AnyBokeh: Physics-Guided Any-to-Any Bokeh Editing with Optical Fingerprint Transfer

Xinyu Hou Xiaoming Li Zongsheng Yue Chen Change Loy
S-Lab, College of Computing & Data Science
Nanyang Technological University
Singapore, 639798
{xinyu.hou, ccloy}@ntu.edu.sg, {csxmli, zsyzam}@gmail.com

Abstract

Depth-of-field control is a fundamental tool in photography, yet post-capture bokeh editing from a single image remains challenging. A practical editor should handle images captured under arbitrary focus and aperture settings. Existing methods typically assume an all-in-focus input, or first recover an all-in-focus image before rendering new bokeh. Such pipelines pass only the reconstructed sharp image to the renderer, without explicitly preserving the source blur state, and may propagate reconstruction artifacts into the final edit. We introduce **AnyBokeh**, a physics-guided framework for **any-to-any bokeh editing** under the ideal thin-lens model. Instead of treating source blur merely as a degradation to be removed, AnyBokeh estimates the **source blur state** with a signed circle-of-confusion map and a disparity map. By modeling the linear relation between the signed circle of confusion and the disparity difference, a source-specific optical fingerprint is extracted and transferred to the desired focus and aperture setting. A generative editor conditioned on both source and target circle-of-confusion maps then performs relative blur synthesis, enabling spatially adaptive deblurring, preservation, and defocus rendering. To support physically supervised learning, we further construct a high-fidelity synthetic dataset with accurate depth, focus distance, and full EXIF metadata. Experiments on real-world benchmarks show that AnyBokeh achieves faithful and controllable results on any-to-any bokeh editing, all-in-focus-to-bokeh rendering, and defocus deblurring, while avoiding all-in-focus reconstruction and test-time bokeh-level calibration commonly required by existing approaches. The code and dataset are available at `https://github.com/itsmag11/AnyBokeh`.

1 Introduction

Depth-of-field (DoF) control is an essential aesthetic tool in photography, guiding visual attention and conveying scene structure through selective focus and optical defocus, commonly perceived as bokeh. The strength and spatial distribution of bokeh are governed by the image formation process: scene points away from the focal plane are projected as circles of confusion (CoC) whose size depends on aperture, focus distance, focal length, and scene depth. In practice, a large physical aperture, corresponding to a small f-number such as $f/2.0$, produces a shallow DoF and strong blur in out-of-focus regions, while a small physical aperture such as $f/22.0$ increases the DoF and makes a wider depth range appear sharp. Although these optical effects can be controlled during capture, post-capture DoF manipulation from a single in-the-wild photograph remains highly challenging.

The difficulty mainly comes from two unknowns. First, the input may already contain spatially varying defocus blur: some regions are sharp, while others are mildly or severely out of focus. A practical editor must understand this source blur state to decide which regions should be sharpened, preserved, or further defocused. Second, the camera parameters that determine the absolute blur scale,

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

such as focus distance, aperture, focal length, sensor size, and image resolution, are often incomplete or unavailable for in-the-wild images. Together, these two unknowns make any-to-any bokeh editing fundamentally different from standard bokeh rendering: the editor must infer the current blur state of the source image while also determining the target blur scale under the desired optical setting.

Existing methods address these two ambiguities only partially. Many bokeh rendering methods assume an all-in-focus (AIF) input [17, 33, 27, 26, 46, 32], where the source blur state is trivial and only the target bokeh needs to be synthesized. This assumption simplifies the problem, but restricts these methods to already sharp inputs. Recent refocusing pipelines [35] relax the input constraint by first reconstructing an AIF image from an arbitrary input and then rendering new bokeh. We refer to this forced AIF reconstruction as an *AIF bottleneck*: the input is projected into a sharp intermediate representation. Although recovering this image may implicitly reason about the source blur, only the sharp result is passed to the renderer, so source-specific optical cues are not available for subsequent target bokeh synthesis. Moreover, severe defocus deblurring is inherently ambiguous and can introduce unnecessary hallucinated details or structural errors that propagate into the final edit, even when the target only requires preserving or adding blur. In addition, because complete optical metadata is usually unavailable, existing pipelines often require *bokeh-level calibration* [26, 46, 35], i.e., a per-image, ground-truth-guided search for a renderer-specific blur scale.

To address these limitations, we propose **AnyBokeh**, a physics-guided framework for any-to-any bokeh editing, i.e., changing focus and aperture from an arbitrary source setting to an arbitrary target setting under the ideal thin-lens model. Instead of treating source blur merely as a degradation to be removed, AnyBokeh explicitly estimates the source blur state as a signed source CoC map, together with a disparity map (inverse depth). By modeling the linear relation between signed CoC diameter and disparity difference, we estimate a source-specific CoC–disparity slope, which absorbs the global optical and scaling factors that map disparity differences to image-space defocus. We call this slope an *optical fingerprint*. This quantity is physically equivalent to the bokeh-level parameter K used in prior rendering methods [26, 46, 35], as both control how disparity differences are converted into CoC magnitude. In our formulation, we denote it by κ_{src} to emphasize that this CoC scale is estimated directly from the observed source blur state and disparity, rather than specified externally. We then transfer this optical fingerprint to the desired focus and aperture setting to obtain the target CoC map, avoiding AIF reconstruction and test-time bokeh-level calibration.

AnyBokeh synthesizes the target image with a generative editor conditioned on the source image, source CoC, and target CoC. We refer to this design as *dual-CoC conditioning*: both the current and desired blur states are provided to the model, turning any-to-any bokeh editing into relative blur synthesis. This allows the editor to explicitly model the transition from source blur to target blur and perform spatially adaptive bokeh synthesis. To train the CoC and disparity estimator under accurate physical supervision, we further construct **UnrealBokeh**, a high-fidelity synthetic dataset rendered with ground-truth depth, focus distance, aperture, and focal length. Our contributions are:

- We propose a CoC-mediated optical fingerprint transfer framework for any-to-any bokeh editing. By estimating source CoC and disparity, our method extracts a source-specific CoC–disparity slope κ_{src} and analytically transfers it to the target focus and aperture setting, avoiding AIF reconstruction and test-time bokeh-level calibration.
- We introduce dual-CoC conditioning for relative blur synthesis. By conditioning the editor on both source and target CoC, our model explicitly observes both the input and desired output blur states, enabling spatially selective deblurring, preservation, and defocus synthesis.
- We build a high-fidelity synthetic dataset for physically supervised CoC learning. Using Unreal Engine, we render diverse scenes with ground-truth depth, focus distance, aperture, and focal length across multiple optical configurations for robust CoC and disparity estimation.

2 Related Work

Bokeh Rendering. Classical depth-of-field rendering relies on physically based simulation, which is realistic but expensive and requires complete 3D geometry [29, 13]. Image-space post-filtering [18, 21, 36, 3, 25] improves efficiency but struggles with color bleeding and partial occlusion at depth discontinuities [6, 33], which physical lens models and multiview or light-field synthesis [12, 39, 20, 41] only partially mitigate. To better handle these geometric artifacts, recent research has shifted toward layered and neural rendering. Early deep models and benchmarks study natural

camera bokeh rendering from AIF images [17], while later methods combine depth estimation, layered scene representations, differentiable rendering, or neural priors to improve photorealism and boundary handling. MPIB [27] uses multiplane images and background inpainting for occlusions, BokehMe [26] and BokehMe++ [28] combine classical filtering with neural refinement, Dr. Bokeh [33] introduces a differentiable occlusion-aware renderer, and Bokehlicious [32] learns aperture-aware attention without explicit depth. Diffusion priors have recently been used to inject bokeh conditions into text-to-image synthesis [11, 42] and to perform one-step neural lens blur [46]. Despite significant advances in synthesis quality and efficiency, these rendering methods all require AIF inputs. Moreover, they typically control blur strength through a renderer-specific bokeh-level parameter, which often requires per-image tuning when camera metadata is incomplete. To support arbitrary inputs, DiffCamera [38] trains a DiT [24] on simulated depth-of-field pairs, using a physics-inspired stacking constraint and depth dropout to control focus points and bokeh levels. However, its defocus maps for bokeh-to-clear rendering are derived from clear images, which may cause information leakage. Genfocus [35] cascades a DeblurNet that recovers an AIF image and a BokehNet, trained with semi-supervised learning on synthetic and real data, that renders the target bokeh with controllable focus and blur intensity. Since only the recovered AIF image is passed to BokehNet, the source optical state is not explicitly retained for rendering, and full-image deblurring may introduce unnecessary hallucination. In contrast, AnyBokeh edits the defocused input directly by modeling both source and target blur states, without an AIF intermediate, so that regions that only need preservation or further defocus are never forced through deblurring.

Defocus Deblurring. Defocus deblurring recovers an AIF image from spatially varying blur. Unlike uniform motion blur, defocus blur is tightly coupled with scene depth and aperture, and its magnitude can change abruptly across depth discontinuities and occlusion boundaries. Early methods perform spatially varying deconvolution with estimated defocus maps or kernels [2, 47], but are sensitive to inaccurate blur estimation and often introduce artifacts around complex depth transitions. Learning-based methods exploit dual-pixel [1, 23] or quad-pixel [7] sensors for aperture-dependent sub-view cues, or model defocus with adaptive filters [19], disparity-aware kernels [40], or implicit inverse kernels [30, 34]. DRBNet [31] learns deblurring from light-field-generated bokeh and real defocus images, while Restormer [43] is an efficient Transformer for high-resolution restoration. As regression models, both may struggle to recover high-frequency structures under severe defocus, producing structural distortions or overly smooth textures in heavily blurred regions. These works show that effective deblurring requires not only generic restoration capacity, but also an understanding of the relation between blur magnitude and scene geometry, which AnyBokeh models explicitly through CoC and disparity.

3 Preliminaries

Circle-of-Confusion. Under the thin-lens model with a circular aperture, a scene point away from the focal plane is imaged as a defocus disk, commonly referred to as the circle of confusion (CoC). Let N denote the f-number, f the focal length, S_1 the focus distance, and S_2 the scene-point distance. All distances are in millimeters, and N is dimensionless. The signed physical CoC diameter is

$$CoC^{mm} = \frac{f^2}{N(S_1 - f)} \frac{(S_2 - S_1)}{S_2} . \tag{1}$$

The sign indicates whether the scene point lies in front of or behind the focal plane, while the magnitude gives the blur diameter. Throughout the paper, we use CoC to denote signed CoC diameter unless otherwise specified.

