



Received: 30/07/2026

Revised: 03/09/2026

Accepted: 28/09/2026

Published online: 30/09/2026

Original Research Article



Open Access under the CC BY -NC-ND 4.0 license

UDC: 534.874.1; 629.07;654.9

AUTONOMOUS SEARCH AND RESCUE DRONE WITH REAL-TIME YOLO OBJECT DETECTION AND GPS NAVIGATION USING RASPBERRY PI 4

Turmaganbet U.K.¹, Turlykozhasheva D.A.¹, Akhtanov S.N.¹, Temesheva S.A.^{1*},
Ussipov N.M.¹, Tao M.²

¹ al-Farabi Kazakh National University, Almaty, Kazakhstan

² School of Electronics and Information, Northwestern Polytechnical University, Xian, China

*Corresponding author: temesheva_symbat4@live.kaznu.kz

Abstract. The development of intelligent unmanned aerial vehicles with onboard computer vision capabilities is increasingly relevant to search and rescue operations. However, real-time human detection on compact UAV platforms remains challenging due to limited computational resources, varying environmental conditions, and the small size of distant targets. This study presents a UAV-based human detection and localization system integrating a Raspberry Pi 4 Model B, YOLOv8, and u-blox NEO-M8N GPS. The system performs onboard processing of the video stream and associates detected targets with the UAV's geographical position. The YOLOv8 model was trained using the VisDrone2019-DET dataset and evaluated through real-world flight experiments under different distances, altitudes, and lighting conditions. The system achieved an average detection confidence of 0.67, with a maximum confidence of 0.81, and a real-time processing speed of 4–8 FPS. Human targets were detected at distances of up to 30 m, while GPS positioning errors ranged from 1.6 to 2.8 m, with a mean error of approximately 2.12 m. The results demonstrate the capability of the proposed UAV platform to perform onboard human detection and GPS-based localization in real-time, providing a basis for UAV-assisted search and rescue applications.

Keywords: Unmanned aerial vehicles, search and rescue, object detection, computer vision, autonomous navigation, embedded systems, real-time processing.

1. Introduction

Unmanned aerial vehicles (UAVs) have become an important technology for search and rescue (SAR) operations, providing rapid aerial surveillance, efficient coverage of large areas, and direct access to hazardous or inaccessible locations. Compared with traditional ground searches, UAVs can reduce response time and improve situational awareness by transmitting live video and collecting geographic information during flight. These capabilities are especially valuable in forests, mountainous terrain, and disaster-affected areas [1-3].

Recent advances in artificial intelligence have significantly improved the autonomy of UAV systems. Computer vision algorithms, particularly the YOLO family of object detectors, enable drones to identify people in aerial images with high processing speed, making them suitable for real-time applications on embedded platforms. Combined with GPS navigation, onboard detection allows each identified target to be associated with geographic coordinates, providing rescue teams with accurate location information [4,5].

Despite these advances, practical deployment of UAV-based SAR systems remains challenging because human targets often appear as small objects against complex aerial backgrounds, while varying illumination, shadows, and limited onboard computational resources can affect detection performance. Previous studies have addressed lightweight object detection and autonomous navigation; however, practical UAV-based SAR

systems still face challenges in combining real-time onboard inference with GPS-based localization and reliable field operation on resource-constrained embedded platforms [6-11].

This study presents the design and experimental validation of an autonomous FPV UAV based on Raspberry Pi 4 that integrates lightweight YOLO-based human detection with GPS navigation. The proposed system performs onboard real-time video processing, associates detected human targets with GPS coordinates, and records detection data to support search and rescue operations.

The main contributions of this work are:

- 1) Development and engineering integration of a compact, Raspberry Pi 4-based UAV platform for onboard human target detection;
- 2) Implementation of real-time YOLOv8 detection synchronized with GPS coordinates using standard, unmodified network architecture;
- 3) Experimental validation of the complete hardware–software system under real flight conditions, evaluating detection accuracy, processing speed, and GPS positioning error.

The article is organized into five sections. Section II presents related work on UAV-based search-and-rescue systems and deep learning-based object detection. Section III describes the materials and methods, including the proposed UAV system architecture, hardware and software configuration, YOLOv8 training, and experimental methodology. Section IV presents the experimental results, including YOLOv8 detection performance, real-time inference results, and GPS positioning accuracy. Section V concludes the study by summarizing the main findings and outlining directions for future work.

