Human recognition is widely used in variety of fields such as autonomous vehicles, surveillance field, automatons, assisting blind peoples and many more. Many machine learning (ML) and deep learning (DL) algorithms exist for video analysis the main motive of these algorithms is to find human in complicated image. The research presented in this paper focuses on the development of an energy-efficient, smart CCTV camera system for real-time human detection, utilizing the YOLOv8 (You Only Look Once) model. The problem addressed is the need for more advanced, autonomous surveillance systems capable of human detection under various background conditions, overcoming the limitations of traditional CCTV systems, which require constant manual monitoring. The proposed system was trained on the PASCAL VOC 2012 dataset and optimized through hyperparameter tuning, achieving high accuracy and real-time performance. Key results demonstrate that the YOLOv8 model, implemented on the NVIDIA Jetson Nano platform, offers remarkable accuracy, precision, and energy efficiency. It consistently detects human figures in real-time, even in non-ideal conditions like poor lighting or complex backgrounds. This success can be attributed to YOLOv8's cross-stage partial network (CSPNet) architecture, which enhances its ability to process images quickly and accurately, ensuring it meets the demands of continuous surveillance. The distinguishing features of this system are its energy-efficient design and adaptability to diverse environmental conditions. These characteristics not only solve the challenge of real-time human detection but also make the system a robust and scalable solution for modern security and surveillance applications

Keywords: human detection, CCTV, deep learning, YOLOv8, NVIDIA Jetson nano, CSPNet

UDC 681
DOI: 10.15587/1729-4061.2024.310401

DEVELOPMENT OF AN ENERGY-EFFICIENT CCTV CAMERA SYSTEM FOR REAL-TIME HUMAN DETECTION USING YOLOV8 MODEL

Meghana Deshpande**
Corresponding author
Research Scholar, Assistant Professor*
E-mail: meghanaspujari@gmail.com
Alok Agarwal
Research Guide, Professor*
Department of Electronics & Communication Engineering
Aravali College of Engineering and Management
Jasana, Tigaon Rd, Neharpar Faridabad, Haryana, India, 121102
Rupali Kamathe**
Research CO-Guide, Professor*
*Department of Electronics & Communication Engineering
Shri Jagdishprasad Jhabarmal Tibrewala University
56GW+P63, Churu Road, Vidyanagari, Churela, Rajasthan, India, 333001
**Department of Electronics and Telecommunication (E&TC) Engineering
Progressive Education Society's Modern College of Engineering
GRGW+F5J, 1186/A, Off J.M. Road, Shivaji Nagar, Modern Engineering College Rd, Shivajinagar, Pune, Maharashtra, India, 411005

Received 18.09.2024
Received in revised form 01.11.2024
Accepted 25.11.2024
Published 27.12.2024

How to Cite: Deshpande, M., Agarwal, A., Kamathe, R. (2024). Development of an energy-efficient cctv camera system for real-time human detection using yolov8 model. Eastern-European Journal of Enterprise Technologies, 6 (9 (132)), 47–59. https://doi.org/10.15587/1729-4061.2024.310401

1. Introduction

The field of human detection in surveillance systems continues to be a key focus of research due to the growing demand for enhanced security measures in various sectors, including public safety, infrastructure protection, and autonomous systems [1, 2]. With rapid advancements in ML and DL algorithms, there is a significant shift toward automating human detection in real-time video analysis [3]. This shift is crucial for creating intelligent surveillance systems that can operate efficiently and autonomously in diverse environmental conditions. The relevance of this research topic lies in its direct application to real-world scenarios where timely detection and response are essential for maintaining security and preventing incidents. Traditional CCTV systems, while still widely used, suffer from significant limitations. These systems require continuous manual monitoring, which is labor-intensive and prone to human error, particularly due to fatigue and distraction [4]. Furthermore, the inability of these systems to provide real-time alerts or differentiate between routine activities and significant events results in large volumes of data that need to be manually reviewed, reducing the system's overall effectiveness [5]. Given the rise in security threats and the increasing complexity of environments requiring surveillance, the need for more advanced solutions has never been more pressing [6]. Hence, research into more intelligent, automated systems for real-time human detection remains critical [7].

