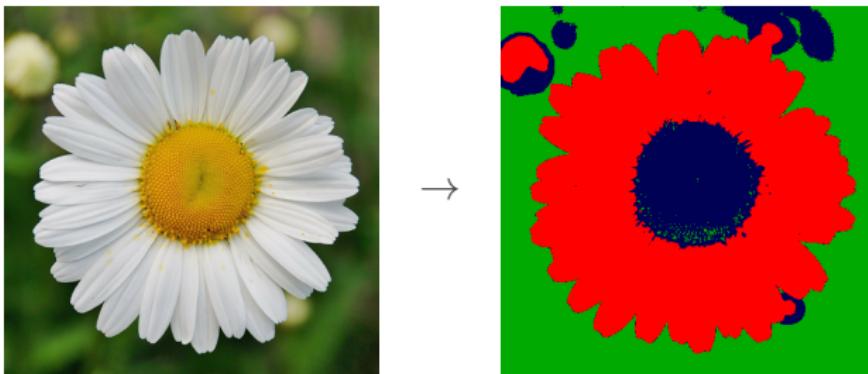


SEGMENTATION

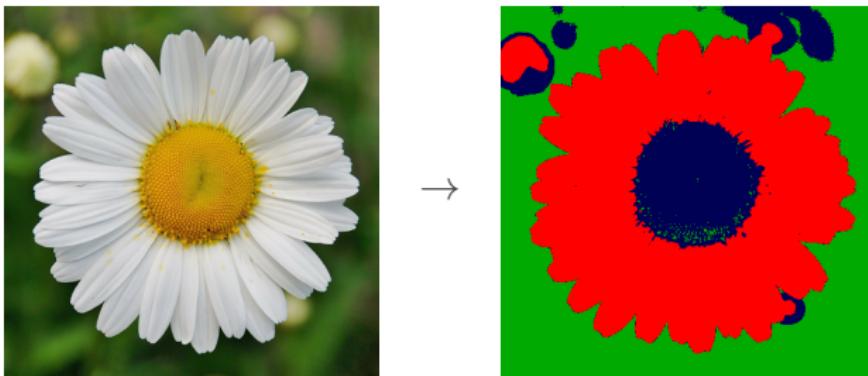
Définition

La segmentation (*segmentation*) consiste à partitionner une image f suivant un critère d'homogénéité.



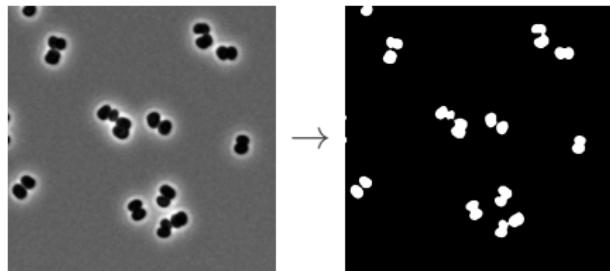
Définition

La segmentation (*segmentation*) consiste à partitionner une image f suivant un critère d'homogénéité.



- La partition est un ensemble de régions R_i disjointes qui recouvrent l'intégralité de l'image.
- Le critère d'homogénéité est vérifié par chaque région R_i et n'est pas vérifié pour l'union de deux régions adjacentes.

Exemples de méthodes de segmentation

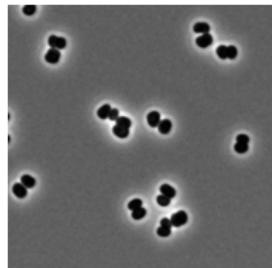


[NASA/JPL-Caltech]

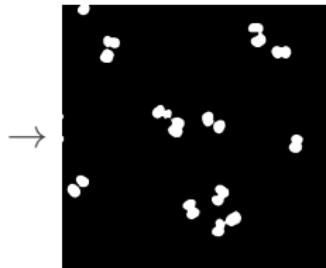
Critère :
niveaux de gris

Méthode :
seuillage de l'histogramme

Exemples de méthodes de segmentation

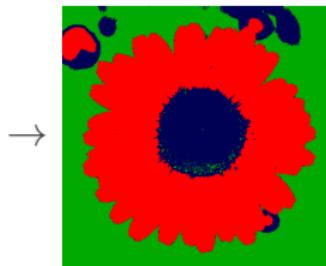


[NASA/JPL-Caltech]



Critère :
niveaux de gris

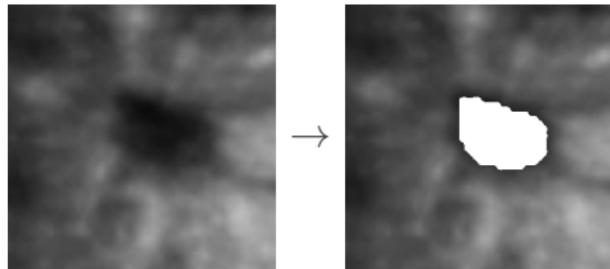
Méthode :
seuillage de l'histogramme



Critère :
couleur

Méthode :
classification

Exemples de méthodes de segmentation

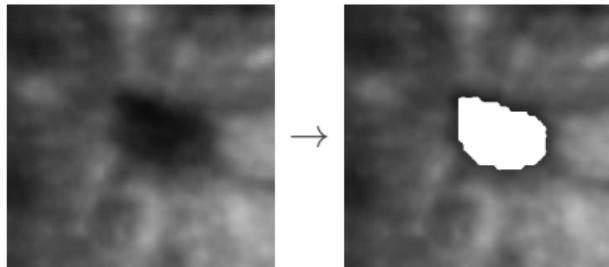


[Kozegar 2018]

Critère :
différence d'intensité

Méthode :
croissance de région

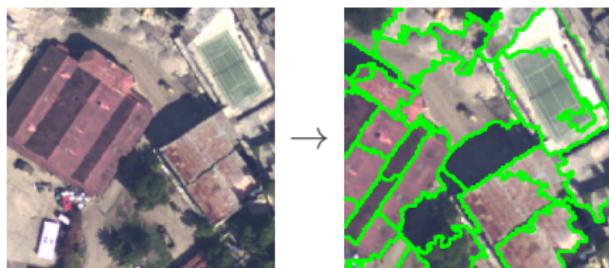
Exemples de méthodes de segmentation



[Kozegar 2018]

Critère :
différence d'intensité

Méthode :
croissance de région



[ICube/Sertit]

Critère :
couleur + taille des régions

Méthode :
SLIC [Achanta et coll. 2012]

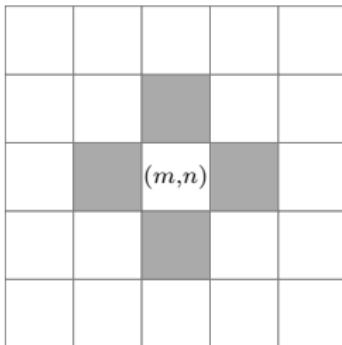
Sommaire

- Relations entre les pixels
- Quelques méthodes de segmentation :
 - seuillage de l'histogramme
 - classification des intensités de l'image
 - méthodes basées régions
 - méthodes basées modèles
 - ligne de partage des eaux
- Comment évaluer la performance d'une méthode de segmentation ?

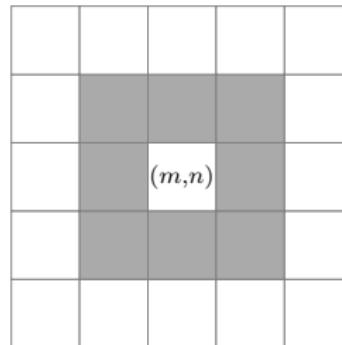
Relations entre les pixels

Voisinage

4-voisinage



8-voisinage



Les pixels en gris sont les voisins du pixel (m,n) .

Relations entre les pixels

Composante connexe

Une composante connexe (*connected component*) est un groupe de pixels tel qu'on puisse aller d'un pixel de ce groupe à un autre pixel de ce groupe en passant par des pixels du même groupe voisins entre eux.

Relations entre les pixels

Composante connexe

Une composante connexe (*connected component*) est un groupe de pixels tel qu'on puisse aller d'un pixel de ce groupe à un autre pixel de ce groupe en passant par des pixels du même groupe voisins entre eux.



