifique, et par  $N_A$  et  $N_L$  les CEP spécifiques dans  $\mathbb{R}^2$  et  $\mathbb{R}^1$  (ou le cas échéant leurs moyennes de rotations). Il vient alors :

$$V_V = A_A = L_L$$

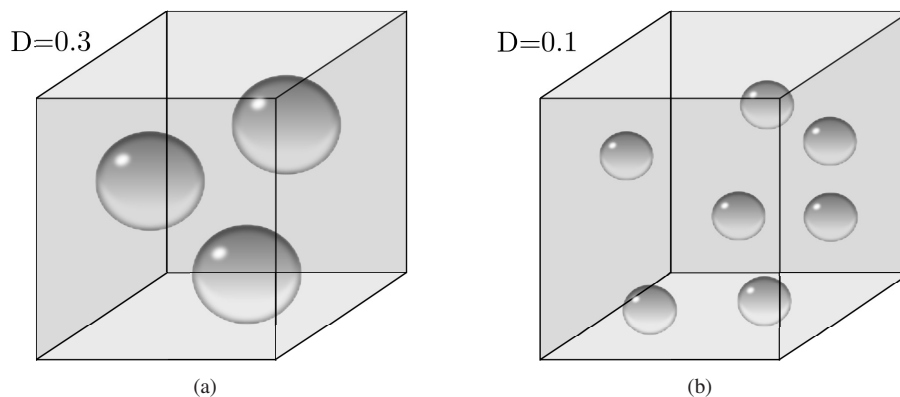
$$S_V = 4N_L = \frac{4}{\pi}U_A$$

$$M_V = 2\pi N_A$$

D'une manière générale, toutes les mesures que l'on est amené à présenter de façon probabiliste, comme la covariance ou les granulométries, se rattachent à l'objet déterministe sous jacent par représentation glissante.

## 2.6. Individus et granulométries

Le troisième volet des choix que nous nous sommes imposés a trait aux *individus*, c'est-à-dire à certaines zones du plan ou de l'espace obtenues par segmentation. Ces régions n'ont pas à être connexes, il suffit que l'on puisse les labelliser. Ce seront les villages en imagerie satellite, les mitoses en lame mince histologique,



**Figure 2.6.** Le nombre moyen des grains en sections horizontales vaut 0.9 dans le cas (a) et 0.7 dans le cas (b)

les silhouettes détectées par une caméra de surveillance, etc. Cela dit, les individus étant le plus souvent des composantes connexes, c'est sur elles que portent les exemples ci-dessous, et en particulier, le premier d'entre eux destiné à montrer la différence entre granulométrie en nombre et en mesure. Lorsque les deux cubes unité de la figure 2.6 sont balayés par des sections horizontales, elles contiennent en moyenne 0.9 particules dans le cas (a) et 0.7 dans le cas (b)<sup>2</sup>, car les sphères de  $\mathbb{R}^3$  apparaissent en section proportion du produit de leur diamètre par leur nombre (autrement dit, la CEP dans  $\mathbb{R}^3$  ne se déduit pas de celle dans  $\mathbb{R}^2$ ).

### 2.6.1. Comptages sans biais

Un des aspects les plus difficiles de la mesure est de réaliser qu'en général le comptage d'objets ne se déduit pas des fonctionnelle de Hadwiger :

- on ne peut pas compter n'importe comment tous les objets qu'on voit : on risque de générer des biais en faveur des objets les plus grands, les plus longs, etc ;
- pour résoudre le problème on peut associer un point particulier à chaque objet, car un point sans dimension n'est pas affecté par les effets de bord ;
- exemples : extrémités de fibres, bissecteur en 3D ;
- on doit parfois corriger un nombre obtenu en tenant compte de sa géométrie : son orientation par exemple ;
- parfois un biais est inévitable de part la nature de la technique d'imagerie (exemple des fibres en coupe), on doit alors l'identifier et le quantifier si possible.

Nous pouvons préciser la distinction entre mesure pour un ensemble et mesure pour ses individus. Nous allons nous concentrer sur les granulométries.

<sup>2</sup> Coupons en dix sections horizontales. Dans le 1er cas, chaque grain apparaît dans 3 des 10 sections, donc son nombre moyen est 0.3 ; comme ils sont trois, ça fait 0.9 au total. Dans le 2ème cas, chaque grain apparaît dans une des 10 sections seulement, donc une moyenne de 0.1, et au total, 0.7.

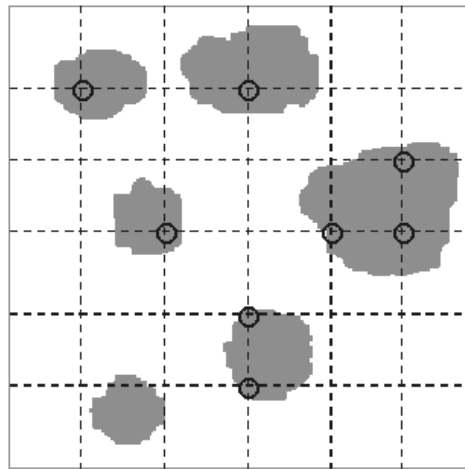


Figure 2.7. Exemple d'échantillonnage par grille régulière

### 2.6.2. Granulométries en nombre et en mesure

Comme dans le pâté d'alouettes, quand on parle de la statistique de tailles d'objets, il convient de préciser si l'on affecte une taille à chaque point de l'ensemble étudié, ou si l'on a segmenté des individus et que la taille porte sur chacun de ceux-ci. Dans le second cas, il faut encore préciser si l'on compte chaque individu :

- pour le même poids (par exemple, dans le cas de la formule sanguine), et c'est l'analyse en nombre ;
- ou en pourcentage de son volume (par exemple, dans un tamisage), et c'est l'analyse en mesure.

Chaque procédure expérimentale choisit implicitement entre ces deux possibilités. Par exemple, en échantillonnant l'espace selon une grille régulière, on pondère en mesure, à condition toutefois de compter chaque particule proportionnellement au nombre des noeuds de la grille qu'elle rencontre (voir figure 2.7).

Il en va de même lorsqu'on décide d'extraire le grain qui contient l'origine. En stéréologie, la pondération en surface sur les sections planes exprime une pondération en volume des objets 3D correspondants. Enfin les spectres granulométriques selon des ouvertures par adjonction sont toujours définis en termes de mesure, car il n'y a pas ici d'individus, mais une taille associée à chaque point  $x \in X$  (le rayon du plus grand disque contenant  $x$  et inclus dans  $X$ ).

On ne parvient pas en général, à passer des ouvertures par adjonction, par des cercles par exemple, d'une population d'individus à la distribution en nombre de leur surfaces. Il faudrait pour y arriver modéliser aussi les formes de ces objets, en considérant par exemple qu'il s'agit de sphères. Une démarche pertinente ici se déduit des érosions linéaires : nous y revenons un peu plus loin.

#### 2.6.2.1. Correction de Miles-Lantuejoul

A ces difficultés s'en ajoute une autre spécifique aux granulométries en nombre. Supposons que l'on segmente des individus  $X_i$  dans divers champs, et que l'on attribue à chaque individu une mesure quelconque globale  $a_i$ . Or, plus un individu est gros, plus il a des chances d'apparaître dans un champ, mais d'autre part s'il est très gros, il ne sera jamais vu en totalité. Comment dans ces conditions construire sans biais la granulométrie en nombre des  $a_i$  ?

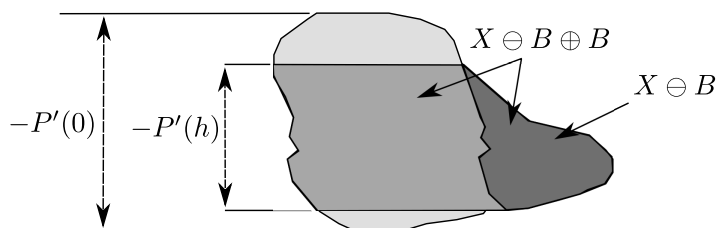


Figure 2.8. Interprétation géométrique de l'érosion et de l'ouverture linéaires.

Ce point est délicat, il a été résolu indépendamment par Ch. Lantuéjoul [LAN 80] et R.E. Miles [MIL 75], dont la solution porte leur noms. Elle est traitée au chapitre ??.

### 2.6.3. Granulométries linéaires

Lorsqu'on ne connaît rien sur les formes des individus, l'opération qui apporte le plus d'informations consiste à les étudier au moyen de leurs traversées linéaires. De fait, c'est paradoxalement l'une des érosions les plus simples, celle  $X \ominus B(h)$ , où  $B(h)$  est le segment de longueur  $h$ , qui fournit l'une des moissons de mesures les plus riches. Présentons la dans le cadre stationnaire de  $\mathbb{R}^3$ , en supposant pour simplifier que les granulométries sont isotropes. On note :

- $P(h)$  la proportion d'érodé (en longueur en surface ou en volume) selon le segment de longueur  $h$  ;
- $F(h)$  et  $G(h)$  les deux fonctions de répartition des cordes, en nombre et en mesure respectivement, de densités  $f(l)$  et  $g(l)$  ;
- $F(h^k)$  et  $\mathcal{M}(h^k)$  les moments en nombre et en mesure respectivement.

On voit facilement que :

$$1 - F(h) = \frac{P'(h)}{P'(0)} = \text{pourcentage des cordes } \geq h$$

$$1 - G(h) = \frac{P(h) - hP'(h)}{P(0)}$$

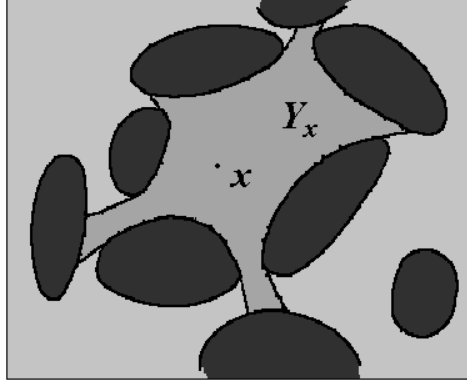
La figure 2.8, bien que dessinée dans le cadre non stationnaire, montre toutefois clairement d'où proviennent les différents facteurs de  $F(h)$  et  $G(h)$ .

Les moments en nombre et en mesure se déduisent des expressions précédentes. L'on a :

$$E[h] = \int_0^\infty [1 - F(h)] dh = -P(0)/P'(0)$$

et plus généralement :

$$E[h^d] = [n(n-1)/-P'(0)] \int_0^\infty h^{(n-2)} P(h) dh,$$



**Figure 2.9.** Étoile  $Y_x$  des pores au point  $x$ .

et pour les moments en mesure  $\mathcal{M}[h^d]$ .

$$\mathcal{M}[h^d] = [n(n+1)/P(0)] \int_0^\infty h^{n-1} P(h) dh.$$

Les interprétations géométriques sont instructives. Commençons par les moments en mesure, en assignant, à chaque point  $x$  des pores, la zone  $Y_x$  accessible en vision directe à partir de  $x$  (voir figure 2.9). Cette sorte de « pieuvre » varie en chaque point et sa moyenne  $E[\text{Mes } Y_x]$ , nommée *étoile*, représente la taille moyenne des cavités poreuses. Elle se déduit des moments en mesure par les relations :

$$E[\text{Mes } Y_x] = \frac{\pi}{3} \mathcal{M}[h^2] \quad \text{dans } \mathbb{R}^2$$

$$E[\text{Mes } Y_x] = \frac{\pi}{3} \mathcal{M}[h^3] \quad \text{dans } \mathbb{R}^3$$

où les traversées  $h$  sont prises dans les pores. Si l'on se place dans les grains et que ceux-ci sont convexes, l'étoile des grains n'est rien d'autre que leur aire moyenne (dans  $\mathbb{R}^2$ ) ou leur volume moyen (dans  $\mathbb{R}^3$ ), comptés en mesure.

Le comportement à l'origine de  $P(h)$  donne lieu lui aussi à des interprétations stéréologiques, mais différentes. On a :

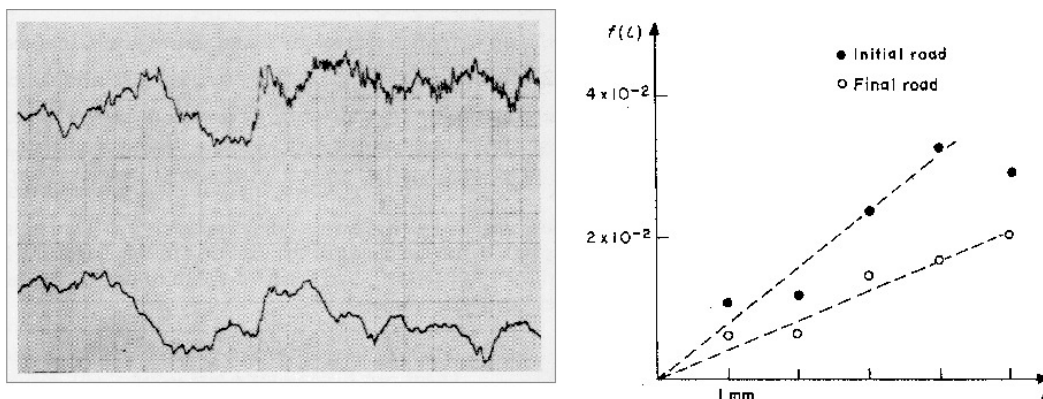
$$-\frac{\pi}{2} P'(0) = \text{périmètre spécifique dans } \mathbb{R}^2$$

$$-4P'(0) = \text{surface spécifique dans } \mathbb{R}^3.$$

Si l'on suppose maintenant que l'ensemble étudié est assez régulier pour admettre des courbures finies sur sa frontière, et que sa granulométrie en nombre  $F(h)$  a pour dérivée la densité  $f(h)$ , celle-ci présente alors un comportement linéaire au voisinage de l'origine :

$$f(r) = \frac{rE(C^2)}{8} \quad \text{dans } \mathbb{R}^2,$$

$$f(r) = r \frac{E(3C^2 - C')}{8} \quad \text{dans } \mathbb{R}^3,$$



**Figure 2.10.** Mesure de l'usure de la rugosité d'une chaussée, à partir de profils de sa surface.

où, dans  $\mathbb{R}^2$ ,  $C$  désigne le rayon de courbure, et, dans  $\mathbb{R}^3$ ,  $C$  et  $C'$  sont les courbures moyenne et totale. En particulier, si  $X$  est le sous-graphe (dans  $\mathbb{R}^3$ ) d'une fonction numérique de  $\mathbb{R}^2$ ,  $C'$  disparaît et il reste :

$$f(r) = \frac{3}{8}rE(C^2).$$

Cette dernière relation a été appliquée avec succès à la mesure de la rugosité d'une route, à partir de profils pris sur sa surface (voir figure 2.10). Deux séries de 1.000 et 10.000 passages de camions de 6,5 tonnes allant à 65 Km/h ont été simulées sur un cirque d'essais (Laboratoire Central des Ponts et Chaussées). En mesurant  $f(h)$  dans six directions et en appliquant la formule ci-dessus, on trouve :

$$\begin{aligned} E(C^2) &= 2,210^{-2}mm^{-2} && \text{avant passage,} \\ E(C^2) &= 10^{-2}mm^{-2} && \text{après passage.} \end{aligned}$$

Il est remarquable qu'un modèle aussi régulier (courbures en tout point) conduise cependant à des résultats aussi pertinents.

## 2.7. Extension aux niveaux de gris

Les mesures qui précèdent s'appliquent aux images binaires, qui sont essentiellement des images segmentées. C'est un cas courant mais non universel. En fait les fonctionnelles de Minkowski s'étendent aux fonctions numériques en niveaux de gris, par exemple en prenant la classe  $\mathcal{F}_{\mathcal{R}}$  des fonctions  $f$  de  $\mathbb{R}^2$  (ou  $\mathbb{R}^d$ ) dans  $[-\infty, +\infty]$  dont toutes les sections sont des éléments de l'anneau convexe. Les fonctionnelles  $W$  deviennent des applications de  $\mathcal{F}_{\mathcal{R}}$  vers  $[-\infty, +\infty]$ , et les concepts décrits plus haut s'étendent aisément, par exemple la  $C$ -additivité devient :

$$W(f \vee g) + W(f \wedge g) = W(f) + W(g) \quad f, g \in \mathcal{F}_{\mathcal{R}};$$

Cependant intégrer l'axe  $T$  des niveaux de gris implique des précautions du fait de l'hétérogénéité potentielle des dimensions spatiales et d'intensité. La notion utile ici est celle de *dimensionnalité* [SOI 92]. Une

fonctionnelle  $W$  est dimensionnelle si :

$$W[\lambda.f(\mu.x)] = \lambda^k . \mu^p . W[f(x)] \quad x \in \mathbb{R}^d, f \in \mathcal{F}_{\mathcal{R}}$$

avec  $k$  et  $p$  entiers positifs les dimensions de  $W$ . On dérive ainsi certaines mesures facilement.

### 2.7.1. Aires et volumes

En sommant les sections, on obtient le volume  $V(f)$  (potentiellement infini) et l'histogramme cumulé  $G_f(t)$  :

$$V(f) = \int_{\mathbb{R}^2} f(x)dx = \int_T A[X_t(f)]dt$$

$$G_f(t) = \int_{-\infty}^t A[X_t(f)]dt / V(f)$$

### 2.7.2. Gradient et périmètre

En étendant la formule de Steiner à l'anneau convexe, le périmètre est égal à la dérivée à l'origine de la dilatation par un disque :

$$r \rightarrow 0 \Rightarrow [A(X_t \oplus rB) - A(X_t)] \rightarrow rU(X_t)$$

Dans l'anneau convexe, cette limite est bornée supérieurement par la somme finie  $\sum [A(Y_i(t) \oplus rB) - A(Y_i(t))]$ , on peut donc écrire :

$$\int_t U[X_t(f)]dt = \lim_{r \rightarrow 0} [(V(f \oplus rB) - V(f))/r]$$

Le membre de droite est le gradient morphologique (dit de Beucher), qui correspond donc géométriquement à la somme des périmètres de toutes les sections.

$$\int_t U[X_t(f)]dt = \int_{\mathbb{R}^2} g^+(x)dx$$

### 2.7.3. Caractéristique d'Euler-Poincaré

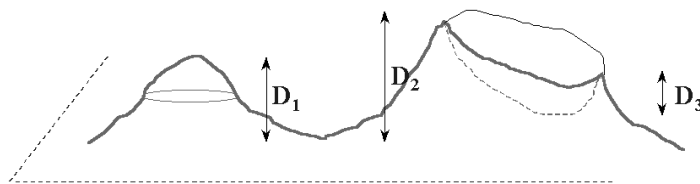
La somme des CEP sur toutes les sections de  $f$  mesure l'intégrale des différences de hauteur en  $f$ . Pour la figure 2.11, on obtient :

$$\int_t \nu[X_t(f)]dt = D1 + D2 - D3$$

### 2.7.4. Un contre-exemple : la longueur d'une courbe

Alors que le volume, l'histogramme cumulé, le gradient et la CEP sont tous dimensionnels, ce n'est pas le cas par exemple pour la longueur d'une courbe sur la variété  $\mathbb{R}^1 \otimes T$ , telle que donnée par exemple par :

$$l = \int [1 + [f'(x)]^2]dx$$



**Figure 2.11.** CEP pour les fonctions numériques.

En effet, si on multiplie les valeurs d'une image par 2, la longueur sur les zones plates ne change pas mais les autres sont augmentées proportionnellement au gradient. C'est pourtant cette longueur qu'optimisent de nombreux algorithmes de segmentation.

A condition de maîtriser les aspects de dimensionnalité, l'analyse d'images en niveaux de gris ouvre la possibilité d'effectuer des mesures sur les images sans forcément nécessiter une segmentation.

## 2.8. Conclusion

Dans ce chapitre nous avons vu les principes généraux permettant de réaliser certains types de mesures, dites *stéréologiques* dans des images, avec des garanties sur leur propriétés. Nous avons vu comment associer ces mesures à des objets binaires, suivant une segmentation, et également comment obtenir certaines informations (dites *dimensionnelles*) sur des images à niveaux de gris, sans passer par une étape de segmentation.

Néanmoins, tous les problèmes inhérents à la mesure en champ fini ne sont pas traités par cette approche. On doit également s'intéresser au problème de l'*échantillonnage*, à la fois en général, et dans le contexte d'une vision géométrique partielle, ainsi qu'au problème des mesures qui ne sont pas stéréologiques. Parmi celles-ci, on peut noter en particulier le problème du comptage. En effet, comme vu plus haut, la fonctionnelle de comptage n'est pas  $C$ -additive. Le « simple » comptage d'objets est souvent une étape essentielle dans beaucoup de chaînes d'analyse d'image, or c'est une étape difficile et semée d'embûches.

Le chapitre suivant apporte certaines réponses à ce sujet. On pourra aussi consulter avec profit le livre de A. Baddeley [BAD 05], qui présente la stéréologie sous l'angle statistique, y compris les méthodes classiques pour réaliser des comptages sans biais en 2D et 3D. Dans la suite du livre, le chapitre 3 évoque de nouveau les granulométries. Enfin, la dernière partie du présent livre présente toute une série d'applications qui sont autant d'illustrations des principes de mesures généraux en analyse d'image.



## Chapitre 3

# Distances, granulométries, squelette

Dans ce chapitre, nous présentons une série de concepts et d'opérateurs basés sur la notion de distance. Comme souvent en morphologie, on peut présenter un concept de plusieurs manières équivalentes et complémentaires. Ici, la problématique est de présenter des méthodes de caractérisation d'ensembles de points, fondées sur des considérations métriques, géométriques et topologiques : en particulier la notion de squelette. Cette notion est importante en reconnaissance des formes, et trouve de nombreuses applications dans la pratique.

### 3.1. Squelettes

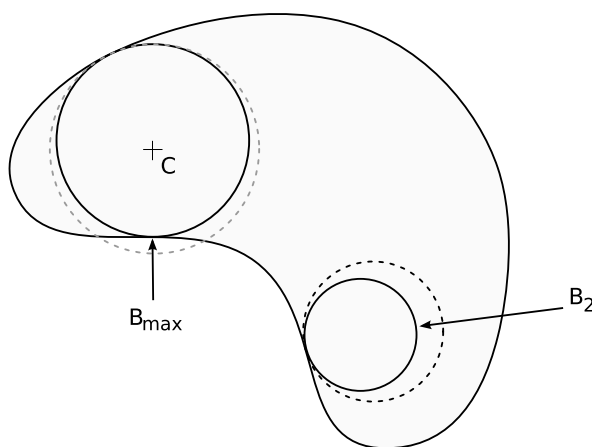
La notion de *squelette*, que nous allons définir de plusieurs façons, est très utile à de nombreux niveaux. Intuitivement, le squelette d'un objet (ensemble de points)  $X$  représente l'ensemble des lignes « médianes », c'est-à-dire situées à égale distance de deux parties —au moins— de la frontière de  $X$ . Historiquement, une définition du concept général a été proposée par Listing en 1861 [LIS 61], sous le nom de « diagramme cyclomatique », qui résulte de la contraction dans le plan euclidien d'une courbe fermée, sans modification de topologie. Cette notion est essentiellement topologique au sens de l'homologie.

Une définition plus précise et géométrique cette fois, celle de l'*axe médian* est due à Blum, cent ans plus tard [BLU 61, BLU 67]. Elle fait appel à la notion de boule maximale.

#### 3.1.1. Boules maximales

Les boules maximales sont un concept géométrique simple qui va nous permettre de définir le squelette d'un ensemble de points.

**DÉFINITION 3.1 (Boule).**— La boule  $B_r(x)$ , de rayon  $r$  et de centre  $x$ , est l'ensemble des points  $y$  de  $\mathbb{R}^n$  tels que  $d(x, y) < r$ , où  $d$  désigne une distance (par exemple, la distance euclidienne ou une distance discrète usuelle).



**Figure 3.1.** Illustration d'une boule maximale dans le cas euclidien. La boule  $B_{max}$  est maximale car aucune autre boule ne peut la contenir et être simultanément incluse dans l'ensemble. En revanche la boule  $B_2$  n'est pas maximale

**DÉFINITION 3.2 (Boule maximale).**— Une boule maximale  $B_r^{max}$  de rayon  $r$  pour un ensemble  $X$  est une boule incluse dans  $X$  telle que  $\forall s > r$ , si  $B_r^{max} \subset B_s$ , alors  $B_s \not\subset X$ .

Cette notion est illustrée par la figure 3.1. On constate sur cette figure avec un ensemble à la frontière régulière, que la boule maximale  $B_{max}$  est tangente à la frontière de l'ensemble en deux points distincts. C'est un cas courant mais ce n'est pas le cas général. Le point  $C$  dans cette figure est le centre de  $B_{max}$ . Il est clairement situé à la même distance des deux points tangents de  $B_{max}$  à la frontière de l'ensemble.

### 3.1.2. Fronts de feu

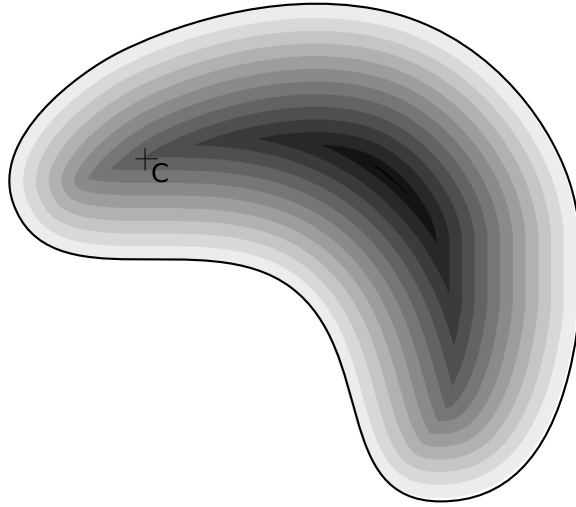
Quasi simultanément à la définition de Blum sont apparues celle de Calabi et Hartnett [CAL 68], ainsi que celle de Montanari [MON 68], qui font appel au concept de *front de feu*.

**DÉFINITION 3.3 (Front de feu et fonction d'extinction).**— Supposons qu'un feu soit allumé sur la frontière de  $X$  et se propage vers l'intérieur de  $X$  à vitesse constante et isotrope, et qu'un point éteint ne se rallume jamais, alors les lieux des points de  $X$  où plusieurs fronts de feu différents se rencontrent s'appellent les points d'extinction. La fonction qui à chaque point d'extinction associe le temps auquel le point s'est éteint, équivalent à la distance à la frontière de l'ensemble, est appelée fonction d'extinction (quench function en anglais).

Cette définition est conceptuellement utile, car elle permet de proposer des algorithmes de construction du squelette qui s'appuient sur l'idée de « rogner » la frontière de l'objet. De plus on a la propriété suivante :

**PROPRIÉTÉ 3.4.**— Pour un ensemble compact  $X$  suffisamment régulier<sup>1</sup> dans l'espace euclidien, les centres de boules maximales de  $X$  sont ses points d'extinction et réciproquement.

1. Ici on admettra qu'il faille pouvoir définir la notion de disque tangent à la frontière  $\partial X$  de  $X$ . On pourra prendre comme condition de régularité que  $\partial X$  soit partout doublement dérivable.



**Figure 3.2.** Illustration du concept des fronts de feu : le feu se propage de la frontière de l'objet vers l'intérieur, de manière isotrope. Ici le point  $C$  est un point d'extinction, et un centre de boule maximale

Il n'est pas immédiatement évident que les deux définitions soient équivalentes, mais cette équivalence est relativement facile à montrer. Intuitivement, le lieu de rencontre  $C$  de plusieurs fronts de feu différents, du fait de la propagation à vitesse constante, est situé à la même distance de  $n$  points distincts  $A_1, A_2, \dots, A_n$  de la frontière de  $X$ . Du fait de la propagation isotrope, les segments  $[CA_i]$  sont tous de même longueur, normaux à la frontière de  $X$  et sont des rayons d'une boule  $B_{C,A_i}$  centrée en  $C$ . D'autre part, cette boule est nécessairement incluse dans  $X$  sinon un autre point distinct de la frontière de  $X$  aurait éteint  $C$  avant les fronts issus des points  $A_i$ . Finalement, toute boule contenant  $B_{C,A_i}$  et incluse dans  $X$  doit aussi nécessairement être normale à  $\partial X$  en tous les points  $A_i$ . Elle est donc forcément centrée en  $C$ , et de rayon  $[CA_i]$ , donc la boule  $B_{C,A_i}$  est maximale.

Réciproquement, soit  $B_C$  une boule maximale de  $X$ , centrée en  $C$ . Puisqu'elle est maximale, elle intersecte  $\partial X$  en au moins deux points distincts  $A$  et  $B$ . En effet, supposons qu'elle n'intersecte  $\partial X$  en aucun point, alors par définition on peut inclure un cercle strictement dans  $X \setminus B_C$ , et donc  $B_C$  n'est pas maximale. C'est toujours vrai si  $B_C$  intersecte  $\partial X$  en un seul point  $A$ , avec la restriction que toutes les courbes sont tangentes en  $A$  : la frontière de  $X$ , celle de  $B_C$  et celle du cercle. Finalement, les fronts de feux issus de  $A$  et  $B$  se rencontrent en  $C$ .

La définition des fronts de feu est imagée mais assez peu rigoureuse et liée au continu. Il peut être utile de considérer les propriétés du squelette dans ce domaine.

### 3.1.3. Propriétés du squelette dans le continu

Avec les précautions d'usage, les définitions précédentes s'appliquent dans le domaine du continu, c'est à dire des variétés de dimension arbitraire, munies d'une métrique Riemannienne. On se contentera ici de  $\mathbb{R}^n$  munie de la métrique euclidienne. On considérera comme précédemment l'ensemble  $X$  à squelettiser comme compact avec une frontière partout doublement dérivable. Pour éviter certains cas limites, on considérera que l'intérieur de  $X$ , noté  $\overset{\circ}{X}$  est connexe.

Dans ce domaine, les deux définitions données sont équivalentes, et possèdent les propriétés suivantes :

PROPRIÉTÉ 3.5.– *Propriété du squelette dans le continu :*

- *non-continuité* : une déformation continue d'un ensemble peut provoquer des changements non-continus dans le squelette. Par exemple, le squelette euclidien d'un disque est un point (son centre), mais un changement continu à sa frontière peut se traduire par l'apparition non continue d'une branche ;
- *homotopie* : le squelette est homotope à la forme de départ ;
- *négligeabilité* : la mesure du squelette au sens de Lebesgue est nulle. Cela se traduit par le fait que le squelette ne contient aucun voisinage ;
- *invariance par transformation isométrique* : appliquer une telle transformation (par exemple une rotation) avant ou après le calcul du squelette conduit au même résultat ;
- *liens avec la fonction distance* :
  - la fonction associant aux points du squelette leur distance à la frontière de l'objet (la fonction d'extinction) permet de reconstruire parfaitement la forme d'origine ;
  - le squelette est centré dans l'objet, au sens où tout point du squelette se trouve à égale distance d'au moins deux points distincts de la frontière de l'objet ;
  - pour tout point appartenant au squelette, la ligne de plus grande pente ascendante, pour la fonction distance, partant de ce point est également contenue dans le squelette ;
  - les centres de boules maximales sont des points particuliers de la fonction distance : les points d'arêtes. En ces points, la ligne de plus grande pente est strictement plus petite que 1.

Ces propriétés sont explorées en détail dans [MAT 88a, MAT 88b]. Beaucoup sont démontrées dans [RIV 87]. La non-continuité du squelette se traduit dans les faits par une grande sensibilité au bruit. Un seul point ajouté à la frontière d'un objet peut induire la formation d'une nouvelle branche du squelette. Un petit changement arbitraire de  $X$  peut impliquer un changement de topologie et des changements radicaux dans l'apparence du squelette. En général on cherche à régulariser ces effets par filtrage, soit par celui de la forme de  $X$ , soit du squelette lui-même [VIN 91a, ATT 95].

L'homotopie du squelette n'est pas garantie par tous les algorithmes de squelettisation discrets. La plupart de ceux-ci cependant tentent de garantir cette propriété importante, sur laquelle nous reviendrons dans la section 3.6. La préservation de l'homotopie du squelette est une propriété utilisée dans la reconnaissance de formes.

Le fait que le squelette soit négligeable dans le continu est parfois traduit par le besoin qu'il soit fin, c'est à dire qu'il puisse être effacé par une érosion par la boule élémentaire de la connectivité de la grille utilisée pour le calculer. Certains algorithmes tentent de produire un squelette d'épaisseur un pixel partout, ce qui n'est pas nécessairement compatible avec d'autres propriétés du squelette, telle le centrage.

### 3.2. Squelettes dans les espaces discrets

On ne peut pas calculer, et encore moins utiliser de manière pratique le squelette continu, mais on va chercher, dans le domaine discret, à reproduire tout ou partie de ses caractéristiques.

Tout d'abord, notons qu'il existe des algorithmes de squelettisation qui partent de la définition du continu, et formulent le problème dans ce domaine. Par exemple, on peut modéliser un feu de forêt au moyen d'une évolution de contour par Equation aux Dérivées Partielles (EDP), et trouver les points du squelette par détection des points de chocs, qui sont les points de l'espace où seule une solution faible de l'EDP existe, ou

encore en maintenant les Centre de Boules Maximales (CBM) comme points d'ancrages pour l'évolution du contour [LEY 92, SID 99, TOR 03]. *In fine*, malgré leur élégance théorique, ces méthodes trouvent toujours un squelette discret du fait que les EDP utilisées pour le calcul sont discrétisées. Il est difficile de prouver quoi que ce soit à propos des résultats obtenus du fait que les solutions ne sont qu'approchées. Finalement les choses peuvent se compliquer encore en 3D et plus.

On ne peut pas reproduire dans le domaine discret toutes les propriétés des squelettes continus, mais on peut en revanche choisir tel ou tel compromis en fonction des besoins.

On va chercher à calculer, pour un objet discret  $X$ , un objet  $S(X)$  :

- homotope à l'ensemble  $X$  ;
- mince : c'est-à-dire d'un nombre minimal de pixel ou de voxel de large, ou encore un objet effaçable par une érosion élémentaire ;
- qui contienne les centres de boules maximales discrètes ;
- centré dans l'objet  $X$ , c'est-à-dire que tout point de  $S(X)$  doit se situer à égale distance, au sens de la trame, de deux points distincts de la frontière de  $X$ .

La dernière propriété est en général impossible à satisfaire strictement. Par exemple en 8-connexité dans le plan  $\mathbb{Z}^2$ , pour un ensemble  $X$  constitué d'un carré  $10 \times 10$ , les centres de boules maximales, qui sont dans le cadre de cette trame des carrés de taille  $n \times n$  avec  $n$  impair, ne sont pas situés à égale distance de tous les points du bord de  $X$ .

Dans les sections suivantes, nous allons détailler deux approches permettant de définir le squelette d'une forme dans un espace discret. La première approche, décrite dans la section 3.3, repose sur la notion de granulométrie, familière en morphologie mathématique. Elle a l'avantage d'induire des généralisations intéressantes reposant par exemple sur des familles d'éléments structurants non symétriques ou non connexes. La seconde approche consiste à utiliser des distances discrètes pour définir les notions de boule et de boule maximale relativement à un objet donné (section 3.4). Cette dernière approche permet l'utilisation de la distance euclidienne exacte, qui assure pour le squelette une moindre sensibilité à l'orientation de l'objet dans l'espace, une complète invariance par rotation étant impossible à garantir.

### 3.3. Familles granulométriques et squelettes

#### 3.3.1. Famille granulométrique

Dans un premier temps, nous nous contenterons de définir une manière de caractériser la taille d'un ensemble de points. Nous verrons par la suite que cette caractérisation est riche de conséquences.

**DÉFINITION 3.6.**– *On définit d'abord la notion de famille granulométrique au moyen des étapes suivantes :*

- 1) on définit une famille d'érosions  $\{\alpha_i\}$  dites élémentaires par des éléments structurants convexes indexés par un entier  $i$  ;
- 2) on construit la famille d'éléments structurants  $\kappa_i = \alpha_i \dots \alpha_2 \alpha_1$  ;
- 3) ces éléments structurants définissent une famille granulométrique  $\{\kappa_i\}$  ;
- 4) le plus souvent on utilise une famille homogène, c'est-à-dire lorsque toutes les érosions élémentaires sont identiques :  $\forall i, \alpha_i = \alpha_1$ , ce qui implique  $\kappa_i = (\alpha_1)^i$ .

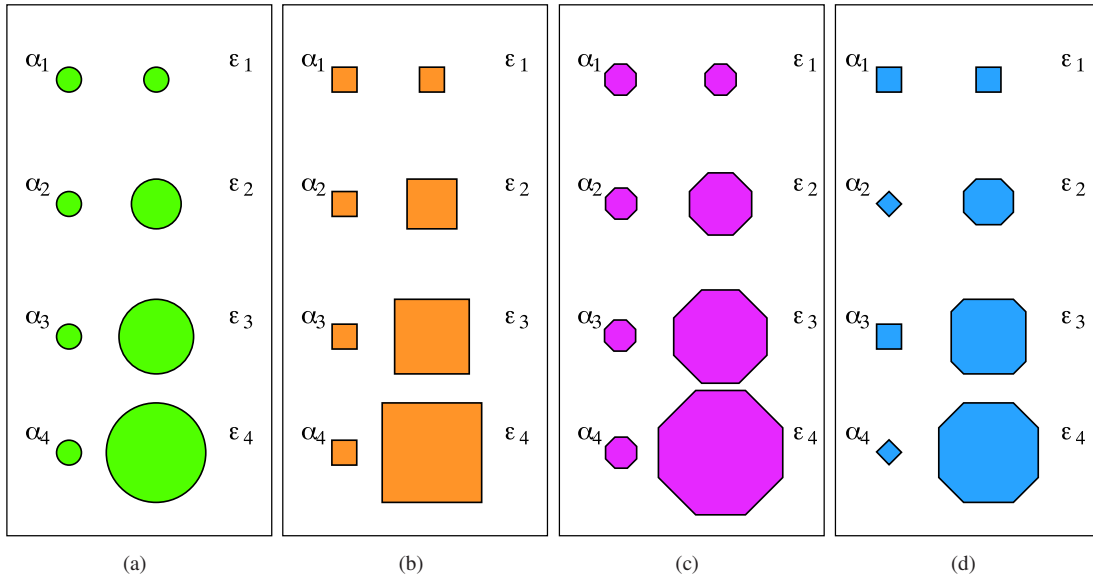


Figure 3.3. Famille granulométrique homogène (a, b, c) et hétérogène (d)

Les familles hétérogènes peuvent servir à obtenir un contrôle plus fin sur la granulométrie, en particulier elles permettent d'utiliser des éléments structurants plus isotropes. Ces familles sont illustrées en figure 3.3.

PROPRIÉTÉ 3.7.– On a la propriété suivante, par construction :  $\forall j \leq i, \gamma_{\kappa_i} \circ \gamma_{\kappa_j} = \gamma_{\kappa_i}$ . On dit que chaque  $\kappa_i$  est ouvert par tous les  $\kappa_j$  pour  $j \leq i$ . Cette propriété est appelée « propriété d'absorption ».

### 3.3.2. Application des granulométries

Les familles granulométriques permettent de définir des opérations incrémentales avec des propriétés désirables :

- on peut trivialement étendre la définition en niveaux de gris ;
- par exemple, les granulométries binaires ou en niveaux de gris peuvent servir à caractériser des textures ou à obtenir des informations sur l'image sans segmentation ;
- on donne des exemples d'application des granulométries dans *Morphologie Mathématique 1*, au chapitre ??, ainsi que dans cet ouvrage, dans la partie applications, aux chapitres ?? et ??.

### 3.3.3. Formule des érodés ultimes

L'érodé ultime est constitué de points particuliers du squelette.

DÉFINITION 3.8.– On définit l'érodé ultime de la façon suivante :

$$U(X) = \bigcup_{i \in \mathbb{N}} U_i(X), \text{ où } U_i(X) = \varepsilon_{\kappa_i}(X) \setminus \delta_{\varepsilon_{\kappa_i}(X)}^{\infty}[\varepsilon_{\kappa_{i+1}}(X)] \quad (3.1)$$

Ici,  $\delta_A^\infty(B)$  dénote la reconstruction de l'ensemble  $B$  dans le masque  $A$ , comme introduite dans *Morphologie Mathématique 1*, chapitre ?? . On utilise l'élément structurant élémentaire de la trame pour réaliser l'opération (voir plus loin).

L'érodé ultime possède les propriétés suivantes :

- croissance ;
- anti-extensité ;
- idempotence (c'est donc une ouverture) ;
- tous les  $U_i$  sont disjoints ;
- si on dilate un point de  $U_i$  par l'élément structurant  $\kappa_i$ , le résultat est appelé une *boule maximale*.

La définition d'une boule maximale discrète rejoint celle du continu : dans le continu on a une famille continue de boules.

Le processus d'obtention de l'érodé ultime est illustré en figure 3.4. Chaque point d'un érodé ultime correspond à un centre de boule maximale, donc convexe au sens défini par la famille d'éléments structurants  $\alpha_i$ . Les composantes connexes des érodés ultimes permettent d'identifier et éventuellement de séparer des parties approximativement convexes d'une particule [SER 82]. L'érodé ultime de chaque composante est la dernière étape avant sa disparition par les érosions successives, d'où son nom. Nous verrons plus loin dans ce chapitre le lien entre les érodés ultimes et la fonction distance, lorsque nous aurons défini cette dernière.

### 3.3.4. Formule de Lantuéjoul

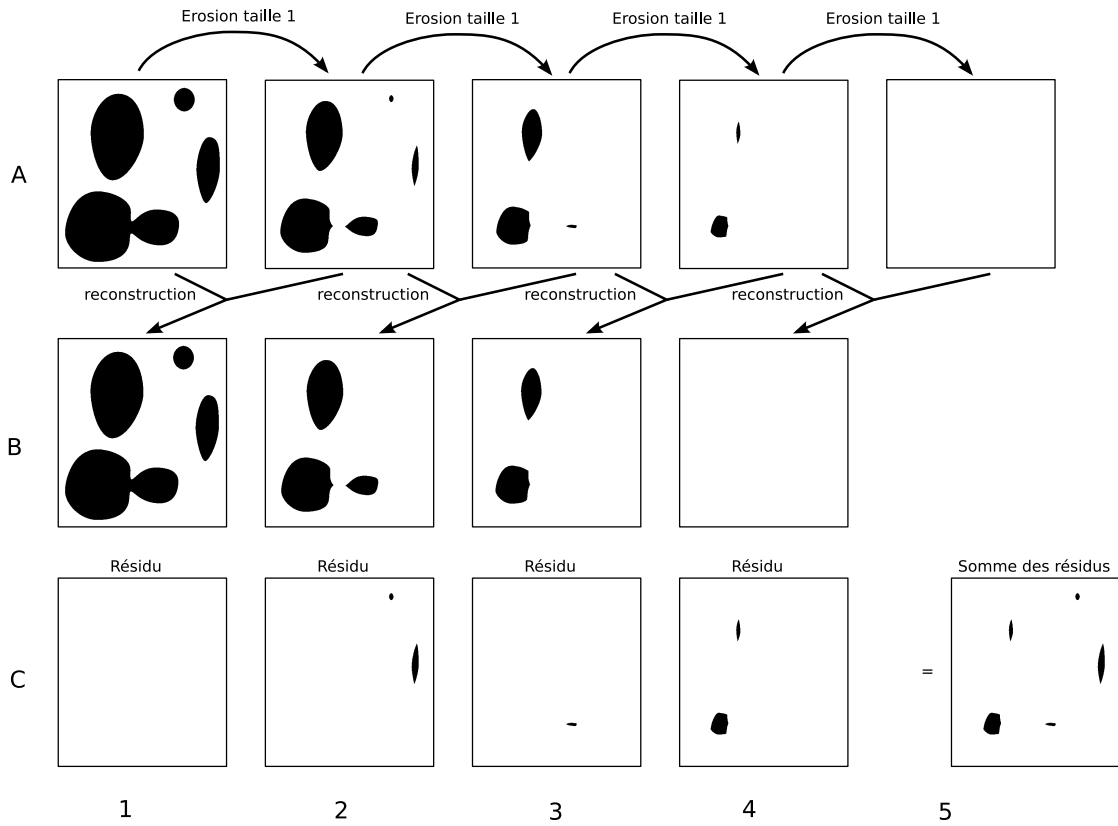
De sa définition, on déduit que l'érodé ultime est un sous-ensemble du squelette. On peut adapter la formule des érodés ultimes pour produire une autre définition du squelette. Cette formule est due à Lantuéjoul [LAN 78, LAN 80] :

**DÉFINITION 3.9 (Formule de Lantuéjoul).**– *Le squelette  $S(X)$  d'un ensemble  $X$  est l'ensemble des points définis par la formule suivante :*

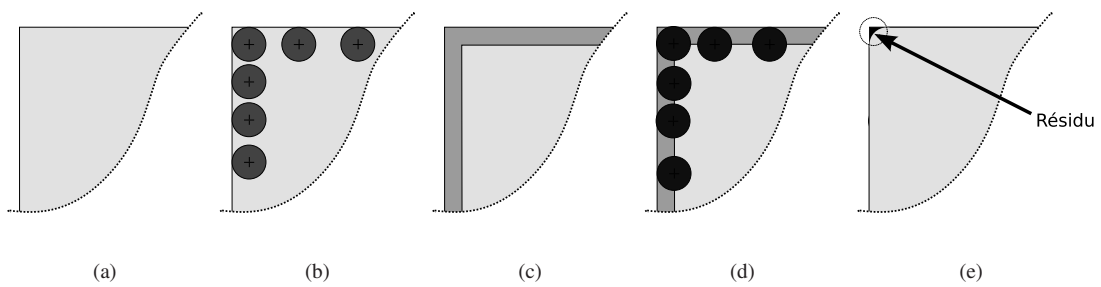
$$S(X) = \bigcup_{i \in \mathbb{N}} S_i(X), \text{ où } S_i(X) = \varepsilon_{\kappa_i}(X) \setminus \gamma_1[\varepsilon_{\kappa_i}(X)] \quad (3.2)$$

Ici,  $\gamma_1$  est l'ouverture par la boule unitaire, celle réalisée avec le plus petit élément structurant de la famille granulométrique choisie. Dans le domaine du continu elle peut être infinitésimale [MAT 92], mais ce n'est pas une nécessité. Dans le cadre discret, elle est réalisée avec  $\alpha_1$ , qui est souvent l'élément structurant élémentaire de la grille (par exemple un carré  $3 \times 3$  en 8-connexité). Cette définition s'inspire de celle des érodés ultimes, en remplaçant l'ouverture par reconstruction par l'ouverture unitaire. Les résidus de l'ouverture unitaire éliminés de chaque érosion successive constituent les éléments du squelette de Lantuéjoul. Cette procédure est illustrée en deux parties, d'abord en figure 3.5, l'illustration de la partie  $A \setminus \gamma_1(A)$  de l'équation, qui correspond aux résidus de l'ouverture. Ensuite en figure 3.6, on montre l'union des résidus pour la famille d'érosions  $A = \varepsilon_{\kappa_i}(X)$ , avec  $X$  un rectangle. Dans les deux cas on a pris un disque comme élément structurant.

Dans un cadre purement discret, la formule de Lantuéjoul s'applique toujours. En figure 3.7 nous avons l'exemple du squelette de Lantuéjoul d'une figure simple discrète, en prenant comme boule unité le voisinage 4-connexe (le point central plus ses quatre voisins les plus proches). L'important ici est de choisir une famille  $\varepsilon_{\kappa_i}$  granulométrique, par exemple avec un  $\alpha_i$  constant.

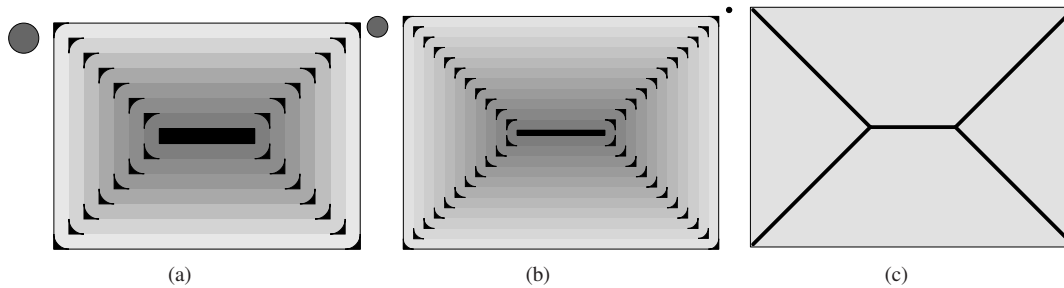


**Figure 3.4.** *Processus d'obtention de l'érodé ultime. En première ligne (ligne A), on a les érosions successives  $A_1, A_2$ , etc, avec une boule unité comme élément structurant. En deuxième lignes (ligne B), les reconstructions binaires successives de  $A_{i+1}$  dans  $A_i$ . Ainsi  $B_2$  est-il la reconstruction de  $A_3$  dans  $A_2$ . En troisième ligne (ligne C), on a les résidus  $C_i = A_i - B_i$ . Finalement, l'érodé ultime est l'union  $\cup_i C_i$ .*

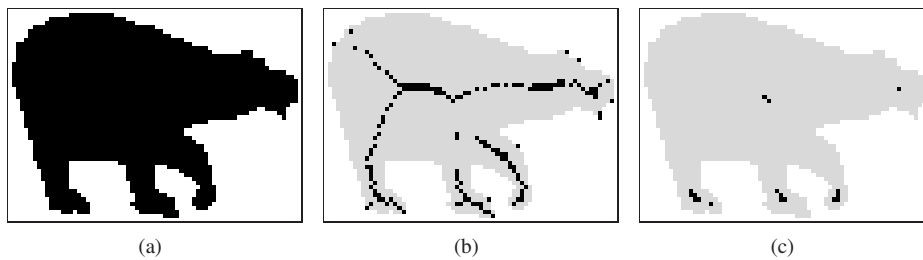


**Figure 3.5.** *Un résidu par la formule de Lantuéjoul, c'est-à-dire le résultat de  $A \setminus \gamma_1(A)$  : (a) Une partie d'un objet  $A$ , en gris clair ; (b) processus d'érosion par une boule unité ; (c) résultat de l'érosion, en gris clair sur fond sombre ; (d) processus de dilatation par la même boule ; (e) résultat de l'ouverture, et son résidu en noir*





**Figure 3.6.** La famille des résidus de l'ouverture pour une figure simple : un rectangle. Les érosions successives forment des rectangles emboîtés, du plus clair au plus foncé. La boule unité utilisée est en haut à gauche de chaque rectangle. Les résidus sont en noir. En (a) on montre le résultat avec une large boule unité pour l'ouverture. Si le diamètre de la boule unité tends vers zéro (b), soit le cas de la boule unité infinitésimale, on finit par obtenir le cas limite du squelette continu, en (c).



