http://doi.org/10.35784/iapgos.6523
received: 27.08.2025 | revised: 09.02.2026 | accepted: 06.03.2026 | available online: 30.03.2026

DETECTION OF HUMANS IN DRONE IMAGES USING DEEP LEARNING TECHNIQUES

Sobhana Mummaneni[1], Naga Deepika Ginjupalli[1], Pragathi Dodda[1], Novaline Jacob[2], Sanjay Raj Emmanuel Katari[2]
[1]Siddhartha Academy of Higher Education, Deemed to be University, Department of Computer Science, Vijayawada, India, [2]Advanced Data Processing Research Institute, Department of Space, Hyderabad, India

Abstract. *Interest in creating reliable and fast human detection systems has increased due to the growing use of Unmanned Aerial Vehicles (UAVs) in a variety of applications, including search and rescue operations and surveillance. The YOLOv8 deep learning algorithm, specially designed for drone imagery, is used in this paper's real-time human recognition method. Because it is effective and precise at identifying people in complicated backdrops and with different lighting conditions, YOLOv8, an enhanced version of the You Only Look Once (YOLO) object detection algorithm, is used. The YOLOv8 model is trained on a large dataset of annotated drone images, encompassing diverse scenarios and environments to facilitate robust human detection. This study uses a UAV Human Dataset encompassing various environmental conditions, ensuring robust performance in challenging scenarios. This study contributes to advancing drone-based technologies for human detection, offering a scalable and efficient solution for real-world deployment. The YOLOv8 model achieved a precision of 88.83%, recall of 71%, and mAP score of 74%.*

Keywords: drones, human detection, Convolutional Neural Networks, YOLOv8

WYKRYWANIE LUDZI NA OBRAZACH Z DRONÓW Z WYKORZYSTANIEM TECHNIK GŁĘBOKIEGO UCZENIA

Streszczenie*. Zainteresowanie tworzeniem niezawodnych i szybkich systemów wykrywania ludzi wzrosło ze względu na rosnące wykorzystanie bezzałogowych statków powietrznych (UAV) w różnych zastosowaniach, w tym w operacjach poszukiwawczo-ratowniczych i nadzorze. W niniejszym artykule w metodzie rozpoznawania ludzi w czasie rzeczywistym wykorzystano algorytm głębokiego uczenia YOLOv8, zaprojektowany specjalnie do obrazów z dronów. Ze względu na swoją skuteczność i precyzję w identyfikowaniu osób na skomplikowanym tle i w różnych warunkach oświetleniowych, zastosowano YOLOv8, ulepszoną wersję algorytmu wykrywania obiektów You Only Look Once (YOLO). Model YOLOv8 został wytrenowany na dużym zbiorze danych zawierającym opatrzone adnotacjami obrazy z dronów, obejmującym różnorodne scenariusze i środowiska, aby ułatwić niezawodne wykrywanie ludzi. W niniejszym badaniu wykorzystano zbiór danych UAV Human Dataset obejmujący różne warunki środowiskowe, zapewniający niezawodne działanie w trudnych scenariuszach. Badanie to przyczynia się do rozwoju technologii wykrywania ludzi opartych na dronach, oferując skalowalne i wydajne rozwiązanie do zastosowań w rzeczywistych warunkach. Model YOLOv8 osiągnął precyzję na poziomie 88,83%, czułość na poziomie 71% i wskaźnik mAP na poziomie 74%.*

Słowa kluczowe: drony, wykrywanie ludzi, konwolucyjne sieci neuronowe, YOLOv8

Introduction

Deep learning approaches combined with unmanned aerial vehicles (UAVs), popularly called drones, have revolutionized several fields, including surveillance and disaster relief. Real-time human detection is one of these crucial applications and has ramifications for public safety, security, and search and rescue operations [12]. The globalization of devices connected to the Internet of Things is crucial in our swiftly changing society as we move towards smart homes and cities. Although video surveillance has traditionally used Closed-Circuit Television (CCTV) [4] cameras, their monitoring, position sharing, and coverage are limited. In comparison, drones fitted with vision sensors present a more adaptable and scalable surveillance method, offering increased mobility and wider coverage.

Drone surveillance [11] is a revolutionary method of monitoring and information gathering from above. Drones with sophisticated sensors and high-definition cameras may record aerial video in real-time, giving important information for a range of uses. Drones are essential to search and rescue operations because they can locate survivors, examine large regions affected by the disaster, and direct rescue crews to exact locations. Drones can also optimize irrigation techniques, detect pests, and monitor crop health in agricultural contexts [5]. Drones are used by law enforcement for search and rescue missions, traffic monitoring, and surveillance of major events. Drones are an essential instrument for surveillance in both urban and remote areas because of their adaptability and manoeuvrability, which also makes them an economical and effective way to collect data and monitor areas.

Across the Natural Disaster Mitigation and Management (NDMM) division, the use of drones to survey areas affected by disasters has become increasingly popular. It takes a lot of time and is prone to error for human observers to analyse videos manually. Finding people buried beneath rubble after earthquakes or floods is made possible with the use of drone-captured photos for human identification. Unmanned aerial vehicle (UAV) models and drones with cameras can be used to examine impacted areas and pinpoint specific locations that need help for efficient research, security, and search and rescue (SAR) [16] activities.

