A Novel Illumination-Robust Hand Gesture Recognition System With Event-Based Neuromorphic Vision Sensor

Guang Chen, *Member, IEEE*, Zhongcong Xu, Zhijun Li, *Senior Member, IEEE*, Huajin Tang, *Senior Member, IEEE*, Sanqing Qu, Kejia Ren, and Alois Knoll, *Senior Member, IEEE*

***Abstract*— The hand gesture recognition system is a noncontact and intuitive communication approach, which, in turn, allows for natural and efficient interaction. This work focuses on developing a novel and robust gesture recognition system, which is insensitive to environmental illumination and background variation. In the field of gesture recognition, standard vision sensors, such as CMOS cameras, are widely used as the sensing devices in state-of-the-art hand gesture recognition systems. However, such cameras depend on environmental constraints, such as lighting variability and the cluttered background, which significantly deteriorates their performances. In this work, we propose an event-based gesture recognition system to overcome the detriment constraints and enhance the robustness of the recognition performance. Our system relies on a biologically inspired neuromorphic vision sensor that has microsecond temporal resolution, high dynamic range, and low latency. The sensor output is a sequence of asynchronous events instead of discrete frames. To interpret the visual data, we utilize a wearable glove as an interaction device with five high-frequency (>100 Hz) active LED markers (ALMs), representing fingers and palm, which are tracked precisely in the temporal domain using a restricted spatiotemporal particle filter algorithm. The latency of the sensing pipeline is negligible compared with the dynamics of the environment as the sensor's temporal resolution allows us to distinguish high frequencies precisely. We design an encoding process to extract features and adopt a lightweight network to classify the hand gestures. The recognition accuracy of our system is comparable to the state-of-the-art methods. To study the robustness of the system, experiments considering illumination and background variations are performed, and the results show that our system is more robust than the state-of-the-art deep learning-based gesture recognition systems.**

***Note to Practitioners*—This article addresses the robustness of the hand gesture recognition system that is important for gesture recognition-based applications. Existing methods rely on either the large-volume data to train a deep learning model or to restrict the applied environments (e.g., an ideal environment without dynamic background). However, a vision-based deep learning model requires large computational resources, while the ideal environment limits the practicality of the system. In this work, we introduce a biologically inspired neuromorphic vision sensor and an ALM glove and build a novel gesture recognition system to tackle the above issue. The neuromorphic vision sensor has a microsecond temporal resolution and a high dynamic range. With these properties, the sensing system of our prototype operates in a very low-latency space, which, in turn, ensures that our gesture recognition system is robust to illumination variance and dynamic background. Thus, this work is valuable to the research of illumination-robust gesture recognition systems. Preliminary experiments suggest that our system prototype is feasible, but it has not yet been incorporated into an online gesture recognition system nor tested with complex gestures. In future work, we will concentrate on the improvement of the signal processing methods that advance the current system to complex and practical applications.**

***Index Terms*— Active LED marker (ALM), biologically inspired learning, biologically inspired signal processing, dynamic and active-pixel vision sensor (DAVIS), event-based neuromorphic vision, hand gesture recognition, illumination-robust system, wearable device.**

I. Introduction

HAND gesture is an intuitive and ubiquitous approach to represent people's thoughts and intentions naturally and directly. In the last decade, hand gesture recognition has witnessed remarkable progress due to the rapid development of computer vision and machine learning. Accordingly, applications such as human–machine interface [1], [2], body sign language [3]–[5], and virtual reality [6] are developed rapidly.

Manuscript received October 1, 2019; revised June 21, 2020, October 9, 2020, and November 12, 2020; accepted November 29, 2020. Date of publication January 18, 2021; date of current version April 7, 2021. This article was recommended for publication by Editor M. Zhang upon evaluation of the reviewers' comments. This work was supported in part by the European Union's Horizon 2020 Framework Programme for Research and Innovation under the Specific Grant Agreement 945539 (Human Brain Project SGA3), in part by the Shanghai AI Innovative Development Project 2018, and in part by the National Science and Technology Major Project of the Ministry of Science and Technology of China under Grant 2018AAA0102900. *(Corresponding author: Zhijun Li.)*

Guang Chen is with the School of Automotive Studies, Tongji University, Shanghai 200092, China, and also with the Chair of Robotics, Artificial Intelligence and Real-time Systems, Technical University of Munich, 80333 Munich, Germany.

Zhongcong Xu is with the Electrical and Computer Engineering Department, National University of Singapore, Singapore 119077.

Zhijun Li is with the Department of Automation, University of Science and Technology of China, Hefei 230026, China (e-mail: zjli@ieee.org).

Huajin Tang is with the College of Computer Science and Technology, Zhejiang University, Hangzhou 310058, China.

Sanqing Qu is with the School of Automotive Studies, Tongji University, Shanghai 200092, China.

Kejia Ren is with the Department of Mechanical Engineering, Johns Hopkins University, Baltimore, MD 212182681 USA.

Alois Knoll is with the Chair of Robotics, Artificial Intelligence and Real-Time Systems, Technical University of Munich, 80333 Munich, Germany.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TASE.2020.3045880.

Digital Object Identifier 10.1109/TASE.2020.3045880

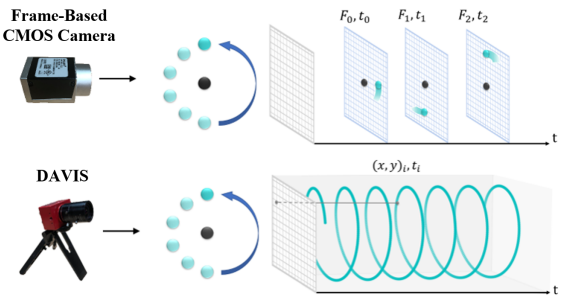


Fig. 1. Illustration of the different working principles of frame-based CMOS camera and neuromorphic vision sensor. The green ball is rotating around a black ball. Frame-based CMOS camera captures intensity values at a fixed rate, while the neuromorphic vision sensor (DAVIS) captures intensity changes asynchronously.

In recent years, many of the state-of-the-art works [7]–[13] develop their gesture recognition algorithms with conventional monocular cameras. Although they achieve great success in regards to the recognition accuracy, the robustness and adaptation abilities at the system level are ignored by most of them. However, as Rautaray and Agrawal [14] mentioned, illumination variations bring difficulties to gesture detection and recognition. The challenging lighting conditions are confounding, a typical approach to alleviate this problem is fusing different data modalities and developing multimodal gesture recognition methods [8], [15]. RGB-D camera is natively suitable for the fusion of data modalities, it captures both the depth information and RGB images; thus, it provides an alternative solution for hand gesture recognition. Although some approaches based on RGB-D cameras have demonstrated good performance in environments with illumination variation [16], [17], there are still two major issues. First, they need to design sophisticated architectures and carefully initialize individual modalities. Second, they are developed on data sets that are collected with common lighting condition changes, such as saturation, high contrast shadows, and light flicker. However, their adaption abilities are still unknown because the analysis of influences caused by luminance variance is ignored.

To tackle the above issues, we propose a biologically inspired vision-based hand gesture recognition system in this work. The bio-inspired vision sensor used in our system is an event-based neuromorphic vision sensor, named dynamic and active-pixel vision sensor (DAVIS) [18], [19]. Instead of measuring the absolute brightness of all pixels at a constant rate, they capture the per-pixel brightness changes (called events) asynchronously,[1] as shown in Fig. 1. This results in outstanding properties compared with frame-based CMOS cameras: very high temporal resolution and low latency (in the order of microsecond), very high dynamic range (140 dB), and low power consumption [20], [21]. Thus, they have a large potential for gesture recognition in challenging scenarios where CMOS cameras do not have a comparative performance [22], [23].

[1]The DAVIS sensor also generates images at fixed frame rate, which are called active pixel sensor (APS) frames that are the same as RGB images by standard CMOS cameras.

A recent work [24] presents a detailed experimental analysis and indicates that color markers are more robust to uneven illumination, thus marker-based models can outperform bare-hand systems in real-world scenarios. The design of our prototype system pays particular attention to the robustness and adaption abilities; thus, we develop a wearable glove with high-frequency (>100 Hz) active LED markers (ALMs). Five ALMs that blink at different frequencies are fixed to the thumb, forefinger, middle finger, ring finger, and palm, respectively. The blink frequencies exceed the frequency triggered by changeable environmental illuminations by a large margin. The reason for using ALMs is that we are able to interpret the event stream at the microsecond level; thus, the latency of our system's sensing component is negligible compared with the dynamics of the background. This property is crucial for the following feature extraction and recognition algorithms and enhances the robustness and adaption abilities of the entire system as well. We adopt a restricted spatiotemporal particle (RSTP) filter to extract the trajectories of the ALMs. Inspired by skeleton-based action recognition algorithms [11]–[13], [25], we design a set of motion capture fuzzy membership functions to generate representative feature sequences for gestures based on the ALMs' trajectories. Finally, we use a shallow long short-term memory (LSTM) network to classify the hand gestures.

In summary, our contributions to this work are as follows.

1) A novel hand gesture recognition system is developed. Our system addresses the challenges faced by most of the state-of-the-art systems: the illumination variations and cluttered background in the scenarios. We tackle them at the sensor-level by introducing a recently developed neuromorphic vision sensor in our system, which is fundamentally different from other works that aim to solve these issues at the algorithm level (e.g., relying on a large-scale data set and elaborated deep neural networks).
2) The latency of our sensing pipeline is negligible compared with the dynamics of the background. We design a wearable glove with high-frequency ALMs that fully exploit the properties (high temporal resolution and low latency) of the DAVIS sensor. Experiments are performed to exam the robustness and adaption abilities our system gains. Results show that our system is much more robust compared with state-of-the-art deep learning-based methods in challenging scenarios.

