

Article

Vision-Based Dual-Mode Collision Risk-Warning for Aircraft Apron Monitoring

Emre Can Bingol *, Hamed Al-Rawashidy and Konstantinos Banitsas

Department of Electronic and Electrical Engineering, Brunel University of London, London UB8 3PH, UK; konstantinos.banitsas@brunel.ac.uk (K.B.)

* Correspondence: emreca.bingol@brunel.ac.uk

Highlights

What are the main findings?

- Under identical detector inputs (optimised YOLOv8-Seg) and without tracker specific tuning, DeepSORT delivered the most stable identity tracking on the 997-frame Microsoft Flight Simulator (MSFS) simulation-based incident reenactment benchmark using the airplane-only MOTChallenge ground truth: Multi-Object Tracking Accuracy (MOTA) 92.77%, recall 93.27%, and one ID switch.
- A dual-mode incident-warning framework was developed: (i) a reactive module based on segmentation-mask proximity and (ii) a proactive module based on short-horizon trajectory extrapolation and future-Intersection-over-Union (IoU) risk triggering. The modules can be used independently or jointly.

What are the implications of the main findings?

- The MSFS reenactment sequence and its associated labels provide a reproducible testbed that helps mitigate the scarcity of annotated apron-incident data for detection, tracking and risk studies.
- A scaled Unmanned Aerial Vehicle (UAV)/laboratory validation protocol is defined to assess end-to-end feasibility on UAV-captured imagery (reported qualitatively via representative frames and warning overlays).

Abstract

Ground incidents on airport aprons can cause substantial operational disruption and economic loss, while conventional surveillance (e.g., Surface Movement Radar (SMR), Closed-Circuit Television (CCTV)) often lacks the resolution and proactive decision support required for close-proximity operations. This study proposes a UAV-deployable, camera-agnostic Computer Vision (CV) framework for collision-risk warning from elevated viewpoints. An optimised YOLOv8-Seg backbone performs multi-class aircraft segmentation (airplane, wing, nose, tail, and fuselage) and is integrated with four MOT algorithms under identical evaluation settings. For quantitative tracker benchmarking, DeepSORT provides the strongest overall performance on the airplane-only MOTChallenge-format ground truth (MOTA 92.77%, recall 93.27%). To mitigate the scarcity of annotated apron-incident data, a labelled 997-frame MOT dataset is created via an MSFS simulation-based reenactment inspired by the 2018 Asiana–Turkish Airlines wing-to-tail event at Istanbul Ataturk Airport. The framework further introduces a dual-module warning mechanism that can operate independently: (i) a reactive module using image-plane proximity derived from segmentation masks, and (ii) a proactive module that predicts short-horizon conflicts via trajectory extrapolation and IoU-based future overlap analysis. The approach is evaluated on multiple simulated incident scenarios

Academic Editor: San Jiang

Received: 30 January 2026

Revised: 23 February 2026

Accepted: 27 February 2026

Published: 2 March 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/).

and assessed on a real apron video from Hong Kong International Airport; additionally, laboratory-scale UAV experiments using diecast aircraft models provide end-to-end feasibility evidence on unmanned-platform imagery. Overall, the results indicate timely warnings and practical feasibility for low-overhead UAV-enabled apron monitoring.

Keywords: Unmanned Aerial Systems; collision-risk warning; apron safety; simulation-based reenactment; YOLOv8-Seg; DeepSORT; multi-object tracking; aircraft tracking; trajectory prediction; computer vision; UAV monitoring

1. Introduction

Air transport demand continues to increase, intensifying airport surface operations and placing greater pressure on apron safety. The International Air Transport Association (IATA) projects global airline passenger numbers will reach about 5.2 billion in 2026 [1], leading to denser aircraft movements in constrained ground environments. Although flight safety has improved significantly over recent decades, aircraft ground operations remain vulnerable to ramp incidents, such as wing-to-wing, wing-to-tail or wing-to-nose incidents that result in structural damage, operational disruption, and substantial economic cost. Without additional preventive measures, the annual financial impact of ground damage is expected to approach USD 10 billion by 2035 [2].

To manage these risks, airports employ a range of surveillance and monitoring technologies. Surface Movement Radar (SMR) supports wide area tracking under adverse weather conditions, yet its spatial resolution is often insufficient for precise separation in congested apron areas [3]. Cooperative systems such as Automatic Dependent Surveillance–Broadcast (ADS-B) provide reliable state information for equipped aircraft, but they neither detect non-cooperative objects nor enable component-level geometric proximity analysis between aircraft [4–6]. While Light Detection and Ranging (LiDAR) and ultrasonic sensing offer accurate local measurements [7], their large-scale deployment is constrained by cost and infrastructure requirement. Consequently, visual surveillance via camera networks remains an indispensable component for verifying ground situations, complementing radar data where coverage gaps exist.

Optical camera systems such as Closed-Circuit Television (CCTV) are widely deployed at airports and offer detailed visual coverage of apron operations; however, they operate in a largely passive manner, rely heavily on continuous human monitoring, and suffer performance degradation under adverse weather or low-visibility conditions [8]. These limitations necessitate the integration of intelligent video analytics [9]. Advances in Computer Vision (CV) and Deep Learning (DL) have enabled automated detection and tracking of apron objects, reducing operator workload and improving consistency [10,11]. Prior research has applied CV-based methods to tasks such as Foreign Object Debris (FOD) detection [12–14], perimeter surveillance [15], and aircraft turnaround monitoring [16]. Nevertheless, most existing vision-based approaches remain limited to detection or short-term tracking and provide little capability for anticipating imminent spatial conflicts between moving aircraft.

To directly address these limitations, this study presents an Unmanned Aerial Vehicle (UAV)-deployable, vision-based incident-risk warning framework for apron environments. The system moves beyond static detection by integrating robust Multi-Object Tracking (MOT) with predictive risk analysis, and it is designed for overhead monitoring geometry relevant to UAV-based inspection (emulated via MSFS incident reenactment and assessed on elevated real-world video). Building upon our previous work on optimised aircraft component segmentation [17], in which we conducted a comparative evaluation of multiple detector families (including newer YOLO variants) and segmentation

configurations on the same aircraft/component dataset, we selected YOLOv8-Seg as the most suitable perception backbone for apron aircraft and component delineation. Accordingly, in this paper we keep the detector fixed (optimised YOLOv8-Seg) to isolate the effects of MOT selection and the proposed reactive/proactive warning logic under identical detection outputs. The proposed solution introduces a dual-layer safety strategy and is evaluated through a combination of MSFS simulation scenarios, real-world apron videos (no-incident control), and laboratory-scale UAV validation experiments.

1.1. Motivation and Research Questions

Apron operations can still produce clearance conflicts and ground-contact incidents despite layered surveillance and standard operating procedures. Current monitoring is often descriptive rather than predictive, which is problematic in close-proximity manoeuvres with occlusions and complex taxi/pushback dynamics. This motivates a practical, low-cost supplementary safety layer from elevated viewpoints. Table 1 lists selected publicly reported ground incidents (2018–2025) with airport identifiers (IATA), underscoring persistent separation-awareness challenges [18].

Table 1. Notable aircraft ground incidents involving collision and structural damage (2018–2025).

Date	IATA/Airport	Involved Aircraft/Operators	Incident Type	Reported Context	Ref.
5 January 2018	YYZ/Toronto Pearson International Airport, Canada	Sunwing Boeing 737-800 vs. WestJet Boeing 737-800 (taxi)	Wing-to-tail contact; post-impact fire	Collision during taxiing; fuel leak led to tail-section fire; minor injury reported	[19]
13 May 2018	IST/Istanbul Ataturk Airport, Türkiye	Asiana Airlines Airbus A330 vs. Turkish Airlines Airbus A321 (parked)	Wing-to-tail (vertical stabiliser) collision	Taxiing aircraft struck the vertical stabiliser of a parked aircraft; both grounded for inspection/repair	[20]
13 February 2019	AMS/Amsterdam Schiphol International Airport, Netherlands	Boeing 747 vs. Boeing 787	Wing-to-horizontal stabiliser collision during pushback	During pushback, a B747 wingtip struck a B787 stabiliser due to non-standard positioning, poor visibility, and communication failure; damage occurred	[21]
22 May 2019	IST/Istanbul Airport (IGA), Türkiye	Turkish Airlines Boeing 777-300ER	Wingtip contact with infrastructure	Wing struck a lighting pole during taxi; no injuries reported	[22]
10 February 2020	ZRH/Zurich Airport, Switzerland	Qatar Airways Airbus A350-900 vs. Helvetic Airways Embraer ERJ-190 (parked)	Wing-to-tail (rudder) collision during pushback	During pushback, left wing contacted the rudder of a parked aircraft; both removed from service for repairs	[23]
21 May 2021	MDW/Chicago Midway International Airport, USA	Southwest B737-700 vs. Southwest B737-800	Ground collision during taxi (near gate area)	During gate-area taxiing, a B737-800 winglet struck a stopped B737-700 stabiliser with insufficient clearance; aircraft damage resulted, with no injuries	[24]
28 September 2022	LHR/London Heathrow Airport, UK	Korean Air Boeing 777-300ER vs. Icelandair Boeing 757-256	Wing-to-tail (rudder) collision during taxi	Taxiing aircraft wing collided with the rudder of another aircraft; damage to wingtip and rudder; no injuries reported	[25]
4 October 2023	STN/London Stansted Airport, UK	Ryanair Boeing 737-800 vs. ground service vehicle	Aircraft wing-to-vehicle collision	During taxiing, vehicle driver did not notice approaching aircraft in time; damage to aircraft wing and vehicle roof	[26]
6 April 2024	LHR/London Heathrow Airport, UK	Virgin Atlantic Boeing 787-9 (towed) vs. British Airways Airbus A350	Wingtip-to-wingtip collision during towing	Attributed to manoeuvring error by tug operator while towing the idle aircraft	[27]
8 January 2025	ORD/Chicago O'Hare International Airport, USA	American Airlines Boeing 737-800 vs. United Airlines Boeing 787-10	Wingtip-to-tail collision during taxi	Taxiing aircraft wing collided with the tail of another taxiing aircraft; no injuries reported; both grounded for checks	[28]
10 April 2025	DCA/Ronald Reagan National Airport, USA	American Airlines Bombardier CRJ900 vs. American Airlines Embraer E175	Wingtip collision during taxi	Wing contacted another aircraft wing; both taken out of service	[29]

Addressing this need, our previous work established a robust perception backbone by benchmarking aircraft component detection models and optimising the selected approach [17]. This study advances that perception foundation to the next stage by generating real-time early warnings through reliable MOT integration with reactive and proactive risk assessment. In this context, platform-agnostic approaches that combine CV/DL perception with tracking and predictive safety logic, validated against realistic scenarios, are essential for scalable apron surveillance solutions.

To systematically address these challenges and validate the proposed framework, this study is guided by the following Research Questions (RQs):

1. How would a simulation-based incident reenactment be designed to reconstruct representative aircraft ground-incident scenarios and provide a reproducible evaluation dataset when real incident footage is scarce?
2. Under identical detection conditions, which modern MOT algorithm delivers the most robust and stable tracking of airplane in close-proximity and occlusion-rich apron scenes?
3. To what extent can a vision-based safety framework provide reliable early warnings by combining a reactive proximity criterion with a proactive, trajectory-driven collision-risk prediction mechanism?
4. To what extent is the proposed framework deployable on unmanned platforms, and can its feasibility be examined across simulation-based incident reenactments, real-world apron videos (no-incident control), and laboratory-scale UAV experiments?

1.2. Key Contributions

To bridge the gap between static, passive surveillance and predictive safety support, this study makes the following contributions:

1. **Simulation-Based Benchmark Dataset for Apron Ground Incidents:** We reconstructed the 2018 Asiana–Turkish Airlines wing-to-tail ground incident within a high-fidelity simulation environment and released a labelled 997-frame MOT benchmark to support reproducible evaluation of detection-tracking and warning pipelines under realistic collision geometries that are rarely available as annotated real-world footage.
2. **Fair Benchmarking of MOT Algorithms under a Fixed Detector Backbone:** Four modern trackers (DeepSORT, StrongSORT, ByteTrack, BoT-SORT) were compared under identical detection outputs from an optimised YOLOv8-Seg backbone. Quantitative MOT metrics are reported on the airplane-only MOTChallenge ground truth, identifying DeepSORT as the most robust tracker in occlusion-rich close-proximity scenes; complementary qualitative examples illustrate component-level mask tracking behaviour used by the warning modules.
3. **Dual-Mode Incident-Risk Warning Framework:** We introduce two distinct warning modules that can operate independently: (i) a reactive module that uses segmentation-mask proximity cues in the image plane to trigger immediate alerts, and (ii) a proactive module that extrapolates short-term trajectories over a user-defined prediction horizon and triggers future-risk warnings via IoU-based overlap analysis.
4. **Comprehensive Simulation-to-Reality Validation:** The proposed framework was rigorously validated through a triple-domain strategy: (1) high-fidelity MSFS simulation scenarios, (2) real-world surveillance footage from Hong Kong International Airport to demonstrate generalisation, and (3) laboratory-scaled UAV experiments using a physical drone and diecast aircraft models. Together, these evaluations support the framework's feasibility as a low-cost, deployable perception and warning module for UAV-assisted apron monitoring.

1.3. Structure of the Paper

The remainder of this paper is organised as follows: Section 2 reviews related work in airport ground safety technologies, DL-based object detection, and MOT algorithms. Section 3 details the methodology, encompassing the simulation-based dataset generation, the benchmarking framework, and the mathematical formulation of the dual-layer safety modules. Section 4 presents the experimental results and discussion, covering the quantitative performance of tracking algorithms and qualitative assessment across simulation, real-world footage, and laboratory-scaled UAV tests. Finally, Section 5 summarises the main findings and highlights avenues for future work.

2. Related Work

2.1. Apron Ground Safety and Surveillance Limitations

Conventional airport surface surveillance provides broad situational awareness but remains limited for close-range safety reasoning in congested apron settings. In particular, SMR can continuously map surface traffic; however, it does not directly provide fine-grained object-type classification and may be affected by shadowing as well as resolution and update-rate limitations, which can undermine reliable conflict detection in dense scenes [30,31]. Similarly, ADS-B improves positional awareness for equipped aircraft, but its effectiveness is bounded by operational and security constraints, which restrict its suitability as a standalone safety layer in complex surface environments [6,32,33]. Camera networks are widely deployed and cost-effective, but their safety value is diminished under degraded visibility and, crucially, due to the scalability limits of human monitoring. Without intelligent analytics, CCTV largely remains a recording tool rather than an active safety system [9,34]. Prior studies therefore highlight the need for multi-sensor integration, where cameras become particularly valuable at close range and where CV/DL-based methods can enable automatic detection, classification, and tracking from rich video streams to support earlier identification of hazardous situations [10,35].

2.2. Vision-Based Aircraft Detection and Fine-Grained Component Perception

Remote sensing-based aircraft detection has become a mature substream of vision research, driven by the need to identify aircraft in large-scale scenes where targets are small, densely clustered, and embedded in complex backgrounds [36,37]. Two-stage detectors have been widely adopted to improve small-target sensitivity: for example, Zhu et al. enhanced Faster R-CNN with multilayer feature fusion and a soft-decision Non-Maximum Suppression (NMS) variant, reporting 94.25% accuracy on a Google Earth dataset [38]. More recently, Zeng et al. proposed a top-down strategy that first localises airport areas using multi-source cues and U-shaped network (U-Net) segmentation, then applies a Feature Enhancement Faster R-CNN (FEF-R-CNN) to strengthen weak aircraft features; they reported 97.71% Average Precision (AP) for FEF-R-CNN and 95.63% AP on RSOD, using FROM-GLC10, OSM, Google Images, and RSOD sources [36]. In parallel, one-stage pipelines have been tailored for efficiency, but typically require architectural modifications to handle small objects: Tahir et al. adapted YOLO by upsampling the prediction grid and achieved 90.20% accuracy on a DigitalGlobe satellite imagery dataset [39], while Wang et al. integrated Super-Resolution Generative Adversarial Gan (SRGAN) with YOLOv3 (SR-YOLO) and improved the AP from 92.35% to 96.13% on the UCAS-AOD dataset, additionally reporting 95.12% recall [40]. Despite these advances, the satellite setting predominantly evaluates detection accuracy rather than component-level perception or interaction risk reasoning, limiting its direct transfer to apron incident-prevention objectives that require fine-grained, close-range geometric interpretation.

Beyond satellite imagery, a substantial body of work has focused on aircraft and ground-service detection using CCTV and elevated camera viewpoints, where targets appear small, partially occluded, and embedded in highly cluttered apron scenes. Thai et al. developed a CNN-based airside surveillance framework using calibrated camera footage from Houston and Obihiro airports, enabling aircraft detection, tracking, and push-back monitoring; their system achieved 73.36% AP in Houston and 87.3% AP in Obihiro, highlighting the feasibility of multi-operation monitoring from fixed cameras while remaining sensitive to viewpoint and scene variability [41]. To address the persistent small-object challenge, Zhou et al. proposed ASSD-YOLO, extending YOLOv7 with attention mechanisms, transformer encoding, and dedicated small-target layers. Evaluated on the ASS-Dataset and RSOD, the method achieved 93.5% mean Average Precision (mAP), outperforming the baseline YOLOv7 by up to 20% for aircraft and service vehicles [42]. Focusing on real-time performance, Li et al. introduced the lightweight RPDNet architecture for detecting small aircraft in complex airport videos, reporting a 5.4% mAP improvement over YOLOv3 with 40.5% fewer parameters on a dedicated Zhengzhou Xinzheng Airport dataset [43]. More recently, Zhou et al. presented AD-YOLO, integrating a Swin Transformer and channel-spatial attention into YOLOv7 to balance accuracy and speed, achieving 71.6% mAP at over 100 FPS on their MASD airport dataset [44]. While these studies demonstrate significant progress in CCTV-based airport surveillance, they primarily emphasise detection accuracy and runtime efficiency, with limited consideration of fine-grained aircraft components or downstream interaction and collision-risk reasoning required for proactive apron safety.

Beyond whole-aircraft detection, only a limited number of studies have addressed instance or semantic segmentation of aircraft components in airport environments, aiming to support finer-grained safety analysis. Yilmaz and Karşligil applied Mask R-CNN with a ResNet-101 backbone to apron security camera footage, achieving 100% airplane detection and 98% tail accuracy, but reported substantially lower performance for smaller components such as rear doors (45–76%), highlighting challenges related to occlusion and class imbalance in real scenes [45]. To mitigate data scarcity, Utomo et al. employed Mask R-CNN with systematic image augmentation on the Fine-Grained Visual Classification of Aircraft (FGVC) dataset, detecting wings, engines, fuselage, and tails with a maximum mAP of 90.02%, though validation was limited to controlled image benchmarks rather than operational apron videos [46]. More recently, Thomas et al. proposed a DeepLabV3-based semantic segmentation framework and introduced the AMC-Tr dataset, achieving 84.0% IoU and 91.47% accuracy using transfer learning and a custom Focal Dice Loss; however, the method focuses on static segmentation and does not consider temporal consistency or tracking across frames [47]. Overall, existing component-level studies demonstrate the feasibility of fine-grained aircraft perception but remain constrained by dataset scale, viewpoint diversity, and the limited integration of tracking and risk-aware logic, leaving a clear gap for segmentation-informed tracking and incident-risk warning frameworks in dynamic apron scenarios.

