COMPTE RENDU - TRAVAUX PRATIQUES

Numerical Tour of Image Processing for denoising

Groupe

URQUIZA Andrea

BECKER Gonzalo

Date d'édition : 25 juin 2025
Version : 1.0

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Sommaire

1. Etablissement d'une base d'ondelettes

1.1. Étape de décomposition

Pour construire une base orthogonale de chaque V_j, une fonction unique appelée fonction d'échelle ϕ peut être dilatée et translatée.

1.1.1. Filtre associé à la fonction d'échelle

L'approximation d'une fonction est réalisée par une projection orthogonale sur V_j. Toute fonction d'échelle est spécifiée par un filtre discret appelé filtre miroir conjugué. De cette façon, on peut arriver à la relation suivante, qui résulte du développement pour trouver la base orthonormale.

$$\widehat{\phi}(2\omega) = \frac{1}{\sqrt{2}} \, \widehat{h}(\omega)\widehat{\phi}(\omega)$$

Sachant $\widehat{\phi}(\omega)$, on procède à obtenir $\widehat{h}(\omega)$:

$$\left(\frac{sin(\omega)}{\omega}\right)^3 e^{-i\omega} = \frac{1}{\sqrt{2}} \, \widehat{h}(\omega) \left(\frac{sin(\omega/2)}{\omega/2}\right)^3 e^{-i\omega/2}$$

Étant donné que

$$sin(\omega) = 2 \; sin\left(\frac{\omega}{2}\right) cos\left(\frac{\omega}{2}\right)$$

il est possible de résoudre pour obtenir :

$$\widehat{h}(\omega) = \sqrt{2} \; cos^3\left(\frac{\omega}{2}\right) e^{-i\omega/2}$$

1.1.2. Choix d'un filtre g associé à l'ondelette ψ

Selon la définition de $\widehat{g}$, on sait que :

$$\widehat{g_1}(\omega) = e^{-i\omega} \; h^*(\omega + \pi)$$

En sachant $\widehat{h}(\omega)$ et que $cos(\omega/2 + \pi/2) = -sin(\omega/2)$, le résultat suivant est obtenu :

$$\widehat{g_1}(\omega) = -i \; \sqrt{2} \; sin^3\left(\frac{\omega}{2}\right) \; e^{-i\omega/2}$$

Dans le cadre de la théorie des ondelettes, il est possible de montrer que la base construite est orthogonale si et seulement si les deux conditions suivantes sont satisfaites :

$$|\widehat{g}(\omega)|^2 + |\widehat{g}(\omega + \pi)|^2 = 2$$

$$\widehat{g}(\omega) \; \widehat{h}^*(\omega) \;\; + \;\; \widehat{g}(\omega + \pi) \; \widehat{h}^*(\omega + \pi) \;\; = \;\; 0$$

La deuxième relation implique la quadrature entre $\widehat{g}$ et $\widehat{h}$. Avec la nouvelle fonction $\widehat{g}$, il est facile de démontrer que la quadrature ne se respecte plus, ou, autrement dit, que la base construite n'est plus orthogonale.

1.1.3. Calcul de l'ondelette ψ a partir du filtre g

Les ondelettes orthogonales portent les détails nécessaires pour augmenter la résolution d'une approximation du signal. Il est possible d'obtenir la relation suivante pour ψ :

$$\widehat{\psi}(\omega) = \frac{1}{\sqrt{2}} \, \widehat{g}(\omega/2) \; \widehat{\phi}(\omega/2)$$

Sachant $\widehat{\phi}$ et $\widehat{g}$, on obtient $\widehat{\psi}$:

$$\widehat{\psi}(\omega) = -\frac{i\omega}{4} \left(\frac{sin(\omega/4)}{\omega/4}\right)^4 e^{-i\omega/2}$$

1.2. Étape de Reconstruction

1.2.1. Choix de $\widetilde{h}$

Pour que la reconstruction soit parfaite, il est nécessaire que

$$\widehat{h}^*(\omega)\ \widehat{\widetilde{h}}(\omega)\ +\ \widehat{h}^*(\omega+\pi)\ \widehat{\widetilde{h}}(\omega+\pi) = 2$$

Si on prend $\widetilde{h} = h$, elle devient

$$\left|\widehat{h}(\omega)\right|^2 + \left|\widehat{h}(\omega+\pi)\right|^2 = 2$$

sce qui, par définition de h, est toujours satisfait.

