Session : Segmentation d'images

- Segmenter : Définition, rôle
- Classification : *K-means, Mean-shift*
- Contour déformable : *Snakes, Gradient vector flow*
- Séparation de graphes : *Graph-cuts*
- Région déformable : *Level set (Chan-Vese)*
- Modèles profonds : *Unet, SAM*

Perception d’une image

Image ≠ pixels

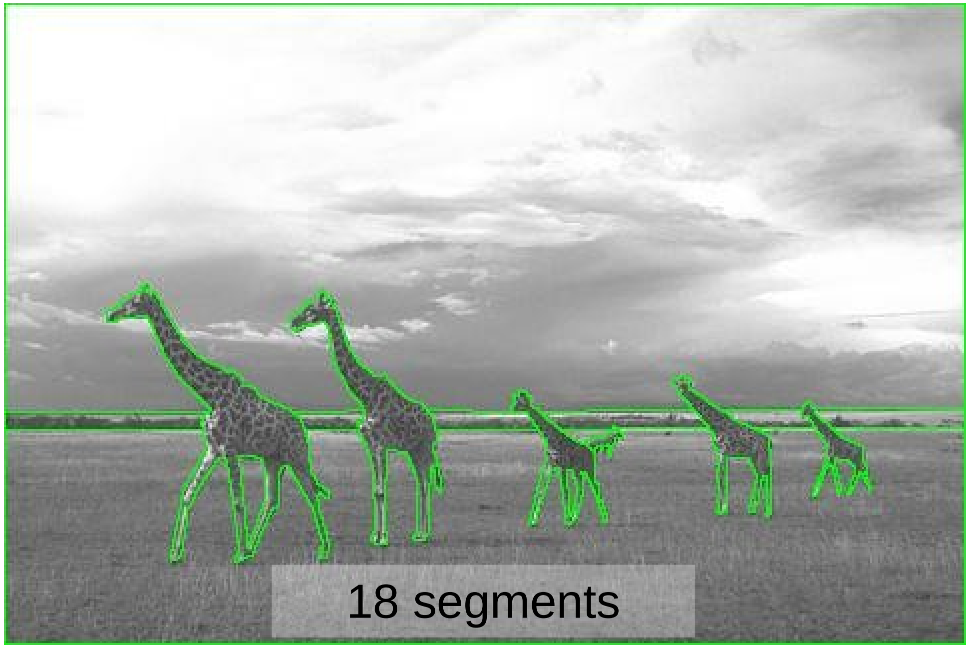
Image = objets, formes

Berkeley segmentation dataset
(https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds)

Perception d’une image

Image ≠ pixels

Image = objets, formes



Berkeley segmentation dataset (https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds)

Séparation objet/fond

Segmentation sémantique

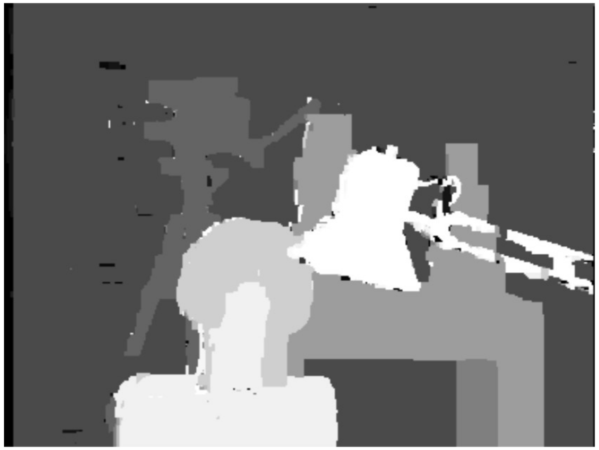
Suivi d’objet

« Bilateral space video segmentation ». Märki et al. CVPR 2016

Reconstruction stéréo

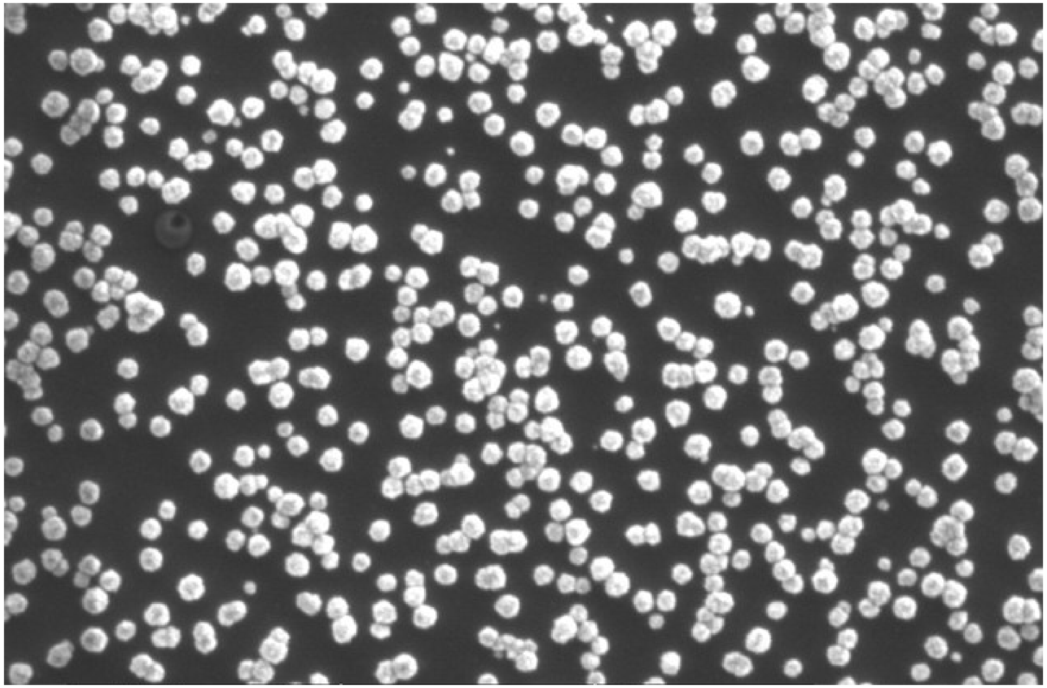
Image de gauche

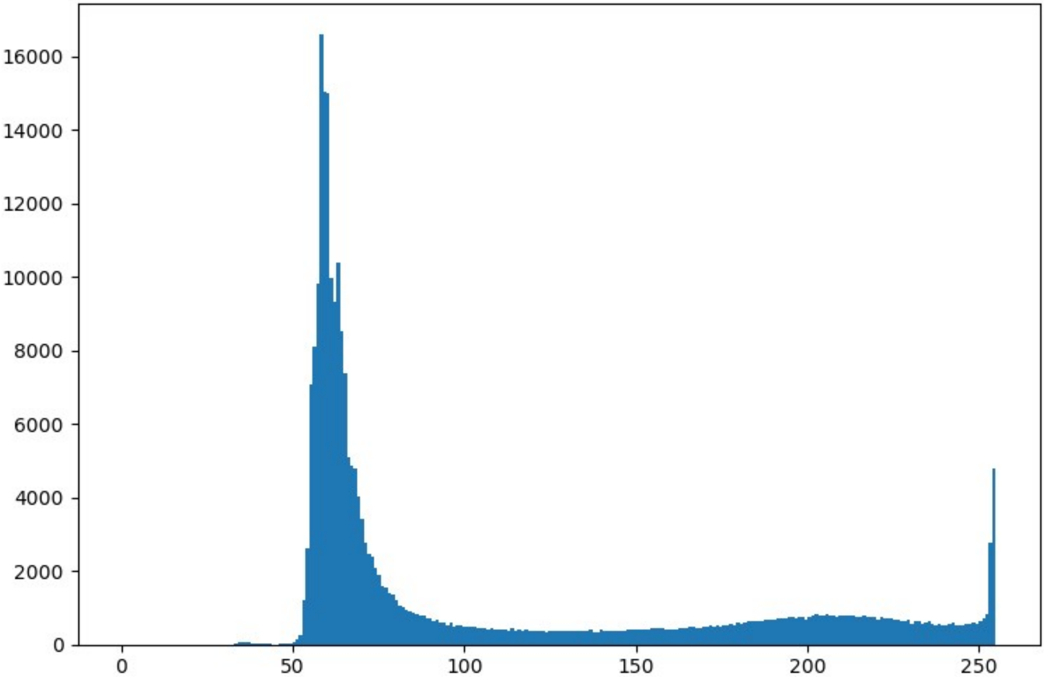
Image de droite



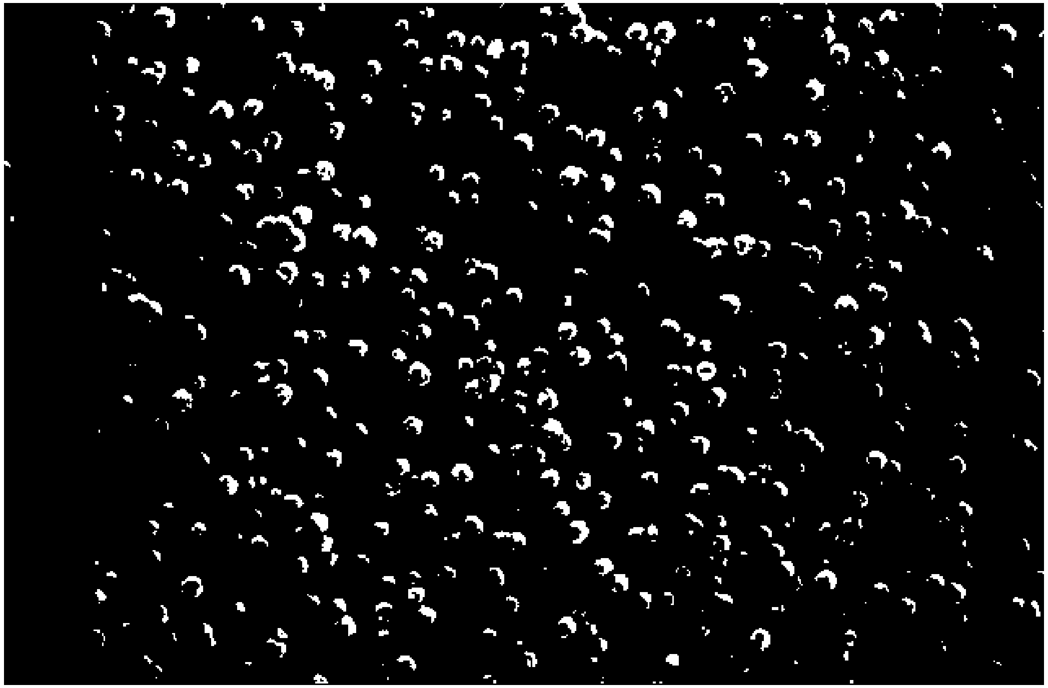
Carte de disparité (graph-cuts)

