Abstract

Event cameras have arisen as an alternative solution for solving pose estimation problems due to the many advantageous benefits of the technology. In this paper, we present a method of tracking the pose of an event camera through continuous tracking. Although our algorithm is feature-based, it alters the procedures of extraction and matching by performing continuous tracking in the spatiotemporal domain. We suggest that pseudo-images be maintained, which can be generated for every event timestamp. This representation enables the use of image-based features and tracking upon the arrival of an arbitrary event. We show that the sequential procedures of feature extraction and matching are unnecessary for event-based vision. We verified our algorithm using a public dataset in addition to data we acquired under poor illumination conditions to demonstrate the effective performance of our method under various circumstances for continuous-time trajectory estimation.

Introduction

Neuromorphic vision systems, also known as Dynamic Vision Sensors (DVS) [1], are a promising alternative to conventional imaging sensors with its advantages. Frame-based cameras produce an image, which is time-synchronized scene data on a speciﬁc interval. Such sequential data representations are intuitive and effective for visual odometry (VO) [2] and simultaneous localization and mapping (SLAM) [3] even with a single camera. However, as the sensor integrates the number of photons received in a fixed interval and summarizes the continuous photon flux into discrete images, changes in exposure time are not properly demonstrated and some information is even ignored during the blind period.

In order to overcome these problems, bio-inspired neuromorphic sensors and event cameras were introduced. Event-based vision systems are based on the rate at which photons enter each pixel, avoiding integration and asynchronously responding to changes in luminance in log scale. Therefore, event cameras retain a higher dynamic range (up to 130dB, which is significantly higher than the 60dB of conventional cameras) and lower latency (only several µs).

However, as event cameras follow Address-Event-Representation (AER) to demonstrate every luminance change in individual pixels, which are produced asynchronously in each pixel, not all algorithms based on image-based techniques are applicable for these type of sensors. Frame-based cameras produce discrete and sequential images that are mostly independent of motion. Therefore, to handle VO, data correspondence should be established for sequential images and poses should be estimated for each frame. Conversely, event cameras sense luminance changes in the AER format, which is comparatively continuous and non-sequential.

Therefore, sequential procedures of classical image processing such as feature extraction, descriptor matching, and optimization methods with reprojection error are not necessary for event-based algorithms.

Recently, methods of discovering the temporal correlation between event signals have been introduced. As it is easier for VO to build event-based frame images and be applicable for frame-based algorithms, model-based methods [4], [5] were introduced in the early stages. In addition, further probabilistic approaches [6], [7] and the method of utilizing temporal resolution by assigning weights to each event [8] were introduced. However, most of the aforementioned methods were based on the estimation of camera trajectory in sequences with fixed timing, which is no longer required for VO in the AER format. Therefore, an algorithm that could estimate camera trajectory without the limitations of fixed-timing was required. In this paper, we propose a method of estimating camera pose with a sequential format without a fixed time interval. We aim to fully utilize the temporal resolution and dynamic range of event cameras by:

• Defining event-based features and an efficient tracking method based on event potential, which is a pseudo-image generated from the event stream at an arbitrary time.

• Proposing an event-driven pose estimation framework that is driven by the number of events instead of time, guaranteeing a baseline of sufficient size and preserving the asynchronous nature of event-based features.

Related Works

The unique advantages of AER and event cameras have encouraged researchers to develop event-based motion estimation. As described in the previous section, event cameras have advantages and challenges for utilization. Therefore, researchers have developed new ways of finding relationships between events that are continuous yet asynchronous.

Early research developed approaches based on probabilistic aspects for event-based VO. An early method that saw success involved calculating posterior probability with temporal thresholding. [9] implemented Expectation Maximization (EM) on events for rotational motion estimation. This study was expanded to involve the optimization of the 6-degrees of freedom (DOF) trajectory with 3D reconstructions [7]. The authors of [6] implemented a particle filter to minimize ray distance through all features in 2D. This study was broadened to include 3D SLAM in [10], but this was only effective in planar scenes and trajectory.

Temporal thresholding, which sets effective events in a sequence with a fixed time interval, generates motion-related frame images and appeared to be a fair option for trajectory estimation. However, the reintegration of discrete signal into sequences with a fixed temporal window resulted in the loss of the asynchronous aspects of each event. Therefore, in order to achieve higher levels of accuracy and fully utilize the temporal resolution of event cameras, new methods of binding events were required.

In response to such demands, several studies attempted to utilize the high temporal resolution of event cameras by assigning weights to each event. [8] introduced a direct method of accurately estimating angular velocity and compensating events from motion. This enabled the generation of virtual event-based frames that conserve asynchronous aspects as the frames could be produced at any desired time. In combination with nonlinear optimization, [11] succeeded in integrating inertial measurement units (IMU) for 6-DOF-event VO. This study demonstrated a standard pipeline for event-based visual inertial navigation systems (VINS). However, the algorithm proposed in this study requires the generation of each motion-compensated event frame and feature. Although this approach was robust and effective, as in the case of feature-based VINS, approaches that were more straightforward than transforming AER into classical feature-based were available. [12] proposed a method of optimizing the trajectory of events and IMU by introducing cubic spline interpolation into [13]. However, Ultimate SLAM [14] was nonetheless the state-of-the-art option for integrating sensor measurements with motion-compensated event frames. However, the proposed algorithm should be altered to zero velocity in advance if there are few motions. Moreover, the proposed algorithm required updates to the spline parameter upon receiving each event, which made it too expensive in terms of computation.

