IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 10, 2019, accepted December 21, 2019, date of publication December 26, 2019,
date of current version January 14, 2020.

Digital Object Identifier 10.1109/ACCESS.2019.2962505

Video Deblurring via Temporally and Spatially Variant Recurrent Neural Network

RUNHUA JIANG, LI ZHAO, TAO WANG, JINXIN WANG, AND XIAOQIN ZHANG
College of Computer Science and Artificial Intelligence, Wenzhou University, Wenzhou 325035, China

Corresponding author: Li Zhao (lizhao@wzu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61922064, in part by the Zhejiang Provincial Natural Science Foundation under Grant LR17F030001 and Grant LQ19F020005, and in part by the Project of Science and Technology Plans of Wenzhou City under Grant C20170008, Grant G20150017, and Grant ZG2017016.

ABSTRACT The camera shake and high-speed motion of objects often produce a blurry video. However, it is hard to recover sharp videos using existing single or multiple image deblurring methods, as the blur artifacts in blurry videos are both temporally and spatially varying. In this paper, we propose a temporally and spatially variant recurrent neural network for video deblurring, in which both temporally and spatially variants employ ConvGRU blocks and a weight generator to capture spatio-temporal features. Meanwhile, the proposed model is trained in an end-to-end manner, where the model input and output are set to the same number. Thus, our model does not reduce the number of frames both in training and testing stages, which is important in practical applications. Quantitative and qualitative evaluations on standard benchmark datasets demonstrate that the proposed method outperforms the current state-of-the-art methods.

INDEX TERMS Spatio-temporal features, video deblurring, variant recurrent neural network.

I. INTRODUCTION

Motion blur is a common phenomenon in videos. In low-light conditions, camera shake and object movement often produce blurs at the time of exposure. In addition, even when the light is satisfied, the fast movement of the objects also causes blur artifacts in a video. This problem triggers lots of works on video deblurring, which aims at recovering sharp frames from the input blurred frames. The video deblurring methods are widely used in many areas of computer vision, such as denoising [1], tracking [2] and classification [3].

Early works mainly focus on single image deblurring, which recovers a sharp image given a single blurred image. Compared with the single image deblurring, video deblurring is more challenging, because it involves modeling joint spatial and temporal information in several frames. Some existing video deblurring methods [4]–[6] take a large batch of frames as input to model their long term dependence, and estimate the short term temporal relationship by gradually scanning these frames. However, we note that the short term consistency of temporal information within the frames is not fully captured, since it is labile in continuous frames. In addition, as their methods directly capture the long term information from all inputs, the captured temporal information often lacks the variable short term information, which may cause the discontinuity between generated frames. Moreover, these methods use multiple blurred frames to generate one sharp frame. This reduces the number of frames in the newly synthesized video. The missing frames may contain important information, which is essential for applications.

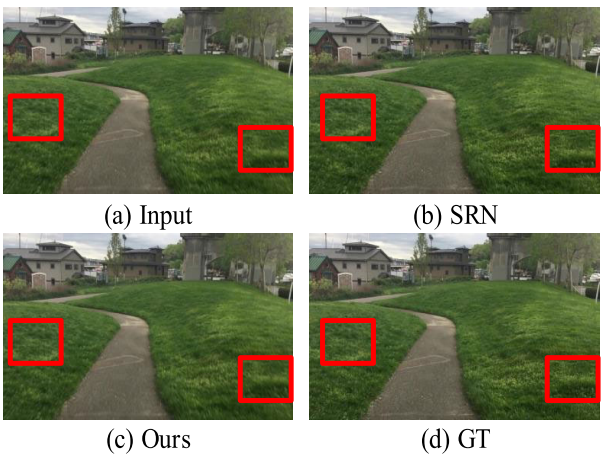

FIGURE 1. An example of deblurring results. Our model generates a sharper frame which looks more realistic compared with SRN [7].

In order to handle the aforementioned problems, a spatially and temporally variant recurrent neural network is proposed in this paper. The proposed network contains three parts:

The associate editor coordinating the review of this manuscript and approving it for publication was Mauro Gaggero.

the temporally variant block, the spatially variant block and the frame reconstruction block. The first two blocks are embedded with ConvGRU [8] blocks and a weight generator separately, which guides each module to work effectively. Together with the weight generators and ConvGRU blocks, the two blocks estimate information about the variant blur kernels and restore feature maps about sharp frames. Finally, the frame reconstruction block uses the outputs of the first two blocks to reconstruct the deblurred frames. In this way, our model can effectively learn the temporal information with different lengths and spatial information, which are important for video deblurring. Fig. 1 shows the visual results of the proposed video deblurring method. Compared with SRN [7], our model can produce more realistic deblurring effects.

The main contributions of this work are summarized as follows:

- We propose a model with the temporal and spatial blocks, which applies ConvGRU blocks in a deep recurrent neural network to capture joint long term and short term spatio-temporal features for video deblurring.
- Our proposed model is trained together with the two weight generators, in which the temporal and spatial information of the model input and output is hardly lost. Thus, our model does not reduce the number of frames both in training and testing stage, which is important in practical applications.

The rest of this paper is structured as follows. Section II briefly reviews related works on image deblurring, video deblurring and recurrent neural networks. The proposed method and experiments are presented in Section III and Section IV. Section V concludes this paper.

II. RELATED WORK

Our work is closely related to three topics: image deblurring, video deblurring and recurrent neural networks. These topics are discussed in the following subsections.

A. IMAGE DEBLURRING

Image deblurring aims at restoring a sharp image from a blurred one. Most successful approaches to image deblurring [9], [10] are based on the uniform blur model as follows:

$$B = k * S + N, \tag{1}$$

where B represents a blurred image, k refers to the unknown blur kernel, and S is the sharp image. The operation of $*$ is the convolution, and N is a noise term.

There are two kinds of image deblurring methods: non-blind image deblurring [10]–[14], and blind image deblurring [15]–[19]. The solutions of the non-blind deblurring depend on an assumption that the blur kernels are known in advance. In order to acquire the S, early non-blind deblurring methods [11]–[13] use the classical Lucy-Richardson algorithm, which is an iterative algorithm based on Bayesian analysis. In most cases, blind deblurring is an ill-posed problem where S is not uniquely determined by B and k. Therefore, many blind deblurring methods rely on heuristics, image statistics and hypothetical blur kernels. For example, [9], [10], [20], [21] are based on an iterative method. Both of them use parametric prior models to estimate the motion kernel and the sharp image at each iteration.

Recently, data fitting term is also used for image deblurring [6], [22]–[25]. Pan *et al.* [22] propose a data-driven approach to learn a data fitting function, which is used to estimate the blur kernels for blind image deblurring. Whyte *et al.* [23] describe a novel method for non-uniform blind deblurring depended on a parametrized geometric model of the blurring process. Sun *et al.* [24] use a convolution neural network (CNN) to estimate the blur kernels.

