

Segmentation et Analyse d'images (partie 1)

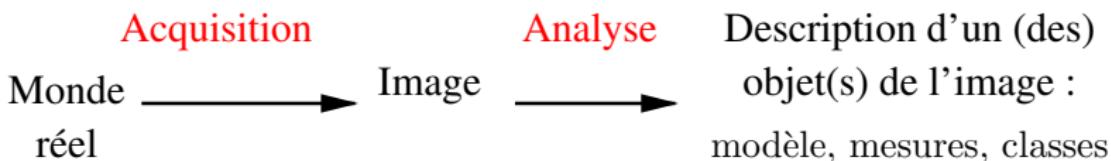
Anne Vialard

LaBRI, Université Bordeaux 1

Sommaire

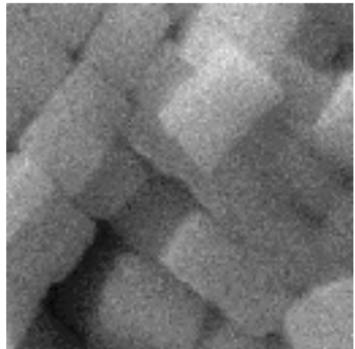
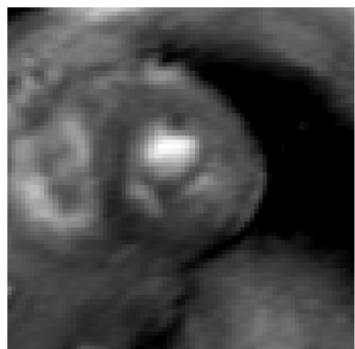
- 1 Introduction
- 2 La segmentation d'images : généralités
- 3 Segmentation : approches région
- 4 Segmentation : Approche frontière
- 5 Autres approches de la segmentation

Problématique I



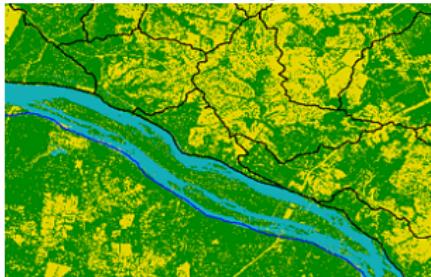
- On cherche à **imiter** le système de vision humain
- Importance des connaissances **a priori** pour les processus de haut-niveau
- **Pas de solution générale** mais un ensemble de solutions adaptées à des problèmes particuliers

Des cas difficiles... I



Exemples d'applications I

- Analyse d'image satellite

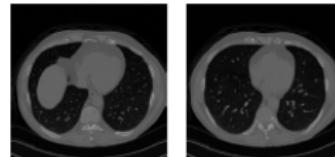


- Analyse de document

**Recueil de
s, sans ord**

- Recherche basée sur le contenu
- Contrôle qualité
- Contrôle du déplacement d'un robot
- Vidéosurveillance

- Aide au diagnostic médical



- Imagerie cellulaire

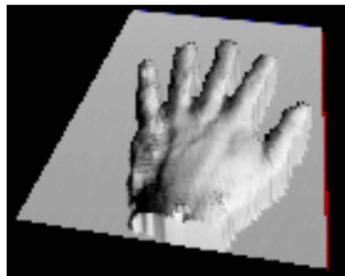
Sous-domaines de l'analyse d'image I

- Segmentation / Reconstruction
- Reconnaissance de formes (Pattern recognition) : Associer une catégorie à une (partie d'une) image

prétraitements → extraction de paramètres → catégorisation

- Vision 3D

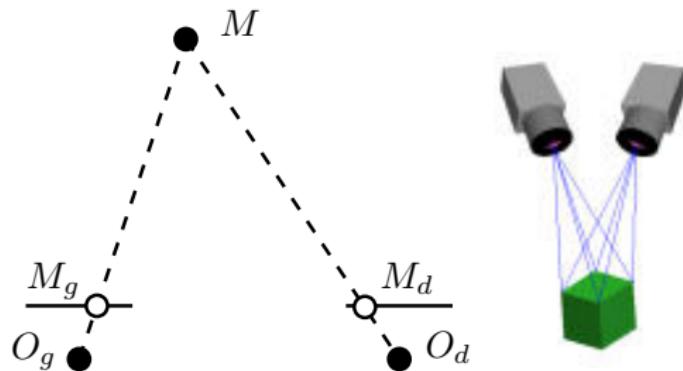
Shape from shading : Retrouver une forme 3D à partir d'une image 2D. On déduit la profondeur des variations de couleur.



Sous-domaines de l'analyse d'image II

Stereovision (Shape from stereo) : Retrouver la profondeur des points d'une scène à partir de deux projections de cette scène (points de vues différents). Etapes :

- mise en correspondance des deux images
- reconstruction



- Analyse de mouvement

Plan du cours I

1 Segmentation d'image

- approches basées région : découpe, fusion, découpe/fusion
- approches basées contour
- autres approches : watersheds, Mumford Shah, modèles déformables, level sets, champs de Markov
- problèmes spécifiques de la segmentation 3D

2 Géométrie discrète appliquée à l'analyse d'image

- courbes et surfaces / régions
- algorithmes de suivi de frontière
- représentation d'une partition 2D/3D
- droites et plans discrets, algorithmes de reconnaissance
- distances discrètes, transformée en distance, squelettisation

Plan du cours II

3 Mesures dans une image

- attributs de région : moments géométriques, convexité, attributs topologiques
- attributs géométriques d'un contour : longueur, normale et courbure en un point, points dominants
- extensions en 3D

Bibliographie I

- The Computer Image - A. Watt, F. Policarpo - Addison Wesley
- Fundamentals of Digital Image Processing - AK Jain - Prentice Hall
- Digital Image Processing - RC Gonzalez, RE Woods - Prentice Hall
- Computer Vision : A Modern Approach - DA Forsyth, J Ponce - Prentice Hall
- Le traitement des images - H. Maître - Hermès
- Analyse d'images : filtrage et segmentation - JP Cocquerez, S. Philipp - Masson
- Digital Geometry : Geometric Methods for Digital Image Analysis - R. Klette, A. Rosenfeld - Morgan Kaufmann

Bibliographie II

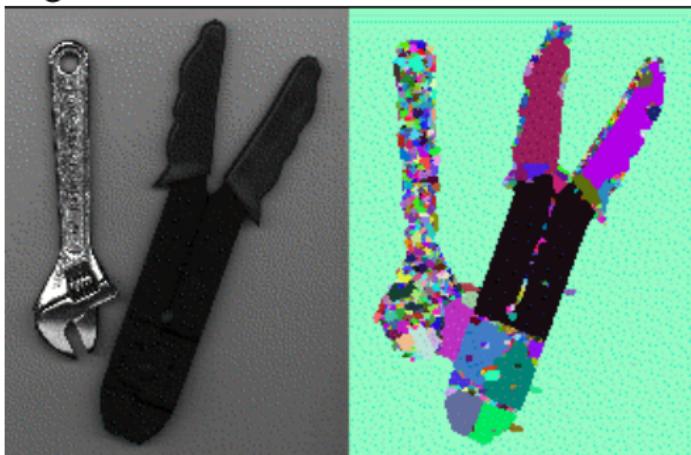
- Géométrie discrète et images numériques - ouvrage collectif sous la direction de D. Coeurjolly, JM Chassery, A. Montanvert - traité IC2 Hermès
- Image Processing, Analysis and Machine Vision - M. Sonka, V. Hlavac, R. Boyle - Chapman & Hall Computing
- Introductory Techniques for 3-D Computer Vision - E. Trucco, A. Verri - Prentice Hall

Sommaire

- 1 Introduction
- 2 La segmentation d'images : généralités
- 3 Segmentation : approches région
- 4 Segmentation : Approche frontière
- 5 Autres approches de la segmentation

Segmentation : approche région ou frontière I

- **segmentation** : décomposition d'une image en régions qui ont un sens (?), les “objets” de l'image.
- **segmentation** = étiquetage des pixels/voxels de l'image.
- pixels/voxels de même étiquette = pixels/voxels de même région



Segmentation : approche région ou frontière II

- Facile pour un être humain : connaissances préalables, image vue dans sa totalité, déductions (par exemple pour les frontières cachées)
- Méthodes :
 - Approche **région** : grouper pixels/voxels semblables
⇒régions homogènes.
 - Approche **frontière** : rechercher pixels/voxels dissemblables
⇒contours/surfaces entre zones hétérogènes
 - (Approche **hybride** : mélange des deux précédentes)

Segmentation d'image : exemple 1

Étiquetage en deux composantes
(Approche région : champ de Markov)

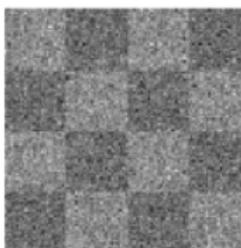


Image bruitée

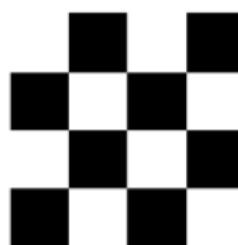
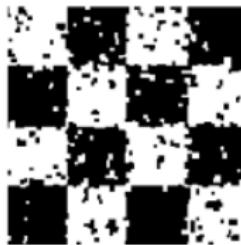
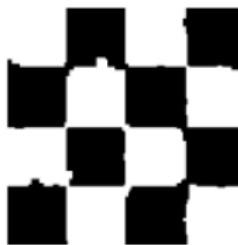


Image segmentée idéale



segmentation par ICM



Segmentation par RS

Segmentation d'image : exemple 2

Étiquetage en quatre composantes
(Approche région : champ de Markov)

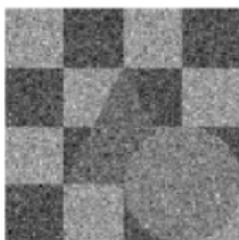


Image bruitée



Image segmentée idéale



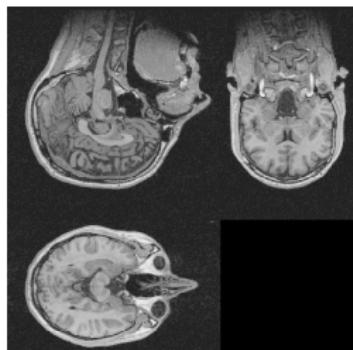
segmentation par ICM



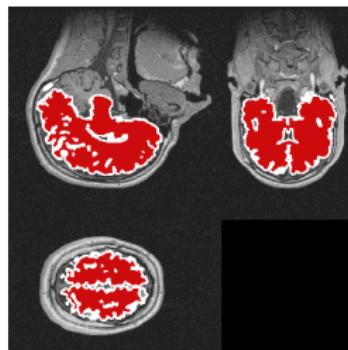
Segmentation par RS

Exemple d'application

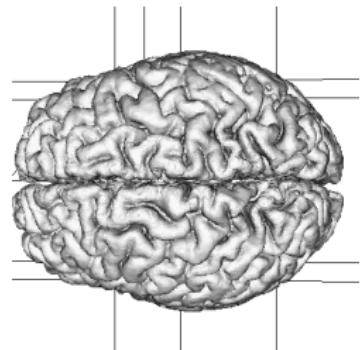
Reconstruction 3D du cortex cérébral
(Approche contour : modèle déformable)



