Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Decentralized Coordination and Acoustic Source Localization in Autonomous Drone Swarms

Md Shahanur Islam Shagor

shagormd.s.i@iunp.edu.rs

Voronezh State University of Forestry and Technologies https://orcid.org/0009-0003-7730-3202

Md Shohorab Hossain Shehab

Daffodil International University

Research Article

Keywords: GPS-denied navigation, visual-inertial odometry, error-state Kalman filter, MSCKF, first-estimates Jacobian, UAV autonomy

Posted Date: August 26th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10812536/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Toward Secure Decentralized Coordination and Acoustic Source Localization in Autonomous Drone Swarms

Md Shahanur Islam Shagor[1],* · Md Shohorab Hossain Shehab[2]

[1] Department of Computer Science and Microelectronics Engineering, Voronezh State University of Forestry and Technologies, Voronezh, Russia

[2] Daffodil International University, Dhaka, Bangladesh

* Corresponding author: shagormd.s.i@iunp.edu.rs | Co-author: hossain242-35-232@diu.edu.bd

Abstract

GPS-denied unmanned aerial vehicles require more than a nominally accurate state estimator: the estimator must preserve observability, reject numerically unsafe updates, expose uncertainty, and remain reproducible across restarts and software revisions. This paper develops a complete mathematical and systems analysis of a consistency-aware visual-inertial estimation architecture derived from the current v2.0.0 software research release of an open GPS-denied UAV platform. The estimator uses a 15-dimensional error state with midpoint inertial propagation, explicit process-noise discretization, first-estimates Jacobians (FEJ), a bounded multi-state constraint Kalman filter (MSCKF) camera window, multi-view triangulation, SVD-based feature-state elimination, normalized-innovation chi-square gating, Joseph-form covariance correction, zero-velocity updates, and fail-closed transactional commit semantics. A new spectral geometry diagnostic is introduced to separate low image residual from informative triangulation geometry. In a reproducible three-view Monte Carlo study using the implementation noise level of 1.5 pixels, a 0.10 m camera span can satisfy the existing reprojection gate while accepted landmarks still exhibit 1.19 m mean three-dimensional error, demonstrating why residual-only confidence is insufficient. Analytical scaling shows that eight camera clones produce a 63-dimensional augmented error state and a 31.01 KiB dense covariance. Repository replay, compiler, sanitizer, software-HIL, and latency artifacts are evaluated conservatively, with software evidence explicitly separated from physical-flight claims.

Keywords: GPS-denied navigation; visual-inertial odometry; error-state Kalman filter; MSCKF; first-estimates Jacobian; UAV autonomy

1 Introduction

Autonomous aerial robots increasingly operate in locations where Global Navigation Satellite System (GNSS) measurements are absent, degraded, reflected, jammed, or deliberately spoofed. Indoor inspection, disaster response, warehouse operation, subterranean search, dense urban flight, and close-quarter industrial inspection all require a vehicle to continue estimating its motion when satellite positioning is unavailable. In such settings, the localization problem is not simply to replace GNSS with another sensor. The robot must combine high-rate inertial motion information with exteroceptive observations while retaining a defensible representation of uncertainty and a safe behavior when those observations become geometrically weak or numerically inconsistent.

Visual-inertial odometry (VIO) is attractive because a camera and an inertial measurement unit (IMU) provide complementary information. The IMU resolves rapid rotational and translational dynamics but accumulates bias-driven drift; the camera supplies geometric constraints that can arrest that drift but is vulnerable to motion blur, low texture, illumination change, dynamic objects, and limited field of view. The practical estimator therefore sits at the intersection of nonlinear estimation, numerical linear algebra, real-time software, and safety-aware autonomy. The correctness of the overall navigation stack depends on all four.

Extended Kalman filter (EKF) VIO has a long research history. The multi-state constraint Kalman filter (MSCKF) avoids maintaining every visual landmark as a persistent state by expressing feature tracks as constraints across a bounded window of camera poses [2, 3]. This structure is computationally attractive for embedded aerial platforms. However, classical EKF linearization can create inconsistency by injecting artificial information in unobservable directions [4, 5]. First-estimates Jacobian (FEJ) methods address this by evaluating selected Jacobians at frozen first estimates rather than repeatedly relinearizing the same variable at updated estimates. A robust implementation must also guard covariance symmetry, positive semi-definiteness, timestamp ordering, innovation conditioning, and error-state reset after quaternion injection.

A recurring weakness in practical GPS-denied systems is the interpretation of a small image residual as evidence of accurate localization. A small residual only states that a measurement is compatible with the current linearized model. It does not guarantee that the underlying geometry is informative. Two or three nearly co-located camera views can produce low reprojection error while the recovered feature depth remains poorly conditioned. Likewise, a filter can produce a finite state while its covariance becomes asymmetric, indefinite, or over-confident. These failure modes motivate a geometry-before-confidence principle: residual magnitude should be interpreted only together with observability, triangulation conditioning, innovation covariance, and state uncertainty.

The open software platform analyzed in this study has evolved substantially beyond an earlier GPS-denied prototype. Its current v2.0.0 research baseline includes a C++20 autonomy and sensor-fusion runtime, a Go supervisory control plane, a Python/PySide6 operator interface, deterministic simulation and software-HIL tooling, failure injection, an active/shadow estimator architecture, error-state estimator hardening, stationary detection and ZUPT, FEJ support, a bounded MSCKF camera-state window, feature triangulation, structureless feature constraints, covariance-aware camera-state retirement, and cross-platform validation infrastructure. The present paper does not treat the entire repository as a single monolithic contribution. Instead, it isolates the estimation mathematics and the evidence needed to support claims about that mathematics.

This distinction is essential for intelligent robotics. Deployment manifests, dashboard screens, safety managers, network protocols, or AI modules can coexist in the same repository without proving localization accuracy or flight readiness. Conversely, an estimator can be mathematically coherent without proving that it survives rotor vibration, sensor desynchronization, electromagnetic interference, radio congestion, or aggressive flight. This paper therefore uses explicit evidence classes and does not upgrade software-only validation into an airworthiness claim.

The central thesis is that a GPS-denied estimator should be modeled as a sequence of transactional state transitions. Each propagation or correction begins from a committed state, constructs a candidate state and covariance, validates numerical and

geometric invariants, and commits only when all required predicates hold. The idea is simple but has strong consequences: invalid timestamps do not partially advance the state, singular innovation matrices do not publish corrections, failed covariance checks do not leak into subsequent cycles, and shadow-estimator experiments cannot silently change the active flight-authority estimate.

This paper makes six contributions.

- a complete implementation-consistent derivation of the 15-D error-state inertial propagation and correction model, including midpoint attitude propagation, process-noise discretization, Joseph-form covariance update, and reset Jacobians;
- a mathematical reconstruction of the current FEJ-MSCKF path, including camera cloning, covariance augmentation, triangulation, normalized feature residuals, SVD null-space projection, chi-square gating, stacked updates, and fail-closed camera-state retirement;
- a new spectral triangulation diagnostic, $\gamma A = \lambda\min(A)/\lambda\max(A)$, that quantifies geometric strength independently of reprojection residual and exposes cases in which the existing baseline/reprojection gates accept weak depth geometry;
- a reproducible Monte Carlo sensitivity analysis, analytic state-dimension and memory bounds, projected-measurement dimension calculations, chi-square thresholds, process-noise growth calculations, and a closed-form ZUPT variance example based on the implementation defaults;
- a conservative synthesis of deterministic replay, compiler, sanitizer, software-HIL, and latency artifacts from the current repository revision, including explicit treatment of a stale Phase 22.1 validation report that conflicts with later release-level completion metadata; and
- a claim boundary for future physical validation, showing exactly which additional experiments are required before the software evidence can support statements about real UAV localization accuracy, radio behavior, or free-flight safety.

The remainder of the paper is organized as follows. Section 2 reviews visual-inertial filtering, observability, GPS-denied robotics, and geometry-aware validation. Section 3 defines the evaluated software revision, safety boundary, evidence model, and research questions. Sections 4 and 5 derive the inertial, error-state, FEJ, and MSCKF mathematics. Section 6 places the estimator in the wider GPS-denied navigation and dynamic-environment safety context. Section 7 presents analytical and numerical evaluation. Section 8 evaluates repository evidence. Sections 9–11 discuss implications, limitations, reproducibility, and conclusions.

2 Related work

2.1 Visual-inertial filtering and smoothing

The Kalman filter established the recursive minimum-variance framework for linear Gaussian state estimation [1]. Modern VIO applies nonlinear extensions of that idea to rigid-body motion on manifolds. Mourikis and Roumeliotis introduced the MSCKF, in which visual features that persist across several camera poses constrain a sliding window of cloned poses without becoming permanent states [2]. Li and Mourikis later analyzed consistency and calibration effects in high-precision EKF VIO [3]. These methods remain relevant for embedded aerial robots because they offer bounded state growth and predictable recursive updates.

Optimization-based VIO systems instead retain a window of past states and repeatedly optimize a nonlinear objective. OKVIS uses keyframe-based nonlinear optimization [6], while VINS-Mono combines preintegrated inertial terms, visual reprojection, initialization, failure recovery, and pose-graph functionality [9]. IMU preintegration on manifolds reduces the cost of repeatedly integrating high-rate inertial measurements between keyframes [7]. ORB-SLAM3 extends visual and visual-inertial SLAM across multiple maps [10]. Filtering and smoothing should not be viewed as mutually exclusive philosophies; they expose different trade-offs among relinearization accuracy, memory, computational structure, and timing determinism.

Direct filtering approaches such as ROVIO use photometric residuals rather than only feature coordinates [8]. OpenVINS provides a modular research implementation of sliding-window visual-inertial filtering, online calibration, FEJ treatments, and simulation/evaluation tools [11]. These systems demonstrate that a modern estimator is not only a set of equations: calibration, timestamp handling, state management, measurement gating, and reproducibility infrastructure strongly influence real behavior.

2.2 Observability, FEJ, and consistency

EKF inconsistency is particularly important in navigation systems with gauge freedoms. Huang, Mourikis, and Roumeliotis showed that repeatedly evaluating Jacobians at updated estimates can make the linearized system appear more observable than the underlying nonlinear system [4]. Hesch et al. extended observability-based consistency analysis to vision-aided inertial navigation and showed how spurious information can accumulate along unobservable directions [5]. A filter may then report a covariance that is smaller than the actual estimation error, creating over-confidence rather than simply random noise.

FEJ addresses this problem by freezing selected linearization points at the first estimate used for a variable. For a camera clone or feature, the measurement residual can still be evaluated at the current nominal estimate, while the Jacobian is evaluated at the stored first estimate. This separation is important: the residual expresses current disagreement, whereas the Jacobian preserves a consistent local model of how that observation constrains the state. The analyzed implementation stores first-estimate position and orientation for cloned camera states and uses them when FEJ is enabled.

