Repurposing Pre-trained Video Diffusion Models for Event-based Video Interpolation
Supplementary Material

Contents

A. Video Results

Please refer to our project page: https://vdm-evfi.github.io/ for video results, which clearly demonstrate that our reconstructions provide superior consistency and generalization compared to other baselines.

B. Video Generation Task Results

As explained in the main paper, our method supports Event-based Video Generation, an extrapolation task that relies on only one frame (start or end) and events, unlike interpolation, which uses both frames. This constraint in video generation leads to error accumulation in the generated video, as shown in the last video of the website. We present a comparison of video generation and interpolation results on the BS-ERGB dataset, for video generation, only the left-end frame next to skip frames and corresponding events are used to generate the skipped frames, as shown in Table 1, relying on information from only one side instead of both start and end frames causes the PSNR and SSIM metrics to drop significantly compared to the interpolation results.

We also show a qualitative comparison in Figure 1, as we can see because the video generation task is a extrapolation task that only uses the information from one side (start or end frame) of frames and events, it cannot avoid hallucination for the occluded/missing parts that are not present in the condition image. In contrast, the video generation task can mitigate the hallucination well by using complementary information from both frames. In addition to hallucination, relying on frame information from only one side leads to error accumulation during generation, as shown in Figure 2, the results in video generation (incorrect color accumulation on the finger) are inconsistent with the information provided by the frame on the other side. In contrast, interpolation ensures the reconstructed video remains consistent with the information from both frames. The comparison of video consistency is best observed in the last video of the website, which compares event-based video generation with event-based video interpolation.

Task	BS-ERGB (3 skips)		
	PSNR ↑	SSIM ↑	LPIPS ↓
Video Generation	25.79	0.84	0.11
Video Interpolation	27.74	0.88	0.12

Table 1. Comparison of our pipeline's performance on Video Generation and Video Interpolation tasks on the BS-ERGB dataset.

C. Clear-Motion Test Sequences

To robustly evaluate the zero-shot generalization performance of all models on unseen real-world event-based video frame interpolation scenarios, we collected the Clear-Motion Test Sequences solely for testing purposes.

C.1. Event-RGB Aligned Video Capture Setup

In this section, we will present the capture setup for our event-rgb aligned video sequences, as shown in Figure 3.

We use the Prophesee EVK4 HD as our event camera, offering a capture resolution of 1280 × 720. For the RGB camera, we use the BFS-U3-31S4C-C Blackfly S, which provides a resolution of 2048 × 1536 and supports up to 55 frames per second (fps). To align the field of view for both cameras, we utilize the Thorlabs CCM1-BS013 30 mm Cage

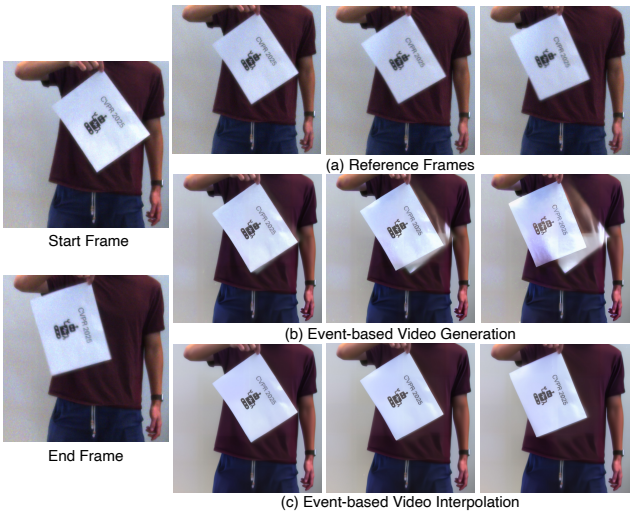


Figure 1. An example illustrating the difference between event-based video generation, which relies solely on the start frame, and event-based video interpolation, which uses both start and end frames to infer the interpolated frames. We present the 3rd, 6th, and 9th interpolated frames for the 11 skips interpolation between start and end frames on Clear-Motion test sequences. In the video generation scenario, the model hallucinates the occluded parts behind the paper due to the lack of information in the start frame. In contrast, video interpolation avoids hallucination as the end frame provides the necessary information.