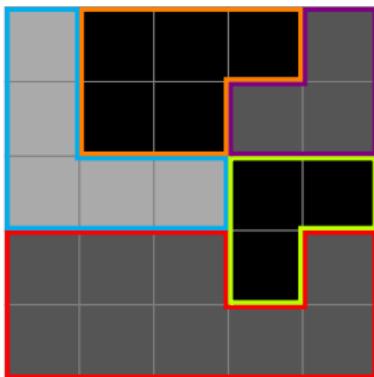
Combien de composantes connexes ?



Relations entre les pixels

Composante connexe

Une composante connexe (*connected component*) est un groupe de pixels tel qu'on puisse aller d'un pixel de ce groupe à un autre pixel de ce groupe en passant par des pixels du même groupe voisins entre eux.

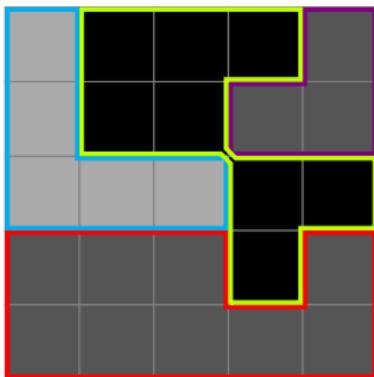


5 composantes connexes
en 4-voisinage.

Relations entre les pixels

Composante connexe

Une composante connexe (*connected component*) est un groupe de pixels tel qu'on puisse aller d'un pixel de ce groupe à un autre pixel de ce groupe en passant par des pixels du même groupe voisins entre eux.



4 composantes connexes
en 8-voisinage.

Remarques

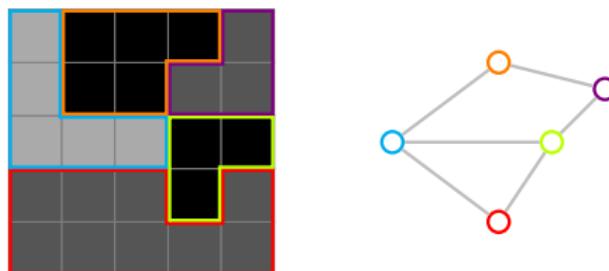
- Chaque région R_i est une composante connexe.

Remarques

- Chaque région R_i est une composante connexe.
- Le résultat de la segmentation n'est pas unique (dépend du critère d'homogénéité, de la méthode, de l'initialisation, etc.).

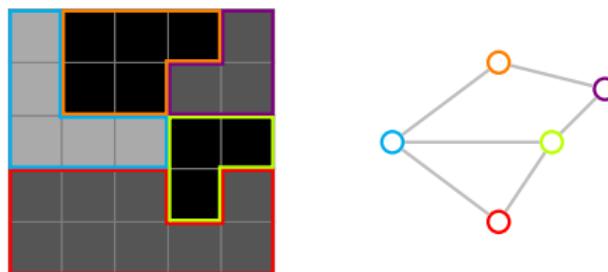
Remarques

- Chaque région R_i est une composante connexe.
- Le résultat de la segmentation n'est pas unique (dépend du critère d'homogénéité, de la méthode, de l'initialisation, etc.).
- Une segmentation peut s'interpréter comme un graphe (nœuds = régions, liens entre régions voisines)



Remarques

- Chaque région R_i est une composante connexe.
- Le résultat de la segmentation n'est pas unique (dépend du critère d'homogénéité, de la méthode, de l'initialisation, etc.).
- Une segmentation peut s'interpréter comme un graphe (nœuds = régions, liens entre régions voisines)

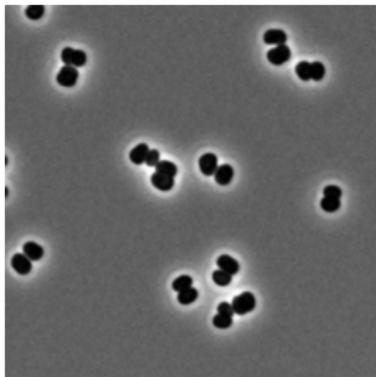


- La segmentation donne une représentation haut niveau de l'image.

Seuillage de l'histogramme

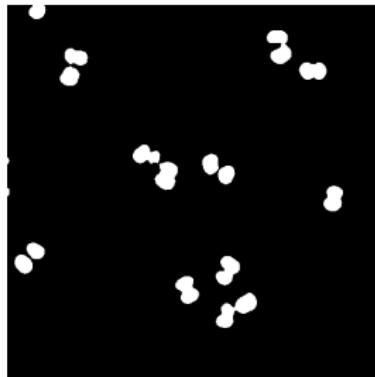
Seuillage

Distinction pixels clairs et foncés → binarisation de l'image



[NASA/JPL-Caltech]

f



g

$$g(m,n) = \begin{cases} 1 & \text{si } f(m,n) \geq T \\ 0 & \text{si } f(m,n) < T \end{cases} \quad \text{où } T \text{ est le seuil}$$

Comment choisir le seuil T ?

Seuillage



Image originale
(256 niveaux de gris
sur $\{0, \dots, 255\}$)



Seuil à 70

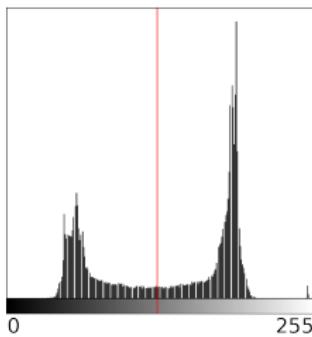


Seuil à 150

Un outil intéressant pour le choix du seuil : l'histogramme.

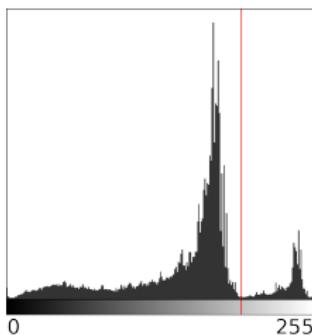
Seuillage

Dans certains cas, le choix du seuil est facile :



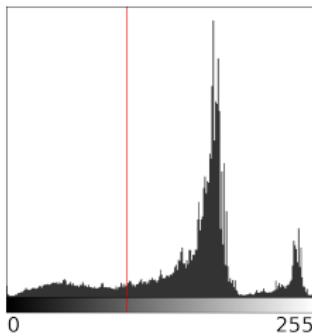
Seuillage

Dans d'autres cas, le choix du seuil est moins évident :



Seuillage

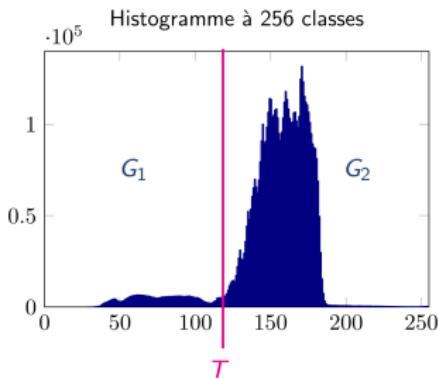
Dans d'autres cas, le choix du seuil est moins évident :



Méthode de Otsu

La méthode de Otsu [Otsu 1979] permet de déterminer le seuil qui minimise la *variance intra-classe* $\sigma_w^2(T)$ (moyenne pondérée des variances de chaque classe) :

$$\begin{aligned} T &= \arg \min_T \sigma_w^2(T) \\ &= \arg \min_T q_1(T)\sigma_1^2(T) + q_2(T)\sigma_2^2(T) \end{aligned}$$



Méthode de Otsu

$$T = \arg \min_T q_1(T) \sigma_1^2(T) + q_2(T) \sigma_2^2(T)$$

Pour la classe 1 :

Proportion :
$$q_1(T) = \sum_{i=0}^T p(i)$$

Moyenne :
$$m_1(T) = \frac{1}{q_1(T)} \sum_{i=0}^T i p(i)$$

Variance :
$$\sigma_1^2(T) = \frac{1}{q_1(T)} \sum_{i=0}^T (i - m_1(T))^2 p(i)$$

Les intensités i sont à valeurs dans $\{0, \dots, L-1\}$,
 $p(i) = h(i)/(M \times N)$ est l'histogramme normalisé
et h l'histogramme.

Méthode de Otsu

$$T = \arg \min_T q_1(T) \sigma_1^2(T) + q_2(T) \sigma_2^2(T)$$

Pour la classe 2 :

Proportion :

$$q_2(T) = \sum_{i=T+1}^{L-1} p(i)$$

Moyenne :

$$m_2(T) = \frac{1}{q_2(T)} \sum_{i=T+1}^{L-1} i p(i)$$

Variance :

$$\sigma_2^2(T) = \frac{1}{q_2(T)} \sum_{i=T+1}^{L-1} (i - m_2(T))^2 p(i)$$

Les intensités i sont à valeurs dans $\{0, \dots, L-1\}$,
 $p(i) = h(i)/(M \times N)$ est l'histogramme normalisé
et h l'histogramme.

