

## TP2 : Transformée de Fourier et Traitements fréquentiels

Auteur : Abdoul-Djalil OUSSEINI HAMZA

### INTRODUCTION

Suite au cours de traitement d'images sur la partie transformée de Fourier et traitements fréquentiels vu en classe, il nous a été demandé d'élaborer un rapport sur le travail effectué à cet effet.

Ce travail est agencé en deux grande partie : Une première partie qui traite de la transformée de Fourier en la calculant et calculant aussi sa transformée inverse et une seconde partie qui fait le traitements fréquentiels notamment les filtres passe-bas et les filtres passe-haut applicables au spectre d'une image donnée en entrée afin de réduire les bruits ou faire ressortir les contours présents dans cette dernière.

Dans la suite, nous décrivons et analysons chacune de ses parties.

### **Partie 1 : Transformée de Fourier**

Il a été conçus un programme nommé **FourierTrans.cpp** qui calcule la transformée de Fourier d'une image donnée en entrée par l'utilisateur. La compilation et l'exécution se fait de manière suivante :

- ➔ **Compilation** : Ouvrir un terminal et se positionner à l'aide de la commande **cd**, dans le répertoire contenant le fichier source et le makefile et compiler ce fichier au moyen de la commande **make** et un fichier exécutable va se créer.
- ➔ **Exécution** : Exécuter ce fichier nouvellement créé à partir de la commande **./FourierTrans** suivie du nom de l'image.

Nous illustrons avec des images différentes (images normales et bruitées) les expérimentations réalisées avec notre programme.

