COMPUTER VISION

Microsaccade-inspired event camera for robotics

Botao He[1,2]*, Ze Wang[3,4], Yuan Zhou[2,3], Jingxi Chen[1], Chahat Deep Singh[1], Haojia Li[5], Yuman Gao[2,3], Shaojie Shen[5], Kaiwei Wang[4], Yanjun Cao[3], Chao Xu[2,3], Yiannis Aloimonos[1,6,7], Fei Gao[2,3]*, Cornelia Fermüller[1,6,7]*

Neuromorphic vision sensors or event cameras have made the visual perception of extremely low reaction time possible, opening new avenues for high-dynamic robotics applications. These event cameras' output is dependent on both motion and texture. However, the event camera fails to capture object edges that are parallel to the camera motion. This is a problem intrinsic to the sensor and therefore challenging to solve algorithmically. Human vision deals with perceptual fading using the active mechanism of small involuntary eye movements, the most prominent ones called microsaccades. By moving the eyes constantly and slightly during fixation, microsaccades can substantially maintain texture stability and persistence. Inspired by microsaccades, we designed an event-based perception system capable of simultaneously maintaining low reaction time and stable texture. In this design, a rotating wedge prism was mounted in front of the aperture of an event camera to redirect light and trigger events. The geometrical optics of the rotating wedge prism allows for algorithmic compensation of the additional rotational motion, resulting in a stable texture appearance and high informational output independent of external motion. The hardware device and software solution are integrated into a system, which we call artificial microsaccade–enhanced event camera (AMI-EV). Benchmark comparisons validated the superior data quality of AMI-EV recordings in scenarios where both standard cameras and event cameras fail to deliver. Various real-world experiments demonstrated the potential of the system to facilitate robotics perception both for low-level and high-level vision tasks.

INTRODUCTION

Humans still outperform the most advanced robots in visual perception. Our visual systems have evolved over millions of years to help us efficiently obtain the information necessary to act in our environments. A characteristic of human vision is fixational eye movements, which are small, involuntary displacements of the eyeball. The largest of these eye movements are called microsaccades (*1*). They ensure that vision does not fade during fixations (*2*) by generating movement and stimuli in visual neurons and enhancing perception of spatial detail (*3*). Without microsaccades, humans cannot maintain the perception of static objects. For a demonstration, see Fig. 1 and Movie 1. The question we ask here is: Can we adopt this active perception mechanism in robot vision?

A bioinspired visual motion sensor, known as the silicon retina, dynamic vision sensor (*4*), or event camera, has recently gained increasing attention in robotics. Using analog microcircuits at every pixel, it can achieve a temporal resolution of several microseconds and has much higher dynamic range than standard cameras. Event cameras have shown great potential in many visual navigation tasks, including dynamic obstacle sensing (*5–8*), localization in challenging lighting conditions (*9–12*), and specific applications such as autonomous inspection (*13*) or space situational awareness (*14*). However, along with these functional advantages, some of their natural properties also present unique challenges.

[1]Department of Computer Science, University of Maryland, College Park, MD 20742, USA. [2]College of Control Science and Engineering, Zhejiang University, Hangzhou, China. [3]Huzhou Institute of Zhejiang University, Huzhou, China. [4]College of Optical Science and Engineering, Zhejiang University, Hangzhou, China. [5]Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong, China. [6]Institute for Advance Computer Studies, University of Maryland, College Park, MD 20742, USA. [7]Institute for Systems Research, University of Maryland, College Park, MD 20742, USA.
*Corresponding author. Email: fgaoaa@zju.edu.cn (F.G.); fermulcm@umd.edu (C.F.); botao@umd.edu (B.H.)

Event cameras only respond to motion. An event at a pixel is triggered when the logarithm of the intensity changes by a certain threshold. Thus, the readings occur at image edges but depend on both the motion and the scene texture. No events are recorded at edges parallel to the camera motion, and thus an event camera moving horizontally does not "see" horizontal scene edges. As a result, event cameras do not produce a stable and persistent texture, and they cannot maintain high informational output all the time, which makes accurate and long-term data association very difficult. However, data association is essential for most algorithms used in robot visual perception systems, such as optical flow estimation or feature tracking. The challenge of maintaining it has become a bottleneck for event-based vision in real-world applications.

In the past decade, many works attempted to eliminate this problem using software approaches. Most event-based data association methods rely on features like corner points (*15–17*) and optical flow (*18–20*). However, because of the varying texture appearance, feature detection and tracking are not accurate or stable, and so far, there are very few robotics applications. In recent years, some works (*12*, *17*, *21*, *22*) associated events with previous data maintained either in the form of two- or three-dimensional (2D/3D) event maps or reconstructed intensity images and optimized the correspondence between new and maintained data. The maintained maps or images contain more information and have enhanced texture stability, thus resulting in more robust performance. However, these methods suffer from noise when the event camera moves slowly or is static, resulting in severe robustness issues when such conditions persist over extended time intervals. Some works combined event sensors with regular cameras for optical flow estimation (*23*) and stable feature tracking (*24–26*). By fusing events with absolute brightness information, features can be detected in the intensity images and tracked with events. However, the introduction of regular cameras limits the system's dynamic range, thus hindering its application in challenging lighting environments. All of the above methods attempt to maintain a stable texture appearance using software solutions. Although they offer some mitigation, they fall short of providing a complete solution. We observed that the issues of texture instability and information loss are fundamentally introduced by sensor characteristics instead of algorithm imperfections.

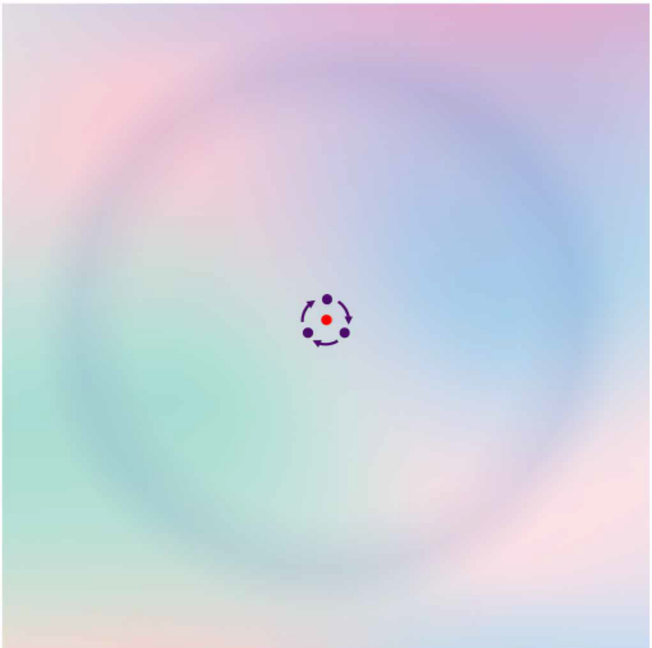


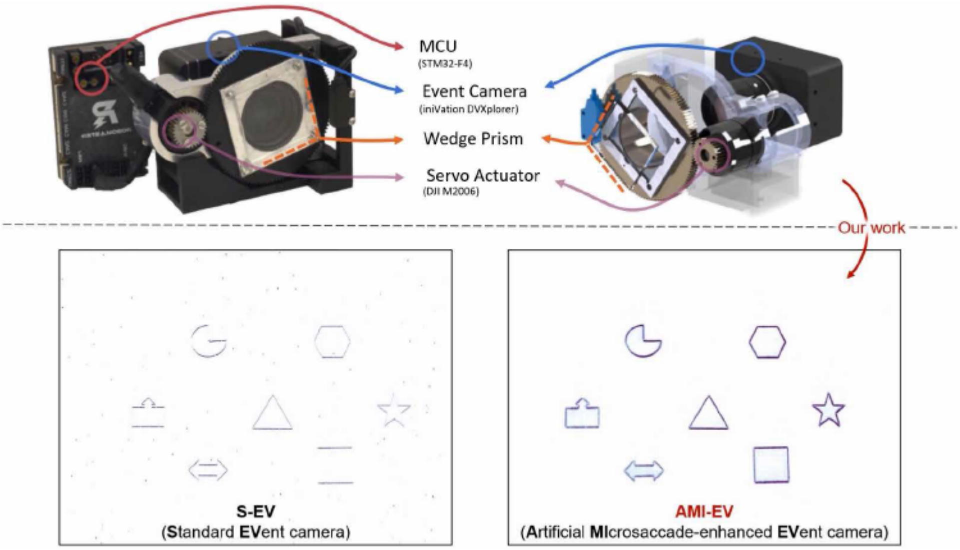
Fig. 1. Demonstration of how microsaccades counteract visual fading. A simple yet intuitive example demonstrating visual fading and how microsaccades counteract it. We recommend enlarging the image to at least 15 cm by 15 cm and keeping one's eyes 40 cm away from the screen. After a few seconds of fixation on the red spot, the bluish annulus and the background will fade. This is because microsaccades are suppressed during this time, and, therefore, the eye cannot provide effective visual stimulation to prevent peripheral fading. On the other hand, when saccading between the purple spots, the annulus is always experienced, possibly fading slower even though the saccades are small, typically 0.5° to 1.0° depending on the viewer's distance from the figure.

In recent years, people have tried to address this problem via active vision approaches. Several studies have integrated event cameras with other active sensors, such as structured light or lasers (*27–31*), to facilitate motion-independent event sensing. These studies introduce specialized sensor configurations that demonstrate impressive results in tasks like depth estimation, 3D reconstruction, and surface normal estimation, but the unique setups limit their adaptability to diverse applications. Moreover, these configurations tend to be more susceptible to specific illumination conditions and material types, constraining their broader utility. Some previous works emulated the human microsaccade mechanism by introducing additional motion into the event camera system (*32*, *33*). By shaking the event camera and introducing movements in different directions using a pan-tilt mechanism, saccade-like motions were introduced, and more information (events) could be recorded from multiple saccades. However, discrete sensor movements are difficult to implement in robotics systems. This is because of the substantial inertia of the electronic perception system; achieving high-frequency vibrations necessitates considerable torque, which is challenging to accomplish using currently available lightweight actuators. Therefore, to effectively address the issue of fading, alternative approaches inspired by nature, rather than strictly mimicking it, are required.

Our aim is to develop a similar artificial microsaccade (AMI) mechanism that varies the direction between the scene texture and the image motion. Although this can be done with saccades, it can also be achieved by manipulating the direction of the incoming light. Moreover, if the direction of the incoming light can be steered continuously rather than in discrete steps, the efficiency will also be improved. This is the basic idea that we use to design our system that will "see" events at all edges of the scene and will not miss any because of its motion.

Proposed solution

This paper identifies and resolves fundamental challenges to achieving accurate and stable event-driven data association from the perspective of hardware-software joint design. Instead of simply replicating nature, we propose a nature-inspired but more effective solution that uses an AMI mechanism to manipulate the direction of the incoming light, named AMI-enhanced event camera (AMI-EV). The AMI-EV actively senses visual information using a rotating wedge prism in front of an event camera. By actively triggering events in areas of high spatial frequency, such as edges, AMI-EV maintains the appearance of texture and high informational output, even when the sensor does not move. Figure 2A illustrates the hardware, Fig. 2B the refraction of the wedge mechanism, and Fig. 2C the imaging. Details of the rotating wedge-prism mechanism and the compensation algorithm are described in Materials and Methods and in movie S1. The compensation algorithm makes our system a plug-in-and-use solution with existing event-based perception



Movie 1. Demonstration of microsaccades and overview of the proposed system.

