Lens-to-Lens Bokeh Effect Transformation. NTIRE 2023 Challenge Report

Marcos V. Conde* Manuel Kolmet† Tim Seizinger* Tom E. Bishop† Radu Timofte*
Xiangyu Kong Dafeng Zhang Jinlong Wu Fan Wang Juewen Peng Zhiyu Pan
Chengxin Liu Xianrui Luo Huiqiang Sun Liao Shen Zhiguo Cao Ke Xian
Chaowei Liu Zigeng Chen Xingyi Yang Songhua Liu Yongcheng Jing
Michael Bi Mi Xinchao Wang Zhihao Yang Wenyi Lian Siyuan Lai
Haichuan Zhang Trung Hoang Amirsaeed Yazdani Vishal Monga Ziwei Luo
Fredrik K. Gustafsson Zheng Zhao Jens Sjölund Thomas B. Schön Yuxuan Zhao
Baoliang Chen Yiqing Xu JiXiangNiu

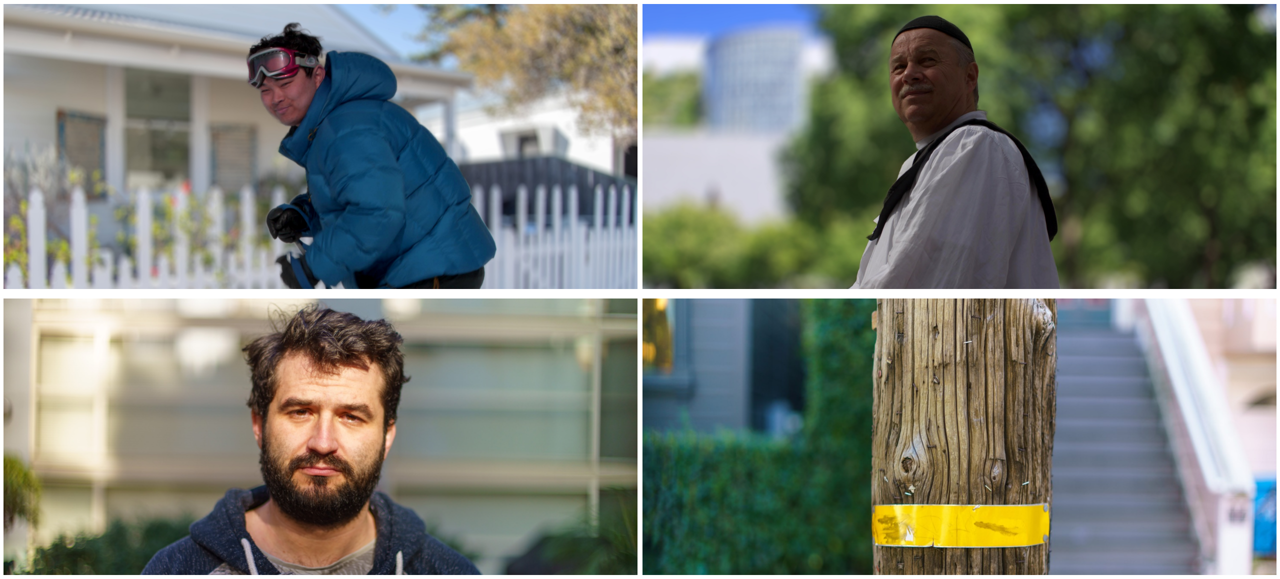

Figure 1. Samples from our **Bokeh Effect Transformation Dataset (BETD)**. (Top) Synthetic samples. (Bot.) Real captures. The synthetic images are generated by applying estimated space-varying PSFs. We use a *Sony Alpha 7R II and IV* cameras with different Sony and Canon 50mm lenses set to *f/1.4*, *f/1.8* and *f/16* apertures, to give a range of different bokeh effect conditions to map between.

Abstract

We present the new Bokeh Effect Transformation Dataset (BETD), and review the proposed solutions for this novel task at the NTIRE 2023 Bokeh Effect Transformation Challenge. Recent advancements of mobile photography aim to reach the visual quality of full-frame cameras. Now, a goal in computational photography is to optimize the Bokeh effect itself, which is the aesthetic quality of the blur in out-of-focus areas of an image. Photographers create this aesthetic effect by benefiting from the lens optical properties.

The aim of this work is to design a neural network capable of converting the the Bokeh effect of one lens to the effect of another lens without harming the sharp foreground regions in the image. For a given input image, knowing the target lens type, we render or transform the Bokeh effect accordingly to the lens properties. We build the BETD using two full-frame Sony cameras, and diverse lens setups.

To the best of our knowledge, we are the first attempt to solve this novel task, and we provide the first BETD dataset and benchmark for it. The challenge had 99 registered participants. The submitted methods gauge the state-of-the-art in Bokeh effect rendering and transformation.

* Organizers and corresponding authors. Computer Vision Lab, University of Würzburg, Germany. † Glass Imaging, Inc., CA.
{marcos.conde, radu.timofte}@uni-wuerzburg.de

NTIRE 2023 webpage: https://cvlai.net/ntire/2023/
Code: https://github.com/Glass-Imaging/NTIRE23BokehTransformation

1. Introduction

Computational photography research and recent advancements of mobile cameras aim to reach the visual quality of professional full-frame DSLR cameras [14, 23]. One of the most popular effects in photography is Bokeh, which is the aesthetic quality of the blur in out-of-focus areas of an image. This is shown in Fig. 1. In professional full-frame photography, this effect is controlled by the optical design of a lens, its aperture setting, the distance to the subject, and the focal length of the lens.

Therefore different Bokeh styles – for the same input image or scene – can be created using different lens designs and aperture settings. Some more detail comparing the subtle differences that appear in out-of-focus highlights of an image are shown in the zoom-in captured by two different lenses of the same scene in Fig. 2. Note the bokeh shapes differ in complex ways across the image.

However, due to the physical limitations of mobile cameras *e.g*. limited sensor size, these cannot produce a pleasant Bokeh effect naturally. In this case, the effect has to be created in post-processing, which is the main focus and application of most algorithms for Bokeh rendering.

Classical approaches [4, 39, 58, 64, 72] render Bokeh styles by controlling the shape and size of the blur kernel, which is usually an estimated point spread function (PSF). However, these methods might produce unpleasant artifacts such as chromatic aberration and depth discontinuities. Moreover they typically do not model the full natural space-varying and non-uniform nature of Bokeh from real lenses such as those effects shown in Fig. 2.

Deep learning-based methods [22, 43, 46, 60, 63] represent the *state-of-the-art* for this problem, but they have difficulty simulating different real Bokeh styles, and only produce the style present in the training data. Moreover, these methods lack a mechanism to produce large blur size on high-resolution images, as they are limited by the fixed receptive field of the neural network, and the blur size present in the training data. A common approach to render Bokeh consists in segmenting out the foreground (*e.g*. face, person, or main object of interest) in the photo, and then blurring the background [25, 52, 53, 74]. A similar approach is to blur the image based on a (estimated) depth map [21, 43]. We can also find end-to-end deep learning-based solutions [22, 25, 51] capable of transforming wide to shallow depth-of-field images automatically.

Despite the active research in this topic, rendering photorealistic Bokeh is still a challenging task. Moreover, we still find unexplored the **Bokeh Transformation** task.

We define this task as follows: for a given input (all-in-focus, out-of-focus or in-between) image A with known lens-type and aperture setting, knowing the target lens type and setting, we aim to produce or transform the corresponding effect B while preserving the foreground intact.

In this work, we aim to study different deep learning solutions capable of rendering or converting the Bokeh effect of one lens to the effect of another lens without harming the sharp foreground regions in the image. To the best of our knowledge, we present the first dataset and benchmark for this task, the Bokeh Effect Transformation Dataset (BETD) [12, 51].

2. Related Work

Classical Bokeh rendering methods require a single image and its corresponding depth map [3, 19, 39, 64, 66]. More advanced classical rendering also require the complete 3D scene information [43], however, these are not practical for the use-case studied in this work.

Multiple rendering approaches split the task into: depth estimation [21], semantic segmentation [9], and classical rendering [39, 44, 52, 53, 58, 72]. This task decomposition also implies decoupling the image into at least background and foreground, and execute rendering from back to front.

These modular or model-based approaches are flexible, and potentially adaptable for modern ISPs [13], however, they might **struggle at depth discontinuities** due to: occlusions modifying the blur effect (part of the lens aperture sees behind the foreground and part does not); semi-transparency of hair; and incorrect segmentations. Furthermore their performance depends highly on the marginal performance from each of the modules *e.g*. the quality of the estimated depth maps, or the quality of the background-foreground segmentation.

During the recent years we can observe a trend towards using deep learning to simulate the rendering process as an end-to-end operation. We find early works such as Nalbach *et al*. [42] and Xiao *et al*. [65] where the authors train neural networks to produce a bokeh effect from an all-in-focus image and its corresponding perfect depth map. Wang *et al*. [60] proposes an automatic rendering system comprised of depth estimation, lens blur, and guided upsampling to generate high-resolution depth-of-field (DoF) images from a single all-in-focus image. Peng *et al*. proposes BokehMe [43], a framework that combines neural and classical rendering achieving *state-of-the-art* results.

Other deep learning-based methods [22, 25, 32, 35, 46, 51, 63] do not require any prior information such as depth maps, which are not easy to capture in real-world scenes. These methods usually follow a encoder-decoder architecture [49], and map the all-in-focus input images into shallow DoF images in an end-to-end manner.

