Progressive edge-sensing dynamic scene deblurring

Tianlin Zhang^{1,3}, Jinjiang Li² (✉), and Hui Fan²

© The Author(s) 2021.

Abstract Deblurring images of dynamic scenes is a challenging task because blurring occurs due to a combination of many factors. In recent years, the use of multi-scale pyramid methods to recover high-resolution sharp images has been extensively studied. We have made improvements to the lack of detail recovery in the cascade structure through a network using progressive integration of data streams. Our new multi-scale structure and edge feature perception design deals with changes in blurring at different spatial scales and enhances the sensitivity of the network to blurred edges. The coarse-to-fine architecture restores the image structure, first performing global adjustments, and then performing local refinement. In this way, not only is global correlation considered, but also residual information is used to significantly improve image restoration and enhance texture details. Experimental results show quantitative and qualitative improvements over existing methods.

Keywords image deblurring; dynamic scenes; multi-scale; edge features

1 Introduction

Blind deblurring of dynamic scenes is a basic low-level ill-posed inverse task in computer vision, its purpose being to recover sharp images from blurred images, with or without estimation of the unknown non-uniform blurring kernel. Proposed solutions to this problem may be based on traditional image processing

or neural networks. One approach simplifies the problem by approximating the non-uniform blurring by uniform blurring, and restores the ground truth image and blurring kernel. However, due to the irregular motion offset trajectory in space, this cannot be generalized to true blurring. Therefore, much research has considered non-uniform blurring [1–5], and extended the degrees of freedom of the blurring model from uniform to non-uniform. In order to limit the solution space for non-uniform blurring, many natural image priors [6–8] have been proposed for regularization, but they still only consider non-uniform blurring caused by simple camera rotation and in-plane translation, which are certainly not enough to provide sharp images and blurring kernels.

Image quality degradation caused by blurring can be represented by a mathematical model:

$$B = KS + n \quad (1)$$

where B and S are the blurred image and the sharpened image respectively. K is an unknown or known blurring kernel, that is, a blurring matrix. Each row is a local blurring kernel, which is combined with the sharp image to generate blurred pixels. n is additive noise. Since deblurring has a large solution space, K or B is generally constrained to simplify the problem of finding of S . In the past, traditional dynamic scene deblurring was generally done by using accurate image segmentation [9] or motion estimation [10]. Kim et al. proposed joint segmentation of non-uniform blurred images based on an energy model. They estimate the nonlinear blurring kernel within segments and use parameter sharing. A groundbreaking addition of a non-static background reduces dynamic scene deblurring into a local deblurring problem. However, introducing other additional data processing causes the blurring kernel originally estimated to deviate further from the true blur kernel. Errors in the estimated blurring

1 School of Information and Electronic Engineering, Shandong Technology and Business University, Yantai 264005, China. E-mail: zhangtianlin@sdtbu.edu.cn.

2 School of Computer Science and Technology, Shandong Technology and Business University, Yantai 264005, China. E-mail: J. Li, lijnjiang@sdtbu.edu.cn (✉); H. Fan, fanhui@sdtbu.edu.cn.

3 Institute of ZhongKe Network Technology, Yantai 264005, China.

Manuscript received: 2021-06-04; accepted: 2021-06-13

kernel cause undesirable ringing artifacts. In order not to add redundant information, Kim and Lee approximated the blurring kernel as locally linear, so that the motion flow and latent image can be estimated at the same time. Thus, a non-segmented method was proposed to deal with this problem.

With the rapid development of deep learning, neural networks have been widely used in image processing [11–13]. For the problem of image deblurring, they were first used for non-blind deconvolution [14]. Xu et al. [15] removed blurring by restoring the sharp image with a given blurring kernel. They used separable kernels that can be decomposed into filters to form a deconvolution CNN. In Ref. [16], coarse-to-fine stacked multiple CNNs are used to simulate iterative optimization, where a blurred image synthesized by a uniform blurring core is used for training due to lack of pairs of real blurred images and corresponding sharp images. In Ref. [17], a classification convolutional neural network is used to estimate the local linear blurring kernel.

In recent years, people have proposed parametric models based on deep convolutional neural networks (CNNs) to replace image formation models. In order to obtain pairs of blurred images and sharp images for network training, Schuler et al. [16] used Gaussian blurring of sharp images from the ImageNet dataset [18], and proposed a blind deblurring algorithm based on a CNN. Feature extraction, kernel estimation, and latent image estimation steps were carried out in a coarse-to-fine iterative manner. The method in Ref. [19] predicted the deconvolution kernel in the frequency domain; both blurring kernel prediction and image prior are based on early learning methods. The models generated by these methods can be trained to simulate the nonlinear relationship between blurred images and ground truth, effectively overcoming the limited representative ability of traditional image processing methods to describe dynamic scenes.

Generative adversarial networks (GANs) are also used for deblurring because they have advantages in preserving detailed edges and generating approximate images. Kupyn et al. [20] used CNNs as a generator and discriminator, and calculated the loss through content loss and adversarial loss. On this basis, they improved the network [21], and proposed a generative confrontation network based on a feature pyramid network and a discriminator with least square loss.