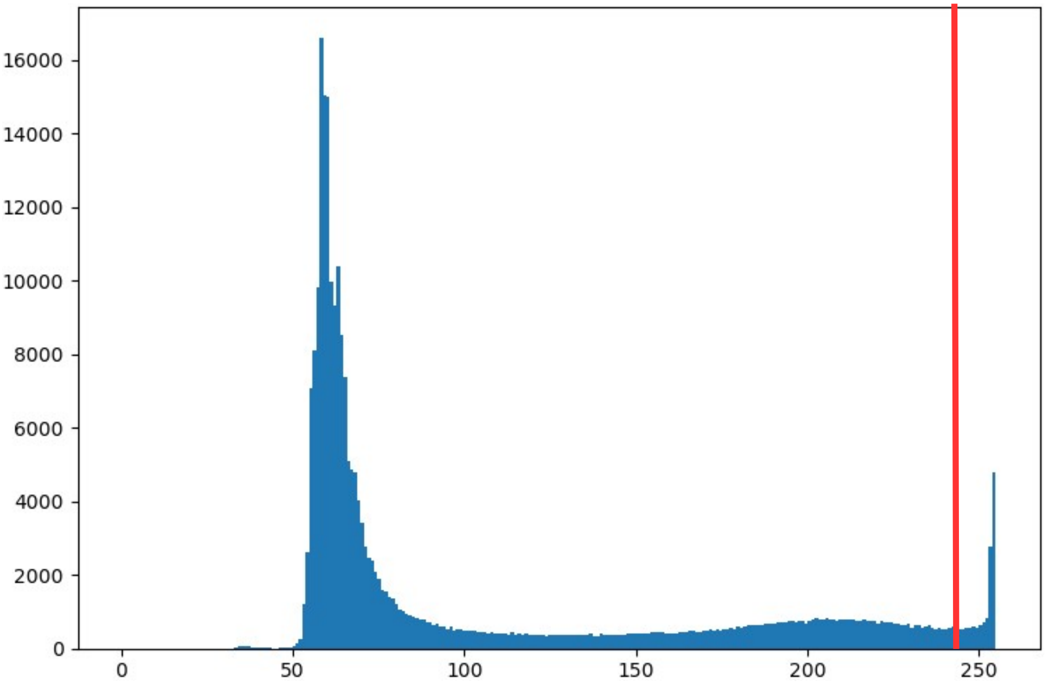
Méthode naïve : seuillage



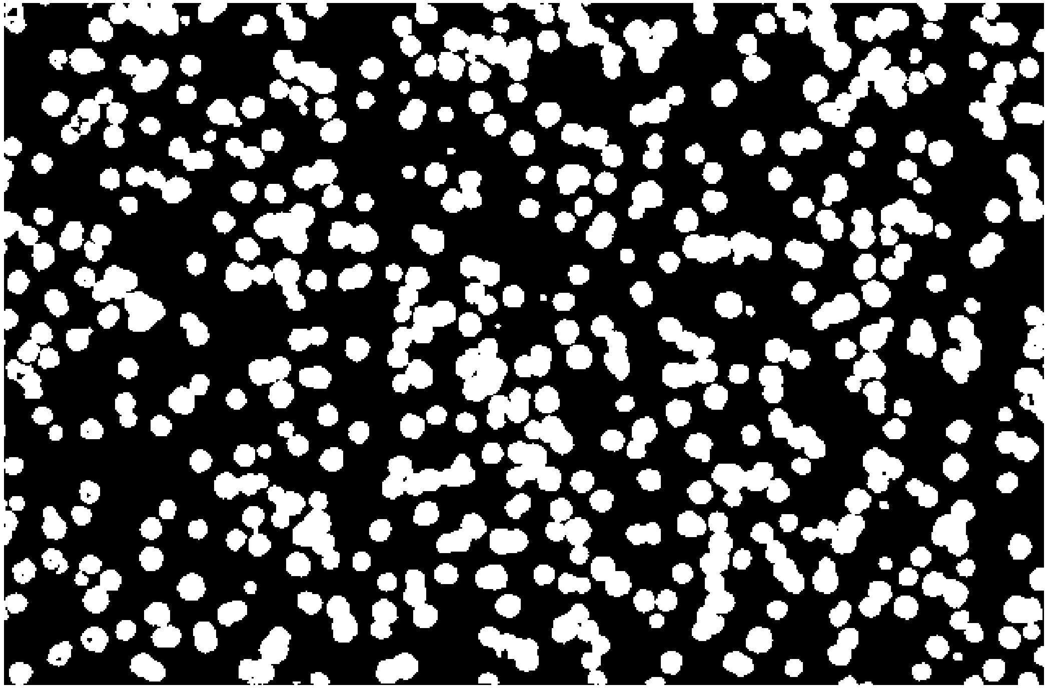


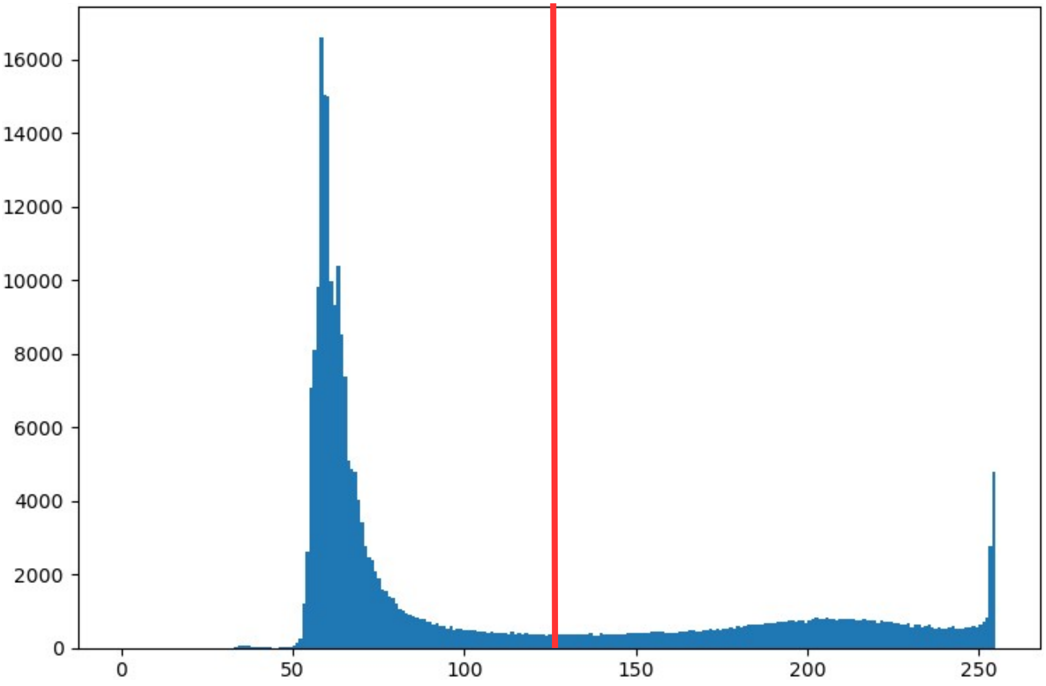
Méthode naïve : seuillage



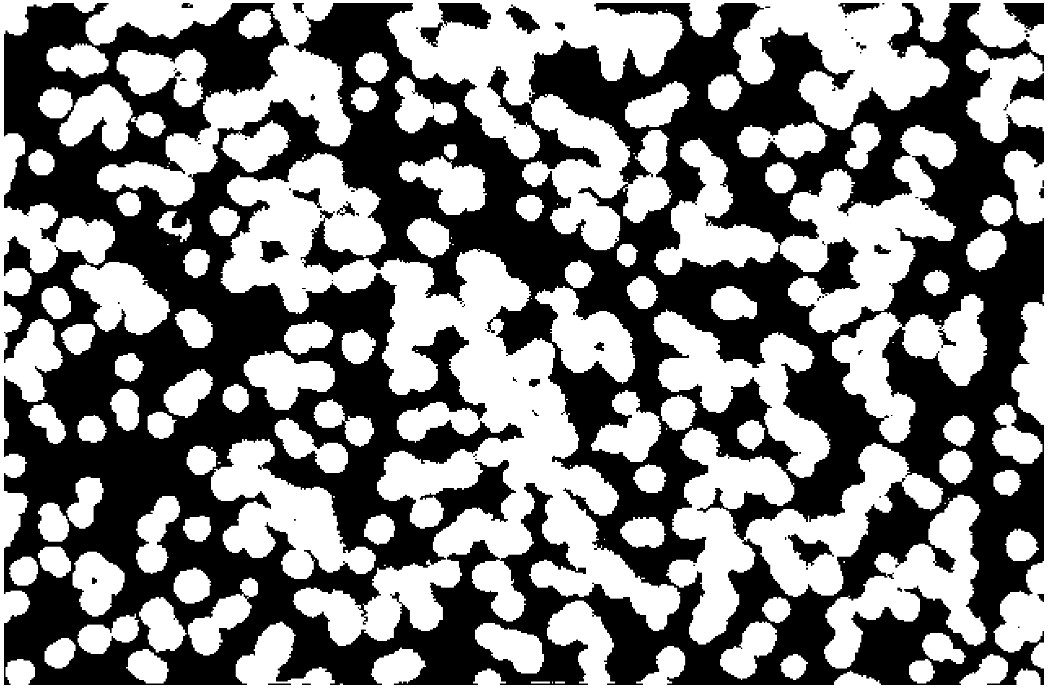


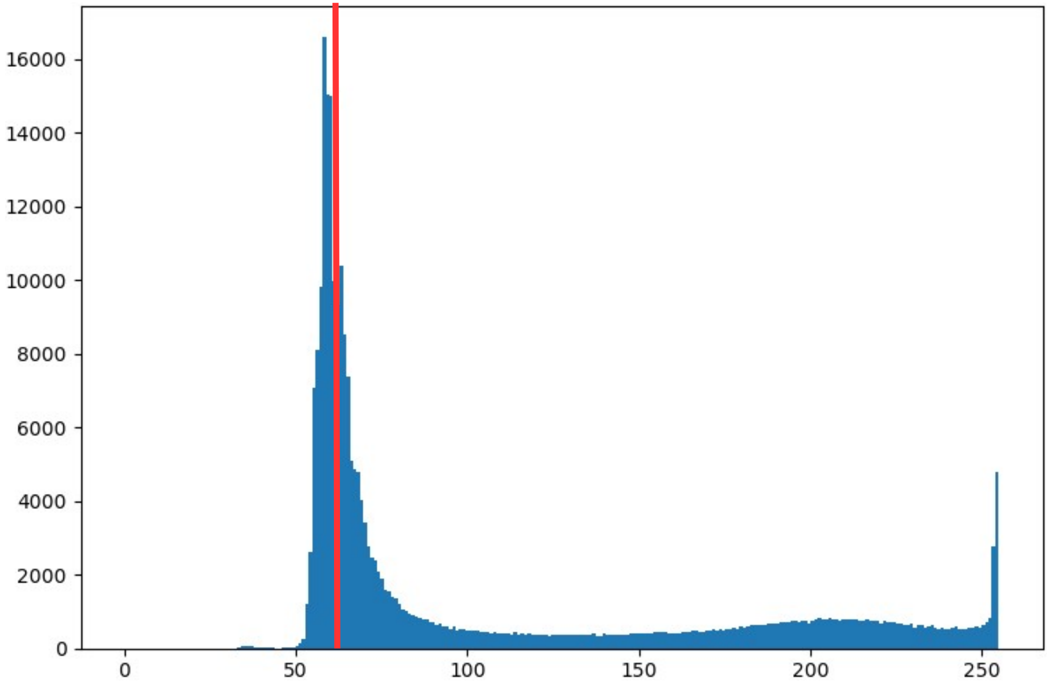
Méthode naïve : seuillage



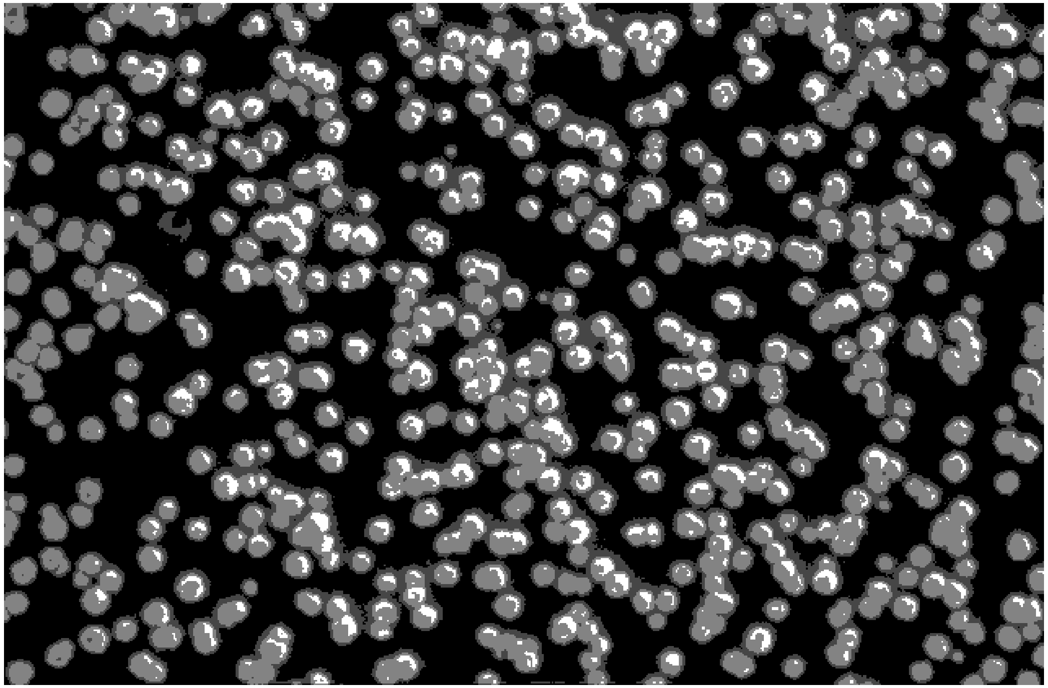


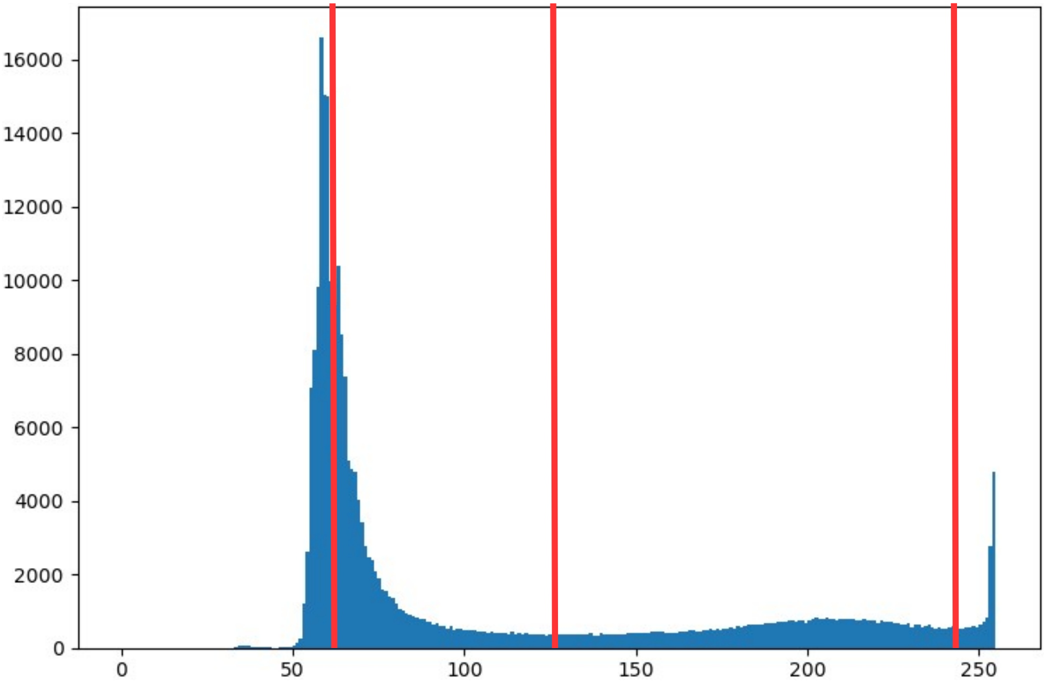
Méthode naïve : seuillage



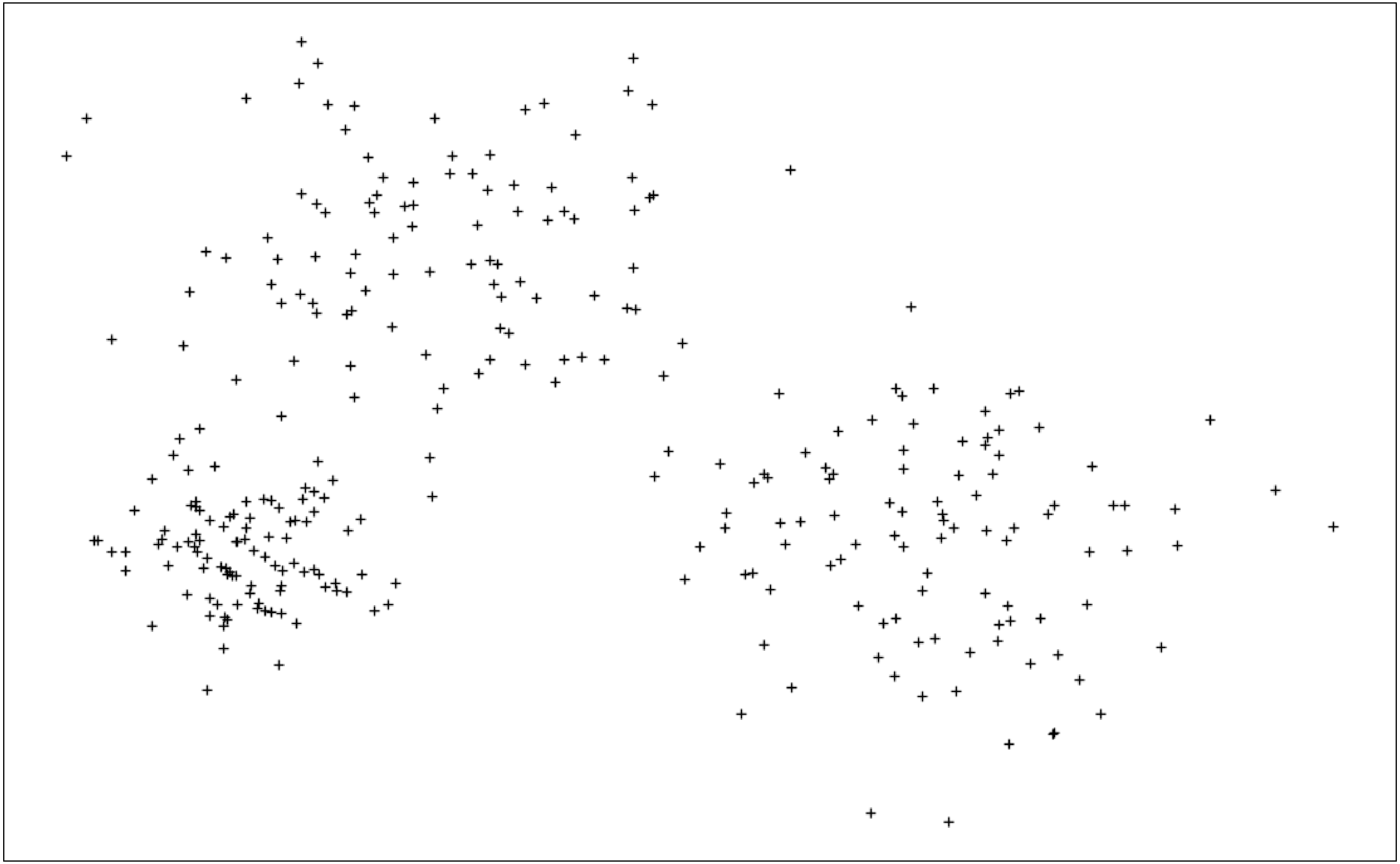


Méthode naïve : seuillage

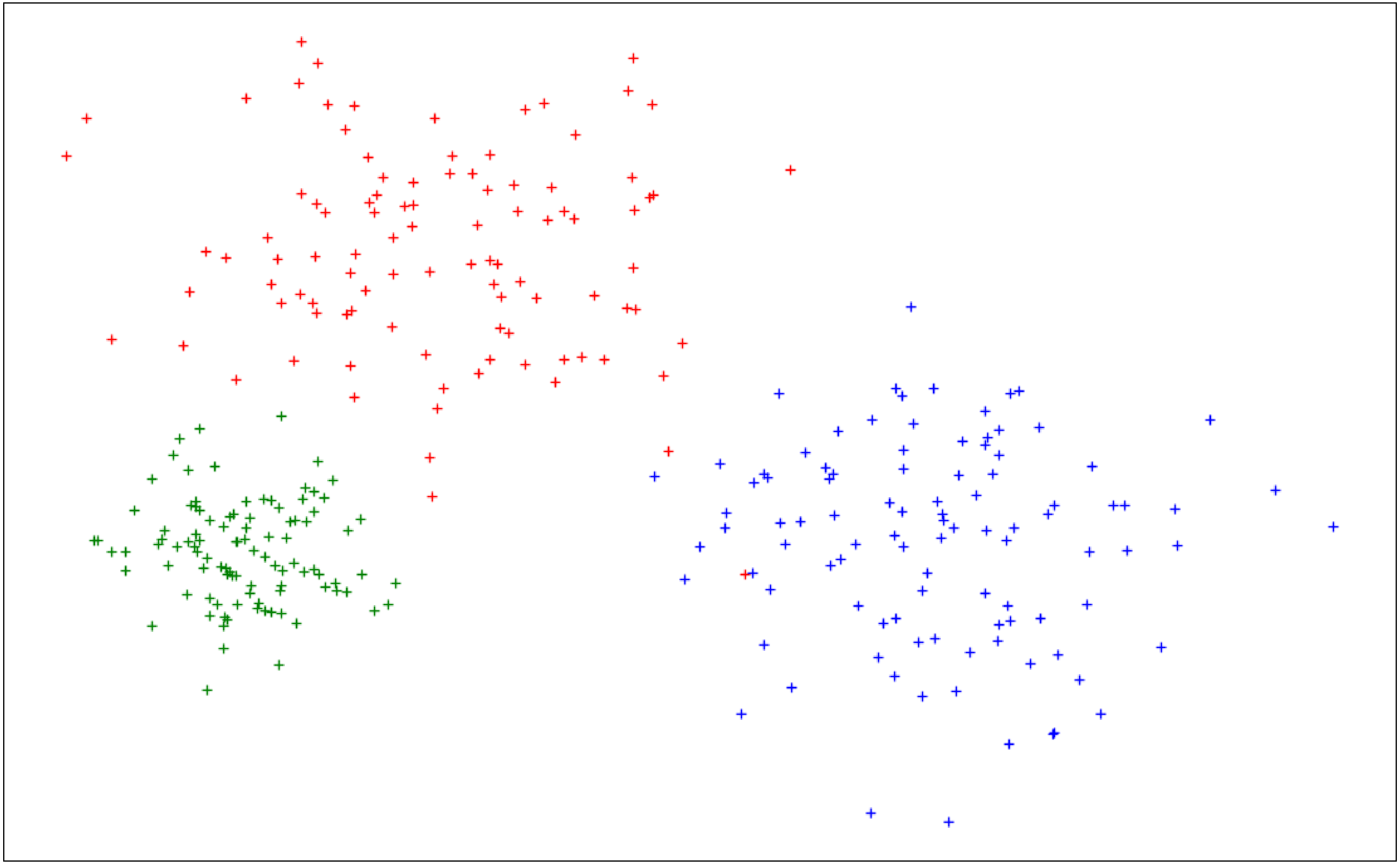




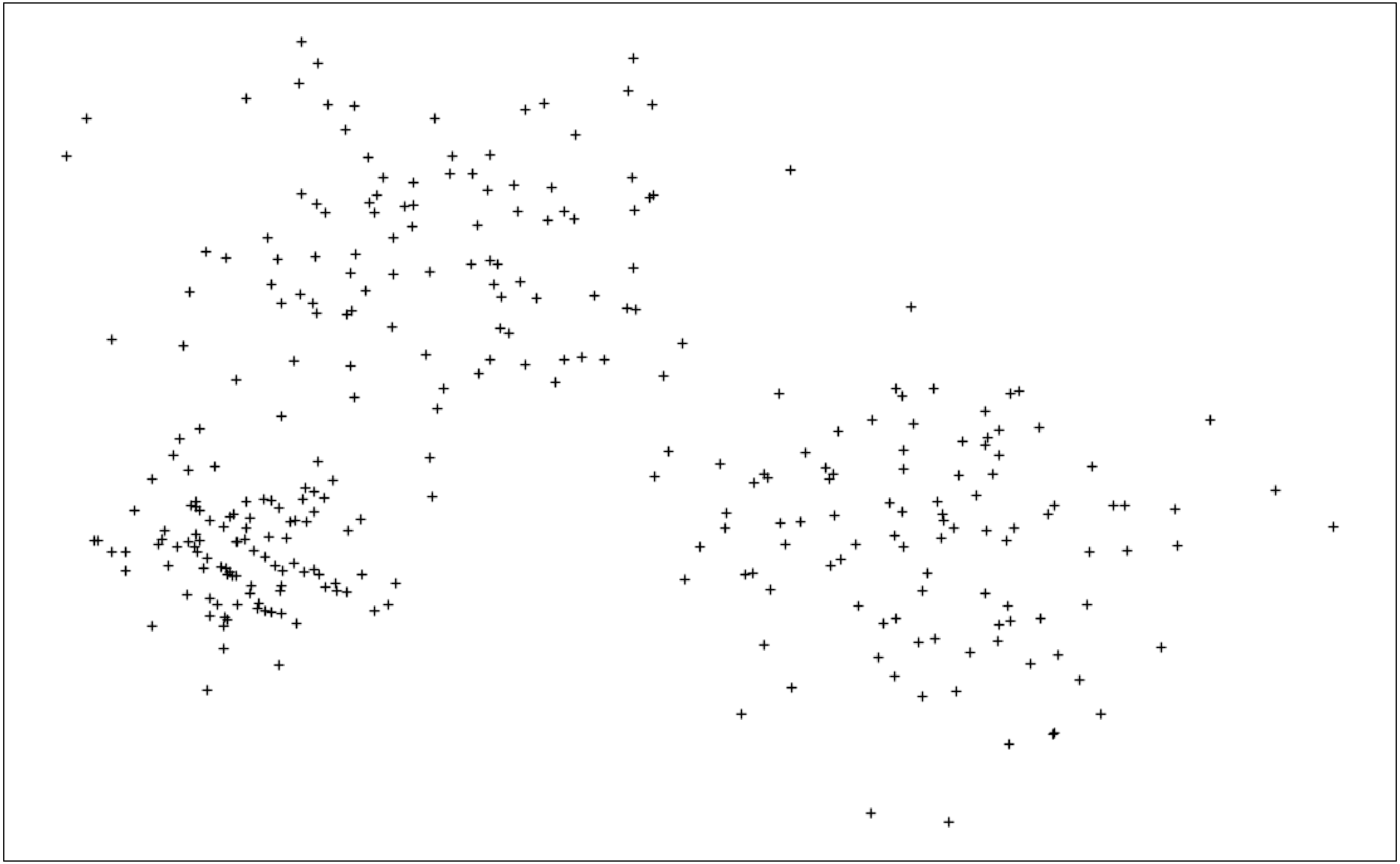
K-means



K-means

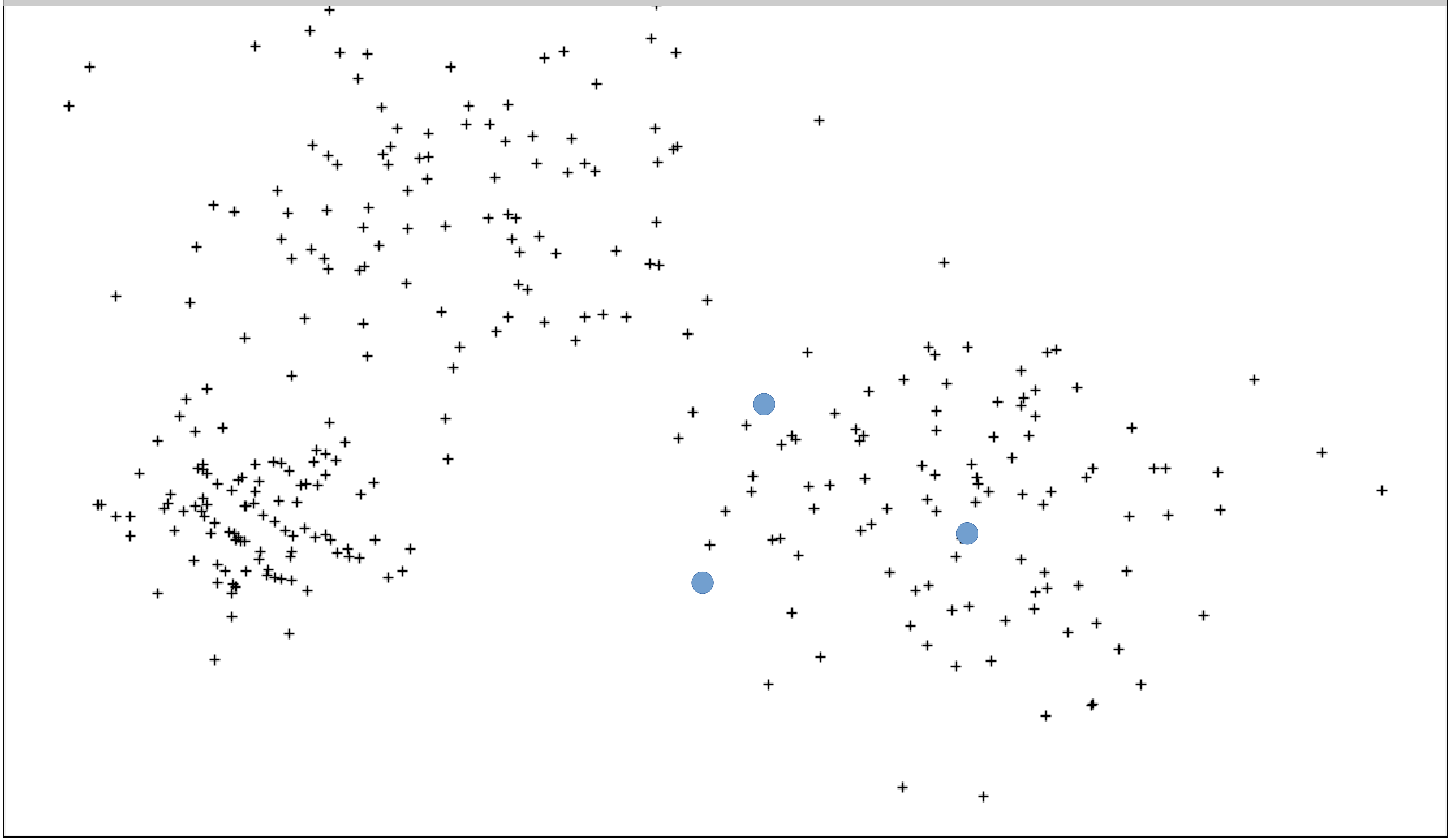


K-means



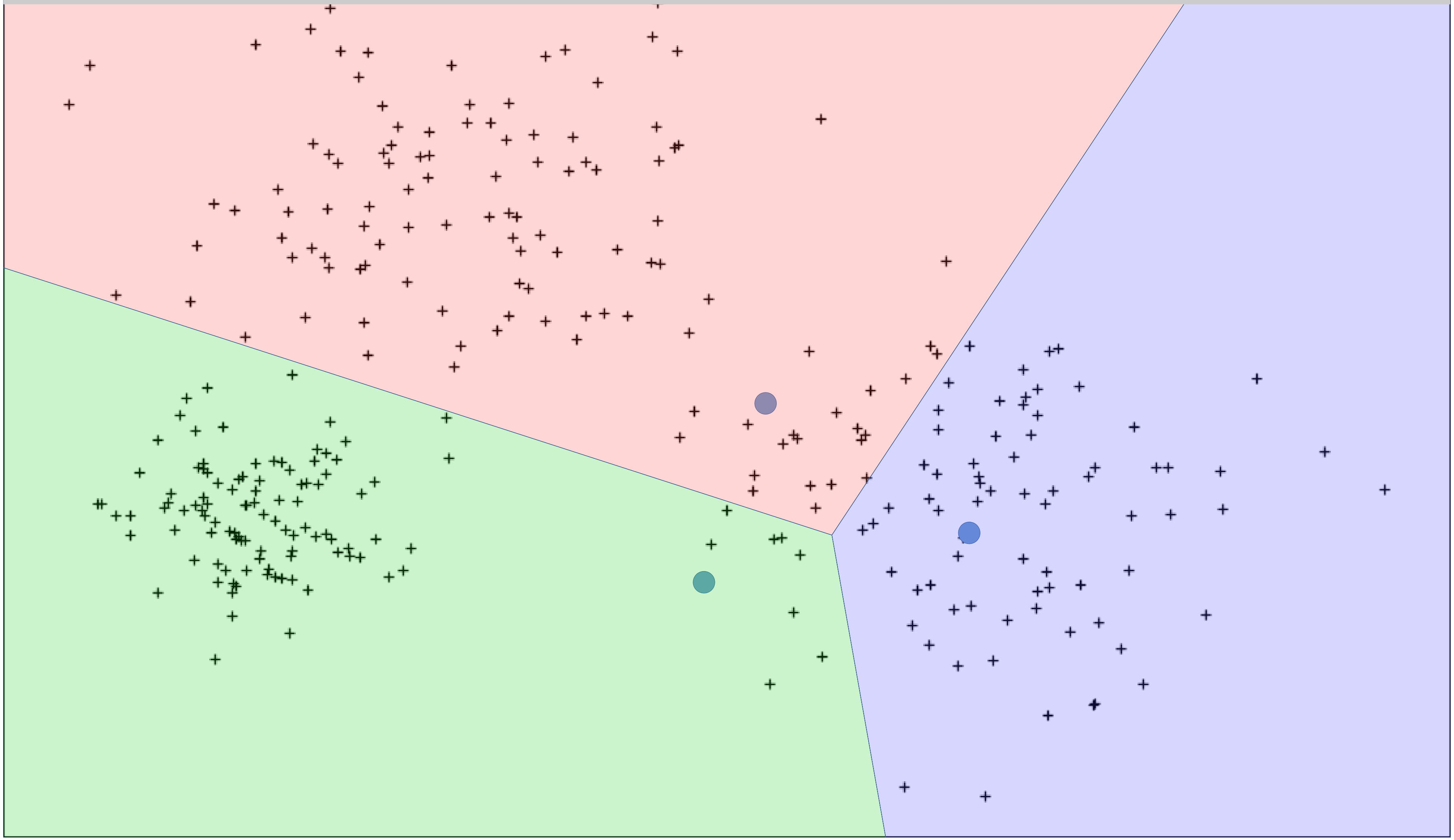
