Appendix of Bilateral Event Mining and Complementary for Event Stream Super-Resolution

A. Dataset and Training Configuration

Synthetic Data Generation Method: We synthesized the NFS-syn and RGB-syn datasets using an event simulator [6]. Specifically, we initially downsampled the NFS dataset [4] and RGB-DAVIS dataset [8] using bicubic interpolation to obtain low-resolution images. The original resolution of the NFS dataset is 1280×720, and we applied $2(4, 8, 16)\times$ downsampling, while the RGB-DAVIS dataset has an original resolution of 1520×1440, and we downsampled it by $2(4, 8)\times$. Subsequently, we generated event streams through the event simulator, utilizing default initial parameters. In the NFS-syn dataset, we considered the $16\times$ downsampled 80×45 data as the minimum resolution and the $2\times$ downsampled 360×180 data as the maximum resolution, resulting in LR-HR pairs at $2(4, 8)\times$. For the RGB-syn dataset, we used the $8\times$ downsampled 190×180 data as the minimum resolution and the $2\times$ downsampled 760×720 data as the maximum resolution, resulting in LR-HR pairs at $2(4)\times$. And we applied data augmentations such as random flipping and polarity inversion to the dataset, and randomly split it for training and testing.

Training Configuration: During training, we applied several enhancements to the event stream data, including vertical and horizontal flipping of event count images and flipping event stream polarities, each with a probability of 50%. We randomly partitioned the data into training and testing sets for EventNFS, NFS-syn, and RGB-syn. Training and testing were performed on each dataset.

B. Pseudo Code of BIE

We provide the pseudo code of BIE, as presented in Algorithm 1.

Algorithm 1 Pseudo code of BIE

```
# xp, xn: features of positive, negative event, respectively
# xint: the cross-layer interaction representation

def BIE(xp, xn, xint):
    b, c, h, w = xp.shape
    scale_factor = c ** 0.5

    # (1) update xp and xn
    xp_ = convp(xp)
    xn_ = convn(xn)

    # (2) obtain the query Q and the value V
    Qp = conv_qp(LayerNorm(Conv2D(torch.cat([xint, xp],dim=1)))).view(b,c,-1)
    Qn = conv_qn(LayerNorm(Conv2D(torch.cat([xint, xn],dim=1)))).view(b,c,-1)
    Vp = conv_vp(xp).view(b,c,-1)
    Vn = conv_vn(xn).view(b,c,-1)

    Kp = Vp.permute(0,2,1)
    Kn = Vn.permute(0,2,1)

    # (3) calculate attention scores
    A_n2p = torch.bmm(Qp, Kn) * scale_factor
    A_p2n = torch.bmm(Qn, Kp) * scale_factor

    # (4) information propagation
    x_n2p = torch.bmm(torch.softmax(A_n2p,dim=-1), Vn).view(b,c,h,w)
    x_p2n = torch.bmm(torch.softmax(A_p2n,dim=-1), Vp).view(b,c,h,w)

    # (5) gated information fusion
    wp = torch.sigmoid(conv_wp1(xp_) + conv_wp2(x_n2p))
    xp = wp*xp_ + (1-wp)*x_n2p
    wn = torch.sigmoid(conv_wn1(xn_) + conv_wn2(x_p2n))
    xn = wn*xn_ + (1-wn)*x_p2n

    # (6) update xint
    xint = conv_int(torch.cat([Qp.view(b,c,h,w), Qn.view(b,c,h,w)], dim=1)) + xint

    return xp, xn, xint
```

C. More Ablation Study

C.1. Effect of CIR

To verify the effectiveness of CIR in our proposed BIE, we simply remove the CIR in BIE to make a qualitative comparison with BIE (for the quantitative comparisons, please refer to Tab. 3 in the **main paper**.). Moreover, we present the previous LR frames which are denoted as

$LR - i, i = 1, 2$. As shown in Figure 1, the introduction of CIR in the BIE enables our model to more effectively capture contextual information from different events in previous frames and utilize information that does not exist in the current frame, thereby enhancing the performance.

