International Journal of Computer Vision (2023) 131:1840–1856
https://doi.org/10.1007/s11263-023-01793-y

Memory Based Temporal Fusion Network for Video Deblurring

Chaohua Wang[1] · Weisheng Dong[1] · Xin Li[2] · Fangfang Wu[1] · Jinjian Wu[1] · Guangming Shi[1]

Received: 10 June 2022 / Accepted: 6 April 2023 / Published online: 20 April 2023

Abstract
Video deblurring is one of the most challenging vision tasks because of the complex spatial-temporal relationship and a number of uncertainty factors involved in video acquisition. As different moving objects in the video exhibit different motion trajectories, it is difficult to accurately capture their spatial-temporal relationships. In this paper, we proposed a memory-based temporal fusion network (TFN) to capture local spatial-temporal relationships across the input sequence for video deblurring. Our temporal fusion network consists of a memory network and a temporal fusion block. The memory network stores the extracted spatial-temporal relationships and guides the temporal fusion blocks to extract local spatial-temporal relationships more accurately. In addition, in order to enable our model to more effectively fuse the multiscale features of the previous frame, we propose a multiscale and multi-hop reconstruction memory network (RMN) based on the attention mechanism and memory network. We constructed a feature extractor that integrates residual dense blocks with three downsample layers to extract hierarchical spatial features. Finally, we feed these aggregated local features into a reconstruction module to restore sharp video frames. Experimental results on public datasets show that our temporal fusion network has achieved a significant performance improvement in terms of PSNR metrics (over 1dB) over existing state-of-the-art video deblurring methods.

Keywords Video deblurring · Temporal fusion network (TFN) · Memory network · Local spatial-temporal information · Reconstruction memory network (RMN)

1 Introduction

Video has become an important medium for people in modern society to communicate with each other. Social media platforms such as TikTok and Kwai have become the dominant mobile apps for video sharing. However, various uncertainty factors, from camera shakes to moving objects, can cause serious quality degradation in video acquired by smartphones. Depending on the source of blurring (e.g., uniform vs. nonuniform), the complex spatial-temporal relationship of the blurred video is often difficult to capture accurately. Additionally, how to develop computationally efficient video deblurring methods to support energy-constrained applications is nontrivial. Despite decades of research, blind motion deblurring has remained an open problem in video technology.

Recently, a variety of deep learning-based video deblurring methods have been proposed. These methods focus mainly on how to combine useful information from adjacent frames to predict the current frame. Models based on the convolutional neural network (CNN) (Su et al., 2017; Wang et al., 2019) make the current frame stacked with adjacent frames as input for prediction. For reconstructing the target frame more effectively, some CNN-based models (Zhang et al., 2014, 2013, 2014) fuse the blurred prior information extracted by an optical flow network or other preprocessing methods with the input frames. However, the optimality of

Communicated by Jian Sun.

✉ Weisheng Dong
wsdong@mail.xidian.edu.cn

Chaohua Wang
3267928656@qq.com

Xin Li
xin.li@ieee.org

Fangfang Wu
ffwu_xd@163.com

Jinjian Wu
jinjian.wu@mail.xidian.edu.cn

Guangming Shi
gmshi@xidian.edu.cn

[1] Xidian University, Xián, China

[2] West Virginia University, Morgantown, USA

these fusion methods is often questionable due to the complex spatial-temporal dependence of video frames.

The analogy between video data and time series has inspired the class of models based on a recurrent neural network (RNN) (Jiang et al., 2020; Kim et al., 2017; Zhou et al., 2019; Nah et al., 2019) for video deblurring. When processing the current frame, RNN-based approaches count only on information from previous frames (i.e., assuming a causal neighborhood in the temporal domain). This method of processing frame sequences with causality constraint can not capture the motion relationship that is often bidirectional. Most recently, there has been a surge of work exploiting self-attention mechanisms to address the issue of blur nonuniformity (Zhong et al., 2020; Tsai et al., 2021; Wu et al., 2020). These networks usually focus on extracting global features of video frames in a coarse to fine manner by CNNs. However, the coarse-grained attention mechanism cannot effectively capture *local motion-related features* of the video, which is considered a major weakness of existing video deblurring methods.

The reconstruction of information in a video frame is achieved by the multiscale features of the encoder and decoder. For example, when a video frame is reconstructed, two features at the same scale, one at the encoder and the other at the decoder, contain a feature transformation relationship from blur frame to sharp frame. The two features of this transformational relationship play different roles, so we cannot treat them equally. However, most existing methods (Zhu et al., 2021; Zamir et al., 2021; Park et al., 2020; Kim et al., 2022; Chu et al., 2022) fuse them in the same way through concatenation and addition, thus ignoring the reconstruction process between the two features. In order to enable the model to mine and utilize useful information of previous video frames when reconstructing the current video frame, we proposed a multihop and multiscale reconstruction memory network (RMN). Our RMN maps the two features of the previous video frame to a memory cell in a key-value pair. When the current frame is reconstructed, the memory cell is queried and read, thus achieving more effective mining and utilization of the reconstructed frame.

For video deblurring, the spatial-temporal relationship of video is difficult to accurately capture. Unlike other deblurring models, we mainly capture and fuse spatial-temporal information by mining local spatial features from input video sequences during the reconstruction. We proposed a memory-based temporal fusion network (TFN) to capture local spatial-temporal relationships. In addition, to enable our model to more effectively fuse the multiscale features of the previous frame, we propose a multiscale and multihop reconstruction memory network based on the attention mechanism and memory network. The new insights brought about by our model consist of the following four aspects.

Extraction of Local Spatial-Temporal Relationships. To enable the model to capture local spatial and temporal relationships for video reconstruction, we propose a temporal fusion block, which performs time-domain fusion and reconstruction according to the correlation in the local feature groups of the bottleneck layer.

Memory of the Local Spatial-Temporal Relationship. To facilitate the extraction of local spatial-temporal relationships more accurately, we add a memory network on the basis of our temporal fusion block, which includes an updating mechanism and a memory cell. The memory cell stores the spatial-temporal relationship that has been extracted. Our model will use the update mechanism to read and update the local spatial-temporal relationship stored in the memory cell several times, to capture the local spatial-temporal relationship more accurately.

Reconstruction Memory Network. To allow our model to fuse the multiscale features of the previous frame more effectively, we propose a multiscale and multihop reconstruction memory network based on the attention mechanism and memory network as Sukhbaatar et al. (2015).

Improved Performance. Because our model can focus on video frames throughout the time domain, we can achieve an improved trade-off between deblurring performance (measured by PSNR) and computational complexity (measured by runtime), as shown in Fig. 1.

In the application of video deblurring, we focused on the question of how to effectively capture the local spatial-temporal relationship from all input video frames. Experiments on standard datasets show that our model recovers the details of blurred video frames more effectively for video deblurring than other state-of-the-art (SOTA) deblurring methods. As shown in Fig. 1, our model has achieved a significant performance improvement in terms of PSNR metrics (over $1dB$) over existing state-of-the-art video deblurring methods, including the latest RNN-MBP (Zhu et al., 2021).

2 Related Works

2.1 Video Delburring

Video deblurring tasks have become increasingly more important in the field of computer vision. In addition to improving video quality, video deblurring plays an important role in other video processing tasks, such as visual tracking (Wu et al., 2011; Lee et al., 2011), behavior recognition, etc. Research on the video deblurring task has gradually evolved from the single image deblurring algorithm (Kim & Lee, 2014; Xu & Jia, 2010; Pan et al., 2016) based on model guidance to the more challenging video deblurring method based

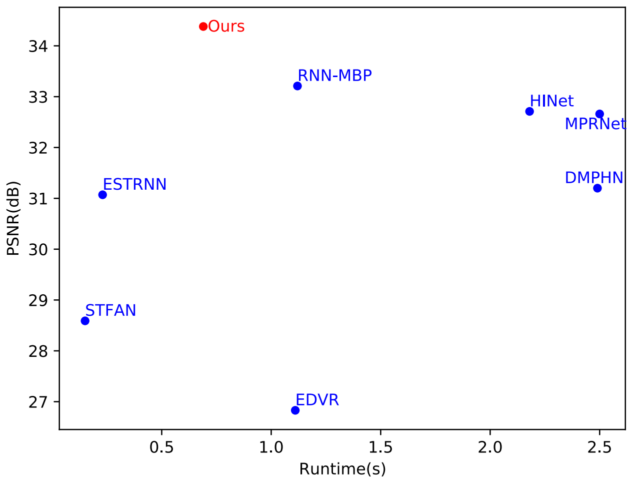


Fig. 1 PSNR versus running time of deblurring video frames comparison of our model and other competing video deblurring methods on GOPRO dataset

on deep learning. Conventional image deblurring methods (Harmeling et al., 2010; Hirsch et al., 2011; Kohler, 2012; Krishnan et al., 2011) focus on how to predict the blur kernel based on different assumptions, such as nonblind or blind blur prior. Nonblind image deblurring methods (Cho et al., 2011; Schmidt et al., 2013; Schuler et al., 2013) rest on the assumption that the blur kernel can be obtained in advance. Due to blind deblurring being an ill-posed problem, these methods (Gupta et al., 2010; Jin et al., 2017; Dong et al., 2017; Park & Lee, 2017) often rely on different constraints on hypothetical blur kernels or image statistics. All of these methods rely on parametric prior models to predict the blur kernel and sharp images.

In recent years, many achievements have been made in the field of computer vision based on deep learning methods. Therefore, researchers began to focus on video-deblurring algorithms based on the deep learning network. In the spatial domain, convolutional neural networks are usually used to extract spatial features that are expressive of motion information of video frames. CNN-based methods are usually used to extract the interframe relationship of video frames by concatenating adjacent frames as input to the model. Su et al. (2017) build an encoder-decoder network for video deblurring composed of CNN layers and residual structures. To estimate the motion state of real-world blur videos, some methods (Sim & Kim, 2019) use the spatial features of adjacent frames to make predictions for per-pixel blur kernels. Then, they are convoluted with the blur kernel on the corresponding blur frames to predict sharp images. In addition, there are some networks that combine other prior information, such as optical flow (Pérez et al., 2013) or traditional image prior (Zhou et al., 2020), which can express the state of motion for video deblurring. The method proposed in Wang et al. (2019) uses the variation in the position of hidden features between frames to build the alignment network and then merges the aligned features into the temporal and spatial domains.

