Realistic Compound-Lens Defocus Blur Synthesis

Yunkyu Lee[1], Woohyeok Kim[1], and Sunghyun Cho[1]

POSTECH, Pohang, Korea
{lyk1012,woohyeok,s.cho}@postech.ac.kr

Abstract. Defocus blur degrades fine image structures and limits visual perception, which can adversely affect downstream vision tasks. Although recent deep learning deblurring methods have achieved strong performance, their effectiveness depends on training data and often degrades across cameras and lenses due to limited optical diversity and realism in existing datasets. In this paper, we propose a pipeline for synthesizing realistic defocus deblurring datasets for diverse compound lenses. It integrates efficient wave-optics PSF computation via Debye CZT propagation, depth-aware defocus rendering with occlusion handling, and blur synthesis in the radiometrically linear space with camera ISP simulation. This unified pipeline enables the scalable generation of photorealistic defocus datasets with diverse lens characteristics. Using our pipeline, we generate CLDefocus, a large-scale synthetic dataset containing lens-diverse defocus image pairs. We further analyze the limitations of real-captured defocus datasets and show that such imperfections can bias full-reference evaluation. Extensive experiments demonstrate that models trained on CLDefocus achieve improved cross-device generalization compared to models trained on existing real and synthetic datasets. Code and dataset are available at: `https://github.com/lykelee/CLDefocus`.

Keywords: Defocus Deblurring · Optical Modeling · Dataset Synthesis

1 Introduction

Defocus blur arises when light rays from an object point fail to converge on the sensor plane, forming a blurred region known as the circle of confusion (CoC). Optical systems employing large apertures often exhibit a shallow depth of field, which naturally leads to defocus blur. Although such blur can be deliberately exploited for aesthetic purposes in photography, it is generally undesirable in computer vision, as it degrades fine details and hampers visual understanding [43]. This degradation directly affects downstream tasks such as object detection, face recognition, and semantic segmentation [18, 48].

Recent advances in deep learning have led to significant progress in defocus deblurring [1, 23, 36, 37, 41, 50], surpassing traditional blur-map-based deconvolution methods [20, 45]. However, the effectiveness of these methods is fundamentally constrained by the availability of realistic and diverse training datasets. This dependency naturally raises the question of how defocus datasets are constructed, and what limitations they impose on generalization.

Real-captured defocus datasets are typically constructed by repeatedly capturing the same scene with different apertures [1, 23]. This process scales poorly across cameras and lenses, limiting the diversity of optical characteristics and hindering cross-device generalization [49]. Moreover, changing optical settings often introduces brightness variations and spatial misalignment [23, 25], and the resulting ground-truth images may still contain residual defocus blur or alignment errors.

Meanwhile, synthetic datasets have also been investigated. Early methods approximated defocus blur using disk or Gaussian kernels [11, 35, 46], which fail to capture the lens-specific characteristics of real optics. More recent works compute physically derived point spread functions (PSFs) via wave-optics simulations [10, 15, 26, 32]. However, these approaches often incur substantial computational cost and require empirically tuned sampling densities to avoid aliasing. Such requirements hinder scalable PSF generation across diverse lens parameters, and most prior works therefore consider only a limited set of lens designs or fixed configurations rather than large-scale depth-varying defocus datasets. Moreover, defocus blur is often synthesized directly in the non-linear sRGB color space without modeling the camera image signal processor (ISP) [40], reducing photometric realism.

In this paper, we propose a realistic compound lens defocus blur synthesis framework. It combines efficient and stable wave-optics PSF computation, depth-layered occlusion-aware defocus rendering, and linear-space blur synthesis with camera ISP simulation. This unified pipeline enables scalable synthesis of photorealistic defocus datasets across diverse lenses and focus configurations. It also avoids the alignment artifacts inherent in real-captured datasets while improving the photometric realism and computational efficiency of synthetic defocus generation.

Constructing such a realistic pipeline presents several practical challenges. Wave-optics PSF computation under varying apertures and focus distances is computationally demanding and highly sensitive to sampling conditions [44]. We address this by adopting a Debye CZT-based propagation framework [16, 24] with explicit sampling criteria for stable and efficient PSF computation. Rendering spatially varying defocus introduces another challenge because PSFs change continuously with scene depth, making direct per-pixel convolution with depth-varying PSFs computationally expensive. To enable efficient rendering, we discretize depth using signed CoC values and perform layered compositing with occlusion handling [14, 19]. Photometric inconsistency also arises when blur is synthesized directly in the non-linear sRGB color space. To address this issue, we perform blur synthesis in the radiometrically linear color space with explicit ISP simulation rather than naive convolution in the sRGB space [40].

Using the proposed pipeline, we generate *CLDefocus*, a large-scale synthetic dataset across diverse lens and focus configurations. It consists of 40,000 training pairs, 1,000 validation pairs, and 1,000 test pairs. We also analyze the limitations of real defocus datasets and show that geometric and photometric inconsistencies in ground-truth pairs can bias full-reference evaluation, highlighting the need for physically grounded synthetic data. Extensive experiments demonstrate that training on CLDefocus improves cross-device generalization compared to existing real and synthetic datasets.

Our contributions are summarized as follows:

– We propose an efficient wave-optics PSF generation method for compound lenses based on the Debye CZT with explicit sampling criteria.
– We develop a realistic defocus rendering pipeline with depth-aware occlusion handling and blur synthesis in the radiometrically linear space with ISP-aware image formation.
– We present CLDefocus, a large dataset with lens-diverse synthetic defocus image pairs. We also analyze the limitations of real-captured defocus datasets and show that training on CLDefocus improves cross-device generalization across multiple real defocus benchmarks.

2 Related Work

2.1 Defocus Deblurring Methods

Early works [2, 22, 46, 52] primarily focus on estimating defocus maps from images, which are then used with a conventional non-blind deconvolution method [20]. These methods depend on hand-crafted priors or simplified blur-kernel assumptions, limiting their robustness in complex real-world scenarios. With the increasing availability of training data, learning-based methods have become dominant [1, 23, 27, 36, 37, 41, 42, 47, 50]. End-to-end networks learn spatially varying blur patterns directly from data and significantly outperform traditional approaches. However, their performance strongly depends on the diversity and realism of the training data, yet lens-diverse and physically accurate defocus datasets remain relatively limited.

2.2 Real-Captured Defocus Datasets

Acquiring high-quality blurred–sharp image pairs for defocus deblurring is inherently challenging, as it requires repeated captures of the same static scene under different aperture settings while keeping the camera pose fixed, limiting the scale and lens diversity of real-captured datasets. In practice, changing the aperture between captures often introduces brightness variations, spatial misalignment, and residual defocus even in the corresponding sharp images [23, 25].