**Figure 3.7.** Le squelette de Lantuéjoul dans le cas discret : en (a) silhouette d'un ours, en (b) son squelette de Lantuéjoul, avec comme boule unité le voisinage 4-connexe, en (c) son érodé ultime avec la même boule unité

Nous avons la propriété importante suivante :

PROPRIÉTÉ 3.10.– *Le squelette de Lantuéjoul est constitué des centres de boules maximales.*

Sans introduire de preuve formelle, on peut voir pourquoi c'est bien le cas. Les résidus de Lantuéjoul à l'itération  $i$  d'un ensemble  $X$  du calcul de l'union sont les points conservés par une érosion de taille  $i$  et effacés par une ouverture de taille  $i + 1$ . Ils sont donc centre d'une boule de taille  $i$  contenue dans  $X$ , mais aucune boule de taille  $i + 1$  ou supérieure ne peut être centrée en ce point.

Observons que tout ensemble de points non reconstruits par l'ouverture par reconstruction dans la formule des érodés ultimes ne le serait pas non plus dans la formule de Lantuéjoul, d'où la propriété suivante.

PROPRIÉTÉ 3.11.– *L'érodé ultime est un sous-ensemble du squelette de Lantuéjoul.*

### 3.4. Distances discrètes

Cette section présente une rapide introduction aux principales distances discrètes utilisées en traitement d'images. Pour un traitement plus détaillé, le lecteur pourra se reporter à [THI 07], dans cette collection. A

partir d'une de ces distances, on déduit immédiatement les notions de boule et de boule maximale pour un objet  $X$  donné, et l'on appelle *axe médian de  $X$*  l'ensemble des centres des boules maximales pour  $X$ .

Considérons un ensemble  $E$  (dans la suite nous prendrons généralement  $E = \mathbb{Z}^n$ , avec dans la plupart des exemples  $n = 2$ ), une *distance sur  $E$*  est une application  $d$  de  $E \times E$  dans  $\mathbb{R}^+$  qui vérifie :

$$\begin{aligned} \forall x, y \in E, \quad d(x, y) &= d(y, x) && \text{(symétrie)} \\ \forall x, y \in E, \quad d(x, y) &= 0 \Leftrightarrow x = y && \text{(séparation)} \\ \forall x, y, z \in E, \quad d(x, z) &\leq d(x, y) + d(y, z) && \text{(inégalité triangulaire)} \end{aligned}$$

Si l'on considère un sous-ensemble  $Y$  de  $E$  et un point  $x$  de  $E$ , la *distance de  $x$  à  $Y$*  est définie par  $d(x, Y) = \min\{d(x, y) \mid y \in Y\}$ .

Soit  $X$  un sous-ensemble strict de  $E$  dit « objet », la *carte de distance de  $X$*  est l'application  $D_X$  de  $E$  dans  $\mathbb{R}^+$  définie par :

$$\forall x \in E, \quad D_X(x) = d(x, \overline{X})$$

où  $\overline{X}$  désigne le complémentaire de  $X$  dans  $E$ . Par abus de langage, on conservera l'appellation « carte de distance » même si  $d$  ne vérifie pas l'inégalité triangulaire.

En traitement des images numériques, on a beaucoup utilisé les distances connues sous les noms de « distance de Manhattan » et « distance de l'échiquier » (en 2D), car ce sont les distances les plus simples à calculer [ROS 68]. Nous les noterons respectivement  $d_4$  et  $d_8$ , en référence au nombre de points qui sont à distance 1 d'un point donné. On les définit par :

$$\begin{aligned} d_4(x, y) &= \sum_{i=1}^n |y_i - x_i| \\ d_8(x, y) &= \sup_{i=1}^n |y_i - x_i| \end{aligned}$$

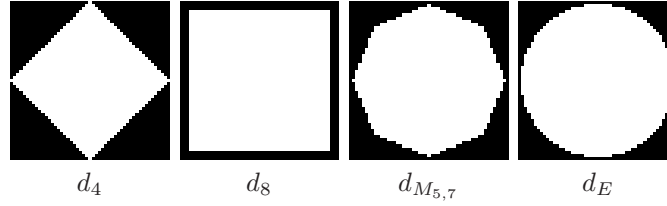
Le problème majeur de ces distances est leur non-invariance par rotation. En pratique, cela signifie que si l'on effectue une mesure de distance sur un objet extrait d'une image numérique, on obtiendra des résultats sensiblement différents selon l'orientation de l'objet lors de la prise de vue. Notons qu'une parfaite invariance par rotation ne peut être atteinte dès lors que l'on utilise des images discrètes, cependant on souhaiterait que les effets des rotations sur la mesure des distances ne soient pas supérieurs au pas de discrétisation.

Pour tenter de pallier ce défaut des distances  $d_4$  et  $d_8$ , les distances dites de chanfrein ont été introduites et étudiées [MON 68, BOR 84]. Pour définir ces distances on doit supposer que l'ensemble  $E$  est muni d'une structure de module (voir [THI 07]). On appelle *masque de chanfrein* un ensemble fini de couples formés d'un déplacement  $v_k$  et d'une pondération  $p_k$  :

$$M = \{(v_k, p_k) \mid v_k \in E, p_k \in \mathbb{R}^+, k \in \{1, \dots, m\}\}$$

vérifiant les conditions suivantes :

- i) chacun des  $v_k$  et des  $p_k$  est non nul,
- ii)  $M$  possède une symétrie centrale, c'est-à-dire que si  $(v, p)$  est dans  $M$  alors  $(-v, p)$  est aussi dans  $M$ ,
- iii)  $M$  contient une base de  $E$ , c'est-à-dire que pour tout  $x$  de  $E$ , il existe un  $m$ -uplet  $(a_1 \dots a_m)$  d'entiers positifs ou nuls tel que  $\sum_{1 \leq k \leq m} a_k v_k = x$ .



**Figure 3.8.** Boules discrètes obtenues à partir des distances  $d_4$ ,  $d_8$ ,  $d_{M_{5,7}}$ , et de la distance euclidienne  $d_E$

Soient  $x$  et  $y$  deux points de  $E$ , on pose :

$$d_M(x, y) = \min \left\{ \sum_{1 \leq k \leq m} a_k p_k \mid y = x + \sum_{1 \leq k \leq m} a_k v_k, a_k \in \mathbb{N} \right\}$$

Par exemple, le masque suivant permet de retrouver la distance  $d_4$  :

$$M_4 = \{((1, 0), 1), ((0, 1), 1), ((-1, 0), 1), ((0, -1), 1)\}.$$

Le masque :

$$M_{5,7} = \{((1, 0), 5), ((1, 1), 7), ((0, 1), 5), ((-1, 1), 7), ((-1, 0), 5), ((-1, -1), 7), ((0, -1), 5), ((1, -1), 7)\}$$

est souvent utilisé en 2D. En jouant sur le nombre de couples du masque et les pondérations utilisées, il est possible d'obtenir des approximations plus ou moins précises de la distance euclidienne, dont nous allons parler plus bas. On peut comparer, figure 3.8, les formes des boules obtenues à partir des différentes distances.

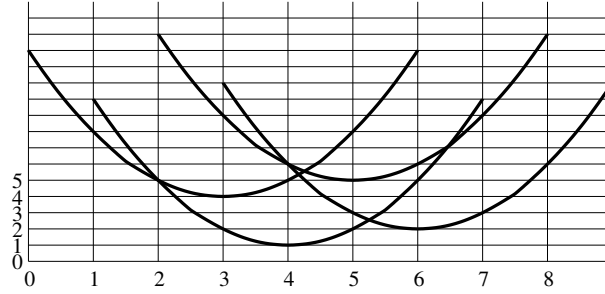
Cependant, la plus faible sensibilité aux effets des rotations ne peut être obtenue que par l'utilisation de la distance euclidienne  $d_E$  :

$$\begin{aligned} d_E^2(x, y) &= (y - x)^2 = \sum_{i=1}^n (y_i - x_i)^2 \\ d_E(x, y) &= \sqrt{d_E^2(x, y)} \end{aligned}$$

La distance euclidienne au carré  $d_E^2$ , qui n'est pas une distance car elle ne vérifie pas l'inégalité triangulaire, est cependant suffisante dans de nombreuses applications, et plus simple à manipuler informatiquement que la distance euclidienne : en effet elle ne fait intervenir que des entiers lorsque les points sont à coordonnées entières. De plus, si l'on sait calculer une carte de distance relative à  $d_E^2$ , il est facile d'en déduire une carte de distance euclidienne.

Pour calculer une carte de distance relative à  $d_E^2$ , l'algorithme naïf (application directe de la définition) a une complexité quadratique en fonction du nombre de points de l'image, ce qui est très inefficace en pratique. En 1980, Danielsson [DAN 80b] propose un algorithme linéaire en temps de calcul permettant de calculer une approximation très correcte de ce type de carte de distance pour les images à deux dimensions.

Cependant, ce n'est qu'assez récemment que ce problème a reçu une solution vraiment satisfaisante : un algorithme donnant un résultat exact, linéaire en temps de calcul, et généralisable quelle que soit la dimension  $n$  de l'espace. Cet algorithme, publié en 1996 par Hirata [HIR 96], se base sur des travaux antérieurs de Saito et Toriwaki [SAI 94], et fut retrouvé indépendamment par Meijster *et al.* [MEI 00]. Une autre approche, basée sur la notion de diagramme de Voronoï, permet également d'obtenir un algorithme efficace et exact [MAU 03]. Dans les deux cas, il s'agit d'algorithmes dits « séparables », qui calculent récursivement une carte de distance de dimension  $n$  à partir de cartes de dimension  $n - 1$  calculées indépendamment les unes des autres (ligne par ligne, plan par plan, etc.) Nous allons décrire en détail le premier de ces algorithmes dans le cas où  $n = 2$ .



**Figure 3.9.** Paraboles obtenues pour :  $j = 3, d_{X_1}(x_1, j) = 4$ ;  
 $j = 4, d_{X_1}(x_1, j) = 1$ ;  $j = 5, d_{X_1}(x_1, j) = 2$ ;  $j = 6, d_{X_1}(x_1, j) = 5$

Posons  $E = [0 \dots \ell - 1] \times [0 \dots h - 1]$ . Pour calculer la carte de distance  $D_x$  relative à un sous-ensemble  $X$  de  $E$ , on doit calculer, pour tout point  $x$  de  $E$  la quantité

$$d_X(x) = \min\{(y_1 - x_1)^2 + (y_2 - x_2)^2 \mid y \in \overline{X}\} \quad (3.3)$$

Posons, pour tout  $j \in [0 \dots h - 1]$

$$d_{X_1}(x_1, j) = \min\{(y_1 - x_1)^2 \mid (y_1, j) \in \overline{X}\} \quad (3.4)$$

On peut ré-écrire (3.3) sous la forme

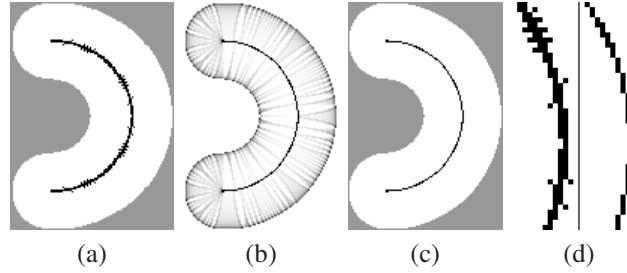
$$d_X(x) = \min\{d_{X_1}(x_1, j) + (j - x_2)^2 \mid 0 \leq j < h\} \quad (3.5)$$

Le calcul des valeurs de  $d_{X_1}$  s'effectue indépendamment d'après (3.4) pour chaque ligne de l'image, et les  $d_{X_1}$  sont stockés dans un tableau. Ensuite, on calcule les valeurs  $d_X$  par (3.5) indépendamment colonne par colonne.

Pour interpréter (3.5) géométriquement, remarquons que lorsque  $x_1$  est fixé, comme c'est le cas lorsqu'on effectue les calculs pour une colonne donnée, et pour un  $j$  donné, l'expression  $d_{X_1}(x_1, j) + (j - x_2)^2$  définit une parabole dont le minimum est atteint pour  $x_2 = j$  et vaut  $d_{X_1}(x_1, j)$ . Les calculs relatifs à (3.5) s'apparentent donc à la recherche de l'enveloppe inférieure d'une famille de paraboles (voir figure 3.9).

L'algorithme décrit dans [HIR 96, MEI 00] consiste à calculer les points d'abscisse entière de cette enveloppe en deux passes, l'une considérant les demi-paraboles ascendantes et l'autre, les demi-paraboles descendantes. Une structure de pile permet de n'opérer sur chaque parabole que deux opérations : l'une d'empilement, l'autre de dépilement, ce qui assure la linéarité de l'algorithme. De plus, sa mise en œuvre ne requiert qu'une vingtaine de lignes de code.

En suivant la même approche, Coeurjolly a proposé des algorithmes de complexité optimale pour calculer un sous-ensemble de l'axe médian euclidien exact suffisant pour la reconstruction de l'objet original, pour effectuer cette reconstruction, et pour calculer la fonction qui à chaque point  $x$  d'un objet  $X$  associe l'ensemble des points de  $\overline{X}$  qui sont à distance minimale de  $x$  [COE 03, COU 07a]. Cette fonction, appelée projection sur  $\overline{X}$ , joue un rôle essentiel dans la définition et le calcul de la fonction bissectrice, qui fait l'objet de la section 3.5 de ce chapitre.



**Figure 3.10.** (a) : un ensemble  $X$  et son axe médian euclidien (en noir) ;  
 (b) : la fonction bissectrice  $\theta_X$  (les points sombres correspondent aux grands angles) ; (c) : l'axe médian filtré, basé sur les valeurs de  $\theta_X$  ; (d) : détail de l'axe médian non-filtré et filtré

### 3.5. Fonction bissectrice

Désignons par  $E$  l'ensemble  $\mathbb{R}^n$  ou  $\mathbb{Z}^n$ , soit  $S$  un sous-ensemble non vide de  $E$ , et soit  $x \in E$ . La *projection de  $x$  sur  $S$* , désignée par  $\Pi_S(x)$ , est l'ensemble des points  $y$  de  $S$  qui sont à distance minimale de  $x$  ; plus précisément  $\Pi_S(x) = \{y \in S, \forall z \in S, d(y, x) \leq d(z, x)\}$ .

L'*angle bissecteur* d'un point  $x$  de  $X$  peut être défini, dans le cadre continu, comme le plus grand angle non signé formé par  $x$  (en tant que sommet) et deux points quelconques de la projection de  $x$  sur  $\overline{X}$  [MEY 79, TAL 92]. En particulier, si  $|\Pi_{\overline{X}}(x)| = 1$ , alors l'angle bissecteur de  $x$  vaut zéro. La *fonction bissectrice de  $X$*  est la fonction qui associe à chaque point  $x$  de  $X$  son angle bissecteur.

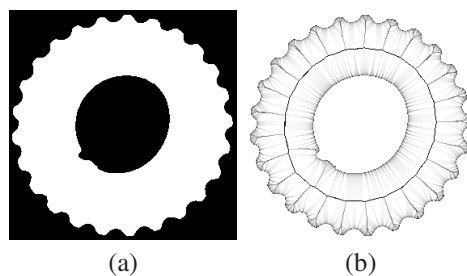
Cette fonction bissectrice a été utilisée dans [ATT 96] pour bâtir un critère de filtrage pour les squelettes basés sur les diagrammes de Voronoï dans le plan continu. Elle a également été adaptée au cas discret dans [TAL 92, MAL 98, COU 07a]. Nous donnons ici la définition proposée dans [COU 07a].

**DÉFINITION 3.12.**— Soit  $X \subset E$ , et soit  $x \in X$ . La projection étendue de  $x$  sur  $\overline{X}$ , notée  $\Pi_{\overline{X}}^e(x)$ , est l'union des ensembles  $\Pi_{\overline{X}}(y)$ , pour tout  $y$  dans  $\Gamma_4(x)$  tel que  $d^2(y, \overline{X}) \leq d^2(x, \overline{X})$ .

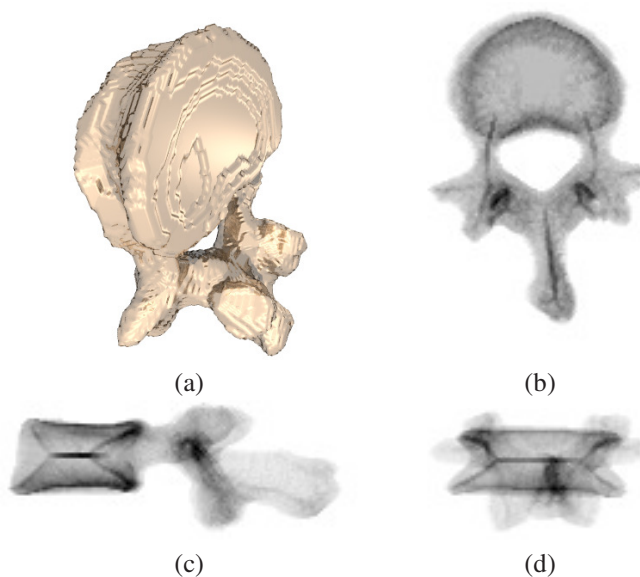
L'*angle bissecteur (discret)* de  $x$  dans  $X$ , noté  $\theta_X(x)$ , est l'angle non signé maximum entre les vecteurs  $\vec{xy}$ ,  $\vec{xz}$ , pour tous les  $y, z$  dans  $\Pi_{\overline{X}}^e(x)$ . En particulier, si  $|\Pi_{\overline{X}}^e(x)| = 1$ , alors  $\theta_X(x) = 0$ . La fonction bissectrice (discrète) de  $X$ , notée  $\theta_X$ , est la fonction qui associe à chaque point  $x$  de  $X$  son angle bissecteur dans  $X$ .

Dans [COU 07a], un algorithme de complexité optimale est proposé pour calculer  $\Pi_{\overline{X}}$ . Pour chaque point  $x$  de  $X$ , nous devons ensuite calculer  $\Pi_{\overline{X}}^e(x)$  en utilisant la relation d'adjacence  $\Gamma$  et la carte de distance  $D_X^2$ . La dernière étape pour obtenir l'angle bissecteur consiste à calculer l'angle non signé maximum entre toutes les paires de vecteurs  $\{\vec{xy}, \vec{xz}\}$ , pour tous les  $y, z$  dans  $\Pi_{\overline{X}}^e(x)$ . En pratique, le nombre d'éléments dans les projections étendues est petit ; donc considérer toutes les paires possibles constitue le meilleur choix, bien qu'il existe des algorithmes de complexité sous-quadratique pour effectuer cette étape.

Dans la figure 3.10, nous montrons un objet  $X$  avec son axe médian (a) et sa fonction bissectrice  $\theta_X$  (b). Nous illustrons l'usage de cette fonction pour éliminer des points non-essentiels de cet axe médian : dans (c), nous montrons les points de l'axe médian (en noir) qui ont un angle bissecteur supérieur à 40 degrés. Un détail des deux axes est agrandi dans (d). Notons que seuls les angles bissecteurs des points de l'axe médian doivent être calculés pour cette application. La figure 3.11 montre la fonction bissectrice d'une forme 2D plus complexe.



**Figure 3.11.** (a) : un objet  $X$  (en blanc); (b) : la fonction bissectrice de  $X$

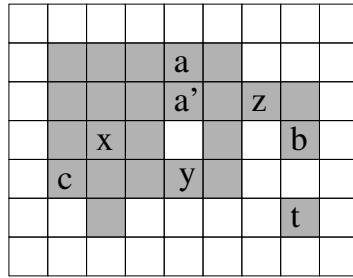


**Figure 3.12.** (a) : une vue d'un objet  $X$ , sous-ensemble de  $\mathbb{Z}^3$  (vertèbre).  
(b,c,d) : la fonction bissectrice de  $X$ , illustrée à la manière d'une « radiographie » : le niveau de gris d'un point correspond à la moyenne des angles bissecteurs des points sur une ligne droite parallèle à l'un des trois axes

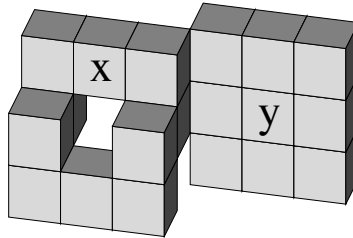
La définition et le calcul de cette fonction bissectrice discrète peuvent être étendus sans difficulté à  $\mathbb{Z}^3$ . Pour conclure cette section, nous présentons figure 3.12 une illustration de la fonction bissectrice d'un objet tri-dimensionnel (une vertèbre).

### 3.6. Transformations homotopiques

Comme nous l'avons vu, l'axe médian discret n'est pas en général homotope à l'objet original. Les algorithmes pour obtenir l'axe médian procèdent en général par amincissements successifs, ce que l'on peut voir comme un processus qui retire séquentiellement des points de l'objet. La notion de point simple permet de garantir qu'une telle transformation préserve la topologie.



**Figure 3.13.** Points simples en 2D. L'ensemble  $X$  est composé des pixels gris. Les pixels  $a, a', b, c$  sont simples tandis que  $x, y, z, t$  ne sont pas simples



**Figure 3.14.** Un ensemble  $X$  de voxels. Les voxels  $x$  et  $y$  ne sont pas simples

Intuitivement, un point  $x$  d'un objet discret  $X \subset \mathbb{Z}^n$  est un point simple si l'on peut retirer  $x$  de  $X$  sans modifier ses caractéristiques topologiques. Cette notion, que l'on voit apparaître dans les travaux séminaux de Duda, Hart et Munson [DUD 67] d'une part, Golay [GOL 69] et Rosenfeld [ROS 70] d'autre part, a été depuis l'objet d'une abondante littérature (voir par exemple [KON 89]). En particulier, des caractérisations locales des points simples ont été proposées, dans les espaces de dimension 2, 3 et même 4, sur lesquelles des algorithmes d'amincissement homotopique efficaces sont basés.

Dans la figure 3.13, les points (ou pixels)  $x, y, z, t$  ne sont pas simples : le retrait de  $x$  de l'ensemble de pixels  $X$  crée une nouvelle composante connexe du complémentaire  $\overline{X}$  de  $X$  ; le retrait de  $y$  fusionne deux composantes connexes de  $\overline{X}$  ; le retrait de  $z$  scinde une composante connexe de  $X$  ; et le retrait de  $t$  détruit une composante connexe de  $X$ . Par contre, les pixels  $a, b$  et  $c$  sont simples.

A travers ces exemples, on comprend qu'en 2D la notion de connexité (à la fois pour  $X$  et pour  $\overline{X}$ ) suffit pour caractériser les points simples.

Les choses se compliquent en 3D. Considérons par exemple l'ensemble  $X$  de la figure 3.14. On peut retirer le voxel  $x$  ou le voxel  $y$  de  $X$  sans scinder, fusionner, créer ou détruire une composante connexe de  $X$  ou de  $\overline{X}$ . Cependant, ni  $x$  ni  $y$  ne sont des points simples. La raison en est que, si l'on retire  $x$ , on supprime un tunnel et que si l'on retire  $y$ , on crée un nouveau tunnel. La notion de tunnel peut être formalisée par le groupe fondamental introduit par Poincaré, il s'agit d'un invariant topologique que préserve toute déformation continue.

Cependant, il est encore possible de caractériser en 3D les points simples par des conditions locales qui sont seulement basées sur la connexité (voir [BER 94a, BER 94b]), nous donnerons une de ces caractérisations par la suite. Il est déjà remarquable qu'une notion globale, la notion de point simple, puisse être caractérisée

localement ; le fait que l'on puisse se contenter de la connexité pour obtenir une telle caractérisation en 3D est également surprenant. En revanche, la connexité seule ne permet plus de caractériser les points simples en dimension supérieure à 3. Dans [COU 08] est présentée une définition des points simples basée sur la notion de collapse, ainsi que des caractérisations locales valables dans les espaces de dimension 2, 3 et 4. Cet article expose aussi les difficultés qui font qu'à partir de la dimension 5, ce type de caractérisation n'est plus envisageable.

### 3.6.1. Voisinages, connexité

Nous rappelons ici les définitions de base de la topologie digitale pour les images binaires [KON 89].

Un point  $x \in \mathbb{Z}^d$  ( $d = 2, 3$ ) est défini par  $(x_1, \dots, x_d)$  avec  $x_i \in \mathbb{Z}$ . Nous considérons les relations de voisinage  $\Gamma_4$  et  $\Gamma_8$  définies, pour tout point, par  $x \in \mathbb{Z}^2$  :

$$\begin{aligned}\Gamma_4(x) &= \{y \in \mathbb{Z}^2 \mid |y_1 - x_1| + |y_2 - x_2| \leq 1\}, \\ \Gamma_8(x) &= \{y \in \mathbb{Z}^2 \mid \max(|y_1 - x_1|, |y_2 - x_2|) \leq 1\},\end{aligned}$$

et les relations de voisinage  $\Gamma_6$ , et  $\Gamma_{26}$  et  $\Gamma_{18}$  définies par, pour tout point  $x \in \mathbb{Z}^3$  :

$$\begin{aligned}\Gamma_6(x) &= \{y \in \mathbb{Z}^3 \mid |y_1 - x_1| + |y_2 - x_2| + |y_3 - x_3| \leq 1\}, \\ \Gamma_{26}(x) &= \{y \in \mathbb{Z}^3 \mid \max(|y_1 - x_1|, |y_2 - x_2|, |y_3 - x_3|) \leq 1\}, \\ \Gamma_{18}(x) &= \{y \in \Gamma_{26}(x) \mid |y_1 - x_1| + |y_2 - x_2| + |y_3 - x_3| \leq 2\}.\end{aligned}$$

Dans la suite, nous désignerons par  $n$  un nombre tel que  $n \in \{4, 8, 6, 18, 26\}$ . Nous définissons  $\Gamma_n^*(x) = \Gamma_n(x) \setminus \{x\}$ . Le point  $y \in E$  est dit  $n$ -adjacent au point  $x \in E$  si  $y \in \Gamma_n^*(x)$ . Un  $n$ -chemin est une séquence de points  $x_0 \dots x_k$  telle que  $x_i$  est  $n$ -adjacent à  $x_{i-1}$  pour  $i = 1 \dots k$ .

Soit  $X \subseteq E$ , nous disons que deux points  $x, y$  de  $X$  sont  $n$ -connectés dans  $X$  s'il existe un  $n$ -chemin dans  $X$  entre ces deux points. Ceci définit une relation d'équivalence. Les classes d'équivalence pour cette relation sont les *composantes  $n$ -connexes de  $X$* . Un sous-ensemble  $X$  de  $E$  est dit  $n$ -connexe s'il se compose exactement d'une composante  $n$ -connexe.

L'ensemble composé de toutes les composantes  $n$ -connexes de  $X$  est noté  $C_n(X)$ . Un sous-ensemble de  $Y$  de  $E$  est dit  $n$ -adjacent à un point  $x \in E$  s'il existe un point  $y \in Y$  qui soit adjacent à  $x$ . L'ensemble des composantes  $n$ -connexes de  $X$   $n$ -adjacentes au point  $x$  est noté  $C_n^x(X)$ . Remarquons que  $C_n(X)$  et  $C_n^x(X)$  sont des ensembles de sous-ensembles de  $X$  et non des ensembles de points. De plus, si  $S$  est un ensemble, nous désignerons par  $|S|$  le nombre de ses éléments.



### 3.6.2. Nombres de connexité et points simples

Donnons tout d'abord la définition des nombres de connexité pour le cas 2D. Nous verrons par la suite que le cas 3D est plus complexe.

Soit  $X \subseteq \mathbb{Z}^2$  et  $x \in \mathbb{Z}^2$ . Les *nombres de connexité*  $T_n(x, X)$  sont définis par :  $T_n(x, X) = |C_n^x[\Gamma_8^*(x) \cap X]|$

Si nous utilisons une  $n$ -connexité pour  $X$  nous devons utiliser une  $\bar{n}$ -connexité pour  $\bar{X}$ , par exemple en 2D la 4-connexité pour  $X$  est associée à la 8-connexité pour  $\bar{X}$  (et vice-versa), et en 3D la 6-connexité pour  $X$  est associée à la 18- ou la 26-connexité pour  $\bar{X}$  (et vice-versa). Ceci est nécessaire pour avoir une correspondance entre la topologie de  $X$  et celle de  $\bar{X}$  (voir [KON 89]). De plus, il est parfois nécessaire en 3D de distinguer la 6-connexité associée avec la 18-connexité de la 6-connexité associée avec la 26-connexité. Pour rendre explicite cette distinction, nous utiliserons le symbole  $6^+$  pour indiquer la notion de 6-connexité associée avec la 18-connexité (voir [BER 94a]). Donc, nous avons les possibilités suivantes en 2D :  $(n, \bar{n}) = (4, 8)$  ou  $(8, 4)$  ; et en 3D :  $(n, \bar{n}) = (6, 26)$ ,  $(26, 6)$ ,  $(6^+, 18)$  ou  $(18, 6^+)$ .

Dans le cas 3D, la définition des nombres de connexité repose sur la notion de voisinage géodésique. Soit  $X \subseteq \mathbb{Z}^3$  et  $x \in \mathbb{Z}^3$ , le  $n$ -voisinage géodésique de  $x$  dans  $X$  d'ordre  $k$  est l'ensemble  $\Gamma_n^k(x, X)$  défini récursivement par :

$$\begin{aligned}\Gamma_n^1(x, X) &= \Gamma_n^*(x) \cap X, \text{ et} \\ \Gamma_n^k(x, X) &= \cup\{\Gamma_n(y) \cap \Gamma_{26}^*(x) \cap X, y \in \Gamma_n^{k-1}(x, X)\}.\end{aligned}$$

En d'autres termes,  $\Gamma_n^k(x, X)$  est l'ensemble composé des points  $y$  de  $\Gamma_{26}^*(x) \cap X$  tels qu'il existe un  $n$ -chemin  $\pi$  de  $x$  à  $y$  de longueur inférieure ou égale à  $k$ , tous les points de  $\pi$ , sauf éventuellement  $x$ , appartenant à  $\Gamma_{26}^*(x) \cap X$ . Les *voisinages géodésiques*  $G_n(x, X)$  sont définis par :  $G_6(x, X) = \Gamma_6^2(x, X)$ ,  $G_{6^+}(x, X) = \Gamma_6^3(x, X)$ ,  $G_{18}(x, X) = \Gamma_{18}^2(x, X)$ , et  $G_{26}(x, X) = \Gamma_{26}^1(x, X)$ .

Nous pouvons maintenant donner une définition [MAL 93, BER 94a, BER 94b] des nombres de connexité en 3D.

**DÉFINITION 3.13.**– Soit  $X \subseteq \mathbb{Z}^3$  et  $x \in \mathbb{Z}^3$ . Les nombres de connexité  $T_n(x, X)$  sont définis par :  $T_n(x, X) = |C_n[G_n(x, X)]|$

Notons qu'une formulation en termes de voisinages géodésiques permet également de retrouver la définition des nombres de connexité en 2D.

Si nous utilisons la  $n$ -connexité pour  $X$  et la  $\bar{n}$ -connexité pour  $\bar{X}$ , les nombres  $T_n(x, X)$  et  $T_{\bar{n}}(x, \bar{X})$  permettent de décrire les caractéristiques topologiques du point  $x$  dans l'objet  $X$ . En particulier, les nombres de connexité permettent de détecter si un point est simple ou non [BER 94a, BER 94b], en 2D comme en 3D :

**PROPRIÉTÉ 3.14.**– Soit  $X \subseteq E$  et  $x \in X$ . Le point  $x$  est  $n$ -simple si et seulement si  $T_n(x, X) = 1$  et  $T_{\bar{n}}(x, \bar{X}) = 1$ .

L'interprétation intuitive de cette caractérisation est que pour qu'un point soit simple, il faut et il suffit qu'il ait dans son voisinage une seule composante « objet » et une seule composante « fond ».

### 3.6.3. Amincissement homotopique

Il reste encore à déterminer comment décider de l'ordre dans lequel les points simples doivent être retirés. Nous présentons, avec l'algorithme 1, une stratégie d'amincissement qui consiste à contrôler cet ordre à l'aide d'une fonction numérique appelée fonction de priorité. Cette fonction associe à tout point  $x$  de  $X$  un nombre entier ou réel  $P(x)$ , qui représente la priorité du point  $x$ . Les points de  $X$  seront traités selon les valeurs croissantes de  $P$ . On pourra donner à certains points  $x$  la priorité  $P(x) = +\infty$ , ce qui aura pour signification que ces points ne doivent pas être retirés ; autrement dit, les points de priorité infinie seront des points d'ancrage (voir [DAV 81, VIN 91a, PUD 98]) pour l'amincissement.

---

#### Algorithme 1 : amincissement guidé

---

**Données :**  $X \subseteq E$ ,  $P$  une fonction de  $X$  dans  $\mathbb{Z}$  ou  $\mathbb{R}$

**Résultat :**  $X$

**répéter**

    | Soit  $x$  un élément de  $X$  tel que  $x$  est simple pour  $X$  et  $P(x)$  est minimal ;

    |  $X = X \setminus \{x\}$  ;

**jusqu'à stabilité ;**

---

Nous verrons dans la suite (section 3.6.5) les problèmes spécifiques qui se posent lorsque la fonction de priorité est une carte de distance euclidienne.

### 3.6.4. Algorithmes séquentiels et parallèles d'amincissement

Les transformations homotopiques décrites dans ce chapitre sont de nature séquentielle, au sens où après modification de chaque point simple, il est nécessaire de prendre en compte cette modification dans les tests de simplicité d'autres points. Autrement dit, on ne garantit pas de préserver la topologie si l'on retire simultanément plusieurs points simple : par exemple dans la figure 3.13, si l'on retire les deux points simples  $a$  et  $a'$ , on fusionne deux composantes du complémentaire de l'objet.

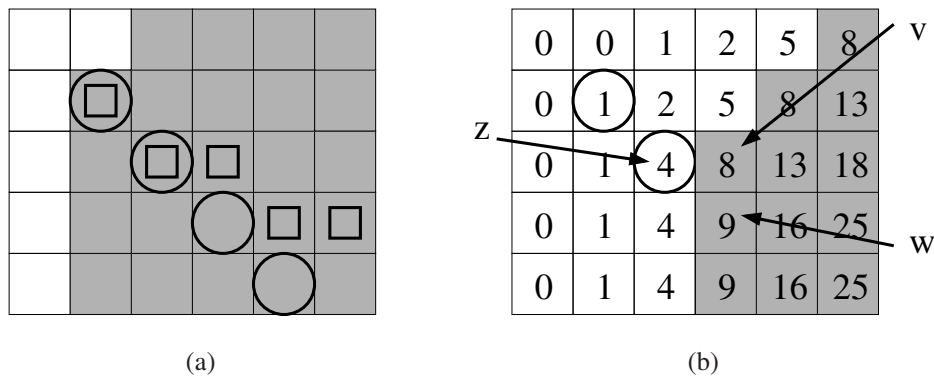
Il en résulte que parfois, des choix arbitraires doivent être effectués quant à l'ordre dans lequel les points simples sont examinés, menant à des résultats différents. Les stratégies d'amincissement en parallèle, non décrites ici, permettent en revanche d'obtenir des squelettes définis de façon unique.

Cependant, les conditions permettant de garantir la préservation de la topologie lorsque l'on retire des points en parallèle sont bien plus complexes à établir que la caractérisation des points simples. De nombreuses tentatives, durant ces quarante dernières années, ont été faites pour essayer de résoudre ce problème [COU 06]. La notion d'ensemble non-simple minimal, introduite par C. Ronse [RON 88], permet de tester si un ensemble donné de points simples peut être enlevé tout en préservant la topologie. La notion de point P-simple proposée dans [BER 95] permet de plus, de concevoir des algorithmes parallèles d'amincissement 3D dont le caractère homotopique est garanti par construction. Récemment, un cadre général pour l'étude de l'amincissement parallèle en dimension quelconque a été développé par G. Bertrand [BER 07]. Ce cadre, qui a pour centre la notion de noyau critique, généralise à la fois celui des ensembles non-simples minimaux et celui des points P-simples. Le lecteur intéressé pourra consulter le site <http://www.esiee.fr/~info/ck> pour une revue complète sur les noyaux critiques.

### 3.6.5. Squelette basé sur la distance euclidienne

Les méthodes de squelettisation basées sur l'amincissement homotopique procurent la garantie formelle que l'objet original et son squelette ont la même topologie. Pour obtenir un squelette homotopique contenant l'axe médian euclidien, la méthode la plus simple consiste à calculer un amincissement homotopique ultime de l'objet  $X$  contraint par  $AM(X)$ , c'est-à-dire à retirer itérativement des points simples de  $X$  n'appartenant pas à  $AM(X)$  jusqu'à stabilité, en prenant la carte de distance euclidienne de  $X$  comme fonction de priorité de manière à sélectionner en premier les points qui sont les plus proches du fond. Cela peut se faire en utilisant la procédure d'amincissement guidée vue plus haut (Algorithm 1), avec  $P = D_X$  et  $Y = MA(X)$ .

Le défaut de cette méthode a été bien analysé dans [TAL 92] : elle ne garantit pas que les points du squelette homotopique obtenu qui se trouvent hors de l'axe médian soient bien centrés dans l'objet, plus précisément, certaines parties du squelette peuvent ne pas se trouver sur les lignes de plus grande pente de la carte de distance euclidienne. Dans la figure 3.15, nous montrons sur un exemple le type de problème qui peut se produire.



**Figure 3.15.** Illustration d'une distortion géométrique du squelette (a) : partie d'un objet (en gris), points du squelette trouvés par un algorithme d'amincissement guidé par la carte de distance (indiqués par des cercles), chemin de plus grande pente vis-à-vis de cette même carte (indiqués par des carrés). (b) : une étape d'amincissement (voir le texte). Les nombres indiquent les valeurs de la carte de distance.

Dans la figure 3.15b, les nombres correspondent aux carrés des distances euclidiennes de chaque point au plus proche point du fond. Le point entouré, qui a pour valeur 1, est l'un des points appartenant à l'ensemble de contrainte  $Y$  (voir l'algorithme alg :AmincissementGuide), c'est-à-dire un point de l'axe médian. Supposons que tous les points de valeur inférieure ou égale à 8 aient été traités par l'algorithme. A cette étape, les points en gris sont toujours dans l'objet  $X$ , ainsi que les deux points entourés (le point à 1 car il est dans  $Y$ , et celui à 4 car il a été détecté comme non simple à l'étape où il a été traité). Tous les autres points sont en dehors de  $X$ . Clairement, le point  $v$  de valeur 8 adjacent à  $z$  de valeur 4 sera sélectionné avant son voisin  $w$  de valeur 9, et puisqu'il sera simple à ce stade, il sera retiré de  $X$ . Ce comportement sera reproduit à des étapes ultérieures, créant une branche du squelette en diagonale, en contradiction avec la propriété du squelette continu qui dit que cette branche devrait suivre la ligne de plus grand pente de la carte de distance. En effet, calculons les pentes des segments  $zv$  et  $zw$  dans notre exemple :  $(\sqrt{8} - \sqrt{4})/1 \approx 0.83$ , et  $(\sqrt{9} - \sqrt{4})/\sqrt{2} \approx 0.71$ . Donc, le point  $v$  devrait être conservé dans le squelette plutôt que le point  $w$ , d'après ce critère.

Pour pallier ce défaut, une stratégie développée dans [COU 07a] consiste à définir une fonction de priorité qui prend en compte à la fois la carte de distance et une fonction auxiliaire définie dans le voisinage de chaque

point du squelette détecté au cours de l'exécution de l'algorithme. Soit  $x$  un tel point, alors à tout voisin  $y$  de  $x$  qui est toujours dans  $X$  mais pas dans l'ensemble  $Y$ , nous associons la valeur  $p_y = D_X(x) + (D_X(y) - D_X(x))/d(x, y)$ , avec  $D_X(x) = \sqrt{D_X^2(x)}$ . La nouvelle fonction de priorité, pour tout point  $y$ , est définie par  $\min(p_y, D_X(y))$ . Nous voyons que  $(D_X(y) - D_X(x))/d(x, y)$  est la pente de  $xy$ , donc les voisins de  $x$  seront examinés par ordre croissant de pente, puisque la valeur  $p_y$  est toujours inférieure ou égale à la valeur  $D_X(y)$  (pour tous  $x, y$  dans  $\mathbb{Z}^2$  ou  $\mathbb{Z}^3$  avec  $x \neq y$ , on a  $d(x, y) \geq 1$ ).

Par exemple, dans la situation précédente, nous avons  $D_X(v) = \sqrt{8} \approx 2.83$ ,  $D_X(w) = 3$ ,  $p_v = \sqrt{4} + (\sqrt{8} - \sqrt{4})/1 = \sqrt{8}$  et  $p_w = \sqrt{4} + (\sqrt{9} - \sqrt{4})/\sqrt{2} \approx 2.71$ ; donc le point  $w$  sera sélectionné avant  $v$  avec cette stratégie. L'algorithme est décrit ci-dessous.

---

**Algorithme 2** : squelette euclidien
 

---

**Données** :  $X \subseteq E$ ,  $D_X$  la carte de distance euclidienne de  $X$ ,  $Y \subseteq X$

**Résultat** :  $Z$

$Z = X$ ;

$Q = \{(D_X(x), x)\}$ ; où  $x$  est un point quelconque de  $X \setminus Y$ ;

$R = \{(p_x, x)\}$ ; où  $x$  est un point quelconque de  $X \setminus Y$  adjacent à  $Y$ , et où

$p_x = \min\{D_X(z) + (D_X(x) - D_X(z))/d(x, z), z \in Y\}$ ;

**tant que**  $Q \neq \emptyset$  ou  $R \neq \emptyset$  **faire**

choisir  $(p, x)$  dans  $Q \cup R$  tel que  $p$  soit minimal;

$Q = Q \setminus \{(p, x)\}$ ;  $R = R \setminus \{(p, x)\}$ ;

**si**  $x \in Z \setminus Y$  **alors**

**si**  $x$  est simple pour  $Z$  **alors**

$Z = Z \setminus \{x\}$ ;

**sinon**

$Y = Y \cup \{x\}$ ;

$R = R \cup \{(p_y, y)\}$ ; avec  $y \in \Gamma(x) \cap (Z \setminus Y)$  et avec

$p_y = D_X(x) + (D_X(y) - D_X(x))/d(x, y)$ ;

---

La complexité de cet algorithme dépend de la structure de données utilisée pour représenter les ensembles  $Q$  et  $R$ . Plus précisément, cette structure de données doit permettre d'effectuer efficacement le choix de  $(p, x)$  au début de la boucle « tant que », ainsi que les insertions de nouveaux couples. Si l'on utilise par exemple un arbre binaire équilibré [COR 90], la complexité globale de l'algorithme est en  $O(n \log n)$ , où  $n$  est le nombre de points de l'image.

### 3.7. Conclusion

Dans ce chapitre nous avons introduit les notions de squelette, d'axe médian, de distance, de granulométrie et d'amincissement, ainsi que certains liens entre ces notions.

La définition d'une série de dilatations d'un point par une famille d'éléments structurants définit une famille granulométrique, par analogie avec le tamisage en géologie. Cette famille produit une notion de distance, et un opérateur : la fonction distance. Celle-ci, outre son utilité en soit permet de définir certains points caractéristiques d'une forme binaire : les centres de boules maximales. On a montré comment calculer ces centres, dans

le continu et le discret, au moyen du choix d'une famille granulométrique. La notion de distance peut être mise à profit, car ces points sont également particuliers sur la fonction distance.

Afin de retrouver certaines caractéristiques désirables du squelette continu dans le discret, en particulier la conservation de l'homotopie, nous avons été amenés à introduire les notions de points simples et d'amincissement homotopique. Ces notions s'appliquent dans les dimensions supérieures à 2, dans les images à niveaux de gris et dans des métriques euclidiennes discrètes.

Nous avons également évoqué les notions de robustesse, et d'invariance par rotation. Avec certaines précautions, il est désormais possible de définir un squelette efficace, et bien défini, possédant de bonnes propriétés de robustesse et d'invariance par rotation. L'étude de la stabilité des squelettes est toujours un sujet de recherche actif, avec des contributions provenant de différentes disciplines, en particulier la géométrie algorithmique. Attali, Boissonnat et Edelsbrunner recensent dans [ATT 09] les contributions les plus marquantes dans ce domaine. Citons en particulier le  $\lambda$ -*medial axis* introduit par Chazal et Lieutier dans [CHA 05], qui possède une propriété de continuité limitée à certaines valeurs du paramètre  $\lambda$ , et qui a été récemment adapté au cadre discret dans [CHA 09].



## Chapitre 4

# Morphologie et Algorithmes

### 4.1. Introduction

Dans ce chapitre nous abordons le problème très important de la mise en œuvre des opérateurs, filtres et méthodologies d'analyse d'images vues dans les chapitre précédents.

D'une manière générale, on présente souvent un nouvel opérateur par sa description mathématique ; toutefois, celle-ci n'est pas toujours simple à traduire sous forme algorithmique. Pourtant, un tel opérateur a normalement un intérêt au delà de sa simple description : on cherche souvent à l'utiliser dans la pratique. Pour cela, il faut réussir à franchir le pas entre les mathématiques pures et la pratique de la programmation. On doit alors exprimer la partie mathématique d'un opérateur sous une forme applicable : un algorithme. Remarquons que si la description mathématique sert à appréhender l'opérateur, cette forme est rarement une aide à l'implantation matérielle ou à l'implémentation logicielle. Ceci explique, sans doute, pourquoi de nombreuses méthodes permettant de réaliser les opérateurs morphologiques ont été proposées. L'évolution des processeurs fait aussi que de nouvelles méthodes sont encore imaginées pour des opérateurs connus depuis plusieurs décennies.

Plutôt que d'inventorier l'ensemble des algorithmes et des structures de données proposés pour les opérations morphologiques, ce qui donnerait un catalogue gigantesque et rébarbatif, nous nous concentrerons dans ce chapitre sur des aspects purement algorithmiques de la morphologie mathématique. Nous laisserons donc de côté certains aspects spécifiques à leur réalisation par logiciel et par matériel.

Un algorithme est, depuis les Babyloniens jusqu'à Ada Lovelace [STU 87], défini formellement comme une suite d'opérations permettant de résoudre un problème par un calcul. En morphologie mathématique, des filtres (ou opérateurs) s'appliquent généralement à des ensembles ou des fonctions et leur définition est donnée de façon formelle, mathématique.

Un algorithme est donc l'expression d'une solution effective permettant d'obtenir le résultat de l'application de ces opérateurs sur des données en entrée. La traduction en algorithme des mathématiques a pour but de faciliter la mise en œuvre d'un opérateur dans un ordinateur sous forme de programme tout en s'affranchissant de tout langage informatique. Aussi la description algorithmique doit conserver une forme abstraite pour que

la mise en oeuvre puisse être réalisée dans n'importe quel environnement (plateforme, langage, boîte à outils, bibliothèque).

Il existe en informatique théorique une modélisation abstraite et générique des ordinateurs réels, appelée la machine de Turing [TUR 36]. Dans cette modélisation, il est possible, mais peu pratique, de mettre en œuvre tout algorithme correct. Dans l'intérêt de la lisibilité et de la compacité de la description, nous utiliserons une notation relativement plus intuitive qui, en particulier, fait appel à des *structures de données* non triviales.

Dans un premier temps (section 4.2), nous allons discuter de la traduction sous forme informatique des structures et définitions utilisées en morphologie mathématique. Ensuite (section 4.3), nous allons couvrir les différents aspects liés aux algorithmes dans le cadre propre de la morphologie mathématique. Il sera notamment question de taxonomie, de compromis et de classes d'algorithmes. Nous intégrerons ces différents aspects en analysant, en section 4.4, le cas de la reconstruction morphologique. Enfin, la section 4.5 sera consacrée aux perspectives historiques et à un parcours de la bibliographie.

## 4.2. Traduction définition / algorithme

### 4.2.1. Structures de données

Avant de parler d'algorithme, il faut tout d'abord considérer quelles sont les données à traiter et comment elles se matérialisent concrètement lorsqu'elles ne sont plus seulement mathématiques.

