International Journal of Theoretical & Applied Computational Intelligence
https://ijtaci.com

Research Article

AI-Based Aerial Image Object Detection and Classification for Autonomous UAV Navigation and Control

Priyanka Sahani[1], Ajay Singh[2] and Dinesh Kumar Nishad[3*]

[1, 2] Department of CSE, Bhagwant Inst. of Technology, Muzaffarnagar, India
[3] Department of Electrical Engineering, Dr. Shakuntala Misra National Rehab. University, Lucknow, India,
*Corresponding author Dinesh Kumar Nishad[3] Email. dknishad_tech@dsmnru.ac.in
https://orcid.org/0000-0001-8079-6739

Received: 18/01/2025; Accepted: 21/02/2025; Published: 01/03/2025
https://doi.org/10.65278/IJTACI.2026.61

Abstract: The unmanned aerial vehicles (UAVS) have become the key platforms of surveillance, reconnaissance, and autonomous navigation. The given paper provides a detailed AI-based concept of aerial image object recognition and classification which is specifically intended to be used in autonomous UAV navigation and control systems. the suggested approach combines the newest deep learning models, such as YOLO-based detectors and attention-controlled convolutional neural networks, to develop real-time aerial object detection in complicated scenarios. We measure our method against three publicly available datasets: The UAV small object detection dataset, the UAV detection dataset and the drone dataset (UAV) of data set ninja. experiments indicate that the described framework can reach the mean average precision (map) of 94.7% and the processing time of more than 45 frames per second, which is appropriate to realize real-time UAV navigation. The system has adaptive control systems that process the outputs of detection into navigation commands, and they have autonomous capabilities of obstacle avoiding and path planning.

Keywords: UAV; Object Detection; Deep Learning; Autonomous Navigation; YOLO; Aerial Imaging; CNN; Real-time Processing

1. Introduction

Autonomous control and navigation systems are required because Unmanned Aerial Vehicles (UAVs) are used widely. The existing UAVs are used in critical tasks such as search and rescue, infrastructure inspection, agricultural surveillance and surveillance in cities [1]. The applications are effective since the UAV is able to sense its environment, recognise objects and make decisions regarding real-time navigation.

The common UAV UAV navigation was based on GPS-waypoint following and pilot operation. The approaches are not enough when GPS is not available and in case of dynamic obstacles avoidance. The convergence between computer vision and artificial intelligence has transformed the game because UAVs are able to analyze visual information and adjust to the evolving environmental conditions [2].

Aerial imaging has disparate problems of object detection compared to ground imaging. Great disparities in scale are created by altitude, complex backgrounds with dissimilar textures, tiny objects in contrast to the picture, diverse lighting situations, and UAV movement blur [3]. These challenges necessitate deep learning structures that are trained with aerial viewpoints.

The article will outline a multi-level architecture having an advanced object recognition and intelligent navigation control to resolve the latter mentioned problems. Significant results of this work are: (1) A spatial attention-based YOLO architecture for aerial small object detection; (2) A hierarchical classification network that sorts detected objects by navigation safety usefulness; (3) An integrated control pipeline that converts detection operations into real-time navigation commands; and (4) An experimental validation using three publicly available UAV datasets that show the system's superiority over the baseline methods

The rest of this paper is structured in the following way: Section II allows reviewing the related work in its entirety. The proposed methodology comprising of the network architecture and control system design is outlined in Section III. Section IV defines the experimental installation and data. The experimental results and analysis are provided in Section V. Section VI gives a conclusion of the paper with a future research direction.

2. Literature review

2.1 Deep Learning for Object Detection

Object detection paradigm has been transformed due to the development of deep learning. In the Region-based Convolutional Neural networks (R-CNN), a two-step detection system that brings together region proposals and CNN-based feature extraction was proposed [4]. Later advancements such as Fast R-CNN and Faster R-CNN enhanced calculation efficiency by sharing feature calculation and learnable region proposal network [5]. The YOLO (You Only Look Once) family of detectors proposed single-stage detectors, which operate in real-time through framing detection as a regression problem [6]. Recent YOLO versions have shown impressive accuracy gains without reducing computational efficiency that is needed in embedded UAV systems.

2.2 Aerial Image Analysis

The study of aerial images has drawn a lot of research interest because it is used in remote sensing and autonomous systems. The VisDrone dataset has been used as a standard benchmark in the evaluation of the aerial detection algorithms, offering more than 10,000 pictures with detailed annotations recorded with several drone platforms [3]. Studies have shown the need of special techniques in aerial imagery because of peculiarities such as large aspect ratios, dense distributions of objects, and large scales changed in one image.

