Learning Adaptive Parameter Representation for Event-Based Video Reconstruction

Daxin Gu, Jia Li, *Senior Member, IEEE*, and Lin Zhu, *Member, IEEE*

***Abstract*—Event-based video reconstruction aims to generate images from asynchronous event streams, which record the intensity changes exceeding specific contrast thresholds. However, the contrast thresholds are varied among pixels with manufacturing imperfections and circumstancing interference, which causes undesirable events. It may cause the existing works to output blurry frames with unpleasing artifacts. To address this, we propose a novel two-stage framework to reconstruct images with learnable parameter representations. The learnable representation of the contrast threshold is extracted with a transformer network from corresponding asynchronous events in the first stage. Then a UNet architecture is utilized in the second stage to fuse the representations with the event encoding features to refine the decoding features in spatiotemporal dimensions. The representation learned from asynchronous events can adapt to the variety of contrast thresholds when processing event data in diverse scenes, motivating the proposed framework to generate high-quality frames. Quantitative and qualitative experimental results on the four public datasets show that our approach achieves better performance.**

***Index Terms*—Event-based vision, video reconstruction, transformer, contrast threshold.**

I. Introduction

EVENT cameras (ECs) represent a groundbreaking advancement in bio-inspired sensors [1], which capture visual signals by recording intensity changes over a defined contrast threshold (CT). In comparison to traditional frame-based cameras, ECs produce asynchronous event streams characterized by extremely low temporal latency and a high dynamic range [2]. To bridge the gap between traditional frame-based vision and the novel event streams, it is essential to reconstruct video from asynchronous event streams [3], [4].

Early reconstruction methods [12], [13], [14], [15] emulate the inherent imaging principles of ECs, directly integrating event intensity with multiple hand-crafted priors. While demonstrating the potential of video reconstruction under predetermined contrast thresholds, these approaches are limited by manufacturing imperfections and circumstantial interference (CT may neither temporally constant nor spatially homogeneous in real scenes [9]). Oversimplified contrast threshold estimation strategies [14] may introduce reconstruction errors, resulting in severe blurriness. Recent deep learning-based reconstruction methods, exemplified by E2VID [5], propose recurrent convolutional models to reconstruct videos without relying on predetermined CT assumptions. Building on this, [6], [16], [17] enhance the architecture with various maneuvers, improving the details of reconstructed frames. Moreover, there has been an exploration of novel network architectures for video reconstruction tasks, such as [8] employing vision transformers [18], [19] using spiking neural networks, [20] using Hypernetworks, and [21] using the CycleGAN architecture [22]. These approaches have demonstrated impressive performance, yet may struggle to generalize when a significant domain shift occurs between target events and training data (see Fig. 1(a)).

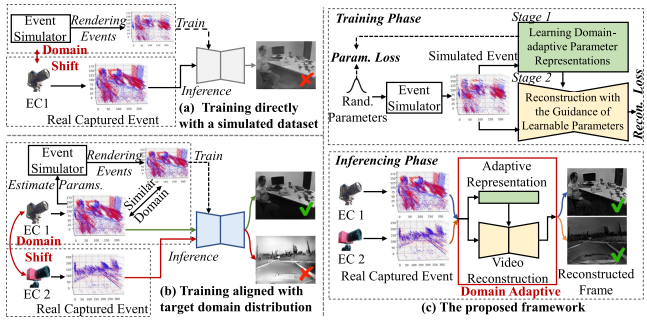

Fig. 1. The motivation of our framework. Compared to direct training with simulated events ((a) e.g., [5], [6], [7], and [8]) and aligning training data to a specific target domain ((b) e.g., [9], [10], and [11]), our framework leverages learnable representations to adapt to target domain events, enabling generalization across diverse event distributions (c).

To address this, recent efforts [9], [10], [11] focus on reducing the domain gap by simulating training events with attributes similar to the target domain. However, these methods typically involve estimating thresholds for a specific dataset, generating training data, and then training existing models. This process results in models that perform optimally only when confronted with data sharing a similar distribution of thresholds, constraining their overall generalization (see Fig. 1(b)). Moreover, utilizing the generated data to directly train existing networks fails to fully exploit the inherent connections between parameters and reconstruction models. Consequently, they perform well only within a specific target domain, lacking robust generalization capabilities.