Une image  $f$  est une fonction d'un espace  $\mathcal{E}$  vers un espace  $\mathcal{V}$ . Comme il est impossible dans la pratique de stocker et manipuler une infinité de données, ces espaces sont toujours discrétisés en  $E \subset \mathcal{E}$  et  $V \subset \mathcal{V}$  :

$$f : \begin{cases} E & \longrightarrow V \\ p & \longmapsto f(p). \end{cases}$$

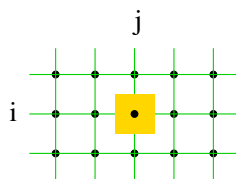


Figure 4.1. Points définis par une grille carrée et un pixel

$E$  est souvent doté d'une topologie discrète avec une notion de voisinage. Par convention d'écriture, un élément de  $E$  sera noté  $p$  (pour « point ») et un voisin de  $p$  sera noté  $n$ . Considérons le maillage inscrit dans  $\mathcal{E}$  où  $p$  est un nœud et où les arcs traduisent la notion de voisinage. Le cas le plus courant correspond à un échantillonnage régulier d'une partie de  $\mathcal{E}$ . Le maillage est alors régulier et les points appartiennent à une grille. La figure 4.1 montre une topologie classique d'image bi-dimensionnelle définie sur une grille carrée.

Un point  $p$  d'une telle image est facilement repérable grâce à deux indices entiers  $(i, j)$  et la définition de l'image  $f$ , association d'une valeur en chaque point, peut être représentée en mémoire par un tableau à 2 dimensions :  $f(p)$  s'apparente alors à  $f[i, j]$ . Cette représentation, très classique, permet de stocker et de



modifier la valeur de  $f$  en chaque point de façon indépendante. L'affectation à  $f$  d'une valeur  $v \in V$  au point  $p \in E$  sera notée sous la forme abstraite  $f(p) := v$  même si, dans les faits, il s'agira de réaliser  $f[i, j] := v$ .

Au regard de la réalisation,  $f$  n'est plus vraiment une fonction mathématique mais une variable (une mémoire) représentant une fonction à un instant  $t$  donné de l'exécution d'un algorithme. D'un point de vue formel, un algorithme crée une suite de fonctions  $f_t$  et chaque affectation exprime la modification de  $f_t$  en  $f_{t+1}$ . En langage algorithmique, la variable  $f$  permet d'abstraire l'existence de ces multiples fonctions ; c'est une sorte de fonction qui évolue au cours de l'algorithme. L'algorithmique commence à prendre ses distances vis-à-vis des mathématiques.

#### 4.2.2. Forme et étendue du domaine

En informatique, la représentation d'une image  $f$  par un tableau suppose que le domaine de définition de  $f$  est fini. Dans la pratique, une image bi-dimensionnelle est généralement rectangulaire et  $E$  s'apparente à un sous-ensemble de  $\mathbb{Z}^2$  ou de  $\mathbb{N}^2$ . Itérer sur les points d'une telle image revient à former deux boucles afin de parcourir les cellules du tableau, comme montré ci-dessous à gauche.

<pre> <b>for</b> <math>i := 1</math> <b>to</b> <math>n_{lignes}</math>   <b>for</b> <math>j := 1</math> <b>to</b> <math>n_{colonnes}</math>     ... // utilisation de <math>f[i, j]</math> </pre>	<pre> <b>for_all</b> <math>p \in E</math>   ... // utilisation de <math>f(p)</math> </pre>
---	--

Néanmoins, on préférera exprimer la forme abstraite du parcours, ci-dessus à droite, plus proche du mathématicien que de l'informaticien. Cette forme, plus générale que la précédente, ne fait pas apparaître d'hypothèses implicites concernant  $f$  : le domaine de définition de  $f$  peut ne pas être rectangulaire, ni même bi-dimensionnel. La formulation abstraite d'algorithmes permet de cacher les détails de représentation des données, permettant ainsi de se concentrer sur l'aspect opérationnel des algorithmes. En revanche, cela ne doit pas nous dispenser de préciser les structures de données sous-jacentes. Lorsqu'il s'agit d'évaluer la complexité d'un algorithme, on ne peut pas occulter les opérations élémentaires réalisées sur les données. Si le parcours d'un ensemble de  $N$  points, sans contrainte particulière sur l'ordre de parcours, a une complexité en  $O(N)$ , l'accès aléatoire à une valeur associée à un point  $p$  donné a une complexité dépendante de la structure de données utilisée.

Dans le cas d'une image dont les données sont stockées par un tableau en mémoire, on bénéficie d'un accès aléatoire en temps constant : la lecture et l'écriture de la valeur de l'image en un point  $p$  sont réalisées en  $O(1)$ , quelque soit le parcours des points dans lequel s'inscrit cet accès. La représentation des données d'une image sous forme de tableau est généralement passe-partout. Elle est également compacte puisque la mémoire utilisée pour la description de la structure en elle-même, en supplément des données, est très réduite. En revanche, cette représentation n'est pas toujours la meilleure. Considérons une image décrivant le contour d'un objet. L'intérêt d'encoder cette image sous la forme d'un tableau bi-dimensionnel est que l'on peut tester en temps constant si un point du plan discret appartient au contour. D'un autre côté, cet encodage est inefficace sur deux autres aspects. Tout d'abord, ce tableau doit au moins avoir la taille du rectangle englobant le contour, ce qui est beaucoup plus coûteux en mémoire que de stocker la liste des points du contour. Ensuite, le parcours des points du contour nécessite des recherches dans le tableau : pour l'obtention d'un premier point de contour, puis pour tout passage au point suivant, dans le voisinage 2D du point courant. Un algorithme qui fonctionne sur le contour d'un objet, par exemple une dilatation, peut avoir intérêt à utiliser une structure plus adaptée.

Au final, toute structure de données utilisée au sein d'un algorithme, qu'il s'agisse d'une image ou non, est susceptible d'affecter la complexité de cet algorithme.

### 4.2.3. Structure d'ensembles de points

La représentation classique d'une fonction sous la forme d'un tableau permet également de définir sur  $E$  des ensembles de points. En effet, lorsque l'espace destination de  $f$  est l'ensemble des booléens ( $V = \mathbb{B}$ ),  $f$  peut s'interpréter comme la fonction caractéristique d'un ensemble de points, précisément :  $F = \{p \in E \mid f(p) = true\}$ .  $f$  est alors une image binaire ; l'ensemble  $F \subset E$  représente un objet et son complémentaire  $E \setminus F$  constitue le fond de l'image. Le parcours des points de l'ensemble  $F$  s'effectue en parcourant le domaine  $E$  et en ne retenant que les points  $p$  tels que  $f(p) = true$ . Dans la suite, nous ne distinguerons pas, dans leur notation, un ensemble ( $F \subseteq E$ ) et sa fonction caractéristique ( $F : E \rightarrow \mathbb{B}$ ).

### 4.2.4. Raccourcis d'écriture

Au final, pour faciliter l'expression des algorithmes nous adopterons les raccourcis d'écriture suivants :

	Forme longue	Raccourcis
Parcours d'un ensemble de points $F \subseteq E$	<b>for_all</b> $p \in E$ , <b>if</b> $F(p) = true$ , ...	<b>for_all</b> $p \in F$ , ...
Copie d'une constante $c$ à toutes les positions d'une image $f$	<b>for_all</b> $p \in E$ , $f(p) := c$	$f := c \square$
Recopie du contenu de l'image $f_1$ dans l'image $f_2$	<b>for_all</b> $p \in E$ , $f_2(p) := f_1(p)$	$f_2 := f_1$

Avec de telles formes abstraites d'écriture, nous pouvons ignorer certaines difficultés pratiques. Par exemple, rien n'impose qu'une image bi-dimensionnelle soit rectangulaire (son domaine de définition peut-être quelconque). Lorsqu'il s'agit d'examiner les voisins d'un point, il faut pouvoir gérer le cas particulier des points situés sur le bord des images ; néanmoins la forme abstraite d'une itération sur le voisinage  $\mathcal{N}(p)$ , quelle que soit la structure de données utilisée pour décrire et gérer ce voisinage, s'écrira :

**for\_all**  $n \in \mathcal{N}(p)$   
...

Ainsi le lecteur d'un algorithme pourra se focaliser sur l'aspect algorithmique et ne sera pas pollué par des détails d'implémentation.

Enfin, pour préciser qu'un parcours des points d'une image s'effectue dans l'ordre vidéo classique (pour chaque ligne de haut en bas, pour chaque colonne de gauche à droite), nous utiliserons le symbole  $\triangleright$ . L'ordre inverse, anti-vidéo (pour chaque ligne de bas en haut, pour chaque colonne de droite à gauche), sera précisé par le symbole  $\triangleleft$ .

### 4.2.5. De la définition à la réalisation

Comme vu précédemment, si les opérateurs morphologiques sont très souvent exprimés sous la forme d'une expression mathématique, leur traduction en algorithme n'est pas toujours simple et directe. De plus, dans les

cas favorables où l'on dispose d'une telle traduction, cette dernière est rarement la traduction la plus appropriée, au sens de l'efficacité du programme résultant. En mathématique, on se préoccupe essentiellement du fait que l'algorithme est correct plutôt qu'efficace dans la pratique.

Nous allons prendre l'exemple de la dilatation binaire. La définition d'une dilatation peut s'opérer de plusieurs manières différentes :

$$X \oplus B = \bigcup_{b \in B} X_b \quad (4.1)$$

$$= \{ x + b \in E \mid x \in X, b \in B \} \quad (4.2)$$

$$= \{ p \in E \mid \exists b \in B, p - b \in X \} \quad (4.3)$$

La définition (4.1) peut se traduire par l'algorithme (1) donné en figure 4.2 (lignes de 1 à 16). En termes informatiques, le point d'entrée est SETDILATION, et utilise la fonction TRANSLATE. En termes algorithmiques, on constate que la méthode reprend exactement la définition, ce qui suffit à justifier que l'algorithme est correct.

Dans une unité de calcul, un facteur de coût essentiel est l'accès à la mémoire, en lecture mais surtout en écriture. On ne va compter ici que les écritures mémoires (affectations) à *true* pour simplifier. On constate que cette première version nécessite  $|B| \times |X| \times 2$  affectations, où la notation  $| \cdot |$  désigne le nombre d'éléments d'un ensemble.

La seconde version, qui est issue de la définition (4.2), consomme deux fois moins d'affectations que la précédente. Elle est représentée en figure 4.2 par la routine DILDIRECT (lignes de 34 à 47). Quant à l'algorithme issu de la formule (4.3), il est nettement plus efficace car il ne consomme que  $|X \oplus B|$ . On a ici remplacé un produit par une somme. Ce dernier algorithme, donné en figure 4.2 par la routine DILREVERSE (lignes de 50 à 66), est l'implémentation classique de la dilatation telle qu'on la rencontre dans les bibliothèques logicielles de traitement d'images.

Sur ces exemples simples, on voit bien la distance entre la formulation concise issue des définitions et la mise en œuvre effective en algorithmes, nettement plus verbeuse.

L'exemple de la dilatation est essentiellement utile pour réaliser la distance qui peut séparer définition et algorithme. Nous allons maintenant donner un exemple d'algorithme plus complet pour illustrer diverses stratégies algorithmiques pour un même opérateur.

La plupart des algorithmes de morphologie mathématique sont de complexité pseudo-polynomiale. Par exemple, l'algorithme trivial de dilatation (une double boucle) a une complexité de  $\mathcal{O}(N \times M)$  avec  $N$  le nombre de points de l'image et  $M$  le nombre de points de l'élément structurant.

### 4.3. Taxonomie des algorithmes

Dresser une taxonomie des algorithmes utilisés en morphologie mathématique est une tâche difficile pour plusieurs raisons. D'une part, il est impossible de citer en quelques pages ne serait-ce que parce que la communauté scientifique continue à proposer des nouveaux algorithmes, même pour opérateurs connus depuis les

```

1 // algorithme (1)
2
3 SETDILATION( $X$  : Image of  $\mathbb{B}$ ,
4              $B$  : Set of Point)
5    $\rightarrow$  Image of  $\mathbb{B}$ 
6 begin
7   data  $X_b, U$  : Image of  $\mathbb{B}$ 
8   // initialisation à l'ensemble vide
9    $U := \text{false}$   $\square$ 
10  for_all  $b \in B$ 
11    // calcul de  $X_b$ 
12     $X_b := \text{TRANSLATE}(X, b)$ 
13    // mise à jour de  $U$ 
14     $U := \text{UNION}(U, X_b)$ 
15  return  $U$ 
16 end
17
18 TRANSLATE( $X$  : Image of  $\mathbb{B}$ ,
19            $b$  : Point)
20    $\rightarrow$  Image of  $\mathbb{B}$ 
21 begin
22   data  $O$  : Image of  $\mathbb{B}$ 
23   // initialisation à l'ensemble vide
24    $O := \text{false}$   $\square$ 
25   // calcul de l'ensemble
26   for_all  $p \in X$ 
27     if  $p + b \in E$ 
28        $O(p + b) := \text{true}$ 
29   return  $O$ 
30 end
31
32
33
34 // algorithme (2)
35
36 DILDIRECT( $X$  : Image of  $\mathbb{B}$ ,
37            $B$  : Set of Point)
38    $\rightarrow$  Image of  $\mathbb{B}$ 
39 begin
40   data  $O$  : Image of  $\mathbb{B}$ 
41    $O := \text{false}$   $\square$  // initialisation
42   for_all  $p \in E$ 
43     for_all  $b \in B$ 
44       if  $X(p) = \text{true}$  and  $p + b \in E$ 
45          $O(p + b) := \text{true}$ 
46   return  $O$ 
47 end
48
49
50 // algorithme (3)
51
52 DILREVERSE( $X$  : Image of  $\mathbb{B}$ ,
53             $B$  : Set of Point)
54    $\rightarrow$  Image of  $\mathbb{B}$ 
55 begin
56   data  $O$  : Image of  $\mathbb{B}$ 
57   for_all  $p \in E$ 
58     for_all  $b \in B$ 
59       if  $p - b \in E$  and  $X(p - b) = \text{true}$ 
60         // existence
61          $O(p) := \text{true}$ 
62         goto next
63      $O(p) := \text{false}$  // pas d'existence
64   label next
65   return  $O$ 
66 end

```

Figure 4.2. Dilatations ensemblistes

années 70 ! D'autre part, il n'existe pas vraiment un ensemble à la fois réduit et satisfaisant de critères sur lesquels pourraient s'appuyer une telle taxonomie. Enfin, il n'y a pas d'algorithme universel pour l'ensemble des opérateurs morphologiques. Chaque opérateur amène son lot d'algorithmes, de particularités, de compromis, de structures de données, etc.

#### 4.3.1. Critères de taxonomie

Les critères de taxonomie qui permettraient de classer les algorithmes sont multiples et variés. Pour illustration, nous en donnons ci-après une première liste, non exhaustive :

- le type utilisé de structures auxiliaires de données (file, arbre ou autre) ;
- la nature du parcours des points dans l'image ;
- la complexité de l'algorithme ;

- l’occupation mémoire requise ;
- les propriétés remarquables de l’algorithme ;
- les conditions (et donc les limites) d’utilisation de l’algorithme ;
- la classe d’opérateurs ou de filtres concernée ;
- la portée de l’algorithme ;
- son intention ;
- son sujet (les données sur lesquelles il porte).

De plus, les contraintes que subit l’utilisateur et qui sont liées à son contexte fournissent également d’autres critères possibles de taxonomie : son domaine d’application, la nature des données, ses objectifs, etc.

À défaut de parcourir cette liste en détail, notons que certains critères permettent d’opposer certains algorithmes tandis que d’autres ne permettent que de les distinguer. Les deux tableaux ci-dessous donnent respectivement quelques exemples de ces deux cas de figure.

critère : portée de l’algorithme	
<b>spécifique</b> décomposition de l’élément structurant (afin d’accélérer le calcul des dilatations et érosions) ;	<b>générale</b> algorithme de Dijkstra (pour le calcul d’une fonction distance) ; algorithme de Viterbi (pour le filtrage par élagage décrit au chapitre ??.3 du volume 1) ; algorithme de Prim ou de Kruskal (pour la segmentation par calcul de l’arbre de poids minimum décrite au chapitre ?? du volume 1).

critère : propriété	
<b>parallèle</b> algorithme classique de dilatation ou d’érosion par élément structurant (la version (3) décrite plus loin en section 4.2.5) ; détection des points simples	<b>séquentiel</b> fonction distance de chanfrein (Cf. chapitre ??.4.3 du volume 1) ; filtres alternés séquentiels (Cf. chapitre ??.2.7 du volume 1)

critère : nature des données	
<b>faible quantification</b> <sup>1</sup> tri par distribution ou par base ; utilisation des représentations par arbre (Cf. chapitre ??.2 du volume 1)	<b>forte quantification</b> tri rapide, par tas

critère : parcours des points	
<b>par front</b> dilatation par distance ; la majorité des algorithmes de calcul de la ligne de partage des eaux (Cf. chapitres ??.5 et 4 du volume 1)	<b>par passe</b> dilatation triviale ; transformée en tout ou rien (Cf. chapitre ??.1.3 du volume 1)

Le second tableau ci-dessous illustre des critères permettant de discriminer les algorithmes.

critère : structures auxiliaires
tableaux, files, queues de priorité, arbres, graphes, etc.

critère : intention
simplification des données, obtention d'une transformée, calculs / estimations sur les données, partitionnement, etc.

critère : sujet de l'algorithme
pixels, textures, objets, régions, contours, etc.

#### 4.3.2. *Compromis*

La mise en œuvre d'une chaîne de traitement d'images n'est qu'un cas particulier de la réalisation de calculs scientifiques par l'informatique. Lorsqu'un module de cette chaîne est un opérateur morphologique, il ne peut pas échapper aux lois de ce cadre plus général. En particulier, une contrainte que l'on rencontre bien souvent est de trouver un compromis acceptable entre les trois notions antagonistes suivantes :

- la performance attendue en temps d'exécution. Certaines applications nécessitent de tourner en temps-réel ; à l'opposé, d'autres applications peuvent se permettre de prendre tout leur temps. Outre la question du temps nécessaire à l'exécution de l'opération se pose celle de sa variabilité. Une implantation matérielle préférera un temps d'exécution constant, quitte à allonger, dans l'absolu, le temps de calcul utilisé ;
- le stockage. Les ressources utilisées en termes d'utilisation mémoire et/ou d'espace disque sont généralement limitées tandis que les données à traiter, pour une même application, sont généralement d'un ordre de grandeur de taille constant. Le choix d'un algorithme et des structures de données qu'il utilise est alors souvent conditionné par les ressources de stockage disponibles ;
- la précision du résultat. Le résultat d'un calcul peut être soit exact soit approché et, dans ce dernier cas, un niveau donné de précision est généralement attendu. Dans la pratique, on peut souvent se contenter d'un résultat approché.

Suivant les contraintes imposées par le contexte applicatif, la mise en œuvre d'un opérateur morphologique doit se positionner dans le triangle formé par ces trois notions antagonistes, illustré en figure 4.3. Privilégier la vitesse d'exécution, par exemple, se fait *a priori* en sacrifiant de la précision et/ou des ressources. Notons que les architectures modernes permettent la réalisation en temps réel de nombreuses opérations morphologiques. Dans ce cas, la notion de précision n'est plus pertinente.

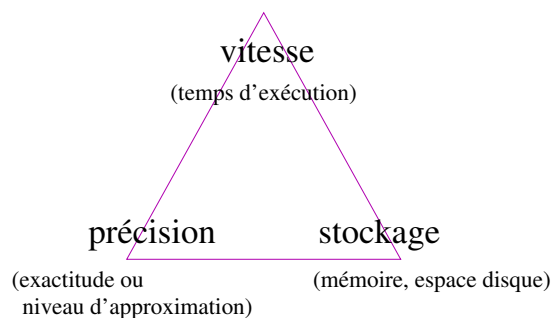


Figure 4.3. *Triangle de compromis*

<pre> 1 POINT_WISE(<i>f</i> : Image, 2           <i>h</i> : Fonction) 3   → <i>o</i> : Image 4 5 <b>begin</b> 6   <b>for_all</b> <i>p</i> ∈ <i>E</i> 7     <i>o</i>(<i>p</i>) := <i>h</i>(<i>f</i>(<i>p</i>)) 8 <b>end</b> </pre>	<pre> 1 SLIDING_WINDOW(<i>f</i> : Image, 2                <i>w</i> : Window, 3                <i>h</i> : Fonction) 4   → <i>o</i> : Image 5 <b>begin</b> 6   <b>for_all</b> <i>p</i> ∈ <i>E</i> 7     <i>o</i>(<i>p</i>) = <i>h</i>( { <i>f</i>(<i>q</i>)   <i>q</i> ∈ <i>w</i>(<i>p</i>) } ) 8 <b>end</b> </pre>
---	---

Figure 4.4. Deux canevas : « point-à-point » et « fenêtre glissante »

### 4.3.3. Classes d'algorithmes et canevas

Un algorithme agissant sur des images doit généralement s'appuyer sur les parties suivantes :

- un ou plusieurs parcours d'image ;
- des structures de données auxiliaires (d'autres images, ou d'autres structures, classiques ou non) ;
- une logique de traitement, en plusieurs phases, souvent initialisation, boucles, finalisation ;
- souvent une notion, importante en morphologie, de voisinage et d'inspection autour d'un point.

Considérons l'ensemble des algorithmes connus qui permettent de mettre en œuvre les opérateurs morphologiques. Nous pouvons alors observer que plusieurs groupes d'algorithmes se forment tels qu'au sein de chaque groupe, les algorithmes se ressemblent. Les algorithmes d'un groupe donné partagent un même schéma algorithmique, c'est-à-dire un même enchaînement d'opérations, une même utilisation de structures de contrôle (comme les boucles), ainsi que les mêmes données auxiliaires. Nous sommes en présence d'une classe d'algorithmes.

Nous nommerons *canevas* la représentation d'une classe d'algorithmes sous la forme d'un « algorithme à trous. » Ce canevas, ou motif algorithmique, est exactement comparable à un patron en couture, où le choix de la matière, de la couleur et des ornements reste encore à définir. La figure 4.4 montre deux canevas très classiques de traitement d'images. À gauche, un traitement point-à-point sur les valeurs des pixels d'une image ; la valeur au point  $p$  dans l'image de sortie  $o$  est obtenue en appliquant une fonction  $h$  à la valeur de ce même point  $p$  dans l'image d'entrée  $f$  :  $o(p) := h(f(p))$ . Avec  $h = C$  (fonction de complémentation), ce canevas devient l'opérateur de complémentation ; avec  $h = lum$  (fonction de luminance), il s'agit alors d'une conversion d'image couleur à image à niveaux de gris.

Le canevas de droite, figure 4.4, évalue chaque valeur  $o(p)$  de l'image de sortie en fonction de l'ensemble des valeurs de l'image d'entrée appartenant à une fenêtre  $w$  centrée en  $p$ . C'est le canevas des convolutions avec  $h(\{f(q) \mid q \in w(p)\}) = \sum_q g(p-q) f(q)$ , où  $g$  est la fonction noyau de la convolution, et de la dilatation sur les fonctions avec  $h = \vee$  (supremum).

L'intérêt des canevas algorithmiques est multiple :

- ils sont généraux. Chaque canevas n'est pas spécifique à un traitement en particulier. Au contraire, chaque canevas est transposable afin de réaliser d'autres traitements ; pour cela, il suffit de définir sa partie variable ( $h$  dans les exemples ci-avant) ;
- ils sont abstraits. Leur expression ne fait aucune hypothèse qui limiterait leur utilisation à certaines natures de données. Par exemple, la présence d'une double boucle sur les coordonnées des points dans les canevas de

la figure 4.4 signifierait implicitement qu'ils sont restreints aux images bi-dimensionnelles, ce qui est clairement faux ;

– ils sont pédagogiques. Ils matérialisent un méta-algorithme dont la compréhension d'un point de vue algorithmique entraîne celle des opérateurs qui vont en être dérivés.

#### 4.4. Exemple de la reconstruction géodésique

Les diverses classes d'algorithmes que nous allons présenter ci-après n'ont pas été proposées simultanément. Elles se situent dans une perspective historique dont nous parlerons dans la section 4.5.

En guise d'illustration de ces classes, nous allons montrer plusieurs façons de mettre en œuvre, en algorithmes, un même opérateur morphologique : la reconstruction géodésique par dilatation d'une fonction (une image en niveaux de gris par exemple) [VIN 93b].

Cet exemple en forme d'exercice de style est significatif ; il permet d'illustrer une grande partie des approches qui sont utilisées par d'autres opérateurs.

##### 4.4.1. La version mathématique : algorithme parallèle

Dans un premier temps, nous pouvons mettre en œuvre la reconstruction géodésique par dilatation en partant de sa définition mathématique :

$$\mathcal{R}_g^\delta(f) = \lim_{n \rightarrow \infty} \delta_g^n(f) = \delta_g^\infty(f)$$

où :

$$\begin{aligned} \delta_g^1(f) &= \delta(f) \wedge g, \\ \delta_g^{n+1}(f) &= \delta(\delta_g^n(f)) \wedge g. \end{aligned}$$

Cette définition de la reconstruction est à rapprocher de celle donnée au chapitre ??2.2 du précédent ouvrage. Par rapport à cette dernière, ensembliste, il s'agit d'une généralisation aux fonctions. Ici  $\delta$  est la dilatation géodésique :  $\delta(f) = f \oplus B$ , où  $B = \mathcal{N}(0) \cup \{0\}$ , avec  $\mathcal{N}$  le voisinage considéré et 0 l'origine de l'espace.

Une hypothèse implicite pour que cet opérateur, ainsi défini, ait un sens est que l'on doit avoir  $f \leq g$ . Dit autrement, la fonction marqueur  $f$  à dilater doit être « sous » la fonction masque  $g$ . Comme dans le cas ensembliste, le but de la reconstruction sur les fonctions consiste à retrouver le contenu de  $g$  à partir d'une image  $f$ .

La définition de cette reconstruction est, en soi, déjà un algorithme : il s'agit d'itérer jusqu'à convergence une dilatation géodésique suivie du conditionnement.

La récursion s'implante naturellement à l'aide d'une boucle et l'algorithme se termine par la stabilité du résultat (aucune modification durant la dernière itération), précisément sur une image d'extension finie. L'algorithme a une convergence garantie, néanmoins très lente en pratique. En effet, au cours des passes successives, on considère de nouveau des zones d'images qui ont déjà atteint une condition locale de convergence. Cet algorithme est illustré par la routine RD\_PARALLEL en figure 4.5.



```

1  RD_PARALLEL(f : Image,
2             g : Image)
3  → o : Image
4  begin
5
6  data
7    o' : Image
8    stability :  $\mathbb{B}$ 
9
10 // initialisation
11 o := f
12
13 // itérations
14 repeat
15   o' := o // échange
16
17 // opère
18 for_all p ∈ E
19   o(p) := max{ o'(q) |
20                q ∈  $\mathcal{N}(p) \cup \{p\}$  }
21
22 // conditionne
23 for_all p ∈ E
24   o(p) := min{ o(p), g(p) }
25
26   stability := (o = o')
27 until stability
28 return o
29
30 end
31
33 RD_SEQUENTIAL(f : Image,
34               g : Image)
35 → o : Image
36 begin
37
38 data
39   o' : Image
40   stability :  $\mathbb{B}$ 
41
42 // initialisation
43 o := f
44
45 // itérations
46 repeat
47   o' := o // mémorisation
48
49 // passe 1
50 for_all p ∈ E ▷
51   o(p) := min{ max{ o(q) |
52                  q ∈  $\mathcal{N}^-(p) \cup \{p\}$  }, g(p) }
53
54 // passe 2
55 for_all p ∈ E ◁
56   o(p) := min{ max{ o(q) |
57                  q ∈  $\mathcal{N}^+(p) \cup \{p\}$  }, g(p) }
58
59   stability := (o = o')
60 until stability
61 return o
62
63 end

```

**Figure 4.5.** Canevas de reconstruction (partie 1/2). L'algorithme parallèle (à gauche) et l'algorithme séquentiel (à droite) sont décrits respectivement en sections 4.4.1 et 4.4.2

#### 4.4.1.1. Algorithmes similaires

Ce type d'algorithme, « modification itérative jusqu'à stabilité, » n'est pas propre au domaine de la morphologie mathématique ; il est utilisé, entre autres, pour réaliser des diffusions.

La complexité d'une passe est ici pseudo-polynomiale avec le nombre de points de l'image et la connexité. La complexité de l'algorithme, dans le pire des cas (une image d'une courbe de Peano), est polynomiale en  $\mathcal{O}(M \times N^2)$ .

En revanche, cet algorithme est facilement parallélisable puisque les calculs en chaque point à l'intérieur des boucles ne dépendent que de l'image de travail obtenue à l'itération précédente. Contrairement aux autres algorithmes présentés par la suite, il n'existe en effet aucune dépendance entre le calcul courant au point *p* et les calculs aux points voisins lors d'une même itération.

#### 4.4.2. Algorithme séquentiel

Une amélioration de la complexité est possible en remarquant que la reconstruction peut s'exprimer de manière séquentielle. Dans la version précédente parallèle, on calculait chaque dilatation indépendamment (lignes 18 et 20). L'image  $o$  se calcule à partir de l'image  $o'$  obtenue à l'itération précédente.

Dans cette nouvelle version, il devient inutile de calculer une image  $o'$  : la dilatation a lieu en place : elle est calculée dans un voisinage et est inscrite immédiatement dans l'image de travail  $o$ . Une modification réalisée au point  $p$  peut se propager au cours de la même passe sur l'image.

Dans l'algorithme tel qu'il est donné, pour garder un pendant avec l'algorithme parallèle, on a gardé une copie dans une image  $o'$  pour tester la condition de stabilité, mais on peut aussi compter au cours de chaque passe le nombre de pixels modifiés. S'il est nul après une passe (une boucle dans chaque sens), la stabilité est atteinte.

La complexité de cet algorithme n'est théoriquement pas meilleure que dans le cas précédent : on peut en effet exhiber un objet binaire à reconstruire en forme de courbe de PEANO qui exige un nombre de passes sur l'image proportionnel au nombre de points de la courbe. En revanche, dans le cas des objets convexes, on peut obtenir une reconstruction complète en deux passes sur l'image. Dans la pratique, on constate que les images naturelles comportent de nombreuses parties localement convexes, qui sont reconstruites efficacement.

##### 4.4.2.1. Algorithmes similaires

La classe des algorithmes similaires à celui-ci est vaste. Elle comprend en particulier les calculs de carte de distance discrètes, telle la distance de chanfrein, et la distance pseudo-euclidienne de Danielsson [DAN 80a].

#### 4.4.3. Algorithme à base de file d'attente

Dans cette version de l'algorithme de reconstruction, on utilise une structure de données supplémentaire : la file d'attente simple (*FIFO* en anglais : *First In, First Out* ; PAPS en français : Premier Arrivé, Premier Servi). L'idée générale est de réaliser la dilatation à l'aide d'un front qui se propage dans l'image, tout en étant conditionnée par l'image  $g$  (ligne 92).

L'avantage premier de cette approche est la simplicité de la formulation : la dilatation s'effectue de manière indépendante d'un nombre de passes. On évite ainsi potentiellement de grands parcours sur l'image qui ne sont plus sujets à modification.

La plupart des algorithmes basés sur des files d'attente utilisent le même schéma, donné ci-dessous :

- on initialise la file d'une certaine manière ;
- tant que la file n'est pas vide :
  - 1) on retire un point de la file,
  - 2) on effectue le traitement en ce point,
  - 3) on ajoute conditionnellement les voisins de ce point dans la file.

Ce schéma correspond exactement à un parcours en largeur d'abord dans le graphe de voisinage de l'image. D'autres parcours sont possibles en intervertissant les étapes 1 et 3, ce qui donne localement un parcours en profondeur d'abord.

```

65 RD_QUEUE_BASED(f : Image,
66                 g : Image)
67   → o : Image
68 begin
69   data
70     q : Queue of Point
71     M : Image
72
73   // initialisation
74   M := REGIONAL_MAXIMA(f)
75   for_all p ∈ M
76     for_all n ∈  $\mathcal{N}(p)$ 
77       if n ∉ M
78         q.PUSH(p)
79   o := f
80
81
82
83
84
85
86
87   // propagation
88   while not q.EMPTY()
89     p := q.FIRST()
90     for_all n ∈  $\mathcal{N}(p)$ 
91       if  $o(n) < o(p)$  and  $o(n) \neq g(n)$ 
92          $o(n) := \min\{o(p), g(n)\}$ 
93         q.PUSH(n)
94
95   return o
96 end

```

```

98 RD_HYBRID(f : Image,
99           g : Image)
100  → o : Image
101 begin
102   data
103     q : Queue of Point
104
105   // initialisation
106   o := f
107
108   // une séquence de 2 passes
109   for_all p ∈  $E \triangleright$ 
110      $o(p) := \min\{ \max\{o(q) \mid$ 
111                  $q \in \mathcal{N}^-(p) \cup \{p\}\}, g(p) \}$ 
112   for_all p ∈  $E \triangleleft$ 
113      $o(p) := \min\{ \max\{o(q) \mid$ 
114                  $q \in \mathcal{N}^+(p) \cup \{p\}\}, g(p) \}$ 
115   // avec mise en queue
116   for_all n ∈  $\mathcal{N}^+(p)$ 
117     if  $o(n) < o(p)$  and  $o(n) < g(n)$ 
118       q.PUSH(p)
119
120   // propagation
121   while not q.EMPTY()
122     p := q.FIRST()
123     for_all n ∈  $\mathcal{N}(p)$ 
124       if  $o(n) < o(p)$  and  $o(n) \neq g(n)$ 
125          $o(n) := \min\{o(p), g(n)\}$ 
126         q.PUSH(n)
127
128   return o
129 end

```

**Figure 4.6.** Canevas de reconstruction (partie 2/2). Les algorithmes à base de file d'attente (à gauche) et hybride (à droite) sont décrits respectivement en sections 4.4.4 et 4.4.3

Ici, l'initialisation part d'une détection des maxima régionaux. Le contour extérieur de ces maxima est mis dans la file. Ces maxima sont propagés par la file d'attente. L'opération réalisée par la file d'attente n'est pas une dilatation complète, au sens où à aucun moment l'opération de propagation ne correspond à une dilatation par élément structurant, même élémentaire. On dilate simplement les maxima. L'opération réalisée est néanmoins une dilatation algébrique (Cf. chapitre ?? du volume 1). De ce fait, l'opération globale incluant le masque *g* est identique à la reconstruction définie plus haut [VIN 93b].

Le cœur de l'algorithme, présenté en figure 4.6, colonne de gauche, réside dans la propagation : lignes 87–93. On constate facilement qu'on ne traite qu'une unique fois un point *p* de l'image – la variable du même nom dans les algorithmes décrits. Ce n'est pas le cas des algorithmes précédents. Le cœur de l'algorithme est donc de complexité linéaire.

En revanche, la partie d'initialisation exige de calculer les maxima régionaux (ligne 74) ce qui est relativement coûteux : l'opération est équivalente à un étiquetage, qui est de complexité quasi-linéaire par union-find [TAR 75].

Plus généralement, l'utilisation d'une structure à accès aléatoire, comme une file d'attente ou une pile, s'avère efficace dans la mesure où une étape préparatoire extrait les bonnes informations. On peut à nouveau faire le lien avec la question de la redondance. Dans un processus de dilatation, morphologique ou géodésique, l'information pertinente est la position et la valeur du maximum local. Les alternatives consistent soit à exploiter les informations pertinentes sans se soucier de leur position et de leur valeur –il s'agit d'un algorithme aveugle au contenu–, soit à trouver préalablement les informations utiles et à les propager dans un second temps. Parmi les approches algorithmiques les plus rapides (mais aussi les plus complexes), on trouve celles qui sélectionnent les informations locales et les propagent directement, en intégrant les phases de détection et de propagation lors d'un unique parcours de l'image. L'algorithme proposé dans [VAN 05] pour l'ouverture morphologique est de ce type. On analyse les valeurs et, lorsqu'un extremum local est trouvé, la détection laisse place à la propagation jusqu'à trouver un nouvel extremum local.

Nous verrons dans la section suivante une reconstruction plus efficace.

#### 4.4.3.1. Remarques

**Algorithmes similaires :** parmi les algorithmes similaires en terme d'implémentation on peut citer :

- pour l'exploration en largeur d'abord : fonction distances [RAG 93], SKIZ, squelettisation [VIN 91a], dilatations ordonnées [ZAM 80] ;
- et pour l'exploration en profondeur d'abord : le calcul de l'arbre des composantes [SAL 92].

**Utiliser une pile au lieu d'une file :** l'usage d'un algorithme à pile (LIFO, PADS), au lieu d'une file, peut s'avérer pertinent lorsque la structure de pile est utilisée pour stocker des informations relatives à certaines données. Néanmoins, l'utilisation d'une pile est souvent à déconseiller. Le comportement d'un algorithme à pile est récursif puisqu'on empile les traitements en attente avant de les effectuer. Aussi, même si l'algorithme est théoriquement correct, il peut dans certains cas devenir inefficace, car pénalisé par la gestion de la structure de données, dont la taille peut croître jusqu'à la taille de l'image elle-même.

**Utilisation d'une file à priorité :** une file d'attente simple ne gère qu'un aspect des objets qui y résident : leur ordre d'entrée dans la pile, qui correspond souvent à un ordonnancement spatial dans les images. Il est utile parfois d'utiliser une priorité plus générale, tenant compte par exemple des niveaux de gris. Une structure spéciale très efficace pour les images à petit nombre de niveaux de gris a été proposée par F. Meyer dans le cadre d'un algorithme de la LPE : la file d'attente hiérarchique [MEY 91]. Plus généralement, on peut utiliser une *file à priorité*, structure classique en algorithmique, qui peut être implémentée par un *tas* ou un arbre binaire complet par exemple.

**Complexité :** suivant les algorithmes et les opérations qu'ils utilisent, certaines structures de données sont plus ou moins appropriées. Une étude des différentes structures de données classiques utilisables en morphologie a été proposée par E. Breen and D. Monro [BRE 94], démontrant bien la différence entre la théorie et la pratique dans ce domaine. En particulier les files de Fibonacci [FRE 87] bien que théoriquement efficaces se révèlent plutôt lentes en pratique. En terme de recommandation, lorsque l'on cherche une file à priorité efficace utilisable avec tout type de donnée, dont le tri est stable<sup>2</sup>, les *splay-queues* peuvent être un bon choix [SLE 85].

---

2. ce qui signifie que l'ordre d'entrée dans la pile est préservé, ce qui est souvent souhaitable en imagerie

#### 4.4.4. *Algorithme hybride*

Dans cet algorithme, en figure 4.6, colonne de droite, on retrouve d'abord une partie séquentielle limitée à deux passes. Durant celles-ci, la reconstruction est menée dans les régions convexes de l'image. Dans une seconde partie, on retrouve la propagation de l'algorithme à base de queue qui complète la reconstruction jusqu'à convergence.

L'avantage de cette méthode tient en plusieurs points. La recherche des maxima régionaux est inutile, on placera sur la queue la frontière obtenue à la suite des deux passes séquentielles. D'autre part, une partie majeure de la reconstruction, dans le cas habituel des images naturelles est effectuée au cours des passes séquentielles. La propagation finale, plus coûteuse à cause de la gestion de la structure de données de la queue, n'est menée que sur une petite portion de l'image.

Les algorithmes hybrides pour lesquels il existe une réelle synergie entre les différentes approches sont relativement rares dans la littérature.

##### 4.4.4.1. *Remarques*

On retrouve les cas défavorables des algorithmes précédents : courbes fractales par exemple. Dans ce cas, la file reste de taille faible mais on fait beaucoup de boucles. Une façon de rendre cet algorithme plus performant est d'utiliser une file d'attente implémentée par un tableau circulaire (beaucoup plus compacte en mémoire et plus efficace lors des insertions et suppressions d'éléments).

Indépendamment du dernier cas de l'utilisation des files d'attente à priorité, cet exemple met en lumière que l'algorithme effectue une série d'opérations sur l'image et sur la structure de données. Certains auteurs étudient la complexité de leur algorithme en termes de manipulation des données de l'image, oubliant qu'il y a un coût lié à l'utilisation d'une structure de données additionnelle. Sous sa forme simple, cette structure de données est par exemple l'utilisation d'une mémoire-tampon, dont l'utilisation à un coût non négligeable, notamment en écriture. Mais caractériser finement le coût des structures additionnelles est une opération délicate, notamment parce qu'elle est liée au matériel utilisé.

#### 4.4.5. *Algorithme par Union-Find*

L'algorithme d'Union-Find est une recherche de classes d'équivalences sur un graphe. Il est présenté en figure 4.7. L'algorithme est relativement complexe mais on peut sinon l'expliquer, du moins donner sa teneur générale. Il est constitué de trois phases : une initialisation, une étape d'union et une étape d'étiquetage (find).

##### 4.4.5.1. *Idée générale de l'algorithme*

Le cœur de l'algorithme est un changement de représentation de l'image : on passe du domaine des pixels, sans notion de connection autre que locale, à une structure d'arbre, dont la racine est dans les parties basses de l'image et les feuilles dans les parties hautes.

En phase d'initialisation, tous les points de l'image  $g$  sont triés par niveau de gris décroissant dans  $S$  (ligne 169).

Dans la phase d'union, on parcourt les points de l'image dans cet ordre en parcourant  $S$  séquentiellement. Tout point considéré est soit isolé, auquel cas il forme un singleton (un nouveau maximum régional), ou bien connecté à un maximum régional de  $g$ , auquel cas il forme la nouvelle racine de l'arbre associé. Un point

```

130 MAKE_SET( $p$  : Point)
131 begin // crée le singleton  $\{p\}$ 
132    $parent(p) := p$ 
133 end
134
135 IS_ROOT( $p$  : Point)  $\rightarrow \mathbb{B}$ 
136 begin // teste si  $p$  est un représentant
137   return  $parent(p) = p$ 
138 end
139
140 FIND_ROOT( $p$  : Point)  $\rightarrow$  Point
141 begin // trouve le représentant de  $p$ 
142   if IS_ROOT( $p$ )
143     return  $p$ 
144   else
145      $parent(p) :=$  FIND_ROOT( $parent(p)$ )
146   return  $parent(p)$ 
147 end
148
149 DO_UNION( $n$  : Point,  $p$  : Point)
150 begin // unit deux arbres
151    $r :=$  FIND_ROOT( $n$ )
152   if  $r \neq p$ 
153     if  $g(r) = g(p)$  or  $g(p) \geq o(r)$ 
154        $parent(r) := p$ 
155        $o(p) := \max(o(r), o(p))$ 
156     else
157        $o(p) := \text{MAX}$ 
158   end
159
160 RD_UNION_FIND( $f$  : Image,
161                $g$  : Image)
162    $\rightarrow o$  : Image
163 begin
164   data
165      $parent$  : Image of Point
166      $S$  : Array of Point
167
168   // initialisation
169    $o := f$ 
170    $S :=$  SORT( $g$ ) // suivant  $\triangleright$  et  $g(p) \downarrow$ 
171
172   // première phase
173   for_all  $p \in S$ 
174     MAKE_SET( $p$ )
175     for_all  $n \in \mathcal{N}(p)$  if DEJA_VU( $n$ )
176       DO_UNION( $n, p$ )
177
178   // seconde phase
179   for_all  $p \in S^{-1}$ 
180     if is_root( $p$ )
181       begin
182         if  $o(p) = \text{MAX}$ ,  $o(p) := g(p)$ 
183       end
184     else
185        $o(p) := o(parent(p))$ 
186   return  $o$ 
187 end

```

**Figure 4.7.** Reconstruction par dilatation avec l'union-find. Cet algorithme est décrit en section 4.4.5

important est que ce maximum régional de  $g$  est lié également à un maximum régional de  $f$ . Au fur et à mesure du parcours de  $S$ , on crée ainsi une forêt d'arbres, qui correspondent à tous les maxima régionaux de  $g$ .

Dans la phase finale (find), on distingue les points qui sont tels que  $g > f$ , auxquels cas ils leur est attribué la valeur maximale de  $f$  dans la composante connexe du maximum régional de  $g$ , ou bien le cas  $g = f$ , auquel cas leur valeur est inchangée.

Au final, on a dilaté la fonction  $f$  sous  $g$ , ce qui correspond exactement à l'opérateur de reconstruction.

#### 4.4.5.2. Détails

L'image de sortie  $o$  est utilisée tout du long. On y stocke l'état de traitement des composantes. L'image  $o$  prend ses valeurs finales dans la dernière phase de traitement.

Pour toutes les zones plates où au final on aura  $o = g$ , on ne gère pas de structure d'arbre (ce sont tous des singletons).

La fonction DEJA\_VU s'évalue à la volée sans stockage intermédiaire.

Si les valeurs de  $g$  sont faiblement quantifiées (exemple : image sur 8 bits), on peut réaliser un tri linéaire (radix sort ou tri par histogramme). Au final l'algorithme est quasi-linéaire dans le cas où FIND\_ROOT opère avec un arbre équilibré et quand on utilise une technique dite de compression de chemin [TAR 75].

#### 4.4.5.3. Comparaison avec les méthodes précédentes

Cet algorithme est quasi-linéaire dans le pire cas. Il n'est pas forcément plus rapide dans la pratique que l'algorithme hybride, mais il est emblématique de la mise en œuvre moderne des opérateurs connexes : ouverture/fermetures algébriques par attributs, nivellements, ligne de partage des eaux [BRE 96, JON 99, MEI 02, GÉR 05].

La représentation par forêt d'arbres en particulier permet une description théorique des opérateurs connexes particulièrement riche [FAL 04, NAJ 06]. Voir également les chapitres ?? et ?? de Morphologie Mathématique 1.

Dans les méthodes parallèles et séquentielles, le facteur limitant au point de vue complexité était le nombre de passes sur l'image. Pour l'algorithme à base de queue et hybride, le facteur limitant provient de la gestion de la file d'attente, sachant qu'on peut devoir, dans les cas défavorables, parcourir une grande partie de l'image.

Enfin pour le cas de l'Union-Find, le goulet d'étranglement est la recherche de racine de l'arbre.

#### 4.4.6. Comparaison de ces algorithmes

Afin de comparer les cinq algorithmes décrits précédemment, nous pouvons reprendre certains critères de taxonomie donnés en section 4.3.1. Le tableau ci-dessous illustre d'une part la diversité des algorithmes possibles pour traduire un même opérateur et, d'autre part, la difficulté de l'utilisation de ces critères. En effet, certains algorithmes ne sont pas monolithiques ; ils répondent alors à plusieurs critères. L'algorithme hybride, par exemple, n'est qu'« à moitié séquentiel » et utilise à la fois des passes et un front de propagation. Pour l'algorithme par Union-Find, on peut affirmer qu'il est séquentiel puisque, avec l'initialisation (le tri) et les deux passes, l'image est parcourue exactement trois fois ; l'ordre de parcours n'est cependant pas celui des séquences classiques vidéo et anti-vidéo.

algorithme	propriété	parcours	structures
parallèle	parallèle	passes	aucune
séquentiel	séquentiel	passes	aucune
à base de file	néant	front	file
hybride	1/2 séquentiel	2 passes + front	file
avec Union-Find	séquentiel	3 passes	tableau et arbre

Pour comparer les performances de ces cinq algorithmes de reconstruction géodésique par dilatation, nous avons pris pour  $g$  l'image lena (taille  $512 \times 512$  pixels) et pour  $f$  le conditionnement par  $g$  du maximum point à point entre l'image lena originale et lena tournée de  $90^\circ$  dans le sens horaire. Le voisinage considéré est celui de la 4-connexité. Comme il s'agit d'une comparaison, nous n'avons pas utilisé d'optimisation particulière (l'utilisation d'arithmétique de pointeurs par exemple) ; pour information, ces temps peuvent être facilement réduits d'un même facteur multiplicatif significatif.

algorithme	temps (en sec.)
parallèle	25,28
séquentiel	3,18
à base de file	0,65
hybride	0,34
avec Union-Find	0,34

Ce tableau montre parfaitement à quel point un même opérateur peut se traduire dans la pratique par des performances radicalement différentes suivant l'algorithme qui l'implémente. L'obtention d'un « bon » algorithme, possédant des propriétés en adéquation avec les attentes du praticien, est finalement une science en soi.

#### 4.5. Perspectives historiques et notes bibliographiques

L'historique des divers algorithmes et les liens entre la morphologie mathématique et les autres domaines apparentés à celle-ci prendrait à lui seul un livre entier. Nous devons nous contenter ici de quelques notes.

##### 4.5.1. Pré- et péri-morphologie

Comme toutes les disciplines scientifiques, la morphologie mathématique n'est pas née, pas plus qu'elle n'évolue, dans un vide intellectuel. Elle est une étape vers une meilleure connaissance de la représentation spatiale des objets physiques ou virtuels. Avant que Matheron et Serra ne baptisent la morphologie mathématique en 1962 [MAT 02], l'analyse d'image existait déjà et de nombreux algorithmes avaient été développés dans des disciplines annexes. De même, la morphologie continue à évoluer dans un contexte lui-même mouvant. Il est utile de fixer quelques repères algorithmiques dans ce foisonnement créatif.

###### 4.5.1.1. Algorithmes à base de graphes

La représentation des images étant le plus souvent sous forme de graphe régulier, il est peu étonnant que de nombreux algorithmes de morphologie mathématique dérivent d'algorithmes classiques. On peut citer les chemins minimaux Dijkstra [DIJ 59], le problème de l'arbre de poids minimum [JOS 56, PRI 57]. L'algorithme classique d'Union-Find [TAR 75] est également un algorithme classique de graphes très utilisé en morphologie mathématique. Nous conseillons l'ouvrage de référence [GON 95] pour tous ces algorithmes.

Il est plus que probable que toute la littérature classique ou récente sur les algorithmes à base de graphes n'ait pas encore été exploitée dans le contexte de la morphologie mathématique. Il s'agit donc d'une mine très riche pour de futurs résultats.