More recently, in Brehm et al. (2020), video deblurring task has been implemented in two stages. In the first stage of single-image deblurring, dilated convolution is introduced to improve the expression ability and reduce the parameters of the deblurring model. In the video deblurring stage, a multiscale feature mixing method is proposed to improve performance. Yan et al. (2020) built a joint learning model for optical flow and video deblurring. The two subtasks were trained in turn and hidden features were integrated during training to improve performance. In the temporal domain, researchers tend to use recurrent neural network (RNN)-based methods to extract temporal relationships from video sequences. In Kim et al. (2017), Kim proposed an RNN structure that merges dynamic temporal features from previous frames with the current frame. Then (Zhou et al., 2019; Zhang et al., 2020b) improves the performance of deblurring by iteratively updating the hidden state through RNN cells. Nah et al. (2019) built a network based on the RNN structure that simultaneously trains the alignment task and the deblurring task. Zhong et al. (2020) extracts the spatial feature of the current frame through the RDB cell and reuses the RDB cell to process the next frame. Then, features that contain temporal and spatial attributes are used to reconstruct the sharp frame through a global attention model. Most recently, multi-attention CNN (Wang et al., 2021), deep dynamic scene deblurring (Zhang et al., 2021a), and occlusion-aware network (Xu et al., 2021) have also been proposed for video deblurring.

2.2 Attention Mechanism

Attention mechanism in general is a learnable guide that can make the network select only important information for processing, to improve the efficiency of the neural network. It originated from natural language processing (NLP) (Bahdanau et al., 2015; Vaswani et al., 2017; Yang et al., 2015) and has been successfully adopted in many fields, such as object recognition (Ba et al., 2015), image generation (Zhang et al., 2018a), and meta-learning (Cao et al., 2019). Multihead attention is proposed in (Vaswani et al., 2017) and deals with NLP tasks such as machine translation through self-attention and co-attention mechanisms. This attention mechanism replaces the LSTM and CNN modules as the feature extractor, allowing the model to process the time-series data at the same time. In addition, it can allow parallel computation, reduce training time, and reduce performance degradation due to long-term dependencies.

Most recently, researchers have introduced this self-attention mechanism into the field of computer vision. The network proposed in Dosovitskiy et al. (2021) extracts the features of all the input images through the self-attention layer. The memory requirement of this attention mechanism is the quadratic power of the input image capacity, which seriously hinders the applicability of self-attention to long-sequence and multidimensional input data. To reduce the parameters of the self-attention model, researchers have proposed many improved methods. The presence of LambdaNetworks (Bello, 2021) provides a way to solve this problem by converting the context to a single linear function (Lambda Layer), which allows the network to capture long-range interactions without having to build expensive attention maps. Recently, the self-attention model has been applied to video deblurring tasks (Kim et al., 2018; Gast & Roth, 2019; Purohit & Rajagopalan, 2020). In Purohit and Rajagopalan (2020), the video deblurring method builds local connections in different spatial locations through the self-attention layer. However, these methods based on self-attention mechanism only focus on local features at different locations in one image, which cannot effectively capture the spatial-temporal relationships within the entire video sequence. In this paper, we propose an efficient temporal fusion model to capture blur information by focusing on local features throughout the video sequence.

2.3 Memory Networks

Memory network is a learning model (Sukhbaatar et al., 2015) that stores and uses additional information to solve current tasks. In Sukhbaatar et al. (2015), the author thinks that traditional deep learning models (RNN, LSTM, GRU, etc.) (Cho et al., 2014b; Chung et al., 2014; Cho et al., 2014a) use hidden states or the attention mechanism as their memory function, but the memory generated by this method is too small to accurately record all the content expressed in a paragraph. Much information is lost in encoding the input into dense vectors. Therefore, the author proposed a reading and writing external memory module and combined it with an inference component to train, and finally obtained a flexible memory module. Memory networks generally consist of a memory cell and an update mechanism. The memory cell stores information useful for solving the task at hand, usually in the form of key-value pairs or vectors, such as diagrams in Wikipedia or vector representations of text. The memory network will query and read the memory cell according to the update mechanism and update the contents of the memory cell at the same time. Finally, the memory network reads relevant information from the memory unit according to the current problem. Memory networks were proposed and developed from NLP (Sukhbaatar et al., 2015; Kumar et al., 2016; Liu & Perez, 2017). Because memory networks can retain historical information, they are often combined with RNN to solve temporal problems, such as video caption (Lin & Zhang, 2021; Ai et al., 2020) or object tracking (Zhou et al., 2022). Memory networks have also been used for low-level enhancement tasks, such as image deblurring (Tai et al., 2017; Zhang et al., 2020a). Tai et al. (2017) introduces a memory block, consisting of a recursive unit and a gate unit, to extract persistent memory through an adaptive learning process.

In this paper, we store the local spatial-temporal information learned by the temporal fusion model in the memory cell. Furthermore, we designed a novel update mechanism. In this way, our model can extract relevant information from historical motion information saved in the memory cell for video deblurring.

3 Proposed Approach

In this section, an overview of the proposed model will be presented first. We will then discuss the details of each component of the proposed model.

3.1 Overall Architecture

The blurred information between adjacent frames often varies spatially and temporally. Additionally, the spatial-temporal relationship among video sequences is spatially local. Based on this observation, the key to the task of video deblurring is how to adaptively capture and utilize the local spatial-temporal relationship among input frames. Recently, researchers have begun to use the attention mechanism to exploit the spatial and temporal relationship in video frames (Purohit & Rajagopalan, 2020; Lei et al., 2020). They focus on the spatial-temporal features from video frames to extract motion-related information. However, it is often difficult to accurately extract the local motion state using coarse-grained attention on the global feature.

In this paper, we build a hierarchical encoder to extract high-level features from video frames. Then, at the bottleneck layer, we collect local features from the input frames at each position to construct local feature groups. To capture and exploit the local spatial-temporal information within the local feature groups, we designed a temporal fusion network that can use the information stored in the memory cell to guide the model to capture the local spatial-temporal information more accurately. In addition, we proposed a multiscale and multihop reconstruction memory network that can make full use of the information on each scale of the previous video frame to guide the reconstruction task.

The architecture of the proposed model is shown in Fig. 2a. It consists of three components: encoder-decoder, temporal fusion network, and reconstruction memory network. Each encoder and decoder contains down-sampling or up-sampling layers with R RDB (Zhang et al., 2018b, 2021) layers, as shown in Fig. 2b (the encoder architecture). And R

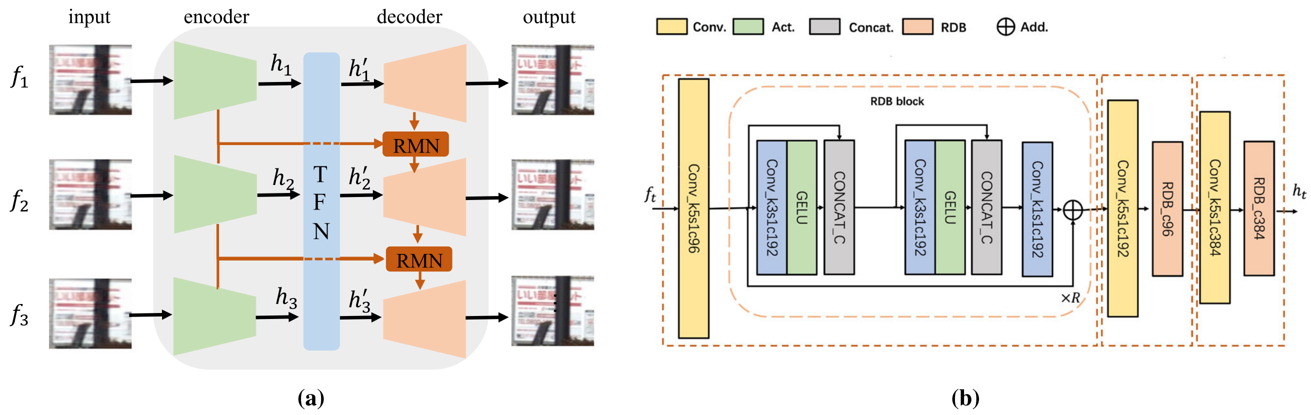


Fig. 2 **a** The architecture of the proposed network. Our model consists of encoder-decoder, temporal fusion network (TFN), and reconstruction memory network (RMN). **b** The architecture of the encoder. f_t refers to the tth input frame, where $t \in \{1 \cdots T\}$, and T is the number of input frames. We make $T = 3$ to facilitate the presentation of the model structure. For details of each convolution layer and each RDB layer, k, s, and c denote the size, stride, and channels of the kernel, respectively. h_t refers to the output feature extracted by the encoder. And h_t is mapped to $h_t^{'}$ by the TFN module

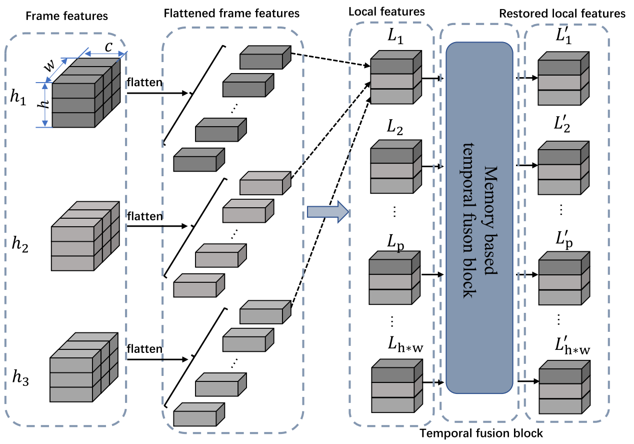


Fig. 3 The architecture of the temporal fusion module. $h_t \in R^{h \times w \times c}$, $t \in \{1 \cdots T\}$ refers to the final output feature of the encoder, where h, w and c represent the height, width, and number of channels, respectively. $L_p \in R^{T \times c}$, $p \in \{1 \cdots h * w\}$ refers to the local feature at position p collected in the form $\{h_t\}$

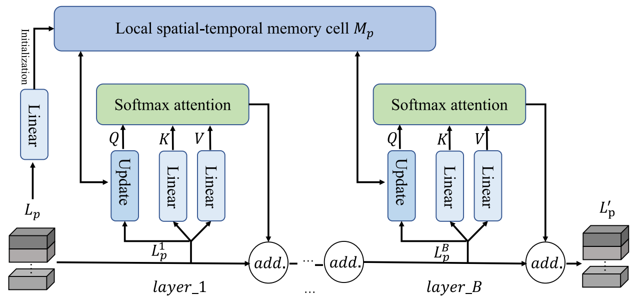


Fig. 4 The detailed architecture of our local spatial-temporal memory-based temporal fusion module. The input and output of the temporal fusion module are L_p and reconstructed $L_p^{'}$. It consists of B attention layers and a memory cell M_p, which is initialized by L_p. L_p refers to the local feature, and $L_p^{'}$ is the output of our module. $add(.)$ refers to the elementary addition

is the number of RDB layers. We denote the tth input frame as f_t, where $t \in \{1 \cdots T\}$, and T is the number of input frames. And we make $T = 3$ to facilitate the presentation of the model structure, which is shown in Fig. 2a. h_t refers to the output feature extracted by the encoder. Then the temporal fusion network will extract the local spatial-temporal relationship from h_t. $h_t^{'}$, the output of the temporal fusion module will contain the extracted spatial-temporal relationship and will be input to the decoder, which has a symmetrical structure with the encoder. During the decoding stage, our RMN will guide the decoder to reconstruct the current frame with the multiscale information of the previous video frame.

