Multimedia Tools and Applications (2023) 82:17055–17074
https://doi.org/10.1007/s11042-022-14116-z

Two-stage single image Deblurring network based on deblur kernel estimation

Ying Cheng Lu[1] · Tzu Pu Liu[1] · Chang Hong Lin[1]

Received: 5 January 2022 / Revised: 23 September 2022 / Accepted: 25 October 2022 /
Published online: 7 November 2022

Abstract

Image deblurring for dynamic scenes is a serious challenge in computer vision. Motion blur is caused by camera shaking or object movement during the exposure time. Many photos cannot be reproduced at the moment they were taken, its contents cannot be restored if motion blur occurs. In this article, we proposed a deblurring system that uses a two-stage convolutional neural network (CNN) to achieve image deblurring through a joint learning strategy. The first-stage network predicts the deblur kernel of each pixel and pre-deblurs the input image, and then the second-stage network directly predicts clear images based on U-Net architecture. In the first-stage network, the deblur kernel uses the surrounding information to restore the centre pixel, which can effectively remove the tiny motion blur. To additionally deal with large motion blur, we extend the second-stage network is used to compensate for the limited receptive field of the first-stage deblurring kernel. We evaluate the proposed method on benchmark blur datasets. Experimental results show that the proposed method can produce better results than state-of-the-art methods, both quantitatively and qualitatively. The proposed method can achieve the best PSNR at 32.59db, 27.21db and 31.96db for the GOPRO, Köhler, and Su datasets, respectively.

Keywords Image Deblurring, Image quality improvement · Deep learning · Convolution neural network (CNN) · Joint learning

1 Introduction

As technology develops, more applications use images for recognition and analysis. Motion blur is one of the common photography artifacts in dynamic environments. It is an effect caused by the relative movements of the camera, the object, or the background. An image

✉ Chang Hong Lin
chlin@mail.ntust.edu.tw

[1] Department of Electronic and Computer Engineering, National Taiwan University of Science and Technology, No. 43, Keelung Rd., Da'an Dist, Taipei, Taiwan

generated by a camera is actually the accumulated scenes over the exposure time. Therefore, when objects are moving, the camera would record the entire process of moving within the exposure time in the image, which is why the image becomes blurred. The performance will be affected if the input image is blurred. Image deblurring technology not only allows us to restore lost images but also helps some high-level image processing methods to improve performance. The blurred images have various distortions that make the images difficult to recognize.

The deep learning techniques handle the difficulty of finding useful features, because it can learn the useful features through training on a large amount of data. The learning-based methods have shown advanced performance in many computer vision problems [5, 23, 24, 29, 41], including the deblurring problem [8, 22, 24, 27, 30, 32–34, 44]. Many network modules and functional units for image restoration have been exploited in the past, including recursive residual learning [2, 42], dilated convolutions [2, 39], attention mechanisms [6, 42], encoder-decoders [3, 44], and generative models [19, 45]. More recent works [8, 22, 30, 32–34, 44] used the end-to-end training networks to reach the goal of the image deblurring. There are some works [24, 32, 34, 44] used multi-hierarchy architecture, which makes each level of network focus on learning features in different size, and then combine those networks to solve the problem.

The proposed two stages deblurring network is composed of two different networks to deal with different types of motion blur. The first stage network predicts the deblur kernel of each pixel and pre-deblurs the input image, and then the second stage network further eliminates the blur to output the final deblurring result. Since latent pixels' information are scattered in a motion blurred image, the deblur kernel is to use the surrounding information to restore the center pixel, which can effectively remove the small motion blur. However, the deblur kernel is not effective in large motion blur, so the second stage network is used to compensate for the limited receptive field of the first stage deblur kernel.

In this article, we present a single image deblurring system. We briefly summarize the contributions of this work as follows:

1) We proposed a two stages deblurring network to deal with different types of motion blur.
2) In the first stage, we use the pixel-wise kernel estimation network to predict the deblur kernel of each pixel to restore the sharp image in pixel-level.
3) In the second stage, we use the image deblurring network to further refine the blurred image with a global view.
4) We use several image processing techniques to augment the training dataset. This allows the training data to be more diversified so that the proposed network can be adapted to a variety of different blur scenarios without overfitting.
5) We use four different loss functions to train our proposed network. This allows the proposed network has more information of the difference between output and ground truth to reach a better deblurring result.
6) We adopt joint learning to train the two stages network simultaneously. This allows the proposed network to find the global optimum of the whole task.

As mentioned above, our method can achieve a good deblurring result. We demonstrate our method and some state-of-the-art method on four different datasets. The results show that our method has better deblurring performance both qualitatively and quantitatively.

2 Related work

Conventional approaches usually require explicit estimation of the blur kernel, and then deconvolve the kernel with the blurry image to generate a sharp image. There have been several works on estimating the uniform blur kernels, and these methods usually assume that the blur caused by camera shake during the exposure time are uniform with negligible in-plane camera rotation.

Uniform blur kernel estimation usually assume that the blur caused by camera shake during the exposure time are uniform with negligible in-plane camera rotation. Fergus et al. [7] proposed a variational Bayes approach with natural image statistics to estimate the blur kernel. This method uses an iterative approach to improve the estimate of the motion kernel and sharp image on each iteration. Hence, the running time, as well as the stopping criterion, is a significant problem for these kinds of algorithms. Cho et al. [4] proposed a fast deblurring method, which exploited the blurred strong edges to reliably estimate blur kernel and accelerate both latent image estimation and kernel estimation in an iterative deblurring process. Shan et al. [28] proposed a model of the spatial randomness of noise to separate the errors that arise during the blur kernel estimation, as well as a new local smoothness prior that reduces ringing artifacts. However, the uniform blur is unreasonable in reality, and ideal assumptions cannot achieve good results.

Real camera blur is non-uniform, there is also a lot of work on non-uniform blur kernel estimation. They are mainly in predicting non-uniform blur. Hirsch et al. [11] divide the image into several locally uniform overlapping-patch to predict each blur kernel of different patches. Gupta et al. [9] model the camera motion as a motion density function to estimate spatially variant blur kernels. Sun et al. [33] propose a deep learning approach to predict the probabilistic distribution of motion blur at the patch level. There have also been some works that rely on an accurate image segmentation mask to estimate different blur kernels for corresponding image regions. Pan et al. [25] split an image into different layers according to moving objects and assume that each layer corresponds to a blur kernel. Kim et al. [15] proposed a deblurring framework that can adaptively combine different blur models to estimate the spatially varying blur kernels. Kim and Lee [14] approximated the blur kernel to be locally linear and proposed an approach that estimates both the latent image and the locally linear motions jointly. The problem with non-uniform in real camera blur is also a problem faced in the past studies.

Due to the recent rapid development of deep neural networks, various advanced methods have also been proposed. As for the high-frame-rate camera becomes available, Kernel-Free for image deblurring started to be mentioned and studied. We can acquire a large number of blur and blur-free image pairs synthetic by consecutive frames. Therefore, many works directly restore the sharp image by learning the mapping functions from blur to blur-free images through a convolution neural network without estimating blur kernels. Lim et al. [22] proposed a deep spectral-spatial network, which used a two stages encoder-decoder network in a cascaded scheme to restore the sharp image by learning both spectral and spatial features. Zou et al. [46] proposed an architecture called SDWNet, obtaining different receptive fields by using dilated convolution modules, and the wavelet transform module makes the restored image contain more high-frequency details. Ye et al. [40] proposed a scale-iterative upscaling network, which has two levels and implements the iterative process of downsampling a series of tasks with smaller image scale. These methods all use a multi-level architecture and deblur from a tiny scale. Liang et al. [20] proposed a raw image deblurring network architecture consisting of spatial and color encoder, and bidirectional cross-modal attention. The architecture leads a shorter runtime and good large-scale shaking elimination. However, these architectures cannot be more reserved for details. To achieve better results, we proposed a two-stage single image deblurring network based on deblur kernel estimation, which recovers the original sharp image from the cascaded architecture by estimating the deblur kernel in pixel-level and learning latent blur features. Based on our proposed method, good results are indeed obtained.

