Estimation de profondeur mono-image par réseaux de neurones et flou de défocalisation

Marcela Carvalho[1] Bertrand Le Saux[1] Pauline Trouvé[1] Andrés Almansa[2] Frédéric Champagnat[1]

[1] DTIS, ONERA, Université Paris-Saclay, FR-91123 Palaiseau, France
[2] Université Paris Descartes, FR-75006 Paris, France

{marcela.carvalho, bertrand.le_saux, pauline.trouve}@onera.fr

Résumé

L'estimation de la profondeur monoculaire à l'aide de réseaux de neurones profonds a atteint aujourd'hui d'excellentes performances. Cependant, il est difficile d'établir l'influence respective de l'architecture, de la fonction de coût et des conditions d'expérimentations sur ces résultats. Dans cet article, nous présentons une nouvelle architecture, appelée D3-Net, pour l'estimation de profondeur monoculaire. Cette architecture, simple à entraîner et ne reposant pas sur des modèles analytiques de la scène nous permet d'étudier l'influence de différentes fonctions de coûts (standards et proposées dans l'état de l'art) et différentes conditions expérimentales sur les performances d'estimation de profondeur. Cette étude nous a amené à choisir une fonction de coût correspondant à la norme $\mathcal{L}_1$, à laquelle on ajoute une fonction de coût adversaire lors qu'un grand nombre de données est disponible. Notre méthode atteint alors les performances de l'état de l'art sur la base NYUv2. De plus les approches d'estimation de profondeur par apprentissage exploitent uniquement les structures géométriques des scènes et ne prennent pas en compte un indice depuis longtemps utilisé pour l'estimation de profondeur : le flou de défocalisation. Nous présentons ici une analyse sur données simulées qui montre le gain en performance lorsque la base contient des images avec du flou de défocalisation. Nous étudions également l'influence du flou dans la prédiction de profondeur en observant l'incertitude du modèle avec une approche de réseau de neurones bayésienne.

Mots Clef

Estimation de la profondeur, apprentissage profond, profondeur par le flou de défocalisation, DFD, régression.

Abstract

Monocular depth estimation using deep neural networks has achieved excellent performances nowadays. However, it is hard to establish respective influence of the architecture, the loss function and the experimental conditions on the performance. In this paper, we propose a new architecture, called D3-Net, for monocular depth estimation. This architecture, simple to train and not based on analytical models of the scene allows us to study the influence of different loss functions (standard and proposed in the state of the art) and different experimental conditions on depth estimation performance. This study led us to choose a loss function corresponding to the standard $\mathcal{L}_1$, to which we add an adversarial loss function when a large amount of data is available. Our method reaches state-of-the-art performance on the NYUv2 basis. However, the depth estimation approaches using deep learning only exploit the geometrical structures of the scenes and do not take into account a long-used cue for depth estimation : defocus blur. Here we present a simulated data analysis that shows the gain in performance when the database contains images with defocus blur. We also investigate the influence of blur in depth prediction by observing model uncertainty with a Bayesian neural network approach.

Keywords

Depth estimation, deep learning, depth from defocus, DFD, regression.

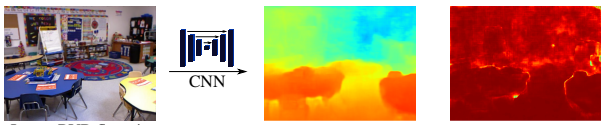


FIGURE 1 – Illustration de l'estimation de la profondeur monoculaire et de l'incertitude du modèle avec l'ensemble de données NYUv2 [24] synthétiquement floutés et un réseau de neurones convolutif (CNN).

1 Introduction

L'estimation de profondeur est un problème majeur dans la vision par ordinateur notamment pour des applications concernant l'interaction homme-machine, la réalité augmentée et la robotique. Les capteurs 3D traditionnels exploitent typiquement la vision stéréoscopique, le mouvement ou la projection d'une lumière structurée. Cependant, ces capteurs dépendent de l'environnement (soleil, texture) ou nécessitent plusieurs périphériques (caméra, pro-

jecteur), ce qui conduit à des systèmes très encombrants. De nombreux efforts ont été faits pour construire des systèmes compacts : les plus remarquables sont peut-être les caméras *light field* qui utilisent une matrice de microlentilles devant le capteur.

Récemment, plusieurs approches d'estimation de la profondeur basées sur l'apprentissage profond ont été proposées, commençant par [5]. Ces méthodes utilisent un seul point de vue (une seule image) et optimisent généralement une régression sur la carte de profondeur de référence. Le premier défi concerne l'architecture réseau, qui suit habituellement les avances proposées chaque année dans le domaine de l'apprentissage profond : VGG16 [29, 4, 20], réseaux résiduels (ResNet) [10, 17, 33, 21]. Le deuxième défi est la définition d'une fonction de perte appropriée pour la régression en profondeur. Ainsi, la relation entre les réseaux et les fonctions objectives est complexe et leurs influences respectives sont difficiles à distinguer.

Les méthodes précédentes exploitent les aspects géométriques de la scène uniquement pour en déduire la profondeur. Or un autre indice connu pour l'estimation de profondeur est le flou de défocalisation [25, 31, 23, 28, 34]. Cependant, l'estimation de profondeur à l'aide du flou de défocalisation (*Depth from Defocus*, DFD) avec une caméra conventionnelle et une seule image souffre d'une ambiguïté dans l'estimation de profondeur par rapport au plan focal et à la zone aveugle liée à la profondeur de champ de la caméra, où aucun flou ne peut être mesuré. De plus, pour estimer la profondeur d'une scène floue inconnue, le DFD nécessite un modèle de scène et un calibrage de flou pour le relier à une valeur de profondeur.

