

Research Article

Dynamic Vision Sensor Tracking Method Based on Event Correlation Index

Hengyi Lv ¹, Yang Feng,^{1,2} Yisa Zhang,^{1,2} and Yuchen Zhao¹

¹Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China

²College of Materials Science and Opto-Electronic Technology, University of Chinese Academy of Sciences, Beijing 100049, China

Correspondence should be addressed to Hengyi Lv; lv_hengyi@163.com

Received 4 August 2020; Revised 5 September 2020; Accepted 15 April 2021; Published 27 April 2021

Academic Editor: Átila Bueno; atila.bueno@unesp.br

Copyright © 2021 Hengyi Lv et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Dynamic vision sensor is a kind of bioinspired sensor. It has the characteristics of fast response, large dynamic range, and asynchronous output event stream. These characteristics make it have advantages that traditional image sensors do not have in the field of tracking. The output form of the dynamic vision sensor is asynchronous event stream, and the object information needs to be provided by the relevant event cluster. This article proposes a method based on the event correlation index to obtain the object's position, contour, and other information and is compatible with traditional tracking methods. Experiments show that this method can obtain the position information of the moving object and its continuous motion trajectory and analyze the influence of the parameters on the tracking effect. This method will have broad application prospects in security, transportation, etc.

1. Introduction

As an important research direction in the field of computer vision, object tracking is widely used in security, traffic, and unmanned driving [1–3]. The image sensors we use for object tracking daily are imaging in frames. This imaging method, although the output image is intuitive, is more pleasing to the eye. However, this fixed frame rate imaging method will lose the object information between frames during object tracking or blur because the object moves too fast, which will affect the accuracy of object tracking. And, a faster frame rate will bring higher power consumption and require more storage to store data.

In order to solve these problems, with the development of bionics, the dynamic vision sensor (DVS) emerged [4–9], which is a new type of image sensor that generates event flow based on changes in light intensity. The principle of the sensor is similar to the retina. Its pixel structure is shown in Figure 1. The pixel structure is composed of three parts [10–12]. The first part is a current-voltage logarithmic conversion light sensor circuit for sensing light intensity. Photoelectric conversion: this part is similar to cone cells in the retina. The second part is a variable amplifying circuit

similar to the bipolar cells in the retina, using a switched capacitor amplifying structure whose function is to complete sampling and amplification. The third stage is mainly composed of two comparators, which are similar to the ganglion cells in the retina. When the light intensity becomes weak, it generates an OFF signal, and when the light intensity becomes strong, it emits an ON signal.

The sensor images the position where the light intensity changes in the field of view, thus reducing the amount of data. Since there is no need to expose for charge accumulation, the light intensity change can be detected continuously, so there is a very high time resolution between events. Benefit from its logarithmic photoelectric conversion unit, the sensor has a large dynamic range of 120 dB [10].

However, due to the special imaging method and unique data format of dynamic vision sensors, traditional tracking algorithms are not applicable. Here, we propose a new method to calculate the event correlation index of each event and extract the location information and contour information of the object from it. This method not only reduces the dimensionality of the three-dimensional event stream but also uses existing tracking algorithms, such as centroid tracking algorithm. Compared to simply compressing the

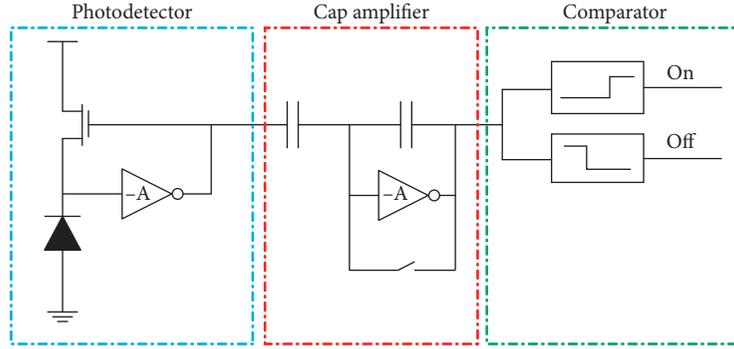


FIGURE 1: Schematic diagram of the pixel structure of the dynamic vision sensor. Photodetector is used for photocurrent-voltage logarithmic conversion, cap amplifier is used to amplify voltage changes, and comparator is used for threshold comparison.

three-dimensional event stream into a two-dimensional binary image, this method can preserve the spatiotemporal correlation of events, thereby reducing noise impact on tracking algorithms.

The rest of the paper is organized as follows: Section 2 reviews the work related to the dynamic vision sensor tracking method, Section 3 introduces the algorithm of this article, Section 4 will experiment and evaluate the method, and Section 5 will analyze the experimental results. In section 6, conclusion will be summarized.

2. Related Works

With the advancement of semiconductor design and technology, the resolution of dynamic vision sensors has been further improved, and the readout rate of the event stream has also been greatly increased. The resolution of the latest dynamic vision sensor from Prophesee in France has reached 1280×720 [13], and the maximum event readout rate reaches 1066 Meps. The Cele-X V DVS of China's CelePixel Technology Co., Ltd. has a resolution of 1280×800 and a readout rate of 160 Meps [14]. In this case, processing each event in the event stream in turn to determine whether it is an object is a great challenge to the calculation speed of the tracking system. Moreover, traditional tracking methods suitable for frames are difficult to use in event streams. Although each event in the event stream contains location information, a single event cannot effectively convey the information of the moving object, and it is even impossible to determine whether the event is generated by the object or noise. The events generated by the object have high temporal and spatial correlation, and only by using these events can we obtain the object's location information and time information.

Most of the existing tracking methods based on dynamic vision sensors use event clustering to extract object location information. The cluster determination depends on the distance between events and the number of events that are closer. The event distance in the cluster is less than a certain threshold, and the number of events is more than a certain threshold, which is defined as an object. In [15], a cluster-based method inspired by the traditional mean shift method has been used to track the arm of a robot football goalkeeper. Other work that uses cluster-based methods to

track moving objects is reflected in the paper [16] published by Schraml and Belbachir. Compared with the literature [15], Schraml's algorithm is different in the way the events are allocated to the cluster. The allocation of newly generated events depends on the 3D Manhattan distance in space and time between the event and the cluster. Compared with the traditional Euclidean distance, this clustering method can suppress noise. Because of the low memory usage, the cluster-based method is suitable for embedded vision systems, but the cluster size needs to be adjusted according to specific goals, so the above methods are only suitable for specific scenarios.

