Trajectory reconstruction by means of an event-camera-based visual odometry method and machine learned features

S. Chiodini[1,a *], G. Trevisanuto[2,b], C. Bettanini[1,c], G. Colombatti[1,d] and M. Pertile[3,e]

[1]Department of Industrial Engineering, University of Padova, via Venezia 1, Padova (Italy)

[2]CISAS - Center for Studies and Activities for Space "Giuseppe Colombo", University of Padova, via Venezia 15, Padova (Italy)

[3]School of Engineering, University of Padova, Via Gradenigo 6/a, Padova, Italy

[a]sebastiano.chiodini@unipd.it, [b]giovanni.trevisanuto@studenti.unipd.it, [c]carlo.bettanini@unipd.it, [d]giacomo.colombatti@unipd.it, [e]marco.pertile@unipd.it

Keywords: Visual Odometry, Computer Vision, Machine Learning

Abstract. This paper presents a machine learned feature detector targeted to event-camera based visual odometry methods for unmanned aerial vehicles trajectory reconstruction. The proposed method uses machine-learned features to enhance the accuracy of the trajectory reconstruction. Traditional visual odometry methods suffer from poor performance in low light conditions and high-speed motion. The event-camera-based approach overcomes these limitations by detecting and processing only the changes in the visual scene. The machine-learned features are crafted to capture the unique characteristics of the event-camera data, enhancing the accuracy of the trajectory reconstruction. The inference pipeline is composed of a module repeated twice in sequence, formed by a Squeeze-and-Excite block and a ConvLSTM block with residual connection; it is followed by a final convolutional layer that provides the trajectories of the corners as a sequence of heatmaps. In the experimental part, a sequence of images was collected using an event-camera in outdoor environments for training and test.

Introduction

Bio-inspired systems are becoming increasingly widespread in the field of robotics. The advantages are related to the reduced use of resources, both in terms of power consumption and computational load. In terms of perception, Event-based vision sensors, such as Dynamic Vision Sensor (DVS) devices represent one of the intriguing advancements in image sensor technology. These devices incorporate in-pixel circuitry that can detect temporal changes in intensity and communicate these changes as binary "events" to the external world. Essentially, only the pixels that detect changes in light intensity transmit data, enabling data compression at the sensor level and facilitating low-latency operations. This is made possible because individual pixel changes can be transmitted without the need to read out full frame image frames [1]. Event-based cameras offer significant advantages over traditional cameras. Latency, which is the time delay in processing sensor data, is a critical factor, event-based cameras drastically reduce latency by transmitting data through events, which have microsecond-level latencies. Furthermore, event-based cameras possess a remarkably high dynamic range of 130 dB compared to the 60 dB range of standard cameras. This makes them well-suited for scenes with substantial illumination changes.

Certainly, one of the most promising applications for this type of camera is in the navigation of highly agile robots such as drones [2], as well as for the aspects of entry, descent, and landing of planetary probes [3]. To utilize these sensors for such purposes, it is necessary to adapt or invent new algorithms for Visual Odometry (VO). This ensures that the cameras can effectively support the navigation and mapping tasks required in these dynamic scenarios.

For VO, it is crucial to have keypoints that are repeatable and accurate across consecutive frames. Currently, there are handcrafted methods inspired by classical computer vision theory that allow the extraction of a series of features, such as the eHarris-based approach [3]. Inspired by the work of [4], we have chosen to utilize machine-learned features that exhibit a certain level of temporal stability. In this work, we present the method for training these machine-learned features, demonstrate how to integrate them into a visual odometry system, and showcase some preliminary results.

Method

The adopted event keypoint detection method is adapted from work of [4] and is based on receiving an event tensor (also called event cube) $E(x, y, t)$ of dimension $H \times W \times B$ as input and predicts a set of heatmaps as keypoint location. Regarding the event tensor input, H and W represent the height and width of the image sensor, respectively, and B indicates the number of temporal bins, which is 12 in our case. Generation of the event tensor involves several steps: as first the change in light $L_{xy,i}$ at pixel (x_i, y_i) crosses a threshold, a spike is generated; then the event camera outputs a spike stream with coordinates (x_i, y_i) and timestamp t_i, finally event stream is converted into an event tensor by considering an integration period Δt.

