# Visual–Inertial Odometry / SLAM Assignment — Technical Report

**Project repository:** `bigalex95/VIO-SLAM-Assignment`
**Framework used:** Basalt (visual–inertial odometry)
**Dataset:** EuRoC MAV (stereo + IMU + ground truth)
**Date:** 2025-12-26
**Author:** bigalex95

---

## Abstract

This project implements and evaluates a complete visual–inertial odometry (VIO) pipeline using the **Basalt** framework on the **EuRoC MAV** benchmark. The work includes: (1) a comparison of popular VIO/SLAM frameworks, (2) justification for choosing Basalt, (3) configuration and tuning based on available camera/IMU calibration and dataset sensor metadata, and (4) quantitative evaluation using trajectory error metrics. The final system achieves **centimeter-level accuracy** on representative EuRoC sequences, with **ATE RMSE of 0.069 m** on MH_01_easy and **0.054 m** on V1_03_difficult, demonstrating robust tracking under both baseline and aggressive-motion conditions.

---

## 1. Introduction

Visual–Inertial Odometry estimates a platform’s 6-DoF motion by fusing camera measurements (feature observations / photometric constraints) with inertial measurements (angular velocity and linear acceleration). Compared to pure visual odometry, IMU fusion significantly improves robustness during fast rotations, short-term motion blur, and low-texture intervals, while providing metric scale (especially in stereo-inertial settings).

### 1.1 Objectives

1. Build a reproducible VIO pipeline that runs on EuRoC MAV sequences.
2. Compare common VIO/SLAM frameworks and select one suitable for the assignment.
3. Tune configuration parameters using dataset-provided calibration and sensor specifications.
4. Evaluate trajectory accuracy using standard metrics and present results clearly.

---

## 2. Framework Comparison

Several open-source VIO/SLAM frameworks were considered. The evaluation focused on usability, reproducibility, dataset support, calibration requirements, and expected performance.

### 2.1 Evaluation Criteria

- **Sensor modalities supported:** mono/stereo, IMU integration, time synchronization assumptions
- **Calibration workflow:** camera intrinsics & distortion, camera–IMU extrinsics, IMU noise model
- **Reproducibility:** availability of dataset runners, reference configs, and consistent evaluation tooling
- **Robustness:** stability under aggressive motion and challenging visual conditions
- **Engineering overhead:** build complexity, dependency footprint, and runtime performance

### 2.2 Candidate Frameworks (Summary)

**ORB-SLAM3 (Visual / Visual–Inertial SLAM)**

- **Pros:** strong tracking; supports visual-inertial; widely used; optional loop closure.
- **Cons:** engineering overhead can be higher; reproducible VI performance may require careful tuning and pipeline alignment with dataset formats.

**VINS-Mono / VINS-Fusion (Optimization-based VIO)**

- **Pros:** classical baseline; widely referenced; often good results when tuned well.
- **Cons:** sensitive to parameterization (noise, time offset, feature tracking); ROS-centric workflows can add friction depending on environment.

**OKVIS (Optimization-based VIO)**

- **Pros:** accurate optimization pipeline; strong academic lineage.
- **Cons:** setup and calibration assumptions can be demanding; less “turnkey” for rapid iteration.

**Basalt (Visual–Inertial Odometry)**

- **Pros:** modern, efficient VIO; good dataset support; practical configuration; strong robustness; real-time oriented implementation.
- **Cons:** still requires correct calibration/extrinsics/timing; best performance depends on accurate IMU noise parameters.

---

## 3. Why Basalt Was Chosen

Basalt was selected as the main framework due to:

1. **Practical reproducibility:** straightforward dataset execution once calibration/config are correct.

2. **Strong tightly coupled VIO:** IMU constraints stabilize estimation during aggressive motion and reduce drift.
3. **Efficient implementation:** designed with real-time performance in mind.
4. **Clear separation of calibration and runtime parameters:** supports a clean "calibration file + dataset runner + config" workflow.
5. **Alignment with EuRoC sensor setup:** EuRoC provides stereo camera + IMU + ground truth, matching Basalt's intended use cases.

---

## 4. Sensor Data Used for Configuration and Tuning

The estimator's accuracy depends heavily on correct sensor modeling. The project uses EuRoC-provided metadata and calibration files, plus IMU datasheet-based parameters for the ADIS16448.

### 4.1 Camera Measurements (Stereo)

**Data:**

- Left/right camera image streams (grayscale), known resolution and frame rate (EuRoC standard).
- Camera intrinsics: `fx, fy, cx, cy`.
- Distortion model and coefficients.
- Stereo extrinsics (relative transform between cam0 and cam1).

**Use in tuning:**

- Intrinsics and distortion must match the dataset calibration to ensure correct projection and feature tracking.
- Stereo extrinsics determine triangulation consistency and metric scale.

## 4.2 IMU Measurements (ADIS16448)

**Data:**

- Gyroscope: angular velocity `ω` (rad/s or dataset-defined units).
- Accelerometer: linear acceleration `a`.