2. Related Works

Recent studies have investigated deep learning-based methods for human and object detection in UAV-assisted search and rescue applications. In [3], the authors developed a drone-based autonomous human identification system for real-time search and rescue missions. The proposed approach combines human detection with UAV-based localization and autonomous operation. In [2], the authors investigated autonomous human detection using a lightweight deep learning model deployed on embedded UAV hardware. In [8], the authors presented a state-of-the-art review of human detection from UAV imagery for search and rescue missions, covering existing detection methods, datasets, and evaluation approaches.

YOLO-based architectures have been widely investigated for real-time detection in aerial imagery. In [1], the authors studied YOLO-based detection for real-time search and rescue and evaluated different architectural modifications, including FPN, BiFPN, PB-FPN, an additional P2 detection head, attention modules, and deconvolution. The experiments were conducted using VisDrone for initial training and the Heridal dataset for SAR-specific fine-tuning. The best-performing configuration achieved a mAP@50 of 0.802 while maintaining real-time inference on Jetson Nano hardware.

In [6], the authors proposed an enhanced YOLO11 model for maritime search and rescue object detection. The architecture incorporates a Space-to-Depth module in the backbone, content-aware upsampling in the neck, and an additional detection head for shallow features. The proposed modifications were designed to improve the detection of small, overlapping, and rarely occurring objects. Experiments on the SeaDronesSee dataset showed improved performance compared with YOLOv8, YOLO11, and MambaYOLO, including under lightweight model configurations. In [7], the authors investigated small-object detection for maritime search and rescue using a virtual testbed developed in Unreal Engine. The study focused on customizing YOLO models and using a simulated environment for training and evaluation. The virtual testbed was used to reproduce different maritime SAR conditions and to investigate the performance of object detection models under controlled scenarios. Model architecture and computational efficiency have also been investigated in UAV-based SAR applications. In [9], the authors evaluated different backbone architectures for YOLOv10 across several model variants, including Nano, Small, Medium, Large, and X-Large configurations. The study compared EfficientNet, GhostNet, MobileNet, ResNet50, and Swin backbones in terms of detection accuracy, inference speed, and computational requirements. The YOLOv10-Nano configuration with a MobileNet backbone achieved 80.5 FPS with a mAP@0.5:0.95 of 0.3945, while larger configurations provided higher detection accuracy at increased computational cost.

Thermal infrared imagery has also been investigated with YOLO-based object detection. In [12], the authors evaluated YOLO models for object detection in thermal infrared images. The study examined the performance of different YOLO architectures on thermal data and investigated their applicability to object detection under conditions where thermal information can provide an alternative to visible-spectrum imagery.

Autonomous UAV navigation has been considered together with computer vision-based detection. In [13], the authors investigated autonomous UAV navigation for search and rescue missions using computer vision and convolutional neural networks. The system combines visual perception with UAV navigation and object tracking. At the multi-UAV level, in [4], the authors proposed an open collaborative platform for multiple drones supporting search and rescue operations. The platform addresses coordination and information exchange between UAVs during SAR missions. In [5], the authors developed an autonomous search and rescue drone integrating onboard computing, flight control, and deep learning-based object detection.

In [10], the authors reviewed smart drone technologies for security surveillance and search and rescue applications, including sensing, artificial intelligence, autonomous operation, and communication technologies. In [11], the authors reviewed metaheuristic algorithms for node placement in UAV communication networks, covering optimization approaches for UAV-assisted communication systems. However, existing approaches still have limitations related to small-object detection, computational complexity, and real-time deployment. Architectural modifications can improve detection accuracy but may increase model complexity, while lightweight models can reduce computational requirements at the cost of detection performance.

3. Materials and Methods

3.1 System Architecture

The proposed UAV system consists of two main components: the flight platform and the ground equipment. The flight platform integrates the propulsion system, flight controller, GPS/compass module, onboard computing unit, camera, battery, and communication receiver. The ground equipment provides remote control and monitoring of the UAV during operation. The flight controller is responsible for real-time flight stabilization, sensor processing, and motor control, while the Raspberry Pi 4 Model B performs onboard image processing and YOLO-based object detection. The camera provides a real-time video stream to the Raspberry Pi, where individual frames are processed to identify objects of interest. The u-blox NEO-M8N GPS module provides the current geographical position of the UAV, which is associated with the detected objects [14]. The separation of flight-control and computational functions allows the flight controller to maintain stable UAV operation independently of the computational load generated by the object detection process. The overall electrical and data connections between the main components of the proposed platform are illustrated in Fig.1.