In order to use fine image information as a feature aid, a multi-scale network structure is used for blind image deblurring, extending the coarse-to-fine scheme to deep CNN methods [22–24], which first restore the potentially sharp image at the coarse scale, and then use it for the fine scale. In addition to the recent use of independent feature extraction layers on network structures of different scales [23], there have been works on sharing network parameters in multi-scale pyramids [24] or other ways of sharing parameters [22]. However, the use of deep network deblurring still has great challenges before it can be applied. In order to inherit ideas from traditional coarse-to-fine optimization methods, most multi-scale networks use a large number of training parameters. Even if the number of parameters is reduced by parameter sharing, skip connections, and other methods, multi-scale methods have two main limitations.

Firstly, in order to maintain the integrity of blurred edges of objects and sensitivity to large-scale blurring, the network generally employs a large filter and excessively stacked convolutional layers. But this comes at the cost of the number of parameters and the speed of inferencing: the calculation cost and memory consumption are greatly increased.

Secondly, past experiments have shown that in multi-scale modules, the introduction of coarser or finer scale space inputs to further train model parameters does not improve the overall deblurring performance of models: simply increasing the spatial scale does not lead to better results.

Taking current deep learning deblurring approaches into account, this paper proposes a method of progressive dynamic scene deblurring. We combine a multi-scale architecture and an encoder–decoder structure to construct a nonlinear function, and then use the residual information in the image to further optimize the results, to overcome the above limitations. Multi-scale methods are widely used to recover sharp images from blurred images in a coarse-to-fine manner. It is difficult for a single deep network to directly generate sharp images from severely blurred images. We speculate that it is much easier to recover a sharp latent image from a slightly blurred image than from a strongly blurred image. Recently, Park et al. [25] verified this idea using a benchmark dataset and an iterative method. Therefore, the task is treated as a gradual process,

usually including two stages: the first step is to use a large filter to generate a large receptive field, restoring the area with a larger degree of blurring, and generating a coarse initial deblurred image. The second step refines the texture structure in the image to give the final output.

Taking this into account, we have designed a progressive multi-scale edge-sensing residual network (PMERN). It consists of two units, an information integration unit (IIU) and a detail optimization unit (DOU). In the IIU, the entire network uses a modified encoder/decoder architecture. We splice the salient edges of the blurred image into the encoder as an auxiliary branch to help the network accurately locate the blurred area. In the decoder, we try to change the single operation mode of deconvolution in the network, and then fuse multi-scale blurred images in the decoder to simulate the process of restoring sharp images at different scales. The weight is automatically adjusted according to the blur features contained in the blurred image. Therefore, in the deep convolution, the shallow feature information will not be lost due to the excessive number of encoder layers. This significantly simplifies the training process and brings obvious stability gains. In the DOU, edge structures are further refined by learning the residual image. By using multiple blur features of different dimensions for residual learning, the network deblurring effect is improved.

There are two advantages of this network. Firstly, since we merge images of different resolutions in the deconvolution network, the training time is much less than for a scale cascade structure. Secondly, compared to a scale recursive structure, no special parameter sharing method is needed.

Our main contributions are as follows: (i) A new solution overcoming the limitations of the current depth deblurring model which stacks depth and loop iteration. Compared to the previous fixed-level architecture, our network is more flexible. (ii) Changing the traditional multi-scale method. There is no need to explicitly train the network with images at different scales as input to the scale cascade; it perfectly combines the multi-scale architecture and the network structure. (iii) A comprehensive qualitative and quantitative evaluation using benchmark and real datasets shows this method to be better than the best existing

dynamic scene deblurring methods, yet using the fewest parameters: see Table 1.

Table 1 Comparison of training time (hours) and model parameters (kB) for our method and other classic methods

Method	Nah et al.	Tao et al.	Zhang et al.	Ours
Training time (h)	656	213	195	149
Parameters (kB)	45,844	26,880	3397	1401

2 Related work

In this section, we briefly review the main applications of multi-scale concatenation and encoder–decoder methods to image deblurring. In blind deblurring methods, both the blurring kernel and the image have certain a priori assumptions. However, these methods have little effect on large-scale blurring nuclei. The method proposed by Fergus et al. [26] completely abandons the constraints on image prior hypotheses. It has two main steps: estimation of the convolution kernel and deblurring. An in-depth study of the gradient distribution of blurred images and non-blurred images is provided, and a deblurring algorithm is proposed based on the gradient distribution model. This method introduces a coarse-to-fine strategy in traditional deblurring. On this basis, almost all traditional methods based on energy optimization repeatedly deal with the problem of dynamic deblurring, optimizing from a coarse scale and gradually expanding upwards between iterations until it reaches full scale.