The method of clustering events based on the Gaussian mixture model (GMM) came into being, and these works are reflected in the literature [17, 18]. Piatkowska et al. call this method K-Gaussian clustering method [17]. In this algorithm, events are modeled by Gaussian clustering. Later, Lagorce et al. improved the method, in which the spatial distribution of events was modeled by bivariate Gaussian [18]. This is also inspired by the mean shift algorithm. Determine which cluster the event belongs to in the event stream, and then update the cluster.

Literature [19] proposed event coherence detection algorithm. This method divides the event stream into 32 or 64 blocks according to space, performs event correlation detection to extract the event clusters, and then matches the newly discovered object with objects in the tracker. However, when the object has a large geometric size and is on the boundary of space, the object will be divided into multiple by mistake, which affects the tracking effect.

This paper proposes a method that uses the event stream in a fixed period of time and the Gaussian kernel convolution method to compress the three-dimensional event stream into a two-dimensional image that retains the event correlation, so that the traditional image processing method can be used to extract the relevant events. Space coordinates are used to determine the location information of the object and then track the object. The advantage of this method is that it can directly obtain the location information of the event cluster and in turn obtain the event stream at that location. Not only can simple traditional image processing methods be used but also the event stream of the object can be retained to analyze the continuous motion trajectory and status.

3. Algorithms

Object tracking is the process of locating the position of the object in the subsequent frames according to the position of the object in the first frame of the video sequence. In the traditional method, first locate the position of the object in the first frame and then search around for the object that matches the previous frame in the subsequent frames. The dynamic vision sensor outputs the event stream asynchronously. The event stream contains the events generated by the object movement and the noise of the sensor itself. The correlation between events cannot be directly reflected in the event stream data, which makes it impossible to obtain object information directly from the event stream. Although each event contains its own location information, a single event cannot express the information of the moving object.

According to the above analysis, the object tracking method based on the dynamic vision sensor is divided into

two parts: (1) the event stream is sliced according to a fixed time period, the object detector is used in the slice to obtain the object position information, and the information of objects in the first slice is stored in the tracker; (2) match the objects in the tracker with the objects found in the subsequent event stream and update the tracker. We propose the following algorithm.

3.1. Object Detector

- (1) Collect the event stream ES of the time period T because a certain number of events are required to obtain the correlation of the events. The correlation of events is quantified by a two-dimensional Gaussian kernel and expressed by the event correlation index. The calculation method is

$$I_{e_i} = \sum_{j=1}^N \frac{1}{2\pi\sigma_1\sigma_2} e^{-\frac{1}{2} \left(\frac{(x_j - x_i)^2 + (y_j - y_i)^2}{\sigma_1^2} + \frac{(t_j - t_i)^2}{\sigma_2^2} \right)}, \quad (e_j(x_j, y_j, t_j) \in \text{ES} \cap x_j \neq x_i \cap y_j \neq y_i). \quad (1)$$

In equation (1), σ_1 and σ_2 are the standard deviations of the space distance and time distance, respectively. The coordinates and time of occurrence of the event are independent of each other, so $\rho = 0$ in the normal distribution, and $e_j(x_j, y_j, t_j)$ is the supporting event of $e_i(x_i, y_i, t_i)$.

- (2) After obtaining the event correlation index of each event in the event stream for this period, add all the

event correlation indexes of each pixel as the grey value and store it in the corresponding location in a two-dimensional matrix with the same size as the sensor resolution. In this way, the event correlation image ECI is obtained, and the pixel value of the image is

$$p_{ECI}(x_i, y_i) = \begin{cases} \sum I_{e_{(x_i, y_i, t)}} & \exists e(x_i, y_i, t) \in \text{ES}, \\ 0, & \text{otherwise,} \end{cases} \quad 1 \leq x_i \leq m, 1 \leq y_i \leq n, \quad (2)$$

where m and n are the horizontal and vertical resolution of the sensor, respectively.

Use OTSU [20] to adaptively acquire the threshold λ and binarize ECI according to equation (3) and divide the picture into two parts: the object and the background. The acquired threshold needs to be

judged with the minimum threshold. If the threshold is lower than the minimum threshold, it means that there is no event cluster whose correlation meets the requirements in the event stream, that is, there is no moving object.

$$p_{WB}(x_i, y_i) = \begin{cases} 1, & p_{ECI}(x_i, y_i) \geq \lambda, \\ 0, & p_{ECI}(x_i, y_i) < \lambda, \end{cases} \quad 1 \leq x_i \leq m, 1 \leq y_i \leq n, \quad (3)$$

where $p_{WB}(x_i, y_i)$ is grey value of pixel (x_i, y_i) in binary image.

- (3) In order not to lose events with a low correlation index of the object edge, the binary image needs to be dilated. The object edge event correlation index is lower. This is because there are more events near the

centre of the object than events near the edge, so the correlation index is lower than the centre.

- (4) At this point, the object position and contour information can already be obtained on the binary image, so that according to the contour range, use equation (4) to extract the object events OBJ^k in the event stream:

$$OBJ^k = \{e(x_i, y_i, t_i) | \min(O_k(y_i)) \leq x_i \leq \max(O_k(y_i)) \cap \min(O_k(x_i)) \leq y_i \leq \max(O_k(x_i))\}. \quad (4)$$

Here, $O_k(x, y)$ is the contour curve of object K. These events are the events generated by the object movement. According to equation (5), the object centroid (x, y) is obtained and used to update the tracker:

$$X_{objk} = \frac{\sum x_i}{N}, Y_{objk} = \frac{\sum y_i}{N}, \quad e(x_i, y_i, t_i) \in OBJ^k, \quad (5)$$

where N is the number of events in OBJ^k .