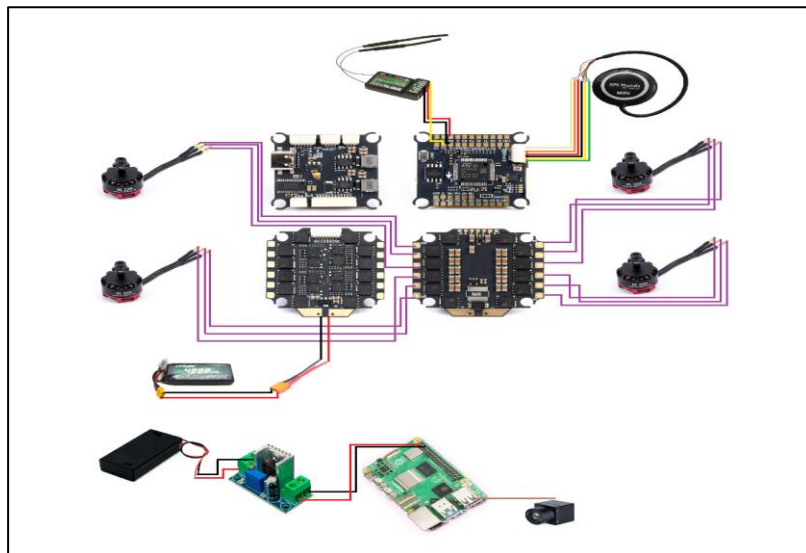


Fig.1. Architecture of the Autonomous Drone System.

Figure 2 presents the functional architecture of the proposed UAV system. The architecture is divided into two main subsystems: flight control and AI-based processing. The flight-control subsystem consists of the flight controller, electronic speed controllers (ESCs), and motors, which provide UAV stabilization and propulsion.

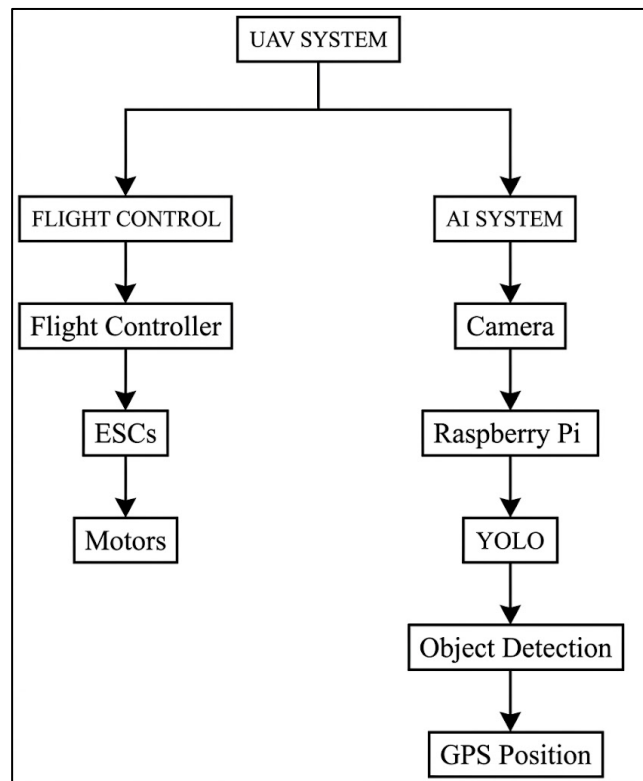


Fig.2. Functional architecture of the proposed UAV-based object detection system.

The AI subsystem includes the camera, Raspberry Pi onboard computer, and YOLO detection model. The camera provides image data to the Raspberry Pi, where YOLO performs real-time object detection. The detected objects are subsequently associated with the UAV's GPS position to obtain their geographical coordinates.

3.2 Hardware Configuration

The UAV platform was assembled using commercial components for flight control, propulsion, navigation, communication, and onboard computing. The main hardware components and their functions are summarized in Table 1. The F405 flight controller with an integrated 65A 4-in-1 ESC serves as the main flight-control unit. It is based on the STM32F405 ARM Cortex-M4 microcontroller and ICM42688P IMU, providing real-time sensor processing, flight stabilization, and motor control. The integrated ESC controls four EMAX RS2205 2300KV brushless motors equipped with HQ5045 BN propellers [15]. Navigation is provided by the u-blox NEO-M8N GPS module with an integrated compass.

