DC^2: Dual-Camera Defocus Control by Learning to Refocus

Hadi Alzayer[1,2] Abdullah Abuolaim[1] Leung Chun Chan[1]
Yang Yang[1] Ying Chen Lou[1] Jia-Bin Huang[2] Abhishek Kar[1]

[1]Google [2]University of Maryland, College Park

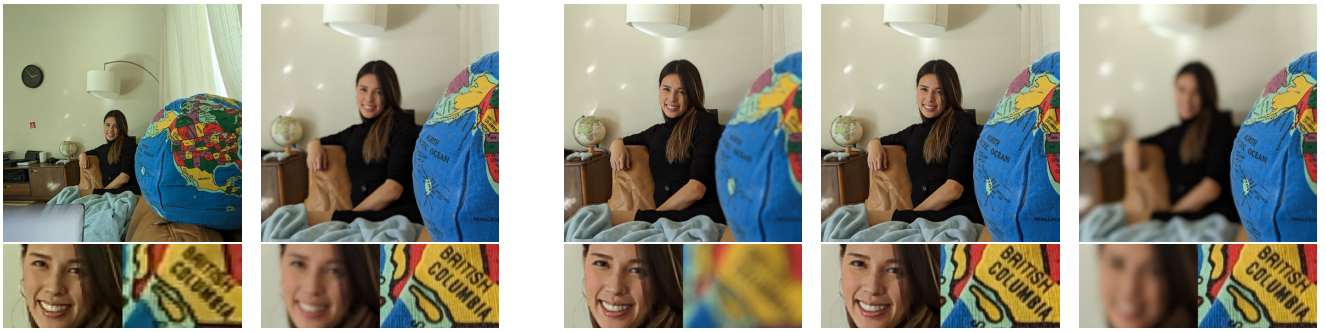


Figure 1. **Post-capture depth-of-field (DoF) control from dual camera.** (*Left*) Using photos captured from dual cameras with a wide field of view (FoV) and ultra-wide FoV, our method enables various DoF manipulations. (*Right*) We showcase our results of refocusing (changing the focal plane), deblurring (creating all-in-focus imagery), and synthesizing a shallower DoF (producing bokeh effects).

Abstract

*Smartphone cameras today are increasingly approaching the versatility and quality of professional cameras through a combination of hardware and software advancements. However, fixed aperture remains a key limitation, preventing users from controlling the depth of field (DoF) of captured images. At the same time, many smartphones now have multiple cameras with different fixed apertures - specifically, an ultra-wide camera with wider field of view and deeper DoF and a higher resolution primary camera with shallower DoF. In this work, we propose DC^2, a system for **defocus control** for synthetically varying camera aperture, focus distance and arbitrary defocus effects by fusing information from such a dual-camera system. Our key insight is to leverage real-world smartphone camera dataset by using image refocus as a proxy task for learning to control defocus. Quantitative and qualitative evaluations on real-world data demonstrate our system's efficacy where we outperform state-of-the-art on defocus deblurring, bokeh rendering, and image refocus. Finally, we demonstrate creative post-capture defocus control enabled by our method, including tilt-shift and content-based defocus effects.*

1. Introduction

Smartphone cameras are the most common modality for capturing photographs today [13]. Recent advancements in computational photography such as burst photography [18], synthetic bokeh via portrait mode [48], super-resolution [55], and more have been highly effective at closing the gap between professional DSLR and smartphone photography. However, a key limitation for smartphone cameras today is depth-of-field (DoF) control, i.e., controlling parts of the scene that appear in (and out of) focus. This is primarily an artifact of their relatively simple optics and imaging systems (e.g., fixed aperture, smaller imaging sensors, etc.). To bridge the gap, modern smartphones tend to computationally process the images for further post-capture enhancements such as synthesizing shallow DoF (e.g., portrait mode [37, 48]). However, this strategy alone does not allow for DoF *extension* or post-capture refocus. In this work, we propose Dual-Camera Defocus Control (*DC*^{2}), a framework that can provide post-capture **defocus control** leveraging multi-camera systems prevalent in smartphones today. Figure 1 shows example outputs from our framework for various post-capture DoF variations. In particular,

our method is controllable and enables image refocus, DoF extension, and reduction.

Post-capture defocus control is a compound process that involves removing defocus blur (i.e., defocus deblurring) and then adding defocus blur selectively based on the scene depth. Defocus deblurring [2, 4, 5, 23, 27, 31, 35, 39, 40, 42, 43, 59, 60], itself, is challenging due to the nature of the defocus point spread function (PSF) formation which can be spatially varying in size and shape [28, 46]. The PSF's shape and size are not only depth dependent, but also vary based on aperture size, focal length, focus distance, optical aberration, and radial distortion. Synthesizing and adding defocus blur [9, 17, 21, 33, 37, 38, 48, 57, 58] is also difficult and requires an accurate depth map along with an all-in-focus image. Additionally, it requires realistic blur formation and blending around the object's boundaries. Most prior work has addressed defocus deblurring and synthesizing defocus blur as two isolated tasks. There has been less work on post-capture defocus control (e.g., image refocusing [22, 34, 41]). The image refocusing literature [22, 34] has focused on light-field data captured with specialized hardware. While the results in [51, 52] are the state-of-the-art, light-field data is not representative of smartphone and DSLR cameras by lacking realistic defocus blur and spatial resolution [12].

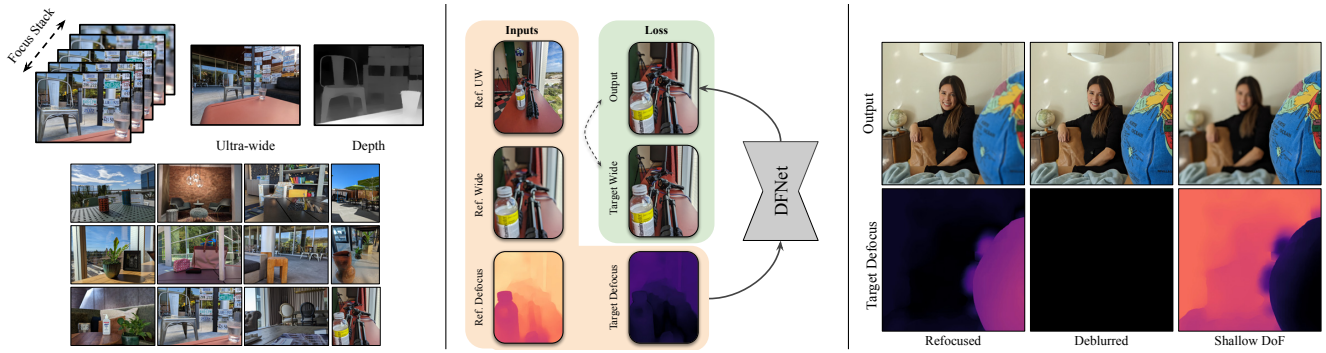
Most modern smartphones are now equipped with two or more rear cameras to assist with computational imaging. The primary camera – often referred to as the wide camera or **W** – has a higher resolution sensor, a higher focal length lens but a relatively shallower DoF. Alongside **W** is the ultra-wide (**UW**) camera, often with a lower resolution sensor, lower focal length (wider field of view) and wider DoF. Our critical insight is to leverage this unique camera setup and cross-camera DoF variations to design a system for realistic post-capture defocus control. Differently from prior work, we tackle the problem of defocus control (deblurring *and* adding blur) and propose using real-world data easily captured using a smartphone device to train our learning-based system. Our primary contributions in this work are as follows:

- We propose a learning-based system for **defocus control** on dual-camera smartphones. This subsumes the tasks of defocus deblurring, depth-based blur rendering, image refocusing and enables arbitrary post-capture defocus control.
- In the absence of defocus control ground-truth, we enable training our system on real-world data captured from a smartphone device. To achieve that, we reformulate the problem of defocus control as learning to refocus and define a novel training strategy to serve the purpose.
- We collect a dataset of diverse scenes with focus stack data at controlled lens positions the **W** camera and accompanying **UW** camera images for training our system. Additionally, we compute all-in-focus images using the focus stacks to quantitatively evaluate image refocus, defocus deblurring and depth-based blurring tasks and demonstrate superior performance compared to state-of-the-art (SoTA) methods across all three tasks.
- Finally, we demonstrate creative defocus control effects enabled by our system, including tilt-shift and content-based defocus.

