Deep Learning methods for monocular 3D vision systems

Rémy Leroy

université PARIS-SACLAY

Deep Learning methods for monocular 3D vision systems

Méthodes d'apprentissage profond pour systèmes de vision 3D

Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 580, Sciences et Technologies de l'Information et de la Communication (STIC)
Spécialité de doctorat : Sciences du traitement du signal et des images
Graduate School : Informatique et sciences du numérique. Référent : Faculté des sciences d'Orsay

Thèse préparée dans l'unité de recherche **Traitement de l'information et systèmes (Université Paris-Saclay, ONERA)**,
sous la direction de **Frédéric CHAMPAGNAT**, Directeur de recherche (ONERA),
la co-direction de **Bertrand LE SAUX**, Chercheur sénior (ESA-ESRIN),
le co-encadrement de **Pauline TROUVÉ-PELOUX**, Chargée de recherche (ONERA)

Thèse soutenue à Paris-Saclay, le 10 mars 2023, par

Rémy LEROY

Composition du jury

Membres du jury avec voix délibérative

Loïc DENIS Professeur, Université de Saint-Etienne	Président & Rapporteur
Thierry CHATEAU Professeur, Université Clermont Auvergne	Rapporteur & Examinateur
Bernadette DORIZZI Professeure émérite, Télécom SudParis	Examinatrice
Renaud MARLET Chercheur sénior, École des Ponts & Valeo.ia	Examinateur

THESE DE DOCTORAT
NNT : 2023UPASG021

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Méthodes d'apprentissage profond pour systèmes de vision 3D
Mots clés : apprentissage profond, nuage de points 3D, co-conception

Résumé : Dans cette thèse, nous étudions l'apport de l'apprentissage profond pour les systèmes de vision 3D monoculaire, de l'acquisition de l'image au traitement. Nous proposons d'abord Pix2Point, une méthode d'estimation de nuage de points 3D à partir d'une seule image en utilisant des informations de contexte, et entraînée avec une fonction de coût de transport optimal. Pix2Point réalise une meilleure couverture des scènes lorsqu'il est entraîné sur des nuages de points lacunaires que les méthodes d'estimation de profondeur monoculaire, entraînées sur des cartes de profondeur lacunaires. Deuxièmement, pour exploiter les indices de profondeur provenant du capteur, nous proposons une méthode de régression de profondeur à partir d'un patch défocalisé. Cette méthode surpasse la classification et la régression directe, sur données simulées et réelles. Enfin, nous abordons la conception d'un système de vision RVB-D, composé d'un capteur dont l'image est traitée par notre réseau de régression de profondeur basée sur la défocalisation et par un réseau de défloutage d'image. Nous proposons un cadre d'optimisation multi-tâches, conjointement aux paramètres des capteurs et des réseaux, et nous l'appliquons à l'optimisation de la mise au point d'une lentille chromatique. Le paysage d'optimisation présente plusieurs optima liés à la tâche de régression en profondeur, tandis que la tâche de défloutage semble moins sensible au paramètre de mise au point. En résumé, cette thèse propose plusieurs contributions exploitant les réseaux de neurones pour l'estimation 3D monoculaire et ouvre la voie d'une conception conjointe de systèmes RVB-D.

Title : Deep Learning methods for monocular 3D vision systems
Keywords : Deep learning, 3D point clouds, Co-design

Abstract : In this thesis, we explore deep learning methods for monocular 3D vision systems, from image acquisition to processing. We first propose Pix2Point, a method for 3D point cloud prediction from a single image using context information, trained with an optimal transport loss. Pix2Point achieves a better coverage of the scenes when trained on sparse point clouds than monocular depth estimation methods, trained on sparse depth maps. Second, to exploit sensor depth cues, we propose a depth regression method from a defocused patch, which outperforms classification and direct regression, on simulated and real data. Finally, we tackle the design of a RGB-D monocular vision system for which the image is processed jointly by our defocus-based depth regression method and a simple image deblurring network. We propose an end-to-end multi-task optimisation framework of sensor and network parameters, that we apply to the focus optimisation for a chromatic lens. The optimisation landscape presents multiple optima, due to the depth regression task, while the deblurring task appears less sensitive to the focus. This thesis hence contains several contributions exploiting neural networks for monocular 3D estimation and paves the way towards end-to-end design of RGB-D systems.

Table of contents

Synthèse en français

L'estimation 3D monoculaire est utile pour de nombreuses applications qui requièrent des solutions compactes, telles que la robotique, la réalité virtuelle, l'inspection industrielle ou médicale. Il s'agit d'un problème difficile à résoudre dû à l'ambiguïté de profondeur, mais les récentes avancées en matière d'apprentissage profond ont montré des résultats remarquables pour cette tâche. Dans cette thèse, nous avons exploré deux approches majeures pour améliorer l'estimation 3D monoculaire par apprentissage profond. La première est une approche à échelle globale utilisant des informations contextuelles pour estimer des nuages de points, et la seconde est une approche à l'échelle locale pour l'estimation de la profondeur en s'appuyant sur des indices de flou de défocalisation. Nous avons également abordé l'optimisation conjointe des paramètres d'un système optique afin de fournir aux modules de traitement davantage d'indices pour la tâche considérée.

La reconstruction et la compréhension d'une scène reposent sur les méthodes d'estimation de la 3D. Les méthodes les plus anciennes pour extraire la 3D reposent sur des images acquises par stéréo-photogrammétrie, mais l'apprentissage profond a récemment montré d'excellentes capacités pour l'estimation monoculaire de la profondeur. Ces résultats nécessitent la constitution d'un ensemble de données d'entraînement suffisamment vaste et riche, souvent le résultat d'un traitement fastidieux, comme par exemple la base de données de référence KITTI. Au lieu de cela, nous avons abordé le problème d'estimation monoculaire de nuages de points 3D extérieurs issus de données natives LiDAR parcimonieuses. Nous proposons Pix2Point, une approche utilisant l'apprentissage profond pour la prédiction de nuages de points 3D monoculaire, capable de traiter des scènes extérieures difficiles. Notre méthode s'appuie sur une architecture de réseau de neurones hybride 2D-3D entraînée de bout-en-bout par minimisation d'une divergence de transport optimal entre les nuages de points. Nous avons montré que notre approche simple permet d'obtenir une meilleure couverture 3D des scènes extérieures que les méthodes de l'état de l'art pour l'estimation monoculaire de la profondeur, entraînées dans des conditions similaires.

Pour un système optique donné, il est établi depuis plusieurs décennies que le flou de défocalisation peut être utilisé pour estimer la profondeur localement. Cette tâche est historiquement appelée *depth from defocus* (DFD). Elle a été traité par des méthodes non supervisées utilisant des *a priori* basés modèles, et plus récemment, par l'apprentissage supervisé de réseaux de neurones. La plupart de ces méthodes de DFD modélisent généralement cette tâche comme un problème de classification parmi un ensemble de flous de défocalisation potentiels liés à une profondeur. Cependant, la profondeur

est un paramètre continu et les approches de classification induisent des erreurs de quantification. Pour éviter ce problème, nous avons développé une nouvelle approche pour la régression continue de la profondeur à partir du flou de défocalisation sur des patchs, en adaptant un modèle de classification simple. Pour cela, nous utilisons pendant l'entraînement un codage de la profondeur réelle, dit de *soft-assignment*, en un vecteur de probabilité d'appartenance, puis une échelle de régression pour prédire les valeurs de profondeur intermédiaires. Notre méthode est plus performante que la classification et la régression directe sur des images simulées provenant d'ensembles de données de textures structurées ou naturelles, et sur des données réelles provenant d'une expérience de DFD active.

Le flou de défocalisation étant caractérisé par les paramètres du système d'acquisition optique, on peut se demander s'il existe une combinaison optimale de système optique et de traitement par réseau de neurones pour la DFD. Ce problème complexe peut être traité à l'aide de la *deep co-design*, une approche récente qui traite de l'optimisation conjointe d'un système optique et d'un réseau de neurones. Nous avons proposé d'utiliser cette approche pour l'optimisation conjointe de la mise au point d'une caméra et de notre réseau de régression du flou. Cette optimisation est réalisée grâce à un modèle optique basé sur le tracé de rayons différentiable, qui fournit un modèle réaliste de la caméra, y compris des aberrations optiques. L'utilisation du DFD pour l'estimation de la profondeur implique que la caméra acquiert des images de mauvaise qualité. Pour résoudre ce problème, il est possible d'utiliser un traitement de restauration d'image. Alors que dans la littérature les approches de *deep co-design* ne réalisent leur optimisation que pour une seule tâche, nous avons envisagé une approche multi-tâche pour la restauration 3D et de restauration d'image. Nous nous sommes intéressé à l'optimisation de la mise au point d'un système optique réel pour les tâches de DFD et de restauration d'image indépendamment dans un premier temps. De ces expériences, nous avons observé que l'optimisation conjointe optique/réseaux pour la tâche de DFD converge vers différentes valeurs de mise au point suivant l'initialisation de ce paramètre, tandis que l'optimisation pour la tâche de restauration d'image montre une certaine insensibilité du système complet par rapport à la mise au point. De plus, la valeur de mise au point trouvée pour le système offrant la meilleure performance pour chacune des tâches est différente. Nous avons ensuite mis en œuvre une optimisation multi-tâche simple. Cette approche permet d'atteindre des systèmes plus performants en DFD qu'en mono-tâche. En résumé, cette thèse propose plusieurs contributions en apprentissage profond pour l'estimation 3D monoculaire, et ouvre la voie vers des systèmes de vision monoculaire 3D mieux conçus.

1 - Introduction

1.1 . Motivation

Living beings have to extract information from their environment through their senses to locate themselves in it, build a representation of it and make decisions.

Some species, like bats or dolphins, use an active approach to build that representation space. Such species emit a sound wave in a given direction and collect echoes from their environment, these echoes are processed by their neural system to infer a spatial (and dynamic) representation of their environment. Technologies like sonar, radar and LiDAR are based on this principle. Other species, like humans, developed a passive approach to do so by using the parallax from their binocular vision.

Computer vision aims to give machines the same abilities to observe, analyse and understand the world they occupy, given the visual information from one or more cameras. Computer vision comprises various challenges for scene understanding, especially the challenge of 3D geometry estimation. Solving this challenge is helpful for many applications, such as autonomous driving vehicles, industrial inspection for manufactured parts, and virtual or augmented reality. Historical 3D estimation techniques also exploit the parallax by making use of two or more images of the scene [Faugeras, 1993, Hartley and Zisserman, 2004]. These techniques rely on visual feature extraction in each of the images, followed by a corresponding feature matching. The same feature will appear at a different location in each image, and this location disparity informs us of the depth of the feature point relative to the calibrated imaging system. Most advances in multi-view 3D estimation are related to feature extraction and matching. The Semi-Global Matching algorithm (SGM) [Hirschmueller, 2008] is one of the most popular stereo-matching methods for disparity map estimation, as it proposes a good accuracy/computational complexity for real-time applications.

However, using information from two cameras is not always possible, either for cost or space constraints. Moreover, a precise calibration between the camera and a large baseline is required for accurate measurements. Plenoptic cameras use micro-lens matrices in front of the sensor to capture the light field of the scene, *i.e.*, the light intensity as well as the direction of light rays. Plenoptic images can be post-processed to refocus the image and to give 3D information of the scene [Ng et al., 2005, Perwass and Wietzke, 2012], however, 3D capabilities are limited compared to previous methods due to the small baseline between micro-lenses. 3D estimation is also possible with one camera using motion parallax. This is approach is referred to in the literature

as Structure from Motion (SfM) [Ullman, 1979]. The 3D reconstruction quality depends on a good pose estimation between the multiple views and is also subject to scale uncertainty.

Previously stated methods lean on multiple views of the environment to perform the reconstruction. However, in this thesis, we are interested in a single-view 3D estimation approach, which offers a solution that is more compact, economical, and that does not require knowing the relative pose of the images. We will tackle this challenging task by using deep learning techniques, which demonstrated outstanding performances for many computer vision tasks, including single-view 3D estimation.

1.2 . Single-view 3D estimation

To complete the task of single-view 3D estimation, **learning-based techniques** have been investigated and improved for the past few years [Saxena et al., 2006, Eigen et al., 2014, Carvalho et al., 2018b, Lee et al., 2019a, Bhat et al., 2021] and reached solid performances. All of these methods estimate depth maps, *i.e.*, a raster image which provides a depth value for each pixel. Yet, 3D can also be represented using point clouds, which are sets of points sampling the surfaces. This latter form is commonly used in robotics as it corresponds to the raw measurement for active 3D sensors like LiDAR. Although useful, only a few deep learning methods tackled the problem of single-view 3D point cloud estimation [Fan et al., 2017, Xia et al., 2018, Mandikal and Radhakrishnan, 2019], all applying only to 3D object models, and none of them deal with real outdoor scenes. Thereby, the first question addressed in this thesis is "*How can a deep learning method for point cloud estimation from a single image extend to real outdoor scenes?*".

For training their models, the aforementioned methods depend on **rich and highly post-processed data**, such as the NYU depth dataset V2 [Silberman et al., 2012] for an indoor setting, or the KITTI vision benchmark suite [Geiger et al., 2013] for an outdoor setting. These clean databases are the result of computationally expensive processing to accumulate and filter the points, which we would like to avoid for a practical and cost-effective motive. Therefore, we can wonder "*How would a deep learning method perform using unrefined and sparse data for training the neural network?*".

The aforementioned methods use the context contained in the images to make their estimations, but physical cues on the sensor can be used as well. One such visual depth cue is the **defocus blur**, which can guide monocular 3D estimation, either using several views as in [Pentland, 1987], or a single view using statistical priors [Zhu et al., 2013, Trouvé et al., 2011, Buat et al., 2021]. Regarding deep learning techniques, defocus blur can also be added to the context information to help these models to perform depth estimation on full-

scale images [Carvalho et al., 2018a, Anwar et al., 2021]. However, the estimates of these previous methods use both the context and the blur information, so we can wonder "*How would a deep learning model perform using only the defocus blur to **regress** the depth locally?*".

The defocus blur is characterised by the optical system that constructs the image on the sensor. Previous works proposed to use coded aperture, diffractive lenses, or lenses with chromatic aberrations to enhance the depth information on the sensor, which can be retrieved using corresponding estimator [Levin et al., 2007, Zhou et al., 2009, Martinello and Favaro, 2011, Trouvé et al., 2013]. The parametrisation of those unconventional optics is obtained via the specification and optimisation of performance criteria. The joint optimisation of the optics and the estimator is referred to as **co-design**, and a particular instance of it, called *deep optics* or *deep co-design*, has recently been addressed and considers the estimator as a neural network model. *Deep co-design* approaches require to have a fully differentiable image formation model with respect to its parameters. Current *deep co-design* approaches consider either Fourier optics [Haim et al., 2018, Chang and Wetzstein, 2019, Ikoma et al., 2021], or differentiable ray tracers [Sun et al., 2021, Halé et al., 2021]. The purpose of co-design for the problem of Monocular Depth Estimation (MDE), the task of interest for this thesis, is to find the optical system that will inject as much depth information in the image as possible using defocus blur. On the other hand, this blur will deteriorate the overall image quality, so the question is then "*How to optimise jointly optics and neural network parameters for **3D estimation and image deblurring?***".

1.3 . Objectives & Contributions

In this thesis, we demonstrate how machine learning techniques can improve all parts of the 3D perception pipeline, from smart sensing to information processing, by addressing the questions that emerged from the shortcomings of previous methods :

1. *How can a deep learning method for point cloud estimation from a single image extend to real outdoor scenes?*
2. *How would a deep learning method perform using unrefined data for training the neural network?*
3. *How would a deep learning model perform using only the defocus blur to regress the depth locally?*
4. *How to optimise optics and deep learning model parameters jointly for both 3D estimation and image deblurring?*

Figure 1.1 gives an overview of the thesis's main questions and chapter organisation, as detailed hereafter.

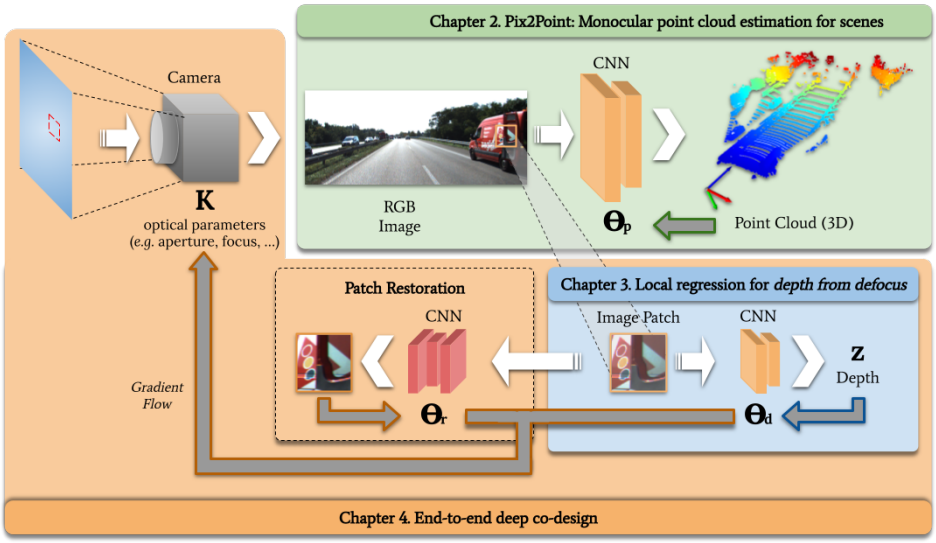


Figure 1.1 – Outline of the thesis with respect to the 3D vision processing chain.

Questions 1 and 2 are addressed in **chapter 2**, in which we propose Pix2Point, a deep learning model trained using point clouds for outdoor scenes, by using the famous KITTI vision benchmark dataset that provides image/point-cloud matching data. Question 3 is addressed in **chapter 3**, in which we propose a lightweight deep learning approach to regress locally the depth using defocus blur. Our approach does not require any scene prior or optic calibration. It is validated experimentally in the context of surface inspection. Question 4 is addressed in **chapter 4**, in which we explore the joint co-design of an optical system for 3D estimation and image restoration tasks. Lastly, we conclude and open new questions that would lead to future works in **chapter 5**.

1.4 . Publications

This work led to multiple scientific publications and talks.
One Published article in a journal with a review committee :

- **R. Leroy**, P. Trouvé-Peloux, B. Le Saux, B. Buat, F. Champagnat. (2022). "Learning local depth regression from defocus blur by soft-assignment encoding". Applied Optics. 61(29).

One published paper in an international conference with a reading committee and proceedings :

- **R. Leroy**, P. Trouvé-Peloux, F. Champagnat, B. Le Saux, M. Carvalho. (2021). "Pix2Point : Learning outdoor 3D using sparse point clouds and optimal transport". Int. Conf. on Machine Vision and Applications (MVA). *poster*.

Published papers in national conferences with reading committees and proceedings :

- **R. Leroy**, B. Le Saux, M. Carvalho, P. Trouvé-Peloux, F. Champagnat. (2020). "Pix2point : prédiction monoculaire de scènes 3D par réseaux de neurones hybrides et transport optimal". RFIAP.
- **R. Leroy**, P. Trouvé-Peloux,B. Le Saux, B. Buat, F. Champagnat. (2022). "Régression locale de la profondeur grâce au flou de défocalisation et à un réseau de neurones entraîné par soft-assignment". GRETSI. *poster*.

Oral presentations (without proceedings) :

- **R. Leroy**, P. Trouvé-Peloux, B. Le Saux, F. Champagnat. "Towards end-to-end design of a monocular sensor for 3D point cloud prediction". CLIM workshops 2021.
- **R. Leroy**, P. Trouvé-Peloux, B. Le Saux, F. Champagnat. "Estimation Locale de Flou de Défocalisation par Réseau de Neurones" GDR ISIS JIONC 2022.
- **R. Leroy**, P. Trouvé-Peloux, B. Le Saux, F. Champagnat. "Deep Neural Networks for 3D Monocular Estimation". ODAS 2022.

To be submitted :

- **R. Leroy**, M. Dufraisse, P. Trouvé-Peloux, B. Le Saux, J.-B. Volatier, F. Champagnat. "Multi-Task Deep Co-design". Applied Optics.

2 - Monocular 3D estimation from sparse training data

Contents

Recently, deep learning techniques have revolutionised 3D estimation from images, allowing to obtain excellent results even with a single view [Eigen et al., 2014, Carvalho et al., 2018b, Fu et al., 2018, Amiri et al., 2019, Lee et al., 2019a]. These impressive results rely upon large and highly accurate databases like KITTI [Geiger et al., 2013] that involve the simultaneous collection of stereo pairs and LiDAR data, post-processing and temporal integration in order to provide accurate, reliable and dense ground truth for learning purposes. The overall process requires large-scale cooperation and is therefore lengthy. In this chapter, in contrast, we are interested in solving monocular 3D estimation for outdoor scenes using rough unfiltered data such as sparse LiDAR point clouds.

Most state-of-the-art methods for 3D reconstructions usually use depth map as a means of representation, a raster image indicating the distance from the point of view to the surface of the observed scene pixel-wise [Eigen et al., 2014, Saxena et al., 2006, Carvalho et al., 2018b, Lee et al., 2019a]. 3D can also be represented as a point cloud corresponding to a 3D sampling of the considered scene surfaces, which can be acquired using native 3D sensors such as LiDARs. Unlike depth maps, this latter mode of representation does not suffer from alignment and rigid sampling on the image grid. Besides, depth maps are usually smoothed to provide dense results, leading to less reliable depth values locally. Finally, the Pseudo-LiDAR [Wang et al., 2019] and Pseudo-LiDAR ++ [You et al., 2019] methods have shown a significant beneficial contribution of the point cloud representation compared to depth maps for the task of detection and localisation of obstacles in scenes. Therefore, the point cloud representation is a good candidate for 3D estimation. Hence, in this chapter, we choose to output directly 3D point clouds from a single image, as illustrated by Figure 2.1.

However, handling 3D point clouds leads to several technical challenges related to architecture able to process such data, learning scheme and associated loss to measure point cloud discrepancy. In this thesis, we have tackled

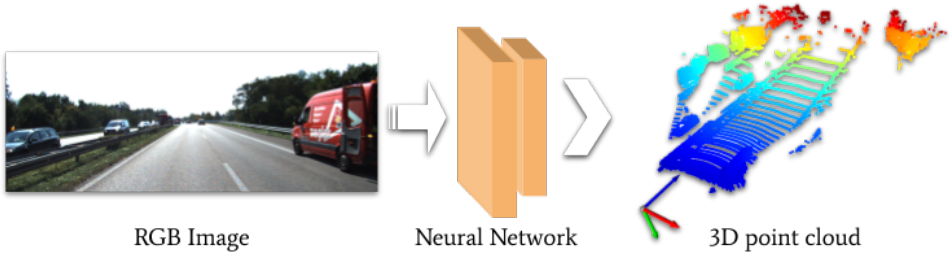


Figure 2.1 – A RGB image is translated directly to a 3D point cloud by a trained neural network.

these challenges with the development of a method named Pix2Point, a deep-learning approach for single-view 3D point cloud estimation. This method is trained on a sparse point cloud dataset.

