

TP4 – Détection d’objets dans une image

L’objectif de ce TP est de détecter et de compter les individus d’une colonie de flamants roses à partir d’une image aérienne (cf. figure 1). Ce comptage, qui permet aux écologues d’étudier l’évolution des populations de flamants roses, se faisait « à la main » jusque dans les années 2000. Une idée simple consiste à positionner N disques dans l’image de telle sorte que le niveau de gris moyen à l’intérieur des disques soit le plus élevé possible. Tous les disques ont le même rayon R . Les coordonnées (réelles) des centres des disques sont tirées selon une loi uniforme. Une fois les N disques positionnés dans l’image, on tente de **maximiser** la somme des niveaux de gris moyens à l’intérieur des disques en procédant comme suit : pour chaque disque traité à tour de rôle, on tire de nouvelles coordonnées ; on accepte la proposition si elle fait croître le niveau de gris moyen. Lancez le script `exercice_0` qui vous est fourni. Quel est manifestement son défaut vis-à-vis de la détection ?



FIGURE 1 – Image aérienne d’une colonie de flamants roses.

Exercice 1 : détection de flamants roses par MAP

Ce problème permet d’illustrer une autre application des champs de Markov dans laquelle, contrairement à ce qui a été vu jusqu’à présent, les variables aléatoires du champ de Markov ne sont pas associées aux pixels de l’image, mais aux N disques positionnés dans l’image. Une réalisation de la variable aléatoire \mathbf{C}_i , $i \in \{1, \dots, N\}$, associée au $i^{\text{ème}}$ disque, est la position $\mathbf{c}_i = (x_i, y_i) \in \mathbb{R}^2$ de son centre, son rayon R étant fixe.

Comme pour le script `exercice_0`, le terme d’attache aux données de l’énergie, qui correspond à la vraisemblance, est égal à l’opposé du niveau de gris moyen $\bar{I}(\mathbf{c}_i)$ à l’intérieur du disque de rayon R centré en \mathbf{c}_i . Cela correspond à l’intuition suivante : la probabilité qu’un flamant rose se trouve en \mathbf{c}_i est d’autant plus élevée que l’image est localement plus claire.

Par ailleurs, s’il est pertinent de modéliser un flamant rose par un disque, on ne peut autoriser n’importe quel recouvrement entre disques. Nous introduisons donc un a priori global sur une réalisation $\mathbf{c} = (\mathbf{c}_i)_{i \in \{1, \dots, N\}}$ du champ de Markov $\mathbf{C} = (\mathbf{C}_i)_{i \in \{1, \dots, N\}}$, qui pénalise toutes les paires de disques dont les centres sont trop proches (les flamants ne pouvant être empilés les uns sur les autres!). L’énergie correspondant à la probabilité a posteriori du modèle complet s’écrit donc ($\beta > 0$ est un paramètre) :

$$U(\mathbf{c}) = - \sum_{1 \leq i \leq N} \bar{I}(\mathbf{c}_i) + \beta \sum_{1 \leq i < j \leq N} \delta \left(\|\mathbf{c}_j - \mathbf{c}_i\| \leq \sqrt{2} R \right) \quad (1)$$

Faites une copie du script `exercice_0`, de nom `exercice_1`, que vous modifierez de manière à trouver la configuration qui minimise cette énergie, en supposant que $\beta \rightarrow +\infty$. Il est donc inutile de calculer explicitement l’énergie $U(\mathbf{c})$. En revanche, vous devez vérifier, pour tout nouveau disque \mathbf{c}_i tiré aléatoirement, que les inégalités $\|\mathbf{c}_j - \mathbf{c}_i\| \leq \sqrt{2} R$ sont fausses, $\forall j \in \{1, \dots, N\} \setminus \{i\}$, **y compris lors du tirage de la configuration initiale.**

S'il permet effectivement de localiser les N taches les plus claires de l'image ne présentant pas de fort recouvrement, vous constatez en revanche que ce script n'est pas adapté au *comptage* des flamants roses. La notion de *processus ponctuel marqué* permet de pallier ce défaut.

Processus ponctuel marqué

Contrairement à un champ de Markov, qui est un champ aléatoire, un *processus ponctuel marqué* est un *processus aléatoire*, pour lequel la dimension temporelle entre en jeu. L'énergie à minimiser s'écrit sous une forme très similaire à (1) :

$$U(\mathbf{c}) = \sum_{1 \leq i \leq N} U_i(\mathbf{c}_i) + \beta \sum_{1 \leq i < j \leq N} \delta \left(\|\mathbf{c}_j - \mathbf{c}_i\| \leq \sqrt{2} R \right) \quad (2)$$

mais le nombre N de disques, c'est-à-dire de variables aléatoires, peut désormais varier au cours du temps. Pour une configuration $\mathbf{c} = (\mathbf{c}_i)_{i \in \{1, \dots, N\}}$ à N individus, l'espace des configurations est \mathbb{R}^{2N} . Par conséquent, l'espace complet des configurations, dans lequel le processus ponctuel marqué est défini, est $\emptyset \cup \bigcup_{N=1}^{+\infty} \mathbb{R}^{2N}$.

