Thomas LE MARREC
Corentin MANTION



# Compression d'images par ondelettes

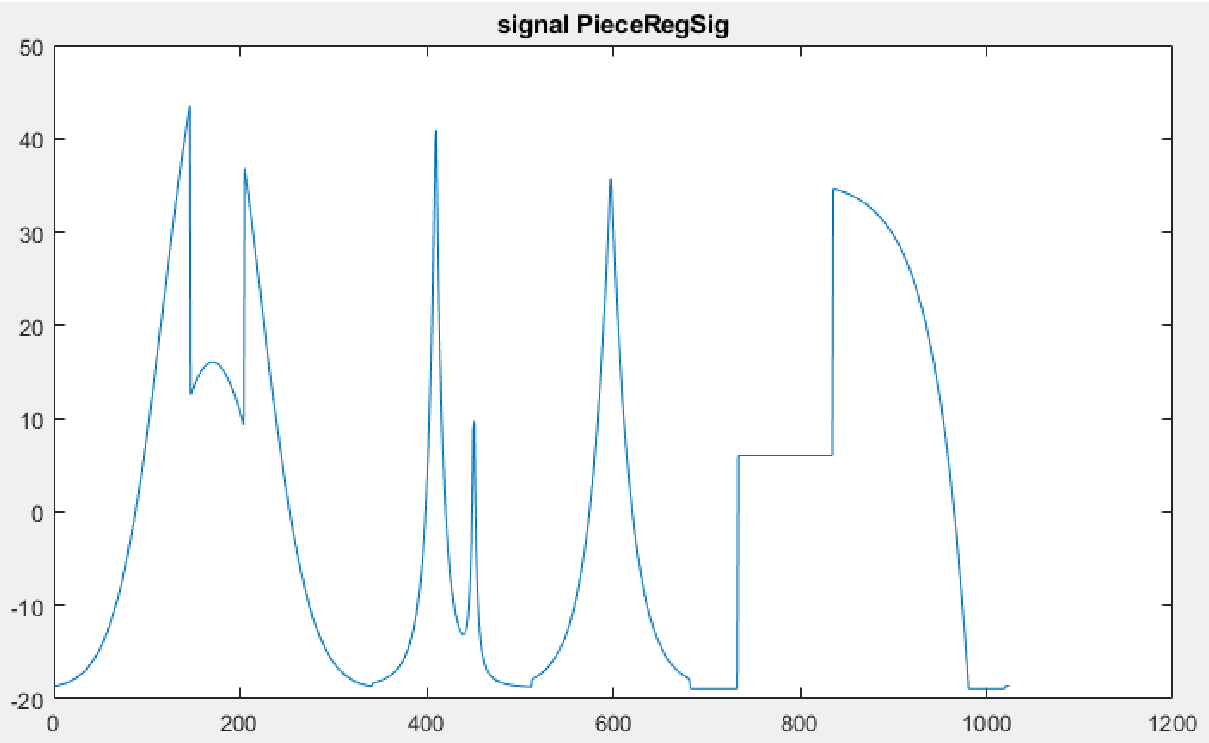
## Introduction

Au cours de cette séance de travaux pratiques, nous appliquerons nos connaissances du cours de théorie de l'information afin d'étudier et de mettre en œuvre une méthode de compression d'image.

La problématique sera ici de comprendre la décomposition en ondelettes, et de développer une méthode de compression à partir de celle-ci.

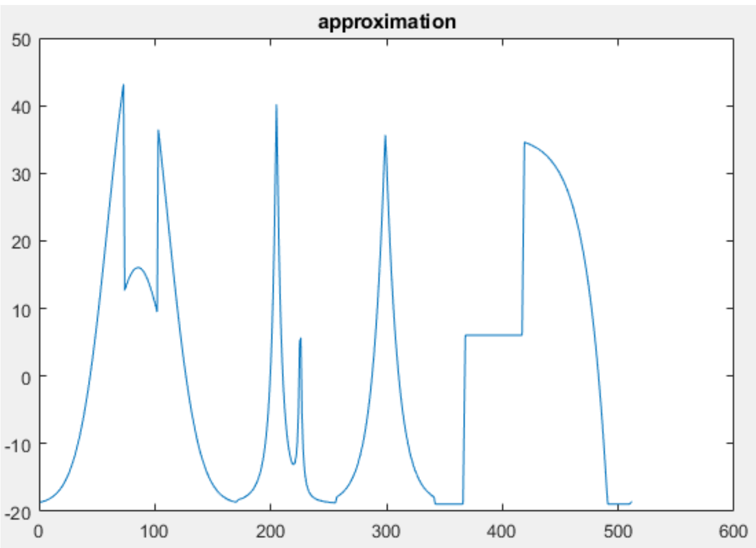
## Décomposition en ondelettes et reconstruction parfaite

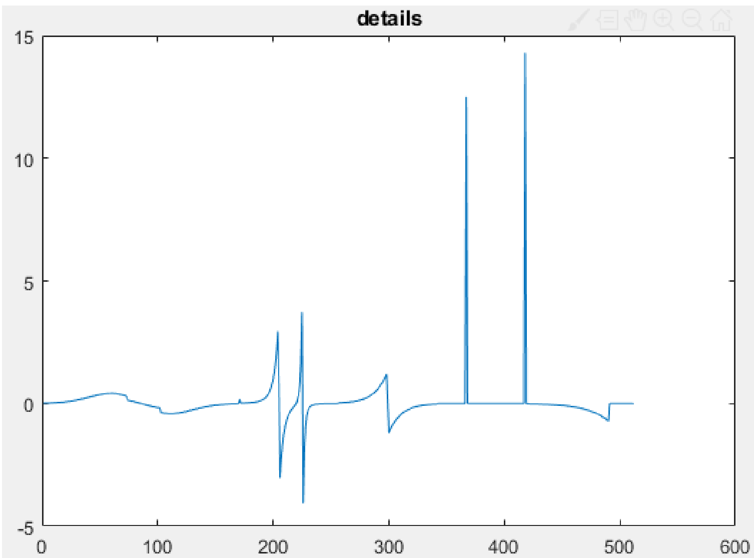
La courbe que l'on obtient en chargeant PieceRegSig




### Première décomposition dans la base de Haar

Voici le résultat de la décomposition des signaux dans la base de Haar :