B. VIDEO DEBLURRING

Compared with image deblurring, video deblurring is more challenging as it needs consider the problem of modeling temporal information. Generally, there are two main methods for video deblurring: deconvolution based methods, multi-frame aggregation and fusion based methods. For example, [26]–[28] are typical representatives of deconvolution based methods. Specially, Li *et al.* [27] first put temporal information into consideration for video deblurring. It solves the deblurring problem by minimizing an energy function defined on a multi-image deconvolution. However, previous deconvolution-based methods may not work well when facing various motions from dynamic blur scenarios. To handle this problem, the authors in [28] propose a novel energy method which uses pixel-wise kernel estimation and [26] takes the effect of depth variations on blur into consideration.

Multi-frame aggregation and fusion methods use the fact that not all video frames have the same amount of blurs. Pixel values can be sharpened using the values in nearby frames. Cho *et al.* [29] propose a patch-based alignment algorithm to recover sharp frames. Klose *et al.* [30] project pixel values into a single reference frame for pixel fusion. Recently, multi-frame aggregation and fusion approaches based on deep learning have been widely used in video deblurring. In [31], a recurrent neural network is used to learn spatio-temporal information between multiple consecutive frames to fuse a central sharp frame. Ren *et al.* [32] solve the video deblurring problem with the help of the semantic segmentation of multiple frames. Tan *et al.* [33] put forward a kernel-free method to restore sharp frames by using the same contents among continuous frames. DBN [4] takes multiple continuous frames as input to produce the middle sharp frame. It uses 2D convolution to model spatio-temporal information. Different from the DBN, Tae *et al.* [34] propose a network layer that enforces temporal consistency between consecutive frames, and a recurrent network to reconstruct the deblurred frames.

C. RECURRENT NEURAL NETWORKS

Benefiting from the rapid requirement of sequential information processing tasks (*e.g.*, natural language processing, video super-resolution, video deblurring), recurrent neural networks have made great progress. Current methods usually

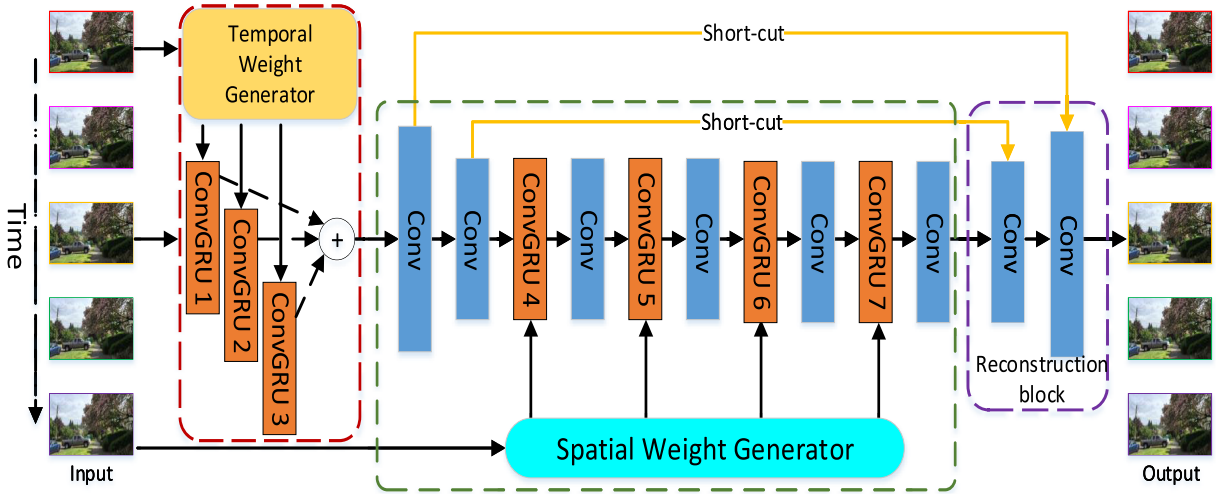

FIGURE 2. The architecture of the proposed network. The block enclosed in red is the temporally variant block. The block enclosed in green is the spatially variant block. The block enclosed in purple is the frame reconstruction block. There are five input frames and five output frames during training and testing.

follow the standard Recurrent Neural Network (RNN) [35]. However, ordinary RNN is difficult to train, because it is easy to cause the vanish or explode gradient problem during training stage [36], [37]. After finding the problem of ordinary RNN, the Long Short Term Memory (LSTM) [38] architecture is proposed to address it. Although LSTM solves the above problem by designing a long-short term dependence mechanism, it requires large training datasets to obtain a better generalization ability due to its huge number of parameters. In order to handle this problem, Cho *et al.* [39] propose a GRU architecture to improve the LSTM. Although the GRU inherits the strengths of both RNN and LSTM, it still lacks the ability of considering spatial coherence across images [8], [35]. To tackle this drawback, authors in [8] propose a long-term recurrent convolutional network that takes convolutions as the basis of ConvGRU. Moreover, Liu *et al.* [40] propose the spatially variant RNN, where spatially-variant weights of the RNN are learned by a deep CNN. By utilizing the deep CNN, the spatially variant RNN does not need to use a large number of parameters since spatial information of an image can be propagated by the RNN.

III. PROPOSED APPROACH

In this section, the overall architecture of the proposed network is presented. After that, each component of the proposed network is introduced in detail.

A. OVERALL ARCHITECTURE

Architecture of the proposed network is shown in Fig. 2. It consists of three components: the temporally variant block, the spatially variant block, and the frame reconstruction block. The temporally variant block models the temporal information in a set of five blurry frames. Then, the spatially variant block uses four ConvGRU blocks and a series of convolutional layers for deblurring. Finally, the frame reconstruction block restores sharp frames by using features produced by the spatially variant block. In order to accelerate the network convergence, two skip connections from the spatially variant block to the reconstruction block are used. In addition, as the input and output of the proposed network are five frames, the loss function is the Mean Square Error of five frames. Formally, it can be represented as following:

$$\mathcal{L}_{MSE} = \frac{1}{KWH}\sum_{k=1}^{K}\sum_{x=1}^{W}\sum_{y=1}^{H}(I^{sharp}_{k,x,y} - G(I^{blurry})_{k,x,y})^2. \quad (2)$$

In Eq. (2), K, W and H are the number of frames, the width and height of a frame. The I^{sharp} and $G(I^{blurry})$ represent the sharp frame and the corresponding deblurred frame.

TABLE 1. The detailed architecture of the temporal weight generator and the spatial weight generator.

