

Segmentation et détection d'indices

Marie-Odile Berger, INRIA Nancy Grand Est

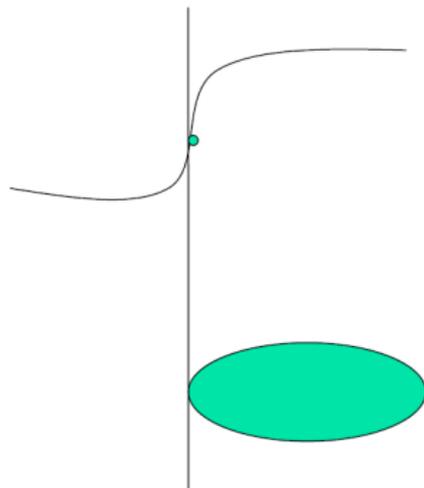
Objectif fréquent du traitement d'images: détecter/identifier des courbes, des régions ou des objets dans une image

- Les méthodes sans a priori et leurs limites
- Imposer des contraintes de **régularité** dans la détection
- **Apprendre** les propriétés des objets pour les détecter

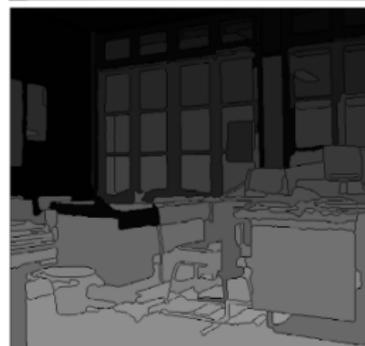
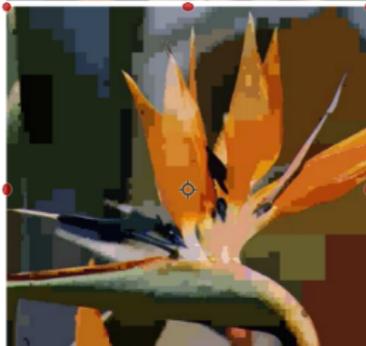
Part I

Les méthodes sans a priori

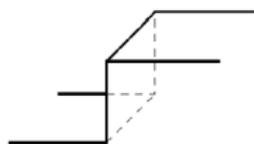
- Contour
 - limite d'un objet
 - variation brutale de l'intensité
- Région: zone d'intensité homogène
- plus récemment: les points d'intérêt
- encore plus récemment: les méthodes par apprentissage



