

**Analyse par ondelettes****Travaux pratiques****Feuille 2 :****Transformée de Fourier****Exercice 1 Prise en main de Megawave**

1. Configuration pour l'utilisation de Megawave (à faire une seule fois avant la première utilisation de Megawave) :
  - Modification de l'environnement du compte utilisateur :  
*ouvrir une Konsole*  
Lancer la commande :  
`source /usr/local/megawave2/sys/lib/ix86/.profile_megawave2`
  - Créer votre compte Megawave2 avec la commande :  
*mwnewuser*
2. Téléchargement et compilation des programmes Megawave :
  - Télécharger (et sauvegarder dans votre `my_megawave2/src`) le fichier `tp2.tgz`. Il se trouve sur la page <http://www.math.univ-toulouse.fr/~fmalgouy/> rubrique "enseignement", puis "Analyse par ondelettes".
  - Aller dans votre compte Megawave  
`cd $MY_MEGAWAVE2/src/`
  - Décompresser l'archive  
`tar xvzf tp2.tgz`
  - Compiler les modules Megawave  
`cmw2_all TP2/`
3. Pour voir le code correspondant aux algorithmes que vous utiliserez, il faut ouvrir les fichiers `C` :  
`gedit TP2/*.c &`
4. Préparation des résultats :
  - création d'un répertoire :  
`mkdir resultats_tp2`
  - déplacement de l'image sur laquelle on fera les expériences :  
`cp resultats_tp1/barbara.gif resultats_tp2/`
  - Changement de répertoire :  
`cd resultats_tp2`
  - Lancement d'une commande Megawave :  
`voir_base_Fourier 64 10 30 resultat`
  - Visualisation du résultat :  
`xv resultat &`
5. Utilisation de scripts pour gagner du temps et pouvoir refaire un exercice :
  - Ouverture du fichier contenant le script `bash`  
`gedit correction_tp2.sh &`

- Écriture de l'entête du fichier *bash*

Écrire sur la première ligne du fichier :

```
#!/bin/bash
```

- Utilisation du fichier *bash* à l'aide de copier-coller

Faire un copier-coller, depuis le terminal vers le fichier *bash* des lignes :

```
voir_base_Fourier 64 10 30 resultat
```

```
xv resultat &
```

Vous pouvez continuer la suite du TP en copiant/éditant les lignes de commandes dans le fichier *bash*. Pour exécuter les commandes, il vous suffit de faire un copier-coller des lignes de commande à exécuter dans le terminal. Ainsi, vous retrouverez votre travail (avec les bonnes lignes de commandes) lors de votre prochaine séance de travail et au moment de la rédaction de votre compte rendu de TP.

**NB :** Nous utilisons le fichier *bash* ci-dessus de manière extrêmement simpliste et minimaliste. Une utilisation plus performante de *bash* permettrait de mettre en place des expériences plus sophistiquées.

- Pour lancer le fichier *bash* dans le terminal, il faut modifier ses droits :

```
chmod 700 correction_tp2.sh
```

Vous pourrez ensuite l'exécuter :

```
./correction_tp2.sh
```

- Vous gagnerez du temps en faisant des copier-coller des test *if* et des boucles *for* de votre script *correction\_tp1.sh*.

## Exercice 2 Visualisation de la base de Fourier

La transformée de Fourier d'un élément de  $\mathbb{R}^{N^2}$  prend la forme

$$\widehat{w}_{k,l} = \sum_{m,n=0}^{N-1} w_{m,n} e^{-2i\pi \frac{km+ln}{N}},$$

pour tout  $(k,l) \in \{0, \dots, N-1\}^2$  et son inverse prend la forme

$$w_{m,n} = \frac{1}{N^2} \sum_{k,l=0}^{N-1} \widehat{w}_{k,l} e^{2i\pi \frac{km+ln}{N}}, \quad (1)$$

pour tout  $(m,n) \in \{0, \dots, N-1\}^2$ .

