



Projet Image

Débruitage d'images

Compte-rendu n°7

Adam Bonbon

Alaric Hunot-Martin

Louis Jean

Master 2 IMAGINE
Université de Montpellier

1er décembre 2024

Table des matières

1	Introduction	2
2	NAFNet	2
2.1	Résultats expérimentaux	2
2.2	Performances sur le bruit sel et poivre	2
3	Ajout de GAN à notre modèle	3
4	Débruitage Vidéo avec notre Modèle	4
4.1	Notre modèle d'image appliqué à la vidéo	4
4.2	Entraînement sur Vidéo	5
4.3	Essai du modèle préentraîné NAFNet sur DAVIS	5
5	Interface graphique	6
6	Conclusion	7
7	Problèmes techniques et perspectives	7

1 Introduction

Cette semaine, nous avons continué d'évaluer NAFNet et avons quasiment finalisé notre application graphique.

2 NAFNet

Nous avons évalué les performances du modèle de débruitage NAFNet sur deux types de bruit : **sel et poivre** et **gaussien**. Ces tests ont été réalisés sur un ensemble de 300 images en couleur bruitées, prétraitées à l'aide de scripts Python.

L'objectif principal était de comparer l'efficacité de NAFNet sur ces deux types de bruit et de déterminer la pertinence du modèle pour un débruitage généralisé.

2.1 Résultats expérimentaux

2.2 Performances sur le bruit sel et poivre

Les métriques suivantes ont été obtenues :

- **PSNR moyen (Bruitée vs Original)** : 17.59 dB
- **SSIM moyen (Bruitée vs Original)** : 0.370
- **PSNR moyen (Débruitée vs Original)** : 19.42 dB
- **SSIM moyen (Débruitée vs Original)** : 0.448

2.2.1 Performances sur le bruit gaussien

Les résultats pour ce type de bruit sont :

- **PSNR moyen (Bruitée vs Original)** : 19.70 dB
- **SSIM moyen (Bruitée vs Original)** : 0.309
- **PSNR moyen (Débruitée vs Original)** : 26.05 dB
- **SSIM moyen (Débruitée vs Original)** : 0.694

2.2.2 Analyse des résultats

Les scores PSNR et SSIM montrent que le modèle NAFNet est significativement plus performant sur les images bruitées de type gaussien. Pour le bruit sel et poivre, les améliorations restent limitées, ce qui suggère que NAFNet est moins adapté à ce type de bruit.

2.2.3 Scores NIMA

Pour évaluer la qualité des images, nous avons utilisé une nouvelle fois les scores NIMA sur différents ensembles d'images. Les résultats moyens sont les suivants :

- **Images originales** : 5.310
- **Bruit sel et poivre** : 5.311
- **Bruit gaussien** : 4.886

- Débruitage (sel et poivre) : 4.905
- Débruitage (gaussien) : 5.103

Observation : Curieusement, le score NIMA des images bruitées sel et poivre est légèrement supérieur à celui des images originales, ce qui peut suggérer une sensibilité particulière de cette métrique à ce type de bruit. Cependant, pour le bruit gaussien, les scores montrent une amélioration cohérente après débruitage, indiquant que NIMA est peut-être plus adapté pour évaluer ce type de bruit.

3 Ajout de GAN à notre modèle

Le GAN (*Generative Adversarial Network*) est conçu pour débruiter des images corrompues par un bruit combiné, comprenant du bruit gaussien et du bruit sel et poivre. Il est constitué de deux modules principaux : un **générateur** et un **discriminateur**, chacun jouant un rôle distinct dans le processus de débruitage.

1. Le Générateur (CGNet)

Architecture : Le générateur est un réseau convolutionnel avec trois couches de convolution 3×3 , des activations *ReLU*, et une sortie à trois canaux (RGB). Il prend une image bruitée en entrée et produit une image supposément débruitée.

Rôle dans le débruitage : Le générateur transforme une image bruitée en une version propre en minimisant les pertes entre l'image générée et l'image originale. L'entraînement à "tromper" le discriminateur améliore la qualité visuelle et la fidélité des détails.

2. Le Discriminateur

Architecture : Le discriminateur est un réseau convolutionnel conçu pour classifier une image comme étant "propre" ou "générée". Il utilise des noyaux 4×4 , des activations *LeakyReLU*, et des normalisations par batch (*BatchNorm*). La sortie est une probabilité, calculée via une fonction *Sigmoïde*.

Rôle dans le débruitage : Le discriminateur guide le générateur à produire des images plus réalistes en fournissant un retour sur la qualité des images générées.

3. Ajout de Bruit Combiné

Pour simuler des scénarios réalistes, le dataset applique un bruit combiné :

- **Bruit gaussien** : Simule des erreurs d'acquisition d'images.
- **Bruit sel et poivre** : Simule des dégradations aléatoires en pixelisant l'image.

