

Real-World Deep Local Motion Deblurring

Haoying Li^{1,2}, Ziran Zhang^{1*}, Tingting Jiang^{2*},
Peng Luo^{1*}, Huajun Feng^{1†}, Zhihai Xu¹

¹ College of Optical Science and Engineering, Zhejiang University

² Research Center for Intelligent Sensing Systems, Zhejiang Laboratory

{lhaoying, naturezhanghn, luop, fenghj, xuzh}@zju.edu.cn, eagerjtt@zhejianglab.com

Abstract

Most existing deblurring methods focus on removing global blur caused by camera shake, while they cannot well handle local blur caused by object movements. To fill the vacancy of local deblurring in real scenes, we establish the first real local motion blur dataset (ReLoBlur), which is captured by a synchronized beam-splitting photographing system and corrected by a post-progressing pipeline. Based on ReLoBlur, we propose a Local Blur-Aware Gated network (LBAG) and several local blur-aware techniques to bridge the gap between global and local deblurring: 1) a blur detection approach based on background subtraction to localize blurred regions; 2) a gate mechanism to guide our network to focus on blurred regions; and 3) a blur-aware patch cropping strategy to address data imbalance problem. Extensive experiments prove the reliability of ReLoBlur dataset, and demonstrate that LBAG achieves better performance than state-of-the-art global deblurring methods and our proposed local blur-aware techniques are effective.

Introduction

Single image deblurring has been persistently analyzed (Wang, Li, and Wang 2017; Zhou et al. 2019; Jin et al. 2021; Zhou, Li, and Change Loy 2022) and motion blur can be categorized into two kinds: global motion blur and local motion blur. In a global motion blurred image, blur exists in all regions of the image, and is usually caused by camera shake (Schelten and Roth 2014; Zhang et al. 2018). Fantastic progress has been made in global motion deblurring (Nah, Hyun Kim, and Mu Lee 2017; Kupyn et al. 2019; Chen et al. 2021). However, local motion deblurring is under few explorations, where blurs only exist in some of the regions of the image, and are mostly caused by object movements captured by a static camera.

Deep local motion deblurring in real scenes is a vital task with many challenges. Firstly, there is no public real local motion blur dataset for deep learning. Secondly, local motion deblurring is a complicated inverse problem due to the random localization of local blurs and the unknown blur extents. Besides, the blurred regions occupy only a small proportion of the full image, causing a deep neural network to

pay too much attention to the background. This data imbalance issue is contrary to the goal of local deblurring.

To tackle the local motion blur problems, data is the foundation. Existing deblurring datasets (Nayar and Ben-Ezra 2004; Köhler et al. 2012; Hu et al. 2016; Nah, Hyun Kim, and Mu Lee 2017) are mainly constructed for global deblurring with camera motion. Among them, a widely used approach is synthesizing blurred images by convolving either uniform or non-uniform blur kernels with sharp images (Boracchi and Foi 2012; Schuler et al. 2015; Chakrabarti 2016). However, this approach cannot assure the fidelity of blurred images. Another kind of approach shakes the camera to mimic blur caused by camera trembling, and averages consecutive short-exposure frames to synthesize global blurred images (Nah, Hyun Kim, and Mu Lee 2017; Nah et al. 2019). However, this kind of averaging approach cannot simulate real overexposure outliers due to the dynamic range of each frame (Chang et al. 2021). In this paper, we establish a real local motion blur dataset, ReLoBlur, captured by a static synchronized beam-splitting photographing system, which can capture local blurred and sharp images simultaneously. ReLoBlur only contains blur caused by moving objects, without camera motion blur. Moreover, we propose a novel paired image post-processing pipeline to address color cast and misalignment problems occurred in common beam-splitting systems (Tai et al. 2008; Rim et al. 2020). To the best of our acknowledgment, ReLoBlur is the first real local motion blur dataset captured in nature.

Since deep learning is proven to be successful in the global deblurring task, a direct way to remove local blur is to borrow ideas from deep global deblurring models. However, unlike global motion blur, local motion blur changes abruptly rather than smoothly at object boundaries, and the sharp background remains clear (Schelten and Roth 2014). Therefore, global deblurring networks fail to handle local motion blur and may raise image artifacts in sharp backgrounds. Enlightened by MIMO-Unet (Cho et al. 2021), we propose a Local Blur-Aware Gated deblurring method (LBAG) which localizes blurred regions and predicts sharp images simultaneously with novel local blur-aware deblurring techniques. To localize blurred regions, LBAG is trained to predict local blur masks under the supervision of local blur mask ground-truths, which are generated by a blur detection approach based on background subtraction.

*These authors contributed equally.

†Corresponding author

tion. With the help of predicted local blur masks, we safely introduce a gate block to LBAG, which guides the network to focus on blurred regions. To address the data imbalance issue, we propose a Blur-Aware Patch Cropping strategy (BAPC) to assure at least 50% of the input training patches contain local blur. We conduct experiments on ReLoBlur dataset and evaluate our proposed method in terms of PSNR, SSIM, aligned PSNR (Rim et al. 2020), weighted PSNR (Jamali, Karimi, and Samavi 2021) and weighted SSIM, the latter two of which specifically measure local performances. To sum up, our main contributions are:

- We establish the first real local motion blur dataset, ReLoBlur, captured by a synchronized beam-splitting photographing system in daily real scenes, and corrected by a post-processing pipeline. ReLoBlur contains 2405 image pairs and we will release the dataset soon.
- We develop a novel local blur-aware gated network, LBAG, with several local blur-aware techniques to bridge the gap between global and local deblurring.
- Extensive experiments show that ReLoBlur dataset enables efficient training and vigorous evaluation, and the proposed LBAG network exceeds other SOTA deblurring baselines quantitatively and perceptually.