Dans cet article, nous proposons d'une part une nouvelle architecture de réseau simple à entraîner de bout en bout et dédiée à l'estimation de profondeur. Nous étudions comment des choix particuliers de fonctions de perte et de conditions expérimentales affectent les performances de la prédiction de profondeur. Cette analyse nous a amené à choisir une fonction de coût qui permet à notre approche d'atteindre l'état de l'art en estimation de profondeur monoculaire. D'autre part, nous proposons d'intégrer le flou de la défocalisation à la puissance des réseaux neuronaux. Nous montrons que l'association d'images défocalisées avec un réseau de neurones permet à la fois de dépasser les performances obtenues avec des images nettes mais également d'éviter les limitations classiques du DFD, sans nécessiter un modèle analytique de scène. Enfin, pour mieux comprendre l'influence du flou sur les performances du réseau, nous étudions également l'influence du flou dans la prédiction de profondeur en observant l'incertitude du modèle avec une approche de réseau de neurones bayésienne.

2 État de l'art

2.1 Apprentissage profond de la profondeur

La publication d'un jeu de données RVB-D à grande échelle, NYUv2 [24], a rendu possible l'estimation de profondeur par des réseaux de neurones convolutifs profonds (*Deep Convolutional Neural Networks*, DCNN). Le premier réseau convolutif, proposé par Eigen *et al.* [5], est une architecture multi-échelle associée à une fonction de coût invariante à l'échelle (voir $\mathcal{L}_{eigen}$ dans le Tableau 1) qui encourage les pixels voisins à avoir des valeurs de profondeur similaires. [4] a étendu ce travail en ajoutant des gradients de première ordre dans leur fonction de perte ($L_{eigengrad}$) afin d'imposer une structure locale sur la carte de profondeur. D'autres travaux ultérieurs basent l'entraînement sur la régression par pixel et utilisent les fonctions de perte standard en régression, comme l'erreur absolue moyenne ($\mathcal{L}_1$) et l'erreur quadratique moyenne ($\mathcal{L}_2$) [17, 33, 21]. Les contributions de ces travaux résident alors dans les architectures de réseau et l'utilisation des champs de conditions aléatoires (CRF). Laina *et al.* [17] proposent une brève comparaison de résultats entre la norme $\mathcal{L}_2$ et la norme $\mathcal{L}_{berhu}$. Ce travail a été étendu dans [21] avec l'adoption d'une $\mathcal{L}_1$.

Une nouvelle méthode de régression a été introduite récemment par Goodfellow *et al.* [7] pour produire des images réalistes à partir de vecteurs de bruit : les Réseaux Génératifs Adversaires (*Generative Adversarial Networks*, GAN). Ce travail a été étendu dans [12] pour conditionner les sorties générées à une image d'entrée (*conditional-GAN*, cGAN). Les GANs fonctionnent en définissant une pénalité adversaire modélisée par un réseau qui classifie la vraisemblance de la sortie. Jung *et al.* [14] a adopté avec succès cette idée pour effectuer une prédiction de profondeur avec une stratégie d'entraînement en deux phases : le réseau est d'abord entraîné avec une $\mathcal{L}_1$ et est ensuite affiné avec une perte adversaire.

Enfin, Kendall et Gal [16] ont proposé un réseau bayésien basé sur [15, 11, 13] combiné à une nouvelle fonction de régression qui capture l'incertitude des données et du modèle pour améliorer l'apprentissage.

Tous les travaux mentionnés ici utilisent les derniers réseaux à la pointe de l'état de l'art pour améliorer les performances d'estimation de profondeur, tout en adoptant différentes fonctions de coût. Cependant, l'influence respective de l'architecture et de la fonction de coût n'est pas facile à appréhender à partir de ces travaux car aucun d'entre eux n'a effectué une comparaison complète entre toutes ces fonctions de coût. Dans cet article, nous effectuons une comparaison des différentes fonctions de pertes de la littérature (standards et personnalisées) pour différentes conditions expérimentales, avec une même architecture, appelée D3-Net [3]. De plus, nous apportons un nouvel éclairage sur l'utilisation de la perte adversaire qui nécessite une grande quantité de données pour être efficace. Le réseau, combinant la meilleure fonction de coût suivant notre étude et l'architecture D3-Net se rapproche alors des meilleures performances sur NYUv2, est aussi beaucoup plus simple à entraîner en par rapport à [33, 14] car il peut être entraîné de bout-à-bout et n'a pas besoin de modèle analytique.

Les méthodes précédentes utilisent uniquement les informations géométriques de la scène, autrement dit des images nettes à Grande Profondeur de Champ (GPC). Or un indice sur la profondeur est contenu dans les images à Faible Profondeur de Champ (FPC) et peut être exploitée par traitement pour estimer la profondeur.

2.2 *Depth from Defocus* (DFD)

Dans le domaine de la *computational photography*, plusieurs travaux ont étudié l'utilisation du flou de défocalisation pour déduire la profondeur, commençant par [25]. En effet, comme illustré dans la Figure 2, la quantité de flou d'un objet peut être liée à sa profondeur en utilisant l'optique géométrique :

$$\epsilon = Ds \cdot \left| \frac{1}{f} - \frac{1}{d_{out}} - \frac{1}{s} \right|, \tag{1}$$

où f représente la distance focale, D, le diamètre de l'objectif, d_{out} la distance de l'objet par rapport à l'objectif et s la distance entre le capteur et l'objectif.
Les travaux les plus récents utilisent une seule image (*Single-Image* DFD, SIDFD). L'enjeu est alors d'estimer le flou à partir d'une image d'une scène inconnue. Plusieurs approches utilisent des modèles analytiques pour la scène tels que les modèles à bords de plages [34] ou les modèles de scène statistique avec une distribution gaussienne [18, 31]. Cependant, ces approches sont limitées à des scènes compatibles avec le modèle proposé.
Par ailleurs, le SIDFD souffre de deux limitations : premièrement, il y a une **ambiguïté** dans l'estimation de la profondeur si l'objet est derrière ou devant le plan de focalisation ; deuxièmement, dans la profondeur de champ de la caméra, aucune variation de flou ne peut être mesurée, ce qui conduit à une **zone aveugle**. L'ambiguïté peut être résolue en utilisant un diaphragme codé asymétrique [28], ou en réglant la distance focale à l'infini, mais au prix d'une réduction de l'intensité lumineuse atteignant le capteur ou d'une grande profondeur de champ (zone aveugle).
Enfin, inférer la profondeur à partir d'une quantité de flou de défocalisation nécessite une étape d'étalonnage. Une exception est [23], où une projection de données sur des sous-espaces orthogonaux est obtenue à l'aide d'un jeu d'images avec du flou et formées pour chaque profondeur potentielle.

