# Depth Estimation from a Single Image using Deep Learned Phase Coded Mask

Harel Haim, Shay Elmalem, Raja Giryes, *Member, IEEE,* Alex M. Bronstein, *Fellow, IEEE,* and Emanuel Marom, *Fellow, IEEE*

***Abstract*—Depth estimation from a single image is a well known challenge in computer vision. With the advent of deep learning, several approaches for monocular depth estimation have been proposed, all of which have inherent limitations due to the scarce depth cues that exist in a single image. Moreover, these methods are very demanding computationally, which makes them inadequate for systems with limited processing power. In this paper, a phase-coded aperture camera for depth estimation is proposed. The camera is equipped with an optical phase mask that provides unambiguous depth-related color characteristics for the captured image. These are used for estimating the scene depth map using a fully-convolutional neural network. The phase-coded aperture structure is learned jointly with the network weights using back-propagation. The strong depth cues (encoded in the image by the phase mask, designed together with the network weights) allow a much simpler neural network architecture for faster and more accurate depth estimation. Performance achieved on simulated images as well as on a real optical setup is superior to state-of-the-art monocular depth estimation methods (both with respect to the depth accuracy and required processing power), and is competitive with more complex and expensive depth estimation methods such as light-field cameras.**

***Index Terms*—Coded Aperture, Phase Mask, Depth Reconstruction, Deep Learning, Computational Camera,.**

## I. Introduction

PASSIVE depth estimation is a well-known challenge in computer vision. A common solution is based on stereo vision, where two calibrated cameras capture the same scene from different views (similarly to the human eyes), and thus the distance to every object can be inferred by triangulation. Yet, such a dual camera system significantly increases the form factor, cost and power consumption.

The current electronics miniaturization trend (high quality smart-phone cameras, wearable devices, etc.) requires a compact and low-cost solution. This requirement dictates a more challenging task: passive depth estimation from a single image. While a single image lacks the depth cues that exist in a stereo image pair, there are still some depth cues such as perspective lines and vanishing points that enable depth estimation to some degree of accuracy. The ongoing deep learning revolution did not overlook this challenge, and some neural network-based approaches to monocular depth estimation exist in the literature [1]–[8].

H. Haim was with the Faculty of Electrical Engineering, Tel-Aviv University, Tel-Aviv, 6997801 Israel, and currently with the Department of Computer Science, University of Toronto, Toronto, Canada.

S. Elmalem, R. Giryes, A.M. Bronstein and E. Marom are with the Faculty of Electrical Engineering, Tel-Aviv University, Tel-Aviv, 6997801 Israel

Eigen *et al.* [1] introduced a deep neural network for depth estimation that relies on depth cues in the RGB image. They used a multi-scale architecture with coarse and fine depth estimation networks concatenated to achieve both dynamic range and resolution. Two later publications by Cao *et al.* [2] and Liu *et al.* [3] employed the novel fully-convolutional network (FCN) architecture (originally presented by Long *et al.* [9] for scene semantic segmentation) for monocular depth estimation. In [2] the authors used a residual network [10], and refined the results using a conditional random field (CRF) prior, external to the network architecture. Similar approach of using CRF to refine a DL model initial result was also used by Li *et al.* [4]. In [3] a simpler FCN model was proposed, but with the CRF operation integrated inside the network structure. This approach was further researched using deeper networks and more sophisticated architectures [5], [6]. The challenge was also addressed in the unsupervised learning approach, as presented by Garg *et al.* [7] and Godard *et al.* [8].

Common to all these approaches is the use of depth cues in the RGB image 'as-is', as well as having the training and testing on well-known public datasets such as the NYU depth [11], [12] and Make3D [13]. Since the availability of reliable depth cues in a regular RGB image is limited, these approaches require large architectures with significant regularization (Multiscale, ResNets, CRF) as well as separation of the models to indoor/outdoor scenes. A modification of the image acquisition process itself seems necessary in order to allow using a simpler model, generic enough to encompass both indoor and outdoor scenes.

Imaging methods that use an aperture coding mask (both phase or amplitude) became more common in the last two decades. One of the first and prominent studies in this field was carried out by Dowski and Cathey [14], where a cubic phase mask was designed to generate a constant point spread function (PSF) throughout the desired depth of field (DOF). Similar ideas were presented later in [15] using a random diffuser with focal sweep [16], or by using an uncorrected lens as a type of spectral focal sweep [17]. When a depth-independent PSF is achieved, an all-in-focus image can be recovered using non-blind deconvolution methods. However, in all these methods the captured and restored images have a similar response in the entire DOF, and thus depth information can only be recovered to some extent using monocular cues.

In order to generate optical cues, the PSF should be depth-dependent. Related methods use an amplitude coded mask [18], [19] or a color-dependent ring mask [20], [21] such that objects at different depths exhibit a distinctive spatial/spectral

structure. The main drawback of these strategies is that the actual light efficiency is only 50% in [18], [19], 60% in [20] and 80% in [21], making them unsuitable for low light conditions. Moreover, those techniques (except [21]) are based on the same low DOF setup, having a $f = 50mm, f/1.8$ lens (27.8mm aperture). Thus, they are also unsuitable for small-scale cameras since they are less sensitive to small changes in focus.

**Contribution.** In this paper, we propose a novel deep learning framework for the joint design of a phase-coded aperture element and a corresponding FCN model for single-image depth estimation. A similar phase mask has been proposed by Milgrom *et al.* [22] for extended DOF imaging; its major advantage is light efficiency above 95%. Our phase mask is designed to increase sensitivity to small focus changes, thus providing an accurate depth measurement for small-scale cameras (such as smartphone cameras).

In our system, the aperture coding mask is designed for encoding strong depth cues with negligible light throughput loss. The coded image is fed to a FCN, designed to decode the color-coded depth cues in the image, and thus estimate the depth map. The phase mask structure is trained together with the FCN weights, allowing end-to-end system optimization. For training, we created the 'TAU-Agent' dataset[1] containing pairs of high-resolution realistic animation images and their perfectly registered pixel-wise depth maps.

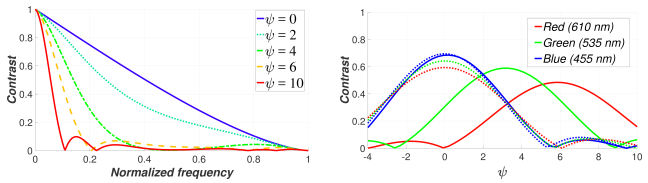
Since the depth cues in the coded image are much stronger than their counterparts in a clear aperture image, the proposed FCN is much simpler and smaller compared to other monocular depth estimation networks. The joint design and processing of the phase mask and the proposed FCN lead to an improved overall performance: better accuracy and faster run-time compared to the known monocular depth estimation methods. Also, the achieved performance is competitive with more complex, cumbersome and higher cost depth estimation solutions such as light-field cameras.

The rest of the paper is organized as follows: Section II presents the phase-coded aperture used for encoding depth cues in the image, and its design process. Section III describes the FCN architecture used for depth estimation and its training process. Experimental results on synthetic data as well as on real images acquired using an optical setup with a manufactured optimal aperture coding mask are presented in Section IV. Our system is shown to exhibit superior performance in depth accuracy, system complexity, run-time and required processing power compared to competing methods. Section V concludes the paper.

## II. Phase-Coded Aperture Imaging for Depth Estimation

The need to acquire high-quality images and videos of moving objects in low-light conditions establish the well-known trade-off between the aperture size (F#) and the DOF in optical imaging systems. With conventional optics, increasing the light efficiency at the expense of reduced DOF poses inherent limitations on any purely computational technique, since the out-of-focus blur may result in information loss in parts of the image.

[1]Dataset is available for download at http://www.tau.ac.il/~harelhai/TAUAgent/home.html or addTheTorontoMirrorUrl.




(a) (b)

Fig. 1. **Spatial frequency response and color channel separation.** (a) Optical system response to normalized spatial frequency for different values of the defocus parameter $\psi$. (b) Comparison between contrast levels for a single normalized spatial frequency (0.25) as a function of $\psi$ for clear aperture (dotted) and when our new trained phase mask is used (solid).

These limitations can be overcome by manipulating the image acquisition process. A recent study by Haim *et al.* [23] used Milgrom's aperture phase coding technique [22] to achieve extended DOF imaging. In [23], the authors proposed a method for utilizing the diversity between color channels (expressed in their respective PSF) to find the corresponding blurring model for each small image patch, and used this model to restore the image. Here we adopt a similar phase mask for depth reconstruction. We show that this mask introduces depth-dependent color cues throughout the scene, which lead to fast and accurate depth estimation. Due to the optical cues based depth estimation, our method generalization ability is better compared to the current monocular depth estimation methods.

