ispiacademy : Pages:1–17

Original Research

Testing and Performance Evaluation Methods for UAS Detect-and-Avoid Using Simulated and Live Encounter Scenarios

David Kiptoo[1] and Samuel Njuguna[2]

[1]University of Nairobi, Department of Mechanical Engineering, University Way, Nairobi 00100, Kenya.
[2]Egerton University, Department of Mechanical and Manufacturing Engineering, Njoro Campus, Nakuru 20115, Kenya.

Abstract
As UAS operations expand in density and complexity, detect-and-avoid capabilities are expected to provide an acceptable level of risk with respect to mid-air collisions, while remaining compatible with existing traffic and scalable to emerging concepts such as dense urban operations. Evaluating detect-and-avoid performance under both nominal and off-nominal encounter geometries requires carefully constructed methodologies that integrate high-fidelity simulation, controlled live-flight experiments, and rigorous statistical analysis. This paper examines test and evaluation constructs for detect-and-avoid systems with emphasis on harmonizing simulated encounter sets and live scenarios, enabling quantitative assessment of detection performance, maneuver guidance, interoperability, and residual risk. The discussion focuses on systematic strategies for constructing encounter models, defining operational scenarios, specifying performance metrics, and integrating measurement uncertainty, without relying on a single environment or data source. A particular emphasis is placed on traceability between modeled encounters and operational use cases, including those characterized by sparse surveillance, heterogeneous equipage, and mixed levels of automation. The paper outlines approaches that connect algorithmic behavior with safety-relevant indicators such as conflict rate, loss-of-well-clear frequency, and collision probability, in a way that allows incremental validation and refinement. The resulting framework enables transparent interpretation of detect-and-avoid performance across simulated and live encounter campaigns while remaining adaptable to different system architectures and operational concepts.

1. Introduction

Detect-and-avoid functions for uncrewed aircraft systems are intended to support safe integration into airspace where conventional see-and-avoid responsibilities have historically been assigned to onboard human pilots [1]. The transition from pilot-centric visual acquisition to sensor- and algorithm-based detect-and-avoid induces structural changes in how conflicts are observed, predicted, and resolved. Performance evaluations that were once implicitly tied to human capability must now be formalized in terms of measurable detection probabilities, trajectory prediction accuracy, guidance effectiveness, and the resulting effect on collision risk. These evaluations must be conducted under operationally plausible encounter geometries that span cooperative and non-cooperative intruders, surveillance outages, latency, track fragmentation, and maneuver uncertainty.

Developing systematic test and performance evaluation methods for such systems presents several challenges [2]. Operationally representative encounters are rare in routine traffic and potentially hazardous to replicate directly; safety-critical edge cases often occupy small volumes of the encounter state space yet dominate risk; and detect-and-avoid implementations exhibit complex couplings among sensing, estimation, logic, and guidance. A single methodology, whether purely simulation-based or purely experimental, will not reliably span this landscape. Instead, a combination of simulated encounter sets, hardware-in-the-loop configurations, and controlled live scenarios is needed, along with mathematical structures that enable consistent interpretation of results across these contexts.

The evaluation problem is not restricted to detection range or guidance responsiveness taken in isolation. Detect-and-avoid functions interact with communication protocols, air traffic separation minima, latency in command-and-control links, and pilot or autonomy behavior in both ownship and intruder aircraft [3]. Consequently, performance must be evaluated across multiple layers: encounter geometry generation, sensing and tracking, conflict detection and trajectory prediction, maneuver selection, and closure back to airspace-level risk measures. Capturing these layers requires test constructs that define measurable quantities at each stage while ensuring compatibility with higher-level safety arguments and regulatory performance expectations.

This paper develops a structured view of these constructs for detect-and-avoid systems in both simulated and live encounter scenarios. The aim is to provide a neutral yet explicit framework that can accommodate different sensor modalities, algorithmic architectures, and operational environments. The subsequent sections discuss the operational and regulatory context within which detect-and-avoid systems operate, formulate encounter models and scenario generation methods suited for both simulation and experimentation, describe algorithmic abstractions amenable to analysis, outline simulation-based and live test methodologies, and propose statistical techniques for synthesizing evidence into performance indicators relevant to safety assessments [4]. The discussion is organized to avoid prescribing a single design but instead to isolate fundamental elements required for traceable and reproducible evaluation.

Table 1. *Scope and Core Elements.*

Aspect	Description	Role in Paper
Operational shift	From human see-and-avoid to UAS detect-and-avoid	Motivates structured performance evaluation
Evaluation focus	Use of simulated and live encounters	Links evidence across test environments

Table 2. *Evaluation Drivers.*

Driver	Description	Implication
Airspace integration	Mixed crewed and uncrewed operations	Requires compatible detect-and-avoid behavior
System coupling	Sensors, logic, guidance, human or autonomy	Necessitates multi-layer performance metrics

Table 3. *Key Foundational Concepts.*

Concept	Summary	Evaluation Use
Well-clear region	Geometric-temporal exclusion volume	Basis for loss-of-well-clear indicators
Decision surface	Boundary in estimated state space	Determines alert sensitivity and stability

Table 4. *Margins and Scenario Equivalence.*

Concept	Summary	Evaluation Use
Alert margin	Time between first alert and projected boundary entry	Feasibility of avoidance maneuvers
Scenario equivalence	Matching key kinematic descriptors	Aligns simulated and live encounter data

Table 5. *Operational Context Dimensions.*

Context	Characteristics	Implication
Controlled airspace	Structured routes, cooperative surveillance	Emphasis on interoperability and nuisance control
Uncontrolled airspace	Heterogeneous, non-cooperative traffic	Emphasis on coverage and robustness

Table 6. *Encounter Modeling Elements.*

Element	Description	Purpose
Relative state	Position and velocity in local frame	Defines encounter geometry
Dynamics model	Kinematic or stochastic evolution	Generates trajectories for testing

Table 7. *Scenario Design Considerations.*

Aspect	Description	Evaluation Role
Coverage	Range of miss distances and approach angles	Samples nominal and stressing cases
Importance sampling	Biased selection with weights	Supports rare-event performance estimation

Table 8. *Functional Mapping of Components.*

Component	Input	Output
Detection and tracking	Sensor measurements	Estimated intruder states
Alerting logic	Estimated encounter states	Alert levels and triggers

2. Foundational Performance Concepts for Detect-and-Avoid Evaluation

A structured evaluation of detect-and-avoid performance for uncrewed aircraft systems requires explicit articulation of the performance concepts that connect algorithmic behavior, sensing capabilities, encounter geometries, and resulting safety and operational effects. Before specifying encounter models or test campaigns, it is necessary to define the functional roles of detect-and-avoid subsystems in terms that admit quantification, are compatible with diverse architectures, and can be consistently interpreted across simulated and live scenarios. This section introduces foundational concepts underpinning such evaluations, including formal representations of well-clear maintenance and collision avoidance functions, the decomposition of performance into observables, the incorporation of uncertainty and latency, and the mapping from local detect-and-avoid outcomes to aggregate indicators of acceptability within a given operational context. [5]