The rest of this article is organized as follows. In Section II, we discuss the related works. In Section III, we describe all details of our illumination-robust hand gesture recognition system. Section V demonstrates the experiments and results, and finally, we conclude our work in Section VI.

II. Related Works

The basic and general principle components of a common hand gesture recognition system can be summarized as data acquisition, hand localization, feature extraction, and gesture classification.

According to the sensors used to capture the motion, gesture recognition systems can be generally divided into

two basic categories: vision-based systems [11]–[13], [26] and nonvision-based systems [27]. Among the vision-based systems, the monocular RGB camera has been widely utilized, and most of the state-of-the-art gestural recognition methods with RGB cameras elaborate on convolutional neural network (CNN) architectures and achieve high recognition accuracy. Wang *et al.* [7], [8] and Tran *et al.* [28] develop their methods using CNNs based on spatiotemporal features extracted at body and hand levels. In recent years, the emergence of depth camera has significantly facilitated the development of hand gesture recognition. Two recent works [29], [30] review the 3-D hand gesture recognition algorithms based on several typical depth cameras, such as Microsoft Kinect. Pisharady and Saerbeck [31] conduct 2-D and 3-D sensor-based hand gesture recognition comparison experiments and conclude that 2-D appearance-based approaches are more computationally efficient, while 3-D model-based approaches have a better generality for distance, orientation, and viewing angle.

However, frame-based cameras have their fundamental drawbacks. The monocular camera is highly sensitive to illumination variation [14], and the data redundancy of 3-D depth sensors makes the computation resource consumption extremely high [32]. Compared with conventional frame-based cameras, the neuromorphic cameras have several outstanding properties that can significantly improve recognition performance, while feature extraction of events is totally different. These works [33]–[37] develop effective feature representation method for the event stream; also, some studies develop event-based hand gesture recognition systems. Lee *et al.* [32] are the first to develop an event-based hand gesture recognition system with dynamic vision sensors (DVSs), and it proposes a processing method for raw events named leaky integrate-and-fire (LIF) neurons. Amir *et al.* [22] develop an energy-efficient real-time hand gesture recognition system with event-based processors; also, one of its main contributions is a new hand-gesture data set collected under three illumination conditions with an event-based camera. However, it focuses on real-time performance without making any further analysis of the influences of different illumination conditions. Maro and Benosman [38] take full advantage of the high temporal resolution of event-based cameras to remove dynamic backgrounds caused by walking, which makes the hand gesture recognition algorithm suitable for outdoor scenarios. A recent work [39] proposes a graph-based spatiotemporal feature for neuromorphic vision sensors, which makes a great contribution to action recognition tasks using event-based cameras.

Although hand gesture recognition has witnessed considerable progress and achieved great success in applications, there are still some important factors that can degrade the robustness of recognition systems, such as lighting variability and cluttered backgrounds. Confronted with the challenges, there have been some effective algorithms. Rautaray and Agrawal [14] summarize the early methods adopted to increase invariance against illumination variability, such as approximating the chromaticity of skin rather than its apparent color value in color space. Besides, in [40] and [41], the effects of illumination are analyzed via normalization, modeling, and invariant representation. Similarly, in [42], time-varying illumination is handled by dynamic illumination estimation. The recent works [43], [44] are focused on shadow removing at the learning stage. Wang *et al.* [43] propose a new illumination-invariant feature based on the approximation estimation strategy of bidirectional reflectance distribution function to move cast shadows, but its efficiency still needs to be optimized. Instead of constructing models to represent illumination at the learning stage or conducting complicated illumination preprocessing, we adopt an event-based vision sensor and a high-frequency ALM glove to improve the illumination invariance of recognition performance from the system perspective. With advantageous properties of the event-based camera, the high-frequency ALMs can be distinguished simply in the light-changing scenarios with disturbed dynamic backgrounds, and thus, the robustness of our system can be improved effectively.

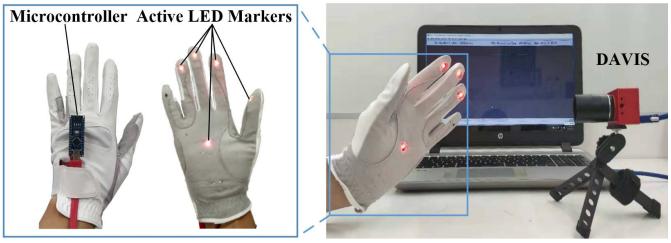


Fig. 2. Illustration of our hand gesture recognition system. A system prototype consists of an event-based neuromorphic vision sensor and an ALM glove.

III. Neuromorphic Gesture Recognition System

In this section, the neuromorphic gesture recognition system is introduced. We systematically describe the implementation and algorithm of our system, including the system prototype, ALM tracking algorithm, feature extraction method, and the lightweight gesture classifier.

A. System Prototype

The prototype of our illumination-robust hand gesture recognition system consists of an event-based neuromorphic vision sensor (DAVIS346[2]) and an ALM glove that is shown in Fig. 2. As a novel sensor, DAVIS346 has a high dynamic range, which makes it suitable for scenes with illumination changes in a wide range. Its temporal resolution (μs) is several orders of magnitude higher than frame-based cameras that capture the scene at fixed frame rates (ms). This motivates us to build a gesture recognition system that operates in a low-latency space, which is not affected by the motions and dynamics in the environment. To achieve this, we design an ALM glove that contains five high-frequency (>100 Hz) ALMs. Five ALMs are fixed to the thumb, forefinger, middle finger, ring finger, and palm. Compared with a CMOS camera-based gesture recognition system, our system has two differences. First, the data recorded by our system are the changing directions of the intensities instead of the intensity values at each pixel. Second, the ALMs operate in the frequency

[2]https://inivation.com/dvs/dvs-product-variants/

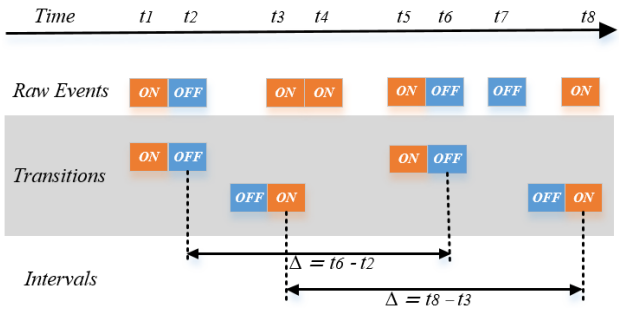


Fig. 3. Illustration of processing a sequence of raw events that occur on the same pixel within a certain temporal window. Event data processing consists of three stages: raw events, transitions, and intervals. Raw events are translated into intervals (Δ) via transitions, and intervals serve as robust estimators for blinking periods of ALMs.

domain at high frequencies, which calls for novel processing algorithms that are different from image-based approaches.

B. ALM Tracking Method

This section describes the ALM tracking method with event streams. The purpose is to extract the trajectory of each ALM. These trajectories will be further processed by the feature extraction method to classify the gesture. The ALM tracking method consists of two phases: the event data processing and the RSTP filter.

1) Event Data Processing: The aim of the event data processing phase is to transfer the raw event data into the direct input of the RSTP filter. Inspired by Censi *et al.* [45] and Chen *et al.* [21], our event data processing process has three stages: raw events, transitions, and intervals (see Fig. 3).

a) Raw events: Event data stream generated by a DAVIS camera can be described as tuples: $\{t_i, p_i, (x_i, y_i)\}$, where i represents the index of an event in event stream, p_i is its polarity and $p_i \in \{\text{on, off}\}$, the on polarity represents a positive change of relative light intensity (from dark to light), while off represents a negative one (from light to dark), t_i is the occurrence timestamp of the event with the unit of μs, and (x_i, y_i) is the coordinate of the event in the image plane, $x_i \in \{0 \ldots 345\}$ and $y_i \in \{0 \ldots 259\}$.

b) Transitions: The transition is designed to represent the polarity changes of the raw events and each transition inherits timestamp from certain raw events. Compared with the last event polarity, if the current event polarity has switched, then a transition is generated. Also, a transition has two kinds of polarities $\in$ {positive, negative}, which reflects the changing directions of events' polarity. If the current event's polarity transfers from on to off, a negative transition is generated, and it gains a timestamp from off event; on the contrary, a reverse transformation generates a positive transition with on event's timestamp. We describe the transition as $\{t_k, T_{i,k}, (x_i, y_i)\}$ $(i \geq 2)$, where $T_{i,k}$ is the kth transition generated by the ith and $(i-1)$th events at pixel (x_i, y_i), and t_k is the timestamp of $T_{i,k}$. The transition is a prerequisite for calculating intervals in the next stage.

c) Intervals: We define the interval as the time between two successive transitions with the same polarity (i.e., both of the transitions are positive or negative). An illustration is shown in Fig. 3. An interval is described as $\{\Delta_n, (x_i, y_i)\}$, representing the nth interval at pixel (x_i, y_i). Intuitively, we can speculate that statistics of intervals has a strong correlation with the blinking frequencies of ALMs. As shown in Fig. 4(a), we can clearly observe that the distribution of $\{\Delta_n, (x_i, y_i)\}$ is well approximated by a Gaussian distribution with a mean of 5000 μs for an ALM blinking at the frequency of 200 Hz. Similar distributions can be seen in Fig. 4(b)–(d). Thus, intervals can serve as an important estimator for the ALMs. However, as shown in Fig. 4(c) and (d), the robustness of the intervals declines as blinking frequencies of ALMs increase because the distribution of intervals cannot be accurately approximated by unimodal Gaussian distribution. Hence, the ALM frequencies should be carefully selected. It is worth noting that the frequency selection is a result of trial and error by observing the histogram of intervals generated with five ALMs simultaneously. At the same time, we need to consider the timer limitation of the microcontroller. In this work, we manually select five blinking frequencies of ALMs as 167, 200, 250, 286, and 333 Hz. As shown in Fig. 4(f), five peaks are clearly visible and well separated. Except for the ALM blinking frequency, we also explore the statistical property of intervals generated by dynamic background.

