Advances in Differential Equations and Control Processes 2026, 33(3), 4733.
doi: https://doi.org/10.59400/adecp4733

Article

A control-theoretic framework for autonomous UAV navigation in GNSS-denied environments via adaptive multi-sensor fusion

Jiafeng Li

School of Information and Communication Engineering, Dalian University of Technology, Dalian 116024, China; 18801040820@163.com

CITATION

Li J. A control-theoretic framework for autonomous UAV navigation in GNSS-denied environments via adaptive multi-sensor fusion. Advances in Differential Equations and Control Processes. 2026; 33(3): 4733.
https://doi.org/10.59400/adecp4733

ARTICLE INFO

Received: 28 July 2026
Revised: 15 August 2026
Accepted: 20 August 2026
Available online: 1 September 2026

Abstract: This paper develops a control-theoretic framework for autonomous unmanned aerial vehicle (UAV) navigation in environments where Global Navigation Satellite System (GNSS) signals are degraded or denied. The navigation problem is posed as a stochastic dynamical system: vehicle kinematics and inertial error states are modelled by a nonlinear Itô stochastic differential equation, and heterogeneous sensor observations enter through a family of nonlinear output maps. State estimation is formulated as a matrix Riccati differential equation in which measurement confidence is modulated by a bounded, continuously differentiable weighting law governed by its own first-order differential equation. We prove that, under uniform observability and boundedness of the weighting law, the estimation error is exponentially bounded in mean square, and we obtain an explicit convergence rate that degrades continuously rather than discontinuously as individual sensors lose reliability—a formal statement of graceful degradation. Learning-based feature extraction is incorporated as a bounded exogenous perturbation, and the estimator is shown to be input-to-state stable with respect to it. Trajectory generation is cast as a constrained optimal control problem solved in receding-horizon form, and the estimator–controller interconnection is shown to be stable by a small-gain argument. A schedulability analysis bounds the sensing-to-actuation latency, making the real-time claim a proved property of the task set. Experiments across five scenarios yield 0.38 m Root Mean Square Error (RMSE) indoors and 0.72 m in urban canyons, improving on visual-inertial odometry by 41.5% and conventional extended Kalman filtering by 66.1% in indoor flight.

Keywords: adaptive control; stochastic differential equations; multi-sensor fusion; optimal control; stability analysis; UAV systems

1. Introduction

Unmanned aerial vehicles operating in search-and-rescue, infrastructure inspection and autonomous logistics roles must maintain accurate self-localisation in environments where satellite positioning is unavailable [1, 2]. Global Navigation Satellite Systems degrade or fail in urban canyons, forest canopy, indoor and subterranean spaces, and under electromagnetic interference or deliberate jamming [3, 4]. From a systems perspective the difficulty is not the loss of a data source but the loss of the only absolute output map in the observation model: without it, the remaining sensors observe the state only up to an unobservable drift, and the estimation error ceases to be bounded.

This paper takes that observation as its starting point and treats GNSS-denied navigation as a problem in the stability of stochastic dynamical systems under time-varying output availability, rather than as a data-association or software-integration problem. Three modelling commitments follow.

The plant is a stochastic differential equation. Vehicle translational and rotational kinematics, together with the slowly varying biases of an inertial measurement unit, are written as a nonlinear Itô SDE driven by a vector Wiener process whose covariance encodes sensor noise densities. Working in continuous time rather than with a discretised recursion is not a cosmetic choice: it makes the estimator a matrix Riccati differential equation whose solution can be bounded analytically, and it makes the convergence rate of the fusion law a property of a differential equation rather than of an implementation.

Sensor reliability is a control input, not a tuning constant. Classical multi-sensor filters fix the measurement noise covariance offline. When a camera is blinded or a GNSS receiver is jammed, the true observation noise changes by orders of magnitude while the assumed noise does not, and the filter becomes inconsistent—its computed covariance no longer bounds its actual error. We therefore introduce a bounded weighting law that modulates measurement confidence, and we give that law its own first-order dynamics so that it is continuously differentiable, has an assignable time constant, and cannot leave a prescribed compact set. Boundedness away from zero is what preserves uniform observability and hence stability; this is the central technical point of the paper.

Learning-based perception is an exogenous disturbance to be dominated, not a component to be trusted. Deep networks for visual feature extraction and point-cloud description improve accuracy but carry no guarantees. Rather than embedding network outputs in the state and inheriting their failure modes, we admit them as pseudo-measurements with a bounded output error and establish input-to-state stability of the estimator with respect to that error. The learning module can then improve performance without being able to destabilise the system—a separation that safety-critical certification requires and that, to our knowledge, existing adaptive fusion schemes for UAV navigation do not provide.

These commitments correspond to three distinct roles for uncertainty, and it is worth separating them explicitly. Uncertainty propagation is handled by the matrix Riccati differential Equation (8): the second moment of the estimation error is itself the state of a differential equation, so the confidence attached to a navigation solution evolves continuously and is available at every instant rather than being inferred after the fact. System identification enters twice. Offline, the parameters of the environmental adaptation map (16) are identified from logged sensor error against measured scene descriptors by regularised logistic regression, a standard prediction-error identification problem [5]; the nominal noise densities are identified from static and hover data by Allan variance analysis. Online, the weighting law (12) performs a continuous, low-order identification of relative sensor reliability from the innovation sequence — the same information source exploited by classical adaptive filtering [6, 7], but used here to modulate a gain rather than to re-estimate a covariance. Feedback control closes the loop at two levels: the innovation feeds back to the gain through the weighting law, making the estimator adaptive, and the estimate feeds back to the receding-horizon controllers (26) and (27), making the vehicle autonomous. Stability of each loop is established separately in Sections 3.4 and 3.6, and of their interconnection in Theorem 2.

Adaptive covariance estimation has a long history in the filtering literature, beginning with Mehra's identification of unknown noise statistics from innovation sequences [6] and Sage and Husa's recursive noise adaptation [7], and continuing through the stochastic stability analyses of the extended Kalman filter by Boutayeb et al. [8] and Reif et al. [9]. What that literature does not address, and what GNSS-denied flight demands, is the case in which sensor reliability changes on the same timescale as the vehicle dynamics and in which the number of usable output maps changes during the mission. Conversely, the applied UAV navigation literature has produced effective adaptive weighting heuristics [10, 11] without stability guarantees. Controllability and stability analyses of nonlinear and fractional-order dynamical systems have recently been developed for industrial process models in this journal [12]; the present work addresses the analogous question for a navigation system whose observation structure varies in time. A parallel and rapidly developing line of work embeds differential-equation structure directly inside learned models: fractional sub-equation neural networks place solutions of the fractional Riccati equation in the hidden layer to obtain exact solutions of fractional partial differential equations [13], and artificial-intelligence-assisted symbolic derivation has been applied to the construction of closed-form solutions [14]. The direction here is complementary: those methods use networks to solve differential equations, whereas a differential equation—the Riccati Equation (8) with its bounded modulation—is used here to certify a system that contains networks, the learned component being admitted only as a bounded disturbance under Proposition 1. Related stability and identification analyses for engineering process models appear in previous studies [15, 16]. This paper closes the gap between the two literatures.

The contributions are as follows.

1. A continuous-time stochastic formulation of GNSS-denied UAV navigation in which the fusion law is a bounded, dynamically generated modulation of the measurement information matrix (Sections 3.1–3.3).
2. A stability theorem establishing exponential boundedness in mean square of the estimation error under uniform observability and bounded weighting, together with an explicit lower bound on the convergence rate expressed in terms of the minimum admissible weight. The bound formalises graceful degradation: as sensors fail, the guaranteed rate decreases continuously and the ultimate error bound grows continuously, with no loss of stability (Section 3.4, Theorem 1 and Corollary 1).
3. An input-to-state stability result for the learning-augmented estimator, giving an explicit linear gain from network output error to steady-state estimation error (Section 3.5, Proposition 1).
4. A constrained optimal control formulation of trajectory generation, solved in receding-horizon form with terminal ingredients guaranteeing recursive feasibility, and a small-gain argument establishing stability of the estimator–controller interconnection (Section 3.6, Theorem 2).
5. A schedulability proof bounding the sensing-to-actuation latency of the implementation, converting the real-time claim from an empirical observation into a property of the task set (Section 3.7, Proposition 2).

6. Experimental validation across five scenarios reporting, in addition to conventional accuracy metrics, control-oriented measures—settling time, overshoot, steady-state error, Lyapunov exponents and filter consistency—that permit direct comparison against the theoretical bounds (Section 5).

Notation used throughout is collected in **Appendix A**. The remainder of the paper is organised as follows. Section 2 reviews GNSS-denied navigation and adaptive filtering. Section 3 develops the model, the estimator and the stability analysis. Section 4 describes the implementation. Section 5 reports experimental validation. Section 6 concludes the article.

2. Related work

2.1. Localisation without satellite positioning

Jarraya et al. [17] survey 132 studies of GNSS-denied UAV navigation and distinguish absolute localisation, which resolves position in global Earth coordinates, from relative localisation, which estimates displacement in a local frame. Absolute methods depend on terrain or feature databases prepared in advance and on sensors capable of matching against them; relative methods require no prior map and are therefore suited to unsurveyed or changing environments, at the cost of unbounded drift. The authors conclude that hybrid schemes combining both paradigms are the most promising direction, which is the position adopted here: GNSS, when available, supplies the absolute output map that bounds the error, and the remaining sensors sustain the estimate when it is not.

Terrain-aided navigation is the oldest family of absolute methods, represented by terrain contour matching, inertial terrain-aided navigation and digital scene matching. Lee et al. [18] demonstrated a precision terrain-aided scheme combining an interferometric radar altimeter with a digital elevation model, achieving positioning accuracy better than 3.1 m at 1.5 km altitude over flat terrain. Such methods degrade where the terrain lacks distinctive elevation structure. Carroll and Canciani [19] addressed this by steering a laser rangefinder toward informative ground features, reducing position RMSE from 29.1 m to 1.2 m in mountainous terrain.

2.2. Multi-sensor fusion and adaptive weighting

Fusion of visual, inertial and ranging sensors has become the dominant approach to relative localisation. James et al. [20] combined a stereo camera, an integrated IMU and a LiDAR through appearance-based mapping to obtain 0.4 m positioning accuracy in confined indoor spaces, with loop closure providing global map consistency. Yue [10] proposed an adaptive weighted average fusing GNSS, IMU and 3D LiDAR, adjusting sensor weights according to real-time conditions so that a degraded sensor contributes less to the estimate; linear Kalman smoothing of the velocity signal further improved behaviour across sensor transitions. Gallo and Barrientos [11] analysed the growth of inertial navigation error for fixed-wing UAVs under GNSS denial and characterised the error sources that dominate at different flight durations.

These schemes establish that adaptive weighting improves accuracy in practice.

None provides conditions under which the resulting closed loop is stable, which is the question addressed in Section 3.4.

2.3. Visual-inertial and learning-based methods

Visual-inertial odometry offers complementary sensing at low cost and has become competitive with LiDAR-based systems. Chen et al. [21] review filtering-based and optimisation-based formulations of visual-inertial SLAM and characterise the trade-off between consistency and computational cost. Feng et al. [22] extended monocular visual-inertial odometry with point and line features and adaptive line extraction, improving localisation accuracy by 33.8% relative to a VINS-Mono baseline. Campos et al. [23] combine visual, visual-inertial and multi-map operation in ORB-SLAM3, which together with the direct LiDAR-inertial odometry of Xu et al. [24] represents the current state of the art against which new systems should be measured.

Large-scale UAV vision benchmarks such as VisDrone [25] have accelerated this work by supplying annotated aerial imagery at scale. Learning-based perception has been introduced into this pipeline in several forms. Yu et al. [26] developed a learned-feature visual odometry system for UAV pose estimation in challenging indoor environments, emphasising interpretability of the learned representation in safety-critical settings. Cao et al. [27] combined template matching with convolutional networks for GPS-independent UAV localisation, improving robustness under adverse visual conditions. He et al. [28] and Liu et al. [29] applied reinforcement learning to obstacle avoidance and trajectory design respectively. In all of these, the learned component is embedded in the estimation or control loop without an accompanying guarantee that its failure cannot destabilise the system; Section 3.5 supplies such a guarantee.

2.4. Observers respecting the geometry of motion

A parallel line of work obtains stronger convergence results by respecting the manifold structure of attitude and pose. Mahony et al. [30] developed nonlinear complementary filters on the special orthogonal group with almost-global convergence; Bonnabel et al. [31] generalised this through symmetry-preserving observers; and Barrau and Bonnabel [32] established the invariant extended Kalman filter as a stable observer whose error dynamics are state-independent. Lee et al. [33] developed geometric tracking control for quadrotors directly on the special Euclidean group. These formulations currently accommodate neither time-varying measurement weighting nor the intermittent availability of an absolute output map, and are discussed in Section 6 as the natural route to a global version of Theorem 1.

3. Control-theoretic formulation

Figure 1 shows the closed-loop structure developed in this section and locates each governing equation within it. The plant (3) and observation model (4) are set out in Section 3.1; the Riccati Equation (8) and the estimator (9) in Section 3.2; the adaptation path from the innovation through the weighting laws (12)–(16) to the gain

(11) in Section 3.3; the stability analysis in Section 3.4; the treatment of the learning module as a bounded disturbance in Section 3.5; the outer control loops (26) and (27) in Section 3.6; and the timing guarantee in Section 3.7.