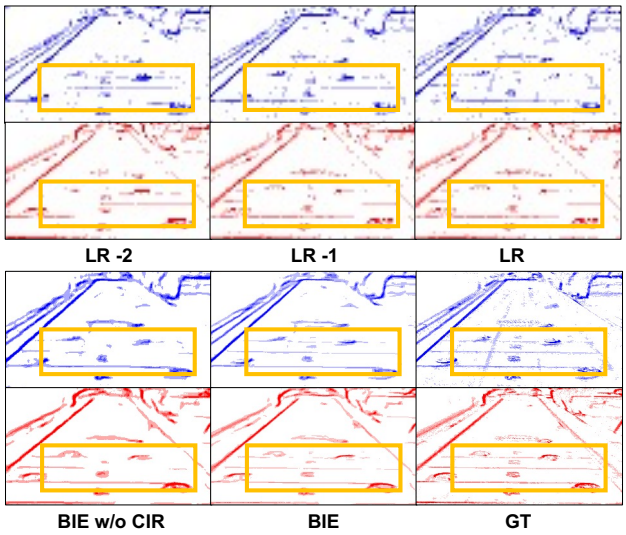

Figure 1. Qualitative comparison between BIE equipped with and without CIR.

C.2. Hyperparameters of Architecture

We conducted experiments to explore the impact of hyperparameters on the super-resolution results of the BMCNET, specifically focusing on two aspects: 1) ablation of layers in the residual network module, and 2) ablation of the channel size in feature extraction. To investigate the influence of model hyperparameters on the super-resolution outcomes, we modified the model parameters and retrained them on the NFS-syn dataset. The $4\times$ SR results on both the NFS-syn dataset are presented in Tab. 1. Additionally, we report their model parameters and FLOPs for model and time complexity analysis. The results show that adjusting the number of layers has a minor impact on the performance of BMCNET, but it significantly reduces FLOPs. On the other hand, reducing the channel size substantially decreases the parameter count while significantly affecting the performance. After considering the trade-off between efficiency and effectiveness, we have set C=128 and N=5 in practice.

Effectiveness of The Number of Global Structures in BIE. To investigate the effect of the number of global structures M in BIE, we conduct a series experiments by setting different M. The channel C=128, resolution (H, W) of LR events is $(80, 45)$, and the scale S=4. The results are present in Tab. 2. In practice, we set M=128 from the consideration of the trade-off between efficiency and performance.

(C, N)	NFS-syn	# Params	# FLOPs
(32, 5)	0.598	**0.2** M	**2.2** G
(64, 5)	0.571	0.7 M	8.9 G
(128, 5)	0.552	2.7 M	35.7 G
(256, 5)	0.553	10.6 M	141.7 G
(128, 3)	0.554	2.5 M	22.9 G
(128, 10)	**0.547**	3.2 M	66.3 G

Table 1. The effect of channel C and the number of basic blocks N in BMCNET. The FLOPs is calculated on the LR events of NFS-syn dataset with resolutions of 80×45.

Methods	NFS-syn	# Params	# FLOPs
M=8	0.581	**335.6** K	**1.19** G
M=16	0.580	342.8 K	1.21 G
M=32	0.578	357.2 K	1.24 G
M=64	0.579	385.9 K	1.30 G
M=128	0.577	443.5 K	1.42 G
M=256	0.576	558.5 K	1.50 G
M=512	**0.574**	778.7 K	2.13 G

Table 2. The effect of the number of global structures in BIE. The experiments are conducted on NFS-syn dataset for $4\times$ SR on RMSE metrics

C.3. The Generalization in Real Event Dataset

To validate the generalization of our models, we applied the models trained on the synthetic dataset NFS-syn to perform super-resolution on the real dataset EventNFS [1]. From Tab. 3, it can be observed that although our methods BMCNET-plain and BMCNET still outperform other approaches, this advantage is somewhat diminished. There could be two reasons for this. First, the minimum resolution of EventNFS is 55×31, which is too small, causing significant degradation of event information and hindering the effective recovery of high-resolution event streams. Second, differences between real and synthetic event streams may result in a weakened performance of the model. Investigating methods to mitigate the impact of differences between real and synthetic event streams is an important direction for our future research.

