BlinkVision: A Benchmark for Optical Flow, Scene Flow and Point Tracking Estimation using RGB Frames and Events

Yijin Li[1,2*], Yichen Shen[1*], Zhaoyang Huang[2*], Shuo Chen[1], Weikang Bian[3], Xiaoyu Shi[3], Fu-Yun Wang[3], Keqiang Sun[3], Hujun Bao[1], Zhaopeng Cui[1], Guofeng Zhang[1†], and Hongsheng Li[3,4,5†]

[1] State Key Lab of CAD&CG, Zhejiang University
[2] Avolution AI [3] CUHK MMLab [4] Shanghai AI Laboratory
[5] CPII under InnoHK

Abstract. Recent advances in event-based vision suggest that they complement traditional cameras by providing continuous observation without frame rate limitations and high dynamic range which are well-suited for correspondence tasks such as optical flow and point tracking. However, so far there is still a lack of comprehensive benchmarks for correspondence tasks with both event data and images. To fill this gap, we propose **BlinkVision**, a large-scale and diverse benchmark with rich modality and dense annotation of correspondence. BlinkVision has several appealing properties: **1) Rich modalities:** It encompasses both event data and RGB images. **2) Rich annotations:** It provides dense per-pixel annotations covering optical flow, scene flow, and point tracking. **3) Large vocabulary:** It incorporates 410 daily categories, sharing common classes with widely-used 2D and 3D datasets such as LVIS and ShapeNet. **4) Naturalistic:** It delivers photorealism data and covers a variety of naturalistic factors such as camera shake and deformation. BlinkVision enables extensive benchmarks on three types of correspondence tasks (i.e., optical flow, point tracking and scene flow estimation) for both image-based methods and event-based methods, leading to new observations, practices, and insights for future research. The benchmark website is https://www.blinkvision.net/.

1 Introduction

Modern image-based computer vision technology still cannot match the accuracy and robustness of human vision in many areas. One possible reason is that traditional cameras suffer from motion blur and limited frame rates, and they often rely on well-lighted conditions. In contrast, event cameras [13, 36] detect changes in intensity at each pixel as a stream of asynchronous events, which

[*] Yijin Li, Yichen Shen, and Zhaoyang Huang contributed equally to this work.
[†] Hongsheng Li and Guofeng Zhang are the corresponding authors.

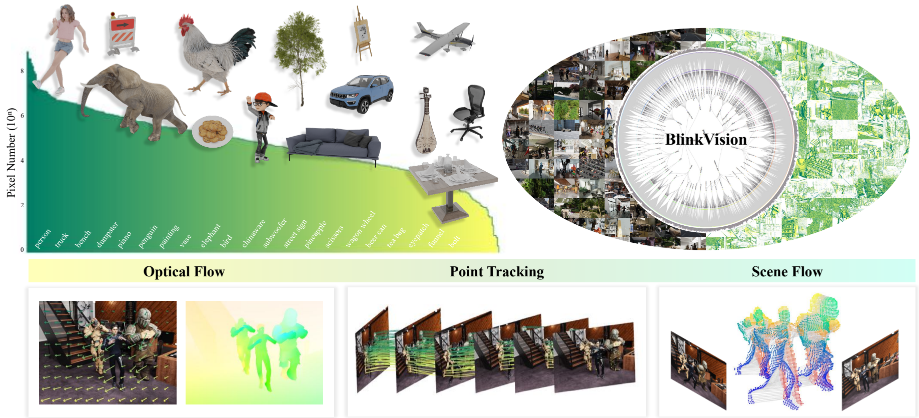

Fig. 1: BlinkVision is a large-scale and diverse benchmark with rich modality and dense annotation of correspondence. It covers 410 daily categories, sharing common classes with popular 2D and 3D datasets. The per-category object distributions, scene structure hierarchy, data samples, and supported applications of BlinkVision are shown in this figure.

eliminates frame rate limitations and enables operation within a high dynamic range. However, they can not capture fine-grained details as traditional cameras do. Looking at how human vision works, we find that two cells in the human retina, i.e., cones and rods, work similarly to these two types of cameras, respectively. According to the duplex theory of vision [45], which posits that rods and cones serve different functions, their combination ensures the robustness of human visual processing. Therefore, we believe that combining the advantages of traditional cameras and event cameras can significantly enhance computer vision systems, providing more comprehensive and adaptive vision capabilities.

However, there are currently only a few benchmarks [15, 32, 34] providing both event data and RGB frames, which hinder the development of algorithms that fully exploit the event data and fuse information from both modalities. This shortfall is particularly prominent in the domain of pixel correspondence estimation [8, 27, 28, 61], i.e., optical flow, scene flow, and point tracking estimation. Previous benchmarks built for pixel correspondence are either highly biased to specific scenes [15, 63] or rather simple [53]. A primary factor is that obtaining such precise pixel-wise annotations for these tasks is expensive.

To boost the research in this area, we present BlinkVision, a synthetic benchmark for optical flow, scene flow and point tracking estimation using RGB frames and events. Our dataset has several appealing properties: **1) Rich modalities:** BlinkVision encompasses three visual modalities: final RGB images, clean RGB images and event data. The final RGB images reflect real-world challenges like motion blur and limited dynamic range while the clean RGB images are devoid of such imperfections. The clean RGB images can be seen as the latent images [48]

Table 1: A comparison between BlinkVision and other widely-used benchmarks on pixel correspondence estimation. "Occ", "Chara" and "Cats" is the abbreviation for occlusion, character and categories, respectively. R^{lvis} denotes the ratio of the 1.2k LVIS [20] categories being covered. "N/A" denotes the dataset does not provide category labels.

Datasets	Event	Clean	Final	Optical Flow	Scene Flow	Point Tracking	Occ	Chara	Animal	Cats	R^{lvis}(%)
MVSEC [64]	✓	✗	✓	✓	✗	✗	✗	✗	✗	N/A	0
DSEC [15]	✓	✗	✓	✓	✗	✗	✗	✗	✗	N/A	0
BlinkFlow [34]	✓	✓	✗	✓	✗	✗	✓	✓	✗	55	4.1
EKubric [59]	✓	✓	✗	✓	✓	✓	✓	✗	✗	17	0.9
Sintel [9]	✗	✓	✗	✓	✗	✗	✓	✓	✓	N/A	0
KITTI [17]	✗	✗	✓	✓	✓	✗	✗	✗	✗	N/A	0
FlyingThings [42]	✗	✓	✗	✓	✓	✗	✓	✗	✗	55	4.1
Kubric [18]	✗	✓	✗	✓	✓	✓	✓	✗	✗	17	0.9
TAP-Vid [12]	✗	✓	✗	✗	✗	✓	✗	✓	✗	N/A	0
PointOdyssey [62]	✗	✓	✗	✗	✗	✓	✓	✓	✓	N/A	0
BlinkVision (Ours)	✓	✓	✓	✓	✓	✓	✓	✓	✓	410	33.4

of event cameras. **2) Rich annotations:** It provides annotations covering optical flow, scene flow, and point tracking. Unlike benchmarks [15, 17, 62] which only contain sparse annotations, our data provide dense per-pixel annotations of each image, covering objects including moving cars, deformable characters, and animals. **3) Large vocabulary:** It incorporates 410 daily categories, sharing common classes with widely-used 2D and 3D datasets such as ImageNet [33], LVIS [20], and ShapeNet [10], as depicted in Fig. 1. To our knowledge, our data have the widest variety of objects in existing pixel correspondence benchmarks. This expansive vocabulary is pivotal, enabling rigorous exploration of algorithmic generalization across varied objects. **4) Naturalistic:** BlinkVision employs high-quality assets and rendering tools and thus is able to deliver photorealism data. Besides, it covers a variety of naturalistic factors such as camera shake and deformation.