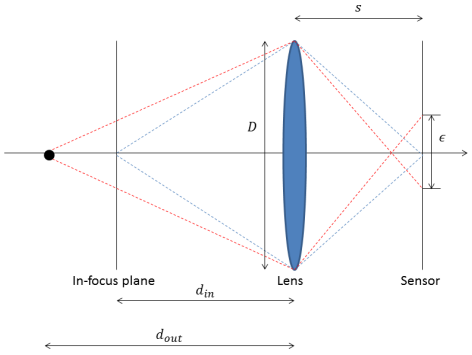


FIGURE 2 – Illustration du principe du DFD. Les rayons provenant du point flouté (point noir) convergent avant le capteur et s'étalent sur un disque de diamètre ϵ.

2.3 L'apprentissage de la profondeur par le DFD

A notre connaissance, seuls quelques articles de la littérature utilisent le flou de la défocalisation comme indice pour apprendre la profondeur à partir d'une seule image. Srinivasan *et al.* [30] utilisent le flou de défocalisation seulement au moment de l'entraînement d'un réseau dédié à l'estimation de profondeur monoculaire à partir d'images GPC. Hazirbas *et al.* [9] utilisent un jeu d'images FPC avec plusieurs mises au point différentes, ce qui est plutôt lié aux approches de *depth from focus* qu'au DFD. Enfin, [1] présente un réseau pour l'estimation de la profondeur et la déconvolution de l'image en utilisant une seule image FPC. Il compare les performances des architectures de la littérature entraînées sur des images GPC avec son architecture entraînée sur des images FPC. Cependant, l'influence du flou de défocalisation seul sur les performances n'est donc pas étudiée puisque les architectures comparées sont différentes. De plus, le flou généré dans [1] n'a pas d'interprétation physique en terme de paramètres capteurs.
Dans cet article, nous présentons une étude de l'influence du flou sur la performance de l'estimation de profondeur par apprentissage profond. Pour cela nous menons :

- des comparaisons de performance de D3-Net sur des images GPC et FPC simulées (générées à partir d'un ensemble de cartes de profondeur réelles et d'images GPC) ;
- des comparaisons de performances entre plusieurs réglages de caméra (variation de la mise au point) ;
- des analyses de l'influence du flou de défocalisation sur le réseau neuronal à l'aide de cartes d'incertitudes et diagramme d'erreurs par profondeur.

3 Estimation de la profondeur monoculaire avec des images GPC

Dans cette section, nous présentons d'abord l'architecture du réseau D3-Net [3], puis nous présentons les comparaisons de performances obtenues avec différentes fonctions de coût pour différentes conditions d'expérimentation. Pour comparer les performances, nous utilisons les mesures d'erreur standard proposées dans [5, 19] et une base de données standard, NYUv2 [24] qui contient environ 230k paires d'images d'intérieur provenant de 249 scènes pour l'entraînement et de 215 scènes pour le test. NYUv2 contient également un ensemble de données plus petit avec 1449 paires d'images RVB et de profondeur alignées, dont 795 paires sont utilisées pour l'entraînement et 654 paires pour les tests. Notons que pour effectuer des comparaisons justes, nous effectuons soigneusement tous les processus d'entraînement en gardant les paramètres du réseau sans aucun changement.

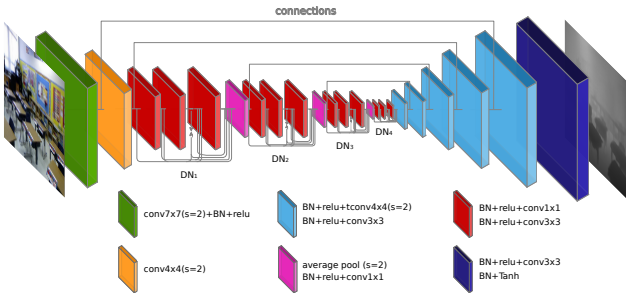


FIGURE 3 – Architecture de D3-Net. La partie codeur correspond à une version modifiée de DenseNet-121, où nous avons remplacé le max-pooling par une convolution 4x4 avec *stride*=2 (bloque jaune).

3.1 D3-Net

La Figure 3 présente l'architecture présentée par [3] : elle est basée sur un réseau DenseNet-121 [11] pour la partie encodeur, où nous avons remplacé un max-pooling par une convolution 4x4 avec *stride*=2 (bloc jaune). Ici, les blocs denses, DBx, $x \in [1, 2, 3, 4]$, contiennent respectivement 6, 12, 24 et 16 ensembles de convolutions 1x1 et 3x3 comme indiqué dans la Figure 3. Le décodeur comprend des blocs de convolutions transposées 4x4 avec *stride*=2 et 3x3 avec *stride*=1 pour suréchantillonner des cartes de caractéristiques à une résolution plus élevée. Les parties de l'encodeur et du décodeur sont connectées via des *skip-connections* comme proposé par [26], pour améliorer le flou d'information contextuel. Contrairement aux architectures précédentes [33, 14], notre réseau peut être entraîné en une seule phase et ne nécessite aucun modèle analytique supplémentaire comme les CRF [32, 33]

3.2 Fonctions de coût

Les fonctions de coûts considérées dans cet article sont présentées dans le tableau 1. Notons que concernant la fonction de perte $\mathcal{L}_{gan}$, nous modifions le *conditionnal patch GAN* proposé dans [12] pour l'adapter à la tâche d'estimation de profondeur. Le réseau discriminateur est conçu pour mesurer et classifier si une carte de profondeur d'entrée est vraie ou fausse. Les vraies cartes correspondent aux profondeurs de la vérité terrain et les fausses cartes correspondent aux profondeurs générées par D3-Net. Ce réseau est entraîné pour remplacer les fonctions de perte nécessitant formulation analytique. Cependant, pour lisser les prédictions du GAN et guider son entraînement, nous ajoutons une $\mathcal{L}_1$ à la sortie de D3-Net. La structure de *patch* permet au discriminateur de pénaliser les prédictions par imagette au lieu de pénaliser l'image entière, ce qui conduit à des résultats avec des détails plus fins.

