

# Travaux Pratiques - IMA201

## Filtrage et Restauration

---

## 1 Préliminaires : utilisation de python

### 1.1 Avant de commencer le TP (Rappel)

Il faut exécuter les commandes

```
export PATH=/cal/softs/anaconda/anaconda3/bin:$PATH
source /tsi/tp/TP/Init/debut_tp_filtrage_python.sh
spyder programmation/TPfilt_restau.py &
```

La première ligne vous permet d'utiliser python en version 3 distribué sous forme anaconda. Vous pouvez l'ajouter dans vos fichiers de configuration afin d'utiliser toujours cette version de python.

La seconde ligne crée un répertoire TPIMAFILT dans lequel se trouvera un lien symbolique vers le répertoire images. Vous serez automatiquement mis dans ce répertoire. Les images se trouvent dans le (lien symbolique vers le) répertoire images (dans lequel vous ne pouvez pas créer ou sauvegarder des images, créez-les dans votre répertoire de travail). Vous pouvez aussi bien sûr travailler sur les images de votre choix.

Vous trouverez dans le répertoire programmation un fichier tp\_initiation.py qui contient quelques commandes pour commencer sous python. Il faut d'emblée exécuter les SECTIONS 1 et 2 afin d'inclure les packages nécessaires au TP. La SECTION 2 définit des fonctions utiles pour ce TP. Ensuite la SECTION 3 contient les commandes correspondant aux différentes questions du TP précédent (pour rappel d'usage) . À la toute fin des exemples spécifiques au TP actuel sont donnés

Vous pouvez utiliser un autre environnement que spyder, assurez-vous seulement que vous êtes dans le bon répertoire lorsque vous voulez accéder aux images. Les commandes python `pwd` et `cd` sont utiles pour cela.

### 1.2 Les images dans python

Pour rappel le fichier TPfilt\_restau.py contient des commandes pour visionner des images et charger des images sous python.

## 2 Transformation géométrique

Utiliser la fonction (`rotation`) pour transformer une image de votre choix. Quelle différence y-a-t-il entre la méthode à plus proche voisin et la méthode bilinéaire ?

Que constatez-vous sur une image qui aurait subi huit rotations de 45 degrés (en bilinéaire et en plus proche voisin) ?

Que constatez-vous si vous appliquez la rotation avec un facteur de zoom inférieur à 1 (par exemple 1/2) ? Qu'aurait-il fallu faire pour atténuer l'effet constaté ?

## 3 Filtrage linéaire et médian

Dans cette section on se propose de comparer les filtres linéaire et médian. Pour appliquer un filtrage linéaire il faut d'abord générer un noyau. Le filtrage linéaire consiste à convoluer le noyau avec l'image. Vous pouvez utiliser les commandes `get_gau_ker` et `get_cst_ker` pour générer un noyau gaussien ou constant. Pour appliquer ces noyaux il faut utiliser la commande `filtre_lineaire`. La commande `median_filter` applique un filtre médian<sup>1</sup>.

---

1. utiliser la commande `help` pour savoir comment utiliser chaque commande

- Expliquer le rapport entre la taille du noyau (size) renvoyé par `get_gau_ker` et le paramètre cette commande.
- Après avoir ajouté du bruit à une image simple telle que **pyramide.tif** ou **carre\_orig.tif** et avoir filtré le résultat avec des filtres linéaires, expliquez comment on peut évaluer (sur des images aussi simples) la quantité de bruit résiduel (la commande `var_image` donne la variance d'une partie d'une image).
- Appliquer un filtrage médian à une image bruitée et comparer le résultat avec un filtrage linéaire.
- Faites une comparaison linéaire/médian sur l'image **pyra-impulse.tif**. Que constatez-vous ? Expliquer la différence de comportement entre filtrage linéaire et médian sur le point lumineux situé en haut à droite de l'image **carre\_orig.tif**.

## 4 Restauration

- Appliquer un filtre linéaire à une image puis utilisez la fonction `filtre_inverse`<sup>2</sup>. Que constatez-vous ? Que se passe-t-il si vous ajoutez très peu de bruit à l'image floutée avant de la restaurer par la **commande** précédente ?
- Comment pouvez-vous déterminer le noyau de convolution qu'a subi l'image **carre\_flou.tif** ?
- Après avoir ajouté du bruit à cette image utilisez la fonction `wiener` pour restaurer cette image. Cette commande implémente la formule de restauration :

$$\hat{f}(\omega) = \hat{g}(\omega) \cdot \frac{\overline{\hat{K}}(\omega)}{|\hat{K}|^2(\omega) + \lambda\omega^2}$$

où  $\lambda$  (qui est spécifié par l'utilisateur) est proportionnel à la variance du bruit ( $\sigma^2$ ) et inversement proportionnel à la dépendance du spectre de l'image en fonction de  $\omega^2$ . Faites varier le paramètre  $\lambda$  et commentez les résultats.

## 5 Applications

### 5.1 Comparaison filtrage linéaire et médian

Pour une image simple telle que **carre\_orig.tif** et un bruit d'écart-type 5, trouver la taille du noyau constant qui réduit le bruit dans les mêmes proportions qu'un filtre médian circulaire de rayon 4. (explicitiez l'algorithme utilisé)

### 5.2 Calcul théorique du paramètre de restauration

Ici on travaille à nouveau sur l'image **carre\_flou.tif** que l'on bruite et restaure par wiener. Modifiez la fonction `wiener` afin qu'elle utilise le spectre de l'image dégradée à place de  $\lambda\omega^2$ .

On rappelle que la formule du filtrage de Wiener, connaissant la densité spectrale de l'image, notée  $\sigma_s^2(\omega)$  est

$$\hat{f}(\omega) = \frac{\overline{\hat{K}}(\omega)}{|\hat{K}(\omega)|^2 + \frac{\sigma_b^2}{\sigma_s^2(\omega)}}$$

La valeur de  $\sigma_s^2(\omega)$  sera prise comme égale au carré de la TF du signal dégradé et la valeur  $\sigma_b^2$  sera la variance du bruit multiplié par le nombre de pixels de l'image pour une raison de normalisation de la TF.

---

2. Si vous utilisez un noyau constant ne le prenez **pas** d'une taille qui divise la taille de l'image