

# Learning to Deblur using Light Field Generated and Real Defocus Images

Lingyan Ruan<sup>1\*</sup> Bin Chen<sup>2\*</sup> Jizhou Li<sup>3</sup> Miuling Lam<sup>1†</sup>

<sup>1</sup>City University of Hong Kong <sup>2</sup>Max-Planck-Institut für Informatik <sup>3</sup>Stanford University

<http://lyruan.com/Projects/DRBNet>

## Abstract

*Defocus deblurring is a challenging task due to the spatially varying nature of defocus blur. While deep learning approach shows great promise in solving image restoration problems, defocus deblurring demands accurate training data that consists of all-in-focus and defocus image pairs, which is difficult to collect. Naive two-shot capturing cannot achieve pixel-wise correspondence between the defocused and all-in-focus image pairs. Synthetic aperture of light fields is suggested to be a more reliable way to generate accurate image pairs. However, the defocus blur generated from light field data is different from that of the images captured with a traditional digital camera. In this paper, we propose a novel deep defocus deblurring network that leverages the strength and overcomes the shortcoming of light fields. We first train the network on a light field-generated dataset for its highly accurate image correspondence. Then, we fine-tune the network using feature loss on another dataset collected by the two-shot method to alleviate the differences between the defocus blur exists in the two domains. This strategy is proved to be highly effective and able to achieve the state-of-the-art performance both quantitatively and qualitatively on multiple test sets. Extensive ablation studies have been conducted to analyze the effect of each network module to the final performance.*

## 1. Introduction

The use of large camera aperture can increase the luminous flux so that the image can be captured with a shorter exposure time. However, this also reduces the depth of field (DOF) - only points near the focal plane can be captured sharply, while a point far from the focal plane will cast to the camera sensor, instead of a single image point, a spot called the Circle of Confusion (COC) [25] and result in defocus blur. Shallow DOF is sometimes an aesthetic effect pursued sedulously by the photographer [7, 30], but it may also



Figure 1. Cross correlation between defocused and all-in-focus image pair in DPDD [2] (top) and LFDOF dataset [29] (bottom).

degrade important visual information. Thus, restoring an all-in-focus image from its defocused version is highly demanded to reveal the latent information and benefit to artificial intelligence applications such as object detection [26] and text recognition [19]. Despite its great potential, defocus deblurring remains a challenging problem due to its spatially varying nature - every point has its own diameter of COC depending on the depth of the corresponding scene point. Besides, the shape of COC varies with respect to the relative position from the optical axis. To address defocus blur, the most intuitive way is to first estimate the blur kernel for each pixel, then apply non-blind deconvolution [5, 12, 15, 17, 24, 32]. However, both steps have limitations. First, blur kernel estimation is not accurate and is often based on simple Gaussian [12, 15, 24] or disk kernel [5] assumption. Second, deconvolution tends to introduce ringing artifact on edges due to the Gibbs phenomenon [41] even if an accurate blur kernel is given.

Recently, researchers have adopted end-to-end deep neural networks to directly restore sharp images from defocus blur [2, 16, 36], which largely outperform the conventional two-step approaches in terms of performance and efficiency. These networks are all trained on a dataset called Dual-Pixel Defocus Deblurring (DPDD) [2] which is captured sequentially with different aperture sizes to attain defocused and all-in-focus image pairs. However, it is hardly possible to capture defocused and all-in-focus pairs with accurate correspondence in two shots, especially for outdoor scenes due to moving objects (*e.g.*, plants, cars) and illuminance variation. To this end, another dataset LFDOF [29] is built utilizing the benefit of light field refocusing and synthetic aper-

\* denotes equal contribution and † denotes corresponding author.

ture to generate a large number of defocused images with a variety of DOFs and focal distances from a single light field sample. To examine the consistency among the image pairs, we select similar scenes from the two datasets and calculate the cross correlation between defocused and all-in-focus pairs. As shown in Fig 1, LFDOF has strong cross correlation in the sharp regions, whereas DPDD does not hold consistence even at the sharp regions (the tree trunk is in focus in both defocused and all-in-focus images). However, despite good pixel-wise consistency of LFDOF, the defocus blur generated from light field data is not the same as that captured with a conventional digital camera (see Sec. 3). In this paper, we intend to make full use of the advantages of LFDOF and DPDD datasets to train a deep network for defocus deblurring. In summary, the contributions of this paper are as follows:

- We analyze the characteristics of two defocus blur datasets LFDOF and DPDD and develop a novel training strategy for single image defocus deblurring. We also estimate and compare the Point Spread Function (PSF) of light field generated defocus blur against that captured with a conventional digital camera.
- We propose an end-to-end network architecture equipped with a novel dynamic residual block to reconstruct the sharp image in a coarse-to-fine manner.
- We conduct extensive experiments to evaluate the effect of each network module and demonstrate the state-of-the-art performance quantitatively and qualitatively on multiple test sets.

## 2. Related Work

**Conventional Methods** Conventional defocus deblurring methods typically follow a two-step approach consisting of defocus map (blur level for each pixel) estimation [12, 37, 46] followed by non-blind deconvolution [6, 14, 17]. Much effort has been made to improve the accuracy of defocus map as it influences the deblurring performance significantly [12, 24, 32, 32]. However, this approach usually requires intensive computation while ending up with limited performance due to the defective intermediate defocus map.

**Defocus Blur Dataset** There are several publicly available datasets for problems related to defocus blur. Defocus blur datasets comprising real defocused RGB images and binary masks are built by Shi *et al.* [32] and Zhao *et al.* [43]. But they can only work for blur detection due to the absence of all-in-focus images. Abuolaim and Brown [2] built the DPDD dataset using a dual-pixel camera and capturing the defocus and all-in-focus pairs in two successive shots. Ruan *et al.* [29] proposed the light field based defocus deblurring dataset LFDOF leveraging the synthetic aperture and refocusing features of light field technology [22], where the im-

age pairs were acquired in single shots. Lee *et al.* [16] provided a benchmark test set consisting of 50 scenes captured with a dual-camera system with beam splitter. Adopting existing RGBD dataset [28, 33] to generate defocus images based on the depth maps, Lee *et al.* [15] have simulated single defocused images while Abuolaim *et al.* [3] and Pan *et al.* [23] have simulated dual defocused pairs. However, the images in these datasets are synthetic, thus lacking realism. In this paper, we use the DPDD and LFDOF datasets that are captured in real-world scenes.