To make full use of BlinkVision, we set up a public benchmark website that allows uploading results and provides a public leaderboard. Besides, we evaluated both existing image-based and event-based methods on three typical correspondence tasks (i.e. optical flow, point tracking, and scene flow estimation). The results reveal new observations and challenges and serve as the baseline for future approaches. Specifically, we first study the robustness of existing image-based methods under large frame intervals and extreme illumination and point out the new challenge for these methods. Second, the benchmark results on existing event-based methods show that current methods do not fully unleash the potential of event cameras. Third, we show that fine-tuning existing methods on the training set of BlinkVision significantly boosts the generalizability, demonstrating the vast diversity of BlinkVision. Finally, the broad diversity and accessible category labels in BlinkVision allow us, for the first time, to analyze the performance of correspondence tasks on different categories. We believe BlinkVision has the potential to serve as a cornerstone benchmark for advancing the development of more robust computer vision systems.

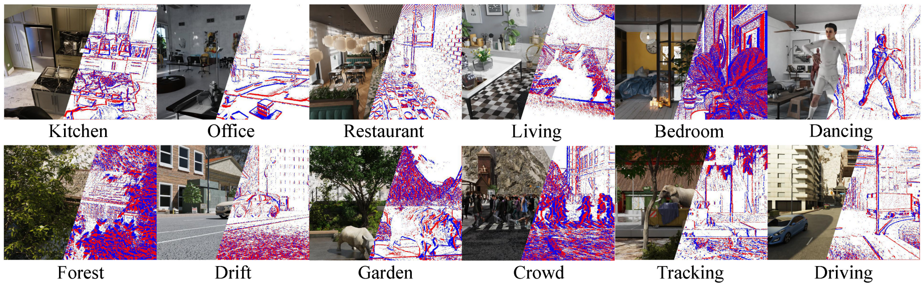

Fig. 2: Scenes samples in the proposed BlinkVision benchmark.

2 Related Work

A comprehensive overview and comparison between these benchmarks and BlinkVision is listed in Table 1.

Image-based optical flow and scene flow benchmarks. Early evaluations depended on synthetic datasets, like the famous "Yosemite" sequence [7]. MPI Sintel [9] was one of the most representative synthetic benchmarks derived from a short open-source animated 3D movie and it became one of the most popular benchmark datasets. KITTI [17] was almost the most well-known among real data. It computed ground truth for static scenes through the data from a 3D laser scanner and the ego-motion data of the car. In a later version, KITTI extended the ground truth to rigidly moving cars by fitting CAD models of cars. FlyingThings3D [42] was a more recent synthetic dataset. It contained the ground truth of both optical flow and scene flow. FlyingThings3D and KITTI were the two most commonly used benchmarks in scene flow evaluation. However, on both datasets, ground truth about scene flow was not available in occluded regions. Furthermore, on KITTI, ground truth for foreground points was either missing or approximated by a fitted car CAD model.

Image-based point tracking benchmarks. The first point-tracking dataset was FlyingThings++ [21], which was based on FlyingThings3D. It was initially designed for training the network. Due to the lack of evaluation benchmarks, it was also used for evaluation in the early stages. Later, Doersch *et al.* [12] proposed a real data benchmark named TAP-Vid, which was based on two real-world datasets: DAVIS [49] and Kinetics [30]. TAP-Vid relied on manual annotation and could not handle occlusions. PointOdyssey [62] was a newly proposed benchmark that was based on synthetic data and therefore could provide dense ground truth. However, it did not guarantee that every pixel had ground truth because it only tracked the mesh vertices of the object. Furthermore, the naturalism of PointOdyssey was limited to its interior parts. Its outdoor portion had no realistic layout, just a skybox with randomly dropped objects.

Event-based Benchmarks. Early event-based optical flow benchmarks [53] limited the camera to only rotational motion and inferred ground truth from the rotational motion of the camera. Two later benchmarks, DSEC [15] and MVSEC [64] computed the ground truth through LiDAR SLAM. Similar to KITTI, these two benchmarks limited the ground truth to the static elements of the scene. Besides, they had a rather limited motion pattern. More recently, Li *et al.* [34] proposed a large-scale diversiform synthetic benchmark named BlinkFlow, which presented more challenging scenes and complex motion patterns. As for the scene flow benchmark, Wan *et al.* [59] recently converted the existing FlyingThings [42] and Kubric [18] datasets to event datasets through video-to-event [14] technology. However, due to the limited frame rates in the original video data, this step inevitably produced artifacts in the event data. Wan *et al.* also extended the real-world event dataset DSEC with scene flow ground truth. As for the event-based point-tracking benchmark, existing methods usually employed Structure-from-Motion [54] (SfM) technology to associate keypoints across multiple frames to obtain correspondence ground truth. Event Camera Dataset [46] and EDS dataset [22] were two common benchmarks in this area. Limited by the sparsity and accuracy of SfM, these benchmarks did not support the evaluation of arbitrary point tracking.

3 BlinkVision Dataset

In this section, we describe how to build BlinkVision, including the scene setup, data rendering, and generation for the ground truth labels. The process is based on Blender [2] as it provides photorealistic rendering and a flexible data interface that allows us to obtain customized correspondence ground truth. At the last, we introduce the statistics and distribution of the BlinkVision data.