1.2.2. Choix de $\widetilde{g}$

Les relations entre $\widetilde{g}$, $\widehat{h}$, $\widehat{\widetilde{h}}$ et $\widehat{g}$ pour assurer une reconstruction parfaite sont les suivantes :

$$\widehat{h}^*(\omega+\pi)\ \widehat{\widetilde{h}}(\omega)\ +\ \widehat{g}^*(\omega+\pi)\ \widehat{\widetilde{g}}(\omega) = 0$$

$$\widehat{h}^*(\omega)\ \widehat{\widetilde{h}}(\omega)\ +\ \widehat{g}^*(\omega)\ \widehat{\widetilde{g}}(\omega) = 2$$

Si on prend la deuxième, où $\widetilde{h} = h$, on arrive au résultat recherché :

$$\widehat{\widetilde{g}}(\omega) = \frac{2 - \left|\widehat{h}(\omega)\right|^2}{\widehat{g}^*(\omega)}$$

2. Décomposition/reconstruction multi-échelle par ondelettes

2.1. Préliminaires

2.1.1. Sous-échantillonnage d'un signal par mise à zéro d'un point sur deux

$$\sum_{n=-\infty}^{\infty} x_{2e}[n]\, e^{-in\omega} = \sum_{n=-\infty}^{\infty} (x[2n]\, e^{-i2n\omega} + x[2n+1]\, e^{-i(2n+1)\omega}) =$$

$$\frac{1}{2}\left(\sum_{n=-\infty}^{\infty}\left(x[2n]\, e^{-i2n\omega} + x[2n+1]\, e^{-i(2n+1)\omega}\right) + \sum_{n=-\infty}^{\infty}\left(x[2n]\, e^{-i2n\omega} - x[2n+1]\, e^{-i(2n+1)\omega}\right)\right) =$$

$$\frac{1}{2}\left(\sum_{n=-\infty}^{\infty} x_{2e}[n]\, e^{-in\omega} + \sum_{n=-\infty}^{\infty} x_{2e}[n]\, e^{-in(\omega+\pi)}\right) = \frac{1}{2}\left(\widehat{x}[\omega] + \widehat{x}[\omega + \pi]\right)$$

2.1.2. Insertion de zéros dans un signal

$$\widehat{\check{x}} = \sum_{n=-\infty}^{\infty} \check{x}[n]\, e^{-in\omega} = \sum_{n=-\infty}^{\infty} x[n]\, e^{-i2n\omega} = \widehat{x}[2\omega]$$

2.1.3. Relation entre convolution et corrélation

Le filtrage d'un signal $x(t)$ par un filtre h est donné par une convolution :

$$(x * h)(t) = \int_{-\infty}^{\infty} x(\tau)\, h(t-\tau) dt$$

En temps discret, cela devient

$$(x * h)[n] = \sum_{k=-\infty}^{\infty} x[k]\, h[n-k]$$

Alors que la corrélation entre x et h se calcule comme :

$$R(x, h) = \sum_{k=-\infty}^{\infty} x[k]\, h[k-n]$$

d'où la possibilité de remplacer $\overline{h}[n] = h[-n]$.

2.2. Décomposition multi-échelle d'un signal

A fin d'obtenir les coefficients sachant

$$a_j[p] = \langle x(t) | \phi_{j,p} \rangle$$

on démontre les relations entre les coefficients d'approximation et de détail du niveau

$$j$$

$$\phi_{j+1,p} = \sum_{n=-\infty}^{\infty} \langle \phi_{j+1,p}, \phi_{j,n} \rangle \phi_{j,n}$$

Étant donné $t' = 2^{-j}t - 2p$ on a :

$$\langle \phi_{j+1,p}, \phi_{j,n} \rangle = h[n-2p]$$

Alors :

$$\phi_{j+1,p} = \sum_{n=-\infty}^{\infty} = h[n-2p]\phi_{j,n}$$

Ainsi en prenant le produit interne entre f et le vecteur de chaque coté de l'équation on obtient l'équation qu'on chercher a démontrer.

Pour démontrer la deuxième équation on fait un traitement pareil :

$$\psi_{j+1,p} = \sum_{n=-\infty}^{\infty} \langle \psi_{j+1,p}, \phi_{j,n} \rangle \phi_{j,n}$$

En prenant $t' = 2^{-j}t - 2p$

$$\langle \psi_{j+1,p}, \phi_{j,n} \rangle = g[n-2p]$$

Ainsi on obtient :

$$\psi_{j+1,p} = \sum_{n=-\infty}^{\infty} = g[n-2p]\phi_{j,n}$$

Ainsi en prenant le produit interne entre f et le vecteur de chaque coté de l'équation on obtient l'équation qu'on chercher a démontrer.