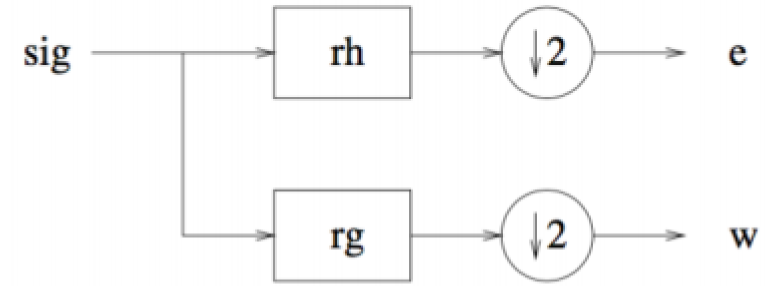


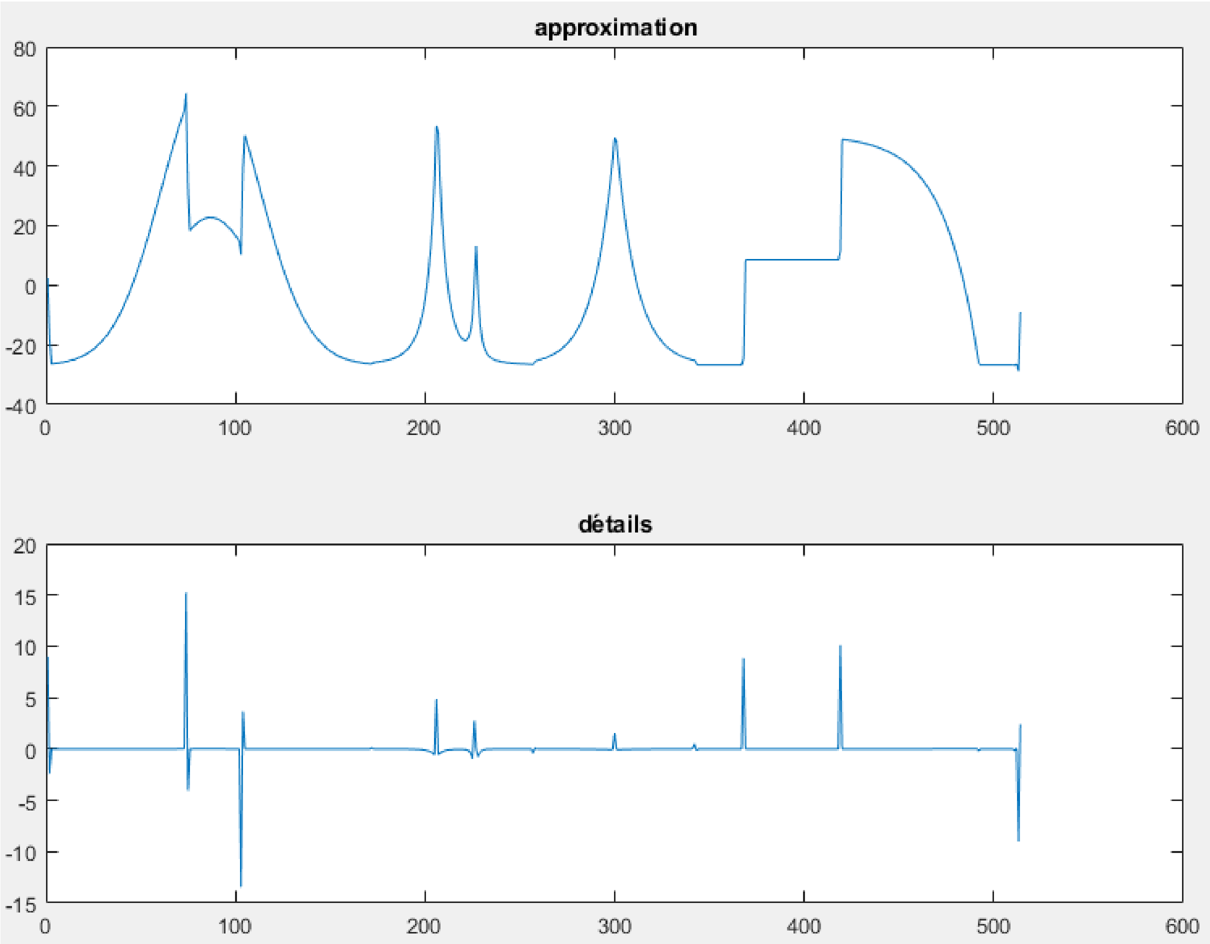
Cela correspond bien au résultat attendu : une approximation proche du signal d'origine, et le détail présente des pics au voisinage des sauts du signal.

## Convolution

Nous utilisons maintenant les filtres de Daubéchie pour obtenir notre décomposition du signal, selon le schéma suivant :



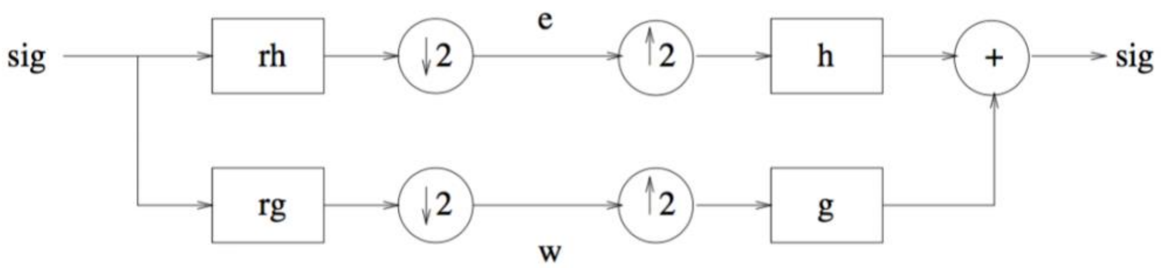

Voici le résultat de la cascade de filtre de Daubéchie :