Images 3D (3 coupes)



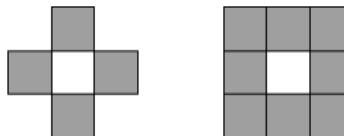
Segmentation



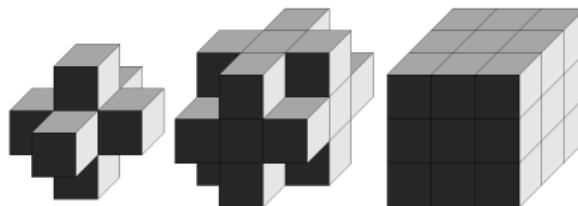
Reconstruction 3D

Définitions - Notations I

- **pixel/voxel** : élément de $X \subset \mathbb{Z}^2 / \mathbb{Z}^3$
- **connexités 2D** : 4-connexité / 8-connexité



- **connexités 3D** : 6-connexité / 18-connexité / 26-connexité



- **région** R : partie (connexe) de X
- **taille** $|R|$ de R = nb de pixels/voxels de R
- **bord** d'une région R : δR = contour interpixel de R (suite des arêtes de pixels séparant R et son complémentaire)
 $|\delta R|$ = longueur en nombre d'arêtes

Définitions - Notations II

- **image** : application I de X à valeurs dans $E =$
 - 1 $\{0, 1\}$: image binaire
 - 2 $\{0, \dots, 255\}$: image en niveaux de gris 8 bits
 - 3 $\{0, \dots, 2^{16} - 1\}$: image en niveaux de gris 16 bits
 - 4 $\{0, \dots, 255\}^3$: image couleur (composantes RGB)
 - 5 ...
- **histogramme** h : application de E dans \mathbb{Z}^+ des occurrences de chaque valeur de I
- **valeur moyenne** d'une région R : $\mu_R = \frac{1}{|R|} \sum_{p \in R} I(p)$.
- **variance** d'une région R : $\sigma_R^2 = \frac{1}{|R|} \sum_{p \in R} (I(p) - \mu_R)^2$.
- **segmentation** : application de X dans un espace d'étiquettes (généralement $1, \dots, K$).

Segmentation : première formalisation

Étiqueter \equiv partitionner

Définition de [Horowitz75]

X : domaine de l'image /

P : prédicat défini sur l'ensemble des parties de X , dépend de /
segmentation de X : $(S_i)_{i=1..n}$, sous-ensembles de X tels que

- ① $X = \bigcup_{i=1}^n S_i$ (Partition de X)
- ② $\forall i \in 1..n$, S_i est connexe et $P(S_i) = \text{vrai}$
- ③ $\forall i, j \in 1..n$, S_i adjacent à S_j et $i \neq j \Rightarrow P(S_i \cup S_j) = \text{faux}$

Exemples de prédicat d'homogénéité :

- $P(R) = \text{Vrai} \Leftrightarrow \sigma_R < 5$
- $P(R) = \text{Vrai} \Leftrightarrow \forall p \in R, |I(p) - \mu_R| < 10$

Exercice

$$\text{Image } I = \begin{matrix} 100 & 45 \\ 55 & 0 \end{matrix}$$

Partition de l'image I ?

$$P(R) = \text{Vrai} \Leftrightarrow \forall p \in R, |I(p) - \mu_R| < 30$$

Difficultés

Unicité ? Stabilité ? Calculabilité ?

Sommaire

- 1 Introduction
- 2 La segmentation d'images : généralités
- 3 Segmentation : approches région
 - Seuillage / classification
 - Méthodes de division et fusion
 - Méthodes d'agrégation
- 4 Segmentation : Approche frontière
- 5 Autres approches de la segmentation

3 - Segmentation : approches région

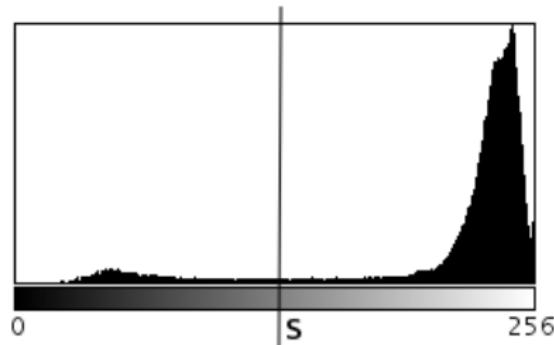
Seuillage / classification

Méthodes de division et fusion

Méthodes d'agrégation

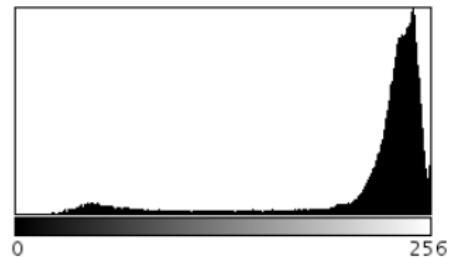
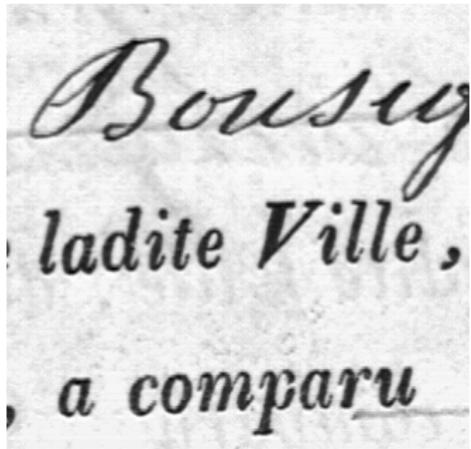
Segmentation par seuillage

- But : affecter chaque pixel d'une image en niveaux de gris à une **classe**. classes = intervalles de niveaux de gris
- Principe :
 - extraire des seuils à partir de l'histogramme (image/région)
 - classification d'un pixel p par comparaison de $I(p)$ aux seuils



Seuillage et histogramme I

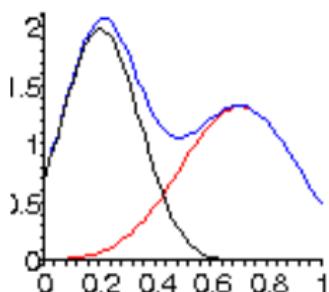
- histogramme de I dans $R \subset X$
≈ **distribution** des valeurs dans R .
- Exemple : objet assez uniforme
⇒ histogramme ≈ gaussienne de variance faible
- histogramme **bimodal** : partie de X avec deux objets de moyennes différentes



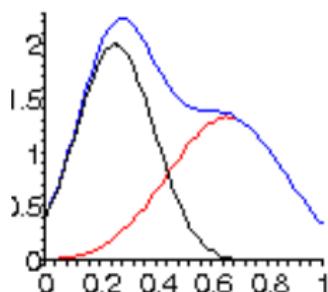
- Seuillage : trouver le(s) **seuil(s)** qui sépare(nt) au mieux les deux objets (ou plus).

Seuillage et histogramme II

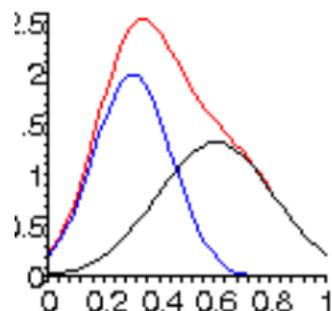
Devient difficile lorsque les moyennes se rapprochent.



$$(\mu_1 = 0.2, \sigma_1 = 0.2) \\ (\mu_2 = 0.7, \sigma_2 = 0.3)$$



$$(\mu_1 = 0.25, \sigma_1 = 0.2) \\ (\mu_2 = 0.65, \sigma_2 = 0.3)$$



$$(\mu_1 = 0.3, \sigma_1 = 0.2) \\ (\mu_2 = 0.6, \sigma_2 = 0.3)$$

Segmentation par seuillage

pixel : (x, y) ,

niveau de gris : $I(x, y)$

propriété locale : $P(x, y)$

seuil utilisé pour classer le pixel (x, y) : $S(x, y)$

3 types de méthodes de seuillage :

- seuillage *global* : $S(x, y) \stackrel{\text{def}}{=} S(I(x, y))$
- seuillage *local* : $S(x, y) \stackrel{\text{def}}{=} S(I(x, y), P(x, y))$
- seuillage *dynamique* : $S(x, y) \stackrel{\text{def}}{=} S(I(x, y), P(x, y), x, y)$

Si 2 classes, on parle de **binarisation**.

Exemple de seuillage global 1/3

Binarisation [Otsu79]

- Découpe de l'histogramme h de façon à minimiser l'erreur de partition.
- Idée : minimiser la variance à l'intérieur de chaque classe (C_1 et C_2). $p(n)$ probabilité du niveau de gris n , t seuil

$$p(C_1) = \sum_0^t p(n), \mu_{C_1} = \frac{\sum_0^t np(n)}{p(C_1)}, \sigma_{C_1}^2 = \frac{\sum_0^t (n - \mu_{C_1})^2 p(n)}{p(C_1)}$$

$$p(C_2) = \sum_{t+1}^N p(n), \mu_{C_2} = \frac{\sum_{t+1}^N np(n)}{p(C_2)}, \sigma_{C_2}^2 = \frac{\sum_{t+1}^N (n - \mu_{C_2})^2 p(n)}{p(C_2)}$$

$$\sigma_{intra}^2 = p(C_1)\sigma_{C_1}^2 + p(C_2)\sigma_{C_2}^2$$

$$\sigma_{inter}^2 = p(C_1)p(C_2)(\mu_{C_1} - \mu_{C_2})^2$$

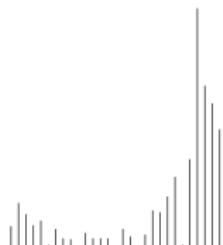
Minimiser σ_{intra}^2 est équivalent à maximiser σ_{inter}^2

Exemple de seuillage global 2/3

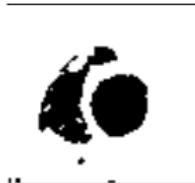
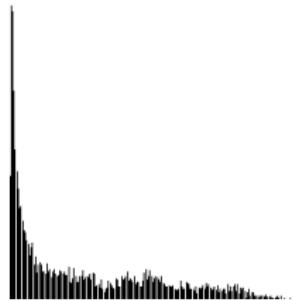
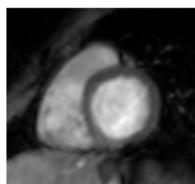
CE Recueil de réflexio
vations, sans ordre, &
suite, fut commencé pou
une bonne mère qui fait :

CE Recueil de réflexio
vations, sans ordre, &
suite, fut commencé pou
une bonne mère qui fait :
seuil = 50

CE Recueil de réflexio
vations, sans ordre, &
suite, fut commencé pou
une bonne mère qui fait :
seuil = 128



Exemple de seuillage global 3/3



seuil = 130



seuil = 82

inversion N/B

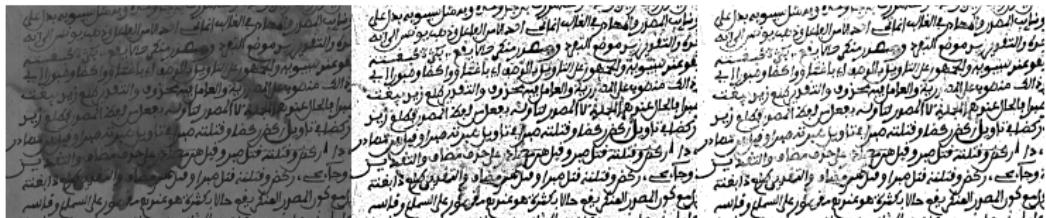
Exemple de seuillage local 1/2

Binarisation [Sauvola00]

Idée : le seuil est adapté au contraste local.

$$S(x, y) = \mu(x, y)(1 + k(\frac{\sigma(x, y)}{R} - 1))$$

- $\mu(x, y)$ moyenne des ng dans la fenêtre de calcul
- $\sigma(x, y)$ écart type des ng dans la fenêtre de calcul
- valeurs standard des paramètres $k = 0.5$, $R = 128$



$k=0.1$, $k=0.2$, rayon de la fenêtre = 15

Exemple de seuillage local 1/2

Binarisation [Bhanu01]

Idée : si un point appartient à C_i , la majorité de ses voisins (voisinage 8-connexe) appartient aussi à C_i .

