



# Segmentation en régions

Caroline Petitjean

# Segmentation Methods

- Region Growing
- Region Splitting
- Region Splitting and Merging
- Autres techniques de segmentation contours

# Définitions

## Définition de la segmentation

- Ensemble de zones homogènes dans l'image (homogénéité de texture, de couleur, de mouvement...).
- Partition de l'image  $I$  en sous-ensembles disjoints non-vides  $R_i$  pour  $i = 1, 2, \dots, M$  appelées régions contenant des pixels connexes au sens d'un prédicat et tels que  $I = \bigcup_{i=1}^M R_i$
- Description en objets : simplification de l'image.

## Région

- Ensemble de pixels topologiquement connexes et ayant des attributs similaires : niveaux de gris, couleur, texture, mouvement.
- Une région souvent différente d'un objet de la scène

## Remarques

- Dualité contours/régions : intérieur d'un contour fermé = région
- Segmentation des contours  $\Rightarrow$  voir cours sur les contours
- Attributs utilisés pour la segmentation très variés (niveaux de gris, couleur, texture, mouvement)  $\Rightarrow$  dans ce cours les méthodes utilisent les niveaux de gris, mais peuvent être étendues à d'autres attributs.

# Segmentation



A quoi sert la segmentation en régions ?

Important prérequis pour les étapes de mesure, de compréhension de la scène :

- reconnaissance d'objets
- indexation : rechercher dans une base d'images, les images "ressemblantes" à une image initiale
- compression
- recalage d'images, mises en correspondance

# Segmentation

Quelques exemples d'applications

Exemple de Segmentation de visages

Applications en indexation, biométrie.



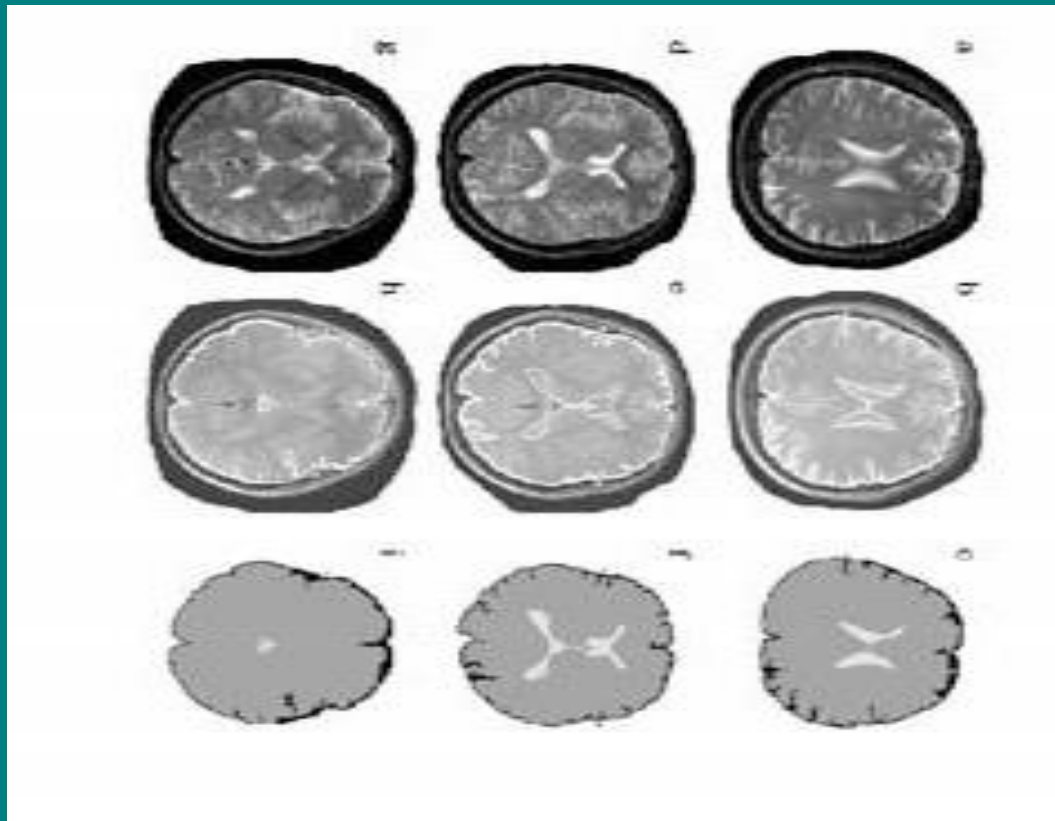
LIRIS Laboratoire d'InfoRmatique en Images et Systèmes d'information

UMR 5205 CNRS/INSA de Lyon/Université Claude Bernard Lyon 1/Université Lumière Lyon

2/Ecole Centrale de Lyon

# Segmentation

Quelques exemples d'applications - Imagerie médicale IRM  
Étude des maladies neurologiques qui altèrent la distribution entre le parenchyme du cerveau (matière grise) et liquide céphalo-rachidien.

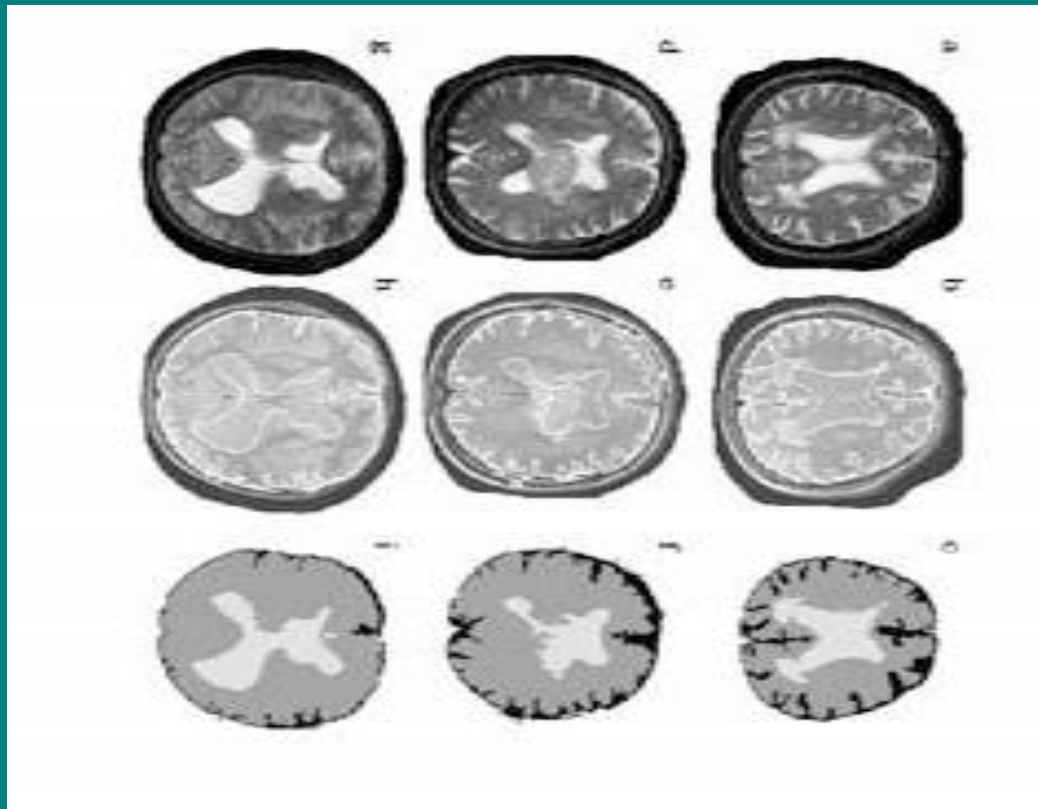


*Segmentation of Brain Parenchyma and Cerebrospinal Fluid in Multispectral Magnetic Resonance*  
(Arvid Lundervold and Geir Storvik)

Published in IEEE Transactions on Medical Imaging, Vol. 14, No. 2, June 1995, pp. 339-349.

# Segmentation

Quelques exemples d'applications - Imagerie médicale IRM : cas pathogènes  
Étude des maladies neurologiques qui altèrent la distribution entre le parenchyme du cerveau (matière grise) et liquide céphalo-rachidien.



Segmentation of Brain Parenchyma and Cerebrospinal Fluid in Multispectral Magnetic Resonance  
(Arvid Lundervold and Geir Storvik)  
Published in IEEE Transactions on Medical Imaging, Vol. 14, No. 2, June 1995, pp. 339-349.

# Définitions

## Prédicat $\mathcal{P}$

Proposition logique : 1 si la propriété est VRAIE, 0 si elle est fausse

## Prédicat d'homogénéité

ou prédicat d'uniformité : mesure l'uniformité d'une région.

**Exemple :**

1 si variance d'intensité  $<$  seuil

0 sinon

## Partition

Ensemble de régions  $\mathcal{R}_i$  tel que :

$$\left\{ \begin{array}{l} \forall i, j (\mathcal{R}_i \cap \mathcal{R}_j) = \text{ensemble vide} \\ \cup \mathcal{R}_i = \text{image} \\ \forall i : \mathcal{R}_i \neq \text{ensemble vide} \end{array} \right.$$



## Méthodes globales

- Classification puis étiquetage en composantes connexes
  - Classification à partir d'histogrammes
  - Méthodes de *clustering* (k-means, c-means)
- À partir d'une extraction de contours

## Méthodes locales

Se basent sur les notions de prédicats et de partitions

- Ascendantes (bottom-up) : croissance de régions
- Descendantes (top-down) : division-fusion

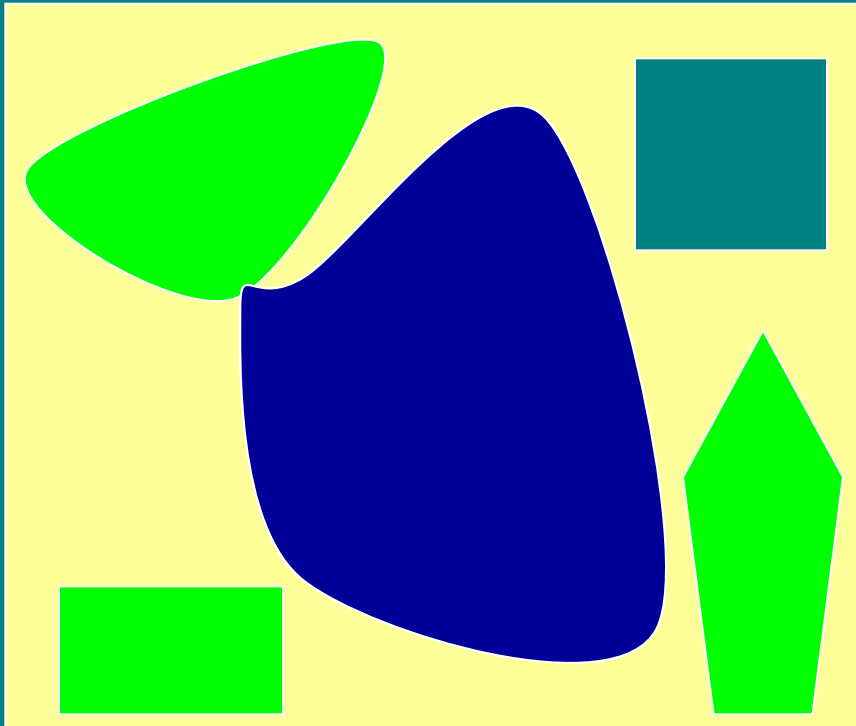
## Méthodes mixtes régions-contours

- Ligne de partage des eaux
- Coopération région-contours

# Croissance de régions

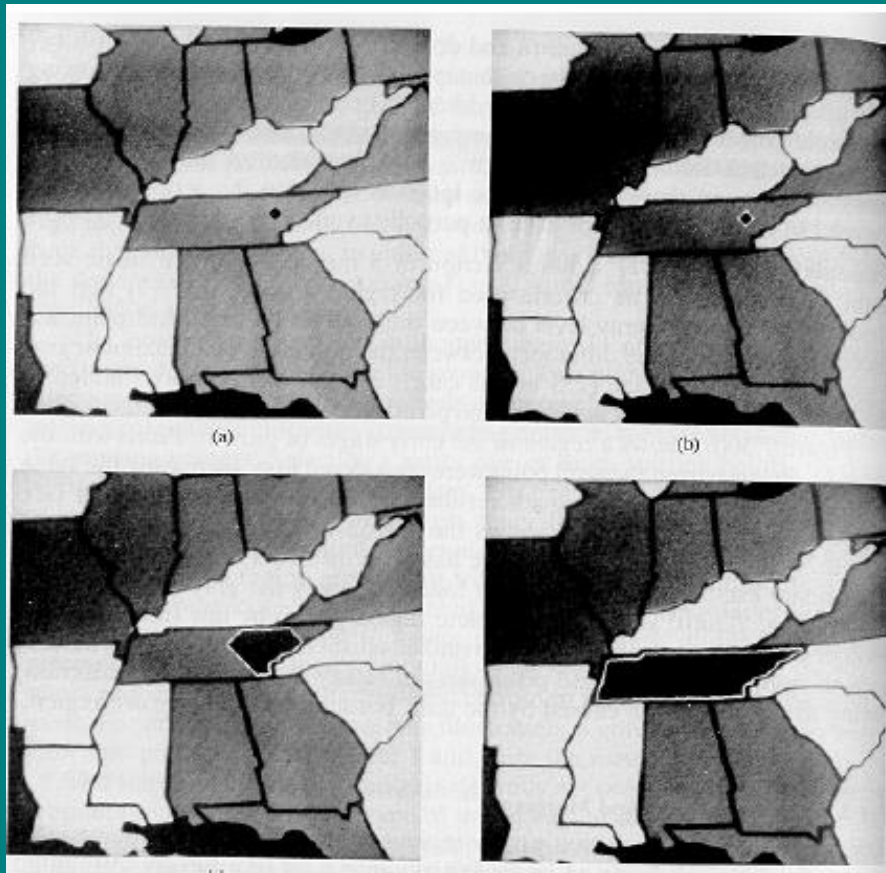
## Principe général

- Approche ascendante
- Départ d'un pixel (ou d'un groupe de pixels) d'amorce (pixel **germe**)
- Analyse de ses pixels voisins et analyse du **critère d'homogénéité**  $\mathcal{P}$
- Croissance de la région jusqu'au **critère d'arrêt** (plus aucun pixel ne satisfait le critère)



