TP7 – Segmentation par *mean-shift*

La méthode *mean-shift* est un outil mathématique qui permet d'estimer les *modes* (maximums locaux) de la fonction de densité d'un nuage de points. Comaniciu et Meer ont montré, dans un article paru en 1999, intitulé *Mean shift analysis and applications*, que le *mean-shift* pouvait servir à segmenter une image. Aujourd'hui, cette méthode de segmentation fait partie des méthodes les plus souvent citées. Elle présente l'avantage d'être *totalement non supervisée*, c'est-à-dire qu'il n'est pas nécessaire de fixer par avance le nombre de régions.

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 $\widehat{f}(\mathbf{x})$, $\mathbf{x} \in \mathbb{R}^d$, s'écrit :

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

où :

- Le *noyau* K est une fonction $\mathbb{R}^d \to \mathbb{R}$ positive, 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 le nombre de régions a priori, on fixe un paramètre dont la valeur influe directement sur ce nombre.

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

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

où V_d désigne le volume de la sphère unité en dimension d. De (1) et (2), on tire l'expression suivante de $\nabla\widehat{f}(\mathbf{x})$:

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

où $S_h(\mathbf{x})$ désigne l'ensemble des observations $\mathbf{x}_i$ qui se trouvent dans la sphère centrée en $\mathbf{x}$, de rayon h (en dimension d), et $n_{\mathbf{x}} = \operatorname{card}\{S_h(\mathbf{x})\}$. L'expression (3) fait apparaître le *vecteur mean-shift*, défini comme suit :

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

Ce vecteur joint le point $\mathbf{x}$ à la moyenne des observations situées dans $S_h(\mathbf{x})$. Nous disposons donc d'une estimation de $\nabla\widehat{f}(\mathbf{x})$, sans qu'il soit nécessaire d'estimer *explicitement* la fonction $\widehat{f}$. En effet, l'expression (3) de $\nabla\widehat{f}(\mathbf{x})$ se déduit simplement du calcul de la moyenne des observations situées dans $S_h(\mathbf{x})$. Cette propriété donne tout son intérêt à la méthode *mean-shift*.

Pour trouver le mode le plus proche d'une observation $\mathbf{x}_0$, il suffit de suivre la direction du gradient de $\widehat{f}(\mathbf{x})$. Au lieu d'une « descente de gradient » (méthode de minimisation, cf. le TP10 de l'UE TAV), il faut effectuer ici une « montée de gradient » (méthode de maximisation). En partant de $\mathbf{x}^{(0)} = \mathbf{x}_0$, le mode le plus proche de $\mathbf{x}_0$ peut donc être trouvé par l'itération suivante :

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + M_h(\mathbf{x}^{(k)}) = \frac{1}{n_{\mathbf{x}^{(k)}}} \sum_{\mathbf{x}_i \in S_h(\mathbf{x}^{(k)})} \mathbf{x}_i \tag{5}$$

Exercice 1 : partition d'un nuage de points de $\mathbb{R}^2$ par *mean-shift*

Le script `exercice_0.m` affiche la superposition de $N = 3$ nuages de points de $\mathbb{R}^2$, tirés aléatoirement selon trois lois normales (fonction `randn`). Il affiche ensuite le chemin qui mène d'un point de $\mathbb{R}^2$, sélectionné par l'utilisateur par un clic de souris, au mode le plus proche de la fonction de densité, *sans estimer* cette fonction, à l'aide de la méthode *mean-shift* (fonction `meanshift`).

Après avoir testé ce script, faites-en une copie, de nom `exercice_1.m`, que vous modifierez de manière à afficher une partition du nuage, que vous matérialiserez par la couleur d'affichage (utilisez pour cela la liste de couleurs définies dans la variable `couleurs`).

Exercice 2 : 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 pour déterminer $S_h(\mathbf{x})$, la position et la couleur sont traitées différemment ($\mathbf{x}$, $\mathbf{x}_i$ et $M_h(\mathbf{x})$ sont des vecteurs de $\mathbb{R}^3$). 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 $\mathbf{x}$, et ayant une couleur proche de celle de $\mathbf{x}$, c'est-à-dire tels que $\|I(\mathbf{x}_i) - I(\mathbf{x})\|^2 \leqslant h^2$. À chaque itération, la couleur d'un pixel $\mathbf{x}$ est remplacée par la moyenne des couleurs des pixels de $S_h(\mathbf{x})$. La méthode *mean-shift* permettant de segmenter une image couleur dépend donc 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'itération.

Faites une copie de la fonction `meanshift`, de nom `meanshift_2`, que vous modifierez de manière à segmenter les images couleur lues par le script `exercice_2.m`, en vous aidant de cette description. 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. Le second paramètre de sortie de la fonction `meanshift_2` est le nombre `C` de classes de la partition, qui est affiché par `exercice_2.m` grâce aux fonctions `title` et `num2str`.