At the core of detect-and-avoid evaluation lies the notion of preserving a well-clear volume between ownship and intruder aircraft under realistic sensing and maneuvering constraints. Let r(t) denote the

Table 9. *Prediction and Guidance Aspects.*

Function	Description	Evaluation Focus
Conflict prediction	Future relative state assessment	Timeliness of hazard indication
Guidance selection	Maneuver under operational constraints	Separation maintenance and stability

relative position vector between ownship and intruder in a local horizontal-vertical frame and v(t) the corresponding relative velocity. A minimal representation treats r(t) and v(t) as trajectories in a continuous state space subject to aircraft dynamics and external disturbances. A well-clear definition imposes geometric and temporal constraints that delineate states considered operationally acceptable. Although specific thresholds may vary by airspace class or concept of operations, the evaluation framework views them as parameters defining an exclusion region in the relative state space [6]. A detect-and-avoid implementation must, with high consistency, prevent trajectories from entering this region when reasonably avoidable given the information and maneuver authority available. The evaluation task, therefore, is to characterize the conditions under which such prevention is achieved or not achieved, rather than to embed any particular threshold selection as an intrinsic property of the methodology.

To formalize these ideas while maintaining concise expressions, consider a loss-of-well-clear indicator function L operating on the relative state at time t. For example, one may write

$$L(t) = [7] \begin{cases} 1 & \text{if } r(t) \in W, \\ 0 & \text{otherwise,} \end{cases}$$

where W is the chosen well-clear region. In this formulation, detect-and-avoid performance can be partially summarized through probabilities or frequencies associated with events such as the first time t at which L(t) becomes 1, conditional on specified encounter classes and system configurations. However, the binary indicator alone is too coarse to support nuanced evaluation, as it does not reflect proximity to the boundary, available time margins, or the quality of earlier advisories. Accordingly, the framework introduces continuous metrics, such as predicted time to boundary crossing, minimum separation achieved given advisories and responses, and stability of alerts, each defined with reference to L(t) but conveying more detailed information about system behavior in the approach to potential loss-of-well-clear.

Detect-and-avoid algorithms do not operate on the true state directly but on estimates formed from sensor data, communication links, and filtering processes [8]. Let $\hat{r}(t)$ and $\hat{v}(t)$ denote the estimated relative position and velocity available to the detect-and-avoid logic, and let $\epsilon_r(t)$ and $\epsilon_v(t)$ represent associated estimation errors. These quantities are influenced by sensor noise, biases, latency, track association decisions, and possible gaps in surveillance coverage. A generic expression for the estimated state in terms of true state and error can be written as

$$\hat{r}(t) = r(t) + \epsilon_r(t),$$

with an analogous relation for velocity. Performance concepts must therefore be framed in a way that distinguishes between failures attributable to fundamental limitations in the information set and those attributable to algorithmic choices given that information [9]. For instance, a scenario in which the intruder remains undetected due to low radar cross section or lack of cooperative equipage presents different interpretive implications than a scenario in which the intruder is tracked but the logic delays or suppresses alerts despite converging trajectories.

A foundational construct is the detect-and-avoid decision surface: a boundary in the space of estimated encounter states that separates regions where alerts or maneuvers are commanded from regions where no action is taken. Abstractly, let x(t) denote a feature vector derived from $\hat{r}(t)$, $\hat{v}(t)$, and possibly higher-order predictions or uncertainty measures. The detect-and-avoid logic can be viewed as computing a decision variable D(x(t)) and comparing it to calibrated thresholds. A simplified representation is

$$a(t) = [10] \begin{cases} 1 & \text{if } D(x(t)) \geq \eta, \\ 0 & \text{if } D(x(t)) < \eta, \end{cases}$$

where a(t) indicates whether a designated alert or advisory is active and η is a threshold selected to balance sensitivity and robustness. Although actual systems exhibit multiple alert levels and more complex hysteresis, this abstraction suffices for defining conceptual performance measures. The location and shape of this decision surface influence detection timeliness, susceptibility to nuisance alerts, and resilience to noise and latency. Evaluation methodologies must be capable of probing the behavior of D(x(t)) under varied encounter and sensor conditions, even when its exact implementation is proprietary or embedded.

Latency and update rate are central performance determinants [11]. Information used by detect-and-avoid logic may be delayed relative to the true state due to sensor processing, data fusion, network transmission, and cockpit or autopilot implementation. Let Δ represent an effective decision latency. Conceptually, the logic bases its assessments at time t on a state that approximates the true state at time $t - \Delta$. For an approaching intruder, this compresses the available time margin between first reliable detection and potential loss-of-well-clear. A simple indicator of effective margin at initial alert time t_a can be expressed as a function of relative kinematics: [12]

$$M = t_{\mathrm{LoWC}} - t_a,$$

where t_{LoWC} is the time at which L(t) would first become 1 in the absence of any avoidance maneuver. The distribution of M across encounters, conditional on scenario class and system configuration, forms a key metric: larger positive margins indicate earlier alerts, while small or negative margins indicate advisories that may be infeasible to execute in time. An evaluation framework must capture how Δ, sensing range, and decision surface parameters jointly influence this margin distribution, and how execution variability further modifies realized separation outcomes.

Another foundational concept is the operational envelope of detect-and-avoid applicability. UAS may operate in regions with conflicting constraints on altitude, speed, minimum separation, and traffic complexity [13]. Detect-and-avoid performance must be interpreted with respect to the envelope within which the system is intended and demonstrated to function. Let O denote a set of operating conditions parameterized by airspeed ranges, altitude bands, equipage assumptions, and surveillance infrastructure. Tests conducted outside O can be informative for robustness assessment but should not be conflated with core performance claims. Within the framework, O acts as a conditioning domain: encounter models, sensor models, and performance metrics are explicitly tied to specified subsets of O, avoiding implicit extrapolation.

To support integration of simulated and live testing, the concept of scenario equivalence is introduced. Two encounters, one simulated and one flown, are considered equivalent for a given performance question if they match, within defined tolerances, on a set of summary descriptors such as initial range, relative bearing, closure rate, altitude separation, and equipage attributes. Let s denote a vector of such descriptors. For encounter i, either simulated or live, define s_i as its descriptor vector [14]. Equivalence for evaluation purposes is then approximated by proximity of s_i to a target scenario descriptor s× , measured using a suitable norm with application-specific tolerances. While this notion does not equate full trajectory histories, it enables alignment of heterogeneous data sources when estimating metrics that depend primarily on these descriptors, such as distributions of alert margins or false alert rates in particular geometric classes.

