

# Travaux Pratiques Traitement Numérique du Signal

Alexis TREILLES  
Téo FREJAVILLE

Mars 2023



Encadrant :  
Marc Donias

---

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Étude des fréquences</b>	<b>3</b>
<b>3</b>	<b>Étude du filtre passe-bas</b>	<b>4</b>
<b>4</b>	<b>Étude du filtre coupe-bande</b>	<b>6</b>
<b>5</b>	<b>Solution finale pour filtrer l'image</b>	<b>8</b>
5.1	Observation du résultat du filtrage de la fréquence en $Fx = -0.1/fy + 0.4$ . . . . .	8
5.2	Recherche des fréquences visibles sur l'image . . . . .	8
<b>6</b>	<b>Code complet :</b>	<b>11</b>

---

# 1 Introduction

Lors de ce travail pratique, nous avons exploré la représentation fréquentielle des images pour analyser leurs différentes composantes spectrales, y compris le bruit potentiel. L'objectif principal était d'identifier, dans le domaine des fréquences, la signature d'une trame visible sur une image donnée, `monument.bmp`, et de concevoir un filtre linéaire RIF adapté pour atténuer cette trame. Notre approche s'est articulée autour de la création d'un filtre "coupe-bande" de type gaussien en trois étapes distinctes. Tout le processus, y compris la création du filtre final ainsi que des formes intermédiaires, a été réalisé exclusivement dans le domaine spatial. Nous avons débuté par la conception d'un filtre "passe-bas" basé sur une fonction gaussienne bidimensionnelle centrée et isotrope. Ensuite, ce filtre a été modulé par un signal sinusoïdal afin de le centrer sur la fréquence de la trame, le transformant ainsi en un filtre "passe-bande". Enfin, une dernière opération effectuée dans le domaine spatial a permis de convertir ce filtre en un filtre "coupe-bande", répondant ainsi à notre objectif initial.

## 2 Étude des fréquences

Dans cette partie du compte rendu, nous aborderons les différentes fréquences composant notre trame. Nous avons comme trame à analyser l'image suivante :



Figure 1: Image à filtrer

Pour faire notre analyse, nous avons utilisé la transformée de Fourier en 2 dimensions, puis nous l'avons affichée à l'aide d'un spectre 2D. Ainsi, nous avons obtenu :

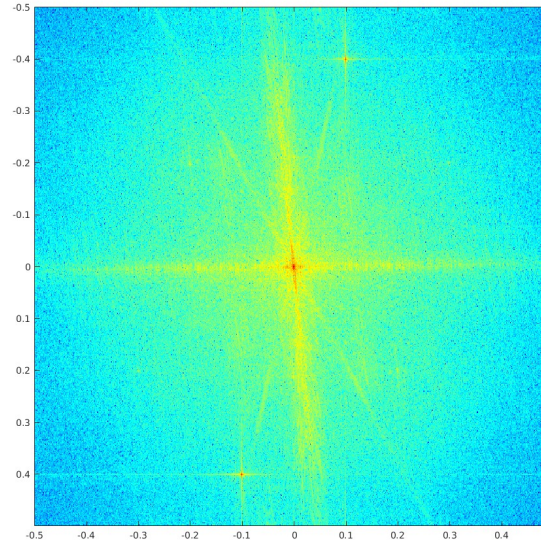


Figure 2: Spectre 2D de l'image originale

Nous constatons que le spectre possède deux raies excentrées. La première de coordonnées normalisées  $f_{0x} = -0.1$  et  $f_{0y} = 0.4$  et la seconde de coordonnées  $f_{0x} = 0.1$  et  $f_{0y} = -0.4$ . L'objectif est donc d'éliminer ces raies. Nous obtenons donc par un filtre passe-bas.

