

TP3 : FILTRAGE D'UNE IMAGE : DOMAINE SPATIAL

## 1. Objectifs

Le but de ce TP est de se familiariser avec certaines fonctions de la toolbox de traitement d'images et de comparer les performances de différents filtres pour le débruitage d'images.

## 2. Types de dégradation d'une image

- **Le bruit additif**, qui affecte tous les pixels de l'image. Dans ce TP, nous considérerons un bruit blanc additif Gaussien, de moyenne nulle et de variance  $\sigma^2$ . Le bruit Gaussien affecte à la fois les basses et les hautes fréquences. Il est caractérisé par sa variance  $\sigma^2$  : plus  $\sigma^2$  est élevé, plus l'image est dégradée.
- **Le bruit impulsif**, n'affecte que certains pixels de l'image. Dans ce TP, nous considérerons un bruit sel et poivre (salt & pepper), qui est une dégradation de l'image sous la forme de pixels noirs et blancs répartis au hasard. On le caractérise par le pourcentage p de pixels modifiés : plus p est élevé, plus l'image est dégradée.

**Y = imnoise(X, 'gaussian', m, v)**

**Y = imnoise(X, 'salt & pepper', p)**

- Appliquer sur l'image cameraman.tif un bruit blanc Gaussien de variance  $\sigma^2 = 0.01$ .
- Afficher sur la même figure l'image originale et l'image bruitée. Faire varier  $\sigma^2$  et commenter.
- Refaire le même travail avec un bruit poivre et sel avec un pourcentage p = 0.05. Faire varier p et commenter.

## 3. Filtrage d'une image

**Filtrage linéaire** : Pour réaliser un filtrage linéaire, il faut d'abord définir le masque h à utiliser. Pour cela, soit on le définit de façon analytique, soit on utilise la fonction fspecial de MATLAB.

- Méthode analytique

**h = ones(3,3)/9;** % Filtre moyennneur de taille 3 x 3

**h = [1 0 1 ; 0 2 0 ; 1 0 1]/6;** % Filtre ad hoc

- Méthodes en utilisant fspecial

**h = fspecial('average', [3 3]);** % Filtre moyennneur de taille 3 x 3

**h = fspecial('gaussian', [5 5], 0.5);** % Filtre gaussien de taille 5 x5 et d'écart type 0.5

- On réalise ensuite le filtrage de l'image grâce à la commande **imfilter** :

**Y = imfilter(X, h, 'replicate');** % X : image normalisée (valeurs entre 0 et 1) et h : masque de convolution

Les filtres considérés dans cette section permettent de lisser une image. Ce sont des passe-bas normalisés de sorte

que la somme des coefficients du masque est égale à 1. Ces filtres peuvent aussi servir à restaurer une image

bruitée, c'est-à-dire à se rapprocher de l'image d'origine au moyen d'opérations sur l'image bruitée.

**Filtrage non-linéaire** : Le filtrage non-linéaire que nous allons considérer ici est le filtrage médian, qui peut être réalisé grâce à la commande **medfilt2** :

**Y = medfilt2(X, [3 3]);** % Filtrage médian de taille 3 x 3

Pour les images bruitées :

- Appliquer un filtre moyennneur de taille  $3 \times 3$  sur les deux images dégradées, Le bruit a-t-il été atténué ?
- Appliquer un filtre médian de taille  $3 \times 3$  sur les deux images dégradées, Le bruit a-t-il été atténué ?
- Modifier les tailles des filtres, lequel donne la meilleure performance (PSNR)? Les résultats vous semblent-ils logiques ?conclure.

Afin de pouvoir quantifier la qualité du débruitage, on va utiliser une mesure objective appelée Peak Signal to

Noise Ratio (PSNR) et définie par : 
$$PSNR = 10 \log_{10} \left( \frac{R^2}{\frac{1}{MN} \sum_{m=1}^M \sum_{n=1}^N (I^o(m,n) - I^d(m,n))^2} \right)$$

Où  $I^o$  et  $I^d$  sont respectivement les images originelles et débruitées et R est la dynamique du signal (valeur maximale possible pour un pixel). Sachant qu'on considère en général qu'un excellent débruitage offre un PSNR d'au moins 20 dB.

Si l'on travaille sur des images normalisées, on a  $R = 1$  et le PSNR peut se calculer facilement grâce à l'instruction :

**EQM=mean2((I0-Id).^2); PSNR=-10\*log10(EQM);**



#### 4. Travail demandé

- Appliquer sur l'image cameraman.tif un bruit blanc Gaussien de moyenne nulle et de variance  $\sigma^2 = 0.01$ .

```
X=imread ('.....');
X=double(X)/255;
Yb = imnoise('gaussian'.....);
```

- Afficher sur la même figure l'image originale et l'image bruitée.

```
figure(1);subplot(231); imshow(.....);title('image orig')
subplot(232); imshow(....) ;title('bruit additif Gaussien ')
```

- Faire varier  $\sigma^2$  et commenter.