3.1 Scene Setting

Previous synthetic benchmarks [9, 43] generally rely on open-source movies to avoid heavy scene construction. However, these movies are biased and do not cover diverse enough scenarios. In order to establish a comprehensive evaluation benchmark, we manually assemble a collection of scenarios that are as rich and diverse as possible. We first look for ready-made scenes that are photorealistic and visually diverse. Specifically, we purchased 40 indoor scenes and 13 outdoor scenes from Evermotion Archinteriors Collection [3] that cover common scenes such as living rooms, kitchens, offices, bedrooms, restaurants and gardens. The original scene is static. To enhance the realism, we procured 29 scanned and high-fidelity human bodies from ActorCore [1] and 88 artist-designed animals, together with 104 free characters from Mixamo [4]. These assets are rigged. We re-targeted them to a wide range of motion models (e.g., more than 100 human motions including dancing, walking, talking, etc.) and placed them in various locations. The human's motions mainly come from motion capture [1] and the animal's motions are designed by the artist to closely emulate their natural motions. Furthermore, we built 50 additional outdoor scenes that cover cases such as a drone flying through the forest and a camera following a human or car from the crowd, which are common but rarely included in existing correspondence benchmarks. We set the camera trajectory by referencing the shooting trajectories of real-world videos, including handheld shots, car shots, and drone shots that cover non-uniform motion such as sudden stops and sharp turns. Finally, we obtained 80 indoor sequences and 63 outdoor sequences, where 56 indoor sequences and 47 outdoor sequences are for testing and the others are used for fine-tuning. Some samples of our data are shown in Fig. 2. In the supplementary, we also provide video samples. All the assets used in the testing and fine-tuning are totally disjoint, even including trees.

3.2 Multi-modality Data Rendering

We use Blender to render brightness B in linear color space. The following simulations of real-world image capturing and event data are both based on the linear color space. We employ tone mapping and gamma correction on B to obtain commonly used sRGB images. We call these sRGB images "RGB (clean)" because they do not suffer from effects like blurs or overexposed. On the contrary, we call the simulated real-world image capture "RGB (final)".

Simulation of Real-world Image Capturing. Real-world image capture suffers from motion blur and limited dynamic range. While Blender supports the simulation of the former, the latter needs additional processing. The latter effect usually leads to overexposed or underexposed. To simulate it, we follow previous work [40] and employ the following steps: (1) Random exposure ratio. We uniformly sample the exposure ratio w in the log2 space within $[-3, 3]$ and multiply it by B to simulate underexposed or overexposed, which gives us the augmented HDR image $H = B \times 2^w$. (2) Dynamic range clipping. We clip H according to the formulation $\mathcal{C}(H) = \min(H, 1)$. This step leads to information loss for pixels in the overexposed regions. (3) Non-linear mapping. To align with how humans see a scene, a camera typically uses a non-linear camera response function (CRF) to modify the contrast of the captured image, which can be formulated by $I_n = \mathcal{F}(I_c)$. We randomly sample CRFs from an existing dataset [19]. (4) Quantization. The pixel values are quantized to 8 bits by $\mathcal{Q}(I_n) = \lfloor 255 \times I_n + 0.5 \rfloor / 255$. This step causes information loss in underexposed and smooth gradient areas.

Event Data. Event cameras work by responding to changes in the logarithmic brightness signal (i.e., $L = \log B$) asynchronously and independently for each pixel [13]. An event is triggered when the change in brightness (either increase or decrease) since the last event at that pixel reaches a threshold of $\pm T$ (with $T > 0$):

$$p_k(L(u_k, t_k) - L(u_k, t_k - \Delta t)) \geq T, \tag{1}$$

where Δt is the time since the last event triggered at pixel location u_k and at time t_k. $p_k \in \{-1, 1\}$ is the polarity of the brightness change. To model the event generation process, we need access to a continuous representation of the

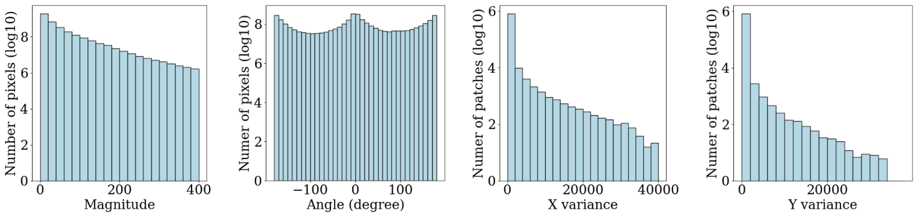

Fig. 3: Statistics of optical flow in BlinkVision.

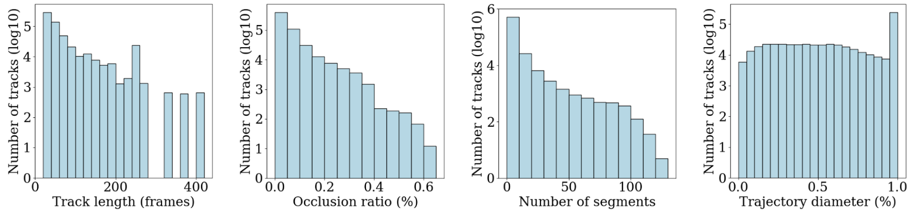

Fig. 4: Statistics of point trajectories in BlinkVision. To save time, we sample the tracks using a grid with a size of 20. Trajectory segments are defined as contiguous sections of point trajectories, with interruptions caused by occlusion. The diameter is the maximum distance a point moves over time. We clip the trajectory diameter and divide it by the diagonal length of the image to obtain the ratio.

visual signal for each pixel. In practice, it is approximated by rendering images at high frame rates for efficiency [34, 51]. Events are then synthesized based on frame-by-frame pixel differences. More specifically, we adaptively sample frames according to [34, 51] to ensure that the maximum pixel displacement between two sampling timestamps is bounded. In practice, we employ frame interpolation before simulating the event data to further reduce the pixel displacement. We use DVS-Voltmeter [37] to synthesize events because it can simulate complex noise effects (such as noise effects of temperature and parasitic photocurrent), thus generating realistic events.

3.3 Ground Truth Generation

Blender provides optical flow data between two consecutive frames and a segmentation mask for each object. However, we cannot directly obtain these data from Blender: (1) pixel correspondence across multiple frames; (2) scene flow between frames; (3) semantic category for each object.

Pixel Correspondence. The data of (1) and (2) can be computed in a unified framework. Given camera poses $\mathtt{T}$ (camera-to-world), depth maps $\mathtt{Z}$ at frames i and j, and the camera intrinsic K, for the pixel location $\mathtt{u}$ at frame i, we first project it to world coordinate: $P = T_i Z_i(\mathtt{u}) K^{-1} \tilde{\mathtt{u}}$, where $\tilde{\mathtt{u}}$ is the homogeneous coordinates of $\mathtt{u}$. Then we re-project the 3D point P to frame j through $d\mathtt{u}_j = KT_j^{-1}P$, where $\mathtt{u}_j$ is the corresponding pixel location and d is the depth value at frame j. In this way, we get the pixel correspondence between any two frames and the corresponding depth, i.e., (1) and (2).

