



ELE8812 – Rapport de laboratoire 4

Analyse multirésolution

Julien ANTOINE
1813026

Maxime SCHMITT
1719088

23 mars 2016

1 Introduction

L'objectif de ce laboratoire est de se familiariser avec la transformée en ondelettes rapide et certaines de ses applications. On s'intéresse dans la première partie à son utilisation pour la compression d'images puis dans la seconde partie à son utilité pour la correction de défauts localisés. Tout au long de ce laboratoire, on utilisera différents types d'ondelettes afin de se sensibiliser sur l'effet de ce choix sur le résultat de la transformée.

2 Transformée en ondelettes rapide d'une image

2.1 Transformée directe et transformée inverse

On s'intéresse dans un premier temps à l'ondelette de Haar, les réponses impulsionnelles des filtres de codage-décodage en sous-bandes relatifs à celle-ci sont données dans la figure 1. On retrouve ensuite la Transformée en Ondelettes Rapide (TOR) de l'image *Lenna.tif* utilisant 3 niveaux d'échelle dans la figure 2. On reconstruit ensuite l'image en effectuant la TOR inverse, le résultat est affiché en vis-à-vis avec l'image originale dans la figure 3. On constate qu'on ne distingue pas à l'oeil nu de différences entre l'image originale et celle reconstruite. De plus, une vérification numérique sur la différence entre ces deux images (maximum de la valeur absolue de la différence pixel par pixel) révèle qu'elles sont bien identiques.

Ces mêmes opérations, effectuées cette fois avec l'ondelette, db4 sont présentées sur les figures 4, 5 et 6.

On remarque des différences entre les TOR utilisant l'ondelette de Haar (figure 2) et celles utilisant l'ondelette db4 (figure 5). Cette dernière semble conserver dans les coefficients de détail des détails de plus petite taille que ceux conservés dans le cas de l'ondelette de Haar. Cela est logique puisque le choix d'une fonction donnée pour l'ondelette correspond à un choix d'échelle correspondant : l'ondelette de Haar (db1) correspond à des détails "plus gros" que ceux de l'ondelette db4.

