

Projet: Comparaison de deux méthodes non locales de débruitage d'images

Adam Fayret

23 novembre 2025

1 Introduction

Ce rapport présente le projet d'imagerie : Comparaison de deux méthodes non locales de débruitage d'images. Ce projet comporte deux grandes parties, l'implémentation de la méthode *Non Local Means* (NLM) et l'implémentation de *Block Matching 3D* (BM3D). Au vu des complications pour faire ce projet en groupe certaines parties n'ont pas pu être finalisée.

2 Non Local Means

2.1 Explication de la méthode Non-Local Means

La méthode Non Local Means est une méthode qui se base sur les redondances de l'image et a été évoqué dans le cours de filtrage pour la restauration. Son but est de remplacer la couleur d'un pixel avec la couleur des pixels qui lui sont similaires. Mais un pixel dit similaire n'a pas de raison d'être proche du pixel étudié. Cette méthode s'appuie sur ce constat pour utiliser une méthode dite non locale. Pour cela on vient scanner des patchs de l'image dans une fenêtre de recherche où la distance utilisé est la distance euclidienne entre les patchs. Dans le document fourni sur IPOL il est question de deux méthodes de NLM, nous avons donc implémenté ces deux versions en Python :

- **NLM Pixelwise** : Chaque pixel est traité individuellement en calculant une moyenne pondérée des pixels ayant des voisins similaires dans une fenêtre de recherche.
- **NLM Patchwise** : Les pixels sont traités par patches, ce qui réduit la redondance des calculs et améliore la qualité du débruitage.

Dans les deux cas le calcul des poids est

$$w(p, q) = e^{-\frac{\max(d^2 - 2\sigma^2, 0.0)}{h^2}}$$

2.2 Résultats de la méthode (à mi-parcours)

La figure 1 présente une comparaison visuelle des résultats obtenus avec les deux méthodes NLM.

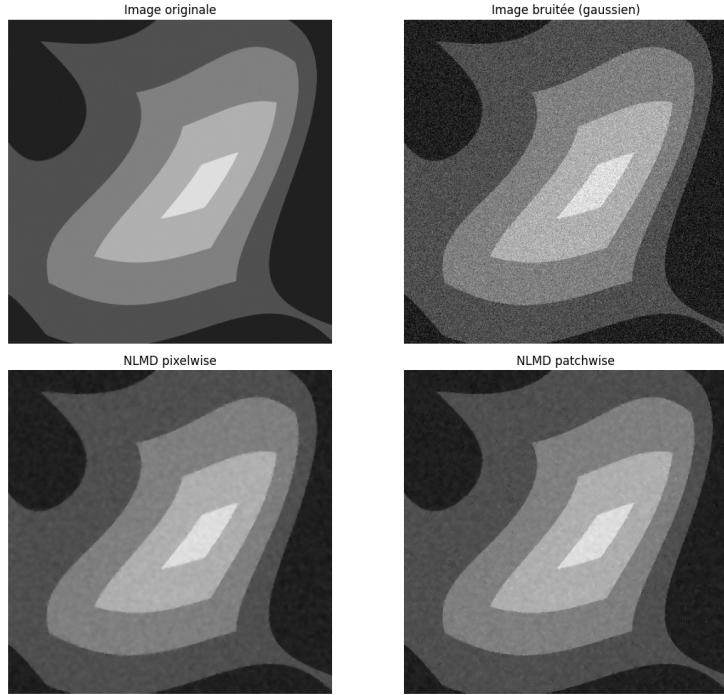


FIGURE 1 – Resultats comparatif des debruitages en NLM

2.3 Observations

- Le NLM Patchwise offre une meilleure préservation des détails et une réduction plus efficace du bruit par rapport au NLM Pixelwise.
- Les temps de calcul sont plus élevés pour le NLM Patchwise, mais cela reste acceptable pour des images de taille modérée.

2.4 Amélioration de la méthode

J'ai mis en place un padding sur les images ainsi qu'une normalisation. Cela permet d'améliorer le résultat de la méthode patchwise. J'ai aussi pris en compte les réglages des hyperparamètres du papier de recherches afin d'améliorer les résultats. Ces deux améliorations gardent le temps de calcul assez loin (peut-être un problème de ma machine) mais qui permettent des résultats très bons.

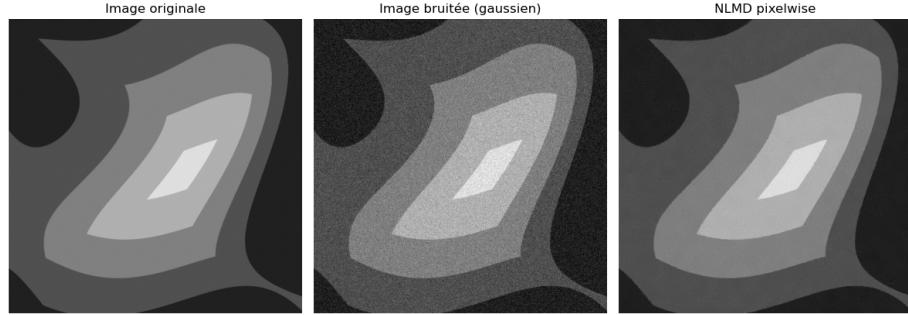


FIGURE 2 – Resultats finaux après amelioration

FIGURE 3 – Voici une image bruitée avec un bruit gaussien où $\sigma=15$ et $h=0.4*\sigma$