Related Works

Image Deblurring Datasets

Local motion blur dataset draws little attention, while global blur datasets update rapidly. Approaches for generating global blurred images include: 1) convolving sharp images blur kernels (Chakrabarti 2016; Schuler et al. 2015; Sun et al. 2015); 2) averaging consecutive frames at very short intervals and selecting the central frame as the sharp image (Nah, Hyun Kim, and Mu Lee 2017; Nah et al. 2019); 3) using a coaxial beam-splitting system to simultaneously capture global blurred-sharp image pairs (Rim et al. 2020). Approach 1 and 2 are not suitable to capture real blur, as illustrated in . Approach 3 has drawbacks in color cast which is obvious in RealBlur dataset (Rim et al. 2020). This should be avoided because it may reduce the training performance in local deblurring tasks (We discuss this in our project’s homepage¹). To the best of our acknowledgment, there is no public dataset containing real local blur. We establish the first real local motion blur dataset, ReLoBlur, captured by a simultaneous photographing system and corrected through our post-processing pipeline.

Single Image Deep Deblurring Methods

Single image deep deblurring methods mainly focus on global blur tasks. Nah et al. (Nah, Hyun Kim, and Mu Lee 2017) introduced a multi-scale convolutional neural network that restores sharp images in an end-to-end manner. SRN-DeblurNet (Tao et al. 2018) overcame the training instability of multi-scaled deblurring networks by sharing network weights across scales. DeblurGAN (Kupyn et al. 2018) enhanced perceptual image quality of global deblurring by using perceptual losses. DeblurGAN-v2 (Kupyn et al. 2019)

¹https://leiali.github.io/ReLoBlur_homepage/index.html

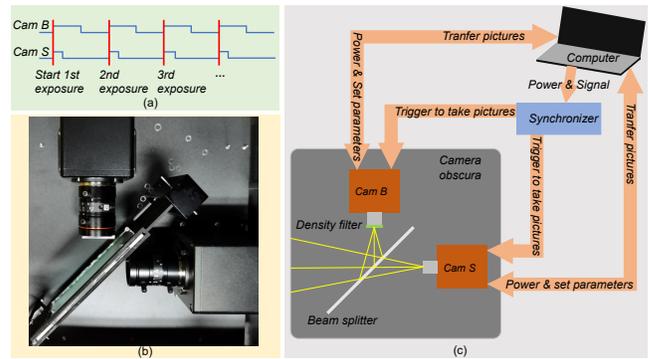


Figure 1: Overview of the paired image acquisition system: (a) the exposure mode following the local motion-blurred image formation; (b) a real picture of Camera B, S and the beam-splitting plate inside a camera obscura; (c) the synchronized beam-splitting photographing system.

accelerated DeblurGAN on training speed. Chen et al. (Chen et al. 2021) proposed HINET and refreshed the global deblurring score in the 2021 NITIRE Challenge. Nevertheless, these methods are not well-adopted in local deblurring tasks. Some algorithms (Schelten and Roth 2014; Pan et al. 2016) managed to deblur locally, but were limited within kernel estimation. Enlightened by Cho et al. (Cho et al. 2021), we propose a local blur-aware gated deblurring method for deep local deblurring, which recovers local blurred images by implementations of a gate block, a blur-aware patch cropping strategy and a local blur foreground mask generator.

ReLoBlur Dataset

In this section, we firstly describe the formation of local motion blur. Then, we introduce our image acquisition process based on the blur model. Finally, we introduce the paired image post-processing pipeline.

The Formation of Local Motion Blur

During camera exposure, the moving object’s actions accumulate as the camera sensor receives light every time, forming spatial aliasing, which presents local blur in real captured images. The local motion-blurred image formation can be modeled as:

$$B(x, y) = \text{ISP} \left(\int_{T_1}^{T_2} f(t, x, y) dt \right), \quad (1)$$

where B denotes the locally blurred color image, $T_2 - T_1$ denotes the total time of an exposure, and (x, y) denotes the pixel location. $f(t, x, y)$ means the photon response of pixel (x, y) at a time t . When $T_1 \rightarrow T_2$, Eq. 1 is the formation of the corresponding sharp ground-truth image. ISP means image signal processing operations, generating a Bayer raw image into a colorful RGB image.

Paired Image Acquisition

We manage to capture real local motion-blurred images by reproducing the formation of local motion blur mentioned

above. We use a scientific camera (Camera B) to collect local blurred image in a long exposure time t_L to accumulate local blur of moving objects. Simultaneously, we use another same camera (Camera S) to capture the corresponding sharp images in a very short exposure time t_S . The two cameras start exposing at the same time but end after different exposure times, as shown in Fig. 1(a). In order to share the two cameras same scene, Camera B and S are placed on the reflection end and transmission end of a beam-splitting plate, respectively. Each camera is placed 45 degrees to the 50% beam-splitting plate (the transmittance and reflectivity per spectrum is 50%), as shown in Fig.1(b)(c). In front of Camera B is a density filter with transmittance $\tau = \frac{t_S}{t_L}$, assuring the equivalence of photon energy received by the two cameras. Both Camera B and S are connected to a synchronizer, which triggers the two cameras to start exposing simultaneously in every shot but end exposure after different times. A computer is connected to the synchronizer, Camera B and S, powers and transmits data. In this way, the beam splitting photographing system can capture locally blurred images and their corresponding sharp images simultaneously according to the formation of local motion blur.

Before capturing image pairs, we adjust the settings of the synchronized beam-splitting photographing system according to the relationship between the exposure time and the object distance:

$$t_S = \frac{c \times n \times d}{l' \times v}, \quad (2)$$

where c , n , d , l' , v and t_S denote sensor-pixel side length, desired blurry pixels, object distance, image distance, object moving speed and short exposure time of Camera S, respectively, as shown in Fig. 1 (d). Eq. 2 indicates that the short exposure time is closely related to desired blurry pixels and object distance. Hence, we adjust the exposure time according to the desired blurry pixels and object distances, and we set the long exposure times as $t_L = \frac{t_S}{\tau}$.