2.1 . Related Work

Single image 3D estimation has been addressed in terms of 3D point set with PSGN by Fan *et al.*, [Fan et al., 2017], a method that aims to predict an unordered set of points sampling the envelope of an object using a single view of it and its location in the image. Mandikal and Babu [Mandikal and Radhakrishnan, 2019] address the limitations of PSGN regarding the poor number of predicted points with DensePCR, a pyramidal structure allowing to multiply the number of points. Xia *et al.*, [Xia et al., 2018], also tackles the generation of a monocular point cloud for objects using prior knowledge of their shapes, making it robust to occlusions and varying poses. Sun *et al.*, [Sun et al., 2019] have explored monocular point cloud generation using self-supervision mechanisms. Lastly, a generative flow-based model allowing single-view object point cloud prediction has been proposed with C-flow [Pumarola et al., 2020]. It leverages a back-and-forth prediction loop from image to point cloud, then to image for consistency.

It is important to note that the aforementioned point cloud works only consider the reconstruction of a single **3D object model**, *i.e.*, on data that are obtained through demanding procedures, either scanned objects using RGB-D sensors or laser scanners, or handcrafted models. These procedures do not apply to real-life scenes with various settings and where lies multiple objects. [Denninger and Triebel, 2020] tackles the problem of monocular volumetric reconstruction with occlusion completion only for **indoor scenes**. In addition to the input RGB image, this approach requires a corresponding normal image that is hardly obtainable for outdoor scenes.

None of these methods deals with the reconstruction of complex **outdoor**

scenes in the form of 3D point clouds, solely conditioned by a single RGB image, and trained on sparse point clouds.

2.2 . Contributions and chapter organisation

In this chapter, we present our method for 3D point cloud estimation from a single image. Our contributions are :

- a first approach to reconstruct a 3D point cloud for an entire **outdoor scene** given only a single image using a 2D-3D hybrid neural network architecture, inspired by DensePCR [Mandikal and Radhakrishnan, 2019].
- an **end-to-end learning** scheme of the hybrid model using a sparse point clouds dataset.
- a **first benchmark** for the single-view sparse-point-cloud estimation problem with a comparison of our method to state-of-the-art monocular depth map prediction methods.
- an **ablation study** of the neural network architecture.
- a study about the sensitivity of our models to the **types of point clouds** used for training.

This chapter is structured as follows : in Section 2.3, we first describe the various components of the proposed method, namely Pix2Point, *i.e.*, the neural network architecture and the point cloud loss functions for the optimisation process, as well as the quality criteria for point cloud reconstruction. Then in Section 2.4, we define various experimental settings to test empirically several neural network architectures, loss functions, and types of data used for training. We present and analyse the results for each scenario in Section 2.5. We conclude this chapter with a discussion on the limitations and improvements to the Pix2Point method in Section 2.6.

This work appeared in the proceedings for the 2020 French National Conference RFIAP [Leroy et al., 2020], and for the 2021 International Machine Vision and Applications conference (MVA) with a poster presentation [Leroy et al., 2021].

2.3 . Description of Pix2Point

Pix2Point, illustrated by Figure 2.2, has the following characteristic : it predicts a set of 3D point coordinates with an arbitrary number of elements given a single colour image, the number of elements, or points, is fixed before training. There are two principal components to the method : the neural network architecture, and the parameter optimisation scheme. We will describe each component in that order.

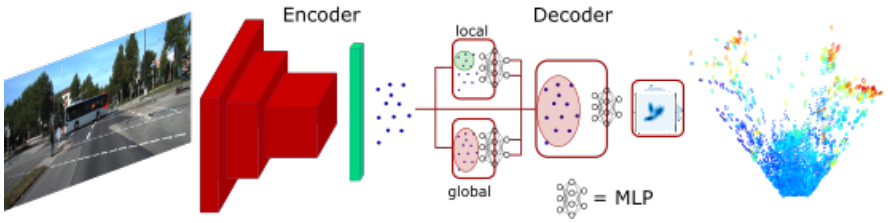


Figure 2.2 – Pix2Point : 3D point set prediction for real outdoor scenes from a single image. A first 2D CNN module encodes the RGB image for the following fully connected layer to predict a coarse 3D point cloud. Then a 3D point-wise densification module grows of the number of points of this point cloud using PointNet-like MLP. All models are trained end-to-end to minimise point set distances, *e.g.* , Optimal Transport or chamfer distances.

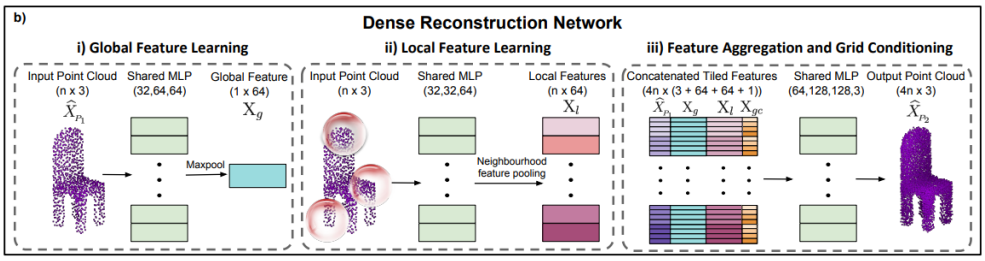


Figure 2.3 – Illustration of the densification method from [Mandikal and Radhakrishnan, 2019]. The method comprises the computation of one global feature vector for the whole point cloud and a local feature vector for each point, that will be used to compute the denser point cloud.

2.3.1 . Neural Network Architecture

Similarly to DensePCR [Mandikal and Radhakrishnan, 2019], Pix2Point's architecture consists of an encoding module to predict a first coarse point cloud that will be enriched using a densification module.

Encoding The encoding block is a series of convolution, pooling and normalisation layers to extract a feature description of the full RGB input image, which is then processed by a fully connected layer to obtain a first coarse set of 3D point coordinates. This is exactly the fully connected layer that fixes the image resolution the network can process, as well as the total number of 3D points. We try and compare several renowned feature encoding approaches following either VGG [Simonyan and Zisserman, 2014], DenseNet [Huang et al., 2017] and ResNet [He et al., 2016] architectures in section 2.5. We refer to them as *backbones*.

Decoding We refer to decoding as the densification of the first coarse 3D point cloud, as illustrated in Figure 2.3. We duplicate every point k times, and to describe each point we concatenate : the 3D coordinates, both global point cloud feature vector and the corresponding local feature vector, obtained using dedicated PointNet-like shared Multi Layer Perceptron (MLP) [Qi et al., 2017a, Qi et al., 2017b], and lastly a grid alignment feature vector in order to identify every clone of the same point and to suggest geometric information between every clone. This point description is processed by another shared MLP resulting in 3D coordinates for 1 point.

2.3.2 . Optimisation Scheme

To account for simplicity, the parameters of our models are trained in an end-to-end fashion unlike DensePCR [Mandikal and Radhakrishnan, 2019]. The choice of the loss function used for training enforces the achievable performances of our approach. Unlike the depth map prediction methods which exploit the grid structure of the image for the evaluation of errors, our method uses distances between unordered point sets. These distances require an additional computationally expensive step to match points between the predicted and the target point clouds. In the following, we expose two usual distances for this task.

Chamfer distance The chamfer distance is the average of squared Euclidean distances to the nearest neighbour from one set to the other. It is defined between two point-clouds S_1 and S_2 as follows

$$d_C(S_1, S_2) = \frac{1}{|S_1|} \sum_{x \in S_1} \Delta(x, S_2) + \frac{1}{|S_2|} \sum_{y \in S_2} \Delta(y, S_1), \tag{2.1}$$

where $\Delta(\cdot, S) = \min_{y \in S} \| \cdot - y \|_2^2$.

Figure 2.4 illustrates the nearest neighbour pairing used to compute the chamfer distance for a 1D example.

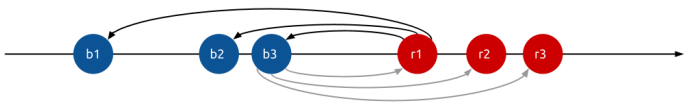


Figure 2.4 – Nearest neighbour pairing for the chamfer distance computation for a 1D case.

Optimal Transport or OT distance To compute this distance, also known as Earth Mover's distance, one has to find a one-to-one mapping ϕ from one

set to the other that minimises the sum over each point of the squared distance between them and their corresponding image :

$$d_{OT}(S_1, S_2) = \min_{\phi: S_1 \to S_2} \sum_{x \in S_1} \|x - \phi(x)\|_2^2. \tag{2.2}$$

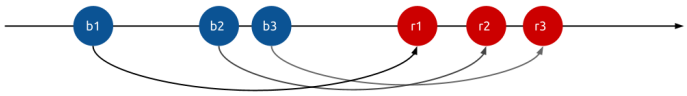


Figure 2.5 – Optimal transport pairing for a 1D case.

This minimised sum is called the OT distance and it informs about the eventual discrepancy between point sets distributions. The above formulation requires that both point clouds have the same cardinality, this will be the case in our experiments, however, it is possible to slightly change the problem formulation to handle unbalanced cardinality.

Figure 2.5 illustrates the optimal transport map, deduced from the minimisation problem, used to compute the OT distance for a 1D example.

The exact computation of an OT distance is time and memory expensive especially for several thousand elements lying in a space strictly higher than 2 dimensions, hence, we consider in our work an approximation of the OT distance obtained by adding a regularising term and using the Sinkhorn-Knopp algorithm [Cuturi, 2013, Feydy et al., 2019].

Test on a toy example To provide a better understanding of the convergence behaviour of each distance function, we first consider the following 2D toy example, as shown by Figure 2.6. We set a target point cloud with 100 fixed elements, and then define a source point cloud, whose elements are given with random initial coordinates. The objective is to move the elements of the source point cloud altogether in order to fit the target point cloud. The update of the coordinates for the source point cloud is performed iteratively by minimising either distance function using a gradient descent algorithm. Even though the chamfer distance and the OT distance have a similar objective, minimising those two distances for fitting a target point cloud from a random initial point cloud leads to a significantly different outcome. Indeed, the source point cloud resulting from the minimisation of the chamfer distance exhibits two significant behaviours. The first is that multiple source points tend to stack over one target point, and the second is that one source point finds itself lying in the middle of its closest target points. Those two kinds of local optima can be easily interpreted, as the chamfer distance is constructed with a sum of

independent local terms. In this regard, the chamfer distance is more sensitive to the initialisation and to the order of magnitude of the learning rate. In contrast, the source point cloud obtained by minimisation of the OT distance offers greater coverage of the target point cloud, this is coming from the global minimisation term of the OT distance.

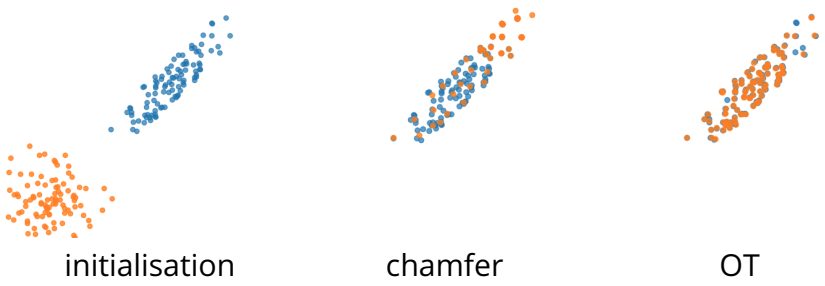


Figure 2.6 – 100 2D point coordinates optimised by gradient descent algorithm (orange) according to the chamfer distance and the OT distance with respect to the target point cloud (blue).

2.3.3 . Quality measures for point cloud reconstruction

From either optimisation, we observe two distinct qualities for the final source point cloud with respect to the target point cloud ; The first quality is about the precision of the point coordinates ; The second quality is about the overall coverage of the target by the source point cloud. To assess these qualities, we use the performance criteria from [Tylecek et al., 2018], namely accuracy and completeness. We also define and propose to use relative accuracy. All 3 measures are defined hereafter.

Completeness is the coverage in per cent of the target point cloud by the predicted points. A target point is covered if a predicted point lies in its surrounding (*i.e.*, fixed radius ball). This is illustrated for a 2D elementary configuration in Figure 2.7.

Accuracy is the distance d, in meter, from the r-th percentile of the distances to the nearest neighbour, from the predicted point cloud to the ground-truth point cloud. It measures the longest distance to the nearest neighbour among the predicted points closest to the ground truth. Figure 2.8 illustrates the accuracy measure for the same 2D elementary configuration.

Relative accuracy is similar to the accuracy, where every distance to their nearest neighbour is divided by the distance of the corresponding target point

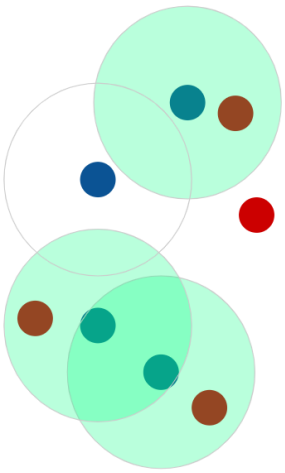


Figure 2.7 – Illustrated example of the completeness for a given radius threshold value, of the 2D blue point cloud by the red point cloud. In this case, the completeness is $3/4$, or 75%.

from the origin. It provides a higher penalty for short-range prediction errors.

In the following, we formally define these measures. First, we define P_j and T_j as the predicted and target point clouds of the j-th scene respectively. Every point from one point cloud is provided with the nearest neighbour distance to the other point cloud. $\delta_i^{(j)} = \min_{y \in T_j} \|x_i - y\|$ is the distance from a predicted point $x_i \in P_j$ to the closest target point. Reciprocally we note $\gamma_i^{(j)} = \min_{x \in P_j} \|x - y_i\|$ where $y_i \in T_j$.

We also provide respective relative distances $\delta_i^{rel(j)} = \frac{\min_{y \in T_j} \|x_i - y\|}{\|y\|}$, and $\gamma_i^{rel(j)} = \frac{\min_{x \in P_j} \|y_i - x\|}{\|x\|}$.

Let $\Delta = \{\delta_i^{(j)}, \forall j, i\}$ and $\Gamma = \{\gamma_i^{(j)}, \forall j, i\}$, we compute the performance measures as follows :

Accuracy	accuracy(r) = {d s.t. $\frac{\lvert\Delta<d\rvert}{\lvert\Delta\rvert} = r$}
Relative accuracy	relative accuracy(r) = {d s.t. $\frac{\lvert\Delta^{rel}<d\rvert}{\lvert\Delta^{rel}\rvert} = r$}
Completeness	completeness(d) = $\frac{\lvert\Gamma<d\rvert}{\lvert\Gamma\rvert}$

where d is the nearest neighbour distance threshold, $r \in]0, 1]$ and $|\cdot|$ denotes the cardinality.

2.4 . Experiments on Real Outdoor Scenes

This section presents the dataset we are considering for real scene point cloud estimation from a single image and some implementation details.

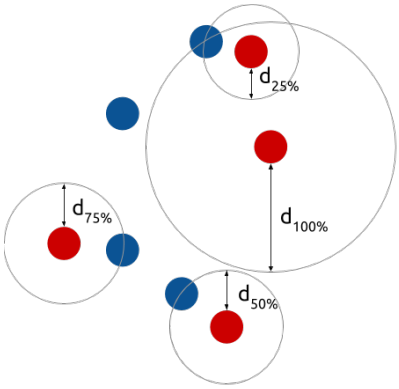


Figure 2.8 – Illustrated example of the accuracy for the 2D red point cloud, with respect to the blue point cloud. Nearest blue neighbour distances from red points are computed and ordered; The accuracy for a given percentile is the corresponding nearest neighbour distance.

2.4.1 . Dataset

To assess our method, we operate on RGB image sequences of real urban scenes and corresponding LiDAR point cloud acquisitions from the KITTI depth estimation benchmark dataset [Geiger et al., 2013]. Every scene point cloud is an accumulation of filtered LiDAR acquisitions over a few successive time instants. We use the split defined by [Eigen et al., 2014], that is 22 600 training scenes and 697 testing scenes.

2.4.2 . Implementation Details

Experiments were conducted using the *Pytorch* Framework [Paszke et al., 2019]. We kept the original image resolution and cropped every picture to 1224×370 pixel definition. Due to heavy computational cost for loss back-propagation, parameters were updated after every sample forward, making batch normalisation ineffective. Instance normalisation was applied instead [Ulyanov et al., 2016]. The number of predicted points was determined to fully load the 8GB GPU during training. Therefore, 2500 elements point clouds are first predicted by the fully connected encoding module, then up-scaled by a DensePCR-like module making a point cloud with 10k elements.

Training vs. Testing point clouds : Using an OT loss enables, in principle, the comparison of any ground-truth point cloud to the predicted one, however, in practice, computation and optimisation of an OT loss are computationally much more efficient with point sets of equal cardinality. Therefore, ground-truth point cloud databases are randomly sub-sampled to 10k points, which is as many points as Pix2Point predicts. In comparison with depth maps, 10k points amount to 15% of the depth map information on average, which makes our method train on sparse data. When testing we measure perfor-

mances with respect to the whole ground-truth point cloud.

2.5 . Experimental Results

In this section we first present the performances of Pix2Point using various encoding backbones and losses, then we compare our method to depth map prediction approaches through evaluation metrics defined in 2.3.3.

2.5.1 . Network parameter study

We trained several models with varying encoding backbones and loss functions. We considered the following configurations : Pix2Point architecture with VGG backbone and training on the minimisation end-to-end of either the chamfer or OT distance, and Pix2Point with ResNet backbone and minimising the OT distance. The performances of these models are given in Table 2.1 respectively as P2P-VGG-C, P2P-VGG-OT and P2P-ResNet-OT. From these figures, we can notice that the minimisation of chamfer distance thrives toward predictions with low local error, and minimising the OT distance grants predictions with higher completeness, hence, better coverage of the scenes. In order to find if these distances could help each other, combinations of both distances have been tested. However, they lead to convergence issues during training and overall worse performances due to opposite objectives of the distances. Changing the backbone from VGG to ResNet has also a slight impact on the completeness and accuracy. The small gain in relative accuracy indicates that far predictions are more accurate. We also provide a comparison to a similar image-to-point-cloud approach, DensePCR [Mandikal and Radhakrishnan, 2019], initially proposed for 3D graphics models.

Table 2.1 – Comparison of 3D scene reconstructions on KITTI. We report completeness and accuracy. All methods are trained with 10k point clouds.

Approaches	Completeness ↑(in %)			Accuracy ↓	
	50cm	25cm	10cm	in m	rel.
P2P-ResNet-OT	**71.35**	**48.82**	**15.12**	1.92	0.18
P2P-VGG-OT	67.4	47.7	14.7	1.79	0.19
P2P-VGG-C	64.4	36.0	8.0	**0.85**	**0.05**
DensePCR	59.9	23.5	3.5	*1.77*	0.18
BTS	67.59	31.29	6.28	1.23	0.06
AdaBins	65.86	27.52	5.71	1.25	0.06

2.5.2 . Comparison to depth prediction approaches

The current dominant approach to 3D estimation from a single image consists of predicting corresponding depth maps by leveraging the power of image-to-image translation networks. On KITTI, these methods are trained on pseudo-dense depth maps built by accumulating several consecutive LiDAR acquisitions. For comparison in a similar setting, we train two state-of-the-art models for monocular depth estimation from the KITTI challenge [1], namely BTS [Lee et al., 2019a] and AdaBins [Bhat et al., 2021], on the same 10k-point-cloud as Pix2Point. At inference time, dense depth maps are projected back to 3D using known camera parameters. Performance comparison with various flavours of Pix2Point is reported in Table 2.1, where we choose to measure the accuracy at $r < 90\%$ to include most of the points and discard eventual outliers, and the completeness is measured for neighbourhood radii of 50cm, 25cm and 10cm.

These results reveal that Pix2Point, with only 15M parameters, trained with Chamfer distance performs better than BTS and Adabins, respectively 45M and 78M parameters. Moreover, when trained with the OT distance, Pix2Point accuracy decreases but it covers three times more points than depth map approaches for the closest neighbourhoods. These observations can be made through Figure 2.9 where we show point cloud predictions and the coverage error map for each method (for comparison all predictions are visualized with 10k points). This error map displays for each ground-truth point the distance to its nearest predicted point. We display the error from 0 to 50cm using the jet colour map. While AdaBins and BTS preserve fine features, all Pix2Point variants achieve better coverage of the scene and a lower error, especially for far-away elements, that are not retrieved by AdaBins and BTS (see for instance the right part of the bottom scene).

In this section, we showed experimentally the benefit of training the model using 3D point clouds to recover a good coverage of the scene, especially when the data is spatially sparse.

2.5.3 . Sensitivity to point clouds nature

The surfaces of a scene can be punctually sampled in many various ways. In this section, we interrogate the training of a neural network model to point clouds obtained from different sampling schemes.

Type of point cloud sampling

For this experiment, we considered the following sampling scheme for every image of the KITTI benchmark dataset, providing different types of point clouds :

1. http://www.cvlibs.net/datasets/kitti/eval_depth.php?benchmark=depth_prediction

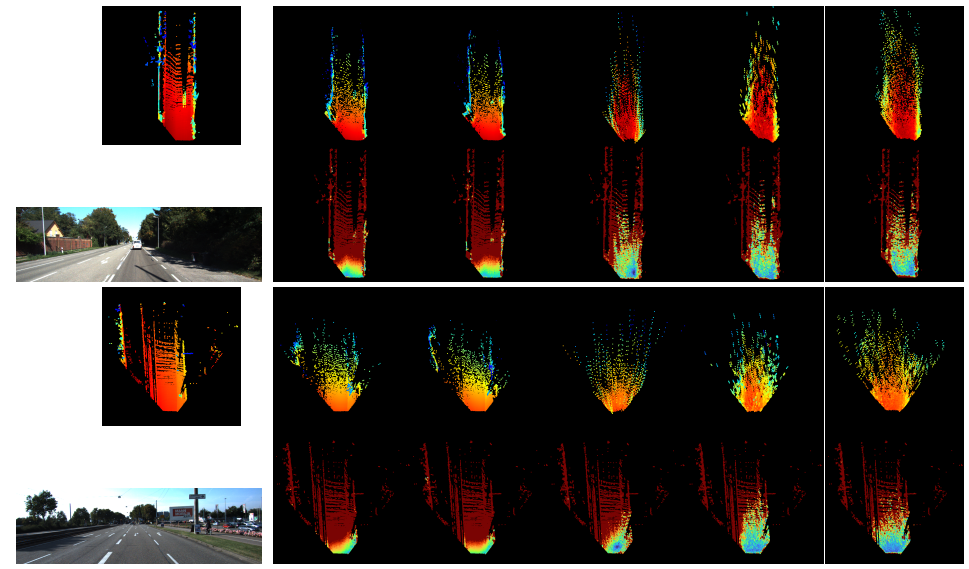


Figure 2.9 – For each scene, first row : 3D ground-truth and predictions for the RGB image according to AdaBins [Bhat et al., 2021], BTS [Lee et al., 2019a], our Pix2Point VGG-chamfer, VGG-OT and ResNet-OT, all trained on 10k points. We follow the 3D representation of [Caccia et al., 2019]. Bird's eye view where the colour encodes the altitude. Second row : the input RGB image and the ground-truth-to-prediction error map for each method. Errors are from 0 (blue) to 50cm (red).