3.2. Tracker Update

$$ID(OBJ_{T_1}^k) = ID(OBJ_{T_2}^l) \leftrightarrow \begin{cases} \min(R_k(X_{objl})) \leq Y_{objk} \leq \max(R_k(X_{objl})) \\ \min(R_k(Y_{objl})) \leq X_{objk} \leq \max(R_k(Y_{objl})) \end{cases} \quad (6)$$

where $ID(OBJ^k)$ is the ID of object K, T_1 and T_2 are two adjacent consecutive time periods and the duration is T , and R_k is the bounding rectangle of the object.

- (3) When the object in the tracker fails to match the detected object for a long time, it will be deleted and the ID will no longer be used.

4. Experiments

The technical focus of the method in this paper is to extract the position and contour of the event cluster formed by the moving object in the event stream, so as to obtain the object events in the event stream. Therefore, in the experiment, the effect of obtaining the object contour from the event cluster will be tested. At the same time, the tracker matching effect will also be tested.

The experimental device uses DAVIS 346, the sensor resolution is 346×260 , the time resolution of the event is 1us, and the event format is $e(x_i, y_i, p_i, t_i)$, where x_i and y_i are the pixel coordinates of the event and p_i is the polarity of the event. The light intensity changes from dark to bright, the polarity is 1, and the opposite polarity is 0; t_i is the time stamp.

The data used in the experiment was acquired under natural light indoors. In order to simulate a small object moving at a high speed, the spot of the laser pointer was used to move quickly on the whiteboard, so that a fast moving point object was formed in the field of view.

The parameter settings in the algorithm are set as follows: time period $T=2$ ms, $\sigma_1=1$, $\sigma_2=0.5$, and the expanded structure element is a rectangle with a side length of 5. The object is not matched for more than 20 ms and will be deleted. Figure 2 is a spatiotemporal scatter plot of a 100 ms event stream, where the red points are positive and the blue points are negative. Although there are a lot of noise and hot pixels in the figure, it can be intuitively seen that the object event clusters are continuous. Hot pixels are blue lines in Figure 2, which are generated by pixels that send events incorrectly all the time. Visually, object event cluster can be found to have a strong correlation, but background activity in Figure 2 has no correlation with other events.

Through the algorithm in this paper, the event correlation index image is obtained, the contour position of the object is extracted, and the object is tracked. The correlation index image and object contour at some moments are shown in Figure 3:

The images on the left of Figure 3 are event correlation index images of 6 periods, each time period is 2 ms, and the

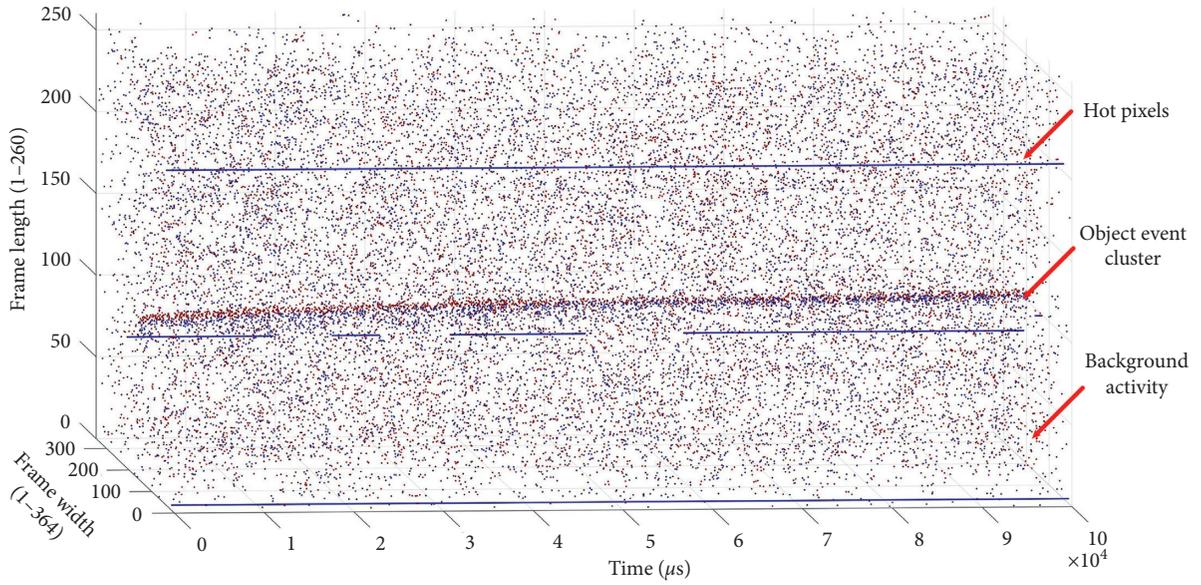


FIGURE 2: Three-dimensional scatter plot of moving object event stream. The horizontal dense event cluster is formed by the object movement. There is also hot pixel and background activity in event stream, which is marked in the figure. Hot pixels are blue lines, which are generated by pixels that send events incorrectly all the time.

