Efficient Video Deblurring Guided by Motion Magnitude

Yusheng Wang[1], Yunfan Lu[2], Ye Gao[4], Lin Wang[2,3], Zhihang Zhong[1], Yinqiang Zheng[1], and Atsushi Yamashita[1]

[1] The University of Tokyo
`{wang,yamashita}@robot.t.u-tokyo.ac.jp`
`zhong@is.s.u-tokyo.ac.jp`, `yqzheng@ai.u-tokyo.ac.jp`
[2] AI Thrust, Information Hub, HKUST Guangzhou
[3] Department of Computer Science and Engineering, HKUST
`{yunfanlu,linwang}@ust.hk`
[4] Tokyo Research Center, Huawei
`gaoye1984@yahoo.co.jp`

Abstract. Video deblurring is a highly under-constrained problem due to the spatially and temporally varying blur. An intuitive approach for video deblurring includes two steps: a) detecting the blurry region in the current frame; b) utilizing the information from clear regions in adjacent frames for current frame deblurring. To realize this process, our idea is to detect the pixel-wise blur level of each frame and combine it with video deblurring. To this end, we propose a novel framework that utilizes the motion magnitude prior (MMP) as guidance for efficient deep video deblurring. Specifically, as the pixel movement along its trajectory during the exposure time is positively correlated to the level of motion blur, we first use the average magnitude of optical flow from the high-frequency sharp frames to generate the synthetic blurry frames and their corresponding pixel-wise motion magnitude maps. We then build a dataset including the blurry frame and MMP pairs. The MMP is then learned by a compact CNN by regression. The MMP consists of both spatial and temporal blur level information, which can be further integrated into an efficient recurrent neural network (RNN) for video deblurring. We conduct intensive experiments to validate the effectiveness of the proposed methods on the public datasets. Our codes are available at `https://github.com/sollynoay/MMP-RNN`.

Keywords: Blur estimation, motion magnitude, video deblurring

1 Introduction

Video deblurring is a classical yet challenging problem that aims to restore consecutive frames from the spatially and temporally varying blur. The problem is highly ill-posed because of the presence of camera shakes, object motions, and depth variations during the exposure interval. Recent methods using deep convolutional networks (CNNs) have shown a significant improvement in video

deblurring performance. Among them, alignment-based methods usually align the adjacent frames explicitly to the current frame and reconstruct a sharp frame simultaneously or by multiple stages [28,35,44,19,15]. Such methods show stable performance for video deblurring; however, the alignment or warping processes require accurate flow estimation, which is difficult for blurry frames, and usually the computational cost is relatively high. Recurrent neural network (RNN)-based methods achieve deblurring by passing information between adjacent frames [13,18,43]; such methods usually have lower computation costs but with lower restoration quality compared with the alignment-based methods.

So far, many deblurring methods have utilized priors to improve the restoration quality. The priors usually come from optical flows or image segmentation. Inter-frame optical flow-based methods [19,15] directly borrow pixel information from the adjacent frames to improve the quality of the current frame. By contrast, intra-frame optical flow estimates the pixel movement during the exposure time of a blurry frame. Intra-frame optical flow-based methods [8,1] estimate the flow during the exposure time, and usually restore the sharp frame by energy minimization optimization. Image segmentation-based methods [4,2,37,22,25] utilize motion, semantic or background and foreground information to estimate the prior for deblurring.

However, previous priors for video deblurring suffer from one or more of the following problems. First, image segmentation and inter-frame optical flow estimation on blurry images are error-prone. And it is necessary to estimate the prior and implement video deblurring simultaneously via complex energy functions [21,12,22]. Second, inter-frame optical flow estimation requires heavy computational cost. For instance, the state-of-the-art (SoTA) methods, PWC-Net requires 181.68 GAMCs [30] and RAFT small model (20 iterations) requires 727.99 GMACs [34] on the 720P (1280 x 720) frames. However, a typical efficient video deblurring method is supposed to have 50∼300 GMACs. Third, motion blur is directly correlated to the intra-frame optical flow. Although it is possible to directly estimate the intra-frame optical flow from a single image [8,1], the restored results suffer from artifacts.

An intuitive way for video deblurring is to detect the blurry region in the image first and then utilize information of clear pixels from adjacent frames. The detection of blurry region task can be considered as a blur estimation problem, which separates the image into binarized sharp and blurry regions, or directly tells the blur level of each pixel. Previous blur estimation methods, *e.g.*, [5,26], usually binarize the image into sharp and blurry regions and manually label the regions as it is difficult to determine the blur level of each pixel automatically. However, the labelling process may be inaccurate and usually requires much human effort.

In this work, we propose a motion magnitude prior (MMP) to represent the blur level of a pixel which can determine the pixel-wise blur level without manually labelling. Recent works use high frequency sharp frames to generate a synthetic blurry frame [17]. Inspired by this, the pixel movement during the exposure time of a blurry frame, i.e., the MMP, can be estimated from the
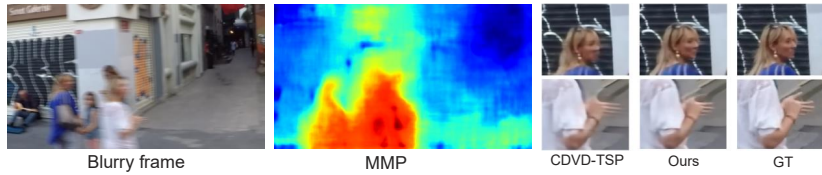
| Blurry frame | MMP | CDVD-TSP | Ours | GT |

Fig. 1. The blurry frame and the estimated MMP with results compared to the SoTA method [19].

average magnitude of bi-directional optical flow from the high-frequency sharp frames. The value of MMP positively correlated to the level of motion blur. We propose a compact CNN to learn the MMP.

