AI-Powered Runway Safety: YOLO11-Based Detection of Foreign Object Debris

Phani Kiran Yellakara[1], Srinivasarao Surapu[2]

[1,2]*Collins Aerospace, Bangalore, Karnataka, 560064, India*

phani.kiran.yellakara@collins.com
srinivasarao.surapu@collins.com

ABSTRACT

Runway safety is a critical aspect of aviation operations, with Foreign Object Debris (FOD) posing severe risks to aircraft during takeoff and landing. Incidents such as the Air France Flight 4590 (Concorde) crash have demonstrated the devastating impact of undetected FOD. Manual inspection methods remain the standard but are time-consuming, error-prone, and limited by environmental conditions. With global FOD-related costs reaching billions annually, there is a clear need for intelligent, automated detection systems. This study introduces an artificial intelligence (AI)-powered approach using computer vision and deep learning to detect and classify runway debris in real time. The work compares different generations of object detection models, showing progressive improvements from earlier YOLO versions to the latest architecture. While YOLOv8 demonstrated notable accuracy gains, the most effective results were obtained using YOLO11, which delivered the highest detection performance when trained on a composite dataset of open-source and custom runway imagery. Model robustness was further improved through data augmentation, class balancing, and annotation using the Computer Vision Annotation Tool (CVAT). The trained YOLO11 model was deployed via a web-based application with an Angular frontend and Flask backend, enabling fast, precise detection and a user-friendly interface. While the current system is optimized for image-based detection, future work will address real-time video integration, edge device deployment, and interoperability with airport safety systems. Limitations include partial coverage of real-world conditions and small debris detection challenges, which are being addressed through ongoing dataset expansion and model refinement. This work is a crucial step toward a comprehensive PHM framework for aviation infrastructure.

Keywords: Runway Safety, Foreign Object Debris, YOLO11, Object Detection, Deep Learning, Aviation Safety, FOD-A Dataset, Computer Vision, PHM, Edge AI, Artificial Intelligence, CVAT

1. INTRODUCTION

Runway safety is a paramount concern in aviation, and the presence of foreign object debris (FOD) presents a significant threat to aircraft and operational integrity. Manual inspection, the traditional method for FOD detection, is plagued by limitations such as infrequency, reliance on human sight, and susceptibility to environmental factors. This has led to a clear demand for automated, intelligent detection systems to enhance safety and reduce financial losses, which are estimated at over $22.7 billion annually (Foreign Object Debris Detection System Cost-Benefit Analysis Report).

This project addresses this critical need by developing an AI-powered system for real-time detection and classification of runway FOD. The core of the system is the YOLO11 deep learning model, which is a state-of-the-art object detection algorithm known for its efficiency and high accuracy. The system is designed to provide rapid, precise alerts, thereby mitigating the risks associated with undetected debris. The real-time, data-driven insights from this system are positioned as a foundational element for a robust Prognostics and Health Management (PHM) framework for airport runway infrastructure.

2. LITERATURE REVIEW

Existing FOD detection methods, including manual inspections and sensor-based systems, have proven to be limited in their effectiveness and scalability. Deep learning-based computer vision techniques offer a superior alternative, ensuring enhanced safety, operational efficiency, and automation. Modern CNN-based detectors like YOLO and Faster R-CNN provide superior generalization by extracting deep semantic features. The choice of algorithm for this project was based on a comparative analysis of key performance aspects, as shown in Table 1.

The YOLO (You Only Look Once) family of algorithms is particularly well-suited for real-time applications due to its single-stage, anchor-based approach. While previous studies have explored models like YOLOv5 and YOLOv8 for FOD detection, the application of the more advanced YOLO11 remains underexplored. Prior work has explored improved versions of earlier models, such as Song, Yuan and Ding (2022) using an enhanced YOLOv4 for track FOD detection, highlighting the continuous evolution of these models to improve performance. YOLO11 is optimized for FOD detection through targeted architectural enhancements. The C2PSA module uses spatial attention to focus on relevant regions and preserve positional details, which is crucial for distinguishing small debris from runway textures. This is complemented by the C3k2 block, which captures fine-grained details, and SPPF, which accelerates multi-scale processing. Together, these improvements yield a higher mAP with fewer parameters, resulting in a highly accurate and computationally efficient model ideal for real-time FOD detection on various hardware platforms. These features make YOLO11 an ideal choice for this project, balancing computational efficiency with state-of-the-art accuracy.