Uncertainty treatment is foundational to any credible performance evaluation. Encounter models, sensor characterizations, behavioral response models, and environmental conditions all carry uncertainties that propagate into performance metrics. Within this section, the emphasis is on conceptual roles rather than specific numerical values [15]. One may regard any performance metric θ as a function of underlying uncertain parameters ξ, so that

$$\theta = f(\xi).$$

Here, ξ aggregates quantities such as encounter rate parameters, intruder speed distributions, latency distributions, and compliance levels with advisories. Evaluation seeks not a single point estimate of θ, but a characterization over the plausible range of ξ, leading to intervals or envelopes that reflect epistemic and aleatory uncertainty. Tests and simulations are then interpreted as observations providing information about ξ or directly about θ, and the methodology prescribes how evidence from different sources modifies uncertainty without assuming complete certainty in any single modeling element. [16]

A further foundational element is the separation between algorithm-internal measures and externally observable performance. Many detect-and-avoid implementations compute internal quantities such as probabilistic conflict metrics, look-ahead risk measures, or optimization costs. While these are informative for design and tuning, evaluation in operationally relevant terms must be based on quantities that can be independently observed or reconstructed, such as recorded alert times, executed maneuvers, and realized separations. The framework encourages mapping internal measures to observable surrogates whenever possible, but it treats internal measures as auxiliary, not primary, evidence for safety or effectiveness [17]. This distinction is important when comparing different systems whose internal logic may be inaccessible; evaluation must rest on transparent, externally verifiable indicators.

Interoperability with surrounding airspace users provides another conceptual dimension. Detect-and-avoid advisories executed by UAS should not systematically induce conflicts or unacceptable compressions of separation with third-party aircraft. This leads to the notion of compatibility metrics: measures that assess whether typical advisory patterns remain consistent with standard right-of-way rules, expected maneuver conventions, and controller or pilot expectations. Although detailed modeling of multi-actor interactions resides in subsequent sections, the foundational concept is that detect-and-avoid evaluation extends beyond dyadic ownship-intruder geometry to include the broader traffic context [18]. Test designs incorporating multiple intruders or background traffic must therefore define metrics that capture the degree to which detect-and-avoid advisories remain compatible with that context, both in simulation and live scenarios.

Finally, these foundational concepts collectively motivate requirements for traceability in any test and evaluation campaign. For each reported metric, the underlying encounter definitions, operating conditions, sensor and latency assumptions, decision surface characteristics, and uncertainty treatments must be identifiable and, to the extent possible, reproducible. This does not imply that all such elements are known exactly, but rather that their roles are explicitly represented in the structure of the evaluation. When tests are extended to new airspace classes, new sensor technologies, or alternative detect-and-avoid algorithms, the same conceptual scaffolding can be applied: define well-clear and alerting concepts, specify the operational envelope, characterize sensing and latency, identify decision structures, construct scenario equivalences between simulation and live data, and propagate uncertainties in a disciplined manner [19]. Establishing these foundations early in the evaluation process enables subsequent sections of the study to focus on specific model choices, encounter sets, simulation architectures, live test designs, and statistical analyses, while maintaining conceptual continuity and interpretability across diverse detect-and-avoid implementations and operational concepts.

3. Background on UAS Detect-and-Avoid Operational Context

Detect-and-avoid systems for UAS are generally structured to fulfill two complementary functions: remain-well-clear maintenance and collision avoidance in last-resort conditions. These functions must operate within surveillance capabilities that may include automatic dependent surveillance, transponder-based systems, primary radar, non-cooperative sensors, or combinations thereof. The detect-and-avoid logic typically observes a processed track picture, subject to latency, measurement noise, track swaps, and dropouts, and issues alerts or maneuver guidance according to defined thresholds on time to loss-of-well-clear, predicted miss distance, or probabilistic indicators.

The operational context constrains acceptable performance characteristics [20]. In controlled airspace, UAS may operate with cooperative surveillance and structured trajectories, which influence the encounter model toward relatively constrained speed and altitude distributions. In lower-level or uncontrolled environments, intruders can include small uncrewed aircraft, general aviation traffic, or non-cooperative objects, for which surveillance participation may be low and dynamic behavior less predictable. These differing contexts induce distinct encounter priors, traffic densities, and closure rates that any evaluation framework must explicitly capture, rather than assuming a single canonical encounter set.

A central notion is the mapping from detect-and-avoid performance to safety objectives [21]. Safety arguments often reference bounds on the rate of mid-air collisions per flight hour or per operation, and on the frequency of losses of well clear. Detect-and-avoid evaluations must therefore deliver intermediate metrics that can be translated into these system-level objectives. Metrics of interest include probability of timely detection of an impending loss-of-well-clear, probability of nuisance or unnecessary alerts in benign encounters, rate of maneuver advisories that are operationally feasible, and the residual probability that the logic fails to prevent a loss-of-well-clear or collision given a threatening encounter. Because detect-and-avoid decisions are typically executed in closed-loop with pilot or autopilot responses, these metrics are conditioned not only on algorithmic outputs but also on behavioral models of response delay, adherence, and variability.

In addition, detect-and-avoid systems are deployed in layered architectures that include procedural mitigations, geofencing, altitude reservations, and traffic flow constraints [22]. The incremental contribution of detect-and-avoid relative to these other mitigations must be evaluated without overstating its role. This motivates methods that can estimate how much incremental risk reduction is achieved under clearly specified assumptions while acknowledging uncertainties in underlying encounter models, sensor performance, and operational behavior. Addressing such uncertainties requires statistical formulations that are compatible with both simulation and live data, allowing performance evidence from different sources to be combined.

Within this context, test and performance evaluation methodologies must be designed so that any chosen scenario set, simulation configuration, or live campaign can be traced back to assumptions about airspace structure, traffic behavior, equipage levels, and rules of the air. Without such traceability, performance estimates risk being either optimistic or pessimistic in ways that are difficult to assess [23]. The following sections develop encounter modeling, algorithmic abstractions, and evaluation workflows that explicitly connect these assumptions to measurable outcomes.

4. Encounter Modeling and Scenario Generation

Encounter modeling formalizes the stochastic processes that generate relative states between an ownship equipped with detect-and-avoid and one or more intruder aircraft. Let the relative state at time t be represented as a vector

$$z(t) = \begin{bmatrix} x(t) \\ [24]\, y(t) \\ h(t) \\ v_x(t) \\ v_y(t) \\ [25]\, v_h(t) \end{bmatrix}$$

where horizontal components describe relative position and velocity in a local frame and h denotes relative altitude. An encounter model specifies probability laws over initial states and over subsequent evolution of z(t) under assumptions on pilot behavior, flight plan structures, and environmental disturbances.