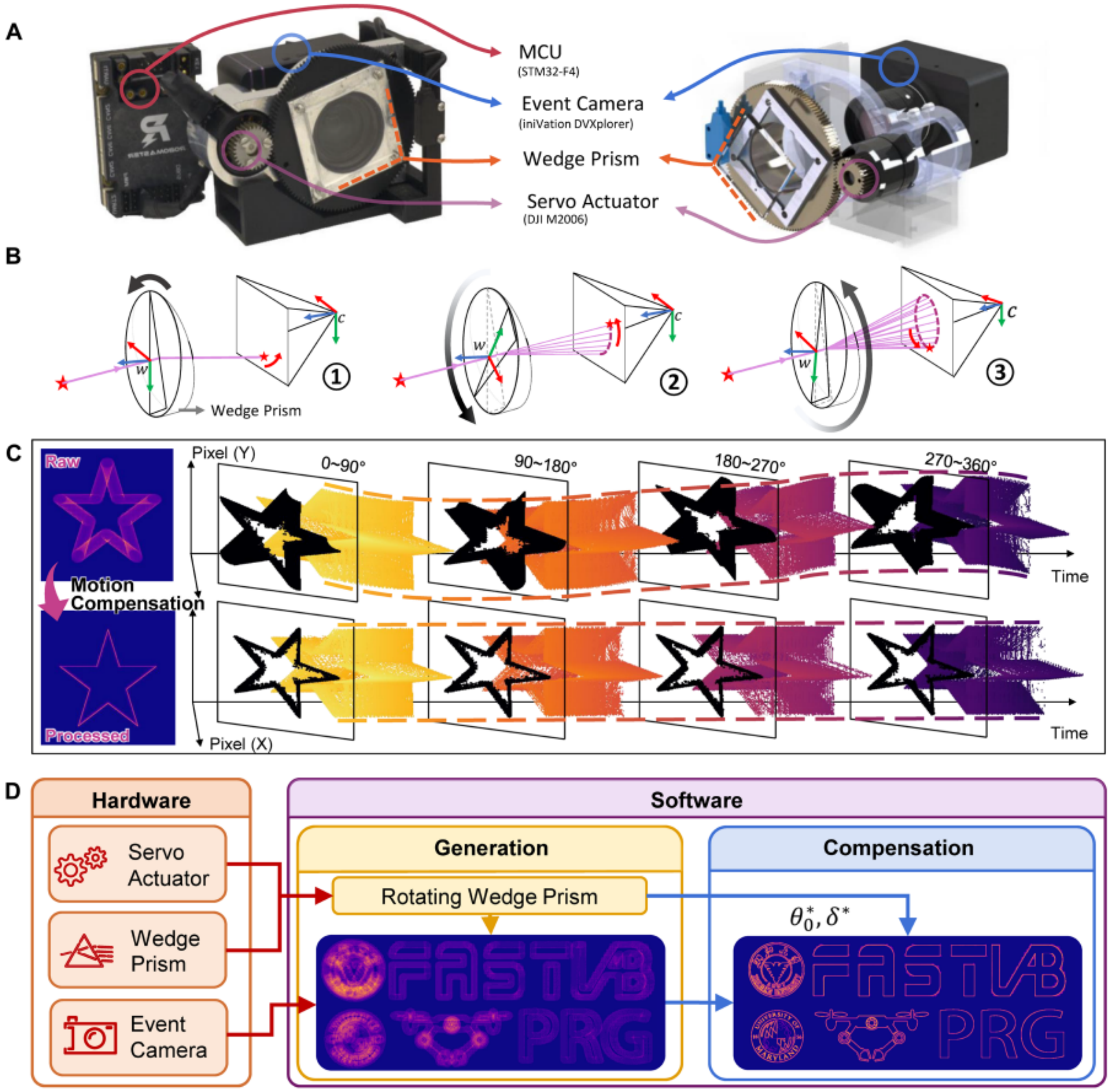


Fig. 2. Overview of our entire system, including both hardware and software. (**A**) Real-world hardware and computer-aided design (CAD) model. (**B**) Illustration of the incoming light refraction as the wedge prism rotates. (**C**) Event generation and compensation process, with the images on the left resulting from accumulating the event streams shown on the right. (**D**) System overview.

algorithms. We validated the potential of the system by applying it to many different low- and high-level vision tasks as detailed in Results. To facilitate future research, we also released the hardware design, the software for AMI generation, calibration and compensation, a simulation platform, and a translator for interfacing with public event camera datasets. With these tools (*34*), developers can generate their own AMI-EV datasets for their specific tasks from simulation, existing event-based vision datasets, and real-world environments.

RESULTS

In this section, we present the design of our AMI mechanism and then demonstrate its advantages due to its capability of maintaining stable and high informational output. To demonstrate the system's potential in facilitating robotics perception research, we evaluated it on various state-of-the-art event-based algorithms in several typical applications. The results verify that the proposed system is highly effective in improving performance across the board.

AMI generation and compensation

To generate events on all edges, we used the working principle of the wedge-prism deflector (*35*). When the prism rotates, it actively adjusts the direction of the incoming light, as illustrated in Fig. 2B. At the beginning of the procedure, the wedge prism has a certain orientation and deflects the incoming light at a fixed angle, as shown in Fig. 2B(1). Then, the actuator module drives the optical deflector module to rotate along the z axis of the camera, z_c, to make the incoming light constantly change its deflection, as shown in Fig. 2B(2). This enables the incoming light to continually generate events as it creates motion on the image plane with a circle-like trajectory, as shown in Fig. 2B(3). As a result, continuously changing rotational motion is induced in the camera. Because the AMI is in all directions in the image plane, the output event stream contains all boundary information of the scene, as shown in Fig. 2 (C and D). Compared with previous works (*32*, *33*) that move the camera instead of the prism, the moving parts of our system do not contain fragile components such as the camera, thus rendering it more robust for high-speed rotation. Moreover, our system operates under constant-speed rotation, which is a smoother motion than the vibrational motion considered in (*32*, *33*). A discussion of the optimal refraction angle and frequency of the rotations for different tasks is available in Materials and Methods.

Another important part of the proposed software framework is the AMI compensation. This is one of the major advantages of our approach compared with previous works (*32*, *33*), which inevitably suffer from motion blur and decreased accuracy. Looking at an image created by binning the events over a small time interval, which we call an accumulated event image (see Fig. 2C), blurred boundaries are observed in the absence of motion compensation. To obtain sharp edges, events triggered by the same incoming light ray direction must be moved to the same pixel. This requires calibrating the wedge orientation and compensating for the spatial displacement of the events introduced by the wedge motion. Given that our actuator system is equipped with an absolute position sensor (rotary encoder), the compensation parameters only need to be calibrated once and can be used directly for subsequent recordings. The technical details of the calibration and compensation algorithms are provided in Materials and Methods. The compensation is illustrated in the second row of Fig. 2 (C and D), and movie S1 shows the procedure.

Quantitative evaluation of texture enhancement

Experiment setup

To verify the effectiveness of the proposed system in texture enhancement, we conducted experiments on three representations: event stream, accumulated event images, and reconstructed intensity images. In each experiment, the performance of our system was tested against a standard event camera (S-EV). For all cases, two motion scenarios were considered: (i) no motion and (ii) motion with six degrees of freedom. All data were collected using the customized platform shown in fig. S1. The platform was equipped with an S-EV, an AMI-EV, and an Intel Realsense D435 camera (*36*) that provides red-green-blue, grayscale, and depth images. The hardware framework refers to the design of (*37*). Further configuration details can be found in Supplementary Methods.

Event stream

The event stream is a fundamental representation of event data from which all other event representations are derived. Therefore, enhancing the quality of the event stream can substantially improve the performance of a robotic perception system. In this experiment, we aimed to demonstrate that our system can generate an event stream of higher quality, containing more environmental information, than the S-EV. The quality was evaluated using the point distribution, a common metric for evaluating the quality of 3D point clouds. Previous works on spatial point cloud processing (*38*, *39*) have shown that a uniform distribution of points across the environment surface is preferable, because it indicates that the point cloud has captured all the necessary data. In the case of a spatial-temporal point cloud or an event stream, the point distribution is determined by both the scene structure and the motion. However, the same metric can still be used if we apply constraints on the scene. If the scene is static and all edges have the same illumination change, the point distribution is determined only by camera motion. In this scenario, a narrower distribution means that there is a higher proportion of events that share similar density. This results in a more uniform event density across the event stream, thus leading to a more stable representation of scene features that is less affected by camera motion. Therefore, the uniformity of the event stream can measure the influence of camera motion on the output.

In our experiment, we used kernel density estimation (KDE) to compute the density of events at their locations. The variance of the KDE distribution serves as an indicator of the uniformity of the event density at the location of the events. A lower variance suggests that a greater number of events share similar density, leading to a more stable representation of scene features. The experiment environment contained edges oriented in various directions, with an even spatial distribution throughout. Figure 3D illustrates that AMI-EV produced a more uniform point distribution than S-EV, with a variance of 0.196 compared with 0.425 for S-EV. This indicated that the output event stream of AMI-EV was more stable. In addition, as detailed in Supplementary Methods, the AMI-EV data had a lower ratio of low-density components, which are more likely due to noise and provide little useful structural information (fig. S10).

Accumulated event image

The accumulated event image is the most commonly used visualization in event-based vision tasks. Thus, enhancing its quality will substantially improve subsequent applications that process event data in a manner akin to image processing. In this study, we showed that the accumulated event images produced by our system exhibited superior stability and displayed less dependence on camera motion.

For this experiment, we first extracted the edges in the grayscale images using the Canny edge detector (*40*). Because the motion was small and the illumination was stable, we used them as ground truth for the edges in the environment. Next, image registration was applied to align the images among the S-EV, AMI-EV, and ground truth. Last, we measured the performance of capturing edges using two metrics: optimal dataset scale F1 (ODS-F) score and entropy, as shown in Fig. 3 (A, B, and E). ODS-F score is a commonly used metric for edge detection tasks (*41*, *42*), whereas entropy is a widely used parameter to quantify the amount of information present in an image. Both metrics were positively correlated with texture completeness in our experiments. Referring to the figure, AMI-EV showed stable and complete recordings of edges when the camera was in motion. Furthermore, we see that the output from the AMI-EV showed less dependence on camera motion than that of the S-EV.

In Fig. 3 (A and B), our system demonstrated higher and more consistent ODS-F scores, which can be attributed to the AMI mechanism. In certain motion patterns, such as the second snapshot in Fig. 3B, where the movement is parallel to most of the edges in the environment, the recordings from the S-EV can be greatly affected, whereas our system remains stable. Moreover, as shown in Fig. 3B, our system produced substantial improvements in the image entropy metrics compared with S-EV, indicating that it more effectively recorded complete edge information. The entropy was calculated on the binarized event map, and only the most representative part of the result is displayed here. For more detailed results, the reader is referred to fig. S8 in Supplementary Methods.