In computer vision, object detection is being advanced through the use of deep learning techniques. Deep learning models [13] have revolutionized object detection by utilizing neural networks and convolutional architectures to allow automated systems to locate and identify things inside pictures or video streams with remarkably high accuracy and efficiency. Robust detection across multiple object classes and environmental conditions is made possible by deep learning algorithms that extract complex features from visual data using methods like YOLO models, single-shot detectors (SSD), and region-based convolutional neural networks [3] (R-CNN). In addition, the fusion of deep learning and computer vision methods has spurred progress in real-time object detection, opening up new applications in fields such as medical imaging, augmented reality, and autonomous cars. As research in this field continues to evolve, deep learning algorithms stand at the forefront, driving innovation and unlocking new possibilities for object detection in diverse domains [17].

This paper presents a comprehensive investigation into the development and implementation of a real-time human detection system using YOLOv8 and drone technology. We delve into the technical details of YOLOv8, highlighting its architecture and key features that make it well-suited for aerial surveillance [10] tasks. Furthermore, we discuss the challenges associated with human detection from drone images and how YOLOv8 addresses these challenges.

We validate our proposed system experimentally and evaluate its performance to show that it can effectively identify humans in real-time from drone-captured photos. Furthermore, we present an analysis of our system's performance in comparison to current approaches, demonstrating YOLOv8's [14] advantage in terms of processing speed and detection accuracy.

As a whole, this work advances the field of airborne surveillance applications utilizing human detection [18] technology by providing a workable and effective method for monitoring and analysing data in real time across a variety of fields. By harnessing the power of YOLOv8 and drone technology, we aim to enhance security, safety, and situational awareness in diverse operational environments.

1. Literature review

To improve search and rescue operations, this study [7] proposes a human identification system using drones. In real-time aerial recordings, the drone recognizes signals of human presence. After that, it moves independently toward the uncertain location to get a better look and verify human presence. Now that operator override is possible, the drone is fully self-sufficient. Following photo processing, it gives the operator directions for the next tasks and a selection of the frames for verification. An object detector consisting of a Tensor flow neural network that has been specially constructed is used to identify human items on the ground. A single-board computer processes photos in real time receives flight directions from the flight controller for autonomous navigation, and reads GPS location. Because of its autonomous architecture, the drone can recognize human presence in real-time aerial footage and can navigate to potentially dangerous areas without direct human intervention.

The goal of the suggested approach [8] of data merging and YOLOv8 is to improve autonomous driving's object identification accuracy in inclement weather. By using custom datasets with diverse harsh weather scenarios, applying transfer learning, data augmentation techniques, and training on merged datasets, the approach improves the model's ability to detect objects in challenging weather conditions. Results show that training on merged datasets outperforms training on individual datasets, indicating the effectiveness of the data merging technique in improving detection accuracy. By identifying a boat and an airplane, the suggested approach of data merging using YOLOv8 for object detection in bad weather improved the accuracy from 0.781 to 0.824. Nevertheless, in an alternative situation, the accuracy dropped from a mAP50 score of 0.747 to 0.597 as the boat and the airplane were absent in the training datasets.

YOLO- based architecture combined with gradient boosting for the detection and recognition of various human actions, addressing the challenges associated with action recognition from drone-captured imagery. They leverage the "Okutama-Action" dataset [1], which contains diverse scenarios while controlling for acquisition parameters like camera angle and flight altitude. By integrating YOLOv5 with a gradient-boosting classifier, the authors aim to create a scalable and efficient system for single-image action identification. Through an ablation study, they assess different YOLOv5 architectures and evaluate their method's performance on the Okutama-Action dataset. Results indicate that their approach outperforms previous methods on the dataset, attributed to both YOLOv5's [2] efficiency and the suitability of their pipeline for the dataset's characteristics in terms of bias-variance trade-off. They highlight limitations related to gimbal angle, altitude, drone speed, and environmental conditions, proposing potential future directions such as exploring YOLOv7 or self-attention transformers for improved action detection.

TensorFlow object detection API is used for object detection on drone footage. It compares SSD and Faster R-CNN, two exemplary convolutional target detection systems, using base feature extractors such as ResNet50, GoogleNet/Inception, and MobileNet [15]. Both models achieve a maximum accuracy of 99% and an average of around 85%, demonstrating greater detection accuracy for people, cars, trees, and buildings. SSD prioritizes scale, aspect ratio, and prediction sampling location, resulting in an average frame processing time of 115 ms but with a lower detection rate. Conversely, Faster R-CNN exhibits higher accuracy, recognizing almost 95% of objects in the scene, albeit with a longer average frame processing time of at least 140 ms. Overall, SSD outperforms Faster R-CNN in terms of speed; the latter is more accurate but slower overall, particularly when the proposed regions are limited to increased performance. Furthermore, MobileNet, ResNet50, and GoogleNet/Inception V2 runtime and memory needs were examined. Less than 1 GB is all that MobileNet uses, while ResNet50 uses about 5 GB on Faster R-CNN, and GoogleNet/Inception V2 is somewhere in the middle at less than 2 GB. For all models and feature extractors combined, a substantial relationship between running time and memory is seen.