Advances in DL, particularly the use of convolutional neural networks (CNNs), have demonstrated superior performance in object detection tasks, including human recognition [8, 9]. CNNs are highly effective at processing pixel data and have been pivotal in advancing video and image analysis. One of the most prominent architectures in this domain is You Only Look Once (YOLO), which has gained widespread recognition for its speed and accuracy in real-time object detection [10]. The YOLO family of models, including the latest YOLOv8, processes images in a single pass, making it particularly suitable for applications requiring immediate responses, such as human detection in surveillance systems. The growing adoption of smart surveillance systems further highlights the importance of this research. Modern systems not only need to detect humans accurately but must also perform efficiently under varying environmental conditions, including changes in lighting, background complexity, and object occlusion [11, 12]. This demand is further compounded by the requirement for energy efficiency, particularly in continuous surveillance systems that must operate around the clock. By integrating DL algorithms like YOLOv8 with energy-efficient platforms such as the NVIDIA Jetson Nano, researchers can address the dual challenge of high performance and low power consumption.

Therefore, the study of advanced human detection techniques within surveillance systems remains highly relevant in today's scientific landscape. The ability to create more autonomous, real-time systems that adapt to dynamic environments is essential for modern security infrastructures. This research topic not only aligns with the current needs of the industry but also addresses gaps in existing technologies by focusing on improving accuracy, speed, and energy efficiency, making it a timely and necessary area of investigation.

2. Literature review and problem statement

In recent literature, various studies have explored the effectiveness of YOLO models for human detection and related applications. The research in [13] paper presents YOLOv8 for human detection in thermal images, specifically for search and rescue missions. The study demonstrates that YOLOv8 achieves high precision around 95 % in challenging conditions like low light and adverse weather. However, unresolved issues remain, such as reduced accuracy in complex environments due to occlusions and thermal reflections. These challenges could be addressed through methods like sensor fusion or data augmentation. Prior research has explored similar enhancements, but the balance between model accuracy and processing speed remains a challenge, indicating the need for further studies to optimize these factors. Work in [14] presents research on human action recognition and localization using the YOLO model in video analytics. It demonstrated that YOLO can efficiently detect and label actions in real-time, but unresolved issues include handling occlusion and background clutter, which may affect detection accuracy. These challenges arise due to the complexity of real-world environments, such as varying lighting conditions and occluded objects, which limit the model's performance. A potential solution could involve integrating multiple sensor inputs or advanced image preprocessing techniques. This indicates the need for further research to improve detection accuracy in complex environments while maintaining computational efficiency. Our proposed approach aims to address these limitations by leveraging the YOLOv8 model, which offers faster processing and better accuracy than earlier versions.

The paper by [15] presents research on human behavior recognition using DL, specifically employing the YOLOv3 model. It shows that YOLOv3 achieved an accuracy of 80.2 % at approximately 15 FPS in recognizing human behaviors. However, handling overlapping objects and complex environmental conditions is an issue which affect detection accuracy. The computing needs of processing huge datasets and the difficulties of precisely detecting behaviors in cluttered situations are probably the causes of these limitations. Enhancing network architectures or adding more sensor inputs are potential solutions to these problems. Similar methods have been employed in other investigations, however they frequently come with significant computational expenses. Work reported by [16] shows a comparative analysis of various human detection techniques, including frame subtraction, HOG, and the YOLO algorithm. It shows that while YOLO offers the highest accuracy (up to 96.8 %), unresolved issues like slower response times and difficulties in high-density or occluded environments persist. These limitations may arise from YOLO's high computational demand and challenges in complex environments. Solutions could include algorithm optimization or hybrid methods. This approach has been attempted in prior research using HOG and frame subtraction, but they suffer from reduced accuracy and reliability in dynamic settings. This suggests the need for further research to enhance detection in real-time, high-accuracy systems. Work [17] reports YOLOv8 model for human detection in thermal imaging, particularly distinguishing injured from non-injured individuals. It shows that the model achieved a mAP of 0.772 and processing times of 12.3 ms for images, but unresolved issues remain, such as slower processing in video datasets and difficulty detecting humans in complex postures. These challenges stem from the higher computational demand of video processing and thermal imaging limitations. A way to overcome this could be integrating tracking algorithms or improving hardware efficiency. While similar approaches, such as using YOLOv5, have been explored, they often neglect injured human detection. The suggested approach YOLOv8 for real-time detection in diverse environments can tackle these limitations by optimizing, improving both accuracy and computational efficiency for smart CCTV applications.