# Region Growing

8 neighbors, predicate:  $|z - z_{seed}| < 0.1(\max_z - \min_z)$



Multiple regions  
can be grown in  
parallel using  
multiple seeds

# Region Growing

- **How do we choose the similarity criteria (predicate)?**
  - The homogeneity predicate can be based on any characteristic of the regions in the image such as:
    - average intensity
    - variance
    - color
    - texture

# Croissance de région

## Avantages

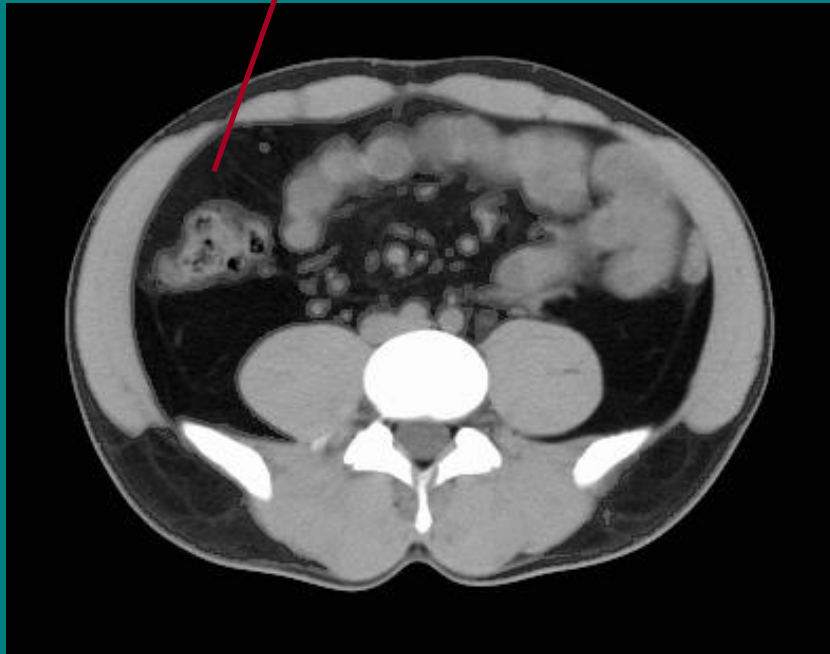
- Rapide
- Facile à mettre en œuvre

## Inconvénients

- Algorithme très sensible au bruit
- Obtention de frontières non-régulières
- Peu efficace dans le cas de dégradés
- L'ordre du traitement des pixels germes peut avoir une influence sur le résultat
- Choix critique de la valeur des seuils
  - Risque de sur-segmentation, sous-segmentation
  - Dépend du type d'images à traiter

# Croissance de régions

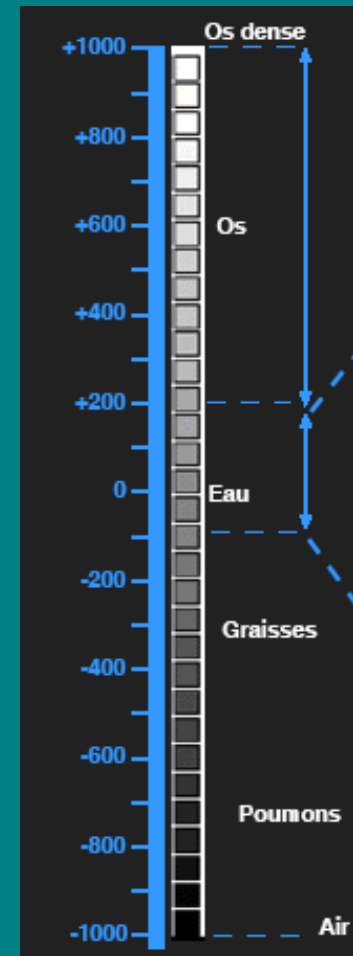
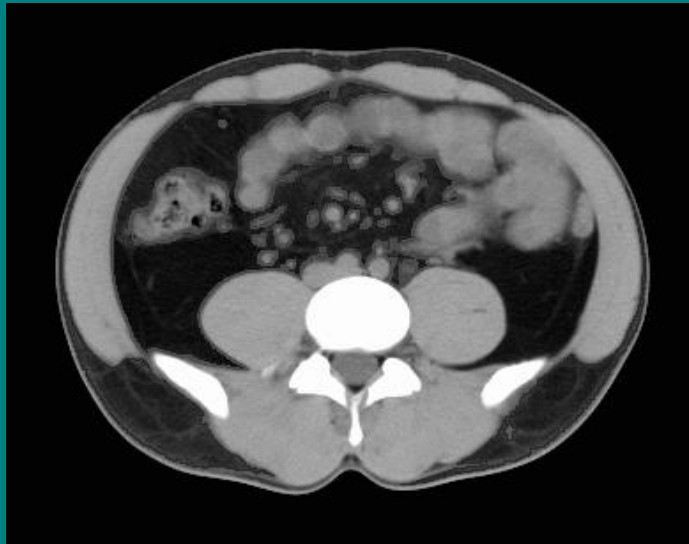
Segmentation de la graisse sous-cutanée et viscérale sur des images scanner



# Croissance de régions

- Image scanner codées entre -1000 et +1000

Prédicat d'homogénéité Graisse : -120 à -60 HU

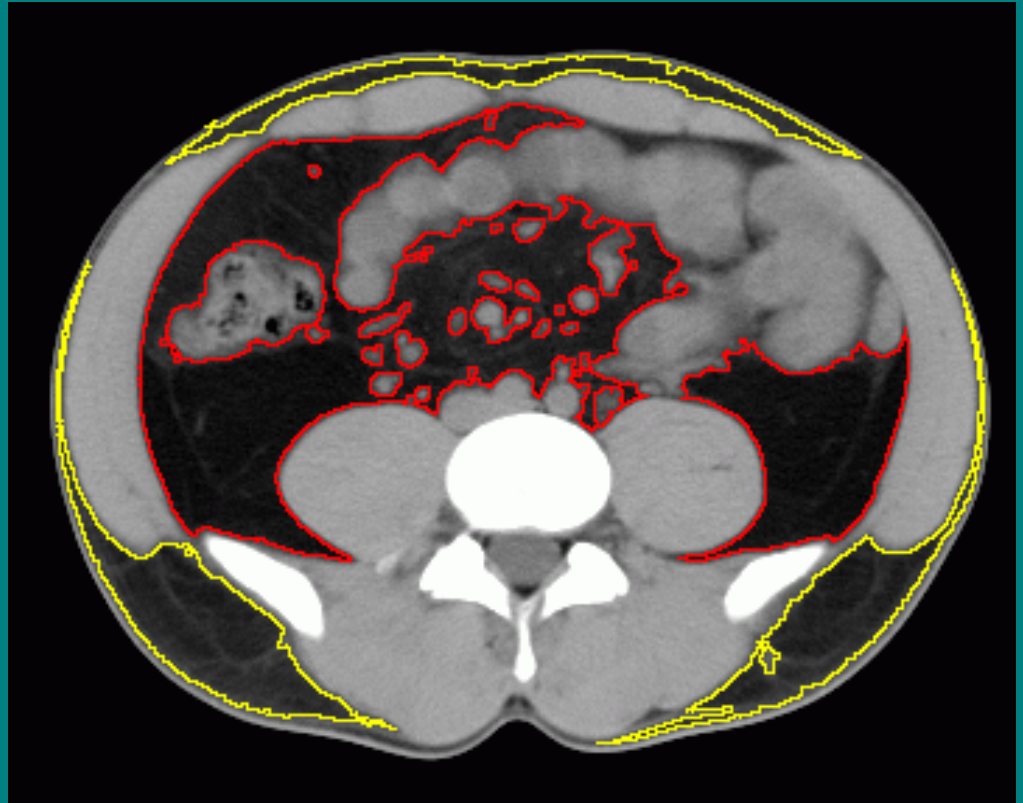


# Croissance de régions

- Segmentation par croissance de région selon un critère de seuillage

A partir du germe :

pixel  $\in$  région  
si son intensité  $\in$   
[-120,-60]





# Segmentation Methods

- Region Growing
- Region Splitting
- Region Splitting and Merging
- Autres techniques de segmentation contours

# Algorithme de division

## Principe

Au départ : l'image entière sur laquelle le prédicat d'homogénéité  $P$  n'est pas respecté en général

Si le prédicat n'est pas respecté : division en régions

Chaque région est examinée à son tour, etc, jusqu'à arriver au niveau du pixel

▫

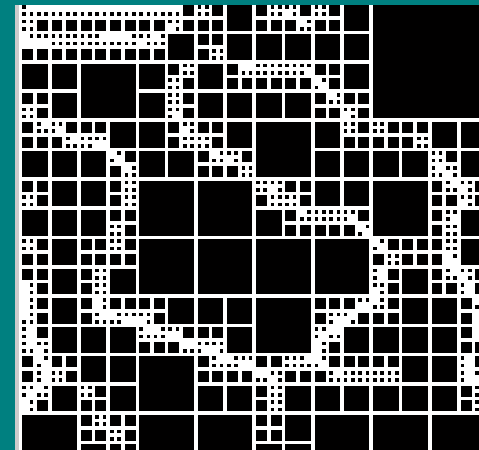
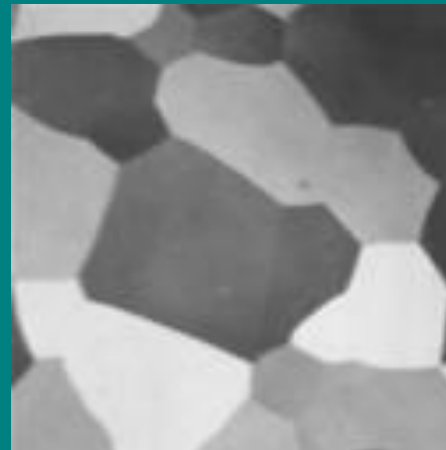
## Paramètres

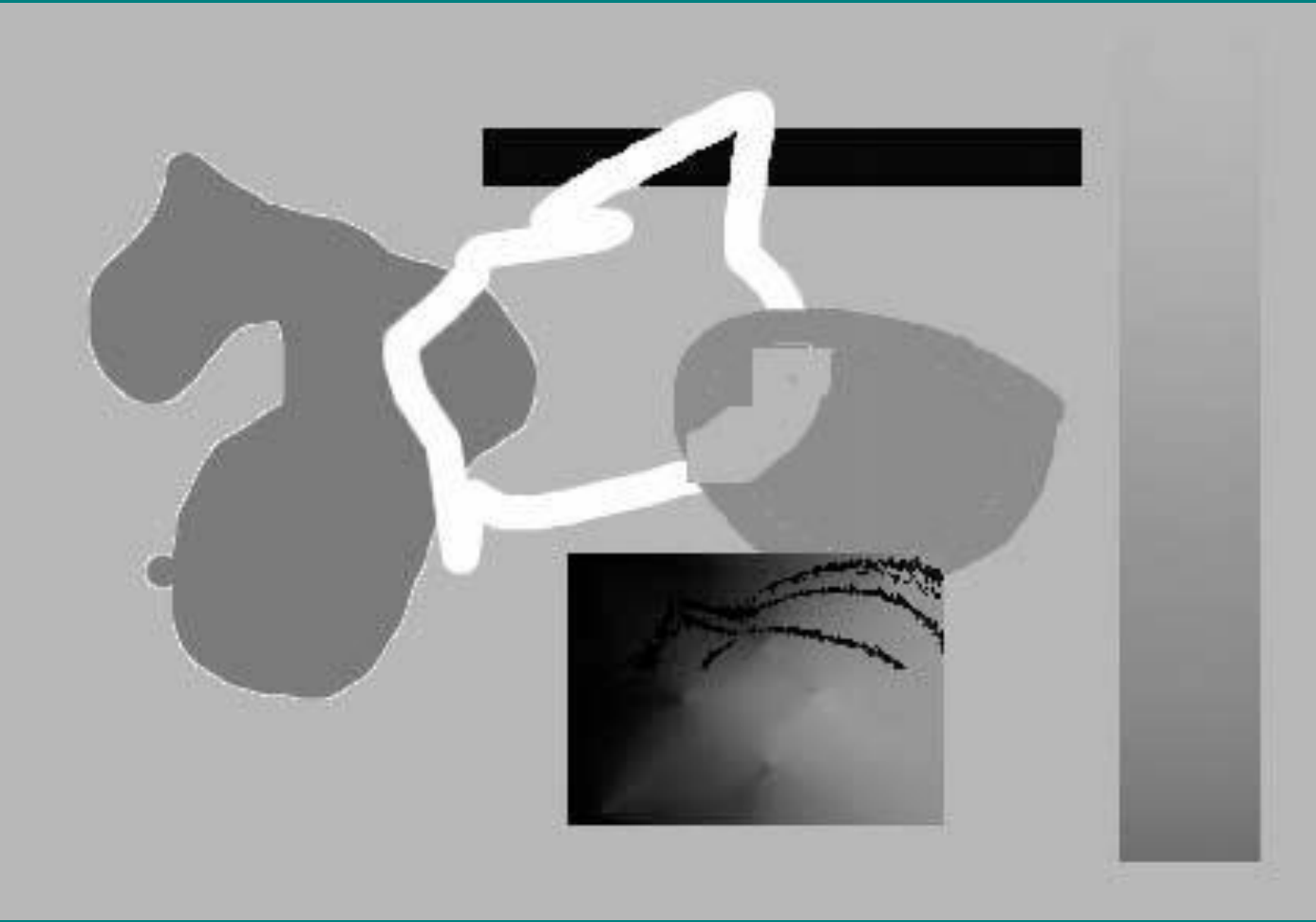
- Critère d'hétérogénéité utilisé
- Forme des régions : rectangulaire, carrée, hexagonale, Voronoï...
- Type de décomposition : par 4, 6, etc