### 3 Étude du filtre passe-bas

Nous avons défini notre filtre passe-bas avec une gaussienne de la forme suivante :

$$G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

Afin que notre filtre gaussien converge et prenne un maximum de valeur, nous avons utilisé la règle des  $3\sigma$  qui nous permet de prendre 99,7 % des valeurs. Voici le filtre :

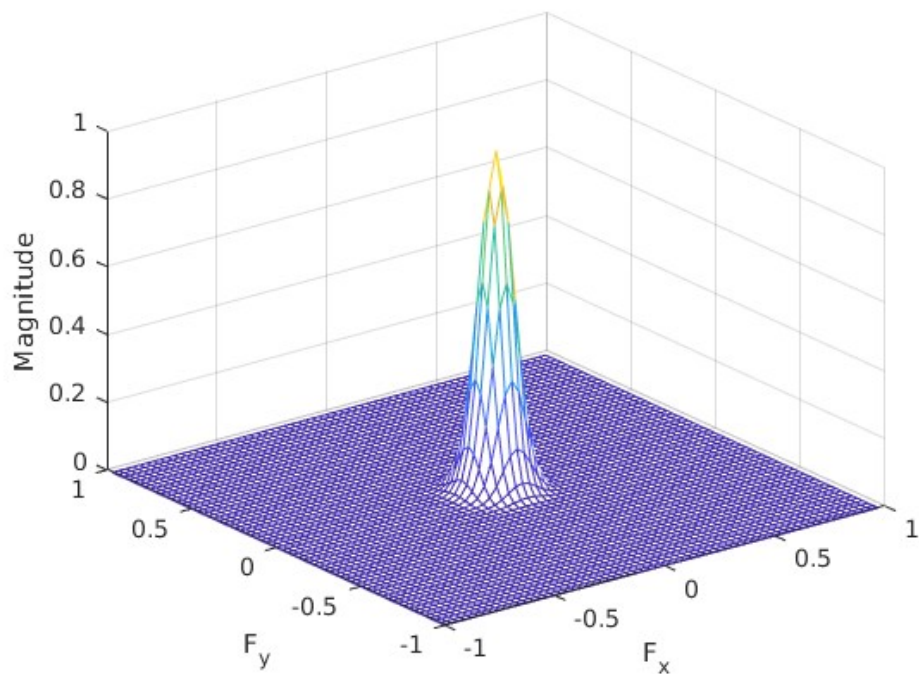


Figure 3: Filtre passe-bas gaussien

Une fois notre filtre gaussien défini, nous l'avons convolué avec notre image et ainsi nous avons obtenu :



Figure 4: Résultat de la convolution

Nous constatons que la convolution permet de rendre l'image plus nette en mettant en évidence les contours. En revanche, un bruit est présent sur toute l'image. Nous devons le filtrer afin de rendre l'image la plus nette possible.

---

## 4 Étude du filtre coupe-bande

À partir du passe-bande précédent, on doit simplement obtenir un filtre coupe-bande, car le passe-bande appliqué à l'image nous retourne uniquement le bruit. Le coupe-bande nous permet donc de récupérer l'image sans le bruit. On a procédé de la manière suivante :

On crée une matrice vide et on y soustrait simplement le résultat de notre filtre passe-bande. On obtient le résultat suivant :