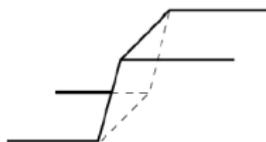
Exemples de bas niveau



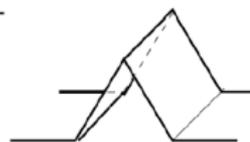
Modèles de contours



marche d'escalier



rampe



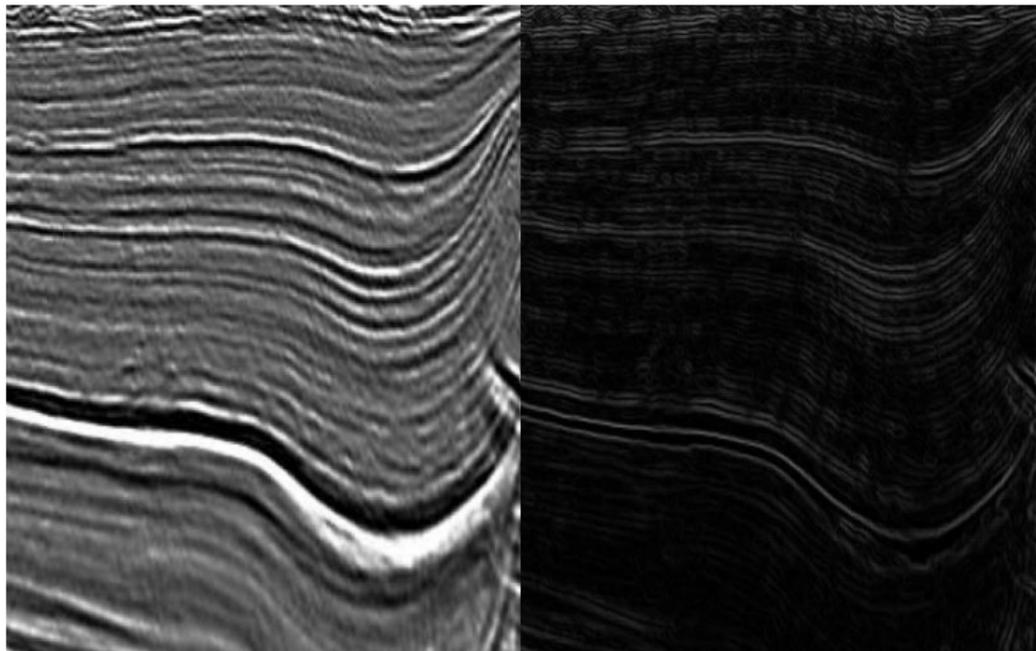
toit

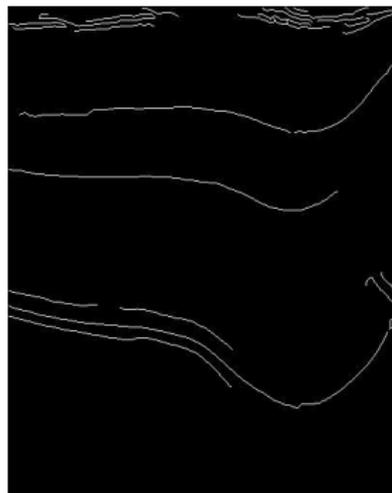
- La dérivation est un problème mal conditionné!
 - une petite perturbation de la fonction initiale cause des perturbations très importantes sur la dérivée
 - soit $f_2(t) = f_1(t) + \epsilon \cos(\omega t)$
 - alors $f_2'(t) = f_1'(t) - \epsilon \omega \sin(\omega t)$
 - $f_2'(t)$ et $f_1'(t)$ sont aussi proches que l'on veut. Pour ω grand, les dérivées peuvent être très éloignées.
- Détection de contour: **lisser** pour enlever le bruit puis **dérivé**.
- Dans le cas d'un filtre de convolution
 $(I * G_\sigma)' = I * G'_\sigma$ (convolution avec le filtre dérivé)

Historique de la segmentation

- Contour: changement brutal de l'intensité lumineuse.
- → extrema de la dérivée première
- mais l'image est numérisée: calcul approximatif de la dérivée $f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$ en prenant $h=1$...
- Le calcul de dérivée nécessite un pré-filtrage des images (souvent filtre gaussien) $G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} * I$; $G_y = \begin{bmatrix} -1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * I$
- **Canny** a proposé un détecteur plus élaboré pour la détection de contour
- Point de contour détecté si $|\nabla(I)|$ est localement maximal. Reste ensuite à chaîner ces points

Le module du gradient





`edge(i,'canny',.8)`



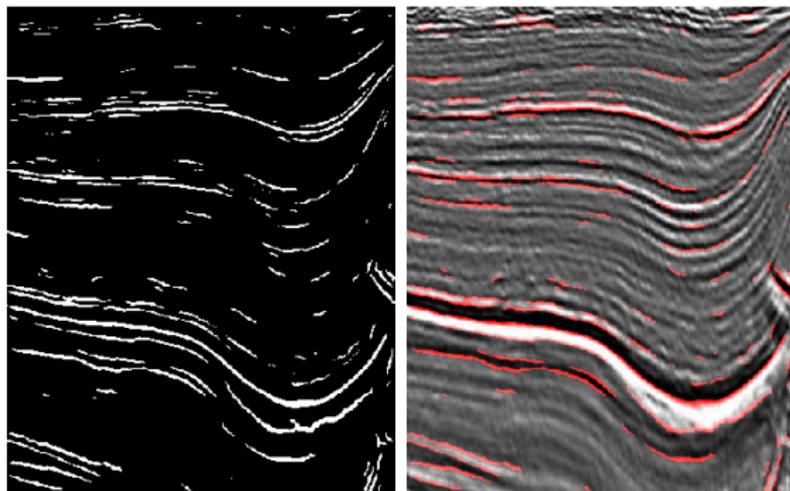
`edge(i,'canny',.5)`



`edge(i,'canny',.3)`

Constat: difficile de trouver un seuil global donnant des contours satisfaisants

Extraction de contours: avec apprentissage

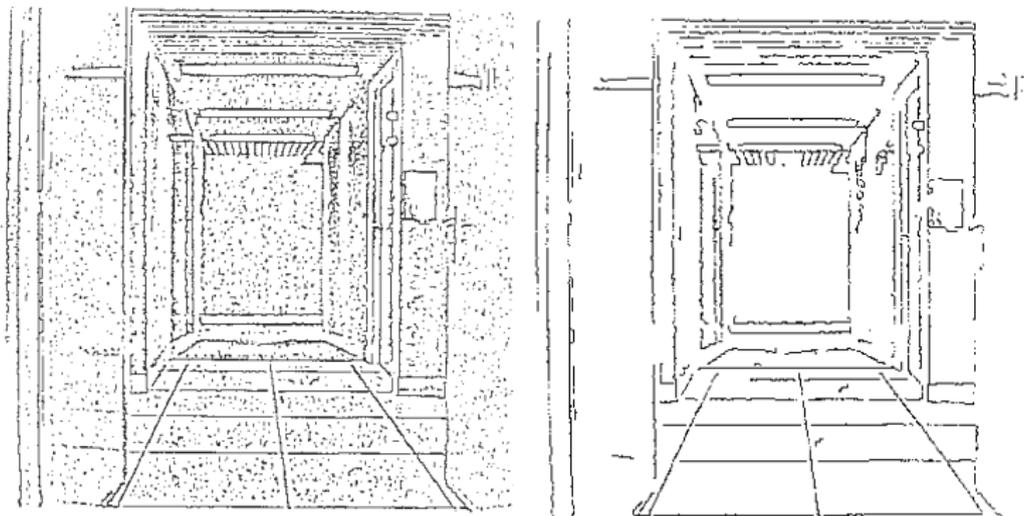


Méthode: fournir des exemples de segmentation en plus des images →
Moins de bruit dans les détections

Dollar and Zitnick, ICCV 2013: Structured Forests for Fast Edge
Detection.