### A. Out-of-focus imaging

An imaging system acquiring an out-of-focus (OOF) object can be described analytically using a quadratic phase error in its pupil plane [24]. In the case of a circular exit pupil with radius $R$, the defocus parameter is defined as

$$\begin{aligned}\psi &= \frac{\pi R^2}{\lambda}\left(\frac{1}{z_{\mathrm{o}}}+\frac{1}{z_{\mathrm{img}}}-\frac{1}{f}\right)=\frac{\pi R^2}{\lambda}\left(\frac{1}{z_{\mathrm{img}}}-\frac{1}{z_{\mathrm{i}}}\right)\\ &= \frac{\pi R^2}{\lambda}\left(\frac{1}{z_{\mathrm{o}}}-\frac{1}{z_{\mathrm{n}}}\right),\end{aligned} \tag{1}$$

where $z_{\mathrm{img}}$ is the sensor plane location of an object in the nominal position ($z_{\mathrm{n}}$), $z_{\mathrm{i}}$ is the ideal image plane for an object located at $z_{\mathrm{o}}$ , and $\lambda$ is the optical wavelength. Out-of-focus blur increases with the increase of $|\psi|$; the image exhibits gradually decreasing contrast level that eventually leads to information loss (see Fig. 1(a)).

### B. Mask design

Both Milgrom *et al.* [22] and Haim *et al.* [23] have shown that phase masks with a single radially symmetric ring introduce diversity between the responses of the three major color channels (R, G and B) for different focus scenarios, such that the three channels jointly provide an extended DOF. In order to allow more flexibility in the system design, we use a mask with two or three rings, whereby each ring exhibits a different

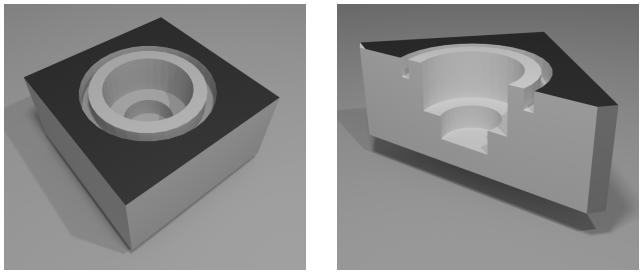
wavelength-dependent phase shift. In order to determine the optimal phase mask parameters within a deep learning-based depth estimation framework, the imaging stage is modeled as the initial layer of a CNN model. The inputs to this coded aperture convolution layer are the all-in-focus images and their corresponding depth maps. The parameters (or weights) of the layer are the radii $r_i$ and phase shifts $\phi_i$ of the mask's rings.

Such layer forward model is composed of the coded aperture PSF calculation (for each depth in the relevant depth range) followed by imaging simulation using the all-in-focus input image and its corresponding depth map. The backward model uses the inputs from the next layer (backpropagated to the coded aperture convolutional layer) and the derivatives of the the coded aperture PSF with respect to its weights, $\partial PSF/\partial r_i$, $\partial PSF/\partial \phi_i$, in order to calculate the gradient descent step on the phase mask parameters. A detailed description of the coded aperture convolution layer and its forward and backward models is presented in the Appendix. One of the important hyper-parameters of such a layer is the depth range under consideration (in $\psi$ terms). The $\psi$ range setting, together with the lens parameters (focal length, F# and focus point) dictates the trade-off between the depth dynamic range and resolution. In this study, we set the range to $\psi = [-4, 10]$; its conversion to a metric depth range is presented in section IV. As mentioned above, the optimization of the phase mask parameters is done by integrating the coded aperture convolutional layer into the CNN model detailed in the sequel, followed by the end-to-end optimization of the entire model. To validate the coded aperture layer, we compared the case where the CNN (described in the following section) is trained end-to-end with the phase coded aperture layer to the case where the phase mask is held fixed to its initial value. Several fixed patterns were examined; the training of the phase mask improved the classification error by 5% to 10%.

For the setup we used, the optimization process yielded a three rings mask such that the outer ring is deeper than the middle one as illustrated in Fig. 2. Such a design poses significant fabrication challenges for the chemical etching process used at our facilities. Since an optimized three-rings mask surpass the two-ring mask only by a small margin, in order to make the fabrication process simpler and more reliable, a two-ring limit was set in the training process; this resulted in the normalized ring radii $r = \{0.55, 0.8, 0.8, 1\}$ and phases $\phi = \{6.2, 12.3\}$ [rad] (both $\psi$ and $\phi$ are defined for the blue wavelength, where the RGB wavelengths taken are the peak wavelengths of the camera color filter response: $\lambda_{R,G,B} = [610, 535, 455]nm$). Figure 1(b) shows the diversity between the color channels for different depths (expressed in $\psi$ values) when using a clear aperture (dotted plot) versus our optimized phase mask (solid plot).

## III. FCN for Depth Estimation

We now turn to describe the architecture of our fully convolutional network (FCN) for depth estimation, which relies on optical depth cues encoded in the image, provided by the phase coded aperture incorporated in the lens as described in Section II. These cues are used by the FCN model to estimate the scene depth map. Our network configuration is inspired by the FCN structure introduced by Long *et al.* [9]. In this work, an ImageNet classification CNN was converted to a semantic segmentation FCN by adding a deconvolution block to the ImageNet model, and fine-tuning it for semantic segmentation (with several architecture variants for increased spatial resolution). For depth estimation using our phase coded aperture camera, a totally different 'inner net' should replace the 'ImageNet model'. The inner net should classify the different imaging conditions (i.e. $\psi$ values), and the deconvolution block will turn the initial pixel labeling into a full depth estimation map. We tested two different 'inner' network architectures: the first based on the DenseNet architecture [25], and the second based on a traditional feed-forward architecture. An FCN based on both inner nets is presented, and the trade-off is discussed. The following subsections present the $\psi$ classification inner nets, and the FCN model based on them for depth estimation.



(a) (b)

Fig. 2. **Aperture phase coding mask.** (a) 3D illustration of the optimal three-ring mask (b) cross-section of the mask. The area marked in black acts as a circular pupil.

### A. $\psi$ classification CNN

As presented in Section II, the phase coded aperture is designed along with the CNN such that it encodes depth-dependent cues in the image by manipulating the response of the RGB channels for each depth. Using these strong optical cues, the depth slices (i.e. $\psi$ values) can be classified using some CNN classification model.

For this task, we tested two different architectures; the first one based on the DenseNet architecture for CIFAR-10, and the second based on the traditional feed-forward architecture of repeated blocks of convolutions, batch normalization [26] and rectified linear units [27] (CONV-BN-ReLU, see Fig. 3). In view of the approach presented in [28], pooling layers are omitted in the second architecture, and stride of size 2 is used in the CONV layers for lateral dimension reduction. This approach allows much faster model evaluation (only 25% of the calculation in each CONV layer), with minor loss in performance.

To reduce the model size and speed up its evaluation even more, the input (in both architectures) to the first CONV layer of the net is the captured raw image (in mosaicked Bayer pattern). By setting the stride of the first CONV layer to 2, the filters' response remains shift-invariant (since the Bayer pattern period is 2). This way the input size is decreased by a

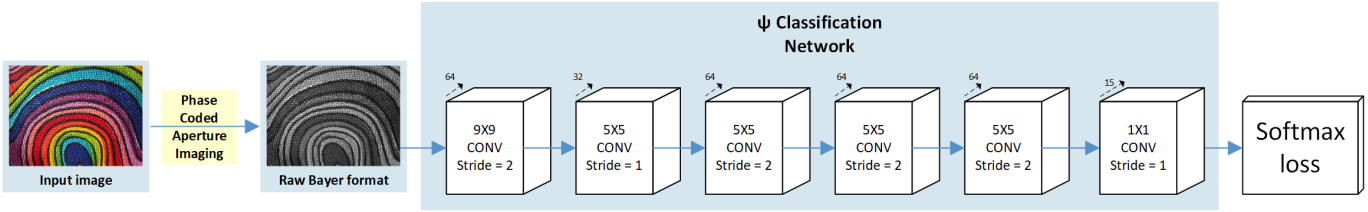



Fig. 3. **Neural network architecture for the depth classification CNN:** (the 'inner' net in the FCN model in Fig. 4). Spatial dimension reduction is achieved by convolution stride instead of pooling layers. Every CONV block is followed by BN-ReLU layer (not shown in this figure).

factor of 3, with minor loss in performance. This also omits the need for a demosaicking stage, allowing faster end-to-end performance (in cases where the RGB image is not needed as an output, and one is interested only in the depth map). One can see the direct processing of mosaicked images as a case where the CNN representation power 'contains' the demosaicking operation, and therefore it is not really needed as a preprocessing step.

