

Table des matières

Table des matières	1
1 Introduction à morphologie mathématique	3
1.1 Notions de base	3
1.1.1 La trame digitale	3
1.1.2 Images binaires et à niveaux de gris	7
1.1.3 Opérations de base sur les ensembles	8
1.1.4 propriétés des opérations sur les ensembles	8
1.2 Transformations morphologiques	10
1.2.1 Erosion et dilatation	10
1.2.1.1 Éléments structurants	10
1.2.1.2 Définitions	11
1.2.1.3 Une application : les gradients	14
1.2.2 Transformations en tout ou rien	14
1.2.3 Transformées géodésiques	15
1.2.3.1 Transformées élémentaires géodésiques	15
1.2.3.2 Reconstruction	15
1.2.4 Ouvertures et Fermetures	17
1.2.4.1 Les transformations chapeaux hauts-de-forme	18
1.3 Filtrage	19
1.4 Ligne de partage des eaux	20
1.5 Mise en œuvre informatique	23
Conclusion	26
Table des figures	27
Index	27

Chapitre 1

Introduction à la morphologie mathématique

La Morphologie Mathématique (MM) est une théorie de l'analyse d'image dont les premiers débuts datent de la fin des années soixante avec les travaux de G. Matheron et J. Serra, qui ont continués les travaux de recherche de H. Hadwiger [?, ?] et de H. Minkowski [?, ?] sur la théorie des ensembles. La plus grande partie de cette théorie a été développée au Centre de Morphologie Mathématique (CMM), à l'École des Mines de Paris.

La MM a pris une importance internationale depuis plusieurs années, si on compte le nombre de conférences, articles, livres qui la présentent ou au moins lui font une place, ainsi que le nombre croissant de matériels spécialisés, de programmes d'application et de réalisations industrielles qui y font appel.

Ce chapitre n'est pas un cours de morphologie mathématique, mais simplement un rappel de ce qu'on a besoin de savoir pour bien comprendre le cours.

Pour plus d'informations, on lira avec profit les ouvrages de référence de cette discipline : les deux livres de Serra [?, ?], ceux de Matheron [?, ?], le livre de Soille [?]. On peut aussi citer le livre de Coster et Chermant [?], la thèse de Beucher [?] et celle de Vincent [?].

1.1 Notions de base

Nous nous limiterons dans le cadre de cette présentation le plus souvent au cas discret, et aux images à deux dimensions, sauf quand l'évocation du cas général sera plus clair. La plupart des transformations que nous allons définir s'appliquent dans un sous-ensemble fini de \mathbb{Z}^2 , pour des raisons liées à la technologie de l'analyse d'image (les ordinateurs ne pouvant faire appel à une résolution infinie ou à des images de dimensions infinies).

1.1.1 La trame digitale

Les points de la trame bidimensionnelle sont généralement appelés *pixels* (pour *picture elements*). La grille ou trame digitale permet de définir les relations de voisinage entre les pixels d'une image. Ce graphe, notée G n'est qu'un ensemble de couples de pixels (soit un sous-ensemble de $\mathbb{Z}^2 \times \mathbb{Z}^2$). C'est à dire :

$$\forall p, q \in \mathbb{Z}^2, p \text{ voisin de } q \Leftrightarrow (p, q) \in G$$

On suppose toujours qu'un point n'est pas son propre voisin et que la relation *est voisin de* est transitive et symétrique. G est donc un graphe non-orienté sur \mathbb{Z}^2 . Le voisinage d'un pixel p au sens de la trame digitale est :

$$\forall p \in \mathbb{Z}^2, N_G(p) = \{q \in \mathbb{Z}^2, (p, q) \in G\} \quad (1.1)$$

On peut étendre cette définition à tout ensemble A de pixels en considérant l'ensemble des pixels de \mathbb{Z}^2 voisins des pixels de A et n'appartenant pas à A :

$$\forall A \subset \mathbb{Z}^2, N_G(A) = \{q \in \mathbb{Z}^2 \setminus A, \exists p \in A, (p, q) \in G\}$$

Le chemin entre deux pixels p et q est défini de la façon suivante :

Définition 1.1 *Un chemin C de longueur $l(C) = n$ et d'extrémités p et q dans la trame G est un $(n + 1)$ -uplet $(p_0, p_1, p_2, \dots, p_n)$ de pixels tels que :*

- $p_0 = p$ et $p_n = q$,
- $\forall i \in [1, n], p_i \in N_G(p_{i-1})$.

On appelle *lacet* un chemin dont les deux extrémités coïncident. La notion de chemin permet d'introduire deux notions très utiles :

- *La connexité.* Soit A un ensemble de pixels inclus dans \mathbb{Z}^2 et x un pixel de A . La composante connexe $C_x(A)$ qui contient x est l'union des chemins d'origine x inclus dans A .
- *La distance d_G* associée à G . La distance entre deux pixels p et q est donnée par la longueur minimale des chemins qui les joignent.

$$\forall p, q \in \mathbb{Z}^2, d_G(p, q) = \inf \{l(C), C \text{ chemin joignant } p \text{ et } q \text{ dans } G\}$$

On peut définir sur la trame sur laquelle on travaille une *distance euclidienne* en plongeant cette trame dans \mathbb{R}^2 .

$$d(p, q) = \|\vec{pq}\|$$

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 1.1).

Le plan discret \mathbb{Z}^2 est généralement représenté par une trame bidimensionnelle cartésienne, soit carrée, soit hexagonale (voir figure 1.2 et 1.3).

Les morphologistes préfèrent généralement travailler sur une trame hexagonale (6-connexité), car celle-ci possède de meilleures propriétés : elle est plus isotrope que la trame carrée et elle vérifie la *propriété de Jordan digitale* [?], selon laquelle tout lacet simple non dégénéré, c'est à dire sans point double, qui ne se touche pas et qui n'est pas réduit à un point sépare le plan \mathbb{Z}^2 en deux composantes distinctes dont l'une est d'extension finie et l'autre infinie. La trame carrée, elle, ne vérifie pas cette propriété (voir figure 1.4). En effet, en trame carrée, on doit déjà décider si un point possède comme voisin les quatre points les plus proches (4-connexité) ou en comptant en plus les 4 points diagonaux, les 8 points les plus proches (8-connexité). On est souvent obligé, lorsqu'on met en œuvre une transformation sur une trame digitale carrée de considérer le fond en 4-connexité et les objets en 8-connexité, ou bien réciproquement.

Malgré tous ses défauts la trame carrée est toujours la plus utilisée, car l'immense majorité du matériel de traitement d'images (cartes d'acquisition, caméras, écrans, etc) travaille dans cette trame.

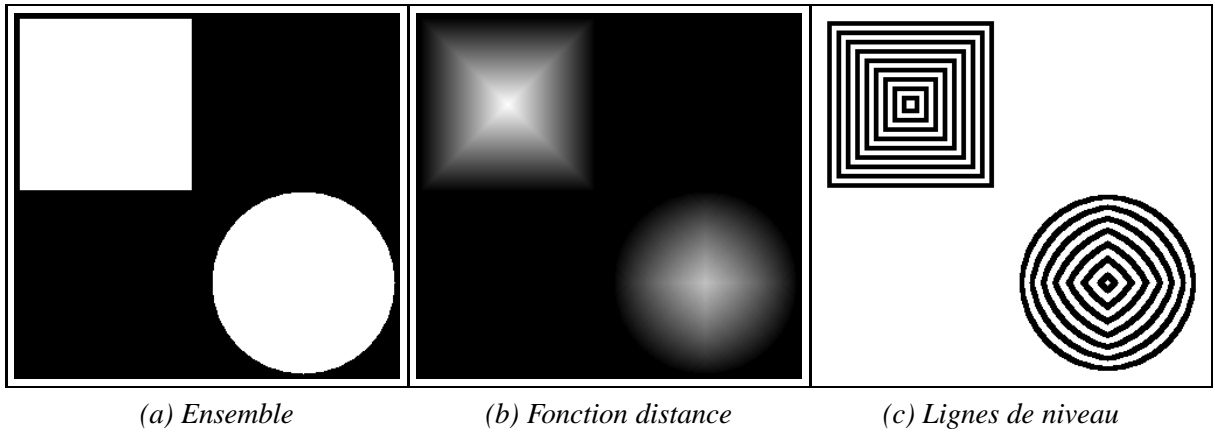


FIG. 1.1: La fonction distance d'un ensemble est l'application qui en chaque point de l'ensemble associe sa distance au complémentaire. Fonction distance ici en 8-connexité

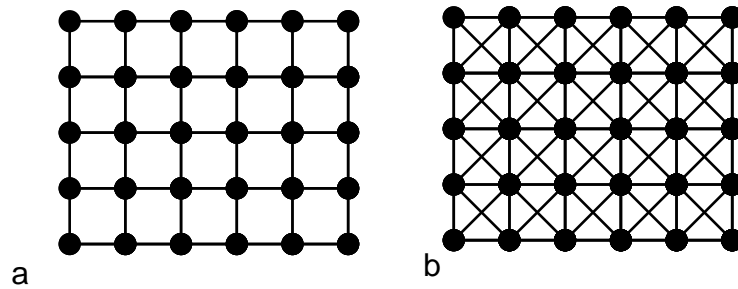


FIG. 1.2: Portions de trame carrée. On peut choisir d'affecter à chaque point 4 voisins (trame a) ou 8 voisins (trame b).

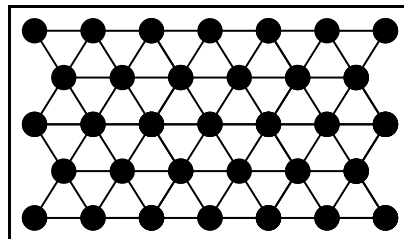


FIG. 1.3: Portion de trame hexagonale.

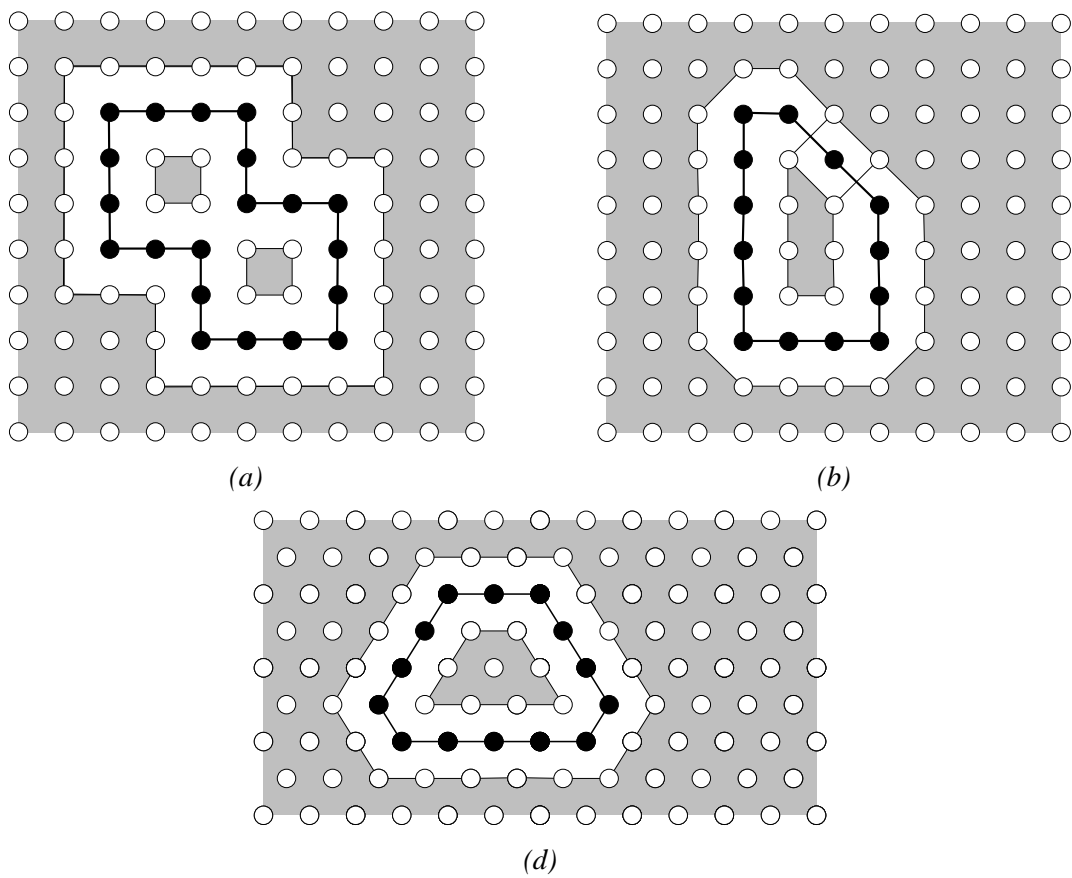


FIG. 1.4: Propriété de Jordan digitale. La trame carrée ne vérifie pas cette propriété. En *a* le lacet simple non dégénéré sépare le plan en trois composantes connexes. En *b* il ne sépare rien du tout. En *c* la propriété de Jordan est vérifiée, comme toujours en trame hexagonale.

Il est possible de convertir une image 6-connexe en image 4- ou 8-connexe, mais on perd toujours un peu de précision.

De toute manière un grand nombre de propriétés intuitives dans le plan continu ne sont plus vraies dans la trame digitale. Par exemple il existe souvent plus d'un chemin minimal entre deux points (figure 1.5). Il est parfois possible et utile (voir la fin de ce chapitre) de considérer le plongement de la trame discrète dans le plan euclidien. Cette approche apporte cependant son lot de difficultés. Nous en expliciterons quelques unes au chapitre ??.

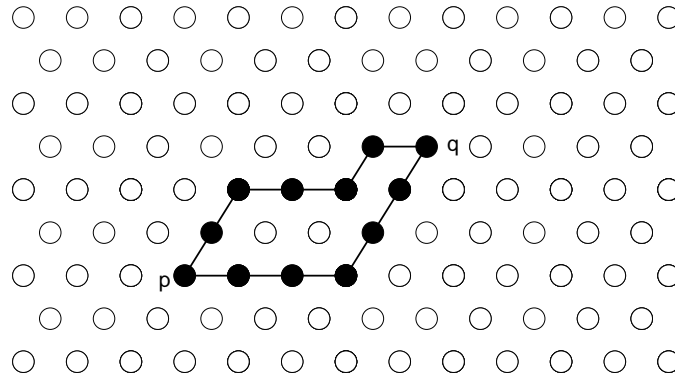


FIG. 1.5: Plusieurs chemins distincts peuvent être de longueur minimale dans une trame discrète.

1.1.2 Images binaires et à niveaux de gris

Définition 1.2 Une image bidimensionnelle discrète binaire I est une application d'un sous-ensemble \mathcal{D}_I de \mathbb{Z}^2 dans $[0, 1]$.

$$I : \mathcal{D}_I \subset \mathbb{Z}^2 \rightarrow [0, 1] \quad (1.2)$$

Pour chaque pixel d'une image binaire, on associe la valeur 1 ou 0 suivant qu'il fait partie d'un objet ou du fond. Classiquement le domaine de définition \mathcal{D}_I d'une image est rectangulaire. En pratique une image bidimensionnelle peut être considérée comme un tableau rectangulaire de données. Dans le cas des images binaires ces données sont des *bits* (0 ou 1), d'où le nom *bitmap* en anglais, qui les désigne le plus souvent.

Définition 1.3 Une image bidimensionnelle discrète à niveaux de gris I est une application d'un sous-ensemble \mathcal{D}_I de \mathbb{Z}^2 dans \mathbb{Z} .

$$I : \mathcal{D}_I \subset \mathbb{Z}^2 \rightarrow \mathbb{Z} \quad (1.3)$$

Une image binaire n'est donc rien d'autre qu'une image à niveaux de gris à deux niveaux : 0 et 1. La figure 1.6 présente des exemples d'images binaire et à niveaux de gris.

Sur l'image binaire de cette figure, le niveau 0 est en blanc et le niveau 1 est en noir. À l'opposé, sur l'image en niveaux de gris, un niveau de gris est d'autant plus élevé qu'il apparaît clair sur l'image. On respectera toujours cette convention.

Pour mieux comprendre et interpréter le résultat des transformations morphologiques sur les images à niveau de gris, il est utile de les considérer comme un relief topographique, chaque pixel ayant une élévation proportionnelle à son intensité. Cette analogie entre les reliefs et les images est féconde ; en effet, cette représentation permet d'appliquer les transformations valables sur les ensembles aux images à niveaux de gris. On définit le *sous-graphe* SG d'une image à niveaux de

gris comme la partie de l'espace à trois dimensions située en dessous du graphe de l'image. Plus précisément :

$$SG(I) = \{(x, t) \in \mathbb{Z}^2 \times \mathbb{Z}, t \leq I(x)\} \quad (1.4)$$

La figure 1.7 présente quelques exemple de sous-graphes.

1.1.3 Opérations de base sur les ensembles

Les opérations booléennes de base sur les ensembles sont l'intersection \cap et la réunion \cup . On peut appliquer ces opérateurs directement aux images binaires. Pour les images à niveau de gris on doit utiliser la notion de *sup* \vee et d'*inf* \wedge . Dans le cas digital, ces notions sont équivalentes aux notions de maxima et minima (voir figure 1.8, dans le cas des images 1D). Pour tout I et J images on a :

$$\forall x \in \mathcal{D}_I \begin{cases} (I \vee J)(x) & = \max(I(x), J(x)) \\ (I \wedge J)(x) & = \min(I(x), J(x)) \end{cases} \quad (1.5)$$

On peut noter que ces opérateurs peuvent être définis à partir de leur sous-graphe ; en effet :

$$SG(I \vee J) = SG(I) \cup SG(J) \quad (1.6)$$

$$SG(I \wedge J) = SG(I) \cap SG(J) \quad (1.7)$$

Un autre opérateur important est la *complémentation*. La complémentation d'une image I notée I^c est obtenue par

$$\forall x \in \mathcal{D}_I, f^c(x) = MAX - f(x) \quad (1.8)$$

où MAX est la plus grande valeur autorisée pour l'image en cours (par exemple, 1 pour une image binaire, 255 pour une image codée sur 8 bits).

Une dernière opération dont nous avons besoin est la *translation*. On notera la translation d'une image I par un vecteur \vec{b} par I_b , et on définit cette translation par :

$$\forall x \in I, I_b(x) = I(x - \vec{b}) \quad (1.9)$$

1.1.4 propriétés des opérations sur les ensembles

La plupart des opérateurs morphologiques sont définis comme des opérations sur les ensembles. Voici un rappel des principales propriétés qui peuvent être associées aux opérations sur les ensembles :

- **idempotence**. Une opération f est *idempotente* si elle donne le même résultat qu'elle soit appliquée une fois ou bien deux fois de suite.

$$f \text{ idempotent} \Leftrightarrow \forall I \text{ image}, f(f(I)) = f(I). \quad (1.10)$$

- **extensivité**. Une transformation f est *extensive* si son résultat est plus grand que l'image de départ, pour toute image. Une opération est *anti-extensive* si son résultat est plus petit que l'image de départ.

$$f \text{ extensive} \Leftrightarrow \forall I \text{ image}, f(I) \geq I. \quad (1.11)$$

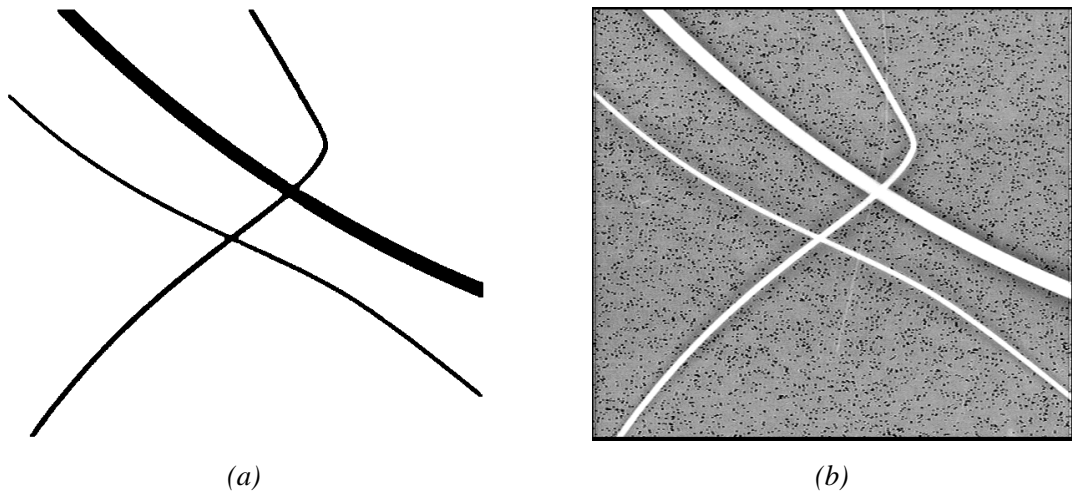


FIG. 1.6: Exemple d'images binaire et à niveaux de gris.

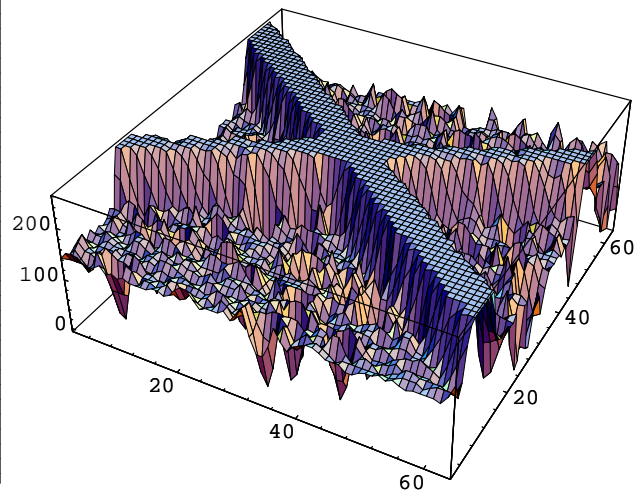
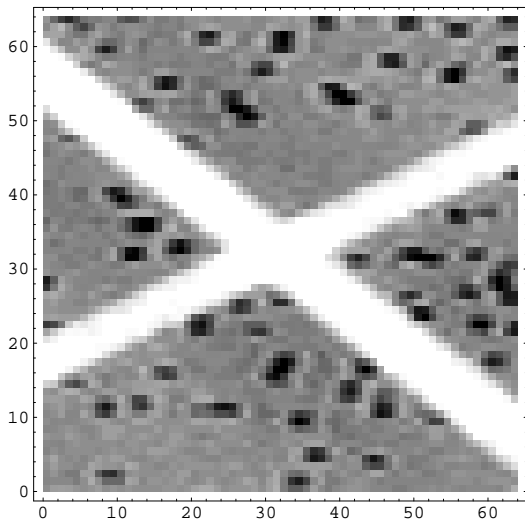
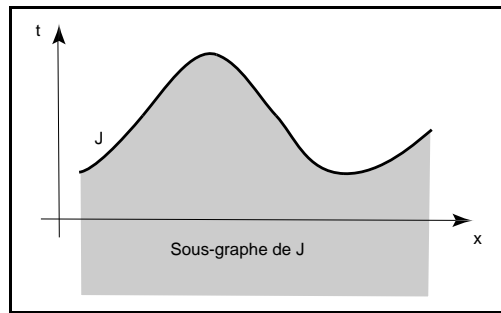
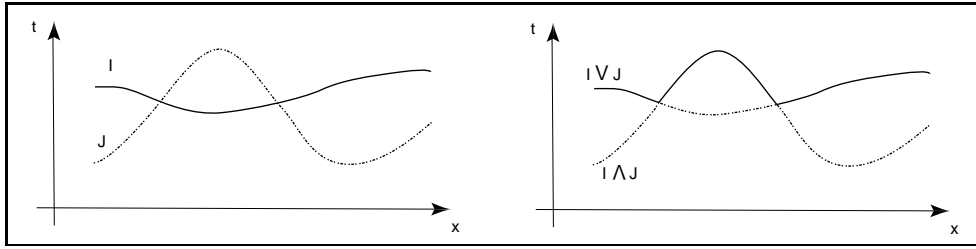


FIG. 1.7: Sous-graphe d'un signal monodimensionnel et d'une image.

FIG. 1.8: Sup et inf de deux images I et J .

– **croissance.** Une transformation f est *croissante* si elle préserve l'ordre.

$$f \text{ croissante} \Leftrightarrow \forall I, J \text{ image}, I \leq J \Rightarrow f(I) \leq f(J). \quad (1.12)$$

Une dernière propriété des opérations sur les ensemble dont il est utile de parler est la conservation ou non de l'homotopie. En simplifiant on peut dire que deux objets sont homotopes si on peut passer de l'un à l'autre par déformation continue, c'est à dire sans créer de particule et sans combler de trous. Par exemple une tasse (à une anse) et un tore sont homotopes.

1.2 Transformations morphologiques

Le formalisme orthodoxe pour décrire les opérations morphologiques et la morphologie mathématique en général est le cadre de la théorie des treillis complets, c'est-à-dire des ensembles partiellement ordonnés pour lesquels toute famille d'éléments admet une borne sup et une borne inf. Pour ne pas alourdir ce chapitre nous nous contenterons d'évoquer ce cadre, au risque de perdre beaucoup en rigueur. Les lecteurs exigeants pourront se reporter aux ouvrages de base cités en introduction.

Conformément à ce que permet ce cadre, cependant, dans tout ce qui va suivre on évitera de faire la distinction entre les images binaires et les images à niveaux de gris, le premier étant un cas particulier du second.

1.2.1 Erosion et dilatation

1.2.1.1 Éléments structurants

Dans le cadre des treillis complets, on appelle *érosion* toute opération qui commute avec l'inf, et *dilatation* toute opération qui commute avec le sup. Pour chaque érosion on peut définir une unique dilatation dite *duale*, et réciproquement. En pratique pour définir la notion d'érosion et celle de dilatation on utilise ce qu'on appelle un *élément structurant* (ÉS), c'est à dire un ensemble connu B pour lequel on définit une *origine*, c'est à dire simplement un point particulier.

Un éléments structurants couramment utilisé pour les images discrètes est l'élément isotrope au sens de la trame de taille unité (noté D) :

$$D = \{p \in \mathbb{Z}^2, d_G(0, p) \leq 1\} \quad (1.13)$$

Où 0 est l'origine du plan (et celle de l'ÉS). Cet S est la *boule unit* dans la trame considre.

En trame hexagonale, D est l'hexagone élémentaire, alors qu'en trame carrée, D est respectivement une croix en 4-connexité et un carré unité en 8-connexité (voir figure 1.9).

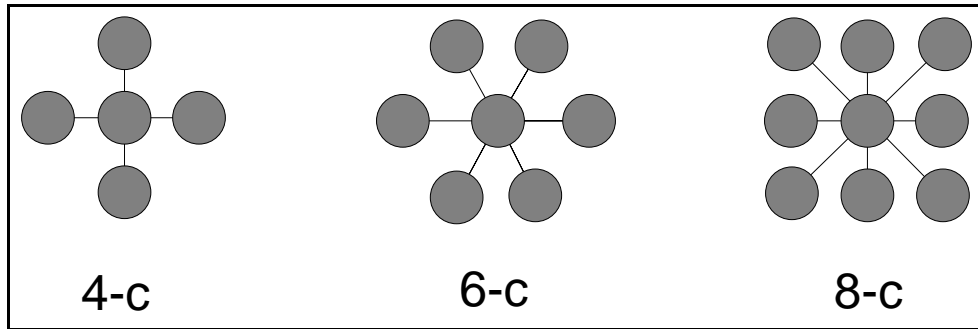


FIG. 1.9: ES élémentaires pour les trames standard (carrée 4-connexte, hexagonale 6-connexte, carrée 8-connexte).

De même l'élément structurant isotrope (toujours au sens de la trame) de *taille* n est noté nD et est défini par :

$$nD = \{p \in \mathbb{Z}^2, d_G(0, p) \leq n\} \quad (1.14)$$

En trame carrée et en 8-connextité, $3D$ est le carré centré de côté $3 \times 2 + 1 = 7$ pixels, par exemple (voir figure 1.10).

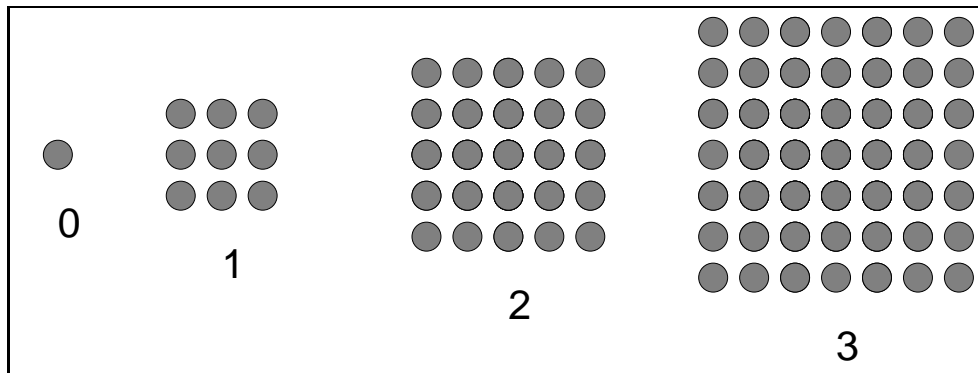


FIG. 1.10: ÉS de taille croissante pour la trame carrée (tailles 0 (identité), 1, 2 et 3) en 8-connextité.

On peut définir une infinité d'éléments structurants. Dans le cas des images à niveaux de gris, ceux-ci ne sont pas forcément binaires, on parle alors parfois de *fonction structurante* (voir figure 1.11), ou d'*élément structurant 3-D*. Cependant, en considérant leur sous-graphe, comme dans le cas d'une image, ils se traitent théoriquement comme dans le cas binaire en considérant les opérations sur les ensembles.