- $P(p) = (P_{C_1}(p), P_{C_2}(p))$ vecteur de probabilité associé à un pixel p
 $P_{C_i}(p)$: probabilité que le pixel p appartienne à C_i
 $P_{C_1}(p) + P_{C_2}(p) = 1$
- $Q(p) = (Q_{C_1}(p), Q_{C_2}(p))$ vecteur de compatibilité associé à un pixel p
 $Q_{C_i}(p) = \frac{1}{8} \sum_{q \in V_8(p)} P_{C_i}(q)$
- Critère à maximiser :
 $\sum_{image} (P_{C_1}(p)Q_{C_1}(p) + P_{C_2}(p)Q_{C_2}(p))$

Exemple de seuillage local 2/2

Binarisation [Bhanu01]

Algorithme itératif : μ seuil initial.

$$P_{C_1}^0(p) = \begin{cases} \frac{I(p)-\mu}{NgMax} + 0.5 & \text{si } I(p) > \mu \\ \eta \frac{I(p)-\mu}{NgMax} + 0.5 & \text{sinon (}\eta\text{ coef entre 0.5 et 1)} \end{cases}$$

$$P_{C_1}^{n+1}(p) = \begin{cases} (1 - \alpha_1)P_{C_1}^n(p) + \alpha_1 & \text{si } Q_{C_1}^n(p) > 0.5 \\ (1 - \alpha_2)P_{C_1}^n(p) & \text{sinon} \end{cases}$$

- α_1, α_2 coefs entre 0 et 1.

⇒ on renforce la probabilité pour p d'appartenir à C_1 si la probabilité d'appartenance à C_1 de son voisinage est forte.

- Itérations successives jusqu'à majorité (90%) des pixels bien étiquetés ($P_{C_1}(p) > 0.9$ ou $P_{C_2}(p) > 0.9$)

Exemple de seuillage dynamique I

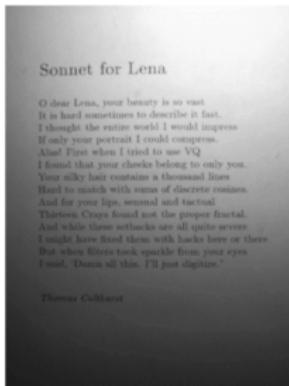
Binarisation [Chow,Kaneko72]

- Image découpée en blocs
- Calcul d'un seuil pour chaque bloc :
l'histogramme du bloc est-il bimodal ?
 - Si oui, le seuil trouvé est affecté au centre du bloc
 - Si non, le seuil prend pour valeur la moyenne des seuils des blocs voisins.
- Le seuil de chaque pixel est calculé par interpolation bilinéaire à partir des seuils des centres des blocs voisins.

Problème lié au découpage en blocs : régions tronquées.

Exemple de seuillage dynamique II

Binarisation [Chow,Kaneko72]



Sonnet for Lena

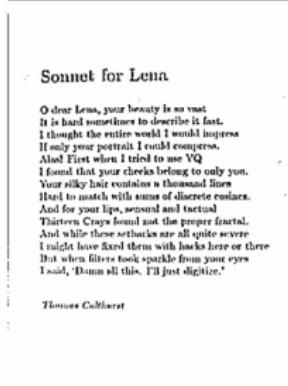
O dear Lena, your beauty is so vast
It is hard sometimes to describe it fast.
I thought the entire world I could impress
If only your portrait I could compass.
Alas! First when I tried to use VQ
I found that your cheeks belong to only you.
Your silky hair contains a thousand lines
Hard to match with sums of discrete cosines.
And for your lips, sensual and tactful
Thirteen Crays found not the proper fractal.
And while these setbacks are all quite severe
I might have fixed them with locks here or there
But when filters took sparkle from your eyes
I said, "Damn all this, I'll just digitize."

Thomas Collier



Sonnet for Lena

O dear Lena, your beauty is so vast
It is hard sometimes to describe it fast.
I thought the entire world I could impress
If only your portrait I could compass.
Alas! First when I tried to use VQ
I found that your cheeks belong to only you.
Your silky hair contains a thousand lines
Hard to match with sums of discrete cosines.
And for your lips, sensual and tactful
Thirteen Crays found not the proper fractal.
And while these setbacks are all quite severe
I might have fixed them with locks here or there
But when filters took sparkle from your eyes
I said, "Damn all this, I'll just digitize."



Sonnet for Lena

O dear Lena, your beauty is so vast
It is hard sometimes to describe it fast.
I thought the entire world I could impress
If only your portrait I could compass.
Alas! First when I tried to use VQ
I found that your cheeks belong to only you.
Your silky hair contains a thousand lines
Hard to match with sums of discrete cosines.
And for your lips, sensual and tactful
Thirteen Crays found not the proper fractal.
And while these setbacks are all quite severe
I might have fixed them with locks here or there
But when filters took sparkle from your eyes
I said, "Damn all this, I'll just digitize."

Thomas Collier

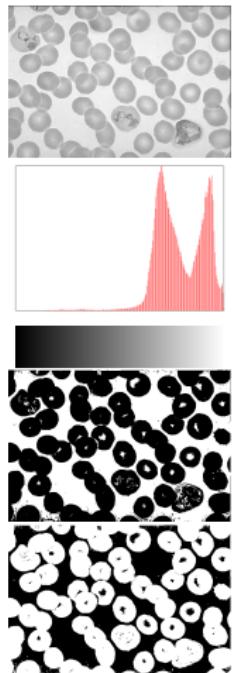
Classification

Principe :

- Pic de l'histogramme \simeq composante image.
- Découper l'histogramme en k classes.
- Etiqueter chaque pixel avec le numéro de sa classe.

Découpage en k classes :

- Utilisateur.
- Détection de “vallées” sur l'histogramme.
- Approximation par mélange de k Gaussiennes.
- Algorithme des “ k -means”...



Exemple de classification : Méthode des k-means

Méthode itérative pour découper l'histogramme en k classes (k entier choisi au départ).

Algorithme :

- Choisir arbitrairement k couleurs $\{c_1, \dots, c_k\}$ dans l'histogramme.
- Tant que les c_i sont modifiées dans la boucle, faire
 - Pour chaque couleur de l'histogramme déterminer la couleur c_i la plus proche.
 - La classe C_i est l'ensemble des couleurs qui sont plus proches de c_i que de n'importe laquelle des c_j .
 - Remplacer chaque c_i par la moyenne de sa classe C_i (ici, le nombre de pixels de l'image qui ont la couleur c_i intervient dans le calcul).
- Les classes “résultats” sont les C_i .

Méthode des k-means : exemple

recherche de 2 classes



image



1



2



3



4



5



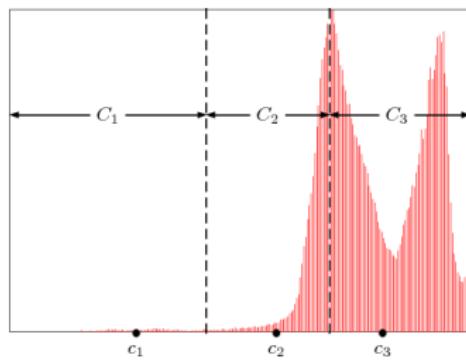
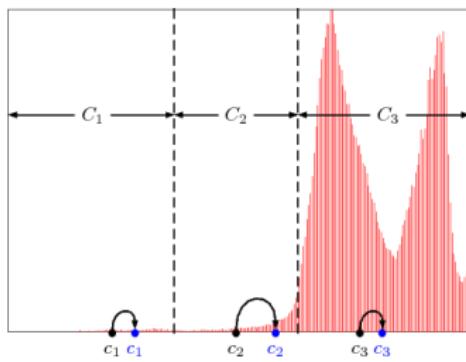
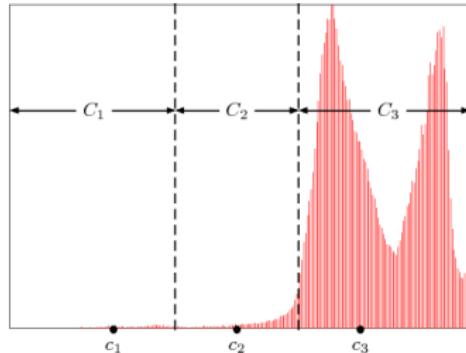
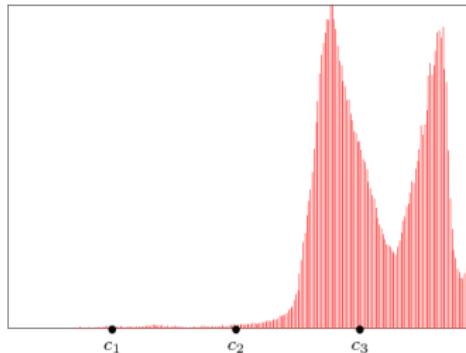
6



résultat

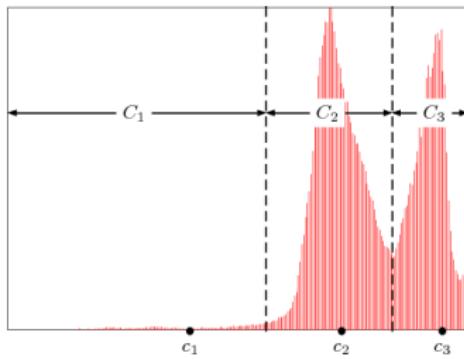
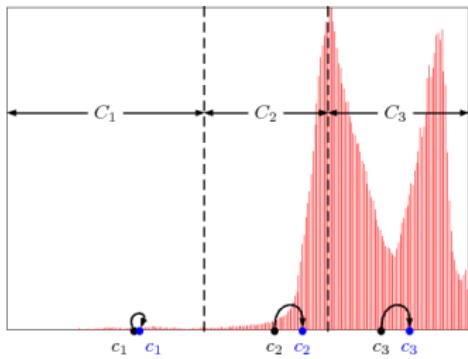
Méthode des k-means : exemple I

recherche de 3 classes



Méthode des k-means : exemple II

recherche de 3 classes



Binarisation / classification (fin)

- Intérêt : étiquetage très rapide basé sur l'histogramme de l'image.
- souvent initialisation d'un algorithme de segmentation plus haut niveau, qui prend en compte la *localisation* des valeurs des pixels.
- les étiquettes sont souvent filtrées par **relaxation** : pour tout pixel $[x, y]$, la nouvelle étiquette de $[x, y]$ est l'étiquette qui apparaît le plus grand nombre de fois dans un voisinage de $[x, y]$.
- Il existe des techniques de **classification floue** qui attribuent pour chaque couleur une probabilité d'appartenance à une classe.
fuzzy c-means [Dunn73, Bezdek81]

3 - Segmentation : approches région

Seuillage / classification

Méthodes de division et fusion

Méthodes d'agrégation

Division

Approche top-down

Focalisation de l'attention : de l'échelle la plus grossière vers les détails les plus fins.

