



A Real-Time Rock-Paper-Scissor Hand Gesture Recognition System Based on FlowNet and Event Camera

Xuemei Xie^(✉), Shu Zhang, Jinjian Wu, Xun Xu, Guangming Shi,
and Jianyu Chen

School of Artificial Intelligence, Xidian University, Xi'an 710071, Shaanxi, China
xmxie@mail.xidian.edu.cn

Abstract. Gesture recognition is one of the most popular tasks in computer vision, where convolutional neural networks (CNNs) based method has obtained the state-of-the-art performance. However, it is generally acknowledged that CNNs need a large amount of data to achieve such performance. Event Camera is a kind of biologically inspired event-based camera, which can keep the information of moving objects and remove the redundant background data. In this paper, we set up a rock-paper-scissor hand gesture recognition system based on FlowNet and Event Camera. Event camera is used to acquire event data. Then we propose an algorithm for the proposed gesture recognition. Specifically, FlowNet2.0 is employed to extract the motion representation of the pre-processed visual data, and a CNN classification network is applied to recognize the symbols extracted according to the motion representation. As a comparison, we also apply the rock-paper-scissor gesture recognition algorithm on traditional camera. The experimental results show that the proposed system based on Event Camera gets better performance, and to some extent, weaken the dependence on the training data. The whole system achieves 94.0% out-of-sample accuracy and allows computation at up to 30 fps.

Keywords: Gesture recognition · Event camera · FlowNet2.0 · Convolutional neural network

1 Introduction

Gesture recognition provides a means of natural and intuitive interaction between human and machine, and has high theoretical significance and practical value. With the development of gesture recognition technology, it shows its application potential in the field of natural Human-Computer Interaction (HCI) technology. However, the existing gesture data is mostly taken by a normal RGB camera with frame rate of 30 frames per second (fps). This frame-based data has a motion blur problem in the case of relatively fast gesture motion, which

affects the gesture recognition performance. An naive solution to the problem of motion blur is increasing the frame rate of the normal camera, but the continuous image frames obtained by the high frame rate camera contain much static redundant information. The non-demand-driven shooting method records the complex background environment in the scene. Meanwhile, implicitly increasing the amount of training data for convolutional neural networks (CNNs) which have been demonstrated successfully on human gesture recognition.

Event Camera [4, 6, 12, 18] is a new type of biomimetic sensor, with the advantages of removing redundant information, fast sensing capability, high dynamic range sensitivity and low power consumption. Compared to a normal frame-based camera, event camera acquires visual information in the completely different way, the frameless sampling allows continuous and asynchronous data in spatiotemporal domain. This demand-driven shooting method only record the illumination change caused by the gesture. The event stream generated in this way eliminates the redundant background, so low transmission bandwidth is needed. The microsecond time resolution of event camera ensures the continuity of gesture motion, without the limitation of exposure time and frame rate. In addition, event camera works well in the bright or dark environment due to its high dynamic range.

Since the advent of event camera, there have been multiple application scenarios in computer vision and robotics field. This paper mainly studies the dynamic gesture recognition method based on event camera. In this work, we set up a real-time rock-paper-scissor hand gesture recognition system. The rock-paper-scissor gesture is a good preliminary application, its interesting operation and presence of competing make it popular all over the world. Firstly, dynamic gesture data are gotten by using the event camera DAVIS240C to capture the preset gesture type. Secondly, the data conversion of three-dimensional event stream is needed. Thirdly, the converted data is segmented to extract meaningful gestures using FlowNet2.0 which estimate optical flow based on CNN structure. Finally, another CNN network is used to extract meaningful gesture features for final gesture recognition. The experimental results show that the performance of event camera is more accurate than normal RGB camera, further indicating that the event camera weaken the dependence on training data to a certain extent, and compared with normal RGB camera, can get the comparable results with less data.