Aspect	**Yolo**	**Fast R-CNN**	**SSD**
Speed	Real-time (30-120 FPS)	Slow (5-10 FPS)	Faster than Fast R-CNN (~20-60 FPS)
Accuracy	High (State-of-the-Art)	High (But slower inference)	Moderate
Small Object Detection	Excellent (with multi-scale features)	Good	Moderate
Ease of Deployment	Highly portable, supports edge devices	Challenging, requires high compute	Moderate
Semantic Segmentation	Unified support	Not Supported	Not Supported
Resource Efficiency	Optimized for hardware constraints	Computationally expensive	Moderate

Table 1. Comparative Analysis of Object Detection Algorithms

3. DATA PREPARATION AND PROCESSING

The development of a high-performance, deep learning-based object detection model for safety-critical applications is predicated on a meticulously prepared dataset. This phase focused on establishing a robust, diverse, and well-balanced dataset to ensure the YOLO11 model's generalization capabilities and mitigate the risk of biased predictions.

3.1. Dataset Collection and Annotation

To create a comprehensive training resource, a combined dataset was curated from two primary sources. The open-source Foreign Object Debris in Airports (FOD-A) Dataset provided a foundational set of images captured under varied lighting and weather conditions. This was augmented with a custom-collected image dataset to introduce further diversity and enhance the model's ability to generalize to new, unseen environments. All images were meticulously annotated using the Computer Vision Annotation Tool (CVAT), a process that involved manually creating precise bounding boxes and assigning class labels to each instance of FOD. This hybrid approach ensured a rich dataset that is both extensive and tailored to real-world operational challenges.

3.2. Preprocessing and Data Partitioning

A critical step in preparing the data for the YOLO11 framework was the conversion of the original PASCAL VOC annotations from the FOD-A dataset into the required YOLO format. A custom Python script was developed to parse the XML files and normalize the bounding box coordinates, ensuring seamless compatibility.

The combined dataset was then partitioned into training, validation, and testing sets with a stratified split ratio of 75%, 15%, and 10%, respectively. Stratified splitting was employed to ensure that the class distribution of FOD objects was proportionally represented across all three sets. This rigorous approach prevents data leakage and ensures that the model's performance metrics are evaluated on a truly representative and unbiased sample of the data.

3.3. Data Balancing and Augmentation

Class imbalance is a significant challenge in real-world object detection, particularly for rare but critical objects. To address this, a structured balancing methodology was implemented. Instead of simple oversampling, a k-th sampling approach was used to proportionally reduce the number of samples in overrepresented classes while preserving their diversity. Concurrently, data augmentation techniques were applied selectively only to the training subset of underrepresented classes. These augmentations included a variety of geometric and photometric transformations, such as:

- Geometric Transformations: Random rotations, horizontal and vertical flips, and scaling to increase the model's robustness to variations in object orientation and size.
- Photometric Transformations: Brightness, contrast, and noise adjustments to simulate diverse environmental and lighting conditions.
- This controlled augmentation strategy ensured that the dataset remained fair and prevented the creation

of artificially inflated classes, thereby enhancing the model's ability to accurately detect rare FOD instances.

3.4. Dataset Extension and Validation

The final step involved integrating the custom-collected images, which had been subjected to the same augmentation and stratification procedures, with the processed FOD-A dataset. This extension resulted in a final dataset that was well-balanced and optimized. Summary of the data splits is given in Table 2. The systematic data preparation process, from collection and annotation to balancing and augmentation, laid a strong foundation for the subsequent model training, directly contributing to the high mAP and overall robustness of the final system.

Split	Original	Balanced	Extended
Train	25853	24011	24606
Val	5165	3990	4108
Test	3454	2658	2737
Total	34472	30659	31451

Table 2. Summary of Data splits

4. MACHINE LEARNING MODEL DEVELOPMENT

The YOLO11 pre-trained model was used with transfer learning, leveraging knowledge from large-scale datasets to accelerate training and improve generalization. Key hyperparameters were carefully selected, including a batch size of 16, 16 workers for data loading, and specific settings for mosaic (0.8) and mix-up (0.4) augmentation to enhance robustness. To address the remaining class imbalance, class weights in the data.yaml file was fine-tuned, assigning higher weights to classes with lower detection accuracy to improve recall. This systematic approach optimized the model for efficient and accurate detection of small objects within complex runway backgrounds.

5. MODEL PERFORMANCE EVALUATION

The model's performance was rigorously evaluated using key metrics. In a comparative analysis of Foreign Object Debris (FOD) detection models, YOLOv8 demonstrated a performance gain over YOLOv5, with YOLOv8 achieving a mAP@95 of 88.35% over YOLOv5's mAP@95 of 80.64% (Kumari, Dixit and Agrawal, 2024). Despite these gains, the most effective model was YOLO11. In this paper, YOLO11 model is fine-tuned on the original dataset achieved a mAP@95 score of 88.88%, while the model trained on the enhanced dataset achieved a mAP@95 score of 89.28%. With a low inference time of 94.4ms, making it suitable for real-time applications. Precision-Recall curves and confusion matrices demonstrated the model's high performance across most classes. While the model performed well, minor challenges such as false positives and small debris detection were identified as areas for future improvement