▫

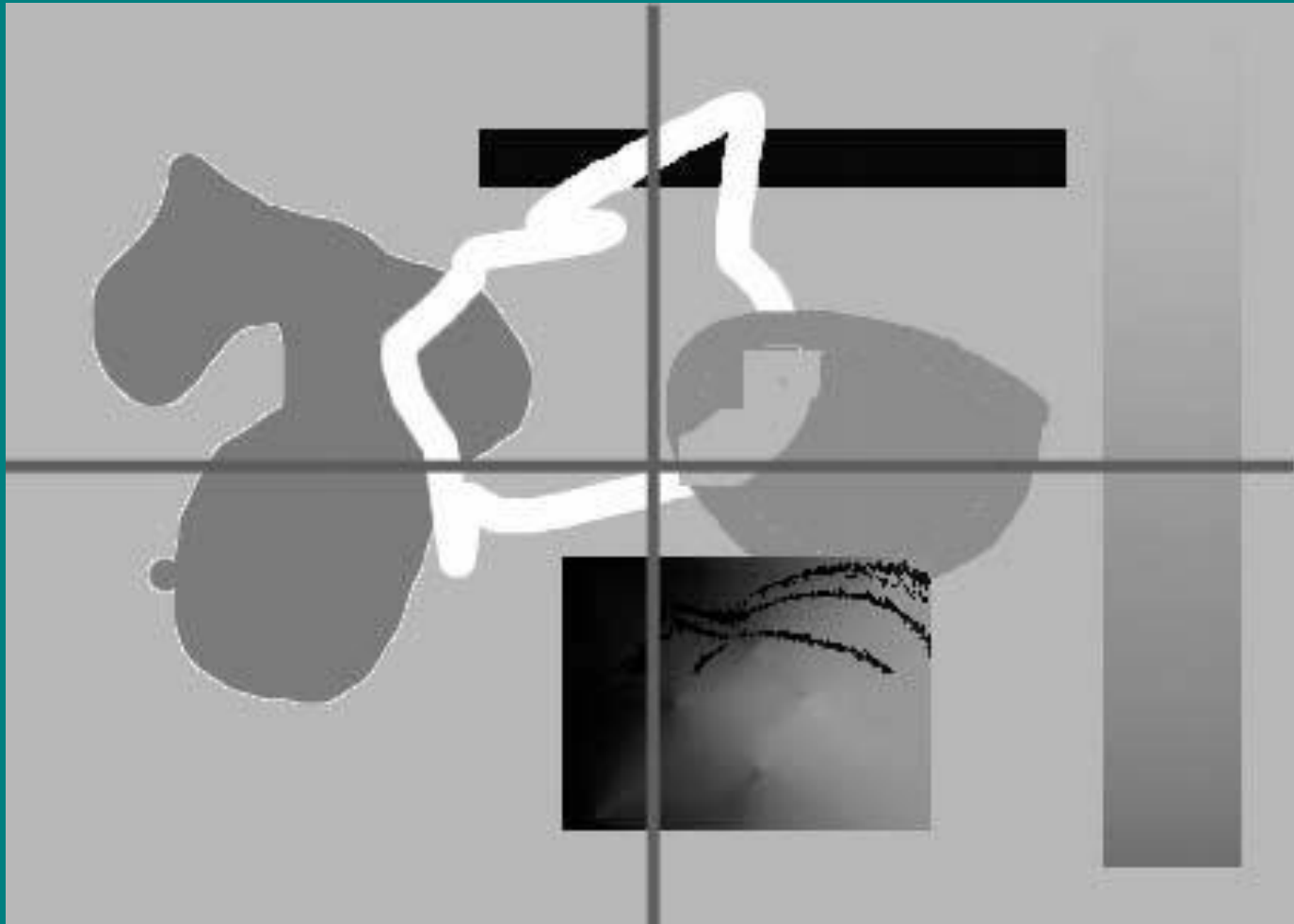
## Prédicat d'hétérogénéité inter-régions

▫  $\forall (R_u, R_v)$  2 régions adjacentes  
 $\Rightarrow P(R_u \cap R_v) = \text{FAUX}$