In every scene, we capture a pair of whiteboard images for color correction, and a pair of static background images as reference for geometrical alignment in post-processing. We capture 2405 pairs of RAW images, conduct post-progressing to correct image degradations and convert RAW images to colorful RGB images with the image size of 2152×1436 . ReLoBlur includes but is not limited to indoor and outdoor scenes of pedestrians, vehicles, parents and children, pets, balls, plants and furniture. The local motion blur sizes range from 15 pixels to 70 pixels in blurry images, and no more than 6 pixels in sharp images. On average, the blur regions take up 11.75% of the whole image area in ReLoBlur dataset. We show our ReLoBlur dataset in Fig. 2 and our project's home page 1.

Paired Image Post-processing

As shown in Fig. 3, common coaxial systems raise several image degradations: 1) physically, the beam-splitting plate's transmission-reflection ratio varies with the incident angle, causing location-oriented color cast (Wang 2009; Fu et al. 2010), which reduces visual perception and may worsen local deblurring performance; 2) though the cameras and lens

at the transmission and reflection ends are of the same module, there still exists a slight brightness difference between a pair of blur and sharp images, due to the transmittance and reflectivity deviation of the beam-splitting plate and the unavoidable discrepancy of photovoltaic conversion of the two sensors; 3) despite carefully adjusting cameras' locations, spatial misalignments still exist because of unavoidable mechanical error. In this paper, we design a paired image post-processing pipeline to correct the above problems. The post-processing pipeline includes color correction, photometrical alignment, ISP and geometric alignment, as shown in Fig. 3.

Color Correction To solve the location-oriented color cast problem, we apply color correction coefficients α to conduct color correction on Bayer RAW images:

$$[P'_{R_k}, P'_{G_k}, P'_{B_k}] = [\alpha_{R_k}, \alpha_{G_k}, \alpha_{B_k}] \cdot [P_{R_k}, P_{G_k}, P_{B_k}], \quad (3)$$

where P and P' denote RAW pixel values of a local patch in different color channels before and after color correction, respectively. $k \in \{1, 2, 3, \dots, K\}$ is the patch number and K is the total patch number. The color correction coefficient α could be obtained by multiplying pixel coordinates and location-oriented color constants $\{a_0, a_1, a_2, \dots, a_9\}$:

$$\begin{bmatrix} \alpha_{R_1} & \dots & \alpha_{R_k} & \dots & \alpha_{R_K} \\ \alpha_{G_1} & \dots & \alpha_{G_k} & \dots & \alpha_{G_K} \\ \alpha_{B_1} & \dots & \alpha_{B_k} & \dots & \alpha_{B_K} \end{bmatrix} = [1 \ 1 \ 1]^T \times \begin{bmatrix} a_0 & a_1 & \dots & a_9 \end{bmatrix} \times Z, \quad (4)$$

$$Z = \begin{bmatrix} x_1^3 & x_1^2 y_1 & x_1 y_1^2 & \dots & y_1^2 & x_1 & y_1 & 1 \\ \vdots & & & & & & & \vdots \\ x_K^3 & x_K^2 y_K & x_K y_K^2 & \dots & y_K^2 & x_K & y_K & 1 \end{bmatrix}^T \quad (5)$$

The location-oriented color constants vary with different cameras. We manage to fix the location-oriented color constants by single camera color calibration in the laboratory.

For each camera, we capture a RAW image of a standard 6500K transmissive lightbox in a darkroom, and divide a RAW image into k patches. Each patch's location is its central pixel location (x_k, y_k) . Because there is no obvious color cast at the central region of each camera, we choose the central patch of each camera as the target patch. We calculate α of each patch k by using Eq. 3, replacing P' with the average pixel value of the patch k , and P with the average pixel value of the central patch. After obtaining α of each patch by channels, the location-oriented color constants are fixed by inversely deviating Eq. 5. In real capturing, we obtain color correction coefficient α applying Eq. 5 pixel by pixel and correct the color cast on RAW images by Eq.3.

Photometrical Alignment To eliminate the brightness difference between the sharp and locally blurred images, we photometrically align the locally blurred images to their corresponding sharp images through every color channel. Different from (Rim et al. 2020), we adjust the image brightness by parameterizing a brightness correction coefficient β :

$$\beta = (\beta_R, \beta_G, \beta_B) = \left(\frac{\overline{P_{RS}}}{\overline{P_{RB}}}, \frac{\overline{P_{GS}}}{\overline{P_{GB}}}, \frac{\overline{P_{BS}}}{\overline{P_{BB}}} \right), \quad (6)$$