Despite the promising results, these neural rendering methods **lack controllability**, as the trained neural network can produce only the style of effect present in the training data, and the blur range is limited by their receptive field. In addition, we must note that all the referred methods are still far from simulating the look of a real Bokeh effect gener-

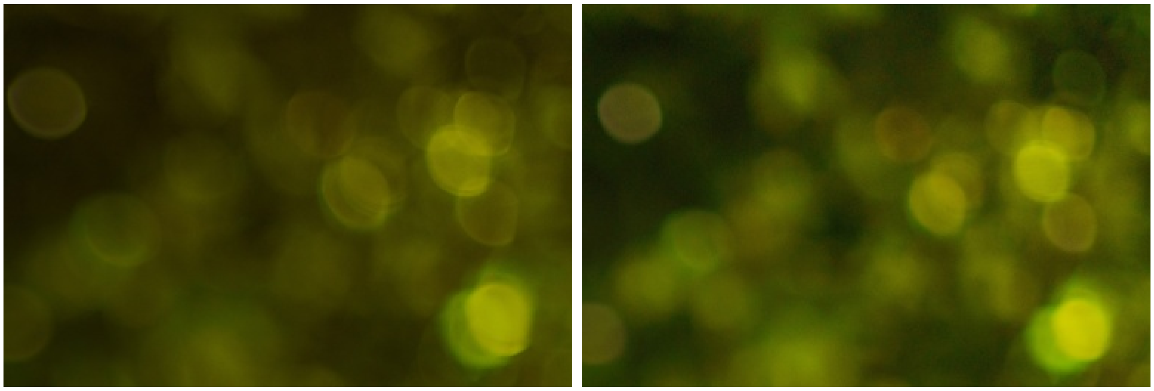

Figure 2. **Zoom in on Real captures of the same scene with different lens/aperture parameters**. Unlike many existing bokeh rendering methods, natural DSLR lenses exhibit various characteristices that are aesthetically desirable to model, such as space-varying, non-circular blur ("cat-eye" Bokeh effect towards corners), aperture blade shapes and clipping, aberrations or bokeh fringing, non-uniform bokeh intensity at large aperture settings, as well as diffraction which is also present at smaller aperture settings.

ated with professional full-frame cameras for several of the reasons already described that have not been well modeled.

Since the ultimate goal is improving mobile photography, it is also important to address the method complexity considering the computational limitations of mobile devices. Ignatov *et al*. proposed several challenges and studies [22, 24, 25] on efficient Bokeh rendering on mobile devices, being able to deploy the models on different target platforms [25]. These challenges use the popular large-scale *Everything is Better with Bokeh!* (EBB!) dataset [22] containing more than 10 thousand images collected in the wild. By controlling the aperture size of the lens, pairs of images with wide (aperture *f/16*) and shallow (aperture *f/1.8*) depth-of-field were taken, resulting in a normal sharp photo and one exhibiting a strong Bokeh effect.

In this report we describe our new BEDT dataset which goes a step further by considering different aperture settings across different lenses; we then present the new benchmark task which involves mapping from one lens/aperture-setting to another. We cover participants' solutions to the challenge which cover a wide variety of neural rendering approaches, from end-to-end networks to multi-stage processing and diffusion models.

Related NTIRE 2023 Challenges. The NTIRE 2023 Lens-to-Lens Bokeh Effect Transformation Challenge is part of the NTIRE 2023 Workshop series of challenges on: night photography rendering [54], HR depth from images of specular and transparent surfaces [68], image denoising [34], video colorization [27], shadow removal [56, 57], quality assessment of video enhancement [37], stereo super-resolution [59], light field image super-resolution [62], image super-resolution ($\times 4$) [73], 360° omnidirectional image and video super-resolution [7], lens-to-lens bokeh effect transformation [12, 51], real-time 4K super-resolution [15, 69], HR nonhomogenous dehazing [1], efficient super-resolution [33].

3. BETD Dataset and Benchmark

To pose the novel challenge of *Lens-to-Lens Bokeh Effect Transformation*, we have gathered training and testing datasets consisting of source-target image pairs.

Training set backgrounds are created by gathering natural images from the web [5] during which we prioritise natural scenes and sharp images. Each of these images is then artificially blurred with two different lens simulations to create a source-target pair. Note that the Bokeh transformation is bidirectional: e.g. wide *f/16* $\longleftrightarrow$ narrow *f/1.8*, or in other words, sharper photo $\longleftrightarrow$ strong Bokeh effect. This implies a new level of difficulty compared with previous work. Moreover, we use different two lenses, each with two different sets of apertures as we will explain next.

To create realistic artificial background blur, we capture the *space-varying* point spread functions (PSFs) of multiple commercial lenses attached to DSLR cameras, and then locally convolve them with our gathered images. This process models the nature of real aberrations, field variation, changing aperture shapes etc, as shown in Fig. 2.

To further increase the realism of the training set, we also add segmented portraits into the foreground that are spared from the artificial blur, resulting in images of sharp people in front of blurred backgrounds as shown in Fig. 1 (Up). The foregrounds were obtained by using the foregrounds and human segmentation masks from the iHarmony dataset [5, 16]. To obtain sharper edges, we dilated the segmentation masks and then ran an Alpha Matting model to optimize the segmentation. This also results in some semitransparent areas e.g. around hair, that can add to the complexity and realism of the challenge.

In total, our artificial data contains 20000 and 500 images for the training and validation sets, respectively.

Camera setup. Both the simulated and real images are based on *Sony Alpha 7R II* and *Sony Alpha 7R IV* professional cameras with a Sony 50mm lens set to *f/1.8* and *f/16* apertures and a Canon EF 50mm lens set to *f/1.8* and *f/1.4* apertures. For each real or synthetic pair, we provide the corresponding metadata for the source and target images *e.g.* `Sony50mmf1.8BS` → `Canon50mmf1.4BS`. We also provide a nominal disparity value that indicates the relative distance of the foreground to the background, such that the amount of blur can vary for each image pair, which adds additional variety to the dataset.

Evaluation. For our final testing set, we use another 100 artificially blurred images pairs, as well as 100 real image pairs that were captured with the same lenses in the real world, Fig. 1, Fig. 2, Fig. 3 and Fig. 5 show samples. The resolution of the RGB images is 1584 × 1056. We keep private the test ground-truth and the alpha masks (see Fig. 4). For the real captures, which are captured simultaneously with two cameras via a beamsplitter setup, we perform additional post-processing alignment and color normallization across the lenses which can vary in their spectral responses. This is desired to focus evaluation on the aesthetic character of the blur shape changes, and the behavior around foreground/background transitions.

To evaluate the performance of the proposed models, we use the established Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM) metrics to measure the closeness between model predictions and targets. We also consider perceptual metrics [11, 18, 71], especially for the comparison of real captures since these are not perfectly aligned. We use standard LPIPS [71] in this analysis.

Additionally, we measure the fidelity and appearance of the foreground and background on the images *w.r.t* the ground-truth. These metrics account for possible perturbations of the foreground (*e.g.* face, person, object of interest). The complete benchmark can be consulted in Tab. 1.

4. Methods and Resuls

4.1. Overview

Here we summarize the core ideas behind the most competitive solutions. Each proposed solution will be covered in the following sections.

1. **Decoder-Encoder** architectures following U-Net [49]. These sorts of networks are standard in image restoration [8, 14, 70]. The number of encoder and decoder blocks, as the structure of the blocks *e.g.* residual blocks, NAFBlocks [8] varies in each solution.

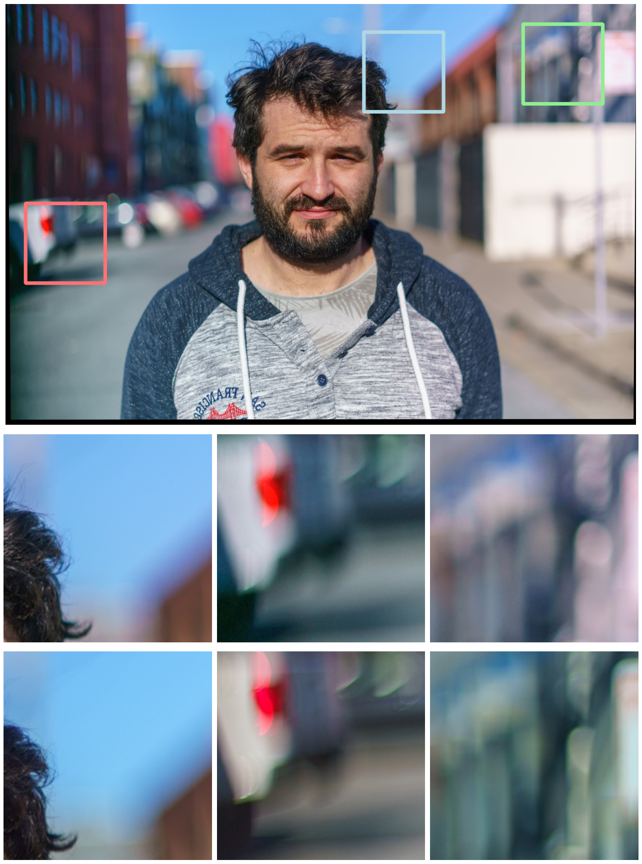

Figure 3. Real captures from BETD [12]. The first row of crops corresponds to the setting `Sony50mmf1.4`. The second row corresponds to the setting `Canon50mmf1.4`.

2. The **baseline** of most the promising methods is NAFNet [8], an efficient and *state-of-the-art* approach for image restoration. Some of the characteristics of this method are: simplied channel attention and the combination of LayerNorm [2] and GeLU [20].