2 Related Work on Event Camera Based Gesture Recognition

Owing to the complexity and diversity of hand gesture, most studies define specific gesture based on their own research purposes. In order to accurately recognize event streams, researchers use different methods for their specific gestures. The gesture recognition based on event camera can roughly be divided into three categories: specific hand shape recognition, gesture motion trajectory recognition and gesture motion change recognition.

The following works are specific hand shape recognition where the hand shape is fixed and shook slightly in front of the event camera. Rivera-Acosta et al. [19] convert the American sign language gesture events into images and digital image processing is applied to reduce noise, detect contour, extract and adjust characteristic pixel of contour. The Artificial Neural Network is used to classification and achieve 79.58% accuracy on 720 samples of 24 gestures. Lungu et al. [14] propose a method to recognizes the rock-paper-scissor gesture. Events are collected by slightly shaking fixed hand shape and accumulated into fixed event-number frames. These frames are fed into a CNN, which yielding 99.3% accuracy on 10% test data.

The following works are gesture motion trajectory recognition where hand shape is unchanged but hand position changes over time. Amir et al. [3] present the first gesture recognition system combined TrueNorth neurosynaptic processor with event camera and propose the first and the only one gesture event dataset(DvsGesture). A sequence of snapshots of event stream are collected and these concatenated snapshots are feed into CNN. Lee et al. [11] describes a gesture interface based on a stereo pair of event camera. The motion trajectory is detected by using leaky integrate-and-fire(LIF) neurons. Sixteen feature vectors are extracted from each spotted trajectory and classified by hidden Markov model gesture models. Achieved ranged from 91.9% to 99.5% accuracy depending on subject.

The following works are gesture motion change recognition where both hand shape and hand position change over time. Park et al. [17] propose a demosaicing method based on event-based images that offers substantial performance improvement of far-distance gesture recognition based on CNN. Ahn et al. [2] classify rock-paper-scissor gestures. Events within 20ms are converted to a frame. Events number in one frame less than the given threshold can be regarded as delivery point (key frame is used in this paper). The statistical features like distribution of width within the hand or the number of connected components for each segment are extracted and achieving 89% accuracy on 60 rock-paper-scissor gestures.

3 Event Camera

The event camera used in this paper is the Dynamic and Active Pixel Vision Sensor(DAVIS) [4] which is produced by INIVATION company. More specifically, DAVIS240C of DAVIS series product is used, and the device is shown in Fig. 1(a). DAVIS combines Dynamic Vision Sensor (DVS) and CMOS Active Pixel Sensor (APS) technology to output both asynchronous event data and simultaneous image frames. DAVIS has independent and asynchronous pixels that in response to illumination change. The pixel schematic of DAVIS is shown in Fig. 1(b).

We only use the event data here, so we focus on introducing the intensity response principle of DVS, as shown in Fig. 2. Each pixel output a event whenever the logarithmic illumination change (V_p) greater than the user-defined threshold and output a stream of events over time. Polarity attribute is contained in

each Event. Increasing in illumination is referred as ON event and decreasing in illumination is referred as OFF event.

The commonly used notation for an event is as follows:

$$e = [x, y, t, p]^T \tag{1}$$

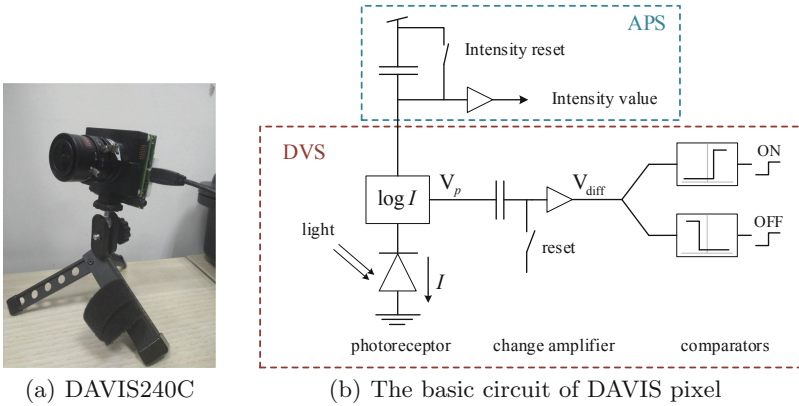


Fig. 1. Event camera and its pixel schematic

In which the event e indicates that the pixel located at $[x, y]^T$ on the pixel array of event camera output a event due to an illumination change at time t . The polarity attribute is encoded as $p \in [-1, 1]$, in which $p = 1$ is referred as ON event and $p = -1$ is referred as OFF event. It is worth mentioning that, events are conveyed at a temporal resolution of $1 \mu s$ and the event data rate is depend on the illumination change rate in the scene.

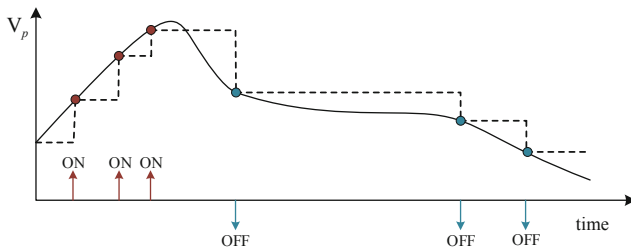
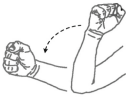




Fig. 2. Intensity response principle of DVS

4 Framework and System Components

Dynamic gestures usually have three phases of motion: preparation, stroke, and retraction [9]. The main information of gesture is mainly contained in the temporal sequence of the stroke phase. For the rock-paper-scissor gesture in this paper, there are also three phases, as shown in Table 1. The gestures in the preparation phase are consistent and cannot be distinguished. Key gesture is contained in stroke phase. The retraction phase is the transition phase to the next preparation phase. Therefore, it is necessary to divide the continuous dynamic gesture sequence, extract the key gesture, and then recognize the key gesture.

Table 1. Description of Gesture action

Three motion phases	Description of three phases
	Preparation: keep the fist form of hand, move the arm down.
	Stroke: delivery gesture(rock, paper or scissor) is needed to decide before reaching the lowest point. After reaching the lowest point, hand stays awhile.
	Retraction: the arm is pulled back up and the hand is slowly returned to the fist form of preparation.

How to deal with asynchronous event stream is the key basis for dynamic gesture recognition. Event stream is essentially a low-level visual spike signal. A single Event carries very limited information. A joint representation of multiple events can represent more descriptive information about the visual scene. Therefore, the three-dimensional event stream is mapped to a two-dimensional plane and low-level spike signal is converted to structural features. Our gesture recognition system has three major components, as shown in Fig. 3.

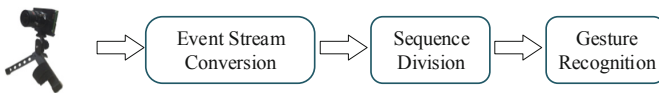


Fig. 3. Architecture of the proposed framework

4.1 Event Stream Conversion

There are two conversion methods. One is time-type conversion, which is related to time interval T . The other one is quantity-type conversion, which is related to the number of events N_e . The two conversion methods are shown in Fig. 4. It can be seen from Fig. 4(b) that the quantity-type conversion method cannot meet the real-time requirement because it does not consider the event occurrence time. Therefore, the time-type conversion method is chosen in this paper.

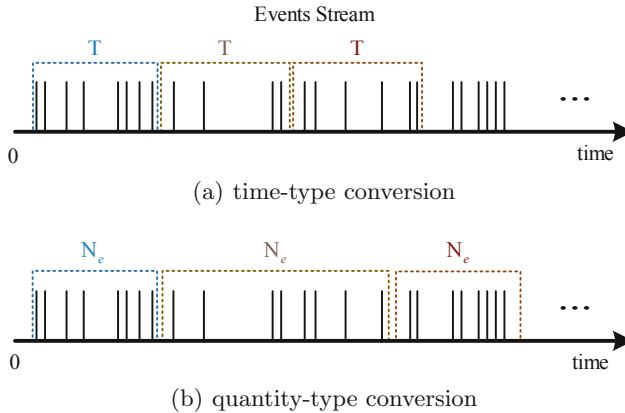


Fig. 4. Two conversion methods

A time interval value T is determined according to the gesture data to construct a sequence of images. The accumulated image frames of three different T values are shown in Fig. 5. The artifacts can be clearly seen in images of 30 ms. The artifact phenomenon is weakened, but it still exists in images of 20 ms. The artifact problem can be solved in images of 10 ms. It is more appropriate to choose the T value of 10 ms.