Despite these challenges, several real-captured defocus datasets have been proposed, but they remain limited in scale, camera diversity, or image formation. RTF [11] and RealDOF [23] provide only test sets with a small number of scenes, and thus cannot be used to train learning-based defocus deblurring models. DPDD [1] provides 500 blurred–sharp pairs captured by a Canon DSLR with dual-pixel (DP) sensors to obtain sub-aperture views and corresponding all-in-focus ground truth. However, its scale remains limited and it is restricted to a single camera system. LFDOF [41] increases dataset scale using light-field cameras, yet its image formation process differs from that of conventional cameras, potentially limiting generalization. SDD [25] captures data in dynamic driving scenarios, where blurred and sharp images are not perfectly aligned, requiring a dedicated training framework to handle the misalignment.

2.3 Synthetic Defocus and Lens Blur Modeling

Synthetic defocus generation has been explored as an alternative to real data. Early methods relied on simplified thin-lens models, approximating blur with disk or Gaussian kernels [2, 22, 46]. While theoretically simple, these approximations fail to capture the lens-specific characteristics of real optical systems.

More physically grounded approaches compute PSFs using wave-optics simulations. Some works adopt simple Fourier transform-based PSF formulations [9, 31], while others employ more rigorous diffraction models such as the Huygens' principle [13], which is often formulated as the Rayleigh–Sommerfeld integral [10, 26, 32]. Although physically grounded and theoretically accurate, the Huygens' principle requires stringent sampling densities to avoid aliasing artifacts. In practice, increasing the defocus distance or aperture size necessitates finer sampling grids, leading to rapidly growing computational cost. As a result, most prior studies restrict their evaluation to a limited number of lenses or primarily consider in-focus imaging, rather than systematically generating large-scale, depth-varying defocus datasets across diverse compound lenses.

In addition, many synthetic approaches perform blur rendering using simplified convolution models that do not fully account for occlusions in depth-of-field imaging or the camera imaging pipeline. ISP-aware modeling has been shown to be important for realistic blur synthesis [40], yet it is often neglected. In contrast, our work unifies physically grounded PSF computation across diverse compound lenses, depth-aware occlusion-consistent rendering, and ISP-aware image formation within a scalable dataset synthesis framework.

3 PSF Computation

Our realistic blur synthesis pipeline primarily relies on physically accurate point spread functions (PSFs) across diverse lenses, focus settings, scene depths, and field positions. In this section, we describe how our pipeline enables efficient and stable computation of those PSFs. Unlike thin-lens approximations based on geometric optics [22], we model wave propagation through a compound lens. To this end, we adopt a hybrid approach that combines ray tracing with wave-optics propagation [10, 26]. Specifically, we acquire wavefront samples on the exit pupil sphere via ray tracing, fit them with Zernike polynomials [34] to obtain a smooth analytical representation, and compute the sensor-plane wave field using the Debye formulation accelerated by the chirp Z-transform (CZT). The resulting field yields lens-specific PSFs, enabling scalable generation across diverse lens configurations and defocus conditions. We first introduce the Debye CZT for efficient PSF computation, followed by its application on compound lenses.

3.1 Debye CZT

We introduce the diffraction model and numerical implementation used for PSF generation. A defocused PSF is the intensity of the wave formed by a lens on a sensor plane displaced from the ideal focus. This setting is naturally described by the Debye formulation [24], which models diffraction of converging light near focus. We first present its Fourier form, and then discuss the sampling requirement and efficient implementation for a desired sensor region of interest (ROI).

The Debye formulation expresses the propagated wave field $\mathbf{E}$ at an image point as the integral of the plane-wave spectrum over the exit pupil (XP) sphere, as illustrated on the left side of Fig. 1. For an on-axis point source, Leutenegger et al. [24] rewrite the Debye formulation as a Fourier transform:

$$\mathbf{E}(x, y, z) = -\frac{jf}{\lambda k^2}\mathcal{F}\left\{\mathbf{E}_t(\theta, \phi)\frac{\exp(jk_z z)}{\cos\theta}\right\}(x, y, z), \tag{1}$$

where $\mathbf{E}_t$ denotes the wave field on the XP sphere, f is the focal length, λ is the wavelength, k is the wave number, z is the axial defocus displacement, and j denotes the imaginary unit. The Fourier transform is with respect to the x, y coordinates of the transverse wave vectors $\mathbf{k}$, which are proportional to the wave number and the normal vector on the XP sphere. This Fourier form allows the sensor-plane field to be computed in the Fourier domain after discretization.

For off-axis point sources, we adopt the approach proposed by Cai et al. [6]. We first rotate the coordinate system to align the focal point with the new optical axis. Then we apply the on-axis Debye formulation with the image plane tilted with respect to the new coordinates. This extends Eq. (1) and leads to a more involved formula. Details are provided in the supplementary material.

When the continuous Debye formulation is discretized onto a finite wave grid, a sufficient number of samples is required to prevent aliasing. Following the analysis in Leutenegger et al. [24], the sampling condition is given by:

$$N > N_{\mathrm{inf}}, \quad N_{\mathrm{inf}} = \frac{4\mathrm{NA}^2}{\sqrt{n_t^2 - \mathrm{NA}^2}}\frac{|z|}{\lambda}, \tag{2}$$

where N is the number of samples per coordinate, N_{inf} is the lower bound of N, NA denotes the numerical aperture, and n_t is the refractive index of the transmission medium. This condition originates from the phase factor $\exp(jk_z z)$ in Eq. (1) and implies that higher NA and stronger defocus require more samples. It does not account for aliasing caused by rapid variations in $\mathbf{E}_t$, which can be non-negligible when lens aberrations are large. We therefore set $N = 2N_{\mathrm{inf}}$, which suppresses aliasing across all lenses and defocus levels considered.