3.3 Comparaison de performances

La figure 4 montre l'évolution des performances du réseau avec différentes pertes lorsqu'il est entraîné avec différentes tailles de la base de données. Nous utilisons trois divisions différentes de NYUv2 : les 795 paires du sous-ensemble de la base déjà mentionné, 12k paires provenant d'échantillons équidistants de l'ensemble de données com-

Fonction de coût		Equation
Absolue moyenne	$\mathcal{L}_1$	$\frac{1}{N}\sum_i^N \lvert l_i \rvert$
Quadratique moyenne	$\mathcal{L}_2$	$\frac{1}{N}\sum_i^N (l_i)^2$
Invariante à l'échelle [5]	$\mathcal{L}_{eigen}$	$\frac{1}{N}\sum_i^N d_i^2 - \frac{\lambda}{N^2}(\sum_i^N d_i)^2$
Invariante à l'échelle avec gradients [4]	$\mathcal{L}_{eigengrad}$	$\frac{1}{N}\sum_i^N d_i^2 - \frac{\lambda}{2N^2}(\sum_i^N d_i)^2 + \frac{1}{N}\sum_i^N [(\nabla_x di)^2 + (\nabla_y di)^2]$
BerHu [17]	$\mathcal{L}_{berhu}$	$\begin{cases} \mathcal{L}_1(l_i) & \mathcal{L}_1(l_i) \leqslant c, \\ \frac{\mathcal{L}_2(l_i)+c^2}{2c} & else. \end{cases}$
Huber [17]	$\mathcal{L}_{huber}$	$\begin{cases} \mathcal{L}_1(l_i) & \mathcal{L}_1(l_i) \geqslant c, \\ \frac{\mathcal{L}_2(l_i)+c^2}{2c} & else. \end{cases}$
Least Squared Adversarial [22, 12]	$\mathcal{L}_{gan}$	$\frac{1}{2}\mathbb{E}_{x,y\sim p_{data}(x,y)}[(D(x,y) - 1)^2] + \frac{1}{2}\mathbb{E}_{x\sim p_{data}(x)}[(D(x,G(x)) - C)^2] + \lambda\mathcal{L}_{L1}(G(x)))$

TABLE 1 – Liste des fonctions de coût pour la régression. Soit y_i et $\hat{y}_i$ la vérité terrain et la distance estimée en mètres, $l_i = y_i - \hat{y}_i$, $d_i = log(y_i) - log(\hat{y}_i)$, G, le réseau du générateur, D, le réseau discriminateur et x, l'entrée RVB.

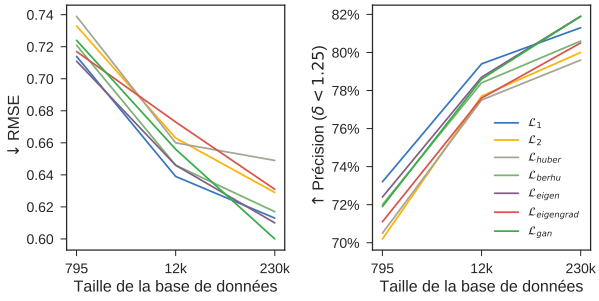
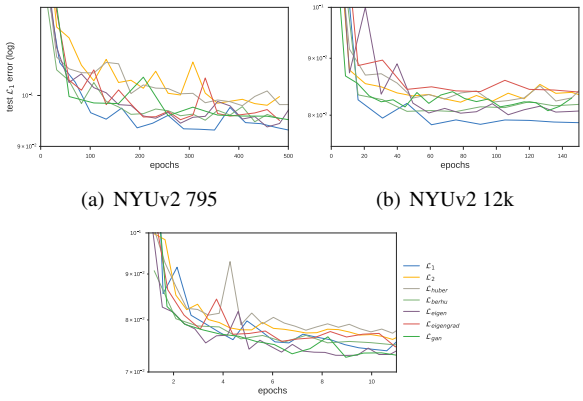


FIGURE 4 – Évolution des performances de D3-Net pour différentes tailles de la base d'entraînement et différentes fonctions de coût.



(a) NYUv2 795 (b) NYUv2 12k

(c) NYUv2 230k

FIGURE 5 – Comparaison de la vitesse de convergence pour différentes fonctions de coût et divisions de NYUv2.

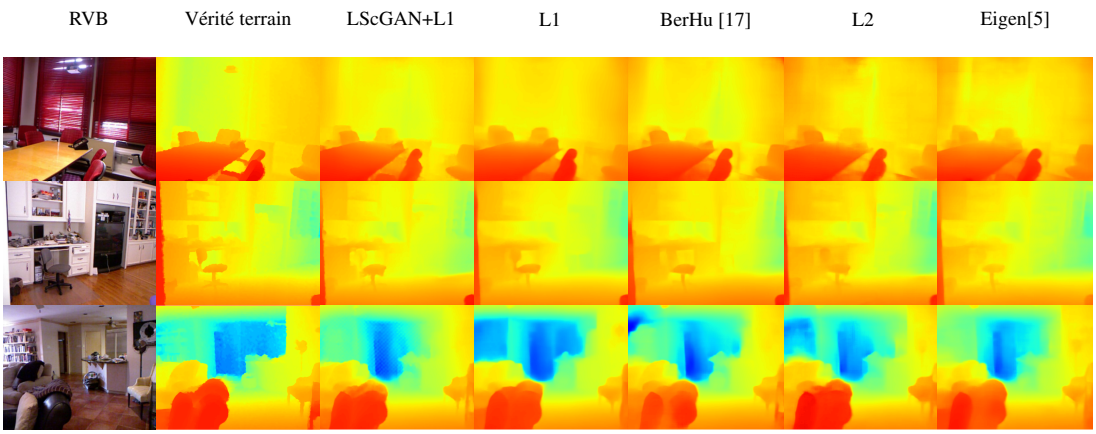


FIGURE 6 – Comparaison qualitative de D3-Net entraîné avec différentes pertes de régression de la littérature de profondeur à partir d'images monoculaires. Les valeurs plus bas de la profondeur et des incertitudes sont représentées par des couleurs plus chaudes.