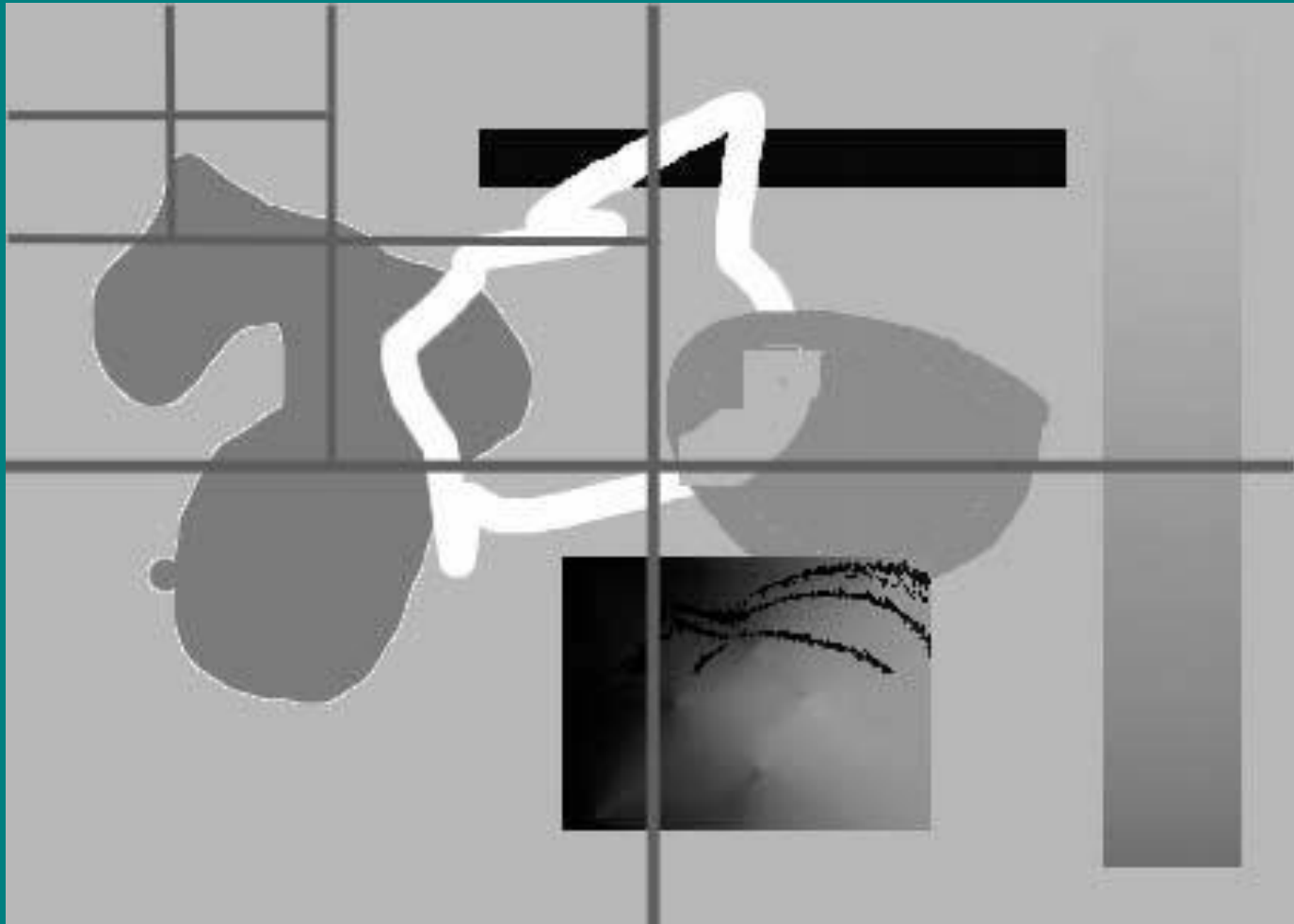
1<sup>ère</sup> étape : division



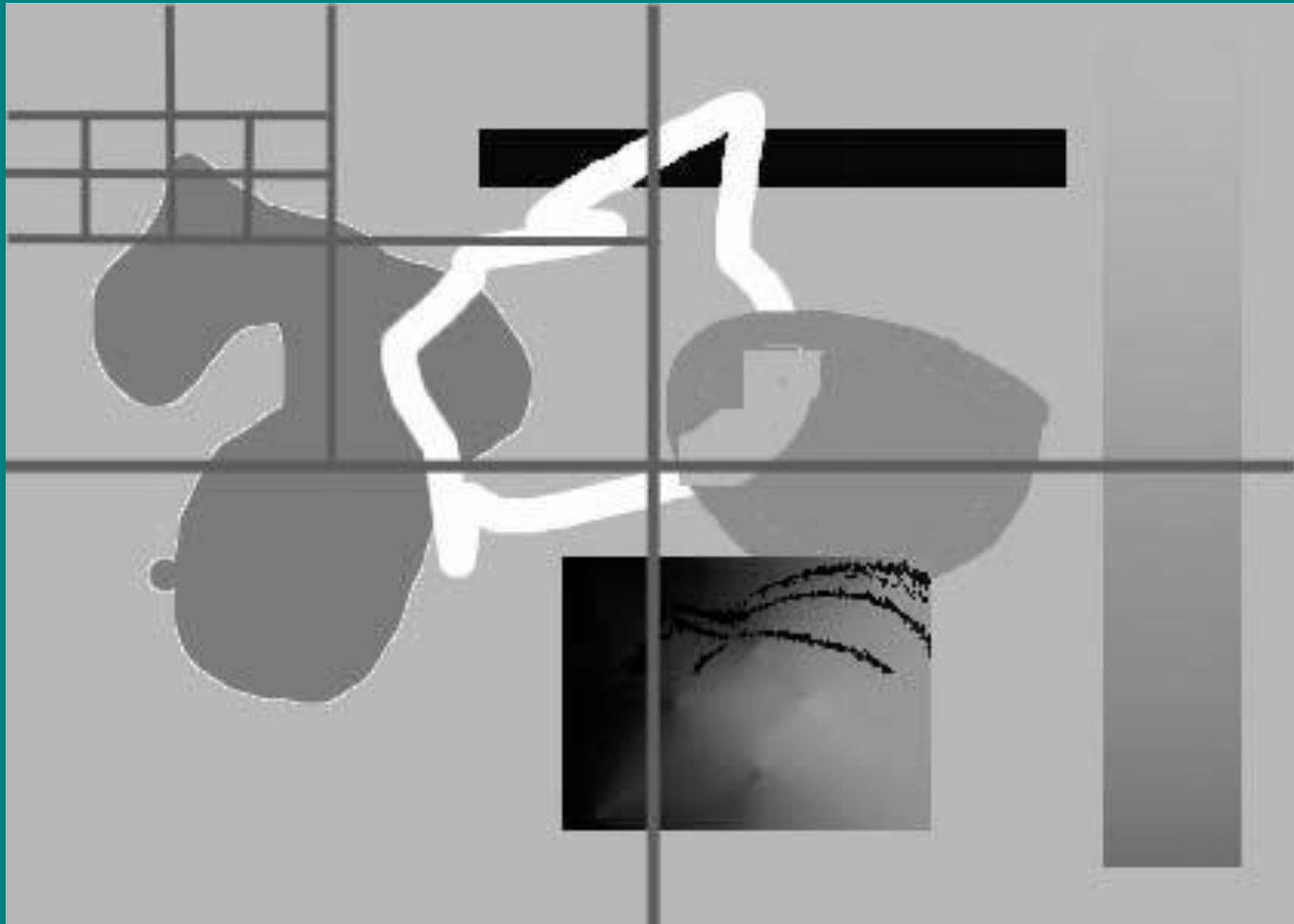
1<sup>ère</sup> étape : division



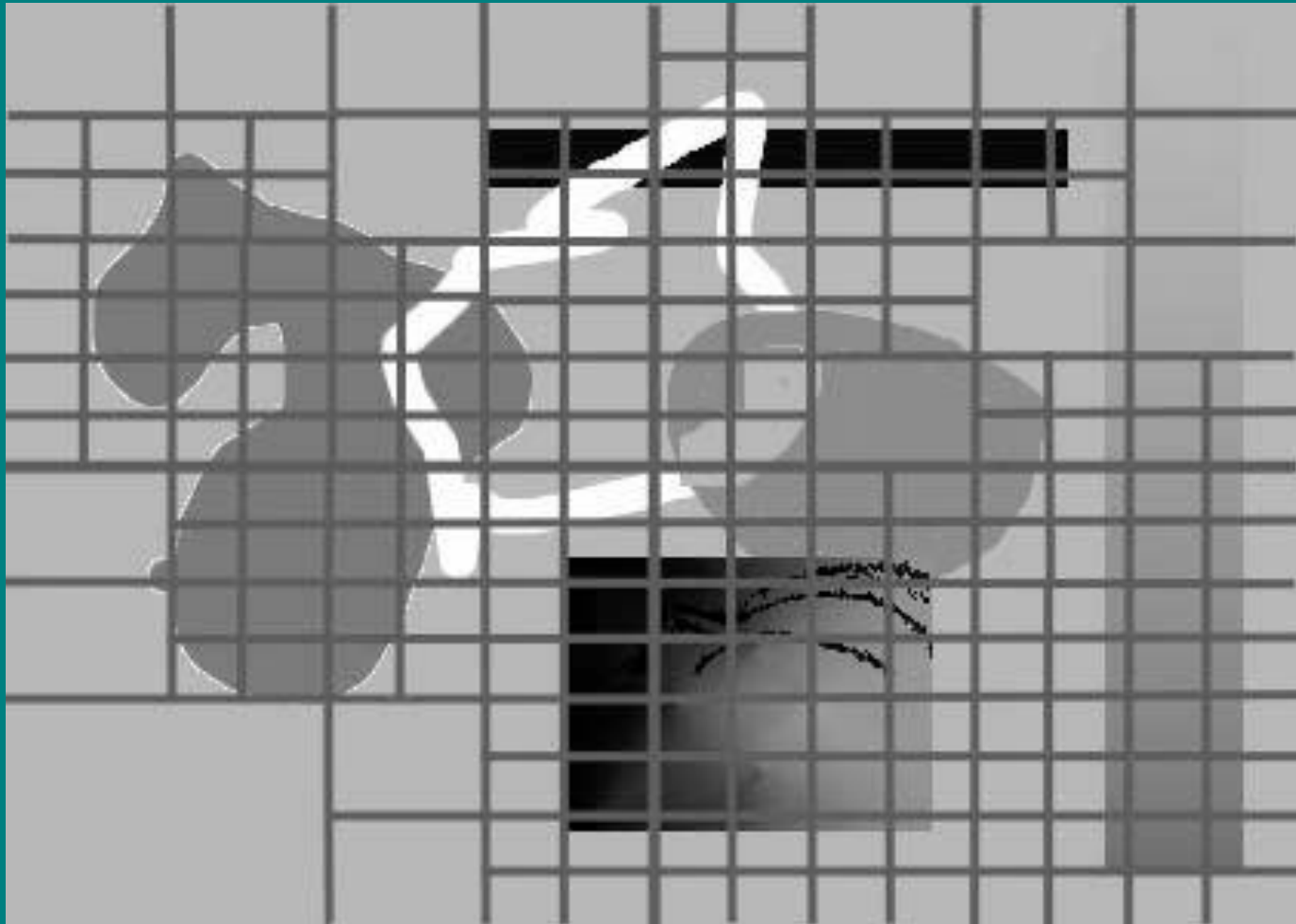
1<sup>ère</sup> étape : division



1<sup>ère</sup> étape : division



1<sup>ère</sup> étape : division





# Algorithme de division

## Exemples de prédicats d'hétérogénéité

La région  $\mathcal{R}_i$  est divisée si  $\mathcal{P} = 1$ .

- **Contraste** de la région

$$\mathcal{P} = \text{VRAI SI } \max_{\mathcal{R}_i}(I(p)) - \min_{\mathcal{R}_i}(I(p)) > \sigma$$

- **Écart-type** de la région :

$$\mathcal{P} = \text{VRAI SI } \sqrt{\frac{1}{N} \sum_{\mathcal{R}_i} (I(x, y) - \mu)^2} > \sigma \text{ avec } N = \text{Card}(\mathcal{R}_i) \text{ et } \mu \text{ moyenne de } I \text{ sur la région.}$$

- **Distance interquartile D** sur la région (distance séparant les 25 % inférieurs des 25 % supérieurs de l'histogramme)  $\mathcal{P} = \text{VRAI SI } D > \text{seuil}$

- **Différences limitées** :

$$\mathcal{P} = \text{VRAI SI } \forall (p, q) \text{ voisins } |I(p) - I(q)| > \sigma$$

- **Entropie**

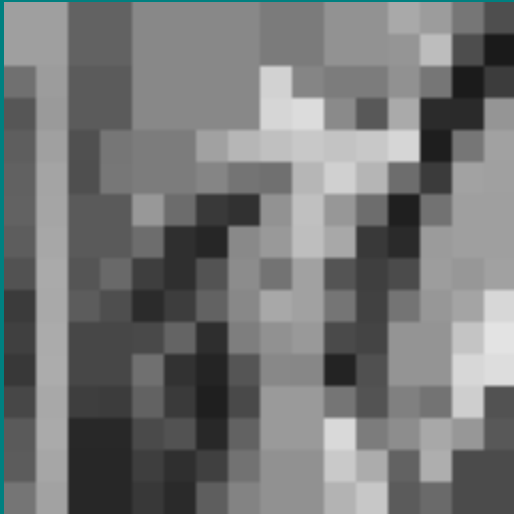
$$\mathcal{P} = \text{VRAI SI } - \sum_{\mathcal{R}_i} p(I) \log(p(I)) > \sigma$$

# Algorithme de division

## Algorithme récursif

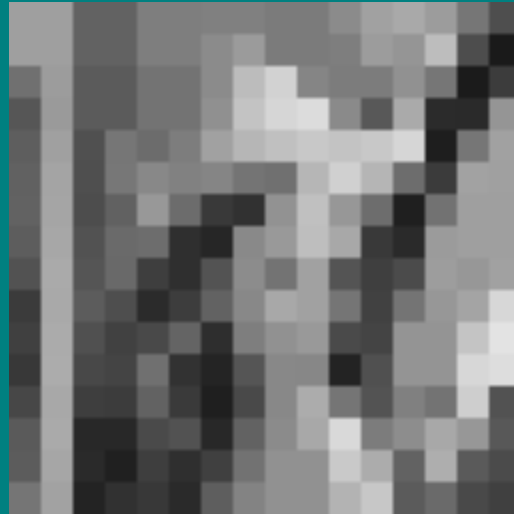
```
function Quadtree (Image, IRes, ligne1, ligne2, col1, col2, seuil, taillemin) Crit ←
calculcritere()
Taille ← (ligne2-ligne1).(col2-col1)
Si Crit>seuil et Taille>taillemin faire MilieuCol ← (col2+col1)/2 MilieuLigne ←
(ligne2+ligne1)/2
    Quadtree (Image, IRes, ligne1, MilieuLigne, col1, MilieuCol, seuil, taillemin)
    Quadtree (Image, IRes, ligne1, MilieuLigne, MilieuCol, col2, seuil, taillemin)
    Quadtree (Image, IRes, MilieuLigne, ligne2, col1, MilieuCol, seuil, taillemin)
    Quadtree (Image, IRes, MilieuLigne, ligne2, col1, MilieuCol, seuil, taillemin)
Sinon
    moyenne ← 0
    Pour x=[ligne1, ligne2] et y=[col1, col2]
        Faire moyenne ← moyenne+Image(x,y)
    Fin Pour
    moyenne← moyenne/(Taille)
    Pour x=[ligne1, ligne2] et y=[col1, col2]
        faire IRes(x,y) ← moyenne
    Fin Pour
Fin fonction
```

# Exemples. Prédicat d'hétérogénéité écart-type d'intensité



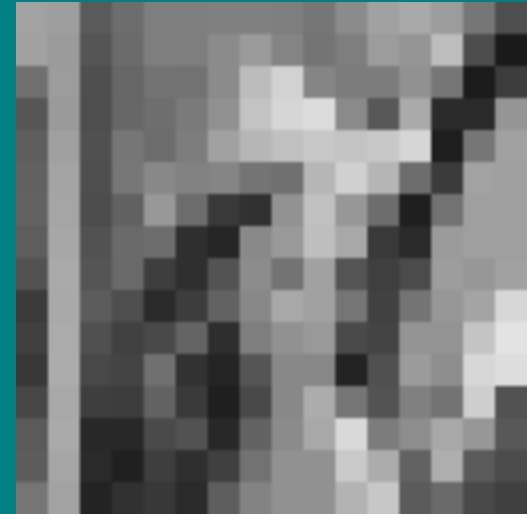
Seuil : 30

Taille min : 100



Seuil : 20

Taille min : 100



Seuil : 10

Taille min : 100



Seuil : 10  
Taille min : 40



Seuil : 10  
Taille min : 20



Seuil : 10  
Taille min : 10

# Exemples. Prédicat d'hétérogénéité écart-type d'intensité



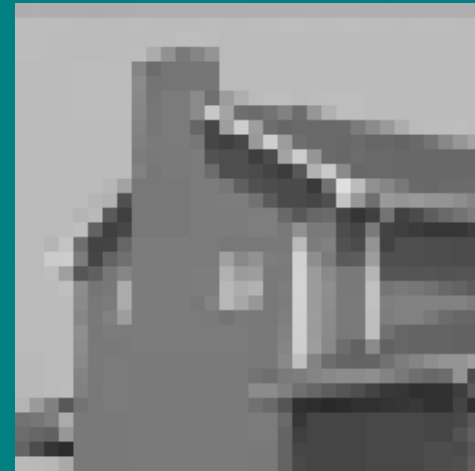
Seuil : 30

Taille min : 100



Seuil : 20

Taille min : 100



Seuil : 10

Taille min : 100



Seuil : 10  
Taille min : 40



Seuil : 10  
Taille min : 20

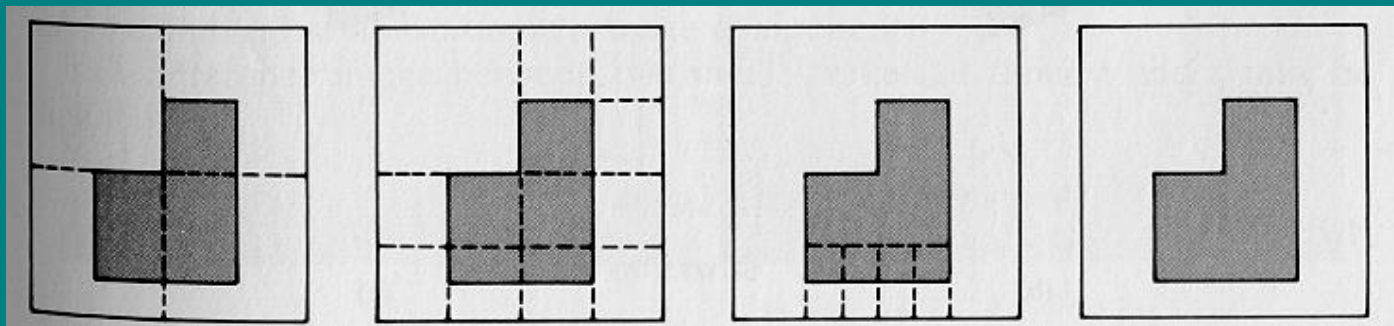


Seuil : 10  
Taille min : 10

# Algorithme de division

Après division

- Deux régions adjacentes peuvent avoir les mêmes propriétés (couleur, niveau de gris, attribut de texture)
- Nécessité de fusionner ces régions : analyse des régions 2 à 2



(1) Split into four disjoint quadrants any region  $R_i$  where  $P(R_i)=\text{False}$

(2) Merge any adjacent regions  $R_j$  and  $R_k$  for which  $P(R_j \cup R_k)=\text{True}$ ;

(3) Stop when no further merging or splitting is possible

# Algorithme de division-fusion



thresholding



split and merge



$P(R_i) = \text{True}$  if

$|z_i - m_i| \leq 2\sigma_i$  for 80% of the pixels in  $R_i$

$(m_i, \sigma_i)$  are the mean and standard deviation of pixels in  $R_i$

# Algorithme de division-fusion : structures de données

Pour simplifier la division - fusion

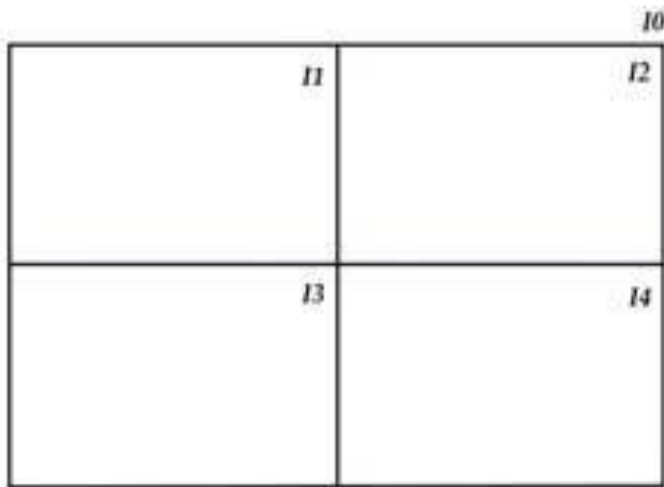
Utilisation de structures de données spécifiques :

- Le tétra-arbre
- Les graphes d'adjacence de régions

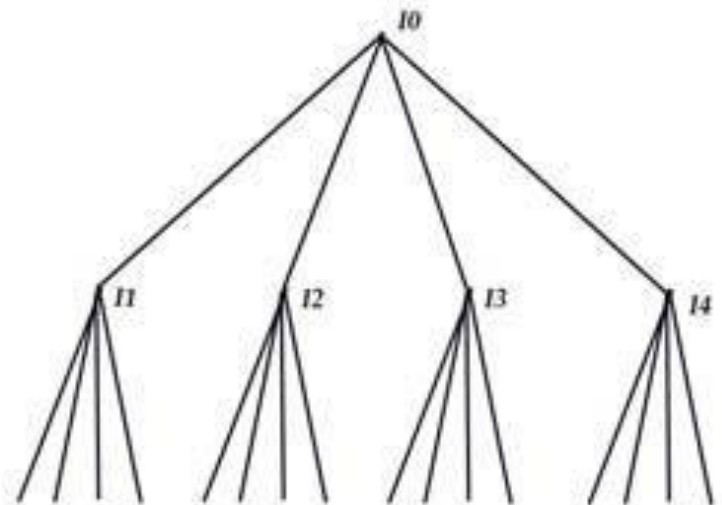
# Algorithme de division-fusion : structures de données

## Tétra-arbre (quad-tree)

- Structure arborescente
- Un nœud = partition rectangulaire
- Racine de l'arborescence (niveau 0) : image entière dont le nombre de lignes et de colonnes est un nombre puissance de deux
- Chaque nœud au niveau  $k$  possède 4 nœuds fils au niveau  $k + 1$



Division de l'image en quadrants



Structure de données Quadtree



# Algorithme de division-fusion : structures de données

## Analyse du tétra-arbre

- 1ere étape :

- Chaque bloc (nœud du tétra-arbre) est analysé (division ou non ?)