When implementing the Fourier-based computation of the Debye formulation, a straightforward approach is to use the fast Fourier transform (FFT). However, in the standard FFT, the output sampling pitch and ROI cannot be chosen freely, as the output grid is fixed by the input wavefront sampling grid [44]. In contrast, our PSFs must be evaluated on a sensor-aligned grid

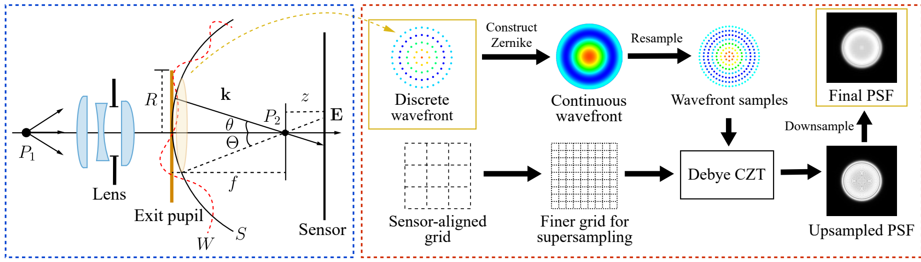


Fig. 1: Overview of our PSF computation process. The left shows the physical setup of the Debye formulation. A point source P_1 emits light, which passes through the lens and forms a wavefront W. S is the exit pupil sphere that is centered at a focal point P_2 and passes through the exit pupil center. We model W as a converging spherical wave on S with deviations (aberrations). The Debye formulation integrates all wave field within a cap with half-angle Θ to obtain the wave field on the image plane with defocus z. The right shows the PSF computation process with the CZT. We acquire discrete wavefront samples from the ray tracing result. Then we fit them with Zernike polynomials and resample a sample grid required by the CZT. We then perform the Debye CZT on a finer grid, obtaining a supersampled PSF. Finally, we downsample it to obtain a PSF that matches the sensor resolution.

whose sampling pitch matches the sensor pixel pitch. To this end, we employ the chirp Z-transform (CZT) [16,24] to flexibly control both the sampling resolution and the spatial extent for sampling. The CZT allows flexible ROI selection and sampling resolution while maintaining computational efficiency via Bluestein's algorithm [3,38]. We refer to this implementation as the **Debye CZT**. Detailed derivations are provided in the supplementary material.

The Debye formulation is often written in vectorial form, where the wave field is vector-valued and polarization is explicitly modeled. While this is important for high-NA optics and polarization-sensitive imaging, it is unnecessary in our setting. We therefore use a scalar diffraction approximation by treating wave fields as scalar-valued. This choice is supported by two considerations. First, the photographic lenses used in our study have a low NA, as described in Sec. 4.1. Second, natural scenes typically exhibit negligible polarization effects [21].

3.2 Computing Lens PSFs

While the Debye CZT has been used for microscopy with on-axis and analytically defined pupil functions [29], existing literature provides limited guidance on how to apply it to photographic compound lenses. We use a three-stage procedure to compute depth- and field-dependent PSFs of a compound lens, given the focusing distance, object depth, field position, and sensor resolution. As illustrated on the right side of Fig. 1, the procedure consists of: (1) wavefront sampling on the exit pupil sphere, where discrete wavefront samples are obtained from the lens prescription through ray tracing; (2) Zernike-based wavefront representation, which reconstructs a smooth analytical wavefront from the samples; and (3) diffraction computation via the Debye CZT, which propagates the wavefront to the sensor plane and produces a PSF for the desired pixel pitch and region of interest. Further details are provided in the supplementary material.

Wavefront Sampling on the Exit Pupil Sphere. For a given lens design, we perform dense ray tracing from a point source at the given depth and field through the lens. We project all traced rays onto the exit pupil (XP) sphere and compute the optical path difference (OPD) for each ray. The OPDs are then converted into phase delays, yielding discrete wave field samples. Furthermore, to handle complicated pupil shapes of off-axis fields, we additionally calculate implicit function values to indicate whether a ray is blocked inside the lens. They are used to reconstruct a ray clipping mask that determines the pupil shape.

Zernike-Based Wavefront Representation. Ray tracing provides wavefront samples only at specific discrete positions, which do not generally form uniform grids that can be used for CZT. Hence, we fit the sampled wavefront with Zernike polynomials to construct a continuous analytical representation. We then resample the wavefront onto the sampling grid required by the CZT. This model captures lens-specific aberrations and serves as the input field $\mathbf{E}_t$ in Eq. (1).

Diffraction Computation via Debye CZT. We evaluate the sensor-plane wave field using the Debye formulation accelerated with the chirp Z-transform (CZT). Let the desired PSF array be $P \in \mathbb{R}^{k\times k}$ on the sensor grid. Although the CZT allows direct sampling on this grid, insufficient output sampling may introduce aliasing artifacts. To mitigate this, we upsample the output grid by a factor $u > 1$, compute the wave field E_u, and form the intensity $P_u = |E_u|^2$. The final PSF P is obtained by downsampling P_u by a factor of u to match the target resolution. In practice, we use $u = 5$, which removes observable aliasing in all tested configurations.

4 Defocus Dataset Synthesis

In this section, we present a realistic blur synthesis pipeline for generating defocused images from compound lens designs. We synthesize depth-dependent defocus blur using the PSFs generated in Sec. 3, constructing physically consistent blurred–sharp training and test pairs from sharp RAW images through a depth-aware rendering pipeline (Fig. 2). Our synthesis operates across RAW, linear RGB, and non-linear sRGB color spaces, following the ISP pipeline of RSBlur [40]. Blur synthesis is performed in linear RGB, while the final pairs are produced in the non-linear sRGB color space.

4.1 Lens Collection & Filtering

To generate diverse PSFs, we collect compound lens designs from the publicly available collection [39]. We download Zemax files under “Photographic primes”

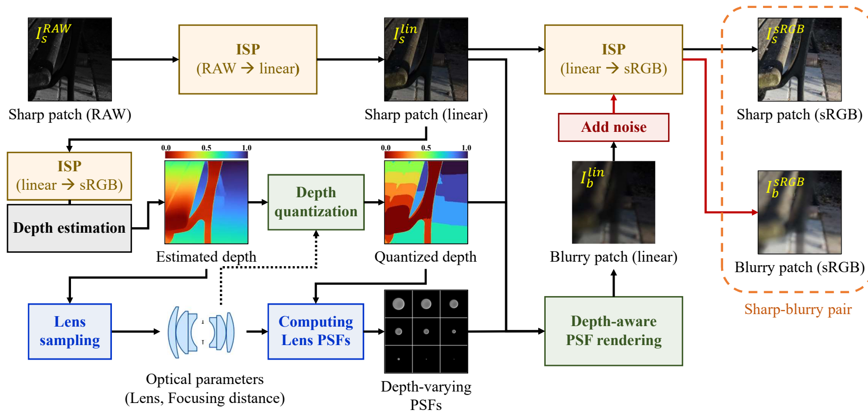


Fig. 2: Overview of our data synthesis pipeline. From a sharp RAW image, we first sample a RAW patch and convert it to linear RGB via a partial ISP. Using the corresponding depth map, we sample a lens and a focusing distance that satisfy our constraints on blur size and practical PSF computation. We then quantize the depths while minimizing defocus discontinuity and build a multi-layered representation. Next, we compute PSFs for varying depths using the Debye CZT and perform depth-of-field imaging in the linear space, yielding the blurred patch. Finally, we synthesize noise and apply the ISP to obtain the blurred patch in the sRGB space. We also obtain the corresponding sharp patch by passing the linear sharp patch through the ISP.