Principe

Point de départ = une sous-segmentation (toute l'image, une région à raffiner)

Découper les régions non homogènes

Division récursive

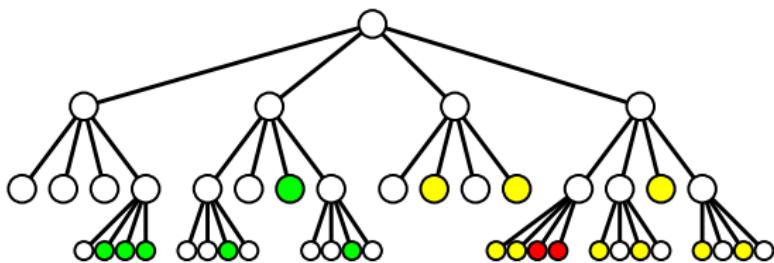
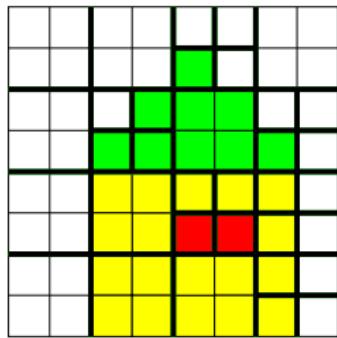
- ➊ Partitionner la région, par exemple à partir d'une partition de son histogramme.
- ➋ Pour chaque région résultante, si possible (et nécessaire) revenir en 1.

Problème : on ne revient pas sur une région découpée à tort.

Division par quadtree 1/2

Codage de l'image par un arbre - définition récursive :

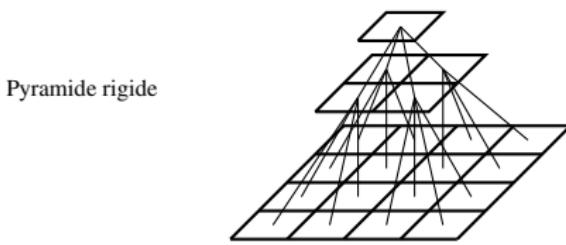
- racine : image entière
- découpage en 4 de la portion d'image correspondant à un noeud si elle n'est pas homogène \Rightarrow chaque noeud a 4 fils.



Méthode de segmentation par division basée sur un critère d'homogénéité (et pas sur l'histogramme).

Division par quadtree 2/2

- le quadtree est un cas particulier de pyramide, une pyramide étant une suite de graphes représentant une image à différents niveaux de résolution.



- inconvénient : les découpes sont contraintes par le cadre rigide de la définition
- utilisable comme partition initiale pour une méthode de fusion.

Fusion

Approche bottom-up

Principe

Point de départ = une partition de l'image en régions homogènes

Fusionner tout couple de régions adjacentes qui vérifie un critère d'homogénéité.

⇒ définir un prédicat *Fusionne*(R_i, R_j) où R_i et R_j sont deux régions adjacentes.

Caractéristiques d'une région R_i :

μ_i : moyenne des n. g.

σ_i : écart type des n. g.

$|R_i|$: nombre de pixels

δR_i contour, $|\delta R_i|$ périmètre

$|\delta R_i \cap \delta R_j|$ longueur de la frontière commune à R_i et R_j

Tests d'homogénéité pour la fusion

On peut simplement mesurer l'homogénéité de la région
 $R = R_i \cup R_j$:

- la variance des n. g. des pixels de R est inférieure à un seuil
- la proportion des pixels de R dont le n. g. est à l'extérieur de $[\mu_R - \sigma_R, \mu_R + \sigma_R]$ est inférieure à un seuil
- autres ?

Critère de [Beveridge89] pour la fusion

$$f(R_i, R_j) = f_{sim}(R_i, R_j) \sqrt{f_{taille}(R_i, R_j)} f_{cont}(R_i, R_j)$$

- Critère de similarité :

$$f_{sim}(R_i, R_j) = \frac{|\mu_i - \mu_j|}{\max(1, \sigma_i + \sigma_j)}$$

- Critère de taille :

$$f_{taille}(R_i, R_j) = \min\left(2, \frac{\min(|R_i|, |R_j|)}{T_{opt}}\right), T_{opt} : \text{fixé suivant la taille de l'image}$$

- Critère de frontière commune :

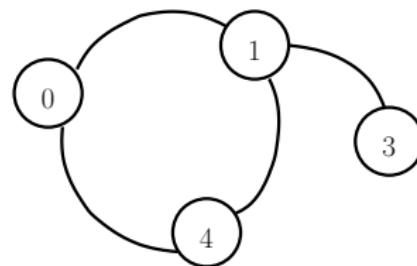
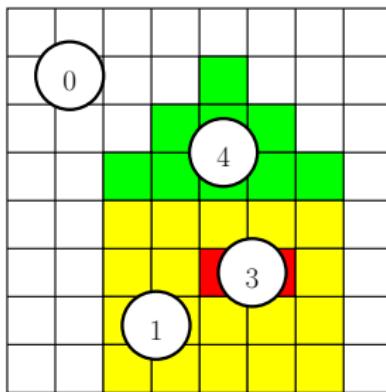
$$f_{cont}(R_i, R_j) = \begin{cases} C(R_i, R_j) & \text{si } \frac{1}{2} \leq C(R_i, R_j) \leq 2 \\ \frac{1}{2} & \text{si } C(R_i, R_j) < \frac{1}{2} \\ 2 & \text{sinon} \end{cases}$$

$$C(R_i, R_j) = \frac{\min(|\delta R_i|, |\delta R_j|)}{4|\delta R_i \cap \delta R_j|}$$

Structure de données pour la fusion : RAG

Graphe d'adjacence de régions (Region Adjacency Graph) :
graphe non orienté où les noeuds correspondent aux régions
 \exists arête entre 2 noeuds ssi les 2 régions correspondantes sont adjacentes.

A utiliser avec une description géométrique des régions.



fusion de régions adjacentes = contraction d'une arête +
suppression éventuelle d'arêtes multiples

Fusion par pyramides adaptatives 1/3

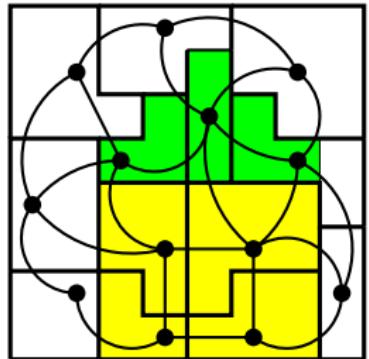
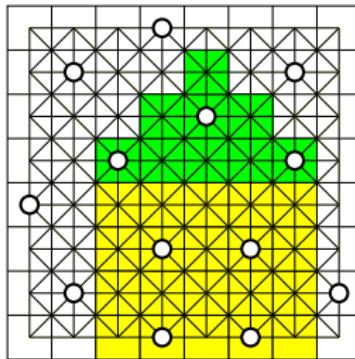
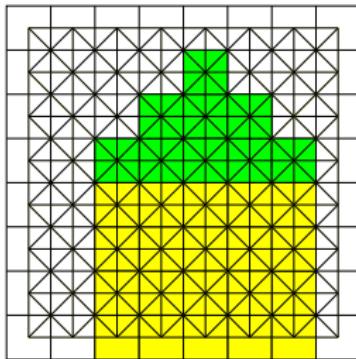
- Structure de données hiérarchique dédiée à un algorithme de fusion spécifique : fusion de groupes de régions et non de couples de régions.
- Une pyramide est un ensemble de graphes reliés entre eux, chaque graphe représentant une partition de l'image (graphe d'adjacence).

Algorithme :

- Base de la pyramide : image initiale, maille 8-connexe.
- Passage d'un niveau au suivant :
 - 1 Extraction des sommets survivants
 - 2 Fusion.

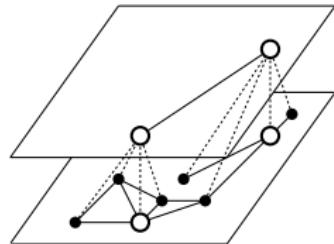
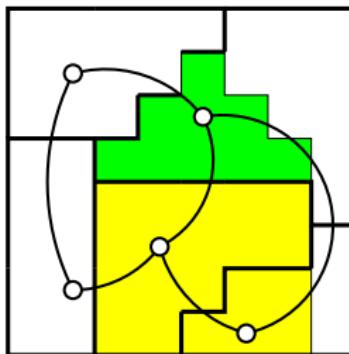
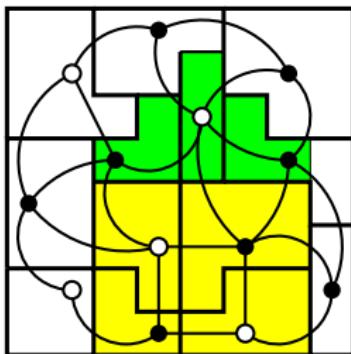
Fusion par pyramides adaptatives 2/3

- 1 Extraction des survivants. Deux survivants ne doivent pas être adjacents et tout non survivant est adjacent à au moins un survivant.
Critère de choix : minimum local pour la variance par exemple + vérification des contraintes
- 2 Fusion. On fusionne chaque sommet non survivant avec un sommet survivant adjacent (choix du plus similaire).



Pyramides adaptatives 3/3

- Puis on itère...



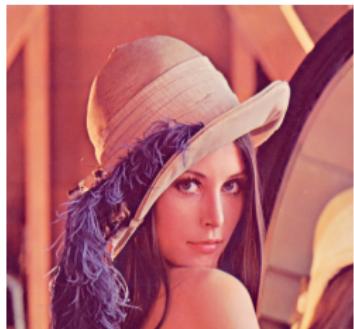
Intérêt : réduction rapide du graphe, résultat indépendant du sens de parcours de l'image.

Division / fusion (split and merge)

- Segmentation initiale quelconque (par exemple $(R_i)_{i=1..n}$ découpage de l'image en blocs de taille fixe)
- Subdiviser les régions non homogènes
 $\forall i \in 1..n$, $P(R_i)$ faux \Rightarrow découper R_i
- Fusionner les régions non maximales
 R_i adj R_j et $Fusionne(R_i, R_j)$ vrai \Rightarrow fusionner R_i et R_j

De meilleurs résultats sont obtenus en alternant divisions et fusions.

