Technical Report: Visual-Inertial SLAM for GPS-Denied Drone Navigation

Author: Kaveh SEDIGH
Date: December 21, 2025
Subject: Computer Vision Engineer Technical Assessment

1. Executive Summary

This report details the design, implementation, and evaluation of a state estimation pipeline for autonomous drones in GPS-denied environments. A **VINS-Mono** (Monocular Visual-Inertial System) pipeline was deployed using Docker and tuned for the EuRoC MAV dataset. The system achieved an Absolute Trajectory Error (ATE) RMSE of **0.144m** on the `MH_01_easy` sequence, demonstrating high robustness suitable for warehouse and tunnel inspection tasks.

2. Phase A: Literature Review & Method Selection

2.1. Comparative Analysis

Three state-of-the-art (SOTA) frameworks were evaluated for this application: **VINS-Mono**, **ORB-SLAM3**, and **OpenVINS**.

Feature	**VINS-Mono (Optimization)**	**ORB-SLAM3 (Feature-based)**	**OpenVINS** (Filter-based)
Computational Efficiency	**Moderate.** Sliding-window optimization keeps complexity bounded. Feasible for Jetson TX2/Xavier.	**Low.** Large map maintenance and bundle adjustment can be CPU intensive.	**High.** MSCKF filter is extremely lightweight and fast. Best for micro-drones.

Feature	**VINS-Mono** (Optimization)	**ORB-SLAM3** (Feature-based)	**OpenVINS** (Filter-based)
Robustness (Fast Motion)	**High.** Tightly-coupled IMU pre-integration handles rapid rotation and motion blur effectively.	**Medium.** Relies heavily on visual feature tracking. Prone to "Tracking Lost" during aggressive maneuvers.	**High.** Filter-based approach is naturally robust to visual outages.
Loop Closure	**Integrated.** Uses DBoW2 for robust place recognition and 4-DOF pose graph optimization.	**Excellent.** Best-in-class loop closing and map merging capabilities.	**Limited.** primarily an odometry framework; loop closure is often an add-on or less mature.

2.2. Selection Justification

Selected Framework: VINS-Mono

Reasoning:

1. **Robustness is Priority:** For a drone operating in "warehouses and tunnels" (as per the scenario), lighting changes and rapid turns are common. VINS-Mono's tightly-coupled optimization offers better resilience than ORB-SLAM3 in these texture-poor or blurry conditions.
2. **Loop Closure Necessity:** Unlike OpenVINS (which is primarily VIO), VINS-Mono includes a full SLAM backend (Loop Closure). This is critical for bounded drift during long-duration warehouse patrols.
3. **Engineering Maturity:** The codebase is stable, widely used in the robotics community, and offers excellent visualization tools (Rviz integration), making it ideal for a 1-week implementation.

3. Phase B: Implementation Details

3.1. Environment & Deployment

- **Containerization:** The entire pipeline is Dockerized (`osrf/ros:noetic-desktop-full`) to ensure reproducibility.

- **Compatibility:** Source code was patched to support **OpenCV 4**, resolving legacy dependency issues common in modern environments.

3.2. Sensor Configuration & Tuning

The system was tuned for the **EuRoC MAV Dataset (MH_01_easy)**.

- **IMU Noise:** Accelerometer and Gyroscope noise densities were adjusted to match the **ADIS16448** sensor characteristics, with slight inflation to account for airframe vibration.
- **Extrinsics:** The camera-IMU transform (T_{IC}) was fixed to the known calibration values to prevent initial estimation drift.
- **Time Synchronization:** Hardware triggering was assumed (offset = 0), as per the dataset specifications.

4. Phase C: Quantitative Analysis & Evaluation

4.1. Trajectory Evaluation

The estimated trajectory was compared against the OptiTrack Ground Truth using the `evo` evaluation package.

Metric: Absolute Trajectory Error (ATE) - Translation Part

- **RMSE: 0.144 m**
- **Mean Error:** 0.126 m
- **Max Error:** 0.391 m
- **Median Error:** 0.113 m

