

RAPPORT DE PROJET

Apprentissage automatique de la chaîne de développement image

Auteurs :
Rafaël Brutti
Florian Lamalle

Encadrant :
Raphaël Achddou

Soutenu le 16 février 2022

Table des matières

1	Introduction	3
2	État de l'art	4
2.1	Méthodes de débruitage	4
2.2	Pipeline du traitement des images	4
3	Nouveau pipeline	5
4	Données	6
5	Expériences	7
6	Conclusion	11

1 Introduction

Dans le milieu de la photographie, la chaîne de développement de l'image est une étape importante permettant de traiter et restaurer l'image capturée de la meilleure des manières. Cette chaîne de développement suit plusieurs étapes allant de la capture de l'image RAW (brut) en passant par la balance des blancs et allant jusqu'au débruitage.

La capture de l'image RAW consiste à reconstituer l'information d'une scène à l'aide d'un capteur comptant les photos pendant un certain temps d'exposition. Le temps d'exposition est une variable primordiale. En effet, à cause de la nature quantique de la lumière le nombre de photos (définissant un pixel) va définir un processus stochastique et donc introduisant un bruit sur l'image. Ainsi, un temps de pose court induit une quantité de lumière moindre sur le capteur et donc plus de bruit alors qu'un temps de pose long laisse passer une plus grande quantité de lumière mais doit capturer une scène fixe sinon il y a présence de flou sur le rendu final.

La difficulté est d'autant plus grande lorsque les photos sont prises dans le noir ou plus généralement dans des endroits très peu lumineux en raison du peu de photons présents et du faible SNR (Signal-to-Noise Ratio) qui permet de quantifier la qualité de la transmission d'une information ici la qualité de restitution de la scène.

L'article étudié [[Chen chen and Koltun, 2018](#)] est le premier à présenté une différente approche pour ce genre de problème et sort du schéma classique de traitement de l'image digitale mais un traitement utilisant le deep learning en entrainant un fully- convolutional network. L'idée de l'article est de remplacer toute la chaîne de développement de l'image par un réseau prenant en entrée la donnée brute du capteur et donnant en sortie l'image RGB. Leur idée est d'utiliser pour les images d'entraînement des couples d'images de la même scène avec un faible temps d'exposition donc énormément bruité et une autre avec un long temps d'exposition qui servira de groundtruth de résultat à obtenir. Il s'agit donc d'un apprentissage supervisé.

On se propose ici de tester la généralisation de la méthode c'est-à-dire de reprendre le réseau et de l'entraîner sur nos données à nous afin de voir si les résultats sont tout aussi promettants que les leurs, d'analyser les possibles défauts du réseau ainsi que de vérifier l'ordonnancement de la trame de Bayer car a priori le réseau ne fonctionne pas si les trames de Bayer sont différentes. Une trame de Bayer est l'ensemble des filtres permettant de séparer les couleurs d'une image RGB.

On se propose également d'étudier l'adaptabilité du réseau à d'autres appareils que ceux présentés dans [[Chen chen and Koltun, 2018](#)] et d'effectuer une étude qualitative des résultats.

2 État de l'art

2.1 Méthodes de débruitage

Les méthodes classiques de débruitage supposent que l'image bruitée est de la forme

$$\tilde{U} = AU + N$$

où \tilde{U} désigne l'image bruitée, U la vraie image, N le bruit inconnu mais souvent supposé comme suivant une loi de Poisson.

Si le nombre de photons est suffisant (ce qui correspond donc à un long temps d'exposition), on peut utiliser le résultat d'approximation d'une loi de Poisson par une loi normale et donc supposer alors le bruit comme étant gaussien.

Ce type de modèles classiques donnent lieu à des méthodes de débruitage classique comme le filtre de Wiener ou la regularisation de Tykhonov par exemple.

Dans le domaine de l'imagerie avec peu de lumières d'autres approches ont déjà été testées comme certaines utilisant du sparse coding dans [Mairal and Zisserman, 2009] ou 3D transform-domain filtering (BM3D) dans [Dabov and Egiazarian, 2007].

De manière générale, c'est la méthode BM3D qui présentent les meilleurs résultats.

D'autres approches utilisant du deep learning ont également été proposées avec l'utilisation d'auto encoders, multi-layer perceptrons ou encore de convolutional networks.

