
Article

Distributed Counter-UAV Early Warning: Acoustic–Visual Information Consensus and Fuzzy–Bayesian Threat Assessment

Shang-En Tsai *, Chia-Han Hsieh, Wei-Cheng Sun and Sin-Dao Shen

Department of Computer Science and Information Engineering, Chang Jung Christian University,
Tainan City 711301, Taiwan

* Correspondence: sean@mail.cjcu.edu.tw; Tel.: +886-6-2785123 (ext. 6154)

Highlights

What are the main findings?

- This study proposes a distributed counter-UAV early-warning framework integrating acoustic–visual tracking, MTIC, and Fuzzy–Bayesian threat assessment.
- The system achieves near-centralized tracking accuracy with about 97% lower payload bandwidth and remains stable under latency, packet loss, node disconnection, and association mismatch.

What are the implications of the main findings?

- Counter-UAV early warning and response-decision support can be decentralized without depending on a single fusion center.
- Resilient multi-node UAV-defense networks can maintain interpretable threat assessment and real-time operation under degraded wireless conditions.

Abstract

This paper presents a distributed counter-UAV early-warning and response-decision support framework for low-altitude UAV defense. Acoustic–visual edge nodes generate local state packets and exchange compact information-filter parameters through Multi-Target Information Consensus (MTIC), avoiding centralized fusion and reducing payload bandwidth. The MTIC naïvety-handling mechanism is extended to heterogeneous acoustic–visual sensing, improving robustness to partial observations, packet loss, and node disconnection. A Fuzzy–Bayesian threat-assessment layer converts fused distance, velocity, and heading cues into interpretable response recommendations with calibrated confidence. Implemented on ROS 2/Fast DDS with tiered QoS, software-assisted IEEE 1588 synchronization, and Preempt-RT scheduling, the framework achieves within about 5% of centralized accuracy while reducing payload bandwidth by up to about 97% relative to the main centralized baseline. Simulation and hardware-in-the-loop tests on three- and five-node mesh topologies show software-assisted sub-millisecond synchronization (200–500 μ s offset), bounded latency, gradual AUC degradation under association mismatch, and end-to-end feasibility under controlled packet loss. Overall, the system provides a resilient, deployment-oriented architecture for distributed C-UAV early warning.

Academic Editor: Xiwang Dong

Received: 31 May 2026

Revised: 30 July 2026

Accepted: 4 August 2026

Published: 5 August 2026

Copyright: © 2026 by the authors.
Licensee MDPI, Basel, Switzerland.
This article is an open access article
distributed under the terms and
conditions of the [Creative Commons
Attribution \(CC BY\) license](https://creativecommons.org/licenses/by/4.0/).

Keywords: counter-UAV; distributed sensor fusion; acoustic–visual tracking; information consensus; fuzzy Bayesian decision; wireless mesh network; edge computing; ROS 2

1. Introduction

1.1. The Case for Distributed Defense

The proliferation and potential misuse of first-person-view (FPV) drones require distributed sensor networks capable of detecting, tracking, and supporting timely response to low-altitude, low-radar-cross-section, highly maneuverable targets [1–3]. A companion manuscript [4], currently under review, addressed the single-node perception-to-estimation challenge, proposing a deterministic safety-island kernel with unified measurement replay that produces high-quality local 3D state estimates from asynchronous acoustic and visual measurements. This architectural separation is also consistent with recent edge-UAV evidence showing that a Jetson-class perception branch paired with an MCU safety island can preserve bounded-latency state continuity under delay, jitter, and dropout by rejecting stale detections and replaying time-stamped measurements through a deterministic estimator [5,6]. The present paper addresses the multi-node challenge: how to fuse these local estimates into a shared global trajectory estimate and translate it into actionable response-decision support—without relying on a centralized fusion center.

Centralized architectures present three structural vulnerabilities. Single point of failure: the fusion center is a high-value node for wireless interference or physical disruption; its loss can disrupt the entire sensor network. Bandwidth bottleneck: continuously streaming high-frequency visual bounding boxes (≥ 60 Hz per node), acoustic spectral features, and state vectors to a central node can saturate the limited capacity of wireless mesh networks, especially under multi-hop routing. Latency amplification: the round trip from edge node to center and back adds non-trivial delay to the perception–decision loop, reducing the available response timeline against fast-approaching FPV targets.

1.2. From Local Tracks to Global Decisions

Each edge node, equipped with the safety-island kernel [4], outputs a local state estimate packet $(\hat{x}_n, P_n, \mu_n, \hat{x}(tn + 0.3), tn)$ at 200–400 Hz. However, each local estimate is limited by the single node’s sensor perspective: buildings may occlude the line of sight, wind gusts may degrade acoustic SNR, and geometric dilution of precision may inflate the position uncertainty. Fusing N_c such local estimates into a global optimum requires a mechanism that (a) operates without a central coordinator; (b) tolerates node disconnection and packet loss; (c) correctly handles “naïve” nodes that lack observations of a particular target; and (d) produces a fused trajectory of sufficient quality to support downstream response decisions. The challenge of fusing delayed or out-of-sequence measurements (OOSMs) in distributed networks has been extensively studied for delay scales on the order of hundreds of milliseconds [7,8].

Beyond trajectory fusion, the system must translate the kinematic state into a binary response-prioritization output (threat or non-threat) with quantified confidence, distinguishing potentially hostile FPV UAVs from benign objects such as friendly UAVs or birds. This output must be robust to single-node false alarms and adaptive to time-varying environmental conditions.

As an operational example, consider four edge nodes deployed around a 100 m core-defense perimeter protecting a small urban facility. A low-altitude FPV UAV may be

visible to only one or two cameras because of building occlusion, while acoustic direction-of-arrival estimates may still be available from additional nodes but with larger delay and uncertainty. In the proposed architecture, each node emits a replay-consistent local state packet, exchanges only compact information-filter parameters with its neighbors, reconstructs a shared trajectory through MTIC, and then applies the Fuzzy–Bayesian layer to decide whether a response alert should be raised for operator review. This example illustrates why the paper focuses on multi-node consensus and calibrated response-decision support rather than on the upstream object-detection module alone.

The scope and deployment assumptions are as follows. The present manuscript is intentionally scoped to the distributed fusion, wireless-mesh communication, and response-decision support layers. It does not claim to solve all aspects of counter-UAV deployment, such as full outdoor flight-test certification, long-range radar integration, actuator selection, or autonomous interdiction. The validation combines simulation with real-node and hardware-in-the-loop mesh experiments on three-node and five-node Jetson Orin Nano deployments. Therefore, the results should be interpreted as laboratory-scale evidence for the distributed consensus and decision pipeline under measured wireless impairments, with full outdoor flight testing and larger protected-area trials left as deployment-stage validation. The system-level architecture and layer partition are further summarized in Supplementary Figure S1 and Supplementary Table S1.

1.3. Contributions

- MTIC information consensus for C-UAV: The Multi-Target Information Consensus algorithm is applied to the counter-UAV domain, operating in the information-filter dual space ($J = P^{-1}$) and exchanging only compact statistical parameters. Average-consensus iterations converge under standard connected-graph and admissible step-size conditions. Building on Kamal et al.’s [9] zero-information injection principle—whereby a fully non-observing node contributes zero observation-information gain so that its participation does not distort the global estimate—this work deepens the treatment to “partial naïvety” states unique to heterogeneous acoustic–visual networks. In the purely visual sensor networks studied by Kamal et al., naïvety is binary: a node either detects a bounding box or it does not. In the present acoustic–visual setting, a node may instead hold a valid acoustic direction-of-arrival estimate while lacking a visual bounding box, yielding non-zero but modality-asymmetric observation information. Maintaining consensus stability under such partial, multi-modal observability—where acoustic and visual information gains arrive with order-of-magnitude latency differences—provides a practical extension of MTIC to heterogeneous acoustic–visual networks, where modality-asymmetric observations and latency differences must be handled without destabilizing consensus.
- Fuzzy multi-attribute threat quantification: Piecewise nonlinear membership functions map distance, velocity, and heading angle to $[0, 1]$ threat indices with explicit operational semantics: a 100 m core-defense perimeter with quadratic decay, and a cosine-constrained velocity–heading joint function that zeroes threat for receding targets.
- Distributed Bayesian group decision: A conjugate-prior Bayesian framework accumulates threat evidence across nodes and over time. Dynamic likelihood weighting adapts to environmental conditions (wind $\rightarrow$ down-weight acoustic; darkness $\rightarrow$ down-weight visual). A response alert is issued when the lower bound of the 95% credible interval exceeds 0.9 and the fused position covariance satisfies $\text{tr}(P_{\text{pos}}) < 1 \text{ m}^2$.
- Tiered DDS QoS and PTP time synchronization: Differentiated Quality-of-Service policies for acoustic alerts (Reliable), visual features (Best-Effort), and consensus

matrices (Reliable) are designed for lossy wireless mesh networks. IEEE 1588 PTP with GPS PPS achieves software-assisted sub-millisecond cross-node synchronization (200–500 μ s offset under CSMA/CA contention) under Preempt-RT scheduling.

- End-to-end latency budget: The complete delay chain from acoustic first detection to response alert broadcast is decomposed and quantified, indicating that the system supports the response loop well within the response timeline for FPV targets.

The novelty boundary is defined as follows. The perception algorithms, deterministic replay kernel, information-filter representation, fuzzy membership functions, and Bayesian updating are not individually claimed as entirely new mathematical primitives. The technical contribution of this manuscript is their system-level and algorithmic integration for heterogeneous acoustic–visual counter-UAV networks: (i) adapting MTIC from visual-only multi-target tracking to delay-compensated acoustic–visual local-state packets; (ii) defining partial-naïvety handling for modality-asymmetric observation gains; (iii) quantifying the bandwidth–accuracy–latency trade-off on lossy wireless mesh links; and (iv) coupling distributed trajectory consensus to an interpretable response-decision support layer with calibrated confidence.

2. Related Work

2.1. Distributed Sensor Fusion and Consensus Algorithms

Distributed estimation replaces the centralized fusion center with peer-to-peer information exchange. The information filter—the algebraic dual of the Kalman filter—represents the state in terms of the information matrix $J = P^{-1}$ and information state vector $\hat{y} = J\hat{x}$. In this representation, fusion of independent observations reduces to matrix addition, a property that naturally supports distributed architectures. Average-consensus protocols, rooted in spectral graph theory [10,11], guarantee that iterative local averaging converges to the global mean on any connected graph, with convergence rate governed by the algebraic connectivity (Fiedler value) [12]. Alternative distributed estimation paradigms include covariance intersection [13] and Kullback–Leibler consensus [14], each offering different trade-offs between communication cost and estimation accuracy. Kamal et al. [9] proposed the Multi-Target Information Consensus (MTIC) algorithm for distributed multi-target tracking in visual sensor networks, demonstrating near-centralized accuracy with minimal inter-node communication.

The first research gap is that MTIC was originally validated in visual-only networks with relatively homogeneous latency. Its application to heterogeneous acoustic–visual C-UAV networks—where acoustic measurements carry order-of-magnitude-larger delays and wireless mesh links suffer packet loss—has not been studied. The present work addresses this gap by building MTIC on top of the delay-compensated safety-island outputs [4].

2.2. Multi-Target Data Association

While cross-node target identity matching remains a critical prerequisite for consensus-based fusion—addressed by methods ranging from global nearest-neighbor (GNN) assignment to joint probabilistic data association (JPDA) and multi-hypothesis tracking (MHT) [8,15]—this paper assumes pre-established track-to-track association via an upstream spatial-gating protocol, isolating the fusion and decision layers downstream and focusing on overcoming the latency and bandwidth bottlenecks inherent in decentralized information consensus.

The assumption of upstream association is therefore explicit rather than hidden. A lightweight spatial-gating layer is suitable when the number of targets is small and the

tracks are well separated, whereas JPDA or MHT provides greater robustness in dense swarms at higher computational cost. This trade-off is evaluated later through mismatch injection and a JPDA comparison, so association uncertainty is treated as a quantified deployment risk rather than as an unexamined prerequisite.

2.3. Threat Assessment for Low-Altitude Small Targets

Threat assessment for low-altitude, slow-speed, small (LSS) targets has been approached via fuzzy multi-attribute decision making (MADM) [16,17], which maps continuous kinematic features to threat indices through membership functions, and via Bayesian inference frameworks [18,19] that accumulate probabilistic evidence over time. Distributed Bayesian group decision-making has also been studied in robot swarms [20], where agents reach consensus on an environmental state through local likelihood exchange.

The second research gap is that existing C-UAV threat assessment is predominantly centralized. How to generate consistent threat assessments at every node independently—while tolerating communication unreliability and single-node false alarms and maintaining interpretability of the decision criteria—is the specific challenge addressed here.

2.4. Real-Time Communication and Time Synchronization

ROS 2 with its DDS middleware provides a mature framework for distributed robotic systems. Quality-of-Service (QoS) policies (Reliability, History, Durability) control the trade-off between delivery guarantees and latency. IEEE 1588 Precision Time Protocol (PTP) [21] achieves sub-microsecond synchronization over Ethernet; combined with GPS pulse-per-second (PPS) signals, it provides an absolute time reference. The Preempt-RT Linux kernel patch reduces scheduling jitter to microsecond levels.

The communication gap is that most ROS 2 real-time studies target wired Ethernet. Wireless mesh environments introduce packet loss, variable latency, and IP fragmentation that degrade both DDS reliability and PTP accuracy. Tiered QoS tuning for heterogeneous C-UAV data flows over wireless mesh is underexplored. Synthesis: existing work advances individual layers—sensing, fusion, decision, or middleware—in isolation. What remains missing is an end-to-end distributed C-UAV architecture that jointly addresses multi-target information consensus, interpretable threat assessment, and deployment-feasible communication under degraded wireless conditions. This paper addresses that gap.