Division / fusion : exemple 1



Lenna

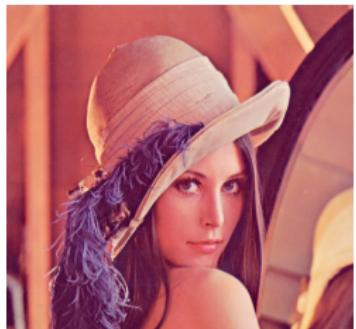


Après division



Après fusion

Division / fusion : exemple 2



Lenna



Sélection région



Division

Division / fusion : exemple 3



après division



après fusion

3 - Segmentation : approches région

Seuillage / classification

Méthodes de division et fusion

Méthodes d'agrégation

Méthodes d'agrégation de pixels

Region growing

- Sélection de groupes de pixels dans les zones homogènes de l'image.
- Croissance des germes par ajout de pixels connexes **similaires**



Méthodes d'agrégation de pixels

```
growSeed(seed)
    R.init()
    R.add(seed)
        while R.hasNeighbor()
            p = R.getNextNeighbor()
            if pred(p, R)
                R.add(p)
```

Exemple de définition de $\text{pred}(p, R)$: $\frac{|I(p) - \mu_R|}{\sigma_R} \leq T$

Variante adaptant le critère à la taille de la région :

$$\frac{w(|R|)}{T_1} \frac{|I(p) - \mu_R|}{\sigma_R} + \frac{(1-w(|R|))}{T_2} \sigma_{R \cup p} \leq 1$$

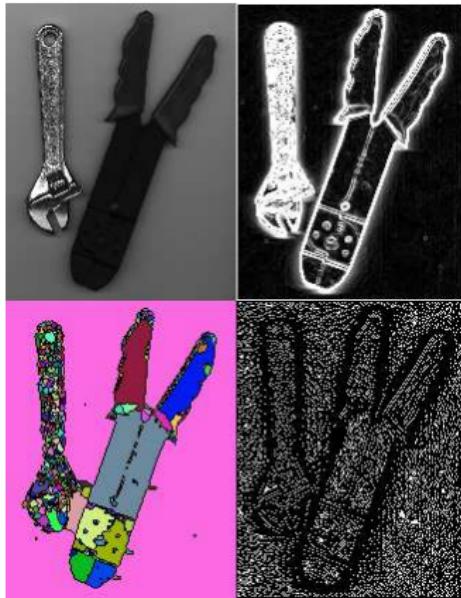


Autres : critère géométrique, compétition entre régions croissant en parallèle

Ligne de partage des eaux

Idée

Détection des bassins d'écoulement sur l'image de la norme du gradient



Ligne de partage des eaux

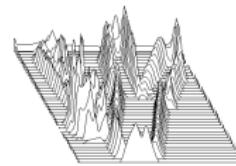
Exemple



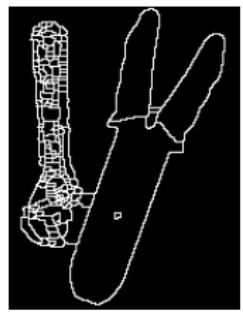
image



Norme gradient



carte d'élévation

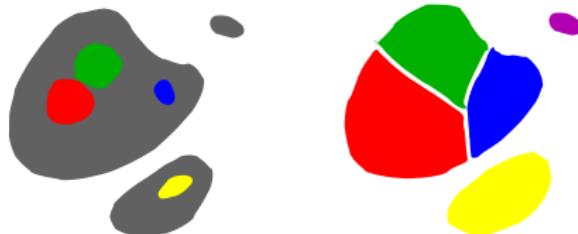
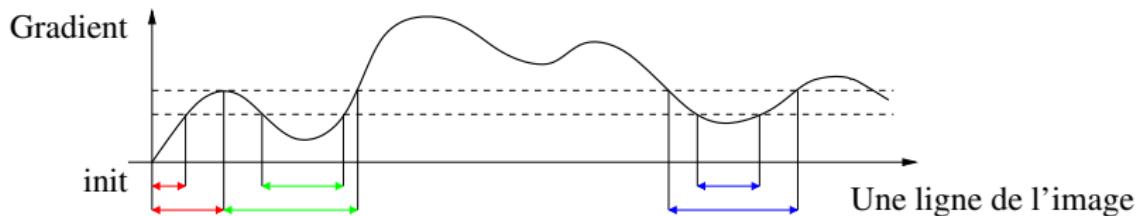


bassins

Ligne de partage des eaux

Calcul par immersion

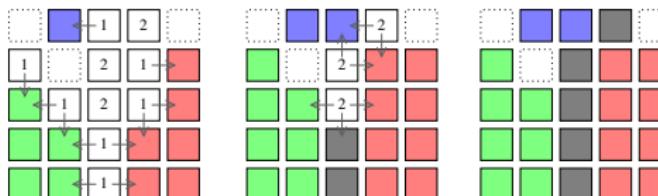
- germes : pixels de plus faible gradient
- étape i : niveau $\nearrow \Rightarrow$ nouveaux pixels (p_i) de gradient plus élevé
Si p_i touche un bassin existant, l'agréger à ce bassin
Sinon p_i est un nouveau germe (bassin)



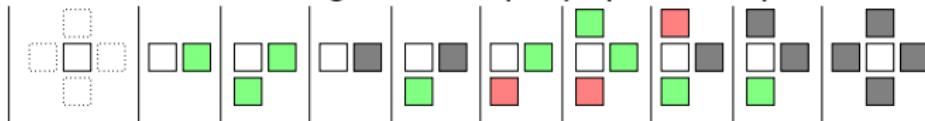
Ligne de partage des eaux

Algorithme [Vincent,Soille91]

	UNLABELLED
0	MASK (distance)
	WATERSHED
	Label0
	Label1
	Label2



- Trier les pixels par ordre croissant d'altitude
- Par groupe de pixels (p_i) de même altitude
 - traiter les pixels par distance à un bassin existant
 - examiner le voisinage de chaque p_i pour l'étiqueter



- Si un p_i non étiqueté (nouveau bassin), affectation d'une nouvelle étiquette et propagation de cette étiquette à ses voisins non étiquetés

Ligne de partage des eaux

Sensibilité au bruit

beaucoup de minima \Rightarrow beaucoup de petites régions (on peut lisser l'image de la norme du gradient pour limiter ce défaut).



Ligne de partage des eaux

Marqueurs

Des marqueurs donnés par un utilisateur permettent d'éviter la sur-segmentation.

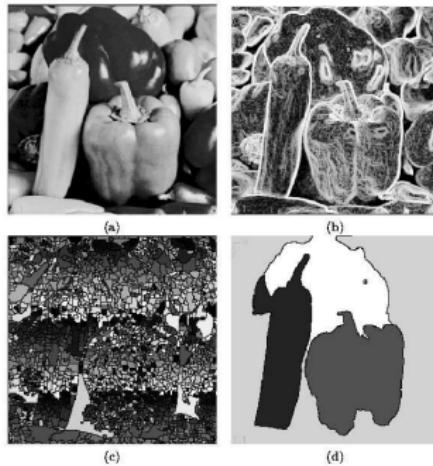


Figure 5.51: Watershed segmentation: (a) original; (b) gradient image, 3×3 Sobel edge detection, histogram equalized; (c) raw watershed segmentation; (d) watershed segmentation using region markers to control oversegmentation. Courtesy W. Higgins, Penn State University.

Ligne de partage des eaux

Dépendance vis-à-vis du détecteur de contours

Ligne de partage des eaux calculée sur l'image du Laplacien.

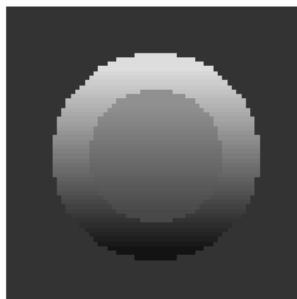


Sommaire

- 1 Introduction
- 2 La segmentation d'images : généralités
- 3 Segmentation : approches région
- 4 Segmentation : Approche frontière
- 5 Autres approches de la segmentation

Segmentation : approche frontière

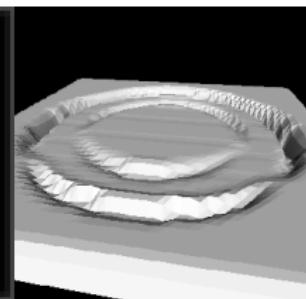
Recherche des contours à partir du gradient



Image



Norme du gradient



Norme du gradient seuillée



Gradient d'une image 2D

Niveau de gris de l'image au point (x, y) : $I(x, y)$

Gradient : $\nabla I(x, y) = (G_x, G_y) = \left(\frac{\partial I(x, y)}{\partial x}, \frac{\partial I(x, y)}{\partial y} \right)$

Module/Norme du gradient : $G = \sqrt{G_x^2 + G_y^2}$

Orientation du gradient : $\theta = \arctan\left(\frac{G_y}{G_x}\right)$

Exemple de calcul simple : masques de Sobel

$y \downarrow \rightarrow x$

-1	0	1
-2	0	2
-1	0	1

-1	-2	-1
0	0	0
1	2	1

Gradient d'une image 3D

Niveau de gris de l'image au point (x, y, z) : $I(x, y, z)$

Gradient :

$$\nabla I(x, y, z) = (G_x, G_y, G_z) = \left(\frac{\partial I(x, y, z)}{\partial x}, \frac{\partial I(x, y, z)}{\partial y}, \frac{\partial I(x, y, z)}{\partial z} \right)$$

Module/Norme du gradient : $G = \sqrt{G_x^2 + G_y^2 + G_z^2}$

Orientation du gradient : $\theta = \arctan\left(\frac{G_y}{G_x}\right)$, $\phi = \arctan\left(\frac{G_z}{G_x}\right)$

Masque de Sobel pour le calcul de G_y (coupes suivant l'axe z)

$y \downarrow$

-1	-2	-1	-2	-4	-2	-1	-2	-1
0	0	0	0	0	0	0	0	0
1	2	1	2	4	2	1	2	1

Image des contours

Calcul du gradient en chaque point de l'image

- Filtres simples : Prewitt, Sobel
- Filtre de Canny : $\frac{\partial I}{\partial x}$ calculé par convolution avec $x \mapsto Axe^{-\frac{x^2}{2\sigma^2}}$ (dérivée de Gaussienne). Similaire pour y .
- Filtre de Deriche : $\frac{\partial I}{\partial x}$ calculé par convolution avec $x \mapsto Axe^{-\alpha|x|}$. Similaire pour y .
- Autres : filtre de Shen-Castan...