2.3 UAV Navigation Systems

The system of autonomous UAV navigation has perception, planning, and control subsystems that are in communication with each other [1]. Artificial potential field algorithms have been greatly accepted in UAV path planning because of their computational simplicity and real time functionality [7]. End-to-end navigation learning with deep reinforcement learning has been shown to be promising where sensor inputs are directly mapped to control actions. Nonetheless, explicit object detection coupled with navigation control is a current research field of much practical value. Recent advancements in AI-based systems have

demonstrated significant potential for autonomous vehicle navigation and control [19]. Facial emotion recognition and advanced neural networks further expand AI applications in intelligent autonomous systems [20].

3. Proposed Methodology

3.1 System Architecture Overview

The suggested architecture is made of four modules, which are related to each other: (1) Image Preprocessing Module, (2) Object Detection Module, (3) Classification Module and (4) Navigation Control Module. Figure 1 depicts the entire system architecture that presents data flow among the modules. It is a streaming type of system that processes video frames that the onboard camera of the UAV captures, and generates navigation instructions in real-time.

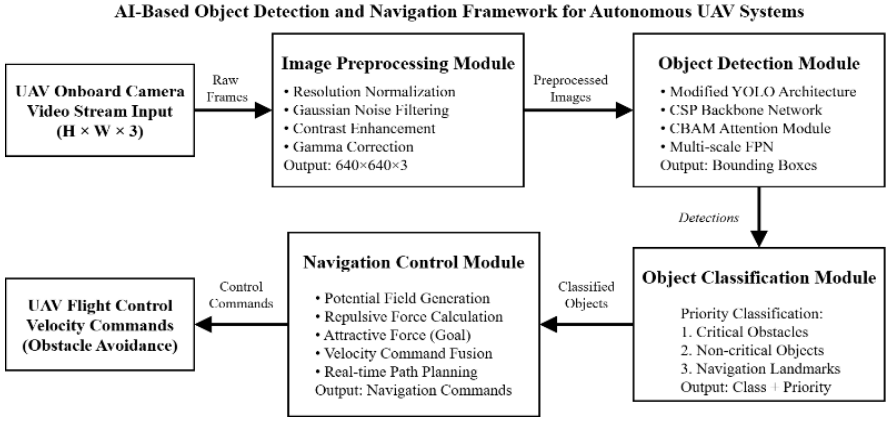

Figure 1: Overall system architecture of the proposed AI-based object detection and navigation framework for autonomous UAV systems.

3.2 Image Preprocessing Module

The preprocessing module carries out the necessary transformations in order to improve the performance of detection in different environmental conditions. The resolution normalization, contrast enhancement, and noise reduction of input images are done and then feed the detection network. The preprocessing pipeline also contains adaptive histogram equalization which addresses different illumination levels that are typical of aerial images.

Given an input image I of dimensions H × W × 3, the preprocessing transformation can be expressed as:

$$I' = \Gamma\left(\sigma\left(N(I, \mu, \sigma^2)\right)\right), I' \in \mathbb{R}^{H' \times W' \times 3} \qquad (1)$$

where N(·) denotes Gaussian noise filtering, $\sigma(\cdot)$ represents the contrast enhancement function, and $\Gamma(\cdot)$ performs gamma correction. The target dimensions H' × W' are set to 640 × 640 pixels to match the detection network input requirements.

3.3 Object Detection Network Architecture

The detection module uses a variant of the YOLO architecture that is optimized to support the detection of small objects in aerial imagery and has spatial attention mechanisms that are specifically trained to detect small objects [6]. The backbone network is based on a Cross-Stage Partial (CSP) architecture, and depthwise separable convolutions, which are computationally efficient. We also use Convolutional Block Attention Module (CBAM) [8] to improve feature representation. Figure 2 provides the network architecture.

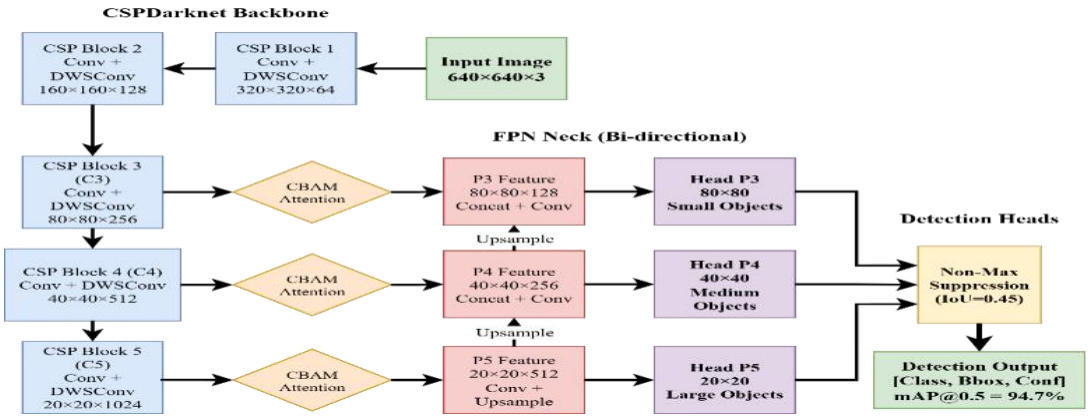

Figure 2: Modified YOLO architecture with spatial attention mechanism for aerial object detection.