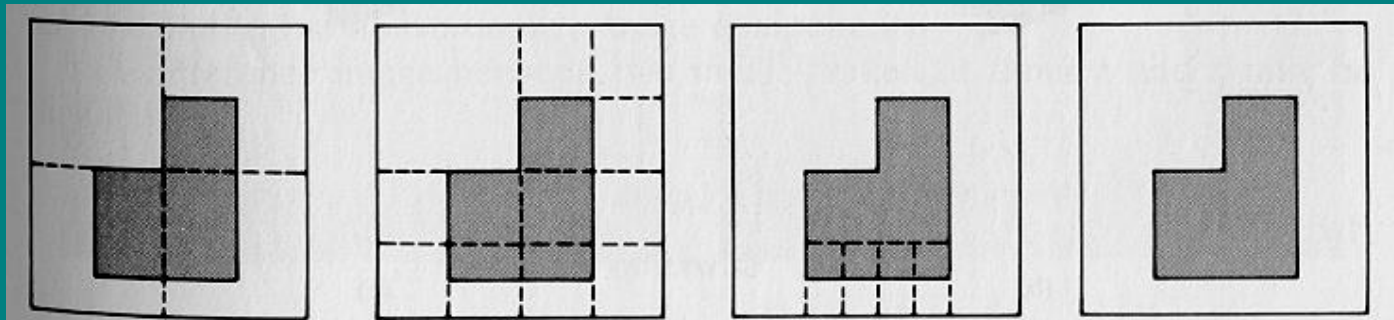
- Arrêt : lorsque tous les sous-blocs respectent un critère d'homogénéité

Après 1ere étape : certains blocs adjacents possèdent des caractéristiques identiques

- 2eme étape :

- Fusion des blocs répondant à un critère d'homogénéité.

- Arrêt : lorsque plus aucune fusion n'est possible.



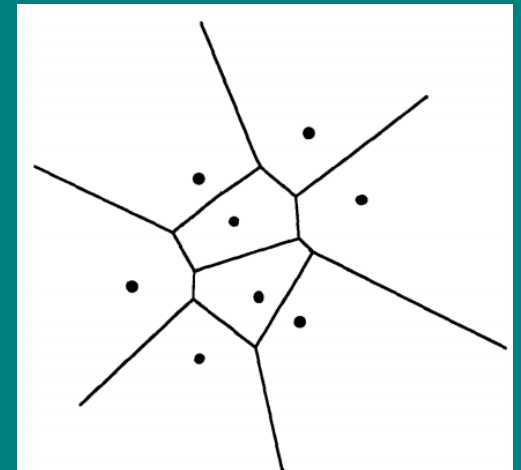
# Algorithme de division-fusion : avantages et inconvénients

## Avantages

- Méthode à la fois globale et locale : globale lors de la division, et locale lors de la fusion
- Moins sensible au bruit que la croissance de régions

## Inconvénients

- Méthodes parfois complexes d'un point de vue algorithmique : manipulation de lourdes structures de données
  - Découpage en régions non fidèles à l'image originale : régions rectangulaires dans le cas du quadtree
- améliorations possibles : partitions de Voronoï



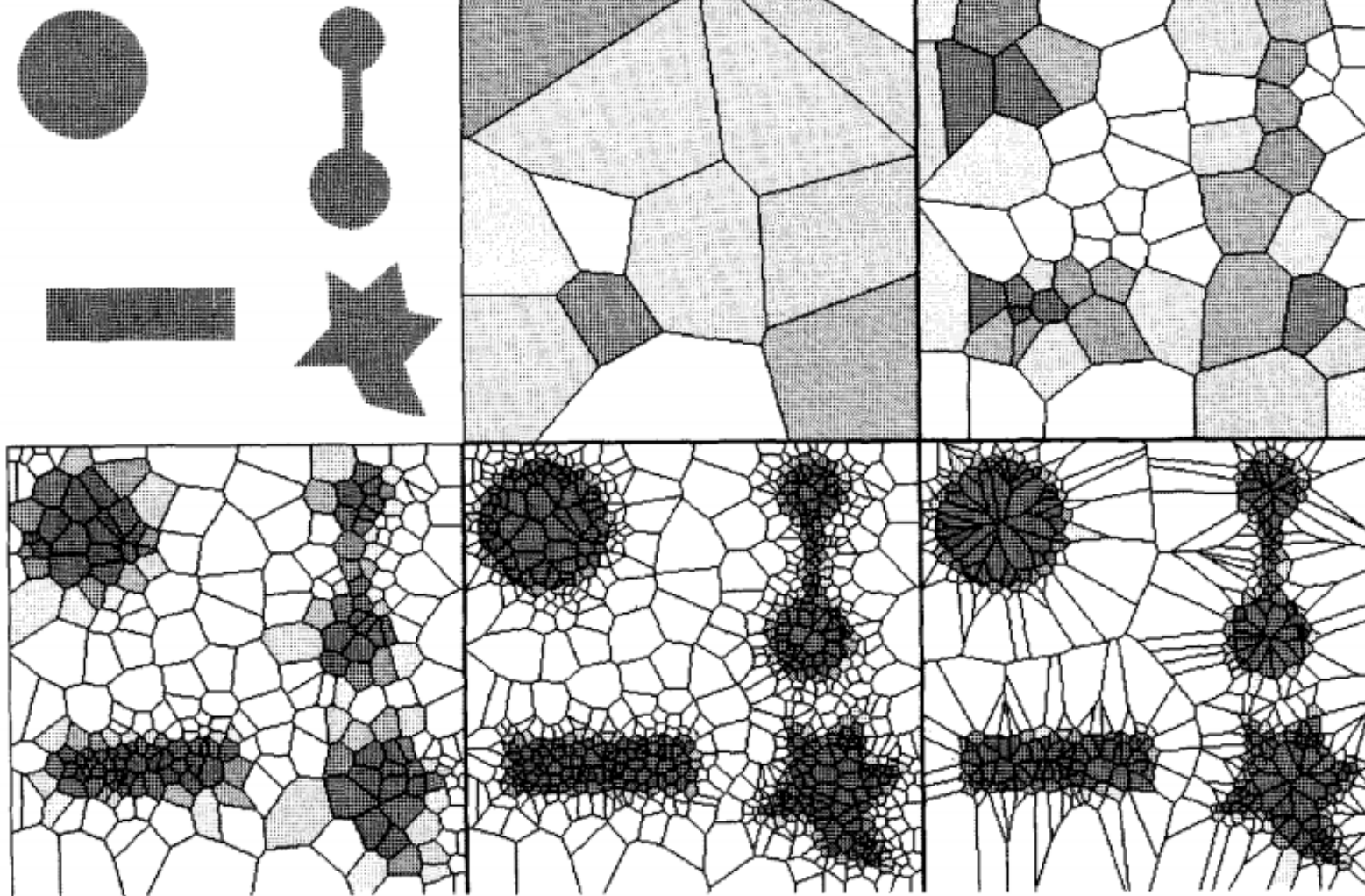
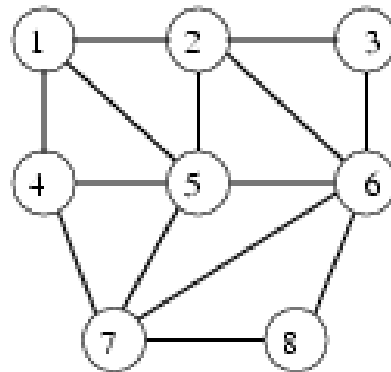
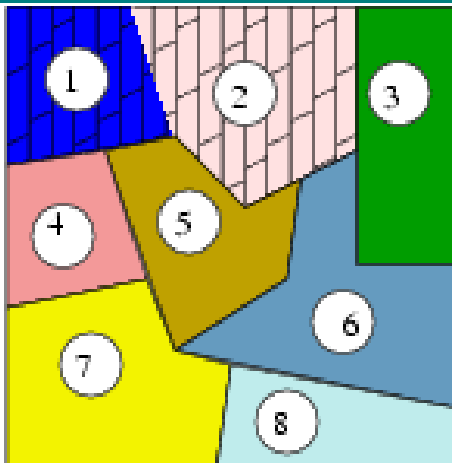


Fig. 4. — Illustration sur une image élémentaire (image 1) du processus de segmentation. La phase d'initialisation (image 2) met en œuvre 13 germes. La phase itérative est illustrée avec les images 3 à 5 avec respectivement 60-268-781 germes. Après la phase de fusion l'image est représentée avec 490 germes (image 6). On remarquera la topographie des polygones traduisant la forme des objets à segmenter.

- Chassery et al 1991

# Algorithme de division-fusion : structures de données

- Le graphe d'adjacence de régions
  - Structure de données utilisées après une sur-segmentation (partition trop fine par rapport au résultat attendu)
  - Une région est représentée par un nœud
  - Un arc définit une relation d'adjacence
  - Une fonction de similarité est définie entre deux nœuds

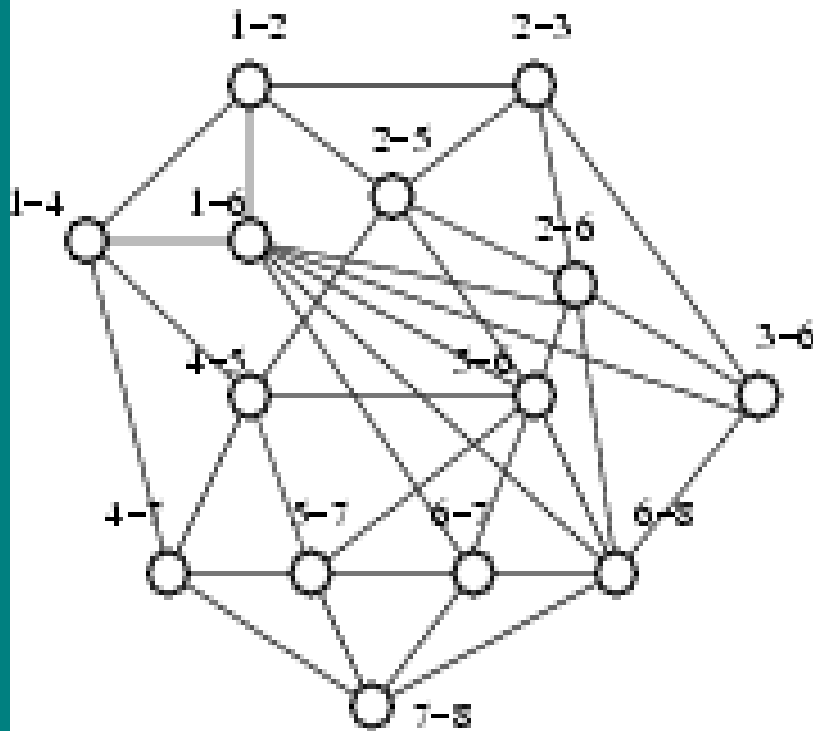


Principe :

- Tri de tous les nœuds adjacents dans une liste

- Regroupement des deux meilleurs candidats : fusion

- Mise à jour de la liste et itération



	1	2	3	4	5	6	7	8
1		1	0	1	1	0	0	0
2	1		1	0	1	1	0	0
3	0	1		0	0	1	0	0
4	1	0	0		1	0	1	0
5	1	1	0	1		1	1	0
6	0	1	1	0	1		1	1
7	0	0	0	1	1	1		1
8	0	0	0	0	0	1	1	

Outils de manipulation du GAR.