2. Related Work

Defocus Deblurring Defocus blur leads to a loss of detail in the captured image. To recover lost details, a line of work follows a two-stage approach: (1) estimate an explicit defocus map, (2) use a non-blind deconvolution guided by the defocus map [23,42]. With the current advances in learning-based techniques, recent work perform single image deblurring directly by training a neural network end-to-end to restore the deblurred image [2, 27, 31, 39, 40, 43]. Due to the difficulty of the defocus deblurring task, other works try to utilize additional signals, such as the dual pixel (DP) data to improve deblurring performance [4,5,35,59,60]. DP data is useful for deblurring as it provides the model with defocus disparity that can be used to inform deblurring. While the DP data provides valuable cues for the amount of defocus blur at each pixel, the DP views are extracted from a single camera. Therefore, the performance of the DP deblurring methods drops noticeably and suffer from unappealing visual artifacts for severely blurred regions.
In the same vein, we aim to exploit the **UW** image as a complementary signal already available in modern smartphones yet ignored for DoF control. By using the **UW** image with different DoF arrangements, we can deblur regions with severe defocus blur that existing methods cannot handle because of the fundamental information loss. Nevertheless, we are aware that using another camera adds other challenges like image misalignment, occlusion, and color mismatches which we address in Section 4.3.

Bokeh Rendering Photographers can rely on shallow DoF to highlight an object of interest and add an artistic effect to the photo. The blur kernel is spatially variant based on depth as well as the camera and optics. To avoid the need of estimating depth, one work magnifies the existing defocus in the image to make the blur more apparent without explicit depth estimate [7]. Since recent work in depth estimation improved significantly [30, 44], many shallow DoF rendering methods assume having depth [37] or estimate depth in the process [48,57]. Using an input or estimated depth map, a shallow DoF can be synthesized using classical rendering methods [9, 17, 38, 48], using a neural network to add the



(a) Dual camera image refocus dataset. (b) Proxy task: Learn to refocus. (c) Evaluate on arbitrary defocus maps.

Figure 2. **Image refocus as a proxy task.** Since we cannot gather a *real* dataset for arbitrary focus manipulation, our idea is to train a model to perform *image refocus* using a target defocus map as an input. At the test time, our trained model can perform arbitrary focus manipulation by feeding it an arbitrary target defocus map.

synthetic blur [21, 33, 50] or a combination of classical and neural rendering [37]. With that said, shallow DoF synthesis methods typically assume an all-in-focus image or an input with a deep DoF.

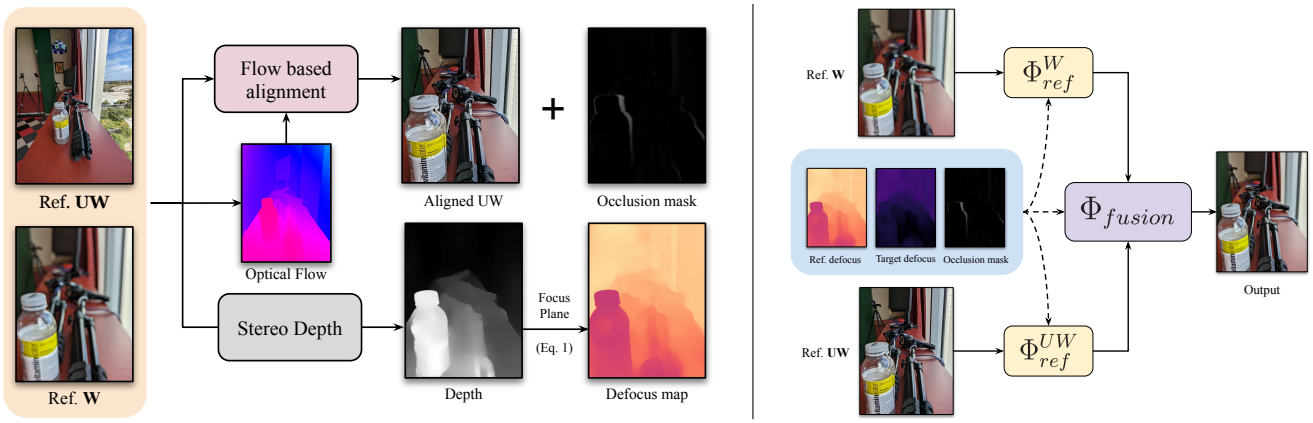
Our proposed framework learns to blur as a byproduct of learning to refocus with the insight that the refocus task involves both deblurring and selective blurring. Unlike prior work that addressed either defocus deblurring or image bokeh rendering, we introduce a generic framework that facilitates post-capture full defocus control (e.g., image refocusing).

Image Refocus and DoF Control At capture time, the camera focus can be adjusted automatically (i.e., autofocus [3, 6, 19]) or manually by moving the lens or adjusting the aperture. When the image is captured, it can still be post-processed to manipulate the focus. Nevertheless, post-capture image refocus is challenging as it requires both deblurring and blurring. Prior work uses specialized hardware to record a light field which allows post-capture focus control [34, 53]. However, light field cameras have low spatial resolution and are not representative of smartphone cameras. An alternative to requiring custom hardware is to capture a focus stack, and then merge the frames required to simulate the desired focus distance and DoF [10, 22, 29, 36], but the long capture time restricts using focus stacks to static scenes. Research on single-image refocus is limited due to its difficulty, but the typical approach is to deblur to obtain an all-in-focus image followed by blurring. Previous work used classical deblurring and blurring [8] to obtain single image refocus, and the most notable recent single-image-based image refocus is RefocusGAN [41], which trains a two-stages GAN to perform refocusing. The limited research on software-based image refocus is likely due to the challenging task that involves both defocus deblurring and selective blurring. In our work, we provide a practical setup for post-capture image refocus without the restrictions of inaccessible hardware or the constraint of capturing a focus stack. We do so by leveraging the dual camera that is available in modern smartphones.

Image Fusion. Combining information from images with complementary information captured using different cameras [36, 47] or the same camera with different capture settings [15, 18] can enhance images in terms of sharpness [22, 36, 47], illuminant estimation [1], exposure [11, 15, 18, 36], or other aspects [16, 32, 47, 49]. With the recent prevalence of dual-camera smartphones today, researchers have pursued works that target this setup. One line of work has used dual-camera for super-resolution to take advantage of the different resolutions the cameras have in still photos [51, 56, 64] as well as in videos [26]. The dual-camera setup has also been used in multiple commercial smartphones, e.g., Google Pixel devices to deblur faces by capturing an ultra-wide image with faster shutter time and fusing with the wide photo [25]. To our knowledge, we are the first to investigate using the dual-camera setup for defocus control.