3. **Multi-stage Training.** Since there are many different combinations of transformations and lens types, training becomes more complex. This technique allows to maximize learning by alternating different learning rates, augmentations and loss functions.

4. **Metadata.** The most powerful and flexible approaches encode the lens type and aperture *e.g.* `Sony50mmf16BS` as an additional feature in the network. By doing this, the methods can be conditioned towards different lens types and transformations. We consider this a powerful feature as controllable multi-lens Bokeh was unexplored.

Method	Synthetic + Real			Synthetic		Real		Foreground/Background	
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	SSIM ↑	LPIPS ↓	PSNR$_F$ ↑	SSIM$_B$ ↑
NAFBET [30]	35.264	**0.9362**	0.0985	45.861	0.9960	0.8416	0.2186	47.512	0.9553
SBTNet [45]	34.572	0.9361	**0.0966**	44.714	0.9945	0.8435	0.2224	47.889	0.9559
CBTNet	32.326	0.9333	0.1076	41.060	0.9910	0.8420	0.2230	46.875	0.9500
BokehOrNot [67]	32.288	0.9327	0.1130	41.003	0.9899	0.8423	0.2199	48.280	0.9488
SGLMS	32.076	0.9324	0.1076	40.651	0.9896	0.8419	0.2161	47.024	0.9484
IR-SDE	30.866	0.9297	0.1301	38.681	0.9847	0.8427	0.2387	44.905	0.9418
BGNet	30.327	0.9281	0.1249	37.804	0.9827	0.8415	0.2178	46.589	0.9410
DoubleGAN [21]	27.970	0.9213	0.1542	33.908	0.9691	0.8455	0.2175	41.522	0.9312
Synthetic	28.599	0.9128	0.2181	34.990	0.9580	-	-	48.163	0.9132
EBokehNet-s [51]	34.543	0.9350	0.1039	44.687	0.9942	0.8414	0.2206	47.220	0.9530
EBokehNet [51]	**35.521**	**0.9362**	0.0993	46.285	0.9962	0.8412	0.2208	47.577	0.9557

Table 1. **NTIRE 2023 Lens-to-Lens Bokeh Effect Transformation Challenge Results**. The methods are ranked by PSNR/SSIM. The metrics PSNR$_F$ and SSIM$_B$ refer to the foreground and background reconstruction respectively. The models were tested on unseen real captures and synthetic rendered content. The official challenge baseline method is EBokehNet [51]. We provide as reference "Synthetic" which indicates the metrics for the unprocessed source images. We can observe a performance gap between synthetic and real images, we will discuss the possible reasons in our conclusions.

Method	Input	Training Time (Hrs.)	Ensemble	Metadata	# Params. (M)	GPU
NAFBET	512x512	144	No	Yes	115	A100
SBTNet	288x288	120	No	Yes	265	GTX1080
CBTNet	1920x1440	120	Yes	Yes	182.24	TitanX
BokehOrNot	384x384	70	No	Yes	21.4	A100
SGLMS	512x512	36	No	Yes	7	TitanXP
IR-SDE	1920x1440	72	No	Yes	78	A100
BGNet	1152x1152	20	No	No	12.77	RTX3090
DoubleGAN	1024x1408	48	No	No	5	RTX3090
EBokehNet-s [51]	1024x1024	44	No	Yes	1.1	RTX3090
EBokehNet [51]	512x512	48	No	Yes	20.3	RTX3090

Table 2. For reproducibility purposes, we include a summary of implementation details for each method. We show the dimension of the input RGB image used for training the models, the approximated training time in hours, model complexity and platform.

4.2. Efficient Bokeh Rendering

We use as baseline EBokehNet [51], an efficient *state-of-the-art* solution for Bokeh Effect Rendering and Transformation. Our method can render (assuming an all-in-focus input) or transform the Bokeh effect of one lens to the effect of another lens while respecting the foreground regions in the image. Moreover we can control the effect by feeding the lens properties *i.e*. type (Sony or Canon) and aperture, into the neural network as an additional input. Therefore we can control how strong is the Bokeh effect, and simulate different lenses. The key features are:

1. Efficient encoder-decoder architecture based on NAFNet [8] with a modified baseline block.
2. Positional encoding (PE). Since Bokeh varies depending on the spatial location, we add to the decoder blocks explicitly the xy coordinates similar to CoordConv [36]. We do this by concatenating the 2-channel positional encoding with the corresponding input features for each decoder block.
3. We control the Bokeh effect by injecting the encoded lens properties into the deep features similar to [26].

This method also archives *state-of-the-art* results on the popular EBB! benchmark [22] for simple Bokeh Rendering. The small version EBokehNet-s, with only 1M parameters, represents the most efficient solution proposed in this challenge, while being ranked 2nd in terms of performance.

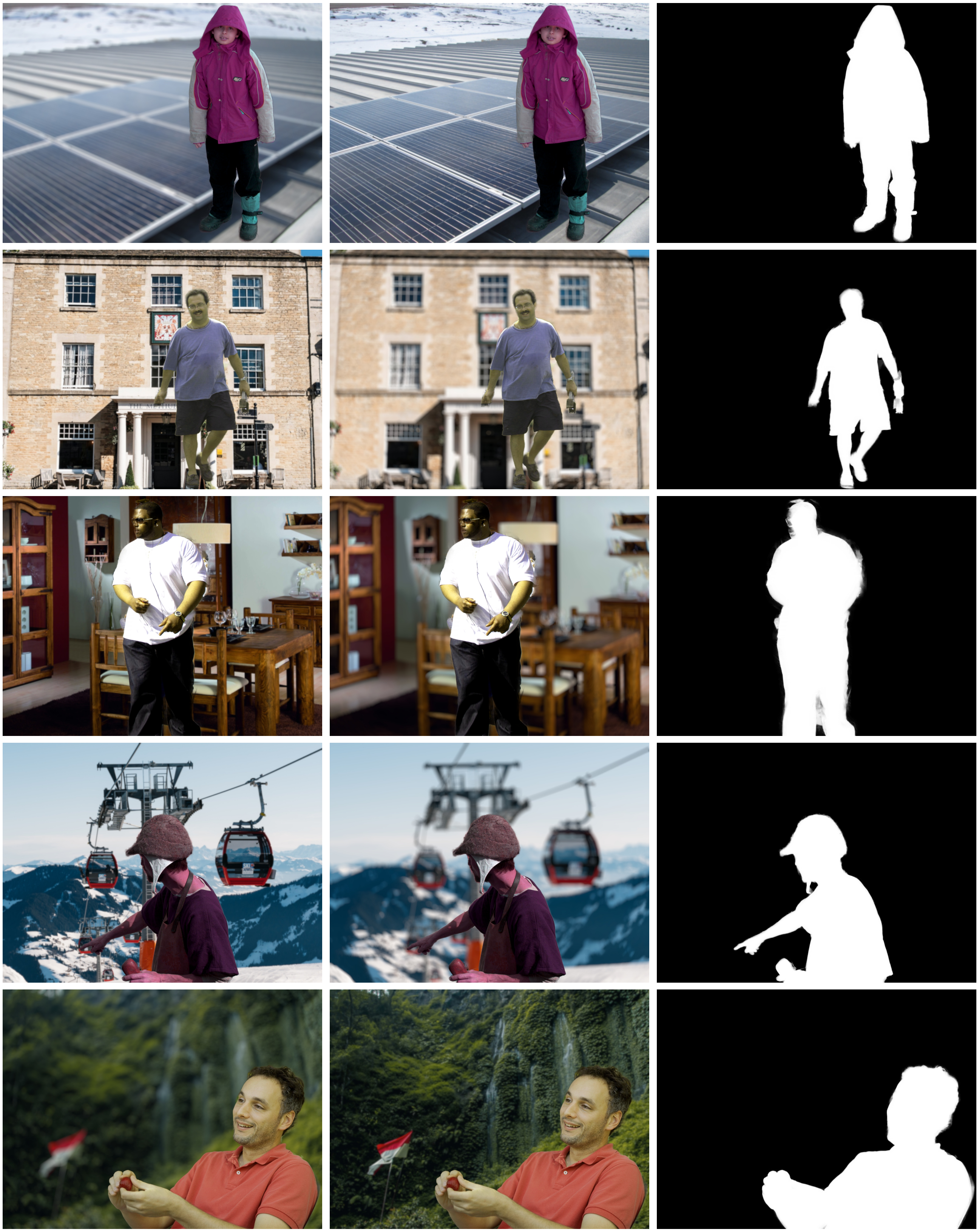

Figure 4. **Synthetic samples** from our BETD dataset. From left to right: source image, target image, alpha mask.

