



Année universitaire

2019 - 2020

Majeure IMI – Partie 3 – 5ETI

**Compression et techniques avancées en
image**

TP de Tatouage d'images

Eric Van Reeth

1 Organisation du TP

1.1 Objectif

Après avoir vu en cours le processus d'insertion d'un tatouage dans le domaine de la DCT, il s'agira lors de ce TP d'insérer un **tatouage invisible et robuste dans l'espace de Fourier**. Par la suite, il s'agira d'implémenter une fonction de décodage capable de détecter la présence (ou non) du tatouage attendu, et d'évaluer la robustesse de la méthode vis-à-vis de différentes attaques.

1.2 Déroulement

Ce TP s'effectue en binôme par poste informatique (sous LINUX).

Le langage utilisé sera Python, en utilisant les bibliothèques `numpy`, `matplotlib` et `opencv` pour les opérations sur les images (doc en ligne d'opencv).

L'utilisation de l'IDE PyCharm est conseillée pour faciliter l'implémentation et le debug des codes.

Pour lancer PyCharm et la bonne version d'openCV, entrer dans un terminal les commandes suivantes :

```
export PYTHONPATH=/sync/IMI/python_libs  
/sync/Robotic/pycharm/bin/pycharm.sh &
```

L'image *cameraman.tif* fournie sur cpe-campus sera utilisée tout au long du TP.

1.3 Évaluation

Un compte-rendu sera demandé par binôme, dans lequel il n'est pas nécessaire d'inclure votre code. En revanche, il est fortement conseillé de décrire en détails la démarche algorithmique suivie, et d'inclure des graphes ou des images mettant en avant :

- le bon fonctionnement de votre algorithme
- le choix des paramètres clés
- les interprétations personnelles

Le compte-rendu sera rendu au **format pdf sur cpe-campus** dans un délai d'**une semaine après la séance**.

2 Contenu du TP

Question 1 : Étant donné que l'image utilisée est à coefficients réels, rappelez les propriétés de symétrie de sa transformée de Fourier.

Question 2 : Lire l'image *cameraman.tif* et calculer sa transformée de Fourier discrète à deux dimensions.

Vérifiez la propriété citée plus haut.

Affichez son module en échelle logarithmique.

Question 3 : Générez un tatouage T , sous forme de vecteur contenant $K = 1024$ valeurs aléatoires, de moyenne nulle et d'écart-type 1.

Question 4 : Comme dans le cas de l'insertion de tatouages dans le domaine de la DCT, le tatouage est inséré sur les coefficients de Fourier d'amplitude élevée.

1. Rappelez pourquoi il est intéressant d'insérer le tatouage sur les coefficients de forte amplitude
2. Rappelez où sont situés en général les coefficients de forte amplitude dans l'espace de Fourier. S'agit-il des hautes ou des basses fréquences ?
3. En respectant toujours la propriété de la transformée de Fourier citée plus haut, proposez une stratégie d'insertion du tatouage sur les coefficients de Fourier. Implémenter l'insertion de T dans une fonction à part, qui prendra au minimum en entrée :
 - la TFD-2D à tatouer
 - le tatouage T à insérer
 - le coefficient α de pondération du tatouage

Question 5 : On cherche maintenant à évaluer à quel point l'insertion du tatouage a dégradé l'image originale.

1. Pour la même instance du tatouage T , tracez l'évolution du PSNR résultant (en dB) en fonction de α :

$$PSNR(\alpha) = 10 \cdot \log_{10} \left(\frac{MAX_I^2}{MSE(\alpha)} \right) \quad (1)$$

Ici, MAX_I correspond au maximum d'intensité que peuvent prendre les pixels de l'image (255 sur 8 bits), et MSE l'erreur quadratique moyenne entre l'image originale et l'image tatouée : $MSE(\alpha) = \frac{1}{N} \sum_1^N (I[i] - I_{T\alpha}[i])^2$.

2. Étant donné que l'on cherche à insérer un tatouage invisible, déterminer en justifiant votre réponse, une valeur de α qu'il vous paraît pertinent d'utiliser.

Question 6 : En s'inspirant de l'exemple du tatouage sur la DCT, implémentez l'étape de détection du tatouage dans une fonction à part.

Cette fonction prendra en entrée (entre autres) un seuil permettant de décider si le tatouage détecté correspond à T .

Décrivez la démarche suivie et les points clés de l'implémentation.

Question 7 : Vérifier le bon fonctionnement de la détection du tatouage T , en insérant en entrée de votre fonction une image contenant un autre tatouage T_b (et/ou l'image sans tatouage).

Question 8 : Vérifier la robustesse de votre méthode d'insertion vis-à-vis de plusieurs types d'attaques de votre choix (compression/décompression, rotation, translation, opération sur l'histogramme, ...). Analyser et discuter ces résultats. Conclure sur la méthode implémentée.

3 Aide Python

3.1 Commandes utiles

- Lire une image : `I = plt.imread('cameraman.tif')`
- TFD-2D : `np.fft.fft2()`

- Recentrage de la TFD-2D : `np.fft.fftshift()`
- Génération d'un vecteur contenant des valeurs aléatoires : `np.random.randn()`

3.2 Affichage d'une image

L'affichage d'images sera fait avec la librairie `matplotlib.pyplot` importée en tant que `plt` :

```
plt.figure()
plt.imshow(img, cmap='gray') #colormap binary
plt.title('My title')
plt.show()
```

3.3 Écriture d'une image

La fonction suivante permet d'enregistrer l'image courante (avec des options sur la qualité de compression jpeg notamment) :

```
plt.savefig()
```

3.4 Affichage en « subplot »

L'affichage de 3 images dans la même figure est effectué grâce à la commande `subplot` :

```
plt.subplot(131)
plt.imshow(img1, 'binary') #colormap 'binary'
plt.title('Thresholded Image')
plt.subplot(132)
plt.imshow(img2, 'gray') #colormap 'gray'
plt.title('Distance Transform')
plt.subplot(133)
plt.imshow(img3, 'jet') #colormap 'jet'
plt.title('Labels')
plt.show()
```