Existing methods, such as Faster-RCNNs, SSD, and Region-based Fully Convolutional Networks (R-FCN), have been employed for human detection and action recognition. However, these methods have limitations, including small datasets and fewer classes. In this study [6], the authors propose using the YOLOv3 algorithm for detection and action recognition, utilizing a dataset consisting of 1996 images across eight classes. Comparative analysis reveals that YOLOv3 achieves the highest accuracy of 94.9% compared to existing algorithms, which range from 53% to 93%. The proposed model not only demonstrates superior accuracy but also reduces the time required for image detection to 0.40 milliseconds, outperforming existing methods.

This paper [9] introduces an aerial image detection system aimed at addressing the challenges in UAV multitarget detection encountered in both civil and military applications. The proposed model introduces several innovations to enhance performance and robustness. Firstly, the Bi-PAN-FPN concept is applied to enhance the neck portion of YOLOv8, resolving the prevalent problem of misidentifying and failing to detect small targets in aerial photos. Second, a portion of the C2f module is replaced by the GhostblockV2 structure, which is integrated into the framework of the benchmark model. This structure decreases the number of model parameters while mitigating information loss during long-distance feature transmission. The proposed algorithm's performance is assessed using the popular benchmark dataset VisDrone2019. Results indicate significant improvements over the baseline model, with a 9.06% increase in mean average precision (mAP) on the test set and a 13.21% reduction in parameters. Moreover, the proposed method demonstrates superior accuracy compared to six other algorithms. The proposed model achieves a mean average precision (mAP) of 88.83% at an IoU threshold of 0.5, while maintaining a real-time inference speed of 293 frames per second (FPS).

2. Proposed methodology

The process flow diagram of the proposed system starts with data loading, data preprocessing, model building, training the model, evaluation, and testing. Fig. 1 represents the flow diagram of the proposed system. The proposed model consists of 5 modules.

Data Collection: The dataset we used for training and testing the models is the image dataset titled "UAV Human Dataset". We have randomly chosen some 2000 images from the dataset where 1600 images are used for training and the remaining for testing the model. Once the model is tested properly it is validated on the dataset provided by 'ADRIN'.

Data Pre-processing: This step is essential for working with any kind of dataset. Because uncleaned(noisy) data will not give better results. Preprocessing of the dataset is done in two steps in our research Data Augmentation, and Data Labelling. The dataset obtained after preprocessing is used to train the model.

Data Augmentation: The images in the UAV Human Dataset are flipped and rotated for better processing. The size of the dataset is also increased to improve the accuracy of the model.

Fig. 1 depicts the process of the proposed model from loading the images collected from the UAV Human Dataset. Then the data is pre-processed by doing data augmentation, and data labelling. Then the images in the dataset is divided into training and testing. The YOLOv8 model is created and the images are given to the model for training. Model performance is evaluated using testing images. When a new unseen human image is given to the trained model the model can detect the image as human.

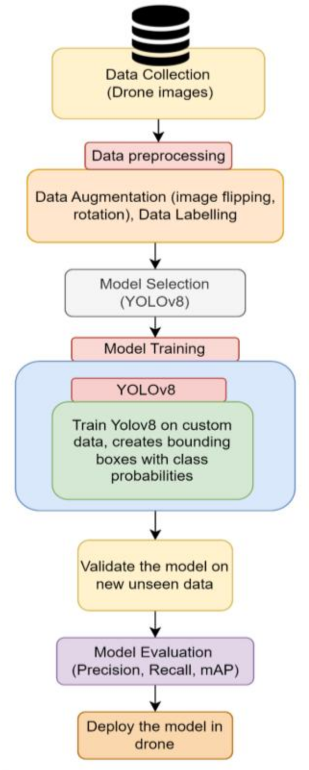

Fig. 1. Architecture of the YOLOv8 model for real-time human detection

Annotate the dataset using tools like LabelImg to mark bounding boxes around objects and assign class labels. This step creates ground truth annotations required for training the YOLOv8 model.

***Data Labelling**:* The images in the dataset are labelled using LabelImg software. The labelled images are saved in the YOLO format (.txt) and the images and their corresponding labels should be in the same directory in separate folders. LabelImg is a free and open-source graphical application for annotating images in object detection tasks. It is commonly used in machine learning projects where supervised learning algorithms require labelled datasets for training.