Layer	Temporal Weight Generator		Spatial Weight Generator	
	Operation	Output channel	Operation	Output channel
0	Conv	16	-	-
1-2	Conv	96	Conv	64
3	Maxpool+Conv	128	Maxpool+Conv	128
4	Conv	128	Conv	128
5	Maxpool+Conv	256	Maxpool+Conv	256
6-7	Conv	256	Conv	256
8	Maxpool+Conv	512	Maxpool+Conv	512
9	Conv	512	Conv	512
10	Conv	256	Conv	256
11	Upsample+Conv	128	Upsample+Conv	128
12	Conv	128	Conv	128
13	Upsample+Conv	256	Upsample+Conv	256
14	Upsample+Tanh+Conv	96	Tanh+Conv	128

B. TEMPORALLY VARIANT BLOCK

The main difference between image deblurring and video deblurring is that video deblurring takes many consecutive frames as inputs, while image deblurring takes only one. Previous methods such as [29], [41], [42] directly

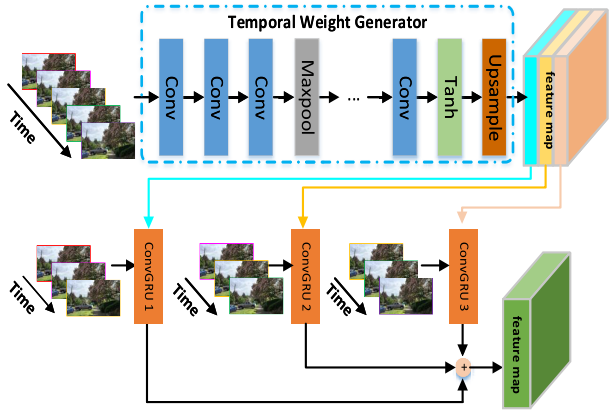

FIGURE 3. The structure of the temporally variant block. Detailed components of the temporal weight generator are presented in Table 1.

exploit patches across frames to restore sharp patches. However, these methods require the alignment of the blurred frames or the computation of optical flow [41]. All of them cause high computation consume. In this paper, the temporally variant block is proposed to capture temporal information more efficiently. As shown in Fig. 2, the proposed temporally variant block contains three ConvGRU blocks and a temporal weight generator. The temporally variant block is illustrated in more detail in Fig. 3.

Formally, one ConvGRU block is designed as follows:

$$z_t = \sigma(W_z * x_t + U_z * h_{t-1}), \tag{3}$$
$$r_t = \sigma(W_r * x_t + U_r * h_{t-1}), \tag{4}$$
$$\hat{h}_t = tanh(W * x_t + U * (r_t \odot h_{t-1})), \tag{5}$$
$$h_t = (1 - z_t)h_{t-1} + z_t\hat{h}_t, \tag{6}$$

where $*$ denotes convolution, $\odot$ is the dot product operation. W, W_z, W_r and U, U_z, U_r are convolution kernels, x_t and h_t are the input and output of ConvGRU block at time t. By taking Eq. (3−6), h_t is computed from h_{t-1} and $\hat{h}_t$, which are the output at time $t-1$ and the new output generated at time t. However, temporal relationship captured by transforming h_{t-1} cannot represent global relationship among all inputs. As discussed in the literature [43], deep CNN is able to extract high-level information from amounts of images and often show strong ability of generalization. Therefore, in this work, a deep CNN (*i.e.*, the temporal weight generator) is adopted to generate the global relationship and provide h_{t-1}. In addition, as discussed in the literature [8], [44], motion of video patches is usually restricted to a local neighborhood, and ConvGRU is able to extract temporal patterns from different time scales. Therefore, we empirically take three frames sequentially chosen from five inputs (see Fig. 3) as x_t.

As can be seen from Table 1, the temporal weight generator contains fifteen convolution layers, three maxpooling layers, three upsample layers, and one Tanh layer. Different from [4], it takes several convolution layers to estimate temporal relationship among all input frames. For better presentation, feature maps generated by the temporal weight generator are visualized in Fig. 4. To be specific, three consecutive frames

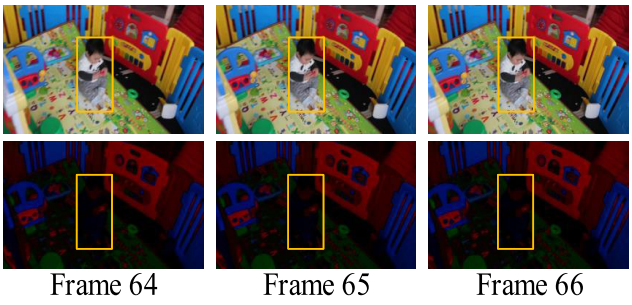

FIGURE 4. Visualizations of feature maps generated by temporal weight generator. From left to right, there are three consecutive frames. Among the three frames, only toys grasped by the child and background are moving.

and corresponding feature maps are presented. Among the three consecutive frames, only toys grasped by the child and background are moving. It can be find that among the three visualizations, the moving background and the toy are more salient than the non-moving child.

C. SPATIALLY VARIANT BLOCK

As proposed in [45], recurrent neural networks can be used to deblur single image with the assistance of pixel-wise weight generator. The motivation of [45] can be summarized as following:

$$y[n] = \sum_{m=0}^{M} k[m]x[n-m], \tag{7}$$

where y represents the blurred signal, M refers to the size of the kernel k and m is the position of 1D signal x. The input x can be restored by following:

$$\begin{aligned} x[n] &= \frac{y[n]}{k[0]} - \sum_{m=1}^{M} \frac{k[m]}{k[0]}\left(\frac{y[n-m]}{k[0]} - \frac{\sum_{l=1}^{M} k[l]x[n-m-l]}{k[0]}\right) \\ &= \frac{y[n]}{k[0]} - \sum_{m=1}^{M} \frac{k[m]y[n-m]}{k[0]^2} + \sum_{m=1,l=1}^{M,M} \frac{k[m]k[l]y[n-m-l]}{k[0]^2} \\ &= \cdots \end{aligned} \tag{8}$$

The existing study [45] uses four RNN layers and four convolution layers to approximate Eq. (8). However, this method may not effectively solve the problem of video deblurring because blurs in blurred videos have more dramatic changes [5]. In this work, the ConvGRU block is adopted to formulate the spatially variant block for following reasons. First, ConvGRU blocks are able to preserve spatial topology and temporal relationship among consecutive frames, while RNN can only preserve the temporal relationship [8]. Second, compared with RNN, ConvGRU blocks can better tackle these drastic blurs through various gates. Third, compared with other convolutional recurrent blocks (*e.g.*, convolutional RNN, convolutional LSTM), ConvGRU blocks have less parameters. In addition, the proposed spatially variant block

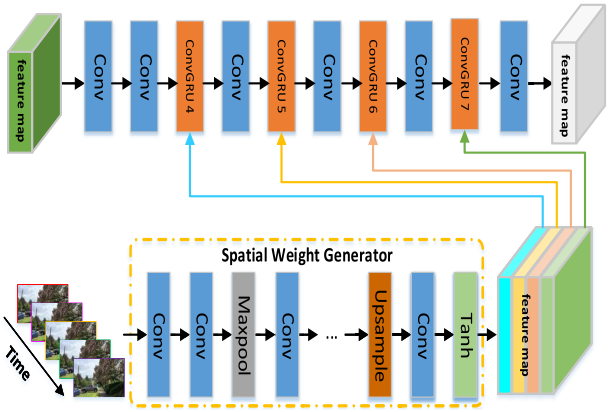

FIGURE 5. The structure of the spatially variant block. Green cube in upper-left is feature maps generated by the temporally variant block. Detailed components of the temporal weight generator are presented in Table 1.