Stereo keypoints Extraction and matching of key points from salient parts in stereo images to obtain 3D coordinates. Extracting this type of point cloud requires 2 finely calibrated cameras and elementary geometry. Spatial resolution and coordinate confidence for the extracted point cloud depend on the image resolution, the baseline of the binocular system, and the texture of the scene.

LiDAR Raw data are acquired from an active laser scanner by time-of-flight measurement. These data represent point clouds with low spatial resolution, whose density decreases with the distance, and showing a characteristic streak structure from the rotary motion of the sensor. Extracting this type of point cloud requires active laser scanners, which sample points regardless of the texture, but can also generate many outliers from the reflection of incident rays.

KITTI-Depth Combination of the two previous methods, accumulated over several frames. The stereo keypoint point cloud is used to filter out outliers

of the accumulated LiDAR point clouds.

Experimental settings

Regarding the neural network architecture, we considered the Pix2Point architecture with a VGG encoding backbone and trained one model per type of point cloud, *i.e.*, 3 models.

Training vs. Testing The models are learned using only one type of point cloud of 10k elements. Test point clouds have also 10k elements and are of the same type that the training point clouds.

Table 2.2 – Performances in completeness for various threshold values, and in accuracy at 90%, for Stereo, LiDAR and KITTI-Depth point clouds. Models are trained and tested on point clouds having 10k elements. Best values are highlighted in bold.

Type	Completeness ↑(in %)			Accuracy ↓
	1m	50cm	10cm	in m
Stereo	51.39	27.0	0.75	6.02
LiDAR	86.11	68.16	10.23	2.29
KITTI-Depth	**87.39**	**70.88**	**15.28**	**2.18**

Performance comparison Table 2.2 shows the performance in completeness and accuracy of the model trained for each point cloud type. For the stereo point clouds, only 51.39% of the target is covered at 1m, down to 0.75% at 10cm, and 90% of the predicted points are at less than 6.02m from any target point. These are poor performances in comparison to training on LiDAR or KITTI-Depth point clouds, which achieve similar performances; slightly more than 86% completeness at 1m and 10% at 10cm, and 90% of the predicted point at less than 2.3m from any target point. This gap in performance shows that training on stereo point clouds, compared to LiDAR-like point clouds, is a more difficult task.

To understand the origin of this adversity, we can look at the distribution of the points for both point cloud types,*i.e.*, stereo and LiDAR-like point clouds. Figure 2.10 shows a top view of the corresponding stereo (blue) and KITTI-Depth (red) point clouds for a given KITTI outdoor scene (RGB image). KITTI-Depth point clouds are well diffused over the scene, with a high density of points closer to the camera that decreases with the distance. Meanwhile, stereo point clouds have a greater spatial dispersion and higher distribution

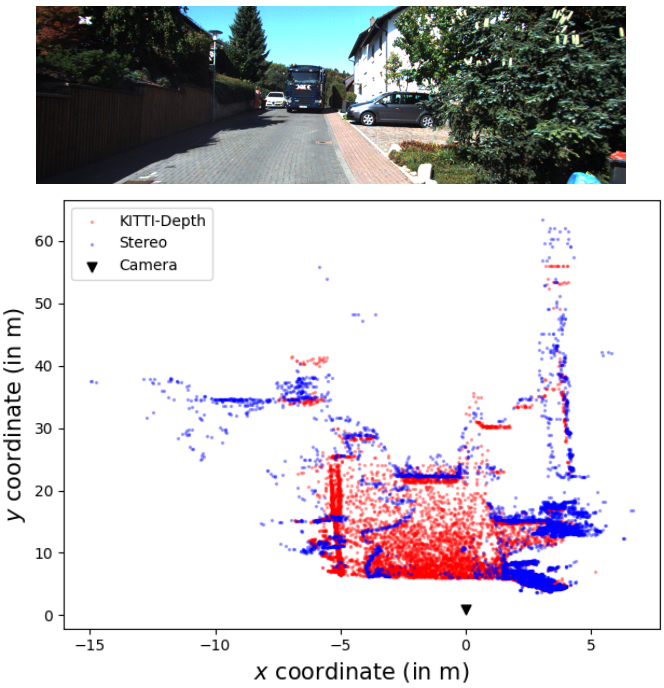


Figure 2.10 – (Top) RGB image of an outdoor scene from the KITTI dataset, (Bottom) Top view of overlapping KITTI-Depth (red), and stereo keypoints (blue) point clouds for the corresponding scene.

variability through training and testing examples. We can also note that, unlike LiDAR points, stereo points are extracted based on sharp textures in the image, and not sharp geometry variation in the scene, as we can see in the example of Figure 2.10, stereo points lying on the road mainly come from the border of the shadow projected on the road.

Point clouds obtained from the stereo sampling are noticeably inefficient in training our model despite its affordable passive nature. In contrast, point clouds actively acquired are more stable for training our model with a slight gain when the data is post-processed.

2.6 . Conclusion

In this chapter we presented an approach to tackle the problem of **point cloud reconstruction for complex outdoor scenes from a single RGB image**, using a lightweight 2D-3D hybrid neural network. The proposed method recovers properly distributed point clouds by taking advantage of an **optimal transport loss**. We also provided the first benchmark for this novel task on the KITTI dataset and introduced performance metrics to assess the quality of point cloud reconstruction. We show that our method outperforms state-of-

the-art depth map prediction methods when trained with **sparse data**. We also showed that point clouds from active acquisition methods, which have uniform coverage of the scene, were beneficial to the performance of the estimation model, compared to point clouds from passive approaches.

Perspectives for improvements Some parts of the method can be addressed to improve the point cloud estimation.

First, the coarse scale point cloud is constructed by the fully connected layer using the whole feature image at the end of the encoder, or equivalently, the receptive field of the fully connected layer is the entire image. Therefore the 3D coordinates of any point depend on the whole image, meaning a texture or intensity change locally in the image can affect the entire prediction. Another limitation coming from the fully connected layer, as previously stated in 2.3.1, is that the input image resolution and the total number of 3D points are fixed by design, making the model rigid. These two shortcomings can be mitigated by dividing the image into smaller regions and having a dedicated neural network for each region to predict a local point cloud. The global point cloud would be the aggregation of the local point clouds, potentially **increasing the total number of points**. Another approach to **preserve spatial neighbourhood information** would be to consider a fully convolutional network, in that case, the total number of predicted points would be determined by the input image resolution.

Secondly, we observed that close objects in the scene were not finely estimated by our method. One explanation could be that the current architectures of our neural networks make our method unable to differentiate between coarse-scale point clouds for which we can have higher error tolerance, such as road or building surfaces and fine-scale point clouds which would account for possible obstacles on the road. A multi-scale estimation mechanism could also be added in order to enable that scale differentiation. One approach would be first to estimate a coarse scale point cloud, then use it to condition immediately finer scale estimations.

All the presented methods in this chapter make use of the context information contained in the images to perform their estimations, yet other physical depth cues captured by imaging sensors can be exploited. One such cue is the defocus blur, leading to the next chapter, where we will address the problem of single-image depth from defocus using deep learning techniques.

3 - Learning local depth estimation from a single image using defocus blur

Contents

In the previous chapter, we introduced several deep learning approaches using the context information from large-field-of-view images to solve the problem of monocular depth estimation. Yet, depth information can be encoded locally by the sensor.

The defocus blur is one such powerful clue, and associated Depth from Defocus (DFD) methods have a long history in computer vision and raise the question of blur estimation from an image for unknown scenes. Early methods use multiple acquisitions of the same scene for different focus [Pentland, 1987, Nayar and Nakagawa, 1994]. More recently, model-based methods for single image DFD, have been proposed relying on the specification of a scene prior, either analytical or statistical [Zhou et al., 2009, Trouvé et al., 2011]. Finally, learning-based methods have emerged [Martinello and Favaro, 2011, Haim et al., 2018, Buat et al., 2021], and following model-based methods [Levin et al., 2007, Trouvé et al., 2011] still tackle this problem as a classification approach using a predefined finite set of Point Spread Function (PSF) associated with a depth. Yet, the depth measure belongs to continuous space, making classification approaches working on a discrete space accountable for unavoidable quantisation error. In order to overcome this limitation, we propose in this chapter a regression approach for single image DFD using deep learning. We will refer to single image DFD as DFD throughout the manuscript.

Deep learning classification networks output a membership probability vector corresponding to each potential class. In a classification setting, the estimated depth is the depth label with the highest membership probability. However, training of such networks usually relies on the hard-assignment encoding of the true depth, *i.e.*, the corresponding target membership vector has a non-zero probability value only for the depth class closest to the true depth. Hence many depth values can be assigned to the same class. To alleviate the quantisation error and avoid the many-to-one mapping, we propose to use a soft-assignment encoding of the true depth, which provides a unique dispatch of the membership weights on adjacent depth classes.

We show that our method outperforms classification approaches as well as direct regressions, on structured and natural texture datasets using small patches. Finally, we train and test our method on real data from a recently published paper on active DFD for industrial surface inspection [Buat et al., 2021], surfaces on which a Random Binary (RB) pattern is projected.

Our method, described in Section 3.2, solely requires a training set of patch/value pairs, without any blur nor scene analytical model, nor additional information. Our method is able to process many image formats, such as grayscale, RGB, or RAW. The latter format is considered in our experiments as it preserves the blur information. Our method is validated in simulation on structured and natural scenes (Section 3.3.1) and in an experimental setting on real data from an active DFD experiment [Buat et al., 2021] (Section 3.3.2). In each case, we compare the estimation results using the proposed soft encoding with direct regression and hard encoding-based methods. We conclude and enumerate a set of perspectives in Section 3.4).

These works resulted in a communication to the French national conference GRETSI 2022 [Leroy et al., 2022b] and a journal article to Optica's Applied Optics [Leroy et al., 2022a].

3.1 . What is Depth from Defocus?

As its name suggests, Depth from Defocus (DFD) aims to estimate the depth of an object based on its apparent defocus blur in the image. For a conventional optical system, at fixed aperture size, and fixed focal length, the blur is linked only to the relative depth of the object to the system, as illustrated by Figure 3.1. The image of an object located in the focal plane appears sharp. If this object moves away from the focal plane, rays of light coming from the object no longer hit the sensor at a single point, and a circle of confusion grows bigger the further the object moves away, leading to a defocus blur. This evidence makes the depth estimation problem analogous to a blur estimation problem. Depth from Defocus methods aim to measure the blur and they use the aforementioned equivalence to estimate the depth.

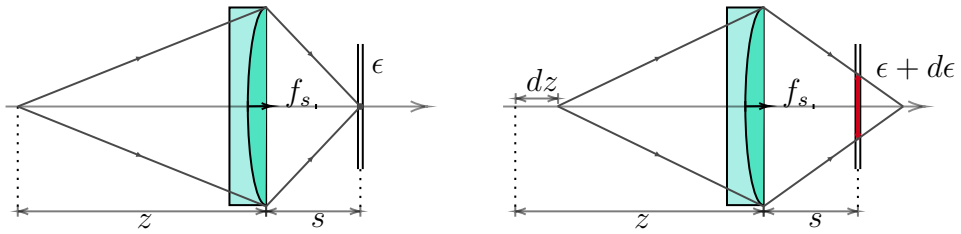


Figure 3.1 – Defocus blur using a thin lens model. For a sensor placed at a distance s after the lens, light rays are focused on a point located at a distance z before the lens (left). Moving the point at a distance dz away from z affects the radius ϵ of the circle of confusion (right).

3.1.1 . Related Works : Learning Depth from Defocus

Global Depth from Defocus with CNN

Several works have proposed to leverage defocus cues at a **global scale** for depth map prediction using deep learning [Carvalho et al., 2018a, Lee et al., 2019b, Shajkofci and Liebling, 2020, Ranftl et al., 2020, Anwar et al., 2021], even using unconventional optics such as phase mask or freeform lens to improve depth estimation [Haim et al., 2018, Chang and Wetzstein, 2019, Wu et al., 2019, Mel et al., 2022]. Besides, several learning methods for blur type classification or deblurring also extract an intermediate relative defocus map during the image processing [Zhang et al., 2018, Ma et al., 2022]. These methods are effective to estimate a relative defocus or depth maps from an image having a spatially varying defocus blur size, but the spatial variation of the Point Spread Function (PSF) due to optical aberrations is not taken into account. Besides, complex networks are involved, requiring a relatively large input, so they work on a **global scale**. Hence, a local depth prediction method seems to be more suited, especially for low-cost sensors having uncorrected optical aberrations.

Local Depth from Defocus by classification

In the single image case, common patch-based approaches consist of a selection of a blur within a finite set of potential blurs, using a selection criterion derived from maximum likelihood (ML) approaches [Trouvé et al., 2011, Buat et al., 2021, Zhu et al., 2013]. Recently, methods using supervised training of neural network models on image patches have been proposed for local blur parameter [Sun et al., 2015], or depth [Haim et al., 2018] classification. These DFD methods proceed by classification, while in practice real data involve continuous depth variation. This introduces a systematic estimation error due to the quantisation step. Reducing this step increases the computational cost for ML methods and reduces the number of examples for each class, implying convergence issues for learning methods with a given database. Finally, these classification approaches omit the existing neighbourhood relationship between depth classes [Fu et al., 2018].

Local Depth from Defocus by regression

Alternatively, methods for blur parameters or depth estimation from an image patch have also been proposed in the literature. For instance, [D'Andrès et al., 2016] use the vector of likelihoods obtained using a scene prior and blur model parameters as an input of a regression tree to regress the blur parameter values. [Yan and Shao, 2016] use a general regression neural network on a prefiltered patch version, after a blur type identification step. [Kashiwagi et al., 2019] use patch localisation as an attention map to benefit from the lens aberrations to regress the depth. In the work of [Shajkofci and Liebling, 2020], a Resnet is used directly to regress blur parameters using a relatively large patch size (typically 128×128). Direct depth regression from a small defocused patch, without any preprocessing, nor additional information is not trivial, especially due to the *regression to the mean* problem, as discussed in [Haim et al., 2018].

3.2 . From Classification to Regression

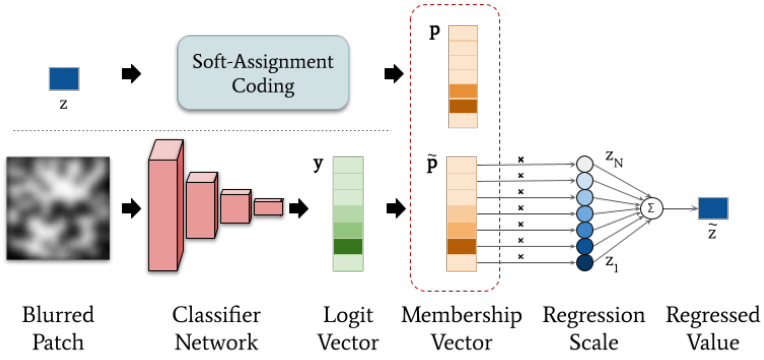


Figure 3.2 – Overview of the proposed method. A fully convolutional classifier estimates a logit vector from a blurred patch. A softmax operator is used to obtain the membership vector $\tilde{\mathbf{p}}$. The regressed depth value is obtained by linear combination of the membership vector $\tilde{\mathbf{p}}$ and a regression scale. The true depth is encoded into a target membership vector using soft-assignment.

We propose a depth regression network based on a simple depth classification model, as this model shows good training stability and performances in blur estimation [Haim et al., 2018, Sun et al., 2015]. Figure 3.2 illustrates the proposed method and architecture. A fully Convolutional Neural Networks (CNN) operates as the classifier network used in [Haim et al., 2018] and returns a logit vector $\tilde{\mathbf{y}}$, then a membership vector $\tilde{\mathbf{p}} = \{\tilde{p}_i\}_{i=1}^{N}$ is obtained by a softmax operator, N being the number of classes. To obtain regressed depth values, a linear layer, referred to as *regression scale*, is parameterised by $\mathbf{z} = \{z_i\}_{i=1}^{N}$

and is applied to $\tilde{\mathbf{p}}$ to yield our estimated value $\tilde{z}$.

In the following, we describe several approaches for training this architecture, and in particular the proposed soft-assignment encoding.

3.2.1 . Output Space Regression

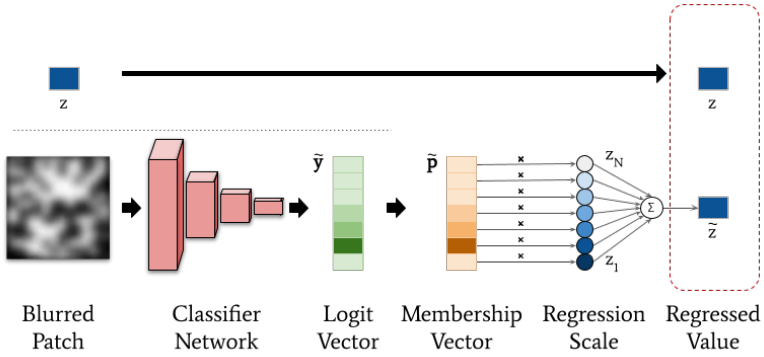


Figure 3.3 – Architecture of regression network from a classifier.

Supervised Deep Learning models for regression are usually trained by minimising a data fidelity term on the output of the network. A first attempt to regress the depth is to define a linear form that will convert the membership vector $\tilde{\mathbf{p}}$ coming from a classifier network into the depth scalar $\tilde{z}$. We named that linear form as the regression scale. As shown by Figure 3.3, the model parameters and the regression scale are then learned to minimise a L_2 loss directly on the true depth values expressed as

$$\mathcal{L}_{out} = (\tilde{\mathbf{p}}^T \mathbf{z} + b - z)^2 + \lambda_r \|\tilde{\mathbf{y}}\|_1, \tag{3.1}$$

where $\mathbf{z}$ and b are learned parameters. In this setting, $\mathcal{L}_{out}$ is invariant by permutations of $\tilde{\mathbf{p}}$ and $\mathbf{z}$ indices, generating multiple local minima and complicating the learning phase.

3.2.2 . Latent Space Regression

An auxiliary approach is to consider a data fidelity term over a latent space variable instead. The data fidelity term for classification approaches is usually a cross-entropy term over the softmax logit vector $\mathbf{p}$, defined as

$$\mathcal{L}_{CE} = -\sum_i p_i \log \tilde{p}_i. \tag{3.2}$$

This raises the question of the encoding, *i.e.*, the assignment of a target membership vector $\mathbf{p}$ to the true depth.

Hard-Assignment With this encoding, a true depth value z is assigned to only one class $j = \arg\min_i |z - z_i|$, or equivalently to a Dirac membership

probability vector $\mathbf{p}$ with $p_j = 1$ and $p_{i \neq j} = 0$, as depicted in Figure 3.5. An estimate of z is obtained with the $\arg\max$ operator : $\tilde{z} = z_{\arg\max_i(\tilde{\mathbf{p}}_i)}$. This coding scheme corresponds to usual classification approaches. A way to mitigate the $\arg\max$ classification results, including the misclassification error sensitivity, and obtain continuous depth values, is to use the soft-argmax operator defined as

$$\tilde{z} = \sum_{i=1}^{N} \tilde{p}_i \cdot z_i = \tilde{\mathbf{p}}^T \mathbf{z}. \tag{3.3}$$

Soft-Assignment Another approach is to guide the estimations using prior class relationships, as in the work of [Proença and Gao, 2020], where a soft-assignment encoding is used for pose estimation. Soft-assignment encodes a given depth value into a multi-class membership probability vector $\mathbf{p}$. This probability vector will be used as a weighting to enable precise decoding of intermediate values, hence a continuous estimation. We consider z_i as a depth landmark associated to class i. The class membership probability of a given sample at depth z is assigned using a kernel K with the classical rule [Liu et al., 2011]

$$p_i = K(z_i, z) / \sum_j K(z_j, z). \tag{3.4}$$

For a lossless decomposition, we use B-spline kernel of order 1 such as : $K(z_i, z) = [\delta - |z_i - z|]_+$, where δ is the distance between two consecutive landmarks. Figure 3.5 depicts the corresponding membership probability vector p of a true value z using 7 landmarks $\{z_i\}_{i=1}^{7}$. The estimated value $\tilde{z}$ is obtained via the soft-argmax operator : $\tilde{z} = \tilde{\mathbf{p}}^T \mathbf{z}$.

Table 3.1 shows a summary of the loss function and the estimation method for each of the above approaches.

3.2.3 . Naive Regression

The naive regression approach consists in considering an encoder network that ends with only one scalar instead of the classification logit vector and performing the optimisation minimising an L2 distance between that scalar and the actual value. There is no regression scale within this neural network architecture, as illustrated by Figure 3.4.

Table 3.1 – Summary of respective encoding, loss functions and estimation rules for each considered approach.

Method	Encoding	Loss function	Estimation
Output Regression	None	$(\tilde{z} - z)^2 + \lambda_r \lVert\tilde{\mathbf{y}}\rVert_1$	$\tilde{z} = \tilde{\mathbf{p}}^T \mathbf{z} + b$
Classification	Hard	$\mathcal{L}_{CE}(\tilde{\mathbf{p}}, \mathbf{p})$	$\tilde{z} = z_{\text{argmax}(\tilde{\mathbf{p}})}$
Hard-Assignment	Hard		$\tilde{z} = \tilde{\mathbf{p}}^T \mathbf{z}$
Soft-Assignment	Soft		$\tilde{z} = \tilde{\mathbf{p}}^T \mathbf{z}$

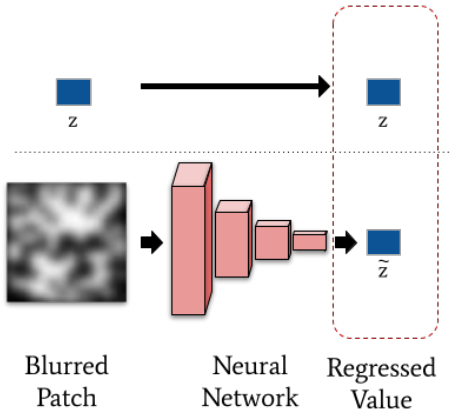


Figure 3.4 – Architecture of naive regression.