K-means

Supposons que des centres sont donnés



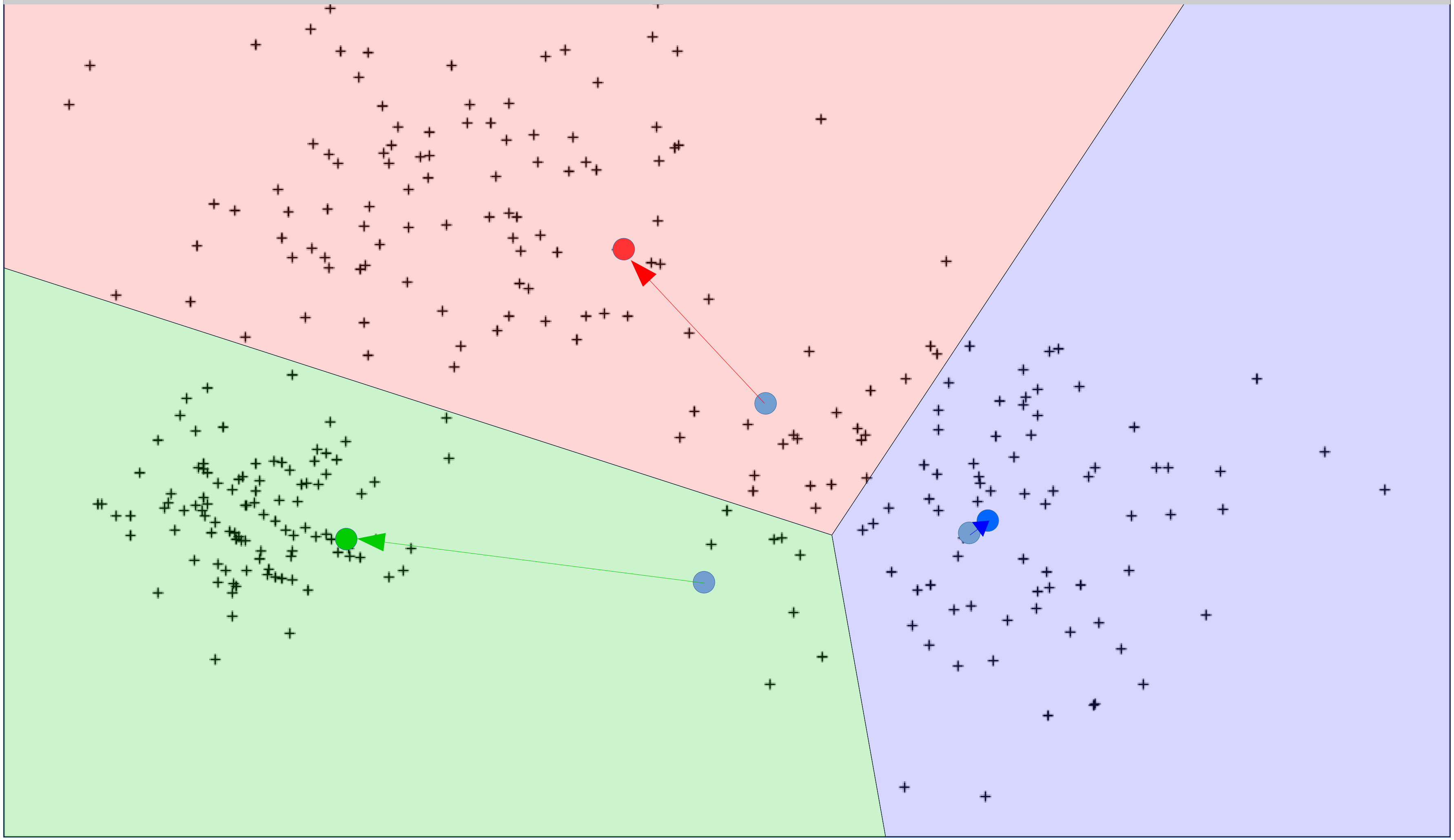
K-means

On détermine la classe des points (appariés au centre le plus proche)



K-means

On met à jour les centres (points moyens par classe)… Et on itère.



K-means : Algorithme

1) K classes S_t

2) Initialiser les centres aléatoirement $(c_1,\ldots c_K)$

3) Déterminer la classe t des points $\{p_i\}$ (centre le plus proche)

$$t_i = \underset{1 \leq t \leq K}{\operatorname{argmin}} |p_i - c_t|$$

4) Mettre à jour les centres (moyenne sur la classe)

$$c_t = \frac{1}{\operatorname{card}(S_t)} \sum_{p_i \in S_t} p_i$$

5) Si les centres ont bougé, itérer à partir de 3)