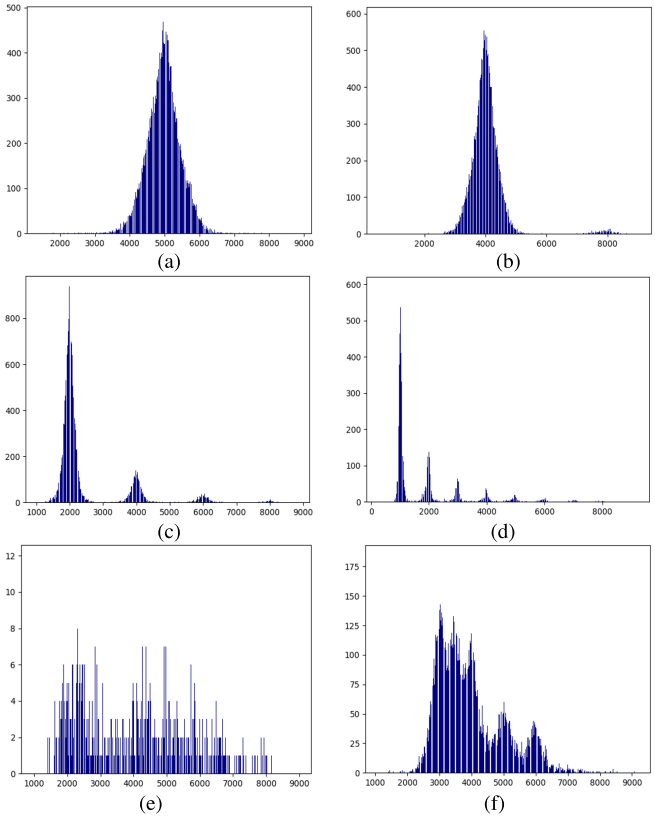


Fig. 4. Histograms of intervals are generated by blinking ALMs in a dynamic environment. The x-axis represents detected interval values in the unit of μs. (a) Single ALM with a blinking frequency of 200 Hz and a period of 5000 μs. (b) Single ALM with a blinking frequency of 250 Hz and a period of 4000 μs. (c) Single ALM with a blinking frequency of 500 Hz and a period of 2000 μs. (d) Single ALM with a blinking frequency of 1000 Hz and a period of 1000 μs. (e) Dynamic background without ALMs. (f) Five ALMs blinking simultaneously with frequencies of 167, 200, 250, 286 and 333 Hz.

As shown in Fig. 4(e), intervals of the dynamic background are chaotic. In summary, the distributions of intervals can be approximated with five different Gaussian distributions. We write the Gaussian distribution with $N(p_m, \sigma_m^2)$, where $m \in \{0, 1 \ldots 4\}$ means the index of ALM, p_m is the mean, and σ_m is the standard deviation.

2) Restricted Spatiotemporal Particle Filter: Based on the transitions and intervals, we develop an RSTP filter to detect and track five ALMs simultaneously.

a) Evidence map: We adopt a sliding-time window strategy to process the asynchronous event stream. For each ALM, we construct an evidence map $E_{m,k}(x, y)$, where $m \in \{0, 1 \ldots 4\}$ represents the index of ALM, (x, y) is the pixel coordinate corresponding to the resolution of camera, $k \in \{1 \ldots N_{\text{iter}}\}$ means the index of sliding-time windows, and N_{iter} depends on the length of raw event data stream. The evidence map can be interpreted as the likelihood that the ith ALM is detected at pixel (x, y) in the kth sliding-time window. The window length in our algorithm is 10 ms, and the step size is 5 ms. Each window represents a 10-ms length of a slice from the event stream. As discussed above, intervals can be approximated by the Gaussian as $N(p_m, \sigma_m^2)$, so each interval $\{\Delta_n, (x_i, y_i)\}$ contributes to the evidence map and rises the probability that ALMs settles at (x_i, y_i). The evidence map is computed as

$$E_{m,k}(x, y) = \sum_{n=1}^{N_{\text{int}}} N\left(\Delta_n \middle| p_m, \sigma_m^2\right) \tag{1}$$

where m is the index of ALM, k is the index of sliding-time window, N_{int} represents the total number of intervals, and Δ_n means the valid intervals at pixel (x, y).

For each ALM, we assign 2000 random particles. However, after the normalization, some particles have extremely small evidence values that lead to the tracking inefficient. Therefore, after the particle selection, we conduct a reselection process that is expressed as

$$\sum_{j=1}^{N_p} E_{m,k}(x_j, y_j) < T \tag{2}$$

where $j \in \{1 \ldots N_p\}$ is the index of the particle, N_p is the particle number, (x_j, y_j) is the coordinate of the particle, and T represents the reselection threshold. In our experiment, T is set to 0.5. When (2) is satisfied, we replace particles that have smaller evidence values with particles having higher evidence values and then normalize them again.

b) Spatial restriction: As shown in Fig. 4(f), the histogram cannot be totally separated; some parts still overlap with each other. For example, an interval of 3500 μs generated by the 333-Hz ALM will inevitably affect the evidence of 286-Hz ALM. Fig. 5(a) and (b) shows that, when the evidence values at the palm and the thumb are relatively weak, the tracking results tend to drift toward other fingers. However, from the spatial perspective, we discover that the evidence value for each ALM can be differentiated easily. Since the moving distance of an ALM is lower than 3 pixels within 5 ms, while the distances among ALMs are much larger, we put a spatial restriction to alleviate the mutual influence among ALMs. When computing the evidence map, we set an additional threshold to restrict the distance between the interval occurrence position and the current ALM position. The expression is

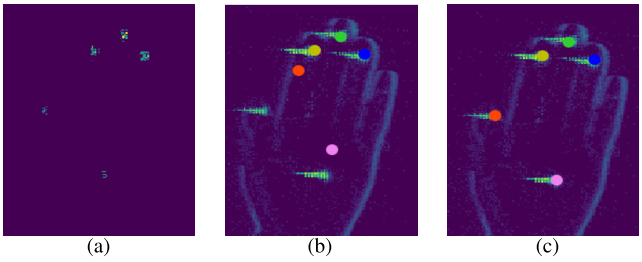


Fig. 5. Tracking results with traditional particle filter and RSTP filter. (a) Evidence map for the particle filter. (b) Tracking results of a traditional particle filter without spatial restriction. (c) Tracking results of our RSTP filter (with restriction).

$$E_{m,k}(x, y) = \begin{cases} \varepsilon, & \text{if } \left\|(x, y) - \left(x_l^m, y_l^m\right)\right\| \geq 15 \\ \displaystyle\sum_{n=1}^{N_{\text{int}}} N\left(\Delta_n \middle| p_m, \sigma_m^2\right), & \text{otherwise} \end{cases} \tag{3}$$

where m is the ALM index, k is the index of the sliding-time window, (x, y) is the coordinate of evidence map, (x_l^m, y_l^m) is the latest position of the mth ALM, $\|(x, y) - (x_l, y_l)\|$ means the distance of two points, and ε is set to $1e-4$ to keep particle weight updating.

The evidence map is applied to update weights of particles as

$$w_{j,k}^m = w_{j,k-1}^m \times E_{m,k}(x_j, y_j) \tag{4}$$

where $w_{j,k}^m$ is the weight of the particle, j is the index of the particle, and m and k represent the index of ALM and sliding-time window, respectively. With the updated particle weight, we check the degeneration degree of the current particles via computing effective number of particles N_{eff} as

$$N_{\text{eff}} = \frac{1}{\sum_{j=1}^{N_p} \left(w_{j,k}^m\right)^2} \tag{5}$$

where N_p is particle number. If $N_{\text{eff}} < \text{Th}_{\text{eff}} \times N_p$, a resampling is inevitable. In our experiments, Th_{eff} is set to 0.5. In addition, the motion model is simplified because the moving distance is lower than 3 pixels in a sliding-time window. In prediction stage, we simply add motion $\sim N(0, 3)$ to the current position. Finally, the normalization of weights is conducted as

$$W_{j,k}^m = \frac{w_{j,k}^m}{\sum_{i=1}^{N} w_{j,k}^m}. \tag{6}$$

The details of the RSTP filter are shown in Algorithm 1.