2.3 Calibration, timing, and benchmark evidence

Visual-inertial fusion depends on calibrated spatial and temporal relationships between camera and IMU. Mirzaei and Roumeliotis analyzed IMU-camera calibration and its observability [15], while Li and Mourikis treated camera-IMU time offset as an estimated quantity and examined its identifiability [14]. These works are especially relevant to aerial robotics because even a strong estimator can be destabilized by biased extrinsics or asynchronous measurements.

Public benchmarks such as EuRoC [12] and TUM VI [13] provide synchronized visual-inertial data and ground truth for comparative trajectory evaluation. The repository studied here does not yet contain a completed EuRoC/TUM-VI accuracy benchmark for the current shadow ESKF-MSCKF path. Consequently, this paper does not fabricate absolute trajectory RMSE or claim superiority over VINS-Mono, OpenVINS, or ORB-SLAM3. Instead, it focuses on mathematical correctness, numerical failure handling, deterministic replay, and the analytical conditions under which a measurement should be trusted. A public-dataset comparison is identified as a required next experiment.

2.4 Zero-velocity updates and complementary localization

Inertial drift can be reduced during stationary intervals by adding a zero-velocity observation. Zero-velocity-aided inertial navigation has been widely studied for pedestrian and wearable systems [16, 17], but the same principle is useful in aerial systems during known stationary bench periods, landing, or tightly controlled hover-like conditions when the detector assumptions are valid. The current implementation uses a hysteretic stationary detector with separate entry and exit thresholds and rate-limited ZUPT updates.

UWB ranging and time-difference-of-arrival localization provide complementary absolute or semi-absolute information under GNSS denial. UWB ranging performance in multipath environments has been extensively characterized [18], while hyperbolic TDOA localization admits closed-form and iterative estimators [19]. These modalities are valuable when visual geometry collapses, but they introduce their own synchronization and geometry requirements. This paper includes them as complementary system models rather than as the main evaluated estimator.

2.5 Dynamic-environment UAV autonomy and IJIRA context

GPS-denied navigation becomes more demanding when the environment contains moving obstacles, narrow corridors, smoke, repetitive texture, or illumination changes. Velocity-obstacle and reciprocal collision-avoidance formulations provide geometric tools for short-horizon collision reasoning [20, 21]. Multi-UAV systems add communication and coordination concerns [22, 23], while broader UAV surveys emphasize sensing, communications, safety, and regulatory challenges [24].

Recent work in the International Journal of Intelligent Robotics and Applications has evaluated vision-based quadrotor navigation in constrained GPS-denied environments [25], intelligent close-quarter aerial inspection [26], and the broader evolution of robot localization from classical estimators toward intelligent autonomy [27]. The journal has also published EKF localization implementations for mobile robots [28] and surveys of localization methods in other difficult domains such as underwater robotics [29]. The present study is positioned within this systems-oriented robotics context: its novelty is not a new camera detector or a new optimizer, but a mathematically explicit, fail-closed, consistency-aware estimator and evidence framework for GPS-denied UAV software.

2.6 Research gap

The literature establishes strong theory for VIO, FEJ, filtering, smoothing, calibration, and benchmarks. A remaining engineering-research gap lies in connecting that theory to a continuously evolving autonomy codebase without overstating evidence. Four specific gaps motivate this paper: (i) estimator equations and source code often diverge as software evolves; (ii) residual-based confidence can ignore geometric degeneracy; (iii) numerical checks are frequently documented as implementation details rather than as explicit estimator invariants; and (iv) CI, replay, software-HIL, and field evidence are often blended into a single “validated” label. The methodology below is designed to close these gaps at the software-research level.

3 System context, scope, and research design

3.1 Evaluated platform revision

The evaluated platform is the public repository “UVA-GPS-Denied-Navigation-in-Dynamic-Environments” at the current main-branch revision c040073969a6111f4c652d08e8de8629d70b2317, merged on 21 August 2026, together with the v2.0.0 software research release documentation [36]. The repository combines C++20 onboard autonomy and sensor fusion, a Go supervisory backend, Python/PySide6 operator tooling, simulation and software-HIL scripts, AI/autonomy modules, deployment assets, and a staged estimator-hardening program. The v2.0.0 release explicitly states that it is a software/research release and does not claim physical HIL, production-radio qualification, tethered or free flight, or operational airworthiness.

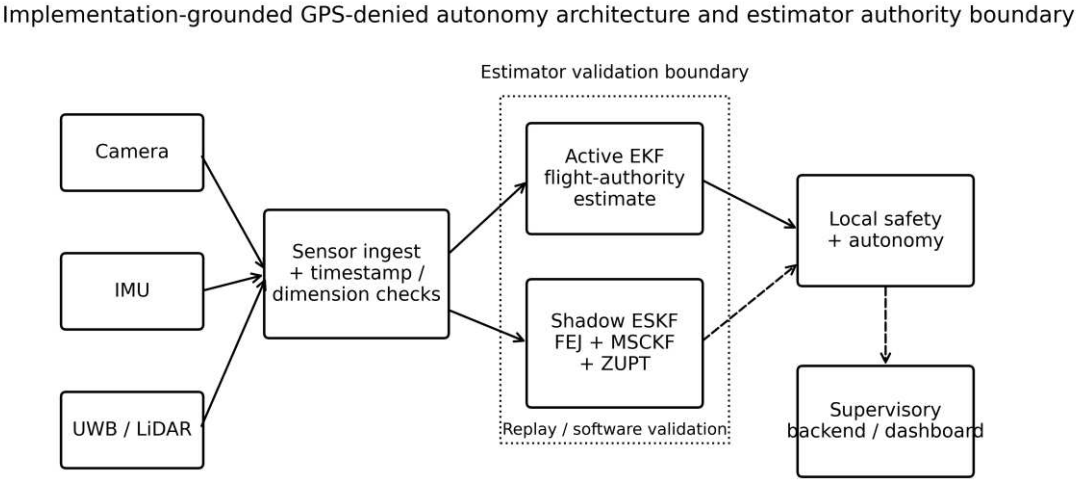


Fig. 1 Implementation-grounded GPS-denied autonomy architecture and estimator authority boundary

The estimator path uses an active/shadow architecture. The existing active EKF remains authoritative, while the newer error-state estimator is evaluated in a shadow path. Stages 17–22 incrementally add error-state hardening, stationary detection and ZUPT, FEJ, MSCKF camera cloning, triangulation, and feature-constraint updates. The release documentation deliberately

excludes automatic estimator promotion. This separation is retained in the paper: mathematical analysis of the shadow estimator is not presented as proof that it is already the flight-authority estimator.

3.2 Evidence classes

Table 1 Evidence classes used in this paper

Class	Meaning	What it can support	What it cannot support
E1 — source mathematics	Equations reconstructed from current implementation and configuration	Implementation semantics, dimensions, gates, transaction logic	Flight accuracy or physical sensor robustness
E2 — deterministic software	Unit, replay, regression, compiler and sanitizer artifacts	Repeatability, failure-path behavior, numerical regression evidence	Hardware timing, RF behavior, airworthiness
E3 — software-HIL / benchmark	Repository mission harness and wall-clock software measurements	Software path latency and mission-state propagation in the tested environment	Actuator-level closed-loop timing or free-flight performance
E4 — paper-specific numerical analysis	Reproducible calculations derived from published source parameters	Geometry sensitivity, scaling, gate thresholds, illustrative uncertainty effects	Empirical field accuracy unless validated on ground-truth data
E5 — deferred physical evidence	Future synchronized HIL, tethered and free-flight trials	Potential operational claims after completion	Not available in the current study

3.3 Research questions

RQ1. Can the current estimator be expressed as a self-consistent 15-D error-state system in which propagation and correction are committed only after finite-state, quaternion, covariance, innovation, and dimension checks pass?

RQ2. How do FEJ, multi-view triangulation, MSCKF null-space projection, statistical gating, and bounded camera-state retirement interact, and where can geometric degeneracy remain hidden behind a small reprojection residual?

RQ3. What are the quantitative state-dimension, covariance-memory, measurement-dimension, process-noise, and ZUPT uncertainty consequences of the current configuration?

RQ4. What do the current deterministic replay, compiler/sanitizer, software-HIL, and latency artifacts actually establish, and which claims remain unsupported without ground-truth and physical-flight experiments?

3.4 Fail-closed estimator contract

Let Xk denote the committed estimator state immediately before measurement k. Processing produces a candidate state $\tilde{X}k$ only after deterministic input and model checks. The public state transition is therefore not "always apply the Kalman equations," but a guarded mapping:

$$X_k^+ = \begin{cases} \tilde{X}_k, & \mathcal{V}\left(\tilde{X}_k, \tilde{P}_k, S_k, z_k\right) = \text{true}, \\ X_k^-, & \text{otherwise}. \end{cases} \tag{1}$$

The validity predicate $\mathcal{V}$ contains the implemented checks for finite values, timestamp monotonicity and time-step bounds, quaternion norm, covariance symmetry and non-negative variance, compatible dimensions, innovation-covariance conditioning, successful factorization, and—in the visual path—geometric and statistical gates. The transaction model in Fig. 2 is used throughout the derivation.

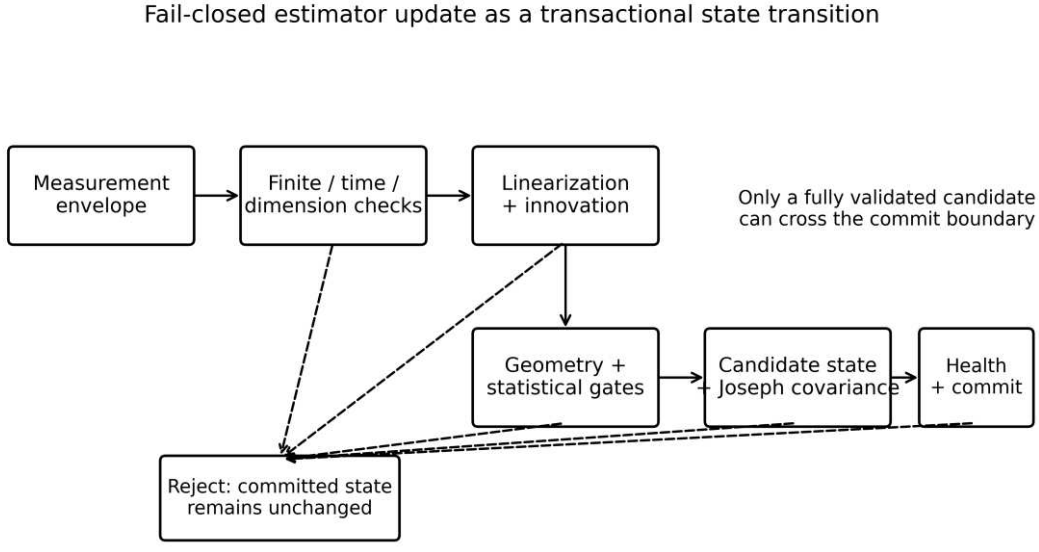


Fig. 2 Fail-closed estimator update as a transactional state transition

4 Error-state inertial estimation

4.1 Frames, nominal state, and error state

Let W denote the local world frame and B the IMU/body frame. The quaternion qWB rotates a body-frame vector into W. The nominal state contains position p, velocity v, quaternion q, accelerometer bias ba, and gyroscope bias bg:

$$\mathbf{x} = \left[\mathbf{p}^T\ \mathbf{v}^T\ \mathbf{q}_{WB}^T\ \mathbf{b}_a^T\ \mathbf{b}_g^T\right]^T \tag{2}$$