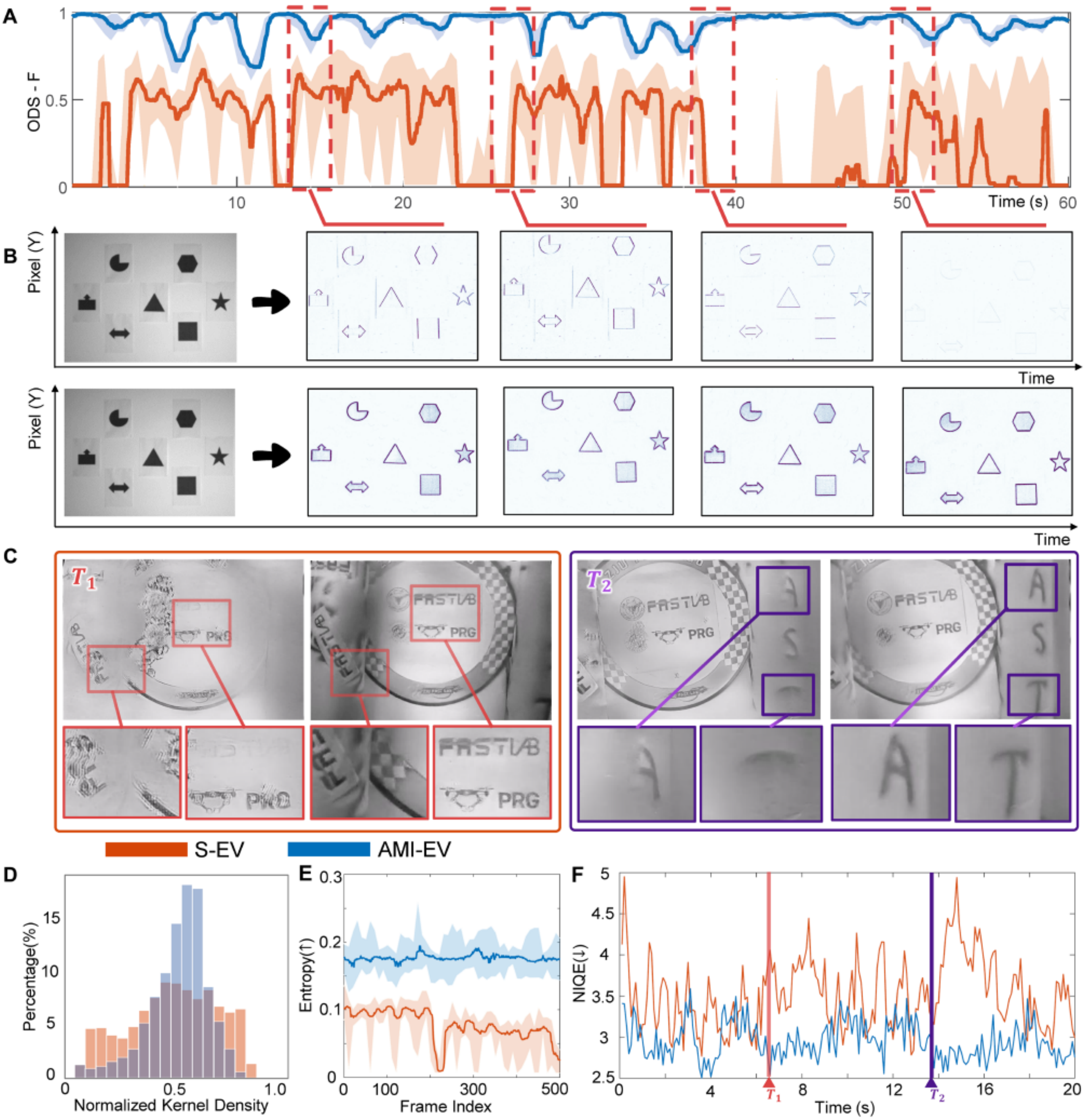


Fig. 3. Illustration of our approach's improvement on texture enhancement. (**A**) The ODS-F (higher is better) is used to measure the structural completeness of the accumulated event images. (**B**) Temporal snapshots of (A). (**C**) Comparison of the reconstructed grayscale images. (C) is the snapshots of (F), the color red for the box is used to indicate that the system is static, and purple denotes that the system is moving upward (along *y* axis). (**D**) Histogram of event density distribution for the original event stream and our enhanced event stream. More detailed illustrations can be found in fig. S10. (**E**) Entropy comparison of accumulated event images. In (A) and (E), solid curves indicate the median value over a time window of 10 data points. In contrast, the top and bottom bounds of the transparent regions indicate their maximum and minimum values. (**F**) Quantitative comparison of the reconstructed image quality using the NIQE (lower is better) (*47*).

Reconstructed intensity image

The enhancement of reconstructed intensity image quality is critical for event-based robot vision because such representation is essential in tasks like high–frame rate video generation (*43*, *44*). In the experiment, we first reconstructed videos using the event cameras at 1000 frames per second (fps), which is a typical frame rate used in high-speed imaging (*45*, *46*). Then, we used the natural image quality evaluator (NIQE) (*47*), which intuitively assesses how natural an image is to quantitatively evaluate the image quality.

The results are shown in Fig. 3 (C and F). Figure 3F shows the NIQE metric computed over a time interval with two time instances (T_1 and T_2) highlighted, as shown in the two snapshots in Fig. 3C. At T_1, the system is static, and at T_2, it is moving. We see that both cameras show satisfying image reconstruction performance when the robot is moving (right side of Fig. 3C). The proposed method achieved better performance because it can provide more information in regions that lack camera motion, such as horizontal edges when the robot is turning. When the robot is static, the performance of the S-EV decreases because of perceptual fading, as shown in the left side of Fig. 3C. More details about perceptual fading can be found in the perceptual fading effect in event cameras section in Supplementary Methods. On the other hand, the AMI mechanism effectively addresses the perceptual fading problem by actively providing more environmental information. Readers can refer to movie S2 for a more detailed illustration. In rare scenarios, the motion of the prism negates the optical flow induced by the motion of the camera, resulting in few events. In such scenarios, AMI-EV's performance degrades by a small margin. For example, at the 48th second in movie S2, there is a frame where the phenomenon of perceptual fading occurs, especially noticeable at the location of the "FAST Lab" logo on the image.

Feature detection and matching

The following experiments demonstrate the performance of the proposed system for feature detection and matching. These are the most representative tasks in low-level vision and the basic building blocks for various robotics applications. Event-based feature detection and matching are attracting increasing interest (*15*, *16*, *48*) because of the sensor's advantages of high dynamic range (HDR) and high temporal resolution. However, the performance of existing methods depends on the camera motion. The proposed system delivers high-quality features independent of camera motion and retains the benefits of event cameras. Movie S3 shows the experiments.

The environments used in the experiments are shown in Fig. 4A. We used four typical scenarios: a structured environment, an unstructured environment, a challenging illumination environment, and a dynamic environment. The first three scenarios were used for corner feature detection and tracking, and the last was used for motion feature detection and matching, also known as motion segmentation. For all experiments, we compared the proposed system with grayscale cameras and S-EV. We directly extracted features from the asynchronous event stream without any accumulation, preserving the high temporal resolution (in the order of microseconds) of the data. In these experiments, the wedge angle was set to 1.0°, and the rotating frequency was 12 Hz, which was sufficient to allow for motion compensation at the speed used, as shown in movie S3 (see the analysis in Materials and Methods).

Corner detection and tracking

We used the three experimental environments shown in Fig. 4A. After AMI generation, the corner events were extracted using a widely used event-based corner detector (*15*). Next, the extracted features are compensated to eliminate the effect of the wedge rotation. Figure 4B shows that our system detected and tracked more corner features and provided more information than S-EV in all three scenarios. The texture in S-EV became unstable because of changing motion, resulting in incomplete corner detection and unstable tracking. Furthermore, our system, along with S-EV, outperformed the standard camera in challenging illumination scenarios because of the event sensor's HDR, as shown in Fig. 4B(3). The quantitative results presented in Fig. 4B(4) demonstrated that our system achieved a substantially longer tracking lifetime than S-EV, although at the cost of slightly reduced accuracy (~1.5 pixels). The error in accuracy is primarily due to numerical computations and imperfect clock synchronization introduced during AMI compensation. Therefore, the error is independent of the camera's motion. For a more detailed analysis of this error, refer to the choice of deflector angle and rotating speed section.

Moreover, our system and the S-EV had a notably higher update rate than standard cameras, which is crucial in high-dynamic scenarios, as shown in Fig. 4B(5). In conclusion, our system was the only camera system that robustly detected and tracked corner features in all three typical scenarios. The results demonstrated that it effectively solved the corner detection and tracking tasks, especially in challenging illumination scenarios.

Motion segmentation

The event camera is well suited for segmenting fast-moving objects, and there is already a wide range of applications, including dynamic obstacle avoidance (*5*, *6*, *49*) and high-speed counting (*50*, *51*). This experiment aimed to demonstrate that our system and S-EV have a better performance than standard cameras for this task and that the additionally introduced motion in our system does not affect the performance.

The goal of the experiment was to segment independently moving objects from the background. In the experiment, the camera introduced motion in the background while a separately thrown ball moved independently. For motion segmentation on S-EV and AMI-EV, we adapted the methods from (*52*) and (*5*), which can provide per-event segmentation. Specifically, we used the idea of camera-motion compensation (*12*, *53*) by maximizing the sharpness of motion-compensated images and detecting moving objects as non-sharp regions using clustering techniques. For the standard camera, we applied one of the state-of-the-art methods (*54*) as our benchmark, which detects fast-moving objects as a truncated distance function to the trajectory by learning from synthetic data.

Comparing results from S-EV and AMI-EV in Fig. 4C, we see that the introduced motion did not influence the accuracy and robustness of the proposed system in motion segmentation tasks. However, the standard camera suffered from motion blur and low temporal resolution and could effectively capture the motion information, thus resulting in poor performance. More details can be seen in movie S3.

Human detection and pose estimation

This experiment demonstrated the potential of applying the proposed system in a popular high-level vision problem, human detection, and pose estimation. Event cameras are particularly well suited for detection tasks that involve fast motion and have attracted interest in recent years (*55*–*57*). However, previous methods need either the assistance of grayscale images to update the detection (*55*) or the initialization of the pose estimation (*56*). Moreover, they do not apply to dynamic environments where the camera moves. In this experiment, we demonstrated the potential of the proposed AMI mechanism in achieving robust high-speed human motion estimation. To obtain better texture and intensity information, we used images reconstructed from events as the event representation, which have been proven to be robust in different scenarios, including the dynamic one (*43*, *58*, *59*). We used one of the most popular human pose estimation algorithms, called OpenPifPaf (*60*), to conduct human detection and pose estimation.

We evaluated accuracy and robustness using intersection over union (IoU) and percentage of detected joints (PDJ). These evaluations

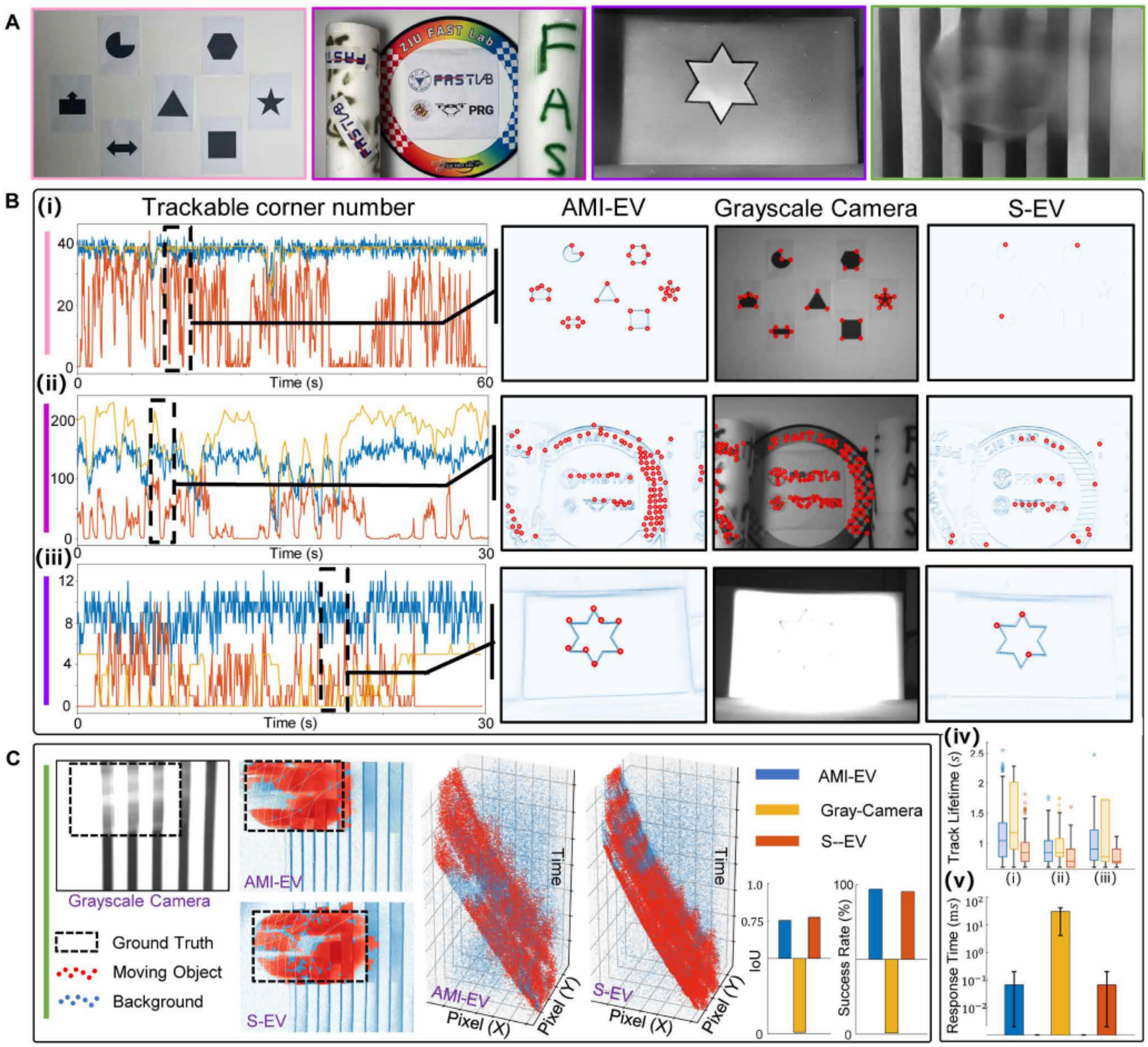


Fig. 4. Evaluation of feature detection and matching. (**A**) Environment setups of four experiments. (**B**) Results of the corner detection and tracking experiments. The left column of (i) to (iii) provides a comparison of the number of trackable corners, and the three right columns show snapshots. (iv) and (v) are metric comparisons visualized using box and bar graphs. (iv) indicates the lifetime of all trackable corners, and (v) shows the response time. (**C**) Results of the motion segmentation experiment. Blue parts indicate the background, and red parts indicate independently moving objects.

were made in relation to the video frame rate, which denotes the fps that the stand event-to-video algorithm, E2VID (*43*), can generate. As shown in Fig. 5, the AMI-EV demonstrated better performance at different frame rates. When using our system, the frame rate can be configured to be substantially higher than S-EV while maintaining image quality. More details can be found in movie S4.