To express the CoC in image pixels (px), the physical blur diameter is converted according to the sensor size and image resolution:

$$CoC^{px} = CoC^{mm} \cdot \frac{W_{res}}{W_{sensor}} , \tag{2}$$

where W_{res} is the image width in px and W_{sensor} is the physical sensor width in mm. Note that any image resizing changes the CoC magnitude in pixel units. In our Unreal Engine [9] synthetic dataset (Sec. 4.3), the per-pixel scene depth, focus distance, focal length, f-number, sensor size, and image resolution are all available. We therefore compute dense ground-truth signed CoC maps paired with each rendered RGB image, providing direct supervision for learning CoC estimation.

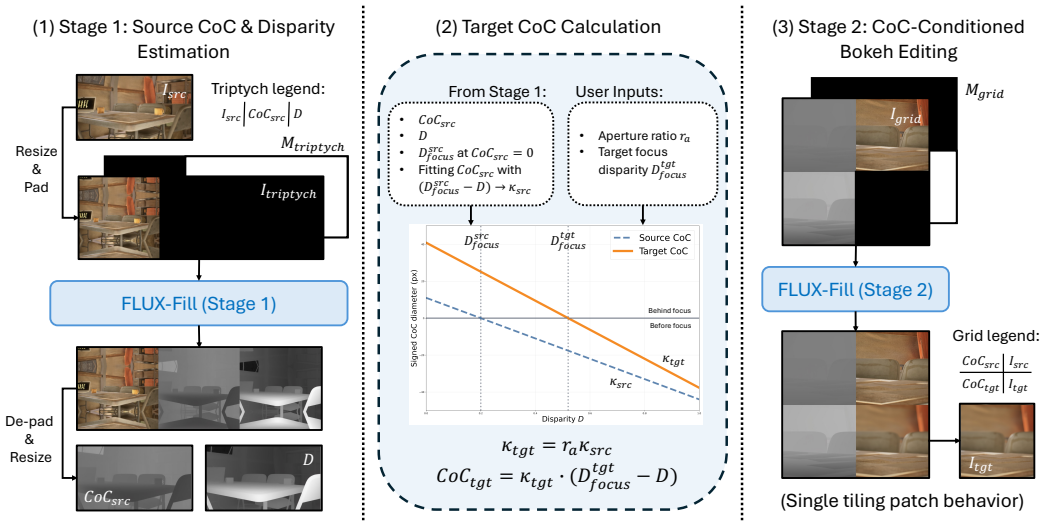


Figure 1: Overview of the AnyBokeh framework. (1) Stage 1 predicts the signed source CoC map CoC_{src} and disparity map D from the source image to recover the source blur state and geometry. (2) The target CoC is analytically computed by transferring the source optical fingerprint κ_{src} to the desired focus and aperture setting. (3) Stage 2 performs CoC-conditioned bokeh editing using the source image, CoC_{src}, and CoC_{tgt} to synthesize the final target image.

Optical Fingerprint. Eq. 1 shows that the CoC is determined by both scene geometry and camera optics. In typical non-macro photography, the focus distance is much larger than the focal length, i.e., $S_1 \gg f$. Thus, $S_1 - f \approx S_1$, and the signed CoC in physical units can be approximated as

$$CoC^{mm} \approx \frac{f^2}{N \cdot S_1} \frac{(S_2 - S_1)}{S_2} = \frac{f^2}{N} \cdot \left(\frac{1}{S_1} - \frac{1}{S_2}\right). \tag{3}$$

Let $D = 1/S_2$ denote the disparity of a scene point and $D_{focus} = 1/S_1$ denote the focus disparity. Following the bokeh rendering literature [26, 46, 35], we use "disparity" to represent inverse depth rather than the lateral displacement between stereo views. Substituting this into Eq. 2 gives the pixel-space CoC

$$CoC^{px} = \underbrace{\left(\frac{f^2 \cdot W_{res}}{N \cdot W_{sensor}}\right)}_{\kappa} \cdot (D_{focus} - D). \tag{4}$$

This formulation reveals a linear relationship between signed CoC and disparity difference, where the slope κ is shared by all pixels of the same source image. Therefore, even when EXIF metadata is unavailable or incomplete, the slope of this relationship can be estimated from the predicted source CoC and disparity. It absorbs the global factors that determine the CoC scale, including aperture, focal length, sensor size, image resolution, and the scale convention of the estimated disparity. We refer to this source-specific slope as the *optical fingerprint*, since it characterizes how the camera configuration and depth representation convert disparity differences into image-space defocus.

4 Methodology

4.1 Stage 1: CoC and Disparity Estimation

Stage 1 of our method estimates the source blur state and geometry from a single input image. Given the source image I_{src}, it jointly predicts two dense maps: the signed source CoC map CoC_{src} and the disparity map D. The source CoC describes the defocus already present in the input image, while disparity provides the geometric coordinate needed to transfer the optical fingerprint to a new focus setting. Predicting these two quantities together encourages the model to reason about defocus and geometry in a coupled manner: CoC_{src} provides direct evidence of the observed blur state, while D explains the scene geometry underlying that blur.

Estimating CoC_{src} and D from defocused inputs is inherently ambiguous, especially around depth discontinuities and severely blurred structures where high-frequency evidence has been weakened. We therefore rely on a generative prediction model with a strong image prior, which can infer plausible scene boundaries and coherent geometry from degraded visual cues. This prior is particularly useful for producing sharp and spatially consistent dense maps, rather than overly smooth estimates that would lead to inaccurate CoC–disparity slopes. The CoC map is normalized as detailed in Sec. A.1.

We predict CoC_{src} and D on the full resized image rather than using tiled inference. This is important because disparity is defined up to a global scale convention, and independently predicting different tiles may lead to inconsistent disparity scales or discontinuities across tile boundaries. Such inconsistencies would directly affect the estimated CoC–disparity slope and therefore the optical fingerprint. Instead, Stage 1 predicts globally consistent low-resolution maps from the complete image context, which are then de-padded and upsampled to the original resolution. This formulation is agnostic to the unknown camera metadata or prior image transformations of an in-the-wild photo. Stage 1 does not require knowing how the image was captured or resized before being observed. It simply estimates the signed CoC diameter visible on the current input canvas. As a result, the predicted CoC_{src} and D provide a consistent basis for estimating the source optical fingerprint. The framework of Stage 1 is shown in Fig. 1 (1).

Target CoC Calculation. Given the source CoC and disparity from Stage 1, our goal is to construct a target CoC map that specifies the desired blur state for Stage 2. We primarily consider the relative editing setting, where the source image provides observable defocus cues and the user specifies the desired change in focus and aperture. This setting is suitable for in-the-wild bokeh editing, since complete EXIF metadata is often unavailable, but the source blur pattern still carries information about the underlying optical scale. An optional absolute editing mode is described in Sec. A.2.

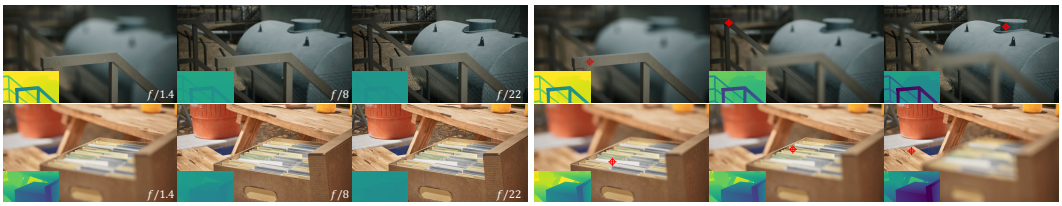
As derived in Eq. 4, the signed CoC is approximately linear with respect to the disparity difference $(D^{src}_{focus} - D)$. The source focus disparity D^{src}_{focus} can be directly obtained as the disparity at $CoC_{src} = 0$. For a given source image, the slope κ_{src} captures the source-specific CoC scale and serves as the optical fingerprint that both source and target images share. Since both CoC_{src} and D are generative predictions, a naive least-squares fit is sensitive to noisy pixels. We therefore estimate κ_{src} with a robust extreme-median (EM) estimator: each pixel i gives a local slope $\kappa_i = CoC_{src,i}/(D^{src}_{focus} - D_i)$, and κ_{src} is the median of κ_i over the 5% of pixels with the largest $|CoC_{src}|$. Larger $|CoC_{src}|$ gives a more reliable local slope, while pixels closer to $CoC = 0$ are more likely to be dominated by prediction noise. The median further suppresses outliers around occlusion boundaries. On held-out UnrealBokeh scenes, EM reduces the κ_{src} error compared with least squares and RANSAC [10], and the resulting target CoC remains stable across Stage 1 sampling seeds (Sec. A.3). Once κ_{src} is estimated from the predicted CoC_{src} and D, we transfer it to the target setting by scaling it according to the desired aperture change. Let r_a denote the aperture ratio from the source to the target setting, e.g., $r_a = N_{src}/N_{tgt}$ when the focal length and sensor scale are fixed. The target CoC–disparity slope and target CoC are computed as

$$\kappa_{tgt} = r_a \kappa_{src}\,, \qquad CoC_{tgt} = \kappa_{tgt}\left(D^{tgt}_{focus} - D\right)\,, \tag{5}$$

where D^{tgt}_{focus} is the user-specified target focus disparity. An illustration is shown in Fig. 1 (2). This formulation transfers the optical scale observed in the source image to the desired focus and aperture setting without requiring complete camera metadata. Unlike previous methods [26, 46, 35], the target CoC is obtained directly from the source optical fingerprint and user control, avoiding test-time bokeh-level calibration. Moreover, since the target CoC is derived by scaling the source optical fingerprint, the predicted blur strength remains consistent with the defocus scale observed in the input image, rather than being determined by an independently searched parameter.

4.2 Stage 2: CoC-Conditioned Bokeh Editing

Given the source image I_{src}, CoC_{src}, and CoC_{tgt}, Stage 2 learns the mapping $(I_{src}, CoC_{src}, CoC_{tgt}) \rightarrow I_{tgt}$. Unlike conventional AIF-to-bokeh rendering [17, 33, 27, 26, 46, 32], our setting handles inputs that may already contain defocus blur. Therefore, the model needs to determine not only where to add blur, but also where to preserve existing content or recover sharper structures. To this end, we use *dual-CoC conditioning*: both CoC_{src} and CoC_{tgt} are provided to the editor as explicit blur-state guidance. The source CoC describes the blur already present in the input image, while the target CoC specifies the desired blur distribution after editing. Together, they define



(1) Change of aperture. Aperture values are marked in right bottom corner. (2) Change of focus distance. Focus points are marked by crosshair markers.

Figure 2: Samples from UnrealBokeh. For a fixed scene, camera viewpoint, and focal length, we render images under different aperture and focus settings. Dense ground-truth CoC maps are computed from scene depth and camera parameters, with a normalized value of 0.5 corresponding to CoC$= 0$.

Table 1: Dataset comparison. UnrealBokeh provides complete optical supervision and dense geometry, enabling ground-truth CoC computation for Stage 1 training. *The actual downloadable resolution of RealBokeh differs from the 6000×4000 stated in the paper [32].