Because a unit quaternion has one constraint, the estimator uses a 15-D local error state:

$$\delta\mathbf{x} = \left[\delta\mathbf{p}^T\ \delta\mathbf{v}^T\ \delta\boldsymbol{\theta}^T\ \delta\mathbf{b}_a^T\ \delta\mathbf{b}_g^T\right]^T \in \mathbb{R}^{15} \tag{3}$$

The default initial standard deviations are 0.10 m for position, 0.05 m s−1 for velocity, 0.05 rad for attitude error, 0.01 m s−2 for accelerometer bias, and 0.001 rad s−1 for gyroscope bias. The base covariance P0 is therefore block diagonal under the default reset.

$$\mathbf{P}_0 = \mathrm{diag}(0.10^2\mathbf{I}_3,\ 0.05^2\mathbf{I}_3,\ 0.05^2\mathbf{I}_3,\ 0.01^2\mathbf{I}_3,\ 0.001^2\mathbf{I}_3) \tag{4}$$

4.2 IMU measurement model

The measured specific force am and angular rate ωm are modeled as the true quantities corrupted by bias and white noise:

$$\mathbf{a}_m = \mathbf{a}_{true} + \mathbf{b}_a + \mathbf{n}_a, \qquad \boldsymbol{\omega}_m = \boldsymbol{\omega}_{true} + \mathbf{b}_g + \mathbf{n}_g \tag{5}$$

The corrected inputs used in propagation are

$$\mathbf{a} = \mathbf{a}_m - \mathbf{b}_a, \qquad \boldsymbol{\omega} = \boldsymbol{\omega}_m - \mathbf{b}_g \tag{6}$$

The implementation uses g = [0, 0, −9.81]ᵀ m s−2 in W. With R(q) the body-to-world rotation matrix, the nominal continuous dynamics are

$$\dot{\mathbf{p}} = \mathbf{v}, \quad \dot{\mathbf{v}} = \mathbf{R}(q)\mathbf{a} + \mathbf{g}, \quad \dot{q} = 1/2\,\Omega(\boldsymbol{\omega})q, \quad \dot{\mathbf{b}}_a = \mathbf{n}_{ba}, \quad \dot{\mathbf{b}}_g = \mathbf{n}_{bg} \tag{7}$$

4.3 Midpoint inertial propagation

For an accepted interval Δt, the implementation evaluates translational acceleration at a midpoint orientation. A half-step quaternion is first propagated from qk using the bias-corrected angular rate:

$$q_{k+\frac{1}{2}} = q_k \otimes \mathrm{Exp}_q(1/2\,\boldsymbol{\omega}_k\Delta t) \tag{8}$$

The midpoint world acceleration is

$$\mathbf{a}_{W,k+\frac{1}{2}} = \mathbf{R}\left(q_{k+\frac{1}{2}}\right)\mathbf{a}_k + \mathbf{g} \tag{9}$$

The nominal position, velocity, and attitude updates are then

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \mathbf{v}_k\Delta t + 1/2\,\mathbf{a}_{W,k+\frac{1}{2}}\Delta t^2 \tag{10}$$

$$\mathbf{v}_{k+1} = \mathbf{v}_k + \mathbf{a}_{W,k+\frac{1}{2}}\Delta t \tag{11}$$

$$q_{k+1} = \mathrm{normalize}\left(q_k \otimes \mathrm{Exp}_q(\boldsymbol{\omega}_k\Delta t)\right) \tag{12}$$

The time step is not accepted blindly. A timestamp must be finite and, after the first sample, must be non-duplicate, monotonic when required, and satisfy 10−6 s ≤ Δt ≤ 0.1 s under the current validation defaults. A rejected sample does not update the last accepted timestamp, which prevents a bad packet from shifting the timing origin for the next valid packet.

4.4 Continuous error dynamics

Using [u]× for the 3×3 skew-symmetric matrix of vector u, the implementation constructs the continuous error-state dynamics matrix Fc. With R = R(qk+1/2), its non-zero blocks are

$$F_{p,v} = I_3, \quad F_{v,\theta} = -R[a]_\times, \quad F_{v,b_a} = -R, \quad F_{\theta,\theta} = -[\omega]_\times, \quad F_{\theta,b_g} = -I_3 \tag{13}$$

The noise mapping Gc acts on the 12-D vector w = [naᵀ ngᵀ nbaᵀ nbgᵀ]ᵀ:

$$G_{v,n_a} = -R, \quad G_{\theta,n_g} = -I_3, \quad G_{b_a,n_{ba}} = I_3, \quad G_{b_g,n_{bg}} = I_3 \tag{14}$$

The continuous IMU noise covariance uses the configured standard deviations σna=0.02 m s−2, σng=0.005 rad s−1, σnba=10−4, and σnbg=10−5:

$$Q_{imu} = \mathrm{diag}\left(\sigma_{na}^2 I_3,\ \sigma_{ng}^2 I_3,\ \sigma_{nba}^2 I_3,\ \sigma_{nbg}^2 I_3\right) \tag{15}$$

4.5 Discrete transition and process covariance

The discrete transition begins with a first-order approximation Φ = I + FcΔt, after which the implementation explicitly inserts second-order position couplings:

$$\Phi_{p,v} = I_3\Delta t, \quad \Phi_{p,\theta} = -1/2\,R[a]_\times\Delta t^2, \quad \Phi_{p,b_a} = -1/2\,R\Delta t^2 \tag{16}$$

$$\Phi_{v,\theta} = -R[a]_\times\Delta t, \quad \Phi_{v,b_a} = -R\Delta t, \quad \Phi_{\theta,b_g} = -I_3\Delta t \tag{17}$$

The mapped continuous covariance and discrete approximation are

$$Q_c = G_c Q_{imu} G_c^T \tag{18}$$

$$Q_d = \Phi Q_c \Phi^T\Delta t + Q_{pos}(\Delta t) \tag{19}$$

Qpos adds the implementation-specific acceleration-noise terms 0.25σna²Δt4 to the position diagonal and 0.5σna²Δt3 to the position-velocity cross block. The base covariance propagation is

$$P_{k+1}^{-} = \Phi P_k^{+} \Phi^T + Q_d \tag{20}$$

When camera clones are present, the same base transition is embedded in an augmented identity transition; process noise is added only to the base 15-D block. The augmented covariance is symmetrized after propagation before it is eligible for commit.

4.6 Generic correction and innovation conditioning

For a measurement residual r with Jacobian H and measurement covariance Rz, the innovation covariance is

$$S = HP^{-}H^T + R_z \tag{21}$$

Before factorization, the implementation checks that S is finite, symmetric within tolerance, positive definite above a minimum eigenvalue, and not excessively ill-conditioned. Denoting eigenvalues of the symmetrized S by λi, the condition number is

$$\kappa(S) = \frac{\lambda_{max}(S)}{\lambda_{min}(S)} \tag{22}$$

The current source rejects S when λmin≤10−12 or κ(S)>10^12. The Kalman gain is obtained by an LDLT solve rather than an explicit matrix inverse:

$$K = P^{-}H^T S^{-1} \tag{23}$$

$$\delta x = Kr \tag{24}$$

4.7 Joseph covariance form and reset

A numerically fragile covariance correction is P+=(I−KH)P−. The implementation instead uses Joseph form, which preserves symmetry and positive semi-definiteness more robustly under finite precision:

$$P_J = (I - KH)P^{-}(I - KH)^T + KR_zK^T \tag{25}$$

The attitude error is injected multiplicatively into the nominal quaternion:

$$q^{+} = \text{normalize}\left(q^{-} \otimes \text{Exp}_q(\delta\theta)\right) \tag{26}$$

After injection, the local attitude-error coordinates change. The first-order reset Jacobian is

$$G_{reset}(\delta\theta) = I_3 - 1/2\,[\delta\theta]_\times \tag{27}$$

For an augmented state, the reset Jacobian contains one block for the base attitude and one for every camera-clone attitude. The covariance becomes

$$P^{+} = J_{reset} P_J J_{reset}^T \tag{28}$$

Only after the candidate nominal state, clone states, and covariance pass finite-value, quaternion, symmetry, and non-negativity checks does the estimator commit them. Otherwise the pre-update state is retained.

4.8 Stationary detection and ZUPT

The stationary detector uses the difference between measured acceleration magnitude and gravity together with gyroscope magnitude. Let

$$e_a = |\| a_m \|_2 - g|, \qquad e_\omega = \| \omega_m \|_2 \tag{29}$$

An entry candidate satisfies ea≤0.15 m s−2 and eω≤0.02 rad s−1. The current window contains 20 samples; at least 16 must satisfy the entry condition for at least 0.15 s. Exit uses looser thresholds of 0.25 m s−2 and 0.04 rad s−1 and an exit count of 10, creating hysteresis.

When stationary, the zero-velocity measurement is

$$z_v = 0 = v + n_v, \qquad R_v = \sigma_v^2 I_3, \qquad \sigma_v = 0.01 \text{ m s}^{-1} \tag{30}$$

$$r_v = -v, \qquad H_v = [0_3\ I_3\ 0_3\ 0_3\ 0_3] \tag{31}$$

Automatic ZUPT is rate limited to at most 10 Hz. In the decoupled scalar case with prior velocity variance Pvv and measurement variance Rv, the posterior variance is

$$P_{vv}^{+} = \frac{P_{vv}^{-} R_v}{P_{vv}^{-} + R_v} \tag{32}$$

5 FEJ-MSCKF visual constraints and geometry-aware gating

5.1 Camera model and current visual residual

For a world point pf observed by a camera clone with position pc and orientation qwc, the point in the camera frame is

$$p_f^C = R(q_{wc})^T \left(p_f^W - p_c^W\right) = [X\ Y\ Z]^T \tag{33}$$

For pinhole intrinsics fx, fy, cx, cy, the predicted pixel is

$$\pi\left(p_f^C\right) = \left[f_x X/Z + c_x,\ f_y Y/Z + c_y\right]^T \tag{34}$$

The base visual update uses pixel residuals, while the MSCKF feature-constraint path stores normalized bearings and forms a normalized residual. The projection Jacobian for normalized coordinates is

$$J_\pi = \begin{bmatrix} 1/Z & 0 & -X/Z^2 \\ 0 & 1/Z & -Y/Z^2 \end{bmatrix} \tag{35}$$

5.2 First-estimates Jacobian

Let ($\bar{p}$c,$\bar{q}$c) be the first estimate stored when a camera state is cloned. FEJ evaluates the measurement Jacobian at the frozen first estimate even though the residual is computed from the current nominal state. For one observation, the implemented clone-state and feature Jacobians can be written

$$H_{x,i} = \left[J_\pi \bar{R}_{CW} \quad -J_\pi \left[\bar{p}_f^C\right]_\times\right]_{\text{clone } i} \tag{36}$$

$$H_{f,i} = -J_\pi \bar{R}_{CW} \tag{37}$$

where $\bar{R}_{CW}=R(\bar{q}_{wc})^T$. Corrupted or non-finite FEJ snapshots are rejected under validation mode rather than silently replaced by the current state.

