TP3 – Segmentation par la méthode *mean-shift*

La méthode *mean-shift*, qui provient des mathématiques, permet d'estimer les *modes* (maximums locaux) de la fonction de densité d'un nuage de points. Comaniciu et Meer ont montré dans [1, 2] que le *mean-shift* pouvait servir à segmenter une image. Aujourd'hui, cette méthode de segmentation est une des plus souvent citées dans les articles de recherche. Elle présente le gros avantage d'être *non supervisée*, c'est-à-dire qu'il n'est pas utile de préciser à l'avance le nombre de régions recherchées.

Une manière de trouver les modes consiste à suivre la direction du gradient de la fonction de densité. La méthode *mean-shift* utilise une approche non paramétrique, c'est-à-dire qu'elle ne fait pas d'hypothèse sur le modèle des données. Elle effectue une analyse locale par fenêtre, appelée *fenêtre de Parzen*. Pour un échantillon de n observations $\mathbf{x}_i \in \mathbb{R}^d$, $i \in [1, n]$, une estimation de la fonction de densité $\hat{f}(\mathbf{x})$, $\mathbf{x} \in \mathbb{R}^d$, s'écrit :

$$\hat{f}(\mathbf{x}) = \frac{1}{n\,h^d} \sum_{i=1}^{n} K\left(\frac{\mathbf{x}-\mathbf{x}_i}{h}\right) \tag{1}$$

où :

- Le *noyau* K doit être positif, borné, d'intégrale égale à 1 et à décroissance rapide.
- Le paramètre $h > 0$ permet de régler la taille du voisinage pris en compte. Plus h est petit, plus le nombre de modes détectés augmente. Cela permet de « démystifier » la segmentation non supervisée : au lieu de fixer a priori le nombre de régions, on fixe un paramètre dont la valeur influe directement sur ce nombre.

Il est intéressant de choisir le *noyau d'Epanechnikov* (ou noyau « parabolique »), qui est à support borné :

$$K\left(\frac{\mathbf{x}-\mathbf{x}_i}{h}\right) = \begin{cases} \dfrac{d+2}{2\,c_d}\left(1-\dfrac{\|\mathbf{x}-\mathbf{x}_i\|^2}{h^2}\right) & \text{si } \|\mathbf{x}-\mathbf{x}_i\| < h \\ 0 & \text{sinon} \end{cases} \tag{2}$$

où c_d désigne le volume de la sphère unité en dimension d et $\|.\|$ la distance euclidienne. En effet, le gradient de la densité $\hat{f}(\mathbf{x})$ s'écrit alors, d'après (1) :

$$\nabla \hat{f}(\mathbf{x}) = \frac{1}{n\,h^d}\,\frac{d+2}{c_d\,h^2} \sum_{\mathbf{x}_i \in S_h(\mathbf{x})} (\mathbf{x}_i - \mathbf{x}) = \frac{1}{n\,h^d}\,\frac{d+2}{c_d\,h^2}\, n_{\mathbf{x}} \left[-\mathbf{x} + \frac{1}{n_{\mathbf{x}}} \sum_{\mathbf{x}_i \in S_h(\mathbf{x})} \mathbf{x}_i\right] \tag{3}$$

où $S_h(\mathbf{x})$ désigne l'ensemble des observations $\mathbf{x}_i$ se trouvant dans la sphère de rayon h (en dimension d), centrée en $\mathbf{x}$, et $n_{\mathbf{x}} = \operatorname{card}\{S_h(\mathbf{x})\}$. À chaque itération, chaque $\mathbf{x}$ est remplacé par $\mathbf{x} + \delta_{\mathbf{x}}\,\nabla\hat{f}(\mathbf{x})$, où $\delta_{\mathbf{x}}\,\nabla\hat{f}(\mathbf{x})$ constitue un « petit déplacement » dans la direction du gradient, afin de se rapprocher du mode le plus proche. En remplaçant $\delta_{\mathbf{x}}$ par $\left[\frac{1}{n\,h^d}\,\frac{d+2}{c_d\,h^2}\,n_{\mathbf{x}}\right]^{-1}$ dans (3), cette mise à jour revient à remplacer $\mathbf{x}$ par la moyenne $M_h(\mathbf{x})$:

$$M_h(\mathbf{x}) = \frac{1}{n_{\mathbf{x}}} \sum_{\mathbf{x}_i \in S_h(\mathbf{x})} \mathbf{x}_i \tag{4}$$