2.2 Pipeline du traitement des images

En tant que pipeline déjà existant pour la chaîne de développement d'image au-delà de la chaîne classique, il en existe d'autres qui présentent de bons résultats comme le burst imaging pipeline présenté dans [H. Jiang and Wandell, 2017] ou le L3 pipeline présenté dans [S.W.Hasinoff and R.Geiss, 2016].

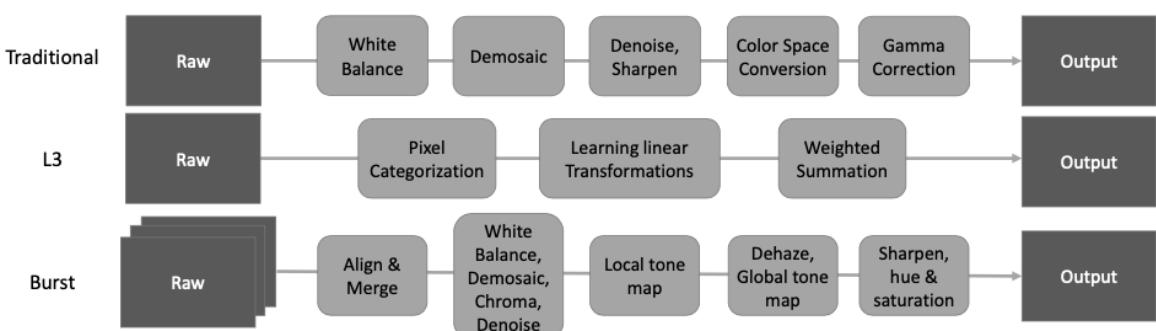


FIGURE 1 – Architecture of traditional, L3 and burst pipeline

3 Nouveau pipeline

Dans [Chen chen and Koltun, 2018], l'idée est restaurer des images prises dans des conditions de très basse luminosité à l'aide d'un réseau fully-convolutional network à l'aide de deux images, une avec un court temps d'exposition en entrée du réseau et une image avec un long temps d'exposition servant d'étiquettes, de groundtruth en sortie du réseau.

Le réseau réalise toute la chaîne de développement de l'image et ne traite pas seulement l'image RGB après développement du pipeline traditionnel.

L'architecture du pipeline [2] est comme suit

- En entrée, le réseau reçoit l'image RAW donc sa trame de Bayer de taille $H \times W \times 1$ qu'il va transformer en vraie tenseur de taille $\frac{H}{2} \times \frac{W}{2} \times 4$ ce qui revient à décomposer l'image en 4 canaux en fonction des couleurs 2 verts, 1 rouge et 1 bleu
- On soustrait le niveau de noir du capteur de l'appareil au tenseur obtenu précédemment
- On multiplie maintenant par le ration d'amplification δ permettant de régler le contraste
- On utilise ensuite un réseau *ConvNet* [3] qui ressort un tenseur de format $\frac{H}{2} \times \frac{W}{2} \times 12$
- Ce tenseur est ensuite converti en tenseur de format $H \times W \times 3$ pour chacune des couleurs et on a enfin une image RGB.

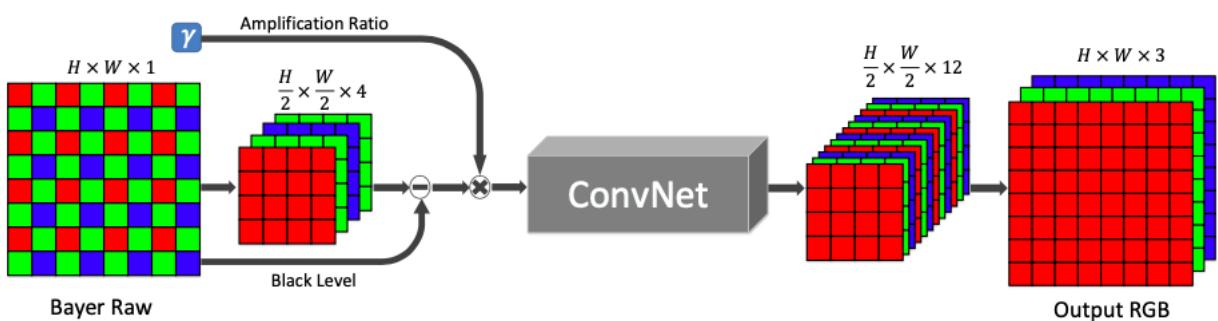


FIGURE 2 – Architecture of the new pipeline

Le réseau *ConvNet* a une architecture en U comme décrit dans [O. Ronneberger and Brox, 2015]. La première partie du réseau est un réseau contractant par couches successives où les opérateurs de pooling sont remplacés par des opérateurs de upsampling permettant de distinguer les "high resolution features". L'autre partie du réseau, le réseau dilatant permet grâce à un up-sampling de localiser les features.