A practical approach is to generate initial relative positions and headings from traffic density fields and route structures, then propagate trajectories using simple kinematic or stochastic dynamics with bounded accelerations. For example, one may adopt a discrete-time evolution of the form [26]

$$z(t + \Delta t) = Fz(t) + w(t)$$

where F encodes constant-velocity or coordinated-turn dynamics, and w(t) represents bounded stochastic perturbations. To remain within the line-width constraint, F may be a simple block-diagonal operator, and w(t) a zero-mean random vector with compact support. Variants can incorporate turn rates, altitude capture profiles, and response to conflict alerts, with parameters drawn from empirically calibrated distributions. [27]

Scenario generation for testing detect-and-avoid systems must satisfy several properties. First, scenarios should cover a relevant range of miss distances, closure rates, approach angles, and vertical profiles, including cases that do not result in conflict to assess false alert tendencies. Second, they should represent both typical operational encounters and low-probability but safety-relevant geometries near the boundaries of detect-and-avoid capabilities. Third, scenarios must be constructed such that they can be instantiated consistently in both simulation and live experiments with controllable deviations.

Operationally relevant encounter sets can be defined in terms of constrained sampling regions in the space of initial relative states and closure geometries [28]. Let Ω denote the set of encounter initial conditions considered operationally plausible under a given concept of operations. Scenario generation then selects samples $\{z_0^i\}$ from Ω according to a measure that may be proportional to estimated encounter likelihoods or intentionally biased toward stressing conditions. In the latter case, importance sampling concepts can be applied so that performance metrics estimated from biased samples can be reweighted to approximate those under an unbiased encounter distribution, provided weights remain numerically stable.

For integration with live testing, each simulated encounter is associated with a realizable trajectory pair that respects dynamic limits and range airspace constraints. Mapping from abstract relative states to executable waypoints is non-trivial when accounting for wind, surveillance latency, and tracking errors [29]. Scenario design must incorporate margins that ensure the realized live trajectory remains within an acceptable deviation from the intended geometry. This implies a two-stage approach: generate an idealized encounter in relative coordinates, then solve for ownship and intruder reference trajectories that approximate this relative evolution subject to aircraft performance and safety constraints.

An important consideration is the representation of non-cooperative intruders. For such intruders, state information used by detect-and-avoid is derived from onboard sensors, potentially with intermittent detection and clutter. Scenario generation must include variability in detection probability, false tracks, and dropped tracks in a manner consistent with the sensor model used by the detect-and-avoid implementation [30]. The encounter model thus becomes a joint distribution over kinematics and measurement processes, forming the foundation for both simulation-based and live experiment evaluations that seek internal consistency.

5. Detect-and-Avoid Algorithmic Framework and Formalization

Detect-and-avoid systems can be abstracted as mappings from sensed trajectories to alert states and maneuver advisories. Let y(t) denote the sequence of measurements or tracks available to the system, including ownship and intruder states as estimated by onboard or ground-based sensors. The detect-and-avoid logic implements a decision function

$$a(t) = \Phi(y_{[0,t]})$$

where a(t) represents the current advisory or alert level and Φ is a causal mapping acting on the observation history [31]. This abstraction covers algorithms ranging from simple rule-based thresholds to stochastic optimization and model-predictive guidance.

Conflict prediction mechanisms typically model future relative motion under assumptions on both ownship and intruder behavior. Consider a simplified predictive model in which the relative state evolves according to

$$\hat{z}(t+\tau) = G(\tau)\,\hat{z}(t)$$

for a prediction horizon τ, where G encapsulates constant-velocity or bounded-acceleration assumptions. A loss-of-well-clear region W in the relative state space is defined according to horizontal and vertical distance thresholds and possibly temporal criteria [32]. The detect-and-avoid system declares a predicted loss-of-well-clear when

$$\exists \tau \in [0, T] : \hat{z}(t+\tau) \in W.$$

To limit expression width, W can be implicitly defined by short inequalities on relative distance and altitude, understood as part of the modeling context.

Probabilistic extensions model intruder trajectories as random processes and compute risk indicators based on predicted distributions [33]. Let $p(z, t)$ denote the probability density of relative state under these models. A simple probabilistic conflict indicator can be written as

$$R(t) = \int_0^T \int_W p(z, t+\tau)\, dz\, d\tau.$$

The detect-and-avoid logic may trigger alerts when R(t) exceeds calibrated thresholds [34]. This representation links algorithmic parameters to interpretable risk measures, enabling test methodologies to examine how variations in encounter geometry and sensor uncertainty map into R(t) values and subsequent advisory timelines.

Guidance logic determines recommended maneuvers given a detected or predicted conflict. Let u(t) denote a candidate ownship control command, constrained within an admissible set U that respects performance limits and mission constraints. A simplified cost function can be expressed as

$$J(u) = [35]\alpha\, C_{\mathrm{sep}}(u) + \beta\, C_{\mathrm{dev}}(u)$$

where C_{sep} penalizes proximity to W and C_{dev} penalizes deviation from the nominal trajectory or operational constraints, with non-negative weights α, β. The detect-and-avoid system selects guidance commands that aim to reduce J(u) while ensuring feasibility. Performance evaluation must inspect not only whether selected maneuvers maintain separation but also whether they remain consistent with operational norms, do not induce excessive oscillations, and allow for realistic human or autopilot execution.

To support systematic testing, the detect-and-avoid algorithm is further characterized by timing parameters, alerting logics, hysteresis, and filter settings. These parameters influence detection latency, sensitivity to noise, and stability of advisories. Formalization in terms of input-output properties allows one to define measurable quantities such as detection time, time margin to predicted loss-of-well-clear at first alert, rate of advisory changes, and minimal separation achieved under closed-loop execution [36]. These quantities serve as the interface between algorithmic description and test methodology, enabling both simulation and live experiments to be evaluated using common definitions.

6. Simulation-Based Performance Evaluation Methodologies

Simulation-based evaluation offers a controlled environment to explore detect-and-avoid performance over large sets of encounter geometries that may be impractical or unsafe to realize in flight. To

Table 10. *Simulation Metrics.*

Metric	Description	Use
Loss-of-well-clear	Boundary violation occurrence	Measures residual risk in scenarios
Alert time margin	Time before projected conflict at alert	Assesses responsiveness of logic

Table 11. *Simulation Modeling Ingredients.*

Ingredient	Role	Evaluation Aspect
Behavior models	Pilot or autopilot response patterns	Closed-loop realism
Sensor models	Noise, latency, dropouts	Effect on detection reliability

Table 12. *Live Scenario Execution.*

Element	Description	Evaluation Use
Planned geometry	Target relative trajectories	Basis for mapping to modeled encounters
Realized geometry	Measured flown trajectories	Validates behavior under real effects

Table 13. *Live Data and Safety Constructs.*

Aspect	Description	Role
Instrumentation	Reference positions and logs	Reconstructs advisory timelines
Safety buffers	Minimum separation margins	Constrains scenario severity

Table 14. *Evidence Integration Concepts.*

Concept	Description	Purpose
Class-conditional metrics	Scenario-based probabilities	Capture performance by encounter type
Combined sources	Simulation with live updates	Reflects both coverage and realism

maintain relevance, simulation frameworks must integrate encounter models, sensor models, detect-and-avoid logic, and vehicle dynamics in a manner that reflects key temporal and spatial scales. For each simulated encounter, the framework records whether alerts are issued, guidance is followed, and whether loss-of-well-clear or near-mid-air conditions occur, under predefined criteria. [37]