AMI-EV simulator and translator

AMI-EV simulator

To facilitate future research, we also developed a simulator. The code was released in (*34*). The simulator was based on our previous work, WorldGen (*61*), which allows the generation of 3D photorealistic scenes with the user having control of features, like the scene texture and the camera and lens properties. The simulator allows the user to generate a task-specific synthetic AMI-EV. Figure 6A illustrates an example of a scene created for human pose estimation. The simulator provided the synthetic AMI-EV data along with a list of visual representations of the scene. See Supplementary Methods for more details on the simulator.

AMI-EV translator

In addition to the simulator, we also provided a translator to create a synthetic AMI-EV from standard datasets. The proposed translator supports three types of inputs: grayscale images, grayscale images combined with events, or events only. With appropriate video

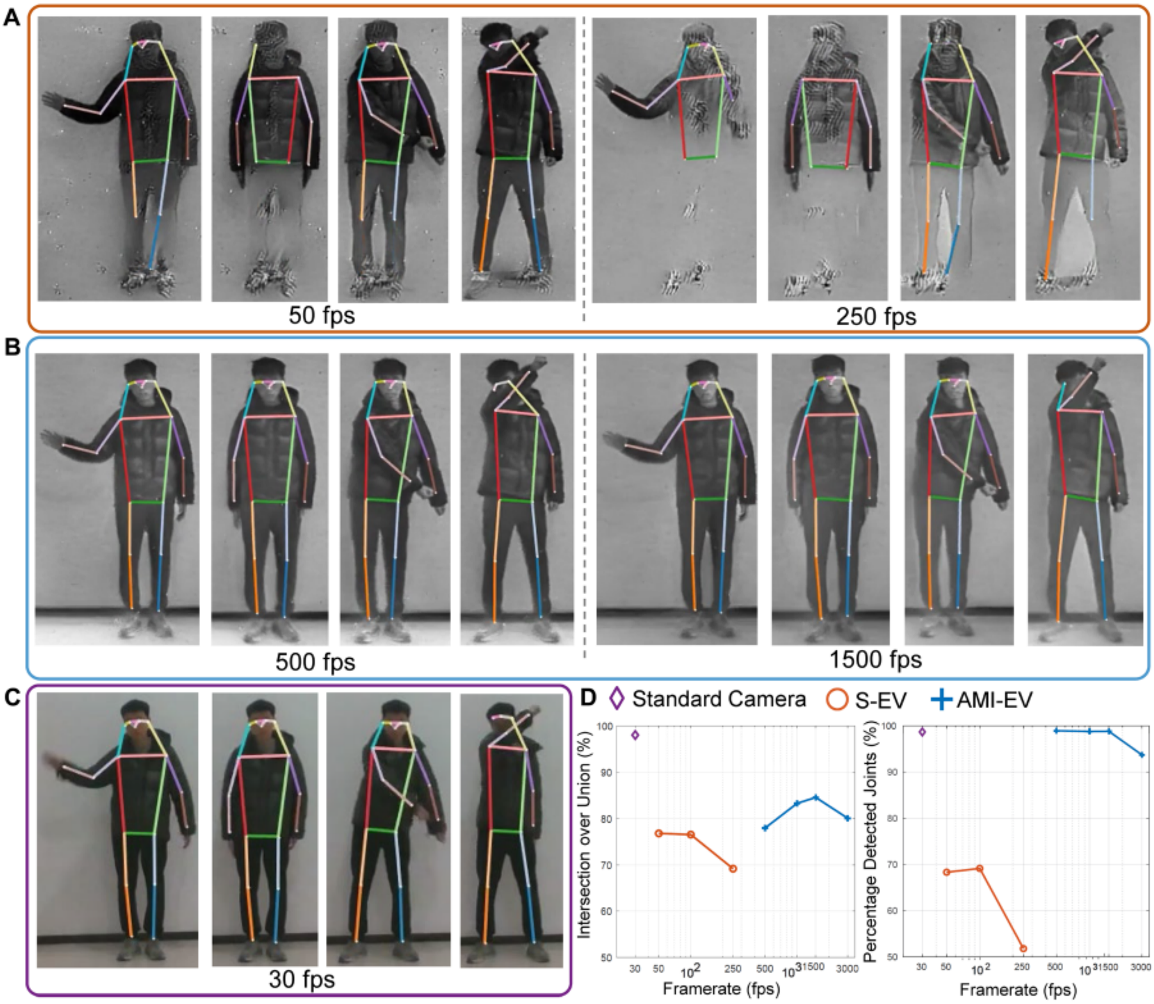


Fig. 5. Evaluation of human detection and pose estimation. (**A** to **C**) Results of human pose estimation for S-EV (A), AMI-EV (B), and a standard camera (C) on four actions: waving the hand, shaking arms, baseball batting action, and ping-pong batting. The former two actions are slow, and the latter two are fast, which caused motion blur in RGB frames. (**D**) Metric comparisons. The frame rate denotes the fps that E2VID (*43*) is configured to generate. IoU provides a measure of human detection performance, and PDJ is a measure of the detected joints' localization precision and completeness. Because the sampling frame rate varies greatly from different sensors, we use the semilog plot (*x* axis has log scale) to visualize the data.

interpolation algorithms, high–frame rate videos can be generated. Subsequently, these high–frame rate videos are fed into a specially designed AMI module to produce the output AMI event stream. To understand the working principle of the AMI-EV translator in detail, refer to Supplementary Methods and fig. S5. Figure 6B shows translation examples from two typical event-based datasets, called Neuromorphic-Caltech101 (*32*) and Multi Vehicle Stereo Event Camera Dataset (*62*), which are both widely used for evaluating event-based 3D perception and recognition tasks. Further results can be found in Supplementary Methods.

DISCUSSION

By emulating the biological microsaccade mechanism, a texture-enhancing event vision system that enables high-quality data association has been proposed and evaluated. Stable texture appearance and high informational output are maintained with our system consisting of a rotating wedge filter in front of an event camera. We also provided a compensation algorithm to account for the motion from the wedge filter. Our results show that the output of the compensation is compatible with most representative event-based data–processing methods with minimal loss of accuracy and latency. For low-level tasks, there is a margin of error introduced by this compensation, in particular in specialized tasks like optical flow estimation over short time intervals. As shown in fig. S11, the performance of optical flow estimation to static objects degrades by 0.19 pixel of end point error. However, the benefits of preserving stable texture generally outweigh this drawback. For high-level tasks, the effect of this loss is negligible given that it does not compromise the performance of advanced recognition or detection tasks.

We demonstrated experimentally that our device can acquire more environmental information than traditional S-EVs. It can maintain a high-informational output while preserving the advantages of event cameras, such as HDR and high temporal resolution. Extensive validation experiments demonstrated that our system has potential for use in various field robotics applications ranging from low-level to high-level vision tasks. It can achieve better feature extraction in low-level vision tasks and help the robot recognize and understand the environment better in high-level vision tasks.

In summary, our proposed system fundamentally eliminates the motion dependency problem in event-based vision using a bioinspired mechanism. This hardware improvement enables our system to easily achieve high-quality data output compared with S-EVs. Furthermore, the proposed software allows our system to be used for elaborate mission-specific requirements.

Future work

As shown before, the proposed hardware device and software solution allow better data association for event-based vision. However, the system is less energy efficient than an S-EV because of the additional mechanical structure. In addition, the different data format also calls for additional data-processing methods.

To make the hardware more energy efficient, future research will need to improve the AMI-generating mechanism both in the hardware and software. Most actuators of this size consume energy from watts to a few tens of watts, which is higher than the S-EV. To achieve less power consumption, one could replace the mechanical structure with electro-optic materials and control the incoming light direction by optic phase array (*63*) technology. Specifically, by dynamically controlling the optical properties of electro-optic materials like liquid crystal display (*64*), the direction of the incoming light can be steered. Such approaches can achieve a control frequency of more than 5 kHz by micro-electromechanical system (*65*) while maintaining low power consumption, which has been validated and applied in the computational imaging field (*65*, *66*). Another possible solution is to optimize the rotation speed and adapt it to the specific scenario. The effect of the added AMI motion decreases with faster scene motion. High-speed rotation is more effective for low-dynamic scenarios; thus, its use could be adapted to the speed. For certain tasks, the system could operate at low speed or even stop once an adequate amount of data has been collected for analysis or increase its rotational speed in response to diminishing texture. However, designing specific action strategies for different application scenarios remains a challenge.

The proposed device also creates an event data format where a periodic motion is encoded into the event stream. This raises a question: Is there a more efficient and effective way to process the new data than compensating for it? In this work, the compensation algorithm removes the added motion from the output stream to make it compatible with existing event-based algorithms. However, this method also introduces some discretization errors and adds computational costs. Although the error (around 1.5 to 2.0 pixels) is acceptable for most robotics applications and the system can still work in real time on onboard computers, the additional error and computation may be problematic in applications where precise measurements are needed or for small robots with limited computation resources. Moreover, the current compensation procedure amalgamates the events of both polarities and thus loses the polarity features. Future work can consider a more complex fitting model, such as an oriented ellipse, instead of a circular motion to further decrease the compensation error. To eliminate the compensation error fundamentally, we may need a method that can work directly on the generated event stream and use the motion information without moving the pixel locations. We will also investigate training a neural network to regress the accurate pixel-wise compensations function. In the spirit of event-based work, we could train a spiking neural network (SNN). We believe that the best way would be to train the network as a regressor using the method of conversion; for example, we first trained an artificial neural network and then converted this network into an SNN (*67*, *68*).