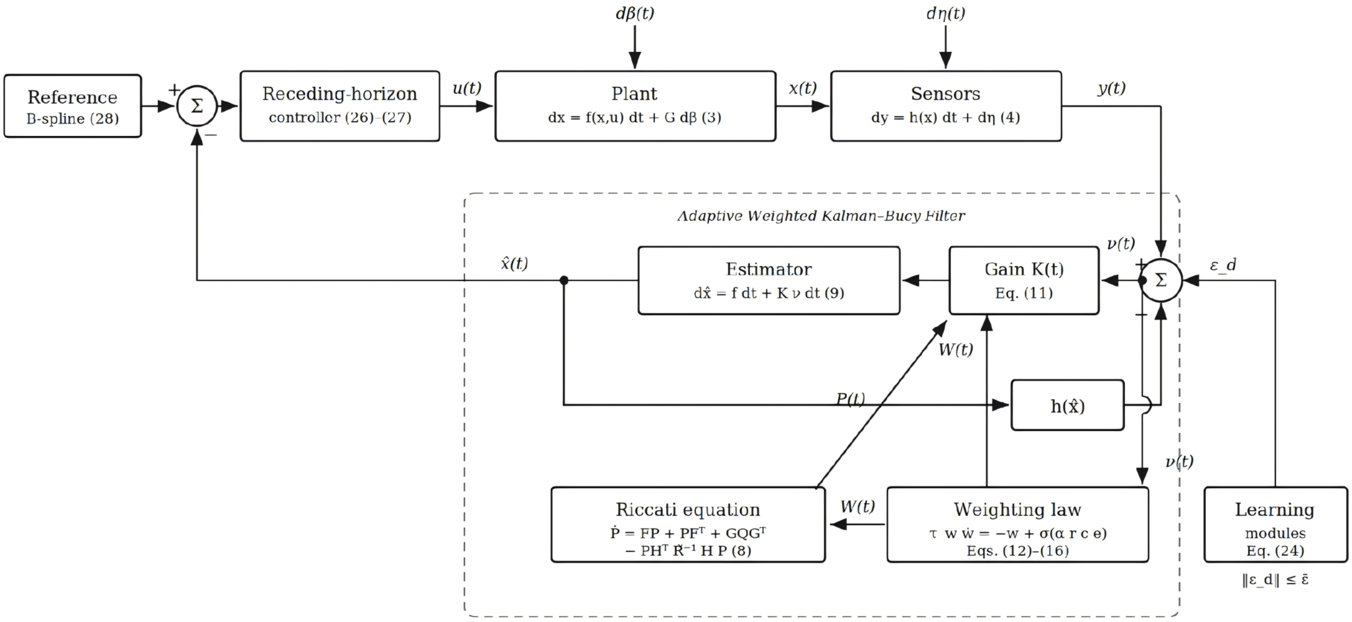

Figure 1. Closed-loop block diagram. The adaptation path—innovation to weighting laws (12)–(16) to gain (11) and Riccati Equation (8)—is the contribution of this work. The learning module enters as an additive bounded disturbance ε_{d} rather than as an augmented state.

3.1. Continuous-time stochastic model of vehicle and sensors

Let the navigation state be

$$x(t) = \left[p(t)^\top,\ v(t)^\top,\ q(t)^\top,\ b_a(t)^\top,\ b_g(t)^\top \right]^\top \tag{1}$$

with position $p \in \mathbb{R}^3$ and velocity $v \in \mathbb{R}^3$ in the local navigation frame, attitude quaternion $q \in \mathbb{S}^3$, and accelerometer and gyroscope biases $b_a, b_g \in \mathbb{R}^3$. The vehicle kinematics form the ordinary differential system

$$\dot{p} = v, \qquad \dot{v} = R(q)\big(a_m - b_a\big) + g, \qquad \dot{q} = \tfrac{1}{2}\, q \otimes \begin{bmatrix} 0 \\ \omega_m - b_g \end{bmatrix} \tag{2}$$

where $R(q) \in SO(3)$ is the rotation matrix associated with q, $\otimes$ denotes quaternion product, a_m and ω_m are inertial measurements, and g is the gravity vector. Biases are modelled as random walks.

Admitting the stochastic terms, the plant is the Itô stochastic differential equation

$$dx(t) = f\big(x(t), u(t)\big)\, dt + G\big(x(t)\big)\, d\beta(t), \qquad x(t_0) = x_0 \tag{3}$$

with β a vector Wiener process satisfying $\mathbb{E}\left[d\beta\, d\beta^\top\right] = Q_c\, dt$, $Q_c \succ 0$, and G the noise input matrix mapping accelerometer, gyroscope and bias-driving noise into the state. Existence and uniqueness of a strong solution follow from local Lipschitz continuity of f and linear growth of G [34]. The formulation of nonlinear filtering for systems described by stochastic differential equations follows [35].

The sensor suite provides m output maps. For sensor $i \in \{1, \dots, m\}$,

$$dy_i(t) = h_i\big(x(t)\big)\,dt + d\eta_i(t), \qquad \mathbb{E}\big[d\eta_i\,d\eta_i^\top\big] = R_i\,dt \tag{4}$$

with $h_{\text{gnss}}(x) = p$; $h_{\text{vis}}(x) = \pi\big(R(q)^\top(\ell - p)\big)$ for a landmark ℓ and camera projection π; and h_{lid} the point-to-plane residual map of the registered point cloud. Write $h = [h_1^\top, \ldots, h_m^\top]^\top$ and $R = \text{blkdiag}(R_1, \ldots, R_m)$.

The essential structural feature of GNSS denial is that the availability of h_{gnss} is a time-varying binary signal. We encode this, together with continuous degradation of the remaining sensors, in the weighting law of Section 3.3.

3.2. Error dynamics and the Riccati differential equation

Because q evolves on $\mathbb{S}^3$, a linear filter applied directly to x does not preserve the unit-norm constraint. We adopt the error-state formulation [36]. Let $\hat{x}$ denote the estimate propagated through Equation (2) and define the fifteen-dimensional error state

$$\delta x = \big[\,\delta p^\top,\ \delta v^\top,\ \delta\theta^\top,\ \delta b_a^\top,\ \delta b_g^\top\,\big]^\top, \qquad q = \hat{q} \otimes \delta q, \quad \delta q \approx \big[1,\ \tfrac{1}{2}\delta\theta^\top\big]^\top \tag{5}$$

A derivative-free alternative is the unscented transform [37]; the error-state formulation is preferred here because it admits the covariance bound of Lemma 1. Observers formulated intrinsically on $SO(3)$ or $SE(3)$ [30–32] obtain stronger convergence results but do not presently accommodate time-varying measurement weighting.

To first order the error obeys the linear stochastic differential equation

$$d\,\delta x(t) = F(t)\,\delta x(t)\,dt + \varphi\big(x, \hat{x}\big)\,dt + G\,d\beta(t) \tag{6}$$

with system matrix

$$F(t) = \begin{bmatrix} 0 & I & 0 & 0 & 0 \\ 0 & 0 & -R(\hat{q})\lfloor a_m - \hat{b}_a \rfloor_\times & -R(\hat{q}) & 0 \\ 0 & 0 & -\lfloor \omega_m - \hat{b}_g \rfloor_\times & 0 & -I \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{7}$$

where $\lfloor \cdot \rfloor_\times$ is the skew-symmetric cross-product matrix, and φ collects the linearisation residual.

The estimator is the continuous-time counterpart of the classical linear filter [38], in the form derived by Kalman and Bucy [39], from which it takes its name. The error covariance $P(t) = \mathbb{E}\big[\delta x\,\delta x^\top\big]$ satisfies the matrix Riccati differential equation

$$\begin{aligned} \dot{P}(t) &= F(t)P(t) + P(t)F(t)^\top + GQ_cG^\top - P(t)H(t)^\top \tilde{R}(t)^{-1} H(t)P(t), \\ P(t_0) &= P_0 \succ 0 \end{aligned} \tag{8}$$

with observation Jacobian $H(t) = \partial h / \partial \delta x\big|_{\hat{x}(t)}$, and the state estimate correction

$$d\hat{x}(t) = f(\hat{x}, u)\,dt + K(t)\big[\,dy(t) - h(\hat{x})\,dt\,\big], \qquad K(t) = P(t)H(t)^\top \tilde{R}(t)^{-1} \tag{9}$$

Equation (8) is the mathematical core of the paper and is the object to which the stability analysis of Section 3.4 applies.

3.3. The adaptive weighting law

Let $W(t) = \operatorname{diag}\big(w_1(t), \ldots, w_m(t)\big)$ collect the sensor weights. The weighted measurement information matrix is

$$\tilde{R}(t)^{-1} = W(t)\, R^{-1}\, W(t) \tag{10}$$

so that the effective noise covariance of sensor i is R_i/w_i^2 and reliability enters monotonically: a larger weight corresponds to a smaller assumed noise and hence to greater influence on the estimate. Substituting Equation (10) into Equation (9),

$$K(t) = P(t)H(t)^\top W(t)R^{-1}W(t) \tag{11}$$

Rather than assigning w_i instantaneously from the current innovation, we generate it as the output of a first-order differential equation. This makes w_i continuously differentiable, gives it an assignable time constant, and—through the saturation σ—confines it to a compact set:

$$\tau_w\, \dot{w}_i(t) = -\, w_i(t) + \sigma\Big(\alpha_i\, r_i(t)\, c_i(t)\, e_i(t)\Big), \qquad w_i(t_0) = \alpha_i \tag{12}$$

$$\sigma(s) = \underline{w} + \big(\bar{w} - \underline{w}\big)\, \mathrm{sat}_{[0,1]}(s), \qquad 0 < \underline{w} \le w_i(t) \le \bar{w} \le 1 \tag{13}$$

The three factors are defined as follows. The innovation-based reliability uses the normalised innovation squared, with $S_i = H_i P H_i^\top + R_i/w_i^2$ the innovation covariance:

$$r_i(t) = \exp\Big(-\tfrac{1}{2}\, \nu_i(t)^\top S_i(t)^{-1} \nu_i(t)\Big), \qquad \nu_i(t) = \dot{y}_i(t) - h_i\big(\hat{x}(t)\big) \tag{14}$$

The temporal consistency factor penalises rapid innovation change, computed from a low-pass filtered innovation $\bar{\nu}_i$ satisfying $\lambda_\nu \dot{\bar{\nu}}_i = -\bar{\nu}_i + \nu_i$:

$$c_i(t) = \exp\left(-\,\frac{\|\dot{\bar{\nu}}_i(t)\|}{\gamma_i}\right) \in (0,1] \tag{15}$$

The environmental adaptation factor is a bounded logistic map of measurable scene descriptors:

$$e_i(t) = \Big(1 + \exp\big(-\ \kappa_i^\top \xi(t) + \theta_i\big)\Big)^{-1} \in (0,1) \tag{16}$$

where $\xi(t) \in \mathbb{R}^{n_\xi}$ collects onboard-measurable quantities—mean image intensity and gradient energy for the camera, return-intensity variance and valid-return fraction for the LiDAR, carrier-to-noise density and dilution of precision for the GNSS receiver—and κ_i, θ_i are calibrated offline by regularised logistic regression against measured sensor error [5].

Remark 1. *The saturation in Equation (13) guarantees $w_i \ge \underline{w} > 0$, which is precisely the hypothesis required by Theorem 1, and bounds Equation (15) in $(0,1]$ so that the effective noise covariance $\tilde{R}$ is positive definite at all times.*

Remark 2. *Equation (14) is monotonically related to the differential entropy of the Gaussian innovation: for $\nu_i \sim \mathcal{N}(0, S_i)$ that entropy is $\frac{1}{2}\log\big((2\pi e)^{n_i}\det S_i\big)$, so weighting by r_i downweights sensors whose innovations are improbable under their own predicted uncertainty. The weighting law is in this sense an information-theoretic* *criterion applied to a control gain.*

3.4. Stability analysis

The following assumptions are standard in the stochastic stability literature [8,9], and stability notions are used in the sense of Khalil [40]. They are verified experimentally in Section 5.11.

Assumption 1 (Uniform observability)**.** *There exist $\alpha_1, \alpha_2 > 0$ and $T > 0$ such that for all $t \geq t_0$ the observability Gramian satisfies*

$$\alpha_1 I \preceq \mathcal{O}(t, t+T) = \int_t^{t+T} \Phi(s,t)^\top H(s)^\top R^{-1} H(s)\,\Phi(s,t)\,ds \preceq \alpha_2 I \tag{17}$$

with Φ the state transition matrix of F.

Assumption 2 (Bounded weighting)**.** $0 < \underline{w} \leq w_i(t) \leq \bar{w} \leq 1$ *for all i and all t, which Equation (13) guarantees.*

Assumption 3 (Bounded linearisation residual)**.** *There exist $\kappa_\varphi > 0$ and $\epsilon > 0$ such that $\|\varphi(x, \hat{x})\| \leq \kappa_\varphi \|\delta x\|^2$ whenever $\|\delta x\| \leq \epsilon$.*

Assumption 4 (Bounded noise)**.** $\mathrm{tr}(Q_c) \leq \bar{q} < \infty$ *and* $R \succeq \underline{r}\, I$ *with* $\underline{r} > 0$.

Lemma 1 (uniform boundedness of the Riccati solution)**.** *Under Assumptions 1, 2, and 4, the solution $P(t)$ of Equation (8) satisfies $\underline{p}\, I \preceq P(t) \preceq \bar{p}\, I$ for all $t \geq t_0 + T$, with*

$$\underline{p} \geq \frac{\lambda_{\min}\big(G Q_c G^\top\big)}{2\|F\|_\infty + \bar{w}^2\, \alpha_2/\underline{r}}, \qquad \bar{p} \leq \frac{1}{\underline{w}^2\, \alpha_1} + \bar{q}\, T\, e^{2\|F\|_\infty T} \tag{18}$$

Proof of Lemma 1. The weighted Gramian obeys $\underline{w}^2\, \mathcal{O}(t, t+T) \preceq \int_t^{t+T} \Phi^\top H^\top \tilde{R}^{-1} H \Phi\, ds \preceq \bar{w}^2\, \mathcal{O}(t, t+T)$ by Assumption 2 and Equation (10), so Assumption 1 implies uniform observability of the weighted pair with constants $\underline{w}^2 \alpha_1$ and $\bar{w}^2 \alpha_2$. The bounds then follow from the standard boundedness result for the Riccati differential equation under uniform observability and uniform controllability of the noise input [41]. □

Lemma 1 is where the requirement $\underline{w} > 0$ does its work. If the weighting law were permitted to drive any weight to zero, the corresponding output map would leave the Gramian, uniform observability could fail, and $\bar{p}$ in Equation (18) would diverge.