1.2.1.2 Définitions

Lorsqu'on considère un ensemble X et un élément structurant B , la dilatation $\delta_B(X)$ de X par B est l'union des points x tels que B_x (B translaté en x) intersecte X :

$$\delta_B(X) = \{x, B_x \cap X \neq \emptyset\} \quad (1.15)$$

l'érosion $\epsilon_B(X)$ de X par B est l'ensemble des points x tels que B soit totalement inclus dans X si B est centré en x .

$$\epsilon_B(X) = \{x, B_x \subset X\} \quad (1.16)$$

La figure 1.11 présente un exemple de dilatation d'un signal monodimensionnel par une fonction structurante selon cette définition.

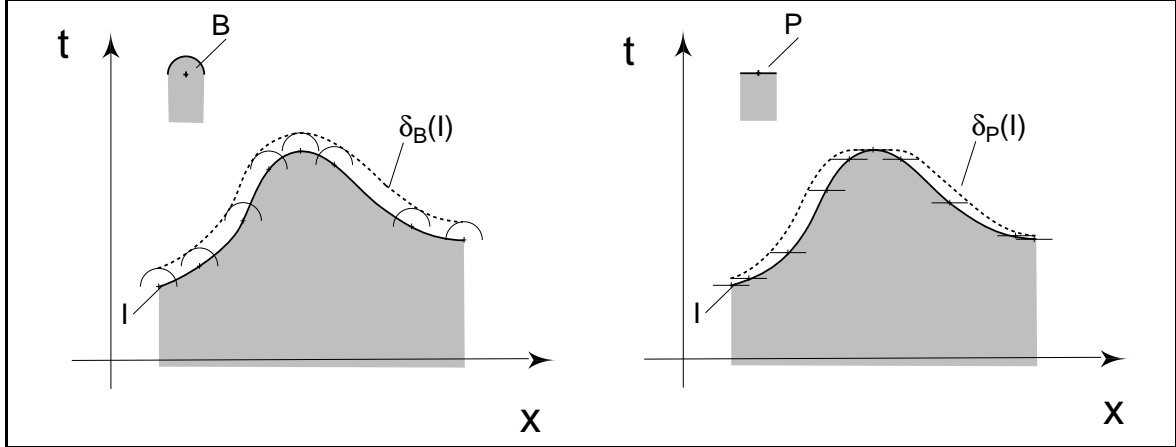


FIG. 1.11: Dilatation d'un signal par un élément structurant non plan (fonction structurante) et un élément structurant plan. Dilater par un ÉS plan revient à prendre le maximum de la fonction sur la fenêtre définie par l'ÉS.

Nous allons un peu simplifier cette expression. Supposons tout d'abord que nous ne nous intéressons qu'aux éléments structurants dits *plans*, c'est à dire ceux valables pour les images binaires, et que nous laissons de côté les fonctions structurantes générales¹. On peut alors définir la dilatation d'une image I par B en un point x comme la valeur maximale de l'image dans la fenêtre constituée par B lorsqu'elle est située en x (voir encore la figure 1.11).

$$\delta_B(x) = \max\{x_k, k \in B\} \quad (1.17)$$

Dans le même cadre, on définit alors l'érosion :

$$\epsilon_B(x) = \min\{x_k, k \in B\} \quad (1.18)$$

Les opérations de dilatation et d'érosion sont définies de façon duales l'une de l'autre vis à vis de l'opération de complémentarité.

$$\epsilon_B(I) = [\delta_B(I^c)]^c \quad (1.19)$$

La figure 1.12 présente des exemples de dilatation et d'érosion par des carrés dans les cas binaires et numériques.

On note souvent l'érosion et la dilatation à partir des soustractions et des additions de Minkowski, ainsi :

$$\delta_B(I) = I \oplus \check{B} \quad (1.20)$$

$$\epsilon_B(I) = I \ominus \check{B} \quad (1.21)$$

¹En général l'intensité d'un pixel de l'image et sa position spatiale ne sont pas définis par les mêmes unités ni les mêmes dimensions.

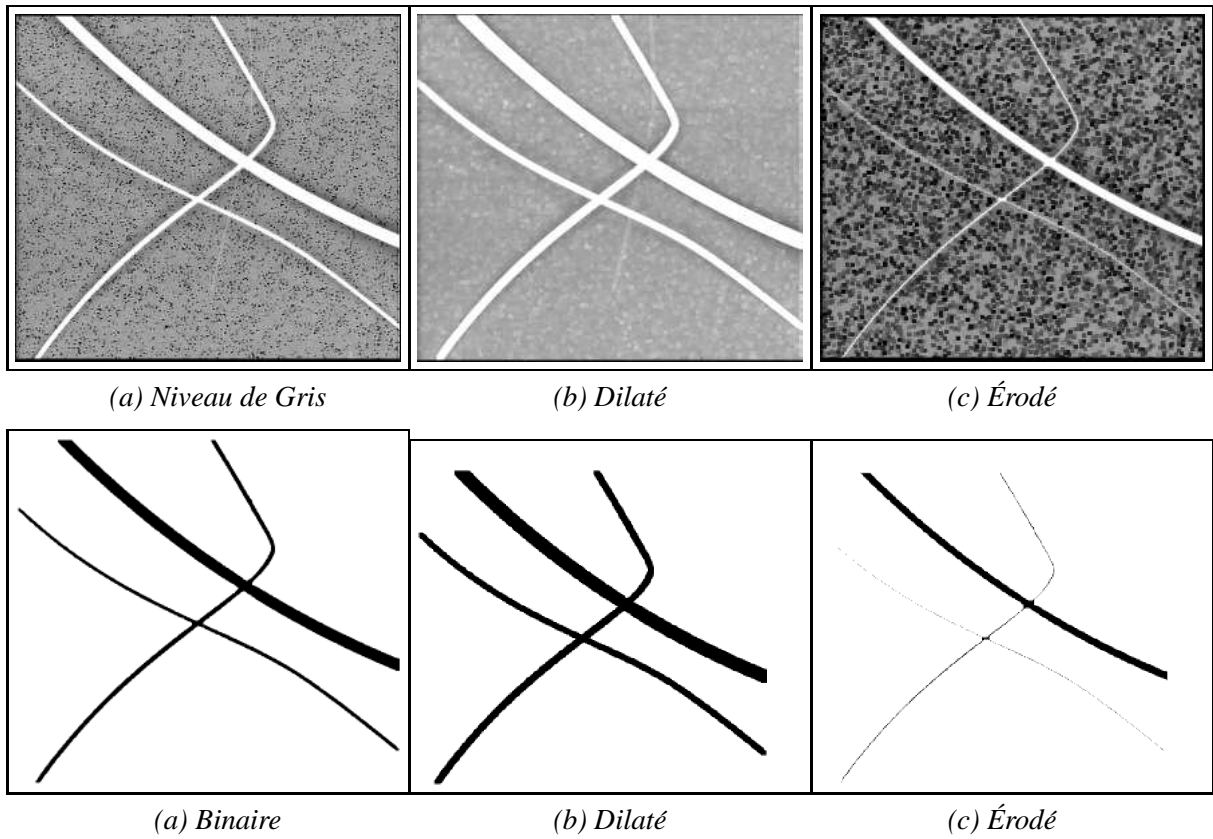


FIG. 1.12: Dilatations et érosions en niveaux de gris et en binaire par un élément structurant carré symétrique.

Où \check{B} est le transposé de B , c'est à dire le symétrique de B par rapport à l'origine :

$$\check{B} = \{x, -x \in B\} \quad (1.22)$$

Lorsque l'élément structurant est symétrique (la plupart du temps) l'écriture du \check{B} devient inutile.

Il y a beaucoup à dire sur les propriétés de l'érosions et de la dilatation. Nous nous contenterons de quelques unes :

- Erosion et dilatation sont des transformations croissantes
- La dilatation (par exemple) par la boule de taille n peut être obtenue par n itérations de dilations par la boule de taille unité. En général :

$$\delta_{B_1}[\delta_{B_2}](I) = \delta_{\delta_{B_1}(B_2)}(I) \quad (1.23)$$

$$\epsilon_{B_1}[\epsilon_{B_2}](I) = \epsilon_{\delta_{B_1}(B_2)}(I) \quad (1.24)$$

- Il n'y a pas d'opération inverse ni de la dilatation, ni de l'érosion.

1.2.1.3 Une application : les gradients

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

$$\Gamma(I) = \delta_B(I) - \epsilon_B(I) \quad (1.25)$$

et les deux variantes

$$\Gamma^+(I) = \delta_B(I) - I \quad (1.26)$$

$$\Gamma^-(I) = I - \epsilon_B(I) \quad (1.27)$$

Appelés *gradient externe* et *gradient interne*. B est la boule unité de la trame considérée. On verra un exemple de gradient à propos de la ligne de partage des eaux.

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.

1.2.2 Transformations en tout ou rien

On considère les transformations qui tiennent compte à la fois des points qui appartiennent à l'ensemble et des points qui n'y appartiennent pas. On a donc 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 :

$$X \star T = (X \ominus T_1) \cap (X^c \ominus T_2) \quad (1.28)$$

On appelle ces opérations *transformations en tout ou rien* ou *HMT* (hit or miss transforms).

Deux types de transformations essentiellement font appel aux transformées en tout ou rien, ce sont les *amincissements* et les *épaississements*. Dans la première transformation on retranche de l'ensemble de départ le résultat de la transformée en tout ou rien, dans le second cas on l'ajoute.

Ces transformations peuvent par exemple servir à calculer des squelettes, à détecter les points isolés ou les points multiples, etc.

1.2.3 Transformées géodésiques

La *distance géodésique* $d_R(x, y)$ d'un point x à un point y à l'intérieur d'un ensemble R est le plus petit chemin de x à y restant à l'intérieur de R .

$$d_R(x, y) = \min\{l(C_{xy}), C_{xy} \in R\} \quad (1.29)$$

Cette distance vaut par convention $+\infty$ si aucun chemin n'existe entre x et y à l'intérieur de R .

L'ensemble R est appelé le *masque*. On va pouvoir introduire des transformations morphologiques *géodésiques*, en considérant uniquement la partie de l'élément structurant qui reste à l'intérieur du masque. Par exemple la boule géodésique isotrope (selon la trame) de rayon r est :

$$B_R(x, r) = \{y \in R, d_R(x, y) \leq r\}$$

1.2.3.1 Transformées élémentaires géodésiques

En mode géodésique strict, le seul structurant que l'on puisse considérer est la boule géodésique isotrope. La dilatation et l'érosion géodésiques d'un ensemble X inclus dans un masque R par la boule géodésique de taille 1 s'écrivent alors :

$$\delta_R^{(1)}(X) = \{y \in R, B_R(y, 1) \cap X \neq \emptyset\} \quad (1.30)$$

$$\epsilon_R^{(1)}(X) = \{y \in R, B_R(y, 1) \subset X\} \quad (1.31)$$

Le résultat de l'opération est donc toujours compris dans le masque. On peut noter par exemple la dilatation par :

$$\delta_R^{(1)}(X) = (X \oplus D) \cap R \quad (1.32)$$

Où D est l'élément structurant isotrope de base de la trame (§ 1.2.1).

En termes de min et max, la dilatation géodésique s'exprime par :

$$\delta_R^{(1)}(X) = \min\{\delta_D(X), R\} \quad (1.33)$$

Les dilatations successives sont obtenues par itérations de la dilatation de taille 1 :

$$\delta_R^{(n)}(X) = \underbrace{\delta_R^{(1)}(\delta_R^{(1)}(\dots\delta_R^{(1)}(X)))}_{n \text{ fois}} \quad (1.34)$$

La figure 1.13 présente un exemple de dilatations géodésiques successives dans les cas binaires et numériques.

L'érosion et la dilatation géodésique sont duales par complémentation :

$$\epsilon_R^{(n)}(R \setminus X) = R \setminus \delta_R^{(n)}(X) \quad (1.35)$$

1.2.3.2 Reconstruction

Un des premiers intérêts de la dilatation géodésique est de permettre l'opération de *reconstruction*. À 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 1.14 un tel exemple de transformation.

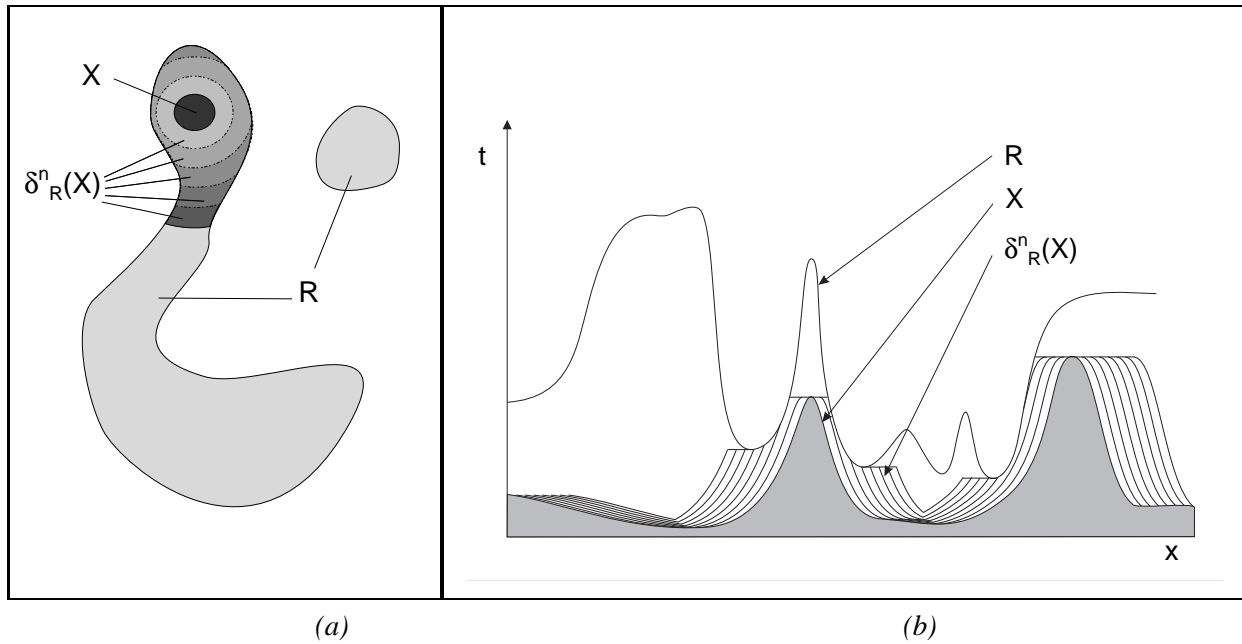


FIG. 1.13: Dilatation géodésique. *a* : cas binaire, *b* cas numérique (élément structurant plan).

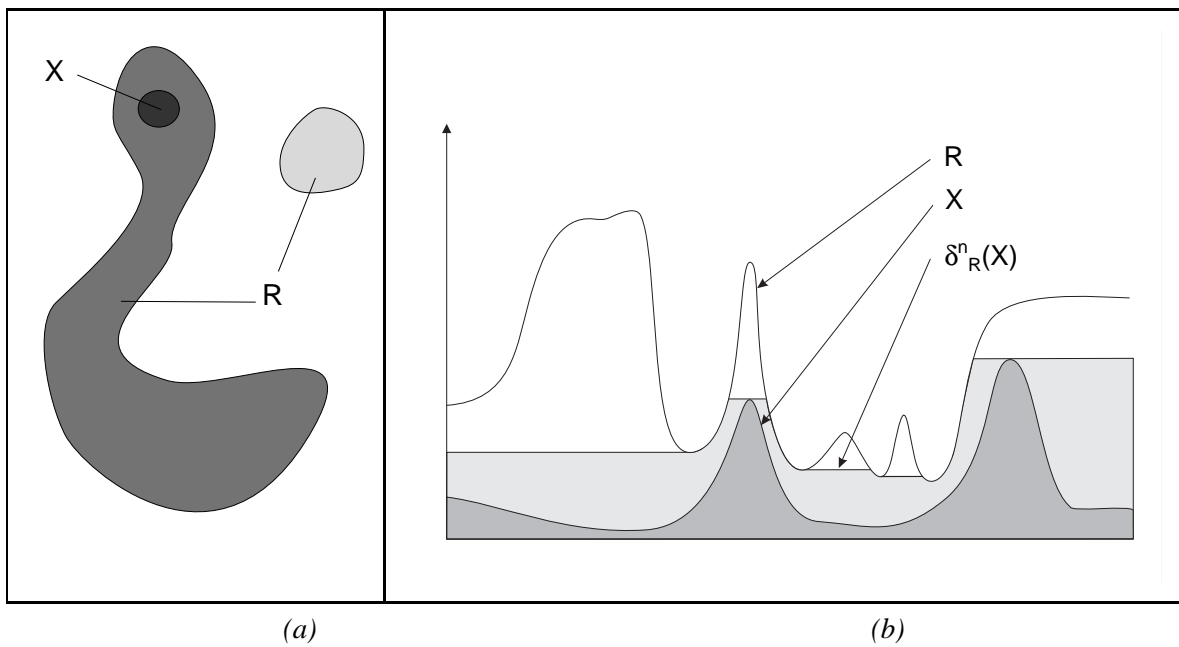


FIG. 1.14: Reconstruction. On retrouve la forme des ensembles initiaux en *a*. En *b* on ne reconstruit pas tout car on a utilisé un ÉS plan.

1.2.4 Ouvertures et Fermetures

Dans le cadre de la morphologie mathématique sur treillis complets, une ouverture est une opération croissante, anti-extensive, et idempotente. Parmi celles-ci on appelle *ouvertures morphologiques* γ_B celles que l'on peut définir à l'aide d'un élément structurant B . On appelle *fermeture* ϕ_B l'opération duale. Plus précisément :

$$\gamma_B X = \delta_{\tilde{B}}(\epsilon_B(X)) \quad (1.36)$$

$$\phi_B X = \epsilon_{\tilde{B}}(\delta_B(X)) \quad (1.37)$$

la figure 1.15 illustre l'effet des ouvertures et des fermetures dans les cas numérique et binaire.

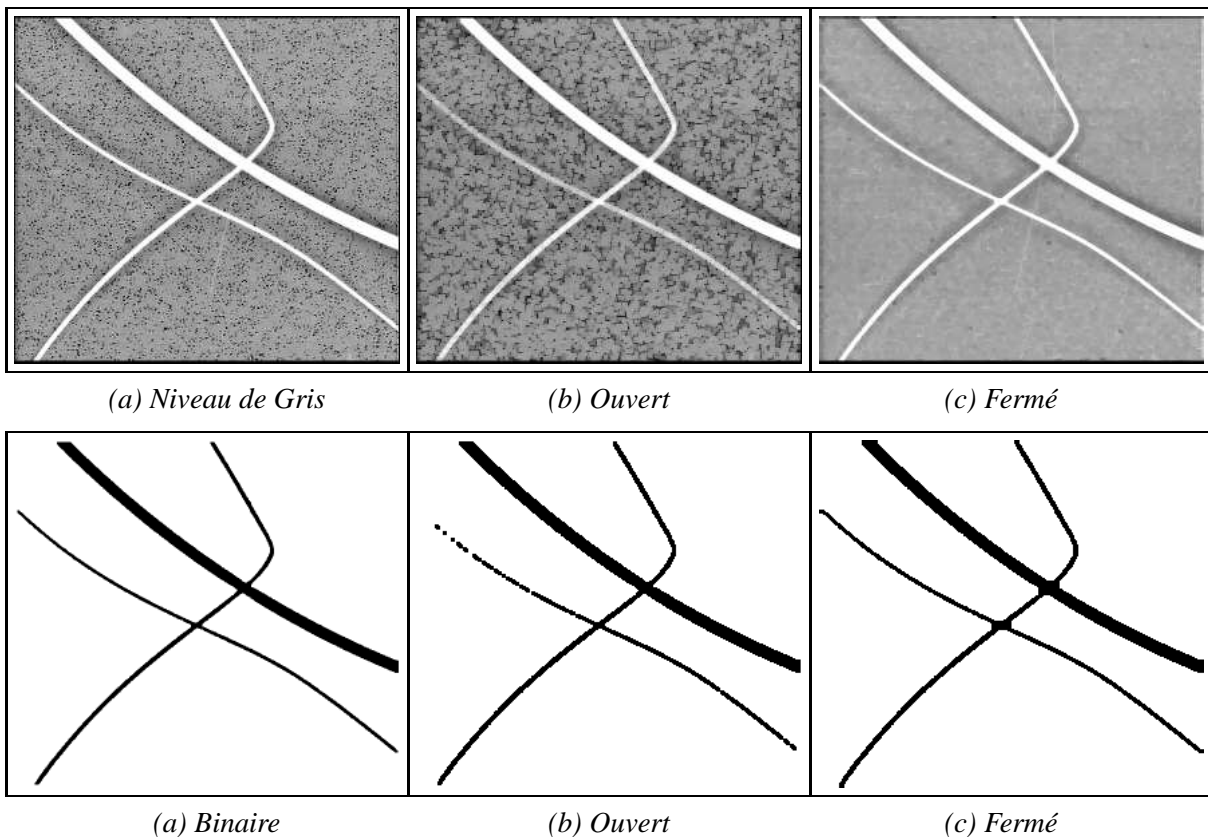


FIG. 1.15: Fermetures et ouvertures en niveaux de gris et en binaire (ÉS carré de taille 2.)

On vérifie sur ces images que l'ouverture et la fermeture ne donnent en effet pas le même résultat. On le répète : ces opérations n'ont pas d'inverse. On voit sur ces figures que l'opération duale γ rattrape un peu l'effet de l'opération initiale (par rapport à une érosion ou une dilatation simple), mais que certains détails sont perdus. En particulier l'ouverture fait disparaître les pics (selon la taille de l'ÉS) et la fermeture comble les vallées. On peut utiliser ces opérations pour simplifier les images sur lesquelles on travaille ou pour permettre de détecter les petits détails. On constate sur ces exemples très simples que la forme de l'élément structurant est visible.

Une ouverture (fermeture) très utilisée est l'ouverture (la fermeture) *par reconstruction*. On fait suivre une érosion (une dilatation) d'une reconstruction (reconstruction duale). On vérifie facilement

que le résultat est une ouverture (fermeture). Son intérêt est d'être plus adaptative que l'ouverture ou la fermeture simple. En particulier on y reconnaît beaucoup moins la forme de l'élément structurant.

Les opérations qui font ainsi appel à la reconstruction font appel à une importante famille d'opérateurs dits *connexes* [?, ?].

Un tel exemple d'ouverture connexe que nous aurons l'occasion d'utiliser un peu plus loin est l'ouverture *surfactive*, proposée par Matheron [?, pp.196-200] et récemment mis en œuvre par Vincent [?]. L'ouverture surfactive d'une image I est l'image J dont tous les maxima ont une surface projetée plus grande qu'un paramètre a . On peut aussi la définir comme le sup des ouvertures par tous les éléments structurants de surface a donnée. La figure 1.16 présente un exemple de *fermeture* surfactive (l'opération duale avec une condition sur les minima).

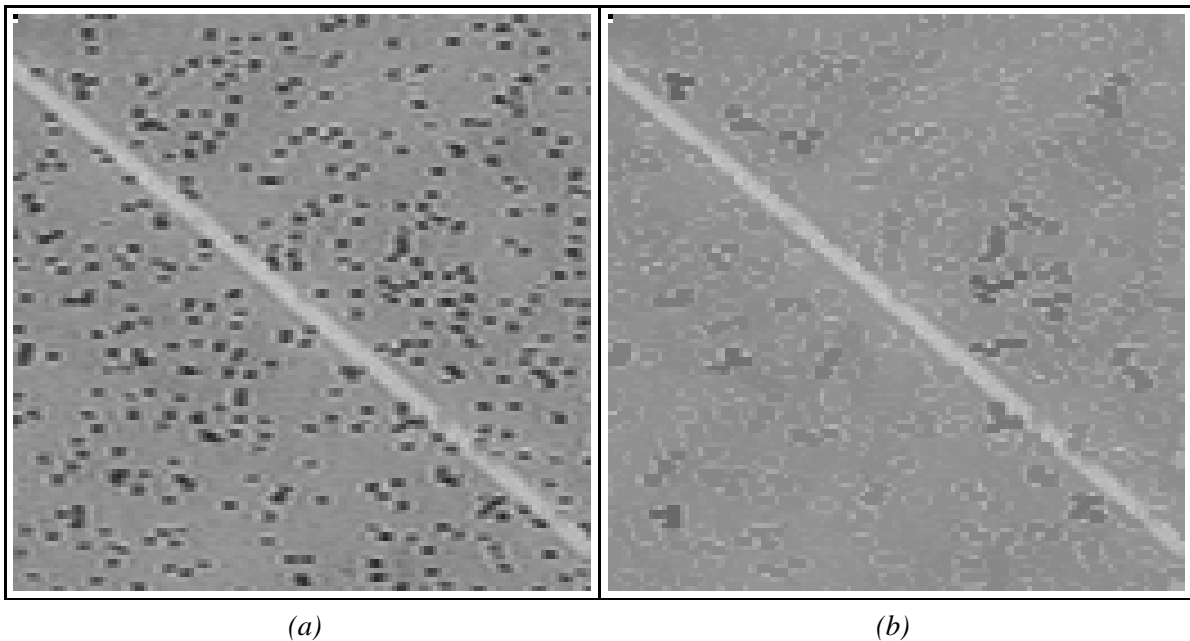


FIG. 1.16: Fermeture surfactive de taille 20 (pixels carrés). Les minima de petite taille ont été comblés de façon adaptative. Les pics de l'image n'ont pas été affectés (par exemple, la fibre).

1.2.4.1 Les transformations chapeaux hauts-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 1.17 illustre le cas des chapeaux hauts-de-forme blancs.

Il existe autant de façons d'obtenir des transformations chapeaux hauts-de-forme que de types d'ouvertures. D'une manière générale, on appelle les transformations qui font appel à des opérations morphologiques combinées avec des différences des *résidus*. Les chapeaux hauts-de-forme, les squelettes, les gradients et quelques autres transformations sont des résidus.

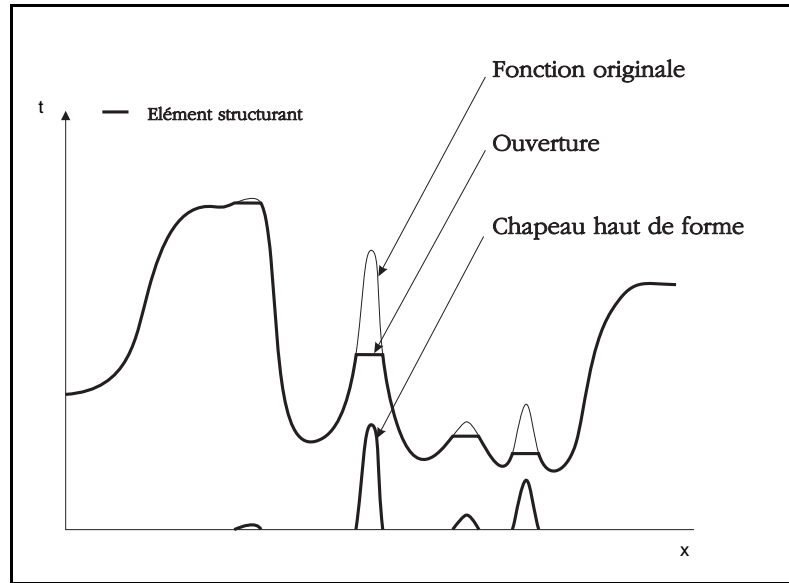


FIG. 1.17: Top-hat sur un signal 1-D.

1.3 Filtrage

On remarquera que l'on a, tout au long de cet exposé, évité le terme de \mathcal{F} filtre \mathcal{F} pour décrire l'une ou l'autre des opérations présentées. Ce n'est pas par hasard. En effet, si en traitement du signal classique on désigne par \mathcal{F} filtre \mathcal{F} à peu près toutes les types de traitements, en morphologie mathématique, ce terme a une signification bien précise : Un filtre morphologique dans le cadre d'un treillis complet T est un opérateur *croissant* et *idempotent* [?]. Si \mathcal{F} est un tel opérateur, alors

$$\forall B, B' \in T, B \subset B' \longleftarrow \mathcal{F}(B) \subset \mathcal{F}(B') \quad (1.38)$$

et

$$\forall B \in T, \mathcal{F}(\mathcal{F}(B)) = \mathcal{F}(B) \quad (1.39)$$

Ni une érosion ni une dilatation ne sont donc des filtres morphologiques en général, puisque ces opérations ne sont pas idempotentes (sauf en utilisant un point pour élément structurant, ce qui donne l'application identité). Par contre les ouvertures et les fermetures sont des filtres morphologiques, et d'une manière générale, les compositions de fermetures et d'ouvertures sont également des filtres morphologiques.

On a déjà fait la remarque que les opérations de base de la morphologie mathématique considéraient de façon différente les blancs et les noirs des images à niveau de gris. Or, pour nombre d'applications, il peut être nécessaire de traiter ceux-ci de la manière la plus égalitaire possible. D'autre part on peut vouloir réaliser une même opération pour différentes tailles d'objets, ou plus généralement indépendamment du contenu de l'image. Dans un cas comme dans l'autre il faut réaliser des compositions d'opérations morphologiques.

La théorie du filtrage morphologique est là pour nous aider à réaliser ces compositions 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, réalisés avec des éléments structurants acceptables pour une granulométrie (c'est à dire convexes, que

l'on peut déduire les uns des autres par des homothéties et de taille croissante indexés par n : exemple : la famille des ÉS carrés).

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

$$\Phi_n(x_i) = \phi_n \gamma_n \phi_{n-1} \gamma_{n-1} \dots \phi_1 \gamma_1 \quad (1.40)$$

et le FAS noir par

$$\Psi_n(x_i) = \gamma_n \phi_n \gamma_{n-1} \phi_{n-1} \dots \gamma_1 \phi_1 \quad (1.41)$$

La théorie des filtres morphologiques est assez longue et ne saurait tenir dans ces quelques pages. Par conséquent nous nous contentons de montrer quelques exemples d'application élémentaires (figure 1.18).

Les filtres alternés séquentiels peuvent donc servir à diminuer l'importance du bruit d'origine aléatoire dans les images aussi bien binaires qu'à niveau de gris, facilitant ainsi une étape ultérieure de segmentation. D'autre part, les filtres alternés séquentiels étant des compositions d'opérations à base d'ÉS, on peut, en jouant sur ces ÉS, amplifier, préserver ou faire disparaître les formes que l'on souhaite, dans les limites du choix des ÉS utilisables. Généralement, on ne module pas les opérations que l'on souhaite opérer sur ses 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 $\ddot{\text{}}$ sur mesure $\ddot{\text{}}$ en fonction du *contenu* de l'image.