3. Établissement d'une base d'ondelettes

3.1. Étape de décomposition

On définit les fonction d'après les résultats obtenus dans la partie théorique :

```
function phi = Phi(w) % fonction d'echelle
    phi = (sin(w./2)./w./2).^3 .* exp(-1i.*w/2);
end


function h = H(w) % filtre echelle
    h = sqrt(2).*cos(w./2).^3.*exp(-1i.*w./2);
end


function g = G(w)
    g = -1i.*sqrt(2).*sin(w./2).*exp(-1i.*w/2);
end

function psi = Psi(w)
    psi = -1i.*(w./4).*exp(-1i.*w./2).*(sin(w./4)/(w./4)).^4;
end
```

Les figures montrées ci-après illustrent les spectres de la fonction d'échelle et l'ondelette et les filtres associées respectivement Fig. 1. Dans la Figure 2 on observe les filtres de reconstruction associés a chaque fonction. Finalement dans la Figure 3 on illustre les coefficients de la réponse impulsionnelle des filtrés calculés entre l'instant -2 a l'instant 3.

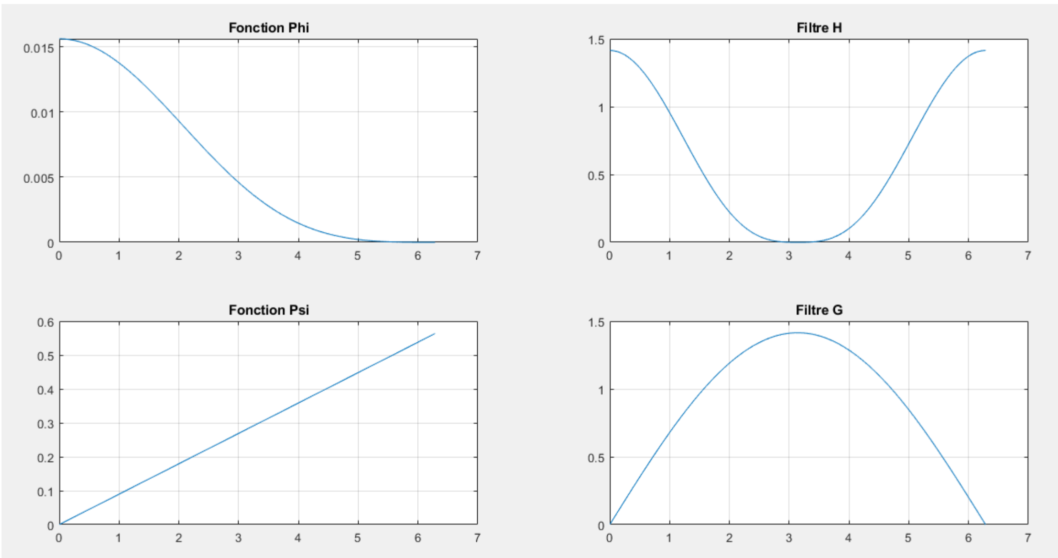

FIGURE 1 – Courbes fonctions et filtres

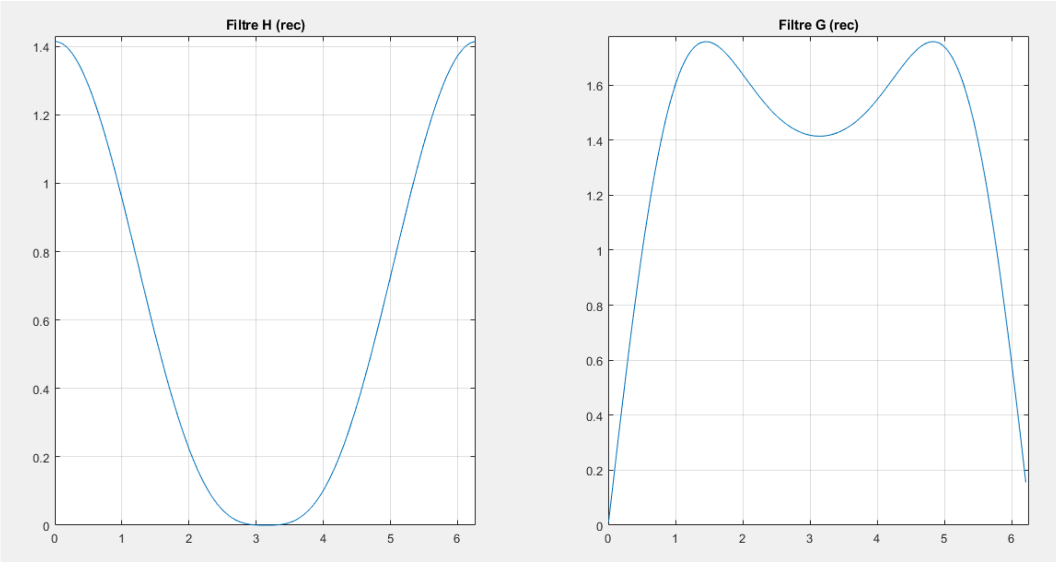

FIGURE 2 – Filtres de reconstruction

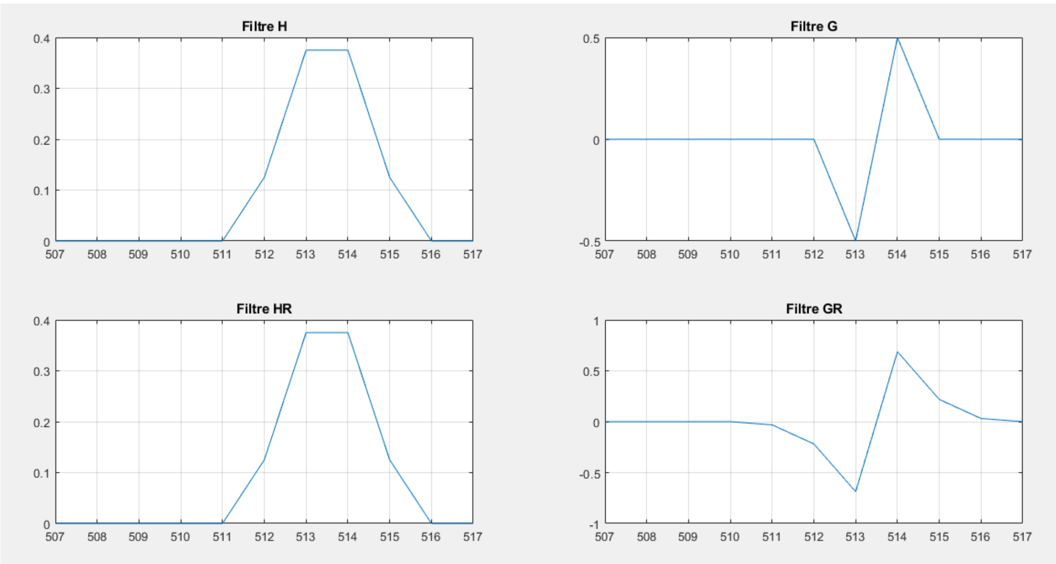

FIGURE 3 – Récupération des coefficients

4. Préliminaire

4.1. Fonction a trous

La fonction FiltreATrous(signal, echelle) qui insère 2^{chelle} zéros entre chaque valeur du signal.