NB : très similaires en pratique, extensibles directement en 3D

Image des contours

Idée directrice

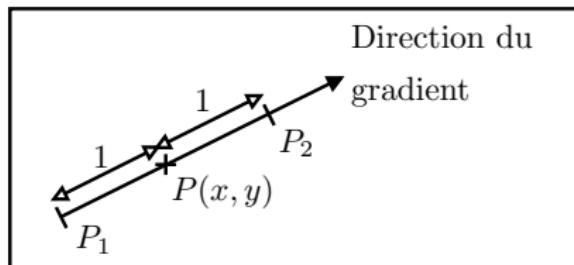
maximum local de gradient \Rightarrow présence d'un contour

Méthode pour obtenir une image des contours :

- ① estimation du gradient en chaque point de l'image
- ② extraction des maxima locaux de la norme du gradient dans la direction du gradient
- ③ sélection des maxima locaux significatifs par seuillage
- ④ fermeture des contours en traçant les chemins suivant une ligne de crête dans l'image de la norme du gradient

Image des contours

Extraction des maxima locaux de la norme du gradient



Gr : norme du gradient en P

Gr_1 : norme du gradient en P_1

Gr_2 : norme du gradient en P_2

Présence d'un maximum local : $Gr > Gr_1$ et $Gr > Gr_2$

Calcul de Gr_1 et Gr_2 par interpolation linéaire

Image des contours

Seuillage par hystérésis des maximums locaux

But : limiter la fragmentation des contours obtenus

2 seuils : $S_h > S_b$

On garde :

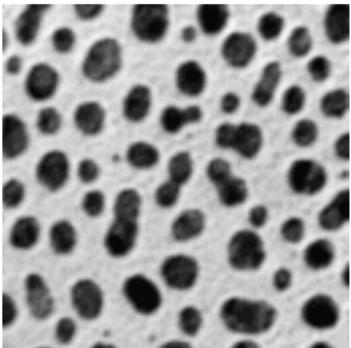
- les maximums locaux de valeur supérieure à S_h
- les maximums locaux de valeur supérieure à S_b
appartenant à une composante connexe de maxima
locaux ($\geq S_b$) contenant au moins une valeur $\geq S_h$

Résultat : une image binaire (image des contours)

Image des contours

Seuillage par hystérésis : exemple 1

Image à segmenter



Maxima locaux du gradient

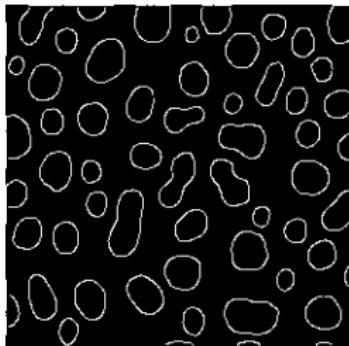
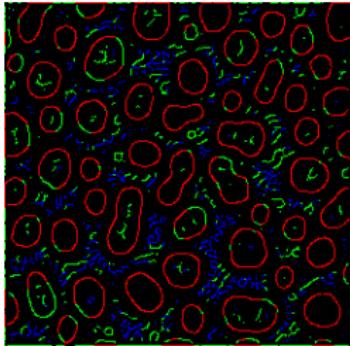
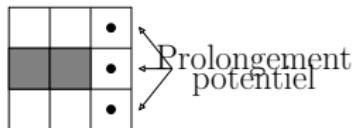


Image des contours

Fermeture des contours

Idée : suivre une ligne de crête dans l'image de la norme du gradient à partir de chaque extrémité de contour.

- 1 Repérer les points extrémité (énumération des configurations possibles)



- 2 Choix entre les points candidats : on explore tous les chemins possibles à partir de chaque point candidat. Le poids d'un chemin peut être défini comme la somme de la norme du gradient en chacun de ses points.

Sommaire

- 1 Introduction
- 2 La segmentation d'images : généralités
- 3 Segmentation : approches région
- 4 Segmentation : Approche frontière
- 5 Autres approches de la segmentation
 - Modèles déformables
 - Méthodes level-set
 - Formulations énergétiques de la segmentation

5 - Autres approches de la segmentation

Modèles déformables

Méthodes level-set

Formulations énergétiques de la segmentation

Modèles déformables ; contours actifs

Principe : approche variationnelle

- famille de formes possibles
- chaque forme a une énergie (un réel)
- recherche du minimum

$$E_{totale}(C) = E_{interne}(C) + E_{externe}(C)$$

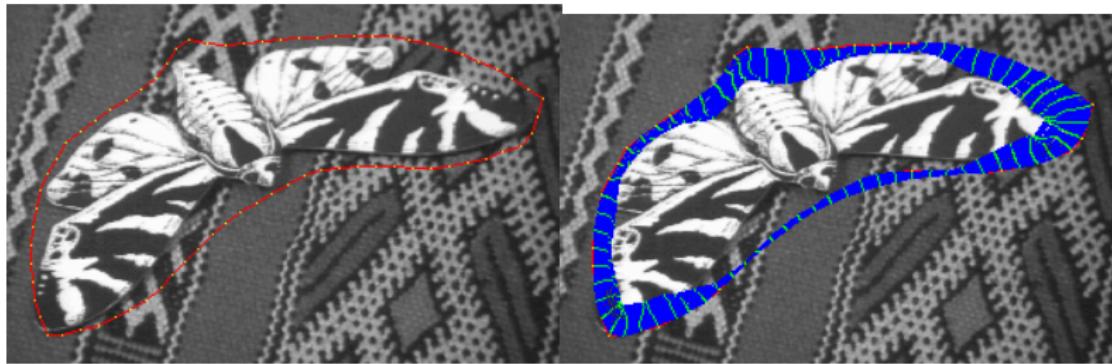
- **Energies externes** : recherche de contours dans une image, interaction avec l'utilisateur, ressemblance avec une forme prédéfinie, ...
- **Energies internes** : tension et torsion, facilitent l'extraction de régions dont le contour est flou ou fragmenté en **comblant** l'information manquante

Cadre très général : segmentation, stéréovision, suivi de formes dans les vidéos, ...

Contours actifs : exemple 2D

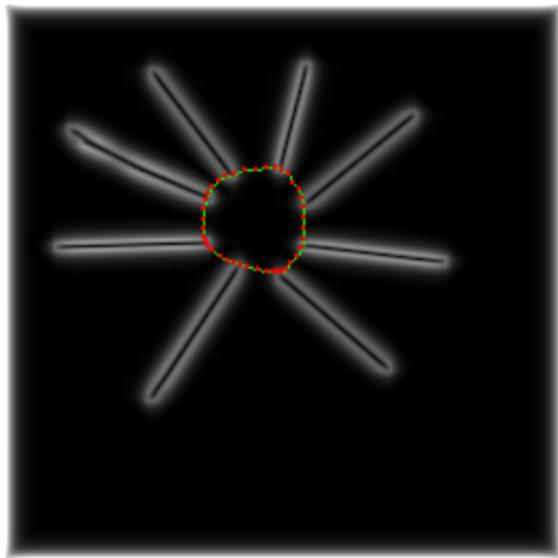
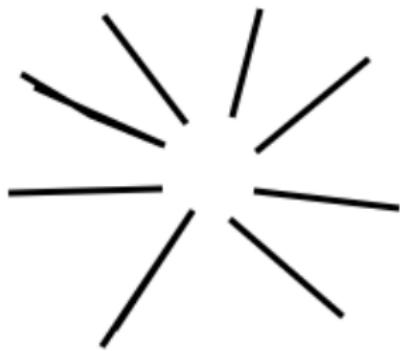
Contour actif ou **Snake** : optimisation itérative

- initialisation : courbe assez proche du contour à extraire
- itérations : déformations du contour actif de façon à ce qu'il atteigne une position d'énergie minimum.

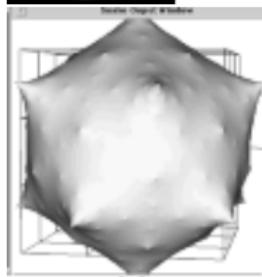
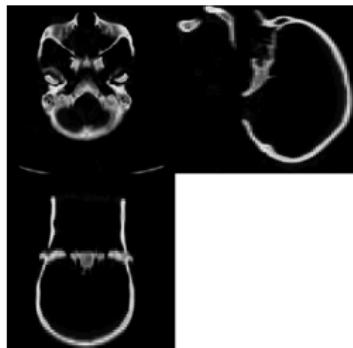


Contours actifs : exemple 2D

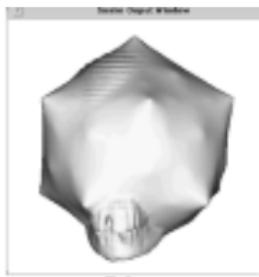
Extraction de formes “conceptuelles”



Contours actifs : exemple 3D



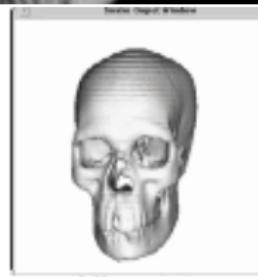
(a)



(b)



(c)



(d)

Contours actifs : formalisation 2D

[Kass et al. 87]

Représentation paramétrique du contour actif :

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

$$E_{interne}(C) = \int_0^1 \alpha \left| \frac{\partial v(s)}{\partial s} \right|^2 + \beta \left| \frac{\partial^2 v(s)}{\partial s^2} \right|^2 ds$$

⇒ un contour actif a une énergie interne faible lorsque c'est une courbe "régulière" (ni trop étiré, ni trop tordu).

$$E_{image}(C) = \lambda \int_0^1 -|\nabla I(v(s))|^2 ds$$

⇒ un contour actif a une énergie image faible lorsque qu'il est positionné sur un contour de l'image.
Il reste à trouver le contour qui minimise leur somme.

Surfaces actives : formalisation 3D

[Terzopoulos et al. 91]

Représentation paramétrique de la surface active :

$$S = \{v(r, s) = (x(r, s), y(r, s), z(r, s)); (r, s) \in [0, 1]^2\}$$

$$\begin{aligned} E_{interne}(S) &= \int_0^1 \int_0^1 \alpha_r \left| \frac{\partial v(r, s)}{\partial r} \right|^2 + \alpha_s \left| \frac{\partial v(r, s)}{\partial s} \right|^2 + \\ &\quad \beta_{rs} \left| \frac{\partial^2 v(r, s)}{\partial s \partial r} \right|^2 + \beta_{rr} \left| \frac{\partial^2 v(r, s)}{\partial r^2} \right|^2 + \beta_{ss} \left| \frac{\partial^2 v(r, s)}{\partial s^2} \right|^2 dr ds \end{aligned}$$

$$E_{image}(S) = \lambda \int_0^1 \int_0^1 -|\nabla I(v(r, s))|^2 dr ds$$

Contours actifs : calcul de l'évolution 2D

1- Discrétisation du contour en N points :

$$i = 0..N - 1, X[i] = x\left(\frac{i}{N}\right) \text{ ou } X(ih) \text{ avec } h = \frac{1}{N}.$$

$$X = \begin{bmatrix} \dots \\ x_i \\ \dots \end{bmatrix} \quad Y = \begin{bmatrix} \dots \\ y_i \\ \dots \end{bmatrix}$$

2- Energie externe : forces dérivant des énergies

$$f_x(X, Y) = \begin{bmatrix} \dots \\ \frac{\partial |\nabla I|^2}{\partial x}(x_i, y_i) \\ \dots \end{bmatrix} \quad f_y(X, Y) = \begin{bmatrix} \dots \\ \frac{\partial |\nabla I|^2}{\partial y}(x_i, y_i) \\ \dots \end{bmatrix}$$

