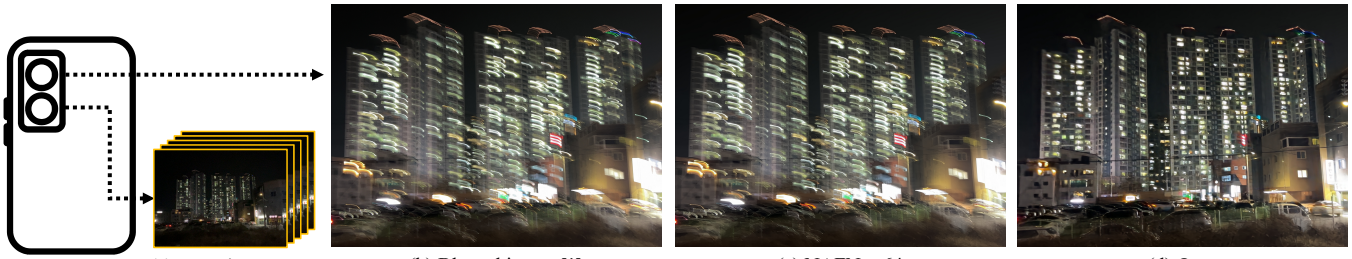
Deep Hybrid Camera Deblurring for Smartphone Cameras

Jaesung Rim
POSTECH
South Korea
jsrim123@postech.ac.kr

Junyong Lee*
Samsung AI Center Toronto
Canada
j.lee8@samsung.com

Heemin Yang
POSTECH
South Korea
heeminid@postech.ac.kr

Sunghyun Cho
POSTECH
South Korea
s.cho@postech.ac.kr



(a) Burst images **U** (1280 × 720) (b) Blurred image W (3840 × 2160) (c) NAFNet-64 (d) Ours

Figure 1: Our hybrid camera system and deblurred results. We simultaneously capture a long-exposure wide image W and short-exposure burst ultra-wide images U from a smartphone, and utilize the burst images to deblur W. Our method produces significantly sharper results compared to NAFNet-64 [Chen et al. 2022], a state-of-the-art single-image deblurring method.

ABSTRACT

Mobile cameras, despite their significant advancements, still have difficulty in low-light imaging due to compact sensors and lenses, leading to longer exposures and motion blur. Traditional blind deconvolution methods and learning-based deblurring methods can be potential solutions to remove blur. However, achieving practical performance still remains a challenge. To address this, we propose a learning-based deblurring framework for smartphones, utilizing wide and ultra-wide cameras as a hybrid camera system. We simultaneously capture a long-exposure wide image and short-exposure burst ultra-wide images, and utilize the burst images to deblur the wide image. To fully exploit burst ultra-wide images, we present HCDeblur, a practical deblurring framework that includes novel deblurring networks, HC-DNet and HC-FNet. HC-DNet utilizes motion information extracted from burst images to deblur a wide image, and HC-FNet leverages burst images as reference images to further enhance a deblurred output. For training and evaluating the proposed method, we introduce the HCBlur dataset, which consists of synthetic and real-world datasets. Our experiments demonstrate that HCDeblur achieves state-of-the-art deblurring quality. Codes and datasets are available at https://cg.postech.ac.kr/research/HCDeblur.

CCS CONCEPTS

• **Computing methodologies** → **Computational photography**.

KEYWORDS

motion deblurring, hybrid camera fusion, mobile imaging, deep neural networks

ACM Reference Format:
Jaesung Rim, Junyong Lee, Heemin Yang, and Sunghyun Cho. 2024. Deep Hybrid Camera Deblurring for Smartphone Cameras. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers '24 (SIGGRAPH Conference Papers '24), July 27-August 1, 2024, Denver, CO, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3641519.3657507

1 INTRODUCTION

While mobile cameras have significantly improved, they still struggle in low-light environments due to their small sensors and lenses, leading to longer exposure time and motion blur from hand movement or moving objects. To remove blur, singe-image deblurring methods [Cho and Lee 2009; Fergus et al. 2006; Pan et al. 2016; Shan et al. 2008; Sun et al. 2013; Xu and Jia 2010; Xu et al. 2013] have been widely studied. Recently, learning-based deblurring methods [Chen et al. 2022, 2021; Cho et al. 2021; Kupyn et al. 2018, 2019; Nah et al. 2017; Tao et al. 2018; Wang et al. 2022; Zamir et al. 2022, 2021] have significantly improved the deblurring performance. Nonetheless,

*Work done prior to joining Samsung AI Center Toronto.

SIGGRAPH Conference Papers '24, July 27-August 1, 2024, Denver, CO, USA

ACM ISBN 979-8-4007-0525-0/24/07.
https://doi.org/10.1145/3641519.3657507

as shown in Fig. 1-(c), the performance of single-image deblurring is still limited, especially for large blurs.

Several methods have been proposed to improve the deblurring performance by exploiting additional inputs, such as a short-exposure image and event data. Reference-based deblurring methods [Chang et al. 2021; Lai et al. 2022; Mustaniemi et al. 2020; Zhao et al. 2022] employ an extra short-exposure image as a reference image to deblur a long-exposure image. These methods align and fuse a blurred image and a reference image to restore a high-quality deblurred image. However, they often struggle to restore sharp details, especially for severely blurred images, due to the challenges of aligning a severely blurred image with a reference image accurately. Event-guided deblurring methods [Cho et al. 2023; Haoyu et al. 2020; Jiang et al. 2020; Kim et al. 2022b; Sun et al. 2022; Xu et al. 2021; Zhang et al. 2023] simultaneously capture a blurred image and event data using a specially designed dual-camera comprising an RGB camera and an event camera, and achieve significant deblurring performance utilizing motion information extracted from event data. However, these methods require an additional event camera, which is not commonly found on most commodity cameras.

In this paper, we propose HCDeblur, a practical image deblurring framework designed for modern smartphones like the Apple iPhone and Samsung Galaxy series, which now commonly feature multi-camera systems. Our approach utilizes such multi-camera systems as a *hybrid camera system*, which consists of a primary camera and a secondary camera with a higher frame rate. The hybrid camera system simultaneously captures a long-exposure image and a burst of short-exposure images. These burst images provide crucial information on pixel-wise camera and object motions during the exposure time, which is challenging to acquire from a single image, while it can significantly enhance the deblurring performance. Additionally, burst images offer high-frequency details that can complement high-frequency details lost in a blurred image. HCDeblur leverages both motion and detail information from burst images and achieves unparalleled performance in image deblurring (Fig. 1-(d)).

The concept of hybrid camera deblurring was first proposed by Ben-Ezra and Nayar [Ben-Ezra and Nayar 2003] to resolve the ill-posedness of image deblurring. Specifically, they propose to use a low-resolution high-speed camera as a secondary camera. Using a low-resolution high-speed camera, they capture a burst of short-exposure images. These images are then used to estimate a uniform motion blur kernel, which is subsequently used for the non-blind deconvolution of a blurred image taken by a primary camera. Tai et al. [2008; 2010] further extended the idea to handle non-uniform deblurring and video deblurring. However, these approaches rely on classical blur models and optimization methods, which severely limits their performance. In contrast, our approach is a learning-based approach that adopts a hybrid camera system. Our approach is tailored for real-world smartphone cameras and shows superior performance in real-world scenarios.

Our framework employs the wide and ultra-wide cameras of a smartphone as the primary and secondary cameras, respectively, as depicted in Fig. 1. The wide camera, typically the main camera in smartphones, captures images at a slow shutter speed. Conversely, the ultra-wide camera, due to its broader field of view (FOV), can gather motion and detail information for the entire region of the image captured by the wide camera. This secondary camera simultaneously captures a burst of low-resolution images at a higher shutter speed and a high frame rate.

Once a wide image and a burst of ultra-wide images are captured, our framework estimates a deblurred image of the wide image with the aid of burst ultra-wide images. To this end, we introduce a deep neural network equipped with two sub-networks: a Hybrid Camera Deblurring Network (HC-DNet) and a Fusion Network (HC-FNet). HC-DNet uses burst images to construct pixel-wise blur kernels and exploits the blur kernels to obtain a deblurred image of the wide image. While HC-DNet produces a deblurred result of superior quality compared to previous single-image deblurring methods thanks to the blur kernels, its results may still contain artifacts and remaining blur due to information loss caused by blur and inaccurate blur kernels. To mitigate this, HC-FNet, which is inspired by burst imaging techniques [Bhat et al. 2021; Dudhane et al. 2022; Mehta et al. 2023], refines the output of HC-DNet using the entire sequence of the burst images as reference images.