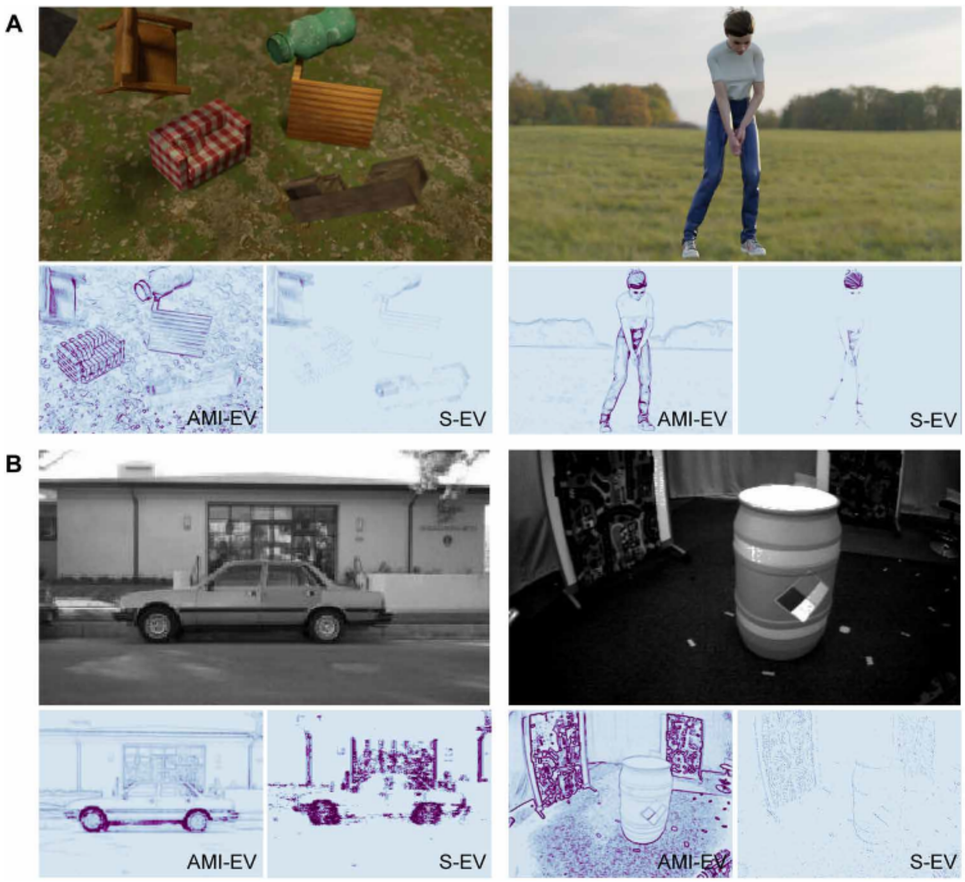


Fig. 6. Pictures generated by the released software package. (**A**) (Left) 3D-rendered scene with multiple moving objects; (right) golf scene. (**B**) Output of the released translator. (Left) Image from the Neuromorphic-Caltech 101 dataset and two event count images generated from an S-EV and an AMI-EV, respectively; (right) scene from Multi Vehicle Stereo Event Camera Dataset (*75*).

MATERIALS AND METHODS

Hardware architecture

The proposed hardware platform is 82 mm by 54 mm by 62 mm. Its total weight is 322 g, including a 131-g event camera (with lens) and a 41-g external microcomputing unit (MCU) for actuator control. Our system comprises four modules: the optical deflector module, the actuator module, the event camera module, and the MCU, as shown in Fig. 2A. The blueprints have been released (*34*) to benefit future research.

For the optical deflector module, a wedge prism was mounted in front of the camera lens to deflect the incoming light at a fixed angle from x_w, the x axis in a coordinate frame w attached to the wedge prism (shown in Fig. 2B). The actuator module drove the optical deflector module rotating around z_c, the z axis of the coordinate frame attached to the camera frame c, also shown in Fig. 2B. Our platform uses a DJI M2006 Brushless DC Motor (*69*) with a customized reduction gear and an absolute position encoder. With the modified gear module, the actuator weighted 57 g (including the electronic speed controller) and provided 0.11-N·m torque at 1500 rpm, which satisfied our rotation speed and torque requirements. Moreover, by adding a photoelectric sensor to sense the prism's orientation, the motor's incremental encoder signal could be transformed to get an absolute orientation measurement as needed for the AMI calibration. For the camera module, we adopted the DVXplorer event camera (*70*). It has a spatial resolution of 640 pixels by 480 pixels and supports time synchronization with external sensors. The microcontroller unit was used to control the actuator's motion, receive position feedback, and synchronize the event camera with the actuator. We used the DJI Robomaster Development Board (*71*), whose weight is 40 g, to simplify the development.

Choice of deflector angle and rotation speed

In this section, we experimentally evaluate the influence of the rotation speed and prism angle on the data volume and compensation accuracy, and subsequently, we discuss good choices for different tasks. As demonstrated in Fig. 7, increasing the degree of tilted angle of the wedge prism and rotation speed led to a larger number of events but also higher motion compensation errors.

Two factors governed the selection of rotational speed: the duration of the maintained time window and the compensation error. Considering the former, the data must originate from at least a quarter of the rotation period, because this is the smallest unit containing a pair of orthogonal motions necessary for activating edges in all directions and thereby ensuring texture stability. For instance, an event count image typically comprises data spanning a 33-ms duration. Consequently, one rotation period should last $33 \times 4 = 132$ ms (455 rpm) to guarantee the inclusion of all environmental information within a single frame. In practice, the rotational speed must surpass this minimum requirement to counteract the influence of sensor noise.

The second issue was the compensation error. As illustrated in the left subfigure of Fig. 7B, the error, represented by the SD of the event distribution, surged markedly beyond 720 rpm for 0.5° and 1.0° prisms. This escalating influence can be attributed to small synchronization errors among the sensors, which amplify as the rotational speed increases. Furthermore, this effect bears a connection to the deflector angles. In light of the above analysis, the rotational speed was set to 720 rpm for all real-world experiments to achieve a balance between texture stability and compensation accuracy.

The selection of the deflector angle was task specific. As shown in Fig. 7A, the geometric structure was similar across the output of all three tested prisms, with the primary differences being data volume and compensation accuracy. For tasks that prioritize data intensity, such as corner detection and tracking, a larger tilt angle was preferable, provided that accuracy was maintained. This is because a prism with a larger tilt angle can generate more events in a given time, as shown in Fig. 7B, and these events are mostly found in areas with rich texture features, such as corner points. This leads to increased robustness in such tasks. Conversely, a smaller tilt angle was more suitable for tasks emphasizing contour completeness and compensation accuracy as long as data sufficiency was ensured. For instance, in tasks like human pose estimation or semantic segmentation, the completeness and sharpness of object boundaries are more critical than the data intensity. According to the left subfigure of Fig. 7B, both the 0.5° and 1.0° prisms exhibited satisfactory compensation accuracy at a rotational speed of 720 rpm. In the right subfigure, the 1.0° prism displayed higher data intensity than the 0.5° one. The 2.0° prism, although it had the highest data intensity, had a compensation error too high to be practical. Therefore, in this work, we chose a 1.0° prism with a rotation speed of 720 rpm for the feature detection and matching experiments and a 0.5° prism with 720 rpm in all other experiments, as well as in the simulator and translator.

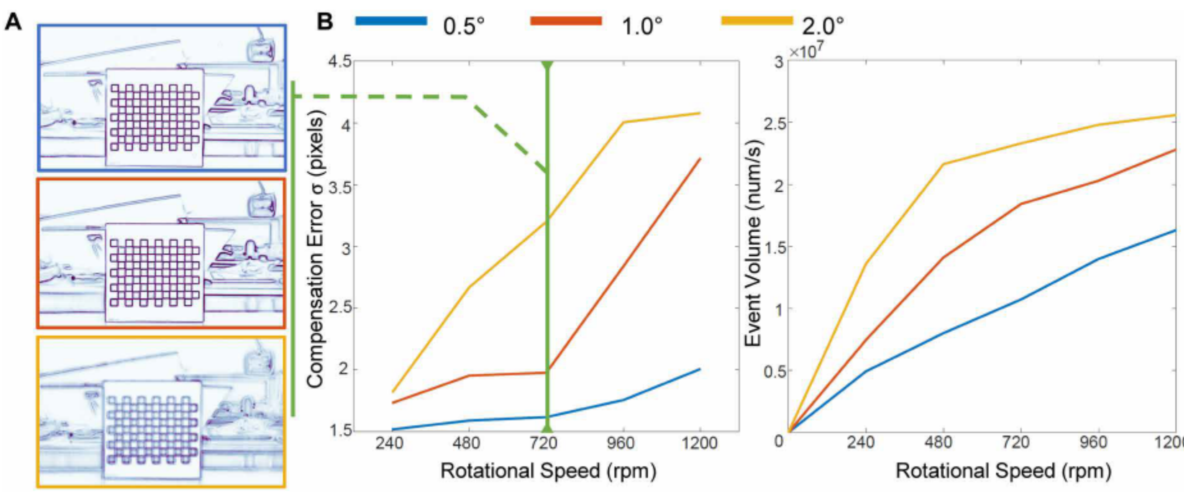


Fig. 7. Compensation error and data volume for different combinations of deflector angles and rotational speeds. (**A**) A snapshot of compensation performance with the rotational speed at 720 rpm. The colors on the image boundaries indicate the deflector angles. (**B**) Quantitative results. Details about how to calculate the compensation error can be found in Supplementary Methods. The event volume is measured by bandwidth analysis, as detailed in fig. S9.

2D wedge-prism camera model

Figure 8A illustrates the optical model with a 2D cross section of the wedge-prism camera model. The incoming light $v_{in} \in \mathbb{S}^1$ denoting a unit vector on the left was transmitted and deflected twice (v' and v_{out}) through the wedge prism and then focused on the camera image plane I at pixel p_i. According to Snell's law (*72*), the relationship between Φ_i and Φ_p can be described as

$$\begin{gathered}\sin\Phi_i = n \cdot \sin\Phi_p \\ \Phi_p = \arcsin\left(\frac{\sin\Phi_i}{n}\right)\end{gathered} \tag{1}$$

where Φ_i is the angle between $\mathbf{v_{in}}$ and $\mathbf{z_c}$ and Φ_p is the angle between $\mathbf{v_p}$ and $\mathbf{z_c}$, respectively. n is the refractive index of the prism material, which was set to 1.55 in the experiments. The refractive index of the air is regarded as 1.0. Therefore, vector $\mathbf{v_p}$ can be represented as

$$\mathbf{v_p} = R\left(\widehat{\mathbf{v_i} \times \mathbf{z}_c}, \Phi_i - \Phi_p\right) \cdot \mathbf{v_i} \tag{2}$$

where $R(a, b)$ denotes a rotation along axis a with an angle of b (counterclockwise as the positive direction) and $\hat{\mathbf{v}}$ denotes the normalized vector $\mathbf{v}$.

Similarly, the relationship between Φ_q (angle between $\mathbf{v_p}$ and $\mathbf{z_w}$) and Φ_o (angle between $\mathbf{v_o}$ and $\mathbf{z_w}$) can be written as

$$\Phi_o = \arcsin\left(n \cdot \sin\Phi_q\right) \tag{3}$$

where Φ_q can be expressed as

$$\Phi_q = \arcsin\left(\|\mathbf{v_p} \times \mathbf{z}_w\|\right) \tag{4}$$

Last, the output light vector $\mathbf{v_o}$ can be represented as

$$\mathbf{v_o} = R\left(\widehat{\mathbf{v_p} \times \mathbf{z}_w}, \Phi_q - \Phi_o\right) \cdot \mathbf{v_p} \tag{5}$$

Summarizing, $\mathbf{v_o}$ can be fully represented by $\mathbf{v_i}$ and $\mathbf{z}_w$. The transmission through the prism is described by a function $g\left(\mathbf{v_i}, z_c\right) \in \mathbb{S}^1$ as

$$g\left(\mathbf{v_i}, \mathbf{z}_w\right) = R\left(\widehat{\mathbf{v_p} \times \mathbf{z}_w}, \Phi_q - \Phi_o\right) \cdot R\left(\widehat{\mathbf{v_i} \times \mathbf{z}_c}, \Phi_i - \Phi_p\right) \tag{6}$$

Because $\mathbf{v_p}$, $\mathbf{z}_w$, $\mathbf{v_i}$, and $\mathbf{z_c}$ are in the same plane, $\widehat{\mathbf{v_p} \times \mathbf{z}_w}$, $\widehat{\mathbf{v_i} \times \mathbf{z_c}}$, and $\widehat{\mathbf{z}_w \times \mathbf{z_c}}$ are parallel to each other. According to the pinhole camera model (*73*), the angles between $\mathbf{v_p}$ and $\mathbf{z}_w$, $\mathbf{v_i}$, and $\mathbf{z_c}$ are larger than 90°, which means that $\widehat{\mathbf{v_i} \times \mathbf{z_c}} = \widehat{\mathbf{z_c} \times \mathbf{z}_w}$ and $\widehat{\mathbf{v_p} \times \mathbf{z}_w} = \widehat{\mathbf{z}_w \times \mathbf{z_c}}$. Therefore, $g(\mathbf{v_i}, z_c)$ can also be written as

$$\begin{aligned} g\left(\mathbf{v_i}, \mathbf{z_c}\right) &= R\left(\widehat{\mathbf{z}_w \times \mathbf{z_c}}, \Phi_q - \Phi_o\right) \cdot R\left(\widehat{\mathbf{z_c} \times \mathbf{z}_w}, \Phi_i - \Phi_p\right) \\ &= R\left(\widehat{\mathbf{z}_w \times \mathbf{z_c}}, \Phi_q - \Phi_o\right) \cdot R\left(\widehat{\mathbf{z}_w \times \mathbf{z_c}}, \Phi_p - \Phi_i\right) \\ &= R\left(\widehat{\mathbf{z}_w \times \mathbf{z_c}}, \Phi_q - \Phi_o + \Phi_p - \Phi_i\right) \\ &= R\left(\widehat{\mathbf{z}_w \times \mathbf{z_c}}, \delta\left(\mathbf{v_i}, \mathbf{z}_w\right)\right) \end{aligned} \tag{7}$$

where $\delta(\mathbf{v_i}, \mathbf{z}_w) = \Phi_q - \Phi_o + \Phi_p - \Phi_i$ because these variables are determined by $\mathbf{v_i}$ and $\mathbf{z}_w$ according to Snell's law (*72*).