3. Materials and Methods

3.1. System Context and Upstream Local-State Interface

This paper builds on the perception and estimation modules developed in the companion manuscript [4], currently under review, which processes asynchronous acoustic and visual signals to produce delay-compensated local state estimates. The measurement-replay mechanism in [4] eliminates asynchronous timing bias before fusion—a prerequisite for MTIC to produce meaningful global estimates. The sole interface between [4] and the present paper is the state packet independently emitted by each node at every estimation tick:

$$Packet_i(t_n) = (\hat{x}_n, P_n, \mu_n, \hat{x}(t_n + 0.3), t_n) \quad (1)$$

where $\hat{x}_n \in \mathbb{R}^9$ is the 3D state estimate (position, velocity, acceleration), $P_n \in \mathbb{R}^{9 \times 9}$ is the error covariance, μ_n is the IMM mode-probability vector, $\hat{x}(t_n + 0.3)$ is the 0.3 s prediction, and t_n is the tick timestamp. Crucially, the measurement-replay mechanism

in [4] ensures that $\hat{x}_n$ and P_n are free of asynchronous timing bias—a prerequisite for MTIC to produce meaningful global estimates.

Deterministic measurement replay. To make the timing-alignment assumption self-contained, each edge node implements a buffer-rollback-update-propagate procedure before emitting the packet in Equation (1). Let the local estimator keep a time-ordered buffer $\mathcal{B}_n$ containing predicted and posterior states around the current tick t_n . When an acoustic or visual measurement z_m with sensor timestamp $t_m < t_n$ arrives out of sequence, the estimator first rolls back to the buffered prediction at t_m , applies the delayed measurement update, and then re-propagates the corrected posterior to t_n .

Acoustic propagation-delay compensation. For acoustic measurements, the sensor timestamp used by the replay buffer is not treated as the packet-arrival time at the middleware layer. Let $c \approx 343$ m/s denote the speed of sound and let d be the estimated source-to-array range. The acoustic propagation time is $t_{prop} = d/c$; for a target near the 100 m core-defense perimeter this corresponds to approximately 0.29 s, which is much larger than the visual propagation delay. Accordingly, the acoustic measurement is assigned a propagation-compensated physical timestamp $t_m = t_{arrival} - t_{prop}$ before entering the replay buffer. The buffer-rollback-update-propagate equations below then apply the measurement update at t_m and propagate the corrected posterior forward to the current tick t_n . This links the deterministic replay mechanism directly to the physical sound-propagation delay and prevents delayed acoustic evidence from being fused as if it were synchronous visual evidence.

Rollback and delayed update at t_m :

$$(\hat{x}_{m|m-1}, P_{m|m-1}) \leftarrow B_n(t_m), \quad (2)$$

$$K_m = P_{m|m-1} H_m^T (H_m P_{m|m-1} H_m^T + R_m)^{-1}, \quad (3)$$

$$\hat{x}_{m|m} = \hat{x}_{m|m-1} + K_m [z_m - h_m(\hat{x}_{m|m-1})] \quad (4)$$

$$P_{m|m} = (I - K_m H_m) P_{m|m-1}. \quad (5)$$

Forward replay from t_m to t_n :

$$\hat{x}_{n|m} = F^{n-m} \hat{x}_{m|m}, \quad (6)$$

$$P_{n|m} = F^{n-m} P_{m|m} (F^{n-m})^T + Q_{n-m}. \quad (7)$$

The state packet in Equation (1) is then emitted using $\hat{x}_n = \hat{x}_{n|m}$ and $P_n = P_{n|m}$. Thus, late acoustic or visual measurements are incorporated at their physical timestamps rather than at their arrival times. This replay-consistent packet construction removes asynchronous timing bias before MTIC fusion and allows the theoretical basis of the present paper to stand at the packet-interface level without relying on unpublished implementation details.

The system comprises N_C edge nodes connected via an 802.11s wireless mesh network modeled as a connected graph. Consensus is maintained on any connected subgraph and recovers within one cycle upon node reconnection. Each node simultaneously tracks J targets; a naïveté flag indicates whether a node lacks a current observation of a given target, in which case it contributes zero observation information to the consensus while still functioning as a relay (Section 3.2.3).

3.2. Multi-Target Information Consensus (MTIC)

3.2.1. Information-Filter Dual Space

The information filter [22,23] represents the Gaussian state estimate in its natural (canonical) parameterization. Combined with the interacting multiple model (IMM) framework [24], this representation supports heterogeneous motion-mode tracking:

$$J = P^{-1} \text{ (information matrix)} \quad (8)$$

$$\hat{y} = J\hat{x} \text{ (information state vector)} \quad (9)$$

This representation offers three advantages for distributed fusion. First, fusion of independent observations is additive: the fused information matrix is the sum of individual information matrices, and the fused information state is the sum of individual information states. This contrasts with the covariance form, where fusion requires matrix inversion. Second, the absence of observation is naturally represented as zero information contribution ($J_{obs} = 0, \hat{y}_{obs} = 0$), requiring no special-case handling. Third, the parameters to be communicated are the same data types (matrices and vectors) regardless of the number of measurements, enabling fixed-format network packets.

From the safety-island output [4], node i extracts the local prior for target j :

$$J_i^{j-} = (P_i^j)^{-1}, \quad \hat{y}_i^j = J_i^{j-} \hat{x}_i^{j-} \quad (10)$$

At the beginning of each consensus cycle, these local priors are synchronized across nodes by using the fused posterior from the previous cycle as a common prior propagated to the current timestamp. Hence, during the consensus reconstruction stage, $J_i^{j-} = J_j^-$ and $\hat{y}_i^{j-} = \hat{y}_j^-$ for all nodes. If node i has a valid local measurement update for target j (i.e., the safety-island kernel performed at least one measurement update since the last consensus cycle), the observation information gain is defined as

$$G_i^j = J_i^{j+} - J_i^{j-}, \quad U_i^j = \hat{y}_i^{j+} - \hat{y}_i^{j-} \quad (11)$$

where the superscript $+$ denotes the posterior after the local measurement update and $-$ denotes the prior (prediction only).

3.2.2. Average Consensus Iteration Protocol

For any variable a_i maintained at node i (which may be a scalar, vector, or matrix), the average-consensus update at communication round k is

$$a_i(k) = a_i(k-1) + \varepsilon \sum_{m \in N_i} (a_m(k-1) - a_i(k-1)) \quad (12)$$

where ε is a step-size parameter satisfying $\varepsilon \in (0, 1/d_{max})$ with d_{max} being the maximum node degree in the graph. Under the network-topology assumptions stated in Section 3.1, the fundamental convergence theorem guarantees

$$\lim_{k \rightarrow \infty} a_i(k) = \frac{1}{N_C} \sum_{m=1}^{N_C} a_m(0), \forall i \quad (13)$$

That is, all nodes converge to the global arithmetic mean of the initial values, regardless of the network topology (provided it is connected). The convergence rate is determined by the spectral gap of the graph Laplacian: denser and more regular graphs converge faster [10]. For typical deployment topologies (grid, random geometric graph with sufficient density), 3–5 consensus rounds suffice to achieve >95% convergence to the global mean. Each round requires node i to transmit a_i to its neighbors N_i and receive a_i' from each neighbor.

For a 9-dimensional state, the information matrix contributes 45 unique elements after exploiting symmetry, and the information state vector contributes 9 elements, for a total of 54 floating-point values per target per round. Assuming 32-bit floating-point

representation, this corresponds to $54 \times 4 = 216$ bytes per target per round. With $K = 5$ consensus rounds, the communication cost becomes $216 \times 5 = 1080$ bytes ≈ 1.08 KB per target per consensus cycle. For $J = 3$ targets and an update rate of 10 Hz, the resulting one-way transmission load per node is $1.08 \times 3 \times 10 \approx 32.4$ KB/s. This bandwidth remains negligible compared with even a single compressed image stream. It should be noted that this 32.4 KB/s figure reflects payload-only traffic. In a real wireless mesh deployment, protocol overhead (DDS/RTPS headers, IP/UDP encapsulation, and reliability ACKs) typically adds 20–40% to the payload, yielding an offered load of approximately 39–45 KB/s per node. For a node with d neighbors, the per-link outgoing traffic is approximately $45/d$ KB/s. The total network traffic across all N_c nodes scales as $O(N_c \cdot d \cdot K)$, but because each node only communicates with its direct neighbors (not the entire network), the aggregate load grows linearly with the number of edges rather than quadratically with N_c . Even under worst-case assumptions ($d = 4$, 40% overhead, 10% retransmission rate), the total offered load per node remains below 60 KB/s—well within the capacity of 802.11s mesh links.

3.2.3. Global Posterior Reconstruction and Naïvety Handling

Following the standard MTIC reconstruction procedure [14], after K rounds of average consensus on the auxiliary variables, each node independently reconstructs the globally fused posterior. The fused information matrix equals the shared prior plus the sum of the observation gains contributed by all nodes, and the fused state vector is obtained from the corresponding information-state representation. This result is mathematically equivalent to centralized information-filter fusion, with approximation error that decreases as the number of consensus rounds increases. The explicit auxiliary-variable definitions and derivation are provided in the Supplementary Material.

In real deployments, some nodes may have no observation of a particular target due to building occlusion, sensor failure, or angular blind spots. When node i has no observation of target j ($\beta_{oi}^j = 1$), MTIC enforces

$$U_i^j = 0, \quad G_i^j = 0 \quad (14)$$

Hence, for a naïve node,

$$W_i^j = \frac{J_j^-}{N_c}, \quad \hat{v}_i^j = \frac{\hat{y}_j^-}{N_c} \quad (15)$$

In other words, the zero-information constraint established by Kamal et al. [14] for fully non-observing nodes—contributing the shared prior while injecting no observation information—elegantly prevents such nodes from distorting the fused posterior with spurious local evidence. The node continues to relay neighbors' messages, preserving graph connectivity without compromising estimation fidelity. This property is particularly important in urban environments where line-of-sight availability varies substantially across nodes. The Kamal et al. [14] formulation was, however, developed and validated for homogeneous visual sensor networks, in which naïvety is binary: a node either contributes a bounding-box measurement or it does not. Heterogeneous acoustic-visual counter-UAV networks introduce a qualitatively richer regime, termed here “partial naïvety”: a node may possess a valid acoustic direction-of-arrival (DOA) estimate yet lack a visual bounding box for the same target, yielding non-zero but modality-asymmetric observation-information gains. Because acoustic measurements carry order-of-magnitude-larger end-to-end latency than visual measurements, the per-modality information gains $G_{ij}^{acoustic}$ and G_{ij}^{visual} may be available at different phases within the same consensus cycle, threatening the synchronicity assumption that underpins round-wise convergence. Maintaining symmetric positive definiteness (SPD) and consensus stability under this asymmetric, multi-modal delay environment requires per-modality

information-gain decomposition and phase-aware aggregation within the zero-injection framework—an implementation-level extension of the MTIC framework for heterogeneous acoustic–visual sensing. A formal proof that the extended consensus update preserves SPD under exact arithmetic and converges on connected graphs is provided in the Supplementary Materials.

3.2.4. MTIC Algorithm Summary

The complete MTIC procedure per consensus cycle consists of eight steps: (1) receive local safety-island packets; (2) convert to information space; (3) compute observation gains (or enforce zero gain for naïve nodes); (4) construct consensus auxiliary variables; (5) iterate K rounds of neighbor-to-neighbor averaging; (6) reconstruct the global posterior; (7) stabilize the information matrix via forced symmetrization; and (8) pass the fused trajectory to the threat-assessment module.

Numerical stabilization. Under exact arithmetic, the consensus-averaged information matrix remains symmetric positive definite (SPD) whenever all local information matrices are SPD. In practice, finite-precision accumulation on resource-constrained 32-bit edge hardware may introduce symmetry drift or ill-conditioning, especially after repeated consensus and inversion operations. To support safe runtime inversion, the implementation applies forced symmetrization followed by optional eigenvalue flooring after each consensus cycle. This correction is a practical numerical safeguard rather than a limitation of the MTIC algebra itself. A future square-root information-consensus formulation would further improve numerical robustness. The eigenvalue floor is applied conservatively as a lower-bound safeguard against numerical singularity, not as an artificial confidence amplifier; it is not allowed to reduce the reported covariance trace below the response-alert threshold or to create a response alert by itself. The overall MTIC workflow is illustrated in Figure 1.