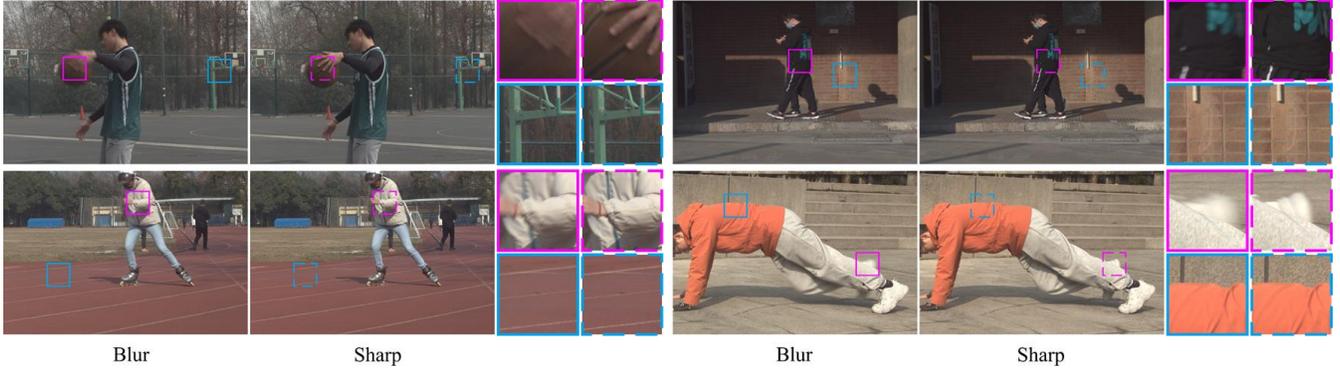


Figure 2: Examples of ReLoBlur dataset: the 1st and 5th columns are local motion-blurred images. The 2nd and 6th columns are the corresponding sharp images. The pink solid boxes and pink dotted boxes denote the locally blurred regions from locally blurred images, and the corresponding sharp regions from sharp images. The blue solid boxes and the blue dotted boxes denote the sharp regions from locally blurred images and the corresponding sharp regions from sharp images.

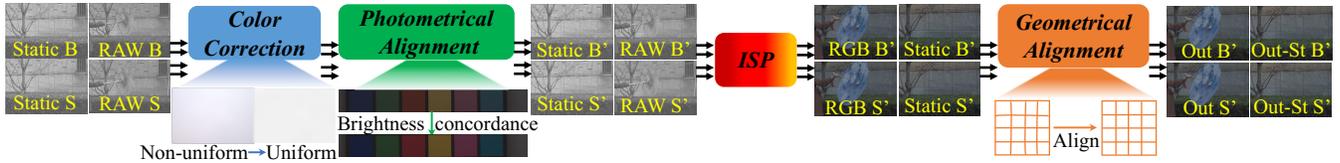


Figure 3: Paired image post-processing pipeline.

where \bar{P} denotes the channel mean value of a blur (or sharp) image. R , G and B stand for red, green and blue channels, respectively. B and S denote blur and sharp image, respectively. Set P_B and P'_B to be the pixel value of a locally blurred image before and after photometrical alignment, respectively. For each pixel at location (x, y) , a locally blurred image captured by Camera B is photometrically aligned to:

$$P'_B(x, y) = \beta \cdot P_B(x, y). \quad (7)$$

ISP Operations After photometrical alignment, we generate RGB images from RAW images by sequential ISP operations, as described in Eq. 1. We firstly demosaic the RAW images using Menon’s method (Menon, Andriani, and Calvagno 2006). Then we conduct white-balancing, color-mapping and Gamma correction in turns.

Geometrical Alignment We geometrically align locally blurred images to their corresponding sharp images by estimating optical flow and conduct image interpolation. In order to align the static background in every scene, we firstly use Camera B and S to capture a pair of static images as a reference image pair. While capturing, the photographing system, the background and foreground objects all remain still. Thus, the reference image pair can be used to calibrate the optical flow between the two cameras. In calibration, we use *Pyflow* method (Pathak et al. 2017) to calculate the optical flow. In the reference image pair, because the foreground object stands at the same location where it will move, there is almost no depth difference. The computed optical flow can be generally applied to all blurred-sharp image pairs in

the same scene. To better maintain edges, we utilize CMTF method (Freedman and Fleming 2016) for interpolation.

To prove the authenticity of ReLoBlur dataset, we quantitatively and perceptually evaluate the results of our paired image post-processing pipeline, including color correction, photometrical alignments and geometrical alignments. Details are shown in our project’s home page 1.

Local Blur-Aware Gated Model

To bridge the gap between global and local deblurring, in this section, we introduce our local blur-aware gated deblurring model, LBAG, which detects blurred regions and restores locally blurred images simultaneously. Fig. 4 provides an overview of our proposed model. Deriving from MIMO-UNet (Cho et al. 2021), we design a multi-scale Unet module. Different from global deblurring models, we specifically design a local blur detection method, gate blocks and blur-aware patch cropping strategy for local deblurring.

Ground-Truth Local Blur Mask Generation

To supervise the training of blurred region detection, providing ground-truth local blur masks is necessary. We develop a Local Blur Foreground Mask Generator (LBFMG) to generate ground-truth local blur masks of training data, based on Gaussian Mixture-based background/foreground segmentation method (Zivkovic 2004), which outputs the foreground with the help of input backgrounds. In locally blurred images, all static regions are regarded as backgrounds. Capturing all the static regions separately in natural scenes is impracticable, because some moving objects are closely con-

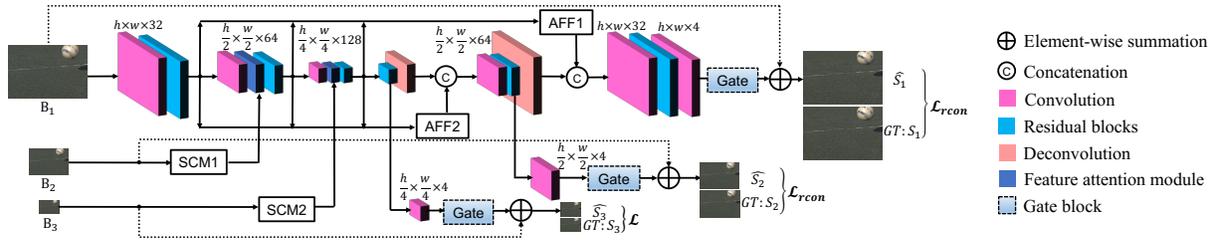


Figure 4: LBag network: SCM and AFF denote shallow convolutional module asymmetric feature fusion module, respectively.