Both inner classification net architectures are trained on the Describable Textures Dataset (DTD) [29]. About 40K texture patches (32x32 pixels each) were selected from the dataset. Each patch was 'replicated' in the dataset 15 times, where each replication corresponds to a different blur kernel (corresponding to the phase coded aperture for $\psi = -4, -3, ..., 10$). The first layer of both architectures represents the phase-coded aperture layer, whose inputs are the clean patch and its corresponding $\psi$ value.

After the imaging stage is done (as explained in II), an Additive White Gaussian Noise (AWGN) is added to each patch to make the network robust to the typical noise level that exists in images taken with a real-world camera. Though increasing the noise level improves the robustness, it is important to consider the trade-off that exists between noise robustness and depth estimation accuracy, which limits the amount of noise that should be added in training, making the noise level a hyper-parameter one should tune. In our tests, when we set a specific noise level, the accuracy of the depth results is deteriorated for inputs with higher noise level (as one would expect). At the same time, when we train the CNN with relatively high noise levels, the system becomes more robust to noise at the expense of accuracy reduction for images with lower noise. Therefore, $\sigma = 3$ is chosen as a good compromise since it resembles the noise level of images of a well-lighted scene taken with the selected camera. Of course, one may consider a different noise level, according to the target camera and its expected noise level.

Data augmentation via four rotations is used to increase the dataset size as well as achieving rotation invariance. The dataset size is about 2.4M patches, where 80% of it is used for training and 20% is used for validation. Both architectures were trained to classify into 15 integer values of $\psi$ (between $-4$ and 10) using the softmax loss. These nets are used as an initialization for the depth estimation FCN, as presented in III-C.

### *B. RGB-D Dataset*

The deep learning based methods for depth estimation from a single image mentioned in Section I [1]–[8] rely strongly on the input image details. Thus, most studies in this field assume an input image with a large DOF such that most of the acquired scene is in focus. This assumption is justified when the photos are taken by small aperture cameras as is the case in datasets such as NYU Depth [11], [12] and Make3D [13] that are commonly used for the training and testing of those depth estimation techniques. However, such optical configurations limit the resolution and increase the noise level, thus reducing the image quality. Moreover, the depth maps in these dataset are prone to errors due to depth sensor inaccuracies and calibrations issues (alignment and scaling) with the RGB sensor.

Our method is based on a phase-coded aperture imaging process, which encodes the image. To train or evaluate our method on images not taken with our camera, the phase coded aperture imaging process has to be simulated on those images. To simulate the imaging process properly, the input data should contain high resolution, all in-focus images with low noise, accompanied by accurate pixelwise depth maps. Evaluating depth datasets such as NYU depth [11], [12] and Make3D [13] for our coded aperture imaging simulation is impossible due to the limited image and depth resolution provided in these datasets, which limit the possibility of simulating our image acquisition process on these datasets. Proper input for such imaging simulation may be generated primarily using 3D graphic simulation software.

We use the popular MPI-Sintel depth images dataset [30], created by the Blender 3D graphics software. The Sintel dataset contains 23 scenes with a total of 1k images. Yet, because it has been designed specifically for optical flow evaluation, the depth variation in each scene is limited. Thus, we could only use about 100 unique images, which are not enough for training. The need for additional data has led us to create a new Sintel-like dataset (using Blender) called 'TAU-Agent', which is based on the recent open movie 'Agent 327'. This new animated dataset, which relies on the new render engine 'Cycles', contains 300 realistic images (indoor and outdoor), with a resolution of $1024 \times 512$, and corresponding pixelwise depth maps. With rotations augmentation, our full dataset contained 840 scenes, where 70% are used for training and the rest for validation.

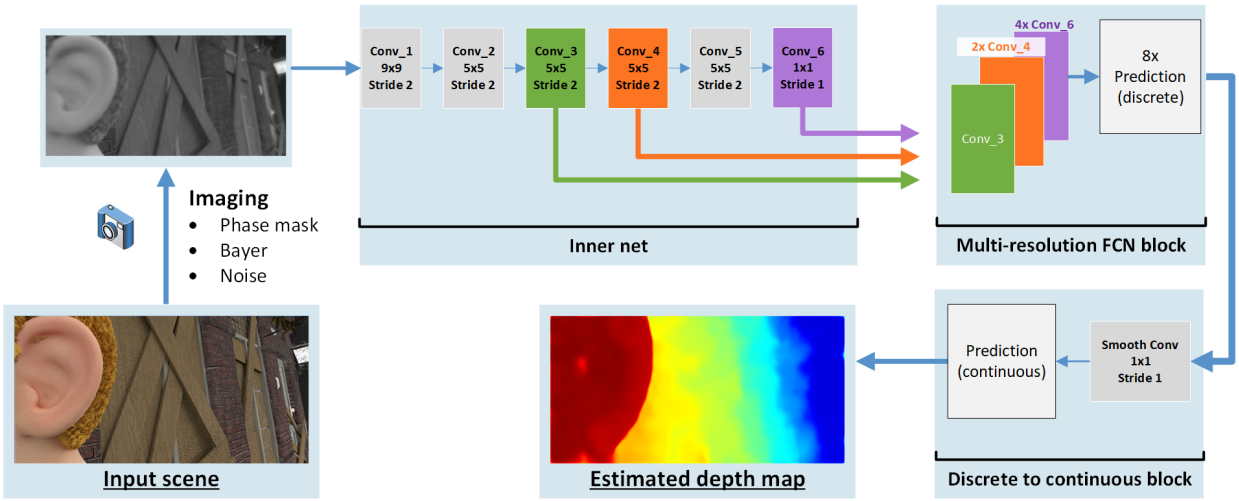
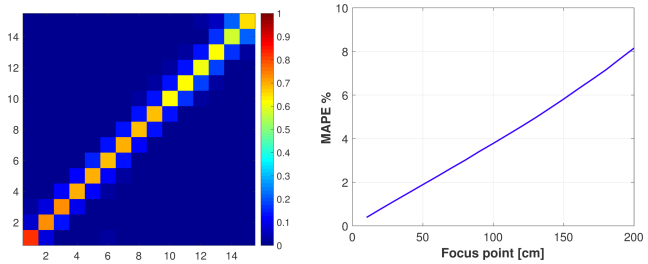



Fig. 4. Network architecture for the depth estimation FCN. The depth ($\psi$) classification network (see Fig. 3) is wrapped in a deconvolution framework to provide depth estimation map equal to the input image size.




(a) Confusion matrix (b) MAPE vs Focus point

Fig. 5. (a) Confusion matrix for the depth segmentation FCN validation set (b) MAPE as a function of the focus point using our continuous net.

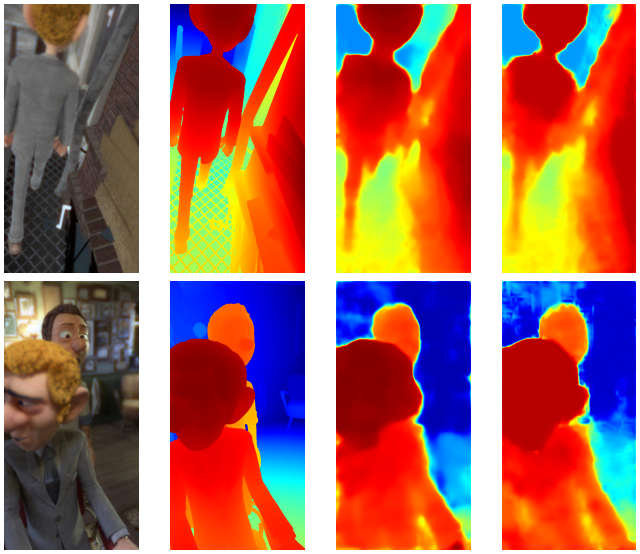
### C. Depth estimation FCN

In similarity to the FCN model presented by Long *et al.* [9], the inner $\psi$ classification net is wrapped in a deconvolution framework, turning it to a FCN model (see Fig. 4). The desired output of our depth estimation FCN is a continuous depth estimation map. However, since training continuous models is prone to over-fitting and regression to the mean issues, we pursue this goal in two stages. In the first one, the FCN is trained for discrete depth estimation. On the second step, the discrete FCN model is used as an initialization for the continuous model training.