L'étape éventuelle de chaînage

reste à chaîner et à éliminer les petites structures pour obtenir des courbes:



Difficulté: le contour n'est pas extrait en tant que courbe mais comme un ensemble de points isolés.

- Les méthodes par classification, utilisant la proximité photométrique des pixels
- Les méthodes par croissance de région
- La segmentation vue comme l'optimisation d'un critère: la fonctionnelle de Mumford et Shah

Segmentation utilisant seulement la photométrie

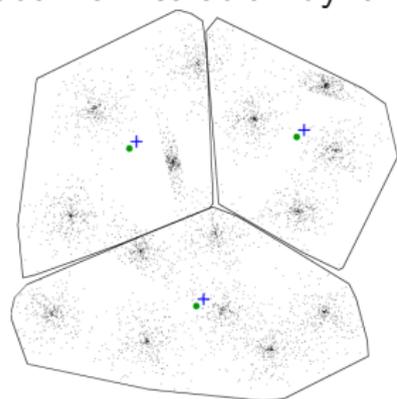
- idée de base: regrouper les pixels de photométrie voisine (modes de l'histogramme)
- Aucune contrainte spatiale: des pixels isolés peuvent faire partie de la même région.
- un exemple d'outils: la **classification par k-means** (ou k-moyennes): vise à partitionner les n observations $x_i (i < n)$ dans k ensembles afin de minimiser la distance intra cluster
- initialisation à $t = 0$: les moyennes des clusters μ_1, \dots, μ_k . Puis alterner les deux étapes
 - Assignment des cluster:

$$S_i(t) = \{x_j \text{ tel que } \|x_j - \mu_i\| \leq \|x_j - \mu_l\| \forall 1 \leq l \leq k\}$$

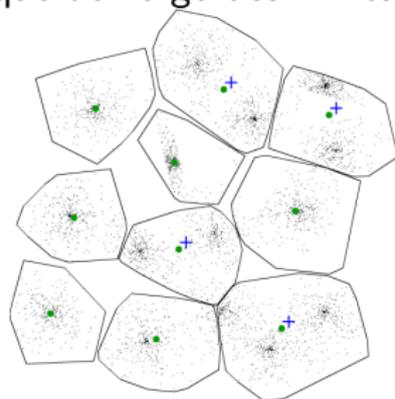
- mise à jour $\mu_i(t+1) = 1/\#S_i(t) \sum_{j \in S_i(t)} x_j$

Segmentation par k-means

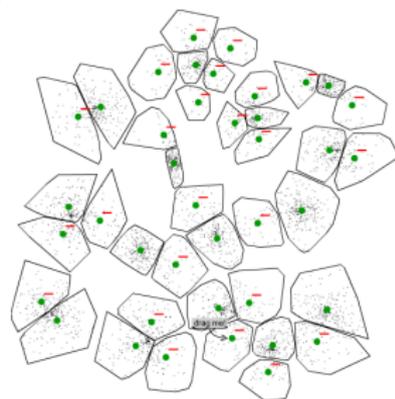
Voir le site https://en.wikipedia.org/wiki/K-means_clustering pour la visulisation dynamique de l'algo des k-means



3 classes



10 classes



40 classes

Segmentation par k-means

Inconvénients: le résultat dépend de l'initialisation et du choix de k

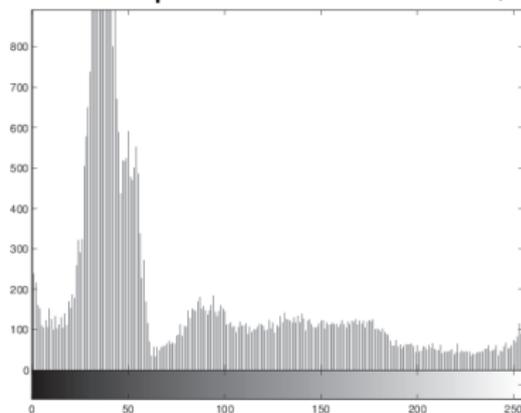


5 classes



9 classes

Centres pour 5 classes: 98.8, 7.6, 39.9, 158.4, 228.8



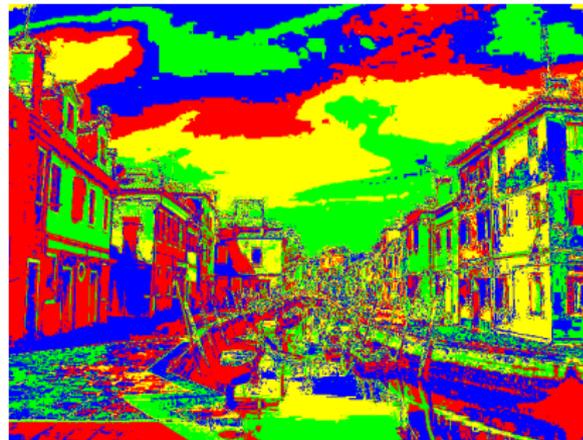
Segmentation d'une image couleur ou multispectrale

Une donnée est ici un triplet de valeurs (r,g,b).