**CNN-based Methods** Abuolaim and Brown [2] adopted an U-Net-like architecture to restore the sharp image in an end-to-end manner and their follow-up work [1] incorporated a single-encoder multi-decoder architecture to further improve the performance. Lee *et al.* [16] proposed a network equipped with iterative filter adaptive module to tackle the spatially varying defocus blur and auxiliary reblurring module to enhance the restoration performance. Son *et al.* [36] proposed an effective way to simulate inverse kernels via kernel sharing parallel atrous convolution block. The aforementioned networks are all trained on DPDD dataset [2]. Ruan *et al.* [29] addressed single image defocus deblurring based on the conventional two-step strategy that takes the intermediate defocus map as the guidance for the deblurring step. Some other works used dual views to address the defocus deblurring problem [23, 38]. However, we focus on single image defocus deblurring in this paper.

**Dynamic Filtering** Dynamic filtering, also called filter adaptive convolution, has been successfully adopted in various low level vision tasks, such as denoising [21], video deblurring [44], super-resolution [10, 13, 40] and defocus deblurring [16], *etc.*, since first introduced by Jia *et al.* [9]. It aims to learn per-pixel kernels instead of a single kernel over the entire image and thus is capable of handling non-uniform or spatially varying degradation. Our proposed dynamic residual block is partially inspired by dynamic filtering, but instead of directly applying convolution with the learned per-pixel kernels to get the output, we treat them as the learned dynamic residuals along with the direct residual for better performance. Also, being different from the method of Lee *et al.* [16] which transforms the features with the learned kernels to remove the spatially varying blur in the feature space, we learn the dynamic residual in the image domain in a coarse-to-fine manner to restore the sharp image. Section 5.3 demonstrates the superior performance of our proposed method against existing works in addressing single image defocus deblurring.

## 3. Conventional Digital Camera vs. Light Field Camera

We begin with describing the defocus blur discrepancy between the images captured with conventional digital cam-

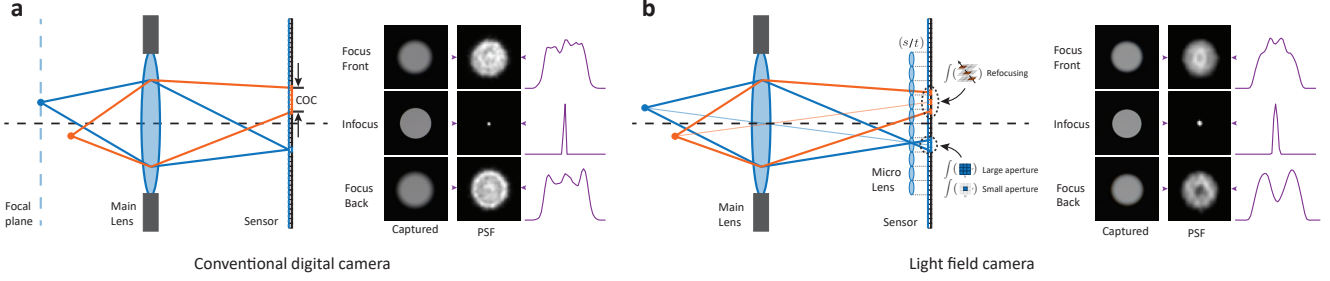


Figure 2. The defocus blur formation process of: (a) conventional digital camera (Canon EOS R5), and (b) Light field camera (Lytro Illum). The captured disk pattern, estimated PSF, and center line profile of PSF image are shown alongside the imaging optical path diagram. The diagrams of light field synthetic aperture and refocusing process are shown in (b).

eras (Canon EOS R5) and light field cameras (Lytro Illum) respectively as shown in Fig. 2.

For a conventional digital camera (Fig. 2a), the rays emitted from a scene point on the focal plane converge to a single pixel of the image sensor by the main lens, while a point away from the focal plane projects to a patch of pixels on the sensor in a circular shape (COC), causing defocus blur. For a light field camera, a micro lens array is placed in front of the sensor, thus the rays coming from the main lens are re-distributed to the pixels under micro lenses, which means that each pixel does not only record the integrated illuminance but also the directional information of the rays. Each sub-aperture view of a light field only records a small part ( $\frac{1}{14 \times 14}$  for Lytro Illum) of the full aperture. After capturing, the aperture size can be further synthesized by integrating an appropriate subset of samples from multiple sub-aperture views. Similarly, refocusing to different depth can be achieved by integrating pixels along different directions on the epipolar plane image (EPI) [4] as shown in Fig. 2b.

To understand the differences between the defocus blur produced by these two types of cameras, we estimate and visualize their PSFs using the algorithm proposed by Manan and Langer [20]. The PSFs of three typical cases: front focus, in focus, and back focus are estimated and shown alongside the imaging optical path diagram in Fig. 2. Generally, the PSFs produced by the digital camera follows the diffraction pattern of single Airy disk, while the PSFs produced by the light field camera resemble the patterns of multiple Airy disks, which can be explained by the synthetic nature of light field generated defocus blur. More PSF estimations can be found in the supplementary materials.

## 4. Methodology

Single image defocus deblurring aims to recover the latent sharp image  $\hat{y}$  from an observed input  $x$  that is distorted by defocus blur. A deep network can be trained as a mapping function  $\mathcal{F}$  parameterized by  $\theta$ :

$$\hat{y} = \mathcal{F}_\theta(x) \quad (1)$$

A loss function should be tailored to optimize  $\theta$  in order to minimize the distance between  $\hat{y}$  and  $y$ :

$$\hat{\theta} = \arg \min_{\theta} \sum_i \mathcal{L}(\mathcal{F}_\theta(x_i), y_i), \quad (2)$$

where  $(x_i, y_i)$  are defocused and all-in-focus image pairs. As shown in Fig. 3, we design our network  $\mathcal{F}$  in an encoder-decoder [27] structure. The encoder ( $\mathcal{E}$ ) extracts the multi-scale pyramidal features, which are then added to the corresponding scale of decoder ( $\mathcal{D}$ ) by skip connections to stabilize the network training. Two residual blocks [8] are added to each scale of decoder. We will show in Sec. 5.3 that simple encoder-decoder structure cannot handle defocus deblurring well.