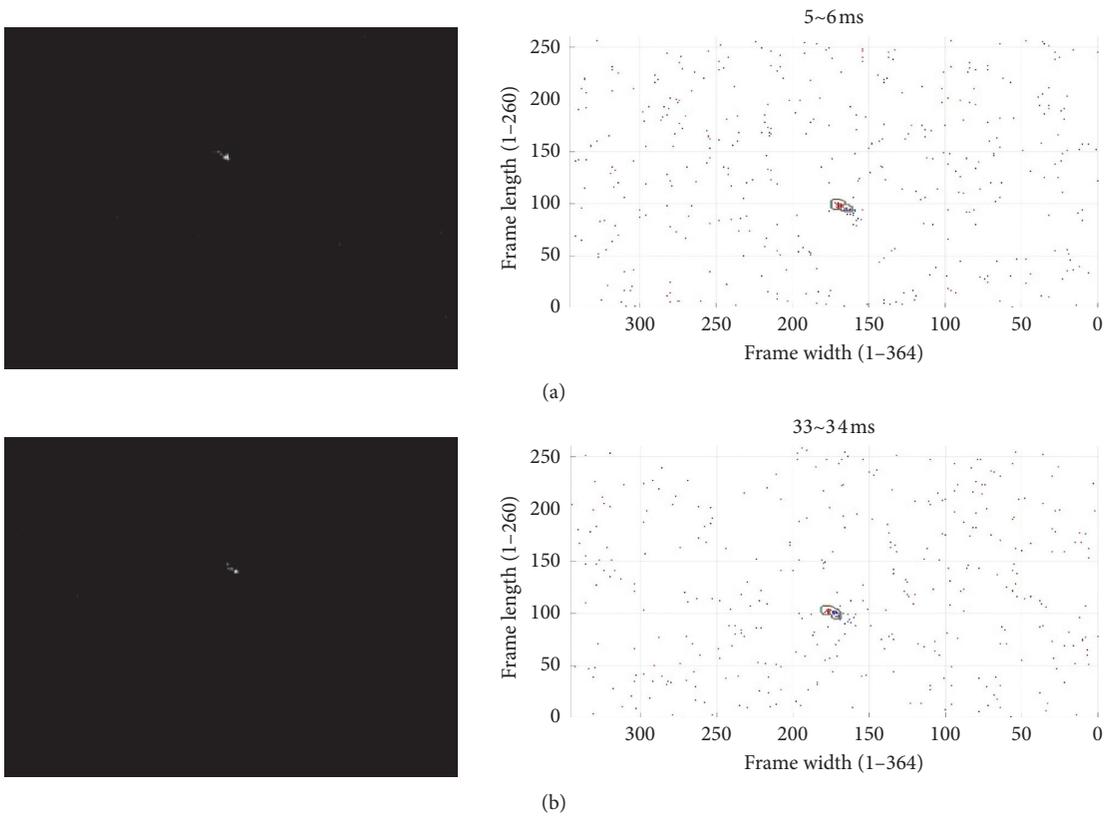


FIGURE 3: Continued.

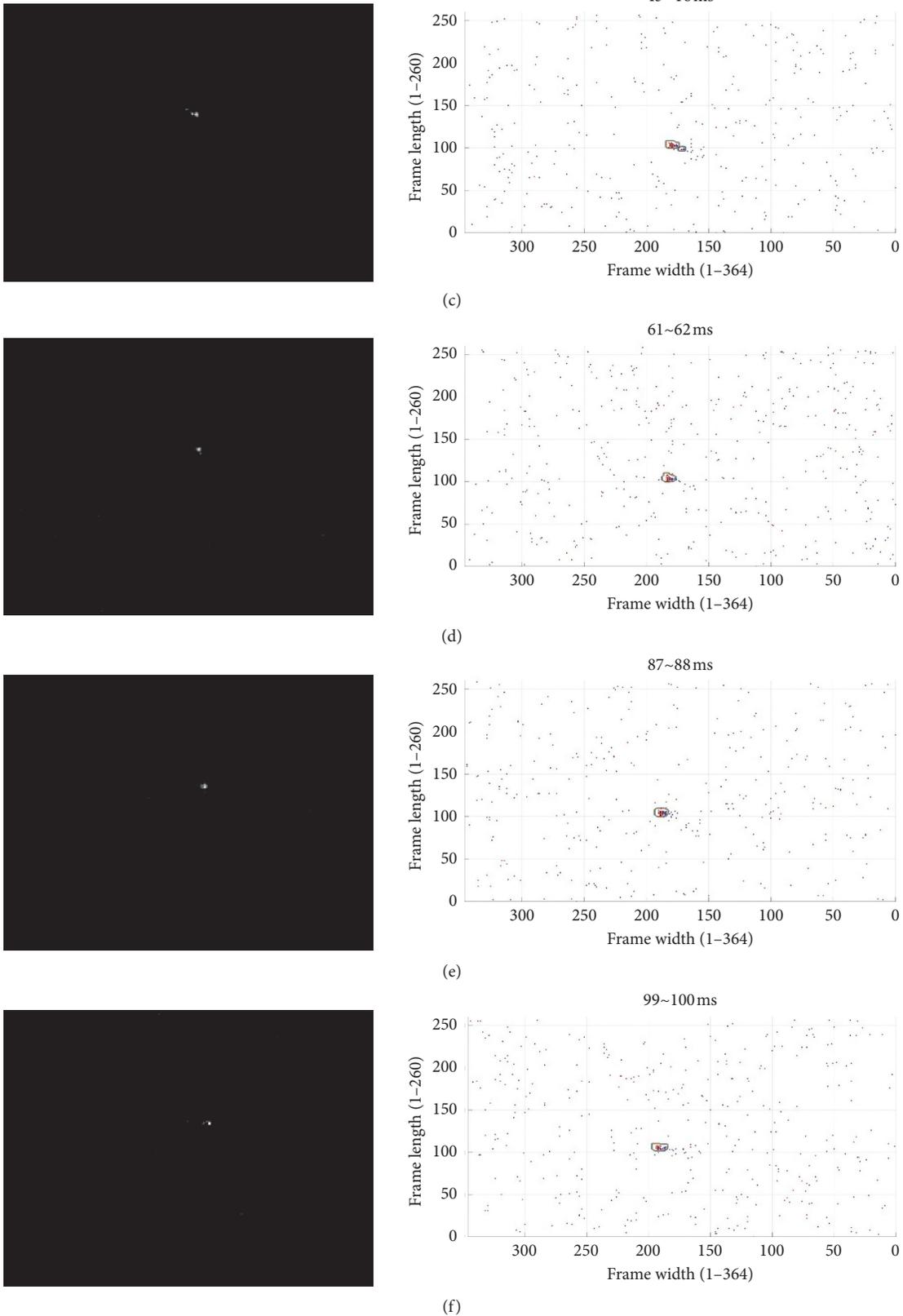


FIGURE 3: Event correlation index diagrams for some time periods (left) and 3D spatiotemporal diagrams of event flow (right). These figures are visualizations of the event streams in six time periods, and the black boxes in the right figures mark the outline information of the object.