2.3. MOT in Apron Environments

Recent studies have investigated MOT of aircraft and ground vehicles to support airport surface monitoring, primarily by coupling DL-based detectors with online association frameworks. Ahmed et al. enhanced SORT by fusing appearance and motion cues within F-SORT, combined with an improved Fused RetinaNet, achieving 72.75% MOTA and 82.89% IDF1 on aerial airport video sequences while demonstrating improved robustness to small and occluded aircraft [48]. Focusing on optical remote-sensing videos, Su et al. proposed the TDNet pipeline integrating R-FCN detection with MT-KCF tracking, reporting MOTA values above 85% for simultaneous ship and aircraft tracking despite the

challenges posed by low resolution and weather variability [49]. In apron surveillance scenarios, Xu et al. employed an improved YOLOv5 detector with a SORT-like association scheme to track aircraft and ground service equipment during turnaround operations, achieving an average MOTA of 95.09% on custom apron video datasets while targeting process monitoring rather than safety-critical interactions [50]. More recently, Mazzeo et al. combined YOLOv8 with StrongSORT for UAV-based MOT on the VisDrone2019 dataset, obtaining 42.03% MOTA under challenging conditions involving scale variation and appearance changes [51]. While these studies show steady progress in tracking accuracy, they largely focus on platform-level targets and general MOT metrics and often provide limited treatment of dense apron occlusions and rarely explore how tracking outputs can be exploited for early incident-risk warning using fine-grained proximity cues.

2.4. Collision Risk Estimation and Early Warning

Recent studies using CV and DL have begun to address collision risk on aprons by linking perception with basic predictive models. For example, Zhu et al. combined YOLOv7 detection with a Long Short-Term Memory (LSTM) model to estimate aircraft trajectories during towing, providing real-time wingtip warnings on a custom video dataset [52]. While effective, this approach is tailored to a specific towing scenario and focuses on short-term interactions. Sun et al. took a broader, system-level view [53]. They used Petri nets, Brownian motion, and XGBoost to classify apron risk and re-plan paths based on operational data, achieving over 95% accuracy in risk classification. However, Sun et al.'s framework was validated purely through simulation, without using real-time visual data for close-range perception. In contrast, Gaikwad et al. developed a vision-based system for autonomous taxiing, using segmentation for lane-keeping and a Linear Quadratic Regulator (LQR) controller for obstacle avoidance on custom airport videos [54]. While it enhances navigation safety, the method does not explicitly quantify collision risk between specific aircraft structures (e.g., wing–tail clearances). Looking at another application, Ywet et al. proposed R-YOLO, which merges detection with LSTM prediction for Urban Air Mobility safety at vertiports, showing accurate short-term forecasts on simulation datasets [55]. Yet their framework is not designed for the dense, close-proximity risks characteristic of traditional airport aprons. Overall, prior work highlights the promise of coupling perception with prediction but often lacks segmentation-informed proximity reasoning and a dual-mode warning logic that is evaluated under realistic apron interaction scenarios.

In summary, as reflected in Table 2, significant progress has been made in CV systems for airport surface operations, spanning aircraft detection from remote sensing and CCTV viewpoints to MOT of aircraft and service vehicles, as well as early forms of collision-risk estimation and warning. However, these approaches are often investigated in isolation. Detection-focused studies typically report frame-level performance and runtime efficiency without enforcing temporal continuity, whereas tracking-oriented studies commonly benchmark MOT metrics without integrating explicit safety or incident-risk warning logic. Likewise, risk-estimation frameworks are frequently restricted to short-horizon or scenario-specific predictions and may not exploit fine-grained image-plane proximity cues relevant to close-range aircraft interactions (e.g., wing–tail clearances). Evaluations are also often limited to short clips or simulation-only settings, with insufficient coverage of dense apron dynamics in continuous videos. Consequently, an integrated approach that combines validated fine-grained perception, continuous tracking, and explicit incident-risk warning logic remains underexplored in the literature. Such a framework is evaluated across a simulation-based incident reenactment, real-world apron videos, and UAV-relevant laboratory-scale imagery.

Table 2. Representative studies on aircraft detection, tracking, and collision-risk warning relevant to apron environments.

Dataset	Core Method	Primary Purpose	Key Achievement	Main Limitation	Ref
Satellite (FROM-GLC10, RSOD)	FEF-R-CNN	Aircraft Detection	97.71% AP allowing airport-level localisation	No component-level or close-range reasoning	[36]
CCTV (ASS-Dataset)	YOLOv7 + Attention	Small-Object Detection	93.5% mAP for aircraft and vehicles	Detection-only; no temporal reasoning	[42]
AMC-Tr Dataset	DeepLabV3	Segmentation	84.0% IoU for aircraft parts	Static images; no tracking	[47]
Aerial Airport Videos	F-SORT + RetinaNet	MOT of Aircraft	72.75% MOTA, 82.89% IDF1	No safety or risk logic	[48]
Apron CCTV Videos	YOLOv5 + SORT	Turnaround Tracking	95.09% MOTA	Process monitoring only	[50]
VisDrone2019 (UAV)	YOLOv8 + StrongSORT	UAV-based MOT	41.03% MOTA, Robust under scale variation	No collision reasoning	[51]
Towing Videos	YOLOv7 + LSTM	Wingtip Collision Warning	Real-time short-horizon alerts	Scenario-specific	[52]
Operational Data (Sim)	Petri Net + XGBoost	Risk Classification	>95% accuracy	No visual perception	[53]
AirSim/MOT	R-YOLO + LSTM	UAM Safety	Accurate trajectory prediction	Not apron or component-focused	[55]

3. Methodology

3.1. Experimental Domains and Data Sources

To assess the proposed vision-based pipeline for aircraft tracking and collision-risk warning in apron surveillance and UAV-assisted monitoring, we adopted a multi-domain evaluation strategy using three complementary data sources: (i) a controlled simulation-based incident reenactment video created in Microsoft Flight Simulator (MSFS; Microsoft Corporation, Redmond, WA, USA; released in 2020) for repeatable high-risk scenarios; (ii) real-world apron videos recorded from a high-elevation fixed viewpoint to examine performance under operational visual conditions; and (iii) a laboratory-scale experiment in which diecast aircraft models were filmed by a UAV-mounted camera to validate the same detection, tracking, and risk-reasoning pipeline on unmanned-platform imagery. All reported experiments focus on nominal-visibility (clear/daylight) operating conditions, which are the conditions covered by the datasets/scenarios used in this study; extending the analysis to adverse weather and low-light degradations (e.g., rain, fog, glare, and night-time illumination) is discussed as part of the deployment-oriented considerations in Section 4.9.1.

3.1.1. Simulation-Based Incident Reenactment Video (MSFS)

The primary experimental source is a controlled incident reenactment sequence created in Microsoft Flight Simulator (MSFS), depicting a close-proximity apron interaction in which the wing of a moving aircraft approaches and contacts the tail region of a parked aircraft. The sequence was inspired by the 2018 Istanbul Ataturk Airport ground incident involving an Asiana Airlines aircraft and a parked Turkish Airlines aircraft and was reconstructed to represent the key spatial characteristics of a wing–tail contact scenario. Because the original incident’s camera/sensor metadata (e.g., intrinsics, extrinsics, or time-synchronised kinematics) was not available to us, the reenactment is not claimed to be a calibrated replica of the real recording, but rather a controlled, incident-inspired benchmark for reproducible evaluation. MSFS was used to develop and stress-test the tracking and warning logic under safe, repeatable conditions, and no detector or tracker weights were trained on this sequence. The scene was recorded from a fixed, elevated viewpoint representing an apron surveillance camera position near a taxiway–apron junction. The original video was captured at 1920×1080 resolution and 60 fps for approximately 35 s, then uniformly sampled to 997 frames for dataset construction and evaluation, yielding an effective frame rate of $f_{\text{eff}} \approx 28.49$ fps. We used the dataset to benchmark four

tracking-by-detection multi-object trackers (ByteTrack, DeepSORT, StrongSORT, and BoT-SORT) under a fixed detector backbone and identical pre- and post-processing conditions.

3.1.2. Real-World Apron Footage (Hong Kong; No-Incident Control)

To complement the controlled simulation evidence, a real-world apron video recorded from a high-elevation fixed camera viewpoint was included (publicly available footage; no ground-truth annotations available). The sequence contains multiple aircraft movements but no incident, and was therefore used as a no-incident control to qualitatively assess whether the pipeline can (i) detect aircraft, (ii) maintain stable identities in a visually complex apron scene, and (iii) avoid false risk alarms in the absence of hazardous interactions. This evaluation is used as an operational robustness check rather than a controlled benchmark.

3.1.3. Laboratory-Scale UAV Experiment (Diecast Aircraft + Drone Video)

Finally, to align the study with unmanned-platform validation expectations, we conducted a laboratory-scale UAV experiment in which a UAV-mounted camera recorded two diecast aircraft models undergoing controlled close-proximity motion. This laboratory trial is included as a feasibility and integration check to verify that the end-to-end detection–tracking–warning pipeline operates on UAV-captured imagery with practical aerial-video artefacts (e.g., viewpoint changes, camera motion/vibration, and compression). Unlike the MSFS reenactment (controlled incident geometry for repeatable benchmarking) and the real apron video (no-incident control under real background complexity), the laboratory trial specifically supports the platform claim that the proposed logic can run on aerial video acquired from an elevated sensor; accordingly, it prioritises controllability and safety over full operational realism. The term ‘laboratory-scale’ refers to a controlled indoor feasibility setup rather than a calibrated geometric scale-model measurement configuration, and risk reasoning is performed in the image plane (pixel space) using the same pipeline. While the laboratory scene is intentionally simplified (uniform/low-clutter background) to provide a safe and repeatable feasibility check of the UAV-capture-to-warning pipeline, environmental realism is primarily addressed through the MSFS simulation-based incident reconstruction and the real-world apron footage described in Sections 3.1.1 and 3.1.2. The laboratory protocol is detailed in Section 3.7, and the corresponding outputs are reported in the Results/Discussion section.

3.2. Ground Truth Construction and Dataset Specification

The quantitative MOT evaluation is based on a custom annotated dataset derived from the MSFS sequence. The dataset comprises 997 frames at 1920×1080 resolution and includes a single class (airplane) for unambiguous identity tracking and metric computation. Two aircraft were assigned persistent identities across the sequence, resulting in 1991 airplane annotations. Dataset statistics are summarised in Table 3.

Table 3. Dataset overview for MSFS incident reenactment-based aircraft MOT benchmarking.

Category	Metric	Value
General Dataset Properties	Total Number of Images (Frames)	997
	Image Resolution	1920×1080
	Total Number of Annotations	1991
	Average Annotations per Image	≈ 2
	Total Number of Classes	1
Class-Level Annotation Distribution	Airplane	1991
Annotation Density per Image	Average Image Size	2.07 MP (megapixel)

For quantitative MOT benchmarking, only the airplane class was annotated in MOTChallenge format. In contrast, the incident-warning experiments use part-level YOLOv8-Seg outputs (e.g., airplane, wing, tail, nose, fuselage) for fine-grained risk reasoning in the reactive and proactive modules. A representative frame from the MSFS simulation-based reenactment and a real-world reference image from the 2018 Asiana Airlines–Turkish Airlines ground incident are shown in Figure 1 to illustrate the visual similarity of the scene geometry and aircraft interaction that motivated the MOT benchmarking setup.



Figure 1. Visual comparison between the MSFS simulation-based incident reenactment and the real incident scene that motivated the benchmark construction. (a) Representative frame from the MSFS reenactment sequence, captured from a fixed elevated viewpoint to emulate UAV-relevant monitoring geometry. (b) Real-world reference image from the 2018 Asiana Airlines–Turkish Airlines wing-to-tail ground incident at Istanbul Ataturk Airport, shown for qualitative comparison of scene geometry and aircraft interaction [20].

To support reliable identity-based evaluation, frames were manually annotated using instance segmentation masks with additional zoom-based inspection in close-proximity and overlap regions. Persistent track identities were manually verified across the full sequence to enable identity-sensitive MOT metrics (e.g., IDF1 and identity switches).

For benchmarking, annotations were exported in COCO format and converted to MOTChallenge ground-truth format, where each object is represented as: (frame id, track id, x, y, width, height, confidence, class id, visibility).

Because the original labels were mask-based, bounding rectangles were derived from the segmentation masks during conversion. The track-id field was manually created and propagated across frames, since persistent identities are not provided by default in the export source. This produced an unambiguous ground-truth file for fair tracker comparison. The resulting ground truth will be made publicly available upon publication to support future apron-level MOT benchmarking research.

3.3. Selection of MOT Algorithms and Fair Benchmark Protocol

Four tracking-by-detection algorithms were selected to represent complementary data-association paradigms for apron monitoring. DeepSORT was included as a widely used baseline that combines Kalman-filter motion prediction with appearance-based association via deep embeddings, which is relevant when multiple visually similar aircraft coexist in the same scene [56]. StrongSORT was selected as an enhanced DeepSORT-

family variant with improved embedding handling and association refinements for greater robustness under viewpoint and illumination changes [57]. ByteTrack was included as a contrasting paradigm that preserves association continuity by using both high- and low-confidence detections, which can be advantageous under partial occlusion and detector-score fluctuations [58]. Finally, BoT-SORT was selected as a modern SORT-family tracker that integrates motion and appearance cues with additional robustness components, including camera-motion compensation and an improved Kalman-state design, providing a strong contemporary reference [59].

To ensure a fair comparison, all trackers were evaluated under an identical perception and evaluation pipeline. The same detector backbone, the optimised YOLOv8-Seg model from our prior work [17], was used to produce detections for every tracker, and the same input video stream and frame sequence were applied throughout. For quantitative MOT benchmarking, airplane-class detections (bounding boxes and confidence scores) were used as tracker inputs. The detection configuration (including confidence thresholding and NMS) was fixed across experiments, and each tracker received the same detection output structure. No tracker-specific parameter tuning was performed; default hyperparameters from the respective official implementations were used to avoid scenario-specific overfitting and preserve reproducibility. Under these conditions, performance differences are attributable primarily to tracker association behaviour rather than detector or preprocessing variations.

3.4. Experimental Configuration for MOT Comparison

All experiments were executed in Google Colab Pro using an NVIDIA A100 GPU with 40 GB VRAM, under a fixed software stack (Python 3.11.12, CUDA 12.4, PyTorch 2.6.0+cu124). The framework is platform-agnostic in the sense that the detection–tracking–warning pipeline is not tied to a specific sensor or airframe; however, the results reported in this paper are obtained via offline inference on this GPU environment, and onboard deployment introduces latency and power constraints (discussed in Section 4.9.1). The MOT comparison used a tracking-by-detection pipeline in which the optimised YOLOv8-Seg detector (best.pt) generated per-frame detections, which were filtered to the airplane class before tracker association.

For tracking, detector outputs were fed to each tracker to associate detections and maintain identities across frames. All trackers used the same video, frame sequence, and detector weights, with default configurations (no tracker-specific tuning). The controlled benchmarking settings are summarised in Table 4.

Table 4. Experimental settings for simulation-based MOT benchmarking under a fixed detector backbone and identical inputs across trackers (UAV-relevant elevated viewpoint, consistent hardware/software stack, and shared detection configuration).

Setting	Value
Hardware:	NVIDIA A100 GPU (40 GB VRAM); 2–4 vCPUs; 25 GB RAM
Execution environment:	Google Colab Pro
Software stack:	Python 3.11.12; CUDA 12.4; PyTorch 2.6.0+cu124
Input stream:	Simulation-based reenactment incident video sequence
Video properties:	1920 × 1080; 60 FPS (recorded); 997 uniformly sampled frames used; $f_{\text{eff}} \approx 28.49$ (used); ~35 s duration
Target class:	airplane
Detector:	Optimised YOLOv8-Seg model (best.pt)
Confidence threshold (conf):	0.25
NMS IoU threshold:	0.45
History length and Velocity window	$N = 30$ frames, $M = 10$ most recent steps.

Prediction horizon:	$T_{\text{pred}} = 5$ s (user-defined; 5 s used in all reported proactive experiments)
IoU warning threshold:	$\tau_{\text{IoU}} = 0.10$ (user-defined; conservative early-warning setting)
Reactive distance thresholds:	$\tau_{\text{collision}} = 40$ px, $\tau_{\text{warning}} = 80$ px (user-defined image-plane, scene-dependent distance threshold)
Trackers compared:	ByteTrack; DeepSORT; StrongSORT; BoT-SORT
Tracker configuration:	Default settings; no tracker-specific parameter tuning
Outputs:	Overlay MP4 export; per-frame tracking logs for metric computation

For qualitative inspection, tracked outputs were rendered as MP4 overlay videos with identity labels and trajectories.

3.5. Performance Evaluation Metrics

Tracker performance was evaluated using standard MOT metrics covering tracking accuracy, localisation precision, and identity consistency. We report the CLEAR MOT metrics (MOTA, MOTP), identity metrics (IDF1, IDSW), and TP/FP/FN counts, from which precision and recall are computed.

3.5.1. Multi-Object Tracking Accuracy (MOTA)

MOTA is used as the primary overall MOT indicator [60], as it jointly penalises false negatives, false positives, and identity switches across the sequence. It is defined as:

$$\text{MOTA} = 1 - \frac{\sum_t (FN_t + FP_t + IDSW_t)}{\sum_t GT_t} \quad (1)$$

where t denotes the frame index, and FN_t , FP_t , and $IDSW_t$ represent the numbers of false negatives, false positives, and identity switches at frame t , respectively. GT_t denotes the number of ground-truth objects present at frame t . The denominator $\sum_t GT_t$ therefore represents the total number of ground-truth object instances across the evaluated sequence. Higher MOTA indicates better tracking performance (maximum = 1, or 100%).

3.5.2. Multi-Object Tracking Precision (MOTP)

MOTP measures localisation precision for correctly matched targets [60]. In the CLEAR MOT formulation used here, it is computed as the average localisation error over all matches:

$$\text{MOTP} = \frac{\sum_{i,t} d_{i,t}}{\sum_t c_t} \quad (2)$$

where c_t is the number of successful matches at frame t , and $d_{i,t}$ is the localisation error for the i -th match (here taken as $1 - \text{IoU}$). Under this distance-based definition, lower MOTP indicates tighter alignment between predicted and ground-truth boxes.

3.5.3. Identity Switches (IDSW) and Identity F1 Score (IDF1)

IDSW counts the number of incorrect identity changes along tracked trajectories; lower values indicate better identity consistency during interactions or partial occlusions [60]. IDF1 measures identity preservation quality over time and is defined as the harmonic mean of identification precision and identification recall [61]:

$$\text{IDF1} = \frac{2 \cdot \text{IDTP}}{2 \cdot \text{IDTP} + \text{IDFP} + \text{IDFN}} \quad (3)$$

where IDTP , IDFP , and IDFN denote true positive identity matches, false positive identity assignments, and false negative identity matches, respectively. Higher IDF1 indicates more stable identity preservation.

3.5.4. Precision and Recall

Precision and recall summarise the balance between correct detections, false alarms, and missed detections in the evaluated MOT pipeline:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (4)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (5)$$

Here, TP , FP , and FN are obtained from MOT evaluation matches against the airplane-only ground truth. These metrics summarise the false-positive/false-negative trade-off under the same evaluation setting.

3.6. Dual-Mode Collision Risk Assessment Modules

Two complementary incident-risk modules are used in this study: (i) a reactive module, which evaluates instantaneous proximity from segmentation-mask geometry in the current frame, and (ii) a proactive module, which predicts short-horizon future occupancy and evaluates overlap risk using a future-IoU proxy. The modules are methodologically distinct and can be used independently.

3.6.1. Reactive Module: Mask-Based Proximity Analysis as a Pixel Space Risk Proxy

The reactive module produces a frame-wise risk state (Safe, Warning, Collision) by measuring the minimum separation between aircraft segmentation mask boundaries in pixel space, rather than using bounding-box proximity alone. This provides a tighter geometric proxy for close interactions involving aircraft extremities (e.g., wingtip/tail).

(A) The role of the process flow and the diagram.

Figure 2 summarises the per-frame workflow. YOLOv8-Seg provides detections and masks; DeepSORT assigns persistent IDs from bounding boxes; track IDs are then associated with the corresponding masks; the minimum mask-contour distance is computed for each aircraft pair; and a threshold rule maps the result to a three-level warning state.