D. More Visual Results

D.1. More Qualitative Comparison Results

In Figure 2, Figure 3, and Figure 4, we present the $4\times$ super-resolution results on the NFS-syn, RGB-syn, and EventNFS-real datasets for bicubic, SRFBN [5], RLSP [2], RSTT [3], EventZoom [1], RecEvSR [9], and our methods BMCNET-plain and BMCNET. The low-resolution (LR) resolutions for NFS-syn, RGB-syn, and EventNFS-real are

Methods	EventNFS-real*	
	2×	4×
bicubic	0.872	0.948
SRFBN [5]	0.837	0.901
RLSP [2]	0.837	0.917
RSTT [3]	0.812	0.887
EventZoom [1]	1.043	1.117
RecEvSR [9]	0.822	0.897
BMCNet-plain	<u>0.804</u>	**0.869**
BMCNet	**0.792**	<u>0.877</u>

Table 3. Quantitative analysis results for super-resolution on EventNFS-real dataset using bicubic, SRFBN [5], RLSP [2], RSTT [3], EventZoom [1], RecEvSR [9], and our methods BMCNET-plain and BMCNET. * denotes that all models are trained on the synthetic dataset NFS-syn and then applied to super-resolution on the real dataset EventNFS-real. We report RMSE results for each super-resolution scale.

80×45, 190×180, and 55×31, respectively. It is evident from the results that our BMCNET-plain and BMCNET excel in integrating overall structural information to complement and rectify detailed information, resulting in richer details and clearer edges.

D.2. Event-Based Video Reconstruction

As shown in Figure 5, we applied bicubic, SRFBN [5], RLSP [2], RSTT [3], EventZoom [1], RecEvSR [9], and our methods BMCNET-plain and BMCNET to perform $4\times$ super-resolution on the downsampled NFS-syn dataset. Subsequently, we utilized the E2VID [7] to reconstruct the events based on the super-resolved event stream. It is evident that our methods BMCNET-plain and BMCNET achieve superior reconstruction with finer details and fewer artifacts, validating the effectiveness of our approach.