Let each encounter i be represented by a trajectory pair and associated measurements, and let X_i denote a binary indicator that a specified safety event occurs, for example a loss-of-well-clear. For N simulated encounters drawn (possibly with biasing) from an encounter distribution, an empirical estimate of event probability is

$$\hat{p} = \frac{1}{N} \sum_{i=1}^{N} X_i .$$

Table 15. *Risk Structuring Elements.*

Element	Description	Evaluation Role
Factorized risk	Decomposed collision probability	Situates detect-and-avoid among mitigations
Uncertainty treatment	Intervals and sensitivity checks	Expresses robustness of conclusions

Table 16. *Synthesized Perspective.*

Theme	Summary	Implication
Combined evidence	Joint use of simulation and live data	Supports interpretable performance view
Stable framework	Explicit models and metrics	Enables adaptation to evolving operations

If biased encounter sampling is used with importance weights w_i, a weighted estimator can be expressed as

$$\hat{p}_w = \frac{\sum_{i=1}^{N} w_i X_i}{\sum_{i=1}^{N} w_i}.$$

Simulation methodologies must control numerical variance of such estimators by balancing coverage of rare but critical scenarios with stable weighting schemes.

Performance evaluation extends beyond single event probabilities [38]. Metrics include the distribution of time margins at initial alert, minimal achieved separation conditioned on an alert, the frequency of nuisance alerts in non-threatening encounters, and the interaction of multiple intruders. To capture these, the simulation framework records continuous variables such as time to closest point of approach, instantaneous separation at advisory issuance, and number of advisory reversals. Let T_i denote a time margin for encounter i; one may compute empirical cumulative distributions and confidence bands to assess how often detect-and-avoid provides margins above specified thresholds.

To introduce advanced modeling while respecting line width constraints, consider a simple hazard rate representation for losses of well clear. Define a non-negative process $\lambda(t)$ representing instantaneous hazard given the state and detect-and-avoid actions [39]. An approximate relationship between hazard and cumulative probability of loss-of-well-clear over a horizon T can be written as

$$P_{\text{LoWC}} \approx \int_0^T \lambda(t)\, dt.$$

In simulation, $\lambda(t)$ may be approximated using regression or classification models trained on trajectory features and advisory histories. The evaluation then examines how algorithm configurations alter $\lambda(t)$ profiles, without attributing all risk variation solely to detect-and-avoid.

An additional aspect is modeling execution variability. Human-in-the-loop or autopilot-in-the-loop responses to advisories can be represented by stochastic delays and deviations from commanded maneuvers [40]. For each encounter, the closed-loop trajectory thus depends on both detect-and-avoid outputs and sampled execution parameters. Denote by θ a vector of behavioral parameters, with distribution informed by experimental data or conservative assumptions. Simulation then estimates performance metrics integrated over θ, which can be expressed conceptually as

$$\hat{p} = \frac{1}{N}\sum_{i=1}^{N} g(z_0^i, \theta_i)\,[41]$$

where g is an indicator of the safety event outcome. This formulation underscores that evaluation outcomes reflect combined uncertainties in encounters and behavioral responses, motivating transparent documentation of assumptions.

Ultimately, simulation-based methods are expected to provide broad coverage and sensitivity analyses rather than definitive performance guarantees in isolation. Their role is to explore relationships between detect-and-avoid design parameters and observable metrics, to map regions of adequate performance, and to identify scenarios that warrant targeted live testing. Structured encounter sets and reproducible simulation configurations ensure that results can be inspected, challenged, and updated as system designs and operational concepts evolve. [42]

7. Live Testing Methodologies for Detect-and-Avoid Systems

Live testing complements simulation by exposing detect-and-avoid systems to real sensor phenomenology, communication effects, and unmodeled dynamics. Because safety constraints limit the degree of closeness and aggressiveness achievable in flight, live methodologies require careful design so that collected data are informative while margins to unsafe conditions are maintained. This is typically achieved by using piloted or uncrewed surrogate intruders, predefined geometries, and real-time safety monitors that can override or constrain trajectories.

A live encounter scenario is defined by planned trajectories for ownship and intruder, with initial conditions chosen to approximate target relative states from the encounter model. Let $z_{\text{plan}}(t)$ denote the planned relative state and $z_{\text{act}}(t)$ the realized relative state. Deviations arise from wind, navigation errors, pilot execution, and control system limitations [43]. Performance evaluation must account for such deviations explicitly, for example by defining tolerances within which an encounter is considered valid for a given target class. If deviations exceed tolerances, the data can still inform sensor and tracking performance but may require careful handling before being used for evaluating alerting logic.

Instruments used during live tests include high-fidelity position reference systems, datalink logs, onboard detect-and-avoid outputs, cockpit video, and ground-based radar or optical tracking. Synchronization of these data sources is essential to reconstruct the time history of relative states and advisory decisions. Let t_k denote discrete time stamps at which all relevant quantities are aligned [44]. At each t_k, one may record detect-and-avoid state variables such as alert level, predicted time to loss-of-well-clear, and suggested maneuver vector, as well as ground-truth relative position derived from reference sensors. These aligned records form the basis for computing live analogues of the metrics defined in the simulation environment.

An important methodological element is safety case compatibility. Live testing is performed under constraints that ensure predicted separation remains above specified minima with buffers, even if detect-and-avoid or pilot responses do not perform as expected [45]. As a result, actual mid-air collision risk during testing is kept far below operational tolerances, and many scenarios are intentionally truncated or diverted when safety margins decrease. Performance evaluation must interpret advisory timelines in light of these interventions. For example, if a safety pilot intervenes and modifies the ownship trajectory, one can still assess whether detect-and-avoid alerts up to that point were timely relative to what would have been required in the absence of intervention.

To represent the linkage between planned and realized encounters, consider a mapping that classifies each executed test into a nominal scenario label based on realized parameters. Let S denote a set of scenario bins defined over relative approach angle, initial range, closure rate, and altitude [46]. For each executed encounter j, one estimates a tuple of realized parameters and assigns it to an element of S if within tolerance. Performance metrics can then be computed conditional on these bins, providing a structured view of detect-and-avoid behavior across the realized live scenario space, even if it does not exactly match the originally planned encounter set.

Live testing also reveals practical issues less evident in simulation, including track initiation latency, dropouts, interference between multiple sensors, and human factors in interpreting alerts. While these effects are often scenario-specific, systematically logging their occurrences allows for characterization of robustness. For instance, one may compute empirical frequencies of mis-correlated tracks or of alert oscillations under certain geometries [47]. Though sample sizes are typically limited, such observations can guide refinements of both detect-and-avoid logic and supporting models used in simulation.