plet et la base complète avec 230k paires d'images.
Comme on peut s'y attendre, l'augmentation de la base de donnée conduit à de meilleurs résultats dans tous les cas. Cependant, les fonctions de coût évoluent différemment d'une taille de base de données à l'autre. Globalement, $\mathcal{L}_1$ et $\mathcal{L}_{eigen}$ présentent les meilleures performances pour différentes tailles de l'ensemble de données. D'autre part, $\mathcal{L}_{gan}$ devient très efficace lorsque le réseau est entraîné avec une grande quantité de données. Les GANs ont une instabilité bien connue (*mode-collapse* [22]) qui, dans notre cas, peut être contournée avec plus de données.
La Figure 5 compare les vitesses de convergence, $\mathcal{L}_1$ et $\mathcal{L}_{eigen}$ semblent aussi converger plus efficacement que les autres fonctions de coût et obtiennent de meilleures prédictions plus rapidement. Cela reste vrai pour les deux plus petites taille de base de données, mais lors de l'apprentissage du modèle avec 230k, le modèle GAN et $\mathcal{L}_{eigen}$ surpassent les autres fonctions de coût. Nos meilleurs résultats avec D3-Net sont alors ceux obtenus avec la base de données complète et la fonction $\mathcal{L}_{gan}$. Comme nous pouvons observer dans le Tableau 2, nos résultats font partie des meilleurs de l'état de l'art.
La Figure 6 compare les cartes de profondeur obtenues pour différentes fonction de coûts en entraînant le réseau sur l'ensemble de données complète NYUv2. Nous pouvons d'abord remarquer que $\mathcal{L}_2$ a tendance à lisser les prédictions. En effet, sur les petites différences, $\mathcal{L}_2$ a une tendance à étaler les erreurs, ce qui favorise des résultats lisses. En revanche, $\mathcal{L}_1$ pénalise plus les faibles erreurs et favorise des résultats plus contrastés. $\mathcal{L}_{berhu}$ propose de tirer parti de $\mathcal{L}_1$ pour de très petites erreurs et d'utiliser $\mathcal{L}_2$ pour le cas complémentaire et tandis que $\mathcal{L}_{huber}$ tire parti de $\mathcal{L}_2$ pour de très petites erreurs et utilise $\mathcal{L}_1$ pour le cas complémentaire. Dans les deux cas, les résultats restent lisses : la présence d'un terme en $\mathcal{L}_2$ dégrade les estimations. En comparaison, $\mathcal{L}_{gan}$, $\mathcal{L}_{eigen}$ et $\mathcal{L}_1$ présentent de belles prédictions visuelles confirmant les résultats quantitatifs précédents. L'approche patch-GAN peut conduire le modèle à capturer des détails à haute fréquence (contours, petits objets). Ces caractéristiques peuvent être clairement observées par exemple dans la première rangée, où les contours des différentes chaises dans le dos sont bien prédits avec $\mathcal{L}_{gan}$ et $\mathcal{L}_1$ par rapport à $\mathcal{L}_{berhu}$ et $\mathcal{L}_{huber}$, par exemple, qui les ignorent presque.
Dans cette section, nous avons montré que sur de petites bases de données, $\mathcal{L}_1$ et $\mathcal{L}_{eigen}$ produisent les meilleures performances et lorsque la taille de l'ensemble de données augmente, la performance bénéficie de l'utilisation d'une fonction de coût adversaires.

4 Apport du flou de défocalisation

Nous portons à présent notre intérêt sur l'apport du flou de défocalisation sur l'apprentissage de la prédiction de profondeur. Nous proposons ici une comparaison de performances de D3-Net sur des bases d'images GPC ou FPC, avec différents réglages de la caméra. Puis nous menons une étude d'incertitude du modèle à l'aide de réseaux de neurones Bayésiens. Notons que nous avons choisi d'utiliser la base NYUv2-795 pour de raisons de vitesse d'entraînement, D3-Net est donc utilisé avec $\mathcal{L}_1$ en tant que fonction de coût.

4.1 Simulation d'images FPC

Les images FPC sont simulées à partir de la carte de profondeur obtenue par une Kinect et de l'image GPC à l'aide de l'approche par couches successives de [8] où chaque image défocalisée est la somme d'images floues multipliées par des masques prenant en compte la profondeur de l'objet local et l'occlusion des objets de premier plan.
Comme [30], nous avons choisi de modéliser le flou comme une fonction de disque dont le diamètre varie avec la profondeur selon l'équation 1. Pour générer des images floues physiquement réalistes, nous choisissons des paramètres correspondant à une caméra synthétique avec une focale de 15mm, une ouverture de 2.8 et une taille de pixel de 5.6μm. Deux réglages de plan focal sont testés, respectivement à 2m et 8m de la caméra. La Figure 8 montre un exemple d'image FPC synthétique et la Figure 7 montre la

variation du diamètre de flou ϵ par rapport à la profondeur, pour les deux réglages. Comme illustré sur la Figure 7, régler le plan de mise au point à 2m correspond à une caméra avec une faible profondeur de champ. Il existe une ambiguité de part et d'autre du plan de mise au point. Régler le plan de mise au point à une profondeur plus grande, ici 8m, correspond à un réglage sans ambiguité sur le flou de défocalisation mais avec une grande profondeur de champ. Pour ce réglage, la comparaison des performances avec les méthodes SIDFD peut alors être effectuée. Nous avons choisi deux méthodes de la littérature SIDFD [34, 31] qui estiment la quantité de flou localement.