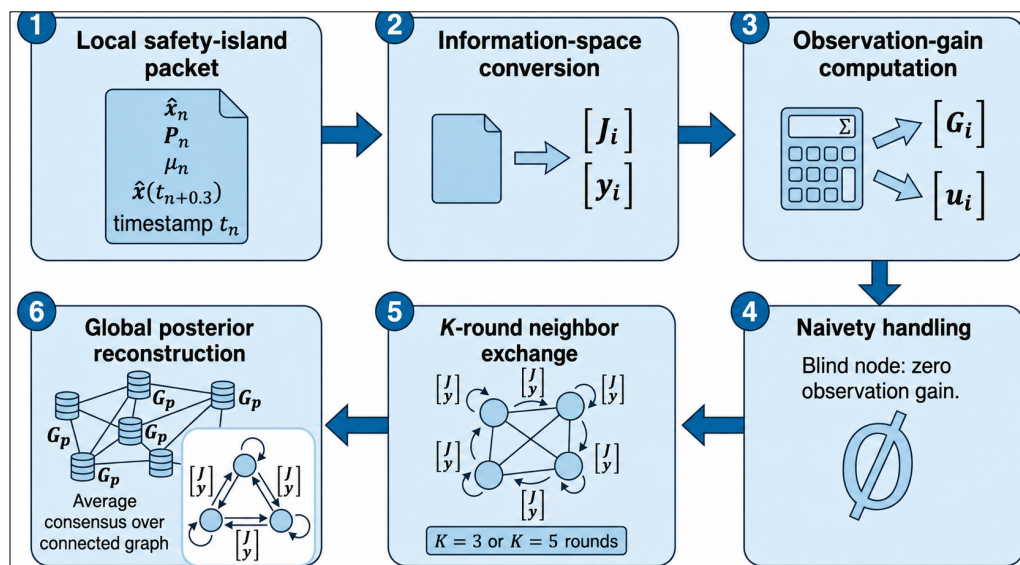


Figure 1. MTIC workflow in the information-filter dual space. Local state estimates are converted into information variables, naïve nodes inject zero observation information, neighbor-to-neighbor average consensus is iterated for K rounds, and each node independently reconstructs the globally fused posterior target state.

3.2.5. Communication Efficiency Analysis

The bandwidth advantage of MTIC over centralized raw-data fusion is substantial, as quantified in Table 1 for a 9-dimensional state.

Table 1. Bandwidth comparison. MTIC reduces network traffic by approximately one to two orders of magnitude.

Architecture	per Target	per Cycle Payload Traffic (3 Targets, 10 Hz)	Ratio
Centralized (raw bounding boxes + audio features)	~10–60 KB	300–1800 KB/s	1×
MTIC (K = 5 consensus rounds)	~1.08 KB	~32 KB/s	~0.02×–0.1×

3.3. Fuzzy–Bayesian Threat Assessment

3.3.1. Fuzzy Multi-Attribute Threat Quantification

1. Attribute selection

Three kinematic attributes are extracted from the MTIC global trajectory for target j : distance $d = \|ptarget - passet\|$, speed $v = \|vtarget\|$, and relative heading angle $\theta = \text{angle}(vtarget, passet - ptarget)$. These three quantities capture the fundamental dimensions of tactical threat: how close the target is (reaction time), how fast it is moving (response difficulty), and whether it is heading toward the defended asset (approach intent).

2. Distance threat membership function $U(d)$

The core-defense perimeter is set at 100 m based on operational analysis: once a target breaches this boundary, it has entered the response range of short-range counter-UAV systems and poses an elevated asset-protection risk. The piecewise nonlinear membership function, grounded in fuzzy set theory [25], is as follows:

The sensor-physics rationale is as follows. The 100 m perimeter is also consistent with the passive acoustic–visual sensing regime evaluated in this work. At longer ranges, stereo-triangulation depth uncertainty increases rapidly with distance, while acoustic signal-to-noise ratio and propeller-harmonic distinctiveness decrease under wind and urban reflections. The perimeter therefore should not be interpreted as a universal counter-UAV range, but as a conservative operating boundary within which the evaluated acoustic–visual geometry, local-state covariance, and Fuzzy–Bayesian response logic remain interpretable and auditable.

$$U(d) = \begin{cases} 1, & \text{if } d \leq 100; \\ \frac{1}{\left[1 + \left(\frac{d-100}{2}\right)^2 \times 10^{-4}\right]}, & \text{if } d > 100 \end{cases}. \quad (16)$$

For $d > 100$ m, the quadratic term in the denominator produces smooth nonlinear decay. At $d = 200$ m, $U \approx 0.80$; at $d = 500$ m, $U \approx 0.20$; at $d = 1000$ m, $U \approx 0.05$. The function is strictly bounded in $[0,1]$ and captures the physical intuition that threat escalates nonlinearly as range decreases.

3. Velocity–heading joint threat function $U(v, \theta)$

Speed alone does not determine threat; the target's heading relative to the defended asset is also critical. A high-speed target flying away from the asset poses limited immediate risk. The joint function incorporates this directional cue via a cosine constraint:

$$U(v, \theta) = \left\{ \frac{\cos(\theta) \times v}{40}, \text{if } -90^\circ \leq \theta \leq 90^\circ; 0, \text{otherwise} \right\}. \quad (17)$$

The reference speed 40 m/s (≈ 144 km/h) represents the upper end of typical FPV drone performance. The cosine term smoothly modulates threat: maximum when $\theta = 0^\circ$ (direct approach), diminishing as the heading deviates, and exactly zero for $\theta > 90^\circ$

(receding target). The function value may exceed 1.0 for targets faster than 40 m/s on a direct approach; in practice, it is clamped to [0,1] in the downstream fusion.

4. Composite threat index

The composite fuzzy threat index for target j is

$$T^j = w_d \cdot U(d^j) + w_v \cdot U(v^j, \theta^j) \quad (18)$$

where $w_d + w_v = 1$ are operational weights. For close-range perimeter defense, $w_d = 0.6$ and $w_v = 0.4$ emphasize proximity. For counter-swarm operations against distant but fast-approaching targets, w_v may be increased. These weights are configurable parameters, not learned, preserving operational interpretability.

3.3.2. Distributed Bayesian Confidence Update

1. Bayesian inference framework

The hidden state “target j is an active threat” is modeled as a Bernoulli random variable. The posterior threat probability given all observations $Z_1:t$ is updated via Bayes’ rule:

$$P(\text{Threat} \mid Z_{1:t}) \propto P(Z_t \mid \text{Threat}) \cdot P(\text{Threat} \mid Z_{1:t-1}). \quad (19)$$

A Beta conjugate prior $\text{Beta}(\alpha_0, \beta_0)$ is initialized with uniform parameters ($\alpha_0 = \beta_0 = 1$), representing no prior information. Each new MTIC-fused trajectory update or fuzzy threat index serves as observation evidence Z_t . The Beta posterior updates are

$$\alpha_t = \alpha_{t-1} + T^j, \quad \beta_t = \beta_{t-1} + (1 - T^j) \quad (20)$$

where $T^j \in [0, 1]$ is the fuzzy threat index from Equation (18). The posterior mean threat probability is $\alpha_t / (\alpha_t + \beta_t)$, and the 95% credible interval narrows as evidence accumulates.

2. Dynamic likelihood weighting

Environmental conditions affect the reliability of different sensing modalities. The system dynamically adjusts the effective weight of evidence based on the following:

Wind speed: an anemometer at each node reports ambient wind speed. Above a threshold (e.g., 15 m/s sustained), acoustic feature reliability degrades significantly. The system scales the acoustic-derived component of T^j by a wind attenuation factor $\gamma_{wind} \in (0, 1]$, reducing its contribution to the Beta update.

Illumination: a lux sensor at each node monitors ambient light. In low-light or backlight conditions, visual detection confidence drops. The visual-derived component of T^j is scaled by an illumination factor $\gamma_{lux} \in (0, 1]$.

The independence of sensor errors across nodes is a critical assumption: even if one node’s acoustic measurement is corrupted by a local wind gust, neighboring nodes may have clean acoustic data. Across N_c nodes, the probability of all nodes simultaneously producing a false alarm decreases exponentially, providing substantial robustness against single-node false positives.

3. Response-alert trigger logic

The system continuously monitors the Beta posterior for each tracked target. A response alert is issued when all of the following conditions are simultaneously satisfied:

Condition 1 (threat confidence): the lower bound of the 95% credible interval of the posterior threat probability exceeds 0.9. This ensures that the system is highly confident in the threat classification, not merely that the point estimate is high.

Condition 2 (trajectory quality): the trace of the MTIC-fused position covariance satisfies $\text{tr}(P_{pos}) < 1 \text{ m}^2$ (i.e., the aggregate position-variance measure is below the

prescribed response threshold). This prevents response alerts based on poorly localized targets.

Condition 3 (multi-node corroboration): at least two nodes independently report non-zero observation gain for the target ($\beta_{oi}^j = 0$ for ≥ 2 nodes). This guards against issuing a response alert based on a single node's potentially erroneous detection.

When all three conditions are met, the node broadcasts a response-support packet via the DDS priority channel:

$$target_id, x_{global}, v_{global}, \hat{x}(t + 0.3), T^j, CI_{95\%}, timestamp). \quad (21)$$

Downstream response systems or operator consoles receive this packet and select the appropriate operator-approved response. The system supports operator-supervised and human-in-the-loop modes, selectable via operational configuration. The complete Fuzzy-Bayesian threat-assessment framework is summarized in Figure 2.

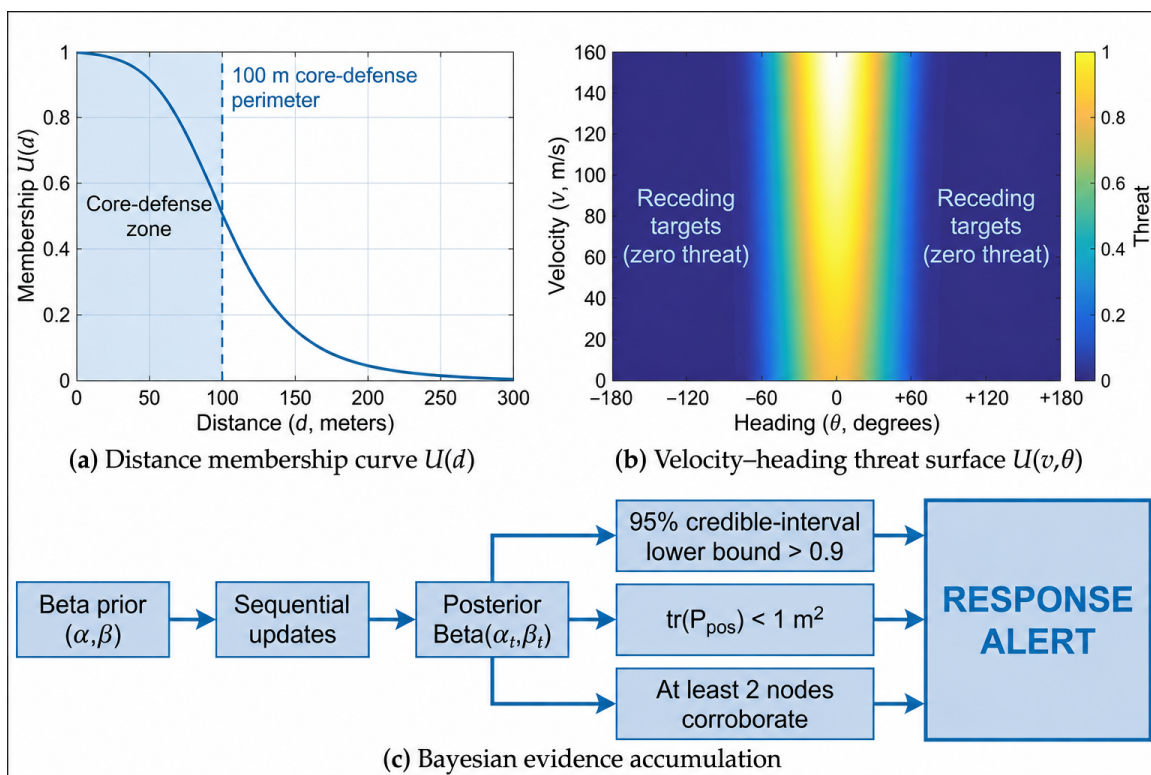


Figure 2. Fuzzy-Bayesian threat assessment framework. (a) Distance membership function $U(d)$ with 100 m core-defense perimeter. (b) Velocity-heading joint threat surface $U(v, \theta)$ with cosine constraint. (c) Bayesian evidence accumulation and three-condition response-alert trigger.

3.4. Edge Computing and Real-Time Communication Infrastructure

3.4.1. Edge-Node Hardware and Runtime Modules

Each edge node is built around an NVIDIA Jetson Orin Nano running JetPack 6 (L4T R36.3), providing an Ampere-architecture GPU with Tensor Cores for neural-network inference and an ARM Cortex-A78AE CPU cluster for control and communication. Each node simultaneously executes (a) the perception front-end [4] (YOLO [26] inference, stereo/DOA processing); (b) the safety-island kernel [4] (IMM-UKF/MS-C-EKF with replay); (c) the MTIC consensus module; and (d) the Fuzzy-Bayesian threat assessment. CPU cores are partitioned to isolate real-time critical tasks [27] (safety island, I2S capture) from soft real-time tasks (YOLO inference, MTIC consensus). This heterogeneous role assignment is consistent with recent embedded-UAV studies showing that Jetson-class

platforms are suitable for perception-rich edge AI, whereas STM32-class microcontrollers remain effective for deterministic, low-power onboard functions under tight resource constraints [5,6].

3.4.2. Acoustic Front-End

Each node's acoustic front-end employs INMP441 MEMS microphones (61 dB SNR, 60 Hz–15 kHz) interfaced via I2S; complete pin-mapping and device-tree configuration are provided in the open-source repository README. Audio is captured at 48 kHz with 1–2% CPU overhead. Outdoor windscreens extend acoustic detection range by 31–131% and improve broadband SNR by 1.8–4.4 dB under strong wind [28]. The MUSIC direction-of-arrival algorithm [29] and stereo triangulation methods [30] are detailed in the companion manuscript [4]. Recent multimodal drone-perception studies further support the use of fused acoustical and optical sensing for robust small-UAV detection and tracking in outdoor environments [31].