For training and evaluating our method, we also present the HCBlur dataset, which consists of two sub-datasets: HCBlur-Syn and HCBlur-Real. HCBlur-Syn is a synthetically generated dataset for training and evaluation, and comprises 8,568 pairs of blurred wide images and ground-truth sharp images with corresponding sharp ultra-wide burst images. On the other hand, HCBlur-Real is a real dataset for evaluation, and provides 471 real-world pairs of a blurred wide image and burst ultra-wide images without ground-truth sharp images. Our experimental results using the HCBlur dataset show that our method significantly outperforms state-of-the-art deblurring methods.

To summarize, our contributions include:

- HCDebur, a learning-based hybrid camera deblurring framework specifically designed for smartphone cameras,
- HC-DNet and HC-FNet, which utilize burst ultra-wide images for deblurring and refining an input wide image, respectively, and
- the HCBlur dataset for training and evaluating hybrid camera deblurring methods.

2 RELATED WORK

Single-image Deblurring. Classical blind deconvolution methods [Cho and Lee 2009, 2017; Levin et al. 2009; Pan et al. 2016; Shan et al. 2008; Xu and Jia 2010] alternatingly estimate a blur kernel and a sharp image to deblur a blurred image. Their performance is limited due to restrictive blur models and the ill-posedness of the deblurring problem. Recently, learning-based methods have significantly improved the deblurring performance by learning the deblurring process from the training dataset. Numerous deblurring networks have been proposed, such as multi-scale [Cho et al. 2021; Nah et al. 2017; Tao et al. 2018], multi-stage [Zamir et al. 2021; Zhang et al. 2019], multi-scale and multi-stage [Kim et al. 2022a], NAFNet [Chen et al. 2022], GAN-based [Kupyn et al. 2018, 2019], and transformer-based [Tu et al. 2022; Wang et al. 2022; Zamir et al. 2022] methods. However, learning-based methods still struggle with the generalization problem due to the difficulty of learning arbitrary possible motion blur [Fang et al. 2023; Kaufman and Fattal 2020].

Kernel-based Deblurring. Kernel-based deblurring methods estimate blur kernels from a blurred image and use them in deblurring networks to improve the generalization ability and performance. Kaufman et al. [2020] incorporate a kernel estimation network and a deblurring network for uniform motion. Zhang et al. [2021] estimate non-uniform motion offsets from a blurred image and utilize them as offsets of deformable convolution [Zhu et al. 2019] in a deblurring network. Fang et al. [2023] proposed a non-uniform kernel estimation network based on the normalizing flow model and utilize estimated blur kernels for computing channel attention. However, estimating accurate blur kernels from a single blurred image is still challenging due to the ill-posedness of the deblurring problem.

Reference-based Deblurring. Reference-based deblurring methods utilize an extra short-exposure image as a reference image. Mustaniemi et al. [2020] proposed LSD_2 for jointly deblurring and denoising. LSFNet [Chang et al. 2021] and D2HNet [Zhao et al. 2022] employ deformable convolution [Zhu et al. 2019] to align and incorporate features of a blurred image and a short exposure image. Several methods for smartphone cameras utilize extra ultra-wide images as additional inputs. Lai et al. [2022] proposed a face deblurring method that leverages an ultra-wide short-exposure image to deblur a blurred wide image. Alzayer et al. [2023] proposed a defocus blur control method utilizing an extra ultra-wide image. Our approach differs from theirs by utilizing burst reference images for deblurring, not a single reference image.

Hybrid Camera Deblurring. Ben-Ezra and Nayar [2003] first introduced the idea of hybrid camera deblurring. Tai et al. [2008; 2010] extended the idea to non-uniform deblurring and video deblurring. Li et al. [2008] proposed a method of hybrid imaging for motion deblurring and depth map super-resolution. Concurrent to our work, Shekarforoush et al. [2023] recently introduced a novel framework that also utilizes a single long-exposure image and a burst of short-exposure images to reconstruct a noise-free sharp image. To this end, their framework integrates a kernel-based deblurring [Zhang et al. 2021] and a burst denoising network [Bhat et al. 2021]. While their work also uses burst images as ours, ours is specifically tailored for real-world smartphone cameras, addresses misalignment between a long-exposure image and burst images, and presents realistic datasets.

Burst Image Enhancement. Burst image enhancement is a task to reconstruct a high-quality image from a burst of images. Numerous approaches have been proposed for burst super-resolution, denoising and deblurring such as DBSR [Bhat et al. 2021], BIPNet [Dudhane et al. 2022], Burstormer [Dudhane et al. 2023], GMTNet [Mehta et al. 2023], Fourier burst accumulation [Delbracio and Sapiro 2015], and a permutation-invariant network [Aittala and Durand 2018]. While our work also employs burst images, it differs from these burst image enhancement approaches as our goal is to deblur a long-exposure wide image by utilizing extra short-exposure burst images.

3 DEEP HYBRID CAMERA DEBLURRING

Fig. 2 shows an overview of HCDeblur. HCDeblur takes a long-exposure wide image W and a burst of short-exposure ultra-wide images $\mathbf{U} = \{U_1, \cdots, U_N\}$ captured by our hybrid camera system (Sec. 3.1) as input, where U_i is the i-th ultra-wide image, and N is the number of ultra-wide images. The wide and ultra-wide cameras have different FOVs and optical centers, resulting in geometric misalignment between them. Thus, to obtain blur kernels from $\mathbf{U}$ that are aligned to W, we align W and $\mathbf{U}$ using the FOV alignment (Sec. 3.2). We also compute pixel-wise motion trajectories across $\mathbf{U}$ (Sec. 3.2) for constructing blur kernels. Then, we perform kernel-based deblurring in HC-DNet (Sec. 3.3) using the blur kernels constructed from the motion trajectories and obtain a deblurred image W_D. We further refine W_D using the detail information in $\mathbf{U}$ using HC-FNet (Sec. 3.4) and obtain a final result W_F. In the following, we describe each component of HCDeblur in more detail.

3.1 Hybrid Camera System

For the hybrid camera system, we employ the wide and ultra-wide cameras of an Apple iPhone 13 Pro Max in our experiments, whose focal lengths are 26mm and 13mm, respectively. To simultaneously capture W and $\mathbf{U}$, we developed an iOS app using the multi-camera API [Apple 2024]. Our app captures a wide image at 4K resolution (2160×3840) with exposure times ranging from 1/15 to 1/2 sec.. At the same time, our app captures a burst of ultra-wide images of a resolution of 720P (720×1280) at 60 frames per sec. (FPS). As wide images have a twice longer focal length, and 3× more pixels along the horizontal and vertical axes than ultra-wide images, the objects in wide images appear 6× larger than those in ultra-wide images. To prevent blur in burst images, we limit the maximum exposure time of the ultra-wide camera to 1/120 sec.. Due to hardware limitations, the exposure times of cameras are not perfectly synchronized. Thus, for synchronizing images from the two cameras, we also record the timestamps of the beginning and end of the exposure of both wide and ultra-wide cameras.

3.2 FOV Alignment and Motion Estimation

FOV Alignment. Once a wide image W and a burst of ultra-wide images $\mathbf{U}$ are captured, we first align $\mathbf{U}$ to W in the FOV alignment step. To this end, we find a single homography that best aligns W and $\mathbf{U}$ using the plane sweep method [Collins 1996]. Specifically, we first find the best depth $\hat{d}$ by solving:

$$\hat{d} = \underset{d \in D}{\operatorname{argmin}}\, MSE(W, \mathbb{W}(U_{\text{avg}}, H_d)) \tag{1}$$

where D is a set of depth candidates, MSE indicates mean-squared-error, and $\mathbb{W}$ is a warping function. U_{avg} is the average image of $\mathbf{U}$. We use U_{avg} to take the blur in W into account. H_d is a homography for inverse warping, which is defined as $H_d = K_u E d K_w^{-1}$, where K_u and K_w are intrinsic matrices of the ultra-wide and wide cameras, respectively, and E is a relative extrinsic matrix. K_w, K_u and E are obtained by stereo calibration. Once $\hat{d}$ is found, we compute its homography $H_{\hat{d}}$ for aligning $\mathbf{U}$, which we denote by $\hat{H}$.