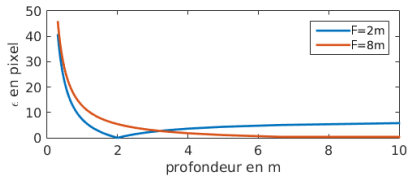


FIGURE 7 – Variation de diamètre du flou vs. profondeur pour les deux paramètres de mise au point à 2m et 8m.

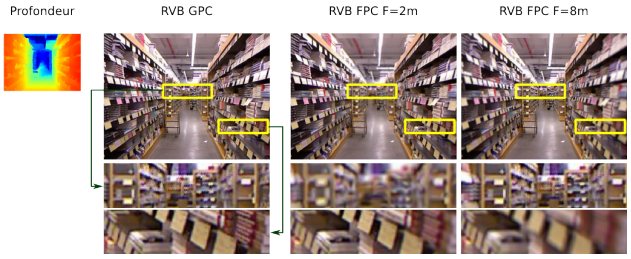


FIGURE 8 – Exemples d'images FPC synthétiquement générées à partir d'une image de la base de données NYUv2 pour deux réglages de plan focal de caméra : 2 et 8 m.

4.2 Comparaison de performance

Le Tableau 3 montre les résultats de performance de D3-Net en utilisant d'abord les images GPC, puis les images FPC avec les deux paramètres de mise au point. Ces résultats sont comparés aux performances de deux méthodes

Méthodes	Erreur				Précision		
	rel	log10	rms	rmslog	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Images RVB GPC							
Saxena [27]	0.349	-	1.214	-	44.7%	74.5%	89.7%
Eigen [4] (VGG16)	0.158	-	0.641	0.214	76.9%	95.0%	98.8%
Laina [17]	0.127	0.055	0.573	0.195	81.1%	95.3%	98.8%
Xu [33]	0.121	0.052	0.586	-	81.1%	95.4%	98.7%
Cao [2]	0.141	0.060	0.540	-	81.9%	96.5%	99.2%
D3-Net	0.135	0.059	0.600	0.199	81.9%	95.7%	98.7%
Jung[14]	0.134	-	0.527	-	82.2%	97.1%	99.3%
Kendall and Gal [16]	0.110	0.045	0.506	-	81.7%	95.9%	98.9%

TABLE 2 – Mesures de performance obtenues par des méthodes de l'état de l'art de l'estimation en profondeur avec des images GPC de l'ensemble de données NYUv2. Nos résultats sont en bleu et correspondent aux performances de D3-Net avec une fonction de coût adversaire. Résultats extraits des papiers originaux.

locales de mesure du flou de défocalisation pour le paramètre de mise au point de 8m [34, 31].

Méthodes	Erreur				Précision		
	rel	log10	rms	rmslog	$\delta < 1.25\delta < 1.25^2$		$\delta < 1.25^3$
Images RVB GPC - NYUv2 795							
D3-Net GPC	0.226	-	0.779	-	65.8%	89.2%	96.7%
Images RVB avec flou supplémentaire - NYUv2 795							
D3-Net F=2m	0.068	0.028	0.328	0.110	96.1%	99.0%	99.6%
D3-Net F=8m	0.060	-	0.403	-	95.2%	99.1%	99.9%
Zhuo [34] F=8m	0.273	-	1.088	-	51.7%	83.1%	95.1%
Trouvé [31] F=8m	0.429	0.289	1.856	0.956	39.2%	52.7%	61.5%
Anwar [1]	0.094	0.039	0.347	-	-	-	-

TABLE 3 – Comparaison des performances de D3-Net en utilisant des images GPC, des images FPC avec des mises au point à 2 et 8m, et deux approches SIDFD [34, 31] pour un réglage de mise au point à 8m.

Plusieurs conclusions peuvent être tirées du Tableau 3. Tout d'abord, il y a une amélioration significative de la performance de l'estimation de la profondeur lors de l'utilisation d'images floues au lieu d'images GPC. Deuxièmement, D3-Net surpasse les méthodes SIDFD classiques, sans nécessiter un modèle de scène analytique ni d'étalonnage de flou. En outre, il existe une sensibilité de la performance en fonction de la position du plan de mise au point. Le meilleur réglage pour ces tests est le plan de mise au point à 2m, ce qui correspond à une faible quantité de flou pour la plupart des objets mais avec une ambiguïté de flou. Cela montre que le réseau utilise effectivement les indices de flou et qu'il est capable de surmonter l'ambiguïté de la profondeur en utilisant des informations structurelles géométriques. Le Tableau 2, présente aussi les résultats de [1]. Cependant, une comparaison stricte avec nos résultats n'est pas possible car les paramètres de flou pour la génération de l'ensemble de données synthétiques par [1] ne sont sont pas les mêmes. Nous pouvons cependant conclure que D3-Net atteint des résultats comparables avec une approche d'estimation en profondeur performante.

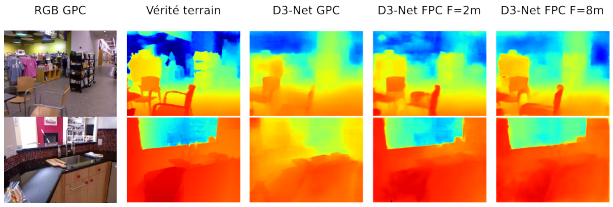


FIGURE 9 – Comparaison qualitative des prédictions avec les configurations de flou de défocalisation proposées.