In addition, with the rapid development of medical assistance technology, deblurring is also applied to medical images. Tien et al. [35] proposed the CycleDeblur GAN, which combined the CycleGAN and Deblur-GAN deep learning models to improve the quality of chest CBCT images. It also does produce good results in post-processing CT imaging. Ahmed et al. [1] proposed an unsupervised bilinear model by using convolutional neural networks as parameters. It also achieves good results in the performance of removing blur.

3 Proposed deblurring network

The proposed deblurring network is based on two different U-net architecture [26] to achieve a single image deblurring method, the flow chart is shown in Fig. 1. In the training process, the input blur image would first be randomly pre-processed through a series of data augmentation tasks, such as cropping, flipping, rotation, and hue and saturation adjustments. The first stage pixel-wise kernel estimation network estimates a 5×5 deblur kernel for each pixel to eliminate local blur in the pixel-level. The second stage image deblurring network learns latent blur features from the original blurry input image and the preliminary result image from the first stage. In the system trained by our proposed deblurring network, the first stage deblurring aims to eliminate local blur in pixel-level, while the second stage deblurring generates the final deblur image with a global view.

3.1 Pixel-wise kernel estimation network

We introduce the first stage of the proposed deblurring network. The proposed pixel-wise kernel estimation network is based on the U-net architecture [26]. The overall architecture of

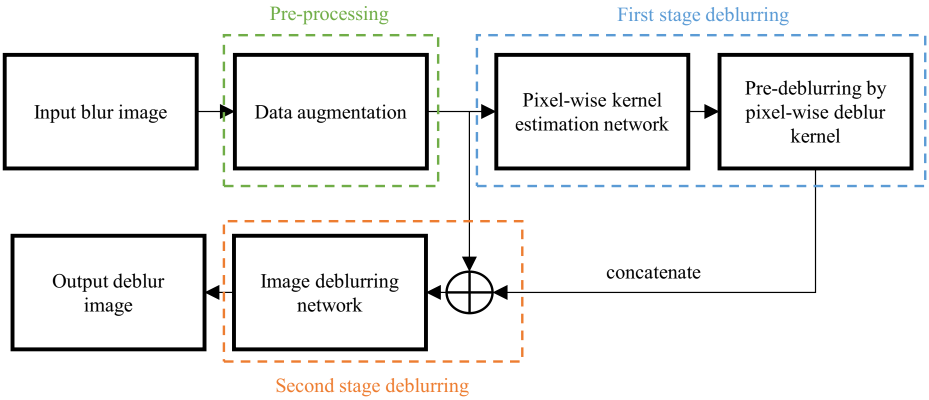

Fig. 1 Flow chart of the whole system

the proposed kernel estimation network is shown in Fig. 2. The input of the proposed network is a 3-channel blurry image, and the output is a 25-channel result, which indicates that the predicted window size of the pixel-wise deblur kernel in the proposed method setting is 5×5. After that, the output pixel-wise deblur kernel of this network will be passed to the dynamic local filtering [12, 13] to get a pre-deblurring result.

Because of the motion blur is caused by camera shaking or object moving in the captured image, so the original pixel's information will scatter into the surrounding pixels. To deal with the motion blur, we propose the pixel-wise kernel estimation network to predict the pixel-wise deblur kernel of the blurry input image. It can use the surrounding pixel's information to solve the effect of motion blur. Finally, we can acquire the pre-deblurring result to feed into the next stage deblurring network, as shown in Fig. 3.

The pixel-wise deblur kernel is to restore the center pixel's information from the neighbor pixels, as shown in Fig. 3b. Because of these predicted deblur kernels restore the blurry image in pixel-level, so it has a good ability to restore local and non-uniform blur regions. However, the large motion blur cannot be well restored due to the limitation of the window size of the predicted deblur kernels.

We adopt the dynamic local filtering [12, 13] on the blurry input image and pixel-wise deblur kernel. The operation of dynamic local filtering can be divided into three parts. First, extract the blurry input patches of each pixel and then do an element-wise product with the pixel-wise deblur kernel of the corresponding pixel position. The regions out of the image range are filled with zero values. Finally, add all of the element-wise product results to generate the pre-deblurring result, as shown in Fig. 3c. It represents the pixel-wise deblur

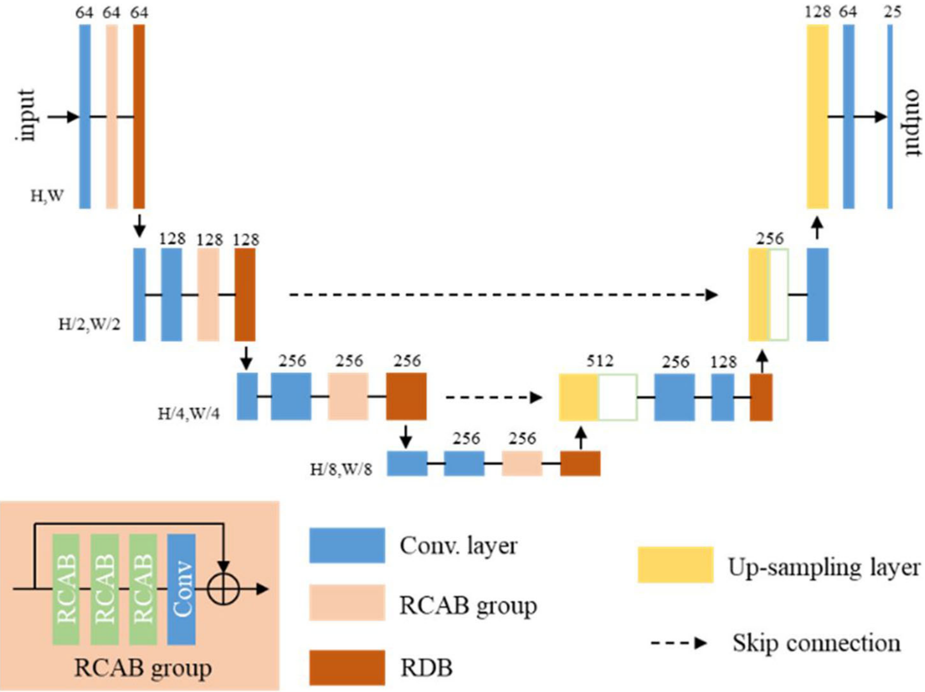

Fig. 2 The proposed pixel-wise kernel estimation network. H and W represent the image dimension. The number on each block represents the output channel size