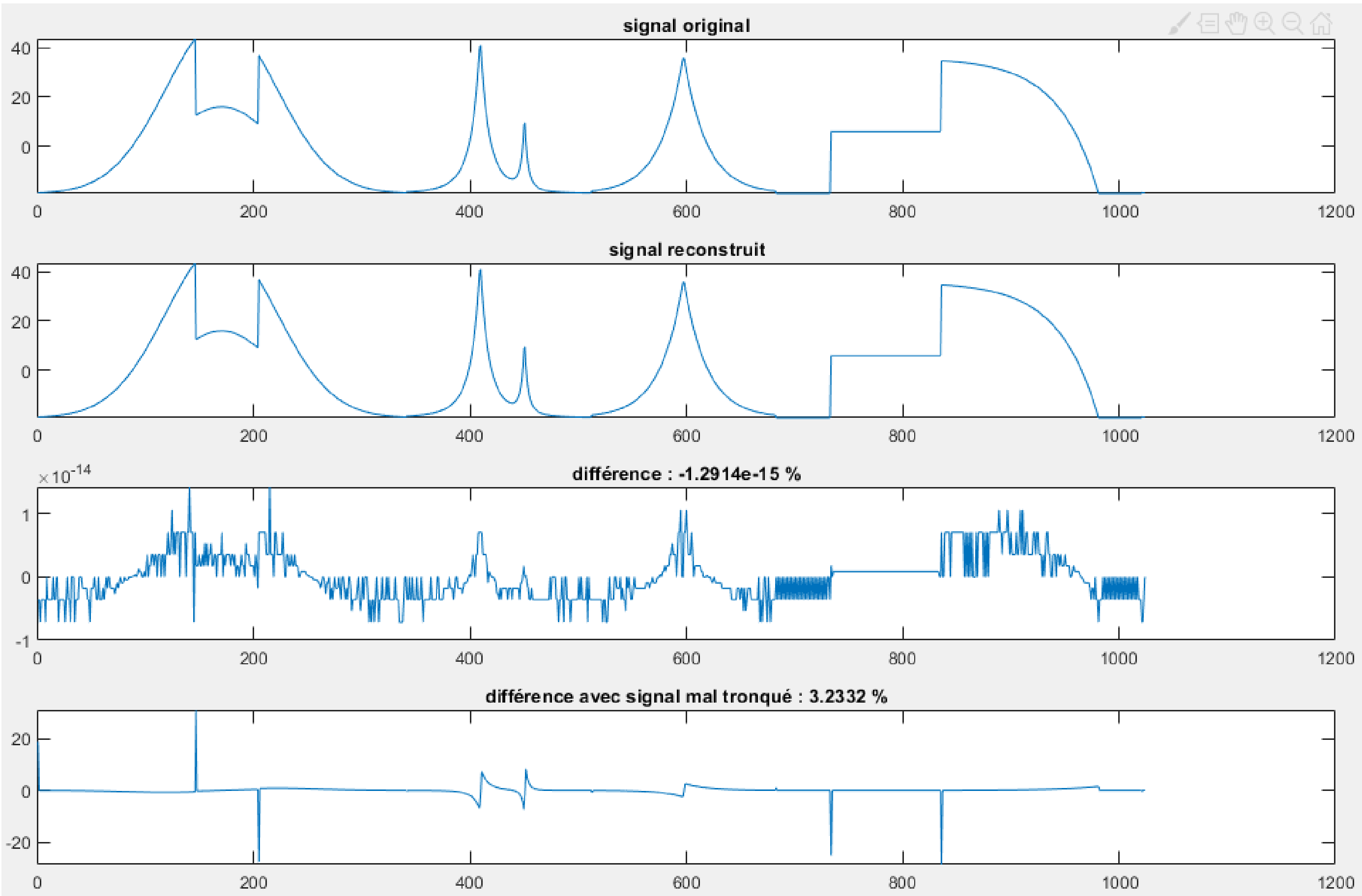

Nous pouvons remarquer la présence de discontinuités au bord des signaux d'approximation et de détail. Cela est du au fait que le signal qu'on convolue est lui-même discontinue à ses extrémités puisqu'il est fini.

## Reconstruction

Nous allons maintenant reconstruire le signal décomposé par la cascade de filtre de Daubéchie, en suivant le schéma suivant :




Nous sur-échantillonnons en intercalant un signal de zéros aux signaux e et w.
L'approximation et le détail restent bien sur les mêmes que dans la partie précédente.
Voici le résultat qu'on obtient en reconstruisant le signal selon la méthode décrite dans le schéma :

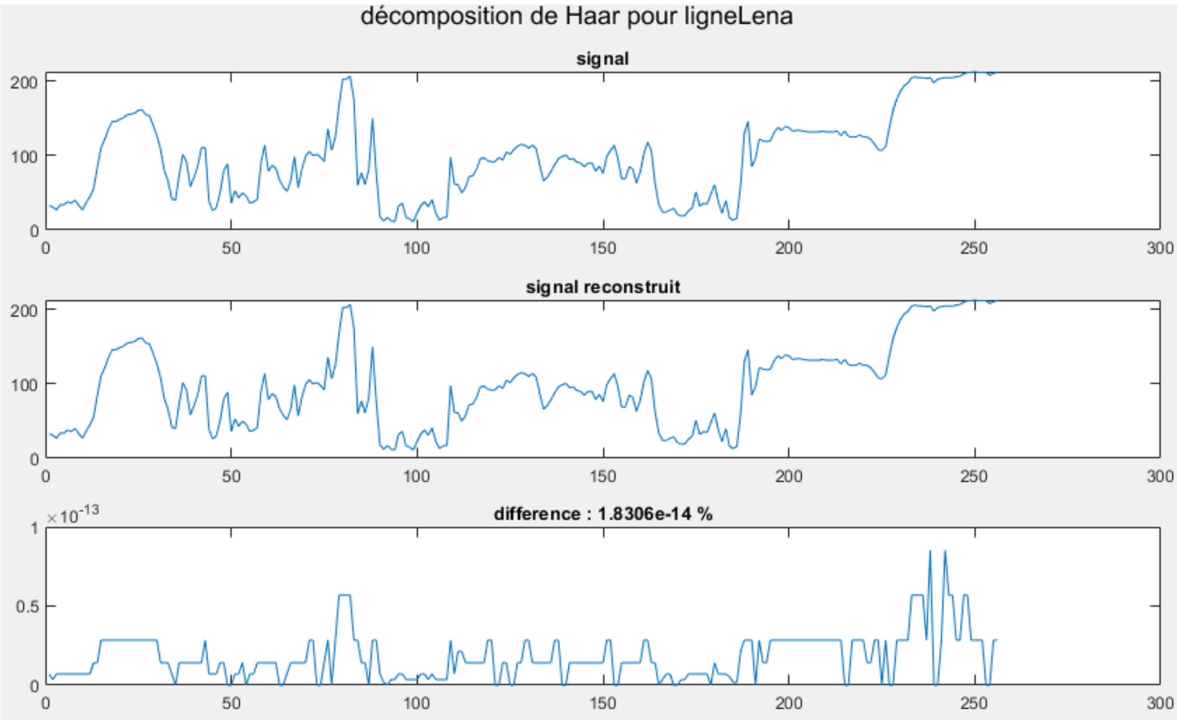



On remarque que la différence lorsque le signal est tronqué comme prévu n'est pas nulle mais l'ordre de grandeur est clairement négligeable par rapport à l'amplitude du signal d'origine ($1{,}2914 * 10^{-15}\%$). Lorsque le signal est mal tronqué, la différence est cette fois remarquable, l'ordre de grandeur de l'erreur est beaucoup plus important : 3,2332%.
Le bon troncage du signal est donc important.