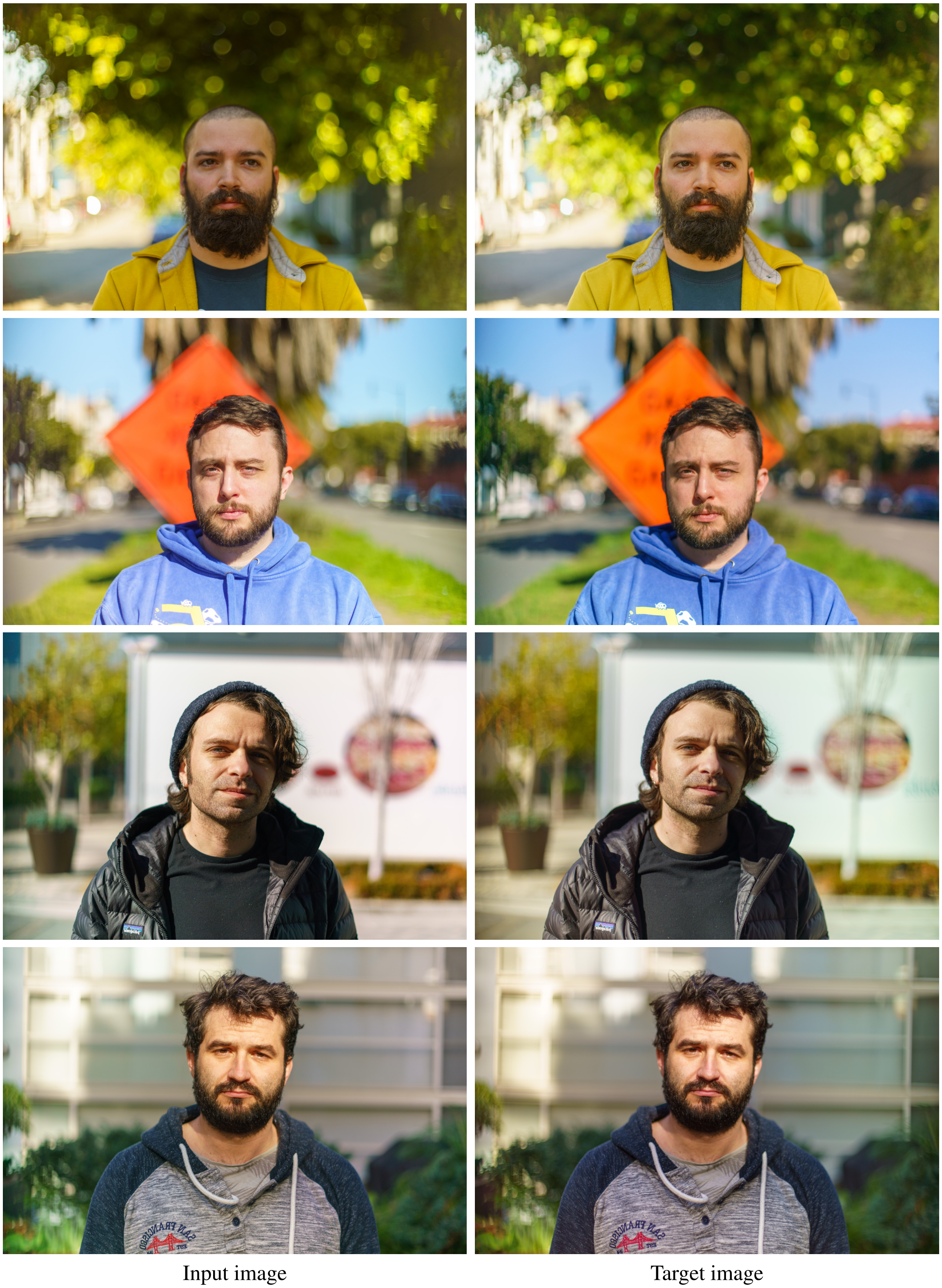

Figure 5. **Real captures** from our BETD. These images were captured using the setups Sony50mmf1.8BS and Canon50mmf1.4BS. The proposed models are able to do a bidirectional conversion between both setups Sony↔Canon. Since the images are not perfectly aligned, we use perceptual metrics such as SSIM and LPIPS [71] to evaluate the results. Images courtesy of Glass Imaging, Inc.

4.3. NAFBET

Team SRC-B proposes *NAFBET* [30]. Based on the image restoration model NAFNet [8], they add encoding and decoding parameters blocks to adapt the transition between different lens bokeh effects.

As shown in Figure 6, the authors design the model based on NAFNet [8], and add the lens parameters of the input (source) image and the lens parameters of the output (target) image at the front end of the encoder and decoder. The insertion method works as follows:

$$\theta_{src} = \beta(\alpha\, len_{src} + disparity) \quad (1)$$

$$\theta_{tgt} = \beta(\alpha\, len_{tgt} + disparity) \quad (2)$$

$$F_2 = F_1 + \theta_{src} F_1 \quad (3)$$

where F_1 and F_2 are deep extracted features.

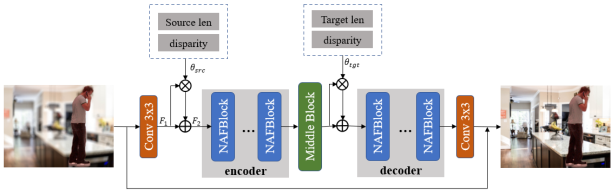

Figure 6. *Team SRC-B* proposed NAFBET [30] architecture.

Implementation Details The model was trained using only the new BETD [12, 51] dataset. In each training batch, each paired images (source and target) are cropped to 512 × 512 and augmented by random flipping and rotation. The learning rate is initialized as 2×10^{-5} and weight decay is 1×10^{-4}. The network is trained for 10^6 iterations in total by minimizing L1 loss function with AdamW optimizer. The team uses Pytorch and one A100 GPU.

4.4. SBTNet: Selective Bokeh Transformation

Team AIA-Smart proposes *SBTNet* [45] to tackle the task of bokeh transformation. As shown in Fig. 7, SBTNet contains several steps. First, they design AlphaNet to predict the alpha map of the object in focus, which facilitates preserving the sharp boundaries of focused objects in transformed results. To extract more global information, they implement AlphaNet with a U-Net [49] architecture where the bottom layers are replaced with short distance attention (SDA) and long distance attention (LDA) [61].

Since bokeh transformation includes the transformation of lens type and blur amount, the authors made the following designs for these two points. (i) they encode the lens type of the source image and the target image into a 2-channel one-hot map. Considering the cat-eye effect of camera lens (the bokeh balls are not circular at the corners of images), they additionally add a 2-channel coordinate map as input to reflect the degree of the cat-eye effect in different positions. (ii) for the blur amount transformation, we can argue that the ratio of blur amounts between source images and target images is most important other than the specific blur amount of images. Therefore, the team proposes several FeaNets -with the same architecture- to extract the multi-scale features with different blur amount transformation. Each FeaNet corresponds to a f-number pair. For example, FeaNet-1.4/16 means that the f-number of the source image is 1.4, while the f-number of the target image is 16.0. Thus, the blur ratio between the source image and the target image can be calculated by 1.4/16.

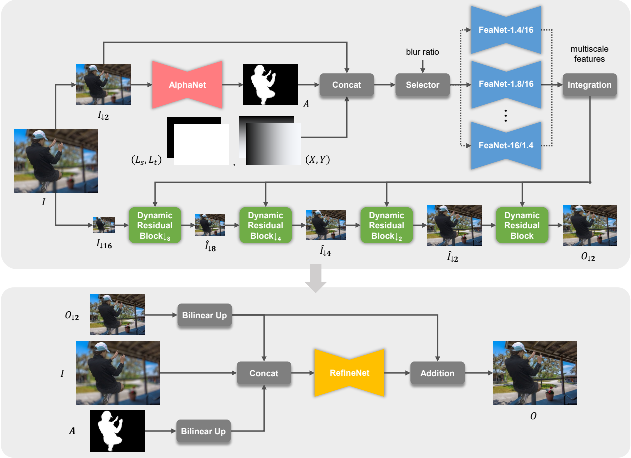

Figure 7. *Team AIA-Smart* proposed SBTNet framework.

In practice, during training, they select a particular FeaNet for each training sample, and during inference, they can interpolate the features of two neighboring FeaNets to obtain results with an intermediate blur ratio. This process is termed as integration. Then, they use 4 dynamic residual modules to obtain results progressively. Compared with interpolation in image level, interpolation in feature level performs better and has less parameters. The architecture here is similar to DRBNet [50]. Finally, the above predicted alpha map and output image are both in 1/2 resolution, so the team further designs RefineNet (as a simple U-Net [49]) to obtain full-resolution results.

Implementation Details The implementation is based on PyTorch. The training strategy contains 3 stages where Adam [29] optimizer is used for optimization. RefineNet is not used during the first 2 stages. At training stage 1, the team only uses the data where the f-number of source images is 1.8 and the f-number of target images is 16.0. Additionally, only FeaNet-1.8/16 is available. The learning rate is set to 10^{-4}. The model is trained for 300 epochs with a batch size of 8. The training time is around 30h. At training stage 2, all of the data are used, and with the input of different f-number pairs, corresponding FeaNet is active. They initialize the parameters of all FeaNets with the parameters of FeaNet-1.8/16. The learning rate is 10^{-4} for FeaNet and 10^{-5} for other structures. The model is trained for 100 epochs with a batch size of 32. The training time is around 60h. At training stage 3, the parameters except for RefineNet are fixed. The learning rate is set to 10^{-4}. The model is trained for 100 epochs with a batch size of 8. The training time is around 30h.

During inference, SBTNet can perform any arbitrary bokeh transformation by interpolating the features of FeaNet with the neighboring blur ratio.

Dataset The model was trained using only the new BETD [12, 51] dataset. At training stages 1 and 2, they first resize images to half resolution, and the inputs are randomly cropped into the size of 288×288. At training stage 3, there is no image resizing, and the cropping size is changed into 576×576. The solution is open-sourced at https://github.com/JuewenPeng/SBTNet.

4.5. CBTNet

Team NUS-LV Bokeh proposes *CBTNet*, a controllable bokeh transformation model based on U-net [49] structure, which consists of two components: foreground segmentation model and conditional background rendering model.

The overall architecture of the model is shown in Fig. 8. The model consists of two main components: (i) a foreground segmentation network based on U2NET [6]. (ii) is a conditional background rendering model [47] containing two branches, the debokeh branch and the bokeh branch. Depending on the metadata, the input image will go through the debokeh branch or the bokeh branch.