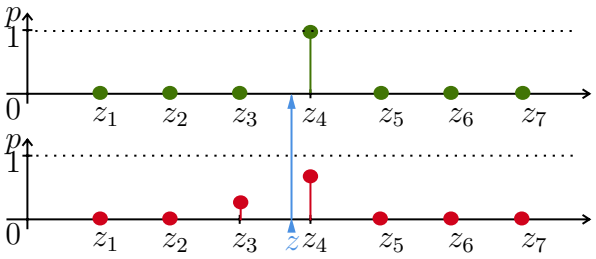


Figure 3.5 – Illustration of hard-assignment (top) and soft-assignment encoding (bottom) for a true depth value z (blue) on 7 classes. The corresponding hard-assignment code will result in a probability of 1 for the closest depth landmark, *i.e.*, z_4. Whereas the soft-assignment coding will dispatch the weights on adjacent depth landmarks, *i.e.*, z_3 and z_4.

3.2.4 . Network Architecture

For the classification network, we consider a simple CNN similar to the InnerNet proposed in [Haim et al., 2018] that is composed of 6 successive normalisation and convolution layers. It is designed to take a 32×32 grayscale image patch as input and to predict a scalar output. We provide details of the neural network architecture in Table 3.2.

Image normalisation The first layer of the network is a normalisation of the image pixels, according to the image format, to process images independently from their light exposure. The operation is a channel-wise operation for grey-scale and RGB images. When processing RAW images, we consider a normalisation operation of the image that operates independently for each colour channel by taking into account the particular Bayer RGGB layout. First, the first-order and second-order statistical moments, respectively μ_C and σ_C, where C denotes the colour, are computed for each colour channel indepen-

dently over the whole patch. Then, each pixel is normalised using the usual rule $p'_{ij} = (p_{ij} - \mu_C)/\sigma_C$. This operator ensures every pixel shows the same intensity dynamic.

Optimisation : The optimisation is done at a learning rate $\gamma = 1e{-}3$ using Adam with the moments $\beta_1 = 0.9$ and $\beta_2 = 0.999$.

Table 3.2 – Description of the neural network used in all the experiments. It is a simple CNN without skip connection inspired by the InnerNet from [Haim et al., 2018]. N is the number of classes

#	Layer	patch shape	kernel (stride)	feature space input	feature space output	padding
1	Image Normalisation	32×32	-			
2	Conv2D	32×32	9×9 (2)	1	64	4
3	BatchNorm2D + ReLU	16×16	-			
4	Conv2D	16×16	5×5 (2)	64	64	2
5	BatchNorm2D + ReLU	8×8	-			
6	Conv2D	8×8	5×5 (2)	64	64	2
7	BatchNorm2D + ReLU	4×4	-			
8	Conv2D	4×4	5×5 (2)	64	64	2
9	BatchNorm2D + ReLU	2×2	-			
10	Conv2D	2×2	5×5 (2)	64	64	2
11	BatchNorm2D + ReLU	1×1	-			
12	Dropout2D (p=0.2)	1×1	-			
13	Conv2D	1×1	1×1 (1)	64	N	0
14	Softmax	1×1	-	N	N	
15	Conv2D	1×1	1×1 (1)	N	1	0

3.3 . Experiments

In this section, we detail the various experimental settings we considered to challenge and characterise the proposed method described in Section 3.2.

As stated in Section 3.1, the DFD problem could be solved by a blur measurement approach under the assumption of known PSF calibration. For this reason, we tested first our method in a Gaussian blur measurement task using simulated data. Then we extended our test approach to a real DFD using experimental depth-annotated data.

3.3.1 . Blur Estimation on Simulated Data

In order to demonstrate the ability of our method to discern blurs, we first build a simulated dataset. To model the defocus blur, we use a Gaussian model which is a standard approximation used in the literature on DFD. Hence, the goal here is to estimate the standard deviation of a patch blurred with an isotropic Gaussian PSF. Each training example consists of a simulated blurred

and noisy sharp image patch and the corresponding Gaussian standard deviation σ in pixel. We consider two datasets of sharp images for training. The first dataset is a set of Simulated Random Binary (S-RB) texture images with 50% probability used in [Buat et al., 2021] for active chromatic DFD. This dataset comprises 10000 different patterns, 7500 are used for training and 2500 for testing. The second dataset is the Describable Texture Dataset [Cimpoi et al., 2014] (DTD), a collection of natural and artificial texture images, distributed equally across 48 texture categories. Homogeneous patches have been filtered out as they would be insensitive to blur. The filtering results in a 5640 image dataset with 4512 training and 1128 testing examples. The models are trained to regress 70 uniformly spaced blur sizes, *i.e.*, the standard deviation for the 2D Gaussian PSF, from 0.4 pixel to 3.0 pixels. Figure 3.6 shows examples of blurred patches for both datasets.

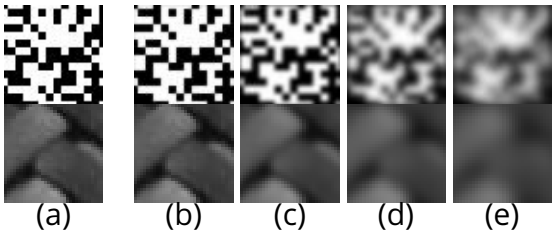


Figure 3.6 – (a) Image patches for the S-RB pattern (top) and DTD (bottom) datasets, and corresponding blur standard deviation σ : (b) 0.4, (c) 1.0, (d) 2.0, and (e) 3.0 pixels.

Table 3.3 shows performance metrics for the best model of the methods described in Table 3.1, trained on both datasets. We chose to encode the actual depth over a very low number of classes ($N = 7$). We also test the use of the ordinal loss [Fu et al., 2018]. It only implies increasing the size of the membership vector by 2, as this loss characterises the probability for the depth to lie between two ordered depth classes. The output space regression approach seems to be ill-conditioned and leads to a model that predicts the expectation of the blur value over the training set, this phenomenon is referred to as *regression to the mean issue* in [Haim et al., 2018].

In comparison, assuming the ordinality of depths as in [Fu et al., 2018] leads to significantly better performances. The classification approach highly improves the depth estimation score, with another slight improvement by adding a soft-argmax operation. Finally, our method based on soft-assignment encoding clearly outperforms the other regression approaches on both datasets. Figure 3.7 shows for both datasets and at each tested blur value σ the mean estimated blur value and the corresponding standard deviation (a) and bias (b), for the classification, the hard-assignment, and our method using soft-assignment. For the S-RB dataset, a quantisation of the estimation

Table 3.3 – Absolute and relative errors (RMSE and MAE) results on S-RB pattern and DTD datasets for several state-of-the-art methods and the proposed soft-assignement method. * indicates that only the local scale classification architecture of [Haim et al., 2018] is considered. All methods are trained using $N = 7$ classes.

Method	RMSE	MAE	RMSE	MAE
S-RB dataset	(in pix)		relative (in %)	
Output Regression	0.76	0.65	8.9	58.2
Ordinal	0.35	0.19	6.9	24.9
Classification *[Haim et al., 2018]	0.13	0.11	1.1	8.3
Hard-Assignment	0.12	0.10	1.1	7.9
Soft-Assignment	**0.01**	**0.01**	**0.01**	**0.6**
Naive Regression	0.02	0.01	0.1	0.8
DTD dataset				
Output Regression	0.76	0.65	8.9	58.2
Ordinal	0.34	0.24	4.4	20.8
Classification *[Haim et al., 2018]	0.3	0.23	2.47	17.12
Hard-Assignment	0.26	0.19	2.1	14.41
Soft-Assignment	**0.23**	**0.18**	**1.94**	**13.1**
Naive Regression	0.29	0.22	2.5	17.0

is clearly visible for both classification and hard-assignment approaches as well as a dispersion of the estimations near the borders of representation classes. Whereas our approach fits closely the identity line. In comparison, the greater diversity in texture and the potential native blur in DTD images cause greater dispersion in the estimations. While the overall error is greater for all, our approach performs better than hard-assignment and classification approaches. Misclassifications induce a bias that is necessarily positive (resp. negative) near the lowest (resp. highest) values of σ, making the estimation curve tilted as the estimated value can only be higher (resp. lower) in average.

The gap in performance between the S-RB and DTD datasets comes from the different nature of texture. RB images are built from simulation using a constrained scene model, and hence have high contrast, while DTD images have more variability in contrast and texture, making the training more challenging.

Regarding the naive regression, this approach performs almost as well as the soft-assignment on the S-RB dataset and slightly worse on the DTD dataset. This highlight the weaker robustness of the naive approach to greater texture variability.

3.3.2 . Depth Estimation on Real Data

The goal of this section is to demonstrate the ability of our method to estimate the depth of real data. For this experiment, we use a dataset of real

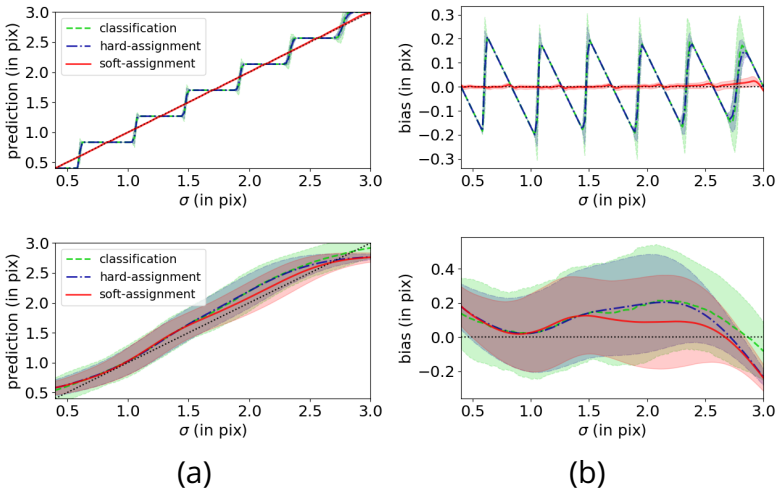


Figure 3.7 – Mean predicted blur value σ (a), bias (b) and confidence interval for classification, hard-assignment and soft-assignment approaches trained on the S-RB patterns (top) or DTD (bottom).

images used for industrial surface inspection using active DFD [Buat et al., 2021]. This dataset is an image collection of an RB pattern projected on a flat surface that sweeps a distance range going from 300 mm to 350 mm with a step of 0.2 mm, as illustrated in Figure 3.8. We refer to this dataset as Real Random Binary (R-RB). The camera has a focal length f=25 mm, with an aperture of $f/4$, and axial chromatic aberration characterised by a difference of 200 μm between the focal length for the red and for the blue channels. The chromatic aberrations separate the respective in-focus planes of the RGB channels to increase the depth cues encoded by the defocus blur in the image. As the three colour channels have different blurs, we process the RAW images to avoid any blur perturbation due to demosaicing. The experimental data exhibit strong off-axis aberrations, therefore we divide the full image into smaller rectangular regions where the PSF shape is assumed to be shift-invariant. We chose to use a regression scale with 15 depth classes linearly spaced between 300 and 350mm.

Table 3.4 shows the performances of our method on the central region of a 3 by 3 division, compared to hard-assignment and classification methods, the method proposed by Buat *et al.*, [Buat et al., 2021] for 51 linearly spaced depth classes, and the naive regression. We also tested our method using 51 classes. Our method achieves significantly smaller estimation errors compared to other methods supporting the benefit of using a soft assignment for depth regression. Using either 15 or 51 classes lead to similar performances, with a slight improvement for 51 classes. This slight benefit is not worth the added cost of training our method with 51 classes.

The naive regression approach shows a significantly large estimation error

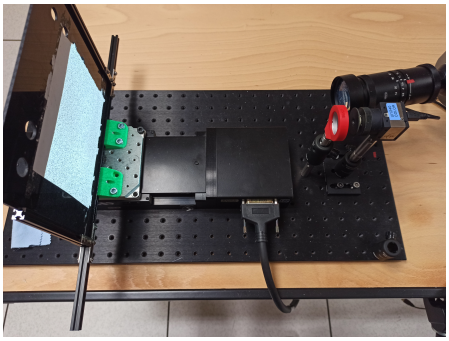


Figure 3.8 – Top view of the R-RB dataset acquisition setup [Buat et al., 2021]. On the right, the camera is mounted with a chromatic add-on, next to the pattern projector. On the left, the fronto-parallel screen that can translate along the optical axis and on which the RB pattern is projected.

due to a folding of the estimations near the edge of the depth domain, and also because of a greater bias in the estimations. These poor performances confirm the high sensitivity of this approach to slight perturbations, such as texture diversity or spatial variation of the PSF within the central region of experimental images.

Figure 3.9 shows the bias, standard deviation of estimations per depth and average bias for our method and that of Buat *et al.*, [Buat et al., 2021] trained and tested on the same region. Both methods exhibit a small estimation bias, however, our method shows less deviation in the estimations and an average bias closer to 0 mm.

Table 3.4 – RMSE and MAE, absolute and relative for classification (subpart of [Haim et al., 2018]), the proposed soft-assignment approach and Buat *et al.*, [Buat et al., 2021] approach trained on R-RB dataset, near the optical center.

Method (# classes)	RMSE (in mm)	MAE (in mm)	RMSE relative (in %)	MAE relative (in %)
Classification *[Haim et al., 2018] (15)	1.17	9.8 e-1	3.6 e-2	3.0 e-1
Hard-Assignment (15)	7.5 e-1	6.0 e-1	2.3 e-2	1.9 e-1
Soft-Assignment (15)	**4.7 e-1**	**3.7 e-1**	**1.5 e-2**	**1.2 e-1**
Soft-Assign. (51)	**4.6 e-1**	**3.7 e-1**	**1.4 e-2**	**1.1 e-1**
Buat *et al.*, [Buat et al., 2021] (51)	9.7 e-1	5.7 e-1	2.9 e-2	1.7 e-1
Naive Regression	3.90	1.65	1.3e-1	5.1e-1

Figure 3.11 shows depth map estimation using the method of [Buat et al., 2021] and ours on large RAW images for two 3D printed objects. In order to take into account optical aberrations, as in [Buat et al., 2020], we consider image subdivisions, using respectively $4\times$ 4 and 5×5 overlapping grids, as

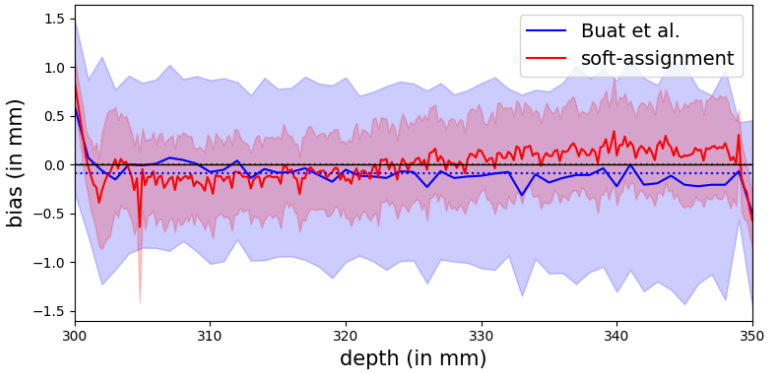


Figure 3.9 – Bias with estimation standard deviation per depth, average bias (dotted) for the method of Buat *et al.*, [Buat et al., 2021] and our proposed soft-assignment method trained on R-RB dataset.

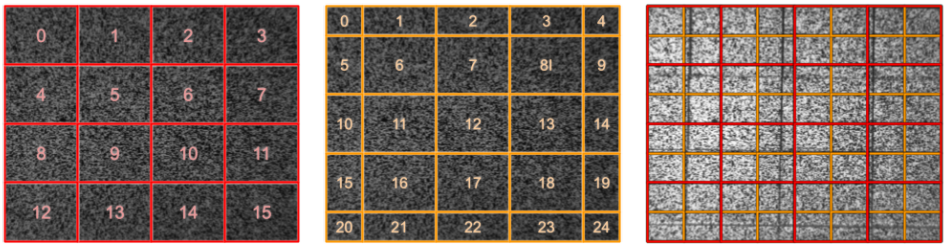


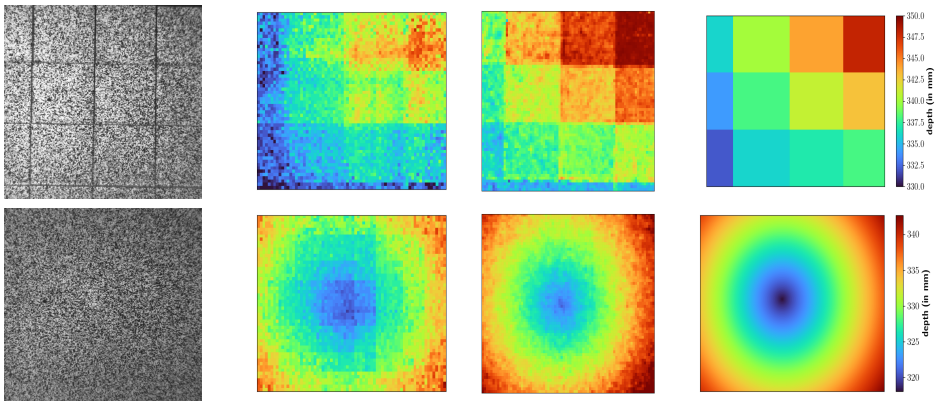
Figure 3.10 – Overlay of real data training data with (a) $4\times$ 4 division, and (b) 5×5 division. The final estimation is the fusion of overlapping subdivisions estimation. (c) Test object data with the fusion overlay.

shown in Figure 3.10. The covariance matrices learned for the method of Buat *et al.*, [Buat et al., 2021] and the proposed architecture are trained separately using 32×32 patches from each subdivision, so 41 different models are trained. Hence as each image patch belongs to two subdivisions, we compute the mean of the two depths regression obtained using the corresponding trained models. To produce the depth map, we process patches with overlapping of 50%.

The first object is a set of steps that is characterised by depth discontinuities, the second object is a cone that is characterised by a linear spatial variation of depth. A reference depth map is provided for each object. It consists of a 3D printing specification of the models completed with a single reference depth value measured with a telemeter on a characteristic point of the model [Buat et al., 2021]. Estimated depth maps are very similar to the reference depth maps on the whole image for both methods.

The method of Buat *et al.*, produces granular depth maps with visible bor-

ders of training subdivisions, especially on the cone example. On the contrary, the depth map produced by our method is smoother and sharper at the edges of each step, while circular depth levels are clearly visible on the cone depth map. These results highlight the efficacy of the proposed method, and its robustness in particular to patches showing either continuous or discontinuous depths values.



input RAW [Buat et al., 2021] ours reference depth map

Figure 3.11 – Test of the proposed method on RAW images of 3D printed objects : steps (top) showing depth discontinuity, and cone (bottom) showing linear variation of depth. From left to right : input RAW image, estimated depth map for the method of Buat *et al.*, [Buat et al., 2021], our proposed method, and the reference depth map obtained with a telemeter and known 3D object profile.

3.4 . Conclusion

In this chapter, we tackled the problem of Depth from Defocus using deep learning. One major drawback of state-of-the-art DFD methods is that they proceed using a classification approach, which does not take into account the continuous nature of depth.

In order to account for the continuous nature of depth, we proposed to explore methods for **depth regression on small patches** showing defocus blur. We especially developed a regression approach that takes advantage of a simple and stable classification architecture and a linear operator, named regression scale. The particularity of our method is that the training process lies in the **soft-assignment encoding** of the ground truth depth value to build a targeted membership probability vector that will be ultimately estimated by the network. The regression is then performed using the regression scale on the estimated membership probability vector.

Our method is simple, requires no image prior nor additional information, and can be applied to any image/value matching data. We applied our approach to the DFD problem and showed it performs well on simulated and experimental data for small patches, on both planar scenes and natural 3D objects.

Moreover, our method could extend to a more generic blur characterisation problem, in order to estimate the blur parameter as in [Debarnot and Weiss, 2022]. Further works will follow including the analysis of various experimental settings, such as the robustness to different noise levels and the patch size. The current approach for processing full-resolution images and taking into account the spatial variation of the defocus blur relies on learning a model locally dedicated to a small image portion. This approach can become time and memory expensive. A way to mitigate this limitation would be to inject the patch location into the network to regularise the estimation as in [Kashiwagi et al., 2019].

Previously presented methods all rely on the defocus blur that originates from the optical system and the imaging settings, such as the focus and the aperture size. One can wonder if an optimal optical setting exists for the DFD task. In the next chapter, we will develop a co-design method to solve the joint optimisation of both the optical system and the neural network parameters for the DFD task.

4 - Deep Co-Design for depth from defocus and depth of field extension

Contents

In the previous chapter, we showed how a neural network could regress locally the depth using the information brought by defocusing blur. As the defocus blur is closely related to the optical parameters, the question about the choice of these parameters for the most depth-informative blur arises. In other terms, is it possible to learn the neural network and the optical system parameters jointly for DFD ? Moreover, as the DFD deteriorates the image quality, can we pair up the image deblurring and the DFD is one single co-design framework ?

The joint optimisation of optics and image processing is a paradigm that was first introduced by [Dowski and Cathey, 1995] with the design of a cubic-phase-modulation phase mask for Extended Depth of Field (EDOF). This was followed with the works of [Diaz et al., 2009, Falcón et al., 2017, Fontbonne et al., 2019, Lévêque et al., 2020] that consider an MSE criterion over the deblurred image from a Wiener filter for designing phase masks. Co-design for the problem of DFD has also been addressed with [Levin et al., 2007], proposing to use a Kullback-Leibler divergence score to discriminate potential blurs in the frequency domain on a constrained set of binary coded aperture. [Martinello and Favaro, 2011] learns projections of simulated blurs onto orthogonal feature spaces to help blur identification according to a coded aperture. From this approach, a suitable binary-coded mask can be chosen by maximising the discriminating power for the learned feature space. [Trouvé et al., 2013, Trouvé-Peloux et al., 2014, Buat et al., 2022] propose to use depth estimators derived from maximum likelihood, and use the Cramér-Rao Bound to

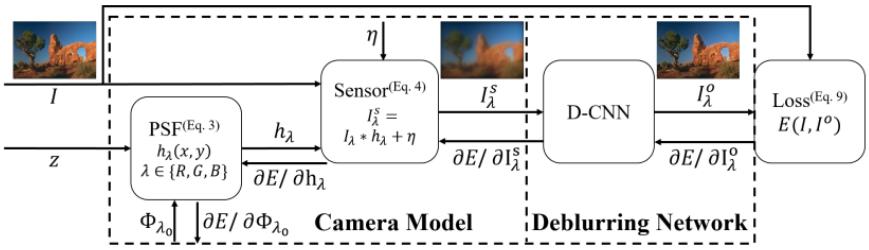


Figure 4.1 – Schematics of the deep co-design from [Akpinar et al., 2019] for image deblurring. The neural network (D-CNN) processes the image resulting from the differentiable camera model and provides an estimation of the deblurred image. A reconstruction error is computed, and the gradient of the error is propagated through the network to the camera model to update their parameters.

estimate a theoretical depth estimation accuracy and find the best parameters for conventional and unconventional optics.