Note that alignment between sharp images and a blurred image is a challenging task, especially when the blur is large, and this limits the performance of previous reference-based deblurring methods. In contrast, our approach enables more accurate alignment as we can synthetically generate U_{avg} from a burst of ultra-wide reference images. It is also important to note that our FOV alignment, which uses a single homography to align W and $\mathbf{U}$, may result in

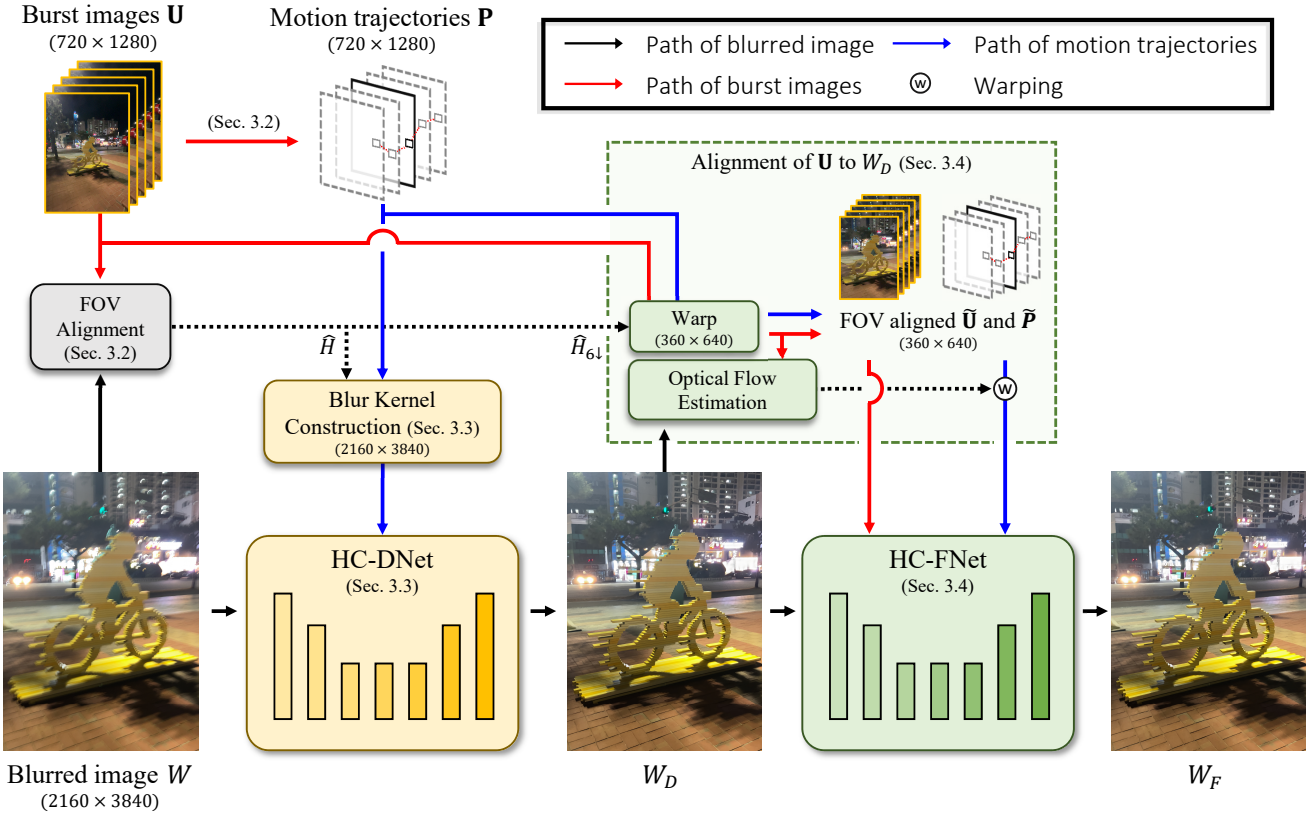
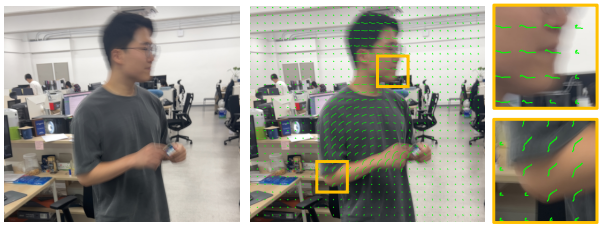


Figure 2: Overview of HCDeblur. Our framework takes a long-exposure wide image W and a burst of short-exposure ultra-wide images U as inputs. We estimate a homography matrix $\hat{H}$ for aligning U in the FOV alignment (Sec. 3.2) and compute pixel-wise motion trajectories P (Sec. 3.2). HC-DNet performs kernel-based deblurring by exploiting blur kernels K constructed from P (Sec. 3.3). After deblurring, an additional alignment step is adopted to align the burst images to the deblurred wide image W_D. HC-FNet further enhances the deblurred image by using the entire sequence of the burst images as reference images (Sec. 3.4).

some residual misalignment. However, our method is designed to be robust against such issues. This is because HC-DNet employs a homography not for directly transferring details from **U** to W, but for aligning blur kernels. These kernels typically vary smoothly in space, making them less sensitive to minor misalignments. Furthermore, once HC-DNet produces a deblurred image W_D, we can achieve precise alignment between W_D and **U** by estimating the optical flow between them, and use them for HC-FNet.

Pixel-wise Motion Trajectories Estimation. Alongside the FOV alignment, we also estimate pixel-wise motion trajectories across the ultra-wide images **U**, which will be used for constructing blur kernels and for the alignment of the ultra-wide images for HC-FNet. To this end, we estimate optical flows between adjacent images in **U**. Specifically, we denote by c the index of the temporally center image in **U**. Then, we estimate the optical flow from U_{i+1} to U_i for $i < c$, and estimate the optical flow from U_{i-1} to U_i for $i > c$. As a result, we obtain $N-1$ optical flow maps. From the estimated optical flow maps, we construct pixel-wise motion trajectories $\mathbf{P} = \{P_1, \cdots, P_N\}$ by accumulating the optical flow maps. Specifically, at each pixel position, **P** contains a motion trajectory consisting of N displacement vectors where the i-th vector is a 2D displacement vector from the center image to the i-th image. For



(a) Blurred image W (b) Visualization of **K**

Figure 3: Blur kernels constructed from pixel-wise motion trajectories.

optical flow estimation, we adopt RAFT [Teed and Deng 2020] in our experiments.

3.3 Kernel-based Deblurring using HC-DNet

Blur Kernel Construction. To deblur W using HC-DNet, we construct blur kernels using the motion trajectories estimated from **U**. To this end, we first warp the motion trajectories using the homography $\hat{H}$ and obtain warped motion trajectories $\hat{\mathbf{P}} = \{\hat{P}_1, \cdots, \hat{P}_N\}$. While the length of motion trajectories, N, varies as it is determined by the exposure time, HC-DNet takes input tensors of fixed channel

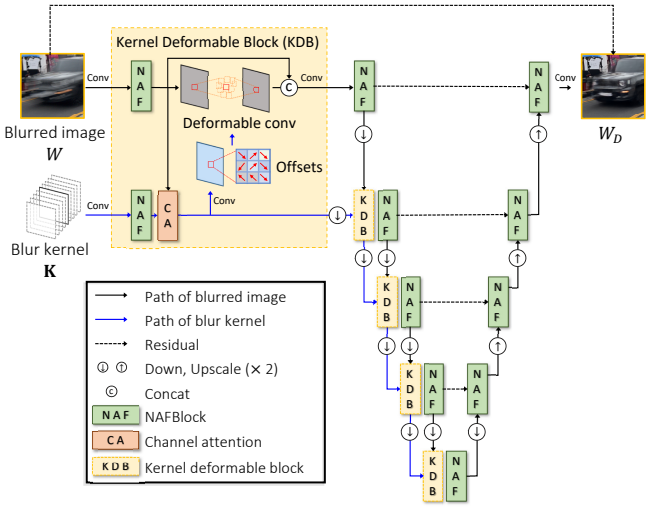


Figure 4: Architecture of HC-DNet.

