LIFT-VINS

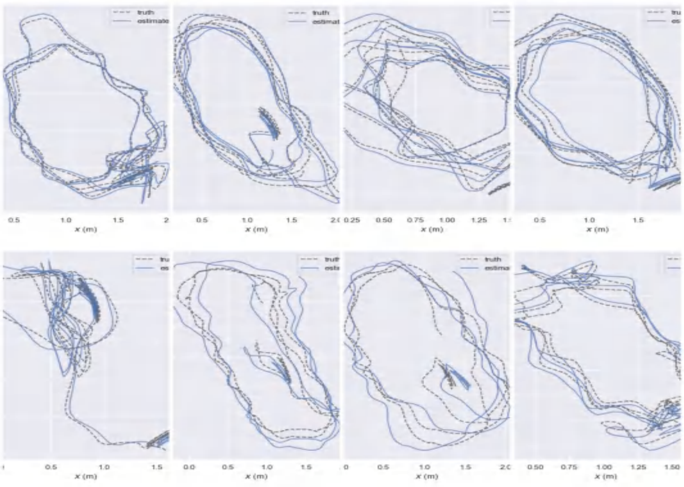
Feng Jielin, Dong Yuxiang, Zhao Liqun, Zou Zhenghao

Northwestern Polytechnical University

June 29, 2023

Abstract

LIFT-VINS is an improved VI-SLAM system based on the VINS-MONO open-source framework. Compared with the original framework, this paper constructs a fast corner extraction method in the front end that can overcome the light robustness and motion blur, and combines Optical flow method to match corners. The system in this article can achieve precise estimation in complex environments. In the case of loss of localization, rapid relocation can be achieved to ensure the robustness of localization.



Keypoint Detection

The method of feature extraction at the front end of our system is to refer to the idea of Unsuperpoint to obtain feature points that are robust and fast for lighting.

Unsuperpoint uses unsupervised learning to share a backbone network and three subtask modules (key extraction, description subcomputation, and score computation). The network structure is lightweight and can share weight information to improve the coupling degree between tasks.A schematic diagram of the backbone network structure is shown in Figure 1.

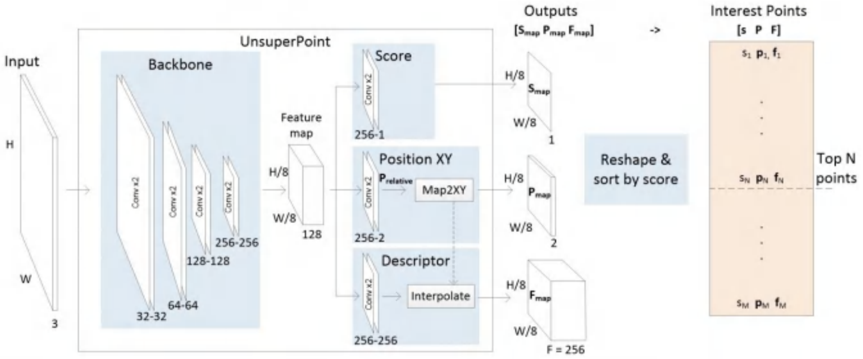


Figure 1 The network backbone of Unsuperpoint

Therefore, the total network loss function consists of four parts, and each has a weight coefficient added to it. The first loss term L USP is the unsupervised point (USP) loss of USP scores for learning locations and points of interest. The second loss term L uni_xy is a regularization term to encourage a uniform distribution of relative point positions. The last two loss terms, L desc and Lderr, only optimize the descriptor. Ldesc needs to learn descriptors, while Ldesstorr is just a regularization term that reduces overfitting by decorrelating descriptors.

$$L_{total} = a_{usp} L^{usp} + a_{uni_{x}y} L^{uni_x y} + a_{desc} L^{desc} + a_{decorr} L^{decorr}$$

The final experimental effect can prove that the key points extracted by Unsuperpoint have better repeatability and effectiveness, which makes the estimation effect more robust and accurate.

i. Visual Keyframe

Select the method for visual keyframing involves using the KLT sparse optical flow algorithm to track existing features and detect new corner features in each frame. Keyframes are selected based on two criteria: the mean parallax and tracking quality. The mean parallax criterion considers the average displacement of tracked features compared to the previous keyframe, while the tracking quality criterion ensures a minimum number of tracked features is maintained. This methodology helps maintain a consistent distribution of features and prevents feature loss. The utilization of rotation compensation using gyroscope measurements aids in accurate keyframe selection. Overall, these techniques contribute to robust feature tracking and reliable estimation of camera motion in visual-inertial navigation systems.

ii. IMU Pre-integration

IMU pre-integration uses nearly identical numerical results with On-manifold preintegration for real-time visual–inertial odometry and IMU preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation, But a different derivation process is used.

iii. System Inertial

For the initialization process, the overly loosely coupled method aligns the IMU pre-integration with the pure visual structure to obtain the necessary initial values. Based on the keyframes in the preset sliding window, check enough parallax to estimate a rough visual pose and map scale. The visual keyframes and IMU data are then synchronized to optimize gravity and IMU bias parameters. Finally, pass the estimate to the backend VIO to complete the initialization process.

iv. Back-end and loopback detection

The backend optimization typically involves performing nonlinear optimization, such as bundle adjustment, to jointly optimize the camera poses and landmark positions based on the observed feature tracks. This optimization step minimizes the reprojection error between the observed features and their corresponding projections in the estimated camera poses, thus refining the camera trajectory and landmark positions.

Loop closure detection is a crucial component in VINS-MONO that addresses the problem of accumulated drift over time. By identifying loop closures, the backend optimization can adjust the estimated trajectory to correct accumulated errors and improve the overall accuracy.

v. Performance

Relying on the algorithm we modified, the effect obtained on the dataset provided by the official competition is shown in the table I.

Table I : Test dataset targeting evaluation

Datas Sequence	APE/RPE(mm) average	ARE/RRE(deg) average	Completeness(%) average
C1-11_train	44.75/5.07	177.81/5.97425	92.70
D1-10_train	54.42/9.24	179.12/10.54	91.09
C1-11_test	120.00/8.028	175.89/6.00	60.69
D1-10_test	234.81/33.38	175.40/8.02	58.30

Our method can complete the positioning of most scenes, and the positioning accuracy is high. Although the performance is not good in some complex scenarios, the repositioning effect is very good, which proves that our SLAM system has excellent robustness.