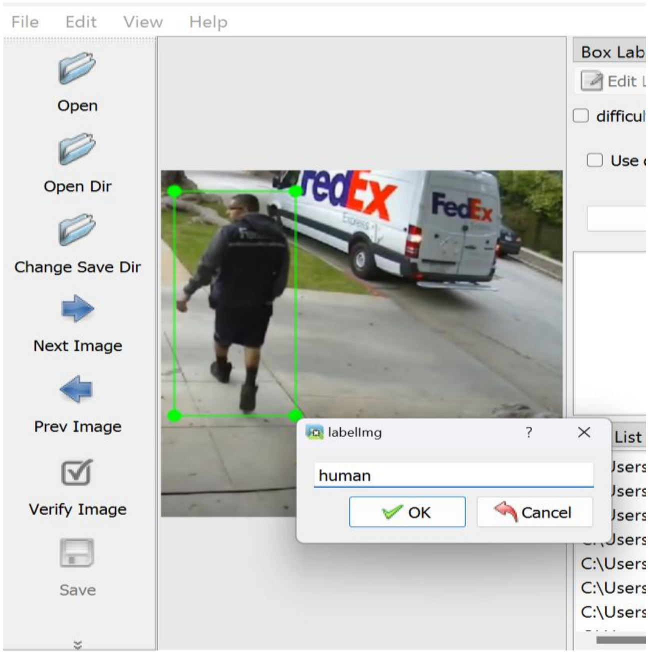

Fig. 2. Labelling of images

Fig. 2 depicts the labelling of images in the UAV Human Dataset. LabelImg is written in Python and uses Qt for its graphical interface. Firstly, LabelImg should be installed. After installation, you can launch LabelImg by running it. Once LabelImg is running, you can load images by clicking on 'Open Dir' or 'Open File' and navigating to the directory or file containing the images you want to label. LabelImg provides tools for drawing bounding boxes around objects in images. Then the image is annotated as 'human'.

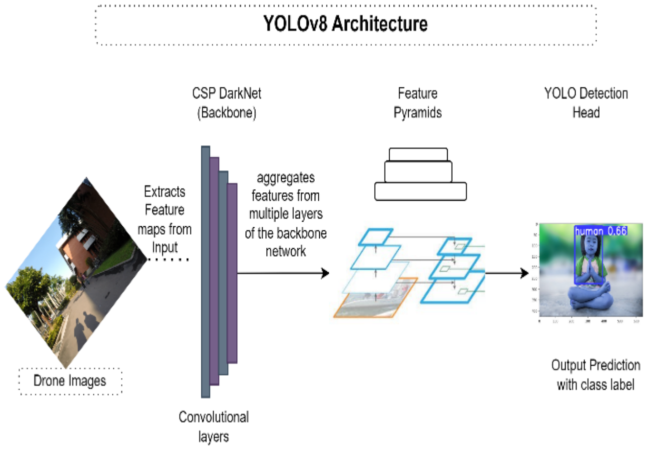

Fig. 3. YOLOv8 Architecture

Model Selection: The YOLO architecture, which is renowned for its quickness and precision in real-time object identification, is the foundation of YOLOv8. To boost detection performance, YOLOv8 especially includes architectural design enhancements.

Feature Extraction Backbone: YOLOv8 usually uses a feature extraction backbone, like CSPDarknet or Darknet, to efficiently extract features from input images. These backbones consist of convolutional layers that capture hierarchical representations of input images.

Detection Head Design: To anticipate bounding boxes, objectness scores, and class probabilities for identified objects, YOLOv8 makes use of a detection head made up of convolutional layers. This detection head is designed to output predictions at multiple scales to handle objects of various sizes.

Model Development: The proposed model was built using the YOLOv8 algorithm. Augment the training data to increase the diversity of the dataset. Train the model using the dataset specified in the YAML file located at `yaml_path`. The training process is set to run for 20 epochs with a batch size of 32 samples. The initial learning rate (`lr0`) is set to 0.001, and it gradually decreases to 0.1 (`lrf`) throughout training. Train the YOLOv8 model on the annotated dataset using the Adam optimization algorithm. During training, the model adjusts its parameters to minimize the defined loss function and improve detection performance. Images are resized to a square of size 640x640 pixels before being fed into the model.

Model Evaluation: To evaluate the model's performance on unseen data, the trained model is tested on the validation set that helps assess the generalization ability of the model and identify potential overfitting issues. Performance is measured using metrics such as mean average precision (mAP), which measures the precision and recall of object detection predictions across different confidence thresholds. The YOLOv8 model achieved a precision of 84%, with a recall of 71%, and an mAP score of 74%.

Algorithm 1 describes the process of training the YOLOv8 model. A pre-trained YOLOv8 model is downloaded and a configuration file is built to train the YOLOv8 with training data and evaluate it.

Algorithm 1: Algorithm for building YOLOv8 model

Input:	Pre-processed UAV Human Image dataset.
Output:	Trained YOLOv8 model for human detection.
1:	BEGIN
2:	Installed the Ultralytics package.
3:	Load the Preprocessed Image dataset.
4:	from sklearn.model selection train test split is imported to split the data in the ratio of 80:20 into training and testing samples.
5:	Load the pre-trained YOLOv8 model.
6:	Configure the yaml file.
7:	Train the model over training data and evaluate it on test data.
8:	END

3. Results and discussion

To perform Real-Time Human Detection on drone images we used a dataset "UAV Human Dataset." There are 2000 images in the dataset we used 1600 images for training and the remaining for testing the model. Once the model is tested properly it is validated on the dataset provided by 'ADRIN'. The images in the dataset are increased by performing data augmentation also the images are flipped and rotated to improve the accuracy of the YOLOv8 and Fast RCNN models. As YOLO is a supervised Learning Algorithm, In supervised learning, the algorithm is trained on a dataset where each input (image) is associated with a corresponding target output (bounding boxes and class labels). So, the dataset's images are annotated using the LabelImg tool to create bounding boxes. Fast RCNN uses region proposals to detect the human. YOLOv8 outperformed the Fast RCNN model by achieving a precision of 84%, recall of 71%, and mAP score of 74%.