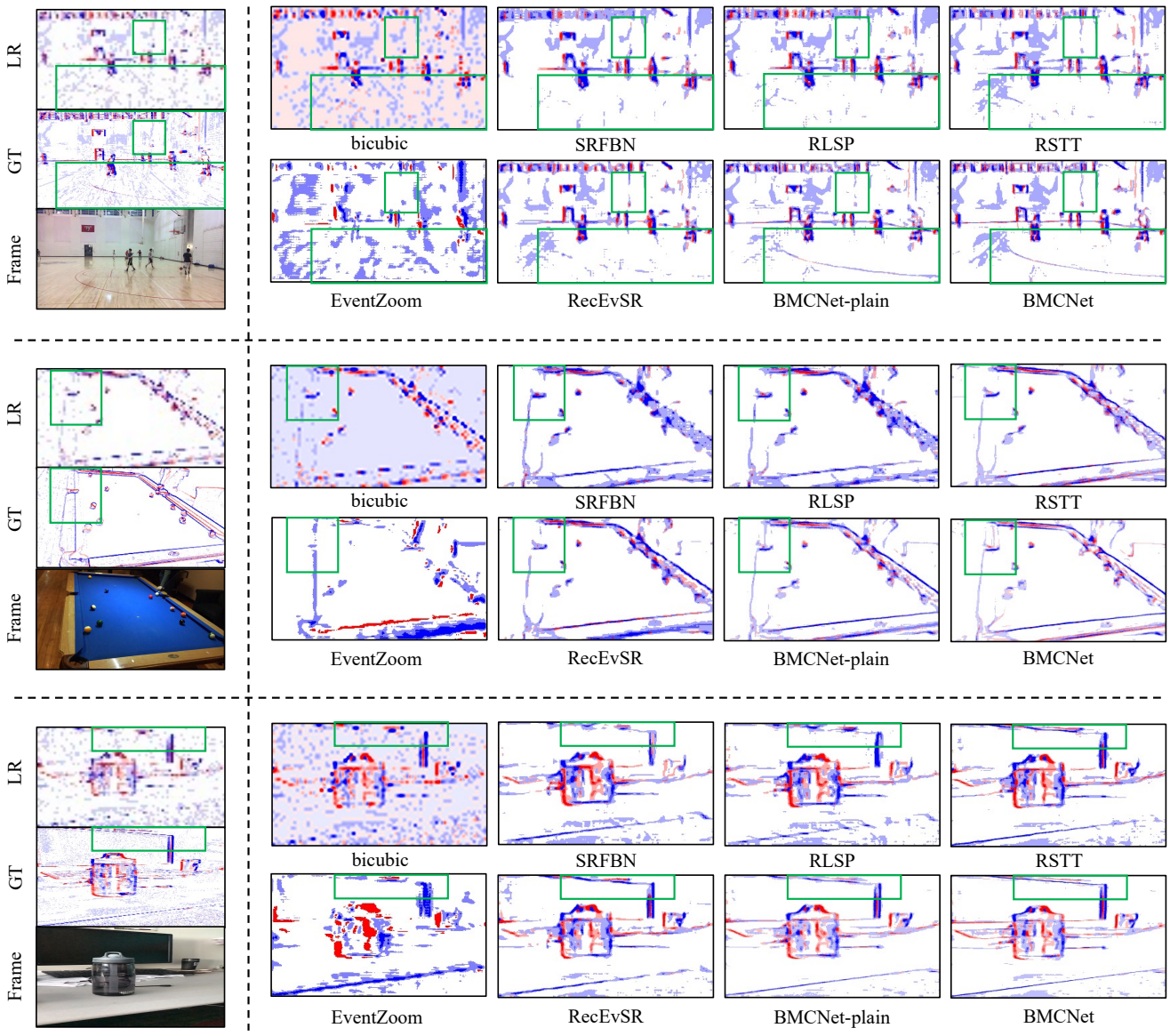

Figure 2. Super-Resolution Results at $4\times$ Scale on the NFS-syn Dataset for bicubic, SRFBN [5], RLSP [2], RSTT [3], EventZoom [1], RecEvSR [9], and our methods BMCNET-plain and BMCNET. **[Best viewed with zoom-in.]**

Figure 3. Super-Resolution Results at 4× Scale on the RGB-syn Dataset for bicubic, SRFBN [5], RLSP [2], RSTT [3], EventZoom [1], RecEvSR [9], and our methods BMCNET-plain and BMCNET. **[Best viewed with zoom-in.]**