1.4 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*. Imaginons une goutte d'eau tombant sur le relief crée par l'image. Celle-ci va s'écouler jusqu'à ce qu'elle atteigne un minimum régional. À 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 1.19).

En pratique on calcule la ligne de partage des eaux non en partant des endroits où tombent des gouttes d'eau, mais en $\ddot{\text{}}$ inondant $\ddot{\text{}}$ l'image en partant des minima de celle-ci. La ligne de partage des eaux est l'ensemble des points où au moins deux nappes d'eau se rencontrent.

La ligne de partage des eaux (LPE) est à la base d'un algorithme de segmentation puissant et souple proposé par Beucher et Lantuéjoul [?, ?]. 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, seule 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 inonde ensuite ce gradient à partir de tous les marqueurs (et non à partir des minima $\ddot{\text{}}$ naturels $\ddot{\text{}}$ 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.

On reprend ici une partie de l'exemple très simple de la thèse de Gratin (image originale : image d'un œuf en RMN, *source Neils Roberts, University of Liverpool*) sur la figure 1.20 :

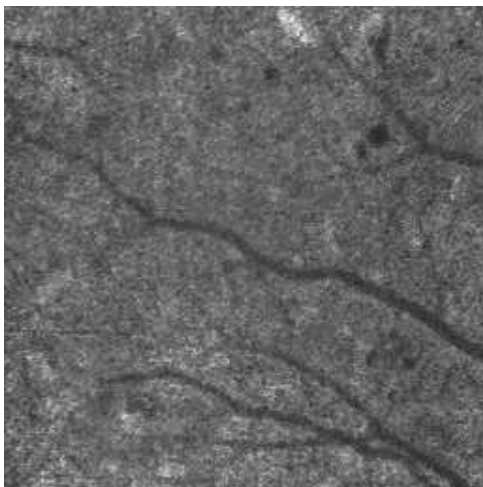
Sur cet exemple les marqueurs du jaune, du blanc et de l'extérieur de l'œuf sont donnés à la main. Il est possible (et Gratin en donne un exemple) de rechercher des marqueurs automatiques de ces



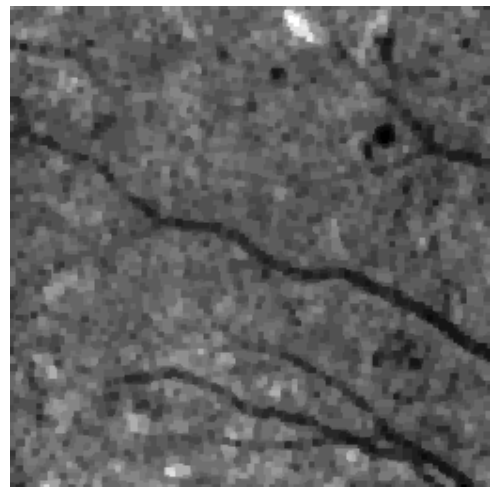
(a)



(b)



(c)



(d)

FIG. 1.18: Utilisation des filtres alternés séquentiels. (a) Image binaire, (b) FAS blanc taille 2, (c) Fond d'œil, (d) FAS noir taille 1

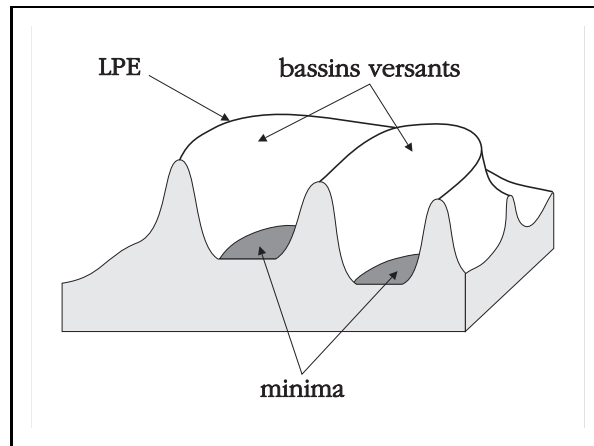
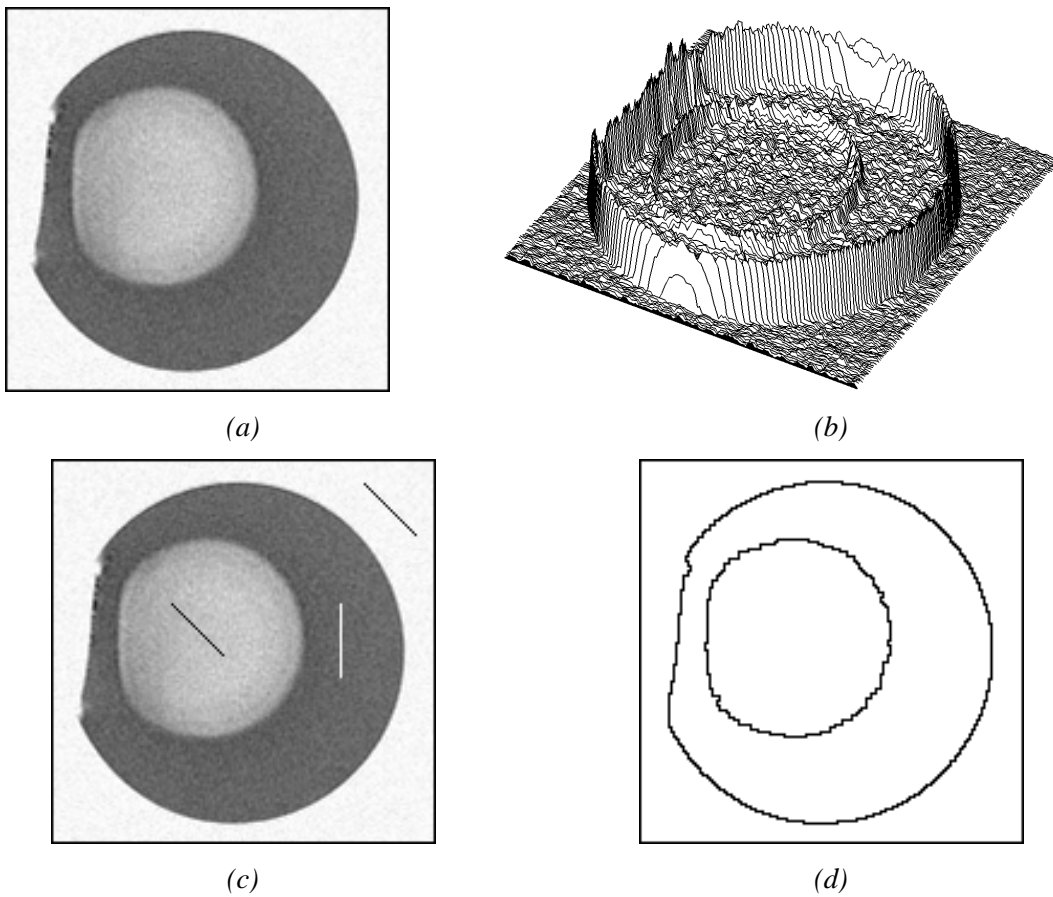


FIG. 1.19: La ligne de partage des eaux.

FIG. 1.20: 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.

parties. Contrairement à l'apparence très simple de l'image, un simple seuillage ne donne pas de bons résultats, alors que segmentation par LPE donne un résultat parfait. L'essentiel ici étant de montrer le déroulement de la méthodologie de segmentation par LPE, on n'insistera pas davantage.

1.5 Mise en œuvre informatique

La mise en œuvre informatique des algorithmes de MM est un sujet passionnant, et nous en parlerons un peu dans les deux chapitres qui suivent, mais il ne nous est pas possible de présenter l'ensemble des méthodes informatiques qui permettent de mettre en œuvre toutes ces transformations. Le nombre de cas particuliers est vraiment trop important.

Il est toutefois assez utile d'exposer en quelques mots quelles sont actuellement les méthodes générales de mise en œuvre, et quelles sont leurs particularités.

Le monde des algorithmes de morphologie mathématique est grossièrement séparé en trois catégories :

1. Les algorithmes parallèles. Ces algorithmes opèrent d'une image d'entrée vers une image de sortie différente et fonctionnent de la manière suivante : on balaye les pixels de l'image d'entrée et on calcule la nouvelle valeur du point considéré en fonction de la valeur du pixel courant et de celle des pixels dans son voisinage. L'ordre de traitement n'importe pas. On pourrait ainsi imaginer de traiter en parallèle (d'où leur nom) tous les pixels de l'image d'entrée. Ces algorithmes sont mis en œuvre de façon efficace dans les machines de traitement d'image spécialisées [?], mais sont généralement lents sur les stations de travail courantes.
2. Les algorithmes séquentiels. Ceux-ci font aussi intervenir un ou plusieurs balayages de l'image d'entrée, mais le résultat du traitement est placé dans cette même image, pour être exploité lors du déplacement immédiatement ultérieur du balayage. L'ordre des balayages de l'image est essentiel pour ces algorithmes. En général ces algorithmes sont bien plus rapides sur station de travail courante que les algorithmes parallèles. Certaines machines d'analyse d'images spécialisées commencent à faire appel à ces techniques, en particulier celles utilisant le processeur PIMM1 de LSI-LOGIC, développé à l'Ecole des Mines [?]. On considère généralement que les algorithmes séquentiels sont un ordre de grandeur plus rapide que leur équivalent parallèle.
3. Les algorithmes ordonnés. Cette dernière catégorie est la plus récemment développée. Les méthodes ordonnées tirent avantage de la capacité des ordinateurs non spécialisés à accéder n'importe quelle partie de la mémoire facilement, et non seulement à partir de balayages. On compte parmi les algorithmes ordonnés :
 - Ceux à base de chaînes et de lacets. Initiés par M. Schmitt [?] et L. [?]. Fonctionnent à partir des contours des objets (binaires) et sur des triangulations (donc trame hexagonale uniquement, en 2-D).
 - Ceux à base de files d'attente simples (FA). Proposés par L. Vincent (voir sa thèse), ces algorithmes fonctionnent essentiellement pour les images binaires (il existe des exceptions : la reconstruction numérique, la LPE, par exemple), à partir du contour des objets, et sont adaptable à n'importe quelle trame.
 - Ceux à base de Files d'Attente Hiérarchiques (FAH). Proposés par F. Meyer [?], ces algorithmes étendent le concept des FA au cas numérique. Ceux-ci fonctionnent pour n'importe quelle trame également. L'idée de base derrière les FAH est de fournir un tris permanent des pixels qui sont entrés dans cette structure.

- Ceux à base de tas (*heap*). Proposés très récemment par L. Vincent [?]. Ces structures permettent de corriger certains défauts des FAH, en particulier le fait que l'on ne peut traiter qu'un faible nombre de niveaux de gris en même temps avec les FAH. Les tas n'ont pas ce genre de limitations (on peut ainsi travailler avec eux sur des images en flottant, et non nécessairement en entier). On paye cette souplesse par des temps de calcul un peu plus longs (en général, cet accroissement de temps de calcul n'est pas sensible). Dans le cadre de cette thèse, une contribution importante à l'élaboration de ces algorithmes a été apportée. C'est pourquoi nous donnons quelques détails supplémentaires : les tas s'utilisent globalement de la même manière que les FAH : on dispose de 4 fonctionnalités :
 - (a) `HEAP-init` Créer une structure de tas (argument : une taille). Cette taille peut augmenter au fur et à mesure de l'utilisation du tas, mais cet aspect des choses est normalement invisible de l'utilisateur.
 - (b) `HEAP-free` Libérer une structure de tas. A la fin de son utilisation, on libère toute la taille allouée au tas.
 - (c) `HEAP-put` Ajouter un point (pointeur de pixel) au tas (argument : le pixel en question).
 - (d) `HEAP-get` Enlever un point du tas. Pas d'argument, mais cette fonction retourne le premier pointeur entré dans le tas avec le niveau de gris (priorité) le plus faible.

La structure interne des tas est celle d'un arbre binaire complet, mis en œuvre de façon particulière (voir [?]). Un exemple un algorithme de calcul de la fonction distance euclidienne utilisant les tas est présent maintenant.

Algorithme : Fonction distance euclidienne par Tas

- **Données**
 - *imIn*, image d'entrée binaire.
 - *imOut*, image de sortie.
 - *imX*, image coordonnée *X* des vecteurs de propagation.
 - *imY*, image coordonnée *Y* des vecteurs de propagation.
- **Variables**
 - \vec{Ox}, \vec{Oy} , vecteurs unitaires (abscisse et ordonnée respectivement)
 - *vx*, *vy*, variables entières
 - *dist*, *oldist*, variables entières
 - *p*, *p'*, pixels
- **Initialisation**

```

HEAP-init(imIn.sizeX * imIn.sizeY);
Pour tout pixel p de imIn
  imOut[p] = imIn[p]; /* copie de in dans out */
  imX[p] = imY[p] = +∞;
  Si imIn[p] = 1 et si  $\exists p'$ , voisin de p, imIn[p'] = 0 alors
    HEAP-add(p);

```



```

     $imX[p] = \vec{p'p} \cdot \vec{Ox}; imY[p] = \vec{p'p} \cdot \vec{Oy};$ 
  FinSi
FinPour
• Boucle principale
  TantQue ( $p \leftarrow HEAP-get()$ ) n'est pas NUL
  Pour tout  $p'$  voisin de  $p$ 
  Si ( $imOut[p'] = 1$ )
     $imOut[p] = \sqrt{imX[p]^2 + imY[p]^2} + 1;$ 
  Pour tous les voisins  $p'$  de  $p$ 
    Si  $imOut[p] = 1$ 
       $oldist = imX[p']^2 + imY[p']^2$ 
       $vx = imX[p] + \vec{pp'} \cdot \vec{Ox}; vy = imY[p] + \vec{pp'} \cdot \vec{Oy};$ 
       $newdist = vx^2 + vy^2;$ 
      Si ( $newdist < oldist$ ) alors
         $imOut[p'] = newdist + 1;$ 
         $HEAP-add(p');$ 
        /* on affecte réellement imOut APRÈS la propagation */;
         $imOut[p'] = 1;$ 
         $imX[p'] = vx; imY[p'] = vy;$ 
      FinSi
    FinSi
  FinPour
FinSi
FinPour
FinSi
FinPour
FinTantQue
/* Étape optionnelle */
Pour tout pixel  $p$  de  $imOut$ ;
  Si  $imOut[p] > 0$  alors  $imOut[p] \leftarrow imOut[p] - 1;$ 
FinPour
 $HEAP-free();$ 

```

Cet algorithme peut calculer la fonction distance euclidienne de tous les ensembles binaires quelque soit leur taille, à la différence de celui de Gratin. L'algorithme de Gratin [?] utilise en effet une FAH, structure qui ne permet pas de traiter un grand nombre de niveau de gris en même temps (typiquement 256 au maximum), or l'établissement de la fonction distance pour un ensemble binaire suffisamment grand peut réclamer beaucoup de ces niveaux². Celui de Gratin et le nôtre possèdent cependant les mêmes qualités ceci mis à part, en particulier au niveau temps de calcul. Voici une comparaison des temps de calculs obtenus en comparaison avec l'algorithme de Danielsson [?], la référence en ce domaine.

Ces temps de calcul s'entendent sur une station de travail NeXT 68040/25MHz (équivalent Sun SPARC I). On constate dans certains cas un gain de l'ordre de 10 par rapport à l'algorithme (séquentiel) de Danielsson. Pour les problèmes bien adaptés, c'est cet ordre de grandeur que l'on constate en faveur des algorithmes ordonnés.

²Ragnemalm dans [?] propose une méthode pour résoudre ce problème, mais qui ne donne pas entièrement satisfaction, des erreurs pouvant survenir pour des ensembles binaires assez grands

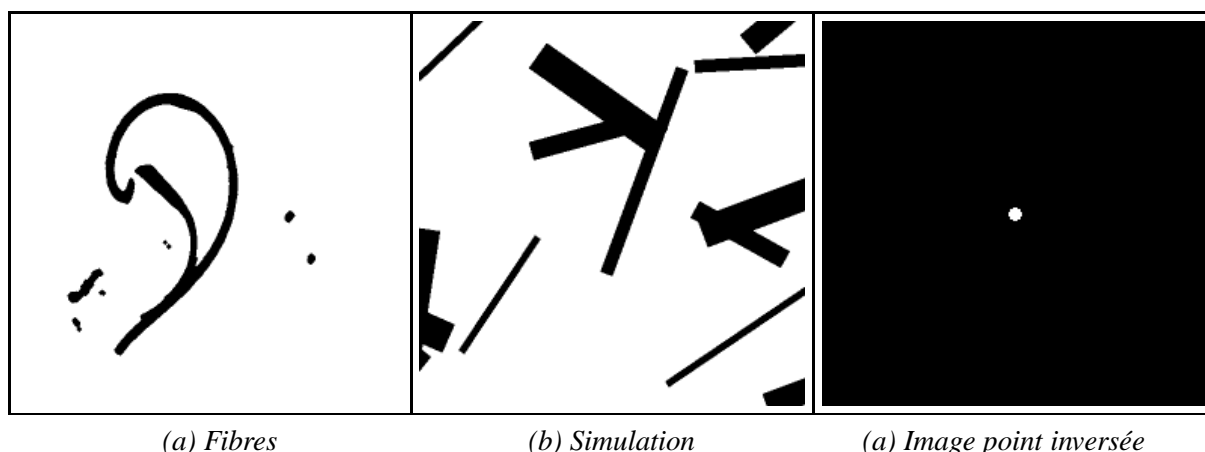


FIG. 1.21: Images ayant servi pour la comparaison des algorithmes de fonction distance euclidienne.

<i>Image</i>	<i>Danielsson</i>	<i>Le nôtre</i>
Fibres	22.3 s	1.7 s
Simulation	22.1 s	3.9 s
Point inversé	22.5 s	24.6 s

TAB. 1.1: Temps de calcul comparés pour l'algorithme de Danielsson et le nôtre

Conclusion

Dans ce chapitre nous avons évoqué un certain nombre d'opérateurs plus ou moins classiques en MM. Sans vouloir les résumer, on peut faire quelques remarques d'ordre général sur ceux-ci :

- Les outils morphologiques supposent la plupart du temps une connaissance *a priori* de ce qu'on recherche. À cela plusieurs raisons : la notion d'élément structurant permet d'indiquer ce qu'on recherche ou ce qu'on veut éliminer. D'autre part la plupart des opérations morphologiques n'ont pas d'inverse. On ne peut pas définir l'inverse d'une érosion dans le cas général, l'application d'une telle opération fait donc perdre de l'information à l'image. On a donc intérêt à savoir ce qu'on recherche afin de ne pas le faire disparaître. Ce n'est pas pour autant que l'on ne peut pas *améliorer* une image (même visuellement) en lui appliquant un traitement morphologique [?].
- Les opérateurs morphologiques vont généralement par paires : ceci est l'illustration de la dualité par rapport à la complémentation de l'image, qui est à la base même de la plupart des opérateurs de MM.
- Bien que nous nous proposons de traiter exclusivement des images bidimensionnelles à niveaux de gris dans le cadre de cette thèse, on ne doit pas oublier que la MM opère sur tout treillis complet. Des traitements morphologiques ont été réalisés sur des graphes [?, ?], sur des images à trois dimensions [?], de même que sur des images multispectrales (cas des images couleur ou des images satellites [?]).

Table des figures

1.1	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 .	5
1.2	Portions de trame carrée	5
1.3	Portion de trame hexagonale.	5
1.4	Propriété de Jordan digitale	6
1.5	Plusieurs chemins distincts peuvent être de longueur minimale dans une trame discrète.	7
1.6	Exemple d'images binaire et à niveaux de gris.	9
1.7	Sous-graphe d'un signal monodimensionnel et d'une image.	9
1.8	Sup et inf de deux images I et J	10
1.9	ES élémentaires pour les trames standard (carrée 4-connex, hexagonale 6-connex, carrée 8-connex).	11
1.10	ÉS de taille croissante pour la trame carrée (tailles 0 (identité), 1, 2 et 3) en 8-connexité.	11
1.11	Dilatation d'un signal par un élément structurant non plan (fonction structurante) et un élément structurant plan	12
1.12	Dilatations et érosions en niveaux de gris et en binaire par un élément structurant carré symétrique.	13
1.13	Dilatation géodésique	16
1.14	Reconstruction	16
1.15	Fermetures et ouvertures en niveaux de gris et en binaire (ÉS carré de taille 2.)	17
1.16	Fermeture surfacique	18
1.17	Top-hat sur un signal 1-D.	19
1.18	Utilisation des filtres alternés séquentiels. (a) Image binaire, (b) FAS blanc taille 2, (c) Fond d'œil, (d) FAS noir taille 1	21
1.19	La ligne de partage des eaux.	22
1.20	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.	22
1.21	Images ayant servi pour la comparaison des algorithmes de fonction distance euclidienne.	26

Image Analysis and Processing

Introduction

Sébastien Ourselin & Hugues Talbot

{Sebastien.Ourselin,Hugues.Talbot}@csiro.au.

CSIRO ICT Centre / Mathematical & Information Sciences
University of Sydney

1st Semester 2004

image processing, introduction – p.1/55

Content of the whole course

1. Theory
 - Linear and non-linear filtering
 - Transforms and enhancements
 - Segmentation and texture analysis
2. Applications
 - A medical application
 - A task automation application
3. Tutorials

image processing, introduction – p.2/55

Syllabus

1. **Introduction, image perception and representation**
2. Enhancements – Histogram & pixelwise transforms. (S)
3. Transforms – FFT, Laplace, Z, Hough. (H)
4. Filtering – Linear filters. (S)
5. Compression – Image coding. (Dan Popescu)
6. Texture analysis – Modelling, classification, segmentation. (H)
7. Geometry – Transforms, matching. (S)
8. Mathematical Morphology I – non-linear filtering. (H)
9. Mathematical Morphology II – distances, residues, HMT. (H)
10. Segmentation I – Thresholding, split & merge, snakes. (S)
11. Segmentation II – Watershed, SRG, recent PDE methods. (H)
12. Application I – Medical. (S)
13. Application II – Automation. (H)

Introduction to image processing/image analysis

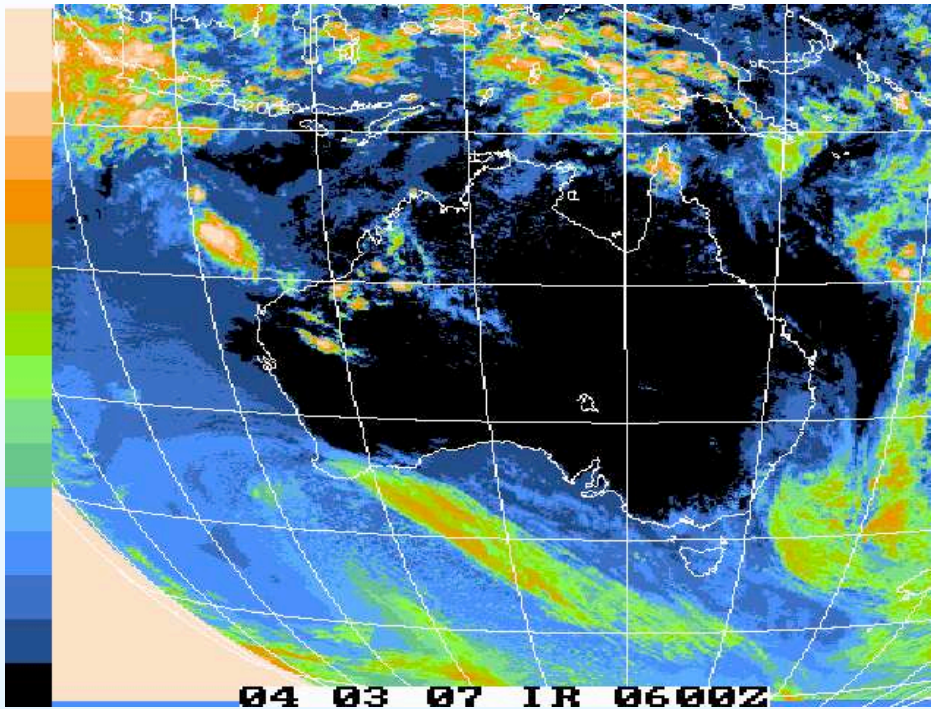
What is Image Analysis/Image Processing

- Two different things: Image Processing is the act of *modifying* an image.
- The goal of image processing is to make visible things that aren't.
- Image Analysis is the act of getting information out of images.
- The goal of image analysis is to interpret images.
- In the process of conducting image analysis, we will use many image processing techniques (e.g. denoising, feature extraction, etc).

Examples of image processings

- False colour rendering
- Deblurring
 - spatial
 - motion
 - optical sections
- Enhancements
 - Brightness/contrast (histogram equalization)
 - Denoising
 - Geometrical transforms (rotation, scaling, affine, etc)
- Coding (compression)
- Feature extraction (e.g. texture)

Example of false colour map



Source: Australian Bureau of Meteorology

image processing, introduction – p.7/55

Example of deblurring



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

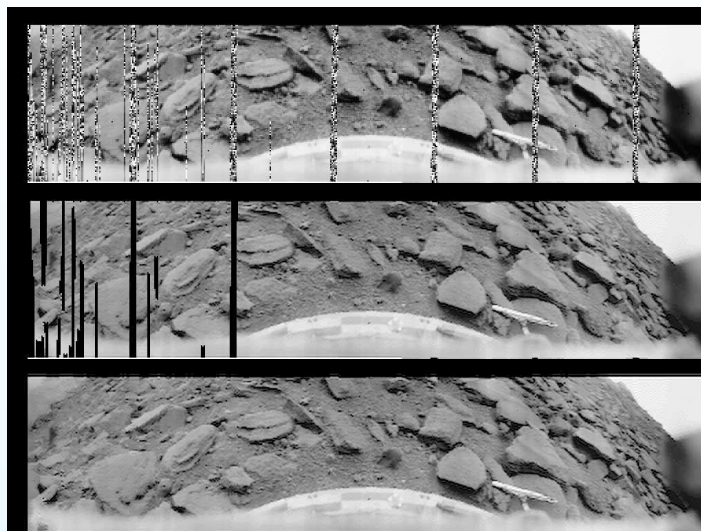
image processing, introduction – p.8/55

Example of deblurring



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

Example of advanced data interpolation



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

Source: http://www.mentallandscape.com/V_DigitalImages.htm

The Murphy principle works on Venus



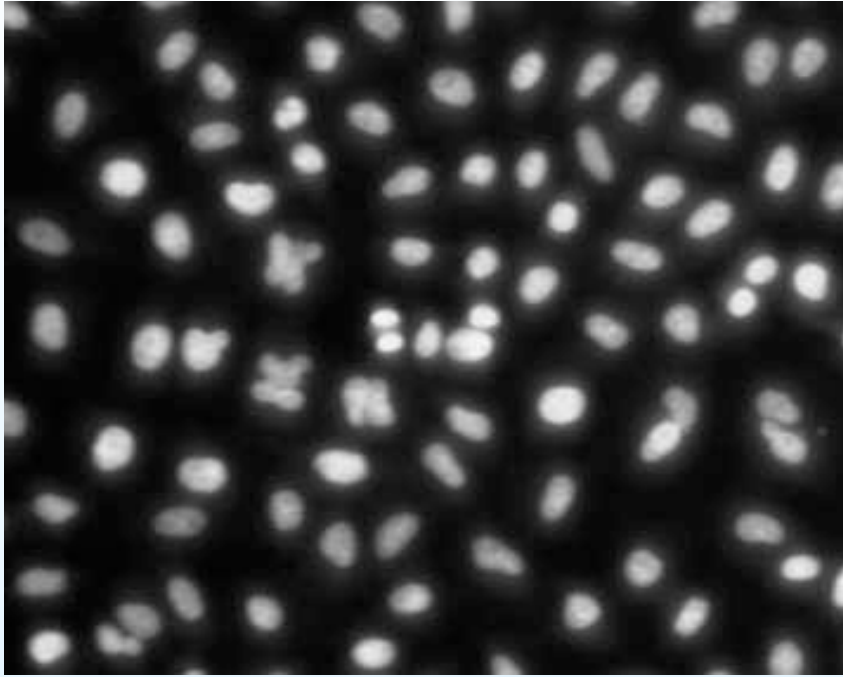
Venera 13, camera 1, spring loaded soil sampler

The Murphy principle works on Venus



Venera 14, camera 1, spring loaded camera hood sampler

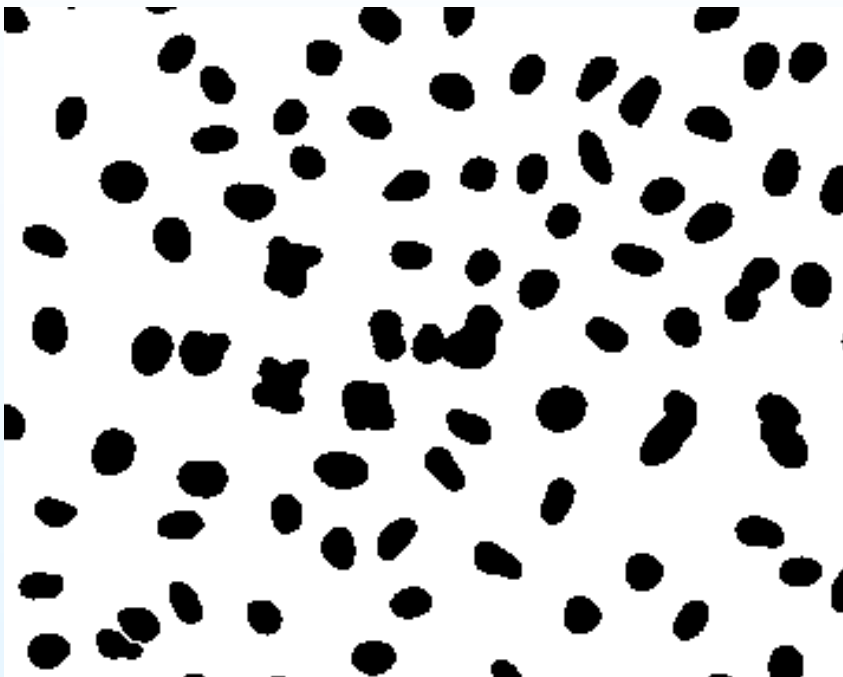
Examples of image analysis: cell counting



Cell nuclei

image processing, introduction – p.11/55

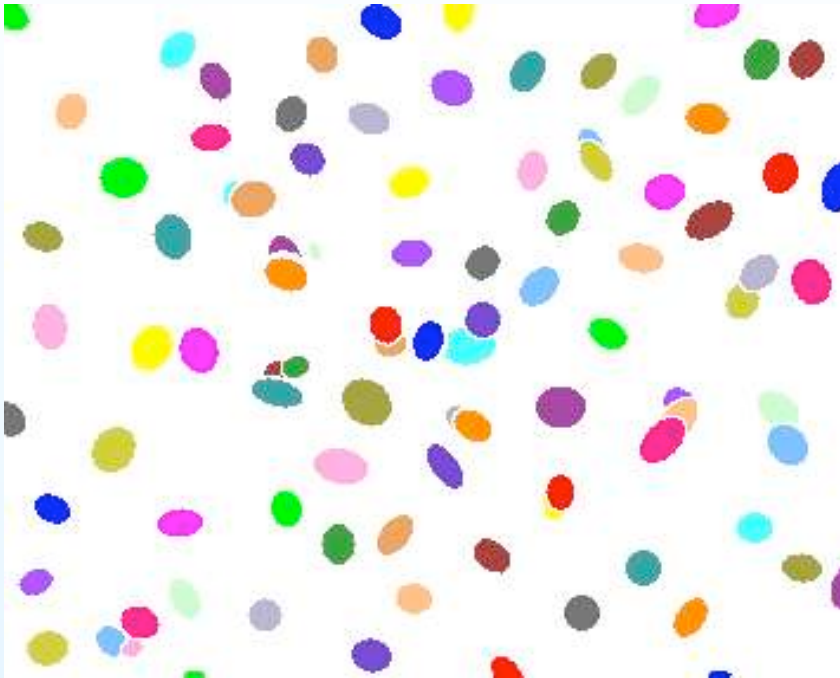
Examples of image analysis: cell counting



Segmentation

image processing, introduction – p.11/55

Examples of image analysis: cell counting



Object splitting

image processing, introduction – p.11/55

Why do we process images ?