3. Learning to Refocus as a Proxy Task

As mentioned, smartphone cameras tend to have fixed apertures limiting DoF control at capture time. In our work, we aim to unlock the ability to synthetically control the aperture - by transferring sharper details where present and synthesizing realistic blur. However, to train such a model, we run into a chicken and egg problem: we require a dataset of images captured with different apertures, which isn't possible with smartphones. An alternative solution could be to generate such a dataset synthetically, but modeling a realistic point spread function (PSF) for the blur kernel is non-trivial [5]. Professional DSLRs provide yet another alternative [20] but often require paired captures smartphone / DLSR captures to reduce the domain gap. Ideally, we would like to use the same camera system for both training and



(a) Data processing (b) DFNet

Figure 3. **Data processing and high-level architecture.** (*Left*) To be able to use the reference inputs for our Detail Fusion Network, we need to align the inputs and a depth estimate to approximate the defocus map of the reference **W** and the target defocus map we would like to synthesize. We use flow-based alignment with PWCNet [45] and use the stereo depth estimated using portrait mode [48]. (*Right*) Our Detail Fusion Network (DFNet) consists of refinement modules to refine the reference inputs combined with a fusion module that predicts blending masks to combine the two refined inputs.

evaluation. To resolve this, we observe that a somewhat parallel task is image refocus. When we change the focus distance, the defocus radius is adjusted in different parts of the image, involving a combination of pixels getting deblurred and blurred. This suggests that image refocus is at least as hard as scaling the DoF. Motivated by this observation, we make the hypothesis that by training a model on image refocus as a *proxy task*, we can use the same model to control the DoF at test time as we show in Figure 2. The key idea is to provide the model with reference and target defocus maps (Section 4.1) as input, and at test time control the model behavior by manipulating this target defocus map.

4. Method

To train a model on our proxy task, we need to collect a dataset of focus stacks for the wide camera and a paired ultra-wide frame which can be used as a guide due to its deeper DoF. In Figure 3 we show the high-level structure of our method dubbed DC^2. The primary module that we train is the Detail Fusion Network (DFNet), which requires a reference wide frame, (aligned) reference ultra-wide frame, and estimated defocus maps. In Section 4.1, we describe how we collect the focus stack data and process it to obtain the inputs needed for DFNet. We then describe the architecture details of DFNet in Section 4.2, which is motivated by the dual-camera input setup.

4.1. Data Processing

Using the Google Pixel 6 Pro as our camera platform, we captured a dataset of 100 focus stacks of diverse scenes, including indoor and outdoor scenarios. For each scene, we sweep the focus plane for the wide camera and capture a complete focus stack. We simultaneously capture a frame from the ultra-wide camera, which has a smaller aperture, deeper DoF, and fixed focus. For each frame, we use optical-flow-based warping using PWCNet [45] and following prior work [25] to align the ultra-wide frame with the wide frame. Since the alignment is imperfect (e.g., in textureless regions and occluded boundaries), we estimate an occlusion mask that can be used to express potentially misaligned regions for the model. To estimate defocus maps, we require the metric depth. We use the depth map embedded in the Pixel camera's portrait mode output which can estimate metric depth using dual camera stereo algorithms [63] with a known camera baseline. To compute the defocus map associated with each frame, we use the following formula for the radius of the circle of confusion c

$$c = A \frac{|S_2 - S_1|}{S_2} \frac{f}{S_1 - f} \tag{1}$$

where A is the camera aperture, S_1 is the focus distance, S_2 is the pixel depth, and f is the focal length. In Figure 2a, we show a visualization of a focus stack, associated **UW**, stereo depth, and a collection of sample scenes.

4.2. Model Architecture

Our method performs detail fusion on two primary inputs: the reference wide (**W**) and ultra-wide (**UW**) images. Since we train the model to refocus, **W** is expected to be treated as a base image, while **UW** is a guide for missing high-frequency details.

Based on this intuition, we propose **Detail Fusion Network (DFNet)** that has two refinement paths: **W** refinement path (Φ^W_{ref}), **UW** refinement path (Φ^{UW}_{ref}), and a fusion

module (Φ_{fusion}) that predicts blending masks for the refined **W** and refined **UW**. Note that the **W** refinement path never gets to see the **UW** frame and vice versa. We use a network architecture based on Dynamic Residual Blocks Network (DRBNet) [39] for our refinement modules with multi-scale refinements. For the fusion module, we use a sequence of atrous convolutions [14] for an increased receptive field and predict a blending mask for each scale. To preserve high-frequency details in the blending mask, we add upsampling layer and residual connections when predicting the blending mask of the larger scale. During training, we blend the outputs of Φ^{W}_{ref} and Φ^{UW}_{ref} and compute the loss for all scales for improved performance. In Figure 3 we show a high-level diagram of our architecture and how each component interacts with the others. By visualizing the intermediate outputs between our different modules, we observe that the network indeed attempts to maintain the low-frequency signal from**W** while utilizing high-frequency signals from **UW**. Please refer to the supplementary material for a detailed model architecture and a deeper analysis of model behavior and visualizations.

4.3. Training Details

We train our model by randomly sampling slices from the focus stack in our training scenes. For each element in the batch, we randomly sample a training scene, and sample two frames to use as reference and target images, respectively. While we can approximate depth from all pairs, severely blurry frames can have unreliable depth. To address that, we use the stereo pair with the greatest number of matched features to use for the scene depth to compute the defocus maps. We train on randomly cropped 256x256 patches, using a batch size of 8, and a learning rate of 10^{-4} for 200k iterations, and then reduce the learning rate to 10^{-5} for another 200k iterations using Adam [24]. Our reconstruction loss is a combination of L_1 loss on pixels and gradient magnitudes, SSIM loss [54], and perceptual loss [61]. For a target wide frame $\mathbf{W}_{tgt}$ and a model output y, the loss is

$$\begin{aligned} L_{total} = L_1(\mathbf{W}_{tgt}, y) + L_1(\nabla\mathbf{W}_{tgt}, \nabla y) \\ L_{SSIM}(\mathbf{W}_{tgt}, y)) + L_{VGG}(\mathbf{W}_{tgt}, y) \end{aligned} \tag{2}$$

5. Experimental Results

We train our method to perform defocus control through training on the *proxy task* of image refocus. As a result, our model can perform a variety of related defocus control tasks. Specifically, we evaluate our method on defocus deblurring, synthesizing shallow DoF, and image refocus.

Evaluation metrics. We use the standard signal processing metrics, i.e., the peak signal-to-noise ratio (PSNR) and the structural similarity index measure (SSIM). We also report the learned perceptual image patch similarity (LPIPS) [62].

Table 1. **Defocus deblurring evaluation.** Performance on generating all-in-focus images from a single slice in the focus stack. The best results are in bold numbers.

