**Path Estimation for multi-UAV with VINS-mono**

2021.05-07

Jeewon Kim, KAIST EE

**Introduction**

**1. VINS-mono Paper Review**

1.1 Description of VINS-mono [1]

VINS-mono is a technology that performs path estimation with one mono camera and one IMU. It is useful at disaster sites without GPS and it is also low-cost. The steps of VINS-mono are: keyframe selection – IMU Preintegration – body frame initialization – VIO(tightly coupled) – loop detection – relocalization(tightly coupled) – 4DOF pose graph optimization.

The advantages compared to other VINS are:

(1) using loosely align IMU pre-integration so that initialization and self-calibration can be done in the starting.

(2) marginalizing tightly coupled VIO process enables fast computation that is appropriate for low-latency pose estimation.

(3) tightly coupled cost function and sliding window makes it find more accurate and more numbers of loop.

(4) 4DOF enforces the global consistency.

(5) pose graph optimization and relocalization run asynchronously in two separate threads so relocalization thread can immediately use the most optimized pose graph.

(6) optimization automatically merges current map and previous-built map.

1.2. Terms in VINS-mono

- IMU Preintegration: Can estimate relative bias change between two frames with IMU measurements so that estimator takes less computation. It is loosely aligned value since you can calculate solely with IMU measurements.

- PnP(perspective-n-point): Calculate orientation and position of camera from the feature with n 3D points and 2D projection points to the image.

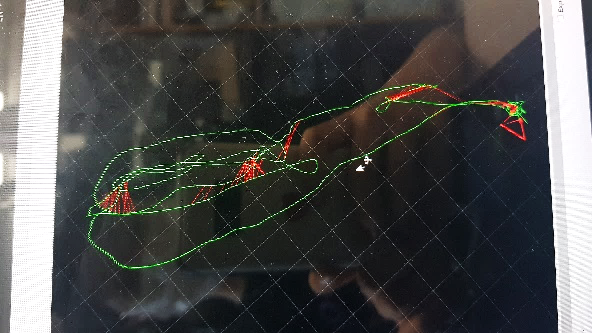
- Five-point algorithm: if n=5 in PnP algorithm

- Loop detection: Check if the robot has visited the same position. Can re-localize when the loop closure is detected and perform global pose graph optimization.

- Tightly coupled: minimize the sum of camera residual cost and IMU residual cost

**2. Multi-robot path estimation using VINS-mono**

Kwangyik Jung(@URL) developed VINS-mono that can estimate path by operating two sequence bag files of drones by adding keyframe subscriber from another robot so that currently running drone can detect more loops. The original code has only one subscriber of pose graph. However, the multi-robots system needs to subscribe information from several other robots. In this project, I will extend subscribers into four to perform path estimation of running drone with four other drones.

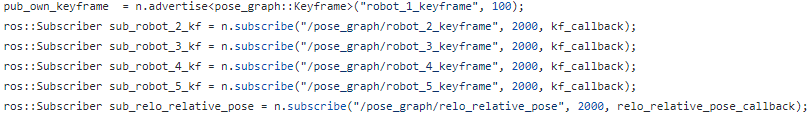


[Figure] Path estimation of MH\_01 with MH\_02(control group)

**Research**

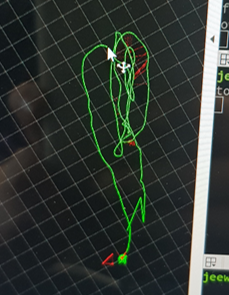
**1. Making more keyframe subscribers**

Currently running robot should subscribe all topics from other four robots, so I additionally built more keyframe subscriber.



**2. Keyframe-based Relocalization**

New challenging point was that drone has to come to the initial position, but it didn’t perform estimation in the last section normally.



[Figure] Path estimation after applying more subscribers

It is because MH sequences have different initial positions and facing directions but VINS-mono is using pose-based relocalization. Thus error in R/T(relative transformation?) will be occurred. So I modified the method of calculating R/T in relocalization callback function to keyframe-based. I created 2D point and 3D point of previous robot keyframe and applied PnP algorithm

**3. Multi-thread**

Robot came back to initial point, but still path estimation didn’t performed continuously in the end section.



[Figure] Path estimation after modifying relocalization callback

I judged that there was a topic loss or overlap problem, the previous one was ran in single-thread and subscribers use the same keyframe callback function but the topic is different. The topics of previous-ran robots couldn’t be subscribed if they came in simultaneously. So I applied multi-thread and lock function to prevent the overlap of topic subscription.

/main

 /keyframe\_callback



[Figure] Path estimation after applying ros multi-thread

**Conclusion**

I studied and parsed the code of VINS-mono. Then I modified the double-robot path estimation code derived from VINS-mono to enable path estimation for five robots. I extended the keyframe subscribers and there was a problem that estimator couldn’t estimate the path normally in the end of the sequence. The challenging problems were solved by applying vision-based relative transform calculation and multi-thread, and it succeeded to estimate path with the keyframe of four more previously-ran robots. Yet, I didn’t calculate the accuracy of the estimated path. In the further research, I should calculate the accuracy and compare with the accuracy of the result of original estimator, and improve the performance by making it detect more loops with other robots. Also I found out that there is research of collaborative localization for multi-MAV that performs optimization in centralized form(co-VINS) [2] so I would refer to that research.

**References**

[1] T. Qin, P. Li and S. Shen, "VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator," in IEEE Transactions on Robotics, vol. 34, no. 4, pp. 1004-1020, Aug. 2018, doi: 10.1109/TRO.2018.2853729. <https://github.com/HKUST-Aerial-Robotics/VINS-Mono>

[2] Vemprala, Sai, and Srikanth Saripalli. "Monocular vision based collaborative localization for micro aerial vehicle swarms." 2018 International Conference on Unmanned Aircraft Systems (ICUAS). IEEE, 2018.