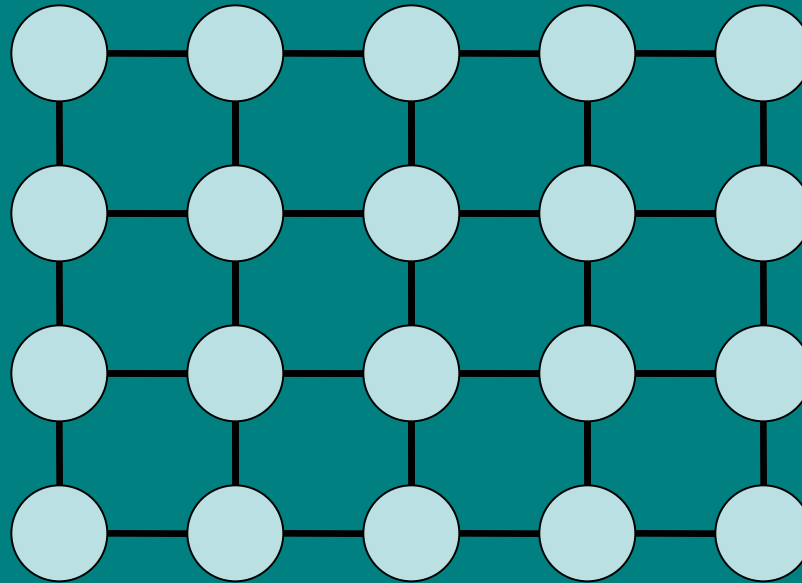
# Algorithme de fusion : graphes d'adjacence

Exemple de fusion à partir du GAR

- Pondération de chaque arête du GAR proportionnellement à la distance (en niveaux de gris ou colorimétrique, etc) entre deux régions
- A chaque itération
  - recherche des régions reliées par l'arête de poids minimal
  - les poids des arêtes sont mis à jour en fonction de l'aire des régions et de leur distance colorimétrique (privilégier les régions adjacentes de faible surface)
- Arrêt de l'algorithme :
  - soit lorsqu'un nombre d'itérations est atteint
  - soit lorsque les poids des arêtes atteignent une valeur limite

# Technique de graphe

- On considère l'image comme un graphe



Un graphe :  
ensemble de  
points + ensemble  
de liens

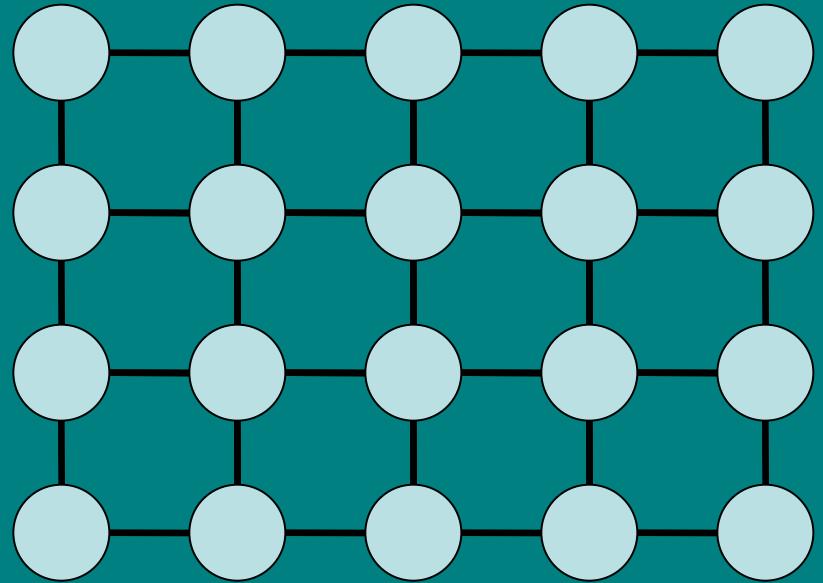
- Intérêt : représentation compacte, structurée, complète, facile à manipuler
- Pixel = noeud

# Technique de graphe

- Construction d'une matrice de coût

Si zone uniforme : coût élevé

Si zone de contour : coût faible



Segmenter l'image consiste à trouver le chemin de coût minimal dans le graphe



# Technique de graphe

## ■ Exemple

*NdG*

	1	2	3
1	● [5]	● [6]	● [1]
2	● [6]	● [7]	● [0]
3	● [7]	● [1]	● [3]

*Construction d'une matrice de coût telle que :*

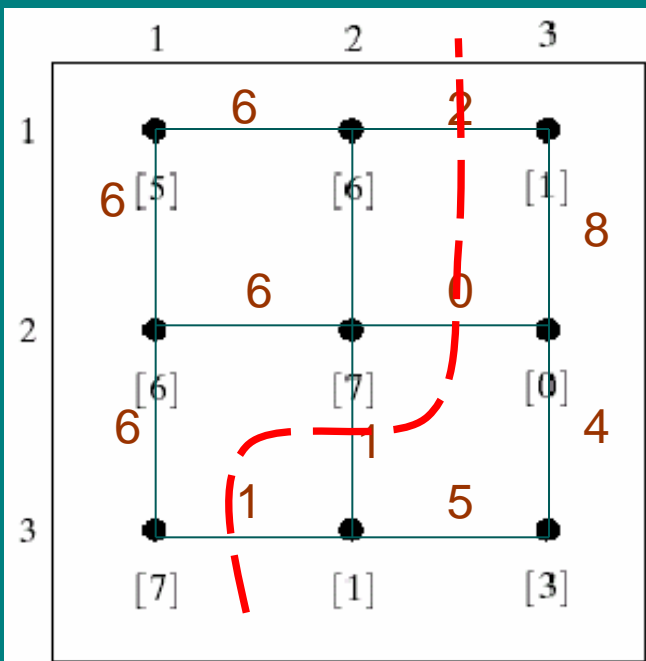
$$C(p,q) = \text{Max} - |I(p) - I(q)|$$

# Technique de graphe

## ■ Exemple

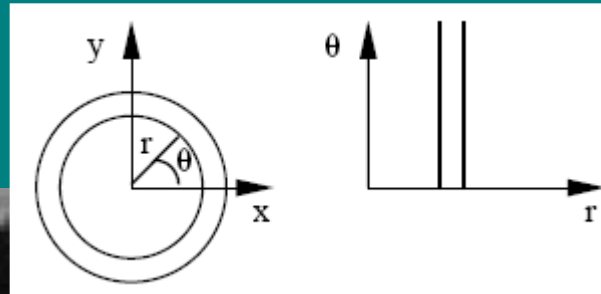
$$C(p,q) = \text{Max-}|I(p) - I(q)|$$

*Recherche du chemin de coût minimal*

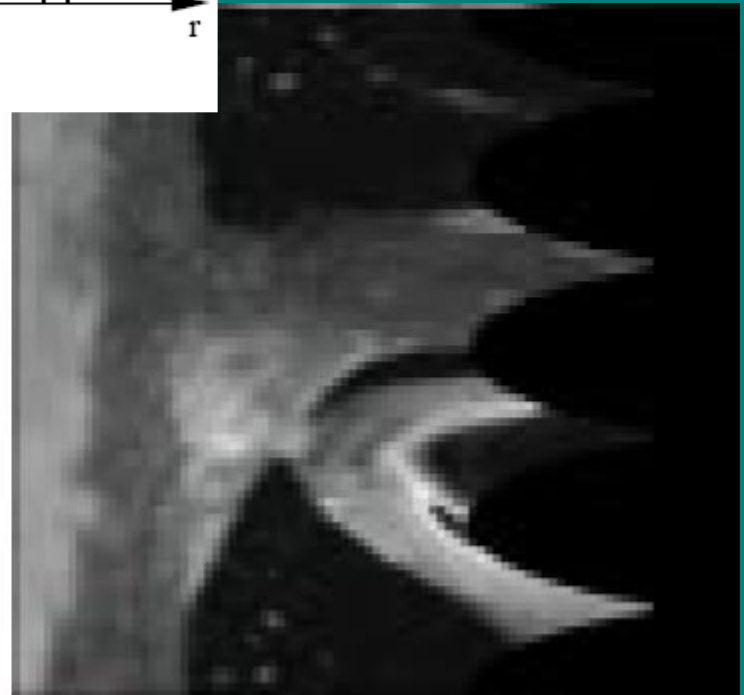
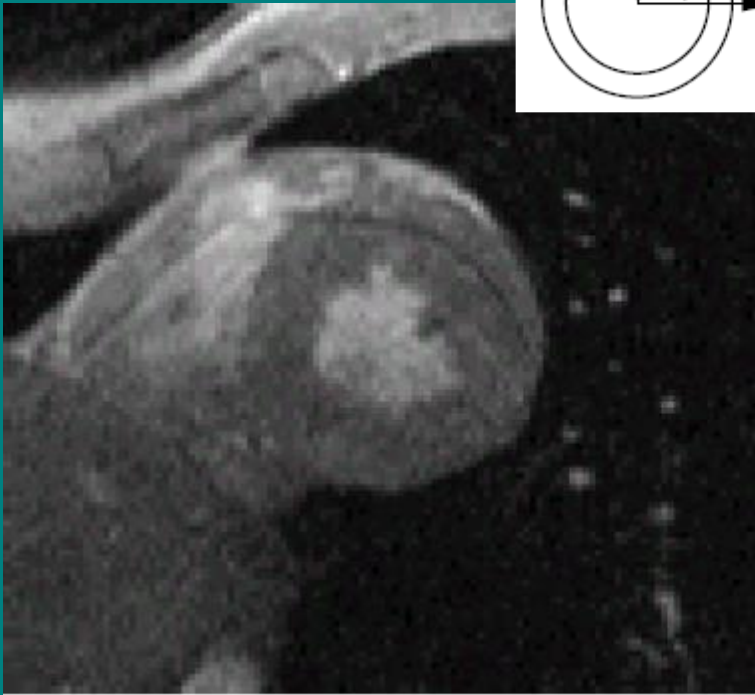


# Technique de graphe

- IRM cardiaque → Transformation en coordonnées polaires

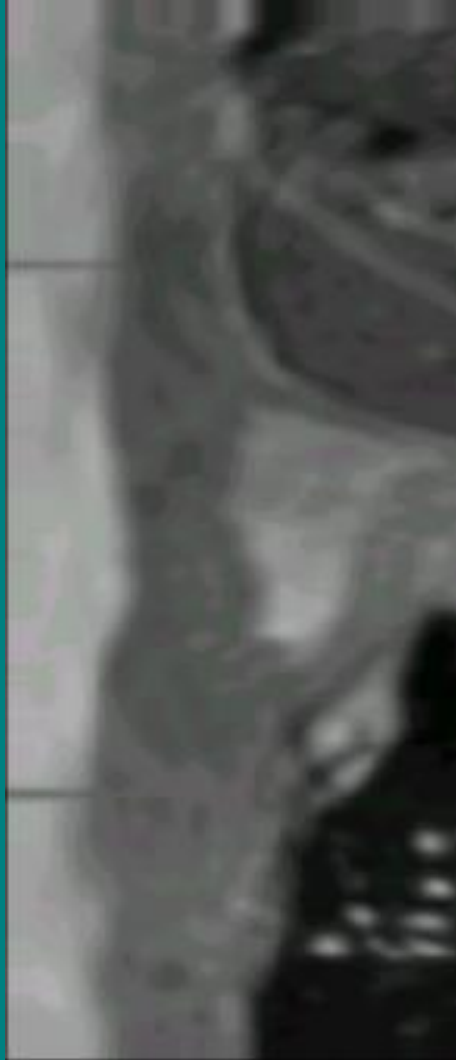


Source : Lalande et al. 1999

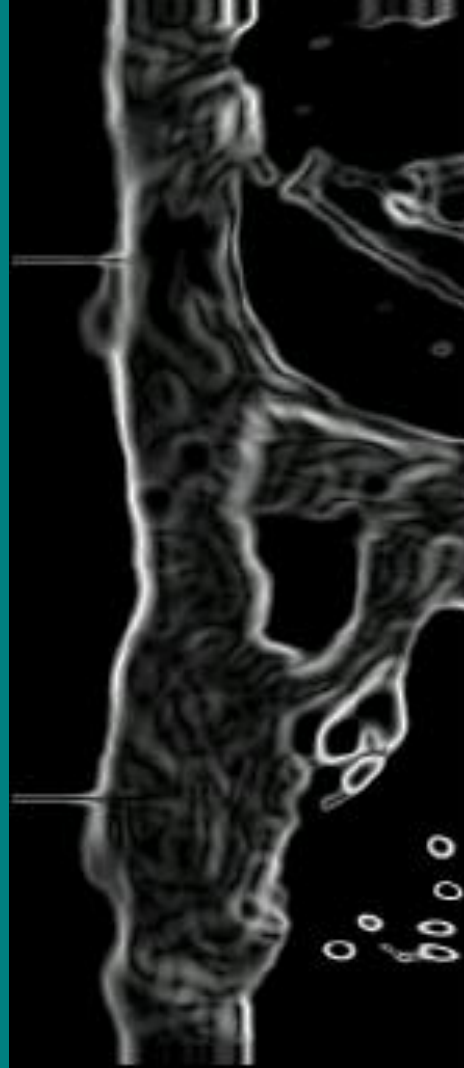


# Technique de graphe

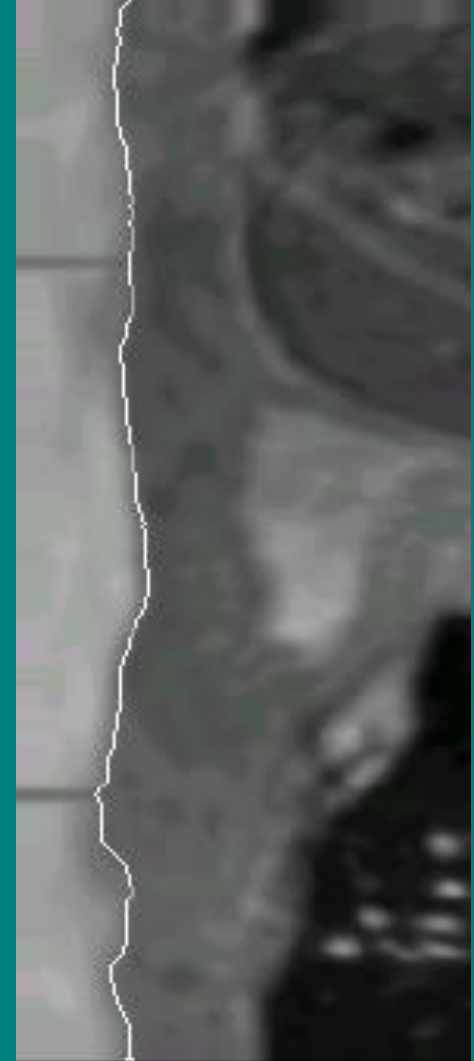
Image originale (coord.polaires)