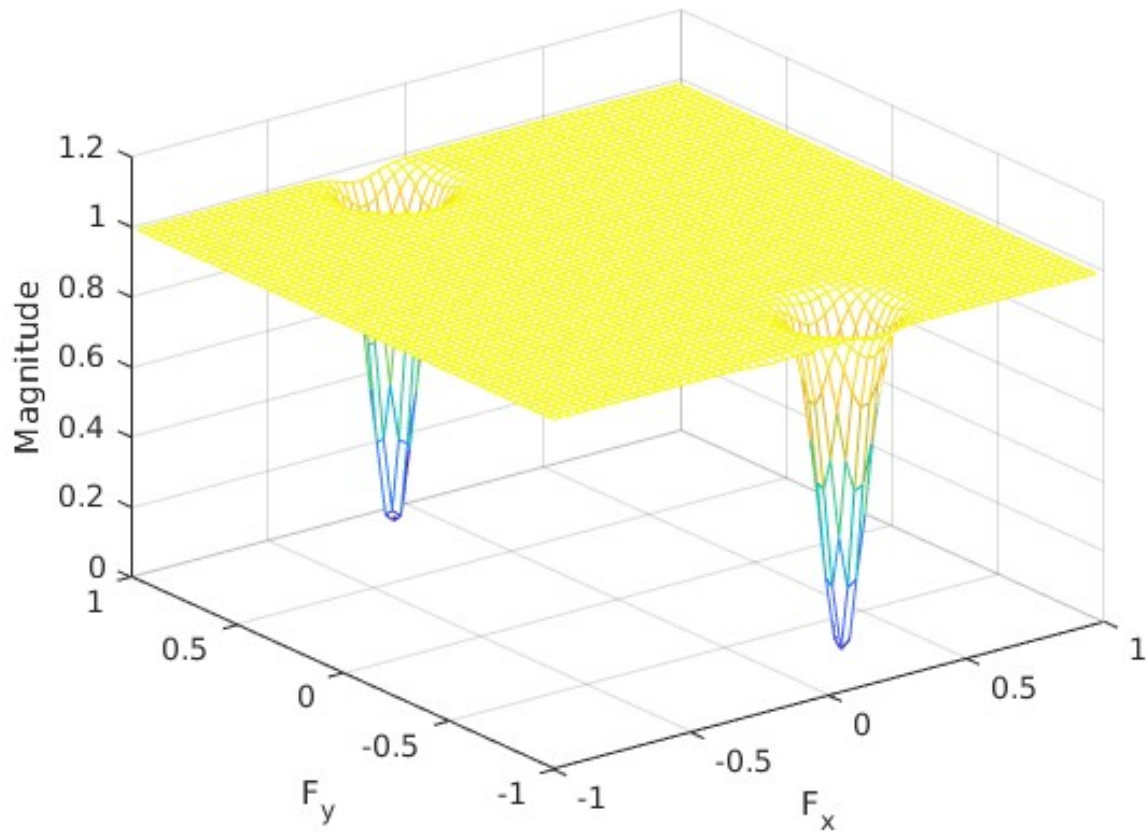


Figure 5: Spectre de l'image filtrée

On a donc l'image suivante :