The YOLOv8 model demonstrates strong performance with a precision of 88.83%, recall of 71%, and an mAP score of 74%, making it highly effective for human detection in drone-based surveillance and SAR applications. On edge devices like the Jetson Nano, the model achieves a real-time FPS of around 5-15 at a resolution of 640x640, depending on applied optimizations. On more powerful GPUs, FPS can reach 30-60, with the potential for even higher rates when optimizations such as TensorRT or model pruning are utilized. These results highlight the balance between detection accuracy and real-time processing capability, making YOLOv8 suitable for a range of UAV applications.

This human detection model's performance is assessed using several measures. Utilizing the Precision-recall, Precision-confidence, Recall-confidence, F1-confidence, and Confusion Matrix curves, the deep learning model is examined. True Positive (TP) refers to an instance where the model's projected result was also positive in real life. This shows that the model correctly identified the occurrence of the condition. False positive (FP) is used to describe situations where the model projected a good outcome but the actual outcome was negative. This suggests that a condition that was not present was mistakenly recognized as present by the model. True Negative (TN) refers to a situation in which the model's projected outcome was negative and the outcome was negative as well. This shows that the model correctly identified a condition's absence. False Negative (FN) refers to a situation in which the model projected a negative outcome, but the outcome was favourable. This suggests that a condition was present but the model was unable to detect it.

Fig. 4 represents the confusion matrix of the YOLOv8 model when applied to the test data. It is noticeable that the model can detect human image samples more effectively compared to non-human images.

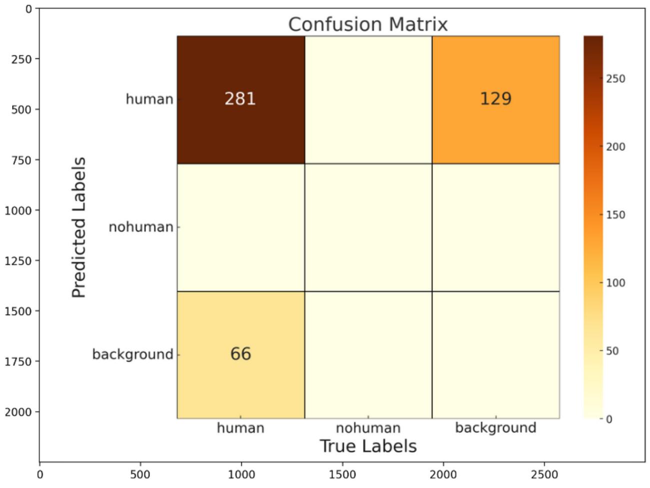

Fig. 4. Confusion Matrix of YOLOv8 model

The YOLOv8 model's precision-recall graph is shown in Fig. 5. The model obtained a precision score of 0.742 at mAP@0.5, which indicates that, when assessed using an IoU threshold of 0.5, the YOLOv8 model for human detection achieved a mean average precision of 0.742.

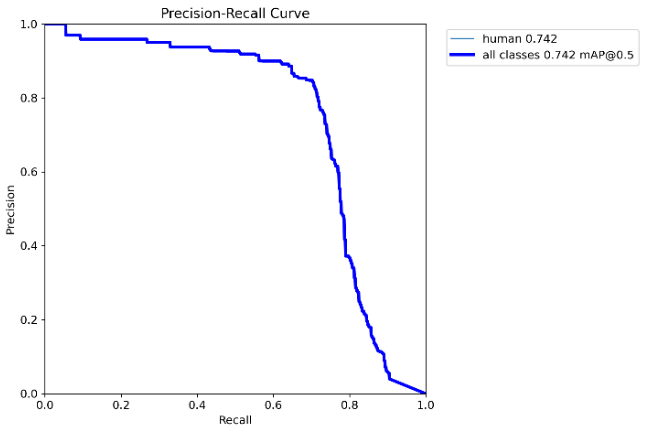

Fig. 5. Precision-Recall curve of YOLOv8 model

Fig. 6 describes the precision confidence graph of the YOLOv8 model, the model achieved all classes 1.00 at 0.927 indicating that across all classes (not just human detection), the precision reaches 1.00 (or 100%) at a confidence threshold of 0.927.

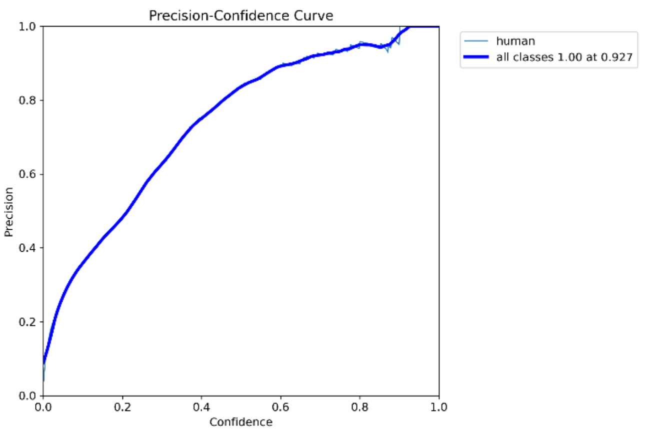

Fig. 6. Precision-Confidence curve of YOLOv8 model

Fig. 7 describes the Recall Confidence graph of the YOLOv8 model, the model achieved all classes 0.89 at 0.00 indicating that across all classes (not just human detection), the recall reaches 0.89 (or 89%) at a confidence threshold of 0.00.