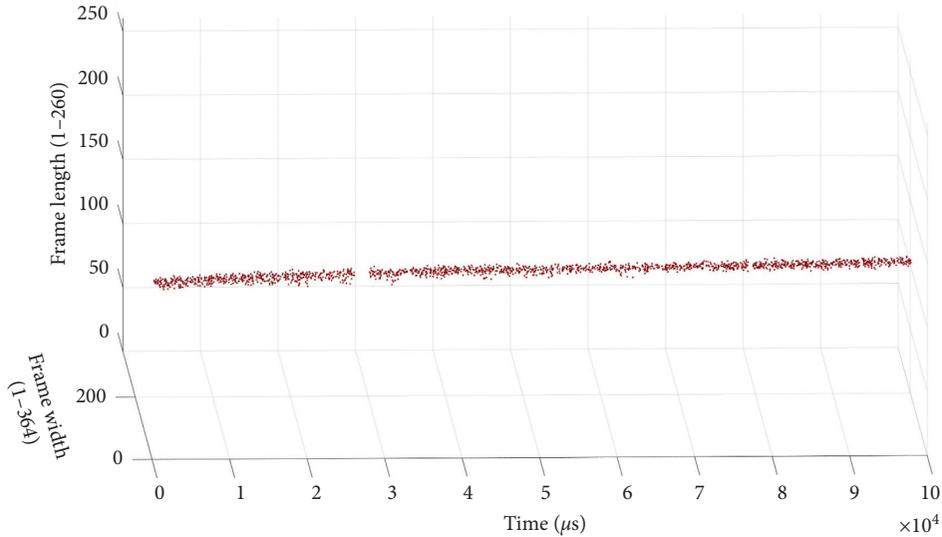


FIGURE 4: Object tracking effect diagram, the red part is the event stream of the object, every red dot is an event of the object, and events cluster shows trajectory of the target.

grey scale of the image represents the correlation strength of the event generated by the pixel, which means that there is a target at the location that has generated an event. The images on the right are diagrams of the event stream in a 2 ms time period, in which the time axis is facing the inside of the screen, only look at the spatial position of the event, and the target events can be marked with a box. It can be seen that the correlation of events within the outline is significantly higher than that outside the outline.

Figure 4 is the tracking effect diagram of the tracker. The red dots are the acquired events of the object. It can be seen from the figure that the event stream of the object is completely retained, and the trajectory is clearly and coherently in the three-dimensional space-time image. It can provide support for the subsequent object's motion trajectory analysis. The events of the target extracted by the algorithm in the original event stream are marked. The left image in Figure 5 is the original event stream, from which the position and shape of the target can be seen in the centre of figure. In the figure on the right, the red events are the target events extracted by the algorithm, and the blue events are the original events. Figure 5 shows that the algorithm can accurately obtain the target position and shape.

5. Analysis

The tracking method based on the event correlation index focuses on selecting the variance of the Gaussian kernel, and the appropriate variance should highlight the relevant events. According to the characteristics of the Gaussian kernel, the larger the σ , the larger the range of the Gaussian kernel, and more events support the central pixel.

When the space width parameter σ_1 in the Gaussian kernel is unchanged, the change of the time width parameter σ_2 will affect the discovery of the object and the determination of the object contour range; when σ_2 is smaller, the correlation of the event in the time dimension will be ignored. A small

amount of noise that is close in time will also get a higher correlation index, which will cause the discovery of false objects. When σ_2 is larger, more previous events will provide support, but the events on the edge of the object do not have the support provided by previous events, and then the correlation index of these events is lower than that inside the object, and the object contour will be smaller than the actual object. Figure 6 is a three-dimensional image of Gaussian kernels with different σ_2 when is 1, and the three images in Figure 7 are the results of obtaining the object event in a 2 ms time period when σ_2 is 0.1, 0.5, and 1, respectively. When $\sigma_2 = 0.1$, there are false objects caused by noise on the left side of the figure. When $\sigma_2 = 0.5$, the obtained object contour contains the events generated by the object. When $\sigma_2 = 1$, the obtained object contour is small, and it fails to include the sparse events generated by the object edge.

When the time width parameter σ_2 in the Gaussian kernel is unchanged, the change of the space width parameter σ_1 will also affect the determination of the object contour range. When the space width is small, the correlation index provided by the support events of different space distances is not much different and the value is small, which is not conducive to calculating the threshold. Therefore, the space width should be greater than one pixel, and the support events of different distances provide different correlations. In this way, a meaningful correlation index can be obtained, and too large space width will increase the interference of noise on object discovery. Figure 8 shows the three-dimensional images of Gaussian kernels with different d_1 when σ_2 is 0.5. The three images in Figure 9 are the results of obtaining the object event in a 2 ms time period when σ_1 is 0.5, 1, and 2. When $\sigma_1 = 0.5$, because the correlation index is too small to determine the appropriate threshold, a lot of noise is obtained. When $\sigma_1 = 1$, the object contour obtained includes the events generated by the object. When $\sigma_1 = 2$, affected by the hot pixels, a false object is obtained at the bottom right of the picture.

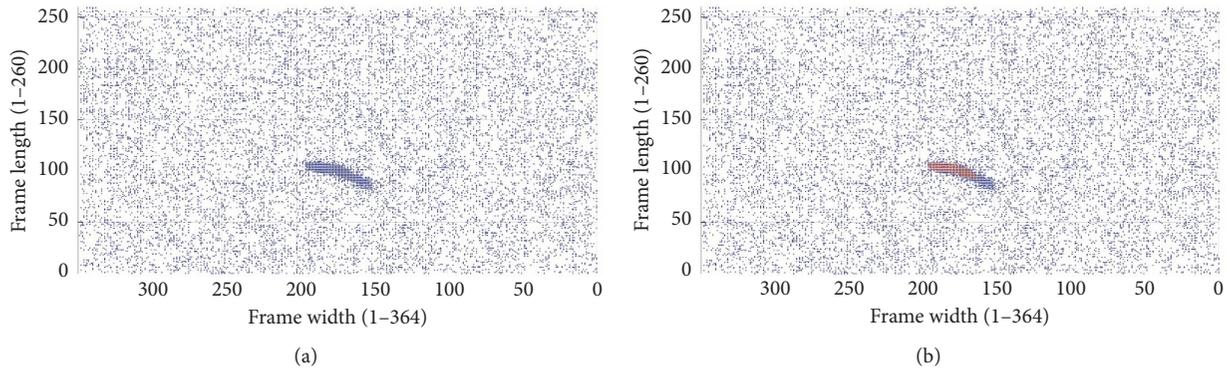


FIGURE 5: Event stream marked with object events compared with raw event stream. The event stream marked with the target event basically covers the location of the target and the shape of the target. (a) Raw event stream. (b) Event stream marked with object events.