5 classes

Segmentation d'une image couleur ou multispectrale



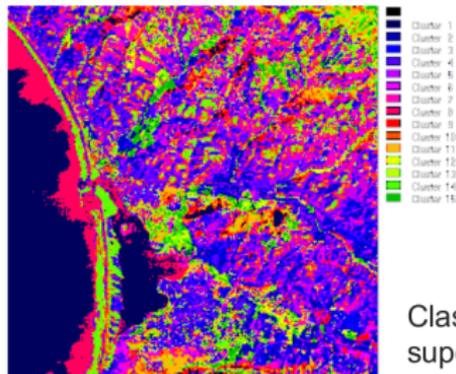
15classes

Segmentation/Classification d'images hyperspectrales



Morro Bay, Californie

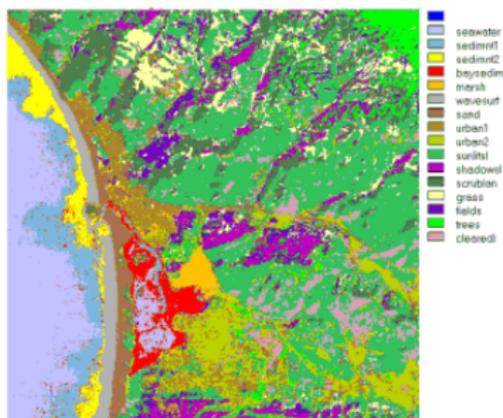
[tiré de Dinnet-Tremeau]



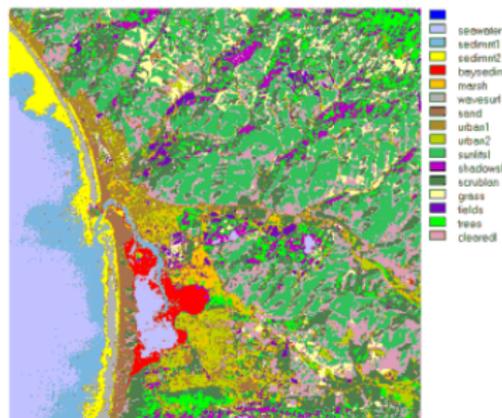
Classification non
supervisée (15 classes)

Classification supervisée d'images hyperspectrales

En apportant des informations ou des exemples sur les classes, on arrive à de meilleurs résultats... mais ils dépendent de la méthode:



Classification supervisée
(16 classes) par maximum
de vraisemblance

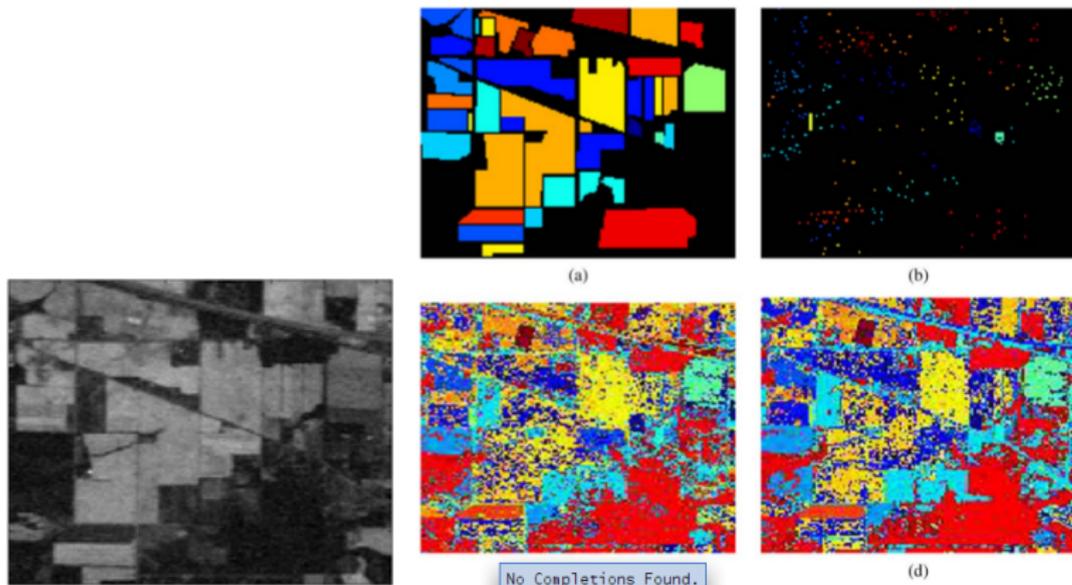


Classification supervisée
(16 classes) par
minimisation de la distance

[tiré de Dinnet-Tremeau]

Classification supervisée d'images hyperspectrales

image 224 bandes: [A. Villa: Hyperspectral Image Classification With Independent Component Discriminant Analysis .IEEE Trans on geosciences and remote sensing]



(a) vérité terrain (b) données d'apprentissage (c) classif avec SVM (d) ICDA.

TP: segmentation d'une image avec des Kmeans

but: montrer l'influence des paramètres (nombre de classes, type de distances) et de la représentation des couleurs choisies (tp_seg).