In this letter, we introduce a two-stage framework aimed at learning a domain-adaptive parameter representation of CTs

Manuscript received 12 December 2023; revised 18 May 2024; accepted 19 July 2024. Date of publication 25 July 2024; date of current version 2 August 2024. This work was supported in part by the National Natural Science Foundation of China under Contract 62132002 and Contract 62302041, and in part by the China National Postdoctoral Program under Contract BX20230469. The associate editor coordinating the review of this article and approving it for publication was Prof. Saurabh Prasad. *(Corresponding authors: Lin Zhu; Jia Li.)*

Daxin Gu and Jia Li are with the State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing 100191, China (e-mail: daxingu@buaa.edu.cn; jiali@buaa.edu.cn).

Lin Zhu is with the School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China (e-mail: linzhu@bit.edu.cn).

This letter has supplementary downloadable material available at https://doi.org/10.1109/LSP.2024.3433403, provided by the authors.

Digital Object Identifier 10.1109/LSP.2024.3433403

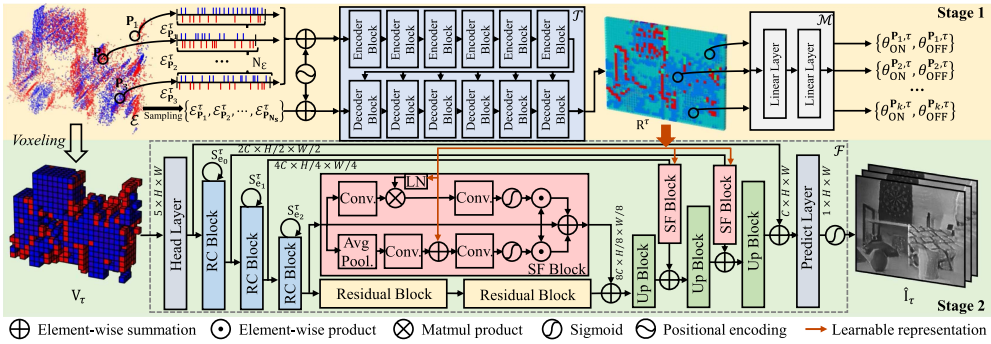

Fig. 2. The pipeline of our proposed network. Stage one encodes event streams into a learnable representation of contrast thresholds using a Transformer. In stage two, this representation is seamlessly integrated into a CNN network, enhancing event-based video reconstruction. .

and reconstructing high-quality videos with the guidance of parameter representation. Specifically, we utilize a Transformer-based [23] parameter estimation model to learn the representation of CTs based on diverse simulated event data, which exploits the asynchronous nature of event camera sufficiently. Additionally, we design a spatiotemporal feature fusion module, incorporating the adaptive parameter representation with the global texture representation extracted from the corresponding event voxel [24] in channel and spatial dimensions separately. Our approach combines the strengths of the domain-adaptive parameter representation and the global texture representation, enhancing the interpretability of the reconstruction model and adapting to various scenes.

Our main contributions are: 1) We propose a novel two-stage framework, which learns asynchronous adaptive representation for event-based video reconstruction. 2) We design a transformer-based parameter estimation model, which can provide adaptive representations response to the pixel-wise contrast thresholds. 3) We develop a spatiotemporal feature fusion module for integrating the adaptive parameter representation into the video reconstruction framework efficiency. 4) Our framework exhibits robust performance across various datasets compared to existing methods.

II. The Approach

A. Overview

The architecture of our proposed network is illustrated in Fig. 2, which consists of two stages:

Stage one: The event streams $\mathcal{E}$ are firstly processed into pixel-independent temporal-isometric pieces as the lossless input of the Transformer $\mathcal{T}$. After calculation, the learnable representations $\mathbf{R}^\tau$, including all pixels at time τ, are extracted and utilized to estimate the pixel-wise CT of target domain event cameras via an MLP structure.

Stage two: Instead of introducing the pixel-wise learnable representations to reconstruct videos directly, we sample part of event pieces $\{\mathcal{E}^\tau_{\mathbf{P}_1}, \mathcal{E}^\tau_{\mathbf{P}_2}, \ldots, \mathcal{E}^\tau_{\mathbf{P}_{\mathrm{N_s}}}\}$ at time τ at random position $\mathbf{P}$ for inference efficiency. $\mathrm{N_s}$ is the sampling number of event pieces. Then, $\mathbf{R}^\tau$ is fused to each feature in the feature pyramid extracted from event voxel grid $\mathbf{V}_\tau$ with the proposed Spatiotemporal Feature Fusion Block (SF Block). After that, the fused feature pyramid is fed into the network $\mathcal{F}$ to reconstruct videos with the guidance of learnable parameters.