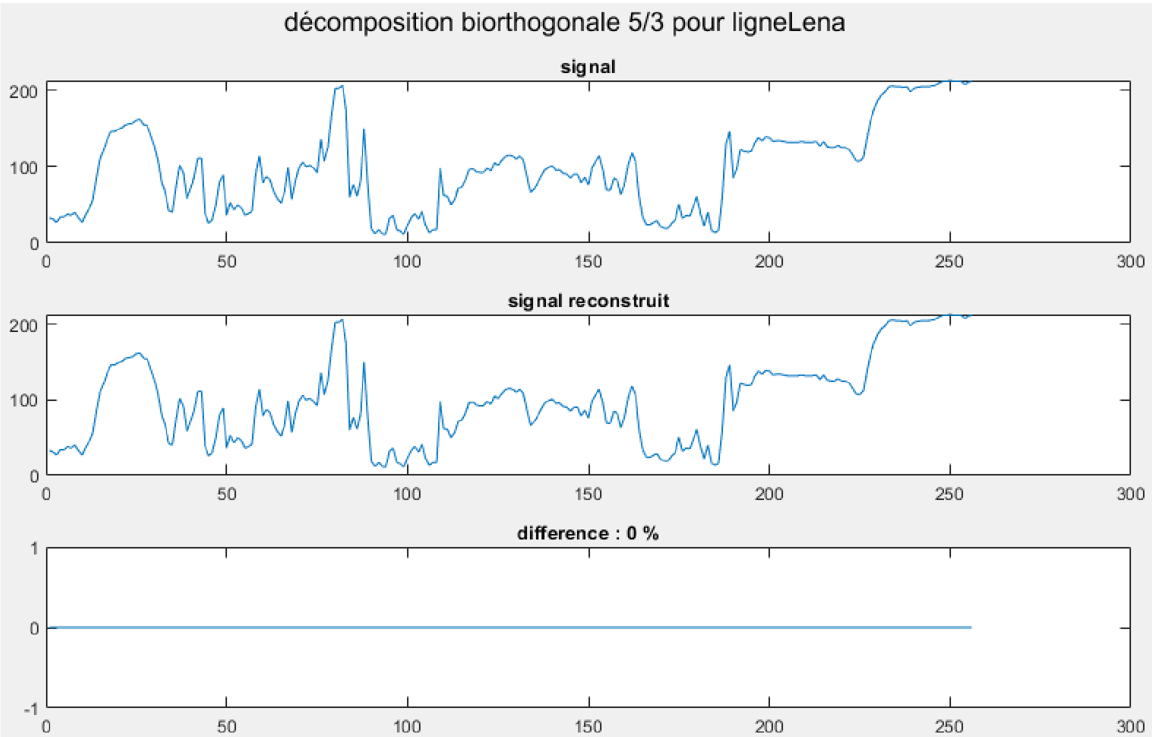
Thomas LE MARREC
Corentin MANTION


# Compression des Signaux 1D

Voici le résultat des décompositions puis des reconstructions du signal ‘ligneLena’ pour différentes méthodes :

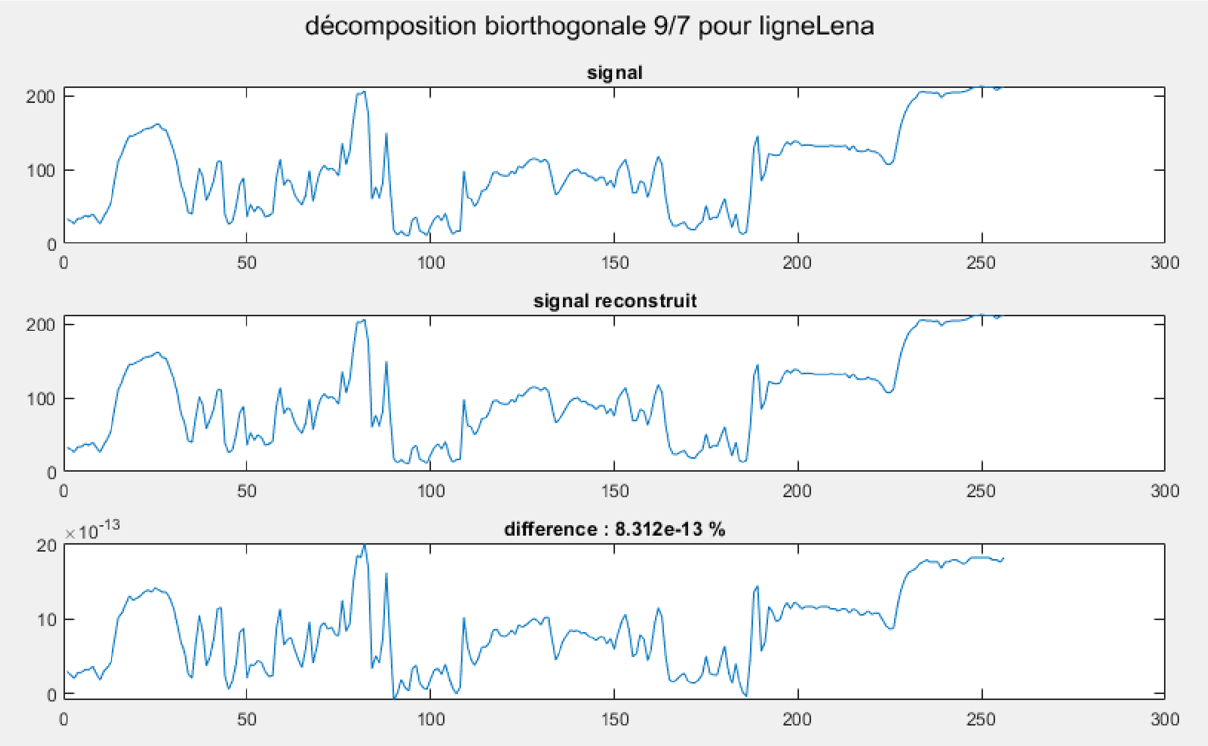



On observe le même ordre de grandeur d’erreur pour la décomposition de Haar que lors des précédents essais sur la base de Haar.