###### 4.5.1.2. Algorithmes de géométrie et de topologie discrète

La géométrie discrète est un champ de recherche actif lié à celui de la morphologie mathématique proprement dite. Son but est de redéfinir et d'exploiter, algorithmiquement dans un contexte discret, les objets et opérateurs de la géométrie classique. Par exemple, droites, plans, intersections, vecteurs normaux, etc [REV 91]. Les propriétés de ces objets sont nettement différentes de celles de leurs correspondants dans le continu. Un ouvrage de référence récent en français est [CŒU 07].



Parmi les algorithmes de la géométrie discrète les plus utiles en morphologie mathématique, on compte en particulier les fonctions distances [ROS 66, BOR 84, SHI 92], qui permettent de mettre en œuvre nombre d'opérateurs, tels les érosions et dilations binaires [RAG 92].

La topologie discrète est une discipline cherchant à définir des opérateurs topologiques dans les espaces discrets tels les images, mais également sur des graphes arbitraires, des surfaces ou des complexes [KON 89, BER 07]. Le lien avec la morphologie mathématique est très fort, en particulier pour les opérateurs d'amin-cissement [KON 95] et de squelettisation [MAN 02]. La ligne de partage des eaux est également un opérateur topologique [COU 05, COU 09b].

#### 4.5.1.3. Algorithmes inspirés du continu

Le continu n'est pas représentable exactement dans un ordinateur. Cependant, on peut considérer des objets mathématiques intrinsèquement continus, tels les équations aux dérivées partielles, pour résoudre des problèmes d'analyse d'images. Cette approche permet de produire des algorithmes aux propriétés intéressantes.

En prenant pour point de départ les algorithmes de segmentation, tel celui des contours actifs [KAS 98], on trouve des liens avec la squelettisation [LEY 92], ainsi que des généralisations de la ligne de partage des eaux incluant des contraintes de courbure [NGU 03].

Les algorithmes de marche rapide [SET 96a] sont essentiellement équivalents à un algorithme flexible de calcul de la fonction distance géodésique euclidienne [SOI 91], en métrique scalaire ou tensorielle [SET 01]. Ils permettent dans certains contextes de proposer une morphologie mathématique du continu [SAP 93]. Il est à noter que l'algorithme de Sethian n'est précis qu'au premier ordre. Une méthode rapide pour calculer la fonction distance euclidienne géodésique exacte est encore un problème ouvert.

Ces méthodes ont été utilisées en morphologie par exemple en filtrage connectif [MEY 00], en remplaçant l'opérateur de dilatation par une propagation continue. Des formulations de la LPE dans le continu [NAJ 93, MAR 96] peuvent être résolues en exploitant les algorithmes de marche rapide.

L'intérêt principal d'une formulation continue est de s'affranchir de la notion de pixel ; dans une certaine mesure, on peut définir une dilatation de rayon arbitraire, et non seulement entier. On peut ainsi également faire de la morphologie sur des variétés arbitraires, par exemple des surfaces triangulées, bien que cette approche ne soit pas la seule.

Nous verrons ci-après d'autres liens avec le continu au travers des algorithmes inspirés par les théories linéaires.

#### 4.5.1.4. Optimisation discrète et continue pour la segmentation

Les algorithmes de type « contours actifs » (*snakes*) [KAS 98], ou « lignes de niveau » (*level-sets*) [OSH 88, MER 94, SET 96b] sont comparables dans leur approche : il s'agit, dans le domaine de l'analyse d'image, de proposer une formulation d'optimisation continue, basée le plus souvent sur une descente de gradient. Cette formulation peut être exploitée en segmentation, l'objectif étant d'optimiser la position d'un ou de plusieurs contours fermés ou non, en 2D ou de surfaces en 3D, définissant ainsi une région d'intérêt. L'intérêt de ces formulations est la possibilité d'affecter un poids plus ou moins important aux différents aspects d'une segmentation (plus ou moins régulière, intégrant ou non des termes de textures ou de suivi du mouvement, etc.), ainsi que la gestion de la topologie. En revanche, certaines formulations complexes ne peuvent pas être optimisées de manière globale.

Ces dernières années, la communauté de vision et d'analyse d'images ont redécouvert l'intérêt des formulations plus simples mais permettant d'être optimisées globalement, avec les coupures de graphes [BOY 04], les flots maximaux continus [APP 06], ou encore les marches aléatoires [GRA 06]. En effet, ces formulations sont plus fiables, plus faciles d'emploi et moins sensibles au bruit. Il y a des liens entre ces différentes techniques et la ligne de partage des eaux [ALL 07, COU 09a].

#### 4.5.1.5. Algorithmes d'analyse linéaire

Par analyse linéaire, on entend tout le domaine lié aux transformées intégrales (Fourier, Radon, ondelettes, etc.) qui historiquement descendent du traitement du signal. Dans ce domaine, la structure de base est celle de groupe, avec comme opérateur l'addition. Pour les signaux, et pour certains types d'images ou de problèmes, cette structure est importante. Par exemple, on approxime souvent le bruit d'acquisition par un bruit gaussien additif, qu'une déconvolution permet d'éliminer de manière optimale. En imagerie médicale, comme le cas de la tomographie, la superposition additive des signaux est une hypothèse parfaitement raisonnable.

La morphologie mathématique n'est pas linéaire, la structure de base étant celle de treillis avec le supremum ou l'infimum comme opérateurs. Néanmoins, il existe des liens entre les deux approches, non seulement au niveau des applications, mais aussi de certains outils, comme par exemple l'analyse multi-résolution [HEI 00], encore appelée *scale-space* [JAC 96, VAC 01]. Pour plus d'informations, le lecteur pourra se référer au chapitre ?? de Morphologie Mathématique 1.

Pour finir, une curiosité : on peut définir la dilatation à partir d'une convolution par :

$$\delta_B[I] = (I \star B) > 0, \quad (4.4)$$

où  $I$  est une image binaire et  $B$  un élément structurant arbitraire. En implémentant la convolution par transformée de FOURIER rapide, cet algorithme permet la dilatation en temps constant par rapport à  $B$ . C'est le seul algorithme connu avec cette caractéristique.

#### 4.5.2. Historique des développements algorithmiques en morphologie mathématique

Dès le début, le développement de la morphologie mathématique en tant que discipline théorique autant qu'appliquée fut lié à un développement matériel et logiciel : l'analyseur de texture, réalisé à l'école des Mines [MAT 02]. Par la suite, la plupart des progrès dans cette discipline se sont révélés être le résultat d'une synergie constante entre les applications, la théorie, les algorithmes et les développements matériels.

Serra, dans [MAT 02], donne un historique des premières années de la morphologie mathématique.

Les développements matériels des premières années, portant sur des architectures dédiées au traitement d'images, étaient une nécessité, du fait des faibles puissances de calcul des ordinateurs d'alors. L'architecture matérielle typique implique un accès séquentiel (lecture vidéo) aux données, et non aléatoire.

##### 4.5.2.1. Algorithmes parallèles

Algorithmiquement, cela implique presque nécessairement un traitement *parallèle* des données. Ici le terme désignant des algorithmes qui génèrent un résultat en chaque pixel indépendamment de celui effectué sur les autres pixels, comme l'algorithme de gauche de la figure 4.5 et décrit en section 4.4.1. Ces algorithmes peuvent en effet généralement être distribués facilement entre plusieurs processeurs par exemple, mais ce n'est pas une nécessité. En termes matériels, les algorithmes parallèles s'appliquent bien à une architecture SIMD *Single*

*Instruction Multiple Data* et à la limite au modèle où chaque pixel dispose d'un processeur (rétines artificielles [MAN 00]). Parmi les développements matériels exploitant des algorithmes parallèles de morphologie mathématique, on compte le Morpho-Pericolor [BIL 92] et le Cambridge Instrument Quantimet 570 [KLE 90] ainsi que le processeur dédié (*ASIC*) PIMM1 [KLE 89].

Les premiers algorithmes de calcul de ligne de partage des eaux, de squelettisation et de filtrage morphologique ont été décrits et mis en œuvre sous forme parallèle, voir par exemple [BEU 79a, MON 68].

#### 4.5.2.2. Algorithmes séquentiels

Certains algorithmes peuvent être décrits sous forme *séquentielle*, ici désignant un algorithme qui utilise le résultat courant pour dériver le résultat suivant, en adoptant le plus souvent un *ordre de balayage*, comme dans la figure 4.5 de droite, et décrit en section 4.4.2. Les algorithmes séquentiels sont souvent plus rapides que les algorithmes parallèles, du moins sur les ordinateurs standards, car ils exploitent la redondance locale des images. Un algorithme séquentiel typique est celui de la fonction distance [ROS 66]. Peu d'algorithmes séquentiels de morphologie ont été mis en œuvre au niveau matériel, mais on peut citer l'étude de Lemonier [LEM 96] qui a, entre autres, proposé un algorithme séquentiel pour la ligne de partage des eaux.

#### 4.5.2.3. Algorithmes en largeur d'abord

Plus tard, alors que les ordinateurs montaient en puissance, est apparu comme productif l'idée d'explorer les pixels en partant de la frontière des objets, sans pour autant suivre un ordre imposé par la structure de la mémoire, mais en utilisant une structure de donnée adaptée. Parmi cette famille d'algorithmes, on compte ceux à base de file d'attente [VIN 90], de lacets [SCH 89] et de files à priorité [MEY 90a]. Un algorithme typique de ce type d'approches est celui de la ligne de partage des eaux par inondation [VIN 91c, MEY 90b]. Ce type d'algorithme en revanche se prête mal à la mise en œuvre sur matériel dédié de type FPGA. En effet, les structures de données sous-jacentes sont délicates d'utilisation et le goulet d'étranglement n'est pas la vitesse de l'unité centrale, mais celle de l'accès à la mémoire.

#### 4.5.2.4. Algorithmes inspirés des graphes

Les algorithmes en largeur d'abord sont classiques dans les problèmes de graphe. L'idée de continuer dans cette direction et d'adapter les algorithmes classiques des graphes est ensuite apparue. Parmi les algorithmes de morphologie mathématique inspirés des graphes, on peut compter l'IFT (*Image Foresting Transform*) [FAL 04]. Plus fondamentalement, l'idée de considérer une image réellement comme un graphe et, en particulier, celle de valuer également les arêtes, s'est finalement imposée. C'est une idée qui permet par exemple de définir assez naturellement la notion de gradient discret : la différence numérique entre les deux sommets valués liés par une arête [COU 07c]. Elle permet également de définir une frontière comme une coupure de graphe et non comme une chaîne de sommets, résolvant ainsi nombre de problèmes topologiques. Cette notion permet de produire de nouveaux algorithmes efficaces et de jeter des ponts avec d'autres disciplines, en particulier au niveau des méthodes de segmentation [COU 09a].

#### 4.5.2.5. Algorithmes topologiques

Outre la notion essentielle de point simple permettant des algorithmes efficaces, préservant l'homotopie des images, de nombreux travaux considèrent de manière essentielle la topologie des images. Parmi les notions essentielles on trouve celle d'arbre des coupes, utilisée en section 4.4.5.1 de ce même chapitre et décrite, de façon détaillée, au chapitre ?? du précédent volume. La notion d'arbre des coupes, de par sa représentation efficace des régions et bassins versants d'une image, permet la mise en œuvre de nombreux algorithmes, par exemple la segmentation hiérarchique, les nivellements et encore d'autres filtrages [NAJ 06].

#### 4.5.2.6. Filtrage

Les algorithmes de filtrage morphologique forment une classe intéressante en soi. Tout d'abord les références utiles sur le filtrage en morphologie mathématique sont, outre les livres de Serra [SER 82, SER 88a], un article assez complet sur la théorie des filtres morphologiques par Serra et Vincent [SER 92] et les articles de Heijmans et Ronse [HEI 90, RON 91]. Un article plus introductif est celui de Heijmans [HEI 96]. Aucun de ces articles ne traite en détail des aspects algorithmiques, pourtant essentiels. Voici une énumération incomplète mais illustrative de certains problèmes étudiés en morphologie.

##### *Érosions et dilations rapides.*

De nombreux auteurs ont proposé des méthodes efficaces de calcul des opérateurs de base de la morphologie mathématique. La plupart des bibliothèques logicielles de MM (y compris parmi les plus connues, nous ne citerons pas de noms) calculent un max ou un min sur une fenêtre de manière peu efficace, avec un algorithme avec  $O(MN)$  comparaisons, si  $M$  est le nombre de pixels dans l'image et  $N$  le nombre de pixels dans la fenêtre. Il est souvent possible de décomposer les éléments structurants en parties plus facilement calculables [XU 91]. Parmi les éléments structurants les plus courants, on compte les polygones réguliers convexes en 2D. On peut décomposer ceux-ci en opérations séparables par segments de droites. Un travail significatif a consisté à produire des algorithmes de filtrage morphologique par segments en temps constant par rapport à  $N$  [HER 92, BRE 93, GIL 02, VAN 05]. Le résultat sont des algorithmes capable de calculer les filtres morphologiques avec des ES polygonaux réguliers convexes en temps linéaire (avec  $O(N)$  comparaisons), pour tout  $M$ . Il est à noter qu'un résultat similaire en 3D ou plus est un problème ouvert à ce jour, sauf pour le cas du parallélépipède dont les faces sont perpendiculaires aux directions de la trame et quelques autres cas particuliers.

Pour les opérations morphologiques avec éléments structurants arbitraires en  $nD$ , il existe un algorithme en  $O(\sqrt[n]{N^{n-1}}M)$  [VAN 96]. Un algorithme effectivement plus rapide a été proposé en 2D seulement [URB 08]. Plusieurs algorithmes ont été proposés dans le cas binaire [JI 89, VIN 91b] de complexité asymptotiquement linéaire.

##### *Ouvertures et amincissements algébriques et arbre des coupes.*

Le filtrage en morphologie mathématique s'appuie plus sur la notion d'ouverture et fermeture que sur celle de dilatation et érosion. Il est courant de définir une notion d'ouverture/fermeture *algébrique* qui n'est pas liée directement à celle d'élément structurant, mais à celle d'*attribut* [BRE 96] et de connexion [CRE 93, CRE 97, HEI 99]. Cette notion est proche de celle de la reconstruction explicitée dans ce chapitre.

Cette notion permet de définir un grand nombre de filtres très efficaces. Historiquement la première implémentation du filtrage par aire est due à Vincent [VIN 92, VIN 93a, VIN 94]. La notion générale a été étendue aux attributs [BRE 96] non nécessairement croissants, donnant lieu à des opérations appelées *amincissements* algébriques, mais de principe similaire. Une implémentation efficace de ces opérateurs a été proposée dans [MEI 02], puis une généralisation en partant de l'arbre des coupes par *union-find* dans [GÉR 05]. Récemment, la notion de connectivité a été étendue à la notion d'hyper-connectivité [WIL 06].

La notion de connectivité par chemins permet également de définir des opérateurs par composition, par exemple en utilisant des familles de droites [SOI 01] ou de chemins [HEI 05, TAL 07] qui permettent le filtrage et la détection d'objets fins, avec de multiples applications comme par exemple en télédétection [VAL 09].

##### *Filtrage spatialement variable*

Plus récemment, des opérateurs efficaces permettant le filtrage par opérateurs non invariant par translation (ou spatialement variables) ont été proposés [CHA 94, BOU 08a, BOU 08b]. Ce type de filtrage adapte en chaque pixel la forme de l'élément structurant utilisé en fonction du contenu local de l'image. Cela permet par exemple de tenir compte de la perspective, de la texture [LER 06], de l'orientation [VER 08], etc. Ces filtres peuvent se révéler efficace dans le cadre du filtrage inverse [TAN 09a, TAN 09b].

#### *Cas $nD$*

Le cadre de la MM s'étend naturellement aux dimensions supérieures [GES 90, GRA 93b]. Comme l'écrit Gratin, les difficultés ne sont pas souvent d'ordre théoriques mais pratiques : les images sont plus grandes, la connectivité plus complexe, les problèmes de bord plus nombreux. Un point délicat est l'échantillonnage de directions. En 2D, de nombreux algorithmes échantillonnent les directions par exemple pour réaliser un filtrage avec des familles de segments avec des orientations isotropes. En 3D et plus, c'est plus difficile, car un échantillonnage isotrope des directions est impossible, résultat connu depuis Platon [HEA 56]. Du fait de l'amélioration des capteurs, des instruments et des ordinateurs, l'analyse des images en 3D et plus (3D + temps par exemple) est de fait de plus en plus courante dans la pratique. Les grands champs d'applications sont pour le moment l'imagerie médicale, les sciences des matériaux et l'imagerie biologique et bio-moléculaire.

#### **4.6. Conclusion**

Dans ce chapitre, nous avons essayé de montrer au lecteur la distance qui existe entre la formulation mathématique d'un opérateur morphologique et sa traduction algorithmique. À partir d'un exemple, nous avons également illustré le fait qu'il n'existe généralement pas un unique algorithme permettant la mise en œuvre d'un opérateur mais plusieurs, dont les caractéristiques peuvent être quelquefois radicalement opposées.

L'algorithmique, qu'elle soit dédiée à la morphologie mathématique ou non, reste un domaine ouvert. Les opérateurs de traitements d'images développés au cours du temps sont de plus en plus exigeants en termes de charge de calculs. À cela s'ajoute la croissance constante du volume des données à traiter. Aussi, on est de plus en plus contraint à attacher une importance particulière à la mise en œuvre algorithmique des opérateurs de traitement d'images, et ce, que cette mise en œuvre soit logicielle ou matérielle.

Les challenges futurs de ce domaine sont de plusieurs natures.

Tout d'abord, les observateurs ont notés la difficulté de la reproductibilité des méthodes et résultats présentés dans la communauté scientifique. Partant d'un article scientifique, le chemin qui mène à l'implémentation est long : à la distance entre les formes mathématique et algorithme s'ajoute également le parfois douloureux passage de l'algorithme à sa mise en œuvre sous forme de programme. En conséquence, la perte de capitalisation d'informations et donc de connaissances est observable. Beaucoup de solutions proposées dans la littérature sont laissées de côté et très peu d'articles présentent des comparaisons entre un nombre significatif de solutions. Un effort devrait être mené afin de doter la communauté du traitement d'images une plate-forme mutualisée d'algorithmes avec leur implémentation.

Un second challenge concerne la mise en œuvre d'algorithmes. Ces derniers sont intrinsèquement abstraits et, dans ce chapitre, nous leur avons conservé cette forme abstraite. En effet, ces algorithmes, tels que décrits, s'appliquent aussi bien sur de classiques images bi-dimensionnelles que sur des signaux (images 1D), volumes (3D) ou des graphes et complexes topologiques. Aucune hypothèse implicite supplémentaire n'est venue réduire le champ d'application de ces algorithmes. Malheureusement, la traduction de la forme algorithmique en programme est souvent accompagnée d'une perte de généralité de l'algorithme. Telle bibliothèque dédiée

à la morphologie mathématique ne permettra pas de traiter des images 3D, telle autre de prendre un élément structurant plat de forme quelconque, telle autre d'imaginer doter un espace de couleur d'une relation d'ordre pour accéder à la morphologie, etc. Même une « simple » dilatation, donnée par l'algorithme (3) en figure 4.2, devient, une fois programmée, une dilatation restreinte à un nombre limité de cas. Notons que des solutions existent, permettant d'obtenir la généricité la plus grande possible, sans sacrifier la facilité d'utilisation ni l'efficacité [LEV 09].

Un troisième challenge est celui de l'effort communautaire. Il est de plus en plus admis que pour être recevable avant et après publication, un algorithme doit être assorti d'une implémentation librement accessible. En effet, tout effort de ré-implémentation est effectivement du temps perdu. L'absence dans la communauté morphologique d'une plate-forme de développement commune est certainement un point qui aura freiné l'adoption de certains algorithmes dans la communauté plus vaste des chercheurs et utilisateurs de l'analyse d'image.

Enfin, un dernier défi réside dans le développement des moyens informatiques. Nous sommes passés à l'heure des processeurs multi-cœurs, des cartes multi-processeurs, et des grappes de calcul. Les outils de traitement d'images, et en particulier les algorithmes, doivent être adaptés à ces nouveaux environnements.

## Table des figures





## Bibliographie

- [ALL 07] ALLÈNE C., AUDIBERT J.-Y., COUPRIE M., COUSTY J., KERIVEN R., « Some links between min-cuts, optimal spanning forests and watersheds », *Mathematical Morphology and its Applications to Signal and Image Processing, proc. 8th International Symposium on Mathematical Morphology*, vol. 1, p. 253-264, Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, 2007.
- [APP 06] APPLETON B., TALBOT H., « Globally Minimal Surfaces by Continuous Maximal Flows », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, n° 1, p. 106-118, 2006.
- [ATT 95] ATTALI D., SANNITI DI BAJA G., THIEL E., « Pruning discrete and semicontinuous skeletons », *Procs. Conf. Image Analysis and Processing*, vol. 974 de *Lecture Notes in Computer Science*, p. 488-493, Springer, 1995.
- [ATT 96] ATTALI D., MONTANVERT A., « Modelling noise for a better simplification of skeletons », *Procs. International Conference on Image Processing*, vol. 3, p. 13-16, 1996.
- [ATT 09] ATTALI D., BOISSONNAT J., EDELSBRUNNER H., « Stability and Computation of the medial axis – a State-of-the-Art Report », MÖLLER T., HAMANN B., RUSSELL B., (DIR.), *Mathematical Foundations of Scientific Visualization, Computer Graphics, and Massive Data Exploration*, p. 109–125, Springer-Verlag, 2009.
- [BAD 05] BADDELEY A., VEDEL JENSEN E. B., *Stereology for statisticians*, Monographs on Statistics and Applied Probability, CRC Press, Boca Raton, Floride, 2005.
- [BER 94a] BERTRAND G., « Simple points, topological numbers and geodesic neighborhoods in cubic grids », *Pattern Recognition Letters*, vol. 15, p. 1003-1011, 1994.
- [BER 94b] BERTRAND G., MALANDAIN G., « A new characterization of three-dimensional simple points », *Pattern Recognition Letters*, vol. 15, n° 2, p. 169-175, 1994.
- [BER 95] BERTRAND G., « On P-simple points », *Comptes Rendus de l'Académie des Sciences, Série Math.*, vol. I, n° 321, p. 1077-1084, 1995.
- [BER 05] BERTRAND G., « On topological watersheds », *Journal of Mathematical Imaging and Vision*, vol. 22, n° 2-3, p. 217-230, mai 2005.
- [BER 07] BERTRAND G., « On critical kernels », *Comptes Rendus de l'Académie des Sciences, Série Math.*, vol. I, n° 345, p. 363-367, 2007.
- [BEU 79a] BEUCHER S., LANTUÉJOUL C., Sur l'utilisation de la ligne de partage des eaux en détection de contours, rapport n° N-598, Ecole des Mines de Paris, mai 1979.
- [BEU 79b] BEUCHER S., LANTUÉJOUL C., « Use of watersheds in contour detection », *International Workshop on Image Processing*, p. 2.1-2.12, CCETT/IRISA, Rennes, septembre 1979.
- [BEU 94] BEUCHER S., « Watershed, hierarchical segmentation and waterfall algorithm », SERRA J., SOILLE P., (DIR.), *Mathematical Morphology and its Applications to Image Processing*, p. 69-76, Kluwer Academic Publishers, 1994.

- [BIL 92] BILODEAU M., Architecture logicielle pour processeur de morphologie mathématique, thèse de doctorat, Ecole nationale supérieure des Mines de Paris, 1992.
- [BIR 95] BIRKHOFF G., *Lattice Theory*, 3<sup>e</sup> édition, vol. 25 de *American Mathematical Society Colloquium Publications*, American Mathematical Society, 1995.
- [BLU 61] BLUM H., « An associative machine for dealing with the visual field and some of its biological implications », BERNARD E. E., KARE M. R., (DIR.), *Biological Prototypes and synthetic systems*, vol. 1, p. 244-260, 2nd Annual Bionics Symposium, Cornell University, Plenum Press, New-York, 1961.
- [BLU 67] BLUM H., « A transformation for extracting new descriptors of shape », WATHEN-DUNN W., (DIR.), *Models for the Perception of Speech and Visual Form*, p. 362-380, M.I.T. Press, Cambridge, MA, États Unis, 1967.
- [BOO 96] VAN DEN BOOMGAARD R., DORST L., MAKRAM-EBEID S., SCHAVEMAKER J., « Quadratic structuring functions in mathematical morphology », MARAGOS P., SCHAFER R., BUTT M., (DIR.), *Mathematical Morphology and its Applications to Image and Signal Processing*, p. 147-154, Kluwer Academic Publishers, Atlanta, GA., 1996.
- [BOR 84] BORGEFORS G., « Distance transformations in arbitrary dimensions », *Computer Vision, Graphics, and Image Processing*, vol. 27, p. 321-345, 1984.
- [BOU 32] BOULIGAND G., « *Géométrie infinitésimale directe* », Chapitre 12, Vuibert, 1932.
- [BOU 08a] BOUAYNAYA N., CHARIF-CHEFCHAOUNI M., SCHONFELD D., « Theoretical Foundations of Spatially-Variant Mathematical Morphology Part I : Binary Images », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, n° 5, p. 823-836, IEEE Computer Society, 2008.
- [BOU 08b] BOUAYNAYA N., SCHONFELD D., « Theoretical Foundations of Spatially-Variant Mathematical Morphology Part II : Gray-Level Images », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, n° 5, p. 837-850, IEEE Computer Society, 2008.
- [BOY 04] BOYKOV Y., KOLMOGOROV V., « An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision », *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, n° 9, p. 1124-1137, septembre 2004.
- [BRE 93] BREEN E., SOILLE P., « Generalization of van Herk recursive erosion/dilation algorithm to lines at arbitrary angles », FUNG K., GINIGE A., (DIR.), *Proc. DICTA'93 : Digital Image Computing : Techniques and Applications*, p. 549-555, Australian Pattern Recognition Society, Sydney, Australie, décembre 1993.
- [BRE 94] BREEN E., MONRO D., « An evaluation of priority queues for mathematical morphology », SERRA J., SOILLE P., (DIR.), *Mathematical Morphology and its Applications to Image Processing*, p. 249-256, Kluwer Academic Publishers, 1994.
- [BRE 96] BREEN E., JONES R., « Attribute openings, thinnings, and granulometries », *Computer Vision and Image Understanding*, vol. 64, n° 3, p. 377-389, 1996.
- [BRE 00] BREEN E., JONES R., TALBOT H., « Mathematical Morphology, a Useful Set of Tools for Image Analysis », *Statistics and Computing*, vol. 10, p. 105-120, CSIRO, 2000.
- [CAL 68] CALABI L., HARTNETT W. E., « Shape recognition, prairie fires, convex deficiencies and skeletons », *Amer. Math. Monthly*, vol. 75, p. 335-342, 1968.
- [CHA 94] CHARIF-CHEFCHAOUNI M., SCHONFELD D., « Spatially-variant mathematical morphology », *Proceedings of International Conference on Image Processing (ICIP)*, vol. 2, p. 555-559, novembre 1994.
- [CHA 05] CHAZAL F., LIEUTIER A., « The Lambda Medial Axis », *Graphical Models*, vol. 67, n° 4, p. 304-331, 2005.
- [CHA 09] CHAUSSARD J., COUPRIE M., TALBOT H., « A discrete lambda-medial axis », *Proceedings of the 15th international conference on Discrete Geometry for Computer Imagery (DGCI'09)*, vol. 5810 de *Lecture Notes in Computer Science*, p. 421-433, Springer, octobre 2009.
- [COE 03] COEURJOLLY D., « d-dimensional reverse Euclidean distance transformation and Euclidean medial axis extraction in optimal time », *Proceedings of Discrete Geometry for Computer Imagery*, vol. 2886 de *Lecture Notes in*

- Computer Science*, p. 327-337, 2003.
- [CÈU 07] CÈURJOLLY D., MONTANVERT A., CHASSERY J., (DIR.), *Géométrie discrète et images numérique*, IC2 signal et image, Hermès, 2007, ISBN 978-2746216433.
- [COM 07] COMBARET N., TALBOT H., « Robust 3D segmentation of composite materials fibres », BANON G. J. F., BARRERA J., BRAGA-NETO U. D. M., HIRATA N. S. T., (DIR.), *Proceedings*, vol. 2, p. 25-26, Universidade de São Paulo (USP), Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, octobre 2007.
- [COR 90] CORMEN T. H., LEISERSON C., RIVEST R., *Introduction to algorithms*, MIT Press, 1990.
- [COU 05] COUPRIE M., NAJMAN L., BERTRAND G., « Quasi-linear algorithms for the topological watershed », *Journal of Mathematical Imaging and Vision, special issue on Mathematical Morphology*, vol. 22, n° 2-3, p. 231-249, mai 2005.
- [COU 06] COUPRIE M., Note on fifteen 2D parallel thinning algorithms, rapport n° IGM2006-01, Université de Marne-la-Vallée, 2006.
- [COU 07a] COUPRIE M., COEURJOLLY D., ZROUR R., « Discrete bisector function and Euclidean skeleton in 2D and 3D », *Image and Vision Computing*, vol. 25, n° 10, p. 1543-1556, 2007.
- [COU 07b] COUSTY J., BERTRAND G., NAJMAN L., COUPRIE M., Watersheds, minimum spanning forests, and the drop of water principle, rapport n° IGM2007-01, LabInfo-IGM, UMR 8049, Université Paris-Est, 2007, <http://igm.univ-mlv.fr/LabInfo/rapportsInternes/2007/01.pdf>.
- [COU 07c] COUSTY J., BERTRAND G., NAJMAN L., COUPRIE M., « Watersheds, minimum spanning forests, and the drop of water principle », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, n° 8, p. 1362-1374, août 2007.
- [COU 08] COUPRIE M., BERTRAND G., « New characterizations of simple points in 2D, 3D and 4D discrete spaces », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, n° 9, p. 637-648, 2008.
- [COU 09a] COUPRIE C., GRADY L., NAJMAN L., TALBOT H., « Power watersheds : a new image segmentation framework extending graph cuts, random walker and optimal spanning forest », *12th International Conference on Computer Vision (ICCV'09)*, p. 731-738, septembre 2009.
- [COU 09b] COUSTY J., BERTRAND G., COUPRIE M., NAJMAN L., « Collapses and watersheds in pseudomanifolds », WIEDERHOLD P., BARNEVA R. P., (DIR.), *13th International Workshop on Combinatorial Image Analysis (IWCIA'09)*, vol. 5852 de *Lecture Notes in Computer Science*, p. 397-410, Springer, novembre 2009.
- [CRE 93] CRESPO J., SERRA J., SCHAFFER R., « Image segmentation using connected filters », SERRA J., SALEMBIER P., (DIR.), *Mathematical morphology and its applications to signal processing*, p. 52-57, Universitat Politècnica de Catalunya, Barcelone, Espagne, mai 1993.
- [CRE 97] CRESPO J., SCHAFFER R. W., « Locality and Adjacency Stability Constraints for Morphological Connected Operators », *Journal of Mathematical Imaging and Vision*, vol. 7, p. 85-102, 1997.
- [DAN 80a] DANIELSSON P.-E., « Euclidean distance mapping », *Computer Graphics and Image Processing*, vol. 14, p. 227-248, 1980.
- [DAN 80b] DANIELSSON P., « Euclidean distance mapping », *Computer Graphics and Image Processing*, vol. 14, p. 227-248, 1980.
- [DAV 81] DAVIES E., PLUMMER A., « Thinning algorithms : a critique and a new methodology », *Pattern Recognition*, vol. 14, p. 53-63, 1981.
- [DEH 62] DE HOFF R., « The determination of the size distribution of ellipsoidal particles from measurements made on random plane sections », *Transactions of the AIME*, vol. 224, p. 474-477, 1962.
- [DEL 48] DELESSE A., Volume fractions from Area fractions of minerals in optical microscopy, rapport, TBA, 1848, Cited by Baddeley.

- [DIJ 59] DIJKSTRA E., « A note on two problems in connexion with graphs », *Numerische Mathematik*, vol. 1, p. 269-271, 1959.
- [DUD 67] DUDA O., HART P., MUNSON J., Graphical data processing research study and experimental investigation, rapport n° AD650926, United States Army Electronics Command, 1967.
- [DUR 30] DURAND G., « Théorie des ensembles. Points ordinaires et points singuliers des enveloppes de sphères », *Comptes-rendus de l'Académie des Sciences*, vol. 190, p. 571-573, 1930.
- [DUR 31] DURAND G., Sur une généralisation des surfaces convexes, thèse de doctorat, Faculté des sciences de Paris, 1931.
- [FAL 04] FALCAO A. X., STOLFI J., DE ALENCAR LOTUFO R., « The Image Foresting Transform : Theory, Algorithms, and Applications », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, n° 1, p. 19-29, 2004.
- [FRE 87] FREDMAN M. L., TARJAN R. E., « Fibonacci heaps and their uses in improved network optimization algorithms », *Journal of the ACM*, vol. 34, n° 596-615, 1987.
- [GÉR 05] GÉRAUD T., « Ruminations on Tarjan's Union-Find algorithm and connected operators », *Mathematical Morphology : 40 Years On, Proceedings of the International Symposium (ISMM)*, vol. 30 de *Computational Imaging and Vision*, p. 105-116, Kluwer Academic Publishers, Paris, France, avril 2005.
- [GES 90] GESBERT S., HOWARD V., JEULIN D., MEYER F., « The use of basic morphological operations for 3D biological image analysis », *Trans. Roy. Microsc. Soc.*, vol. 1, p. 293-296, London, juillet 1990.
- [GIL 02] GIL J., KIMMEL R., « Efficient dilation, erosion, opening and closing algorithms », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, n° 12, p. 1606-1617, 2002.
- [GOL 69] GOLAY J., « Hexagonal parallel pattern transformations », *IEEE Transactions on Computers*, vol. C-18, n° 8, p. 733-740, 1969.
- [GON 95] GONDRAN M., MINOUX M., *Graphes et algorithmes*, 3<sup>e</sup> édition, Eyrolles, Paris, 1995.
- [GRA 93a] GRATIN C., De la représentation des images au traitement morphologique d'images tridimensionnelles, thèse de doctorat, Ecole des Mines de Paris, janvier 1993.
- [GRA 93b] GRATIN C., De la représentation des images au traitement morphologique d'images tridimensionnelles, thèse de doctorat, Ecole des Mines de Paris, 1993.
- [GRA 06] GRADY L., « Random Walks for Image Segmentation », *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, n° 11, p. 1768-1783, novembre 2006.
- [HAD 57] HADWIGER H., *Vorlesungen über Inhalt, Oberfläche und Isoperimetrie*, Springer-Verlag, 1957.
- [HAU 63] HAUG H., (DIR.), *First International Congress for Stereology*, Vienna, 1963.
- [HEA 56] HEATH T. L., *The thirteen books of Euclid's elements*, 2<sup>e</sup> édition, vol. (3 volumes), Dover Publications, 1956, ISBN 0-486-60088-2, 0-486-60089-0, 0-486-60090-4.
- [HEI 90] HEIJMANS H., RONSE C., « The algebraic basis of mathematical morphology : I. Dilations and erosions », *Computer Vision, Graphics, and Image Processing*, vol. 50, p. 245-295, 1990.
- [HEI 94] HEIJMANS H., *Morphological image operators*, Advances in Electronics and Electron Physics Series, Academic Press, Boston, 1994.
- [HEI 96] HEIJMANS H., « Morphological Filters for Dummies », MARAGOS P., (DIR.), *Mathematical morphology and its applications to Image and Signal processing*, p. 127-138, Kluwer Academic Publishers, Atlanta, GA, mai 1996, Proceedings of the 3rd International Symposium on Mathematical Morphology.
- [HEI 99] HEIJMANS H., « Connected morphological operators for binary images », *Computer Vision and Image Understanding*, vol. 73, n° 1, p. 99-120, 1999.
- [HEI 00] HEIJMANS H., GOUTSIAS J., « Nonlinear multiresolution signal decomposition schemes—Part II : Morphological wavelets », *IEEE Transactions on Image Processing*, vol. 9, n° 11, p. 1897-1913, novembre 2000.

- [HEI 05] HEIJMANS H., BUCKLEY M., TALBOT H., « Path openings and closings », *Journal of Mathematical Imaging and Vision*, vol. 22, p. 107-119, 2005.
- [HER 92] VAN HERK M., « A fast algorithm for local minimum and maximum filters on rectangular and octogonal kernels », *Pattern Recognition Letters*, vol. 13, p. 517-521, 1992.
- [HIR 96] HIRATA T., « A unified linear-time algorithm for computing distance maps », *Information Processing Letters*, vol. 58, n° 3, p. 129-133, 1996.
- [JAC 96] JACKWAY P., « Gradient watersheds in morphological scale-space », *IEEE Transactions on Image Processing*, vol. 5, n° 6, p. 913-921, juin 1996.
- [JI 89] JI L., PIPER J., TANG J., « Erosion and dilation of binary images by arbitrary structuring elements using interval coding », *Pattern Recognition Letters*, vol. 9, p. 201-209, 1989.
- [JON 99] JONES R., « Connected filtering and segmentation using component trees », *Computer Vision and Image Understanding*, vol. 75, n° 3, p. 215-228, 1999.
- [JOS 56] JOSEPH B. KRUSKAL J., « On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem », *Proceedings of the American Mathematical Society*, vol. 7, n° 1, p. 48-50, février 1956.
- [KAS 98] KASS M., WITKIN A., TERZOPOULOS D., « Snakes : Active Contour Models », *International Journal of Computer Vision*, vol. 1, n° 4, p. 321-331, 1998.
- [KHA 90] KHALIMSKY E., KOPPERMAN R., MEYER P., « Computer graphics and connected topologies on finite ordered sets », *Topology Appl.*, vol. 36, p. 1-17, 1990.
- [KLE 89] KLEIN J. C., PEYRARD R., « PIMM1, an image processing ASIC based on Mathematical Morphology », *Proceedings of the Second annual IEEE ASIC Seminar and Exhibit*, p. 1-4, 1989.
- [KLE 90] KLEIN J. C., COLLANGE F., BILODEAU M., « A bit plane architecture for an image processor implemented with P.L.C.A. gate array », *proceedings of ECCV 1990*, vol. 427 de *Lecture Notes in Computer Science*, p. 33-49, Springer, 1990.
- [KON 89] KONG Y. T., ROSENFELD A., « Digital topology : introduction and survey », *Comp. Vision, Graphics and Image Proc.*, vol. 48, p. 357-393, 1989.
- [KON 95] KONG T., « On topology preservation in 2-D and 3-D thinning », *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 9, n° 5, p. 813-844, 1995.
- [LAN 78] LANTUÉJOUL C., La squelettisation et son application aux mesures topologiques des mosaïques polycristallines, thèse de doctorat, Ecole des Mines de Paris, 1978.
- [LAN 80] LANTUÉJOUL C., « Skeletonization in quantitative metallography », HARALICK R., SIMON J., (DIR.), *Issues in Digital Image Processing*, vol. 34 de *NATO ASI Series E*, p. 107-135, Sijthoff & Noordhoff, Alphen aan den Rijn, 1980.
- [LEM 96] LEMONNIER F., Architecture électronique dédiée aux algorithmes rapides de segmentation basés sur la morphologie mathématique, thèse de doctorat, Ecole nationale supérieure des Mines de Paris, 1996.
- [LER 06] LERALLUT R., DECENCIÈRE E., MEYER F., « Image filtering using morphological amoebas », *Image and Vision Computing*, vol. 25, n° 4, p. 395-404, 2006.
- [LEV 09] LEVILLAIN R., GÉRAUD T., NAJMAN L., « Milena : Write Generic Morphological Algorithms Once, Run on Many Kinds of Images », WILKINSON M., ROERDINK J., (DIR.), *Mathematical Morphology, Proceedings of the 9th International Symposium (ISMM)*, vol. 5720 de *Lecture Notes in Computer Science*, p. 295-306, Springer-Verlag, Groningen, Pays-Bas, August 2009.
- [LEY 92] LEYMARIE F., LEVINE M., « Simulating the Grassfire Transform Using an Active Contour Model », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, n° 1, p. 56-75, IEEE Computer Society, 1992.

- [LIS 61] LISTING J., *Abhandlungen der Mathematischen Classe der Königlichen Gesellschaft der Wissenschaften zu Göttingen*, 1861.
- [MAL 89] MALING D., *Measurements from maps : principles and methods of cartometry*, Pergamon Press, Oxford, 1989.
- [MAL 93] MALANDAIN G., BERTRAND G., AYACHE N., « Topological segmentation of discrete surfaces », *Int Journal of Comp Vision*, vol. 10, n° 2, p. 183-197, 1993.
- [MAL 98] MALANDAIN G., FERNÁNDEZ-VIDAL S., « Euclidean Skeletons », *Image and Vision Computing*, vol. 16, p. 317-327, 1998.
- [MAN 83] MANDELBROT B., *The fractal geometry of nature*, W.H. Freeman and Company, New York, 1983.
- [MAN 00] MANZANERA A., *Vision Artificielle Rétinienne*, thèse de doctorat, Ecole Nationale Supérieure des Télécommunications, 2000.
- [MAN 02] MANZANERA A., BERNARD T., PRÊTEUX F., LONGUET B., « N-dimensional skeletonization : a unified mathematical framework », *Journal of Electronic Imaging*, vol. 11, n° 25, p. 25-37, SPIE, 2002.
- [MAR 96] MARAGOS P., « Differential Morphology and Image-Processing », *IEEE Transactions on Image Processing*, vol. 5, n° 6, p. 922-937, 1996.
- [MAT 75] MATHERON G., *Random sets and integral geometry*, Wiley, New York, 1975.
- [MAT 88a] MATHERON G., « Example of topological properties of skeletons », SERRA J., (DIR.), *Image Analysis and Mathematical Morphology*, vol. 2, Theoretical Advances, p. 217-238, Academic Press, London, 1988.
- [MAT 88b] MATHERON G., « On the negligibility of the skeleton », SERRA J., (DIR.), *Image Analysis and Mathematical Morphology*, vol. 2, Theoretical Advances, p. 239-256, Academic Press, London, 1988.
- [MAT 89] MATHERON G., *Estimating and choosing - An essay on probability in practice*, Springer, Berlin, 1989.
- [MAT 92] MATTIOLI J., SCHMITT M., « Inverse problems for granulometries by erosions », *Journal of Mathematical Imaging and Vision*, vol. 2, n° 3, p. 217-232, 1992.
- [MAT 02] MATHERON G., SERRA J., « The birth of mathematical morphology », TALBOT H., BEARE R., (DIR.), *Proceedings of VIth International Symposium on Mathematical Morphology*, p. 1-16, Commonwealth Scientific and Industrial Research Organisation, Sydney, Australie, avril 2002.
- [MAU 03] MAURER C., QI R., RAGHAVAN V., « A Linear Time Algorithm for Computing Exact Euclidean Distance Transforms of Binary Images in Arbitrary Dimensions », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, n° 2, p. 265-270, 2003.
- [MEH 99] MEHNERT A., JACKWAY P., « On Computing the Exact Euclidean Distance Transform on Rectangular and Hexagonal Grids », *Journal of Mathematical Imaging and Vision*, vol. 11, n° 3, p. 223-230, 1999.
- [MEI 00] MEIJSTER A. C., ROERDINK J. B. T. M., HESSELINK W. H., « A general algorithm for computing distance transforms in linear time », GOUTSIAS J., VINCENT L., BLOOMBERG D., (DIR.), *Mathematical morphology and its applications to image and signal processing 5th*, vol. 18 de *Computational Imaging and Vision*, p. 331-340, Kluwer Academic Publishers, Palo Alto, CA, États Unis, 2000.
- [MEI 02] MEIJSTER A., WILKINSON M., « A comparison of algorithms for connected set openings and closings », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, n° 4, p. 484-494, 2002.
- [MER 94] MERRIMAN B., BENCE J., OSHER S., « Motion of multiple junctions : a level set approach », *Journal of Computational Physics*, vol. 112, p. 334-363, 1994.
- [MEY 79] MEYER F., *Cytologie quantitative et morphologie mathématique*, thèse de doctorat, École des Mines de Paris, France, 1979.
- [MEY 90a] MEYER F., *Algorithmes à base de files d'attente hiérarchique*, rapport n° NT-46/90/MM, Ecole des Mines de Paris, septembre 1990.

- [MEY 90b] MEYER F., BEUCHER S., « Morphological segmentation », *Journal of Visual Communication and Image Representation*, vol. 1, n° 1, p. 21-46, septembre 1990.
- [MEY 91] MEYER F., « Un algorithme optimal de ligne de partage des eaux », *Reconnaissance des Formes et Intelligence Artificielle, 8e congrès*, p. 847-857, AFCET, Lyon-Villeurbanne, 1991.
- [MEY 94] MEYER F., « Topographic distance and watershed lines », *Signal Processing*, vol. 38, n° 1, p. 113-125, juillet 1994.
- [MEY 00] MEYER F., MARAGOS P., « Nonlinear scale-space representation with morphological levelings », *Journal of Visual Communication and Image Representation*, vol. 11, n° 3, p. 245-265, 2000.
- [MIL 75] MILES R. E., « On the elimination of edge effects in planar sampling », HARDING E., KENDALL D., (DIR.), *Stochastic geometry*, p. 228-247, Wiley, Londres, 1975.
- [MON 68] MONTANARI U., « A method for obtaining skeletons using a quasi-Euclidean distance », *Communications of the ACM*, vol. 15, n° 4, p. 600-624, octobre 1968.
- [NAJ 93] NAJMAN L., SCHMITT M., « Definitions and some properties of the watershed of a continuous function », SERRA J., SALEMBIER P., (DIR.), *Mathematical morphology and its applications to signal processing*, p. 76-81, 1993.
- [NAJ 94] NAJMAN L., SCHMITT M., « Watershed of a continuous function », *Signal Processing*, vol. 38, n° 1, p. 99-112, Elsevier North-Holland, Inc., 1994.
- [NAJ 96] NAJMAN L., SCHMITT M., « Geodesic Saliency of Watershed Contours and Hierarchical Segmentation », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, n° 12, p. 1163-1173, décembre 1996.
- [NAJ 05] NAJMAN L., COUPRIE M., BERTRAND G., « Watersheds, mosaics and the emergence paradigm », *Discrete Applied Mathematics, special issue on Discrete Geometry*, vol. 147, n° 2-3, p. 301-324, avril 2005.
- [NAJ 06] NAJMAN L., COUPRIE M., « Building the component tree in quasi-linear time », *IEEE Transactions on Image Processing*, vol. 15, n° 11, p. 3531-3539, 2006.
- [NGU 03] NGUYEN H. T., WORRING M., VAN DEN BOOMGAARD R., « Watersnakes : Energy-Driven Watershed Segmentation », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, n° 3, p. 330-342, IEEE Computer Society, 2003.
- [OSH 88] OSHER S., SETHIAN J., « Fronts propagating with curvature-dependent speed : Algorithms based on the Hamilton-Jacobi formulation », *J. Comput. Phys.*, vol. 79, n° 1, p. 12-49, 1988.
- [PRI 57] PRIM R., « Shortest connection networks and some generalisations », *Bell System Technical Journal*, vol. 36, 1957.
- [PUD 98] PUDNEY C., « Distance-ordered homotopic thinning : a skeletonization algorithm for 3D digital images », *Computer Vision and Image Understanding*, vol. 72, n° 3, p. 404-413, 1998.
- [RAG 92] RAGNEMALM I., « Fast erosion and dilation by contour processing and thresholding of distance maps », *Pattern Recognition Letters*, vol. 13, p. 161-166, 1992.
- [RAG 93] RAGNEMALM I., « The Euclidean Distance Transform in Arbitrary Dimensions », *Pattern Recognition Letters*, vol. 14, n° 11, p. 883 - 888, 1993.
- [REV 91] REVEILLÈS J., Géométrie discrète, calculs en nombres entiers et algorithmique, Thèse d'État, Université Louis Pasteur, Strasbourg, 1991.
- [RIV 87] RIVIÈRE A., Classification des points d'un ouvert d'un espace euclidien relativement à la distance au bord, étude topologique et quantitative des classes obtenues, thèse de doctorat, Université de Paris-Sud Orsay, novembre 1987.
- [RIV 93] RIVEST J., SOILLE P., BEUCHER S., « Morphological gradients », *Journal of Electronic Imaging*, vol. 2, n° 4, p. 326-336, octobre 1993.
- [ROE 01] ROERDINK J. B. T. M., MEIJSTER A., « The Watershed Transform : Definitions, Algorithms and Parallelization Strategies », *Fundamenta Informaticae*, vol. 41, n° 1-2, p. 187-228, 2001.