and "Photographic zooms", retaining 1,281 files after removing incompatible ones. To reflect practical photographic optics, we filter the designs by basic properties such as F-number and total track length, keeping only lenses within typical photography ranges. We further exclude lenses with NA ≥ 0.6, which fall outside standard photographic regimes and require excessive sampling under Eq. (2), as well as designs exhibiting severe aberrations that destabilize wavefront reconstruction and PSF computation. After filtering, 700 lens designs remain for PSF generation. The detailed criteria are provided in the supplementary material.

4.2 Depth Estimation & Lens Sampling

Given a collection of sharp RAW source images, which we describe in Sec. 5, we sample a RAW patch from each image as a sharp source patch. The sampled RAW patch $I_s^{\mathrm{RAW}} \in \mathbb{R}^{h\times w\times 1}$ is then converted to the linear space via the partial ISP, which applies demosaicing, white balancing, and color correction. This produces the linear sharp patch $I_s^{\mathrm{lin}} \in \mathbb{R}^{h\times w\times 3}$. For training pairs, we also apply a few augmentations (rotation, flip, resizing, and exposure scaling) at this stage. We obtain the sRGB patch I_s^{sRGB} via the forward ISP. For each patch, we also obtain the corresponding depth map. To obtain the depth map, we run a monocular estimator, Depth Pro [4] on the full-resolution sharp sRGB image (obtained via the forward ISP), and crop the resulting metric depth map to match the linear sharp patch.

Next, we randomly sample a compound lens L and a focusing distance f for the patch. To ensure feasible rendering, candidate lenses are evaluated using paraxial approximations. Lenses are discarded if the maximum circle of confusion (CoC) over the patch exceeds a predefined threshold (limiting PSF size) or if the required sampling number for the Debye CZT becomes excessive. After this filtering, a valid (L, f) pair is sampled from the remaining lenses. The detailed sampling process and filtering criteria are provided in the supplementary material.

4.3 Depth-Aware PSF Rendering

Computing distinct PSFs for all continuous depth values is computationally infeasible. We therefore quantize the depth map into K discrete layers based on the *signed* CoC, which distinguishes points in front of and behind the focal plane. Depth values whose signed CoC difference is smaller than one sensor pixel are grouped into the same layer to reduce quantization artifacts.

This produces layered color images $\{C_i\}_{i=1}^K$ and opacity masks $\{A_i\}_{i=1}^K$:

$$C_i \in \mathbb{R}^{h\times w\times 3}, \quad A_i \in \{0,1\}^{h\times w}. \tag{3}$$

Here, C_i is the color image of the i-th layer, and A_i indicates its spatial support. We note that regions occluded by foreground layers might contribute to the blurred result. Hence, we inpaint those regions and set opacity masks to 1.

Blur synthesis is performed entirely in the radiometrically linear space. For each layer i, we retrieve the corresponding depth-dependent PSF $P_i \in \mathbb{R}^{k\times k}$ and convolve it with C_i. Since defocus with multiple depths cannot be expressed as a single spatially invariant convolution, we adopt the layered compositing formulation [14, 19] and accumulate contributions from back to front:

$$C_i^{\text{blur}} = P_i * C_i + (1 - P_i * A_i) \cdot C_{i-1}^{\text{blur}}, \tag{4}$$

where $*$ denotes convolution and $C_i^{\text{blur}} \in \mathbb{R}^{h\times w\times 3}$ is the intermediate composite after processing layers $1, \ldots, i$. The final blurred linear image is $I_b^{\text{lin}} = C_K^{\text{blur}}$.

To ensure photometric consistency, I_b^{lin} is converted to RAW space via the inverse partial ISP, sensor noise is synthesized, and the forward ISP is applied to obtain the blurred sRGB image $I_b^{\text{sRGB}} \in \mathbb{R}^{h\times w\times 3}$. The sharp counterpart I_s^{sRGB} is generated by passing I_s^{lin} through the same ISP without degradation, yielding the blurred–sharp pair $(I_b^{\text{sRGB}}, I_s^{\text{sRGB}})$. Random sampling of source images, augmentations, lenses, and focusings enables diverse training and testing pairs. Additional implementation details are provided in the supplementary material.

5 Experiments

Implementation. For dataset synthesis, we implemented the PSF computation and the synthesis pipeline using JAX [5], a Python package for GPU-accelerated numerical computations. We also adopted ZERNIPAX [12] for efficient Zernike

Table 1: Quantitative results on full-reference metrics (PSNR, SSIM, LPIPS [51]). The best and second-best results are highlighted in bold and underlined, respectively.

Test set	CLDefocus			RTF			RealDOF			DPDD		
Train set	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
CLDefocus	**32.16**	**0.865**	**0.201**	**26.62**	**0.845**	0.242	24.74	0.764	**0.303**	24.73	0.776	0.265
SYNDOF	27.58	0.790	0.296	24.55	0.743	0.261	21.64	0.654	0.474	23.77	0.743	0.315
DPDD	28.27	0.825	0.263	25.95	0.833	**0.207**	**25.03**	**0.771**	0.335	**26.11**	**0.817**	**0.223**

Table 2: Quantitative results on no-reference metrics (NIQE [30], MUSIQ [17], TOPIQ [7]). The best and second-best results are highlighted in bold and underlined, respectively.

Test set	CLDefocus			RTF			RealDOF			DPDD		
Train set	NIQE↓	MUSIQ↑	TOPIQ↑	NIQE↓	MUSIQ↑	TOPIQ↑	NIQE↓	MUSIQ↑	TOPIQ↑	NIQE↓	MUSIQ↑	TOPIQ↑
CLDefocus	7.724	**39.850**	**0.348**	4.263	**59.144**	0.533	**5.029**	**35.787**	**0.303**	**4.657**	56.696	0.487
SYNDOF	12.048	36.904	0.316	**3.855**	57.279	**0.539**	6.285	23.905	0.241	5.067	55.819	0.493
DPDD	**7.508**	38.961	0.345	4.196	58.822	0.531	6.030	31.350	0.270	4.746	**59.473**	**0.507**

polynomial evaluation. We primarily used NRKNet [36] as a deblurring model. We followed its original training protocols as much as possible. For example, NRKNet was originally trained for 4,000 epochs with batch size 4 for DPDD, which has 350 training pairs. We used $350{,}000 = 4{,}000 \cdot 350/4$ iterations to match the total iterations. We observe the same trend for other deblurring models [8, 37, 50] and provide the corresponding results in the supplementary material.