The proposed MMP can directly indicate the *spatial distribution* of blur level in one image. Besides, if the overall value of MMP is low, the image is sharp, and vice versa. *Temporal information* is also included in the prior. The CNN can be further merged into a *spatio-temporal network* (RNN) for video deblurring. For convenience, we use an efficient RNN with residual dense blocks (RDBs) [41,42] as our backbone video deblurring network (Sec. 3.2). For the utilization of MMP, we design three components: a) for the intra-frame utilization, we propose a motion magnitude attentive module (MMAM) to inject the MMP into the network; b) for the inter-frame utilization, different from other RNN-based methods [13,18,43] which only pass the deblurred features to the next frame, we also pass the features before deblurring to the next frame. The blurry frame with pixels of different blur level is weighted by the MMAM, as such, pixels under low blur level can be directly utilized by the next frame; c) for loss-level utilization, since the motion magnitude of network output can reflect the deblur performance. That is, the sharper the image is, the lower the average score is. Therefore, we can also use the prior as a loss term to further improve the optimization. Figure 1 is an example of the learned MMP from the network and the estimation result. High quality results can be generated from the proposed framework.

In summary, our contributions are three folds. (I) We propose a novel motion magnitude prior for blurry image and its lightweight generation method. To the best of our knowledge, this is the first work to apply motion magnitude prior for the video deblurring task. (II) We propose a motion magnitude prior guided network, which utilizes the prior at intra-frame level, inter-frame level and loss level. (III) Our proposed method achieves competitive results with SoTA methods while with relatively lower computational cost.

2   Related Works

2.1   Prior for Deblurring

As mentioned above, image segmentation, inter-frame optical flow and intra-frame optical flow have been frequently used as priors in the deblurring tasks.

Image Segmentation In early studies, segmentation priors have been proposed for dynamic scene deblur. Cho *et al*. [4] segmented the images into multiple regions of homogeneous motions while Bar *et al*. [2] segmented the images into foreground and background layers. Based on [2] , Wulff *et al*. [37] focused on estimating the parameters for both foreground and background motions. Using segmentation priors can handle the dynamic scenes; however, with simple models, it is difficult to segment the non-parametrically varying complex motions. Ren *et al*. [22] exploited semantic segmentation which significantly improves the optical flow estimation for the deblurring tasks. Shen *et al*. [25] proposed a human-aware deblurring method by combining human segmentation and learning-based deblurring. Although these methods on the basis of physical models show promising results, the deblurring performance is highly related to the blur kernel estimation results. That is, inaccurate estimation of blur kernels results in severe artifacts.

Inter-frame optical flow The inter-frame optical flows are usually directly estimated on blurry frames, and are used to warp the adjacent frames to the center frame before inputting to the network [28,19,15]. The flow from blurry frame is inaccurate. To solve the problem, Pan *et al*. [19] proposed a method to estimate optical flow from the intermediate latent frames using a deep CNN and restore the latent frames based on the optical flow estimations. A temporal sharpness prior is applied to constrain the CNN model to help the latent frame restoration. However, calculating inter-frame optical flow requires heavy computation and the temporal sharpness prior cannot deal with all frames with no sharp pixels.

Intra-frame optical flow With deep learning, it is even possible to directly estimate intra-frame optical flows from a blurry frame [31,8,1], followed by energy function optimizations to estimate the blur kernels. However, the optimizations are usually difficult to solve and require a huge computational cost. The restored images also suffer from artifacts. Moreover, the definition of intra-frame is ambiguous as, during the exposure time, the movement of one pixel may be non-linear, which cannot be represented by a 2D vector.

Others Statistical priors and extreme channel priors have shown their feasibility in single image deblurring [24,38,20,39]. Such methods are valid under certain circumstances, but are sensitive to noise and still require accurate estimation of the blur kernels. Differently, we propose a motion magnitude prior learned by a compact network. The prior shows pixel-wise blur level of the blurry image and can be easily applied to the video deblurring framework. It can detect which part of the image is blurry and to what extent it is blurred.

2.2   Blur Estimation

Blur estimation has been studied for the non-uniform blur. It is similar to image segmentation prior by specifically segmenting the image into blur and non-blur regions. In [5], horizontal motion or defocus blurs were estimated and the image is deconvolved for deblurring. Shi *et al*. [26] studied the effective local blur features to differentiate between blur and sharp regions which can be applied to the blur

region segmentation, deblurring and blur magnification. The images are usually separated into blur and non-sharp regions; however, it is difficult to separate an image under a binarized way for dynamic scenes. Instead of classifying the pixels into blurry and sharp regions, we use a continuous way to represent the blur maps by estimating the motion magnitude of each pixel.

2.3 DNN-based Deblurring

For single image deblurring, SoTA methods apply self-recurrent module on the multi-scale, multi-patch, or multi-stage to solve the problem [17,33,7,29,40]. Despite the high performance, they usually require a large computational cost. For video deblurring, the problem is less ill-posed and can be solved with less computational cost. The learning-based video deblurring methods can be grouped into alignment-based methods [28,35,44,19,15] and RNN-based methods [13,18,43]. The former usually explicitly aligns the adjacent frames to the center frame for deblurring. This can be achieved by directly warping the adjacent frames to the center frame [28], warping the intermediate latent frames, or both [19,15]. Wang *et al*. [35] implemented the deformable CNN [6] to realize the alignment process. Zhou *et al*. [44] proposed a spatio-temporal filter adaptive network (STFAN) for alignment and deblurring in a single network. Son et al. blur-invariant motion estimation learning to avoid the inaccuracy caused by optical flow estimation on blurry frames [27]. Although the alignment-based methods can achieve relatively high performance, the alignment process is usually computationally inefficient. In addition, the alignment has to be accurate, otherwise it may degrade the performance of deblurring. To ensure high quality results, multi-stage strategies were applied by iteratively implementing the alignment-deblurring process, which may lead to huge computational cost [19,15].