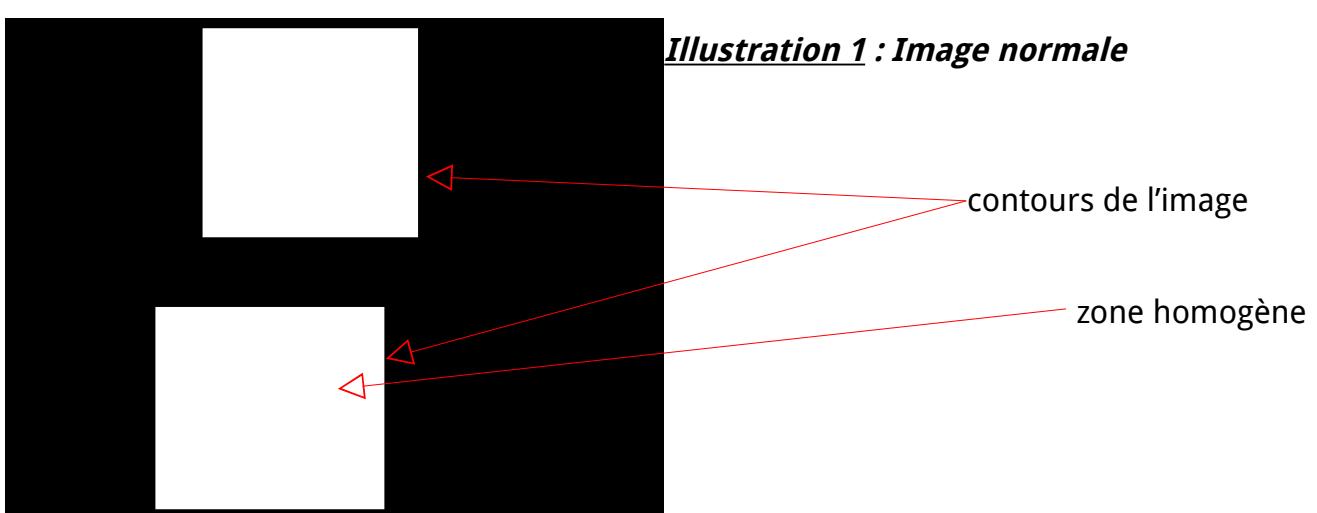


Figure 1 : Image Entrée

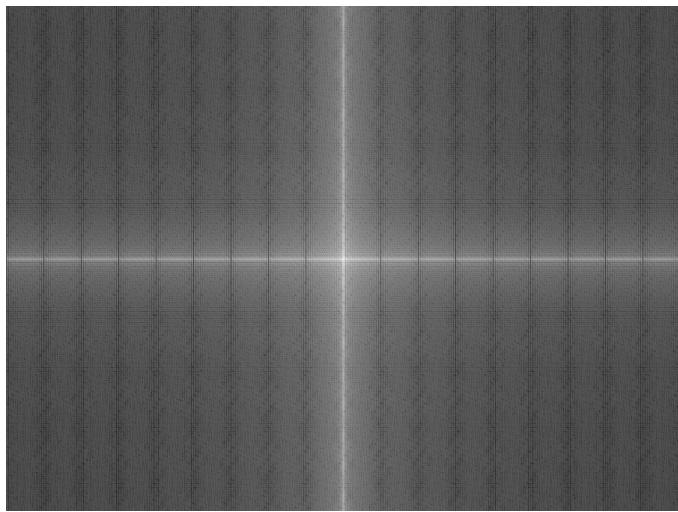


Figure 2 : Spectre de la transformée de Fourier

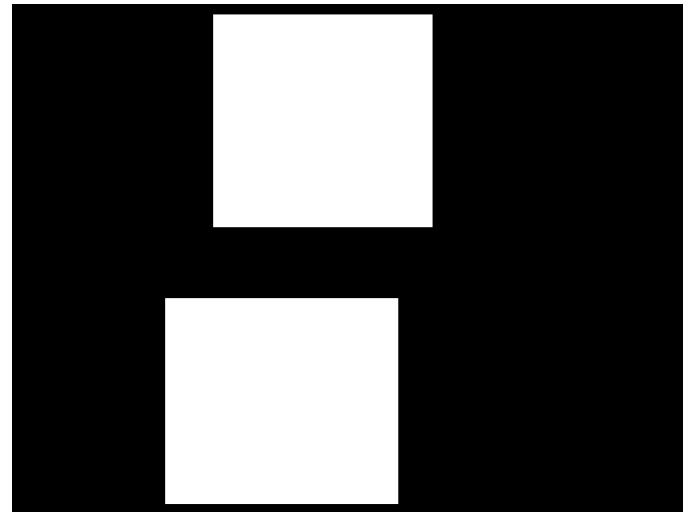


Figure 3 : Image Transformée-Inverse

Nous pouvons dégager quelques points provenant des résultats issus des images ci-dessus:

- L'image en entrée (originale) est la même que la transformée inverse. ceci indique que le spectre obtenu n'a subit aucun traitement suite à la transformation de Fourier.
- Il en ressort de l'analyse de la transformée de Fourier que le point blanc lumineux au centre de l'image correspond à l'origine du spectre et représente aussi la valeur moyenne des pixels de l'image.
- On peut remarquer aussi sur la transformée de Fourier que les composantes les plus influentes du domaine fréquentiel (points les plus brillants sur l'image) suivent la rotation géométrique des objets sur l'image.
- On peut voir sur la transformée de Fourier que l'image normale est constituée des basses et hautes fréquences, ainsi, les régions éloignées du centre correspondent aux hautes fréquences et celles plus près du centre correspondent aux basses fréquences.