Dataset	**UnrealBokeh**	RealBokeh [32]	EBB! [17]
# Images	11,477	27,451	9,774
# Editing Pairs	382,888	224,750	9,774
Resolution	2016×1120	2000×1500*	$\approx 1550\times1024$
Aperture Range	$f/22$–$f/1.4$	$f/20$–$f/2$	$f/16$,$f/1.8$
Multiple Focus Distances	✓	✗	✗
Complete EXIF Metadata	✓	✗	✗
Dense GT Depth	✓	✗	✗
Computable Dense GT CoC	✓	✗	✗

a relative blur transition, allowing the editor to reason about whether each region should be deblurred, preserved, or further defocused. Both CoC maps are normalized as detailed in Sec. A.1. This design is important for any-to-any editing because the target CoC alone does not reveal the source blur state. If CoC_{src} is omitted, the model must infer the existing defocus solely from image appearance, which is ambiguous and can lead to inaccurate bokeh synthesis (see Sec. 5.4). Combined with the target CoC calculation in Sec. 4.1, Stage 2 completes the any-to-any editing pipeline by turning the transferred optical fingerprint into the final refocused image. The framework is shown in Fig. 1 (3).

4.3 UnrealBokeh Synthetic Dataset

Training Stage 1 requires dense supervision for both source CoC and disparity estimation. Accurate CoC supervision depends not only on scene depth, but also on optical parameters such as focus distance, focal length, aperture, sensor size, and image resolution. Such annotations are difficult to obtain from real photographs, especially for in-the-wild images where complete metadata, depth, and focus annotations are typically unavailable. Existing bokeh datasets are not sufficient for this purpose. Real datasets such as EBB! [17] and RealBokeh [32] provide paired images captured under different camera settings, but do not include the dense geometry and complete optical metadata needed to compute ground-truth CoC maps. Synthetic datasets generated by classical ray tracing [26, 46] provide scalable training data, but often rely on simplified layer-based geometry that cannot fully represent complex scene depth.

To enable physically supervised CoC learning, we construct **UnrealBokeh**, a high-fidelity synthetic dataset rendered in Unreal Engine [9]. For each image, we record RGB, dense depth, focus distance, aperture, focal length, sensor size, and image resolution, allowing us to compute dense signed CoC maps from the thin-lens model. Sample images with corresponding CoC maps from UnrealBokeh are shown in Fig. 2. We render diverse indoor and outdoor scenes under controlled optical variations and form source–target editing pairs by varying focus distance and aperture while keeping scene geometry and camera viewpoint fixed. The resulting dataset contains 11,477 images and 382,888 editing pairs. Comparisons with existing datasets are listed in Tab. 1. We split the data by environment–camera group to avoid scene and viewpoint leakage across training and validation. UnrealBokeh provides physically grounded supervision for Stage 1 and aligned source–target pairs for training Stage 2. Additional dataset construction details are provided in Appendix A.4.

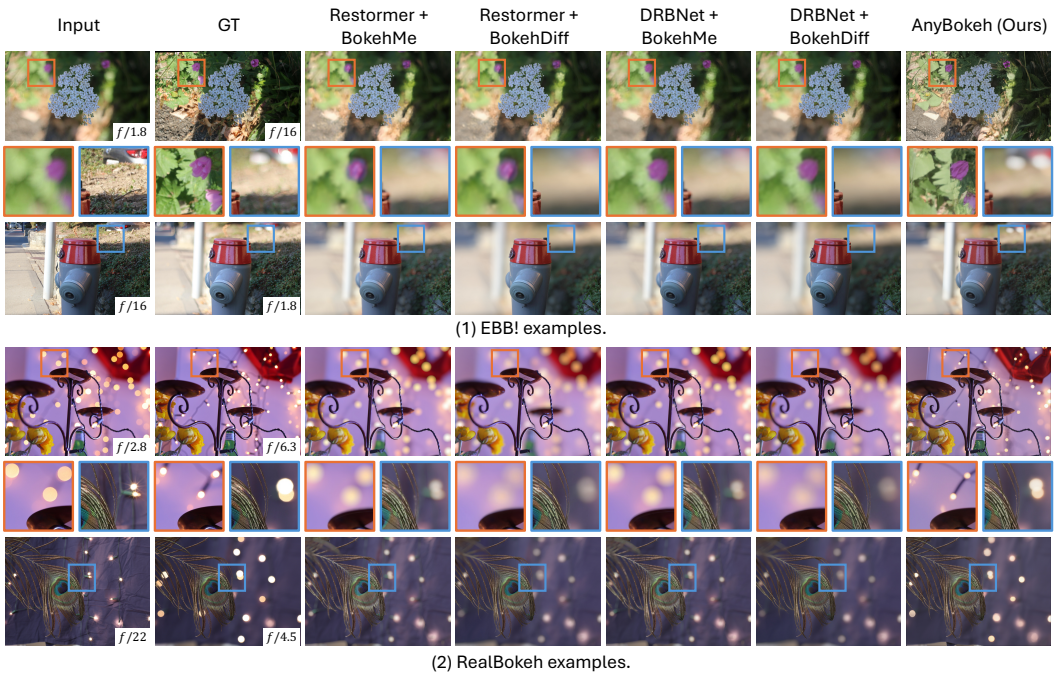


Figure 3: Qualitative comparison on any-to-any bokeh editing. Aperture values are shown for the input and ground-truth target images. The top row of each dataset shows bokeh removal, and the bottom row shows bokeh enhancement. Compared with deblur-then-render baselines, AnyBokeh performs spatially appropriate deblurring or defocus synthesis more effectively, adjusting bokeh balls faithfully by enlarging or shrinking them according to the aperture changes.

5 Experiments

Evaluation Setup. We evaluate AnyBokeh under three settings: any-to-any bokeh editing, AIF-to-bokeh rendering, and defocus deblurring (bokeh-to-AIF). In the any-to-any setting, the source and target images are captured from the same scene and viewpoint but under different focus distances and aperture values, so the source image may already contain spatially varying defocus. In the AIF-to-bokeh setting, the source image is all-in-focus, and the target image contains stronger defocus, corresponding to the conventional bokeh rendering task. In the defocus deblurring setting, the source image is defocused, and the target image is all-in-focus. Following prior bokeh rendering methods [26, 46, 32, 35], we use LPIPS [44] and DISTS [8] to measure reference-based perceptual similarity, and FID [15] for distribution-level realism. FID uses 2048-dimensional Inception features and is computed once per method, dataset, and task on the full evaluation set with identical samples.

Datasets. Although AnyBokeh is trained only on the synthetic UnrealBokeh dataset, we evaluate it on real-world bokeh datasets to assess zero-shot generalization. We use EBB! [17] and RealBokeh [32] following [26, 46, 32]. For EBB!, we use the Val294 split and construct evaluation pairs for all three tasks. For RealBokeh, we randomly sample 300 pairs from the entire test set for any-to-any evaluation. For both AIF-to-bokeh and bokeh-to-AIF tasks, we use valid pairs from the first 50 test IDs. For all settings, source–target pairs are selected from the same scene and viewpoint to ensure that the evaluation focuses on blur editing rather than geometric misalignment.

Baselines. We compare AnyBokeh with representative open-source methods for bokeh rendering and defocus deblurring. For bokeh rendering from AIF inputs, we evaluate BokehMe [26] and BokehDiff [46], as both methods provide controllable defocus synthesis from AIF images and are trained with synthetic bokeh supervision. Depth maps required by both methods are generated by Depth-Pro [5]. Focus points estimated by foreground masks [45] or edge differences are shared by all methods, including ours. For defocus deblurring, we use Restormer [43] and DRBNet [31] with their official single-image defocus deblurring checkpoints, without retraining on UnrealBokeh or the evaluation sets. Since existing bokeh rendering methods do not natively support arbitrary

Table 2: Quantitative comparison on any-to-any bokeh editing. Existing bokeh rendering methods cannot directly handle defocused inputs, so we evaluate them with deblur-then-render pipelines. Bokeh Calibration required by baselines is described in Sec. 5.1. Best results are marked in **bold**.

AIF Estimator	Bokeh Module	Editing Protocol		EBB!			RealBokeh		
		Direct Edit	No Calib.	LPIPS↓	DISTS↓	FID↓	LPIPS↓	DISTS↓	FID↓
Restormer	BokehMe	✗	✗	0.4451	0.2634	38.2801	0.3190	0.1399	48.2732
Restormer	BokehDiff	✗	✗	0.4449	0.2663	42.5686	0.3360	0.1505	61.6716
DRBNet	BokehMe	✗	✗	0.4424	0.2588	36.8142	0.3211	0.1415	48.6855
DRBNet	BokehDiff	✗	✗	0.4444	0.2634	42.1294	0.3424	0.1557	63.3783
–	**AnyBokeh (Ours)**	✓	✓	**0.4010**	**0.1426**	**24.6042**	**0.3033**	**0.0944**	**33.6655**

defocused inputs, we construct deblur-then-render baselines for any-to-any editing by pairing an AIF estimator with a bokeh module. The AIF estimator first predicts an all-in-focus image from the source input, and the bokeh module then renders the target defocus effect. This evaluation protocol represents the natural way to extend AIF-based bokeh rendering methods to any-to-any editing, and directly contrasts with AnyBokeh's CoC-conditioned relative editing formulation. We exclude Genfocus [35] from quantitative comparison because its released weights are trained on our evaluation benchmarks, and its training code is unavailable. Using the released model would make the pure zero-shot comparison unfair, while retraining it using our synthetic dataset is not feasible. We instead compare with it on held-out UnrealBokeh scenes in Sec. A.7.

More implementation details can be found in Appendix A.5.

5.1 Any-to-Any Bokeh Editing

Any-to-any bokeh editing is the main setting of our work. As discussed above, existing AIF-to-bokeh rendering methods cannot be directly applied to defocused inputs, so we evaluate them through deblur-then-render pipelines by pairing an AIF estimator with a bokeh renderer. A remaining challenge is how to set the bokeh-level parameter used by the renderer. Since these pipelines do not estimate a source-specific optical fingerprint, their target blur scale must be provided externally. A common strategy in prior work is to perform per-image search over the bokeh-level parameter K using the ground-truth target image as guidance [26, 46, 35]. However, such target-guided search is unavailable in blind post-capture editing and would make the comparison overly optimistic. Instead, we estimate a metadata-derived pseudo CoC-diameter scale following Bokeh Diffusion [11]:

$$K_{pseudo} \approx \frac{f^2 S_1}{N(S_1 - f)} \cdot \frac{W_{res}}{W_{sensor}}, \tag{6}$$

where the notation follows Sec. 3. In practice, K_{pseudo} may still differ from the effective bokeh-level parameter required by each renderer due to inaccurate depth estimation, missing or imprecise metadata, image resizing, and renderer-specific blur conventions. We therefore introduce a single global calibration scale γ and set

$$K_{calib} = \gamma K_{pseudo}. \tag{7}$$

K_{calib} corresponds to the bokeh-level parameter that prior methods search for. For each method–dataset combination, we search the per-image γ once on a subset of 100 sampled image pairs and then fix their average as the global scale for all test images. This protocol avoids per-image target-guided search while still allowing each baseline to use a reasonable global bokeh-level calibration.