In [18], YOLO has been used for person detection from an overhead view, achieving a TPR of 95 % and a FPR of 0.2 %. Despite its effectiveness, the model struggles in crowded scenes with overlapping individuals, primarily due to training on frontal-view datasets. This limitation highlights the need for models specifically trained on overhead datasets. Prior approaches using background subtraction or feature-based methods exhibit higher error rates. Thus, further research is needed to enhance detection accuracy in overhead scenarios. Our proposed approach addresses these issues by leveraging YOLOv8, trained on specialized overhead datasets, to improve real-time detection accuracy and robustness in surveillance systems. In [19], YOLOv4 method is employed on improving the speed and accuracy of object detection. It shows that YOLOv4 achieves a mAP of 43.5 % at 65 FPS, offering significant improvements over previous YOLO versions. Although, there are certain challenges such as occlusion or low-light conditions in handling complex environments, which hinders detection performance. The integration of advanced feature aggregation techniques or optimizing for specific tasks can be a way to overcome these ambiguities. Building on these advancements, our suggested method with YOLOv8 further optimizes for energy economy and real-time performance, making it appropriate for sophisticated surveillance applications in smart CCTV systems. Research on the use of YOLOv3 for persons detection in crowd surveillance systems is presented in the publication by [20]. It demonstrates that YOLOv3 successfully detected people even in the presence of partial occlusions, achieving a mAP of 78.3 %. Reduced confidence in detection for persons who are overlapping or distant is one of the outstanding difficulties, though. These difficulties are caused by the intricacy of controlling opacity and crowded scenes. Using sophisticated tracking algorithms and training on a wider range of datasets could be one possible remedy. Even while there are comparable methods, they frequently have drawbacks in congested settings. This emphasizes the need for more study to increase the precision of detection in intricate, real-time monitoring systems.

3. The aim and objectives of the study

The aim of this study is to design and develop a smart, energy-efficient CCTV camera system that can detect humans in real-time, even in challenging environmental conditions, by utilizing the YOLOv8 DL model. The system is intended to enhance security and surveillance by providing a more accurate and responsive solution than traditional CCTV systems.

To achieve this aim, the following objectives are accomplished:

– to design and conceptualize an energy-efficient CCTV system framework utilizing the YOLOv8 model for robust and real-time human detection across varied conditions;

– to optimize YOLOv8 through hyperparameter tuning, adjusting learning rate, batch size, and training epochs to enhance accuracy;

– to implement the YOLOv8 model into the CCTV camera system, leveraging its CSPNet architecture for real-time human detection under various conditions;

– to ensure energy-efficient operation on the NVIDIA Jetson Nano platform, allowing the system to deliver high performance with minimal energy consumption.

4. Materials and methods of research

4. 1. Object and hypothesis of the study

The object of this study is the smart system for real-time human detection. This system integrates the YOLOv8 DL model to ensure high accuracy and responsiveness under diverse environmental conditions. The hypothesis is that the YOLOv8 model, optimized with hyperparameter tuning and implemented on the NVIDIA Jetson Nano platform, can significantly enhance real-time human detection capabilities while maintaining energy efficiency, even under challenging conditions such as poor lighting, cluttered backgrounds, and motion dynamics. The assumptions made in the study are as follows:

1. The PASCAL VOC 2012 dataset is representative of real-world surveillance environments and scenarios.

2. The NVIDIA Jetson Nano has adequate computational power and efficiency for real-time deployment of the YOLOv8 model.

3. The YOLOv8 architecture's CSPNet provides the necessary speed and precision for human detection in continuous surveillance tasks.