**Dynamic Residual Block (DRB)** Inspired by the dynamic filtering approach [9, 10, 16, 34, 40], we design a residual version of dynamic filtering block  $\mathcal{R}$  to better handle the spatially varying defocus blur. We connect the dynamic residual block to each scale of decoder in a cascaded fashion to restore the latent sharp image progressively. Each dynamic residual block can be formulated as:

$$\hat{y}_{\downarrow n} = \mathcal{R}(\hat{x}_{\downarrow n}, d_{\downarrow n}; \theta_r), \quad (3)$$

where  $\hat{x}_{\downarrow n}$  represents one of the input of  $\mathcal{R}$ , which is also the  $\times 2$  up-sampled version of the output from the previous dynamic residual block:  $\hat{x}_{\downarrow n} = \uparrow_2(\hat{y}_{\downarrow \frac{n}{2}})$ . Note that the input of the first dynamic residual block is  $x_{\downarrow 8}$ , which is a direct down-sample from the input image  $x$ . Specifically, as illustrated in the green inset in Fig. 3, the input  $\hat{x}_{\downarrow n}$  is passed to three convolution layers and concatenated with the equivalent size feature map  $d_{\downarrow n}$  from decoder  $\mathcal{D}$ , then sent to two paths: one is to estimate the dynamic kernel volume  $\mathbf{K}$  and another is to estimate the residual  $\Delta r_{\downarrow n}$ . The estimated dynamic kernel volume  $\mathbf{K}$  is then convolved with the input  $\hat{x}_{\downarrow n}$  to obtain the dynamic residual  $\Delta \hat{x}_{\downarrow n}$ :

$$\Delta \hat{x}_{\downarrow n} = \hat{x}_{\downarrow n} \otimes \mathbf{K} \quad (4)$$

The dynamic filtering procedure is depicted within the gray inset in Fig. 3. Finally the output of each dynamic residual

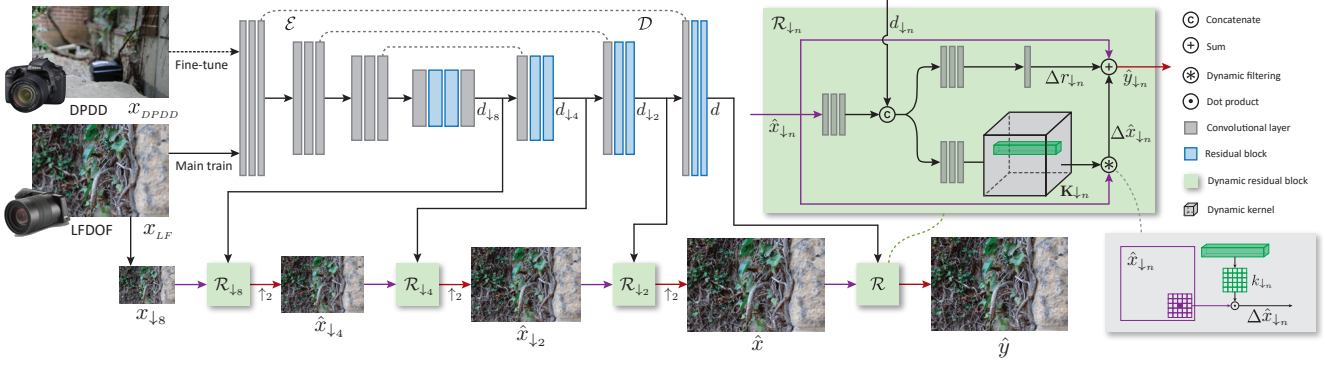


Figure 3. Our network architecture is mainly composed of encoder  $\mathcal{E}$ , decoder  $\mathcal{D}$  and four dynamic residual modules  $\mathcal{R}$ . The LFDOF dataset is used for main training and the DPDD dataset is used for network fine-tuning.

block can be calculated by:

$$\hat{y}_{\downarrow n} = \hat{x}_{\downarrow n} + \Delta r_{\downarrow n} + \Delta \hat{x}_{\downarrow n}, \quad (5)$$

We visualize a small patch in each step of our four dynamic residual blocks in Fig. 4, which clearly shows how the latent sharp image is reconstructed step by step from its defocused version. Specifically, the dynamic residual  $\Delta \hat{x}_{\downarrow n}$  extracts the high frequency features like edges and corner points, which are lost during the defocus blur formation process, while the residual  $\Delta r_{\downarrow n}$  focus on the low frequency features that represent the essential content covered by defocus blur. The dynamic residual  $\Delta \hat{x}_{\downarrow n}$ , residual  $\Delta r_{\downarrow n}$  and the input image  $\hat{x}_{\downarrow n}$  jointly contribute to recovering the latent sharp image. Section 5.3 demonstrates the effectiveness of the proposed DRB.

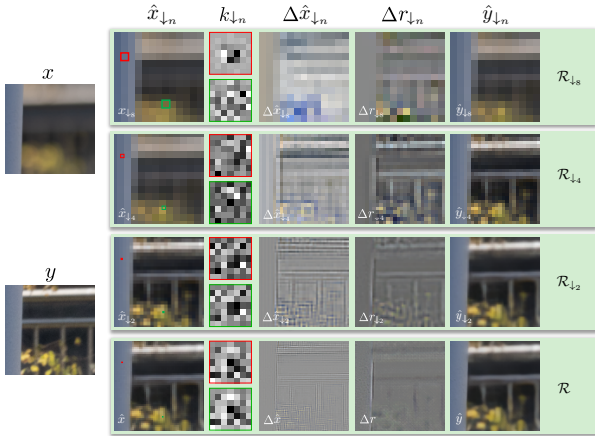


Figure 4. Visualization of each step in DRB on all scales. From left to right: original input  $x$  and ground truth  $y$ , input of DRB  $\hat{x}_{\downarrow n}$ , dynamic kernel on two pixels  $k_{\downarrow n}$ , dynamic residual  $\Delta \hat{x}_{\downarrow n}$ , residual  $\Delta r_{\downarrow n}$ , output of DRB  $\hat{y}_{\downarrow n}$ .

**Training Strategy and Loss** As demonstrated in Sec. 3, the defocus blur produced by a conventional digital camera and a light field camera are different. To remedy this gap, we propose a training strategy that leverages the strength

and overcomes the shortcoming of light field data. Specifically, we apply the light field generated dataset LFDOF in the main training round for its highly accurate image correspondence. Then, we fine-tune the network using DPDD dataset to alleviate the differences between the two domains. Different losses are used in each stage. In the main training stage,  $\ell_1$  norm is used:

$$\mathcal{L}_1^{\text{LF}} = \|\hat{y}_{\text{LF}} - y_{\text{LF}}\|_1 \quad (6)$$

During the fine-tuning stage, pixel-wise loss should be avoided because of the misalignment (induced from two-shot) between the defocused and all-in-focus pairs in DPDD dataset. We apply the VGG-based feature loss [11] in this step to transfer the learned knowledge to the target domain in the feature space, thus avoiding exact matching in the image space. We extract the feature maps from the 2nd, 7th, and 14th layer of the pre-trained VGG-19 network [35], denoted as  $\phi$ .

$$\mathcal{L}_{\text{VGG}}^{\text{DPDD}} = \|\phi(\hat{y}_{\text{DPDD}}) - \phi(y_{\text{DPDD}})\|_1 \quad (7)$$

Please note that the losses are applied to all scales. We will present in Sec. 5.3 that our training strategy contributes significantly to the final restoration performance.