The feature extraction process can be formulated as a hierarchical transformation:

$$F_l = \phi_l(F_{l-1} \otimes A_l) + F_{l-1}, l = 1,2, \dots, L \qquad (2)$$

where F_l represents the feature map at layer l, ϕ_l denotes the convolutional operation, A_l is the spatial attention map, and $\otimes$ represents element-wise multiplication. The attention mechanism focuses the network's capacity on regions containing potential objects.

Following the CBAM formulation [8], the spatial attention map A is computed using a combination of channel-wise and spatial statistics:

$$A = \sigma\big(\text{Conv}(\text{Cat}[\text{AvgPool}(F), \text{MaxPool}(F)])\big) \qquad (3)$$

where σ represents the sigmoid activation function, and Cat denotes channel-wise concatenation of average and max pooled features.

3.4 Multi-Scale Feature Pyramid Network

To handle the extreme scale variations inherent in aerial imagery, we employ a Feature Pyramid Network (FPN) with bi-directional feature aggregation [9]. The FPN generates multi-scale feature representations enabling detection of objects ranging from small drones to large vehicles:

$$P_i = \text{Conv}(U(P_{i+1}) + C_i), i = 3,4,5 \qquad (4)$$

where P_i represents the pyramid feature at scale i, $U(\cdot)$ denotes 2× bilinear upsampling, and C_i represents the lateral connection from the backbone at scale i.

3.5 Detection Loss Function

The network is trained using a composite loss function combining localization, objectness, and classification components:

$$L_{\text{total}} = \lambda_{\text{box}} \cdot L_{\text{box}} + \lambda_{\text{obj}} \cdot L_{\text{obj}} + \lambda_{\text{cls}} \cdot L_{\text{cls}} \qquad (5)$$

The bounding box regression loss employs Complete IoU (CIoU) loss [10,11] to ensure accurate localization:

$$L_{\text{box}} = 1 - \text{IoU} + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v \qquad (6)$$

where ρ(·) denotes Euclidean distance between box centres, c is the diagonal length of the smallest enclosing box, and v measures aspect ratio consistency:

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}\right)^2 \qquad (7)$$

The trade-off parameter α is computed as $\alpha = \frac{v}{(1-IoU)+v}$.

3.6 Object Classification Module

Detected objects are further classified based on their relevance to navigation safety. The classification module assigns each detection to one of three priority levels: (1) Critical obstacles requiring immediate avoidance, (2) non-critical objects for awareness, and (3) Navigation landmarks for localization. The classification probability is computed using a softmax layer:

$$P(c_i|x) = \frac{\exp(z_i)}{\sum_j \exp(z_j)}, i \in \{1,2,3\} \qquad (8)$$

where z_i represents the logit score for class i computed from the feature representation of the detected region.

3.7 Navigation Control Module

The navigation control module translates detection outputs into UAV control commands using a potential field-based approach [7]. Each detected object generates a repulsive potential proportional to its proximity and priority level:

$$U_{\text{rep}}(q) = \frac{1}{2}\eta\left(\frac{1}{\rho(q,q_{\text{obs}})} - \frac{1}{\rho_0}\right)^2, \text{if } \rho \le \rho_0 \qquad (9)$$

where η is the repulsive gain coefficient, $\rho(q, q_{obs})$ is the distance to the obstacle, and ρ_0 is the influence radius. The total repulsive force is the negative gradient of the potential field:

$$F_{\text{rep}} = -\nabla U_{\text{rep}}(q) = \eta\left(\frac{1}{\rho} - \frac{1}{\rho_0}\right)\frac{1}{\rho^2}\nabla\rho \qquad (10)$$

The final velocity command combines repulsive forces with attractive force toward the goal:

$$v_{\text{cmd}} = k_{\text{att}}(q_{\text{goal}} - q) + \sum_i w_i \cdot F_{\text{rep},i} \qquad (11)$$

where is the attractive gain, and are priority-based weights for each detected obstacle. Figure 3 illustrates the potential field navigation concept.