4. Fonctions de Perte

- **Perte pixel-wise (MSE)** : Mesure les différences pixel par pixel entre l'image générée et l'image propre, guidant le générateur dans la reconstruction globale.
- **Perte des Hautes Fréquences** : Évalue la capacité du générateur à préserver les détails structurels (contours et textures).
- **Perte adversariale (BCE)** : Entraîne le générateur à produire des images suffisamment réalistes pour tromper le discriminateur.

Résumé des Contributions au Débruitage

- Le générateur supprime efficacement le bruit tout en préservant les détails visuels.
- Le discriminateur pousse le générateur à produire des images plus réalistes.
- Les pertes combinées équilibrent la fidélité numérique et la qualité perceptuelle.
- L'ajout de bruit combiné renforce la robustesse du modèle face à des scénarios réels.



Figure 1: nouveau résultat avec GAN

4 Débruitage Vidéo avec notre Modèle

4.1 Notre modèle d'image appliqué à la vidéo

Nous avons mis en place un code pour effectuer le **débruitage de vidéos** en utilisant un modèle préentraîné basé sur CGNet. L'idée était de tester notre modèle sur des vidéos afin d'évaluer sa capacité à supprimer efficacement différents types de bruit tout en préservant les détails visuels. Le processus fonctionne de manière simple : chaque vidéo est traitée comme une séquence d'images (frames). On commence par ajouter un bruit combiné sur les frames, avec un mélange de bruit gaussien (écart-type de 20) et de bruit sel et poivre (probabilité de 0.05). Ensuite, chaque frame bruitée est passée dans le modèle CGNet pour être restaurée. À la fin, toutes les frames restaurées sont assemblées pour recréer une vidéo débruitée.

Nous avons testé ce code sur (environ 50 vidéos) issues du dataset DAVIS. Les résultats sont satisfaisants : le modèle parvient à supprimer efficacement le bruit ajouté tout en préservant les détails essentiels des vidéos. Ce pipeline démontre que notre modèle, initialement conçu pour des images, peut également bien fonctionner sur des vidéos, offrant une qualité visuelle améliorée et une suppression des artefacts de bruit convaincante.



Figure 2: frame noised



Figure 3: frame denoised

4.2 Entrainement sur Vidéo

Nous avons essayé de développer un modèle spécifiquement conçu pour traiter des vidéos, afin d'améliorer les performances de débruitage sur le dataset DAVIS. Ce modèle utilise une architecture plus avancée, avec des blocs résiduels et un système d'encodage/décodage, pour mieux s'adapter aux caractéristiques des séquences vidéo.

Pour l'entraînement, nous avons extrait les frames des vidéos et ajouté un bruit combiné (gaussien et sel et poivre) pour simuler des conditions dégradées. Le modèle apprend ensuite à restaurer ces frames bruitées en les rapprochant des images originales.

Cependant, l'entraînement est très long en raison de la taille des données et de la complexité du modèle. Le processus est encore en cours et ne devrait probablement se terminer que demain. Cela reflète les défis liés à l'utilisation de vidéos haute résolution pour un débruitage précis.

4.3 Essai du modèle préentraîné NAFNet sur DAVIS

Nous avons essayé d'utiliser le modèle **NAFNet**, pré-entraîné pour le débruitage, afin de traiter les vidéos du dataset **DAVIS**. Le modèle a été appliqué pour restaurer les frames des vidéos bruitées et produire des vidéos débruitées.

Cependant, les résultats montrent que le modèle parvient à réduire une partie du bruit, mais le rendu global reste assez médiocre, avec des artefacts visuels importants et des zones souvent ternes, donnant parfois un effet proche du noir et blanc.

Bien que le modèle fonctionne correctement sur certaines images fixes, son adaptation aux vidéos complètes semble insuffisante pour garantir une bonne qualité sur DAVIS. Cela peut être dû à des différences dans les distributions de données entre le dataset utilisé pour l'entraînement initial et celui de DAVIS.