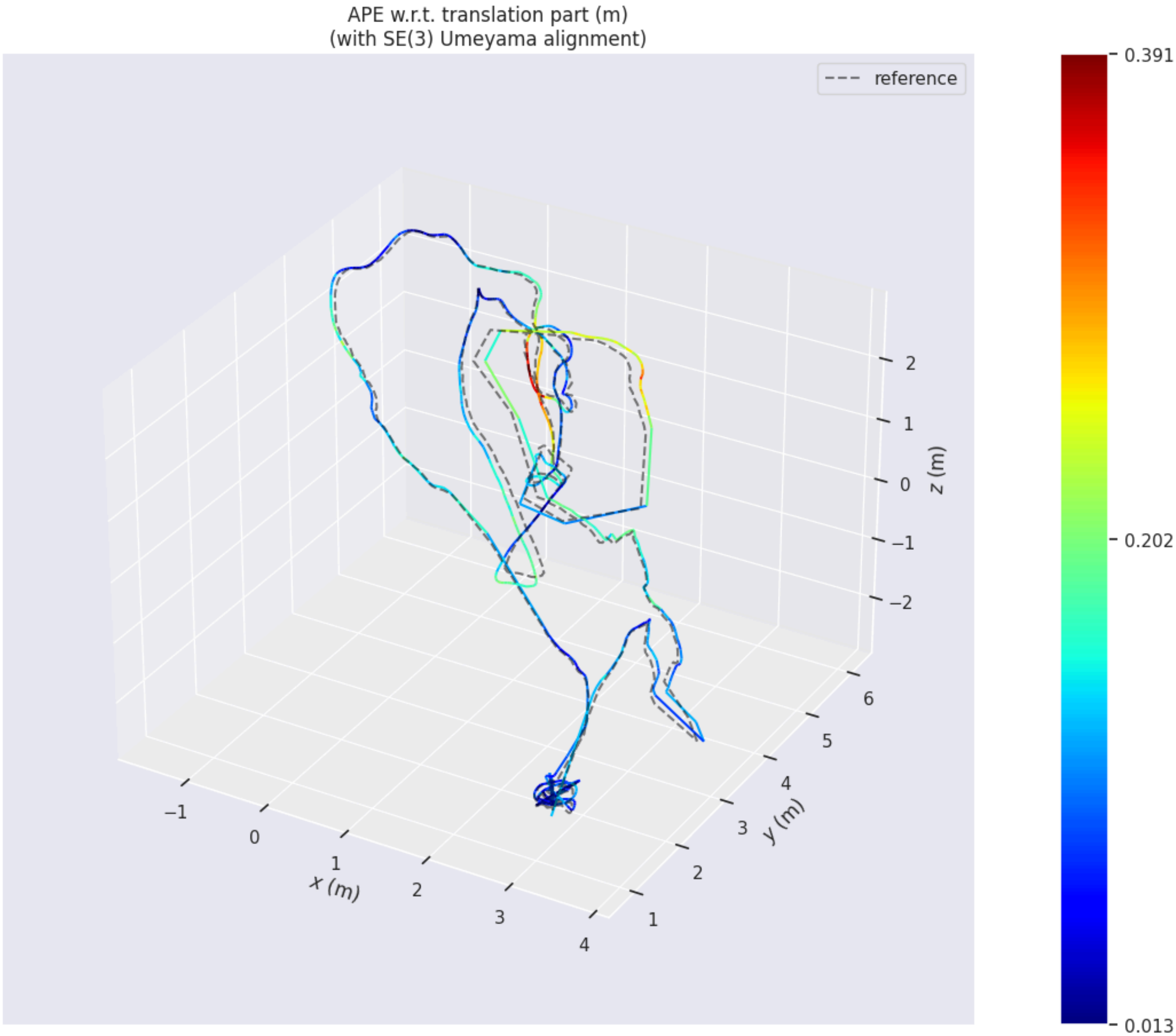
Metric: Relative Pose Error (RPE) - Translation Part