The accumulated Events image and the image of ordinary RGB camera are shown in Fig. 6. Further highlight the characteristic of event camera, with no redundant data. The dynamic gesture data of 11 subjects is collected using libcaer [1] software for about 60 s, and directly converted to AVI videos with a frame rate of 100.

4.2 Sequence Division

Optical Flow Estimation. The motion information caused by gesture movement is an important factor in image content changes. To reflect the motion information of the gesture, the motion of each pixel in the image can be described by optical flow. The purpose of optical flow is to approximate the spatial motion field that cannot be directly obtained from the image sequence.

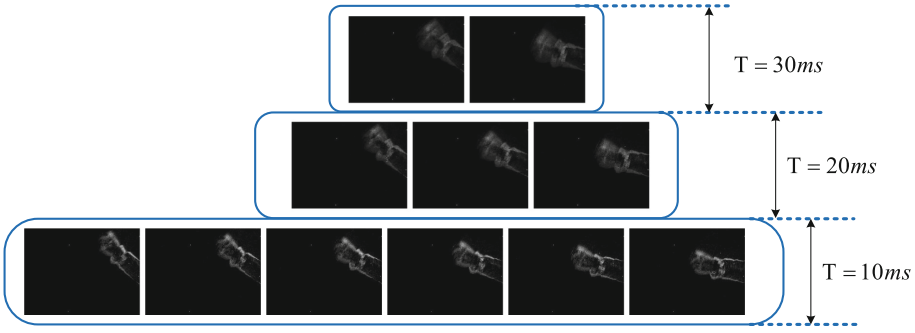


Fig. 5. Accumulated images for three T value



Fig. 6. Accumulated image (left) and RGB image (right)

Traditional optical flow algorithms [5, 13, 15, 16] based on the main assumption that the invariance of light intensity under small movements are not applicable in real life. In general, traditional optical flow is difficult to achieve in both real-time and accuracy. Recently, state of the art deep learning optic flow FlowNet2.0 [7] focus on high-level motion and can be used for practical applications that require accurate and fast optical flow computation which can be applied in real-time scenarios.

It is necessary to verify whether the converted events image can be directly applied to the FlowNet2.0. The converted Events images are fed into FlowNet2.0, and the output results are shown in Fig. 7. It can be seen from the figure that the optical flow images have a clear boundary contour. Thus, the converted events images can be directly applied to FlowNet2.0.

Decision Module. It can be seen from Fig. 7 that the optical flow in y direction can be used to extract key gestures. The vertical optical flow information obviously and intuitively reflects the three stages of rock-paper-scissor. During the preparation phase, the moving speed is faster, the speed of stroke phase is gradually reduced until the hand is static, and then the speed gradually increases during the retraction phase.

In decision module, we use the changes of motion phases in velocity magnitude to get the frames contained desired hand posture. The magnitude of optical flow represents motion speed value of each pixel between adjacent frames. Considering