On rappelle aussi que les éléments de  $\mathbb{R}^{N^2}$  et leur transformée de Fourier sont définis sur  $\mathbb{Z}^2$  par périodisation, c'est à dire

$$w_{m+t_1N, n+t_2N} = w_{m,n}, \text{ pour } (m,n) \in \{0, \dots, N-1\}^2 \text{ et } (t_1, t_2) \in \mathbb{Z}^2,$$

$$\widehat{w}_{k+t_1N, l+t_2N} = \widehat{w}_{k,l}, \text{ pour } (k,l) \in \{0, \dots, N-1\}^2 \text{ et } (t_1, t_2) \in \mathbb{Z}^2.$$

1. Vérifier que si, pour tout  $(m,n) \in \{0, \dots, N-1\}^2$ ,  $w_{m,n} \in \mathbb{R}$ , on a alors, pour tout  $(k,l) \in \{0, \dots, N-1\}^2$

$$\widehat{w}_{-k,-l} = \overline{\widehat{w}_{k,l}},$$

où  $\bar{z}$  désigne le nombre complexe conjugué de  $z \in \mathbb{C}$ .

2. Dédurre de (1) et de la question précédente que, si  $w \in \mathbb{R}^{N^2}$ , pour tout  $(m,n) \in \{0, \dots, N-1\}^2$ ,

$$w_{m,n} = \frac{1}{N^2} \sum_{k,l=0}^{N-1} \mathcal{R}e(\widehat{w}_{k,l}) \cos\left(2\pi \frac{km+ln}{N}\right) - \text{Im}(\widehat{w}_{k,l}) \sin\left(2\pi \frac{km+ln}{N}\right),$$

où  $\mathcal{R}e(\cdot)$  désigne la partie réelle et  $\text{Im}(\cdot)$  la partie imaginaire d'un nombre complexe.

Ainsi, la transformée de Fourier exprime une image comme une somme de cosinus et de sinus. Le but des questions suivantes est de visualiser les fonctions  $\cos\left(2\pi \frac{km+ln}{N}\right)$  (les fonctions  $\sin\left(2\pi \frac{km+ln}{N}\right)$  sont similaires).

3. Décrire le module *voir\_base\_Fourier*.

4. Utiliser le module *voir\_base\_Fourier* pour visualiser les éléments de la base de Fourier correspondant aux paramètres

$$(N, k, l) = (64, 4, 2), (64, 14, 7), (64, 24, 12), (64, 40, 52), (64, 50, 57), (64, 60, 62).$$

5. Faire un unique schéma représentant les fréquences de la question précédente. Pour quelles fréquences l'image oscille t'elle le plus? Où sont les hautes fréquences?
6. Utiliser le module *voir\_base\_Fourier* pour visualiser les éléments de la base de Fourier correspondant aux paramètres

$$(N, k, l) = (64, 20, 0), (64, 19, 6), (64, 18, 9), (64, 17, 11), (64, 14, 12) \\ (64, 0, 20), (64, 6, 19), (64, 9, 18), (64, 11, 17), (64, 12, 14)$$

7. Faire un unique schéma représentant les fréquences ci-dessus. Que dire de la "direction" des éléments de la base de Fourier?

### Exercice 3 Compression

Lorsque l'on considère une façon de représenter les images (ici on les représente par leurs coefficients de Fourier), l'une des propriétés très importante est la possibilité de représenter (ou d'approximer) une image avec peu de coefficients. Par exemple, en compression d'images, on ne garde (et ne code) que les grands coefficients, les autres étant mis à zéro. Ainsi, si l'on voulait compresser une image  $u \in \mathbb{R}^{N^2}$  avec la base de Fourier, on ne coderait que les coefficients dont le module est supérieur à une valeur  $\tau > 0$  donnée, pour obtenir une image  $u'$ . C'est à dire que  $u'$  est telle que, pour tout  $(k, l) \in \{0, \dots, N-1\}^2$

$$\widehat{u}'_{k,l} = \begin{cases} \widehat{u}_{k,l} & , \text{ si } (k, l) \in C(\tau), \\ 0 & , \text{ si } (k, l) \notin C(\tau), \end{cases}$$

avec

$$C(\tau) = \{(k, l) \in \{0, \dots, N-1\}^2, |\widehat{u}_{k,l}| \geq \tau\}.$$

1. Lire le module *Megawave seuille\_Fourier.c* et dire ce qu'il fait.
2. Utiliser le module *seuille\_Fourier* pour les valeurs de  $\tau = 1, 5, 25, 125$  pour construire les image  $u'$ , à partir de l'image *barbara.gif*. Visualiser ces images et visualiser les modules de leur transformée de Fourier (avec la fonction *voir\_fft*).