3.2 Memory-Based Temporal Fusion

As mentioned above, a key contribution of this paper is the design of the temporal fusion network. The detailed structure of the temporal fusion network is shown in Figs. 3 and 4. The key motivation behind this module is to extract and use the spatial-temporal relationship to guide the video reconstruction task. First, the temporal fusion network will fuse local features based on their correlations. To make their correlations more accurate, we will use the information stored in the memory cell to guide the calculation of the correlations. Based on the calculated correlations, the relevant information from the adjacent frames is extracted and fused adaptively, as shown in Fig. 3. It should be noted that the proposed temporal fusion module can *simultaneously* pay attention to the local features of all input video frames, which is different from other video processing methods based on the attention mechanism (Suin & Rajagopalan, 2021).

Our temporal fusion network (TFN) can solve two important problems in existing video deblurring methods (Dosovitskiy et al., 2021; Zhong et al., 2020). First, some video deblurring algorithms based on recurrent neural networks, such as RNN or LSTM, can process the video in only one direction. In contrast, our *bidirectional* model can work in both directions (backward and forward), so the spatial-temporal relationship can be captured completely. Second, most existing video deblurring algorithms focus only on the overall features of the video frame. Such a coarse-grained attention mechanism cannot accurately capture the spatial-temporal information of every object in the video. In contrast, our temporal fusion network targets the local spatial-temporal features of different positions in the video, which can accurately capture and utilize the local spatial-temporal information for video deblurring.

As shown in Fig. 3, h_t represents the final output feature of the encoder, where $t \in \{1 \cdots T\}$, and $T = 3$ is the number of input frames. Note that $h_t \in R^{h \times w \times c}$ is a high-level feature, where h, w, and c represent the height, width, and number of channels, respectively. First, we flatten $\{h_1 \cdots h_T\}$ and gather the local feature $L_p \in R^{T \times c}$ at the same position p as shown by the dotted line in Fig. 3, where $p \in \{1 \cdots h * w\}$. The local feature L_p contains the local motion information of the input video sequence that can be captured by our temporal fusion model. Then, our model can store and use this motion information for the reconstruction of images. After the model representation, the local feature L_p will be restored to $L_p^{'} \in R^{T \times c}$. And $L_p^{'}$, $p \in \{1 \cdots h * w\}$, will be reassigned to the new feature map $h_t^{'}$, $t \in \{1 \cdots T\}$, based on their position p, which is shown in Fig. 2a. These new feature maps will finally be used as input to the decoder to generate restored video frames. The workflow of our model is shown in Fig. 3.

To further zoom in, we show the detailed design of our temporal fusion block in Fig. 4. The local feature L_p contains a local spatial-temporal relationship at position p. And $M_p \in R^{T \times c}$ is the local spatial-temporal memory at position p, and is initialized by L_p through a linear layer. First, we map L_p to key-value pairs (K and V) through two different linear layers. In this paper, our aim is to capture the local spatial-temporal relationship by combining historical local spatial-temporal information. Therefore, we make the query (Q), mapped by another linear layer from L_p, to query the relevant information from the memory cell M_p and update the memory cell at the same time. L_p reads and updates the memory cell through the update gate as the following formulas:

$$I_p{}^S = M_p{}^{S-1} + L_p^S \tag{1}$$

$$H_p{}^S = tanh(W_h \odot I_p{}^S) \tag{2}$$

$$F_p{}^S = \sigma(W_f \odot I_p{}^S) \tag{3}$$

$$ST_p{}^S = F_p{}^S \odot H_p{}^S \tag{4}$$

$$LT_p{}^S = (1 - F_p{}^S) \odot M_p{}^{S-1} \tag{5}$$

$$M_p{}^S = ST_p{}^S + LT_p{}^S \tag{6}$$

$$Q = W_q \odot M_p{}^S \tag{7}$$

where $L_p^S \in R^{T \times c}$ and $M_p^{S-1} \in R^{T \times c}$, T is the number of input frames, $S \in \{1, 2, 3, \cdots, B\}$, B is the number of our temporal fusion layers, and c is the number of channels. L_p^S represents the local spatial-temporal feature extracted by the $(S-1)$th temporal fusion layer, and L_p^1 is the local spatial-temporal feature L_p. $M_p{}^S$ stands for the local spatial-temporal memory cell M_p after the Sth update. In particular, M_p^0 is the initial memory cell that is initialized by L_p through a linear layer. Because the local feature L_p contains the local spatial-temporal information of all input video frames. So, we can initialize the local spatial-temporal memory cell $M_p{}^0$ by L_p using a simple linear layer or a convolution layer.

Note that $I_p{}^S$ is the input information of the memory update block in Sth temporal fusion layer, which is simply calculated by adding the current spatial-temporal feature L_p^S and the historical spatial-temporal feature $M_p{}^{S-1}$. The function of Eq. (2) is to calculate the hidden memory state $H_p{}^S$, which is simply calculated through the linear layer and the activation function of tanh. $W_h \in R^{T \times c}$ represents the parameter of the linear layer. Because the updated $M_p{}^S$ can be reached through some filtering operations in $H_p{}^S$, we call $H_p{}^S$ the hidden memory state. In fact, we can already use $H_p{}^S$ as our new memory cell $M_p{}^S$ just as the RNN model does. However, there will be serious problems in this way, among which the most important is that the updated $M_p{}^S$ is easily affected by the current local spatial-temporal feature L_p^S, so it cannot maintain the long-term memory capacity of historical information.

To ensure that our memory cell can contain both short-term memory stored in $H_p{}^S$ and long-term memory stored in $M_p{}^{S-1}$. We design the operations as Eqs. (3)–(6). The function of Eq. (3) is to construct a filter $F_p{}^S$ for hidden memory state $H_p{}^S$ to obtain useful information $ST_p{}^S$ that we want to retain (we call it short-term memory). $W_f \in R^{T \times c}$ represents the parameter of a linear layer. σ denotes the activation function of the sigmoid. The function of Eq. (5) is to obtain long-term memory $LT_p{}^S$ from the historical memory state $M_p{}^{S-1}$ based on the extracted short-term memory. Therefore, we use $1 - F_p{}^S$ as a filter here. Now, in Eq. (6), we can get the updated memory state $M_p{}^S$, which is calculated by adding short-term memory $ST_p{}^S$ and long-short memory $LT_p{}^S$. Finally, we can get the updated Q from the updated memory cell $M_p{}^S$ for our temporal fusion model, where $W_q \in R^{T \times c}$ represents the parameter of a linear layer.

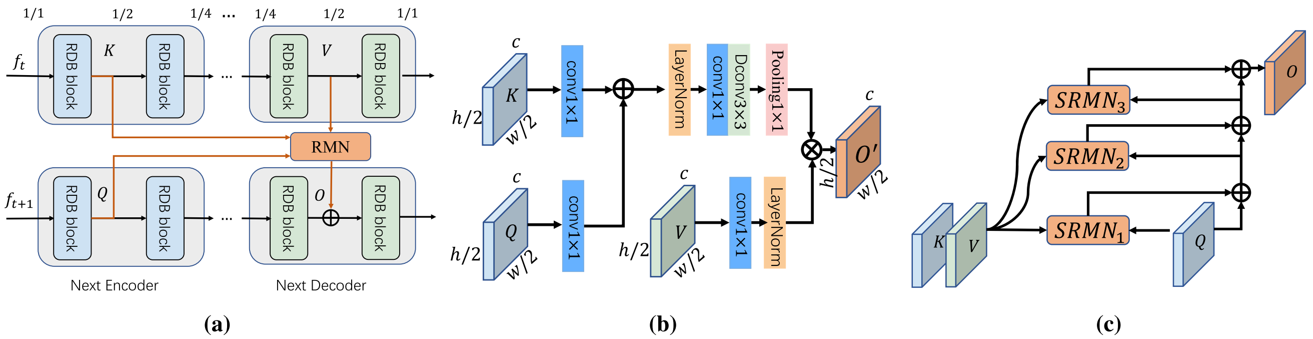


Fig. 5 The structure of the reconstruction network on the 1/2 scale. **a** The overall structure of reconstruction memory network (RMN) in the proposed deep network, where the RDB block is the feature extractor with a down/up-sample layer. **b** Single-hop reconstruction memory network (SRMN). (c) Multi-hop reconstruction memory network (with 3 hops). f_t refers to the tth input frame, where $t \in \{1 \cdots T\}$, and T is the number of input frames

In summary, the core of Eqs. (1)–(6) is to extract useful long-term memory ${LT_p}^S$ from historical information ${M_p}^{S-1}$ and useful short-term memory ${ST_p}^S$ from the current hidden memory state ${H_p}^S$.

After we obtain the feature of Q, K, and $V \in R^{T\times c}$, the attention map can be computed as:

$${A_p}^S = softmax(Q \odot K^T), \tag{8}$$

where a_{ij}, as the element of matrix ${A_p}^S \in R^{T\times T}$, denotes the attention score of the local feature in the ith frame to the local feature in the jth frame at position p. This attention map describes the temporal correlation of local spatial features in the video. So, the attention map can well describe the local spatial-temporal relationship of the video through training. Finally, the restored local feature $L_p^{'}$ can be obtained using the following formula:

$$L_p^{'} = {A_p}^S \odot V \tag{9}$$

Since $L_p^{'}$ and L_p have the same dimension, we can stack the above temporal fusion layers to capture the local spatial-temporal relationship of all video frames more effectively.