**Illustration 2**: Image bruitée



Figure 4 : Image bruitée

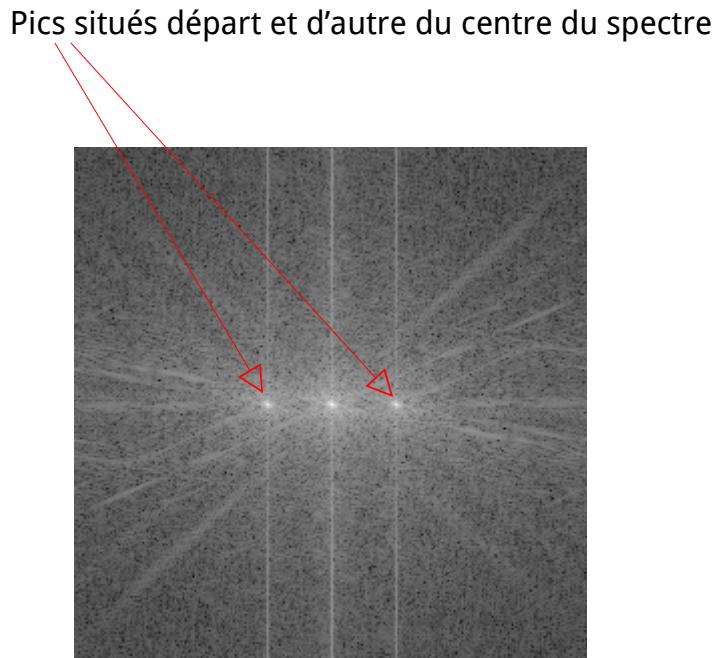


Figure 5 : Spectre de l'image bruitée



Figure 6 : Image Transformée de Fourier

Nous avons entrée une image bruitée afin de voir comment se comporte la Transformée de Fourier avec celle-ci. Les bruits sinusoïdaux présentent dans l'image font apparaître deux points pics au niveau de spectre de la transformée de Fourier, et le troisième pic correspond au centre du spectre et la moyenne des valeurs des pixels de l'image.

On remarque aussi que l'image obtenue après la transformée inverse est la même que celle entrée avant, aucun changement n'a été apporté.

La différence se remarque au niveau du spectre, les deux points situant sur les lignes horizontales sont les résultats du changement de l'intensité de l'image dans les zones spatiales.

Comme précédemment, Les zones correspondantes au basses fréquences et aux hautes fréquences demeurent les mêmes que pour **Illustration 1** avec les même caractéristiques rigoureusement identiques.

### **Partie 2 : Traitements fréquentiels**

Dans cette partie nous avons implémenter un programme qui répond aux exigences du TP. Le programme est dénommé **Traitements.cpp** endroit où se trouve les fonctions et **Filtres.cpp** qui sert de programme principal.

- **Compilation** : Ouvrir un terminal et se placer à l'aide de la commande **cd**, dans le répertoire contenant le fichier source et le makefile et compiler ce fichier au moyen de la commande **make** et un fichier exécutable va se générer.
- **Exécution** : Exécuter ce fichier nouvellement créer à partir de la commande **./Filtres** suivie du nom de l'image. Ensuite deux choix s'offrent à l'utilisateur. Le premier est d'appliquer un filtre passe-bas au spectre de l'image entrée et l'autre est d'appliquer un filtre passe-haut au spectre de la même image. Une fois son choix effectué, l'utilisateur entre la fréquence de coupure souhaitée pour le filtre sélectionné.