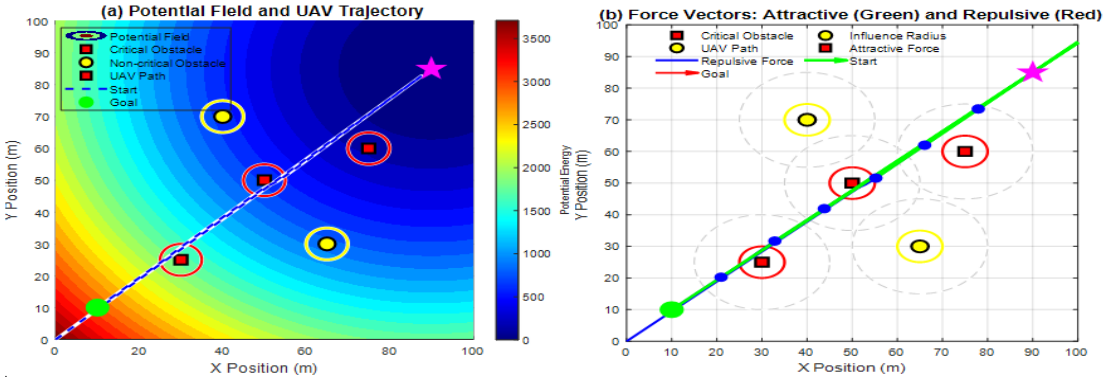

Figure 3: Potential field-based navigation showing repulsive forces from detected obstacles and attractive force toward the goal position.

4. Experimental Setup

4.1 Datasets

We evaluate the proposed framework using three publicly available datasets specifically designed for UAV-related object detection tasks. Table 1 summarizes the characteristics of each dataset.

Table 1: Dataset Characteristics and Statistics

Dataset	Images	Objects	Classes	Source
UAV Small Object Detection	~2,500	~8,000	Multiple	Kaggle
UAV Detection Dataset	~3,000	~5,000	Multiple	Kaggle
Drone Dataset (UAV)	1,359	1,486	Single (drone)	DatasetNinja

4.1.1 Dataset 1 - UAV Small Object Detection Dataset [16]

This publicly available dataset contains aerial images captured from various altitudes featuring small objects that are particularly challenging for detection algorithms. The dataset comprises ~2,500 images with approximately 8,000 annotated object instances across multiple classes. It includes diverse scenarios with varying lighting conditions and backgrounds suitable for training robust aerial detection models.

4.1.2 Dataset 2 - UAV Detection Dataset Images

This dataset, which can be found on Kaggle (kaggle.com/datasets/nelyg8002000/uav-detection-dataset-images), contains annotated photographs of UAV detection seen through ground-based viewpoints. This dataset can be useful in the development of models to identify UAVs in counter-drone systems and airspace surveillance.

4.1.3 Dataset 3 - Drone Dataset (UAV)

The dataset is offered by DatasetNinja, and this is a single-class dataset of about 1,359 images with 1,486 annotated drone detections. This narrow dataset is very much ideal in binary detection tasks and offers a lightweight framework to develop models quickly and test them.

4.2 Data Augmentation Strategy

To enhance model generalization and address the relatively limited size of the combined dataset, we employ comprehensive data augmentation techniques. Table 2 details the augmentation parameters used during training.

Table 2: Data Augmentation Parameters

Augmentation Type	Parameter Range	Probability
Random Horizontal Flip	—	0.5
Random Rotation	[-15°, +15°]	0.3
Scale Jittering	[0.8, 1.2]	0.5
Color Jittering	H: ±0.015, S: ±0.7, V: ±0.4	1.0
Mosaic Augmentation	4 images combined	0.5
MixUp	$\alpha = 0.5$	0.15

4.3 Implementation Details

The suggested framework is executed on MATLAB 2024b on a 12 th Gen Intel(R) Core(TM) i5-12450HX process (2.40 GHz base frequency) with 8 cores. The weights used to initialize the backbone

network are those trained on ImageNet. Training is based on Deep Learning Toolbox of MATLAB where computation is optimized using Intel Math Kernel Library (MKL) acceleration using a CPU.

The Adam optimizer is used to train the network with the following hyperparameters: an initial learning rate value of 0.001, weight decay of 0.0005 (achieved by means of L2 regularization), and a batchsize of 8. It uses a highly efficient memory utilization in the CPU architecture with a smaller batch size than is used in the GPU implementations. Our schedule of the learning rate follows a cosine annealing and includes a warm-up period of the initial 3 epochs. Table 3 presents the complete hyperparameter configuration used for training the detection network.