Table 1. Hardware configuration of the proposed UAV platform.

<i>Component</i>	<i>Model / Specification</i>	<i>Function</i>
Flight controller	F405 + STM32F405 + ICM42688P	Flight stabilization and sensor processing
ESC and propulsion	65A 4-in-1 ESC, EMAX RS2205 2300KV ×4, HQ5045 BN	Motor control and propulsion
GPS/Compass	u-blox NEO-M8N	Positioning and orientation
Onboard computing	Raspberry Pi 4 Model B	Image processing and YOLO inference
Power system	4S LiPo, 4200 mAh	UAV and onboard power supply

The module supports multiple GNSS constellations and communicates with the flight controller through UART. The FlySky FS-iA6B receiver provides wireless remote-control communication with the flight controller. The Raspberry Pi 4 Model B is used as the onboard computing platform for image processing and

YOLO-based object detection. The flight controller and Raspberry Pi operate as separate processing units, allowing flight stabilization and AI-based image processing to be performed independently. The UAV is powered by a LiPower 4200 mAh 4S LiPo battery. A DC-DC step-down converter provides a regulated 5 V supply for the Raspberry Pi to ensure stable onboard computing during operation.

3.3 Software Pipeline and YOLO Training Configuration

The YOLOv8 model was trained on the VisDrone2019-DET [16] dataset using an NVIDIA GeForce RTX 4090 GPU. The training pipeline was configured for 100 epochs with a batch size of 16 and an image resolution of 1280×720 pixels. Standard data augmentation techniques provided by the Ultralytics [17] framework were applied during training to enhance model robustness under varying illumination and aerial perspectives, including mosaic augmentation, random horizontal flipping, HSV color space jittering (hue, saturation, value), and random scaling. Following training, the optimized weights were deployed directly onto the Raspberry Pi 4 Model B onboard computer for real-time inference using the CPU runtime environment.

For each detected object, the system records the detection information together with the corresponding UAV positional data obtained from the GPS module. The resulting data include the detected class, confidence score, bounding-box coordinates, geographical coordinates, and timestamp. This pipeline enables real-time object detection and provides the information required for subsequent evaluation of the UAV-based detection and localization system.

3.4 Experimental Methodology

The proposed UAV-based object detection system was evaluated through real-world flight experiments using a human target. A total of 5 flight tests were conducted at an approximate flight altitude of 5 m, with the UAV-to-target distance reaching up to 30 m during the experiments. Each test flight evaluated the system localization accuracy over one of five pre-designated ground reference points. Onboard positional telemetry was recorded using the u-blox NEO-M8N GPS module. Reliable positional accuracy was maintained during flights with a 3D fix status, 11 active satellites, and a Position Dilution of Precision (PDOP) of 1.87. Ground truth coordinates for the reference points were measured prior to the flights using a high-precision handheld GNSS receiver with differential correction as a baseline for comparison. The detection capability of the system was evaluated under different environmental and operational conditions, including clear weather, cloudy conditions, direct sunlight, shadowed areas, moving targets, and different flight conditions. During each flight, the camera provided a 1280×720 video stream to the Raspberry Pi 4 Model B, where the YOLOv8 model performed onboard real-time inference.

The detection performance was evaluated using precision, recall, F1-score, mAP@0.5, and mAP@0.5:0.95. Precision and recall were calculated as:

$$precision = \frac{TP}{TP+FP}, \quad (1)$$

$$recall = \frac{TP}{TP+FN}, \quad (2)$$

where TP denotes true positive detections, FP denotes false positive detections, and FN denotes false negative detections.

The mAP@0.5 metric was used to evaluate the mean Average Precision at an IoU threshold of 0.5, while mAP@0.5:0.95 represents the mean Average Precision calculated over IoU thresholds from 0.5 to 0.95 with a step of 0.05. The real-time performance of the system was evaluated using the frames per second (FPS) metric:

$$FPS = \frac{N}{T} \quad (3)$$

where N is the number of processed frames and T is the processing time in seconds.

GPS performance was evaluated using the coordinates obtained from the u-blox NEO-M8N module during flight. The localization error was determined by comparing the recorded position with the corresponding reference position.

4. Results

The experimental results demonstrate the performance of the proposed Raspberry Pi-based YOLO detection system under real-world UAV operating conditions. The trained YOLOv8 model was evaluated using the VisDrone2019-DET dataset, which contains aerial images from 10 object classes, including pedestrian and people. A Precision–Recall (PR) curve was used to evaluate the detection performance of the trained model, as shown in Figure 3.