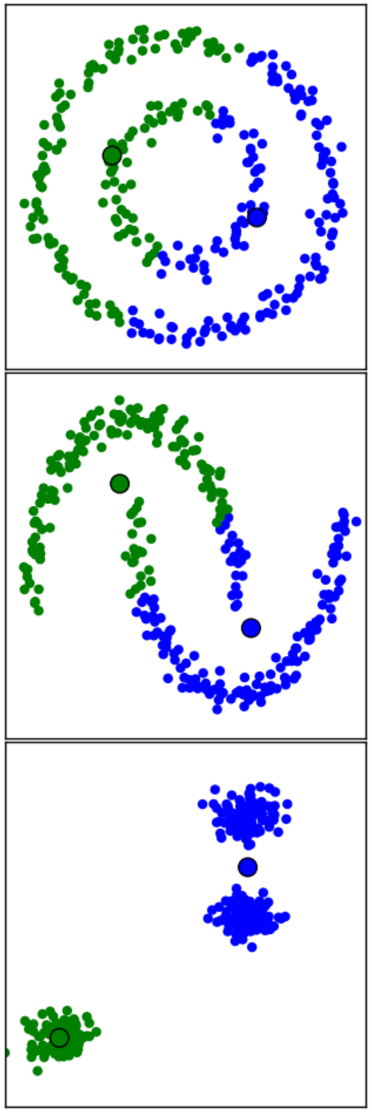
K-means : Pros & Cons

Pros

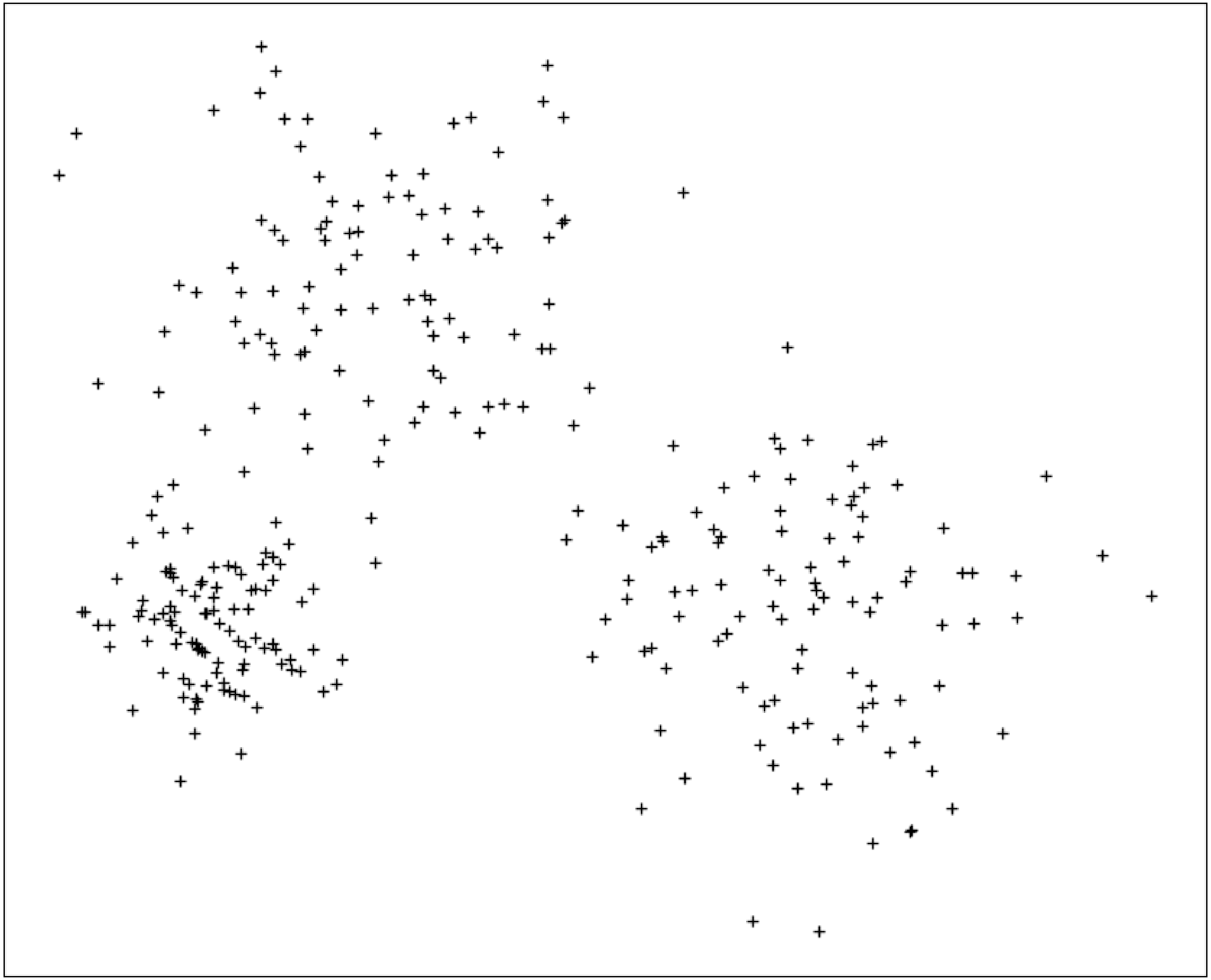
- Très simple
- Converge vers un minimum local

Cons

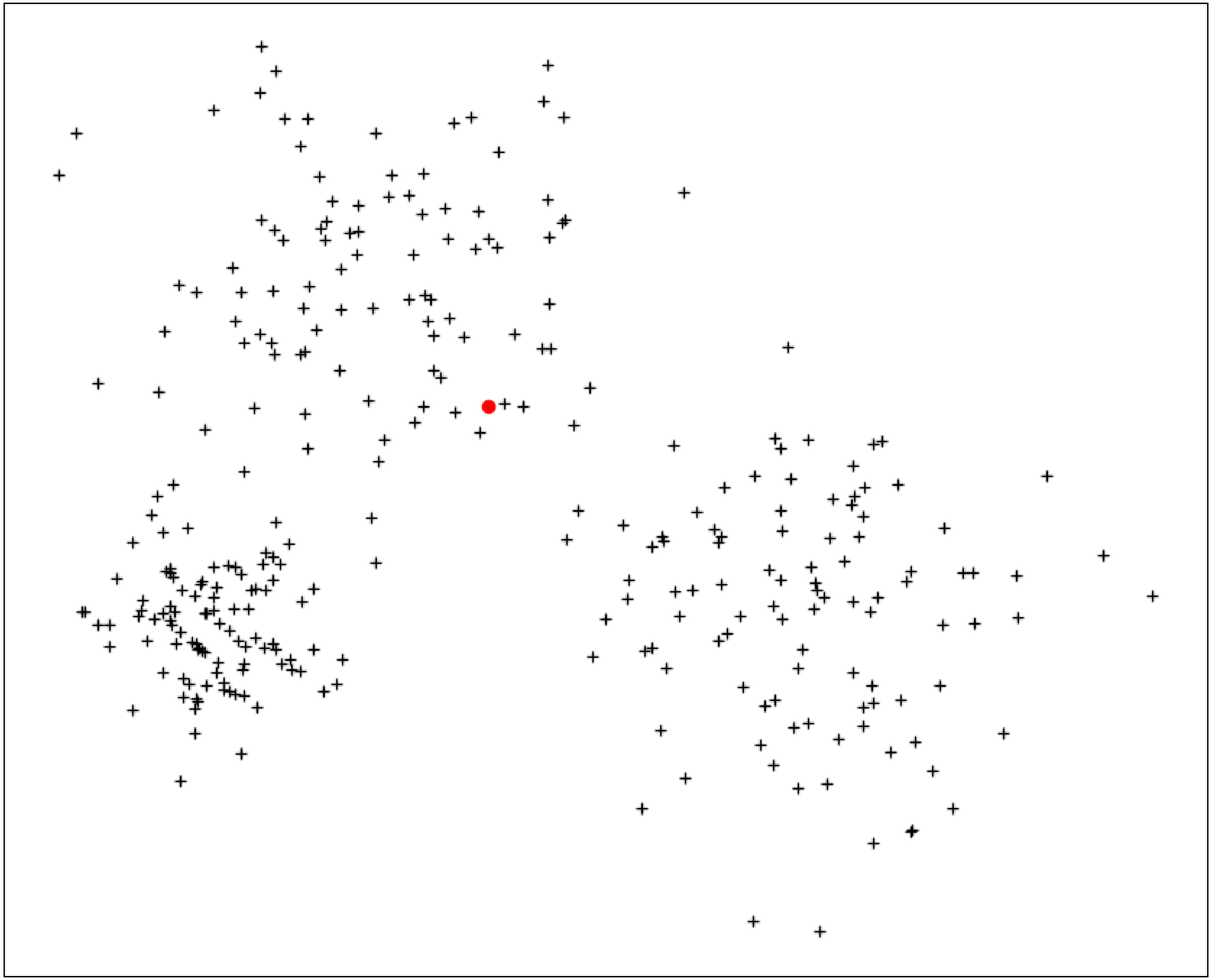
- Gourmand en mémoire
- Nécessité de choisir K a priori
- Sensible à l'initialisation (minimum local)
- Sensible aux points aberrants (critère quadratique)
- Uniquement adapté aux classes de forme circulaire



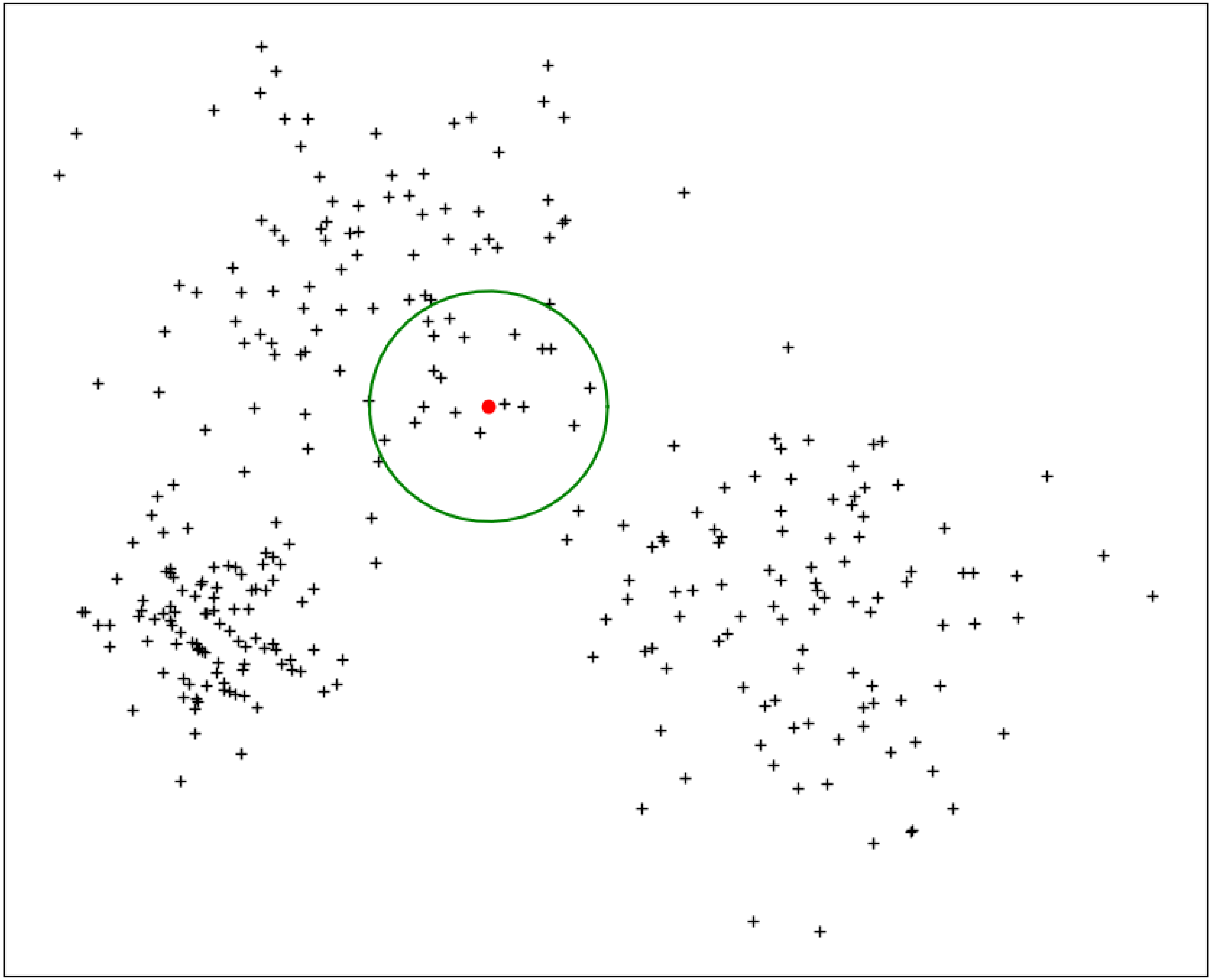
Mean-shift



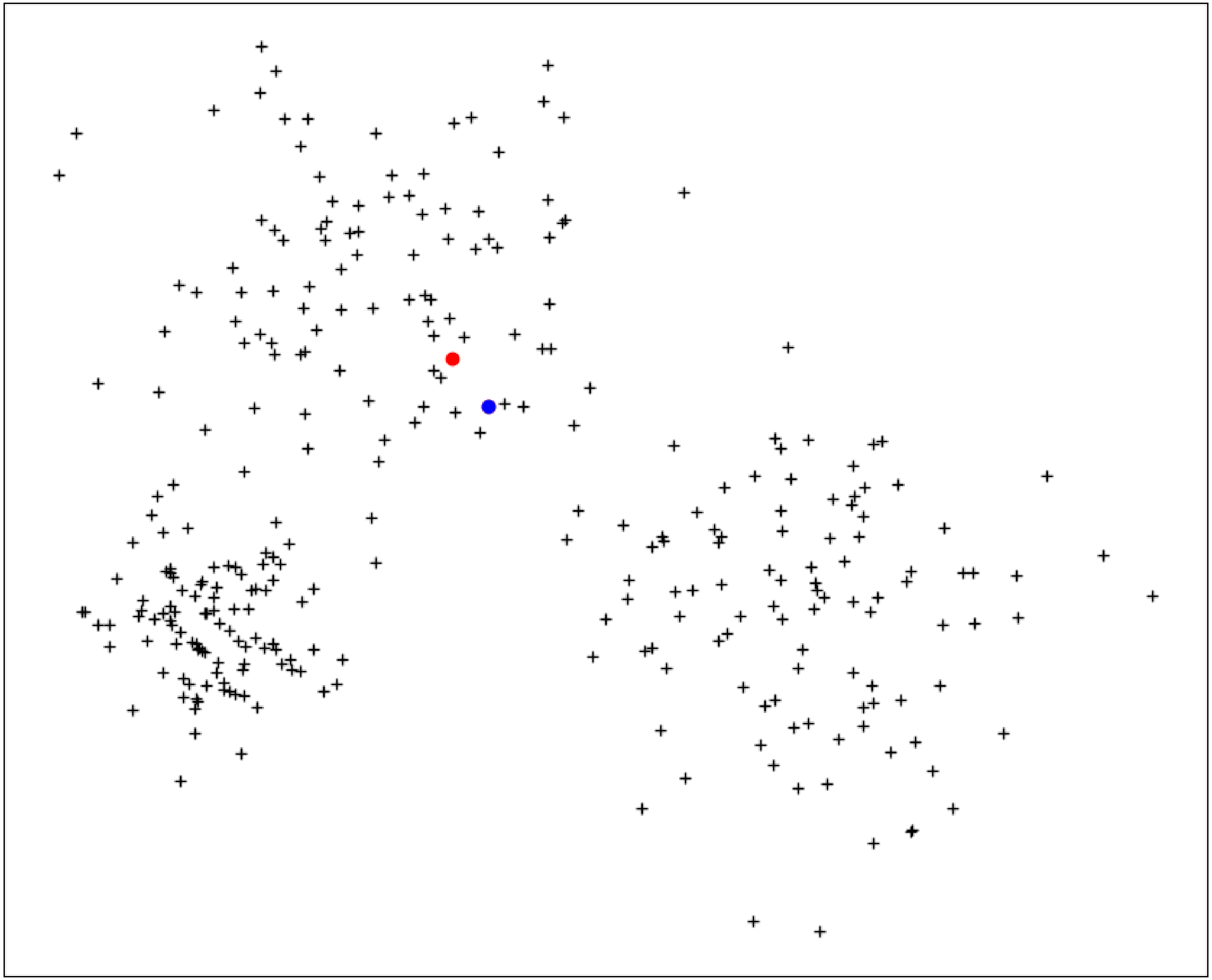
Mean-shift



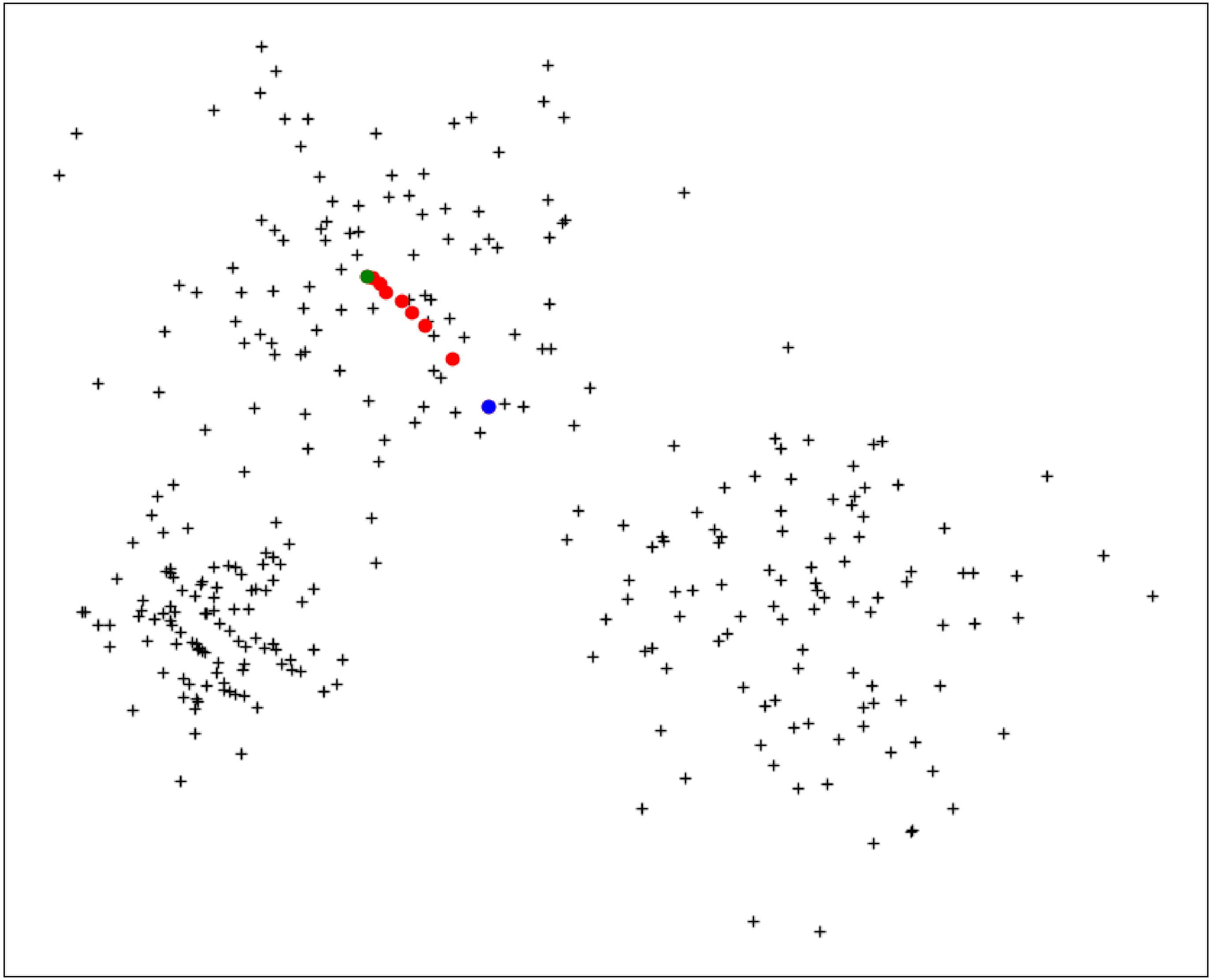
Mean-shift



Mean-shift



Mean-shift



Mean-shift : algorithme

Pour tous les points

1) Initialiser le centre avec le point
2) Calculer la moyenne sur un voisinage
3) Mettre à jour le centre avec cette moyenne
4) Itérer jusqu'à convergence