The audio-preprocessing pipeline is as follows. The implementation uses timestamped ALSA frames as the input to the local safety-island pipeline. Each frame is band-limited to the UAV-propeller harmonic region, windowed for spectral estimation, and passed to the direction-of-arrival module before being packaged as a delayed measurement with its original sensor timestamp. This design separates acoustic feature extraction from MTIC: the distributed layer receives only the replay-consistent local state packet in Equation (1), while the raw audio stream remains local to the edge node. Full pin mapping, capture settings, and middleware configuration files are provided in the Supplementary Material and the public repository. The detailed INMP441-to-Jetson Orin Nano I2S pin mapping is provided in Supplementary Table S2.

3.4.3. Middleware Communication and Tiered QoS Design

The distributed system is built on ROS 2 [32,33] with eProsima Fast DDS as the middleware implementation. Recent studies [34,35] have characterized the performance challenges of DDS over wireless links. In lossy wireless mesh environments, default DDS configurations suffer from IP fragmentation, congestion-buffer overflow, and ineffective retransmission. The system implements tiered QoS policies matched to the characteristics of each data flow, as summarized in Table 2.

The implementation and configuration are as follows. The reported experiments use the JetPack 6/L4T R36.3 software stack on Jetson Orin Nano nodes, Fast DDS as the ROS 2 middleware implementation, an 802.11s mesh backhaul, software-assisted IEEE 1588 time synchronization, and Preempt-RT scheduling for the timing-critical safety-island and I2S-capture threads. The repository provides the Fast DDS XML and QoS profile files used to reproduce the middleware settings, including the reliability and queue-depth choices summarized in Table 2.

The QoS stress rationale is as follows. The QoS split is deliberately cross-layer: high-rate visual bounding boxes use Best Effort to avoid head-of-line blocking and retransmission storms, whereas compact acoustic alerts and MTIC consensus matrices use Reliable delivery because their packet sizes are small and missing matrix exchanges can destabilize consensus reconstruction. The controlled packet-loss results in Section 4.7 quantify the point at which this Reliable delivery policy begins to trade robustness for delay under severe multi-hop loss.

Table 2. Tiered QoS configuration for heterogeneous C-UAV data flows.

Data Flow	Reliability	History	Durability	Rationale
Acoustic DOA/decision alerts	Reliable	Keep Last (10)	Transient Local	Must-deliver: small packets, high criticality
Visual bounding boxes	Best Effort	Keep Last (1)	Volatile	High rate (≥ 60 Hz); retransmission would cause congestion
MTIC consensus matrices	Reliable	Keep Last (5)	Volatile	Lossless matrix sync; missing data causes divergence

3.4.4. Precision Time Synchronization (PTP + GPS PPS)

Accurate cross-node time alignment is essential for MTIC, because the exchanged information matrices must correspond to the same physical time step. In the present implementation, synchronization over the 802.11s wireless mesh is conservatively treated as sub-millisecond, not microsecond-level, synchronization. This wording is intentional: on standard Jetson Orin Nano platforms, PTP hardware timestamping is not exposed on the default network interface (as confirmed by `ethtool -T` returning PTP Hardware Clock: none), and therefore the deployed wireless-mesh synchronization should be interpreted as software-assisted IEEE 1588 [21] synchronization with practical error in the 200–500 μ s range under CSMA/CA contention. At the maximum target speed of 40 m/s, a 500 μ s synchronization error corresponds to only 2 cm of spatial displacement, which is negligible relative to the meter-level covariance of the fused trajectory. Future upgrades to true microsecond-level synchronization would require either a PTP-capable external network interface with hardware timestamping or a wireless implementation that exposes NIC-level TSF-based timestamping; such hardware support is outside the scope of the current platform configuration.

For absolute time reference, each Grandmaster Clock is disciplined by a GPS pulse-per-second (PPS) signal, providing a UTC-traceable 1 Hz reference with nanosecond-level edge accuracy. The combination of PTP (for inter-node relative synchronization) and GPS PPS (for absolute time anchoring) ensures that all timestamps in the system—acoustic acquisition, visual frame capture, safety-island ticks, and MTIC consensus rounds—are globally consistent.

3.4.5. Real-Time Scheduling and Computational Isolation

All edge nodes run a Preempt-RT Linux kernel. Safety-island and I2S capture threads are pinned to isolated CPU cores under `SCHED_FIFO` scheduling (priority 95), suppressing scheduling jitter to its hardware minimum and guaranteeing trustworthy sensor timestamps before data enters the Kalman filter or MTIC consensus loop. The edge-node hardware–software stack and communication backbone are shown in Figure 3.

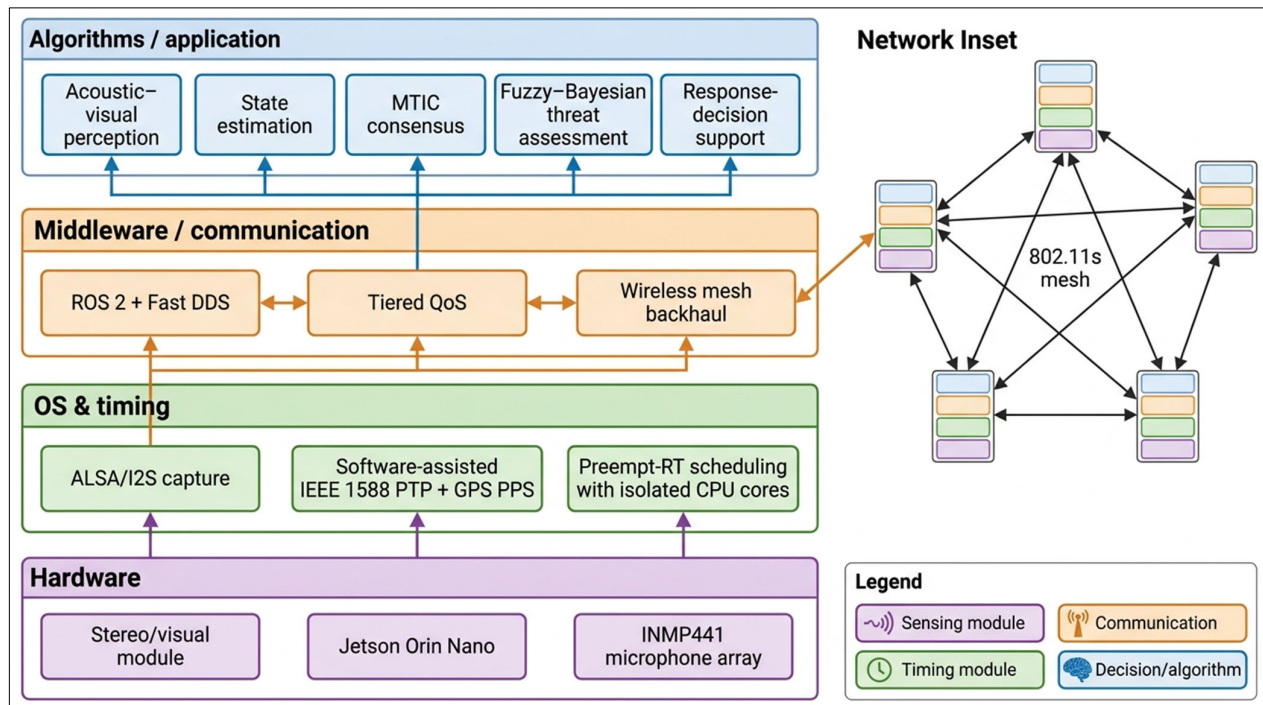


Figure 3. Edge node hardware–software stack and real-time communication backbone. Each Jetson-based node integrates acoustic and visual sensing, real-time Linux scheduling, ROS 2/Fast DDS messaging, and PTP-disciplined time synchronization, while the wireless mesh network supports reliable exchange of consensus and priority-decision messages.

4. Results

4.1. Real-Node Timing, Synchronization, and Consensus Validation

Preliminary hardware-in-the-loop testing on a three-node Jetson Orin Nano testbed confirms that single-hop wireless-mesh round-trip latency is 25–65 ms, software-PTP clock offset remains within 180–420 μ s, and nominal packet loss stays below 2%. At the maximum target speed of 40 m/s, the 500 μ s synchronization upper bound corresponds to only 2 cm of spatial displacement—negligible relative to the meter-level trajectory covariance. These measurements validate the real-time communication assumptions of Section 3.4. Detailed hardware configuration and timing-distribution plots are provided in the Supplementary Material.

The effect of synchronization error is evaluated as follows. The synchronization measurements are included in the reported timing budget rather than treated as an ideal assumption. Under degraded wireless conditions, packet loss and contention mainly affect MTIC through delayed consensus completion and increased DDS retransmission time; clock-offset error contributes a much smaller geometric component. Even the conservative 500 μ s offset upper bound corresponds to approximately 2 cm at 40 m/s, which is far below the meter-level tracking covariance used in the response-alert condition. Consequently, the degraded-network experiments in Sections 4.4 and 4.7 stress the dominant latency and packet-loss mechanisms while still accounting for measured software-PTP jitter.

Extending the HIL validation to real three-node and five-node 802.11s mesh topologies, Table 3 quantifies K -round consensus convergence against centralized fusion. In the three-node mesh, $RMSE_{pos}$ decreases from 1.55 m at $K = 1$ to 1.08 m at $K = 5$, corresponding to about 97% of centralized accuracy, while consensus-cycle completion time grows from 42 ms to 208 ms. In the five-node mesh, $K = 5$ reaches 1.12 m (about 95%

of centralized accuracy), but increases cycle-completion time to 280 ms, peak one-hop RTT to 72 ms, and packet-loss/PTP-jitter to 3.5%/410 μ s. These measurements expose a practical K-round contention trade-off in which marginal accuracy gains beyond $K = 3$ are increasingly offset by 802.11s MAC-layer delay and contention.

Table 3. Real-node consensus convergence and 802.11s mesh timing characteristics for three-node and five-node deployments.

Deployment Topology	Consensus Rounds (K)	$RMSE_{pos}$ (m)	Centralized-Accuracy Ratio	Consensus-Cycle Completion (ms)	Peak One-Hop RTT (ms)	Packet Loss/PTP Jitter
3-node mesh	K = 1	1.55	~65%	42	28	<1.0%/190 μ s
3-node mesh	K = 3	1.15	~92%	126	35	1.2%/210 μ s
3-node mesh	K = 5	1.08	~97%	208	42	1.8%/240 μ s
5-node mesh	K = 1	1.68	~58%	55	45	1.5%/280 μ s
5-node mesh	K = 3	1.25	~88%	165	58	2.4%/320 μ s
5-node mesh	K = 5	1.12	~95%	280	72	3.5%/410 μ s

4.2. Experimental Design

The multi-node evaluation extends the simulation framework [4] to a distributed topology. The simulator instantiates $N_c \in \{3, 5, 8, 12\}$ virtual edge nodes, each running a complete safety-island kernel [4] and the MTIC + threat-assessment stack proposed in this paper. A wireless mesh network simulator models packet loss (0–15%), variable link latency (5–50 ms per hop), and multi-hop routing. The evaluated multi-node scenarios are summarized in Table 4.

Table 4. Experimental scenarios and baseline/ablation configurations.

ID	Type	Configuration	Purpose/Description
Scenarios			
M1	Scenario	Single target	scenarios S1–S4 defined in [4], extended to a multi-node topology
M2	Scenario	Multi-target	2–5 FPV targets from different directions
M3	Scenario	Swarm stress test	8 targets in close formation; stresses data association
M4	Scenario	Mixed threat	3 hostile FPV + 2 friendly UAVs + 1 bird
Baselines/Ablations			
C1	Baseline	Centralized fusion (all data to center)	Accuracy upper bound reference
C2	Ablation	MTIC without naïvety handling	Naïvety mechanism necessity
C3	Ablation	MTIC with naïvety (complete)	Distributed vs. centralized accuracy
C4	Ablation	MTIC + fuzzy only (no Bayesian)	Bayesian evidence accumulation value
C5	Ablation	MTIC + Bayesian only (no fuzzy)	Fuzzy attribute design value
C6	External	Covariance Intersection (CI)	External distributed-fusion reference
Ours	Full	MTIC + Fuzzy–Bayesian pipeline	Complete proposed system

4.3. Experimental Scenarios and Baselines

Table 4 consolidates the four evaluation scenarios (M1–M4) and six baseline/ablation configurations (C1–C6, plus the full proposed system). Configuration C6 implements a Covariance Intersection (CI) baseline [12], providing an external distributed-fusion reference that does not require cross-correlation estimation.

The single-node contextual baseline is defined as follows. A single edge node running only the upstream safety-island estimator is not a distributed-fusion competitor but a contextual baseline for what is lost without network cooperation: it has no cross-node corroboration, cannot compensate for local occlusion, and cannot reduce single-node false alarms through independent evidence. The centralized baseline C1 represents the

accuracy upper bound, while C3 and C6 quantify the benefits of distributed fusion under realistic communication constraints. The distributed scenario layouts and target classes are illustrated in Figure 4.