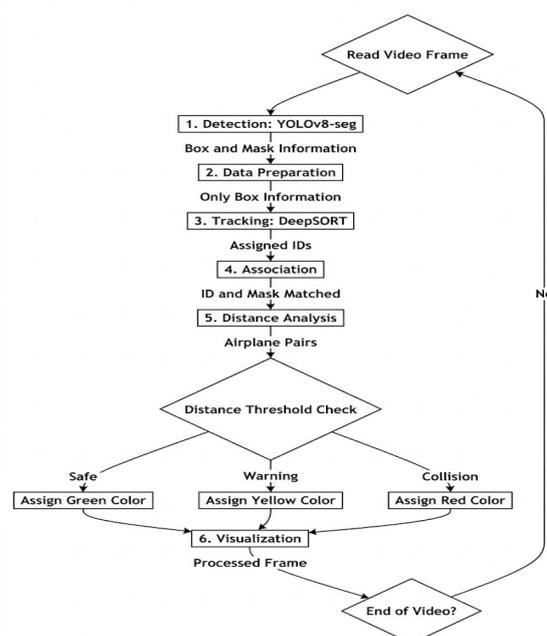


Figure 2. Frame-wise processing loop of the reactive module. YOLOv8-Seg performs aircraft detection and instance segmentation, DeepSORT assigns persistent IDs, and the minimum mask-contour

distance is thresholded to output Safe/Warning/Collision alerts. Arrows indicate the processing flow; decision branches correspond to the threshold outcomes (safe/warning/collision) and the loop continues until the end of the video.

(B) Matching Track ID with Mask

YOLOv8-Seg returns paired detections (bounding box + instance mask) [62], while DeepSORT tracks bounding boxes and maintains persistent identities across frames [56]. Therefore, each DeepSORT track is linked to its corresponding YOLOv8-Seg mask in the same frame via nearest-neighbour association in the bounding-box centre space, as implemented in this study.

Let the centre of the bounding box of the i -th track at time t be $c_i^{trk}(t) = (x_i^{trk}(t), y_i^{trk}(t))$, and let the centre of the j -th detection bounding box at time t be $c_j^{det}(t) = (x_j^{det}(t), y_j^{det}(t))$. The detection index assigned to track i is obtained by:

$$j^*(i, t) = \arg \min_j \|c_{i,t}^{trk} - c_{j,t}^{det}\|_2 \quad (6)$$

Equation (6) defines the nearest-centre track-to-mask association rule used in the proposed implementation.

The association distance in Equation (7) is the standard Euclidean (ℓ_2) distance [63] in the bounding-box centre space:

$$\|c_{i,t}^{trk} - c_{j,t}^{det}\|_2 = \sqrt{(x_{i,t}^{trk} - x_{j,t}^{det})^2 + (y_{i,t}^{trk} - y_{j,t}^{det})^2} \quad (7)$$

This step ensures that each persistent track identity is paired with the correct instance mask for subsequent proximity computation and warning logic. Each YOLOv8-Seg detection provides a paired bounding box and mask [62]; therefore, the association is performed at the detection-box level to link each DeepSORT track to its corresponding segmentation instance before contour-based proximity is computed. In dense part-level scenes (e.g., small tail/wing components under close proximity), this step can be made more robust by replacing centre proximity with an IoU-gated one-to-one assignment (e.g., maximising box IoU between tracks and detections) [56], which we treat as a deployment-oriented refinement. Non-rectangular mask geometry primarily affects the subsequent contour-distance proxy, whereas the ID-mask pairing follows the per-instance box-mask coupling returned by the detector.

(C) Minimum Mask Distance for Pixel Space Risk Estimation

After identity-to-mask pairing, collision risk is quantified by the minimum separation between the boundaries of two aircraft masks in pixel space. For a given frame t , let the instance masks of aircraft i and k be denoted by $M_{i,t}$ and $M_{k,t}$, and let their corresponding boundary point sets (contours) be

$$C_{i,t} = \{p_1, p_2, \dots, p_n\}, \quad C_{k,t} = \{q_1, q_2, \dots, q_m\}, \quad (8)$$

where p and q denote contour points sampled from the two masks. If the two masks overlap (i.e., they have a non-zero intersection), the separation is defined as zero, corresponding to contact-level interaction:

$$\text{if } |M_{i,t} \cap M_{k,t}| > 0, \quad d_{\min}(t) = 0. \quad (9)$$

Otherwise, the minimum contour-to-contour distance is computed as

$$d_{\min}(C_{i,t}, C_{k,t}) = \min \left(\min_{p \in C_{i,t}} (p, C_{k,t}), \min_{q \in C_{k,t}} (q, C_{i,t}) \right) \quad (10)$$

where the distance from a point p to a contour C is defined as the minimum Euclidean distance [63] to any point on that contour:

$$\text{dist}(p, C) = \min_{q \in C} \|p - q\|_2 = \sqrt{(p_x - q_x)^2 + (p_y - q_y)^2} \quad (11)$$

In implementation, $\text{dist}(p, C)$ is obtained using OpenCV `cv2.pointPolygonTest` [64], which returns the shortest distance from a point to a polygonal contour. The resulting d_{\min} provides an interpretable pixel-space proximity proxy that reflects the closest interaction regions between aircraft masks (e.g., nose-to-nose or wingtip-to-tail proximity).

(D) Risk Thresholding and Output Semantics

The minimum mask distance $d_{\min}(t)$ is mapped to a discrete risk state using pixel-space thresholds. Let τ_c denote the collision threshold and τ_w denote the warning threshold, with $\tau_c < \tau_w$. Thresholds are user-defined parameters and were kept fixed across all reported reactive experiments. The reactive decision rule is defined as:

$$\text{state}(t) = \begin{cases} \text{Collision,} & d_{\min}(t) \leq \tau_c, \\ \text{Warning,} & \tau_c < d_{\min}(t) \leq \tau_w, \\ \text{Safe,} & d_{\min}(t) > \tau_w. \end{cases} \quad (12)$$

In this study, the thresholds were set to $\tau_c = 40$ pixels and $\tau_w = 80$ pixels, yielding a three-level warning logic that captures clear separation, hazardous proximity, and contact-level interaction in an interpretable pixel-space form. The output is visualised by rendering the tracked aircraft masks with colour-coded semantics: green indicates safe separation, yellow indicates warning-level proximity, and red indicates a collision. For traceability, each aircraft is additionally annotated with its track identity and a bounding box, and the corresponding warning message is overlaid on the processed video frames. Note that these pixel thresholds are image-plane quantities and can be calibrated to physical separation distances when camera geometry and scale are available. Because the proxy d_{\min} is measured in pixels, the numeric values of τ_c and τ_w are tied to the imaging geometry (e.g., camera intrinsics/field of view and camera–scene distance/altitude) and are therefore not inherently viewpoint-invariant. Accordingly, the specific values reported here ($\tau_c = 40$ px, $\tau_w = 80$ px) should be interpreted as scene-specific operating parameters under the fixed viewpoints used in our evaluation, rather than universal physical clearances. In deployment, transfer across different UAV altitudes or camera intrinsics can be supported either by mapping pixel distances to metric thresholds via camera calibration (intrinsics/extrinsics or an estimated ground-plane homography, optionally aided by UAV telemetry such as altitude and gimbal pose), or by using scale-normalised thresholds expressed relative to the observed target scale (e.g., normalising by the apparent aircraft size in the image), which reduces sensitivity to altitude/focal-length changes. We treat these geometry-aware mappings as deployment-oriented extensions and discuss them in Section 4.9.

(E) Algorithmic Summary and Reproducible Implementation

The complete reactive per-frame pipeline is summarised in Algorithm A1 (see Appendix A), covering YOLOv8-Seg inference, DeepSORT update, ID–mask association, contour-distance computation, threshold-based state assignment, and annotated-frame exporting.

3.6.2. Proactive Module: Trajectory Prediction and Future IoU Proxy

This proactive module provides an early-warning signal by forecasting short-horizon future occupancies of tracked aircraft (the airplane class in most experiments and airplane-part classes in a single experiment) and evaluating their predicted spatial overlap. In contrast to the reactive module, which operates on instantaneous proximity, this module uses a user-defined prediction horizon T_{pred} (s) and triggers a warning when the IoU between predicted future bounding boxes exceeds a user-defined threshold τ_{IoU} .

In all reported proactive experiments, the IoU threshold was set to $\tau_{IoU} = 0.10$ as a conservative early-warning setting. Because the forecasted boxes are approximate (image-plane, first-order extrapolation) and can exhibit only small overlaps before a physical interaction becomes imminent, a lower τ_{IoU} prioritises sensitivity (reducing missed warnings) at the expense of a higher false-alarm tendency. This choice is aligned with safety-critical monitoring, where earlier caution can be preferable to late detection. Importantly, τ_{IoU} is user-configurable in the range $[0,1]$ and can be tuned to site policy, viewpoint, and the acceptable false-alarm rate.

The trajectory extrapolation in this module adopts a constant-velocity assumption estimated from recent motion history. This choice is intentional: our aim is a lightweight, transparent early-warning mechanism that does not require additional training data or a platform-specific dynamics model, and whose behaviour can be directly controlled via T_{pred} and τ_{IoU} . In apron operations, aircraft taxi/pushback motion is typically slow and locally smooth over short time windows; therefore, for short horizons (e.g., a few seconds), linear extrapolation provides a practical risk proxy for early warning rather than a high-fidelity trajectory model. We explicitly note that the assumption can degrade under non-linear manoeuvres such as pushback turns, braking/acceleration, steering changes, stop-go motion, or abrupt avoidance actions, which may increase false alarms or missed warnings. We did not adopt learning-based predictors (e.g., LSTM/GRU) because they require representative trajectory training data and introduce additional model complexity and hyperparameter sensitivity; similarly, Kalman-filter variants were not the focus here because they require a specific process/measurement model and careful tuning for each operational setting. These limitations are acknowledged in Section 4.9 Limitations and Future Work.

In implementation, the velocity vector is re-estimated at every frame using only the most recent M centre displacements (here $M = 10$) rather than the full history buffer, which helps the forecast react to gradual speed changes. Nevertheless, the predictor remains a first order (constant-velocity) image-plane model and is not intended to capture abrupt non-linear manoeuvres (e.g., emergency braking or sharp turning). For such regimes, the system can be configured with a shorter user-defined horizon T_{pred} to preserve the local-linearity assumption; in this study we report results with $T_{pred} = 5$ s as a representative short-horizon setting (see Table 4).

Figure 3 summarises the end-to-end proactive workflow: ID-consistent tracking, motion-history buffering, horizon-based trajectory extrapolation, future-IoU evaluation, and in-frame warning visualisation.

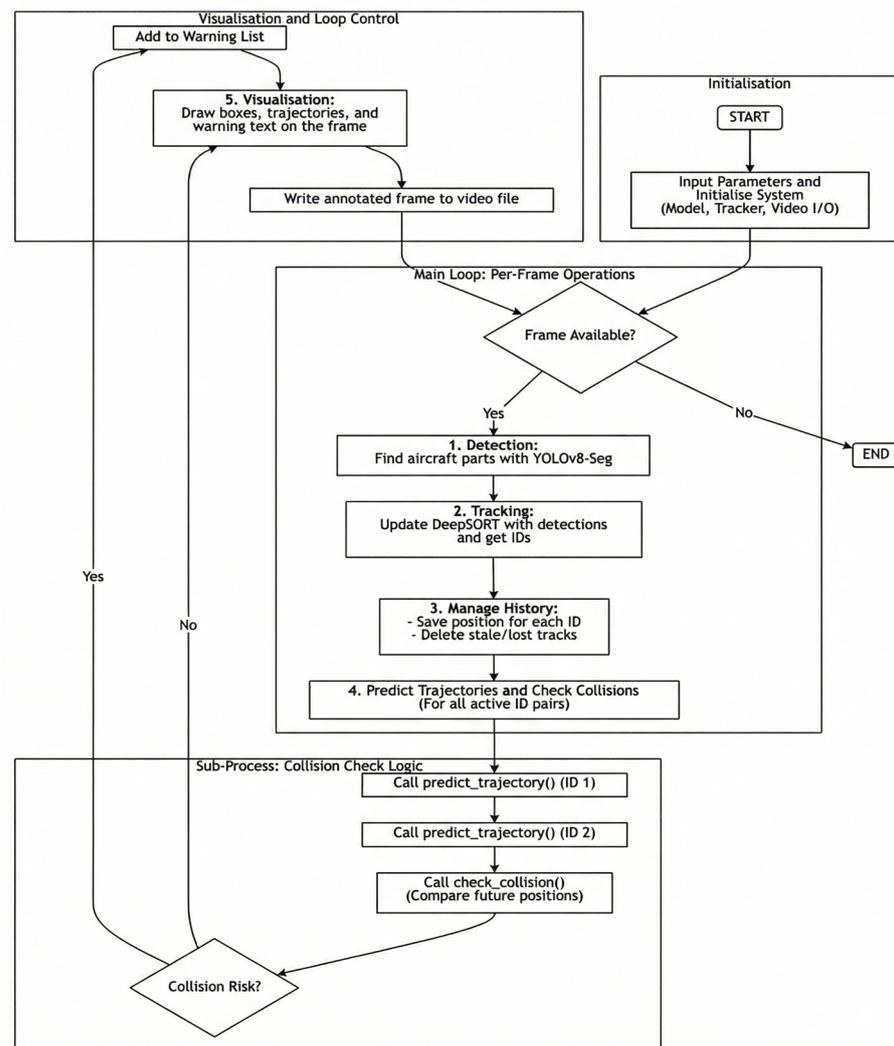


Figure 3. End-to-end per-frame workflow of the proactive incident-risk module, showing detection and ID maintenance, motion-history buffering, forward-projected bounding boxes, and IoU-based risk triggering with on-frame warning visualisation. Arrows indicate the direction of the processing flow, and decision nodes branch according to the corresponding condition outcomes.

(A) Track history buffer and state maintenance

For each confirmed DeepSORT track i , we maintain a fixed-length history buffer that stores the most recent bounding boxes, denoted $b_i(t) = [x_1, y_1, x_2, y_2]$ at frame t . The buffer is implemented as a deque with maximum length $N = 30$ frames. In addition, each track stores the last frame index at which it was observed. Tracks that remain unobserved beyond the tracker's age limit (max_age) [56] are removed to prevent stale identities from contributing to future-risk checks.

(B) Velocity estimation from recent motion

Given the history buffer $H_i(t)$, the module estimates image-plane motion using bounding-box centres. For a bounding box $b_i(t) = [x_1(t), y_1(t), x_2(t), y_2(t)]$, the centre is:

$$C_i(t) = \left(\frac{x_1(t) + x_2(t)}{2}, \frac{y_1(t) + y_2(t)}{2} \right) \quad (13)$$

To reduce sensitivity to older dynamics, velocity is estimated from only the most recent $M = \min(10, |H_i(t)|)$ boxes. Using per-frame centre differences $\Delta c_i(\cdot)$, we define the recent-window mean velocity estimator used in this study as:

$$\bar{v}_i(t) = \frac{1}{M-1} \sum_{k=1}^{M-1} (c_i(t-k+1) - c_i(t-k)) \quad (14)$$

expressed in pixels/frame. This yields a short-horizon constant-velocity approximation suitable for near-term forecasting in the image plane [56].

(C) Forward projected and future occupancy box construction

Let f denote the video frame rate used by the processing pipeline and let the user-defined prediction horizon be T_{pred} (s). The horizon in frames is $K = \lceil T_{\text{pred}} f \rceil$. (If uniform frame sampling is applied, f corresponds to the effective frame rate f_{eff} .) The future centre is extrapolated as:

$$\hat{c}_i(t+K) = c_i(t) + K \cdot \bar{v}_i(t) \quad (15)$$

Equation (15) defines the short-horizon constant-velocity centre extrapolation used in the proposed proactive module, consistent with common motion-model assumptions used in tracking-based forecasting [56].

The future bounding box $\hat{b}_i(t+K)$ is then obtained by translating the most recent box to the predicted centre while keeping its width and height fixed to the most recent observation. The resulting future box serves as an occupancy proxy indicating where the tracked object is expected to be at the prediction horizon.

(D) Future IoU proxy and collision-warning rule

For each pair of active tracks (i, j) , the module computes the *IoU* between their predicted future boxes:

$$IoU(\hat{b}_i, \hat{b}_j) = \frac{|\hat{b}_i \cap \hat{b}_j|}{|\hat{b}_i| + |\hat{b}_j| - |\hat{b}_i \cap \hat{b}_j| + \varepsilon} \quad (16)$$

Equation (16) uses the standard *IoU* overlap measure for bounding boxes [65]. A collision risk is flagged when:

$$IoU(\hat{b}_i(t+K), \hat{b}_j(t+K)) > \tau_{IoU} \quad (17)$$

where τ_{IoU} is a user-defined sensitivity parameter and ε is a small constant to avoid division by zero. When the condition holds, the pair (i, j) is added to a warning set W_t . In the visual output, the system overlays a warning message indicating a predicted overlap risk approximately T_{pred} seconds (i.e., K frames) ahead, together with the involved track IDs (and, when applicable, the corresponding part labels).

(E) Visualisation and computational characteristics

For each active track, the module draws: (i) the current bounding box, (ii) the predicted future box, and (iii) a line segment connecting the current and future centres to illustrate the estimated motion direction. If $W_t \neq \emptyset$, a 'COLLISION' warning banner (i.e., predicted risk under the future-IoU proxy) and pairwise warnings are rendered on the frame. The risk check is performed pairwise over all active tracks, resulting in $O(n_t^2)$ IoU evaluations per frame for n_t active identities; in practice, the implementation first computes all future boxes once per frame and then evaluates IoU over all unique unordered ID pairs. The complete per-frame proactive decision logic, including history buffering, horizon-based forward projection, and future-IoU thresholding for warning generation, is summarised in Algorithm A2 (see Appendix B).

3.7. Scaled UAV/Laboratory Validation Protocol

A laboratory-scale UAV experiment was conducted to validate the proposed detection-tracking-warning pipeline on real UAV-acquired video under controlled and repeatable conditions, complementing the main simulation-based evaluation.

3.7.1. Experimental Setup and Data Acquisition

A consumer UAV with an RGB camera captured indoor video at 1920×1080 resolution and 60 FPS from a fixed altitude of 1.2 m using an elevated, slightly oblique (near-nadir) viewpoint. Figure 4 illustrates the laboratory-scale acquisition assets, including the UAV platform and the two diecast aircraft models used to generate controlled interaction sequences.

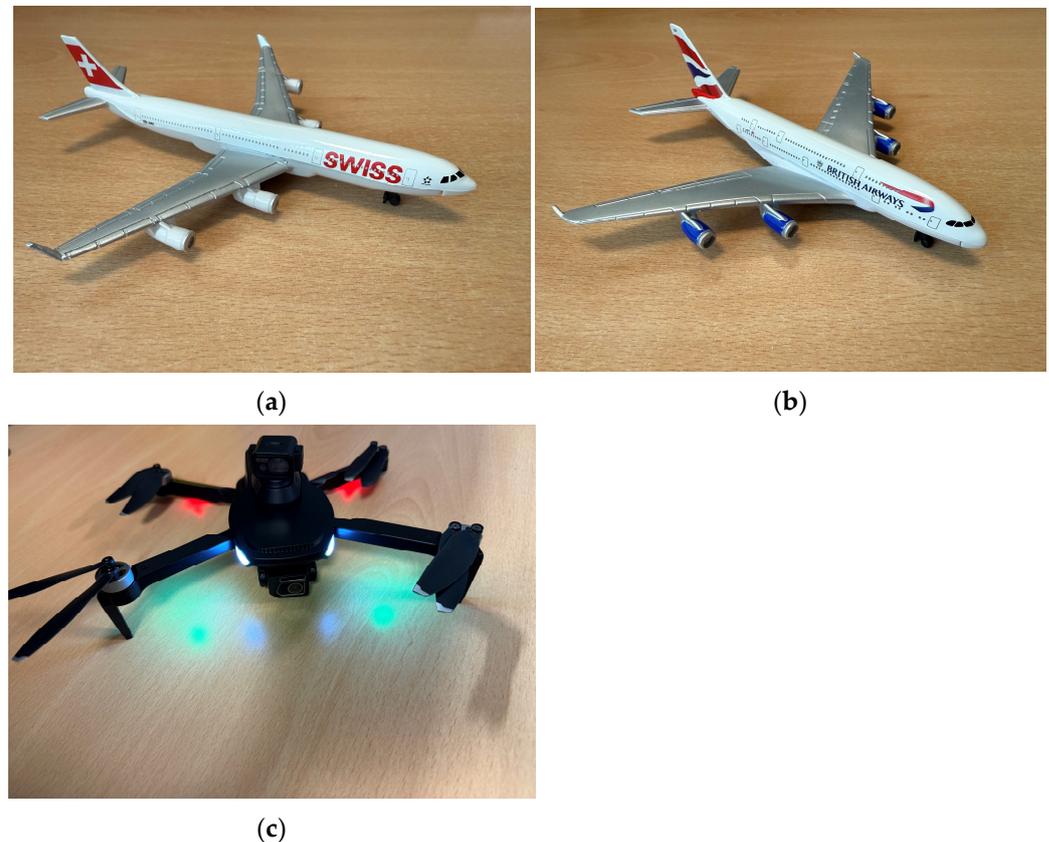


Figure 4. Laboratory-scale UAV validation assets used for indoor data acquisition: (a) diecast aircraft model (Swiss livery), (b) diecast aircraft model (British Airways livery), and (c) consumer quadrotor UAV used to acquire elevated UAV footage with a slightly oblique viewpoint.

