



# HAI927I

## Projet Image - Débruitage

### Compte rendu n°2

Rey Emilien

Reynier Théo

M2 IMAGINE  
Faculté des Sciences  
Université de Montpellier

15 Novembre 2025



### Résumé

L'objectif du projet est de créer un encodeur, auto-encodeur et un GAN spécialisé dans le traitement des images bruitées. Ce deuxième compte rendu décrira nos avancées et les changements par rapport à ce que nous avons dit dans le premier compte rendu.

# 1 Recherches

Après plusieurs recherches, nous avons décidé de ne finalement pas utiliser DnCNN. Nous n'avons pas trouvé de pistes qui nous permettraient d'utiliser ce modèle efficacement notamment pour la partie décodage.

Nous avons donc continué nos recherches et nous avons trouvé l'architecture U-Net. Il s'agit d'un auto-encodeur convolutionnel symétrique conçu à l'origine pour la segmentation d'images mais qui peut être étendu pour le débruitage d'images. D'après nos recherches, cela semble être le meilleur point de départ pour la suite du projet.

Pour la partie du GAN, nous sommes encore en train de rechercher comment réaliser cette partie et comment l'intégrer avec l'autoencodeur.

## 2 Tests avec les images de la base CIFAR-10

Nous avons récupéré une base implémentant un modèle U-Net pour le débruitage d'images. Nous l'avons testée avec les images provenant de CIFAR-10 que nous avons bruitées la semaine dernière.

Les résultats obtenus étaient très mauvais malgré 10000 images pour l'entraînement et plusieurs epochs. Nous pensions que le problème provenait du fait que les images étaient trop petites. Nous avons donc essayé sur des images plus grandes. Finalement, le problème venait d'une mauvaise correspondance entre les images normales et bruitées lors de l'entraînement.

Nous avons donc pu obtenir les résultats suivants :

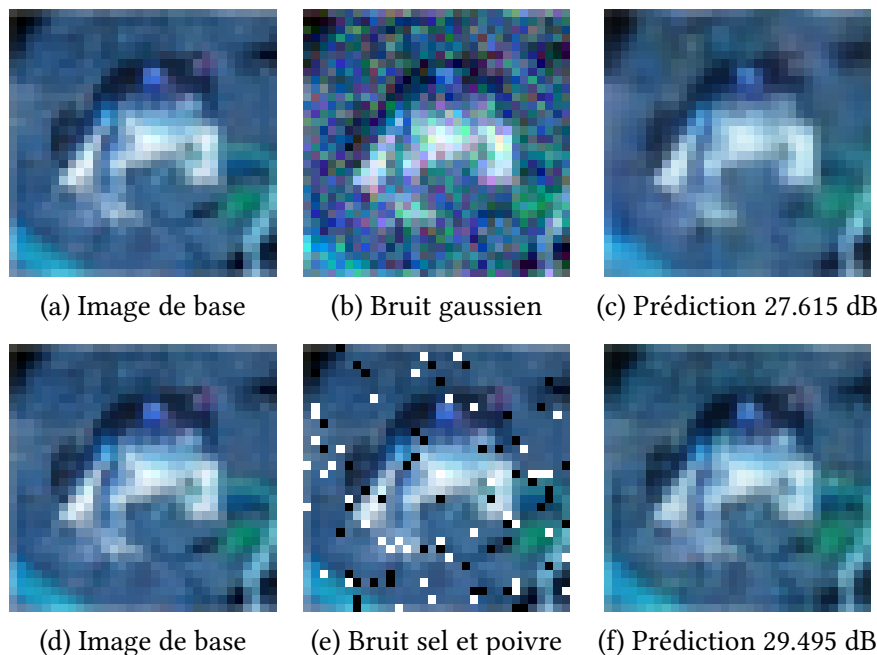


Fig. 1. – Bruitage et débruitage d'une image de la base CIFAR-10

On voit que le résultat est légèrement meilleur sur les images bruitées avec le bruit sel et poivre. On pense que c'est dû au fait que ce bruit est beaucoup plus facilement identifiable que le bruit gaussien. On remarque aussi sur le bruit gaussien que le résultat semble légèrement flouté par rapport à l'image d'origine. Le fait que l'image de base soit en  $32 \times 32$  et donc très petite fait partie du problème mais on voit que pour sel et poivre nous n'avons pas ce problème.