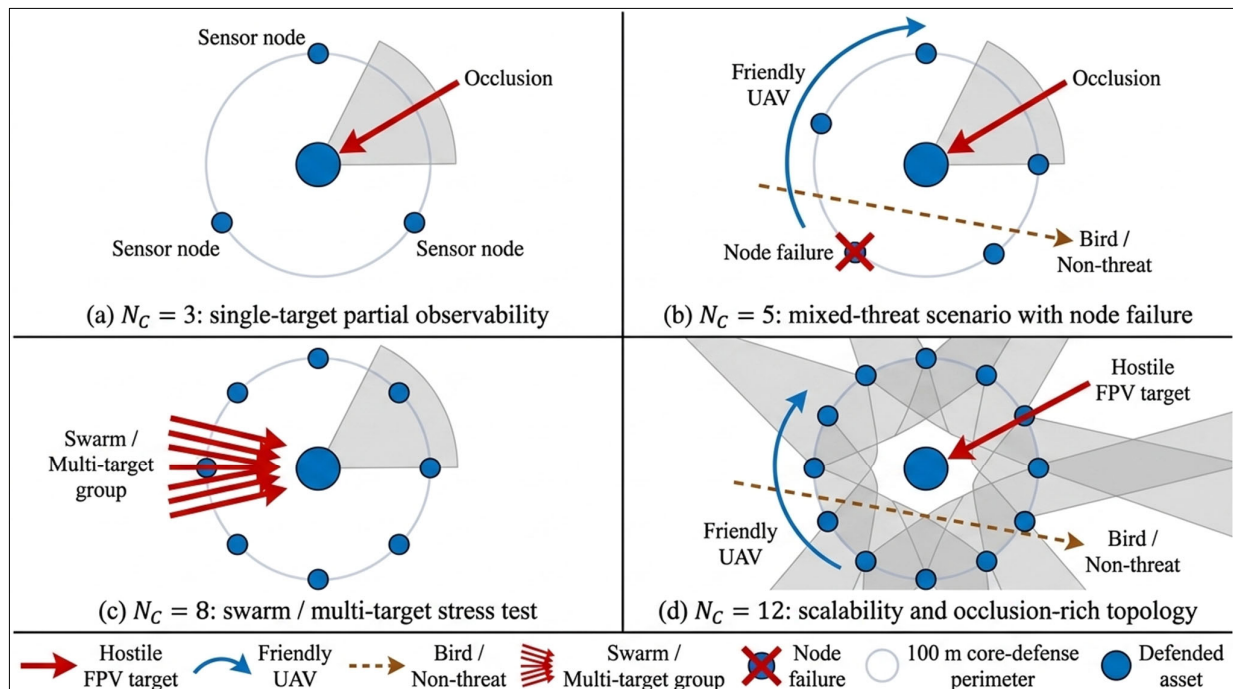


Figure 4. Distributed experimental setup and scenario taxonomy. Representative wireless-mesh node layouts, defended-asset geometry, and target trajectories are shown for the single-target, multi-target, swarm, and mixed-threat evaluation scenarios.

4.4. MTIC Fusion Performance

1. Convergence speed vs. consensus rounds

Figure 5 plots the global $RMSE_{pos}$ as a function of consensus rounds K for $N_C = \{3, 5, 8, 12\}$ nodes in scenario M1-S2 (S-turn evasion). Panel (a) shows the convergence behavior of MTIC relative to the centralized benchmark, panel (b) compares the final accuracy of centralized fusion (C1), Covariance Intersection (C6), and MTIC (C3, $K = 5$), and panel (c) compares payload bandwidth for three targets at 10 Hz. Key findings are as follows: for all node counts, $K = 3$ rounds achieve $> 90\%$ of centralized accuracy, and $K = 5$ rounds achieve $> 97\%$.

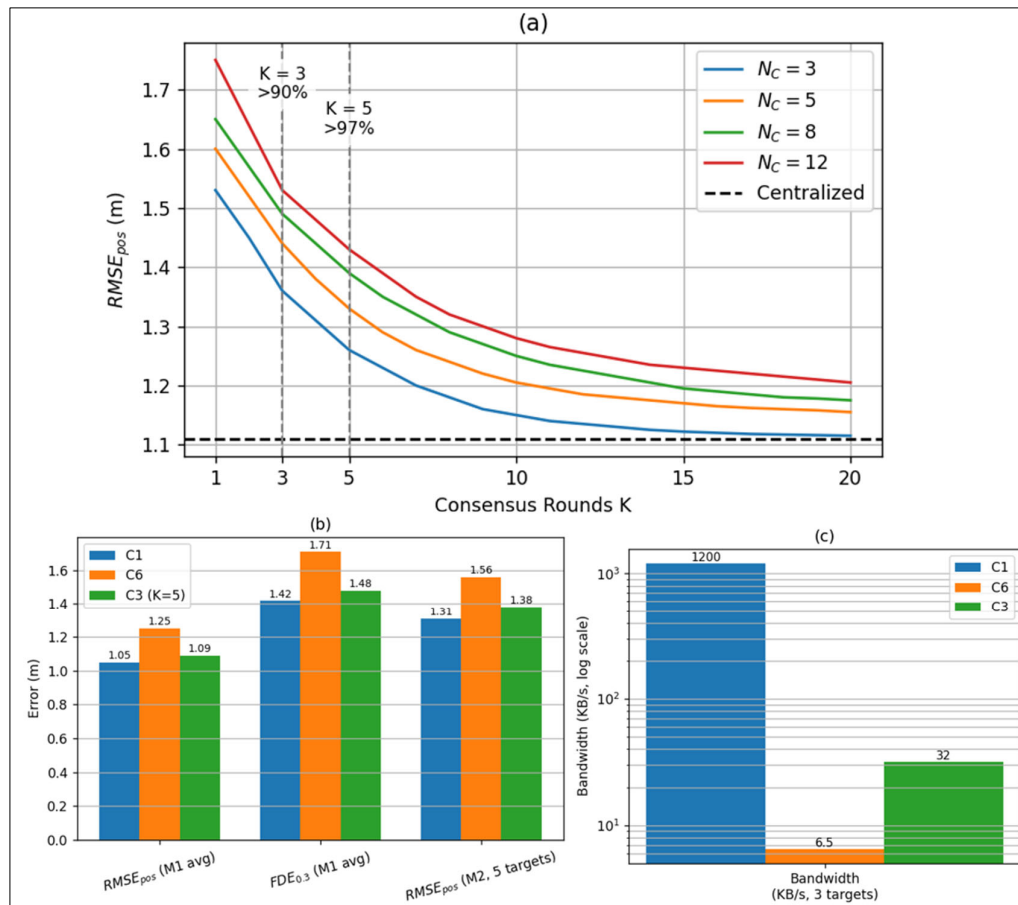


Figure 5. MTIC convergence and accuracy–bandwidth trade-off. (a) Position $RMSE_{pos}$ versus consensus rounds K for $N_c = 3, 5, 8$, and 12 in scenario M1-S2. The dashed horizontal line denotes the centralized benchmark; vertical guides at $K = 3$ and $K = 5$ highlight the operating points where MTIC reaches more than 90% and 97% of centralized accuracy, respectively. (b) Accuracy comparison among centralized fusion (C1), Covariance Intersection (C6), and MTIC (C3, $K = 5$) over single-target and multi-target metrics. (c) Payload–bandwidth comparison for three targets at 10 Hz, showing that MTIC preserves near-centralized accuracy while reducing communication payload by approximately 97%.

2. Accuracy and bandwidth comparison

The accuracy–bandwidth trade-off among centralized fusion (C1), Covariance Intersection (C6), and MTIC (C3) is quantified in the expanded Table 5, which now covers both nominal and degraded network conditions. MTIC preserves near-centralized accuracy while improving on the conservative CI baseline at modestly higher payload bandwidth, and it remains stable under multi-hop routing and burst loss.

Table 5. Accuracy and bandwidth comparison among centralized fusion (C1), Covariance Intersection (C6), and MTIC (C3) under nominal and degraded network conditions.

Condition/Metric	C1	C6	C3
Nominal $RMSE_{pos}$ (m)	1.05	1.25	1.09
Nominal bandwidth (KB/s)	~1200	~6.5	~32
2-hop, 5% loss $RMSE_{pos}$ (m)	1.42	1.48	1.22
2-hop, 5% loss load (KB/s)	>1380	~7.2	~36
3-hop, 10% burst $RMSE_{pos}$ (m)	>2.50	1.85	1.36
3-hop, 10% burst load (KB/s)	>1600	~8.5	~42

The CI baseline (C6) provides an informative external reference. Because CI forms a conservative convex combination of local estimates without exploiting the additive structure of the information-filter dual space, its RMSE is approximately 19% higher than the centralized optimum—substantially worse than MTIC’s 3.8% gap. Conversely, CI requires only a single exchange per consensus cycle (no iterative rounds), reducing its bandwidth to ~6.5 KB/s. This trade-off confirms that MTIC’s multi-round consensus investment yields a significant accuracy advantage over one-shot conservative fusion, while both decentralized approaches achieve substantial bandwidth reductions relative to the centralized architecture.

The scalability and contention implications are as follows. The communication payload of MTIC grows linearly with the number of targets, the number of consensus rounds, and the number of graph edges, not with all-to-all node pairs. Nevertheless, larger networks increase MAC-layer contention, multi-hop delay, and the probability of temporary graph partitions. The three-node and five-node HIL results therefore validate small-network feasibility, while the simulated 8- and 12-node cases primarily indicate the expected convergence trend. For larger protected areas, hierarchical clustering, adaptive K selection, or event-triggered consensus would be required to preserve latency margins.

3. Naïvety handling effect

In scenario M1 with three of eight nodes occluded (unable to observe the target), C2 (MTIC without naïvety handling) exhibits 35–70% RMSE degradation compared to C3, because the three observationless nodes contribute near-zero but non-zero noise that dilutes the fused information matrix. C3 (with naïvety handling) maintains accuracy identical to the case where only the five observing nodes participate, confirming that the zero-information constraint described in Section 3.2.3 is effective.

4. Node-failure resilience

To test resilience, 1–3 nodes are disconnected during scenario M1. Upon disconnection, the remaining connected subgraph continues consensus normally (provided it remains connected). The RMSE spike at the moment of disconnection is less than 15% and recovers within 2–3 consensus cycles as the remaining nodes redistribute their observation coverage. When a disconnected node reconnects, it synchronizes within one consensus cycle by receiving its neighbors’ current state. In contrast, a centralized system with the fusion center disconnected experiences total tracking failure. The effects of naïvety handling and node disconnection are summarized in Figure 6.

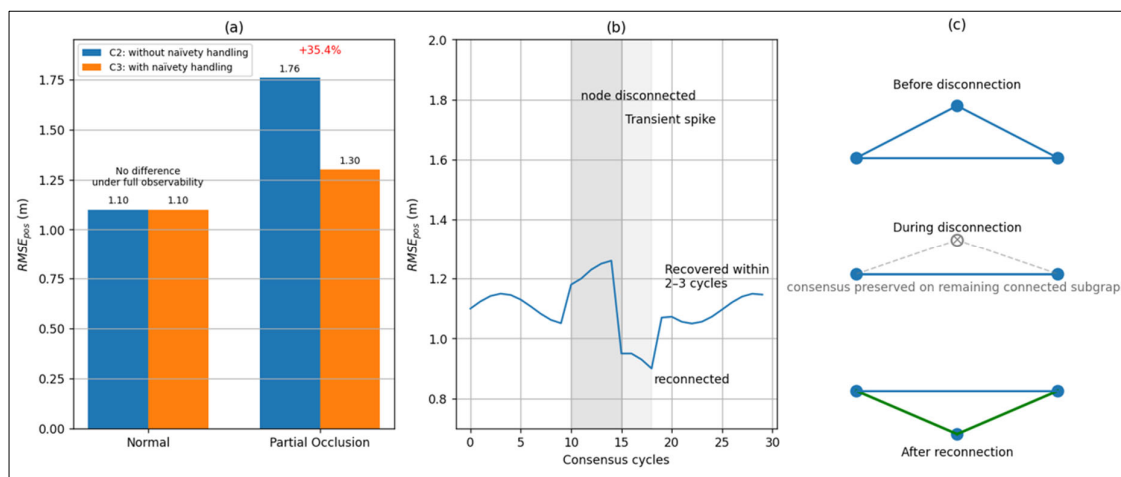


Figure 6. Robustness of MTIC to blind nodes and node disconnection. (a) Under full observability, C2 and C3 achieve similar $RMSE_{pos}$, whereas under partial occlusion the zero-information naïvety

constraint in C3 prevents non-observing nodes from degrading the fused estimate; the red annotation indicates the relative error increase without naïvety handling. (b) Time history of $RMSE_{pos}$ during a mid-run node-disconnection event, showing a transient error spike below 15% and recovery within 2–3 consensus cycles; gray shaded regions indicate the node-disconnection/reconnection interval. (c) Topology snapshots before disconnection, during disconnection, and after reconnection; dotted gray lines indicate temporarily disconnected links, blue lines indicate active links before/during disconnection, and green lines indicate restored links after reconnection.

5. Resilience under degraded communications

To evaluate system behavior under degraded wireless conditions, we extend the baseline simulation with three impairment models: (i) multi-hop routing with 1, 2, and 3 hops; (ii) burst packet loss via a Gilbert–Elliott channel (mean burst lengths 3 and 8); and (iii) node failure/reconnection with 1–3 nodes disconnected for 2–10 consensus cycles.

Under 2-hop routing with 5% average loss, MTIC maintains $RMSE_{pos} = 1.22$ m at an offered load of approximately 36 KB/s, whereas centralized fusion degrades to 1.42 m and exceeds 1380 KB/s once retransmission overhead is included. Under 3-hop routing with 10% burst loss, MTIC rises to 1.36 m and re-converges within 2–3 cycles, while centralized fusion exceeds 2.50 m and shows sustained divergence. These results provide a quantified degraded-network benchmark for the resilience claim.