Figure 6: Image avec une fréquence supprimée



---

## 5 Solution finale pour filtrer l'image

### 5.1 Observation du résultat du filtrage de la fréquence en $Fx = -0.1/fy + 0.4$

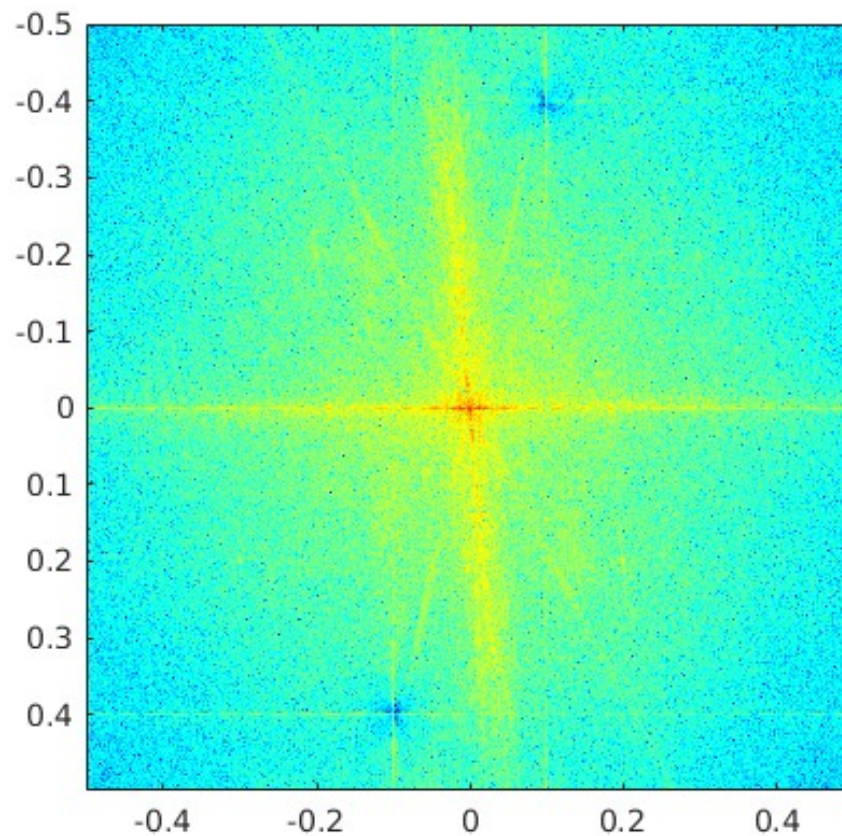


Figure 7: Spectre de l'image filtrée

### 5.2 Recherche des fréquences visibles sur l'image

Ici, on observe un motif autre, notamment sur le toit et les murs, plus globalement sur les surfaces uniformes :





Figure 8: Observations des fréquences secondaires

Or, on n'arrive pas à observer cette fréquence sur le spectre de l'image.

Notre démarche pour identifier ces fréquences a été de supposer qu'elles devaient apparaître sur le spectre mais n'étaient pas visibles car l'échelle d'affichage du spectre de couleurs ne le permettait pas. La solution correcte aurait été de changer les couleurs d'affichage du spectre ; cependant, il nous a paru plus simple d'atténuer les basses fréquences :

$$G2 = \text{vide} - G - G_{xy}; \quad (1)$$

Avec : vide notre matrice de zéros,  $G$  notre filtre à la fréquence traitée initialement et  $G_{xy}$  notre filtre passe-bas gaussien de base.

Ceci a pour effet de changer la plage de couleur de l'affichage de nos fréquences. On a donc pu observer les résultats suivants sur le spectre :

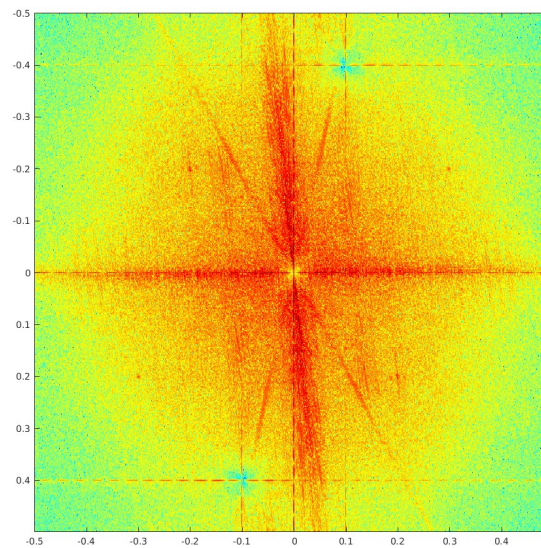


Figure 9: Observation des fréquences secondaires

---

Grâce à ce "remapping" des couleurs, on arrive à observer les fréquences qu'on observait sur l'image mais qui n'apparaissaient pas sur le spectre précédent.

On peut donc extraire :

1.  $fx_1 = 0.2$
2.  $fy_1 = 0.2$
3.  $fx_2 = -0.3$
4.  $fy_2 = 0.2$

On traite donc ces deux fréquences en même temps de cette manière :

$$G2 = \text{vide} - G - G2 - G3; \quad (2)$$

On obtient le résultat suivant :



Figure 10: Image avec toutes les fréquences parasites filtrées

On n'observe plus les motifs sur les surfaces uniformes.