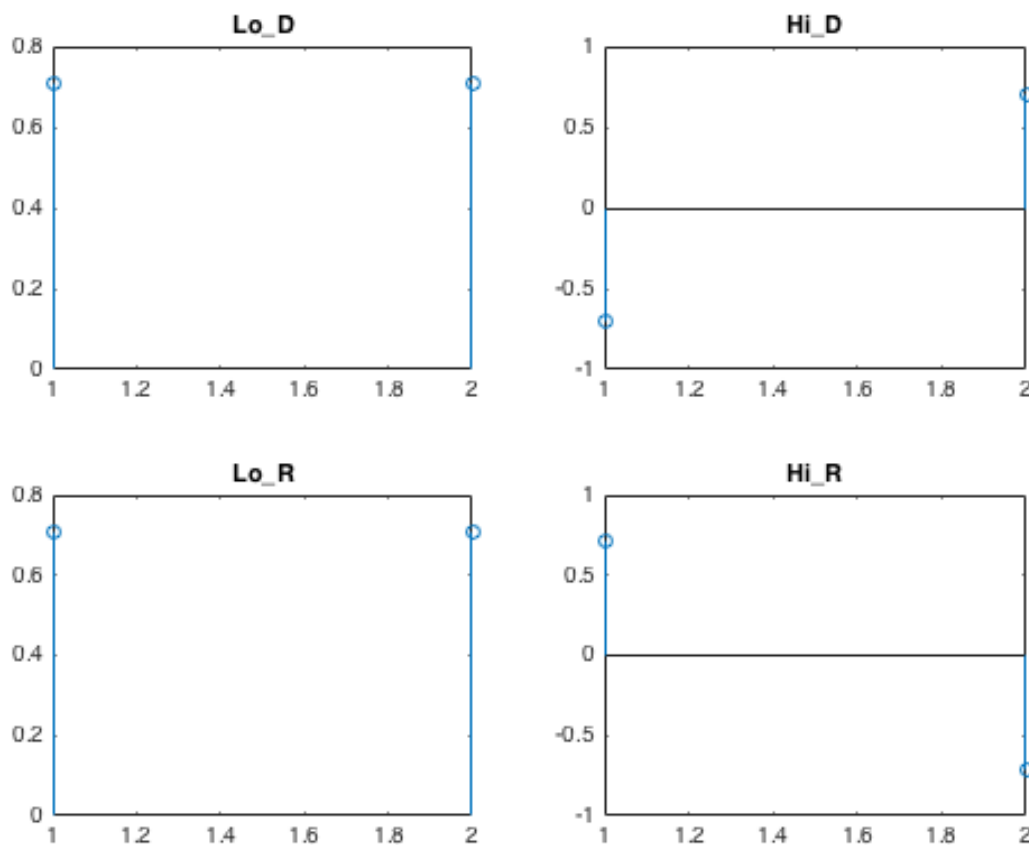


FIGURE 1 – Réponses impulsionnelles des filtres de codage-décodage en sous-bandes de l'ondelette de **Haar**

