



Ecole Nationale Supérieure
des Télécommunications de Bretagne

Mineure MTS 204 - Bureau d'études
Introduction au traitement d'images

9 novembre 2015

Les programmes Matlab, appliqués aux images réelles et illustrant les exercices, sont disponibles dans :

`http://public.enst-bretagne.fr/~burdin/mts204/`

ou

sous moodle dans le cours MTS204.

Rapatrier chez vous l'archive des programmes Matlab.

Pour lancer Matlab, taper les commandes suivantes :

```
SETUP CLASSROOM2011b  
matlab
```

Exercice 1

I. On considère le masque de convolution 2D suivant :

$$\begin{bmatrix} 0 & \frac{1}{2} & 0 \\ \frac{1}{2} & 1 & \frac{1}{2} \\ 0 & \frac{1}{2} & 0 \end{bmatrix}$$

1. Calculer la fonction de transfert de ce filtre 2D¹.

1. Voir poly, diapos 150 à 166

2. Calculer sa réponse fréquentielle puis représenter son gain.
3. Donner l'équation de filtrage 2D.
4. Quelle est cette famille de filtres ?

II. Application

Lancer le programme *exo1* plusieurs fois avec différentes images, différents types de bruit et paramètres, et différents filtres.

5. Sélectionner une image et un type de bruit. Changer la taille du filtre et les coefficients (Ne mettre que des coefficients positifs et normaliser le filtre).
6. Analyser l'effet sur l'image (zones homogènes et contours).
7. Que se passe-t-il si vous mettez des coefficients négatifs dans le filtre ?

Exercice 2

On se propose d'étudier un filtre réduisant le bruit tout en préservant les contours (filtre de Nagao)².

Soit l'image synthétique $I(i, j)$ suivante :

$$I(i, j) = \begin{bmatrix} 50 & 45 & 40 & 40 & 27 \\ 10 & 18 & 42 & 36 & 18 \\ 25 & 35 & 41 & 38 & 10 \\ 95 & 128 & 100 & 25 & 35 \\ 100 & 125 & 136 & 20 & 32 \\ 110 & 130 & 145 & 130 & 22 \end{bmatrix},$$

Pour chaque pixel de l'image $I(i, j)$, on définit 9 voisinages :

On utilise, comme le montre la figure précédente :

Figure 1.a : 1 voisinage V_1

Figure 1.b : 4 voisinages V_2, \dots, V_5 (par rotation de 90°)

Figure 1.c : 4 voisinages V_6, \dots, V_9 (par rotation de 90°)

Le principe de ce filtre est d'attribuer, au point (i, j) à filtrer, la moyenne du voisinage ayant la variance minimale en niveaux de gris parmi les 9 choix possibles.

2. Voir poly, diapos 172 à 174

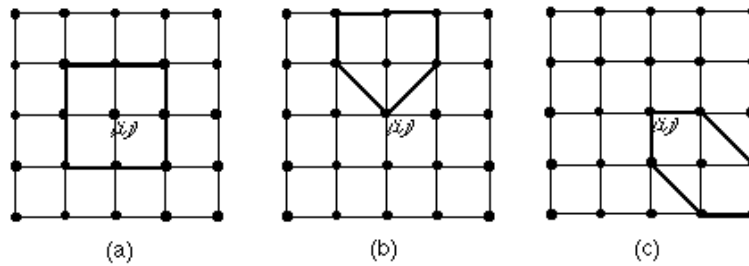


FIGURE 1 – Voisinages considérés autour du point à filtrer

1. Pour le point de coordonnées $(4,3)$ (niveau de gris 100), construire les 9 voisinages à la main et calculer en ligne, pour chacun, la moyenne et l'écart-type. (On pourra utiliser les fonctions $mean(V_i)$ et $std(V_i, 1)$ de Matlab.)
2. Donner la valeur de ce pixel dans l'image filtrée et le masque du filtre retenu pour ce point.
3. Analyser la pertinence de ce filtre par rapport à la topologie locale du point de coordonnées $(4,3)$.
4. Lancer le programme *nagao2* pour voir l'effet du filtre de Nagao sur une image.

Les filtres sont utilisés en particulier pour débruiter les images. Afin de comparer les différents traitements, il est possible de définir des paramètres quantifiés d'évaluation du débruitage dépendant, comme par exemple, le PSNR (Peak Signal Noise Ratio).

5. Chercher la définition du PSNR et implémenter la formule.
6. Sélectionner une image et créer 5 nouvelles images en la bruitant avec un niveau de bruit gaussien croissant.
7. Utiliser le filtre de Nagao et calculer le PSNR correspondant aux 5 images bruitées. Tracer la courbe du PSNR pour le filtre de Nagao.
8. Utiliser plusieurs lisseurs (*exo1*) et calculer le PSNR correspondant aux 5 images bruitées pour chacun d'eux. Tracer leur courbe de PSNR en superposant à la courbe précédente.
9. Utiliser un filtre plus évolué dit "déconvolution de Wiener" et calculer le PSNR correspondant aux 5 images bruitées. Tracer la courbe du PSNR en superposant aux courbes précédentes.
10. Commenter.

Exercice 3

I. On donne le masque de convolution suivant :

$$\begin{bmatrix} -1 & 0 & 1 \\ -k & 0 & k \\ -1 & 0 & 1 \end{bmatrix} \quad \downarrow i \quad \rightarrow j$$

1. Montrer que ce filtre 2D est séparable.
2. Quelle est la fonction de filtrage réalisée dans chacune des directions.
3. Ecrire de deux façons l'équation de filtrage.

II. Soit une image représentant un contour du type échelon :

$$I(i, j) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & A & A & A & A & A & A \\ 0 & 0 & 0 & 0 & 0 & A & A & A & A & A & A \\ 0 & 0 & 0 & 0 & 0 & A & A & A & A & A & A \\ 0 & 0 & 0 & 0 & 0 & A & A & A & A & A & A \\ 0 & 0 & 0 & 0 & 0 & A & A & A & A & A & A \\ 0 & 0 & 0 & 0 & 0 & A & A & A & A & A & A \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

1. A l'aide de Matlab et de l'image ci-dessus, vérifier que le masque de la question I est séparable. (*On pourra prendre $k = 2$.*)
2. Que réalise ce filtre 2D ?
3. Modifier le code Matlab pour afficher les contours dans l'autre direction.
4. Rajouter une ligne pour afficher tous les contours de l'image. On utilisera les valeurs des pixels des deux images précédentes comme coordonnées du champs vectoriel gradient, orthogonal aux contours.