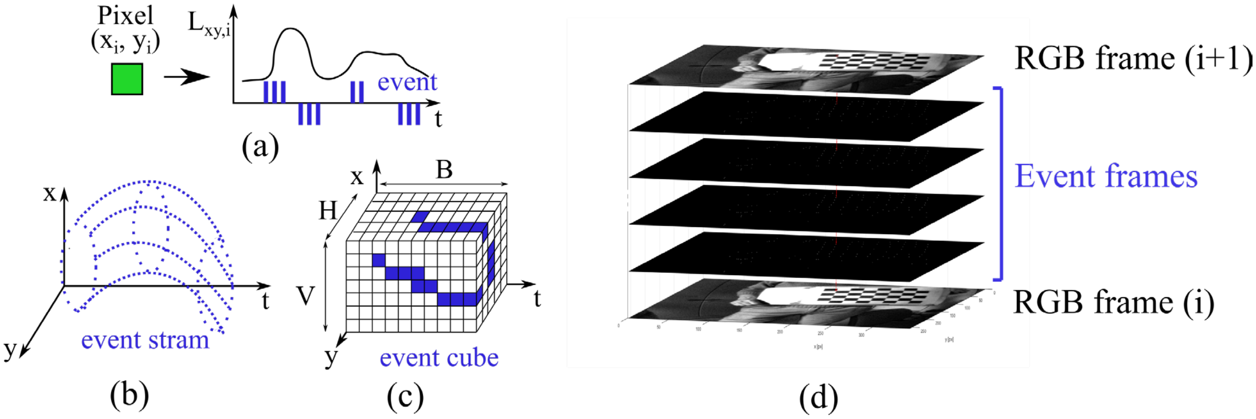

Figure 1 Event tensor generation: (a) when the change in light $\boldsymbol{L_{xy,i}}$ *pixel* $(\boldsymbol{x_i}, \boldsymbol{y_i})$ *crosses a threshold a spike is generated, (b) the event camera outputs a spike stream in time* $\boldsymbol{t_i}$ *and space* $(\boldsymbol{x_i}, \boldsymbol{y_i})$*, (c) the event stream is converted in an event tensor considering and integration period* $\boldsymbol{\Delta t}$*. (d) Event tensor* $\boldsymbol{E(x, y, t)}$ *used for detector training, the training points are detected using Harris on grayscale frame and interpolating their position on the event frames.*

The event tensor E(x, y, t) is utilized for detector training, where training points are detected using Harris on grayscale frames and then their positions are interpolated between two consecutive event frames and filtered based on epipolar constraint. The whole event tensor generation and training point selection is depicted in Figure 1.

The loss function is based on the Binary Cross Entropy (BCE) between the predicted heatmaps $\mathcal{H}_h(x, y)$ and the interpolated keypoints positions $\widehat{\mathcal{H}}_h(x, y)$:

$$\mathcal{L} = \sum_{h\in[1,n_h]} \sum_{(x,y)} \mathrm{BCE}(\mathcal{H}_h(x, y), \widehat{\mathcal{H}}_h(x, y)) \tag{1}$$

The first sum is over the n_h predicted heatmaps. The second sum is over the image locations (x, y). The neural network architecture used in this work is based on [5] and consists of a fully convolutional network with five layers, each utilizing 3x3 kernels. Each layer has 12 channels and incorporates residual connections. ConvLSTMs are employed in the second and fourth layers. The final layer, responsible for heatmap prediction, is a conventional convolutional layer. The remaining feed-forward layers utilize Squeeze-Excite (SE) connections. The training parameters are given in Table 1.

Table 1 Training parameters.

Num epochs	**Learning rate**	**Num ev cubes**	**Num ev**	**Num keypoints**	**Num keypoint/ev frame (avg)**
40	0.0001	6750	13.5*10^6	5442164	67.187

Results

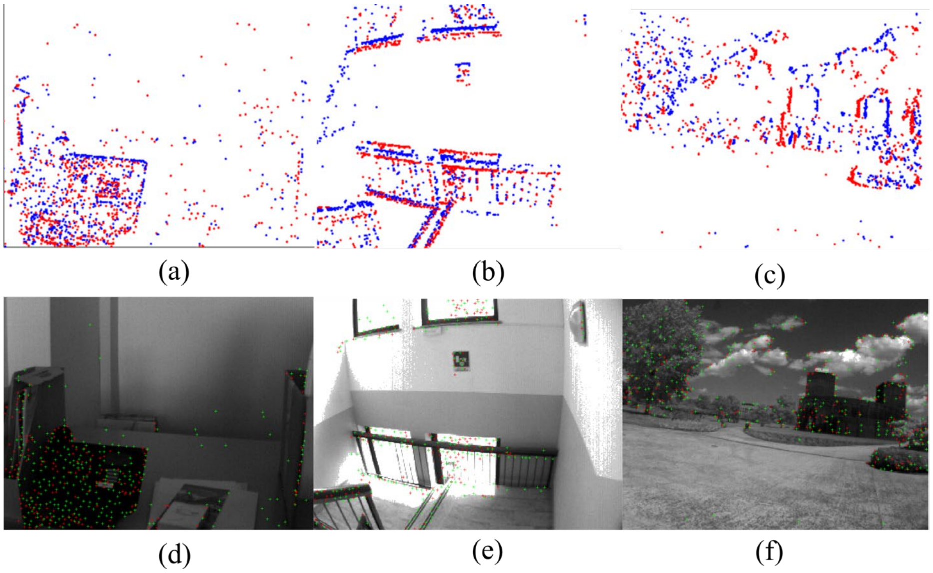
(a) (b) (c)
(d) (e) (f)

Figure 2 (a-c) Events frames obtained from the integration of 2000 events. (d.e) Corresponding RGB images of keypoints extracted from the respective event frames using the machine-learned detector (red) and a handcrafted feature detector such as eHarris (green).