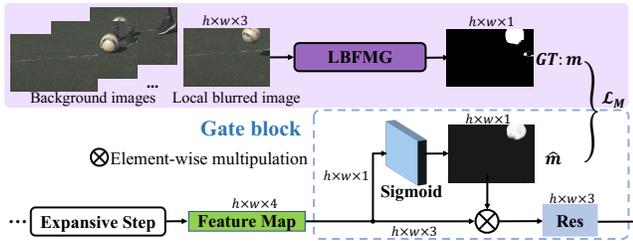


Figure 5: Ground-truth local blur mask generation (top) and the gate block (bottom).

nected to the static items. To obtain the input background for a locally blurred image B_T , we put all sharp and blurred images of the same scene except for B_T into LBFMG to update the background. Finally, B_T is put into LBFMG to generate its foreground mask as the ground-truth mask. The procedure of LBFMG is shown in Fig. 5 and we put more details of it in our project’s home page 1.

LBag: Local Blur-Aware Gated Network

We propose a local blur-aware gated network, LBag, based on MIMO-Unet (Cho et al. 2021). LBag exploits the MIMO-Unet architecture as the backbone, and gate blocks at the end of the network, as shown in Fig. 4 and Fig. 5. The backbone consists of three contracting steps, three expansive steps, two shallow convolutional (SCM) modules and two asymmetric feature fusion (AFF) modules. We take a locally blurred image with a different scale as an input of every extracting step, as many CNN-based deblurring methods demonstrated that multi-scale images can better handle different levels of blur (Michaeli and Irani 2014; Liu et al. 2015; Sun et al. 2015; Nah, Hyun Kim, and Mu Lee 2017; Suin, Purohit, and Rajagopalan 2020). To localize the local blurred regions, a gate block follows every expansive step.

As shown in Fig. 5, the **gate block** divides an input 4-channel feature map into a 3-channel latent and a 1-channel latent. The 1-channel latent passes through a sigmoid layer, forming a 1-channel pixel-level local blur mask prediction with pixel values ranging from 0 to 1, indicating how likely they are to be blurry (the higher the value, the more likely they are in locally blurred regions). For joint training, a mask-predicted loss is computed: $\mathcal{L}_M = MSE(m, \hat{m})$, where m and \hat{m} denote ground-truth local blur mask and predicted local blur mask. At the end of the gate structure, we multiply the 3-channel latent and the predicted local blur

mask to compute the residual image of a locally blurred image and its corresponding sharp image. The gate block helps the network to localize the locally blurred regions, so that the global deblurring backbone could only modify pixels in the predicted locally blurred regions, without harming the static background of a local blurred image.

Finally, the multi-scaled residual images are added to their corresponding multi-scaled locally blurred images to form predicted sharp images. A reconstruction loss \mathcal{L}_{recon} is calculated for supervised deblurring.

$$\mathcal{L}_{recon} = \lambda_2 \mathcal{L}_{ShMAE} + \lambda_3 \mathcal{L}_{ShSSIM} + \lambda_4 \mathcal{L}_{ShMSFR}$$

\mathcal{L}_{recon} includes mean absolute error (MAE) loss \mathcal{L}_{MAE} , SSIM loss \mathcal{L}_{SSIM} and multi-scale frequency reconstruction (MSFR) loss (Cho et al. 2021) \mathcal{L}_{MSFR} . The total loss of LBag is written as:

$$\mathcal{L} = \lambda_1 \mathcal{L}_M + \lambda_2 \mathcal{L}_{MAE} + \lambda_3 \mathcal{L}_{SSIM} + \lambda_4 \mathcal{L}_{MSFR}, \quad (8)$$

where $\lambda_1 = 0.01$, $\lambda_2 = \lambda_3 = 1$, and $\lambda_4 = 0.1$. A shift-invariant operation is applied on the total loss for better image reconstruction, which is explained detailedly in our project’s home page 1.

Blur-Aware Patch Cropping Strategy

In the local deblurring task, the local blurred regions only occupy a small percentage of the full image area (the average percentage is 11.75% in ReLoBlur dataset). And thus a deep neural network can be easy to pay much more attention to the clear background than the blurred regions. To tackle this data imbalance problem, we develop a blur-aware patch cropping strategy (BAPC) in training. Specifically, we randomly crop 256×256 patches from each training image with a 50% chance; otherwise, we randomly select the patch center ctr from the pixel in blurred regions marked by the ground-truth local blur mask, and then crop the patch centered with ctr . In this way, BAPC assures that LBag can pay more attention to blurred regions and thus solves the image imbalance problem.

Experiments and Analyses

Experiments

Experimental Settings LBag is trained and evaluated on ReLoBlur dataset. We split the ReLoBlur dataset into 2010 pairs for training and 395 pairs for testing, without repeated scenes occurring in each split set. Since the backbone of LBag has the same objective as global deblurring networks, we can initialize the parameters of our backbone us-

ing the MIMO-UNet weights which are pre-trained on Go-PRO dataset for fast convergence. We denote the LBAG with pre-trained model initialization as LBAG+. We crop the images to 256×256 patches as the training inputs using BAPC strategy. For data augmentation, each patch is horizontally or vertically flipped with a probability of 0.5. We use Adam (Kingma and Ba 2014) as the optimizer, with a batchsize of 12 and an initial learning rate of 10^{-4} , which is halved every 100k steps. The training procedure takes approximately 70 hours (300k steps). For a fair comparison, we trained LBAG and the baseline deblurring methods for the same steps on 1 GeForce RTX 3090 with 24GB of memory. The model configuration of the baseline methods follows their origins.