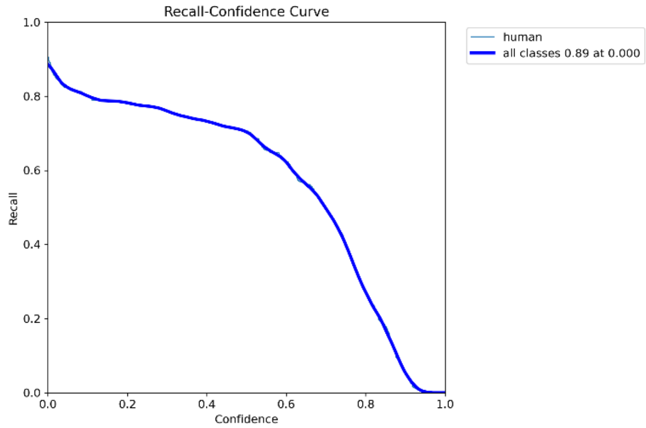

Fig. 7. Recall-Confidence curve of YOLOv8 model

Fig. 8 describes the F1 Confidence graph of the YOLOv8 model, the model achieved all classes 0.76 at 0.494 indicating that across all classes (not just human detection), the recall reaches 0.76 (or 76%) at a confidence threshold of 0.494.

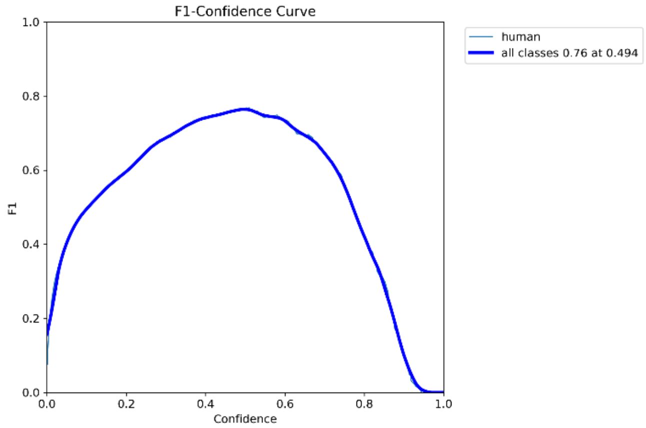

Fig. 8. F1-Confidence curve of YOLOv8 model

Fig. 9 shows the snapshot of the application that depicts the case of a 'Human'. In the GUI developed for Human Detection Using YOLOv8, the GUI takes input as an image and detects for humans. In this case, the model detects it as Human.

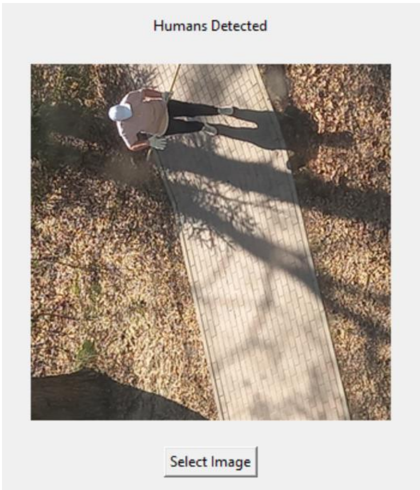

Fig. 9. Human detection by YOLOV8 model

Fig. 10 shows the snapshot of the application that depicts the case of a 'Human' by creating bounding boxes. The way how YOLOv8 detects humans is shown clearly by drawing bounding boxes with the label person and a confidence score of 0.89.

Fig. 10. Human detection with bounding boxes by YOLOV8 model

Fig. 11 shows the snapshot of the application that depicts the case of a 'No Human'. In the GUI developed for Human Detection Using YOLOv8, the GUI takes input as an image and detects it for humans. In this case, the model detects it as a No Humans.

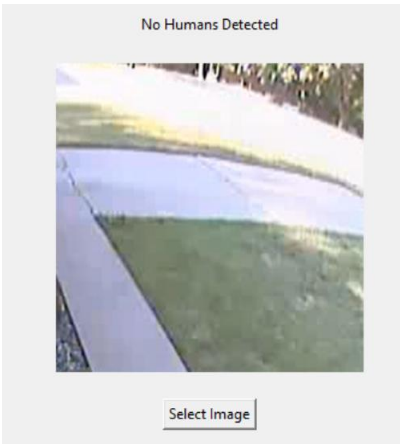

Fig. 11. No humans were detected by the YOLOV8 model

Fig. 12 shows the snapshot of the application when the model deals with faraway objects. The model detects humans in the frame who are close. The model is not able to detect the humans who are very far in the image. The model is only able to detect medium-sized humans in the image.

Fig. 12. Human detection when humans are far away

Fig. 13 illustrates the application's performance with medium-sized objects positioned at a medium distance within the image. The model can detect all the humans in the image as they are medium-sized.