- si vous tapez “Color-Based Segmentation” dans l’aide de matlab, vous saurez comment faire une segmentation avec des kmeans en utilisant l’espace de couleur Lab
- Utilisez cette doc pour faire une segmentation de l’image spot.jpg et tester le résultat avec différentes valeurs de K et différents types de distances (sqEuclidean, cityblock, cosine)
- Effectuer maintenant directement la segmentation sur les images rgb. Que constatez vous?
- voir
(../Doc/representationCouleur.pdf, p25)

Abandonner l'idée du bas niveau parfait!

Constat: Les méthodes de bas niveau standard ne sont souvent pas suffisantes pour détecter des objets complexes en présence de bruit.

- **utiliser des connaissances** géométriques pour détecter des indices en accord avec le modèle (souvent paramétrique; Hough) ou estimer le modèle (statistique robuste)
- introduire des **a priori** sur ce qu'on recherche, souvent de régularité. Ex contours actifs et modèle de Chan et Vese
- **apprendre** des éléments permettant de particulariser le détecteur. Ex PCA sur la forme, sur la phométrie...

- A partir des années 1990, volonté d'avoir des processus d'extraction
 - moins ad hoc!
 - respectant des propriétés d'invariance (pas de délocalisation par filtrage)
 - considérant les contours en tant que courbes (et non comme un ensemble de points)
- Les méthodes:
 - **filtrage anisotropique**: lisser l'image en préservant les structures
 - formuler les problèmes en terme d'**optimisation**: l'indice cherchée est un objet complexe (courbe, surface) minimisant une fonctionnelle traduisant l'adéquation à l'image et les propriétés **souhaitables** de cet objet.
 - les **équations aux dérivées partielles** permettent de résoudre ces problèmes d'optimisation.

Part II

Extraction de formes paramétrées dans un cadre bruité

Souvent, le modèle des formes recherché est connu analytique (cercle, droite,...) → utiliser cette connaissance pour identifier les points appartenant à la forme des points de bruit

- La transformée de Hough
- les approches de type RANSAC (RANdom SAmple Consensus)

Les algorithmes de type *Random Sample Consensus* (RANSAC)[Fischler80]

Méthode: plutôt que d'utiliser tous les points disponibles et tenter d'éliminer les mauvais après, utiliser un petit nombre de points pour calculer une solution initiale et déterminer l'ensemble des points compatibles avec cette estimation dit ensemble de **consensus**.

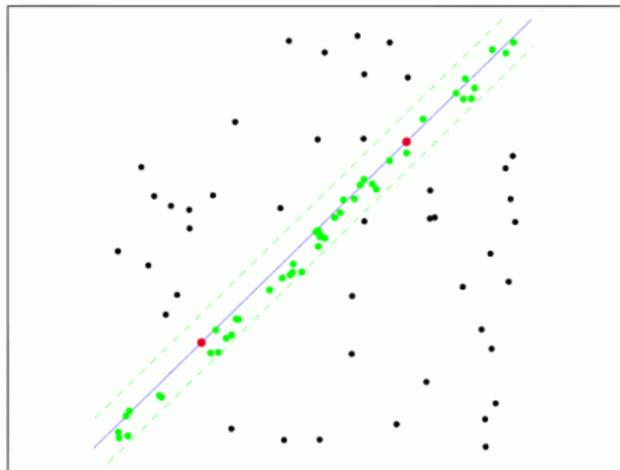
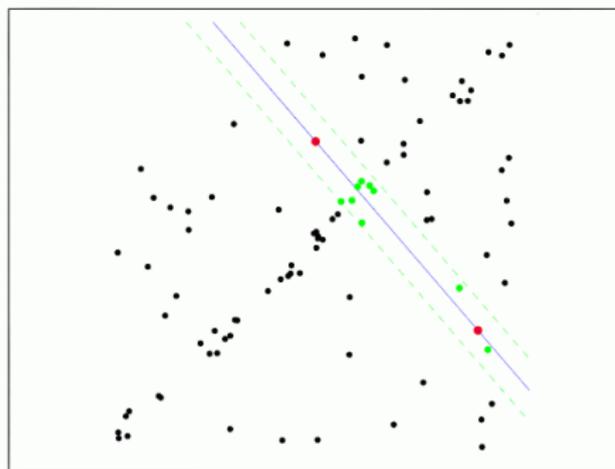
Algorithme RANSAC

on considère un ensemble de données D . Soit q le nombre minimal de données permettant d'estimer le paramètre.

- Générer des hypothèse plausibles de formes (pas toutes!)
- Déterminer les mesures compatibles avec cette hypothèse, c'est à dire les mesures situées à une distance inférieure à ϵ de l'hypothèse

RANSAC dans le cas de droites

Hypothèse plausible: toute droite définie par deux points présents dans l'ensemble de mesures



sources: R. Raguram
Ensemble de consensus en vert.

Algorithme RANSAC

Soit un ensemble de données D .

Soit q le nombre minimal de données permettant d'estimer le paramètre du modèle cherché (2 pour une droite, 3 pour un cercle,...)

Itérer le processus:

- choisir aléatoirement un ensemble S_1 de q points dans D et calculer le paramètre p .
- Utiliser p pour déterminer le sous ensemble de point S_1^* compatible avec p à une tolérance près. (S_1^* est l'ensemble de *Consensus*).