Dire comment se comporte  $\#C(\tau)$  (où  $\#$  désigne le cardinal d'un ensemble) et la qualité de l'image obtenue lorsque  $\tau$  augmente.

**Remarque :** La quantité  $\#C(\tau)$  est importante puisqu'elle correspond au nombre de coefficients que l'on doit coder pour représenter  $u'$ . Ainsi elle correspond (grosso modo) à la longueur du code.

3. Ajuster  $\tau$  de manière à approximer l'image *barbara.gif* en ne gardant que 10% de ses coefficients de Fourier. Visualiser et décrire le résultat et sa transformée de Fourier.
4. Montrer que si  $\tau \leq \tau'$

$$C(\tau') \subset C(\tau)$$

et que  $l^2(\tau) \leq l^2(\tau')$  avec, pour tout  $t \geq 0$

$$l^2(t) = \frac{1}{N^2} \left( \sum_{(k,l) \notin C(t)} |\widehat{u}_{k,l}|^2 \right)^{\frac{1}{2}}.$$

**Remarque :**  $l^2(\tau)$  quantifie l'erreur que l'on fait en approximant  $u$  par  $u'$ .

5. Pour une valeur de  $\tau \geq 0$ , on obtient une valeur  $\#C(\tau)$  et une valeur  $l^2(\tau)$ . Si l'on fait varier  $\tau$ , on obtient une courbe. Plus cette courbe décroît rapidement plus la façon de représenter les images (ici Fourier) est efficace en compression d'images (on a une erreur faible bien que l'on code peu de coefficients). Le but de cette question est de représenter cette courbe.

- (a) Lire le module Megawave *courbe\_seuillage\_Fourier.c* et dire ce qu'il fait.
- (b) Lancer les commandes :
 

```

courbe_seuillage_Fourier barbara.gif courbe
plot courbe &

```

 pour construire la courbe donnant l'erreur en fonction du nombre de coefficients que l'on doit coder.
- (c) Visualiser et commenter la courbe obtenue à la question précédente.

#### Exercice 4 Filtres passe bande, analyse d'images

Une autre utilité de la transformée de Fourier est de pouvoir faire des convolutions rapidement, même si le support du noyau de convolution est étendu.

Nous allons illustrer ceci à travers un problème d'analyse d'images. Dans ce problème, on veut faire une convolution avec un noyau  $h$  dont la transformée de Fourier est la somme de deux Gaussiennes :

$$\widehat{h}_{k,l} = e^{-\frac{(k-f_k)^2+(l-f_l)^2}{\sigma^2}} + e^{-\frac{(k+f_k)^2+(l+f_l)^2}{\sigma^2}},$$

pour  $(f_k, f_l) \in \{-\frac{N}{2}, \dots, \frac{N}{2} - 1\}^2$ ,  $\sigma > 0$  et pour tout  $(k, l) \in \{-\frac{N}{2}, \dots, \frac{N}{2} - 1\}^2$ .

1. Vérifier que  $h$  est bien à valeur réelle :  $h \in \mathbb{R}^{N^2}$ .
2. Vérifier que pour  $d \in \mathbb{R}^{N^2}$  définie par

$$d_{m,n} = \begin{cases} 1 & , \text{ si } m = n = \frac{N}{2} \\ 0 & , \text{ sinon,} \end{cases}$$

on a

$$(h * d)_{m,n} = h_{m-\frac{N}{2}, n-\frac{N}{2}}.$$

C'est à dire que  $h * d$  est l'image  $h$ , translatée par le vecteur  $(\frac{N}{2}, \frac{N}{2})$ .