Enfin, la Figure 9 et les colonnes 4 et 6 de la Figure 10 montrent des exemples de cartes de profondeur prédites. Les cartes de profondeur obtenues avec des images floues sont plus nettes qu'avec des images GPC. En effet, le flou de défocalisation fournit au réseau des informations de profondeur locales qui conduisent à une meilleure segmentation de la carte en profondeur.

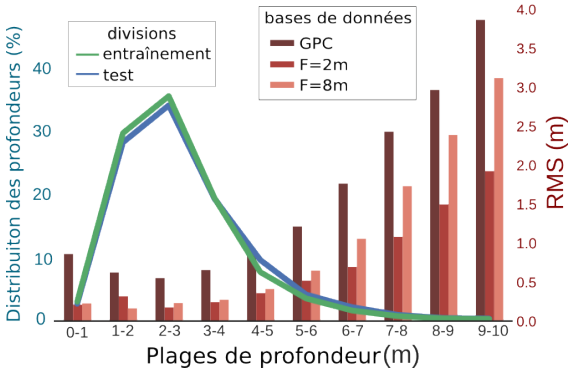


FIGURE 11 – Distribution des profondeurs pour des ensembles d'entraînement et test de NYUv2 et RMSE par plage de profondeur pour NYUv2 GPC, et FPC à 2 et 8m.

4.3 Incertitudes sur le modèle

Pour aller plus loin dans la compréhension de l'influence du flou dans la profondeur, nous évaluons l'incertitude épistémique du modèle de réseau profond suivant les travaux dans [15, 16, 6], afin d'observer la variation de la sortie du réseau en faisant varier les poids du réseau pour une image d'entrée.

Nous plaçons une distribution a priori sur les poids du réseau pour remplacer les paramètres déterministes du poids au moment du test [16]. Nous utilisons la méthode de *dropout* de Monte Carlo [6] pour mesurer l'inférence variationnelle en plaçant des couches de *dropout* pendant l'entraînement et aussi pendant la phase de test. En suivant [15], nous produisons 50 échantillons pour chaque image, ensuite nous calculons la prédiction moyenne et la variance de ces prédictions pour générer l'incertitude du modèle.

La Figure 10 présente des exemples de prédiction de réseau, d'erreur moyenne et d'incertitude épistémique pour l'ensemble de données NYUv2 avec des images GPC et FPC pour une mise au point à 2m. L'erreur moyenne est produite en utilisant l'image de la vérité terrain, alors que la variance ne dépend que de la distribution a priori du modèle. Pour les deux configurations, les variances les plus élevées sont observées dans les zones non texturées et dans les arêtes. Cependant, le modèle avec flou a une incertitude moins diffuse : elle est concentré sur les bords de l'objet, et ces objets sont mieux segmentés par rapport au modèle GPC. Dans la première ligne, nous observons des niveaux élevés d'incertitude sur les zones proches de la bibliothèque. Le modèle appris sur des images FPC réduit une partie de cette variance. Dans la deuxième ligne, nous observons que le modèle GPC a des difficultés à trouver un objet près de la fenêtre, ceci est surmonté par des flous présents sur le modèle en focus. Enfin, la dernière ligne présente un exemple concret où les deux modèles ont des variances de prédiction élevées principalement dans la partie moyenne supérieure, où il y a un trou. Cependant, le premier modèle présente également une erreur moyenne et une variance élevées dans la zone inférieure, contrairement au modèle avec flou.

4.4 Analyse d'erreur par profondeur

Nous étudions également l'erreur de prédiction par profondeur lors de l'utilisation d'images GPC et d'images FPC. La Figure11 montre la répartition de l'erreur par tranches de profondeur ainsi que la distribution de profondeur pour les images d'entraînement et de test de l'ensemble de données NYUv2.

Pour les images GPC, les erreurs semblent être corrélées au nombre d'exemples dans l'ensemble de données. En effet, une erreur minimale est obtenue pour 2m, correspondant à la profondeur avec approximativement le plus grand nombre d'exemples, selon la figure11.a. En utilisant le flou de défocalisation, la répartition des erreurs ressemble à l'augmentation quadratique de l'erreur avec la profondeur typique de l'estimation de la profondeur passive. En outre, le gain en précision est plus élevé pour les profondeurs proches car la variation du flou est plus importante à des profondeurs proches, comme l'illustre la Figure 7.

Enfin, le réglage de mise au point de 2m ne montre pas une augmentation d'erreur à 2m qui correspondrait à la zone aveugle de SIDFD ce qui montre que l'approche proposée surmonte également cette limitation classique du DFD.

5 Conclusion

Dans cet article, nous avons présenté une nouvelle méthode d'estimation de profondeur monoculaire qui atteint des très bonnes performances par rapport à l'état de l'art sans avoir besoin de modèles analytiques et avec une seule phase d'entraînement. Cette méthode est issue d'une comparaison de différentes fonctions de perte de régression qui a montré que sur de petites bases de données, les pertes $\mathcal{L}_1$ et $\mathcal{L}_{eigen}$ produisent les meilleures performances et lorsque la taille de l'ensemble de données augmente, la performance peut bénéficier de l'utilisation de pertes adversaires.

Nous avons également étudié l'influence du flou de défocalisation comme un indice dans une estimation de profondeur monoculaire en utilisant une approche d'apprentissage profond. Nos expériences montrent que l'utilisation d'images floues surpasse l'utilisation d'images complètement nettes, sans avoir besoin d'un modèle de la scène ni d'étalonnage de flou. En outre, l'utilisation combinée d'un flou de défocalisation et d'une information de structure géométrique sur l'image, apportée par l'utilisation d'un réseau profond, évite les limitations classiques du SIDFD avec une caméra conventionnelle, telle qu'une ambiguïté de profondeur ou une zone aveugle. Ces observations sont prometteuses et ouvrent la voie à d'autres études sur l'optimisation des paramètres de la caméra et des modalités d'acquisition pour l'estimation de la profondeur en utilisant le flou de défocalisation et l'apprentissage en profondeur, ce qui est une perspective de ce travail.