The models (Swiss livery: approx. length 14.6 cm, wingspan 12.7 cm; British Airways livery: approx. length 15.2 cm, wingspan 16.5 cm) were placed on a planar surface under controlled illumination with a uniform background. To produce repeatable interactions, one model remained stationary while the other was translated along a predefined path using a thin string, yielding (i) Scenario A: no-incident close pass (no contact) and (ii) Scenario B: controlled contact (wing–tail contact).

3.7.2. Processing Pipeline and Reported Outputs

The UAV videos were processed using the same detector, tracker, and risk modules as in the main study, without additional tuning. YOLOv8-Seg detections/segmentations were associated with persistent identities via DeepSORT, after which the reactive and proactive modules generated warning decisions. We report representative UAV frames showing (1) detection and ID continuity, (2) proactive forward-projected boxes, and (3) incident-warning overlays (including T_{pred} and IoU threshold τ_{IoU}). These qualitative results provide end-to-end feasibility evidence for the proposed pipeline on UAV-captured imagery at the laboratory scale.

4. Result and Discussion

This section reports and interprets the performance of the proposed apron incident-warning framework in three stages: (i) quantitative MOT benchmarking on the MSFS sequence using the airplane-only ground truth, (ii) qualitative visual inspection of tracker behaviour (airplane-only and part-aware), and (iii) selection of the tracking backbone for the downstream reactive and proactive risk modules. The results are discussed with respect to tracker continuity, identity stability, and their implications for risk estimation.

4.1. Quantitative Comparison of MOT Algorithms on the MSFS Reenactment Dataset

Using the 997-frame MSFS sequence and the manually created MOTChallenge-format airplane-only ground truth, four trackers (ByteTrack, DeepSORT, StrongSORT, and BoT-SORT) were benchmarked under the fixed evaluation protocol described in Section 3.5. Table 5 reports MOT metrics (MOTA, MOTP, IDF1, precision/recall) and error counts (TP, FP, FN, ID switches).

Table 5. Quantitative comparison of MOT trackers on the 997-frame simulation-based apron sequence using airplane-only MOTChallenge-format ground truth (total GT instances = 1991).

Metric	ByteTrack	DeepSORT	StrongSORT	BoT-SORT
Tested Frames	997	997	997	997
MOTA (%)	82.82	92.77	82.92	83.02
MOTP (%)	90.59	88.74	90.65	90.6
IDF1 (%)	77.34	80.45	77.40	77.5
Precision (%)	100	99.52	100	100
Recall (%)	82.97	93.27	83.07	83.17
ID Switches	3	1	3	3
True Positive	1652	1857	1654	1656
False Positive	0	9	0	0
False Negative	339	134	337	335

DeepSORT achieved the strongest overall performance, with the highest MOTA (92.77%), highest recall (93.27%), lowest FN (134), and lowest ID switches (1). These results indicate better track continuity and identity stability than the other trackers, which is particularly important for downstream warning modules that depend on uninterrupted temporal evidence. By contrast, ByteTrack, StrongSORT, and BoT-SORT show similar but lower recall ($\approx 83\%$) and substantially higher FN counts (339, 337, and 335, respectively), implying more frequent track loss.

Although DeepSORT produced a small number of false positives (FP = 9; precision = 99.52%), this trade-off is acceptable in the present application because missed detections and track discontinuities are typically more harmful than occasional spurious tracks for proximity and trajectory-based risk assessment. Accordingly, the quantitative evidence supports selecting DeepSORT for the downstream incident-warning experiments.

4.2. Qualitative MOT Behaviour (Visual Evidence)

Following the quantitative benchmarking on the 997-frame MSFS reenactment sequence (Table 5), representative visual outputs are provided to examine tracker behaviour under apron-like close-proximity interactions. This subsection serves two purposes: (i) to qualitatively corroborate the metric differences in Table 5, and (ii) to inspect ID continuity and association robustness in situations where brief track interruptions may degrade downstream risk estimation.

4.2.1. Airplane-Only Tracking

Figure 5 compares the four trackers on a representative frame using airplane-only detections. Each panel shows YOLOv8-Seg detections with tracker-assigned identities (ByteTrack, DeepSORT, StrongSORT, and BoT-SORT) visualised as bounding boxes and ID labels. These examples are included to illustrate tracker outputs under identical inputs and complement the quantitative comparison in Table 5; the primary performance ranking remains based on the dataset-level, ground-truth-referenced metrics.

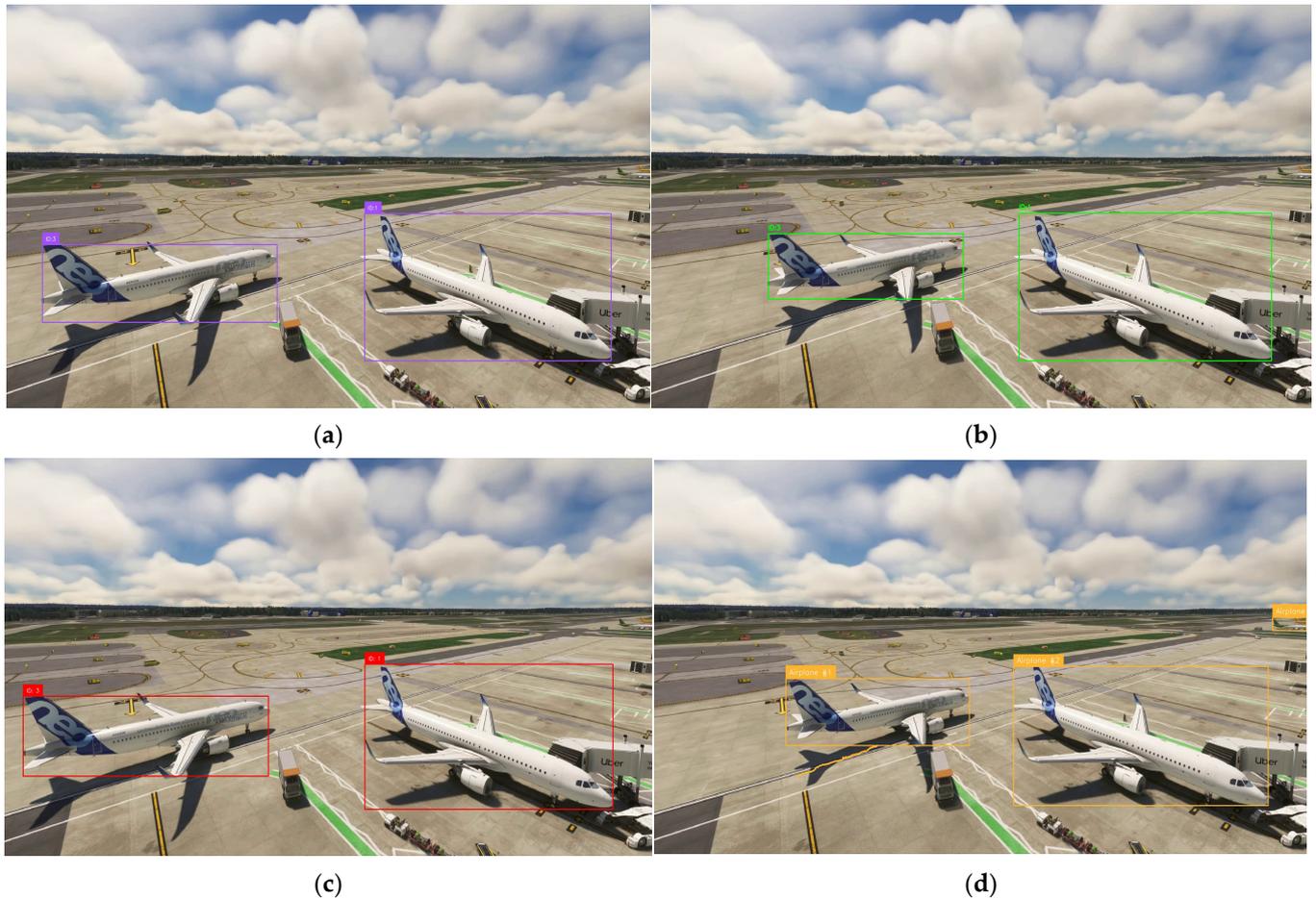


Figure 5. Airplane-only MOT outputs on the MSFS reenactment sequence: (a) ByteTrack, (b) DeepSORT, (c) StrongSORT, and (d) BoT-SORT. All panels use the same detector and identical inference settings; differences arise from the tracking association logic. The frames are selected from comparable phases of the interaction to support visual comparison.

4.2.2. Part-Aware Tracking

Figure 6 extends the qualitative analysis to part-aware tracking, where aircraft components (airplane, fuselage, nose, wing, tail) are assigned persistent IDs. This analysis demonstrates the framework's ability to maintain identities for safety-relevant aircraft parts (e.g., wing–tail interactions), which is more challenging than airplane-only tracking due to smaller targets, tighter spatial proximity, and increased sensitivity to partial occlusions. A representative StrongSORT segmentation overlay is also included for illustration only, showing that the optimised YOLOv8-Seg-based perception backbone supports both bounding-box and mask-based visual outputs.

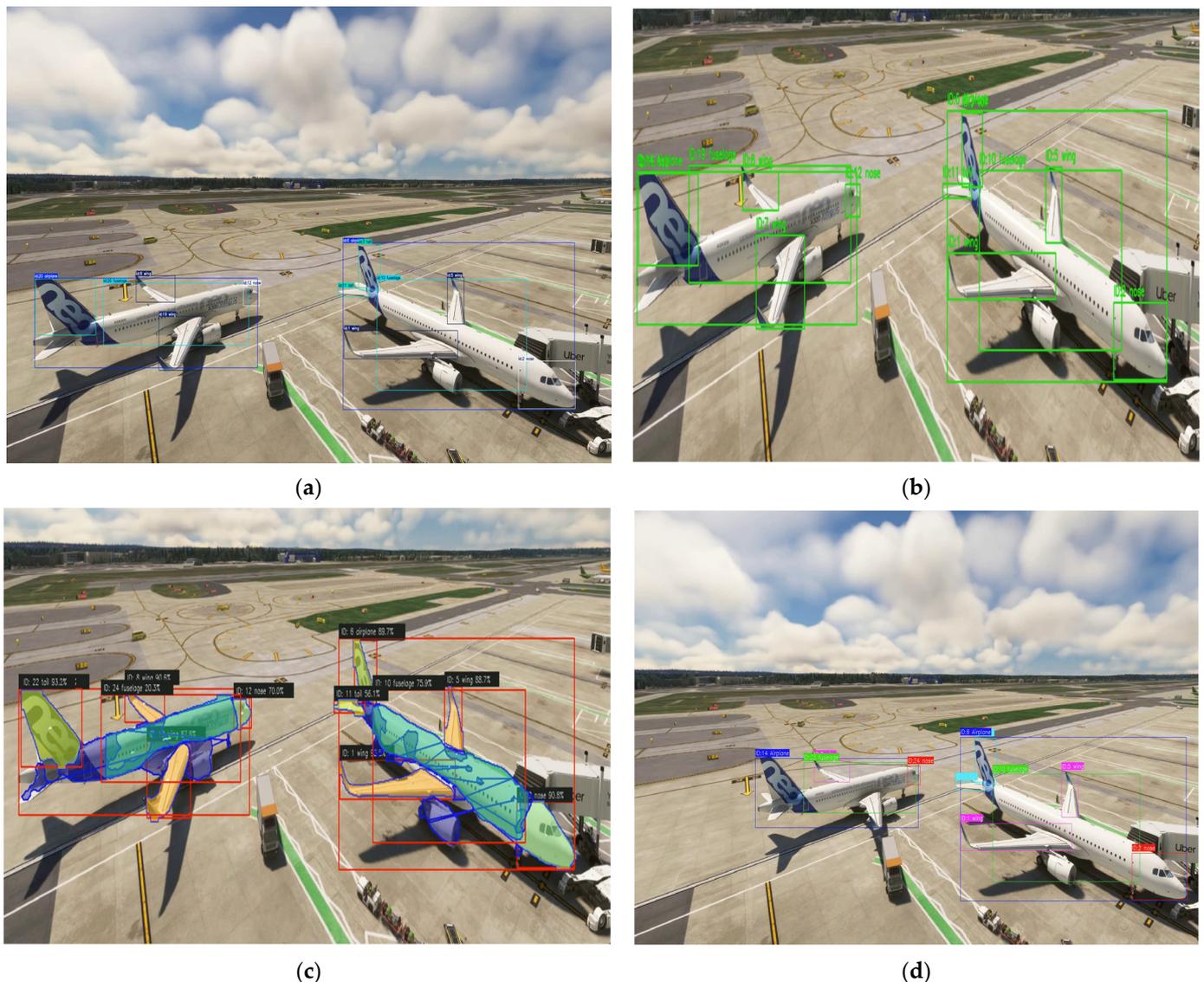


Figure 6. Part-aware MOT outputs on the MSFS reenactment sequence: (a) ByteTrack, (b) DeepSORT, (c) StrongSORT (representative mask overlay shown for illustration), and (d) BoT-SORT. Panels illustrate identity assignment for aircraft parts (airplane, fuselage, nose, wing, tail) during close-proximity motion. All trackers are driven by the same optimised YOLOv8-Seg-based detections and identical inference settings; the mask visualisation in (c) is included solely to demonstrate the availability of segmentation outputs from the perception backbone.

4.3. Final Tracker Selection for Risk Modules

Based on the quantitative benchmark (Table 5) and the representative visual outputs (Figures 5 and 6), DeepSORT was selected as the unified tracking backbone for the downstream incident-risk modules. This choice is driven by the requirements of risk estimation, as both the reactive proximity proxy and the proactive future-IoU proxy (Section 3.6) rely on uninterrupted temporal evidence and consistent identities for motion-history construction and conflict forecasting. Quantitatively, DeepSORT achieved the highest MOTA (92.77%), highest recall (93.27%), lowest FN (134), and lowest ID switches (1) (Table 5), indicating stronger track continuity and identity stability. This is consistent with the role of MOTA in the CLEAR MOT framework [60] and with the design of DeepSORT’s appearance-assisted association, which improves track persistence under close proximity and partial occlusions [56]. Given the fixed benchmarking protocol with identical detector

outputs and inference settings across trackers [66,67], DeepSORT provides the most reliable foundation for the incident-warning experiments in Sections 4.4–4.6.

4.4. Reactive Module (Mask-Distance Proxy) and Scenario Results

This subsection reports scenario-based results for the reactive mask-distance proxy module (Section 3.6.1), which evaluates instantaneous collision risk from the minimum Euclidean separation (pixel space) between segmentation masks of two tracked aircraft. After mask-ID association, contour-based minimum separation is computed using OpenCV point-to-contour distance operations; overlapping masks are assigned zero separation. Across all scenarios, the same detector-tracker backbone and reactive logic are used, while only the interaction geometry changes. Each scenario is illustrated using three canonical output states: Safe, Warning (“DANGER! DANGER!” overlay), and Collision/Contact.

4.4.1. Scenario 1(Reactive): Wing-Tail Contact (MSFS Incident Reenactment Inspired by a 2018 Event)

Scenario 1 in (Figure 7) produces a wing-tail contact geometry motivated by the 2018 Istanbul Ataturk Airport ground incident involving an Asiana Airlines A330 and a parked Turkish Airlines A321 [68]. This configuration is operationally relevant because minimum clearance often occurs at aircraft extremities (e.g., wingtip vs. tail), where axis-aligned bounding boxes can be overly conservative. The results show a clear progression from safe separation to warning and contact as the wingtip-to-tail clearance decreases, supporting the use of segmentation-mask distance as a tighter proxy for extremity-driven proximity.



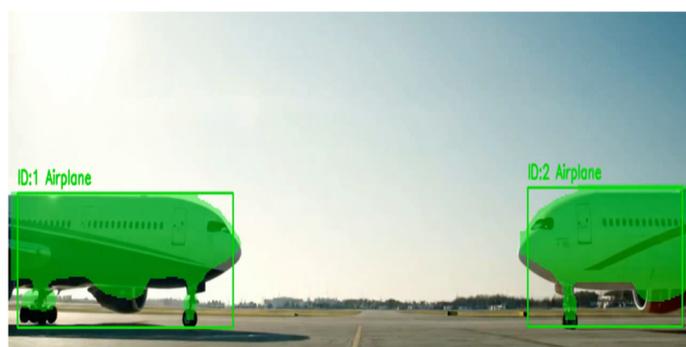


(c)

Figure 7. Reactive mask-distance proxy outputs for Scenario 1 (wing–tail interaction): (a) Safe (masks well separated), (b) Warning (labelled as “DANGER! DANGER!” in the overlay) (minimum mask-to-mask distance falls below the user-defined warning threshold), and (c) Collision/Contact (masks reach contact or the collision threshold is met). Colour coding follows the reactive module semantics (green: safe, yellow: warning, red: collision).

4.4.2. Scenario 2 (Reactive): Nose-to-Nose Convergence

Scenario 2 (Figure 8) evaluates the reactive proxy under a symmetric head-on (nose-to-nose) convergence geometry. Compared with Scenario 1, this configuration provides a less ambiguous minimum-distance cue because the separation is driven primarily by the frontal contours of both aircraft. The observed transition from Safe to Warning to Collision/Contact confirms that the reactive module responds consistently across a distinct interaction geometry using only instantaneous frame-wise mask geometry, without trajectory history or prediction.



(a)



(b)



(c)

Figure 8. Reactive mask-distance proxy outputs for Scenario 2 (nose-to-nose convergence): (a) Safe (masks well separated), (b) Warning (labelled as “DANGER! DANGER!” in the overlay) (minimum

mask distance crosses the user-defined warning threshold), and (c) Collision/Contact (mask contact or collision threshold met).

4.4.3. Scenario 3 (Reactive): Crowded Apron with Moving–Parked Interaction, Nose-to-Tail Convergence

Scenario 3 (Figure 9) examines the reactive module in a more cluttered apron-like setting with a moving–parked interaction culminating in nose-to-tail contact. This scenario is relevant because background clutter and nearby objects may increase segmentation variability and complicate association in close-proximity scenes. Despite these conditions, the module preserves the expected three-state behaviour (Safe → Warning → Collision/Contact), indicating that the mask-distance proxy remains interpretable in visually more congested scenes.

Taken together, Scenarios 1–3 show that the reactive module provides an interpretable frame-wise safety envelope controlled by user-defined pixel thresholds (Section 3.6.1). The scenario set also clarifies the operational meaning of the three output states: Safe (no immediate threat), Warning (close proximity requiring attention), and Collision/Contact (critical contact state).