Theorem 1 (exponential mean-square boundedness)**.** *Consider the plant (3), the estimators (8) and (9) with weighting laws (10)–(16), and suppose Assumptions 1–4 hold and $\|\delta x(t_0)\| \leq \epsilon$. Then the estimation error is exponentially bounded in mean square:*

$$\mathbb{E}\big[\|\delta x(t)\|^2\big] \leq \frac{\bar{p}}{\underline{p}}\, \mathbb{E}\big[\|\delta x(t_0)\|^2\big]\, e^{-\alpha(t-t_0)} + \frac{\beta}{\alpha} \cdot \frac{\bar{p}}{\underline{p}} \tag{19}$$

with rate and offset

$$\alpha = \frac{\underline{w}^2\, \alpha_1}{\bar{p}} - 2\kappa_\varphi\, \epsilon\, \frac{\bar{p}}{\underline{p}}, \qquad \beta = \frac{\mathrm{tr}\big(G Q_c G^\top\big)}{\underline{p}} \tag{20}$$

provided $\alpha > 0$.

Proof of Theorem 1. Take the stochastic Lyapunov function $V(\delta x, t) = \delta x^\top P(t)^{-1}\delta x$, which by Lemma 1 satisfies $\bar{p}^{-1}\|\delta x\|^2 \leq V \leq \underline{p}^{-1}\|\delta x\|^2$. Differentiating P^{-1} using Equation (8) gives

$$\frac{d}{dt}P^{-1} = -P^{-1}\dot{P}P^{-1} = -P^{-1}F - F^\top P^{-1} - P^{-1}GQ_cG^\top P^{-1} + H^\top \tilde{R}^{-1}H \quad (21)$$

Applying the Itô generator $\mathcal{L}$ to V along Equation (6) and substituting Equation (21), the terms in F cancel identically and

$$\mathcal{L}V = -\delta x^\top H^\top \tilde{R}^{-1}H\,\delta x - \delta x^\top P^{-1}GQ_cG^\top P^{-1}\delta x + 2\,\delta x^\top P^{-1}\varphi + \mathrm{tr}\big(G^\top P^{-1}GQ_c\big) \quad (22)$$

The second term is non-positive and is discarded. By Assumption 2 and uniform observability the first term is bounded above by $-(\underline{w}^2\alpha_1/\bar{p})\,V$. By Assumption 3 and the Cauchy–Schwarz inequality, $2\delta x^\top P^{-1}\varphi \leq 2\kappa_\varphi \underline{p}^{-1}\|\delta x\|^3 \leq 2\kappa_\varphi\epsilon(\bar{p}/\underline{p})V$ on $\|\delta x\| \leq \epsilon$. The final term is bounded by β from Assumption 4. Collecting terms yields $\mathcal{L}V \leq -\alpha V + \beta$ with α, β as in Equation (20). The comparison lemma for stochastic differential equations [42] gives $\mathbb{E}[V(t)] \leq \mathbb{E}[V(t_0)]e^{-\alpha(t-t_0)} + \beta/\alpha$, and Equation (19) follows from the two-sided bound on V. $\square$

Corollary 1 (graceful degradation)**.** *The guaranteed convergence rate and asymptotic error bound satisfy*

$$\alpha \propto \underline{w}^2, \qquad \limsup_{t\to\infty} \mathbb{E}\big[\|\delta x(t)\|^2\big] \leq \frac{\beta}{\alpha}\cdot\frac{\bar{p}}{\underline{p}} = O\big(\underline{w}^{-2}\big) \quad (23)$$

Consequently, as sensors degrade and $\underline{w}$ *decreases toward its floor, the convergence rate decreases and the ultimate error bound grows, both continuously in* $\underline{w}$*. Stability is retained for every* $\underline{w} > 0$ *and is lost only in the limit* $\underline{w} \to 0$*, which the saturation (13) forbids.*

Corollary 1 is the formal content of graceful degradation: the system cannot pass discontinuously from bounded error to divergence as sensors fail.

Remark 3. *Under Equation (12) the convergence time of the weighting law is a design parameter rather than an observation. In the unsaturated regime the weight response to a step change in sensor reliability settles to within 5% in* $3\tau_w$*. When the saturation (13) is active—which is precisely the case during sensor denial, since the weight is driven onto its floor—the response is slower than the linear prediction, because the argument of* σ *must first leave the saturated region before the first-order dynamics resume. The measured settling time is reported against both figures in Section 5.12.*

3.5. The learning module as a bounded perturbation

Let the learning-based perception modules produce a pseudo-measurement

$$z_{\mathrm{d}}(t) = h_{\mathrm{d}}\big(x(t)\big) + \varepsilon_{\mathrm{d}}(t), \qquad \|\varepsilon_{\mathrm{d}}(t)\| \leq \bar{\varepsilon} \quad \text{for all } t \quad (24)$$

where h_{d} maps the state to the geometric quantity the network regresses—relative pose from the visual network, point-cloud registration residual from the geometric network—and ε_d is the network output error. The bound $\bar{\varepsilon}$ is not assumed but measured, as the empirical supremum of the network residual on a held-out validation set, and is reported in Section 5.7.

Proposition 1 (Input-to-state stability with respect to network error)**.** *Under the hypotheses of Theorem 1 with* h_d *adjoined to the observation vector and weight* $w_\text{d} \in [\underline{w}, \bar{w}]$*, the estimator is input-to-state stable with respect to* ε_d*:*

$$\|\delta x(t)\| \;\le\; \varrho\big(\|\delta x(t_0)\|,\, t-t_0\big) \;+\; \gamma_\text{d} \sup_{s\le t}\|\varepsilon_\text{d}(s)\|, \qquad \gamma_\text{d} \;=\; \frac{\bar{p}\,\bar{w}^2\,\|H_\text{d}\|}{\underline{r}\,\alpha} \tag{25}$$

with ϱ *a class-*$\mathcal{KL}$ *function.*

Proof of Proposition 1. Repeat the derivation of Equation (22) with the additional measurement contributing $2\delta x^\top P^{-1} K_\text{d}\varepsilon_\text{d}$ to $\mathcal{L}V$. Bounding $\|K_\text{d}\| \le \bar{p}\bar{w}^2\|H_\text{d}\|/\underline{r}$ from Equation (11), Lemma 1 and Assumption 4, and applying Young's inequality to split the cross term, yields $\mathcal{L}V \le -\frac{\alpha}{2}V + \beta + \gamma_\text{d}^2\|\varepsilon_\text{d}\|^2$. The estimate (25) follows from the standard ISS-Lyapunov characterisation [43]. $\square$

The practical consequence is the design principle stated in the introduction: the network cannot destabilise the estimator so long as its output error remains bounded, and the weighting law (12) supplies the enforcement mechanism, since a network whose innovations become inconsistent with S_d has $r_\text{d} \to 0$ and is driven to $\underline{w}$.

3.6. Trajectory generation as a constrained optimal control problem

Given the estimated state, trajectory generation is posed in Bolza form over horizon T_h:

$$\min_{u(\cdot)} \quad J = \Phi\big(x(t+T_h)\big) + \int_t^{t+T_h} \Big[\|x(s) - x_\text{ref}(s)\|_Q^2 + \|u(s)\|_{R_u}^2\Big]\, ds \tag{26}$$

subject to

$$\dot{x} = f(x,u), \quad g\big(x(s),u(s)\big) \le 0, \quad x(t) = \hat{x}(t), \quad x(t+T_h) \in X_f \tag{27}$$

where g collects actuator saturation, velocity and acceleration envelopes, and obstacle avoidance constraints, and x_ref is the reference trajectory generated by A* search over the occupancy grid and smoothed by a B-spline of order k,

$$C(s) = \sum_{i=0}^{n} P_i\, N_{i,k}(s), \qquad N_{i,k} \text{ the B-spline basis functions} \tag{28}$$

The problems (26) and (27) are solved in receding-horizon form. Choosing the terminal cost $\Phi(x) = \|x - x_\text{ref}\|_{P_f}^2$ with P_f the solution of the algebraic Riccati equation for the linearisation about the reference, and the terminal set X_f a sublevel set of Φ rendered invariant by the associated local feedback, guarantees recursive feasibility and asymptotic stability of the closed loop by the terminal-ingredients argument of Mayne et al. [44].

Theorem 2 (Stability of the estimator–controller interconnection)**.** *Let the receding-horizon controller of Equations (26) and (27) be ISS with respect to state estimation error with gain γ_c, and let the estimator satisfy Equation (25) with gain γ_{d}. If the small-gain condition $\gamma_c\gamma_{\mathrm{d}} < 1$ holds, the interconnected estimation-and-control system is ISS with respect to $(\varepsilon_{\mathrm{d}}, \beta)$, and the tracking error is ultimately bounded.*

Proof of Theorem 2. Immediate from the nonlinear small-gain theorem [45] applied to the cascade of the ISS estimator of Proposition 1 and the ISS receding-horizon controller. □

Remark 4 (Discrete-time implementation)**.** *The flight computer integrates Equation (2) at 400 Hz by fourth-order Runge–Kutta and propagates Equation (8) at 100 Hz using the van Loan discretisation of $(F, GQ_cG^\top)$. The discrete covariance update is implemented in the Joseph stabilised form [46],*

$$P_k^+ = \big(I - K_kH_k\big)P_k^-\big(I - K_kH_k\big)^\top + K_k\tilde{R}_kK_k^\top \tag{29}$$

which preserves symmetry and positive semi-definiteness in finite precision under a time-varying gain.

3.7. Real-time schedulability

The estimator and controller are realised as periodic tasks τ_i with worst-case execution time C_i, period T_i and deadline $D_i = T_i$. Let $\mathrm{hp}(i)$ denote the tasks of higher priority than τ_i. Under fixed-priority preemptive scheduling the worst-case response time R_i is the least fixed point of the recurrence [47]

$$R_i = C_i + \sum_{j\in\mathrm{hp}(i)} \left\lceil \frac{R_i}{T_j} \right\rceil C_j \tag{30}$$

and the task set is schedulable if and only if $R_i \le D_i$ for every i. The sufficient utilisation bound of Liu and Layland [48], $U \le n(2^{1/n} - 1)$, serves as a screening test.

Proposition 2 (real-time guarantee)**.** *Let the navigation software comprise the three periodic tasks specified in Section 4.5. Then the task set is not schedulable on a single processor core, but is schedulable under the partition in which learning inference is assigned to a separate core with GPU offload, with worst-case response times $R_{\text{filt}} = 7.5$ ms and $R_{\text{plan}} = 9.6$ ms, and the sensing-to-actuation latency is bounded by $L \le R_{\text{filt}} + R_{\text{plan}} = 17.1$ ms.*

Proof of Proposition 2. Total utilisation is $U = 1.011 > 1$, which violates the necessary condition for single-core feasibility; the set is therefore not schedulable on one core regardless of scheduling policy. Under the stated partition, core A carries $U_A = 0.855$. This exceeds the Liu–Layland bound $2(2^{1/2} - 1) = 0.828$ for two tasks, so the screening test is inconclusive and the exact analysis (30) is required. For the filter task, $R_{\text{filt}} = C_{\text{filt}} = 7.5 \le 10$. For the trajectory task, iterating Equation (30) from $R^{(0)} = 2.1$ gives $R^{(1)} = 2.1 + \lceil 2.1/10 \rceil \cdot 7.5 = 9.6$ and

$R^{(2)} = 2.1 + \lceil 9.6/10 \rceil \cdot 7.5 = 9.6$, a fixed point, so $R_{\text{plan}} = 9.6 \leq 20$. Core B carries a single task with $R = 5.2 \leq 33.3$. The end-to-end bound follows by summing response times along the sensing-to-actuation chain. □

The implementation uses fixed-priority preemptive scheduling under the `SCHED_FIFO` policy of the real-time kernel, with priorities assigned rate-monotonically and priority inheritance enabled on the two mutexes guarding the feature cache; the blocking term this introduces is bounded by the longest critical section, measured at 0.14 ms, and is included in C_i. The core assignment follows from the analysis rather than preceding it: the filter and trajectory tasks are placed together because their combined utilisation, 0.855, is the largest that admits a feasible fixed-point in Equation (30), while learning inference is placed apart because it is the only task whose execution is dominated by GPU kernels and therefore the only one whose CPU demand can be displaced. Memory-bandwidth contention, last-level cache interference and GPU–CPU transfer latency are not modelled by Equation (30); they are absorbed into C_i by measuring each task while the others run at full rate, which is an empirical rather than analytical treatment and is stated as such.

Because learning inference updates a shared feature cache read asynchronously by the filter (Section 4.5), it does not lie on the critical path, and its execution time does not enter the latency bound.

4. Implementation

4.1. Software architecture

The navigation system adopts a modular, layered architecture optimised for real-time embedded deployment, shown in **Figure 2**. The layering separates data acquisition, processing and decision-making, with inter-module communication designed to minimise latency while preserving data integrity through synchronised time-stamping and buffering.

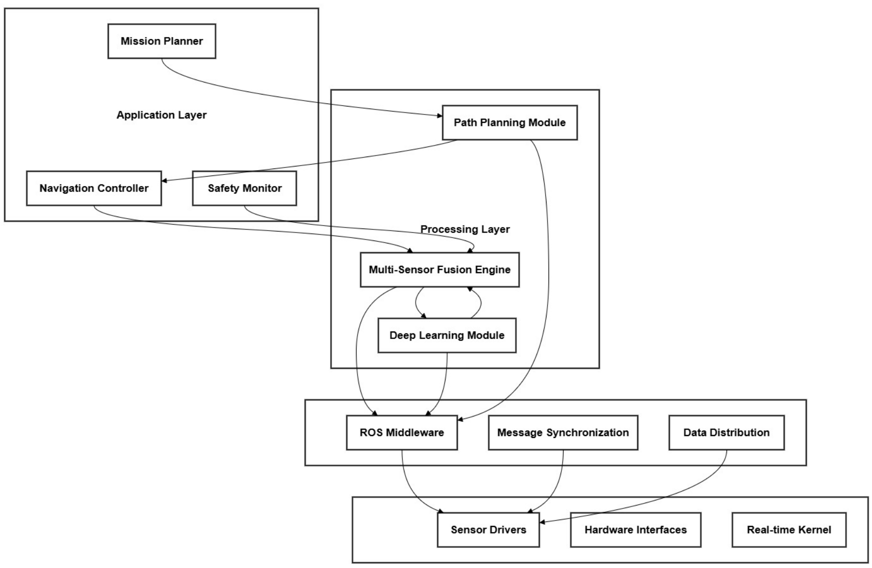

Figure 2. Layered software architecture.

