

HAI92TI - Projet image I2.2 :

Christal

MAURIN Christina - COQUERON Solal

Master 2 Informatique IMAGINE

Contexte et objectifs

Qu'est-ce que le débruitage en photographie ?



Prise de vue



Photo bruitée



Filtrage

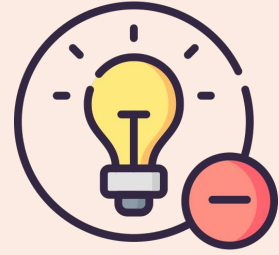


Photo débruitée



Montée des ISO

ET/OU



Faible luminosité



Notre objectif !

Méthodes traditionnelles

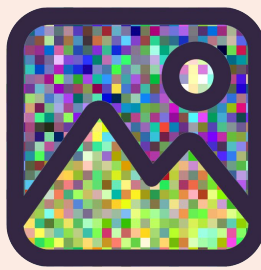
Méthode avec réseau de neurones

État de l'art : Méthodes traditionnelles

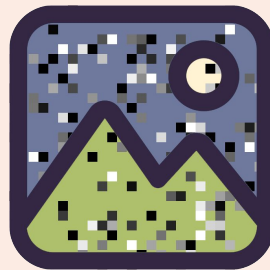
Plusieurs type de bruit :



Gaussien



Chromatique



Poivre et Sel

• Plusieurs type de filtres :

1/16 x

1	2	1
2	4	2
1	2	1

Gaussien

a	b	c
d	e	f
g	h	i

e remplacé par médiane de [a,i]

Médian

1/9 x

1	1	1
1	1	1
1	1	1

Moyenneur

$$W(i, j) = e^{-\frac{(i-x)^2 + (j-y)^2}{2s^2}} \cdot e^{-\frac{(I(i, j) - I(x, y))^2}{2r^2}}$$

Bilatéral

0	1	0
1	-4	1
0	1	0

Laplacien

+ EPDFP : Efficient Poisson Denoising for Photography

Méthodes traditionnelles : EPDFP

Papier : *Efficient poisson denoising for photography*

- Bruits liées aux capteurs d'images (bruit de photon)
- Transformation en image avec photons par pixels
- Transformation du bruit de poisson en bruit gaussien
- Appliquer un filtre classique
- Transformations inverses pour avoir l'image débruitée



Notre
implémentation

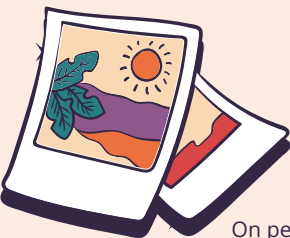




Nous avons beaucoup trop de possibilités à tester pour pouvoir toutes les afficher, voici donc quelques exemples significatifs



Image originale



On peut aussi utiliser des images en niveaux de gris

Résultats obtenus

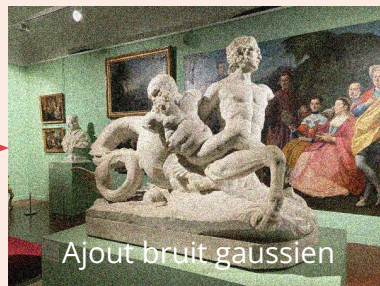


0.02

3



PSNR : 29.50 dB
SSIM : 0.86
RMSE : 8.54
SNR : 23.72 dB
BRISQUE : 28.59



20

9

75

75



PSNR : 26.25 dB
SSIM : 0.60
RMSE : 14.43
SNR : 19.15 dB
BRISQUE : 46.66



20



PSNR : 27.04 dB
SSIM : 0.78
RMSE : 11.91
SNR : 20.83 dB
BRISQUE : 14.43



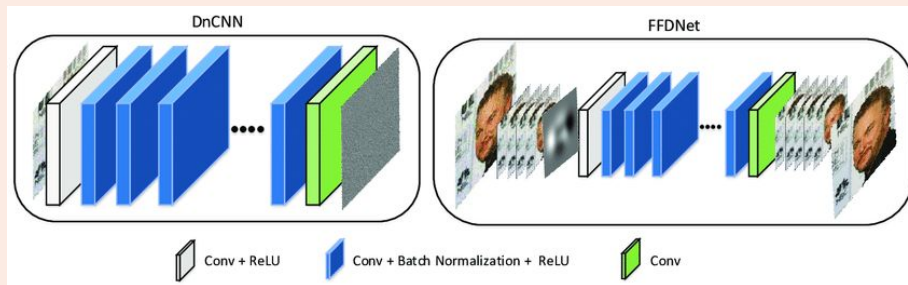
×5

État de l'art : Réseaux de neurones

Méthodes existantes :