B. Learning Domain-Adaptive Parameter Representations

To establish a texture-independent mapping $\mathbf{R}^\tau$ from changes in scene brightness to the parameters of ECs (θ_{ON} and θ_{OFF}), both LSTM and Transformer architectures can extract and aggregate valuable information from continuous event streams. Given that parameter estimation involves a pixel-wise task where there are correlations between different pixels, and these correlations may even be independent of pixel order, opting for a Transformer over an LSTM appears more reasonable. This is due to the attention of the Transformer, which allows for greater flexibility in handling such correlations.

1) Event Input: As the event streams $\mathcal{E}$ record that each event $\mathbf{e}$ occurs in position $\mathbf{P}$, at time $\tau > 0$ and event polarity $p = \pm 1$, we define a temporal-isometric piece of event streams $\mathcal{E}^\tau_{\mathbf{P}}$ as a temporal continuous sequence of events occurs in position $\mathbf{P}$, starts from time τ. In this way, the event stream can be expressed as $\mathcal{E}^\tau_{\mathbf{P}} = \{\mathbf{e}_i\}_{i=1}^{\mathrm{N}_\varepsilon} = \{\mathbf{P}_i, \tau_i, p_i\}_{i=1}^{\mathrm{N}_\varepsilon}$, where $\mathrm{N}_\mathcal{E}$ is the length of the event sequence. Then, we embed this event sequence to a one-dimensional vector by multiplying the time of each event τ with its corresponding polarity p and regularize this vector to the range from -1 to 1 as the input $\mathcal{V}^\tau_{\mathbf{P}} \in \mathbb{R}^{\mathrm{N}_\mathcal{E}}$ of Transformer $\mathcal{T}$. Follows [23], we further introduce sinusoidal positional encoding $\rho_{\mathbf{P}} \in \mathbb{R}^{\mathrm{N}_\mathcal{E}}$ to map each $\mathcal{V}^\tau_{\mathbf{P}}$ into a latent one-dimensional embedding token $\mathbf{z}^\tau_{\mathbf{P}} \in \mathbb{R}^{\mathrm{N}_\mathcal{E}}$, which is formulated as $\mathbf{z}^\tau_{\mathbf{P}} = \mathcal{V}^\tau_{\mathbf{P}} + \rho_{\mathbf{P}}$. After event embedding and position encoding, the token sequence $\{\mathbf{z}^\tau_{\mathbf{P}_i}\}_{i=1}^{\mathrm{N_s}}$ can be processed by Transformer $\mathcal{T}$ subsequently.

2) Network Architecture: Our Transformer $\mathcal{T}$ consists of 6 vanilla encoder blocks and decoder blocks, respectively. Same as [23], each encoder block is comprised of self-attention operations and feed-forward layer, which are adopted with residual connection with layer normalization. Then the decoders are appended to output the final latent representation $\mathbf{R}^\tau_{\mathbf{P}} \in \mathbb{R}^{\mathrm{N}_\mathcal{E}}$ depending on the key and value vectors generating from encoders and the input token $\mathbf{z}^\tau_{\mathbf{P}}$ of decoders. This design endows our Transformer with the capacity of learning to extract information from pixel-independent tokens and ensures the event tokens are processed in parallel among pixels during the training phase. Additionally, we introduce the Multi-Layer Perceptron (MLP, $\mathcal{M}$) architecture to estimate the probability distribution of CTs ($\theta^\tau_{\mathrm{ON}}$ and $\theta^\tau_{\mathrm{OFF}}$) from the learnable representation $\mathbf{R}^\tau_{\mathbf{P}}$. The proposed $\mathcal{M}$ consists of two linear layers followed by ReLU and Sigmoid, which produce 1024 and 2 channels, respectively. To provide a robust representation that can reflect the intrinsic attributes of ECs without the interference of abnormal cases, we

randomly sample $\mathrm{N_s}$ events sequences and input corresponding tokens to decoders during the inference phase. After that, the last output representation is extracted as the $\mathbf{R}^\tau$ to improve the outputs of the network.

C. Reconstruction With the Guidance of Learnable Parameters

To incorporate the adaptive parameter representation with the global texture representation, we develop a reconstruction network including spatiotemporal feature fusion modules. By combining the strengths of the domain-adaptive parameter representation and the global texture representation, our framework can enhance the interpretability of the reconstruction model and adapt to various scenes.