Méthode de Otsu

Implémentation

Calculer pour tous les seuils $T = \{0, \dots, L - 1\}$ la variance intra-classe $\sigma_w^2(T)$, et retenir le seuil T qui minimise $\sigma_w^2(T)$.

Pour $T = 0 \rightarrow L - 1$:

| Calculer $\sigma_w^2(T)$

Choisir $\hat{T} = \arg \min_T \sigma_w^2(T)$

Méthode de Otsu

Remarque

La variance σ^2 des intensités de l'image s'écrit :

$$\sigma^2 = \sigma_w^2 + \sigma_{1,2}^2$$

où $\sigma_{1,2}^2$ est la *variance inter-classe* (variance pondérée des moyennes de chaque classe).

Méthode de Otsu

Remarque

La variance σ^2 des intensités de l'image s'écrit :

$$\sigma^2 = \sigma_w^2 + \sigma_{1,2}^2$$

où $\sigma_{1,2}^2$ est la *variance inter-classe* (variance pondérée des moyennes de chaque classe).

⇒ Minimiser la variance intra-classe σ_w^2 est équivalent à maximiser la variance inter classe $\sigma_{1,2}^2$ (puisque σ^2 reste constant).

Méthode de Otsu

Remarque

La variance σ^2 des intensités de l'image s'écrit :

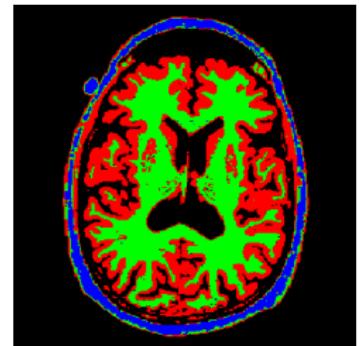
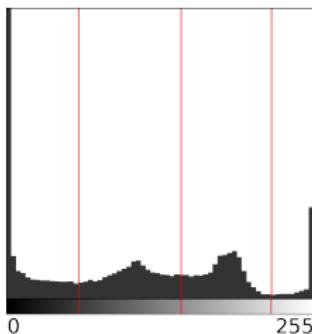
$$\sigma^2 = \sigma_w^2 + \sigma_{1,2}^2$$

où $\sigma_{1,2}^2$ est la *variance inter-classe* (variance pondérée des moyennes de chaque classe).

- ⇒ Minimiser la variance intra-classe σ_w^2 est équivalent à maximiser la variance inter classe $\sigma_{1,2}^2$ (puisque σ^2 reste constant).
- ⇒ Construire deux groupes de « pixels qui se ressemblent » revient à construire deux « groupes très dissemblables » de pixels.

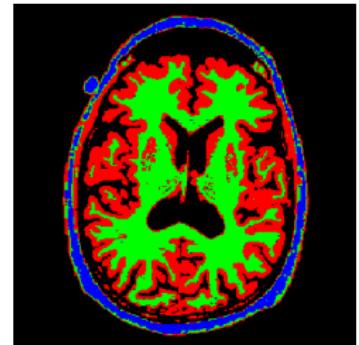
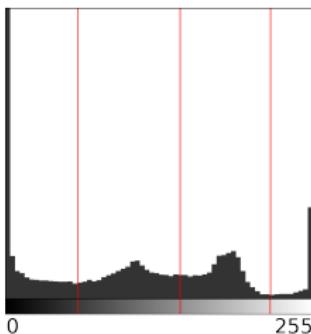
Seuillage multiple

Lorsque plusieurs modes sont visibles sur l'histogramme, il est possible d'utiliser plusieurs seuils pour aboutir à plusieurs classes :



Seuillage multiple

Lorsque plusieurs modes sont visibles sur l'histogramme, il est possible d'utiliser plusieurs seuils pour aboutir à plusieurs classes :



En particulier, la méthode de Otsu peut être étendue à plusieurs seuils, mais la complexité calculatoire augmente grandement avec le nombre de classes !

Méthode de Otsu



La méthode de Otsu nécessite le réglage d'un paramètre par l'utilisateur

Méthode de Otsu



La méthode de Otsu nécessite le réglage d'un paramètre par l'utilisateur → Faux

Méthode de Otsu



La méthode de Otsu nécessite le réglage d'un paramètre par l'utilisateur → Faux

La méthode de Otsu tend à avoir des intensités, au sein d'une classe, les plus proches possibles

Méthode de Otsu



La méthode de Otsu nécessite le réglage d'un paramètre par l'utilisateur → Faux

La méthode de Otsu tend à avoir des intensités, au sein d'une classe, les plus proches possibles → Vrai

Méthode de Otsu



La méthode de Otsu nécessite le réglage d'un paramètre par l'utilisateur → Faux

La méthode de Otsu tend à avoir des intensités, au sein d'une classe, les plus proches possibles → Vrai

La méthode de Otsu tend à avoir des intensités entre deux classes les plus proches possibles

Méthode de Otsu



La méthode de Otsu nécessite le réglage d'un paramètre par l'utilisateur → Faux

La méthode de Otsu tend à avoir des intensités, au sein d'une classe, les plus proches possibles → Vrai

La méthode de Otsu tend à avoir des intensités entre deux classes les plus proches possibles → Faux

Méthodes de classification

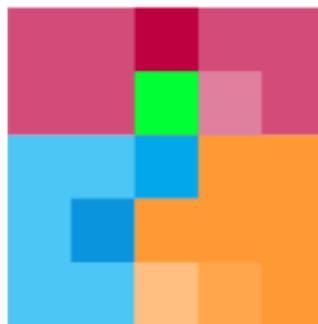
Méthodes de classification

Comment segmenter une image multibande ?

Méthodes de classification

Comment segmenter une image multibande ?

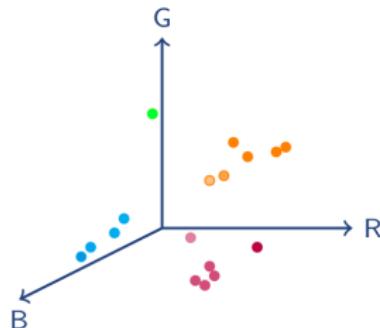
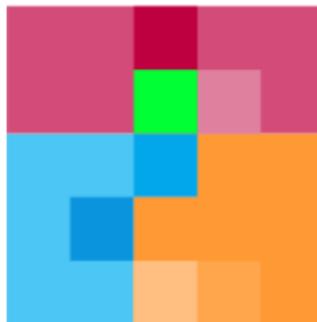
Par exemple, une image RVB possède trois bandes, pour chacune desquelles peut être déterminée un histogramme.



⇒ segmentation de l'image à l'aide d'un histogramme impossible!

Méthodes de classification

Un pixel est maintenant représenté par un vecteur à valeurs dans $\mathbb{R}^B \Rightarrow$ travailler directement dans l'espace \mathbb{R}^B !



Le principe des méthodes de classification (ou plus exactement de coalescence) (*clustering*) est de regrouper les pixels en groupes homogènes.

Méthodes de classification

- Algorithme des k-moyennes [Steinhaus 1957, MacQueen 1967]
- Modèles paramétriques (mélange de lois)
- *Mean-shift* [Fukunaga & Hostetler 1975]
- SLIC [Achanta et coll. 2012]
- ...

Algorithme des k-moyennes

L'algorithme des k-moyennes (*k-means*) est une méthode itérative qui affecte chaque point de l'espace \mathbb{R}^B à l'un des K groupes (*clusters*) (K choisi par l'utilisateur).

Initialisation aléatoire de K centroïdes

Répéter tant que les centroïdes varient :

 Pour chaque point :

 Calcul des distances du point à tous les centroïdes

 Affectation du point au groupe le plus proche

 Calcul du centroïde de chacun des groupes

Algorithme des k-moyennes

Exemple

Pour simplifier, on considère une image à deux bandes (\Rightarrow espace à deux dimensions) à segmenter en $K = 2$ classes (\Rightarrow deux couleurs).

Initialisation des centroïdes

Tant que les centroïdes varient :

 Pour chaque point :

 Distances aux centroïdes

 Affectation

 Mise à jour des centroïdes



Algorithme des k-moyennes

Exemple

Pour simplifier, on considère une image à deux bandes (\Rightarrow espace à deux dimensions) à segmenter en $K = 2$ classes (\Rightarrow deux couleurs).