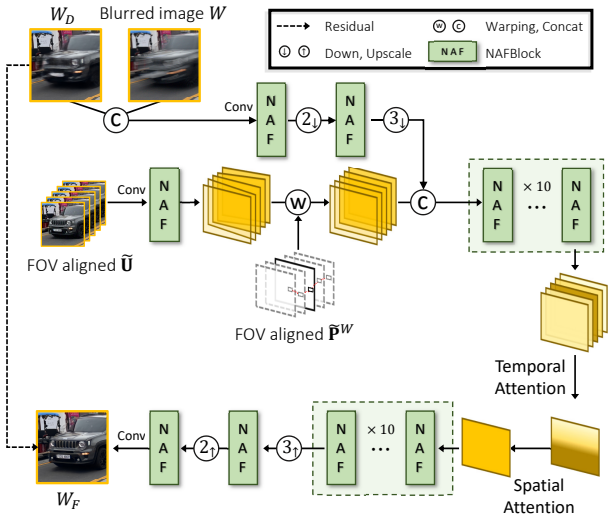


Figure 5: Architecture of HC-FNet.

sizes. Thus, we construct blur kernels of a fixed length from the motion trajectories by resampling them.

To obtain blur kernels accurately synchronized to the exposure time of the wide image W, we resample the motion trajectories considering the timestamps of the wide and ultra-wide images. Specifically, we denote the timestamps of the beginning and end of the exposure of the i-th ultra-wide image as $t_{i,s}$ and $t_{i,e}$. We first compute the relative timestamp r_i as:

$$r_i = \frac{(t_{i,s} + t_{i,e})/2 - t_s^W}{t_s^W - t_e^W} \tag{2}$$

where t_s^W and t_e^W represent the timestamps of the beginning and end of the exposure of W, respectively. We then construct the blur kernel $\mathbf{K} = \{K_0, K_{0.125}, \cdots, K_{0.875}, K_1\}$ by interpolating the motion trajectories for the pre-defined set of nine timestamps. Formally, K_t is computed as:

$$K_t = (t - r_i) \cdot \frac{\hat{P}_{i+1} - \hat{P}_i}{r_{i+1} - r_i} + \hat{P}_i, \tag{3}$$

where $t \in \{0, 0.125, \ldots, 0.875, 1\}$, r_i and r_{i+1} are the nearest relative timestamps of $\mathbf{U}$ with respect to each t, and $\hat{P}_i$ and $\hat{P}_{i+1}$ are the displacement vector maps corresponding to r_i and r_{i+1}, respectively. Finally, we concatenate K_t for all t and obtain $\mathbf{K}$ of a channel size 18. Fig. 3 shows an example of estimated blur kernels.

HC-DNet. HC-DNet takes a wide image W and pixel-wise blur kernels $\mathbf{K}$ as inputs and produces a deblurred wide image W_D (Fig. 4). The network adopts a U-Net architecture, in which we compose each level of the encoder and decoder networks with NAFBlocks [Chen et al. 2022]. To exploit $\mathbf{K}$, we design a novel kernel deformable block (KDB), which we attach before each level of the encoder. In KDB, the features of W and $\mathbf{K}$ are initially extracted by NAFBlocks. Channel attentions [Hu et al. 2018] are then computed from the features of W and $\mathbf{K}$ that are attended on features of blur kernels, which is to prevent utilization of potentially inaccurate blur kernels. Subsequently, the attended features of $\mathbf{K}$ are used for predicting offsets and weights of a deformable convolution layer [Zhu et al. 2019], which adaptively handles the features of W according to $\mathbf{K}$, leading to more effective utilization of $\mathbf{K}$ in the deblurring process. Then, the feature of W and the output from the deformable convolution layer are concatenated and subsequently processed through a convolutional layer.

3.4 Burst Image-based Refinement using HC-FNet

Alignment of $\mathbf{U}$ *to* W_D. For the fusion of $\mathbf{U}$ and W_D, we align $\mathbf{U}$ to the 6× downsampled version of W_D, which is denoted by $W_{D,6\downarrow}$. To this end, we warp both $\mathbf{U}$ and $\mathbf{P}$ using a 6× downscaled version of $\hat{H}$, which is denoted as $\hat{H}_{6\downarrow}$, and obtain FOV-aligned ultra-wide images $\tilde{\mathbf{U}} = \{\tilde{U}_1, \cdots, \tilde{U}_N\}$ and motion trajectories $\tilde{\mathbf{P}} = \{\tilde{P}_1, \cdots, \tilde{P}_N\}$. Note that $\tilde{\mathbf{U}}$ are not downsampled but only aligned to match the FOV of the wide image. Then, for more accurate alignment, we estimate the optical flow $\tilde{F}$ between $W_{D,6\downarrow}$ and the center frame $\tilde{U}_c$ of $\tilde{\mathbf{U}}$. Finally, we compute the optical flow $\tilde{P}_i^W$ from $W_{D,6\downarrow}$ to the i-th FOV aligned ultra-wide image $\tilde{U}_i$ by combining $\tilde{P}_i$ and $\tilde{F}$.

HC-FNet. Fig. 5 shows the overall architecture of the HC-FNet, which is based on the architecture of a burst enhancement network [Bhat et al. 2021] and NAFBlock [Chen et al. 2022]. HC-FNet takes $\tilde{\mathbf{U}}$, and the deblurred wide image W_D as input, and produces a refined wide image W_F. HC-FNet also takes W as additional input to obtain information that might be missing from W_D. Then, HC-FNet extracts features from each $\tilde{U}_i$ using a NAFBlock and aligns the feature map to W_D using $\tilde{P}_i^W$. After the alignment, the features of W_D and W are concatenated to the feature map of each FOV-aligned ultra-wide image $\tilde{U}_i$. These combined features of each ultra-wide image are then processed through NAFBlocks in a frame-wise manner. We then merge the resulting features using the temporal and spatial attention module (TSA) [Wang et al. 2019] into a single frame. Finally, the merged features are further processed and upsampled by additional NAFBlocks and upsampling layers.

Table 1: Quantitative comparisons on the HCBlur-Syn and HCBlur-Real datasets. For the HCBlur-Real dataset, we use no-reference metrics (i.e., NIQE [Mittal et al. 2012b], BRISQUE [Mittal et al. 2012a], and TOPIQ [Chen et al. 2024] trained on SPAQ [Fang et al. 2020]) for evaluation. The number of parameters, MACs, and inference times of the kernel-based deblurring methods and HCDeblur include those for the optical flow estimation when the burst size of U is eight. HCDeblur$_{small}$ indicates a lightweight version of HCDeblur, which uses a smaller version of RAFT [Teed and Deng 2020].

	HCBlur-Syn	HCBlur-Real			
	PSNR ↑ / SSIM ↑	NIQE ↓ / BRISQUE ↓ / TOPIQ ↑	Params. (M)	MACs (G)	Time (Sec.)
MIMO-UNet+ [Cho et al. 2021]	22.42 / 0.5966	5.49 / 27.87 / 0.44	16.11	19563	4.52
Uformer-B [Wang et al. 2022]	23.31 / 0.6332	5.17 / 29.63 / 0.55	50.88	12483	15.11
HINet [Chen et al. 2021]	23.60 / 0.6360	5.42 / 29.74 / 0.47	88.67	21606	3.28
NAFNet-32 [Chen et al. 2022]	23.61 / 0.6364	5.05 / 29.50 / 0.53	17.11	2038	1.32
NAFNet-64 [Chen et al. 2022]	24.18 / 0.6607	4.83 / 28.09 / 0.55	67.89	8018	3.18
LSFNet [Chang et al. 2021]	22.32 / 0.6058	4.87 / 30.80 / 0.53	10.20	4844	1.36
D2HNet [Zhao et al. 2022]	22.64 / 0.6125	4.83 / 31.56 / 0.56	79.58	9871	2.51
NAFNet-Ref	24.37 / 0.6585	4.56 / 29.81 / 0.60	18.14	2316	1.73
UFPNet [Fang et al. 2023]	24.44 / 0.6701	4.79 / 28.14 / 0.56	78.37	20105	6.30
MotionETR [Zhang et al. 2021]	25.24 / 0.6951	4.52 / 26.00 / 0.63	10.68	11439	4.68
HCDeblur$_{small}$ (only HC-DNet)	25.80 / 0.7144	4.14 / 25.66 / 0.67	10.62	2186	1.71
HCDeblur$_{small}$	26.50 / 0.7281	3.97 / 24.79 / 0.69	12.69	3286	3.16
HCDeblur (only HC-DNet)	26.13 / 0.7251	4.11 / 25.51 / 0.67	14.89	5189	2.35
HCDeblur	26.76 / 0.7373	3.95 / 24.65 / 0.69	16.95	6288	3.85