AnyBokeh does not require this calibration. Its target CoC is computed by transferring the source optical fingerprint estimated from CoC_{src} and D (visual examples shown in Fig. 5), so the target blur scale is derived from the input image and the user-specified focus/aperture change. As shown in Tab. 2, AnyBokeh outperforms the deblur-then-render baselines across both EBB! and RealBokeh. These results suggest that directly modeling the source and target blur states is more effective than first converting the input into an AIF image. This advantage does not come solely from the generative backbone: forcing the same AnyBokeh models to first reconstruct an AIF image before editing clearly degrades performance (Sec. A.7). Visual results are showcased in Figs. 3 and 6.

Table 3: Bokeh rendering from all-in-focus inputs. The same bokeh-level calibration protocol as in the any-to-any setting is applied. AnyBokeh uses the full Stage 1+Stage 2 pipeline with relative target CoC transfer and requires no bokeh-level calibration. Best results are marked in **bold**.

Method	No Calib.	EBB!			RealBokeh		
		LPIPS↓	DISTS↓	FID↓	LPIPS↓	DISTS↓	FID↓
BokehMe	✗	0.3682	0.1780	38.6188	0.3166	0.1439	59.8446
BokehDiff	✗	0.4016	0.2411	58.0800	0.3309	0.1520	71.3716
AnyBokeh (Ours)	✓	**0.3615**	**0.1475**	**36.0778**	**0.3016**	**0.1125**	**45.8692**

Table 4: Defocus deblurring (bokeh-to-AIF) results. No calibration is needed for any method in this setting. Best results are marked in **bold**.

Method	EBB!			RealBokeh		
	LPIPS↓	DISTS↓	FID↓	LPIPS↓	DISTS↓	FID↓
Restormer	**0.4308**	0.1943	56.5597	**0.3125**	0.1352	57.1709
DRBNet	0.4320	0.1775	50.4203	0.3237	0.1302	51.3630
AnyBokeh (Ours)	0.4406	**0.1378**	**38.5373**	0.3423	**0.1033**	**44.1042**

5.2 Bokeh Rendering from AIF Inputs

We further evaluate the conventional AIF-to-bokeh rendering setting to test whether AnyBokeh remains effective when the source blur signal is weak. For BokehMe and BokehDiff, we use the same global bokeh-level calibration protocol as in Sec. 5.1. For AnyBokeh, we run the full pipeline and compute the target CoC using relative optical fingerprint transfer. As shown in Tab. 3, AnyBokeh achieves the best performance across all metrics on both datasets. This shows that, although AnyBokeh is designed for the more general any-to-any setting, it also performs strongly under the standard AIF-to-bokeh task. These results indicate that relative optical fingerprint transfer remains effective even when the source blur signal is weak, demonstrating the applicability of our framework to conventional AIF-to-bokeh rendering. The second and fourth rows of Fig. 3 showcase AIF-to-bokeh examples.

5.3 Defocus Deblurring

We evaluate defocus deblurring to examine whether AnyBokeh can effectively recover sharp structures when the target blur state requires removing existing defocus. We compare with Restormer [43] and DRBNet [31]. As shown in Tab. 4, AnyBokeh achieves the best DISTS and FID on both EBB! and RealBokeh, indicating improved perceptual structure and distribution-level realism. AnyBokeh obtains slightly worse LPIPS than the restoration baselines. This reflects a common trade-off in generative deblurring: severely blurred regions contain limited high-frequency evidence, and synthesized plausible details may improve visual sharpness but differ from the reference, which can be penalized by reference-based metrics. In contrast, restoration baselines may produce smoother outputs with less aggressive detail recovery that remain closer to the reference in feature space while preserving residual defocus. Qualitative comparisons in Fig. 7 further illustrate this behavior.

5.4 Ablation Studies

Single-stage vs. Two-stage. We evaluate the necessity of our two-stage design by comparing it against a single-stage alternative that follows a more conventional depth-based pipeline. This variant first estimates monocular depth from the source image, then computes both source and target CoC maps from the estimated depth and EXIF metadata, and finally feeds these maps together with the source image into Stage 2. For the target CoC, we combine the source depth with the target optical settings, since using target depth would introduce exposure bias. We use Depth-Pro [5] for high-quality monocular depth estimation. We evaluate this variant on 100 challenging RealBokeh pairs with strong source defocus, where the source f-number $< f/4.0$. Because the single-stage variant

Table 5: Ablation studies. We evaluate the effect of the two-stage design and source CoC conditioning. For both ablations, no tiling is used. All inputs are resized by the short edge and center-cropped. Best results are marked in **bold**.

Ablation	Method	No Calib.	LPIPS↓	DISTS↓	FID↓
Pipeline	Single-stage	✗	**0.3058**	0.1425	77.8112
	Two-stage (Ours)	✓	0.3110	**0.1363**	**76.8071**
Source CoC	w/o Source CoC	✓	0.3118	0.1413	81.0614
	w. Source CoC (Ours)	✓	**0.3110**	**0.1363**	**76.8071**

does not recover a source-specific optical fingerprint from the observed blur–disparity relation, we apply the same calibration protocol as in Sec. 5.1. As shown in Tab. 5, the two-stage design improves DISTS and FID over the single-stage baseline, indicating better perceptual structure and realism. The single-stage variant obtains slightly better LPIPS, likely due to the same reason discussed in Sec. 5.3. The single-stage variant relies on depth estimated directly from severely defocused inputs, leading to inaccurate scene geometry understanding and suboptimal editing. In contrast, our Stage 1 jointly predicts CoC_{src} and disparity from the defocused input, allowing the model to learn blur-aware geometry and estimate a transferable optical fingerprint. See Fig. 4 (1) for visual comparison.

With vs. Without Source CoC. We further evaluate dual-CoC conditioning in Stage 2 by removing CoC_{src} from the input while keeping CoC_{tgt} unchanged. This variant tests whether the target blur specification alone is sufficient for bokeh synthesis. As shown in Tab. 5, using CoC_{src} improves all metrics. Without CoC_{src}, the editor must infer the input blur state only from image appearance, which is ambiguous in regions with spatially varying or severe defocus. Providing both CoC_{src} and CoC_{tgt} explicitly specifies the blur-state transition, giving the model stronger guidance for more accurate editing, as shown in Fig. 4 (2).

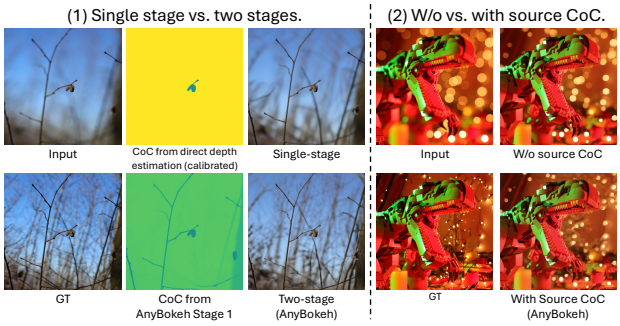


Figure 4: Visual comparisons for ablation studies. We evaluate the effect of the two-stage design and source CoC conditioning. For both ablations, no tiling is used. All inputs are resized by the short edge and center-cropped.

Additional controlled studies on forced AIF reconstruction and the choice of depth estimator are provided in Sec. A.7. We further discuss the limitations of our approach in Sec. A.10.

6 Conclusion

We introduced **AnyBokeh**, a physics-guided framework for any-to-any bokeh editing. Rather than forcing arbitrary inputs through an all-in-focus intermediate, AnyBokeh treats the source blur state as useful optical context and transfers its CoC-based optical fingerprint to the desired target focus and aperture setting. With dual-CoC conditioning, the editor explicitly models the relative transition between source and target blur states, enabling spatially adaptive bokeh editing without per-image bokeh-level calibration. Experiments across any-to-any editing, AIF-to-bokeh rendering, and bokeh-to-AIF deblurring demonstrate the effectiveness of this formulation. Our ablations further show that both source CoC estimation and dual-CoC conditioning are important for faithful relative bokeh editing. Together with the proposed **UnrealBokeh** dataset, which provides per-pixel depth and complete EXIF metadata for computing ground-truth signed CoC maps, our work offers a practical step toward physically grounded post-capture DoF editing from in-the-wild images.

Acknowledgements. This research is supported by the National Research Foundation, Singapore under its NRF Investigatorship Programme (Award No. NRF-NRFI11-2026-0008). It is also supported by Singapore MOE AcRF Tier 2 (MOE-T2EP20224-0003).

References

[1] Abdullah Abuolaim and Michael S. Brown. Defocus deblurring using dual-pixel data. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[2] Yosuke Bando and Tomoyuki Nishita. Towards digital refocusing from a single photograph. In *Proceedings of the Pacific Conference on Computer Graphics and Applications (PG)*, 2007.

[3] Jonathan T. Barron, Andrew Adams, YiChang Shih, and Carlos Hernández. Fast bilateral-space stereo for synthetic defocus. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[4] Black Forest Labs. FLUX. `https://github.com/black-forest-labs/flux`, 2024.

[5] Aleksei Bochkovskii, Amaël Delaunoy, Hugo Germain, Marcel Santos, Yichao Zhou, Stephan R. Richter, and Vladlen Koltun. Depth Pro: Sharp monocular metric depth in less than a second. In *International Conference on Learning Representations (ICLR)*, 2025.

[6] Benjamin Busam, Matthieu Hog, Steven McDonagh, and Gregory Slabaugh. SteReFo: Efficient image refocusing with stereo vision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2019.

[7] Hang Chen, Yin Xie, Xiaoxiu Peng, Lihu Sun, Wenkai Su, Xiaodong Yang, and Chengming Liu. Quad-pixel image defocus deblurring: A new benchmark and model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[8] Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli. Image quality assessment: Unifying structure and texture similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 44(5): 2567–2581, 2022.

[9] Epic Games. Unreal Engine 5.6. `https://www.unrealengine.com`, 2026.

[10] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[11] Armando Fortes, Tianyi Wei, Shangchen Zhou, and Xingang Pan. Bokeh Diffusion: Defocus blur control in text-to-image diffusion models. In *Proceedings of the SIGGRAPH Asia Conference (SIGGRAPH Asia)*, 2025.

[12] Thomas Hach, Johannes Steurer, Arvind Amruth, and Artur Pappenheim. Cinematic bokeh rendering for real scenes. In *Proceedings of the European Conference on Visual Media Production (CVMP)*, 2015.