Initialisation des centroïdes

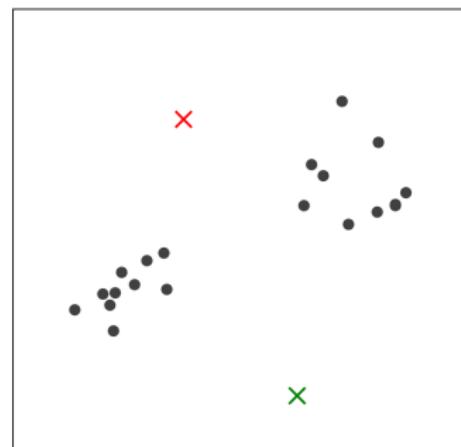
Tant que les centroïdes varient :

 Pour chaque point :

 Distances aux centroïdes

 Affectation

 Mise à jour des centroïdes



Algorithme des k-moyennes

Exemple

Pour simplifier, on considère une image à deux bandes (\Rightarrow espace à deux dimensions) à segmenter en $K = 2$ classes (\Rightarrow deux couleurs).

Initialisation des centroïdes

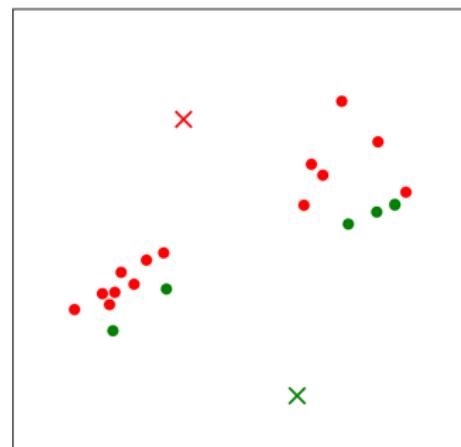
Tant que les centroïdes varient :

 Pour chaque point :

 Distances aux centroïdes

 Affectation

 Mise à jour des centroïdes



Algorithme des k-moyennes

Exemple

Pour simplifier, on considère une image à deux bandes (\Rightarrow espace à deux dimensions) à segmenter en $K = 2$ classes (\Rightarrow deux couleurs).

Initialisation des centroïdes

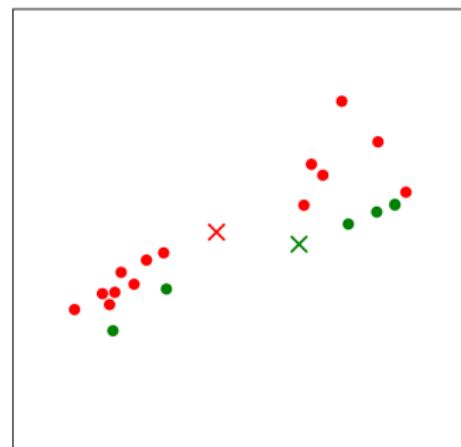
Tant que les centroïdes varient :

 Pour chaque point :

 Distances aux centroïdes

 Affectation

 Mise à jour des centroïdes



Algorithme des k-moyennes

Exemple

Pour simplifier, on considère une image à deux bandes (\Rightarrow espace à deux dimensions) à segmenter en $K = 2$ classes (\Rightarrow deux couleurs).

Initialisation des centroïdes

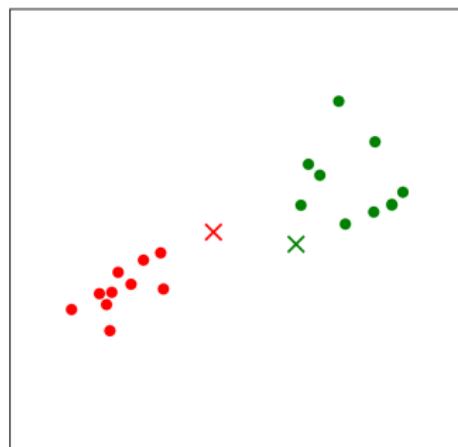
Tant que les centroïdes varient :

 Pour chaque point :

 Distances aux centroïdes

 Affectation

 Mise à jour des centroïdes



Algorithme des k-moyennes

Exemple

Pour simplifier, on considère une image à deux bandes (\Rightarrow espace à deux dimensions) à segmenter en $K = 2$ classes (\Rightarrow deux couleurs).

Initialisation des centroïdes

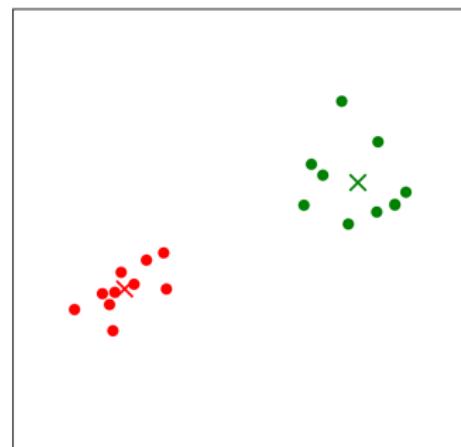
Tant que les centroïdes varient :

 Pour chaque point :

 Distances aux centroïdes

 Affectation

 Mise à jour des centroïdes



Algorithme des k-moyennes

Exemple

Pour simplifier, on considère une image à deux bandes (\Rightarrow espace à deux dimensions) à segmenter en $K = 2$ classes (\Rightarrow deux couleurs).

Initialisation des centroïdes

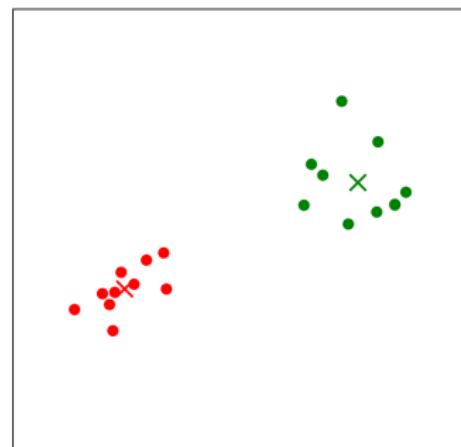
Tant que les centroïdes varient :

 Pour chaque point :

 Distances aux centroïdes

 Affectation

 Mise à jour des centroïdes



Algorithme des k-moyennes

Exemple

Pour simplifier, on considère une image à deux bandes (\Rightarrow espace à deux dimensions) à segmenter en $K = 2$ classes (\Rightarrow deux couleurs).

Initialisation des centroïdes

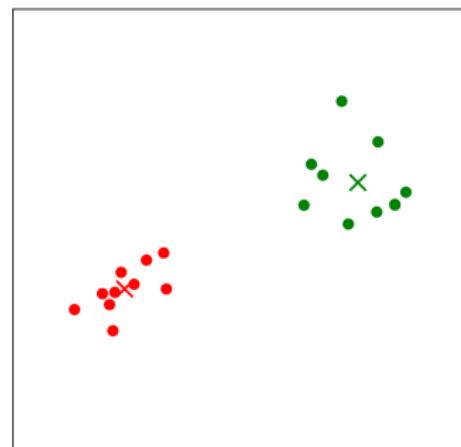
Tant que les centroïdes varient :

 Pour chaque point :

 Distances aux centroïdes

 Affectation

 Mise à jour des centroïdes



Algorithme des k-moyennes



Image originale



$K = 2$



$K = 4$

Algorithme des k-moyennes

Avantages

- Méthode simple
- Implémentation facile
- Méthode généralement rapide
- Classes de variance conditionnelle minimale
- Fonctionne correctement lorsque les clusters sont sphériques



Algorithme des k-moyennes

Inconvénients

- Nécessite de connaître le nombre de classes
- Sensible aux minima locaux, donc à l'initialisation
- Peut être lent en grande dimension
- Échoue pour des structures non sphériques

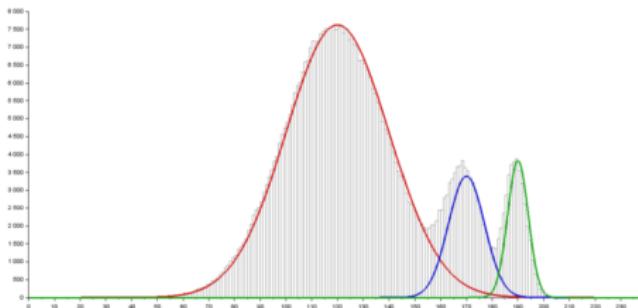
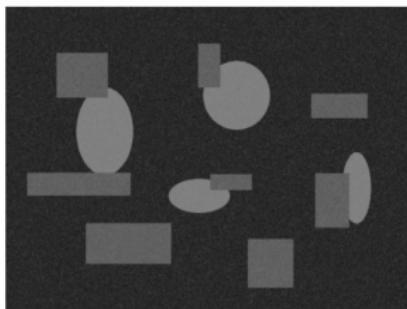


- Sensible aux valeurs aberrantes



Modèles paramétriques

L'histogramme de l'image est modélisé par un mélange de lois (*mixture model*) : on dispose d'un modèle paramétrique représentatif des classes présentes dans l'image.