Fin lorsque: un nombre donné de tirages a été effectué ou lorsqu' une taille minimale de l'ensemble de consensus a été atteinte.

Détection de lignes avec RANSAC I

Matlab contient une fonction `ransac`. Je vous propose d'étudier son fonctionnement sur la détection de droites. (il y a bien d'autres exemples dans Matlab mais c'est un peu plus gourmand en temps de calcul).

- 1 le profil de la fonction `ransac` est le suivant:

```
[model,inlierIdx] =
```

```
ransac(data,fitFcn,distFcn,sampleSize,maxDistance)
```

A quoi correspondent en pratique les 2 fonctions `fitFcn` et `distFcn`, et les deux paramètres `sampleSize` et `maxDistance` présentes dans `RANSAC`.

- 2 Regardez l'exemple développé dans matlab sur le fitting de droites (quand vous tapez `ransac` dans la fenêtre `help`). Pourquoi une étape de re-estimation de chaque droite à partir des inliers est-elle nécessaire?

- 3 Vous travaillerez avec deux ensembles de données `pointsMoyenBruit.mat` et `pointsGrosBruit.mat` (cf figure 1). Ces données ont été bruitées à partir de points appartenant aux droites $y = 2x$, $y = -x$ et $y = -.5x + 2.5$ avec $x \in [-5, 5]$. Mettez en place la procédure `ransac` sur nos deux exemples de manière à récupérer les trois droites.

Détection de lignes avec RANSAC III

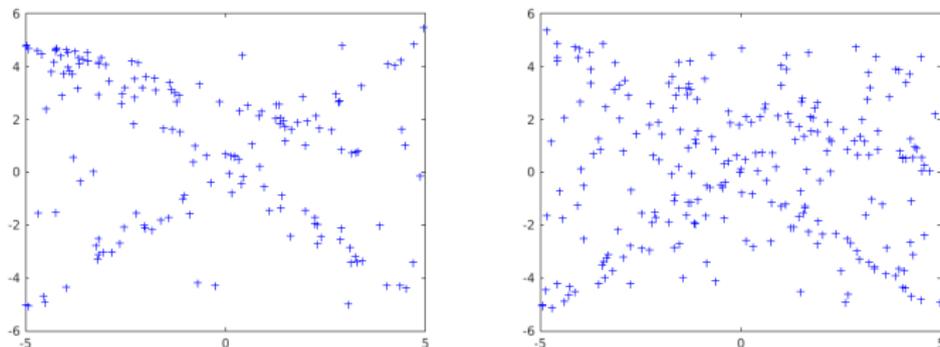


Figure: Ensemble de points noyés dans un bruit *moyen* (gauche) et *fort* (droite)

Détection de coniques [Zhang97]

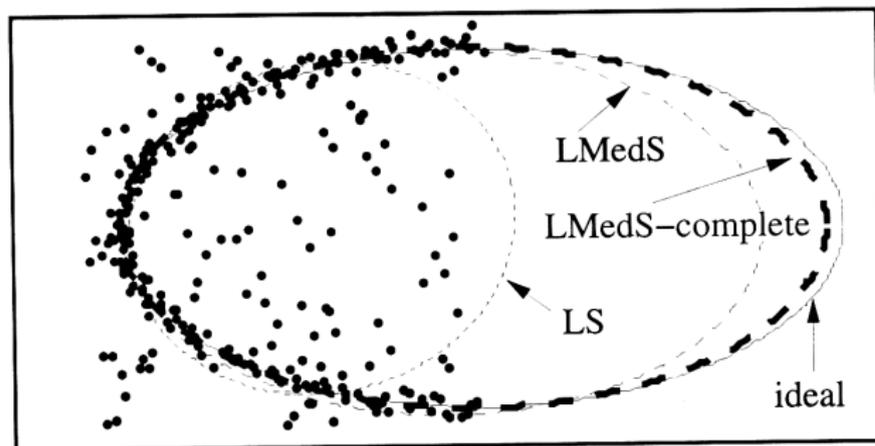
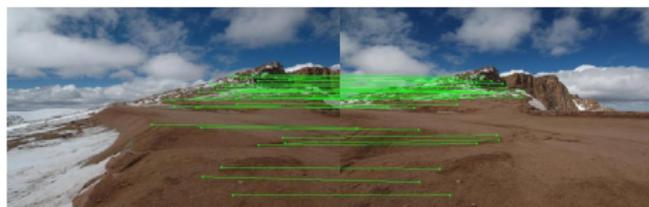
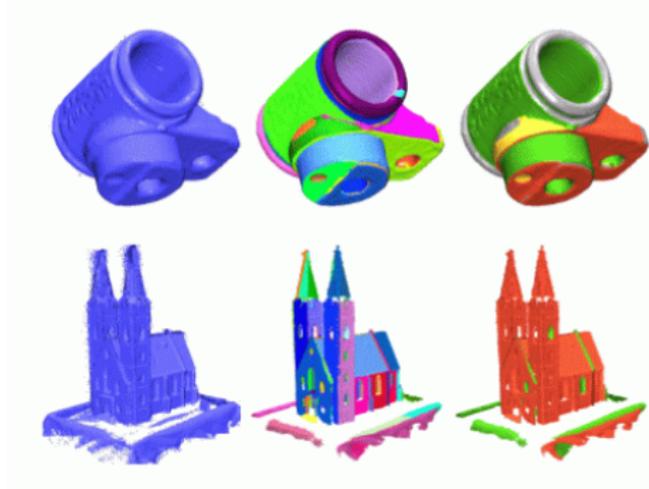