```
function fn = FiltreATrous(signal, echelle) % function a trous
    n = 2^(echelle)-1;
    fn_prov = reshape([signal;zeros(n,numel(signal))],1,[]);  %signal
       1D
    fn = [zeros(1, floor(n/2)) fn_prov(1:(numel(fn_prov)-n)) zeros(1,
       n-floor(n/2))];

end
```

5. Décomposition/reconstruction multi-échelle par ondelettes

5.1. Construction des filtres de décomposition et de reconstruction pour un niveau donné

La fonction FiltreATrousFrequentiel(filtre0, n, nEch) retourne le spectre du filtre a trous de niveau n a partir de niveau 0 sur nEbEch échantillons en applicant le zéro padding et la fonction qui implémente l'algorithme a trous.

```
function fn = FiltreATrousFrequenciel(filtre0, n,nEch) % function a
   trous
    f1 = FiltreATrous(filtre0, n);
    fn = fft([f1 zeros(1,nEch-numel(f1))]);

end
```

Dans la figura 5 on observe la réponse impulsionnelle des filtres de décomposition et de reconstruction. On remarque que a chaque niveau d'échelle on obtient une structure diffèrent du filtre.

Dans la figure 6 on observe la réponse fréquentielle du filtre a trous ou tous les filtres ont les mêmes nombre de coefficients mais différente dilatation.