- Image processing covers many different fields, among the most common:
 - Image data transmission (coding, compression, steganography);
 - Image filtering (enhancement, deblurring, special effects)
 - Image understanding (segmentation, pattern recognition, machine intelligence).
- Some application areas:
 - Science (medicine, materials science, biology, astronomy, remote sensing)
 - Law enforcement (face recognition, hyper-resolution)
 - Arts (special effects, film restoration, communication)

image processing, introduction – p.12/55

Why do we process images (cont.)?

- It is economically sensible to use image processing:
 - Automation of visual tasks (surveillance, quality control, industrial vision)
 - Cheap and/or plentiful captors (cameras are everywhere)
 - Enhanced senses (IR, UV, X-rays)
- When there is no other choice
 - Hubble space telescope in its early years
 - Robots in remote or dangerous locations (on Mars at present)
 - Forensic data (in its widest sense)
- When we want to make full use of the data
 - High-depth, high resolution sensors,
 - Multi-spectral data,
 - Rendering of 3D data,
- And far more.

image processing, introduction – p.13/55

Imaging basics: formation and perception

image processing, introduction – p.14/55

What are images ?

- Usually an image is a 2-D array of single values (grey-level images) or a 2-D array of 3-component vectors (R-G-B images). A single element of these arrays is called a pixel (**picture element**).
- Images can be 3-D or more (n-D arrays).
- Images can have time series of images (movies).
- Images can have a vector with many components for each pixel (multi- and hyperspectral images).
- Images can be collections of spatially connected arrays of various meanings (components).
- Image collections can be related spatially (stereo pairs).

Image can be fairly complicated things. We will mostly deal with the 2-D, grey-level or colour, single component case.

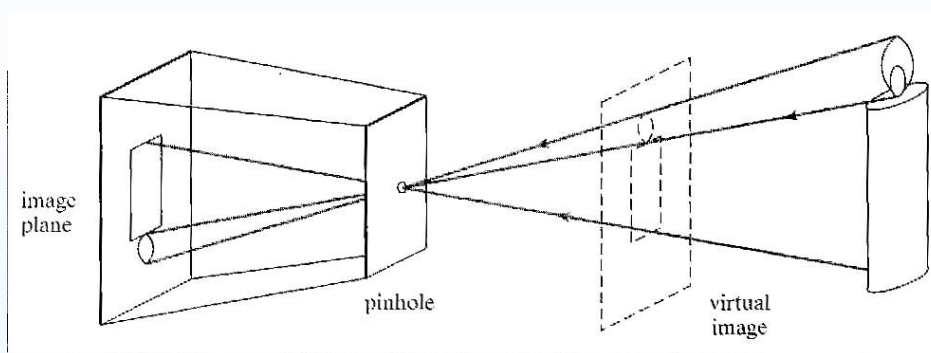
What does an image represent ?

- Projection of reality onto a surface/volume
- Mathematical image

$$I = f(R)$$

- Whole slew of problems associated with function f .
- It is worth spending some time on understanding the process of image formation and perception.

Image formation: pinhole camera

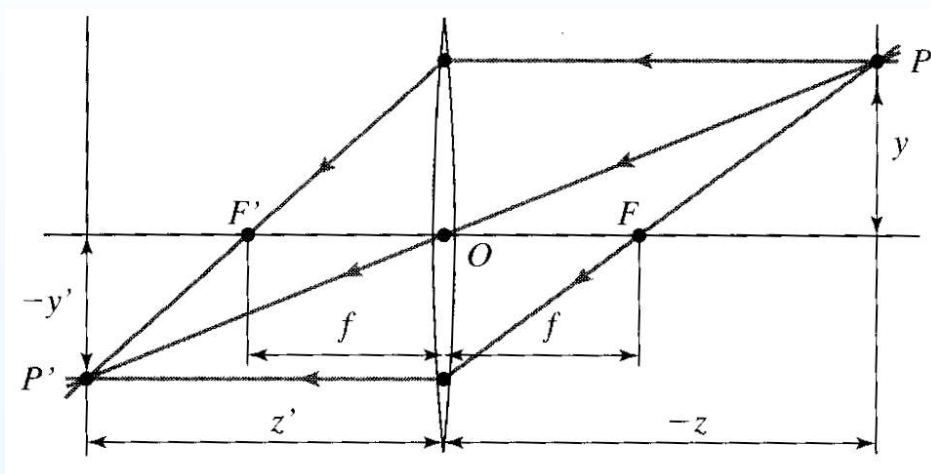


Advantages: Simple geometry, always in focus, useful model.

Drawbacks: Too simplistic: no diffraction, no aberrations.

image processing, introduction – p.17/55

Thin lens camera



Actually a somewhat realistic model for some instruments, e.g. telescopes. Issues:

- Depth of field, F number,
- Field of view,
- Diffraction, etc.

image processing, introduction – p.18/55

Real cameras

Real cameras have:

- Thick, multiple lens
- Geometric aberrations: spherical lens, barrel-pincushion, vignetting, etc.
- Chromatic aberrations: due to refraction index dependent of wavelength.

image processing, introduction – p.19/55

Real cameras: spherical aberration

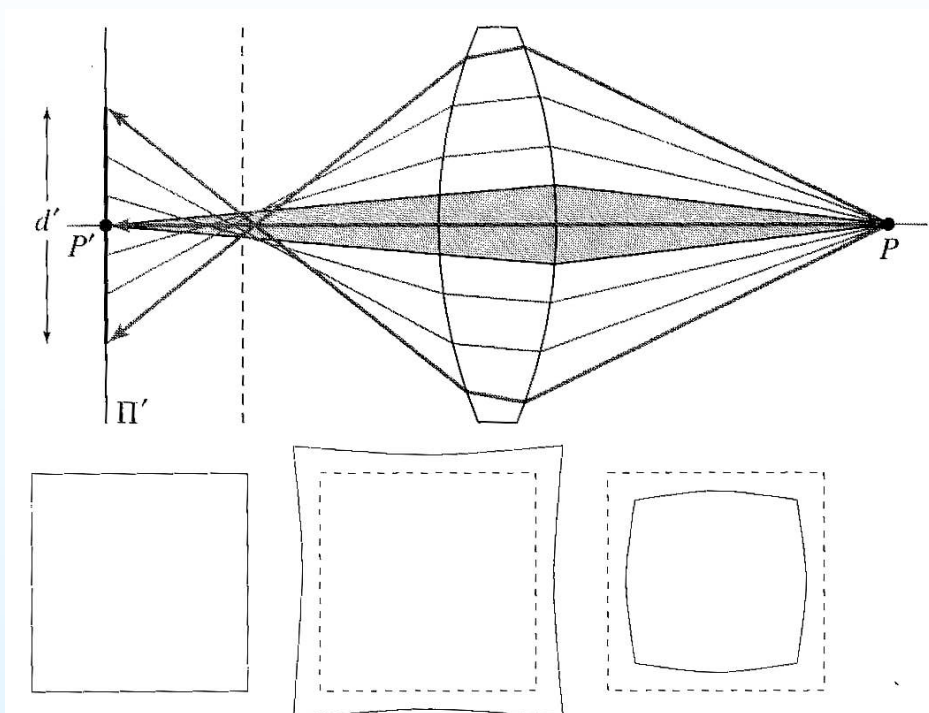


image processing, introduction – p.20/55

Real cameras: vignetting

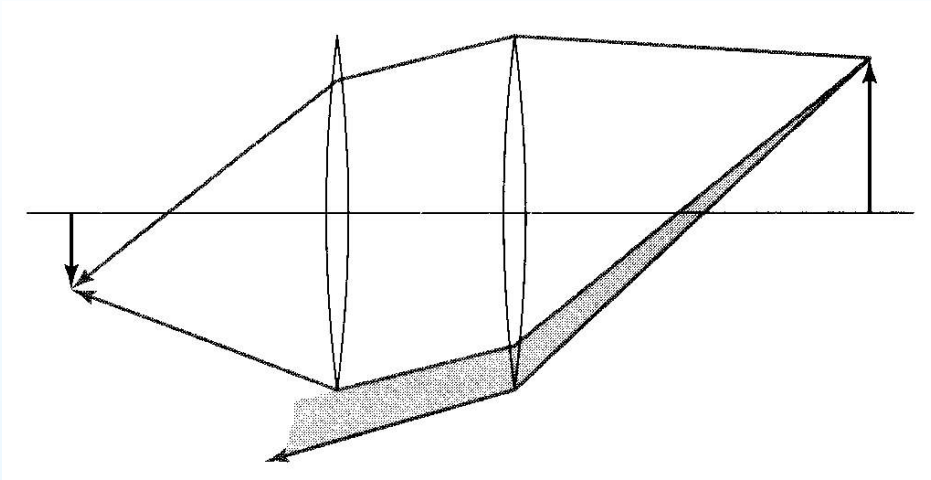


image processing, introduction – p.21/55

Sensors: photography

First photography: ca 1816, now lost. Among oldest photos on record is the following, original also lost in late 19th century.

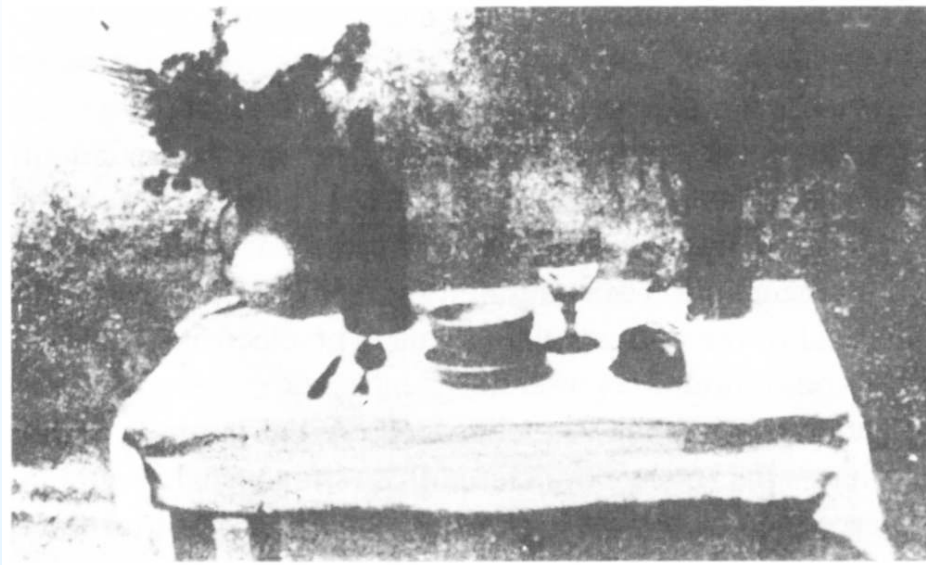


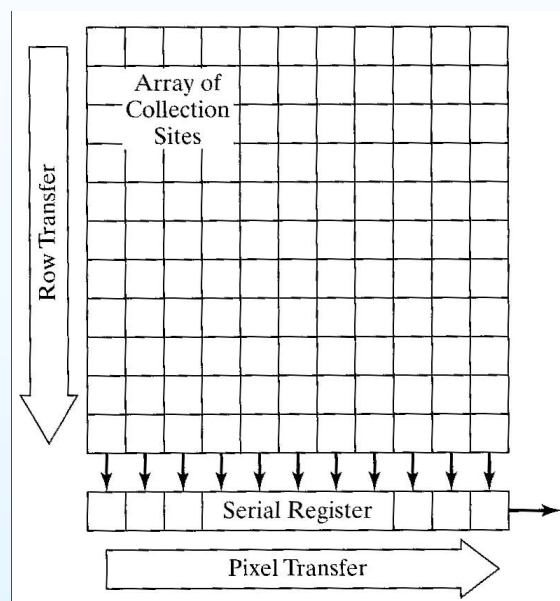
image processing, introduction – p.22/55

Sensors: history

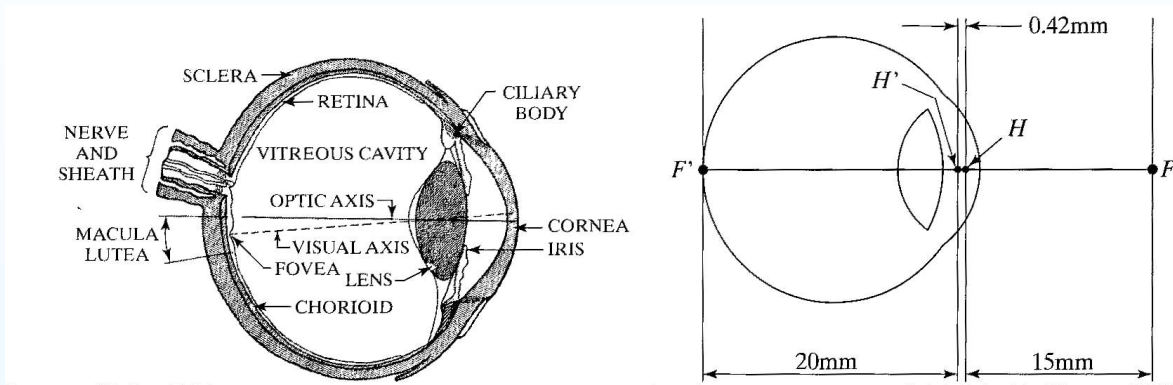
- First recorded photography: 1822
- Daguerréotypes: 1839 (popular photography)
- Wet plate: 1850 – Negative/positive process, immediate processing required.
- Gelatine process: 1870 – Latent image
- Photographic film: 1889 by Eastman
- Cinema : 1895 by Lumière brothers
- Colour photography: 1908
- Television: 1920s

Sensors: CCDs

Invented 1970, used in astronomy and in TV. Ubiquitous nowadays, being displaced in consumer electronics by CMOS sensors.



The eye



Rods and cones, colour perception,
Neurons/optic nerve,
Brightness control, tracking, interpretation, etc.
Physiology still being researched.

image processing, introduction – p.25/55

Light

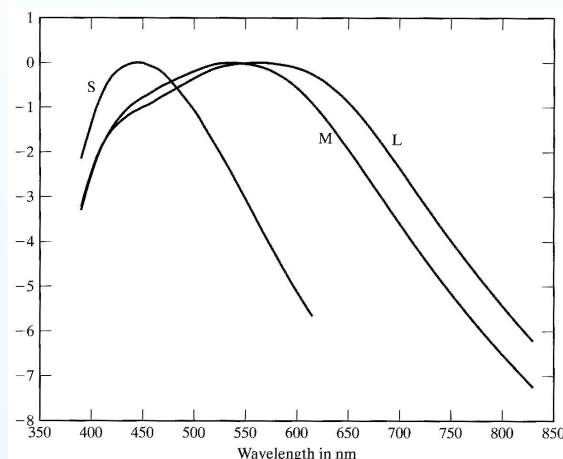
- Visible light is a subset of the electro-magnetic spectrum, from 380 to 720nm wavelength.
- A light source is characterized by (i) its strength and (ii) its spectrum.
- A monochromatic source is characterized by a wavelength λ and its luminance L .
- The eye:
 - Cannot distinguish individual frequencies (unlike hearing).
 - Instead obeys the GRASSMAN Laws:

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

i.e: the addition of N colours is equivalent to the addition of a white flux L_w and the flux of a single resulting monochromatic colour L_r of wavelength λ_r (3 variables)

image processing, introduction – p.26/55

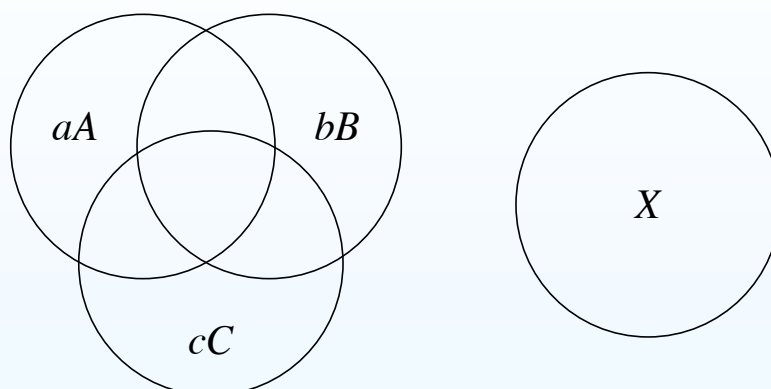
Colour perception



- There are 3 types of colour sensors in the eye (cones);
- Each type have a different spectral response.
- This suggests a 3-stimulus colour representation scheme : combining 3 stimuli of 3 different wavelengths can produce most colours seen in nature.

image processing, introduction – p.27/55

Colour matching experiment



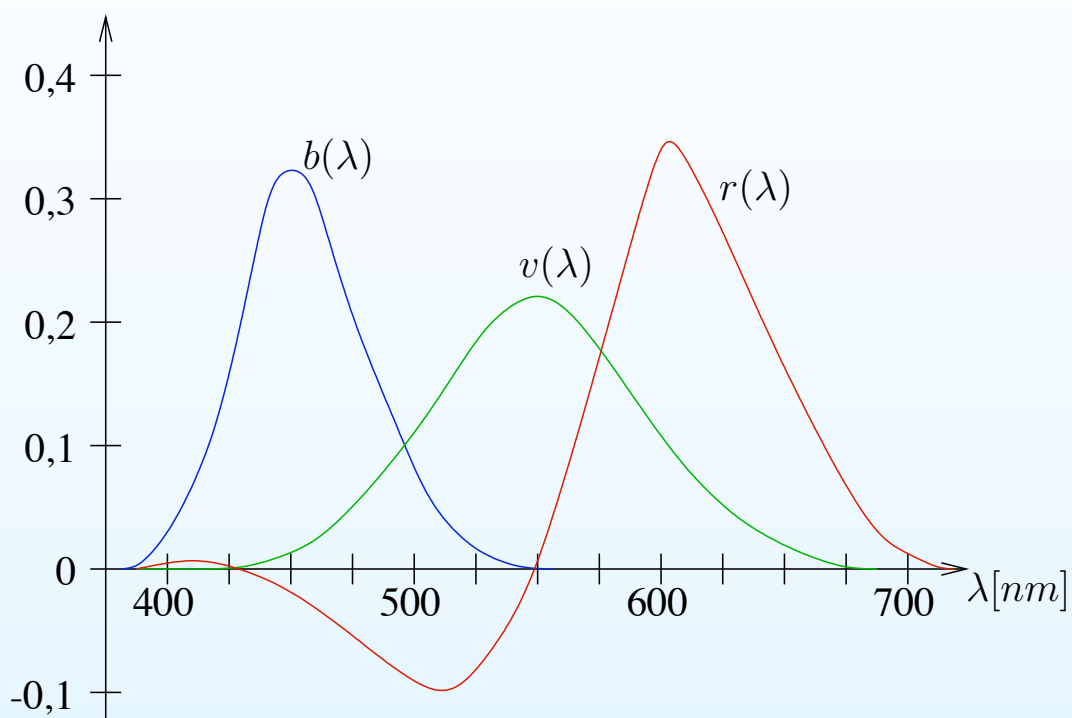
- A , B and C are 3 “primary” colours (light sources projected on a screen),
- a , b , c are 3 ponderation factors,
- Via experimentations, one tries to obtain colour X from $aA + bB + cC$.
- If a match is impossible, one or more of the primary colours is added to X instead, giving the equivalent to negative a , b or c .

image processing, introduction – p.28/55

CIE RGB standard

- In 1931, the Commission Internationale de l'Éclairage (CIE) did the colour matching experiment using 3 monochromatic sources: red (645.2 nm), green (526.3.1 nm) and blue (444.4 nm).
- This gave rise to the CIE RGB system.
- With this system, it is possible to represent most natural colours
- Some wavelengths require negative weights.

Colour matching functions



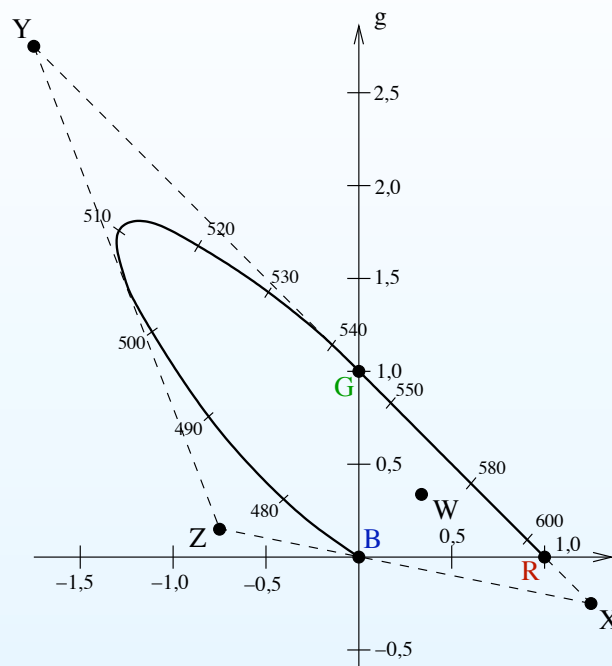
CIE XYZ standard

- To avoid negative weights, CIE designed a tri-stimulus system derived from RGB called XYZ, where all weights are positive.
- RGB / XYZ conversion by linear combination:

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

- Because a 3-stimulus (3-D) system is hard to represent, one most often uses the xyz system, which is normalized w.r.t $X + Y + Z$ (i.e. $x = \frac{X}{X+Y+Z}$).
- In this system only 2 variables are independent.

XYZ system



xy representation

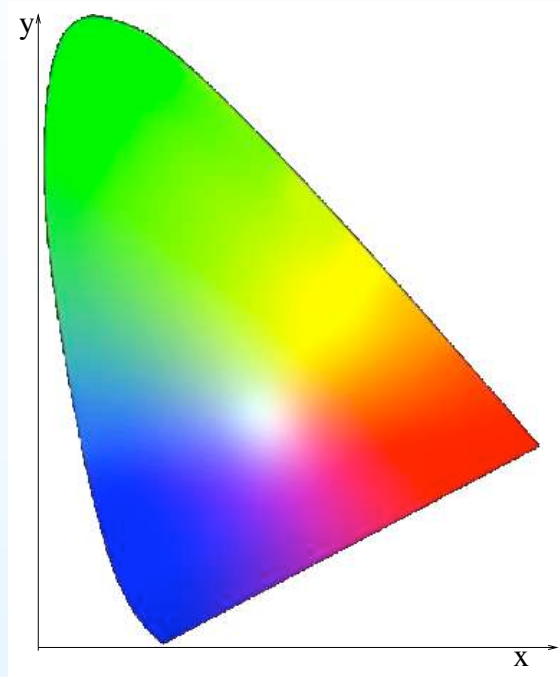


image processing, introduction – p.33/55

More on xy

- Monochromatic colours are on the inverted U.
- Equal energy point at $x = y = 1/3$.
- Again, xy is a reduced version of the full 3D colour space.
- The xy chromatic diagram is the locus of chrominance of maximal luminosity.
- For all its goodness, xy is not very intuitive. Other systems decompose colours in Hue, Saturation and Lightness.

image processing, introduction – p.34/55

Colour temperature

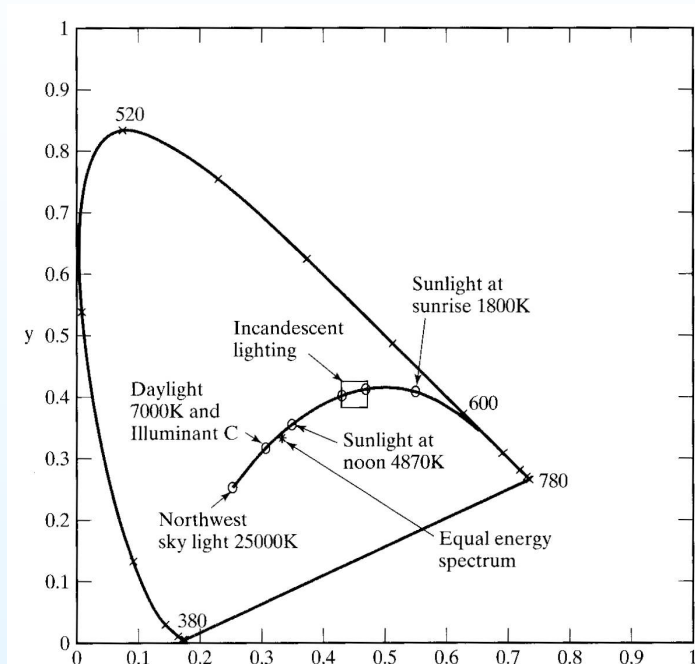
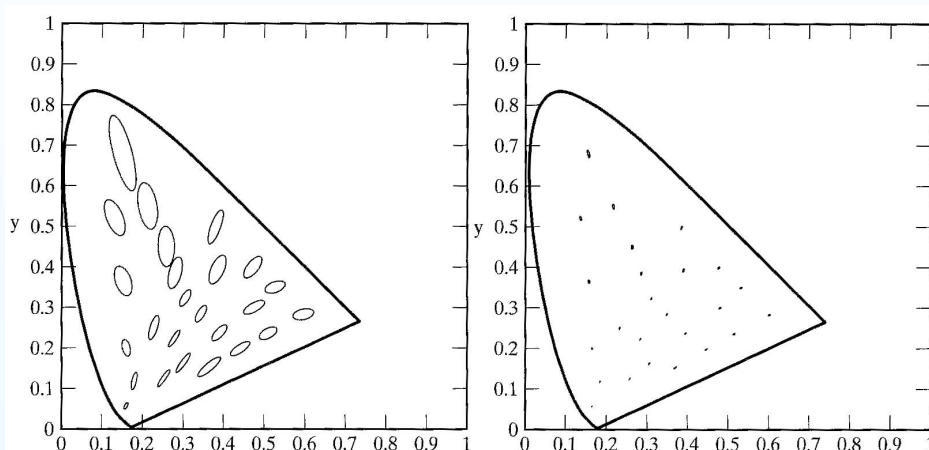


image processing, introduction – p.35/55

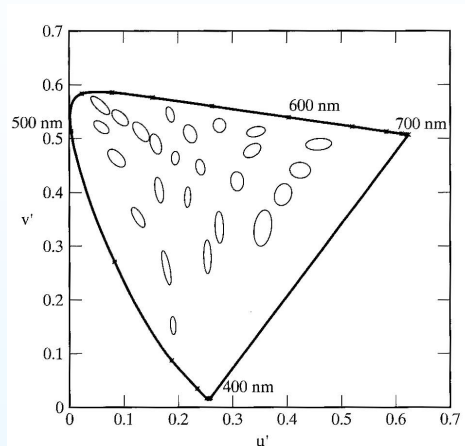
Colour difference perception



- Another human observer experiment: how far do you need to move away from a colour to notice a difference?
- Can we create a colour space where the differences are homogeneous, a *uniform* colour space?
- I.e.: can we reduce the ellipses above (MacAddam ellipses) to circles?

image processing, introduction – p.36/55

CIE $u'v'$ system



- Simple linear transform:

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

- This ignores differences in brightness
- A variety of non-linear transforms can be used

image processing, introduction – p.37/55

CIE Lab system

- Among those is the CIE Lab system.