However, this only works for static objects. To handle dynamic objects and even those objects with deformation motion such as humans and animals, we bake the face index and barycentric coordinate $(\lambda_1, \lambda_2, \lambda_3)$ of each mesh face into textures. After rendering the texture, we locate the triangular face that the tracked pixel belongs to (indexed by the rendered face index) and obtain the vertices' 3D position (V_1, V_2, V_3) of that face via Blender's API. The 3D position of tracked pixels can be computed through $P = \lambda_1 V_1 + \lambda_2 V_2 + \lambda_3 V_3$. This step replaces the back projection for the static objects. It is worth noting that we need to triangulate every mesh face in the scenes before baking.

Semantic Labeling. BlinkVision contains thousands of objects which makes manual semantic labeling expensive. To this end, we develop an automatic labeling framework that is based on open-vocabulary semantic understanding [50,60]. First, we pre-define a category list based on LVIS [20]. The category together with category descriptions is encoded into text embedding $F_t \in \mathbb{R}^{N\times D}$ with a pre-trained text encoder, i.e., CLIP, where N is the number of categories and D is the feature dimension. For each object asset, we render it individually to a 224×224 image. The process is rather fast while keeping photorealistic because it does not need to compute complex path tracing between objects. We use the pre-trained image encoder from CLIP [50] to extract the image embedding $F_i \in \mathbb{R}^D$. The probability of the image belonging to one of the pre-defined categories is computed through $P(c|F_i) = \mathrm{softmax}\left(F_i \cdot (F_t)^\top\right)_c$.

3.4 Dataset Overview

BlinkVision consists of 40 training sequences and 103 testing sequences. The training set provides 107,880 frame pairs for optical flow and scene flow, and 1025 sub-sequences for point tracking. The frame rate for RGB images and ground truth data is 20 FPS. The frame rate is not practically significant because we adjusted it to ensure sufficient motion between adjacent frames. This adjustment reduces the data size uploaded to the benchmark website. For each sequence of BlinkVision, we selected the sub-sequences by starting tracking from the first frame. Once the overlap with the reference frame (i.e., how many tracked points remain inside the image) is smaller than 40%, we start to track a new reference frame. If the overlap is smaller than 20%, we stop this track. Additionally, we discard short tracks that are less than 20 frames long because they are less challenging without long-term motion accumulation, which could result in the benchmark easily reaching saturation. As a result, we generate 1025 sub-sequences where each sub-sequence contains 640×480 dense point trajectories. For the test set, we selected 12,804 frame pairs for evaluating optical flow and

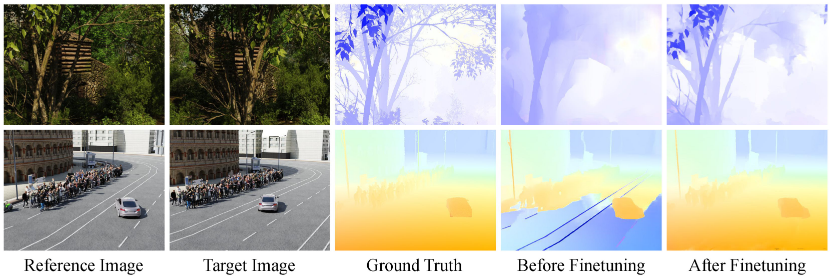

Fig. 5: Qualitative results of FlowFormer++ [55] before and after fine-tuning on the training set of BlinkVision.

scene flow, and 865 sub-sequences for evaluating point tracking. BlinkVision includes raw data on depth, normal, camera pose, etc. At this stage, we focus on pixel-level motion tasks. The additional raw data will be released later, which will benefit other tasks such as depth estimation [35] and SLAM [23,24,39].

Table 1 summarizes key statistics of BlinkVision compared to related works. BlinkVision contains more than 40K 3D models in 410 categories. It shares some common categories with 2D and 3D datasets [10,20] , and bears a huge diversity in semantics, geometry, and appearance, enabling a wide range of research topics.

We provide statistics of optical flow in Fig. 3. For benchmarking, we discard those pixels whose flow magnitudes are larger than half of the image diagonal. The latter situation usually occurs when the object is too close to the camera, and its magnitude can even approach infinity. As a result, the magnitude shown in Fig. 3 has an upper bound of 400. The broad distribution in Fig. 3 reveals the diverse motions in BlinkVision. Besides, the diverse distribution of "X Variance" and "Y Variance" also indicates the presence of complex object shapes in BlinkVision. We also provide statistics of point trajectories in Fig. 4. BlinkVision allows the evaluation of point tracking with different trajectories' lengths, from small to big, corresponding to different downstream applications like video editing [25,41] or augmented reality [31,39]. The broad distribution of "occlusion ratio" and "number of segments" indicates that BlinkVision brings diverse and challenging data. The occlusion ratio has an upper bound that is less than 100 percent due to our strategy of sub-sequences selection. To quantify the amount of motion a point undergoes, we calculate its trajectory diameter, which is the maximum distance between any two positions of the point over its entire trajectory. We observe that point motion in BlinkVision is diverse and nearly uniformly distributed across the entire image plane.

4 Benchmark

We release all the data except for the ground truth for the test split. For a fair comparison, we create a public benchmark website with leaderboards. We also release the evaluation code for our online benchmarking.

Table 2: Optical flow result of RGB-frame-based methods. "St." denotes Stride. "†" denotes fine-tuning. "FF" denotes FlowFormer [26] and "FF++" denotes FlowFormer++ [55].

Method	EPE - clean				
	St. 1	St. 2	St. 4	St. 8	Avg.
RAFT [56]	1.83	4.62	10.18	20.46	9.27
GMA [29]	1.82	4.37	9.52	18.68	8.60
FF [26]	1.60	3.77	7.57	16.03	7.24
FF++ [55]	**1.54**	**3.57**	**7.43**	**16.26**	**7.20**
RAFT† [56]	1.33	2.96	6.39	12.72	5.85
FF++† [55]	**1.09**	**2.35**	**4.88**	**10.58**	**4.73**
Method	EPE - final				
	St. 1	St. 2	St. 4	St. 8	Avg.
RAFT [56]	2.54	5.69	11.27	22.67	10.54
GMA [29]	2.56	5.72	11.21	21.35	10.21
FF [26]	2.26	4.80	9.24	18.23	8.63
FF++ [55]	**2.28**	**4.83**	**9.39**	**19.08**	**8.90**
RAFT† [56]	1.94	3.88	7.35	15.17	7.08
FF++† [55]	**1.66**	**3.34**	**6.42**	**12.73**	**6.04**

Table 3: Optical flow result of event-based methods. "St." denotes Stride. "†" denotes fine-tuning.