All these methods aim to find a suitable optical system with respect to analytically defined performance and estimation models, such as Wiener filters or maximum likelihood. However, neural networks offer more degrees of freedom regarding image processing tasks. Therefore we address in the chapter the joint optimisation of neural networks and optical parameters.

4.1 . Related works

The deep co-design approach makes use of deep learning tools to jointly optimise the optical and neural network parameters for a given task. The neural network process the image resulting from the differentiable image formation model and provides an estimate for the given task. Then, an associated loss function is evaluated, and the gradient of the loss is propagated through the network to the image formation model to update their parameters through a gradient-based optimisation scheme. Figure 4.1 illustrates one instance of deep co-design framework for EDOF [Akpinar et al., 2019]. Because the images are processed by a neural network, a great variety of tasks can be addressed.

For instance, regarding optical components optimisation for the MDE, [Haim et al., 2018] proposed to learn an annular phase-coded mask and a simple fully-convolutional classification neural network jointly. [Wu et al., 2019, Chang and Wetzstein, 2019] both proposed to co-optimise the height profile of a phase mask and the parameters of a U-Net to regress depth maps. Similarly, [Mel et al., 2022] proposed a several-step training scheme that jointly optimises a phase-coded mask and a U-Net for depth estimation first, then trains

and fine-tunes the depth estimation model and a deblurring model for the given optimal phase mask. [Sitzmann et al., 2018, Elmalem et al., 2018, Akpinar et al., 2019] addressed depth of field extension.

One challenge of the deep co-design is to define the differentiable imaging formation model that will enable the optimisation of the optical parameters. Each of these works aims to design an optimal phase mask for their respective task and use a differentiable model of Fourier optics to form the images. This model assumes a paraxial thin-lens approximation for a simplified lens system. Differential Ray-Tracing (DRT) is another image formation model that requires the lens system specification, *i.e.*, number of surfaces, curvature and layout. This model can lead to the fine design of complex optical systems. This model has been used to optimise the focus distance of a known optics in [Halé et al., 2021] and for the optimisation of a complete optical system in [Sun et al., 2021] for EDOF. A greater computational cost accompanies the DRT, compared to Fourier optics, as it takes into account the complete optical design. [Dufraisse et al., 2022] compared both models for co-design perspectives, and observed that a Gaussian approximation on the DRT model granted a good compromise between computational cost and accuracy of the PSF.

We summarise all the aforementioned methods into Table 4.1, detailing the task considered for optimising the optical element, as well as the image formation model used for the optimisation. We can notice a great number of methods tackling phase mask optimisation using a Fourier optics model, as well as the emergence of methods dealing with DRT models instead. All these methods consider the optimisation of the optical system for a single task. The first take on deep co-design for both DFD and EDOF has been proposed in [Ikoma et al., 2021], where a phase mask is optimised with a neural network estimating RGB-D images.

In the following, we propose to address the multi-objective deep co-design of one particular optical parameter, that is the focus distance, using a DRT model. We also investigate the impact of the initialisation on the the resulting optical system, which, except from [Fontbonne, 2021], has a limited number of comments in the deep co-design literature.

4.2 . Chapter organisation

In this chapter, we explore the joint optimisation of a ray-tracing optical model and two neural networks. The first neural network deals with the DFD, and the second with the deblurring for image quality recovery over a predefined depth range, for the EDOF task.

We give more details about the considered optical system, the optical model, and the neural networks in Section 4.3. First, we address the deep co-design for each task independently to understand how the optimisation af-

Table 4.1 – Summary of deep co-design methods showing their respective task of interest, the image formation model used and the optical component that is optimised. FO : Fourier Optics, DRT : Differential ray-tracer.

Method	Task	Image formation	Optimised component
[Sitzmann et al., 2018]	EDOF/super-resolution	FO	Phase mask
[Elmalem et al., 2018]	EDOF	FO	Phase mask
[Haim et al., 2018]	MDE	FO	Phase mask
[Wu et al., 2019]	MDE	FO	Phase mask
[Chang and Wetzstein, 2019]	MDE	FO	Phase mask
[Akpinar et al., 2019]	EDOF	FO	Phase mask
[Halé et al., 2021]	EDOF	DRT	Focus
[Sun et al., 2021]	EDOF	DRT	Lens design
[Ikoma et al., 2021]	MDE +EDOF	FO	Phase mask
[Mel et al., 2022]	MDE (EDOF)	FO	Phase mask

fects the optical system, and if a unique optical system that would be satisfactory for both tasks exists. To test the uniqueness, we propose to look at the obtained optimal settings given various initialisation for an optical system with only one degree of freedom.

Then we consider the simultaneous optimisation of both neural networks with a common optical system. See Figure 4.2 for an overview of the optimisation chain.

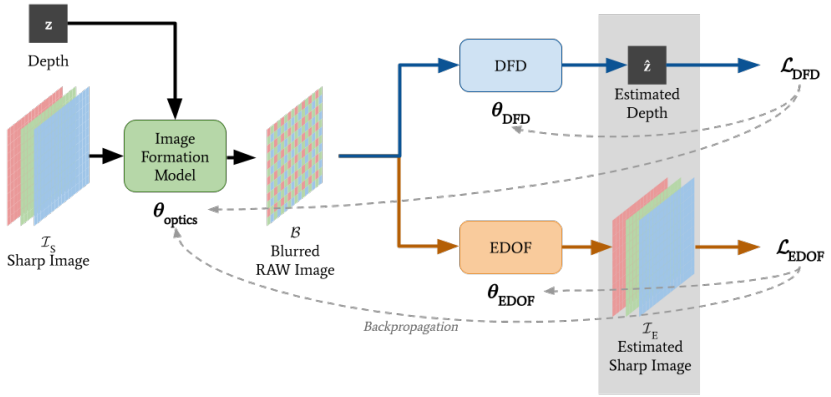
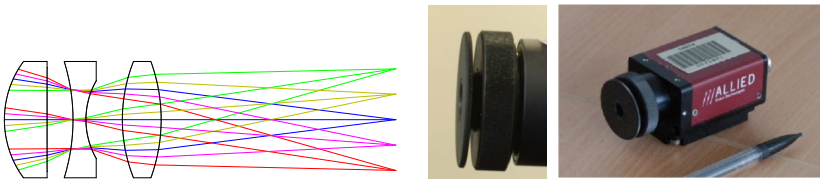


Figure 4.2 – Overview of the multi-task co-design framework.

4.3 . Settings

In order to simplify the problem of the joint optimisation, we consider the optical model of an actual unconventional optical system that has been de-



(a) (b)

Figure 4.3 – (a) Schematics of the lens triplet, also known as the Cooke triplet, composing the optical system. (b) View of the camera mounted with the lens triplet. [Trouvé et al., 2013]

signed in [Trouvé et al., 2013] for DFD. This optics is characterised by a lens that has been built to display sufficient longitudinal chromatic aberration to have separated RGB depth of field in the required depth range. This design was also constrained to reduce all other aberrations as much as possible in order to maintain good image quality. The design starting point was a classical Cooke 25 mm f/4 triplet made of two convergent lenses separated by a divergent lens, which is the triplet aperture stop because this configuration naturally helps to reduce lateral chromatic aberration. Several choices of glass for the triplet have been compared and it turned out that a focal shift of 200 µm, obtained with the glasses N-BK7/LLF1/N-BK7, was an amount of chromatic aberration that correctly separated the RGB depth of field in the depth range of 1 to 5 m, which was the original depth range of interest in this reference for robotic application. As the lens specifications are known, we can provide them to a DRT model, which enables the deep co-design of that particular optics. Here, as in [Buat et al., 2021], this lens is used in a closer depth range from 300mm to 350mm. This optical system has only one degree of freedom left, which is the focus distance, on which we perform the joint optimisation.

4.3.1 . Image formation model

For our experiments, we consider the following image formation model for patches

$$\mathcal{B} = Bayer(\mathcal{I}_S \star H_{\theta_{optics}}(z)) + \eta \tag{4.1}$$

Where $\star$ is the convolution operator, $\mathcal{I}_S$ an RGB sharp image, $H_{\theta_{optics}}$ the chromatic PSF as a function of depth, and η a Gaussian noise. The Bayer operator converts a 3-channel RGB image into a 1-channel RGGB Bayer image of the same resolution that accounts for the actual camera sensor. θ_{optics} is the single optical parameter to optimise, in our case, the focus distance.

We chose to estimate the PSFs using the differentiable ray-tracing model developed in [Volatier et al., 2017] and also used in [Halé et al., 2021], as it accounts for the real optical system layout, and can simulate the spatial variation of the PSF and off-axis aberrations. The computation of the PSF is done by casting from a punctual source numerous rays in different directions that

will propagate through the lens system according to reflection laws until they reach the sensor. This results in a spot diagram

$$\mathcal{S} = \{s_i \in \mathbb{R}^2, i \in [\![1, N_{rays}]\!]\} \tag{4.2}$$

, that accounts for every possible location s_i where a ray hit the sensor, *i.e.*, a density map from which a PSF can be inferred. We chose to use the Gaussian approximation of the model for the PSF, as it offers a good trade-off between computational cost and accuracy, as discussed in [Dufraisse et al., 2022]. Hence, the shape of the PSF is given by the standard deviation obtained from the spot diagram. Even though DRT handles off-axis aberrations, we decide for our preliminary study to consider an optimisation setting near the axis.

4.3.2 . Depth estimation method

To perform the DFD estimation, we use the method previously described in Chapter 3, with the architecture presented in Section 3.2.4 with 15 classes for the encoding, and the associated loss function $\mathcal{L}_{DFD}$ is the cross-entropy loss between the true and predicted membership probability vectors, respectively $\mathbf{p}$ and $\tilde{\mathbf{p}}$.

$$\mathcal{L}_{DFD} = -\sum_{i=1}^{N} p_i \log(\tilde{p}_i), \tag{4.3}$$

with :

$$\tilde{p}_i = \Psi_{\theta_{DFD}}(\mathcal{B})|_i = P(z = z_i|\mathcal{B}, \theta_{DFD}). \tag{4.4}$$

The estimated depth is obtained using the following linear form :

$$\hat{z} = \sum_{i=1}^{N} \tilde{p}_i z_i. \tag{4.5}$$

Here the set of parameters to optimise is θ_{DFD} of 400K parameters, which is a relatively small number of parameters compared to other neural network architectures for depth estimation. We find it important to have as few processing parameters as possible not to undermine the depth cues provided by the optical model.

4.3.3 . Deblurring method

In order to perform the deblurring for the EDOF task, we consider a neural network architecture inspired by the Residual Encoder-Decoder network (RED-Net) [Mao et al., 2016]. This architecture comprises a sequence of convolution layers (encoder) followed by the same number of "deconvolution" layers (decoder) and makes use of periodic additive skip connections between the encoder and decoder at equivalent scales. We are interested to retrieve an

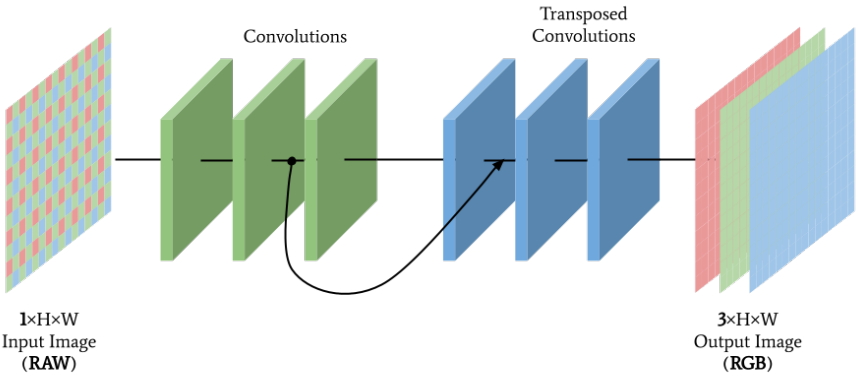


Figure 4.4 – Adaptation of the RED-Net for deblurring RAW patches. The network performs the demosaicing and outputs an RGB image.

RGB image $\mathcal{I}_E$ from the RAW image $\mathcal{B}$. So, in addition to deblurring, the network performs demosaicing. The architecture of the neural network is illustrated in Figure 4.4.

$$\mathcal{I}_E = \Phi_{\theta_{EDOF}}(\mathcal{B}). \tag{4.6}$$

The loss function for this task is a simple variation of the ℓ_1 distance between $\mathcal{I}_E$ and $\mathcal{I}_S$, the true sharp patch image :

$$\mathcal{L}_{EDOF} = \ell_1(\mathcal{I}_E, \mathcal{I}_S) \times \sigma_S, \tag{4.7}$$

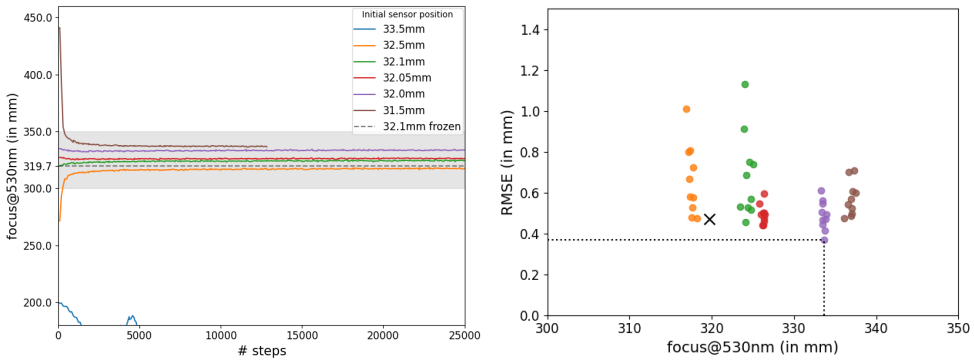
where σ_S is the standard deviation of the pixel intensities per channel of the ground truth sharp image. This weights down the patch examples having smooth textures and little contrast.

4.3.4 . Optimisation scheme

For the gradient-based optimisation, we considered the Adam optimiser over 80 epochs, with two different learning rates for the optical parameters and the neural network parameters. In the following experiments, we set $\eta_{optics} = 0.01$ and $\eta_{net} = 0.001$.

4.4 . Deep co-design for a single task

In this section, we address the optical and neural network joint optimisation for each task separately in order to understand what optical system suits best the task. As a reminder, the optical parameter we optimise is the focus plane distance for the green colour, corresponding to a wavelength of 530nm,



(a) (b)

Figure 4.5 – (a) Evolution of the focus during training for the DFD on RB patterns given various initial sensor positions. The shaded area shows the depth range of interest. (b) Scatter plot of the RMSE performances with respect to the focus plane distance for the last 10 epochs (over 80). The cross marks the best performance obtained for the optimisation with fixed optics at focus=319.7mm. The dotted lines indicate the RMSE and focus distance values for the best DFD system.

or equivalently, the distance of the sensor to the lens system. To address the sensitivity to the initialisation for the joint optimisation, we consider several initial sensor positions : 31.5, 32.0, 32.05, 32.1, 32.5, and 33.5mm.

We perform the optimisation and the testing on two texture datasets already described in Chapter 3 : RB pattern, and DTD. Each patch is associated with a depth value drawn randomly and uniformly between 300mm and 350mm, which defines the depth range of interest.

4.4.1 . Deep co-design of DFD on Random Binary patterns

We first consider the joint optimisation of the focus distance and the neural network parameters on RB patterns.

Figure 4.5.a shows the evolution of the focus during the joint optimisation for each initial sensor position. The shaded area defines the depth range of possible values for training and testing. Some initial sensor positions are chosen to make a focus out of this range : 33.5, 32.5, and 31.5mm.

We observe that, except for the initialisation at 33.5mm that doesn't converge, all the other models tend to quickly reach a stable focus distance inside the depth range. However, the optimal focus is different depending on the initial sensor position value. Besides, the ordering between the initial focus distances and between the optimised focus distances seems to be preserved. This might suggest that the optimisation landscape might contain several local minima for the sensor position.

In order to have a definite answer on which system, *i.e.*, focus, to consi-

der for our task, we characterise the performances of each potential system using the RMSE on a validation set. For each optimisation trajectory, we show the RMSE and corresponding focus for the last 10 epochs in Figure 4.5.b. Note that the trajectory that did not converge, corresponding to a sensor position initialised at 33.5mm, is out of the window because of its unsatisfactory estimation performances. We also display, using a cross mark, the best RMSE performance of a DFD network trained on image patches from an optical system focused and fixed at 319.7mm. The best performance achieved by this particular optimisation setting is similar to the other co-design approaches. Yet, it is important to mention that several settings obtained by optimising the focus distance offer lower RMSE. We can notice that the obtained performances for each trajectory are concentrated around distinct focus distances, which corresponds to the previously observed convergence in focus distance. However, for a given focus point, we can witness a dispersion in performances which results from the variability in the neural network parameters during training and we can evaluate a quantitative measure of these dispersions using the average and standard deviation. We report these measures for the final sensor position [and corresponding focus distance], the final loss, and the final RMSE in Table 4.2. From this table, we can note that these measures for the loss are similar for each trajectory, meaning they have reached a similar objective for different optical parameters.

Table 4.2 – Review for the joint optimisation of the sensor position and the DFD model for various initial sensor positions on RB patterns. Average value ($\pm$ standard deviation) over the 10 last epochs.

sensor position [focus @530nm] (in mm)		Loss	RMSE
initial	final	CE	(in mm)
33.50 [199.1]	34.92 [147.1 (**±3.4**)]	2.69 (±2.0e-2)	14.4 (**±6.6e-2**)
32.50 [271.4]	32.11 [317.5 (**±3.6e-1**)]	6.01e-1 (±5.9e-2)	6.64e-1 (**±1.63e-1**)
32.10 [319.7]	32.06 [324.4 (**±3.4e-1**)]	5.80e-1 (±4.3e-2)	6.81e-1 (**±2.00e-1**)
32.05 [327.1]	32.05 [326.2 (**±2.2e-1**)]	5.79e-1 (±4.4e-2)	4.94e-1 (**±4.5e-2**)
32.00 [334.8]	32.00 [333.5 (**±2.5e-1**)]	5.78e-1 (±3.8e-2)	4.88e-1 (**±6.8e-2**)
31.50 [441.1]	31.98 [336.8 (**±2.9e-1**)]	5.97e-1 (±4.4e-2)	5.70e-1 (**±7.9e-2**)

Table 4.2 reports marginal statistics for each optimisation trajectory, so the best focus setting, here 333.5mm, does not correspond to a system experimentally obtained during training. When choosing the most suitable optical parameters, we would rather base our decision on a system experimentally obtained. Therefore the decision is made based on Figure 4.5.b. Hence we would choose to focus at **333.7mm** for the depth estimation as it is the setting with the lowest known RMSE of 0.37mm, which is highlighted with the dotted lines. Note that the difference in performance with other settings is small.

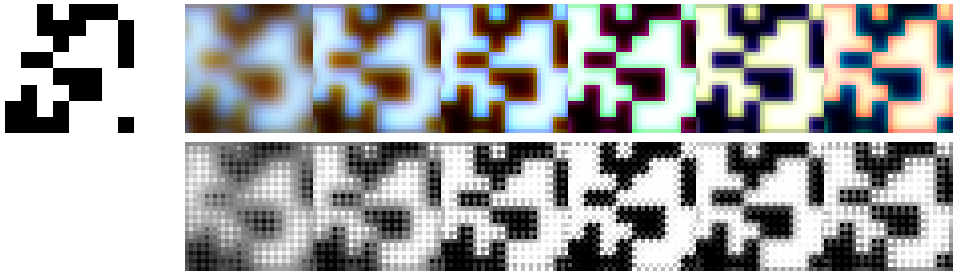


Figure 4.6 – Example of a simulated RB pattern at various depths with the optimal system : (top) RGB images to visualise the chromatic blur, and (bottom) RAW images that are processed by the neural network.

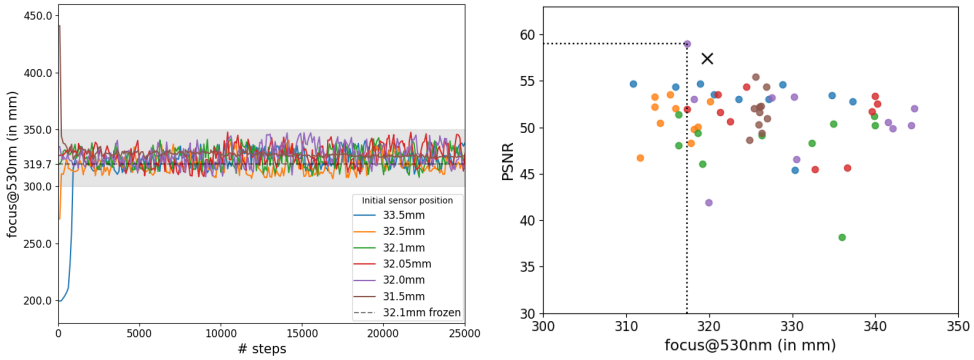
We also show some examples of a RB pattern acquired at various depths for the chosen setting in Figure 4.6. The neural network processes the RAW images, so we display the RGB images for visualisation of the chromatic blur.

4.4.2 . Deep co-design for EDOF on Random Binary patterns

Now we investigate the deep co-design for the problem of deblurring for a large depth range, namely EDOF. As before, we perform end-to-end optimisations of the deblurring neural network and the optical sensor position, for various optical initialisation and depth values.

Figure 4.7.a shows the evolution of the focus during training for each sensor position initialisation. The optimisation process guides every system to have a focus distance into the depth range of interest. However, in most cases, the sensor position suffers from significant variations inside the depth range of interest during the optimisation and it does not converge to a definite focus value. Only one trajectory initialised at 31.5mm shows convergence in sensor position during training. As for DFD, these different behaviours may account for the complex optimisation landscape. It would also suggest that the focus distance is not a determinant parameter for the EDOF network to be able to process the image as long as it is within the depth range of interest. We explain this insensitivity from the simplicity of the EDOF task for deblurring a RB patterns.