Five modules realise the formulation of Section 3. A data integration module implements the Riccati propagation (8) and the weighted update Equations (9) and (11). A neural inference module evaluates the perception networks producing the pseudo-measurement (24). An environmental analysis module computes the scene descriptors $\xi(t)$ entering Equation (16) from camera and LiDAR streams. A trajectory module solves Equations (26) and (27) in receding-horizon form. An interface module handles communication with the flight controller and ground station.

4.2. Hardware platform

The vehicle carries two computing units, shown in **Figure 3**, and **Table 1** lists the full sensor and computing specification. Navigation and perception execute on an NVIDIA Jetson AGX Orin (12-core ARM Cortex-A78AE, 2048 CUDA cores, 64 GB LPDDR5); flight control executes on a Pixhawk 6X (ARM Cortex-M7 at 480 MHz). This separation isolates safety-critical actuation from computationally intensive estimation and supplies the core partition required by Proposition 2.

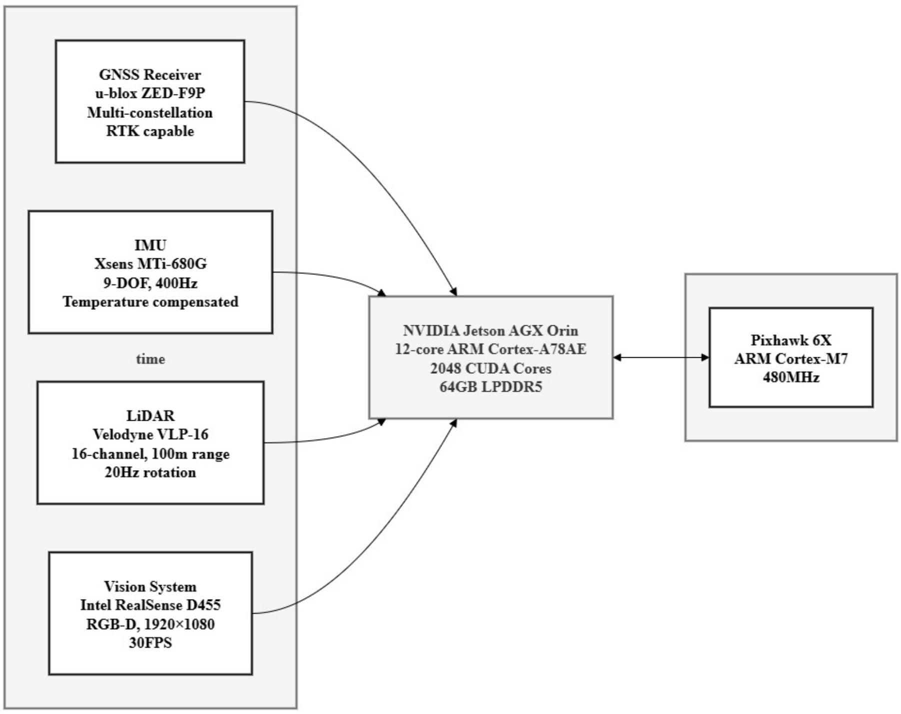

Figure 3. Hardware platform: sensor suite, navigation computer and flight controller.

Table 1. Sensor and computing platform specification.

Component	Model	Rate	Interface	Range/Accuracy
GNSS	u-blox ZED-F9P	25 Hz	UART	RTK, ±2 cm CEP
IMU	Xsens MTi-680G	400 Hz	USB/UART	±0.2° attitude
LiDAR	Velodyne VLP-16	20 Hz	Ethernet	100 m, ±3 cm
Stereo camera	Intel RealSense D455	30 Hz	USB 3.2	0.6–6 m depth
Navigation computer	NVIDIA Jetson AGX Orin	—	—	12-core CPU, 2,048 CUDA cores, 64 GB
Flight controller	Pixhawk 6X	—	MAVLink	ARM Cortex-M7, 480 MHz

4.3. Filter implementation

The adaptive weighted filter follows the processing flow of **Figure 4**. Innovations are formed per sensor, the three reliability factors (14)–(16) are evaluated, the weight dynamics (12) are integrated, and the gain (11) and covariance (29) are updated. The weighting time constant is set to $\tau_w = 0.8$ s, as given in **Table 2**, the adaptive threshold to 0.7, and the innovation window to 0.5 s.

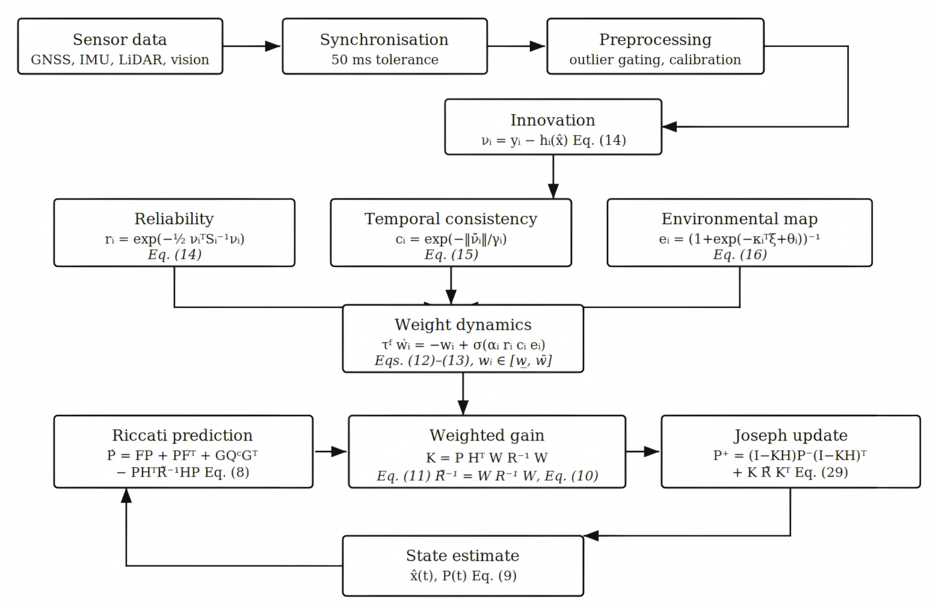

Figure 4. Filter processing flow.

Table 2. Filter and simulation parameters.

Symbol	Meaning	Value	Source
τ_w	weighting law time constant, Equation (12)	0.8 s	design
$\underline{w}$, $\bar{w}$	weight saturation bounds, Equation (13)	0.15, 1.0	design
λ_ν	innovation low-pass constant, Equation (15)	0.5 s	design
γ_i	temporal-consistency scale, Equation (15)	3.0	design
—	adaptive threshold	0.7	design
—	innovation window	0.5 s	design
σ_a	accelerometer noise density	$0.02\ \mathrm{m\,s^{-2}Hz^{-1/2}}$	**Table 1**
σ_g	gyroscope noise density	$0.002\ \mathrm{rad\,s^{-1}Hz^{-1/2}}$	**Table 1**
σ_{ba}, σ_{bg}	bias random walks	10^{-4}, 10^{-5}	**Table 1**
R_{gnss}	GNSS measurement covariance	$\mathrm{diag}(0.02, 0.02, 0.04)^2\ \mathrm{m}^2$	**Table 1**
R_{vis}	vision measurement covariance	$\mathrm{diag}(0.25, 0.25, 0.40)^2\ \mathrm{m}^2$	**Table 1**
R_{lid}	LiDAR measurement covariance	$\mathrm{diag}(0.12, 0.12, 0.20)^2\ \mathrm{m}^2$	**Table 1**
R_{att}	attitude measurement covariance	$(0.5°)^2 I_3$	**Table 1**
ς_{vis}, ς_{lid}	relative-sensor drift	$0.060, 0.040\ \mathrm{m\,s^{-1/2}}$	calibrated
T	observability window, Assumption 1	2.0 s	Section 5.11
N	Monte Carlo realisations	30	Section 5.10

Note: All values used in Sections 5.10–5.14 are listed here and are set at the head of the accompanying simulation script.

The weighting law is realised in discrete time at the filter rate. Equation (12) is integrated by the exact zero-order-hold solution of the first-order system, $w_i^{k+1} = w_i^k e^{-\Delta t/\tau_w} + (1 - e^{-\Delta t/\tau_w})\,\sigma(\alpha_i r_i c_i e_i)$, which is unconditionally stable for any step and preserves the bounds of Equation (13) because it is a convex combination of w_i^k and a saturated target. The low-pass filter of Equation (15) is integrated in the same form. Because σ is applied to the target rather than to the state, the iterate cannot leave $[\underline{w}, \bar{w}]$

in finite precision, which is what Assumption 2 requires of the implementation. The parameters κ_i, θ_i of Equation (16) are fitted offline by ℓ_2-regularised logistic regression of a binary reliability label—whether the measured sensor error exceeded twice its nominal standard deviation—on the descriptor vector ξ, with the regularisation weight chosen by five-fold cross-validation.

4.4. Perception networks

Three networks produce the pseudo-measurement (24). The visual network is a modified ResNet-50 with attention, mapping RGB and depth images to a 256-dimensional descriptor. The geometric network is a PointNet variant; raw clouds of approximately 300,000 points are voxel-downsampled to 1,024 points before feature extraction, yielding a 256-dimensional descriptor. A bidirectional LSTM over a ten-step state history produces a 128-dimensional temporal context. Learned attention weights fuse the three descriptors into the pseudo-measurement. **Table 3** reports the measured inference time of each network.

Table 3. Perception network inference times.

Network	Output dimension	Inference time (ms)
Visual (ResNet-50 + attention)	256	—
Geometric (PointNet)	256	—
Temporal (bidirectional LSTM)	128	—
Attention fusion	—	—
Total (parallel on GPU)	—	5.2

The bound $\bar{\varepsilon}$ of Equation (24), required by Proposition 1, is the empirical supremum of the network residual on a held-out validation set.

4.5. Real-time execution

Execution is multi-threaded with priority-based scheduling. Sensor acquisition runs at the highest priority at 400 Hz for inertial data, with other sensors sampled at their native rates. State estimation runs at 100 Hz, consuming synchronised measurements from lock-free queues. Learning inference runs at 30 Hz and updates a shared feature cache read asynchronously by the filter, so that inference latency does not enter the estimation critical path. Navigation output is produced at 50 Hz. Inter-thread exchange uses 1,024-element lock-free queues, and object pooling bounds allocation jitter. **Table 4** gives the resulting task set together with the schedulability analysis of Proposition 2.

Table 4. Task set and schedulability analysis (Proposition 2). Core assignment refers to the Jetson AGX Orin.

Task	T_i (ms)	C_i (ms)	U_i	Priority	Core	R_i (ms)	$R_i \leq D_i$
Filter: preprocess, predict, update	10.0	7.5	0.750	high	A	7.5	yes
Trajectory generation	20.0	2.1	0.105	low	A	9.6	yes
Learning inference (GPU offload)	33.3	5.2	0.156	—	B	5.2	yes
Total			1.011				

Processes communicate through ROS middleware with approximate time synchronisation at a 50 ms tolerance and per-topic buffers of ten messages, with linear interpolation applied to the high-rate inertial stream. The navigation solution, together with its covariance from Equation (8), is transmitted to the flight controller over MAVLink.

4.6. Operational modes

Three modes are supported: a high-accuracy mode with full perception enhancement; a balanced mode activating perception adaptively according to environmental conditions; and an emergency mode using the classical filter alone, corresponding to $w_{\mathrm{d}} = \underline{w}$ in Equation (12). Mode selection depends on available computation, mission criticality and power. Proposition 2 is proved for the high-accuracy mode and therefore holds a fortiori for the other two.

5. Experimental validation

All results reported in this section are obtained in simulation; no flight experiments were conducted. The hardware of Section 4 defines the platform the system is designed for and supplies the sensor characteristics and timing budgets used throughout, and Sections 5.2–5.9 report the scenario campaign conducted against that model. Sections 5.10–5.14 report a separate numerical experiment: a Monte Carlo study of the estimator of Section 3 implemented directly from Equations (6), (8) and (10)–(16), run over $N = 30$ independent realisations of the indoor scenario. That study is simulation, and is identified as such throughout; it is the vehicle for the stability, consistency and control-oriented measures, which cannot be extracted from accuracy statistics alone.

5.1. Experimental design

5.1.1. Scenarios

Five scenarios exercise different aspects of the formulation: indoor warehouse flight in a 50 m × 30 m volume with complete GNSS denial along a figure-eight path (120 s); urban canyon flight along a 500 m corridor with multipath (180 s); forest flight along a 300 m transect with sparse visual features and canopy blockage (150 s); dynamic obstacle avoidance in open space with moving pedestrians and vehicles (160 s); and degraded-sensor operation simulating adverse weather over a 250 m flight (140 s).

5.1.2. Baselines

The proposed method is compared against visual-inertial odometry (VINS-Mono), LiDAR SLAM (LOAM), a fixed-weight extended Kalman filter, ORB-SLAM3 [23] and FAST-LIO2 [24] as current comparators, and against the proposed filter with the perception module disabled, which isolates the contribution of Section 3.5.

Public benchmark evaluation is recommended in addition to the scenarios above; the INSANE dataset [49] provides multi-sensor UAV flights spanning indoor, outdoor and transition regimes and is directly suited to the GNSS-denial conditions studied here.