3.3 Reconstruction Memory Network

In the reconstruction of the current frame, some existing methods (Zhu et al., 2021; Zamir et al., 2021; Park et al., 2020; Kim et al., 2022; Chu et al., 2022) usually first add or concatenate the features in the encoder and decoder of the previous frame and then fuse with the features of the current frame. However, this method ignores that the two features in the encoder and decoder play different roles, since there is a reconstruction process between the two features. In order to further improve the feature fusion by taking the reconstruction process into account, we propose a multiscale and multihop reconstruction memory network based on the attention mechanism and memory network as Sukhbaatar et al. (2015), which is shown in Fig. 5a.

In Fig. 5a, f_t refers to the tth input frame, where $t \in \{1 \cdots T\}$, and T is the number of input frames. On the 1/2 scale, we first make K and V of the tth frame as the memory cell. Then, in the decoding stage of the $(t+1)$th frame, we will read useful information from V based on the correlation between Q and K. The query result O will be combined with the feature at the corresponding scale in the decoding stage to form a new feature. Similarly, we used RMN to extract information across all scales, except for the bottleneck layer. Of course, since there was no reconstruction process before the first frame, we did not extract the information during the reconstruction of the first frame.

Figure 5b and c show single-hop RMN (SRMN) and multihop RMN (MRMN). In Fig. 5b, on the 1/2 scale, we first need to calculate the correlation between Q and K. So, we first add Q and K after they are mapped by the convolution layer and then extract the deep features through the feature extraction block (including a layer norm, a convolution layer and a depth convolution layer). Finally, a 1×1 spatial pooling layer is used to calculate the correlation of Q and K. Then we can get useful information $O^{'}$ by multiplying the correlation with V, which is mapped by a convolution layer and a layer norm.

Figure 5c shows the structure of multihop RMN. For simplicity, we set the hop number of the network as 3. After calculating the single-hop query, the calculated result $O^{'}$ will be added to the original Q as input to the next SRMN. The final output of the MRMN is expressed as O, which contains useful information about the previous frame.

So far, we have described single-hop and multi-hop reconstruction memory networks at 1/2 scale. Except for the bottleneck layer, the multi-hop reconstruction memory network at other scales constitutes our proposed multi-hop and multiscale reconstruction memory network.

4 Experiments

4.1 Dataset

First, we test our network on GOPRO dataset (Nah et al., 2017), which is a public benchmark dataset for the video deblurring task. GOPRO is synthesized by averaging high-FPS videos. These videos consist of successive frames in a variety of scenarios, which are captured by handheld devices such as the GoPro Hero 4 black and the iPhone 6 s. This video deblurring dataset consists of a quantitative and qualitative subset. There are 71 videos that contain 6708 synthetic blurry frames generated by averaging seven adjacent frames in a quantitative subset. These video frames are 1280×720 in size. The videos in the qualitative subset consist of 22 different scenes without ground truth data. To reduce computational requirements and facilitate comparisons with other video deblurring methods, we choose the subset used in Nah et al. (2019). There are 22 training videos and 11 evaluation videos in this subset. The beam-splitter dataset (BSD) generated by a beam splitter system is a real-world dataset built for video deblurring. This beam splitter system consists of two cameras with the same configurations but different exposure schemes to generate blurry/sharp video pairs. The video has a size of 1280×720. In our experiment, we use the subset $BSD\_2ms16ms$ to test the deblurring performance of our model in the real-world dataset, where $2ms$ and $16ms$ represent the exposure times of the two cameras, respectively.

4.2 Implementation Details

During the training stage, we train our model for 800 epochs using the ADAM optimizer (Kingma & Ba, 2015). We set the initial learning rate to 10^{-4}. We use the random patch of size 256×256 in 5 consecutive frames as input to train our model. For a fair comparison, we used the same data augmentation processes for each model, such as image normalization, horizontal, and vertical inversion. In the test stage, our model could reconstruct five consecutive frames of images at a time. And we will splice the 256×256 image blocks into a complete image for testing. We implemented our network using the PyTorch framework and 4 NVIDIA RTX3090 GPU cards. The training batch is set to 4, and each card used about 17GB memory. The test batch is set to 1, and each card used about 10GB memory. The total training time of the model under these conditions is about 3 days. The loss function is defined as L_2 loss for our model as follows:

$$L_2 = \frac{1}{TCHW} \sum_{t}^{T} \| P_t^{'} - P_t \|^2, \tag{10}$$

where T, C, H, W refer to the number of input frames and channels, as well as the height and width of each frame. $P_t^{'}$ refers to the tth generated sharp frame, and P_t denotes the ground truth of the tth frame.

4.3 Model Analysis

First, we compare our network with other SOTA video deblurring methods on the GOPRO dataset. There are many variants of our model that can be adjusted by changing the model parameters. For example, we name our model as $T_9 R_3 B_6 C_{96}$, where $T_\#$ means the number of input frames is # (such as 9), $R_\#$ means the number of RDB layers is #, $B_\#$ means the number of temporal fusion layers is #, and $C_\#$ means that the number of feature channels in our model is #. In general, the larger $C_\#$ is, the higher computation cost is required. We use PSNR and SSIM (Hore & Ziou, 2010) as evaluation indicators to perform a quantitative analysis of the video deblurring performance in Table 1. Apparently, our model reached the highest PSNR, which means that our model can effectively use the local motion relationship between video frames to improve the deblurring performance of the model. And our model has achieved a significant performance improvement in terms of PSNR metrics (over $1dB$) over existing state-of-the-art video deblurring methods. Furthermore, we also tested the speed of the model in video deblurring, and the experimental results showed that our model achieved a good balance in the speed and performance of video reconstruction. Because our model can reconstruct all input video frames, unlike other models that need to rely on multiple video frames to predict the middle frame. Therefore, our model has obvious advantages in reconstruction speed. Then we also compare our network with other SOTA video deblurring methods on the BSD dataset, as shown in Table 2. The experimental results show that our model can also perform well on real-world datasets.

To further verify the effectiveness of our model in the video deblurring task, we show the deblurred images generated by our model with other models in Fig. 6. The blurry scenes displayed in the first row include the global motion blur of camera movement and the local motion blur of character movement, while the second row shows only the local motion blur of object movement. When comparing these generated deblurred images, we can see that our model has made great progress in restoring the local details of the image in these two blurry scenes. Similarly, visual comparisons were made on the BSD testing dataset, as shown in Fig. 7. By

Table 1 Quantitative results on GOPRO

Networks	PSNR	SSIM	Params (M)	FLOPS (G)	Time (s)
EDVR (Wang et al., 2019)	26.83	0.843	20.6	194.2	1.11
STFAN (Zhou et al., 2019)	28.59	0.861	5.37	35.4	0.15
ESTRNN (Zhong et al., 2020)	31.07	0.902	18.12	206.7	0.23
DMPHN (Zhang et al., 2019)	31.2	0.940	21.7	–	2.49
BANet (Tsai et al., 2021)	32.44	0.957	–	–	–
MPRNet (Zamir et al., 2021)	32.66	0.9590	20.1	760.1	2.50
HINet (Chen et al., 2021)	32.71	0.9590	88.7	170.7	2.18
RNN-MBP (Zhu et al., 2021)	33.32	0.9627	16.4	496.0	1.12
Ours	34.48	0.9573	69.86	353.05	0.69

Table 2 Quantitative results on BSD

Networks	PSNR	SSIM
STRCNN (Kim et al., 2017)	30.33	0.902
DBN (Su et al., 2017)	31.75	0.922
IFIRNN (Nah et al., 2019)	31.53	0.919
ESTRNN (Zhong et al., 2020)	31.95	0.925
Ours	33.18	0.957

comparing the quality of the generated images with the corresponding PSNR scores, it can be seen that our model can not only accurately recover the global blur, but also be better at recovering the details of the image, such as the scenes of the license plate number and the text of the display board.