On the other hand, RNN-based methods transfer information between the RNN cells and usually have lower computational cost. Kim *et al*. [13] propose an RNN for video deblurring by dynamically blending features from previous frames. Nah *et al*. [17] iteratively updated the hidden state with one RNN cell before the final output. Zhong *et al*. [43] proposed an RNN network using RDB backbone with globally fusion of high-level features from adjacent frames and applied attention mechanism to efficiently utilize inter-frame information. In this paper, we propose an RNN with RDBs by utilizing the MMP. We use a compact CNN to estimate the MMP to reduce the computational cost. The MMP consists of spatial and temporal information, which can improve the effectiveness of information delivery between RNN cells. Our proposed method outperforms the SoTA performance.

3 The Proposed Approach

3.1 Motion Magnitude Prior

Motion Magnitude Prior from Optical Flow In this section, we introduce preparing the ground truth of motion magnitude prior (MMP). For the learning-based deblurring methods, the datasets for training and validation are usually
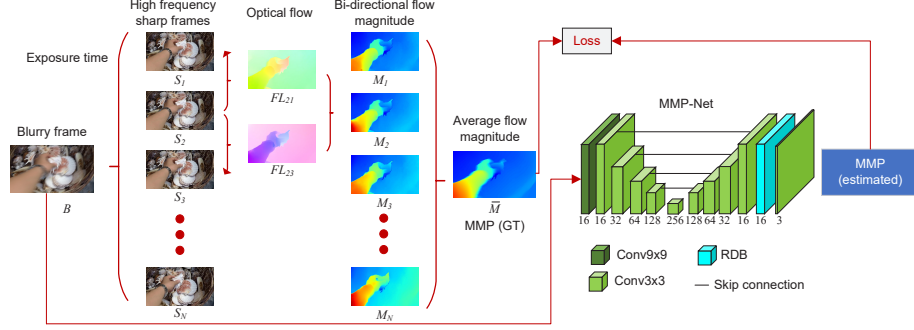
Fig. 2. Motion magnitude prior. High frequency sharp frames are used to generate synthetic blurry frames. Meanwhile, we estimate the bi-directional optical flows for each frame and calculate the average magnitude of bi-directional optical flows. Then, we take the average of the motion magnitude maps of all the latent frames. Finally, we use a modified UNet-like structure to learn the motion magnitude prior by regression.

synthesized by high-frequency sharp frames [28,17,35]. It is based on the fact that images tend to be blurry after accumulating multiple sharp images with slight misalignment caused by motion [9]. The blur level of each pixel in the blurry frame is positively correlated to the motion magnitude of the pixel during exposure time. Although directly measuring the motion magnitude of one pixel is difficult, it is possible to be calculated by latent sharp frames during exposure time. Inspired by this, we generate blurry frame and blur level map pairs based on high-frequency sharp image sequence, as shown in Fig. 2. Denoting the blurry image as B and sampled sharp image as $\{S_1, S_2, \ldots, S_N\}$, the blurry image simulation process can be represented as follows.

$$B = c\left(\frac{1}{N}\sum_{i=1}^{N}S_i\right),\tag{1}$$

where $c(.)$ refers to the camera response function (CRF) [32]. During exposure time, we sample N sharp images to generate a blurry image. To measure the movement of pixels, we calculate the optical flow between the sharp frames. We use bi-directional flows to represent the pixel movement of one sharp frame. For instance, as shown in Fig. 2, for frame 1, we calculate the optical flow FL_{21} and FL_{23} to represent the pixel movement condition. For each pixel (m, n), the optical flow between frame i and j in x and y direction can be represented as $u_{i,j}(m, n)$ and $v_{i,j}(m, n)$, respectively. We denote the motion magnitude M for frame i as follows.

$$M_i(m, n) = \frac{\sqrt{u_{i,i-1}^2(m, n) + v_{i,i-1}^2(m, n)} + \sqrt{u_{i,i+1}^2(m, n) + v_{i,i+1}^2(m, n)}}{2}.\tag{2}$$

For frame 1 and frame N, we only use FL_{12} and $FL_{N,N-1}$ for calculation, respectively. Then, to acquire the pixel-wise motion magnitude for the synthetic blurry frame, we calculate the average movement during exposure time.

$$\bar{M} = \frac{1}{KN} \sum_{i=1}^{N} M_i, \tag{3}$$

where $\bar{M}$ refers to the motion magnitude or blur level map for the blurry frame. K is used to normalize the value at each position in MMP to 0~1. K is determined by the maximum value of the MMP before normalization in the training dataset which was set to 15.

The Learning of Motion Magnitude Prior In this paper, we use the GO-PRO raw dataset [17] to generate blurry image and MMP pairs. The GOPRO benchmark dataset used 7~13 successive sharp frames in raw dataset to generate one blurry frame. We also use 7~13 to generate the blurry image and MMP pairs. We use the SoTA optical flow method RAFT [34] to estimate the optical flow. We propose a compact network to learn the blur level map by regression. We apply a modified UNet [23], as shown in Fig. 2. At the beginning, we use a 9 × 9 convolution layer to enlarge the reception field. The features are downsampled and reconstructed with a UNet-like structure. Then, a residual dense block (RDB) is used to refine the result. To process a 720P (1280 × 720) frame, the computational cost is only 38.81 GMACs with a model size of 0.85M parameters. At last, we can estimate the MMP for each frame using the compact network. In this paper, we apply the MMP as guidance to the video deblurring network. We will describe it in the following section.