5 Interface graphique

Cette semaine, nous avons grandement amélioré l'application graphique, en intégrant toutes les méthodes que nous souhaitions. Nous avons aussi amélioré le confort de navigation pour l'utilisateur. Voici toutes les méthodes disponibles et fonctionnelles pour l'instant dans l'application :

- Filtre médian
- Filtre moyenneur
- Filtre bilatéral
- Filtre de Wiener
- Variation totale
- Ondelettes de Haar
- BM3D

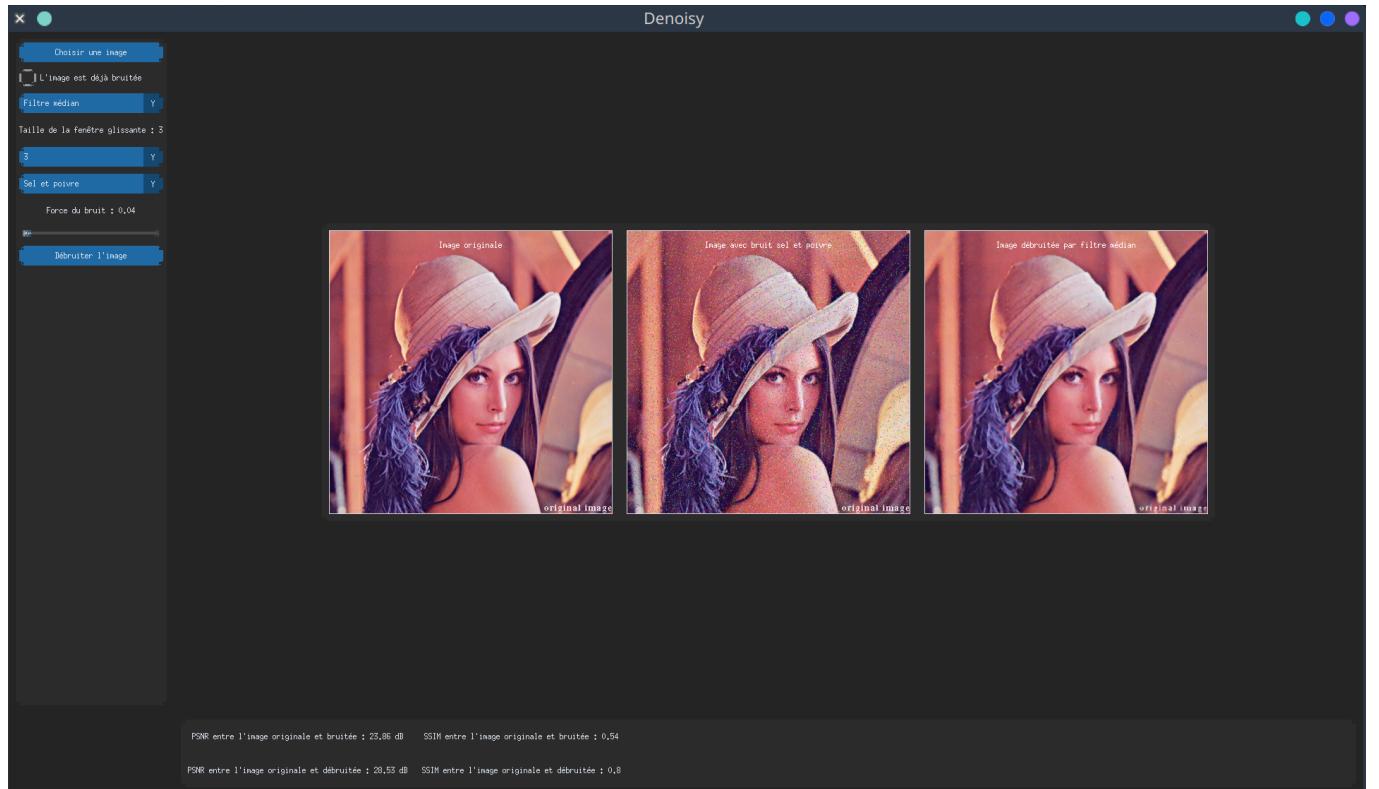


Figure 4: Interface graphique

Nous avons également rajouté le SSIM comme métrique de qualité.

Cependant, il reste encore les modèles CNN à intégrer dans l'application (CGNet et NAFNet) ainsi que l'évaluation de la qualité des images par NIMA. L'application est prête à les recevoir, mais nous devons bien configurer nos fichiers .pth et .k5 d'abord. Nous le ferons lundi tous ensemble.

6 Conclusion

Les résultats obtenus montrent une performance prometteuse de NAFNet pour le bruit gaussien. Notre application commence à être utilisable par le public.

7 Problèmes techniques et perspectives

Lors de l'une de nos sessions, un problème avec le GPU a interrompu l'entraînement du modèle, limitant ainsi nos expérimentations. Nous espérons pouvoir reprendre ces tests lundi prochain. Dans le cas contraire, comme perspective, il sera nécessaire de rechercher un autre modèle capable de mieux gérer les images bruitées de type sel et poivre, afin d'améliorer les résultats de débruitage pour ce cas spécifique.

La semaine prochaine, nous allons faire les derniers correctifs / petites améliorations sur notre projet, et allons commencer la préparation de la vidéo, puisque nous avons déjà le script.

Merci pour le temps et l'attention que vous avez consacrés à la lecture de ce compte-rendu.