The overall model framework of the bokeh branch or debokeh branch can be seen in Fig. 9, which consists of two cascaded U-net and conditional MLP. The U-net [49] is mainly composed of Modulated Residual Blocks (MRBs) and Modulated ScaleFusion (MSF) — see Fig. 10 and 11.

The authors use the provided metadata *e.g*. type of lens, as conditional vectors for conditional background rendering model in both the training and inference stages.

The team uses mainly the new BETD dataset [12, 51], and a external dataset for the training of foreground segmentation model. During the training process, they first trained the foreground segmentation model using the provided alpha mask as the ground-truth. Next, they trained the first conditional U-net [49] network of the background rendering model, then froze it, and trained the second cascaded conditional U-net. In order to achieve better performance, they fine-tuned multiple models with different data and **ensemble** them for inference.

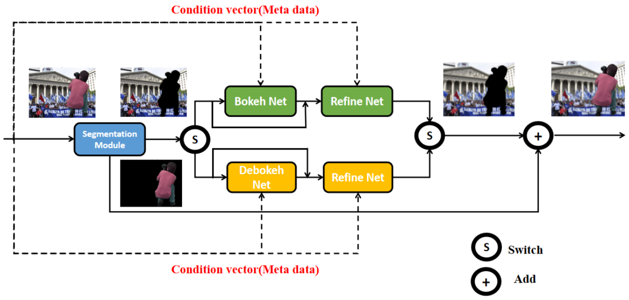

Figure 8. *Team NUS-LV Bokeh* CBTNet architecture.

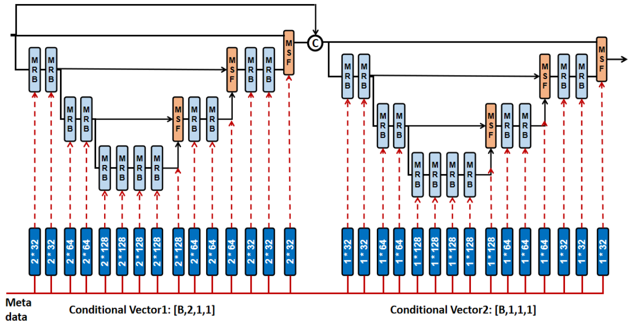
MRB
MSF
C
2 * 32
2 * 64
2 * 128
1 * 32
1 * 64
1 * 128
Meta data
Conditional Vector1: [B,2,1,1]
Conditional Vector2: [B,1,1,1]

Figure 9. CBTNet Conditional Rendering Model.

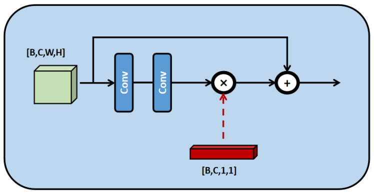

Figure 10. CBTNet: Modulated Residual Blocks (MRBs).

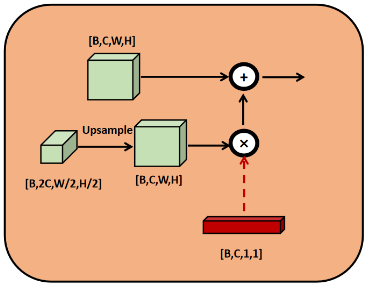

Figure 11. CBTNet: Modulated ScaleFusion (MSF).

Implementation Details The team used Pytorch framework and Adam [29] optimizer. For the foreground segmentation model, they utilized the L1 loss with a learning rate 1e-3. This model was trained for 24hrs using one TianX GPU with a batch size of 6. For the conditional background rendering model, they used a combination of L1, MSE and SSIM losses. The learning rate is initialized to 1e-4.

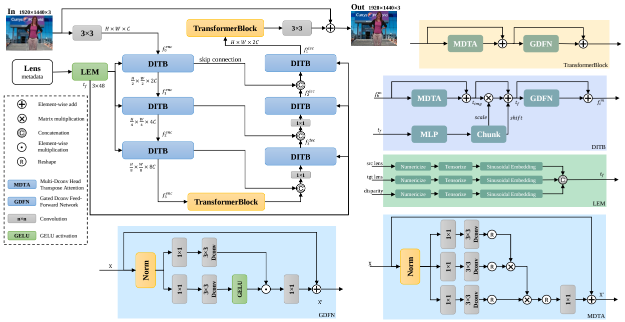

Figure 12. *Team BokehOrNot* architecture [67].

4.6. BokehOrNot

Team BokehOrNot proposes a method for Bokeh effect transformation based on lens and aperture size [67].

The conventional approach to single-photo bokeh transformation relies solely on the visual features of the image. In contrast, the new BETD dataset [12, 51] incorporates extra lens information to enhance the learning process. To this end, the team proposes an embedding technique that facilitates the transformation of lens type, aperture size, and disparity in the input images based on the image restoration model Restormer [70]. This method enables our model to learn distinct bokeh styles with increased accuracy and verisimilitude. Additionally, the authors seek to alleviate the undesired blurring of the foreground by leveraging alpha masks in the later stages of the training process. Specifically, they exclude the foreground area from the loss function to prevent the bokeh effect from being applied in this region. The overall framework is illustrated inFig. 12.

The model builds on Restormer [70], enhancing its functionality. The authors "tensorize" the source and target lens information as well as the disparity in the forward stage of the model, which are subsequently fed through a novel sinusoidal embedding module to generate a tensor suitable for model learning. These lens information tensors are concatenated and being processed by a MLP layer, and then concatenate with the image input tensors. The resulting longer tensor is passed through an optimized transformation block. The new transformation block involves scale and shift processing with residual added.

Technical Details The team uses only the BETD [12, 51] dataset. All the images are cropped to 256 × 256 or 384 × 384 patches randomly before feeding to the network.

Training and Inference: The training process contains two stages. The first stage uses 256 × 256 input size and the loss is calculated between the output image from the model and the target image. The second enlarges the input size to 384 × 384, and incorporates new loss function that calculates only the bokeh area loss, not including the foreground area which should remain as original as possible.

The optimizer is Adam [29] with a fixed learning rate of 1e-4 and a batch size of 4 in the first training phase, and a fixed learning rate of 5e-5 and a batch size of 2 in the second training phase. By conducting this, the distribution of transformation pairs obtains more blur-to-sharp transformations, which enhances the model robustness on sharpening transformation.

4.7. SGLMS

Team IPAL Bokeh proposes an end-to-end network for Bokeh Effect Transformation with Lens Mapping guided by foreground segmentation, which is named Segmentation-Guided Lens-Mapping Scheme (SGLMS). The main structure of the proposed network is shown in Fig. 13. It includes two main components, the Foreground Segmentation Module (FSM), and the Lens Mapping Module (LMM). The FSM will generate a foreground mask estimation based on the input image. The LMM will transform the blur part in the Bokeh image from one lens to another lens. The output image is generated by fusing the transformed image and the source image, guided by the foreground mask to keep the sharp foreground regions in the image.

Inspired by [28], the FSM has two branches: the detail prediction Branch, which will predict the detail of the foreground boundary, and the Semantic prediction Branch, which will give a general prediction of the foreground. The fusion branch will fuse the result of the two branches to generate the foreground alpha image. The structure of the FSM is shown in figure 14, and detail will be described in the Global Method Description Section.

For each type of lens, the blur kernel $\mathbf{k}$ is different from other types of lens. Therefore, the LMM is designed with multi-encoders and multi-decoders to deal with the Bokeh effect transformation between multi-lens. For the Bokeh effect transformation task between four types of lenses, the LMM will consist of four encoders and four decoders, as shown in Fig. 15. The encoder will be chosen based on the source lens information, and the decoder will be chosen based on the target lens information.

Fig. 16 shows one encoder-decoder connection from the LMM. The base architecture is designed as a U-Net [38],

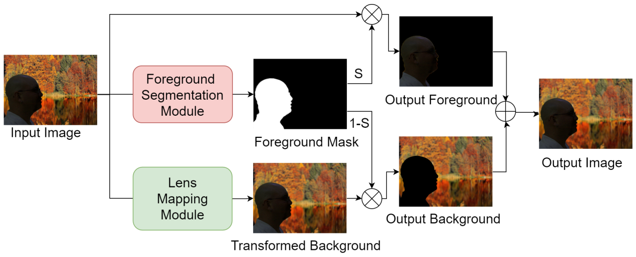

Figure 13. *Team IPAL Bokeh* proposed SGLMS architecture.

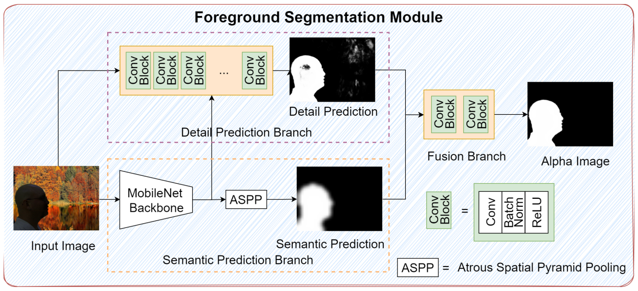

Figure 14. SGLMS Foreground Segmentation Module.