To train the discrete depth FCN, the Sintel and Agent datasets RGB images are blurred using the coded aperture imaging model, where each object is blurred using the corresponding blur kernel associated with its depth (indicated in the ground truth pixelwise depth map). The imaging is done in a quasi-continuous way, with $\psi$ step of $0.1$ in the range of $\psi = [-4, 10]$. This imaging simulation can be carried in the same way as the 'inner' net training, i.e. using the phase coded aperture layer as the first layer of the FCN model. However, such step is very computationally demanding, and does not provide significant improvement (since the phase-coded aperture parameters tuning reached its optimum in the inner net training). Therefore, in the FCN training stage, the optical imaging simulation is done as a pre-processing step with the best phase mask achieved in the inner net training stage. In the discrete training step of the FCN, the ground-truth depth maps are discretized to $\psi = -4, -3, .., 10$ values. The Sintel/Agent images (after imaging simulation with the coded aperture blur kernels, RGB-to-Bayer transformation and AWGN addition), along with the discretized depth maps, are used as the input data for the discrete depth estimation FCN model training. The FCN is trained for reconstructing the discrete depth of the input image using softmax loss.

After training, both versions of the FCN model (based on the DenseNet architecture and the traditional feed-forward architecture) achieved roughly the same performance, but with a significant increase in inference time (x3), training time (x5) and memory requirements (x10) for the DenseNet model. When examining the performance, one can see that most of the errors are on smooth/low texture areas of the images, where our method (that relies on texture) is expected to be weaker. Yet, in areas with 'sufficient' texture, there are enough encoded depth cues, enabling good depth estimation even with relatively simple DNN architecture. This similarity in performance between the DenseNet based model (which is one of the best CNN architectures known to date) to a simple feed-forward architecture is a clear example to the inherent power of optical image processing using coded aperture; a task driven design of the image acquisition stage can potentially save significant resources in the digital processing stage. As such, we decided to keep the simple feed-forward architecture as the chosen solution.

To evaluate the discrete depth estimation accuracy, we calculated a confusion matrix for our validation set ($\sim 250$ images, see Fig. 5(a)). After 1500 epochs, the net achieves accuracy of 68% (top-1 error). However, the vast majority of the errors are to adjacent $\psi$ values, and on 93% of the pixels the discrete depth estimation FCN recovers the correct depth with an error of up to $\pm 1\psi$. As already mentioned above, most of the errors originate from smooth areas, where no texture



(a) Input (b) GT (c) L1 (d) L2

Fig. 6. **Depth estimation results on simulated image from the 'Agent' dataset:** (a) original input image (the actual input image used in our net was the raw version of the presented image), (b) Continuous ground truth (c-d) Continuous depth estimation achieved using the L1 loss (c) and the L2 loss (d).

exists and therefore no depth dependent color-cues were encoded. This performance is sufficient as an initialization point for the continuous depth estimation network.

The discrete depth estimation (segmentation) FCN model is upgraded to a continuous depth estimation (regression) model using some modifications. The linear prediction results serve as an input to a $1X1$ CONV layer, initialized with linear regression coefficients from the $\psi$ predictions to continuous $\psi$ values ($\psi$ values can be easily translated to depth values in meters, once lens parameters and focus point are known).

The continuous network is fine-tuned in an end-to-end fashion, with lower learning rate (by a factor of 100) for the pre-trained discrete network layers. The same Sintel & Agent images are used as an input, but with the quasi-continuous depth maps (without discretization) as ground truth, and L2 or L1 loss. After 200 epochs, the model converges to Mean Absolute Difference (MAD) of $0.6\psi$. Again, we found that most of the errors originate from smooth areas (as detailed in Section IV-A hereafter).

## IV. Experimental results and comparison

### A. Validation set results

As a basic sanity check, the validation set images can be inspected visually. In Fig. 6 it can be seen that while the depth cues encoded in the input image are hardly visible to the naked eye, the proposed FCN model achieves quite accurate depth estimation maps compared to the ground truth. Most of the errors are concentrated in smooth areas, as mentioned in Section III-C. The continuous depth estimation smooths the initial discrete depth recovery, achieving a more realistic result.

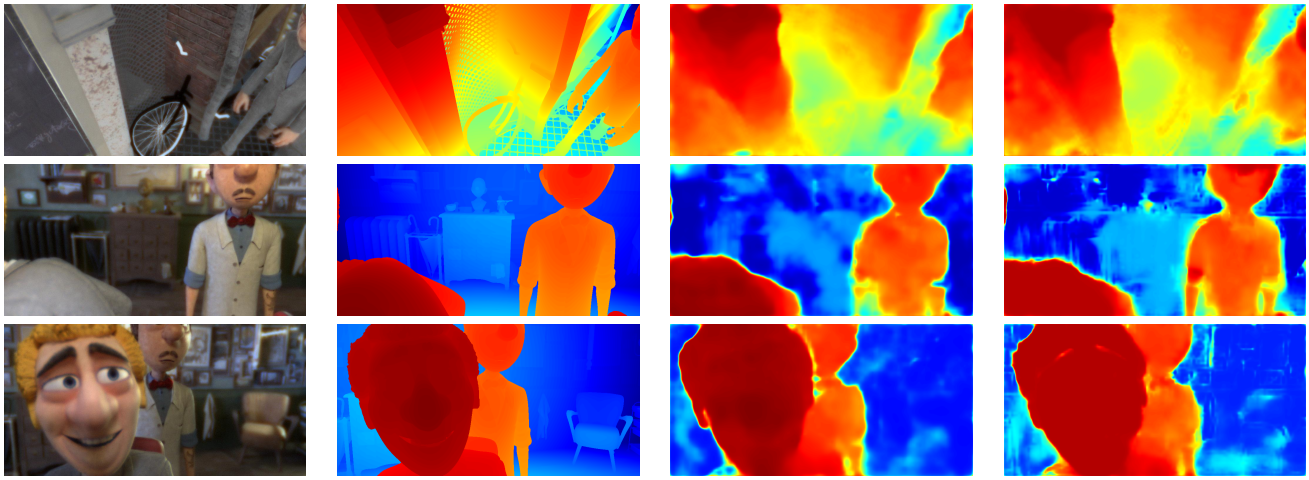
As mentioned above, our method estimates the blur kernel ($\psi$ value), using the optical cues encoded by the phase coded aperture. An important practical analysis is the translation of the $\psi$ estimation map to a metric depth map. Using the lens parameters and the focus point, transforming from $\psi$ to depth is straight-forward (see Section II). Using this transformation, the relative metric depth error can be analyzed. The $\psi = [-4, 10]$ domain is spread to some depth dynamic range, depending on the chosen focus point. Near-by focus point dictates small dynamic range and high depth resolution, and vice versa. However, since the FCN model is designed for $\psi$ estimation, the model (and its $\psi's$ related MAD) remains the same. After translating to metric maps, the Mean Absolute Percentage Error (MAPE) is different for each focus point. Such analysis is presented in Fig. 5(b), where the aperture diameter is set to $2.3[mm]$ and the focus point changes from $0.1[m]$ to $2[m]$, resulting with a working distance of $9[cm]$ to $30[m]$. One can see that the relative error is roughly linear with the focus point, and remains under $10\%$ for relatively wide focus-point range. A summary of the depth estimation performance with several error measures is presented in Table I

Additional simulated scenes examples are presented in Fig. 7. The proposed FCN model achieves accurate depth estimation maps compared to the ground truth. Notice the difference in the estimated maps when using the $L1$ loss (Fig. 7(c)) and the $L2$ loss (Fig. 7(d)). The $L1$ based model produces smoother output but reduces the ability to distinguish between fine details, while the $L2$ model produces noisier output but provides sharper maps. This is illustrated in all scenes where the gap between the body and the hands of the characters is not visible, as observed in Fig. 7(c). Note that in this case the L2 model produces a sharper separation (Fig. 7(d)). The estimated maps in Fig. 7(c-d) also presents a few limitations of our method. In the top row, the fence behind the bike wheel is not visible since the fence wires are too thin. In the middle and bottom rows, the background details are not visible due to low dynamic range in these areas (the background is too far from the camera). One may overcome the dynamic range limitations by changing the aperture size/focus point, as explained in the following.