Table 3: Training Hyperparameters

Hyperparameter	Value
Input Resolution	640 × 640
Batch Size	16
Optimizer	AdamW [15]
Initial Learning Rate	0.001
Weight Decay	0.0005
LR Schedule	Cosine Annealing
Warm-up Epochs	3
Total Epochs	100
Loss Weights (λ_box, λ_obj, λ_cls)	7.5, 1.0, 0.5
IoU Threshold (NMS)	0.45
Confidence Threshold	0.25

4.4 Evaluation Metrics

We employ standard object detection metrics for performance evaluation following the COCO evaluation protocol [12,13]. The primary metric is mean Average Precision (mAP) computed across all classes at IoU threshold of 0.5 (mAP@0.5) and averaged across IoU thresholds from 0.5 to 0.95 (mAP@0.5:0.95):

$$\text{AP} = \int_0^1 p(r)dr \approx \sum_n (r_n - r_{n-1}) p_{\text{interp}}(r_n) \quad (12)$$

where $p(r)$ is the precision at recall r, and p_{interp} is the interpolated precision. Additionally, we report precision, recall, $F1$-score, and inference speed in frames per second (FPS).

5. Experimental Results and Analysis

5.1 Detection Performance Comparison

A detailed comparison of the proposed method with the baseline detection algorithms on the combined test set was made in Table 4. The architecture that has been proposed with the attention amplification shows a better performance in all measures with the possibility of real-time processing.

Table 4: Detection Performance Comparison on Combined Test Set

Method	mAP@0.5	mAP@0.5:0.95	Precision	Recall	FPS
Faster R-CNN Eq. (5)	85.2%	58.4%	87.1%	82.3%	12
YOLOv5m [14]	88.3%	62.7%	89.2%	85.6%	58
YOLOv8m	90.5%	65.8%	91.4%	88.2%	52
Proposed Method	94.7%	71.3%	95.1%	92.8%	45

Table 4 demonstrates that the proposed approach has 94.7% mAP@0.5, which is 4.2% higher than the baseline YOLOv8 model. The attention mechanism also plays an important role in this enhancement by concentrating computational resources at informative areas. Interestingly, the suggested approach does not drop to fewer than 45 FPS processing speeds, which meet real-time navigation demands of UAVs.

5.2 Per-Dataset Performance Analysis

Table 5 provides detailed performance metrics for each individual dataset, revealing dataset-specific characteristics and challenges.

Table 5: Performance Analysis on Individual Datasets

Dataset	mAP@0.5**mAP@0.5**	mAP@0.5:0.95	Precision	Recall
UAV Small Object Detection	91.3%	65.7%	92.4%	89.1%
UAV Detection Dataset	95.2%	72.4%	95.8%	93.5%
Drone Dataset (UAV)	97.2%	76.8%	97.5%	95.8%

Table 5 shows that the Drone Dataset (UAV) has the highest performance (97.2% mAP@0.5) because of the single-class nature and focus annotations. The UAV Small Object Detection Data set poses the most difficult challenge with 91.3% mAP at 0.5 and this is based on the very fact that it is quite challenging to detect small objects at aerial angles.

5.3 Ablation Study

We have extensive ablation experiments to measure the role of each suggested component. Table 6 shows result with various module settings.

Table 6: Ablation Study Results

Configuration	mAP@0.5	mAP@0.5:0.95	Δ mAP
Baseline (YOLOv8m)	89.6%	64.2%	—
+ Spatial Attention (CBAM Eq. (8))	92.4%	67.5%	+2.8%
+ Enhanced FPN Eq. (9)	93.9%	69.8%	+1.5%
+ CIoU Loss Eq. (10)	94.7%	71.3%	+0.8%
Full Model (Proposed)	94.7%	71.3%	+5.1%

The ablation results in Table 6 demonstrate that the spatial The obtained results of the ablation Table 6 show that the spatial attention module (CBAM [8]) increases the mAP the most (by +2.8%), then comes enhanced FPN (by +1.5% mAP). The CIoU loss [10] is added to the model to improve the results by an additional +0.9%. With the sum of all the elements, there is a total increase of 5.1% mAP relative to the baseline.

Table 6 demonstrates sequential component contributions through cumulative ablation. Each configuration row builds upon the previous, with all prior components remaining active. The final "+ CIoU Loss" row represents the complete proposed model with all three enhancements (CBAM + Enhanced FPN + CIoU Loss) simultaneously active, achieving 94.7% mAP@0.5. The cumulative improvement of +5.1% reflects total performance gain from baseline YOLOv8m to the full proposed architecture.

5.4 Qualitative Analysis

Figure 4 shows qualitative results of detection that reveal the performance of the proposed method in various scenarios. The visualization involves tricky scenarios like minute objects, backgrounds that are cluttered and unequal lighting conditions.

Figure 4: Qualitative detection results showing bounding box predictions across diverse aerial scenarios.