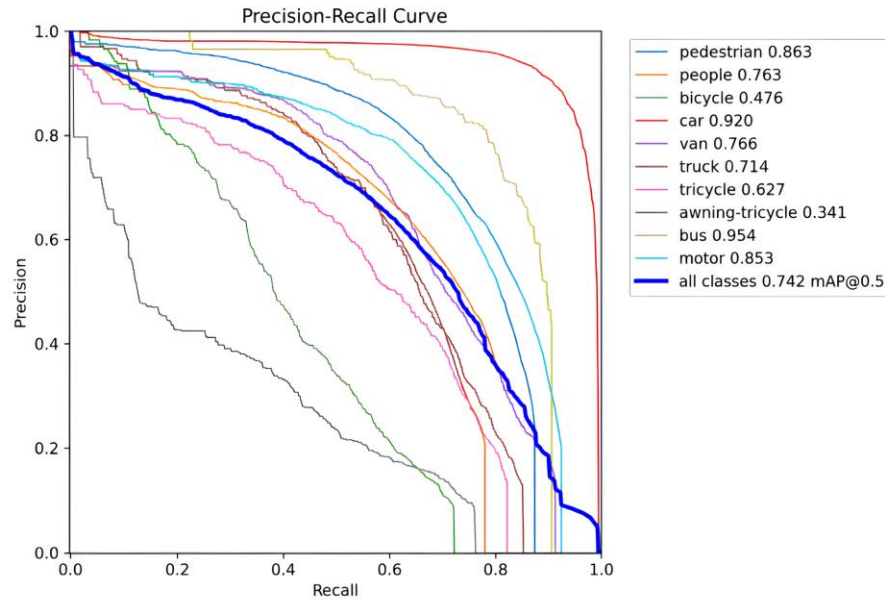


Fig.3. Precision and Recall Curve.

The system achieved an average detection confidence of 0.67, with a maximum confidence of 0.81. The real-time processing speed ranged from 4 to 8 FPS, while human targets were detected at distances of up to 30 m. The GPS-based localization error was approximately 2 m. The class-wise evaluation showed different detection performance across the object categories. Table 2 presents detection performance of YOLOv8 model.

Table 2. Detection performance of YOLOv8 on VisDrone2019-DET dataset.

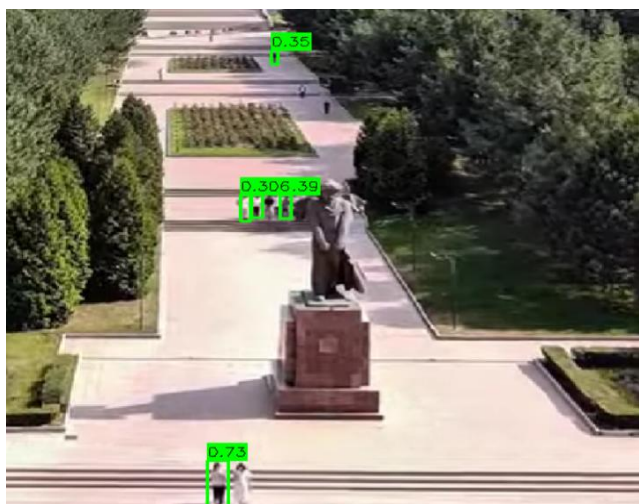
Class	AP@0.5
Bus	0.954
Car	0.920
Pedestrian	0.863
Motor	0.853
Van	0.766
People	0.763
Truck	0.714
Tricycle	0.627
Bicycle	0.476
Awning-tricycle	0.341
All classes (Overall mAP@0.5)	0.742

The highest AP was obtained for bus (0.954), followed by pedestrian (0.863), motor (0.853), and people (0.763). The lowest AP was observed for awning-tricycle (0.341). The overall mAP@0.5 was 0.742. The real-world performance of the complete hardware and software architecture was evaluated during field flight operations. The onboard metrics, including processing speed, detection confidence levels, operating range, and positioning accuracy, are detailed in Table 3. The real-time detection results obtained from UAV camera frames are presented in Figure 4, demonstrating the detection of human targets during flight.

Table 3. Performance results.

