Supplemental Materials:
Complementing Event Streams and RGB Frames for Hand Mesh Reconstruction

Jianping Jiang[†1,2,3], Xinyu Zhou[†4], Bingxuan Wang[1,2,3], Xiaoming Deng[‡5,6], Chao Xu[4], Boxin Shi[‡1,2,3]
[1] National Key Laboratory for Multimedia Information Processing, School of Computer Science, Peking University
[2]National Engineering Research Center of Visual Technology, School of Computer Science, Peking University
[3]AI Innovation Center, School of Computer Science, Peking University
[4]National Key Lab of General AI, School of Intelligence Science and Technology, Peking University
[5]Institute of Software, Chinese Academy of Sciences [6]University of Chinese Academy of Sciences

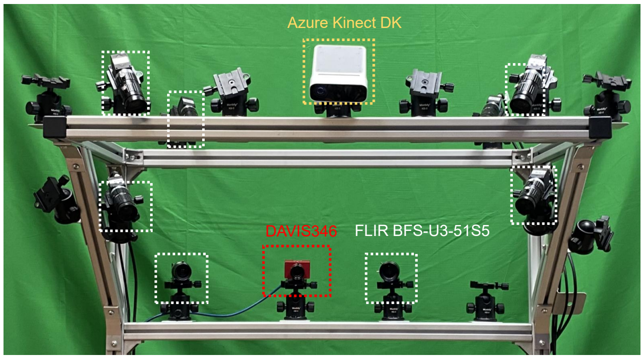


Figure 9. Multi-camera system for capturing indoor sequences. An event camera (DAVIS346, red circle) is synchronized with 7 RGB cameras (FLIR BFS-U3-51S5, yellow circles) to capture multi-view RGB images and monocular event streams. An RGB-D camera (Azure Kinect DK, white circle) is used as an auxiliary camera in the calibration step for precise calibration.

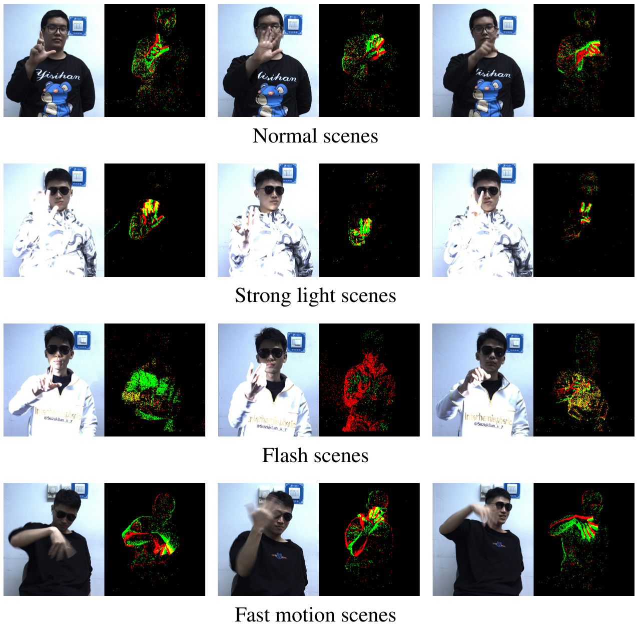
Overview. In the supplemental materials, we first introduce the details of indoor and outdoor real world datasets and synthetic dataset in Sec. 7. Then we show supplemental experiment results in Sec. 8. Finally, we illustrate the details of comparison methods in Sec. 9 and the implementation details in Sec. 10.

7. Datasets

To supplement the section of Datasets in the main paper, we show details about the indoor and outdoor sequences of EvRealHands and simulation process of the synthetic data.

7.1. Indoor Sequences

Capture system. The indoor sequences of EvRealHands is captured in a multi-camera system [4, 15]. As shown in Fig. 9, in our multi-camera system, 7 RGB cameras (FLIR, 2660×2300 pixels) and an event camera (DAVIS346, 346×260 pixels) capture data from different views simultaneously. After synchronizing all the cameras with an external 15 Hz Transistor-Transistor Logic (TTL) signal, we calibrate all the cameras with a moving chessboard [6] with RGB images from FLIR camera, APS frame from DAVIS346, and depth images from the RGB-D camera.