Fig. 13. Human detection when humans are a medium distance away

Fig. 14 illustrates the application's performance with large-sized objects that are close to the image. The model can detect all the humans in the image as they are large-sized.

Fig. 14. Human detection when humans are close

Fig. 15 illustrates the application's performance with a smaller number of humans in the image. The model can detect all the humans in the image, as it is sparsely populated.

Fig. 15. Human detection when humans are sparsely populated

Fig. 16 illustrates the application's performance with a greater number of humans in the image. The model can detect most of the humans in the image, as it is densely populated.

Fig. 16. Human detection when humans are densely populated

Model performance is sensitive to environmental factors, including image lighting. Dim lighting significantly reduces the model's human detection accuracy.

Table 1. Comparison of performance of proposed work with previous implementations

Literature	Algorithm	Accuracy
K. Jayalath, and S. R. Munasinghe [7]	Fast R-CNN	79%
Ahmad, Tasweer, et al. [1]	YOLOv5, FPN	84%
Sun, Chenfan, et al. [15]	SSD, Fast RCNN	78%
Proposed work	YOLOv8	88.83%

4. Conclusion

In conclusion, our study on real-time human detection utilizing deep learning techniques, specifically YOLOv8 and Fast R-CNN, has demonstrated notable advancements in aerial surveillance and security applications. Through rigorous evaluation, YOLOv8 exhibited superior precision, achieving an impressive 82% accuracy compared to Fast R-CNN's 70%. This substantial margin underscores the efficacy of YOLOv8 in detecting humans efficiently and accurately from drone images in real-time scenarios. The enhanced precision of YOLOv8 not only strengthens its applicability in surveillance tasks but also underscores its potential to contribute significantly to various fields such as search and rescue operations, law enforcement, and environmental monitoring. As we continue to refine and optimize deep learning methodologies, YOLOv8 stands as a promising solution for real-time human detection from aerial imagery, offering heightened performance and reliability in critical situations. Furthermore, our study adds to the current discussion on how well deep learning models do in comparison when it comes to human detection tasks. While YOLOv8 has demonstrated superior performance in various object detection benchmarks, including speed and accuracy, its efficacy specifically in the context of real-time human detection from drone images remains to be thoroughly explored. Through empirical evaluation and comparative analysis, we seek to elucidate the strengths and limitations of YOLOv8 and Fast R-CNN in this domain.

Acknowledgments

The authors would like to express their sincere gratitude to Dr. PV Radhadevi, Director of ADRIN, for providing the funding that made this research possible. This invaluable support allowed us to make significant progress in our work on remote sensing and edge computing.

References

[1] Ahmad, T., Cavazza, M., Matsuo, Y., & Prendinger, H. (2022). Detecting Human Actions in Drone Images Using YoloV5 and Stochastic Gradient Boosting. *Sensors*, *22*(18), 7020. https://doi.org/10.3390/s22187020

[2] Chen, W., Huang, H., Peng, S., Zhou, C., & Zhang, C. (2021). YOLO-face: A real-time face detector. *The Visual Computer*, *37*(4), 805–813. https://doi.org/10.1007/s00371-020-01831-7

[3] Deng, Z., Sun, H., Zhou, S., Zhao, J., Lei, L., & Zou, H. (2018). Multi-scale object detection in remote sensing imagery with convolutional neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, *145*, 3–22. https://doi.org/10.1016/j.isprsjprs.2018.04.003

[4] Dilshad, N., Hwang, J., Song, J., & Sung, N. (2020). Applications and Challenges in Video Surveillance via Drone: A Brief Survey. *2020 International Conference on Information and Communication Technology Convergence (ICTC)*, 728–732. https://doi.org/10.1109/ICTC49870.2020.9289536

[5] Hafeez, A., Husain, M. A., Singh, S. P., Chauhan, A., Khan, Mohd. T., Kumar, N., Chauhan, A., & Soni, S. K. (2023). Implementation of drone technology for farm monitoring & pesticide spraying: A review. *Information Processing in Agriculture*, *10*(2), 192–203. https://doi.org/10.1016/j.inpa.2022.02.002

[6] Hassan, N. I., Tahir, N. Md., Zaman, F. H. K., & Hashim, H. (2020). People Detection System Using YOLOv3 Algorithm. *2020 10th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, 131–136. https://doi.org/10.1109/ICCSCE50387.2020.9204925

[7] Jayalath, K., & Munasinghe, S. R. (2021). Drone-based Autonomous Human Identification for Search and Rescue Missions in Real-time. *2021 10th International Conference on Information and Automation for Sustainability (ICIAfS)*, 518–523. https://doi.org/10.1109/ICIAfS52090.2021.9606048

[8] Kumar, D., & Muhammad, N. (2023). Object Detection in Adverse Weather for Autonomous Driving through Data Merging and YOLOv8. *Sensors*, *23*(20), 8471. https://doi.org/10.3390/s23208471

[9] Li, Y., Fan, Q., Huang, H., Han, Z., & Gu, Q. (2023). A Modified YOLOv8 Detection Network for UAV Aerial Image Recognition. *Drones*, *7*(5), 304. https://doi.org/10.3390/drones7050304