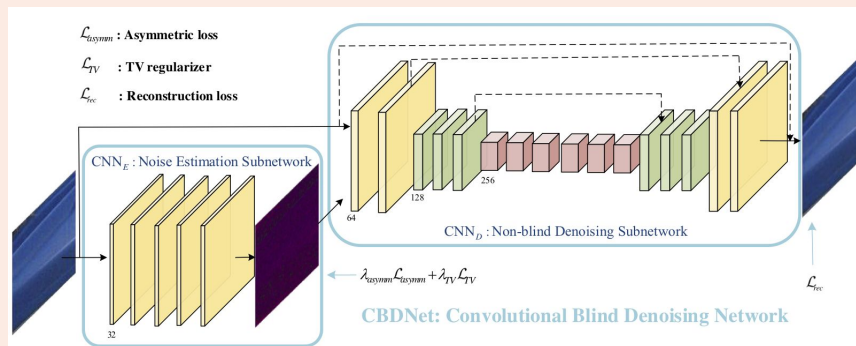
DnCNN

FFDNet



Méthode essayée :

CBDNet



Méthode retenue :

MPRNet



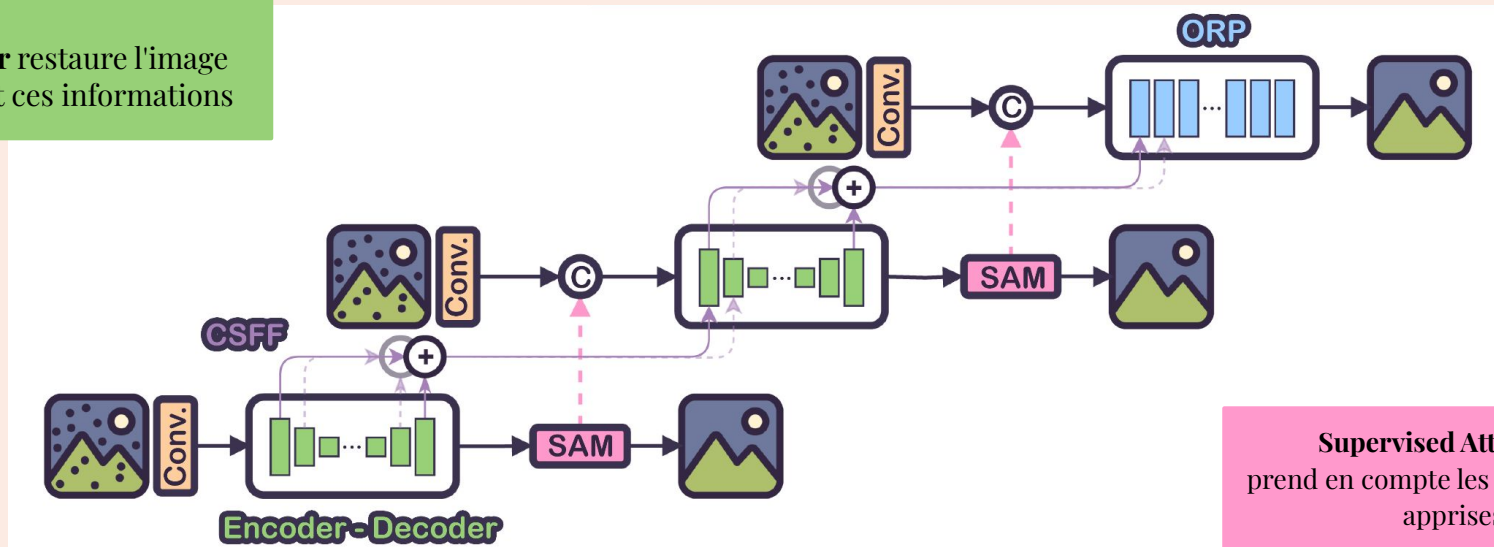
Architecture du réseau de neurones

Multi-Stage Progressive Image Restoration

- **Encodeur** capture informations à différentes échelles
- **Décodeur** restaure l'image en utilisant ces informations