Fig. 7: Comparison of different techniques in the presence of outliers in data

Applications de RANSAC

D'innombrables applications depuis sa création



Segmentation de nuages de point [Schnabel07]

Calcul de la transformation entre deux images (construction panoramique)

Part III

Les méthodes basées sur la régularisation

- De nombreux problèmes de segmentation sont formulés comme des problèmes d'optimisation d'une énergie
- Avec un (des) termes d'adéquation à l'image et un terme assurant la régularité de la solution
- Exemple remarquable:
 - Contours actifs [Kass, Witkin et Terzopoulos]: un contour est une courbe suffisamment régulière dont les points ont en moyenne un gradient fort.

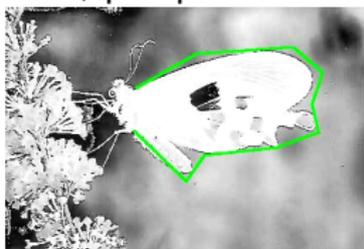
$$\min_{v(s)} \underbrace{\int \alpha |v'(s)|^2 + \beta |v''(s)|^2}_{\text{regularite}} - \underbrace{|\nabla I(v(s))|^2}_{\text{Contour}} ds$$

$$\int \alpha |v'(s)|^2 + \beta |v''(s)|^2 - |\nabla I(v(s))|^2 ds$$

- Pas de solution explicite: v est solution de l'EDP

$$\alpha v''(s) + \beta v''''(s) - \frac{\partial}{\partial v} |\nabla I(v(s))| = 0$$

- Solution itérative à partir d'une courbe initiale: la solution initiale doit être assez proche de l'objet à segmenter
- Il est impossible de changer de topologie: on ne peut détecter qu'une seule courbe, pas plusieurs.



- Le type de l'EDP à résoudre dépend fortement de l'énergie utilisée... (forme de l'énergie et non convexité, calcul de la dérivée difficile, choix et convergence du schéma numérique)



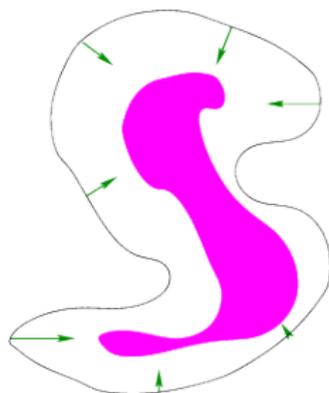
- Est-il possible d'envisager d'autres énergies, véhiculant la même idée (régularité des frontières) qui soient plus faciles à optimiser? avec des schémas numériques bien établis?
- Peut-on autoriser les changements de topologie?

- De nouvelles énergies, de nouvelles EDP avec de meilleures propriétés: les snakes geodesiques [Caselles97]

$$\int g(|\nabla I(v(s))|) |v'(s)| ds$$

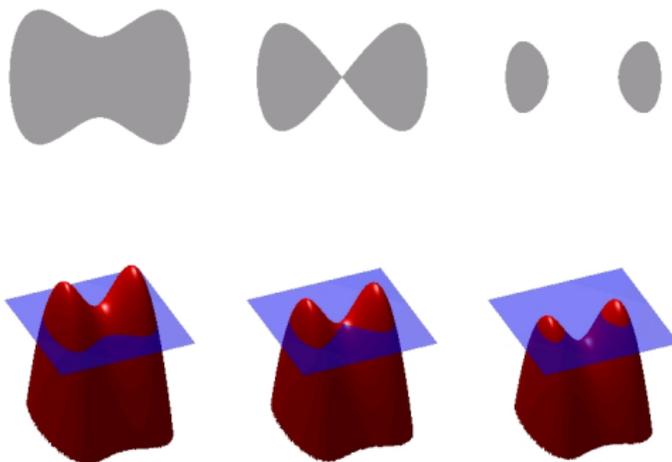
ou g est une fonction décroissante du gradient, comme celle de Perona Malik ($g = \frac{1}{1+|\nabla I|^2}$)

- Même comportement intuitif: minimum atteint pour des gradient forts et une bonne régularité.
- La courbe suit une équation d'évolution $\frac{\partial c}{\partial t} = \alpha N$ avec α dépendant de la courbure et du gradient.