- Refaire le même travail avec un bruit poivre et sel avec un pourcentage  $p = 0.05$ .

```
Ys = imnoise(X,'.....&.....',.....);
subplot(233); imshow(....) ;title('bruit impulsif')
subplot(234); imshow(....) ;title('bruit impulsif')
```

- Faire varier  $p$  et commenter.

Pour les trois filtres et les deux images bruitées :

**Pour l'image1 (Bruit blanc Gaussien de moyenne nulle et de variance  $\sigma^2 = 0.01$ ) :**

- Appliquer un filtre moyennneur de taille  $3 \times 3$ .

```
h_m = fspecial('average',.....);
ybmo = imfilter(Yb,h_m,'replicate');
```

- Appliquer un filtre gaussien de taille  $3 \times 3$  et de variance  $\sigma^2 = 0.8$ .

```
h_g = fspecial('gaussian',.....);
ybg = imfilter(.....,h_g,'replicate');
```

- Appliquer un filtre médian de taille  $3 \times 3$ .

```
ybm = medfilt2(.....);
subplot(234); imshow(ybm);title(' Filtre moyennneur ');
subplot(235); imshow(ybg);title('Filtre gaussien ');
subplot(236); imshow(ybm);title('Filtrage médian ');
```

- Le bruit a-t-il été atténué ?.....

- Modifier les tailles des filtres, lequel donne la meilleure performance (PSNR)?

```
EQMb=mean2( (X-Yb).^2 );PSNRb=-10*log10(EQMb);
EQMmo=mean2( (X-ybm).^2 );PSNRmo=-10*log10(EQMmo);
EQMg=mean2( (X-ybg).^2 );PSNRg=-10*log10(EQMg);
EQMmed=mean2( (X-ybm).^2 );PSNRmed=-10*log10(EQMmed);
```

- Les résultats vous semblent-ils logiques ?

**Pour l'image 2 ( Bruit poivre et sel avec un pourcentage  $p = 0.05$ )**

- Appliquer un filtre moyennneur de taille  $3 \times 3$ .

```
h_m = fspecial('average',.....);
ysmo = imfilter(Ys,h_m,'replicate');
```

- Appliquer un filtre gaussien de taille  $3 \times 3$  et de variance  $\sigma^2 = 0.8$ .



```
h_g = fspecial('gaussian', ..... , .....)
```

```
ysg = imfilter(.....,h_g,'replicate');
```

- Appliquer un filtre médian de taille  $3 \times 3$ .

```
ysmed = medfilt2(..... , .....);
```

```
subplot(234); imshow(ysmoy);title(' Filtre moyennneur ')
```

```
subplot(235); imshow(ysg);title('Filtre gaussien ')
```

```
subplot(236); imshow(ysmed);title('Filtrage médian ')
```

- Le bruit a-t-il été atténué ?.....

- Modifier les tailles des filtres, lequel donne la meilleure performance (PSNR)?

```
EQMb=mean2( (X-Yb).^2 );PSNRb=-10*log10(EQMb);
```

```
EQMmoy=mean2( (X-ysmoy).^2 );PSNRmoy=-10*log10(EQMmoy);
```

```
EQMg=mean2( (X-ysg).^2 );PSNRg=-10*log10(EQMg);
```

```
EQMmed=mean2( (X-ysmed).^2 );PSNRmed=-10*log10(EQMmed);
```

.....

.....

- Les résultats vous semblent-ils logiques ?

.....

.....

<i>Filtres</i>	<i>Taille</i>	<i>Image1 (Bruit blanc Gaussien )</i> <i>PSNR=.....dB.</i>	<i>Image 2 ( Bruit poivre et sel)</i> <i>PSNR=.....dB.</i>
<b><i>Filtre moyennneur</i></b>	3x3		
	5x5		
	7x7		
	11x11		
	15x15		
<b><i>Filtre gaussien</i></b>	3x3		
	5x5		
	7x7		
	11x11		
	15x15		
<b><i>Filtre médian</i></b>	3x3		
	5x5		
	7x7		
	11x11		
	15x15		

- Conclusion :

.....

.....

.....

.....

### Question

Choisissez une image en niveaux de gris et appliquez un bruit blanc gaussien avec un écart-type à 0.2. Appliquez un filtre gaussien de taille 15 x 15 et choisissez l'écart-type qui permet d'avoir le meilleur PSNR. Que pensez-vous du résultat obtenu ?

.....

.....

.....

.....