3 étapes

Original Resolution Processing : Traitement de l'image à sa résolution d'origine → préservation des détails spatiaux fins



Supervised Attention Module : prend en compte les caractéristiques apprises à chaque étape

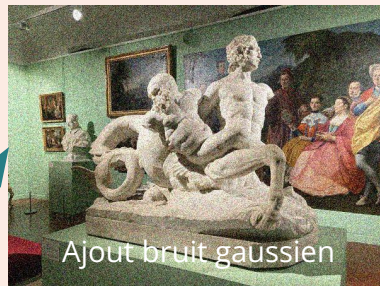
Crossed-Stage Feature Fusion : propagation des caractéristiques contextualisées à plusieurs échelles

Résultats obtenus

MPRNet Denoising



Image originale



20



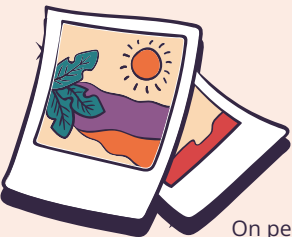
PSNR : 24.04 dB
SSIM : 0.78
RMSE : 16.01
SNR : 18.48 dB
BRISQUE : 46.02



10

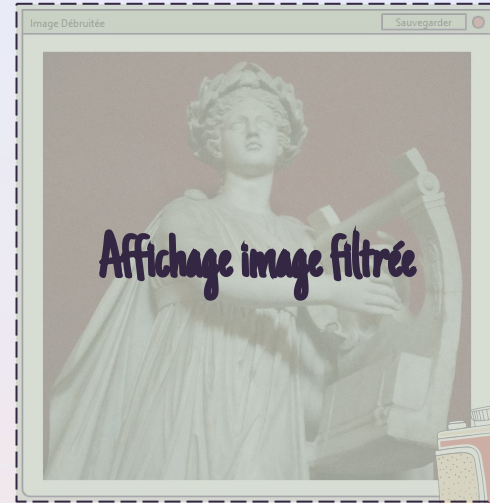
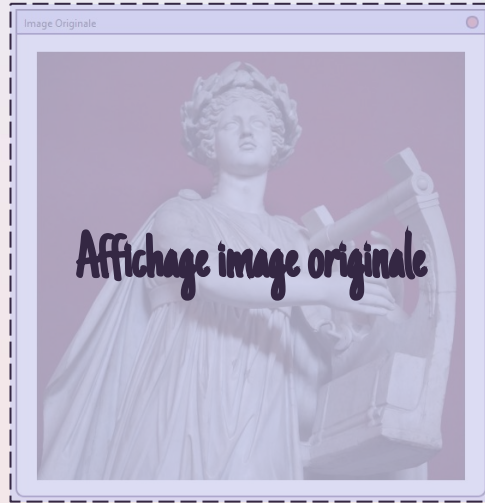
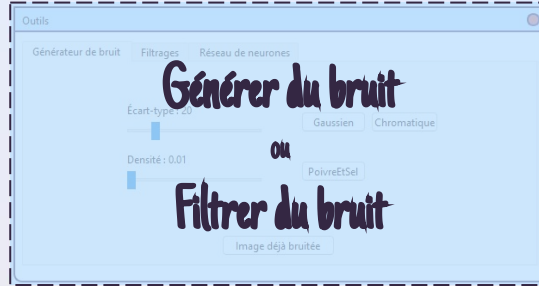


PSNR : 29.95 dB
SSIM : 0.86
RMSE : 8.11
SNR : 24.21 dB
BRISQUE : 22.64



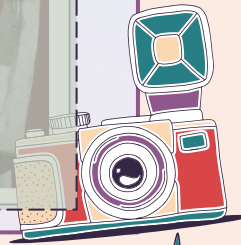
On peut aussi utiliser des images en niveaux de gris

