TP4 – Déconvolution

Indication : une archive contenant des signaux utiles au TP est à récupérer sur la page web suivante : `http://www.syncpoint.fr/#Teaching`

On désigne par $\boldsymbol{f}$ la représentation vectorielle d'une image $f[n,m]$, avec $n \in [\![1,N]\!]$ et $m \in [\![1,M]\!]$. La DFT 2D associée à une image $\boldsymbol{f}$ sera notée par une majuscule $\boldsymbol{F}$, avec :

$$F[k,l] = \sum_{n=1}^{N}\sum_{m=1}^{M} f[n,m]\mathrm{e}^{2i\pi(k-1)(n-1)/N}\mathrm{e}^{2i\pi(l-1)(m-1)/M}$$

On notera $\boldsymbol{\Lambda_h}$ la matrice de transformation associée à la convolution circulaire d'un filtre $\boldsymbol{h}$ avec une image de taille $N \times M$:

$$\boldsymbol{g} = \boldsymbol{\Lambda_h}\cdot\boldsymbol{f} \;\Rightarrow\; g[p,q] = \sum_{n=1}^{N}\sum_{m=1}^{M} f[n,m]\hat{h}[n-p+1, m-q+1],$$

où $\hat{h}$ correspond à h périodisé :

$$\forall(p,q)\in\mathbb{Z}^2,\; \hat{h}[k+pN, l+qM] \equiv h[k,l].$$

Comme pour les images, la DFT 2D du filtre $\boldsymbol{h}$ sera noté $\boldsymbol{H}$.

Enfin, on notera par $\boldsymbol{\gamma_f}$ la matrice d'autocorrélation d'un vecteur image $\boldsymbol{f}$, et par $\boldsymbol{\Gamma_f}$ la DFT 2D de $\boldsymbol{\gamma_f}$. $\boldsymbol{\Gamma_f}$ correspond à la densité spectrale de puissance de l'image $\boldsymbol{f}$ (théorème de Wiener-Khinchin) :

$$\Gamma_{\boldsymbol{f}}[k,l] = \frac{1}{N}\left|F[k,l]\right|^2$$

1. **Création de l'image originale et sa version dégradée**
   Dans ce TP, l'image $\boldsymbol{g}$ correspond à la version dégradée d'une image origine $\boldsymbol{f}$:
   $$\boldsymbol{g} = \boldsymbol{\Lambda_h}\cdot\boldsymbol{f} + \boldsymbol{b},$$
   où $\boldsymbol{h}$ est un filtre déterministe et $\boldsymbol{b}$ est un bruit blanc gaussien de variance σ_b.
   (a) Importer le fichier image `camera.pgm` de l'archive (lien au début du sujet) :
   ```
   f = double(imread('camera.pgm'));
   ```
   Calculer sa DFT 2D $\boldsymbol{F}$ à l'aide de la fonction `fft2`.

(b) Importer le filtre $\boldsymbol{h}$, disponible dans le fichier `im_filter.mat` :
```
load im_filter.mat;
```
Calculer sa DFT 2D $\boldsymbol{H}$.

(c) Créer le bruit blanc gaussien $\boldsymbol{b}$ et calculer l'image dégradée $\boldsymbol{g}$. Pour calculer le produit de convolution circulaire, la méthode la plus simple (et la plus efficace en terme de complexité) est de passer dans l'espace de Fourrier, où la convolution circulaire est transformée en multiplication *element-wise* :
```
g = ifft2(H.*F) + b
```
Afficher les images f et g avec la fonction `imshow`. Selon vous, quel type de dégradation venez vous de réaliser ? Calculer l'erreur quadratique moyenne entre l'image originale et sa version dégradée avec la fonction `std`.

2. **Restauration de l'image par déconvolution**

On cherche maintenant à produire une image restaurée $\boldsymbol{f_e}$, à partir de l'image dégradée $\boldsymbol{g}$. L'image originale $\boldsymbol{f}$ est indisponible dans la réalité mais elle est fournie dans le cadre de ce TP de façon à pouvoir évaluer l'erreur de restauration.

On estimera l'image $\boldsymbol{f}$ à l'aide de la relation suivante :

$$\boldsymbol{f_e} = (\boldsymbol{\Lambda_h}^\top \cdot \boldsymbol{\Lambda_h} + \mu \boldsymbol{\Lambda}_{\gamma_{\boldsymbol{f}}}^{-1})^{-1} \cdot \boldsymbol{\Lambda_h}^\top \cdot \boldsymbol{g}$$

avec $\mu > 0$. Cette relation peut être approchée efficacement à l'aide d'une DFT 2D par la relation suivante :

$$F_e[k,l] = \frac{H[k,l]^* \; G[k,l]}{|H[k,l]|^2 + \frac{\mu}{\Gamma_{\boldsymbol{f}}[k,l]}}$$

(a) Comment choisir μ pour mettre en œuvre un filtrage inverse ? Observer l'image reconstruite. Calculer ses valeurs minimales et maximales et commenter.

(b) En choisissant des valeurs non nulles de μ et en prenant $\Gamma_{\boldsymbol{f}}[k,l] = 1$, on réalise une régularisation quadratique. Faire varier μ. Que se passe t'il quand ce paramètre est grand ? Quelle est la valeur optimale du facteur de régularisation ?

(c) On modélise l'image $\boldsymbol{f}$ par un bruit blanc d'écart-type σ_f. En posant $\Gamma_{\boldsymbol{f}}[k,l] = \sigma_f^2$ et $\mu = \sigma_b^2$, on retombe sur le filtrage de Wiener. Mettre en œuvre ce filtre et mesurer ses performances. Comment cette méthode se situe t'elle par rapport à la précédente ? Comment pourrait on améliorer les résultats ?