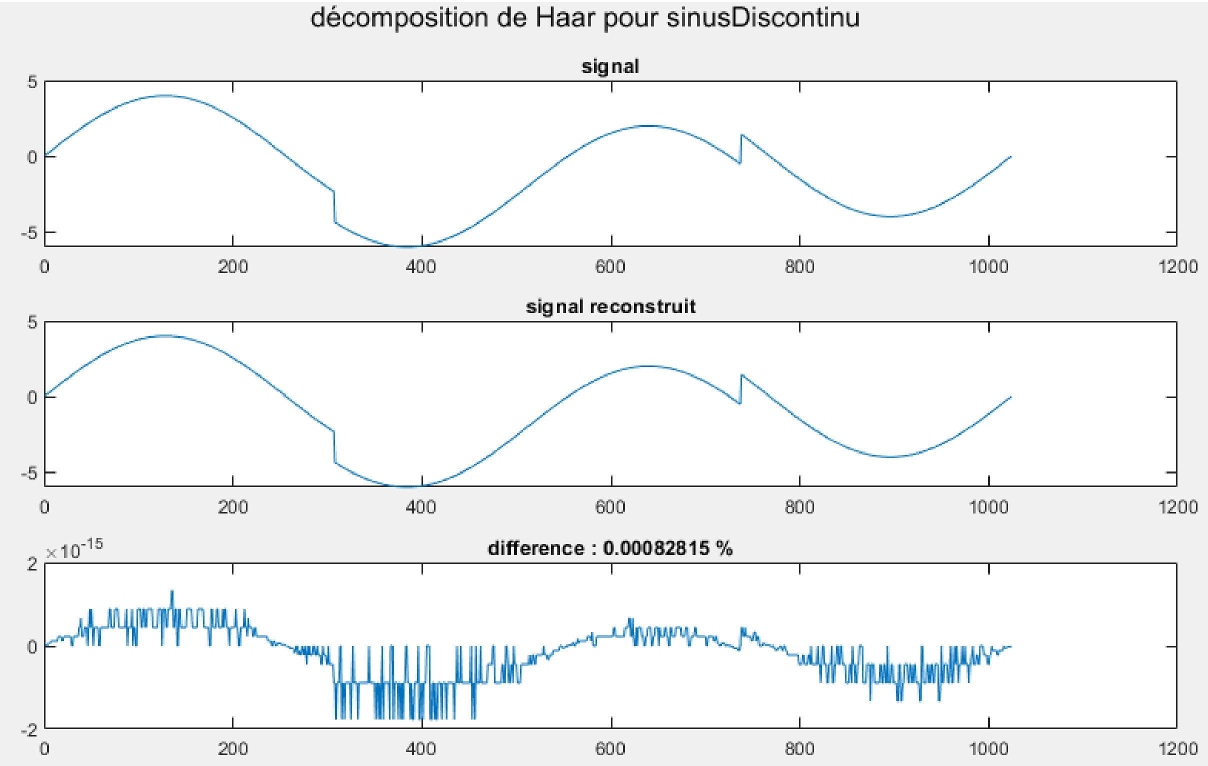



Cette fois on observe une erreur nulle pour la reconstruction du signal ligneLena par la méthode biorthogonale 5/3. Cette méthode permet donc une reconstruction parfaite des signaux.

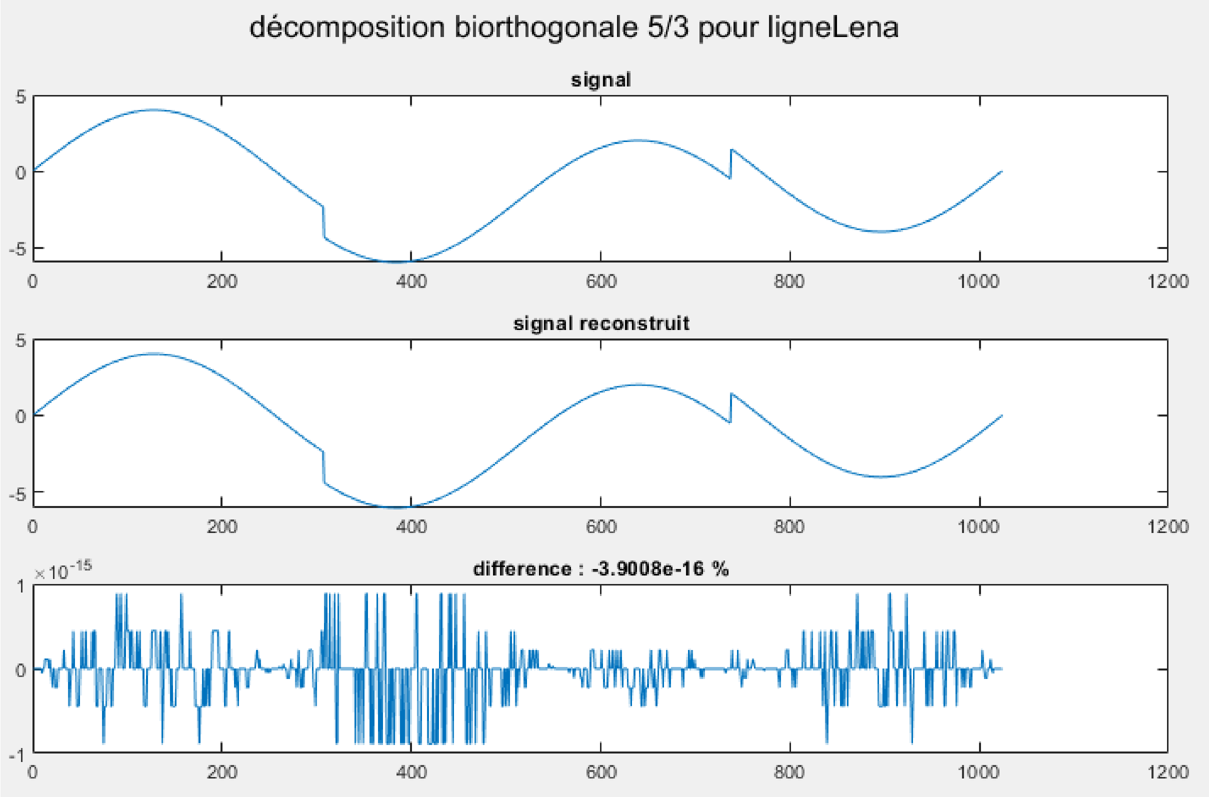
La décomposition biorthogonale 9/7 pour le signal LigneLena ne montre pas de bons résultats par rapport aux deux autres méthodes, mais l'erreur reste dans un ordre négligeable :