On the other hand, another approach that involved the reminding of surface of active events (SAE) was introduced in [15]. [16] fitted a surface in an XYT space with SAE to estimate the effective temporal length for each event. These studies suggested finding the effective size of the temporal window and to only ﬁlter relevant events with adaptive temporal window sizes. The proposed algorithm provides a hint for utilizing the high temporal resolution characteristics of events by processing with individual timings.

In [17], a modified version of SAE with exponentially decaying kernels was introduced for VO. In this study, events are accumulated into exponential time-surface and calculated for an optimal camera pose by optimizing re-projection error with other measurement models in stereovision. The study focused on directly estimating the camera pose with exponentially decaying kernels, which provided a smoothed representation of SAE. The authors introduced a robust descriptor of SAE in addition to refinement techniques with tree assignment methods. As a result, the authors were able to enhance the mean tracking frequency on the scale. Although their work is based on the complex model and the user-defined descriptor, it shows that exponentially decaying kernels are suitable for event-based asynchronous tracking.

In our research, we introduce a method of continuously tracking features in the generated frames to alter the VO framework of stitching events into motion-compensated frames.

Method

In the proposed method, the framework of temporally sequential feature extraction and matching is modified by introducing exponentially decaying event-based potential (Ve). In contrast to motion-compensated frames, the proposed event potential representation encodes motion in itself. Our method alters the building of sequential event-based images by maintaining pseudo-image states stimulated by the occurrence of positive and negative events. As events are generated upon changes in luminance, assuming a static environment, the rate of events purely depends on the transmutation of the camera state. Therefore, it is more efficient for VO to track according to the number of incoming events than fixed time intervals, as a sufficient number of event signals guarantees an appreciable size of baseline.

By estimating the magnitude of optical flow in each feature, we can determine the optimal update rate for each point and track features asynchronously according to its individual timing. With the estimated translation rate of features, the search area can be narrowed down to very small neighboring pixels (size of five), reducing computation and increasing the robustness of feature tracking.

A. Event Potential (Ve) and Event-based Features

An event potential is an artificial pseudo-image that is generated from an event stream; this image-like format does not require a specific temporal window, but rather preserves exponentially decaying scalar values for every event. As biological neurons accumulate discrete spikes into potential with temporal decay, this artificial image is defined as an event potential. This pseudo-image is updated upon the arrival of each individual positive and negative event, increasing or decreasing the value contained in each pixel by a specific amount. Therefore, pixels with more recent events possess larger values and older pixels contain smaller values. An exponential decay function is selected for more efficient computations.

Given the event potential, feature tracking is the procedure of identifying a spatiotemporal relationship between event streams around specific points. As an event potential encodes camera motion in itself to illustrate the temporal arrangement of events according to the value, the intensity of the generated pseudo-image is directly related to the time between the last event occurrence in each pixel. Therefore, defining a feature descriptor on the image domain of an event potential should encode information of the temporal patterns of event arrival around the local window. We chose to compare random N points around a feature, which is identical to BRIEF descriptors [18]. Defining a binary feature descriptor built upon a small window enables the tracking of event-based features by comparing approximated temporal ordering between spatiotemporally close events.

B. Variable Timing Tracking and Motion Estimation

Due to the characteristics of exponential decay, the ratio of values of two distinct pixels are preserved without the arrival of further events. This allows us to inversely estimate the gradient of SAE for each point, given the time elapsed after the arrival of events. The magnitude of a gradient in SAE is identical to the optical flow strength, and the expected time period of a feature to translate a single pixel in an image is inversely proportional to optical flow magnitude. Therefore, we can predict the transition timing of each feature by calculating the gradient of SAE, which is approximated from the event potential.

Utilizing this characteristic, each feature can be tracked asynchronously. However, in most controlled environments, setting different temporal intervals did not yield significant effects. Therefore, we performed tracking with fixed interval numbers for events and conducted feature-based VO for every n-event arrival.

Fig. 2. The next state of an event potential is easily computed by summing the decayed Ve(t) and the new events. As we defined decay as an exponential function, it is possible to obtain the event potential at a specific time by simply multiplying a constant value e−τt and adding newly arrived events. In our setting, we used τ = 30.

Fig. 3. Event potential (Ve) can be obtained for every event timestamp or after a specific number of events. In order to limit the search area for feature tracking, n = 6000 was used in our case.

Algorithm 1: Event-driven tracking and pose estimation