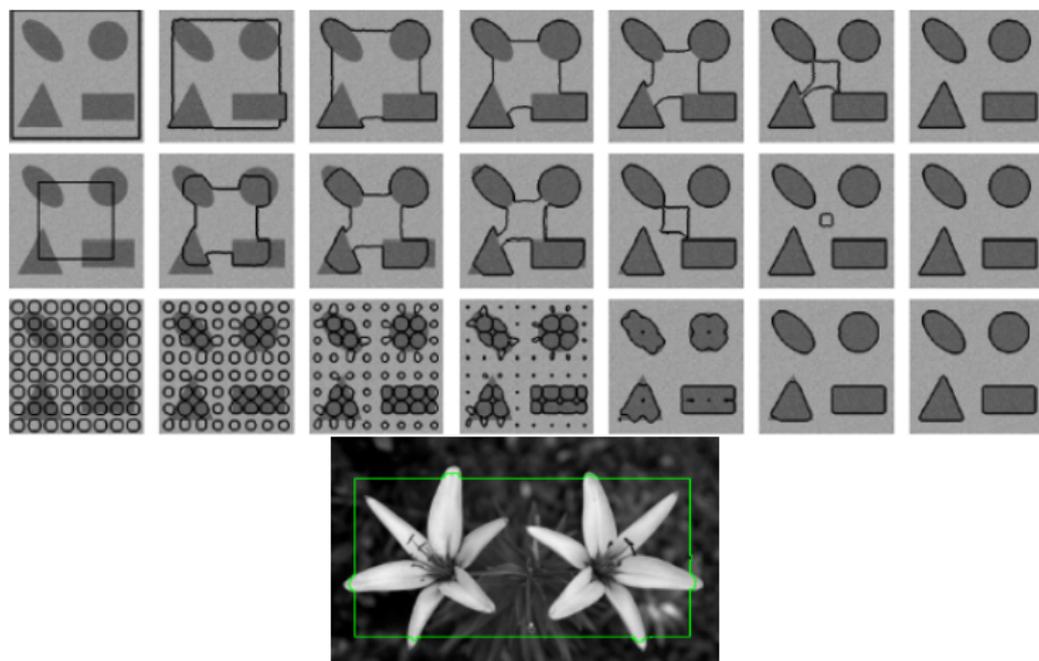
Level-sets et changement de topologie [Malladi95]

- Idée des level-sets: modéliser les objets comme courbe de niveau d'une surface



- l'équation d'évolution sur les droites est remplacée par l'équation d'évolution sur la surface.
- Les changements de topologie deviennent possible!

Levels sets: exemples



Innombrables améliorations

- en 3D [Gerig]

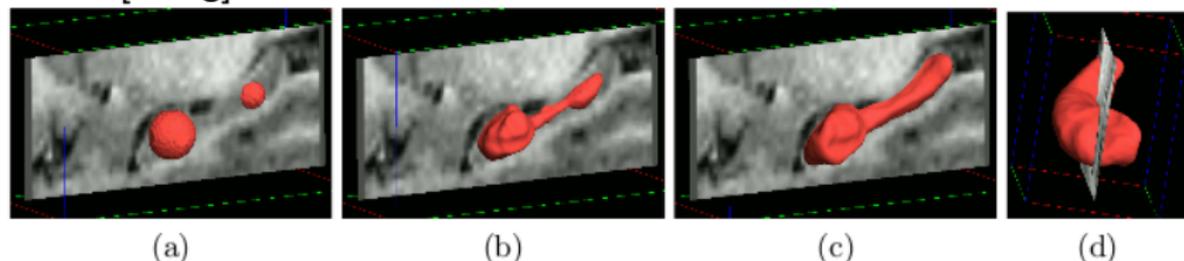


Fig. 1. Segmentation of left hippocampus from MRI: (a) initialization by two bubbles, (b) 6 iterations, (c) 18 iterations (final), and (d) final segmentation with rotated view ($c_{MCF} = 1$, $r_{MCF} = -1$, $r_{\nabla g} = 1$, $r_c = 2$, $\alpha = 1.4$, $r_s = 0$, and $c_s = 0.8$).

- Avec des indices autres que les contours: régions, texture
- Application à la reconstruction multi-vues...

[Chan et Vese 2001]: active contours without edges.

On suppose que l'image u_0 est formée de deux régions à détecter.; On formalise la détection de ces régions comme la recherche d'une courbe délimitante C et de deux niveaux c_1 et c_2 par le minimum de la fonction:

$$\mu \text{length}(C) + \nu \text{Area}_{\text{Inside}(C)} +$$

$$\lambda_1 \int_{\text{inside}(C)} (u_0(x, y) - c_1)^2 dx dy + \lambda_2 \int_{\text{outside}(C)} (u_0(x, y) - c_2)^2 dx dy$$

idée: détection en régions intégrant une régularisation via les termes d'aire et de périmètres

Rq: avec la programmation par level sets, on peut avoir **plus d'une composante connexe**.

TP: Test des méthodes de type contours actifs

Vous avez à disposition plusieurs méthodes de type contours actifs

- utiliser la fonction *activecontour* de matlab. *roipoly* permet de spécifier interactivement un contour initial.
- un démonstrateur de Chan et Vese est également accessible sur internet: [ici](#). Tester l'influence du terme de longueur μ
- un démonstrateur de contours actifs sur <http://www.ipol.im/pub/art/2012/abmh-rtmsa/>. Tester en particulier l'usage de la force *ballon*.
- Tester ces détecteurs sur diverses images (*cellule.jpg*)
- **Note:** le site ipol contient des démonstrations en ligne et des codes de très nombreuses méthodes de vision par ordinateur.