Normal scenes

Strong light scenes

Flash scenes

Fast motion scenes

Figure 10. Examples of indoor sequences from EvRealHands. RGB frames (left) and corresponding event streams (right) in normal, strong light, flash and fast motion scenes.

Data acquisition. We show examples from our dataset in Fig. 10. In the sequence of normal scenes, we capture RGB

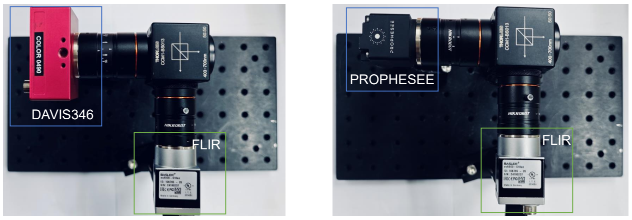


Figure 11. Hybrid camera system with an event camera and an RGB camera.

images without motion blur under everyday indoor lighting. When subjects keep hands static, the foreground scarcity issue of event-based Hand Mesh Reconstruction (HMR) appears. We capture 457 seconds of data under strong light by keeping two glare flashlights on with 2000 lumen. We set the exposure time of 6 annotation RGB cameras to 0.5 ms to avoid overexposure and that of 1 reference RGB camera to 15 ms to make its RGB images overexposed. Therefore, we obtain images with high-quality from annotation cameras for multi-view annotation and overexposed images from the reference camera as training and evaluation data. To simulate background overflow issue, we collect sequences under flash light of 317 seconds by making flashlights strobe at 1 Hz. Besides, we also collect 69 seconds of fast motion sequences. To simulate motion blur issues of RGB-based HMR, the subjects shake hands rapidly and fingers appear as ghost in the images.

Annotation. Following [12], we first annotate 21 2D keypoints on each RGB view with Mediapipe [17] and correct the unqualified annotations manually. By triangulating 2D keypoints from 7 RGB views, we obtain 3D joints. Then we fit the MANO model to the 3D joints to get the hand shape for each timestamp.

7.2. Outdoor Sequences.

Capture system. In order to collect data for qualitatively evaluation of the generalization performance of existing methods in outdoor scenarios, we build a hybrid camera system to collect data for qualitatively measuring the generalization performance of existing methods in outdoor scenarios. As shown in Fig. 11, the hybrid camera system consists of an RGB camera (FLIR BFS-U3-51S5), an event camera (DAVIS346 Mono or PROPHESEE GEN 4.0) and a beam-splitter (Thorlabs CCM1-BS013).

Data acquisition. We collected 12 sequences of 240 seconds from three subjects, of which 6 sequences are captured using DAVIS346 and the rest using PROPHESEE. The outdoor sequences face challenging issues, such as varying natural light conditions, pedestrian interference, and motion blur (including 6 sequences with fast motion).

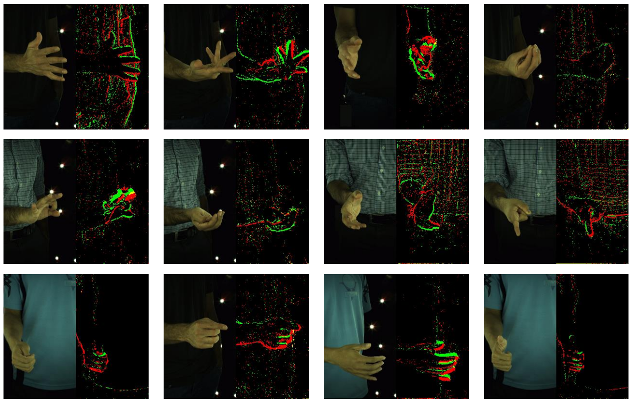


Figure 12. Visualization of our synthetic dataset generated using INTERHAND2.6M [12] and v2e event simulator [7]. Examples of RGB frames (left) and corresponding event streams (right) are displayed side by side.