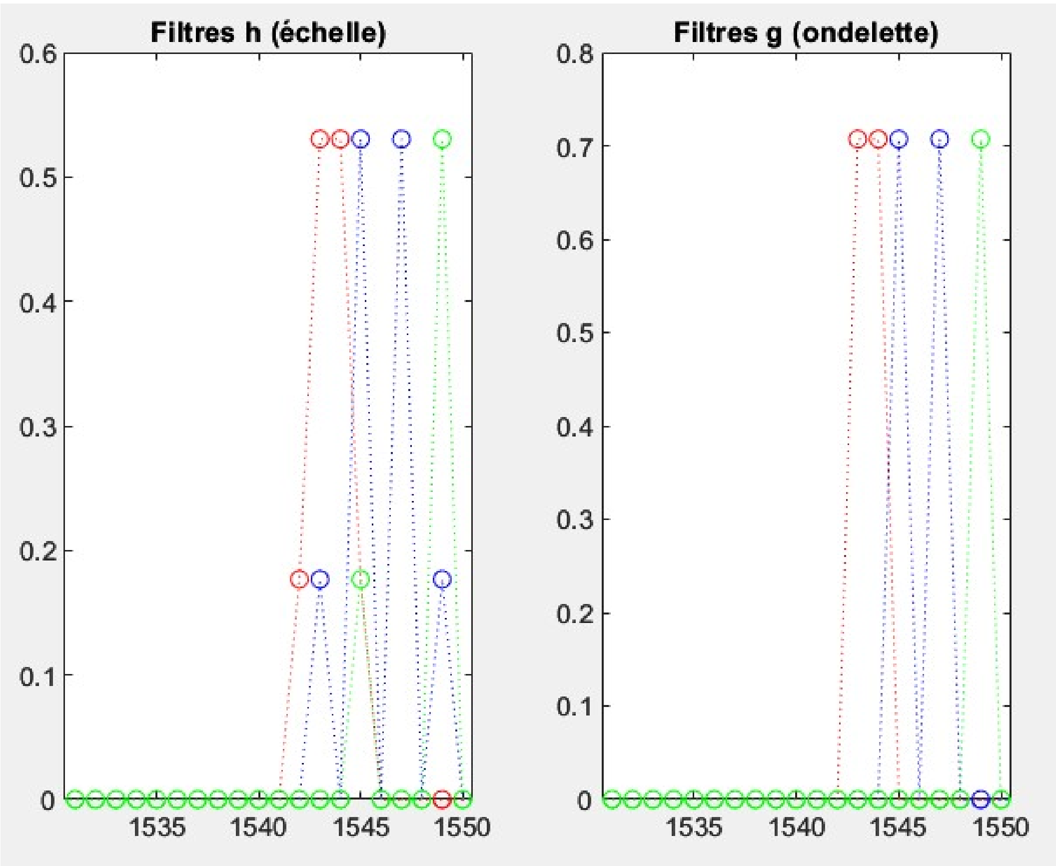

FIGURE 4 – Filtres a trous temporel

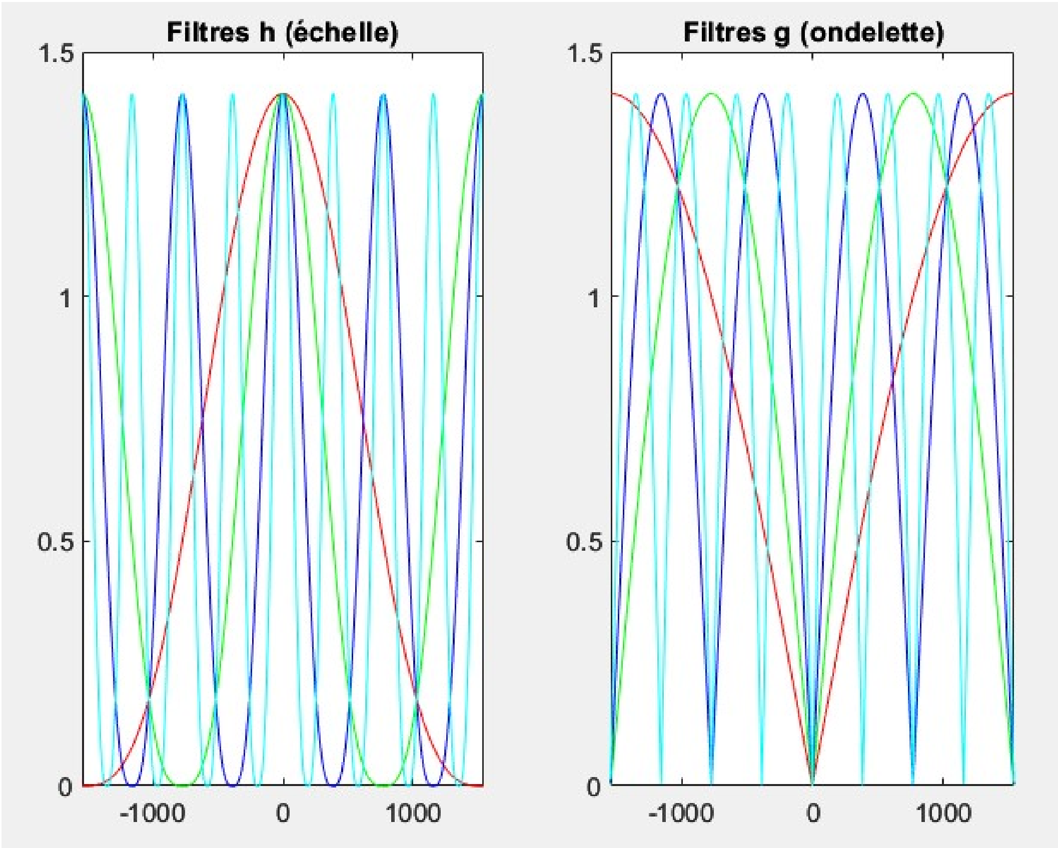

FIGURE 5 – Filtres a trous fréquentiel

5.2. Calcul des coefficients d'approximation et de détail d'un signal

La fonction Decomposition calcule les coefficients d'approximation et de détail du signal en passant par le domaine fréquentiel.

```
function [ax, dx] = Decomposition(signal , hf , gf)

    nbScale = size(hf,1);
    nbEch = size(signal,2);

    ax = zeros(nbScale+1, nbEch);
    dx = zeros(nbScale+1, nbEch);

    ax(1,:) = signal;

    for s = 1:nbScale