### 3 Tests avec des images plus grandes

Afin de voir si le problème était la taille des images, nous avons également récupéré un échantillon d'images sur [images.cv](https://images.cv) de taille  $128 \times 128$ . Nous avons donc testé sur cet échantillon (environ 1000 images) avec du bruit gaussien et du bruit sel et poivre.

Nous avons obtenu les résultats suivants :

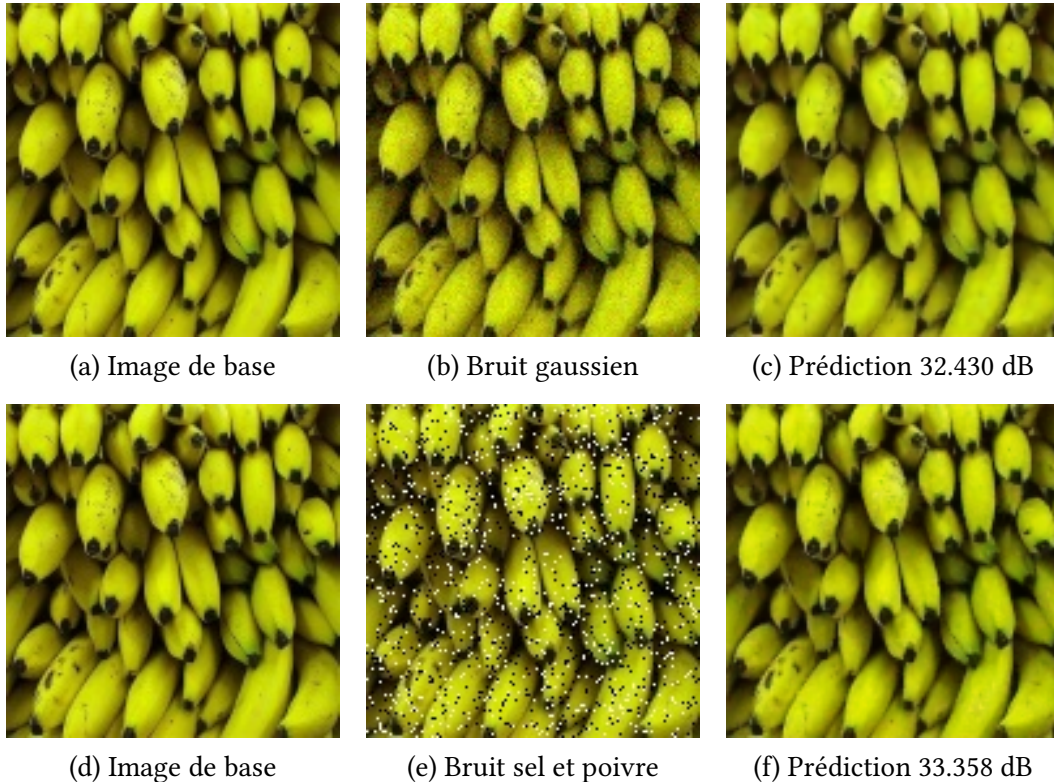


Fig. 2. – Bruitage et débruitage d'une image  $128 \times 128$

Malgré le faible nombre d'images, nous avons obtenu plus de 30 dB en utilisant le PSNR ce qui montre une qualité d'image satisfaisante. Même à l'œil nu, nous ne discernons que très peu de différences entre l'image débruitée et l'image d'origine.

### 4 Observations

- La taille des images peut contribuer à l'augmentation du PSNR. Cependant, plus les images sont grandes plus le temps de calcul sera long.
- La quantité de bruit sur une image dégrade fortement la reconstruction.
- La reconstruction depuis un bruit sel et poivre donne un meilleur PSNR qu'un bruit gaussien.

### 5 À venir

Pour la semaine suivante :

- Ajouter davantage d'images  $128 \times 128$  pour voir si le PSNR augmente davantage.
- Essayer le bruit de Poisson.
- Trouver les paramètres optimaux pour maximiser le PSNR.
- Trouver et commencer à implémenter le GAN

## 6 Bibliographie

- [CIFAR-10 dataset](#) , vaste dataset d'images  $32 \times 32$ .
- [DnCNN](#) , Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising
- [Poivre et Sel](#) , type de bruit dans une image.
- [Gaussien](#) , type de bruit dans une image.
- [Poisson](#) , type de bruit dans une image.
- [GAN](#) , GAN-based Noise Model for Denoising Real Images.