The study adopts several simplifications to streamline its scope and ensure focused analysis. It specifically addresses human presence detection without considering complex behaviors, gestures, or activities, thereby narrowing the objective to essential surveillance tasks. Environmental testing scenarios are confined to typical indoor and outdoor conditions, excluding extreme situations such as severe weather or highly dynamic lighting variations. Additionally, the model's hardware setup assumes an ideal configuration, overlooking potential limitations posed by lower-spec systems or alternative hardware platforms. These simplifications help concentrate the research on the core functionality of real-time human detection, ensuring clarity and feasibility in the evaluation process.

4. 2. Equipment and software used

The research employed a combination of advanced hardware and software components to design, train, and test the proposed system. On the hardware side, the NVIDIA Jetson Nano Developer Kit, featuring a quad-core ARM Cortex-A57 CPU, a 128-core Maxwell GPU, and 4GB LPDDR4 memory, was selected for its suitability in edge-based AI applications. For real-time video input, an Intex IT-CAM 09 webcam with a resolution of 0.3 megapixels was utilized, while an Intel Core i5-10400 CPU with 64 GB RAM served as the high-performance computer system for model training and validation tasks. On the software front, Python 3.12.4 was chosen for its rich ecosystem of libraries and tools supporting ML and data processing. TensorFlow 2.17 and Keras provided the frameworks necessary for model training and deployment, complemented by libraries like NumPy and Pandas for data manipulation, OpenCV for video processing, and Matplotlib for performance visualization. The development environment was configured in Jupyter Notebook within a virtual environment, ensuring proper dependency management and a streamlined workflow. Together, these components formed a robust foundation for the research.

4. 3. Methodology and experimental conditions

The research followed a systematic approach to design, optimize, and implement the YOLOv8 model for real-time human detection in a smart CCTV system. The process began with dataset preparation, where the PASCAL VOC 2012 dataset was preprocessed and annotated to ensure compatibility with YOLOv8 training requirements. Object classes were labeled, and configurations for the model architecture were established. The training phase utilized pre-trained weights to accelerate the convergence process, while hyperparameters such as learning rate, batch size, and the number of epochs were systematically adjusted to optimize model performance. Metrics including training and validation losses were monitored across epochs to refine the model and prevent overfitting.

Once trained, the YOLOv8 model was deployed on the NVIDIA Jetson Nano platform, selected for its computational efficiency and energy-saving capabilities. The system was integrated with a webcam to process real-time video feeds, utilizing the YOLOv8 architecture to detect and classify human figures. The deployment process ensured that the system could operate seamlessly in dynamic environments while maintaining high accuracy and responsiveness. Rigorous testing validated the system's performance across multiple scenarios, ensuring its robustness and scalability for real-world applications.

Furthermore, the system was evaluated in both static and dynamic camera setups, simulating stationary and mobile surveillance contexts. Additionally, the robustness of the YOLOv8 model was tested against diverse backgrounds, including cluttered and minimalistic settings, to measure its ability to detect human figures amidst environmental noise. Lastly, energy consumption was analyzed by monitoring the NVIDIA Jetson Nano's power usage in idle and active states, verifying its efficiency and suitability for continuous surveillance tasks. This comprehensive testing framework ensured a reliable assessment of the system's capabilities and limitations.

4. 4. Dataset description

In the present work, PASCAL VOC (Visual Object Classes) dataset is employed to train the YOLOv8 model. This dataset is a widely recognized benchmark in the field of computer vision, designed for evaluating the performance of various object detection, classification, and segmentation algorithms. Sample images from the PASCAL VOC dataset are shown in Fig. 1, illustrating the range of object classes and environmental conditions present.

Fig. 1. Sample images from PASCAL VOC dataset

The dataset comprises a rich collection of images that span multiple object categories, including animals, vehicles, household items, and more, as exemplified in the provided figure. The dataset includes meticulously annotated images, where each object within an image is labeled with a bounding box and associated class, facilitating precise object detection and classification tasks. This diversity makes the PASCAL VOC dataset particularly valuable for developing and testing algorithms that must perform well across varied real-world scenarios.