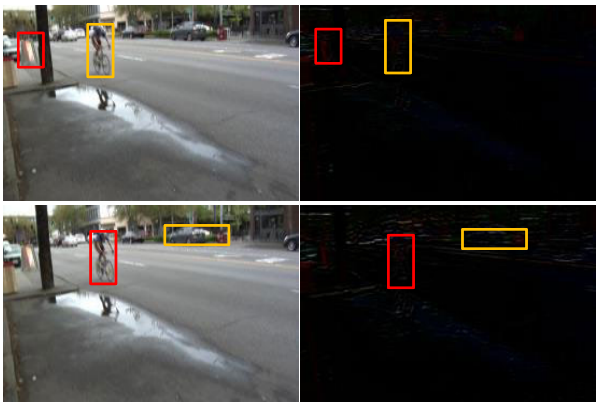

FIGURE 6. Visualizations of feature maps generated by the spatial weight generator. The left images are real-world blurry images, and the right images are feature maps generated by the spatial weight generator.

takes features generated by the temporally variant block and consecutive frames as inputs, while [45] takes a single image.

The proposed spatially variant block has a weight generator, four ConvGRU blocks and six convolution layers as Fig. 5 illustrates. The kernel size of the first convolution layer in Fig. 5 is set to 3, the stride and padding are set to 1. The corresponding parameters of the second convolution layer are 4, 2 and 1, and the corresponding parameters of other convolution layers are 1, 1 and 0. Two LeakyReLU layers with slope equals to -0.1 are used after the first two convolution layers. Detailed architecture of spatial weight generator is presented at Table 1.

During deblurring, the spatial wight generator first estimates spatial topology of all frames. Then, four ConvGRU blocks (*i.e.*, ConvGRU 4, ConvGRU 5, ConvGRU 6, ConvGRU 7) and convolution layers conduct the deblurring process. However, during going through the proposed network, spatio-temporal features extracted by the temporal weight generator are easy to loss due to the message passing inefficiency [46]. Inspired by [47], the spatial weight generator is adopted to extract spatio-temporal features and embed them into the latent feature space of the proposed model. Thus, the spatial topology and temporal relationship among consecutive frames are hardly lost. For better presentation, we also visualize feature maps generated by the spatial weight generator in Fig. 6. In the above figure, the left two images are taken from real-world blurry videos, and the right images are visualizations. It can be seen that the right images show blurry regions of left images. In addition, by observing the right two images, it is easy to find that they are able to indicate motions among consecutive frames.

D. FRAME RECONSTRUCTION BLOCK

The frame reconstruction block contains two convolution layers. The kernel size in the first of these convolution layers is set to 9, padding and stride are set to 4 and 1 respectively. The kernel in the second convolution layer is set to 3, while padding and stride are both set to 1.

The first convolution layer takes deblurred features produced by the spatially variant block as inputs. After the first convolution layer, bilinear interpolation is used to magnify the feature size by a factor of 2. At the second convolution layer, the feature channel is reduced from 32 to 5. As illustrated in Fig. 2, there are two skip connections from the first two convolution layers of spatially variant block to the frame reconstruction block.

By combining the temporally variant block and the spatially variant block into a unified reconstruction framework, the information in the blurry frames can be almost transformed from inputs to outputs in the proposed networks, therefore it can obtain a same number of deblurred image frames.

IV. EXPERIMENTS

In this section, experiments are presented to show the effectiveness of proposed network. Firstly, we introduce the public benchmark dataset used in this paper. Secondly, we discuss the training details of the proposed network. Then, the effectiveness of different parts of the model is analysed. Finally, we compare the proposed network with some state-of-the-art methods.

A. DATASET

Su *et al.* [4] propose a benchmark dataset (*i.e.*, VideoDeblurring dataset) for video deblurring. These videos which contain about 100 frames of 1280×720 size are captured via iPhone 6s, GoPro Hero 4 black, and Canon 7D at 240 FPS. After capturing these videos, blurry videos are generated by averaging consecutive seven frames. To be specific, the VideoDeblurring dataset consists of two subsets: a quantitative subset and a qualitative subset. The first subset contains 6,708 synthetic blurry frames with corresponding 71 ground truth videos. Videos in the second subset are obtained from 22 different scenes without ground truth data. Therefore, we split the quantitative subset to a training set and a testing set. The training set contains 61 videos (*e.g.*, IMG-0019, IMG-0036 and 720p-240fps-1), while the