<i>Experimental Parameter</i>	<i>Measured Value</i>
Target Detection Class	Human (Pedestrian / Person)
Onboard Hardware Platform	Raspberry Pi 4 Model B
Video Stream Resolution	1280 × 720 px
Onboard Processing Speed	4–8 FPS
Average Detection Confidence	0.67
Maximum Detection Confidence	0.81
Maximum Detection Distance	Up to 30 m
Mean GPS Localization Error	~2.12 m

As shown in Figure 4, the YOLOv8 model successfully detected human targets in the UAV video frames by generating bounding boxes and confidence scores. The confidence values varied depending on the target distance and apparent object size, with higher confidence observed for closer targets and lower confidence for more distant individuals.

**Fig. 4.** Real-time human detection using the proposed UAV system.

As shown in Figure 4, the YOLOv8 model successfully detected human targets in the UAV video frames by generating bounding boxes and confidence scores. The confidence values varied depending on the target distance and apparent object size, with higher confidence observed for closer targets and lower confidence for more distant individuals. The UAV demonstrated stable flight and reliable GPS positioning during the experiments. The GPS data obtained during flight were successfully associated with the detected targets for subsequent localization. The UAV position obtained from the satellite navigation system during the experiment is presented in Figure 5.

**Fig. 5.** GPS position of the UAV during the experimental flight.

The GPS measurements showed positioning errors ranging from 1.6 to 2.8 m across the five reference points. The mean positioning error across all measurement points was 2.12 m, with a standard deviation (SD) of 0.44 m and a Root Mean Square Error (RMSE) of 2.16 m. The lowest error of 1.6 m was obtained at Point 3, while the highest error of 2.8 m was observed at Point 2. These statistical metrics demonstrate consistently positioning repeatability of the onboard navigation system under real flight conditions. The GPS positioning results obtained at the five reference points are presented in Table 4.

Table 4. GPS positioning accuracy at five reference points.

Measurement Point	Reference Coordinates	GPS Measurement	Error (m)
Point 1	43.2381° E, 76.9452° N	43.2383°, 76.9450°	2.1
Point 2	43.2395° E, 76.9468° N	43.2393°, 76.9471°	2.8
Point 3	43.2402° E, 76.9445° N	43.2404°, 76.9446°	1.6
Point 4	43.2378° E, 76.9479° N	43.2376°, 76.9478°	1.9
Point 5	43.2391° E, 76.9461° N	43.2390°, 76.9463°	2.2

The experimental results showed that the detection performance varied depending on target distance, illumination, object motion, and environmental conditions. Higher confidence values were generally obtained for closer targets under favorable lighting conditions, while distant or low-light targets resulted in lower detection confidence and loss of occasional detection. UAV motion and increased computational load also affected the real-time processing performance. Despite these factors, the proposed system successfully combined YOLO-based human detection with GPS positioning, allowing detected targets to be associated with their geographical coordinates during UAV operation. A quantitative comparison with recent related works is summarized in Table 5.

Table 5. Performance comparison with related UAV-based object detection systems.

Ref.	Model	Target / Dataset	Hardware Platform	Accuracy (mAP@0.5)	Speed (FPS)
[1]	Modified YOLO (FPN/BiFPN)	VisDrone / Heridal	NVIDIA	0.802	~15–25
[6]	Enhanced YOLO11	SeaDronesSee	Embedded / GPU	–	–
[9]	YOLOv10-Nano (MobileNet)	VisDrone / Benchmark	Desktop/Workstation GPU	0.394	80.5
Proposed System	Standard YOLOv8	VisDrone / Real Flight	Raspberry Pi 4 Model B (CPU)	0.742	4–8

Compared to high-throughput implementations such as [9], which achieved 80.5 FPS using YOLOv10-Nano on high-performance desktop GPU hardware, the proposed system operates at a lower rate of 4 to 8 FPS. This difference is primarily due to the severe hardware constraints of the Raspberry Pi 4 Model B, which relies exclusively on a quad-core ARM Cortex-A72 CPU without dedicated Neural Processing Units (NPUs) or onboard discrete GPUs.

Despite operating at lower frame rates than GPU-based systems, a processing speed of 4 to 8 FPS remains fully adequate for real-time UAV search and rescue operations. Assuming a standard search flight speed of 2 to 4 m/s at an altitude of approximately 5 m, processing 4 to 8 FPS provides updated target detections and synchronized GPS coordinates every 125 to 250 milliseconds. This spatial and temporal resolution ensures continuous coverage without localization blind spots, confirming the practical real-time applicability of the proposed low-cost, CPU-based system.

