

Responsable TP : Dr. SID AHMED Soumia

TP3 : Filtrage Spatial

Objectif

A la fin de ce TP, l'étudiant sera capable de :

1. Comprendre le concept de la convolution dans le domaine spatial.
2. Comprendre le concept de filtrage linéaire et non linéaire.
3. Réduire le bruit d'une image à l'aide de différents types de filtres.

I. Filtrage spatial linéaire (convolution)

Dans Matlab, le filtrage par convolution linéaire peut-être effectuer analytiquement en définissant le filtre h à utiliser ou en utilisant la fonction *fspecial* de Matlab. Si vous souhaitez supprimer le bruit ou lisser votre image, vous pouvez utiliser les fonctions *fspecial()* et *imfilter()* pour créer et appliquer un filtre spécifique pour lisser l'image donnée.

Méthodes analytiques :

```
h = ones(5, 5)/25;           % Filtre moyennneur de taille 5 x 5
```

Méthodes en utilisant fspecial

```
h = fspecial('average', [5 5]); % Filtre moyennneur de taille 5 x 5, Le premier argument de est le nom du filtre, et le second argument est la taille du filtre.
```

-Taper '*doc fspecial*' dans l'espace de commande de Matlab pour voir l'explication de cette fonction. Vérifier le masque généré pour les différents types.

On réalise ensuite le filtrage de l'image grâce à la commande *imfilter* : $Y = imfilter(X, h)$;

%Le premier argument de la fonction *imfilter()* est l'image que vous souhaitez lisser, et le second argument est le filtre que vous avez créé à l'aide de la fonction *fspecial()*.

II. Filtrage spatial non linéaire

Le filtrage non-linéaire que nous allons considérer dans ce TP est le filtrage médian, qui peut être réalisé grâce à la fonction *medfilt2* :

```
Y = medfilt2(X, [3 3]);      % Filtrage median de taille 3 x 3 pour l'image X
```

III. Le bruit

Pour ajouter du bruit à une image sous MATLAB, on utilise la commande *imnoise* :

`Y = imnoise(X,'gaussian',m,v) % Applique un bruit additif gaussien de moyenne m et de variance v à une image X`

`Y = imnoise(X,'salt & pepper',p) % Applique un bruit poivre et sel de pourcentage p à une image X`

IV. Le PSNR

Le rapport signal / bruit de crête (PSNR) est le rapport entre la puissance maximale possible d'une image et la puissance du bruit corrompant qui affecte la qualité de sa représentation. Pour estimer le PSNR d'une image, il est nécessaire de comparer cette image à une image propre idéale avec la puissance maximale possible.

Le PSNR est défini comme suit :

$$PSNR = 10 \log_{10} \left(\frac{R^2}{\frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N [x^0(m,n) - x^d(m,n)]^2} \right)$$

Où x^0 et x^d sont respectivement les images originales (bruitée) et débruitées et R est la valeur maximale possible pour un pixel (par exemple pour 8bits R=255).

Sous MATLAB :

`PSNR=20*log10(255) -10*log10(std2(X-Y)) ; % X : image originales (bruitée), Y : image débruitée.`

V. Manipulations

Partie 1

1. Ouvrir sous MATLAB un script vide nommé TP3_votre_nom.m
2. Lire l'image « hestain.png » la stocker dans une matrice im1.
3. Appliquer sur l'image im1 un bruit blanc Gaussien de variance $\sigma^2 = 0.01$ et stocker le résultat dans la matrice noisy1. En utilisant *subplot* afficher sur la même figure (1) l'image originale et l'images bruitée. Commenter.
4. Appliquer sur l'image im1 un bruit poivre et sel avec un pourcentage de pixels modifiés p=0.05 et stocker le résultat dans la matrice noisy2. En utilisant *subplot* afficher sur la même figure (2) l'image originale et l'image bruitée. Commenter.
5. Afficher sur une même figure (3) les images de im1, noisy1 et noisy2.

Partie 2

On continue de travailler sur les images im1, noisy1 et noisy2 utilisées dans la partie 1.

1. Appliquer un filtre moyennneur de taille 3×3 sur l'image noisy1 et stocker le résultat dans la matrice filtered1. En utilisant *subplot* afficher sur la même figure im1, noisy1 et filtered1. Le bruit a-t-il été atténué ?
2. Appliquer un filtre médian de taille 3×3 sur l'image noisy2 et stocker le résultat dans filtered2. En utilisant *subplot* afficher sur la même figure im1, noisy2 et filtered2. Le bruit a-t-il été atténué ?
3. Calculer le PSNR pour les deux simulations précédemment réalisées. Sachant qu'on considère en général qu'un excellent débruitage offre un PSNR d'au moins 20 dB, les résultats vous semblent-ils logiques ?
4. Tester les 6 filtres suivants sur noisy1, puis sur noisy2. Est-ce qu'il y a une diminution du bruit après le filtrage ?
 - (a) Filtre moyennneur : 3×3 et 5×5
 - (b) Filtre Gaussien de taille 15×15 : $\sigma = 2$, et $\sigma = 0.5$
 - (c) Filtre médian : 3×3 et 5×5