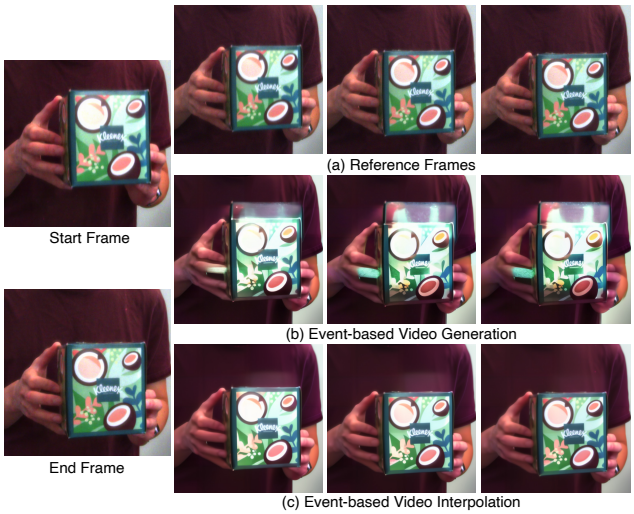


Figure 2. An example illustrating the difference in video consistency between event-based video generation and interpolation for 11 skips between start and end frames on Clear-Motion test sequences, we present the 5th, 8th, and 11th interpolated frames. In the video generation task, color errors on the finger accumulate over time, with the 11th frame (the final interpolated frame) failing to align with the information in the end frame. In contrast, interpolation reduces error accumulation and ensures consistency by leveraging information from both the start and end frames.

Cube-Mounted Non-Polarizing Beam Splitter. Additionally, we perform spatial and temporal alignment to synchronize the events with the captured RGB frames. After the spatial alighment, our final captured RGB frames aligned with event are of resolution 940 $\times$ 720 and 40 fps.

C.2. Details of Data Sequence

The Clear-Motion test sequences are designed to include clear and large motions, encompassing both camera and object movements, as well as objects and motion patterns distinct from those found in most existing real-world Event-based Video Frame Interpolation (EVFI) datasets [2, 4, 5]. This setup enables a straightforward evaluation of the generalization and consistency of various video frame interpolation methods. Table 2 provides details of our collected test sequences, including explanations for each sequence. Category (i) represents sequences with object motion, while Category (ii) represents sequences with camera motion. We also include a Figure 4 to show some example data in our test sequences.

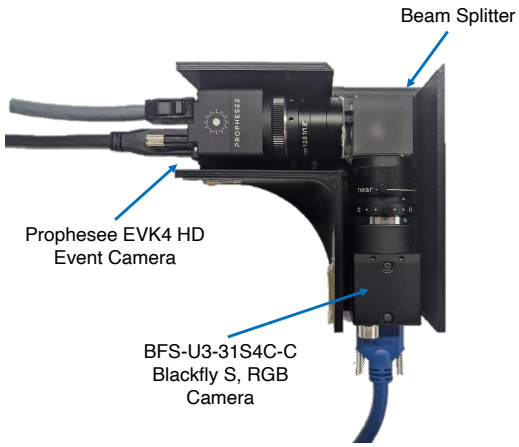


Figure 3. An illustration of our capture setup for the Clear-Motion test sequences of Event-RGB aligned video sequences. The setup consists of three main components: an event camera, an RGB camera, and a beam splitter to align the field of view for both cameras.

D. The Impact of Input Upsampling

As discussed in the main paper, to mitigate the loss of appearance and motion control accuracy caused by the conversion between downsampled latent space and pixel space in Latent Diffusion Models (LDM) [1], we employ test-time optimization in the Per-tile Denoising and Fusion process. This involves first upsampling the input image and event representations by a specified factor and then breaking them into fixed-size overlapping tiles before feeding them into

Sequence	#Frames	Explanation	Category
Paper_Shifting	200	The translational and 3D rotational motion of a paper with a simple texture	(i)
Paper_Waving	200	The waving and 3D rotational motion of a paper with a simple texture	(i)
Paper_Deforming	200	The deformation motion of a paper with a simple texture	(i)
Camera_Far	200	The moving cameras capturing distant objects	(ii)
Camera_Close	200	The moving cameras capturing nearby objects	(ii)
Checkerboard_Planar	200	The planar translation and rotation of a nearby dense checkerboard	(i)
Checkerboard_Depth	200	The motion of a checkerboard along the depth direction	(i)
Checkerboard_3D	200	The 3D translation and rotation of a nearby dense checkerboard	(i)
Texture_Box	200	The translation and rotation of a nearby highly textured box	(i)

Table 2. A detailed description of our collected Clear-Motion test sequences, with sequence name, number of frames and explanation of each sequence. Category (i) includes sequences with object motion, while Category (ii) includes sequences with camera motion.