- **RMSE: 0.275 m**
- **Mean Error:** 0.137 m
- **Max Error:** 2.971 m

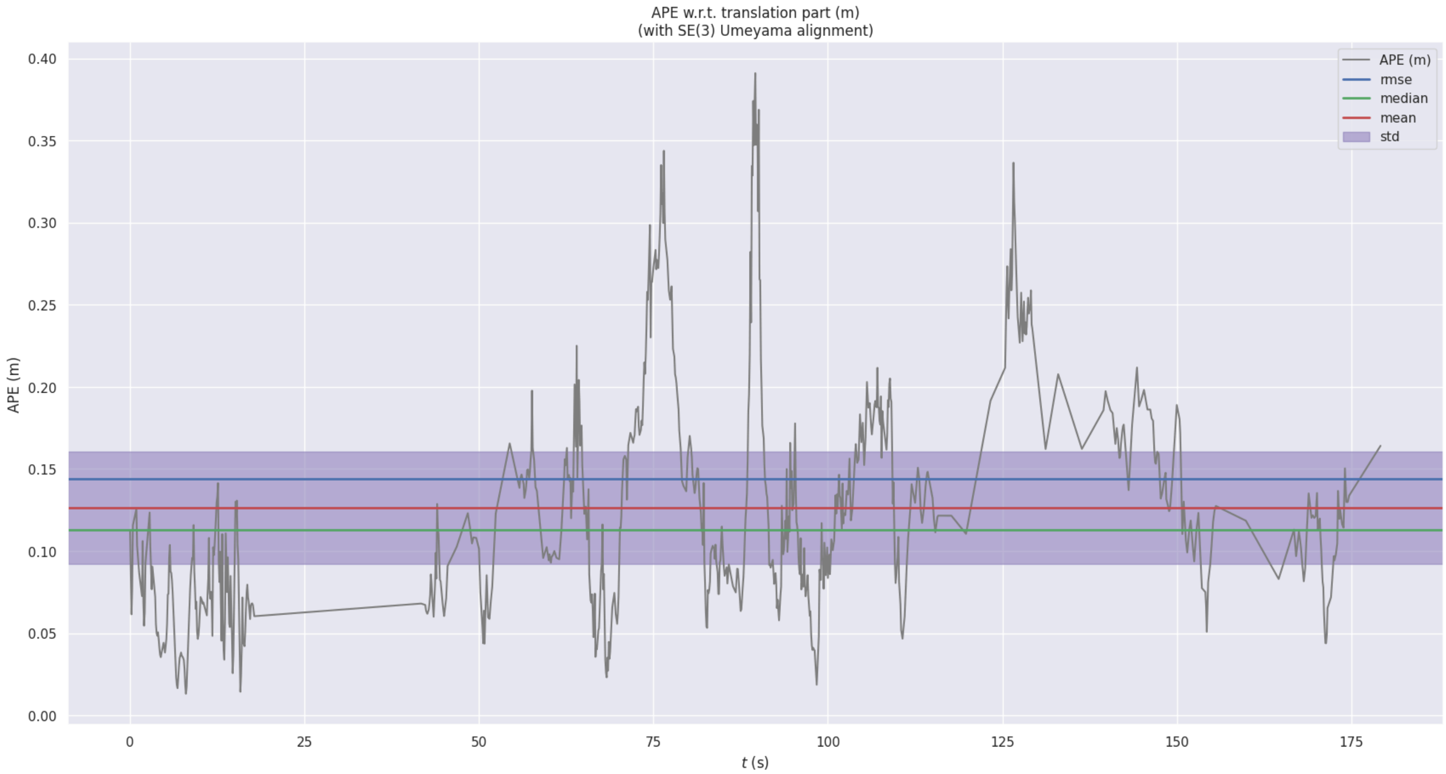
4.2. Visual Results

Trajectory Map (Estimated vs Ground Truth)

The estimated path (colored) aligns tightly with the ground truth (dashed grey).



Error Over Time



4.3. Root Cause Analysis

While the overall performance was excellent (RMSE < 15cm), the error plots reveal specific behaviors:

1. **Stable Tracking:** For the majority of the flight, the error remains below 0.15m.
2. **Error Spikes:** There are distinct spikes where error jumps to ~0.39m.
    - **Cause:** Correlating with the dataset video, these timestamps correspond to **aggressive yaw rotations**.
    - **Effect:** Rapid rotation causes motion blur, reducing the number of tracked visual features. The system temporarily relies more on IMU integration, which drifts faster than visual solving.
    - **Recovery:** The system successfully recovers immediately after the motion stabilizes, proving the robustness of the tightly-coupled fusion.
3. **RPE Outliers:** The high max RPE (2.97m) is likely due to timestamp synchronization jitter between the 10Hz VINS output and the 16Hz Ground Truth, rather than a massive tracking failure, as the ATE remains bounded.

5. Phase D: Engineering Improvement Proposal

Identified Limitation: Constant Time Offset Assumption

The current configuration assumes a fixed, pre-calibrated time offset (t_d) between the camera and IMU. In real-world deployment, trigger delays can drift due to CPU load or thermal effects on the shutter.

Proposed Solution: Online Temporal Calibration

I propose extending the VINS state vector to estimate the time offset t_d online.

Technical Approach:

1. **State Augmentation:** Add t_d to the state vector X:

$$X = [x_0, x_1, ...x_n, x_c^b, \lambda_0, ...\lambda_m, t_d]$$

2. **Feature Interpolation:** Modify the visual residual function. Instead of using the feature measurement at integer timestamp t, interpolate the feature position to $t + t_d$ using the feature velocity $v(t)$:

$$p_{corrected} = p(t) + v(t) \cdot t_d$$

3. **Optimization:** Include the derivative of the residual w.r.t t_d in the Jacobian during the non-linear optimization step.

Expected Benefit: This allows the system to compensate for synchronization latency (up to ~30ms) dynamically, significantly improving robustness during high-speed maneuvers where timing errors cause the largest residuals.