CLDefocus Dataset. We generate CLDefocus using our synthesis pipeline with train, validation, and test splits. For fair evaluation, we use disjoint splits for both sharp source images and lenses. As sharp sources, we employ sharp images of DPDD [1] in RAW format, following its original 350/74/76 split for train/validation/test. The 700 lenses in Sec. 4.1 are split into 420/140/140 lenses. This yields 40,000/1,000/1,000 train/validation/test pairs at 384×384 resolution. To mitigate residual blur in the ground-truth images, we generate 50 % extra patches and discard lower-quality pairs using sharpness filtering [1].

5.1 Comparison with Other Datasets

For comparison, we train NRKNet on three datasets: DPDD [1], SYNDOF [22], and our CLDefocus dataset. DPDD is a real-captured dataset collected with a limited camera configuration, whereas SYNDOF is a synthetic dataset constructed using simplified blur models. For fair comparison, we keep the training protocol identical across datasets. Each model is evaluated on our CLDefocus test set and existing benchmark datasets: RTF [11], RealDOF [23], and DPDD.

Tab. 1 presents the full-reference results, while Tab. 2 reports the no-reference results. On our test set, the CLDefocus-trained model achieves the best performance under both metric types, confirming the physical consistency and diver-

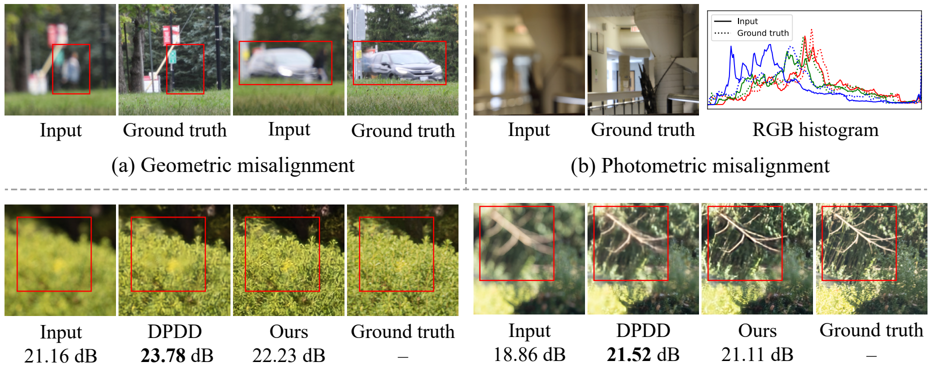
Fig. 3: Qualitative comparisons on the RealDOF dataset.

sity of our dataset. On real benchmarks, the CLDefocus-trained model generally yields lower full-reference scores than the DPDD-trained model. This behavior is attributable to imperfections in real ground-truth images, including residual defocus, brightness inconsistencies, and spatial misalignment, which bias pixel-wise measures such as PSNR and SSIM. We analyze the impact of these ground-truth imperfections in Sec. 5.2.

In contrast, the CLDefocus-trained model achieves overall higher no-reference scores, implying improved perceptual quality. This trend is also observed in Fig. 3, where our model restores sharper details with fewer artifacts. The DPDD-trained model tends to overfit to its specific lens configuration, limiting cross-lens generalization, while the SYNDOF-trained model generalizes poorly due to its simplified blur model. Overall, CLDefocus enhances cross-device generalization through physically grounded PSF modeling and scalable lens diversity. Additional results in the supplementary material further support this trend.

5.2 Limitations of Ground-Truth in Real Datasets

We analyze structural imperfections in real-captured ground-truth pairs. Real-captured defocus datasets inevitably contain residual inconsistencies between input and ground-truth (GT) images. Fig. 4-(a) illustrates geometric misalignment: scene objects may shift, deform, or newly appear between captures due to



(c) Qualitative results on the real dataset along with the corresponding PSNR scores.

Fig. 4: Limitations of ground truth in real-captured defocus datasets. Geometric misalignment caused by dynamic objects and photometric misalignment introduced by aperture changes lead to spatial and color inconsistencies, as reflected in the RGB histograms. These imperfections result in a mismatch between full-reference metrics and visual quality, where perceptually sharper restorations may receive lower metric scores.

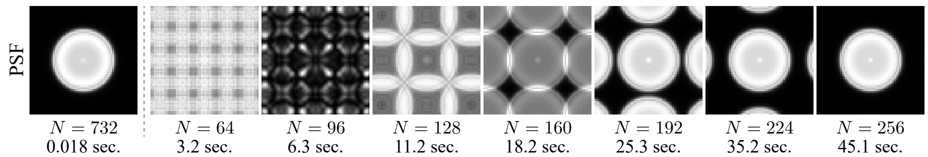
subject motion or scene dynamics. Even in carefully captured datasets such as DPDD [1] or post-processed datasets such as RealDOF [23], such spatial inconsistencies remain. Fig. 4-(b) further shows photometric misalignment. Changing apertures between captures often introduces brightness and color variations [23]. These discrepancies are visually evident and are reflected in the RGB histograms, where the channel-wise intensity distributions differ significantly between input and GT images. Moreover, some GT images still contain residual defocus blur, indicating that the GT itself may not represent a perfectly focused reference.

These inconsistencies directly affect full-reference evaluation. As shown in Fig. 4-(c), even when our result is perceptually sharper, pixel-wise metrics such as PSNR can favor predictions that remain slightly blurred, as they better match misaligned or partially blurred GT images. Consequently, models trained on such datasets may implicitly learn a conservative, blur-preserving bias, yielding higher full-reference scores despite inferior perceptual sharpness.

5.3 Debye CZT vs. Huygens' Principle

We examine the stability and computational efficiency of our Debye CZT framework by comparing it with a propagation method based on the Huygens' principle, which has been adopted in prior wave-optics PSF simulations [10, 26]. For this, we implement a method using the Rayleigh–Sommerfeld integral [28].