Interface graphique



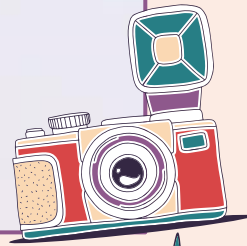
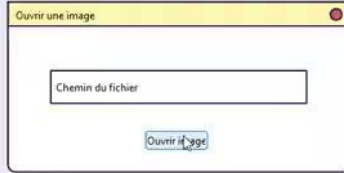
Application développée sur Qt Creator Python

Christal



Démonstration

Christal



Perspectives et conclusion

Améliorations :

Rapidité d'exécution sur des images capturées par des APNs à grande résolution

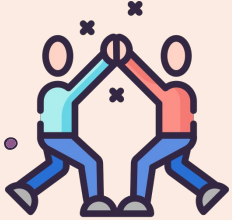
Cela fonctionne mais beaucoup trop lent (Sony a6400 : 4000 x 6000)

Correction de légers bugs et création d'un exécutable

Lancement de l'application via Qt Creator

Changement des paramètres du modèle (MPRNet)

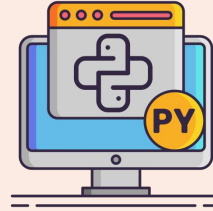
En résumé :



Progression gratifiante
au fil des semaines



Nouvelles
connaissances

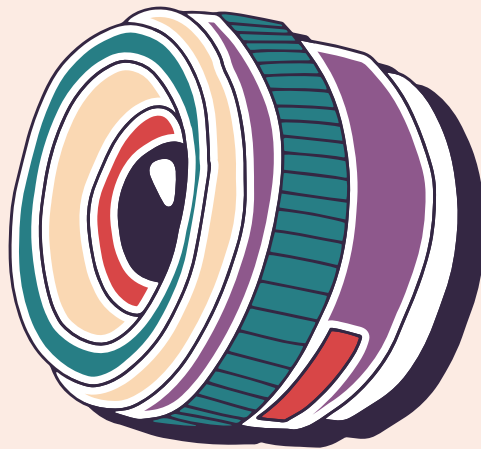


Amélioration visuelle
des algorithmes



Construction
Interface Interactive

Merci pour votre attention !



Annexe : métriques I



Annexe : métriques 2

Moyenne des résultats sur le SIDD small dataset

	Gaussien	Bilateral (9,75,75)	Median (3)	Moyenneur (3)	Papier	MPRNet
PSNR (dB)	33.86	34.29	33.08	33.57	30.25	34.76
SNR (dB)	22.43	22.86	21.74	22.17	18.95	23.35
SSIM	0.85	0.85	0.78	0.81	0.73	0.88
RMSE	5.81	5.62	6.57	6.09	8.98	5.22
BRISQUE	22.84	31.21	25.27	14.01	46.34	26.77

PSNR : mesure la qualité de la reconstruction d'un signal par rapport au bruit présent

SNR : mesure la puissance d'un signal par rapport à la puissance du bruit l'accompagnant

SSIM : mesure la similarité structurelle entre deux images

RMSE : mesure l'erreur moyenne entre les valeurs prédites et les valeurs réelles

BRISQUE : se base sur des caractéristiques statistiques pour estimer la qualité perçue d'une image sans image de référence

Annexe : MPRNet modèle pré-entraîné "denoising"

https://github.com/swz30/MPRNet/tree/main/Denoising/pretrained_models

Patch size : 256x256 → Taille de chaque sous-région extraite pour entraîner le modèle

Batch size : 16 → Nombre d'itérations avant de mettre à jour les poids

Nombre d'epochs : 400 000 → Nombre de fois que l'ensemble des données d'entraînement est présenté au modèle

Optimisateur Adam : 0.0002 à 0.000001 → Ajustement des poids avec diminution au fil de l'entraînement et donc convergence efficace

Annexe : résultats filtrés

Bruit gaussien

Filtre gaussien

Filtre moyennneur

Filtre médian

Filtre bilatéral

Filtre EPDFP

Filtre MPRNet



Annexe : Poster