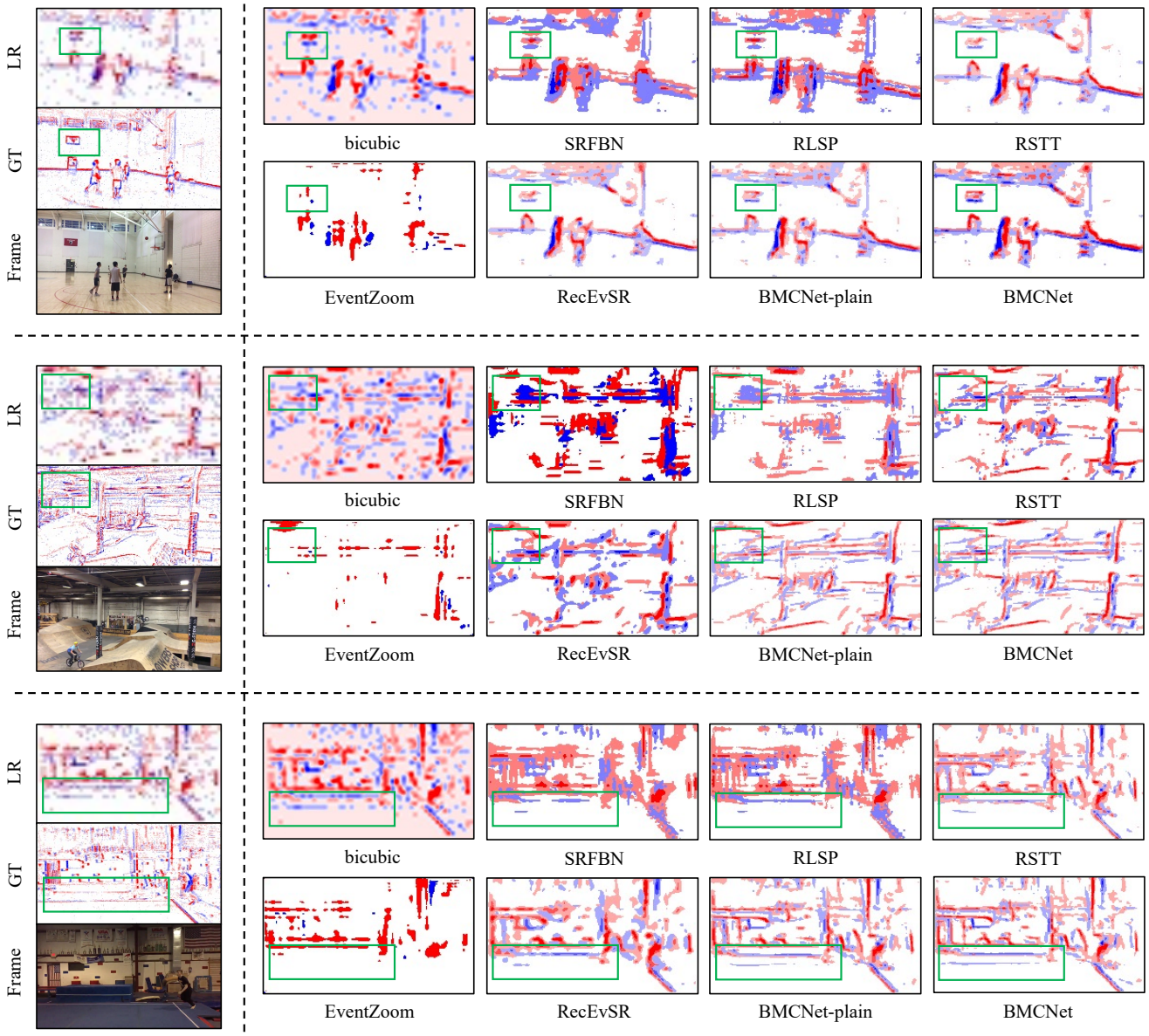

Figure 4. Super-Resolution Results at 4× Scale on the EventNFS-real Dataset for bicubic, SRFBN [5], RLSP [2], RSTT [3], EventZoom [1], RecEvSR [9], and our methods BMCNET-plain and BMCNET. **[Best viewed with zoom-in.]**