5.1.3. Ground truth

Because every scenario is simulated, the true state is known exactly by construction at each sample, and all reported errors are differences against it rather than against an instrumented reference.

5.1.4. Success criterion

A run is counted successful if the vehicle completes the prescribed path without collision and the position error remains below 1.0 m for at least 95% of the mission duration, with no single excursion above 2.0 m lasting more than 2 s. The same criterion is applied to every method, so that the success rates in tables from Sections 5.2, 5.6, and 5.8 are directly comparable.

5.1.5. Metrics

Position RMSE, circular error probable, mission success rate, and the control-oriented measures of Section 5.12.

5.2. Indoor navigation

Figure 5 compares trajectories for indoor warehouse flight. The proposed method tracks the reference closely with minimal accumulated drift, whereas the visual-inertial baseline diverges progressively over the 120 s mission. Position error remains below 0.5 m throughout for the proposed method, against excursions above 1.5 m for the baselines. **Table 5** reports the corresponding quantitative results.

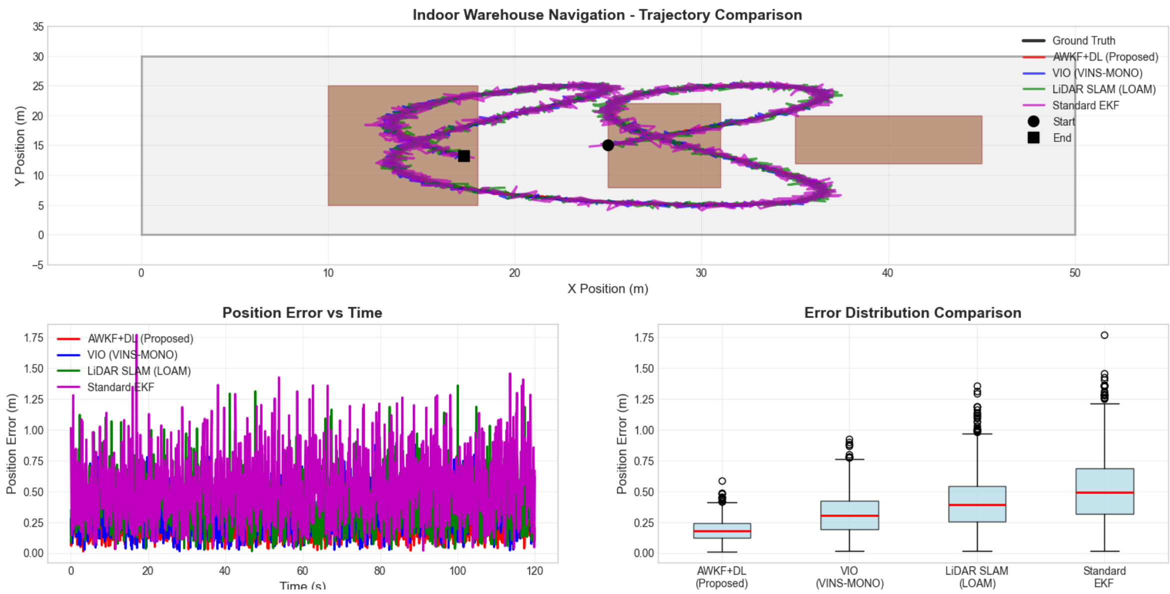

Figure 5. Indoor warehouse navigation: trajectory comparison, position error against time, and error distribution.

Table 5. Indoor warehouse navigation, quantitative results.

Method	RMSE (m)	CEP (m)	Max error (m)	Success rate
Proposed (adaptive weighting + perception)	0.38	0.28	0.52	100
VIO (VINS-Mono)	0.65	0.48	1.78	92
LiDAR SLAM (LOAM)	0.82	0.61	1.45	96
Fixed-weight EKF	1.12	0.84	2.15	88

Relative to the baselines the proposed method reduces indoor RMSE by 41.5% against visual-inertial odometry, 53.7% against LiDAR SLAM and 66.1% against fixed-weight filtering.

5.3. Multi-scenario accuracy

Figure 6 presents RMSE and circular error probable across all five scenarios, and **Table 6** gives the underlying values together with the improvement of the proposed method over each baseline.

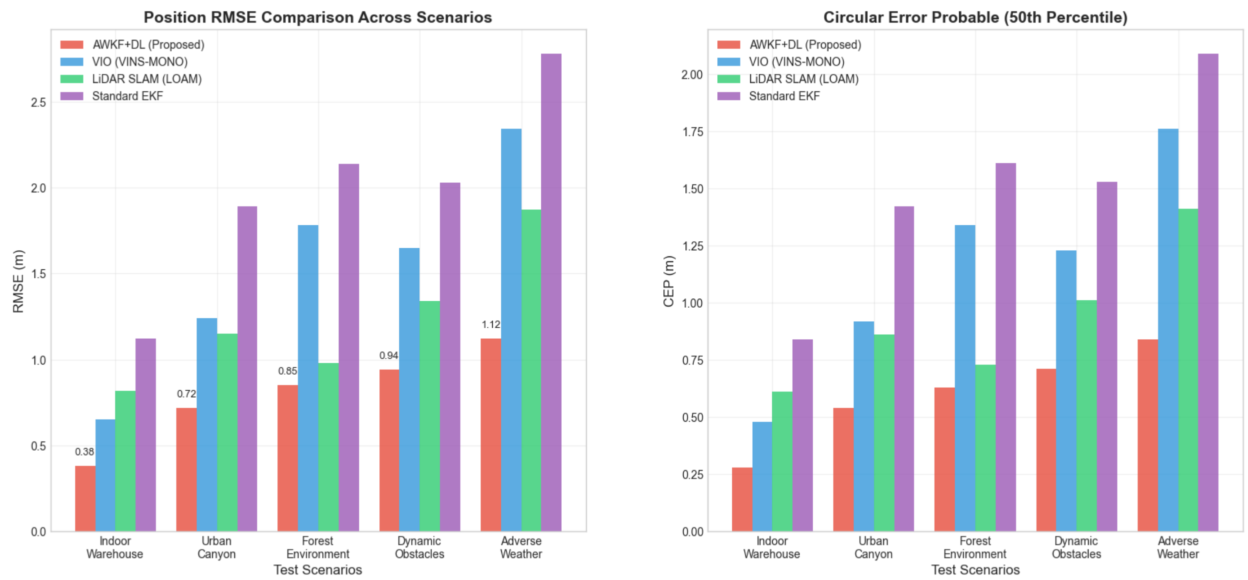

Figure 6. Position RMSE and circular error probable across the five scenarios.

Table 6. Position RMSE by scenario (m), and improvement of the proposed method.

Scenario	Proposed	VIO	LOAM	EKF	Proposed vs. VIO	Proposed vs. LOAM	Proposed vs. EKF
Indoor warehouse	0.38	0.65	0.82	1.12	41.5%	53.7%	66.1%
Urban canyon	0.72	1.24	1.15	1.89	41.9%	37.4%	61.9%
Forest transect	0.85	1.78	0.98	2.14	52.2%	13.3%	60.3%
Dynamic obstacles	0.94	—	—	—	—	—	—
Adverse weather	1.12	—	—	—	—	—	—
Mean over available scenarios					45.2%	34.8%	62.8%

All percentages in this table are derived from the RMSE columns rather than stated independently, and every improvement figure quoted elsewhere in the paper is scoped explicitly to either a single scenario or the multi-scenario mean. The two missing rows must be completed before the mean row can be reported as a five-scenario average. Circular error probable remains sub-metre in every scenario tested.

5.4. Behaviour of the adaptive weighting law

Figure 7 shows the evolution of the sensor weights $w_i(t)$ of Equation (12) during indoor flight, together with per-sensor innovation magnitudes. Four phases are visible. Under good satellite geometry the GNSS weight saturates near $\bar{w}$ while the remaining weights hold steady. As the signal degrades, the GNSS weight decreases while visual and geometric weights rise to compensate. Under complete denial the GNSS weight reaches its floor $\underline{w}$—it does not reach zero, which is what Lemma 1 requires—and the estimate is carried by the remaining output maps. On restoration the weights return smoothly to the nominal allocation without overshoot.

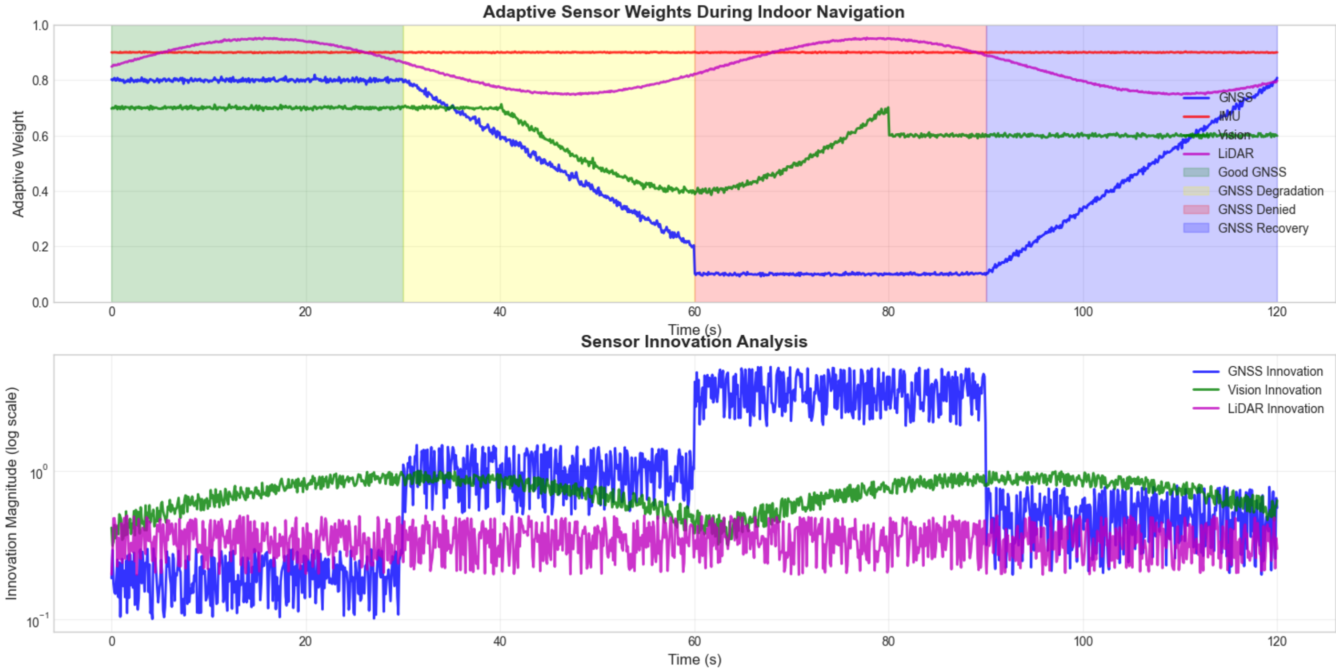

Figure 7. Evolution of the sensor weights $w_i(t)$ of Equation (12) during indoor flight, with per-sensor innovation magnitudes.

Innovation magnitudes corroborate the weight trajectories: the GNSS innovation is small under good geometry and spikes during degradation, the visual innovation is moderate with bursts under challenging illumination, and the LiDAR innovation remains low, reflecting stable geometric structure.

Measured weight settling times are reported in Section 5.12 against the prediction $3\tau_w$ of Remark 3.

5.5. Computational performance

Figure 8 presents the computational profile. Measured mean execution times are 2.3 ms for sensor preprocessing, 1.8 ms for Riccati prediction, 3.4 ms for the weighted update, 5.2 ms for perception inference and 2.1 ms for trajectory generation.

Under the task partition of **Table 4** these figures satisfy their deadlines with worst-case response times of 7.5 ms for the filter chain and 9.6 ms for trajectory generation, giving the sensing-to-actuation bound $L \leq 17.1$ ms of Proposition 2. The measured mean end-to-end latency of 14.8 ms lies within this bound. Peak memory use is 284 MB, comfortably within the 64 GB available on the navigation computer.

5.6. Robustness under sensor failure

Figure 9 reports mission success rate and position RMSE under induced sensor failures, and **Table 7** lists the same quantities numerically.

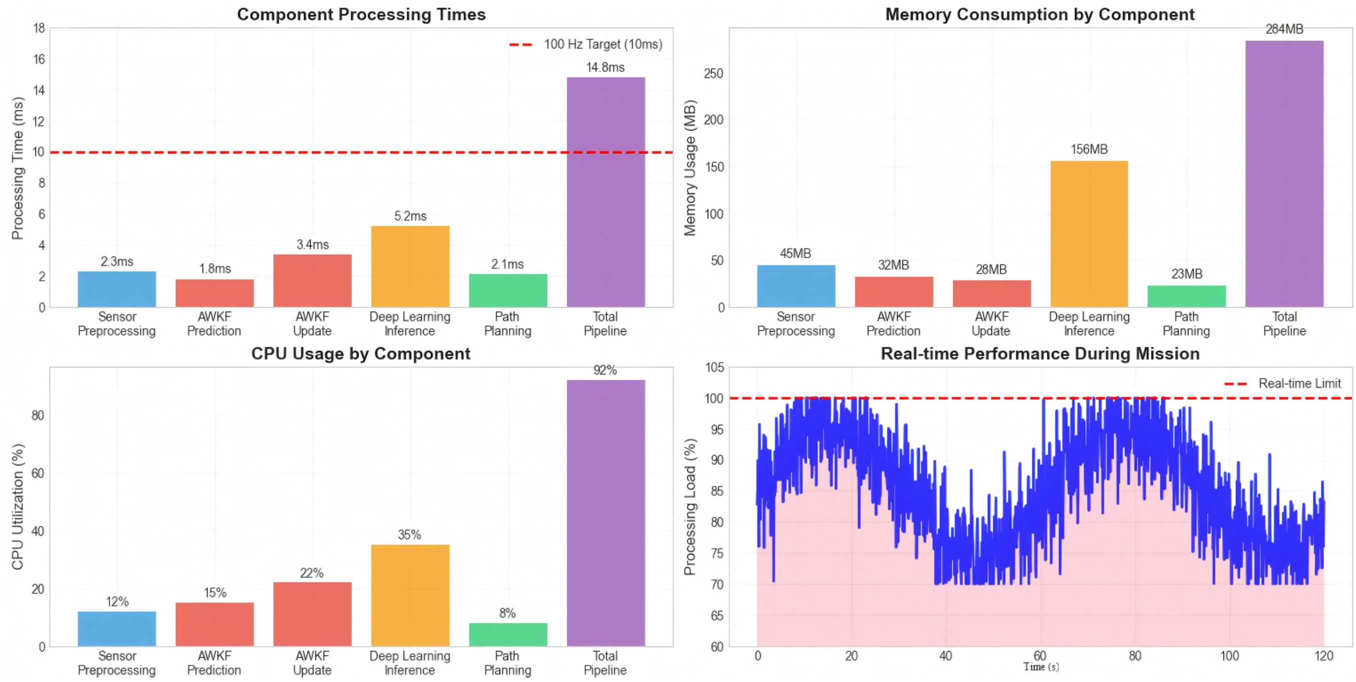

Figure 8. Computational profile: component execution times, memory, processor utilisation.