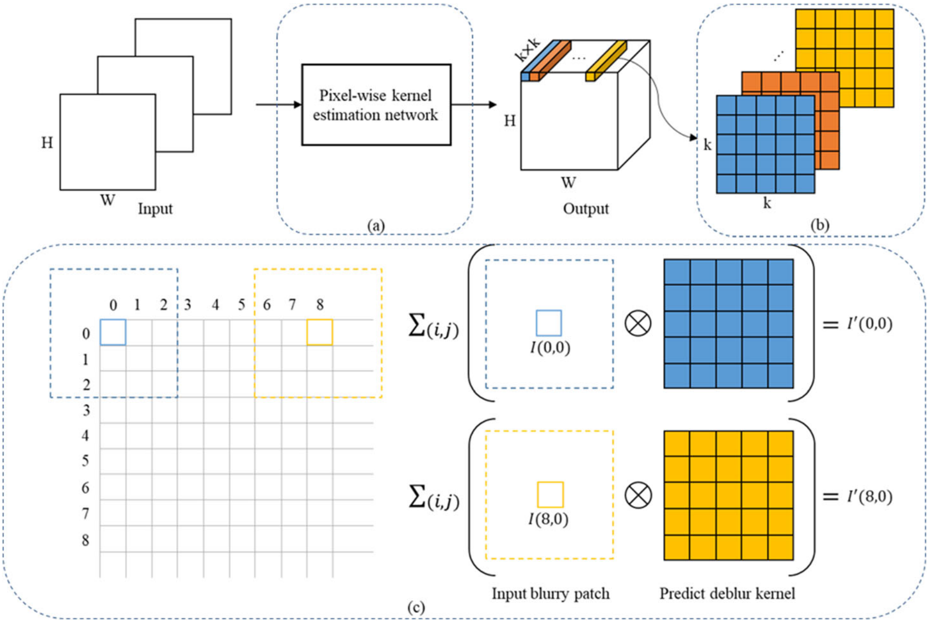

Fig. 3 **a** Pixel-wise kernel estimation network. **b** Pixel-wise deblur kernel. **c** Dynamic local filtering

kernel as position-specific. The pixel-wise deblur kernel in each position is the corresponding predicted deblur kernel of the blurry input image.

The proposed kernel estimation network does not convert the blurry input image into the latent deblur kernel directly at the beginning of the network, so we need to use the convolution layer to extract the blur information. To achieve better performance, we use the residual channel attention block (RCAB) [42] and residual dense block (RDB) [43] to extract the blur feature in the proposed kernel estimation network. The RCAB can select useful feature maps by the channel attention (CA) [42] mechanism, which can increase the weight of important channels, and suppress less important weights. By using the RCAB, the proposed kernel estimation network can adaptively choose the feature maps and reduce the impact of invalid feature maps. Finally, we combine three RCAB and one convolution layer and use the skip connection to compose the RCAB group to get better performance. In addition to use the RCAB, we connect an RDB after the RCAB to get stronger blur information from the extracted feature maps. The RDB not only can increase the depth of the network, but it can also concatenate every output feature map to all subsequent convolution layers to complement the information, since the information may be lost during the convolution operation. Since the predicted pixel-wise deblur kernel has different characteristic compared to the image, so in the decoder of the original dimension, the skip connection will not be applied.

The kernel estimation network directly uses the RCAB proposed in, as shown in Fig. 4. The RCAB structure is to combine the CA and residual blocks. We use the CA mechanism to achieve the feature selection. The residual learning strategy of residual blocks is widely used in deep neural networks by adding an identical mapping with the shortcut connections to alleviate the vanishing gradient problem.

The CA mechanism can adaptively rescale channel-wise features by considering interdependencies among channels, as shown in Fig. 5. In other words, it can adaptively select the feature maps by giving different weights to each feature map.

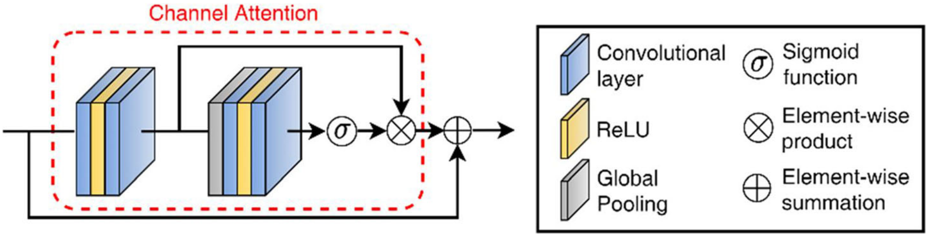

Fig. 4 Residual channel attention block [42]

Here we describe how to achieve the feature selection by the CA mechanism. First, we use the global average pooling function to acquire the channel-wise statistic global spatial information by Eq. (1).

$$z_c = H_{GP}(x_c) = \frac{1}{H \times W} \sum_{i=0}^{H} \sum_{j=0}^{W} x_c(i, j), \tag{1}$$

where x_c represents the input feature maps in the c-th channel, H_{GP} represents the global average pooling function, H and W represent the height and width of input feature maps, respectively.

After doing the average global pooling, we acquire the channel-wise statistic global spatial information, which can express the entire feature map of each channel in a single value. Then, we feed it into the channel down- and up-scaling layer to obtain the scaling factor of each channel, as shown in Eq. (2). The kernel size of the down- and up-scaling convolution layers are 1 × 1. The number of channels reduces by the ratio of r in the down-scaling layer, and then increases back to the original number in the up-scaling layer.

$$s = f(W_U \delta(W_D z)), \tag{2}$$

where f and δ represent the sigmoid and ReLU function, respectively; W_U and W_D represent the convolutional weight set of channels up- and down-scaling layer, respectively; The ratio r in the proposed method setting is 16.

Finally, we can acquire the weighted feature maps by multiplying the scaling factor with input feature maps by Eq. (3).

$$\widehat{x_c}(i, j) = s_c \times x_c(i, j), \tag{3}$$

where s_c represents the scaling factor in the c-th channel; x_c represents the input feature maps in the c-th channel.

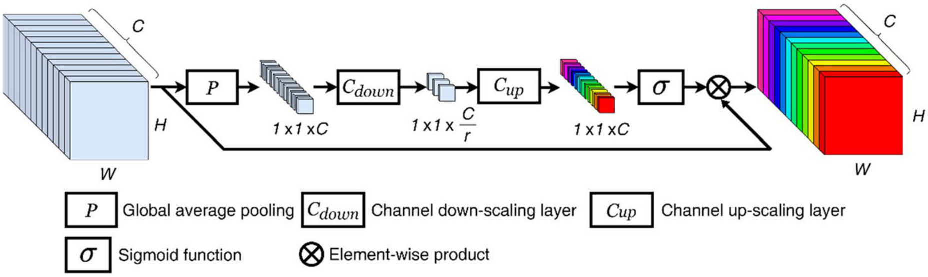

Fig. 5 Channel attention (CA) [42] mechanism

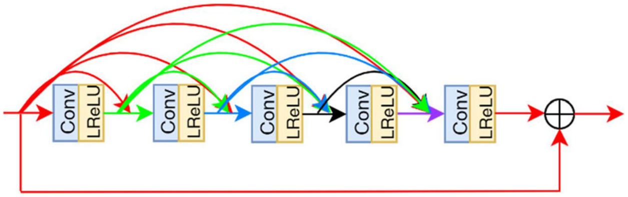

Fig. 6 Residual dense block in the proposed kernel estimation network

To further improve the performance of the proposed kernel estimation network, we use the RDB [43] structure, which is the combination of the residual block [21] and dense block [36]. The residual block can improve the network by the residual learning strategy. The dense block connects each layer to each layer in a feed-forward fashion, if there are *m* layers in a block, the total number of connections will be $m(m + 1)/2$. RDB combines the above advantages, which can alleviate the vanishing gradient problem, enhance feature propagation, and encourage feature reuses.