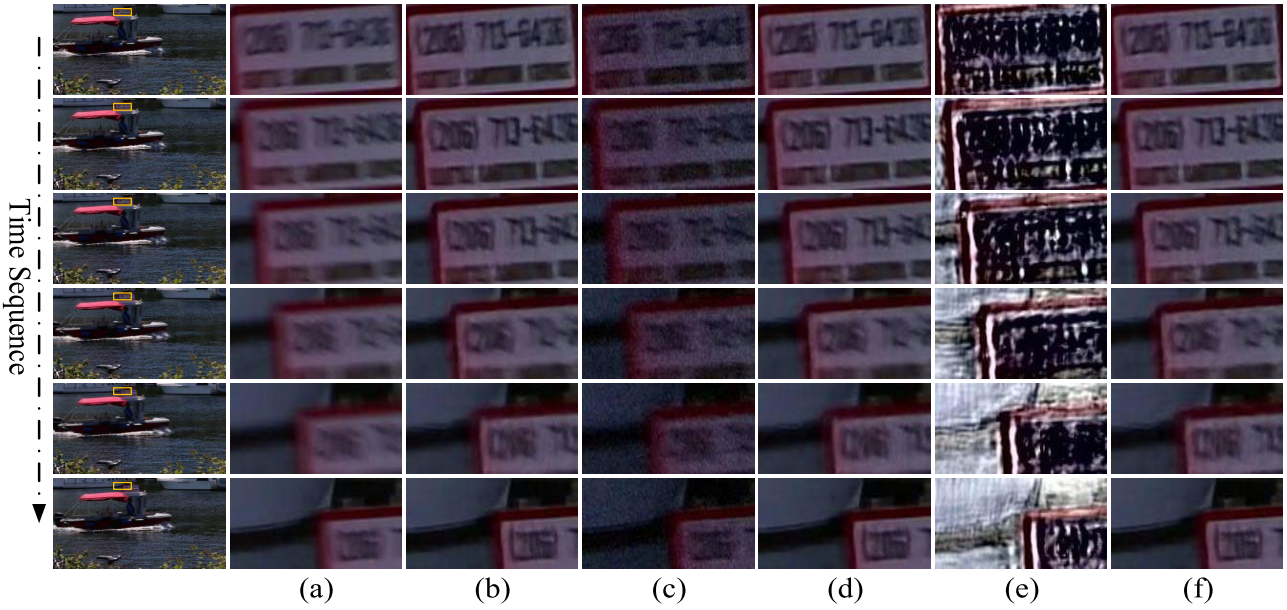

FIGURE 7. Visual comparisons by removing different components from the proposed model. (a) is the input blurry frame. (b) presents images that are generated without the spatial weight generator, while images in (c) are generated without the temporal weight generator. During generating images in (d) and (e), ConvGRU blocks in the spatially variant block and temporally variant block are removed, respectively. (f) presents deblurred images.

testing set contains other 10 videos (*e.g.*, IMG-0021, IMG-0030 and 720p-240fps-2). In total, there are 5,708 blurry-sharp pairs are used for training, and 300 pairs for testing. More details about the training set and testing set are available at https://github.com/shuochsu/DeepVideoDeblurring. We compare the proposed method with many state-of-the-art methods in the testing set. Moreover, we make a visual comparison using the qualitative subset.

B. IMPLEMENTATION DETAILS

Durring training stage, in order to augment training data, we crop 128×128 patches from any location of input frames. At least 712,193 samples are obtained in this way [4]. The batch size is set to 4 in training stage. All the frames are transformed into YCbCr space. The Y channel is used as inputs of the proposed model, and the corresponding Cb, Cr channels are used to restore the generated frame to RGB space. All the weights of the proposed network are initialized via a Gaussion distribution $N(0, 0.01)$.

In this paper, we use Eq. (2) as the loss function to train our network. Empirically, we set the learning rate as 1e-5. Adam with momentum to 0.9 is used to optimize our network. We implement our model with the PyTorch framework and a NVIDIA GTX 1080ti GPU.

C. MODEL ANALYSIS

To better validate the effectiveness of the proposed blocks, we define four sub-networks which contain different components to make quantitative and qualitative comparisons.

1) EFFECTIVENESS OF TEMPORALLY VARIANT BLOCK

The proposed temporally variant block is designed to capture temporal information among frames. Similarly, [5] proposes the 3D convolution to model temporal information. Therefore, a sub-network is proposed in which the temporally variant block is replaced by 3D convolution layers. This sub-network is referred to as 3D-sub. In addition, RNNs, which is proposed in [45], is taken as a special sub-network that does not have the temporally variant block. Therefore, by comparing the RNNs and the proposed network, effectiveness of the temporally variant block can be verified. Moreover, its effectiveness can be further verified by comparing 3D-sub, which takes 3D convolutions to capture temporal information.

TABLE 2. Performance comparison between sub-networks with different components.

Networks	PSNR	SSIM
RNNs [45]	30.05	0.92
3D-sub	30.12	0.90
LSTM-sub	29.60	0.89
Spatial-sub	30.31	0.90
Temporal-sub	30.27	0.90
Ours	31.13	0.91

As illustrated in Table 2, performance of the 3D-sub and RNNs are worse than the proposed network. PSNR of the proposed network is about 1.01 higher than the PSNR for 3D-sub, and 1.08 higher than RNNs. In addition, qualitative comparisons are made by removing ConvGRU blocks and temporal weight generator in this block. To be specific, as can be seen from Fig. 7, images generated without the temporal weight generator tend to be unrealistic. For example, images in the first row is the sharpest frame. Transforming information from this frame to others is beneficial

FIGURE 8. Visual comparison with the state-of-the-art deblurring methods in the quantitative subset.

to overall deblurring process. However, since the temporal weight generator is removed, the proposed model cannot effectively model long-term temporal relationship. Thus, with time sequence, deblurring effects in the (c) column become worse. In the (e) column, almost all image content are lost as ConvGRU blocks in the temporally variant block are removed. To be more specific, these ConvGRU blocks aim at extracting short-term spatio-temporal information from adjacent frames, which is essential for preserving image content. Both of the quantitative and qualitative experiments demonstrate effectiveness of the proposed temporallly variant block.

2) EFFECTIVENESS OF SPATIALLY VARIANT BLOCK

There are many convolutional recurrent neural networks such as ConvRNN, ConvLSTM. For verifying effectiveness of ConvGRU blocks used in the spatially variant block, we construct a sub-network which is named as LSTM-sub. In order to avoid the influence of temporally variant block, the LSTM-sub also takes a 3D convolution layer to model temporal information as 3D-sub does. Therefore, the only difference between 3D-sub and LSTM-sub is that the later one uses four ConvLSTM layers to form the spatially variant block.

As presented in Table 2, both PSNR and SSIM of 3D-sub outperform LSTM-sub. By comparing their components, we find the ConvGRU blocks which are used in 3D-sub improve the PSNR by about 1.7%. In addition, as can be seen from the (d) column of Fig. 7, local regions in same blurry frames cannot be effectively recovered (*e.g.*, the image in the 3-rd row) as the ConvGRU block is removed. By comparing images in (b) and (f), it can be find that heavily blurred images are almost unrestored in the (b) column (*e.g.*, images in the 4-th and 5-th row), while images in (f) column achieve the best visual performance. Thus, effectiveness of the proposed spatially variant block is demonstrated.

TABLE 3. Performance comparisons of the proposed method by varying the number of input frames.

Networks	PSNR	SSIM
I1T1	30.27	0.90
I3T1	30.88	0.91
I7T1	30.03	0.89
I5T1	31.01	0.91
I5T5	30.98	0.91
I5T3	31.13	0.91

3) EFFECTIVENESS OF WEIGHT GENERATORS

In the proposed network, two weight generators are adopted to capture spatio-temporal information for video deblurring. Therefore, effectiveness of the two generators is important to the overall performance. In order to verify the influence of the two weight generators, we also propose two sub networks which lack one of the two weight generators. The two sub networks are named as Spatial-sub and Temporal-sub. In the Spatial-sub, the spatial weight generator is deleted. In the Temporal-sub, we delete the temporal weight generator.