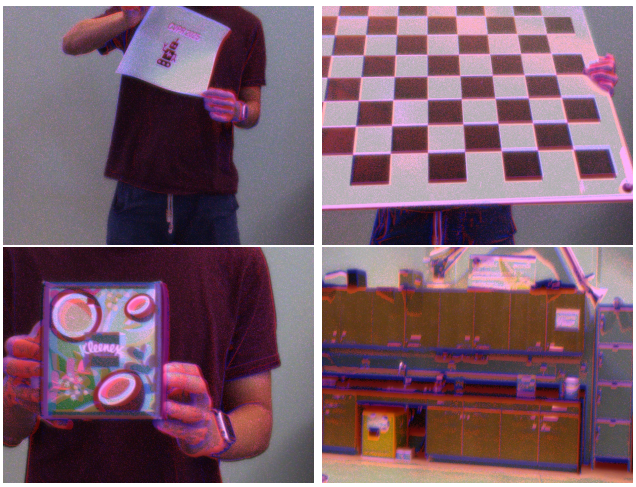


Figure 4. An illustration of example frames overlaid with events from Clear-Motion test sequences.

the video diffusion process. The performance comparison across different upsampling factors is shown in Table 3. As the upsampling factor increases from 1 to 2, our model's performance improves significantly, with PSNR increasing by approximately 3 dB, SSIM by 0.11, and LPIPS decreasing by 0.03. These results demonstrate the effectiveness of upsampling in the Per-tile Denoising and Fusion process, enhancing both the details in reconstructed frames and the accuracy of event-based motion control.

Method	BS-ERGB (3 skips)		
	PSNR ↑	SSIM ↑	LPIPS ↓
Ours_1	24.82	0.77	0.15
Ours_1.5	25.86	0.82	0.16
Ours_2	27.74	0.88	0.12

Table 3. Comparison of the impact of different upsampling factors {1, 1.5, 2} on our model's performance on the BS-ERGB dataset. We use 512 × 320 overlapping tiles with a overlapping ratio 0.1 for the input image and event representations.

To qualitatively assess the effect of upsampling, Figure 5 shows that as the upsampling factor increases from 1 to 2, details on the human eyes and fingers (e.g., nails and textures) improve significantly. Both quantitative and qualitative results highlight the effectiveness of upsampling in the Per-tile Denoising and Fusion process, enhancing the realism and event-based motion control accuracy of video interpolation results.

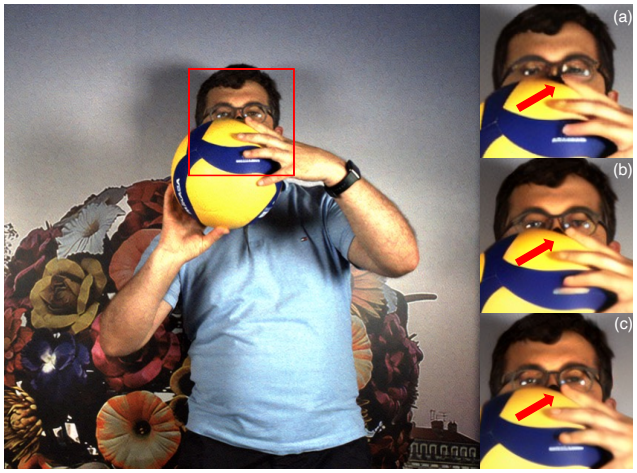


Figure 5. An illustration showcasing the qualitative impact of input upsampling factors. The leftmost image is the reference frame. (a) shows the interpolated result with an upsampling factor of 1, (b) with a factor of 1.5, and (c) with a factor of 2. As the upsampling factor increases from 1 to 2, the details on the human eyes and fingers, highlighted by red arrows, improve significantly.

