EVDM: Event-based Real-world Video Deblurring with Mamba

Supplementary Material

A. More details of our camera system

A.1. Hardware Design

We used two industrial cameras (MV-CS016-10UC) and one event camera (Prophesee EVK4) to construct our camera system, aiming to simultaneously capture blurred and sharp frames. Additionally, we employed two hardware settings to record paired blurred and sharp frames with different exposure time ratios: 1) for an exposure time ratio of 3:1, we used two 50:50 beam splitters; and 2) for an exposure time ratio of 9:1, we utilized a 50:50 beam splitter and a 10:90 beam splitter. Furthermore, we equipped the event camera with a sleeve to ensure that the optical path lengths to all three cameras were consistent.

A.2. Geometric Alignment

Our hardware optical path design ensures that the images captured by the three cameras are spatially aligned. However, due to hardware precision limitations and potential misalignment during the capture process, further geometric alignment of the recorded content is necessary. Following the previously described method [1, 2, 5], we utilize a homography matrix to achieve alignment among the three cameras.

First, for the two RGB cameras, before starting motion video capture in each scene, we capture several still frames. These still frames are then used for feature point matching and homography matrix computation, allowing us to obtain the correction parameters for the homography matrix for each scene.

For the event camera and RGB camera, during the still frame recording step, we introduce slight jitter to the event camera to capture the corresponding edge information of the scene. We then compute feature point matching and the homography matrix using the processed event frames and the still frames.

A.3. Photometric Alignment

We ensured that the light intensity entering the three cameras is consistent through careful hardware design. The specific details are as follows: In the setup with an exposure time ratio of 3:1, the exposure time of the RGB camera recording the blurred frames is three times that of the RGB camera recording the sharp frames. To achieve this, we placed an ND filter with an optical density (OD) of 0.5 in front of the blurred camera. With this configuration, the amount of light accumulated during the exposure process for both the blurred and sharp cameras is nearly equal. Additionally, to maintain consistency in optical path length, we installed an optical sleeve of the same length as the ND filter in front of the sharp camera. Furthermore, to ensure that the light intensity received by the event camera matches that of the blurred camera, we placed an ND filter with an OD of 0.8 in front of the event camera. In the setup with an exposure time ratio of 9:1, a 10:90 beam splitter ensures that the blurred camera captures the same amount of light as the sharp camera during the ninefold exposure duration. The event camera is equipped with an ND filter of 1 OD value to ensure that it receives the same incident light intensity as the blurred camera. Both the blurred and clear cameras are also fitted with optical sleeve of the same length as the ND filters to guarantee consistent optical path lengths.

A.4. Postprocessing

Despite our efforts to ensure consistent light accumulation during exposure through hardware constraints, such as using identical models of cameras and lenses and designing the optical paths, and synchronizing the white balance of the two RGB cameras through the capture software, we observed some color discrepancies in the frames captured by the two RGB cameras. To further correct this difference, we employed a linear color correction model as [4] described. Specifically, we use the following linear formula to align the colors of the sharp frame S and blurred frame B:

$$\alpha S + \beta \approx B, \tag{1}$$

where α and β are estimated using the following method:

$$\begin{aligned} \alpha &= \frac{\sigma_1}{\sigma_2}, \\ \beta &= \mu_1 - \alpha \mu_2. \end{aligned} \tag{2}$$

In this equation, σ and μ represent the standard deviation and mean of the frame, respectively. Due to the blurring of the image affecting the color distribution, it is challenging to obtain suitable mean and variance values directly from the blurred image. Therefore, we compute these parameters using static frames captured at the beginning of each scene. We applied this alignment process to each color channel of the images individually.

B. More details of implementations

B.1. Inplementation details

We train the EVDM without pre-training on 4 NVIDIA 3090 GPUs. For all datasets, We randomly crop patches of size 128 $\times$ 128 for training frames and corresponding event voxels, and the batch size is set to 8 by default. The

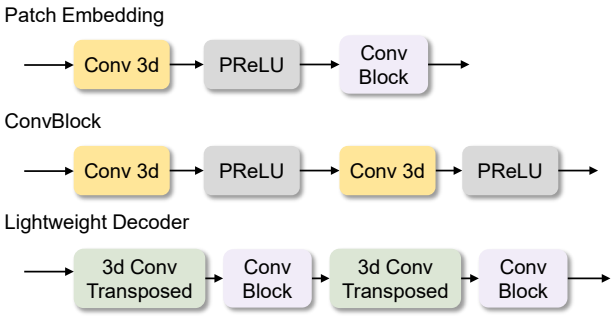


Figure 1. The details of three modules applied in EVDM.

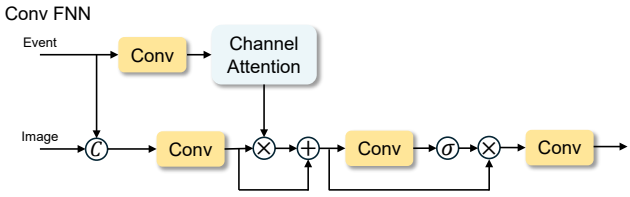


Figure 2. The details of ConvFFN modules applied in EVDM.