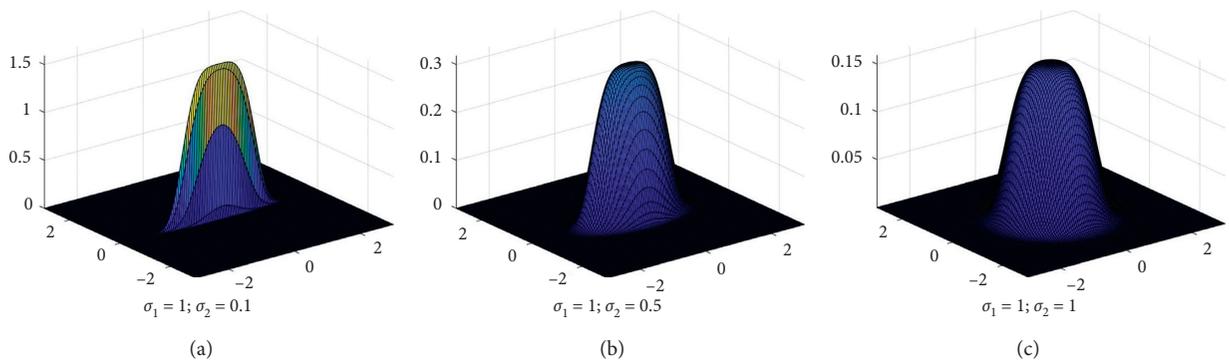


FIGURE 6: Three-dimensional images of Gaussian kernels with different σ_2 . (a) $\sigma_1 = 1; \sigma_2 = 0.1$. (b) $\sigma_1 = 1; \sigma_2 = 0.5$. (c) $\sigma_1 = 1; \sigma_2 = 1$.

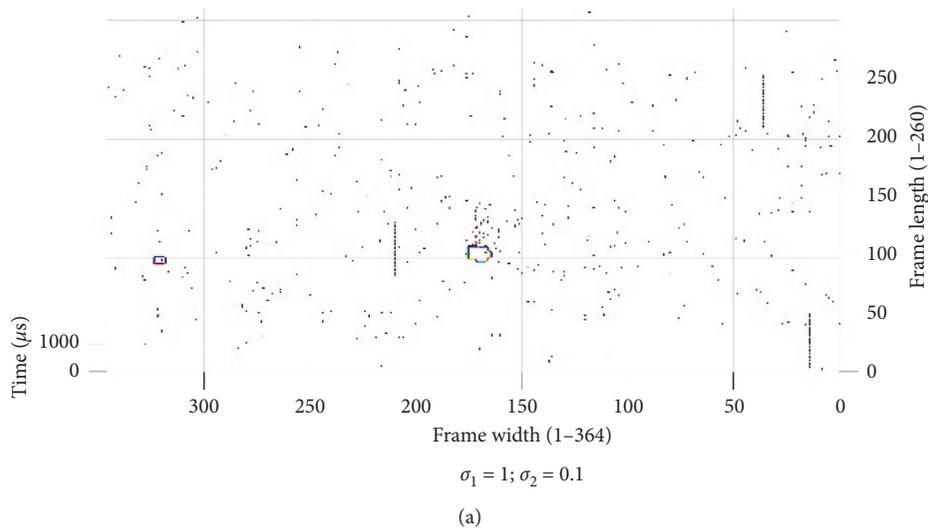


FIGURE 7: Continued.

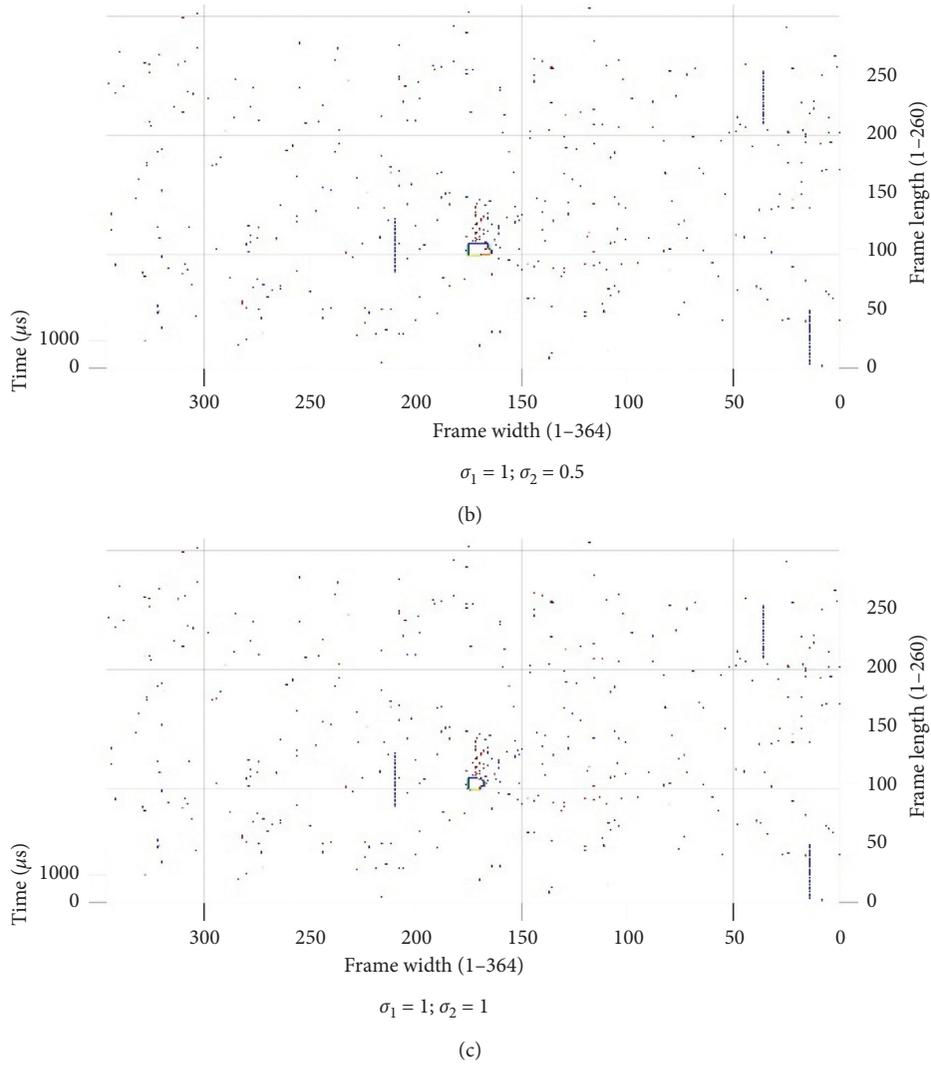


FIGURE 7: Object events obtained with different σ_2 . (a) $\sigma_1 = 1; \sigma_2 = 0.1$. (b) $\sigma_1 = 1; \sigma_2 = 0.5$. (c) $\sigma_1 = 1; \sigma_2 = 1$.