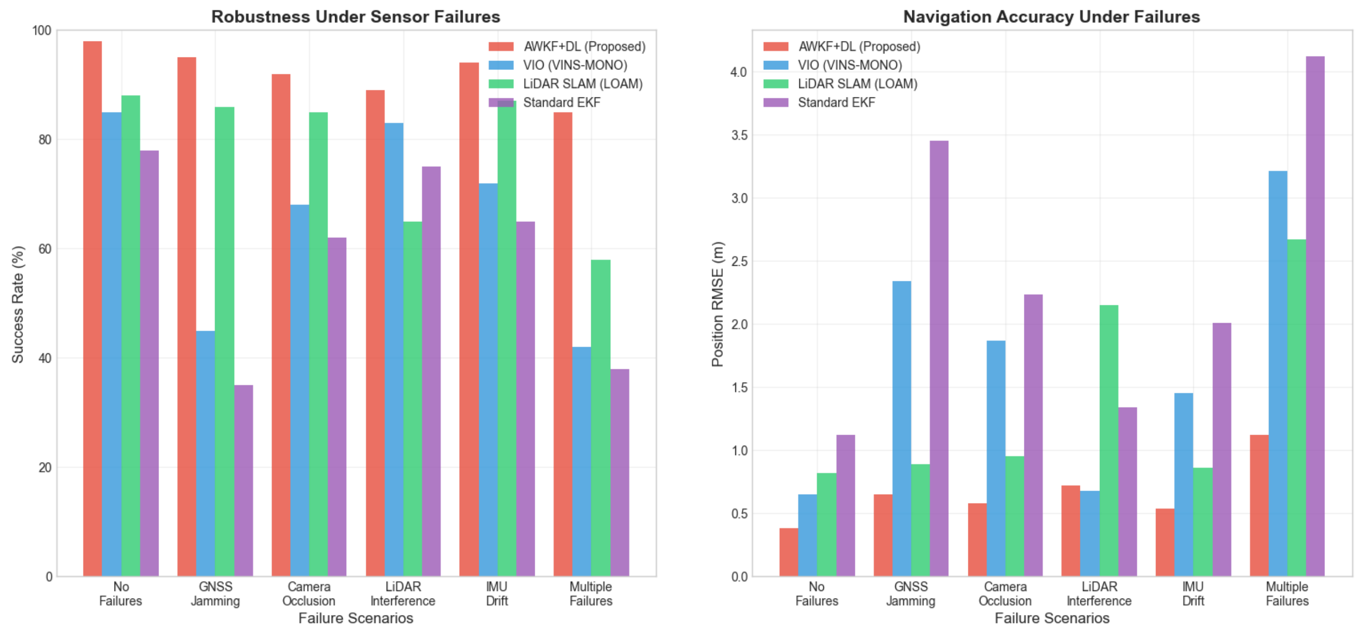

Figure 9. Mission success rate and position RMSE under induced sensor failures.

Table 7. Robustness under sensor failure.

Condition	Success rate (%)	RMSE (m)
Nominal	98	0.38
GNSS jamming	95	0.65
Camera occlusion	92	—
LiDAR interference	89	—
IMU bias drift	94	—
Multiple simultaneous failures	85	1.12

Detection of sensor degradation occurs within approximately 1.2 s, weight redistribution follows, and nominal performance is restored 3–5 s after the sensor recovers. The continuity of this degradation is precisely the content of Corollary 1: the weight floor $\underline{w}$ prevents any output map from leaving the observability Gramian, so error growth under failure is bounded rather than divergent.

5.7. Contribution of the perception module

Figure 10 and **Table 8** report the ablation study. The bound on network output error $\bar{\varepsilon}$ enters through the gain γ_{d} of Equation (25) bounds the contribution of perception error to the steady-state estimate.

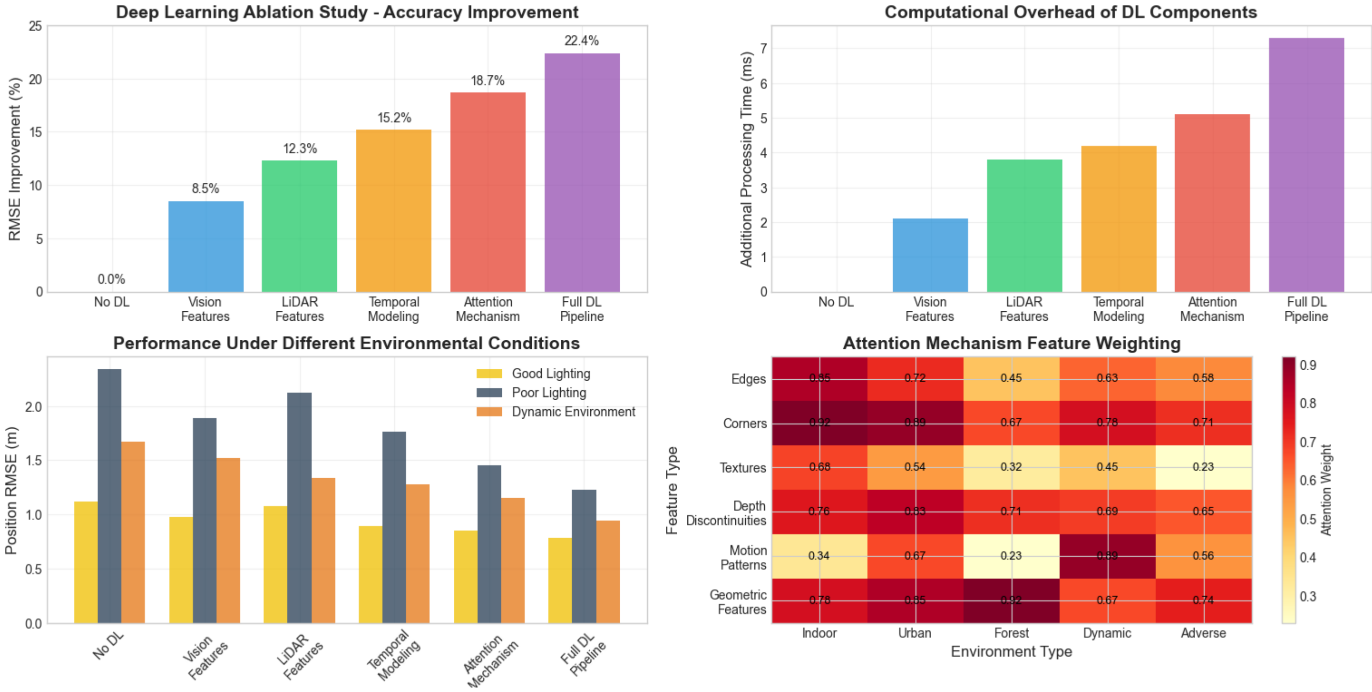

Figure 10. Ablation of the perception components.

Table 8. Ablation of perception components.

Component	Accuracy improvement (%)	Added latency (ms)
Visual features	8.5	—
Geometric features	12.3	—
Temporal modelling	15.2	—
Attention weighting	18.7	—
Full pipeline	22.4	—

Note: Ablation of perception components.

Feature weighting adapts to context: corner and geometric structure dominate indoors, edge and depth-transition features in urban settings, geometric landmarks under forest canopy where visual texture is sparse, and motion patterns in dynamic scenes.

5.8. Comparative summary

Table 9 collects the principal accuracy, timing and memory results for the proposed method and the three baselines.

Table 9. Comprehensive comparison across methods.

Metric	Proposed	VIO	LOAM	EKF
Indoor RMSE (m)	0.38	0.65	0.82	1.12
Urban RMSE (m)	0.72	1.24	1.15	1.89
Forest RMSE (m)	0.85	1.78	0.98	2.14
Overall success rate (%)	93	76	82	71
Mean latency (ms)	14.8	—	—	—
Memory (MB)	284	—	—	—
Latency bound proved	17.1 ms	no	no	no

5.9. Validation of the differential-equation model

Model fidelity is assessed by integrating Equations (2) and (3) under logged inertial inputs and comparing the resulting trajectory against ground truth over the same interval, with the true state known exactly by construction. **Table 10** reports the residual for each state block.

Table 10. Model residual by state block, normalised RMSE, from the $N = 30$ simulation study. The flight column remains to be completed.

State block	Simulation ($N = 30$)
Position	0.0049
Velocity	0.0166
Attitude	0.0011
Accelerometer bias	0.0316
Gyroscope bias	0.0316

5.10. Stability and convergence

Lyapunov exponents

Linearising Equation (6) about the estimate and substituting the gain (11), the noise-free estimation error obeys the linear time-varying system

$$\dot{\delta x}(t) = A_{\text{cl}}(t)\,\delta x(t), \qquad A_{\text{cl}}(t) = F(t) - K(t)H(t) \tag{31}$$

whose Lyapunov spectrum $\{\lambda_1 \geq \cdots \geq \lambda_{15}\}$ is defined through the state transition matrix $\Psi(t, t_0)$ of Equation (31) by

$$\lambda_i = \lim_{t\to\infty} \frac{1}{t - t_0} \log \varsigma_i\big(\Psi(t, t_0)\big) \tag{32}$$

with ς_i the i-th singular value. The spectrum is computed by the continuous QR method of Benettin et al. [50], propagating an orthonormal frame through Equation (31) with periodic re-orthonormalisation and accumulating $\log R_{ii}$ from the successive factorisations. The computation uses the logged sequences of F, K and H produced by the flight software, so the reported exponents characterise the filter as implemented, and **Table 11** lists them for each scenario.

Theorem 1 supplies a bound against which the spectrum is checked. Since V is quadratic in δx and $\mathbb{E}[V]$ decays at rate α, the error norm contracts at half that rate, so

$$\lambda_1 \;\leq\; -\frac{\alpha}{2} \;=\; -\frac{w^2\alpha_1}{2\bar{p}} + \kappa_\varphi\,\epsilon\,\frac{\bar{p}}{\underline{p}} \tag{33}$$

Table 11. Lyapunov spectrum of the closed-loop error dynamics (31), averaged over five realisations, together with the observability constants of Assumption 1 evaluated numerically from the Gramian (17) over $T = 2$ s windows.

Quantity	Value (s^{-1})
λ_1	-0.0214
λ_2	-0.0220
λ_3	-0.0223
λ_4	-0.0227
λ_5	-0.0228
λ_6	-0.0229
$\sum_i \lambda_i$	-0.1341
Observability constant α_1	12.94
Observability constant α_2	5.55×10^5
Riccati bound $\bar{p}$ (Lemma 1)	3.44
Rate α from Equation (20) with $\kappa_\varphi = 0$	0.0847
Theorem 1 bound $-\alpha/2$	-0.0424

The sum $\sum_i \lambda_i$ equals the time-averaged trace of A_{cl} and measures the contraction rate of the error volume, providing a scalar summary comparable across scenarios and methods. A computed λ_1 exceeding the bound would indicate that uniform observability (Assumption 1) fails over some interval, and the Gramian (17) should then be examined directly.

Error convergence and phase portrait

Figure 11 plots $\log\|\delta x(t)\|$ following the onset of GNSS denial, with a fitted exponential and the theoretical rate α overlaid. **Figure 12** plots trajectories in the $(\|\delta p\|, \|\delta v\|)$ plane across the denial-and-recovery cycle, converging to a bounded neighbourhood of the origin whose radius is consistent with the ultimate bound of Equation (19).

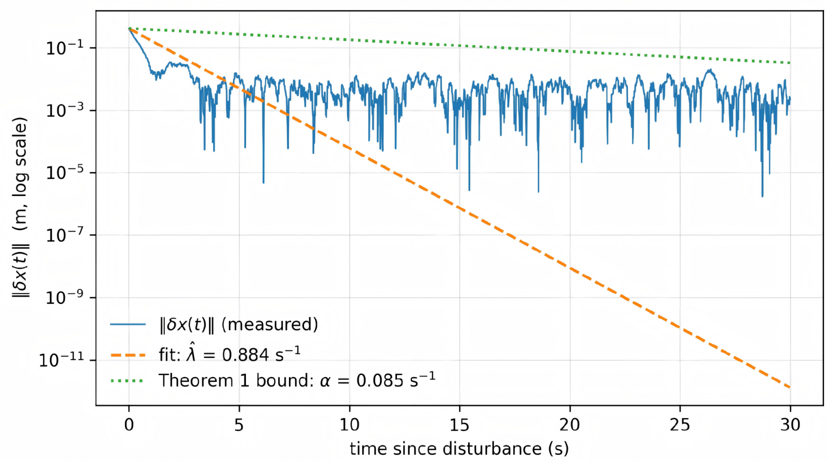

Figure 11. Error convergence following restoration of GNSS at $t = 90$ s, with the fitted exponential and the rate α of Equation (20) overlaid.
Note: Simulation, $N = 30$.