1) Network Architecture: In a nutshell, the proposed UNet-like architecture network [25] consists of a head layer, recurrent convolutional blocks (RC Block), residual blocks, spatiotemporal feature fusion blocks (SF Block), upsampling blocks (Up Block), and a predict layer. The head layer contains a 3×3 convolutional layer and ReLU activate layer to extract feature of input voxel grids $\mathbf{V}_\tau$ preliminarily. The features then go through three RC Blocks to halve the feature size and double its dimensions, respectively. Each RC Block contains a 3×3 convolutional layer and a ConvLSTM layer [26], whose state $\mathbf{S}_{\mathrm{e}}^\tau$ is recorded for next loop. The outputting feature pyramid $\{\mathbf{F}_{\mathrm{r_i}}^\tau \in \mathbb{R}^{2^i C \times \frac{H}{2^i} \times \frac{W}{2^i}}\}_{\mathrm{i}=1}^3$ is also utilized to restore the deep feature to original size of the input voxel grid $\mathbf{V}_\tau$ via three Up Blocks, each of which contains a double interpolation function and a 3×3 convolutional layer with ReLU activate function. Finally, a 1×1 convolutional layer with Sigmoid activate function is utilized as the tail Predict layer to generate a single-channel grayscale image $\hat{\mathbf{I}}_\tau$. Besides, before the Up Block, the feature $\{\mathbf{F}_{\mathrm{r_3}}^\tau\}$ is refined with two Residual Blocks, both of which contain two 3×3 convolutional layers with a skip connection.

$$\mathbf{F}_{\mathrm{r_i}}^{\tau\prime} = \mathrm{Reshape}\left(\mathrm{Conv}\left(\mathbf{F}_{\mathrm{r_i}}^\tau\right)\right), \mathbf{R}^{\tau\prime} = \mathrm{LN}\left(\mathbf{R}^\tau\right), \tag{1}$$

$$\mathbf{F}_{\mathrm{sd_{r_i}}}^\tau = \mathrm{Sigmoid}\left(\mathrm{Conv}\left(\mathrm{Reshape}'\left(\mathbf{F}_{\mathrm{r_i}}^{\tau\prime} \times \mathbf{R}^{\tau\prime}\right)\right)\right). \tag{2}$$

2) Spatiotemporal Feature Fusion Module: We refine feature pyramid $\{\mathbf{F}_{\mathrm{r_i}}^\tau\}_{\mathrm{i}=1}^3$ with learnable representation $\mathbf{R}^\tau$ with SF Blocks in skip connection. The details of SF Block are shown in Fig. 2, which add $\mathbf{R}^\tau$ to each feature in the feature pyramid in both channel-dimension and spatial-dimension. As described in (1), in the spatial dimension, the input feature $\mathbf{F}_{\mathrm{r}}^\tau$ and $\mathbf{R}^\tau$ are squeezed to the same number of channels (2 in our work) with 1×1 convolutional layer and linear layer, respectively. $\mathrm{Reshape}(\cdot)$ is a function that convert the shape of vector from $2^i C \times \frac{H}{2^i} \times \frac{W}{2^i}$ to $\frac{HW}{2^{i+1}} \times 2^i C$. Then, a 1 channel feature map $\mathbf{F}_{\mathrm{sd_{r_i}}}^\tau$ with the same size as $\mathbf{F}_{\mathrm{r_i}}^\tau$ can be get via matmul product between $\mathbf{F}_{\mathrm{r_i}}^{\tau\prime}$ and $\mathbf{R}^{\tau\prime}$ as described in (2). $\mathrm{Reshape}'(\cdot)$ is used to convert the shape of vector from $\frac{HW}{2^{i+1}} \times 1$ to $1 \times \frac{H}{2^i} \times \frac{W}{2^i}$. In the channel dimension, the input feature $\mathbf{F}_{\mathrm{r_i}}^\tau$ is fed into the average pooling layer and a 1×1 convolutional layer to obtain a 1-dimension feature with the same size as $\mathbf{R}^\tau$. Then, the feature is summed with $\mathbf{R}^\tau$ and is fed to a 1×1 convolutional layer to extract a 1×1 feature map $\mathbf{F}_{\mathrm{cd_{r_i}}}^\tau \in \mathbb{R}^{2^i C \times 1 \times 1}$ with same channels as $\mathbf{F}_{\mathrm{r}}^\tau$. This process can be formally described as

$$\mathbf{F}_{\mathrm{cd_{r_i}}}^\tau = \mathrm{Sigmoid}\left(\mathrm{Conv}\left(\mathbf{R}^\tau + \mathrm{Conv}\left(\mathrm{AvgPool}\left(\mathbf{F}_{\mathrm{r_i}}^\tau\right)\right)\right)\right). \tag{3}$$

Finally, as shown in (4), both of feature maps $\mathbf{F}_{\mathrm{sd_{r_i}}}^\tau$ and $\mathbf{F}_{\mathrm{cd_{r_i}}}^\tau$ can be multiplied with the $\mathbf{F}_{\mathrm{r_i}}^\tau$ from their own dimension and added to the original feature $\mathbf{F}_{\mathrm{r_i}}^\tau$ as the final output $\mathbf{F}_{\mathrm{r_i}}^{\tau\prime\prime}$.