As mentioned above, our system is designed to handle $\psi$ range of $[-4, 10]$, but the metric range depends on the focus point selection (as presented above). This codependency allows one to use the same FCN model with different optical configurations. To demonstrate this important advantage, we simulated an image (Fig.10(a)) captured with a lens having an aperture of $3.45[mm]$ ($1.5$ the size of our original aperture used for training). The larger aperture provides better metrical accuracy in exchange of reducing the dynamic range. The focus point was set to $48[cm]$, providing a working range of $39[cm]$ to $53[cm]$. We then produced an estimated depth map, which was translated into point cloud data using the camera parameters (sensor size and lens focal length) from Blender. The 3D face reconstruction shown in Fig. 10(b) validates our metrical depth estimation capabilities and demonstrates the efficiency of our strategy as we were able to create this 3D model in real time.

### B. Real-world results

To test the proposed depth estimation method, several experiments were carried. The experimental setup included an



(a) Input (b) GT (c) $L1$ (d) $L2$

Fig. 7. **Depth estimation results on simulated scenes from the 'Agent' dataset:** (a) original input image (the actual input image used in our net was the raw version of the presented image), (b) Continuous ground truth (c-d) Continuous depth estimation achieved by our FCN network when trained using (c) the $L1$ loss and (d) the $L2$ loss.

(a) lens and phase mask (b) indoor scene side view

Fig. 8. **lab setup**

TABLE I
DEPTH ESTIMATION RESULTS SUMMARY

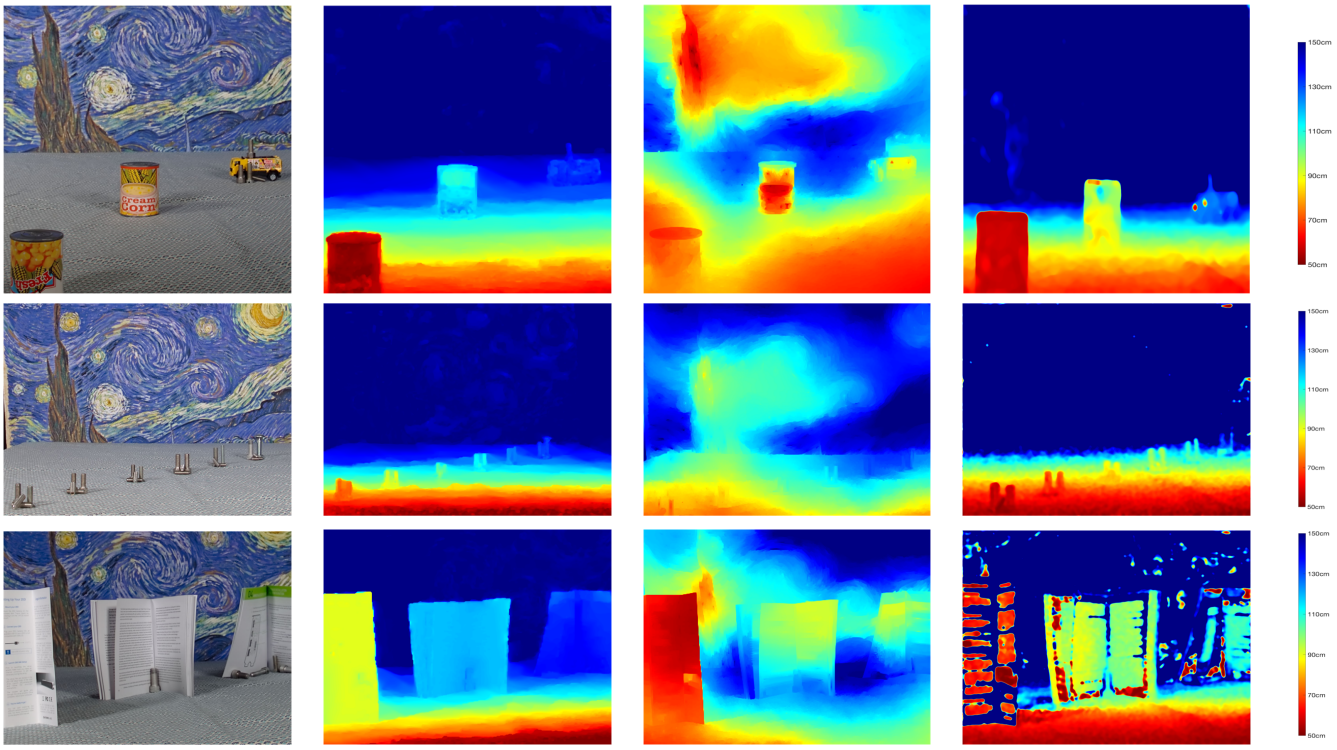
| **Measure** | **Result** |
|---|---|
| Initial discrete depth segmentation | 68% (top-1), 93% ($\pm 1\psi$) |
| Continuous depth estimation error [$\psi$] | 0.6 |
| Val. set- rel. error [m] | 5.5% |
| Val. set- log10 error [m] | 0.056 |
| Val. set- RMS error [m] | 0.12 |
| Experimental scene (spot check, rel.) [m] | 6.25% |
| Run time (Full-HD image) [s] | 0.22 |

$f = 16mm, F/7$ lens (LM16JCM-V by Kowa) with our phase coded aperture incorporated in the aperture stop plane (see Fig.8(a)). The lens was mounted on a UI3590LE camera made by IDS Imaging. The lens was focused to $z_o = 1100mm$, so that the $\psi = [-4, 10]$ domain was spread between $0.5 - 2.2m$. Several scenes were captured using the phase coded aperture camera, and the corresponding depth maps were calculated using the proposed FCN model.

For comparison, two competing solutions were examined on the same scenes: Illum light-field camera (by Lytro), and the monocular depth estimation net proposed by Liu *et al.* [3]. Since the method in [3] assumes an all in-focus image as an input, we used the Lytro camera all in-focus imaging option as the input to [3].

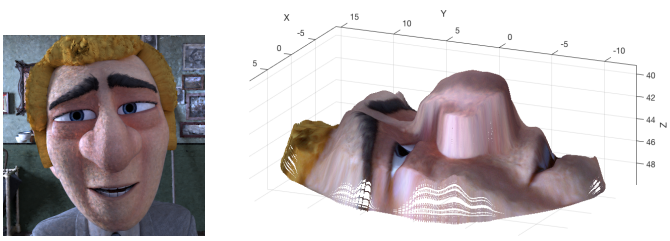
It is important to note that our proposed method provides depth maps in absolute values (meters), while the Lytro camera and [3] provide a relative depth map only (far/near values with respect to the scene). Another advantage of our technique is that it requires the incorporation of a very simple optical element to an existing lens, while light-field and other solutions (like stereo cameras) require a much more complicated optical setup. In the stereo camera, two calibrated and laterally separated cameras are mounted on a rigid base. In the light-field camera, special light-field optics and sensor are used. In both cases the cumbersome optical setup dictates large volume and high cost.

We examined all the solutions on both indoor and outdoor scenes. Several examples are presented, with similar and different focus points. Indoor scenes examples are shown in Fig. 9. Several objects were laid on a table with a poster in the background (see Fig. 8(b) for a side view of the scene). Since the scenes lack global depth cues, the method from [3] fails to estimate a correct depth map. The Lytro camera estimates the gradual depth structure of the scene with good object identification, but provides a relative scale only. Our method succeeds to identify both the gradual depth of the table as well as the fine details of the objects (top row- note the screw located above the truck on the right, middle row- note the various groups of screws). Although some scene texture 'seeps' to our recovered depth map, it causes only a minor error in the depth estimation. A partial failure case appears in the leaflet scene (Fig. 9, bottom row), where our method misses only on texture-less areas. Performance on non-textured areas is the most challenging scenario to our method (since it relies on color-coded cues generated on texture areas), and it is the source for almost all failure cases. In most cases, our net learns to associate non-textured areas with their correct depth using adjacent locations in the scene that happen to have texture and are at similar depth. However, this is not always the case (as can be seen in Fig. 9(d)- bottom), where it fails to do so in the blank white areas. This issue can be resolved using a much deeper network, and it imposes a performance




(a) scene (b) Light-Field (Lytro) (c) monocular (d) ours

Fig. 9. **Indoor scene depth estimation.** Left to right: (a) the scene and its depth map acquired using (b) Lytro Illum camera, (c) Liu *et al.* [3] monocular depth estimation net, (d) our method. As each camera has a different field of view, the images were cropped to achieve roughly the same part of the scene. The depth scale on the right is relevant only for (d). Because the outputs of (b)&(c) provide only a relative depth map (and not absolute as in the case of (d)), their maps were brought manually to the same scale for visualization purposes.