        ax(s+1,:) = ax(s,:).*hf(s,:);
        dx(s+1,:) = ax(s,:).*gf(s,:);
    end
end
```

On obtient les coefficients d'approximation et de détail pour différentes échelles. Dans les figures montrées ci-après on peut observer les spectres des coefficients pour un niveau donné. Ainsi on observe la nature du filtre à chaque niveau ou chacune à la même quantité de coefficients mais la différence d'échelle donne largeurs de bandes différentes.

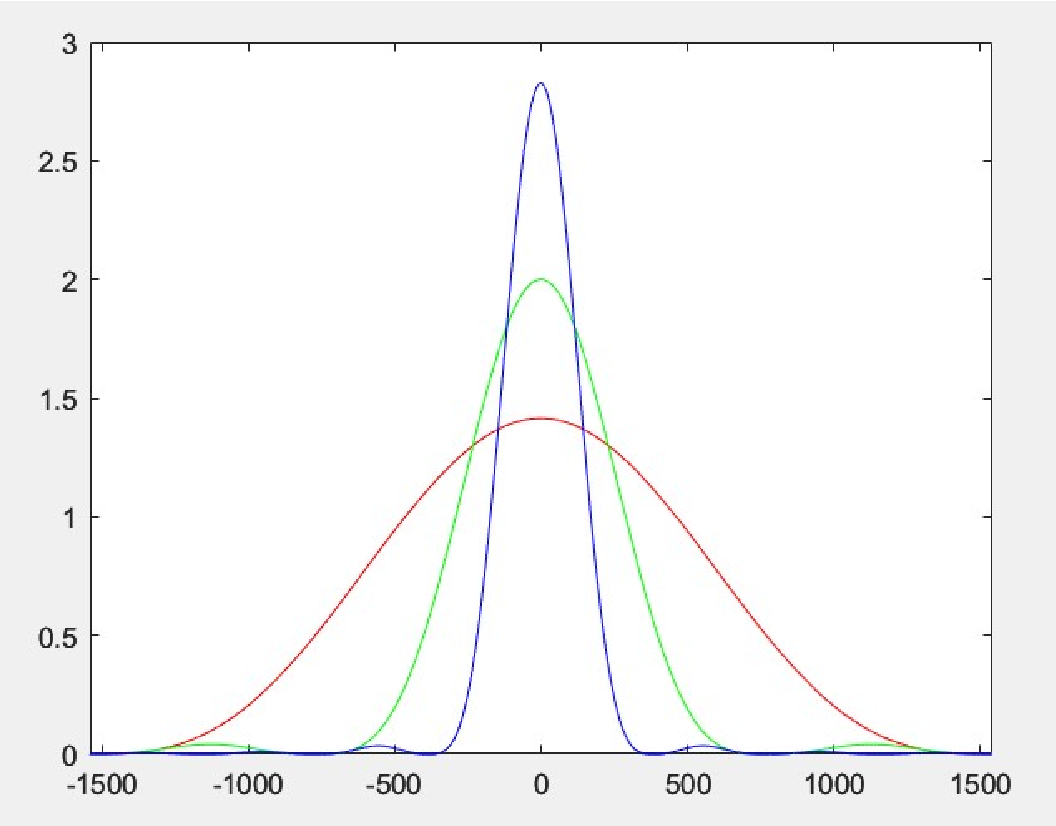

FIGURE 6 – Nature du filtrage de la fonction d'approximation opéré par la fonction d'échelle