In testing, we put locally blurred images with the full image size of 2152×1436 into LBAG and baseline methods. We measure the full image restoration performances in terms of PSNR, SSIM. Because the ground-truth images are not strictly on the temporal centers of the moving tracks, we also calculate PSNR after temporally aligning the deblurred centers and the ground-truth centers of moving objects, which is denoted as $PSNR_a$, according to RealBlur (Rim et al. 2020). To evaluate local deblurring performances, we propose weighted PSNR and weighted SSIM:

$$PSNR_w(S, \hat{S}) = \frac{\sum_{x=1}^N \sum_{y=1}^M PSNR(S(x,y), \hat{S}(x,y)) \times Msk(x,y)}{\sum_{x=1}^N \sum_{y=1}^M Msk(x,y)},$$

$$SSIM_w(S, \hat{S}) = \frac{\sum_{x=1}^N \sum_{y=1}^M SSIM(S(x,y), \hat{S}(x,y)) \times Msk(x,y)}{\sum_{x=1}^N \sum_{y=1}^M Msk(x,y)},$$

where (x, y) , M , N and Msk are pixel location, image width, image height, and the ground-truth local blur mask, respectively. S and \hat{S} denote sharp ground-truth and predicted sharp image, respectively.

Local Deblurring Performance We compare the proposed method on ReLoBlur dataset with the following four deburring methods: DeepDeblur (Nah, Hyun Kim, and Mu Lee 2017), DeblurGAN-v2 (Kupyn et al. 2019), SRN-DeblurNet (Tao et al. 2018) and HINet (Chen et al. 2021). Fig. 6 presents the visual comparison of deblurring results and local blur masks of a boy’s hand and a man’s shoe as instances. The results indicate that LBAG can well predict local blur regions and recover locally blurred images simultaneously. Compared with SOTA deblurring methods, LBAG removes the foreground motion blur without harming the textures, while SRN-DeblurNet deforms the foreground. For example, the hand and the shoe are distorted and lose consistency with the ground truths. Moreover, LBAG network generates sharper images with rich content, while DeepDeblur, DeblurGAN-v2, HINet and MIMO-Unet miss the detailed information. This demonstrates that global deblurring network is not suitable for local motion deblurring without focusing on the extreme blur changes of the foregrounds. Quantitative local deblurring results in Tab. 1 show that LBAG exceeds other methods in terms of PSNR, SSIM, weighted PSNR and weighted SSIM with comparable model parameters. And with a pre-trained MIMO-Unet model, LBAG+ better reconstructs local motion-blurred images because it provides effective deblurring initialization. This proves that our methods are adept at removing local blur as well as maintaining the sharp background contents.

Methods	$\uparrow PSNR$	$\uparrow SSIM$	$\uparrow PSNR_w$	$\uparrow SSIM_w$	$\uparrow PSNR_a$
DeepDeblur	33.05	0.8946	26.51	0.8152	33.70
DeblurGAN-v2	33.85	0.9027	27.37	0.8342	34.30
SRN-DeblurNet	34.30	0.9238	27.48	0.8570	34.88
HINet	34.36	0.9151	27.64	0.8510	34.95
MIMO-Unet	34.52	0.9250	27.95	0.8650	35.42
LBAG	34.66	0.9249	28.25	0.8692	35.39
LBAG+	34.85	0.9257	28.32	0.8734	35.53

Table 1: Quantitative results of comparing local deblurring methods. “ $PSNR_w$ ”, “ $SSIM_w$ ” and “ $PSNR_a$ ” denote weighted PSNR, weighted SSIM and aligned PSNR, respectively. “LBAG+” denotes LBAG with pre-trained MIMO-Unet model.

Training data	$\uparrow PSNR$	$\uparrow SSIM$	$\uparrow PSNR_w$	$\uparrow SSIM_w$
Synthetic data	34.03	0.9015	27.42	0.8366
ReLoBlur data	34.85	0.9257	28.32	0.8734

Table 2: Quantitative results of synthetic-data-trained LBAG+ and ReLoBlur-trained LBAG+.

Analyses

Dataset Analyses To evaluate the effectiveness and non-substitution of ReLoBlur dataset for local motion deblurring task, we train LBAG both on ReLoBlur dataset and synthetic local blurred data. Because there is no public local motion-blurred dataset, we construct synthetic data by adding local motion-blur to the sharp images of our ReLoBlur dataset. The construction of synthetic data is explained in our project’s home page 1. We show an example of the synthetic data in Fig. 7.

The deblurring results are shown in Fig. 8 and Tab. 2. We notice that the deblurring model trained on synthetic data could not remove the local blur and obtains lower quantitative scores on evaluation metrics. This demonstrates that synthetic blur remains a gap with real local blur, and the synthetic local blur data can not replace real local blurred data for the training of local deblurring networks. Our ReLoBlur dataset enables efficient deep local deblurring training and vigorous evaluation compared with the synthetic data.

Analyses of Local Deblurring Model To verify the effects of our proposed local deblurring techniques, we compare LBAG with or without gate blocks, BAPC strategy, SSIM loss term and pre-trained model parameter initialization, as shown in Tab. 3. We have the following observations:

- Comparing line 1 and line 2, we can see that both global metrics (weighted PSNR and weighted SSIM) and local metrics (PSNR and SSIM) drop when removing the gate blocks, indicating that gate blocks can effectively neglect background objects and focus on blurry objects.
- Comparing line 1 and line 3, we find that both global and local metrics drop without BAPC strategy, which proves BAPC’s importance. Besides, the local metrics drop greater than global metrics, which further demonstrates that BAPC strategy encourages our model to pay more attention to blurred regions and facilitates training.