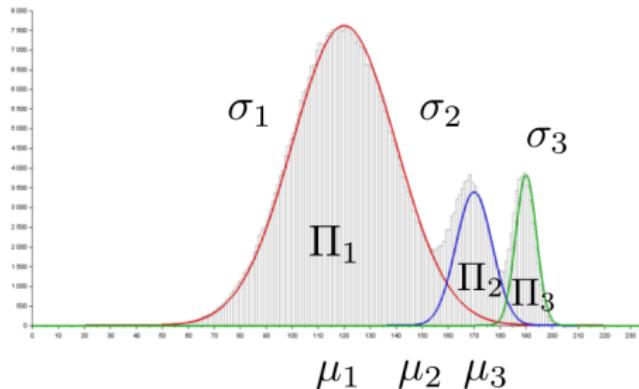
- Lois : souvent gaussiennes (*GMM : Gaussian mixture model*).
- Extension possible à plusieurs dimensions

Modèles paramétriques

Deux étapes :

1 Estimation des paramètres des lois

Poids Π_k , moyennes μ_k , écart-types σ_k



2 Classification

Associer à chaque intensité la classe la plus représentative

Modèles paramétriques

1 Estimation

$$\forall i, \quad p(h(i)|\theta) = \sum_{k=1}^K \frac{\Pi_k}{\sqrt{2\pi\sigma_k^2}} \exp\left(-\frac{(h(i) - \mu_k)^2}{2\sigma_k^2}\right)$$

où K est le nombre de classes et θ regroupe les paramètres inconnus des lois : $\theta = [\Pi_1, \dots, \Pi_K, \mu_1, \dots, \mu_K, \sigma_1, \dots, \sigma_K]$.

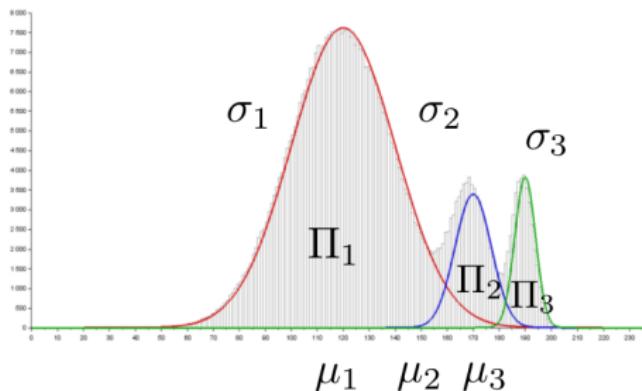
Estimation des paramètres au sens du maximum de vraisemblance :

$$\hat{\theta}^{\text{MV}} = \arg \max_{\theta} \prod_i p(h(i)|\theta)$$

Méthode de résolution : algorithme EM, algorithmes MCMC, ...

Modèles paramétriques

2 Classification



Chaque pixel est affecté à la classe dont il maximise la loi :

$$f_{\text{seg}}(m, n) = \arg \max_{k \in \{1, \dots, K\}} \frac{\Pi_k}{\sqrt{2\pi\sigma_k^2}} \exp \left(-\frac{(f(m, n) - \mu_k)^2}{2\sigma_k^2} \right)$$

Modèles paramétriques

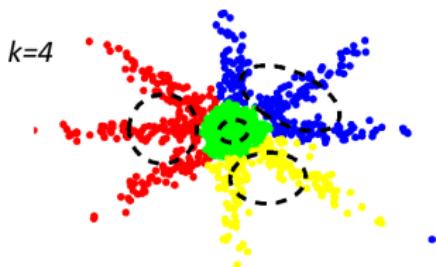
k-moyennes

Estimation uniquement des μ_k

Sensible à l'initialisation

Sensible aux minima locaux

Nécessite de connaître le nombre de classes



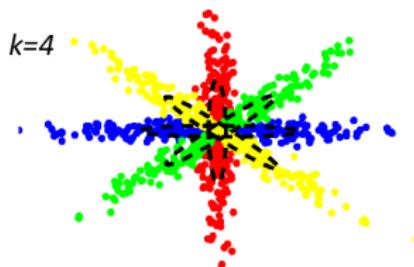
Mélange de gaussiennes

Estimation des μ_k et σ_k

Sensible à l'initialisation

Sensible aux minima locaux

Nécessite de connaître le nombre de classes



Méthodes basées région

Limitation des méthodes de seuillage

La limite fondamentale des méthodes de seuillage est de ne pas prendre en compte l'information de voisinage (seule l'information de distribution des intensités est utilisée).

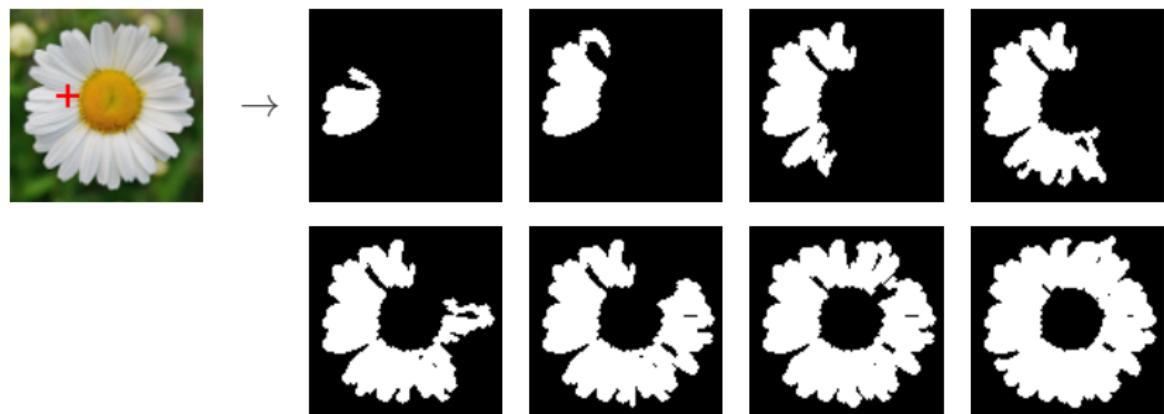
L'avantage des méthodes basées région est d'agréger des pixels spatialement proches et ayant des intensités similaires.

- Croissance de région
- Décomposition / fusion

Croissance de région

(Region growing)

Principe : on part d'un pixel initial (« germe ») et on l'étend en ajoutant les pixels du voisinage satisfaisant le critère d'homogénéité.



Croissance de région

Choix du germe :

- manuellement (dans la zone d'intérêt)
- automatiquement (en évitant les zones de fort contraste = fort gradient)

Croissance de région

Choix du germe :

- manuellement (dans la zone d'intérêt)
- automatiquement (en évitant les zones de fort contraste = fort gradient)

Critère de similarité : si un pixel $f(m,n)$ et une région R sont suffisamment similaires, alors ils sont fusionnés; sinon une nouvelle région est créée.

Exemple de critère :

$$|f(m,n) - \mu_R| < T\sigma_R$$

- T élevé : facile d'agréger des nouveaux pixels à la région.
- T faible : difficile d'agréger des nouveaux pixels à la région.

Choix de la connexité : 4-voisinage ou 8-voisinage.

Croissance de région

- R : région segmentée, initialisée au germe
- S : pixels à tester, initialisé au voisinage du germe (file FIFO : *first in, first out*)

Croissance de région

- R : région segmentée, initialisée au germe
- S : pixels à tester, initialisé au voisinage du germe (file FIFO : *first in, first out*)

tant que S n'est pas vide :

 | p est le premier pixel de la liste S

 | p est retiré de S

 | si p est homogène avec R :

 | ajout à R de p

 | ajout à S des pixels du voisinage de p qui ne sont pas dans
 R et qui ne sont pas incompatibles.

 | sinon :

 | p est marqué comme incompatible.