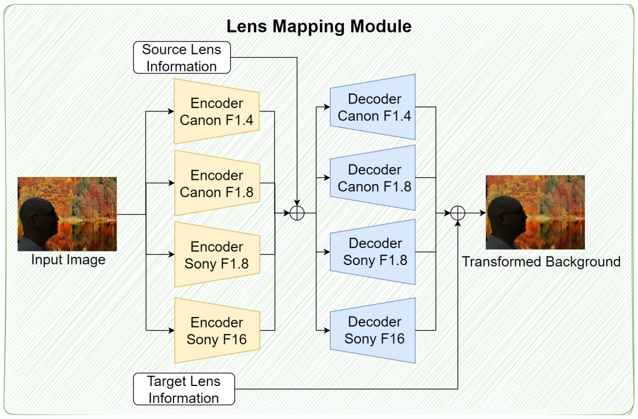

Figure 15. SGLMS Lens Mapping Module.

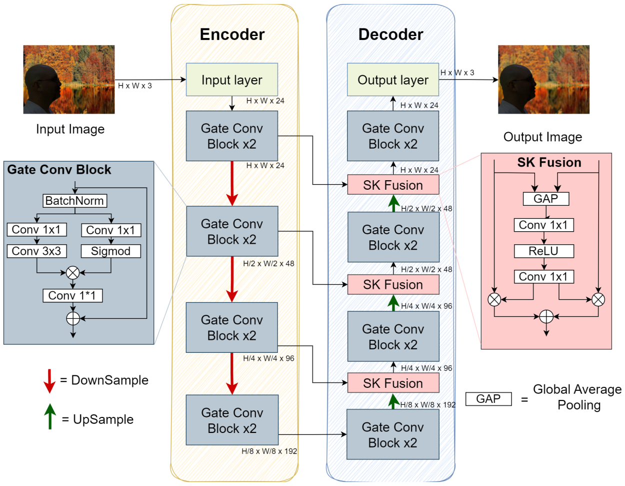

Figure 16. SGLMS: Encoder-Decoder Architecture in LMM.

which produces feature maps of different resolutions and so extracts multi-scale features. There are four layers for both the encoder and the decoder. As opposed to traditional U-nets, the proposed network uses gated convolution and SK fusion blocks [55] instead of traditional convolution blocks.

Technical details The team uses only the BETD [12, 51] dataset, and Pytorch framework. The models are trained using AdamW with learning rete 0.001 and a cosine annealing schedule. The dataset is split using a holdout 80/20 split for training and local validation. The team trained Foreground Segmentation Module (FSM) and Bokeh Effect Transformation Module (BETM) separately. For FSM, they resize the full image into (480, 640) and feed it to the FSM. The FSM is trained for 40 epochs with each kind of lens. For BETM, they randomly cropped images into (512, 512) and randomly flipped them. The BETM is trained with the reduced-size images, but is validated with the full-size images. Finally, the corresponding encoder-decoder of each kind of Bokeh Transformation will be extracted from the whole network and fine-tuned with the corresponding FSM.

4.8. IR-SDE

Team IR-SDE proposes Refusion: Enabling Large-Size Realistic Image Restoration with Latent-Space Diffusion Models [41], based on the IR-SDE [40].

The proposed method leverages the diffusion models for realistic image restoration. Specifically, IR-SDE [40] as the base diffusion framework, which can naturally transform the high-quality image to its degraded counterpart, without caring how complicated the degradation is. As shown in Figure 17, IR-SDE is a mean-reverting SDE in which the forward process is defined as:

$$dx = \theta_t \, (\mu - x) dt + \sigma_t dw, \tag{4}$$

where θ_t and σ_t are time-dependent positive parameters that characterize the speed of the mean-reversion and the stochastic volatility, respectively. Since it is an Ito SDE, the authors derive a reverse-time SDE:

$$dx = \Big[\theta_t \, (\mu - x) - \sigma_t^2 \, \nabla_x \log p_t(x)\Big] dt + \sigma_t d\hat{w}. \tag{5}$$

At test time, the only unknown part is the score $\nabla_x \log p_t(x)$ of the marginal distribution at time t. As other diffusion-based models, they employ a CNN network to estimate the score to backward from the low-quality image to the high-quality image.

It is also worth noting that running the above diffusion model needs to repeatedly evaluate the scores and thus is time-consuming, especially on tasks with high-resolution images. For Bokeh Effect Transformation, all inputs are captured with 1920 × 1440 × 3 pixels, which is a computation disaster for diffusion models. To handle it, the authors propose to perform the restoration on the low-resolution latent space, by incorporating a pretrained U-Net network. Different from latent-diffusion [48] that uses VAE as the compressing model, the proposed U-Net maintains multi-scale connections from the encoder to the decoder, which better captures the image's information and

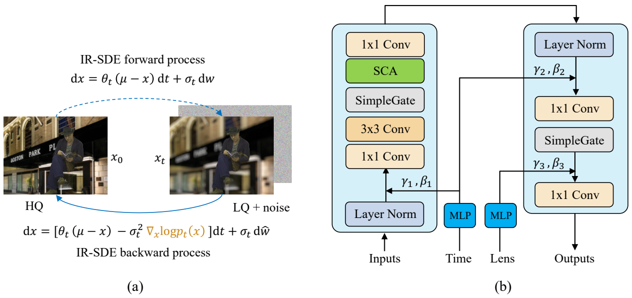

Figure 17. *Team IR-SDE* (a) Image restoration based on the IR-SDE [40]. (b) The modified NAFBlock [8].

thus the reconstructed image is closer to the original image. In this way, they are able to recover accurate HR images, and the training for U-Net [49] is also much easier than VAEs that use additional adversarial training.

Model Design. Unlike other L_1 loss normally trained networks which usually produce smooth/blurry results, the proposed Refusion aims to achieve a highly competitive perceptual performance as well as the distortion scores (PSNR). Since the guiding information is in text formation (such as "Sony50mmf1.4" or "Canon50mmf1.8"), the team manually tensorize them by converting the lens name to opposite numbers (-1 and 1). For example, "Canon50mmf1.4BS" is converted to -1.4, and "Sony50mmf16BS" is converted to 16.

In addition, the model can be further improved by updating the score-network from U-Net [49] to NAFNet [8], which is more efficient and also has a good performance compared with recent Transformers [70]. To adaptively insert the scalar times and lens information into the network, the authors construct a simple MLP to learn two pairs of scale-shift parameters and apply them to the features with affine transforms. Such a network leads to better learning of score function conditioned on current state x_t, original low-quality image x_t, and time t.

Training and Inference description. The team first train the U-Net on the BETD dataset [12, 51] for 300,000 iterations, then train the Reffusion model based on the U-Net for 400,000 iterations. In U-Net training phase, all input images are cropped to 128. In Reffusion model training phase, all input images cropped to 512 × 512, while using U-Net to compress them to 128 × 128. For both models, they use the Lion [10] optimizer with learning rate initialized with 3e-5 and decayed to 1e-7 by the Cosine scheduler. The diffusion step is set to 100. For testing, first compress the image to latent space and perform SDE to recover the clean image, and then decompress the image back to HR space.

4.9. BGNet

Team BIGbaodan proposes BGNet, based on the model DeblurGANv2 [31] and added some data processing methods. The team found DeblurGANv2 [31] to work the best among many other methods for image restoration and deblurring such as NAFNet [8].

According to the provided metadata, the authors divided the data into two categories: from sharp to blur, and from blur to sharp. They divide the image into two parts (foreground and background) using a semantic segmentation model. In this part, the Deeplabv3+ [9] model from `mmsegmentation` is used. The model is trained using as ground-truth the provided alpha masks.

Next, they trained DeblurGANv2 [31] twice. First, using the picture pairs that changed from blur to sharp as input and output. Second, using the picture pairs that changed from sharp to blur as input and output. For training this model, the images are segmented using the ground-truth alpha masks to cut out the foreground. During inference, the segmentation is done using the previously trained Deeplabv3+ [9] model. Once the background has been processed, the foreground is added back to the final image.

Technical Details The authors use only the new BETD [12, 51]. The main model, DeblurGANv2 [31], was trained using Adam optimizer [29] with a learning rate of 0.001 during 200 epochs, using random crops.

4.10. DoubleGAN

Team JiXiangNiu proposes a GAN [17] framework. This framework is inspired in BGGAN [46] and Team ZJUT-Vision solution [25]. They use two consecutive U-net [49] as generators (G). As for discriminator (D), they also use two discriminators in parallel. Specifically, both generator U-net [49] contain 9 residual blocks and transpose convolutions for upsampling. Also they use spatial attention block and channel attention block to enhance performance.

The target loss functions contains two parts, *i.e.* G loss and D loss. In particular, the G loss contains five parts: adversarial loss [17], perceptual loss, L1 loss, SSIM loss and FFT loss. Specifically, the FFT loss is a frequency loss, the team noticed that two images with different Bokeh effect have different frequency components, therefore, the define the FFT loss as a L2 loss in frequency domain, to better guide the model to transfer bokeh effects.

Technical Details The model is implemented with Tensorflow, and trained using Adam [29] optimizer with the learning rate set to 1e-5. The team only uses the BETD [12, 51], and a classical GAN training strategy. The complete training takes two GPU RTX 3090 days.

5. Conclusion

We introduced the novel Bokeh Effect Transformation Dataset (BETD) and benchmark. Previous work focused mainly on rendering realistic Bokeh effects, yet in this work we study neural networks capable of transforming the Bokeh effect of one lens to the effect of another lens without harming the sharp foreground regions in the image. For a given input image, knowing the target lens type and its settings, we can render or transform the Bokeh effect accordingly to the lens style. To study this novel task, we built the BETD dataset using two full-frame cameras, and diverse lens setups, and both simulated and real data.