Centres = points d'accumulation

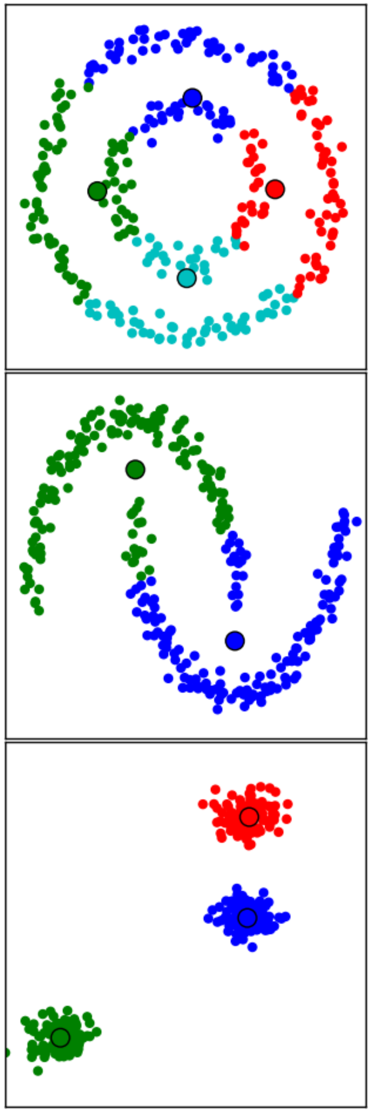
Mean-shift: Pros & Cons

Pros

- Nombre de classes automatique
- Un seul paramètre (taille du voisinage pour la moyenne)
- Générique (pas d'hypothèse de forme de classe)

Cons

- Difficulté de choix de la taille du voisinage
- Très gourmand en temps de calcul
- Problème de passage à l'échelle si la dimension de l'espace des caractéristiques augmente



Espace de caractéristiques

Points ≠ pixels

Points = mesure stockée en chaque pixel

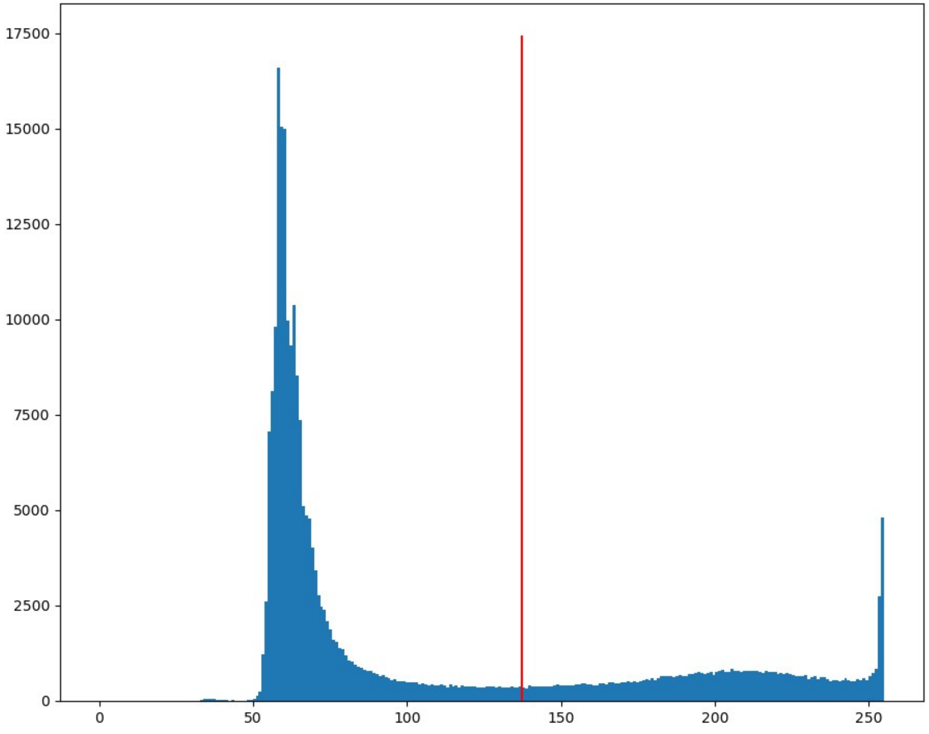
Ex : image N&B = intensité (une dimension)

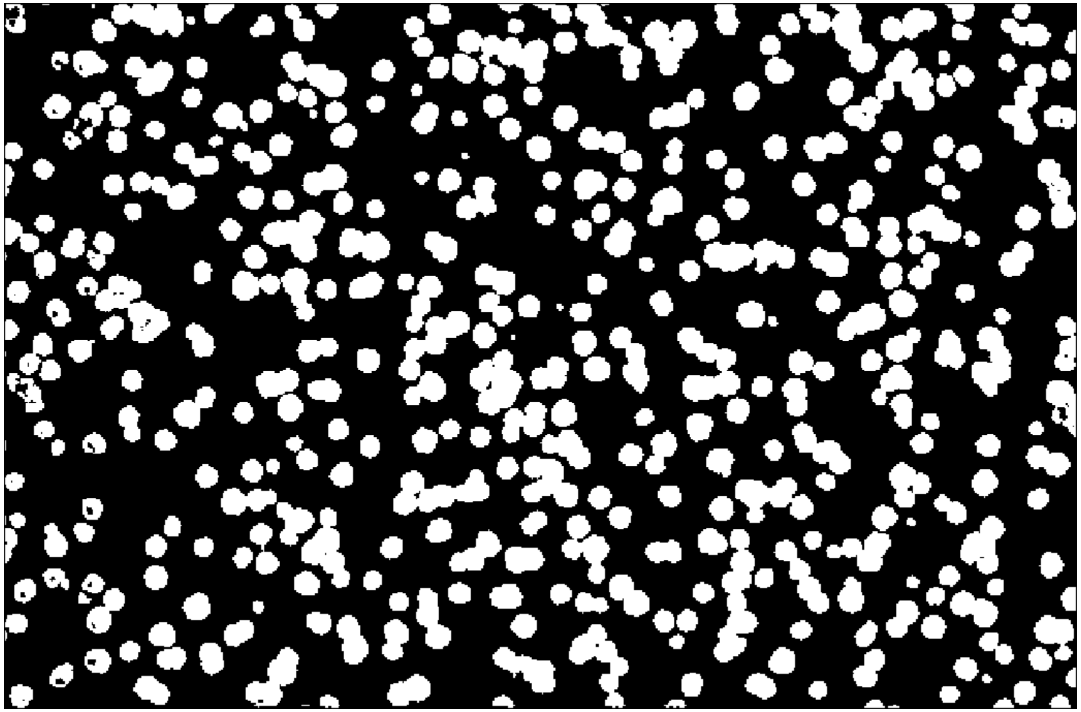
Espace de caractéristiques

Points ≠ pixels

Points = mesure stockée en chaque pixel

Ex : image N&B = intensité (une dimension)





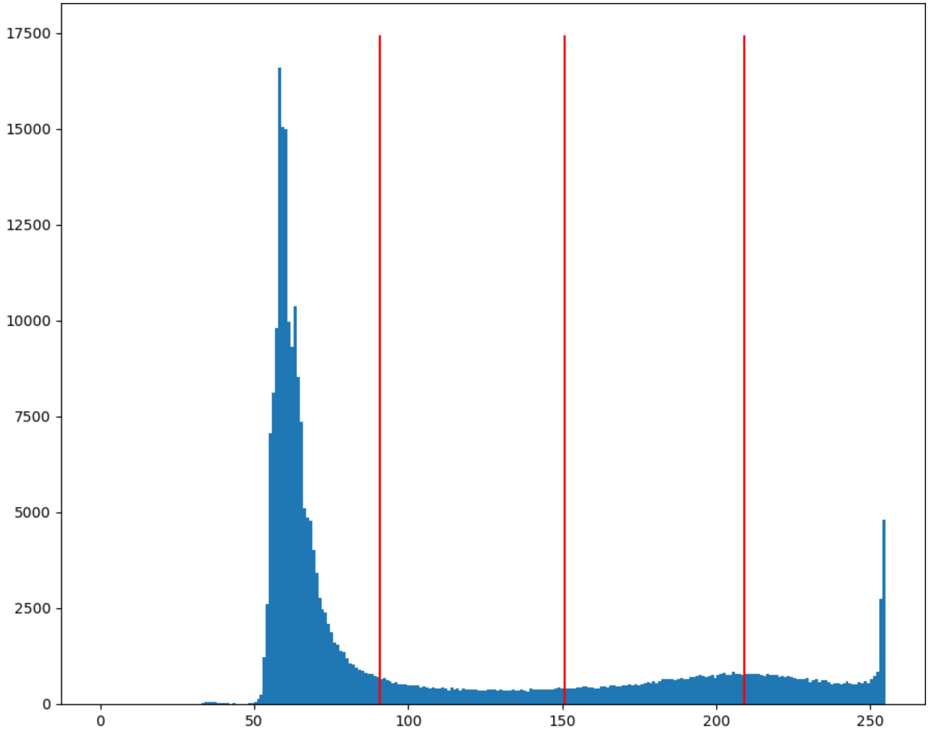
K=2

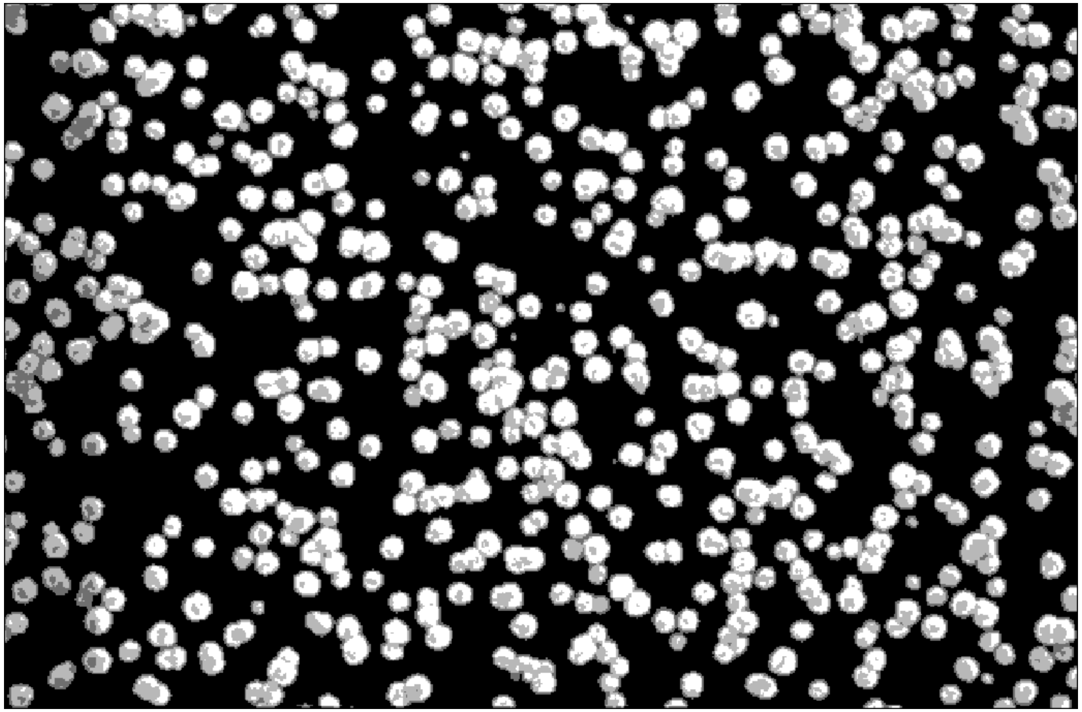
Espace de caractéristiques

Points ≠ pixels

Points = mesure stockée en chaque pixel

Ex : image N&B = intensité (une dimension)





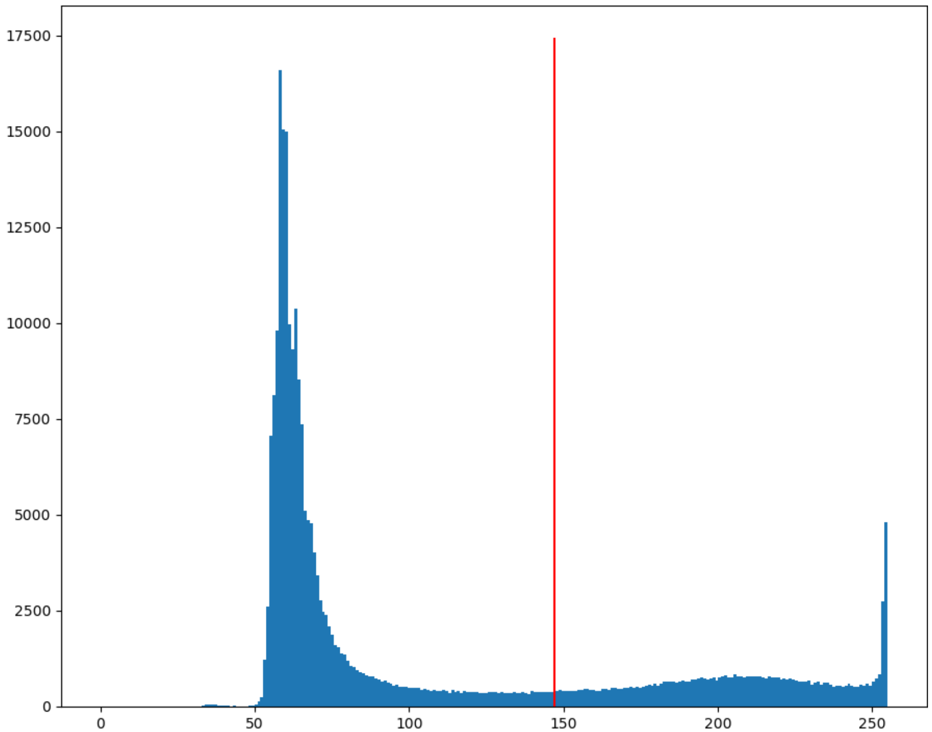
K=4

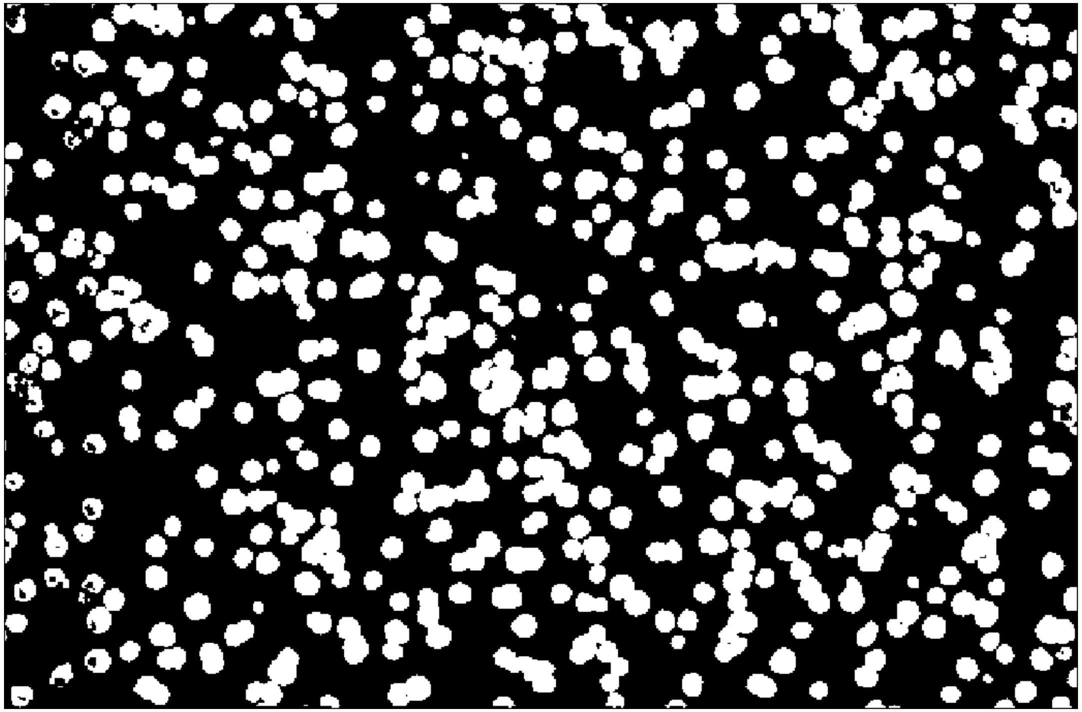
Espace de caractéristiques

Points ≠ pixels

Points = mesure stockée en chaque pixel

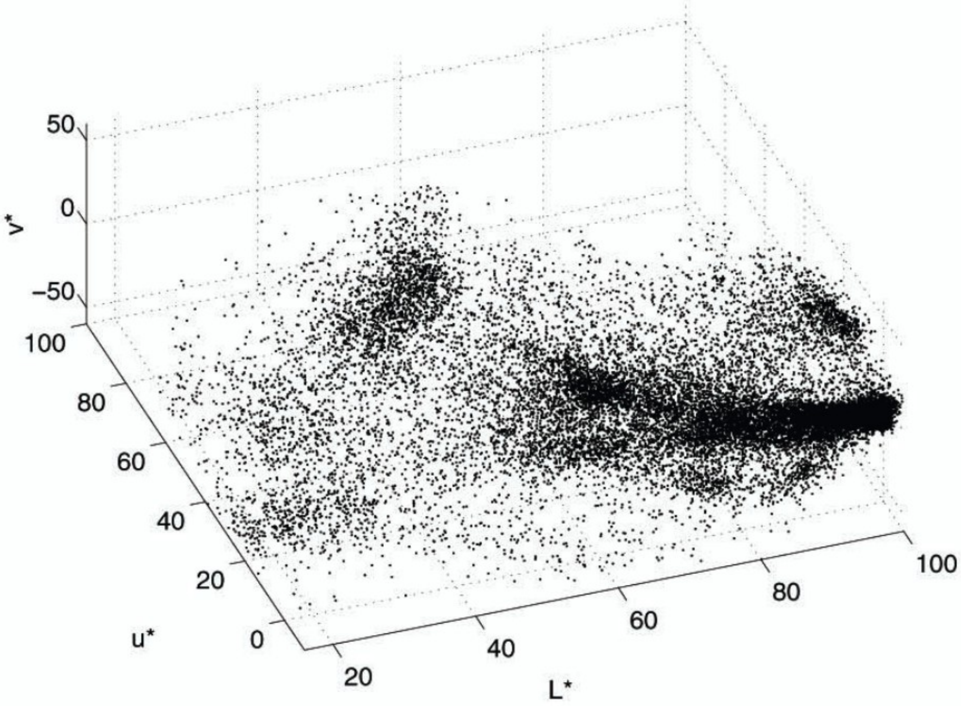
Ex : image N&B = intensité (une dimension)