Event					
Methods	St. 1	St. 2	St. 4	St. 8	Avg.
E-RAFT [16]	2.81	7.04	17.82	**28.60**	14.07
STE-FlowNet [11]	2.59	5.94	13.13	41.89	15.89
E-FlowFormer [34]	**2.41**	**5.66**	**12.53**	30.45	**12.76**
E-RAFT† [16]	1.68	3.63	7.48	14.20	6.75
E-FlowFormer† [34]	**1.51**	**3.00**	**5.96**	**13.60**	**6.02**
Event + RGB (clean)					
Methods	St. 1	St. 2	St. 4	St. 8	Avg.
DCEIFlow [58]	3.31	14.07	34.78	61.56	28.43
DCEIFlow† [58]	**2.19**	**6.30**	**12.54**	**26.11**	**11.79**
Event + RGB (final)					
Methods	St. 1	St. 2	St. 4	St. 8	Avg.
DCEIFlow [58]	4.08	14.24	34.54	61.27	28.53
DCEIFlow† [58]	**2.52**	**6.86**	**13.64**	**28.45**	**12.87**

4.1 Optical Flow

In this section, we comprehensively evaluate the performance of existing optical flow methods on BlinkVision. Specifically, we first analyze the robustness of image-based methods under large frame intervals and extreme illumination. Large motion, severe occlusion, and information loss in these extreme cases pose great challenges to existing image-based methods. In contrast, event cameras naturally depict continuous pixel motion and possess a large dynamic range, thus offering great potential for solving these challenges. We then benchmark existing event-based methods, including event-only methods and event-RGB fusion methods. We find that existing event-based methods cannot fully unleash the potential of event cameras. We analyze the possible reasons and point to possible opportunities for future research.

Experimental Setup. To study the impact of frame intervals on existing methods, we select frames separated by 1, 2, 4, and 8 from the reference frame as target frames. To reduce the file size of results that are uploaded to the benchmark website, we uniformly sample a reference frame from ten consecutive frames. As a result, there are 12,804 frame pairs for testing in total. We follow the previous work [26, 47, 56] and report the EPE (end-point-error) as the evaluation metric.

Results. We benchmark several image-based methods in Table 2. We make the following observations: 1) The error generally increases linearly with stride size. This is expected because doubling the stride yields motion that is twice as fast, resulting in an error that is also expected to be twice as high. 2) Extreme lighting has a heavy effect on existing image-based methods. In the supplementary

Table 4: Point tracking result of RGB-frame-based methods under different frame interval. "St." denotes Stride.

Method	Metric	St. 1	St. 2	St. 4	St. 8	Avg.
RGB(clean)						
PIPs [21]	δ_{avg}	37.77	39.34	36.34	25.50	34.74
	Survival↑	52.28	**55.39**	**54.55**	**47.58**	**52.45**
	MTE↓	50.77	**47.81**	**49.53**	**62.42**	**52.63**
PIPs++ [62]	δ_{avg}	**40.02**	38.25	30.24	16.84	31.34
	Survival↑	**53.44**	53.13	47.57	39.71	48.46
	MTE↓	**45.51**	51.02	59.87	73.17	57.39
Context-TAP [8]	δ_{avg}	38.75	**39.47**	**36.51**	**25.99**	**35.18**
	Survival↑	49.98	53.09	52.61	46.18	50.46
	MTE↓	51.46	49.61	51.83	64.39	54.32
RGB(final)						
PIPs [21]	δ_{avg}	33.02	**34.44**	**31.56**	21.77	30.20
	Survival↑	48.35	**51.56**	**50.72**	**44.56**	**48.80**
	MTE↓	53.94	**51.54**	**53.65**	**66.14**	**56.32**
PIPs++ [62]	δ_{avg}	**35.83**	34.24	27.11	15.32	28.12
	Survival↑	**50.47**	50.36	45.32	38.46	46.15
	MTE↓	**48.37**	53.74	62.34	74.59	59.76
Context-TAP [8]	δ_{avg}	33.46	34.17	31.39	**22.00**	**30.25**
	Survival↑	45.58	48.91	48.53	43.06	46.52
	MTE↓	55.09	53.76	56.33	68.11	58.32

Table 5: Point tracking result of event-based methods. "*" denotes that we track ORB [52] feature points rather than grid sampled points. "DET" denotes DeepETracker [44]

Method	δ_{avg} ↑	Survival↑	MTE↓
Event			
AMH [5]	18.95	37.98	75.44
HASTE [6]	18.60	37.43	77.36
AMH* [5]	26.76	41.93	**62.48**
HASTE* [6]	**27.65**	**42.87**	62.52
Event + RGB (clean)			
DET [44]	21.62	35.44	80.80
DET* [44]	**33.46**	**48.82**	**60.69**
Event + RGB (final)			
DET [44]	19.05	33.52	85.00
DET* [44]	**31.78**	**48.67**	**58.79**

materials, we analyze how different exposure factors lead to performance loss. 3) Fine-tuning on the training split of BlinkVision significantly improves the performance. In Fig. 5 we show qualitative results of FlowFormer++ before and after fine-tuning on BlinkVision. After fine-tuning, the flow predictions are more precise and clear, especially near the object boundary. As shown in Table 2, the benchmark is still challenging for methods after fine-tuning.

We report the evaluation of event-based methods in Table 3. It is surprising that the performance of existing event-based methods is worse than image-based methods and they also suffer from severe degradation under large frame intervals. The reasons are mainly twofold. First, existing training data for events lag behind that of RGB. This can be verified by the performance of E-RAFT [16] and E-FlowFormer [34] after fine-tuning. These two methods are far behind RAFT [56] before fine-tuning, but exceed the fine-tuned RAFT [56] and even FlowFormer++ [55] after fine-tuning (in "Stride-1", "Stride-2" and "Stride-4" of the "final" case), indicating that the poor performance of the former mainly comes from limited training on existing event-based training data. Second, existing event-based methods usually convert events to regular 3D voxel grids before processing, which quantizes the event data and leads to information loss. This problem is more serious when dealing with event data within large frame intervals. As a result, it calls for a more powerful representation of event data and new algorithms that can deal with long-range optical flow estimation.

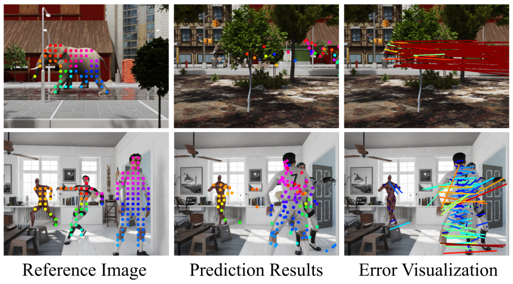

Fig. 6: Qualitative results of PIPs++ [62]. Darker lines in the right figure (e.g., red) indicate larger errors.

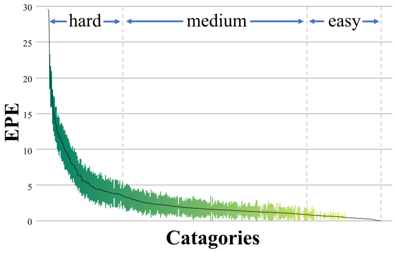

Fig. 7: Performance distribution of FlowFormer++ [55].