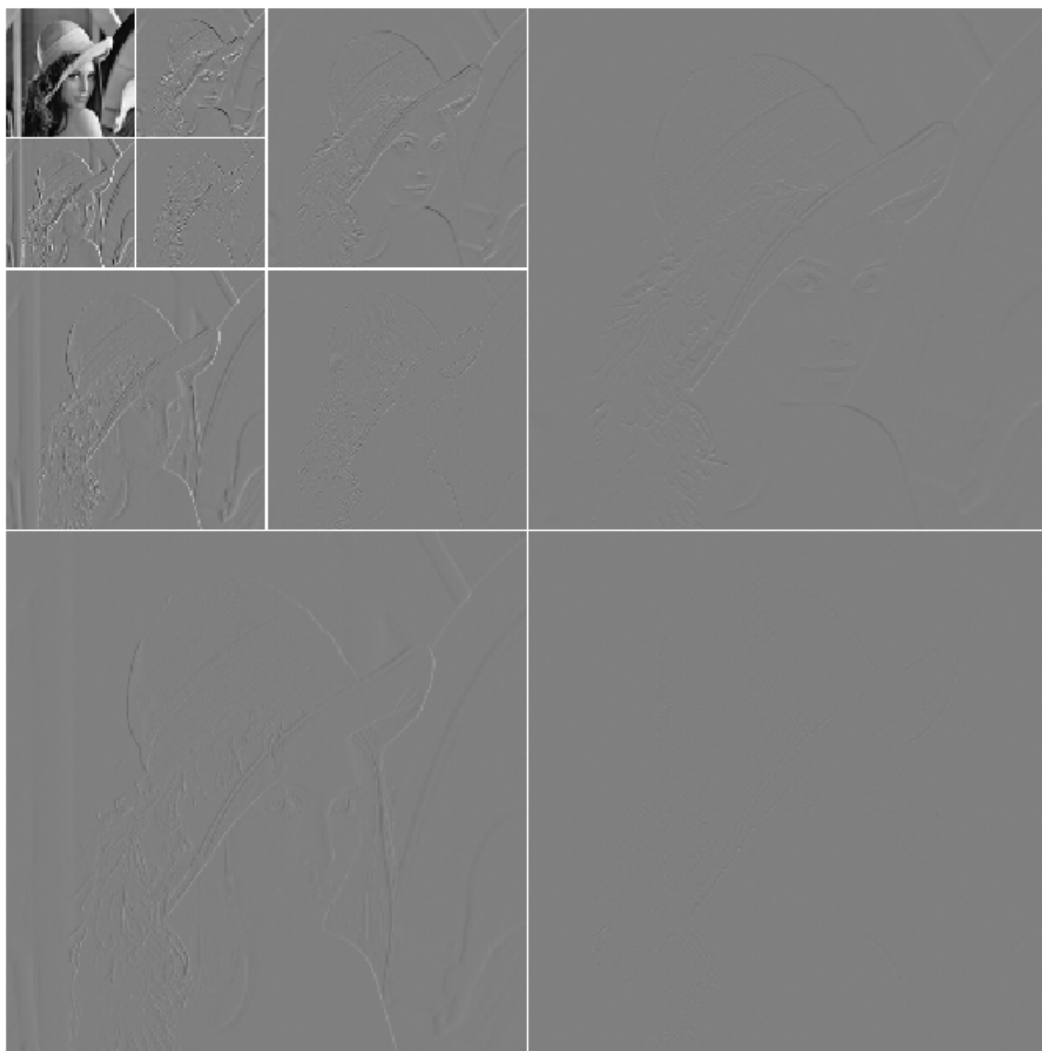


FIGURE 2 – Transformée en Ondelettes Rapide à 3 niveaux d'échelle de l'image *Lenna.tif* avec l'ondelette de **Haar**



FIGURE 3 – Comparaison de l'image originale (à gauche) et de l'image reconstruite avec l'ondelette de **Haar** (à droite)

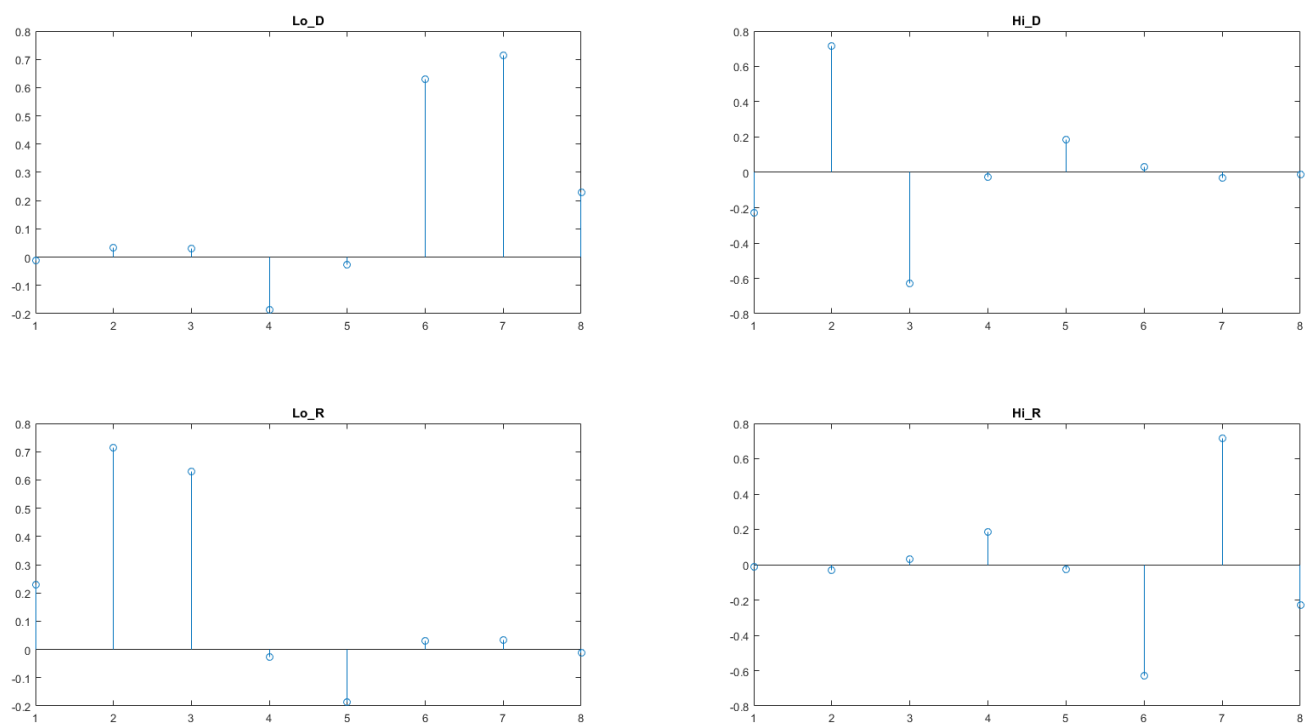


FIGURE 4 – Réponses impulsionnelles des filtres de codage-décodage en sous-bandes de l'ondelette de **Haar**