Mean-shift (K=2)

Espace de caractéristiques : couleurs

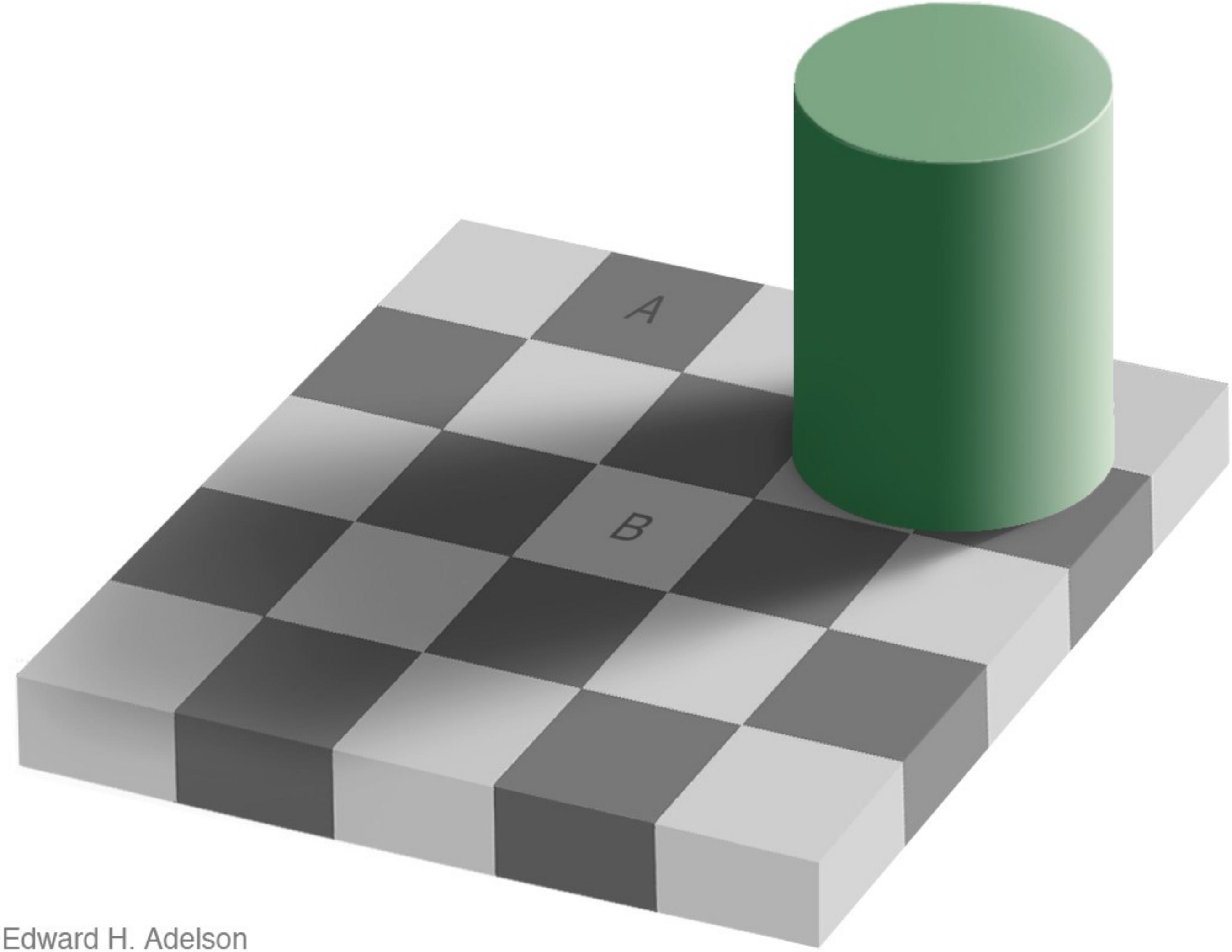


Comaniciu et al. « Mean-Shift: A Robust Approach Toward Feature Space Analysis », IEEE PAMI, 24(5):1-18, 2002.

Espace de caractéristiques : couleurs

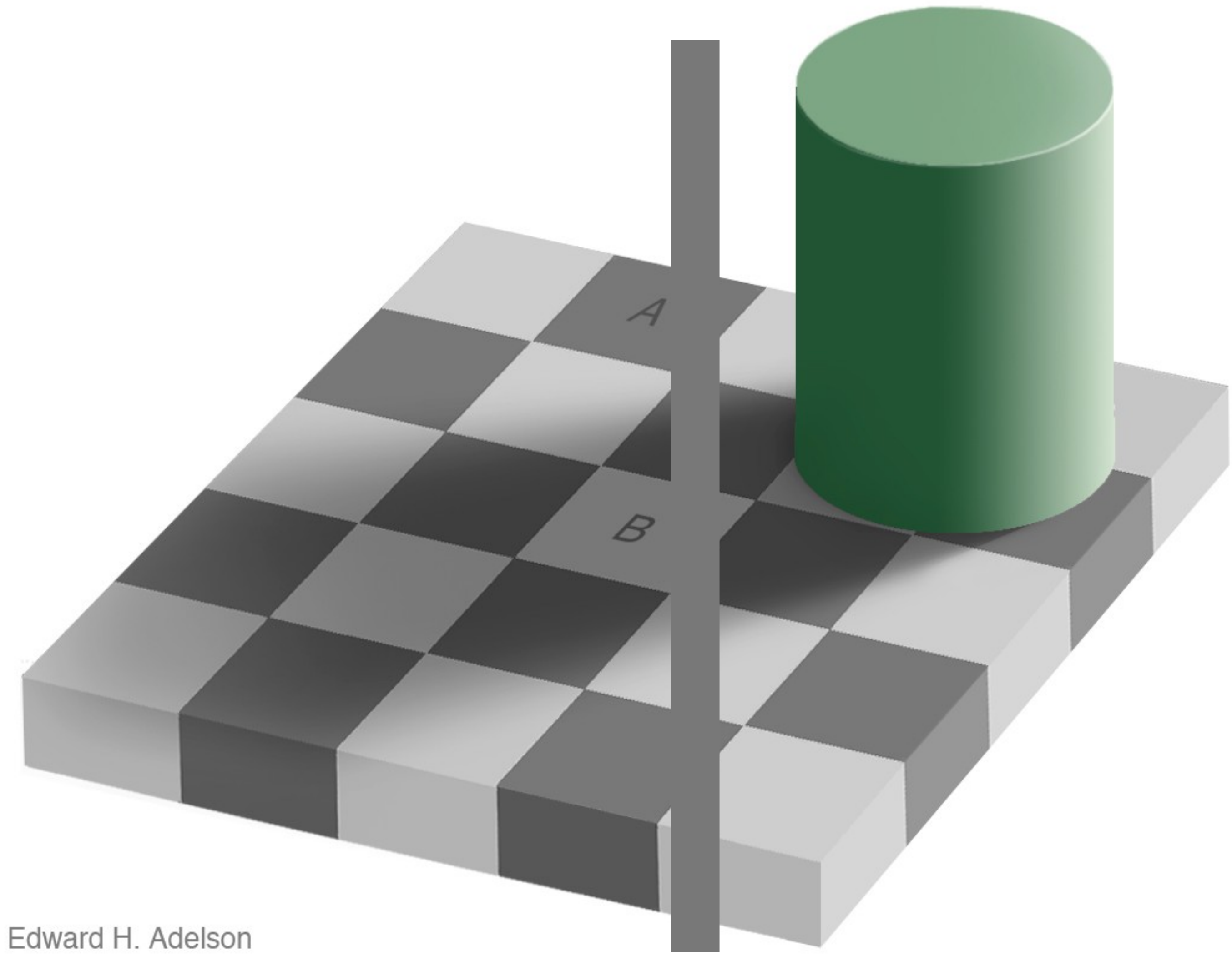
Comaniciu et al. « Mean-Shift: A Robust Approach Toward Feature Space Analysis », IEEE PAMI, 24(5):1-18, 2002.

Limites approche intensité



Edward H. Adelson

Limites approche intensité



Approche contour

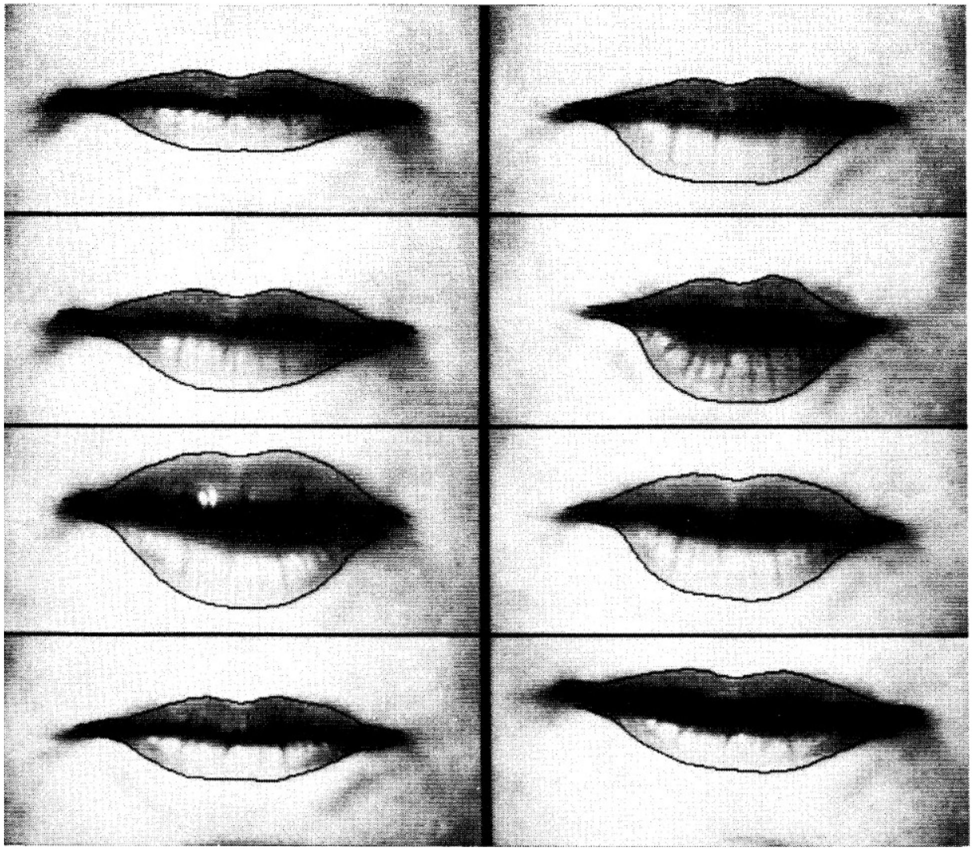
Approche duale : identifier les contours

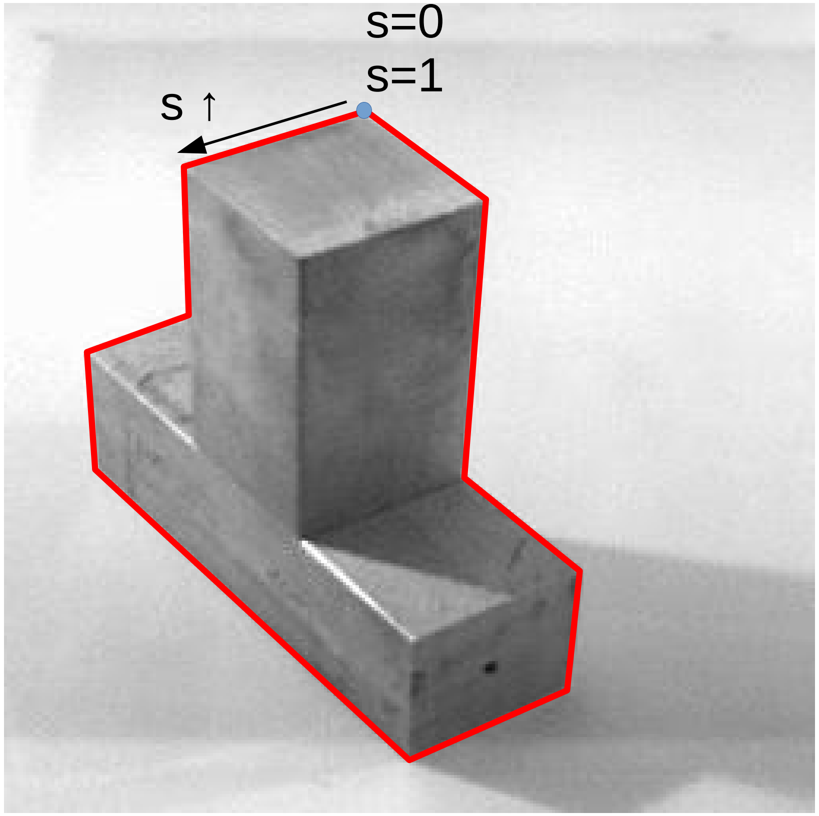
Snakes :

- contour=courbe déformable
- s’adapte aux bords les plus proches de l’image

Snakes : énergie

Contour=courbe paramétrée C(s)





Kass et al « Snakes: Active contour models », IJCV, pp 321-331, 1988

Snakes : énergie

Contour=courbe paramétrée C(s)

Chaque point C(s) soumis à un potentiel → Énergie

$$E_{snake} = \int_0^1 E_{\text{int}}(C(s)) + E_{\text{image}}(C(s))\, ds$$

$$E_{\text{int}}(C(s)) = \frac{1}{2}\left(\alpha(s)|C'(s)|^2 + \beta(s)|C''(s)|^2\right)$$

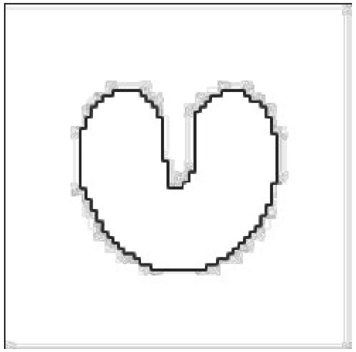
$$\begin{cases} E_{\text{image}}(C(s)) = -|\nabla I(C(s))|^2 \text{(détection de bords)} \\ E_{\text{image}}(C(s)) = I(C(s)) \text{(détection de lignes sombres)} \end{cases}$$

Kass et al « Snakes: Active contour models », IJCV, pp 321-331, 1988

Snakes : énergie externe

$I(x,y)$

$$f(x,y) = -|\nabla I(x,y)|^2$$



$I(x,y)$

$$f(x,y) = I(x,y)$$

Carte des bords de l'image : $E_{\text{image}}(x,y) = f(x,y)$

Kass et al « Snakes: Active contour models », IJCV, pp 321-331, 1988

Snakes : résolution (minimisation E)

Dérivation (équation d'Euler-Lagrange)

$$-\alpha C_{ss} + \beta C_{ssss} + \frac{\partial E_{\text{image}}(C)}{\partial C} = 0$$

Discrétisation

$$AV - F = 0 \Leftrightarrow AV = F$$

Stabilisation

$$(A + \gamma I) V^{t+1} = F + \gamma V^t$$

Kass et al « Snakes: Active contour models », IJCV, pp 321-331, 1988

Snakes : limitations

$$\nabla f(x,y)=\left(\frac{\partial f(x,y)}{\partial x},\frac{\partial f(x,y)}{\partial y}\right)$$