For the EDOF task, we consider the PSNR to assess the goodness of the estimations : the higher, the better. We show in Figure 4.7.b the PSNR and the focus during the last 10 epochs for each optimisation trajectory. Unlike for the DFD, the position of the sensor does not seem to be crucial for the EDOF. However, one sample stands out with a PSNR of 59.1 for a focus at **317.3mm**. Any other focus distance for the system may lead to a similar PSNR close to 55. We also observe, from the cross mark, that the optimisation of the EDOF neural network for a focus distance fixed at 319.7mm leads to a system with a high PSNR, due to greater stability in training for the simple task of RB pattern



(a) (b)

Figure 4.7 – (a) Evolution of the focus during training for the EDOF on RB patterns given various initial sensor positions. The shaded area shows the depth range of interest. (b) Scatter plot of the PSNR performances with respect to the focus plane distance for the last 10 epochs (over 80). The cross marks the best performance obtained for the optimisation with fixed optics at focus=319.7mm. The dotted lines indicate the PSNR and focus distance values for the best EDOF system.

deblurring. Yet, as for DFD, deep co-design discovered a better setting. Again, we report statistics over the last 10 epochs for the final sensor position, loss and PSNR in Table 4.3. We notice the great value of standard deviation for the final sensor position over the last 10 epochs.

Table 4.3 – Review for the joint optimisation of the sensor position and the EDOF model for various initial sensor positions. Average value (± standard deviation) over the 10 last epochs.

sensor position [focus] (in mm) initial	final	Loss L1	PSNR
33.50 [199.1]	32.05 [325.7 (±**7.3**)]	6.63e-4 (±1.03e-4)	53.0 (±2.6)
32.50 [271.4]	32.11 [317.0 (±**3.9**)]	8.63e-4 (±1.53e-4)	50.9 (±2.1)
32.10 [319.7]	32.06 [325.8 (±**8.7**)]	1.07e-3 (±2.16e-4)	48.2 (±3.6)
32.05 [327.1]	32.03 [329.8 (±**9.6**)]	9.16e-4 (±1.62e-4)	51.1 (±2.9)
32.00 [334.8]	32.05 [328.7 (±**9.6**)]	8.66e-4 (±1.54e-4)	51.0 (±4.2)
31.50 [441.1]	32.06 [326.1 (±**0.8**)]	9.64e-4 (±1.88e-4)	51.7 (±1.9)

Conclusion on Random Binary patterns We observed that optimisation for DFD is subject to local minima with similar performance in terms of RMSE, and that EDOF is not sensitive to the sensor position as long as the focus is within the depth range of interest. Moreover, the best estimation for both tasks, *i.e.*, 333.7mm for depth estimation and 317.3mm for deblurring

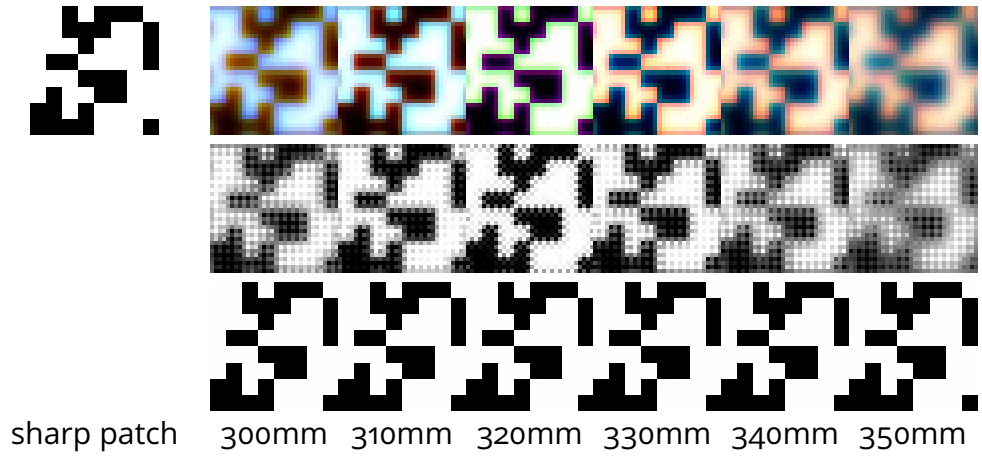


Figure 4.8 – Example of a simulated RB pattern at various depths with the optimal system for EDOF : (top) RGB, (middle) RAW, (bottom) deblurred.

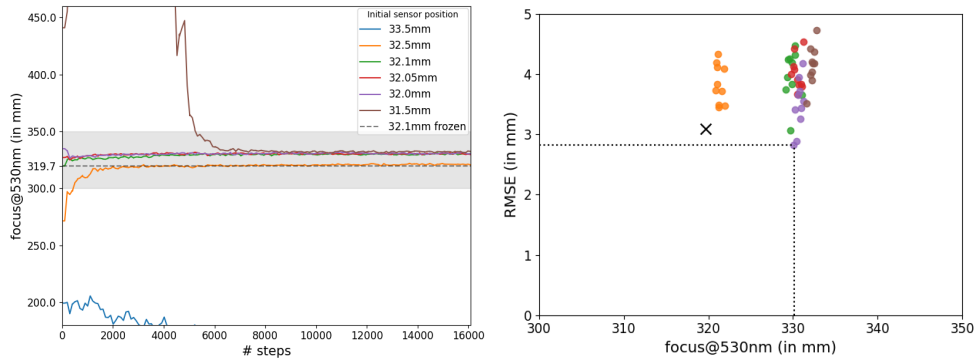
on RB patterns correspond to different settings. If the co-design approach helps to discover better focus settings for each task, the benefit in terms of performance is however limited in these experiments, possibly because of the deblurring task for RB patterns that seems too easy. For this reason, we consider another dataset in the following section.

4.4.3 . Deep co-design for DFD on natural images

We performed the optimisations for the DFD on the DTD used in Chapter 3, which contains a great variety of natural textures, hence, having lower contrast than binary patterns.

Figure 4.9.a shows the evolution of the focus given different initialisations. As with the RB dataset, the convergence of the focus is not assured when the initial focus is far from the depth range of interest, as shown by the blue curve and the late convergence of the brown curve. On this dataset, we can notice two optimal focus distances standing out : 321mm and 330mm. Compared to RB dataset, the greater variability in texture seems to smooth the optimisation landscape and reduce the number of local minima.

When looking at the RMSE for the depth estimations and their respective focus distance on Figure 4.9.b, we can see that each optimal focus distance displays overall similar performances with the lowest RMSE of 2.9mm obtained with a focus at **330mm**. We see that the optimisation with fixed focus, marked by the cross, is close to an optimal focus distance obtained by co-design, and offers a good estimation performance. However, as with the RB dataset, a better system is reached using deep co-design. We report the training statistics over the last 10 epochs in Table 4.4. Again, the final loss value is similar for each optimisation trajectory. We notice that, for DTD, the loss value is twice as high as for RB dataset, which indicates that natural textures



(a) (b)

Figure 4.9 – (a) Evolution of the focus during training for the DFD on DTD given various initial sensor positions. The shaded area shows the depth range of interest. (b) Scatter plot of the RMSE performances with respect to the focus plane distance for the last 10 epochs (over 80). The cross marks the best performance obtained for the optimisation with fixed optics.

are more difficult to process.

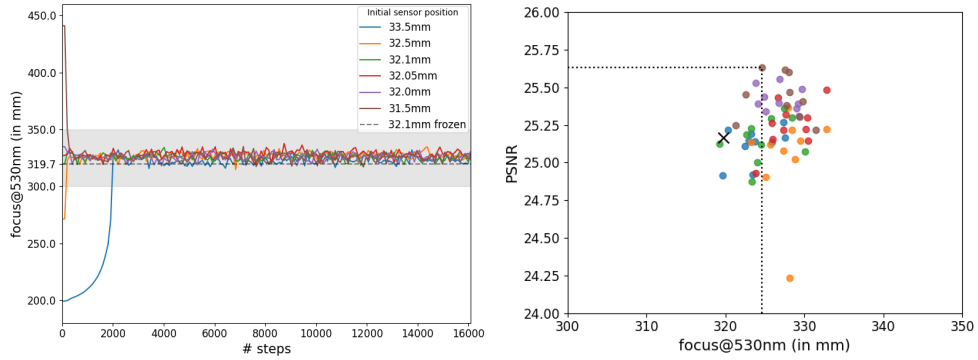
Table 4.4 – Review for the joint optimisation of the sensor position and the DFD model for various initial sensor positions on DTD. Average value (± standard deviation) over the 10 last epochs.

sensor position [focus @530nm] (in mm) initial	final	Loss CE	RMSE (in mm)
33.50 [199.1]	35.85 [126.8 (±1.83)]	2.69 (±2.0e-2)	14.5 (±1.5e-1)
32.50 [271.4]	32.09 [321.1 (±3.7e-1)]	1.00 (±1.1e-1)	3.85 (±3.1e-1)
32.10 [319.7]	32.03 [330.1 (±4.1e-1)]	9.67e-1 (±1.56e-1)	3.98 (±4.0e-1)
32.05 [327.1]	32.03 [330.4 (±4.8e-1)]	9.76e-1 (±9.9e-2)	4.02 (±2.7e-1)
32.00 [334.8]	32.03 [330.6 (±4.2e-1)]	9.56e-1 (±9.2e-2)	3.49 (±4.0e-1)
31.50 [441.1]	32.02 [332.1 (±4.4e-1)]	1.04 (±1.1e-1)	4.15 (±3.1e-1)

4.4.4 . Deep co-design for EDOF on natural images

Lastly, we consider the co-design of the sensor position for the EDOF. Figure 4.10.a shows the evolution of the focus during training given different initialisation. Regardless of the initialisation, the focus falls quickly into the depth range of interest. Yet, there is a noticeable variability in the focus distance during the optimisation process, which happens to be smaller than for RB patterns. This suggests that the greater diversity of texture provided by DTD is more challenging for the EDOF, hence the system is more sensitive to the sensor position.

We show the performance in PSNR and the respective focus distance in Figure 4.10.b. Each optimisation leads to systems with comparable perfor-



(a) (b)

Figure 4.10 – (a) Evolution of the focus during training for the EDOF on DTD given various initial sensor positions. The shaded area shows the depth range of interest. (b) Scatter plot of the PSNR performances with respect to the focus plane distance for the last 10 epochs (over 80). The cross marks the best performance obtained for the optimisation with fixed optics.

mances as long as the focus is between 317mm and 333mm, the best PSNR being 25.62 at 325mm. Again, we observe the benefit of the deep co-design regarding the best achievable PSNR compared to optimising the neural network with fixed focus distance.

We also report the statistics over the last 10 epochs in Table 4.5 and we can notice the smaller standard deviation of the final focus distance for training on DTD compared to RB dataset.

Table 4.5 – Review for the joint optimisation of the sensor position and the EDOF model for various initial sensor positions on DTD. Average value ($\pm$ standard deviation) over the 10 last epochs.

sensor position [focus] (in mm) initial	final	Loss L1	PSNR
33.50 [199.1]	32.07 [323.1 (±2.3)]	6.99e-3 (±1.08e-3)	25.1 (±1.1e-1)
32.50 [271.4]	32.05 [327.7 (±2.7)]	6.72e-3 (±1.05e-3)	25.0 (±2.9e-1)
32.10 [319.7]	32.05 [324.7 (±2.8)]	6.74e-3 (±9.42e-4)	25.2 (±1.4e-1)
32.05 [327.1]	32.05 [328.3 (±2.9)]	6.60e-3 (±9.51e-4)	25.2 (±1.5e-1)
32.00 [334.8]	32.05 [326.2 (±2.5)]	6.86e-3 (±1.02e-3)	25.4 (±7.7e-2)
31.50 [441.1]	32.06 [324.7 (±3.0)]	6.72e-3 (±9.55e-4)	25.4 (±1.4e-1)

Conclusion on natural texture We observed that training on natural texture, having less contrast than binary patterns, leads to fewer local minima for the DFD, and a higher sensitivity to the sensor position for the EDOF. Hence, the benefit of the deep co-design is more visible for natural textures.

4.4.5 . Conclusion on the single-task co-design

In this section, we managed to perform separate end-to-end optimisations of the parameters of a neural network for either DFD or EDOF, and of an image formation model in order to find the best focus for each task.

We noticed similar behaviour for each task regarding convergence and variability of the focus distance during the optimisation on either texture type. Optimisation for DFD must face multiple local optima but with similar performances, and EDOF is less sensitive to the focus distance.

Regarding the performances, we showed that deep co-design approaches can find systems having better performances than just training a neural network for an arbitrarily chosen and fixed optical system.

The optimisation process on natural texture led to advantageous focus distances for each task closer to each other compared to the optimisation on RB patterns. However, since there is no clear optimal focus distance for both tasks, a compromise must be made by hand. Hence, can we delegate the choice of the compromise to the optimiser?

4.5 . Multi-Task Co-Design

To answer that question we consider an end-to-end multi-task optimisation scheme, where both tasks are performed simultaneously using the same image, meaning they are linked with only one parameter : the sensor position. The joint optimisation is performed to minimise the following loss function :

$$\mathcal{L}_{multi} = \mathcal{L}_{DFD} + \lambda \mathcal{L}_{EDOF} \tag{4.8}$$

$\mathcal{L}_{DFD}$ and $\mathcal{L}_{EDOF}$ have a difference of magnitude by a factor 500, as shown in Tables 4.2, 4.5, 4.4 and 4.3. To prioritise the DFD, which is the task of interest for this thesis, We choose to set λ=1.

4.5.1 . Multi-task on Random Binary patterns

As for Section 4.4, we first perform the optimisation on RB patterns.

Figure 4.11 shows the evolution of the focus distance during the optimisation process for various initialisations. We first notice that for every initialisation, the focus falls and remains in the depth range of interest, which seems to be a benefit from the EDOF. Besides, 3 plateaus are reached : 318mm (blue and orange), 326mm (green and red), and 335mm (purple and brown), similarly to the DFD in single-task optimisation.

Regarding the performance in depth estimation, Figure 4.12.a shows the achieved RMSE and corresponding focus during the last 10 epochs of the multi-task optimisation on RB patterns, as well as the lowest RMSE reached by the single-task DFD training (dotted line). Overall reached performances are simi-

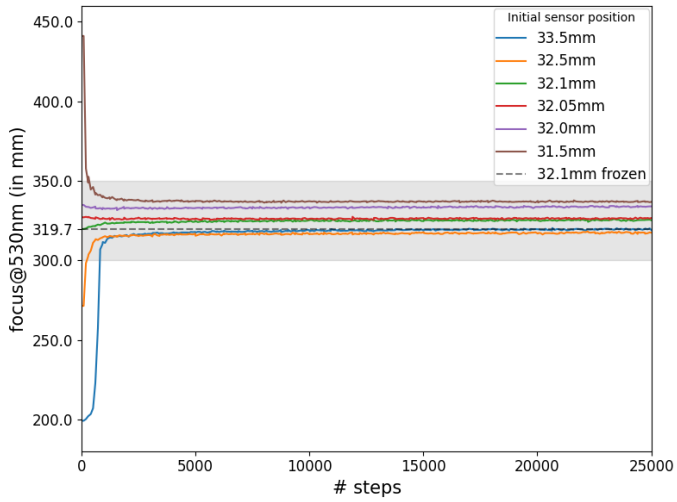


Figure 4.11 – Evolution of the focus during the multi-task training (DFD + EDOF) on RB patterns given various initial sensor positions. The shaded area shows the range of depth of interest.

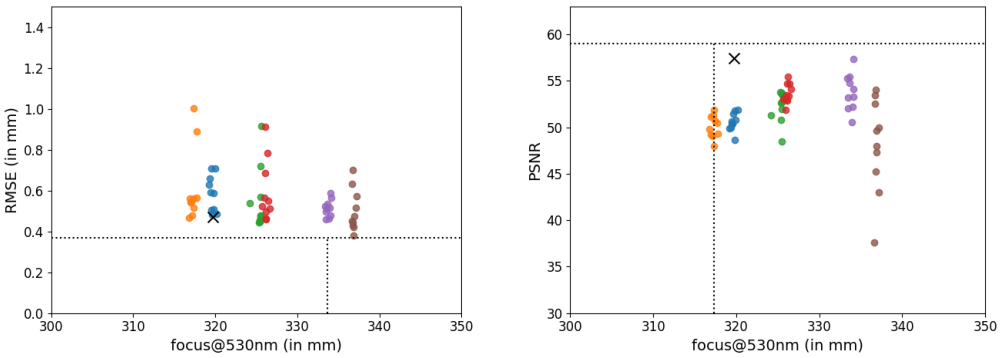
lar to single-task DFD, yet, the lowest RMSE value that is similar to the single-task optimisation is reached for a focus at 337mm instead of 330mm.

Regarding the performance in deblurring, Figure 4.12.b shows the PSNR of the deblurred image with respect to the focus for the multi-task optimisation, and the highest PSNR for the single-task optimisation (dotted line). The highest PSNR for this multi-task approach is slightly lower than for the single-task approach and is reached for a focus at 333mm instead of 317mm. The PSNR value is also really close to the PSNR obtained via fixed optics optimisation. This slight cost in performance for EDOF might be the result of the priority given to the DFD through the loss balancing.

We report the statistics over the last 10 epochs for the final sensor position, the loss for either task, PSNR, and RMSE in Table 4.6. We notice the loss for DFD is similar to the single-task optimisation, while the loss for EDOF is slightly higher due to the loss unbalance.

Table 4.6 – Review for the joint optimisation of the sensor position and the DFD+EDOF model for various initial sensor positions. Average value ($\pm$ standard deviation) over the 10 last epochs.

sensor position [focus] (in mm)		Loss		EDOF	DFD
initial	final	L1	CE	PSNR	RMSE (mm)
33.50 [199.1]	32.10 [319.5 (±3.2e-1)]	1.02e-3 (±2.63e-4)	5.89e-1 (±5.5e-2)	50.6 (±0.9)	5.86e-1 (±8.4e-2)
32.50 [271.4]	32.12 [317.4 (±3.7e-1)]	1.09e-3 (±1.58e-4)	6.00e-1 (±5.4e-2)	50.1 (±1.2)	6.12e-1 (±1.72e-1)
32.10 [319.7]	32.06 [325.3 (±3.2e-1)]	8.91e-4 (±1.68e-4)	5.80e-1 (±4.2e-2)	52.2 (±1.6)	5.49e-1 (±1.46e-1)
32.05 [327.1]	32.05 [326.3 (±2.8e-1)]	7.39e-4 (±1.46e-4)	5.84e-1 (±4.8e-2)	53.6 (±1.0)	5.95e-1 (±1.43e-1)
32.00 [334.8]	32.01 [333.8 (±2.7e-1)]	7.62e-4 (±1.11e-4)	5.84e-1 (±4.6e-2)	53.8 (±1.9)	5.13e-1 (±4.06e-2)
31.50 [441.1]	31.99 [336.9 (±2.2e-1)]	1.17e-3 (±3.37e-4)	5.81e-1 (±4.5e-2)	48.1 (±4.8)	5.02e-1 (±9.8e-2))



(a) (b)

Figure 4.12 – (a) Scatter of RMSE for depth estimation, and (b) PSNR for deblurring, with respect to the focus (at 530nm) for the last 10 epochs (over 80) of DFD+EDOF training on RB patterns. Each colour accounts for an initial sensor position. The horizontal dotted line indicates the best performance achieved during single-task optimisation.

Conclusion on Random Binary patterns We observed that the multi-task optimisation has a stabilising effect on the convergence of the focus distance. However, it did not solve the problem of multiple local optima. And, again, the choice for the optimal optical system is not obvious and is still subject to a compromise.

4.5.2 . Multi-task on natural images

Similarly, we perform the end-to-end multi-task optimisation using natural images from DTD.

Figure 4.13 shows the evolution of the sensor position during the training. We observe the same behaviour as for the multi-task optimisation on the RB dataset, that is the stability to the initialisation brought by the EDOF task, as well as the multiple plateaus for the sensor position brought by the DFD task. Again, the range of optimal focus distances is narrower for natural texture than for RB patterns, illustrating a greater sensibility to the focus position for this database.

Regarding the performance in depth estimation, Figure 4.14.a shows the achieved RMSE with respect to the focus distance. We can note that 2 focus distances achieve better RMSE compared to single-task training. The lowest RMSE reached for this multi-task optimisation is obtained for a focus at 320mm, and happens to be quite lower than the lowest RMSE for the optimisation with a fixed focus at 319.7mm.

Regarding the performance in EDOF, Figure 4.14.b shows the achieved PSNR for the deblurred images with respect to the focus distance. We notice that the highest PSNR achieved by this multi-task approach is slightly smaller

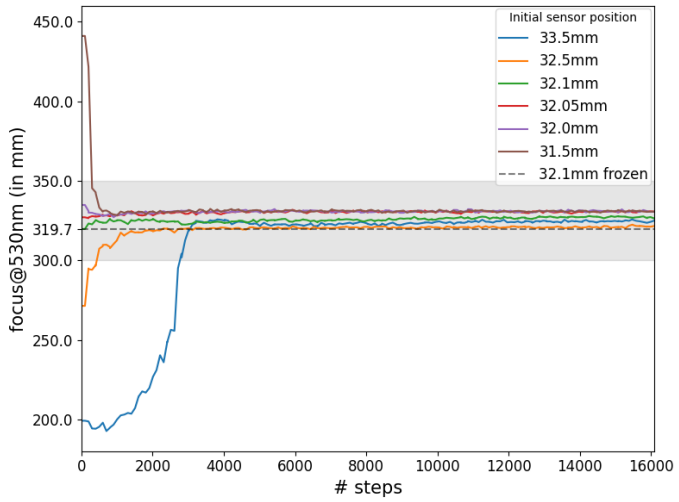


Figure 4.13 – Evolution of the focus during the multi-task training (DFD + EDOF) on DTD patterns given various initial sensor positions. The shaded area shows the range of depth of interest.

than the highest PSNR for the single-task training on DTD.

We report the statistics over the last 10 epochs for the final sensor position, the loss for either task, PSNR, and RMSE in Table 4.7.

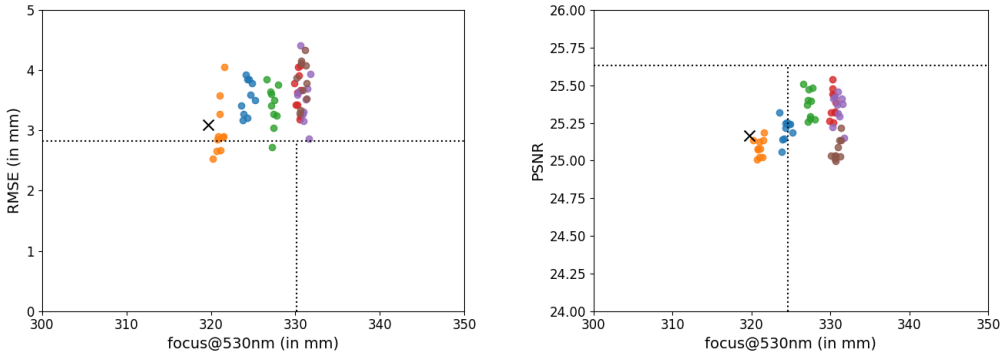
Once more, we can not find a single optimal focus distance that would yield the best estimations for both tasks. At the current stage, a trade-off in performance is implied when choosing a focus distance for the optical system.