8. Statistical Performance Assessment and Safety Argumentation

Combining simulated and live encounter results into an integrated performance assessment requires statistical methods that acknowledge differences in sample size, encounter coverage, and fidelity. Let $\mathcal{E}_s$ and $\mathcal{E}_l$ denote the sets of simulated and live encounters respectively, with associated outcomes on events of interest such as timely alert, loss-of-well-clear, or adherence to guidance. For a given scenario class c in a partition of encounter space, define indicator variables $X_{i,c}$ for each encounter i. One may then estimate class-conditional probabilities

$$\hat{p}_{s,c} = \frac{\sum_{i \in \mathcal{E}_s} X_{i,c}}{|\{i \in \mathcal{E}_s : i \in c\}|}$$

and analogously $\hat{p}_{l,c}$ for live data when sufficient samples exist.

In many cases, live data are sparse relative to the dimensionality of scenario space. A practical strategy is to use simulation-based estimates as priors and update them with live evidence in a Bayesian framework [48]. Without overcomplicating notation, consider a beta-binomial representation for a binary metric in a given class. Denote prior parameters by α_0, β_0 informed by simulation, and observed live outcomes by k successes out of n trials. The posterior parameters become

$$\alpha = [49]\alpha_0 + k, \quad \beta = \beta_0 + n - k.$$

From these, one can derive posterior intervals for the performance metric that blend simulated expectations with live observations. The choice of prior strength relative to live data reflects confidence in simulation fidelity and should be explicitly documented. [50]

For metrics involving continuous quantities such as alert time margins, non-parametric or distributional matching techniques may be used. One approach is to compare empirical distributions from simulation and live data within overlapping scenario regions and compute discrepancy measures. Significant discrepancies may prompt refinement of sensor or behavior models in simulation, or separate treatment of certain operational regimes. The aim is not to force agreement but to understand when simulation misrepresents conditions that materially affect detect-and-avoid performance.

To connect performance metrics with safety objectives, one constructs models of airspace-level risk that incorporate encounter rates, detect-and-avoid effectiveness, and other mitigations [51]. A simplified expression for expected collision probability per encounter, conditioned on detect-and-avoid operation, can be structured as a product of factors representing steps in the defense-in-depth chain. For example,

$$P_{\text{col}} \approx q_{\text{enc}}\, q_{\text{threat}}\, q_{\text{DAAfail}}\, q_{\text{residual}},$$

where each q denotes a conditional probability: occurrence of an encounter within a defined proximity, evolution of that encounter into a threat without intervention, failure of detect-and-avoid to mitigate given a threat, and residual failure of any remaining mechanisms. Detect-and-avoid evaluations primarily inform q_{DAAfail} and related terms, while recognizing that the other factors are influenced by traffic management, equipage, and operational constraints. This structure helps avoid overstating the contribution or necessity of detect-and-avoid by situating it among other elements.

Confidence in these estimates is represented through intervals rather than single-point values [52]. For instance, an upper bound on q_{DAAfail} can be obtained from binomial confidence limits using aggregated evidence across relevant scenario classes. If no failures are observed in n suitably representative encounters, a conservative bound can be expressed in a compact form based on standard approximations, remaining within the width constraint. These bounds can then be propagated through the risk expression to derive upper bounds on collision probabilities under stated assumptions.

An additional consideration is robustness with respect to uncertainties in encounter models and behavior. Sensitivity analysis can be performed by varying key parameters within plausible ranges and recomputing performance and risk metrics [53]. While comprehensive exploration is limited by dimensionality, targeted variations in closure rates, latency, and compliance assumptions can reveal whether detect-and-avoid performance degrades sharply near certain boundaries. When such sensitivities are identified, operational constraints or algorithmic adjustments may be considered, and their effects re-evaluated using the same framework.

Overall, statistical assessment and safety argumentation benefit from a clear separation between observable quantities in tests, intermediate performance metrics, and derived risk indicators. By expressing relationships through concise mathematical formulations and by explicitly stating modeling assumptions, the evaluation remains open to revision as additional data and improved models become available, without requiring fundamental changes to the underlying methodology.

9. Conclusion

Test and performance evaluation methods for uncrewed aircraft detect-and-avoid systems must reconcile the need for broad scenario coverage with the constraints of safe and practical experimentation [54]. Relying solely on simulation obscures real-world sensor and integration effects, whereas relying solely on live testing limits coverage and can under-sample critical edge cases. A combined approach, grounded in explicit encounter modeling, algorithmic formalization, and statistically coherent integration of evidence, provides a structured path to characterizing detect-and-avoid behavior across simulated and live scenarios.

The discussion has outlined how encounter models can be constructed in terms of relative states, stochastic dynamics, and measurement processes, allowing systematic generation of scenarios that are both operationally meaningful and suitable for implementation in simulations and flight tests. Detect-and-avoid functions have been abstracted as mappings from observation histories to advisories, with associated predictive and guidance components expressed through interpretable quantities such as risk integrals and cost functions. This abstraction supports consistent definitions of performance metrics, including detection timeliness, false and nuisance alert frequencies, and minimal achieved separations under closed-loop response. [55]

Simulation-based methodologies enable exploration of large encounter sets, controlled variation of algorithm parameters, and analysis of sensitivities to sensor and behavior assumptions. Live testing methodologies incorporate instrumentation, scenario binning, and explicit treatment of deviations between planned and realized encounters, yielding data that expose integration issues and provide direct observations of system performance in realistic environments. Statistical frameworks that combine these sources of evidence can generate bounded estimates of detect-and-avoid effectiveness and its contribution to collision risk, while explicitly reflecting uncertainties in models and data.

The resulting perspective does not treat detect-and-avoid as a singular guarantee of safety but as one component within a layered architecture of mitigations, whose performance can be characterized using transparent and adaptable methods. As operational concepts, technologies, and regulatory expectations evolve, the described framework can be refined by updating encounter models, incorporating new sensor characteristics, and extending data sets, without altering the core principles by which simulated and live encounter evidence are used to assess detect-and-avoid systems [56].

References

[1] R. Szabolcsi and J. Menyhárt, "Loads affecting ugvs' technical status," *Review of the Air Force Academy*, vol. 13, pp. 15–20, 12 2015.

[2] J. R. M. de Dios, A. de San Bernabe, A. Viguria, A. Torres-González, and A. Ollero, "Combining unmanned aerial systems and sensor networks for earth observation," *Remote Sensing*, vol. 9, pp. 336–, 4 2017.

[3] J. Stutz, B. Werner, M. Spolaor, L. Scalone, J. Festa, C. Tsai, R. Cheung, S. F. Colosimo, U. Tricoli, R. Raecke, R. Hossaini, M. P. Chipperfield, W. Feng, R.-S. Gao, E. J. Hintsa, J. W. Elkins, F. L. Moore, B. C. Daube, J. V. Pittman, S. C. Wofsy, and K. Pfeilsticker, "A new differential optical absorption spectroscopy instrument to study atmospheric chemistry from a high-altitude unmanned aircraft," *Atmospheric Measurement Techniques*, vol. 10, pp. 1017–1042, 3 2017.

[4] Z. P. Waller, "Five factor model personality profiles of university of north dakota unmanned aircraft systems students," *Journal of Unmanned Aerial Systems*, vol. 1, 4 2015.