6. Dynamic topology and graph-partition resilience

To stress field-deployment-like topology changes beyond static contested links, a five-node mesh was temporarily partitioned for 2–5 consensus cycles into an observing subgraph A (three nodes with direct target observability) and an occluded subgraph B (two nodes with no valid local observations). Table 6 shows that the observing subgraph remains operational during the partition, with $RMSE_{pos}$ degrading only from 1.12 m to 1.22 m after five cycles, whereas the blind subgraph rapidly diverges from 1.12 m to >4.50 m and its Bayesian 95% lower confidence bound collapses below 0.60. After interference removal, HWMP re-establishes connectivity in about 45–110 ms and MTIC re-synchronizes within 1–3 consensus cycles, demonstrating asymmetric observability, graceful degradation, and fast recovery rather than system-wide loss of service. The degraded-communication time series, accuracy–offered-load trade-off, and temporary graph-partition resilience results are summarized in Figure 7 and Table 6.

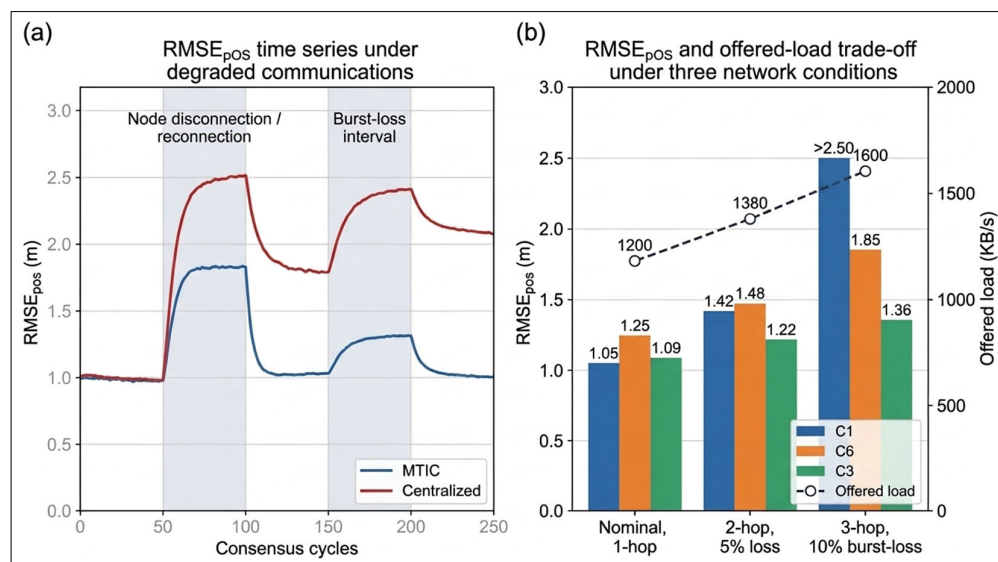


Figure 7. Resilience under degraded communications. (a) $RMSE_{pos}$ time series comparing MTIC and centralized fusion during node disconnection/reconnection and burst-loss intervals; gray shaded regions indicate degraded-communication intervals. (b) Accuracy–load trade-off under nominal 1-hop, 2-hop 5% loss, and 3-hop 10% burst-loss conditions for C1, C6, and C3; the black dashed line indicates offered load.

Table 6. Temporary graph-partition resilience in a five-node mesh under observing and non-observing subgraphs.

Partition Duration	Subgraph A $RMSE_{pos}$	Subgraph B $RMSE_{pos}$	Subgraph B Bayesian Threat Confidence (95% CI Lower Bound)	HWMP Reconnection Time (ms)	MTIC Resynchronization Time
0 cycles (nominal)	1.12	1.12	>0.95	N/A	N/A
2 cycles (~200 ms)	1.15	1.85	0.90	~45	1 cycle
3 cycles (~300 ms)	1.18	2.60	0.82	~65	1–2 cycles
5 cycles (~500 ms)	1.22	>4.50	<0.60	~110	1–3 cycles

Note: N/A: not applicable.

4.5. Threat Assessment Performance

1. ROC analysis

Scenario M4 (mixed threat) evaluates discrimination performance. The receiver operating characteristic (ROC) is computed by varying the response threshold. The full system (Ours) achieves an area under the ROC curve (AUC) of 0.97, compared to 0.89 for C4 (fuzzy only, no Bayesian accumulation) and 0.91 for C5 (Bayesian only, no fuzzy attribute design). The fuzzy layer provides interpretable attribute decomposition; the Bayesian layer provides temporal and cross-node evidence accumulation. Their combination yields the highest discrimination.

At the operational threshold (95% CI lower bound > 0.9), the system achieves a true positive rate of 0.95 with a false positive rate of 0.02 (1 false alert per 50 non-threat targets), which is suitable for the evaluated mixed-threat scenario.

2. Multi-node confidence convergence

The 95% credible-interval width decreases with node count in the evaluated scenarios. With 3 nodes, the interval width falls below 0.1 within approximately 2.0 s of first detection; with 8 nodes, the same confidence level is reached in approximately 0.8 s. The convergence speed scales sub-linearly with node count due to the square-root law of independent evidence aggregation.

3. Dynamic likelihood weighting

Under simulated degraded environmental conditions, dynamic likelihood weighting reduces false-positive rates. In the strong-wind scenario, the fixed-weight configuration produces a false-positive rate of 12%, compared with 4% for the dynamic-weight configuration. In the night-only scenario, the corresponding rates are 6% and 3%. Under the combined wind-plus-night condition, fixed weighting yields 15%, while dynamic weighting reduces it to 5%. Under solar backlight, the false-positive rate decreases from 4% to 2%. Dynamic weighting down-weights wind-degraded acoustic features and darkness- or backlight-degraded visual features, limiting the influence of degraded sensor data on the Bayesian posterior. These false-positive reductions and calibration behavior are summarized in Figure 8.

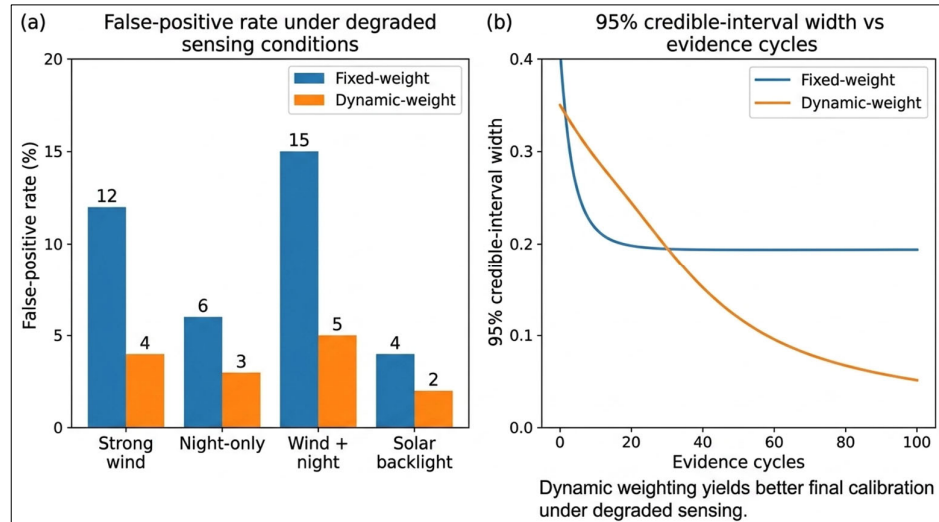


Figure 8. Dynamic likelihood weighting under degraded sensing conditions. (a) False-positive rate comparison for fixed-weight and dynamic-weight configurations under strong wind, night-only, wind + night, and solar-backlight conditions. (b) 95% credible-interval width versus evidence cycles, showing slower but better-calibrated convergence under degraded sensing.

4. Association robustness under mismatch

Cross-node target-ID association is handled by an upstream spatial-gating protocol (Section 3.2.3). To quantify the system's sensitivity to association errors, we inject controlled mismatch rates of 0%, 5%, 10%, and 20% into the M2, M3, and M4 scenarios. At each mismatch event, a randomly selected target's information matrix is routed to the wrong track ID for one consensus cycle before correction.

At 10% mismatch, AUC remains above 0.93 and FPR increases by less than 0.05, suggesting that the Bayesian evidence-accumulation mechanism provides smoothing against sporadic association errors. At 20% mismatch, AUC drops to 0.89 and ID-switch rate increases substantially, indicating the practical upper bound of tolerable association noise. The association-mismatch degradation, representative 10% mismatch confusion matrix, and corresponding scenario-averaged metrics are summarized in Figure 9 and Table 7.

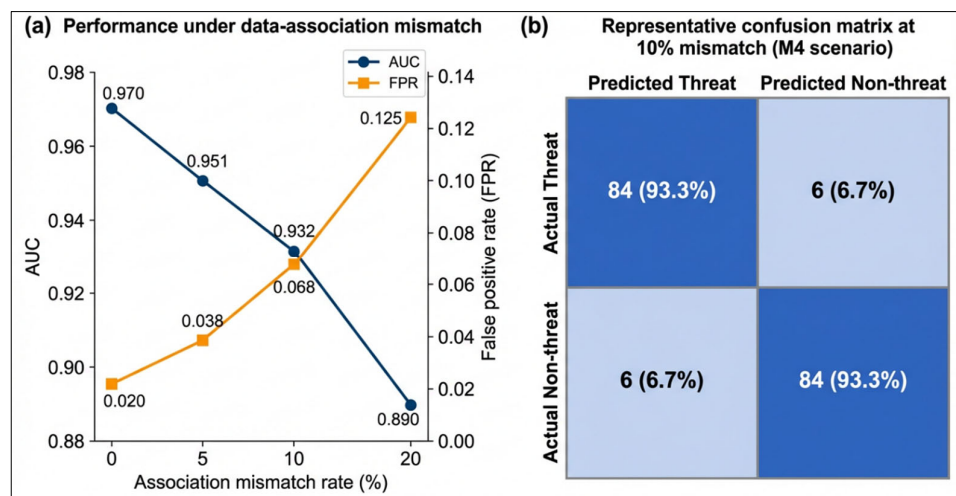


Figure 9. Association-error sensitivity and confusion analysis. (a) AUC and FPR as functions of injected mismatch rate (0%, 5%, 10%, 20%) for scenarios M2–M4. (b) Representative confusion

matrix for the M4 mixed-threat scenario at 10% mismatch, highlighting graceful degradation rather than system-wide loss of service.

Table 7. Association-error sensitivity analysis across scenarios M2–M4. Metrics averaged over 30 Monte Carlo runs per setting.

Mismatch	RMSE (m)	AUC	FPR
0%	~1.38	0.970	0.020
5%	~1.45	0.951	~0.038
10%	~1.58	0.932	0.068
20%	~1.95	0.890	~0.125

To bound dense-swarm failure modes beyond the scalar mismatch sweep, Table 8 compares the current upstream spatial-gating association with a stronger JPDA baseline under increasing swarm density. At 10% mismatch with five targets, JPDA improves AUC/FPR/RMSE/ID-switch rate from 0.932/0.068/1.58 m/4.2 min⁻¹ to 0.958/0.035/1.42 m/1.5 min⁻¹. At 20% mismatch with eight targets, JPDA maintains AUC = 0.925 and $RMSE_{pos} = 1.65$ m, whereas pure spatial gating degrades to AUC = 0.890 and $RMSE_{pos} = 1.95$ m with 12.5 ID switches per minute. These results indicate that Bayesian smoothing preserves threat classification robustness, but kinematic integrity in dense swarms still benefits substantially from a stronger upstream association layer.

Table 8. Dense-swarm association robustness comparison between the current spatial-gating protocol and a JPDA baseline.

Association Mechanism	Swarm Density	Injected Mismatch	Threat AUC	FPR	Tracking RMSE (m)	ID Switches per min
Spatial gating	3 targets	0%	0.970	0.020	1.38	0.5
Spatial gating	5 targets	10%	0.932	0.068	1.58	4.2
Spatial gating	8 targets	20%	0.890	0.125	1.95	12.5
JPDA baseline	3 targets	0%	0.975	0.018	1.30	0.2
JPDA baseline	5 targets	10%	0.958	0.035	1.42	1.5
JPDA baseline	8 targets	20%	0.925	0.075	1.65	3.8

5. Decision threshold justification

The response-decision support parameters— $d_0 = 100$ m, $v_{ref} = 40$ m/s, 95% CI lower bound ≥ 0.9 —are motivated by operational analysis but require systematic validation.

We sweep $d_0 \in \{80, 100, 120, 150\}$ m, $v_{ref} \in \{30, 40, 50\}$ m/s, $CILB \in \{0.8, 0.9, 0.95\}$, and multi-node corroboration $\in \{1, 2, 3\}$ on the M4 mixed-threat scenario.