Table 6: Scene flow result of RGB-frame-based methods. "St." denotes Stride.

EPE-2d					
Method	RGB	St. 1	St. 2	St. 4	St. 8
RAFT-3D [57]	clean	**2.15**	**4.34**	**8.59**	**16.08**
	final	**3.48**	**7.29**	**10.74**	**18.57**
CamLiFlow [38]	clean	2.79	6.05	12.34	23.89
	final	4.28	8.28	16.30	30.61
EPE-3d					
Method	RGB	St. 1	St. 2	St. 4	St. 8
RAFT-3D [57]	clean	1.91	3.83	7.25	12.18
	final	2.02	3.96	7.39	14.21
CamLiFlow [38]	clean	**1.45**	**2.76**	**5.36**	**9.12**
	final	**1.53**	**2.88**	**5.60**	**9.81**

Table 7: Scene flow result of event-based methods. "St." denotes Stride.

EPE-2d					
Method	RGB	St. 1	St. 2	St. 4	St. 8
RPEFlow [59]	clean	1.53	3.50	7.36	17.44
	final	1.92	4.10	8.15	18.38
EPE-3d					
Method	RGB	St. 1	St. 2	St. 4	St. 8
RPEFlow [59]	clean	5.08	9.07	15.77	25.24
	final	5.20	9.38	15.67	25.03

4.2 Point Tracking

In this section, we evaluate the performance of existing methods for long-term point tracking on BlinkVision. Similar to what we analyzed on optical flow, we analyze the robustness under large frame intervals and extreme illumination, and the gaps between existing image-based methods and event-based methods.

Experimental Setup. Although BlinkVision provides per-pixel dense annotations for point tracking, we find that existing methods are not efficient enough to process so much data. As a result, we grid sample the tracked pixels with a grid size of 20 pixels. We follow PointOdyssey [62] and use δ_{avg}, median trajectory error (MTE) and survival rate as the evaluation metrics.

Results. We benchmark several image-based methods in Table 4. Similarly, we observe that extreme lighting has a major impact on existing image-based point tracking methods. In Table 4 we can see that larger frame intervals (such as 4 and 8) severely degrade performance. However, for PIPs [21] and Context TAP [8], increasing the interval from 1 to 2 makes the performance slightly better. We

Table 8: Cross-dataset evaluation of optical flow methods fine-tuned on BlinkVision. "BV" denotes BlinkVision.

RGB				
Method	Data	Sintel (clean)	Sintel (final)	KITTI
RAFT [56]	-	2.08	3.41	5.10
	+BV	**1.69**	**3.04**	**4.66**
Events				
Method	Data	E-Blender	Flying-Objects	E-Tartan
E-RAFT [16]	-	2.66	2.9	3.27
	+BV	**1.92**	**2.67**	**2.55**
E-FlowFormer [34]	-	2.38	2.89	2.91
	+BV	**1.75**	**2.58**	**2.36**
Events + RGB				
Method	Data	E-Blender	Flying-Objects	E-Tartan
DCEIFlow [58]	-	8.96	9.58	7.44
	+BV	**4.76**	**2.37**	**2.88**

Table 9: Performance evaluation for typical categories at each difficult level."FF" denotes FlowFormer [26] and "FF++" denotes FlowFormer++ [55]. We use the EPE as the evaluation metric.

Difficult	Category	RAFT [56]	FF [26]	FF++ [55]
Hard	person	9.82	8.72	**8.61**
	tree	9.79	**9.76**	10.10
	rhinoceros	8.90	7.07	**6.78**
Medium	bench	4.29	**3.60**	3.68
	globe	3.43	2.52	**2.51**
	lantern	2.44	**1.92**	1.95
Easy	notebook	1.08	**0.57**	0.84
	coaster	0.86	0.87	**0.79**
	painting	0.31	0.30	**0.26**

guess it might be the limited temporal receptive field of these two methods (only 8 frames) that makes the accumulated error quickly increase when the stride is too small. Some qualitative results are shown in Fig. 6. BlinkVision contains many challenging cases that cannot be handled by the SOTA methods.

We show the performance of event-based methods in Table 5. We find that event-based approaches perform particularly poorly. In addition to the reasons for insufficient training data and model design, we deduce that the receptive field of event-based point-tracking methods is relatively small and therefore cannot perform well on the task of tracking arbitrary points. To verify the claim, we replace the input of grid sampled positions with ORB [52] feature points, denoted by "*" in the table. The new results perform even better, validating our ideas. Although the new results cannot be strictly compared with image-based methods, it performs better under extreme frame interval, i.e., "Stride-8", which shows the large potential of event-based methods.

4.3 Scene Flow

Experimental Setup. We use the same pairs as the optical flow benchmark and we follow [38, 57] to use 2D EPE and 3D EPE as the evaluation metrics.

Results. We benchmark two state-of-the-art image-based methods, i.e., RAFT-3D [57] and CamliFlow [38] as shown in Table 6 and show the results of event-based methods in Table 7. The conclusions of the optical flow benchmark apply to the scene flow task and present similar challenges for existing methods in this area. We show qualitative results in the supplementary materials.

4.4 Cross-dataset Evaluation

We also fine-tune existing optical flow methods on the BlinkVision training set and then evaluate them on existing representative benchmarks. The results are shown in Table 8. We observe that fine-tuning on BlinkVision brings significant improvement for both image-based methods (2.08 vs. 1.69) and event-based methods (2.38 vs. 1.75 and 9.58 vs. 2.37), which demonstrates the vast diversity of BlinkVision boosts the generalizability of these methods.

4.5 Performance Distribution on Categories

Previous methods in optical flow mainly perform evaluations on biased and limited scenarios, which is not comprehensive and robust enough to demonstrate the ability of the methods for different categories of objects in different scenarios. Thanks to the vast diversity of data and semantic labels provided by BlinkVision, for the first time, we analyze the performance distribution of several image-based optical flow methods on different categories. The results are shown in Fig. 7. The average curve is imbalanced: hard categories usually include complex shapes (e.g., hammocks and shrubs) or with deformable motion (e.g., persons and animals). We thus split the categories into three levels of "difficulty" based on the average curve, and the performance evaluation for typical categories at each level is presented in Table 9. We believe such fine-grained analysis helps understand the generalization capacity of the methods.

5 Conclusion

We propose BlinkVision, a large-scale diversiform benchmark for three types of correspondence tasks, i.e., optical flow, point tracking, and scene flow estimation using RGB frames and events. Extensive benchmarks on BlinkVision point to new challenges for existing image-based approaches and show that existing event-based approaches are far from fully unlocking the potential of event cameras. BlinkVision reveals new observations, challenges, and opportunities for future research into more robust visual systems such as human vision.