5. Conclusion

This study presented a UAV-based real-time human detection and localization system integrating the YOLOv8 model, Raspberry Pi 4 Model B, and u-blox NEO-M8N GPS module. The proposed system was designed to perform onboard object detection while simultaneously obtaining the geographical position of the UAV during flight.

The YOLOv8 model was trained using the VisDrone2019-DET dataset and evaluated under real-world UAV operating conditions. Experimental results showed an average detection confidence of 0.67, with a maximum confidence of 0.81. The onboard system achieved a processing speed of 4-8 FPS and detected human targets at distances of up to 30 m using a 1280×720 video stream. The GPS evaluation demonstrated positioning errors ranging from 1.6 to 2.8 m, with a mean error of approximately 2.12 m across five reference points. The obtained results indicate that the system can combine real-time human detection with GPS-based positional information during UAV operation. The main limitations were related to the computational resources of the Raspberry Pi 4, reduced detection confidence for distant or low-light targets, and variations caused by object motion and environmental conditions.

Future work will focus on improving real-time inference efficiency, enhancing detection performance for small and distant targets, and integrating more accurate localization and tracking methods for autonomous search-and-rescue missions.

Conflict of interest statement

The authors declare that they have no conflict of interest in relation to this research, whether financial, personal, authorship or otherwise, that could affect the research and its results presented in this paper

CRediT author statement

Turmaganbet U.K.: Conceptualization, Supervision, Writing-Original Draft; **Turlykozhaeva D.A.:** Writing-Review & Editing; **Akhtanov S.N.:** Data Curation, Writing-Original Draft; **Temesheva S.A.:** Writing-Review & Editing; **Ussipov N.M.:** Data Curation, Writing-Original Draft; **Tao M.:** Writing-Review & Editing. The final manuscript was read and approved by all authors.

Statement on the use of Artificial Intelligence.

During the preparation of this manuscript, artificial intelligence tools were used solely for language editing and grammatical improvement.

Data Availability Statement

The data that support the findings of this article are available upon reasonable request from the authors.

Funding

This work was supported by the Ministry of Science and Higher Education of the Republic of Kazakhstan (grant number AP32320585).

References

- 1 Ciccone, F., & Ceruti, A. (2025). Real-Time Search and Rescue with Drones: A Deep Learning Approach for Small-Object Detection Based on YOLO. *Drones*, 9(8), 514. <https://doi.org/10.3390/drones9080514>
- 2 Vaman Ravi Prasad, D. A. S. U., & Reddy, P. (2025). Autonomous Human Detection Using Drones in Search and Rescue Operations. SSRN. <https://dx.doi.org/10.2139/ssrn.5232204>
- 3 Jayalath, K., & Munasinghe, S. R. (2021). Drone-based autonomous human identification for search and rescue missions in real-time. In *2021 10th Intern. Conference on Information and Automation for Sustainability (ICIAfS)* (pp. 518-523). IEEE. <https://doi.org/10.1109/ICIAfS52090.2021.9606048>
- 4 Ho, Y.-H., & Tsai, Y.-J. (2022). Open Collaborative Platform for Multi-Drones to Support Search and Rescue Operations. *Drones*, 6(5), 132. <https://doi.org/10.3390/drones6050132>
- 5 Lyu, M., Zhao, Y., Huang, C., & Huang, H. (2023). Unmanned aerial vehicles for search and rescue: A survey. *Remote Sensing*, 15(13), 3266. <https://doi.org/10.3390/rs15133266>
- 6 Zhao, B., Zhao, J., Song, R., Yu, L., Zhang, X., & Liu, J. (2025). Enhanced YOLO11 for lightweight and accurate drone-based maritime search and rescue object detection. *PLOS ONE*, 20(7), e0321920. <https://doi.org/10.1371/journal.pone.0321920>