As presented in Table 2, the Spatial-sub achieves 30.31 dB in terms of PSNR, and Temporal-sub achieves 30.27 dB. Compared with the proposed network, PSNR of the two sub network is about 0.86 dB lower. Even these metrics indicate that removing the two generators will weaken

TABLE 4. Quantitative comparison with state-of-the-art methods on the VideoDeblurring dataset [4].

Method	1	2	3	4	5	6	7	8	9	10	Average (PSNR)
INPUT	24.14	30.52	28.38	27.31	22.60	29.31	27.74	23.86	30.59	26.98	27.14
PSDEBLUR	24.42	28.77	25.15	27.77	22.02	25.74	26.11	19.71	26.48	24.62	25.08
DeblurGAN [48]	25.23	29.17	27.82	27.51	22.58	28.83	26.83	23.84	31.04	26.18	26.90
MSCNN [6]	26.84	31.56	29.29	29.46	24.19	29.94	28.50	25.18	32.07	27.89	28.49
WFA [49]	25.89	32.33	28.97	28.36	23.99	31.09	28.58	24.78	31.30	28.20	28.35
DBN (single) [4]	25.75	31.15	29.30	28.38	23.63	30.70	29.23	25.62	31.92	28.06	28.37
DBN (noalign) [4]	27.83	33.11	31.29	29.73	25.12	32.52	30.80	27.28	33.32	29.51	30.05
DBN (flow) [4]	28.31	33.14	30.92	29.99	25.58	32.39	30.56	27.15	32.95	29.53	30.05
STAN (M/A_A) [50]	28.73	33.34	31.21	30.77	25.33	32.56	30.11	27.07	34.13	29.62	30.29
DMPHN [51]	29.89	33.35	31.82	31.32	26.35	32.49	30.51	27.11	34.77	30.02	30.76
RNNs [45]	-	-	-	-	-	-	-	-	-	-	30.05
IFI-RNN [52]	-	-	-	-	-	-	-	-	-	-	30.73
Ours	28.61	35.01	31.67	31.54	25.48	33.00	31.13	28.26	36.58	30.05	31.13

the overall performance, qualitative experiments are still needed for demonstrating their effectiveness. Therefore, during generating images in the (b) and (c) column of Fig. 7, the two generators are removed from the proposed model. It is easy to find that without the spatially weight generator, heavily blurred frames are not well restored. On the other hand, without the temporally weight generator, deblurring performance becomes worse with time sequence.

4) DIFFERENT FRAMES

We are curious about how the number of input frames influences the performance of the proposed network. Therefore, we vary the number of input frames in the proposed model. Comparison results are shown in Table 3. In the above table, numbers of input frames in the temporally variant block and the spatially variant block are denoted with prefix I1, I3, I5 and I7. Also, we put a suffix T with number of input frames in ConvGRU1, ConvGRU2 and ConvGRU3. For example, I5T3 denotes that both temporally variant block and spatially variant block take five consecutive frames as inputs. Meanwhile, the ConvGRU1, ConvGRU2 and ConvGRU3 take three frames sequentially chosen from the five frames as inputs.

By comparing the I1T1, I3T1, I7T1, and I5T1, it is easy to find with the increase of input frames, PSNR of the I1T1, I3T1, and I5T1 become higher. However, PSNR of the I7T1 is lower than I5T1. This demonstrates that the proposed model can utilize long-term spatio-temporal relationship among consecutive frames to achieve better performance. In addition, for demonstrating the ability of capturing short-term features, inputs of ConvGRU 1, ConvGRU 2 and ConvGRU 3 are changed to 1, 3 and 5 frames (*i.e.*, I5T1, I5T3 and I5T5). As the I5T3 achieves the highest PSNR, it is easy to find that the proposed network can also utilize the short-term features.

D. COMPARISON

In order to demonstrate the effectiveness of our proposed network, we compare it to some state-of-the-art methods such as PSDEBLUR, DeblurGAN [48], MSCNN [6], WFA [49], DBN [4], STAN(M/A_A) [50], DMPHN [51], RNNs [45] and IFI-RNN [52]. In Table 4, PSDEBLUR is the deblurred results of PHOTOSHOP, and INPUT represents the blurry images. For fair comparison, four image deblurring methods and four video deblurring methods are taken. Specifically, DeblurGAN [48] is an end-to-end model for image deblurring, which is based on the adversarial learning. DMPHN [51] utilizes feature maps at different scales to tackle the image deblurring problem. Zhang *et al.* [45] propose a spatially variant recurrent network to deblur a single image. WFA [49] uses multiple frames as inputs to produce a deblurred frame. DBN (single), DBN (noalign), DBN (flow) are three variants of the DBN [4], which stacks 5 copies of one single frame as input. STAN [50] uses a motion estimation and motion compensation module to warp the previous deblurred frame to restore the current frame. The method IFI-RNN [52] is also a recurrent neural network aims at video deblurring. However, the most difference between the IFI-RNN and our method is that hidden states of our model is provided by the two weight generators rather than transformation from former recurrent cells.

Table 4 shows the PSNR values of the generated frames on the test datasets. The proposed method achieves the best result of video deblurring in terms of the PSNR. Compared with DeblurGAN [48] and MSCNN, our method improves the average values of PSNR to 31.13 dB, which proves that our proposed model is better at deblurring blurry videos. For the newest image deblurring methods DMPHN [51] and RNNs [45], their ability of video deblurring are also worse than the proposed model. When compared to video deblurring methods such as MSCNN and WFA, our model outperforms them by about 9.8%. Variations of DBN [4] (DBN (single),