$$\mathbf{F}_{\mathrm{r_i}}^{\tau\prime\prime} = \mathbf{F}_{\mathrm{sd_{r_i}}}^\tau \odot \mathbf{F}_{\mathrm{r_i}}^\tau + \mathbf{F}_{\mathrm{cd_{r_i}}}^\tau \odot \mathbf{F}_{\mathrm{r_i}}^\tau + \mathbf{F}_{\mathrm{r_i}}^\tau. \tag{4}$$

In this way, the features that contain parameter priors of ECs can be sufficiently integrated into the reconstruction network to improve the final reconstruction image quality.

D. Loss Functions

The proposed network contains two stages: target domain CTs estimation and video reconstruction. In the first stage, the MSE loss is used to minimize the gap between the estimated values and ground truth CTs. As for the second stage, we adopt perceptual similarity (LPIPS) [27] to optimize the quality of a single reconstruction frame, and temporal consistency loss [28] to ensure the time-continuous of reconstructed frame sequences. Here, we summarize the complete loss function as

$$\mathcal{L}_{\mathrm{Total}} = \sum_{l=0}^{L-1} \left(\mathcal{L}_l^{\mathrm{LPIPS}} + \omega_{\mathrm{TC}} \mathcal{L}_l^{\mathrm{TC}} + \omega_{\mathrm{PE}} \mathcal{L}_l^{\mathrm{MSE}}\right), \tag{5}$$

where ω_{TC} and ω_{PE} are weighting hyperparameters, L is the maximum length of video sequence. We set $\omega_{\mathrm{TC}} = 2$, $\omega_{\mathrm{PE}} = 1$, and $L = 40$ in our work.

III. Experiments

A. Experiment Settings

1) Dataset: The proposed method is trained on the synthetic events generated by [33]. Similar as [9], we first select abundant images from MS-COCO 2017 dataset [34] and warp several of foreground images on background images to generate 281 high-framerate videos. Then, we randomly product parameters of [33] and utilize them to simulate the training events. We evaluate our proposed methods on four public event-based datasets, including IJRR [29], MVSEC [30], HQF [9], and EVIMO [31] datasets.

2) Benchmarks: We compare our approach with 9 kinds of recent event-based video reconstruction methods, which can be divided into two categories. The first kind of method focuses on enhancing performance by designing novel network architectures and training directly on a simulated dataset (DT), which contains E2VID [35], FireNet [5], SPADE-E2VID [6], SSL-E2VID [7], ET-Net [8], HyperE2VID [20]. The second kind of method aligns the training data to a specific target domain to retrain a more effective event-to-video model (RT), which contains EGAN [21], S2R [9], EFLOW [32].

3) Implementation Details: In our experiment, the Transformer $\mathcal{T}$ is trained for 40 epochs with a batch size of 256, Adam optimization. The initial learning rate is set to $1e^{-4}$. $\mathrm{N_s}$ is 256, $\mathrm{N}_{\mathcal{E}}$ is 512, and the hidden feature channel number is 512 in our work. For our proposed video reconstruction network $\mathcal{F}$, we set C to 32 and crop the event voxel with the corresponding image to the size of 112×112 and train it for 115 epochs with a batch size of 16 on $1\times$ RTX 3090 GPU. Adam optimization with an initial learning rate is $1e^{-5}$. Besides, we measure inference time by recording the average runtime time in milliseconds on IJRR dataset with $1\times$ GTX 1080 GPU (8 GB)).

B. Performance Evaluation

1) Quantitative and Qualitative Experiments: We evaluate the performance on four public event datasets, recorded with