- [RON 88] RONSE C., « Minimal test patterns for connectivity preservation in parallel thinning algorithms for binary digital images », *Discrete Applied Mathematics*, vol. 21, n° 1, p. 67-79, 1988.
- [RON 91] RONSE C., HEIJMANS H., « The algebraic basis of mathematical morphology : II. Openings and closings », *Computer Vision, Graphics, and Image Processing : Image Understanding*, vol. 54, n° 1, p. 74-97, 1991.
- [ROS 66] ROSENFELD A., PFALTZ J., « Sequential operations in digital picture processing », *Journal of the ACM*, vol. 13, n° 4, p. 471-494, 1966.
- [ROS 68] ROSENFELD A., « Distance functions on digital pictures », *Pattern Recognition*, vol. 1, p. 33-61, 1968.
- [ROS 70] ROSENFELD A., « Connectivity in digital pictures », *Journal of the Association for Computer Machinery*, vol. 17, p. 146-160, 1970.
- [ROS 73] ROSENFELD A., « Arcs and curves in digital pictures », *Journal of the Association for Computer Machinery*, vol. 20, p. 81-87, 1973.
- [ROS 75] ROSENFELD A., « A converse to the Jordan curve theorem for digital curves », *Information and Control*, vol. 29, p. 292-293, 1975.
- [SAI 94] SAITO T., TORIWAKI J., « New algorithms for Euclidean distance transformation of an  $n$ -dimensional digitized picture with applications », *Pattern Recognition*, vol. 27, p. 1551-1565, 1994.
- [SAL 92] SALEMBIER P., SERRA J., « Morphological Multiscale Image Segmentation », MARAGOS P., (DIR.), *Visual Communications and Image Processing*, vol. SPIE-1818, p. 620-631, 1992.
- [SAP 93] SAPIRO G., KIMMEL R., SHAKED D., KIMIA B., BRUCKSTEIN A., « Implementing continuous-scale morphology via curve evolution », *Pattern Recognition*, vol. 26, n° 9, p. 1363-1372, 1993.
- [SCH 89] SCHMITT M., Des algorithmes morphologiques à l'intelligence artificielle, thèse de doctorat, Ecole nationale supérieure des Mines de Paris, février 1989.
- [SCH 94] SCHMITT M., MATTIOLI J., *Morphologie Mathématique*, Masson, 1994.
- [SER 82] SERRA J., *Image analysis and mathematical morphology*, Academic Press, Londres, Royaume-Uni, 1982.
- [SER 88a] SERRA J., « Examples of structuring functions and their uses », SERRA J., (DIR.), *Image analysis and mathematical morphology. Volume 2 : Theoretical advances*, Chapitre 4, p. 71-99, Academic Press, 1988.
- [SER 88b] SERRA J., (DIR.), *Image analysis and mathematical morphology. Volume 2 : Theoretical advances*, Academic Press, Londres, Royaume-Uni, 1988.
- [SER 92] SERRA J., VINCENT L., « An overview of morphological filtering », *Circuits Systems Signal Process*, vol. 11, n° 1, p. 47-108, 1992.
- [SET 96a] SETHIAN J., « A Fast Marching Level Set Method for Monotonically Advancing Fronts », *Proceedings of the National Academy of Sciences*, vol. 93(4), p. 1591-1595, 1996.
- [SET 96b] SETHIAN J., *Level Set Methods : Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision, and Materials Science*, Cambridge University Press, Cambridge, 1996.
- [SET 01] SETHIAN J., VLADIMIRSKY A., « Ordered upwind methods for static Hamilton-Jacobi equations », *Proceedings of the National Academy of Science*, vol. 98, n° 20, p. 11069-11074, 2001.
- [SHI 92] SHIH F. Y., MITCHELL O. R., « A mathematical morphology approach to Euclidean distance transformation », *IEEE Transactions on Image Processing*, vol. 1, p. 197-204, 1992.
- [SID 99] SIDDIQI K., BOUIX S., TANNENBAUM A., ZUCKER S., « The Hamilton-Jacobi Skeleton », *International Conference on Computer Vision (ICCV)*, p. 828-834, Corfu, Grèce, 1999.
- [SLE 85] SLEADOR D., TARJAN R., « Self-adjusting Binary Search Trees », *J. Assoc. Comp. Mach.*, vol. 32, p. 652-686, 1985.



- [SOI 91] SOILLE P., « Spatial distributions from contour lines : an efficient methodology based on distance transformations », *Journal of Visual Communication and Image Representation*, vol. 2, n° 2, p. 138-150, juin 1991.
- [SOI 92] SOILLE P., RIVEST J., SERRA J., « Dimensionality in image analysis and processing », *Proc. SPIE, Image Science and Technology*, vol. 1658, San Jose, CA, États Unis, février 1992.
- [SOI 01] SOILLE P., TALBOT H., « Directional morphological filtering », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, n° 11, p. 1313-1329, novembre 2001.
- [SOI 03] SOILLE P., *Morphological Image Analysis*, 2<sup>e</sup> édition, Springer-Verlag, Heidelberg, 2003.
- [STU 87] STUIK D. J., *A concise history of mathematics*, 4<sup>e</sup> édition, Dover, 1987.
- [TAL 92] TALBOT H., VINCENT L., « Euclidean skeletons and conditional bisectors », *Procs. VCIP'92, SPIE*, vol. 1818, p. 862-876, 1992.
- [TAL 07] TALBOT H., APPLETON B., « Efficient complete and incomplete paths openings and closings », *Image and Vision Computing*, vol. 25, n° 4, p. 416-425, avril 2007.
- [TAN 09a] TANKYEVYCH O., TALBOT H., DOKLADAL P., PASSAT N., « Direction-Adaptive Grey-level Morphology. Application to 3D Vascular Brain Imaging », *Proceedings of International Conference on Image Processing (ICIP) 2009*, p. 2261-2264, IEEE, Le Caire, Égypte, 2009.
- [TAN 09b] TANKYEVYCH O., TALBOT H., DOKLADAL P., PASSAT N., « Spatially-variant morpho-Hessian filter : efficient implementation and application », *Proceedings of International Symposium on Mathematical Morphology (ISMM) 2009*, p. 137-148, Groningen, Pays-Bas, 2009.
- [TAR 75] TARJAN R., « Efficiency of a good but not linear set union algorithm », *Journal of the ACM*, vol. 22, n° 2, p. 215-225, avril 1975.
- [THI 07] THIEL E., COEURJOLLY D., « Distances discrètes », COEURJOLLY D., MONTANVERT A., CHASSERY J.-M., (DIR.), *Géométrie discrète et images numériques*, p. 187-209, Hermès, 2007.
- [TOR 03] TORSSELLO A., HANCOCK E., « Curvature correction of the Hamilton-Jacobi skeleton », *International Conference on Computer Vision and Pattern Recognition*, vol. 1, p. 828-834, juin 2003.
- [TUR 36] TURING A., « On computable numbers, with an application to the Entscheidungsproblem », *Proceedings of the London Mathematical Society*, vol. 42, novembre 1936, Reprinted in the *Undecidable*, pages 115-154.
- [UND 70] UNDERWOOD E., *Quantitative Stereology*, Addison-Wesley, Reading, Mass, 1970.
- [URB 08] URBACH E., F. W. M. H., « Efficient 2-D Grayscale Morphological Transformations with Arbitrary Flat Structuring Elements », *IEEE Transactions on Image Processing*, vol. 17, n° 1, p. 1-8, janvier 2008.
- [VAC 01] VACHIER C., « Morphological scale-space analysis and feature extraction », *Proceedings of International Conference on Image Processing*, vol. 3, p. 676-679, 2001.
- [VAL 09] VALERO S., CHANUSSOT J., BENEDIKTSSON J., TALBOT H., WASKE B., « Directional Mathematical Morphology For the detection of the road network in very high resolution remote sensing images », *Proceedings of ICIP 2009*, Le Caire, Égypte, 2009.
- [VAN 96] VAN DROOGENBROECK M., TALBOT H., « Fast computation of morphological operations with arbitrary structuring elements », *Pattern Recognition Letters*, vol. 17, n° 14, p. 1451-1460, 1996.
- [VAN 05] VAN DROOGENBROECK M., BUCKLEY M., « Morphological erosions and openings : fast algorithms based on anchors », *Journal of Mathematical Imaging and Vision, special Issue on Mathematical Morphology after 40 Years*, vol. 22, n° 2-3, p. 121-142, mai 2005.
- [VER 08] VERDÚ-MONEDERO R., ANGULO J., « Spatially-Variant Directional Mathematical Morphology Operators Based on a Diffused Average Squared Gradient Field », *Advanced Concepts for Intelligent Vision Systems*, p. 542-553, 2008.

- [VIN 90] VINCENT L., Algorithmes morphologiques à base de files d'attente et de lacets. Extension aux graphes, thèse de doctorat, Ecole nationale supérieure des Mines de Paris, 1990.
- [VIN 91a] VINCENT L., « Efficient Computation of Various Types of Skeletons », LOEW M., (DIR.), *Medical Imaging V*, vol. 1445, p. 297-311, Society of Photo-Instrumentation Engineers, San Jose, CA, États Unis, 1991.
- [VIN 91b] VINCENT L., « Morphological transformations of binary images with arbitrary structuring elements », *Signal Processing*, vol. 22, n° 1, p. 3-23, janvier 1991.
- [VIN 91c] VINCENT L., SOILLE P., « Watersheds in digital spaces : an efficient algorithm based on immersion simulations », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, n° 6, p. 583-598, juin 1991.
- [VIN 92] VINCENT L., « Morphological area openings and closings for greyscale images », *Proc. Shape in Picture '92, NATO Workshop*, Springer-Verlag, Driebergen, Pays-Bas, septembre 1992.
- [VIN 93a] VINCENT L., « Grayscale area openings and closings, their efficient implementation and applications », SERRA J., SALEMBIER P., (DIR.), *Proc. EURASIP workshop on Mathematical morphology and its applications to signal processing*, p. 22-27, Barcelone, Espagne, mai 1993.
- [VIN 93b] VINCENT L., « Morphological grayscale reconstruction in image analysis : applications and efficient algorithms », *IEEE Transactions on Image Processing*, vol. 2, n° 2, p. 176-201, avril 1993.
- [VIN 94] VINCENT L., « Morphological area openings and closings for greyscale images », O Y.-L., TOET A., FOSTER D., HEIJMANS H., MEER P., (DIR.), *Shape in Picture : Mathematical Description of Shape in Grey-Level Images*, vol. 126 de *NATO ASI Series F*, p. 197-208, Springer-Verlag, 1994.
- [WEI 63] WEIBEL E., *Morphometry of the human lung*, Springer, Berlin, 1963.
- [WIL 06] WILKINSON M. H. F., « Attribute-space connectivity and connected filters », *Image and Vision Computing*, vol. 25, n° 4, p. 426-435, 2006.
- [XU 91] XU J., « Decomposition of Convex Polygonal Morphological Structuring Elements into Neighborhood Subsets », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, n° 2, p. 153-162, 1991.
- [ZAM 80] ZAMPERONI P., « Dilatation und Erosion von konturcodierten Binärbildern », *Microscopica Acta*, vol. Suppl. 4, p. 245-249, 1980.

---

# *Cours de Morphologie Mathématique*

## *Introduction à l'analyse d'images*

Hugues Talbot

talboth@esiee.fr

ISBS 2009-2010

---

Introduction à l'analyse d'images – p. 1/52

---

### Votre enseignant : Hugues Talbot

- ECP 1989, DEA Paris VI IA/RF 1990, Thèse morphologie mathématique MIT (Cambridge USA) + ENSMP (Fontainebleau) 1993 ;
- Expérience industrielle: SSII 1987-1990 (Analyse d'images, réseaux), Isover Saint-Gobain 1990-1994 (Analyse d'images) ;
- Expérience de recherche: CSIRO (Australie) 1994-2004 (Analyse d'images) ;
- Expérience enseignement: ECP 1989-1990 (réseaux), Université de Sydney 2004 ;
- A l'ESIEE depuis novembre 2004. Nombreuses collaborations industrielles : Sanofi, Serbier, Vilmorin, Philips, Siemens, GE etc.
- Responsable à l'ISBS : 3e année option Imagerie, Projets tutorés 2eA et 3eA, responsable des enseignements d'informatique.

Et vous ?

---

Introduction à l'analyse d'images – p. 2/52

# Planning 2009-2010

5	15h-15h55																		
5	16h05-17h					Comp. NG ????????	Chimie Poly.	Chimie Poly.			Chimie Poly.			Chimie Poly.	Chim				
7	Jeu	18-Sep	17-Sep	24-Sep	01-Oct	08-Oct	15-Oct	22-Oct	29-Oct	05-Nov	12-Nov	19-Nov	26						
8	2h-3h	Anglais SH45	Anglais SH45	Anglais SH45	Anglais SH45	Anglais SH45	Anglais SH45	Anglais SH45		Anglais SH45	Anglais SH45	Anglais SH45	Angl						
9	10h-10h55																		
1	11h05-12h		Mécanique 2	Mécanique 2		Mécanique 2		Mécanique 2				Mécanique 2	Meca						
2	12h-13h																		
3	14h-15h																		
4	15h-15h55	Projet Perso	Projet Perso	Projet Perso	Projet Perso	Projet Perso	Projet Perso	Projet Perso		Projet Perso	Projet Perso	Projet Perso	Pr						
5	16h05-17h																		
6	17h-18h																		
7	Vendredi	11-Sep	18-Sep	25-Sep	02-Oct	09-Oct	16-Oct	23-Oct	30-Oct	06-Nov	13-Nov	20-Nov	27						
8	8h-9h		Theo Inf Appl (TIA)	Theo Inf Appl (TIA)	Theo Inf Appl (TIA)	Theo Inf Appl (TIA)	Theo Inf Appl (TIA)	Theo Inf Appl (TIA)		Theo Inf Appl (TIA)	Theo Inf Appl (TIA)	BDR	B						
9	9h-9h55																		
1	10h05-11h																		
2	11h-12h																		
3	13h-14h	MORPHO	MORPHO	MORPHO	MORPHO	MORPHO	MORPHO	MORPHO		MORPHO	Initiation Bioimp.	MORPHO	Int. Bi						
4	14h-14h55																		
5	15h05-16h																		
6	16h-17h																		
8																			
9		INTERCOURS = 10 minutes de pause entre chaque changement de cours								Italique : cours à l'ESIEE (8h-12h et 13h-17h)									
1		? Epreuve terminale TIA								Gras : cours à Paris 12									
2		? Epreuve terminale COM ESIEE																	

- 14h cours, 7h TD, 8h TP.
- Contrôle continu. Examen ? Projet ?

## Contenu général du cours

1. Théorie
  - Introduction à l'analyse d'images
  - Transformées morphologiques de base
  - Filtrage non-linéaire
  - Techniques de segmentation
  - Mesure, stéréologie.
2. Applications
  - Comptage d'objets (e.g. cytologie)
  - Automatisation de mesures
  - Applications médicales.
3. TPs & TDs
4. Évaluations: TDs + Examen livre ouvert.

---

## Suite des cours prévue

---

1. **Introduction à l'analyse d'image ;**
2. Érosion, dilatation ;
3. Ouverture, fermetures ;
4. Résidus: gradients, chapeaux ;
5. Reconstruction, filtres algébriques ;
6. Transformées en tout-ou-rien, squelettisation ;
7. Distances, géodesie ;
8. Segmentation - Ligne de partage des eaux ;
9. Notions de stéréologie.

---

## Qu'est-ce que l'analyse / le traitement d'images

---

- Deux choses différentes: le traitement d'image consiste à modifier (filtrer) une image.
- Le but du traitement d'images est de rendre apparent ou de cacher des choses dans l'image.
- Le but de l'analyse d'image est de mesurer des choses par le biais d'une image.
- Ne pas confondre avec la vision par ordinateur dont le but est d'interpréter le contenu d'une image.
- Au cours de l'action d'analyse on utilise beaucoup de techniques de traitement (e.g. débruitage, reconnaissance des formes, etc).

---

Introduction à l'analyse d'images – p. 7/52

---

## Exemples de traitements d'images

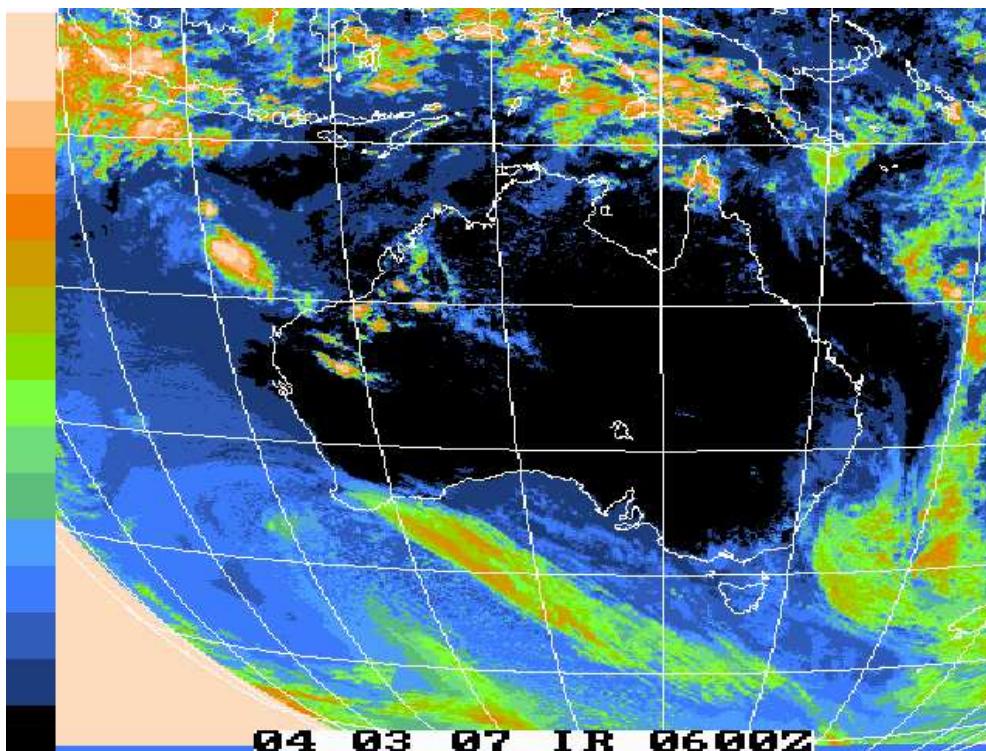
---

- Rendu fausses couleurs
- Retrait du flou (deblurring)
  - spatial
  - de mouvement
  - du à des couches optiques.
- Améliorations
  - Contraste/luminosité (manipulation d'histogrammes)
  - Retrait du bruit
  - Transformées géométriques (rotation, changement d'échelle, affine, etc)
- Codage (compression)
- Extraction de caractéristiques (e.g. texture)

---

Introduction à l'analyse d'images – p. 8/52

## Exemple d'images en fausses couleurs



Source: Australian Bureau of Meteorology

Introduction à l'analyse d'images – p. 9/52

## Exemple of retrait du flou



Source: *Unshake* <http://www.hamangia.freeseve.co.uk/>

Introduction à l'analyse d'images – p. 10/52

---

## Exemple of retrait du flou

---

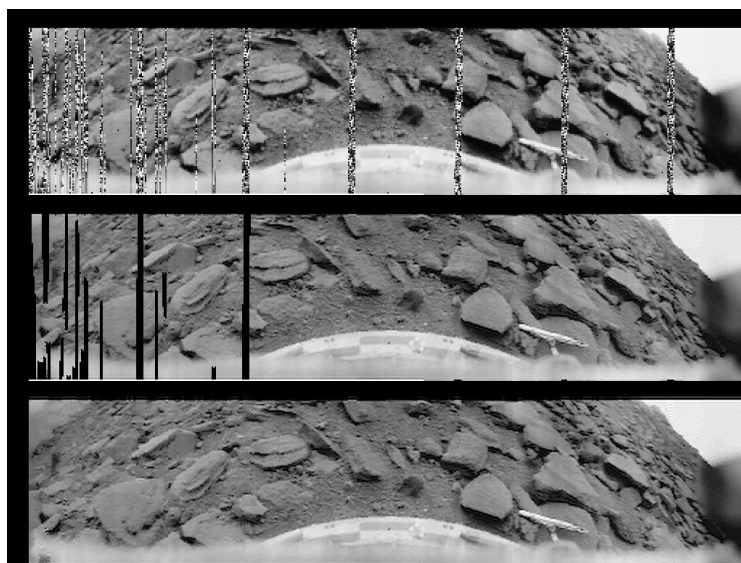


Source: *Unshake* <http://www.hamangia.freemove.co.uk/>

---

## Exemple d'interpolation de données avancé

---



Venera 9, images of Venus: Combination of multiple views, inpainting by PDE flow.

Source: [http://www.mentallandscape.com/V\\_DigitalImages.htm](http://www.mentallandscape.com/V_DigitalImages.htm)



---

## La loi de Murphy marche sur Vénus



Venera 13, camera 1, échantillonnage du sol

---

Introduction à l'analyse d'images – p. 12/52

---

## La loi de Murphy marche sur Vénus



Venera 14, camera 1, échantillonnage du couvercle de caméra

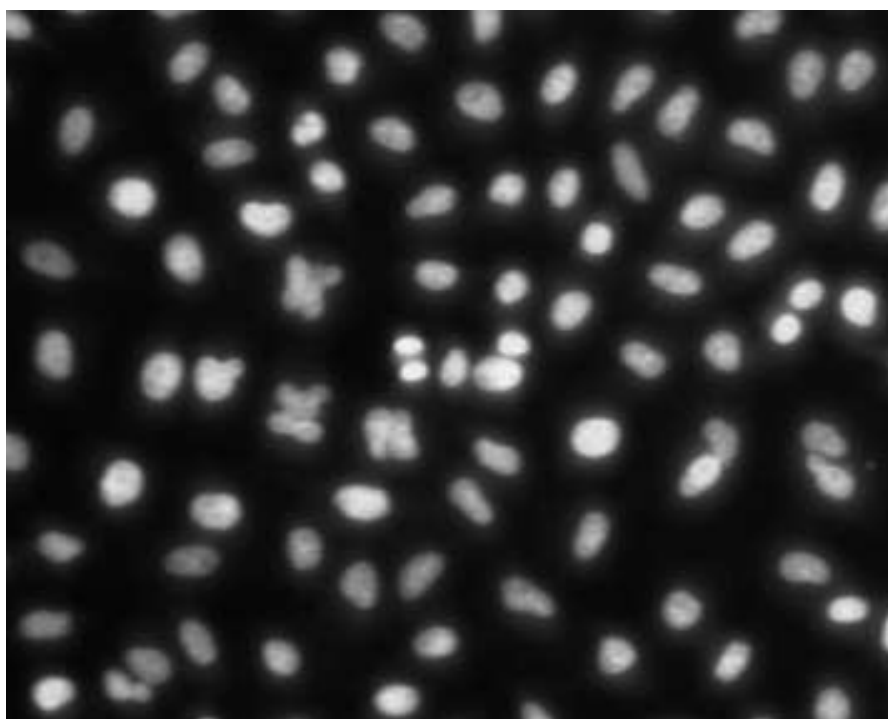
---

Introduction à l'analyse d'images – p. 12/52

---

## Exemples d'analyse: comptage de cellules

---

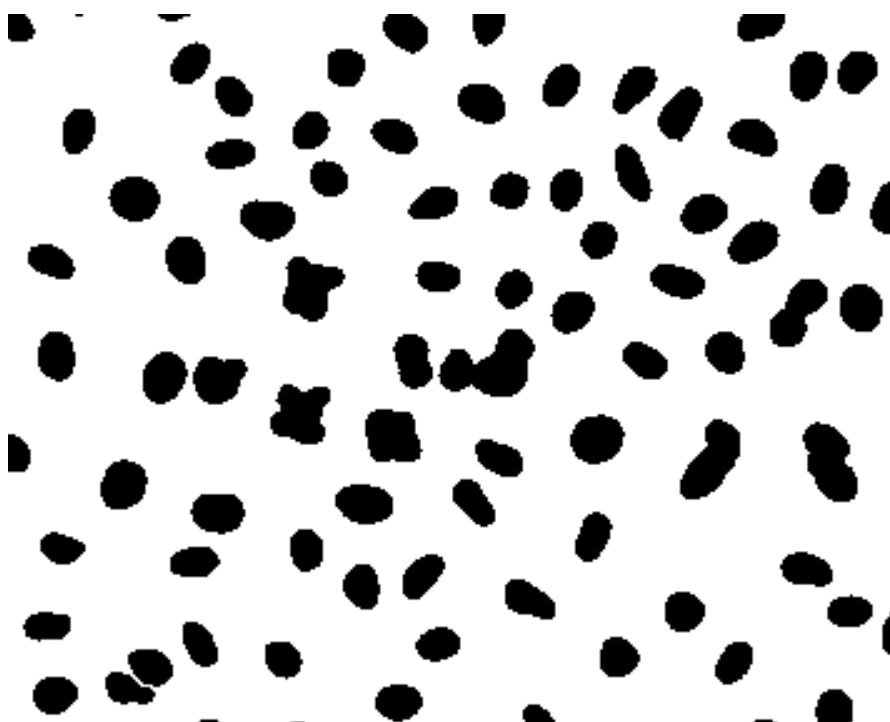


Cell nuclei

---

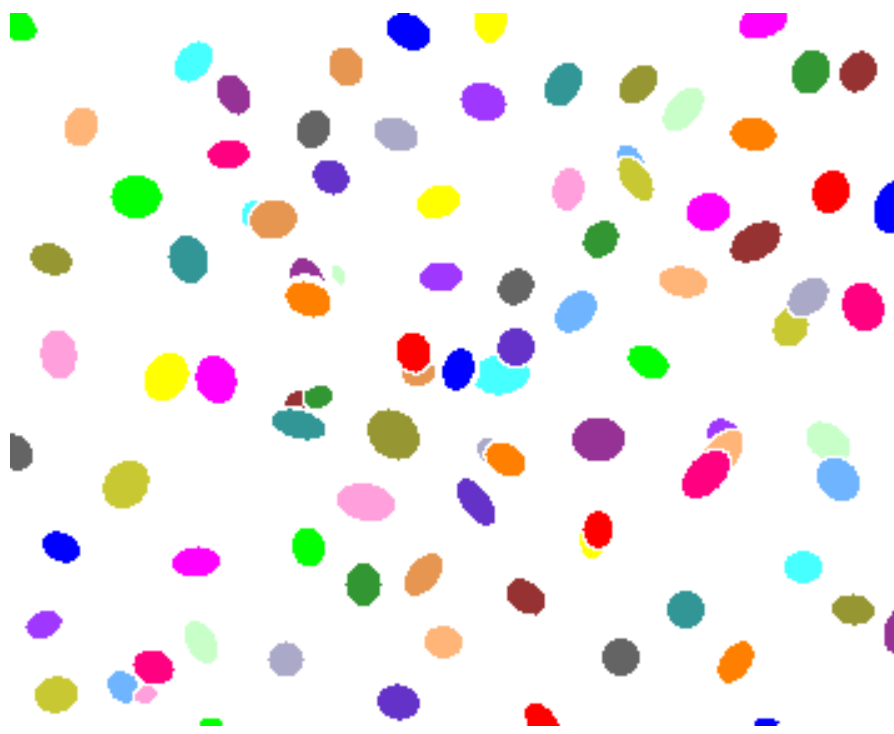
## Exemples d'analyse: comptage de cellules

---



Segmentation

## Exemples d'analyse: comptage de cellules



Object splitting

Introduction à l'analyse d'images – p. 13/52

## Pourquoi traiter les images ?

- Le traitement d'image recouvre de nombreux champs:
  - Transmission de données (codage, compression, steganography);
  - Filtrage (améliorations, retrait du flou, effets spéciaux)
  - Compréhension du contenu (segmentation, reconnaissance des formes, intelligence artificielle).
- Aires d'application:
  - Science (médecine, science des matériaux, biologie, astronomie, télédétection, etc)
  - Police (reconnaissance des visages, hyper-resolution)
  - Arts (effets spéciaux, restauration de films, communications)

Introduction à l'analyse d'images – p. 14/52

---

## Pourquoi traiter les images (cont.)?

---

- Lorsque ça vaudrait le coup économiquement:
  - Automatisation des tâches visuelles (surveillance, contrôle de qualité, vision industrielle)
  - Multiplicité des capteurs (les caméras sont partout)
  - Réalité augmentée (IR, UV, rayons X)
- Lorsqu'on n'a pas le choix:
  - Télescope Hubble, au début (avant la première mission de maintenance),
  - Robots lointains ou dans les endroits dangereux (sur Mars par exemple)
  - Données médico-légales (au sens le plus large)
- Lorsqu'on veut utiliser les données qu'on a au maximum:
  - Capteurs à haute résolution spatiale et niveaux de gris,
  - Données Multi-spectrales,
  - Rendu 3D,
- Et bien plus encore.

*Bases de l'imagerie: formation de l'image*

---

## Qu'est ce qu'une image ?

---

- En général on traite une image comme un tableau à deux dimensions de données uniques (images en niveaux de gris) ou encore de triplets de valeurs, typiquement R-G-B. Un élément de l'image est appelé un pixel (**p**icture **e**lement).
- Une image peut avoir plus de 2 dimensions ;
- On peut avoir une suite temporelle d'images (films) ;
- Chaque pixel d'une image peut être associé à un vecteur de valeurs (images multi et hyper-spectrales) ;
- Une image peut être constitué d'une collection de tableaux connectés spatialement mais de sens variés, appelés *composantes*.
- Un ensemble d'images peuvent être liés spatialement ou sémantiquement (paires d'images stéréo, panorama).

Pour des raisons de simplicité nous nous contenterons du cas 2-D en N&B ou en couleur, pour la plupart de ce cours.

---

## Que représente une image

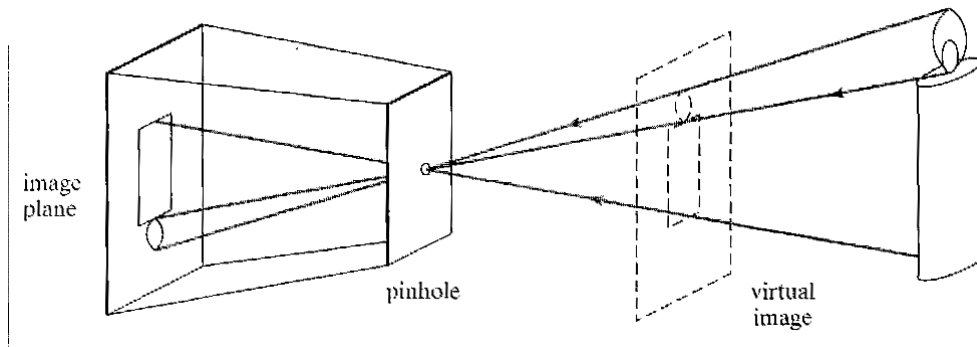
---

- Une projection de la réalité sur une surface ou un volume
- Image mathématique:

$$I = f(R)$$

- La fonction  $f$  est loin d'être simple.
- Il est utile de s'attarder sur le problème de la formation d'image et de la perception.

## Formation d'image: la caméra idéale

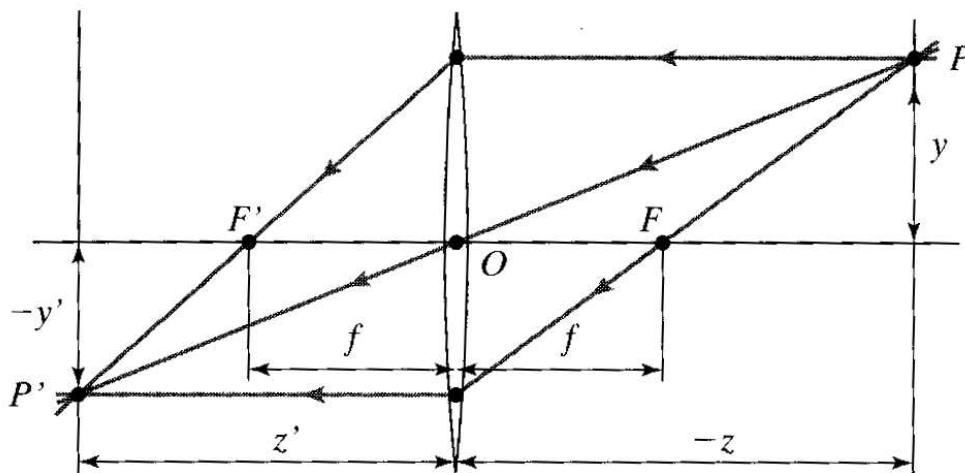


Advantages: Géométrie simple, toujours au point, modèle utile.

Inconvénients: Trop simple: pas d'effets optiques (diffraction, aberrations).

Introduction à l'analyse d'images – p. 19/52

## La caméra à lentille mince



Beaucoup plus réaliste, au moins pour certains instruments (e.g: télescopes).

Problèmes associés:

- Profondeur de champ, ouverture,
- Champ,
- Diffraction, etc.

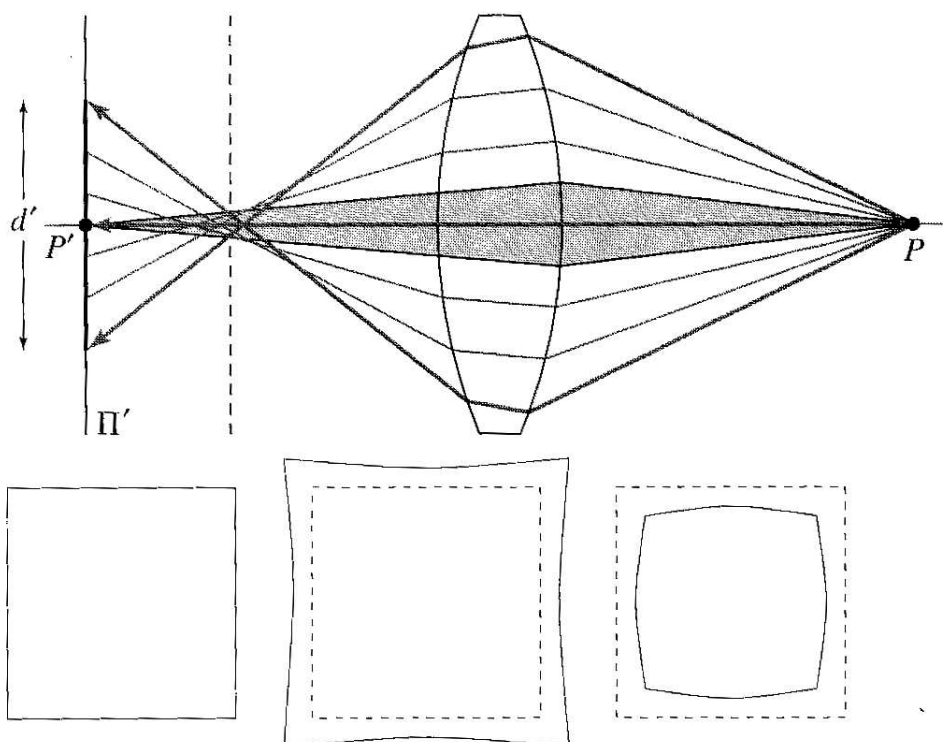
Introduction à l'analyse d'images – p. 20/52

# Caméras réelles

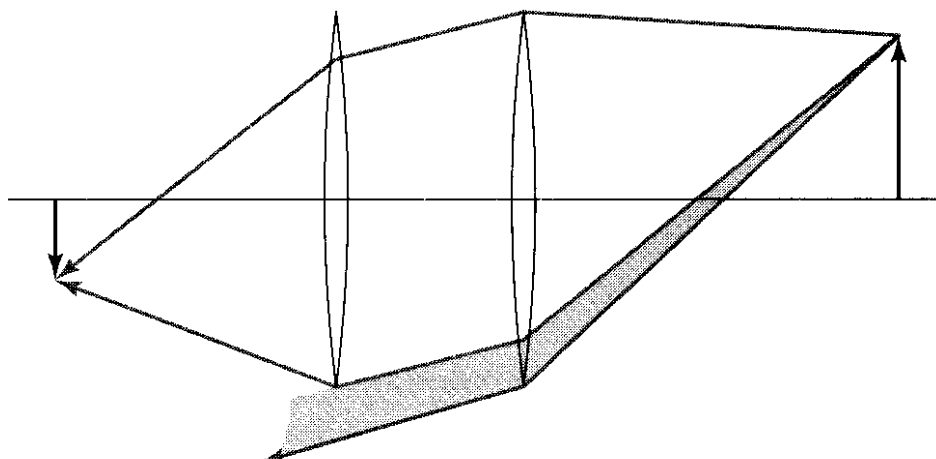
Les vraies caméras ont:

- Des lentilles multiples et épaisses
- Possèdent des aberrations géométriques: lentille sphériques, tonneaux, coussin, vignettage, etc.
- Des aberrations chromatiques due à la différence de réfraction suivant la longueur d'onde.

## Aberration sphérique



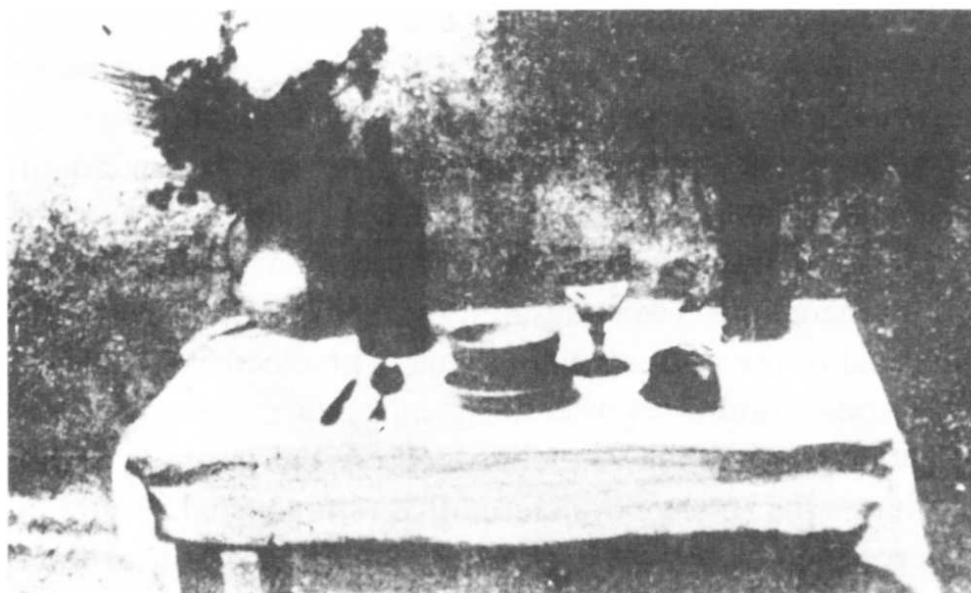
## Vignettage



Introduction à l'analyse d'images – p. 23/52

## Capteurs: photos

Première photographie: ca 1816, maintenant perdue. Une des plus vieilles photos enregistrée est la suivante (original perdu à la fin du 19ème siècle)



Introduction à l'analyse d'images – p. 24/52



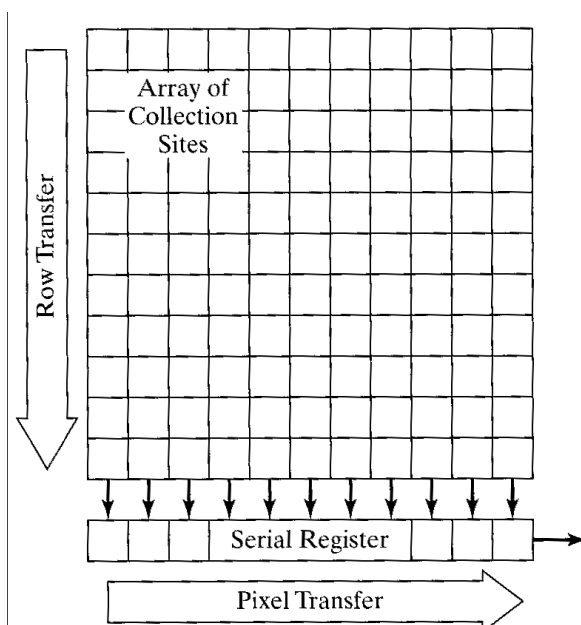
## Historique des capteurs

- Premières photos: 1822
- Daguerrotypes: 1839 (popular photography)
- Plaque humide: 1850 – procédé positif/négatif mais traitement immédiat nécessaire.
- Processus à gélatine : 1870 – image latente
- Film: 1889 by Eastman
- Cinéma : 1895 frères Lumière
- Photo couleur : 1908
- Télévision: 1920s

Introduction à l'analyse d'images – p. 25/52

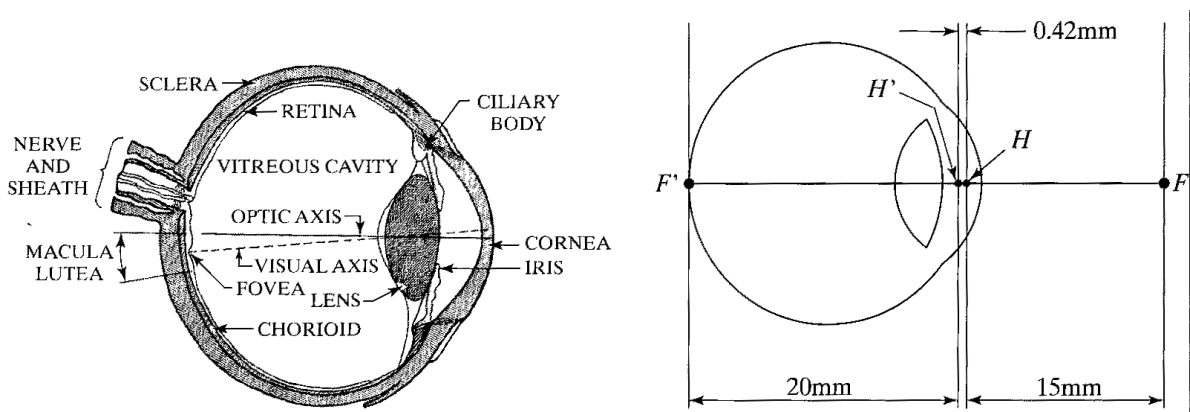
## Capteurs: CCDs

Inventés en 1970, utilisés en TV et astronomie. Trouvés partout maintenant, en train d'être remplacés par des senseurs CMOS pour l'électronique grand public.



Introduction à l'analyse d'images – p. 26/52

# L'oeil



Cônes et battonets, perception de la couleur.  
Neurones / nerf optique,  
Contrôle de la brillance, suivi, filtrage et interprétation  
Complexe physiologie.

# La lumière

- La lumière visible est un sous-ensemble du spectre electro-magnétique, de longueur d'ondes de 380 à 720 nm.
- Une source de lumière est caractérisée par (i) sa puissance et (ii) son spectre.
- Une source monochromatique est caractérisée par sa longueur d'onde  $\lambda$  et sa luminance  $L$ .

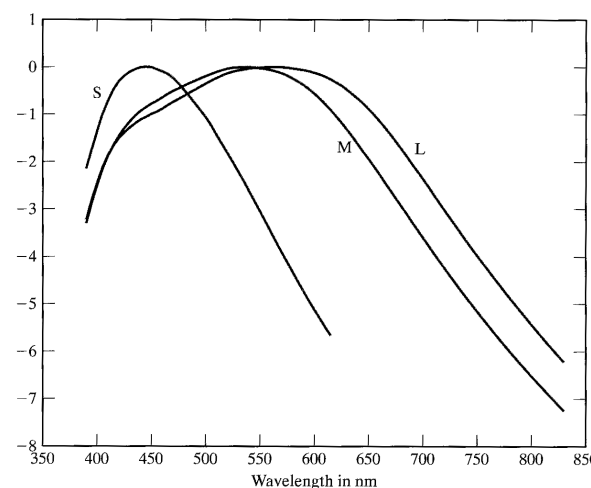
## La vue

- Ne peut distinguer les fréquences individuelles (au contraire de l'ouïe).
- Obéi la loi de GRASSMAN:

$$\sum_{i=1}^N L_i = L_w + L_r$$

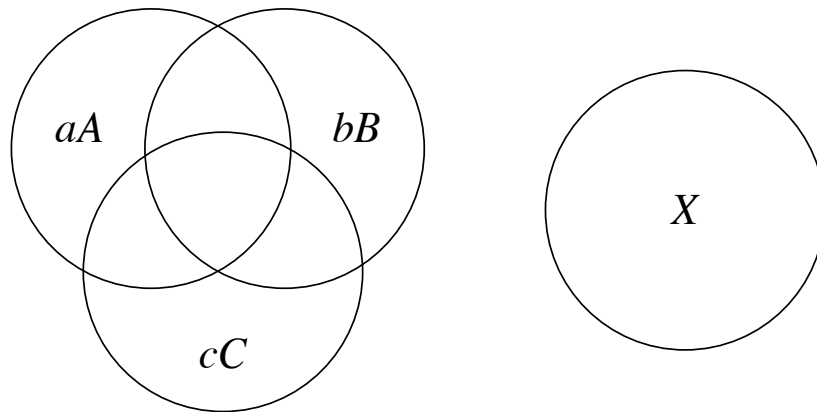
c-à-d: L'addition de  $N$  couleurs est équivalent à l'addition d'un flux blanc  $L_w$  et d'un flux d'une seule couleur monochromatique  $L_r$  de longueur d'onde  $\lambda_r$  (3 variables).

## Perception de la couleur



- Il y a trois types de capteurs couleur dans l'oeil (les cônes) ;
- Chaque type à une réponse spectrale différente.
- Ce qui suggère une représentation à 3 stimulus. On peut en effet reproduire la plupart des couleurs vues dans la nature à partir d'un système à trois sources monochromatiques de longueur d'onde différentes.

## Expérience de comparaison des couleurs



- $A$ ,  $B$  and  $C$  sont 3 couleurs “primaires” (sources de lumière projetées sur un écran),
- $a$ ,  $b$ ,  $c$  sont 3 facteurs de pondération,
- À partir d’expériences, on cherche à obtenir la couleur  $X$  à partir de  $aA + bB + cC$ .
- Si une comparaison exacte est impossible, une ou deux des couleurs primaires sont ajoutées à  $X$ , ce qui donne l’équivalent d’un facteur négatif pour  $a$ ,  $b$  ou  $c$ .

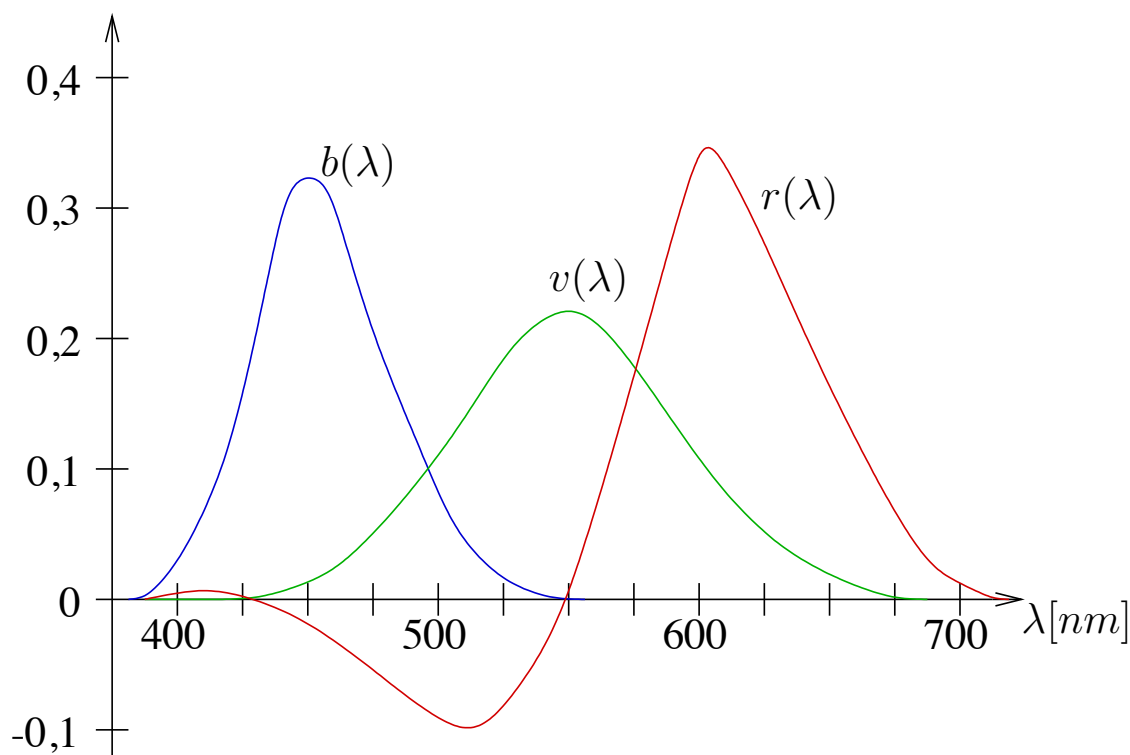
Introduction à l’analyse d’images – p. 31/52

## Standard CIE RGB

- In 1931, la Commission Internationale de l’Éclairage (CIE) à fait l’expérience de la comparaison des couleurs avec 3 sources monochromatiques: rouge (645.2 nm), vert (526.3.1 nm) et bleu (444.4 nm).
- Ce qui a donné le système CIE RGB.
- Avec ce système on peut reproduire la plupart des couleurs naturelles.
- Certaines des longueurs d’ondes réclame des pondérations négatives.