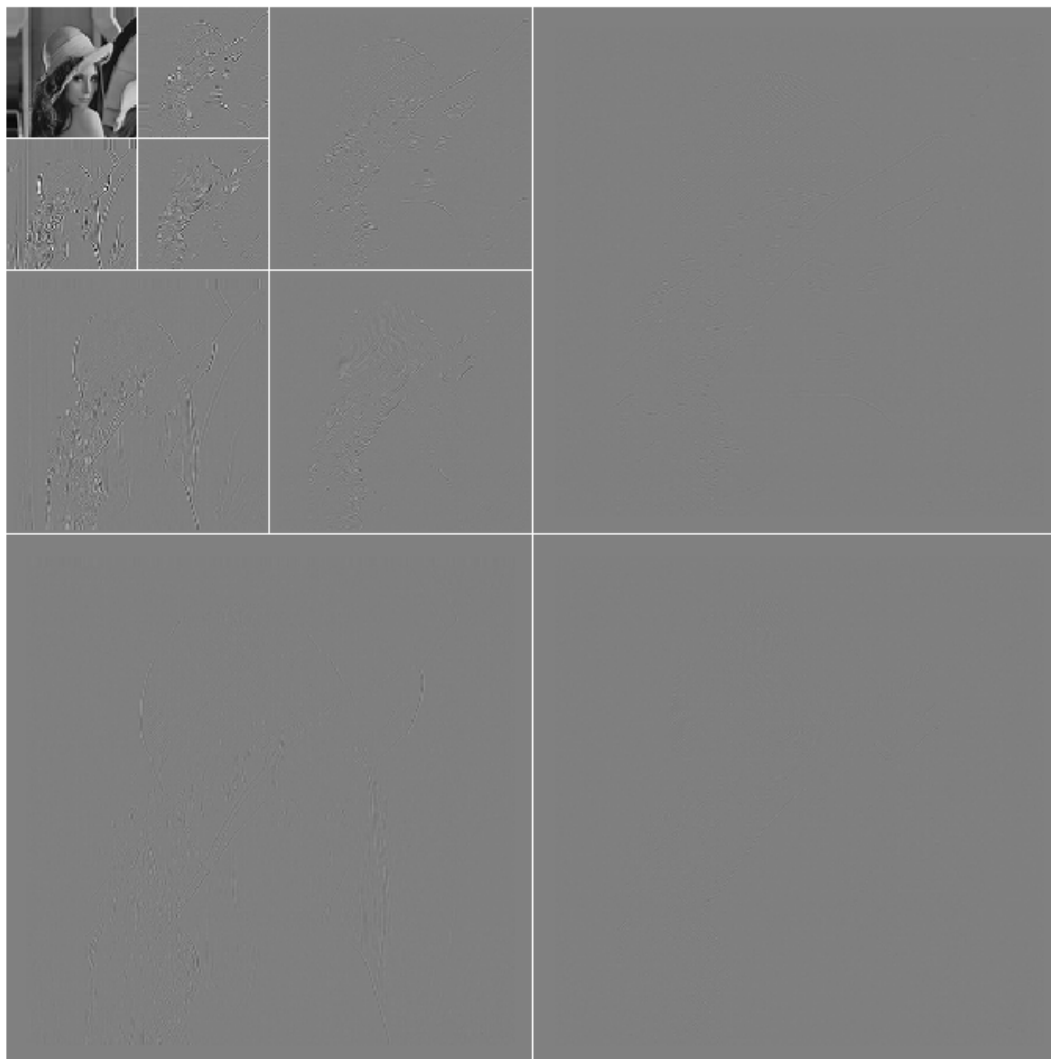


FIGURE 5 – Transformée en Ondelettes Rapide à 3 niveaux d'échelle de l'image *Lenna.tif* avec l'ondelette **db4**