5.3 Camera cloning and covariance augmentation

Each MSCKF camera clone stores position and orientation, giving a 6-D clone error. With n clones the augmented error dimension is

$$D(n) = 15 + 6n \tag{38}$$

For a new clone whose pose is copied from the current base state, the implementation constructs a clone Jacobian Jc that selects base position and attitude error. If P is the pre-clone covariance, the augmented covariance is

$$P_{aug} = \begin{bmatrix} P & PJ_c^T \\ J_cP & J_cPJ_c^T \end{bmatrix} \tag{39}$$

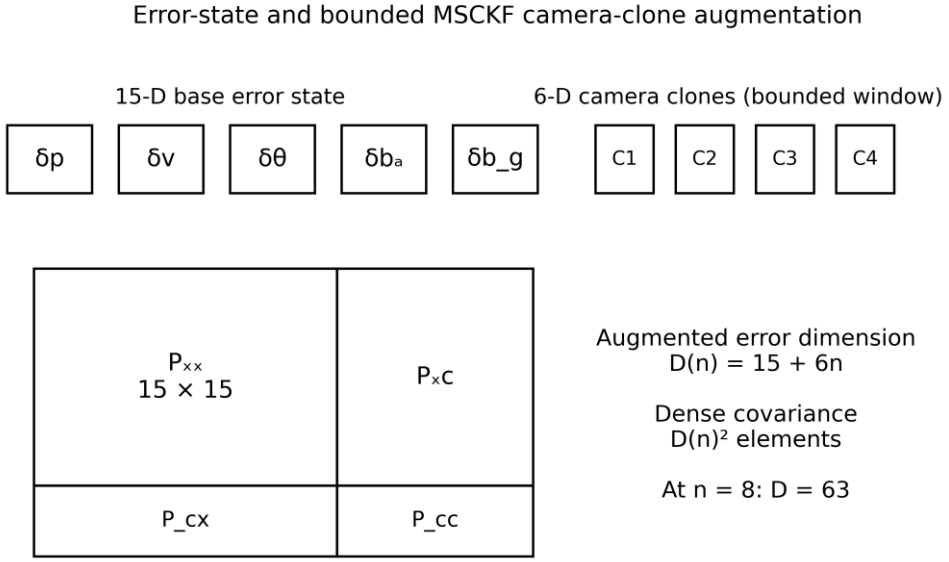


Fig. 3 Error-state and bounded MSCKF camera-clone augmentation

5.4 Multi-view triangulation

For observation i, let di be the unit bearing rotated into W and pi the camera position. The current triangulator forms the orthogonal projector $M_i=I-d_id_i^T$ and solves the least-squares intersection of rays:

$$A = \sum_i (I_3 - d_i d_i^T), \qquad b = \sum_i (I_3 - d_i d_i^T)\, p_i \tag{40}$$

$$\hat{p}_f = A^{-1}b \tag{41}$$

The implementation rejects a track when the smallest eigenvalue of A is below 10^{-9}, when fewer than two observations exist, when the maximum camera baseline is below 0.05 m, when depth lies outside 0.1–50 m, or when the maximum reprojection error exceeds 2.5 pixels. These are necessary safeguards, but a fixed baseline threshold and reprojection gate do not fully quantify conditioning.

5.5 Spectral geometry diagnostic

To explicitly distinguish image fit from depth observability, this paper introduces the dimensionless spectral geometry score

$$\gamma_A = \frac{\lambda_{min}(A)}{\lambda_{max}(A)} = \frac{1}{\kappa(A)} \tag{42}$$

A small γA indicates that the ray-intersection system is nearly singular even if reprojection residuals are small. The score is invariant to a common scaling of A and therefore provides a geometry diagnostic that complements baseline and reprojection thresholds. It is not yet a runtime gate in the analyzed repository; in this paper it is used as an analytical diagnostic and a proposed future acceptance condition.

$$\text{accept}_{geom} \Leftrightarrow b \geq b_{min} \wedge e_{reproj} \leq e_{max} \wedge \gamma_A \geq \gamma_{min} \tag{43}$$

5.6 Structureless MSCKF feature constraint

For a feature observed in m camera clones, stacking normalized residuals gives

$$r = H_x\delta x + H_f\delta p_f + n \tag{44}$$

The nuisance feature error δpf is eliminated by a left-nullspace basis A⊥ whose columns span the orthogonal complement of Hf. The implementation obtains this basis from the full U matrix of an SVD:

$$H_f = U\Sigma V^T, \qquad A_\perp = U_{[:,r_f+1:2m]} \tag{45}$$

$$A_\perp^T H_f = 0, \qquad r_o = A_\perp^T r, \qquad H_o = A_\perp^T H_x \tag{46}$$

When Hf has full rank three, the projected measurement dimension is 2m−3. The code checks both the null-space annihilation norm $\|A_\perp^T H_f\|$ and orthogonality error $\|A_\perp^T A_\perp - I\|$ against a 10^{-8} tolerance when validation is enabled.

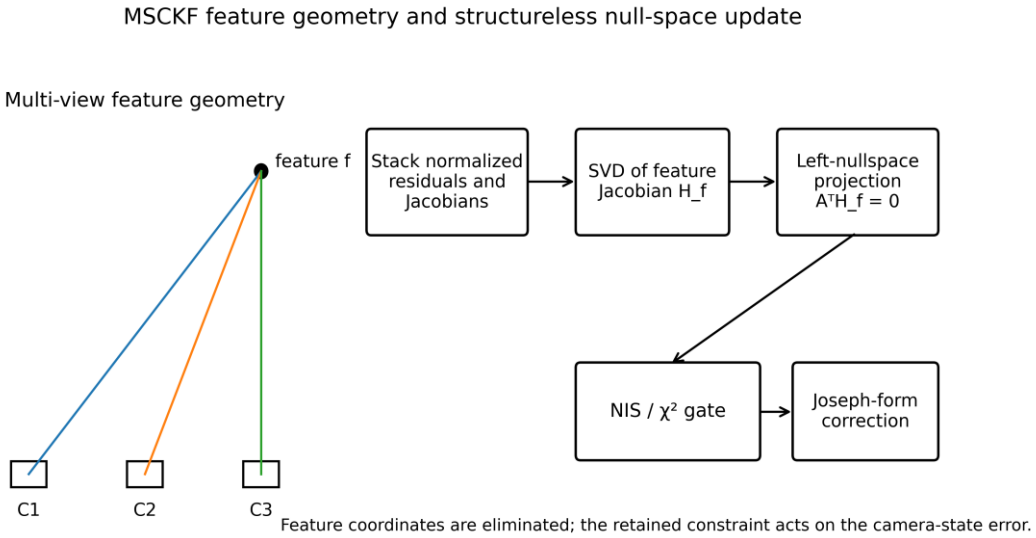


Fig. 4 MSCKF feature geometry and structureless null-space update

5.7 Innovation and chi-square gating

For the projected system, the innovation covariance and normalized innovation squared (NIS) are

$$S_o = H_o P_{aug} H_o^T + R_o \tag{47}$$

$$\text{NIS} = r_o^T S_o^{-1} r_o \tag{48}$$

A projected feature is accepted only if its residual norm is below the configured maximum and NIS does not exceed the 95% chi-square quantile for its actual projected degrees of freedom d. Thus

$$\text{NIS} \le \chi^2_{0.95,d}, \qquad d = \dim(r_o) \tag{49}$$

This degree-of-freedom-aware gate is preferable to reusing a single fixed threshold for all track lengths. Accepted feature systems are stacked deterministically in ascending track identifier order before one transactional correction is attempted.

5.8 Deterministic camera-state retirement

A bounded window requires removal of the oldest clone. Naively deleting the corresponding rows and columns can leave stale feature references or publish a partially mutated state if a later check fails. The current source therefore constructs a retirement plan, computes the retained principal covariance submatrix, checks finite values, symmetry, and positive semi-definiteness, and only then commits clone metadata and covariance together.

$$P_{ret} = P_{aug}[\mathcal{I}, \mathcal{I}], \qquad \mathcal{I} = \text{all indices except the retiring clone block} \tag{50}$$

$$\text{valid}(P_{ret}) \Leftrightarrow \text{finite}(P_{ret}) \wedge \| P_{ret} - P_{ret}^T \|_\infty \le \varepsilon_{sym} \wedge \lambda_{min}(P_{ret}) \ge -\varepsilon_{neg} \tag{51}$$

The default tolerances are εsym=10−9 and εneg=10−10. The transaction also verifies that no feature observation still references the retired clone after candidate pruning. This makes camera-state retirement a fail-closed state transition rather than a destructive in-place mutation.

6 Complementary GPS-denied localization and dynamic-environment safety

6.1 UWB range and TDOA model

Although the principal estimator analyzed here is visual-inertial, the wider platform includes UWB/TDOA localization support. For anchor ai and vehicle position p, a one-way range observation is

$$\rho_i = \| p - a_i \|_2 + \nu_i \tag{52}$$

With anchor 1 as a reference, a TDOA-derived range difference is

$$\Delta\rho_{i1} = \| p - a_i \|_2 - \| p - a_1 \|_2 + c(\delta t_i - \delta t_1) + \nu_{i1} \tag{53}$$

where c is propagation speed and δti is clock offset. The expression makes the synchronization requirement explicit: differencing removes the unknown emission time but not differential receiver clock bias. UWB/TDOA can therefore improve GPS-denied observability only when anchor geometry and timing quality are themselves validated.

6.2 LiDAR depth and local perception

The error-state implementation contains an optional scalar depth correction that is disabled by default in the validated configuration. A simple vertical-depth observation has

$$z_d = p_z + n_d, \qquad H_d = [0\ 0\ 1\ 0\ \cdots\ 0] \tag{54}$$

Keeping the correction disabled by default is scientifically important: the presence of a LiDAR parser or obstacle map does not automatically imply that a scalar LiDAR depth is a valid estimator measurement. Sensor-frame definition, calibration, scene geometry, and measurement semantics must be established before such a correction becomes authoritative.