### **1<sup>er</sup> Cas : Filtre passe-bas**

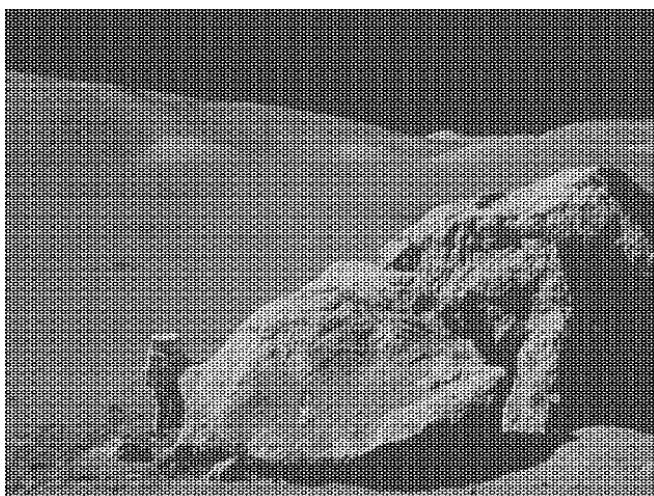


Figure 7 : Image originale

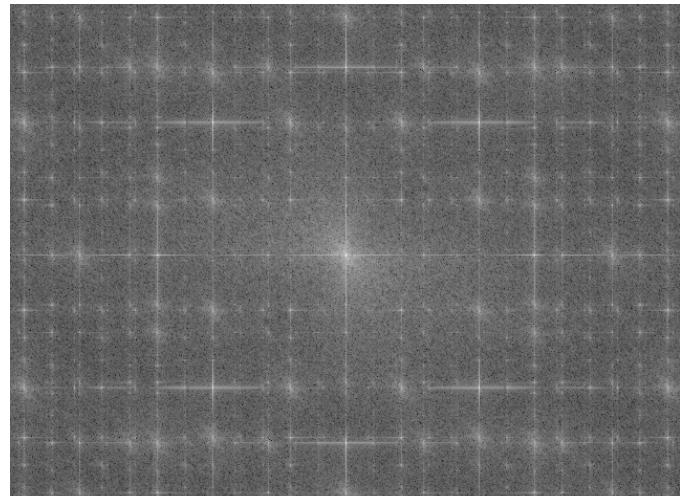


Figure 8 : Spectre de Fourier initial

- *Filtre passe-bas avec fréquence de coupure à 0,5*

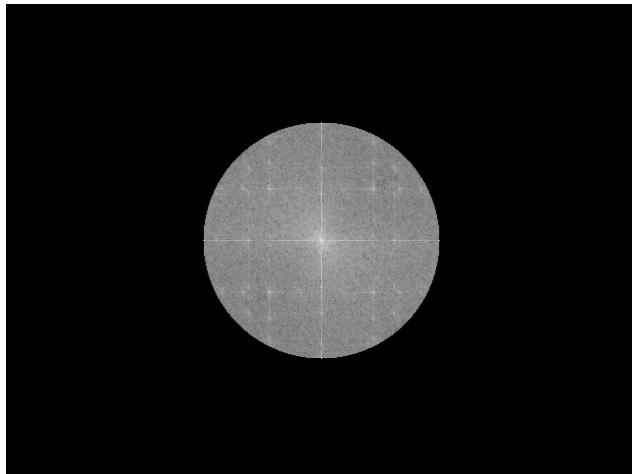


Figure 9 : Spectre filtré de l'image originale



Figure 10 : Image restituée du spectre filtré

- *Filtre passe-bas avec fréquence de coupure à 0,1*

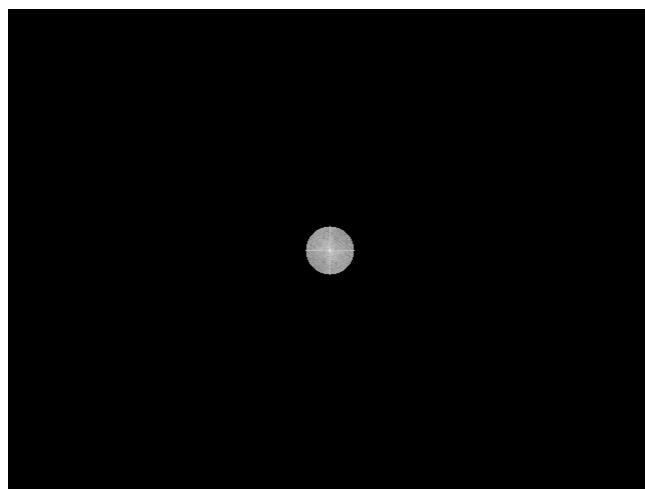


Figure 11 : Spectre filtré de l'image originale

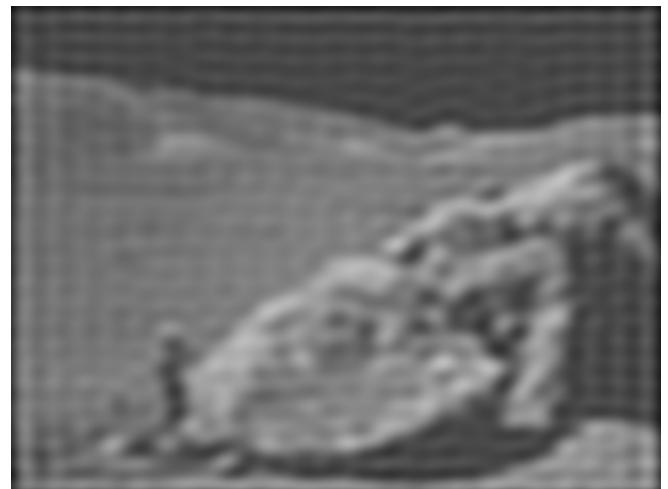


Figure 12 : Image restituée du spectre filtré

**Notons ici que la fréquence de coupure est une fréquence qu'on a définie au dessus de laquelle les autres fréquences sont annulées (filtre idéal).**