Introduction à l’analyse d’images – p. 32/52

## Fonctions de pondération



Introduction à l'analyse d'images – p. 33/52

## Standard CIE XYZ

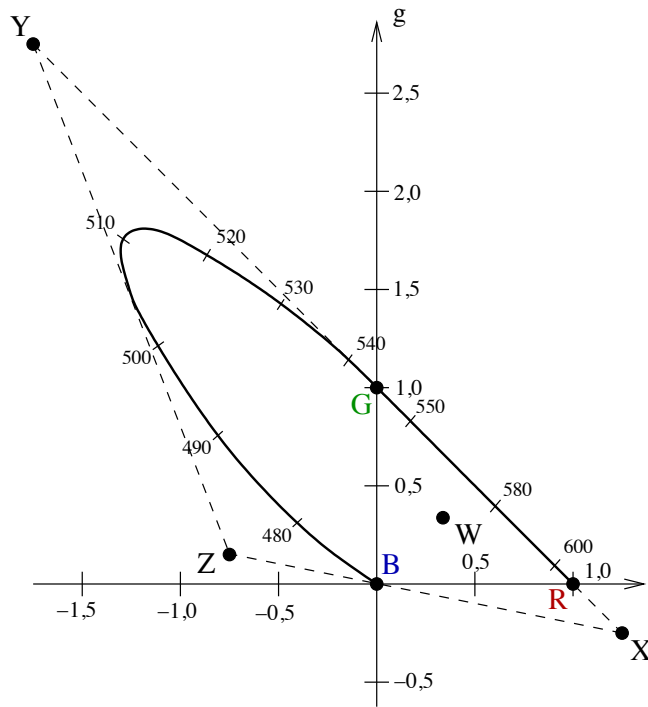
- Pour éviter les poids négatifs, la CIE a conçu un système tri-stimulus dérivé de RGB, appelé XYZ, où tous les poids sont positifs.
- Conversion RGB / XYZ par combinaison linéaire :

$$(1) \quad \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 2,769 & 1,7518 & 1,13 \\ 1 & 4,5907 & 0,0601 \\ 0 & 0,0565 & 5,5943 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

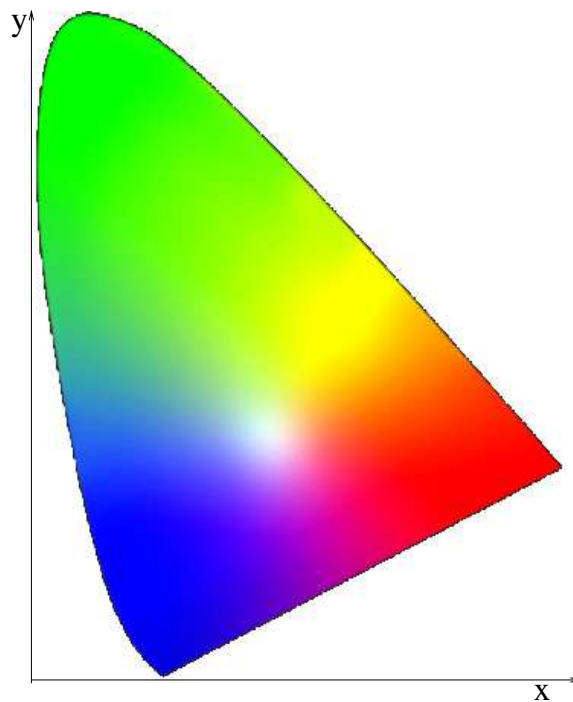
- Le plus souvent on utilise en fait le système  $xyz$  qui est normalisé par rapport à  $X + Y + Z$ , c-à-d :  $x = \frac{X}{X+Y+Z}$ , pour des questions de représentabilité.
- Dans ce système il n'y a que 2 variables indépendentes.

Introduction à l'analyse d'images – p. 34/52

# XYZ system



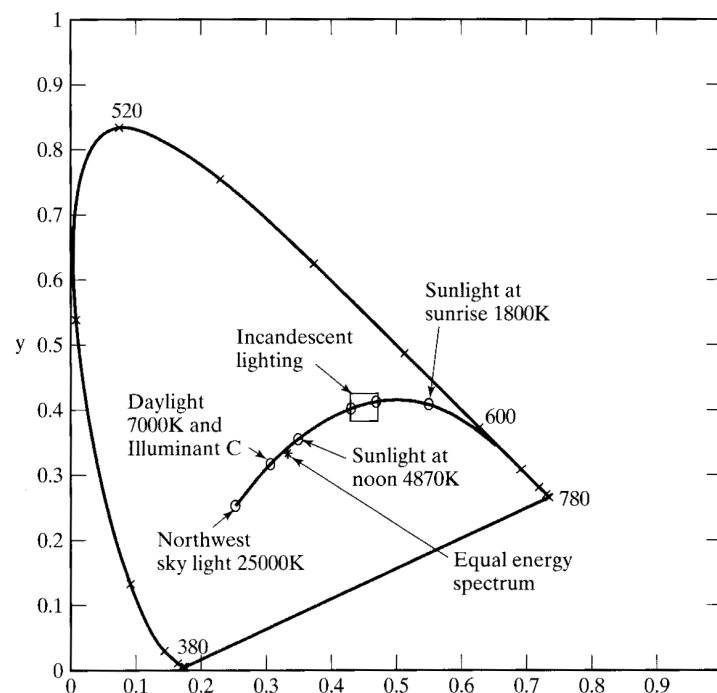
# $xy$ representation



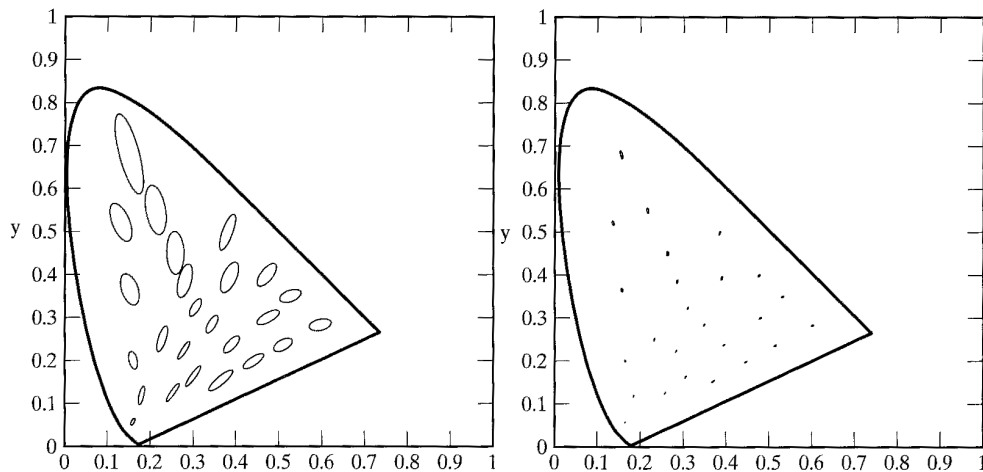
## Plus sur $xy$

- Les couleurs monochromatiques sont le long du U inversé.
- Il y a un point d'égalité d'énergie à  $x = y = 1/3$ .
- $xy$  est une version réduite de l'ensemble complet des couleurs en 3D.
- Le diagramme de chrominance  $xy$  est le lieu des chrominances de luminosité maximale.
- Bien que présentant de nombreux avantages au niveau de l'analyse, d'autres systèmes décomposent les couleurs en teinte (Hue), Saturation et brillance (Brightness).

## Température de couleur



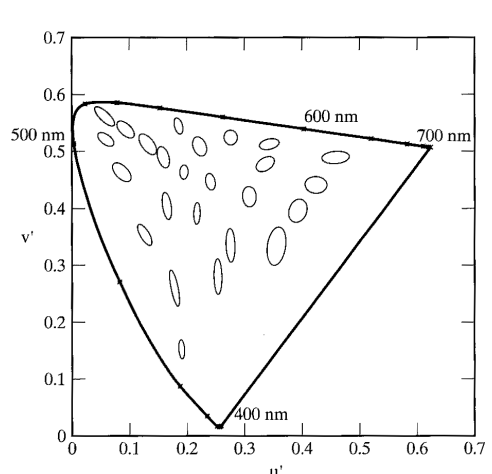
## Perception des différences de couleur



- Une autre expérience: de combien doit-on se déplacer dans l'espace des couleurs pour se rendre compte d'une différence?
- Peut-on créer un espace de représentation des couleurs où les différences sont homogènes ? Un espace *uniforme* ?
- C-à-d: peut-on réduire ces ellipses (les ellipses de MacAddam) en cercles ?

Introduction à l'analyse d'images – p. 39/52

## Système CIE $u'v'$



- Transformée linéaire simple:

$$(u', v') = \left( \frac{4X}{X + 15Y + 3Z}, \frac{9Y}{X + 15Y + 3Z} \right)$$

- Ceci ignore les différences de luminosité
- Diverses transformées non-linéaires ont été proposées.

Introduction à l'analyse d'images – p. 40/52



## Systeme CIE Lab

- Parmi ces systemes, le systeme Lab.

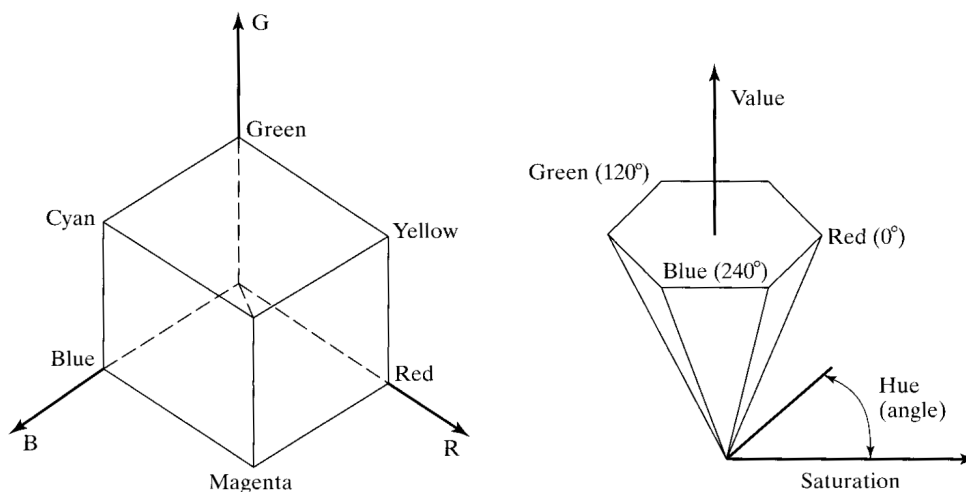
$$L^* = 116 \left( \frac{Y}{Y_n} \right)^{\frac{1}{3}} - 16$$

$$a^* = 500 \left[ \left( \frac{X}{X_n} \right)^{\frac{1}{3}} - \left( \frac{Y}{Y_n} \right)^{\frac{1}{3}} \right]$$

$$b^* = 200 \left[ \left( \frac{Y}{Y_n} \right)^{\frac{1}{3}} - \left( \frac{Z}{Z_n} \right)^{\frac{1}{3}} \right]$$

- Ici  $X_n, Y_n, Z_n$  sont les coordonnees d'un point blanc de reference.
- Lab est important car substantiellement uniforme.

## RGB vs HSV



---

## Systeme CYM substractif

---

- RGB et ses dérivatifs sont des systèmes de représentation des couleurs *additifs*.
- Pour les média imprimés, les couleurs ajoutées à la page soustraient des longueurs d'ondes.
- Le système pour l'impression s'appelle *CYM* (Cyan, Yellow, Magenta). Quand on les ajoute on produit du noir.
- Le plus souvent on ajoute le noir *K* séparément : *CYMK* car reproduire un noir parfait par soustraction est difficile et coûteux en encre.

---

## Représentation couleur de tous les jours

---

- La plupart des ordinateurs, etc utilisent un système *RGB* non calibré (*non CIE*), avec généralement une transformée non-linéaire de la luminosité (le gamma).
- La plupart des imprimantes personnelles utilisent un système *CYMK* non calibré.
- De ce fait: gamme de couleur limitée, problèmes de comparaison, etc. Exception: Apple.
- La TV utilise un système différent appelé  $YC_bC_r$  (luminance/chrominance).

---

## Autres points sur la perception

---

- Effets spatiaux et temporels sont ignorés
  - Adaptation chromatique: capacité du système visuel à s'adapter à une couleur dominante.
  - Assimilation: influence des couleurs environnantes en direction de certaines couleurs.
  - Contraste: influence des couleurs environnantes à s'éloigner d'autres couleurs.

---

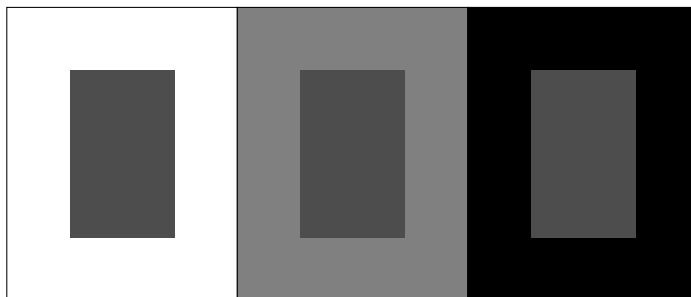
## Autres points sur la perception (suite)

---

- Différences individuelles. Environ 10% des garçons possèdent une forme ou une autre de défaut de perception couleur. Il leur manque un type de cône ou plus, le plus généralement les cônes rouges. Il est théoriquement possible que certaines filles possèdent 4 types de cônes (donc une meilleure perception de la couleur que la population générale), bien que cette hypothèse n'ait jamais été encore confirmée. Des défauts plus graves mais beaucoup plus rares touchent les deux sexes de manière identique.
- L'illumination est un facteur important. Nous sommes tous aveugle à la couleur la nuit (les bâtons sont bien plus sensibles que les cônes).
- Bien d'autres choses encore...

---

## Test de perception



Which is the darkest inner rectangle?

---

## Test d'adaptation chromatique



---

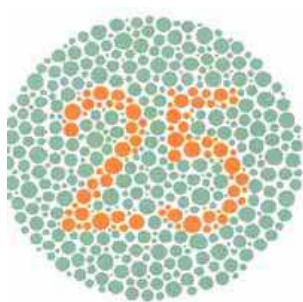
# Test d'adaptation chromatique

---

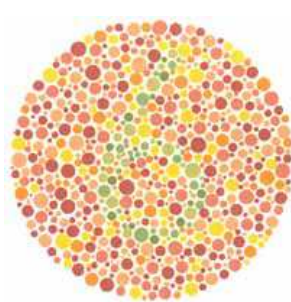
---

# Test de perception couleur

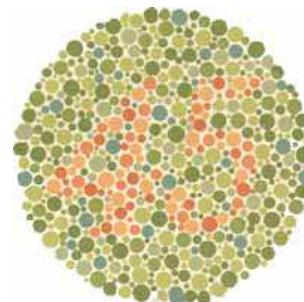
---



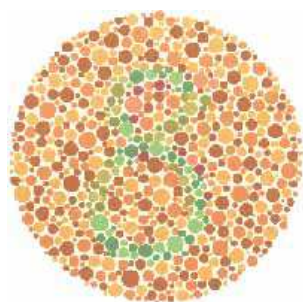
(a)



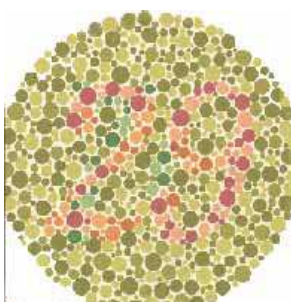
(b)



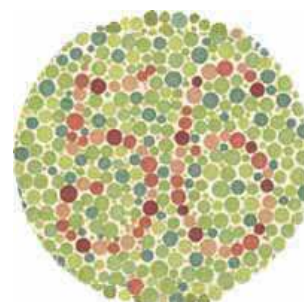
(c)



(d)



(e)



(f)

---

## Application: diagnostic du mélanome

---

Appareil Polartechnics Solarscan<sup>tm</sup>, see <http://www.polartechnics.com.au>.



Solarscan device



Hand piece

---

Introduction à l'analyse d'images – p. 50/52

---

## Segmentation de la couleur absolue

---



Uncalibrated lesion 1

---

Introduction à l'analyse d'images – p. 51/52

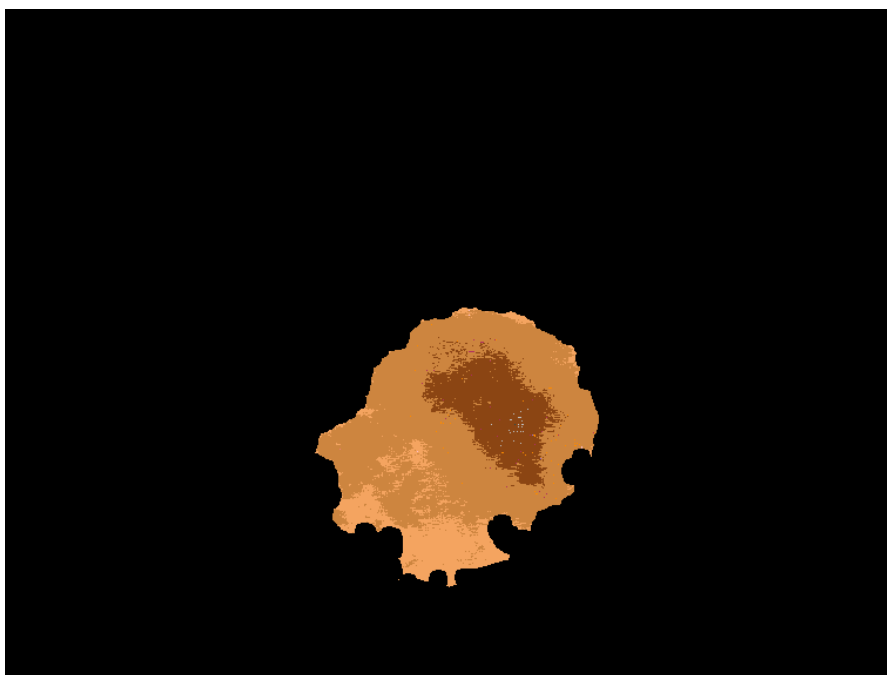
## Segmentation de la couleur absolue



Calibrated lesion 1

Introduction à l'analyse d'images – p. 51/52

## Segmentation de la couleur absolue



Colour identification for lesion 1 -> benign

Introduction à l'analyse d'images – p. 51/52

---

## Segmentation de la couleur absolue



Uncalibrated lesion 2

---

## Segmentation de la couleur absolue



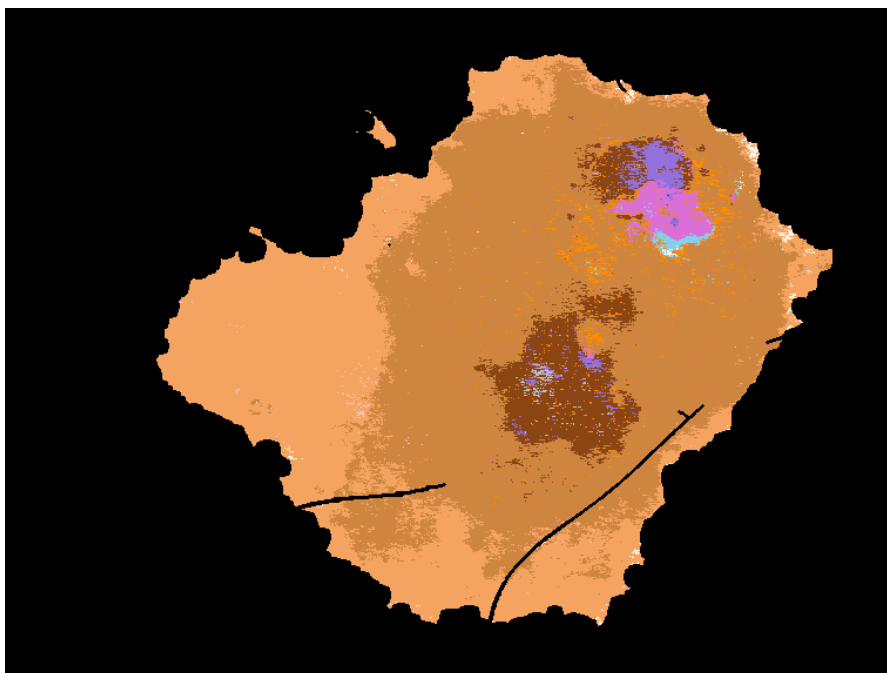
Calibrated lesion 2



---

## Segmentation de la couleur absolue

---



Colour identification for lesion 2 -> melanoma

---

Introduction à l'analyse d'images – p. 51/52

---

## Références et remerciements

---

D.A. Forsyth and Jean Ponce, “Computer Vision, a Modern Approach”.  
Prentice-Hall, USA 2003.

Marc Van Droogenbroeck. “Cours d'Analyse d'Images”. Université de Liège,  
2001-2003.

---

Introduction à l'analyse d'images – p. 52/52

---

*Cours de Morphologie Mathématique*  
*Concepts de base*

Hugues Talbot  
talboth@esiee.fr

ISBS 2009-2010

---

Concepts de base – p. 1/24

---

## Ressources

- Le cours sera mis en ligne au fur et à mesure:  
<http://www.esiee.fr/~talboth/ISBS/Morpho/>
- Photocopie des transparents en retard cause modifications.

---

Concepts de base – p. 2/24

---

## Questions

---

1. Questions sur le cours précédent ?
2. Remarques ?
3. Questions sur la suite du cours?

## *Représentation d'images*

## Tableaux en 2-D de données 8-bit

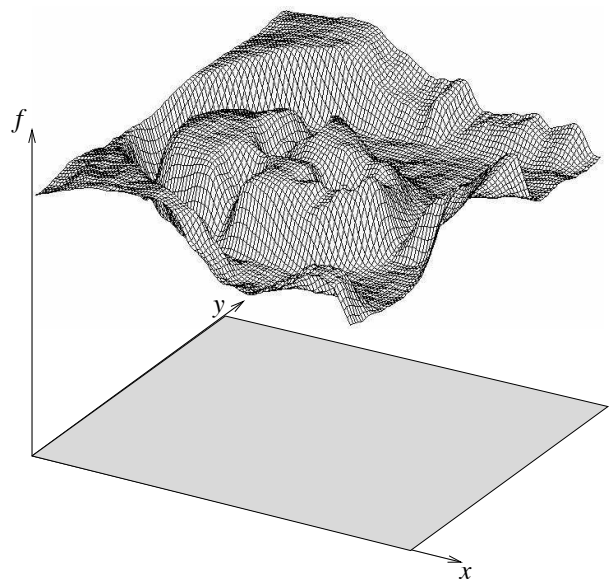
- Représentation la plus simple: collection of 1 (N&B) or 3 (Couleur) tableau(x) 2-D en 8 bits.
- Suffisant pour représenter la plupart des niveaux de gris visibles (256) et des couleurs visibles ( $255^3 = 16$  millions).
- La plupart des problèmes possibles peuvent être illustrées dans ce contexte.

Concepts de base – p. 5/24

## Fonction 2-D



Image



Fonction

Concepts de base – p. 6/24

## Résolution



full



half



1/4



1/8

## Nombre de niveaux de gris



256 gl



64 gl



16 gl



8 gl

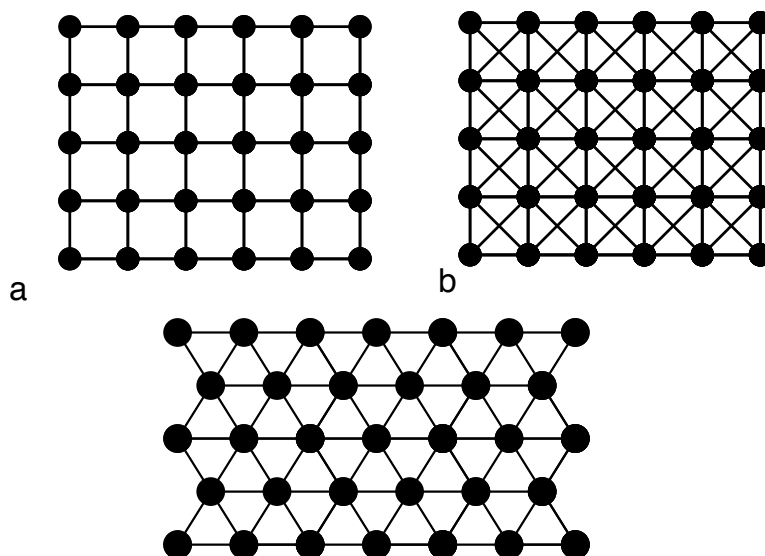


4 gl



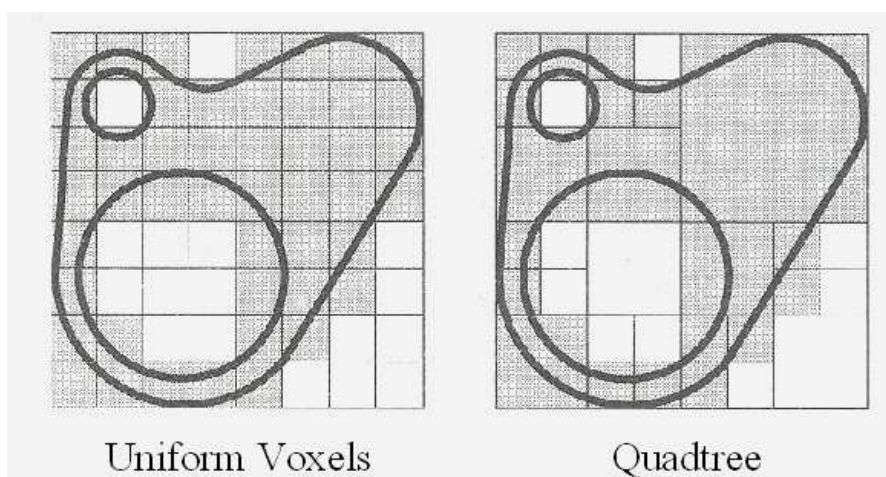
2 gl

# Graphe



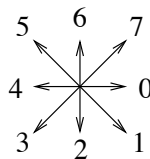
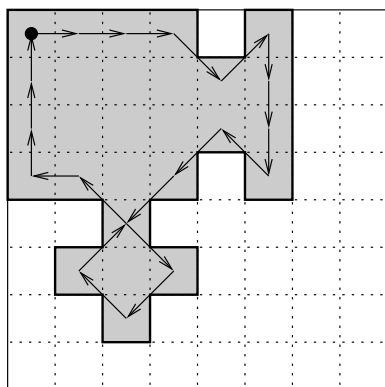
Synonymes: graphe de connexion, grille.

# Autres représentations: Quadrees



Les quadrees sont une représentation hiérarchique de la structure d'une image. L'idée de base étant qu'un quadrant peut être découpé récursivement en 4 autres sous-quadrants.

## Autre représentation: chaîne de contour



Code : 00017222533135754666

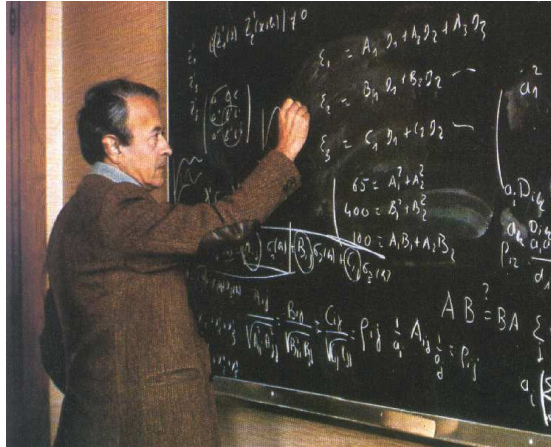
Permettent de représenter les contours de façon efficace.

---

## Un peu d'histoire

---

- La morphologie mathématique (MM) a été conçue au milieu des années 60 à l'ENSM, au laboratoire de Fontainebleau.
- Les inventeurs principaux ont été Georges Matheron et son étudiant en thèse Jean Serra. GM est mort en l'an 2000. JS fait maintenant partie du laboratoire A2SI à l'ESIEE.



---

Concepts de base – p. 13/24

---

## Encore un peu d'histoire

---

- Le nom “Morphologie Mathématique” à été choisi dans un bar.
- La morphologie mathématique est devenue reconnue internationalement seulement après la publication du livre de Serra en Anglais de 1982 et les articles de Haralick/Sternberg/Zhuang dans IEEE PAMI en 1987.

---

Concepts de base – p. 14/24



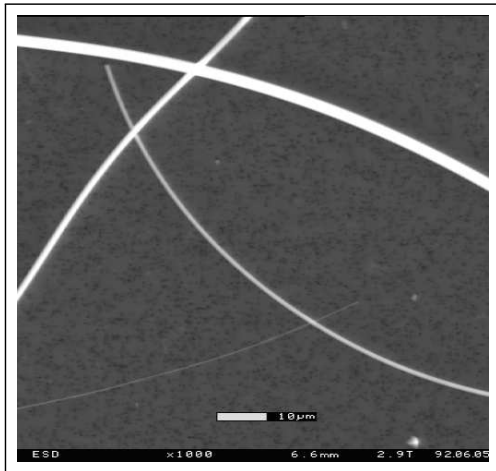
## Histoire (3)

- Certains concepts datent de Minkowski (1901), Birkhoff (1948) et Hawidger (1957).
- Le développement de la MM est très actif:
  - Conférences (ISMM, Scale-space, DGCI)
  - Une société pour la MM (ISMM also...)
  - Des journaux (JMIV, PAMI, etc)
- Intégrée dans une nouvelle série de théories
  - Scale-space
  - EDPs pour la segmentation et le filtrage.

## Démarches en analyse d'images

	Espace "géométrique"	Espace de données
Linéaire	Convolution Fourier, Ondelettes Tomographie (recons. 3-D)	Analyse multivariée Réseaux de neurones Stéréologie
Non-linéaire	Morphologie Géométrie discrète Ensembles aléatoires Segmentation	Démarches syntactiques Grammaires Indexation

## Rappel de ce qu'on veut faire ?



Concepts de base – p. 17/24

## Caractéristiques de la MM

Synergie entre la théorie/les algorithmes/les applications

- La première théorie non-linéaire du traitement d'images.
- Beaucoup d'algorithmes sont rapides et efficaces.
- Beaucoup d'applications.

Concepts de base – p. 18/24

---

## Concept de base: l'ordre

---

- La relation d'ordre est au coeur de la MM.
- Relation d'ordre:
  - $a \leq a$
  - $a \leq b, b \leq a \Rightarrow a = b$
  - $a \leq b, b \leq c \Rightarrow a \leq c$
- Notion de sup=plus petit majorant.
- Notion d'inf: plus grand minorant.

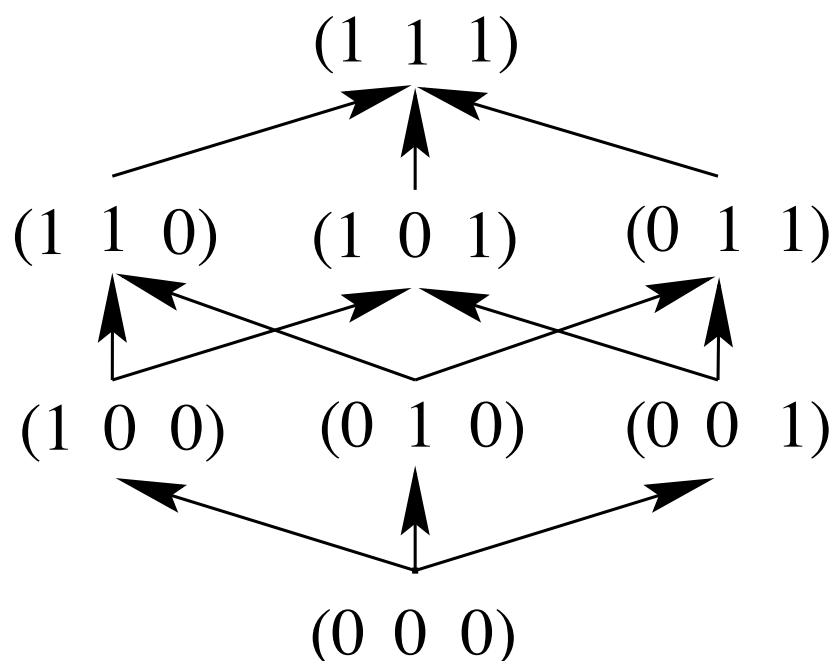
---

## Structure de treillis

---

- La structure principale est le *treillis complet*.
- Un treillis est un ensemble ordonné  $X$  tel que pour deux éléments quelconque  $X_1$  et  $X_2$  de  $X$  on puisse définir un plus petit élément ( $X_1 \wedge X_2$ ) et un plus grand élément ( $X_1 \vee X_2$ ).
- Un treillis *complet* est un ensemble ordonné  $X$  (muni d'une relation d'ordre) tel que n'importe quel sous ensemble  $X_i \subset X$  possède à la fois un plus petit et un plus grand élément.

## Exemple de treillis complet simple



Le treillis des images binaires  $3 \times 1$ .

## Autres exemple de treillis

- Treillis des parties  $\mathcal{P}(E)$  d'un ensemble  $E$ . Ordre défini par l'inclusion.  $\text{Sup}=\cap$ ,  $\text{Inf}=\cup$ . Extrêmes= $E$ ,  $\emptyset$ .
- Treillis des réels. Ordre=ordre numérique sur  $R$ .  $\text{Sup}$  et  $\text{Inf}$  au sens usuel, Extrêmes= $-\infty$ ,  $+\infty$ .
- Treillis des ensembles convexes.  $\text{Sup}$ =enveloppe convexe de l'union.  $\text{Inf}$ =intersection.

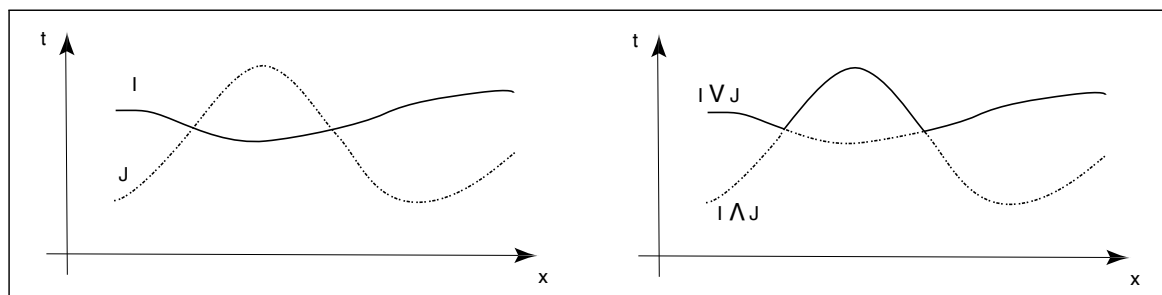
## Exemples de treillis complets intéressants

- Le treillis Booléen des ensembles: définit la MM des images binaires.
- Le treillis des images semi-continues en haut: définit la MM des images en niveaux de gris.
- Le treillis des fonctions multi-valuées: MM des images couleurs.

Note: la dimensionalité n'a pas d'importance (2-D, 3-D, etc), du moins en théorie.

Concepts de base – p. 23/24

## Treillis des fonctions



sup et inf de deux images 1-D

Concepts de base – p. 24/24

---

*Cours de Morphologie Mathématique*  
*Dilatations, érosions*

Hugues Talbot  
talboth@esiee.fr

ISBS / ESIEE 2008

---

Erosions, Dilatations – p. 1/23

---

## Rappel des ressources

- Le cours sera mis en ligne au fur et à mesure:  
<http://www.esiee.fr/~talboth/ISBS/Morpho/>
- Également copie des TDs + TPs (en avance).

---

Erosions, Dilatations – p. 2/23

---

## Rappel du cours précédent

---

- Représentation d'images, résolution spatiale et spectrale.
- Notion d'ordre, ordre partiel
- Notions de majorant, minorant, sup, et inf
- Treillis complets.

---

## Questions

---

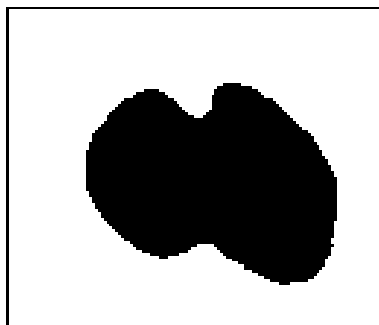
1. Questions sur le cours précédent ?
2. Remarques ?
3. Questions sur la suite du cours?

## Concepts de base, suite

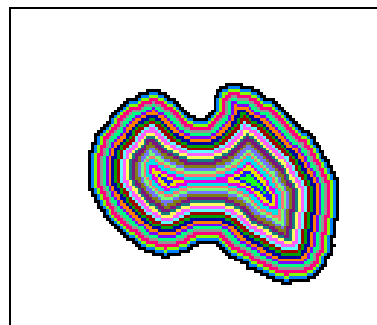
Erosions, Dilatations – p. 5/23

### Concept de base: la distance

- La fonction distance



Noyaux de cellules



Fonction distance

- Dépendente du graphe sous-jacent.

Erosions, Dilatations – p. 6/23



## Concept de base: la mesure

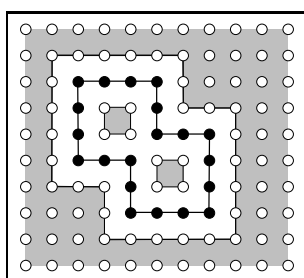
C'est ce qui nous intéresse *in fine*.

- Nombre d'objets.
- Longueur, aire, volume.
- Orientation, élongation.
- La MM est surtout basé sur la mesure, non la forme.

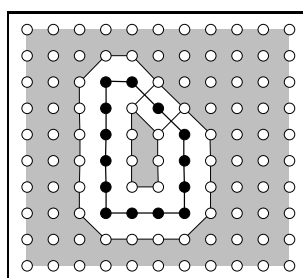
## Concept de base: la grille/le graphe

La grille digitale pose des problèmes:

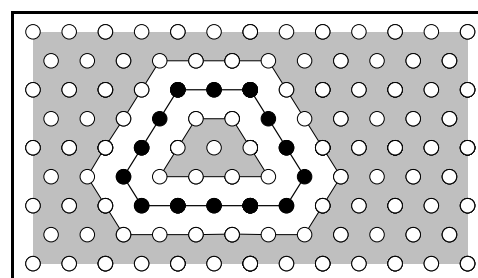
- Connectivité:



4-connecte



8-connecte

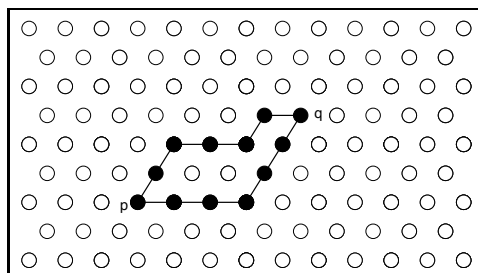


6-connecte

Pour résoudre le problème de la distance 4 ou 8-connecte, on doit considérer le fond 4-connecte et les objets 8-connecte ou vice-versa.

## Grille/graphes et distance

- distance



Il existe plus d'un plus court chemin

Il est possible d'invoquer la distance Euclidienne pour résoudre ce problème mais des problèmes d'implémentation et de rapidité surgissent alors.

## Dualité – complémentarité

- Sup et Inf dans un treillis complet jouent des rôles symétriques.
- Deux opérateurs  $\psi$  et  $\psi^*$  sont duaux si pour tout  $X$ :

$$\psi(X^c) = [\psi^*(X)]^c$$

- Pour le treillis des parties d'un ensemble: complémentation
- Pour le treillis des fonctions dans  $R$ : symétrie par rapport à 0.

---

## Autres points importants

---

- Le lien entre la théorie des ensembles et la théorie des treillis:
  - Décomposition par seuillage
  - Ombre
  - Fonctions bi-valuées
- Importance des propriétés mathématiques quand on définit les opérateurs.

---

## Liens avec d'autres théories

---

MM complémentaire :

- La géométrie discrète (distance, squelettes, etc).
- La théorie des graphes (plus court chemins, LPE, computational geometry, etc).
- les statistiques: modèles aléatoires, théorie de la mesure, stéréologie.
- Théories linéaires: replace  $+$  with  $\wedge$ .
- Scale-space: remplacer le filtrage Gaussien avec des filtres morphologiques.
- Isophotes (level sets): dilatation avec des EDPs, FMM est une fonction distance.
- Informatique, algorithmes.

---

## *Erosions et dilatations*

---

Erosions, Dilatations – p. 13/23

---

### Erosion

Dans le contexte le plus général, une érosion  $\varepsilon$  est un opérateur qui commute avec l'inf.

$$\bigwedge \varepsilon(\cdot) = \varepsilon \bigwedge (\cdot)$$

Ce qui n'est pas très intuitif. On commence par définir l'érosion par élément structurant (ES):

$$\varepsilon_B(X) = \{x | B_x \subseteq X\}$$

C-à-d: est-ce que l'ensemble contient l'ES?

Définition équivalente:

$$\varepsilon_B(X) = \bigcap_{b \in B} X_{-b}$$

---

Erosions, Dilatations – p. 14/23

---

## Erosion sur le treillis des fonctions:

---

Dans ce cas:

$$\varepsilon_B(f) = \bigwedge_{b \in B} f_{-b}$$

Ce qui est équivalent à:

$$[\varepsilon_B(f)](x) = \min_{b \in B} f(x + b)$$

Érosion par un ES non plat (rarement utilisé dans la pratique, sauf avec une parabole)

$$[\varepsilon_{B_v}(f)](x) = \min_{b \in B_v} \{f(x + b) - B_v(b)\}$$

---

## Dilatation

---

La dilatation est le dual de l'érosion: une dilatation  $\delta$  sur un treillis complet commute avec le sup:

$$\bigvee \delta(\cdot) = \delta \bigvee(\cdot)$$

Ce qui est toujours aussi peu intuitif. On commence par la dilatation par un élément structurant (ES):

$$\delta_B(X) = \{x | \check{B}_x \cap X \neq \emptyset\}$$

c-à-d: est-ce que l'ES intersecte l'ensemble.

Q: Quel est le dilaté d'un point ?

Définition équivalente:

$$\delta_B(X) = \bigcup_{b \in B} X_b$$

## Le cas des fonctions:

Dans le cas du treillis des fonctions:

$$\delta_B(f) = \bigvee_{b \in B} f_b$$

Ce qui est équivalent à:

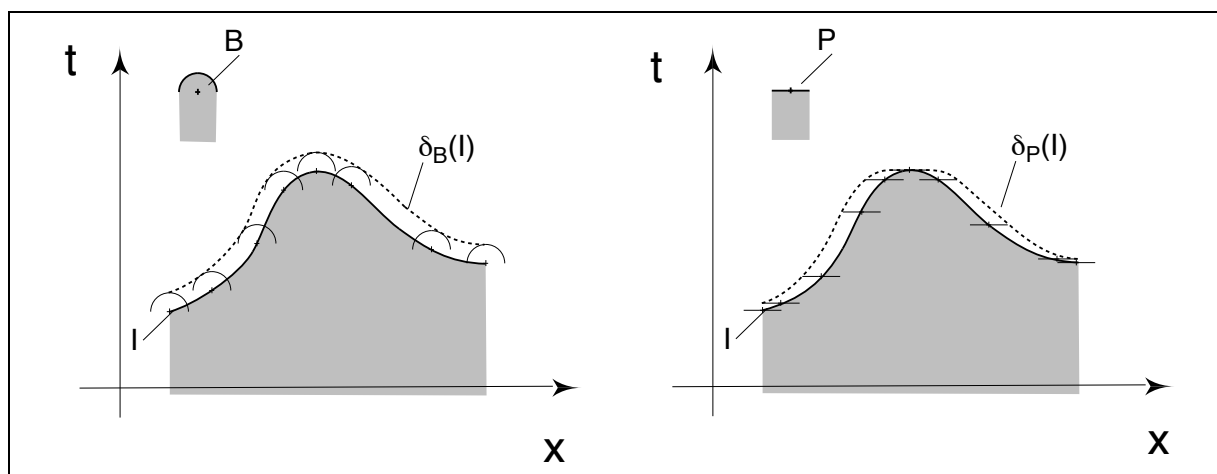
$$[\delta_B(f)](x) = \max_{b \in \tilde{B}} f(x + b) = \max_{b \in B} f(x - b)$$

avec + et - les opérateurs de translation dans ce contexte.

La dilatation par un ES non-plat:

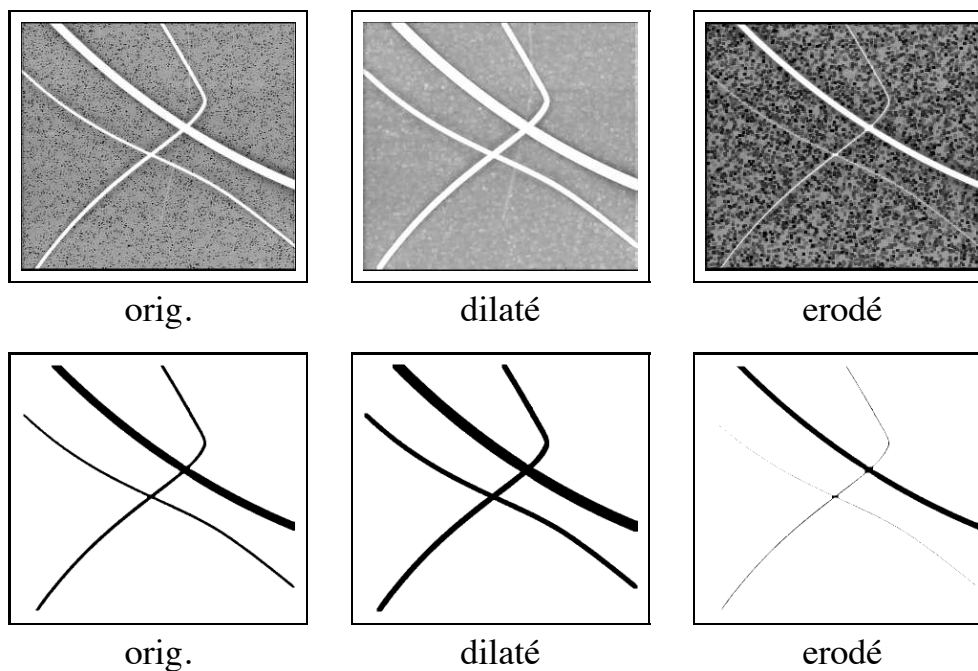
$$[\delta_{B_v}(f)](x) = \max_{b \in B_v} \{f(x - b) + B_v(b)\}$$

## Illustrations - érosion/dilatation



Dilatation d'une image 1-D.

## Illustration (cont.)



Erosions, Dilatations – p. 19/23

## Propriétés

- symétrie:  $\delta_B(X) = \delta_X(B)$ . Même chose pour l'érosion.
- dualité:  $\delta_B(f) = -\varepsilon_B(-f)$
- croissance:  $f \leq g \Rightarrow \delta_B(f) \leq \delta_B(g), \varepsilon_B(f) \leq \varepsilon_B(g)$
- distributivité:  $\delta(\bigvee_i f_i) = \bigvee_i \delta(f_i) ; \varepsilon(\bigwedge_i f_i) = \bigwedge_i \varepsilon(f_i)$
- composition:  $\delta_{B_2} \delta_{B_1} = \delta_{(\delta_{B_2} B_1)} ; \varepsilon_{B_2} \varepsilon_{B_1} = \varepsilon_{(\varepsilon_{B_2} B_1)}$
- relation d'ordre:  $\varepsilon_B \leq I \leq \delta_B$  ssi  $B$  contient son origine.
- invariance par translation.

Erosions, Dilatations – p. 20/23

---

## Algorithme: dilatation en temps constant

---

- Dilatation en temps constant par un segment (fenêtre glissante ou récursive (van Herk))
- composition de dilatations horiz. et vert.: dilatation par un carré.
- dilatation par des polygônes: approximations de disques.

---

## Van Herk algorithm

---

- Image d'entrée 1D de longueur  $nx$  est divisée en block de taille  $\lambda$ , où  $\lambda$  est la longueur de l'élément structurant en pixels.
- Les pixels sont étiquetés de 0 à  $nx - 1$ , en supposant que  $nx$  est un multiple de  $\lambda$  (sinon on rajoute des pixels jusqu'à ce que ce soit le cas).
- On a besoin de deux buffers (tampons)  $g$  et  $h$ .
- Dans le cas de la dilatation, on prend le maximum *récursivement* à l'intérieur des blocks, à droite pour  $g$ , à gauche pour  $h$ .
- Lorsque  $g$  et  $h$  sont contruits, le résultat pour la dilatation en chaque point  $x$  est donné en considérant le maximum des valeurs pour  $g$  à la position  $x + \lambda - o - 1$  et  $h$  à la position  $x - o$ , avec  $o$  l'origine de l'ES.

La dilatation se calcule en temps constant quelle que soit la longueur de l'ES.



---

## Choix de l'élément structurant

---

- Contrairement à ce qu'on pourrait penser la *forme* de l'ES n'est pas tellement critique (il y a des exceptions).
- En particulier l'érosion ou la dilatation ne sont pas très utiles pour définir les formes.
- ES courants: carrés, diamant, polygones, disques Euclidiens, lignes, paires de points.
- Application des ES arbitraires sont rares.
- Changer l'origine de l'ES translate le résultat.

---

*Cours de Morphologie Mathématique*  
*Géodésie*  
*Ouvertures / fermetures*

Hugues Talbot

talboth@esiee.fr

ISBS / ESIEE 2008

---

Ouverture & fermetures – p. 1/22

---

## Rappel du cours précédent

---

- Erosions, dilatations
- Définitions algébrique (commute avec inf ou sup resp.)
- Définitions géométrique (translations d'image ou d'ES)
- s algorithmiques (voisinage)

---

Ouverture & fermetures – p. 2/22

---

## Notation de Minkowski

---

Addition et soustraction ensembliste:

- Addition de Minkowski:

$$A \oplus B = \{y = a + b, a \in A, b \in B\}$$

d'où :

$$\delta_B(A) = A \oplus B$$

- Soustraction de Minkowski:

$$A \ominus B = \{x, x + b \in A, b \in B\}$$

d'où :

$$\epsilon_B(A) = A \ominus B$$

---

## Choix de l'élément structurant

---

- Pour des raisons pratiques le choix de l'ES est toujours limité:
  - ES de la trame (carré, losange (ou diamant), hexagone)
  - Lignes, polygones
  - Bipoints
  - Disques Euclidiens ou leurs applications
  - Cas rares d'ES arbitraires

---

## *Ouvertures et fermetures*

---

Ouverture & fermetures – p. 5/22

---

### Compositions d'érosion/dilatation

- On peut se demander ce qui se passe lorsqu'on compose une dilatation et une érosion ou vice-versa.
- L'un n'est pas l'inverse de l'autre
- Attention aux problèmes de symétrie de l'ES.