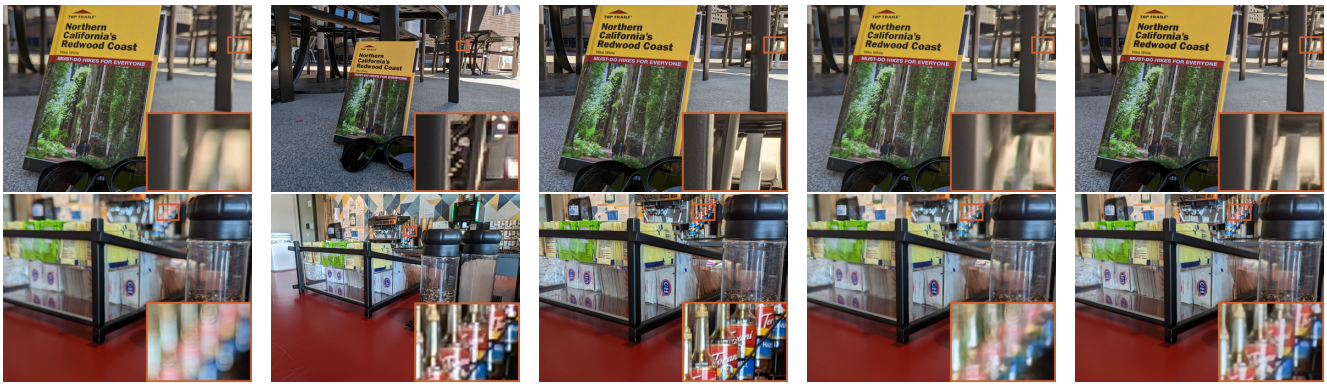
Method	PSNR ↑	SSIM ↑	LPIPS ↓
MDP [2]	23.50	0.674	0.394
IFAN [27]	23.48	0.679	0.371
DRBNet [39]	24.27	0.681	0.377
Ours	**24.79**	**0.704**	**0.351**

5.1. Defocus Deblurring

Task. The goal of defocus deblurring is to remove the defocus blur in the image. For our method to perform defocus deblurring, we simply set the target defocus map to all zeros. To obtain an all-in-focus image as a ground truth, we perform focus stacking using our focus stacks through commercial software provided by HeliconFocus. Then the evaluation task is deblurring individual slices from the focus stack to generate an all-in-focus image. Due to the focus magnification between the focus stack slices, we align the field-of-view (FoV) with the all-in-focus image through a combination of FoV matching and brute-force search for the best scaling and translation parameters that minimize the error. We use the same alignment method when evaluating all the methods to ensure fairness.

Methods. We compare our method with the following single-image defocus deblurring methods: Dynamic Residual Blocks Network (DRBNet) [39], Multi-task DP (MDP) network [2], and Iterative Filter Adaptive Network (IFAN) [27]. Note that these methods do not take the ultra-wide image as input, and the main purpose of the comparison is to highlight the value of leveraging an available dual-camera setup. Our dataset does not contain DP data and thus we are not able to benchmark the DP defocus deblurring methods [4,5,35,59,60]. As for the evaluation on other defocus deblurring datasets (e.g., [4]), our method requires dual-camera input not available in current datasets.

Evaluation. In Table 1, we compare the performance of our method against other defocus deblurring methods. Our method achieves the best results on all metrics with dual camera inputs. Note that our method has never seen all-in-focus outputs / zero target defocus maps during training and learns to deblur via the proxy task. Figure 4 shows two deblurring results of our method against DRBNet [39]. As shown in the zoomed-in insets, our method is able to restore severely blurred regions better compared to DRBNet. In general, single-image defocus deblurring methods suffer from artifacts and tend to hallucinate when restoring severely blurred regions. Therefore, an additional signal such as the **UW** is very useful when the details are completely lost in the input image. While the main task of our



Input **W** | Input **UW** | GT | DRBNet [39] | Ours

Figure 4. **Defocus deblurring.** We showcase the results of our method against SoTA single image defcous deblurring DRBNet [39]. Note that our method restores severely blurred regions in the background that single-image based methods often struggle with.

Table 2. **Bokeh blurring evaluation.** performance on simulating different slices of the focus stack from the all-in-focus image.

Method	PSNR ↑	SSIM ↑	LPIPS ↓
BokehMe [37]	26.65	0.870	0.241
Neural Rend. [37]	27.87	0.874	0.246
Classic Rend. [37]	26.66	0.870	0.241
Ours	**29.78**	**0.898**	**0.172**

Table 3. **Image refocus evaluation.** Performance on re-synthesizing focus planes given an input with different focus plane from the same scene.

Method	PSNR ↑	SSIM ↑	LPIPS ↓
UW + Blur [37]	21.89	0.803	0.364
Deblur [39]+Reblur [37]	26.40	0.833	0.312
Ours	**28.58**	**0.860**	**0.217**

proposed method is not only defocus deblurring, it achieves the SoTA deblurring results quantitatively and qualitatively. These results also demonstrate how generic and flexible our proposed defocus control framework is.

5.2. Shallow DoF Rendering

Task. We also evaluate our method on rendering shallow DoF images. The input to the method is an all-in-focus image, an approximate target defocus map, and the desired output is the image with a synthetic shallow DoF guided by the defocus map. We use the all-in-focus image generated from the focus stack as input and try to reconstruct the various slices in the focus stack using each slice's defocus map as a target.

Methods. We compare against BokehMe [37], a recent state-of-the-art in shallow DoF synthesis that relies on blending the outputs of classic blur synthesis with neural rendering methods. We also evaluate the classical scattering-based blur and the neural renderer within BokehMe in isolation.

Evaluation. In Table 2, we show that our method is competitive with SoTA shallow DoF rendering methods. Note that for DoF reduction, **UW** does not provide a useful signal since the task primarily involves signal removal from **W**, but the model learns to perform this task as a byproduct of training on image refocus. In Figure 5 we show visual results where our model synthesizes realistic blur.

5.3. Image Refocus

Task. Image refocus involves shifting the focus plane and as a result, the near and far focus depths. To evaluate on image refocus, we randomly sample two frames from a focus stack, a reference frame, and a target frame, and evaluate the model performance in reproducing the target frame.

Methods. There is limited work on single-image refocus, the most notable work being RefocusGAN [41]. The idea behind RefocusGAN is to use generative models to deblur the image followed by blurring it. This approach is likely because of the difficulty of realistically switching between different defocus amounts directly [7]. However, we are not able to compare with RefocusGAN as the code and trained models are not available. As an alternative for comparison, we adopt SoTA in defocus deblurring (DRBNet [39]) and SoTA in blurring (BokehMe [37]) for image refocus. We also compare against blurring the aligned **UW** directly since it could approximate an all-in-focus image.

Evaluation. In Table 3 we show that our method outperforms the baseline in image refocus. Note that since we train our method to switch between the reference defocus to the target defocus, the model can implicitly learn to switch between different PSF scales from the data. We show visual results in Figure 6. Note that when the target image contains

Input **W** | Input **UW** | GT | BokehMe [37] | Ours

Figure 5. **Blurring results.** Our method can synthesize shallow DoF from an all-in-focus image with a performance competitive with SoTA in bokeh rendering [37].

Input **W** | Input **UW** | GT | Baseline [37, 39] | Ours

Figure 6. **Refocus results.** We shift the focus plane and demonstrate that we can match the desired refocused image and the target blur without completely deblurring the input. Our method outperforms the baseline that refocuses by deblurring and reblurring the image.

Table 4. **Ablations on Image Input**. Comparison on different input types. Although performance increases by removing the occlusion mask, qualitative performance drops (see Figure 7).

Method	**PSNR** ↑	**SSIM** ↑	**LPIPS** ↓
W only	28.44	0.855	0.260
UW only	22.66	0.822	0.307
No occlusion	**28.81**	**0.864**	0.219
Full input	28.58	0.860	**0.217**

blurry regions like shown on the wall, our method deblurs the input just enough to match the target defocus.