### Observations :

Dans cette première section (1<sup>er</sup> cas) nous avons mené nos expérimentations sur les filtre passe-bas, le filtre passe-bas diminue les bruits mais atténue les détails de l'image. Nous avons entré pour cela dans notre programme une image bruitée et nous l'avons fait passer par un filtre passe-bas avec des fréquences de coupures différentes (0,5 et 0,1) afin de vérifier la diminution des bruits et l'atténuation de l'image dans chaque cas. Les résultats obtenus après l'opération sont satisfaisants. Dans le premier cas, avec une fréquence de coupure à 0,5 (Fig. 10), on constate une faible diminution des bruits et l'image n'est pas assez floue par-rapport à l'image d'origine.

Dans le second cas, pour une fréquence de coupure à 0,1 (Fig. 12), les bruits sont quasiment absents et l'image semble être fortement atténuer (très floue).

Un filtre passe-bas atténue donc les hautes fréquences, le filtrage fait adoucir les détails et réduire les bruits granuleux.

### 2<sup>e</sup> Cas : Filtre passe-haut



Figure 13 : Image originale

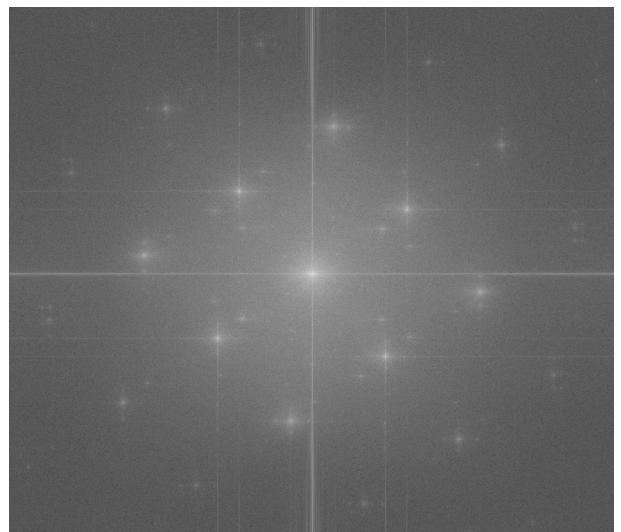


Figure 14 : Spectre de Fourier initial

- *Filtre passe-haut avec fréquence de coupure à 0,4*

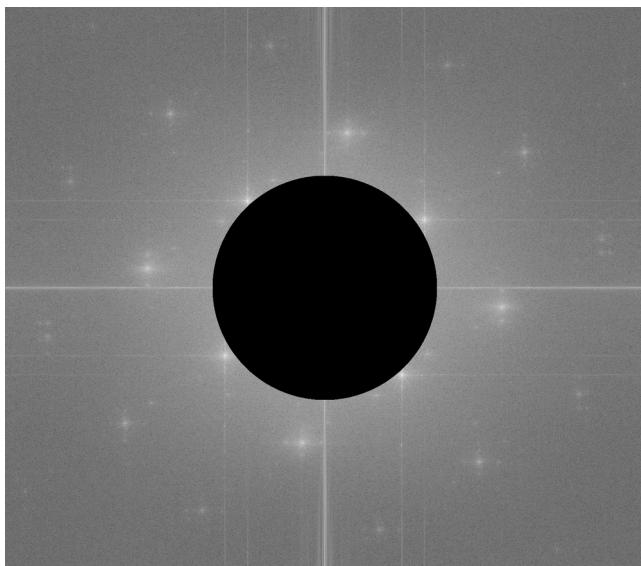


Figure 15 : Spectre filtré de l'image originale

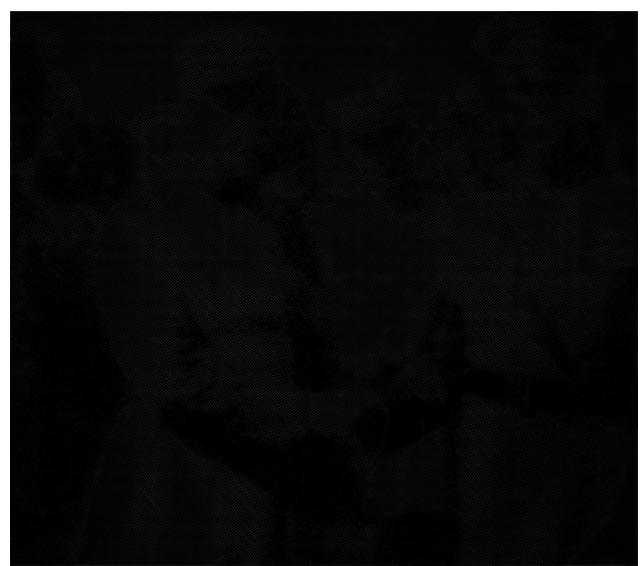


Figure 16 : Image restituée du spectre filtré

*Filtre passe-haut avec fréquence de coupure à 0,1*

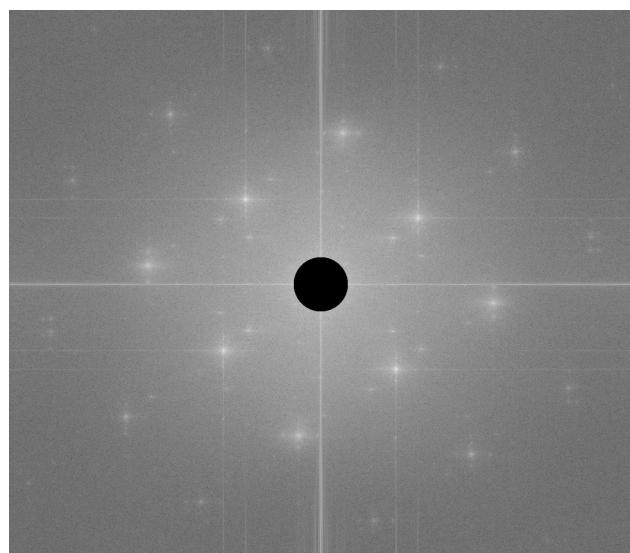


Figure 17 : Spectre filtré de l'image originale



Figure 18 : Image restituée du spectre filtré