C. Feature Extraction

In action recognition, the skeleton-based methods achieve good performance by designing handcrafting features to represent the position, orientation, and motions of the human body parts [46], [47]. In this work, we create two kinds of features:

Algorithm 1 RSTP Filter

function (TRAJECTORIES) = RSTP(Raw Event Stream)
  Initialize j Particles
  Last Position$\leftarrow$0
  for $k \leftarrow 1 : N_{iter}$ **do**
    Calculate $\{\Delta_n, (x, y)\}$ based on *Raw Event Stream*
    for $m \leftarrow 1 : 5$ **do**
      if $\|LastPosition - (x, y)\| < 15$ **then**
        $E_{m,k}(x, y) = \sum_{n=1}^{N_{int}} N(\{\Delta_n, (x, y)\}|p_m, \sigma_m^2)$
      else
        $E_{m,k}(x, y) = \varepsilon$
      end if
      Normalize $E_{m,k}$
      if $\sum_{j=1}^{N_p} E_{m,k}(x_j, y_j) < T$ **then**
        Reselect Particles
      end if
      for $j \leftarrow 1 : 2000$ **do**
        $w_{j,k}^m = w_{j,k-1}^m \times E_{m,k}(x_j, y_j)$
      end for
      Normalize $w_{j,k}^m$
      if $N_{eff} = \frac{1}{\sum_{j=1}^{N_p} (w_{j,k}^m)^2} < Th_{eff} \times N_p$ **then**
        Resample Particles
      end if
      Normalize $w_{j,k}^m$ into $W_{j,k}^m$
      Calculate Position based on $W_{j,k}^m$
      Append Position to Trajectories
      Last Position$\leftarrow$ Position
      for $j \leftarrow 1 : 2000$ **do**
        Add Motion to (x_j, y_j)
      end for
    end for
  end for
end function

the holistic motion feature and the local shape feature. The holistic motion feature describes the motion orientation and velocity of the hand. The local shape feature is composed of the relative position and distance between the keypoints (here keypoints are marked by the ALMs positions). We define our feature matrix as

$$\text{features}_t = [\text{pos}_t, \text{ori}_t, \text{shape}_t] \tag{7}$$

where pos_t is translation feature, ori_t is orientation feature, shape_t is the shape feature, and t is the index of sampled sliding-time windows. We adopt the fuzzy membership functions [47] to normalize these features.

1) Translation Features: The translation feature pos_t contains $[\text{pos}X_t, \text{pos}Y_t]$, which represents the location of the hand. Due to the changing distances between the hand and DAVIS camera, we introduce a scale factor to address this problem.

We define the finger length as the 2-D distance between the ALM position of the finger and the ALM position of the palm. According to the statistical analysis, the average values of finger lengths are 63, 87, 95, and 86 pixels for thumb, forefinger, middle finger, and ring finger. We define the scale factor f_s as the mean of the ratios of the finger lengths at time t_0 to the corresponding average values of finger lengths, where t_0 is the initial time of the gesture

$$f_s = \frac{1}{4}\left(\frac{63}{\|\text{thumb}_{t_0} - \text{palm}_{t_0}\|} + \frac{87}{\|\text{fore}_{t_0} - \text{palm}_{t_0}\|} + \frac{95}{\|\text{middle}_{t_0} - \text{palm}_{t_0}\|} + \frac{86}{\|\text{ring}_{t_0} - \text{palm}_{t_0}\|}\right) \tag{8}$$

where (palm, thumb, fore, middle, ring) means the position of palm, thumb, forefinger, middle finger, and ring finger. For normalization, we introduce a fuzzy membership function

$$\text{Tr}(x, k) = \begin{cases} x, & 0 \le |x| < k \\ \dfrac{k}{|x|}x, & k \le |x| \end{cases} \tag{9}$$

where k is a constant. As the moving distance of a fingertip during an entire period of a gesture is lower than 100 pixels, we compute the translation feature as

$$[\text{pos}X_t, \text{pos}Y_t] = \begin{cases} [0, 0], & t = t_0 \\ \text{Tr}([\text{pos}X_{t-1}, \text{pos}Y_{t-1}] + \dfrac{1}{100} \\ \quad \times \text{Tr}(f_s \cdot (\text{palm}_t - \text{palm}_{t-1}), k_0), 1). \\ \quad \text{otherwise} \end{cases} \tag{10}$$

where k_0 is determined by the interval between successive sampled sliding-time windows.

2) Shape Features: The feature shape_t contains $[\text{shapeT}_t, \text{shapeF}_t, \text{shapeM}_t,$ and $\text{shapeR}_t]$, which represents the shape feature of the thumb, forefinger, middle finger, and ring finger. To describe the shape changes of hand gestures, we compute the finger length at each timestamp t and get the ratio of these finger lengths to their first appearances at t_0 (where t_0 is the initial time of the gesture). Thus, the motion of the hand is converted to a motion in the standard plane. To normalize the shape features, we introduce a normalization function as

$$\text{Sh}(x) = \begin{cases} x - 1, & 0 < x \ge 1 \\ \dfrac{x - 1}{5}, & < x \ge 5 \\ 1, & x > 5. \end{cases} \tag{11}$$

Then, the shape feature of each finger is computed as

$$\begin{aligned} \text{shapeT}_t &= \text{Sh}\left(\frac{\|\text{thumb}_t - \text{palm}_t\|}{\|\text{thumb}_{t_0} - \text{palm}_{t_0}\|}\right) \\ \text{shapeF}_t &= \text{Sh}\left(\frac{\|\text{fore}_t - \text{palm}_t\|}{\|\text{fore}_{t_0} - \text{palm}_{t_0}\|}\right) \\ \text{shapeM}_t &= \text{Sh}\left(\frac{\|\text{middle}_t - \text{thumb}_t\|}{\|\text{middle}_{t_0} - \text{palm}_{t_0}\|}\right) \\ \text{shapeR}_t &= \text{Sh}\left(\frac{\|\text{ring}_t - \text{thumb}_t\|}{\|\text{ring}_{t_0} - \text{palm}_{t_0}\|}\right). \end{aligned} \tag{12}$$

3) Orientation Features: The orientation feature ori_t contains $[\text{ori}_H^t, \text{ori}_P^t, \text{ori}_T^t]$, which represents the hand orientation, palm moving direction, and thumb moving direction, respectively. We define the orientation vector of the hand as $\vec{n}_H^t = \text{middle}_t - \text{palm}_t$. In addition, to represent the moving direction of the hand and fingers, we also define two moving

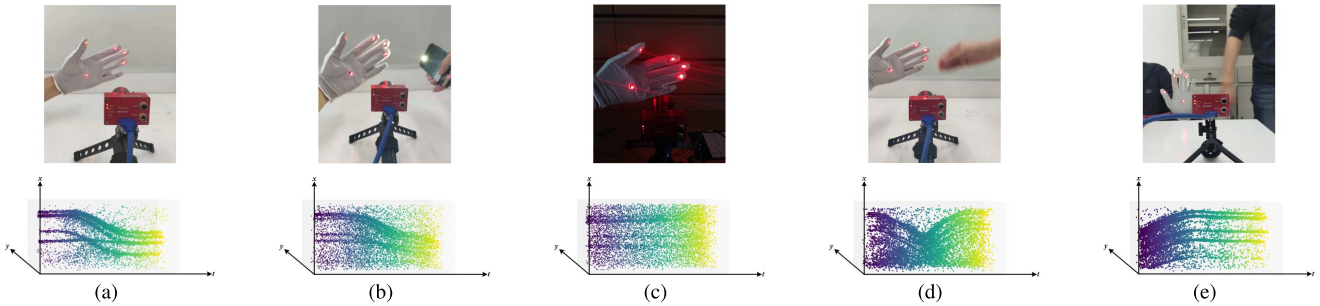


Fig. 6. Illustration of our event-based hand gesture recognition data set. The first raw presents typical environment settings during data collection. The second raw is the visualization of event data recorded in corresponding environments. (a) Recording data in an ideal environment without distraction. (b) Recording data with illumination changes caused by a concurrent flashlight. (c) Recording data in a low-brightness environment. (d) Recording data with dynamic background caused by a waving hand. (e) Recording data with dynamic background caused by a walking person.

orientation vectors as $\vec{n}_P^t = \text{palm}_t - \text{palm}_{t-1}$ and $\vec{n}_T^t = \text{thumb}_t - \text{thumb}_{t-1}$. We choose the orientation of the x-axis, which is expressed as $\vec{e}_x$ as the basic orientation. The fuzzy membership function for the orientation feature is defined as

$$\text{Tw}(\theta) = \frac{\theta - \pi}{2\pi}, \quad 0 \leq \theta \leq 2\pi. \tag{13}$$

The normalized orientation features are computed as

$$\begin{aligned} \text{ori}_H^t &= \text{Tw}\left(\langle \vec{n}_H^t, \vec{e}_x \rangle\right) \\ \text{ori}_P^t &= \text{Tw}\left(\langle \vec{n}_P^t, \vec{e}_x \rangle\right) \\ \text{ori}_T^t &= \text{Tw}\left(\langle \vec{n}_T^t, \vec{e}_x \rangle\right). \end{aligned} \tag{14}$$

D. Classification

The LSTM network [48] is a typical recurrent neural network that can remember the previous information with memory cells. Consequently, it is commonly used to process a temporal sequence of patterns. In our system, we also adopt the LSTM network to capture and learn the feature of hand gestures consecutively. The input sequence is the temporal feature matrix extracted from the ALM trajectories. The number of neurons in the hidden layer is set to 48, and the dimension of the fully connected layer is 48×12, which corresponds to the number of our gesture classes. At last, a softmax function is used to calculate the probability of each class.

