TP2: Filtrage d'images et apprentissage profond

GBM8770 – Automne 2024

Professeure: Farida Cheriet

Chargés de laboratoire : Zacharie Legault et Emmanuelle Richer

---

Objectifs : Ce laboratoire se compose de deux parties. La première partie porte sur le filtrage spectral des images et la deuxième partie sur l'utilisation de l'apprentissage profond sur des images médicales.

Remise du travail : La date de remise est le 7 novembre à 23h30. Une pénalité de 3 points par jour sera appliquée lors d'un retard.

Documents à remettre : Le code et les réponses aux questions sont à compléter dans les notebooks IPython fournis avec cet énoncé (`tp2_seance1.ipynb` et `tp2_seance2.ipynb`).

Le dossier contenant le notebook et toutes les ressources nécessaires à son exécution sont à remettre sous la forme d'une archive ZIP nommée

`GBM8770_TP2_équipe_Nom_matricule_Nom_matricule`

(par exemple : `GBM8770_TP2_0_Richer_1234567_Legault_7654321`).

Commentez votre code ! Lorsque votre code (et donc vos résultats) est incorrect, vos commentaires peuvent nous permettre de valoriser votre approche...

> **!** Attention ! Nous avons toléré certaines mauvaises pratiques dans le TP1 qui ne seront pas acceptées à l'avenir (et nous nous permettrons d'enlever des points).
>
> - N'utilisez pas de double boucle `for` pour itérer sur les pixels d'une image : utilisez les fonctions vectorisées de `numpy`.
> - Lorsque vous affichez une image en tons de gris, spécifiez une colormap appropriée (typiquement `cmap="gray"`).

Séance I.
Filtrage fréquentiel 2D

> **i** Nous vous conseillons d'utiliser la plateforme de traitement d'image présentée par Clément Playout dans le cours, particulièrement le module sur la transformée de Fourier 2D. Vous pouvez vous-même créer des signaux sinusoïdaux (en cliquant sur le bouton Draw sin(x)), changer l'angle et la fréquence du signal, et visualiser le résultat de la FFT 2D.

Exercice I : FFT de signaux théoriques 2D

Soit le signal $S_1(x, y) = \cos(2\pi(xf_x + yf_y))$ paramétré par les fréquences f_x et f_y, échantillonné à une fréquence de $100\,\mathrm{px\,mm^{-1}}$. La fonction `S1(fx, fy)` est déjà implémentée.

Q1. Implémentez la fonction `compute_fft2(signal)` qui prend en argument une image et calcule sa transformée de Fourier (normalisée et avec son origine centrée).

En prenant $f_x = 13\,\mathrm{mm^{-1}}$ et $f_y = 0$, affichez le signal $S_1(x, y)$ et son spectre à l'aide de `plot_fft2()`. Vérifiez la position des pics et leur amplitude (donnée par la colorbar). Les valeurs des positions et d'amplitude sont-elles celles attendues ?

> **i** Vous pouvez utiliser la fonction `np.fft.fftshift` pour centrer le spectre de votre transformée de Fourier.

Le concept de normalisation dans les fonctions de la transformée de Fourier rapide (FFT) peut parfois prêter à confusion. Ce que cela signifie, en pratique, c'est que l'on souhaite que l'amplitude du spectre de la FFT d'un signal de magnitude unitaire (par exemple $signal = \sin(2\pi * f * t)$) reste elle aussi unitaire. Il est donc essentiel de bien comprendre l'implémentation de la transformée de Fourier utilisée et de s'adapter en conséquence.
La fonction np.fft.fft2() dans NumPy applique une normalisation particulière (voir la documentation pour fft2 et fft). Par défaut, la fonction entraînera une multiplication des valeurs du spectre par le nombre d'éléments dans le signal. Vous pouvez soit diviser le spectre par la valeur appropriée, soit ajuster directement la normalisation avec le paramètre `fft2(..., norm=...)`.
Vous pouvez également aller regarder le notebook `Normalisation.ipynb` fourni avec les fichiers de ce TP pour mieux comprendre l'utilité de la normalisation et son implémentation.

Prenez le temps de lire la documentation de la fonction `plot_fft2(signal)` et de bien spécifier les arguments appropriés lorsque vous l'utilisez.

Q2. Affichez S_1 et son spectre pour $f_x = 0$ et $f_y = 13\,\mathrm{mm}^{-1}$ puis pour $f_x = 5\,\mathrm{mm}^{-1}$ et $f_y = 12\,\mathrm{mm}^{-1}$. Quel est l'effet d'une rotation de l'image sur son spectre ?

On étudie maintenant le signal : $S_2(x, y) = \cos(2\pi f r)$ où $r = \sqrt{x^2 + y^2}$.