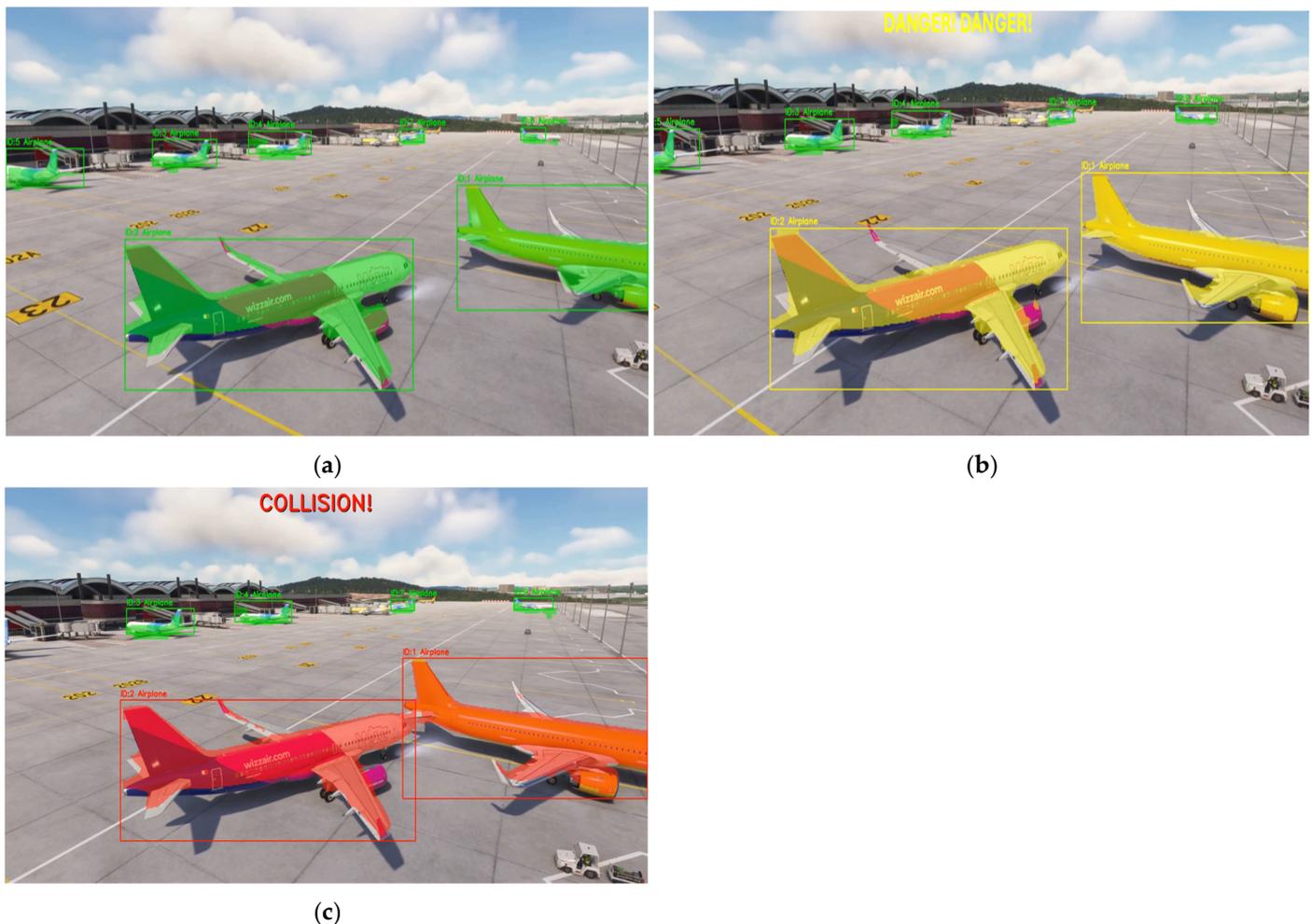


Figure 9. Reactive mask-distance proxy outputs for Scenario 3 (crowded apron; moving–parked nose-to-tail convergence): (a) Safe (masks well separated), (b) Warning (minimum mask distance crosses the user-defined warning threshold), and (c) Collision/Contact (nose-to-tail contact or collision threshold met).

4.5. Proactive Module (Future-IoU Proxy) and Scenario Results

This subsection reports scenario-based results for the proactive incident-warning module (Section 3.6.2), which issues early warnings from forecasted future overlap rather than instantaneous proximity. For each confirmed track, recent motion history is used to construct a forward-projected bounding box at a user-defined prediction horizon T_{pred} , and a warning is triggered when the IoU between projected boxes exceeds the configured threshold τ_{IoU} here ($\tau_{IoU} = 0.1$). In this section, IoU is used as a future-overlap proxy for horizon-based risk indication.

For consistent visual reporting, each scenario is illustrated with two representative phases: (i) no warning, where projected boxes remain separable, and (ii) warning, where the projected overlap exceeds τ_{IoU} . The displayed frames are selected from comparable interaction phases across scenarios and are not constrained to identical timestamps. Scenarios 1–3 reuse the same interaction geometries considered in Section 4.4, enabling a direct qualitative comparison between reactive (instantaneous mask-distance) and proactive (future-IoU) warning logic under matched scene conditions.

4.5.1. Scenario 1(Proactive): Wing–Tail Interaction (Simulation-Based, Incident-Inspired Geometry)

Scenario (Figure 10) uses the incident-inspired wing–tail interaction geometry and is intentionally configured in a part-aware proactive mode. Unlike the remaining proactive scenarios (Sections 4.5.2–4.5.5), which operate in airplane-only mode, this case demonstrates that the pipeline can (i) detect and track safety-critical components (e.g., wing and tail) and (ii) issue a component-specific early warning identifying the predicted conflicting parts within the selected horizon. In the no-warning phase (Figure 10a), the forward-projected bounding boxes of the relevant parts remain separable, and no warning is issued. As the moving aircraft approaches the stationary aircraft, the projected overlap exceeds the configured threshold τ_{IoU} , triggering an early-warning overlay (Figure 10b) for the selected horizon ($T_{pred} = 5$ s). In our implementation, the on-screen message explicitly specifies the involved track identities and component labels (e.g., “Collision between ID:23 (Wing) and ID:3 (Tail) in $\sim T_{pred} = 5$ s”), thereby providing a horizon-based, component-level alert.

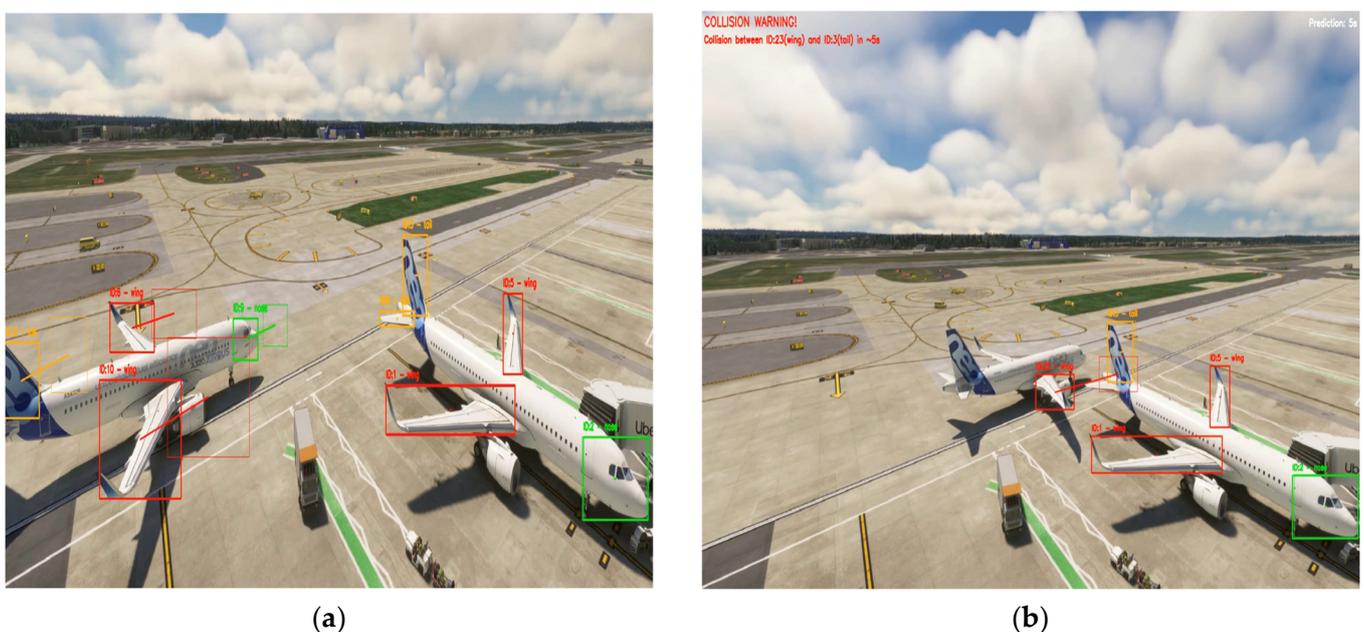


Figure 10. Proactive module outputs for Scenario 1 (MSFS simulation, wing–tail interaction; part-aware warning): (a) no warning, where the projected boxes remain separable; (b) early warning,

triggered when the projected overlap exceeds the user-defined IoU threshold $\tau_{IoU} = 0.1$ within the prediction horizon T_{pred} (here, 5 s). The overlay reports the predicted conflict together with the corresponding part labels and track identities.

4.5.2. Scenario 2 (Proactive): Nose-to-Nose Convergence

Scenario 2 (Figure 11) evaluates the proactive module under an airplane-only head-on (nose-to-nose) convergence geometry. This symmetric closing pattern provides a clear example of proactive behaviour, as warning generation is driven by forecasted overlap at the selected horizon rather than instantaneous contour separation. In the no-warning phase (Figure a), the projected overlap remains below the configured threshold ($\tau_{IoU} = 0.1$). As the approach continues, the projected boxes at $T_{pred} = 5$ overlap beyond τ_{IoU} , triggering the early-warning overlay (Figure 11b). This demonstrates that the module can issue an alert before contact-level proximity is reached, supporting its role as an early-warning layer complementary to reactive proximity cues.

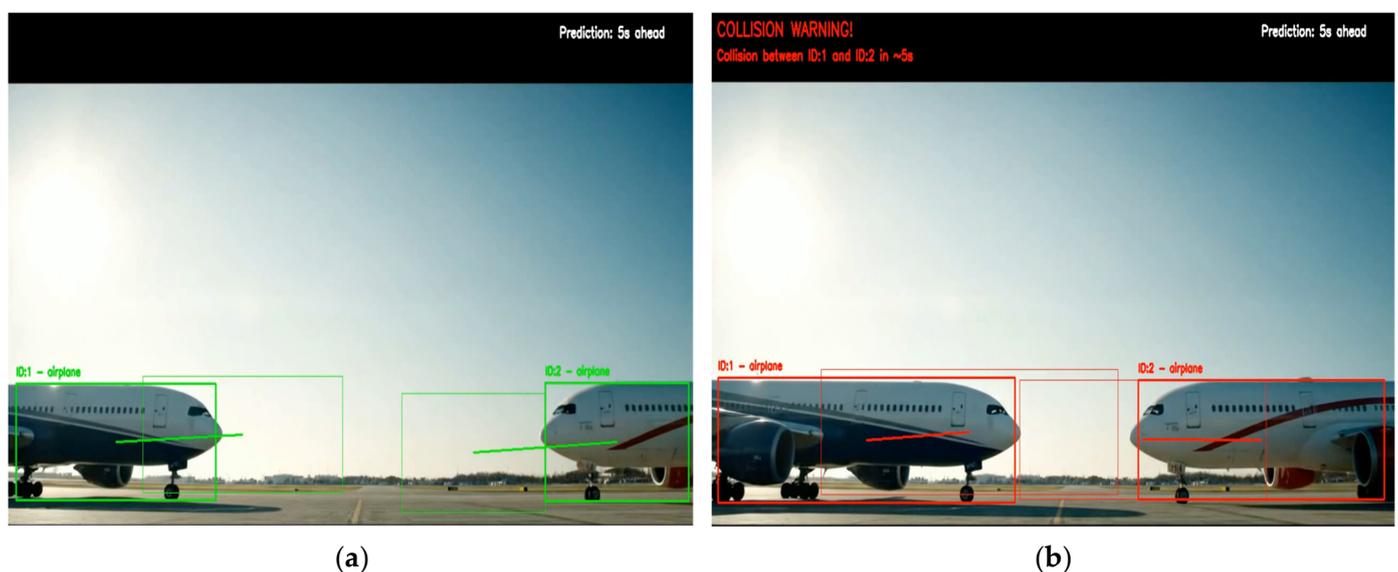


Figure 11. Proactive module outputs for Scenario 2 (nose-to-nose convergence): (a) no warning, (b) warning triggered when the projected overlap exceeds the user-defined IoU threshold $\tau_{IoU} = 0.1$ at the prediction horizon ($T_{pred} = 5$ s).

4.5.3. Scenario 3 (Proactive): Moving–Parked Interaction

Scenario 3 (Figure 12) considers a moving–parked interaction, which reflects a common apron configuration in which one aircraft remains stationary while another manoeuvres in close proximity in a cluttered scene. In the no-warning phase (Figure 12a), the forward-projected bounding boxes at the selected horizon remain sufficiently separable such that the projected overlap stays below the user-defined threshold τ_{IoU} , and the overlay indicates nominal operation. As the moving aircraft continues its approach, the projected boxes at $T_{pred} = 4$ s begin to overlap beyond τ_{IoU} (Figure 12b), triggering the early-warning overlay and highlighting the corresponding aircraft identities.

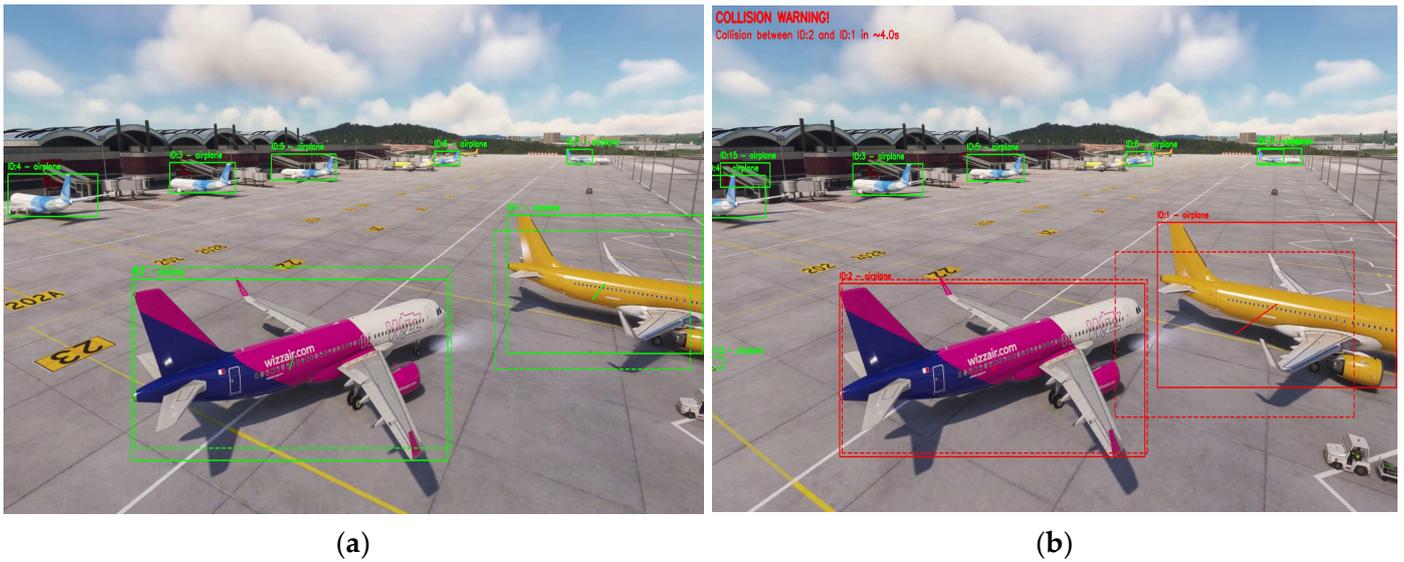


Figure 12. Proactive module outputs for Scenario 3 (moving–parked interaction): (a) no warning, (b) warning triggered when the projected overlap exceeds the user-defined IoU threshold $\tau_{IoU} = 0.1$ at the prediction horizon ($T_{pred} = 4$ s).

4.5.4. Scenario 4 (Proactive): Additional Synthetic Interaction

Scenario 4 (Figure 13) introduces an additional synthetic interaction evaluated only with the proactive module. It complements Scenarios 1–3 by varying viewpoint and scene appearance (higher, UAV-like elevated perspective and different aircraft models) while retaining a close-proximity wing–tail interaction geometry. In the no-warning phase (Figure 13a), the forward-projected boxes at the selected horizon remain separable and the projected overlap stays below $\tau_{IoU} = 0.1$. As the interaction progresses, the projected boxes overlap beyond τ_{IoU} , triggering the early-warning overlay and indicating a predicted conflict within the selected horizon T_{pred} (Figure 13b).

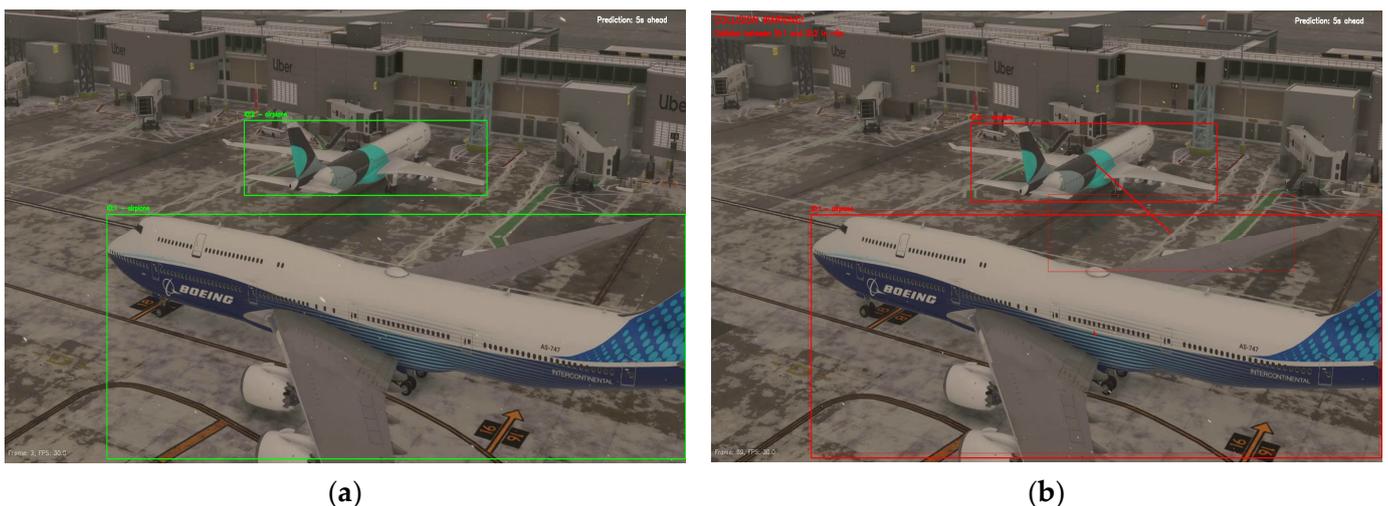


Figure 13. Proactive module outputs for Scenario 4 (additional synthetic wing–tail interaction; UAV-like overhead (elevated) viewpoint): (a) no warning, (b) warning triggered when the projected overlap exceeds the user-defined IoU threshold $\tau_{IoU} = 0.1$ at the prediction horizon.