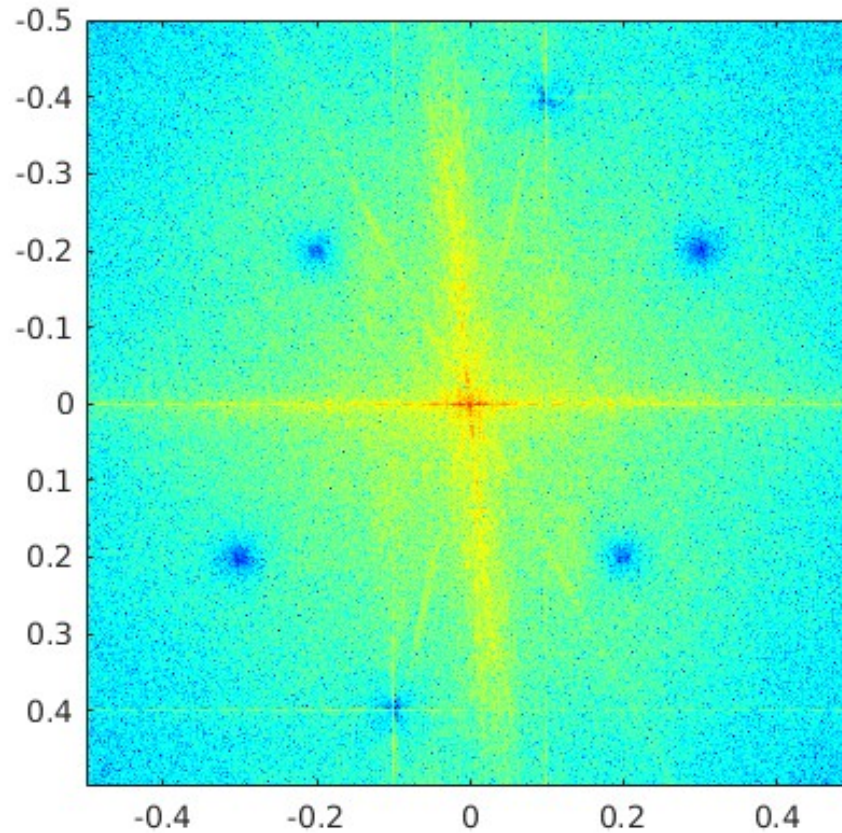


Figure 11: Spectre des fréquences après filtrage

On observe que le spectre est bien filtré sur les fréquences ciblées.

## 6 Code complet :

```

1 clear, close all
2 m=double(imread('monument.bmp'));
3 figure, imshow(uint8(m))
4
5
6 [x,y] = size(m);
7 B = log10(abs(fft2(m)));
8 fx= linspace(-0.5,0.5-1/x,x);
9 fy= linspace(-0.5,0.5-1/y,y);
10
11 %figure, imagesc(fx,fy,fftshift(B))
12 %axis image, colormap(jet)
13 %passe bas
14
15 sigma = 5;
16 %regle des 3 sigma
17 ech = ceil(3*sigma);
18
19 [x,y]=meshgrid(-ech:ech);
20 Gxy = (1/(2*pi*sigma^2))* exp(-(x.^2+y.^2)/(2*sigma^2));
21 figure, freqz2(Gxy)

```

```

22 %figure, surf(Gxy)
23
24 %Cr ation du 2 filtre G passe-bande
25 f0x= -0.1;
26 f0y= 0.4;
27 G=filtrage(f0x,f0y,x,y,Gxy);
28 G2=filtrage(0.2,0.2,x,y,Gxy);
29 G3=filtrage(-0.3,0.2,x,y,Gxy);
30 h1 = conv2(m,G);
31
32 %figure, imshow(uint8(h1))
33 %coupe bande a partir du passe bande
34 vide = zeros(size(G));
35 vide(ceil(size(G)/2),ceil(size(G)/2)) = 1;
36 G2 = vide-G-G2-G3;
37 %figure, freqz2(G2)
38 %figure, surf(G2)
39
40 h2 = conv2(m,G2);
41 figure, imshow(uint8(h2))
42 [x,y]=size(m);
43 B1 = log10(abs(fft2(h2)));
44 fx= linspace(-0.5,0.5-1/x,x);
45 fy= linspace(-0.5,0.5-1/y,y);
46
47 figure, imagesc(fx, fy, fftshift(B1));
48 axis image, colormap(jet)
49
50
51 function Gxy = filtrage(f0x,f0y,x,y, Gxy)
52     Gxy = 2*cos(2*pi*(f0x*x+f0y*y)).*Gxy;
53     %figure, freqz2(Gxy)
54     %figure, surf(Gxy);
55 end

```