Q3. Implémentez `S2(f)`, puis affichez le signal et son spectre pour $f = 20\,\mathrm{mm}^{-1}$ et pour $f = 40\,\mathrm{mm}^{-1}$. Quel est l'effet d'un rétrécissement de l'image sur son spectre ? Affichez le signal et le spectre pour $f = 120\,\mathrm{px\,mm}^{-1}$. Expliquez l'allure du spectre et les aberrations visibles sur le signal.

Fiez-vous à l'implémentation donnée de la fonction `S1(fx, fy)`. Allez regarder la documentation de `np.meshgrid`.

Exercice II : Filtrage spectral

Cet exercice étudie une image angiographique du réseau coronaire : c'est-à-dire une radiographie des artères qui alimentent le coeur dans lesquelles est injecté un agent de contraste.

Q1. Implémentez la fonction `gaussian(std, size, x0=0, y0=0)` qui renvoie une matrice carrée de taille `size` $\times$ `size` contenant les valeurs d'une gaussienne d'écart-type `std` et centrée sur (`x0`, `y0`). Calculez une gaussienne centrée, d'écart-type 7 et de taille 300×300. Affichez son image et son spectre.

Q2. Chargez l'image `angiographie_bruit.png`. Affichez l'image et son spectre. Identifiez sur le spectre les raies qui sont responsables des rayures diagonales sur l'angiographie (donnez les coordonnées des points concernés).

Q3. À l'aide de la fonction `gaussian` implémentée précédemment, concevez un masque pour filtrer ces raies directement dans le domaine de Fourier (on pourra prendre un écart type de 3 pixels pour les gaussiennes). Appliquez le masque à la transformée de fourier de l'image. Affichez le masque et le spectre filtré.

> Vous pourriez décider de filtrer uniquement les raies dominantes, ou également les harmoniques. À vous de choisir.

Q4. Implémentez la fonction `compute_ifft2(fft_signal)` qui prend en argument la transformée de Fourier d'une image et renvoie l'image reconstituée par transformée de Fourier inverse. Reconstituez et affichez l'angiographie nettoyée de ses rayures diagonales.

> N'oubliez pas la normalisation dans la fft et son effet inverse dans la ifft. De plus, la fonction `np.fft.ifft2` retourne une matrice complexe, à vous de récupérer uniquement les valeurs qui vous intéressent.

Pour la suite de l'exercice on travaillera sur l'angiographie nettoyée ou sur son spectre. Sa transformée de Fourier sera notée T_0.

Q5. On souhaite appliquer un filtre passe-bas dont la réponse fréquentielle est une gaussienne centrée sur l'origine d'écart-type 25. Créez le masque gaussien correspondant, appliquez-le à la transformée de Fourier de l'image et reconstituez-la. Affichez l'image reconstituée et son spectre.

On notera T_{LF} la transformée de Fourier après ce filtrage.

Q6. Calculez l'intensité des fréquences qui ont été retirées du spectre à la question précédente, c'est-à-dire la transformée de Fourier T_{HF} telle que $T_0 = T_{\mathrm{LF}} + T_{\mathrm{HF}}$. Affichez l'image reconstituée et son spectre. Quel est le type de ce filtrage ?

Q7. Créez un masque gaussien centré d'écart-type 1.91 px, de taille 11×11 et normalisé pour que sa somme soit 1. Convoluez ce masque avec l'image nettoyée à la question 4. Affichez l'image filtrée et son spectre.

Q8. Démontrez mathématiquement l'équivalence entre les filtrages réalisés aux questions 5 et 7. Avec quel masque faut-il convoluer l'image nettoyée pour réaliser un filtrage passe-haut équivalent à celui de la question 6 mais dans le domaine spatial ?

On rappelle que la transformée de Fourier d'une gaussienne d'écart-type σ_0 est une gaussienne d'écart-type $\frac{1}{2\pi\sigma_0}$:

$$h(x) = \exp\left(\frac{-x^2}{2\sigma^2}\right) \xrightarrow{\text{Fourier}} H(f) = \sigma\sqrt{2\pi}\exp\left(-2(\pi\sigma f)^2\right)$$

Notons aussi que l'écart-type du masque de la question 5 en px^{-1} est de $\frac{25}{300}$, c'est-à-dire l'écart-type de la gaussienne divisé par le nombre de pixel de la largeur (ou de la hauteur) du spectre.

Q9. (Bonus) Calculez et affichez la transformée de Fourier de l'image `cellules.png`. Cette image représente une certaine lignée cellulaire imagée au microscope. Est-ce que la méthode implémentée dans cet exercice permettrait d'éliminer les lignes qui perturbent l'image ? En quoi le spectre de la FFT est-il différent de celui de l'image angiographique ?

Vous pouvez convertir, d'une façon pertinente, l'image `cellules.png` (qui possède 4 canaux) à une image en tons de gris (1 seul canal), puis utiliser les fonctions préalablement réalisées pour afficher l'image et son spectre.

Séance II.
Apprentissage profond appliqué aux images médicales

Énoncé à venir. Bonne relâche !