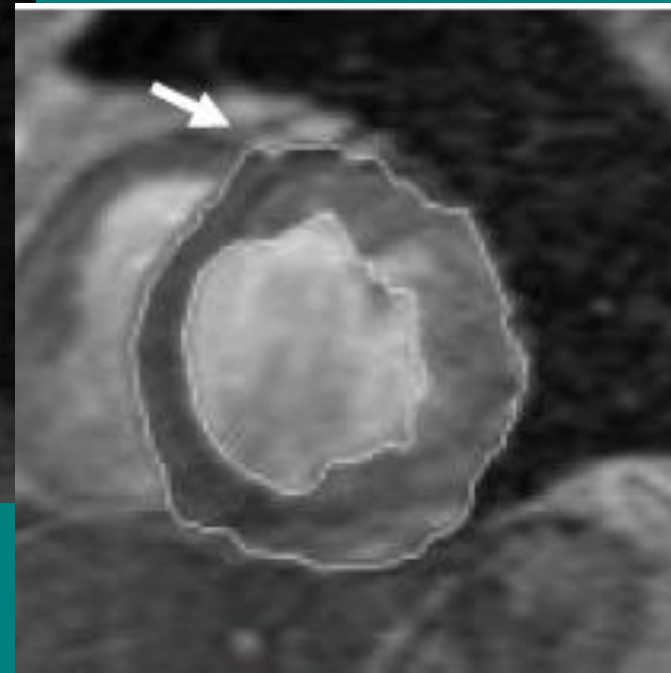
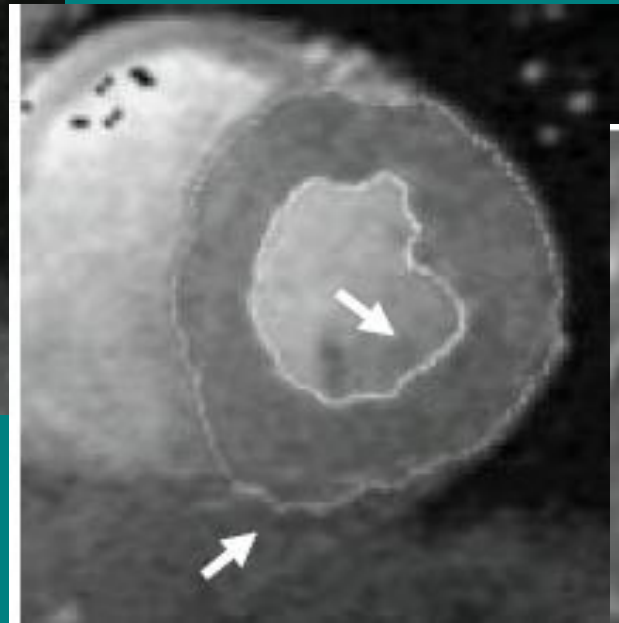
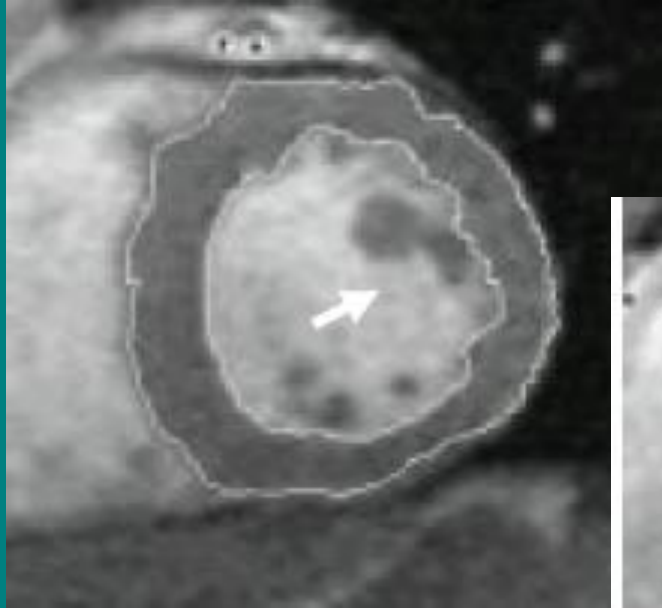
Matrice de coût



Segmentation finale



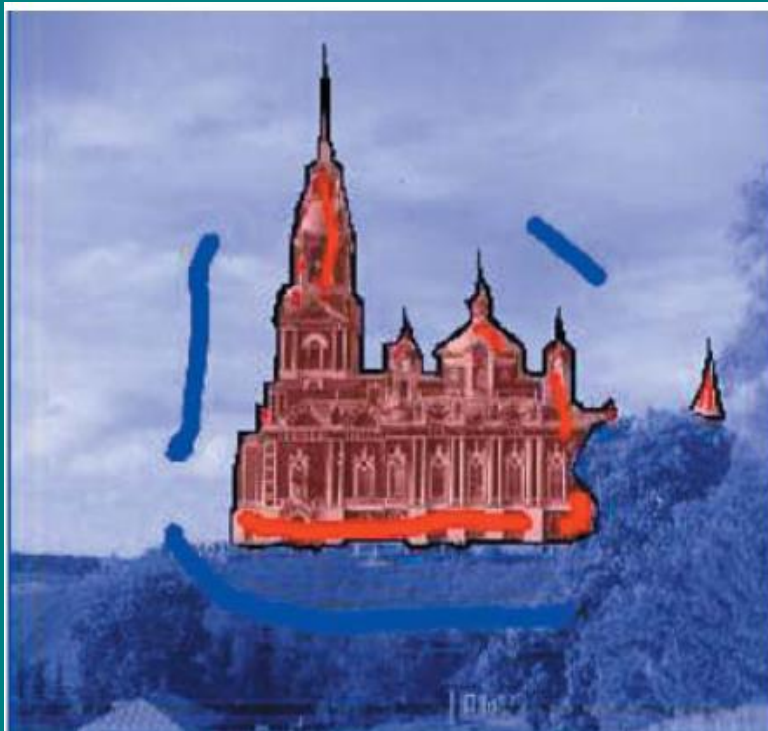
# Technique de graphe



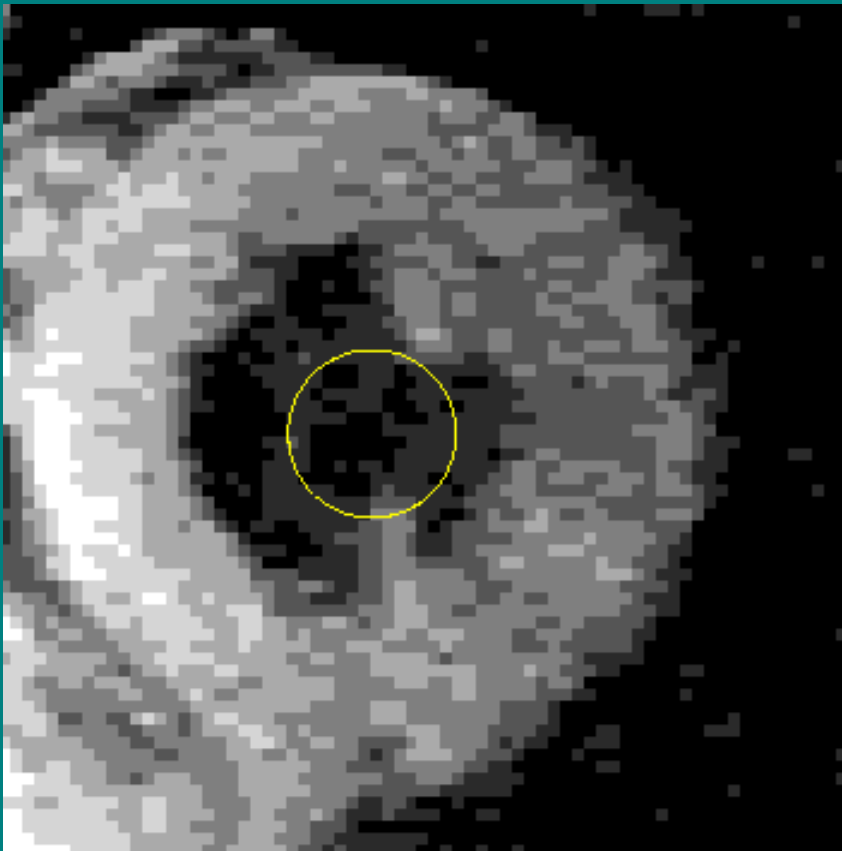
Source : Lalande et al. 1999

# Technique de graphe

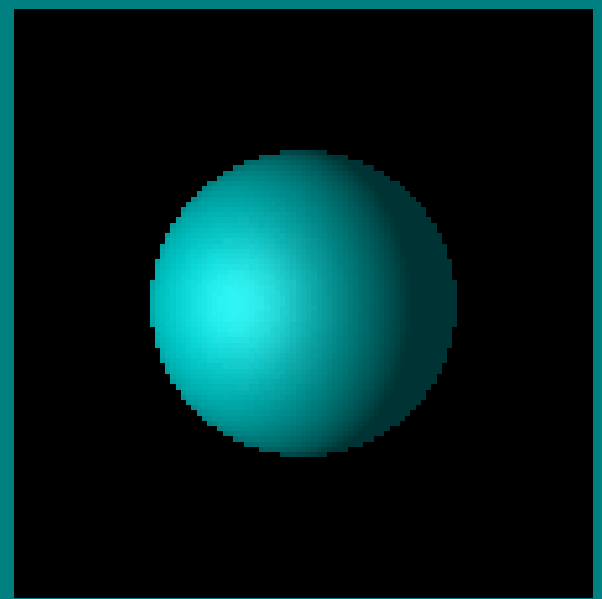
- Graph cut : Coupe optimale dans un graphe avec interaction utilisateur



# Modèles déformables



Xu and Prince, Gradient Vector Flow, John Hopkins Univ.



Xu and Prince, Gradient Vector Flow, John Hopkins Univ.



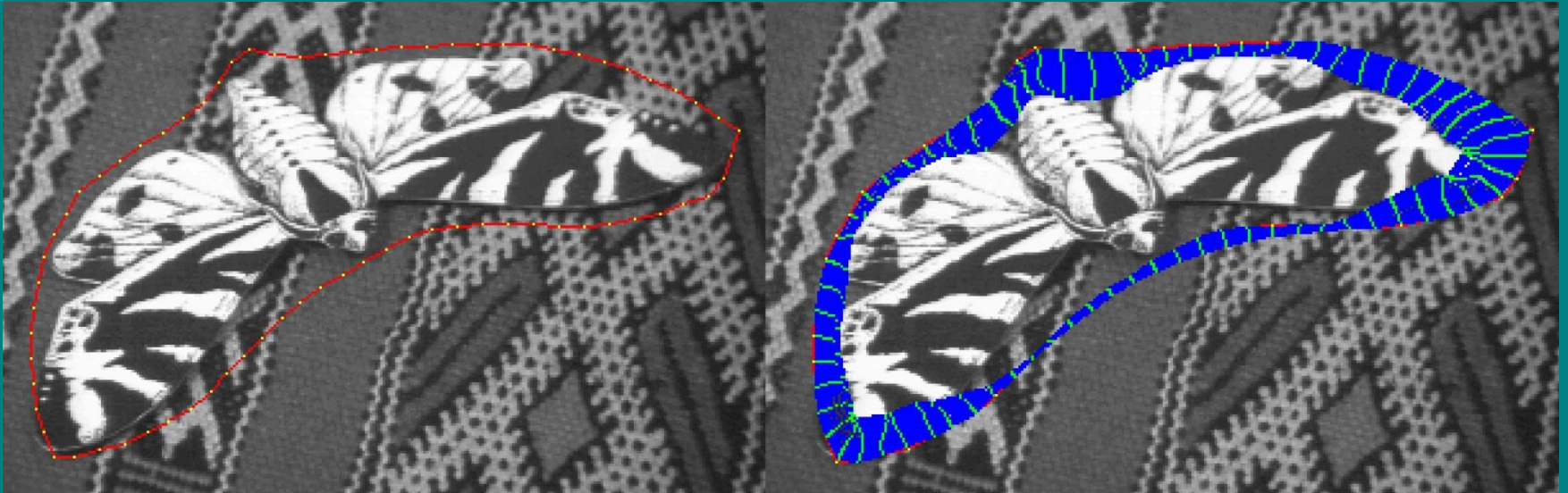
<http://xphilipp.developepez.com/contribuez/>