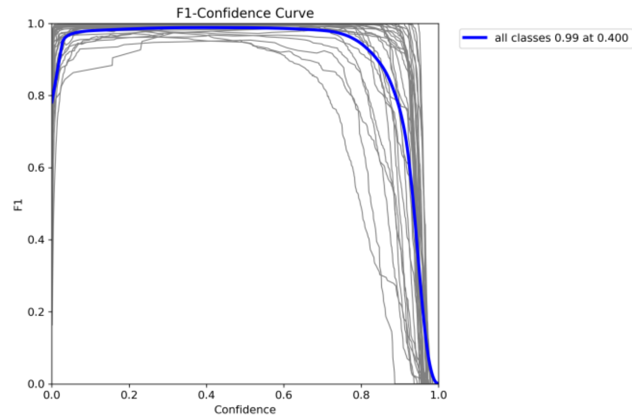

Figure 1. F1-Score vs. Confidence curve

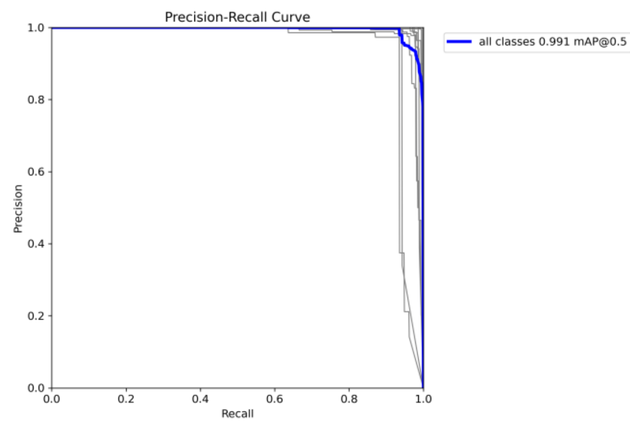

Figure 2: Precision-Recall Curve

6. SYSTEM INTEGRATION

A robust and scalable system architecture is crucial for deploying an AI model in a safety-critical environment. This phase focused on integrating the trained YOLO11 model into a production-oriented system that facilitates real-time FOD detection and supports future extensibility within a PHM framework.

6.1. Backend and Model Deployment

The core of the system's backend is a lightweight Flask REST API, which was chosen for its simplicity and efficiency in handling HTTP requests. After training, the YOLO11 model was exported to the ONNX format. This strategic choice allows for optimized, hardware-agnostic inference, ensuring the model can be deployed seamlessly across various platforms, from cloud servers to resource-constrained edge devices. The API exposes a single endpoint that accepts an image file, performs real-time object detection using the ONNX model, and returns a processed image with bounding boxes and a structured JSON response containing the detected objects and confidence scores.

6.2. Frontend for User Interaction

The user-facing component is a web-based application built with Angular and styled with Bootstrap. The front end serves as a crucial interface for operational personnel. It provides a user-friendly mechanism to upload images and visualize the detection results. The side-by-side display of the original and processed images, coupled with a table of detected objects and their confidence scores, ensures that the system's output is easily interpretable and actionable. This design prioritizes clarity and positive user experience, which is essential for rapid decision-making in aviation safety applications.