---

Ouverture & fermetures – p. 6/22

---

## Ouverture

---

Il n'existe pas d'inverse de l'érosion. Lorsqu'on dilate un objet précédemment érodé avec le même élément structurant, on obtient la plus grande opération morphologique qui puisse récupérer une partie de l'information. Cette opération est appelée l'ouverture.

$$\gamma_B(f) = \delta_B[\epsilon_B(f)]$$

Note: le résultat est indépendant de la position de l'origine.  
En notation ensembliste, on a:

$$\gamma_B(X) = X \circ B = (X \ominus B) \oplus B$$

---

## Définition par les ensembles

---

En terme d'opération sur les ensembles binaires on a:

$$\gamma_B(X) = \bigcup \{B \mid B \subseteq X\}$$

Définition qui ressemble à celle de l'érosion, mais cette fois on garde tout l'ES, et non seulement son origine.

Une ouverture fait disparaître les petites extrusions mais laisse les intrusions inactes.

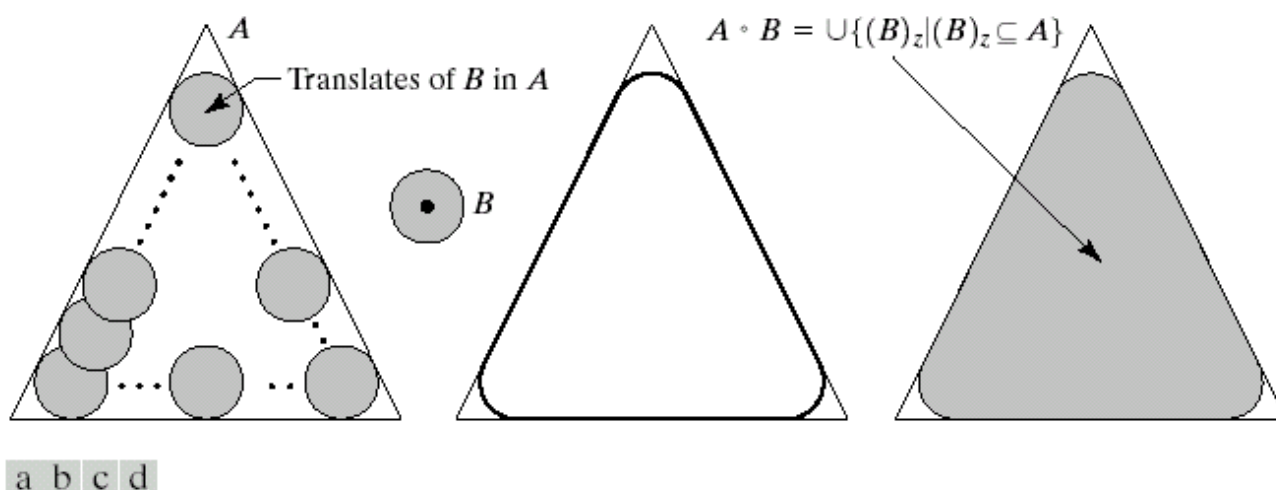
## Définition sur les fonctions

En terme d'opérateur sur les fonctions, on définit l'ouverture par

$$\gamma_B(X) = \bigvee \{B \mid B \leq X\}$$

Sur les fonctions, une ouverture laisse les vallées intacte mais enlève les pics.

## Illustration pour l'ouverture



a b c d

**FIGURE 9.8** (a) Structuring element  $B$  “rolling” along the inner boundary of  $A$  (the dot indicates the origin of  $B$ ). (c) The heavy line is the outer boundary of the opening. (d) Complete opening (shaded).

---

## Fermeture

---

Il n'existe pas d'inverse de la dilatation. Sans information supplémentaire le mieux qu'on puisse faire est d'éroder l'image avec le même ES. Cette approche permet de définir l'opérateur de *fermeture*.

$$\phi_B(f) = \epsilon_B[\delta_B(f)]$$

Le résultat ne dépend pas de la localisation de l'origine.  
En notation ensembliste, on a :

$$\phi_B(X) = X \bullet B = (X \oplus B) \ominus B$$

---

## Définition sur les ensembles

---

En utilisant la notation ensembliste:

$$\phi_B(X) = \bigcap \{B^c \mid X \subseteq B^c\}$$

La fermeture est le complément de l'ouverture.  
Une fermeture garde les extrusions mais enlève les intrusions.

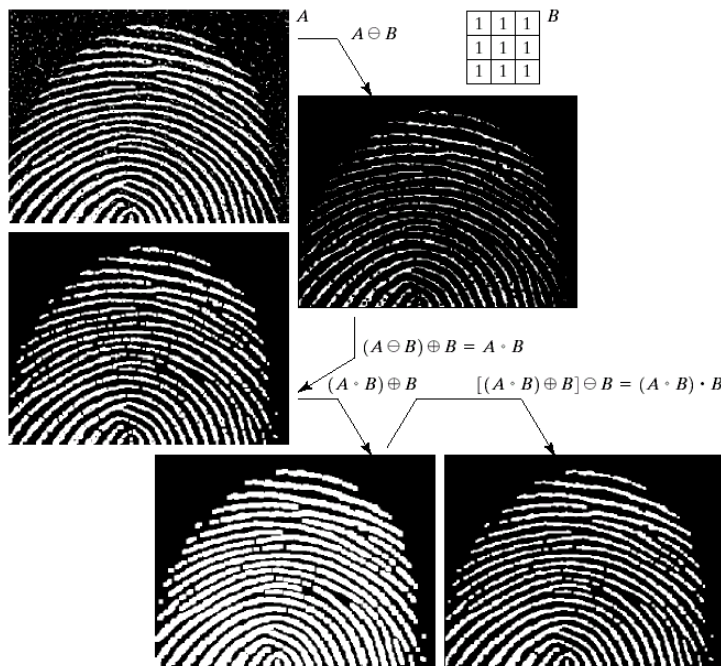
# Définition sur les fonctions

En utilisant la notation fonctionnelle:

$$\phi_B(X) = \bigwedge \{-B | X \geq -B\}$$

Une fermeture remplit les vallées mais laisse les pics intacts.

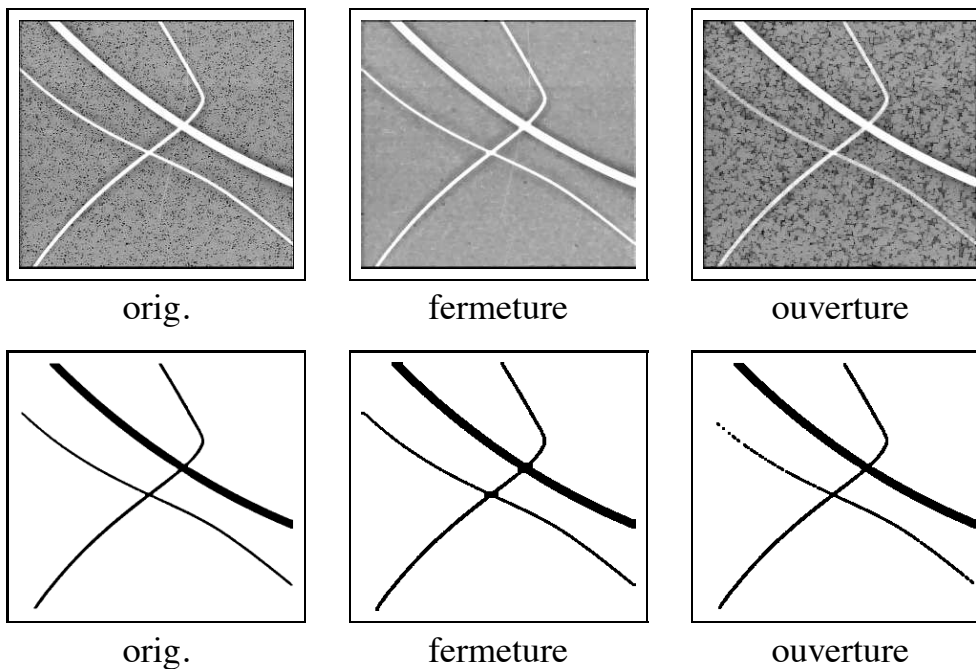
# Suppression du bruit



**FIGURE 9.11**  
 (a) Noisy image.  
 (c) Eroded image.  
 (d) Opening of A.  
 (d) Dilation of the opening.  
 (e) Closing of the opening. (Original image for this example courtesy of the National Institute of Standards and Technology.)



## Ouverture/fermeture: Illustration



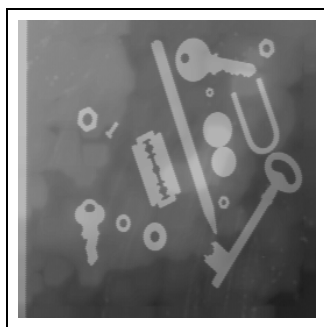
Ouverture & fermetures – p. 15/22

## Propriétés

- dualité:  $\phi_B(f) = -\gamma_B(-f)$
- préservation de l'ordre (extensivité/anti-extensivité):  $\gamma \leq I \leq \phi$
- croissance:  $f \leq g \Rightarrow \gamma(f) \leq \gamma(g); \phi(f) \leq \phi(g)$
- idempotence:  $\gamma\gamma = \gamma; \phi\phi = \phi$

Ouverture & fermetures – p. 16/22

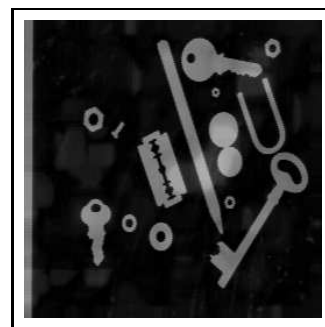
## Application : correction du fond



original.



ouverture avec un grand carré.

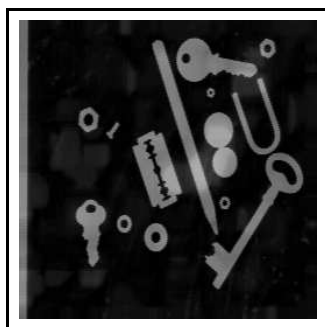


différence  $\times 2$ .

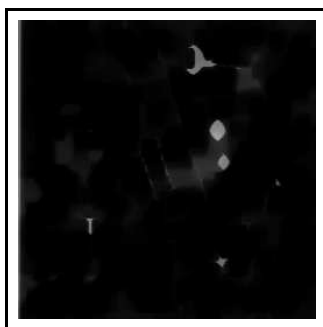
## Ouvertures & fermetures algébriques

- Même propriétés que l'ouverture/fermeture, mais ne sont plus basées sur des ES.
- Sup d'ouvertures est une ouverture.
- Inf de fermetures est une fermeture.
- Exemples (ouvertures):
  - Ouvertures par attributs (par exemple la surface)
  - Sup d'ouvertures par des lignes
  - Ouvertures de rang max
  - Enveloppe convexe
  - Ouverture par des chemins
  - Ouverture par reconstruction

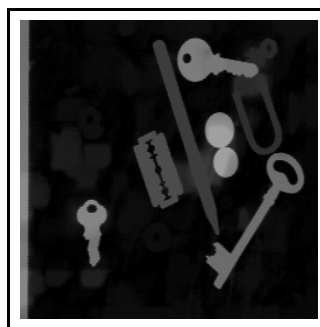
## Exemple d'opérateurs algébriques



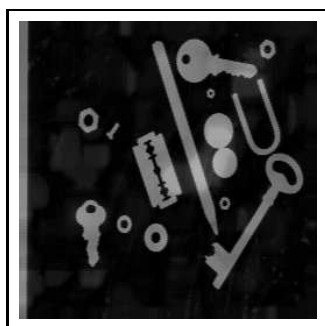
Outils.



Érosion.



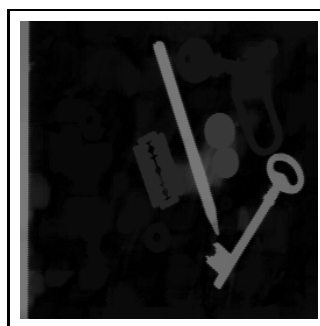
Reconstruction.



Outils.



Union de lignes.

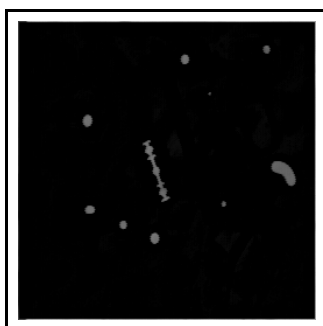


Reconstruction.

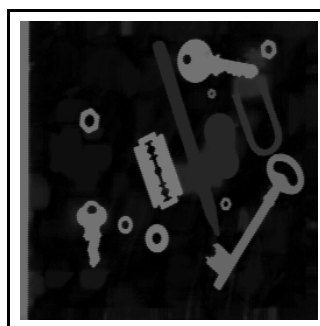
## Opérateurs algébriques (suite)



Fermeture surfacique.

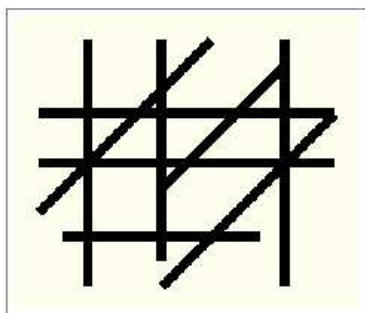


Detection des trous.

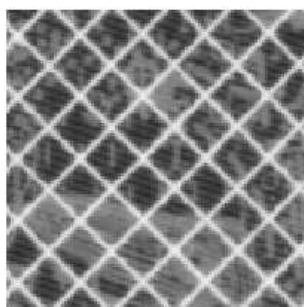


Reconstruction.

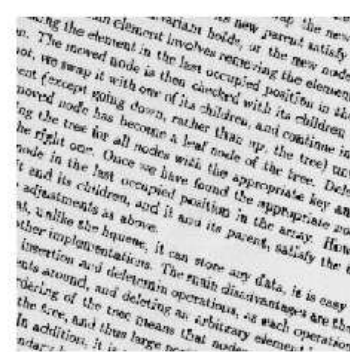
## Rose des directions



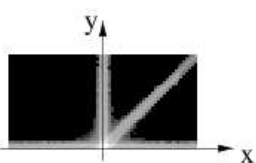
Network of lines



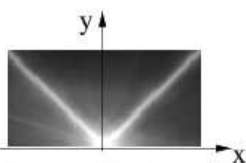
Texture on a soil map



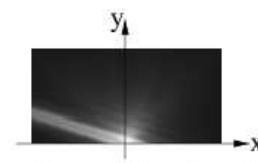
Slanted text



Rose of directions by openings

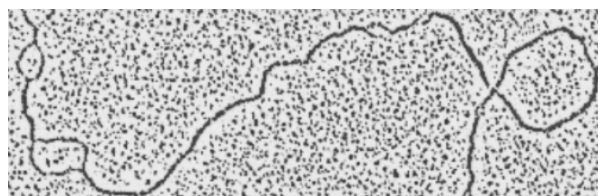


Rose of directions by openings

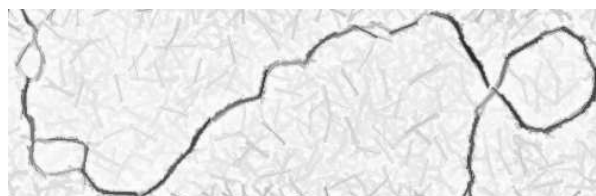


Rose of directions by openings

## Autres fermetures algébriques



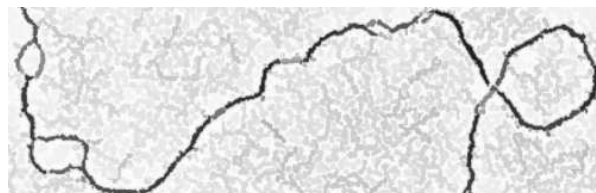
Micrographe d'ADN au microscope



Fermeture par  $\cap$  de lignes



Fermeture par surface



Fermeture par chemins

---

# Cours de Morphologie Mathématique

## Géodésie – Résidus

Hugues Talbot

talboth@esiee.fr

ISBS / ESIEE

2009

---

Géodésie – résidus – p. 1/25

---

### Rappel du cours précédent

---

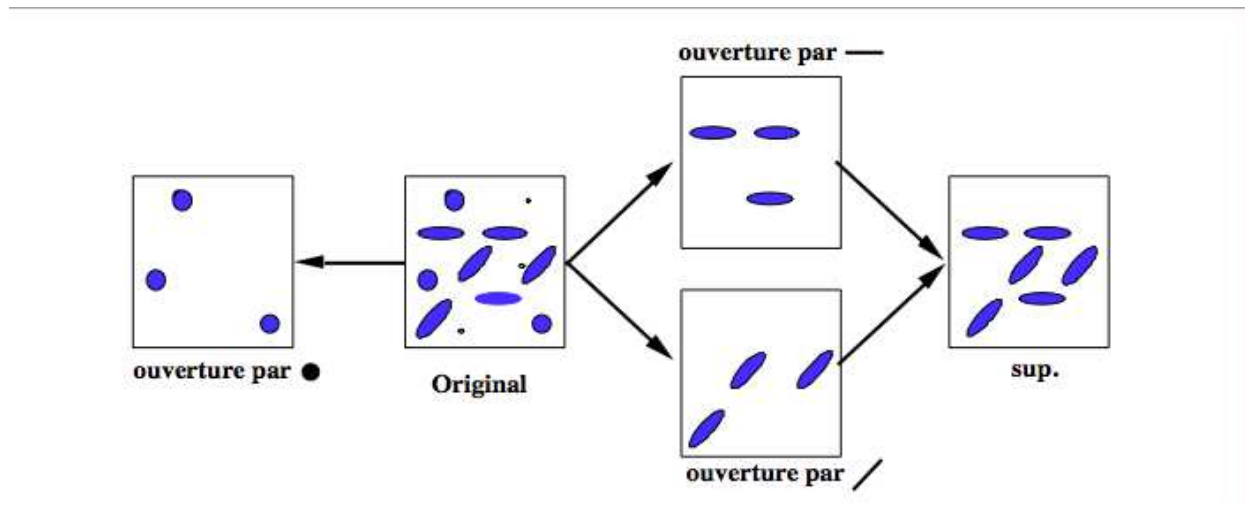
- Ouverture morphologique = composition d'une érosion suivie d'une dilatation *adjointe*.
- Fermeture morphologique = composition d'une dilatation suivie d'une érosion *adjointe*.
- adjointe = ici, qui est complémentaire.
- Ouverture et fermeture algébriques = qui respectent les propriétés des ouvertures et fermetures:
  1. Extensivité (pour la fermeture), anti-extensivité (pour l'ouverture)  
;
  2. Croissance ;
  3. Idempotence.
- Sup d'ouverture est une ouverture. Inf de fermeture est une fermeture.

---

Géodésie – résidus – p. 2/25

## Exemple

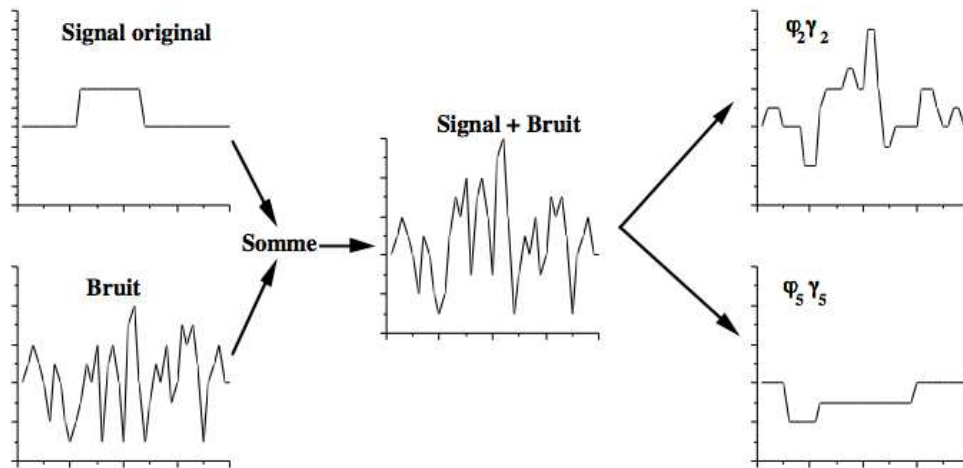
Illustration du filtrage morphologique simple:



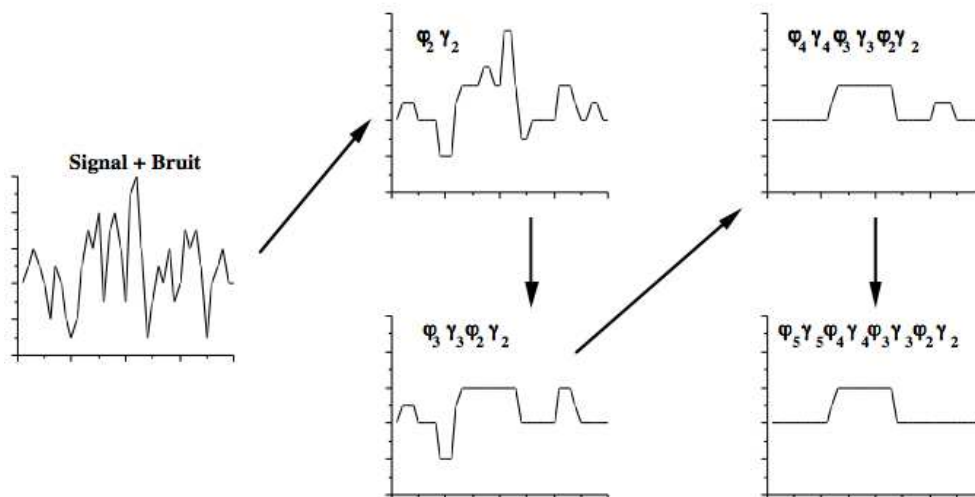
## Filtres alternés séquentiels

- Combinaisons ouvertures/fermetures de taille croissantes - non auto-duales
- Permettent d'obtenir de meilleurs résultats qu'une combinaison simple

# Filtre simple



# Filtre alterné



---

## Granulométries

---

Les ouvertures et fermetures permettent de définir la notion de granulométrie.

- Intuitivement la granulométrie est l'étude de la *taille* des objets. Physiquement elle correspond au tamisage pour les grains et aux études de porosité par fluide pour les trous.
- Pour le tamisage, une suite de tamis à trous de plus en plus gros permettent de classer une population de grains. Un tamis de taille  $\lambda_1$  arrête certains grains qui sont passés par un tamis de taille  $\lambda_2 > \lambda_1$ .
- La forme des grains est importante : renvoi à la notion d'élément structurant.

---

## Granulométries en MM

---

On définit comme granulométrie toute famille d'ouvertures  $\gamma_\lambda$ , avec  $\lambda$  un paramètre positif tel que  $\lambda \geq \mu \Rightarrow \gamma_\lambda \leq \gamma_\mu$ . Cette seconde condition est aussi appelée *loi d'absorption* parce que  $\gamma_\lambda \gamma_\mu = \gamma_\mu \gamma_\lambda = \gamma_{\vee\{\lambda, \mu\}}$ .

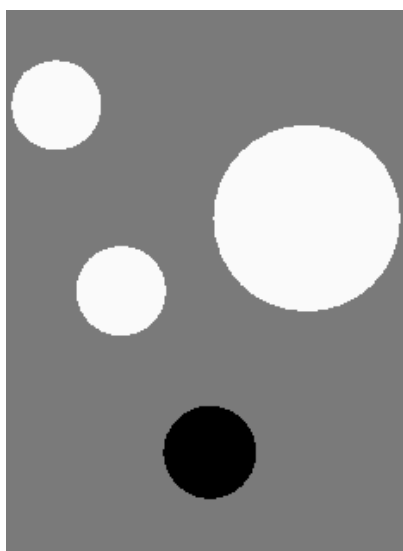
On construit la *courbe granulométrique* de la façon suivante, avec un paramètre *lambda* discret:

$$(1) \quad G_f(\lambda) = \sum f - \sum \gamma_\lambda(f) \quad \lambda \in [0, 1, 2, \dots, R]$$

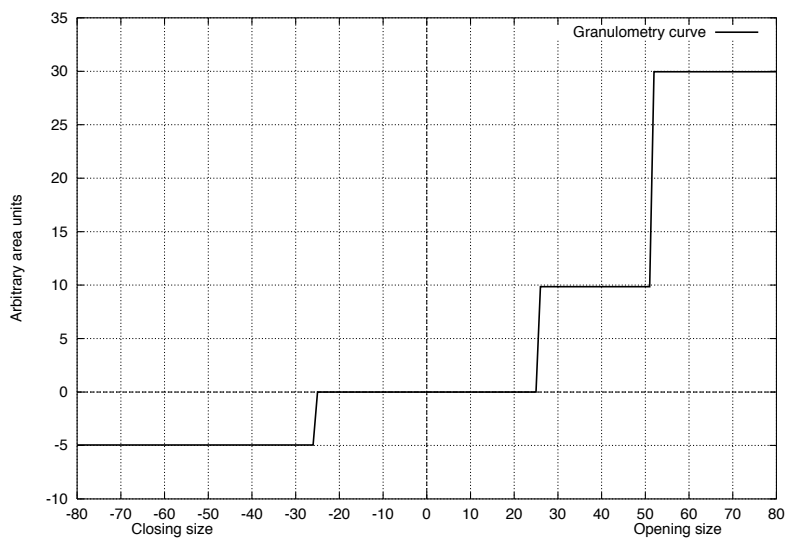
Où  $f$  est l'image,  $\sum f$  est la somme des pixels de l'image (l'intégrale) et  $R$  le point où plus rien ne change. Pour garantir une courbe monotone on doit aussi utiliser un ES convexe. On peut obtenir une courbe duale avec des fermetures.



## Exemple simple



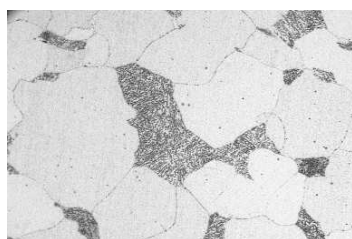
Image



Courbe

## Exemple d'application

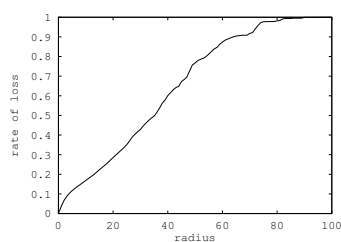
Étude du vieillissement de conduites de vapeur pour génération d'électricité.



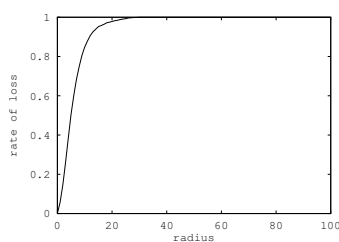
échantillon jeune



échantillon vieux



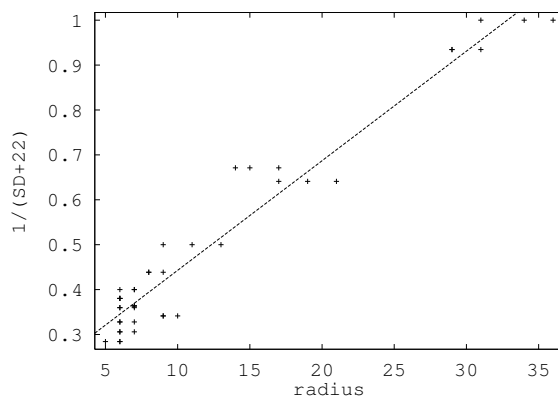
courbe jeune



courbe vieux

## Indicateur de vieillissement

On prend comme mesure de vieillissement le paramètre  $\lambda$  pour lequel 50% du niveau d'idempotence à été atteint.



Courbe de régression granulométrie – vieillissement, avec  $SD = \frac{1}{a+bx} - 22$ ,  
 $a = 0.198769, b = 0.024406$ .

*Notions sur la géodésie*

## Dilatation géodésique

Dilatation unitaire à l'intérieur d'un masque  $g$ :

$$\delta_g^{(1)} = \delta^{(1)} \wedge g$$

De façon récursive:

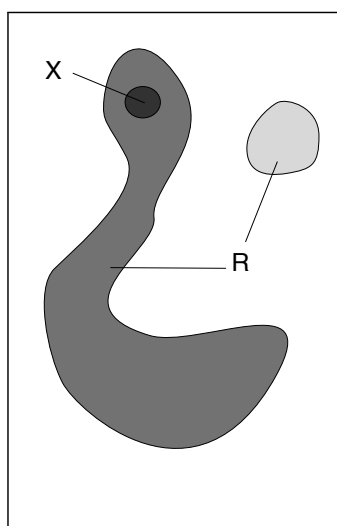
$$\delta_g^{(n)} = \delta_g^{(1)}[\delta_g^{(n-1)}]$$

Note: notion de distance géodésique, toujours plus grande que la distance non-géodésique, et peut-être infinie.

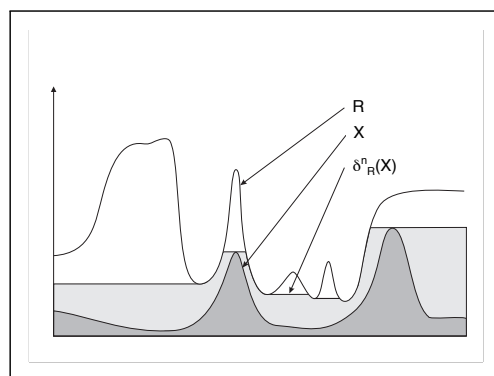
## Reconstruction

La reconstruction est une opération morphologique itérée jusqu'à idempotence:

$$R_g(f) = \delta_g^{(i)}(f)$$

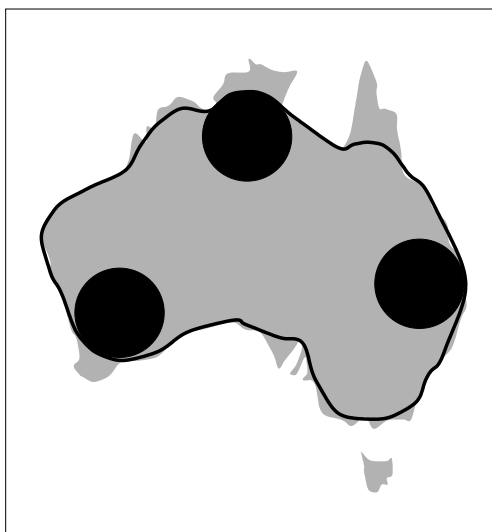


(a)



(b)

## Exemple de reconstruction



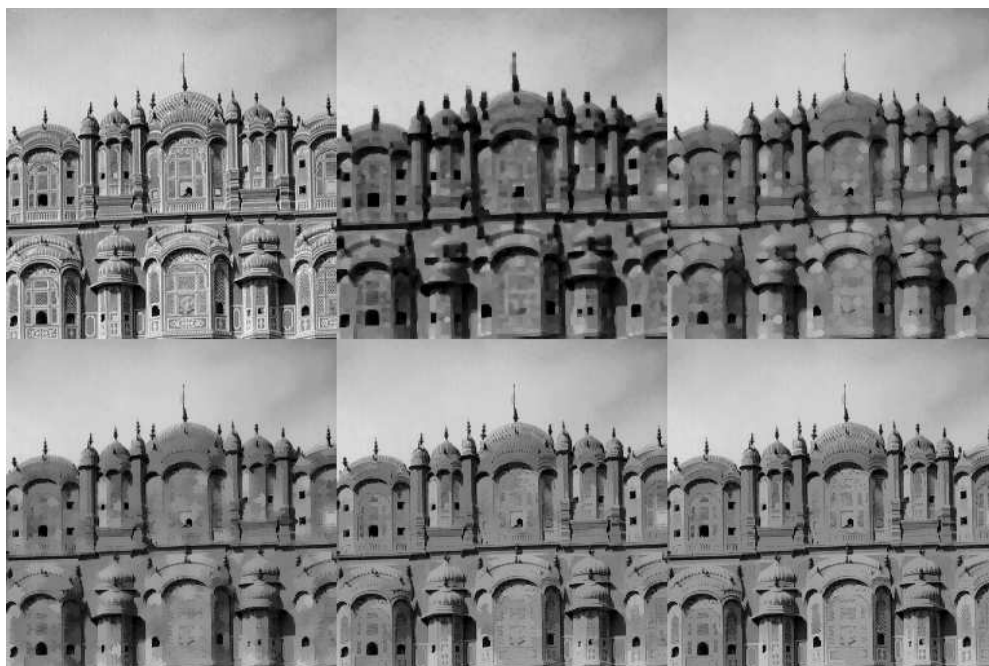
Ouverture par disques



Reconstruction

Notez qu'on a pas récupéré la Tasmanie.

## Reconstruction pour les fonctions



---

## TD

---

- Définition de l'érosion géodesique ?
- Quel est l'intérêt tout particulier de la géodésie dans le cas de l'analyse des images réelles, à savoir non d'extension infinies ?
- Comment utiliser les notions géodesique pour, en morphologie sur les ensembles:
  1. Enlever les particules touchant le bord d'une image ?
  2. Boucher les trous des objets ?

---

## Algorithmes pour la reconstruction

---

- Algorithme trivial trop lent (beaucoup de passes sur l'image)
- Passes récursives dans le sens vidéo et anti-vidéo résoud l'essentiel du problème (mais pas dans les régions en spirales).
- On termine avec un algorithme à base de queues.

Note: la forme de l'ES n'a quasiment pas d'importance (pourquoi?).

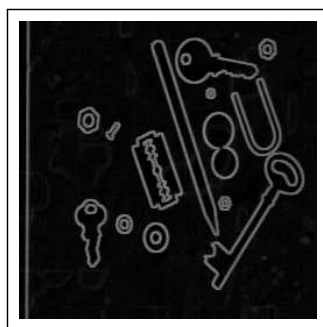
## *Résidus*

### Notions de résidus

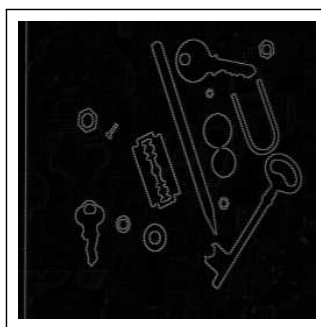
Lorsque toutes les propriétés vues auparavant deviennent utiles.

- Du fait que  $\gamma \leq I \leq \phi$ ,  $CB = I - \gamma$  and  $CN = \phi - I$  ont de bonnes propriétés et se comportent bien. On appelle ces transformations “Chapeaux haut-de-forme”.
- De façon similaire  $D_e = \delta - \epsilon$ ,  $D_i = I - \epsilon$  and  $D_o = \delta - I$  sont toutes des gradients.
- Au coeur de la méthode morphologique : On enlève ce qu’on ne veut pas, on garde ce qu’on veut. On décrit ce qu’on souhaite garder ou enlever par des critères géométriques.

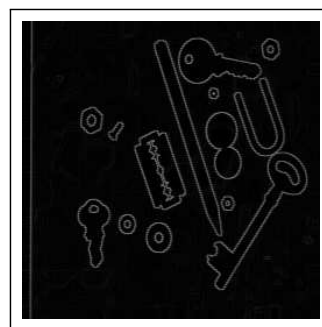
## Exemples de gradients



$\delta - \epsilon$



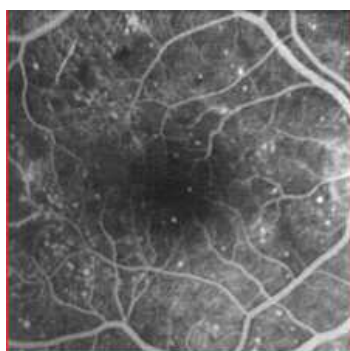
$I - \epsilon$



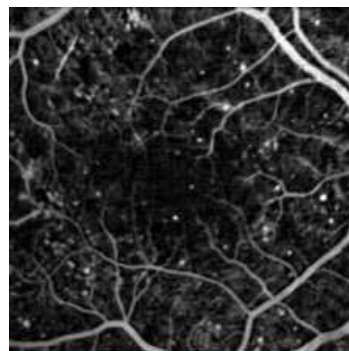
$\delta - I$

Le gradient morphologique ne se compose que de sa magnitude. On perd la direction.

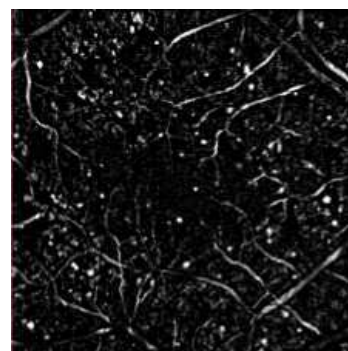
## Exemples de chapeaux



Original



Chapeau ES carré



Chapeau par lignes

Dans cet exemple, le premier chapeau est obtenu en utilisant une ouverture morphologique par carré de taille 3. Le second utilise un sup d'ouvertures par lignes de longueur 15.

---

## Autres résidus

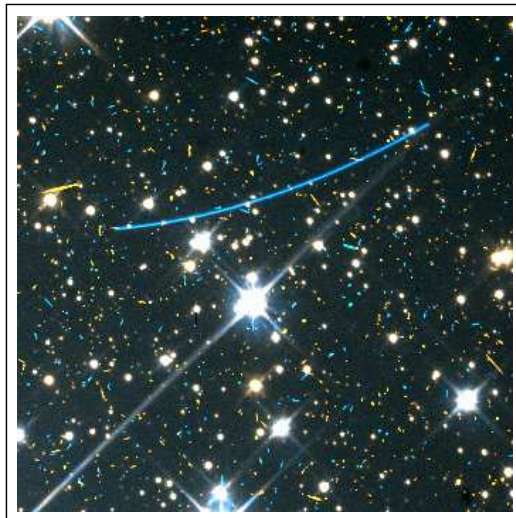
---

- Laplacien morphologique (dérivée seconde) :  $D_o - D_i$
- squelettes, fonction d'extinction, fonction bissectrice.
- Érodés ultimes
- Transformées en tout-ou-rien

---

## Une application : détection de trace d'astéroïdes

---

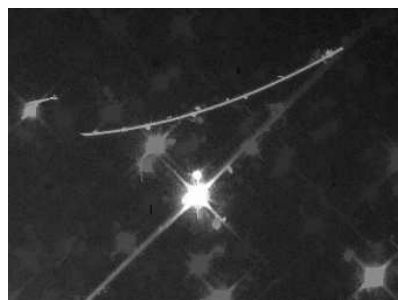




## Traces d'astéroïdes, suite.



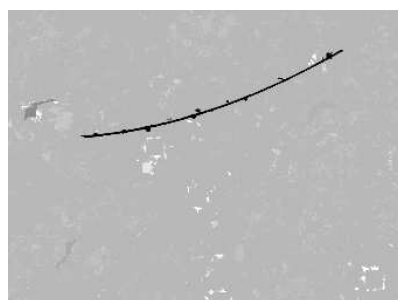
(traces)



(ouv. chemins)



(ouv. droites)



(Différence)

---

# *Cours de Morphologie Mathématique*

## *Résidus*

Hugues Talbot

talboth@esiee.fr

ISBS / ESIEE 2006

---

Résidus – p. 1/21

---

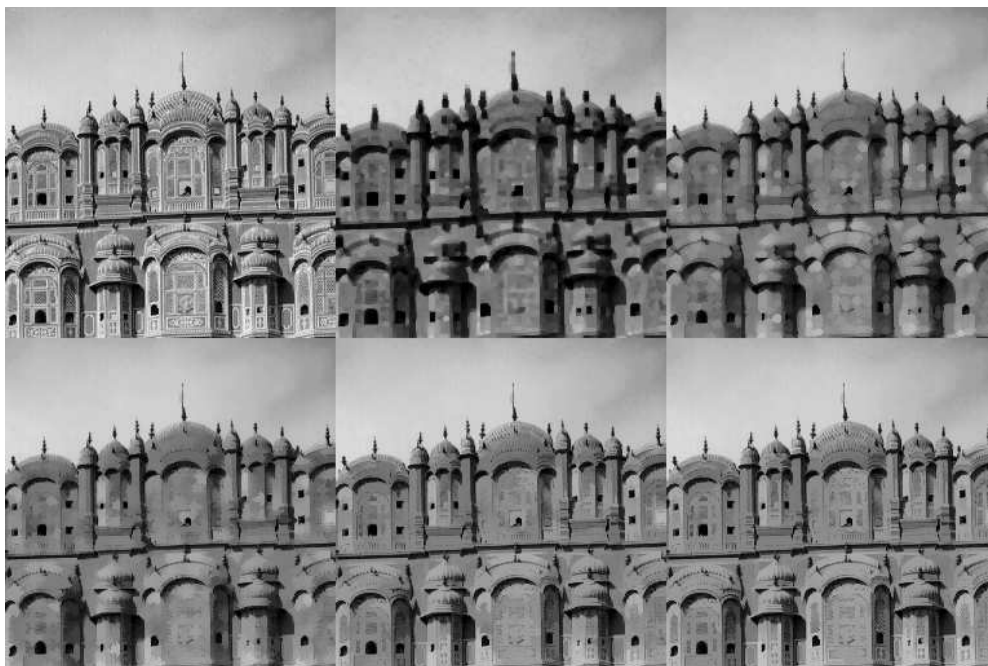
### Rappel du cours précédent

- Notions de géodésie ;
- Reconstruction = dilatation ou érosion géodésique itérée jusqu'à idempotence.
- Reconstruction en niveaux de gris.

---

Résidus – p. 2/21

## Reconstruction pour les fonctions



Résidus – p. 3/21

## Algorithmes pour la reconstruction

- Algorithme trivial trop lent (beaucoup de passes sur l'image)
- Passes récursives dans le sens vidéo et anti-vidéo résoud l'essentiel du problème (mais pas dans les régions en spirales).
- On termine avec un algorithme à base de queues.

Note: la forme de l'ES n'a quasiment pas d'importance (pourquoi?).

Résidus – p. 4/21

---

## Résidus

---

Résidus – p. 5/21

---

### Notions de résidus

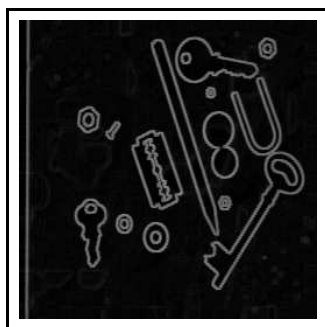
Lorsque certaines des propriétés vues auparavant deviennent utiles.

- Du fait que  $\gamma \leq I \leq \phi$ ,  $CB = I - \gamma$  and  $CN = \phi - I$  ont de bonnes propriétés et se comportent bien. On appelle ces transformations “Chapeaux haut-de-forme”.
- De façon similaire  $D_e = \delta - \epsilon$ ,  $D_i = I - \epsilon$  and  $D_o = \delta - I$  sont toutes des gradients.
- Au coeur de la méthode morphologique : On enlève ce qu’on ne veut pas, on garde ce qu’on veut. On décrit ce qu’on souhaite garder ou enlever par des critères géométriques.

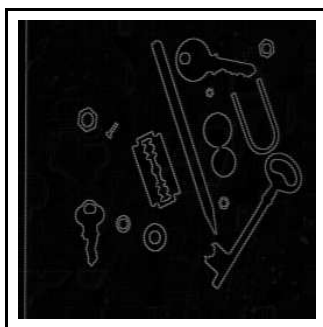
---

Résidus – p. 6/21

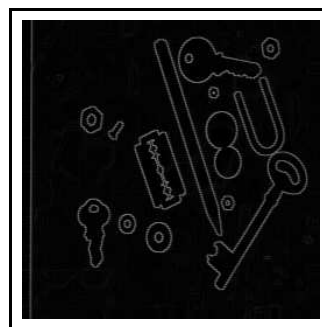
## Exemples de gradients



$\delta - \epsilon$



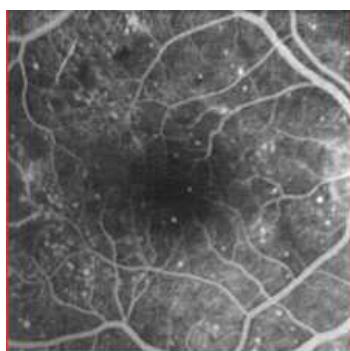
$I - \epsilon$



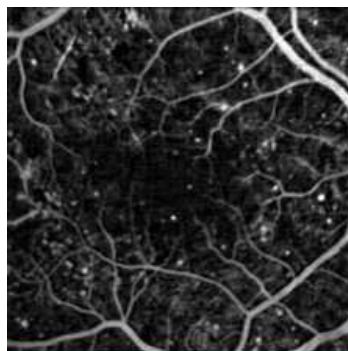
$\delta - I$

Le gradient morphologique ne se compose que de sa magnitude. On perd l'information sur la direction.

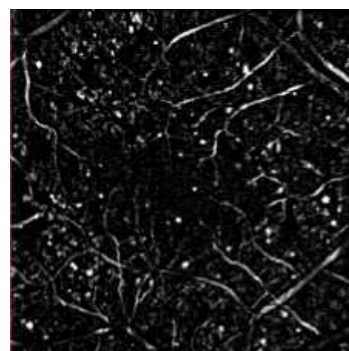
## Exemples de chapeaux



Original



Chapeau ES carré



Chapeau par lignes

Dans cet exemple, le premier chapeau est obtenu en utilisant une ouverture morphologique par carré de taille 3. Le second utilise un sup d'ouvertures par lignes de longueur 15.

---

## Autres résidus

---

- Laplacien morphologique (dérivée seconde) :  $D_o - D_i$
- squelettes, fonction d'extinction, fonction bissectrice.
- Érodés ultimes
- Transformées en tout-ou-rien

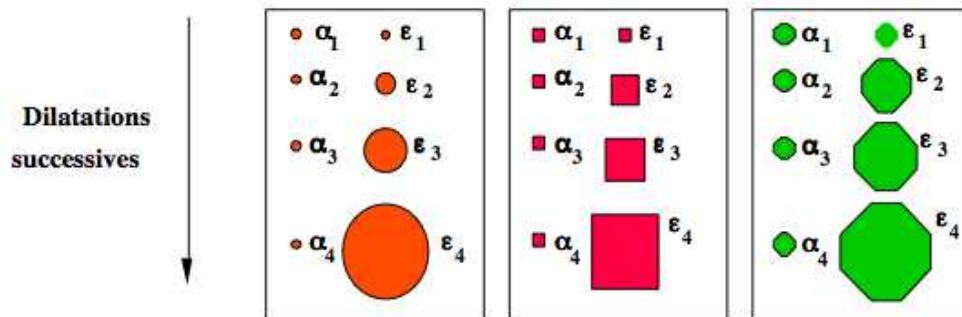
---

## Érodés ultimes

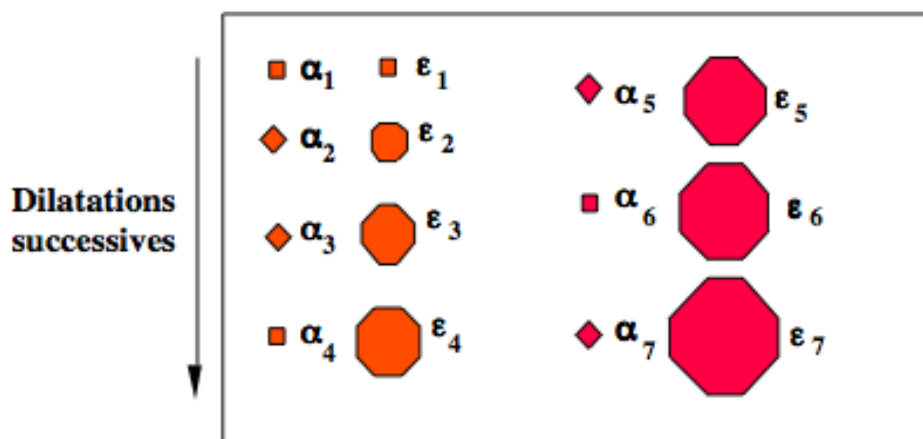
---

1. On définit une famille d'érosions  $\{\alpha_i\}$  dites *élémentaires* par des éléments structurants convexes indexés par un entier  $i$ , et l'ouverture  $\gamma_i$  associée.
2. On construit la famille d'érosion  $\epsilon_i = \alpha_i \dots \alpha_2 \alpha_1$ .
3. Les éléments structurants correspondants à ces érosions définissent une famille *granulométrique*  $\{\delta_i\}$ , car chaque  $\delta_i$  est ouvert par tous les  $\delta_j$  pour  $j \leq i$ .
4. Le plus souvent on utilise une famille homogène, c-à-d lorsque toutes les érosions élémentaires sont identiques:  $\forall i \alpha_i = \alpha_1 = \alpha$ , ce qui implique  $\epsilon_i = (\alpha)^i$ .

## Famille homogène



## Famille inhomogène



Les familles hétérogènes peuvent servir à obtenir un contrôle un peu plus fin sur la granulométrie.