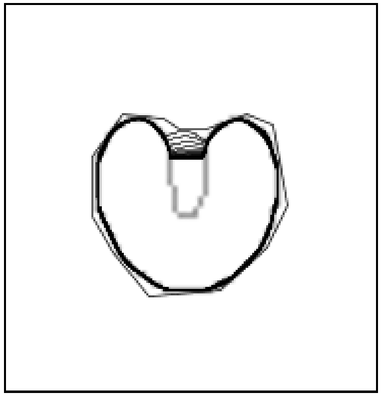
Snakes traditionnels

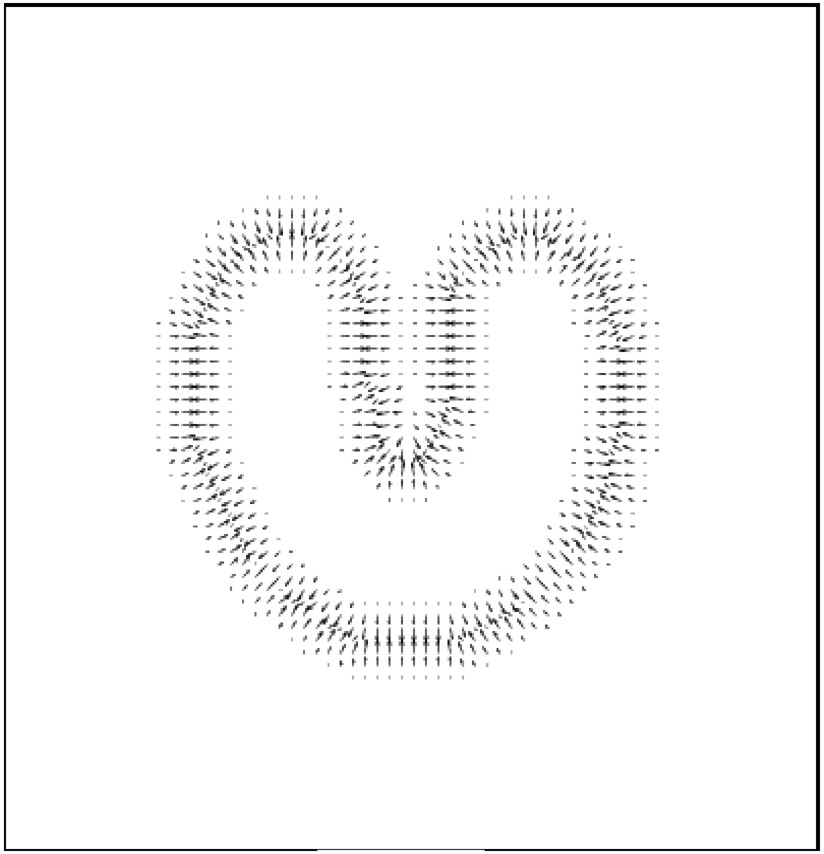
- Sensibles à position initiale
- Difficulté de capturer les zones concaves

$$E_{\text{image}}(C(s))=f(C(s))$$

$$\frac{\partial E_{\text{image}}(C)}{\partial C}=\nabla f(C)$$

$$f(x,y)$$





Xu and Prince « Gradient Vector Flow: A new external force for snakes », CVPR, pp 66-71, 1997

Gradient Vector Flow

Energie externe : remplacer le gradient de f par V

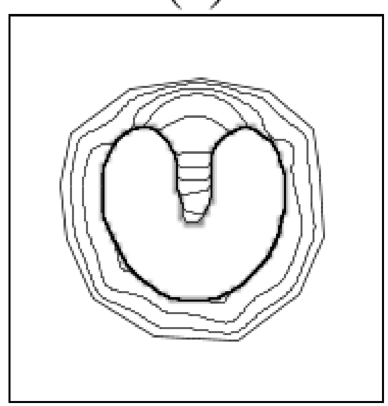
V défini comme minimisant

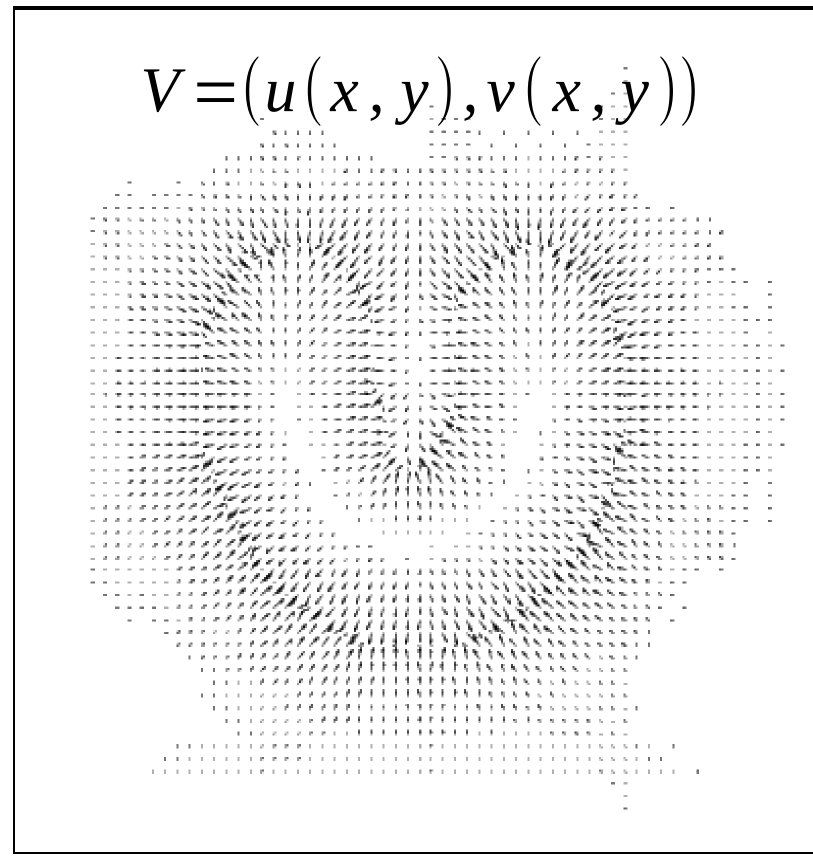
$$E=\iint \mu\left(u_x^2+u_y^2+v_x^2+v_y^2\right)+|\nabla f|^2|V-\nabla f|^2\,dx\,dy$$

1er terme : régularisation

2e terme : V proche de ∇f

(a)





$$V=(u(x,y),v(x,y))$$

Xu and Prince « Gradient Vector Flow: A new external force for snakes », CVPR, pp 66-71, 1997

GVF : résolution

Equations d'Euler-Lagrange

$$\mu \nabla^2 u - (u - f_x)(f_x^2 + f_y^2) = 0$$
$$\mu \nabla^2 v - (v - f_y)(f_x^2 + f_y^2) = 0$$

$(\nabla^2 \text{ est le laplacien})$

Descente de gradient

$$\mu \nabla^2 u - (u - f_x)(f_x^2 + f_y^2) = -\frac{\partial u}{\partial t}$$
$$\mu \nabla^2 v - (v - f_y)(f_x^2 + f_y^2) = -\frac{\partial v}{\partial t}$$

Xu and Prince « Gradient Vector Flow: A new external force for snakes », CVPR, pp 66-71, 1997

Approche contour vs approche région

Contours actifs

- Évolution d'une courbe
- Approche variationnelle → dynamique de la courbe, avec régularisation interne
- Objet = bords = forts gradients
- plus génériquement usage d'un champ énergétique régularisé

Prendre en compte l’espace : approches région

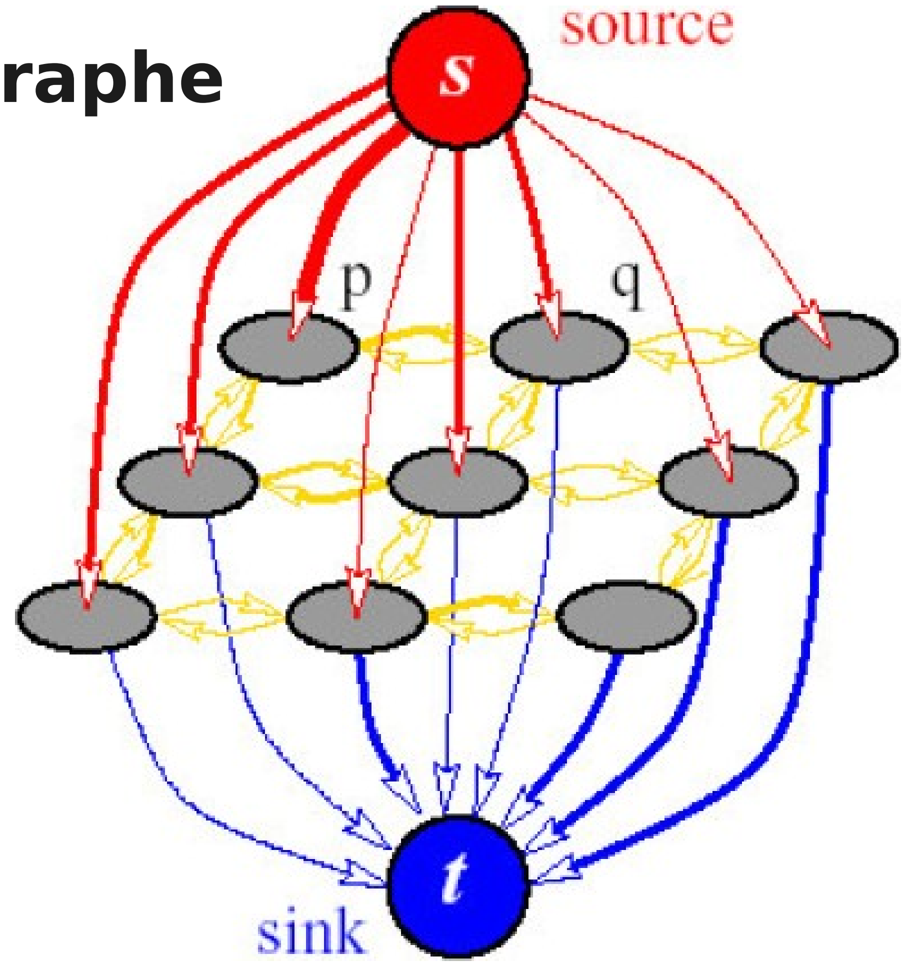
Problème

- Deux pixels proches peuvent avoir deux classes différentes
  - ➔ Segmentation bruitée
- Renforcer l’idée que pixels proches ⇒ même classe
  - ➔ « régularisation spatiale »
- Approche probabiliste :
  - Probabilité que l’intensité soit proche de celle caractéristique de la classe
  - Vs probabilité que deux pixels proches aient le même classe

Graph-cuts

Image représentée par un graphe

- Noeud = pixel
- Arc = voisinage
- Poids =
  - similarité au label (s ou t)
  - ou similarité entre labels voisins



Boykov et al. « Fast approximate energy minimization via graph cuts », IEEE PAMI 23(11):1222-1239, 2001

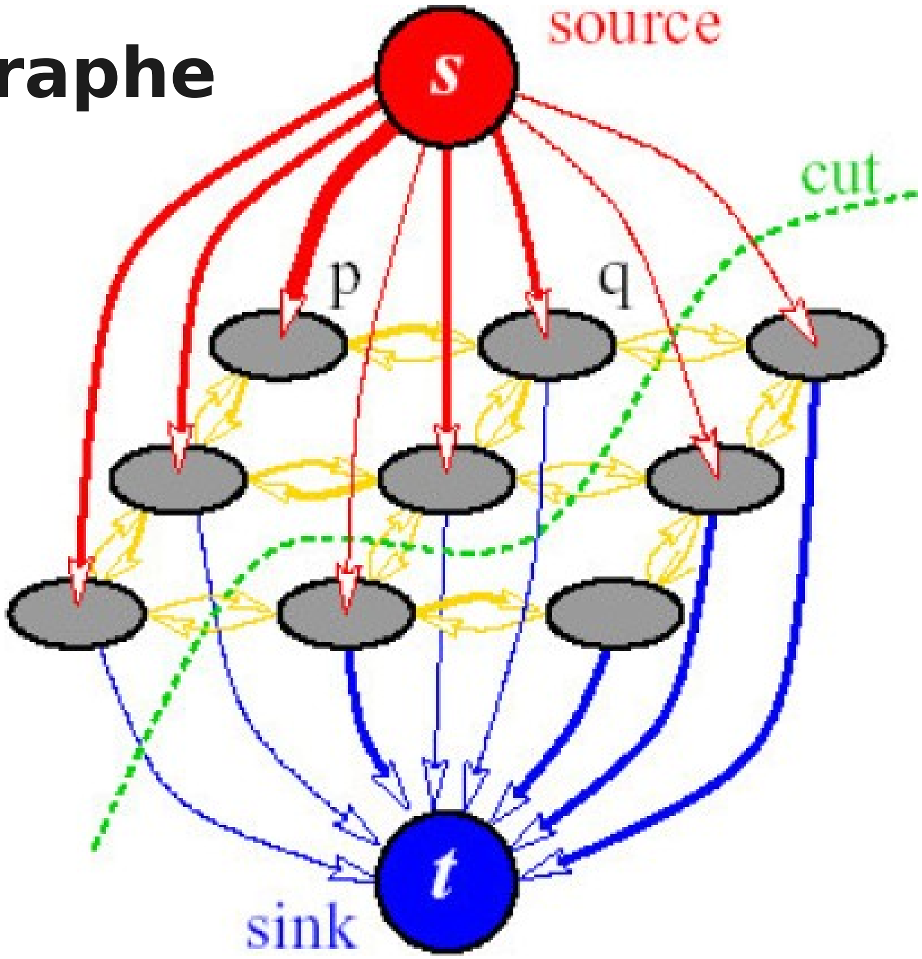
Graph-cuts

Image représentée par un graphe

- Noeud = pixel
- Arc = voisinage
- Poids =
  - similarité au label (s ou t)
  - ou similarité entre labels voisins
- On veut le séparer en 2 graphes
- Coût de la coupe = somme des arcs ôtés

  = somme des similarités
- On cherche une coupe min = flot max

  (Ford-Fulkerson, voir Wikipedia: Maximum Flow Problem)



Boykov et al. « Fast energy minimization via graph cuts », IEEE PAMI 23(11):1222-1239

Modèle de Chan et Vese

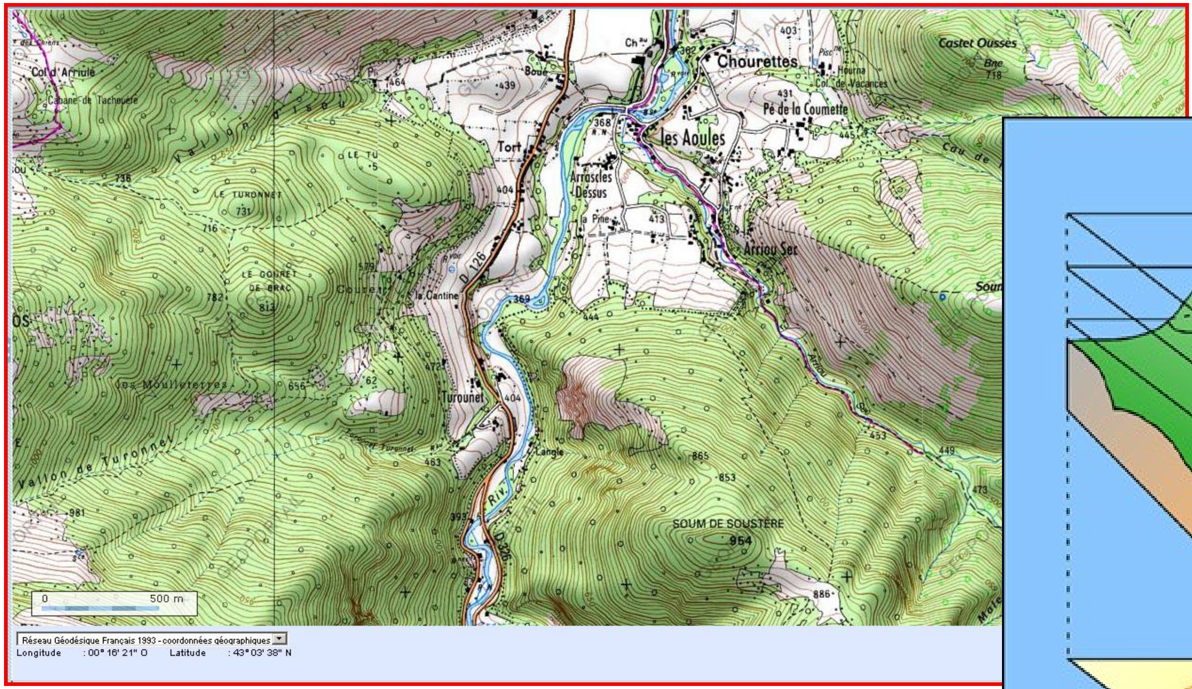
Modèle intensité constante

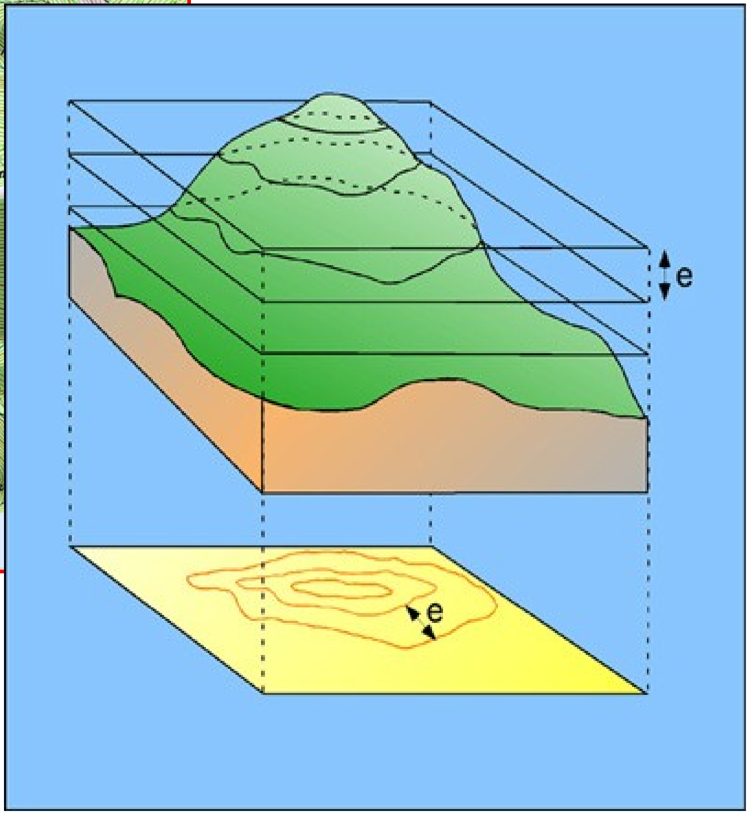
- Un objet d'intensité uniforme c_1
- Un fond d'intensité uniforme c_2
- Régularisation sur la longueur du contour et l'aire de l'objet

$$
\begin{aligned}
E(c_1, c_2, C) = \mu \,\mathrm{Length}(C) + \nu \,\mathrm{Area}(inside(C)) \\
+ \lambda_1 \int_{inside(C)} |I(x,y) - c_1|^2 \, dx\, dy \\
+ \lambda_2 \int_{outside(C)} |I(x,y) - c_2|^2 \, dxdy
\end{aligned}
$$