While we have attempted to model many aspects of this real-world problem, there are still some limitations in using synthetic data only for training; real data can exhibit larger differences in spatial and color alignment, nonlinear sensor effects, more complex depth and occlusion based effects, noise and so on. We believe future studies can build on this work to close the sim-to-real gap that can exist, and for example use a larger set of real lens-to-lens images also for training. By its nature, Bokeh is a aestheticly subjective image feature, and it will also be informative to conduct human preference studies around such results as well as evaluate numerically. However to the best of our knowledge, we are the first attempt to solve this novel task, and we provide the first dataset and benchmark, gauging the state-of-the-art in Bokeh effect rendering and transformation.

Acknowledgements This work was partially supported by the Humboldt Foundation. We thank the NTIRE 2023 sponsors: Sony Interactive Entertainment, Meta Reality Labs, ModelScope, ETH Zürich (Computer Vision Lab) and University of Würzburg (Computer Vision Lab). Glass Imaging thanks Griffin Kettler and his models for assistance with the real images captured for the test portion of the dataset.

6. Appendix

6.1. NTIRE 2023 Team

Title: NTIRE 2023 Lens-to-Lens Bokeh Effect Transformation Challenge Organization
Members: Marcos V. Conde, Tim Seizinger, Radu Timofte
Affiliations: Computer Vision Lab, CAIDAS, IFI, University of Würzburg, Germany

6.2. Glass Imaging, Inc.

Title: NTIRE 2023 Workshop Co-organizers
Members: Manuel Kolmet, Tom E. Bishop
Affiliations: Glass Imaging, Inc. https://glass-imaging.com/

6.3. SRC-B

Title: NAFBET: Bokeh Effect Transformation with Parameter Analysis Block based on NAFNet
Members: Xiangyu Kong, Dafeng Zhang, Jinlong Wu, Fan Wang
Affiliations: Samsung Research China - Beijing (SRC-B)

6.4. AIA-Smart

Title: SBTNet: Selective Bokeh Transformation
Members: Juewen Peng, Zhiyu Pan, Chengxin Liu, Xianrui Luo, Huiqiang Sun, Liao Shen, Zhiguo Cao, Ke Xian
Affiliations: [1] Huazhong University of Science and Technology [2] Nanyang Technological University
https://github.com/JuewenPeng/SBTNet

6.5. NUS-LV-Bokeh

Title: CBTNet: Modulated Bokeh Transformation
Members: Chaowei Liu[1], Zigeng Chen[1], Xingyi Yang[1], Songhua Liu[1], Yongcheng Jing[3], Michael Bi Mi[2], Xinchao Wang[1]
Affiliations: [1]National University of Singapore [2]Huawei [3]University of Sydney
https://github.com/lcwLcw123/BKchallenge

6.6. BokehOrNot

Title: BokehOrNot: Bokeh effect transformation based on lens and aperture size
Members: Zhihao Yang, Wenyi Lian, Siyuan Lai
Affiliations: Uppsala University
https://github.com/indicator0/bokehornot

6.7. IPAL-Bokeh

Title: A Segmentation-Guided Lens-Mapping Scheme for Bokeh Effect Transformation
Members: Haichuan Zhang, Trung Hoang, Amirsaeed Yazdani, Vishal Monga
Affiliations: Department of Electrical Engineering, Pennsylvania State University, USA
http://signal.ee.psu.edu

6.8. IR-SDE

Title: Refusion: Enabling Large-Size Realistic Image Restoration with Latent-Space Diffusion Models
Members: Ziwei Luo, Fredrik K. Gustafsson, Zheng Zhao, Jens Sjölund, Thomas B. Schön
Affiliations: Department of Information Technology, Uppsala University

6.9. BIGbaodan

Title: BGNet: Improving DeblurGAN-v2 for Bokeh
Members: Yuxuan Zhao, Baoliang Chen, Yiqing Xu
Affiliations: Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education, Xidian University, Xi'an, China

6.10. DoubleGAN

Title: DoubleGAN
Members: JiXiangNiu
Affiliations: North China University of Technology

References

[1] Codruta O Ancuti, Cosmin Ancuti, Florin-Alexandru Vasluianu, Radu Timofte, et al. NTIRE 2023 challenge on nonhomogeneous dehazing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 3

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 4

[3] Jonathan T Barron, Andrew Adams, YiChang Shih, and Carlos Hernández. Fast bilateral-space stereo for synthetic defocus. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4466–4474, 2015. 2

[4] Marcelo Bertalmio, Pere Fort, and Daniel Sanchez-Crespo. Real-time, accurate depth of field using anisotropic diffusion and programmable graphics cards. In *Proceedings. 2nd International Symposium on 3D Data Processing, Visualization and Transmission, 2004. 3DPVT 2004.*, pages 767–773. IEEE, 2004. 2

[5] Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. Learning photographic global tonal adjustment with a database of input / output image pairs. In *The Twenty-Fourth IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 3

[6] Haoming Cai, Jingwen He, Yu Qiao, and Chao Dong. Toward interactive modulation for photo-realistic image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 294–303, 2021. 9

[7] Mingdeng Cao, Chong Mou, Fanghua Yu, Xintao Wang, Yinqiang Zheng, Jian Zhang, Chao Dong, Ying Shan, Gen Li, Radu Timofte, et al. NTIRE 2023 challenge on 360° omnidirectional image and video super-resolution: Datasets, methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 3

[8] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. *arXiv preprint arXiv:2204.04676*, 2022. 4, 5, 8, 12

[9] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 2, 12

[10] Xiangning Chen, Chen Liang, Da Huang, Esteban Real, Kaiyuan Wang, Yao Liu, Hieu Pham, Xuanyi Dong, Thang Luong, Cho-Jui Hsieh, et al. Symbolic discovery of optimization algorithms. *arXiv preprint arXiv:2302.06675*, 2023. 12

[11] Marcos V Conde, Maxime Burchi, and Radu Timofte. Conformer and blind noisy students for improved image quality assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 940–950, 2022. 4

[12] Marcos V Conde, Manuel Kolmet, Tim Seizinger, Thomas E. Bishop, Radu Timofte, et al. Lens-to-lens bokeh effect transformation. NTIRE 2023 challenge report. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 2, 3, 4, 8, 9, 10, 11, 12

[13] Marcos V. Conde, Steven McDonagh, Matteo Maggioni, Ales Leonardis, and Eduardo Pérez-Pellitero. Model-based image signal processors via learnable dictionaries. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(1):481–489, Jun. 2022. 2

[14] Marcos V Conde, Florin Vasluianu, Javier Vazquez-Corral, and Radu Timofte. Perceptual image enhancement for smartphone real-time applications. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1848–1858, 2023. 2, 4

[15] Marcos V Conde, Eduard Zamfir, Radu Timofte, et al. Efficient deep models for real-time 4k image super-resolution. NTIRE 2023 benchmark and report. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 3

[16] Wenyan Cong, Jianfu Zhang, Li Niu, Liu Liu, Zhixin Ling, Weiyuan Li, and Liqing Zhang. Dovenet: Deep image harmonization via domain verification. In *CVPR*, 2020. 3

[17] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 12

[18] Jinjin Gu, Haoming Cai, Chao Dong, Jimmy S Ren, Radu Timofte, Yuan Gong, Shanshan Lao, Shuwei Shi, Jiahao Wang, Sidi Yang, et al. Ntire 2022 challenge on perceptual image quality assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 951–967, 2022. 4

[19] 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. 2

[20] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 4

[21] Andrey Ignatov, Grigory Malivenko, Radu Timofte, Lukasz Treszczotko, Xin Chang, Piotr Ksiazek, Michal Lopuszynski, Maciej Pioro, Rafal Rudnicki, Maciej Smyl, et al. Efficient single-image depth estimation on mobile devices, mobile ai & aim 2022 challenge: report. In *Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27,*

2022, Proceedings, Part III, pages 71–91. Springer, 2023. 2, 5

[22] Andrey Ignatov, Jagruti Patel, and Radu Timofte. Rendering natural camera bokeh effect with deep learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 418–419, 2020. 2, 3, 5

[23] Andrey Ignatov, Radu Timofte, Shuai Liu, Chaoyu Feng, Furui Bai, Xiaotao Wang, Lei Lei, Ziyao Yi, Yan Xiang, Zibin Liu, et al. Learned smartphone isp on mobile gpus with deep learning, mobile ai & aim 2022 challenge: report. In *Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III*, pages 44–70. Springer, 2023. 2

[24] Andrey Ignatov, Radu Timofte, Ming Qian, Congyu Qiao, Jiamin Lin, Zhenyu Guo, Chenghua Li, Cong Leng, Jian Cheng, Juewen Peng, et al. Aim 2020 challenge on rendering realistic bokeh. In *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 213–228. Springer, 2020. 3

[25] Andrey Ignatov, Radu Timofte, Jin Zhang, Feng Zhang, Gaocheng Yu, Zhe Ma, Hongbin Wang, Minsu Kwon, Haotian Qian, Wentao Tong, et al. Realistic bokeh effect rendering on mobile gpus, mobile ai & aim 2022 challenge: report. In *Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III*, pages 153–173. Springer, 2023. 2, 3, 12

[26] Jiaxi Jiang, Kai Zhang, and Radu Timofte. Towards flexible blind jpeg artifacts removal. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4997–5006, 2021. 5

[27] Xiaoyang Kang, Xianhui Lin, Kai Zhang, Zheng Hui, Wangmeng Xiang, Jun-Yan He, Xiaoming Li, Peiran Ren, Xuansong Xie, Radu Timofte, et al. NTIRE 2023 video colorization challenge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 3