7.3. Synthetic data

Although EventHands [14] proposes a synthetic dataset to the community, there exists domain gap between the used synthetic pose and real-world pose. Therefore, we use the event simulator v2e [7] to synthesize event streams from a large-scale RGB-based sequential hand dataset INTERHAND2.6M [12]. INTERHAND2.6M captures 2.6 million images from 80∼140 multi-view cameras with various hand poses. Considering that the image resolution (512×334 pixels) in INTERHAND2.6M is different from that of DAVIS346 camera, we first use affine transformation to warp the RGB images as the same scale of real-world event streams (346×260 pixels) and feed them into the v2e simulator [7] to get synthetic event streams. In our synthesizing setup, the positive threshold is set as 0.143 and the negative threshold is 0.225. RGB frames are interpolated ten times to increase the time resolution of synthetic events. In our experiment, we select the right hand sequences of 9 camera views from 4 subjects.

8. Supplemental experiment results

To further evaluate our proposed method, we will illustrate evaluation metrics in Sec. 8.1, show additional qualitative results in Sec. 8.2 and introduce more quantitative results in Sec. 8.3.

8.1. Evaluation metrics

Accuracy. MPJPE/MPVPE is root-aligned mean per joint/vertex position error in Euclidean distance (mm). It measures the distance between predicted and ground truth results. PA-MPJPE/PA-MPVPE measures the MPJPE/MPVPE between ground truth coordinates and 3D aligned predicted coordinates using Procrustes Analysis