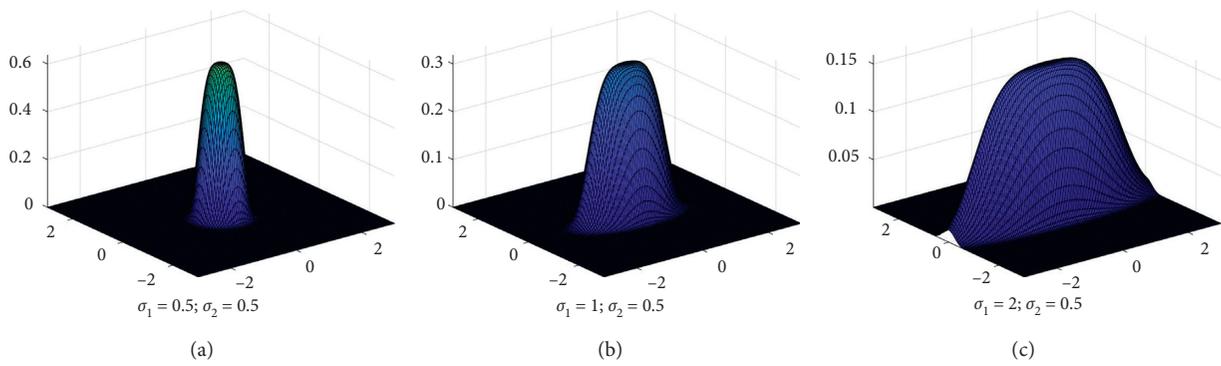


FIGURE 8: Three-dimensional images of Gaussian kernels with different σ_1 . (a) $\sigma_1 = 0.5; \sigma_2 = 0.5$ (b) $\sigma_1 = 1; \sigma_2 = 0.5$ (c) $\sigma_1 = 2; \sigma_2 = 0.5$.

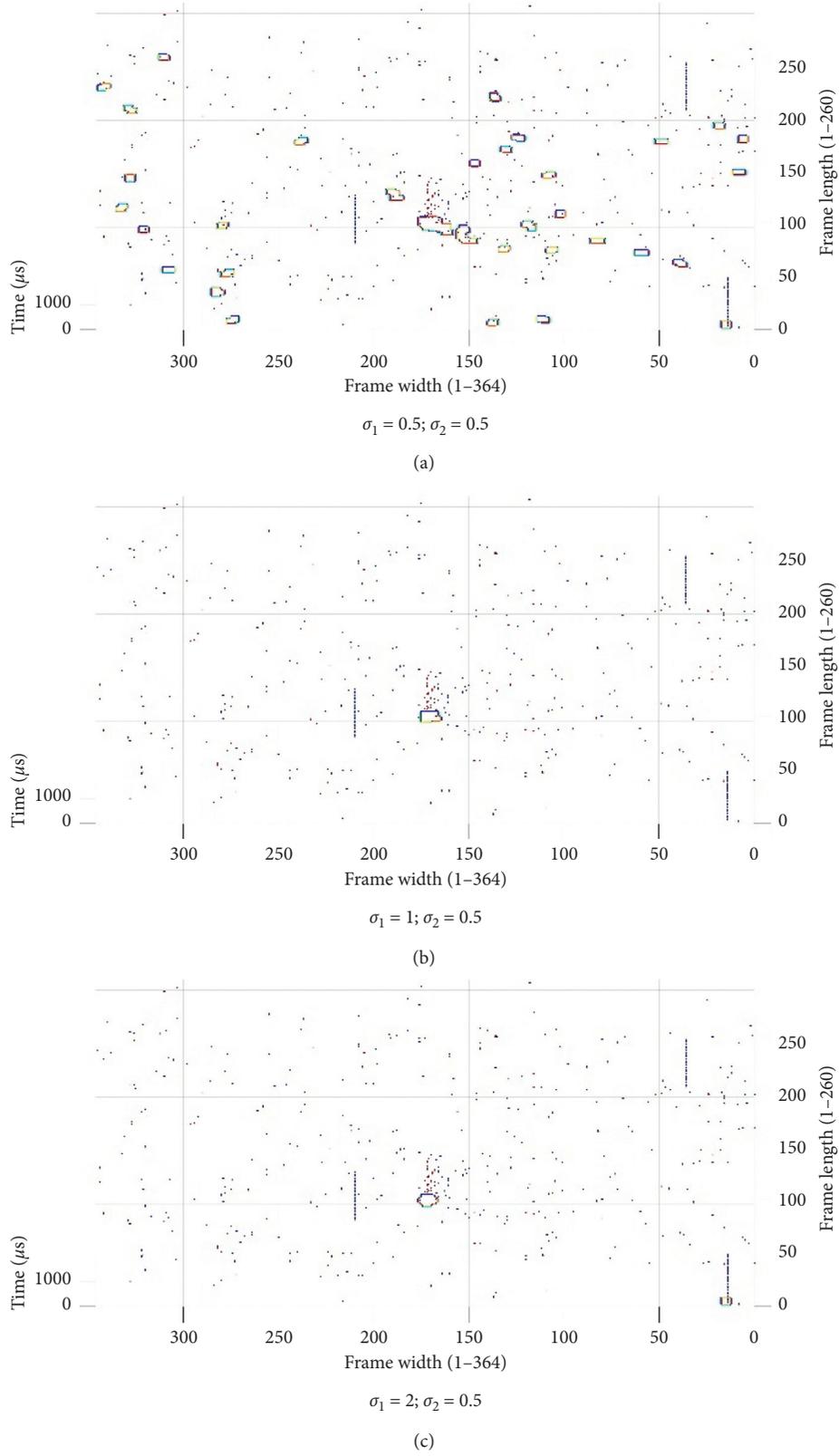


FIGURE 9: Object events obtained with different σ_1 . (a) $\sigma_1 = 0.5; \sigma_2 = 0.5$ (b) $\sigma_1 = 1; \sigma_2 = 0.5$ (c) $\sigma_1 = 2; \sigma_2 = 0.5$.