4.4 Ablation Study

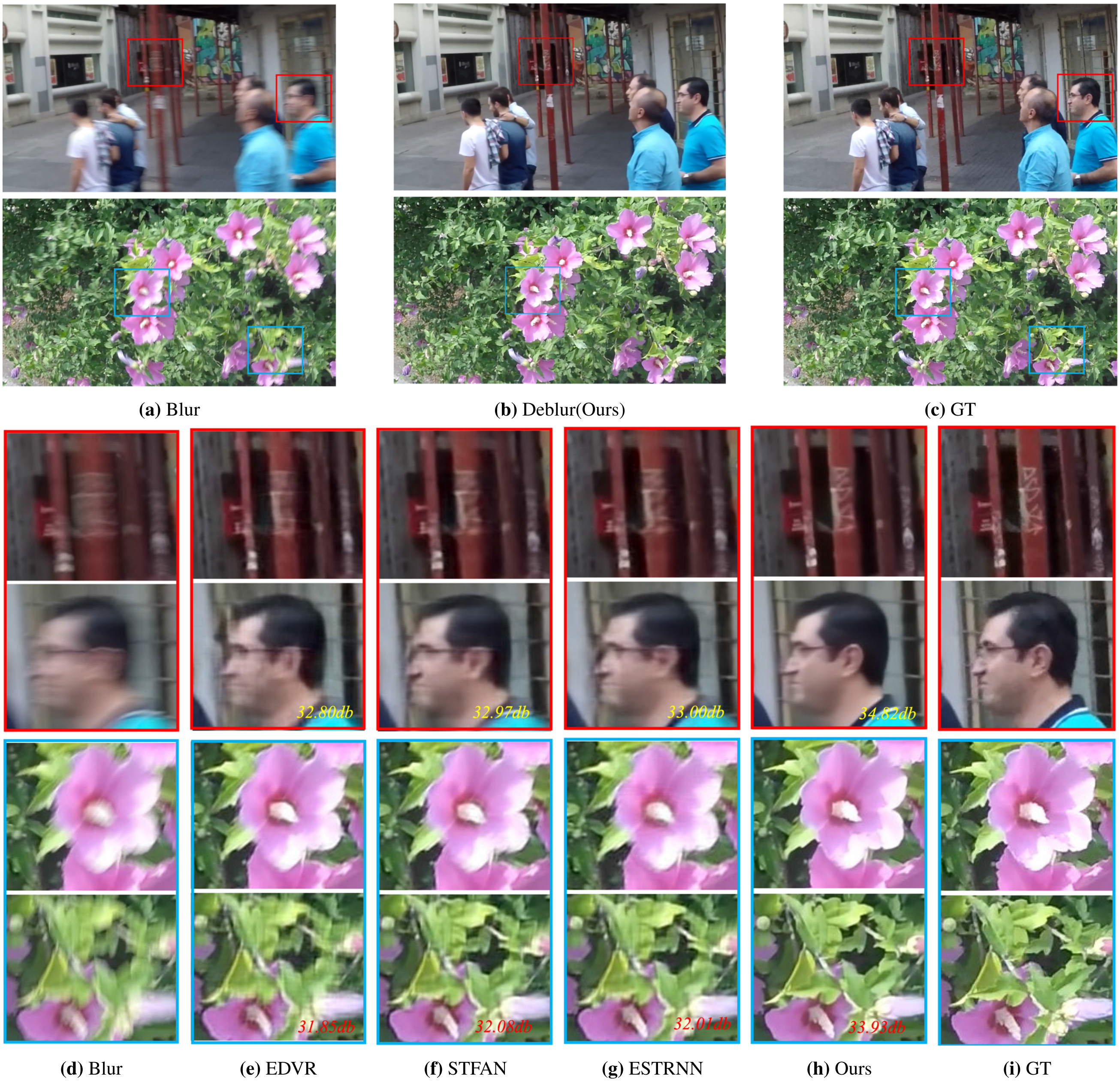
4.4.1 Ablation Study of Hyperparameters

We are interested in the effect of some adjustable parameters of the model on the video deblurring task. Therefore, we vary $T_\#$ (the number of input frames), $R_\#$ (the number of RDB layers) and $B_\#$ (the number of temporal fusion layers) to perform a comparative experiment. The comparison results are shown in Table 3. The comparison result of the RDB layers ($T_3R_1B_3$, $T_3R_2B_3$, $T_3R_3B_3$ and $T_3R_4B_3$ in Table 3) shows that more RDB layers will achieve better performance. The role of the RDB layer in our model is to enhance the feature expression by increasing the receptive field. These enhanced features provide expressive local information for the temporal fusion layer. With the increase of RDB layers, the PSNR score of the model increased by 0.88 dB, 0.62 dB, and 0.56 dB, respectively. The comparison result of the temporal fusion layers ($T_5R_2B_3$,$T_5R_2B_6$, $T_5R_2B_9$ and $T_5R_2B_{12}$ in Table 3) shows that with the increase of the temporal fusion layers, the performance of our model also continues to improve. This proves that our temporal fusion model can achieve long-term memory capability with the help of the memory cell and the update mechanism. The comparison result of the number of input frames ($T_3R_2B_3$, $T_5R_2B_3$ and $T_7R_2B_3$ in Table 3) shows that when the number of input video frames is 5, the maximum improvement of the PSNR score is around 0.4 dB. The more video frames are stored, the higher computational cost is required for the model. Therefore, the number of input videos for the model is generally set to 9 in our ablation experiments to maintain a good balance between model performance and computational cost.

4.4.2 Ablation Study of Window Size in the Temporal and Spatial Domain

Now we analyze the window size of our model in the spatial and temporal domains, and the results are shown in Table 4.

1. *Why global in the temporal domain.* The experiment in Table 3 has shown that our model will improve with increasing input frames T, which indicates that more input frames will provide more useful information for the video reconstruction task. So, our temporal fusion model will perform better by global fusion in the temporal domain.
2. *Why local in the spatial domain.* In general, in the bottleneck layer, the resolution of the video frame is very low in the spatial domain. Therefore, the spatial correlation between the video frames decreases with the distance of the pixels. If the spatial window of the temporal fusion layer is amplified, irrelevant noise will be introduced, which leads to the failure of the model to accurately extract the spatial-temporal relationship of the video, resulting in the deterioration of the model performance. First, in our experiment, the frame feature h_t has the size of 32 × 32 × 768, where 32 is the spatial size and 768 is the number of channels. The number of input frames T is 5. In Table 4, we set different window sizes to change the spatial size of the local spatial-temporal feature L_p. Thus, we can find out which spatial window size is more conducive to the video reconstruction task. For example, we set the spatial window as 1×1 (local in the

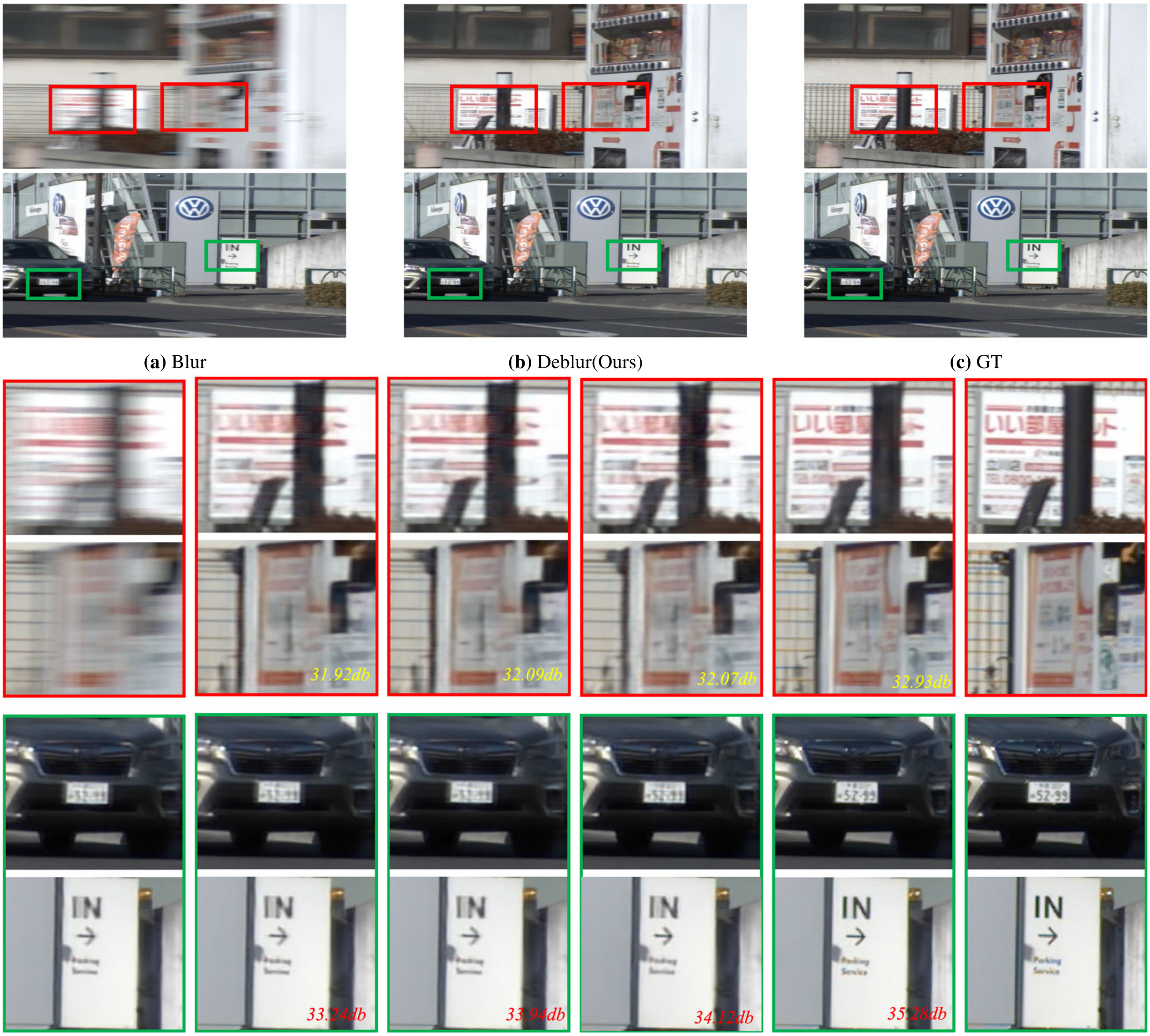


(a) Blur **(b)** Deblur(Ours) **(c)** GT

(d) Blur **(e)** EDVR **(f)** STFAN **(g)** ESTRNN **(h)** Ours **(i)** GT

Fig. 6 Visual comparisons on the GOPRO test dataset. The PSNR score for each image is shown in the lower right corner of the image

spatial domain) and the time window as T (global in the temporal domain). Therefore, our local spatial-temporal feature L_p has the size of 5×1×1×768 (simplified to 5×768). Therefore, our temporal fusion model is local in the spatial domain and global in the temporal domain. The first experiment in Table 4 shows that our spatial-temporal memory M is helpful for our temporal fusion model to obtain accurate spatial-temporal relationships. Then we tried to gradually expand the spatial window of L_p, and this is the same strategy as the SwinTransformer (Liu et al., 2021) which removed the shift window operation from it. Experiments with spatial size of 2×2 and 4×4 all showed that with increasing spatial window, the attention model would fuse more irrelevant information, resulting in degraded performance compared to the model with 1×1 spatial window. Furthermore, if the spatial window is expanded to the size of the image feature 32×32, which is the same approach as VIT (Dosovitskiy et al., 2021) and is global in the spatial domain, the model performance will be further reduced. These experiments show that local spatial correlation is more conducive to video restoration for our model.



(a) Blur **(b)** Deblur(Ours) **(c)** GT

(d) Blur **(e)** STRCNN **(f)** DBN **(g)** ESTRNN **(h)** Ours **(i)** GT

Fig. 7 Visual comparisons on the BSD test dataset. The PSNR score for each image is shown in the lower right corner of the image

4.4.3 Ablation Study of RMN and TFN

The experiments in Table 5 show the comparison results of our proposed RMN and the common feature fusion method $Conv\&Add$. First, the experiment of our temporal fusion model with the RMN block of 3 hops can improve our model by 0.34 dB. And compared to the $Conv\&Add$ method used in Zhu et al. (2021); Zamir et al. (2021); Park et al. (2020); Kim et al. (2022); Chu et al. (2022), our RMN block with 3 hops can exceed it by 0.26 dB. Finally, it can be shown by the experiments that our proposed RMN can extract useful information more effectively than the $Conv\&Add$ method.

The effects of TFN and RMN on our model are shown in Table 6. It can be found that TFN has the greatest influence on the model. Only using RMN cannot well reconstruct a sharp video sequence because RMN can only mine the useful information of the previous frame, while TFN can simultaneously mine the local spatial-temporal relationships of all input video frames. This also shows that our idea of combining TFN with RMN is correct.