3. Lire et dire ce que font les modules Megawave *dirac.c* et *filtre\_image.c*.
4. Lancer les commandes suivantes :
 

```

dirac 512 dirac
filtre_image dirac 126 -85 0.004 filtre
xv filtre &
voir_fft filtre

```

Que contient l'image *filtre* ainsi obtenue?

5. Lancer les commandes suivantes :
 

```

filtre_image barbara.gif 0 0 0.004 bas_bas
xv bas_bas
voir_fft bas_bas

```

 Refaire la même expérience en remplaçant 0.004 par 0.002 et 0.001. Que constatez vous sur les images ainsi obtenues? Leur transformées de Fourier? Que devient le support du noyau  $h$  correspondants?
6. Lancer les commandes suivantes :
 

```

filtre_image barbara.gif 126 -85 0.004 result
fabso result abs_result
xv abs_result
voir_fft result

```

Visualiser l'image *abs\_result* et décrire le contenu de l'image *barbara.gif* dans les zones où *abs\_result* est grande (les zones où *abs\_result* est claire).

7. Trouver les paramètre d'un filtre  $h$  permettant de détecter la partie pendante du foulard de *barbara*.

8. Trouver les commandes pour visualiser le noyau de convolution  $h$  correspondant à l'expérience de la question précédente. Visualiser le noyau  $h$  et expliquer le résultat de la question précédente.

### Exercice 5 Zoom numérique

On rappelle que la formule de Poisson donne la transformée de Fourier d'une image  $u$  définie sur  $\{0, \dots, N-1\}^2$  et obtenue en échantillonnant une image  $w$ , définie sur  $\{0, \dots, KN-1\}^2$ . On a ainsi

$$u_{m,n} = w_{Km,Kn}, \text{ pour } (m,n) \in \{0, \dots, N-1\}^2.$$

et dans le domaine de Fourier

$$\hat{u}_{k,l} = \frac{1}{K^2} \sum_{t_1=0}^{K-1} \sum_{t_2=0}^{K-1} \hat{w}_{k+t_1N, l+t_2N},$$

pour tout  $(k,l) \in \{0, \dots, N-1\}$ .

Ainsi, si le support de la transformée de Fourier de l'image  $w$  est inclu dans

$$\left\{ (k,l) \in \{0, \dots, KN-1\}, -\frac{N}{2} < k, l < \frac{N}{2} \right\}$$

on peut reconstruire  $w$  à partir de  $u$  en remplissant les hautes fréquences de  $u$  avec des 0 (voir Figure 1).

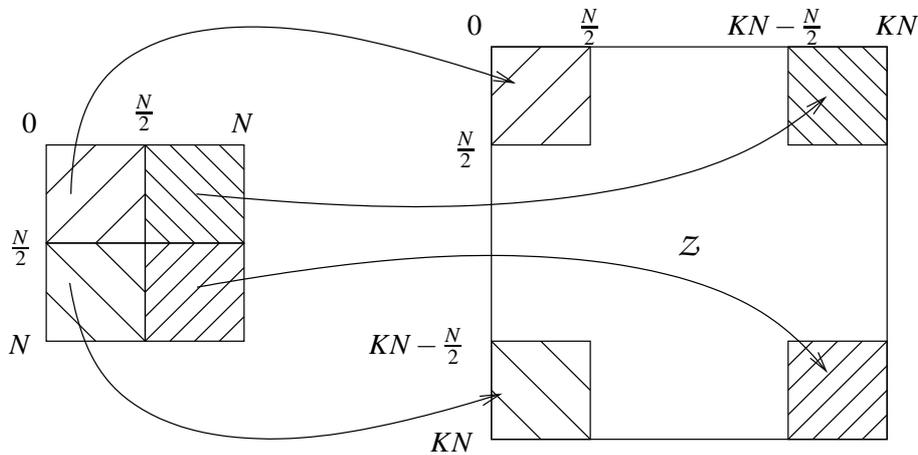


FIGURE 1 – Gauche : Transformée de Fourier de  $u$ ; droite : Transformée de Fourier de  $w$ . Les différentes zones de la transformée de Fourier de  $u$  sont copiées. La zone  $Z$  est remplie avec des zéros.