Contours actifs : calcul de l'évolution 2D

Minimiser l'énergie est équivalent à résoudre (Euler-Lagrange) :

$$-\alpha x''(s) + \beta x^{(4)}(s) = \frac{\partial |\nabla I(v)|^2}{\partial x}$$

$$-\alpha y''(s) + \beta y^{(4)}(s) = \frac{\partial |\nabla I(v)|^2}{\partial y}$$

2- Discrétisation par différences finies :

$$\textcolor{red}{A}X = f_x(X, Y)$$

$$\textcolor{red}{A}Y = f_y(X, Y)$$

$$\textcolor{red}{A} = \frac{1}{h^2} \begin{bmatrix} 6\frac{\beta}{h^2} + 2\alpha & -4\frac{\beta}{h^2} - \alpha & \frac{\beta}{h^2} & 0 & \dots \\ -4\frac{\beta}{h^2} - \alpha & 6\frac{\beta}{h^2} + 2\alpha & -4\frac{\beta}{h^2} - \alpha & \ddots & \ddots \\ \frac{\beta}{h^2} & -4\frac{\beta}{h^2} - \alpha & 6\frac{\beta}{h^2} + 2\alpha & \ddots & \ddots \\ 0 & \ddots & \ddots & \ddots & \ddots \\ \vdots & \ddots & \ddots & \ddots & \ddots \end{bmatrix}.$$

Contours actifs : calcul de l'évolution 2D

3- **Résolution itérative** : calcul des positions successives du contour

Le paramètre γ est l'inverse du pas de temps, mais peut être interprété comme un coefficient de frottement.

A partir d'une position initiale (X_0, Y_0) . On itère :

$$\begin{aligned} X_t &= (A + \gamma I)^{-1}(\gamma X_{t-1} + f_x(X_{t-1}, Y_{t-1})) \\ Y_t &= (A + \gamma I)^{-1}(\gamma Y_{t-1} + f_y(X_{t-1}, Y_{t-1})) \end{aligned}$$

Le contour **se déforme** progressivement jusqu'à atteindre un minimum local de l'énergie.

5 - Autres approches de la segmentation

Modèles déformables

Méthodes level-set

Formulations énergétiques de la segmentation

Méthodes *level-set* : milieu évolutif

Contour déformable → Milieu déformable

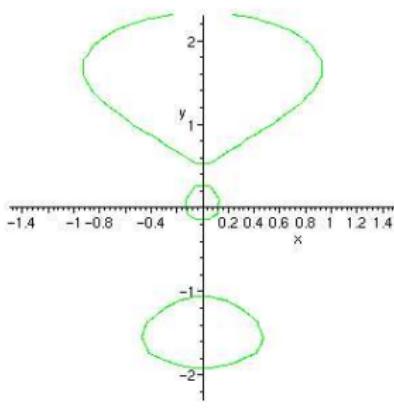
- Propagation de fronts ou approche **level-set**
- **Provenance.** En physique, modélisation de l'évolution des interfaces entre milieux. Ex : feux de prairie
- **Modèle.** Evolution de l'interface déduite de l'évolution du milieu tout entier [Osher et Séthian 88].
 ⇒ interface = isopotentielle dans un champ potentiel.
- **Segmentation.** [Malladi *et al.* 93], [Caselles *et al.* 93]
- **Objectif.** éviter les problèmes topologiques, extension nD naturelle
- **Difficultés.** Traduire l'adéquation modèle/image

Milieu évolutif : une hyper-surface en mouvement (I)

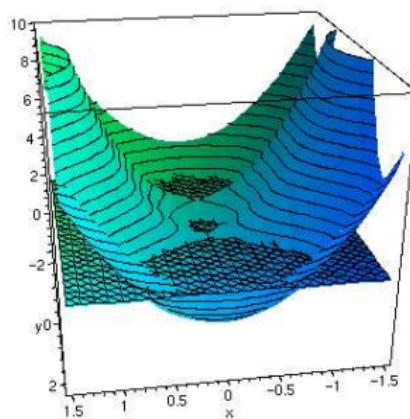
- modéliser l'évolution d'une courbe $C(t)$ dans le plan
- soit $f(t, \mathbf{x}) : [0, \infty[\times \mathbb{R}^2 \rightarrow \mathbb{R}$ une fonction scalaire du plan, telle que $f(t, \mathbf{x}) = \pm d$, avec d distance de \mathbf{x} à $C(t)$.
⇒ f est une sorte de carte de distance signée à $C(t)$.
- $S(t) = \{(\mathbf{x}, f(t, \mathbf{x}))\}$ est une (hyper-)surface de \mathbb{R}^3
 $f(t, \cdot)$ est la carte d'élévation au temps t .
- Si on coupe $S(t)$ par le plan $z = 0$, alors on obtient $C(t)$.

Milieu évolutif : une hyper-surface en mouvement (II)

- $C(t) = \text{niveau } 0 \text{ de la surface } S(t)$



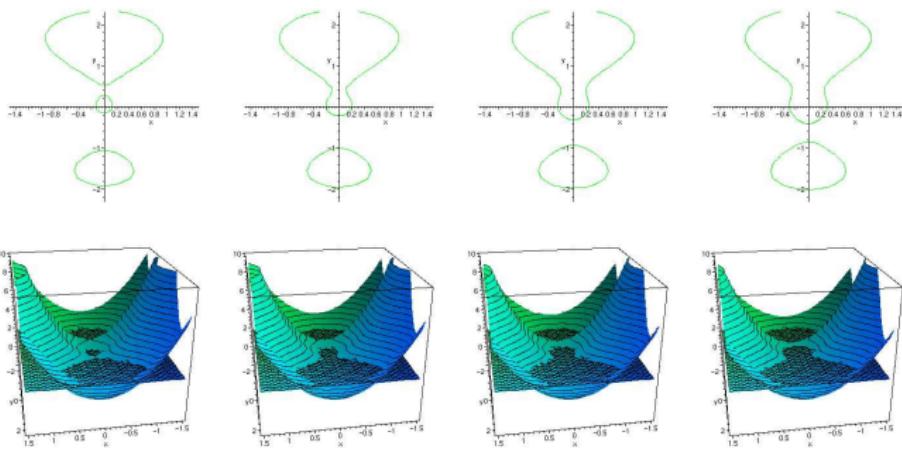
$\$C(t)\$$



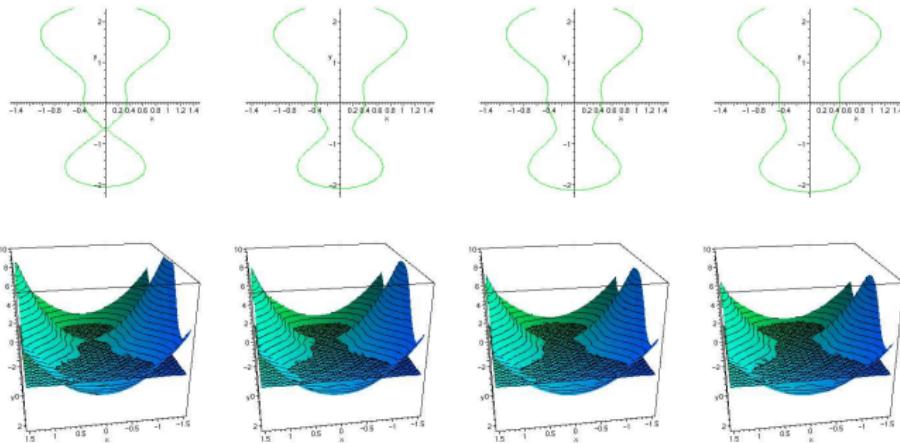
$\$S(t)\$$

Milieu évolutif : une hyper-surface en mouvement (III)

- L'hyper-surface $S(t)$ ne change *jamais* de topologie, $C(t)$ peut changer de topologie.

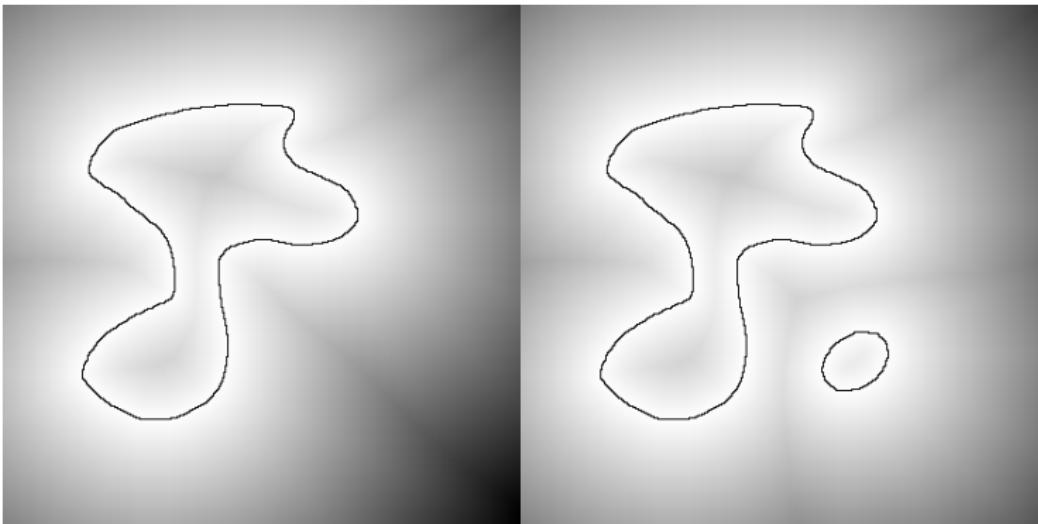


Milieu évolutif : une hyper-surface en mouvement (IV)



⇒ Déformer S (i.e. f) plutôt que C .

hyper-surface = carte de distance au contour



- $f(t, \mathbf{x})$: altitude en tout point \mathbf{x} en fonction du temps t
= distance signée au contour $C(t)$
- on va trouver un processus où :
évolution de $C \Leftrightarrow$ évolution de f

Évolution : idée générale

- le front se gonfle ou se dégonfle suivant sa normale.
- l'hyper-surface $S(t)$ se propage similairement au front $C(t)$.
 - ⇒ chaque niveau de la fonction f se déplace similairement au niveau 0 ($C(t)$).
- le front $C(t)$ a tendance à combler les trous
 - ⇒ le front va plus vite lorsque sa courbure est très négative
 - ⇒ lissage du front
- le front est ralenti s'il passe sur des forts contours de l'image
 - ⇒ vitesse du front = composition de la courbure locale et des contours image locaux.