Acknowledgment. This project was funded in part by National Key R&D Program of China Project 2022ZD0161100, by the Centre for Perceptual and Interactive Intelligence (CPII) Ltd under the Innovation and Technology Commission (ITC)'s InnoHK, by Smart Traffic Fund PSRI/76/2311/PR, by RGC General Research Fund Project 14204021. Hongsheng Li is a PI of CPII under the InnoHK. This work was also partially supported by NSF of China (No. 61932003).

References

1. ActorCore. https://actorcore.reallusion.com/, accessed: 2023-11-17 5
2. Blender. https://www.blender.org/, accessed: 2023-11-17 5
3. Evermotion Archinteriors Collection. https://evermotion.org/, accessed: 2023-11-17 5
4. Mixamo. https://www.mixamo.com/, accessed: 2023-11-17 5
5. Alzugaray, I., Chli, M.: Asynchronous multi-hypothesis tracking of features with event cameras. In: 2019 International Conference on 3D Vision (3DV). pp. 269–278. IEEE (2019) 11
6. Alzugaray, I., Chli, M.: Haste: multi-hypothesis asynchronous speeded-up tracking of events. In: 31st British Machine Vision Virtual Conference (BMVC 2020). p. 744. ETH Zurich, Institute of Robotics and Intelligent Systems (2020) 11
7. Barron, J.L., Fleet, D.J., Beauchemin, S.S.: Performance of optical flow techniques. International journal of computer vision **12**, 43–77 (1994) 4
8. Bian, W., Huang, Z., Shi, X., Dong, Y., Li, Y., Li, H.: Context-TAP: Tracking Any Point Demands Spatial Context Features. arXiv preprint arXiv:2306.02000 (2023) 2, 11, 12
9. Butler, D.J., Wulff, J., Stanley, G.B., Black, M.J.: A naturalistic open source movie for optical flow evaluation. In: Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12. pp. 611–625. Springer (2012) 3, 4, 5
10. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: ShapeNet: An Information-rich 3D Model Repository. arXiv preprint arXiv:1512.03012 (2015) 3, 9
11. Ding, Z., Zhao, R., Zhang, J., Gao, T., Xiong, R., Yu, Z., Huang, T.: Spatio-temporal Recurrent Networks for Event-based Optical Flow Estimation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 525–533 (2022) 10
12. Doersch, C., Gupta, A., Markeeva, L., Recasens, A., Smaira, L., Aytar, Y., Carreira, J., Zisserman, A., Yang, Y.: Tap-vid: A benchmark for tracking any point in a video. Advances in Neural Information Processing Systems **35**, 13610–13626 (2022) 3, 4
13. Gallego, G., Delbrück, T., Orchard, G., Bartolozzi, C., Taba, B., Censi, A., Leutenegger, S., Davison, A.J., Conradt, J., Daniilidis, K., et al.: Event-based Vision: A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(1), 154–180 (2020) 1, 6
14. Gehrig, D., Gehrig, M., Hidalgo-Carrió, J., Scaramuzza, D.: Video to Events: Recycling Video Datasets for Event Cameras. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3586–3595 (2020) 5
15. Gehrig, M., Aarents, W., Gehrig, D., Scaramuzza, D.: DSEC: A Stereo Event Camera Dataset for Driving Scenarios. IEEE Robotics and Automation Letters **6**(3), 4947–4954 (2021) 2, 3, 5
16. Gehrig, M., Millhäusler, M., Gehrig, D., Scaramuzza, D.: E-RAFT: Dense Optical Flow from Event Cameras. In: Proceedings of the International Conference on 3D Vision. pp. 197–206. IEEE (2021) 10, 11, 13
17. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: 2012 IEEE conference on computer vision and pattern recognition. pp. 3354–3361. IEEE (2012) 3, 4
18. Greff, K., Belletti, F., Beyer, L., Doersch, C., Du, Y., Duckworth, D., Fleet, D.J., Gnanapragasam, D., Golemo, F., Herrmann, C., et al.: Kubric: A scalable dataset

generator. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3749–3761 (2022) 3, 5

19. Grossberg, M.D., Nayar, S.K.: What is the space of camera response functions? In: 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings. vol. 2, pp. II–602. IEEE (2003) 6
20. Gupta, A., Dollar, P., Girshick, R.: Lvis: A dataset for large vocabulary instance segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5356–5364 (2019) 3, 8, 9
21. Harley, A.W., Fang, Z., Fragkiadaki, K.: Particle video revisited: Tracking through occlusions using point trajectories. In: European Conference on Computer Vision. pp. 59–75. Springer (2022) 4, 11, 12
22. Hidalgo-Carrió, J., Gallego, G., Scaramuzza, D.: Event-aided direct sparse odometry. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5781–5790 (2022) 5
23. Hu, J., Chen, X., Feng, B., Li, G., Yang, L., Bao, H., Zhang, G., Cui, Z.: Cg-slam: Efficient dense rgb-d slam in a consistent uncertainty-aware 3d gaussian field. arXiv preprint arXiv:2403.16095 (2024) 9
24. Hu, J., Mao, M., Bao, H., Zhang, G., Cui, Z.: Cp-slam: Collaborative neural point-based slam system. Advances in Neural Information Processing Systems **36** (2024) 9
25. Huang, Z., Pan, X., Pan, W., Bian, W., Xu, Y., Cheung, K.C., Zhang, G., Li, H.: Neuralmarker: A framework for learning general marker correspondence. ACM Transactions on Graphics (TOG) **41**(6), 1–10 (2022) 9
26. Huang, Z., Shi, X., Zhang, C., Wang, Q., Cheung, K.C., Qin, H., Dai, J., Li, H.: Flowformer: A Transformer Architecture for Optical Flow. In: Proceedings of the European Conference on Computer Vision. pp. 668–685. Springer (2022) 10, 13
27. Huang, Z., Shi, X., Zhang, C., Wang, Q., Li, Y., Qin, H., Dai, J., Wang, X., Li, H.: FlowFormer: A Transformer Architecture and Its Masked Cost Volume Autoencoding for Optical Flow. arXiv preprint arXiv:2306.05442 (2023) 2
28. Huang, Z., Zhou, H., Li, Y., Yang, B., Xu, Y., Zhou, X., Bao, H., Zhang, G., Li, H.: Vs-net: Voting with segmentation for visual localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6101–6111 (2021) 2
29. Jiang, S., Campbell, D., Lu, Y., Li, H., Hartley, R.: Learning to estimate hidden motions with global motion aggregation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9772–9781 (2021) 10
30. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., et al.: The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 (2017) 4
31. Klein, G., Murray, D.: Parallel tracking and mapping for small ar workspaces. In: 2007 6th IEEE and ACM international symposium on mixed and augmented reality. pp. 225–234. IEEE (2007) 9
32. Klenk, S., Chui, J., Demmel, N., Cremers, D.: Tum-vie: The tum stereo visual-inertial event dataset. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 8601–8608. IEEE (2021) 2
33. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems **25** (2012) 3
34. Li, Y., Huang, Z., Chen, S., Shi, X., Li, H., Bao, H., Cui, Z., Zhang, G.: Blinkflow: A dataset to push the limits of event-based optical flow estimation. arXiv preprint arXiv:2303.07716 (2023) 2, 3, 5, 7, 10, 11, 13