Eventually, on the basis of the pinhole camera model (*73*), $\mathbf{v_o}$ can be projected to the image plane by the camera's intrinsic matrix K, and the wedge-prism camera model can be formulated as

$$p_i = \mathbf{K} \cdot g\left(\mathbf{v_i}, \mathbf{z}_w\right) \cdot \mathbf{v_i} \tag{8}$$

Rotating wedge-prism camera model

Building on the 2D wedge-prism camera model, we next explain the 3D rotating model. In Fig. 8B, the incoming light $\mathbf{v_i} \in \mathbb{S}^2$ is transmitted and deflected twice ($\mathbf{v_p} \in \mathbb{S}^2$ and $\mathbf{v_o} \in \mathbb{S}^2$) through the wedge prism and lastly focused by the lens on the image plane, at $I_{m,\,n}$, where m and n are the indexes of the image pixel.

The rotating wedge-prism camera model introduces a time-varying rotation, which adds a variable θ, as shown in Fig. 8B. Therefore, the transmission from $\mathbf{v_i}$ to $\mathbf{v_o}$ is defined as $G(\mathbf{v_i}, \mathbf{z}_w(\boldsymbol{\theta}))$ generalizing $g(\mathbf{v_i}, \mathbf{z}_w)$ in Eq. 6 with a parameter for time t added because $\mathbf{z}_w(\boldsymbol{\theta})$ is time varying. The transmission function $G(\mathbf{v_i}, \mathbf{z}_w(\boldsymbol{\theta}))$ can be expressed as

$$G\left(\mathbf{v_i}, \mathbf{z}_w(\boldsymbol{\theta})\right) = R\left(\mathbf{v_p} \times \mathbf{z}_w(\boldsymbol{\theta}), \Phi_q - \Phi_o\right) R\left(\mathbf{v_i} \times \mathbf{z_c}, \Phi_i - \Phi_p\right) \tag{9}$$

Thus, the transmission from $\mathbf{v_i}$ to $\mathbf{v_o}$ can be expressed as

$$\mathbf{v_o} = G\left(\mathbf{v_i}, \mathbf{z}_w(\boldsymbol{\theta})\right) \cdot \mathbf{v_i} \tag{10}$$

Last, $\mathbf{v_o}$ can be projected onto the image plane, and the camera's intrinsic matrix is denoted as K. The proposed rotating wedge-prism camera model can be formulated as

$$I(m, n) = \mathbf{K} \cdot G\left(\mathbf{v_i}, \mathbf{z}_w(\boldsymbol{\theta})\right) \cdot \mathbf{v_i} \tag{11}$$

Microsaccade model simplification

With the proposed optical model, the optical properties of our system can be precisely described. However, its accuracy is highly dependent on the spatial resolution of $\mathbf{v_i}$ and θ. The resolution is negatively related to the robustness of the calibration. For example, for a 640 pixel–by–480 pixel-resolution event camera, if the resolution θ is set as 1°, it needs 640 by 480 by 360 parameters to fully describe the model. If so, the calibration process needs a long time to collect enough data for each pixel, and any illumination change during the process will highly influence the results. If we down-sample the resolution, a discretization error will be introduced, resulting in poor compensation performance.

To make the parameter calibration more efficient in memory and computation, we simplified the model and reduced the number of parameters by applying an approximation. First, we decomposed $\mathbf{v_i}$ into two vectors $\mathbf{v}_\perp$ and $\mathbf{v}_\parallel$, where $\mathbf{v}_\perp$ is vertical to $\mathbf{z}_w(\boldsymbol{\theta}) \times \mathbf{z_c}$ and $\mathbf{v}_\parallel$ is parallel to $\mathbf{z}_w(\boldsymbol{\theta}) \times \mathbf{z_c}$. These two vectors can be expressed as

$$\begin{aligned} \mathbf{v}_\parallel &= \left(\mathbf{v_i} \cdot \widehat{\mathbf{z}_w(\boldsymbol{\theta}) \times \mathbf{z_c}}\right) \cdot \left(\widehat{\mathbf{z}_w(\boldsymbol{\theta}) \times \mathbf{z_c}}\right) \\ \mathbf{v}_\perp &= \mathbf{v_i} - \mathbf{v}_\parallel \end{aligned} \tag{12}$$

where $\widehat{\mathbf{z}_w(\boldsymbol{\theta}) \times \mathbf{z_c}}$ is the normalized unit vector of $\mathbf{z}_w(\boldsymbol{\theta}) \times \mathbf{z_c}$. Then, Eq. 10 can be written as

$$\begin{aligned} \mathbf{v_o} &= G\left(\mathbf{v}_\perp + \mathbf{v}_\parallel, \mathbf{z}_w(\boldsymbol{\theta})\right) \cdot \mathbf{v_i} \\ &\approx g\left(\widehat{\mathbf{v}_\perp}, \mathbf{z}_w(\boldsymbol{\theta})\right) \cdot \mathbf{v_i} \\ &= g\left(\widehat{\mathbf{v}_\perp}, \mathbf{z}_w(\boldsymbol{\theta})\right) \cdot \mathbf{v}_\perp + g\left(\widehat{\mathbf{v}_\perp}, \mathbf{z}_w(\boldsymbol{\theta})\right) \cdot \mathbf{v}_\parallel \\ &= \left(g\left(\widehat{\mathbf{v}_\perp}, \mathbf{z}_w(\boldsymbol{\theta})\right) \cdot \widehat{\mathbf{v}_\perp}\right) \cdot \|\mathbf{v}_\perp\| + \mathbf{v}_\parallel \end{aligned} \tag{13}$$

where from Eq. 7, we have that $g\left(\widehat{\mathbf{v}_\perp}, \mathbf{z}_w(\boldsymbol{\theta})\right) = R\left(\widehat{\mathbf{z_c} \times \mathbf{z}_w(\boldsymbol{\theta})}, \delta\left(\mathbf{v_i}, \mathbf{z}_w(\boldsymbol{\theta})\right)\right)$.

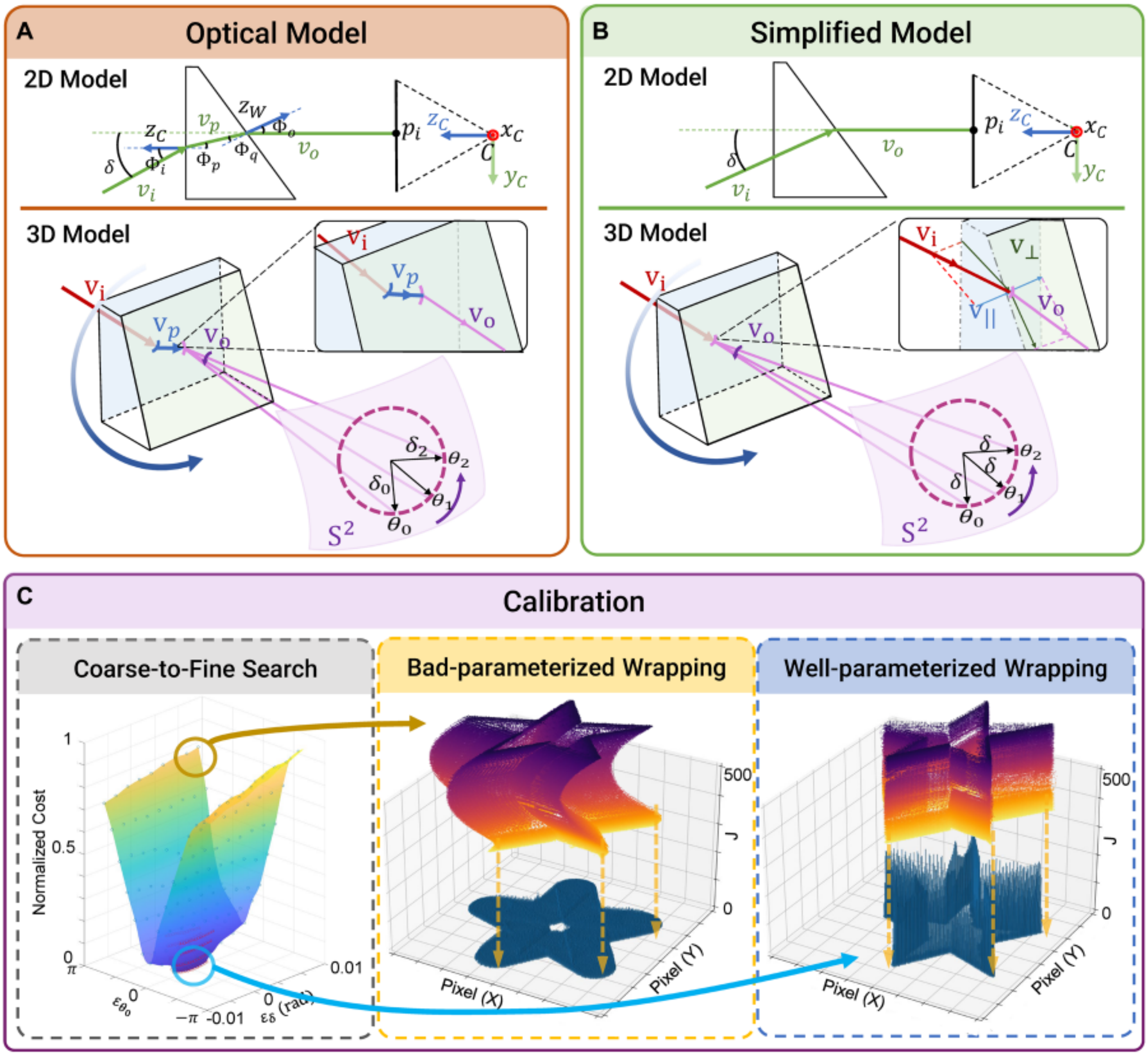


Fig. 8. Demonstration of the optical model, model simplification, and calibration procedures of our system. (**A**) The 2D wedge-prism camera optical model and the 3D rotating model of the proposed AMI-EV. (**B**) A simplified model of (A). (**C**) The calibration procedure. (Left) The coarse-to-fine search procedure, where blue points are samples from coarse search and red points are samples from fine search (bottom of the surface). A surface was fitted to the samples. (Center) Bad estimate of the actual trajectory, with a sharpness cost of 29,569. (Right) Good estimate of the actual trajectory, with a sharpness cost of 2382.

The trajectory of $\mathbf{v}_i$ over time is shown in Fig. 8B. It is close to, but not exactly, a circle in $SO(3)$. This is because the rotation axis $\mathbf{z}_c$ is not aligned with $\mathbf{z}_w(\boldsymbol{\theta})$, resulting in the change of $||\mathbf{v}_i \times \mathbf{z}_w(\boldsymbol{\theta})||$. Therefore, the radius $\delta(\mathbf{v}_i, \mathbf{z}_w(\boldsymbol{\theta}))$ also varies over time, and we denote the set of $\delta(\mathbf{v}_i, \mathbf{z}_w(\boldsymbol{\theta}))$ as $\Delta = \delta^i (i = 1,2\ldots)$.