FIGURE 6 – Comparaison de l'image originale (à gauche) et de l'image reconstruite avec l'ondelette de **db4** (à droite)

2.2 Utilisation pour la compression

On effectue la compression de l'image de la façon décrite dans l'énoncé et on observe alors qu'on peut obtenir un taux de compression très important en ne dégradant que peu l'image en choisissant un bon seuil mais également que si on choisit un seuil trop grand on obtient alors un taux de compression sensiblement meilleur mais la qualité de l'image s'en retrouve dégradée de façon plus importante également. Les figures 7 et 8 illustrent ce propos puisqu'on a sur la première une image bien compressée (taux de compression de 85.45%) mais avec une erreur introduite par la compression faible (erreur quadratique moyenne de 2.69%) alors que sur la seconde on a une image davantage compressée (taux de compression de 98.36%) mais l'erreur introduite est elle aussi bien plus importante (erreur quadratique moyenne de 11.91%) et les artefacts liés à la compression sont alors visibles à l'oeil nu. Les artefacts visibles sont liés au fait que l'on a perdu des informations liées aux hautes fréquences dans l'image.



FIGURE 7 – Image compressée avec une valeur de seuil de 10 (compression de 85.45% et erreur de 2.69%)



FIGURE 8 – Image compressée avec une valeur de seuil de 200 (compression de 98.36% et erreur de 11.91%)

3 Correction de défauts localisés

3.1 Traitement des coefficients de la décomposition

En calculant la TOR de l'image `Lena_r.tif`, on remarque que la ligne blanche, étant verticale, apparaît uniquement dans les coefficients verticaux (Figure 9).

Pour enlever cette ligne blanche, on met à zéro les coefficients verticaux correspondant à la position de la ligne. La Figure 10 représente le meilleur résultat, obtenu avec les paramètres $n = 4$ et $\delta = 1$, où n est le nombre d'échelles, et δ le nombre de pixels autour de la ligne.



FIGURE 9 – Transformée en ondelettes rapide

FIGURE 10 – Ondelette **db4**, $n = 4$ et $\delta = 1$

3.2 Effet du type d'ondelette

Intéressons-nous maintenant à l'influence du type d'ondelette utilisé. La figure 11(a) montre que le résultat obtenu avec l'ondelette de Haar est moins bon qu'avec l'ondelette **db4**. L'ondelette **sym6** (figure 11(b)) fait mieux que Haar, et donne un résultat un peu similaire à **db4**.

(a) Ondelette de Haar, $n = 3$ et $\delta = 1$ (b) Ondelette `sym6`, $n = 4$ et $\delta = 1$

FIGURE 11 – Influence du type d'ondelette utilisé

3.3 Effet de l'orientation du défaut

Etudions à présent l'effet de l'orientation du défaut. L'image considérée comporte une ligne non plus verticale, mais oblique. Dès lors, on ne peut plus se limiter à éliminer les coefficients verticaux uniquement, il faut considérer tous les types de coefficients. Cependant, après quelques expérimentations, il est apparu qu'en ne considérant pas les coefficients horizontaux, le résultat obtenu était meilleur. Il est plus compliqué d'obtenir un résultat aussi satisfaisant que lorsque le trait était vertical, comme le montre la Figure 12.

3.4 Effet du type d'image

Changeons d'image, et appliquons la même transformation que précédemment. On constate à l'aide de la TOR que les coefficients horizontaux ne sont pas très importants au niveau du trait blanc, alors que pour l'herbe et le mur blanc derrière ils le sont. Il est donc préférable de ne jouer que sur les coefficients verticaux et diagonaux, et ainsi préserver les textures de natures horizontales. La Figure 13 montre le résultat obtenu de cette manière.