Sampling Condition. To validate the stability of our Debye CZT, we compare it with the Rayleigh–Sommerfeld with various sampling numbers. In both cases, we first construct the Zernike wavefront on the exit pupil sphere from the same



(a) Debye CZT (b) Rayleigh–Sommerfeld

Fig. 5: Comparison between the Debye CZT and the Rayleigh–Sommerfeld. The Debye CZT determines the sampling density and produces a stable, aliasing-free PSF without empirical tuning. In contrast, the Rayleigh–Sommerfeld provides no explicit sampling criterion, so we adjust the sampling density manually. Insufficient sampling leads to severe PSF warping and geometric distortion. Only sufficiently dense sampling yields a valid PSF, but at the cost of increased runtime, as indicated for each configuration.

lens design. The difference lies in the propagation step to the sensor plane: our method performs propagation via the Debye CZT under the explicit sampling condition derived in Sec. 3.1, whereas the Rayleigh–Sommerfeld directly sums the contribution of each sampled wavefront point to every sensor-plane point. We use "JP2017-003807_Example04P.zmx" in the lens collection [39]. The PSF is computed for an on-axis point at 0.8 m, with the sensor focused at infinity. We set the spatial extent of a PSF to 64×64. For a fair comparison, we use $5\times$ supersampling for the PSF computation in both methods.

For the Debye CZT, Eq. (2) yields $N_{\text{inf}} = 366$, and we set $N = 2N_{\text{inf}} = 732$ following our strategy described in Sec. 3.1. In contrast, as the Rayleigh–Sommerfeld does not provide well-known explicit sampling criteria, we start from $N = 64$ and gradually increase it by 32 to reach 256. As shown in Fig. 5, only $N = 256$ produces a valid PSF, while smaller N values exhibit aliasing manifested as PSF warping. This implies that we also need a proper sampling condition like Eq. (2) for the Rayleigh–Sommerfeld.

Computational Cost. We measured the runtime of PSF computation after evaluating wavefront samples. The Debye CZT took 0.018 s, whereas the Rayleigh–Sommerfeld took 45.1 s for $N = 256$. Computation times for other N values are stated in Fig. 5. The theoretical complexity of the Debye CZT is $\mathcal{O}((N+M)^2 \log(N+M))$, while the Rayleigh–Sommerfeld scales as $\mathcal{O}(N^2M^2)$, where M is the output grid size. Its quadratic dependence on both N and M makes the Rayleigh–Sommerfeld highly sensitive to sampling density. Detailed derivations and complexity analysis are provided in the supplementary material.

5.4 Examples of Generated PSFs

Fig. 6 shows examples of diverse PSFs computed by the Debye CZT. They show diverse shapes and textures of PSFs across different settings. In particular, PSFs of severe off-axis fields often have complex and asymmetric shapes. These imply that simplified blur models are insufficient to simulate realistic defocus blur. We provide more examples and analysis in the supplementary material.

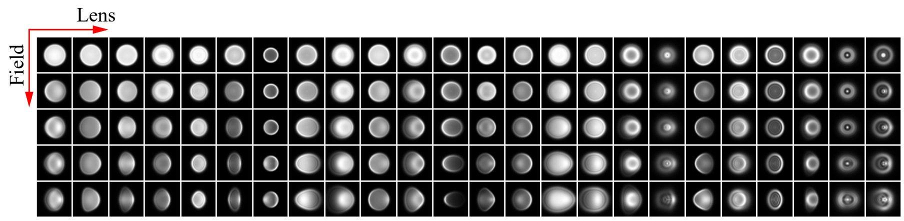


Fig. 6: PSF examples of various lens designs (column) and field positions (row), computed by the Debye CZT.

5.5 Ablation Study

We conduct an ablation study to evaluate the contributions of three components in our synthesis pipeline: (1) realistic lens PSFs, (2) the imaging pipeline, and (3) depth-varying scenes. For each experiment, we maintain all other components, including consistent patch sampling and augmentations, as much as possible.

Realistic Lens PSF. We first investigate the impact of the lens PSFs by replacing them with isotropic Gaussian blur. For fair comparison, we match the blur magnitude of the Gaussian kernels to the original PSFs by deriving their standard deviation σ from the circle of confusion (CoC). Following prior works [22, 33], we set σ = CoC radius/2. As shown in Tab. 3, replacing lens PSFs with Gaussian kernels leads to a significant degradation in performance. Qualitatively, Fig. 7 reveals that substantial blur remains in the restored images, likely because the model trained with Gaussian blur fails to generalize to complex real blur.

Realistic Imaging Pipeline. Next, we analyze the impact of the realistic imaging pipeline. We remove the ISP and noise synthesis in the RAW space. In this setup, we directly sample a sharp RGB patch, synthesize blur in RGB space, and obtain the blurred RGB patch without ISP. The results in Tab. 3 demonstrate that the absence of a realistic imaging pipeline results in a notable performance drop, highlighting its importance.

Depth-Varying Scenes. Finally, we examine the impact of depth-varying scenes. We simplify each patch into a single layer, allowing for blur synthesis via a uniform convolution. As shown in Tab. 3, this leads to only a marginal decrease in overall metrics, while the model remains vulnerable to defocus discontinuities. Specifically, as shown in Fig. 7, the model exhibits concentrated artifacts near edges with sharp defocus transitions. This suggests that modeling depth variance is crucial to handle discontinuous defocus changes near object boundaries.

Table 3: Ablation results on the RealDOF dataset showing the impact of each component. The best, second best, and third best results are highlighted.

Components	PSNR↑	SSIM↑	LPIPS↓	NIQE↓	MUSIQ↑	TOPIQ↑
w/o Lens	22.34	0.667	0.521	7.733	25.162	0.238
w/o ISP	23.49	0.746	0.349	5.191	32.045	0.285
w/o Depth	24.54	0.762	0.317	5.292	34.999	0.300
Full	24.74	0.764	0.303	5.029	35.787	0.303

Input w/o Lens w/o ISP w/o Depth Full Ground truth

Fig. 7: Qualitative ablation results illustrating the impact of each component.

6 Conclusion

In this paper, we proposed a framework for synthesizing realistic defocus deblurring datasets for compound lenses. It integrates efficient wave-optics PSF computation, depth-aware defocus rendering with occlusion handling, and ISP-aware blur synthesis in the radiometrically linear space. This unified framework enables scalable, photorealistic dataset generation with improved computational efficiency over existing approaches. We also presented the CLDefocus dataset and showed that it is competitive with the real-captured DPDD for training.