Overall, Scenario 4 shows that the same future-overlap logic remains applicable under a different visual viewpoint and scene appearance, while also illustrating the operational role of the user-configurable parameters T_{pred} and τ_{IoU} in controlling alert sensitivity.

4.5.5. Scenario 5 (Proactive): Real CCTV Stream (Hong Kong International Airport)

Scenario 5 (Figure 14) evaluates the proactive module on real CCTV footage from Hong Kong International Airport, providing a qualitative operational check under visual conditions distinct from the synthetic MSFS-based scenarios. Due to the high, long-range viewpoint and the small apparent aircraft size, the experiment is conducted in airplane-only mode. Across representative early and late frames, the system detects multiple aircraft, preserves stable identities, and generates consistent forward projections. Since no imminent close-proximity interaction is present in the selected clip, the predicted overlaps remain below τ_{IoU} , and no warnings are issued, which is the expected behaviour for normal apron operations.

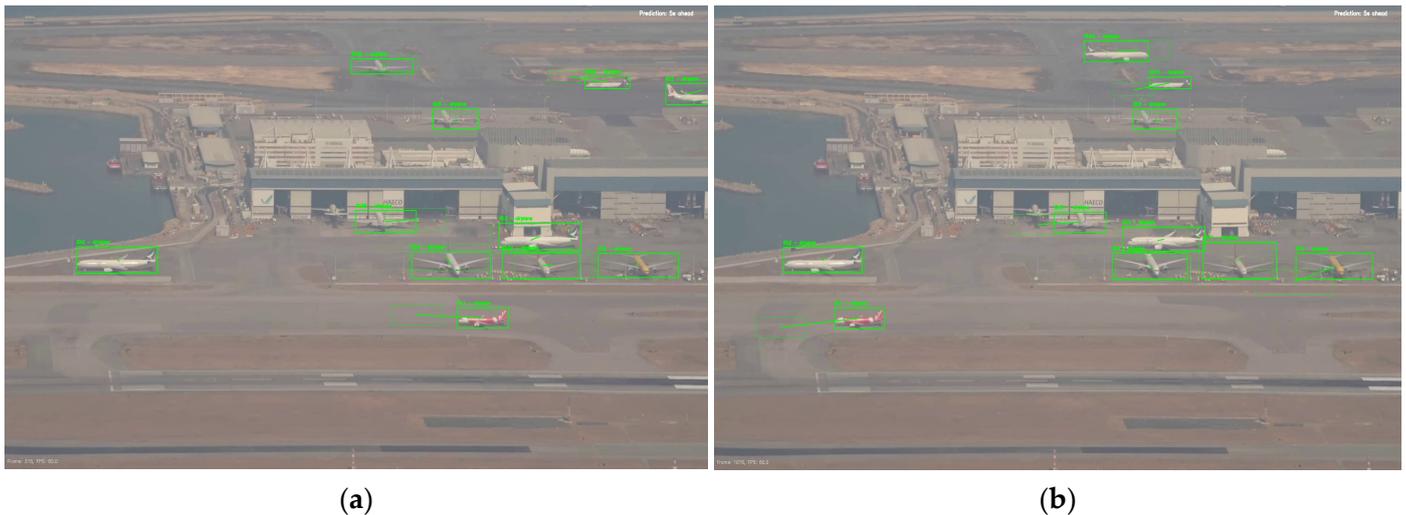


Figure 14. Proactive module outputs for Scenario 5 (real footage, Hong Kong International Airport): representative (a) early and (b) late frames illustrating airplane-only tracking with stable forward projections and no triggered warnings under normal apron activity.

Overall, Scenarios 1–4 demonstrate that the proactive module can issue horizon-based early warnings when the forecasted overlap exceeds the user-defined threshold $\tau_{IoU} = 0.1$, while Scenario 5 illustrates stable behaviour on real apron imagery without induced collision events, supporting the module's practical applicability under multi-aircraft operational scenes.

4.6. Scaled UAV/Laboratory Validation (Lab-Scale Unmanned Platform)

To complement the simulation-based reenactment and real-apron video evidence, we conducted a laboratory-scale unmanned-platform validation using UAV-acquired RGB footage of two diecast aircraft models under controlled indoor conditions. The UAV captured 1920×1080 video footage at 60 FPS from a fixed altitude of 1.2 m with a slightly oblique hovering viewpoint. One model remained stationary, while the other was moved along a repeatable path using a thin string. This elevated oblique viewpoint was chosen to approximate UAV-relevant observation geometry while preserving clear visibility of proximity/contact events. The videos were processed without additional tuning using the same perception-tracking backbone (YOLOv8-Seg + DeepSORT) and the same reactive/proactive risk logic described in Section 3.6. Two controlled scenarios were considered: (i) a no-incident close-pass interaction and (ii) a controlled wing-tail contact interaction. Results are reported qualitatively through representative frames and on-screen warning overlays, consistent with the role of this experiment as an end-to-end feasibility check on real UAV imagery.

4.6.1. Scenario A: No-Incident Close Pass (UAV Footage)

- (a) Reactive module. Figure 15 shows representative frames from the close-pass trial. Across the interaction, the reactive mask-distance proxy remains above the collision/contact threshold (and does not trigger the collision state); the frames illustrate stable detection and ID continuity during a controlled close-pass with no contact.

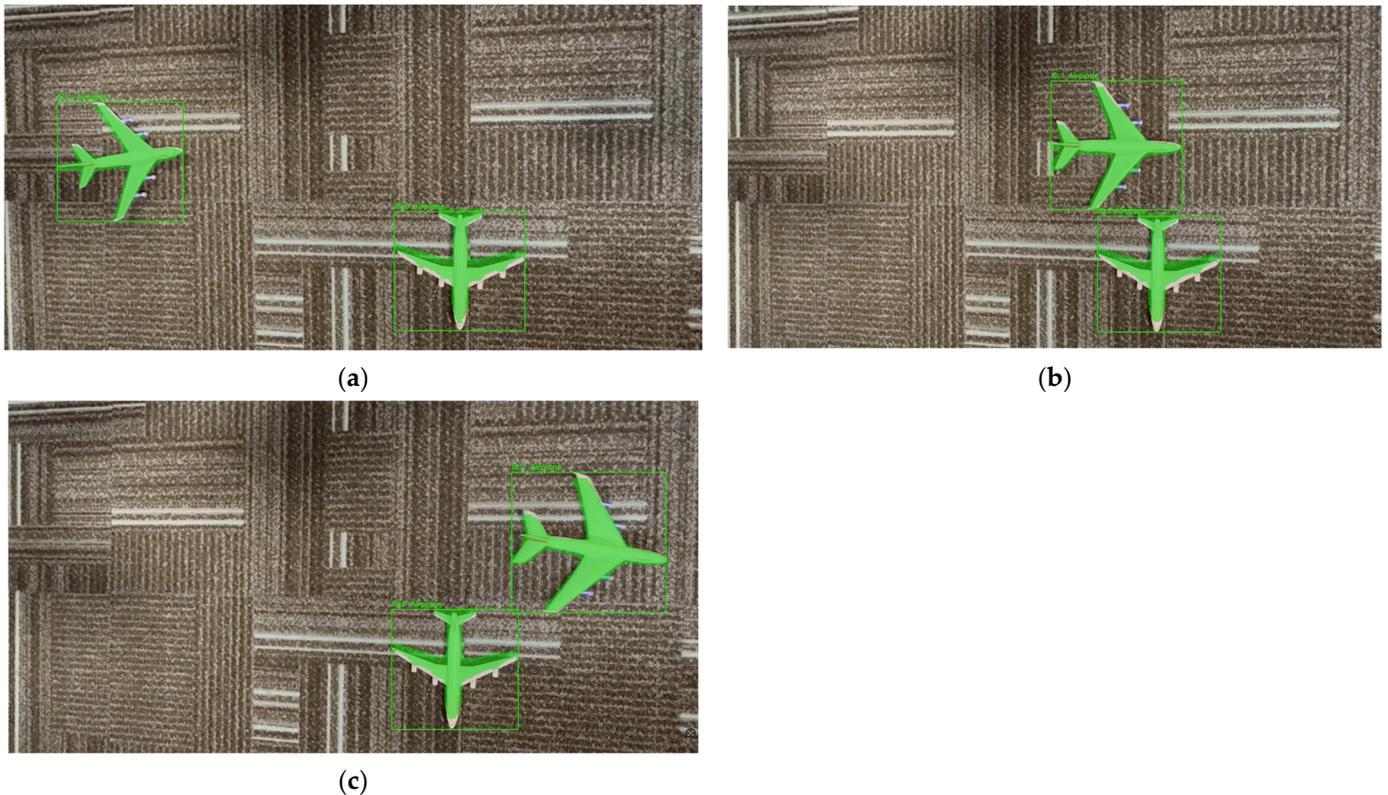


Figure 15. Reactive module outputs on UAV footage (hovering UAV at 1.2 m with slightly oblique viewpoint), Scenario A (no-incident close-pass): (a) early frame, (b) closest-approach frame, and (c) late frame after separation. Overlays indicate stable ID maintenance and a non-collision mask-distance state throughout the controlled no-contact interaction.

- (b) Proactive module. Figure 16 shows the corresponding proactive outputs for the same close-pass trial.

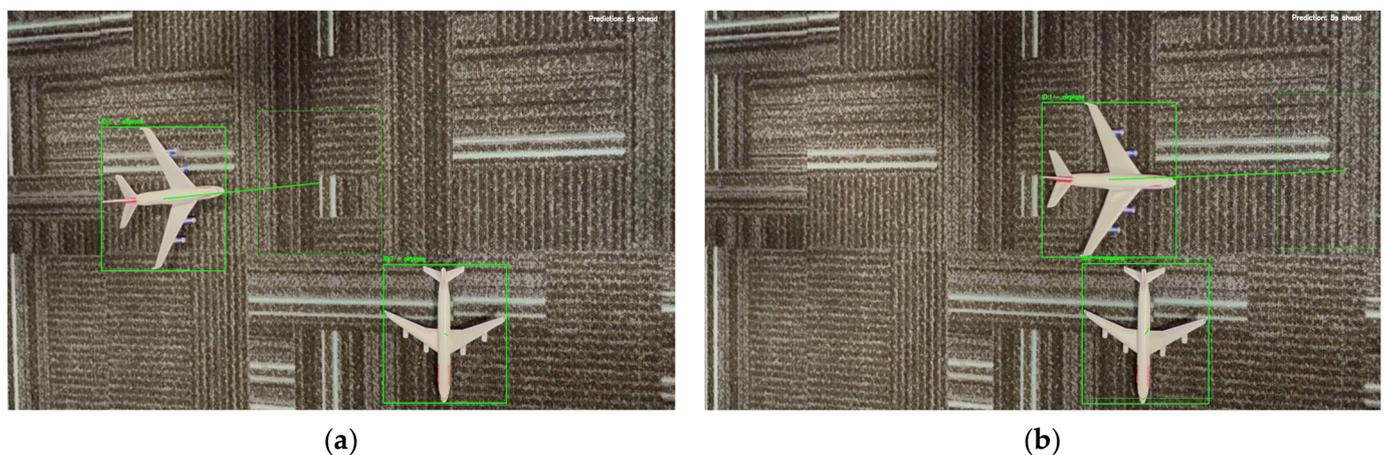


Figure 16. Proactive module outputs on UAV footage, Scenario A (no-incident close-pass): (a) early frame and (b) late frame. Forward-projected boxes remain separable at the selected horizon, the future-IoU proxy stays below the configured threshold, and no early-warning banner is triggered.

Because the interaction does not produce a predicted conflict within the configured horizon ($T_{\text{pred}} = 5 \text{ s}$), the forward-projected bounding boxes remain separable and the projected overlap proxy stays below the threshold $\tau_{\text{IoU}} = 0.1$; therefore, no early-warning banner is triggered. Together, Figures 15 and 16 indicate that the pipeline can process UAV-captured imagery, maintain ID continuity, and avoid false alerts in a controlled no-incident setting.

4.6.2. Scenario B: Controlled Wing–Tail Contact (UAV Footage)

- (a) Reactive module: Figure 17 shows the controlled wing–tail contact trial and illustrates the expected three-level behaviour of the mask-distance proxy: Safe at clear separation, Warning once the warning threshold is violated, and Collision/Contact when the collision criterion is met.

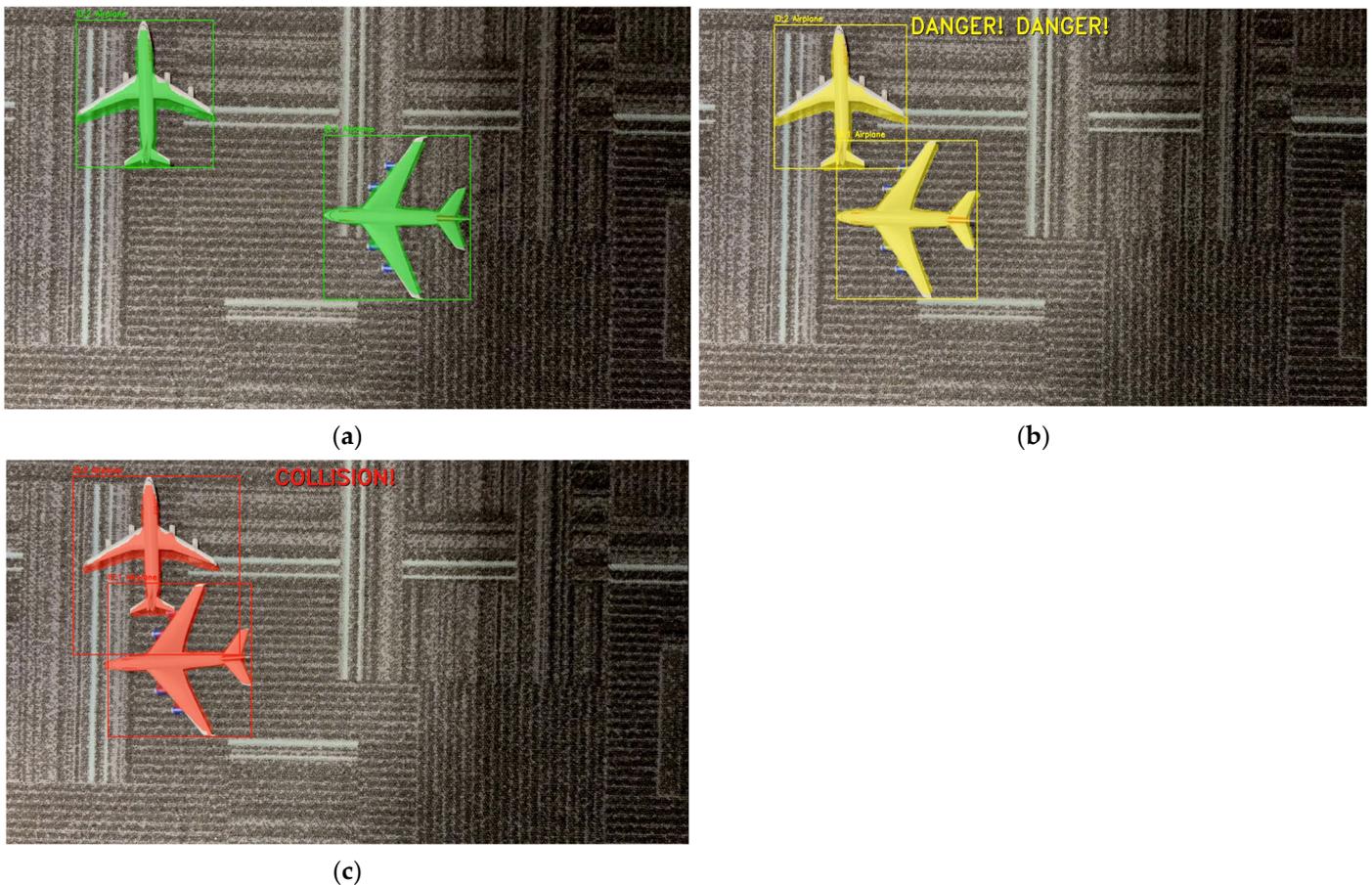


Figure 17. Reactive module outputs on UAV footage, Scenario B (controlled wing–tail contact): (a) Safe, (b) Warning (warning threshold crossed), and (c) Collision/Contact (contact condition reached). The sequence illustrates the frame-wise evolution of the mask-distance proxy and the corresponding colour-coded warning semantics.

- (b) Proactive module. Figure 18 shows that the proactive module can issue an early warning before physical contact by using short-horizon forward projections and triggering when the predicted IoU exceeds the configured threshold. In the early frame (Figure 18a), the projected boxes remain separated and no warning is displayed. In the warning phase (Figure 18b), the projected overlap exceeds the threshold, and a collision-warning banner is generated for the predicted interaction within the selected horizon. Overall, the lab-scale UAV results support the feasibility of applying

the same detection–tracking–warning pipeline to UAV-acquired video under controlled indoor conditions.

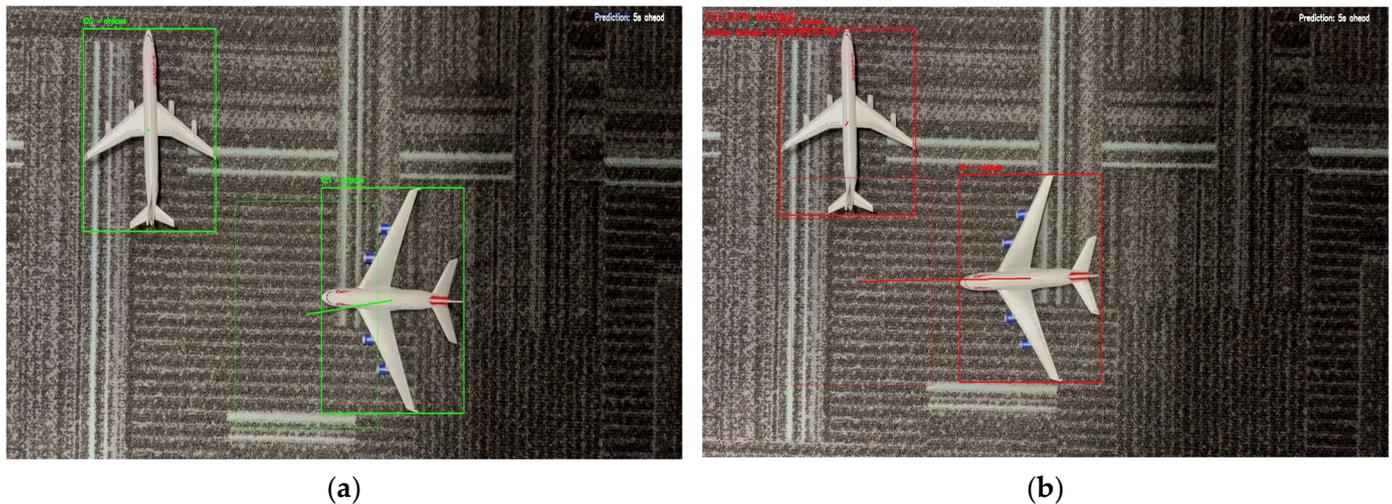


Figure 18. Proactive module outputs on UAV footage, Scenario B (controlled contact trial): (a) early frame (no warning) and (b) warning phase. A horizon-based warning is triggered when the projected IoU between forward-projected boxes exceeds the configured threshold, indicating an anticipated conflict within the selected prediction horizon prior to the physical contact instant.

4.7. Comparative Discussion: Reactive vs. Proactive

The two incident-risk modules are intentionally complementary because they operationalise risk over different time horizons and therefore exhibit different strengths and limitations. The reactive module is instantaneous: it evaluates the current frame only and triggers warnings when the pixel-domain proximity between segmentation masks violates predefined thresholds. In contrast, the proactive module is predictive: it estimates short-horizon future positions from recent motion history and flags a potential conflict when the projected overlap proxy (future IoU) exceeds a user-defined threshold.

