

Travail pratique #2

Traitements d'images dans le domaine spatial et fréquentiel

INF600F - Traitement d'images

Hiver 2022

Table des matières

Pondération	1
Exercice 1 : Filtrage spatial (9 pts)	2
Partie 1 : Padding (3 pts)	2
Partie 2 : Convolution 2D (3 pts)	2
Partie 3 : Validation (3 pts)	2
Exercice 2 : Transformée de Fourier (6 pts)	2
Exercice 3 : Filtrage fréquentiel (10 pts)	3
Partie 1 : Filtres fréquents (3 pts)	3
Partie 2 : Validation des filtres (3 pts)	3
Partie 3 : Application des filtres fréquents (4 pts)	4

L'objet de ce travail est de se familiariser avec le traitement d'images dans le domaine spatial et fréquentiel. Ce travail comporte 3 exercices, et il vaut pour 12.5% de la note finale.

Les données et le notebook à utiliser pour effectuer ce travail pratique se trouvent dans l'archive ZIP de ce TP disponible sur le site web du cours. Consultez également le document **Instructions générales pour les travaux pratiques** sur Moodle pour connaître les exigences du rapport de laboratoire.

Pondération

- Exercice 1 : Filtrage spatial (9 pts)
- Exercice 2 : Transformée de Fourier (6 pts)
- Exercice 3 : Filtrage fréquentiel (10 pts)
- **Total** : 25 points

Exercice 1 : Filtrage spatial (9 pts)

Partie 1 : Padding (3 pts)

Écrivez une fonction `g=imPad(f,r,c,padtype,loc)` pour remplir (*pad*) l'image `f` avec `r` lignes au-dessus et en dessous de l'image, et `c` colonnes à gauche et à droite. Si `padtype='zeroes'`, ou si cet argument est absent lors de l'appel de la fonction, la fonction doit implémenter le *zéro padding*. Si `padtype='replicate'`, le remplissage périodique doit être utilisé. Si `loc` est spécifié comme `loc='post'`, la fonction doit se comporter comme ci-dessus, sauf que `r` lignes sont placées uniquement sous l'image et les colonnes `c` sont placées uniquement à sa droite. **Note** : Votre fonction doit conserver le format des pixels de l'image originale `f`.

Partie 2 : Convolution 2D (3 pts)

Écrivez une fonction `g=twodConv(f,w)` pour effectuer une convolution 2D de l'image `f` et du noyau `w`. Cette fonction doit utiliser le remplissage périodique par défaut. Utilisez la fonction `scipy.ndimage.convolve` comme point de départ. Cette fonction utilise le mode de remplissage `reflect` par défaut, mais vous pouvez contourner ce problème en préremplissant `f` avec un votre fonction `imPad`, puis en supprimant les lignes et les colonnes en excès avant de retourner `g`. Votre fonction doit par défaut convertir les images en nombres réels (`float`) et mettre à l'échelle les entrées dans la plage `[0,1]`. Cependant, la fonction doit également avoir la capacité de désactiver cette mise à l'échelle automatique.

Partie 3 : Validation (3 pts)

Créez une image de taille 512×512 pixels qui se compose d'une impulsion unitaire à l'emplacement $(256, 256)$ et de zéros ailleurs. Utilisez cette image et un noyau de votre choix pour confirmer que votre fonction effectue effectivement une convolution. Affichez vos résultats et expliquez ce que vous avez fait et pourquoi.

Exercice 2 : Transformée de Fourier (6 pts)

Représenter et interpréter correctement une image dans le domaine fréquentiel est un préalable indispensable à tout traitement dans ce domaine. Cette partie vous permettra de vous familiariser avec la représentation dans le domaine de Fourier des images.

Voici les étapes à suivre pour cette question.