**Noise and bias parameters used (as configured):**

```
{
  "imu_update_rate": 200.0,
  "accel_noise_std": [0.016, 0.016, 0.016],
  "gyro_noise_std": [0.0001454441, 0.0001454441, 0.0001454441],
  "accel_bias_std": [0.0003, 0.0003, 0.0003],
  "gyro_bias_std": [0.000019394, 0.000019394, 0.000019394]
}
```

**Use in tuning:**

- These parameters control how strongly the estimator trusts inertial integration versus visual constraints.
- Incorrect values often manifest as drift, oscillations, or reduced robustness during aggressive motion.

## 4.3 Camera–IMU Extrinsics and Time Consistency

**Data:**

- Rigid transform between IMU and camera frame(s).
- Synchronized timestamps (EuRoC provides well-synchronized sensors, but evaluation still depends on consistent time handling).

**Use in tuning:**

- Extrinsics must be correct; small errors can produce systematic trajectory distortion.
- Good time consistency is especially important under high angular rates (e.g., V1 sequences).

---

# 5. Experimental Setup

## 5.1 Datasets and Sequences

Two EuRoC sequences were selected to cover both baseline and stress-test conditions:

1. **MH_01_easy** — baseline sequence
2. Character: moderate motion, relatively stable tracking conditions
3. **V1_03_difficult** — stress-test sequence
4. Character: aggressive motion (high angular rates), motion blur, rapid feature turnover

## 5.2 Execution Pipeline

The project provides a reproducible pipeline using Docker:

- **Production mode:** automated run (build → execute → evaluate → plots/statistics)
- **Development mode:** interactive container for debugging and optional GUI visualization

Outputs include:

- Estimated trajectories ( `results/trajectories/*.csv` )
- Processed ground truth ( `results/groundtruth/*.csv` )
- Evaluation metrics and plots ( `results/evaluation/...` )
- Additional logs/statistics ( `results/stats/...` )

---

# 6. Evaluation Methodology

## 6.1 Metrics

- **ATE (Absolute Trajectory Error):** global trajectory error after rigid alignment to ground truth.
- **RPE (Relative Pose Error):** local motion consistency / drift behavior.

Both are commonly used in VIO/SLAM benchmarking and supported by tools such as `evo`.

### 6.2 Alignment Considerations

To obtain meaningful ATE, the estimated and reference trajectories must be compared in the same coordinate frame. The typical approach is **SE(3) alignment** (rotation + translation). For stereo-inertial EuRoC, scale is metric, so scale correction is not required.

*(Note: include the exact evaluation command / script settings used in your implementation to make the methodology fully reproducible.)*

---

## 7. Results

### 7.1 Headline Quantitative Results

| Dataset | ATE RMSE (m) | RPE RMSE (m) | Duration (s) | Assessment |
|---|---|---|---|---|
| MH_01_easy | 0.069 | 0.023 | 181.9 | Excellent |
| V1_03_difficult | 0.054 | 0.035 | 104.7 | Excellent |

### 7.2 Discussion

**MH_01_easy (Baseline):**

- Achieves low global error (≈ 7 cm ATE RMSE) over a long sequence (~182 s).
- Low RPE indicates strong short-term motion consistency and stable tracking.

**V1_03_difficult (Stress Test):**

- Despite aggressive motion and challenging visual conditions, the system maintains robust tracking.

- Achieves ~5 cm ATE RMSE, demonstrating effective visual–inertial fusion and good calibration/noise modeling.

### 7.3 Comparison to Reference Methods (Context)

A reference comparison (as summarized in the repository) indicates that Basalt-based results are substantially better than a VINS-Mono baseline on these sequences, and competitive with visual SLAM systems that use loop closure (with the caveat that loop closure changes the problem setting by correcting drift globally).

---

## 8. Conclusions

This project demonstrates an end-to-end VIO pipeline using Basalt on the EuRoC MAV benchmark. Basalt was selected due to its strong tightly coupled visual–inertial estimation, practical configuration workflow, and reproducible dataset execution. Using correct stereo camera calibration, accurate camera–IMU extrinsics, and a carefully parameterized ADIS16448 IMU noise model, the system achieved **ATE RMSE in the 5–7 cm range** on representative sequences, including a difficult high-dynamics scenario.

---

## 9. Limitations and Future Work

- **Limited dataset coverage:** results currently reported for two EuRoC sequences; extend to more MH/V sequences to characterize robustness statistically.
- **Loop closure / relocalization:** current pipeline is VIO-only; adding loop closure can reduce long-horizon drift.
- **Parameter sensitivity study:** systematic sweeps for feature tracking thresholds, outlier thresholds, and keyframe policies.
- **Runtime profiling:** report FPS / CPU usage for reproducibility and deployment relevance.

---

## References


- Usenko, V. et al. *The Basalt Framework for Visual-Inertial Odometry*, CVPR 2020.
- Burri, M. et al. *The EuRoC micro aerial vehicle datasets*, IJRR 2016.