Croissance de région

La croissance de région ne fournit pas directement une partition de l'image, mais permet de segmenter une ou plusieurs structures d'intérêt via la sélection de germes adaptés.

Au moins deux points germes sont nécessaires :



Image originale



Image segmentée

Croissance de région

Quelle segmentation est obtenue avec la plus grande valeur de T ?



A



B



Croissance de région

Quelle segmentation est obtenue avec la plus grande valeur de T ?



T petit



T grand



Décomposition / fusion

(*Split and merge*)

Principe :

- 1 Divisions (*split*) successives de chaque région de l'image si elles ne satisfont pas le critère d'homogénéité
⇒ Permet d'aboutir à une « partition initiale ».

- 2 Fusions (*merge*) successives des régions adjacentes satisfaisant le critère d'homogénéité.

Décomposition / fusion

(*Split and merge*)

Principe :

- 1** Divisions (*split*) successives de chaque région de l'image si elles ne satisfont pas le critère d'homogénéité
⇒ Permet d'aboutir à une « partition initiale ».
→ représentation par arbre.
- 2** Fusions (*merge*) successives des régions adjacentes satisfaisant le critère d'homogénéité.

Décomposition / fusion

(*Split and merge*)

Principe :

- 1 Divisions (*split*) successives de chaque région de l'image si elles ne satisfont pas le critère d'homogénéité
⇒ Permet d'aboutir à une « partition initiale ».
→ représentation par arbre.
- 2 Fusions (*merge*) successives des régions adjacentes satisfaisant le critère d'homogénéité.
→ représentation par graphe d'adjacence.

Décomposition / fusion (*Split and merge*)

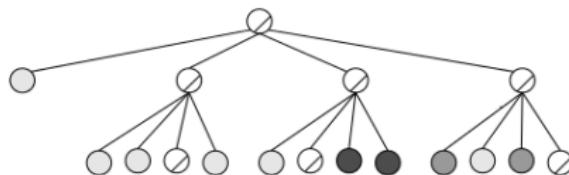
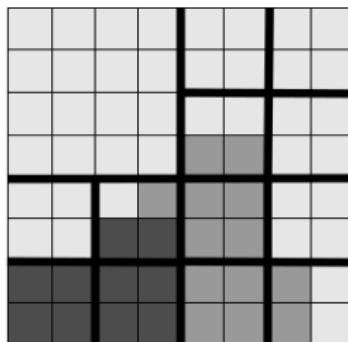
Principe :

- 1 Divisions (*split*) successives de chaque région de l'image si elles ne satisfont pas le critère d'homogénéité
⇒ Permet d'aboutir à une « partition initiale ».
→ représentation par arbre.
- 2 Fusions (*merge*) successives des régions adjacentes satisfaisant le critère d'homogénéité.
→ représentation par graphe d'adjacence.

Les représentations en arbre et par graphe permettent une représentation haut niveau de l'image.

Décomposition / fusion

Le **quad-arbre** (*quad-tree*) est une arborescence dont chaque nœud représente une région et possède quatre fils (la racine représente l'image entière).



Décomposition / fusion

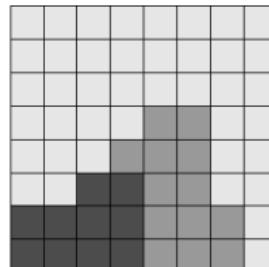
Décomposition

Chaque région R est partitionnée en quatre régions de taille identique si elle ne respecte pas le critère d'homogénéité.

Décomposition / fusion

Décomposition

Chaque région R est partitionnée en quatre régions de taille identique si elle ne respecte pas le critère d'homogénéité.



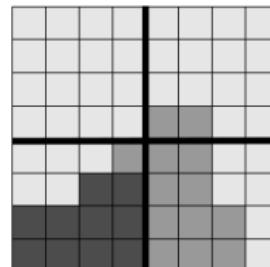
Représentation
par quad-arbre :



Décomposition / fusion

Décomposition

Chaque région R est partitionnée en quatre régions de taille identique si elle ne respecte pas le critère d'homogénéité.



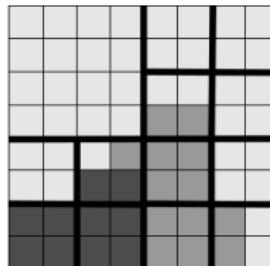
Représentation
par quad-arbre :



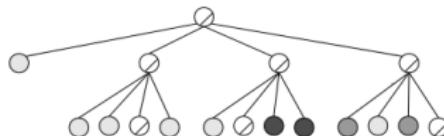
Décomposition / fusion

Décomposition

Chaque région R est partitionnée en quatre régions de taille identique si elle ne respecte pas le critère d'homogénéité.



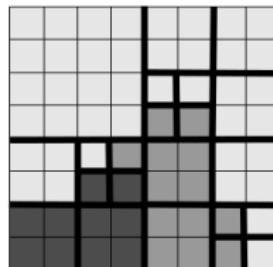
Représentation par quad-arbre :



Décomposition / fusion

Décomposition

Chaque région R est partitionnée en quatre régions de taille identique si elle ne respecte pas le critère d'homogénéité.



Représentation par quad-arbre :



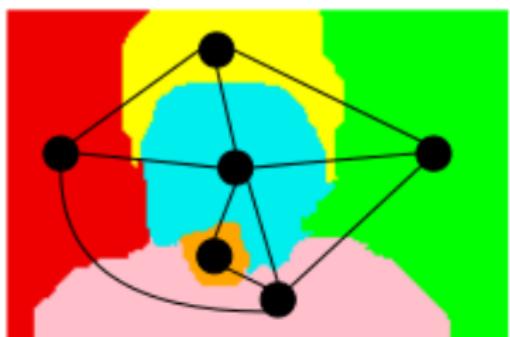
Décomposition / fusion

- La méthode de décomposition par quad-arbre fait apparaître des régions carrées sur l'image segmentée.
- Le problème majeur de cette structure provient de la rigidité des divisions réalisées sur l'image, mais cela fournit une partition initiale de l'image.

Décomposition / fusion

Le **graphe d'adjacence** (*RAG : region adjacency graph*) est un graphe dont :

- les nœuds correspondent à une région de l'image,
- les arêtes relient les nœuds correspondants à deux régions adjacentes (ayant une frontière commune).



Décomposition / fusion

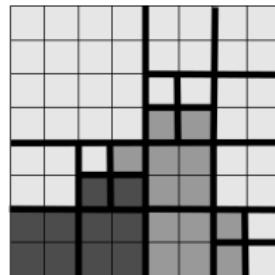
Fusion

À partir d'une partition de l'image (par exemple obtenue avec un quad-arbre), on fusionne les nœuds R_1 et R_2 voisins et dont le critère de similarité sur $R_1 \cup R_2$ est respecté.

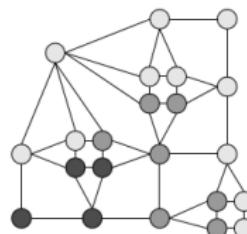
Décomposition / fusion

Fusion

À partir d'une partition de l'image (par exemple obtenue avec un quad-arbre), on fusionne les nœuds R_1 et R_2 voisins et dont le critère de similarité sur $R_1 \cup R_2$ est respecté.



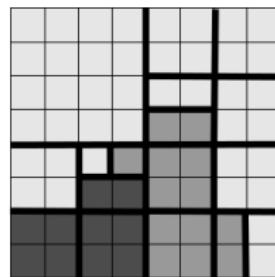
Représentation
par un graphe
d'adjacence :



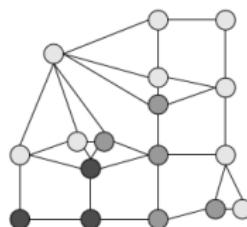
Décomposition / fusion

Fusion

À partir d'une partition de l'image (par exemple obtenue avec un quad-arbre), on fusionne les nœuds R_1 et R_2 voisins et dont le critère de similarité sur $R_1 \cup R_2$ est respecté.



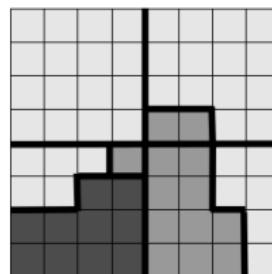
Représentation
par un graphe
d'adjacence :



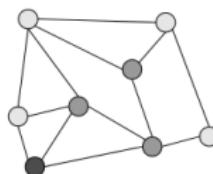
Décomposition / fusion

Fusion

À partir d'une partition de l'image (par exemple obtenue avec un quad-arbre), on fusionne les nœuds R_1 et R_2 voisins et dont le critère de similarité sur $R_1 \cup R_2$ est respecté.