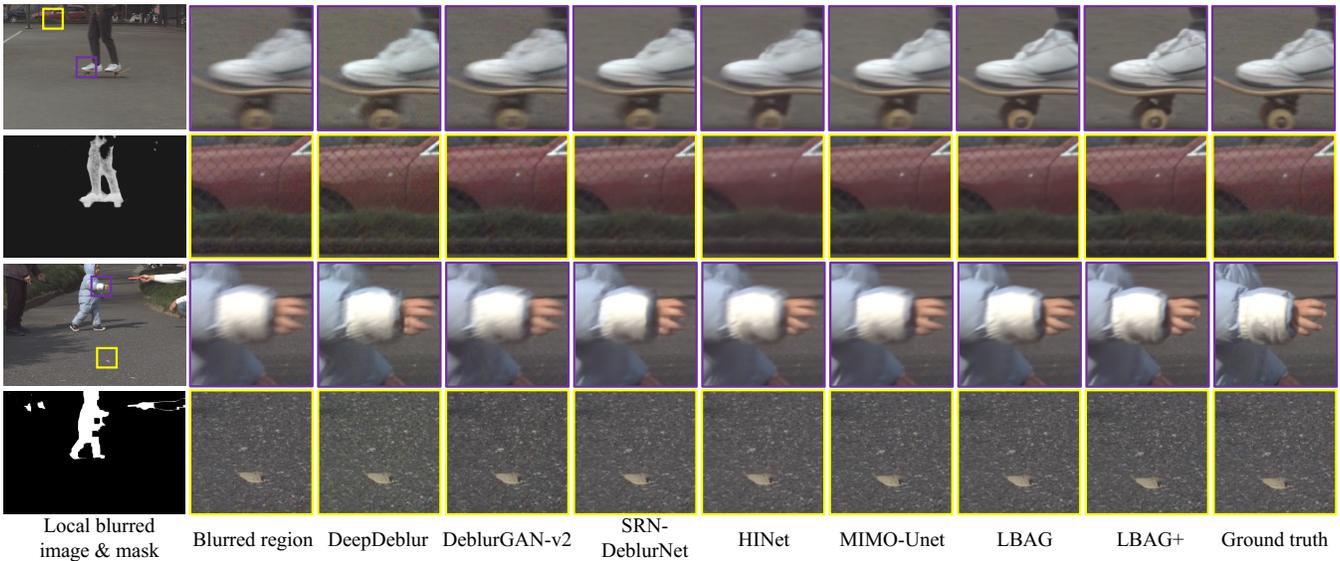


Figure 6: Visual comparison of different deblurring methods on ReLoBlur dataset: purple frames: locally blurred regions and deblurred regions; yellow frames: sharp regions and corresponding regions in deblurred images. More results are shown in our project’s home page.

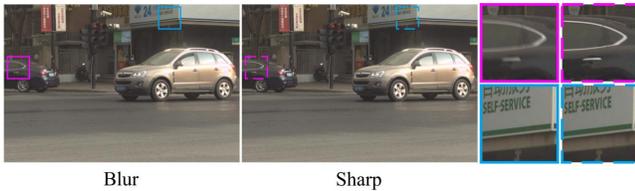


Figure 7: Examples of synthetic data: images in the 1st and 2nd columns are locally blurred images and their sharp ground-truths. The pink solid boxes and pink dotted boxes denote blurred regions from locally blurred images, and the corresponding sharp regions from sharp images. The blue solid boxes and the blue dotted boxes denote sharp regions from locally blurred images, and the corresponding sharp regions from sharp images.

- Comparing line 1 and line 4, we see that SSIM loss term can significantly improve the SSIM-related scores.
- Comparing line 1 and line 5, we notice that the absence of loading pre-trained model limits network expressions, because the pre-trained MIMO-Unet model produces effective deblurring initialization for our gated network.

Conclusion

This paper dealt with local blur problem and bridged the gap between global and local deblurring tasks. We constructed the first real local motion blur dataset, ReLoBlur, containing sharp and locally blurred images in real scenes. We addressed color cast and misalignment problems occurred in original-shot images through our novel post-processing pipeline. To conquer the challenges in local deblurring, we proposed a local blur-aware gated method, LBAG, with lo-

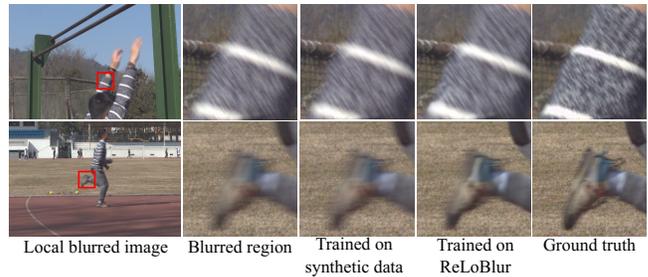


Figure 8: Visual deblurring results of synthetic-data-trained LBAG+ and ReLoBlur-trained LBAG+.

No.	Ga.	BAPC	\mathcal{L}_{SSIM}	Pretr.	\uparrow PSNR	\uparrow SSIM	\uparrow PSNR _w	\uparrow SSIM _w	\uparrow PSNR _t
1(LBAG+)	✓	✓	✓	✓	34.85	0.9257	28.32	0.8734	35.53
2		✓	✓	✓	34.67	0.9256	27.88	0.8680	35.30
3	✓		✓	✓	34.75	0.9255	28.17	0.8695	35.48
4	✓	✓		✓	34.62	0.9254	27.85	0.8639	35.28
5(LBAG)	✓	✓	✓		34.68	0.9256	28.10	0.8677	35.43

Table 3: Ablations of LBAG on ReLoBlur. *Ga.* and *Pretr.* denote abbreviations of gate block and pre-training strategy.

cal deblurring techniques including a local blur region detection method, gate blocks and a blur-aware patch cropping strategy. Extensive experiments show that ReLoBlur enables efficient deep local deblurring and vigorous evaluation, and LBAG outperforms the SOTA local deblurring methods qualitatively and quantitatively. In the future, we will fasten LBAG inference speed and improve the deblurred image quality by introducing generative models.

Acknowledgments

This work is supported by Jiangsu Science and technology development Foundation.