(a) Input image (b) Point cloud map

Fig. 10. **3D face reconstruction**

vs. model complexity trade-off.

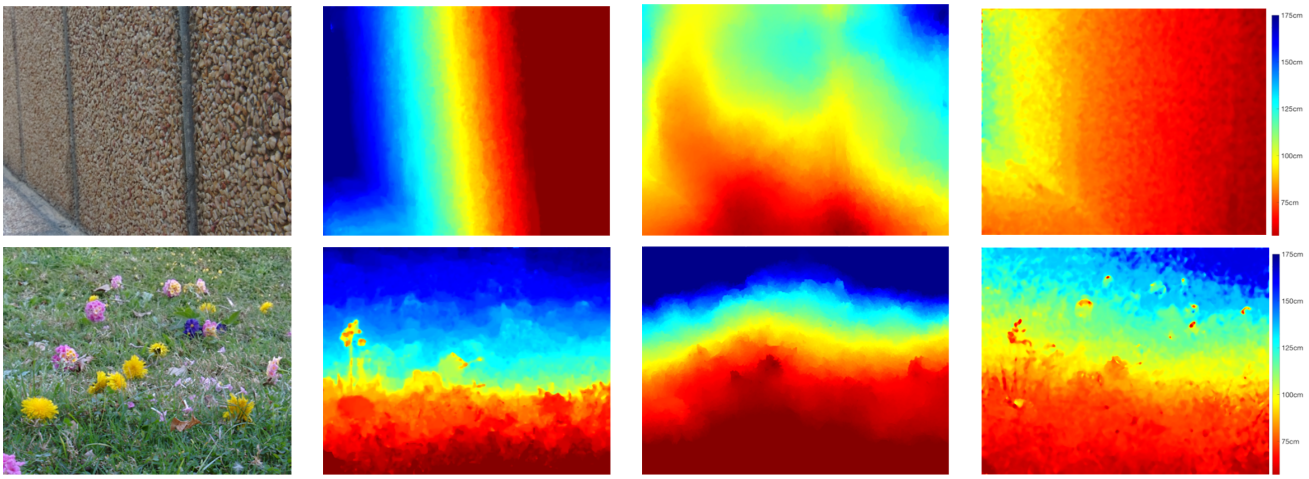
Similar comparison is presented for two outdoor scenes in Fig. 11. On its first row, we chose a scene consisting of a granulated wall. In this example, the global depth cues are also weak, and therefore the monocular depth estimation fails to separate the close vicinity of the wall (right part of the image). Both the Lytro and our phase coded aperture camera achieve good depth estimation of the scene. Note though that our camera has the advantage that it provides an absolute scale and uses much simpler optics.

On the second row of Fig. 11, we chose a grassy slope with flowers. In this case, the global depth cues are stronger. Thus, the monocular method [3] does better compared to the previous examples, but still achieves only a partial depth estimate. Lytro and our camera achieve good results.

Additional outdoor examples are presented in Fig. 12. Note that the scenes in first five rows of Fig. 12 were taken with a different focus point (compared to the indoor and the rest of the outdoor scenes), and therefore the depth dynamic range and resolution are different (as can be seen in the depth scale on the right column). However, since our FCN model is trained for $\psi$ estimation, all depth maps were achieved using the same network, and the absolute depth is calculated using the known focus point and the estimated $\psi$ map.

Quantitative evaluation of the real-world setup with a camera equipped with our phase-coded aperture was performed 'in the wild', since exact depth GT is difficult to acquire in the general case. For quantitative evaluation on real data, we performed a 'spot-check'- we measured the depth recovery error of our network for the known object distances in the lab setting of Fig. 9. We got an average depth estimation error of 6.25%. This accuracy is comparable to Lytro accuracy (Zeller *et al.* [31]) and much better than monocular (25%), while both require a cumbersome calibration phase.

Besides the depth map recovery performance and the simpler optics, another important benefit of our proposed solution is the required processing power/run time. The fact that depth cues are encoded by the phase mask enables much simpler FCN architecture, and therefore much faster inference time. This is due to the fact that some of the processing is done by the optics (in the speed of light, with no processing resources needed). For example, for a full-HD image as an input, our proposed network evaluates a full-HD depth map in $0.22s$ (using Nvidia Titan X Pascal GPU). For the same sized input




(a) scene (b) Light-Field (Lytro) (c) monocular (d) ours

Fig. 11. **Outdoor scenes depth estimation**. Depth estimation results for a granulated wall (upper) and grassy slope with flowers (lower) scenes. Left to right: (a) the scene and its depth map acquired using (b) Lytro Illum camera, (c) Liu *et al.* [3] monocular depth estimation net, (d) our method. As each camera has a different field of view, the images were cropped to achieve roughly the same part of the scene. The depth scale on the right is relevant only for (d). Because the outputs of (b)&(c) provide only a relative depth map (and not absolute as in the case of (d)), their maps were brought manually to the same scale for visualization purposes. Additional examples appear in Fig. 12.

on the same GPU, the net presented in [3] evaluates a 3-times smaller depth map in $10s$ (Timing was measured using the same machine and the implementation of the network available at the authors' website). Of course, if a one-to-one input image to depth map is not needed, the output size can be reduced and our FCN will run even faster.

Another advantage of our method is that the depth estimation relies mostly on local cues in the image. This allows performing of the computations in a distributed manner. The image can be simply split and the depth map can be evaluated in parallel on different resources. The partial outputs can be recombined later with barely visible block artifacts.

## V. Summary and Conclusions

In this paper we have presented a method for real-time depth estimation from a single image using a phase coded aperture camera. The phase mask is designed together with the FCN model using back propagation, which allows capturing images with high light efficiency and color-coded depth cues, such that each color channel responds differently to OOF scenarios. Taking advantage of this coded information, a simple convolutional neural network architecture is proposed to recover the depth map of the captured scene.

This proposed scheme outperforms state-of-the-art monocular depth estimation methods by having better accuracy, more than an order of magnitude speed acceleration, less memory requirements and hardware parallelization compliance. In addition, our simple and low-cost solution shows comparable performance to expensive commercial solutions with complex optics such as the Lytro camera. Moreover, as opposed to the relative depth maps produced by the monocular methods and the Lytro camera, our system provides an absolute (metric) depth estimation, which can be useful to many computer vision applications, such as 3D modeling and augmented reality.

## Appendix
## Phase-coded aperture imaging as a neural network layer

As described in the paper, our depth estimation method is based on a phase-coded aperture lens that introduces depth-dependent color cues in the resultant image. The depth cues are later processed by a Fully-Convolutional Network (FCN) to produce a depth map of the scene. Since the depth estimation is done using deep learning, and in order to have an end-to-end deep learning based solution, we model the phase-coded aperture imaging as a layer in the deep network and optimize its parameters using backpropagation, along with the network weights. In the following we present in detail the forward and backward model of the phase coded aperture layer.