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

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

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

- Here X_n, Y_n, Z_n are the XYZ coordinates of a reference white patch.
- Lab is important because it is substantially uniform.

image processing, introduction – p.38/55

RGB vs HSV

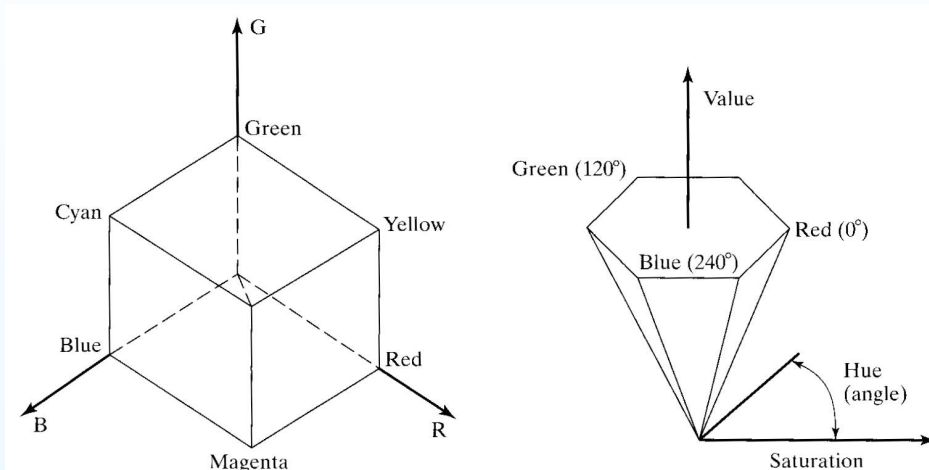


image processing, introduction – p.39/55

CMY subtractive system

- RGB and its derivatives are *additive* colour representation systems.
- For the printed media, colours added to a white page subtract wavelengths.
- The system for printed media is called *CYM* (Cyan, Yellow, Magenta). When added together, this produces black.
- Often *K* is added (black) as a separate colour: *CYMK* because creating a perfect black out of a subtracting system is hard.

image processing, introduction – p.40/55

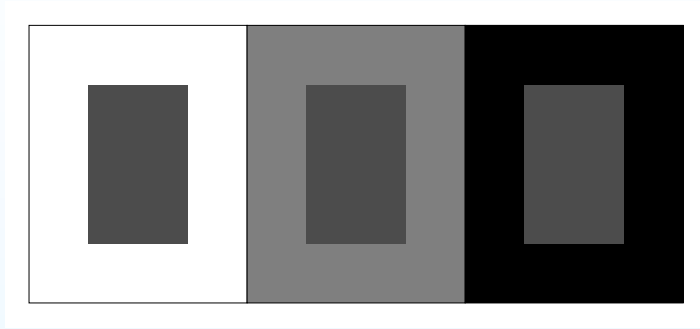
Everyday colour representation

- Computers, consumer electronics, etc use uncalibrated *RGB* (*not* CIE) with non-linear brightness curve (gamma).
- Most printers use uncalibrated *CYMK*
- As a result: limited gamut, colour matching problems.
Exception: Apple hardware.
- TV uses a system called YC_bC_r (luminance/chrominance).

Other points on colour perception

- Spatial and temporal effects are ignored
 - Chromatic adaptation: ability of the visual system to adapt to a dominant colour.
 - Assimilation: influence of surrounding colours towards these colours
 - Contrast: influence of surrounding colour to move away from these colours.
- Everyone is different. About 10% of males have some sort of “colour blindness”, i.e. lack of one or more kind of receptors, usually the red cones. Some rare females can have 4 kinds of receptors.
- Illumination is a big factor. We are all colour blind at night (rod receptors are more sensitive than cones).
- We’ve only scratched the surface.

Example of contrast perception



Which is the darkest inner rectangle?

Example of chromatic adaptation

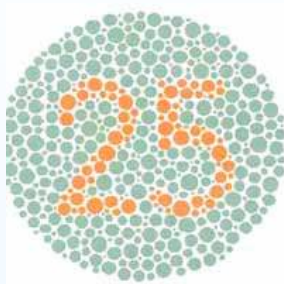


Example of chromatic adaptation

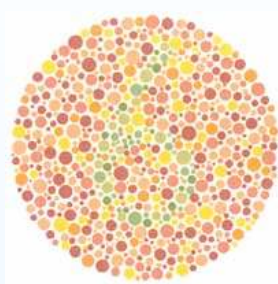


image processing, introduction – p.44/55

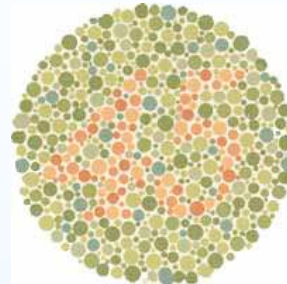
Colour perception test



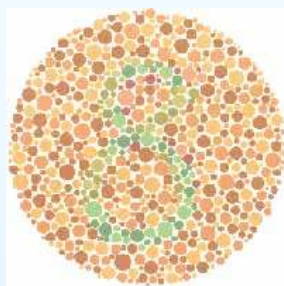
(a)



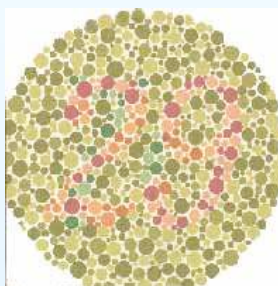
(b)



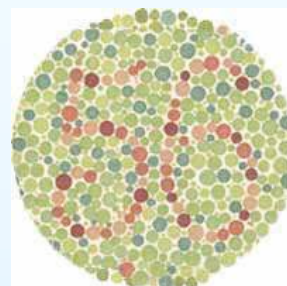
(c)



(d)



(e)



(f)

image processing, introduction – p.45/55

Application: melanoma diagnosis



Solarscan device

Polartechnics

Solarscantm



Hand piece

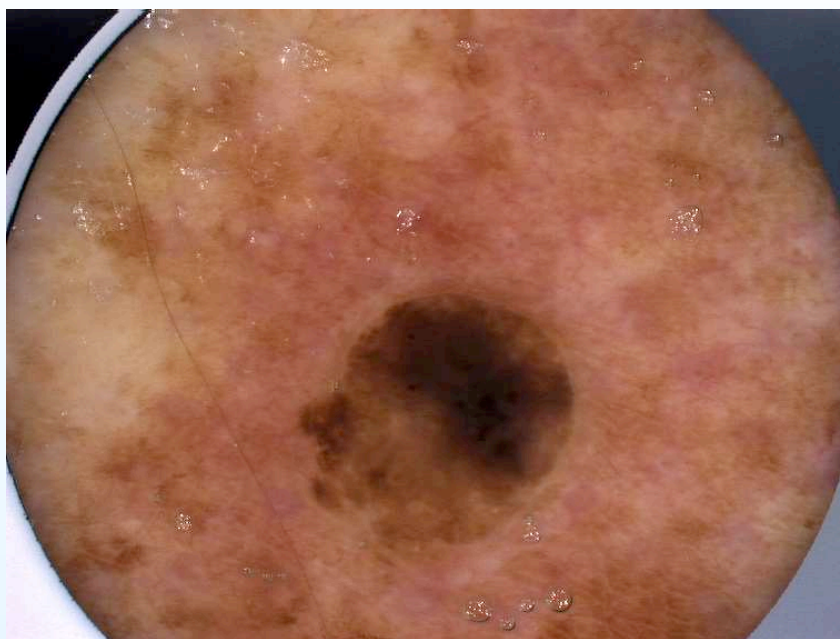
device,

see

<http://www.polartechnics.com.au>.

image processing, introduction – p.46/55

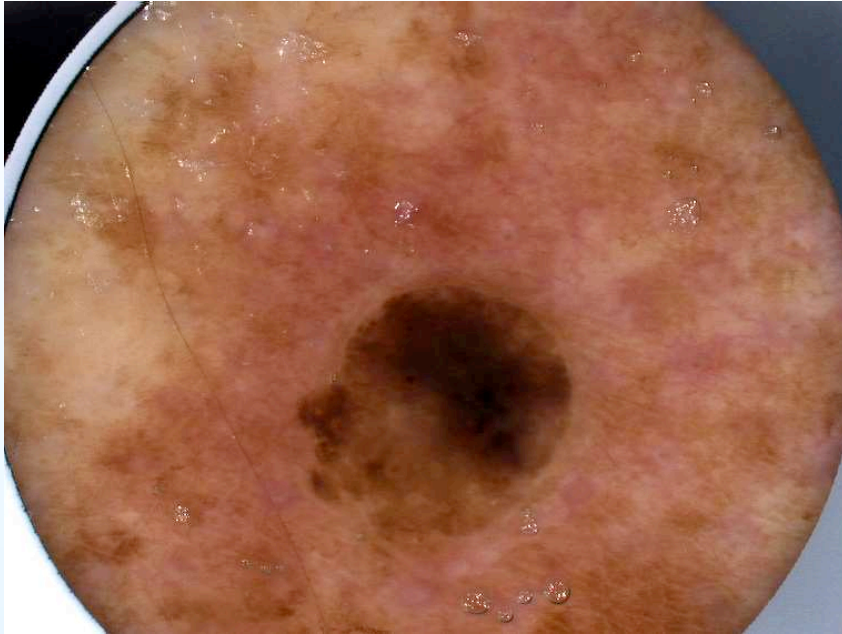
Absolute colour segmentation



Uncalibrated lesion 1

image processing, introduction – p.47/55

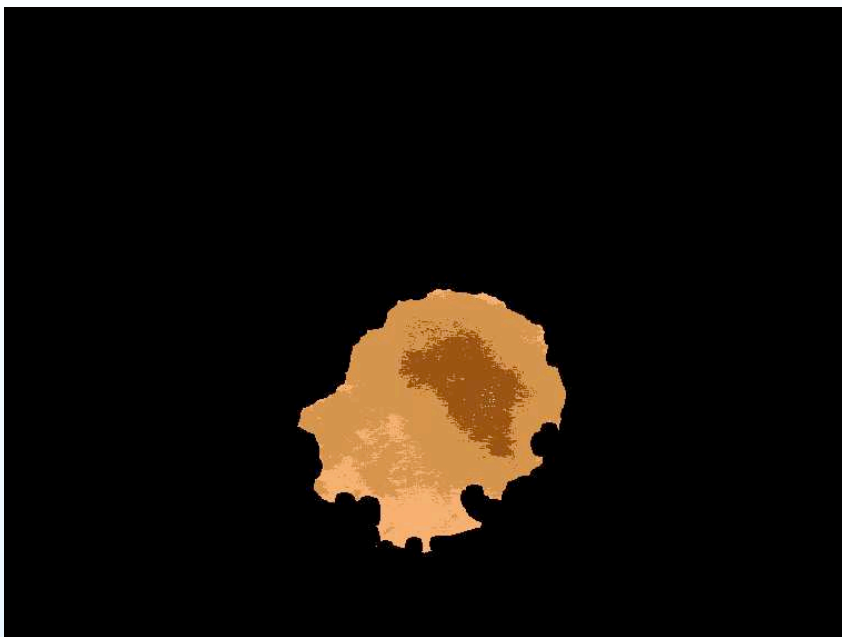
Absolute colour segmentation



Calibrated lesion 1

image processing, introduction – p.47/55

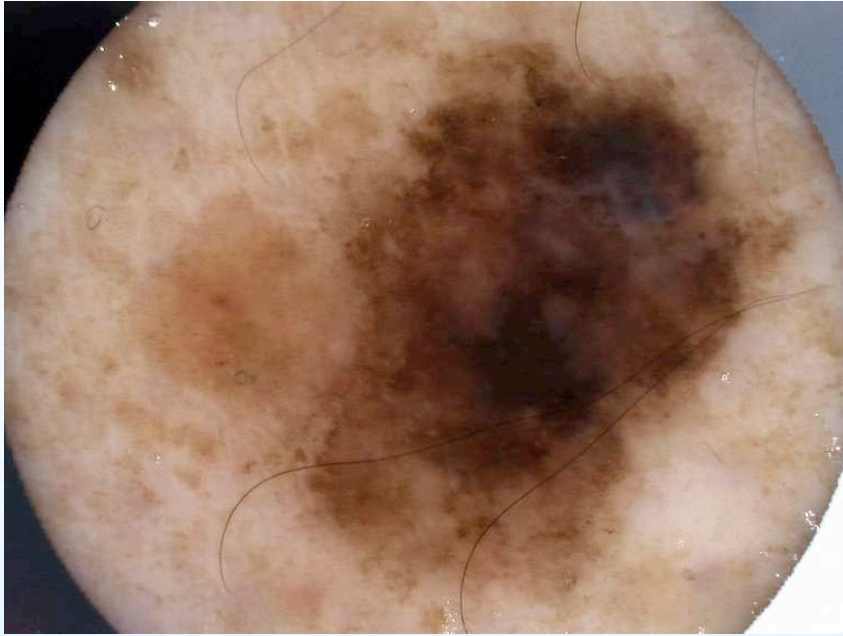
Absolute colour segmentation



Colour identification for lesion 1 -> benign

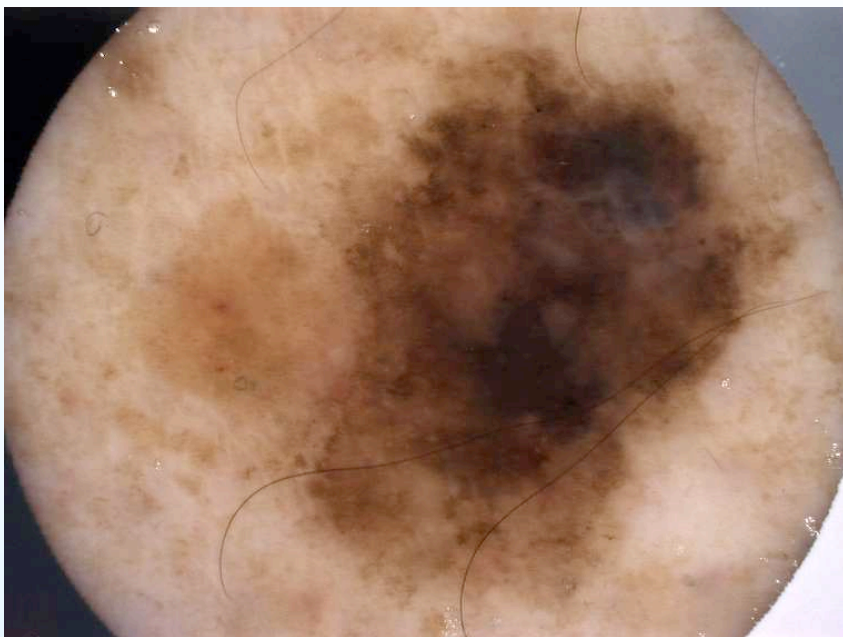
image processing, introduction – p.47/55

Absolute colour segmentation



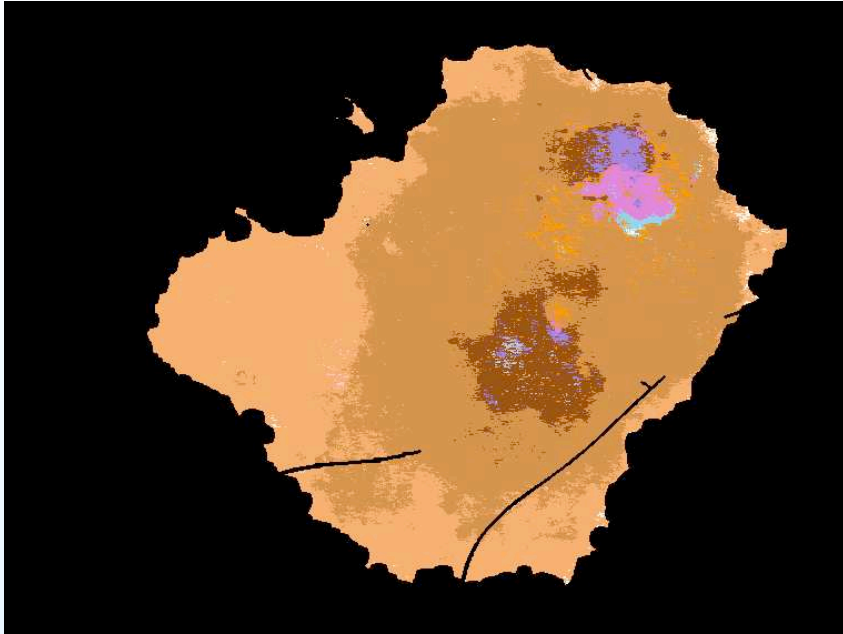
Uncalibrated lesion 2

Absolute colour segmentation



Calibrated lesion 2

Absolute colour segmentation



Colour identification for lesion 2 -> melanoma

Image representation

2-D arrays of 8-bit data

- Simplest image representation: collection of 1 (GL) or 3 (Colour) 2-D arrays of 8-bit data.
- Enough to represent most visible grey levels (256) , and most visible colours ($255^3 = 16$ millions).
- Most problems that occur with more dimensions, more channels, more bit depths, etc. can be illustrated within that framework.

Resolution

Resolution pyramid, influence of the spatial resolution



full



half



1/4



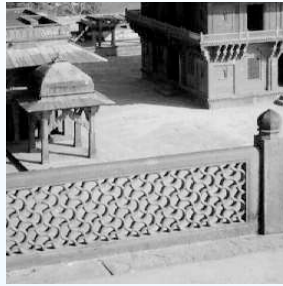
1/8

Bit depth

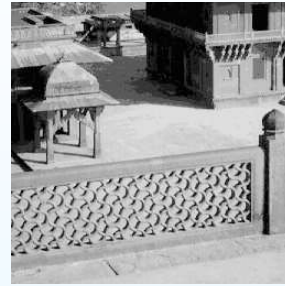
Influence of the number of grey levels.



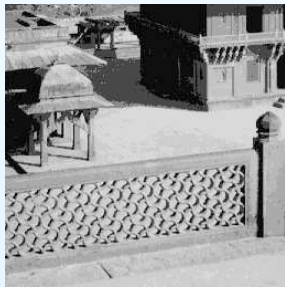
256 gl



64 gl



16 gl



8 gl



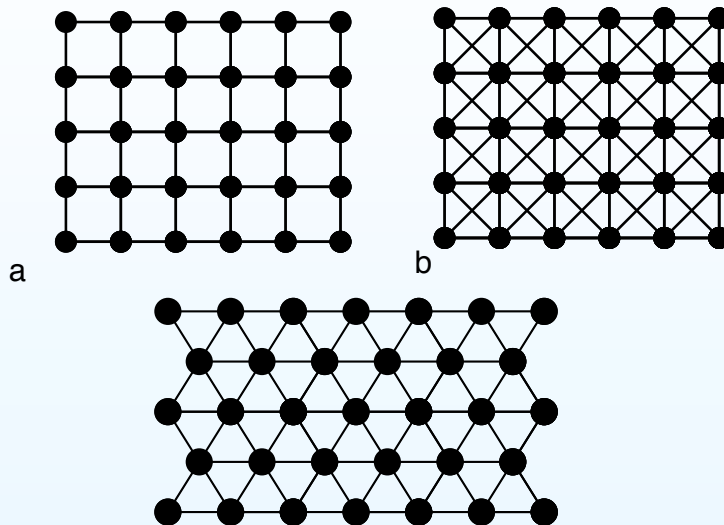
4 gl



2 gl

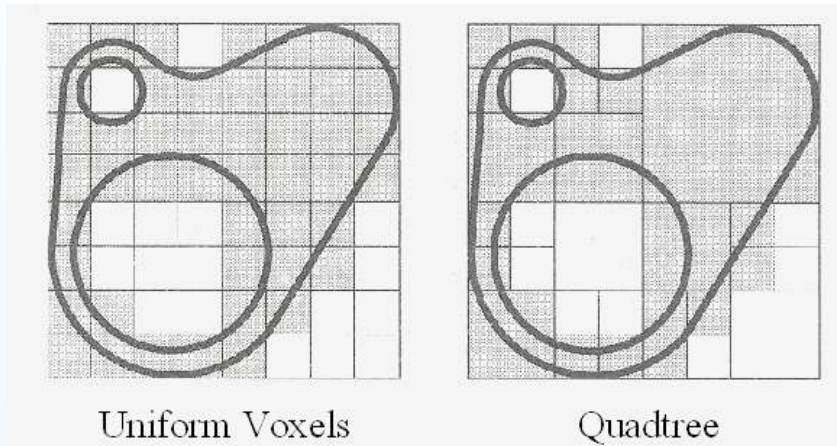
image processing, introduction - p.51/55

Graph



Synonym: connection graph, grid.

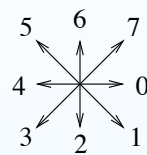
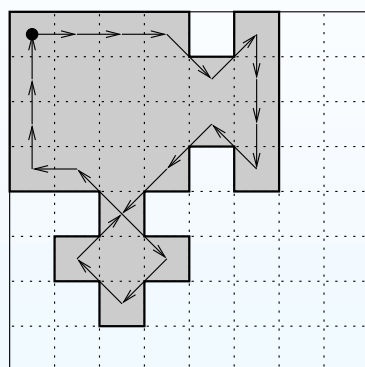
Other representations: Quadrees



Quadrees are a hierarchical space decomposition structure, which allow for more efficient representation than straight pixel in general. The fundamental idea behind the quadtree is that any image can be split into four quadrants. Each quadrant may again be split in four sub quadrants, etc. In the quadtree, the image is represented by a parent node, while the four quadrants are represented by four child nodes, in a predetermined order.

image processing, introduction - p.53/55

Other representations: Chain-coded contours



Code : 00017222533135754666

Chain-coded contours can allow users to store shapes efficiently.

References and acknowledgements

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.

Cours de Morphologie Mathématique

Concepts de base

Hugues Talbot
talboth@esiee.fr

ISBS / ESIEE 2006

Concepts de base – p. 1/2

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. 2/2

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

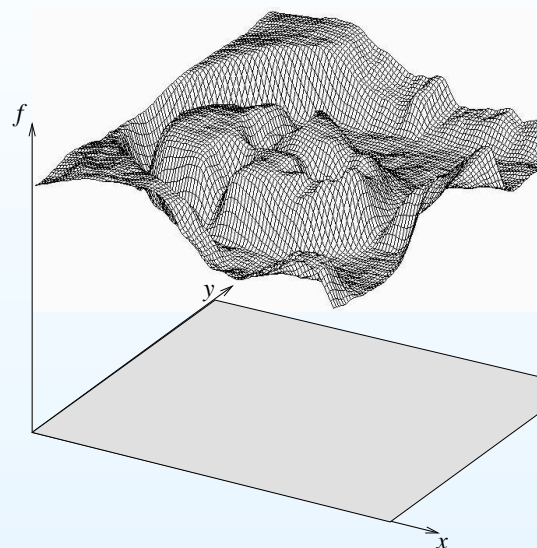
- Représentation la plus simple: collection of 1 (N&B) or 3 (Couleur) tableau(x) 2-D en 8 bits.
- Suffisant pour représenter la plupart des niveaux de gris visibles (256) et des couleurs visibles ($255^3 = 16$ millions).
- La plupart des problèmes possibles peuvent être illustrées dans ce contexte.

Concepts de base – p. 5/2

Fonction 2-D



Image



Fonction

Concepts de base – p. 6/2

Résolution



full



half



1/4



1/8

Nombre de niveaux de gris



256 gl



64 gl



16 gl



8 gl

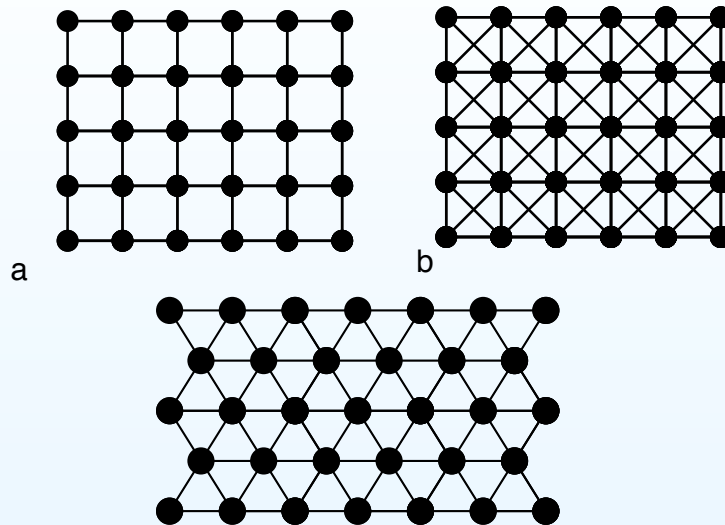


4 gl



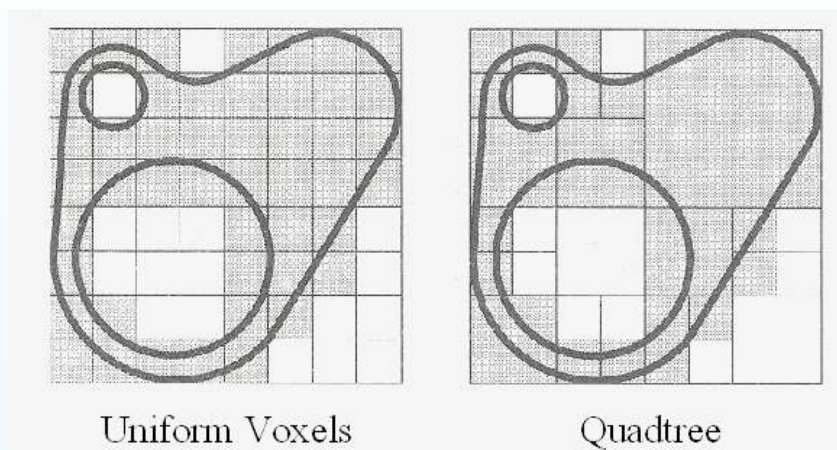
2 gl

Graphe



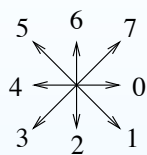
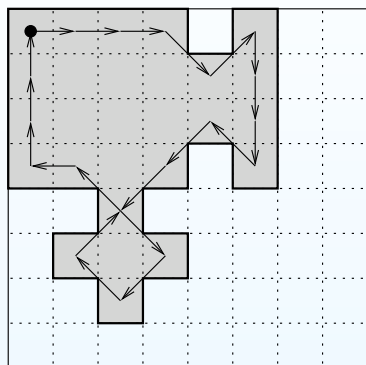
Synonymes: graphe de connexion, grille.

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



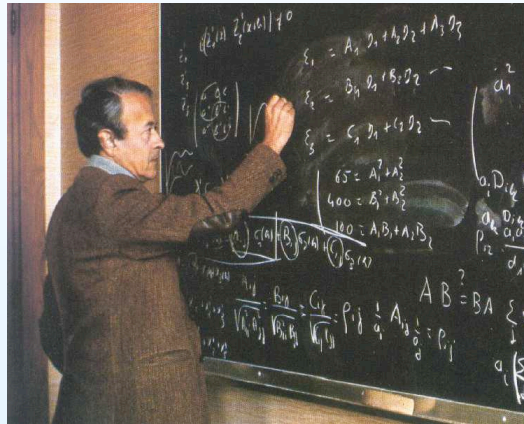
Code : 00017222533135754666

Permettent de représenter les contours de façon efficace.

Concepts de morphologie mathématique

Un peu d'histoire

- La morphologie mathématique (MM) a été conçue au milieu des années 60 à l'ENSMP, 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.



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.

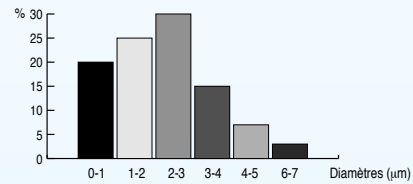
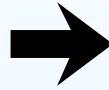
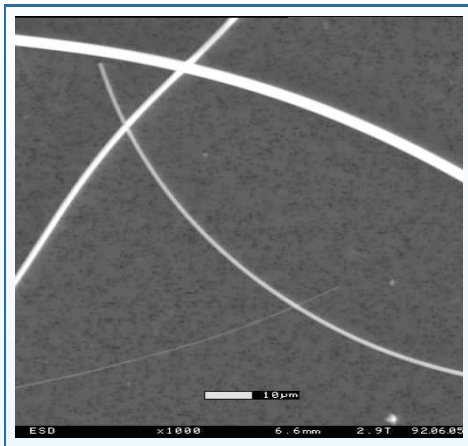
Histoire (3)

- 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 ?



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.

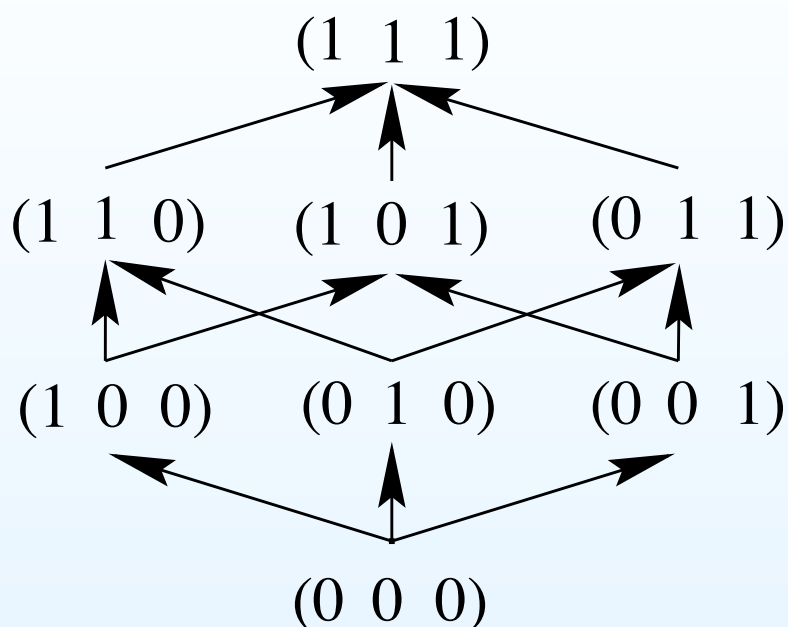
Concept de base: l'ordre

- 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 3×1 .