Table 4.7 – Review for the joint optimisation of the sensor position and the DFD+EDOF model for various initial sensor positions. Average value (± standard deviation) over the 10 last epochs.

sensor position [focus] (in mm)		Loss		EDOF	DFD
initial	final	L_1	CE	PSNR	RMSE (mm)
33.50 [199.1]	32.07 [324.6 (±5.2e-1)]	6.81e-3 (±1.0e-3)	1.03 (±1.2e-1)	25.2 (±7.1e-2)	3.56 (±2.7e-1)
32.50 [271.4]	32.09 [321.2 (±5.7e-1)]	6.90e-3 (±9.1e-4)	1.01 (±1.1e-1)	25.1 (±5.6e-2)	3.03 (±4.5e-1)
32.10 [319.7]	32.05 [327.3 (±4.2e-1)]	6.61e-3 (±9.6e-4)	9.92e-01 (±1.09e-1)	25.4 (±9.0e-2)	3.40 (±3.3e-1)
32.05 [327.1]	32.03 [330.6 (±3.3e-1)]	6.69e-3 (±9.7e-4)	9.74e-01 (±9.8e-2)	25.4 (±9.0e-2)	3.64 (±3.1e-1)
32.00 [334.8]	32.02 [330.9 (±4.7e-1)]	6.78e-3 (±1.0e-3)	9.77e-01 (±1.16e-1)	25.3 (±9.2e-2)	3.54 (±4.1e-1)
31.50 [441.1]	32.03 [330.8 (±3.7e-1)]	6.93e-3 (±9.2e-4)	9.73e-01 (±1.01e-1)	25.1 (±6.8e-2)	3.82 (±3.4e-1)

4.5.3 . Conclusion on multi-task deep co-design

We performed joint optimisation for the focus distance of the optical system and for both DFD and EDOF networks simultaneously, by considering the simplest loss fusion. We observed that the evolution of the focus distance during the optimisation benefited from both tasks, in terms of stability to the initialisation and steadiness due to local optima introduced by the DFD task. We observed a performance gain from the multi-task optimisation, compared to



(a) (b)

Figure 4.14 – (a) Scatter of RMSE for depth estimation, and (b) PSNR for deblurring, with respect to the focus (at 530nm) for the last 10 epochs (over 80) of DFD+EDOF training on DTD. Each colour accounts for an initial sensor position. The horizontal dotted line indicates the best performance achieved during single-task optimisation.

the single-task, for DFD on natural textures. We suppose that multi-task training triggers a different exploration of the optimisation landscape, leading to different outcomes.

As for the single-task optimisation, the choice for the optimal focus is not obvious, as a trade-off in performance for both tasks needs to be considered, mostly because of the multiple local optima.

4.6 . Conclusion

In this chapter, we showed that it is possible to learn a parameter of an optical system and of our DFD network on image patches. For that, we performed supervised training of a chromatic optical system with one degree of freedom, modelled by a differential ray-tracer, and a lightweight neural network to produce depth estimations from blurred image patches. We found that the optimisation process falls in various local optima, or does not converge at all, depending on the initial optical setting, resulting in multiple optimal settings for the optical system. Each of these local optima provides relatively similar performances for the given task of DFD. Additionally, in order to build a full imaging system with good image quality and 3D capabilities, we addressed the problem of deblurring to mitigate the quality deterioration introduced by the defocus blur. We showed that the deblurring task is less sensitive to the optical setting, *i.e.*, the focus distance, compared to DFD.

So, through deep co-design, we found that for DFD and EDOF, there exists a reachable optimal performance that is better than when the optics is fixed. However, the focus distance that provides the best performance is different

for each task, so a compromise is necessary.

We proposed to address this compromise by making use of the deep co-design approach on both tasks simultaneously. The same optical model provides both neural networks with the same images, and the optimisation of the focus distance and network parameters is performed using a loss function that accounts for both tasks with a priority for the DFD. This led to several observations : Regarding the evolution of the focus distance during the learning phase, the optimisation process benefits the robustness to the initial focus distance from the EDOF. Besides, focus distances converge toward multiple optimal values, which might be a result of the priority given to the DFD for the optimisation. Then, regarding the performance for each task, we showed that the multi-task training led to instances of DFD neural networks reaching better or similar performances for DFD at a negligible cost in EDOF capabilities, especially for natural textures that are more challenging than structured binary patterns.

Finally, we found that a clear optimal setting for both tasks could not be obtained from the current deep co-design approach. The main reason for this is the existence of multiple local optima that appear in the DFD optimisation.

Perspectives The optimisation process of DFD for only one parameter (focus distance) is subject to multiple local optima, which makes the optimisation and the choice of the design sensitive to the initialisation. Except from [Fontbonne, 2021], very few comments on such sensitivity for co-design methods exist in the literature, and we showed with this preliminary work the importance of overcoming it.

In order to investigate further this local optima problem, a first step would be to try various optimisers, learning rates, and hyperparameters. A second step could be adding more degrees of freedom in the design of the optical system via more parameters to optimise, such as lens curvatures, and also considering large-field imaging, which is possible using DRT.

DFD and EDOF are closely related tasks, and some works use intermediate depth estimation to regularise deblurring [Zhang et al., 2018, Ma et al., 2022]. Therefore we could also investigate having both branches mutually share information to guide their estimation. It could be, for instance feeding the EDOF network with the depth estimation. We could also explore shared feature encoding between both tasks and two distinct heads for each estimate, or as in [Ikoma et al., 2021], one single convolutional neural network estimating directly the RGB-D image.

5 - Conclusions and Perspectives

5.1 . Conclusion

In this thesis, we addressed the problem of monocular depth estimation using deep-learning methods in order to design novel 3D vision systems, which led to the development of various estimation methods.

First, we presented in Chapter 2 an approach to tackle the problem of point cloud reconstruction for complex outdoor scenes from a single RGB image, using a lightweight 2D-3D hybrid neural network. The proposed method recovers properly distributed point clouds by taking advantage of an optimal transport loss. We also provided the first benchmark for this novel task on the KITTI dataset and introduced performance metrics to assess the quality of point cloud reconstruction. We showed that our method outperforms state-of-the-art depth map prediction methods when trained with sparse data. We also showed that point clouds from active acquisition methods, which have uniform coverage of the scene, were beneficial to the performance of the estimation model, compared to point clouds from passive approaches.

Then, in Chapter 3, we addressed the problem of local depth estimation by exploiting the defocus blur. We developed a regression approach for Depth from Defocus (DFD) on small texture patches, to take into account the continuous nature of depth. This approach takes advantage of a simple and stable classification architecture and a linear operator, named regression scale. The particularity of our method is that the training process relies on the soft-assignment encoding of the ground truth depth value to build a targeted membership probability vector that will be ultimately estimated by the network. The regression is then performed using the regression scale on the estimated membership probability vector. Our method is simple, requires no image prior nor additional information, and can be applied to any image/value matching data. We applied our approach to the DFD problem and showed it performs well on simulated and experimental data for small patches, on both planar scenes and natural 3D objects.

Finally, in Chapter 4, we addressed the problem of joint optimisation of an optical system and a neural network for DFD, in order for the optics to supply the estimator with images having the most informative defocus blur. We showed that it is possible to learn the parameters of the optical system and of the neural network jointly for DFD on image patches. For that, we performed supervised training of a chromatic optical system with one degree of freedom, modelled by a differential ray-tracer, and a lightweight neural network to produce depth estimations from blurred image patches. We found that the DFD task showed multiple optimal settings with similar performance. Additio-

nally, in order to build a full imaging system with good image quality and 3D capabilities, we addressed the problem of deblurring to mitigate the quality deterioration introduced by the defocus blur. We showed that the EDOF task is less sensitive to the focus distance, compared to DFD.

So, through deep co-design, we found that for DFD and EDOF, there exists a reachable optimal system that has a better performance than when the optics is fixed. However, the focus distance that provides the best performance is different for each task, so a compromise is necessary. We used a deep co-design approach on both tasks simultaneously to find a compromise. This led to several observations : regarding the evolution of the focus distance during the learning phase, the optimisation process benefits the robustness to the initial focus distance from the EDOF. Besides, focus distances converge toward multiple optimal values, which might be a result of the priority given to the DFD for the optimisation. Then, regarding the performance for each task, we showed that the multi-task training led to instances of DFD neural networks reaching better or similar performances at a negligible cost in EDOF capabilities, especially for natural textures that are more challenging than structured binary patterns.

5.2 . Perspectives & future works

Following our works, we discern two parts to our perspectives : the first part relates to data we consider for building our models, and the second part accounts for algorithmic developments for each of our methods.

5.2.1 . Collecting "in the wild" data

For the development of our local DFD estimation method, as well as the deep co-design approach, we considered two datasets of image patches having different styles of texture : the Random Binary dataset, whose textures are structured and apply to controlled environments, and the Describable Texture Dataset [Cimpoi et al., 2014], whose textures are more natural and apply to uncontrolled environments. We developed the local DFD method using simulated blurs for both styles of textures and we tested it on the experimental data of [Buat et al., 2020], which involve only RB patterns in a controlled environment setting. However, it would be interesting to also apply our methods to experimental data of natural images, whose blur can be controlled using lenses dedicated to DFD. This latter problem expresses a need to build our own databases. For that exact motive, a portable acquisition platform, for both image and 3D, named *Maratus* has been developed at ONERA prior to my thesis. This platform, shown in Figure 5.1, comprises a stereo baseline of two identical cameras and a RealSense RGB-D camera, which gives *Maratus* both passive and active 3D capabilities. Unconventional camera systems are

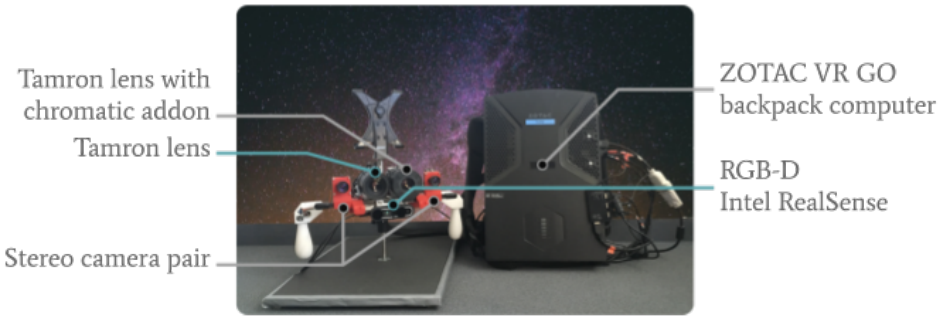


Figure 5.1 – Description of Maratus, a nomad platform for image and 3D acquisition.

also mounted in between the stereo cameras. With that setting, *Maratus* can provide us with databases of images from unconventional optics and corresponding depth.

camera	focal distance	f-number	sensor resolution	pixel size
Stereo	5.5mm	1.8	1600×1200	4.5µm
RealSense D435	1.93mm	2.0	1280×720	3µm
DFD (chromatic add-on)	16mm	1.8	2046×2592	4.8µm
DFD (Cooke)	25mm	4.0	2456×2054	3.45µm

Table 5.1 – Specifications for cameras mounted on *Maratus*.

The first optical system for which *Maratus* supplied depth is a set of two cameras having different apertures, controlled by Tamron lenses. The idea behind this setting was to characterise experimentally the effect of defocus blur for deep learning depth estimation methods [Carvalho et al., 2018a]. Figure 5.2 shows available data for one example of this dataset : (a) and (b) the RealSense RGB-D images, (c) a large depth-of-field image, sharper than (d) the corresponding small depth-of-field image having stronger variation of defocus blur within the depth range of interest. A greater concern was given to chromatic blur during my thesis. In [Trouvé-Peloux et al., 2018], an optical element has been developed in order to enhance the chromatic aberration for conventional optics, giving the resulting camera system 3D capabilities through chromatic DFD. In order to build a database with outdoor chromatic images, we considered adding that particular optical element to one of the Tamron lenses. We also added onto *Maratus*, the Cooke triplet camera system used in Chapter 3, with the focus set at 2 meters away from the camera. Unlike the Tamron lens, the Cooke triplet's lens architecture and specifications are known and can be given to a DRT model, hence allowing experimental validation for the optimisation of the focus setting using deep co-design approaches. Figure 5.3 shows available data for one example of the dataset I have built after calibration of the platform : (a) and (b) are the stereo RGB and

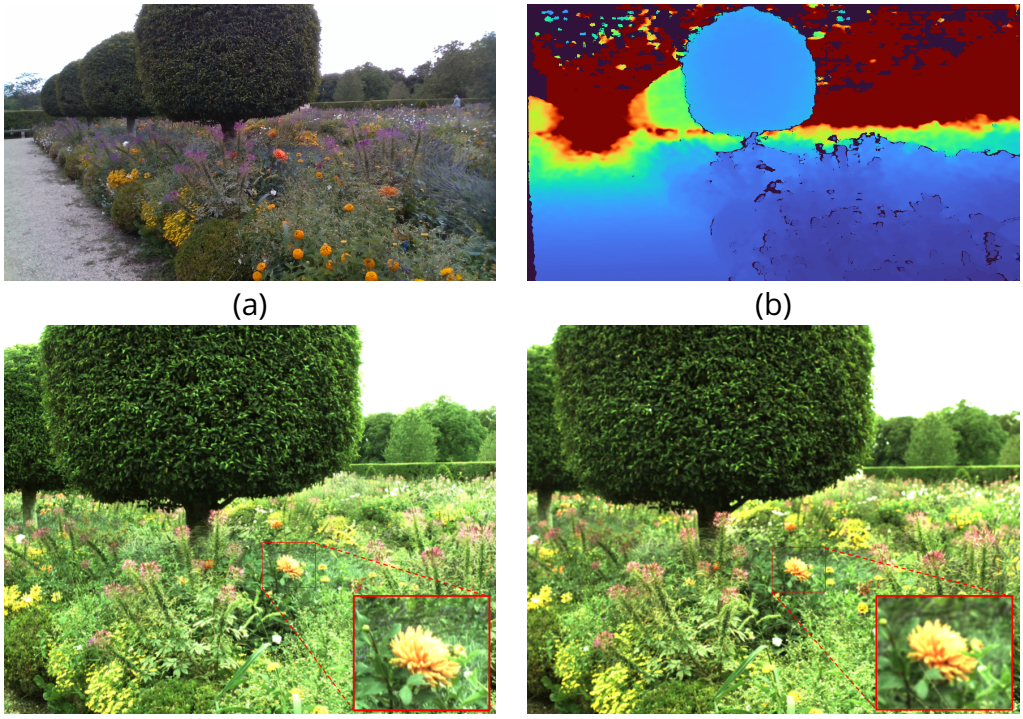


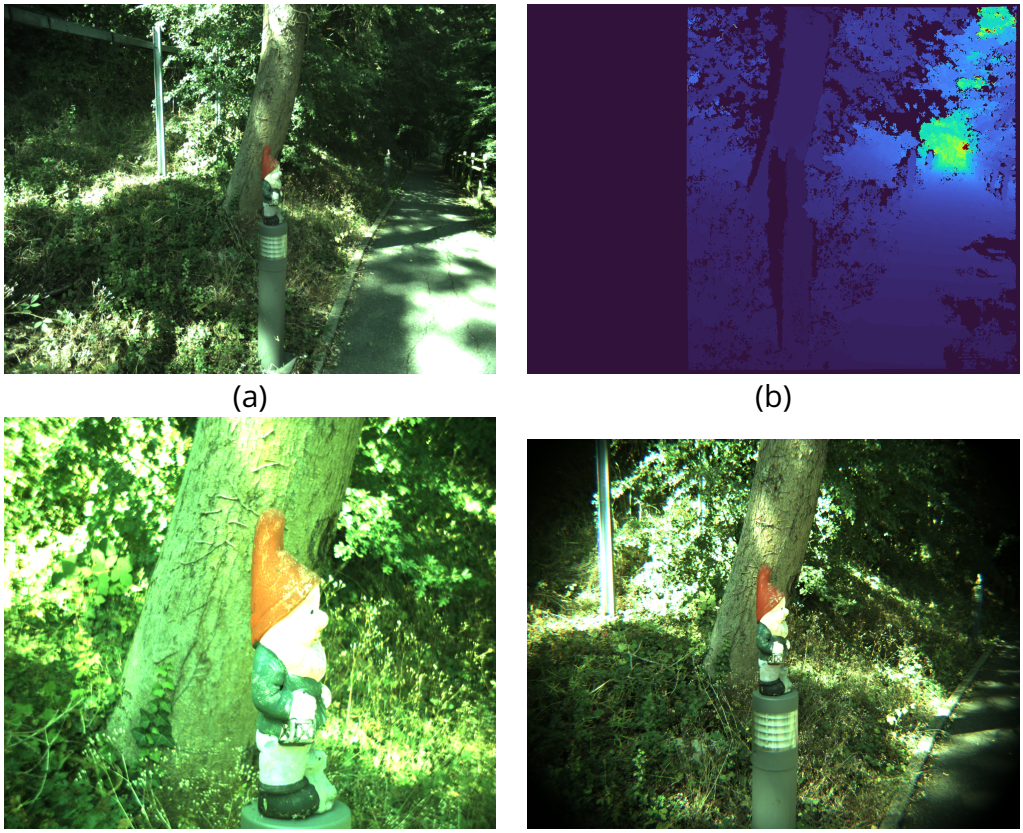
Figure 5.2 – Example of data for the small depth of field set. (a) RealSense RGB, (b) RealSense depth map, (c) Tamron small aperture, and (d) Tamron large aperture images.

projected depth map, (c) the chromatic image from the Cooke triplet, and (d) the image from the Tamron camera with the chromatic add-on. The system with the chromatic Tamron has a greater field of view compared to the Cooke triplet, which will be used to investigate depth estimation and image restoration problems in off-axis regions.

The optical and sensor specifications for each camera mounted on *Maratus* are reported in Table 5.1.

Even though they have not been used yet, these databases will have several uses. In addition to previously stated problems, these databases will help us characterise the robustness of the estimators to "in the wild" settings, *i.e.*, natural images subject to motion blur, and noise inconsistency throughout the image due to uncontrolled lighting.

These databases were built using arbitrary fixed optical settings, however, as we have developed a working deep co-design framework, we could use it to find the best optical setting for the chromatic system, and build tailored databases for outdoor depth estimation.



(a) (b) (c) (d)

Figure 5.3 – Example of data for the chromatic set. (a) Stereo RGB, (b) Stereo depth map, (c) Cooke triplet image, and (d) Tamron image with the chromatic add-on.

5.2.2 . Algorithmic perspectives

Now, we will present the perspectives accompanying the different algorithmic contributions we have made during this thesis.

Outdoor point cloud estimation

We have seen in Chapter 2 that the point cloud estimation of Pix2Point could be improved in several aspects.

The first aspect relates to the global scale estimation and the rigidity of the network for processing the images. These two characteristics originate from the fully connected layer in charge of converting the encoded spatial features from the whole image, into a first coarse point cloud, whose number of elements is fixed by design. However, using the whole image can be detrimental to the estimation of objects at a finer scale, as a local texture or intensity change in the image can affect the entire estimation. These two shortcomings

can be mitigated by dividing the image into smaller regions and having a dedicated neural network for each region to predict a local point cloud. The global point cloud would be the aggregation of the local point clouds, therefore increasing the total number of points. Another approach to preserve spatial neighbourhood information would be to consider a fully convolutional network, in that case, the total number of predicted points would be determined by the input image resolution.

A second aspect relates to the spatial resolution of the estimations. We observed that close objects in the scene were not finely estimated by our method. We explain this behaviour as a result of the global scale approach. This makes our method unable to differentiate between coarse features in the image - such as road or building surfaces, that could be assigned to a coarse-scale point cloud, and for which we could have higher error tolerance - and finer features that would account for possible obstacles on the road, road signs, or pedestrians, for which estimation accuracy is critical. To mitigate this, a multi-scale estimation mechanism could be added in order to enable that scale differentiation. For instance, one approach would be first to estimate a coarse scale point cloud, then use it to condition immediately finer scale estimations.

Maratus can support these aspects with training data acquired in both passive (with stereo) and active (with RealSense) ways.

As semantic understanding of 3D scenes is a crucial task for autonomous driving, a third and broader aspect we could consider is adding semantic segmentation of the estimated point cloud on top of the geometric estimation. This could be implemented using a multi-task learning scheme that could be beneficial for both task, as in [Carvalho et al., 2019].

Combination of global and local estimation

In this thesis, we addressed both global and local estimation approaches independently, however, it would be interesting to develop MDE methods combining both estimation scales. We could find inspiration in the work of [Lee et al., 2019a] that proposes a global scale MDE method guided by local geometrical constraints. The global scale could help regularise the local DFD estimations using context information for patches having low texture, and conversely. Yet, for local approaches to handle large-field images, the problem of off-axis aberrations must be engaged, especially for unconventional optics that may reinforce this phenomenon. It was previously handled in our work by dividing the large-field images into several regions where the PSF is considered invariant, and on which a different model is learned. To address this issue more simply, we could use an approach similar to [Kashiwagi et al., 2019] by providing the estimator with the patch location of the processed patch in

the large-field image. Again, this topic could be supplied with substantial data acquired by *Maratus*.

Deep co-design

In Chapter 4, we highlighted several progression axes resulting from our preliminary work on multi-task deep co-design of an optical system and dedicated neural networks. First, we faced a phenomenon of multiple local convergences that is rarely commented on in the literature, which reveals the sensitivity of the optimal optical settings to the initialisation [Fontbonne, 2021] and to the optimisation algorithm. Our first take on this issue would be to have a better characterisation of that phenomenon by trying different optimisers and settings. The next step would be to consider more degrees of freedom for the optical element to optimise. For instance, the radius of curvature for one lens of the Cooke triplet could be added to the optimisation. Over-parametrisation could avoid the optical system to collapse in a local optimum during training. This local optima issue could also originate from the differential ray-tracing model used for the optimisation. Therefore, we could try a different optical model, such as Fourier optics, to characterise the origin of this problem.

Regarding the choice of the most suitable optical system for both DFD and EDOF, we looked at the optical setting for which each task exhibited the best performance at a given optimisation step when the optical parameter was still free to vary. We saw that the best-reached performance for both tasks could be different for the same focus distance. Therefore, it could be interesting to add a supplementary step to the optimisation in which the optical parameter is fixed after convergence, in order to only optimise the neural network model. Exploring further this topic is currently subject to a PhD proposal in our team. Moreover, a PhD thesis currently conducted by Marius Dufraisse in our team aims to explore deep co-design of complete lens structures [Dufraisse et al., 2022] for higher-level tasks, such as classification and adversarial tasks [Hinojosa et al., 2022].