6. Conclusions

In this work, we propose a new method to obtain the object position in the DVS event stream and track the object and analyze the influence of the parameters used on this method. A single DVS event cannot contain enough object information. Events with high correlation are required to form an event cluster to reflect the object's position and movement status. This method uses the event correlation index to obtain the event cluster of the object, thereby determining the location and shape of the object. This method can apply traditional image processing and object tracking methods to dynamic vision imaging systems, making them compatible with other existing systems. Experiments have proved that this method can get the position of the object and obtain the complete event stream of the object movement.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] M.-X. Jiang, X.-X. Luo, T. Hai, H.-Y. Wang, S. Yang, and A. N. Abdalla, "Visual object tracking in RGB-D data via genetic feature learning," *Complexity*, vol. 2019, Article ID 4539410, 8 pages, 2019.
- [2] M.-X. Jiang, C. Deng, Z.-G. Pan, L.-F. Wang, and X. Sun, "Multiobject tracking in videos based on LSTM and deep reinforcement learning," *Complexity*, vol. 2018, Article ID 4695890, 12 pages, 2018.
- [3] Y. Chen, W. Li, G. Xin, H. Yang, and T. Xia, "An improved strong tracking kalman filter algorithm for the initial alignment of the shearer," *Complexity*, vol. 2019, Article ID 3172501, 12 pages, 2019.
- [4] C. Brandli, R. Berner, M. Minhao Yang, S. Shih-Chii Liu, and T. Delbruck, "A 240×180 130 dB 3 μ s latency global shutter spatiotemporal vision sensor," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 10, pp. 2333–2341, 2014.
- [5] C. Posch, D. Matolin, and R. Wohlgenannt, "A QVGA 143dB dynamic range asynchronous address-event PWM dynamic image sensor with lossless pixel-level video compression," in *Proceedings of the Solid-state Circuits Conference Digest of Technical Papers*, 2010.
- [6] B. Son, Y. Suh, S. Kim et al., "4.1 A 640 \times 480 dynamic vision sensor with a 9 μ m pixel and 300Meps address-event representation," in *Proceedings of the 2017 IEEE International Solid-State Circuits Conference (ISSCC)*, pp. 66–67, San Francisco, CA, USA, February 2017.
- [7] H. Liu, "Design of an RGBW color VGA rolling and global shutter dynamic and active-pixel vision sensor," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*, Lisbon, Portugal, May 2015.
- [8] D. P. Moeys, F. Corradi, C. Li et al., "A sensitive dynamic and active pixel vision sensor for color or neural imaging applications," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 12, no. 1, pp. 123–136, 2018.
- [9] H. Jing, "Asynchronous high-speed feature extraction image sensor," Doctoral thesis, Nanyang Technological University, Singapore, 2018.
- [10] G. Gallego, T. Delbruck, G. M. Orchard et al., "Event-based vision: a survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, p. 1, November 2020.
- [11] M. Guo, R. Ding, and S. Chen, "Live demonstration: a dynamic vision sensor with direct logarithmic output and full-frame picture-on-demand," in *Proceedings of the 2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 456–456, Montreal, QC, Canada, May 2016.
- [12] C. Posch, D. Matolin, and R. Wohlgenannt, "A QVGA 143 dB dynamic range frame-free PWM image sensor with lossless pixel-level video compression and time-domain CDS," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 1, pp. 259–275, 2010.
- [13] T. Finatou, A. Niwa, D. Matolin et al., "5.10 A 1280 \times 720 back-illuminated stacked temporal contrast event-based vision sensor with 4.86 μ m pixels, 1.066GEPs readout, programmable event-rate controller and compressive data-formatting pipeline," in *Proceedings of the 2020 IEEE International Solid-State Circuits Conference - (ISSCC)*, pp. 112–114, San Francisco, CA, USA, February 2020.
- [14] S. Chen and M. Guo, "Live demonstration: CeleX-V: a 1M pixel multi-mode event-based sensor," in *Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1682–1683, Long Beach, CA, USA, June 2019.
- [15] M. Litzenberger, C. Posch, D. Bauer, A. N. Belbachir, and H. Garn, "Embedded vision system for real-time object tracking using an asynchronous transient vision sensor," in *Proceedings of the Digital Signal Processing Workshop-Signal Processing Education Workshop*, Teton National Park, WY, USA, September 2006.
- [16] S. Schraml and A. N. Belbachir, "A spatio-temporal clustering method using real-time motion analysis on event-based 3D vision," in *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, pp. 57–63, San Francisco, CA, USA, June 2010.
- [17] E. Piątkowska, A. N. Belbachir, S. Schraml, and M. Gelautz, "Spatiotemporal multiple persons tracking using Dynamic Vision Sensor," in *Proceedings of the 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 35–40, Providence, RI, USA, June 2012.
- [18] X. Lagorce, C. Meyer, S.-H. Ieng, D. Filliat, and R. Benosman, "Asynchronous event-based multikernel algorithm for high-speed visual features tracking," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 8, pp. 1710–1720, 2015.
- [19] J. Wu, K. Zhang, Y. Zhang, X. Xie, and G. Shi, "High-speed object tracking with dynamic vision sensor," in *Proceedings of the 5th China High Resolution Earth Observation Conference (CHREOC 2018)*, L. Wang, Y. Wu, and J. Gong, Eds., pp. 164–174, Springer Singapore, Singapore, Xi'an, China, March 2019.
- [20] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.