4 HCBLUR DATASET

In this section, we introduce HCBlur-Syn and HCBlur-Real datasets. To construct the HCBlur-Syn dataset, we simultaneously capture wide and ultra-wide sharp videos and synthesize blurred wide images by averaging wide video frames. We developed an iOS app to simultaneously capture wide and ultra-wide videos. We used wide and ultra-wide cameras of an iPhone 13 Pro Max as our camera system. The ultra-wide camera captures videos of 720P resolution (720×1280) at 60 FPS. Simultaneously, the wide camera captures videos of 4K resolution (2160×3840) at 30 FPS, which is the maximum frame rate supported by the multi-camera API [Apple 2024]. Naïvely averaging wide video frames of 30 FPS may result in discontinuous blur due to significant motion between frames. To address this problem, we estimate optical flows between consecutive frames of wide videos and exclude the frames with a maximum displacement greater than 36 pixels. Then, blurred wide images W are generated by averaging the remaining frames of wide videos, and their temporal-center frames are chosen as ground-truth images. The number of averaged frames is randomly sampled from the set of $\{5, 7, 9, 11, 13\}$.

Then, for each blurred wide image W, ultra-wide images corresponding to W are extracted from the ultra-wide videos using their timestamps. Afterward, we randomly select a subset of these extracted burst images, ranging from 5 to 14 frames, to maintain a reasonable number of burst images. To simulate noise on images, we synthesize Poisson-Gaussian noise on W and $\mathbf{U}$ using shot and read noise parameters calibrated from our camera system. Additionally, we adopt a realistic blur synthesis pipeline of Rim et al. [2022] to improve the generalization ability of HCBlur-Syn to real-world images. This pipeline includes frame interpolation and saturated pixels synthesis. More details on synthesizing the HCBlur-Syn dataset can be found in the supplementary material.

Through the process described above, we collected 1,176 pairs of wide and ultra-wide videos of various indoor, daytime, and nighttime scenes, and synthesized 8,568 pairs of a wide blurred image and a sequence of ultra-wide images for constructing the HCBlur-Syn dataset. For training and evaluating deblurring methods, we split the HCBlur-Syn dataset into train, validation, and test sets, each of which consists of 5,795, 880, and 1,731 pairs, respectively.

The HCBlur-Real dataset provides 471 pairs of real-world W and $\mathbf{U}$ from 157 of various night and indoor scenes. To collect the dataset, we simultaneously captured wide images and ultra-wide images using the iOS app for hybrid camera capturing (Sec. 3.1). Since ground-truth images are not included in the dataset, we utilize the HCBlur-Real dataset for evaluation using non-reference metrics [Chen et al. 2024; Mittal et al. 2012a,b].

5 EXPERIMENTS

Implementation Details. We utilize AdamW optimizer [Kingma and Ba 2014] and PSNR loss [Chen et al. 2022] for training our method. The initial learning rate is set to 1e-3 and is gradually reduced to 1e-7 using the cosine annealing strategy [Loshchilov and Hutter 2016]. We train our network in two steps; we first train HC-DNet and then train HC-FNet using the deblurred results of HC-DNet. HC-DNet is trained for 300,000 iterations with a batch size of 32 and HC-FNet is trained for 150,000 iterations with a batch size of 8. During the optimization of HC-FNet, the optical flow network [Teed and Deng 2020] for the alignment of $\mathbf{U}$ to W_D is also jointly optimized. The patch size for training is set to 384 for both HC-DNet and HC-FNet.

5.1 Comparison with State-of-the-Art Methods

To verify the effectiveness of HCDeblur, we evaluate HCDeblur and other deblurring methods on the HCBlur-Syn and HCBlur-Real datasets. We compare HCDeblur with various deblurring methods including single-image, reference-based, and explicit blur kernel-based deblurring methods. Single-image deblurring methods in our evaluation include CNN-based [Chen et al. 2022, 2021; Cho et al. 2021] and transformer-based [Wang et al. 2022] methods. The transformer-based method [Wang et al. 2022] is evaluated using patch-wise testing due to the memory issue on 4K-resolution images.

Regarding reference-based deblurring methods, we include LSFNet [Chang et al. 2021] and D2HNet [Zhao et al. 2022], both of which exploit a short-exposure image for deblurring a long-exposure image. As both models assume short-exposure images of the same spatial size and zoom ratio as those of the long-exposure image, we align and upscale the temporal center of ultra-wide images, and feed it to them as the short-exposure input. Most single-image deblurring networks can be easily extended to take an additional reference image. Thus, to cover such a case in our comparison, we also include a variant of NAFNet-32 [Chen et al. 2022] that is modified for reference-based deblurring, which is denoted by NAFNet-Ref. Specifically, NAFNet-Ref aligns the temporal center of ultra-wide images to a blurred wide image using RAFT [Teed and Deng 2020], and both wide and aligned ultra-wide images are fed to a NAFNet-32. For evaluation, we train both RAFT and NAFNet-32 in an end-to-end manner. More details of NAFNet-Ref are provided in the supplementary material.

Finally, as HC-DNet leverages blur kernels, we also include recent explicit blur kernel-based methods: MotionETR [Zhang et al. 2021] and UFPNet [Fang et al. 2023] in our evaluation. Both methods estimate blur kernels using single-image blur kernel estimation networks and use the blur kernels in deblurring networks. We modified them to use blur kernels estimated from ultra-wide images instead of their kernel estimation networks. All the compared methods are trained on the HCBlur-Syn dataset for fair comparisons.

Table 1 shows a quantitative comparison of HCDeblur and the other methods on the HCBlur-Syn and HCBlur-Real datasets. HCDeblur exhibits superior performance compared to all the other methods on both datasets. Notably, HC-DNet alone still outperforms all the competitors. Our final model achieves the best performance as HC-FNet further refines the deblurred results using the ultra-wide burst images. We also compare a smaller version of HCDeblur: HCDeblur$_{small}$, which uses a small version of RAFT [Teed and Deng 2020] for constructing motion trajectories. As the table shows, HCDeblur$_{small}$ still outperforms all the other methods. This result clearly proves the advantage of utilizing ultra-wide burst images.

Interestingly, the reference-based deblurring methods do not significantly outperform the single-image deblurring methods despite using additional reference images. This is because our hybrid camera system captures low-resolution ultra-wide images while these methods require high-resolution reference images, and also because they are sensitive to the alignment errors between input and reference images. A detailed analysis of the sensitivity of reference-based deblurring on the alignment error is provided in the supplementary material. While NAFNet-Ref performs the best among the single-image and reference-based deblurring methods, it still performs worse than ours.

Thanks to the motion information in the ultra-wide burst images, the explicit blur kernel-based methods, UFPNet [Fang et al. 2023] and MotionETR [Zhang et al. 2021], outperform the single-image and reference-based deblurring methods. Nevertheless, HCDeblur still outperforms both UFPNet and MotionETR thanks to its more carefully designed network architecture of HC-DNet, and the refinement process by HC-FNet.

Figs. 6 and 7 show qualitative evaluations on the HCBlur-Syn and HCBlur-Real datasets, respectively. As the figures show, HCDeblur successfully deblurs 4K images, restoring sharp details of small texts and edges. In contrast, NAFNet-64 [Chen et al. 2022] and NAFNet-Ref fail to restore sharp details. MotionETR [Zhang et al. 2021] sometimes fails to restore sharp details because of its sensitivity to inaccuracies in blur kernels. It uses the trajectories of blur kernels as fixed offsets in deformable convolution [Zhu et al. 2019], making it vulnerable to inaccurate blur kernels. In contrast, HC-DNet effectively addresses errors in blur kernels by learning the deformable convolution offsets from blur kernels, resulting in much sharper deblurred images. Our final model further enhances deblurred images using HC-FNet and achieves the sharpest results.