Chan and Vese « Active contours without edges », IEEE TIP, 10(2):266-277, 2001

Level sets





Level sets

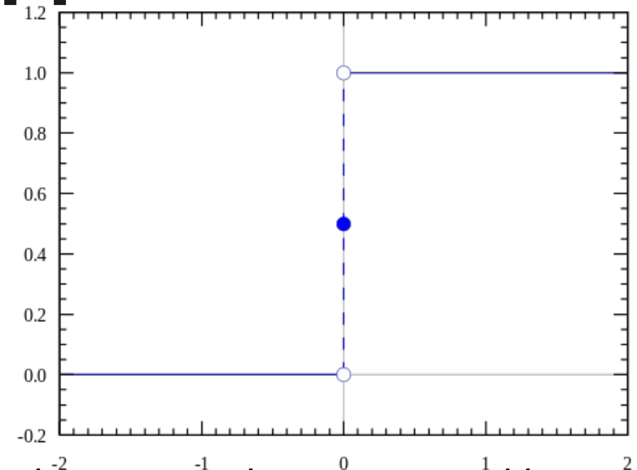
- Bord = isocontour = modèle implicite
- Régularisation = fonction implicite lisse

Reformulation

Emploi d’une fonction implicite

$$\phi(x,y) \text{ telle que } \begin{cases} C=\{(x,y)\in\Omega:\phi(x,y)=0\} \\ inside(C)=\{(x,y)\in\Omega:\phi(x,y)>0\} \\ outside(C)=\{(x,y)\in\Omega:\phi(x,y)<0\} \end{cases}$$

Rappel : fonction Heaviside H (échelon unité)



$$H(s)=\begin{cases} 0 \text{ si } s<0 \\ 1 \text{ si } s>0 \end{cases}$$

$$H'(s)=\delta(s)$$

Chan and Vese « Active contours without edges », IEEE TIP, 10(2):266-277, 2001

Expression formelle

$$E(c_1,c_2,C)=\mu\,\text{Length}(C)+\nu\,\text{Area}(inside(C))+\lambda_1\int_{inside(C)}|I(x,y)-c_1|^2\,dx\,dy$$
$$+\lambda_2\int_{outside(C)}|I(x,y)-c_2|^2\,dxdy$$

$$\text{Length}(C)=\int_\Omega|\nabla H(\phi(x,y))|\,dx\,dy$$

$$\text{Area}(C)=\int_\Omega H(\phi(x,y))\,dx\,dy$$

$$\int_{inside(C)}|I(x,y)-c_1|^2\,dx\,dy=\int_\Omega H(\phi(x,y))|I(x,y)-c_1|^2\,dx\,dy$$

$$\int_{outside(C)}|I(x,y)-c_2|^2\,dx\,dy=\int_\Omega(1-H(\phi(x,y)))|I(x,y)-c_2|^2\,dx\,dy$$

Descente de gradient : $\dfrac{\partial\phi}{\partial t}=-\dfrac{\partial E(c_1,c_2,\phi)}{\partial\phi}$

Euler Lagrange

$$\frac{\partial}{\partial \phi} \mathrm{Area}(C) = \delta(\phi(x,y))$$

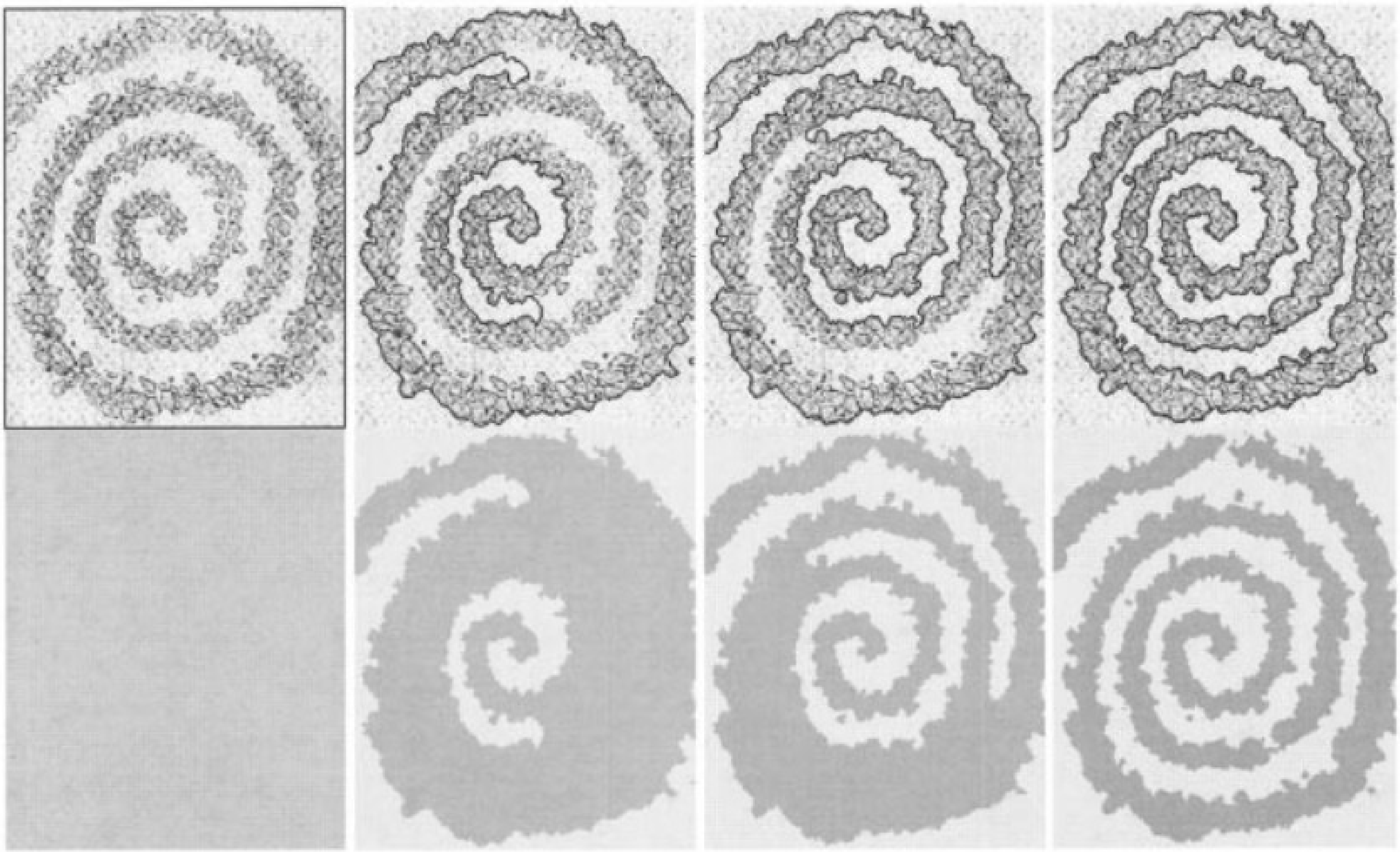
$$\frac{\partial}{\partial \phi} \mathrm{Length}(C) = \delta(\phi) \operatorname{div}\left(\frac{\nabla \phi}{|\nabla \phi|}\right)$$

$$\frac{\partial}{\partial \phi} \int_{inside(C)} |I(x,y) - c_1|^2 dxdy = \delta(\phi) |I(x,y) - c_1|^2$$

$$\frac{\partial}{\partial \phi} \int_{outside(C)} |I(x,y) - c_2|^2 dxdy = -\delta(\phi) |I(x,y) - c_2|^2$$

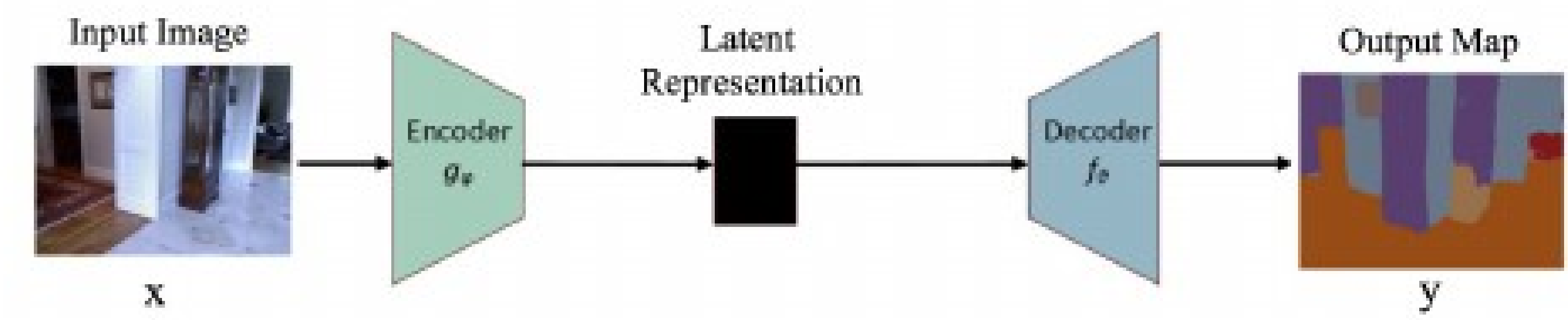
Chan and Vese « Active contours without edges », IEEE TIP, 10(2):266-277, 2001

Exemple



Chan and Vese « Active contours without edges », IEEE TIP, 10(2):266-277, 2001

Architecture encodeur-décodeur



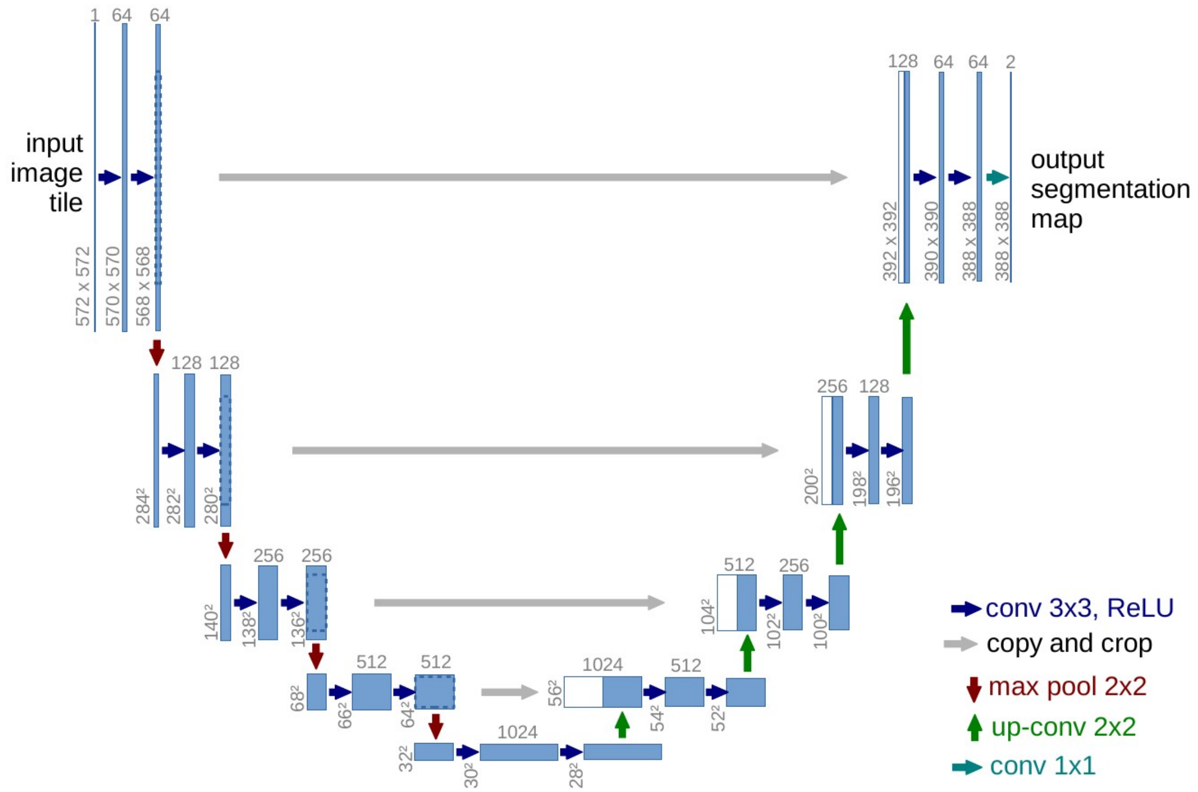
- Objectif : Classification de chaque pixel
- La sortie doit être de même taille que l'entrée
- Encodeur-décodeur : perte de précision
- Régularisation spatiale ?
  - FCN : couche fully connected en fin de réseau vue comme une (grosse) convolution
  - CRF : ajout de contraintes probabilistes sur la co-occurrence de labels à différents niveaux de voisinage
  - U-net, très utilisé en médical mais aussi ailleurs, et SegNet, plus petit

Minaee et al « Image Segmentation Using Deep Learning: A Survey », preprint arXiv:2001.05566, 2020

U-net

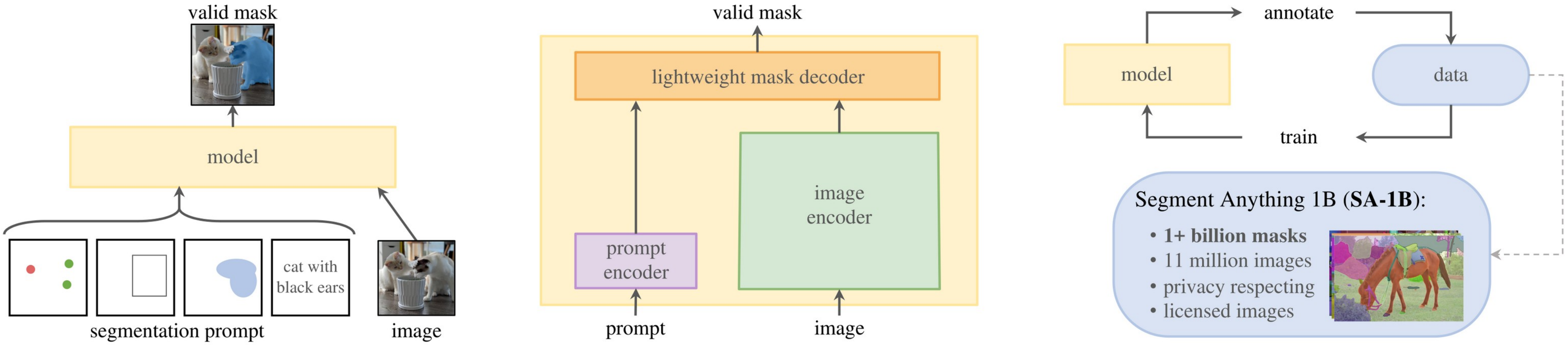
Principe : skip connections

Guider la déconvolution par une information à la bonne résolution



Ronneberger et al « U-Net: Convolutional Networks for Biomedical Image Segmentation », arXiv:1505.04597, 2015

Segment Anything Model (SAM)



(a) **Task**: promptable segmentation (b) **Model**: Segment Anything Model (**SAM**) (c) **Data**: data engine (top) & dataset (bottom)

- Modèle fondamental pour la segmentation (*Foundation Model*)
- 4 tâches : masque, points, boîte, texte
- Entraînement sur SA-1B (11 millions d'images, plus d'1 milliards de masques)
- Annotation : Moteur de données, 3 étapes d'annotation (+ en + auto), de nombreux ré-entraînements
- Génère plusieurs masques avec score de confiance
- Validation sur expériences « zéro-shot »
- https://segment-anything.com (Meta)

En résumé

Deux approches : région vs contour

K-means, mean-shift :

- Simples, mais pas de régularisation spatiale
- Extension en classifiant une carte de caractéristique (ex : texture)

Snakes

- Contour, contrôle local, variations dans objet ou sur contour
- Convergence délicate, pas de changement de topologie
- Extensions nombreuses (GVF, diffusion snakes, Geometric Active Contours…)

Level-sets

- Région, contrôle global (fonction implicite), changement topologie ok
- Schémas numériques complexes (approche variationnelle), gourmands en temps de calcul
- Extensions nombreuses (régularisation, caractéristique image)

Deep learning

- Approches très efficaces, et majoritaires actuellement, même en 3D
- Toujours le problème des données annotées et la question de l'explicabilité