To gather the frames and events required for training, sequences of images were collected in both outdoor and indoor environments. The DAVIS 346 camera from Inivation was employed for the acquisitions. The DAVIS 346 camera is a DVS event camera with a resolution of 346 x 260 pixels and includes an active pixel frame sensor. Figure 2 shows the event frames obtained from the integration of 2000 events and the corresponding RGB images with the detected keypoints. During the testing phase, the event cubes were provided as input to the machine-learned detector to extract the corresponding peak heatmaps. To verify the stability of the keypoints, a Nearest Neighbor (NN) algorithm was employed to track the keypoints in subsequent event frames. Figure 3 shows the graphs depicting the number of keypoints extracted, matched (between two consecutive frames using NN and filtered with RANSAC), and tracked (i.e., keypoints that, after being merged into tracks, belong to a track spanning at least 20 event frames).

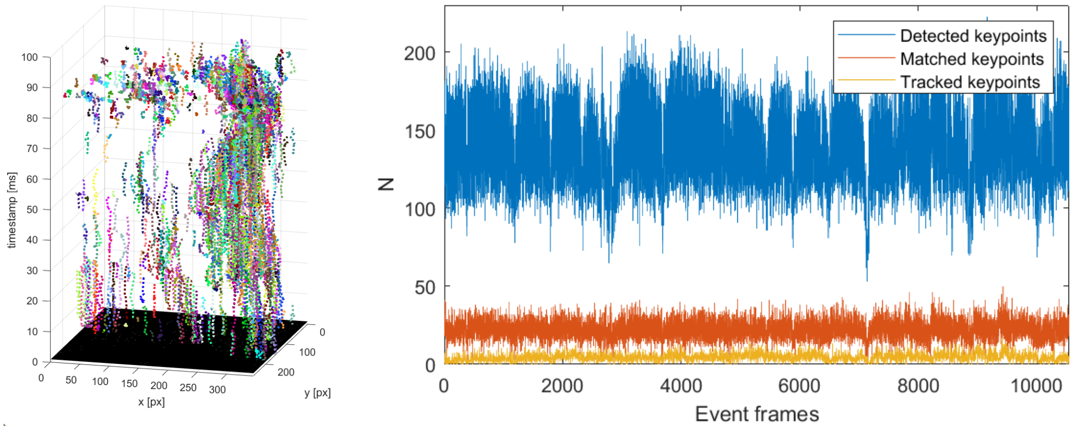

Figure 3 Detected, matched and tracked keypoints for the event stream collected outdoor.

Conclusions

In this work, the initial steps have been taken towards utilizing an event camera for the autonomous navigation of highly agile robots such as drones and entry descent and landing probes. The training of a stable keypoints detector across consecutive frames has been conducted. In future work, we will integrate this keypoint detector into a Visual Odometry pipeline and test the system on a tethered balloon.

Acknowledgements

The DVS/DAVIS technology was developed by the Sensors group of the Institute of Neuroinformatics (University of Zurich and ETH Zurich), which was funded by the EU FP7 SeeBetter project (grant 270324).

References

[1] Lichtsteiner, P., Posch, C., & Delbruck, T. (2008). A 128$\times $128 120 dB 15$\mu $ s latency asynchronous temporal contrast vision sensor. IEEE journal of solid-state circuits, 43(2), 566-576. https://doi.org/10.1109/JSSC.2007.914337

[2] Mueggler, E., Rebecq, H., Gallego, G., Delbruck, T., & Scaramuzza, D. (2017). The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and SLAM. The International Journal of Robotics Research, 36(2), 142-149. https://doi.org/10.1177/0278364917691115

[3] Sikorski, O., Izzo, D., & Meoni, G. (2021). Event-based spacecraft landing using time-to-contact. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 1941-1950). https://doi.org/10.1109/CVPRW53098.2021.00222

[4] Vasco, V., Glover, A., Bartolozzi, C.: Fast Event-Based Harris Corner Detection Exploiting the Advantages of Event-Driven Cameras. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 4144–4149 (2016). https://doi.org/10.1109/IROS.2016.7759610

[5] Chiberre, P., Perot, E., Sironi, A., & Lepetit, V. (2022). Long-Lived Accurate Keypoint in Event Streams. arXiv preprint arXiv:2209.10385.