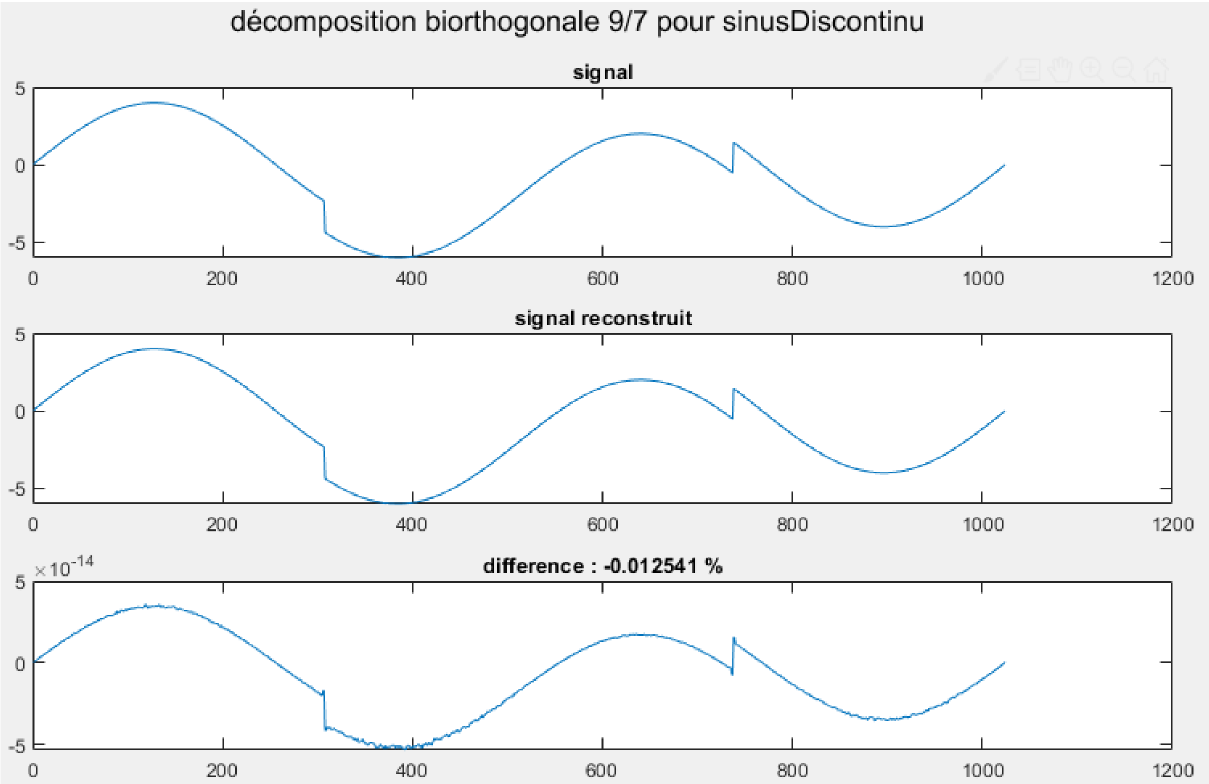

Faisons la même expérience avec cette fois le signal 'sinusDiscontinu' :




Le pourcentage d'erreur cette fois est clairement plus élevé que pour un signal tel que 'ligneLena', de l'ordre de $10^{10}$ fois plus grand.

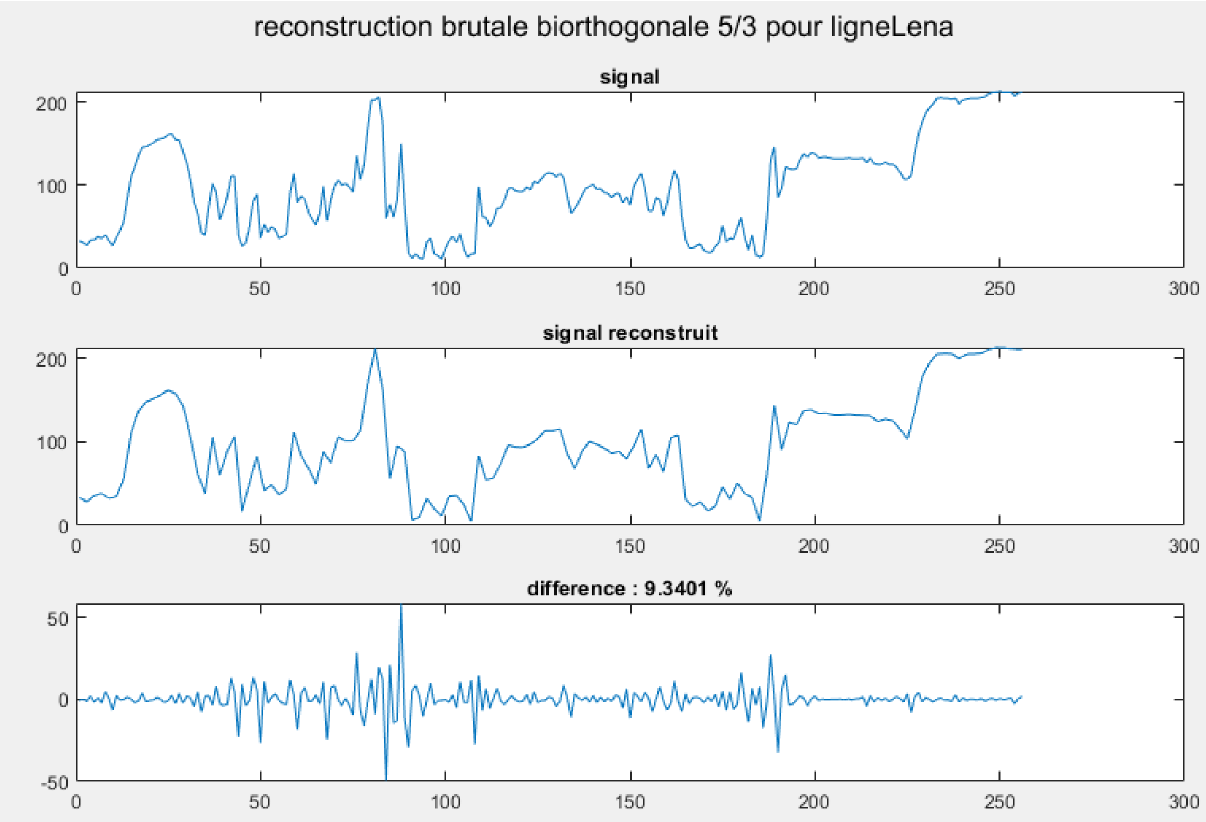



L'erreur cette fois n'est pas nulle mais son ordre de grandeur est toujours négligeable.




Pour cette méthode, l'ordre de grandeur est preque non négligeable.