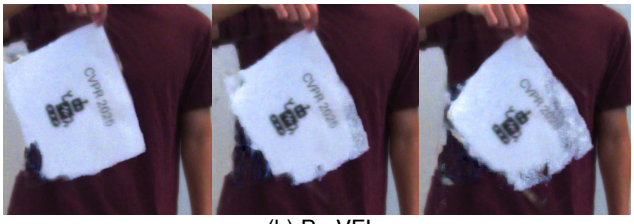
E. More Visual Results

In this section, we provide additional visual results showcasing qualitative comparisons between our method and the baselines, as shown in Figures 6, 7, 8, 9, 10, and 11.

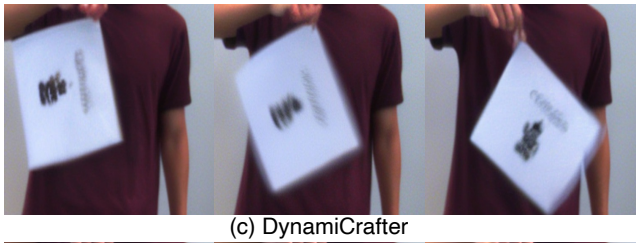
F. Additional Implementation Details

In this section, we provide additional implementation details. The pre-trained video diffusion model we used is Stable Video Diffusion [1] for 14-frame image-to-video generation. We trained our model with an effective batch size of

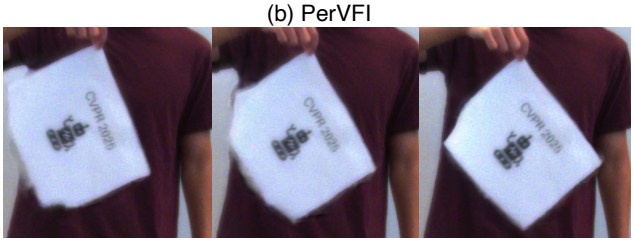
(a) Illustration of start frame, end frame and events in-between overlayed with reference frames



(b) PerVFI



(c) DynamiCrafter

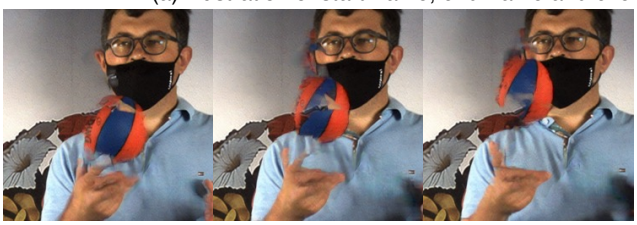


(d) GIMM-VFI

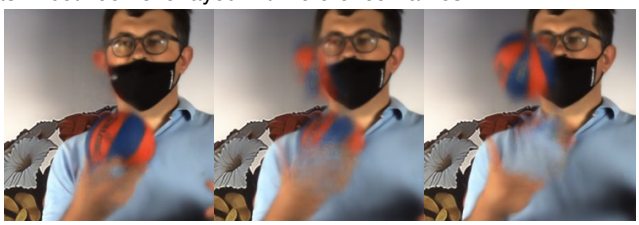
(e) Ours

Figure 6. Additional baseline results on the Clear-Motion sequence Paper_Waving.

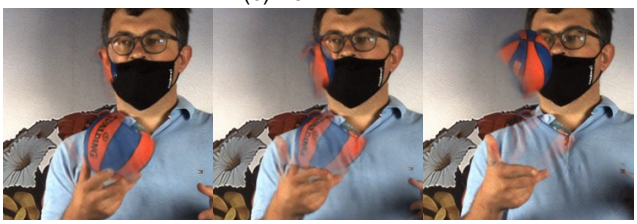
(a) Illustration of start frame, end frame and events in-between overlayed with reference frames



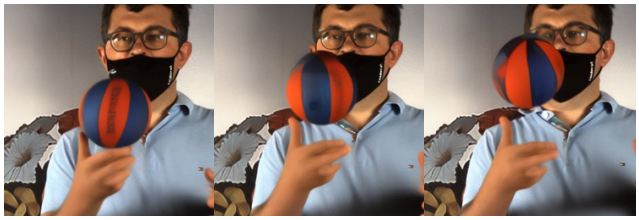
(b) PerVFI



(c) InterpAny-Clearer



(d) RIFE



(e) Ours

Figure 7. Additional baseline results on the BS-ERGB sequence as presented in the main paper.