6.3 Dynamic-obstacle closest approach

For local collision reasoning, let r be the relative position from the UAV to an obstacle and u the relative velocity. Over a horizon T, the constant-velocity time of closest approach is

$$t^* = \mathrm{clip}\left(-\frac{r^T u}{\| u \|_2^2},\ 0,\ T\right) \tag{55}$$

$$d^* = \| r + u t^* \|_2 \tag{56}$$

A safety layer can trigger avoidance when d* falls below a required separation. This model is deliberately outside the estimator correction equations. Localization produces a state and uncertainty; the safety manager consumes that state under a local-authority hierarchy. Peer consensus and backend commands are advisory for mission coordination and are not allowed to delay immediate collision avoidance or emergency behavior.

6.4 Uncertainty-aware diagnostic confidence

A useful diagnostic should not collapse geometry, residual, and covariance into a single unqualified “confidence” number. For analysis, a bounded diagnostic score can be decomposed as

$$C_{diag} = C_{cov}\, C_{innov}\, C_{geom} \tag{57}$$

$$C_{innov} = \exp\left(-\frac{\mathrm{NIS}}{2d}\right), \qquad C_{geom} = \min\left(1, \frac{\gamma_A}{\gamma_{ref}}\right) \tag{58}$$

Ccov may be any monotone function of the position covariance, for example exp(−tr(Ppp)/τp). This score is not claimed as a calibrated probability and is not implemented in the current release. Its purpose is conceptual: a small residual cannot dominate when triangulation geometry is nearly singular or covariance is large. This directly resolves the earlier residual-only interpretation problem at the mathematical level.

7 Analytical and numerical evaluation

7.1 Evaluation design

The paper-specific evaluation uses only parameters visible in the current source or reported repository artifacts. No flight trajectory, hardware timing trace, or ground-truth dataset result is invented. Four numerical analyses are performed: augmented-state scaling, chi-square measurement dimension, process-noise growth, and three-view triangulation sensitivity. A fifth calculation illustrates the variance effect of ZUPT in a decoupled velocity block. All scripts use deterministic random seeds and are described sufficiently for reproduction.

Table 2 Augmented error-state and dense covariance scaling

Camera clones n	D(n)	Covariance elements	Storage (KiB)	vs. base
0	15	225	1.76	1.00×
1	21	441	3.45	1.96×
2	27	729	5.70	3.24×
3	33	1089	8.51	4.84×
4	39	1521	11.88	6.76×
5	45	2025	15.82	9.00×
6	51	2601	20.32	11.56×
7	57	3249	25.38	14.44×
8	63	3969	31.01	17.64×

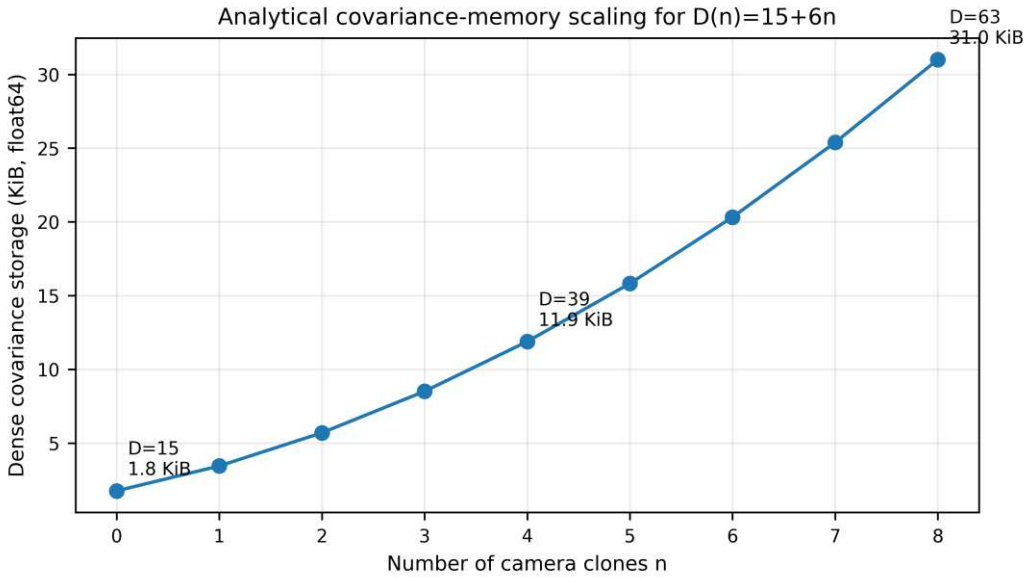


Fig. 5 Analytical covariance-memory scaling for the bounded camera-state window

At the configured maximum of eight camera clones, D=63 and a dense double-precision covariance contains 3969 elements, or 31.01 KiB. This is 17.64 times the element count of the base 15×15 covariance. The absolute memory is small on a desktop, but the $O(D^2)$ covariance operations and $O(D^3)$ worst-case dense factorizations matter on embedded processors and when multiple filters or vehicles run concurrently.

Table 3 Projected feature dimension and 95% chi-square gate for full-rank Hf

Track observations m	Projected DOF d=2m−3	$\chi^2_{0.95,d}$
3	3	7.815
4	5	11.070
5	7	14.067
6	9	16.919
7	11	19.675
8	13	22.362

The 3-observation case produces d=3 and the familiar 7.815 threshold, but longer tracks require larger thresholds. For m=8, d=13 and the correct 95% quantile is 22.362. Using a fixed 7.815 threshold for every MSCKF track would increasingly over-reject valid longer constraints.

Table 4 Process-noise growth under the implementation discretization for a stationary level orientation

Δt (s)	tr(Qd)	tr(Qd,position)	tr(Qd,velocity)
0.005	6.376×10^{-6}	1.502×10^{-10}	6.001×10^{-6}
0.010	1.276×10^{-5}	1.203×10^{-9}	1.200×10^{-5}
0.020	2.555×10^{-5}	9.652×10^{-9}	2.404×10^{-5}
0.050	6.451×10^{-5}	1.523×10^{-7}	6.060×10^{-5}
0.100	1.336×10^{-4}	1.242×10^{-6}	1.248×10^{-4}

The calculation uses the source defaults σna=0.02, σng=0.005, σnba=10−4, σnbg=10−5 and a stationary specific force aligned with gravity. The trace grows approximately linearly at small Δt through the velocity and attitude terms, while position-noise contributions scale with higher powers of Δt. This reinforces the importance of rejecting abnormally large IMU time steps rather than attempting to absorb them as ordinary propagation.

7.2 ZUPT uncertainty example

Using the default initial velocity standard deviation 0.05 m s−1 and ZUPT noise 0.01 m s−1, the decoupled scalar prior variance is 2.5×10−3 and Rv=1×10−4. Equation (32) gives

$$P_{vv}^{+} = \frac{(2.5\times10^{-3})(1\times10^{-4})}{2.6\times10^{-3}} = 9.615\times10^{-5} \tag{59}$$

The corresponding standard deviation becomes 0.00981 m s−1, a 96.15% variance reduction in the simplified uncoupled example. In the full estimator, cross-covariances transfer information to other states, so this calculation is illustrative rather than a claim about complete trajectory drift.

7.3 Triangulation geometry sensitivity

The most important paper-specific numerical experiment tests the statement that low reprojection error does not guarantee accurate triangulation. A point is placed at [0,0,5] m. Three identity-oriented cameras are placed along the x-axis with total span B. Pinhole focal length is 320 pixels and each observation is corrupted by independent zero-mean Gaussian noise with σ=1.5 pixels, matching the implementation default. For each B, 10,000 trials are triangulated using Eqs. (40)–(41). A trial is counted as accepted when the recovered depth is within 0.1–50 m and maximum reprojection error is no greater than 2.5 pixels; the baseline gate is also satisfied for B≥0.05 m. The random seed is fixed.

Table 5 Three-view synthetic triangulation sensitivity after existing depth and reprojection gates

Camera span B	Accepted	Mean 3-D error (m)	Median (m)	95th pct. (m)	Mean κ(A)
0.05	45.1%	2.12	2.24	3.19	4.45×10^{4}
0.10	68.7%	1.19	1.14	2.31	1.57×10^{4}
0.20	78.2%	0.64	0.55	1.45	3.84×10^{3}
0.50	81.2%	0.265	0.227	0.642	6.04×10^{2}
1.00	82.1%	0.136	0.114	0.326	1.52×10^{2}

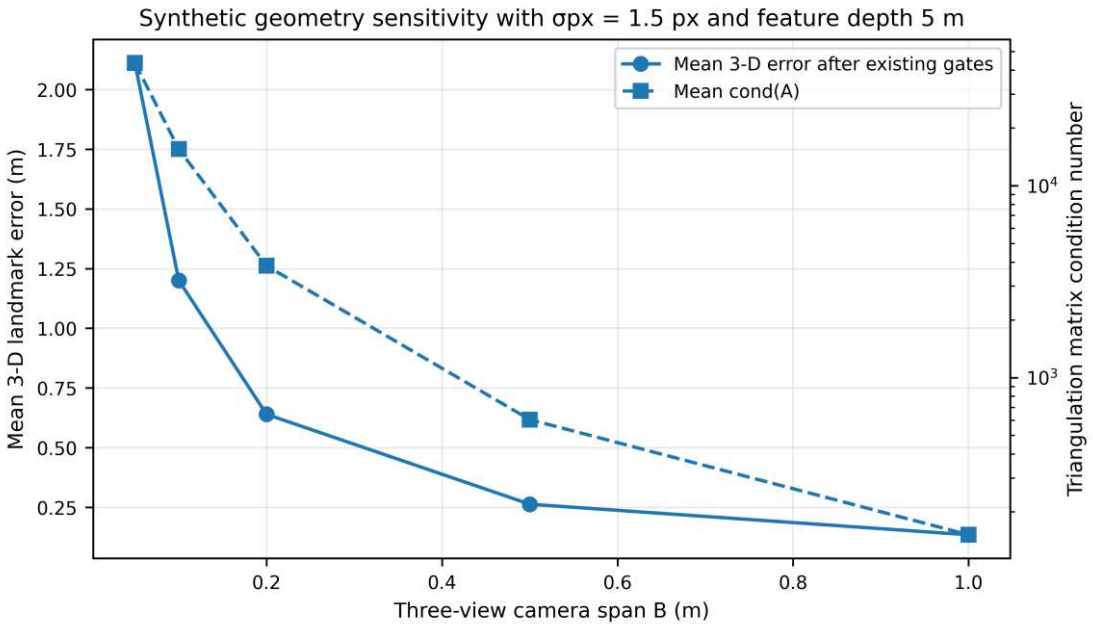


Fig. 6 Geometry sensitivity: weak camera span can satisfy image-space gates while retaining large 3-D error

The B=0.10 m result is the key counterexample to residual-only confidence. Approximately 68.7% of trials pass the existing depth/reprojection tests, yet accepted landmarks still have 1.19 m mean 3-D error and mean triangulation condition number 1.57×10^4. Increasing the span to 0.50 m lowers mean accepted error to 0.265 m and the condition number to about 604. Thus image-space residual and baseline are necessary but not sufficient diagnostics for metric depth quality. The spectral score γA provides a direct way to expose this difference.