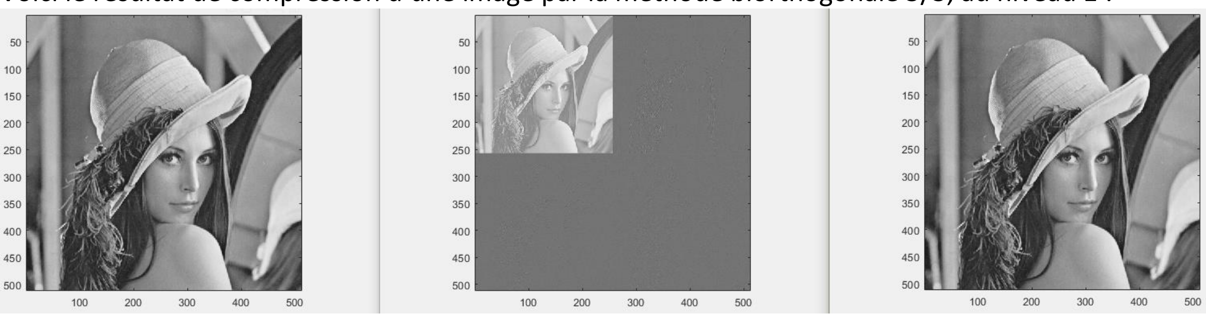
Voici le résultat d’une reconstruction sans prendre en compte le signal w (ou détail) :



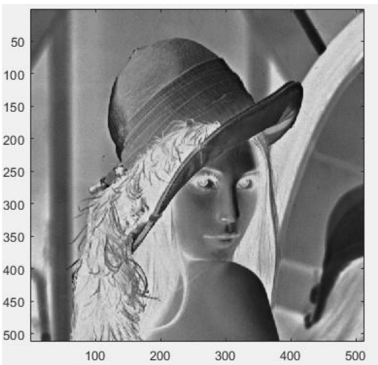

On remarque évidemment que l’ordre de grandeur de l’erreur est clairement non négligeable cette fois. Cependant le signal reste reconnaissable et lisible pour une taille clairement bien inférieur.

## Compression d’une image

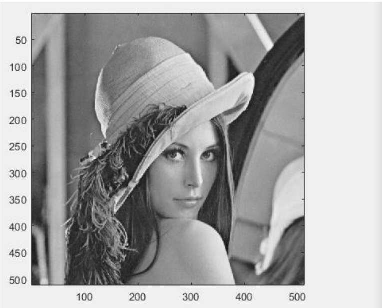
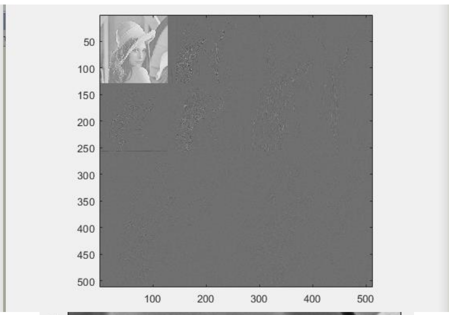
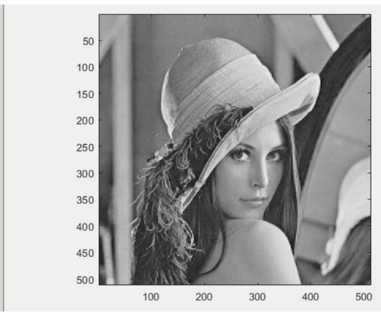
On décompose les images en appliquant les filtres aux lignes puis aux colonnes.
On applique cette méthode une deuxième fois au niveau 2.
Voici le résultat de compression d’une image par la méthode biorthogonale 5/3, au niveau 1 :



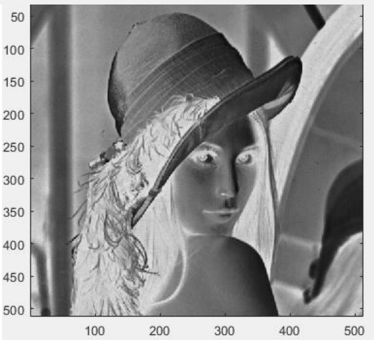
Voici l’image de différence :



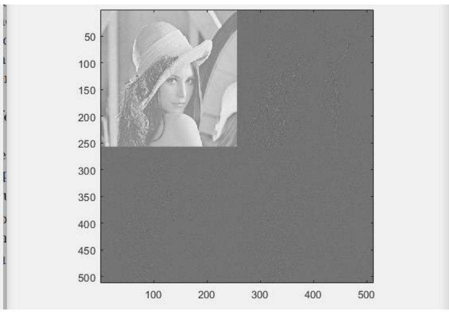
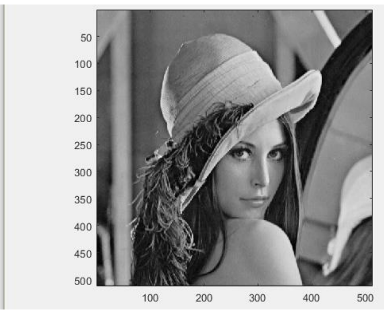
Voici le résultat de la compression d’une image par la méthode biorthogonale 5/3, au niveau 2 :

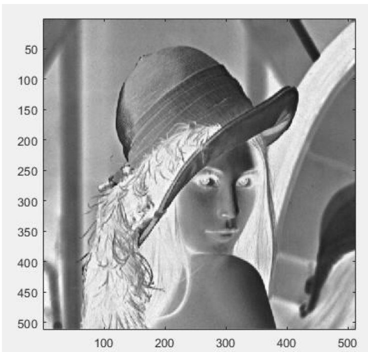




Voici l’image de différence :



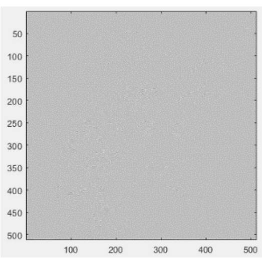
On remarque sur la méthode double une légère décoloration : l’image semble plus claire. Le résultat au niveau 1 cependant semble être une reconstruction parfaite.