## Érodé ultime

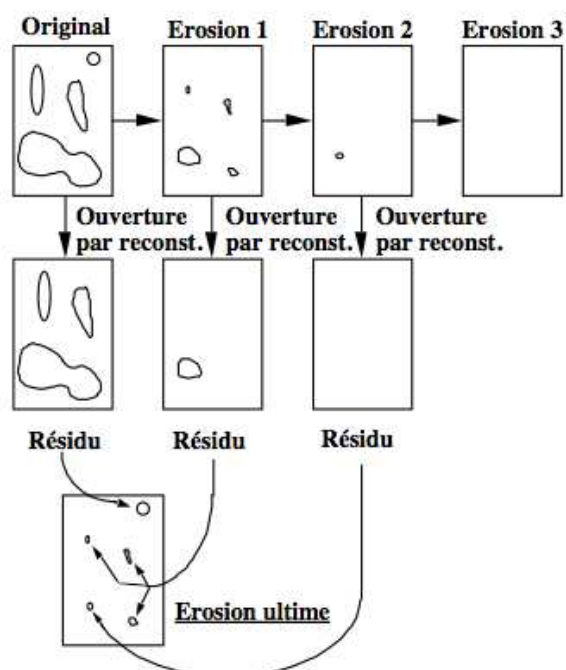
Étant donné ce qui précède, on définit l'érodé ultime  $U$  par la formule suivante:

$$U(X) = \bigcup_i \{U_i(X)\} = \bigcup_i \{\epsilon_i(X) \setminus \gamma^{rec}[\epsilon_i(X), \epsilon_{i+1}(X)], i = 0 \dots N\}$$

l'érodé ultime est

- Croissante
- Anti-extensive
- Idempotente (donc ?)
- Tous les  $U_i$  sont disjoints
- Si on dilate un  $U_i$  par l'ES  $i$ , le résultat est une *boule maximale*.

## Illustration





## Extrémités d'une particule

1. On suppose une particule  $X$  simplement connexe (sans trou)
2. On lui associe un *centroïde* intérieur.
3. Les extrémités de la particule sont alors les érodés ultimes *géodésiques* dans  $X$ , de l'ensemble  $Y$  privé de son centroïde.

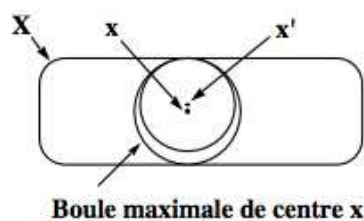
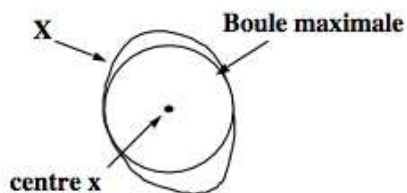


Résidus – p. 15/21

## Boules maximales

- Une boule  $\delta_n(x)$  de centre  $x$  et de taille  $n$  est maximale vis-à-vis de l'ensemble  $X$  s'il n'existe aucun autre indice  $k$  et aucun autre centre  $y$  tel que:

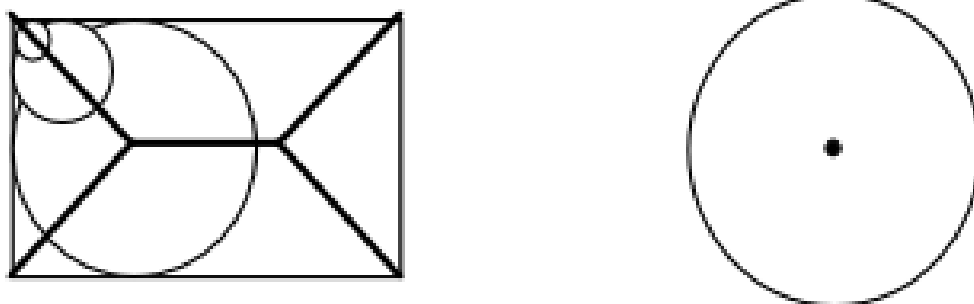
$$\delta_n(x) \in \delta_k(y) \in X$$



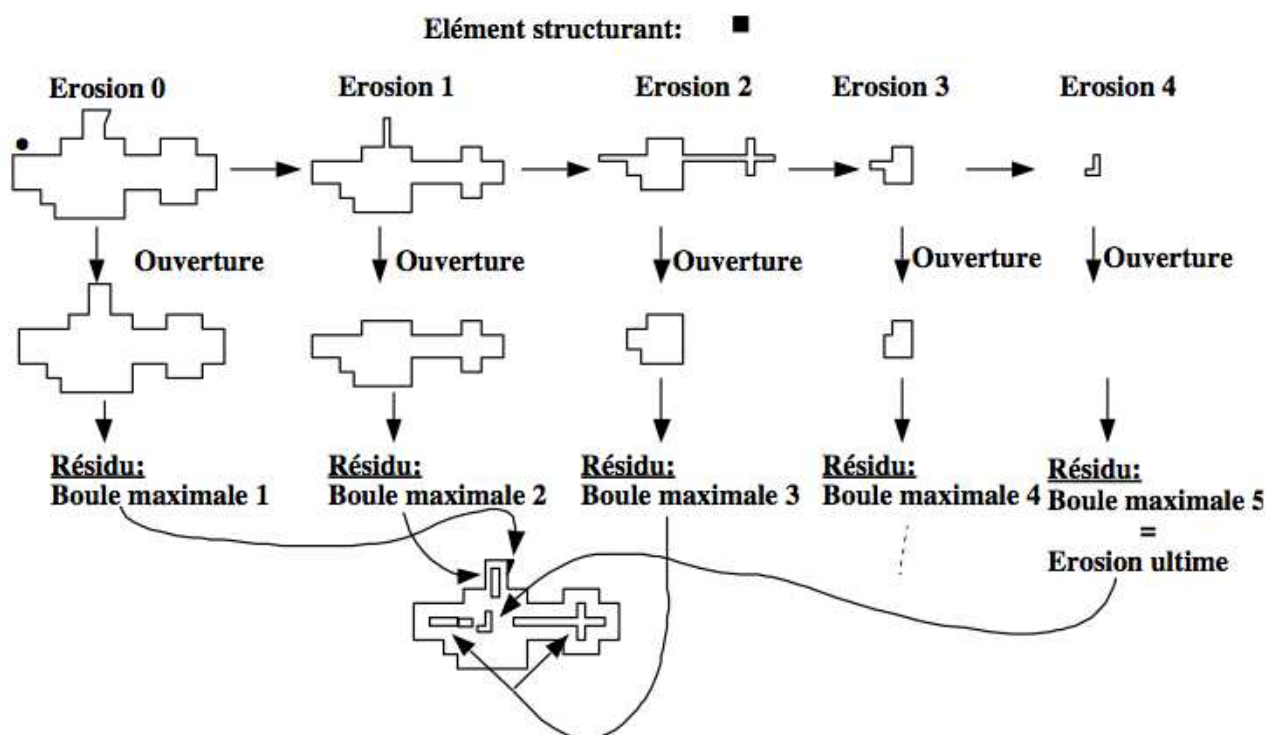
Résidus – p. 16/21

# Squelettes

- L'érosion ultime était déjà un lieu de centre de boules maximales
- Définition: (H. Blum): le *squelette* d'un ensemble  $X$  selon une famille de boules  $\{\delta_n\}$  est le lieu de tous les centres de boules maximales.



# Construction



---

## Formule de Lantuéjoul

---

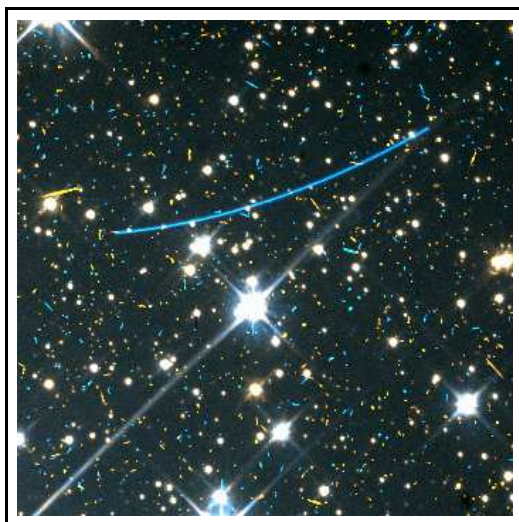
L'algorithme de squelettisation est le même que celui de l'érosion ultime. Il suffit de remplacer l'ouverture par reconstruction par une ouverture unitaire:

$$S(X) = \bigcup_i \{S_i(X)\} = \bigcup_i \{\epsilon_i(X) \setminus \gamma_1[\epsilon_i(X)], i = 0 \dots N\}$$

---

## Une application : détection de trace d'astéroïdes

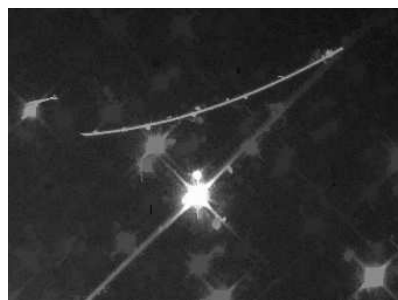
---



## Traces d'astéroïdes, suite.



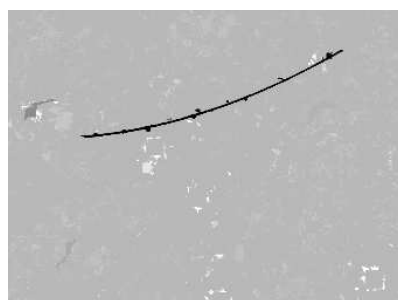
(traces)



(ouv. chemins)



(ouv. droites)



(Différence)

---

# *Cours de Morphologie Mathématique*

## *Segmentation*

Hugues Talbot

talboth@esiee.fr

ISBS / ESIEE

1<sup>er</sup> semestre 2004-2005

---

Segmentation – p. 1/37

---

### TP1: fini?

- Tout le monde a bien fini son rapport ?
- Il me manque:

---

Segmentation – p. 2/37

---

## Rappel du cours précédent

---

- Suite de la géodesie, en niveaux de gris.
- Résidus: gradients, top-hat (chapeau haut-de-forme).
- Érodés ultimes
- Squelettes.

---

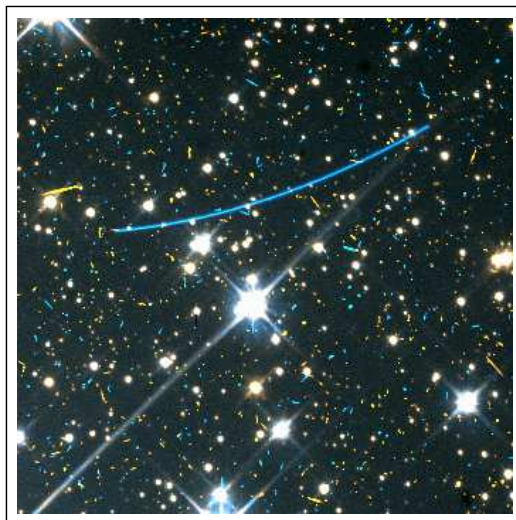
Segmentation – p. 3/37

---

---

## Une application : détection de trace d'astéroïdes

---



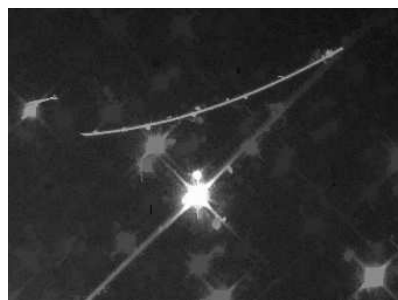
---

Segmentation – p. 4/37

## Traces d'astéroïdes, suite.



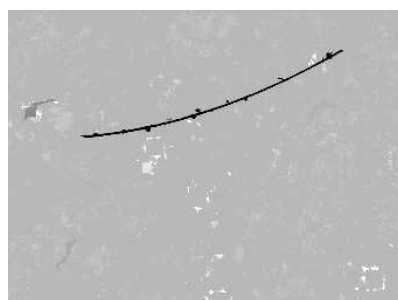
(traces)



(ouv. chemins)



(ouv. droites)



(Différence)

Images NASA/JPL/HST.

Segmentation – p. 5/37

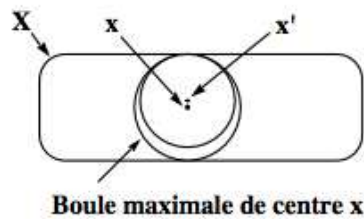
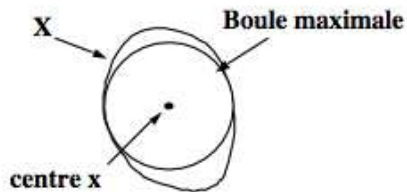
*Squelettes et amincissements*

Segmentation – p. 6/37

## Boules maximales

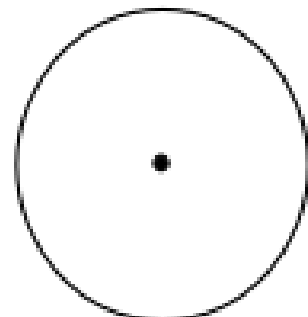
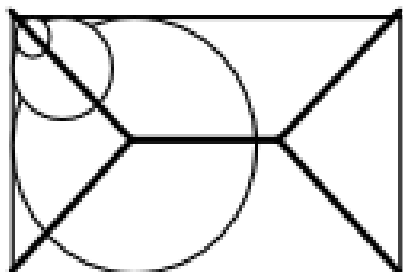
- Une boule  $\delta_n(x)$  de centre  $x$  et de taille  $n$  est maximale vis-à-vis de l'ensemble  $X$  s'il n'existe aucun autre indice  $k$  et aucun autre centre  $y$  tel que:

$$\delta_n(x) \in \delta_k(y) \in X$$



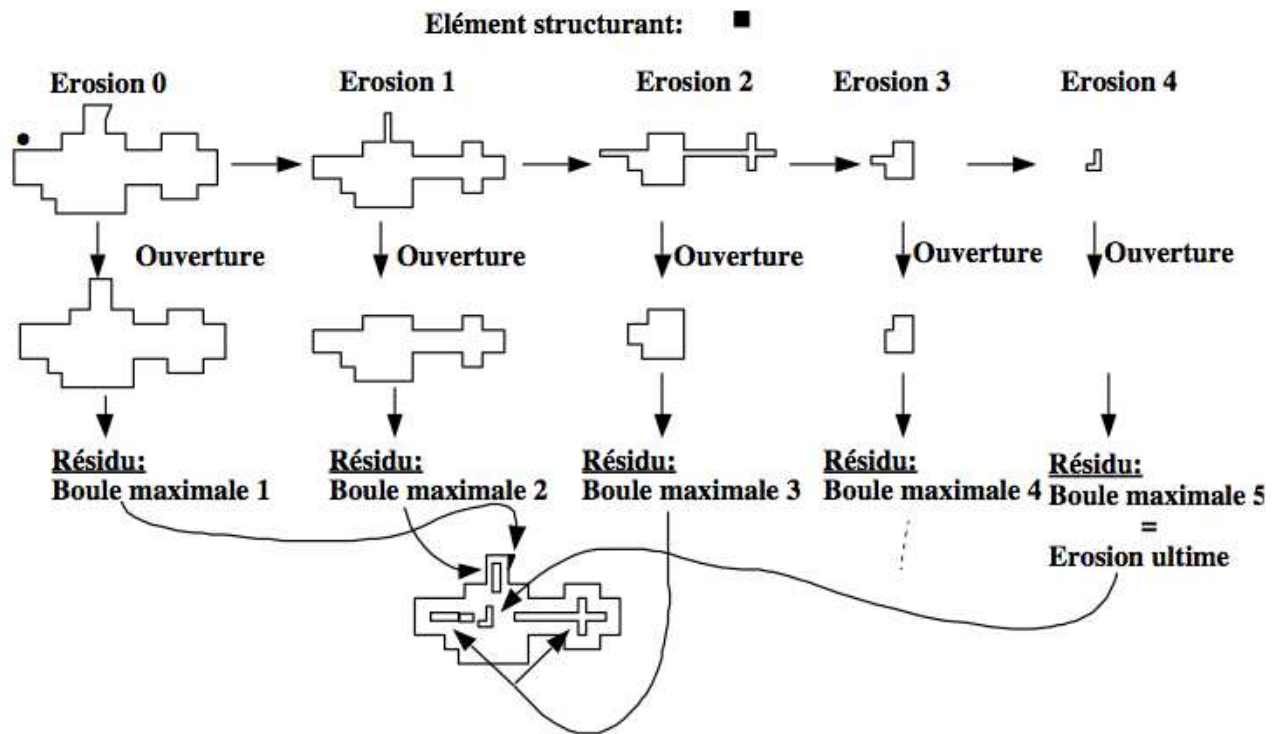
## Squelettes

- L'érosion ultime était déjà un lieu de centre de boules maximales
- Définition: (H. Blum): le *squelette* d'un ensemble  $X$  selon une famille de boules  $\{\delta_n\}$  est le lieu de tous les centres de boules maximales.





# Construction



# Formule de Lantuéjoul

L'algorithme de squelettisation est le même que celui de l'érosion ultime. Il suffit de remplacer l'ouverture par reconstruction par une ouverture unitaire:

$$S(X) = \bigcup_i \{S_i(X)\} = \bigcup_i \{\epsilon_i(X) \setminus \gamma_1[\epsilon_i(X)], i = 0 \dots N\}$$

Lien avec la fonction distance: les érodés ultimes sont les maxima régionaux de la fonction distance. Le squelette de Lantuéjoul est constitué des maxima locaux de la fonction distance.

## Squelette connexe

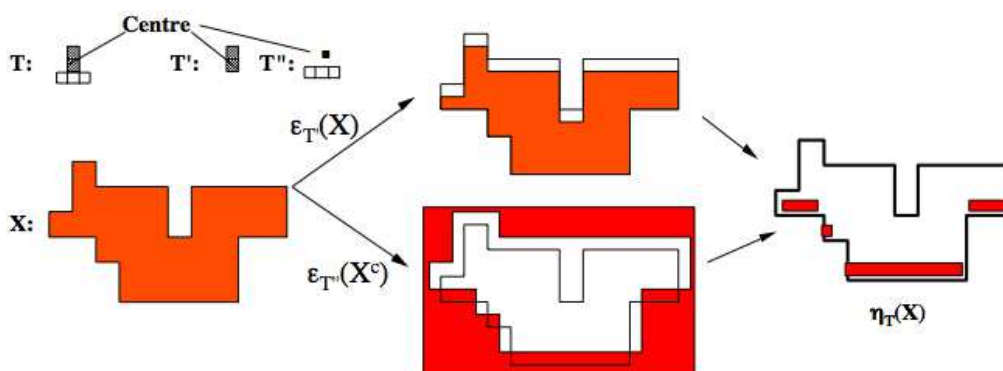
- La formule de Lantuéjoul donne des squelettes non connexes.
- Dans le cas du plan euclidien, on peut montrer qu'avec les bonnes hypothèses le squelette d'un objet compact est connexe.
- Par conséquent il est utile de disposer de méthodes qui donnent un squelette connexe pour des objets connexes:
  - squelette par amincissements
  - squelette par file d'attentes
  - squelette par parcours du contour
  - et bien d'autres méthodes.

Segmentation – p. 11/37

## Transformée en tout-ou-rien

- La transformée en tout-ou-rien généralise l'érosion et la dilatation.
- On utilise un couple d'éléments structurants disjoints  $T = \{T', T''\}$ .

$$\eta_T(X) = \{z, T'' \subseteq X^c \text{ et } T' \subseteq X\} = \epsilon_{T'}(X) \cap \epsilon_{T''}(X^c)$$

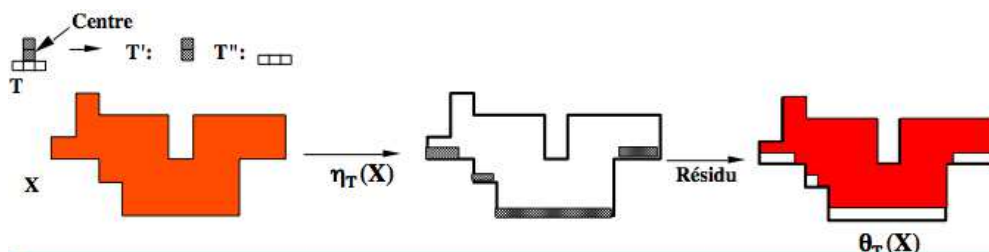


Segmentation – p. 12/37

## Amincissements, épaisissements

- L'amincissement  $\theta_T$  est le résidu entre l'ensemble initial et sa transformée en tout-ou-rien:

$$\theta_T(X) = X \setminus \eta_T(X) = X \setminus [\epsilon_{T'}(X) \cap \epsilon_{T''}(X^c)]$$



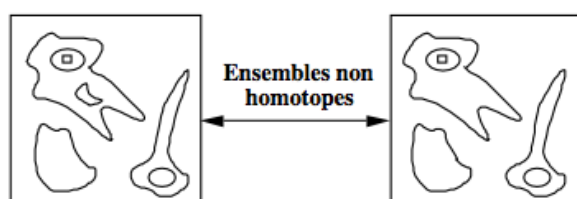
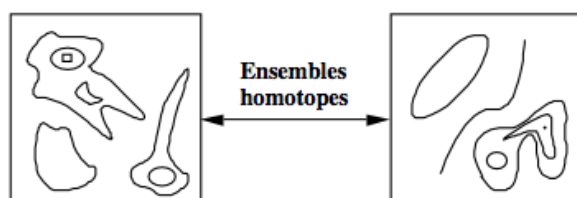
- L'épaissement est défini par dualité.

$$\xi_T(X) = X \cup \eta_T(X^c)$$

- On a:  $\theta_T(X) = [\xi_T(X^c)]^c$ .

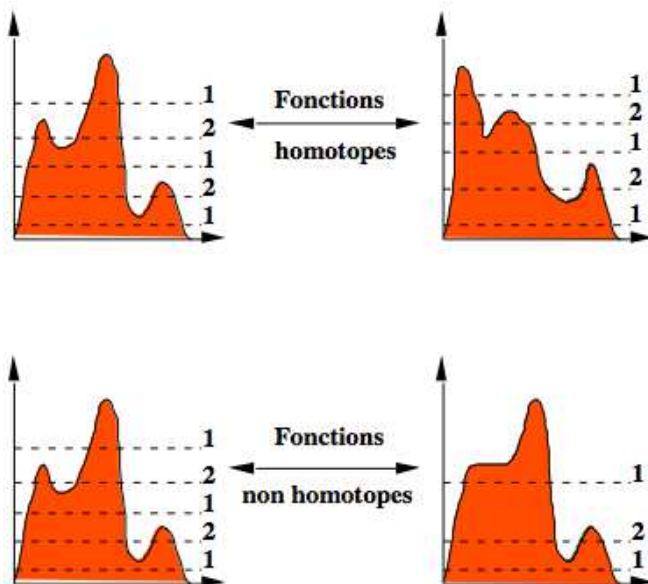
## Homotopie pour les ensembles

- Deux ensembles sont homotopes s'il existe une transformation bi-continue pour passer de l'un à l'autre.
- c-à-d l'ensemble de départ et son transformé sont tels que dans l'un et dans l'autre chaque grain contient le même nombre de pores, et chaque pore contient le même nombre de grains.



## Homotopie pour les fonctions

- Pour les fonctions, on définit l'homotopie à partir de leurs sections planes.
- Deux fonctions sont homotopes si toutes leur sections à même niveau sont homotopes entre elles.



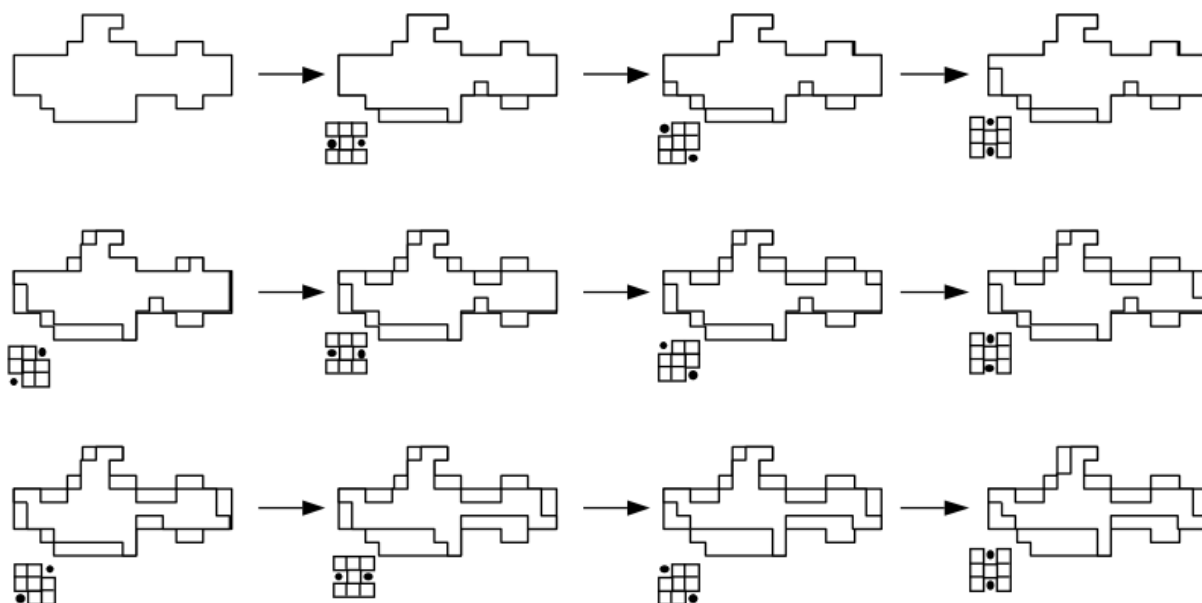
Segmentation – p. 15/37

## Amincissement homotopique

- Un amincissement ou un épaississement sont homotopiques s'ils utilisent un couple d'éléments structurants  $T'$ ,  $T''$  qui préserve l'homotopie.
- Exemple en 8-connexité ?
- En utilisant ce couple et ses rotations de façon séquentielle on génère un *amincissement limite* qui est homotope à l'ensemble de départ et qui ressemble à un squelette:
  - Il est mince
  - Il est homotope à l'ensemble de départ
  - Il est "centré" dans l'ensemble de départ
  - La transformée est idempotente et anti-extensive (Q: est-ce une ouverture ?)

Segmentation – p. 16/37

## Amincissement séquentiel



Segmentation – p. 17/37

## Exercice (TD)

- Amincissement d'un rectangle  $9 \times 5$
- Amincissement d'un rectangle  $9 \times 7$  auquel on enlève le pixel du milieu.

Segmentation – p. 18/37

---

## *segmentation*

---

Segmentation – p. 19/37

---

### Qu'est-ce que la segmentation ?

- Découper une image en régions de contenu sémantique similaire.
- Normalement la frontière d'une région devrait définir la frontière d'un objet.
- En fait les régions segmentées sont des aires de l'image avec des propriétés mesurables similaires.
- La segmentation d'objets requiert une connaissance de haut niveau, qui peut être soit incluse dans la méthode elle-même ou utilisée après la segmentation.

---

Segmentation – p. 20/37

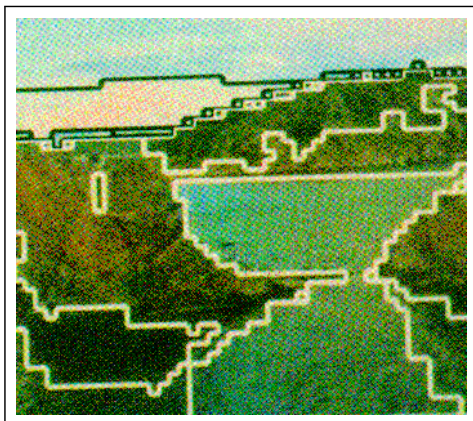
# Méthodes de segmentation

Il existe de nombreuses méthodes:

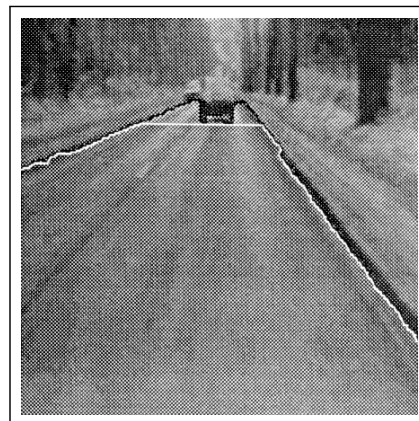
- Seuillage
- Méthodes à partir de l'histogramme d'une image
- Clustering spatial
- Découpage et unions de régions
- Contours actifs
- Lignes de niveaux
- À base de connaissance (IA)
- Méthodes morphologiques

Segmentation – p. 21/37

## Top-down ou bottom-up



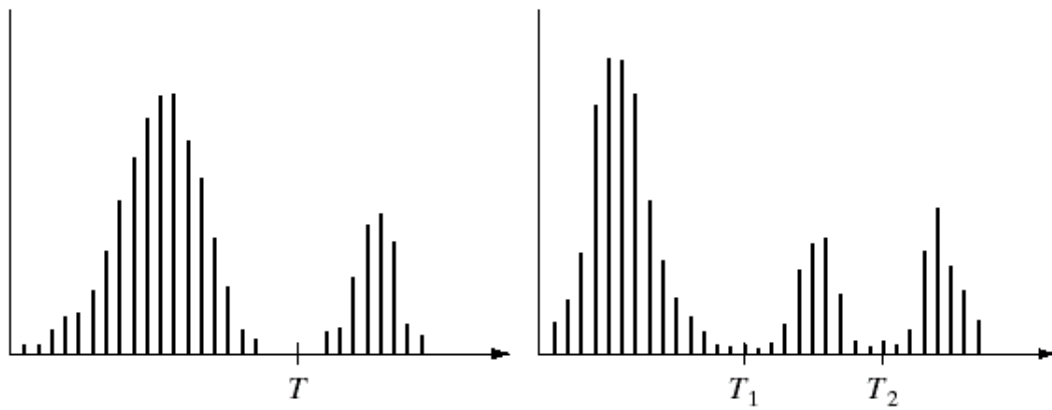
Top-down



Bottom-up

Segmentation – p. 22/37

## Segmentation au niveau du pixel

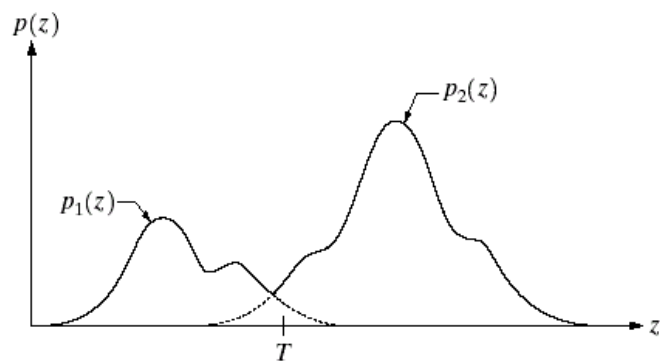


a b

**FIGURE 10.26** (a) Gray-level histograms that can be partitioned by (a) a single threshold, and (b) multiple thresholds.

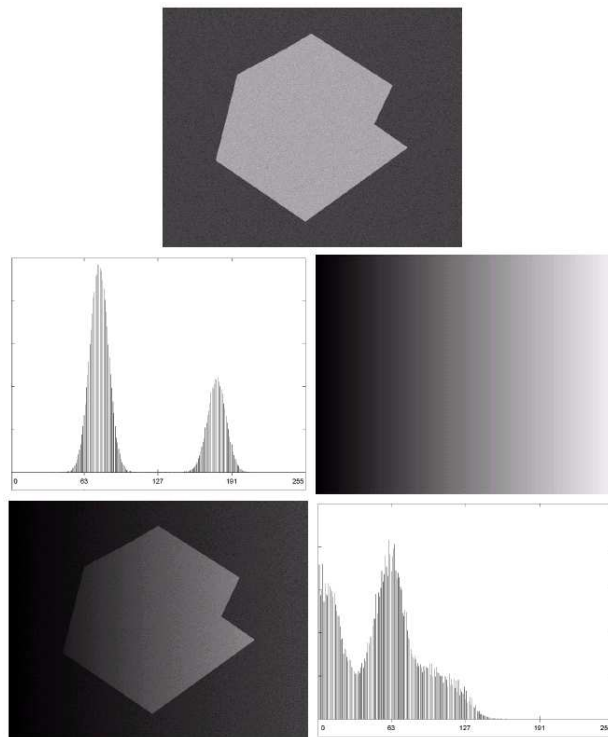
## Modèle

**FIGURE 10.32**  
Gray-level probability density functions of two regions in an image.





## Modèle (suite)

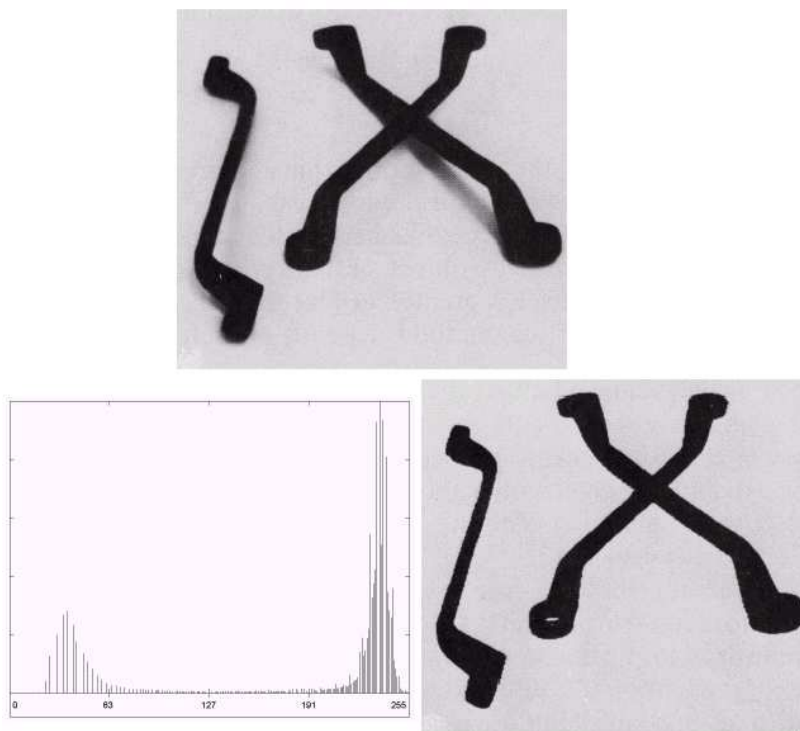


a  
b c  
d e

**FIGURE 10.27**  
(a) Computer generated reflectance function.  
(b) Histogram of reflectance function.  
(c) Computer generated illumination function.  
(d) Product of (a) and (c).  
(e) Histogram of product image.

Segmentation – p. 25/37

## Seuillage

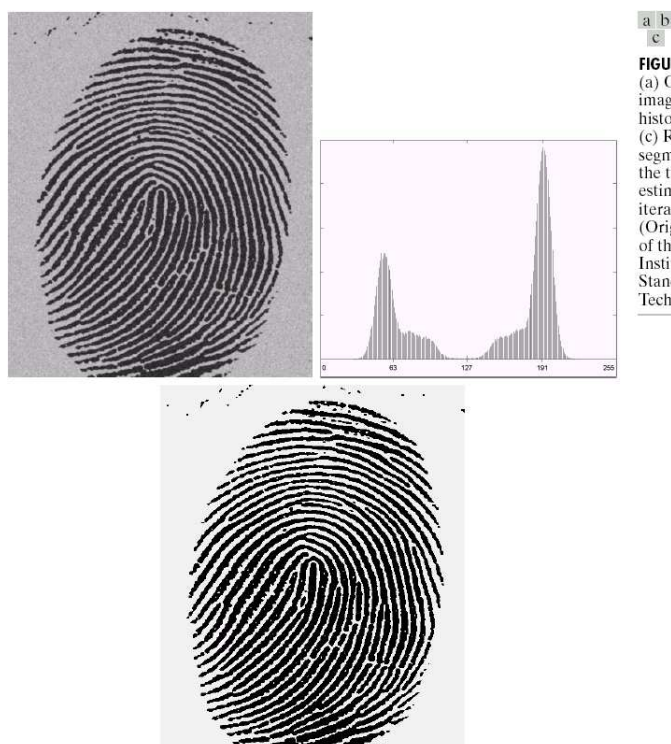


a  
b c

**FIGURE 10.28**  
(a) Original image. (b) Image histogram.  
(c) Result of global thresholding with  $T$  midway between the maximum and minimum gray levels.

Segmentation – p. 26/37

## Exemple de seuillage

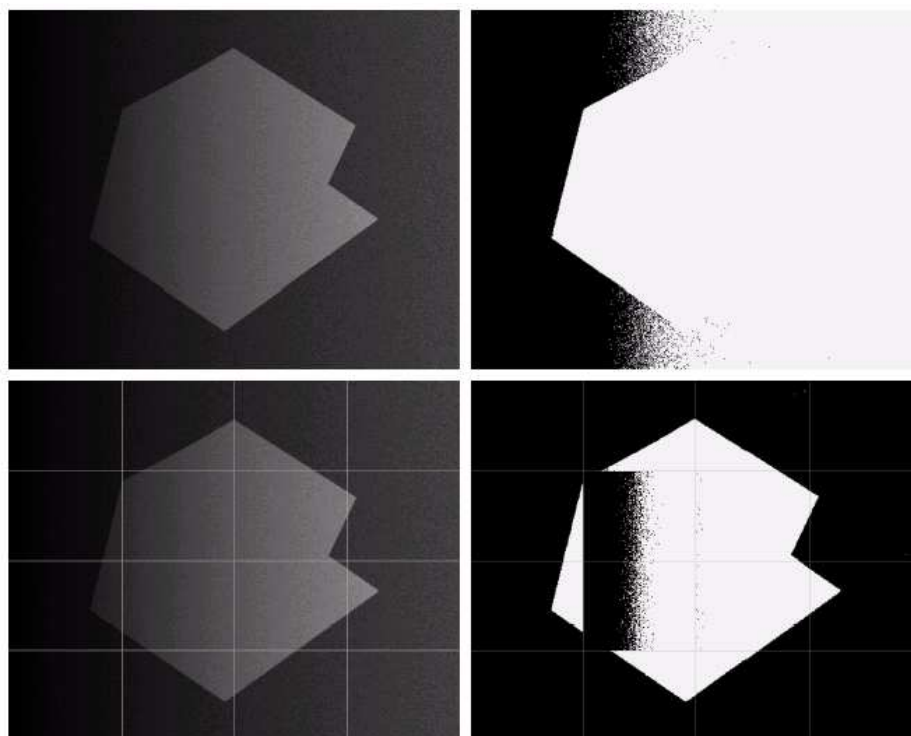


**FIGURE 10.29**  
(a) Original image. (b) Image histogram. (c) Result of segmentation with the threshold estimated by iteration. (Original courtesy of the National Institute of Standards and Technology.)

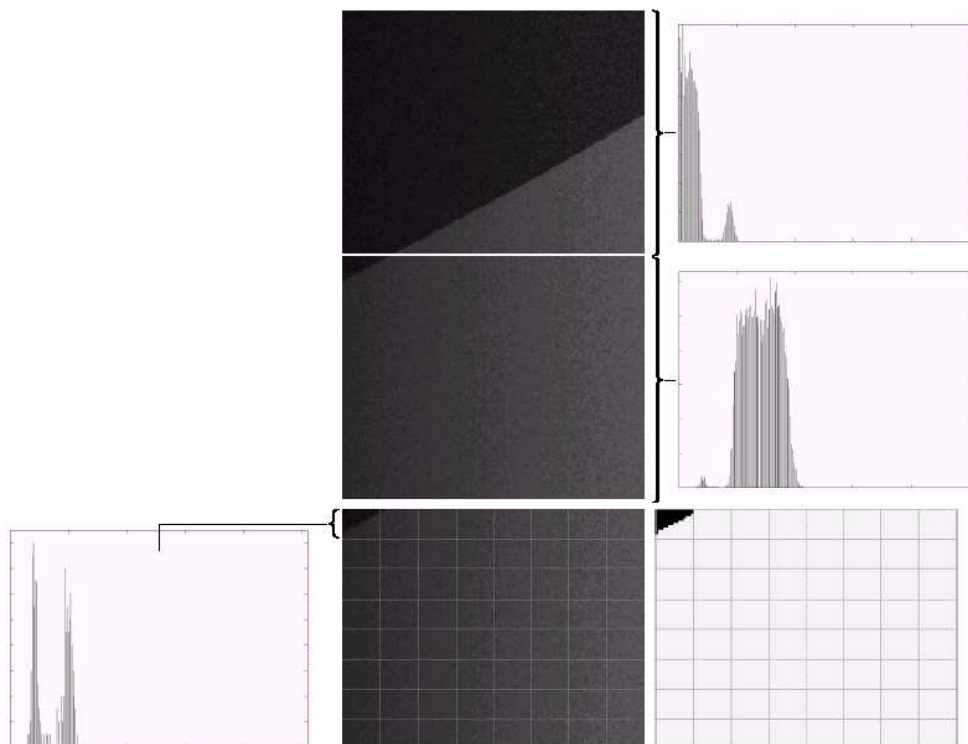
## Seuillage adaptatif

a b  
c d

**FIGURE 10.30**  
(a) Original image. (b) Result of global thresholding. (c) Image subdivided into individual subimages. (d) Result of adaptive thresholding.



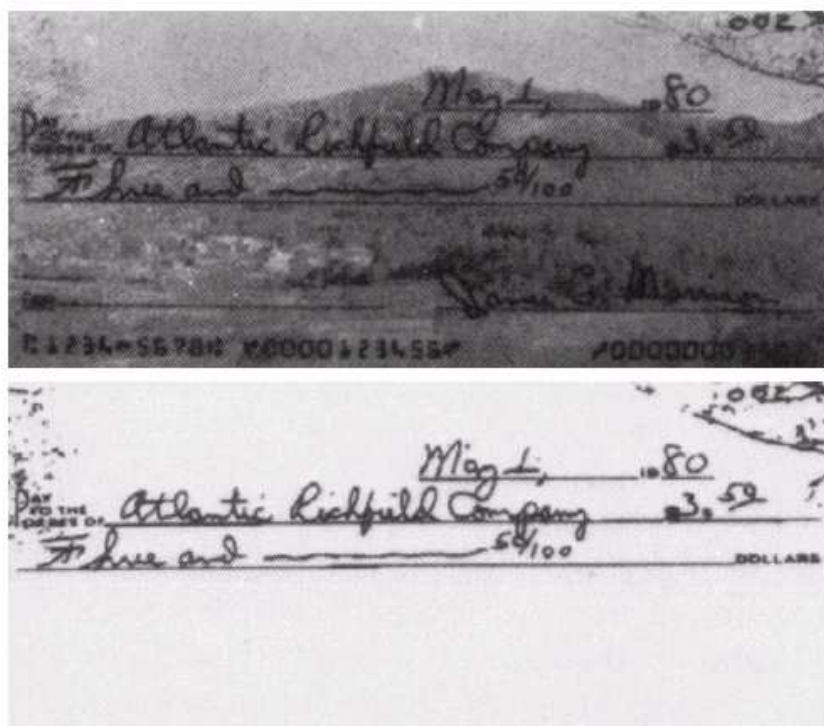
## Seuillage adaptatif (suite)



## Seuillage adaptatif (suite)

a  
b

**FIGURE 10.37**  
 (a) Original image. (b) Image segmented by local thresholding. (Courtesy of IBM Corporation.)



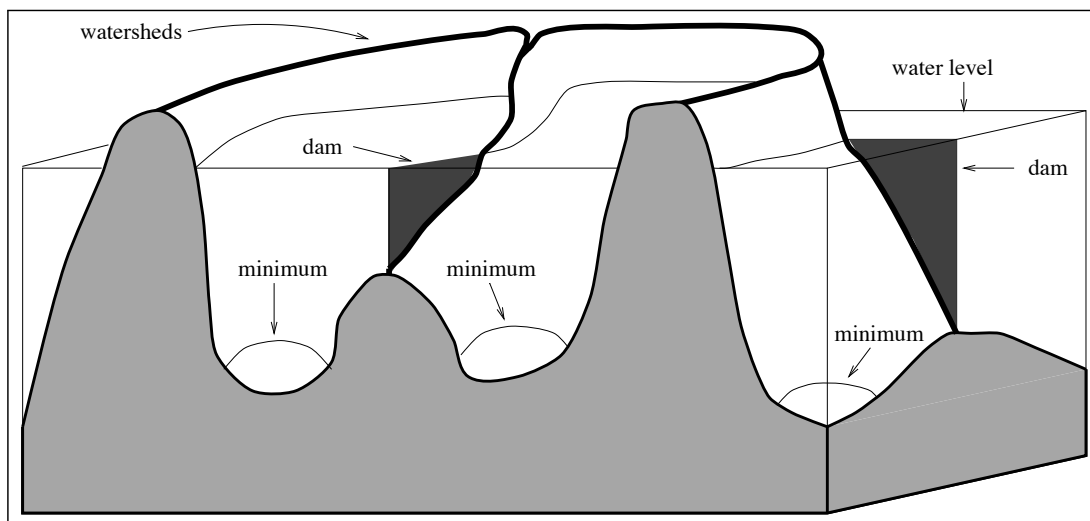
## Seuillage multispectral



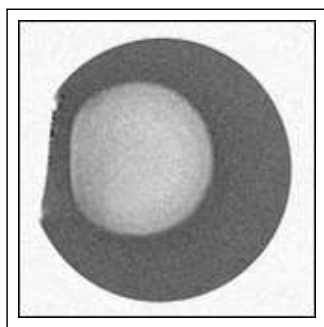
a b c

**FIGURE 10.39** (a) Original color image shown as a monochrome picture. (b) Segmentation of pixels with colors close to facial tones. (c) Segmentation of red components.

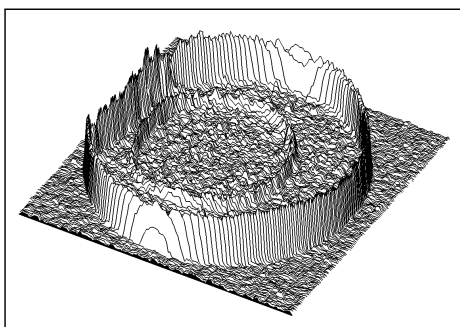
## La ligne de partage des eaux



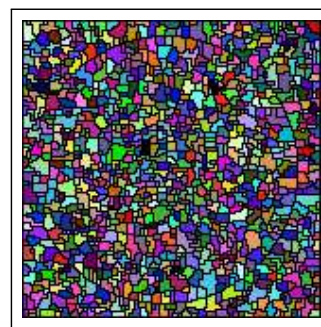
## The watershed line 2



Original

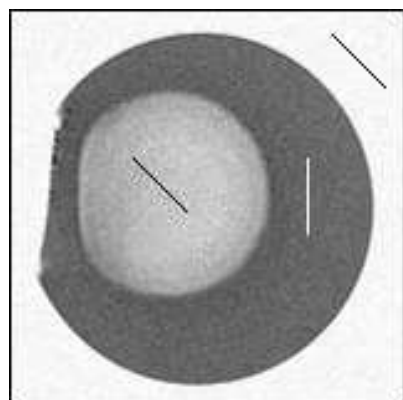


Gradient

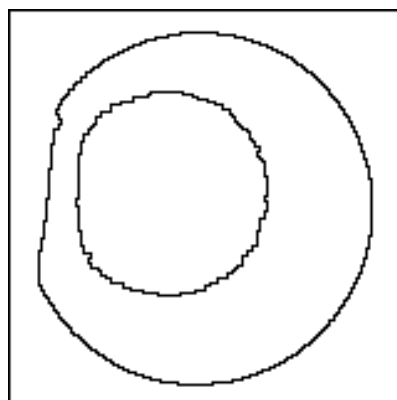


First result

## The watershed line 3



Orig. + Markers



Final

---

## How to use the watershed line?

---

The classical CMM approach:

- Find a good gradient
- Find good external markers
- Find good internal markers

Other approaches:

- Merge catchment basins afterwards (graph approach f.e.)
- Constrain the watershed

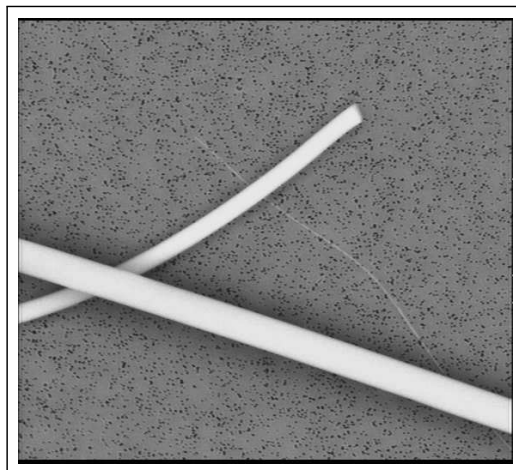
---

Segmentation – p. 35/37

---

## Example of the classical approach (1)

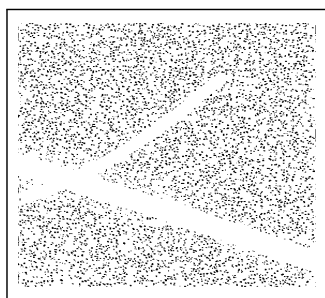
---



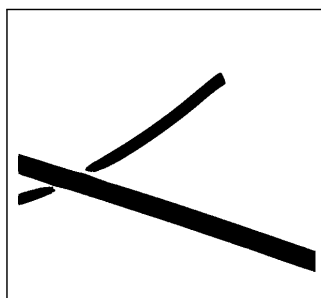
---

Segmentation – p. 36/37

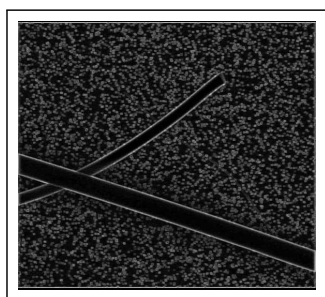
## Examples of the classical approach (2)



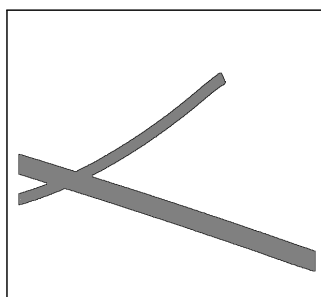
internal



external



gradient



final