2.1 Multi-scale

The multi-scale structure is designed to imitate the traditional coarse-to-fine optimization method. Because this method does not participate in the estimation of the blur kernel, the artifacts caused by kernel estimation error are avoided. Nah et al. [23] proposed a coarse-to-fine neural network to eliminate ambiguity. After the Gaussian pyramid structure, the coarse scale features are used to deblur the fine scale images. At the same time, to speed up model convergence, multi-scale loss and adversarial loss are added to each scale. This method establishes a deep neural network with independent parameters, which leads to the problem of excessive network parameters. To improve the network and simplify the network layer and parameters, Tao et al. [24] proposed a scale recursive network with long- and short-term memory,

using a codec structure with skip connections of different scales and parameter sharing. Due to the large filter size, a large number of training parameters are used in the network, and adding low-resolution input to the multi-scale method does not improve the deblurring performance. Zhang et al. [27] proposed a deeply stacked multi-scale patch network, which takes a multi-scale patch structure as input and refines the entire image through a continuous upper layer. However, VGG16 is used as the weight generation network, which increases the number of network parameters and amount of calculation. In order to further reduce network parameters, Ref. [22] replaced the residual blocks in the subnets of each scale with the nested skip connection structure of the nonlinear transformation module. The network components are composed of three modules: feature extraction, nonlinear transformation, and feature reconstruction.

2.2 Encoder–decoder

Codec structure [28] is a neural network design pattern. It is often used in natural language processing or other sequence-to-sequence prediction tasks. Specifically, the task of the encoder is to obtain a feature map of the input image through neural network learning, and to classify and analyze the pixel values of the low-level regions of the image. The decoder then takes this feature map as input and maps it to the output image. The codec structure has also been successfully applied to various image processing problems, using a symmetrical structure to first compile the input data into a small-size, multi-channel feature map, and then decode the feature map into an output with the same shape as the input. Ronneberger et al. [29] added skip connections between the corresponding feature maps in the encoder and decoder to improve regression capabilities. First, the structure in Ref. [24] is applied to image deblurring. Because the number of layers of the original codec is small, the perceptive field is small. If the number of layers is increased blindly, the size of the feature map will be too small to fully retain spatial information, and number of parameters will increase. Therefore, the author improved the codec, and enlarged the perceptive field, to better recover severe motion blur. Gao et al. [22] shared selective parameters on the basis of the codec for the change of blur image characteristics at coarse spatial scale. Ye et al. [30] proposed a scale-iterative upscaling

network. The model is divided into two layers, each layer using a different U-Net as a sub-module. The first layer performs deblurring operations at a relatively small scale to help the second layer perform large-scale deblurring operations. Similarly, we use an encoder–decoder network to restore the image structure.

3 Network architecture

3.1 Overview

In this section, we first introduce the proposed deblurring network architecture in detail. In addition, the encoder–decoder and salient edge training are described. The purpose of the deep learning network is to learn the end-to-end non-linear mapping between the blurred image and the corresponding sharp image with the assistance of information from the salient edge image, using a progressive processing flow for efficient and precise blind deblurring. In deblurring, the blurred image deviates greatly from the original image in both content and texture, so we cannot directly use a typical residual learning structure. The key idea of the progressive multi-scale residual network (PMRN) is to first reduce the degree of blurring. This operation generates an initial deblurred image with roughly the same structure as the ground truth. We then extract the subtle information by learning the initial result and the residual image from the sharp image. In this way, richer details can be restored to the finally reconstructed sharp image. The entire network can thus be divided into two stages, as shown in Fig. 1.

The first stage IU consists of three parts, namely, the backbone, the prominent edge pyramid branch, and the blurred image multi-layer guide branch. The backbone is composed of an optimized encoder–decoder, which takes a blurred image as input, extracts content features and blur features, and maps them to the output image. The significant edge pyramid branch (SEPB) takes the significant edge map corresponding to the blurred image as input for convolutional down-sampling, and sends the feature map to the encoder for feature extraction. The image multi-layer guided branch (IMGB) takes the blurred image as input for convolutional down-sampling, and sends the feature map to the decoder for deconvolution. As the first stage of image deblurring, this output generates an initial set of sharp latent images.

Fig. 1 Proposed progressive multi-scale edge-sensing residual network.

The output of the IIU is fed back to the second stage DOU and used as the input of the DOU together with the blurred image. We extract structural details by multiplexing residual blocks and allow the network to learn the residual difference between the numerically similar blurred and sharp images. The difference in structural similarity between the initial deblurred image and the sharp image is small, so we chose to loop the residual block. As network parameters mainly come from the reconstruction block, the recursive calculation within the stage can reduce the network complexity. At the same time, removing batch normalization after the convolutional layer of the classic residual block structure helps to improve convergence speed and maintain higher flexibility of the CNN [23,31]. Collaboration of these two stages provides a sharp image.

3.2 Information integration unit

3.2.1 Encoder-decoder architecture

Recently, the successful encoder-decoder structure has also been applied to various computer vision tasks, but the classic encoder-decoder structure is unsuitable for deblurring tasks. However, our fusion encoder-decoder network amplifies the advantages of various CNN structures and helps in training.