1. Vérifier que si le support de la transformée de Fourier de  $w \in \mathbb{R}^{(KN)^2}$  est inclu dans

$$\left\{ (k,l) \in \{0, \dots, KN-1\}, -\frac{N}{2} < k, l < \frac{N}{2} \right\}$$

on a

$$w = I \circ E(w),$$

où

$$\begin{aligned} E : \mathbb{R}^{(KN)^2} &\rightarrow \mathbb{R}^{N^2} \\ (w_{m,n})_{0 \leq m,n < KN} &\rightarrow (w_{Km,Kn})_{0 \leq m,n < N} \end{aligned}$$

et  $I = F^{-1} \circ Z \circ F$ , où  $F$  est la transformée de Fourier et  $F^{-1}$  son inverse et

$$\begin{aligned} Z : \mathbb{C}^{N^2} &\rightarrow \mathbb{C}^{(KN)^2} \\ (\hat{u}_{k,l})_{0 \leq k,l < N} &\rightarrow (\hat{w}_{k,l})_{0 \leq k,l < KN} \end{aligned}$$

avec

$$\widehat{w}_{k,l} = \begin{cases} K^2 \widehat{u}_{k,l} & , \text{ si } 0 \leq k < \frac{N}{2} \text{ et } 0 \leq l < \frac{N}{2} \\ K^2 \widehat{u}_{k,l-(KN-N)} & , \text{ si } 0 \leq k < \frac{N}{2} \text{ et } KN - \frac{N}{2} < l < KN \\ K^2 \widehat{u}_{k-(KN-N),l} & , \text{ si } KN - \frac{N}{2} < k < KN \text{ et } 0 \leq l < \frac{N}{2} \\ K^2 \widehat{u}_{k-(KN-N),l-(KN-N)} & , \text{ si } KN - \frac{N}{2} < k < KN \text{ et } KN - \frac{N}{2} < l < KN \\ 0 & , \text{ sinon.} \end{cases}$$

2. Décrire les étapes d'un algorithme permettant de zoomer une image, utilisant le principe de la question précédente.
3. Lire le contenu du fichier Megawave `zoom_sinus_cardinal.c` et dire ce que fait le module `zoom_sinus_cardinal`.
4. Zoomer une image extraite de `barbara.gif` :  

```
fextract -r barbara.gif img 10 230 128 128
zoom_sinus_cardinal img 4 zoom_img
```
5. Pour une comparaison, on considère le zoom consistant à "agrandir" chaque pixel de l'image  $u$  (appelé zoom par duplication ou réplique). Pour cette méthode de zoom, on reconstruit

$$w'_{Km+t_1,Kn+t_2} = u_{m,n},$$

pour tout  $(m,n) \in \{0, \dots, N-1\}^2$  et  $(t_1, t_2) \in \{0, \dots, K-1\}^2$ .

Lire le module Megawave `zoom_replication.c`. Que fait-il?

6. Appliquer le module `zoom_replication` à l'image `img` de la question 4. Comparer visuellement les résultats obtenus avec ceux de la question 4.
7. Visualiser visuellement les modules des transformées de Fourier des images obtenues aux questions 4 et 6. (Pour cela vous utiliserez le module Megawave `voir_fft`.) Décrire et expliquer ce que vous observez.

### Exercice 6 Discrétisation du gradient

Pour une fonction continûment différentiable  $u$  définie sur  $[0, N]^2$ , on sait que son gradient est défini par

$$\nabla u(x,y) = \begin{pmatrix} \partial_x u(x,y) \\ \partial_y u(x,y) \end{pmatrix} = \begin{pmatrix} \lim_{h \rightarrow 0} \frac{u(x+h,y) - u(x,y)}{h} \\ \lim_{h \rightarrow 0} \frac{u(x,y+h) - u(x,y)}{h} \end{pmatrix}. \quad (2)$$

Dans cet exercice, nous allons définir une façon de calculer un opérateur similaire au gradient pour des éléments de  $u \in \mathbb{R}^{N^2}$ . Notez que la difficulté réside dans le fait que dans (2), on ne peut pas faire tendre  $h$  vers 0 pour des fonctions définies sur une grille.