(a) Illustration of start frame, end frame and events in-between overlayed with reference frames

(b) GIMM-VFI

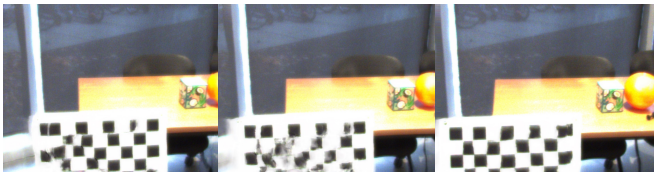
(c) Time Reversal

(d) CBMNet-Large

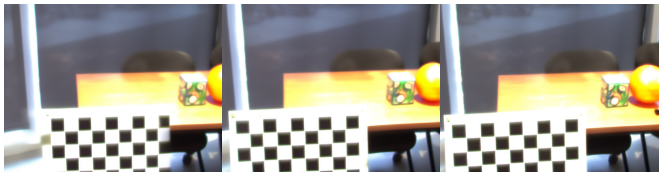
(e) Ours

Figure 8. An illustration showcasing the qualitative comparison on the Clear-Motion sequence Camera_Far, which involves large camera motion capturing distant objects, with 11 skips between the start and end frames. We present the interpolated 4th, 7th, and 10th frames. (Zoom in for the best viewing experience)

(a) Illustration of start frame, end frame and events in-between overlayed with reference frames



(b) GIMM-VFI

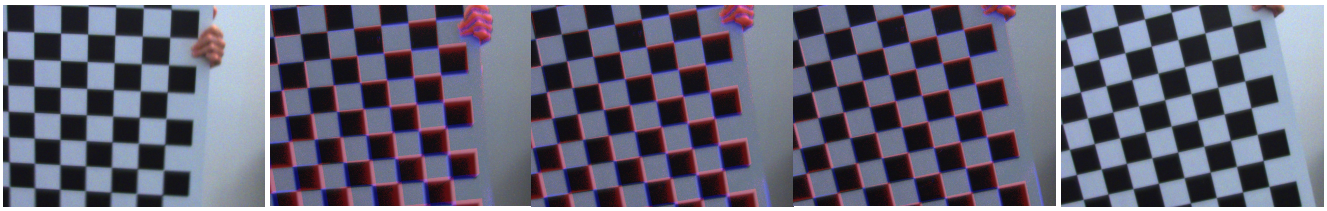


(c) Time Reversal

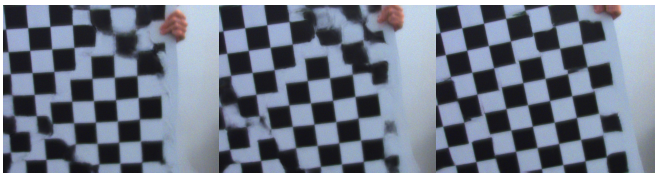
(d) CBMNet-Large

(e) Ours

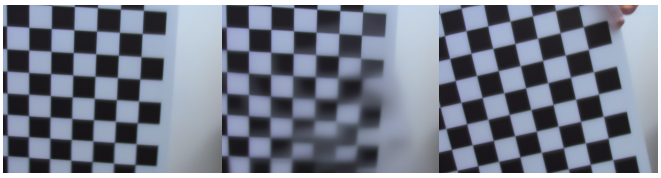
Figure 9. An illustration showcasing the qualitative comparison on the Clear-Motion sequence Camera_Close, which involves large camera motion capturing nearby objects, with 11 skips between the start and end frames. We present the interpolated 4th, 7th, and 10th frames. (Zoom in for the best viewing experience)



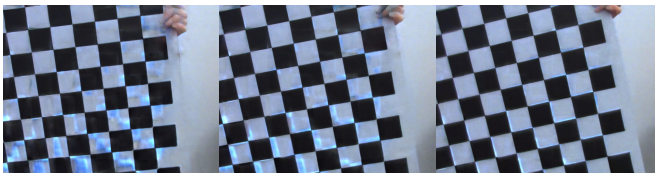
(a) Illustration of start frame, end frame and events in-between overlayed with reference frames