#### Observations :

Les images ci-dessus sont issues de l'expérimentation sur le filtre passe-haut. Le filtre passe-haut accentue les contours et les détails de l'image, mais amplifie les bruits. Comme précédemment, après l'entrée d'une image, nous passons l'image au travers un filtre passe-haut avec des fréquences de coupures différentes (0,4 et 0,1) pour pouvoir observer les comportements du filtre face aux deux fréquences de coupures.

On remarque pour une fréquence de coupure à 0,4 (Fig.16), l'image est peu visible (contraste élevé) et les contours sont moins apparents alors que sur la Fig.18 les contours sont plus visibles mais les bruits ont été accentués. Cela voudrait dire que la transformée de Fourier inverse présente autant plus d'ondulations que la fréquence de coupures est petite. Cela entraîne un flou qui sera d'autant plus réduit que la fréquence de coupure est grande.

Un filtre passe haut favorise donc les hautes fréquences spatiales, comme les détails, et de ce fait, il améliore le contraste. Toutefois, il produit des effets secondaires comme par exemple l'augmentation du bruit dans les images.

### **CONCLUSION**

Au regard de ce qui précède, nous pouvons dire que ce TP 2 nous a permis de mieux comprendre les notions sur la transformée de Fourier et les notions sur les traitements fréquentiels.

Il nous a permis de mettre en œuvre un programme répondant à ces deux questions. La première partie du programme prend une image en argument ensuite calcule et affiche sa transformée de Fourier (détermine le spectre d'une image sur le plan fréquentiel) et en même temps calcule et affiche sa transformée inverse qui permet de restituer une image à partir de son spectre. La seconde partie introduit les filtres, notamment les filtres passe-bas et passe-haut. Les filtres passe-bas diminuent les bruits dans une image mais atténuent ses détails alors que les filtres passe-hauts font plutôt apparaître les contours dans une image mais augmentent les bruits dans celle-ci.