## 5. Experiments

### 5.1. Datasets & Implementation

**Datasets** We perform experiments on five publicly available datasets for defocus deblurring evaluation, including CUHK [31], DPDD [2], LFDOF [29], PixelDP [2] and RealDOF [16], as shown in Tab. 1. Specifically, being different from DPDD which is collected with different apertures in consecutive shots, RealDOF is captured by a customized dual-camera system with two Sony  $\alpha 7R$  IV cameras, which are attached to a vertical rig with a beam splitter and equipped with a multi-camera trigger for simultaneous capturing. The images are then post-processed for geometric and photometric alignments. Note that CUHK and Pix-

Dataset	# Image	Resolution	Collect Method
CUHK [31]	704	$\sim 470 \times 610$	Internet
DPDD [2]	500	$1120 \times 1680$	Canon EOS 5D Mark IV
LFDOF [29]	12k	$688 \times 1008$	Lytro Illum
PixelDP [2]	13	$\sim 1680 \times 1120$	Google Pixel 4 Smartphone
RealDOF [16]	50	$\sim 1536 \times 2320$	Sony $\alpha$ 7R IV

Table 1. Datasets adopted for training and testing.

eIDP have no all-in-focus ground truth, as the former one is collected from the Internet while the latter is due to the fixed aperture of smartphone. We utilize LFDOF and DPDD for training and the remaining datasets for evaluation.

**Implementation** We implement and evaluate our models using PyTorch with Tesla V100-32GB. We use the Rectified-Adam optimizer [18] with  $\beta_1 = 0.9$  and  $\beta_2 = 0.99$ . The initial learning rate is set to  $10^{-4}$  when trained on LFDOF for 200 epochs and  $10^{-5}$  trained on DPDD for 100 epochs. The updating strategy of learning rate is similar to that of Zhu *et al.* [45], where the same learning rate is used for the first 100 epochs and then decayed linearly to zero over the rest 100. We set the batch size to 8 and patch size to  $320 \times 320$  augmented with Gaussian noise, gray-scale image conversion and scaling.

## 5.2. Comparison to the State-of-the-Art Methods

**Evaluation** We compare our proposed method with the four latest learning-based single image defocus deblurring works: DPDNet<sub>S</sub> [2], AIFNet [29], IFANet [16], and KPAC [36]. All these networks are trained on the DPDD dataset [2] except AIFNet which is trained on LFDOF. Specifically, IFANet needs to incorporate dual views in their network training to estimate the disparity map, while others do not. We use the codes and pre-trained weights released by the authors for comparison, then further evaluate their performance using RealDOF dataset [16].

In Tab. 2, we report the quantitative result using three standard evaluation metrics, including Peak Signal-to-Noise Ratio (PSNR), Structural Similarity (SSIM) [39] and Learned Perceptual Image Patch Similarity (LPIPS) [42]. We also list the parameter numbers for readers’ reference. Our proposed method performs the best among all by a significant margin especially on the RealDOF test set. For instance, our network outperforms DPDNet<sub>S</sub>, AIFNet, IFANet and KPAC by 12.6% (2.88dB), 11.5% (2.65dB), 4.2% (1.04dB) and 7.3% (1.76dB) in terms of PSNR respectively. Please note that RealDOF is not used for training, thus it is a fairer benchmark test set for comparing network performance.

Figure 5 shows the corresponding qualitative comparison. Although DPDNet<sub>S</sub> can reduce the defocus blur to some extent, it produces artifacts as shown in the second row in Fig. 5b). AIFNet gives sharp details in some cases, for instance, the leaves in third row in Fig. 5c and the characters in fourth row, but it fails to remove defocus blur in

other cases. This can be explained by their two-step network architecture, in which the final performance is partially determined by the intermediate defocus map. Thus, AIFNet fails to restore high quality details when the estimated defocus map is inaccurate. KPAC performs slightly better than DPDNet<sub>S</sub>, while in general it gives limited performance due to its small model capacity. Despite that IFANet shows competitive restoration performance, our proposed method performs better in restoring text (second and fourth row in Fig. 5f), texture (first row in Fig. 5f) and object boundary (third row in Fig. 5f). It is worth noting that IFANet needs dual views for network training while ours only needs single view. More results regarding the visual comparison, model complexity, computational costs, *etc.* are presented in the supplementary material.

**Generalization Ability** To inspect the generalization ability of our network, we further compare the visual performance of the networks using CUHK [31] and PixelDP [2] datasets. CUHK targets for blur detection with relatively small spatial resolution, and all images are collected from the Internet thus no all-in-focus ground truth are provided. PixelDP is collected with Google Pixel 4 smartphone which has a fixed aperture size and the image data is limited to one of the Green channels in the ray-Bayer frame. Figure 6 reports the visual comparison among the five networks and a similar conclusion can be drawn: our method gives the best generalization ability and can successfully restore the fine details regardless of the camera type. More results are provided in the supplementary material.

## 5.3. Analysis and Discussion

In this section, we conduct comprehensive ablation studies and analysis.

**Why LFDOF?** To understand the necessity of LFDOF for defocus deblurring, we train our network on LFDOF only, DPDD only and both. Here, we use ‘LFDOF & DPDD’ to represent the network trained on LFDOF then fine-tuned on DPDD for convenience. Table 3 and Fig. 7 show the quantitative and qualitative results evaluated using RealDOF test set. It is observed that our network trained on DPDD gives better scores in terms of all metrics than LFDOF. This can be explained by the domain difference between the light field-generated and real defocused images. However, the visual performance is not in line with the quantitative performance. The network trained on LFDOF produces sharper content and details than the one trained on DPDD does owing to the accurate pixel correspondence of LFDOF. However, at the same time, it also introduces artifacts (see wall in Fig. 7) due to the defocus blur discrepancy between light field generated and real data. Our strategy to train the network on LFDOF then fine-tune on DPDD largely outperforms the networks trained on either dataset alone and generates the best quantitative (increase up to 11.57% and

Method	DPDD Dataset			RealDOF Dataset			Params (M)
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	
Input	23.890	0.725	0.349	22.333	0.633	0.524	-
DPDNet <sub>S</sub> [2]	24.388	0.749	0.277	22.870	0.670	0.425	31.03
AIFNet [29]	24.213	0.742	0.309	23.093	0.680	0.413	41.55
IFANet [16]	25.366	0.789	0.217	24.709	0.749	0.306	10.48
KPAC [36]	25.221	0.774	0.226	23.984	0.716	0.336	2.06
Ours	<b>25.725</b>	<b>0.791</b>	<b>0.183</b>	<b>25.745</b>	<b>0.771</b>	<b>0.257</b>	11.69

Table 2. Quantitative comparison between our network against existing learning-based methods on single image defocus deblurring. Both datasets are evaluated using the codes and trained weights provided by the authors.