TABLE I
VIDEO RECONSTRUCTION RESULTS ON FOUR PUBLIC DATASETS

	Method	Inference Time (ms)	IJRR [29] MSE↓	IJRR [29] SSIM↑	IJRR [29] LPIPS↓	MVSEC [30] MSE↓	MVSEC [30] SSIM↑	MVSEC [30] LPIPS↓	HQF [9] MSE↓	HQF [9] SSIM↑	HQF [9] LPIPS↓	EVIMO [31] MSE↓	EVIMO [31] SSIM↑	EVIMO [31] LPIPS↓
DT	E2VID [5]	12.22	0.212	0.424	0.350	0.342	0.173	0.736	0.142	0.533	0.445	0.036	0.825	0.182
DT	FireNet [5]	**3.40**	0.133	0.501	0.321	0.310	0.241	0.727	0.110	0.548	0.507	0.030	0.835	0.192
DT	SPADE-E2VID [6]	30.58	0.091	0.517	0.337	0.137	0.333	0.585	0.095	0.507	0.556	0.020	0.848	0.183
DT	SSL-E2VID [7]	12.22	0.092	0.500	0.380	0.116	0.360	0.691	0.110	0.528	0.540	0.019	0.852	0.150
DT	ET-Net [8]	34.43	0.047	0.617	0.224	0.115	0.353	0.499	0.045	0.650	0.329	0.012	0.855	**0.147**
DT	HyperE2VID [20]	21.82	0.033	0.650	**0.212**	**0.072**	**0.411**	**0.485**	0.049	**0.643**	0.343	0.015	0.836	0.205
RT	EGAN [21]	12.22	0.098	0.458	0.325	0.171	0.251	0.574	0.070	0.498	0.433	0.020	0.841	0.159
RT	S2R [9]	12.22	0.070	0.559	0.236	0.145	0.324	0.543	0.049	0.645	**0.323**	0.015	0.852	0.148
RT	EFLOW [32]	12.22	0.067	0.544	0.295	0.132	0.312	0.502	0.045	0.640	0.359	0.011	0.857	0.155
	Ours	15.26	**0.021**	**0.654**	0.259	0.087	0.384	0.486	**0.041**	0.650	0.377	**0.010**	**0.859**	0.166

Best two results are highlighted in bold and underline.

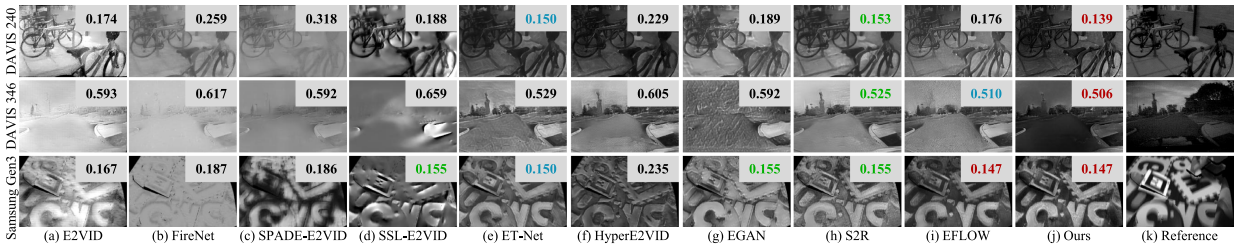

Fig. 3. Video reconstruction results on event data captured using different event cameras (ECs). The sequences from rows 1 to 3 correspond to HQF, MVSEC, and EVIMO datasets, respectively. The best three results based on the LPIPS metric are shown in red, blue, and green, respectively. Best viewed in color.

TABLE III
ABLATION STUDY ON CTS ESTIMATION ON IJRR

Method	ECC [39]	S2R [9]	JAER [12]	LGAN [11]	EFLOW [32]	Ours
RMSE	37.56	32.37	35.10	32.30	32.34	**31.21**
PSNR	16.82	18.10	17.39	18.15	18.17	**18.43**
SSIM	0.409	**0.448**	0.422	0.421	0.415	0.434

Best in bold.

TABLE II
ABLATION STUDY OF SF BLOCK ON IJRR AND MVSEC

Method	Branch CD	Branch SD	IJRR [29]/MVSEC [30] MSE↓	SSIM↑	LPIPS↓
Baseline	✗	✗	0.067 / 0.177	0.542 / 0.287	0.304 / 0.548
Baseline+CF	✓	✗	0.040 / 0.152	0.594 / 0.319	0.307 / 0.546
Baseline+SF	✗	✓	0.048 / 0.164	0.575 / 0.304	0.287 / 0.552
Ours	✓	✓	**0.021 / 0.087**	**0.654 / 0.384**	**0.259 / 0.486**

Best in bold.

various ECs such as DAVIS 240 [36], DAVIS 346 [37] and Samsung Gen3 [38]. The quantitative results are shown in Table I, from which we can see that the proposed method achieves better performance throughout whole four datasets, especially in MSE and SSIM indexes. It proves the adaptive representation can provide sufficient target domain information, which boosts the superior generalization of proposed model. Confronting challenging scenarios, involving low light conditions on MVSEC dataset, depicted in rows 2 of Fig. 3, our method is guided by the adaptive representation significantly and achieves higher LPIPS index. On the other three datasets, the proposed method outperforms the transformer architecture network ET-Net and the latest network HyperE2VID with a great trade-off between accuracy and efficiency. Table I and Fig. 3 illustrate that our approach excels at generating more aesthetically pleasing and high-fidelity reconstruction images using events from different ECs, while previous methods face challenges in handling all datasets effectively. The qualitative experimental results showcase clearer edges, fewer artifacts, and more details in the reconstruction of diverse events captured with various ECs.