FIGURE 12 – Ondelette `sym6`, $n = 4$ et $\delta = 2$ FIGURE 13 – Ondelette `sym6`, $n = 5$ et $\delta = 2$

4 Conclusion

Dans ce laboratoire, nous avons donc pu mettre en oeuvre diverses applications de la transformée en ondelettes et ainsi nous familiariser avec son intérêt pour des applications en compression d'images et en correction de défauts localisés. Nous avons pu observer son efficacité dans ces différents cas et ainsi comprendre l'important recours à cette méthode pour l'analyse multirésolution d'images numériques.

Annexe A - Code partie 3 : ex3.m

```

1  % _____ %
2  % _____ ELE8812 – TP4: Analyse multir solution _____ %
3  % Julien ANTOINE – 1813026 Maxime SCHMITT –
   1805815 %
4  % _____ %
5  % _____ Exercice 3: Transforme en ondelettes rapide _____ %
6  % _____ %
7
8
9  clear all;
10 % close all;
11 clc;
12
13 im = imread('Lenna.tif');
14
15 %%
16 [c1,s1] = wavedec2(im,3,'db1');
17 [c4,s4] = wavedec2(im,3,'db4');
18 imr1 = waverec2(c1,s1,'db1');
19 imr4 = waverec2(c4,s4,'db4');
20
21
22
23 %%
24 % subplot(1,2,1)
25 % wave2gray(c1,s1);
26 % subplot(1,2,2)
27 % wave2gray(c4,s4);
28
29 %%
30 subplot(1,2,1)
31 imshow(im)
32 subplot(1,2,2)
33 imshow(uint8(imr4))
34
35 diff = abs(im-uint8(imr4));
36 maxDiff = max(diff(:));
37 imshow(diff)
38
39
40 %%
41
42 % [Lo_D,Hi_D,Lo_R,Hi_R] = wfilters('db4');

```

```
43 % subplot(2,2,1)
44 % stem(Lo_D)
45 % title('Lo\_D')
46 % subplot(2,2,2)
47 % stem(Hi_D)
48 % title('Hi\_D')
49 % subplot(2,2,3)
50 % stem(Lo_R)
51 % title('Lo\_R')
52 % subplot(2,2,4)
53 % stem(Hi_R)
54 % title('Hi\_R')
55
56
57
58 %%
59 seuil = 10;
60 c = c1;
61 % Indices des coefficients inférieurs au seuil
62 ind = abs(c1) < seuil;
63 % Suppression de ces coefficients
64 c(ind) = 0;
65 % Calcul du taux de compression
66 compression = sum(ind(:))/numel(c)
67 % Reconstitution de l'image compressée
68 imc = waverec2(c,s1,'db1');
69 erreur = sqrt(sum((double(im(:)) - double(imc(:))).^2) / sum(
    double(im(:)).^2))
70 imshow(uint8(imc))
```

Annexe B - Code partie 4 : suppr.m

```
1 function g = suppr(image,a,b,delta)
2 %SUPPR Summary of this function goes here
3 %   Detailed explanation goes here
4
5 [M,N] = size(image);
6 a = round(M*a);
7 b = round(N*b);
8 x1 = a(1);
9 y1 = a(2);
10 x2 = b(1);
11 y2 = b(2);
12 g = image;
13
14 for x = 1:M
15     for y = 1:N
16         d = abs(x*(y2-y1) - y*(x2-x1) - (x1*y2 - x2*y1)) /
            sqrt((y2-y1)^2 + (x2-x1)^2);
17         if d <= delta
18             g(x,y) = 0;
19         end
20     end
21 end
22
23
24 end
```