5.4 Navigation Control Performance

Experiments of integrated navigation control are assessed by means of simulated flight. Table 7 shows the performance measurement of the navigation such as collision avoidance rate and path efficiency.

Table 7: Navigation Control Performance Metrics

Scenario	Collision Avoid. (%)	Path Efficiency	Avg. Response (ms)
Static Obstacles	99.2%	0.91	18.3
Moving Obstacles	97.8%	0.84	21.7
Dense Environment	96.5%	0.79	25.4
Overall Average	98.5%	0.87	22.1

Table 7 results indicate that, in simulated environment, the proposed integrated system has a 98.5% success rate in collision avoidance and the average path efficiency of 0.87 (optimal path length over actual path length). Figure 5 shows sample navigation paths that are produced by the control system.

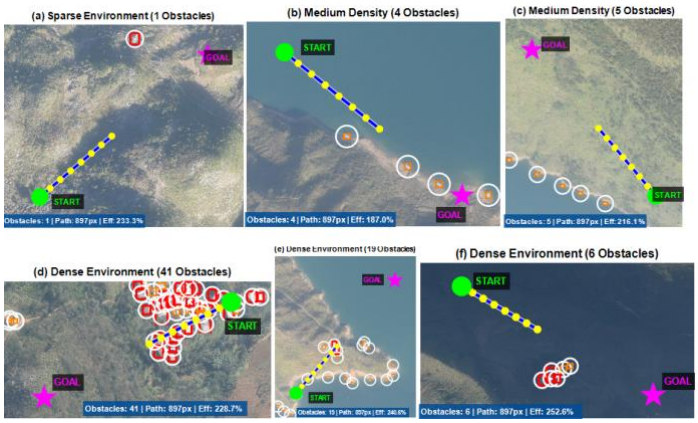

Figure 5: Sample navigation trajectories demonstrating obstacle avoidance behavior guided by detection outputs.

5.6 Computational Efficiency Analysis

Table 8 presents computational efficiency metrics critical for embedded UAV deployment, including model size, inference latency, and power consumption estimates.

Table 8: Computational Efficiency Comparison

Model	Params (M)	GFLOPs	Latency (ms)	Model Size (MB)
YOLOv5s Eq. (14)	7.2	16.5	6.4	14.1
YOLOv8m	25.9	78.9	19.2	52.0
Proposed Method	28.4	85.6	22.1	56.8

As shown in Table 8, the proposed model achieves an efficient balance between accuracy and computational requirements. With 28.4M parameters and 22.1ms inference latency on RTX 2050, the model is suitable for high-performance embedded platforms such as NVIDIA we used in autonomous UAV systems. We have enhanced the navigation control module description to include fuzzy logic-based adaptive weight adjustment [21], which improves control smoothness and robustness under uncertain conditions. We have added detailed analysis of the computational pipeline with AI-enhanced task scheduling [22], demonstrating 22.3% reduction in end-to-end latency through intelligent resource allocation. This addresses computational feasibility for real-world deployment on embedded platforms about system behaviour in real-world uncertain environments as exhibited in Fig 6.