Table 2: Ablation study for variants of HCDeblur. We compare its performance without HC-DNet and with NAFNet-32 [Chen et al. 2022], evaluate different alignment modules (RAFT [Teed and Deng 2020] vs BIPNet [Dudhane et al. 2022]) after the FOV alignment, and examine burst features fusion strategies like averaging (Avg.), transposed attention (Trans Att.) [Mehta et al. 2023], and temporal and spatial attention (TSA) [Wang et al. 2019].

DeblurNet	Alignment	FusionNet	PSNR ↑ / SSIM ↑
×	FOV + RAFT	HC-FNet + TSA	23.22 / 0.5963
NAFNet-32	FOV + RAFT	HC-FNet + TSA	24.05 / 0.6493
HC-DNet	FOV	HC-FNet + TSA	26.63 / 0.7369
HC-DNet	FOV + BIPNet	HC-FNet + TSA	26.66 / 0.7370
HC-DNet	FOV + RAFT	HC-FNet + Avg.	26.66 / 0.7379
HC-DNet	FOV + RAFT	HC-FNet + Trans Att.	26.70 / 0.7367
HC-DNet	FOV + RAFT	HC-FNet + TSA	26.76 / 0.7373

5.2 Ablation Studies

Table 2 shows the effects of different components of HCDeblur. To evaluate the effect of HC-DNet, we also prepare two variants of HCDeblur: one without HC-DNet and the other with NAFNet-32 [Chen et al. 2022], which uses an input blurred image W and the deblurred result of NAFNet-32 instead of W_D, respectively. Both methods significantly underperform the original HCDeblur model, validating the importance of HC-DNet. We also compare different alignment strategies for HC-FNet in Table 2. In the table, FOV means FOV alignment, and BIPNet indicates the alignment module proposed by Dudhane et al. [2022] that implicitly aligns features using deformable convolution. Again, among all the alignment strategies, 'FOV + RAFT' performs the best thanks to the high-quality optical flow maps estimated from a deblurred output W_D

Table 3: Comparison of different schemes for exploiting blur kernels.

Methods	PSNR ↑ / SSIM ↑
HC-DNet	23.18 / 0.6184
+ KAM [Fang et al. 2023]	25.35 / 0.6909
+ Concat	25.39 / 0.6920
+ KGC [Kaufman and Fattal 2020]	25.45 / 0.6953
+ MAB [Zhang et al. 2021]	25.52 / 0.7004
+ KDB w/o Channel Attention	26.05 / 0.7229
+ KDB	26.13 / 0.7251

and **U**. Finally, we also examine different feature fusion strategies for HC-FNet. As the table shows, temporal and spatial attention (TSA) [Wang et al. 2019] achieves better performance than the other fusion strategies.

Kernel Deformable Block. In HC-DNet, KDB is a crucial component for exploiting blur kernels. Here, we verify the effectiveness of the component by comparing it with other approaches: simple concatenation, the kernel guided convolution (KGC) [Kaufman and Fattal 2020], the kernel attention module (KAM) [Fang et al. 2023], and the motion aware block (MAB) [Zhang et al. 2021]. The simple concatenation indicates that features of blur kernels and a blurred image are concatenated and merged by a CNN. KGC utilizes blur kernels to linearly transform features of a blurred image, while KAM employs blur kernels to determine the weights of pixel-wise channel attention. MAB directly uses the trajectories of blur kernels as fixed offsets of deformable convolution [Zhu et al. 2019]. In contrast, KDB learns to estimate the offsets of deformable convolution kernels from the features of blur kernels. We applied each approach to the encoder of HC-DNet instead of KDB. Table 3 shows KDB achieves the highest PSNR and SSIM scores, proving the effectiveness of KDB.

6 CONCLUSION

In this paper, we proposed HCDeblur, a novel learning-based hybrid camera deblurring framework for smartphone cameras. Our framework adopts HC-DNet and HC-FNet, which utilize burst ultra-wide images for blur kernel estimation and refinement of a deblurred image, respectively. Additionally, we have also presented the HCBlur dataset for training and evaluation of the proposed method. Our experiments demonstrate the effectiveness of the proposed method.

Limitations. Our framework uses a large optical flow network, RAFT [Teed and Deng 2020], and computes optical flow across burst images, increasing the computational overhead. The inference time of our system is about four sec., which is too slow to serve as an on-device application on smartphones. However, the optical flow network can be replaced by any efficient network, and HCDeblur could be further optimized using acceleration techniques. Another limitation is that, as our method relies on optical flow, its performance may degrade when optical flow estimation fails, e.g., it may fail for extremely noisy or textureless images. To resolve these issues would be interesting future directions.

ACKNOWLEDGMENTS

This work was supported by Samsung Research Funding & Incubation Center of Samsung Electronics under Project Number SRFC-IT1801-52 and the National Research Foundation of Korea (NRF) grants (No.2023R1A2C200494611) funded by the Korea government (MSIT) and Institute of Information & communications Technology Planning & Evaluation (IITP) grants (No.2019-0-01906, Artificial Intelligence Graduate School Program (POSTECH)) funded by the Korea government (MSIT).

REFERENCES

Miika Aittala and Frédo Durand. 2018. Burst image deblurring using permutation invariant convolutional neural networks. In *Proc. of ECCV*.

Hadi Alzayer, Abdullah Abuolaim, Leung Chun Chan, Yang Yang, Ying Chen Lou, Jia-Bin Huang, and Abhishek Kar. 2023. DC2: Dual-Camera Defocus Control by Learning To Refocus. In *Proc. of CVPR*.

Apple. 2024. AVCaptureMultiCamSession. https://developer.apple.com/documentation/avfoundation/avcapturemulticamsession.

M. Ben-Ezra and S.K. Nayar. 2003. Motion deblurring using hybrid imaging. In *Proc. of CVPR*.

Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. 2021. Deep burst super-resolution. In *Proc. of CVPR*.

Meng Chang, Huajun Feng, Zhihai Xu, and Qi Li. 2021. Low-light Image Restoration with Short-and Long-exposure Raw Pairs. *IEEE Trans. Multimedia* (2021).

Chaofeng Chen, Jiadi Mo, Jingwen Hou, Haoning Wu, Liang Liao, Wenxiu Sun, Qiong Yan, and Weisi Lin. 2024. Topiq: A top-down approach from semantics to distortions for image quality assessment. *IEEE Trans. Image Process.* (2024).

Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. 2022. Simple baselines for image restoration. In *Proc. of ECCV*.

Liangyu Chen, Xin Lu, Jie Zhang, Xiaojie Chu, and Chengpeng Chen. 2021. HINet: Half Instance Normalization Network for Image Restoration. In *Proc. of CVPRW*.

Hoonhee Cho, Yuhwan Jeong, Taewoo Kim, and Kuk-Jin Yoon. 2023. Non-Coaxial Event-guided Motion Deblurring with Spatial Alignment. In *Proc. of ICCV*.

Sunghyun Cho and Seungyong Lee. 2009. Fast Motion Deblurring. *ACM Trans. Graph.* (2009).

Sunghyun Cho and Seungyong Lee. 2017. Convergence Analysis of MAP Based Blur Kernel Estimation. In *Proc. of ICCV*.

Sung-Jin Cho, Seo-Won Ji, Jun-Pyo Hong, Seung-Won Jung, and Sung-Jea Ko. 2021. Rethinking coarse-to-fine approach in single image deblurring. In *Proc. of ICCV*.

Robert T Collins. 1996. A space-sweep approach to true multi-image matching. In *Proc. of CVPR*.

Mauricio Delbracio and Guillermo Sapiro. 2015. Burst Deblurring: Removing Camera Shake Through Fourier Burst Accumulation. In *Proc. of CVPR*.

Akshay Dudhane, Syed Waqas Zamir, Salman Khan, Fahad Shahbaz Khan, and Ming-Hsuan Yang. 2022. Burst Image Restoration and Enhancement. In *Proc. of CVPR*.

Akshay Dudhane, Syed Waqas Zamir, Salman Khan, Fahad Shahbaz Khan, and Ming-Hsuan Yang. 2023. Burstormer: Burst Image Restoration and Enhancement Transformer. In *Proc. of CVPR*.

Yuming Fang, Hanwei Zhu, Yan Zeng, Kede Ma, and Zhou Wang. 2020. Perceptual Quality Assessment of Smartphone Photography. In *Proc. of CVPR*.