IV. Data Set

The motivation of our work is to develop an illumination-robust event-based hand gesture recognition system on top of ALM tracking. To develop our system and evaluate its performance, the data set must contain samples recorded in various scenarios instead of in an ideal environment. We employ a DAVIS346 camera that has a temporal resolution of 1 μs and a high dynamic range of 140 dB to record the raw event stream. There are ten subjects participating in our data collection. Before the recording, we show the hand gesture tutorial videos to make sure that they can follow our gesture definitions exactly. We design 12 different hand gestures, including typical hand motions such as translation, rotation, and circling. The specific gesture definitions are shown in Table I.

TABLE I
List of Gesture Classes Defined in Our Event-Based Gesture Recognition Data Set

Index	Label	Index	Label
1	Move Down	7	Circle Clockwise
2	Move Up	8	Circle Anti-clockwise
3	Move Right	9	Zoom Out
4	Move Left	10	Zoom In
5	Rotate Clockwise	11	Thumb Up
6	Rotate Anti-clockwise	12	Thumb Down

Our data set consists of four sub-data sets: Normal-ALM-ON, Disturbed-ALM-ON, Normal-ALM-OFF, and Disturbed-ALM-OFF. The "Normal" means that the data sets are collected in an ideal environment without illumination variation or cluttered background, while the "Disturbed" means that data sets are recorded in special scenarios with at least one kind of distraction factor (see Fig. 6). There are four distraction factors introduced to our recording procedures: a concurrent flashlight, low-brightness condition, a waving hand, and a walking person as dynamic backgrounds. Since the ALMs that we choose are brighter than the background, we record two data sets, Normal-ALM-OFF and Disturbed-ALM-OFF, with ALMs off for comparison study. Also, two augmented data sets, named Mixed-ALM-ON and Mixed-ALM-OFF, are built, which are the sum of corresponding sub-data sets, respectively. We show the details of our data set in Table II.

V. Experiments

In this section, we evaluate the performance of our hand gesture recognition system. To test the robustness and adaption abilities of our system, we carry out experiments from two aspects: the ALM trajectory tracking experiment and the gesture recognition comparison experiments. In comparison experiments, we compare our system to state-of-the-art deep learning-based approaches with different modalities (Raw Event, Event Frame, and APS Frame) under challenging scenarios.

A. Implementations

1) Data Modalities: The data recorded by DAVIS346 sensor contain both event stream generated by a DVS and APS frames

TABLE II
EVENT-BASED HAND GESTURE RECOGNITION DATA SET

SubDataset	ALM State	Concurrent Flashlight	Low Brightness	Waving Hand	Walking Person	Categories	Samples per Category	Total Gestures
Normal-ALM-ON	ON	✗	✗	✗	✗	12	205	2460
Disturbed-ALM-ON	ON	✓	✓	✓	✓	12	140	1680
Normal-ALM-OFF	OFF	✗	✗	✗	✗	12	180	2160
Disturbed-ALM-OFF	OFF	✓	✓	✓	✓	12	140	1680
Mixed-ALM-ON	ON	✓	✓	✓	✓	12	345	4140
Mixed-ALM-OFF	OFF	✓	✓	✓	✓	12	320	3840

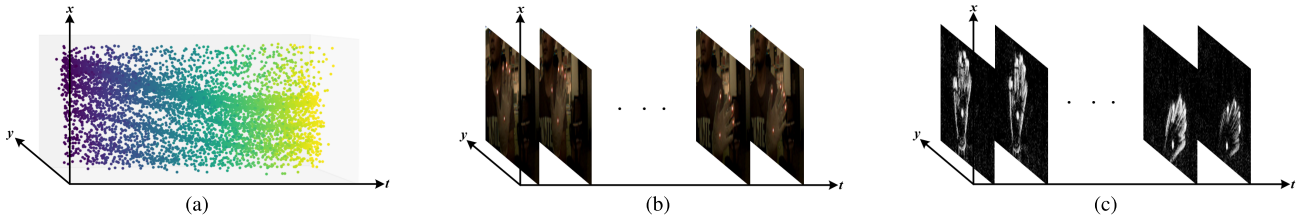


Fig. 7. Visualization of different data modalities. (a) Event stream. (b) APS frame. (c) Event frame.

captured by an integrated built-in CMOS camera. In the following experiments, three modalities are used: Event Stream (with ALM trajectories), APS Frame, and Event Frame. The Event Frame represents the encoded event frames by using the Surface of Active Events (SAE) method, which is widely used in recognition tasks of neuromorphic vision [35], [49]. The SAE encoding method reflects the temporal information since the pixel value and its gradient represent the moving direction and speed of the object, and it translates the raw event into gray-scale images. The pixel value $\sigma(x, y)$ for each Event Frame is calculated by $t_p(x, y)$ and $t_0(x, y)$ (initial timestamp) as follows:

$$\sigma(x, y) = 255 \cdot \frac{t_p(x, y) - t_0(x, y)}{T} \tag{15}$$

where T is the frame intervals and $t_0(x, y)$ is the timestamp of the most recent events at pixel (x, y). Samples of the three modalities are shown in Fig. 7. We can see that the APS Frame and the Event Frame are frame-based images that are originally suitable for deep learning algorithms.

2) Lightweight Model and Comparison: The LSTM network applied in our method is lightweight and has only 48 neurons in the hidden layer. We adopt Adam as the optimizer with an initial learning rate of 0.001. The lightweight LSTM network is evaluated with top-one accuracy in all the experiments with data modality Event Stream. In this work, we also compare our approach to CNN-based methods, as they have been proved to perform well in video-level classification tasks, such as action recognition and gesture recognition. Specifically, two state-of-the-art gesture recognition algorithms, temporal segment network (TSN) [7] and 3-D convolutional network (C3D) [28], are evaluated with data modalities APS Frame and Event Frame.

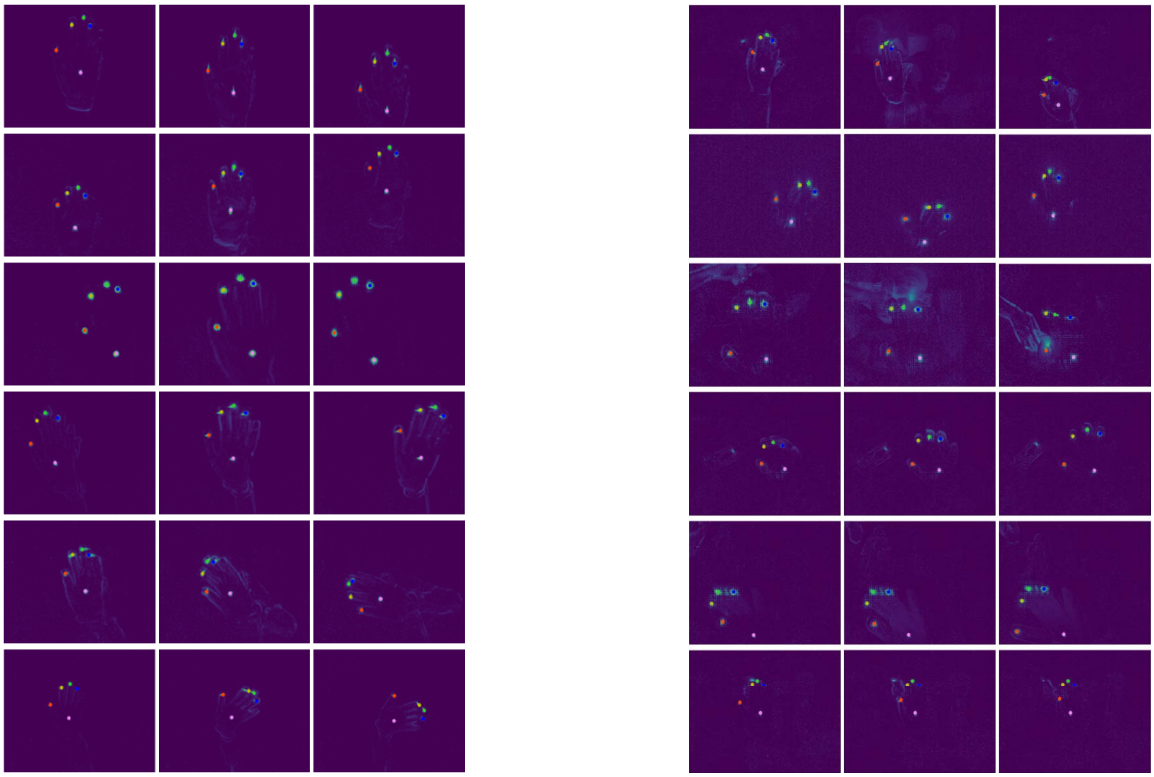
B. ALM Trajectory Tracking Results

In Figs. 8 and 9, we show the ALMs trajectory tracking results under different experimental conditions. Although we cannot quantitatively evaluate the tracking performance due to the impossibility of getting trajectory ground truths, we show the repeatability of tracking trajectories of the same gestures under different experimental settings and the distinction of tracking trajectories of adjacent ALMs. Fig. 8(b) (the first row) clearly shows that, even though the ALMs of the middle finger and ring finger are close to each other, our tracking algorithm can still separate them into two targets. In a similar case with a frame-based camera, this could be a problem because of the high latency. In Fig. 8(b), we can see that the ALMs tracking results are not affected by changing illuminations and dynamic backgrounds. In Fig. 9, we choose the gesture Circle Anticlockwise to demonstrate the trajectory repeatability because, in a rotation gesture, the distances between fingers and the palm remain almost unchanged. For each finger, the ratio of finger-to-palm distance versus middle-finger-to-palm distance is calculated. Fig. 9 shows that, in three lighting conditions, the variation of the calculated ratios is small, which, to a certain extent, demonstrates the repeatability of trajectories and the robustness of the tracking results.