Limitations and Future Work. Our pipeline still has limitations. The Debye CZT may yield less accurate PSFs for severe off-axis fields due to the Debye approximation and failure to capture complex pupil shapes. Moreover, imprecise depth maps might produce implausible synthesis results, and the simple ISP model might not fully capture the diverse ISPs of real-world cameras. The pipeline offers flexibility that remains largely unexplored. Its controllable parameters enable task-specific dataset design. For example, blur levels can be adjusted to construct datasets for curriculum learning, which is difficult to achieve with captured data. We hope future work will further leverage this flexibility for broader tasks.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) (RS-2026-25492695; Basic Science Research Program, RS-2022-NR070870) and by the Institute of Information & Communications Technology Planning & Evaluation (IITP) (RS-2019-II191906, Artificial Intelligence Graduate School Program (POSTECH); IITP-2026-RS-2024-00437866, ITRC Program), funded by the Korea government (MSIT).

References

1. Abuolaim, A., Brown, M.S.: Defocus deblurring using dual-pixel data. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (eds.) Computer Vision – ECCV 2020. Lecture Notes in Computer Science, vol. 12355, pp. 111–126. Springer (2020). https://doi.org/10.1007/978-3-030-58607-2_7
2. Bando, Y., Nishita, T.: Towards digital refocusing from a single photograph. In: 15th Pacific Conference on Computer Graphics and Applications (PG). pp. 363–372. IEEE (2007). https://doi.org/10.1109/PG.2007.22
3. Bluestein, L.: A linear filtering approach to the computation of discrete Fourier transform. IEEE Transactions on Audio and Electroacoustics **18**(4), 451–455 (1970). https://doi.org/10.1109/TAU.1970.1162132
4. Bochkovskii, A., Delaunoy, A., Germain, H., Santos, M., Zhou, Y., Richter, S.R., Koltun, V.: Depth Pro: Sharp monocular metric depth in less than a second. In: International Conference on Learning Representations (ICLR) (2025)
5. Bradbury, J., Frostig, R., Hawkins, P., Johnson, M.J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S., Zhang, Q.: JAX: Composable transformations of Python+Numpy programs. https://github.com/jax-ml/jax (2018)
6. Cai, Y., Wang, Z., Liang, Y., Ren, F., Yao, B., Lei, M., Yan, S.: Direct calculation of tightly focused field in an arbitrary plane. Optics Communications **450**, 329–334 (2019). https://doi.org/10.1016/j.optcom.2019.06.022
7. Chen, C., Mo, J., Hou, J., Wu, H., Liao, L., Sun, W., Yan, Q., Lin, W.: TOPIQ: A top-down approach from semantics to distortions for image quality assessment. IEEE Transactions on Image Processing **33**, 2404–2418 (2024). https://doi.org/10.1109/TIP.2024.3378466
8. Chen, L., Chu, X., Zhang, X., Sun, J.: Simple baselines for image restoration. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022. Lecture Notes in Computer Science, vol. 13667, pp. 17–33. Springer (2022). https://doi.org/10.1007/978-3-031-20071-7_2
9. Chen, L., Li, Y., Dai, J., Gu, J., Xue, T.: A physics-informed blur learning framework for imaging systems. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10913–10922 (2025). https://doi.org/10.1109/CVPR52734.2025.01019
10. Chen, S., Feng, H., Pan, D., Xu, Z., Li, Q., Chen, Y.: Optical aberrations correction in postprocessing using imaging simulation. ACM Transactions on Graphics **40**(5), 192:1–192:15 (2021). https://doi.org/10.1145/3474088
11. D'Andrès, L., Salvador, J., Kochale, A., Süsstrunk, S.: Non-parametric blur map regression for depth of field extension. IEEE Transactions on Image Processing **25**(4), 1660–1673 (2016). https://doi.org/10.1109/TIP.2016.2526907
12. Elmacioglu, Y.G., Conlin, R., Dudt, D.W., Panici, D., Kolemen, E.: ZERNIPAX: A fast and accurate Zernike polynomial calculator in Python. Applied Mathematics and Computation **505**, 129534 (2025). https://doi.org/10.1016/j.amc.2025.129534
13. Goodman, J.W.: Introduction to Fourier optics. W. H. Freeman, 4th edn. (2017)
14. Hasinoff, S.W., Kutulakos, K.N.: A layer-based restoration framework for variable-aperture photography. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 1–8. IEEE (2007). https://doi.org/10.1109/ICCV.2007.4408898

15. Ho, C.J., Belhe, Y., Rotenberg, S., Ramamoorthi, R., Li, T.M., Antipa, N.: A differentiable wave optics model for end-to-end computational imaging system optimization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 28042–28051 (2025). https://doi.org/10.1109/ICCV51701.2025.02603
16. Hu, Y., Wang, Z., Wang, X., Ji, S., Zhang, C., Li, J., Zhu, W., Wu, D., Chu, J.: Efficient full-path optical calculation of scalar and vector diffraction using the Bluestein method. Light: Science & Applications **9**(1), 119 (2020). https://doi.org/10.1038/s41377-020-00362-z
17. Ke, J., Wang, Q., Wang, Y., Milanfar, P., Yang, F.: MUSIQ: Multi-scale image quality transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 5128–5137 (2021). https://doi.org/10.1109/ICCV48922.2021.00510
18. Kong, T., Sun, F., Liu, H., Jiang, Y., Li, L., Shi, J.: FoveaBox: Beyound anchor-based object detection. IEEE Transactions on Image Processing **29**, 7389–7398 (2020). https://doi.org/10.1109/TIP.2020.3002345
19. Kraus, M., Strengert, M.: Depth-of-field rendering by pyramidal image processing. Computer Graphics Forum **26**(3), 645–654 (2007). https://doi.org/10.1111/j.1467-8659.2007.01088.x
20. Krishnan, D., Fergus, R.: Fast image deconvolution using hyper-laplacian priors. In: Advances in Neural Information Processing Systems (NeurIPS). pp. 1033–1041 (2009)
21. Kupinski, M.K., Bradley, C.L., Diner, D.J., Xu, F., Chipman, R.A.: Angle of linear polarization images of outdoor scenes. Optical Engineering **58**(8), 082419 (2019). https://doi.org/10.1117/1.OE.58.8.082419
22. Lee, J., Lee, S., Cho, S., Lee, S.: Deep defocus map estimation using domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 12222–12230 (2019). https://doi.org/10.1109/CVPR.2019.01250
23. Lee, J., Son, H., Rim, J., Cho, S., Lee, S.: Iterative filter adaptive network for single image defocus deblurring. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2034–2042 (2021). https://doi.org/10.1109/CVPR46437.2021.00207
24. Leutenegger, M., Rao, R., Leitgeb, R.A., Lasser, T.: Fast focus field calculations. Optics Express **14**(23), 11277–11291 (2006). https://doi.org/10.1364/oe.14.011277
25. Li, Y., Ren, D., Shu, X., Zuo, W.: Learning single image defocus deblurring with misaligned training pairs. In: Proceedings of the AAAI Conference on Artificial Intelligence (AAAI). vol. 37, pp. 1495–1503 (2023). https://doi.org/10.1609/aaai.v37i2.25235
26. Luo, J., Nie, Y., Ren, W., Cao, X., Yang, M.H.: Correcting optical aberration via depth-aware point spread functions. IEEE Transactions on Pattern Analysis and Machine Intelligence **46**(8), 5541–5555 (2024). https://doi.org/10.1109/TPAMI.2024.3370794
27. Ma, H., Liu, S., Liao, Q., Zhang, J., Xue, J.H.: Defocus image deblurring network with defocus map estimation as auxiliary task. IEEE Transactions on Image Processing **31**, 216–226 (2022). https://doi.org/10.1109/TIP.2021.3127850
28. Marathay, A.S., McCalmont, J.F.: On the usual approximation used in the Rayleigh–Sommerfeld diffraction theory. Journal of the Optical Society of America A **21**(4), 510–516 (2004). https://doi.org/10.1364/josaa.21.000510