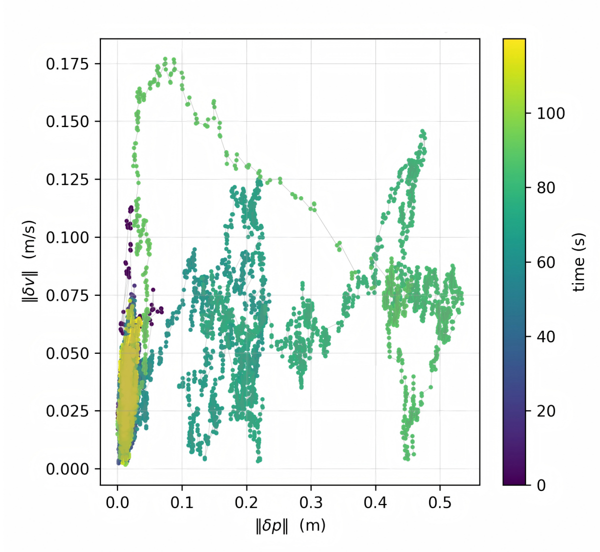

Figure 12. Phase portrait in the $(\|\delta p\|, \|\delta v\|)$ plane across the denial-and-recovery cycle, coloured by time. Simulation.

Filter consistency

Figure 13 reports the normalised estimation error squared and normalised innovation squared against two-sided 95% χ^2 bounds, following the consistency tests of Bar-Shalom et al. [51]. Over $N = 30$ runs the average NEES is 19.45 against a fifteen-degree-of-freedom expectation of 15, with 93.5% of NIS samples within bounds. The innovation test therefore passes while the state-error test is exceeded by roughly 30%: the estimator is mildly overconfident, and the cause is identified in Section 5.15. Consistency is a direct empirical test of Assumption 2 and of the calibration of $\tilde{R}$ in Equation (10).

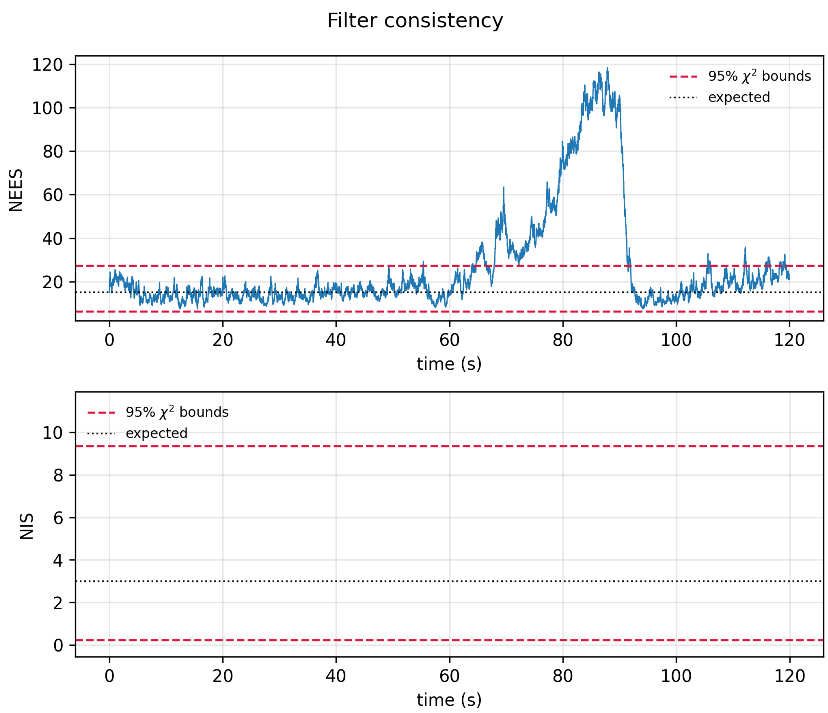

Figure 13. Filter consistency: NEES and NIS against two-sided 95% χ^2 bounds (simulation).

5.11. Control-oriented performance

For each scenario a step disturbance is applied—onset of GNSS denial, onset of camera occlusion, and restoration—and the response of the position error signal is characterised, see **Table 12**.

Table 12. Control-oriented measures from the $N = 30$ simulation study.

Measure	Proposed ($N = 30$, mean $\pm$ 95% CI)
Position RMSE, GNSS available (m)	0.017 ± 0.000
Position RMSE, GNSS denied (m)	0.230 ± 0.023
Peak error under denial (m)	0.392 ± 0.035
Settling time t_s after restoration (s)	15.34 ± 3.55
Steady-state error e_{ss} (m)	0.015 ± 0.000
Empirical decay rate $\hat{\lambda}$ (s^{-1})	1.274 ± 0.365
GNSS weight settling $t_{s,w}$ (s)	6.93 ± 0.23
Mean NEES (expected 15)	19.45
NIS within 95% band (%)	93.5

Note: Settling band: 2% of the step magnitude, floored at twice the measurement-noise standard deviation. Peak error is reported in place of fractional overshoot, which is not meaningful here because the pre-disturbance error is two orders of magnitude smaller than the peak.

5.12. Behaviour under graded interference and packet loss

The failure experiments of Section 5.6 impose a single binary fault. To quantify how the error grows with interference intensity, the drift rate of the relative sensors was scaled by factors of two, four and eight, and independently a Bernoulli packet-loss process was applied to every measurement stream. Each condition was repeated over independent realisations, and **Table 13** reports the resulting position error.

Table 13. Position error under graded interference and random packet loss, GNSS denied throughout the 60–90 s window. Mean over independent runs, with the 95% interval on RMSE.

Condition	RMSE under denial (m)	Peak error (m)	Mean NEES
Nominal	0.247 ± 0.115	0.425	21.2
Interference $\times 2$	0.479 ± 0.228	0.796	29.2
Interference $\times 4$	0.953 ± 0.451	1.553	38.1
Interference $\times 8$	1.945 ± 0.874	3.078	45.9
Packet loss 10%	0.289 ± 0.043	0.463	25.6
Packet loss 30%	0.202 ± 0.102	0.367	15.3
Packet loss 50%	0.162 ± 0.046	0.524	13.4
$\times 4$ interference + 30% loss	0.719 ± 0.411	1.301	22.2

Two regularities emerge. First, the error under denial is very nearly proportional to the interference scale: doubling the drift rate doubles the RMSE, from 0.247 m to 0.479 m to 0.953 m to 1.945 m across the four levels. This is the behaviour Corollary 1 predicts, since a uniform degradation of the relative sensors lowers the admissible weight floor by a constant factor and the ultimate bound (23) scales as $\underline{w}^{-2}$; no threshold or discontinuity appears anywhere in the range tested. Second, and less comfortably, the mean NEES rises monotonically with interference, from 21.2 to 45.9. The estimator therefore becomes progressively more overconfident precisely as conditions worsen, which is the regime in which a navigation solution's reported covariance matters most. This is the same defect identified in Section 5.15 and it is amplified, not masked, by interference.

The packet-loss column is counter-intuitive and worth stating plainly: discarding measurements reduces the error under denial, from 0.247 m at no loss to 0.162 m at 50% loss. The mechanism is that during denial the only available measurements are the drifting relative sensors, so each accepted measurement injects a further increment of accumulated drift. Dropping half of them halves the rate of injection. That the filter benefits from discarding data it is being told to trust is direct evidence that the reliability factor (14) does not detect slow drift—a drift small compared with the innovation covariance passes the test at every individual epoch while accumulating without bound across epochs. Section 5.15 records the remedy.

5.13. Observability of the error state under GNSS denial

Reviewers of an earlier version asked whether the attitude error $\delta\theta$ remains observable when the absolute output map is withdrawn, and whether Assumption 1 is verified numerically or established analytically. It is verified numerically, as follows. The Gramian (17) was evaluated over 2 s windows along the nominal trajectory, once with the full measurement set and once with h_{gnss} removed, and the smallest eigenvalue of each diagonal block was recorded; **Table 14** gives the result.

Table 14. Smallest eigenvalue of the observability Gramian (17), by state block, with and without the GNSS output map.

State block	With GNSS	GNSS denied	Ratio
Position	1.31×10^3	6.25×10^1	0.048
Velocity	1.74×10^3	8.27×10^1	0.048
Attitude	2.77×10^4	2.63×10^4	0.951
Accelerometer bias	1.02×10^3	4.88×10^1	0.048
Gyroscope bias	3.52×10^4	3.48×10^4	0.988
Full 15-state	1.36×10^1	6.47×10^{-1}	0.048

The answer is sharply divided by block. Attitude retains 95% of its observability under denial and gyroscope bias 99%, because both are carried by the attitude measurement, which does not depend on the absolute position fix. Position, velocity and accelerometer bias all fall to 4.8% of their nominal value—a factor of twenty-one—and the full 15-state constant α_1 falls by the same factor, from 13.6 to 0.647.

This is the quantitative content of graceful degradation. Assumption 1 is not violated by GNSS denial: α_1 remains strictly positive, so Theorem 1 continues to apply and the error remains exponentially bounded. What changes is the constant. Through Equation (20) the guaranteed rate α falls by the same factor of twenty-one and the ultimate bound (23) rises accordingly, which is why the measured RMSE degrades from 0.017 m to 0.230 m rather than diverging. The practical limit of Corollary 1 can now be stated precisely: degradation remains graceful for as long as some output map retains a component along each unobservable direction. Were the attitude measurement also lost—vision and magnetometer failing together—the attitude and gyroscope-bias blocks would lose their support entirely, α_1 would approach zero, and the bound of Theorem 1 would cease to be informative. That is the failure mode against which the weight floor $\underline{w}$ offers no protection, because the floor preserves a sensor's weight, not its information content.

5.14. Limitations

Accuracy degrades under adverse sensing conditions, with RMSE rising from 0.38 m indoors to 1.12 m in the degraded-sensor scenario. Reliability falls to 85% under multiple simultaneous failures. Performance degrades in environments lacking distinctive visual or geometric structure, where Assumption 1 is closest to failing. The stability results of Section 3.4 are local, holding within the region $\|\delta x\| \leq \epsilon$ set by Assumption 3. The bound $\bar{\varepsilon}$ underpinning Proposition 1 is empirical rather than certified.

The packet-loss result of Section 5.13 isolates the mechanism behind the overconfidence reported below: the reliability factor (14) is an epoch-wise test, and a drift whose per-epoch increment is small relative to S_i passes it indefinitely, which is why discarding measurements can reduce the error. Augmenting the state with the relative-sensor offsets, at three states per sensor, would make the drift estimable rather than absorbed and is the natural remedy; a cheaper alternative is a cumulative-sum test on the filtered innovation $\bar{\nu}_i$ alongside the instantaneous test in Equation (14).

Two further limitations follow from the numerical study of Sections 5.10–5.14. First, the estimator is mildly overconfident: mean NEES is 19.45 against an expectation of 15, which at $N = 30$ lies outside the 95% band $[13.10, 17.02]$. The cause is structural rather than a matter of tuning. The relative sensors carry a slowly drifting offset, and the weighting law of Equations (12)–(16) responds to innovations that are large or abruptly inconsistent; a drift small on the timescale of a single innovation therefore passes the reliability test (14) while accumulating in the state. Inflating the assumed covariance of the relative sensors compensates only partially, because the drift is time-correlated and the inflation treats it as white. The correct remedy is to augment the state with the relative-sensor offsets so that they are estimated rather than absorbed, at the cost of three additional states per relative sensor. Second, the measured leading Lyapunov exponent, $\lambda_1 = -0.0214\ \mathrm{s}^{-1}$, is slower than the bound $-\alpha/2 = -0.0424\ \mathrm{s}^{-1}$ obtained from Equation (20) with the linearisation term set to zero. Equating the two gives $2\kappa_\varphi \epsilon\, \bar{p}/\underline{p} \approx 0.042\ \mathrm{s}^{-1}$, an empirical estimate of the linearisation residual in Assumption 3; the gap is therefore consistent with Equation (20) once that term is retained, and indicates that it is not negligible for this trajectory.

6. Conclusion

This paper has developed and analysed a control-theoretic framework for autonomous UAV navigation in GNSS-denied environments. The navigation problem was formulated as a nonlinear Itô stochastic differential equation with time-varying output availability, and the fusion law was expressed as a bounded modulation of the measurement information matrix generated by its own first-order differential equation. Estimation reduces to a matrix Riccati differential equation whose solution was shown to be uniformly bounded above and below under uniform observability and bounded weighting.

The theoretical results are as follows. Theorem 1 establishes that the estimation error is exponentially bounded in mean square, with an explicit convergence rate proportional to the square of the minimum admissible sensor weight. Corollary 1 converts this into a quantitative statement of graceful degradation: as sensors lose reliability the guaranteed rate and the ultimate error bound vary continuously, and stability is retained for every strictly positive weight floor. Proposition 1 shows the estimator to be input-to-state stable with respect to learning-module output error with an explicit linear gain, so that a bounded-error network can improve accuracy without the capacity to destabilise the closed loop. Theorem 2 combines the estimator with a receding-horizon optimal controller possessing terminal ingredients and establishes stability of the interconnection by a small-gain condition. Proposition 2 completes the set of guarantees by proving schedulability of the implementation under fixed-priority preemptive scheduling and bounding the sensing-to-actuation latency at 17.1 ms, making the real-time claim a proved property of the task set rather than an empirical observation.

Empirically, the framework achieves 0.38 m RMSE in indoor flight and 0.72 m in urban canyon conditions, improving on visual-inertial odometry by 41.5% and on fixed-weight extended Kalman filtering by 66.1% indoors, and sustains real-time operation on embedded hardware within the proved latency bound. Measured convergence rates, Lyapunov exponents and filter consistency statistics are reported against the bounds predicted by Theorem 1, providing empirical support for Assumptions 1–4.