C. Gesture Recognition Results

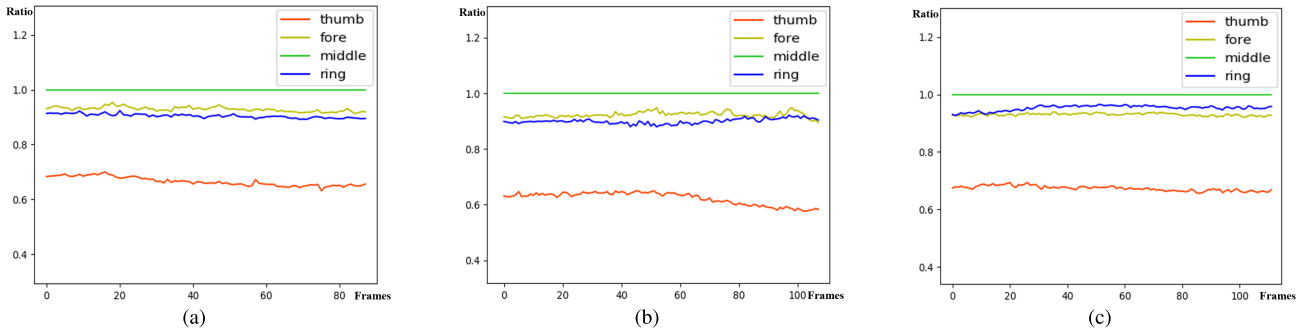
1) Results: Two sets of experiments are conducted to demonstrate the overall performance and the robustness, as well as adaptation abilities of our proposed system.

a) Overall performance: In this experiment, the Mixed-ALM-ON data set is split into the training set, validation set, and testing set with a ratio of 6:2:2, respectively. All of the models are trained from scratch except that the TSN method adopts a pretrained BN-Inception as a backbone. We report the performance with three different testing-set settings: Mixed-ALM-ON, Normal-ALM-ON, and Disturbed-ALM-ON. As we can see from Table III, the modality Event Frame with TSN has the best performance (99.28%), and the modality APS Frame with C3D has the worst performance (95.10%). Although our proposed system with lightweight



(a) (b)

Fig. 8. Visualization of ALMs tracking results. The pink, red, yellow, green, and blue points in each image represent the tracking results of the palm, thumb, forefinger, middle finger, and ring finger, respectively. (a) From top to bottom are the tracking results for Move Down, Move Up, Move Right, Move Left, Rotate Clockwise, and Rotate Anti-clockwise gestures from the Normal-ALM-ON data set. Each row shows three random samples from the corresponding gesture. (b) From top to bottom are the tracking results for Circle Clockwise, Circle Anti-clockwise, Zoom Out, Zoom In, Thumb Up, and Thumb Down gestures from the Disturbed-ALM-ON data set. Each row shows three random samples from the corresponding gesture.



(a) (b) (c)

Fig. 9. Ratios of the fingers' (thumb, forefinger, middle finger, and ring finger) length to middle finger length under different light conditions. (a) Data recorded in an ideal environment. (b) Data recorded with the concurrent flashlight. (c) Data recorded in a low-brightness environment.

LSTM does not achieve the best performance, it has higher overall accuracy (95.36%) than C3D. To analyze the performance of our own method thoroughly, we show the confusion matrix of the results in Fig. 10. As illustrated in the figure, our method has quite good performance for most of the gestures; however, Circle Clockwise and Circle Anti-clockwise (class indices 7 and 8) have worse true-positive rates. Because only the lightweight LSTM network is optimized during the training phase, we conclude that the representative ability of our feature extraction method is poor for the circling gestures. In our method, handcrafting skeleton-based features are predefined. Unlike deep learning architectures, they are not optimized on the data set.

The modality APS Frame with C3D is affected to a larger extent by the testing sources, and the accuracy of the C3D method declines by 4.3% when switching the testing source from the Normal-ALM-ON data set (97.01%) to the Disturbed-ALM-ON data set (92.30%). Moreover, with the APS Frame modality, the accuracy of the C3D method declines by 10.7% when switching the testing source from the Normal-ALM-ON data set (98.44%) to the Disturbed-ALM-ON data set (87.74%). It is worth noting that the training set contains

TABLE III

EXPERIMENTS RESULTS OF OUR METHOD AND CONVOLUTIONAL ARCHITECTURES TRAINED ON THE MIXED-ALM-ON DATA SET WITH DIFFERENT INPUT MODALITIES. STANDARD DEVIATION IS CALCULATED OVER FIVE REPEATED EXPERIMENTS

Modality	Method	Training	Testing	Accuracy(%)
Event Stream	Ours	Mixed-ALM-ON	Mixed-ALM-ON	95.36±0.60
			Normal-ALM-ON	95.48±1.17
			Disturbed-ALM-ON	94.78±0.66
Event Frame	TSN	Mixed-ALM-ON	Mixed-ALM-ON	99.28±0.21
			Normal-ALM-ON	99.49±0.39
			Disturbed-ALM-ON	98.79±0.60
	C3D	Mixed-ALM-ON	Mixed-ALM-ON	95.10±1.22
			Normal-ALM-ON	97.01±1.16
			Disturbed-ALM-ON	92.30±1.87
APS Frame	TSN	Mixed-ALM-ON	Mixed-ALM-ON	98.45±0.92
			Normal-ALM-ON	99.42±0.37
			Disturbed-ALM-ON	97.49±1.49
	C3D	Mixed-ALM-ON	Mixed-ALM-ON	93.60±0.91
			Normal-ALM-ON	98.44±0.67
			Distubred-ALM-ON	87.74±2.14

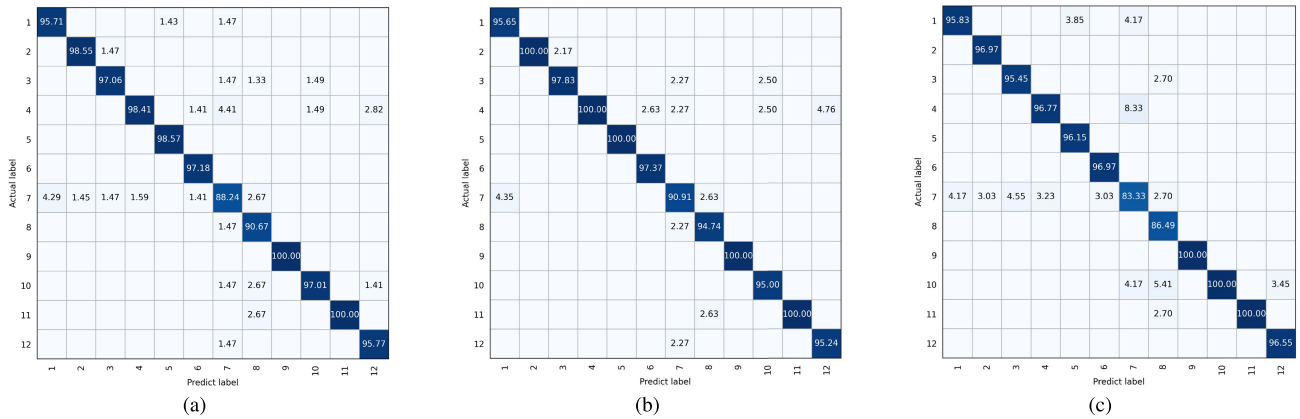


Fig. 10. Classification results of our method trained on the Mixed-ALM-ON data set. (a) Testing on Mixed-ALM-ON. (b) Testing on Normal-ALM-ON. (c) Testing on Disturbed-ALM-ON.

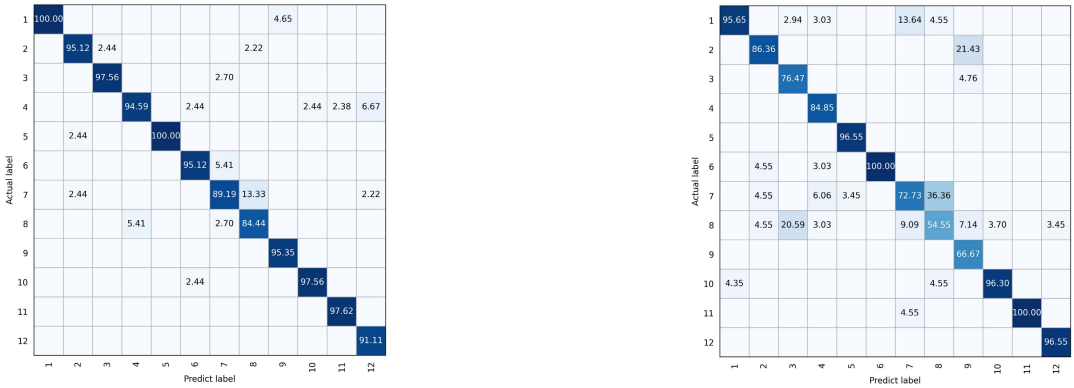
both the data from the Normal-ALM-ON data set and the Disturbed-ALM-ON data set for all the experiments in Table III. In this experiment, TSN, the state-of-the-art algorithm has a constantly outstanding performance that is always better than our method; however, this is based on the assumption that the illumination changes in the testing data set are also incorporated in the training data set, and it cannot reveal the robustness of it when working under new lighting condition settings. Thus, we will analyze the robustness in the following experiments. It is also interesting to see that although both Event Frame and APS Frame are frame-based modalities, the average performance of Event Frame is slightly better than APS Frame. This is because the DAVIS sensor has a larger dynamic range (140 dB) compared with the CMOS camera (60 dB), which can still record the changes of illuminations in either strong-light or dark environments.