7.4 Complexity analysis

Let n be the number of camera clones and m the number of observations in a feature track. The dominant dense covariance dimension is $D=15+6n$. Propagation of the augmented covariance using a block transition can exploit structure, but a generic dense multiplication is $O(D^3)$. The per-feature SVD acts on a $2m\times3$ matrix and is effectively $O(m)$ for bounded m because the feature dimension is fixed at three. The chi-square gate requires solving a $d\times d$ innovation system with $d\leq13$ under the current $m\leq8$ limit. Bounded clone count and bounded track length therefore convert potentially unbounded visual history into a predictable computational envelope.

Table 6 Algorithmic complexity under bounded current configuration

Operation	Dimension	Asymptotic view	Practical bound
IMU nominal propagation	15-D base	$O(1)$	Fixed-size matrix blocks
Augmented covariance propagation	$D=15+6n$	$O(D^3)$ dense upper bound	$n\leq8 \rightarrow D\leq63$
Camera-state augmentation	$D\rightarrow D+6$	$O(D^2)$	At most 8 clones retained
Feature triangulation	m rays	$O(m)$	$m\leq8$
Feature null-space SVD	$2m\times3$	$O(m)$	$2m\leq16$
Projected innovation solve	$d=2m-3$	$O(d^3)$	$d\leq13$
Retained covariance extraction	$D\times D$	$O(D^2)$	Removes one 6-D block

8 Repository validation and measured software evidence

8.1 Estimator-hardening progression

The repository develops the estimator through staged validation rather than a single unversioned code drop. Stage 15 introduced fail-closed EKF validation and deterministic replay. Stage 16 added the active/shadow estimator architecture. Stage 17 hardened the error-state mathematics. Stage 18 added stationary detection and automatic shadow-only ZUPT. Stage 19 added FEJ. Stage 20 added a bounded MSCKF camera-state window. Stage 21 added multi-view triangulation and geometric initialization. Stage 22 added the first complete feature-constraint update path and later source revisions added explicit covariance-aware camera-state retirement.

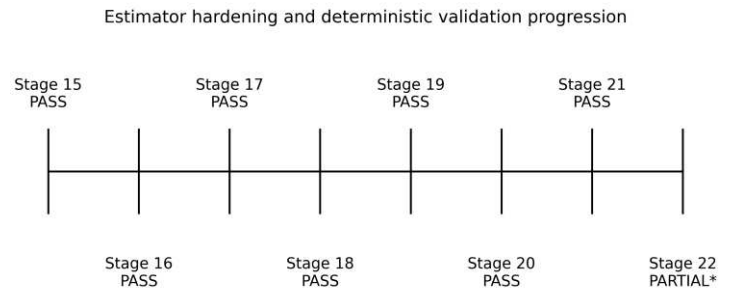


Fig. 7 Estimator hardening and conservative validation-evidence timeline

Table 7 Estimator-stage evidence and conservative interpretation

Stage	Scope	Recorded evidence	Boundary
15	Fail-closed EKF, Joseph covariance, deterministic replay	Unit/replay PASS; same-build deterministic; long-duration replay PASS	Only MSVC available in that phase; software-only
16	Active/shadow estimator coordinator	MSVC/Clang/GCC release PASS; ASan/UBSan PASS	TSan blocked by third-party OpenNI2 in that run
17	ESKF dynamics, injection, reset Jacobian	Unit/replay PASS; dedicated Linux TSan PASS; 0 project-owned races	Shadow only
18	Stationary detector + ZUPT	Replay scenarios deterministic; covariance finite; ASan/UBSan/TSan PASS	Shadow only
19	FEJ snapshots and FEJ Jacobians	Replay deterministic; finite covariance; compiler/sanitizer lanes PASS	Shadow only
20	MSCKF camera-state window	Deterministic growth/eviction; compiler/sanitizer lanes PASS	Window foundation only
21	Triangulation and landmark initialization	Straight/lateral/rotating/low-parallax/dropout replays PASS; sanitizers PASS	No physical camera dataset
22	Feature constraints, null-space, chi-square gate, later retirement source	19 unit tests PASS in checked-in Phase 22.1 report	Checked-in replay report still lists 3 failing evidence scenarios; later release metadata says Stage 22 complete

8.2 Phase 22 documentation discrepancy

A research manuscript should not hide contradictory artifacts. The current release documentation states that the software lifecycle through Stage 22 is complete and the current source contains MSCKF update, marginalization, and retirement code. However, the checked-in file docs/msckf-feature-constraint-update-shadow-validation/VALIDATION_REPORT.md is an intermediate “Phase 22.1” report dated 19 July 2026. It records 19 passing MSVC unit tests but also states that three of nine replay scenarios failed required evidence predicates and that several validation lanes were not rerun after the then-current changes. Because the repository later evolved without updating this report, this paper uses a conservative rule: source code supports implementation claims; the stale report does not support a claim of fully closed Stage 22 replay validation. This is a documentation-quality issue, not evidence that the current source necessarily fails those scenarios, but it prevents stronger retrospective claims.

This conservative treatment resolves a common reproducibility problem in evolving robotics repositories: release notes, code, and historical reports may represent different points in time. The manuscript therefore identifies the immutable commit used for source interpretation and separately labels the artifact date for each validation claim.

8.3 Software-HIL mission-state scenarios

The Phase 7 research-validation harness records five software-level mission scenarios: formation maintenance, GPS-denied navigation, communication-loss recovery, obstacle-avoidance degradation, and emergency-landing behavior. All five scenarios pass the recorded software predicates. The artifact explicitly states that it validates mission-state propagation through the Go control plane rather than actuator-level flight behavior and that no PX4, Gazebo, Ignition, or physical hardware interface was exercised.

Table 8 Recorded Phase 7 software-HIL scenario latency evidence

Scenario	Average (ms)	p95 (ms)	Recorded software observation
Formation maintenance	1.300	1.776	3-drone cluster state and leader visibility
GPS-denied navigation	7.746	20.281	vio-tdoa-fused source, confidence 0.94, four visible anchors
Communication-loss recovery	7.919	23.851	stale detection followed by readiness recovery
Obstacle-avoidance degradation	1.465	1.561	degraded localization state with local/shared obstacles
Emergency landing	13.263	24.626	emergency state propagated to control plane

Across the scenario-level summary, the recorded average is approximately 6.339 ms. The largest scenario p95 is 24.626 ms. These numbers are useful evidence about the tested software harness; they are not a control-loop deadline measurement because sensor sampling, estimator processing, network transport, actuation, and vehicle dynamics are not traced as one synchronized physical path.

8.4 Performance and latency proxies

The Phase 6 benchmark suite provides wall-clock software measurements for representative native and backend paths. The repository itself warns that the sensor timings are automated-harness proxies rather than direct hardware-rate measurements and that process launch noise contributes to the tail. The measured values are reproduced rather than reinterpreted as flight-control frequencies.

Table 9 Repository Phase 6 latency proxies

Path	Mean (ms)	p50	p95	Max	Note
IMU / EKF update path	17.390	13.276	23.970	123.001	Native harness proxy
LiDAR processing path	20.493	12.716	110.891	119.822	Native harness proxy
Sensor fusion path	34.655	19.090	121.180	122.300	Native harness proxy
POST telemetry	2.011	0.939	12.149	22.059	497.254 req/s
POST command	3.081	0.735	15.465	16.525	324.544 req/s

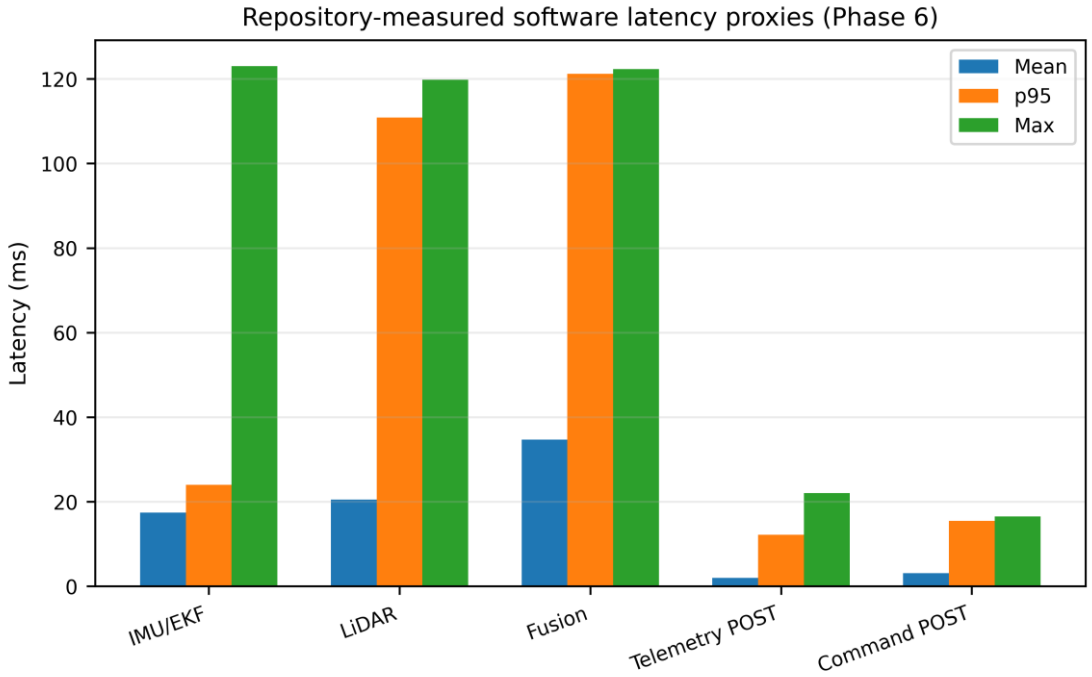


Fig. 8 Repository-measured software latency proxies from the Phase 6 benchmark suite

The native sensor and fusion paths show long tails: the IMU/EKF path has a 23.97 ms p95 but a 123.001 ms maximum, while fusion has a 121.18 ms p95. Therefore, the current evidence does not justify a hard real-time deadline claim. It does justify the narrower statement that the software paths have been instrumented, benchmarked, and subjected to stress/soak analysis in the reported workstation environment.

8.5 Answers to the research questions

RQ1 — mathematical and transactional consistency. The current error-state source can be represented by Eqs. (2)–(32). Propagation and correction are constructed as candidates and committed only after validation, which prevents partial mutation on rejected timestamps, singular innovation systems, invalid quaternions, or covariance-health failures. This supports a strong source-level claim about fail-closed semantics, not a claim of bounded physical estimation error.