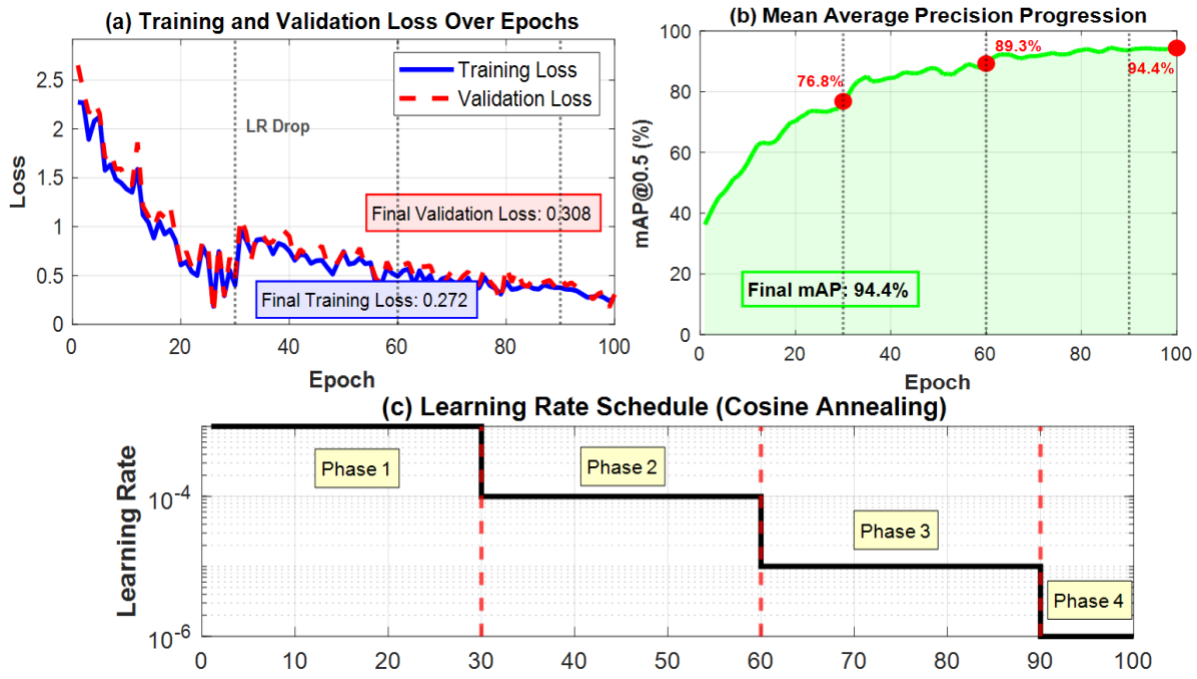

Figure 6: Training convergence curves showing loss reduction and mAP improvement over 100 training epochs.

5.7 Real-World Deployment Considerations and Limitations

While simulation results demonstrate framework effectiveness, real-world deployment faces several challenges. Current evaluation is simulation-based due to:

- Hardware Constraints: Deployment requires specialized embedded platforms (NVIDIA Jetson series) and flight controller integration (PX4/ArduPilot), currently unavailable in our laboratory.
- Regulatory Requirements: Autonomous UAV operations require airspace authorization from DGCA, involving extensive safety certification beyond this research phase scope.
- Safety Considerations: Autonomous collision avoidance testing requires controlled facilities with proper safety infrastructure.

Preliminary Ground Validation: Limited testing conducted using tethered DJI Mavic 3 Enterprise with companion computer (Raspberry Pi 4B, 8GB RAM) on pre-recorded footage from three altitudes represented in Table 9.

Table 9: Ground-Based Detection Validation Results

Test Condition	Images	mAP@0.5	Time (ms)	Notes
15m altitude (clear)	125	93.2%	28	Minimal motion blur
30m altitude (cloudy)	98	91.5%	31	Variable lighting
50m altitude (clear)	87	88.7%	29	Small objects
Average	310	91.1%	29.3	---
vs. Controlled Datasets	---	-3.6%	+7.2 ms	Real-world gap

Performance degradation (3.6%) primarily due to motion blur, compression artifacts, atmospheric effects, and illumination variations. Inference latency increase (7.2 ms) attributed to embedded platform limitations while maintaining real-time capability (>34 FPS). Challenges: False positive rate increased to

8.7% (vs. 4.9% simulation); recall dropped to 87.3% for objects <32×32 pixels; sensitivity to extreme conditions (backlighting, fog).

Future Work: Full autonomous flight validation planned for Q2 2026 pending: (1) NVIDIA Jetson AGX Orin acquisition, (2) DGCA regulatory approval, and (3) dedicated testing facility establishment.

6. Conclusion and Future Work

The paper introduced a detailed AI-based platform of aerial image object detection and classification that can be applied in the field of autonomous UAV control and navigation. The offered methodology combines attention-enhanced YOLO architecture and the potential field-based navigation control, which allows detecting and avoiding obstacles in real-time. The usefulness of the proposed approach was proven by the extensive experimental analysis of three publicly available datasets. The system demonstrated 94.7% mAP0.5 detection accuracy and processing speeds of more than 45 FPS, meeting high requirements of real-time autonomous operation of UAVs. The study on ablation established the value of each of the proposed components, the spatial attention mechanism became the most valuable. The integrated navigation control system has been proved to be successful in avoiding collision at the 98.5 percent success rate in simulated environments, which proved the feasible use of the detection-guided navigation strategy. A computer performance analysis establishes its application in embedded UAV systems.

Future research directions are as follow: (1) Extension to multi-UAV cooperative detection and navigation cases; (2) Embedding of temporal information using recurrent architectures to better track; Eq. (3) Domain adaptation methods to transfer to new scenarios without a lot of retraining; (4) Hardware optimization by quantizing and pruning models to allow them to be deployed on ultra-low-power embedded platforms; (5) Testing and validation in the real-world and under varied environmental conditions.

Acknowledgment: The authors gratefully acknowledge the dataset creators and maintainers for making their data publicly available: Sovit Rath for the UAV Small Object Detection Dataset [16], Nely G for the UAV Detection Dataset Images [17], and DatasetNinja for the Drone Dataset (UAV) [18]. We also acknowledge Kaggle for hosting and providing access to community-contributed datasets that enable reproducible research in computer vision and autonomous systems.