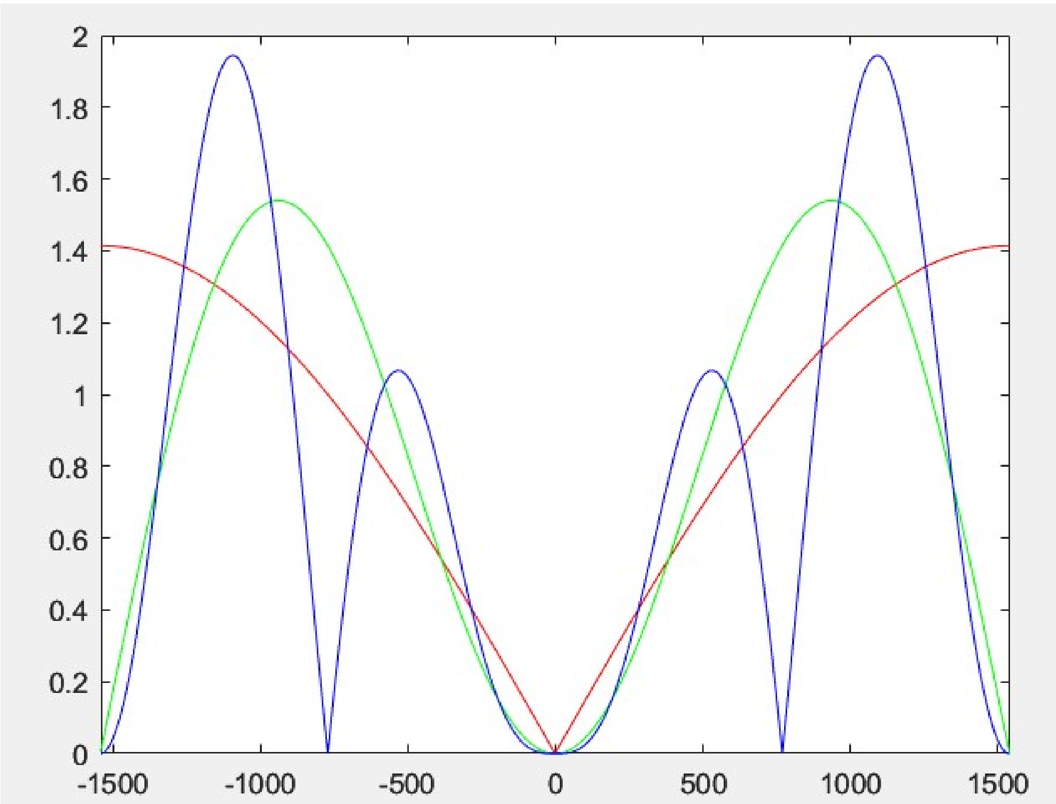

FIGURE 7 – Nature du filtrage de la fonction d'échelle opéré par la fonction d'échelle

5.3. Reconstruction du signal décomposé

Afin de reconstruire les coefficients d'approximation et de détails du niveau le plus profond. La fonction montrée ci-après fait cette opération.

```
function signal = Reconstruction(lastAx,dx,hfD,gfD)

    nbScale = size(dx,1)-1 ;
    nbEch = size(dx,2);

    signal = zeros(nbScale+1,nbEch);
    signal = lastAx(1:nbEch);

    %On reconstruit successivement le signal
    for cpt=nbScale:-1:1
        A = signal(1:nbEch).*hfD(1:nbEch);
        B = dx(cpt, 1:nbEch).*gfD(1:nbEch);
        signal(1:nbEch) = (A+B)/sqrt(2);
    end
end
```

La reconstruction obtenue se montre dans la Fig. 9. Ainsi on ignore les emplacements et les échelles de temps qui correspondent au bruit pour reconstruire l'image.