Table 3 Performance comparisons of the proposed method by varying adjustable parameters $T_{\#}$ $R_{\#}$ $B_{\#}$ on GOPRO dataset

Model	PSNR	SSIM
$T_3R_1B_3$	29.77	0.891
$T_3R_2B_3$	30.65	0.906
$T_3R_3B_3$	31.27	0.923
$T_3R_4B_3$	31.83	0.952
$T_5R_2B_3$	31.05	0.901
$T_5R_2B_6$	31.35	0.935
$T_5R_2B_9$	31.52	0.947
$T_5R_2B_{12}$	31.64	0.950
$T_7R_2B_3$	31.23	0.923

Note that the number of feature channels was cut in half to speed up training

Table 4 The ablation study of the spatial window size in the temporal fusion layer on GOPRO

Attention model	Window size	Memory cell	PSNR
1	T×1 ×1	✓	31.83
2	T×1×1	✗	31.49
3	T×2×2	✗	31.35
4	T×4×4	✗	31.21
5	T×32×32	✗	30.89

Table 6 Comparative analysis of TFN and RMN on GOPRO

Model	TFN	RMN	PSNR
1	✓	✗	31.83
2	✗	✓	31.58
3	✓	✓	32.17

4.5 Discussions

To demonstrate the effectiveness of our memory-based temporal fusion mechanism, we analyze whether the model can correctly capture the spatial-temporal relationship according to the relationships in the attention map ${A_p}^S \in R^{T\times T}$, which is extracted by our temporal fusion module. And a_{ij}, as the element of matrix ${A_p}^S \in R^{T\times T}$, denotes the attention score of the local feature in the ith frame to the local feature in the jth frame at position p.

First, we calculate the average attention map A_p from all temporal fusion layers as:

$$A_p = \frac{1}{B}\sum_{S=1}^{B} A_p^S \tag{11}$$

where B is the number of temporal fusion layers, and $A_p \in R^{T\times T}$.

And then we use $A_p(\frac{T}{2},) \in R^T$ to represent the attention map of the middle frame to other T frames (including the middle frame) at position p.

All the above attention maps are obtained in the bottleneck layer whose feature has the size of $h \times w$ and $p \in \{1 \cdots h * w\}$. Next, we combine all the attention maps $A_p(\frac{T}{2},)$, $p \in \{1 \cdots h * w\}$ from all the positions in the bottleneck layer into the new attention map $A(\frac{T}{2},) \in R^{h\times\ w\times T}$. The attention map between the middle frame and tth frame can be represented as $A(\frac{T}{2}, t) \in R^{h\times\ w\times 1}$.

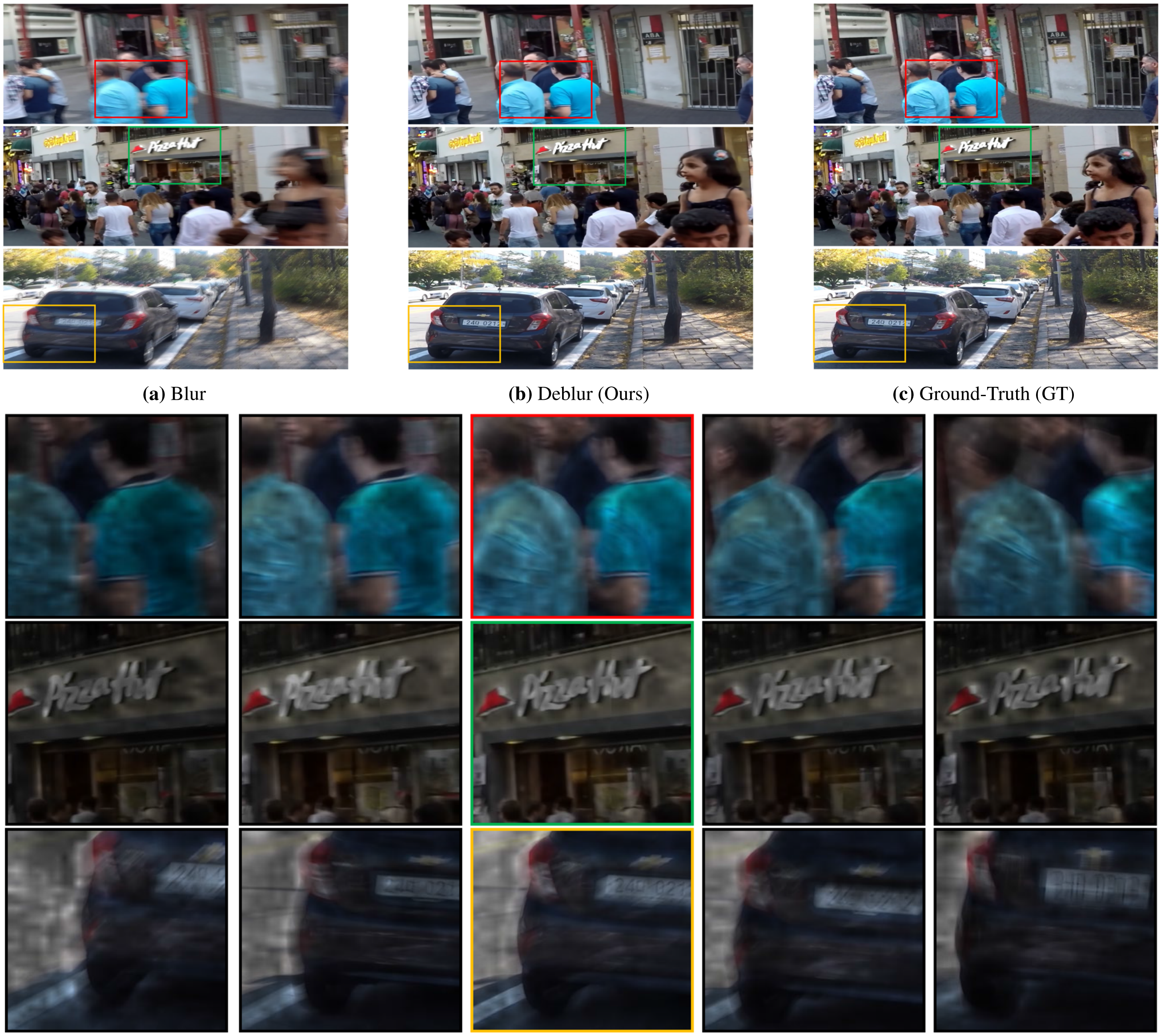
Finally, $A(\frac{T}{2}, t) \in R^{h\times\ w\times 1}$ is expanded to the size of the input image (the size is $H \times W \times 3$), and then each value is normalized to $0 - 1$. And $A(\frac{T}{2}, t) \in R^{H\times\ W\times 1}$ will be multiplied by the tth input image to get the visual attention map, which has the size of $H \times W \times 3$ as shown in Fig. 8d.

Figure 8d shows the model's attention map of the intermediate frame to the other five frames when the intermediate frame is reconstructed. In these maps, the brighter the region, the more attention the model pays to the region. If the position p is set at the center point, it represents the attention scores of the center point in the third frame to the center points in the other five frames. From these attention maps, we can see that our model pays more attention to some blurry areas, such as the contours of the different moving objects. At the same time, these blurred areas represent the trajectories of moving objects. These experimental findings indicate that, with the help of our temporal fusion module, our model can correctly capture the local spatial-temporal information of different objects and solve the problem of video reconstruction by using the motion relations of objects across multiple frames. The frame restored by the proposed temporal fusion model is visually indistinguishable from the ground truth.

Furthermore, we present the two video clips with the lowest and highest PSNR results in the GOPRO dataset in Fig. 9. On the one hand (spatially), video A contains a lot of high-frequency information, such as leaves and branches. And video B mostly contains low-frequency information, such as walls and windows. It is difficult for any video deblur-

Table 5 Ablation study of the proposed reconstruction memory network on the GOPRO

Fusion model	None	Conv&Add	RMN (1 hop)	RMN (2 hops)	RMN (3 hops)
PSNR	31.83	31.91	31.92	32.01	32.17
SSIM	0.952	0.952	0.952	0.953	0.953



(**a**) Blur

(**b**) Deblur (Ours)

(**c**) Ground-Truth (GT)

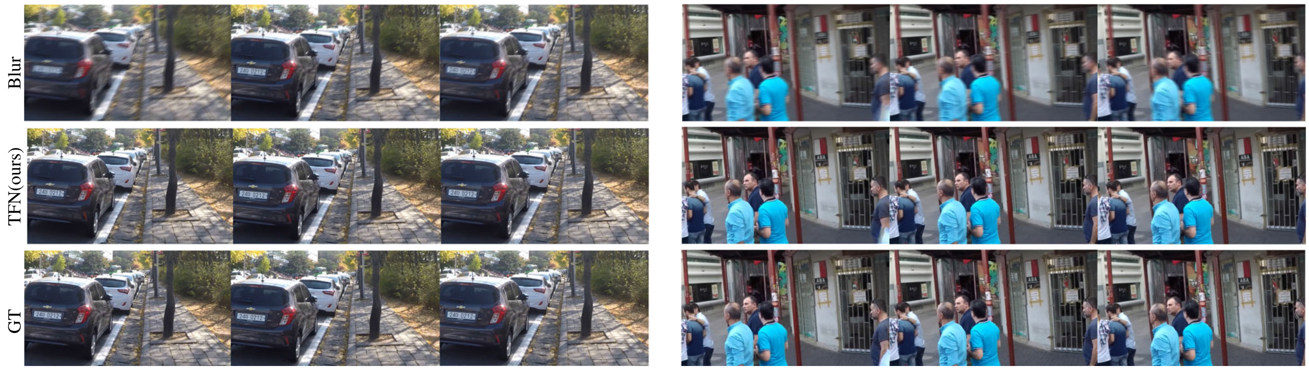
(**d**) Attention map of current frame and adjacent two frames

Fig. 8 Visualizations of attention maps. **a** The input blurred frames. **b** Deblurred frames by the proposed method. **c** The ground truth frames. **d** Attention maps of the middle frame in adjacent frames using the visualization method (Dosovitskiy et al., 2021)

ring model, including ours, to recover the high-frequency details of the image, so the PSNR performance of our model for video A is lower than that for video B. On the other hand (temporally), in video A, only the camera moves and the object remains stationary (i.e., global blur), whereas, in video B, the object and the camera move simultaneously (i.e., local blur). Since our model focuses on local deblurring in the spatial domain and restoring all frames, it is difficult to distinguish global blur from local blur (e.g., in video A), which explains the relatively lower PSNR performance of our model for video A. By contrast, our model is more efficient for video B containing complex local blur because spatial-temporal memory plays the role of implicit motion estimation. In other words, when compared with other video deblurring models, our model can capture and utilize local spatial-temporal information more accurately by combining a temporally bidirectional with a spatially local attention mechanism.

