

TP 3 : FILTRE DE WIENER

La séance de TP se fait sous environnement Windows, sauf si vous avez une nette préférence pour Linux.

Pour commencer la séance

1. Télécharger l'archive. Un fois décompressée on obtient un dossier `TIVA_TP3`.
2. Lancer ensuite `Matlab` et modifier le répertoire de travail en choisissant le répertoire `Bureau/TIVA_TP3` que vous avez créé.
3. Pour les différentes questions, vous pouvez utiliser un «copier-coller» à partir de ce document. Il est fortement recommandé de saisir toutes les commandes dans la fenêtre de l'éditeur que vous avez ouverte. Pour exécuter les commandes saisies, il suffit de les sélectionner avec la souris et d'appuyer sur la touche `F9`.
4. Pour inclure des commentaires dans le programme, ce qui est fortement recommandé, vous devez utiliser le caractère `%`. Tout ce qui suit ce caractère sera négligé lors de l'exécution.

0. Affichage d'image avec `imshow`...

La commande `image` que nous avons utilisée pour afficher des images JPEG en couleur, ne fonctionne malheureusement pas pour les images TIF ou PNG en niveau de gris. Aujourd'hui notre image de base est au format TIF. On utilisera la commande `imshow` pour l'afficher. Comme `imshow.m` se trouve dans le fichier `imshow`, il faut dire à Matlab qu'il aille chercher la fonction dans ce dossier là. Choisissez le répertoire `TIVA_TP3` comme répertoire courant puis exécutez

```
addpath('./imshow/');
```

Maintenant lorsque vous taperez une commande, Matlab ira aussi chercher dans `./imshow/` pour voir si le script ou la fonction correspondante s'y trouve.

1. Plaque floue...

Le but ici est d'utiliser le filtre de Wiener. Une voiture a été prise en photo¹ roulant à bonne allure devant l'Université d'Oxford. La plaque est tellement floue qu'elle est illisible... Mais le flou étant dû à la translation rectiligne de la voiture, il y a bon espoir de pouvoir faire de la déconvolution... Il est recommandé de garder les commandes que vous exécuterez au fur et à mesure dans un script afin de pouvoir assembler facilement les différents morceaux.

1. Cette photo est gracieusement mise à notre disposition par Andrew Zisserman, professeur en Vision Artificielle à l'Université d'Oxford

1. Chargez l'image `car.tif` et visualisez la avec la séquence de commandes `I=imread('car.tif');`
`imshow(I);`
La voiture est floue du fait de son déplacement.
2. Si la voiture était prise exactement de profil, le flou correspondrait exactement à la convolution de l'image nette avec un segment horizontal. Comme la plaque n'est pas très grande, on va considérer que son image floue est obtenue par convolution avec un segment qui n'est pas horizontal, et puisque la convolution avec un segment diagonal est compliqué, on commence par faire une rotation de l'image et extraire une région d'intérêt autour de la plaque. Pour simplifier les choses, cette image est déjà extraite. Elle est sauvée dans `plaque.mat`. Charger l'image avec `load('plaque');` en prenant garde d'être dans le bon dossier. La variable `plaque` devrait être définie dans l'environnement de travail. Pour afficher la liste des variables de l'environnement de travail vous pouvez utiliser la commande `who` où `whos` dans la ligne de commande.

2. Construction du noyau de convolution

1. On construit d'abord le noyau de convolution h : un segment horizontal de longueur L qui permet par convolution de faire la moyenne sur L pixels. On pourra estimer L à l'oeil en zoomant sur l'imagette et essayer plusieurs valeurs différentes. On construit une image qui a les mêmes dimensions que `I`.
`[N1,N2]=size(I);`
`seg=zeros(size(I));`
`seg(1,mod(round(L/2:L/2),N2))=1/L;`
Pourquoi a-t-on construit `seg` comme cela plutôt qu'en le centrant sur l'image ? Justifiez avec la formule de la convolution. Notez qu'on peut obtenir une version centrée avec `fftshift`.
2. Calculez la TFD H de `seg` et affichez son module avec `imagesc` en recentrant la TFD de façon que l'origine soit au centre de l'image grâce à `fftshift`. Avec la commande `plot`, faites une représentation du profil longitudinal du module de H .
3. Calculez la TFD de l'image de la plaque, après l'avoir converti au format double.

3. Application du filtre de Wiener

1. Calculez une matrice W qui correspond au filtre de Wiener. On considérera que $K(u, v)$ est une constante qu'il faudra trouver par essai et erreur, disons $K = 0.001$ pour commencer. La commande `conj` pourra servir.
2. Faites une fonction `deblurr` qui prend en entrée K , L et l'imagette, qui calcule le filtre de Wiener pour ces paramètres, qui l'applique à l'imagette dans le domaine de Fourier et qui retourne la transformée de Fourier inverse. Affichez le résultat avec `imagesc`. On pourra afficher côte à côte l'image originale et l'image défloutée avec la commande `subplot` de la façon suivante :

```
figure(2)
subplot(1,2,1);
imagesc(I)
subplot(1,2,2);
imagesc(DeblurredImage);
```

3. Changez L et K , pour obtenir le meilleur résultat possible. Est-ce qu'on arrive à lire la plaque ?
4. Représentez avec la fonction `plot` le profil longitudinal du module de W . Que se passe-t-il quand K est trop faible ? Trop grand ?
5. Calculez la transformée de Fourier inverse de W . Appelons-la w . Représentez avec la fonction `plot` la section longitudinale de w (Il pourra être judicieux de visualiser d'abord w avec `imagesc`). Interprétez l'application du filtre de Wiener en terme de convolution avec w . Que se passe-t-il quand K est trop faible ? Trop grand ?

<p>Le compte-rendu est à soumettre sur la page du cours sur Educnet avant le 28 octobre 2014.</p>