Zhenxuan Fang, Fangfang Wu, Weisheng Dong, Xin Li, Jinjian Wu, and Guangming Shi. 2023. Self-Supervised Non-Uniform Kernel Estimation With Flow-Based Motion Prior for Blind Image Deblurring. In *Proc. of CVPR*.

Rob Fergus, Barun Singh, Aaron Hertzmann, Sam T. Roweis, and William T. Freeman. 2006. Removing Camera Shake from a Single Photograph. *ACM Trans. Graph.* (2006).

Chen Haoyu, Teng Minggui, Shi Boxin, Wang YIzhou, and Huang Tiejun. 2020. Learning to deblur and generate high frame rate video with an event camera. *arXiv preprint arXiv:2003.00847* (2020).

Jie Hu, Li Shen, and Gang Sun. 2018. Squeeze-and-Excitation Networks. In *Proc. of CVPR*.

Zhe Jiang, Yu Zhang, Dongqing Zou, Jimmy Ren, Jiancheng Lv, and Yebin Liu. 2020. Learning event-based motion deblurring. In *Proc. of CVPR*.

Adam Kaufman and Raanan Fattal. 2020. Deblurring Using Analysis-Synthesis Networks Pair. In *Proc. of CVPR*.

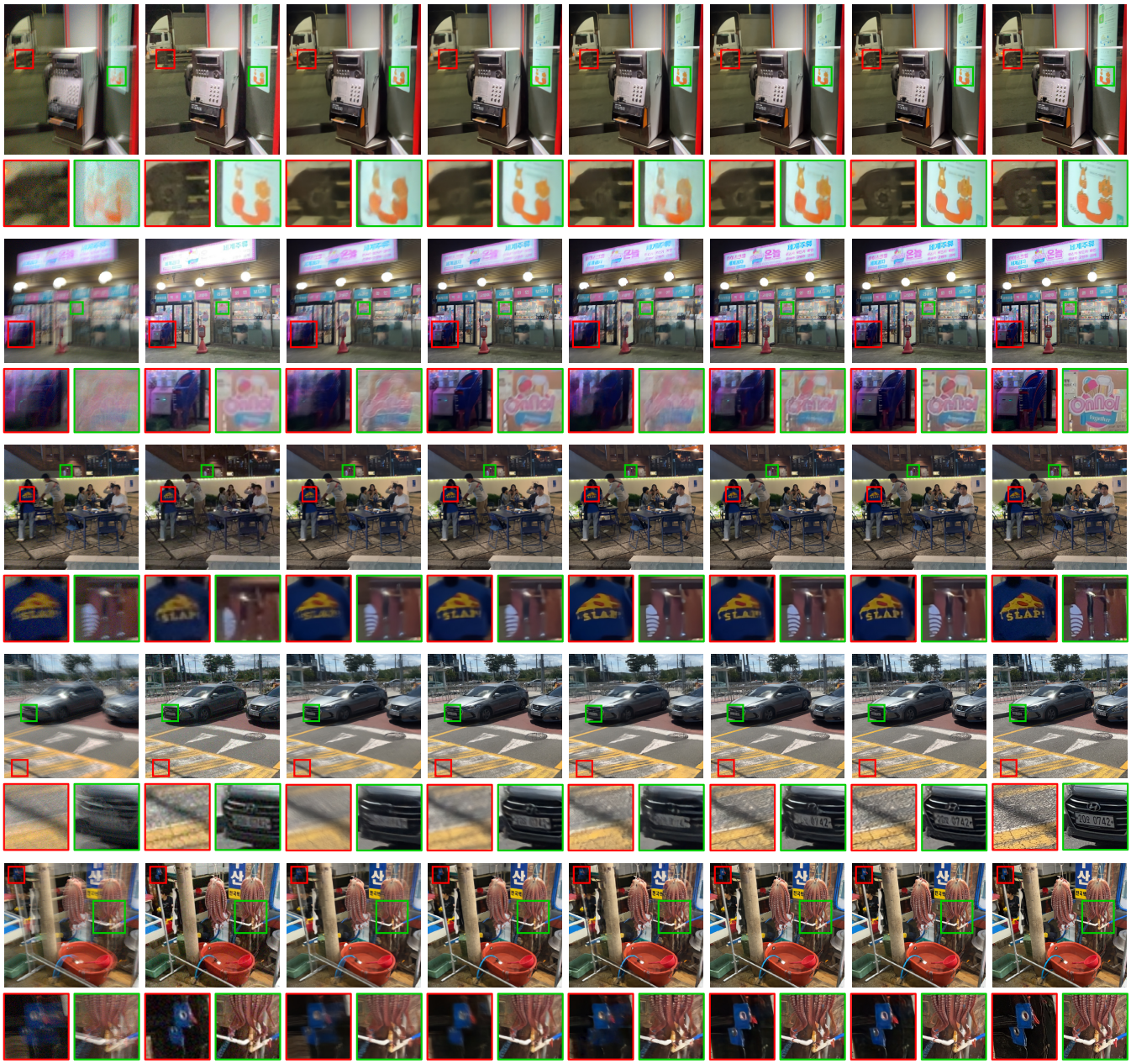
Kiyeon Kim, Seungyong Lee, and Sunghyun Cho. 2022a. MSSNet: Multi-Scale-Stage Network for Single Image Deblurring. In *Proc. ECCVW (AIM)*.

Taewoo Kim, Jeongmin Lee, Lin Wang, and Kuk-Jin Yoon. 2022b. Event-guided deblurring of unknown exposure time videos. In *Proc. of ECCV*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

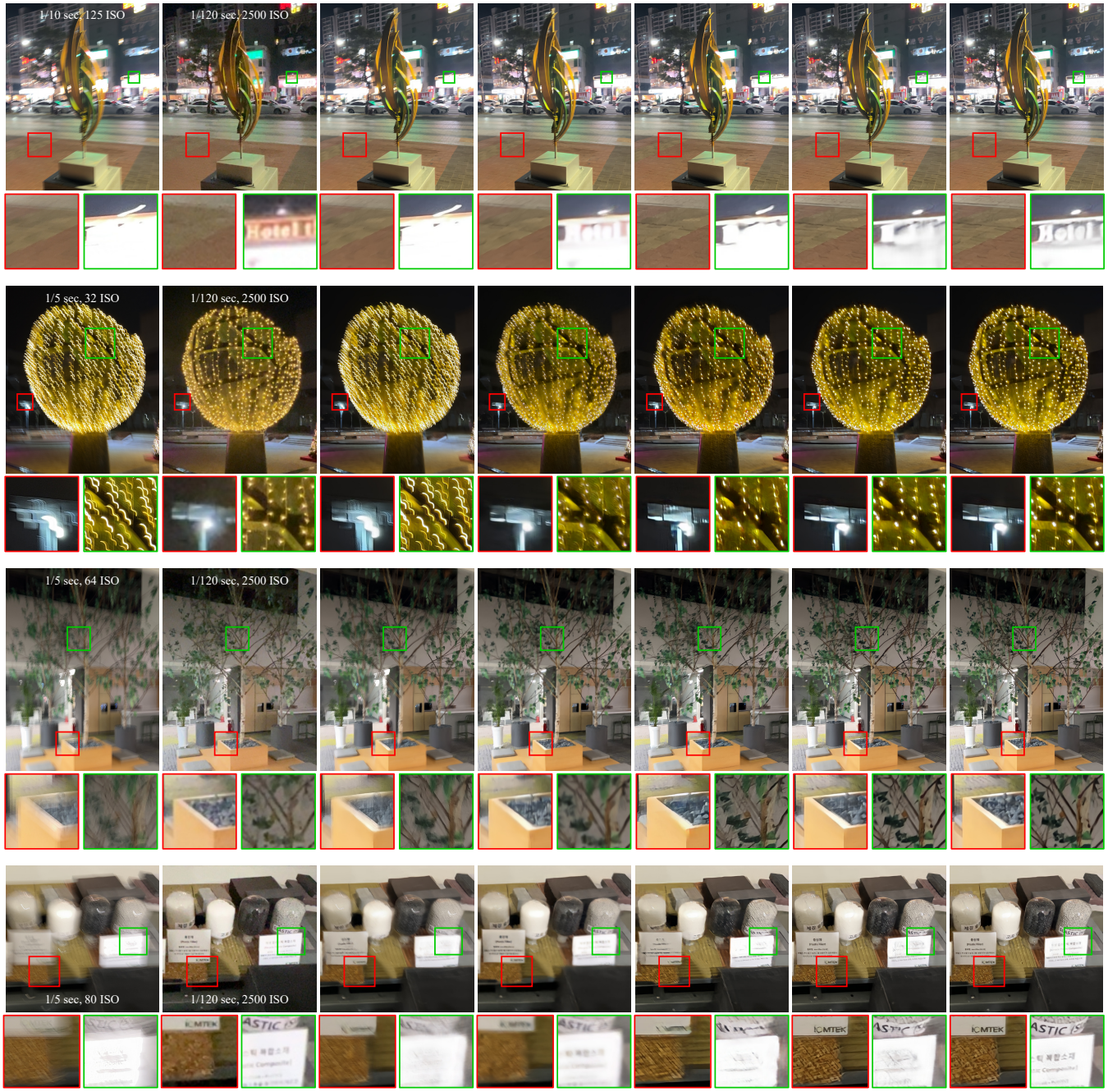
Orest Kupyn, Volodymyr Budzan, Mykola Mykhailych, Dmytro Mishkin, and Jiří Matas. 2018. Deblurgan: Blind motion deblurring using conditional adversarial networks.