This difference directly affects warning lead time and geometric fidelity. The reactive module can only trigger when aircraft become sufficiently close (or are in contact), which limits advance warning time but provides high spatial fidelity at the moment of interaction because it relies on segmentation-mask geometry rather than coarse box overlap. The proactive module, by design, can provide earlier alerts through the prediction horizon ($T_{\text{pred}} = 5$ s in this study), offering additional decision time when recent trajectories support reliable short-horizon extrapolation.

The modules also differ in parameterisation and operational tuning. Reactive behaviour is primarily controlled by pixel-distance thresholds defining safe, warning, and contact states, which makes configuration comparatively straightforward. Proactive behaviour depends on both the prediction horizon T_{pred} and the overlap threshold τ_{IoU} , and is also influenced by tracking persistence settings (e.g., maximum track age) that affect motion-history availability and the stability of pairwise future-overlap estimation.

Their failure modes are correspondingly different. Because the reactive module is based on current-frame segmentation geometry, it is less sensitive to abrupt or newly initiated motion (no prediction is required), but it may be affected by segmentation boundary noise or transient mask fragmentation at close range. The proactive module depends on (i) stable identity maintenance and (ii) the validity of the short-horizon motion model; its performance can degrade under abrupt manoeuvres, speed changes, or strongly non-linear interactions that violate the local linear-motion assumption.

The results across the synthetic incident scenarios, the real CCTV clip, and the laboratory-scale UAV experiments support this complementary interpretation. In the synthetic scenarios, the proactive module provides horizon-based early warnings in conflict-like motion patterns, whereas the reactive module provides close-range confirmation based on instantaneous geometry. In the real CCTV non-conflict scenario, the proactive module remains conservative (i.e., no warning is issued when projected overlaps remain below τIoU). In the controlled indoor UAV tests with diecast aircraft, both modules remain operational under UAV-acquired video, providing additional feasibility evidence for unmanned-platform deployment at laboratory scale.

Taken together, these findings motivate a dual-module operational concept: a proactive module for earlier warning when short-horizon motion is sufficiently predictable, and a reactive module for high-fidelity confirmation near contact. Although the two modules are developed and evaluated as independent methods in this study, the results indicate that a future integrated deployment could benefit from their complementary behaviour.

4.8. Practical Implications for UAV-Based Apron Safety

Ramp/ground occurrences remain a persistent operational and economic risk, and industry safety reports indicate that a substantial proportion of aircraft damage events occur during ground movement and ramp/taxi phases, motivating improved situational awareness and conflict prevention on the movement area [69,70]. In this context, the proposed vision-based framework can be viewed as a cost-effective add-on safety layer that complements existing surveillance by converting video streams into identity-aware incident warnings. Relative to fixed viewpoints (e.g., tower/CCTV), UAV-mounted cameras can provide flexible elevated views that help reduce occlusions and enable targeted monitoring of gates, stands, and congested apron areas where close-proximity interactions are more likely. UAV platforms may also support multimodal sensing (e.g., RGB/thermal/LiDAR), which has been increasingly discussed for aerodrome movement-area monitoring and may improve robustness under challenging visibility and lighting conditions [71].

Operationally, the reactive module offers an intuitive close-range safety envelope, while the proactive module provides earlier decision support by indicating which tracked entities are predicted to conflict within a user-defined horizon, thereby supporting timely supervisory intervention. Within a smart-airport context, such warnings could be routed to apron control or supervisory systems as decision-support cues, consistent with prior discussions on digitally mediated apron-safety monitoring [72,73]. These implications are framed as having deployment potential rather than immediate operational readiness; the present contribution is a validated perception-and-warning foundation for future UAV-enabled monitoring workflows, subject to regulatory, safety-assurance, and operational constraints.

4.9. Limitations and Future Work

While the proposed framework demonstrates consistent incident-warning behaviour across MSFS simulation-based scenarios, real apron footage, and laboratory-scale UAV experiments, several aspects delimit the current scope and motivate future extensions.

4.9.1. Limitation

Image-plane risk proxies: The reactive module uses a pixel-space mask-distance proxy, and the proactive module uses a future-IoU proxy between forward-projected bounding boxes. These design choices support real-time operation and interpretability; however, because both proxies are defined in the image plane, their numeric thresholds are not inherently viewpoint-invariant. As a result, thresholds may require re-tuning when the sensing geometry changes (e.g., camera viewpoint/altitude, field of view, or

intrinsic parameters), particularly if pixel-to-metric consistency is not enforced. Consequently, the specific reactive thresholds used in this study ($\tau c = 40$ px, $\tau w = 80$ px) are not claimed to be directly transferable across viewpoints without geometry-aware scaling.

Adverse visibility/image degradations: The proposed pipeline relies on RGB imagery; therefore, degradations that are common in apron operations (e.g., rain, fog, low illumination, glare, motion blur, and compression artefacts) can reduce segmentation fidelity and propagate to tracking stability (e.g., identity switches), affecting both the reactive mask-distance proxy and proactive forecasting. Accordingly, the results reported in this paper characterise performance under nominal-visibility conditions; improvements in robustness under adverse visibility can be pursued through standard enhancement front-ends and/or training/domain adaptation with representative degraded-visibility data (and complementary sensing, where available).

Locally smooth-motion assumption for short-horizon forecasting: The proactive module uses a short-horizon, constant-velocity extrapolation updated frame-by-frame from recent track history. This design is intended as a lightweight early-warning proxy and is most reliable when motion remains locally smooth over the chosen prediction horizon; under abrupt manoeuvres (e.g., rapid braking/acceleration or sharp turns), forecast accuracy can degrade. In such regimes, a shorter user-defined horizon can be selected, while evaluating more expressive or uncertainty-aware predictors under controlled non-linear motion profiles is a natural extension for future work.

IoU-threshold sensitivity for proactive warnings: While we report $\tau IoU = 0.10$ as a conservative early-warning setting, a systematic sensitivity analysis across different interaction geometries/viewpoints (false-alarm vs. missed-warning trade-offs) remains a natural extension for future work.

Constraints on UAV trials near airports: Large-scale airside UAV data collection is operationally constrained because most aerodromes are protected by flight restriction zones (FRZs) and/or controlled airspace, and UAV operations typically require explicit authorisation and coordination with aerodrome operators and air traffic services. This limits in situ experimentation in operational areas and motivates staged evaluation strategies, such as scaled laboratory validation and permissioned field trials, as a practical pathway toward deployment [74,75].

Operational and regulatory considerations: This study targets apron/taxiway and stand-area monitoring rather than runway operations. Any UAV-enabled deployment within an aerodrome would require site-specific authorisation, an operational risk assessment, and an approved safety case under the applicable aviation regulatory framework, which are beyond the scope of this paper. Importantly, the proposed detection-tracking-warning pipeline is not tied solely to UAV sensing and can also be executed using fixed elevated sensors (e.g., apron surveillance cameras), providing a smart-airport monitoring layer that complements existing safety procedures without introducing UAV-specific operational constraints.

Onboard deployment constraints (latency/power): The experiments in this paper were executed in an offline GPU setting (e.g., Google Colab) to ensure consistent benchmarking. While the proposed pipeline is algorithmically portable, real-time onboard UAV deployment would require meeting latency and power budgets and may benefit from lightweight model variants, GPU/edge accelerators (e.g., Jetson-class devices), and optimisation strategies such as TensorRT/ONNX compilation, quantisation, and resolution/frame-rate trade-offs. A dedicated real-time profiling study on embedded hardware is a natural next step.

4.9.2. Future Work

Permissioned UAV validation at representative geometries: Building on the scaled protocol, an important next step is to conduct permissioned UAV recording campaigns in airport environments subject to flight restrictions (e.g., FRZs or controlled airspace). Such trials could quantify performance across representative altitudes, view angles, and traffic patterns while complying with aerodrome procedures and safety requirements [76].

Pixel-to-metric calibration for operational thresholds: Future studies can map image-plane thresholds to metric safety envelopes via camera calibration (intrinsic/extrinsic) and scene geometry estimation (e.g., ground-plane homography or multi-view 3D reconstruction). Where depth sensing is available (e.g., LiDAR/RGB-D), integrating geometry-aware 3D reconstruction or point-cloud-based ranging can provide more precise metric separation estimates and improve the transferability of safety thresholds across view-points, with the proposed 2D detection–tracking–warning pipeline remaining as the core decision layer.

Connected decision-support integration: A natural extension is to integrate warning outputs into airport digital decision-support workflows, for example, by routing alerts to an apron/tower interface, logging events for safety review, and supporting post-incident analysis. This direction aligns with broader efforts toward digitally augmented monitoring and hybrid digital tower concepts [72,73].

5. Conclusions

This paper presented a vision-based, platform-agnostic framework for apron ground-incident warning that combines optimised deep perception with MOT and complementary risk logic. To address the scarcity of annotated incident footage, we introduced an MSFS simulation-based incident reenactment inspired by the 2018 Asiana–Turkish Airlines wing-to-tail event and derived a labelled 997-frame MOT dataset, enabling reproducible evaluation under incident-relevant geometries.

Under identical detection conditions, benchmarking of four MOT algorithms showed that DeepSORT provides the most reliable identity continuity for apron monitoring, achieving 92.77% MOTA and 93.27% recall on the MSFS reenactment benchmark (airplane-only MOTChallenge-format ground truth) with one ID switch. Building on this tracker selection, we demonstrated a dual-mode safety mechanism: (i) a reactive mask-distance proxy that produces immediate proximity alerts at close range, and (ii) a proactive horizon-based module that forecasts short-term conflicts using forward-projected boxes and a user-defined future-IoU trigger. Together, these modules address both near-instantaneous clearance loss and short-horizon anticipation, providing interpretable warnings with tuneable sensitivity.

The study addresses the Research Questions as follows: (RQ1) MSFS simulation-based incident reenactments can yield incident-representative, label-efficient datasets when real incident footage is limited; (RQ2) under the evaluated close-proximity setting, DeepSORT offers the most stable tracking backbone among the tested trackers; (RQ3) combining reactive and proactive criteria supports consistent warning behaviour across diverse interaction geometries; and (RQ4) the framework demonstrates feasibility across MSFS reenactment scenarios and real apron videos (no-incident control), providing a practical foundation for future unmanned-platform integration. In addition to the MSFS reenactment scenarios and the real apron no-incident control footage, laboratory-scale UAV experiments provided additional qualitative validation on UAV-captured imagery under controlled conditions, demonstrating consistent behaviour in both no-incident close-pass and controlled contact trials.

Author Contributions: Conceptualisation, E.C.B. and H.A.-R.; methodology, E.C.B.; software, E.C.B.; formal analysis, E.C.B.; investigation, E.C.B.; data curation, E.C.B.; writing—original draft preparation, E.C.B.; writing—review and editing, E.C.B., K.B., and H.A.-R.; supervision, H.A.-R. and K.B.; project administration, E.C.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no specific external project funding. The first author is supported by a PhD scholarship from the Ministry of National Education of Türkiye (no grant number). This scholarship did not directly fund the work reported in this paper.

Data Availability Statement: The dataset and annotations generated in this study are available on Roboflowat: <https://app.roboflow.com/ecb-wba09/mot-challenge-dataset/browse?query-Text=&pageSize=50&startingIndex=0&browseQuery=true> (accessed on 28 January 2026).

Acknowledgments: The authors thank colleagues at the Department of Electronic and Electrical Engineering, Brunel University of London, for providing laboratory facilities and technical support during data acquisition and experiments.

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

Abbreviations

The following abbreviations are used in this manuscript:

ADS-B	Automatic Dependent Surveillance–Broadcast
AP	Average Precision
CCTV	Closed-Circuit Television
CNN	Convolutional Neural Network
COCO	Common Object in Context
CV	Computer Vision
DL	Deep Learning
FN	False Negative
FOD	Foreign Object Debris
FEF-R-CNN	Feature Enhancement Faster R-CNN
FGVC	Fine-Grained Visual Classification of Aircraft
FP	False Positive
FPS	Frames Per Second
HIL	Hardware-in-the-Loop
IATA	International Air Transport Association
IDF1	Identification F1 Score
IDSW	Identity Switches
IoU	Intersection-over-Union
LiDAR	Light Detection and Ranging
LQR	Linear Quadratic Regulator
LSTM	Long Short-Term Memory
mAP	Mean Average Precision
MOT	Multi-Object Tracking
MOTA	Multi-Object Tracking Accuracy
MOTP	Multi-Object Tracking Precision
MSFS	Microsoft Flight Simulator
NMS	Non-Maximum Suppression
RGB	Red, Green, Blue
SMR	Surface Movement Radar
SRGAN	Super-Resolution Generative Adversarial Network
TP	True Positive
UAV	Unmanned Aerial Vehicle
YOLO	You Only Look Once
U-Net	U-shaped Network

Appendix A. Reactive Warning Pipeline Pseudocode (Algorithm A1)

Algorithm A1. Reactive mask-based proximity warning procedure

Input: video frames $\{I_t\}$, detector D (YOLOv8-Seg), tracker T (DeepSORT), thresholds $\tau_c = 40$, $\tau_w = 80$;

Output: annotated frames and warning state per frame;

- 1 For each frame I_t do;
- 2 Run detector: $\{(b_k, M_k)\}_{k=1}^K \leftarrow D(I_t)$;
- 3 Extract boxes only: $\{b_k\} \rightarrow$ tracker input;
- 4 Update tracker: $\{(id_i, \hat{b}_i)\}_{i=1}^N \leftarrow T(\{b_k\})$;
- 5 Associate each track id_j with a mask M_j via nearest-neighbour matching in bounding-box centre space;
- 6 Initialise all aircraft states as Safe
- 7 For every aircraft pair (i, j) , compute $d_{\min}(M_i, M_j)$ using contour distance (set $d_{\min} = 0$ if masks overlap)
- 8 If $d_{\min} \leq \tau_c$, set level = Collision; else if $d_{\min} \leq \tau_w$, set level = Warning; else level = Safe
- 9 Update the states of aircraft i and j with the maximum severity level
- 10 Render masks and labels with the corresponding colour and export the frame
- 11 End for

Appendix B. Proactive Warning Pipeline Pseudocode (Algorithm A2)

Algorithm A2. Proactive trajectory-based future-IOU warning procedure

Input: video frames $\{I_t\}$; detector D (YOLOv8-Seg); tracker T (DeepSORT); history length $N = 30$; velocity window $M = \min(10, |H_i|)$; prediction horizon T_{pred} (s); frame rate f ; IOU threshold τ_{IOU} ; tracker age limit max_age

Output: per-frame warning set W_t ; visual overlays;

- 1 For each frame I_t do
- 2 Run detector: $\{(b_k, M_k)\}_{k=1}^{K_d} \leftarrow D(I_t)$
- 3 Extract boxes only: $\{b_k\} \rightarrow$ tracker input
- 4 Update tracker: $\{(id_i, b_i(t))\}_{i=1}^{n_t} \leftarrow T(\{b_k\})$
- 5 For each active track i do
- 6 Update history buffer $H_i \leftarrow H_i \cup \{b_i(t)\}$ (deque, max length N); set $\text{last_seen}_i \leftarrow t$
- 7 Remove stale tracks if $t - \text{last_seen}_i > \text{max_age}$
- 8 Compute centre $c_i(t)$ from $b_i(t)$; estimate $\bar{v}_i(t)$ from the most recent M centre differences
- 9 Compute horizon in frames: $K_h \leftarrow \lfloor T_{\text{pred}} f \rfloor$
- 10 Predict future centre: $\hat{c}_i(t + K_h) \leftarrow c_i(t) + K_h \cdot \bar{v}_i(t)$
- 11 Construct future box $\hat{b}_i(t + K_h)$ by translating $b_i(t)$ to centre $\hat{c}_i(t + K_h)$ while keeping width/height fixed
- 12 End for
- 13 Initialise $W_t \leftarrow \emptyset$
- 14 For each unordered pair of active tracks (i, j) do
- 15 Compute $\text{IoU}(\hat{b}_i(t + K_h), \hat{b}_j(t + K_h))$
- 16 If $\text{IoU}(\hat{b}_i(t + K_h), \hat{b}_j(t + K_h)) > \tau_{\text{IOU}}$ then $W_t \leftarrow W_t \cup \{(id_i, id_j)\}$
- 17 End for
- 18 Render current boxes, future boxes, and centre-to-centre motion lines; if $W_t \neq \emptyset$, overlay warning text and involved IDs; export frame
- 19 End for

References

1. International Air Transport Association (IATA). IATA Forecasts 5.2 billion Passengers in 2026. Available online: <https://www.iata.org/en/pressroom/2025-releases/2025-12-09-01/> (accessed on 17 December 2025).
2. Flight Safety Foundation (FSF). *2022 Safety Report*; Flight Safety Foundation: Alexandria, VA, USA, 2023; p. 11.
3. Chen, X.; Gao, Z.; Chai, Y. The Development of Air Traffic Control Surveillance Radars in China. In *Proceedings of the 2017 IEEE Radar Conference, RadarConf 2017, Seattle, WA, USA, 8–12 May 2017*; IEEE Press: New York, NY, USA, 2017; pp. 1776–1784. <https://doi.org/10.1109/RADAR.2017.7944495>.
4. Federal Aviation Administration (FAA). Automatic Dependent Surveillance–Broadcast (ADS-B). Available online: https://www.faa.gov/about/office_org/headquarters_offices/avs/offices/afx/afs/afs400/afs410/ads-b (accessed on 16 May 2025).
5. García, I.; Martínez-Prieto, M.A.; Bregón, A.; Álvarez, P.C.; Díaz, F. Towards a Scalable Architecture for Flight Data Management. In *DATA 2017—Proceedings of the 6th International Conference on Data Science, Technology and Applications*; Science and Technology Publications: Setubal, Portugal, 2017; pp. 263–268. <https://doi.org/10.5220/0006473402630268>.
6. Yang, A.; Tan, X.; Baek, J.; Wong, D.S. A New ADS-B Authentication Framework Based on Efficient Hierarchical Identity-Based Signature with Batch Verification. *IEEE Trans. Serv. Comput.* **2017**, *10*, 165–175. <https://doi.org/10.1109/TSC.2015.2459709>.
7. Brassel, H.; Zouhar, A.; Fricke, H. 3D Modeling of the Airport Environment for Fast and Accurate LiDAR Semantic Segmentation of Apron Operations. In *Proceedings of the 2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC), San Antonio, TX, USA, 11–15 October 2020*; IEEE Press: New York, NY, USA, 2020. <https://doi.org/10.1109/DASC50938.2020.9256495>.
8. Ding, M.; Ding, Y.-Y.; Wu, X.-Z.; Wang, X.-H.; Xu, Y.-B. Action Recognition of Individuals on an Airport Apron Based on Tracking Bounding Boxes of the Thermal Infrared Target. *Infrared Phys. Technol.* **2021**, *117*, 103859. <https://doi.org/10.1016/j.infrared.2021.103859>.
9. Li, J.; Dong, X. Intelligent Surveillance of Airport Apron: Detection and Location of Abnormal Behavior in Typical Non-Cooperative Human Objects. *Appl. Sci.* **2024**, *14*, 6182. <https://doi.org/10.3390/app14146182>.
10. Atlioglu, M.C.; Gokhan, K.O.C. An AI Powered Computer Vision Application for Airport CCTV Users. *J. Data Sci.* **2021**, *4*, 21–26.
11. Munyer, T.; Brinkman, D.; Huang, C.; Zhong, X. Integrative Use of Computer Vision and Unmanned Aircraft Technologies in Public Inspection: Foreign Object Debris Image Collection. In *Proceedings of the 22nd Annual International Conference on Digital Government Research, Omaha, NE, USA, 9–11 June 2021*; Association for Computing Machinery: New York, NY, USA, 2021; pp. 437–443. <https://doi.org/10.1145/3463677.3463743>.
12. ICAO International Civil Aviation Organization. FOD Management Programme. Available online: <https://www.icao.int/ESAF/Documents/meetings/2024/Aerodrome%20Certification%20Workshop%20Luanda%20Angola%2013-17%20May%202024/Presentations/FOD%20Management%20Programme.pdf> (accessed on 19 June 2024).
13. Mo, Y.; Wang, L.; Hong, W.; Chu, C.; Li, P.; Xia, H. Small-Scale Foreign Object Debris Detection Using Deep Learning and Dual Light Modes. *Appl. Sci.* **2024**, *14*, 2162. <https://doi.org/10.3390/app14052162>.
14. Kucuk, N.S.; Aygun, H.; Dursun, O.O.; Toraman, S. Detection and Classification of Foreign Object Debris (FOD) with Comparative Deep Learning Algorithms in Airport Runways. *Signal Image Video Process.* **2025**, *19*, 316. <https://doi.org/10.1007/s11760-025-03901-6>.
15. Bajpai, A. Attire-Based Anomaly Detection in Restricted Areas Using YOLOv8 for Enhanced CCTV Security. *arXiv* **2024**, arXiv:2404.00645. <https://doi.org/10.48550/arXiv.2404.00645>.
16. Yıldız, S.; Aydemir, O.; Memiş, A.; Varlı, S. A Turnaround Control System to Automatically Detect and Monitor the Time Stamps of Ground Service Actions in Airports: A Deep Learning and Computer Vision Based Approach. *Eng. Appl. Artif. Intell.* **2022**, *114*, 105032. <https://doi.org/10.1016/j.engappai.2022.105032>.
17. Bingol, E.C.; Al-Raweshidy, H. From Benchmarking to Optimisation: A Comprehensive Study of Aircraft Component Segmentation for Apron Safety Using YOLOv8-Seg. *Appl. Sci.* **2025**, *15*, 11582. <https://doi.org/10.3390/app152111582>.
18. International Air Transport Association. Ground Operations Safety. Available online: <https://www.iata.org/en/programs/ops-infra/ground-operations/safety/> (accessed on 12 December 2025).
19. Transportation Safety Board of Canada (TSB). *Air Transportation Safety Investigation Report A18O0002: Ground Collision, Fire and Evacuation (Sunwing Airlines Inc./WestJet Airlines Ltd., Toronto/Lester B. Pearson International Airport, 5 January 2018)*; TSB: Gatineau, QC, Canada, 2018. Available online: <https://www.tsb.gc.ca/sites/default/files/rapports-reports/aviation/A18O0002/eng/a18o0002.pdf> (accessed on 18 February 2026).
20. Republic of Türkiye Ministry of Transport and Infrastructure, Transport Safety Investigation Center (UEIM). *Aircraft Accident Investigation Final Report: TC-JMM/HL7792 (Istanbul Atatürk Airport, 13 May 2018)*; UEIM: Ankara, Türkiye, 2018. Available