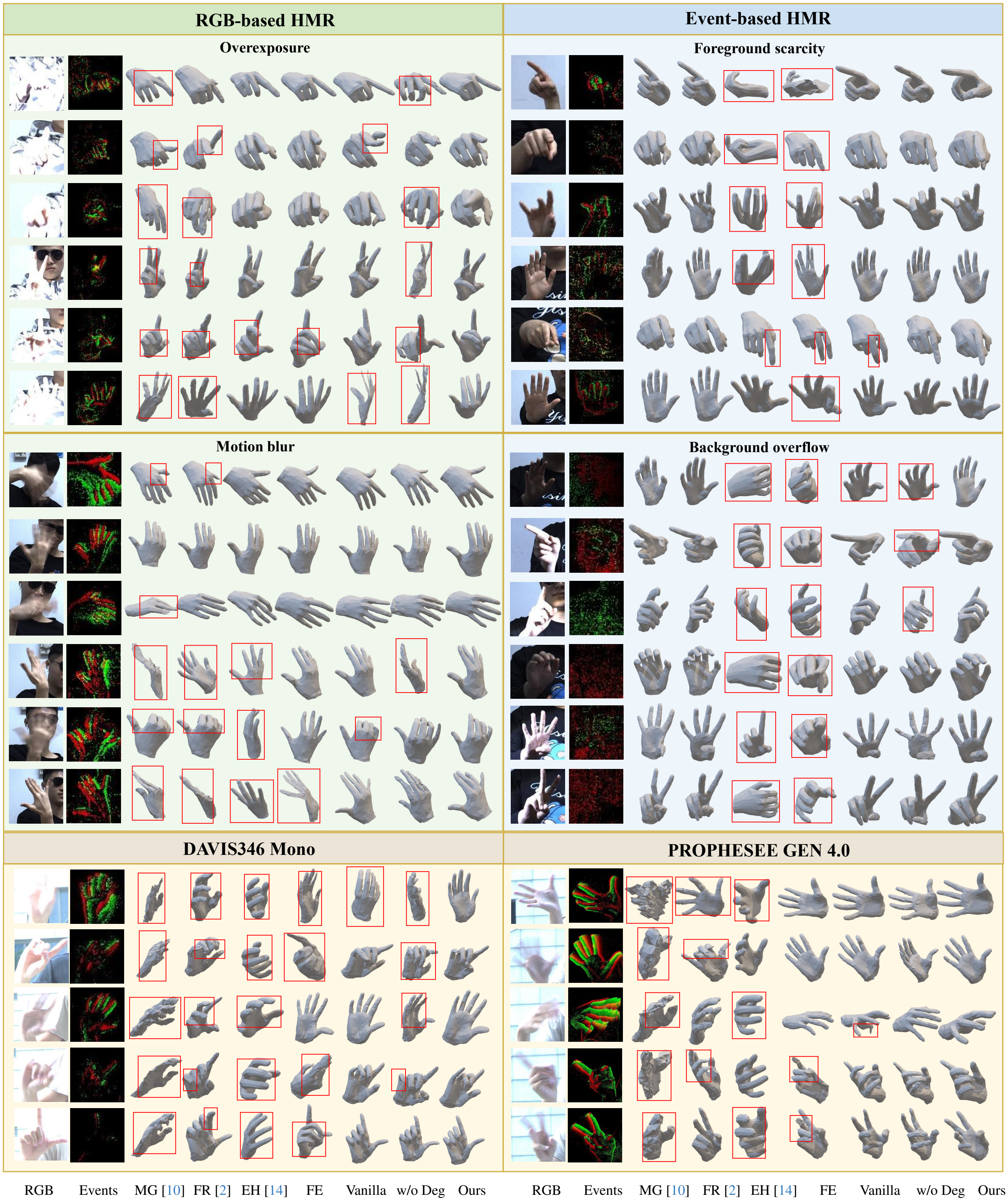


Figure 13. Additional qualitative analysis of HMR methods under challenging issues (green box titled with '*RGB-based HMR*' and blue box titled with '*Event-based HMR*'), outdoor scenes (camel box titled with '*DAVIS346 Mono*'), and PROPHESEE sequences (camel box titled with '*PROPHESEE GEN 4.0*'). For each issue, columns from left to right are RGB images, events, results from Mesh Graphormer (MG) [10], FastMETRO-RGB (FR) [2], EventHands (EH) [14], FastMETRO-Event (FE), EvRGBHand-vanilla (Vanilla), EvRGBHand without EvRGBDegrader (w/o Deg) and EvRGBHand (Ours).

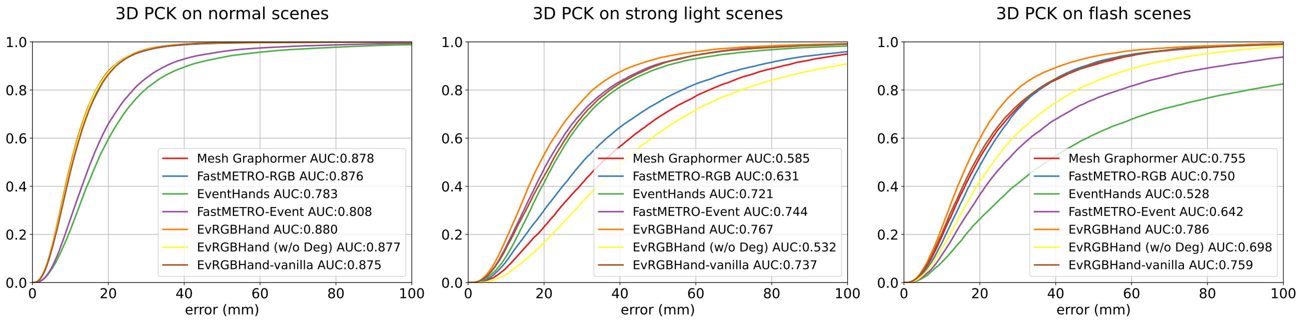


Figure 14. 3D PCK curves of EvRGBHand and other baselines.

(PA) [3]. This metric ignores the scale and global rotation. AUC is the area under the curve of PCK (percentage of correct keypoints) with thresholds ranging from 0∼100 mm for 3D annotated sequences. The lower the metrics above are, the better, except for AUC.

Computational cost. FLOPs is the floating point operations per inference and Params is the count of parameters.

8.2. More qualitative results

As shown in Fig. 13, we show more qualitative results of the comparison between EvRGBHand and other baselines. These qualitative results demonstrate the complementary effects and generalization ability of EvRGBHand for HMR with events and images.

