Real-Time Object Tracking and Trajectory Mapping Using Sparse Optical Flow

Muhammad Umair Ajmal, Zeeshan Haider, Muhammad Huzaifa
School of Electrical Engineering and Computer Science (SEECS)
National University of Sciences and Technology (NUST)
Islamabad, Pakistan
{majmal.bee22seecs, zhaider.bee22seecs, mhuzaifa.bee22seecs}@seecs.edu.pk

***Abstract*—Real-time object tracking is a fundamental problem in computer vision with applications in surveillance, robotics, intelligent transportation systems, and human–computer interaction. Accurate tracking in unconstrained environments is challenging due to noise, occlusion, abrupt motion, and illumination variations. This paper presents a robust classical computer vision framework for real-time object tracking and trajectory mapping using sparse optical flow. The proposed system employs Shi–Tomasi corner detection for feature initialization and the pyramidal Lucas–Kanade method for tracking salient feature points across video frames. To enhance robustness, RANSAC-based outlier rejection is applied to eliminate inconsistent motion vectors, while a Kalman Filter is integrated to model object motion and smooth noisy measurements. A manual region-of-interest initialization mechanism ensures precise target selection and minimizes background interference. Experimental evaluation on real-world video sequences demonstrates stable tracking performance and smooth trajectory estimation without reliance on computationally expensive learning-based models. The results confirm that classical computer vision techniques remain effective for real-time tracking under limited computational and data constraints.**

***Index Terms*—Object Tracking, Sparse Optical Flow, Lucas–Kanade Method, Kalman Filter, Trajectory Mapping, Classical Computer Vision**

I. Introduction

Object tracking aims to estimate the spatial position of a target across consecutive video frames while preserving temporal consistency. Unlike object detection, which operates independently on individual frames, tracking exploits motion continuity and appearance coherence over time. Real-time object tracking is a critical component in applications such as automated surveillance, autonomous navigation, video analytics, and augmented reality.

Despite extensive research, reliable tracking in real-world environments remains challenging due to factors such as motion blur, dynamic backgrounds, partial occlusions, and illumination changes. While modern deep learning-based trackers have achieved remarkable accuracy, their deployment is often limited by high computational cost, large training data requirements, and reduced interpretability.

Classical computer vision techniques offer an attractive alternative for real-time systems, particularly in resource-constrained environments. Feature-based tracking methods leverage salient visual points and geometric consistency to estimate motion efficiently. Sparse optical flow techniques, in particular, provide a favorable balance between computational efficiency and tracking accuracy.

Motivated by these considerations, this work proposes a robust real-time object tracking framework that integrates sparse optical flow with probabilistic motion modeling. By combining spatial feature tracking, geometric validation, and temporal filtering, the proposed approach achieves stable trajectory estimation while maintaining real-time performance without training overhead.

II. Related Work

Early object tracking approaches relied on frame differencing and template matching, which were sensitive to noise and illumination variations. Feature-based methods improved robustness by tracking distinctive points such as corners and edges across frames. Among these, the Lucas–Kanade optical flow algorithm remains one of the most widely adopted methods due to its efficiency and accuracy for small inter-frame motion.

Subsequent research introduced robust estimation techniques such as RANSAC to handle outliers caused by background motion and occlusions. Probabilistic filtering approaches, including Kalman and particle filters, further enhanced tracking stability by modeling object dynamics over time.

Recent deep learning-based trackers employ convolutional neural networks and transformer architectures to learn robust appearance models. However, these methods require extensive training data and powerful hardware, limiting their applicability in real-time and embedded scenarios. In contrast, classical approaches remain relevant due to their interpretability, efficiency, and ease of deployment.

III. System Overview

The proposed tracking system follows a multi-stage processing pipeline designed to achieve robust real-time performance. The pipeline consists of manual target initialization, feature point extraction, sparse optical flow estimation, outlier rejection, probabilistic motion filtering, and trajectory visualization.

Each stage addresses a specific challenge in object tracking. Feature-based motion estimation provides efficiency, RANSAC ensures spatial consistency, and Kalman filtering enforces temporal smoothness. This modular design allows individual components to be extended or replaced without affecting the overall architecture.

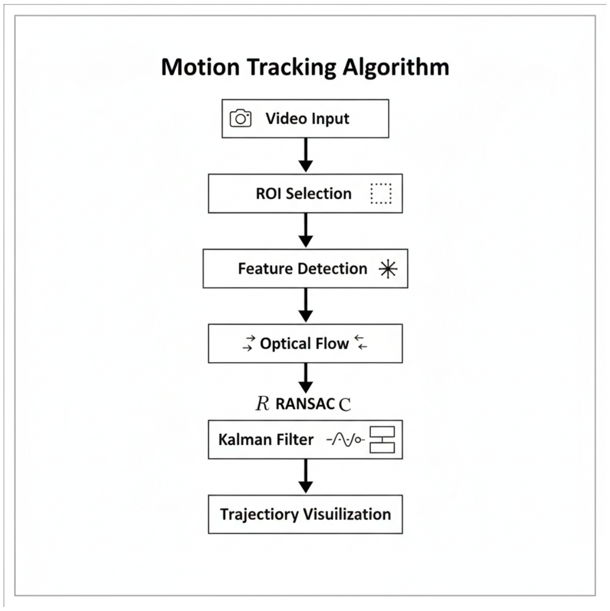

Fig. 1. Block diagram of the proposed real-time object tracking and trajectory mapping pipeline.