3.2 Motion Magnitude Prior-Guided Deblurring Network

In this section, we describe using MMP as guidance to video deblurring network. **Network Structure** The structure of MMP-RNN is shown in Fig. 3. n_c refers to the channel dimension. The center frame B_t is first inputted into MMP-Net to estimate the MMP M_t. Then, B_t and M_t are passed to RNN cell to extract features. Non-deblurred features l_{t-1} and deblurred features h_{t-1} from the previous frame are delivered to feature extraction module (FEM). In this paper, we also use the deblurred features from adjacent frames globally to reconstruct output images. Features f_{t-2}, f_{t-1} from the past frames and f_{t+1}, f_{t+2} from the future frames with the current frame features f_t are inputted to reconstruction module (RM) for image reconstruction. We denote the output result as O_t. RDBs have high performances in low-level tasks which efficiently preserve features and save the computational cost [41,42,43]. In this work, we use RDBs as the backbone for downsampling, feature extraction and implicit alignment.

The FEM receives center frame B_t and corresponding MMP M_t with features l_{t-1} and h_{t-1} from the past frames to extract features of current frame. The structure of FEM is shown in Fig. 3. The Motion Magnitude Attentive Module (MMAM) is used to fuse the information of B_t and M_t. Here, we pass the non-deblurred features l_t to the next FEM and receive l_{t-1} from the previous FEM.
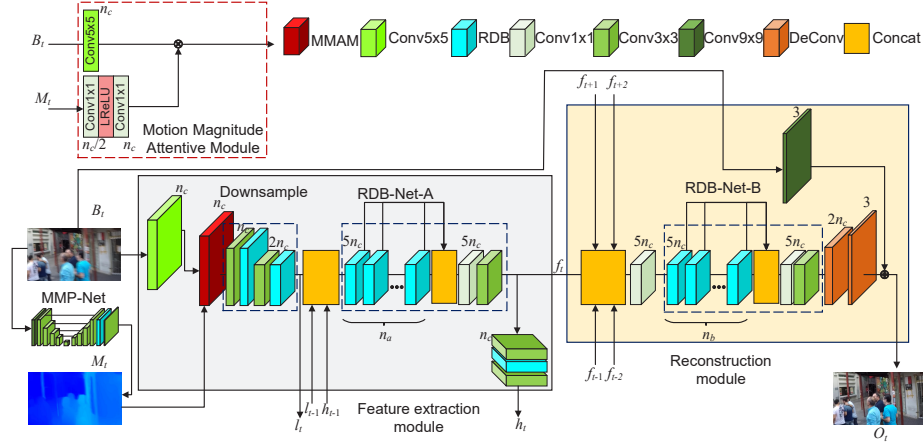
Fig. 3. The structure of MMP-RNN. Both center frame I_t and corresponding blur prior B_t estimated by MMP-Net are passed into RNN cell. We transmit both non-deblurred feature l and deblurred feature h from the previous frame to the next frame. Deblurred features f from the current frame and the adjacent frames are fused globally in decoder to generate final output image.

The non-deblurred features and deblurred features are then concatenated and passed through RDB-Net-A with n_a RDBs. At last, deblurred features h_t are passed to the next FEM and f_t are used for image reconstruction.

RM is designed to globally aggregate high-level features for image reconstruction. We concatenate the features $\{f_{t-2}, \ldots, f_{t+2}\}$ and squeeze them in channel direction using 1×1 convolution operation. We use n_b RDBs in RDB-Net-B to implicitly align the features from different frames and then apply convolution transpose operation to reconstruct the image. A global skip connection is added to directly pass I_t to the output after 9×9 convolution operation. This can better preserve the information of the center frame.

Motion Magnitude Attentive Module Both the blurry and sharp pixels are important to our task. The blurry pixels are the region to be concentrated for deblurring and the sharp pixels can be utilized for deblurring of adjacent frames. Especially for sharp pixels, the value of MMP is close to 0, directly multiplying MMP to image features may lose information of sharp features. To better utilize MMP, we use MMAM to integrate it with image features. The structure of MMAM layer is shown in Fig. 3. The MMP is passed through two 1 $\times$ 1 convolution layers to optimize MMP and adjust the dimension. The MMP is transformed to tensor γ . The operation to integrate γ and the feature from B, x can be represented as $x^{out} = \gamma \otimes x^{in}$, where $\otimes$ refers to element-wise multiplication. The whole operation can better integrate the MMP information with the blurry image.

Feature Transmission We transmit two kinds of features l and h between FEM of the adjacent frames. The non-deblurred feature l_t consists of the information only from the center frame before integration. By contrast, h_t possesses features from the previous frame. The features are integrated in the network as follows.

$$a_t = CAT(l_t, l_{t-1}, h_{t-1}), \tag{4}$$

where CAT refers to concatenation operation. The RNN-based methods implicitly fusing information from previous frames which may sometimes cause lower performance compared to alignment-based methods. Passing non-deblurred features only consisting of the current frame with its blur level information can improve the overall performance of the network.

Motion Magnitude Loss The proposed MMP-Net can estimate pixel-wise motion magnitude of the image. If O_t is an ideal sharp image, inputting O_t into MMP-Net should get a prior with all zeros. In this work, we propose a loss function based on the idea as follows.

$$\mathcal{L}_{MM} = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} M_{i,j}(O_t), \tag{5}$$

where $M(O_t)$ refers to the MMP of the output image O_t. Theoretically, by minimizing the average motion magnitude of O_t, it can generate ideal sharp image.