To fully leverage the high temporal resolution property of event cameras, we achieve high frame rate inference via an asynchronous fusion strategy. Specifically, the event stream with high temporal resolution can be split into discrete temporal bins. These bins, representing discrete event intervals, are configured to surpass the frame rate of traditional RGB cameras in frequency. Subsequently, each of these temporal bins undergoes fusion with the latest RGB frame, facilitated by EvImHandNet. The temporal relationship between the timestamp t_i of an event bin and the timestamp t_j of the corresponding RGB frame can be formulated as follows:

$$j = \arg\min_{k} |t_i - t_k|, t_i - t_k \geq 0. \tag{1}$$

8.3. 3D PCK curves and AUC.

We show 3D PCK curves of the baselines and EvRGBHand under several scenes in Fig. 14. The results show that EvRGBHand outperforms all the methods based on a single sensor on AUC. By complementary usage of events and images, EvRGBHand achieves a higher AUC (0.07 ∼ 0.14) than event-based HMR on normal scenes and flash scenes, and RGB-based HMR on strong light scenes.

9. Details of comparison methods

As shown in Fig. 15, we provide additional explanations about the structures of FastMETRO-Event and EvRGBHand-vanilla. FastMETRO-Event derives from the RGB-based HMR approach, FastMETRO [2]. FastMETRO [2] is an encoder-decoder based transformer framework by disentangling the image embedding and mesh estimation, which can achieve fast convergence, low computation cost, and comparable accuracy. The only difference between FastMETRO-Event and FastMETRO [2] lies in the input: FastMETRO-Event utilizes an event representation instead of an RGB image. Despite this simple substitution, it has outperformed the current state-of-the-art event-based method, EventHands [14].

EvRGBHand-vanilla is built upon the FastMETRO [2] framework, integrating event features and image features as tokens into a transformer encoder. This approach follows the fashion of contemporary multi-modal fusion methods [1, 8, 16].

10. Implementation details

For event representation, we set $N = 7000$ for evaluation. While for training step, the number of events in each stacked event frame is selected randomly from 5000 $\sim$ 9000 for data augmentation. We apply geometric augmentation including scale, rotation and translation.

The details of EvRGBDegrader are as follow:

- **Overexposure (OE):** Color jitter augmentation is adopted with a probability of 0.4 to change the image brightness. And the brightness factor is randomly selected from 0.8 to 4.
- **Motion blur (MB):** Motion blur augmentation is applied with a 0.3 probability. To synthesize blurry images, we first apply video interpolation via estimated optical flow to increase 15 fps videos to 120 fps ones. Then a single blurry hand image is generated by averaging 17 consecutive frames, which are interpolated from 3 sharp sequential frames.

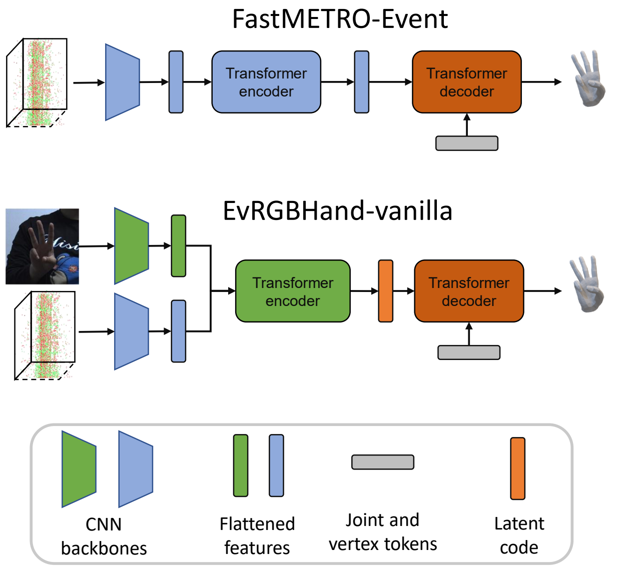


Figure 15. Brief structures of FastMETRO-Event and EvRGBHand-vanilla proposed in the main paper.

- **Background overflow (BO):** Salt-and-pepper noise is applied to each pixel with a probability of 0.2.