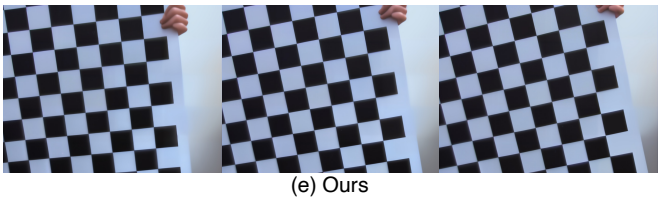
(b) GIMM-VFI



(c) Time Reversal



(d) CBMNet-Large



(e) Ours

Figure 10. An illustration showcasing the qualitative comparison on the Clear-Motion sequence Checkerboard_Planar, which involves large planar motion of a nearby checkerboard, with 11 skips between the start and end frames. We present the interpolated 4th, 7th, and 10th frames. (Zoom in for the best viewing experience)

(a) Illustration of start frame, end frame and events in-between overlayed with reference frames

(b) RIFE

(c) Time Reversal

(d) CBMNet-Large

(e) Ours

Figure 11. An illustration showcasing the qualitative comparison on the HQF dataset for the sequence poster_pillar_1, involving moving cameras capturing nearby posters, with 3 skips between the start and end frames. All interpolated frames are presented. (Zoom in for the best viewing experience)

Method	Run Time (s)	Memory Usage (GB)	Parameters (M)
RIFE	0.5	0.9	9.8
CBMNet-Large	41.7	17.8	22.2
Time-Reversal	62.6	21.5	1524.6
PerVFI	9.3	5.2	13.9
InterpAny-Clearer	0.5	1.1	10.7
DynamiCrafter	92.0	18.3	1438.9
EMA-VFI	1.8	4.8	65.7
GIMM-VFI	3.0	9.5	19.8
Ours	200.1	17.2	2206.8

Table 4. Model Run Time, Memory, and Parameter comparison.

64, using a batch size of 4 per GPU and a gradient accumulation factor of 16. Training was conducted solely on the BS-ERGB dataset, and the model was tested on other unseen datasets without fine-tuning. All training was performed on 4 NVIDIA RTX A6000 GPUs, each with 50GB of memory. For training, we use the AdamW optimizer [3] with a learning rate of 5×10^{-5} and parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1 \times 10^{-8}$, and a weight decay of 1×10^{-2}. Our model is trained on the BS-ERGB dataset for 72 hours to denoise noisy video latents for 3 and 11 skipped frames. All testing and inference on unseen data or datasets are conducted without fine-tuning, using the checkpoint trained on the BS-ERGB dataset.

During training and testing, the input to our adapted video diffusion model consists of 512 × 320 size tiles. Our model is trained solely to denoise/generate video latents using start frames and forward-time events. For testing and inference, we use an overlapping ratio of 0.1 for overlapping tiles and set the number of denoising steps in the video diffusion process to 25. The Per-tile Denoising and Fusion (for high-resolution frame reconstruction and event-based motion control) and Two-side Fusion (for converting video generation to interpolation) are both test-time optimization processes that do not require additional training.

G. Model Run Time, Memory, and Parameter Comparison

Table 4 reports testing results for all models run on a single NVIDIA RTX 4090 GPU. Each method generated 1024×576 frames with run time averaged over 16 frames. VDM based methods (Time-Reversal, DynamiCrafter, and Ours) are more memory-intensive and time-consuming than other methods.

References

[1] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 2, 3

[2] Taewoo Kim, Yujeong Chae, Hyun-Kurl Jang, and Kuk-Jin Yoon. Event-based video frame interpolation with cross-modal asymmetric bidirectional motion fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18032–18042, 2023. 2

[3] Ilya Loshchilov, Frank Hutter, et al. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 5, 2017. 7

[4] Timo Stoffregen, Cedric Scheerlinck, Davide Scaramuzza, Tom Drummond, Nick Barnes, Lindsay Kleeman, and Robert Mahony. Reducing the sim-to-real gap for event cameras. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*, pages 534–549. Springer, 2020. 2

[5] Stepan Tulyakov, Daniel Gehrig, Stamatios Georgoulis, Julius Erbach, Mathias Gehrig, Yuanyou Li, and Davide Scaramuzza. Time lens: Event-based video frame interpolation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16155–16164, 2021. 2