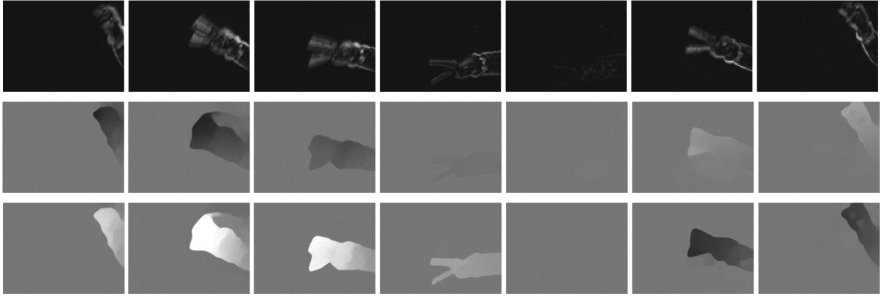


Fig. 7. The horizontal optical flow (second row) and the vertical optical flow (third row) for sample adjacent frames of “scissor” gesture (first row). The brighter or darker color in optical flow images means the greater velocity value.

the hand motion, two thresholds are set, T_v and T_n . When the number of magnitude in the interval $[-T_v, T_v]$ reaches T_n , the key frames are found.

Classification Network. After getting the key frames of the gesture, we take these key frames to train the CNN classification network. In the network designing, we employ a network proposed in [14] for rock-paper-scissor classification. This network is the enhanced version of LeNet, which consists of 5 convolutional layers and 1 fully connected layer. Each convolutional layer followed by RELU activation function and 2×2 max pooling. The structure of CNN network is shown in Fig. 8.

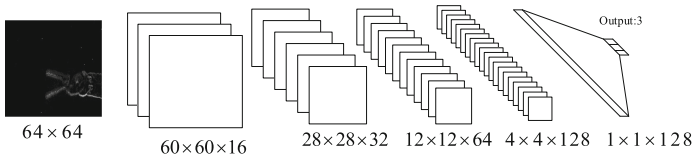


Fig. 8. Classification network structure.

Same classification task with [14], three symbols in rock-paper-scissor game are classified by the classification network. However, events are collected by slightly shaking fixed hand shape and accumulated into fixed event-number frames in [14].

5 Experiments

5.1 Results of Sequence Division

For the experimental setup of the decision module, if the number of key frames satisfying the decision condition is greater than 15, only 15 key frames are

acquired, and vice versa, all key frames satisfying the decision condition are acquired. The first key frame is used as a judgment to determine whether the true key gesture image is lost. First, the T_n is fixed, and it is set to 1000. The key gesture image extraction result of different T_v is compared on 11 collected rock-paper-scissor converted events video. The results are shown in Table 2.

Table 2. Key gesture images of different T_v

T_v	T_n	Experimental data number										Missing number	
		1	2	3	4	5	6	7	8	9	10		11
3	1000	59	63	67	61	76	55	97	63	42	60	72	8
4	1000	59	59	67	61	72	55	93	61	42	60	75	19
5	1000	53	60	67	59	68	55	89	63	42	60	77	30
6	1000	56	62	67	60	66	55	93	63	42	58	74	19

* The red number in the table represent the missing key gesture image extraction results.

The red number in the table represent the missing key gesture image extraction results. It can be seen from the table that when the T_v is 3, the missing key gesture images are the least. Then, in the case of a fixed T_v , the key gesture image extraction results of different T_n are compared on 11 collected rock-paper-scissor converted events video. The results are shown in Table 3. As can be seen from the table, when the T_v is 3 and the T_n is 2000, the missing images are lost the least. Hence, T_v and T_n are set to 3 and 2000 respectively.

Table 3. Key gesture images of different T_n

T_v	T_n	Experimental data number										Missing number	
		1	2	3	4	5	6	7	8	9	10		11
3	1000	59	63	67	61	76	55	97	63	42	60	72	8
3	2000	59	63	67	61	76	55	97	63	42	60	75	5
3	3000	59	63	67	61	74	55	89	63	42	60	74	8
3	5000	59	62	66	55	73	55	88	42	36	60	21	106

* The red number in the table represent the missing key gesture image extraction results.