b) Robustness and adaption abilities: Most of the SOTA works concentrate on the data from a single domain and does not handle the discrepancy between training and testing phases. Usually, transfer learning is used to solve this problem, which incorporates rich privileged information by exploring additional data distribution. The challenge is that, in real-world applications, the testing scenarios are always unpredictable. Instead of tackling this problem from the algorithm level as many previous works do, we attempt to work on it from the system-level perspective. In this experiment, we separate the training set and testing set to investigate how the discrepancy between training and testing phases affects the performances of different modalities and approaches. All the methods are trained on the Normal-ALM-ON data set, and the performances of different testing sets are reported in Table IV. We can see very clearly that the recognition accuracy is strongly influenced by the discrepancy between the training and testing phases. We define the degree of reduction (DoR) as the deterioration of the accuracy by the same method and modality, while the testing set is switched from Normal-ALM-ON to Disturbed-ALM-ON. The DoR is 9.39% for our proposed system. Fig. 11 shows the confusion matrix of the testing results; we can see that the distractions have strong side effects on Move Up, Move Right, Move Left, Circle Clockwise, Circle Anti-clockwise and Zoom Out, but, for the

TABLE IV

EXPERIMENT RESULTS OF OUR METHOD AND CONVOLUTIONAL ARCHITECTURES. METHODS ARE TRAINED WITH THE TRAINING SET OF NORMAL-ALM-ON DATA SET AND TESTED WITH THE TESTING SET OF NORMAL-ALM-ON/DISTURBED-ALM-ON DATA SET. DOR MEANS THE DETERIORATION OF THE RECOGNITION ACCURACY BY THE SAME METHOD AND MODALITY WHEN THE TESTING SET IS SWITCHED FROM THE NORMAL-ALM-ON TO THE DISTURBED-ALM-ON DATA SET. STANDARD DEVIATION IS CALCULATED OVER FIVE REPEATED EXPERIMENTS

Modality	Method	Training	Testing	Accuracy(%)	Degree of Reduction(%)
Event Stream	Ours	Normal-ALM-ON	Normal-ALM-ON Disturbed-ALM-ON	95.81±1.10 86.43±2.10	9.39±1.88
Event Frame	TSN	Normal-ALM-ON	Normal-ALM-ON Disturbed-ALM-ON	99.72±0.23 67.86±2.27	31.86±2.15
	C3D	Normal-ALM-ON	Normal-ALM-ON Disturbed-ALM-ON	95.78±1.14 32.57±3.01	63.20±3.58
APS Frame	TSN	Normal-ALM-ON	Normal-ALM-ON Disturbed-ALM-ON	99.28±0.80 19.34±1.53	79.94±1.62
	C3D	Normal-ALM-ON	Normal-ALM-ON Disturbed-ALM-ON	98.37±0.75 14.04±2.74	84.34±2.51



(a) (b)

Fig. 11. Classification results of our method trained on the Normal-ALM-ON data set. (a) Testing on Normal-ALM-ON. (b) Testing on Disturbed-ALM-ON.

TABLE V

EXPERIMENT RESULTS OF THE CONVOLUTIONAL ARCHITECTURES. WE TRAIN THE MODEL WITH THE TRAINING SET OF NORMAL-ALM-OFF DATA SET AND TEST THE MODEL WITH TESTING SET OF NORMAL-ALM-OFF/DISTURBED-ALM-OFF DATA SET. DOR MEANS THE DETERIORATION OF THE RECOGNITION ACCURACY BY THE SAME METHOD AND MODALITY WHEN THE TESTING SET IS SWITCHED FROM THE NORMAL-ALM-OFF TO DISTURBED-ALM-OFF DATA SET. STANDARD DEVIATION IS CALCULATED OVER FIVE REPEATED EXPERIMENTS

Modality	Method	Training	Testing	Accuracy(%)	Degree of Reduction(%)
Event Frame	TSN	Normal-ALM-OFF	Normal-ALM-OFF Disturbed-ALM-OFF	99.17±0.39 58.45±3.63	40.74±3.86
	C3D	Normal-ALM-OFF	Normal-ALM-OFF Disturbed-ALM-OFF	93.43±1.60 34.15±2.79	58.92±1.83
APS Frame	TSN	Normal-ALM-OFF	Normal-ALM-OFF Disturbed-ALM-OFF	97.14±0.95 18.09±1.24	79.05±1.96
	C3D	Normal-ALM-OFF	Normal-ALM-OFF Disturbed-ALM-OFF	96.11±1.56 9.17±1.41	86.94±2.52

rest gestures, the performance is still stable. Although our system's performance deteriorates, it still achieves a relatively high mean accuracy at 86.43%. In contrast, for the modalities Event Frame and APS Frame, either the TSN or C3D, both of them get a large DoR (31.86% and 63.20% for the TSN with Event Frame and APS Frame and 79.94% and 84.34% for the C3D with Event Frame and APS Frame). We also witness similar performances with the Normal-ALM-OFF and Disturbed-ALM-OFF data sets in Table V. In summary, experimental results show that our system is much more illumination robust than SOTA deep learning-based methods, such as the TSN and C3D. The relatively small DoR of our system demonstrates that the adaption ability, which has been absent in SOTA methods, has been enhanced.

VI. Conclusion

In this work, we propose an illumination-robust hand gesture recognition system. For a long time, the effects of illumination variation and dynamic backgrounds on the recognition performance are underestimated. State-of-the-art methods heavily rely on large-volume data to overcome the performance deterioration caused by the discrepancy between source data (training) and target data (testing). We tackle this challenge from the system level with a novel neuromorphic vision sensor. The key contribution of our work is that we propose a biologically inspired sensing system for gesture recognition that works in μs temporal resolution. Thus, the latency is negligible compared with the dynamics of environments. Our experiment results show that, with a lightweight classifier, our system achieves comparable performance with state-of-the-art methods. Moreover, the experiment proves that our illumination-robust hand gesture recognition system has strong robustness and adaption abilities that are absent in previous works. In the future, we will replace the current handcrafting feature extraction method with convolutional architectures, such as a fully convolutional network. Besides, although the primary objective of this study is not the improvement of recognition accuracy, a simple fusion strategy by fusing the results from Event Stream and Event Frame can be explored to increase the overall performance while maintaining the robustness.

References

[1] S. Mohatta, R. Perla, G. Gupta, E. Hassan, and R. Hebbalaguppe, "Robust hand gestural interaction for smartphone based AR/VR applications," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2017, pp. 330–335.

[2] Z. Li, J. Li, S. Zhao, Y. Yuan, Y. Kang, and C. L. P. Chen, "Adaptive neural control of a kinematically redundant exoskeleton robot using brain–machine interfaces," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 12, pp. 3558–3571, Dec. 2019.

[3] M. J. Cheok, Z. Omar, and M. H. Jaward, "A review of hand gesture and sign language recognition techniques," *Int. J. Mach. Learn. Cybern.*, vol. 10, no. 1, pp. 131–153, Jan. 2019.

[4] R. Cui, Z. Cao, W. Pan, C. Zhang, and J. Wang, "Deep gesture video generation with learning on regions of interest," *IEEE Trans. Multimedia*, vol. 22, no. 10, pp. 2551–2563, Oct. 2020.

[5] D. Guo, W. Zhou, A. Li, H. Li, and M. Wang, "Hierarchical recurrent deep fusion using adaptive clip summarization for sign language translation," *IEEE Trans. Image Process.*, vol. 29, pp. 1575–1590, 2020.

[6] K. M. Sagayam and D. J. Hemanth, "Hand posture and gesture recognition techniques for virtual reality applications: A survey," *Virtual Reality*, vol. 21, no. 2, pp. 91–107, Jun. 2017.

[7] L. Wang *et al.*, "Temporal segment networks for action recognition in videos," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 11, pp. 2740–2755, Nov. 2019.

[8] H. Wang, P. Wang, Z. Song, and W. Li, "Large-scale multimodal gesture recognition using heterogeneous networks," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 3129–3137.

[9] X. S. Nguyen, L. Brun, O. Lezoray, and S. Bougleux, "A neural network based on SPD manifold learning for skeleton-based hand gesture recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, p. 12.

[10] E. Tsironi, P. Barros, C. Weber, and S. Wermter, "An analysis of convolutional long short-term memory recurrent neural networks for gesture recognition," *Neurocomputing*, vol. 268, pp. 76–86, Dec. 2017.

[11] D. Avola, M. Cascio, L. Cinque, G. L. Foresti, C. Massaroni, and E. Rodola, "2-D skeleton-based action recognition via two-branch stacked LSTM-RNNs," *IEEE Trans. Multimedia*, vol. 22, no. 10, pp. 2481–2496, Oct. 2020.

[12] G. Hu, B. Cui, and S. Yu, "Joint learning in the spatio-temporal and frequency domains for skeleton-based action recognition," *IEEE Trans. Multimedia*, vol. 22, no. 9, pp. 2207–2220, Sep. 2020.

[13] K. Su, X. Liu, and E. Shlizerman, "PREDICT & CLUSTER: Unsupervised skeleton based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9631–9640.