First of all, due to the cause of motion blur, a larger receptive field is required for image deblurring. However, this will increase the number of parameters. In addition, if there are too many intermediate feature channels, the feature image will be convolved

to a very small size, making it difficult to extract the features. Secondly, the spatial scale of the feature map in the decoder gradually increases, and detailed information of the original image will be lost, which is not conducive to image reconstruction. So we have adapted the encoder-decoder structure to the needs of the deblurring task. In order to prevent the disappearance of the gradient, we add a skip connection in the encoder and the symmetrical decoder, the purpose being to transfer corresponding features from the encoder branch to the decoder branch. Skip connections can solve the problem of gradient disappearance in deep network layers, and at the same time help backpropagation of the gradient and speed up training.

3.2.2 Significant edge pyramid branch

Motion blur in dynamic scenes often occurs at the edges of moving objects or at the edges of background objects due to camera shake. Blurred edge features carry a lot of positional information for the network to quickly extract feature maps, and better locate the blur areas. Considering the fact that strong edge information is important for reliable deblurring, we use the Sobel operator to extract the salient edges in the image, as shown in Fig. 2.

We use a $2 \times$ convolution kernel and a maximum pooling layer with a step size of 2 to gradually obtain the down-sampled significant edge map. For the salient edge pyramid, we extract the hierarchical representation through the convolutional layers.

Fig. 2 Salient edges.

Finally, the multi-scale spatial features are spliced to the encoder path.

$$E_{\text{ed}}^1 = \text{maxpool}(E_{\text{e}}^{\text{up}}) \quad (2)$$

$$F_{\text{ed}}^1 = \sigma(W_{\text{ed}}^1 * E_{\text{ed}}^1 + b_{\text{ed}}^1) \quad (3)$$

$$E_{\text{ed}}^{i+1} = \text{maxpool}(E_{\text{ed}}^i) \quad (4)$$

$$F_{\text{ed}}^{i+1} = \sigma(W_{\text{ed}}^{i+1} * E_{\text{ed}}^{i+1} + b_{\text{ed}}^{i+1}) \quad (5)$$

where $i \in \{1, 2, 3\}$, E_{ed}^1 is the salient edge map obtained by down-sampling E_{e}^{up} , F_{ed}^1 is the feature extracted from E_{ed}^1 , $*$ represents the convolution operation, and maxpool is the maximum pooling operation. W_{ed}^1 and b_{ed}^1 represent the weight and bias of the first convolution operation in the significant edge pyramid branch respectively, σ is the activation function of the modified linear unit, E_{ed}^{i+1} is the significant edge map obtained by downsampling E_{ed}^i , F_{ed}^{i+1} are the features extracted from E_{ed}^{i+1} , and W_{ed}^{i+1} and b_{ed}^{i+1} represent the weight and bias in the $(i+1)$ -th convolution operation respectively.

Advantages of significant edge branching are firstly, enhanced sensitivity of the network to blurred areas of blurred images, and secondly, the encoder extracts more detailed information for easier analysis.

3.2.3 Image multi-layer guided branch

Unlike the multi-scale approach in previous methods, we guide image reconstruction by down-sampling the original blurred images at different spatial scales and sending them into the decoder. The multi-layer branch of the blurred image can be expressed as

$$F_{\text{ms}}^1 = \sigma(W_{\text{ms}}^1 * B + b_{\text{ms}}^1) \quad (6)$$

$$B_{\text{ms}}^j = \text{maxpool}(F_{\text{ms}}^j) \quad (7)$$

$$F_{\text{ms}}^{j+1} = \sigma(W_{\text{ms}}^{j+1} * B_{\text{ms}}^j + b_{\text{ms}}^{j+1}) \quad (8)$$

where $j \in \{1, 2, 3\}$, F_{ms}^1 are the features extracted from the blurred image B , B_{ms}^j is the blurred image in the coarse scale space obtained by the down-sampling

of F_{ms}^j , and F_{ms}^{j+1} are the features extracted from B_{ms}^j . Here, a convolutional layer with step size 2 reduces the feature map size to half of the original size and doubles the number of channels. In contrast, the deconvolution layer with step size 2 halves the number of feature channels and doubles the size of the feature map.

The multi-scale guidance branch has the following advantages: first, multi-scale images are incorporated into the network structure, which reduces the number of training parameter updates and saves training time, and secondly, it increases the network width of the decoder branch.

3.3 Detail optimization unit

Due to the large gap between the blurred image and the corresponding sharp image, if the normal network learning structure is used directly for deblurring, the blurred edges of the image will have serious ringing artifacts. So following Refs. [23,24,27], the problem needs to be broken down into several sub-problems and completed step by step. Our key idea is to first generate an initial result with a sharp structure, and then concentrate on extracting subtle information by learning the initial result and the residual image of the sharp image.