[13] Paul Haeberli and Kurt Akeley. The accumulation buffer: hardware support for high-quality rendering. In *Proceedings of the Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 1990.

[14] Xiaoyang Han, Jianhua Li, Kewang Deng, Zukai Chen, Xuanke Shi, Sihan Wang, Boxuan Li, Linyan Wang, Siyi Xie, Xin You, Jinsheng Quan, Zhongang Cai, Haiwen Diao, Ziwei Liu, Lei Yang, Dahua Lin, and Quan Wang. Vision as unified multimodal generation. *arXiv preprint arXiv:2607.06560*, 2026.

[15] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[16] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

[17] 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 (CVPRW)*, 2020.

[18] M. Kraus and M. Strengert. Depth-of-field rendering by pyramidal image processing. *Computer Graphics Forum*, 26(3):645–654, 2007.

[19] Junyong Lee, Hyeongseok Son, Jaesung Rim, Sunghyun Cho, and Seungyong Lee. Iterative filter adaptive network for single image defocus deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[20] Sungkil Lee, Elmar Eisemann, and Hans-Peter Seidel. Depth-of-field rendering with multiview synthesis. *ACM Transactions on Graphics*, 28(5):1–6, 2009.

[21] Sungkil Lee, Elmar Eisemann, and Hans-Peter Seidel. Real-time lens blur effects and focus control. *ACM Transactions on Graphics*, 29(4):65:1–65:7, 2010.

[22] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

[23] Liyuan Pan, Shah Chowdhury, Richard Hartley, Miaomiao Liu, Hongguang Zhang, and Hongdong Li. Dual pixel exploration: Simultaneous depth estimation and image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[24] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[25] Juewen Peng, Xianrui Luo, Ke Xian, and Zhiguo Cao. Interactive portrait bokeh rendering system. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2021.

[26] 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 (CVPR)*, 2022.

[27] Juewen Peng, Jianming Zhang, Xianrui Luo, Hao Lu, Ke Xian, and Zhiguo Cao. MPIB: An MPI-based bokeh rendering framework for realistic partial occlusion effects. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.

[28] Juewen Peng, Zhiguo Cao, Xianrui Luo, Ke Xian, Wenfeng Tang, Jianming Zhang, and Guosheng Lin. BokehMe++: Harmonious fusion of classical and neural rendering for versatile bokeh creation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 47(3):1530–1547, 2025.

[29] Michael Potmesil and Indranil Chakravarty. A lens and aperture camera model for synthetic image generation. In *Proceedings of the Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 1981.

[30] Yuhui Quan, Xin Yao, and Hui Ji. Single image defocus deblurring via implicit neural inverse kernels. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[31] 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)*, 2022.

[32] Tim Seizinger, Florin-Alexandru Vasluianu, Marcos V. Conde, Zongwei Wu, and Radu Timofte. Bokehlicious: Photorealistic bokeh rendering with controllable apertures. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

[33] Yichen Sheng, Zixun Yu, Lu Ling, Zhiwen Cao, Xuaner Zhang, Xin Lu, Ke Xian, Haiting Lin, and Bedrich Benes. Dr. Bokeh: Differentiable occlusion-aware bokeh rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[34] Peng Tang, Zhiqiang Xu, Chunlai Zhou, Pengfei Wei, Peng Han, Xin Cao, and Tobias Lasser. Prior and prediction inverse kernel transformer for single image defocus deblurring. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2024.

[35] Chun-Wei Tuan Mu, Cheng-De Fan, Jia-Bin Huang, and Yu-Lun Liu. Generative refocusing: Flexible defocus control from a single image. *arXiv preprint arXiv:2512.16923*, 2025.

[36] 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*, 37(4):1–13, 2018.

[37] Jiyuan Wang, Chunyu Lin, Lei Sun, Rongying Liu, Lang Nie, Mingxing Li, Kang Liao, and Xiangxiang Chu. FE2E: From editor to dense geometry estimator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

[38] Yiyang Wang, Xi Chen, Xiaogang Xu, Yu Liu, and Hengshuang Zhao. DiffCamera: Arbitrary refocusing on images. In *Proceedings of the SIGGRAPH Asia Conference (SIGGRAPH Asia)*, 2025.

[39] Jiaze Wu, Changwen Zheng, Xiaohui Hu, and Fanjiang Xu. Rendering realistic spectral bokeh due to lens stops and aberrations. *The Visual Computer*, 29, 2012.

[40] Yan Yang, Liyuan Pan, Liu Liu, and Miaomiao Liu. K3DN: Disparity-aware kernel estimation for dual-pixel defocus deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[41] Xuan Yu, Rui Wang, and Jingyi Yu. Real-time depth of field rendering via dynamic light field generation and filtering. *Computer Graphics Forum*, 29(7):2099–2107, 2010.

[42] Yu Yuan, Xijun Wang, Yichen Sheng, Prateek Chennuri, Xingguang Zhang, and Stanley Chan. Generative photography: Scene-consistent camera control for realistic text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[43] 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 (CVPR)*, 2022.

[44] 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/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[45] Peng Zheng, Dehong Gao, Deng-Ping Fan, Li Liu, Jorma Laaksonen, Wanli Ouyang, and Nicu Sebe. Bilateral reference for high-resolution dichotomous image segmentation. *CAAI Artificial Intelligence Research*, 3:9150038, 2024.

[46] Chengxuan Zhu, Qingnan Fan, Qi Zhang, Jinwei Chen, Huaqi Zhang, Chao Xu, and Boxin Shi. BokehDiff: Neural lens blur with one-step diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

[47] Xiang Zhu, Scott Cohen, Stephen Schiller, and Peyman Milanfar. Estimating spatially varying defocus blur from a single image. *IEEE Transactions on Image Processing (TIP)*, 22(12):4879–4891, 2013.

A Technical Appendices and Supplementary Material

The supplementary material is organized as follows:

- **CoC Map Normalization** in Sec. A.1: normalization strategies of signed CoC maps for Stage 1 and Stage 2.
- **Optional Absolute CoC Control** in Sec. A.2: an optional target CoC calculation mode compatible with conventional AIF-to-bokeh rendering.
- **Optical Fingerprint Fitting** in Sec. A.3: the extreme-median fitting procedure, comparison with alternative fitting strategies, and robustness across Stage 1 sampling seeds.
- **Additional Details of UnrealBokeh** in Sec. A.4: scene collection, rendering setup, GT CoC computation, pair construction, and data split details of the UnrealBokeh dataset.
- **Implementation Details** in Sec. A.5: training and inference details of our method.
- **Stage-1 Results** in Sec. A.6: quantitative evaluation and visual examples of predicted CoC and disparity maps from Stage 1.
- **Additional Comparisons** in Sec. A.7: controlled forced-AIF and depth-source studies, and comparison with Genfocus.
- **Demo for Arbitrary Aperture and Focus Changes** in Sec. A.8: visual examples of rendering different focus points and apertures from the same source image.
- **Discussion on LPIPS Accuracy** in Sec. A.9: analysis of reference-based LPIPS metric behavior in generative bokeh editing.
- **Discussion and Limitations** in Sec. A.10: discussion on the limitations of our method.

A.1 CoC Map Normalization

We use different CoC normalization strategies for Stage 1 and Stage 2 since CoC plays different roles in the two stages. In Stage 1, CoC is a regression target and must be predicted accurately. In Stage 2, CoC is used as a conditioning signal for image generation and mainly needs to provide stable blur-state guidance.

For Stage 1, directly regressing raw pixel-space CoC values is suboptimal because CoC maps have a large dynamic range. Large out-of-focus regions can dominate the regression target, while small CoC errors near the focal plane are often more important for refocusing control, since they can shift the perceived focus boundary. We therefore use a signed logarithmic normalization:

$$CoC^{log}_{signed} = \mathrm{sign}(CoC)\frac{\log(1+|CoC|)}{\log(1+CoC_{max})}, \qquad CoC^{log}_{norm} = \mathrm{clip}\left(\frac{CoC^{log}_{signed}+1}{2}, 0, 1\right). \tag{8}$$

This transformation maps the raw CoC range $[-CoC_{max}, CoC_{max}]$ to $[0, 1]$, with zero CoC mapped to 0.5. It allocates more representational value range to near-focus regions, where accurate CoC prediction is most critical, while still preserving distinctions among large CoC values. This is important for Stage 1 because the model is explicitly supervised to predict CoC values, and over-saturating large CoC regions would weaken the learning signal for heavily defocused areas.

For Stage 2, CoC maps are not regression targets but conditioning inputs. The editor only needs them to describe the source and target blur states in a stable and bounded form. We therefore use a signed $\tanh$ normalization:

$$CoC^{tanh}_{signed} = \tanh\left(\frac{CoC}{S}\right), \qquad CoC^{tanh}_{norm} = \frac{CoC^{tanh}_{signed}+1}{2}, \tag{9}$$

where S controls the saturation scale and is set as $S = 0.1 \times \max(H, W)$. This maps raw CoC values from $(-\infty, \infty)$ to a bounded range $[0, 1]$ while preserving the CoC sign. Unlike Stage 1, saturation is acceptable here because CoC is used as guidance rather than a supervised output. The bounded representation improves numerical stability and allows the editor to handle arbitrary target CoC magnitudes, while still preserving sufficient information for the model to distinguish relative blur states.

Table 6: Fingerprint-fitting comparison on held-out UnrealBokeh scenes. κ_{src} MAE is measured against the ground-truth slope, and target-CoC metrics are computed on normalized target CoC maps. Best results are marked in **bold**.

Fitting	κ_{src} MAE↓	Target CoC		
		MAE↓	PSNR↑	SSIM↑
Least squares	16.6193	0.0582	27.0616	0.9547
RANSAC	15.9772	0.0567	27.8898	0.9513
EM (Ours)	**14.1435**	**0.0481**	**28.6776**	**0.9589**

Table 7: Robustness over 10 Stage 1 sampling seeds on held-out UnrealBokeh scenes. Each metric is reported as the median over all image pairs.

κ_{src}		Target CoC	
Std	CV	MAD_{mean}	PixStd
3.7973	0.2681	0.0118	0.0148

We avoid using a $\tanh$ normalization for Stage 1 regression because it quickly saturates for large raw CoC values. Once saturated, different large CoC magnitudes become nearly indistinguishable in the normalized space, giving the model little incentive to predict large blur values accurately. This can lead to systematically underestimated CoC in strongly defocused regions.