Équation d'évolution

$$\frac{\partial C}{\partial t} = A(t)\mathbf{n}(t) \Leftrightarrow \frac{\partial f}{\partial t}_{|\mathbf{x}(t)} = -\underbrace{A(t)}_{\hat{A}_{|\mathbf{x}(t)}} |\nabla f_{|\mathbf{x}(t)}|$$

- Soit $\mathbf{x}(t)$ un point du front/contour se propageant à vitesse $A(t)$.
- propagation selon la normale au front $\mathbf{n}(t)$.
- On a $\forall t, f(t, \mathbf{x}(t)) = 0$ (le point reste sur le front).
- En différentiant : $\frac{df}{dt} = \frac{\partial f}{\partial t} + \mathbf{x}' \cdot \frac{\partial f}{\partial \mathbf{x}}$
- Or $\mathbf{x}'(t) = A(t)\mathbf{n}(t)$ et ∇f aligné avec \mathbf{n} au point $\mathbf{x}(t)$.
- D'où $\frac{df}{dt} = 0 = \frac{\partial f}{\partial t} + A|\nabla f|$ en tout point $\mathbf{x}(t)$.
- Equation valable sur tout le **contour** $C(t)$.

$$\frac{\partial f}{\partial t}_{|\mathbf{x}(t)} = -A(t)|\nabla f_{|\mathbf{x}(t)}|$$

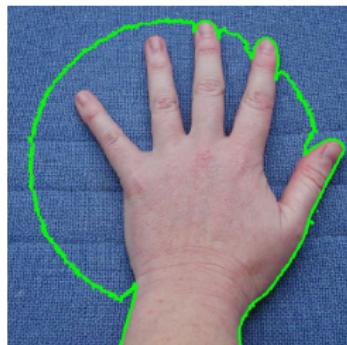
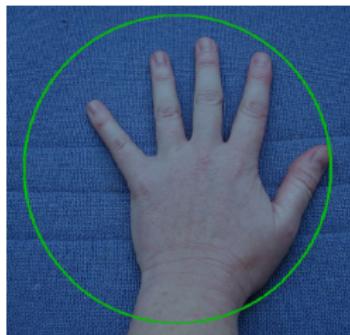
Résolution

- A supposé connu sur tout le contour (cf. après).
- Résolution par **différences finies**
 - grille régulière, de nœuds ij séparés par une distance h .
 - f_{ij}^n approche la solution $f(n\Delta t, ih, jh)$, Δt pas de temps.
 - On peut écrire : $\frac{f_{ij}^{n+1} - f_{ij}^n}{\Delta t} = -\hat{A}_{ij} \left| \nabla_{ij} f_{ij}^n \right|$.
 - ∇_{ij} : opérateur gradient discret
- La fonction de distance f doit être fréquemment réinitialisée (extraction de la position de la courbe et recalcul des distances).

Vitesse du front

- $A(\mathbf{x}) = -g(\mathbf{x})(A_0 + A_1 \kappa(\mathbf{x}))$
 - A_0 : inflation constante (pour propager le front dans le milieu)
 - $A_1 \kappa(\mathbf{x})$: élimine les lieux de forte courbure.
On a : $\kappa(\mathbf{x}) = \operatorname{div} \frac{\nabla f(\mathbf{x})}{|\nabla f(\mathbf{x})|} = \frac{f_{xx}f_y^2 - 2f_xf_yf_{xy} + f_{yy}f_x^2}{(f_x^2 + f_y^2)^{\frac{3}{2}}}$
 - $g(\mathbf{x})$: frein basé sur les contours image.
Exemples : $g(\mathbf{x}) = \frac{1}{1+|\nabla I|^2}$, $g(\mathbf{x}) = \frac{1}{1+|\nabla G_\sigma * I|^2}$

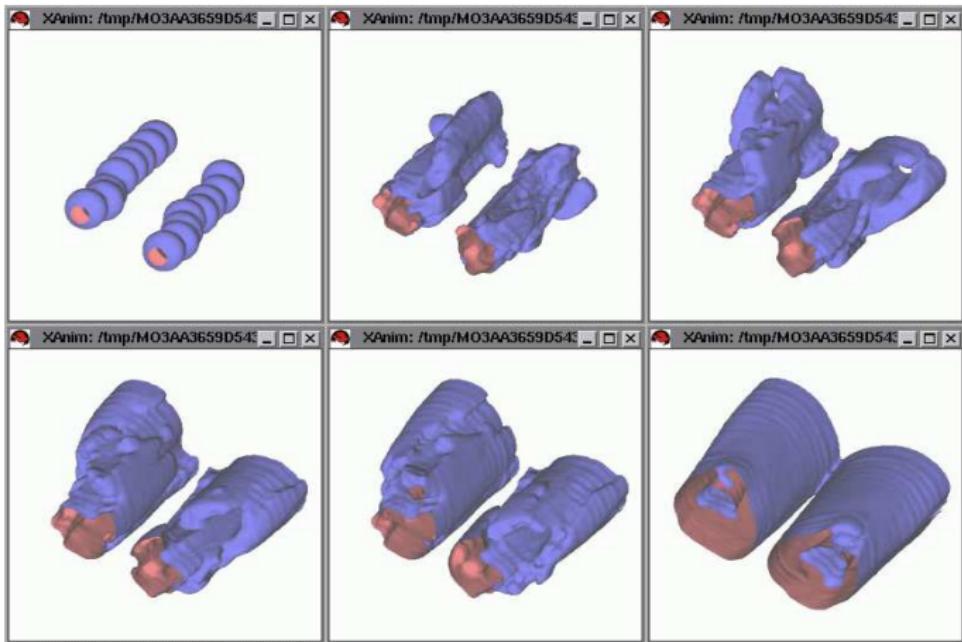
Exemple : segmentation 2D



images de D Lingrand (<http://www.polytech.unice.fr/lingrand>)

Exemple : segmentation 3D

- Segmentation des cuisses sur IRM (Malladi, Sethian)



Synthèse des approches level-set

- Modèle représenté implicitement
- Changements de topologie naturels
- Lien avec les modèles déformables classiques
 - lissage + recherche contour
 - cf [Whitaker 94] qui déforme l'espace avec une minimisation d'énergie sur l'interface.
- intéressant pour la 3D.
- de nombreuses variantes.

Problèmes et limites

- Résolution numérique coûteuse (distance de tout l'espace à l'interface)
 - ⇒ fenêtre de résolution
 - ⇒ ou approximation polygonale
- Inflation constante ou déflation constante
- Arrêt du modèle non déterminé
- Difficile de rajouter de nouvelles contraintes (e.g., interaction utilisateur, autres attracteurs)

5 - Autres approches de la segmentation

Modèles déformables

Méthodes level-set

Formulations énergétiques de la segmentation

Segmentation par minimisation

Segmentation = minimisation d'une énergie

On cherche (R_i) , partition de X , qui minimise l'énergie

$$E(\cup R_i) = \sum_i \underbrace{E_{\text{intra}}(R_i)}_{\searrow \text{ avec homogénéité}} + \sum_{i,j / R_i \text{ adj.}} \underbrace{E_{\text{inter}}(R_i, R_j)}_{\searrow \text{ avec hétérogénéité}} \quad (1)$$

- Problème de la segmentation comme maximisant homogénéité intra et hétérogénéité inter-régions est **mal posé**.
- ⇒ **Régularisation** du problème par addition d'autres contraintes (cf. *snakes*) : longueur des contours entre régions, courbures le long des contours, etc.

Modèle de Mumford-Shah [89] (1/2)

- **Idée** : approcher l'image I par une fonction u "lisse".

$$E(u, \Gamma) = \underbrace{\mu^2 \iint_R (u - I)^2}_{\text{attache aux données}} + \underbrace{\iint_{R-\Gamma} \|\nabla u\|^2 + \nu |\Gamma|}_{\text{régularisation}}. \quad (2)$$

- R : régions, Γ : contours entre régions.
- μ, ν règlent l'influence des termes.
- Simplification : u est constante par morceaux.

$$E(u, \Gamma) = \sum_i \mu^2 \sum_{R_i} (\underbrace{u}_{= \mu_i} - I)^2 + \nu |\Gamma| \quad (3)$$

Modèle de Mumford-Shah [89] (2/2)

- Difficile de trouver le meilleur contour Γ .
- Beaucoup d'approches heuristiques. Exemples :
 - par division-fusion dans des quadtrees [*\[Ackahmiezan93\]*](#)
 - par méthode level-set [*\[Chan01\]*](#)

Segmentation dans un graphe

- Domaine de l'image vu comme un graphe (V, A) , en général la grille
- **segmentation** = étiquetage des sommets en **minimisant** un critère local E
- E = somme d'énergies sur les sommets et sur les arêtes

$$E(\lambda) = \underbrace{\sum_{p \in V} U_p(\lambda_p)}_{\text{attache aux données}} + \underbrace{\sum_{\{p,q\} \in A} U_{p,q}(\lambda_p, \lambda_q)}_{\text{régularisation}}. \quad (4)$$

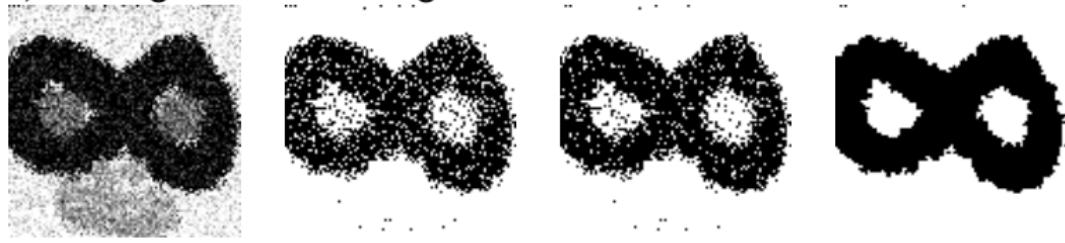
- avec λ étiquetage des sommets de V .

Exemples d'énergies I

- Binarisation d'images : 2 étiquettes (0 et 255) et

$$\begin{aligned}U_p(\lambda_p) &= |I(p) - \lambda_p| \\U_{p,q}(\lambda_p, \lambda_q) &= -\beta \text{ si } \lambda_p = \lambda_q \\&= \beta \text{ sinon}\end{aligned}$$

$\beta \nearrow$ = régularisation augmente



$\beta = 0$

$\beta = 5$

$\beta = 50$

Exemples d'énergies II

- Découpage en k classes connues (μ_i, σ_i)

$$U_p(\lambda_p) = \frac{(I(p) - \mu_i)^2}{2\sigma_i^2}$$

$$\begin{aligned} U_{p,q}(\lambda_p, \lambda_q) &= -\beta \text{ si } \lambda_p = \lambda_q \\ &= \beta \text{ sinon} \end{aligned}$$

Exemple : 3 classes, $(\mu_i, \sigma_i) = ((0, 20), (160, 8), (250, 10))$



$$\beta = 0$$



$$\beta = 3$$



$$\beta = 30$$

Application

- Domaine d'utilisation vaste : binarisation, segmentation, restauration, suivi vidéo, etc.
- Générique : graphes non limités aux grilles d'adjacence de pixels, valables pour des graphes quelconques
- Souple : les énergies traduisent de manière intuitive un problème à résoudre
- **Mais** : cette générericité rend le réglage des énergies et paramètres très difficile pour une application donnée

Optimisation de l'énergie

- Difficile de trouver le minimum de $E(\lambda)$
- **Heuristiques** : par exemple coupes dans les graphes
[Boykov01] (graph cut)
- **Approches stochastiques** et lien avec les champs de Markov : algorithmes qui extraient le minimum global (théoriquement)