5.4. Ablation Study

The key idea of our work is using the ultra-wide camera as a guide to performing DoF control. To evaluate the effects of using **UW**, we train a model using only W (only keeping the Wide refinement module) and similarly training a **UW** only model. We compare their performance on image refocus in Table 4. Note that while the wide input is sufficient when the target involves only blurring or minimal deblurring, it is an ill-posed setup when it requires considerable deblurring. On the other hand, the warped **UW** lower quality severely limits the performance when relying on it completely. We visualize an example in In Figure 7. Note that when using **W** only, deblurring performance is limited. Also we note that when removing the occlusion mask, while signal-processing metrics could see slight improvements, qualitative performance drops as we can observe ghosting artifacts around occluded boundaries.

Applications. Our method allows for arbitrary target defocus maps as an input. In Figure 8 we demonstrate a *tilt-shift* effect, where a large scene appears smaller because of the blur, as well as using a segmentation mask to deblur objects of interest (the person) while blurring the remaining objects.

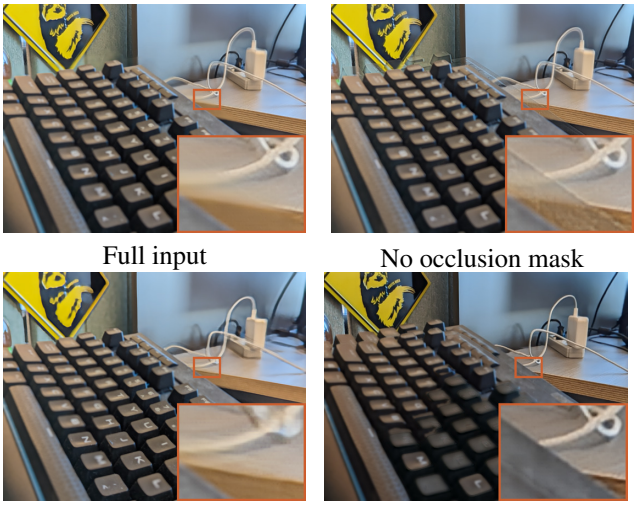


Figure 7. **Ablation.** Using the occlusion mask helps the model avoid transferring warping artifacts to the final image, while using **W** only hinders deblurring performance, and **UW** only suffers from warping artifacts and lower resolution.

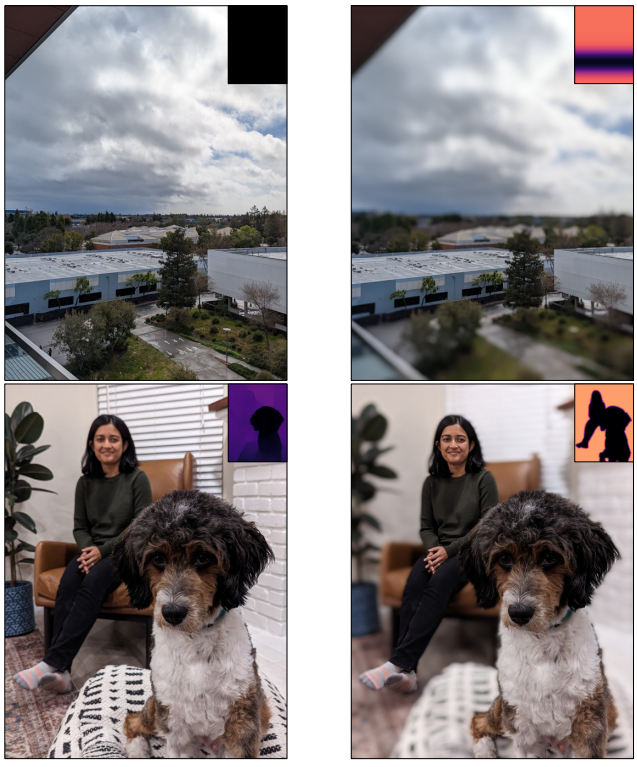


Figure 8. **Creative applications**. (*Top*) we apply tilt shift effect that makes large objects appear as miniatures. (*Bottom*) we use segmentation mask to deblur objects of interest and blur the background.

Figure 9. **Failure cases.** (*Top*) In regions where **UW** is out of focus, as in the table shown above, our method would not be able to restore the sharp texture shown in the **W** image when focused on that object. (*Bottom*) Aligning the **UW** using optical flow based alignment could suffer when **W** is severly blurred like shown on the warping artifacts on the window.

6. Limitations and Conclusion

We present DC^2, a novel framework for defocus control with dual-camera consumer smartphones. We bypass the need for synthetic data and domain gap issues by training with real data captured with a smartphone device. We do so by re-framing the defocus control problem as refocus and designing a learning-based solution for the same. The key idea behind our method is to use **UW** input as an additional signal to control defocus of **W**. Naturally, a limitation then is the DoF of **UW** itself; as objects outside of its DoF might not be sharper than in **W**. In general, our method benefits from asymmetry in the **W** and **UW** camera configurations and likely won't perform as well in systems with identical cameras. Another limitation is our dependence on pre-existing optical flow and stereo depth algorithms which can suffer from severe artifacts with defocus blur (Figure 9). A promising avenues for future work includes utilizing additional cameras to jointly model both scene depth and defocus control.

Acknowledgment We would like to thank Junlan Yang, Xiaotong Wu, Lun-Cheng Chu, Mauricio Delbracio, Yichang Shih, and Seang Chau for their support and fruitful discussions.

References

[1] Abdelrahman Abdelhamed, Abhijith Punnappurath, and Michael S Brown. Leveraging the availability of two cameras for illuminant estimation. In *CVPR*, 2021. 3

[2] Abdullah Abuolaim, Mahmoud Afifi, and Michael S Brown. Improving single-image defocus deblurring: How dual-pixel

images help through multi-task learning. In *WACV*, 2022. 2, 5

[3] Abdullah Abuolaim and Michael Brown. Online lens motion smoothing for video autofocus. In *WACV*, 2020. 3

[4] Abdullah Abuolaim and Michael S Brown. Defocus deblurring using dual-pixel data. In *ECCV*, 2020. 2, 5

[5] Abdullah Abuolaim, Mauricio Delbracio, Damien Kelly, Michael S Brown, and Peyman Milanfar. Learning to reduce defocus blur by realistically modeling dual-pixel data. In *ICCV*, 2021. 2, 3, 5, 11

[6] Abdullah Abuolaim, Abhijith Punnappurath, and Michael S Brown. Revisiting autofocus for smartphone cameras. In *ECCV*, 2018. 3

[7] Soonmin Bae and Frédo Durand. Defocus magnification. In *Computer Graphics Forum*, volume 26, pages 571–579. Wiley Online Library, 2007. 2, 6

[8] Yosuke Bando and Tomoyuki Nishita. Towards digital refocusing from a single photograph. In *15th Pacific Conference on Computer Graphics and Applications (PG'07)*, pages 363–372. IEEE, 2007. 3

[9] Jonathan T. Barron, Andrew Adams, YiChang Shih, and Carlos Hernández. Fast bilateral-space stereo for synthetic defocus. In *CVPR*, pages 4466–4474, 2015. 2

[10] Shiveta Bhat and Deepika Koundal. Multi-focus image fusion techniques: a survey. *Artificial Intelligence Review*, 54(8):5735–5787, 2021. 3