D'autre part, l'énergie individuelle $U_i(\mathbf{c}_i)$ d'un disque ne s'écrit plus $-\bar{I}(\mathbf{c}_i)$, comme dans le modèle (1), car cette énergie doit vérifier les trois contraintes suivantes :

- Elle doit décroître lorsque le niveau de gris moyen $\bar{I}(\mathbf{c}_i)$ croît.
- Elle doit être négative pour un disque \mathbf{c}_i bien positionné, sans quoi la suppression d'un disque bien positionné pourrait faire décroître l'expression (2) de $U(\mathbf{c})$, ce qui serait absurde.
- Elle doit être positive pour un disque \mathbf{c}_i mal positionné, sans quoi l'ajout d'un disque mal positionné pourrait faire décroître $U(\mathbf{c})$, ce qui serait tout aussi absurde.

Une façon de respecter ces trois contraintes consiste à utiliser une *fonction sigmoïde* :

$$U_i(\mathbf{c}_i) = 1 - \frac{2}{1 + \exp \left\{ -\gamma \left[\frac{\bar{I}(\mathbf{c}_i)}{S} - 1 \right] \right\}} \quad (3)$$

où S est un seuil indiquant à partir de quelle valeur du niveau de gris moyen $\bar{I}(\mathbf{c}_i)$ le disque \mathbf{c}_i peut être considéré comme « bien positionné ». Un exemple de sigmoïde du type (3) est représenté sur le graphe de la figure 2.

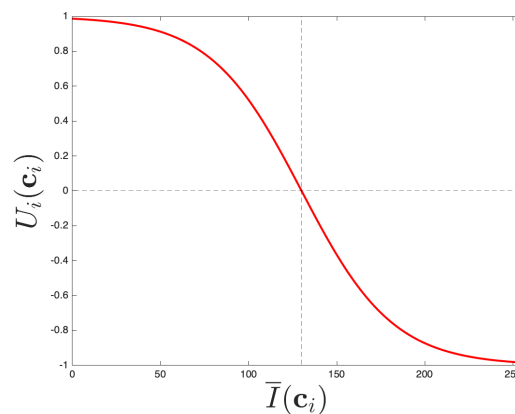


FIGURE 2 – Graphe de la sigmoïde (3), pour $S = 130$ et $\gamma = 5$: $U_i(\mathbf{c}_i)$ change de signe en $\bar{I}(\mathbf{c}_i) = S$; la dérivée en ce point vaut $-\gamma/2$.

Optimisation par naissances/morts multiples

Pour minimiser l'énergie $U(\mathbf{c})$ définie par (2) et (3), vous allez utiliser un algorithme de *naissances/morts multiples*. Le principe de cet algorithme est d'alterner les phases de naissances, où de nouveaux disques sont ajoutés aléatoirement à la configuration courante, et les phases de morts, où les disques les moins pertinents, au sens de l'énergie, sont supprimés. Cette dynamique, intégrée dans un schéma de recuit simulé, converge vers le minimum global de l'énergie. La configuration initiale est l'ensemble vide. L'algorithme consiste à initialiser la température $T \leftarrow T_0$ et le « taux de naissances » $\lambda \leftarrow \lambda_0$, puis à itérer la séquence suivante :

1. **Naissances.** Faire un tirage aléatoire du nombre \tilde{N} de nouveaux disques, suivant une *loi de Poisson* de moyenne λ (utiliser la fonction `poissrnd` de Matlab) :

$$p(\tilde{N}) = e^{-\lambda} \frac{\lambda^{\tilde{N}}}{\tilde{N}!} \quad (4)$$

Ces \tilde{N} nouveaux disques, qui sont positionnés dans l'image de façon aléatoire, suivant une loi uniforme, sont ajoutés à la configuration courante.

2. **Tri des disques.** Une fois la phase de naissances terminée, calculer l'énergie individuelle $U_i(\mathbf{c}_i)$ de chaque disque \mathbf{c}_i de la configuration courante \mathbf{c} , puis classer les disques par valeurs décroissantes de $U_i(\mathbf{c}_i)$.
3. **Morts.** Pour chaque disque \mathbf{c}_i de la configuration courante \mathbf{c} , lue dans cet ordre, calculer la probabilité :

$$p(\mathbf{c}_i|\mathbf{c}) = \frac{\lambda}{\lambda + \exp \left\{ \frac{U(\mathbf{c} \setminus \{\mathbf{c}_i\}) - U(\mathbf{c})}{T} \right\}} \quad (5)$$

et supprimer \mathbf{c}_i , c'est-à-dire faire $\mathbf{c} \leftarrow \mathbf{c} \setminus \{\mathbf{c}_i\}$ avec cette probabilité.