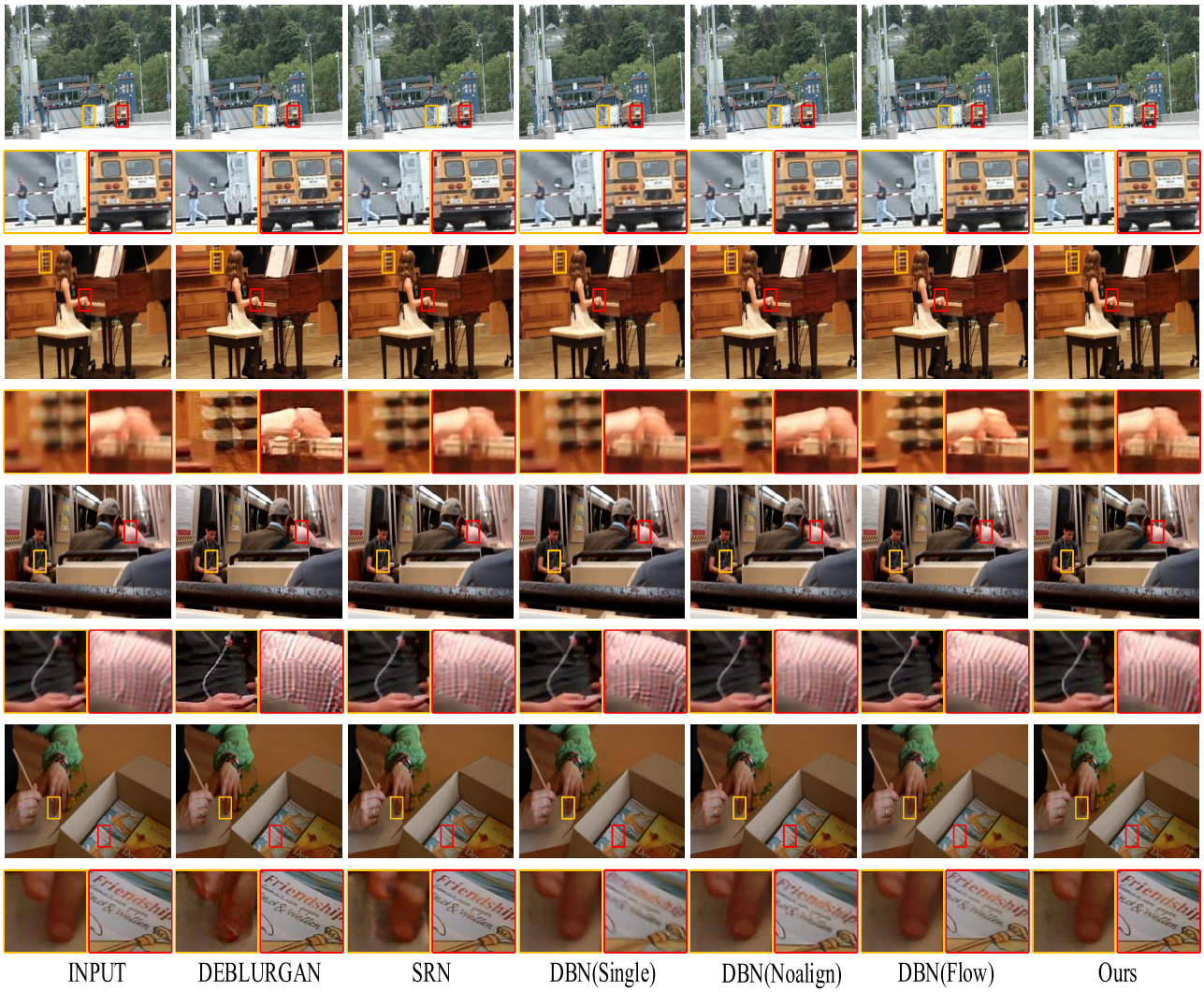

FIGURE 9. Visual comparison with the state-of-the-art deblurring methods in the qualitative subset.

DBN (noalign), DBN (flow)) are all worse than our proposed method. In addition, PSNR of the proposed method is also higher than the newest video deblurring method IFI-RNN. The above results show that our model has better performance for video deblurring. In addition, we make a visual comparison with many state-of-the-art methods using the quantitative subset and the qualitative subset. As shown in Fig. 8 and Fig. 9, the generated frames of our model achieve state-of-the-art visual appearance. This shows that our network can remove motion blur effectively in real scenes.

V. CONCLUSION

In this paper, we propose a novel recurrent neural network with spatially variant and temporally variant blocks, which model long-short term temporal information and spatial information for video deblurring. Our experiments demonstrate that each module in the proposed network can capture the corresponding features effectively. At the same time, our method solves the problem of frame loss in previous methods. Both quantitative and qualitative experiments on standard dataset demonstrate that the proposed method achieves state-of-the-art performance.

REFERENCES

[1] X. Zhang, J. Zheng, D. Wang, and L. Zhao, "Exemplar-based denoising: A unified low-rank recovery framework," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.

[2] X. Zhang, W. Hu, N. Xie, H. Bao, and S. Maybank, "A robust tracking system for low frame rate video," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 279–304, 2015.

[3] X. Zhang, D. Wang, Z. Zhou, and Y. Ma, "Robust low-rank tensor recovery with rectification and alignment," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published.

[4] S. Su, M. Delbracio, J. Wang, G. Sapiro, W. Heidrich, and O. Wang, "Deep video deblurring for hand-held cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 1279–1288.

[5] K. Zhang, W. Luo, Y. Zhong, L. Ma, W. Liu, and H. Li, "Adversarial spatio-temporal learning for video deblurring," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 291–301, Jan. 2019.

[6] S. Nah, T. H. Kim, and K. M. Lee, "Deep multi-scale convolutional neural network for dynamic scene deblurring," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 3883–3891.