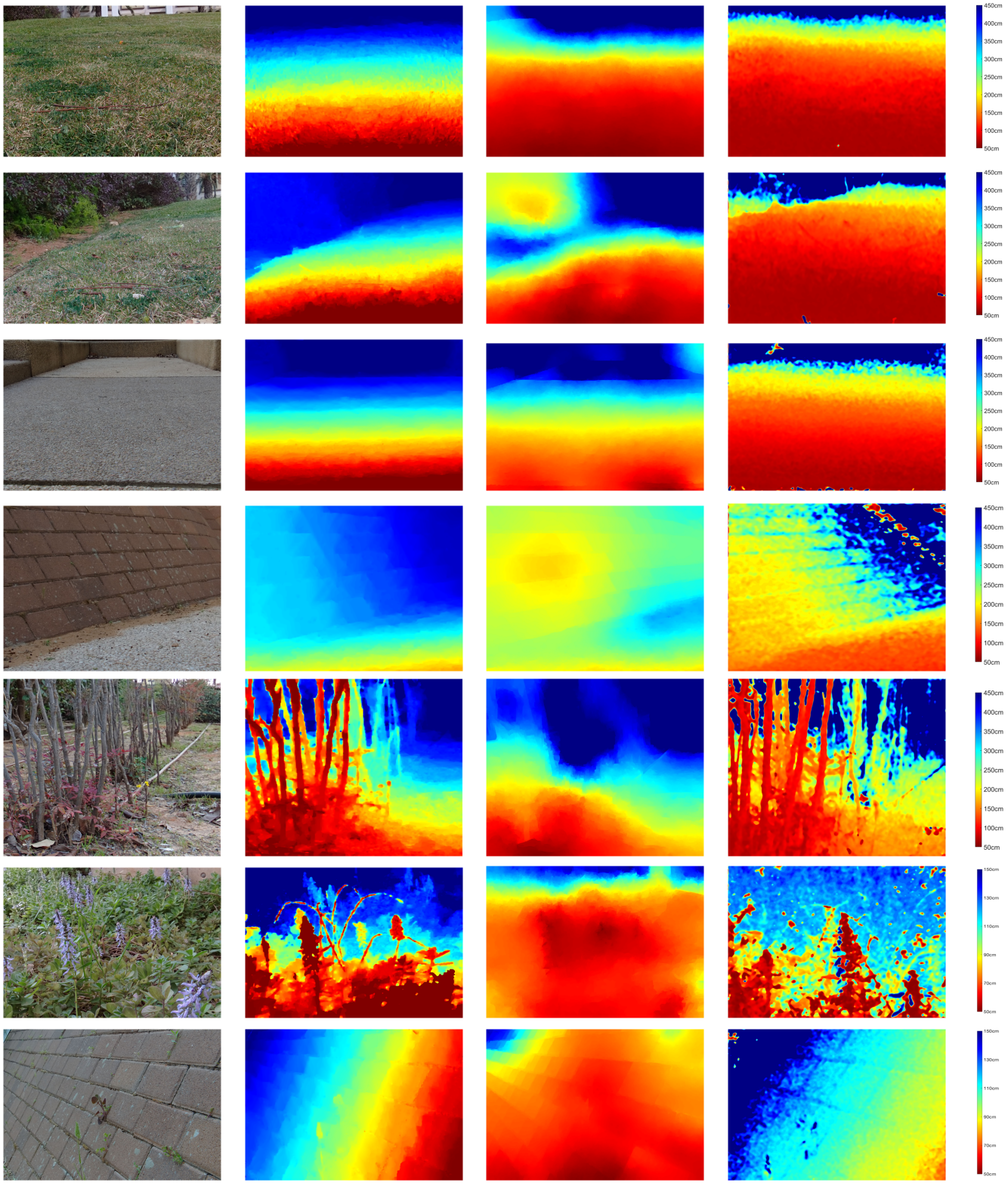
### A. Forward model

Following the imaging system model presented in [24], the physical imaging process is modeled as a convolution of the aberration free geometrical image with the imaging system Point Spread Function (PSF). In other words, the final image is the scaled projection of the scene onto the image plane, convolved with the system's PSF, which contains all the system properties: wave aberrations, chromatic aberrations and diffraction effects.[2] In this model, the PSF calculation contains all the optical properties of the system. Following [24], the PSF of an incoherent imaging system is defined as:

$$PSF = |h_c|^2 = |\mathcal{F}\{P(\rho,\theta)\}|^2, \tag{2}$$

where $h_c$ is the coherent system impulse response, and $P(\rho,\theta)$ is the system's exit pupil function (the amplitude and phase profile in the imaging system exit pupil). The pupil function reference is a perfect spherical wave converging at the image

[2]Note that in this model, the geometric image is a perfect reproduction of the scene (up to scaling), with no resolution limit.




(a) scene (b) Light-Field (Lytro) (c) monocular (d) ours

Fig. 12. **Outdoor scenes depth estimation**. From left to right: (a) the scene and its depth map acquired using (b) Lytro Illum camera, (c) Liu *et al.* [3] monocular depth estimation net, (d) our method. See caption of Fig. 9 for full details.

plane. Thus, for an in-focus and aberration free (or diffraction limited) system, the pupil function is just the identity for the amplitude in the active area of the aperture, and zero for the phase.

**Out-of-Focus (OOF):** An imaging system acquiring an object in OOF conditions suffers from blur that degrades the image quality. This results in low contrast, loss of sharpness and even loss of information. The OOF error is expressed analytically as a quadratic phase wave-front error in the pupil function. To quantify the defocus condition, we introduce the parameter $\psi$. For the case of a circular exit pupil with radius $R$, we define $\psi$ as:

$$\begin{aligned}\psi &= \frac{\pi R^2}{\lambda}\left(\frac{1}{z_{\mathrm{o}}}+\frac{1}{z_{\mathrm{img}}}-\frac{1}{f}\right)=\frac{\pi R^2}{\lambda}\left(\frac{1}{z_{\mathrm{img}}}-\frac{1}{z_{\mathrm{i}}}\right)\\ &=\frac{\pi R^2}{\lambda}\left(\frac{1}{z_{\mathrm{o}}}-\frac{1}{z_{\mathrm{n}}}\right),\end{aligned} \tag{3}$$

where $z_{\mathrm{img}}$ is the image distance (or sensor plane location) of an object in the nominal position $z_{\mathrm{n}}$, $z_{\mathrm{i}}$ is the ideal image plane for an object located at $z_{\mathrm{o}}$, and $\lambda$ is the illumination wavelength. The defocus parameter $\psi$ measures the maximum quadratic phase error at the aperture edge. For a circular pupil:

$$P_{OOF} = P(\rho,\theta)\exp\{j\psi\rho^2\}, \tag{4}$$

where $P_{OOF}$ is the OOF pupil function, $P(\rho,\theta)$ is the in-focus pupil function, and $\rho$ is the normalized pupil coordinate.

**Aperture Coding:** As mentioned above, the pupil function represents the amplitude and phase profile in the imaging system exit pupil. Therefore, by adding a coded pattern (amplitude, phase or both) at the exit pupil,[3] the PSF of the system can be manipulated by some pre-designed pattern. In this case, the pupil function can be expressed as:

$$P_{CA} = P(\rho,\theta)CA(\rho,\theta), \tag{5}$$

where $P_{CA}$ is the coded aperture pupil function, $P(\rho,\theta)$ is the in-focus pupil function, and $CA(\rho,\theta)$ is the aperture/phase mask function. In our case of phase coded aperture, $CA(\rho,\theta)$ is a circularly symmetric piece-wise constant function representing the phase rings pattern. For the sake of simplicity, we will consider a single ring phase mask, applying a $\phi$ phase shift in a ring starting at $r_1$ to $r_2$. Therefore, $CA(\rho,\theta) = CA(\mathbf{r},\phi)$ where:

$$CA(\mathbf{r},\phi) = \begin{cases}\exp\{j\phi\} & r_1<\rho<r_2\\ 1 & otherwise\end{cases} \tag{6}$$

This example can be easily extended to a multiple rings pattern.

**Depth dependent coded PSF:** Combining all factors, the complete term for the depth dependent coded pupil function becomes:

$$P(\psi) = P(\rho,\theta)CA(\mathbf{r},\phi)\exp\{j\psi\rho^2\}. \tag{7}$$

[3]The exit pupil is not always accessible. Therefore, the mask may be added also in the aperture stop, entrance pupil, or in any other surface conjugate to the exit pupil.

Using the definition in (2), the depth dependent coded $PSF(\psi)$ can be easily calculated.

**Imaging Output:** Using the coded aperture PSF, the imaging output can be calculated simply by:

$$I_{out} = I_{in} * PSF(\psi). \tag{8}$$

This model limits us to a Linear Shift-Invariant (LSI) model. However, this is not the case in real imaging systems, and the PSF varies across the Field of View (FOV). This is solved by segmenting the FOV to blocks with similar PSF, and then applying the LSI model in each block.