A.2 Optional Absolute CoC Control

The proposed AnyBokeh also supports absolute CoC control by directly specifying κ_{abs}, which recovers the standard bokeh rendering form:

$$CoC_{tgt} = \kappa_{abs}\left(D^{tgt}_{focus} - D\right). \tag{10}$$

This setting is consistent with conventional bokeh rendering pipelines, which synthesize target defocus from an AIF input using an externally specified blur strength. Since such methods do not preserve or estimate the source optical fingerprint, absolute control is their primary way to determine the target CoC scale, often requiring manual tuning or test-time search in practice. In contrast, AnyBokeh uses absolute control only as an optional interface, e.g., when users want to impose a specific absolute target blur scale. An ideal all-in-focus image with exactly zero CoC is rare in practice, and the predicted source CoC typically provides a usable, though sometimes weak, slope for relative transfer.

A.3 Optical Fingerprint Fitting

Fitting Strategy. Given the estimated source focus disparity D^{src}_{focus}, the extreme-median (EM) estimator computes local slopes $\kappa_i = CoC_{src,i}/(D^{src}_{focus} - D_i)$ from the 5% of pixels with the largest $|CoC_{src}|$ and takes their median as κ_{src}. We compare EM with two common alternatives. Least squares (LSQ) weights all pixels equally, including weakly informative near-zero-CoC pixels whose slope is dominated by prediction noise. RANSAC [10] depends on a global residual threshold and on a dominant inlier distribution, which is often violated around depth discontinuities and occlusion boundaries. As shown in Tab. 6, EM achieves the lowest κ_{src} error and the most accurate target CoC on all metrics.

Robustness to Stage 1 Sampling. Since Stage 1 is a generative model, we rerun it with 10 sampling seeds and fit κ_{src} for each run. For κ_{src}, Std measures the absolute cross-seed variation, and CV $= \text{std}(\kappa_{src})/|\text{mean}(\kappa_{src})|$ measures the variation relative to the mean slope magnitude. For the Stage-2-normalized target CoC condition, MAD_{mean} is the mean absolute difference between each seed map and the 10-seed mean map, averaged over seeds, and PixStd is the pixel-wise standard deviation across seeds, averaged spatially. As shown in Tab. 7, κ_{src} itself shows non-negligible variation across seeds. However, κ_{src} is fitted from the CoC and disparity of the same run and then combined with that same disparity to construct the target CoC, so their errors are coupled rather than independent. Consequently, the final target CoC condition varies only slightly across seeds,

indicating that the variation in κ_{src} does not propagate substantially to the conditioning signal used by Stage 2.

A.4 Additional Details of UnrealBokeh

Scene Collection. UnrealBokeh is rendered using Unreal Engine with diverse indoor and outdoor 3D scenes. We download 3D environments from the Fab marketplace of Unreal Engine [9] and obtain authorization from the authors for photo rendering in their 3D assets for image-level research use. These environments provide realistic geometry, materials, and lighting, enabling more faithful optical supervision than simplified layer-based synthetic data.

Rendering Setup. We instantiate 71 unique camera setups across the collected environments. For each camera setup, we render images under multiple focal lengths selected from $\{11, 24, 35, 55, 85, 135\}$ mm with a small random perturbation of ± 10 mm. For each focal length, we vary the f-number from $f/1.4$ to $f/22.0$ and render images with different focus distances. For every rendered image, we store the RGB image, dense depth map, focus distance, aperture, focal length, sensor size, and image resolution.

Ground-truth CoC Computation. Using the stored depth and camera parameters, we compute dense signed CoC maps following the thin-lens formulation in Sec. 3. The sign of CoC indicates whether a scene point lies in front of or behind the focal plane, and the magnitude denotes the image-space blur diameter. These dense signed CoC maps provide direct supervision for Stage 1 source CoC prediction.

Pair Construction. To construct source–target editing pairs, we group rendered images by environment, camera, and focal length. Within each group, images share the same scene geometry and viewpoint but differ in focus distance and aperture. We pair images from the same group to form aligned bokeh editing pairs. Pairs with overly similar source and target appearances are removed to improve training diversity. This process results in 11,477 rendered images and 382,888 unique source–target editing pairs.

Data Split. We split UnrealBokeh into training and validation sets by environment–camera group, resulting in 340,796 editing pairs in the training set and 42,092 pairs in the validation set. This ensures that images from the same scene and viewpoint do not appear across different splits, preventing leakage between training and validation. We use UnrealBokeh to train Stage 1 with physically accurate CoC and disparity supervision, and to train Stage 2 with aligned source–target editing pairs.

A.5 Implementation Details

For both Stage 1 and Stage 2, we use FLUX.1-Fill-dev[1] [4] as the generative backbone and fine-tune it with LoRA [16]. We set the LoRA rank to 32 and train each stage for approximately 30K iterations with a per-GPU batch size of 1. The learning rate is fixed at 10^{-4} with the AdamW optimizer [22]. Training of both stages is done on 4 NVIDIA A100 80 GB GPUs. Offset noise is applied during fine-tuning of both stages.

For Stage 1, we first resize the longer side of the source image to the model input resolution and apply reflection padding to obtain a square input. The raw CoC map is normalized to $[0, 1]$ by logarithmic normalization as described in Sec. A.1. We formulate the prediction as an inpainting task on a horizontal triptych canvas, where the left panel contains the source image I_{src}, the middle panel corresponds to the source CoC map CoC_{src}, and the right panel corresponds to the disparity map D. During inference, the source image panel is kept visible, while the CoC and disparity panels are masked as the inpainting regions, as illustrated in Fig. 1. The model then fills the two masked panels to produce image-like predictions of CoC_{src} and D. After prediction, we extract the CoC and disparity panels, remove the padding, and upsample both maps back to the original image resolution. Since CoC is defined as an image-space blur diameter in pixels, its magnitude must be rescaled consistently with the image resizing factor. Specifically, if the source image is resized by a factor s before Stage 1 prediction, the recovered CoC map is scaled by $1/s$ when mapped back to the original resolution. The CoC map is also denormalized to its original raw value range. The disparity map is spatially resized, but its values are left unchanged, since it serves as a relative geometric coordinate. The target CoC is computed in the raw value range.

[1] black-forest-labs/FLUX.1-Fill-dev

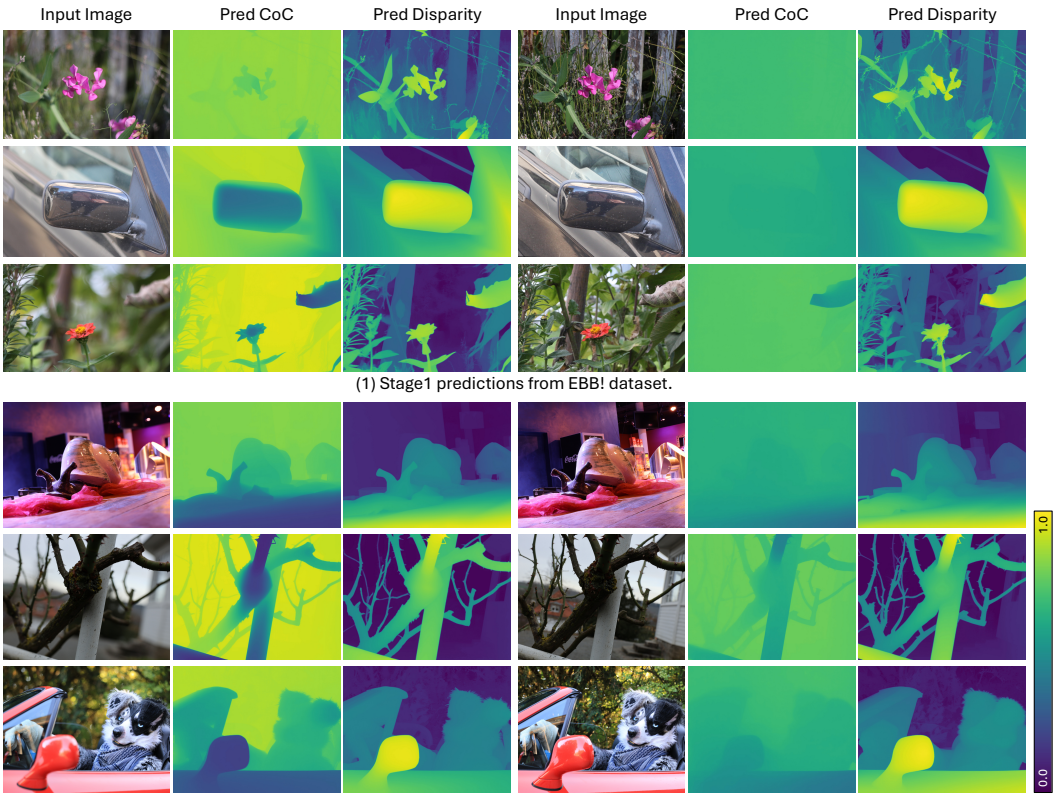


Figure 5: Qualitative Stage 1 predictions on EBB! and RealBokeh. Each row shows the same scene under two aperture settings, with stronger defocus on the left and weaker defocus on the right. Stage 1 predicts larger CoC magnitudes for stronger bokeh while producing consistent disparity maps across aperture changes, indicating that it can separate source blur state from scene geometry. CoC maps are shown after normalization (Sec. A.1), where 0.5 corresponds to $CoC = 0$ (in focus) and values toward 0 or 1 indicate larger signed defocus.

For Stage 2, we use signed $\tanh$ normalization as described in Sec. A.1 for CoC_{src} and CoC_{tgt} to represent arbitrary raw CoC values in a bounded image-like form. The input is arranged as a 2×2 grid as shown in Fig. 1: the top-left cell contains CoC_{src}, the top-right cell contains the source image I_{src}, the bottom-left cell contains CoC_{tgt}, and the bottom-right cell is the target image slot I_{tgt}. During training, we sample aligned image patches and construct one such grid for each patch. The target image slot is masked, and the model learns to synthesize I_{tgt} conditioned on the other three cells. During inference, we apply the same grid formulation to overlapping tiles of the full-resolution image. The predicted target tiles are merged using Hann-windowed overlap-add stitching with a window overlap of 0.2, preserving the original resolution and aspect ratio.