4 Loss function

In deep learning, classification problems can use loss functions such as cross-entropy, softmax, or support vector machine (SVM). If it is a regression problem, the loss function generally adopts L1 or L2 loss. With the development of network architectures, people have made many attempts to find a loss function to replace the widely used L1 and L2 losses. However, the resulting trade-off against perceptual distortion has recently been demonstrated. Advanced loss functions

(such as the adversarial loss of generative adversarial networks [32]) improve the perceptual image quality at the cost of distortion. Moreover, L2 has good sensitivity to errors, but is tolerant of small errors. Most importantly, L2 loss does not consider human visual perception, which is different from the human visual system output. In image deblurring, we not only need to restore the blurred edges in the image to clear structures, but also need to retain the color and detail information of the original image. In order to take human visual perception into consideration, the SSIM loss function [33] fully considers brightness, contrast, and structural indicators, and can restore texture details of the image well. At the same time, it also benefits from the progressive network structure, so the use of negative SSIM to train the dynamic recursive fast multi-scale residual network gives good results.

We formulate it in three parts. The first compares image illumination:

$$l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \quad (9)$$

The second compares image contrast:

$$c(x, y) = \frac{2\gamma_x\gamma_y + C_2}{\gamma_x^2 + \gamma_y^2 + C_2} \quad (10)$$

Finally, the third compares image structure:

$$s(x, y) = \frac{\gamma_{xy} + C_3}{\gamma_x\gamma_y + C_3} \quad (11)$$

where

$$\gamma_{xy} = \frac{1}{N-1} \sum_{i=1}^N (x_i - \mu_x)(y_i - \mu_y) \quad (12)$$

In the above, C_1 , C_2 , C_3 are constant terms to avoid instability when the denominator is close to zero. $x_i(y_i)$ and N are the image signal and the number of signals respectively. μ and γ act on the average intensity and standard deviation of the discrete signals x and y .

The SSIM loss function is obtained by multiplying the above three terms:

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\gamma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\gamma_x^2 + \gamma_y^2 + C_2)} \quad (13)$$

5 Experiments

5.1 Dataset

Synthetic blurred images are inadequate to represent the complexity of real blurred images. The camera movement has 6 degrees of freedom (6D), 3

translational and 3 rotational. Using the sharp image with a convolutional blurring kernel only considers two translational degrees of freedom in the 2D plane [34,36]. In addition, there are factors such as lens distortion, sensor saturation, camera nonlinear transformations, noise, compression, and depth of field that are not simulated by simple synthetic images.

GOPRO is a large-scale deblurring dataset proposed by Nah et al., taken using a GOPRO Hero Black camera. Unlike previous synthetic datasets of blurred images, it uses high-speed cameras to continuously capture short exposure sharp frames, and integrates and averages them to simulate long exposure blurred frames. Images formed in this way are closer to reality, and can simulate complex camera shake and non-uniform blurring caused by multiple target movements in the scene. The GOPRO dataset contains a total of 3214 pairs of sharp and blurred images with an image size of 720×1280 , with 2103 pairs of images for training and 1111 pairs for testing.

The Kohler dataset [36] is a benchmark dataset for evaluating and comparing blind deblurring algorithms. The author records and analyzes the real camera movement over time, then replays it with a robot carrier, and forms a dataset by leaving a series of sharp images on the movement track of the 6D camera. The Kohler dataset has 4 pictures, each picture is blurred with 12 different blur checkpoints, and finally 48 blurred images are formed.

5.2 Details

Our experiments were conducted on a PC equipped with eight TITAN RTX GPUs. We implemented our framework using the Pytorch platform. In addition, pixel filling is used to keep the output and input scales of the feature map unchanged. The Adam algorithm [37] is used to train the initial learning rate at 0.0001 with 10^{-3} weight decay every 30 epochs, with batch size set to 1. In experiments involving iterative model structure, the network shares the same training environment.

Our evaluation comprehensively verified different network structures and various network parameters. To be fair, all experiments were performed on the same dataset with the same training configuration unless otherwise stated. In order to evaluate the performance of our proposed method, we use peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) as our objective evaluation indicators (our

method also performs well on color-based evaluation indicators [38]).

5.3 Ablation study

5.3.1 Progressive deblurring

In order to verify our progressive approach to removing fuzziness, we analyzed the influence of the DOU on the performance of the deblurring network. The GoPro dataset test set was used in the IIU to remove fuzziness. As shown in Fig. 3, subjectively, the output of the IIU is less blurred than the original blurred image, the blurred areas gradually tend to become sharp, and some edges are restored. But there are still undesirable white spots and small areas of blurring. Observing the detailed areas of the image from left to right, the edges of the object gradually become sharper and closer to the ground truth. Thus, it can be concluded that the image deblurring method gradually shifts from the input to light blurring, and finally achieves a sharp image. In Table 2, giving results on the benchmark dataset, the output with fine details is somewhat better than the original blurred image, but the further enhanced structure

Table 2 Quantitative results in asymptotic units on the benchmark dataset

Method	Blurred	IIU	DOU + IIU
PSNR	21.63	28.53	33.32
SSIM	0.8323	0.9248	0.9612

and texture details also play a very important role in achieving more realistic results.

5.3.2 Edge perception

In order to verify the positive effect of edge information on the network, we removed the significant edge pyramid branch from the structure to observe the deblurring effect on the test set, when predicting blur images. Figure 4 subjectively shows that with the assistance of edge information, image edge recovery is very obvious. At the occluded edges of objects, there is no ringing effect, and recovery of distant objects is also obvious.