5.3 . Concluding note

In this thesis, we addressed the different parts composing a 3D monocular vision system, from information sensing to processing. We proposed deep learning methods that could be used in a co-design framework. We believe that intricate optimisation for smart sensing, enabled by deep co-design, would lead to more efficient and compact vision systems, for tasks ranging from 3D perception to higher-level tasks such as classification and decision-making.

Bibliography

[Akpinar et al., 2019] Akpinar, U., Sahin, E., and Gotchev, A. (2019). Learning optimal phase-coded aperture for depth of field extension. In 2019 IEEE International Conference on Image Processing (ICIP), pages 4315–4319.

[Amiri et al., 2019] Amiri, A. J., Loo, S., and Zhang, H. (2019). Semi-supervised monocular depth estimation with left-right consistency using deep neural network. In IEEE Int. Conf. on Robotics and Biomimetics (ROBIO).

[Anwar et al., 2021] Anwar, S., Hayder, Z., and Porikli, F. (2021). Deblur and deep depth from single defocus image. Machine Vision and Applications.

[Bhat et al., 2021] Bhat, S. F., Alhashim, I., and Wonka, P. (2021). AdaBins : Depth estimation using adaptive bins. Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR).

[Buat et al., 2022] Buat, B., Trouvé-Peloux, P., Champagnat, F., and Le Besnerais, G. (2022). Single image depth-from-defocus with a learned covariance : algorithm and performance model for co-design. In SPIE PHOTONICS EUROPE 2022, volume 12136 of Proceedings of SPIE, STRASBOURG, France.

[Buat et al., 2020] Buat, B., Trouvé-Peloux, P., Champagnat, F., Le Besnerais, G., and Simon, T. (2020). Active chromatic depth from defocus for industrial inspection. In Unconventional Optical Imaging II. International Society for Optics and Photonics.

[Buat et al., 2021] Buat, B., Trouvé-Peloux, P., Champagnat, F., and Besnerais, G. L. (2021). Learning scene and blur model for active chromatic depth from defocus. Applied Optics, 60(31).

[Caccia et al., 2019] Caccia, L., van Hoof, H., Courville, A., and Pineau, J. (2019). Deep generative modeling of LiDAR data. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS).

[Carvalho et al., 2018a] Carvalho, M., Le Saux, B., Trouvé-Peloux, P., Almansa, A., and Champagnat, F. (2018a). Deep Depth from Defocus : How can defocus blur improve 3D estimation using dense neural networks? In Proc. IEEE/CVF Eur. Conf. on Computer Vision Worskhops (ECCVW).

[Carvalho et al., 2018b] Carvalho, M., Le Saux, B., Trouvé-Peloux, P., Champagnat, F., and Almansa, A. (2018b). On regression losses for deep depth estimation. In Proc. IEEE Int. Conf. on Image Processing (ICIP).

[Carvalho et al., 2019] Carvalho, M., Le Saux, B., Trouvé-Peloux, P., Champagnat, F., and Almansa, A. (2019). Multitask learning of height and semantics from aerial images. IEEE Geoscience and Remote Sensing Letters, 17(8) :1391–1395.

[Chang and Wetzstein, 2019] Chang, J. and Wetzstein, G. (2019). Deep optics for monocular depth estimation and 3D object detection. In Proc. IEEE/CVF Int. Conf. on Computer Vision (ICCV).

[Cimpoi et al., 2014] Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., , and Vedaldi, A. (2014). Describing textures in the wild. In Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).

[Cuturi, 2013] Cuturi, M. (2013). Sinkhorn distances : Lightspeed computation of optimal transport. In Advances in Neural Information Processing System 26.

[Debarnot and Weiss, 2022] Debarnot, V. and Weiss, P. (2022). Deep-blur : Blind identification and deblurring with convolutional neural networks.

[Denninger and Triebel, 2020] Denninger, M. and Triebel, R. (2020). 3D scene reconstruction from a single viewport. In IEEE/CVF Eur. Conf. on Computer Vision.

[Diaz et al., 2009] Diaz, F., Goudail, F., Loiseaux, B., and Huignard, J.-P. (2009). Increase in depth of field taking into account deconvolution by optimization of pupil mask. Opt. Lett., 34(19) :2970–2972.

[Dowski and Cathey, 1995] Dowski, E. R. and Cathey, W. T. (1995). Extended depth of field through wave-front coding. Appl. Opt., 34(11) :1859–1866.

[Dufraisse et al., 2022] Dufraisse, M., Trouvé-Peloux, P., Volatier, J.-B., and Champagnat, F. (2022). On the use of differentiable optical models for lens and neural network co-design. In Unconventional Optical Imaging III, volume 12136, pages 164–173. SPIE.

[D'Andrès et al., 2016] D'Andrès, L., Salvador, J., Kochale, A., and Süsstrunk, S. (2016). Non-parametric blur map regression for depth of field extension. IEEE Transactions on Image Processing.

[Eigen et al., 2014] Eigen, D., Puhrsch, C., and Fergus, R. (2014). Depth map prediction from a single image using a multi-scale deep network. Advances in neural information processing systems, 27.

[Elmalem et al., 2018] Elmalem, S., Giryes, R., and Marom, E. (2018). Learned phase coded aperture for the benefit of depth of field extension. Opt. Express, 26(12) :15316–15331.

[Falcón et al., 2017] Falcón, R., Goudail, F., Kulcsár, C., and Sauer, H. (2017). Performance limits of binary annular phase masks codesigned for depth-of-field extension. Optical Engineering, 56(6) :065104.

[Fan et al., 2017] Fan, H., Su, H., and Guibas, L. J. (2017). A point set generation network for 3D object reconstruction from a single image. In Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR).

[Faugeras, 1993] Faugeras, O. (1993). Three-Dimensional Computer Vision : A Geometric Viewpoint.

[Feydy et al., 2019] Feydy, J., Séjourné, T., Vialard, F.-X., Amari, S.-i., Trouvé, A., and Peyré, G. (2019). Interpolating between optimal transport and MMD using sinkhorn divergences. In Int. Conf. on Artificial Intelligence and Statistics.

[Fontbonne, 2021] Fontbonne, A. (2021). Conception conjointe combinaison optique / traitement : Une nouvelle approche de la conception optique de haut niveau. PhD thesis. Thèse de doctorat dirigée par Goudail, François Physique université Paris-Saclay 2021.

[Fontbonne et al., 2019] Fontbonne, A., Sauer, H., Kulcsár, C., Coutrot, A.-L., and Goudail, F. (2019). Experimental validation of hybrid optical–digital imaging system for extended depth-of-field based on co-optimized binary phase masks. Optical Engineering, 58(11) :113107.

[Fu et al., 2018] Fu, H., Gong, M., Wang, C., Batmanghelich, K., and Tao, D. (2018). Deep ordinal regression network for monocular depth estimation. In Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR).

[Geiger et al., 2013] Geiger, A., Lenz, P., Stiller, C., and Urtasun, R. (2013). Vision meets robotics : The KITTI dataset. International Journal of Robotics Research (IJRR).

[Haim et al., 2018] Haim, H., Elmalem, S., Giryes, R., Bronstein, A. M., and Marom, E. (2018). Depth estimation from a single image using deep learned phase coded mask. IEEE Transactions on Computational Imaging.

[Halé et al., 2021] Halé, A., Trouvé-Peloux, P., and Volatier, J.-B. (2021). End-to-end sensor and neural network design using differential ray tracing. Optics Express, 29(21) :34748.

[Hartley and Zisserman, 2004] Hartley, R. I. and Zisserman, A. (2004). Multiple View Geometry in Computer Vision.

[He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778.

[Hinojosa et al., 2022] Hinojosa, C., Marquez, M., Arguello, H., Adeli, E., Fei-Fei, L., and Niebles, J. C. (2022). Privhar : Recognizing human actions from privacy-preserving lens. In Computer Vision – ECCV 2022 : 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IV, page 314–332. Springer-Verlag.

[Hirschmueller, 2008] Hirschmueller, H. (2008). Stereo processing by semi-global matching and mutual information. Trans. Pattern Analysis and Machine Intelligence, 30 :328–41.

[Huang et al., 2017] Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. (2017). Densely connected convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4700–4708.

[Ikoma et al., 2021] Ikoma, H., Nguyen, C. M., Metzler, C. A., Peng, Y., and Wetzstein, G. (2021). Depth from defocus with learned optics for imaging and occlusion-aware depth estimation. In Proc. IEEE Int. Conf. on Computational Photography (ICCP).

[Kashiwagi et al., 2019] Kashiwagi, M., Mishima, N., Kozakaya, T., and Hiura, S. (2019). Deep depth from aberration map. In Proc. IEEE/CVF Int. Conf. on Computer Vision (ICCV).

[Lee et al., 2019a] Lee, J., Han, M., Ko, D., and Suh, I. (2019a). From Big to Small : Multi-scale local planar guidance for monocular depth estimation. arXiv preprint arXiv :1907.10326.

[Lee et al., 2019b] Lee, J., Lee, S., Cho, S., and Lee, S. (2019b). Deep defocus map estimation using domain adaptation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.

[Leroy et al., 2020] Leroy, R., Le Saux, B., de Carvalho, M. P., Trouvé-Peloux, P., and Champagnat, F. (2020). Pix2point : prédiction monoculaire de scènes 3d par réseaux de neurones hybrides et transport optimal. In RFIAP.

[Leroy et al., 2022a] Leroy, R., Trouvé-Peloux, P., Saux, B. L., Buat, B., and Champagnat, F. (2022a). Learning local depth regression from defocus blur by soft-assignment encoding. Appl. Opt., 61(29) :8843–8849.

[Leroy et al., 2021] Leroy, R., Trouvé-Peloux, P., Champagnat, F., Le Saux, B., and Carvalho, M. (2021). Pix2Point : Learning outdoor 3D using sparse point clouds and optimal transport. In Int. Conf. on Machine Vision and Applications (MVA).

[Leroy et al., 2022b] Leroy, R., Trouvé-Peloux, P., Saux, B. L., Buat, B., and Champagnat, F. (2022b). Régression locale de la profondeur grâce au flou de défocalisation et à un réseau de neurones entraîné par soft-assignment. In 28Â° Colloque sur le traitement du signal et des images, number 001-0301, pages p. 1205–1208, Nancy. GRETSI - Groupe de Recherche en Traitement du Signal et des Images.

[Lévêque et al., 2020] Lévêque, O., Kulcsár, C., Lee, A., Sauer, H., Aleksanyan, A., Bon, P., Cognet, L., and Goudail, F. (2020). Co-designed annular binary phase masks for depth-of-field extension in single-molecule localization microscopy. Optics Express, 28(22) :32426–32446.

[Levin et al., 2007] Levin, A., Fergus, R., Durand, F., and Freeman, W. (2007). Image and depth from a conventional camera with a coded aperture. In Proc. ACM SIGGRAPH.

[Liu et al., 2011] Liu, L., Wang, L., and Liu, X. (2011). In defense of soft-assignment coding. In 2011 International Conference on Computer Vision. IEEE.

[Ma et al., 2022] Ma, H., Liu, S., Liao, Q., Zhang, J., and Xue, J. (2022). Defocus image deblurring network with defocus map estimation as auxiliary task. IEEE Transactions on Image Processing.

[Mandikal and Radhakrishnan, 2019] Mandikal, P. and Radhakrishnan, V. B. (2019). Dense 3D point cloud reconstruction using a deep pyramid network. In IEEE Winter Conf. on Applications of Computer Vision (WACV).

[Mao et al., 2016] Mao, X., Shen, C., and Yang, Y.-B. (2016). Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. Advances in neural information processing systems, 29.

[Martinello and Favaro, 2011] Martinello, M. and Favaro, P. (2011). Single image blind deconvolution with higher-order texture statistics. In Video Processing and Computational Video. Springer.

[Mel et al., 2022] Mel, M., Siddiqui, M., and Zanuttigh, P. (2022). End-to-end learning for joint depth and image reconstruction from diffracted rotation. arXiv preprint arXiv :2204.07076.

[Nayar and Nakagawa, 1994] Nayar, S. and Nakagawa, Y. (1994). Shape from focus. IEEE Transactions on Pattern Analysis and Machine Intelligence, 16(8) :824–831.

[Ng et al., 2005] Ng, R., Levoy, M., Brédif, M., Duval, G., Horowitz, M., and Hanrahan, P. (2005). Light field photography with a hand-held plenoptic camera. PhD thesis, Stanford University.

[Paszke et al., 2019] Paszke, A., Gross, S., Massa, F., et al. (2019). Pytorch : An imperative style, high-performance deep learning library. In Advances in Neural Information Processing Systems 32.

[Pentland, 1987] Pentland, A. P. (1987). A new sense for depth of field. IEEE Transactions on Pattern Analysis and Machine Intelligence.

[Perwass and Wietzke, 2012] Perwass, C. and Wietzke, L. (2012). Single lens 3d-camera with extended depth-of-field. In Human vision and electronic imaging XVII, volume 8291, pages 45–59. SPIE.

[Proença and Gao, 2020] Proença, P. F. and Gao, Y. (2020). Deep learning for spacecraft pose estimation from photorealistic rendering. In 2020 IEEE International Conference on Robotics and Automation (ICRA). IEEE.

[Pumarola et al., 2020] Pumarola, A., Popov, S., Moreno-Noguer, F., and Ferrari, V. (2020). C-Flow : Conditional generative flow models for images and 3D point clouds. In IEEE/CVF Conf. on Computer Vision and Pattern Recognition.

[Qi et al., 2017a] Qi, C. R., Su, H., Mo, K., and Guibas, L. J. (2017a). PointNet : Deep learning on point sets for 3D classification and segmentation. In IEEE Conf. on Computer Vision and Pattern Recognition, CVPR.

[Qi et al., 2017b] Qi, C. R., Yi, L., Su, H., and Guibas, L. J. (2017b). PointNet++ : Deep hierarchical feature learning on point sets in a metric space. In Advances in Neural Information Processing Systems 30.

[Ranftl et al., 2020] Ranftl, R., Lasinger, K., Hafner, D., Schindler, K., and Koltun, V. (2020). Towards robust monocular depth estimation : Mixing datasets for zero-shot cross-dataset transfer. IEEE transactions on pattern analysis and machine intelligence.

[Saxena et al., 2006] Saxena, A., Chung, S., and Ng, A. Y. (2006). Learning depth from single monocular images. In Advances in Neural Information Processing Systems 18.

[Shajkofci and Liebling, 2020] Shajkofci, A. and Liebling, M. (2020). Spatially-variant cnn-based point spread function estimation for blind deconvolution and depth estimation in optical microscopy. IEEE Transactions on Image Processing, 29 :5848–5861.

[Silberman et al., 2012] Silberman, N., Hoiem, D., Kohli, P., and Fergus, R. (2012). Indoor segmentation and support inference from rgbd images. In ECCV.

[Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv :1409.1556.

[Sitzmann et al., 2018] Sitzmann, V., Diamond, S., Peng, Y., Dun, X., Boyd, S., Heidrich, W., Heide, F., and Wetzstein, G. (2018). End-to-end optimization of optics and image processing for achromatic extended depth of field and super-resolution imaging. ACM Transactions on Graphics (TOG), 37(4) :114.

[Sun et al., 2015] Sun, J., Cao, W., Xu, Z., and Ponce, J. (2015). Learning a convolutional neural network for non-uniform motion blur removal. In CVPR.

[Sun et al., 2021] Sun, Q., Wang, C., Fu, Q., Dun, X., and Heidrich, W. (2021). End-to-end complex lens design with differentiate ray tracing. ACM Trans. Graph.

[Sun et al., 2019] Sun, R., Gao, Y., Fang, Z., Wang, A., and Zhong, C. (2019). Ssl-net : Point-cloud generation network with self-supervised learning. IEEE Access, 7 :82206–82217.

[Trouvé et al., 2013] Trouvé, P., Champagnat, F., Besnerais, G. L., Sabater, J., Avignon, T., and Idier, J. (2013). Passive depth estimation using chromatic aberration and a depth from defocus approach. Appl. Opt., 52(29) :7152–7164.

[Trouvé-Peloux et al., 2014] Trouvé-Peloux, P., Champagnat, F., Le Besnerais, G., and Idier, J. (2014). Theoretical performance model for single image depth from defocus. JOSA A, 31(12) :2650–2662.

[Trouvé-Peloux et al., 2018] Trouvé-Peloux, P., Sabater, J., Bernard-Brunel, A., Champagnat, F., Le Besnerais, G., and Avignon, T. (2018). Turning a conventional camera into a 3D camera with an add-on. Applied Optics.

[Trouvé et al., 2011] Trouvé, P., Champagnat, F., Besnerais, G. L., and Idier, J. (2011). Single image local blur identification. In Proc. IEEE Int. Conf. on Image Processing (ICIP).

[Trouvé et al., 2013] Trouvé, P., Champagnat, F., Le Besnerais, G., Druart, G., and Idier, J. (2013). Design of a chromatic 3D camera with an end-to-end performance model approach. In IEEE Conf. Comput. Vis. Pattern Recog. Workshops.

[Tylecek et al., 2018] Tylecek, R., Sattler, T., Le, H.-A., Brox, T., Pollefeys, M., Fisher, R. B., and Gevers, T. (2018). The second workshop on 3D Reconstruction Meets Semantics : Challenge results discussion. In IEEE/CVF Eur. Conf. on Computer Vision Workshops.

[Ullman, 1979] Ullman, S. (1979). The interpretation of structure from motion. Proceedings of the Royal Society of London. Series B. Biological Sciences, 203(1153) :405–426.

[Ulyanov et al., 2016] Ulyanov, D., Vedaldi, A., and Lempitsky, V. (2016). Instance normalization : The missing ingredient for fast stylization. arXiv preprint arXiv :1607.08022.

[Volatier et al., 2017] Volatier, J.-B., Álvaro Menduiña Fernández, and Erhard, M. (2017). Generalization of differential ray tracing by automatic differentiation of computational graphs. J. Opt. Soc. Am. A, 34(7) :1146–1151.

[Wang et al., 2019] Wang, Y., Chao, W.-L., Garg, D., Hariharan, B., Campbell, M., and Weinberger, K. Q. (2019). Pseudo-LiDAR from visual depth estimation : Bridging the gap in 3D object detection for autonomous driving. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

[Wu et al., 2019] Wu, Y., Boominathan, V., Chen, H., Sankaranarayanan, A., and Veeraraghavan, A. (2019). Phasecam3d — learning phase masks for passive single view depth estimation. In 2019 IEEE International Conference on Computational Photography (ICCP), pages 1–12.

[Xia et al., 2018] Xia, Y., Zhang, Y., Zhou, D., Huang, X., Wang, C., and Yang, R. (2018). RealPoint3D : Point cloud generation from a single image with complex background. arXiv preprint arXiv :1809.02743.

[Yan and Shao, 2016] Yan, R. and Shao, L. (2016). Blind image blur estimation via deep learning. IEEE Transactions on Image Processing.

[You et al., 2019] You, Y., Wang, Y., Chao, W.-L., Garg, D., Pleiss, G., Hariharan, B., Campbell, M., and Weinberger, K. Q. (2019). Pseudo-LiDAR++ : Accurate depth for 3D object detection in autonomous driving. In International Conference on Learning Representations.

[Zhang et al., 2018] Zhang, S., Shen, X., Lin, Z., Mech, R., Costeira, J. P., and Moura, J. M. (2018). Learning to understand image blur. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition.

[Zhou et al., 2009] Zhou, C., Lin, S., and Nayar, S. (2009). Coded aperture pairs for depth from defocus. In 2009 International Conference on Computer Vision, pages 325–332. IEEE.

[Zhu et al., 2013] Zhu, X., Cohen, S., Schiller, S., and Milanfar, P. (2013). Estimating spatially varying defocus blur from a single image. IEEE Transactions on Image Processing.

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Méthodes d'apprentissage profond pour systèmes de vision 3D
Mots clés : apprentissage profond, nuage de points 3D, co-conception

Résumé : Dans cette thèse, nous étudions l'apport de l'apprentissage profond pour les systèmes de vision 3D monoculaire, de l'acquisition de l'image au traitement. Nous proposons d'abord Pix2Point, une méthode d'estimation de nuage de points 3D à partir d'une seule image en utilisant des informations de contexte, et entraînée avec une fonction de coût de transport optimal. Pix2Point réalise une meilleure couverture des scènes lorsqu'il est entraîné sur des nuages de points lacunaires que les méthodes d'estimation de profondeur monoculaire, entraînées sur des cartes de profondeur lacunaires. Deuxièmement, pour exploiter les indices de profondeur provenant du capteur, nous proposons une méthode de régression de profondeur à partir d'un patch défocalisé. Cette méthode surpasse la classification et la régression directe, sur données simulées et réelles. Enfin, nous abordons la conception d'un système de vision RVB-D, composé d'un capteur dont l'image est traitée par notre réseau de régression de profondeur basée sur la défocalisation et par un réseau de défloutage d'image. Nous proposons un cadre d'optimisation multi-tâches, conjointement aux paramètres des capteurs et des réseaux, et nous l'appliquons à l'optimisation de la mise au point d'une lentille chromatique. Le paysage d'optimisation présente plusieurs optima liés à la tâche de régression en profondeur, tandis que la tâche de défloutage semble moins sensible au paramètre de mise au point. En résumé, cette thèse propose plusieurs contributions exploitant les réseaux de neurones pour l'estimation 3D monoculaire et ouvre la voie d'une conception conjointe de systèmes RVB-D.

Title : Deep Learning methods for monocular 3D vision systems
Keywords : Deep learning, 3D point clouds, Co-design

Abstract : In this thesis, we explore deep learning methods for monocular 3D vision systems, from image acquisition to processing. We first propose Pix2Point, a method for 3D point cloud prediction from a single image using context information, trained with an optimal transport loss. Pix2Point achieves a better coverage of the scenes when trained on sparse point clouds than monocular depth estimation methods, trained on sparse depth maps. Second, to exploit sensor depth cues, we propose a depth regression method from a defocused patch, which outperforms classification and direct regression, on simulated and real data. Finally, we tackle the design of a RGB-D monocular vision system for which the image is processed jointly by our defocus-based depth regression method and a simple image deblurring network. We propose an end-to-end multi-task optimisation framework of sensor and network parameters, that we apply to the focus optimisation for a chromatic lens. The optimisation landscape presents multiple optima, due to the depth regression task, while the deblurring task appears less sensitive to the focus. This thesis hence contains several contributions exploiting neural networks for monocular 3D estimation and paves the way towards end-to-end design of RGB-D systems.