Due to the image degraded by the motion blur, the latent pixel information is scattered in a blurred image. We use the 3 × 3 convolution layer to replace the 1 × 1 convolution layer at the end of the RDB as the dimensionality reduction layer and to capture the useful information around the center pixel. The RDB we use in the proposed kernel estimation network is shown in Fig. 6. We use five convolution layers. Each convolution layer is followed by the

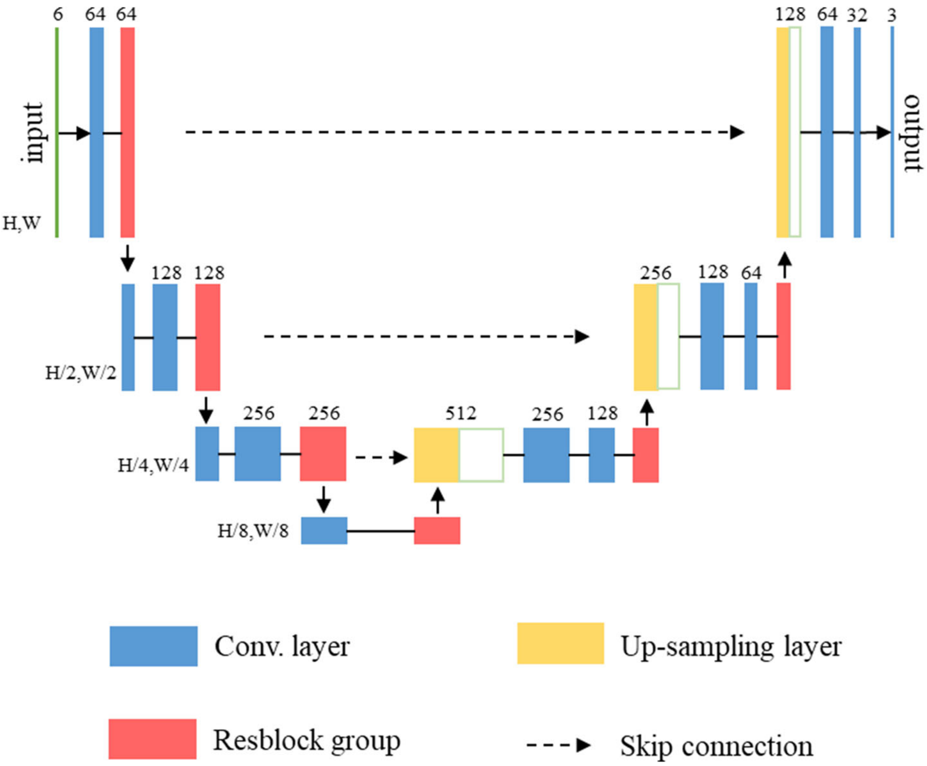

Fig. 7 The proposed image deblurring network. H and W represent the image dimension. The number on each block represents the output channel size

LeakyReLU layer to form the proposed residual dense block with a growth rate is 64, which means that the output of each convolution layer is 64 channels except for the output layer.

3.2 Image Deblurring network

The overall second stage architecture of the image deblurring network is shown in Fig. 7. The proposed image deblurring network can handle the large motion blur by using the encoder-decoder structure. The encoder-decoder structure reduces the feature maps resolution to enlarge the receptive field so that the neural network can learn the general direction of the image, but it is also easy to ignore the details of small places. Therefore, the final result is obtained through the cooperation of the two stages of the network. We concatenate the blurred input image with the pre-deblurring results as the input of the proposed image deblurring network. It can give the network a better starting point and reserve the original image information.

In order to improve the performance of the network, we stack the number of convolution layers to obtain richer feature expressions. But deep neural network will encounter the vanishing/exploding gradient problems. Because the gradient signal from the loss function changes exponentially when it propagates back to earlier layers, it is difficult or even impossible to converge. Therefore, we use the residual learning proposed in ResNet [10] to add a shortcut connection between convolution layers to solve the vanishing/exploding gradient problems. In addition, when the convolution layer does not learn new features, the shortcut connections can be treated as an identical mapping, it can ensure that the performance of the network does not deteriorate to avoid model degradation. The proposed residual block (Resblock) is shown in Fig. 8a. The Resblock group we use in the proposed image deblurring network consists of four Resblocks stacked, as shown in Fig. 8b.

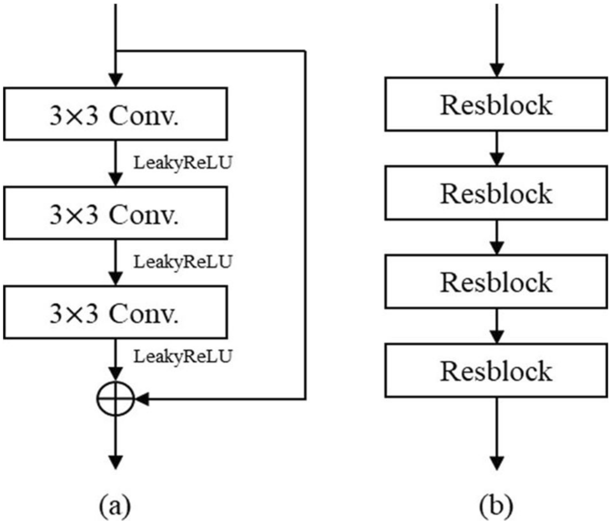

Fig. 8 **a** Resblock **b** Resblock group

3.3 Loss and training detail

In deep learning, we usually have multiple different modules, and each with its own function. We can choose to learn everything together or separately. The joint learning is to combine multiple different modules to train together to find the global optimum for the entire task.

We use joint learning to train the proposed image deblurring network, which is composed of two different networks to deal with different types of motion blur. In the first-stage network, we predict the pixel-wise deblur kernel to remove the small motion blur. In the second-stage network, we learn the blur-to-blur-free mapping function directly from the feature maps, which has a good ability to remove the large motion blur. Therefore, through joint learning, the first and second stages of the proposed network can compensate each other for better results. Figure 9 shows the overall architecture of the joint learning network.

The proposed network is trained by using four different loss functions, such as spatial loss ($\mathcal{L}_{spatial}$), spectral loss ($\mathcal{L}_{specteal}$), structural similarity index measure (SSIM) loss ($\mathcal{L}_{ssim}$), and gradient loss ($\mathcal{L}_{gradient}$). The overall loss function is shown in Eq. (4), and four loss functions are shown in Eq. (5)–(8).

$$\mathcal{L}_{total} = \sum_{i=1,2} \lambda_1 \times \mathcal{L}^i_{spatial} + \lambda_2 \times \mathcal{L}^i_{spectral} + \lambda_3 \times \mathcal{L}^i_{ssim} + \lambda_4 \times \mathcal{L}^i_{gradient} \tag{4}$$

where i is the i-th stage deblurring result; λ is the weight of different loss function.

$$\mathcal{L}^i_{spatial} = \| I^i_{out} - I_{gt} \|_1, \tag{5}$$

$$\mathcal{L}^i_{spectral} = \| \, |F(I^i_{out})| - |F(I_{gt})| \, \|_2, \tag{6}$$

$$\mathcal{L}^i_{ssim} = 1 - \mathrm{SSIM}(I^i_{out}, I_{gt}), \tag{7}$$

$$\mathcal{L}^i_{gradient} = \| G_{H,V}(I^i_{out}) - G_{H,V}(I_{gt}) \|_1, \tag{8}$$

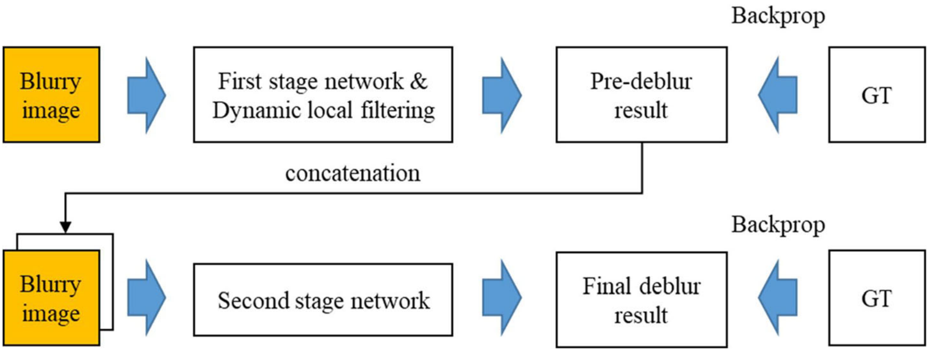

Fig. 9 Overall architecture of the proposed joint learning network

where I_{out} and I_{gt} are deblurring result and ground truth, respectively; $|F(\cdot)|$ is the magnitude of frequency coefficients; SSIM is image quality metrics; $G_{H,\,V}$ is operators that compute image gradient along the horizontal and vertical directions.