5. Results of the analysis of smart energy-efficient CCTV system for real-time human detection

5. 1. Proposed YOLOv8 model and its architecture

The implementation of YOLOv8 in CCTV camera systems for human detection brings several key advantages. YOLOv8's real-time processing capability allows the system to analyze video streams and detect human presence instantaneously, which is crucial for prompt responses in security situations. Its advanced DL architecture ensures high accuracy, reliably identifying humans in various poses and lighting conditions [21, 22]. Additionally, YOLOv8's efficiency in handling high-resolution images and its ability to run on platforms like NVIDIA Jetson Nano make it an ideal choice for enhancing the performance of CCTV surveillance systems. Fig. 2 illustrates a real-time human detection system designed for unmanned security surveillance.

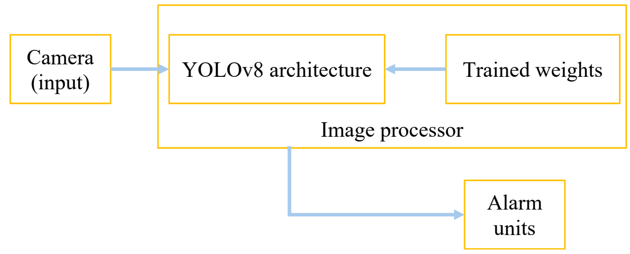


Fig. 2. Real time architecture of human detection system

This system continuously monitors a designated area and automatically generates an alert if a human presence is detected in the images. A web camera captures real-time video at a rate of 30 frames per second, which is then converted into individual images. Each frame undergoes testing using the YOLOv8 architecture implemented on the Jetson Nano platform. The YOLOv8 algorithm, along with its trained weights, is utilized for image classification to distinguish between human and non-human entities. The NVIDIA Jetson Nano is capable of efficiently running the trained YOLOv8 network, ensuring rapid and accurate detection of humans in the surveillance area.

Fig. 3 presents the flowchart illustrating the process of training and deploying the YOLOv8 model for object detection, specifically focusing on detecting humans and calculating relevant metrics in real-time. Firstly, the model is trained on the PASCAL VOC dataset, a well-known benchmark in computer vision that includes 20 distinct classes, such as person, bird, cow, dog, horse, sheep, airplane, bicycle, boat, bus, car, motorbike, train, bottle, chair, dining table, potted plant, sofa, TV, and monitor. Afterwards, the process begins with the preparation of the training dataset and object labels, which are categorized into different objects each associated with its corresponding label. These labeled objects are then fed into the YOLOv8 training model, which is configured using a predefined configuration file that specifies the architecture layers of the YOLOv8 model. In addition to the labeled dataset, the model utilizes pre-trained convolutional weights to accelerate the training process and enhance performance. These weights are typically obtained from a model that has already been trained on a large dataset and can provide a solid foundation for further training.

During the training process, the model is optimized to reduce errors and enhance its ability to detect objects. It goes through the initial data and generates a trained weight file, which contains the features it has learned from the dataset. This trained weight file is then used for real-time image processing, where the model analyzes live camera feeds. The model detects humans (or any other specified objects) in the camera feed and performs the necessary calculations or tasks based on what it identifies. After training on this dataset, the model produces a weight file, which can later be used to test its accuracy and performance. The flowchart below provides an overview of the steps taken to train the YOLO model on a dataset, optimize it, and deploy it for real-time object detection tasks.