Autres exemple de treillis

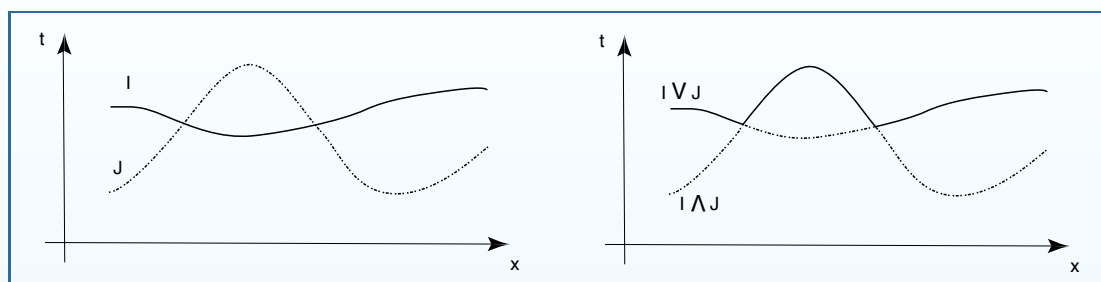
- Treillis des parties $\mathcal{P}(E)$ d'un ensemble E . Ordre défini par l'inclusion. $\text{Sup}=\cap$, $\text{Inf}=\cup$. Extrêmes= E , \emptyset .
- Treillis des réels. Ordre=ordre numérique sur \mathbb{R} . Sup et Inf au sens usuel, Extrêmes= $-\infty$, $+\infty$.
- Treillis des ensembles convexes. Sup =enveloppe convexe de l'union. 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.

Treillis des fonctions



sup et inf de deux images 1-D

Cours de Morphologie Mathématique

Dilatations, érosions

Hugues Talbot
talboth@esiee.fr

ISBS / ESIEE 2006

Erosions, Dilatations – p. 1/2

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. 2/2

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

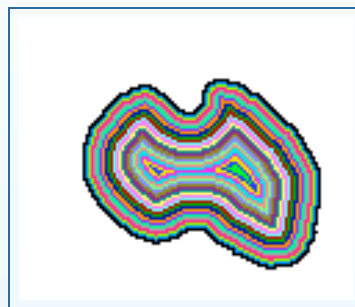
Erosions, Dilatations – p. 5/2

Concept de base: la distance

- La fonction distance



Noyaux de cellules



Fonction distance

- Dépendente du graphe sous-jacent.

Erosions, Dilatations – p. 6/2

Concept de base: la mesure

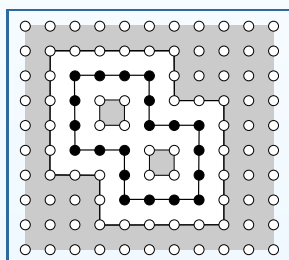
C'est ce qui nous intéresse *in fine*.

- Nombre d'objets.
- Longueur, aire, volume.
- Orientation, élongation.
- La MM est surtout basé sur la mesure, non la forme.

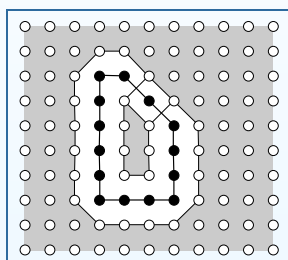
Concept de base: la grille/le graphe

La grille digitale pose des problèmes:

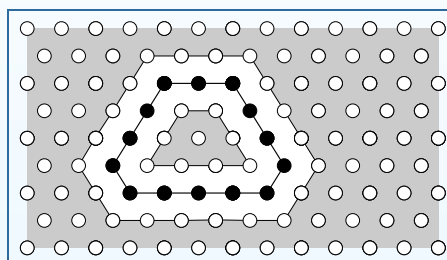
- Connectivité:



4-connecte



8-connecte

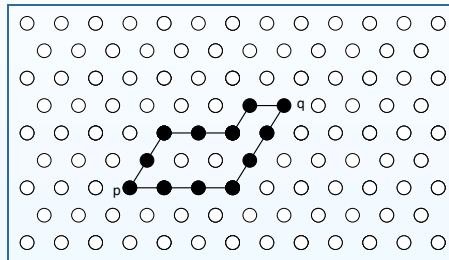


6-connecte

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 ψ et ψ^* 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 complémente :

- La géométrie discrete (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 dilations

Erosion

Dans le contexte le plus général, une érosion ϵ est un opérateur qui commute avec l'inf.

$$\bigwedge \epsilon(\cdot) = \epsilon \bigwedge (\cdot)$$

Ce qui n'est pas très intuitif. On commence par définir l'érosion par élément structurant (ES):

$$\epsilon_B(X) = \{x \mid B_x \subseteq X\}$$

C-à-d: est-ce que l'ensemble contient l'ES?

Définition équivalente:

$$\epsilon_B(X) = \bigcap_{b \in B} X_{-b}$$

Erosion sur le treillis des fonctions:

Dans ce cas:

$$\epsilon_B(f) = \bigwedge_{b \in B} f_{-b}$$

Ce qui est équivalent à:

$$[\epsilon_B(f)](x) = \min_{b \in B} f(x + b)$$

Érosion par un ES non plat (rarement utilisé dans la pratique, sauf avec une parabole)

$$[\epsilon_{B_v}(f)](x) = \min_{b \in B_v} \{f(x + b) - B_v(b)\}$$

Dilatation

La dilatation est le dual de l'érosion: une dilatation δ 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):

$$\delta_B(X) = \{x | B_x \cap X \neq \emptyset\}$$

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:

$$\delta_B(f) = \bigvee_{b \in B} f_{-b}$$

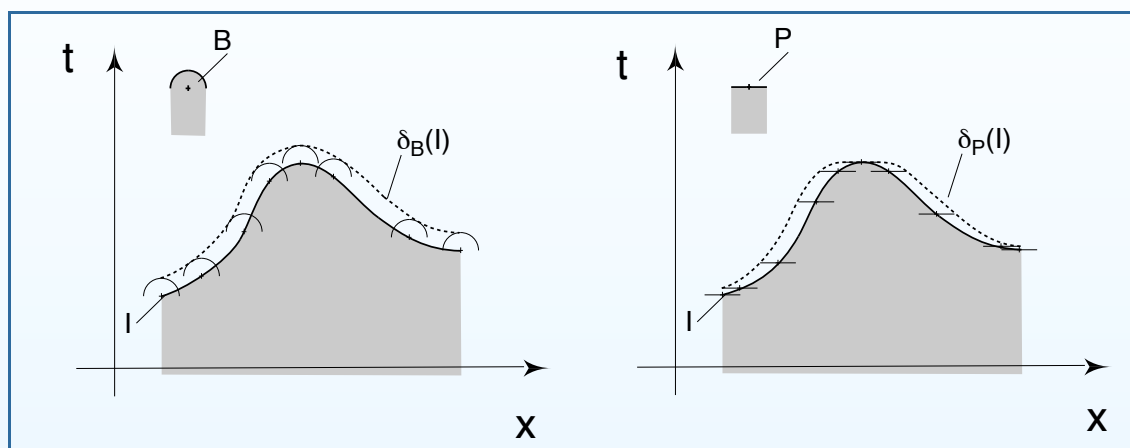
Ce qui est équivalent à:

$$[\delta_B(f)](x) = \max_{b \in B} f(x + b)$$

Dilatation par un ES non-plat:

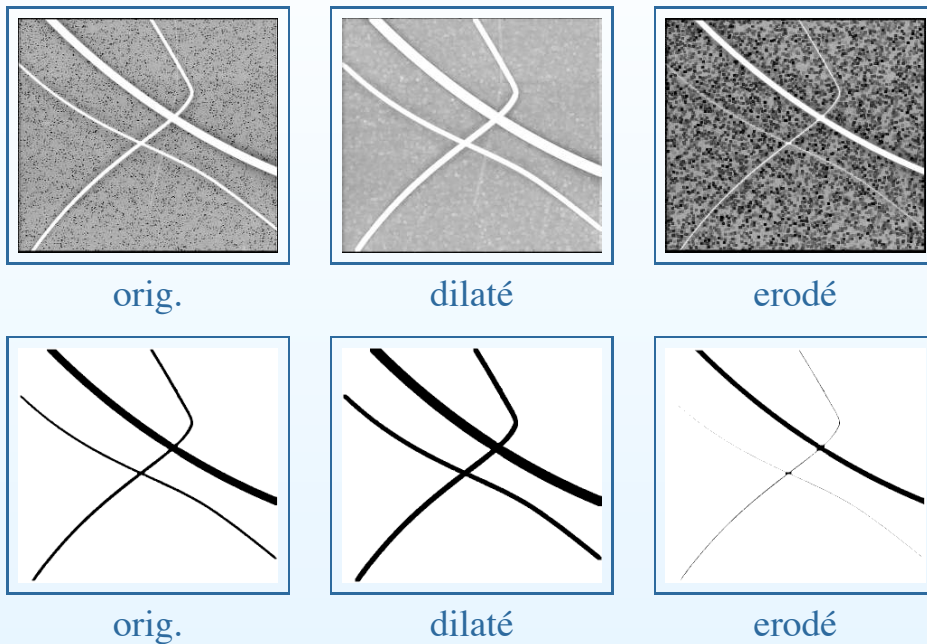
$$[\delta_{B_v}(f)](x) = \max_{b \in B_v} \{f(x + b) + B_v(b)\}$$

Illustrations - érosion/dilatation



Dilatation d'une image 1-D.

Illustration (cont.)



Erosions, Dilatations – p. 19/2

Propriétés

- dualité: $\delta_B(f) = -\epsilon_B(-f)$
- croissance: $f \leq g \Rightarrow \delta_B(f) \leq \delta_B(g), \epsilon_B(f) \leq \epsilon_B(g)$
- distributivité: $\delta(\bigvee_i f_i) = \bigvee_i \delta(f_i) ; \epsilon(\bigwedge_i f_i) = \bigwedge_i \epsilon(f_i)$
- composition: $\delta_{B_2} \delta_{B_1} = \delta_{(\delta_{\tilde{B}_2} B_1)} ; \epsilon_{B_2} \epsilon_{B_1} = \epsilon_{(\delta_{\tilde{B}_2} B_1)}$
- relation d'ordre: $\epsilon_B \leq I \leq \delta_B$ ssi B contient son origine.
- invariance par translation.

Erosions, Dilatations – p. 20/2

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 λ , où λ 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 λ (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 2006

Ouverture & fermetures – p. 1/2

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. 2/2

Notation de Minkowski

Addition et soustraction ensembliste:

- Addition de Minkowski:

$$A \oplus B = \{y = a + b, a \in A, b \in B\}$$

d'où :

$$\delta_B(A) = A \oplus \check{B}$$

- Soustraction de Minkowski:

$$A \ominus B = \{x, x + b \in A, b \in B\}$$

d'où :

$$\epsilon_B(A) = A \ominus \check{B}$$

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

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

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.

$$\gamma_B(f) = \delta_{\tilde{B}}[\epsilon_B(f)]$$

Note: le résultat est indépendant de la position de l'origine.

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.

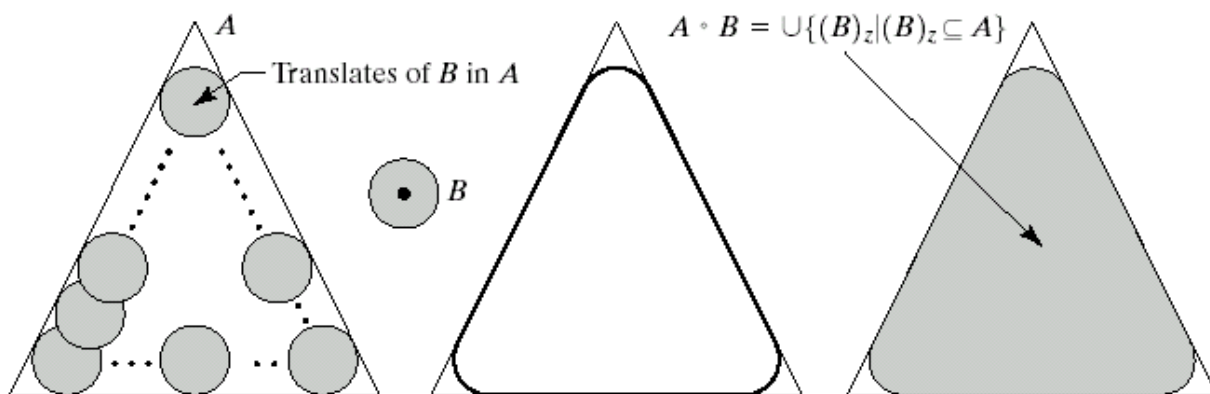
Définition sur les fonctions

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*.

$$\phi_B(f) = \epsilon_{\check{B}}[\delta_B(f)]$$

Le résultat ne dépend pas de la localisation de l'origine.

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

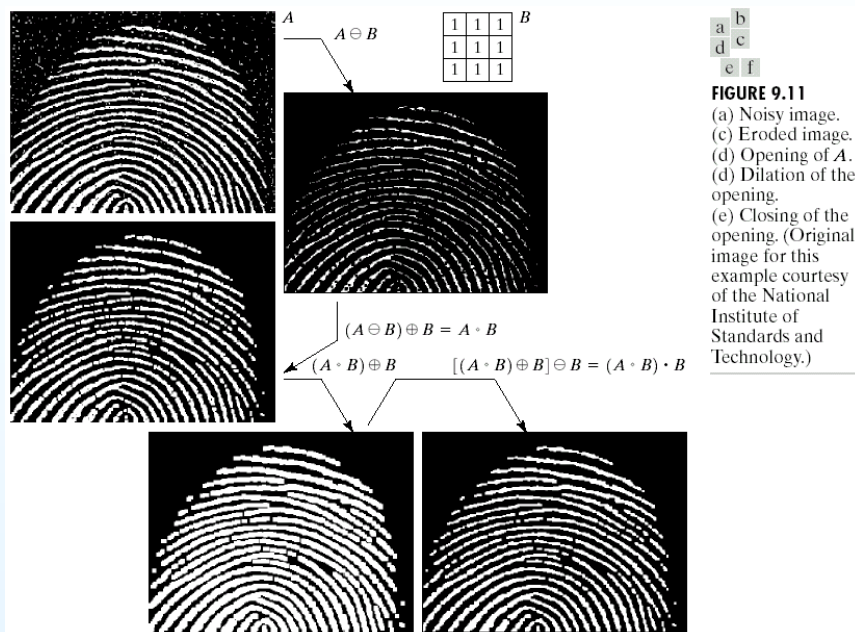
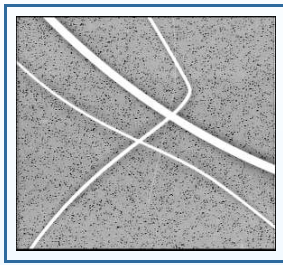
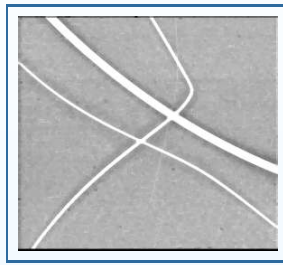


FIGURE 9.11
 (a) Noisy image.
 (c) Eroded image.
 (d) Opening of A .
 (d) Dilation of the opening.
 (e) Closing of the opening. (Original image for this example courtesy of the National Institute of Standards and Technology.)

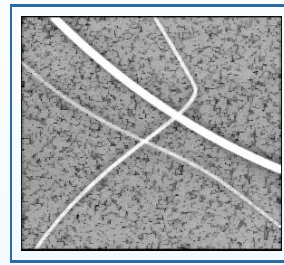
Ouverture/fermeture: Illustration



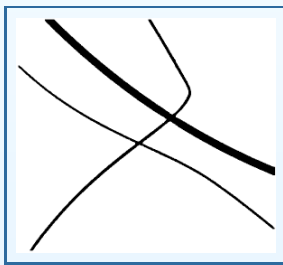
orig.



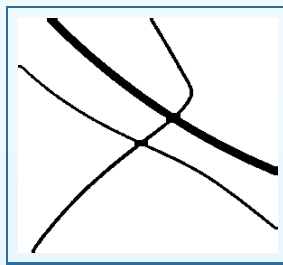
fermeture



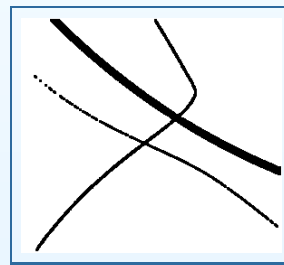
ouverture



orig.



fermeture



ouverture

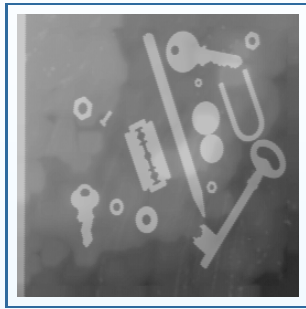
Ouverture & fermetures – p. 15/2

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. 16/2

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

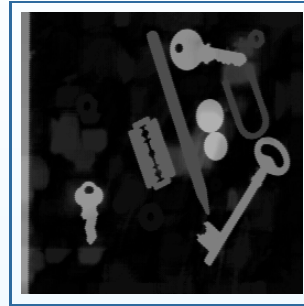
Exemple d'opérateurs algébriques



Outils.



Érosion.



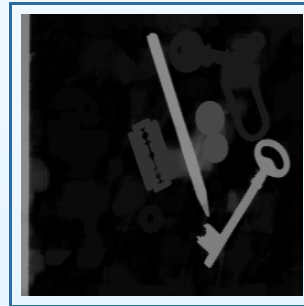
Reconstruction.



Outils.



Union de lignes.



Reconstruction.

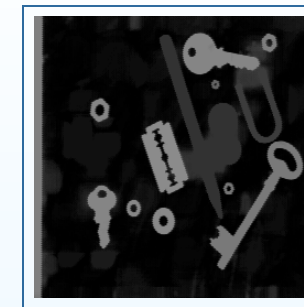
Opérateurs algébriques (suite)



Fermeture surfacique.

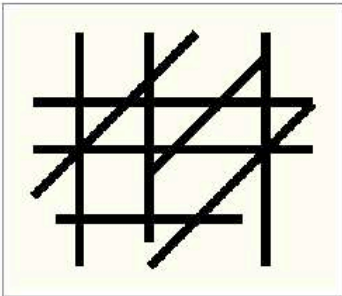


Detection des trous.

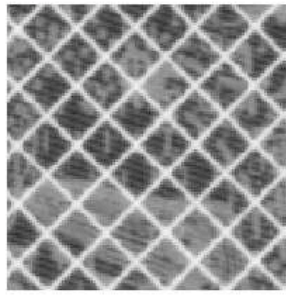


Reconstruction.

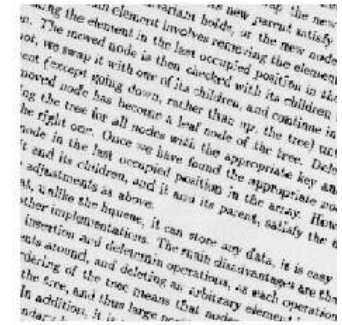
Rose des directions



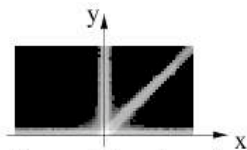
Network of lines



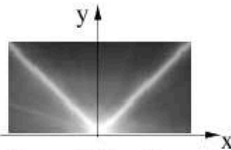
Texture on a soil map



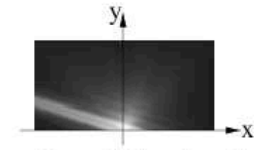
Slanted text



Rose of directions by openings

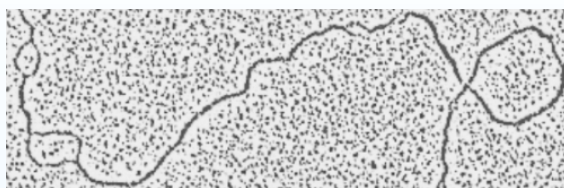


Rose of directions by openings

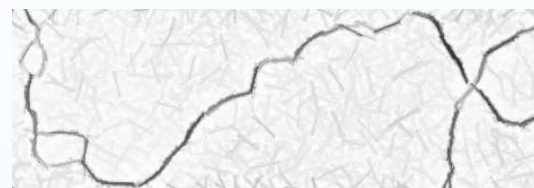


Rose of directions by opening

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

2005

Géodésie – résidus – p.1/25

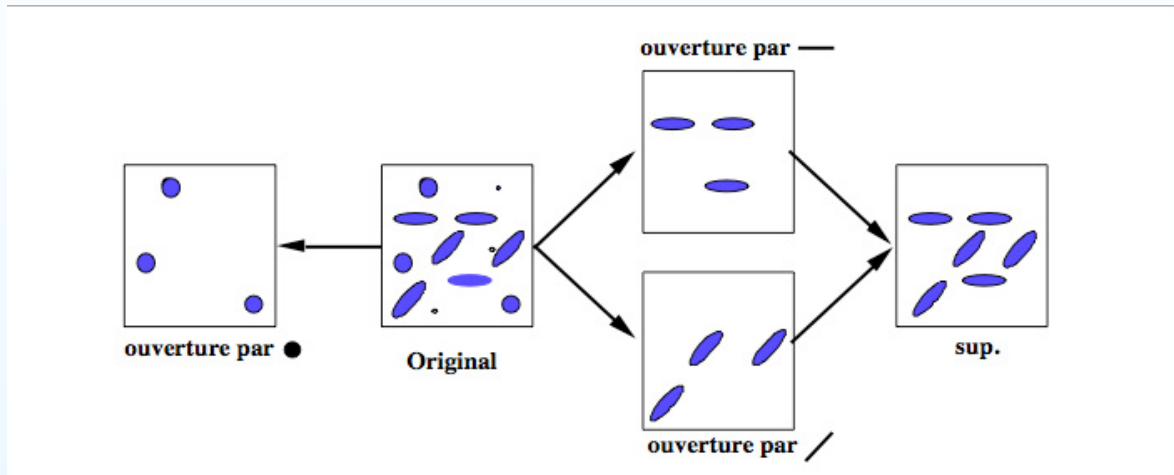
Rappel du cours précédent

- 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

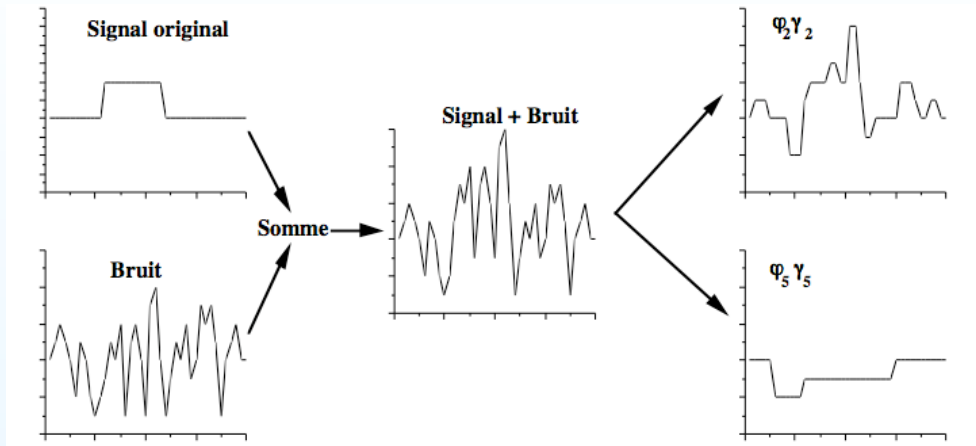
Illustration du filtrage morphologique simple:



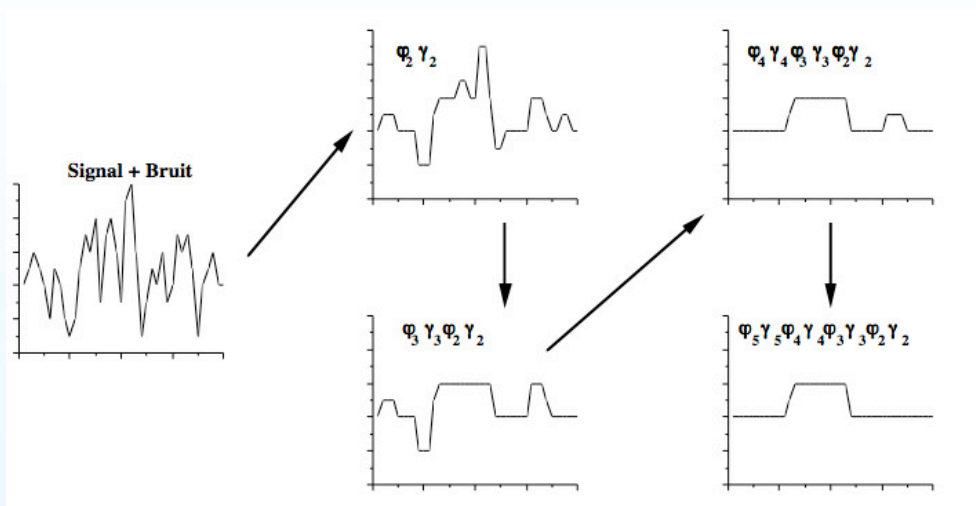
Filtres alternés séquentiels

- 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 λ_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 γ_λ , avec λ 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

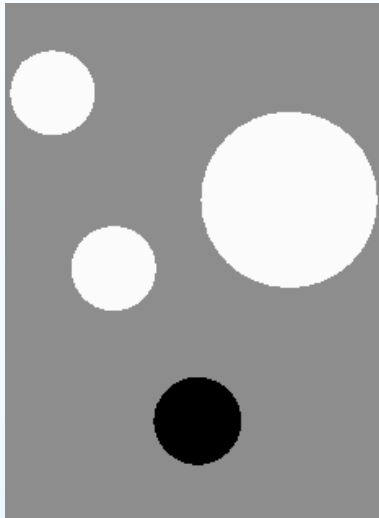
$$\gamma_\lambda \gamma_\mu = \gamma_\mu \gamma_\lambda = \gamma_{\vee\{\lambda, \mu\}}.$$

On construit la *courbe granulométrique* de la façon suivante, avec un paramètre *lambda* discret:

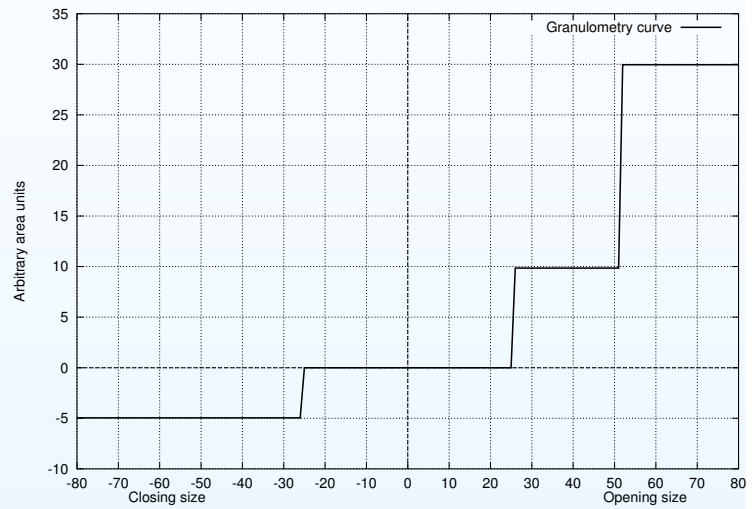
$$(1) \quad G_f(r) = \sum f - \sum \gamma_\lambda(f) \quad \lambda \in [0, 1, 2, \dots, R]$$

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



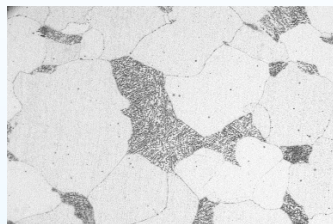
Image



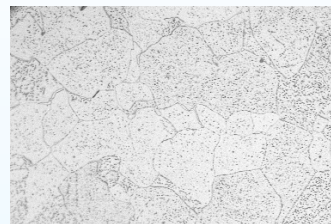
Courbe

Exemple d'application

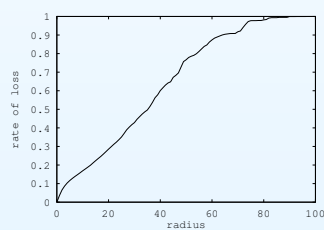
Étude du vieillissement de conduites de vapeur pour génération d'électricité.



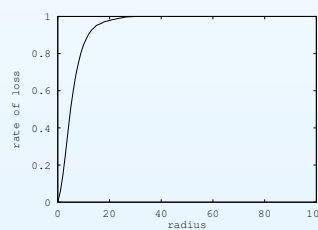
échantillon jeune



échantillon vieux



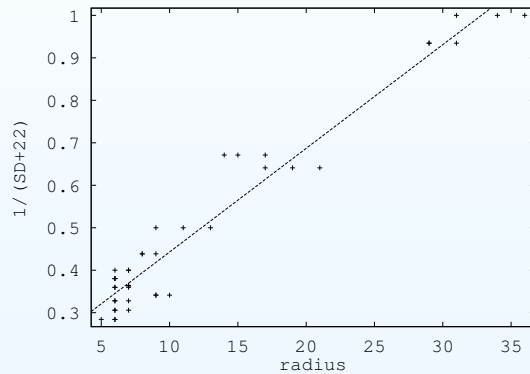
courbe jeune



courbe vieux

Indicateur de vieillissement

On prend comme mesure de vieillissement le paramètre λ pour lequel 50% du niveau d'idempotence a 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:

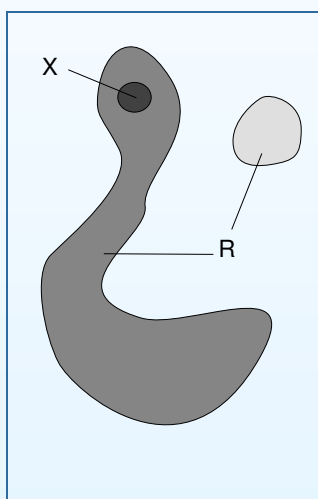
$$\delta_g^{(n)} = \delta_g^{(1)}[\delta_g^{(n-1)}]$$

Note: notion de distance géodésique, toujours plus grande que la distance non-géodésique, et peut-être infinie.