We also consider two kinds of content loss functions for training. We use a modified Charbonnier loss ($\mathcal{L}_{char}$) [3] and gradient loss ($\mathcal{L}_{grad}$) as content loss.

$$\mathcal{L}_{char} = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \sqrt{\sum_{ch}^{r,g,b} (I_{i,j,ch} - O_{i,j,ch}) + \epsilon^2}, \tag{6}$$

where ϵ is a small value which is set to 0.001. m and n are the width and height of the image. I refers to the ground truth and O is the estimated image.

$$\mathcal{L}_{grad} = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (G_{i,j} - \hat{G}_{i,j})^2, \tag{7}$$

where G and $\hat{G}$ refers to the image gradient of I and O, respectively. Then, the total loss is written as follows.

$$\mathcal{L} = \mathcal{L}_{char} + \lambda_1 \mathcal{L}_{grad} + \lambda_2 \mathcal{L}_{MM}, \tag{8}$$

where the weight λ_1 and λ_2 are set to 0.5 and 1.0 in our experiments.

4   Experiment

4.1   MMP-Net

Dataset Generation To learn the deep image prior, we utilize GOPRO dataset [17]. The raw GOPRO dataset consists of 33 high-frequency video sequences with
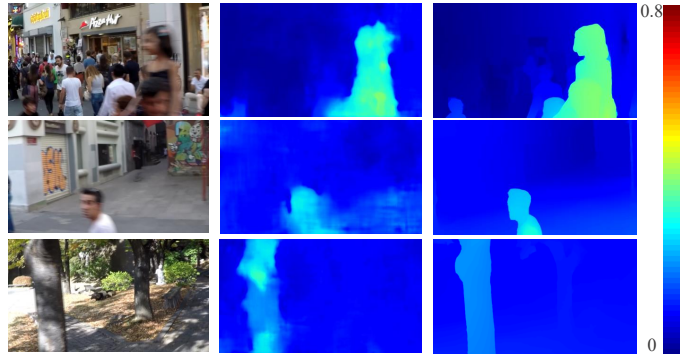
Fig. 4. Examples of motion magnitude estimation on GOPRO test dataset. Each column refers to the input blurry frames, the estimated results, and GT, respectively.

Table 1. The training result of MMP-Net.

	Train	Test (512 patch)	Test (Full image)
$\mathcal{L}_1$	0.0137	0.0169	0.0192

34,874 images in total. In GOPRO benchmark dataset, the sharp images are used to generate synthetic dataset with 22 sequences for training and 11 sequences for evaluation with 2,103 training samples and 1,111 evaluation samples. We use the same data separation as GOPRO benchmark dataset to build the MMP dataset to avoid information leakage during video deblurring. We use 7~11 consecutive sharp frames to generate one blurry frame and corresponding MMP. We generate 22,499 training samples (22 sequences) and use the original GOPRO test dataset. We trim the training dataset by 50% during training.

Implementation details We train the model for 400 epochs with a mini-batch of size 8 using ADAM optimizer [14] with initial learning rate 0.0003. The learning rate decades by half after 200 epochs. The patch size is set to 512×512 for training and validation. The loss $\mathcal{L}_1 = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} |M_{i,j} - \hat{M}_{i,j}|$ is used for MMP training and as a metric for test. Here, $M_{i,j}$ refers to the value of estimated MMP at position i, j and $\hat{M}$ refers to the GT.

Results The training results are listed in Table 1. We visualize our results on GOPRO test dataset in Fig. 4. To prove the generality of the proposed method, we also test the GOPRO trained model to HIDE dataset [25]. HIDE dataset contains blurry images of human. As shown in Fig. 5, our proposed method can also detect the blur caused by human motion on other dataset.

4.2   Video Deblurring

Datasets We test the proposed video deblurring method on two public datasets, GORPRO benchmark dataset [17] and beam-splitter datasets (BSD) [43]. BSD
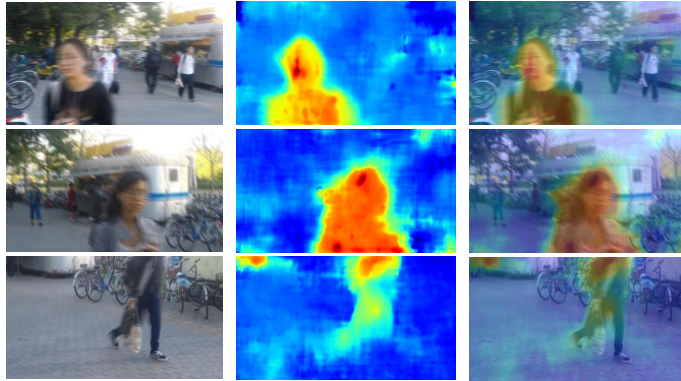
Fig. 5. Examples of test results on HIDE dataset [25]. The model was trained on GOPRO dataset and tested on HIDE dataset. Each column refer to the input blurry images, estimated results, and the estimated results overlaid to the input images. It indicates that the proposed method can successfully detect salient blurry region on other datasets.

is a dataset of images from real world by controlling the length of exposure time and strength of exposure intensity during video shooting using beam-splitter system [11]. It can better evaluate deblurring performance in real scenarios. We use the 2ms-16mes BSD that the exposure time for sharp and blurry frames are 2 ms and 16 ms, respectively. The training and validation sets have 60 and 20 video sequences with 100 frames respectively, and the test set has 20 video sequences with 150 frames. The size of the frames is 640×480.

Implementation details We train the model using ADAM optimizer with a learning rate of 0.0005. We adopt cosine annealing schedule [16] to adjust learning rate during training. We sample 10-frame 256×256 RGB patch sequences from the dataset to construct a mini-batch of size 8 with random vertical and horizontal flips as well as $90°$ rotation for data augmentation for training. We train 1,000 epochs for GOPRO and 500 epochs for BSD, respectively. It is worth mentioning that we try to train the other methods using publicly available codes by ourselves and keep the same hyper-parameters if possible. For CDVD-TSP, we use the available test images for GOPRO and keep the training strategy used in [19] for BSD.