RQ2 — FEJ, geometry, and MSCKF. The feature path freezes clone linearization points under FEJ, triangulates by ray least squares, eliminates feature coordinates with an SVD null space, applies residual and NIS gates, and retires camera states with covariance-health checks. The paper-specific geometry study shows that the existing baseline/reprojection gates can still accept metrically weak geometry. The spectral score γA therefore resolves a conceptual gap between image residual and 3-D observability.

RQ3 — quantitative resource and uncertainty effects. Eight clones yield a 63-D augmented error state and 31.01 KiB dense covariance. Full-rank feature tracks with 3–8 observations produce 3–13 projected degrees of freedom. The correct 95% chi-square threshold therefore varies from 7.815 to 22.362. Process-noise growth increases with Δt, and the simplified ZUPT example reduces velocity variance by 96.15%.

RQ4 — current evidence boundary. Stages 15–21 contain substantial deterministic and cross-toolchain evidence. Software-HIL scenarios and Phase 6 benchmarks provide measured software behavior. The checked-in Phase 22.1 replay report remains

partial even though later source and release metadata include Stage 22 functionality. No current artifact establishes free-flight trajectory accuracy, hardware-synchronized end-to-end latency, production-radio performance, or operational safety.

8.6 Manuscript preparation and verification

The manuscript was prepared by reconstructing equations from the cited source revision, independently recomputing analytical values, and checking literature metadata against publisher or institutional records where available. A large-language-model tool was used to assist with organization, drafting, and language refinement. The authors are responsible for verifying every technical statement, numerical result, citation, and interpretation against the cited source and for approving the final submitted version.

9 Discussion

9.1 Geometry before residual

The triangulation experiment is more than a numerical curiosity. It exposes a structural reason why a residual-only confidence score can be misleading. Reprojection residual is measured in image space. Depth uncertainty, however, depends strongly on the angular separation of viewing rays. When those rays are nearly parallel, a large change in depth produces only a small image displacement. Consequently, the optimizer or filter can fit the pixels well while the metric landmark remains uncertain. Baseline thresholds help, but a single threshold cannot account for target depth, viewing direction, and multi-view configuration. A spectral score derived from A directly measures the conditioning of the triangulation system and is therefore a more principled diagnostic.

This also clarifies why NIS and geometry should be treated as complementary rather than interchangeable. NIS tests whether the residual is statistically plausible under the predicted covariance. If the geometric model is poorly conditioned, the predicted covariance can itself be misleading or the measurement can carry little useful information. A robust acceptance chain should first validate geometry and model rank, then test innovation consistency, and only then interpret the correction magnitude.

9.2 Why Joseph form and transactional commit matter

In exact arithmetic, several algebraically equivalent covariance formulas exist. In finite-precision robotics software, they are not equally robust. Joseph form makes the positive-semidefinite structure explicit and is less vulnerable to asymmetry introduced by floating-point cancellation. The analyzed code further symmetrizes candidate covariances and checks diagonal/eigenvalue conditions before commit. These operations are sometimes dismissed as “defensive programming,” but in a safety-relevant estimator they are part of the mathematical contract: an invalid covariance means the uncertainty model is no longer trustworthy.

Transactional semantics provide a second layer of protection. If a correction fails after nominal variables have been mutated in place, rolling back all coupled state—including clones, feature metadata, timestamps, and covariance—can be error-prone. Constructing a candidate state first makes rollback trivial: the previous committed state remains unchanged. This pattern is generalizable beyond VIO to mapping, calibration, distributed fusion, and learned measurement models.

9.3 FEJ and the meaning of consistency

FEJ does not guarantee low trajectory error. Its purpose is narrower: preserve a linearization structure that respects the intended unobservable directions more faithfully than repeatedly evaluating all Jacobians at the newest estimate. This distinction is important when communicating results. A consistent filter can still be inaccurate because of poor calibration, bad correspondences, unmodeled dynamics, or weak geometry. Conversely, a temporarily accurate filter can be statistically inconsistent and later diverge. The repository’s FEJ diagnostics and corruption tests are therefore evidence about estimator construction, not proof of universal accuracy.

9.4 Bounded MSCKF state as an embedded-systems trade-off

The current maximum of eight camera clones bounds state dimension at 63. This is modest enough for many embedded processors while supporting feature tracks across several poses. The bound also makes worst-case feature residual dimension predictable. However, a small window can discard long-baseline information, whereas a large window increases covariance and update costs. The optimal window therefore depends on motion, feature persistence, sensor rate, and compute budget. A future adaptive window could use geometry score, processing load, and innovation statistics to decide when a clone is informative enough to retain.

9.5 Dynamic environments and estimator-authority separation

A dynamic environment creates two conceptually different problems. First, moving objects can corrupt visual or LiDAR measurements if treated as static landmarks. Second, even with correct localization, the autonomy layer must react to moving obstacles. The paper keeps these problems separate. Feature-tracking and gating belong to the estimator; closest-approach or velocity-obstacle reasoning belongs to safety and planning. The active/shadow architecture provides a further separation: estimator research can run on the same measurement stream without automatically changing flight-authority output.

For a UAV swarm, the same principle should extend to peer information. A remote pose, obstacle digest, or consensus hint can be useful, but it should not become a local state constraint merely because it arrives with a high sender-reported confidence. Peer freshness, trust epoch, covariance, network delay, frame alignment, and cross-correlation all matter. The current platform already prioritizes immediate local safety over peer consensus and backend mission intent; rigorous distributed state fusion is a separate research problem.

9.6 Relevance to intelligent robotics applications

The IJIRA scope spans robotics, automation, mechatronics, sensing, software, and emerging applications. The contribution here is relevant to inspection UAVs, warehouse drones, multi-robot systems, mobile manipulators, and other platforms that need dependable localization under intermittent global references. The core design principles—bounded state, geometry-aware gating, covariance health checks, deterministic replay, and fail-closed commit—are not specific to one airframe. They can be applied wherever a nonlinear estimator sits on the critical path between uncertain sensing and autonomous action.

10 Limitations and threats to validity

- No ground-truth trajectory benchmark is reported for the current shadow ESKF-MSCKF path. The paper therefore does not claim lower RMSE than OpenVINS, VINS-Mono, ORB-SLAM3, OKVIS, or any other estimator. EuRoC and TUM VI evaluation should be added before making comparative accuracy claims.
- The three-view geometry experiment is synthetic and intentionally simplified. It isolates triangulation conditioning under known camera poses and Gaussian pixel noise; it does not model feature association errors, rolling shutter, lens distortion, camera-IMU extrinsic error, time offset, rotor vibration, dynamic landmarks, or photometric effects.
- The spectral threshold γmin is proposed as a diagnostic design variable, not a tuned production constant. Selecting an operational threshold requires dataset- and motion-specific calibration and should be evaluated jointly with NIS, depth, and reprojection gates.
- Stage 22 evidence is internally inconsistent across repository documents. Current source and release metadata include the functionality, while a checked-in intermediate validation report still records partial replay closure. This paper preserves the inconsistency instead of silently selecting the more favorable source.
- Software-HIL and benchmark measurements were produced in repository-specific local environments. They do not include a hardware-synchronized camera/IMU-to-actuator trace and therefore cannot establish real-time flight deadlines.
- Physical HIL, production-radio qualification, tethered multi-UAV experiments, free flight, sensor calibration under vibration, and formal safety certification remain outside the current evidence set.
- UWB/TDOA, LiDAR, swarm networking, AI perception, digital-twin, and backend components are included only as system context. Their presence in the repository does not mean that all have been fused into the same validated estimator in the evidence analyzed here.
- The active estimator remains distinct from the shadow ESKF-MSCKF path. Automatic promotion and rollback are explicitly excluded from v2.0.0, so the paper does not claim that the new estimator currently commands flight.

11 Reproducibility, data, and next validation steps

11.1 Software and revision control

Reproduction should begin from the immutable source revision c040073969a6111f4c652d08e8de8629d70b2317 or the v2.0.0 release tag documented by the repository [36]. The exact compiler, operating system, configuration files, and validation mode should be recorded for every rerun. Repository artifacts should be treated as versioned evidence: an old report should not be assumed to describe a newer source revision.

11.2 Paper-specific numerical study

The analytical calculations in Sect. 7 can be reproduced from the equations and source defaults without proprietary data. For triangulation sensitivity, use three cameras at positions [−B/2,0,0], [0,0,0], and [B/2,0,0] m, identity orientations, a point at [0,0,5] m, fx=fy=320 pixels, principal point zero, and independent Gaussian pixel noise σ=1.5 pixels. For each B, run 10,000 trials with a fixed random seed, triangulate using Eqs. (40)–(41), and apply the 0.1–50 m depth and 2.5-pixel reprojection limits. Report acceptance rate, 3-D error, and $\kappa(A)$.

11.3 Required next experiments

- Run the current estimator on EuRoC and TUM VI with fixed configuration and report ATE/RPE, failure rate, NEES/NIS consistency, CPU time, memory, and deterministic rerun variance.
- Compare FEJ enabled/disabled, ZUPT enabled/disabled, and geometry score enabled/disabled under identical tracks to isolate each mechanism.
- Add synchronized camera-IMU hardware testing with measured time offset, extrinsic uncertainty, vibration, motion blur, and controlled texture degradation.
- Perform physical HIL with actual sensor drivers and actuator interface while tracing monotonic timestamps from sensor acquisition through estimator, safety gate, command generation, and actuator output.
- Conduct multi-drone bench/tethered tests under packet loss, latency, clock drift, and backend disconnection before making swarm-level operational claims.
- Update the Phase 22 validation report so replay, compiler, sanitizer, covariance, null-space, gating, rollback, and retirement evidence correspond to the immutable source revision cited by future papers.

12 Conclusion

This study presents a complete mathematical and systems reconstruction of a current GPS-denied UAV visual-inertial estimator rather than a narrow description of one software feature. The estimator combines a 15-D error-state inertial model, midpoint propagation, explicit process covariance, FEJ, bounded MSCKF camera cloning, multi-view triangulation, SVD-based feature elimination, chi-square gating, Joseph-form covariance correction, ZUPT, and fail-closed camera-state retirement. Its central design principle is transactional: no candidate propagation, correction, or retirement becomes public state unless the associated numerical and structural invariants pass.

The paper also resolves an important confidence problem. A low image residual is not equivalent to accurate metric localization. The three-view numerical study shows that existing depth and reprojection gates can accept landmarks with meter-scale 3-D error when camera geometry is weak. The proposed spectral score γA provides an explicit geometry diagnostic that can be combined with NIS and covariance instead of allowing residual alone to drive confidence.