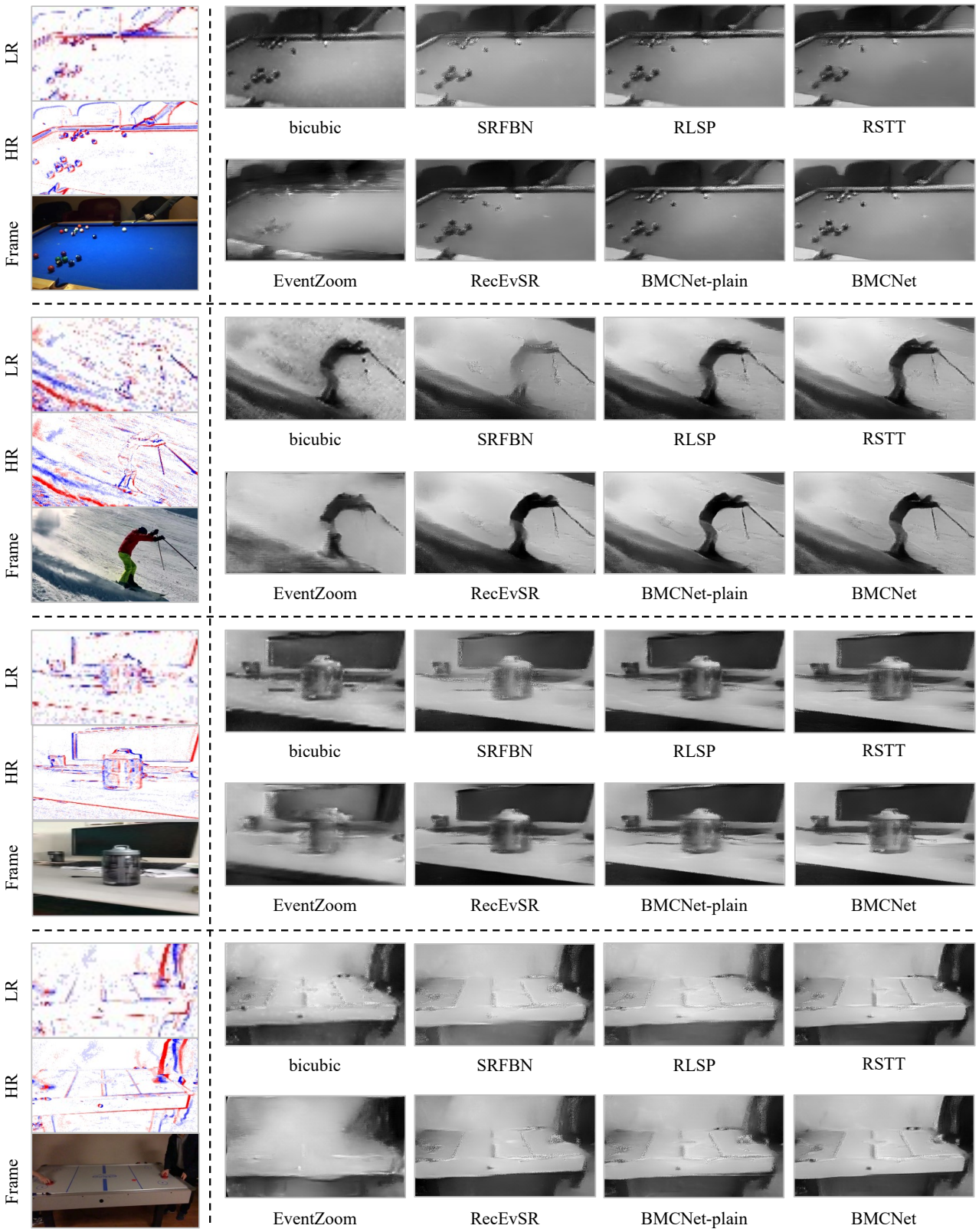

Figure 5. Qualitative analysis results for event-based video reconstruction comparing bicubic, SRFBN [5], RLSP [2], RSTT [3], EventZoom [1], RecEvSR [9], and our methods BMCNET-plain and BMCNET. The results showcase the performance of all methods in 4× super-resolution on the NFS-syn dataset. **[Best viewed with zoom-in.]**

References

[1] Peiqi Duan, Zihao W Wang, Xinyu Zhou, Yi Ma, and Boxin Shi. Eventzoom: Learning to denoise and super resolve neuromorphic events. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12824–12833, 2021. 2, 3, 4, 5, 6, 7

[2] Dario Fuoli, Shuhang Gu, and Radu Timofte. Efficient video super-resolution through recurrent latent space propagation. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 3476–3485. IEEE, 2019. 2, 3, 4, 5, 6, 7

[3] Zhicheng Geng, Luming Liang, Tianyu Ding, and Ilya Zharkov. Rstt: Real-time spatial temporal transformer for space-time video super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17441–17451, 2022. 2, 3, 4, 5, 6, 7

[4] Hamed Kiani Galoogahi, Ashton Fagg, Chen Huang, Deva Ramanan, and Simon Lucey. Need for speed: A benchmark for higher frame rate object tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1125–1134, 2017. 1

[5] Zhen Li, Jinglei Yang, Zheng Liu, Xiaomin Yang, Gwanggil Jeon, and Wei Wu. Feedback network for image super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3867–3876, 2019. 2, 3, 4, 5, 6, 7

[6] Songnan Lin, Ye Ma, Zhenhua Guo, and Bihan Wen. Dvs-voltmeter: Stochastic process-based event simulator for dynamic vision sensors. In *European Conference on Computer Vision*, pages 578–593. Springer, 2022. 1

[7] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. High speed and high dynamic range video with an event camera. *IEEE transactions on pattern analysis and machine intelligence*, 43(6):1964–1980, 2019. 3

[8] Zihao W Wang, Peiqi Duan, Oliver Cossairt, Aggelos Katsaggelos, Tiejun Huang, and Boxin Shi. Joint filtering of intensity images and neuromorphic events for high-resolution noise-robust imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1609–1619, 2020. 1

[9] Wenming Weng, Yueyi Zhang, and Zhiwei Xiong. Boosting event stream super-resolution with a recurrent neural network. In *European Conference on Computer Vision*, pages 470–488. Springer, 2022. 2, 3, 4, 5, 6, 7