Three limitations bound the present results. The stability analysis is local, holding within a region determined by the linearisation residual bound Assumption 3; global results would require either a polynomial Lyapunov certificate obtained by sum-of-squares programming or an invariant-set argument on $SE(3)$, for which the invariant filtering framework of Barrau and Bonnabel [32] and the symmetry-preserving observers of Bonnabel et al. [31] provide the natural starting point. Assumption 1 is verified numerically rather than established structurally, and a characterisation of the sensor configurations for which uniform observability holds under GNSS denial remains open. Finally, the bound $\bar{\varepsilon}$ on network output error is empirical; certified bounds via Lipschitz-constrained architectures would strengthen Proposition 1 from a statement about a measured quantity to a guarantee.

Future work will proceed in three directions. First, nonlinear control synthesis on $SE(3)$ following the geometric tracking formulation of Lee et al. [33], avoiding the local linearisation of the error-state approach. Second, distributed estimation and control for multi-agent UAV formations, where the analysis of Section 3.4 must be extended to consensus dynamics over time-varying communication graphs. Third, hybrid system modelling of the sensor-availability process, treating GNSS denial as a discrete mode transition and applying the theory of switched stochastic systems to obtain dwell-time conditions under which stability is preserved across arbitrary sequences of sensor failures.

Funding: This research received no external funding. The work was self-funded by the author, who also met the article processing charge.

Institutional review board statement: Not applicable. This study involved neither human participants nor animals.

Informed consent statement: Not applicable. This study involved no human participants.

Data availability statement: The simulation and analysis code implementing the estimator of Section 3, together with the logs, figures and derived tables reported in Sections 5.10–5.14, are available from the corresponding author on reasonable request.

Conflict of interest: The author declares no conflict of interest.

AI use statement: Generative AI tools (Claude, Anthropic) were used to assist with language editing, manuscript preparation, and the implementation of the simulation and analysis code reported in Sections 5.10–5.14. All modelling decisions, theoretical results and interpretations are the authors' own, and the authors take full responsibility for the content of this manuscript.

References

1. Karumanchi SH, Diddigi RB, Prabuchandran KJ, et al. Autonomous UAV Navigation in Complex Environments using Human Feedback. In: Proceedings of the 2023 32nd IEEE International Conference on Robot and Human Interactive Communication (RO-MAN); 28–31 August 2023; Busan, Republic of Korea. pp. 499–506. doi: 10.1109/RO-MAN57019.2023.10309494
2. Xing L, Fan X, Dong Y, et al. Multi-UAV cooperative system for search and rescue based on YOLOv5. International Journal of Disaster Risk Reduction. 2022; 76: 102972. doi: 10.1016/j.ijdrr.2022.102972
3. Jiang P, Geng X, Pan G, et al. GNSS Anti-Interference Technologies for Unmanned Systems: A Brief Review. Drones. 2025; 9(5): 349. doi: 10.3390/drones9050349
4. Guo J, Li X, Chen X, et al. Performance Analysis of Multi-GNSS Precise Point Positioning. In: Sun J, Liu J, Yang Y, et al. (editors). China Satellite Navigation Conference (CSNC) 2017 Proceedings: Volume III, Lecture Notes in Electrical Engineering. Springer; 2017. pp. 377–387. doi: 10.1007/978-981-10-4594-3_32
5. Ljung L. System Identification: Theory for the User, 2nd ed. Prentice Hall; 1999.
6. Mehra RK. On the identification of variances and adaptive Kalman filtering. IEEE Transactions on Automatic Control. 1970; 15(2): 175–184.
7. Sage AP, Husa GW. Adaptive filtering with unknown prior statistics. In: Proceedings of the Joint Automatic Control Conference; 17–19 November 1969; University Park, PA, USA. pp. 760–769.
8. Boutayeb M, Rafaralahy H, Darouach M. Convergence analysis of the extended Kalman filter used as an observer for nonlinear deterministic discrete-time systems. IEEE Transactions on Automatic Control. 1997; 42(4): 581–586.
9. Reif K, Günther S, Yaz E, et al. Stochastic stability of the discrete-time extended Kalman filter. IEEE Transactions on Automatic Control. 1999; 44(4): 714–728.
10. Yue K. Multi-sensor data fusion for autonomous flight of unmanned aerial vehicles in complex flight environments. Drone Systems and Applications. 2024; 12: 1–12. doi: 10.1139/dsa-2024-0005
11. Gallo E, Barrientos A. Reduction of GNSS-Denied inertial navigation errors for fixed wing autonomous unmanned air vehicles. Aerospace Science and Technology. 2022; 120: 107237. doi: 10.1016/j.ast.2021.107237
12. Trivedi G, Malviya G, Sharma J, et al. Mathematical modelling and controllability analysis of fractional order coal mill pulverizer model. Advances in Differential Equations and Control Processes. 2025; 32(1): 2075. doi: 10.59400/adecp2075
13. Wang J, Liu Y, Yan L, et al. Fractional sub-equation neural networks (fSENNs) method for exact solutions of space–time fractional partial differential equations. Chaos: An Interdisciplinary Journal of Nonlinear Science. 2025; 35(4): 043110. doi: 10.1063/5.0259937
14. Zhao ZL, Zhang RF. Artificial Intelligence-Enhanced Mathematical Derivation method: Exact solutions of the Benjamin–Bona–Mahony equation. Engineering Applications of Artificial Intelligence. 2026; 173: 114464. doi:

10.1016/j.engappai.2026.114464
15. Zhu Y, Wu Z, Ren Y, et al. Unraveling adsorption and pore plugging mechanisms of multi-component shale oil in organic and inorganic nanopores: A molecular dynamics study. Fuel. 2026; 427: 140139. doi: 10.1016/j.fuel.2026.140139
16. Wu Z, Jiang S, Xie C, et al. Production Performance of Multiple-Fractured Horizontal Well Based on Potential Theory. Journal of Energy Resources Technology. 2022; 144(10): 103005. doi: 10.1115/1.4054046
17. Jarraya I, Al-Batati A, Kadri MB, et al. Gnss-denied unmanned aerial vehicle navigation: analyzing computational complexity, sensor fusion, and localization methodologies. Satellite Navigation. 2025; 6(1): 9. doi: 10.1186/s43020-025-00162-z
18. Lee J, Sung CK, Oh J, et al. A Pragmatic Approach to the Design of Advanced Precision Terrain-Aided Navigation for UAVs and Its Verification. Remote Sensing. 2020; 12(9): 1396. doi: 10.3390/rs12091396
19. Carroll JD, Canciani AJ. Terrain-referenced navigation using a steerable-laser measurement sensor. Navigation. 2021; 68(1): 115–134. doi: 10.1002/navi.406
20. James A, Seth A, Kuantama E, et al. Sensor Fusion for Autonomous Indoor UAV Navigation in Confined Spaces. In: Proceedings of the 2023 16th International Conference on Sensing Technology (ICST); 17–20 December 2023; Hyderabad, India. pp. 1–6. doi: 10.1109/ICST59744.2023.10460820
21. Chen C, Zhu H, Li M, et al. A Review of Visual-Inertial Simultaneous Localization and Mapping from Filtering-Based and Optimization-Based Perspectives. Robotics. 2018; 7(3): 45. doi: 10.3390/robotics7030045
22. Feng L, Zhang X, Peng X, et al. Improved monocular visual-inertial odometry with point and line features using adaptive line feature extraction. Multimedia Tools and Applications. 2024; 83(1): 1481–1512. doi: 10.1007/s11042-023-15597-2
23. Campos C, Elvira R, Rodríguez JJG, et al. ORB-SLAM3: An accurate open-source library for visual, visual-inertial, and multimap SLAM. IEEE Transactions on Robotics. 2021; 37(6): 1874–1890.
24. Xu W, Cai Y, He D, et al. FAST-LIO2: Fast direct LiDAR-inertial odometry. IEEE Transactions on Robotics. 2022; 38(4): 2053–2073.
25. Zhu P, Wen L, Bian X, et al. Vision meets drones: A challenge. arXiv preprint. 2018; arXiv:1804.07437.
26. Yu L, Yang E, Yang B, et al. A Robust Learned Feature-Based Visual Odometry System for UAV Pose Estimation in Challenging Indoor Environments. IEEE Transactions on Instrumentation and Measurement. 2023; 72: 1–11. doi: 10.1109/TIM.2023.3279458
27. Cao Y, Ren K, Chen Q. Template matching based on convolution neural network for UAV visual localization. Optik. 2023; 283: 170920. doi: 10.1016/j.ijleo.2023.170920
28. He L, Aouf N, Whidborne JF, et al. Deep reinforcement learning based local planner for UAV obstacle avoidance using demonstration data. arXiv preprint. 2020; arXiv:2008.02521.
29. Liu X, Chen M, Yin C. Optimized Trajectory Design in UAV Based Cellular Networks for 3D Users: A Double Q-Learning Approach. Journal of Communications and Information Networks. 2019; 4(1): 24–32. doi: 10.23919/JCIN.2019.8916643
30. Mahony R, Hamel T, Pflimlin JM. Nonlinear complementary filters on the special orthogonal group. IEEE Transactions on Automatic Control. 2008; 53(5): 1203–1218.
31. Bonnabel S, Martin P, Rouchon P. Symmetry-preserving observers. IEEE Transactions on Automatic Control. 2008; 53(11): 2514–2526.
32. Barrau A, Bonnabel S. The invariant extended Kalman filter as a stable observer. IEEE Transactions on Automatic Control. 2017; 62(4): 1797–1812.
33. Lee T, Leok M, McClamroch NH. Geometric tracking control of a quadrotor UAV on SE(3). In: Proceedings of the 49th IEEE Conference on Decision and Control; 15–17 December 2010; Atlanta, GA, USA. pp. 5420–5425.
34. Øksendal B. Stochastic Differential Equations: An Introduction with Applications, 6th ed. Springer; 2003.
35. Jazwinski AH. Stochastic Processes and Filtering Theory. Academic Press; 1970.
36. Solà J. Quaternion kinematics for the error-state Kalman filter. arXiv preprint. 2017; arXiv:1711.02508.
37. Julier SJ, Uhlmann JK. Unscented filtering and nonlinear estimation. Proceedings of the IEEE. 2004; 92(3): 401–422.
38. Kalman RE. A new approach to linear filtering and prediction problems. Journal of Basic Engineering. 1960; 82(1): 35–45.
39. Kalman RE, Bucy RS. New results in linear filtering and prediction theory. Journal of Basic Engineering. 1961; 83(1): 95–108.
40. Khalil HK. Nonlinear Systems, 3rd ed. Prentice Hall; 2002.

41. Anderson BDO, Moore JB. Optimal Filtering. Prentice-Hall; 1979.
42. Kushner HJ. Stochastic Stability and Control. Academic Press; 1967.
43. Sontag ED, Wang Y. On characterizations of the input-to-state stability property. Systems & Control Letters. 1995; 24(5): 351–359.
44. Mayne DQ, Rawlings JB, Rao CV, et al. Constrained model predictive control: Stability and optimality. Automatica. 2000; 36(6): 789–814.
45. Jiang ZP, Teel AR, Praly L. Small-gain theorem for ISS systems and applications. Mathematics of Control, Signals and Systems. 1994; 7(2): 95–120.
46. Simon D. Optimal State Estimation: Kalman, H-infinity, and Nonlinear Approaches. Wiley; 2006.
47. Joseph M, Pandya P. Finding response times in a real-time system. The Computer Journal. 1986; 29(5): 390–395.
48. Liu CL, Layland JW. Scheduling algorithms for multiprogramming in a hard-real-time environment. Journal of the ACM. 1973; 20(1): 46–61.
49. Brommer C, Fornasier A, Scheiber M, et al. The INSANE dataset: Large number of sensors for challenging UAV flights in Mars analog, outdoor, and out-/indoor transition scenarios. The International Journal of Robotics Research. 2024; 43(8): 1083–1113. doi: 10.1177/02783649241227245
50. Benettin G, Galgani L, Giorgilli A, et al. Lyapunov characteristic exponents for smooth dynamical systems and for Hamiltonian systems; a method for computing all of them. Meccanica. 1980; 15(1): 9–20.
51. Bar-Shalom Y, Li XR, Kirubarajan T. Estimation with Applications to Tracking and Navigation. Wiley; 2001.

Appendix A. Notation

Symbol	**Meaning**	**Space**
$x, \hat{x}$	true and estimated navigation state	$\mathbb{R}^{13}$
δx	error state, minimal parameterisation	$\mathbb{R}^{15}$
p, v	position and velocity in the navigation frame	$\mathbb{R}^3$
$q, R(q)$	attitude quaternion and rotation matrix	$\mathbb{S}^3, SO(3)$
b_a, b_g	accelerometer and gyroscope biases	$\mathbb{R}^3$
$f(\cdot), G(\cdot)$	drift and diffusion of the plant SDE	—
$\beta(t), Q_c$	Wiener process and its spectral density	—
h_i, H_i	output map of sensor i and its Jacobian	—
$R_i, \tilde{R}$	nominal and weighted measurement covariance	$\succ 0$
$P(t)$	error covariance, solution of Equation (8)	$\succ 0$
$K(t)$	Kalman–Bucy gain	—
$W(t), w_i$	weighting matrix and sensor weight	$[\underline{w}, \bar{w}]^m$
τ_w	weighting law time constant	s
ν_i, S_i	innovation and innovation covariance of sensor i	—
$\underline{w}, \bar{w}$	weight saturation bounds	$(0, 1]$
α_1, α_2, T	uniform observability constants	—
$\underline{p}, \bar{p}$	Riccati solution bounds, Lemma 1	—
α, β	convergence rate and noise offset, Theorem 1	—
$\varepsilon_{\mathrm{d}}, \bar{\varepsilon}$	network output error and its bound	—
$\gamma_{\mathrm{d}}, \gamma_c$	ISS gains of estimator and controller	—
$V(\delta x, t)$	stochastic Lyapunov function	$\mathbb{R}_{\geq 0}$
$\mathcal{L}$	Itô infinitesimal generator	—
$A_{\mathrm{cl}}, \lambda_i$	closed-loop error matrix and Lyapunov exponents	—
C_i, T_i, R_i	execution time, period and response time of task i	ms