29. Miora, R.H.D., Rohwer, E., Kielhorn, M., Sheppard, C., Bosman, G., Heintzmann, R.: Calculating point spread functions: methods, pitfalls, and solutions. Optics Express **32**(16), 27278–27302 (2024). https://doi.org/10.1364/oe.523532
30. Mittal, A., Soundararajan, R., Bovik, A.C.: Making a "completely blind" image quality analyzer. IEEE Signal Processing Letters **20**(3), 209–212 (2013). https://doi.org/10.1109/LSP.2012.2227726
31. Müller, P., Braun, A., Keuper, M.: Classification robustness to common optical aberrations. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW). pp. 3634–3645 (2023). https://doi.org/10.1109/ICCVW60793.2023.00391
32. Müller, P., Braun, A., Keuper, M.: Examining the impact of optical aberrations to image classification and object detection models. IEEE Transactions on Pattern Analysis and Machine Intelligence **48**(3), 2139–2153 (2026). https://doi.org/10.1109/TPAMI.2025.3622234
33. Nazir, S., Vaquero, L., Mucientes, M., Brea, V.M., Coltuc, D.: Depth estimation and image restoration by deep learning from defocused images. IEEE Transactions on Computational Imaging **9**, 607–619 (2023). https://doi.org/10.1109/TCI.2023.3288335
34. Niu, K., Tian, C.: Zernike polynomials and their applications. Journal of Optics **24**(12), 123001 (2022). https://doi.org/10.1088/2040-8986/ac9e08
35. Potmesil, M., Chakravarty, I.: Synthetic image generation with a lens and aperture camera model. ACM Transactions on Graphics **1**(2), 85–108 (1982). https://doi.org/10.1145/357299.357300
36. Quan, Y., Wu, Z., Ji, H.: Neumann network with recursive kernels for single image defocus deblurring. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5754–5763 (2023). https://doi.org/10.1109/CVPR52729.2023.00557
37. Quan, Y., Yao, X., Ji, H.: Single image defocus deblurring via implicit neural inverse kernels. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 12566–12576 (2023). https://doi.org/10.1109/ICCV51070.2023.01158
38. Rabiner, L.R., Schafer, R.W., Rader, C.M.: The chirp Z-transform algorithm. IEEE Transactions on Audio and Electroacoustics **17**(2), 86–92 (1969). https://doi.org/10.1109/TAU.1969.1162034
39. Reiley, D.J.: Lens designs. https://www.lens-designs.com/, accessed: 2026-02-10
40. Rim, J., Kim, G., Kim, J., Lee, J., Lee, S., Cho, S.: Realistic blur synthesis for learning image deblurring. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022. Lecture Notes in Computer Science, vol. 13667, pp. 487–503. Springer (2022). https://doi.org/10.1007/978-3-031-20071-7_29
41. Ruan, L., Chen, B., Li, J., Lam, M.L.: AIFNet: All-in-focus image restoration network using a light field-based dataset. IEEE Transactions on Computational Imaging **7**, 675–688 (2021). https://doi.org/10.1109/TCI.2021.3092891
42. Ruan, L., Chen, B., Li, J., Lam, M.L.: 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). pp. 16283–16292 (2022). https://doi.org/10.1109/CVPR52688.2022.01582
43. Saad, E., Hirakawa, K.: Defocus blur-invariant scale-space feature extractions. IEEE Transactions on Image Processing **25**(7), 3141–3156 (2016). https://doi.org/10.1109/TIP.2016.2555702

44. Schmidt, J.D.: Numerical simulation of optical wave propagation with examples in MATLAB. SPIE Press (2010). https://doi.org/10.1117/3.866274
45. Shi, J., Xu, L., Jia, J.: Discriminative blur detection features. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2965–2972 (2014). https://doi.org/10.1109/CVPR.2014.379
46. Shi, J., Xu, L., Jia, J.: Just noticeable defocus blur detection and estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 657–665 (2015). https://doi.org/10.1109/CVPR.2015.7298665
47. Son, H., Lee, J., Cho, S., Lee, S.: Single image defocus deblurring using kernel-sharing parallel atrous convolutions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 2622–2630 (2021). https://doi.org/10.1109/ICCV48922.2021.00264
48. Wang, L., Li, D., Zhu, Y., Tian, L., Shan, Y.: Dual super-resolution learning for semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3773–3782 (2020). https://doi.org/10.1109/CVPR42600.2020.00383
49. Yang, X., Nguyen, C., Wang, W., Kang, K., Heidrich, W., Li, X.: Efficient depth- and spatially-varying image simulation for defocus deblur. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW). pp. 3047–3057 (2025). https://doi.org/10.1109/ICCVW69036.2025.00318
50. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H.: Restormer: Efficient transformer for high-resolution image restoration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5718–5729 (2022). https://doi.org/10.1109/CVPR52688.2022.00564
51. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 586–595 (2018). https://doi.org/10.1109/CVPR.2018.00068
52. Zhuo, S., Sim, T.: Defocus map estimation from a single image. Pattern Recognition **44**(9), 1852–1858 (2011). https://doi.org/10.1016/j.patcog.2011.03.009