2) Effect of Learnable Parameter Representation: We follow [39] to directly integrate estimated CTs with the corresponding events to generate intensity images and evaluate the quality of images as metrics. The quantitative experimental results on IJRR dataset are shown in Table III. The results demonstrate that the parameter representation learned by the proposed $\mathcal{T}$ can reflect the nature of ECs effectively.

3) Effect of SF Block: We carry out ablation studies on the IJRR and MVSEC datasets and summarize the results in Table II. We first present a Baseline model as a reference group, which does not introduce learnable representation. Then as shown in rows 2 and 3, we fuse learnable representation in channel-dimension and spatial-dimension, respectively. By comparing row 1 with rows 2 and 3, we can conclude that each branch is practical and the introduction of learnable representation can motivate the network explicitly. Finally, the proposed SF Block is assembled in the last row. The comparison from rows 2 to 4 indicates that employing both feature fusion strategies can enhance reconstruction tasks, leading to improved performance.

IV. CONCLUSION

In this letter, we propose a two-stage framework for event-based video reconstruction, which achieves greater generalization across multiple public datasets by learning the parameter representations of ECs and fusing them with elaborate SF Blocks. The inspiring experimental performance illustrates the potential of learning ECs representation, offering benefits for other event-based downstream tasks in future work, such as optical-flow estimation [40] and 3D reconstruction [41].

REFERENCES

[1] P. Lichtsteiner, C. Posch, and T. Delbrück, "A 128 × 128 120 dB 15 μs latency asynchronous temporal contrast vision sensor," *IEEE J. Solid State Circuits*, vol. 43, no. 2, pp. 566–576, Feb. 2008.

[2] I. Schiopu and R. C. Bilcu, "Lossless compression of event camera frames," *IEEE Signal Process. Lett.*, vol. 29, pp. 1779–1783, 2022.

[3] G. Gallego et al., "Event-based vision: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 1, pp. 154–180, Jan. 2022.

[4] X. Zhang, W. Liao, L. Yu, W. Yang, and G.-S. Xia, "Event-based synthetic aperture imaging with a hybrid network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 14235–14244.

[5] H. Rebecq, R. Ranftl, V. Koltun, and D. Scaramuzza, "Events-to-video: Bringing modern computer vision to event cameras," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3857–3866.

[6] P. R. G. Cadena, Y. Qian, C. Wang, and M. Yang, "SPADE-E2VID: Spatially-adaptive denormalization for event-based video reconstruction," *IEEE Trans. Image Process.*, vol. 30, pp. 2488–2500, 2021.

[7] G. Orchard, A. Jayawant, G. Cohen, and N. V. Thakor, "Converting static image datasets to spiking neuromorphic datasets using saccades," *Frontiers Neuroscience*, vol. 9, 2015, Art. no. 437.

[8] W. Weng, Y. Zhang, and Z. Xiong, "Event-based video reconstruction using transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 2543–2552.

[9] T. Stoffregen et al., "Reducing the sim-to-real gap for event cameras," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 534–549.

[10] M. Planamente et al., "DA4Event: Towards bridging the sim-to-real gap for event cameras using domain adaptation," *IEEE Robot. Automat. Lett.*, vol. 6, no. 4, pp. 6616–6623, Oct. 2021.

[11] D. Gu, J. Li, Y. Zhang, and Y. Tian, "How to learn a domain-adaptive event simulator?," in *Proc. 29th ACM Int. Conf. Multimedia Conf.*, 2021, pp. 1275–1283.

[12] C. Brandli, L. Müller, and T. Delbrück, "Real-time, high-speed video decompression using a frame- and event-based DAVIS sensor," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2014, pp. 686–689.

[13] P. Bardow, A. J. Davison, and S. Leutenegger, "Simultaneous optical flow and intensity estimation from an event camera," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 884–892.

[14] C. Reinbacher, G. Graber, and T. Pock, "Real-time intensity-image reconstruction for event cameras using manifold regularisation," in *Proc. Brit. Mach. Vis. Conf.*, 2016.

[15] C. Scheerlinck, N. Barnes, and R. E. Mahony, "Continuous-time intensity estimation using event cameras," in *Proc. Asian Conf. Comput. Vis.*, 2018, vol. 11365, pp. 308–324.