- Lire et afficher l'image `tp2_ex2.tif`.
- Calculer la transformée de Fourier de l'image. Vous devez utiliser les fonctions `fft2` pour calculer la transformée de Fourier, et `fftshift` pour placer la fréquence nulle au centre de l'image. Ces fonctions font partie du module `np.fft`
- Créer la fonction python `display_fft` pour afficher l'amplitude et la phase de la transformée de Fourier. Assurez-vous d'utiliser une transformation logarithmique pour transformer l'échelle d'intensité de l'amplitude pour l'affichage. Vous devez représenter la transformée de Fourier dans le domaine fréquentiel selon la convention usuelle (c.-à-d. fréquence nulle au

centre de l'image), en indexant les axes en fréquences réduites (comprises entre -1/2 et 1/2) (utilisez l'option `extent` de `plt.imshow`)

- Trouver le message codé secret caché dans le domaine fréquentiel de l'image.
- Expliquez comment le message a été crypté dans l'image et pourquoi il n'était pas visible dans l'image originale.

Note : Par convention, la fonction `fft2` ordonne les fréquences de $0 \rightarrow N/2$, puis de $-N/2 + 1 \rightarrow -1$ pour chaque dimension de la matrice transformée. On pourra utiliser la fonction `fftshift` pour placer la fréquence nulle au centre de l'image.

Exercice 3 : Filtrage fréquentiel (10 pts)

Partie 1 : Filtres fréquentiels (3 pts)

Écrivez une fonction `H=lpFilterTF(type, P, Q, param)` pour générer une fonction de transfert `H` d'un filtre passe-bas de taille $P \times Q$, avec les propriétés suivantes. Si `type='ideal'` `param` doit être un scalaire égal à la fréquence de coupure D_0 dans l'équation suivante :

$$H(u, v) = \begin{cases} 1 & \text{si } D(u, v) \leq D_0 \\ 0 & \text{sinon} \end{cases} \quad (1)$$

Si le paramètre `type='gaussian'` doit être un scalaire égal à l'écart type D_0 dans l'équation suivante :

$$H(u, v) = e^{-D^2(u, v)/2D_0^2} \quad (2)$$

Si `type='butterworth'`, `param` doit être un tableau 1×2 (vecteur) contenant la fréquence de coupure et l'ordre du filtre, $[D_0, n]$, dans l'équation suivante :

$$H(u, v) = \frac{1}{1 + [D(u, v)/D_0]^{2n}} \quad (3)$$

Note : Dans ces équations, $D(u, v)$ représente la distance entre chaque point (u, v) dans le domaine fréquentiel et le centre du rectangle de fréquence $P \times Q$, soit

$$D(u, v) = ((u - P/2)^2 + (v - Q/2)^2)^{1/2} \quad (4)$$

Partie 2 : Validation des filtres (3 pts)

- Générez une fonction de transfert de filtre passe-bas idéale de taille 512×512 avec $D_0 = 96$. Affichez votre résultat sous forme d'image.
- Générez une fonction de transfert de filtre gaussien passe-bas de taille 512×512 avec $D_0 = 96$. Affichez votre résultat sous forme d'image.

- Générez une fonction de transfert de filtre passe-bas Butterworth de taille 512×512 avec $D_0 = 96$ et $n = 2$. Affichez votre résultat sous forme d'image.

Partie 3 : Application des filtres fréquentiels (4 pts)

Montrez l'image obtenue après chacun des filtres suivants.

- Lisez l'image `tp2_ex3.tif` et filtrez-la à l'aide d'un filtre passe-bas idéal afin que la grande lettre "a" soit à peine lisible et que les autres lettres ne le soient pas.
- Lisez l'image `tp2_ex3.tif` et filtrez-la à l'aide d'un filtre passe-bas gaussien pour que la grande lettre "a" soit à peine lisible et que les autres lettres ne le soient pas.
- Lisez l'image `tp2_ex3.tif` et filtrez-la à l'aide d'un filtre Butterworth de votre spécification afin que, lorsqu'elle est filtrée et seuillée, l'image résultante ne contienne qu'une partie du grand carré en haut à droite. (*Astuce* : il est plus intuitif de travailler avec le négatif de l'image originale.)