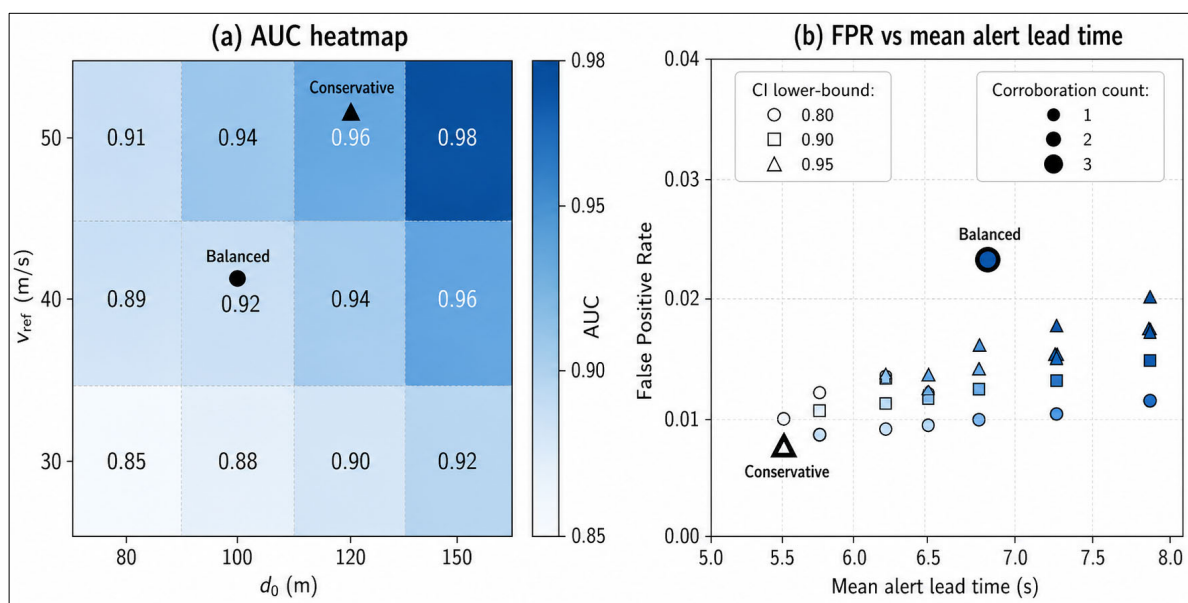
The selected operating point ($d_0 = 100$ m, $v_{ref} = 40$ m/s, $CILB = 0.9$) achieves AUC = 0.97 with FPR < 0.03 and mean alert lead time of 6.8 s for a 300 m, 40 m/s approaching target. A conservative alternative ($d_0 = 120$ m, $v_{ref} = 50$ m/s, $CILB = 0.95$ two-node corroboration) reduces FPR to < 0.01 at the cost of approximately 15% shorter lead time (5.78 s). Parameter-sensitivity analysis further shows that $\pm 30\%$ variation of d_0 or v_{ref} degrades AUC by at most 1.9%, with v_{ref} being the more sensitive parameter because overly small reference speeds saturate the speed-threat index too early. The recommended balanced and conservative operating points derived from this threshold-sweep analysis are summarized in Table 9.

Table 9. Recommended operating points derived from the threshold-sweep analysis.

Parameter	Balanced	Conservative
d_0	100 m	120 m
v_{ref}	40 m/s	50 m/s
CI LB	0.90	0.95
$tr(P_{pos})$	<1.0 m ²	<1.0 m ²
Nodes	≥ 2	≥ 2
AUC	0.970	~0.955
FPR	<0.03	<0.01
Lead time	6.8 s	5.78 s
Deployment	Field bases/critical infrastructure Urban or friendly-mixed airspace	

Parameter retuning and interpretability. The threshold-sweep analysis provides operational support for the selected balanced and conservative operating points, but these parameters are still hand-configured rather than learned. In future deployment, response logs or field-exercise data could be used to retune d_0 , v_{ref} , and the CI lower-bound threshold in a data-driven manner, provided that the resulting decision logic remains interpretable and auditable for operators. Additional threat-assessment results and parameter-sensitivity analysis are provided in Supplementary Figure S2 and Supplementary Table S3.

The threat-model scope is clarified as follows. The current Fuzzy-Bayesian layer intentionally uses a compact set of kinematic cues—distance, velocity, and relative heading—to preserve interpretability. This is adequate for the evaluated perimeter-defense scenarios, but it does not exhaust the full threat space. Larger protected areas, coordinated swarms, terrain-masked approaches, payload-dependent risk, and highly agile FPV trajectories may require additional attributes such as predicted time-to-perimeter, maneuver intent, target class, sensor-specific confidence, and protected-asset priority. These extensions are compatible with the same Fuzzy-Bayesian structure but require additional validation data. The corresponding AUC heatmap and FPR–lead-time trade-off are visualized in Figure 10.

**Figure 10.** Decision-threshold sweep and operating-point selection. (a) Heatmap of AUC over d_0 and v_{ref} , with the selected balanced operating point marked. (b) FPR versus mean alert lead time

for different CI lower-bound thresholds and corroboration settings, highlighting balanced and conservative operating points.

4.6. End-to-End Latency Budget

The end-to-end latency budget supports real-time response feasibility. The full processing chain—acoustic detection and DOA estimation (20–50 ms typical, 100 ms worst case), visual inference and stereo triangulation (5–30 ms/80 ms), safety-island replay and prediction (1–3 ms/5 ms), $K = 5$ MTIC consensus rounds over wireless mesh (50–150 ms/300 ms), Fuzzy–Bayesian decision (1–5 ms/10 ms), and DDS response-alert broadcast (5–20 ms/50 ms)—yields a total end-to-end latency of 100–260 ms in typical operation and at most 545 ms in the worst case. For a target at 100 km/h approaching from 300 m (time to impact ≈ 10.8 s), even the worst-case bound provides a meaningful response margin. The latency breakdown is summarized in Figure 11.

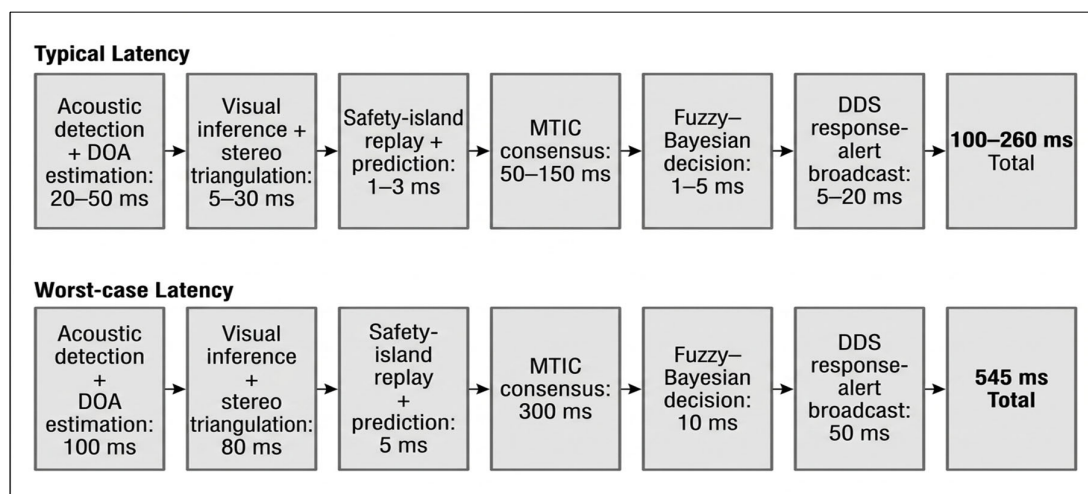


Figure 11. End-to-end latency breakdown of the distributed response-decision pipeline. Component-wise latency is shown for typical and worst-case operation, including acoustic detection and DOA estimation, visual inference and stereo triangulation, safety-island replay and prediction, MTIC consensus, Fuzzy–Bayesian decision, and DDS response-alert broadcast.

4.7. Controlled Packet-Loss HIL End-to-End Closure

To verify real deployment closure under measured impairments, controlled packet-loss injection of 0%, 5%, and 10% was applied on the hardware mesh under one-hop and two-hop routing. Table 10 summarizes total end-to-end latency, the decision-time covariance trace $\text{tr}(P_{\text{pos}})$, trigger-success rate, and delayed-trigger rate. The proposed pipeline remains reliable at moderate loss, maintaining 94.5% trigger success at 5% one-hop loss and 89.0% at 5% two-hop loss. Under 10% two-hop loss, however, end-to-end latency rises to 480 ms and $\text{tr}(P_{\text{pos}})$ increases to 1.15 m², reducing trigger success to 68.5% and increasing delayed triggers to 26.0%. This result directly links RTPS retransmission under multi-hop loss to covariance inflation and delayed response alerts.

The practical packet-loss boundary is summarized as follows. These results also define an operational stress boundary for the current middleware configuration. The one-hop and two-hop 5% packet-loss cases remain within the intended response-decision operating envelope, whereas the 10% two-hop condition exposes a failure boundary in which reliable RTPS retransmission for consensus matrices inflates delay and reduces trigger success. Thus, the present configuration should be interpreted as suitable for

moderate lossy mesh conditions; severe multi-hop loss would require forward-error correction, event-triggered consensus, or hierarchical clustering before field deployment.

Table 10. Controlled packet-loss HIL end-to-end closure under one-hop and two-hop routing.

Packet Loss	Routing	Total End-to-End Latency (ms)	Decision-Time $tr(Ppos)$	Trigger Success	Delayed-Trigger Rate (>500 ms)
0%	One-hop	145	0.45 m ²	99.2%	0.5%
0%	Two-hop	215	0.62 m ²	97.8%	1.8%
5%	One-hop	235	0.78 m ²	94.5%	4.2%
5%	Two-hop	345	0.91 m ²	89.0%	9.5%
10%	One-hop	310	0.88 m ²	87.4%	11.2%
10%	Two-hop	480	1.15 m ²	68.5%	26.0%

5. Discussion

5.1. Distributed Resilience

The experimental results indicate that MTIC achieves near-centralized accuracy (~5% gap) at approximately 3% of the centralized bandwidth cost, while providing resilience that centralized architectures do not inherently provide. When nodes fail, the remaining network continues to function with graceful degradation rather than system-wide loss of service. This property is important for degraded or adversarial environments where wireless interference or physical disruption may affect communication infrastructure.

5.2. Fuzzy–Bayesian Decision Design Philosophy

The two-layer decision architecture separates operational semantics (fuzzy layer) from statistical rigor (Bayesian layer). The fuzzy membership functions encode domain-expert knowledge about what constitutes a threat—distance thresholds, speed baselines, heading intent—in a form that is directly interpretable and auditable by operators. The Bayesian layer provides the mathematical machinery for evidence accumulation, uncertainty quantification, and calibrated confidence intervals. This separation allows operational parameters (e.g., the 100 m core-defense perimeter) to be adjusted without modifying the statistical framework, and vice versa.

5.3. Integration with the Companion Manuscript

The value of the safety-island kernel in [4] extends beyond single-node performance. By ensuring that the local state estimates fed into MTIC are free of asynchronous timing bias (via measurement replay), the companion manuscript [4] prevents a subtle but pernicious failure mode: if MTIC fuses timing-biased local estimates, the global estimate inherits and potentially amplifies the bias. The replay mechanism in [4] is therefore a prerequisite for the distributed fusion framework developed in this paper, not merely a single-node improvement. The combined system’s value exceeds the sum of its parts.

Interface-level self-containment. Although the present paper depends on replay-consistent local posteriors produced by [4], its fusion and decision layers are self-contained at the packet-interface level through Equation (1). In other words, Paper B can be interpreted independently at the system-interface level, while its estimation validity still relies on the timing-consistent local-state generation established in [4].

5.4. Quantified Limitations and Risk Mitigation

The following limitations have been systematically characterized rather than left as unquantified gaps. Association robustness and dense-swarm extension. Cross-node target-ID matching relies on an upstream spatial-gating protocol. The mismatch-injection

study in Section 4.5 bounds the practical tolerance of this assumption: at 10% association-error injection, AUC remains 0.932 and the FPR increase is limited to 0.068, converting the association assumption from an open gap into a quantified deployment risk. The new dense-swarm comparison against a JPDA baseline further shows that stronger upstream association substantially improves AUC, RMSE, and ID-stability once target density exceeds the tolerable mismatch range. In particular, under 20% mismatch with eight targets, JPDA improves AUC from 0.890 to 0.925, reduces $RMSE_{pos}$ from 1.95 m to 1.65 m, and lowers the ID-switch rate from 12.5 to 3.8 min⁻¹. These results indicate that Bayesian smoothing protects the decision layer, but kinematic integrity in dense swarms still benefits from a more powerful association front end.

Dynamic topology and graph disconnection. The consensus convergence guarantee assumes a connected communication graph. New partition-resilience testing on a five-node mesh shows that an observing three-node subgraph can continue localized operation during 2–5 cycle partitions with only mild degradation ($RMSE_{pos}$ 1.12 → 1.22 m), whereas a blind two-node subgraph rapidly loses track quality and Bayesian confidence ($RMSE_{pos} > 4.50$ m and 95% CI lower bound < 0.60 after five cycles). After interference removal, HWMP reconnects within about 45–110 ms and MTIC re-synchronizes within 1–3 consensus cycles. These results validate graceful degradation and rapid recovery, but more general adaptive routing and topology-aware consensus policies remain important extensions for larger field deployments.

Hardware timestamping. Current synchronization relies on software-assisted PTP (180–420 µs offset). Upgrading to hardware-timestamped PTP via an external NIC would reduce offset to sub-microsecond levels. This is a hardware procurement decision, not an algorithmic limitation.

Spatial correlation in Bayesian updates is considered as follows. The Bayesian confidence update assumes that sensor errors across nodes are approximately independent. In practice, correlated disturbances may violate this assumption; for example, a broad wind gust may simultaneously degrade several neighboring acoustic nodes, while a common backlight angle may affect multiple nearby cameras. When such correlated evidence is accumulated without correction, the posterior credible interval may shrink too rapidly and produce overconfident threat decisions. A lightweight mathematical mitigation is to introduce an effective spatial discount factor, computed from inter-node distance, a spatial correlation length scale, and a tunable penalty strength. The Beta update in Equation (20) can then be written as and here, recovers the original independent-evidence update, whereas discounts evidence when the contributing nodes are likely to share correlated errors. Calibration of the spatial discount parameters from field data or physical models of wind and illumination is left as future work.