[16] P. R. G. Cadena, Y. Qian, C. Wang, and M. Yang, "Sparse-E2VID: A sparse convolutional model for event-based video reconstruction trained with real event noise," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 4150–4158.

[17] Y. Lu, D. Shi, R. Li, Y. Zhang, L. Jing, and S. Yang, "SCSE-E2VID: Improved event-based video reconstruction with an event camera," in *Proc. IEEE Int. Conf. Syst., Man, Cyber.*, 2022, pp. 3249–3254.

[18] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Representations*, 2021.

[19] L. Zhu, X. Wang, Y. Chang, J. Li, T. Huang, and Y. Tian, "Event-based video reconstruction via potential-assisted spiking neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 3584–3594.

[20] B. Ercan, O. Eker, C. Saglam, A. Erdem, and E. Erdem, "HyperE2VID: Improving event-based video reconstruction via hypernetworks," *IEEE Trans. Image Process.*, vol. 33, pp. 1826–1837, 2024.

[21] A. Z. Zhu, Z. Wang, K. Khant, and K. Daniilidis, "EventGAN: Leveraging large scale image datasets for event cameras," in *Proc. IEEE Int. Conf. Comput. Photography*, 2021, pp. 1–11.

[22] J. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2242–2251.

[23] A. Vaswani et al., "Attention is all you need," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[24] Q. Shi, Z. Ye, J. Wang, and Y. Zhang, "QISampling: An effective sampling strategy for event-based sign language recognition," *IEEE Signal Process. Lett.*, vol. 30, pp. 768–772, 2023.

[25] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. 18th Int. Conf. Med. Image Comput. Comput.-Assisted Intervention*, 2015, pp. 234–241.

[26] X. Shi, Z. Chen, H. Wang, D. Yeung, W. Wong, and W. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2015, pp. 802–810.

[27] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 586–595.

[28] W. Lai, J. Huang, O. Wang, E. Shechtman, E. Yumer, and M. Yang, "Learning blind video temporal consistency," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 179–195.

[29] E. Mueggler, H. Rebecq, G. Gallego, T. Delbrück, and D. Scaramuzza, "The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and SLAM," *Int. J. Robot. Res.*, vol. 36, no. 2, pp. 142–149, 2017.

[30] A. Z. Zhu, D. Thakur, T. Özaslan, B. Pfrommer, V. Kumar, and K. Daniilidis, "The multivehicle stereo event camera dataset: An event camera dataset for 3D perception," *IEEE Robot. Automat. Lett.*, vol. 3, no. 3, pp. 2032–2039, Jul. 2018.

[31] L. Burner, A. Mitrokhin, C. Fermüller, and Y. Aloimonos, "EVIMO2: An event camera dataset for motion segmentation, optical flow, structure from motion, and visual inertial odometry in indoor scenes with monocular or stereo algorithms," 2022, *arXiv:2205.03467*.

[32] D. Gu, J. Li, L. Zhu, Y. Zhang, and J. S. Ren, "Reliable event generation with invertible conditional normalizing flow," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 2, pp. 927–943, Feb. 2024.

[33] Y. Hu, S. Liu, and T. Delbrück, "v2e: From video frames to realistic DVS events," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 1312–1321.

[34] T. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[35] T. Qin, P. Li, and S. Shen, "VINS-MONO: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, Aug. 2018.

[36] C. Brandli, R. Berner, M. Yang, S. Liu, and T. Delbrück, "A 240× 180 130 dB 3 μs latency global shutter spatiotemporal vision sensor," *IEEE J. Solid State Circuits*, vol. 49, no. 10, pp. 2333–2341, Oct. 2014.

[37] G. Taverni et al., "Front and back illuminated dynamic and active pixel vision sensors comparison," *IEEE Trans. Circuits Syst. II: Exp. Briefs*, vol. 65, no. 5, pp. 677–681, May 2018.

[38] H. E. Ryu, "Industrial DVS design; key features and applications," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019.

[39] Z. Wang, Y. Ng, P. van Goor, and R. E. Mahony, "Event camera calibration of per-pixel biased contrast threshold," in *Proc. Australas. Conf. Robot. Automat.*, 2019.

[40] S. Shiba, Y. Aoki, and G. Gallego, "Fast event-based optical flow estimation by triplet matching," *IEEE Signal Process. Lett.*, vol. 29, pp. 2712–2716, 2022.

[41] A. R. Mangalore, C. S. Seelamantula, and C. S. Thakur, "Neuromorphic fringe projection profilometry," *IEEE Signal Process. Lett.*, vol. 27, pp. 1510–1514, 2020.