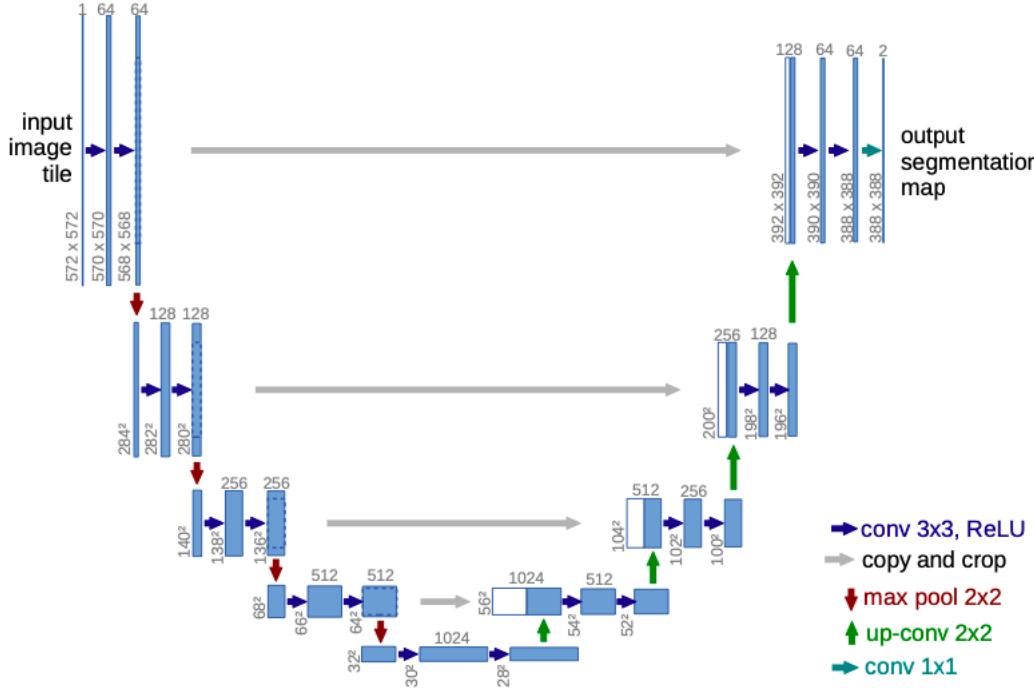


FIGURE 3 – Architecture of the new pipeline

4 Données

La plupart des méthodes de traitement des images en condition de basse luminosité utilisait des jeux de données synthétiques ou alors des images en basse luminosité mais sans images groundtruth donc ceci ne permettait pas de comparer les résultats et cela devenait de l'apprentissage non supervisé.

Comme mentionné dans [Chen chen and Koltun, 2018], il n'existe pas de jeu de données d'images en condition de basse luminosité avec images groundtruth, les auteurs de l'article en ont donc créé un.

Le dataset SID contient 5094 images RAW à temps de pose court (1/30 à 1/10s) avec une référence à temps de pose long (10s à 30s). Les appareils photos utilisés sont un Sony 7S II avec une trame de Bayer et un Fujifilm X-T2 avec une trame X-Trans. Les photos ont été prises avec un trépied afin de capturer des scènes fixes évitant le flou que peut faire intervenir un temps d'exposition relativement long.

5 Expériences

Pour tester la méthode décrite dans [[Chen chen and Koltun, 2018](#)], on a utilisé le code du [github](#) associé à l'article.

Afin de faire des expériences, nous avons entraîné le réseau de l'article par les données d'entraînement du jeu de données SID (See-In-the-Dark).

Cet entraînement se fera sur les ordinateurs de Télécom afin de faire les calculs sur GPU, car sur un ordinateur personnel, le temps d'exécution de l'algorithme est trop grand sans compter la taille du jeu de données qui est massif.

Voici les résultats que l'on peut obtenir en utilisant le modèle sur les données SID.