Current repository evidence is substantial but bounded. Deterministic replay, compiler, sanitizer, software-HIL, and performance artifacts support claims about software correctness, repeatability, failure handling, and measured workstation behavior. They do not support claims about free-flight accuracy, hard real-time deadlines, production radio, or airworthiness. Treating those boundaries as part of the research result is a strength rather than a limitation: it makes the next experiments precise, reproducible, and falsifiable. The resulting framework is suitable as a foundation for rigorous GPS-denied UAV estimator evaluation and for broader intelligent-robotics systems in which uncertainty must remain mathematically meaningful under real software failure modes.

Statements and Declarations

Funding

The authors declare that no funds, grants, or other support were received specifically for the preparation of this manuscript.

Competing interests

The authors declare no financial or non-financial competing interests directly related to this work.

Author contributions

M.S.I.S. conceived the study, developed the research platform and estimator methodology, led the software implementation and validation work, performed the mathematical reconstruction and numerical analysis, and prepared the original manuscript. M.S.H.S. contributed to analytical review, interpretation of results, validation review, and critical revision of the manuscript. Both authors reviewed and approved the final manuscript. The contribution statement should be verified by both authors before submission because the journal submission interface will use the contribution information entered there for publication.

Data availability

The study does not use a proprietary human-participant dataset. Source-derived results use public repository artifacts, configuration files, replay reports, software-HIL outputs, and benchmark reports. The paper-specific numerical analysis is generated from the equations and parameter values reported in the manuscript and can be reproduced without restricted data.

Code availability

The software analyzed in this paper is publicly available at https://github.com/smshagor-dev/UVA-GPS-Denied-Navigation-in-Dynamic-Environments. Reproducibility should use the v2.0.0 release and/or immutable commit c040073969a6111f4c652d08e8de8629d70b2317 cited in Ref. [36].

Ethics approval

Not applicable. The reported work concerns software, mathematical analysis, synthetic numerical experiments, and public repository artifacts; no human participants or animals were involved.

Consent to participate and consent for publication

Not applicable.

Generative AI disclosure

A large-language-model tool was used during manuscript preparation to assist with organization, drafting, and language refinement. All technical content, equations, source-code interpretations, numerical calculations, references, and final wording remain subject to author verification and approval. The authors take responsibility for the final manuscript.

References

[1] Kalman, R.E.: A new approach to linear filtering and prediction problems. J. Basic Eng. 82(1), 35–45 (1960). https://doi.org/10.1115/1.3662552

[2] Mourikis, A.I., Roumeliotis, S.I.: A multi-state constraint Kalman filter for vision-aided inertial navigation. In: 2007 IEEE International Conference on Robotics and Automation, pp. 3565–3572 (2007). https://doi.org/10.1109/ROBOT.2007.364024

[3] Li, M., Mourikis, A.I.: High-precision, consistent EKF-based visual-inertial odometry. Int. J. Robot. Res. 32(6), 690–711 (2013). https://doi.org/10.1177/0278364913481251

[4] Huang, G.P., Mourikis, A.I., Roumeliotis, S.I.: Observability-based rules for designing consistent EKF SLAM estimators. Int. J. Robot. Res. 29(5), 502–528 (2010). https://doi.org/10.1177/0278364909353640

[5] Hesch, J.A., Kottas, D.G., Bowman, S.L., Roumeliotis, S.I.: Consistency analysis and improvement of vision-aided inertial navigation. IEEE Trans. Robot. 30(1), 158–176 (2014). https://doi.org/10.1109/TRO.2013.2277549

[6] Leutenegger, S., Lynen, S., Bosse, M., Siegwart, R., Furgale, P.: Keyframe-based visual–inertial odometry using nonlinear optimization. Int. J. Robot. Res. 34(3), 314–334 (2015). https://doi.org/10.1177/0278364914554813

[7] Forster, C., Carlone, L., Dellaert, F., Scaramuzza, D.: On-manifold preintegration for real-time visual–inertial odometry. IEEE Trans. Robot. 33(1), 1–21 (2017). https://doi.org/10.1109/TRO.2016.2597321

[8] Bloesch, M., Omari, S., Hutter, M., Siegwart, R.: Robust visual inertial odometry using a direct EKF-based approach. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 298–304 (2015). https://doi.org/10.1109/IROS.2015.7353389

[9] Qin, T., Li, P., Shen, S.: VINS-Mono: A robust and versatile monocular visual-inertial state estimator. IEEE Trans. Robot. 34(4), 1004–1020 (2018). https://doi.org/10.1109/TRO.2018.2853729

[10] Campos, C., Elvira, R., Rodríguez, J.J.G., Montiel, J.M.M., Tardós, J.D.: ORB-SLAM3: An accurate open-source library for visual, visual-inertial, and multimap SLAM. IEEE Trans. Robot. 37(6), 1874–1890 (2021). https://doi.org/10.1109/TRO.2021.3075644

[11] Geneva, P., Eckenhoff, K., Lee, W., Yang, Y., Huang, G.: OpenVINS: A research platform for visual-inertial estimation. In: 2020 IEEE International Conference on Robotics and Automation, pp. 4666–4672 (2020). https://doi.org/10.1109/ICRA40945.2020.9196524

[12] Burri, M., Nikolic, J., Gohl, P., Schneider, T., Rehder, J., Omari, S., Achtelik, M.W., Siegwart, R.: The EuRoC micro aerial vehicle datasets. Int. J. Robot. Res. 35(10), 1157–1163 (2016). https://doi.org/10.1177/0278364915620033

[13] Schubert, D., Goll, T., Demmel, N., Usenko, V., Stückler, J., Cremers, D.: The TUM VI benchmark for evaluating visual-inertial odometry. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1680–1687 (2018). https://doi.org/10.1109/IROS.2018.8593419

[14] Li, M., Mourikis, A.I.: Online temporal calibration for camera–IMU systems: theory and algorithms. Int. J. Robot. Res. 33(7), 947–964 (2014). https://doi.org/10.1177/0278364913515286

[15] Mirzaei, F.M., Roumeliotis, S.I.: A Kalman filter-based algorithm for IMU-camera calibration: observability analysis and performance evaluation. IEEE Trans. Robot. 24(5), 1143–1156 (2008). https://doi.org/10.1109/TRO.2008.2004486

[16] Foxlin, E.: Pedestrian tracking with shoe-mounted inertial sensors. IEEE Comput. Graph. Appl. 25(6), 38–46 (2005). https://doi.org/10.1109/MCG.2005.140

[17] Skog, I., Händel, P., Nilsson, J.-O., Rantakokko, J.: Zero-velocity detection—an algorithm evaluation. IEEE Trans. Biomed. Eng. 57(11), 2657–2666 (2010). https://doi.org/10.1109/TBME.2010.2060723

[18] Dardari, D., Conti, A., Ferner, U., Giorgetti, A., Win, M.Z.: Ranging with ultrawide bandwidth signals in multipath environments. Proc. IEEE 97(2), 404–426 (2009). https://doi.org/10.1109/JPROC.2008.2008846

[19] Chan, Y.T., Ho, K.C.: A simple and efficient estimator for hyperbolic location. IEEE Trans. Signal Process. 42(8), 1905–1915 (1994). https://doi.org/10.1109/78.301830

[20] Fiorini, P., Shiller, Z.: Motion planning in dynamic environments using velocity obstacles. Int. J. Robot. Res. 17(7), 760–772 (1998). https://doi.org/10.1177/027836499801700706

[21] van den Berg, J., Guy, S.J., Lin, M., Manocha, D.: Reciprocal n-body collision avoidance. In: Pradalier, C., Siegwart, R., Hirzinger, G. (eds.) Robotics Research, pp. 3–19. Springer, Berlin (2011). https://doi.org/10.1007/978-3-642-19457-3_1

[22] Chung, S.-J., Paranjape, A.A., Dames, P., Shen, S., Kumar, V.: A survey on aerial swarm robotics. IEEE Trans. Robot. 34(4), 837–855 (2018). https://doi.org/10.1109/TRO.2018.2857475

[23] Bekmezci, I., Sahingoz, O.K., Temel, S.: Flying ad-hoc networks (FANETs): a survey. Ad Hoc Netw. 11(3), 1254–1270 (2013). https://doi.org/10.1016/j.adhoc.2012.12.004

[24] Shakhatreh, H., Sawalmeh, A.H., Al-Fuqaha, A., Dou, Z., Almaita, E., Khalil, I., Othman, N.S., Khreishah, A., Guizani, M.: Unmanned aerial vehicles: a survey on civil applications and key research challenges. IEEE Access 7, 48572–48634 (2019). https://doi.org/10.1109/ACCESS.2019.2909530

[25] Hu, Z., Butt, M.Z., Nasir, N., et al.: Vision-based autonomous navigation for quadrotor UAVs in unknown and constrained environments. Int. J. Intell. Robot. Appl. 9, 1743–1754 (2025). https://doi.org/10.1007/s41315-025-00462-2

[26] Reyes-Munoz, J.A., Ortega, A.G., Rizia, M.M., Choudhuri, A., et al.: Autonomous aerial system for intelligent close-quarter inspection. Int. J. Intell. Robot. Appl. (2026). https://doi.org/10.1007/s41315-026-00537-8

[27] Alipourvarmazabadi, M., Ebner, M.: Robot localization: a comprehensive survey from classical methods to intelligent autonomy. Int. J. Intell. Robot. Appl. (2026). https://doi.org/10.1007/s41315-026-00528-9

[28] Yang, C.-Y., Samani, H., Tang, Z., Li, C.: Implementation of extended Kalman filter for localization of ambulance robot. Int. J. Intell. Robot. Appl. 8, 960–973 (2024). https://doi.org/10.1007/s41315-024-00352-z

[29] Maurelli, F., Krupiński, S., Xiang, X., et al.: AUV localisation: a review of passive and active techniques. Int. J. Intell. Robot. Appl. 6, 246–269 (2022). https://doi.org/10.1007/s41315-021-00215-x

[30] Barfoot, T.D.: State Estimation for Robotics. Cambridge University Press, Cambridge (2017). https://doi.org/10.1017/9781316671528

[31] Simon, D.: Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches. Wiley, Hoboken (2006). https://doi.org/10.1002/0470045345

[32] Higham, N.J.: Accuracy and Stability of Numerical Algorithms, 2nd edn. SIAM, Philadelphia (2002). https://doi.org/10.1137/1.9780898718027

[33] Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision, 2nd edn. Cambridge University Press, Cambridge (2004).

[34] Golub, G.H., Van Loan, C.F.: Matrix Computations, 4th edn. Johns Hopkins University Press, Baltimore (2013).

[35] Bar-Shalom, Y., Li, X.R., Kirubarajan, T.: Estimation with Applications to Tracking and Navigation. Wiley, New York (2001). https://doi.org/10.1002/0471221279

[36] Shagor, M.S.I.: UVA-GPS-Denied-Navigation-in-Dynamic-Environments, v2.0.0 software research release and commit c040073969a6111f4c652d08e8de8629d70b2317. GitHub (2026). Accessed 24 Aug 2026