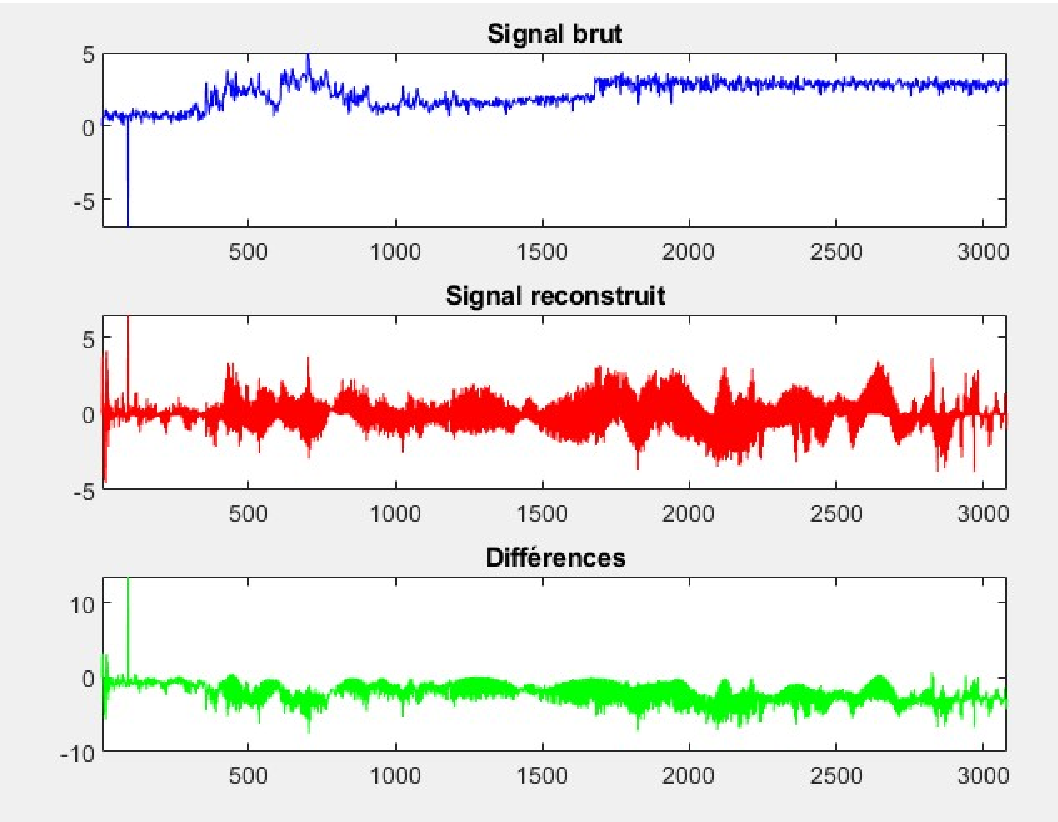

FIGURE 8 – Reconstruction du signal

5.4. Image reconstruite

L'image débruitée se montre dans la fig 10. En dépit du bon fonctionnement des fonctions FiltreATrous et FiltreATrousFrequenciel le débruitage réalisé sur l'image n'est pas effectif.

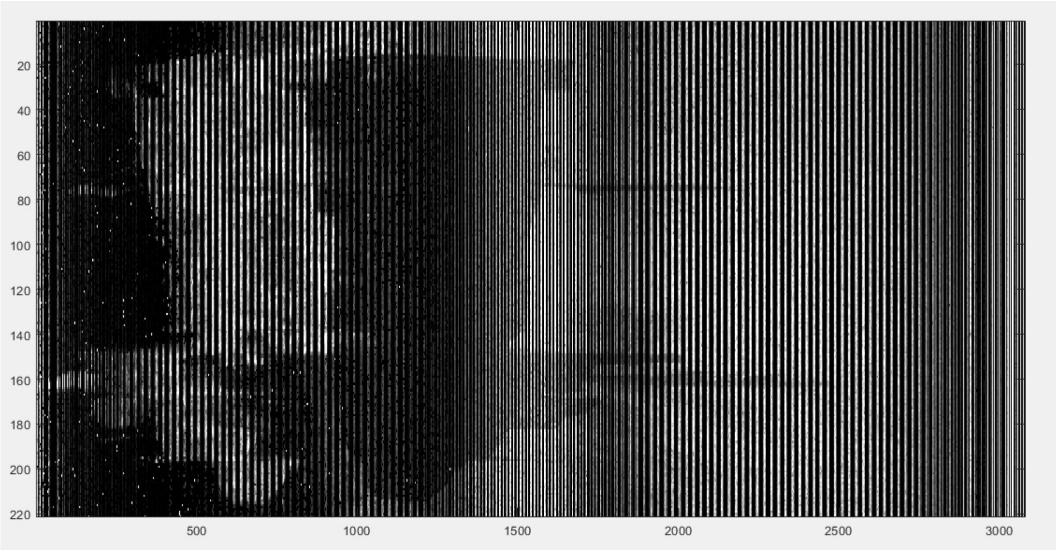

FIGURE 9 – Image reconstruite

5.5. Conclusion

Au travers de cette BE on a réussi à mettre en application les ondelettes et l'algorithme à trous pour implémenter le débruitage d'un image.