5.4 Comparison with other methods

5.4.1 Results using the benchmark dataset

We compared our method to existing image deblurring methods quantitatively on the GOPRO dataset. Then

Fig. 3 Left to right: blurred image, IIU output, complete network output, and sharp image.

Fig. 4 Left to right: blurred image, output with the edge branch removed, and output of the complete network.

we used the Kohler dataset on our training model for testing. This dataset is widely used by traditional methods and learning-based methods for further performance evaluation. Finally, we used the blurred images in the real scenes in the Lai dataset to test the generalization ability of our network. Since our method deals with motion blur, a fair comparison cannot be made to traditional uniform deblurring methods [39, 40]. So we chose the method of Whyte et al. [41] as a representative traditional method for non-uniform blur processing. At the same time, we also compared to similar motion blur removal methods. Both Nah et al. and Tao et al. used multi-scale architectures, but used parameter independence and parameter sharing to construct their deep networks. On this basis, Zhang et al. [27] used different depth stacking and layering methods to adapt to different images. In each case we used the authors' suggested default parameters.

Figure 5 shows our deblurring results on the benchmark dataset (unless otherwise stated, all images evaluated in our experiments are in RGB mode). For a fair experiment, we tested Nah et al.'s method on the Pytorch platform, not the original Lua code: the range of RGB is changed to $[0, 255]$, the loss function only uses L1 loss, not adversarial loss, we used mixed-precision training, and the SSIM function was converted from MATLAB to Python. This unifies the platform used by each method and contributes to objectivity of the experimental results. The author provided 5 different models on the two datasets.

Because all training and testing use the GOPRO dataset, we chose the best GOPRO_L1_amp as the test model for comparison. Three models are provided in the code provided by Tao et al. According to the author's description, the LSTM model works best, so it was used in our comparison. Models with different hierarchical structures were suggested; we chose their best DMPHN_1.2_4.8 model. Kupyn improved the deblurring effect and quality on the basis of the original GAN network implementation, and proposed a new version, DeblurGanv2. As a representative GAN network, we chose to use Inception-ResNet-v2 to implement the best model for testing.

The PSNR and SSIM metrics of the deblurred images from the GOPRO dataset were evaluated in Python, while the PSNR and MSSIM metrics for the Kohler dataset were calculated by the executable file provided by Ref. [36].

It can be seen from Fig. 5 and Table 3 that the structural similarity of PMERN in the GOPRO test is better than for other methods. The effects are particularly obvious in restoring image edge texture details. In the deblurring result of Nah et al.'s method, there are undesirable black patches. In the deblurring result of SNR (signal noise ratio), there are obvious artifacts and faults in the blurred areas. DMPTH achieved good results, but there is still room for improvement in detail and texture. GAN-v2 is too smooth, lacks texture details, and blotchiness appears in some areas. In contrast, PMERN both restores handwriting well, and the edge structure of the image. Edge texture and structural details of objects are retained to a large extent.

5.4.2 Results of real blurred images

Although the GOPRO dataset simulates real blur by averaging continuous frame synthesis, it is synthesized by a high-speed camera, and ground truth sharp images have severe noise and varying degrees of blur. Although the Kohler dataset is a real database, it only contains four different scenes. Therefore, we further tested our model on the real scene dataset of Lai et al.

Deblurring effects on real scenes can better reflect the adaptability of a network to real applications. As shown in Fig. 6, our method can generate sharp images for different scenes, and the recovered texture details are clearer than for other methods. It also works well for text processing. This shows the wide suitability of our network for different applications.

Fig. 5 Visual comparisons on testing dataset. Top to bottom: input (a), results of Whyte et al. (b), Nah et al. (c), Tao et al. (d), Zhang et al. (e), Kupyn et al. (f), our method (g), and sharp images (h).

Table 3 PSNR and SSIM results on test dataset

Method	Whyte et al.	Nah et al.	Tao et al.	Zhang et al.	Kupyn et al.	Ours
PSNR	24.60	29.23	30.10	31.20	30.51	33.32
SSIM	0.8458	0.9162	0.9323	0.9453	0.9121	0.9612

Figure 6 makes a qualitative comparison using the Kohler dataset. Table 4 shows the quality evaluation. Obviously, the restored images are of high visual quality, and our network can adapt to different scenarios.

6 Conclusions

In this article, we overcome the limitations of current image deblurring methods and describe a network structure using a multi-scale variant of blurred edge

Fig. 6 Results on real data.

Table 4 Quality evaluation on Kohler dataset

Method	Whyte et al.	Nah et al.	Kupyn et al.	Zhang et al.	Ours
PSNR	24.68	26.48	27.25	25.56	29.21
SSIM	0.7937	0.8079	0.7900	0.7867	0.8384

perception. This structure effectively integrates edge features and scale information cues. We also propose a progressive network for single image deblurring in dynamic scenes; it is outstanding at restoring image edge details. Our work provides a new idea for subsequent effective multi-scale deblurring of deep networks. Experimental results show that, compared to traditional methods and learning-based methods, this method provides better results on both benchmark datasets and real blurred images.