Annexe C - Code partie 4 : ex4.m

```

1  % _____ %
2  % _____ ELE8812 – TP4: Analyse multir solution _____ %
3  % Julien ANTOINE – 1813026 Maxime SCHMITT –
   1805815 %
4  % _____ %
5  % _____ Exercice 4: Correction de défauts localiss _____ %
6  % _____ %
7  clear all;
8
9  % Paramtres
10 delta = 1;
11 a = [0.1 0.5];
12 b = [0.9 0.5];
13 im = imread('Lenna_r.tif');
14
15 %% Calcul de la transforme en ondelettes db4
16 lvl = 4;
17 [c,s] = wavedec2(im, lvl, 'db4');
18 % wave2gray(c,s);
19
20 for n = 1:lvl
21     imt = wavecopy('v', c, s, n);
22     coef = suppr(imt,a,b,delta);
23     c = wavepaste('v', c, s, n, coef);
24 end
25
26 ims = waverec2(c,s, 'db4');
27 subplot(1,2,1)
28 imshow(im)
29 subplot(1,2,2)
30 imshow(uint8(ims))
31
32 %% Calcul de la transforme en ondelettes db1
33 lvl = 3;
34 [c2,s2] = wavedec2(im, lvl, 'db1');
35 for n = 1:lvl
36     imt = wavecopy('v', c2, s2, n);
37     coef = suppr(imt,a,b,delta);
38     c2 = wavepaste('v', c2, s2, n, coef);
39 end
40
41 ims2 = waverec2(c2,s2, 'db1');
42 subplot(1,2,1)

```

```
43 imshow(im)
44 subplot(1,2,2)
45 imshow(uint8(ims2))
46
47 %% Calcul de la transforme en ondelettes sym6
48 lvl = 4;
49 delta = 1;
50 [c3,s3] = wavedec2(im, lvl, 'sym6');
51 for n = 1:lvl
52     imt = wavecopy('v', c3, s3, n);
53     coef = suppr(imt,a,b,delta);
54     c3 = wavepaste('v', c3, s3, n, coef);
55 end
56
57 ims3 = waverec2(c3,s3, 'sym6');
58 subplot(1,2,1)
59 imshow(im)
60 subplot(1,2,2)
61 imshow(uint8(ims3))
```

Annexe C - Code partie 4 : ex43.m

```

1  % ----- %
2  % ----- ELE8812 – TP4: Analyse multir solution ----- %
3  % Julien ANTOINE – 1813026 Maxime SCHMITT –
   % 1805815 %
4  % ----- %
5  % ----- Exercice 4: Correction de défauts localiss ----- %
6  % ----- %
7  clear all;
8
9  % Paramtres
10 delta = 2;
11 lvl = 4;
12 a = [0.1 0.4];
13 b = [0.9 0.6];
14 im = imread('Lenna_s.tif');
15
16
17 % Calcul de la transforme en ondelettes
18 [c3,s3] = wavedec2(im, lvl, 'sym6');
19 for n = 1:lvl
20     for t = ['v', 'd']
21         imt = wavecopy(t, c3, s3, n);
22         coef = suppr(imt,a,b,delta);
23         c3 = wavepaste(t, c3, s3, n, coef);
24     end
25 end
26
27 ims3 = waverec2(c3,s3, 'sym6');
28 subplot(1,2,1)
29 imshow(im)
30 subplot(1,2,2)
31 imshow(uint8(ims3))

```

Annexe C - Code partie 4 : ex44.m

```

1  % ----- %
2  % ----- ELE8812 – TP4: Analyse multir solution ----- %
3  % Julien ANTOINE – 1813026 Maxime SCHMITT –
   % 1805815 %
4  % ----- %
5  % ----- Exercice 4: Correction de défauts localiss ----- %
6  % ----- %
7  clear all;
8
9  % Paramtres
10 delta = 2;
11 lvl = 5;
12 a = [0.1 0.4];
13 b = [0.9 0.6];
14 im = imread('phare_s.tif');
15
16
17 % Calcul de la transforme en ondelettes
18 [c3,s3] = wavedec2(im,lvl,'sym6');
19 % wave2gray(c3,s3);
20
21 for n = 1:lvl
22     for t = ['v','d']
23         imt = wavecopy(t,c3,s3,n);
24         coef = suppr(imt,a,b,delta);
25         c3 = wavepaste(t,c3,s3,n,coef);
26     end
27 end
28
29 ims3 = waverec2(c3,s3,'sym6');
30 subplot(1,2,1)
31 imshow(im)
32 subplot(1,2,2)
33 imshow(uint8(ims3))

```