5 Conclusion

In this paper, we have developed a local spatial-temporal memory-based temporal fusion model, which focuses on the



Video A (31.64 dB, lower bound)

Video B (35.16 dB, upper bound)

Fig. 9 The figure shows the two video clips A and B with the lowest and highest PSNR scores in the GOPRO dataset

local features of the input blurry frames in the time domain. The method can collect local features of each position from the input video frames to build the local feature group. The temporal fusion mechanism is used to calculate the correlation in these local feature groups, which represents the local spatial-temporal relationship of the video. Based on the local correlation captured by the model, the reconstructed local features of the video frames will contain useful information from adjacent frames. Additionally, we store the motion information learned by the temporal fusion model in the memory cell, which is helpful in extracting relevant information from historical motion information for video recovery. Finally, the decoder will restore these reconstructed features to sharp frames. To enhance the spatial expression of each video frame, we build an encoder that integrates residual dense blocks with three down-sample layers to extract hierarchical features at multiple scales. To make full use of the multiscale features, we have built a decoder that is symmetric to the encoder and used residual networks to connect the features at each scale. Extensive experimental results on public datasets are reported to justify the superiority of the proposed network over current state-of-the-art video deblurring techniques.

Acknowledgements This work was supported in part by the Natural Science Foundation of China under Grant 61991451, Grant 61836008. Xin Li's work is partially supported by NSF awards CMMI-2146015 and IIS-2114644.

Data Availibility The video deblurring datasets that support the findings of this study are available from the GOPRO Nah et al. (2017) dataset and BSD Nah et al. (2019) dataset.

References

Ai, J., Yang, Y., Xu, X., Zhou, J., & Shen, HT. (2020). CC-LSTM: Cross and conditional long-short time memory for video captioning. In: A. D. Bimbo, R. Cucchiara, S. Sclaroff, G. M. Farinella, T. Mei, M. Bertini, H. J. Escalante, R. Vezzani, (Eds.), *Pattern Recognition. ICPR International Workshops and Challenges—Virtual Event*, January 10-15, 2021, Proceedings, Part VI, vol 12666 (Springer, 2020) Lecture Notes in Computer Science, pp. 353–365.

Ba, J., Mnih, V., & Kavukcuoglu, K. (2015). Multiple object recognition with visual attention. In: Y. Bengio, Y. LeCun (Eds.), *3rd International Conference on Learning Representations, ICLR 2015*, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings.

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In: Bengio Y, LeCun Y (eds) In: *3rd International Conference on Learning Representations, ICLR 2015*, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings

Bello, I. (2021). Lambdanetworks: Modeling long-range interactions without attention. In: *9th International Conference on Learning Representations, ICLR 2021*, Virtual Event, Austria, May 3-7, 2021, OpenReview.net.

Brehm, S., Scherer, S., & Lienhart, R. (2020). High-resolution dual-stage multi-level feature aggregation for single image and video deblurring. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020*, Seattle, WA, USA, June 14-19, 2020, IEEE, pp 1872–1881.

Cao, Y., Chen, T., Wang, Z., & Shen, Y. (2019). Learning to optimize in swarms. In: H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, R. Garnett (eds) *Advances in Neural Information Processing Systems* 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, pp 15018–15028

Chen, L., Lu, X., Zhang, J., Chu, X., & Chen, C. (2021). Hinet: Half instance normalization network for image restoration. In: *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2021*, virtual, June 19-25, 2021, Computer Vision Foundation / IEEE, pp 182–192

Cho, K., van Merrienboer, B., Bahdanau, D., & Bengio, Y. (2014a). On the properties of neural machine translation: Encoder-decoder approaches. In: D. Wu, M. Carpuat, X. Carreras, E. M. Vecchi (Eds.), *Proceedings of SSST@EMNLP 2014, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, Doha, Qatar, 25 October 2014, Association for Computational Linguistics, pp 103–111

Cho, K., van Merrienboer, B., Gülçehre, Ç., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014b). Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: A. Moschitti, B. Pang, W. Daelemans (eds) *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014*, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, ACL, pp 1724–1734

Cho, S., Wang, J., Lee, S. (2011). Handling outliers in non-blind image deconvolution. In: D. N. Metaxas, L. Quan, A. Sanfeliu, L. V. Gool (Eds.), *IEEE International Conference on Computer Vision, ICCV 2011*, Barcelona, Spain, November 6-13, 2011, IEEE Computer Society, pp. 495–502.

Chu, X., Chen, L., Chen, C., & Lu, X. (2022). Improving image restoration by revisiting global information aggregation. In: S. Avidan, G. J. Brostow, M. Cissé, G. M. Farinella, T. Hassner (Eds.), *Computer Vision—ECCV 2022: 17th European Conference*, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part VII, Springer, Lecture Notes in Computer Science, vol 13667, pp 53–71

Chung, J., Gülçehre, Ç., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. CoRR abs/1412.3555

Dong, J., Pan, J., Su, Z., & Yang, M. (2017). Blind image deblurring with outlier handling. In: *IEEE International Conference on Computer Vision*, ICCV 2017, Venice, Italy, October 22-29, 2017, IEEE Computer Society, pp. 2497–2505.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. In: *9th International Conference on Learning Representations, ICLR 2021*, Virtual Event, Austria, May 3-7, 2021, OpenReview.net.

Gast, J., & Roth, S. (2019). Deep video deblurring: The devil is in the details. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 0–0.

Gupta, A., Joshi, N., Zitnick, CL., Cohen, MF., & Curless, B. (2010). Single image deblurring using motion density functions. In: K. Daniilidis, P. Maragos, N. Paragios (eds) *Computer Vision—ECCV 2010, 11th European Conference on Computer Vision*, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part I, Springer, Lecture Notes in Computer Science, vol 6311, pp. 171–184.

Harmeling, S., Hirsch, M., & Schölkopf, B. (2010). Space-variant single-image blind deconvolution for removing camera shake. In: J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, A. Culotta (Eds.), *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010*. Proceedings of a meeting held 6-9 December 2010, Vancouver, British Columbia, Canada, Curran Associates, Inc., pp. 829–837.

Hirsch, M., Schuler, CJ., Harmeling, S., & Schölkopf, B. (2011). Fast removal of non-uniform camera shake. In: D. N. Metaxas, L. Quan, A. Sanfeliu, L. V. Gool (eds) *IEEE International Conference on Computer Vision, ICCV 2011*, Barcelona, Spain, November 6-13, 2011, IEEE Computer Society, pp. 463–470.

Horé, A., & Ziou, D. (2010). Image quality metrics: PSNR versus SSIM. In: *20th International Conference on Pattern Recognition, ICPR 2010*, Istanbul, Turkey, 23-26 August 2010, IEEE Computer Society, pp. 2366–2369.

Jiang, R., Zhao, L., Wang, T., Wang, J., & Zhang, X. (2020). Video deblurring via temporally and spatially variant recurrent neural network. *IEEE Access, 8*, 7587–7597.

Jin, M., Roth, S., & Favaro, P. (2017). Noise-blind image deblurring. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, Honolulu, HI, USA, July 21-26, 2017, IEEE Computer Society, pp. 3834–3842.

Kim, K., Lee, S., & Cho, S. (2022). Mssnet: Multi-scale-stage network for single image deblurring. CoRR abs/2202.09652

Kim, TH., & Lee, KM. (2014). Segmentation-free dynamic scene deblurring. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014*, Columbus, OH, USA, June 23-28, 2014, IEEE Computer Society, pp. 2766–2773.

Kim, TH., Lee, KM., Schölkopf, B., & Hirsch, M. (2017). Online video deblurring via dynamic temporal blending network. In: *IEEE International Conference on Computer Vision, ICCV 2017*, Venice, Italy, October 22-29, 2017, IEEE Computer Society, pp. 4058–4067.

Kim, TH., Sajjadi, MS., Hirsch, M., & Scholkopf, B. (2018). Spatio-temporal transformer network for video restoration. In: *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 106–122

Kingma, DP., & Ba, J. (2015). Adam: A method for stochastic optimization. In: Y. Bengio, Y. LeCun (eds) *3rd International Conference on Learning Representations, ICLR 2015*, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings.

Köhler, R., Hirsch, M., Mohler, BJ., Schölkopf, B., & Harmeling, S. (2012). Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database. In: A. W. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, C. Schmid (eds) *Computer Vision: ECCV 2012—12th European Conference on Computer Vision*, Florence, Italy, October 7-13, 2012, Proceedings, Part VII, Springer, Lecture Notes in Computer Science, vol 7578, pp. 27–40

Krishnan, D., Tay, T., & Fergus, R. (2011). Blind deconvolution using a normalized sparsity measure. In: *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011*, Colorado Springs, CO, USA, 20-25 June 2011, IEEE Computer Society, pp. 233–240.

Kumar, A., Irsoy, O., Ondruska, P., Iyyer, M., Bradbury, J., Gulrajani, I., Zhong, V., Paulus, R., & Socher, R. (2016). Ask me anything: Dynamic memory networks for natural language processing. In: M. Balcan, K. Q. Weinberger (Eds.), *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016*, New York City, NY, USA, June 19-24, 2016, JMLR.org, JMLR Workshop and Conference Proceedings, vol 48, pp. 1378–1387.

Lee, HS., Kwon, J., & Lee, KM. (2011). Simultaneous localization, mapping and deblurring. In: D. N. Metaxas, L. Quan, A. Sanfeliu, L. V. Gool (Eds.), *IEEE International Conference on Computer Vision, ICCV 2011*, Barcelona, Spain, November 6-13, 2011, IEEE Computer Society, pp. 1203–1210.

Lei, J., Wang, L., Shen, Y., Yu, D., Berg ,TL., & Bansal, M. (2020). MART: memory-augmented recurrent transformer for coherent video paragraph captioning. In: D. Jurafsky, J. Chai, N. Schluter, J. R. Tetreault (eds) *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, Online, July 5-10, 2020, Association for Computational Linguistics, pp. 2603–2614.

Lin, J., & Zhang, C. (2021). A new memory based on sequence to sequence model for video captioning. In: *2021 International Conference on Security, Pattern Analysis, and Cybernetics, SPAC 2021*, Chengdu, China, June 18-20, 2021, IEEE, pp. 470–476.