AdamW optimizer is employed ($\beta_1 = 0.9$ and $\beta_2 = 0.99$) with the initial learning rate 7×10^{-5} with the cosine annealing schedule where the T_{max} is 400K iteration. Our method can accept video inputs of any length. To balance performance and training efficiency, we use three consecutive video frames as input, which means T is 3. Following the configuration in [2], the number of event voxel bins is set to 16 for the GoPro and EVRB datasets, while it is set to 10 for the T-RED dataset.

B.2. Datasets

GoPro: The GoPro dataset [3] is commonly used in the motion deblurring field. For a fair comparison, we utilize the raw event data provided in EFNet, which is generated by the ESIM simulator with random thresholds. According to the official division, 22 video sequences are used as the training set, while 11 video sequences are used for testing.
EVRB: The EVRB dataset [2] contains 17 videos with a resolution of 960×640, each of which is over 100 frames in length and contains various motion modes. According to the standard division, we use 11 sequences as the train set and 6 sequences as the test set.
T-RED: The T-RED dataset consists of 15 videos for training and 9 videos for testing. Each video includes simultaneously captured blurred frames, sharp frames, and corresponding event data, all at a resolution of 1024×768. Both the training and testing sets include sequences with exposure time ratios of 1:3 and 1:9, capturing various common motion processes in the real world.

C. More details of our EVDM.

In this section, we detail some modules in our EVDM framework. Figure 1 illustrates the architecture of three modules for EVDM:
Patch Embedding. This module processes inputs through a `3D convolution (Conv 3d)` layer followed by a parametric rectified linear unit (`PReLU`).
ConvBlock. The ConvBlock consists of stacked `Conv 3d + PReLU` layers.
Lightweight Decoder. For pixel-wise reconstruction, this module employs a `3D transposed convolution (ConvT)` layer. An optional refining `Conv block` ensures sharpness restoration while maintaining computational efficiency.

Figure 2 presents a channel-attention enhanced convolution feedforward network (ConvFFN):
ConvFFN. A sequential stack of `Conv` layers processes input features, followed by a `Channel Attention` module. This module dynamically weighs feature channels through `Conv + Softmax` operations, emphasizing high-frequency information critical for sharp image reconstruction. The progressive convolution design effectively captures both global and local motion blur characteristics.

References

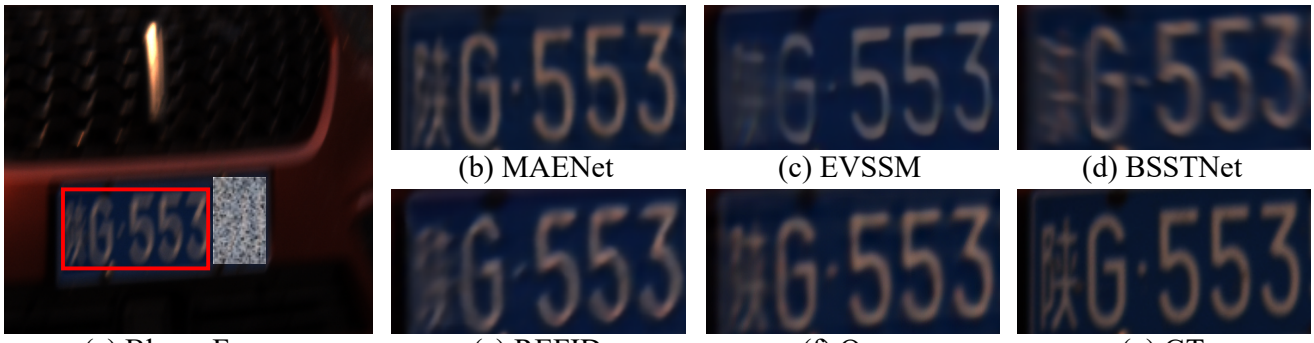
[1] Jianrui Cai, Hui Zeng, Hongwei Yong, Zisheng Cao, and Lei Zhang. Toward real-world single image super-resolution: A new benchmark and a new model, 2019. 1

[2] Taewoo Kim, Hoonhee Cho, and Kuk-Jin Yoon. Cmta: Cross-modal temporal alignment for event-guided video deblurring. In *European Conference on Computer Vision*, pages 1–19. Springer, 2024. 1, 2

[3] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *CVPR*, 2017. 2

[4] Jaesung Rim, Haeyun Lee, Jucheol Won, and Sunghyun Cho. Real-world blur dataset for learning and benchmarking deblurring algorithms. In *Computer vision–ECCV 2020: 16th European conference, glasgow, UK, August 23–28, 2020, proceedings, part XXV 16*, pages 184–201. Springer, 2020. 1

[5] Jaesung Rim, Geonung Kim, Jungeon Kim, Junyong Lee, Seungyong Lee, and Sunghyun Cho. Realistic blur synthesis for learning image deblurring, 2022. 1



(a) Blurry Frame (b) MAENet (c) EVSSM (d) BSSTNet (e) REFID (f) Ours (g) GT

Figure 3. Qualitative comparisons on the T-RED dataset. Zoom in for better view.

Figure 4. Data sample of T-RED.