The proposed method is trained using GOPRO dataset [24]. In the data pre-processing step, we randomly crop the training patch to a size of 256×256, then randomly rotate and flip the patch, and finally adjust the saturation and hue of the patch. We use the Adam [16] optimizer to update the proposed network. The initial learning rate is set to $1e^{-4}$ with a linear decrease to zero after 20% of training, the batch size is set to 4.

4 Experimental results

The proposed method is implemented on the PyTorch framework. Both training and testing were conducted on the same PC with an NVIDIA GTX1080Ti GPU, and an Intel i7–4790@3.60 GHz CPU.

We evaluate our proposed method on three different test datasets to show the effectiveness of the proposed method. We use the same parameters, which are trained based on the GOPRO training set [24] to conduct on three different test datasets. The GORPO dataset [24] use GOPRO4 Hero Black camera to record the 240 fps videos. Then, researchers average the varying number of consecutive frames to the synthesized realistic blurry image. The sharp image corresponding to each blurry image is defined as the middle frame of the consecutive sharp frames that are used to make the blurry image. The GOPRO dataset [24] contains 2103 pairs for training and 1111 pairs for evaluation. For all the following experiments, we used the GOPRO training set [24] to train the proposed system, and tested on different dataset separately.

We use MATLAB for all PSNR, SSIM [38], and MS-SSIM [37] evaluations. The visual comparison with other methods has reproduced by the codes that were provided by the authors.

4.1 Evaluation methods

The PSNR of the deblurring result with the corresponding ground truth sharp image can be obtained by the Eqs. (9) and (10). A higher PSNR value indicates the result is closer to the ground truth sharp image.

$$\mathrm{MSE} = \frac{1}{3mn}\sum_{R,G,B}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left[I_{rgb}(i,j)-I'_{rgb}(i,j)\right]^2, \tag{9}$$

$$\mathrm{PSNR} = 10\cdot log_{10}\left(\frac{{I_{max}}^2}{MSE}\right), \tag{10}$$

where m and n are the image width and height, I and I' are the ground truth sharp image and the deblurring result, I_{max} is the maximum pixel value of the image.

The SSIM [38] metric is to compare the luminance (l), contrast (c), and structure (s) of two images, as shown in Eqs. (12) to (14). The SSIM [38] score of the deblurring result and the corresponding ground truth sharp image is a weighted combination of those comparative measures, where α, β, and γ are weight constants. As shown in eq. (11).

$$\mathrm{SSIM}\left(I,I'\right)=\left[l\left(I,I'\right)\right]^{\alpha}\left[\mathrm{c}\left(I,I'\right)\right]^{\beta}\left[\mathrm{s}\left(I,I'\right)\right]^{\gamma}, \tag{11}$$

where μ_I and $\mu_{I'}$ are the average of I and I'; σ_I and $\sigma_{I'}$ are the variance of I and I'; $\sigma_{II'}$ is the covariance of I and I'; C_1, C_2, and C_3 are constants.

$$l\left(I,I'\right)=\frac{2\mu_I\mu_{I'}+C_1}{\mu_I^2+\mu_{I'}^2+C_1}, \tag{12}$$

$$c\left(I,I'\right)=\frac{2\sigma_I\sigma_{I'}+C_2}{\sigma_I^2+\sigma_{I'}^2+C_2}, \tag{13}$$

$$s\left(I,I'\right)=\frac{\sigma_{II'}+C_3}{\sigma_I\sigma_{I'}+C_3}, \tag{14}$$

The MS-SSIM [37] metric is a multi-scale method to evaluate image details at different resolutions, as shown in Fig. 10. The MS-SSIM [37] of the deblurring result with the corresponding ground truth sharp image uses the Eq. (15).

$$\mathrm{MS{-}SSIM}\left(I,I'\right)=\left[l_M\left(I,I'\right)\right]^{\alpha_M}\cdot\prod_{j=1}^{M}\left[c_j\left(I,I'\right)\right]^{\beta_j}\left[s_j\left(I,I'\right)\right]^{\gamma_j}, \tag{15}$$

where M is scale index; c_j and s_j are contrast comparison (13) and structure comparison (14) at the j-th scale, l_M is the luminance comparison (12) only be computed at scale M; α_M, β_j, and γ_j, similar to (11), are used to adjust the relative importance of different components.

4.2 GORPO testing set

The GOPRO testing set [24] consists of 1111 blur and blur-free image pairs. The results of the proposed method can well restore the image details and text edges due to the use of deblur kernels and four loss functions, as shown in Fig. 11. Quantitative evaluation results with state-

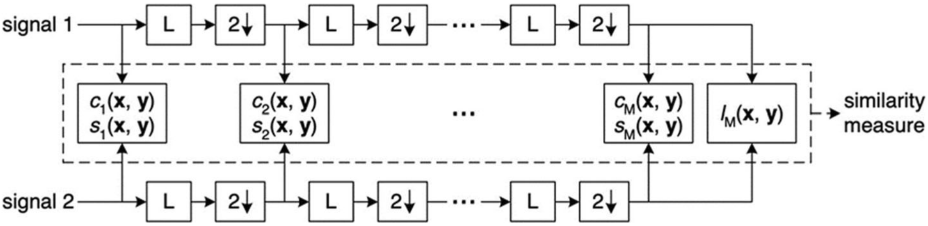

Fig. 10 MS-SSIM system, L: low-pass filtering; 2↓: downsampling by 2

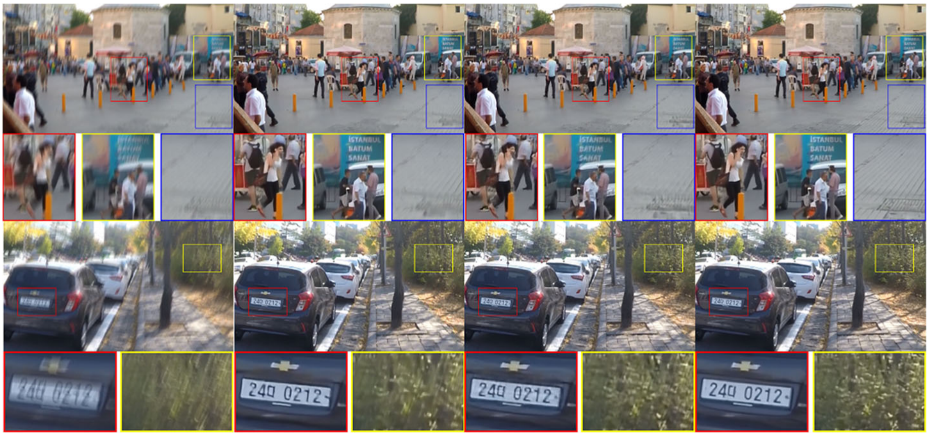

Fig. 11 Deblurring results on the GOPRO testing set [24]. First column: input blurry images. Second column: deblurring results by Nah et al. [24]. Third column: deblurring results by Tao et al. [34]. Fourth column: deblurring results by proposed method

of-the-art methods are shown in Table 1. The proposed method produces the best results in terms of PSNR and SSIM.