[7] X. Tao, H. Gao, X. Shen, J. Wang, and J. Jia, "Scale-recurrent network for deep image deblurring," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8174–8182.
[8] N. Ballas, L. Yao, C. Pal, and A. Courville, "Delving deeper into convolutional networks for learning video representations," 2015, *arXiv:1511.06432*. [Online]. Available: https://arxiv.org/abs/1511.06432
[9] A. Gupta, N. Joshi, C. L. Zitnick, M. Cohen, and B. Curless, "Single image deblurring using motion density functions," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2010, pp. 171–184.
[10] M. Jin, S. Roth, and P. Favaro, "Noise-blind image deblurring," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 3510–3518.
[11] S. Cho, J. Wang, and S. Lee, "Handling outliers in non-blind image deconvolution," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 495–502.
[12] U. Schmidt, C. Rother, S. Nowozin, J. Jancsary, and S. Roth, "Discriminative non-blind deblurring," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 604–611.
[13] C. J. Schuler, B. H. Christopher, S. Harmeling, and B. Scholkopf, "A machine learning approach for non-blind image deconvolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1067–1074.
[14] J. Zhang, J. Pan, W.-S. Lai, R. W. H. Lau, and M.-H. Yang, "Learning fully convolutional networks for iterative non-blind deconvolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 3817–3825.
[15] J. Chen, L. Yuan, C.-K. Tang, and L. Quan, "Robust dual motion deblurring," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.
[16] M. G. Jin, S. Roth, and P. Favaro, "Normalized blind deconvolution," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 694–711.
[17] Q. Shan, J. Jia, and A. Agarwala, "High-quality motion deblurring from a single image," *TOGACM Trans. Graph.*, vol. 27, no. 3, p. 73, Aug. 2008.
[18] H. Zhang, J. Yang, Y. Zhang, N. M. Nasrabadi, and T. S. Huang, "Close the loop: Joint blind image restoration and recognition with sparse representation prior," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 770–777.
[19] X. Zhu, F. Šroubek, and P. Milanfar, "Deconvolving psfs for a better motion deblurring using multiple images," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2012, pp. 636–647.
[20] J. Dong, J. Pan, Z. Su, and M.-H. Yang, "Blind image deblurring with outlier handling," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2478–2486.
[21] H. Park and K. M. Lee, "Joint estimation of camera pose, depth, deblurring, and super-resolution from a blurred image sequence," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 4613–4621.
[22] J. Pan, J. Dong, Y.-W. Tai, Z. Su, and M.-H. Yang, "Learning discriminative data fitting functions for blind image deblurring," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 1068–1076.
[23] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce, "Non-uniform deblurring for shaken images," *Int. J. Comput. Vis.*, vol. 98, no. 2, pp. 168–186, Jun. 2012.
[24] J. Sun, W. Cao, Z. Xu, and J. Ponce, "Learning a convolutional neural network for non-uniform motion blur removal," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 769–777.
[25] M. Noroozi, P. Chandramouli, and P. Favaro, "Motion deblurring in the wild," in *Proc. German Conf. Pattern Recognit.* Cham, Switzerland: Springer, 2017, pp. 65–77.
[26] T. H. Kim and K. M. Lee, "Generalized video deblurring for dynamic scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 5426–5434.
[27] Y. Li, S. B. Kang, N. Joshi, S. M. Seitz, and D. P. Huttenlocher, "Generating sharp panoramas from motion-blurred videos," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2424–2431.
[28] C. Paramanand and A. N. Rajagopalan, "Non-uniform motion deblurring for bilayer scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1115–1122.
[29] S. Cho, J. Wang, and S. Lee, "Video deblurring for hand-held cameras using patch-based synthesis," *TOGACM Trans. Graph.*, vol. 31, no. 4, pp. 1–9, Jul. 2012.
[30] F. Klose, O. Wang, J.-C. Bazin, M. Magnor, and A. Sorkine-Hornung, "Sampling based scene-space video processing," *TOGACM Trans. Graph.*, vol. 34, no. 4, p. 67, Jul. 2015.
[31] P. Wieschollek, M. Hirsch, B. Scholkopf, and H. P. Lensch, "Learning blind motion deblurring," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 231–240.
[32] W. Ren, J. Pan, X. Cao, and M.-H. Yang, "Video deblurring via semantic segmentation and pixel-wise non-linear kernel," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 1077–1085.
[33] F. Tan, S. Liu, L. Zeng, and Z. Bing, "Kernel-free video deblurring via synthesis," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2016, pp. 2683–2687.
[34] T. H. Kim, K. M. Lee, B. Scholkopf, and M. Hirsch, "Online video deblurring via dynamic temporal blending network," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4058–4067.
[35] J. Donahue, L. A. Hendricks, M. Rohrbach, S. Venugopalan, S. Guadarrama, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 677–691, Apr. 2017.
[36] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 1310–1318.
[37] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.
[38] N. Srivastava, E. Mansimov, and R. Salakhutdinov, "Unsupervised learning of video representations using LSTMs," 2015, *arXiv:1502.04681*. [Online]. Available: https://arxiv.org/abs/1502.04681
[39] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," 2014, *arXiv:1406.1078*. [Online]. Available: https://arxiv.org/abs/1406.1078
[40] S. Liu, J. Pan, and M. H. Yang, "Learning recursive filters for low-level vision via a hybrid neural network," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 560–576.
[41] M. Delbracio and G. Sapiro, "Hand-held video deblurring via efficient Fourier aggregation," *IEEE Trans. Comput. Imag.*, vol. 1, no. 4, pp. 270–283, Dec. 2015.
[42] Y. Matsushita, E. Ofek, W. Ge, X. Tang, and H.-Y. Shum, "Full-frame video stabilization with motion inpainting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 7, pp. 1150–1163, Jul. 2006.
[43] Y. Liu, J. Pan, J. Ren, and Z. Su, "Learning deep priors for image dehazing," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2019, pp. 2492–2500.
[44] T. Brox and J. Malik, "Large displacement optical flow: Descriptor matching in variational motion estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 3, pp. 500–513, Mar. 2011.
[45] J. Zhang, J. Pan, J. Ren, Y. Song, L. Bao, R. W. Lau, and M.-H. Yang, "Dynamic scene deblurring using spatially variant recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2521–2529.
[46] G. Wang, K. Wang, and L. Lin, "Adaptively connected neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 1781–1790.
[47] K. Yu, X. Wang, C. Dong, X. Tang, and C. C. Loy, "Path-restore: Learning network path selection for image restoration," 2019, *arXiv:1904.10343*. [Online]. Available: https://arxiv.org/abs/1904.10343
[48] O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas, "DeblurGAN: Blind motion deblurring using conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8183–8192.
[49] M. Delbracio and G. Sapiro, "Burst deblurring: Removing camera shake through Fourier burst accumulation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 2385–2393.
[50] Z. Zhan, X. Yang, Y. Li, and C. Pang, "Video deblurring via motion compensation and adaptive information fusion," *Neurocomputing*, vol. 341, pp. 88–98, May 2019.
[51] H. Zhang, Y. Dai, H. Li, and P. Koniusz, "Deep stacked hierarchical multi-patch network for image deblurring," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 5978–5986.
[52] S. Nah, S. Son, and K. M. Lee, "Recurrent neural networks with intra-frame iterations for video deblurring," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 8102–8111.

RUNHUA JIANG received the B.Sc. degree from the Department of Information Science, Tianjin University of Finance and Economy, China. He is currently pursuing the graduate degree majoring in computer software and theory with the College of Computer Science and Artificial Intelligence, Wenzhou University, China. His research interests include image and video processing, pattern recognition, and machine learning.

LI ZHAO received the B.Sc. degree in automation and the M.Eng. degree in control theory and control engineering from Central South University, China, in 2005 and 2008, respectively. She is currently an Assistant Researcher with Wenzhou University. Her research interests are in pattern recognition, computer vision, and machine learning.

TAO WANG received the B.Sc. degree in information and computing science from Hainan Normal University, China, in 2018. He is currently pursuing the graduate degree with the College of Computer Science and Artificial Intelligence, Wenzhou University, China. His research interests include several topics in computer vision, such as image/video quality restoration, adversarial learning, image-to-image translation, and reinforcement learning.

JINXIN WANG received the bachelor's degree in information and computing science from Wenzhou University, China, where he is currently pursuing the graduate degree with the College of Computer Science and Artificial Intelligence. His research interests include visual tracking, image generation, and deep learning.

XIAOQIN ZHANG received the B.Sc. degree in electronic information science and technology from Central South University, China, in 2005, and the Ph.D. degree in pattern recognition and intelligent system from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China, in 2010. He is currently a Professor with Wenzhou University, China. His research interests are in pattern recognition, computer vision, and machine learning. He has published more than 80 articles in international and national journals, and international conferences, including the IEEE T-PAMI, IJCV, IEEE T-IP, IEEE T-IE, IEEE T-C, ICCV, CVPR, NIPS, IJCAI, AAAI, and among others.