[11] Radu Ciprian Bilcu, Adrian Burian, Aleksi Knuutila, and Markku Vehvilainen. High dynamic range imaging on mobile devices. In *2008 15th IEEE International Conference on Electronics, Circuits and Systems*, pages 1312–1315. IEEE, 2008. 3

[12] Vivek Boominathan, Kaushik Mitra, and Ashok Veeraraghavan. Improving resolution and depth-of-field of light field cameras using a hybrid imaging system. In *ICCP*, 2014. 2

[13] David Cardinal. Smartphones vs cameras: Closing the gap on image quality. https://www.dxomark.com/smartphones-vs-cameras-closing-the-gap \-on-image-quality/, Apr 2021. 1

[14] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 5, 11

[15] Paul E Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. In *ACM SIGGRAPH 2008 classes*, pages 1–10, 2008. 3

[16] Clement Godard, Kevin Matzen, and Matt Uyttendaele. Deep burst denoising. In *ECCV*, September 2018. 3

[17] Thomas Hach, Johannes Steurer, Arvind Amruth, and Artur Pappenheim. Cinematic bokeh rendering for real scenes. In *Proceedings of the 12th European Conference on Visual Media Production*, CVMP '15, New York, NY, USA, 2015. Association for Computing Machinery. 2

[18] Samuel W. Hasinoff, Dillon Sharlet, Ryan Geiss, Andrew Adams, Jonathan T. Barron, Florian Kainz, Jiawen Chen, and Marc Levoy. Burst photography for high dynamic range and low-light imaging on mobile cameras. *ACM Trans. Graph.*, 35(6), dec 2016. 1, 3

[19] Charles Herrmann, Richard Strong Bowen, Neal Wadhwa, Rahul Garg, Qiurui He, Jonathan T Barron, and Ramin Zabih. Learning to autofocus. In *CVPR*, 2020. 3

[20] Andrey Ignatov, Nikolay Kobyshev, Radu Timofte, Kenneth Vanhoey, and Luc Van Gool. Dslr-quality photos on mobile devices with deep convolutional networks. *ICCV*, 2017. 3

[21] Andrey Ignatov, Jagruti Patel, and Radu Timofte. Rendering natural camera bokeh effect with deep learning. In *CVPR Workshops*, 2020. 2, 3

[22] David E Jacobs, Jongmin Baek, and Marc Levoy. Focal stack compositing for depth of field control. *Stanford Computer Graphics Laboratory Technical Report*, 1(1):2012, 2012. 2, 3

[23] Ali Karaali and Claudio Rosito Jung. Edge-based defocus blur estimation with adaptive scale selection. *IEEE Transactions on Image Processing*, 27(3):1126–1137, 2018. 2

[24] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014. cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015. 5

[25] Wei-Sheng Lai, Yichang Shih, Lun-Cheng Chu, Xiaotong Wu, Sung-Fang Tsai, Michael Krainin, Deqing Sun, and Chia-Kai Liang. Face deblurring using dual camera fusion on mobile phones. *ACM Transactions on Graphics (TOG)*, 41(4), jul 2022. 3, 4

[26] Junyong Lee, Myeonghee Lee, Sunghyun Cho, and Seungyong Lee. Reference-based video super-resolution using multi-camera video triplets. In *CVPR*, 2022. 3

[27] Junyong Lee, Hyeongseok Son, Jaesung Rim, Sunghyun Cho, and Seungyong Lee. Iterative filter adaptive network for single image defocus deblurring. In *CVPR*, 2021. 2, 5

[28] Anat Levin, Yair Weiss, Fredo Durand, and William T Freeman. Understanding blind deconvolution algorithms. *IEEE transactions on pattern analysis and machine intelligence*, 33(12):2354–2367, 2011. 2

[29] Shutao Li, James T Kwok, and Yaonan Wang. Combination of images with diverse focuses using the spatial frequency. *Information fusion*, 2(3):169–176, 2001. 3

[30] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *CVPR*, 2018. 2

[31] Haoyu Ma, Shaojun Liu, Qingmin Liao, Juncheng Zhang, and Jing-Hao Xue. Defocus image deblurring network with defocus map estimation as auxiliary task. *IEEE Transactions on Image Processing*, 31:216–226, 2022. 2

[32] Ben Mildenhall, Jonathan T Barron, Jiawen Chen, Dillon Sharlet, Ren Ng, and Robert Carroll. Burst denoising with kernel prediction networks. In *CVPR*, 2018. 3

[33] Hariharan Nagasubramaniam and Rabih Younes. Bokeh effect rendering with vision transformers. *TechRxiv*, 2022. 2, 3

[34] Ren Ng, Marc Levoy, Mathieu Br, Gene Duval, Mark Horowitz, Pat Hanrahan, and Duval Design. *Light Field Photography with a Hand-held Plenoptic Camera*. PhD thesis, Stanford University, 2005. 2, 3

[35] Liyuan Pan, Shah Chowdhury, Richard Hartley, Miaomiao Liu, Hongguang Zhang, and Hongdong Li. Dual pixel exploration: Simultaneous depth estimation and image restoration. In *CVPR*, pages 4340–4349, June 2021. 2, 5

[36] Sujoy Paul, Ioana S Sevcenco, and Panajotis Agathoklis. Multi-exposure and multi-focus image fusion in gradient domain. *Journal of Circuits, Systems and Computers*, 25(10):1650123, 2016. 3

[37] Juewen Peng, Zhiguo Cao, Xianrui Luo, Hao Lu, Ke Xian, and Jianming Zhang. Bokehme: When neural rendering meets classical rendering. In *CVPR*, 2022. 1, 2, 3, 6, 7

[38] Kuldeep Purohit, Maitreya Suin, Praveen Kandula, and Rajagopalan Ambasamudram. Depth-guided dense dynamic filtering network for bokeh effect rendering. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 3417–3426, 2019. 2

[39] Lingyan Ruan, Bin Chen, Jizhou Li, and Miuling Lam. Learning to deblur using light field generated and real defocus images. In *CVPR*, 2022. 2, 5, 6, 7, 11

[40] Lingyan Ruan, Bin Chen, Jizhou Li, and Miu-Ling Lam. Aifnet: All-in-focus image restoration network using a light field-based dataset. *IEEE Transactions on Computational Imaging*, 7:675–688, 2021. 2

[41] Parikshit Sakurikar, Ishit Mehta, Vineeth N. Balasubramanian, and P. J. Narayanan. Refocusgan: Scene refocusing using a single image. In *ECCV*, page 519–535, Berlin, Heidelberg, 2018. Springer-Verlag. 2, 3, 6

[42] Jianping Shi, Li Xu, and Jiaya Jia. Just noticeable defocus blur detection and estimation. In *CVPR*, pages 657–665, 2015. 2

[43] Hyeongseok Son, Junyong Lee, Sunghyun Cho, and Seungyong Lee. Single image defocus deblurring using kernel-sharing parallel atrous convolutions. In *ICCV*, 2021. 2

[44] Pratul P. Srinivasan, Rahul Garg, Neal Wadhwa, Ren Ng, and Jonathan T. Barron. Aperture supervision for monocular depth estimation. In *CVPR*, June 2018. 2

[45] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In *CVPR*, 2018. 4

[46] Huixuan Tang and Kiriakos N Kutulakos. Utilizing optical aberrations for extended-depth-of-field panoramas. In *ACCV*, 2012. 2