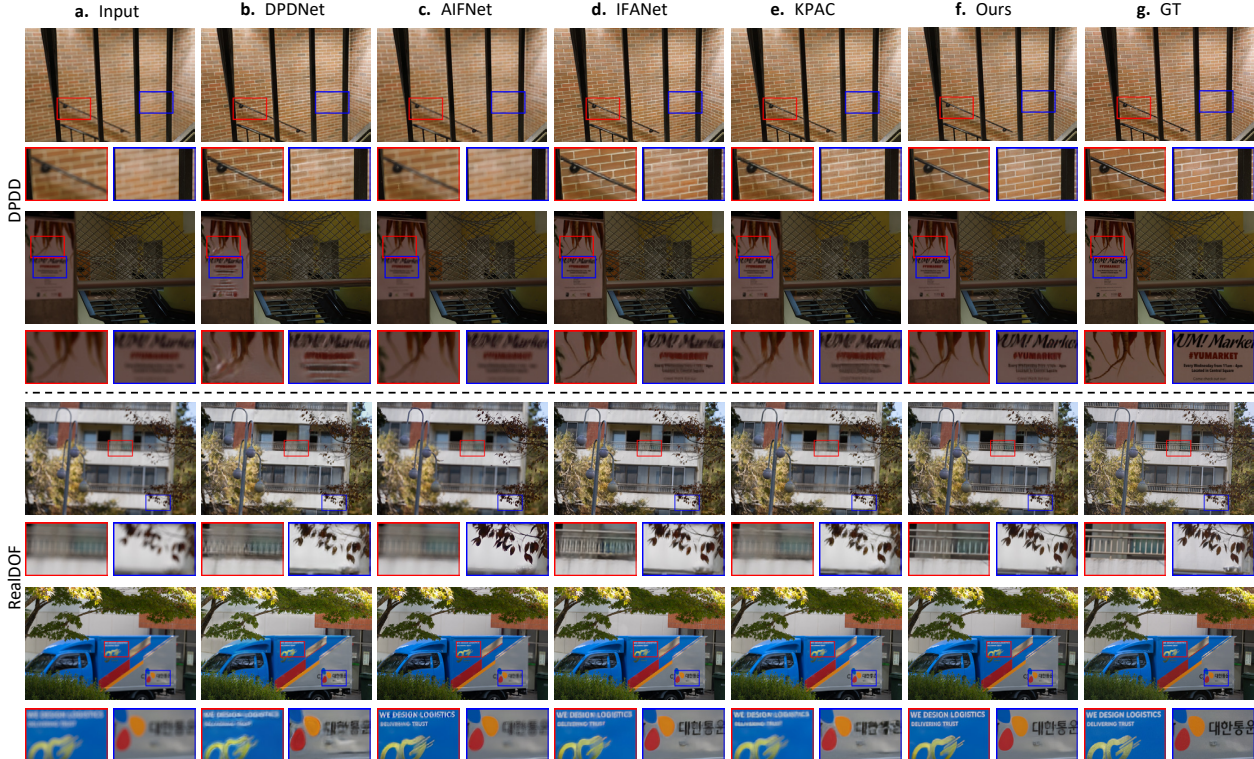


Figure 5. Qualitative evaluation on DPDD and RealDOF datasets among DPDNet<sub>S</sub> [2], AIFNet [29], IFANet [16], KPAC [36] and ours.

4.23% in terms of PSNR) and qualitative results.

Training Dataset	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
LFDOF [29]	23.076	0.698	0.378
DPDD [2]	24.700	0.744	0.337
LFDOF & DPDD (Ours)	<b>25.745</b>	<b>0.771</b>	<b>0.257</b>

Table 3. Quantitative comparison of the proposed network trained on LFDOF only, DPDD only and both datasets. The results are tested on RealDOF test set.

**Performance Gain** To further validate the proposed training strategy, we have retrained two state-of-the-art networks, DPDNet<sub>S</sub> [2] and KPAC [36], to see whether our training scheme could improve the performance of these two networks. AIFNet and IFANet are not listed here because the former requires defocus map as the ground truth

and the latter requires dual views for training. For KPAC, we choose the 3-level model with 2 KPAC blocks for comparison. Except the loss and learning rate which are set to be the same as ours in the two training stages, other parameter settings for DPDNet<sub>S</sub> and KPAC are the same as their original configurations. The network performance are evaluated on 76 test samples in DPDD dataset. Table 4 shows that DPDNet<sub>S</sub>, KPAC and our proposed architecture have gained 0.511dB, 0.249dB, and 0.253dB in terms of PSNR when networks are trained on LFDOF & DPDD. The visual quality has also been largely improved as shown in Fig. 8.

**Loss** Our proposed method trains the network on LFDOF with  $\mathcal{L}_1$  loss, then on DPDD with feature loss. One may wonder if it is possible to (1) directly mix the datasets for training instead of using a two-stage training strategy, and (2) adopt the same  $\mathcal{L}_1$  loss in the fine-tuning stage. To an-