[14] S. S. Rautaray and A. Agrawal, "Vision based hand gesture recognition for human computer interaction: A survey," *Artif. Intell. Rev.*, vol. 43, no. 1, pp. 1–54, Jan. 2015.

[15] D. Wu *et al.*, "Deep dynamic neural networks for multimodal gesture segmentation and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 8, pp. 1583–1597, Aug. 2016.

[16] E. Ohn-Bar and M. M. Trivedi, "Hand gesture recognition in real time for automotive interfaces: A multimodal vision-based approach and evaluations," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 6, pp. 2368–2377, Dec. 2014.

[17] B. Ni, X. Yang, and S. Gao, "Progressively parsing interactional objects for fine grained action detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1020–1028.

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

[19] G. Chen, H. Cao, J. Conradt, H. Tang, F. Rohrbein, and A. Knoll, "Event-based neuromorphic vision for autonomous driving: A paradigm shift for bio-inspired visual sensing and perception," *IEEE Signal Process. Mag.*, vol. 37, no. 4, pp. 34–49, Jul. 2020.

[20] G. Gallego *et al.*, "Event-based vision: A survey," 2019, *arXiv:1904.08405*. [Online]. Available: http://arxiv.org/abs/1904.08405

[21] G. Chen *et al.*, "A novel visible light positioning system with event-based neuromorphic vision sensor," *IEEE Sensors J.*, vol. 20, no. 17, pp. 10211–10219, Sep. 2020.

[22] A. Amir *et al.*, "A low power, fully event-based gesture recognition system," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7243–7252.

[23] G. Chen, L. Hong, J. Dong, P. Liu, J. Conradt, and A. Knoll, "EDDD: Event-based drowsiness driving detection through facial motion analysis with neuromorphic vision sensor," *IEEE Sensors J.*, vol. 20, no. 11, pp. 6170–6181, Jun. 2020.

[24] S. Misra and R. Hussain Laskar, "Comparative framework for vision-based gesturing modes and implementation of robust colour-marker detector for practical environments," *IET Image Process.*, vol. 13, no. 9, pp. 1460–1469, Jul. 2019.

[25] J. Liu, A. Shahroudy, G. Wang, L.-Y. Duan, and A. C. Kot, "Skeleton-based online action prediction using scale selection network," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 6, pp. 1453–1467, Jun. 2020.

[26] M. Abavisani, H. R. V. Joze, and V. M. Patel, "Improving the performance of unimodal dynamic hand-gesture recognition with multimodal training," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1165–1174.

[27] S. K. Leem, F. Khan, and S. H. Cho, "Detecting mid-air gestures for digit writing with radio sensors and a CNN," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 4, pp. 1066–1081, Apr. 2020.

[28] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4489–4497.

[29] H. Cheng, L. Yang, and Z. Liu, "Survey on 3D hand gesture recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 9, pp. 1659–1673, Sep. 2016.

[30] L. Wang, D. Q. Huynh, and P. Koniusz, "A comparative review of recent kinect-based action recognition algorithms," *IEEE Trans. Image Process.*, vol. 29, pp. 15–28, Jul. 2019

[31] P. K. Pisharady and M. Saerbeck, "Recent methods and databases in vision-based hand gesture recognition: A review," *Comput. Vis. Image Understand.*, vol. 141, pp. 152–165, Dec. 2015.

[32] J. H. Lee *et al.*, "Real-time gesture interface based on event-driven processing from stereo silicon retinas," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 12, pp. 2250–2263, Dec. 2014.

[33] X. Lagorce, G. Orchard, F. Galluppi, B. E. Shi, and R. B. Benosman, "HOTS: A hierarchy of event-based time-surfaces for pattern recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 7, pp. 1346–1359, Jul. 2017.

[34] A. Sironi, M. Brambilla, N. Bourdis, X. Lagorce, and R. Benosman, "HATS: Histograms of averaged time surfaces for robust event-based object classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1731–1740.

[35] E. Mueggler, C. Bartolozzi, and D. Scaramuzza, "Fast event-based corner detection," in *Proc. Brit. Mach. Vis. Conf.*, 2017, pp. 1–8.
[36] G. Chen *et al.*, "Neuroaed: Towards efficient abnormal event detection in visual surveillance with neuromorphic vision sensor," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 923–936, Sep. 2020.
[37] R. Xiao, H. Tang, Y. Ma, R. Yan, and G. Orchard, "An event-driven categorization model for AER image sensors using multispike encoding and learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 9, pp. 3649–3657, Sep. 2020.
[38] J.-M. Maro and R. Benosman, "Event-based gesture recognition with dynamic background suppression using smartphone computational capabilities," 2018, *arXiv:1811.07802*. [Online]. Available: http://arxiv.org/abs/1811.07802
[39] Y. Bi, A. Chadha, A. Abbas, E. Bourtsoulatze, and Y. Andreopoulos, "Graph-based spatio-temporal feature learning for neuromorphic vision sensing," *IEEE Trans. Image Process.*, vol. 29, pp. 9084–9098, Sep. 2020.
[40] H. Yujie, L. Jie, and Y. Shi, "A multi-condition relighting with optimal feature selection to robust face recognition with illumination variation," *China Commun.*, vol. 11, no. 6, pp. 99–107, Jun. 2014.
[41] Y. Cheng, L. Jiao, Y. Tong, Z. Li, Y. Hu, and X. Cao, "Directional illumination estimation sets and multilevel matching metric for illumination-robust face recognition," *IEEE Access*, vol. 5, pp. 25835–25845, 2017.
[42] M. de La Gorce, D. J. Fleet, and N. Paragios, "Model-based 3D hand pose estimation from monocular video," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1793–1805, Sep. 2011.
[43] B. Wang, Y. Zhao, and C. L. Philip Chen, "Moving cast shadows segmentation using illumination invariant feature," *IEEE Trans. Multimedia*, vol. 22, no. 9, pp. 2221–2233, Sep. 2020.
[44] X. Hu, L. Zhu, C.-W. Fu, J. Qin, and P.-A. Heng, "Direction-aware spatial context features for shadow detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7454–7462.
[45] A. Censi, J. Strubel, C. Brandli, T. Delbruck, and D. Scaramuzza, "Low-latency localization by active LED markers tracking using a dynamic vision sensor," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Nov. 2013, pp. 891–898.
[46] Q. De Smedt, H. Wannous, and J.-P. Vandeborre, "Skeleton-based dynamic hand gesture recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2016, pp. 1–9.
[47] E. P. Ijjina and K. M. C, "Classification of human actions using pose-based features and stacked auto encoder," *Pattern Recognit. Lett.*, vol. 83, pp. 268–277, Nov. 2016.
[48] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[49] G. Chen *et al.*, "NeuroIV: Neuromorphic vision meets intelligent vehicle towards safe driving with a new database and baseline evaluations," *IEEE Trans. Intell. Transp. Syst.*, early access, Oct. 20, 2020, doi: 10.1109/TITS.2020.3022921.

Guang Chen (Member, IEEE) received the Ph.D. degree from the Faculty of Informatics, Technical University of Munich, Munich, Germany, in 2017.

He was a Senior Researcher with the Chair of Robotics, Artificial Intelligence and Real-Time Systems, Technical University of Munich, from 2016 to 2017. He is currently a Research Professor with Tongji University and a Senior Research Associate (Guest) with the Technical University of Munich. He is also leading the Intelligent Sensing, Perception and Computing Group, Tongji University.

Zhongcong Xu received the B.E. degree in vehicle engineering from Tongji University, Shanghai, China, in 2020. He is currently pursuing the Ph.D. degree in electrical and computer engineering with the National University of Singapore, Singapore.

His research interests include computer vision and video understanding.

Zhijun Li (Senior Member, IEEE) received the Ph.D. degree in mechatronics from Shanghai Jiao Tong University, Shanghai, China, in 2002.

Since 2017, he has been a Professor with the University of Science and Technology, Hefei, China. His current research interests include wearable robotics, teleoperation systems, nonlinear control, and neural network optimization.

Dr. Li has been the Co-Chair of the IEEE SMC TC B^2S and the IEEE-RAS TC on Neuro-Robotics Systems. He also serves as an Associate Editor of several IEEE TRANSACTIONS.

Huajin Tang (Senior Member, IEEE) received the B.Eng. degree from Zhejiang University, Hangzhou, China, in 1998, the M.Eng. degree from Shanghai Jiao Tong University, Shanghai, China, in 2001, and the Ph.D. degree from the National University of Singapore, Singapore, in 2005.

He is currently a Professor with Zhejiang University. His research interests include neuromorphic computing, neuromorphic hardware, and robotic cognition.

Sanqing Qu received the B.E. degree in vehicle engineering from Tongji University, Shanghai, China, in 2020, where he is currently pursuing the Ph.D. degree in vehicle engineering.

His research interests include computer vision, video understanding, and autonomous driving.

Kejia Ren received the B.E. degree in vehicle engineering from Tongji University, Shanghai, China, in 2019. He is currently pursuing the master's degree in robotics with Johns Hopkins University, Baltimore, MD, USA.

His research interests include robotics, state estimation, and optimization.

Alois Knoll (Senior Member, IEEE) received the Diploma (M.Sc.) degree in electrical/communications engineering from the University of Stuttgart, Stuttgart, Germany, in 1985, and the Ph.D. degree *(summa cum laude)* in computer science from the Technical University of Berlin, Berlin, Germany, in 1988.

Since 2001, he has been a Professor with the Department of Informatics, Technische Universität München (TU München), Munich, Germany.