Still, because Δ has hundreds of parameters, further simplification is needed. Thus, we defined a new frame w' that is fixed to the wedge prism and rotates with it. w' has the same origin as w, and their z axes $z_{w'}$ and z_w are aligned. Because $\mathbf{z}_{w'}$ is aligned with the rotational axis z_c, $||\mathbf{v}_i \times \mathbf{z}_w'||$ is constant. Now, $g\left(\widehat{\mathbf{v}_\perp}, \mathbf{z}_w(\boldsymbol{\theta})\right)$ can be represented as

$$
\begin{aligned}
g\left(\widehat{\mathbf{v}_\perp}, \mathbf{z}_w(\boldsymbol{\theta})\right) &= R\left(\widehat{\mathbf{z}_c \times \mathbf{z}_w(\boldsymbol{\theta})}, \delta\left(\mathbf{v}_i, \mathbf{z'}_w\right)\right) \\
&= R\left(\widehat{\mathbf{z}_c \times \mathbf{z}_w(\boldsymbol{\theta})}, \delta\left(\mathbf{v}_i, \mathbf{z}_c\right)\right)
\end{aligned} \tag{14}
$$

In this model, as shown in Fig. 8C, $\mathbf{z}_w(\boldsymbol{\theta}) = R(\mathbf{z}_c, \theta) \cdot R(\mathbf{X}_C, \alpha) \cdot \mathbf{z}_c$, and $\theta(t) = \theta_b + \tilde{\theta}(t)$, where θ_b denotes the bias of initial position between the actuator encoder and the circular trajectory and $\tilde{\theta}(t)$ denotes the angular measurement obtained from the encoder.

Through the above approximation, the trajectory of $\mathbf{v}_i$ is simplified to a circle $\odot\phi(\delta, \theta) \in SO(3)$, which brings two main advantages. First, it only has two parameters, δ and θ, for each pixel, which are easy to calculate, store, and optimize. Second, it is differentiable, which means that it does not lose accuracy because of discretization. Admittedly, this simplification also introduces some errors. Our analysis found that the error is within 2 pixels in a 90° field of view, which can be safely ignored. Details of the analysis are in Supplementary Methods.

Microsaccade calibration and compensation

The calibration procedure calibrates δ and θ_b in an optimization-based manner. The first step of our algorithm is to assign the initial values for δ and θ_b, denoted as δ^0 and θ_b^0. The choice of initial values was based on the hardware setup. δ^0 was set as the refraction angle of the wedge prism, and the zero-position of the encoder determined θ_b^0. In our procedure, we collected a batch of events $E = e_i(i = 1,2\ldots)$ and encoder data over some time $t - t = 2s$. In the experiment, we found that this achieved a good balance between calibration accuracy and computational cost. There was not enough information for shorter time windows, and sometimes it did not converge. Large time windows could result in a heavy burden on the computation because millions of events had to be processed in each iteration. Still, they did not increase the accuracy notably because, with 2 s, there were already 15 or more periods of rotation, which was sufficient for the computation. Next, we transferred the events from the spatial-temporal domain (x, y, t) to the (x, y, θ) domain by synchronizing the events' time stamp with the wedge prism's angular position. Then, we warped all events back to θ_0 to compensate for the rotation.

The warping function is described as $\Pi : \mathbb{R}^3 \rightarrow \mathbb{R}^3$, which warps the event's position on image plane as $\Pi\left(x, y, \bar{\theta}(t)\right)$: $(x, y, \theta) \rightarrow (x', y', \theta_0)$. The warping function can be written as

$$e'_i = \Pi\{(x, y, \theta)\} = \mathbf{K} \cdot g'^{-1}\left(\mathbf{v}_1, \mathbf{z}_w(\boldsymbol{\theta})\right) \cdot \mathbf{K}^{-1} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = (x', y', \theta_0) \quad (15)$$

From the warped events $E' = e'_i(i = 1,2\ldots)$, we constructed the image of wrapped events (IWE) (*74*) H as

$$H = \sum_{e_i \in E'} \zeta(e'_i) \quad (16)$$

where each pixel (i, j) sums the warped events e_i that mapped to it. ζ represents intensity spikes, where $\zeta(e'_i) = 1$ means e'_i is mapped to (i, j), otherwise $\zeta(e'_i) = 0$. To evaluate the quality of this calibration parameter pair, we designed a cost function by leveraging the idea of motion compensation (*52*). Because a well-parameterized IWE will warp events triggered by the same incoming light to the same pixel, the IWE should be sharp. Therefore, we designed our cost function J to measure the sharpness of the IWE:

$$J = \sum_{i,j} b_{i,j}\left(1 + \exp\left(\frac{h(i,j)}{\eta}\right)\right)^{-1} \quad (17)$$

where $h(i, j)$ is the value of pixel (i, j) in H, and η is the scale factor. If $h(i, j)$ is positive, then $b_{i,j}$ is set to 1; otherwise, $b_{i,j} = 0$ so that the cost would not be summed. We used the exponential in the above equation because it heavily weighted pixels with low numbers of events. Therefore, the sharpness of IWE is inversely proportional to the cost, as shown in Fig. 8C. The optimal parameter pair δ, θ_0 was optimized by maximizing the sharpness, or contrast, of IWE: $\min_{\delta, \theta_0} J$.

In practice, the above equation was robustly solved by a coarse-to-fine search. The search process was demonstrated in Fig. 8C. It was formulated as a standard circular function fitting problem as shown in eq. S1. Figure S3 indicates that the optimal solution is unique. Moreover, in Supplementary Methods, we further prove that eq. S1 is convex in a certain domain, which means that it can be solved much faster if the initial guess is precise.

Supplementary Materials

This PDF file includes:
Methods
Figs. S1 to S13
References (*76–83*)

Other Supplementary Material for this manuscript includes the following:
Movies S1 to S4
MDAR Reproducibility Checklist

REFERENCES AND NOTES

1. S. Martinez-Conde, S. L. Macknik, D. H. Hubel, The role of fixational eye movements in visual perception. *Nat. Rev. Neurosci.* **5**, 229–240 (2004).
2. R. G. Alexander, S. Martinez-Conde, Fixational eye movements, in *Eye Movement Research: An Introduction to its Scientific Foundations and Applications*, C. Klein, U. Ettinger, Eds. (Springer International Publishing, 2019), pp. 73–115.
3. M. Rucci, R. Iovin, M. Poletti, F. Santini, Miniature eye movements enhance fine spatial detail. *Nature* **447**, 852–855 (2007).
4. J. A. Leñero-Bardallo, T. Serrano-Gotarredona, B. Linares-Barranco, A 3.6 μ latency asynchronous frame-free event-driven dynamic-vision-sensor. *IEEE J. Solid State Circuits* **46**, 1443–1455 (2011).
5. D. Falanga, K. Kleber, D. Scaramuzza, Dynamic obstacle avoidance for quadrotors with event cameras. *Sci. Robot.* **5**, eaaz9712 (2020).
6. B. He, H. Li, S. Wu, D. Wang, Z. Zhang, Q. Dong, C. Xu, F. Gao, Fast-dynamic-vision: Detection and tracking dynamic objects with event and depth sensing, in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2021), pp. 3071–3078.
7. A. Mitrokhin, Z. Hua, C. Fermüller, Y. Aloimonos, Learning visual motion segmentation using event surfaces, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2020), pp. 14414–14423.
8. N. J. Sanket, C. M. Parameshwara, C. D. Singh, A. V. Kuruttukulam, C. Fermuller, D. Scaramuzza, Y. Aloimonos, EvDodge: Embodied AI for high-speed dodging on a quadrotor using event cameras. arXiv:1906.02919 [cs.RO] (7 June 2019).
9. Y. Zhou, G. Gallego, S. Shen, Event-based stereo visual odometry. *IEEE Trans. Robot.* **37**, 1433–1450 (2021).
10. H. Rebecq, T. Horstschäfer, G. Gallego, D. Scaramuzza, Evo: A geometric approach to event-based 6-DOF parallel tracking and mapping in real time. *IEEE Robot. Autom. Lett.* **2**, 593–600 (2016).
11. A. R. Vidal, H. Rebecq, T. Horstschaefer, D. Scaramuzza, Ultimate slam? Combining events, images, and IMU for robust visual slam in hdr and high-speed scenarios. *IEEE Robot. Autom. Lett.* **3**, 994–1001 (2018).
12. A. Mitrokhin, C. Fermüller, C. Parameshwara, Y. Aloimonos, Event-based moving object detection and tracking, in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2018), pp. 1–9.
13. A. Dietsche, G. Cioffi, J. Hidalgo-Carrió, D. Scaramuzza, Powerline tracking with event cameras, *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2021), pp. 6990–6997.
14. G. Cohen, S. Afshar, B. Morreale, T. Bessell, A. Wabnitz, M. Rutten, A. van Schaik, Event-based sensing for space situational awareness. *J. Astronaut. Sci.* **66**, 125–141 (2019).
15. E. Mueggler, C. Bartolozzi, D. Scaramuzza, Fast event-based corner detection, in *Proceedings of the British Machine Vision Conference (BMVC)* (BMVA, 2017), pp. 33.1–33.11.
16. I. Alzugaray, M. Chli, Asynchronous corner detection and tracking for event cameras in real time. *IEEE Robot. Autom. Lett.* **3**, 3177–3184 (2018).
17. W. Guan, P. Lu, Monocular event visual inertial odometry based on event-corner using sliding windows graph-based optimization, in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2022), pp. 2438–2445.
18. C. Ye, A. Mitrokhin, C. Fermüller, J. A. Yorke, Y. Aloimonos, Unsupervised learning of dense optical flow, depth and egomotion from sparse event data. arXiv:1809.08625 [cs.CV] (23 September 2018).
19. F. Paredes-Vallés, K. Y. Scheper, G. C. De Croon, Unsupervised learning of a hierarchical spiking neural network for optical flow estimation: From events to global motion perception. *IEEE Trans. Pattern Anal. Mach. Intell.* **42**, 2051–2064 (2019).
20. J. Hagenaars, F. Paredes-Vallés, G. De Croon, Self-supervised learning of event-based optical flow with spiking neural networks, vol. 24 of *Advances in Neural Information Processing Systems* (Curran Associates, 2021), pp. 7167–7179.
21. N. Messikommer, C. Fang, M. Gehrig, D. Scaramuzza, Data-driven feature tracking for event cameras, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2023), pp. 5642–5651.