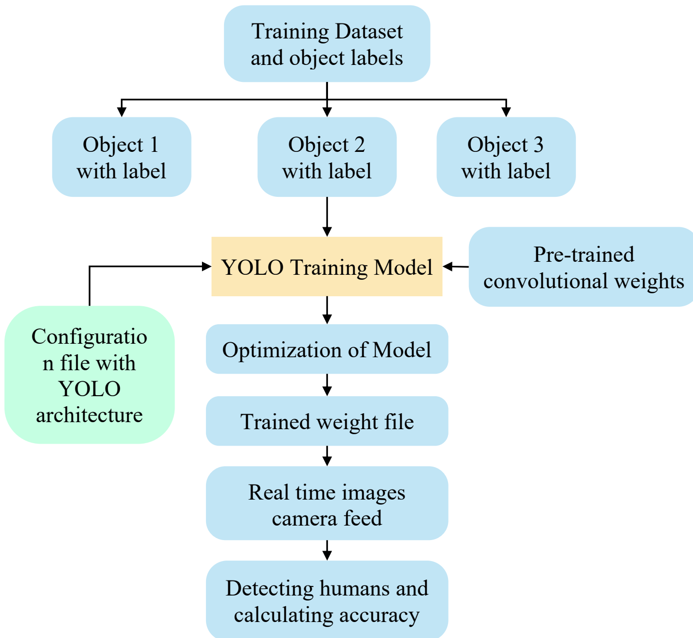


Fig. 3. Training and deployment of YOLOv8 model

5. 2. Optimization of YOLOv8 model

Hyperparameter tuning is a crucial process where parameters and settings are carefully selected and adjusted to improve a model's performance, particularly when training an object detection model. In this study, the model was fine-tuned by varying the number of epochs and other hyperparameters, as outlined in Table 1. The project explored how changing the number of training epochs affects the YOLOv8 model's performance, focusing on training loss, validation loss, and test accuracy. As the number of epochs increases, both training and validation losses decrease significantly, indicating that the model is learning more effectively and reducing errors. Specifically, after 25 epochs, the model achieved its lowest training and validation losses, with values of 13.78 and 12.94, respectively, and reached a high testing accuracy of 87 %. These findings highlight the importance of an adequate training duration, as more epochs can help refine and enhance the model's accuracy. However, it's essential to strike a balance to avoid overfitting, ensuring the model remains robust and performs well on new, unseen data.

Table 1

Optimizing parameters of YOLOv8 model

Number of epochs	Training loss	Validation loss	Test accuracy of YOLOv8 (%)	Hyperparameters	
10 Epochs	14.65	14.39	72	Patience	50
15 Epochs	14.16	13.82	73	Batch	16
25 Epochs	13.78	12.94	87	Workers	8
				Momentum	0.937
				Weight decay	0.001

Table 2 compares different YOLO versions, focusing on their performance in human detection tasks using metrics like mean Average Precision (mAP) and frames per second (FPS). The YOLOv8 model, trained on the PASCAL VOC dataset, outperforms earlier versions such as YOLOv7, YOLOv3, and YOLOv2, achieving the highest mAP of 79 %, which indicates superior accuracy in object detection. Although YOLOv8's FPS of 39.37 is slightly lower than that of YOLOv7 and YOLOv2, it still strikes a strong balance between precision and speed, making it highly effective for real-time applications.

Table 2

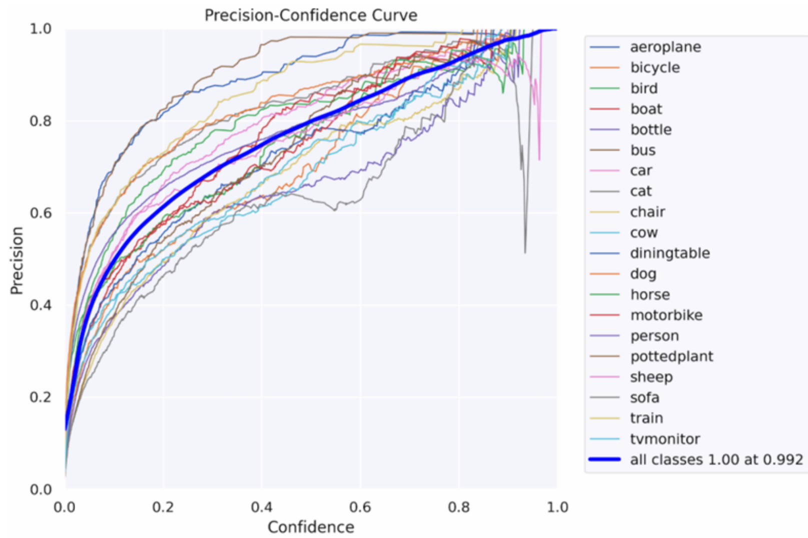
Comparison of YOLO versions on mAP and FPS performance

Model	Backbone	Dataset	mAP(%)	FPS
YOLOv8 (Present work)	CPSnet	PASCAL VOC	79	39.37
YOLOv7 [23]	Darknet	PASCAL VOC	75	55.38
YOLOv5 [24]	New CPS-Darknet-53	MS COCO	55	38
YOLOv4 [20]	CPS Darknet-53	MS COCO	41.2	38
YOLOv3 [25]	Darknet-53	INRIA	70.5	47.22
YOLOv2 [26]	Darknet-19	PASCAL VOC	71	67