[10] Liu, W., Ren, G., Yu, R., Guo, S., Zhu, J., & Zhang, L. (2022). Image-Adaptive YOLO for Object Detection in Adverse Weather Conditions. *Proceedings of the AAAI Conference on Artificial Intelligence*, *36*(2), 1792–1800. https://doi.org/10.1609/aaai.v36i2.20072

[11] Manapongpun, P., Karoonkornsakul, C., Peechaphand, K., Kriengkomol, P., Rajawana, A., Ritmetee, P., Lounsrimongkol, N., & Chenchai, N. (2022). DroneBox: A Fully Automated Drone System for Surveillance Application. *Offshore Technology Conference Asia*, D031S015R003. https://doi.org/10.4043/31685-MS

[12] Mishra, B., Garg, D., Narang, P., & Mishra, V. (2020). Drone-surveillance for search and rescue in natural disaster. *Computer Communications*, *156*, 1–10. https://doi.org/10.1016/j.comcom.2020.03.012

[13] Pal, S. K., Pramanik, A., Maiti, J., & Mitra, P. (2021). Deep learning in multi-object detection and tracking: State of the art. *Applied Intelligence*, *51*(9), 6400–6429. https://doi.org/10.1007/s10489-021-02293-7

[14] Sudharson, D., Srinithi, J., Akshara, S., Abhirami, K., Sriharshitha, P., & Priyanka, K. (2023). Proactive Headcount and Suspicious Activity Detection using YOLOv8. *Procedia Computer Science*, *230*, 61–69. https://doi.org/10.1016/j.procs.2023.12.061

[15] Sun, C., Zhan, W., She, J., & Zhang, Y. (2020). Object Detection from the Video Taken by Drone via Convolutional Neural Networks. *Mathematical Problems in Engineering*, *2020*, 1–10. https://doi.org/10.1155/2020/4013647

[16] Valarmathi, B., Kshitij, J., Dimple, R., Srinivasa Gupta, N., Harold Robinson, Y., Arulkumaran, G., & Mulu, T. (2023). Human Detection and Action Recognition for Search and Rescue in Disasters Using YOLOv3 Algorithm. *Journal of Electrical and Computer Engineering*, *2023*, 1–19. https://doi.org/10.1155/2023/5419384

[17] Wei, B., Hao, K., Tang, X., & Ding, Y. (2019). A new method using the convolutional neural network with compressive sensing for fabric defect classification based on small sample sizes. *Textile Research Journal*, *89*(17), 3539–3555. https://doi.org/10.1177/0040517518813656

[18] Zhang, R., Jing, X., Wu, S., Jiang, C., Mu, J., & Yu, F. R. (2021). Device-Free Wireless Sensing for Human Detection: The Deep Learning Perspective. *IEEE Internet of Things Journal*, *8*(4), 2517–2539. https://doi.org/10.1109/JIOT.2020.3024234

Prof. Sobhana Mummaneni
e-mail: sobhana@vrsiddhartha.ac.in

Sobhana Mummaneni is an associate professor at the Department of Computer Science and Engineering, Siddhartha Academy of Higher Education, Deemed to be University, Vijayawada, India. She holds a Ph.D. in Computer Science and Engineering from Krishna University, and she has been teaching for sixteen years.
Her research interests are in the fields of artificial intelligence, machine learning, data analytics, cyber security, and software engineering. She has published fifty six papers in national and international journals and has seven patents to her name.

https://orcid.org/0000-0001-5938-5740

Eng. Naga Deepika Ginjupalli
e-mail: nagadeepikaginjupalli@gmail.com

Naga Deepika Ginjupalli has graduated from Siddhartha Academy of Higher Education, Deemed to be University, Vijayawada, India, with a degree in computer Science and Engineering. She is interested in data science, machine learning, and artificial intelligence.

https://orcid.org/0009-0005-0436-0008

Eng. Pragathi Dodda
e-mail: pragathidodda@gmail.com

Pragathi Dodda graduated from V Siddhartha Academy of Higher Education, Deemed to be University, Vijayawada, India, with a degree in computer science and engineering.
She has a strong interest in artificial intelligence, data analytics, and machine learning.

https://orcid.org/0009-0003-4879-6298

Ph.D. Novaline Jacob
e-mail: novalinejacob@gmail.com

Novaline Jacob, M.Sc.,M.Tech.,Ph.D, is a scientist at Advanced Data Processing Research Institute (ADRIN), Dept. of Space, Secunderabad.
She has acquired her master's degree in applied geology and remote sensing from Anna University, Chennai.

https://orcid.org/0009-0007-8627-9231

M.Sc. Sanjay Raj Emmanuel Katari
e-mail: esanjayraj@gmail.com

Emanuel Sanjay Raj Katari is heading Video Analytics Section at Advanced Data Processing Research Institute (ADRIN) Dept. of Space. He received degree in M.Sc. (physics) from Osmania University, 1994 and M.Sc. (computer science), 2002. His research interests include computer vision, intelligent video surveillance systems, content-based image retrieval, digital watermarking, visual cryptography and steganalysis.

https://orcid.org/0009-0003-6389-6625