Visual-Inertial ES-EKF Implementation Journey

Date: January 17, 2026 **Status:** Operational / High Precision **Final Metric:** RMSE ~5.9cm (Visual-Inertial Mode)

Executive Summary

We started with an Error-State Kalman Filter (ES-EKF) that suffered from catastrophic divergence (t=17s) and massive bias saturation. Through forensic analysis of the belief states and Jacobian linearization, we stabilized the filter. We then isolated the Prediction step, identifying and resolving configuration conflicts, initialization timing errors, and physical lever-arm effects. The system now tracks trajectory with high fidelity.

Phase 1: The "Explosion" (Stability Fixes)

Symptom: The filter tracked briefly, then position variance exploded vertically at t=17s. Accelerometer Z-bias instantly saturated to -0.5.

1. The Jacobian Mismatch

- **Problem:** The batch_update calculated residuals using the *Current State* but calculated Jacobians using the *First-Estimate* (FEJ). As the robot turned, the gradient (H) no longer pointed "downhill" relative to the residual (z), causing the filter to add error instead of removing it.
- **Fix:** Aligned the Jacobian calculation to use the **Current State** rotation matrix (R_wb).

2. The Z-Axis Civil War

- **Problem:** The filter was trying to estimate Z-axis accelerometer bias while simultaneously enforcing a v_z=0 ground constraint. The filter absorbed gravity model errors into the bias, causing saturation.
- **Fix:** Disabled Z-axis bias estimation (P[12,12]=0, Q_ba[2]=0). This "locked" the vertical belief, stopping the internal conflict.

3. Covariance Consistency

- **Problem:** Manual zeroing of dx for unobserved states (Orientation/Bias) during vision updates broke the correlation chain in the P matrix.
- **Fix:** Removed manual zeroing. Allowed the EKF to update unobserved states via cross-correlation naturally.

Result: RMSE improved from Divergence -> **0.059m (5.9cm)**.

Phase 2: The "Prediction Only" Tests (Logic Fixes)

Symptom: When testing "Prediction Only," the robot refused to move in the estimator, or drifted wildly when rotating.

4. The ZUPT Paradox

- **Problem:** enable_zupt was left True in the config. When the robot moved slowly (start-up), ZUPT triggered, telling the filter v=0 while the accelerometer read a=1.0.
- **Outcome:** The filter "learned" a massive negative bias to mathematically stop the robot, destroying the model for subsequent motion.
- **Fix:** Logic separation. Disabled ZUPT when testing Prediction Mode.

Phase 3: The Physics of Gravity (Calibration Fixes)

Symptom: With ZUPT disabled, the robot drifted linearly backward (-x) at -0.22 m/s^2 while stationary.

5. The "Rocking" Initialization

- **Problem:** The initialize_from_imu function captured data immediately after Gazebo spawn. The robot was pitching up (~1.3°) while settling on its suspension. The filter learned "Nose Up" as "Flat."
- **Fix (Hardware):** You adjusted the **caster wheel size** to mechanically level the robot, aligning the physical chassis with the gravity model.
- **Fix (Software - Alternate):** Implemented a "Settling Time" delay to discard the first 2 seconds of IMU data.

Phase 4: The Turning Drift (Dynamics Fixes)

Symptom: Robot tracked well straight, but drifted sideways (y-axis) during turns.

6. The Lever Arm Effect

- **Problem:** The IMU was not located at the robot's Center of Rotation (CoR). The accelerometer measured Centripetal (\omega \times \omega \times r) and Tangential (\alpha \times r) forces. The EKF interpreted this "swinging" force as linear sliding.
- **Fix:** Identified the need for a physical transform correction (r_b_i) in the prediction step to subtract rotation-induced forces before integration.

Current Status & Next Steps

The filter is now mathematically consistent and physically grounded.
Next Validation:

- **Synthetic Measurement Test:** Implement the "Frozen Jacobian" / Fixed Speed prediction mode to validate the Vision Update logic in isolation, ensuring that camera measurements correctly "pull" the estimate without fighting the IMU dynamics.