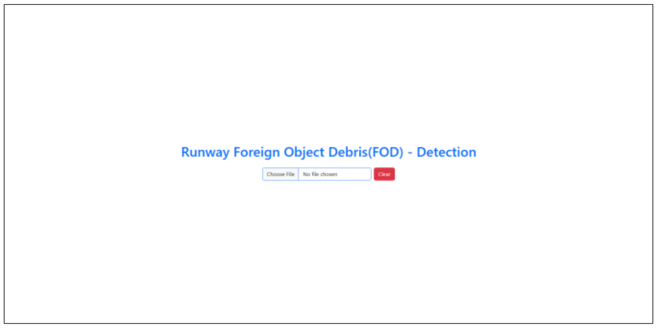

Figure 3: Application Main Page

Figure 4: Showing detected results

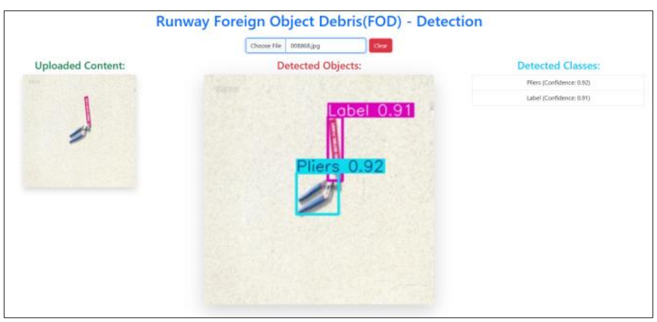

Figure 5: Displaying multiple detection results

Figure 6: Detection of the custom image

6.3. System Architecture

The system's design incorporates Machine Learning Operations (MLOps) best practices to ensure its reliability, reproducibility, and long-term viability, which are fundamental to a PHM solution. Git is used for version control of both the frontend and backend code, facilitating collaborative development. The entire backend application is containerized using Docker, guaranteeing environment consistency and simplifying deployment. This containerized microservice can be easily scaled to handle increased load, such as processing real-time video streams, which is a key recommendation for future work. By embracing these MLOps principles, the system is not merely a proof-of-concept, but a production-ready solution designed for continuous monitoring and a structured deployment pipeline, capable of supporting the iterative improvements required for a comprehensive aviation PHM system.

7. CONCLUSIONS/ RECOMMENDATIONS

7.1. Conclusions

This paper successfully demonstrates the feasibility and effectiveness of an AI-powered system for foreign object debris detection on airport runways using YOLO11. The model, trained on a balanced and augmented dataset, achieves a high mAP@95, proving its potential for real-world aviation safety applications. By providing highly accurate and timely information on FOD, this system acts as a critical component in a comprehensive Prognostics and Health Management (PHM) strategy. The real-time data on debris type, location, and frequency can be used to inform predictive models for maintenance, asset health, and operational risk.

7.2. Recommendations and Future Work

1. Integration of Video and Live CCTV Feed Processing: Extending the current system to handle real-time video feeds from CCTV cameras would enhance its operational utility. Implementing frame-by-frame object detection with efficient tracking algorithms can ensure seamless monitoring.

2. Deployment on Edge Devices and Cloud Services: To reduce latency and computational overhead, the model can be optimized for deployment on edge devices (e.g., NVIDIA Jetson, Intel Movidius) or cloud platforms (e.g., AWS, Azure). This would enable real-time processing without overloading airport surveillance systems.

3. Prognostic Value and PHM Integration: The current detection system provides foundational data for a robust PHM framework. Future work will focus on developing prognostic capabilities, such as predictive maintenance, asset health monitoring, and risk assessment based on historical FOD event data.

4. Automation of Annotation and Dataset Expansion: Incorporating active learning techniques can reduce manual annotation efforts by prioritizing samples that need labelling. Continuously updating the dataset with real-world airport footage can help the model adapt to varied lighting, weather conditions, and debris types.

5. Integration with Airport Safety and Alert Systems: The detection system can be integrated with airport control tower software for automated alerts to ground staff in case of detected hazards.

REFERENCES

S. Jain, M. S. Prasad & R. Gutpa (2022), "A Comparison of Manual and Automotive FOD Detection Systems at Airport Runways," *2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, Noida, India, 2022, pp. 1-5, doi: 10.1109/ICRITO56286.2022.9964557.

A. Kumari, A. Dixit & P. Agrawal (2024), "Deep Learning Based Foreign Object Debris (FOD) Detection on Runway," *2024 International Conference on Emerging Smart Computing and Informatics (ESCI)*, Pune, India, 2024, pp. 1-6, doi: 10.1109/ESCI59607.2024.10497406.

Travis Munyer, Pei-Chi Huang, Chenyu Huang, & Xin Zhong. (2021). FOD-A: A Dataset for Foreign Object Debris in Airports.

A. G V & P. R (2024), "The Modern Approaches for Identifying Foreign Object Debris (FOD) in Aviation," *2024 International Conference on Integrated Circuits and Communication Systems (ICICACS)*, Raichur, India, 2024, pp. 1-5, doi: 10.1109/ICICACS60521.2024.10498909.

D. Song, F. Yuan & C. Ding (2022), "Track Foreign Object Debris Detection based on Improved YOLOv4 Model," *2022 IEEE 6th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, Beijing, China, 2022, pp. 1991-1995, doi: 10.1109/IAEAC54830.2022.9930100.

T. -Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan & S. Belongie (2017) "Feature Pyramid Networks for Object Detection," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 936-944, doi: 10.1109/CVPR.2017.106.

Hilburn, Brian G. & Pesmen, B S (2023) "Foreign Object Debris Detection" https://doi.org/10.21949/1528220

BIOGRAPHIES

Phani Kiran Yellakara received his MTech in Artificial Intelligence and Machine Learning at Birla Institute of Technology and Science, Pilani. He also works as a Principal Engineer at Collins Aerospace. His research interests include computer vision, deep learning, and their applications in aviation safety

Srinivasarao Surapu is a Principal Engineer at Collins Aerospace. His professional work focuses on prognostics and health management (PHM), with expertise in developing AI-powered solutions for complex engineering systems.