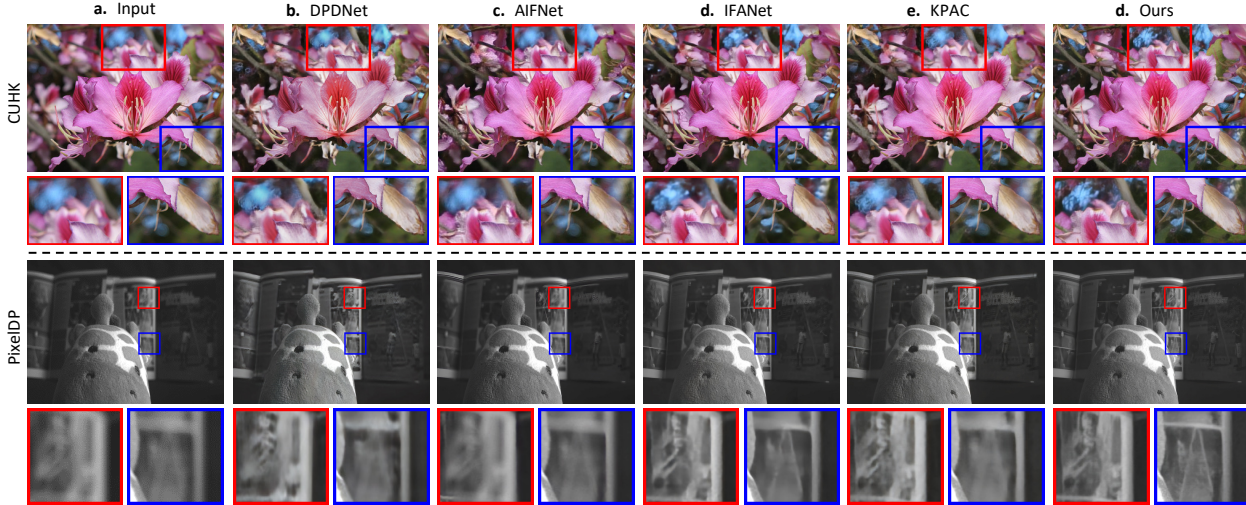


Figure 6. Qualitative comparison among  $\text{DPDNet}_S$  [2], AIFNet [29], IFAN [16], KPAC [36] and ours. Image in the first row is from CUHK [31] and that in the second row is from PixelDP [2]. No all-in-focus ground truth is provided in these two datasets.

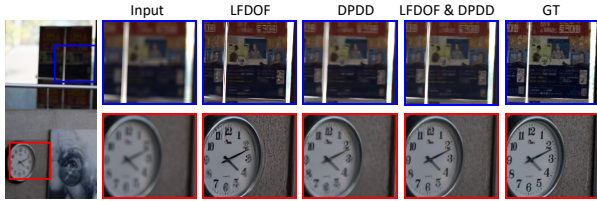


Figure 7. Visual comparison of our network trained on LFDof only, DPDD only and both.

Method	DPDD			LFDof & DPDD		
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
$\text{DPDNet}_S$ [2]	24.388	0.749	<b>0.277</b>	<b>24.899</b>	<b>0.761</b>	0.278
KPAC [36]	25.221	0.774	0.226	<b>25.470</b>	<b>0.780</b>	<b>0.220</b>
Ours	25.472	0.787	0.246	<b>25.725</b>	<b>0.791</b>	<b>0.183</b>

Table 4. Performance gain of  $\text{DPDNet}_S$  [2], KPAC [36] and ours when trained on LFDof & DPDD. The networks are tested on 76 test samples in DPDD dataset.

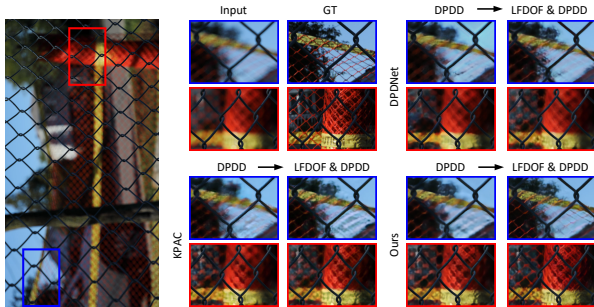


Figure 8. Qualitative comparison of network performance using DPDD only (left column) and LFDof and DPDD datasets (right column) for training. All networks get improved performance.

to answer these questions, we ablate the use of loss, then evaluate their performance on DPDD and RealDOF datasets as shown in Tab. 5 and Fig. 9. To balance the ratio of the two datasets, we augment DPDD dataset (350 images) for

32 times, then mix with LFDof (11261 images). The network performs similarly when using  $\mathcal{L}_1$  loss alone and  $\mathcal{L}_1$  combined with  $\mathcal{L}_{\text{VGG}}$  loss on the mixed dataset, while performing slightly worse than the version trained on DPDD only (see Tab. 3 and Tab. 4). When adopting the same  $\mathcal{L}_1$  loss in two stages, the quantitative result in terms of PSNR and SSIM is comparable with our final one, while the perceptual score LPIPS is relatively worse on DPDD dataset. It is because the per-pixel loss on DPDD may not lead to an optimal performance, even the misalignment is not obvious for human perception (*e.g.*, Fig. 1) but sensitive to the network. In addition, its performance on RealDOF is less favorable with 0.62dB lower than our final one in terms of PSNR. In Fig. 9, the results in red frames are produced by our final model, which yield the most realistic and fine details. Both quantitative and qualitative performance further validate and support the proposed training strategy.

Strategy	No.	Loss		DPDD			RealDOF		
		$S_1$	$S_2$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
Mix	<b>a</b>	$\mathcal{L}_1$	-	25.439	0.793	0.237	24.634	0.751	0.330
	<b>b</b>	$\mathcal{L}_1 + \lambda \mathcal{L}_{\text{VGG}}$	-	25.469	0.793	0.236	24.684	0.751	0.329
Fine-tune	<b>c</b>	$\mathcal{L}_1$	$\mathcal{L}_1$	<b>25.755</b>	<b>0.797</b>	0.232	25.130	0.768	0.310
	<b>d</b>	$\mathcal{L}_1$	$\mathcal{L}_{\text{VGG}}$	25.725	0.791	<b>0.183</b>	<b>25.745</b>	<b>0.771</b>	<b>0.257</b>

Table 5. Quantitative comparison of the training strategy with respect to dataset and loss.  $S_1$  and  $S_2$  represent the main training on LFDof and fine-tune on DPDD respectively. There is only one stage when training on the mixed dataset and  $\lambda$  is set to  $10^{-5}$ .