Benchmark Results The results of our method with SoTA lightweight image [33] and video deblurring methods on GOPRO is shown in Table 2. We use $A_\# B_\# C_\# F_\#$ to represent n_a, n_b, n_c and the length of the input image sequence in our model. We evaluate the image quality in terms of PSNR [10] and SSIM. We also measure the computational cost for each model when processing one 720P frame in terms of GMACs. Running time (s) of one 720P frame is also listed. For IFIRNN, $c2$ refers to the dual cell model and $h3$ refers to 'three times' of hidden state iteration. Our $A_7 B_8 C_{16} F_8$ model outperforms the other methods with only 204.19 GMACs. The visual comparison of the results are shown in Fig. 6.

Table 2. Quantitative results on GOPRO.

Model	PSNR	SSIM	GMACs	Param	Time (s)
SRN [33]	29.94	0.8953	1527.01	10.25	0.173
DBN [28]	28.55	0.8595	784.75	15.31	0.128
IFIRNN ($c2h3$) [18]	29.69	0.8867	217.89	1.64	0.034
ESTRNN ($C_{70}B_7$) [43]	29.93	0.8903	115.19	1.17	0.021
ESTRNN ($C_{90}B_{10}$) [42]	31.02	0.9109	215.26	2.38	0.035
CDVD-TSP [19]	31.67	0.9279	5122.25	16.19	0.729
MMP-RNN ($A_3B_4C_{16}F_5$)	30.48	0.9021	136.42	1.97	0.032
MMP-RNN ($A_7B_8C_{16}F_8$)	31.71	0.9225	204.19	3.05	0.045
MMP-RNN ($A_9B_{10}C_{18}F_8$)	**32.64**	**0.9359**	264.52	4.05	0.059

Table 3. Quantitative results on BSD 2ms-16ms.

Model	PSNR	SSIM	GMACs
DBN [28]	31.33	0.9132	784.75
IFIRNN ($c2h3$) [18]	31.59	0.9209	217.89
ESTRNN ($C_{90}B_{10}$) [43]	31.80	0.9245	215.26
CDVD-TSP [19]	32.06	0.9268	5122.25
MMP-RNN ($A_8B_9C_{18}F_8$)	32.79	0.9365	247.41
MMP-RNN ($A_9B_{10}C_{18}F_8$)	**32.81**	**0.9369**	264.52

The results on BSD of the proposed method with other SoTA methods are shown in Table 3. Here, we use the $A_8B_9C_{18}F_8$ model. The visualization results of the video deblurring on BSD are shown in Fig. 7. Our method outperforms SoTA methods with less computational cost.

4.3 Ablation Study

Network Structure We conduct ablation tests on the proposed method with $A_9B_{10}C_{18}$ model on GOPRO and $A_8B_9C_{18}$ model on BSD. We focus on three parts, the MMAM, motion magnitude loss and the transmission of non-deblurred features as shown in Table 4. On GOPRO, the MMAM with MMP can improve PSNR by 0.31 dB. Together with motion magnitude loss, the prior can improve the score by 0.39 dB. If we remove all the components, the PSNR may drop by 0.58, which can indicate the effectiveness of the proposed method. As for BSD, the PSNR significantly increased by 0.63 dB after fusing prior using MMAM. The motion magnitude loss can improve PSNR by 0.09 dB.

Influence of Prior We also did ablation tests with different types of priors using $A_3B_4C_{80}F_5$ model on GOPRO. As shown in Table 5, we tried the ground truth MMP, the ground truth of flow magnitude of the center frame (from FL_{c1} and FL_{cN}, where c refers to the center frame), the normalized ground truth map ($M/\max(M)$), without MMP and the MMP from MMP-Net. We noticed that the ground truth as an upper boundary can increase the score by 0.37 dB. The
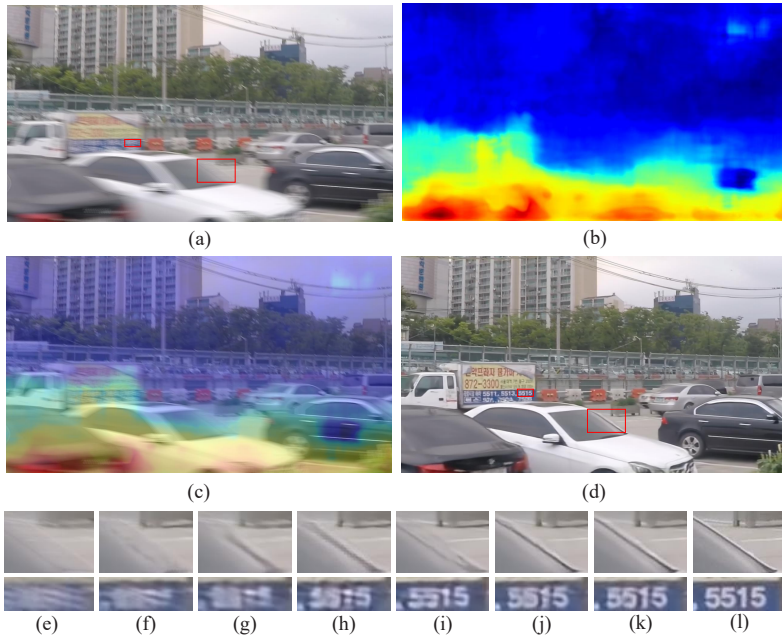
Fig. 6. Visualization results of GoPRO. (a) Blurry frame. (b) the estimated MMP. (c) Overlaying (b) on (a). (d) Result from MMP-RNN. From (e) to (l) are, blurry input, DBN, IFIRNN, ESTRNN, proposed method w/o MMP, MMP-RNN, and the sharp frame, respectively.