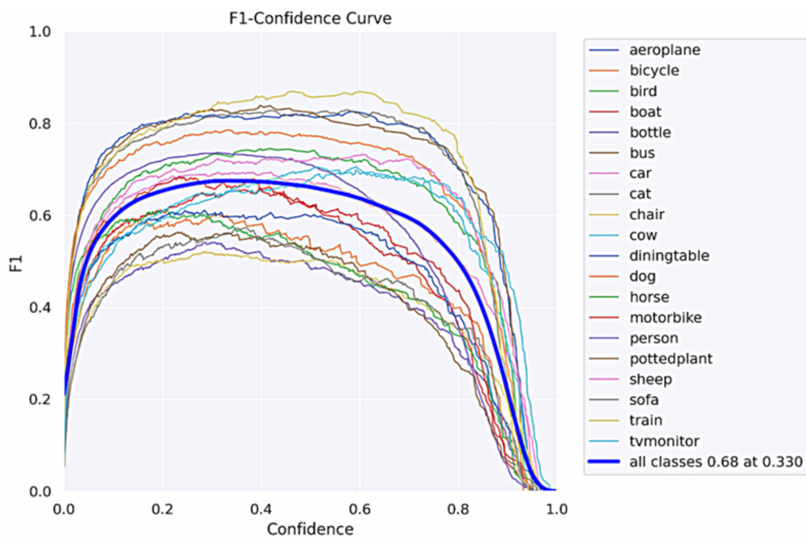
The results highlight YOLOv8's ability to enhance object detection tasks, particularly human detection, by offering improved accuracy while maintaining sufficient processing speed for real-time performance. This balance makes YOLOv8 a robust choice for scenarios where both high precision and timely detection are critical. The combination of a high mAP and reasonable FPS underscores the advancements YOLOv8 brings to the field, effectively bridging the gap between accuracy and efficiency in real-world applications. The table shows how YOLOv8, with its CSPnet architecture, provides significant improvements over previous YOLO versions, solidifying its role in advancing the capabilities of object detection models.

5. 3. Performance evaluation of the proposed YOLOv8 Model

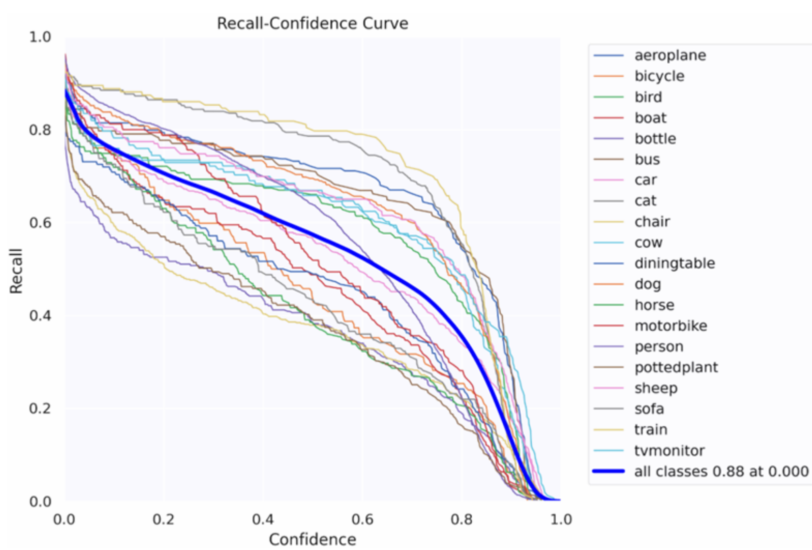
Fig. 4 illustrates the relationship between precision, recall, and F1 score at different confidence levels for various object classes in the PASCAL VOC dataset, as evaluated by the YOLOv8 model. The figure shows both the individual performance of each object class and an aggregated view across all classes. As shown in Fig. 4, *a*, precision generally improves as confidence increases, reaching near-perfect levels at high confidence for most classes. The blue line in the aggregated curve indicates that, on average, the model achieves high precision across all classes at these higher confidence thresholds. Fig. 4, *b* shows the F1 score, which combines precision and recall, peaking at moderate confidence levels. This peak represents a balance between precision and recall before the F1 score decreases as confidence rises further. This decline at higher confidence reflects the typical trade-off where increasing precision often leads to reduced recall. Fig. 4, *c* illustrates the recall curve, which initially rises with confidence but eventually levels off and declines as confidence continues to increase. This pattern suggests a trade-off, with higher confidence improving precision but lowering the model's ability to detect all relevant objects (recall). Overall, these results highlight how the YOLOv8 model's performance varies with different confidence levels across object classes. The model shows strong precision at higher confidence thresholds, but this comes at the expense of recall. The F1 score effectively captures this balance, providing valuable insights into optimizing confidence thresholds for specific applications to achieve the desired trade-off between precision and recall.