From the enlarged view of Fig. 11, we can observe the difference in many details. In the enlargement of the above results, we can see that the structure of the overall image is mostly reconstructed successfully. However, in scene details with indistinct boundaries by Nah et al. [24], shaking still cannot be eliminated. From lower yellow and upper blue rectangle, the green and black forest scene or the gray floor with thin black lines still retains a large degree of blur. Next, we can observe from lower red rectangle. Due to the inability to correct and synthesize details well, some parts of the license plate are also missing in the middle. Therefore, the details cannot be clearly restored, and only good picture restoration can be seen. Besides, in the indistinct-boundaries enlargement by Tao et al. [34], we can observe some details in the enlarged image. The gray floor and thin black lines in blue rectangle still retain a certain degree of blur. While it maintains a better structure to the object, the blur is also preserved and composited together. The license plate from the lower blue rectangle, a white line in the middle can be clearly seen, which looks like a white line produced by superposition. Finally, from the results of the proposed method, we can observe more successful deblurring and sharper detail restoration from the enlarged image. In addition to the number in the license plate being clearly

Table 1 Quantitative results on the GOPRO testing set [24]

Methods	PSNR	SSIM
Kim et al. [14]	23.64	0.8239
Sun et al. [33]	24.64	0.8429
Nah et al. [24]	29.08	0.9135
Tao et al. [34]	30.26	0.9342
Zhang et al. [44]	30.25	0.9351
Gao et al. [8]	30.92	0.9421
Lim et al. [22]	30.62	0.9388
Sim et al. [30]	31.34	0.9474
Ye et al. [40]	30.28	0.9046
Proposed method	**32.59**	**0.9589**

Table 2 Quantitative results on the Köhler dataset [17]

Methods	PSNR	MS-SSIM
Kim et al. [14]	24.68	0.7937
Sun et al. [33]	25.22	0.7735
Nah et al. [24]	26.48	0.8079
Tao et al. [34]	26.75	0.8370
Zhang et al. [44]	24.66	0.7639
Gao et al. [8]*	27.02	0.8387
Lim et al. [22]	27.02	0.8442
Sim et al. [30]*	25.87	0.8109
Ye et al. [40]	26.99	**0.8551**
Proposed method	**27.21**	0.8397

* indicates that we use the code provided by the author to evaluate the score

restored, the lines on the floor and the detailed scenery of the forest can also be clearly seen. Therefore, we can see the proposed method achieves the best results in PSNR and SSIM, which are 32.59 and 0.9589 respectively.

4.3 Köhler dataset

The Köhler dataset [17] is generated by recording trajectories of human camera shaking and then playbacks on a hexapod robot. This dataset consists of 48 blurry images. They provide their own evaluation code; Thus, we report the MS-SSIM [37] instead of the SSIM [38]. Table 2. lists the average PSNR and MS-SSIM values of our proposed method and the state-of-the-art methods for the Köhler dataset [17]. Figure 12 shows some deblurring results for subjective comparison.

From the enlarged view of Fig. 12, there are few differences can be observed from some of the details. In the enlargement of the above results on the Köhler dataset [17], we can see that the details and the structure of the overall image are mostly well reconstructed. However, some differences can be observed. From the results by Nah et al. [24], the enlarged image shows that blur can still be seen. While some details have been sharpened, blurring can still be observed in some figures or lines. The number 9 with several lines in lower red rectangle, show us the effect as a clear example. As for the results by Tao et al. [34], most of the details have been well reconstructed and sharpened. In lower yellow and red rectangle, both the Roman numerals and the number 9 on the clock have been relatively free of wobble and blur. Aside

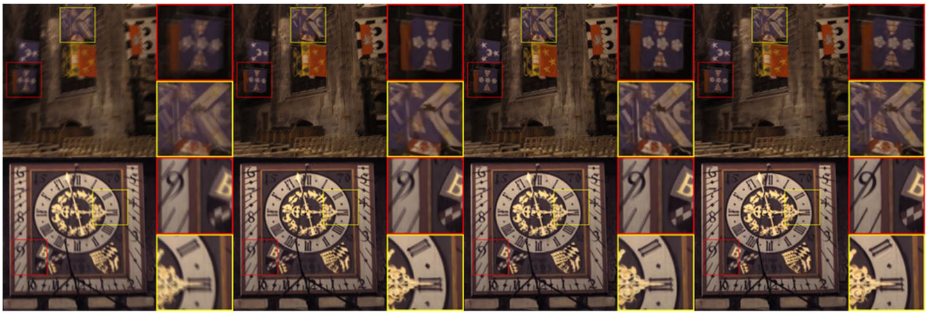

Fig. 12 Deblurring results on the Köhler dataset [17]. First column: input blurry images. Second column: deblurring results by Nah et al. [24]. Third column: deblurring results by Tao et al. [34]. Fourth column: deblurring results by proposed method

Table 3 Quantitative results on the Su dataset [31]

Methods	PSNR	SSIM
Nah et al. [24]*	29.73	0.9198
Tao et al. [34]*	31.09	0.9328
Sim et al. [30]*	30.64	0.9231
Proposed method	**31.96**	**0.9390**

The bold number means the best one among all methods
* indicates that we use the code provided by the author to evaluate the score

from some lines in the enlarged image, it can still be seen that there is still a slight degree of blurring. And finally, from the results of our method, it can be seen that basically the details are sharpened to some extent. Parts of the clock have no visible noticeable blur. But there are gains and losses, we can see that in the enlarged image from lower yellow rectangle, there are slight imprints similar to water ripples below the Roman numerals. Although it can be found in the enlarged image, it does not affect its clarity in the overall picture. Thus, we can see the proposed method achieves the best results in PSNR, which is 27.21. And SSIM, although not the best among them, also got a result of 0.8397.

4.4 Su dataset

The Su dataset [31] consists of 6708 blur and blur-free image pairs with a resolution of 1920 × 1080 or 1280 × 720. They collect 71 videos from multiple devices, i.e., iPhone 6 s, GoPro Hero 4, and Canon 7D, and generate the blurry image by accumulating a number of consecutive frames in specific conditions (the consecutive frames whose relative motions in-between are smaller than one pixel) to approximate a long exposure. We use all 6708 images for testing. Table 3 lists the average PSNR and SSIM [38] values of our proposed method and the state-of-the-art methods for the Su dataset [31]. Figure 13 shows some deblurring results for subjective comparison.