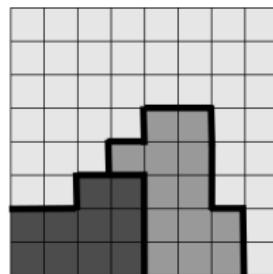
Représentation
par un graphe
d'adjacence :



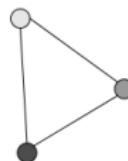
Décomposition / fusion

Fusion

À partir d'une partition de l'image (par exemple obtenue avec un quad-arbre), on fusionne les nœuds R_1 et R_2 voisins et dont le critère de similarité sur $R_1 \cup R_2$ est respecté.



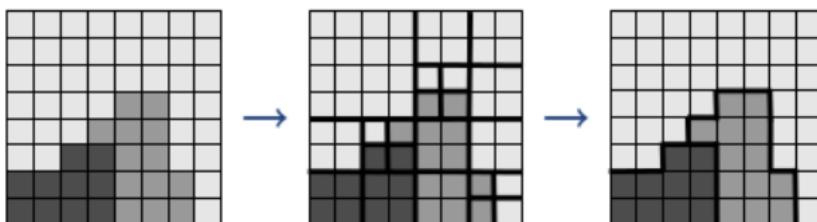
Représentation
par un graphe
d'adjacence :



Décomposition / fusion

Résumé

- 1 Partition initiale en fonction du critère d'homogénéité (par exemple avec un quad-arbre)
- 2 Fusion des zones segmentées adjacentes en fonction du critère d'homogénéité (représentation avec un graphe d'adjacence)



Ligne de partage des eaux (watershed)

Ligne de partage des eaux

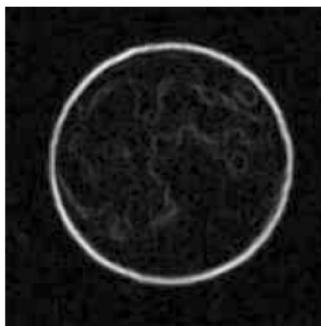
L'image est vue comme une carte d'élévation (terrain 3D) où :

- les régions sont les vallées
- les frontières entre régions sont les crêtes

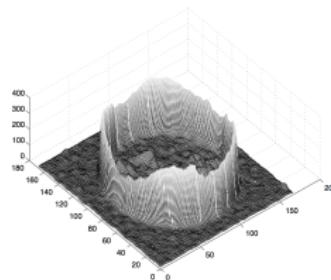
→ utilisation de la norme du gradient de l'image.



Image originale



Gradient de
l'image



Représentation 3D
du gradient

Ligne de partage des eaux

Principe :

- Construire la carte d'élévation
- Remplir progressivement d'eau chaque *bassin versant*
- Lorsque l'eau monte et que deux bassins se rejoignent, la ligne de partage des eaux est marquée comme frontière

Illustration sur un signal 1D :

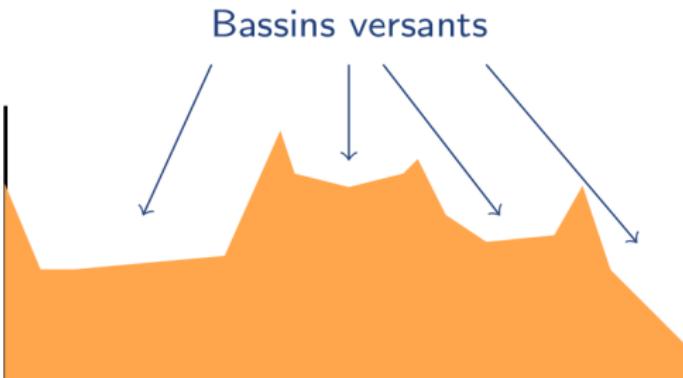


Ligne de partage des eaux

Principe :

- Construire la carte d'élévation
- Remplir progressivement d'eau chaque *bassin versant*
- Lorsque l'eau monte et que deux bassins se rejoignent, la ligne de partage des eaux est marquée comme frontière

Illustration sur un signal 1D :



Ligne de partage des eaux

Principe :

- Construire la carte d'élévation
- Remplir progressivement d'eau chaque *bassin versant*
- Lorsque l'eau monte et que deux bassins se rejoignent, la ligne de partage des eaux est marquée comme frontière

Illustration sur un signal 1D :



Ligne de partage des eaux

Principe :

- Construire la carte d'élévation
- Remplir progressivement d'eau chaque *bassin versant*
- Lorsque l'eau monte et que deux bassins se rejoignent, la ligne de partage des eaux est marquée comme frontière

Illustration sur un signal 1D :



Ligne de partage des eaux

Principe :

- Construire la carte d'élévation
- Remplir progressivement d'eau chaque *bassin versant*
- Lorsque l'eau monte et que deux bassins se rejoignent, la ligne de partage des eaux est marquée comme frontière

Illustration sur un signal 1D :

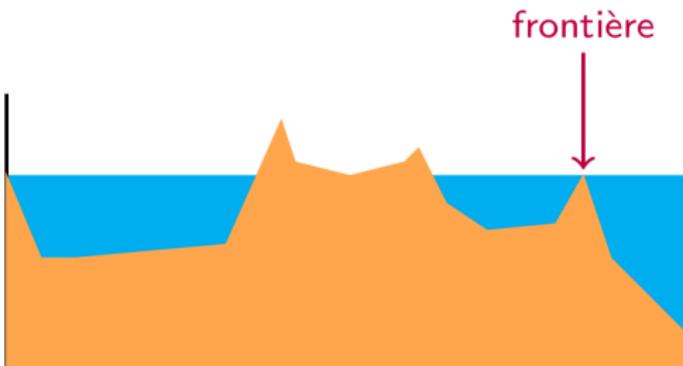


Ligne de partage des eaux

Principe :

- Construire la carte d'élévation
- Remplir progressivement d'eau chaque *bassin versant*
- Lorsque l'eau monte et que deux bassins se rejoignent, la ligne de partage des eaux est marquée comme frontière

Illustration sur un signal 1D :

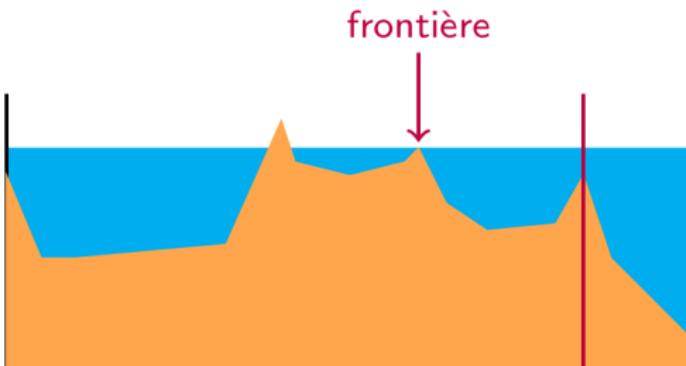


Ligne de partage des eaux

Principe :

- Construire la carte d'élévation
- Remplir progressivement d'eau chaque *bassin versant*
- Lorsque l'eau monte et que deux bassins se rejoignent, la ligne de partage des eaux est marquée comme frontière

Illustration sur un signal 1D :

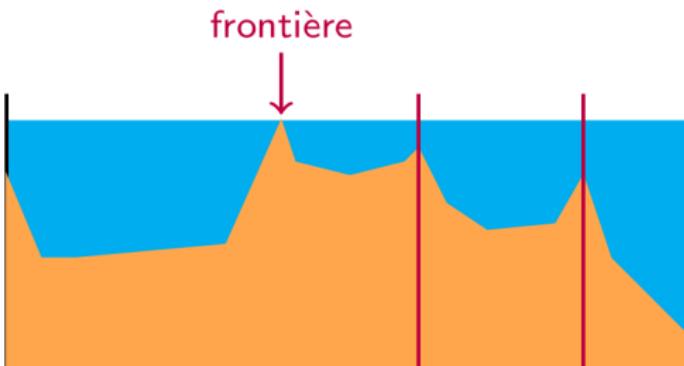


Ligne de partage des eaux

Principe :

- Construire la carte d'élévation
- Remplir progressivement d'eau chaque *bassin versant*
- Lorsque l'eau monte et que deux bassins se rejoignent, la ligne de partage des eaux est marquée comme frontière

Illustration sur un signal 1D :



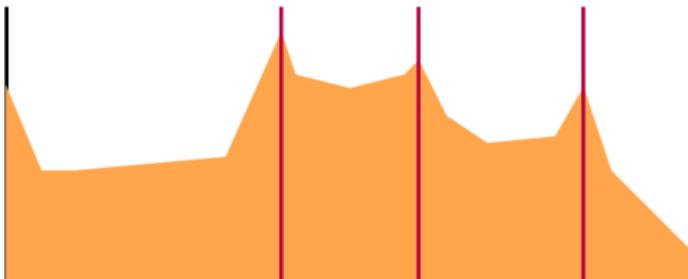
Ligne de partage des eaux

Principe :

- Construire la carte d'élévation
- Remplir progressivement d'eau chaque *bassin versant*
- Lorsque l'eau monte et que deux bassins se rejoignent, la ligne de partage des eaux est marquée comme frontière

Illustration sur un signal 1D :

Segmentation finale



Ligne de partage des eaux

Algorithme :

Calculer le gradient (ou le Laplacien) de l'image.