5.2 Results of Classification

Meanwhile, in order to stress the advantages of event-based pattern, we also use the traditional HIKVISION DS-2CD5A26FWD-IZ camera to collect the rock-paper-scissor gesture videos in different background. Then these video streams are fed into proposed key frames extraction module. The key frames extracted

from that module are collected and finally we get 5205 images and 2539 images for DVS data and HIK data respectively. The difference in the number of two data sets is due to the equivalent frame rate of DVS video data different from the HIK videos. Then these images are divided into training dataset and out-of-sample validation dataset for the classification network. After that, we build up the unique DVS rock-paper-scissor static symbol dataset with accumulated events and the HIK rock-paper-scissor static symbol dataset.

It must be mentioned that, the storage capacity of DVS and HIK static symbol dataset are 32.6 MB and 381.8 MB respectively. DVS effectively remove the redundant background data, which enables efficient and low-power object recognition.

To verify the effectiveness of the 6-layer network used in this paper, we train a 6-layer model, and a AlexNet [10] model respectively on the dataset we collected. Besides, to evaluate the performance of the CNN network, images in 9 videos are selected as the training data, and the remaining images in 2 videos are reserved as out-of-sample validation. Considering the DVS dataset is gray-scale image, the strategies of training network on DVS data are learned from LeNet. Pixel values of DVS dataset is normalized to 0–1 range. For 6-layer network, the batch size is set to 8, and the base learning rates of DVS data and HIK data are 0.01 and 0.001 respectively. For AlexNet, the batch size is set to 16, and the base learning rates setting is the same as 6-layer network. The learning rate policy varies by data and network, e.g. 6-layer network for DVS data decrease the learning rate at iterations of 3000, 15000, 30000, 40000, and AlexNet for DVS data decrease the learning rate at iterations of 3000, 6000, 10000, 15000, 20000. All experiments are implemented with Caffe [8]. Our computer is equipped with Intel Core i7-6700 CPU with frequency of 3.4GHz, NVidia GeForce GTX Titan XP GPU, 128 GB RAM, and the framework runs on the Ubuntu 16.04 operating system.

We compare the performances of two limited training data on 6-layer and AlexNet network. The performances are reported in Table 4. In order to avoid the influence caused by random once selection of out-of-sample validation data, we randomly select 5 groups of 9 training data and 2 out-of-sample validation among the 11 videos. The final result is the average result of the 5 groups.

Table 4. Performance comparison with two limited training data on 6-layer and AlexNet network

Model	Data	Input	Out-of-sample acc.	Params.num	Net.forward (ms)
6-layer	DVS	64×64	0.940	114K	0.67
6-layer	HIK	64×64	0.662	115K	0.73
AlexNet	DVS	227×227	0.944	57M	8.96
AlexNet	HIK	227×227	0.668	57M	9.77

As shown in Table 4, the out-of-sample performance of DVS data on 6-layer network is better than HIK data, which benefit from eliminating redundant data

and only keep the useful information. The performance of HIK data on 6-layer is unsatisfactory, largely because of limited amount of training data for complex backgrounds. Without enough training data, CNN cannot extract good features that are important for discrimination. It also makes the network not robust to image background interference, as poor performance is illustrated on data that the network has not seen before.

Besides, the DVS and HIK data show almost the same performance on AlexNet. Compared to AlexNet, the performance of 6-layer network is comparable. Moreover, 6-layer network has less parameters, and faster processing speed. The parameters in 6-layer network and AlexNet also shows that there is a lot of representation redundancy in AlexNet for DVS data. These experiments prove the effectiveness of the 6-layer on limited DVS training data used in this paper.

6 Conclusion

This paper proposes a system to classify dynamic gestures using event camera. Specifically, we focus on classifying three symbols in the rock-paper-scissor game, which is an preliminary application and the method can be extended to other real world applications. Combing DVS and CNNs to solve gesture recognition problem under complicated background, which also weaken the requirements for training data, to some extent. Different from other end-to-end gesture recognition without specific hand posture recognition. We utilize the FlowNet 2.0 to extract motion representation of the accumulated event frames, and motion representation is used to get the key frames. Then a 6-layer classification network is applied to recognize the symbols in key frames, which has very little time consumption.