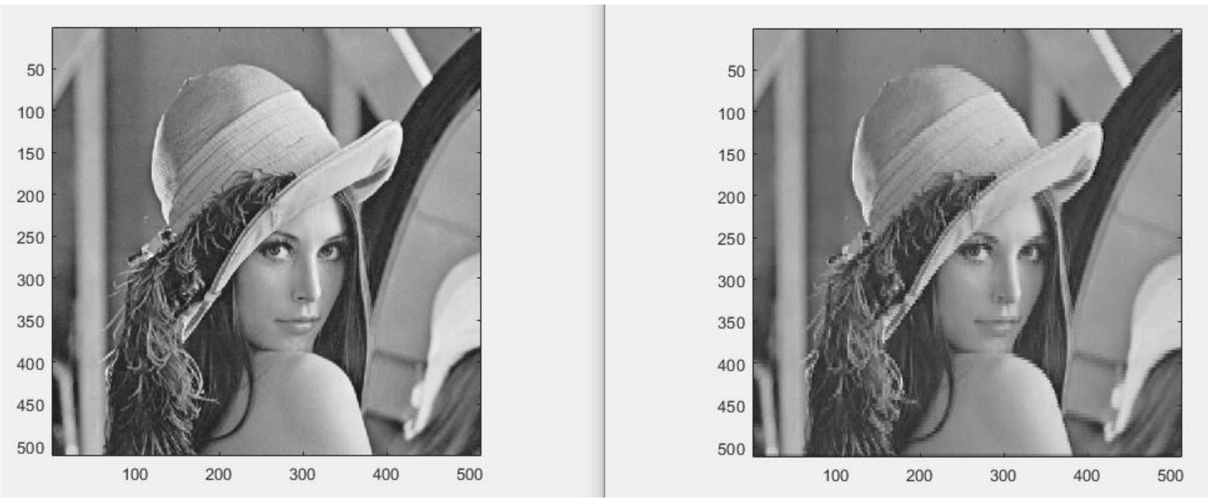
En enlevant la composante v11, on peut percevoir quelques différences à l’œil nu : certains endroits semblent lissés, les niveaux de gris semblent plus unifiés. On ne remarque pas de détérioration de l’image à l’œil nu pour la méthode biorthogonale 5/3.

Voici l’image reconstruite lorsqu’on enlève la composante v00 :

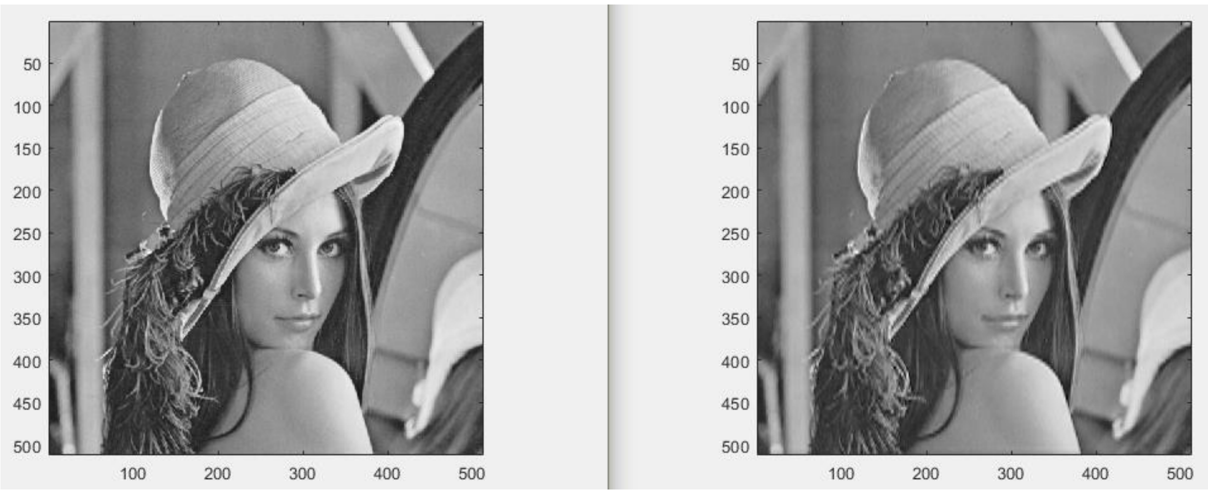


On remarque cette fois que l’image a été totalement détériorée.

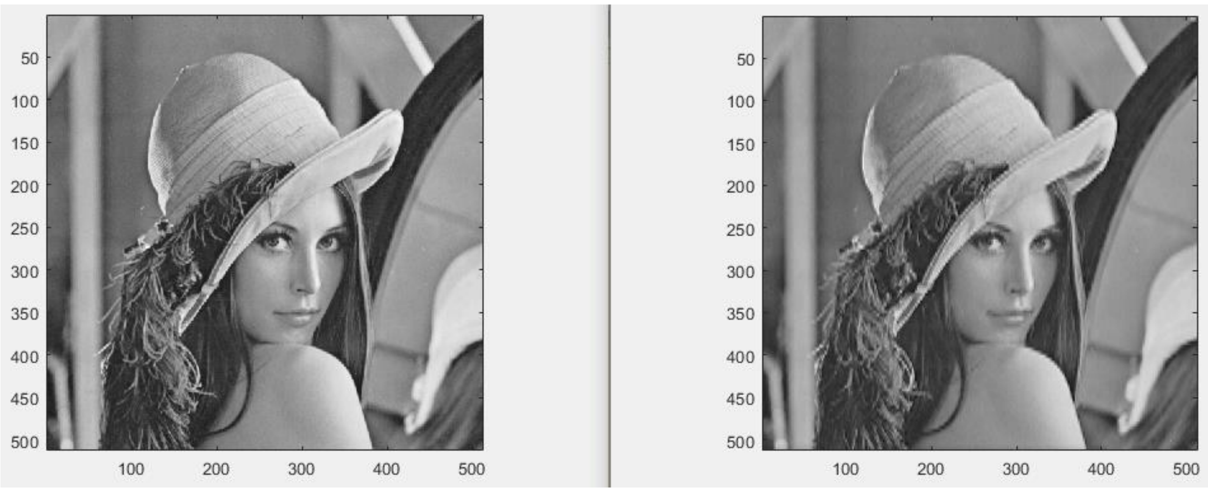
Avec la méthode de Haar à un niveau deux en supprimant la composante v00, on trouve le résultat suivant :




On peut remarquer que l'image reconstruite (à droite) a été détériorée, pixélisée.




Même chose avec la méthode biorthogonale 9/7, cette fois la pixellisation est moins marquée.




Avec la méthode biorthogonale 5/3, la pixellisation est encore un peu moins marquée.

Le calcul du PSNR n’a pas abouti, malgré les conseils du professeur qui m’a assuré que j’utilisais la bonne formule.

# Conclusion

Au travers de ce TP, nous avons pu mettre en application différentes méthodes vues en cours pour compresser des signaux musicaux ou des images selon différentes méthodes. Les différentes méthodes ont des avantages et des inconvénients, mais les méthodes bi-orthogonales, et d’avantage la méthode 5/3, semblent être plus efficaces que les méthodes orthogonales.