- 7 Vaidya, S. K., Sakthivel, V., Majumdar, R., Gupta, P., Jeevan, S., Prakash, P., & Selvamuthukumar, T. (2026). A virtual testbed for maritime search and rescue: Customizing YOLO for small object detection in unreal engine. *Journal of Ocean Engineering and Science*. <https://doi.org/10.1016/j.joes.2025.12.018>
- 8 Abdelnabi, A. A. B., & Rabadi, G. (2024). Human detection from unmanned aerial vehicles' images for search and rescue missions: A state-of-the-art review. *IEEE Access*, 12, 152009-152035. <https://doi.org/10.1109/ACCESS.2024.3479988>
- 9 Mishra, A., Narendra, M., Sinha, A., & Kumar, A. (2025). Dynamic Backbone Optimization of YOLOv10 for Real-Time Object Detection in UAV-Based Search and Rescue Missions. *IEEE Access*, 13, 195975-195986. <https://doi.org/10.1109/ACCESS.2025.3633379>
- 10 Khashush, N. A., Radif, M. J., & Abdalrdha, Z. K. (2025). A review of smart drone technologies for security surveillance and search and rescue. *Journal of Al-Qadisiyah for Computer Science and Mathematics*, 17(4), 155-173. <https://doi.org/10.29304/jqscsm.2025.17.42549>
- 11 Temesheva, S. A., Turlykozhaeva, D. A., Akhtanov, S. N., Ussipov, N. M., Zhunuskanov, A. A., Sun, W., & Tao, M. (2026). A Comprehensive Review of Metaheuristic Algorithms for Node Placement in UAV Communication Networks. *Sensors (Basel, Switzerland)*, 26(3), 869. <https://doi.org/10.3390/s26030869>
- 12 Turmaganbet, U., Zhaxebay, D., Turlykozhaeva, D., Skabylov, A., Akhtanov, S., Temesheva, S., & Tao, M. (2025). Thermal infrared object detection with yolo models. *Eurasian Physical Technical Journal*, 22(2), 121–132. <https://doi.org/10.31489/2025N2/121-132>
- 13 Šiktar, L., Čaran, B., Šekoranja, B., & Švaco, M. (2025, June). Autonomous uav navigation for search and rescue missions using computer vision and convolutional neural networks. In *International Conference on Robotics in Alpe-Adria Danube Region* (pp. 661-669). Springer Nature Switzerland. https://doi.org/10.1007/978-3-032-02106-9_73
- 14 Tran, H. K., Le, V. T., Tran, T. T., & Nguyen, V. D. (2025). Low-Cost Remotely-Operated Vehicle Navigation with GPS and RF Lora Integration. *Archives for Technical Sciences*, 2(33), 297-306. <http://dx.doi.org/10.70102/afts.2025.1833.297>
- 15 Schedl, D. C., Kurmi, I., & Bimber, O. (2021). An autonomous drone for search and rescue in forests using airborne optical sectioning. *Science Robotics*, 6(55), eabg1188. <https://doi.org/10.1126/scirobotics.abg1188>
- 16 Du, D., Zhu, P., Wen, L., Bian, X., Lin, H., Hu, Q., & Liu, Z. (2019). VisDrone-DET2019: The vision meets drone object detection in image challenge results. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 213-226. *IEEE*. <https://doi.org/10.1109/ICCVW.2019.00030>
- 17 Jocher, G., Chaurasia, A., & Qiu, J. (2023). Ultralytics yolo. *Zenodo*. <https://github.com/ultralytics/ultralytics>

AUTHORS' INFORMATION

Turmaganbet, Ulpan - PhD student, Department of Physics and Technology, al-Farabi Kazakh National University, Almaty, Kazakhstan; <https://orcid.org/0009-0002-3782-425X>; uturmaganbet@gmail.com

Turlykozhaeva, Dana - PhD, Researcher, Lecturer, Department of Physics and Technology, al-Farabi Kazakh National University, Almaty, Kazakhstan; <https://orcid.org/0000-0002-7326-9196>; turlykozhaeva.dana@kaznu.edu.kz

Akhtanov, Sayat - PhD, Researcher, Senior Lecturer, Department of Physics and Technology, al-Farabi Kazakh National University, Almaty, Kazakhstan; <https://orcid.org/0000-0002-9705-8000>; akhtanov.sayat@kaznu.edu.kz

Temesheva, Symbat - PhD student, Department of Physics and Technology, al-Farabi Kazakh National University, Almaty, Kazakhstan; <https://orcid.org/0009-0000-2795-9586>; temesheva_symbat4@live.kaznu.kz

Ussipov, Nurzhan – PhD, Researcher, Lecturer, Department of Physics and Technology, al-Farabi Kazakh National University, Kazakhstan; <https://orcid.org/0000-0002-2512-3280>; ussipov.nurzhan@kaznu.kz

Mingliang, Tao - PhD, Researcher, Professor, School of Electronics and Information, Northwestern Polytechnical University, Xian, China, <https://orcid.org/0000-0002-0329-7124>; mltao@nwpu.edu.cn