However, the gesture recognition method in this paper still has some shortcomings. Since the two hard thresholds in decision module would inevitably result in data loss. This inflexible threshold setting method will make the extraction varies from person to person. And Converting the events into images dilute the asynchronous characteristic of events. In the future, a more efficient extraction of key gesture and a more rational use the asynchronous characteristics of events data will propose.

Acknowledgments. This work was supported by Young Fund for High Resolution Earth Observation Conference, Young Star Science and Technology Project (No. 2018KJXX-030) in Shanxi province.

References

1. <https://github.com/inilabs/libcaer>
2. Ahn, E.Y., Lee, J.H., Mullen, T., Yen, J.: Dynamic vision sensor camera based bare hand gesture recognition. In: 2011 IEEE Symposium On Computational Intelligence For Multimedia, Signal And Vision Processing, pp. 52–59. IEEE (2011)

3. Amir, A., et al.: A low power, fully event-based gesture recognition system. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7243–7252 (2017)
4. Brandli, C., Berner, R., Yang, M., Liu, S.C., Delbruck, T.: A 240×180 130 db $3 \mu\text{s}$ latency global shutter spatiotemporal vision sensor. *IEEE J. Solid-State Circuits* **49**(10), 2333–2341 (2014)
5. Horn, B.K., Schunck, B.G.: Determining optical flow. *Artif. Intell.* **17**(1–3), 185–203 (1981)
6. Huang, J., Guo, M., Chen, S.: A dynamic vision sensor with direct logarithmic output and full-frame picture-on-demand. In: 2017 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1–4. IEEE (2017)
7. Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., Brox, T.: Flownet 2.0: evolution of optical flow estimation with deep networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2462–2470 (2017)
8. Jia, Y., et al.: Caffe: convolutional architecture for fast feature embedding. In: Proceedings of the 22nd ACM International Conference on Multimedia, pp. 675–678. ACM (2014)
9. Kendon, A.: Current issues in the study of gesture. In: *The Biological Foundations of Gestures: Motor and Semiotic Aspects*, vol. 1, pp. 23–47 (1986)
10. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems*, pp. 1097–1105 (2012)
11. Lee, J., et al.: Live demonstration: gesture-based remote control using stereo pair of dynamic vision sensors. In: 2012 IEEE International Symposium on Circuits and Systems, pp. 741–745. IEEE (2012)
12. Lichtsteiner, P., Posch, C., Delbruck, T.: A 128×128 120 db $15 \mu\text{s}$ latency asynchronous temporal contrast vision sensor. *IEEE J. Solid-State Circuits* **43**(2), 566–576 (2008)
13. Lucas, B.D., Kanade, T., et al.: An iterative image registration technique with an application to stereo vision (1981)
14. Lungu, I.A., Corradi, F., Delbrück, T.: Live demonstration: convolutional neural network driven by dynamic vision sensor playing roshambo. In: 2017 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1–1. IEEE (2017)
15. Nagel, H.H.: Displacement vectors derived from second-order intensity variations in image sequences. *Comput. Vision Graph. Image Proc.* **21**(1), 85–117 (1983)
16. Nagel, H.H.: On the estimation of optical flow: relations between different approaches and some new results. *Artif. Intell.* **33**(3), 299–324 (1987)
17. Park, P.K., et al.: Performance improvement of deep learning based gesture recognition using spatiotemporal demosaicing technique. In: 2016 IEEE International Conference on Image Processing (ICIP), pp. 1624–1628. IEEE (2016)
18. Posch, C., Matolin, D., Wohlgenannt, R.: A QVGA 143 dB dynamic range frame-free PWM image sensor with lossless pixel-level video compression and time-domain CDS. *IEEE J. Solid-State Circuits* **46**(1), 259–275 (2011)
19. Rivera-Acosta, M., Ortega-Cisneros, S., Rivera, J., Sandoval-Ibarra, F.: American sign language alphabet recognition using a neuromorphic sensor and an artificial neural network. *Sensors* **17**(10), 2176 (2017)