- online: <https://ulasimemniyeti.uab.gov.tr/uploads/pages/hava-araci/tc-jmm-hl-7792-nihai-rapor.pdf> (accessed on 18 February 2026). (In Turkish)
21. Dutch Safety Board. *Collision during pushback: Boeing 747 vs. Boeing 787 at Amsterdam Schiphol Airport (13 February 2019)*; Report No. 2019009; Dutch Safety Board: The Hague, The Netherlands, 2022. Available online: https://onderzoeksraad.nl/wp-content/uploads/2023/11/collision_during_pushback.pdf (accessed on 18 February 2026).
 22. Republic of Türkiye Ministry of Transport and Infrastructure, Transport Safety Investigation Center (UEIM). Announcement of Aircraft Serious Incident Investigation Report: Boeing B777-3F2ER, TC-LJE (Istanbul Airport, 22.05.2019); Official Gazette of the Republic of Türkiye, 9 May 2020, Issue 31122. Available online: <https://legalbank.net/resmi-gazete-dosyalar/20200509/2fc86da702544428956839374546f015.pdf> (accessed on 18 February 2026). (In Turkish)
 23. Aviation Safety Network (ASN). Incident Airbus A350-941 A7-AMH, Monday 10 February 2020. Available online: <https://aviation-safety.net/wikibase/232870> (accessed on 18 February 2026).
 24. National Transportation Safety Board (NTSB). *Aviation Investigation Final Report (Accident No. DCA21LA133, Location: Chicago, Illinois, USA; Date: 21 May 2021)*; NTSB: Washington, DC, USA, 2022. Available online: <https://data.nts.gov/carol-reppen/api/Aviation/ReportMain/GenerateNewestReport/103134/pdf> (accessed on 18 February 2026).
 25. Air Accidents Investigation Branch (AAIB). AAIB Investigation to Boeing 777-300(ER), HL-7782 and Boeing 757-256, TF-FIK: Collision on the Ground, London Heathrow Airport, 28 September 2022. AAIB Bulletin 6/2023; AAIB-28692; 2023. Available online: https://assets.publishing.service.gov.uk/media/6464dc75e14070000cb6e10a/Boeing_777-300_ER_HL-7782_and_Boeing_757-256_TF-FIK_06-23.pdf (accessed on 12 December 2025).
 26. Air Accidents Investigation Branch (AAIB). *AAIB Bulletin: 8/2024 EI-EGD AAIB-29638*; Air Accidents Investigation Branch (AAIB): London, UK, 2024. Available Online: https://assets.publishing.service.gov.uk/media/66b23fc5a3c2a28abb50dddc/AAIB_Bulletin_8-2024.pdf (accessed on 12 December 2025).
 27. Air Accidents Investigation Branch (AAIB). *AAIB Bulletin: 11/2024 G-VDIA and G-XWBC AAIB-29945*; Air Accidents Investigation Branch (AAIB): London, UK, 2024; pp. 40–42. Available Online: https://assets.publishing.service.gov.uk/media/67337a55bfc4a11a06122090/Boeing_787-9_G-VDIA_Airbus_A350-1041_G-ZWBC_11-24.pdf (accessed on 12 December 2025).
 28. Federal Aviation Administration (FAA). *Statements on Aviation Accidents and Incidents. Commercial Aviation/Chicago, Illinois (January 8, 2025): American Airlines Flight 1979 Struck the Tail of United Airlines Flight 219 While Taxiing at Chicago O'Hare International Airport*; FAA: Washington, DC, USA, 2025.
 29. Federal Aviation Administration (FAA). *FAA Statements on Aviation Accidents and Incidents; Commercial Aviation/Washington DC (April 10, 2025): The wingtip of American Airlines Flight 5490 Struck American Airlines Flight 4522 on a Taxiway at Ronald Reagan Washington National Airport*; FAA: Washington, DC, USA, 2025.
 30. Lukin, K.; Mogila, A.; Vyplavin, P.; Galati, G.; Pavan, G. Novel Concepts for Surface Movement Radar Design. *Int. J. Microw. Wirel. Technol.* **2009**, *1*, 163–169. <https://doi.org/10.1017/S1759078709000233>.
 31. Federal Aviation Administration (FAA). Aeronautical Information Manual (AIM), Chapter 4: Air Traffic Control, Section 5: Surveillance Systems. Available online: https://www.faa.gov/air_traffic/publications/atpubs/aim_html/chap4_section_5.html (accessed on 13 December 2025).
 32. Shang, F.; Wang, B.; Li, T.; Tian, J.; Cao, K.; Guo, R. Adversarial Examples on Deep-Learning-Based ADS-B Spoofing Detection. *IEEE Wirel. Commun. Lett.* **2020**, *9*, 1734–1737. <https://doi.org/10.1109/LWC.2020.3002914>.
 33. Yang, H.; Zhou, Q.; Yao, M.; Lu, R.; Li, H.; Zhang, X. A Practical and Compatible Cryptographic Solution to ADS-B Security. *IEEE Internet Things J.* **2019**, *6*, 3322–3334. <https://doi.org/10.1109/JIOT.2018.2882633>.
 34. Kenk, M.A.; Hassaballah, M. DAWN: Vehicle Detection in Adverse Weather Nature Dataset. *arXiv* **2020**, arXiv:2008.05402. <https://doi.org/10.17632/766ygrbt8y.3>.
 35. Phat, T.V.; Alam, S.; Liliith, N.; Tran, P.N.; Binh, N.T. Deep4Air: A Novel Deep Learning Framework for Airport Airside Surveillance. In *Proceedings of the 2021 IEEE International Conference on Multimedia and Expo Workshops, ICMEW 2021, Shenzhen, China, 5–9 July 2021*; IEEE Press: New York, NY, USA, 2021. <https://doi.org/10.1109/ICMEW53276.2021.9456005>.
 36. Zeng, B.; Ming, D.; Ji, F.; Yu, J.; Xu, L. Top-Down Aircraft Detection in Large-Scale Scenes Based on Multi-Source Data and FEF-R-CNN. *Int. J. Remote Sens.* **2022**, *43*, 1108–1130. <https://doi.org/10.1080/01431161.2022.2034194>.
 37. Shi, T.; Gong, J.; Hu, J.; Sun, Y.; Bao, G. Progressive Class-Aware Instance Enhancement for Aircraft Detection in Remote Sensing Imagery. *Pattern Recognit.* **2025**, *164*, 111503. <https://doi.org/10.1016/j.patcog.2025.111503>.

38. Zhu, M.; Xu, Y.; Ma, S.; Li, S.; Ma, H.; Han, Y. Effective Airplane Detection in Remote Sensing Images Based on Multilayer Feature Fusion and Improved Nonmaximal Suppression Algorithm. *Remote Sens.* **2019**, *11*, 1062. <https://doi.org/10.3390/rs11091062>.
39. Tahir, A.; Adil, M.; Ali, A. Rapid Detection of Aircrafts in Satellite Imagery Based on Deep Neural Networks. *arXiv* **2021**, arXiv:2104.11677. <https://doi.org/10.48550/arXiv.2104.11677>.
40. Wang, Y.Y.; Wu, H.; Shuai, L.; Peng, C.; Yang, Z. Detection of Plane in Remote Sensing Images Using Super-Resolution. *PLoS ONE* **2022**, *17*, e0265503. <https://doi.org/10.1371/journal.pone.0265503>.
41. Thai, P.; Alam, S.; Lilith, N.; Nguyen, B.T. A Computer Vision Framework Using Convolutional Neural Networks for Airport Airside Surveillance. *Transp. Res. Part. C Emerg. Technol.* **2022**, *137*, 103590. <https://doi.org/10.1016/j.trc.2022.103590>.
42. Zhou, W.; Cai, C.; Zheng, L.; Li, C.; Zeng, D. ASSD-YOLO: A Small Object Detection Method Based on Improved YOLOv7 for Airport Surface Surveillance. *Multimed. Tools Appl.* **2023**, *83*, 55527–55548. <https://doi.org/10.1007/s11042-023-17628-4>.
43. Li, W.; Liu, J.; Mei, H. Lightweight Convolutional Neural Network for Aircraft Small Target Real-Time Detection in Airport Videos in Complex Scenes. *Sci. Rep.* **2022**, *12*, 14474. <https://doi.org/10.1038/s41598-022-18263-z>.
44. Zhou, W.; Cai, C.; Li, C.; Xu, H.; Shi, H. AD-YOLO: A Real-Time YOLO Network with Swin Transformer and Attention Mechanism for Airport Scene Detection. *IEEE Trans. Instrum. Meas.* **2024**, *73*, 5036112. <https://doi.org/10.1109/TIM.2024.3472805>.
45. Yilmaz, B.; Karşlıgil, M.E. Detection of Airplane and Airplane Parts from Security Camera Images with Deep Learning. In Proceedings of the 2020 28th Signal Processing and Communications Applications Conference, SIU 2020, Gaziantep, Turkey, 5–7 October 2020; IEEE Press: New York, NY, USA, 2020; pp. 21–24. <https://doi.org/10.1109/SIU49456.2020.9302118>.
46. Utomo, S.; Sulistyaningrum, D.R.; Setiyono, B.; Nasution, A.H.I. Image Augmentation For Aircraft Parts Detection Using Mask R-CNN. In Proceedings of the 2024 International Conference on Smart Computing, IoT and Machine Learning, SIML 2024, Surakarta, Indonesia, 6–7 June 2024; IEEE Press: New York, NY, USA, 2024; pp. 186–192. <https://doi.org/10.1109/SIML61815.2024.10578281>.
47. Thomas, J.; Kuang, B.; Wang, Y.; Barnes, S.; Jenkins, K. Advanced Semantic Segmentation of Aircraft Main Components Based on Transfer Learning and Data-Driven Approach. *Vis. Comput.* **2025**, *41*, 4703–4722. <https://doi.org/10.1007/s00371-024-03686-8>.
48. Ahmed, M.; Maher, A.; Bai, X. Aircraft Tracking in Aerial Videos Based on Fused RetinaNet and Low-Score Detection Classification. *IET Image Process.* **2022**, *17*, 687–708. <https://doi.org/10.1049/ipr2.12665>.
49. Su, Z.; Wan, G.; Zhang, W.; Guo, N.; Wu, Y. An Integrated Detection and Multi-Object Tracking Pipeline for Satellite Video Analysis of Maritime and Aerial Objects. *Remote Sens.* **2024**, *16*, 724. <https://doi.org/10.3390/rs16040724>.
50. Xu, J.; Ding, M.; Zhang, Z.Z.; Xu, Y.B.; Wang, X.H.; Zhao, F. Vision-Based Automatic Collection of Nodes of In/Off Block and Docking/Undocking in Aircraft Turnaround. *Appl. Sci.* **2023**, *13*, 7832. <https://doi.org/10.3390/app13137832>.
51. Mazzeo, P.L.; Manica, A.; Distante, C. UAV Multi-Object Tracking by Combining Two Deep Neural Architectures. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; LNCS; Springer: Cham, Switzerland, 2023; Volume 14233, pp. 257–268. https://doi.org/10.1007/978-3-031-43148-7_22.
52. Zhu, H.; Xu, Y.; Xu, Z.; Jiyuan, L.; Zhang, W. Study on Aircraft Wing Collision Avoidance through Vision-Based Trajectory Prediction; SAE Technical Papers; SAE International: Warrendale, PA, USA, 2024. <https://doi.org/10.4271/2024-01-2310>.
53. Sun, J.; Tang, X.; Shao, Q. A Collision Risk Assessment Method for Aircraft on the Apron Based on Petri Nets. *Appl. Sci.* **2024**, *14*, 9128. <https://doi.org/10.3390/app14199128>.
54. Gaikwad, P.; Mukhopadhyay, A.; Muraleedharan, A.; Mitra, M.; Biswas, P. Developing a Computer Vision Based System for Autonomous Taxiing of Aircraft. *Aviation* **2023**, *27*, 248–258. <https://doi.org/10.3846/aviation.2023.20588>.
55. Ywet, N.L.; Maw, A.A.; Lee, J.W. R-YOLO: Enhancing Takeoff/Landing Safety in UAM Vertiports with Deep Learning Model. *IEEE Access* **2023**, *13*, 89045–89058. <https://doi.org/10.1109/ACCESS.2025.3571435>.
56. Wojke, N.; Bewley, A.; Paulus, D. Simple Online and Realtime Tracking with a Deep Association Metric. In Proceedings of the International Conference on Image Processing, ICIP 2017, Beijing, China, 17–20 September 2017; IEEE Press: New York, NY, USA, 2017; pp. 3645–3649. <https://doi.org/10.1109/ICIP.2017.8296962>.
57. Du, Y.; Zhao, Z.; Song, Y.; Zhao, Y.; Meng, H. StrongSORT: Make DeepSORT Great Again. *IEEE Trans. Multimed.* **2023**, *25*, 8725–8737. <https://doi.org/10.1109/TMM.2023.3240881>.
58. Zhang, Y.; Sun, P.; Jiang, Y.; Yu, D.; Weng, F. ByteTrack: Multi-Object Tracking by Associating Every Detection Box. In Proceedings of the Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); LNCS; Springer: Cham, Switzerland, 2022; Volume 13682, pp. 1–21.
59. Aharon, N.; Orfaig, R.; Bobrovsky, B.-Z. BoT-SORT: Robust Associations Multi-Pedestrian Tracking. *arXiv* **2022**, arXiv:2206.14651.

60. Bernardin, K.; Stiefelhagen, R. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics. *EURASIP J. Image Video Process.* **2008**, *2008*, 246309. <https://doi.org/10.1155/2008/246309>.
61. Ristani, E.; Solera, F.; Zou, R.; Cucchiara, R.; Tomasi, C. Performance Measures and a Data Set for Multi-Target, Multi-Camera Tracking. In *Proceedings of the Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; LNCS; Springer: Cham, Switzerland, 2016; Volume 9914, pp. 17–35.
62. Ultralytics Instance Segmentation—Ultralytics YOLO Docs. Available online: <https://docs.ultralytics.com/tasks/segment/> (accessed on 19 February 2026).
63. Boyd, S.; Vandenberghe, L. *Convex Optimization*; Cambridge University Press: Cambridge, UK, 2004.
64. OpenCV. OpenCV: Structural Analysis and Shape Descriptors. Available online: https://docs.opencv.org/4.x/d3/dc0/group__imgproc__shape.html (accessed on 23 February 2026).
65. Naminas, K. Object Detection Metrics: IoU, Precision, Recall, MAP. Available online: <https://labeleyourdata.com/articles/object-detection-metrics> (accessed on 19 February 2026).
66. Leal-Taixé, L.; Milan, A.; Reid, I.; Roth, S.; Schindler, K. MOTChallenge 2015: Towards a Benchmark for Multi-Target Tracking. *arXiv* **2015**, arXiv:1504.01942.
67. Dendorfer, P.; Ošep, A.; Milan, A.; Schindler, K.; Cremers, D.; Reid, I.; Roth, S.; Leal-Taixé, L. MOTChallenge: A Benchmark for Single-Camera Multiple Target Tracking. *Int. J. Comput. Vis.* **2021**, *129*, 845–881. <https://doi.org/10.1007/s11263-020-01393-0>.
68. Aviation Safety Network (ASN). Aircraft Accident Airbus A321-231 TC-JMM Istanbul-Ataturk Airport (IST). Available Online: <https://aviation-safety.net/wikibase/319655>. (accessed on 26 December 2025).
69. International Air Transport Association (IATA). IATA Annual Safety Report-2024 Recommendations for Accident Prevention. 2024. Available Online: https://www.iata.org/contentassets/95e933e1ad794068812f073cf883cb08/recommendations-for-accident-prevention-2024_final.pdf (accessed on 26 December 2025).
70. Flight Safety Foundation. Ground Accident Prevention (GAP). Available online: <https://flightsafety.org/toolkits-resources/past-safety-initiatives/ground-accident-prevention-gap/> (accessed on 26 December 2025).
71. Kovács, B.; Vörös, F.; Vas, T.; Károly, K.; Gajdos, M.; Varga, Z. Safety and Security-Specific Application of Multiple Drone Sensors at Movement Areas of an Aerodrome. *Drones* **2024**, *8*, 231. <https://doi.org/10.3390/drones8060231>.
72. Baláž, M.; Kováčiková, K.; Novák, A.; Vaculík, J. The Application of Internet of Things in Air Transport. *Transp. Res. Procedia* **2023**, *75*, 60–67. <https://doi.org/10.1016/j.trpro.2023.12.008>.
73. Zhang, J.; Tian, X.; Pan, J.; Chen, Z.; Zou, X. A Field Study on Safety Performance of Apron Controllers at a Large-Scale Airport Based on Digital Tower. *Int. J. Environ. Res. Public Health* **2022**, *19*, 1623. <https://doi.org/10.3390/ijerph19031623>.
74. Civil Aviation Authority Aerodromes, Heliports and Spaceports. Available online: <https://www.caa.co.uk/drones/moving-on-to-more-advanced-flying/airspace/aerodromes-heliports-and-spaceports/> (accessed on 19 November 2025).
75. Civil Aviation Authority (CAA). The Drone and Model Aircraft Code. 2024. Available online: <https://www.caa.co.uk/media/5d10tmqu/the-drone-code-march-2024.pdf> (accessed on 19 December 2025).
76. Civil Aviation Authority Airspace Restrictions. Available online: <https://www.caa.co.uk/drones/moving-on-to-more-advanced-flying/airspace/airspace-restrictions/> (accessed on 29 December 2025).

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.