35. Li, Y., Liu, X., Dong, W., Zhou, H., Bao, H., Zhang, G., Zhang, Y., Cui, Z.: Deltar: Depth estimation from a light-weight tof sensor and rgb image. In: European conference on computer vision. pp. 619–636. Springer (2022) 9
36. Li, Y., Zhou, H., Yang, B., Zhang, Y., Cui, Z., Bao, H., Zhang, G.: Graph-based asynchronous event processing for rapid object recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 934–943 (2021) 1
37. Lin, S., Ma, Y., Guo, Z., Wen, B.: DVS-Voltmeter: Stochastic Process-Based Event Simulator for Dynamic Vision Sensors. In: Proceedings of the European Conference on Computer Vision. pp. 578–593. Springer (2022) 7
38. Liu, H., Lu, T., Xu, Y., Liu, J., Li, W., Chen, L.: Camliflow: Bidirectional camera-lidar fusion for joint optical flow and scene flow estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5791–5801 (2022) 12, 13
39. Liu, X., Li, Y., Teng, Y., Bao, H., Zhang, G., Zhang, Y., Cui, Z.: Multi-modal neural radiance field for monocular dense slam with a light-weight tof sensor. In: Proceedings of the ieee/cvf international conference on computer vision. pp. 1–11 (2023) 9
40. Liu, Y.L., Lai, W.S., Chen, Y.S., Kao, Y.L., Yang, M.H., Chuang, Y.Y., Huang, J.B.: Single-image hdr reconstruction by learning to reverse the camera pipeline. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1651–1660 (2020) 6
41. Luo, J., Huang, Z., Li, Y., Zhou, X., Zhang, G., Bao, H.: Niid-net: adapting surface normal knowledge for intrinsic image decomposition in indoor scenes. IEEE Transactions on Visualization and Computer Graphics **26**(12), 3434–3445 (2020) 9
42. Mayer, N., Ilg, E., Hausser, P., Fischer, P., Cremers, D., Dosovitskiy, A., Brox, T.: A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4040–4048 (2016) 3, 4, 5
43. Mehl, L., Schmalfuss, J., Jahedi, A., Nalivayko, Y., Bruhn, A.: Spring: A high-resolution high-detail dataset and benchmark for scene flow, optical flow and stereo. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4981–4991 (2023) 5
44. Messikommer, N., Fang, C., Gehrig, M., Scaramuzza, D.: Data-driven feature tracking for event cameras. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5642–5651 (2023) 11
45. Milner, D., Goodale, M.: The visual brain in action, vol. 27. OUP Oxford (2006) 2
46. Mueggler, E., Rebecq, H., Gallego, G., Delbruck, T., Scaramuzza, D.: The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and slam. The International Journal of Robotics Research **36**(2), 142–149 (2017) 5
47. Ni, J., Li, Y., Huang, Z., Li, H., Bao, H., Cui, Z., Zhang, G.: PATS: Patch Area Transportation with Subdivision for Local Feature Matching. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 17776–17786 (2023) 10
48. Pan, L., Scheerlinck, C., Yu, X., Hartley, R., Liu, M., Dai, Y.: Bringing a blurry frame alive at high frame-rate with an event camera. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6820–6829 (2019) 2

49. Perazzi, F., Pont-Tuset, J., McWilliams, B., Van Gool, L., Gross, M., Sorkine-Hornung, A.: A benchmark dataset and evaluation methodology for video object segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 724–732 (2016) 4
50. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021) 8
51. Rebecq, H., Gehrig, D., Scaramuzza, D.: ESIM: An Open Event Camera Simulator. In: Proceedings of the Conference on Robot Learning. pp. 969–982. PMLR (2018) 7
52. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: Orb: An efficient alternative to sift or surf. In: 2011 International conference on computer vision. pp. 2564–2571. Ieee (2011) 11, 13
53. Rueckauer, B., Delbruck, T.: Evaluation of Event-based Algorithms for Optical Flow with Ground-truth from Inertial Measurement Sensor. Frontiers in neuroscience **10**, 176 (2016) 2, 4
54. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4104–4113 (2016) 5
55. Shi, X., Huang, Z., Li, D., Zhang, M., Cheung, K.C., See, S., Qin, H., Dai, J., Li, H.: Flowformer++: Masked Cost Volume Autoencoding for Pretraining Optical Flow Estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1599–1610 (2023) 9, 10, 11, 12, 13
56. Teed, Z., Deng, J.: RAFT: Recurrent All-pairs Field Transforms for Optical Flow. In: Proceedings of the European Conference on Computer Vision. pp. 402–419. Springer (2020) 10, 11, 13
57. Teed, Z., Deng, J.: Raft-3d: Scene flow using rigid-motion embeddings. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8375–8384 (2021) 12, 13
58. Wan, Z., Dai, Y., Mao, Y.: Learning dense and continuous optical flow from an event camera. IEEE Transactions on Image Processing **31**, 7237–7251 (2022) 10, 13
59. Wan, Z., Mao, Y., Zhang, J., Dai, Y.: Rpeflow: Multimodal fusion of rgb-pointcloud-event for joint optical flow and scene flow estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10030–10040 (2023) 3, 5, 12
60. Xu, J., Liu, S., Vahdat, A., Byeon, W., Wang, X., De Mello, S.: Open-vocabulary panoptic segmentation with text-to-image diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2955–2966 (2023) 8
61. Yang, B., Huang, Z., Li, Y., Zhou, H., Li, H., Zhang, G., Bao, H.: Hybrid3d: learning 3d hybrid features with point clouds and multi-view images for point cloud registration. Science China Information Sciences **66**(7), 172101 (2023) 2
62. Zheng, Y., Harley, A.W., Shen, B., Wetzstein, G., Guibas, L.J.: Pointodyssey: A large-scale synthetic dataset for long-term point tracking. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19855–19865 (2023) 3, 4, 11, 12
63. Zhu, A.Z., Thakur, D., Özaslan, T., Pfrommer, B., Kumar, V., Daniilidis, K.: The Multivehicle Stereo Event Camera Dataset: An Event Camera Dataset for 3D Perception. IEEE Robotics and Automation Letters **3**(3), 2032–2039 (2018) 2

64. Zhu, A.Z., Yuan, L., Chaney, K., Daniilidis, K.: Ev-flownet: Self-supervised optical flow estimation for event-based cameras. In: Kress-Gazit, H., Srinivasa, S.S., Howard, T., Atanasov, N. (eds.) Robotics: Science and Systems (2018) 3, 5