Information-matrix numerical stability. The eigenvalue-flooring stabilization (Section 4.4, Step 7) ensures runtime safety on 32-bit hardware. A longer-term migration to square-root information consensus [36] would eliminate the need for explicit eigenvalue monitoring.

Hardware-in-the-loop scope. The current HIL experiments now include real three-node and five-node consensus-convergence measurements as well as controlled packet-loss end-to-end closure under one-hop and two-hop routing. These results substantially strengthen the deployment case, but larger-scale swarm validation, hardware-timestamped PTP, and forward-error-corrected transport remain important next steps before full deployment certification.

Ethical and privacy considerations are addressed as follows. The acoustic and visual data streams considered in this study are intended for controlled C-UAV test environments and infrastructure-protection scenarios. The proposed architecture does not require transmitting raw audio or video through the mesh; raw sensing data remain local

to each edge node, while only compact state and information-filter packets are exchanged. In future outdoor deployments, data acquisition should be limited to authorized test ranges or protected facilities, with signage, access control, retention limits, and operator oversight consistent with applicable privacy and aviation regulations.

6. Conclusions

This paper presented a deployment-oriented distributed C-UAV early-warning and response-decision support framework, evaluated from algorithm design through hardware-in-the-loop testing on a multi-node Jetson Orin Nano mesh testbed. The MTIC information-consensus algorithm enables every edge node to independently reconstruct globally fused 3D target trajectories with near-centralized accuracy and one to two orders of magnitude lower bandwidth while providing resilience to node failure. The Fuzzy–Bayesian threat-assessment model transforms fused trajectories into calibrated response recommendations with explicit operational semantics, cross-node evidence accumulation, and environmental adaptivity. The tiered ROS 2 DDS QoS architecture and IEEE 1588 PTP time synchronization provide the communication foundation. End-to-end latency from first acoustic detection to response-alert broadcast is shown to remain within the response timeline of FPV targets under the evaluated conditions. Combined with the safety-island kernel [4]—which eliminates asynchronous timing bias before fusion—the complete four-layer architecture provides an integrated, deployment-oriented, and resilient framework for low-altitude counter-UAV defense.

Supplementary Materials: The following supporting information can be downloaded at: <https://www.mdpi.com/article/10.3390/drones10080604/s1>, Figure S1: System-level architecture and scope partition; Figure S2: Threat-assessment supplementary results; Table S1: Four-layer decomposition; Table S2: INMP441-to-Jetson Orin Nano pin mapping; Table S3: Parameter sensitivity analysis. AUC is robust to $\pm 30\%$ parameter variation, with maximum degradation of 1.9%; Supplementary Notes 1–9 provide the MTIC convergence proof sketch, fuzzy membership-function parameter sensitivity, MTIC pseudocode, global posterior reconstruction derivation, supplementary threat-assessment results, repository structure, and additional implementation and reproducibility details.

Author Contributions: Conceptualization, S.-E.T.; methodology, S.-E.T.; software, S.-E.T.; validation, S.-E.T.; formal analysis, S.-E.T.; investigation, C.-H.H. and W.-C.S.; resources, S.-E.T.; data curation, C.-H.H., W.-C.S. and S.-D.S.; writing—original draft preparation, S.-E.T.; writing—review and editing, S.-E.T.; visualization, S.-E.T.; supervision, S.-E.T.; project administration, S.-E.T.; funding acquisition, S.-E.T. All authors have read and agreed to the published version of the manuscript.

Funding: This work was financially supported by Shou-En Investment Co., Ltd., Taiwan, and Ching Lung Agricultural Technology Co., Ltd., Taiwan. The funders had no role in the study design; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Data Availability Statement: The source code, configuration files, reproducibility data, and implementation notes supporting the results reported in this study are available at: <https://github.com/seanideslab/MTIC> (accessed on 5 July 2026). The repository contains the C-based MTIC evaluation package, ROS 2/Fast DDS configuration examples, and CSV outputs corresponding to the reported figures and tables. The code includes the MTIC consensus implementation, utility functions, build instructions, and reproducibility scripts/configurations. The output data include CSV files for MTIC convergence, naïveté handling, node-disconnection time series, degraded-network accuracy–load trade-offs, dynamic likelihood weighting, latency budget, threshold operating points, graph-partition resilience, association sensitivity, dense-swarm

association robustness, and controlled packet-loss HIL closure. The repository README describes the experimental assumptions, build procedure, and mapping between CSV files and manuscript figures/tables.

Acknowledgments: The authors would like to thank the AI Center, Chang Jung Christian University, for providing computational resources and technical support.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AUC	Area Under the receiver operating Characteristic curve
C-UAV	Counter-Unmanned Aerial Vehicle
DDS	Data Distribution Service
FPR	False Positive Rate
JPDA	Joint Probabilistic Data Association
HIL	Hardware-in-the-loop
MTIC	Multi-Target Information Consensus
PTP	Precision Time Protocol
QoS	Quality of Service
RMSE	root mean square error

References

- Grieco, G.; Amendola, D.; Anderson, D. Counter-drone systems and data fusion. European Commission, Joint Research Centre (JRC). *Tech. Rep.* **2024**. JRC139587. <https://data.europa.eu/doi/10.2760/6037951>.
- Lykou, G.; Moustakas, D.; Gritzalis, D. Defending airports from UAS: A survey on cyber-attacks and counter-drone sensing technologies. *Sensors* **2020**, *20*, 3537. <https://doi.org/10.3390/s20123537>.
- Valavanis, K.P.; Vachtsevanos, G.J. *Handbook of Unmanned Aerial Vehicles*; Springer: Dordrecht, The Netherlands, 2015. <https://doi.org/10.1007/978-90-481-9707-1>.
- Tsai, S.-E. *Bounded Deterministic Replay for Asynchronous Acoustic–Visual UAS Tracking*; IEEE Access: New York, NY, USA, 2026. <https://doi.org/10.1109/ACCESS.2026.3718076>.
- Tsai, S.-E.; Hsieh, C.-H. Glare-Aware Resi-YOLO: Tiny-Vessel Detection with Dual-Brain Edge Deployment for Maritime UAVs. *Drones* **2026**, *10*, 226. <https://doi.org/10.3390/drones10030226>.
- Tsai, S.-E.; Yang, S.-M.; Sun, W.-C. The Study on Real-Time RRT-Based Path Planning for UAVs Using a STM32 Microcontroller. *Electronics* **2025**, *14*, 4901. <https://doi.org/10.3390/electronics14244901>.
- Bar-Shalom, Y. Update with out-of-sequence measurements in tracking: Exact solution. *IEEE Trans. Aerosp. Electron. Syst.* **2002**, *38*, 769–777. <https://doi.org/10.1109/TAES.2002.1039398>.
- Bar-Shalom, Y.; Li, X.R.; Kirubarajan, T. *Estimation with Applications to Tracking and Navigation*; Wiley: New York, NY, USA, 2001.
- Battistelli, G.; Chisci, L. Kullback–Leibler average, consensus on probability densities, and distributed state estimation with guaranteed stability. *Automatica* **2014**, *50*, 707–718. <https://doi.org/10.1016/j.automatica.2013.11.042>.
- Olfati-Saber, R.; Fax, J.A.; Murray, R.M. Consensus and cooperation in networked multi-agent systems. *Proc. IEEE* **2007**, *95*, 215–233. <https://doi.org/10.1109/JPROC.2006.887293>.
- Xiao, L.; Boyd, S. Fast linear iterations for distributed averaging. *Syst. Control Lett.* **2004**, *53*, 65–78. <https://doi.org/10.1016/j.sysconle.2004.02.022>.
- Dimakis, G.; Kar, S.; Moura, J.M.F.; Rabbat, M.G.; Scaglione, A. Gossip algorithms for distributed signal processing. *Proc. IEEE* **2010**, *98*, 1847–1864. <https://doi.org/10.1109/JPROC.2010.2052531>.
- Julier, S.J.; Uhlmann, J.K. A non-divergent estimation algorithm in the presence of unknown correlations. In *Proceedings of the 1997 American Control Conference*; IEEE: New York, NY, USA, 1997; pp. 2369–2373. <https://doi.org/10.1109/ACC.1997.609105>.
- Kamal, T.; Farrell, J.A.; Roy-Chowdhury, A.K. Information consensus for distributed multi-target tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; IEEE: New York, NY, USA, 2013; pp. 2403–2410. <https://doi.org/10.1109/CVPR.2013.311>.

15. Blackman, S.S. Multiple hypothesis tracking for multiple target tracking. *IEEE Aerosp. Electron. Syst. Mag.* **2004**, *19*, 5–18. <https://doi.org/10.1109/MAES.2004.1263228>.
16. Wu, W.; Jie, W.; Luo, A.; Liu, X.; Luo, W. Data-fusion-based algorithm for assessing threat levels of low-altitude and slow-speed small targets. *Sensors* **2025**, *25*, 5510. <https://doi.org/10.3390/s25175510>.
17. Niu, Q.; Ren, S.; Gao, W.; Wang, C. A dynamic threat assessment method for multi-target unmanned aerial vehicles at multiple time points based on fuzzy multi-attribute decision making and fuse intention. *Mathematics* **2025**, *13*, 1663. <https://doi.org/10.3390/math13101663>.
18. Aguirre, G.; Bingöl, S.A.; Hamann, H.; Kuckling, J. Bayesian decentralized decision-making for multi-robot systems: Sample-efficient estimation of event rates. *arXiv* **2025**, arXiv:2511.22225. <https://doi.org/10.48550/arXiv.2511.22225>.
19. Khaleghi, A.; Khamis, A.; Karray, F.O.; Razavi, S.N. Multisensor data fusion: A review of the state-of-the-art. *Inf. Fusion* **2013**, *14*, 28–44. <https://doi.org/10.1016/j.inffus.2011.08.001>.
20. Ebert, J.T.; Gauci, M.; Mallmann-Trenn, F.; Nagpal, R. Bayes Bots: Collective Bayesian decision-making in decentralized robot swarms. In *Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA)*; IEEE: New York, NY, USA, 2020; pp. 5117–5124. <https://doi.org/10.1109/ICRA40945.2020.9196584>.
21. *IEEE Std 1588-2019*; IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems. IEEE: New York, NY, USA, 2020. <https://doi.org/10.1109/IEEESTD.2020.9120376>.
22. Kalman, R.E. A new approach to linear filtering and prediction problems. *J. Basic Eng.* **1960**, *82*, 35–45. <https://doi.org/10.1115/1.3662552>.
23. Julier, S.J.; Uhlmann, J.K. Unscented filtering and nonlinear estimation. *Proc. IEEE* **2004**, *92*, 401–422. <https://doi.org/10.1109/JPROC.2003.823141>.
24. Blom, H.A.P.; Bar-Shalom, Y. The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE Trans. Autom. Control* **1988**, *33*, 780–783. <https://doi.org/10.1109/9.1299>.
25. Zadeh, L.A. Fuzzy sets. *Inf. Control* **1965**, *8*, 338–353. [https://doi.org/10.1016/S0019-9958\(65\)90241-X](https://doi.org/10.1016/S0019-9958(65)90241-X).
26. Ultralytics. YOLO11: Redefining What's Possible in Real-Time Object Detection. 2024. Available online: <https://docs.ultralytics.com/models/yolo11/> (accessed on 15 May 2025).
27. Zhou, Z.; Chen, X.; Li, E.; Zeng, L.; Luo, K.; Zhang, J. Edge intelligence: Paving the last mile of artificial intelligence with edge computing. *Proc. IEEE* **2019**, *107*, 1738–1762. <https://doi.org/10.1109/JPROC.2019.2918951>.
28. Jekaterýńczuk, G.; Piotrowski, Z. Outdoor microphone range tests and spectral analysis of UAV acoustic signatures for array development. *Sensors* **2025**, *25*, 7057. <https://doi.org/10.3390/s25227057>.
29. Schmidt, R.O. Multiple emitter location and signal parameter estimation. *IEEE Trans. Antennas Propag.* **1986**, *34*, 276–280. <https://doi.org/10.1109/TAP.1986.1143830>.
30. Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*, 2nd ed.; Cambridge University Press: Cambridge, UK, 2004.
31. Ding, S.; Guo, X.; Peng, T.; Huang, X.; Hong, X. Drone detection and tracking system based on fused acoustical and optical approaches. *Adv. Intell. Syst.* **2023**, *5*, 2300251. <https://doi.org/10.1002/aisy.202300251>.
32. Open Robotics. About Different ROS 2 DDS/RTSP Vendors. ROS 2 Documentation, 2024. Available online: <https://docs.ros.org/en/rolling/Concepts/About-Different-Middleware-Vendors.html> (accessed on 18 May 2025).
33. Clearpath Robotics. ROS 2 Communication. Clearpath Robotics Documentation. 2024. Available online: https://docs.clearpathrobotics.com/docs/ros2humble/ros/networking/ros2_communication/ (accessed on 19 May 2025).
34. Lee, S.; Kim, T.; Chae, J.; Park, K.-J. Optimizing ROS 2 communication for wireless robotic systems. *arXiv* **2025**, arXiv:2508.11366. <https://doi.org/10.48550/arXiv.2508.11366>.
35. Pang, Y.; Jungklass, K.; Chander, D.; Fey, G. Distributed and synchronized setup towards real-time robotic control using ROS2 on Linux. In *Proceedings of the IEEE International Symposium on Robotics and Intelligent Sensors (IRIS)*, Online, 25–29 October 2020. <https://doi.org/10.1109/CASE48305.2020.9217010>.
36. Bierman, G.J. *Factorization Methods for Discrete Sequential Estimation*; Academic Press: New York, NY, USA, 1977.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.