Moreover, event camera will emit temporally noisy outputs caused by the quantal nature of photons and events with leak noise from junction leakage and parasitic photocurrent [7, 13]. These noises are noticeable in strong light and flash scenes. For data augmentation on event streams, we add Gaussian noise with a probability of 0.8 on event streams to simulate temporal noise. The deviation of Gaussian noise is randomly selected from 0.05 to 0.2.

In order to effectively extract hand features, we crop the frames with bounding boxes. We first obtain 3D joints at the target time by linear interpolation (specially for the stacked event frame) and project the 3D joints onto the image plane to get 2D keypoints, which can be exactly covered by an rectangle. The bounding box is a square which shares the same center with the rectangle and has 1.6 times the length of the longer side of the rectangle. The sizes of bounding boxes are 192×192 for both RGB frames and stacked event frames. In our experiments, we use ResNet [5] as our CNN backbones. The number of transformer blocks L is set to 3 and the hidden state dimensions of L blocks are 256. The number of transformer heads is set to 8. For the vertex and joint loss functions, $\lambda_{\mathbf{V}}$ is 100 and $\lambda_{\mathbf{J}}$ is 2000. The initial learning rate is set to 0.0001 and we apply a cosine annealing schedule [11]. We use Adam [9] as the optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and no weight decay. We train EvRGBHand with a batch size of 32 for 50 K iteractions on 2 NVIDIA TITAN X GPUs.

References

[1] Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. VATT: Transformers for multimodal self-supervised learning from raw video, audio and text. In *NeurIPS*, 2021. 4

[2] Junhyeong Cho, Kim Youwang, and Tae-Hyun Oh. Cross-attention of disentangled modalities for 3D human mesh recovery with transformers. In *ECCV*, 2022. 3, 4

[3] John C Gower. Generalized procrustes analysis. *Psychometrika*, 1975. 4

[4] Shangchen Han, Beibei Liu, Randi Cabezas, Christopher D Twigg, Peizhao Zhang, Jeff Petkau, Tsz-Ho Yu, Chun-Jung Tai, Muzaffer Akbay, Zheng Wang, et al. MEgATrack: Monochrome egocentric articulated hand-tracking for virtual reality. *ACM TOG*, 2020. 1

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[6] Janne Heikkila and Olli Silvén. A four-step camera calibration procedure with implicit image correction. In *CVPR*, 1997. 1

[7] Yuhuang Hu, Shih-Chii Liu, and Tobi Delbrück. v2e: From video frames to realistic DVS events. In *CVPRW*, 2021. 2, 5

[8] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and João Carreira. Perceiver: General perception with iterative attention. 2021. 4

[9] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 5

[10] Kevin Lin, Lijuan Wang, and Zicheng Liu. Mesh graphormer. In *ICCV*, 2021. 3

[11] Ilya Loshchilov and Frank Hutter. SGDR: stochastic gradient descent with warm restarts. In *ICLR*, 2017. 5

[12] Gyeongsik Moon, Shoou-I Yu, He Wen, Takaaki Shiratori, and Kyoung Mu Lee. Interhand2. 6M: A dataset and baseline for 3D interacting hand pose estimation from a single RGB image. In *ECCV*, 2020. 2

[13] Yuji Nozaki and Tobi Delbruck. Temperature and parasitic photocurrent effects in dynamic vision sensors. *IEEE Transactions on Electron Devices*, 2017. 5

[14] Viktor Rudnev, Vladislav Golyanik, Jiayi Wang, Hans-Peter Seidel, Franziska Mueller, Mohamed Elgharib, and Christian Theobalt. EventHands: Real-time neural 3D hand pose estimation from an event stream. In *ICCV*, 2021. 2, 3, 4

[15] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *CVPR*, 2017. 1

[16] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. FLAVA: A foundational language and vision alignment model. In *CVPR*, 2022. 4

[17] Fan Zhang, Valentin Bazarevsky, Andrey Vakunov, Andrei Tkachenka, George Sung, Chuo-Ling Chang, and Matthias Grundmann. Mediapipe hands: On-device real-time hand tracking. *arXiv preprint arXiv:2006.10214*, 2020. 2