# Modèles déformables

- Qu'est-ce qu'un modèle déformable pour la segmentation d'image ?
- C'est une forme (= une courbe, un modèle) qui se déforme pour épouser la forme d'un objet dans l'image
  - Position du contour  $\leftrightarrow$  Energie du contour
  - Position finale (optimale)  $\leftrightarrow$  Energie minimum du contour



# Evolution du snake



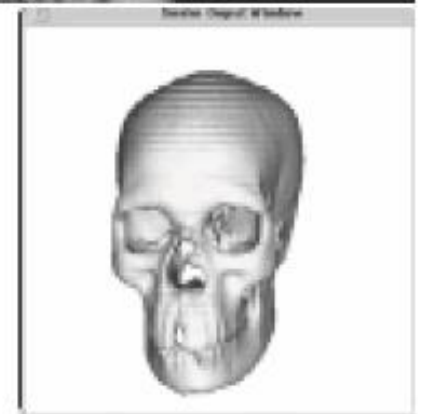
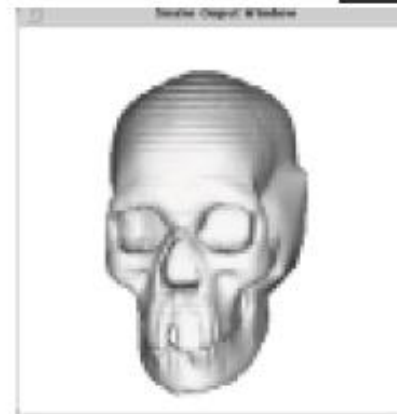
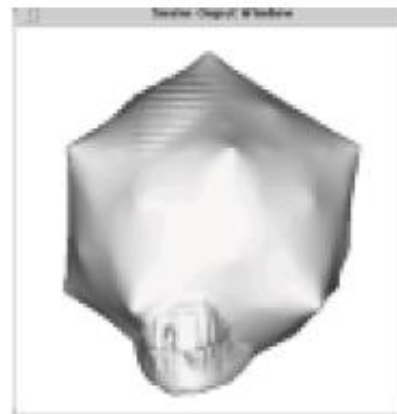
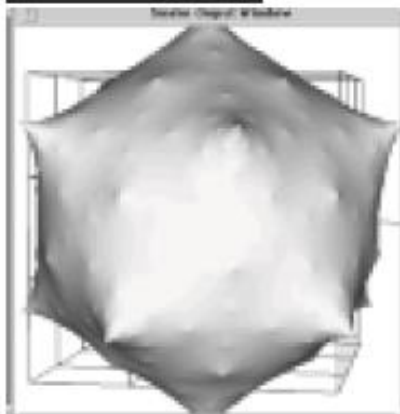
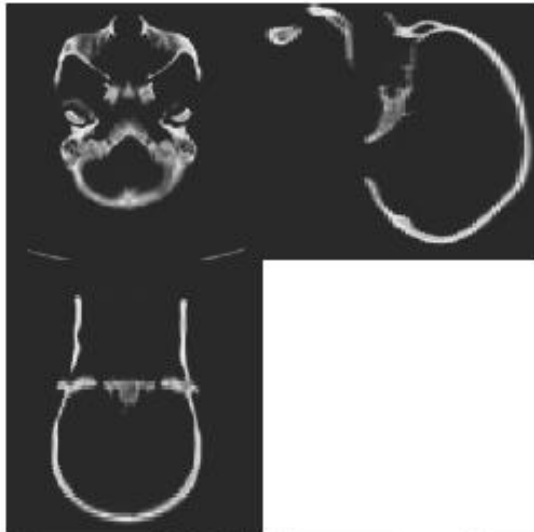
Initialisation :

courbe assez proche du  
contour extraire

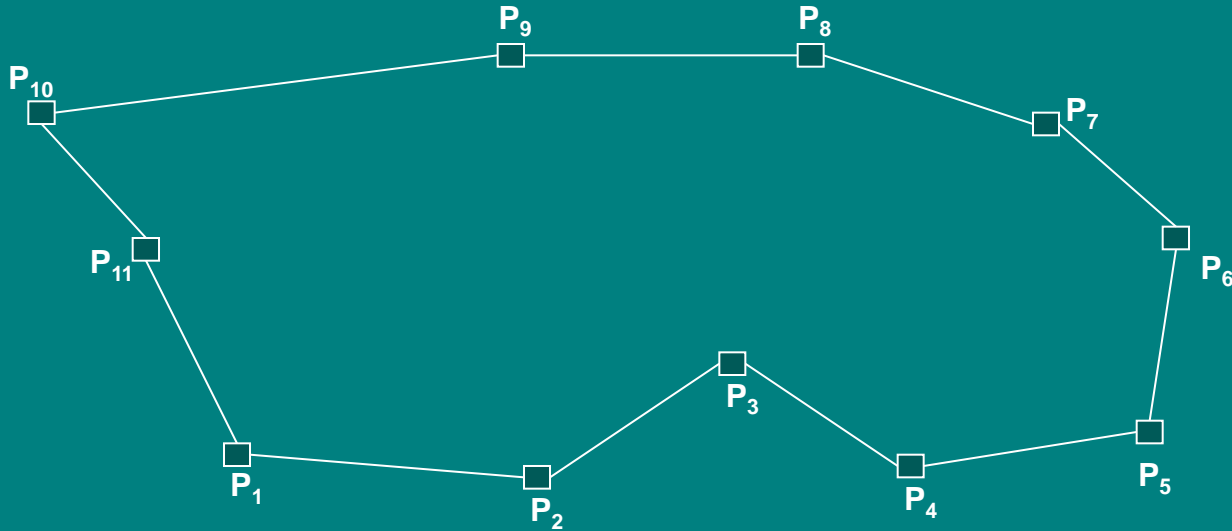
Optimisation itérative :

déformations du contour actif de  
façon ce qu'il atteigne une  
position d'énergie minimum.

# Exemple snake 3D



# Qu'est-ce qu'un snake ?



$$E_{\text{snake}} = E_{\text{interne}} + E_{\text{externe}}$$

- Propriétés intrinsèques
- Longueur, courbure...

- Propriétés locales de l'image autour du snake

# Energie d'un snake

- Formulation paramétrique du contour

$$C = \{v(s) = (x(s), y(s)), s \in [0,1]\}$$

$$\text{Energie totale}(C) = E_{\text{int}}(C) + E_{\text{ext}}(C)$$

- Energie interne : mesure la régularité de la courbe

$$E_{\text{int}} = \int_0^1 (\alpha |v'(s)|^2 + \beta |v''(s)|^2) ds$$

Elasticité

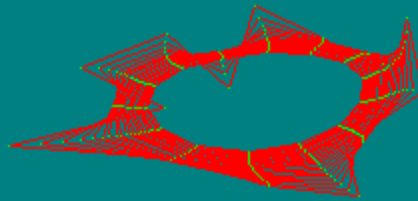
Rigidité

$$E_{\text{elastic}} = \sum_{i=1}^n |P_i - P_{i-1}|^2$$

$$E_{\text{rigidité}} = \sum_{i=1}^{n-1} |P_{i-1} - 2P_i + P_{i+1}|^2$$

# Energie interne d'un snake

Energie *élastique*



Energie *de courbure*



# Energie externe d'un snake

- Energie externe : reliée au contenu de l'image

$$E_{ext} = \lambda \int_0^1 g^2(I(v(s))) ds$$

Avec  $g$  fonction généralement décroissante de gradient de l'image

- Si contour : gradient élevé  $\rightarrow g \approx 0$
- Si zone homogène : gradient faible  $\rightarrow g$  élevé

# Energie externe d'un snake

- Fonction décroissante du gradient

$$E_{\text{ext}} = -\lambda |\nabla I|^2$$

$$E_{\text{ext}}(\vec{x}) = \frac{\lambda}{1 + |\nabla I|^2}$$



$g = 0$

$g = 1$

$I$

$$g(I(C)) = \frac{1}{1 + |\nabla I(C)|^2}$$

# Résolution

- Comment trouver  $C$  qui minimise  $E = E_{\text{int}} + E_{\text{ext}}$  ?



# Résolution

- Comment trouver C qui minimise  $E = E_{\text{int}} + E_{\text{ext}}$  ?
- Par l'équation d'éuler Lagrange

Soit E la fonctionnelle d'énergie

$$E = \int f(s, v(s), v'(s)) ds$$

Pour que E atteigne un extremum, il faut que  $v(s)$  vérifie

$$\frac{\partial f}{\partial v} - \frac{d}{ds} \left( \frac{\partial f}{\partial v'} \right) = 0$$

Dans notre cas :

$$E = \underbrace{\int_0^1 (\alpha |v'(s)|^2 + \beta |v^{(4)}(s)|^2) ds}_{E_{\text{int}}} + \underbrace{\lambda \int_0^1 g^2(I(v(s))) ds}_{E_{\text{ext}}}$$

$$-\alpha v''(s) - \beta v^{(4)}(s) + \lambda \nabla g^2(I(v(s))) = 0$$

**Equation d'évolution du snake**

$$-\alpha v''(s) - \beta v^{(4)}(s) + \lambda \nabla g^2(I(v(s))) = \frac{\partial v(s, t)}{\partial t}$$

# Implémentation

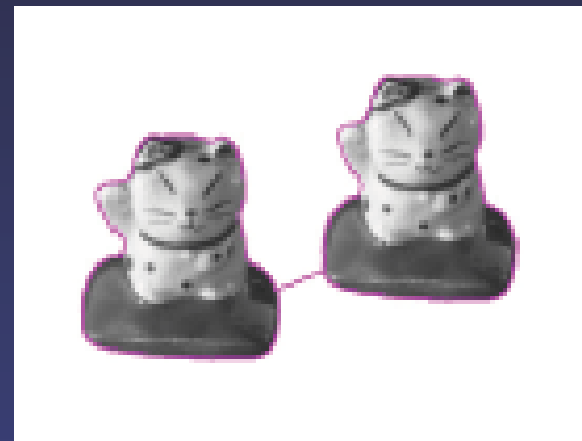
- Soit  $N$  le nombre de points qui constituent le contour
- Plusieurs possibilités
  - Différences finies (Kass et al., 1988)
  - Programmation dynamique (Amini et al., 1990)
  - Greedy algorithm (Williams & Shah, 1992)  
algorithme « glouton »

# Greedy (simplifié)

- Initialisation :
  - Pour chaque point  $P_i$   $E_i = Val\_max$
- Pour chaque point  $i$ 
  - Pour tous les points  $k$  du voisinage  $V(P_i)$  calculer l'énergie  $E_k$ 
    - Si  $E_k < E_i$  alors  $E_i = E_k$  et déplacer le point  $i$  vers le point  $k$
    - Sinon, ne rien faire
- Continuer tant que le nb de points déplacés est supérieur à un seuil

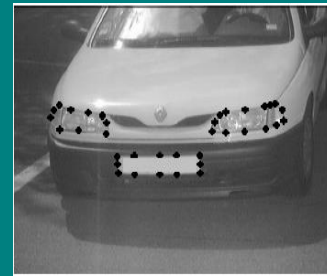
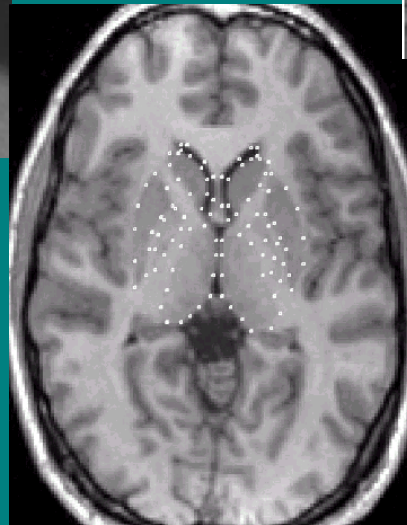
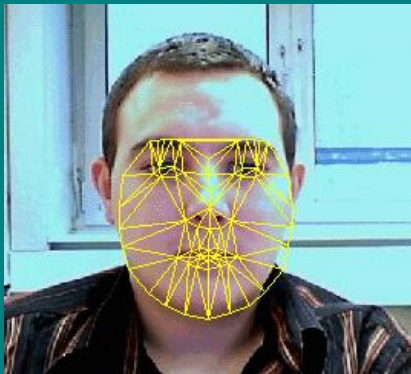
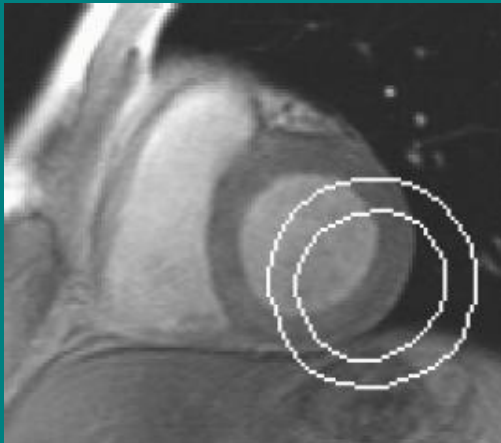
# Conclusions sur les snakes

- Avantage : calculs numériques rapides
- Inconvénients
  - Segmentation multi-objets impossible
  - Phase d'initialisation sensible
  - Approche non intrinsèque



# Segmentation d'une forme connue

- Utiliser l'information de forme dans le processus



Source images : M. Hachama

# Références

## Sources des images

- Introduction to Digital Image Processing, Image Segmentation, Zhou Wang, Dept. of Electrical Engineering, the Univ. of Texas at Arlington, 2006
- Segmentation d'images, Michèle Gouiffès
- Cours de Vision artificielle, Christine Fernandez-Maloigne, Université de Poitiers
- **CS474/674 – Prof. Bebis**