IV. Methodology

A. Region of Interest Initialization

To ensure accurate tracking initialization, the system allows the user to manually select a region of interest (ROI) around the target object. The video stream is paused, and a bounding box is drawn interactively. Restricting feature detection to the ROI minimizes background interference and improves tracking reliability.

B. Feature Point Detection

Salient feature points are extracted within the selected ROI using the Shi–Tomasi corner detector. This method identifies points with strong intensity variations, making them suitable for tracking across frames. Limiting detection to the ROI improves feature relevance and computational efficiency.

C. Sparse Optical Flow Estimation

The pyramidal Lucas–Kanade optical flow algorithm is used to estimate feature point motion between consecutive frames. The method assumes brightness constancy and small inter-frame motion, and its pyramidal implementation enables handling moderate object displacement while maintaining real-time performance.

D. Mathematical Formulation of Optical Flow

The Lucas–Kanade optical flow method is based on the brightness constancy assumption, which states that the intensity of a moving pixel remains constant between consecutive frames:

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t) \tag{1}$$

Fig. 2. Sparse feature points tracked using the Lucas–Kanade optical flow method.

By applying a first-order Taylor expansion and ignoring higher-order terms, the optical flow constraint equation is obtained:

$$I_x u + I_y v + I_t = 0 \tag{2}$$

where (u, v) represent the pixel displacement between frames and I_x, I_y, and I_t denote spatial and temporal image gradients.

The Lucas–Kanade method estimates motion by solving this equation over a local neighborhood using least squares optimization.

E. RANSAC-Based Outlier Rejection

Tracked feature correspondences may include erroneous matches due to noise or background motion. To eliminate such outliers, RANSAC-based homography estimation is applied. Only geometrically consistent feature points are retained for subsequent processing, significantly improving robustness.

F. Kalman Filter Motion Modeling

A Kalman Filter is employed to smooth object trajectories and suppress temporal noise. The object state is modeled using a constant velocity assumption:

$$\mathbf{x}_k = [x_k, y_k, v_{x_k}, v_{y_k}]^T$$

The filter predicts the object position and corrects it using centroid measurements derived from tracked feature points, ensuring stable and continuous trajectory estimation.

G. Kalman Filter State Estimation

The Kalman Filter models object motion using a linear dynamic system:

$$\mathbf{x}_k = A\mathbf{x}_{k-1} + \mathbf{w}_k \tag{3}$$

$$\mathbf{z}_k = H\mathbf{x}_k + \mathbf{v}_k \tag{4}$$

where $\mathbf{x}_k$ represents the system state, $\mathbf{z}_k$ is the measurement vector, and $\mathbf{w}_k$ and $\mathbf{v}_k$ denote process and measurement noise, respectively.

Fig. 3. Kalman filter–smoothed object trajectory visualization.

H. Trajectory Visualization

The filtered object positions are accumulated over time to generate a continuous trajectory. The system visualizes individual feature points, the estimated object center, and the complete motion path, providing intuitive interpretation of object dynamics.

V. Experimental Results and Discussion

The proposed tracking framework was evaluated on real-world video sequences containing moving objects under varying conditions. Experimental results demonstrate that the system successfully maintains object lock under moderate noise, partial occlusions, and illumination changes.

RANSAC-based validation improves spatial consistency, while Kalman filtering significantly reduces trajectory jitter compared to raw optical flow estimates. The system achieves smooth and accurate trajectory mapping while maintaining real-time performance on standard computing hardware.

VI. Limitations and Future Work

The system may experience performance degradation under prolonged occlusion or extremely fast motion that violates optical flow assumptions. Future work includes automatic target detection, multi-object tracking, and integration with deep learning-based detectors for hybrid tracking systems.

VII. Conclusion

This paper presented a robust classical computer vision framework for real-time object tracking and trajectory mapping using sparse optical flow. By integrating feature-based motion estimation, geometric validation, and probabilistic filtering, the proposed approach achieves stable and efficient tracking without reliance on training data. The results confirm the continued relevance of classical vision techniques for real-time applications.

References

[1] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," *Proc. IJCAI*, 1981.

[2] J. Shi and C. Tomasi, "Good features to track," *Proc. CVPR*, 1994.

[3] R. E. Kalman, "A new approach to linear filtering and prediction problems," *ASME Journal of Basic Engineering*, 1960.

[4] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal*, 2000.