[28] Zhanghan Ke, Jiayu Sun, Kaican Li, Qiong Yan, and Rynson WH Lau. Modnet: Real-time trimap-free portrait matting via objective decomposition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 1140–1147, 2022. 10

[29] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 8, 9, 10, 12

[30] Xiangyu Kong, Fan Wang, Dafeng Zhang, Jinlong Wu, and Zikun Liu. Nafbet: Bokeh effect transformation with parameter analysis block based on nafnet. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 5, 8

[31] Orest Kupyn, Tetiana Martyniuk, Junru Wu, and Zhangyang Wang. Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8878–8887, 2019. 12

[32] Brian Lee, Fei Lei, Huaijin Chen, and Alexis Baudron. Bokeh-loss gan: multi-stage adversarial training for realistic edge-aware bokeh. In *Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part II*, pages 619–634. Springer, 2023. 2

[33] Yawei Li, Yulun Zhang, Luc Van Gool, Radu Timofte, et al. NTIRE 2023 challenge on efficient super-resolution: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 3

[34] Yawei Li, Yulun Zhang, Luc Van Gool, Radu Timofte, et al. NTIRE 2023 challenge on image denoising: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 3

[35] Lu Liu, Lei Zhou, and Yuhan Dong. Bokeh rendering based on adaptive depth calibration network. *arXiv preprint arXiv:2302.10808*, 2023. 2

[36] Rosanne Liu, Joel Lehman, Piero Molino, Felipe Petroski Such, Eric Frank, Alex Sergeev, and Jason Yosinski. An intriguing failing of convolutional neural networks and the coordconv solution. *Advances in neural information processing systems*, 31, 2018. 5

[37] Xiaohong Liu, Xiongkuo Min, Wei Sun, Yulun Zhang, Kai Zhang, Radu Timofte, Guangtao Zhai, Yixuan Gao, Yuqin Cao, Tengchuan Kou, Yunlong Dong, Ziheng Jia, et al. NTIRE 2023 quality assessment of video enhancement challenge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 3

[38] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 10

[39] Xianrui Luo, Juewen Peng, Ke Xian, Zijin Wu, and Zhiguo Cao. Bokeh rendering from defocus estimation. In *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 245–261. Springer, 2020. 2

[40] Ziwei Luo, Fredrik K Gustafsson, Zheng Zhao, Jens Sjölund, and Thomas B Schön. Image restoration with mean-reverting stochastic differential equations. *arXiv preprint arXiv:2301.11699*, 2023. 11, 12

[41] Ziwei Luo, Fredrik K Gustafsson, Zheng Zhao, Jens Sjölund, and Thomas B Schön. Refusion: Enabling large-size realistic image restoration with latent-space diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 11

[42] Oliver Nalbach, Elena Arabadzhiyska, Dushyant Mehta, H-P Seidel, and Tobias Ritschel. Deep shading: convolutional neural networks for screen space shading. In *Computer graphics forum*, volume 36, pages 65–78. Wiley Online Library, 2017. 2

[43] Juewen Peng, Zhiguo Cao, Xianrui Luo, Hao Lu, Ke Xian, and Jianming Zhang. Bokehme: When neural rendering meets classical rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16283–16292, 2022. 2

[44] Juewen Peng, Xianrui Luo, Ke Xian, and Zhiguo Cao. Interactive portrait bokeh rendering system. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 2923–2927. IEEE, 2021. 2

[45] Juewen Peng, Zhiyu Pan, Chengxin Liu, Xianrui Luo, Huiqiang Sun, Liao Shen, Ke Xian, and Zhiguo Cao. Selective bokeh effect transformation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 5, 8

[46] Ming Qian, Congyu Qiao, Jiamin Lin, Zhenyu Guo, Chenghua Li, Cong Leng, and Jian Cheng. Bggan: Bokeh-glass generative adversarial network for rendering realistic bokeh. In *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 229–244. Springer, 2020. 2, 12

[47] Xuebin Qin, Hang Dai, Xiaobin Hu, Deng-Ping Fan, Ling Shao, and Luc Van Gool. Highly accurate dichotomous image segmentation. In *ECCV*, 2022. 9

[48] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 11

[49] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. 2, 4, 8, 9, 12

[50] Lingyan Ruan, Bin Chen, Jizhou Li, and Miuling Lam. Learning to deblur using light field generated and real defocus images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16304–16313, 2022. 8

[51] Tim Seizinger, Marcos V Conde, Manuel Kolmet, Tom E Bishop, and Radu Timofte. Efficient multi-lens bokeh effect rendering and transformation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 2, 3, 5, 8, 9, 10, 11, 12

[52] Xiaoyong Shen, Aaron Hertzmann, Jiaya Jia, Sylvain Paris, Brian Price, Eli Shechtman, and Ian Sachs. Automatic portrait segmentation for image stylization. In *Computer Graphics Forum*, volume 35, pages 93–102. Wiley Online Library, 2016. 2

[53] Xiaoyong Shen, Xin Tao, Hongyun Gao, Chao Zhou, and Jiaya Jia. Deep automatic portrait matting. In *European conference on computer vision*, pages 92–107. Springer, 2016. 2

[54] Alina Shutova, Egor Ershov, Georgy Perevozchikov, Ivan A Ermakov, Nikola Banic, Radu Timofte, Richard Collins, Maria Efimova, Arseniy Terekhin, et al. NTIRE 2023 challenge on night photography rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 3

[55] Yuda Song, Yang Zhou, Hui Qian, and Xin Du. Rethinking performance gains in image dehazing networks. *arXiv preprint arXiv:2209.11448*, 2022. 11

[56] Florin-Alexandru Vasluianu, Tim Seizinger, and Radu Timofte. Wsrd: A novel benchmark for high resolution image shadow removal. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 3

[57] Florin-Alexandru Vasluianu, Tim Seizinger, Radu Timofte, et al. NTIRE 2023 image shadow removal challenge report. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 3

[58] Neal Wadhwa, Rahul Garg, David E Jacobs, Bryan E Feldman, Nori Kanazawa, Robert Carroll, Yair Movshovitz-Attias, Jonathan T Barron, Yael Pritch, and Marc Levoy. Synthetic depth-of-field with a single-camera mobile phone. *ACM Transactions on Graphics (ToG)*, 37(4):1–13, 2018. 2

[59] Longguang Wang, Yulan Guo, Yingqian Wang, Juncheng Li, Shuhang Gu, Radu Timofte, et al. NTIRE 2023 challenge on stereo image super-resolution: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 3

[60] Lijun Wang, Xiaohui Shen, Jianming Zhang, Oliver Wang, Zhe Lin, Chih-Yao Hsieh, Sarah Kong, and Huchuan Lu. Deeplens: shallow depth of field from a single image. *arXiv preprint arXiv:1810.08100*, 2018. 2

[61] Wenxiao Wang, Lu Yao, Long Chen, Binbin Lin, Deng Cai, Xiaofei He, and Wei Liu. Crossformer: A versatile vision transformer hinging on cross-scale attention. In *International Conference on Learning Representations (ICLR)*, 2022. 8

[62] Yingqian Wang, Longguang Wang, Zhengyu Liang, Jungang Yang, Radu Timofte, Yulan Guo, et al. NTIRE 2023 challenge on light field image super-resolution: Dataset, methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 3

[63] Zhifeng Wang and Aiwen Jiang. A dense prediction vit network for single image bokeh rendering. In *Pattern Recognition and Computer Vision: 5th Chinese Conference, PRCV 2022, Shenzhen, China, November 4–7, 2022, 2022, Proceedings, Part IV*, pages 213–222. Springer, 2022. 2

[64] Jiaze Wu, Changwen Zheng, Xiaohui Hu, and Fanjiang Xu. Rendering realistic spectral bokeh due to lens stops and aberrations. *The Visual Computer*, 29:41–52, 2013. 2

[65] Lei Xiao, Anton Kaplanyan, Alexander Fix, Matt Chapman, and Douglas Lanman. Deepfocus: Learned image synthesis for computational display. In *ACM SIGGRAPH 2018 Talks*, pages 1–2. 2018. 2

[66] Yang Yang, Haiting Lin, Zhan Yu, Sylvain Paris, and Jingyi Yu. Virtual dslr: High quality dynamic depth-of-field synthesis on mobile platforms. *Electronic Imaging*, 28:1–9, 2016. 2

[67] Zhihao Yang, Wenyi Lian, and Siyuan Lai. Bokehornot: Transforming bokeh effect with image transformer and lens metadata embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 5, 10

[68] Pierluigi Zama Ramirez, Fabio Tosi, Luigi Di Stefano, Radu Timofte, et al. NTIRE 2023 challenge on hr depth from images of specular and transparent surfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 3

[69] Eduard Zamfir, Marcos V Conde, and Radu Timofte. Towards real-time 4k image super-resolution. In *Proceedings*

of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2023. 3

[70] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5728–5739, 2022. 4, 10, 12

[71] 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. 4, 7

[72] Xuaner Zhang, Kevin Matzen, Vivien Nguyen, Dillon Yao, You Zhang, and Ren Ng. Synthetic defocus and look-ahead autofocus for casual videography. *arXiv preprint arXiv:1905.06326*, 2019. 2

[73] Yulun Zhang, Kai Zhang, Zheng Chen, Yawei Li, Radu Timofte, et al. NTIRE 2023 challenge on image super-resolution (x4): Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2023. 3

[74] Bingke Zhu, Yingying Chen, Jinqiao Wang, Si Liu, Bo Zhang, and Ming Tang. Fast deep matting for portrait animation on mobile phone. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 297–305, 2017. 2