FIGURE 4 – Resultats finaux après amelioration

FIGURE 5 – Voici une image bruitée avec un bruit gaussien où $\sigma=20$ et $h=0.4*\sigma$

2.5 Résultats

Pour valider l'efficacité de la méthode de débruitage j'ai utilisé le Peak Noise to Signal Ratio. La méthode est directement implémenté dans skimage.metrics. Pour l'exemple avec l'image Lena le PNSR obtenu est de 28.46 dB. Dans un second document la méthode PSNR est aussi utilisée et fourni pour une image de bruitage de variance 21 un PSNR de 18.48dB. Ces résultats sont meilleurs mais avec un NLM beaucoup plus optimisé (explication si après).

2.6 Remarques :

Le bruit Gaussien simulé dans les exemples est plutôt cohérent avec l'utilisation faite de la méthode de débruitage notamment en imagerie médicale. Dans le document fournis plus tard par mon encadrante, *An Optimized Blockwise Non*

Local Means Denoising Filter for 3D Magnetic Resonance Images, la modélisation du bruit est fait aussi par un bruit gaussien. Il y est aussi proposé des améliorations notamment une implémentation block-wise, un réglage automatique de l'hyperparamètre h , une sélection des pixels pertinents (grâce à l'image faite en 3D) et des calculs en parallèle. La sélection des pixels(voxels) pertinents grâce à l'image en 3D est impossible dans le cas du projet mais la mise en place du calibrage automatique de l'hyperparamètre h lui peut être intéressante pour éviter de se baser uniquement sur le document de recherche.

3 Block Matching 3D

3.1 Explication de la méthode Block-Matching 3D

La méthode Block-Matching 3D est une méthode de débruitage qui exploite elle aussi la redondance des motifs dans une image. Le document fourni par IPOL décrit la méthode en la séparant en deux étapes.

3.2 Etape 1 :

On vient séparer l'image en blocs de pixels puis on recherche les blocs similaires autour d'eux dans une fenêtre de recherche (on utilise la distance euclidienne). Une fois les blocs similaires trouvés on vient les empiler pour leur appliquer une transformation 3D (j'ai choisi la transformation en ondelette). Suite à la transformation on vient appliquer un seuillage afin de filtrer le bruit puis la transformation 3D inverse. Enfin la fonction d'aggregation vient combiner les différentes estimations des pixels données par les patchs proches. À la fin de cette étape une première estimation de correction des patchs est obtenue.

3.3 Résultats :

Pour la méthode BM3D implementée voici les résultats obtenus pour un patch et pour une image.

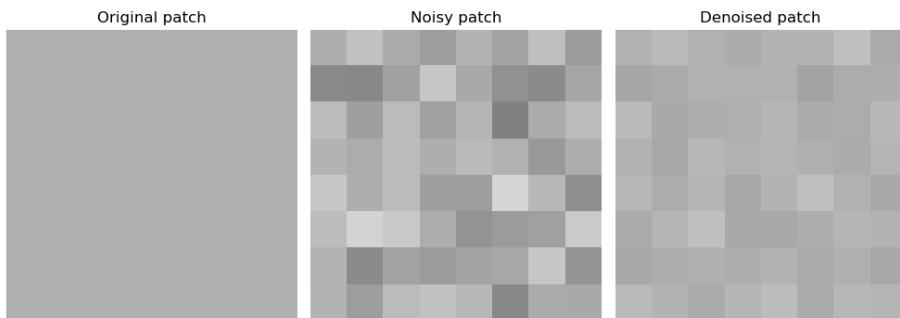


FIGURE 6 – Résultats sur un patch

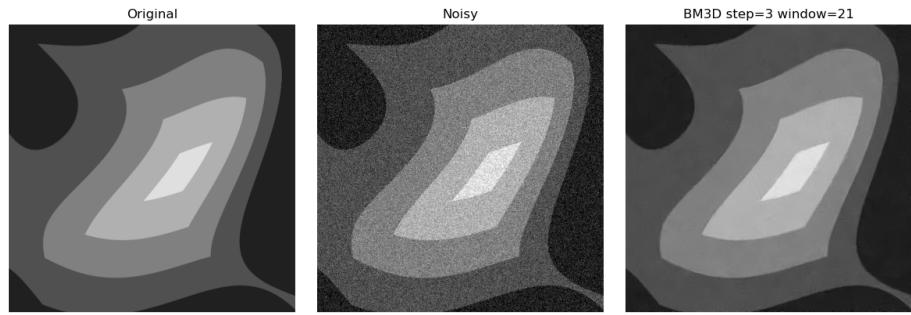


FIGURE 7 – Resultats sur une image

3.4 Etape 2 :

La seconde étape décrite dans le papier vient prendre en compte la première étape de débruitage comme heuristique afin de reprendre les étapes précédentes avec un filtre de Wiener qui améliore les résultats obtenus lors du premier calcul.

4 Conclusion