22. F. Paredes-Vallés, G. C. de Croon, Back to event basics: Self-supervised learning of image reconstruction for event cameras via photometric constancy, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2021), pp. 3446–3455.
23. F. Barranco, C. Fermüller, Y. Aloimonos, Contour motion estimation for asynchronous event-driven cameras. *Proc. IEEE* **102**, 1537–1556 (2014).
24. J. Hidalgo-Carrió, G. Gallego, D. Scaramuzza, Event-aided direct sparse odometry, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2022), pp. 5781–5790.
25. W. Guan, P. Chen, Y. Xie, P. Lu, Pl-evio: Robust monocular event-based visual inertial odometry with point and line features. arXiv:2209.12160 [cs.CV] (25 September 2022).
26. P. Chen, W. Guan, P. Lu, Esvio: Event-based stereo visual inertial odometry. arXiv:2212.13184 [cs.RO] (26 December 2022).
27. C. Brandli, T. A. Mantel, M. Hutter, M. A. Hopflinger, R. Berner, R. Siegwart, T. Delbruck, Adaptive pulsed laser line extraction for terrain reconstruction using a dynamic vision sensor. *Front. Neurosci.* **7**, 275 (2014).
28. N. Matsuda, O. Cossairt, M. Gupta, Mc3d: Motion contrast 3d scanning, in *2015 IEEE International Conference on Computational Photography (ICCP)* (IEEE, 2015), pp. 1–10.
29. M. Muglikar, G. Gallego, D. Scaramuzza, Esl: Event-based structured light, in *2021 International Conference on 3D Vision (3DV)* (IEEE, 2021), pp. 1165–1174.
30. M. Muglikar, L. Bauersfeld, D. P. Moeys, D. Scaramuzza, Event-based shape from polarization, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2023), pp. 1547–1556.
31. M. Muglikar, D. P. Moeys, D. Scaramuzza, Event guided depth sensing, in *2021 International Conference on 3D Vision (3DV)* (IEEE, 2021), pp. 385–393.
32. G. Orchard, A. Jayawant, G. K. Cohen, N. Thakor, Converting static image datasets to spiking neuromorphic datasets using saccades. *Front. Neurosci.* **9**, 437 (2015).
33. A. Yousefzadeh, G. Orchard, T. Serrano-Gotarredona, B. Linares-Barranco, Active perception with dynamic vision sensors. Minimum saccades with optimum recognition. *IEEE Trans. Biomed. Circuits Syst.* **12**, 927–939 (2018).
34. B. He, Z. Wang, Y. Zhou, J. Chen, C. D. Singh, H. Li, Y. Gao, S. Shen, K. Wang, Y. Cao, C. Xu, Y. Aloimonos, F. Gao, C. Fermüller, *Microsaccade-Inspired Event camera for Robotics* (2023); https://zenodo.org/records/8157775.
35. D. Senderakova, A. Strba, Analysis of a wedge prism to perform small-angle beam deviation. *Proc. SPIE* **5036**, 148–151 (2003).
36. Intel, Intel RealSense Depth Camera D435 (2023); https://www.intelrealsense.com/depth-camera-d435/.
37. J. Lin, F. Zhang, R3live: A robust, real-time, RGB-colored, LiDAR-inertial-visual tightly coupled state estimation and mapping package, in *2022 International Conference on Robotics and Automation (ICRA)* (IEEE, 2022), pp. 10672–10678.
38. S. Chen, J. Wang, W. Pan, S. Gao, M. Wang, X. Lu, Towards uniform point distribution in feature-preserving point cloud filtering. *Computat. Vis. Media* **9**, 249–263 (2023).
39. S. Kodors, Point distribution as true quality of lidar point cloud. *Balt J. Mod. Comput.* **5**, 362–378 (2017).
40. J. Canny, A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **PAMI-8**, 679–698 (1986).
41. R. Madaan, D. Maturana, S. Scherer, Wire detection using synthetic data and dilated convolutional networks for unmanned aerial vehicles, in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2017), pp. 3487–3494.
42. P. Arbelaez, M. Maire, C. Fowlkes, J. Malik, Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**, 898–916 (2010).
43. H. Rebecq, R. Ranftl, V. Koltun, D. Scaramuzza, High speed and high dynamic range video with an event camera. *IEEE Trans. Pattern Anal. Mach. Intell.* **43**, 1964–1980 (2019).
44. H. Rebecq, R. Ranftl, V. Koltun, D. Scaramuzza, Events-to-video: Bringing modern computer vision to event cameras, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2019), pp. 3857–3866.
45. S. Iwasaki, C. Premachandra, T. Endo, T. Fujii, M. Tanimoto, Y. Kimura, Visible light road-to-vehicle communication using high-speed camera, in *2008 IEEE Intelligent Vehicles Symposium* (IEEE, 2008), pp. 13–18.
46. B. Nagy, P. Foehn, D. Scaramuzza, Faster than fast: GPU-accelerated frontend for Highspeed Vio, in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2020), pp. 4361–4368.
47. A. Mittal, R. Soundararajan, A. C. Bovik, Making a 'completely blind' image quality analyzer. *IEEE Signal Process. Lett.* **20**, 209–212 (2013).
48. I. Alzugaray, Event-driven feature detection and tracking for visual slam, thesis, ETH Zurich (2022).
49. Y. Zhou, G. Gallego, X. Lu, S. Liu, S. Shen, Event-based motion segmentation with spatiotemporal graph cuts. *IEEE Trans. Neural Netw. Learn. Syst.* **34**, 4868–4880 (2021).
50. K. Bialik, M. Kowalczyk, K. Blachut, T. Kryjak, Fast-moving object counting with an event camera. arXiv:2212.08384 [cs.CV] (16 December 2022).
51. Prophesee, High-speed counting unlocked by event-based vision (2020); https://www.prophesee.ai/2019/09/19/high-speed-counting-event-based-vision/.
52. T. Stoffregen, G. Gallego, T. Drummond, L. Kleeman, D. Scaramuzza, Event-based motion segmentation by motion compensation, in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), pp. 7244–7253.
53. C. M. Parameshwara, N. J. Sanket, C. D. Singh, C. Fermüller, Y. Aloimonos, 0-mms: Zeroshot multi-motion segmentation with a monocular event camera, in *2021 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2021), pp. 9594–9600.
54. D. Rozumnyi, J. Matas, F. Sroubek, M. Pollefeys, M. R. Oswald, Fmodetect: Robust detection of fast moving objects, in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (IEEE, 2021), pp. 3541–3549.
55. L. Xu, W. Xu, V. Golyanik, M. Habermann, L. Fang, C. Theobalt, Eventcap: Monocular 3d capture of high-speed human motions using an event camera, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2020), pp. 4968–4978.
56. S. Zou, C. Guo, X. Zuo, S. Wang, H. Xiaoqin, S. Chen, M. Gong, L. Cheng, Eventhpe: Event-based 3d human pose and shape estimation, in *Proceedings of the IEEE International Conference on Computer Vision* (IEEE, 2021), pp. 10996–11005.
57. Z. Zhang, K. Chai, H. Yu, R. Majaj, F. Walsh, E. Wang, U. Mahbub, H. Siegelmann, D. Kim, T. Rahman, Neuromorphic high-frequency 3d dancing pose estimation in dynamic environment. *Neurocomputing* **547**, 126388 (2023).
58. G. Gallego, T. Delbruck, G. Orchard, C. Bartolozzi, B. Taba, A. Censi, S. Leutenegger, A. J. Davison, J. Conradt, K. Daniilidis, D. Scaramuzza, Event-based vision: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**, 154–180 (2020).
59. L. Wang, K.-J. Yoon, Deep learning for HDR imaging: State-of-the-art and future trends. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**, 8874–8895 (2022).
60. S. Kreiss, L. Bertoni, A. Alahi, Openpifpaf: Composite fields for semantic keypoint detection and spatio-temporal association. *IEEE Trans. Intell. Transp. Syst.* **23**, 13498–13511 (2022).
61. C. D. Singh, R. Kumari, C. Fermuller, N. J. Sanket, Y. Aloimonos, Worldgen: A large scale generative simulator. arXiv:2210.00715 [cs.CV] (3 October 2022).
62. A. Z. Zhu, D. Thakur, T. Ozaslan, B. Pfrommer, V. R. Kumar, K. Daniilidis, The multivehicle stereo event camera dataset: An event camera dataset for 3d perception. *IEEE Robot. Autom. Lett.* **3**, 2032–2039 (2018).
63. P. F. McManamon, T. A. Dorschner, D. L. Corkum, L. J. Friedman, D. S. Hobbs, M. Holz, S. Liberman, H. Q. Nguyen, D. P. Resler, R. C. Sharp, E. A. Watson, Optical phased array technology. *Proc. IEEE* **84**, 268–298 (1996).
64. M. Schadt, Liquid crystal materials and liquid crystal displays. *Ann. Rev Mater Sci* **27**, 305–379 (1997).
65. B. Tilmon, E. Jain, S. Ferrari, S. Koppal, Foveacam: A MEMS mirror-enabled foveating camera, in *2020 IEEE International Conference on Computational Photography (ICCP)* (IEEE, 2020), pp. 1–11.
66. G. Haessig, X. Berthelon, S.-H. Ieng, R. Benosman, A spiking neural network model of depth from defocus for event-based neuromorphic vision. *Sci. Rep.* **9**, 3744 (2019).
67. J. A. Pérez-Carrasco, B. Zhao, C. Serrano, B. Acha, T. Serrano-Gotarredona, S. Chen, B. Linares-Barranco, Mapping from frame-driven to frame-free event-driven vision systems by low-rate rate coding and coincidence processing–application to feedforward convnets. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**, 2706–2719 (2013).
68. P. U. Diehl, G. Zarrella, A. Cassidy, B. U. Pedroni, E. Neftci, Conversion of artificial recurrent neural networks to spiking neural networks for low-power neuromorphic hardware, in *2016 IEEE International Conference on Rebooting Computing (ICRC)* (IEEE, 2016), pp. 1–8.
69. DJI, RoboMaster M2006 P36 Brushless DC Gear Motor (2023); https://www.robomaster.com/en-US/products/components/detail/1277.
70. iniVation AG, DVXplorer (2023); https://inivation.com/.
71. DJI, RoboMaster Development Board Type C (2023); https://www.robomaster.com/en-US/products/components/general/development-board-type-c/info.
72. M. Born, E. Wolf, *Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light* (Elsevier, 2013).
73. E. Renner, *Pinhole Photography: From Historic Technique to Digital Application* (Routledge, 2012).
74. G. Gallego, H. Rebecq, D. Scaramuzza, A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2018), pp. 3867–3876.
75. S. Tulyakov, D. Gehrig, S. Georgoulis, J. Erbach, M. Gehrig, Y. Li, D. Scaramuzza, Time lens: Event-based video frame interpolation, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2021), pp. 16155–16164.
76. Y. Hu, S.-C. Liu, T. Delbruck, v2e: From video frames to realistic DVS events, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2021), pp. 1312–1321.
77. D. Gehrig, M. Gehrig, J. Hidalgo-Carrio, D. Scaramuzza, Video to events: Recycling video datasets for event cameras, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2020), pp. 3586–3595.
78. H. Jiang, D. Sun, V. Jampani, M.-H. Yang, E. Learned-Miller, J. Kautz, Super slomo: High quality estimation of multiple intermediate frames for video interpolation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2018), pp. 9000–9008.

79. X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, W.-C. Woo, Convolutional lstm network: A machine learning approach for precipitation nowcasting, in vol. 28 of *Advances in Neural Information Processing Systems* (Curran Associates, 2015).
80. Intel, Intel NUC 10 Performance kit–NUC10i7FNH (2023); https://www.intel.com/content/dam/www/public/us/en/documents/product-briefs/nuc-10-performance-brief.pdf.
81. Intel, Intel Core i7-10710U Processor (2023); https://www.intel.com/content/www/us/en/products/sku/196448/intel-core-i710710u-processor-12m-cache-up-to-4-70-ghz/specifications.html.
82. S. Tulyakov, A. Bochicchio, D. Gehrig, S. Georgoulis, Y. Li, D. Scaramuzza, Time lens++: Event-basedframe interpolation with parametric non-linearflow and multi-scale fusion, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (IEEE, 2022), pp. 17755–17764.
83. M. Gehrig, M. Millhäusler, D. Gehrig, D. Scaramuzza, E-raft: Dense optical flow from event cameras, in *2021 International Conference on 3D Vision (3DV)* (IEEE, 2021), pp. 197–206.

Acknowledgments: We thank X. Zhou, K. Burner, and S. Shrestha for the valuable suggestions for the manuscript. We sincerely appreciate J. Lin, L. Xu, and Z. Tian for the help in real-world experiments. Furthermore, we are truly grateful for Y. Zhou's suggestions regarding the design of the experiment. Part of the work was done when B.H. was a member of Zhejiang University. Original idea raised at Zhejiang University. **Funding:** This work was supported by the National Natural Science Foundation of China under grant no. 62322314, the National Science Foundation under grant OISE 2020624 supporting Research through International Network-to-Network Collaboration ("AccelNet: Accelerating Research on Neuromorphic Perception, Action, and Cognition"), and the National Natural Science Foundation of China under grant no. 62088101. **Author contributions:** Conceptualization: B.H., F.G., and C.F. Methodology: B.H., Z.W., F.G., C.F., and Y.A. Investigation: B.H., Z.W., Y.Z., J.C., C.D.S., and H.L. Visualization: B.H., Y.G., Z.W., H.L., Y.Z., and C.D.S. Funding acquisition: F.G., C.F., and Y.A. Project administration: F.G., C.F., Y.A., C.X., Y.C., and K.W. Supervision: F.G., C.F., Y.A., C.X., Y.C., K.W., and S.S. Writing–original draft: B.H., Z.W., C.F., F.G., and Y.A. Writing–review and editing: B.H., Z.W., C.F., F.G., and Y.A. **Competing interests:** The authors declare that they have no competing interests. **Data and materials availability:** Data needed to evaluate the conclusions in the paper have been made available for download at https://zenodo.org/records/8157775 (*34*).

Submitted 24 July 2023
Accepted 1 May 2024
Published 29 May 2024
10.1126/scirobotics.adj8124