Pour cela, on considère  $K \geq 1$  et une image  $w \in \mathbb{R}^{(KN)^2}$ . Typiquement dans cet exercice,  $w$  sera une version sur-échantillonnée dans un rapport  $K$  de  $u$ . Comme d'habitude, on suppose  $w$  périodisée en dehors de son support.

On définit, les opérateurs suivants:

$$d_m : \begin{array}{ccc} \mathbb{R}^{(KN)^2} & \longrightarrow & \mathbb{R}^{(KN)^2} \\ (w_{m,n})_{0 \leq m,n < KN} & \longmapsto & (w_{m+1,n} - w_{m,n})_{0 \leq m,n < KN} \end{array}$$

et

$$d_n : \begin{array}{ccc} \mathbb{R}^{(KN)^2} & \longrightarrow & \mathbb{R}^{(KN)^2} \\ (w_{m,n})_{0 \leq m,n < KN} & \longmapsto & (w_{m,n+1} - w_{m,n})_{0 \leq m,n < KN} \end{array}$$

Le but de cet exercice est de construire un algorithme rapide pour calculer les dérivées partielles discrètes pour lesquels un algorithme lent est décrit dans la Table 1.

1. Exprimer les opérateurs  $d_m$  et  $d_n$  comme des convolution avec des noyaux  $h^m$  et  $h^n$  que vous définirez
2. Calculer les transformées de Fourier  $\widehat{h^m}$  et  $\widehat{h^n}$  de  $h^m$  et  $h^n$ .
3. On considère l'algorithme décrit dans la Table 1
  - (a) Pour chaque fréquence  $(k,l) \in \{0, \dots, N\}^2$ , exprimer la transformée de Fourier  $\widehat{\partial_m u}_{k,l}$  en fonction des transformées de Fourier  $\widehat{h^m}$  et  $\widehat{u}$ .

**Entrée :** une image  $u \in \mathbb{R}^{N^2}$

**Sortie :** les dérivées partielles discrètes de  $u$  :  $\partial_m u \in \mathbb{R}^{N^2}$  et  $\partial_n u \in \mathbb{R}^{N^2}$

- On calcule un suréchantillonnage  $w \in \mathbb{R}^{(KN)^2}$  de rapport  $K$  de  $u \in \mathbb{R}^{N^2}$  à l'aide de l'algorithme de l'exercice 5.
- On calcule  $d_m w \in \mathbb{R}^{(KN)^2}$  et  $d_n w \in \mathbb{R}^{(KN)^2}$ .
- On sous-échantillonne  $d_m w$  et  $d_n w$  dans un rapport  $K$  pour obtenir  $\partial_m u \in \mathbb{R}^{N^2}$  et  $\partial_n u \in \mathbb{R}^{N^2}$ .

TABLE 1 – Algorithme de calcul des dérivées partielles discrètes.

(b) En déduire une formule analogue pour  $\widehat{\partial_n u}_{k,l}$ , pour tout  $(k, l) \in \{0, \dots, N\}^2$ .

4. Déduire de la question précédente un algorithme permettant de calculer  $\partial_m u$  et  $\partial_n u$  sans manipuler d'image de taille  $KN \times KN$ .
5. Comparer la complexité de l'algorithme construit à la question 4 à la complexité de l'algorithme décrit dans la Table 1.
6. Implémenter l'algorithme construit à la question 4. (Pour cela, vous pourrez utiliser le canevas `calcul_diff_Fourier.c` qui vous est fourni dans `tp2.tgz`).
7. Visualiser et commenter les résultats obtenus pour les calculs des dérivées partielles, avec plusieurs valeurs de  $K$ . Quelle valeur de  $K$  préconiserez vous?