As we can see from Fig. 13, results by Nah et al. [24] provide a well reconstruction for the whole image. Most of the blur is removed, but the blur and shaking are still noticeable after

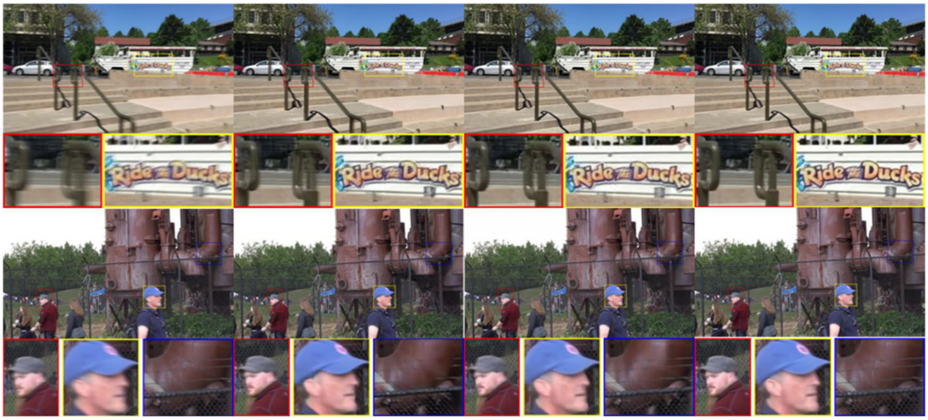

Fig. 13 Deblurring results on the Su dataset [31]. First column: input blurry images. Second column: deblurring results by Nah et al. [24]. Third column: deblurring results by Tao et al. [34]. Fourth column: deblurring results by proposed method

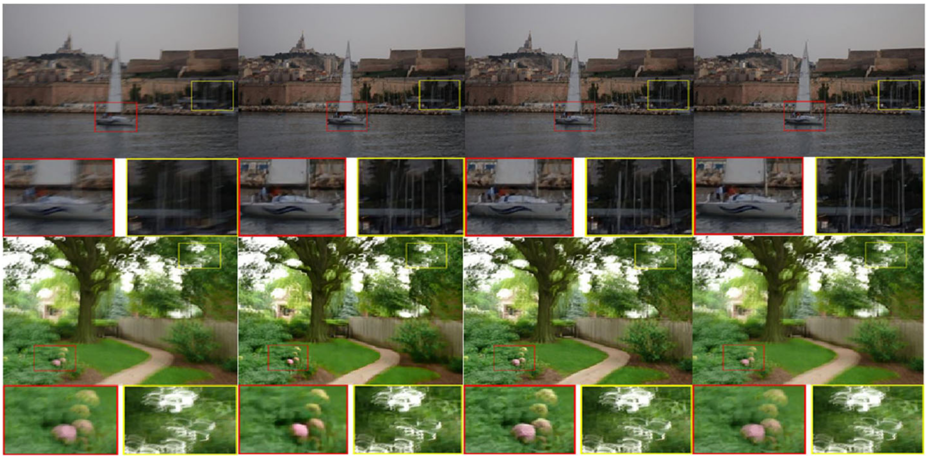

Fig. 14 Deblurring results on the Lai dataset [18]. First column: input blurry images. Second column: deblurring results by Nah et al. [24]. Third column: deblurring results by Tao et al. [34]. Fourth column: deblurring results by proposed method

zooming in on the image. As for results by Tao et al. [34] and our proposed method, we can see that the difference is not large, and the degree of blurring is similar. The only difference is a slight difference in colour and brightness, as well as the sharpening of some details. Therefore, this is also the reason why the PSNR and SSIM values on both sides are similar to other methods. But still, the proposed method achieves the highest scores in PSNR and SSIM, which are 31.96 and 0.9390 respectively.

4.5 Lai dataset

The Lai dataset [18] has the real image dataset and the synthetic dataset, each with 100 blurry images. The real image dataset is generated by capturing the real-world scenarios from

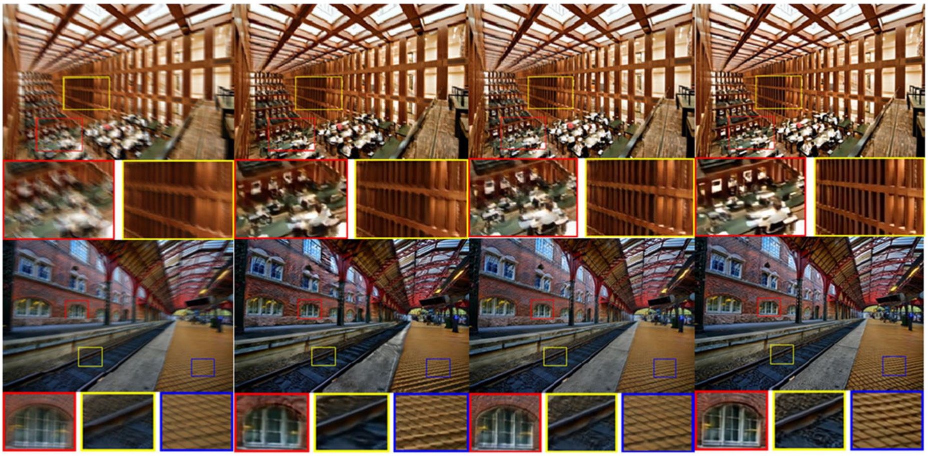

Fig. 15 Deblurring results on the Lai dataset [18]. First column: input blurry images. Second column: deblurring results by Nah et al. [24]. Third column: deblurring results by Tao et al. [34]. Fourth column: deblurring results by proposed method

different cameras. The synthetic dataset is generated by convolving nonuniform blur kernels on sharp images and imposing several common degradations. However, the blurry images and sharp images are not aligned, making the PSNR and SSIM [38] metrics are less correlated with perceptual quality. Therefore, the calculated PSNR/SSIM is meaningless i.e., the deblurring results of our method and Tao et al. [37] have better visual quality but got a lower score in quantitative evaluation. Figure 14 show some deblurring results of the real image dataset, and Fig. 15 show some deblurring results of the synthetic dataset for subjective comparison.

In the real image dataset, it is more challenging to remove the real blur. We failed to restore sharp information in some real blurry images because the blur caused by real camera shake is more complicated. We show a successful result and a failure result in Fig. 14. As shown in upper Fig. 14, our result can restore the sharp information well in the real blurry image, but all methods failed to remove the blur in lower Fig. 14. In the synthetic dataset, our deblurring result performs better than other methods. As shown in upper Fig. 15, our method can restore the sharper edges than other methods on the blurry image generated by the blur kernel. As shown in lower Fig. 15, our method can even restore the details of the railway (stone texture) in the yellow rectangle region, while the other three methods failed.

5 Conclusions

This article proposed a single image deblurring method to directly restore the sharp image from a blurry input image. The proposed deblurring method is featured with the two stages deblurring network to deal with different types of motion blur. The first stage network is used to restore the small motion blur, and the second stage network is used to restore the large motion blur. In order to efficiently combine the two stage networks, we use a jointly learning strategy to train both networks simultaneously.

We used some image processing techniques, i.e., random cropping, rotating, and flipping, and adjusting colour saturation and hue, to augment the training dataset so that the proposed network can be adapted to a variety of different blur scenarios without overfitting. We also used four different loss functions to provide the proposed network with more information to reach a better deblurring result.

To show the effectiveness of the proposed method we evaluate on three different benchmark datasets, which synthetic blurred images using different methods. These datasets generated blur and blur-free image pairs by averaging the different number of consecutive frames captured by high-frame-rate cameras, using the hexapod robot to playback camera shaking trajectories, convolving blur kernel on the sharp image, and capturing the real-world blur image. The experimental results show that the proposed deblurring result outperformed the state-of-the-art methods in PSNR, SSIM, and MS-SSIM evaluation for all the aligned benchmark datasets. Besides, the results of the proposed method show better visual quality compared with the state-of-the-art methods for all benchmark datasets.

In the future, we believe that the blur dataset needs to be more diverse to simulate a variety of real blurs. When using continuous frames to synthesis blurred images, different blocks can be selected to average or discard some frames, because there is not such richness information in real blurred images. Moreover, we would like to make the model smaller and implement the deblurring system on other platforms, i.e., camera or mobile device can bring more convenience to users. And also, we would try to use the proposed deblurring method as a preprocessing step for other high-level image processing applications, such as object recognition, segmentation, and classification, to improve its accuracy.

Acknowledgements This study was funded by the ministry of Science and Technology in Taiwan under MOST 108-2221-E-011-130-.

The authors would like to thank Mr. Ding-Wei Hu, who participated in revising the manuscript.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Data availability The datasets analyzed during the current study are available from references [17, 24, 31, and [18], respectively.