**DRB Configurations** To validate the effect of each component in DRB, we conduct an ablation study and report the results in Tab. 6 and Fig. 10. We compare our final model with four variant networks: removing one component each time and resulting in the block with  $(\hat{x}_{\downarrow n}, \Delta \hat{x}_{\downarrow n}, -)$ ,  $(-, \Delta \hat{x}_{\downarrow n}, \Delta r_{\downarrow n})$ ,  $(\hat{x}_{\downarrow n}, -, \Delta r_{\downarrow n})$ , and baseline that direct out-

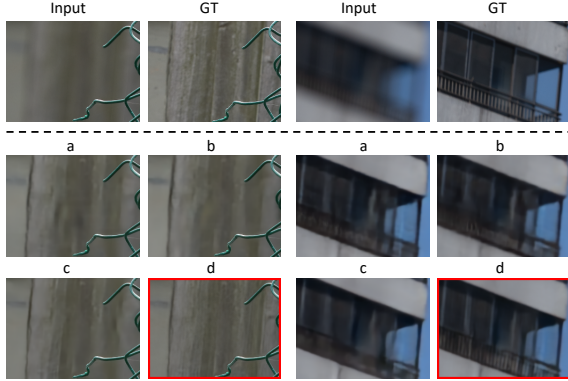


Figure 9. Visual comparison of the training strategy with respect to dataset and loss. Here a, b, c, and d indicate the corresponding training strategies in Tab. 5.

puts the restored result without DRB. Those variants are all with multi-scale architecture. Another variation with one full DRB (remove  $\mathcal{R}_{\downarrow\{8,4,2\}}$  thus leaving  $\mathcal{R}$  only) is also added for comparison. Both quantitative and qualitative result show the final model is able to restore the finest details as shown in Tab. 6 and Fig. 10. For visual quality, Figure 10 demonstrates that only the DRB with all the components is capable of restoring realistic details (see last two in the second row).

$\hat{x}_{\downarrow n}$	$\Delta\hat{x}_{\downarrow n}$	$\Delta r_{\downarrow n}$	multi-scale	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
	baseline			25.327	0.749	0.285
✓	✓		✓	25.539	0.763	0.271
	✓	✓	✓	25.576	0.763	0.267
✓		✓	✓	25.515	0.765	0.274
✓	✓	✓		25.532	0.757	0.272
✓	✓	✓	✓	<b>25.745</b>	<b>0.771</b>	<b>0.257</b>

Table 6. Ablation study on each component in DRB and the multi-scale restoration strategy. Performance is evaluated on RealDOF.

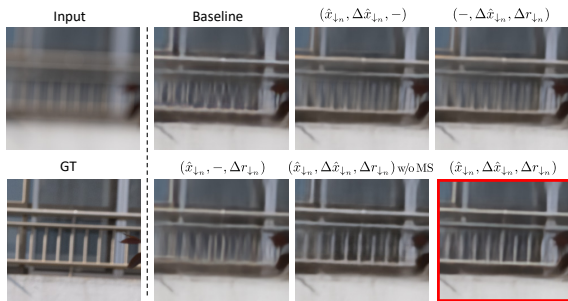


Figure 10. Qualitative results of the ablation study of DRB configurations evaluated on RealDOF.

**Dynamic Kernel Size** We analyze the effect of dynamic kernel size in our restoration performance. As illustrated in Tab. 7, we experimentally identify the best performing kernel size for DPDD dataset to be 9 while that for RealDOF to be 7. Considering the overall performance and parameter number, kernel size 7 is chosen.

Kernel Size	DPDD			RealDOF		
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
5	25.660	0.789	0.185	25.564	0.765	0.271
7	25.725	0.791	0.183	<b>25.745</b>	<b>0.771</b>	<b>0.257</b>
9	<b>25.752</b>	0.790	<b>0.182</b>	25.552	0.764	0.265
11	25.716	<b>0.793</b>	0.183	25.625	0.768	0.265

Table 7. Quantitative evaluation on DPDD and RealDOF datasets with respect to the kernel size in DRB block.

**AIFNet vs. Ours** Both AIFNet and ours are trained on LFDOF dataset. However, AIFNet employs defocus map estimation network followed by deblurring network while our network is an end-to-end architecture without explicit defocus map estimation. In order to compare these two networks, we train and test their performance only on LFDOF. Table 8 shows that our network architecture outperforms AIFNet by 0.726dB in terms of PSNR without the help of defocus map. Accurate defocus map may not be able to boost the restoration performance, whereas inaccurate ones will limit or impede the performance. This further validates the effectiveness of our proposed network architecture.

Method	Evaluation on LFDOF		
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
AIFNet	29.677	0.884	0.202
Ours	<b>30.403</b>	<b>0.891</b>	<b>0.145</b>

Table 8. Quantitative comparison between AIFNet and our network evaluated on 725 images from LFDOF test set. Both networks are trained on LFDOF training set.

## 6. Conclusion

We have proposed a novel method drawing on the synthetic aperture and refocusing features of light fields along with real captured defocus blur dataset to address the single image defocus deblurring problem. Our end-to-end neural network equipped with dynamic residual block is proven to be effective for removing spatially varying defocus blur. We train our network on light field generated dataset with MAE loss for the superior pixel-wise correspondence, then on real defocus dataset with feature loss to fully utilize the advantages of two types of data. We have proved this training strategy can be applied to improve the performance of several existing learning methods. Extensive comparison and ablation studies have demonstrated the effectiveness of our method, which outperforms others by a significant margin on multiple test sets.

**Limitations** Despite showing competitive performance, our proposed method shares some similar limitations with Lee *et al.* [16] and Son *et al.* [36] in handling blur with irregular shapes and defocus blur mixed with object motion. We include these failure cases in the supplementary material. Our future work will take these challenging cases into consideration.