Voilà pourquoi cette méthode s'appelle *mean-shift*, qui signifie littéralement « remplacement par la moyenne ». On peut donc décrire la méthode *mean-shift* par l'algorithme suivant :

<u>Algorithme d'estimation des modes (*mean-shift*)</u> :
$k \leftarrow 1$
Tant que $k \leqslant k_{\max}$ et qu'il existe $\mathbf{x}$ tel que $\|\mathbf{x} - M_h(\mathbf{x})\| > \epsilon$:
(1) Pour chaque $\mathbf{x}$:
  (a) Calculer la moyenne $M_h(\mathbf{x})$ définie en (4)
  (b) $\mathbf{x} \leftarrow M_h(\mathbf{x})$
(2) $k \leftarrow k+1$

Exercice 1 : segmentation d'une image couleur par *mean-shift*

Pour segmenter une image couleur à l'aide de la méthode *mean-shift*, chaque pixel est assimilé à une observation. Un pixel est caractérisé par sa position et ses trois niveaux de couleur, mais la position et la couleur sont traitées différemment pour déterminer $S_h(\mathbf{x})$. En pratique, seuls sont sélectionnés les pixels $\mathbf{x}_i$ se trouvant à l'intérieur d'une fenêtre de taille $(2\,T+1)\times(2\,T+1)$, centrée en $\mathbf{x}$, qui ont une couleur proche de celle de $\mathbf{x}$, c'est-à-dire tels que $\|I(\mathbf{x}_i)-I(\mathbf{x})\| \leqslant h$. Seule la couleur est modifiée au fil des itérations. Par conséquent, $\mathbf{x}$, $\mathbf{x}_i$ et $M_h(\mathbf{x})$ sont des vecteurs de $\mathbb{R}^3$. L'algorithme dépend en définitive de quatre paramètres :

- T et h : seuil spatial et seuil colorimétrique permettant de définir l'ensemble $S_h(\mathbf{x})$.
- $k_{\max}$ et ϵ : paramètres permettant de contrôler l'arrêt de l'algorithme.

En vous aidant de cette description, complétez la fonction `meanshift.m`, puis lancez le script `exercice_1.m`. Observez l'influence de chaque paramètre. Lorsque le résultat vous semble correct, modifiez la valeur de la variable `reduction`. Réglez la valeur de chaque paramètre de manière à obtenir les résultats les plus pertinents, d'un point de vue qualitatif.

Écrivez une fonction `comptage` qui compte le nombre de régions (connexes) de l'image segmentée. Affichez ce nombre au-dessus de l'image segmentée (`title`).

Exercice 2 : fusion des régions similaires (exercice facultatif)

Vous constatez, par exemple pour `piments.png`, que l'image est généralement « sur-segmentée », et ce quel que soit le choix des paramètres. Comaniciu et Meer préconisent de corriger ce défaut de la façon suivante : si la distance entre deux modes est inférieure à un seuil, alors ces deux modes doivent être fusionnés (il est donc nécessaire d'introduire un nouveau paramètre). Dupliquez le fichier `exercice_1.m` sous le nom `exercice_2.m`, que vous modifierez de manière à tenir compte de cette amélioration.

Description succincte du TP4 de segmentation

Les trois premiers TP de segmentation vous ont permis de découvrir trois méthodes de segmentation par classification : classification *supervisée* (champs de Markov), classification *semi-supervisée* (algorithme EM) et classification *non supervisée* (méthode *mean-shift*). Plutôt que de vous présenter une nouvelle méthode de segmentation dans le prochain TP, nous vous demanderons d'utiliser la méthode *mean-shift* pour effectuer le suivi d'un objet (*tracking*) dans une séquence vidéo. En suivant le lien ci-après, vous pourrez vous faire une idée du genre de résultats que l'on peut espérer atteindre :

https://www.irit.fr/~Alain.Crouzil/jaffre/RECHERCHE/dea.html

Et si vous souhaitez commencer à comprendre comment cela fonctionne, vous pouvez lire l'article suivant :

https://www.irit.fr/~Alain.Crouzil/jaffre/PUBLICATIONS/orasis2003_jaffre_crouzil.pdf

sachant que ce quatrième et dernier TP de segmentation sera beaucoup moins guidé que les trois premiers...

Références

[1] D. Comaniciu et P. Meer : Mean shift analysis and applications. *In IEEE International Conference on Computer Vision*, volume 2, pages 1197–1203, 1999.

[2] D. Comaniciu et P. Meer : Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.