In *Proc. of CVPR*.
Orest Kupyn, Tetiana Martyniuk, Junru Wu, and Zhangyang Wang. 2019. DeblurGAN-v2: Deblurring (Orders-of-Magnitude) Faster and Better. In *Proc. of ICCV*.
Wei-Sheng Lai, YiChang Shih, Lun-Cheng Chu, Xiaotong Wu, Sung-Fang Tsai, Michael Krainin, Deqing Sun, and Chia-Kai Liang. 2022. Face Deblurring using Dual Camera Fusion on Mobile Phones. *ACM Trans. Graph.* (2022).
A. Levin, Y. Weiss, F. Durand, and W. T. Freeman. 2009. Understanding and evaluating blind deconvolution algorithms. In *Proc. of CVPR*.
Feng Li, Jingyi Yu, and Jinxiang Chai. 2008. A hybrid camera for motion deblurring and depth map super-resolution. In *Proc. of CVPR*.
Ilya Loshchilov and Frank Hutter. 2016. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983* (2016).
Nancy Mehta, Akshay Dudhane, Subrahmanyam Murala, Syed Waqas Zamir, Salman Khan, and Fahad Shahbaz Khan. 2023. Gated Multi-Resolution Transfer Network for Burst Restoration and Enhancement. In *Proc. of CVPR*.
Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. 2012a. No-reference image quality assessment in the spatial domain. *IEEE Trans. Image Process.* (2012).
Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. 2012b. Making a "completely blind" image quality analyzer. *IEEE Signal Process. Lett.* (2012).
Janne Mustaniemi, Juho Kannala, Jiri Matas, Simo Särkkä, and Janne Heikkilä. 2020. LSD_2 – Joint Denoising and Deblurring of Short and Long Exposure Images with CNNs. In *Proc. of BMVC*.
Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. 2017. Deep Multi-Scale Convolutional Neural Network for Dynamic Scene Deblurring. In *Proc. of CVPR*.
Jinshan Pan, Deqing Sun, Hanspeter Pfister, and Ming-Hsuan Yang. 2016. Blind Image Deblurring Using Dark Channel Prior. In *Proc. of CVPR*.
Jaesung Rim, Geonung Kim, Jungeon Kim, Junyong Lee, Seungyong Lee, and Sunghyun Cho. 2022. Realistic Blur Synthesis for Learning Image Deblurring. In *Proc. of ECCV*.
Qi Shan, Jiaya Jia, and Aseem Agarwala. 2008. High-quality Motion Deblurring from a Single Image. *ACM Trans. Graph.* (2008).
Shayan Shekarforoush, Amanpreet Walia, Marcus A. Brubaker, Konstantinos G. Derpanis, and Alex Levinshtein. 2023. Dual-Camera Joint Deblurring-Denoising. arXiv:2309.08826
Libin Sun, Sunghyun Cho, Jue Wang, and James Hays. 2013. Edge-based Blur Kernel Estimation Using Patch Priors. In *Proc. of ICCP*.
Lei Sun, Christos Sakaridis, Jingyun Liang, Qi Jiang, Kailun Yang, Peng Sun, Yaozu Ye, Kaiwei Wang, and Luc Van Gool. 2022. Event-based fusion for motion deblurring with cross-modal attention. In *Proc. of ECCV*.
Yu-Wing Tai, Hao Du, Michael S. Brown, and Stephen Lin. 2008. Image/video deblurring using a hybrid camera. In *Proc. of CVPR*.
Yu-Wing Tai, Hao Du, Michael S. Brown, and Stephen Lin. 2010. Correction of Spatially Varying Image and Video Motion Blur Using a Hybrid Camera. *IEEE Trans. Pattern Anal. Mach. Intell.* (2010).
Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia. 2018. Scale-Recurrent Network for Deep Image Deblurring. In *Proc. of CVPR*.
Zachary Teed and Jia Deng. 2020. RAFT: Recurrent all-pairs field transforms for optical flow. In *Proc. of ECCV*.
Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. 2022. MAXIM: Multi-Axis MLP for Image Processing. In *Proc. of CVPR*.
Xintao Wang, Kelvin C.K. Chan, Ke Yu, Chao Dong, and Chen Change Loy. 2019. EDVR: Video restoration with enhanced deformable convolutional networks. In *Proc. of CVPRW*.
Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. 2022. Uformer: A General U-Shaped Transformer for Image Restoration. In *Proc. of CVPR*.
Fang Xu, Lei Yu, Bishan Wang, Wen Yang, Gui-Song Xia, Xu Jia, Zhendong Qiao, and Jianzhuang Liu. 2021. Motion deblurring with real events. In *Proc. of ICCV*.
Li Xu and Jiaya Jia. 2010. Two-Phase Kernel Estimation for Robust Motion Deblurring. In *Proc. of ECCV*.
Li Xu, Shicheng Zheng, and Jiaya Jia. 2013. Unnatural L0 sparse representation for natural image deblurring. In *Proc. of CVPR*.
Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. 2022. Restormer: Efficient Transformer for High-Resolution Image Restoration. In *Proc. of CVPR*.
Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. 2021. Multi-Stage Progressive Image Restoration. In *Proc. of CVPR*.
Hongguang Zhang, Yuchao Dai, Hongdong Li, and Piotr Koniusz. 2019. Deep Stacked Hierarchical Multi-Patch Network for Image Deblurring. In *Proc. of CVPR*.
Xiang Zhang, Lei Yu, Wen Yang, Jianzhuang Liu, and Gui-Song Xia. 2023. Generalizing Event-Based Motion Deblurring in Real-World Scenarios. In *Proc. of ICCV*.
Youjian Zhang, Chaoyue Wang, Stephen J Maybank, and Dacheng Tao. 2021. Exposure trajectory recovery from motion blur. *IEEE Trans. Pattern Anal. Mach. Intell.* (2021).
Yuzhi Zhao, Yongzhe Xu, Qiong Yan, Dingdong Yang, Xuehui Wang, and Lai-Man Po. 2022. D2hnet: Joint denoising and deblurring with hierarchical network for robust night image restoration. In *Proc. of ECCV*.
Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. 2019. Deformable convnets v2: More deformable, better results. In *Proc. of CVPR*.



(a) Blurred image W | (b) Center frame U_c | (c) NAFNet-64 | (d) NAFNet-Ref | (e) MotionETR | (f) HCDeblur (only HC-DNet) | (g) HCDeblur | (h) Ground truth

Figure 6: Qualitative comparisons on the HCBlur-Syn dataset. We compare HCDeblur with NAFNet-64 [Chen et al. 2022], NAFNet-Ref, and MotionETR [Zhang et al. 2021].



(a) Blurred image W (b) Center frame U_c (c) NAFNet-64 (d) NAFNet-Ref (e) MotionETR (f) HCDeblur (only HC-DNet) (g) HCDeblur

Figure 7: Qualitative comparison on the HCBlur-Real dataset. We compare HCDeblur with NAFNet-64 [Chen et al. 2022], NAFNet-Ref, and MotionETR [Zhang et al. 2021].