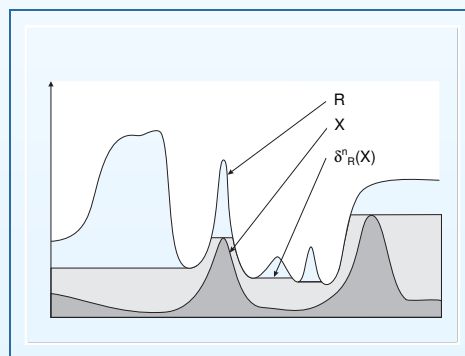
Reconstruction

La reconstruction est une opération morphologique itérée jusqu'à idempotence:

$$R_g(f) = \delta_g^{(i)}(f)$$



(a)



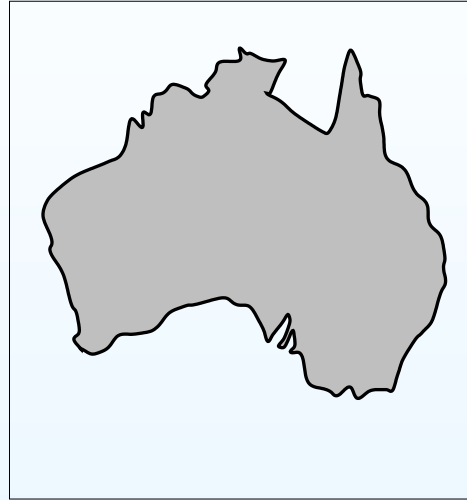
(b)

Exemple de reconstruction



Ouverture par disques

Notez qu'on a pas récupéré la Tasmanie.



Reconstruction

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?).

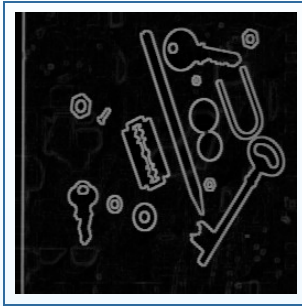
Résidus

Notions de résidus

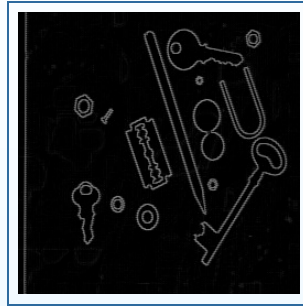
Lorsque toutes les propriétés vues auparavant deviennent utiles.

- 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.

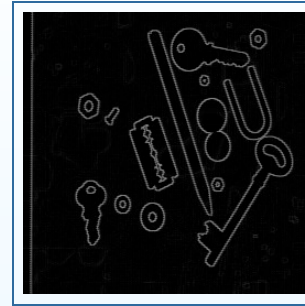
Exemples de gradients



$$\phi - \gamma$$



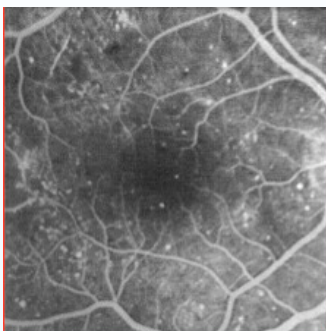
$$I - \gamma$$



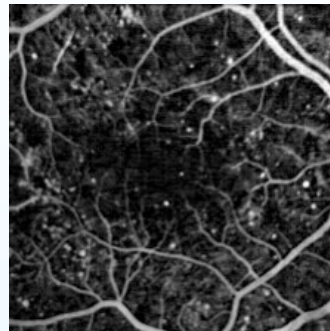
$$\phi - I$$

Le gradient morphologique ne se compose que de sa magnitude. On perd la direction.

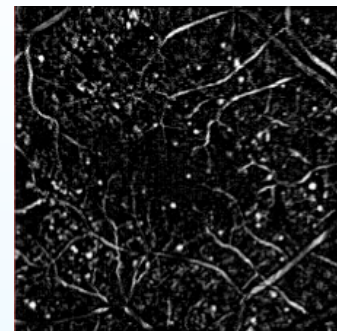
Exemples de chapeaux



Original



Chapeau ES carré



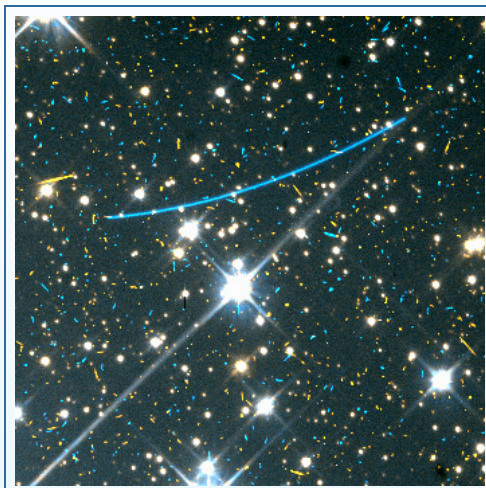
Chapeau par lignes

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

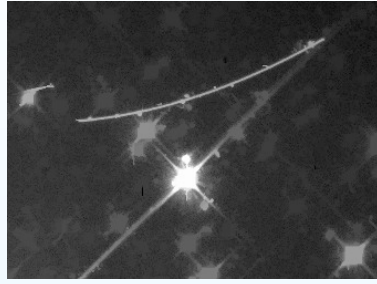
Une application : détection de trace d'astéroïdes



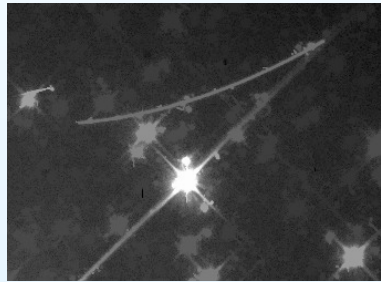
Traces d'astéroïdes, suite.



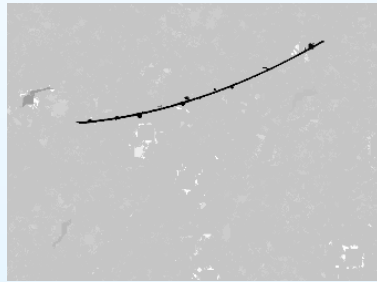
(traces)



(ouv. chemins)



(ouv. droites)



(Différence)

Cours de Morphologie Mathématique

Géodésie – Résidus

Hugues Talbot

talboth@esiee.fr

ISBS / ESIEE

2005

Géodésie – résidus – p.1/25

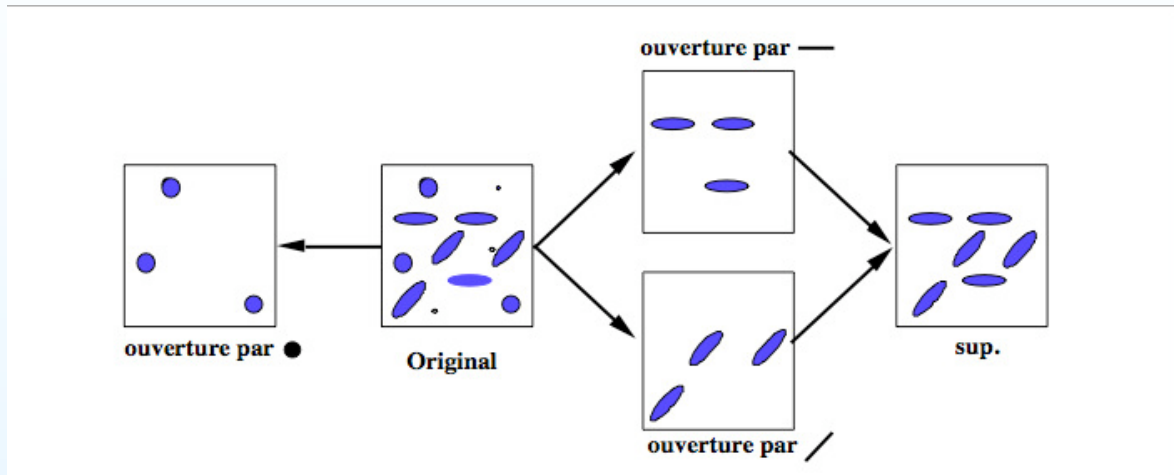
Rappel du cours précédent

- 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

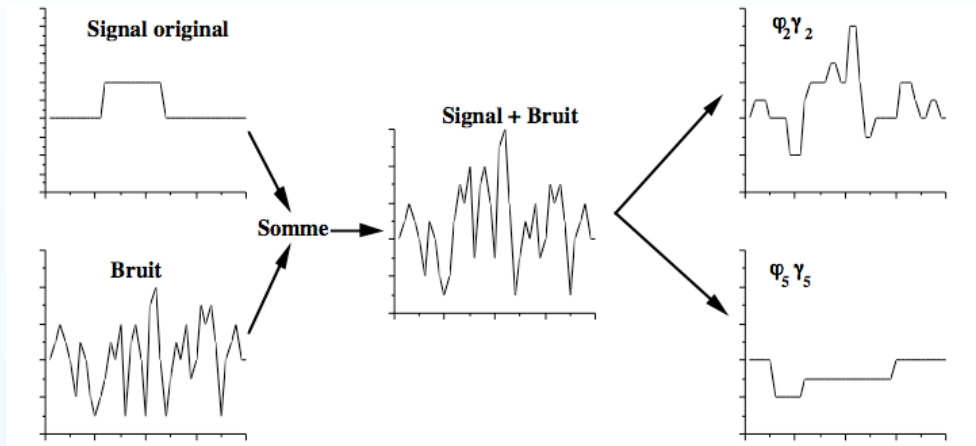
Illustration du filtrage morphologique simple:



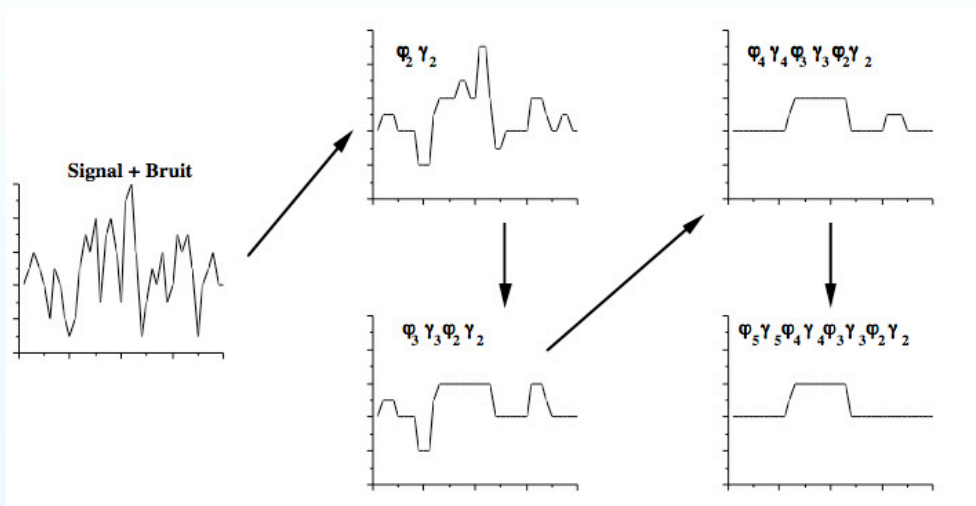
Filtres alternés séquentiels

- 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 λ_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 γ_λ , avec λ 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

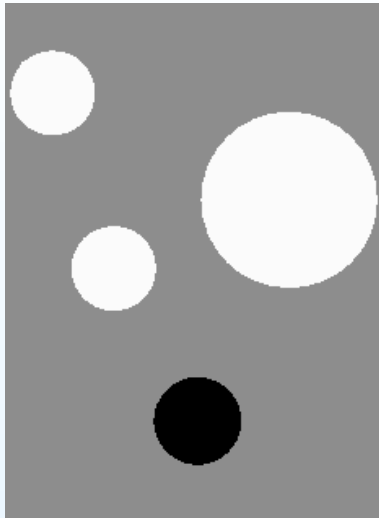
$$\gamma_\lambda \gamma_\mu = \gamma_\mu \gamma_\lambda = \gamma_{\vee\{\lambda, \mu\}}.$$

On construit la *courbe granulométrique* de la façon suivante, avec un paramètre *lambda* discret:

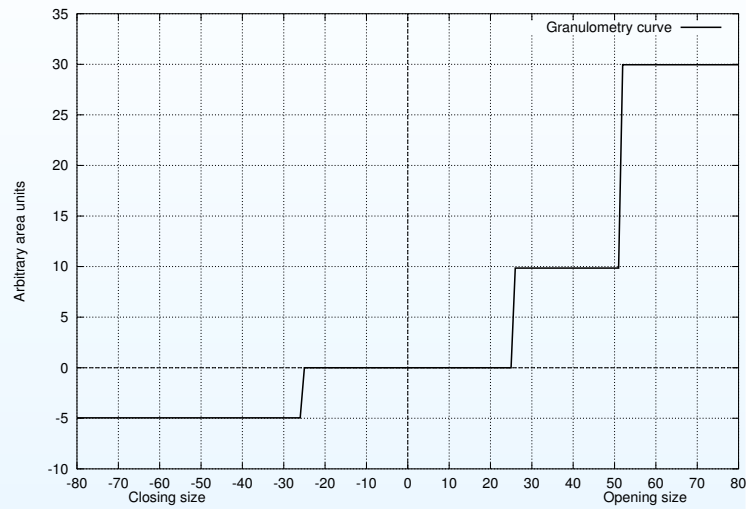
$$(1) \quad G_f(r) = \sum f - \sum \gamma_\lambda(f) \quad \lambda \in [0, 1, 2, \dots, R]$$

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



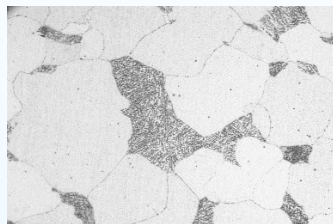
Image



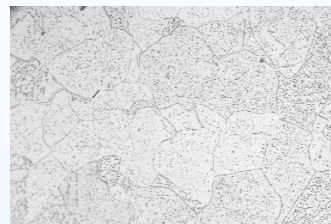
Courbe

Exemple d'application

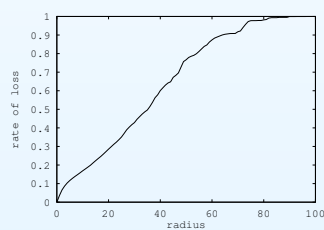
Étude du vieillissement de conduites de vapeur pour génération d'électricité.



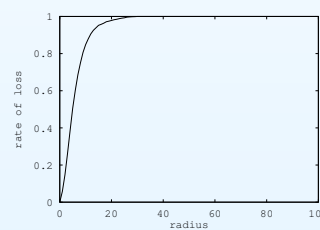
échantillon jeune



échantillon vieux



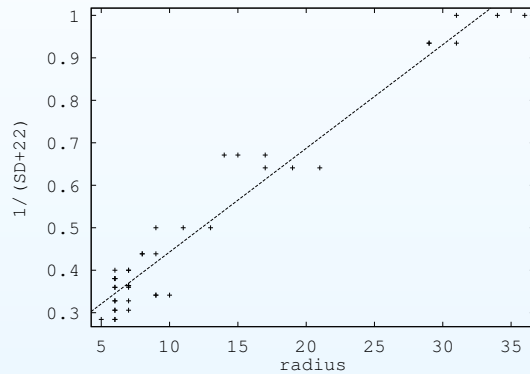
courbe jeune



courbe vieux

Indicateur de vieillissement

On prend comme mesure de vieillissement le paramètre λ pour lequel 50% du niveau d'idempotence a 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:

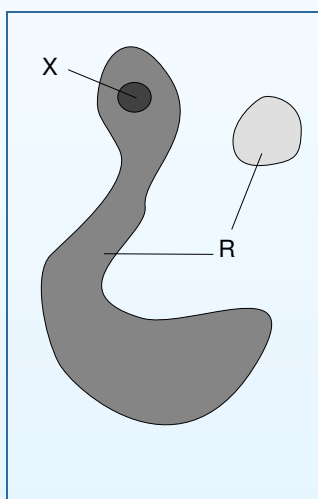
$$\delta_g^{(n)} = \delta_g^{(1)}[\delta_g^{(n-1)}]$$

Note: notion de distance géodésique, toujours plus grande que la distance non-géodésique, et peut-être infinie.

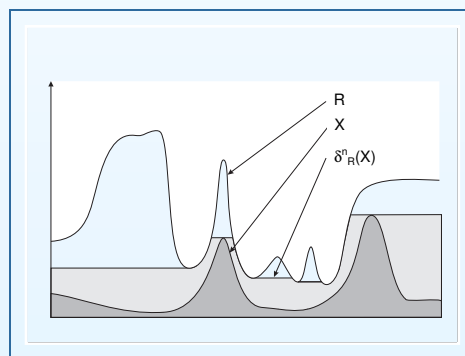
Reconstruction

La reconstruction est une opération morphologique itérée jusqu'à idempotence:

$$R_g(f) = \delta_g^{(i)}(f)$$

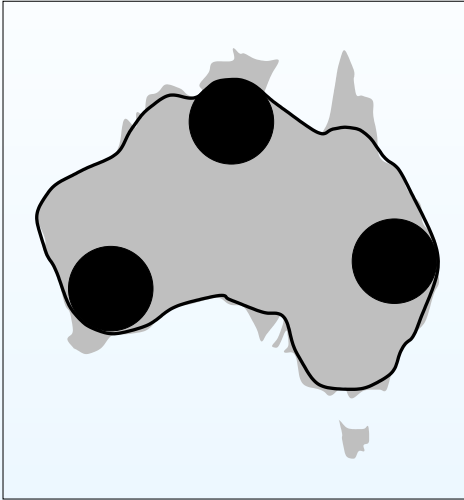


(a)

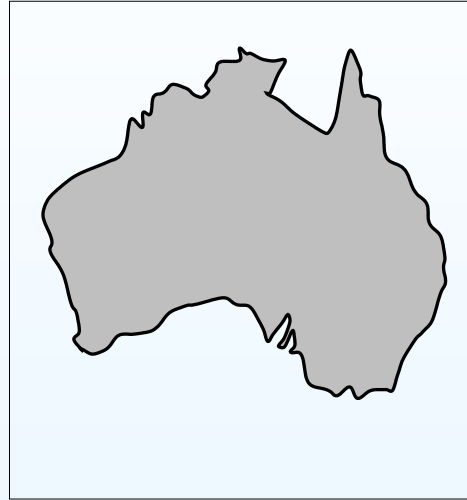


(b)

Exemple de reconstruction



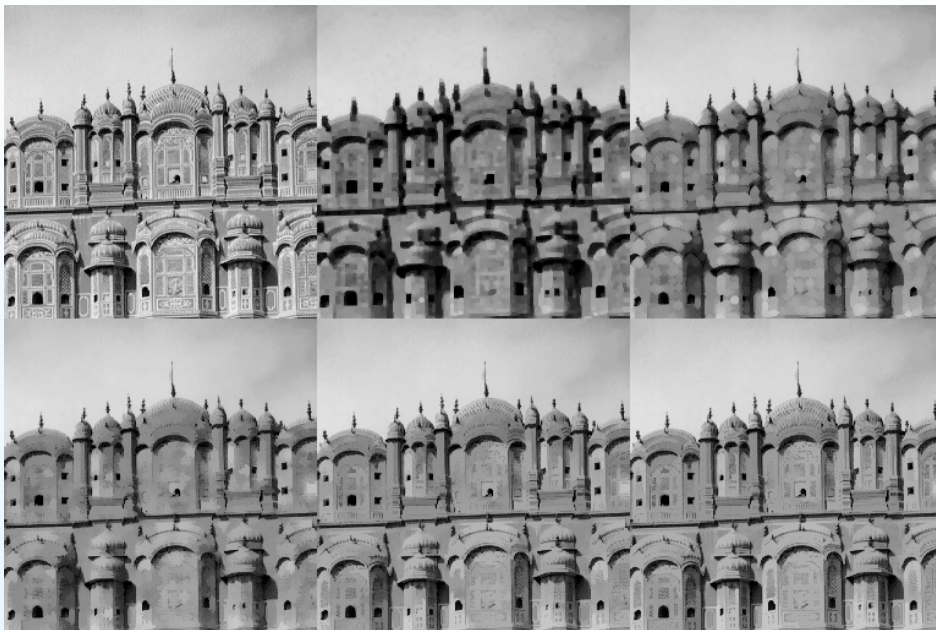
Ouverture par disques



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?).

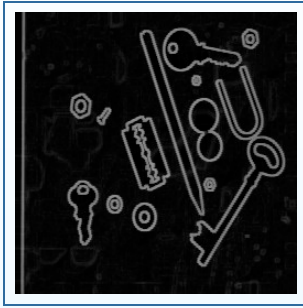
Résidus

Notions de résidus

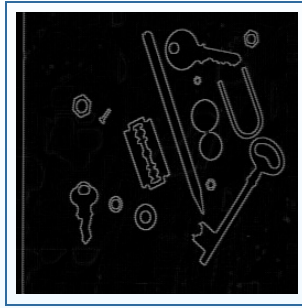
Lorsque toutes les propriétés vues auparavant deviennent utiles.

- 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.

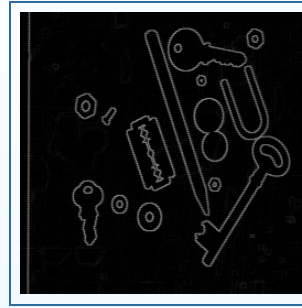
Exemples de gradients



$$\phi - \gamma$$



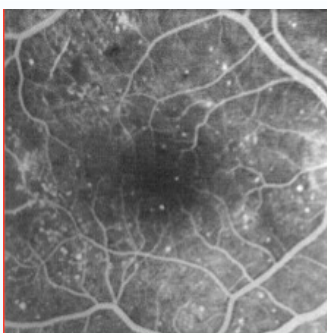
$$I - \gamma$$



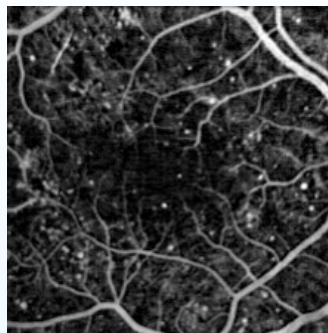
$$\phi - I$$

Le gradient morphologique ne se compose que de sa magnitude. On perd la direction.

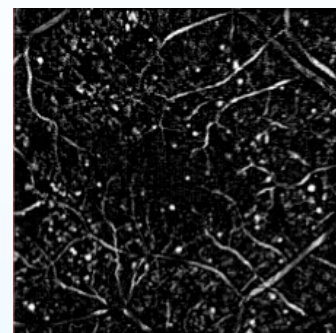
Exemples de chapeaux



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

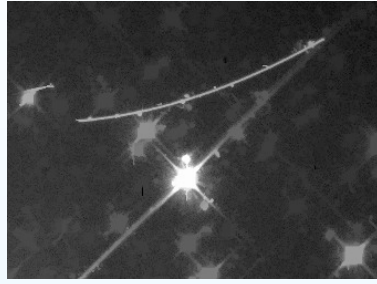
Une application : détection de trace d'astéroïdes



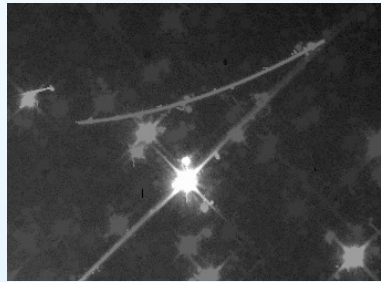
Traces d'astéroïdes, suite.



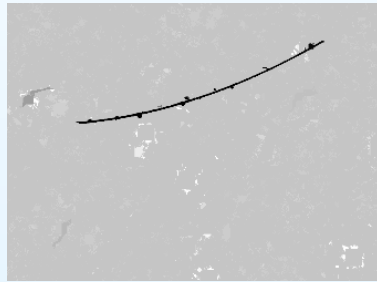
(traces)



(ouv. chemins)



(ouv. droites)



(Différence)

Cours de Morphologie Mathématique
Segmentation

Hugues Talbot
talboth@esiee.fr

ISBS / ESIEE

1^{er} semestre 2004-2005

Segmentation – p.1/38

TP1: fini?

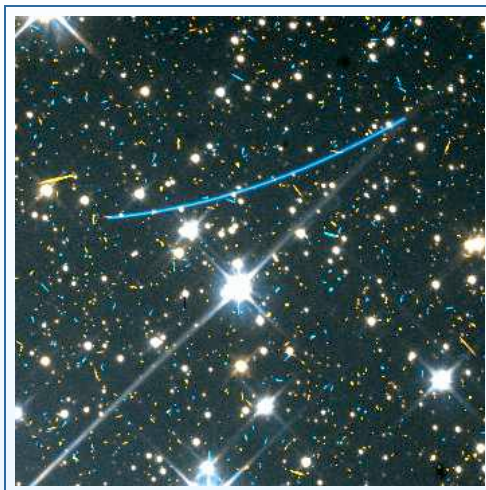
- Tout le monde a bien fini son rapport ?
- Il me manque:

Segmentation – p.2/38

Rappel du cours précédent

- Suite de la géodesie, en niveaux de gris.
- Résidus: gradients, top-hat (chapeau haut-de-forme).
- Érodés ultimes
- Squelettes.

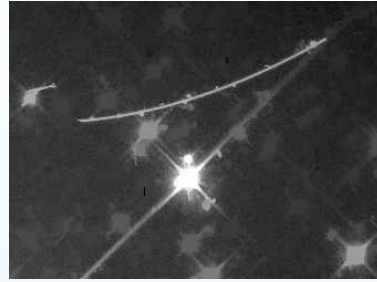
Une application : détection de trace d'astéroïdes



Traces d'astéroïdes, suite.



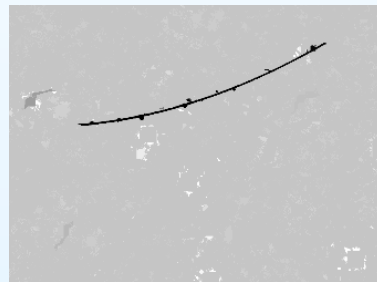
(traces)



(ouv. chemins)



(ouv. droites)



(Différence)

Images NASA/JPL/HST.

Segmentation – p.5/38

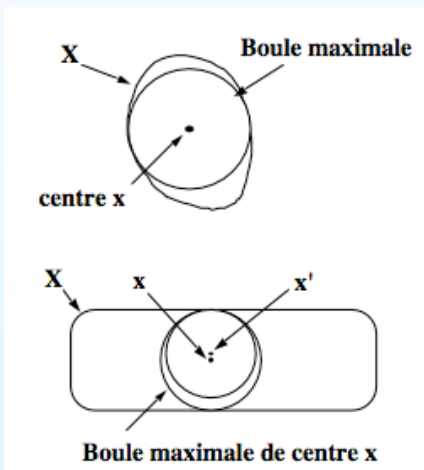
Squelettes et amincissements

Segmentation – p.6/38

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:

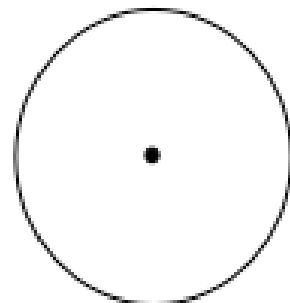
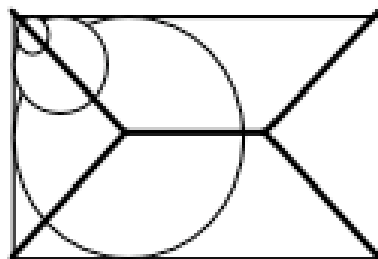
$$\delta_n(x) \in \delta_k(y) \in X$$



Segmentation – p.7/38

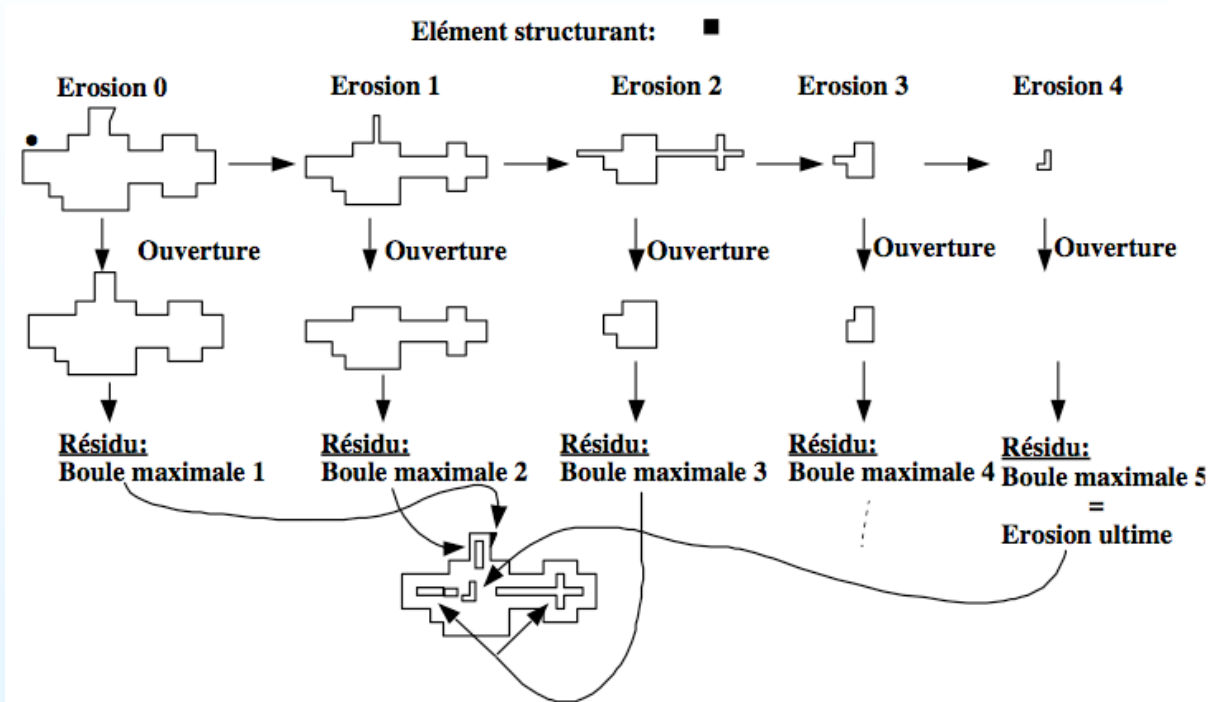
Squelettes

- L'érosion ultime était déjà un lieu de centres 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.