## References

- [1] Abdullah Abuolaim, Mahmoud Afi, and Michael S Brown. Improving single-image defocus deblurring: How dual-pixel images help through multi-task learning. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 1231–1239, 2022. 2
- [2] Abdullah Abuolaim and Michael S Brown. Defocus deblurring using dual-pixel data. In *European Conference on Computer Vision*, pages 111–126. Springer, 2020. 1, 2, 4, 5, 6, 7
- [3] Abdullah Abuolaim, Mauricio Delbracio, Damien Kelly, Michael S Brown, and Peyman Milanfar. Learning to reduce defocus blur by realistically modeling dual-pixel data. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2289–2298, 2021. 2
- [4] Robert C Bolles, H Harlyn Baker, and David H Marimont. Epipolar-plane image analysis: An approach to determining structure from motion. *International Journal of Computer Vision*, 1(1):7–55, 1987. 3
- [5] Laurent D’Andrès, Jordi Salvador, Axel Kochale, and Sabine Süsstrunk. Non-parametric blur map regression for depth of field extension. *IEEE Transactions on Image Processing*, 25(4):1660–1673, 2016. 1
- [6] DA Fish, AM Brinicombe, ER Pike, and JG Walker. Blind deconvolution by means of the richardson–lucy algorithm. *Journal of the Optical Society of America A*, 12(1):58–65, 1995. 2
- [7] Thomas Hach, Johannes Steurer, Arvind Amruth, and Artur Pappenheim. Cinematic bokeh rendering for real scenes. In *Proceedings of the 12th European Conference on Visual Media Production*, pages 1–10, 2015. 1
- [8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 3
- [9] Xu Jia, Bert De Brabandere, Tinne Tuytelaars, and Luc V Gool. Dynamic filter networks. *Advances in Neural Information Processing Systems*, 29:667–675, 2016. 2, 3
- [10] Younghyun Jo, Seoung Wug Oh, Jaeyeon Kang, and Seon Joo Kim. Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern recognition*, pages 3224–3232, 2018. 2, 3
- [11] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer vision*, pages 694–711. Springer, 2016. 4
- [12] Ali Karaali and Claudio Rosito Jung. Edge-based defocus blur estimation with adaptive scale selection. *IEEE Transactions on Image Processing*, 27(3):1126–1137, 2017. 1, 2
- [13] Soo Ye Kim, Hyeonjun Sim, and Munchurl Kim. Koalanet: Blind super-resolution using kernel-oriented adaptive local adjustment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10611–10620, 2021. 2
- [14] Dilip Krishnan and Rob Fergus. Fast image deconvolution using hyper-laplacian priors. In *Advances in Neural Information Processing Systems*, pages 1033–1041, 2009. 2
- [15] Junyong Lee, Sungkil Lee, Sunghyun Cho, and Seungyong Lee. Deep defocus map estimation using domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12222–12230, 2019. 1, 2
- [16] Junyong Lee, Hyeongseok Son, Jaesung Rim, Sunghyun Cho, and Seungyong Lee. Iterative filter adaptive network for single image defocus deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2034–2042, 2021. 1, 2, 3, 4, 5, 6, 7, 8
- [17] Anat Levin, Rob Fergus, Frédo Durand, and William T Freeman. Image and depth from a conventional camera with a coded aperture. *ACM Transactions on Graphics*, 26(3):70–es, 2007. 1, 2
- [18] Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond. In *Proceedings of International Conference on Learning Representations*, 2020. 5
- [19] Pengyuan Lyu, Minghui Liao, Cong Yao, Wenhao Wu, and Xiang Bai. Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. In *Proceedings of the European Conference on Computer Vision*, pages 67–83, 2018. 1
- [20] Fahim Mannan and Michael S Langer. Blur calibration for depth from defocus. In *IEEE Conference on Computer and Robot Vision*, pages 281–288, 2016. 3
- [21] Ben Mildenhall, Jonathan T Barron, Jiawen Chen, Dillon Sharlet, Ren Ng, and Robert Carroll. Burst denoising with kernel prediction networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2502–2510, 2018. 2
- [22] Ren Ng, Marc Levoy, Mathieu Brédif, Gene Duval, Mark Horowitz, and Pat Hanrahan. Light field photography with a hand-held plenoptic camera. *Stanford University Computer Science Tech Report CSTR 2005-02*, 2005. 2
- [23] Liyuan Pan, Shah Chowdhury, Richard Hartley, Miaomiao Liu, Hongguang Zhang, and Hongdong Li. Dual pixel exploration: Simultaneous depth estimation and image restoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4340–4349, 2021. 2
- [24] Jinsun Park, Yu-Wing Tai, Donghyeon Cho, and In So Kweon. A unified approach of multi-scale deep and hand-crafted features for defocus estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1736–1745, 2017. 1, 2
- [25] Michael Potmesil and Indranil Chakravarty. A lens and aperture camera model for synthetic image generation. *ACM SIGGRAPH Computer Graphics*, 15(3):297–305, 1981. 1
- [26] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 1
- [27] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 3

- [28] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3234–3243, 2016. [2](#)
- [29] Lingyan Ruan, Bin Chen, Jizhou Li, and Miu-Ling Lam. Aifnet: All-in-focus image restoration network using a light field-based dataset. *IEEE Transactions on Computational Imaging*, 7:675–688, 2021. [1](#), [2](#), [4](#), [5](#), [6](#), [7](#)
- [30] Parikshit Sakurikar, Ishit Mehta, Vineeth N Balasubramanian, and PJ Narayanan. Refocusgan: Scene refocusing using a single image. In *Proceedings of the European Conference on Computer Vision*, pages 497–512, 2018. [1](#)
- [31] Jianping Shi, Li Xu, and Jiaya Jia. Discriminative blur detection features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2965–2972, 2014. [4](#), [5](#), [7](#)
- [32] Jianping Shi, Li Xu, and Jiaya Jia. Just noticeable defocus blur detection and estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 657–665, 2015. [1](#), [2](#)
- [33] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgb-d images. In *European Conference on Computer Vision*, pages 746–760. Springer, 2012. [2](#)
- [34] Hyeonjun Sim and Munchurl Kim. A deep motion deblurring network based on per-pixel adaptive kernels with residual down-up and up-down modules. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. [3](#)
- [35] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. [4](#)
- [36] Hyeongseok Son, Junyong Lee, Sunghyun Cho, and Seungyong Lee. Single image defocus deblurring using kernel-sharing parallel atrous convolutions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2642–2650, 2021. [1](#), [2](#), [5](#), [6](#), [7](#), [8](#)
- [37] Yu-Wing Tai and Michael S Brown. Single image defocus map estimation using local contrast prior. In *IEEE International Conference on Image Processing*, pages 1797–1800. IEEE, 2009. [2](#)
- [38] Tu Vo. Attention! stay focus! In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 479–486, 2021. [2](#)
- [39] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. [5](#)
- [40] Yu-Syuan Xu, Shou-Yao Roy Tseng, Yu Tseng, Hsien-Kai Kuo, and Yi-Min Tsai. Unified dynamic convolutional network for super-resolution with variational degradations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12496–12505, 2020. [2](#), [3](#)
- [41] Lu Yuan, Jian Sun, Long Quan, and Heung-Yeung Shum. Image deblurring with blurred/noisy image pairs. In *ACM SIGGRAPH*, page 1–es, 2007. [1](#)
- [42] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018. [5](#)
- [43] Wenda Zhao, Fan Zhao, Dong Wang, and Huchuan Lu. Defocus blur detection via multi-stream bottom-top-bottom network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(8):1884–1897, 2019. [2](#)
- [44] Shangchen Zhou, Jiawei Zhang, Jinshan Pan, Haozhe Xie, Wangmeng Zuo, and Jimmy Ren. Spatio-temporal filter adaptive network for video deblurring. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2482–2491, 2019. [2](#)
- [45] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2223–2232, 2017. [5](#)
- [46] Shaojie Zhuo and Terence Sim. Defocus map estimation from a single image. *Pattern Recognition*, 44(9):1852–1858, 2011. [2](#)