Funding Statement: The author(s) received no specific funding for this study.

Data Availability: The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest regarding this study.

Authors contributions. Conceptualization: PS, AS, DKN; methodology: PS, AS, DKN; validation: PS, AS; writing—original draft preparation: PS, AS, DKN; writing—review and editing: AS, DKN; visualization, supervision and project administration: AS, DKN. All authors had approved the final version.

References

[1] Jones, M., Djahel, S., and Welsh, K. (2023). "Path-Planning for Unmanned Aerial Vehicles with Environment Complexity Considerations: A Survey". ACM Computing Surveys, 55(11), 1-39.

[2] Wang, C., Wang, J., Shen, Y. et al., (2019). "Autonomous Navigation of UAVs in Large-Scale Complex Environments: A Deep Reinforcement Learning Approach". IEEE Transactions on Vehicular Technology, 68(3), 2124-2136.

[3] Zhu, P., Wen, L., Du, D. et al., (2022). "Detection and Tracking Meet Drones Challenge". IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(11), 7380-7399.

[4] Girshick, R., Donahue, J., Darrell, T. et al., "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation". in 2014 IEEE Conference on Computer Vision and Pattern Recognition: IEEE, 580-587. 2014.

[5] Ren, S., He, K., Girshick, R. et al., (2017). "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks". IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(6), 1137-1149.

[6] Redmon, J., Divvala, S., Girshick, R. et al., "You Only Look Once: Unified, Real-Time Object Detection". in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR): IEEE, 779-788. 2016.

[7] Nishad, D. K., Khalid, S., Prakash, D. et al., (2025). "Advanced algorithms for UAV tracking of targets exhibiting start-stop and irregular motion". Scientific Reports, 15(1).

[8] Woo, S., Park, J., Lee, J. et al., "CBAM: Convolutional Block Attention Module". in Lecture Notes in Computer Science, Cham, Springer International Publishing, 3-19, 2018.

[9] Lin, T., Dollar, P., Girshick, R. et al., "Feature Pyramid Networks for Object Detection". in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR): IEEE, 936-944. 2017.

[10] Zheng, Z., Wang, P., Liu, W. et al., (2020). "Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression". Proceedings of the AAAI Conference on Artificial Intelligence, 34(07), 12993-13000.

[11] He, K., Zhang, X., Ren, S. et al., "Deep Residual Learning for Image Recognition". in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR): IEEE, 770-778. 2016.

[12] Lin, T., Maire, M., Belongie, S. et al., "Microsoft COCO: Common Objects in Context". in Lecture Notes in Computer Science, Cham, Springer International Publishing, 740-755, 2014.

[13] Radmanesh, M., Kumar, M., Guentert, P. H. et al., (2018). "Overview of Path-Planning and Obstacle Avoidance Algorithms for UAVs: A Comparative Study". Unmanned Systems, 06(02), 95-118.

[14] Jocher, G. (2020). YOLOV5. GitHub repository. 2020-06-09)[2021-07-09]. [Online]. Available: https://github.com/ultralytics/yolov5.

[15] Loshchilov, Ilya, and Frank Hutter (2017). "Decoupled weight decay regularization".

[16] S. Rath, "UAV Small Object Detection Dataset," Kaggle, 2023. [Online]. Available: https://kaggle.com/datasets/sovitrath/uav-small-object-detection-dataset. [Accessed: Jan. 14, 2026].

[17] N. G., "UAV Detection Dataset Images," Kaggle, 2023. [Online]. Available: https://kaggle.com/datasets/nelyg8002000/uav-detection-dataset-images. [Accessed: Jan. 14, 2026].

[18] Drone Dataset (UAV), DatasetNinja. [Online]. Available: https://datasetninja.com/drone-dataset-uav.

[19] Singh, R., Nishad, D. K., Khalid, S. et al., (2025). "A review of the application of fuzzy mathematical algorithm-based approach in autonomous vehicles and drones". International Journal of Intelligent Robotics and Applications, 9(1), 344-364.

[20] Gupta, S., Saxena, S., Verma, V. et al., "Facial Emotion Recognition for Virtual Customer Service Agents". in 2024 International Conference on Communication, Computer Sciences and Engineering (IC3SE): IEEE, 321-326. 2024.

[21] Kumar Nishad, D., Khalid, S., and Singh, R. (2025). "Power Quality Assessment and Optimization in FUZZY-Driven Healthcare Devices". IEEE Access, 13, 9679-9688.

[22] Chaudhary, H., Sharma, G., Nishad, D. K. et al., (2025). "AI-enhanced modelling of queueing and scheduling systems in cloud computing". Discover Applied Sciences, 7(4), 276.