FIGURE 4 – Photo avec un court temps d'exposition



FIGURE 5 – Photo groundtruth



FIGURE 6 – Photo obtenue à la sortie du réseau, SSIM : 0.885, PSNR : 32.9

Nous allons par la suite tester la capacité du réseau à générer une image proche de la groundtruth sur nos données avec nos photos prises dans des conditions de faible luminosité tout en changeant certains paramètres tels que le temps d'ouverture du capteur entre la photo pour un court temps d'exposition et la photo pour un long temps d'exposition et la caméra utilisée afin de tester la généralisation de la méthode pour des trames de Bayer différentes.

Voici, les photos prises par :

- un iPhone 12 (on a pu capturé l'image RAW grâce une application tierce)
- Sensibilité : ISO-320
- Ouverture : $f/1.8$
- Temps d'exposition : 1 s pour le groundtruth et 1/100 pour la photo avec un court temps d'exposition (afin de garder le ratio x100)
- Biais : +0.6 stp
- Distance focale 4 mm

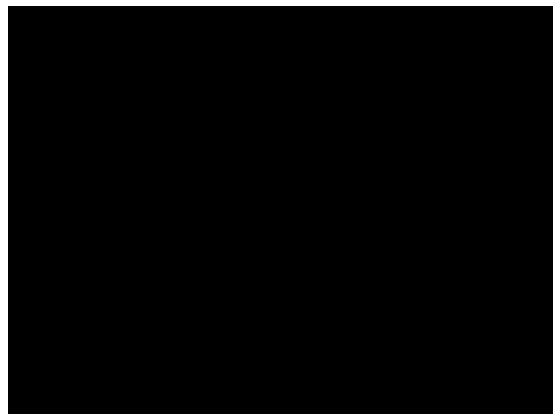


FIGURE 7 – Photo avec un court temps d'exposition



FIGURE 8 – Photo groundtruth



FIGURE 9 – Photo obtenue à la sortie du réseau, SSIM : 0.533, PSNR : 20.9

On constate donc que la méthode réussit à rendre visible et à éclaircir l'image en plus de la débruiter de manière surprenante. Toutefois, en utilisant d'autres métriques telles que SSIM ou PSNR nous remarquons que les résultats sont moins bons. Cela doit être dû au fait que nous n'avions pas exactement les mêmes paramètres que ceux utilisés lors de l'entraînement tels que la trame de Bayer de l'iPhone, les temps d'exposition, la sensibilité ISO et l'illuminance qui diffèrent.

Toutefois, même si les photos précédentes ont été prises par un iPhone qui présentent une trame de Bayer différentes de celle de l'appareil photo Sony utilisé dans [Chen chen and Koltun, 2018] la génération est possible et cela montrant ainsi que le réseau est robuste et s'adapte à différente trame de Bayer.

6 Conclusion

L'article Learning in The Dark propose donc une méthode novatrice permettant de traiter les photos prises dans des conditions de très basse luminosité via un réseau entièrement connecté convolutionnel réalisant l'entièrement de la chaîne de développement d'images à partir de l'image RAW. Cette méthode a priori au vu de ce que l'on a expérimenté fonctionne avec des images issues d'appareils différents et donc se généralise relativement bien. Cependant, expérimentalement, nous nous sommes rendus compte que le réseau présente toutefois quelques défauts qu'il pourrait être intéressant d'étudier ou peut-être de corriger avec plus de temps comme par exemple, en changeant la fonction de perte et utilisant par exemple la perte PSNR, car à la base l'objectif est d'augmenter le rapport signal sur bruit, ou encore la perte SSIM qui présente une réalité visuelle plus instinctive que la loss L1.

Références

- [Chen chen and Koltun, 2018] Chen chen, Qifeng Chen, J. X. and Koltun, V. (2018). Learning to see in the dark. *Conference on Computer Vision and Pattern Recognition (CVPR 2018)*.
- [Dabov and Egiazarian, 2007] Dabov, K. and Egiazarian (2007). Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on Image Processing*.
- [H. Jiang and Wandell, 2017] H. Jiang, Q. T. and Wandell, B. A. (2017). Learning the image processing pipeline. *IEEE Transactions on Image Processing*.
- [Mairal and Zisserman, 2009] Mairal, B. and Zisserman (2009). Non-local sparse models for image restoration. *ICCV*.
- [O. Ronneberger and Brox, 2015] O. Ronneberger, P. F. and Brox, T. (2015). U-net : Convolutional networks for biomedical image segmentation. *MICCAI*.
- [S.W.Hasinoff and R.Geiss, 2016] S.W.Hasinoff, D. and R.Geiss (2016). Burst photography for high dynamic range and low-light imaging on mobile cameras. *ACM Transactions on Graphics*.