[47] Marc Comino Trinidad, Ricardo Martin Brualla, Florian Kainz, and Janne Kontkanen. Multi-view image fusion. In *ICCV*, 2019. 3

[48] Neal Wadhwa, Rahul Garg, David E. Jacobs, Bryan E. Feldman, Nori Kanazawa, Robert Carroll, Yair Movshovitz-Attias, Jonathan T. Barron, Yael Pritch Knaan, and Marc Levoy. Synthetic depth-of-field with a single-camera mobile phone. *SIGGRAPH*, 2018. 1, 2, 4

[49] Jian Wang, Tianfan Xue, Jonathan T Barron, and Jiawen Chen. Stereoscopic dark flash for low-light photography. In *ICCP*, 2019. 3

[50] Lijun Wang, Xiaohui Shen, Jianming Zhang, Oliver Wang, Zhe Lin, Chih-Yao Hsieh, Sarah Kong, and Huchuan Lu. Deeplens: Shallow depth of field from a single image. *ACM Transactions on Graphics*, 37:1–11, 12 2018. 3

[51] Tengfei Wang, Jiaxin Xie, Wenxiu Sun, Qiong Yan, and Qifeng Chen. Dual-camera super-resolution with aligned attention modules. In *International Conference on Computer Vision (ICCV)*, 2021. 2, 3

[52] Ting-Chun Wang, Jun-Yan Zhu, Nima Khademi Kalantari, Alexei A. Efros, and Ravi Ramamoorthi. Light field video capture using a learning-based hybrid imaging system. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2017)*, 36(4), 2017. 2

[53] Yingqian Wang, Jungang Yang, Yulan Guo, Chao Xiao, and Wei An. Selective light field refocusing for camera arrays using bokeh rendering and superresolution. *IEEE Signal Processing Letters*, 26(1):204–208, jan 2019. 3

[54] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 5

[55] Bartlomiej Wronski, Ignacio Garcia-Dorado, Manfred Ernst, Damien Kelly, Michael Krainin, Chia-Kai Liang, Marc Levoy, and Peyman Milanfar. Handheld multi-frame super-resolution. *ACM Trans. Graph.*, 38(4), jul 2019. 1

[56] Yubin Wu, Lianglun Cheng, Tao Wang, and Heng Wu. Infrared and visible light dual-camera super-resolution imaging with texture transfer network. *Signal Processing: Image Communication*, 108:116825, 2022. 3

[57] Ke Xian, Juewen Peng, Chao Zhang, Hao Lu, and Zhiguo Cao. Ranking-based salient object detection and depth prediction for shallow depth-of-field. *Sensors*, 21(5), 2021. 2

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

[59] Shumian Xin, Neal Wadhwa, Tianfan Xue, Jonathan T. Barron, Pratul P. Srinivasan, Jiawen Chen, Ioannis Gkioulekas, and Rahul Garg. Defocus map estimation and deblurring from a single dual-pixel image. *ICCV*, 2021. 2, 5

[60] Ruikang Xu, Zeyu Xiao, Jie Huang, Yueyi Zhang, and Zhiwei Xiong. Edpn: Enhanced deep pyramid network for blurry image restoration. In *CVPR Workshops*, June 2021. 2, 5

[61] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 5

[62] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 5

[63] Yinda Zhang, Neal Wadhwa, Sergio Orts-Escolano, Christian Häne, Sean Fanello, and Rahul Garg. Du2net: Learning depth estimation from dual-cameras and dual-pixels. In *ECCV*, Berlin, Heidelberg, 2020. Springer-Verlag. 4

[64] Zhilu Zhang, Ruohao Wang, Hongzhi Zhang, Yunjin Chen, and Wangmeng Zuo. Self-supervised learning for real-world super-resolution from dual zoomed observations. In *ECCV*, 2022. 3

A. Video Visualization

One major advantage of our method is the fine-grained control we can have on the defocus control. As a result, we can directly simulate changing the focus distance and aperture smoothly just like if we had a DSLR camera with variable focal length and aperture. Please refer to the video provided in the supplementary materials.

B. Detailed Architecture

The model architecture consists of three primary modules: Φ^{W}_{ref} to refine **W**, Φ^{UW}_{ref} to refine **UW**, and a fusion model Φ_{fusion} to predict a blending mask to blend the refined outputs. Both Φ^{W}_{ref} and Φ^{UW}_{ref} use DRBNet architecture [39] that utilize kernels prediction to refine the input. Each refinement module predicts intermediate outputs in a multi-scale setup that can be used to speed up training. Specifically, the model generates refined outputs at the following scales: 8x downsampled, 4x downsampled, 2x downsampled, and the original resolution. To be able to fuse all the multi-scale outputs, Φ_{fusion} consists of several Atrous Spatial Pyramid Pooling (ASPP) convolutions blocks [14] to predict blending mask for each scale. The ASPP blocks for each scale take the refined **W** and **UW** of the associated scale, as well as an upsampled blending mask from the previous ASPP block with a residual connection of the upsampled mask (except for the first ASPP block since it has no preceding blending mask). There are two hyperparameters associated with the blending block for each scale: (1) atrous rates for the atrous convolutions, and (2) the number of channels each intermediate step of atrous convolutions outputs. In table 5, we include a list of the hyperparameters for the blending block associated with each scale.

One issue with training the model in using cropped patches is that the blur kernel is spatially varying depending on the crop position. To resolve the ambiguity, we follow the solution proposed by Abuolaim *et al*. [5] and concatenate a radial mask to the inputs of all modules where the pixel values of the mask are the distance from the original image center, normalized.

Table 5. **Fusion model (Φ_{fusion}) hyperparameters.** The hyeraprameters for the ASPP convolution blocks are the atrous rates for the atrous convolutions, and the channels each layer outputs. The number of atrous convolution layers is the size of the channel list. Note that the final output consists of two channels which correspond to the **W** and **UW** blending masks.

Blending Block	atrous rates	channels
1/8x scale	1,3,5	16, 32, 2
1/4x scale	1, 3, 6, 12	16, 32, 2
1/2x scale	1, 3, 6, 12, 15	16, 32, 2
1x scale	1,3,6, 12, 15, 18	16, 32, 32, 2