Acknowledgements

This research was supported by the National Natural Science Foundation of China (61772319, 62002200, 61976125, 61976124), and Shandong Natural Science Foundation of China (ZR2017MF049).

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

References

- [1] Gupta, A.; Joshi, N.; Lawrence Zitnick, C.; Cohen, M.; Curless, B. Single image deblurring using motion density functions. In: *Computer Vision – ECCV 2010. Lecture Notes in Computer Science, Vol. 6311*. Daniilidis, K.; Maragos, P.; Paragios, N. Eds. Springer Berlin Heidelberg, 171–184, 2010.
- [2] Harmeling, S.; Hirsch, M.; Schölkopf, B. Space-variant single-image blind deconvolution for removing camera shake. In: *Proceedings of the Advances in Neural Information Processing Systems 23*, 829–837, 2010.
- [3] Hirsch, M.; Schuler, C. J.; Harmeling, S.; Schölkopf, B. Fast removal of non-uniform camera shake. In: *Proceedings of the International Conference on Computer Vision*, 463–470, 2011.
- [4] Pan, J. S.; Sun, D. Q.; Pfister, H.; Yang, M. H. Blind image deblurring using dark channel prior. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1628–1636, 2016.
- [5] Xu, L.; Zheng, S. C.; Jia, J. Y. Unnatural L0 sparse representation for natural image deblurring. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1107–1114, 2013.
- [6] Chan, T. F.; Wong, C. K. Total variation blind deconvolution. *IEEE Transactions on Image Processing* Vol. 7, No. 3, 370–375, 1998.
- [7] Cho, S.; Lee, S. Fast motion deblurring. *ACM Transactions on Graphics* Vol. 28, No. 5, 1–8, 2009.
- [8] Pan, J. S.; Hu, Z.; Su, Z. X.; Yang, M. H. Deblurring text images via L0-regularized intensity and gradient prior. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2901–2908, 2014.
- [9] Couzinié-Devy, F.; Sun, J.; Alahari, K.; Ponce, J. Learning to estimate and remove non-uniform image blur. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1075–1082, 2013.
- [10] Kim, T. H.; Ahn, B.; Lee, K. M. Dynamic scene deblurring. In: *Proceedings of the IEEE International Conference on Computer Vision*, 3160–3167, 2013.
- [11] Li, J. J.; Li, G. H.; Fan, H. Image dehazing using residual-based deep CNN. *IEEE Access* Vol. 6, 26831–26842, 2018.
- [12] Li, J. J.; Feng, X. M.; Hua, Z. Low-light image enhancement via progressive-recursive network. *IEEE Transactions on Circuits and Systems for Video Technology* Vol. 31, No. 11, 4227–4240, 2021.
- [13] Zhang, T. L.; Li, J. J.; Hua, Z. Iterative multi-scale residual network for deblurring. *IET Image Processing* Vol. 15, No. 8, 1583–1595, 2021.
- [14] Schuler, C. J.; Burger, H. C.; Harmeling, S.; Schölkopf, B. A machine learning approach for non-blind image deconvolution. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1067–1074, 2013.
- [15] Xu, L.; Ren, J. S. J.; Liu, C.; Jia, J. Y. Deep convolutional neural network for image deconvolution. In: *Proceedings of the 27th International Conference on Neural Information Processing Systems*, Vol. 1, 1790–1798, 2014.
- [16] Schuler, C. J.; Hirsch, M.; Harmeling, S.; Schölkopf, B. Learning to deblur. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 38, No. 7, 1439–1451, 2016.
- [17] Tai, Y. W.; Chen, X. G.; Kim, S.; Kim, S. J.; Li, F.; Yang, J.; Yu, J.; Matsushita, Y.; Brown, M. S. Nonlinear camera response functions and image deblurring: Theoretical analysis and practice. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 35, No. 10, 2498–2512, 2013.
- [18] Deng, J.; Dong, W.; Socher, R.; Li, L. J.; Kai, L.; Li, F. ImageNet: A large-scale hierarchical image database. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 248–255, 2009.
- [19] Chakrabarti, A. A neural approach to blind motion deblurring. In: *Computer Vision – ECCV 2016. Lecture Notes in Computer Science, Vol. 9907*. Leibe, B.; Matas, J.; Sebe, N.; Welling, M. Eds. Springer Cham, 221–235, 2016.
- [20] Kupyn, O.; Budzan, V.; Mykhailych, M.; Mishkin, D.; Matas, J. DeblurGAN: Blind motion deblurring using conditional adversarial networks. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8183–8192, 2018.
- [21] Kupyn, O.; Martyniuk, T.; Wu, J. R.; Wang, Z. Y. DeblurGAN-v2: Deblurring (orders-of-magnitude) faster and better. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 8877–8886, 2019.
- [22] Gao, H. Y.; Tao, X.; Shen, X. Y.; Jia, J. Y. Dynamic scene deblurring with parameter selective sharing and nested skip connections. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3843–3851, 2019.
- [23] Nah, S.; Kim, T. H.; Lee, K. M. Deep multi-scale convolutional neural network for dynamic scene deblurring. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 257–265, 2017.
- [24] Tao, X.; Gao, H. Y.; Shen, X. Y.; Wang, J.; Jia, J. Y. Scale-recurrent network for deep image deblurring. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8174–8182, 2018.

