Event-based Visual Inertial Odometry

Alex Zhu, Nikolay Atanasov, Kostas Daniilidis
University of Pennsylvania

Event-based Cameras
Event-based cameras asynchronously capture changes in log light intensity. Whenever the log light intensity over any pixel changes over a set threshold, the camera immediately returns the pixel location of the change, a timestamp with microsecond accuracy, and the direction of the change (+ or -).

\[ \{(x_i, t_i) : \log(|(x_i, t_i) - \log(|x_i, t_i - t_i)|) \geq \theta\} \]

The camera exhibits extremely low latency and high dynamic range.

Motivation
Given a set of event and inertial measurements, estimate the sensor state \( x(t) = [v_0, v_1, v_2, v_3] \) over time.

Challenges
Unknown data association between events over time, no intensity information.

Prior work
Event-based visual odometry given a prior map [1], combined with grayscale images [2], event-based SLAM methods [3], [4].

Contributions
A novel event association scheme resulting in robust feature tracks by employing two EM-steps and variable temporal frames depending on flow and rotation estimates obtained from the odometry filter. The first visual odometry system for event-based cameras that makes use of inertial information.

Objective
Given a set of event and inertial measurements, estimate the sensor state \( x(t) = [v_0, v_1, v_2, v_3] \) over time.

Feature Tracking
For each spatiotemporal window, propagate the events to a common time using the current state estimate of the flow, and then jointly estimate the soft data association, \( r_{ij} \), between the propagated events from the current and last windows, and the optical flow, \( \nu \). The optical flow is used to estimate the future position of the feature.

EM1: Flow Estimation
Given a set of propagated events, warp them according to the rotation between the current state rotation and the rotation of the first window, \( \omega_i \), and then jointly estimate the soft data association, \( r_{ij} \), scaling, \( s_i \), and translation, \( b_i \), between the current events and those from the first window, \( \omega_i \). This corrects for drift and detected tracks.

EM2: Template Alignment
For each spatiotemporal window, propagate the events to a common time using the current state rotation and the rotation of the first window, \( \omega_i \), and then jointly estimate the soft data association, \( r_{ij} \), scaling, \( s_i \), and translation, \( b_i \), between the current events and those from the first window, \( \omega_i \), \( \omega_i \).

Temporal Window Size: Lifetime Estimation
The size of the next temporal window, \( \tau \), is set to the time taken for a point to move 3 pixels. This constrains the amount of motion in the window to be small, to satisfy the constant optical flow assumption within each window.

References

Visual Inertial Odometry
Filter
The feature positions and IMU measurements are fused using the MSCKF [5] framework, which optimizes the sensor without optimizing over the 3D feature positions. This method provides a fast state update over a window of feature observations.

Outlier Rejection
After each feature tracking update, two-point RANSAC [6] is used to remove outliers and failed tracks. Given rotation from the IMU, the largest set of point pairs that correspond to the same translation between the current and last frame is kept as inliers. In addition, before each feature track is marginalized, the largest set of observations that project to the same 3D point is kept as inliers.