Les pixels ayant l'intensité la plus faible forment les bassins versants initiaux.

Pour chaque niveau d'intensité i :

 Pour chaque groupe de pixels d'intensité i :

 Si adjacents à exactement une région existante :
 ajouter ces pixels dans cette région.

 Si adjacents à plusieurs régions simultanément :
 marquer comme ligne de partage des eaux.

 Sinon, commencer une nouvelle région.

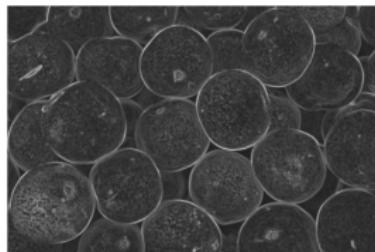
Ligne de partage des eaux

Limitation

Sur-segmentation s'il y a beaucoup de minima locaux dans le gradient.



Image



Gradient



LPE

Ligne de partage des eaux

Limitation

Sur-segmentation s'il y a beaucoup de minima locaux dans le gradient.

Solutions possibles :

- lisser (filtrage passe-bas) le gradient avant d'appliquer l'algorithme
- choisir manuellement les bassins versants d'intérêt avec des marqueurs
- fusionner les minima locaux

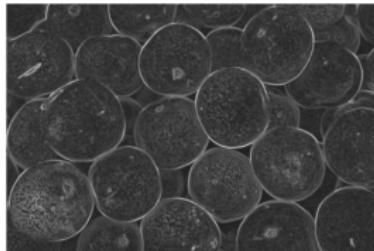
Ligne de partage des eaux

Limitation

Sur-segmentation s'il y a beaucoup de minima locaux dans le gradient.



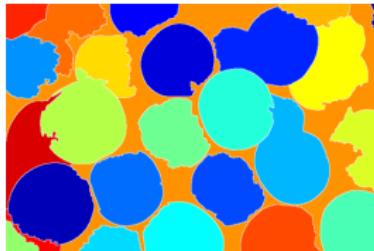
Image



Gradient



LPE



Utilisation de marqueurs

Contours actifs

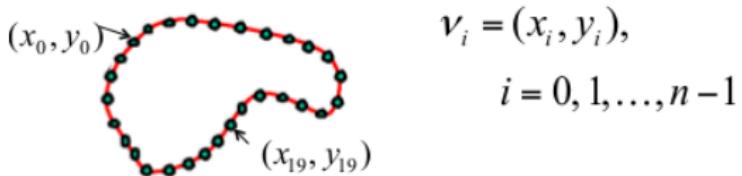
Contours actifs (*snakes*)

Principe : à partir d'un contour initial proche de l'objet à segmenter, le contour évolue de manière itérative et cherche à converger vers les zones de fort gradient (= contour) sous certaines contraintes (forme, longueur, etc.).

Contours actifs (snakes)

Principe : à partir d'un contour initial proche de l'objet à segmenter, le contour évolue de manière itérative et cherche à converger vers les zones de fort gradient (= contour) sous certaines contraintes (forme, longueur, etc.).

Le contour est modélisé par un ensemble de points (x_i, y_i) qui se déplacent légèrement à chaque itération pour déformer le contour.



Contours actifs (*snakes*)

Le contour cherche à minimiser une énergie (ou fonction coût) qui mesure la qualité de la segmentation :

$$E_{\text{totale}} = E_{\text{interne}} + \lambda E_{\text{externe}}$$

- Énergie interne : encourage certaines configurations de forme (régularité, élasticité, a priori de forme, ...)
- Énergie externe : encourage le modèle à converger vers les contours des objets (zones de fort gradient)

Évaluation

Critères d'évaluation de la segmentation

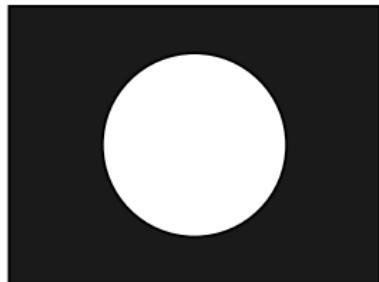


Vérité terrain f^*



Segmentation f

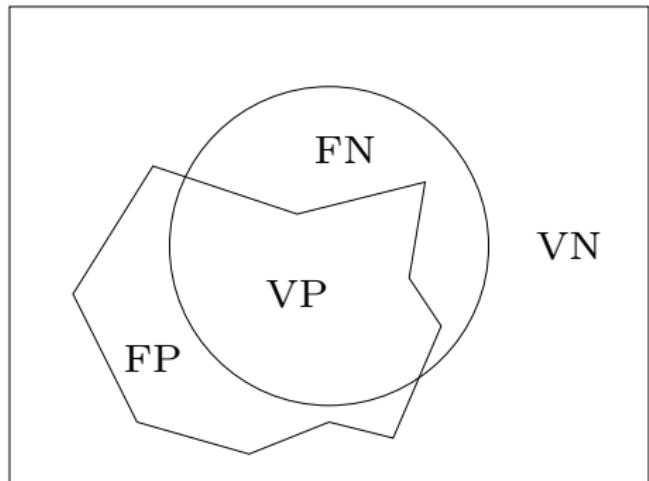
Critères d'évaluation de la segmentation



Vérité terrain f^*



Segmentation f



VP : vrai positif

VN : vrai négatif

FP : faux positif

FN : faux négatif

Critères d'évaluation de la segmentation

- Sensibilité = $\frac{\text{VP}}{\text{VP} + \text{FN}}$
- Spécificité = $\frac{\text{VN}}{\text{VN} + \text{FP}}$
- Dice = $\frac{2 \text{ VP}}{2 \text{ VP} + \text{ FP} + \text{ FN}} = \frac{2 |f \cap f^*|}{|f| + |f^*|}$
- Jaccard = $\frac{\text{VP}}{\text{VP} + \text{FP} + \text{FN}} = \frac{|f \cap f^*|}{|f \cup f^*|}$

Conclusion

Conclusion

- La segmentation consiste à diviser l'image en plusieurs régions homogènes.
- L'homogénéité d'une région est basée sur la couleur, la texture, les contours...
- Les méthodes de segmentation sont très diverses : classification, croissance de région, décomposition/fusion, contours actifs, ensembles de niveaux (*level sets*), ligne de partage des eaux, entropie, modèles markoviens...

Bibliographie

- R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, S. Süsstrunk, « SLIC Superpixels Compared to State-of-the-art Superpixel Methods », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11), p. 2274–2282, 2012.
- K. Fukunaga, L.D. Hostetler, « The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition », *IEEE Transactions on Information Theory*, 21(1) p. 32–40, 1975.
- J.B. MacQueen, « Some Methods for classification and Analysis of Multivariate Observations », 5th Berkeley Symposium on Mathematical Statistics and Probability., p. 281–297, 1967.
- N. Otsu, « A threshold selection method from gray-level histograms », *IEEE Transactions on Systems, Man, and Cybernetics* 9(1) p. 62–66, 1979.
- M. Sezgin, B. Sankur, « Survey over image thresholding techniques and quantitative performance evaluation », *Journal of Electronic Imaging* 13(1), p. 146–165, 2004.
- H. Steinhaus, « Sur la division des corps matériels en parties » *Bulletin de l'Académie Polonaise des Sciences*, 4(12) p. 801–804, 1957.