Liu, F., Perez, J. (2017). Gated end-to-end memory networks. In: M. Lapata, P. Blunsom, A. Koller (eds) *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017*, Valencia, Spain, April 3-7, 2017, Volume 1: Long Papers, Association for Computational Linguistics, pp 1–10

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., & Guo, B. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. In: *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021*, Montreal, QC, Canada, October 10-17, 2021, IEEE, pp. 9992–10002.

Nah, S., Kim, TH., Lee, & KM. (2017). Deep multi-scale convolutional neural network for dynamic scene deblurring. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, Honolulu, HI, USA, July 21-26, 2017, IEEE Computer Society, pp. 257–265.

Nah, S., Son, S., & Lee, KM. (2019). Recurrent neural networks with intra-frame iterations for video deblurring. In: *IEEE Conference* *on Computer Vision and Pattern Recognition, CVPR 2019*, Long Beach, CA, USA, June 16-20, 2019, Computer Vision Foundation / IEEE, pp. 8102–8111.

Pan, J., Hu, Z., Su, Z., Lee, H., & Yang, M. (2016). Soft-segmentation guided object motion deblurring. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, Las Vegas, NV, USA, June 27-30, 2016, IEEE Computer Society, pp. 459–468.

Park, D., Kang, DU., Kim, J., & Chun, SY. (2020). Multi-temporal recurrent neural networks for progressive non-uniform single image deblurring with incremental temporal training. In: A. Vedaldi, H. Bischof, T. Brox, J. Frahm (eds) *Computer Vision—ECCV 2020: 16th European Conference*, Glasgow, UK, August 23-28, 2020, Proceedings, Part VI, Springer, Lecture Notes in Computer Science, vol 12351, pp. 327–343.

Park, H., & Lee, KM. (2017). Joint estimation of camera pose, depth, deblurring, and super-resolution from a blurred image sequence. In: *IEEE International Conference on Computer Vision, ICCV 2017*, Venice, Italy, October 22-29, 2017, IEEE Computer Society, pp. 4623–4631.

Pérez, J. S., Meinhardt-Llopis, E., & Facciolo, G. (2013). TV-L1 optical flow estimation. *Image Process Line, 3*, 137–150.

Purohit, K., & Rajagopalan, AN. (2020). Region-adaptive dense network for efficient motion deblurring. In: *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The hirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020*, New York, NY, USA, February 7-12, 2020, AAAI Press, pp. 11882–11889.

Schmidt, U., Rother, C., Nowozin, S., Jancsary, J., & Roth, S. (2013). Discriminative non-blind deblurring. In: *2013 IEEE Conference on Computer Vision and Pattern Recognition*, Portland, OR, USA, June 23-28, 2013, IEEE Computer Society, pp. 604–611.

Schuler, CJ., Burger, HC., Harmeling, S., & Schölkopf, B. (2013). A machine learning approach for non-blind image deconvolution. In: *2013 IEEE Conference on Computer Vision and Pattern Recognition*, Portland, OR, USA, June 23-28, 2013, IEEE Computer Society, pp. 1067–1074.

Sim, H., & Kim, M. (2019). A deep motion deblurring network based on per-pixel adaptive kernels with residual down-up and up-down modules. In: *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019*, Long Beach, CA, USA, June 16-20, 2019, Computer Vision Foundation/IEEE, pp. 2140–2149.

Su, S., Delbracio, M., Wang, J., Sapiro, G., Heidrich, W., & Wang, O. (2017). Deep video deblurring for hand-held cameras. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, Honolulu, HI, USA, July 21-26, 2017, IEEE Computer Society, pp. 237–246.

Suin, M., & Rajagopalan, A. (2021). Gated spatio-temporal attention-guided video deblurring. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7802–7811.

Sukhbaatar, S., Szlam, A., Weston, J., & Fergus, R. (2015). End-to-end memory networks. In: *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015*, December 7-12, 2015, Montreal, Quebec, Canada, pp. 2440–2448.

Tai, Y., Yang, J., Liu, X., & Xu, C. (2017). Memnet: A persistent memory network for image restoration. In: *IEEE International Conference on Computer Vision, ICCV 2017*, Venice, Italy, October 22-29, 2017, IEEE Computer Society, pp. 4549–4557.

Tsai, F., Peng, Y., Lin, Y., Tsai. C., & Lin, C. (2021). Banet: Blur-aware attention networks for dynamic scene deblurring. CoRR abs/2101.07518.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, AN., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In: Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, pp. 5998–6008.

Wang, X., Chan, KCK., Yu, K., Dong. C., & Loy, CC. (2019). EDVR: video restoration with enhanced deformable convolutional networks. In: *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019*, Long Beach, CA, USA, June 16-20, 2019, Computer Vision Foundation / IEEE, pp. 1954–1963.

Wang, XZT., Jiang, R., Zhao, L., & Xu, Y. (2021). Multi-attention convolutional neural network for video deblurring. In: *IEEE Transactions on Circuits and Systems for Video Technology*

Wu, J., Yu, X., Liu, D., Chandraker, M., & Wang, Z. (2020). DAVID: dual-attentional video deblurring. In: *IEEE Winter Conference on Applications of Computer Vision, WACV 2020*, Snowmass Village, CO, USA, March 1-5, 2020, IEEE, pp. 2365–2374.

Wu, Y., Ling, H., Yu, J., Li, F., Mei, X.,& Cheng, E. (2011). Blurred target tracking by blur-driven tracker. In: D. N. Metaxas, L. Quan, A. Sanfeliu, L. V. Gool (Eds.), *IEEE International Conference on Computer Vision, ICCV 2011*, Barcelona, Spain, November 6-13, 2011, IEEE Computer Society, pp. 1100–1107.

Xu, L., & Jia, J. (2010). Two-phase kernel estimation for robust motion deblurring. In: K. Daniilidis, P. Maragos, N. Paragios (Eds.), *Computer Vision—ECCV 2010, 11th European Conference on Computer Vision*, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part I, Springer, Lecture Notes in Computer Science, vol 6311, pp. 157–170.

Xu, Q., Pan, J., & Qian, Y. (2021). Learning an occlusion-aware network for video deblurring. In: *IEEE Transactions on Circuits and Systems for Video Technology*.

Yan, Y., Wu, Q., Xu, B., Zhang, J., & Ren, W. (2020). Vdflow: Joint learning for optical flow and video deblurring. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020*, Seattle, WA, USA, June 14-19, 2020, IEEE, pp. 3808–3816.

Yang, J., Nguyen, MN., San, PP., Li, X., & Krishnaswamy, S. (2015). Deep convolutional neural networks on multichannel time series for human activity recognition. In: Q. Yang, M. J. Wooldridge (Eds.), *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015*, Buenos Aires, Argentina, July 25-31, 2015, AAAI Press, pp. 3995–4001.

Zamir, SW., Arora, A., Khan, SH., Hayat, M., Khan, FS., Yang, M., & Shao, L. (2021). Multi-stage progressive image restoration. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021*, virtual, June 19-25, 2021, Computer Vision Foundation/IEEE, pp. 14821–14831.

Zhang, H., & Carin, L. (2014). Multi-shot imaging: Joint alignment, deblurring, and resolution-enhancement. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014*, Columbus, OH, USA, June 23-28, 2014, IEEE Computer Society, pp. 2925–2932.

Zhang, H., Wipf, DP., & Zhang, Y. (2013). Multi-image blind deblurring using a coupled adaptive sparse prior. In: *2013 IEEE Conference on Computer Vision and Pattern Recognition*, Portland, OR, USA, June 23-28, 2013, IEEE Computer Society, pp 1051–1058

Zhang, H., Wipf, D. P., & Zhang, Y. (2014). Multi-observation blind deconvolution with an adaptive sparse prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 36*(8), 1628–1643.

Zhang, H., Goodfellow, IJ., Metaxas, DN., & Odena, A. (2018a). Self-attention generative adversarial networks. CoRR abs/1805.08318.

Zhang, H., Dai, Y., Li, H., & Koniusz, P. (2019). Deep stacked hierarchical multi-patch network for image deblurring. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019*, Long Beach, CA, USA, June 16-20, 2019, Computer Vision Foundation/IEEE, pp. 5978–5986.

Zhang, J., Pan, J., Wang, D., Zhou, S., Wei, X., Zhao, F., Liu, J., & Ren, J. (2021). Deep dynamic scene deblurring from optical flow. *IEEE Transactions on Circuits and Systems for Video Technology, 32*, 8250–8260.

Zhang, X., Gao, P., Zhao, K., Liu, S., Li, G., & Yin, L. (2020). Image restoration via deep memory-based latent attention network. *IEEE Access, 8*, 104728–104739.

Zhang, X., Jiang, R., Wang, T., & Wang, J. (2020). Recursive neural network for video deblurring. *IEEE Transactions on Circuits and Systems for Video Technology, 31*(8), 3025–3036.

Zhang, Y., Tian, Y., Kong, Y., Zhong, B., & Fu, Y. (2018b). Residual dense network for image super-resolution. In: *2018 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018, IEEE Computer Society, pp. 2472–2481.

Zhang, Y., Tian, Y., Kong, Y., Zhong, B., & Fu, Y. (2021). Residual dense network for image restoration. *Transactions on Pattern Analysis and Machine Intelligence, 43*(7), 2480–2495.

Zhong, Z., Gao, Y., Zheng, Y., & Zheng, B. (2020). Efficient spatio-temporal recurrent neural network for video deblurring. In: A. Vedaldi, H. Bischof, T. Brox, J. Frahm (eds) Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part VI, Springer, Lecture Notes in Computer Science, vol 12351, pp. 191–207.

Zhou, S., Zhang, J., Pan, J., Zuo, W., Xie, H., & Ren, J. S. J. (2019). Spatio-temporal filter adaptive network for video deblurring. In: *2019 IEEE/CVF International Conference on Computer Vision*, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019, IEEE, pp 2482–2491

Zhou, Y., Xu, J., Tasaka, K., Chen, Z., & Li, W. (2020). Prior-enlightened and motion-robust video deblurring. CoRR abs/2003.11209

Zhou, Z., Li, X., Zhang, T., Wang, H., & He, Z. (2022). Object tracking via spatial-temporal memory network. *IEEE Transactions on Circuits and Systems for Video Technology, 32*(5), 2976–2989.

Zhu, C., Dong, H., Pan, J., Liang, B., Huang, Y., Fu, L., & Wang, F. (2021). Deep recurrent neural network with multi-scale bi-directional propagation for video deblurring. CoRR abs/2112.05150, https://arxiv.org/abs/2112.05150,

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