A.6 Stage 1 Results

Quantitative Results. We evaluate Stage 1 on held-out UnrealBokeh validation scenes, where ground-truth CoC and disparity are available. Since the inputs may be strongly defocused, we compare with Depth-Pro [5] and two generative geometry estimators, SenseNova-Vision [14] and FE2E [37]. All methods receive the same defocused inputs, and predicted disparity is normalized per image to $[0, 1]$. Because SenseNova-Vision and FE2E predict relative depth, physically scaled CoC cannot be derived from their outputs. To provide a favorable comparison for Depth-Pro, we combine its predicted metric depth with ground-truth camera parameters when computing CoC. As shown in Tab. 8, Stage 1 achieves the best results on all CoC and disparity metrics, even though Depth-Pro is given ground-truth optics. This indicates that jointly predicting CoC and disparity from

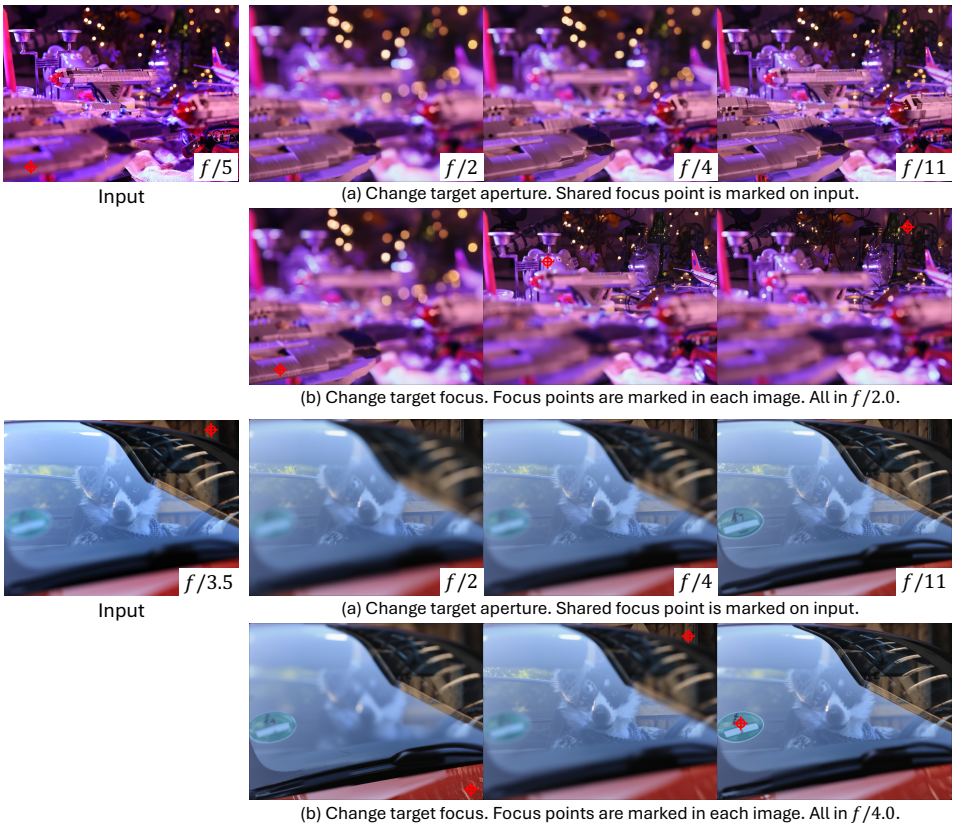


Figure 6: Qualitative results for arbitrary aperture and focus changes. Given a single input image, AnyBokeh supports user-specified aperture rendering and focus relocation without ground-truth target images. In aperture-control examples, the focus point is fixed (marked in the input), and the target f-number is varied. In focus-control examples, the aperture is fixed, and the focus point is moved to different scene locations (marked in each result image), causing the sharp region to shift accordingly.

Table 8: Quantitative Stage 1 evaluation on the UnrealBokeh validation set. Disparity is normalized per image to $[0, 1]$. SenseNova-Vision and FE2E predict relative depth, so physically scaled CoC cannot be derived from them. For Depth-Pro, CoC is computed from its predicted metric depth combined with ground-truth camera parameters. Best results are marked in **bold**.

Method	CoC			Disparity		
	MAE↓	PSNR↑	SSIM↑	MAE↓	PSNR↑	SSIM↑
Depth-Pro [5]	0.1971	14.6624	0.7283	0.1135	18.8306	0.8267
SenseNova-Vision [14]	–	–	–	0.1052	18.9385	0.8147
FE2E [37]	–	–	–	0.1239	17.3705	0.8246
Stage 1 (Ours)	**0.1011**	**19.0429**	**0.9175**	**0.0816**	**21.0009**	**0.8808**

the defocused input yields more reliable blur and geometry estimates than applying general-purpose geometry estimators to blurred images.

Qualitative Results. Fig. 5 shows qualitative predictions of the Stage 1 estimator on EBB! and RealBokeh images. Each row contains the same scene captured with different aperture settings: the left example has a smaller f-number and stronger defocus, while the right example has a larger f-number and weaker defocus. The predicted CoC maps correctly reflect this change in bokeh strength. Specifically, inputs with stronger defocus produce larger CoC magnitudes, whereas inputs with weaker defocus yield CoC values closer to the focus value of 0.5 in the normalized visualization.

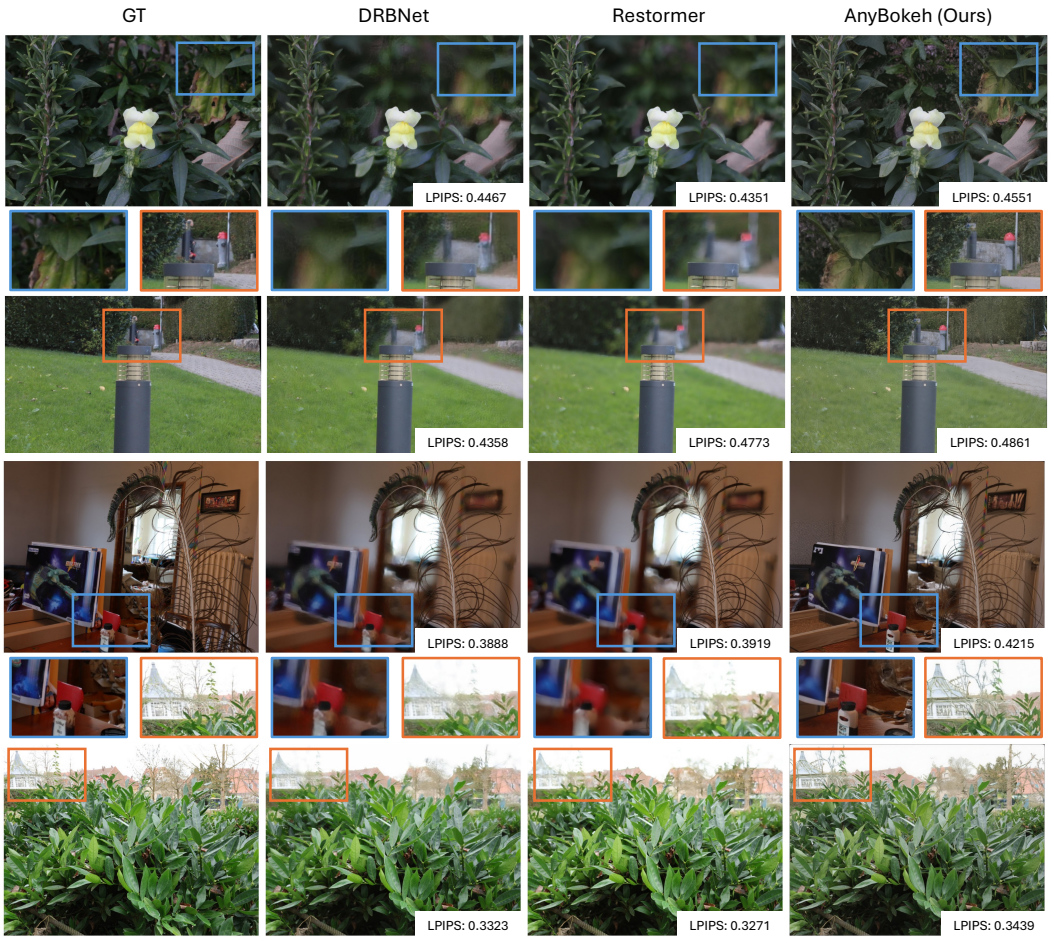


Figure 7: Qualitative examples illustrating the gap between LPIPS [44] and visual quality in defocus deblurring. Although AnyBokeh may obtain higher LPIPS than baselines, it often recovers sharper structures and more realistic local details in severely defocused regions. Restoration baselines can achieve lower LPIPS by producing smoother outputs, but they tend to preserve residual blur and under-recover fine structures.

Here, 0.5 denotes zero raw CoC, while values close to 0 or 1 indicate large signed defocus (as described in Sec. A.1).

The predicted disparity maps remain largely consistent across different aperture settings of the same scene, since scene geometry should be independent of the amount of optical blur. This consistency suggests that Stage 1 learns to separate geometric structure from defocus magnitude, which is essential for estimating a reliable optical fingerprint. Some ambiguity remains in regions where severe defocus removes most geometric evidence, such as background leaves in the third row, but the overall scene layout and major depth discontinuities are still recovered. These results demonstrate that Stage 1 can effectively infer both source CoC and disparity from inputs with varying bokeh levels, providing stable guidance for subsequent target CoC computation and CoC-conditioned editing.

A.7 Additional Comparisons

Direct Editing vs. Forced AIF. A natural question is whether the improvement of AnyBokeh over deblur-then-render baselines comes mainly from its generative backbone. To isolate the effect of mandatory AIF reconstruction, we keep the Stage 1 and Stage 2 models unchanged but force the pipeline to first recover an AIF image and then render the target bokeh from this AIF image. Specifically, we first run the full AnyBokeh pipeline with the target aperture set to $f/22$ and the

Table 9: Controlled comparisons for any-to-any bokeh editing on RealBokeh. (a) The same Stage 1 and Stage 2 models are forced to recover an AIF image before rendering the target. (b) The DRBNet-recovered AIF image and BokehMe renderer are fixed, and only the depth source is changed.

	Method	LPIPS↓	DISTS↓	FID↓
(a) Forced AIF	Ours + forced AIF	0.3933	0.1456	64.05
	Ours	**0.3033**	**0.0944**	**33.67**
(b) Depth source	DRBNet + BokehMe (Depth-Pro)	0.3211	0.1415	48.69
	DRBNet + BokehMe (Stage 1)	0.3220	0.1404	47.58

Table 10: Comparison with Genfocus [35] on held-out UnrealBokeh scenes. LoRA ranks and training steps are listed for Stage 1/Stage 2 (DeblurNet/BokehNet for Genfocus). Best results are marked in **bold**.

Method	LoRA Rank	Steps	LPIPS↓	DISTS↓	FID↓
Genfocus [35]	128/64	60K/100K	**0.2828**	0.1229	44.35
AnyBokeh (Ours)	32/32	30K/30K	0.2900	**0.1070**	**29.65**

target focus kept at the source focus, producing an AIF reconstruction. We then run AnyBokeh again on this AIF image to perform the actual edit toward the target focus and aperture. Since the backbone and target control are identical, the gap in Tab. 9 (a) reflects the cost of the AIF detour. Full AIF reconstruction is unnecessary unless the target region must become completely sharp. For preservation, further defocusing, or partial sharpening, it may hallucinate details that are subsequently re-blurred, allowing reconstruction errors to propagate into the final edit.