Table 4. Ablation tests. NDF refers the transmission of non-deblurred features.

Model			GOPRO		BSD	
MMAM	$\mathcal{L}_{MM}$	NDF	PSNR	GMACs	PSNR	GMACs
✓	✓	✓	**32.64**	264.52	**32.79**	247.41
✗	✓	✓	32.33	225.34	32.16	208.23
✗	✗	✓	32.25	225.34	32.07	208.23
✗	✗	✗	32.06	227.21	32.03	210.09

Table 5. Influence of different types of prior. I. Ground truth, II.Ground truth of flow magnitude of center frame, III. Normalizing each ground truth map to 0∼1, IV. Replacing the MMP and MMAM by spatial-self-attention [36]. V. None, VI. Estimated from MMP-Net.

Prior type	I	II	III	IV	V	VI
PSNR	**30.54**	30.47	30.41	30.37	30.17	30.48

MMP from MMP-Net can increase the PSNR by 0.31 dB. On the other hand, using the flow magnitude of the center frame will decrease the PSNR by 0.07 dB.
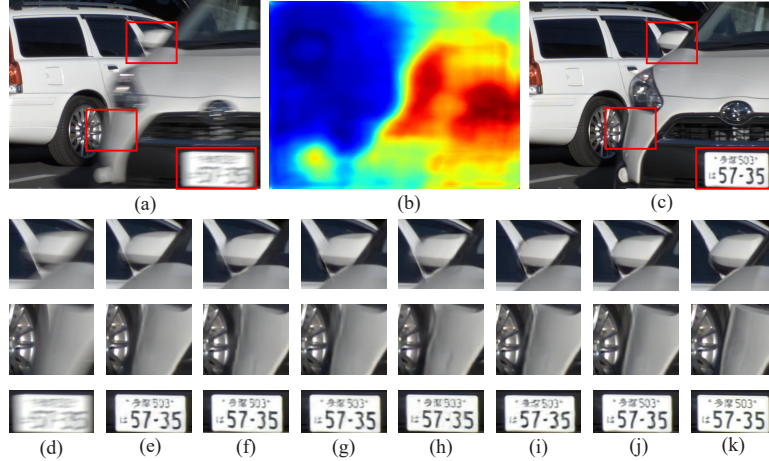
Fig. 7. Visualization results of BSD. a) Blurry frame. (b) the estimated MMP. (c) Result from MMP-RNN. From (d) to (k) are, blurry input, DBN, IFIRNN, ESTRNN, proposed method w/o MMP, MMP-RNN, and the sharp frame, respectively.

The contour of the center frame magnitude only corresponds to the center frame, the attentive field cannot cover the whole blurry object. Normalizing the MMP by the maximum value of each MMP may lose temporal information, especially for some sharp images, the whole MMP value may be close to 1, which may influence the performance. We also compare the proposed method to spatial-self-attention [36]. Our proposed method use a supervised approach to tell the network where to concentrate and the results outperform vanilla spatial-self-attention.

5 Conclusion

In this paper, we proposed a motion magnitude prior for deblurring tasks. We built a dataset of blurry image and motion magnitude prior pairs and used a compact network to learn by regression. We applied the prior to video deblurring task, combining the prior with an efficient RNN. The prior is fused by motion magnitude attentive module and motion magnitude loss. We also transmitted the features before deblurring with features after deblurring between RNN cells to improve efficiency. We tested the proposed method on GOPRO and BSD, which achieved better performance on video deblurring tasks compared to SoTA methods on image quality and computational cost.

References

1. Argaw, D.M., Kim, J., Rameau, F., Cho, J.W., Kweon, I.S.: Optical flow estimation from a single motion-blurred image. In: AAAI (2021) 2, 4
2. Bar, L., Berkels, B., Rumpf, M., Sapiro, G.: A variational framework for simultaneous motion estimation and restoration of motion-blurred video. In: ICCV. pp. 1–8 (2007) 2, 4
3. Charbonnier, P., Blanc-Feraud, L., Aubert, G., Barlaud, M.: Two deterministic half-quadratic regularization algorithms for computed imaging. In: ICIP. pp. 168–172 (1994) 9
4. Cho, S., Matsushita, Y., Lee, S.: Removing non-uniform motion blur from images. In: ICCV. pp. 1–8 (2007) 2, 4
5. Couzinié-Devy, F., Sun, J., Alahari, K., Ponce, J.: Learning to estimate and remove non-uniform image blur. In: CVPR. pp. 1075–1082 (2013) 2, 4
6. Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y.: Deformable convolutional networks. In: ICCV. pp. 764–773 (2017) 5
7. Gao, H., Tao, X., Shen, X., Jia, J.: Dynamic scene deblurring with parameter selective sharing and nested skip connections. In: CVPR. pp. 3848–3856 (2019) 5
8. Gong, D., Yang, J., Liu, L., Zhang, Y., Reid, I., Shen, C., Hengel, A.V.D., Shi, Q.: From motion blur to motion flow: A deep learning solution for removing heterogeneous motion blur. In: CVPR. pp. 3806–3815 (2017) 2, 4
9. Hirsch, M., Schuler, C.J., Harmeling, S., Schölkopf, B.: Fast removal of non-uniform camera shake. In: ICCV. pp. 463–470 (2011) 6
10. Horé, A., Ziou, D.: Image quality metrics: Psnr vs. ssim. In: ICPR. pp. 2366–2369 (2010) 11
11. Jiang, H., Zheng, Y.: Learning to see moving objects in the dark. In: ICCV. pp. 7324–7333 (2019) 11
12. Kim, T.H., Lee, K.M.: Generalized video deblurring for dynamic scenes. In: CVPR. pp. 5426–5434 (2015) 2
13. Kim, T.H., Lee, K.M., Scholkopf, B., Hirsch, M.: Online video deblurring via dynamic temporal blending network. In: ICCV. pp. 4038–4047 (2017) 2, 3, 5
14. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015) 10
15. Li, D., Xu, C., Zhang, K., Yu, X., Zhong, Y., Ren, W., Suominen, H., Li, H.: Arvo: Learning all-range volumetric correspondence for video deblurring. In: CVPR. pp. 7721–7731 (June 2021) 2, 4, 5
16. Loshchilov, I., Hutter, F.: Sgdr: Stochastic gradient descent with warm restarts. In: ICLR (2017) 11
17. Nah, S., Kim, T.H., Lee, K.M.: Deep multi-scale convolutional neural network for dynamic scene deblurring. In: CVPR. pp. 3883–3891 (2017) 2, 5, 6, 7, 9, 10
18. Nah, S., Son, S., Lee, K.M.: Recurrent neural networks with intra-frame iterations for video deblurring. In: CVPR. pp. 8102–8111 (2019) 2, 3, 5, 12
19. Pan, J., Bai, H., Tang, J.: Cascaded deep video deblurring using temporal sharpness prior. In: CVPR. pp. 3043–3051 (2020) 2, 3, 4, 5, 11, 12
20. Pan, J., Sun, D., Pfister, H., Yang, M.: Blind image deblurring using dark channel prior. In: CVPR. pp. 1628–1636 (2016) 4
21. Portz, T., Zhang, L., Jiang, H.: Optical flow in the presence of spatially-varying motion blur. In: CVPR. pp. 1752–1759 (2012) 2
22. Ren, W., Pan, J., Cao, X., Yang, M.H.: Video deblurring via semantic segmentation and pixel-wise non-linear kernel. In: ICCV. pp. 1077–1085 (2017) 2, 4

23. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: MICCAI. pp. 234–241 (2015) 7
24. Shan, Q., Jia, J., Agarwala, A.: High-quality motion deblurring from a single image. ACM Trans. Graph. **27**(3), 1–10 (2008) 4
25. Shen, Z., Wang, W., Shen, J., Ling, H., Xu, T., Shao, L.: Human-aware motion deblurring. In: ICCV (2019) 2, 4, 10, 11
26. Shi, J., Xu, L., Jia, J.: Discriminative blur detection features. In: CVPR. pp. 2965–2972 (2014) 2, 4
27. Son, H., Lee, J., Lee, J., Cho, S., Lee, S.: Recurrent video deblurring with blur-invariant motion estimation and pixel volumes. ACM Trans. Graph. (2021) 5
28. Su, S., Delbracio, M., Wang, J., Sapiro, G., Heidrich, W., Wang, O.: Deep video deblurring for hand-held cameras. In: CVPR. pp. 237–246 (2017) 2, 4, 5, 6, 12
29. Suin, M., Purohit, K., Rajagopalan, A.N.: Spatially-attentive patch-hierarchical network for adaptive motion deblurring. In: CVPR. pp. 3606–3615 (2020) 5
30. Sun, D., Yang, X., Liu, M.Y., Kautz, J.: PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In: CVPR (2018) 2
31. Sun, J., Cao, W., Xu, Z., Ponce, J.: Learning a convolutional neural network for non-uniform motion blur removal. In: CVPR. pp. 769–777 (2015) 4
32. Tai, Y.W., Chen, X., Kim, S., Kim, S.J., Li, F., Yang, J., Yu, J., Matsushita, Y., Brown, M.S.: Nonlinear camera response functions and image deblurring: Theoretical analysis and practice. IEEE Transactions on Pattern Analysis and Machine Intelligence **35**(10), 2498–2512 (2013) 6
33. Tao, X., Gao, H., Shen, X., Wang, J., Jia, J.: Scale-recurrent network for deep image deblurring. In: CVPR. pp. 8174–8182 (2018) 5, 11, 12
34. Teed, Z., Deng, J.: Raft: Recurrent all pairs field transforms for optical flow. In: ECCV. pp. 402–419 (2020) 2, 7
35. Wang, X., Chan, K.C., Yu, K., Dong, C., Loy, C.C.: Edvr: video restoration with enhanced deformable convolutional networks. In: CVPRW (2019) 2, 5, 6
36. Woo, S., Park, J., Lee, J.Y., Kweon, I.S.: Cbam: Convolutional block attention module. In: ECCV (2018) 13, 14
37. Wulff, J., Black, M.J.: Modeling blurred video with layers. In: ECCV. pp. 236–252 (2014) 2, 4
38. Xu, L., Zheng, S., Jia, J.: Unnatural l0 sparse representation for natural image deblurring. In: CVPR. pp. 1107–1114 (2013) 4
39. Yan, Y., Ren, W., Guo, Y., Wang, R., Cao, X.: Image deblurring via extreme channels prior. In: CVPR. pp. 6978–6986 (2017) 4
40. Zamir, S.W., Arora, A., Khan, S., Hayat, M., Khan, F.S., Yang, M.H., Shao, L.: Multi-stage progressive image restoration. In: CVPR (2021) 5
41. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: CVPR. pp. 2472–2481 (2018) 3, 7
42. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image restoration. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020) 3, 7, 12
43. Zhong, Z., Gao, Y., Zheng, Y., Zheng, B.: Efficient spatio-temporal recurrent neural network for video deblurring. In: ECCV. pp. 191–207 (2020) 2, 3, 5, 7, 10, 12
44. Zhou, S., Zhang, J.and Pan, J., Xie, H., Zuo, W., Ren, J.: Spatio-temporal filter adaptive network for video deblurring. In: ICCV. pp. 2482–2491 (2019) 2, 5