### *B. Backward model*

As described in the previous subsection, the forward model of the phase coded aperture layer is expressed as:

$$I_{out} = I_{in} * PSF(\psi). \tag{9}$$

The $PSF(\psi)$ varies with the depth ($\psi$), but it has also a constant dependence on the phase ring pattern parameters $\mathbf{r}$ and $\phi$, as expressed in (7). In the network training process, we are interested in determining both $\mathbf{r}$ and $\phi$. Therefore, we need to evaluate three separate derivatives: $\partial I_{out}/\partial r_i$ for $i=1,2$ (the inner and outer radius of the phase ring, as detailed in (6)) and $\partial I_{out}/\partial\phi$. All three are derived in a similar fashion:

$$\begin{aligned}\frac{\partial I_{out}}{\partial r_i/\phi} &= \frac{\partial}{\partial r_i/\phi}[I_{in} * PSF(\psi,\mathbf{r},\phi)]\\ &= I_{in} * \frac{\partial}{\partial r_i/\phi}PSF(\psi,\mathbf{r},\phi)\end{aligned} \tag{10}$$

Thus, we need to calculate $\partial PSF/\partial r_i$ and $\partial PSF/\partial\phi$. Since both derivatives are almost similar, we start with $\partial PSF/\partial\phi$ and then describe the differences in the derivation of $\partial PSF/\partial r_i$ later. Using (2), we get

$$\begin{aligned}\frac{\partial}{\partial\phi}PSF(\psi,\mathbf{r},\phi) &= \frac{\partial}{\partial\phi}[\mathcal{F}\{P(\psi,\mathbf{r},\phi)\overline{\mathcal{F}\{P(\psi,\mathbf{r},\phi)\}}]\\ &= [\frac{\partial}{\partial\phi}\mathcal{F}\{P(\psi,\mathbf{r},\phi)]\overline{\mathcal{F}\{P(\psi,\mathbf{r},\phi)\}}+\\ &+\mathcal{F}\{P(\psi,\mathbf{r},\phi)[\frac{\partial}{\partial\phi}\overline{\mathcal{F}\{P(\psi,\mathbf{r},\phi)\}}]\end{aligned} \tag{11}$$

We may see that the main term in (11) is $\frac{\partial}{\partial\phi}[\mathcal{F}\{P(\psi,\mathbf{r},\phi)]$ or its complex conjugate. Due to the linearity of the derivative and the Fourier transform, the order of operations can be reversed and rewritten as: $\mathcal{F}\{\frac{\partial}{\partial\phi}P(\psi,\mathbf{r},\phi)\}$. Therefore, the last term remaining for calculating the PSF derivative is:

$$\begin{aligned}\frac{\partial}{\partial\phi}P(\psi,\mathbf{r},\phi) &= \frac{\partial}{\partial\phi}[P(\rho,\theta)CA(\mathbf{r},\phi)\exp\{j\psi\rho^2\}]\\ &= P(\rho,\theta)\exp\{j\psi\rho^2\}\frac{\partial}{\partial\phi}[CA(\mathbf{r},\phi)]\\ &= \begin{cases}jP(\psi,\mathbf{r},\phi) & r_1<\rho<r_2\\ 0 & otherwise\end{cases}\end{aligned} \tag{12}$$

Similar to the derivation of $\partial PSF/\partial\phi$, for calculating $\partial PSF/\partial r_i$ we need also $\frac{\partial}{\partial r_i}P(\psi,\mathbf{r},\phi)$. Similar to (12), we have

$$\begin{aligned}\frac{\partial}{\partial r_i}P(\psi,\mathbf{r},\phi) &= \frac{\partial}{\partial r_i}[P(\rho,\theta)CA(\mathbf{r},\phi)\exp\{j\psi\rho^2\}] \\ &= P(\rho,\theta)\exp\{j\psi\rho^2\}\frac{\partial}{\partial r_i}[CA(\mathbf{r},\phi)]\end{aligned} \tag{13}$$

Since the ring radius is a step function, this derivative has to be approximated. We found that $\tanh(100\rho)$ achieves good enough results for the phase step approximation.

With the full forward and backward model, the phase coded aperture layer can be incorporated as a part of the FCN model, and the phase mask parameters $\mathbf{r}$ and $\phi$ can be learned along with the network weights.

## ACKNOWLEDGMENT

This research was partially supported by ERC-StG SPADE PI Giryes and ERC-StG RAPID PI Bronstein. The authors are grateful to NVIDIA's hardware grant for the donation of the Titan X that was used in this research.

## REFERENCES

[1] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 2366–2374. [Online]. Available: http://papers.nips.cc/paper/5539-depth-map-prediction-from-a-single-image-using-a-multi-scale-deep-network.pdf

[2] Y. Cao, Z. Wu, and C. Shen, "Estimating depth from monocular images as classification using deep fully convolutional residual networks," *CoRR*, vol. abs/1605.02305, 2016. [Online]. Available: http://arxiv.org/abs/1605.02305

[3] F. Liu, C. Shen, G. Lin, and I. Reid, "Learning depth from single monocular images using deep convolutional neural fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2015.2505283

[4] B. Li, C. Shen, Y. Dai, A. van den Hengel, and M. He, "Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[5] H. Jung and K. Sohn, "Single image depth estimation with integration of parametric learning and non-parametric sampling," *Journal of Korea Multimedia Society*, vol. 9, no. 9, Sep 2016. [Online]. Available: http://dx.doi.org/10.9717/kmms.2016.19.9.1659

[6] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," *CoRR*, vol. abs/1606.00373, 2016. [Online]. Available: http://arxiv.org/abs/1606.00373

[7] R. Garg, B. G. V. Kumar, G. Carneiro, and I. D. Reid, "Unsupervised CNN for single view depth estimation: Geometry to the rescue," in *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII*, 2016, pp. 740–756.

[8] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[9] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *CVPR*, Nov. 2015.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

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

[12] N. Silberman and R. Fergus, "Indoor scene segmentation using a structured light sensor," in *Proceedings of the International Conference on Computer Vision - Workshop on 3D Representation and Recognition*, 2011.

[13] A. Saxena, M. Sun, and A. Y. Ng, "Make3d: Learning 3d scene structure from a single still image," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 824–840, May 2009. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2008.132

[14] E. R. Dowski and W. T. Cathey, "Extended depth of field through wave-front coding," *Applied Optics*, vol. 34, no. 11, pp. 1859–1866, 1995.

[15] O. Cossairt, C. Zhou, and S. Nayar, "Diffusion coded photography for extended depth of field," in *ACM Transactions on Graphics (TOG)*, vol. 29, no. 4. ACM, 2010, p. 31.

[16] H. Nagahara, S. Kuthirummal, C. Zhou, and S. K. Nayar, "Flexible depth of field photography," in *Computer Vision–ECCV 2008*. Springer, 2008, pp. 60–73.

[17] O. Cossairt and S. Nayar, "Spectral focal sweep: Extended depth of field from chromatic aberrations," in *Computational Photography (ICCP), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1–8.

[18] A. Levin, R. Fergus, F. Durand, and W. T. Freeman, "Image and depth from a conventional camera with a coded aperture," *ACM Transactions on Graphics*, vol. 26, no. 3, p. 70, 2007.

[19] P. A. Shedligeri, S. Mohan, and K. Mitra, "Data driven coded aperture design for depth recovery," *CoRR*, vol. abs/1705.10021, 2017. [Online]. Available: http://arxiv.org/abs/1705.10021

[20] A. Chakrabarti and T. Zickler, "Depth and deblurring from a spectrally-varying depth-of-field," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 648–661.

[21] M. Martinello, A. Wajs, S. Quan, H. Lee, C. Lim, T. Woo, W. Lee, S.-S. Kim, and D. Lee, "Dual aperture photography: Image and depth from a mobile camera," 04 2015.

[22] B. Milgrom, N. Konforti, M. A. Golub, and E. Marom, "Novel approach for extending the depth of field of barcode decoders by using rgb channels of information," *Optics express*, vol. 18, no. 16, pp. 17 027–17 039, 2010.

[23] H. Haim, A. Bronstein, and E. Marom, "Computational multi-focus imaging combining sparse model with color dependent phase mask," *Opt. Express*, vol. 23, no. 19, pp. 24 547–24 556, Sep 2015. [Online]. Available: http://www.opticsexpress.org/abstract.cfm?URI=oe-23-19-24547

[24] J. Goodman, *Introduction to Fourier Optics*, 2nd ed. MaGraw-Hill, 1996.

[25] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[26] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, D. Blei and F. Bach, Eds. JMLR Workshop and Conference Proceedings, 2015, pp. 448–456. [Online]. Available: http://jmlr.org/proceedings/papers/v37/ioffe15.pdf

[27] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[28] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. A. Riedmiller, "Striving for simplicity: The all convolutional net." *CoRR*, vol. abs/1412.6806, 2014. [Online]. Available: http://dblp.uni-trier.de/db/journals/corr/corr1412.html\#SpringenbergDBR14

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

[30] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, "A naturalistic open source movie for optical flow evaluation," in *European Conf. on Computer Vision (ECCV)*, ser. Part IV, LNCS 7577, A. Fitzgibbon et al. (Eds.), Ed. Springer-Verlag, Oct. 2012, pp. 611–625.

[31] N. Zeller, C. A. Noury, F. Quint, C. Teulière, U. Stilla, and M. Dhome, "Metric calibration of a focused plenoptic camera based on a 3d calibration target," *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. III-3, pp. 449–456, 2016. [Online]. Available: https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/III-3/449/2016/