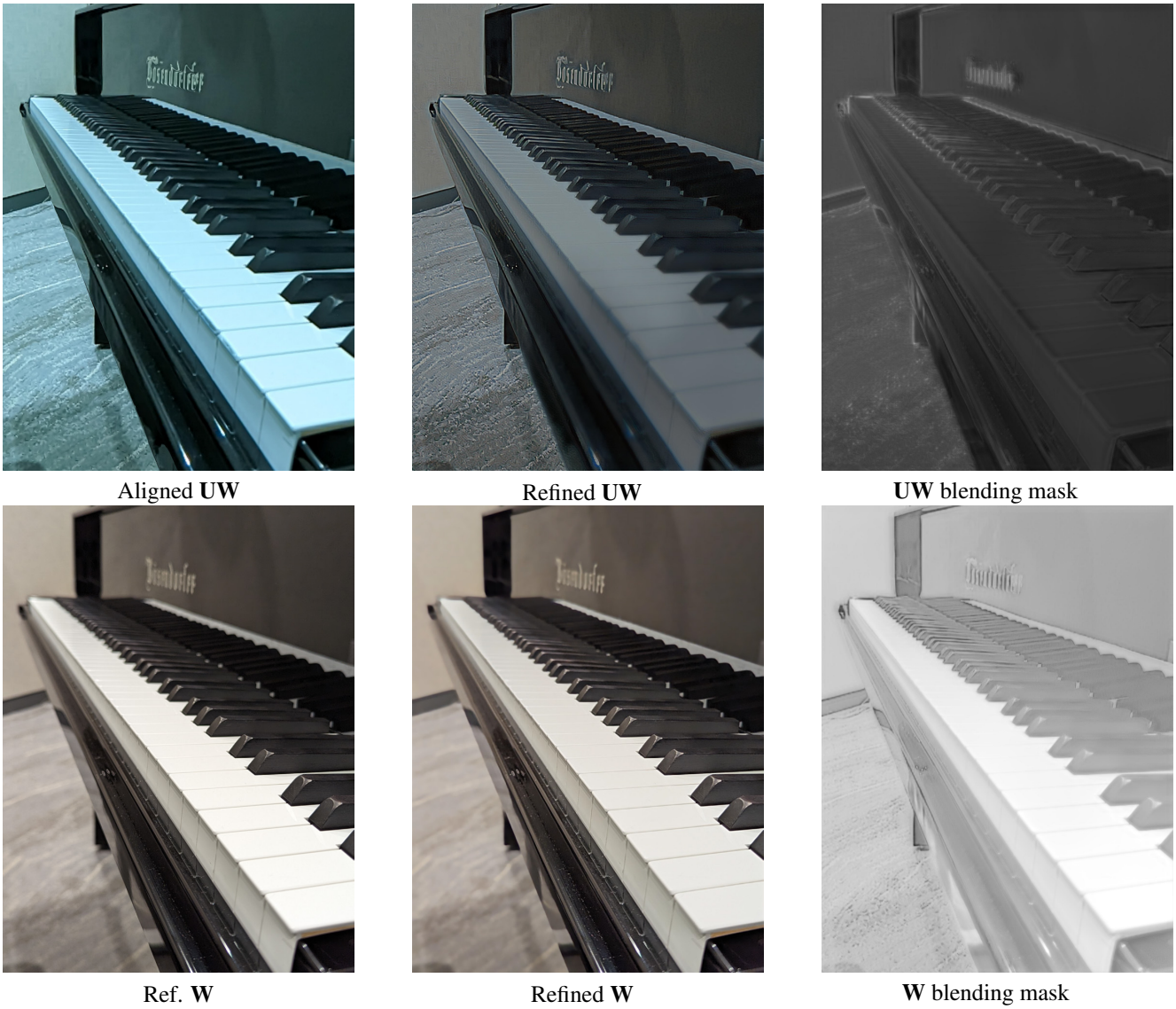
Figure 10. **Disparity from iPhone**. Using portrait mode, we obtained Dual-camera disparity using an iPhone 14 Pro.

C. Model Analysis

The primary motivation behind our architecture design is, depending on the target defocus map, the network can choose to deblur/blur parts of **W** and transfer sharper details from **UW** if necessary. Due to our model design, we can directly visualize the intermediate outputs to understand the model behavior. Specifically, we can visualize the refined **W** and **UW** which are the outputs of Φ^{W}_{ref} and Φ^{UW}_{ref}, as well as the blending masks predicted by the fusion network Φ_{fusion}. We visualize the intermediate outputs of our method on the task of all-in-focus deblurring in Figures 11 and 12. Note that in both examples, the mask associated with **UW** has large values around edges and regions with high frequencies, while the mask for **W** has higher values inlow frequency regions. This supports our hypothesis of having **UW** serve for high frequency details that could be blurry in **W**, while the **W** should be used as a reference to preserve the desired colors even in blurry regions. This behavior makes our method robust to color differences in **W** and **UW** just like we show in Figure 11 where **UW** has incorrect white balance, and in Figure 12 we show how the model avoids relying on **UW** in occluded regions where artifacts may show up in the optical flow alignment.

D. Generalizing to Different Phone Setup

Our method requires only two cameras with different DoFs. This is widely available in modern smartphones since ultra-wide cameras tend to have a deeper DoF due to the small focal length compared to the wide and Telephoto cameras. Our approach that utilizes the defocus map is not specific to a particular device, but rather it can produce fairly good results for any smartphone with a similar **UW+W** dual-camera setup. To use data captured using an iPhone 14 Pro, we used the iPhone's portrait mode to obtain a disparity map (shown in Fig. 10), and warped the **UW** using an opitcal-flow based alignment. In Fig. 13, we show results of our model on data captured by iPhone 14 Pro *without any finetuning*.



Aligned **UW** | Refined **UW** | **UW** blending mask

Ref. **W** | Refined **W** | **W** blending mask

Figure 11. **Intermediate results visualization.** Note that the whitebalance is off in **UW**, but the refinement module does not get affected by that since it primarily preserves the high frequencies in refined **UW**. In the refinement of **W**, we notice that the model deblurs the edges and preserves the low-frequency signals that can be blended with the details from **UW**

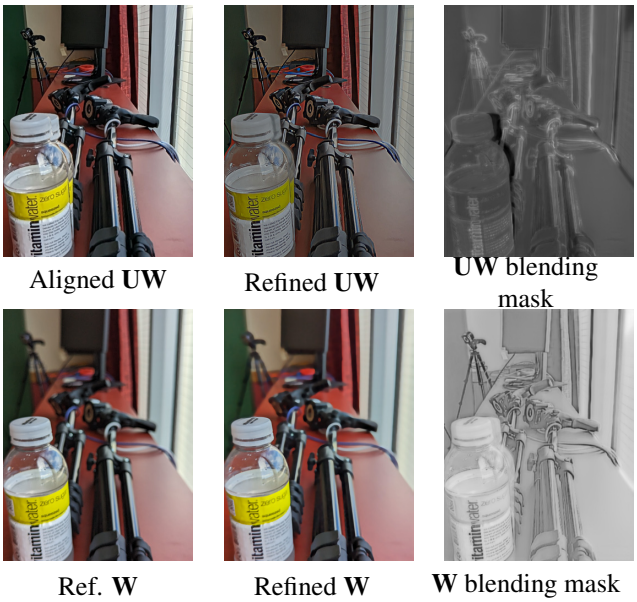


Figure 12. **Intermediate results visualization.** Note that while the aligned **UW** suffers from an alignment artifcat around the bottle, the predicted masks take that into account by setting a low blending value for the occluded region in the **UW** mask and a higher value in the **W** mask.

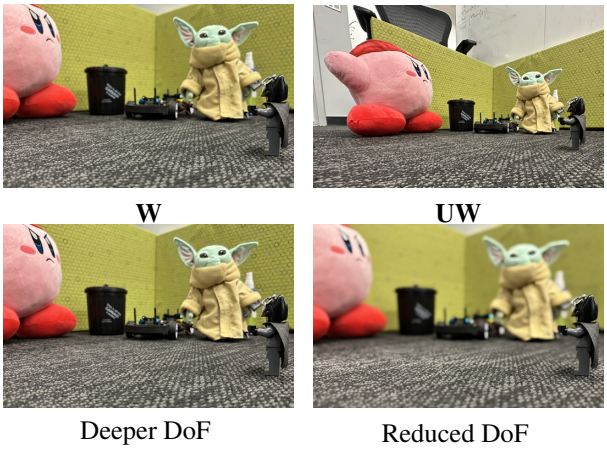


Figure 13. **Results on iPhone 14 Pro**. We ran our model on images from an iPhone 14 Pro, and show that it generalizes with blurring and deblurring despite not finetuning the model on any iPhone data.