[5] U. K. Verfuss, A. S. Aniceto, D. Harris, D. Gillespie, S. Fielding, G. Jimenez, P. Johnston, R. R. Sinclair, A. Sivertsen, S. Solbo, R. Storvold, M. Biuw, and R. Wyatt, "A review of unmanned vehicles for the detection and monitoring of marine fauna," *Marine pollution bulletin*, vol. 140, pp. 17–29, 1 2019.

[6] O. Erenoglu, "Uav for 3d morphological mapping applications: A case study of koru mining site, canakkale, nw turkey," *Tehnicki vjesnik - Technical Gazette*, vol. 28, pp. 1044–1050, 6 2021.

[7] C. Vermeulen, P. Lejeune, J. Lisein, P. Sawadogo, and P. Bouché, "Unmanned aerial survey of elephants.," *PloS one*, vol. 8, pp. e54700–, 2 2013.

[8] D. Wierzbicki, "Multi-camera imaging system for uav photogrammetry.," *Sensors (Basel, Switzerland)*, vol. 18, pp. 2433–, 7 2018.

[9] E. Levandovsky, "Mapping burn extent of large wildland fires from satellite imagery using machine learning trained from localized hyperspatial imagery," *Remote Sensing*, vol. 12, pp. 4097–, 12 2020.

[10] D. Azimov and J. S. Allen, "Analytical model and control solutions for unmanned aerial vehicle maneuvers in a vertical plane," *Journal of Intelligent & Robotic Systems*, vol. 91, pp. 725–733, 9 2017.

[11] N. G. M, Y. K. Guruprasad, Y. Teekaraman, R. Kuppusamy, and A. R. Thelkar, "Generative adversarial networks for unmanned aerial vehicle object detection with fusion technology," *Journal of Advanced Transportation*, vol. 2022, pp. 1–13, 4 2022.

[12] P. Royo, E. Pastor, C. Barrado, R. Cuadrado, F. Barrao, and A. Garcia, "Hardware design of a small uas helicopter for remote sensing operations," *Drones*, vol. 1, pp. 3–, 9 2017.

[13] P.-C. Shao, "Risk assessment for uas logistic delivery under uas traffic management environment," *Aerospace*, vol. 7, pp. 140–, 9 2020.

[14] K.-P. L. Vu, R. C. Rorie, L. Fern, and R. J. Shively, "Human factors contributions to the development of standards for displays of unmanned aircraft systems in support of detect-and-avoid:," *Human factors*, vol. 62, pp. 505–515, 4 2020.

[15] P. J. Sanchez-Cuevas, P. Ramon-Soria, B. C. Arrue, A. Ollero, and G. Heredia, "Robotic system for inspection by contact of bridge beams using uavs," *Sensors (Basel, Switzerland)*, vol. 19, pp. 305–, 1 2019.

[16] B. R. Greene, A. R. Segales, T. M. Bell, E. A. Pillar-Little, and P. B. Chilson, "Environmental and sensor integration influences on temperature measurements by rotary-wing unmanned aircraft systems.," *Sensors (Basel, Switzerland)*, vol. 19, pp. 1470–, 3 2019.

[17] F. Nikodem and S. Kaltenhäuser, "Operation and operation approval of high-altitude platforms," *CEAS Aeronautical Journal*, vol. 11, pp. 1037–1044, 9 2020.

[18] R. P. Sishodia, R. L. Ray, and S. K. Singh, "Applications of remote sensing in precision agriculture: A review," *Remote Sensing*, vol. 12, pp. 3136–, 9 2020.

[19] N. A. Ubina and S.-C. Cheng, "A review of unmanned system technologies with its application to aquaculture farm monitoring and management," *Drones*, vol. 6, pp. 12–12, 1 2022.

[20] S. Al-Janabi, A. Alkaim, and A. Rahem, "An alternative technique to reduce time, cost and human effort during natural or manufactured disasters," *IAES International Journal of Robotics and Automation (IJRA)*, vol. 11, pp. 10–10, 3 2022.

[21] C. J. Fritsch and C. T. Downs, "Evaluation of low-cost consumer-grade uavs for conducting comprehensive high-frequency population censuses of hippopotamus populations," *Conservation Science and Practice*, vol. 2, 10 2020.

[22] U. Jahn, D. Hes, M. Stampa, A. Sutorma, C. Röhrig, P. J. Schulz, and C. Wolff, “A taxonomy for mobile robots: Types, applications, capabilities, implementations, requirements, and challenges,” *Robotics*, vol. 9, pp. 109–, 12 2020.

[23] S. D. Schrijver, “Commercial use of drones: Commercial drones facing legal turbulence: Towards a new legal framework in the eu,” *US-China Law Review*, vol. 16, 8 2019.

[24] A. C. Canolla, M. B. Jamoom, and B. Pervan, “Unmanned aircraft systems detect and avoid sensor hybrid estimation error analysis,” in *17th AIAA Aviation Technology, Integration, and Operations Conference*, p. 4384, 2017.

[25] C. D. Rodin, F. A. de Alcantara Andrade, A. R. Hovenburg, and T. A. Johansen, “A survey of practical design considerations of optical imaging stabilization systems for small unmanned aerial systems.,” *Sensors (Basel, Switzerland)*, vol. 19, pp. 4800–, 11 2019.

[26] D. Buscombe and A. C. Ritchie, “Landscape classification with deep neural networks,” *Geosciences*, vol. 8, pp. 244–, 7 2018.

[27] J. Heard, P. Baskaran, and J. A. Adams, “Predicting task performance for intelligent human-machine interactions.,” *Frontiers in neurorobotics*, vol. 16, pp. 973967–, 9 2022.

[28] P. Bauer, A. Hiba, J. Bokor, and A. Zarandy, “Three dimensional intruder closest point of approach estimation based-on monocular image parameters in aircraft sense and avoid,” *Journal of Intelligent & Robotic Systems*, vol. 93, pp. 261–276, 3 2018.

[29] L. Jurjević, M. Gašparović, A. S. Milas, and I. Balenović, “Impact of uas image orientation on accuracy of forest inventory attributes,” *Remote Sensing*, vol. 12, pp. 404–, 1 2020.

[30] A. Barnawi, K. Kumar, N. Kumar, N. Thakur, B. Alzahrani, and A. Almansour, “Unmanned ariel vehicle (uav) path planning for area segmentation in intelligent landmine detection systems.,” *Sensors (Basel, Switzerland)*, vol. 23, pp. 7264–7264, 8 2023.

[31] A. C. Lee, M. Dahan, A. Weinert, and S. Amin, “Leveraging suas for infrastructure network exploration and failure isolation,” *Journal of Intelligent & Robotic Systems*, vol. 93, pp. 385–413, 4 2018.

[32] C. M. Eaton, E. K. P. Chong, and A. A. Maciejewski, “Multiple-scenario unmanned aerial system control: A systems engineering approach and review of existing control methods,” *Aerospace*, vol. 3, pp. 1–, 1 2016.