4. **Test de convergence.** Si le processus n'a pas convergé, c'est-à-dire si la configuration après l'étape de morts a changé par rapport à l'itération précédente, modifier la température T et le taux de naissances λ en les multipliant par un même coefficient de décroissance $0 < \alpha < 1$. Retourner à l'étape de naissances.

Exercice 2 : détection de flamants roses par processus ponctuel marqué

Faites une copie du script `exercice_1`, de nom `exercice_2`, que vous modifierez de manière à détecter les flamants roses en utilisant l'algorithme ci-dessus, avec les valeurs suivantes des paramètres du modèle : $\beta = 1$, $S = 140$, $\gamma = 5$, $T_0 = 0.1$, $\lambda_0 = 100$, $\alpha = 0.99$. La condition d'arrêt de l'algorithme (étape 4) pourra être simplifiée, en fixant par exemple le nombre d'itérations à `q_max = 200`. Par ailleurs, le graphique de droite (`subplot(1,2,2)`) devra montrer l'évolution de l'énergie $U(\mathbf{c})$ de la configuration courante en fonction du nombre q d'itérations (l'échelle de cette courbe pourra être fixée à `[0 q_max -400 0]`).

Jouez sur les paramètres de manière à compter « au plus juste » le nombre N_{lim} d'individus de l'image `colonie.png`, puis relancez le script `exercice_1` en fixant N à la valeur N_{lim} . Comparez les résultats.

Exercice 3 : modification de la marque (facultatif)

La détection de flamants roses par le script `exercice_2` peut être améliorée en modifiant la « marque » : il semble en effet plus pertinent de modéliser un flamant rose par une ellipse que par un disque. Même en fixant le demi-grand axe a à R et le demi-petit axe b à $3/4 R$, une variable supplémentaire apparaît, à savoir l'angle polaire du grand axe $\theta \in [0, \pi]$. Mais la véritable difficulté de cette modification consiste à calculer l'aire de l'intersection entre deux ellipses, pour laquelle il n'existe pas d'expression analytique dans le cas général. Il est donc pertinent d'utiliser une approche probabiliste pour effectuer ce calcul.

Rappel – Pour calculer l’aire d’un quart de disque de rayon $R = 1$, dont on sait qu’elle vaut $\pi R^2/4 = \pi/4$, une approche probabiliste consiste à tirer aléatoirement N points (x_i, y_i) à l’intérieur du carré $[0, 1] \times [0, 1]$, selon une loi uniforme, et de comptabiliser la proportion de points tels que $x_i^2 + y_i^2 \leq 1$. Il est facile de vérifier que, plus le nombre N de points est élevé, plus cette proportion s’approche effectivement de $\pi/4$.

Cette technique est très commode pour estimer l’aire de l’intersection de deux figures géométriques quelconques, du moment que l’on dispose de leurs équations cartésiennes. Or, un point (x, y) se trouve à l’intérieur d’une ellipse centrée en (x_0, y_0) , de demi-axes (a, b) et d’angle polaire θ , si la condition suivante est vérifiée :

$$\frac{[(x - x_0) \cos \theta + (y - y_0) \sin \theta]^2}{a^2} + \frac{[(x - x_0) \sin \theta - (y - y_0) \cos \theta]^2}{b^2} \leq 1 \quad (6)$$

Pour résoudre l’exercice 3, il est donc conseillé de commencer par écrire une nouvelle version du script `exercice_2`, dans laquelle vous vous inspirerez de cette approche probabiliste pour calculer, d’une part, le niveau de gris moyen à l’intérieur d’un disque, et d’autre part, le nombre de disques dont l’intersection avec un disque donné est supérieure à un seuil. Plus précisément, il est recommandé de tirer aléatoirement une liste de points situés à l’intérieur d’un disque de rayon R , centré à l’origine :

```
P_test = R*(2*rand(N_test,2)-1);
interieur = (P_test(:,1).^2+P_test(:,2).^2<=R^2);
P_test = P_test(interieur,:);
```

Lorsque `N_test = 1000`, le nombre de ces points est statistiquement égal à $1000 \pi/4 \approx 785$, ce qui rend les estimations probabilistes escomptées suffisamment précises. L’intersection entre deux disques sera considérée comme « trop élevée » si plus de 20% des points situés à l’intérieur de l’un des disques se situent également à l’intérieur de l’autre disque.

Une fois cette nouvelle version du script `exercice_2` mise au point, faites-en une copie, de nom `exercice_3`, qu’il vous sera facile de modifier de manière à prendre en compte la nouvelle marque, à savoir une ellipse de demi-grand axe $a = R$ et de demi-petit axe $b = 3/4 R$.