References

1. Ahmed AH, Zou Q, Nagpal P, Jacob M (n.d.) Dynamic Imaging using Deep Bi-linear Unsupervised Representation (DEBLUR). accepted by IEEE Trans Med Imag. https://doi.org/10.1109/TMI.2022.3168559
2. Anwar S, Barnes N (2019) Real Image Denoising with Feature Attention. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp 3155–3164
3. Brooks T, Mildenhall B, Xue T, Chen J, Sharlet D, Barron JT (2019) Unprocessing Images for Learned Raw Denoising. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 11028–11037
4. Cho S, Lee S (2009) Fast motion deblurring. In: ACM SIGGRAPH Asia 2009 papers, pp 1–8
5. Ciregan D, Meier U, Schmidhuber J (2012) Multi-Column Deep Neural Networks for Image Classification. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)., pp 3642–3649
6. Dai T, Cai J, Zhang Y, Xia S, Zhang L (2019) Second-Order Attention Network for Single Image Super-Resolution. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 11057–11066
7. Fergus R, Singh B, Hertzmann A, Roweis ST, Freeman WT 2006 Removing camera shake from a single photograph. In: ACM SIGGRAPH 2006 Papers, pp 787–794
8. Gao H, Tao X, Shen X, Jia J (2019) Dynamic scene deblurring with parameter selective sharing and nested skip connections. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 3848–3856
9. Gupta A, Joshi N, Zitnick CL, Cohen M, Curless B (2010) Single image deblurring using motion density functions. In: European conference on computer vision, Springer pp 171–184
10. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
11. Hirsch M, Schuler CJ, Harmeling S, Schölkopf B (2011) Fast removal of non-uniform camera shake. In: 2011 International Conference on Computer Vision: IEEE, pp 463–470
12. Jia X, De Brabandere B, Tuytelaars T, Gool LV (2016) Dynamic filter networks. Adv Neural Inf Proces Syst 29:667–675
13. Jo Y, Oh SW, Kang J, Kim SJ (2018) Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3224–3232
14. Kim TH, Lee KM (2014) Segmentation-free dynamic scene deblurring. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp 2766–2773
15. Kim TH, Ahn B, Lee KM (2013) Dynamic scene deblurring. In: Proceedings of the IEEE International Conference on Computer Vision, pp 3160–3167
16. Kingma DP, Ba J (2014) Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980
17. Köhler R, Hirsch M, Mohler B, Schölkopf B, Harmeling S (2012) Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database. In: European conference on computer vision,: Springer, pp 27–40
18. Lai W-S, Huang J-B, Hu Z, Ahuja N, Yang M-H (2016) A comparative study for single image blind deblurring. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1701-1709
19. Ledig C, Theis L, Huszár F, Caballero J, Cunningham A, Acosta A, Aitken A, Tejani A, Totz J, Wang Z, Shi W (2017) Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 105–114

20. Liang CH, Chen YA, Liu Y-C, Hsu WH (2022) Raw image Deblurring. IEEE Trans Mult 24:61–72. https://doi.org/10.1109/TMM.2020.3045303
21. Lim B, Son S, Kim H, Nah S, Lee KM (2017) Enhanced deep residual networks for single image super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp 136–144
22. Lim S, Kim J, Kim W (2020) Deep Spectral-Spatial Network for Single Image Deblurring. IEEE Signal Processing Letters (SPL), vol 27. pp 835–839,
23. Milletari F, Navab N,Ahmadi S (2016) V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. In: 2016 Fourth International Conference on 3D Vision (3DV), 25–28 pp 565–571
24. Nah S, Kim TH, Lee KM (2017) Deep Multi-Scale Convolutional Neural Network for Dynamic Scene Deblurring. In : 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp 257–265
25. Pan J, Hu Z, Su Z, Lee H-Y, Yang M-H (2016) Soft-segmentation guided object motion deblurring. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 459–468
26. Ronneberger O, Fischer P, Brox T (2015) U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention: Springer, pp 234–241
27. Schuler CJ, Hirsch M, Harmeling S, Schölkopf B (2016) Learning to Deblur. IEEE Trans Patt Anal Mach Intell (TPAMI) 38(7):1439–1451
28. Shan Q, Jia J, Agarwala A (2008) High-quality motion deblurring from a single image. Acm Trans Graphics (tog) 27(3):1–10
29. Shin H, Roth HR, Gao M, Lu L, Xu Z, Nogues I, Yao J, Mollura D, Summers RM (2016) Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. IEEE Trans Med Imaging 35(5):1285–1298
30. Sim H, Kim M (2019) A Deep Motion Deblurring Network Based on Per-Pixel Adaptive Kernels with Residual Down-Up and Up-Down Modules. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 16–17.pp 2140–2149
31. Su S, Delbracio M, Wang J, Sapiro G, Heidrich W, Wang O (2017) Deep video deblurring for hand-held cameras." In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1279–1288
32. Suin M, Purohit K, Rajagopalan AN (2020) Spatially-Attentive Patch-Hierarchical Network for Adaptive Motion Deblurring. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 3603–3612
33. Sun J, Wenfei C, Zongben X, Ponce J (2015) Learning a Convolutional Neural Network for Non-Uniform Motion Blur Removal. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 7–12 pp 769–777
34. Tao X, Gao H, Shen X, Wang J, Jia J (2018) Scale-Recurrent Network for Deep Image Deblurring. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 8174–8182
35. Tien H, Yang H, Shueng P, Chen J (2021) Cone-beam CT image quality improvement using Cycle-Deblur consistent adversarial networks (Cycle-Deblur GAN) for chest CT imaging in breast cancer patients. Sci Rep 11(1):1133. https://doi.org/10.1038/s41598-020-80803-2
36. Tong T, Li G, Liu X, Gao Q (2017) Image super-resolution using dense skip connections. In: Proceedings of the IEEE international conference on computer vision,, pp 4799–4807
37. Wang Z, Simoncelli EP, Bovik AC (2003) Multiscale structural similarity for image quality assessment. In: The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, vol. 2: Ieee, pp 1398–1402
38. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process 13(4):600–612
39. Yang W, Tan RT, Feng J, Liu J, Guo Z, Yan S (2017) Deep joint rain detection and removal from a single image. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR), pp 1685-1694
40. Ye M, Lyu D, Chen G (2020) Scale-iterative upscaling network for image Deblurring. IEEE Access 8: 18316–18325. https://doi.org/10.1109/ACCESS.2020.2967823
41. Zhang K, Zuo W, Chen Y, Meng D, Zhang L (2017) Beyond a Gaussian Denoiser: residual learning of deep CNN for image Denoising. IEEE Trans Image Process (TIP) 26(7):3142–3155
42. Zhang Y, Li K, Li K, Wang L, Zhong B, Fu Y (2018) Image super-resolution using very deep residual channel attention networks. In: Proceedings of the European conference on computer vision (ECCV), pp 286–301
43. Zhang Y, Tian Y, Kong Y, Zhong B, Fu Y (2018) Residual dense network for image super-resolution.In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2472–2481

44. Zhang H, Dai Y, Li H, Koniusz P (2019) Deep stacked hierarchical multi-patch network for image deblurring. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 5978–5986
45. Zhang H, Sindagi V, Patel VM (2020) Image De-raining using a conditional generative adversarial network. IEEE Trans Circuits Syst Video Technol (TCSVT) 30(11):3943–3956
46. Zou W, Jiang MO, Zhang Y, Chen L, Lu Z, Wu Y (2021) SDWNet: A Straight Dilated Network with Wavelet Transformation for image Deblurring. In: 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), Montreal, BC, Canada, pp 1895-1904

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.