Références

[1] S. Anwar, Z. Hayder, and F. Porikli. Depth estimation and blur removal from a single out-of-focus image. In *BMVC*, 2017.

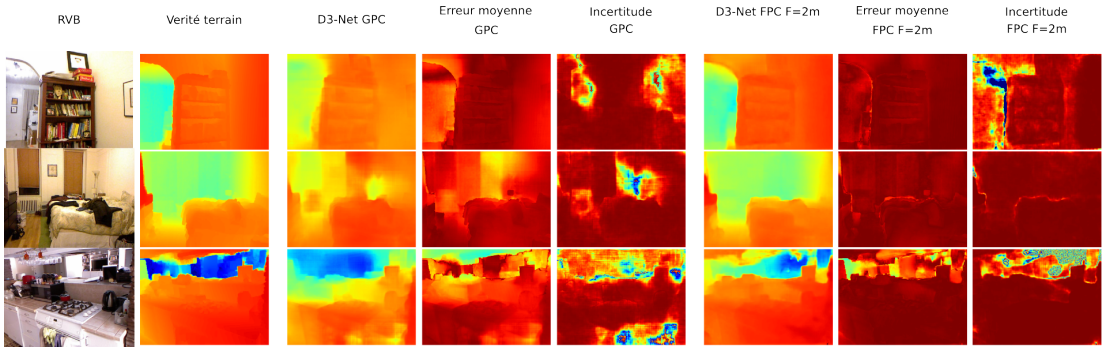


FIGURE 10 – Comparaison qualitative entre les images GPC et FPC avec focus à 2m, l'erreur moyenne et l'incertitude épistémique. Les valeurs ont été échelonnées entre 0 et 10 mètres pour les distances et entre 0-100 pour la variance.

[2] Y. Cao, Z. Wu, and C. Shen. Estimating depth from monocular images as classification using deep fully convolutional residual networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 2017.

[3] M. Carvalho, B. Le Saux, P. Trouvé-Peloux, A. Almansa, and F. Champagnat. On regression losses for deep depth estimation. *ICIP*, 2018.

[4] D. Eigen and R. Fergus. Predicting Depth, Surface Normals and Semantic Labels with a Common Multi-Scale Convolutional Architecture. *ICCV*, 2015.

[5] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. *NIPS*, 2014.

[6] Y. Gal and Z. Ghahramani. Dropout as a bayesian approximation : Representing model uncertainty in deep learning. In *ICML*, 2016.

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. *NIPS*, 2014.

[8] S. W. Hasinoff and K. N. Kutulakos. A layer-based restoration framework for variable-aperture photography. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8, Oct 2007.

[9] C. Hazirbas, L. Leal-Taixé, and D. Cremers. Deep depth from focus. In *Arxiv preprint arXiv :1704.01085*, April 2017.

[10] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[11] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *CVPR*, 2017.

[12] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *CVPR*, 2016.

[13] S. Jégou, M. Drozdzal, D. Vazquez, A. Romero, and Y. Bengio. The one hundred layers tiramisu : Fully convolutional densenets for semantic segmentation. In *CVPRW*. IEEE, 2017.

[14] H. Jung, Y. Kim1, D. Min, C. Oh, and K. Sohn. Depth prediction from a single image with conditional adversarial networks. In *ICIP*, 2017.

[15] A. Kendall, V. Badrinarayanan, and R. Cipolla. Bayesian segnet : Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. *BMVC*, 2017.

[16] A. Kendall and Y. Gal. What uncertainties do we need in bayesian deep learning for computer vision ? *NIPS*, 2017.

[17] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. In *3DV*. IEEE, 2016.

[18] A. Levin, Y. Weiss, F. Durand, and W. Freeman. Understanding and evaluating blind deconvolution algorithms. In *CVPR*, 2009.

[19] F. Liu, C. Shen, and G. Lin. Deep Convolutional Neural Fields for Depth Estimation from a Single Image. *CVPR*, 2015.

[20] F. Liu, C. Shen, G. Lin, and I. D. Reid. Learning Depth from Single Monocular Images Using Deep Convolutional Neural Fields. *TPAMI*, 2015.

[21] F. Ma and S. Karaman. Sparse-to-dense : Depth prediction from sparse depth samples and a single image. *ICRA*, 2018.

[22] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley. Least squares generative adversarial networks. *arXiv preprint ArXiv :1611.04076*, 2016.

[23] M. Martinello and P. Favaro. Single image blind deconvolution with higher-order texture statistics. *VPCV*, 2011.

[24] P. K. Nathan Silberman, Derek Hoiem and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012.

[25] A. P. Pentland. A new sense for depth of field. *IEEE Trans. on PAMI*, 9, 1987.

[26] O. Ronneberger, P. Fischer, and T. Brox. U-net : Convolutional networks for biomedical image segmentation. *MICCAI*, 2015.

[27] A. Saxena, S. H. Chung, and A. Y. Ng. Learning Depth from Single Monocular Images. *NIPS*, 2006.

[28] A. Sellent and P. Favaro. Which side of the focal plane are you on ? In *ICCP*, 2014.

[29] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *ICLR*, 2015.

[30] P. P. Srinivasan, R. Garg, N. Wadhwa, R. Ng, and J. T. Barron. Aperture supervision for monocular depth estimation. *arXiv preprint arXiv :1711.07933*, 2016.

[31] P. Trouvé, F. Champagnat, G. Le Besnerais, and J. Idier. Single image local blur identification. *IEEE ICIP*, 2011.

[32] P. Wang, X. Shen, Z. Lin, S. Cohen, B. Price, and A. L. Yuille. Towards unified depth and semantic prediction from a single image. In *CVPR*, 2015.

[33] D. Xu, E. Ricci, W. Ouyang, X. Wang, and N. Sebe. Multi-scale continuous crfs as sequential deep networks for monocular depth estimation. *CVPR*, 2017.

[34] S. Zhuo and T. Sim. Defocus map estimation from a single image. *Pattern Recognition*, 44, 2011.