[33] A. Canolla, M. B. Jamoom, and B. Pervan, “Interactive multiple model sensor analysis for unmanned aircraft systems (uas) detect and avoid (daa),” in *2018 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, pp. 757–766, IEEE, 2018.

[34] C. Nansen and N. C. Elliott, “Remote sensing and reflectance profiling in entomology,” *Annual review of entomology*, vol. 61, pp. 139–158, 3 2016.

[35] X. Bai, M. Yang, and B. Zhao, “Image-based displacement monitoring when considering translational and rotational camera motions,” *International Journal of Civil Engineering*, vol. 20, pp. 1–13, 7 2021.

[36] M. Doukari, M. Batsaris, A. Papakonstantinou, and K. Topouzelis, “A protocol for aerial survey in coastal areas using uas,” *Remote Sensing*, vol. 11, pp. 1913–, 8 2019.

[37] F. Ahmed and M. Jenihhin, “A survey on uav computing platforms: A hardware reliability perspective.,” *Sensors (Basel, Switzerland)*, vol. 22, pp. 6286–6286, 8 2022.

[38] Y. Lee, J. Lee, and J.-W. Lee, “Holding area conceptual design and validation for various urban air mobility (uam) operations: A case study in seoul–gyungin area,” *Applied Sciences*, vol. 11, pp. 10707–, 11 2021.

[39] F. López-Granados, J. Torres-Sánchez, A.-I. de Castro, A. Serrano-Pérez, F. J. Mesas-Carrascosa, and J. M. Peña, “Object-based early monitoring of a grass weed in a grass crop using high resolution uav imagery,” *Agronomy for Sustainable Development*, vol. 36, pp. 67–, 11 2016.

[40] J. A. García-Berná, S. Ouhbi, B. Benmouna, G. García-Mateos, J. L. Fernández-Alemán, and J. M. Molina-Martínez, “Systematic mapping study on remote sensing in agriculture,” *Applied Sciences*, vol. 10, pp. 3456–, 5 2020.

[41] N. E. Onn, “The principle of ad coelom in respect to the low-altitude airspace for the commercial unmanned aerial vehicles (uavs) in malaysia,” *International Journal of Law, Government and Communication*, vol. 7, pp. 110–123, 12 2022.

[42] J. M. de Dios, K. Lferd, A. de San Bernabe, G. Núñez, A. Torres-González, and A. Ollero, "Cooperation between uas and wireless sensor networks for efficient data collection in large environments," *Journal of Intelligent & Robotic Systems*, vol. 70, pp. 491–508, 8 2012.

[43] O. LYUBICH and O. STRYZHAK, "Transdisciplinary consolidation of financial technologies with knowledge systems," *Fìnansi Ukraïni*, vol. 2021, pp. 54–61, 12 2021.

[44] Y. Lyu, Q. Pan, J. Hu, and C. Zhao, "Obstacle avoidance under relative localization uncertainty," *Science China Information Sciences*, vol. 62, pp. 084201–, 3 2019.

[45] O. K. Sahingoz, "Networking models in flying ad-hoc networks (fanets): Concepts and challenges," *Journal of Intelligent & Robotic Systems*, vol. 74, pp. 513–527, 10 2013.

[46] A. Saleh, M. A. Zulkifley, H. H. Harun, F. Gaudreault, I. Davison, and M. Spraggon, "Forest fire surveillance systems: A review of deep learning methods.," *Heliyon*, vol. 10, pp. e23127–e23127, 12 2023.

[47] E. J. Jensen, L. Pfister, D. E. Jordan, T. V. Bui, R. Ueyama, H. B. Singh, T. Thornberry, A. W. Rollins, R. S. Gao, D. W. Fahey, K. H. Rosenlof, J. W. Elkins, G. S. Diskin, J. P. DiGangi, R. P. Lawson, S. Woods, E. Atlas, M. A. N. Rodriguez, S. C. Wofsy, J. V. Pittman, C. G. Bardeen, O. B. Toon, B. C. Kindel, P. A. Newman, M. J. McGill, D. L. Hlavka, L. R. Lait, M. R. Schoeberl, J. W. Bergman, H. B. Selkirk, M. J. Alexander, J.-E. Kim, B. Lim, J. Stutz, and K. Pfeilsticker, "The nasa airborne tropical tropopause experiment: High-altitude aircraft measurements in the tropical western pacific," *Bulletin of the American Meteorological Society*, vol. 98, pp. 129–143, 1 2017.

[48] D. H. Andrews, "Training research and development: Retrospective of a career in the defense department," *Education Review*, vol. 24, 2 2017.

[49] D. Johnston, "Unoccupied aircraft systems in marine science and conservation," *Annual review of marine science*, vol. 11, pp. 439–463, 7 2018.

[50] N. Soulakellis, S. Chatzistamatis, C. Vasilakos, G. Tataris, A. Papakonstantinou, D. Kavroudakis, K. Topouzelis, O. Roussou, C. Kalloniatis, E.-E. Papadopoulou, K. Chaidas, and P. Kalaitzis, "Synergistic exploitation of geoinformation methods for post-earthquake 3d mapping of vrisa traditional settlement, lesvos island, greece," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLII-3/W4, pp. 491–498, 3 2018.

[51] M.-A. Stamate, C. Pupăză, F.-A. Nicolescu, and C.-E. Moldoveanu, "Improvement of hexacopter uavs attitude parameters employing control and decision support systems.," *Sensors (Basel, Switzerland)*, vol. 23, pp. 1446–1446, 1 2023.

[52] N. K. Sato, T. Tsuji, Y. Iijima, N. Sekiya, and K. Watanabe, "Predicting rice lodging risk from the distribution of available nitrogen in soil using uas images in a paddy field.," *Sensors (Basel, Switzerland)*, vol. 23, pp. 6466–6466, 7 2023.

[53] A. Islam, A. Shankar, A. L. Houston, and C. Detweiler, "University of nebraska unmanned aerial system (uas) profiling during the lapse-rate field campaign," *Earth System Science Data*, vol. 13, pp. 2457–2470, 6 2021.

[54] Z. Birnbaum, A. Dolgikh, V. A. Skormin, E. O'Brien, D. Muller, and C. Stracquodaine, "Unmanned aerial vehicle security using recursive parameter estimation," *Journal of Intelligent & Robotic Systems*, vol. 84, pp. 107–120, 10 2015.

[55] H. Kim, Y. W. Jung, and H. Zhang, "Guest editorial special issue on time-sensitive networks for unmanned aircraft systems.," *Sensors (Basel, Switzerland)*, vol. 21, pp. 6132–, 9 2021.

[56] W. Deigele, M. Brandmeier, and C. Straub, "A hierarchical deep-learning approach for rapid windthrow detection on planetscope and high-resolution aerial image data," *Remote Sensing*, vol. 12, pp. 2121–, 7 2020.