Effect of the Depth Estimator. Our baselines use Depth-Pro [5] depth, whereas AnyBokeh uses its Stage 1 disparity, which may confound the comparison. In Tab. 9 (b), we fix the DRBNet-recovered AIF image and the BokehMe renderer and replace only Depth-Pro with Stage 1 disparity. The two variants perform similarly, indicating that the gap between AnyBokeh and the baselines is not primarily caused by the choice of depth estimator.

Comparison with Genfocus. Since the released Genfocus weights are trained on our real evaluation benchmarks, we compare with it on held-out UnrealBokeh scenes instead. For Genfocus, we compute its bokeh level directly from the ground-truth EXIF metadata following its Eq. (3). As shown in Tab. 10, the two methods achieve comparable overall performance: Genfocus is slightly better in LPIPS, while AnyBokeh performs better in DISTS and FID, despite using smaller LoRA ranks and fewer training steps.

A.8 Demo for Arbitrary Aperture and Focus Changes

We further demonstrate the controllability of AnyBokeh under user-specified aperture and focus changes. Unlike paired benchmark evaluation, this setting does not have ground-truth target images, since the requested focus point and aperture value may correspond to a capture configuration that was never observed. We therefore provide qualitative examples to illustrate whether the model can respond to arbitrary user controls from a single input image.

Fig. 6 shows two types of edits. In the aperture-control examples, the focus point is kept fixed while the target aperture is varied. AnyBokeh consistently weakens or strengthens the defocus effect according to the specified aperture, even with existing bokeh in the input. In the focus-control examples, the aperture is kept fixed while the target focus point is moved to different scene locations. The model correspondingly shifts the sharp region to the selected focus point and adjusts foreground/background blur in a spatially coherent manner.

These examples show that AnyBokeh is not limited to reconstructing predefined source–target pairs. By combining the predicted source CoC and disparity with user-specified focus and aperture controls, our method can synthesize diverse post-capture bokeh edits from arbitrary inputs, supporting both aperture adjustment and focus relocation in a unified framework.

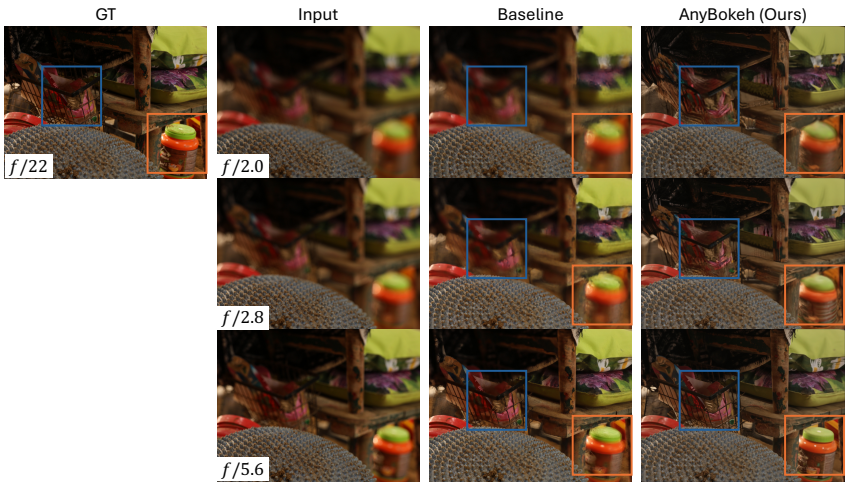


Figure 8: Failure cases under extreme source blur. (1) When the input is severely defocused (first row), missing high-frequency details limit a fully sharp reconstruction, although AnyBokeh improves as the source blur decreases (second and third rows) and remains clearer than the baseline. (2) When very large bokeh balls must be substantially reduced, AnyBokeh may leave residual or hollow bokeh artifacts, while the baseline often fails under such severe defocus. For both cases, the baseline with best quantitative results is used for comparison.

A.9 Discussion on LPIPS Accuracy on Bokeh Editing Results

In the defocus deblurring setting, AnyBokeh achieves better DISTS [8] and FID than the baselines, while obtaining higher LPIPS [44]. This discrepancy reflects a limitation of full-reference metrics for highly ill-posed generative restoration tasks. When an input region is severely defocused, the original high-frequency details are missing. A generative editor may recover sharper structures and synthesize visually plausible textures, but these details do not necessarily match the exact reference image at the pixel or feature level. As a result, LPIPS can penalize perceptually valid high-frequency recovery if the synthesized details differ from the ground truth. Fig. 7 illustrates this behavior. Compared with Restormer [43] and DRBNet [31], AnyBokeh often restores clearer object boundaries and more detailed local structures, especially in heavily blurred regions highlighted by the zoomed crops. However, because the recovered details are not guaranteed to be identical to the reference, the LPIPS score can be worse even when the visual result appears sharper and more realistic. In contrast, the restoration baselines tend to produce smoother outputs with residual defocus blur. Such outputs may remain closer to the reference in the LPIPS feature space, but they often under-recover fine structures and appear less visually sharp. Therefore, LPIPS should be interpreted with caution in this setting. It remains useful for measuring reference similarity, but it does not fully capture the perceptual quality of plausible detail recovery under severe defocus.

A.10 Discussion and Limitations

AnyBokeh is built on the observation that the blur already present in a source image is not merely a degradation to be removed, but also carries useful optical information. By estimating the source CoC and disparity, our method transfers this source-specific optical fingerprint to the desired target setting and performs editing relative to the input blur state. This formulation avoids the need to canonicalize arbitrary inputs into an all-in-focus image, and our experiments show that explicitly conditioning on both source and target CoC provides effective guidance for spatially adaptive deblurring, preservation, and defocus synthesis.

Despite these advantages, AnyBokeh still has limitations. First, the method is bound by the amount of visual evidence preserved in the source image. When the input is extremely defocused, high-frequency textures and fine geometric structures are missing, making faithful reconstruction inherently ambiguous. As shown in Fig. 8 (1), AnyBokeh may not fully recover sharp details from severely blurred inputs, although the reconstruction improves as the source blur becomes weaker and remains clearer than the baseline. A related failure mode occurs for extremely large bokeh balls, especially when the target setting requires a much larger f-number and therefore a substantial reduction of the bokeh size. In such cases, shown in Fig. 8 (2), the model may leave residual bokeh artifacts or produce hollow structures. This is partly because such extreme bokeh balls are rare in our training data, and partly because the underlying scene content is heavily occluded by the large defocus disk. UnrealBokeh is rendered by sampling cameras, objects, and lighting from existing virtual environments rather than from scenes designed for very large bokeh balls, so such cases are underrepresented. Dedicated point-light, scene, and lighting configurations could expand this coverage in future versions. The limited coverage of such extreme bokeh balls in our training data can also affect the reverse direction: when the target setting requires enlarging small highlights into very large bokeh balls, the generated bokeh size may be slightly underestimated. However, this case usually results in less pronounced bokeh rather than structural artifacts. Second, our high-resolution inference relies on a tiling-and-stitching strategy to preserve the native aspect ratio and resolution of arbitrary input images. While this improves output quality and avoids resizing the whole image to a fixed resolution, it increases runtime compared with single-pass inference. Future work could improve efficiency with more compact backbones, better tile fusion strategies, or native high-resolution generative architectures. Third, our optical fingerprint follows the ideal thin-lens CoC model and only captures blur magnitude induced by focus and aperture changes. Lens effects such as non-circular apertures, spatially varying PSFs, and spherical, chromatic, or higher-order aberrations are not modeled.

Overall, these limitations arise mainly in extreme blur or high-resolution settings, while AnyBokeh remains effective for a wide range of practical post-capture bokeh editing scenarios.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction summarize the paper's main contributions, including any-to-any bokeh editing, optical fingerprint transfer, dual-CoC conditioning, and UnrealBokeh.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper includes a dedicated discussion and limitations section in Sec. A.10. It discusses failure cases under extreme source defocus and very large bokeh balls, including the limited coverage of such bokeh balls in UnrealBokeh, as well as the computational cost introduced by high-resolution tiling-and-stitching inference. It also states that the optical fingerprint follows the ideal thin-lens CoC model and does not model non-circular apertures, spatially varying PSFs, or lens aberrations.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper does not present formal theorems. The theoretical components are based on the standard thin-lens CoC formulation and its approximation under stated assumptions, such as circular aperture and non-macro photography where the focus distance is much larger than the focal length. The derivation of the CoC–disparity slope and target CoC transfer is provided with numbered equations and referenced in the methodology.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides the experimental settings, datasets, evaluation protocols, baseline construction, metrics, and bokeh level calibration details needed to reproduce the main results. Additional details on CoC normalization, dataset construction, and implementation details are included in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code and UnrealBokeh dataset will be released.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies the training and evaluation settings, including dataset splits, evaluation tasks, metrics, baselines, optimizer and learning rate, training iterations, LoRA configuration, calibration procedure, and inference details. Additional implementation details are provided in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: The paper does not report error bars or statistical significance tests. The main results are based on deterministic evaluations over fixed benchmark splits and paired test sets using standard image-quality metrics. For the stochastic Stage 1 predictor, we report the cross-seed variation of the fitted optical fingerprint and the resulting target CoC over 10 sampling seeds (Sec. A.3). We acknowledge that reporting confidence intervals or repeated-run statistics could further strengthen the empirical analysis.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Computational resources are provided in implementation details in Sec. A.5.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: All research conducted in this paper conform with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses broader impacts of post-capture bokeh editing. The potential positive impact is that AnyBokeh can make controllable photographic editing more accessible. The potential negative impact is that, like other image editing tools, it could be used to produce misleading visual content by altering the apparent photographic capture condition. However, the method is focused on depth-of-field and bokeh manipulation rather than semantic content generation, which limits the scope of potential misuse.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The work does not involve releasing a high-risk model or scraped dataset. AnyBokeh is designed for controllable depth-of-field editing and does not generate arbitrary semantic content. The proposed UnrealBokeh dataset is synthetically rendered, and real-world datasets are only used for evaluation. Therefore, no additional release safeguards beyond standard research-use distribution are required.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We use existing Unreal Engine environments to render the UnrealBokeh dataset. We contacted the creators of the environments and received permission to capture images from their assets for research dataset construction. We will properly credit the original creators when releasing the dataset and respect the corresponding asset licenses and terms of use.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The paper introduces UnrealBokeh, a new synthetic dataset for physically supervised bokeh editing. Upon release, we will provide accompanying documentation for the dataset and code, including generation details, usage instructions, licensing and attribution information, and known limitations.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: No crowdsourcing experiments and research with human subjects are involved in this paper.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: This paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core methodology, experiments, and scientific claims of the paper do not involve LLMs as an important, original, or non-standard component. Use of LLM is limited to writing, editing, or formatting assistance.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.