Segmentation – p.8/38

Construction



Segmentation – p.9/38

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.

Segmentation – p.10/38

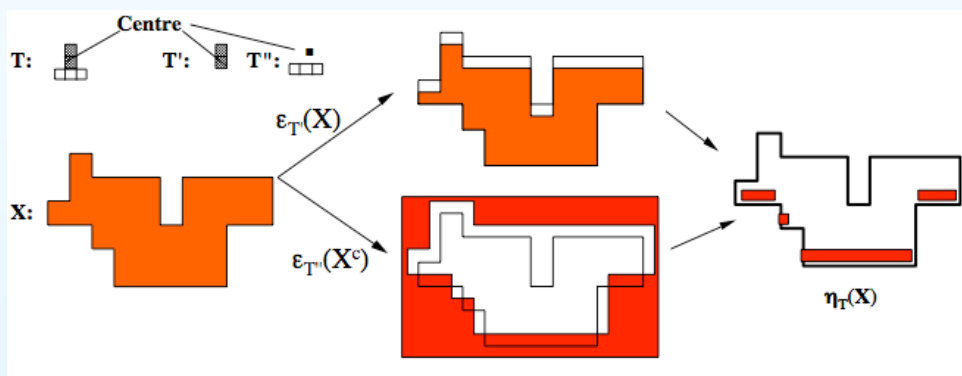
Squelette connexe

- 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.

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 $T = \{T', T''\}$.

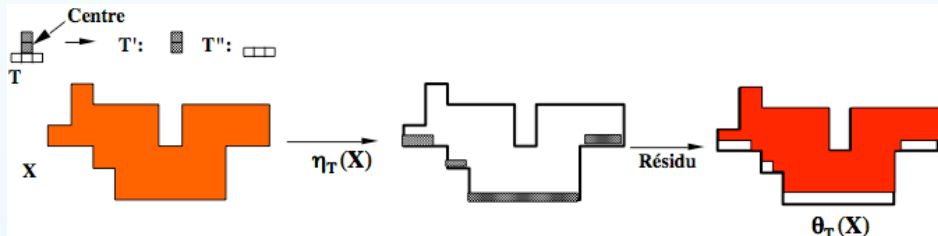
$$\eta_T(X) = \{z, T'' \subseteq X^c \text{ et } T' \subseteq X\} = \epsilon_{T'}(X) \cap \epsilon_{T''}(X^c)$$



Amincissements, épaissements

- L'amincissement θ_T est le résidu entre l'ensemble initial et sa transformée en tout-ou-rien:

$$\theta_T(X) = X \setminus \eta_T(X) = X \setminus [\epsilon_{T'}(X) \cap \epsilon_{T''}(X^c)]$$



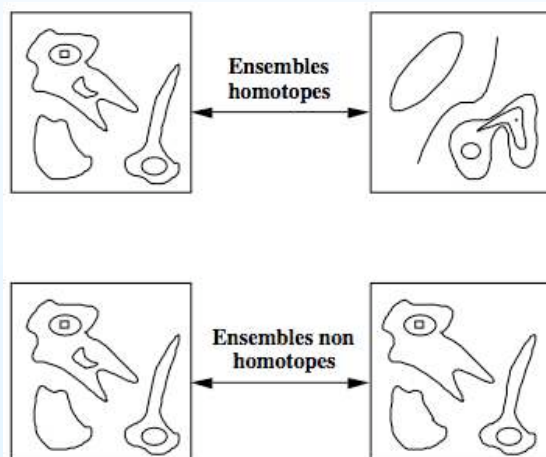
- L'épaissement est défini par dualité.

$$\xi_T(X) = X \cup \eta_T(X^c)$$

- On a: $\theta_T(X) = [\xi_T(X^c)]^c$.

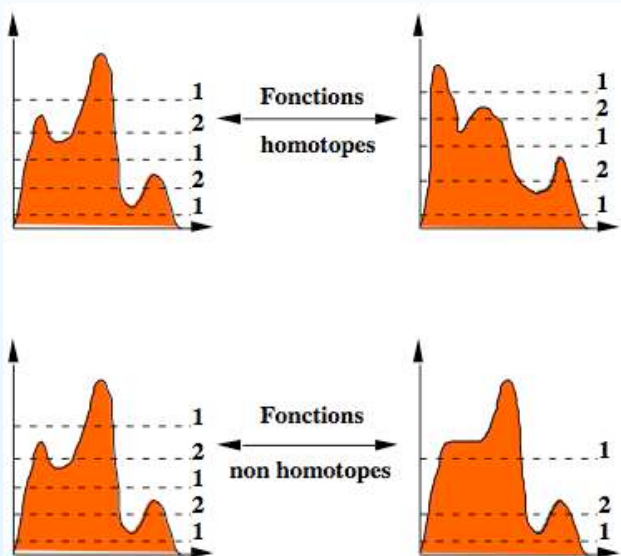
Homotopie pour les ensembles

- 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 contienne le même nombre de pores, et chaque pore contienne 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.



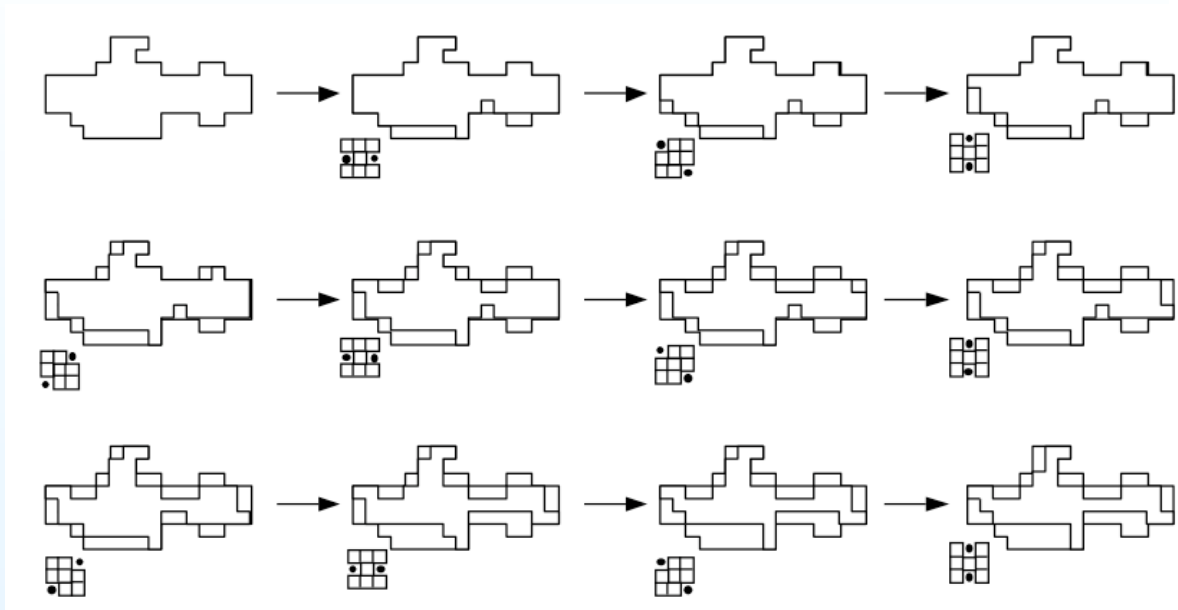
Segmentation – p.15/38

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.16/38

Amincissement séquentiel



Segmentation – p.17/38

Exercice (TD)

- Amincissement d'un rectangle 9×5
- Amincissement d'un rectangle 9×7 auquel on enlève le pixel du milieu.

Segmentation – p.18/38

segmentation

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.

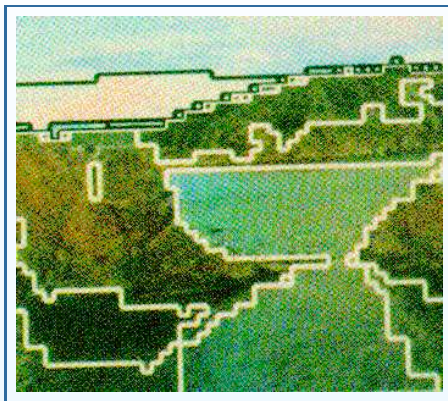
Méthodes de segmentation

Il existe de nombreuses méthodes:

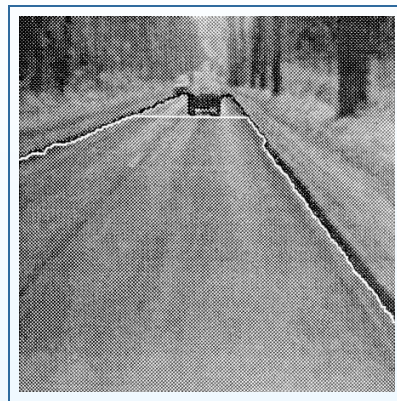
- Seuillage
- Méthodes à partir de l'histogramme d'une image
- Clustering spatial
- Découpage et unions de régions
- Contours actifs
- Lignes de niveaux
- À base de connaissance (IA)
- Méthodes morphologiques

Segmentation – p.21/38

Top-down ou bottom-up



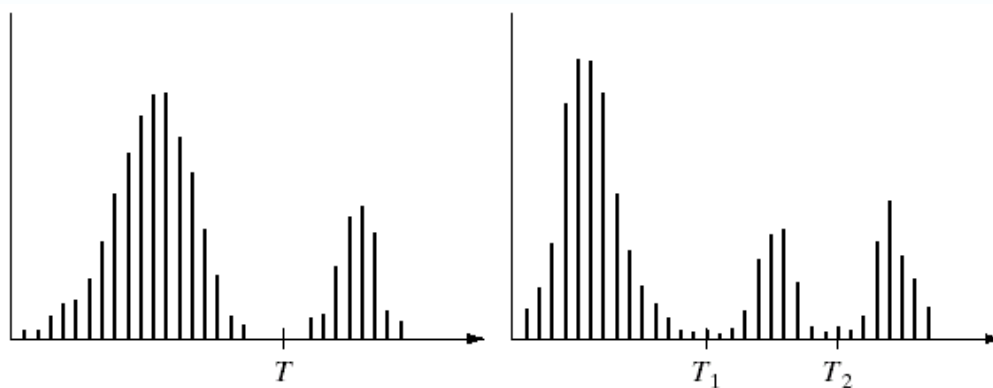
Top-down



Bottom-up

Segmentation – p.22/38

Segmentation au niveau du pixel

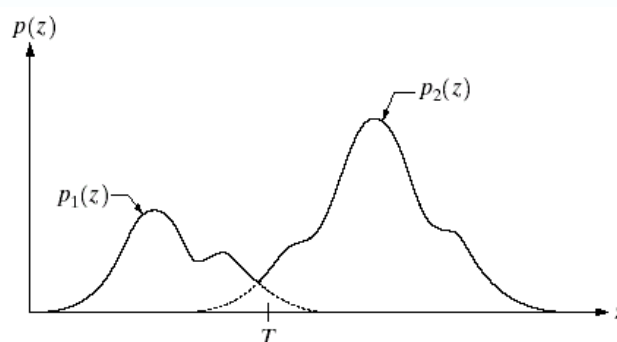


a b

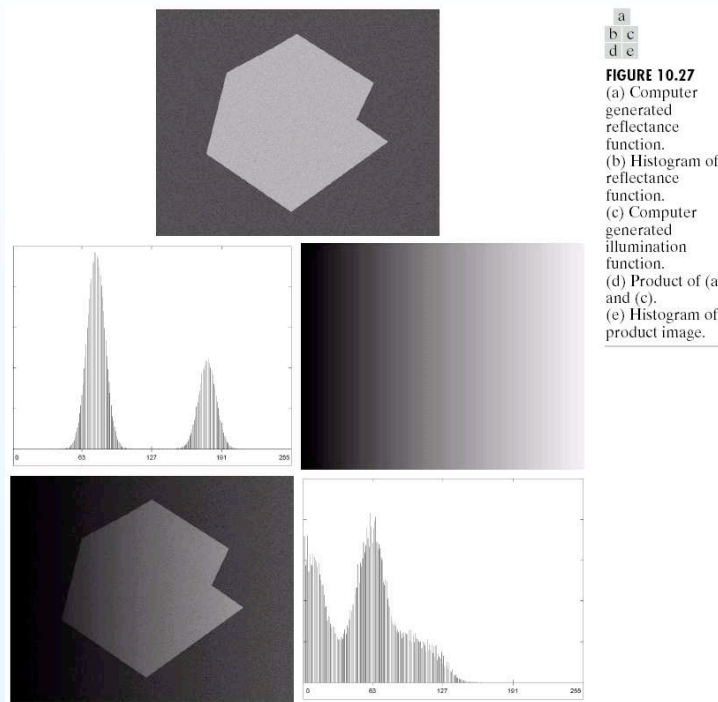
FIGURE 10.26 (a) Gray-level histograms that can be partitioned by (a) a single threshold, and (b) multiple thresholds.

Modèle

FIGURE 10.32
Gray-level probability density functions of two regions in an image.

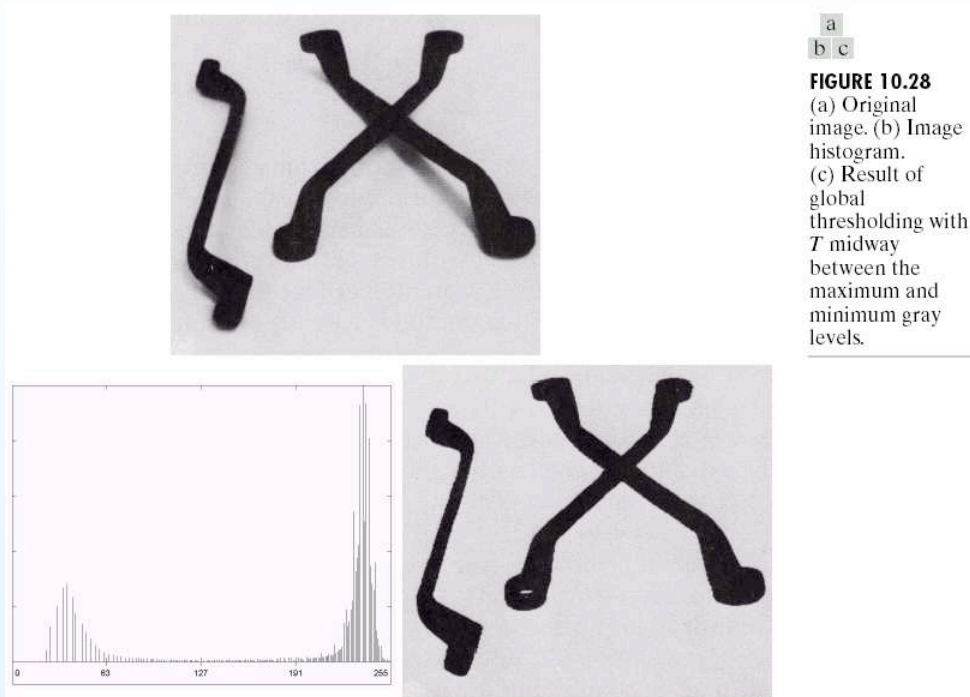


Modèle (suite)



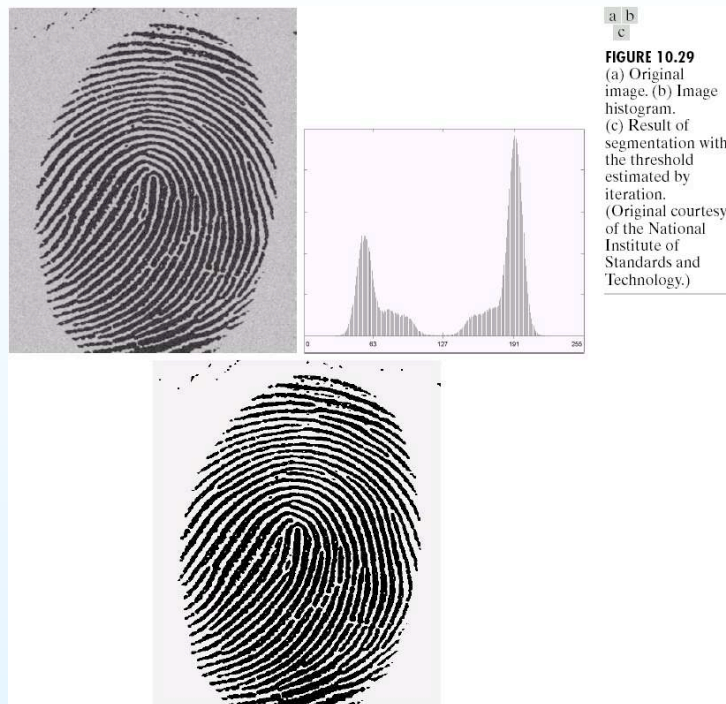
Segmentation – p.25/38

Seuillage



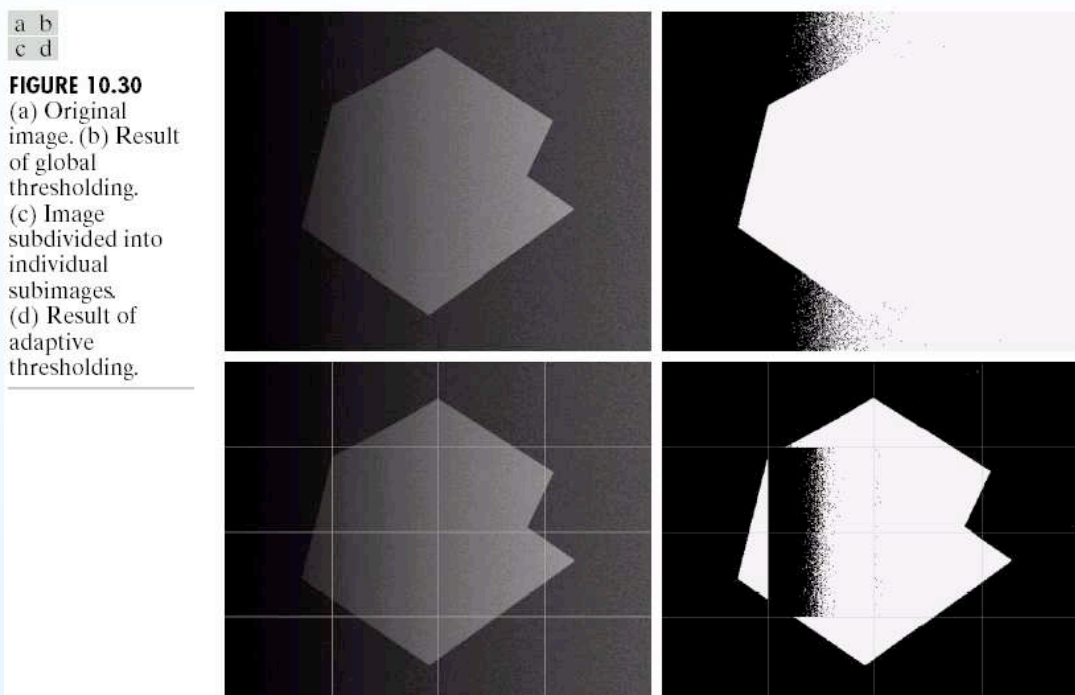
Segmentation – p.26/38

Exemple de seuillage



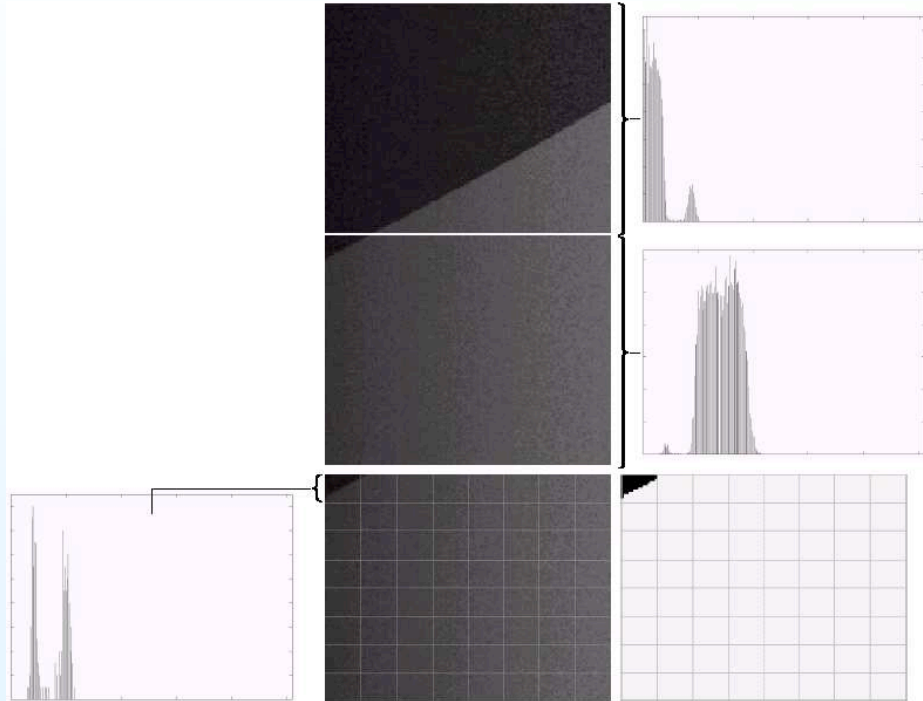
Segmentation – p.27/38

Seuillage adaptatif



Segmentation – p.28/38

Seuillage adaptatif (suite)

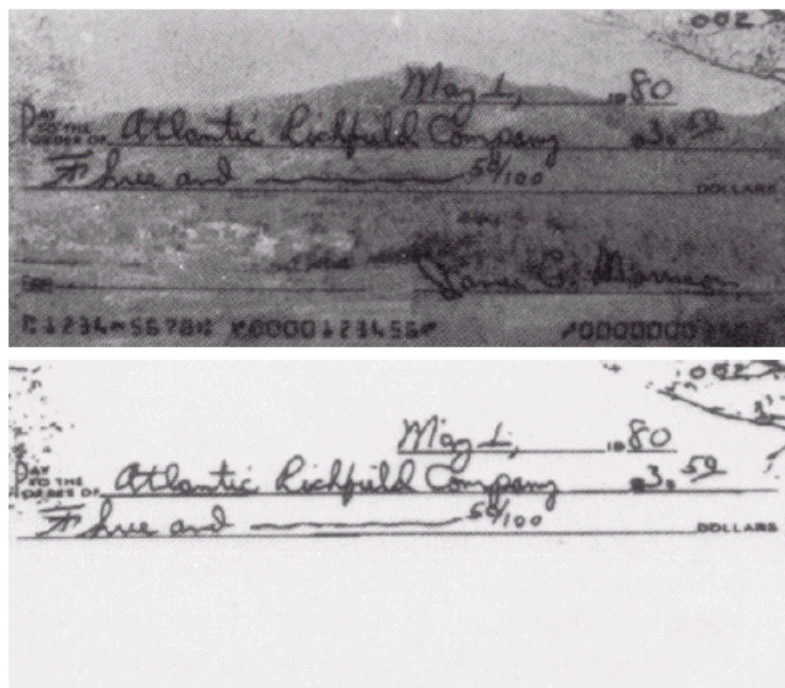


Segmentation – p.29/38

Seuillage adaptatif (suite)

a
b

FIGURE 10.37
(a) Original image. (b) Image segmented by local thresholding. (Courtesy of IBM Corporation.)



Segmentation – p.30/38

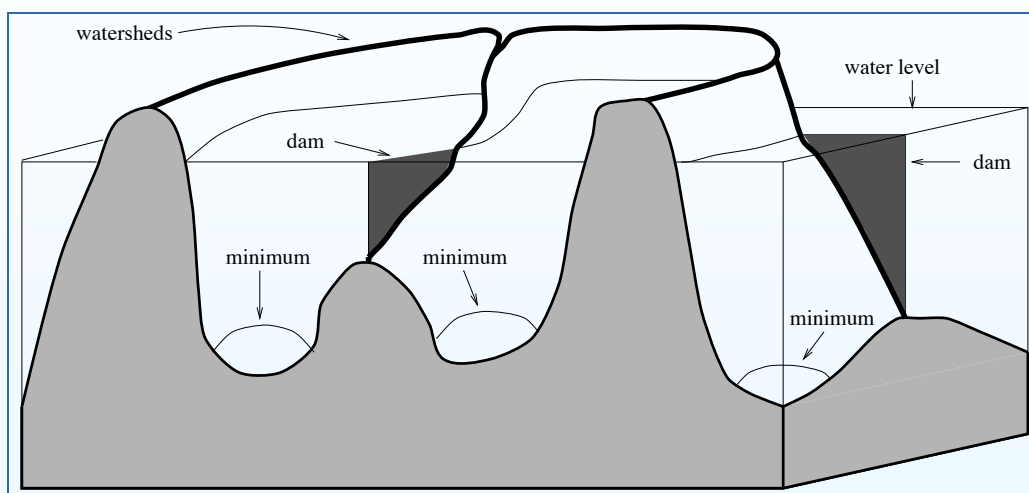
Seuillage multispectral



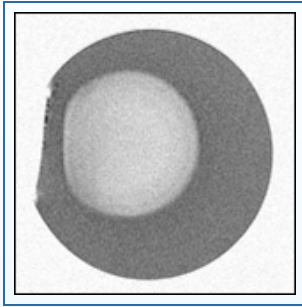
a b c

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.

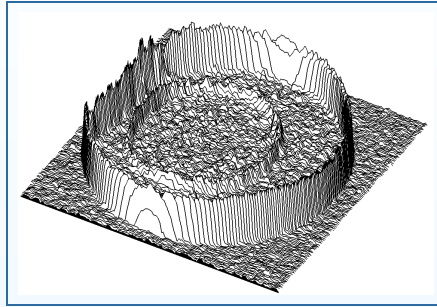
La ligne de partage des eaux



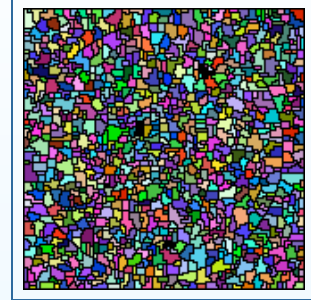
The watershed line 2



Original

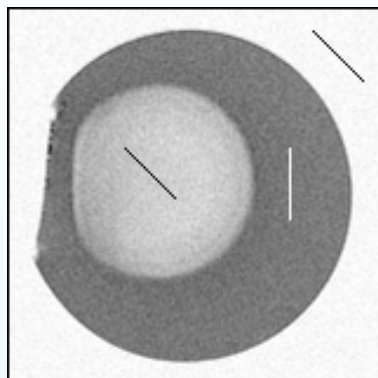


Gradient

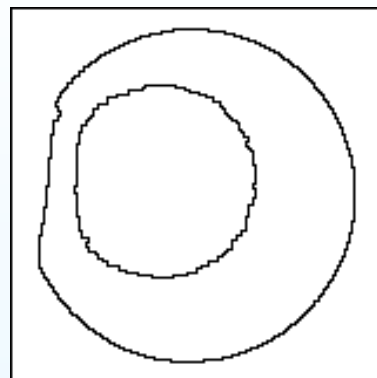


First result

The watershed line 3



Orig. + Markers



Final

How to use the watershed line?

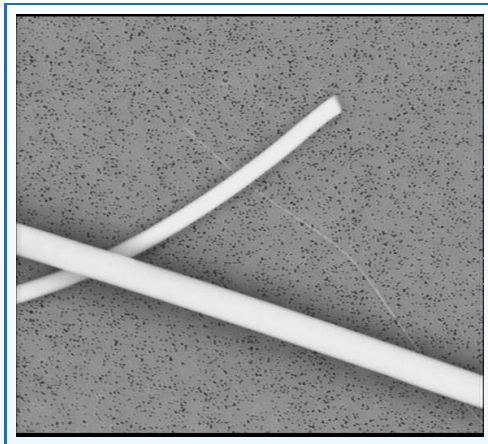
The classical CMM approach:

- Find a good gradient
- Find good external markers
- Find good internal markers

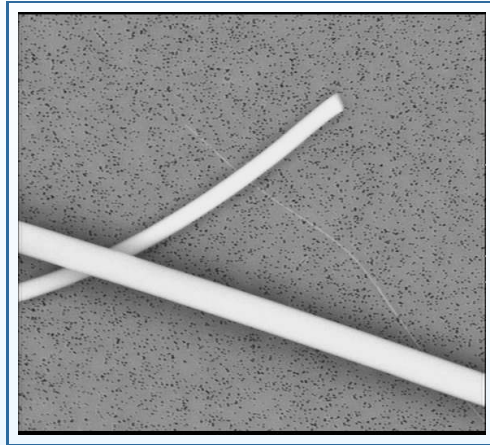
Other approaches:

- Merge catchment basins afterwards (graph approach f.e.)
- Constrain the watershed

Example of the classical approach (1)

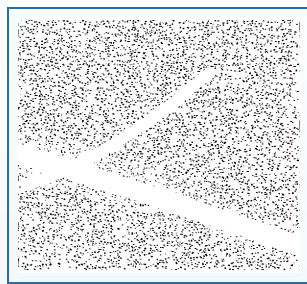


Example of the classical approach (1)

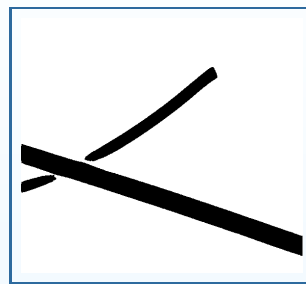


Segmentation – p.37/38

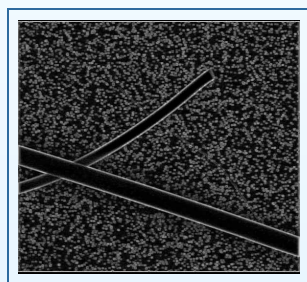
Examples of the classical approach (2)



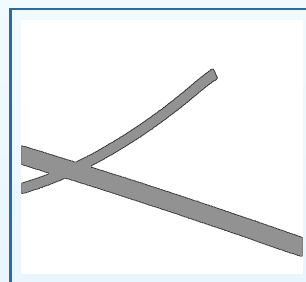
internal



external



gradient



final

Segmentation – p.38/38