References

- Boracchi, G.; and Foi, A. 2012. Modeling the performance of image restoration from motion blur. *IEEE Transactions on Image Processing*, 21(8): 3502–3517.
- Chakrabarti, A. 2016. A neural approach to blind motion deblurring. In *European conference on computer vision*, 221–235. Springer.
- Chang, M.; Yang, C.; Feng, H.; Xu, Z.; and Li, Q. 2021. Beyond camera motion blur removing: How to handle outliers in deblurring. *IEEE Transactions on Computational Imaging*, 7: 463–474.
- Chen, L.; Lu, X.; Zhang, J.; Chu, X.; and Chen, C. 2021. HINet: Half instance normalization network for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 182–192.
- Cho, S.-J.; Ji, S.-W.; Hong, J.-P.; Jung, S.-W.; and Ko, S.-J. 2021. Rethinking coarse-to-fine approach in single image deblurring. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4641–4650.
- Freedman, E.; and Fleming, R. 2016. The Constant MTF Interpolator a resampling technique with minimal MTF losses. In *2016 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, 1–6. IEEE.
- Fu, T.; Yang, Z.; Wang, L.; Cheng, X.; Zhong, M.; and Shi, C. 2010. Measurement performance of an optical CCD-based pyrometer system. *Optics & Laser Technology*, 42(4): 586–593.
- Hu, Z.; Yuan, L.; Lin, S.; and Yang, M.-H. 2016. Image deblurring using smartphone inertial sensors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1855–1864.
- Jamali, M.; Karimi, N.; and Samavi, S. 2021. Weighted Fuzzy-Based PSNR for Watermarking. *arXiv preprint arXiv:2101.08502*.
- Jin, X.; Xu, J.; Tasaka, K.; and Chen, Z. 2021. Multi-task Learning-based All-in-one Collaboration Framework for Degraded Image Super-resolution. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 17(1): 1–21.
- Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Köhler, R.; Hirsch, M.; Mohler, B.; Schölkopf, B.; and Harmeling, S. 2012. Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database. In *European conference on computer vision*, 27–40. Springer.
- Kupyn, O.; Budzan, V.; Mykhailych, M.; Mishkin, D.; and Matas, J. 2018. Deblurgan: Blind motion deblurring using conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 8183–8192.
- Kupyn, O.; Martyniuk, T.; Wu, J.; and Wang, Z. 2019. DeblurGAN-v2: Deblurring (Orders-of-Magnitude) Faster and Better. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- Liu, S.; Wang, H.; Wang, J.; and Pan, C. 2015. Blur-kernel bound estimation from pyramid statistics. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(5): 1012–1016.
- Menon, D.; Andriani, S.; and Calvagno, G. 2006. Demosaicing with directional filtering and a posteriori decision. *IEEE Transactions on Image Processing*, 16(1): 132–141.
- Michaeli, T.; and Irani, M. 2014. Blind deblurring using internal patch recurrence. In *European conference on computer vision*, 783–798. Springer.
- Nah, S.; Baik, S.; Hong, S.; Moon, G.; Son, S.; Timofte, R.; and Mu Lee, K. 2019. Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 0–0.
- Nah, S.; Hyun Kim, T.; and Mu Lee, K. 2017. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3883–3891.
- Nayar, S. K.; and Ben-Ezra, M. 2004. Motion-based motion deblurring. *IEEE transactions on pattern analysis and machine intelligence*, 26(6): 689–698.
- Pan, J.; Hu, Z.; Su, Z.; Lee, H.-Y.; and Yang, M.-H. 2016. Soft-segmentation guided object motion deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 459–468.
- Pathak, D.; Girshick, R.; Dollár, P.; Darrell, T.; and Hariharan, B. 2017. Learning features by watching objects move. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2701–2710.
- Rim, J.; Lee, H.; Won, J.; and Cho, S. 2020. Real-world blur dataset for learning and benchmarking deblurring algorithms. In *European Conference on Computer Vision*, 184–201. Springer.
- Schelten, K.; and Roth, S. 2014. Localized image blur removal through non-parametric kernel estimation. In *2014 22nd International Conference on Pattern Recognition*, 702–707. IEEE.
- Schuler, C. J.; Hirsch, M.; Harmeling, S.; and Schölkopf, B. 2015. Learning to deblur. *IEEE transactions on pattern analysis and machine intelligence*, 38(7): 1439–1451.
- Suin, M.; Purohit, K.; and Rajagopalan, A. 2020. Spatially-attentive patch-hierarchical network for adaptive motion deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3606–3615.
- Sun, J.; Cao, W.; Xu, Z.; and Ponce, J. 2015. Learning a convolutional neural network for non-uniform motion blur removal. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 769–777.
- Tai, Y.-W.; Du, H.; Brown, M. S.; and Lin, S. 2008. Image/video deblurring using a hybrid camera. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 1–8. IEEE.

- Tao, X.; Gao, H.; Shen, X.; Wang, J.; and Jia, J. 2018. Scale-recurrent network for deep image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 8174–8182.
- Wang, L.; Li, Y.; and Wang, S. 2017. DeepDeblur: fast one-step blurry face images restoration. *arXiv preprint arXiv:1711.09515*.
- Wang, W. 2009. Wide-angle and broadband nonpolarizing parallel plate beam splitter. In *4th International Symposium on Advanced Optical Manufacturing and Testing Technologies: Advanced Optical Manufacturing Technologies*, volume 7282, 72821F. International Society for Optics and Photonics.
- Zhang, S.; Shen, X.; Lin, Z.; Měch, R.; Costeira, J. P.; and Moura, J. M. 2018. Learning to understand image blur. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 6586–6595.
- Zhou, S.; Li, C.; and Change Loy, C. 2022. LEDNet: Joint Low-Light Enhancement and Deblurring in the Dark. In *European Conference on Computer Vision*, 573–589. Springer.
- Zhou, S.; Zhang, J.; Zuo, W.; Xie, H.; Pan, J.; and Ren, J. S. 2019. Davanet: Stereo deblurring with view aggregation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10996–11005.
- Zivkovic, Z. 2004. Improved adaptive Gaussian mixture model for background subtraction. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 2, 28–31. IEEE.