- [25] Park, D.; Kang, D. U.; Kim, J.; Chun, S. Y. Multi-temporal recurrent neural networks for progressive non-uniform single image deblurring with incremental temporal training. In: *Computer Vision – ECCV 2020. Lecture Notes in Computer Science, Vol. 12351*. Vedaldi, A.; Bischof, H.; Brox, T.; Frahm, J. M. Eds. Springer Cham, 327–343, 2020.
- [26] Fergus, R.; Singh, B.; Hertzmann, A.; Roweis, S. T.; Freeman, W. T. Removing camera shake from a single photograph. In: *Proceedings of the ACM SIGGRAPH 2006 Papers*, 787–794, 2006.
- [27] Zhang, H. G.; Dai, Y. C.; Li, H. D.; Koniusz, P. Deep stacked hierarchical multi-patch network for image deblurring. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5971–5979, 2019.
- [28] Mao, X. J.; Shen, C. H.; Yang, Y. B. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. *arXiv preprint arXiv:1603.09056*, 2016.
- [29] Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Lecture Notes in Computer Science, Vol. 9351*. Navab, N.; Hornegger, J.; Wells, W.; Frangi, A. Eds. Springer Cham, 234–241, 2015.
- [30] Ye, M. Y.; Lyu, D.; Chen, G. S. Scale-iterative upscaling network for image deblurring. *IEEE Access* Vol. 8, 18316–18325, 2020.
- [31] Lim, B.; Son, S.; Kim, H.; Nah, S.; Lee, K. M. Enhanced deep residual networks for single image super-resolution. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 1132–1140, 2017.
- [32] Mirza, M.; Osindero, S. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.
- [33] Wang, Z.; Bovik, A. C.; Sheikh, H. R.; Simoncelli, E. P. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing* Vol. 13, No. 4, 600–612, 2004.
- [34] Levin, A.; Weiss, Y.; Durand, F.; Freeman, W. T. Understanding and evaluating blind deconvolution algorithms. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1964–1971, 2009.
- [35] Sun, L. B.; Cho, S.; Wang, J.; Hays, J. Edge-based blur kernel estimation using patch priors. In: *Proceedings of the IEEE International Conference on Computational Photography*, 1–8, 2013.
- [36] Köhler, R.; Hirsch, M.; Mohler, B.; Schölkopf, B.; Harmeling, S. Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database. In: *Computer Vision – ECCV 2012. Lecture Notes in Computer Science, Vol. 7578*. Fitzgibbon, A.; Lazebnik, S.; Perona, P.; Sato, Y.; Schmid, C. Eds. Springer Berlin Heidelberg, 27–40, 2012.
- [37] Kingma, D. P.; Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [38] Valberg, A. *Light Vision Color*. John Wiley and Sons, 2005.
- [39] Bai, Y. C.; Cheung, G.; Liu, X. M.; Gao, W. Graph-based blind image deblurring from a single photograph. *IEEE Transactions on Image Processing* Vol. 28, No. 3, 1404–1418, 2019.
- [40] Liu, J.; Yan, M.; Zeng, T. Surface-aware blind image deblurring. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 43, No. 3, 1041–1055, 2021.
- [41] Whyte, O.; Sivic, J.; Zisserman, A.; Ponce, J. Non-uniform deblurring for Shaken images. *International Journal of Computer Vision* Vol. 98, No. 2, 168–186, 2012.

Tianlin Zhang received his B.S. degree in software engineering from Shandong Jianzhu University, Shandong, China, in 2018. Currently, he is an M.S. candidate in the School of Computer Science and Technology, Shandong Technology and Business University, Yantai, China. His research interests include computer graphics, computer vision, and image processing.

Jinjiang Li received his B.S. and M.S. degrees in computer science from Taiyuan University of Technology, China, in 2001 and 2004, respectively, and his Ph.D. degree in computer science from Shandong University, Jinan, China, in 2010. From 2004 to 2006, he was an assistant research fellow at the Institute of Computer Science and Technology of Peking University. From 2012 to 2014, he was a post-doctoral fellow at Tsinghua University, Beijing, China. He is currently a professor at the School of Computer Science and Technology, Shandong Technology and Business University. His research interests include image processing, computer graphics, computer vision, and machine learning.

Hui Fan received his B.S. degree in computer science from Shandong University in 1984. He received his Ph.D. degree in computer science from Taiyuan University of Technology in 2007. From 1984 to 2001, he was a professor in the Computer Department of Taiyuan University Technology. He is currently a professor at Shandong Technology and Business University. His research interests include computer aided geometric design, computer graphics, information visualization, virtual reality, and image processing.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduc-

tion in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made.

The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Other papers from this open access journal are available free of charge from <http://www.springer.com/journal/41095>. To submit a manuscript, please go to <https://www.editorialmanager.com/cvmj>.