a



b



c

Fig. 4. Variation of the performance parameters with confidence for YOLOv8 model across PASCAL VOC object classes: *a* – precision; *b* – recall; *c* – F1 score

Fig. 5 shows the Precision-Recall (PR) curves for each object class in the PASCAL VOC dataset, as evaluated by the YOLOv8 model. The figure highlights the relationship between precision and recall for each class, with the (mAP at a 0.5 Intersection over Union (IoU) threshold also provided. The mAP values vary from 0.473 for the bottle class to 0.883 for the train class, while the overall performance across all classes results in an mAP@0.5 of 0.697. These results indicate significant variation in detection performance depending on the object class, which can be attributed to challenges like object size, shape, and occlusion. While the aggregated curve shows that the YOLOv8 model performs well on average, it also reveals areas for improvement, particularly in classes with lower mAP scores. These findings are essential for guiding future enhancements to the model, with the goal of improving detection accuracy and robustness across all object classes in varied environments.

The confusion matrix offers a clear picture of where the YOLOv8 model excels and where it faces challenges in object detection. Fig. 6 showcases this matrix, giving a detailed breakdown of how well the model classifies different object classes. The model shows strong accuracy for several classes, such as aeroplane (0.77), bus (0.76), cat (0.82), and train (0.86), where most predictions align correctly along the diagonal. These high accuracy rates indicate that the model is quite effective at identifying these specific objects. However, the model struggles more with classes like chair (0.47) and bottle (0.51), where correct predictions are lower, and there is more confusion with other classes. Additionally, the model occasionally misclassifies the background (which represents the absence of any target object) as certain objects like aeroplanes, bicycles, and bottles, suggesting that further refinement is needed to reduce these errors. Overall, while the YOLOv8 model performs well in recognizing most classes, the areas of misclassification underscore the importance of continued optimization to improve accuracy, especially for those classes that are more difficult to classify correctly and those that are often confused with the background.

Fig. 7 illustrates the training and validation loss curves for the YOLOv8 model across four different training durations: 10, 15, 25, and 30 epochs. In Fig. 7, *a*, the training loss curves show an initial rise followed by a decline, reflecting the model's learning process as it gradually reduces errors while processing more data. Notably, the model trained for 25 epochs shows a steady and stable decrease in training loss, indicating effective learning and optimization during this period. Similarly, Fig. 7, *b* displays the validation loss, which follows a comparable pattern with an initial increase and subsequent decrease. The validation loss for the model trained over 25 epochs also steadily declines, suggesting that the model is generalizing well to unseen data. These findings highlight the importance of choosing the right number of training epochs. While more training can improve performance, as demonstrated by the 25-epoch model, training for too long (such as 30 epochs) can lead to overfitting. This is where the model becomes too tailored to the training data, leading to a decrease in its ability to perform well on new data, as seen in the increase in both training and validation loss.

Fig. 8 illustrates how precision-recall curves and learning rates are influenced by different numbers of training epochs. In Fig. 8, *a*, the precision-recall graph shows how the model balances precision and recall, with variations observed as the number of epochs changes. This indicates that the model's ability to maintain a balance between precision and recall is impacted by how long it is trained. Fig. 8, *b* presents the relationship between learning rate and loss, demonstrating how different learning rates affect the model's performance over the training periods.

Results showed that learning rates tend to decrease as the number of epochs increases, which shows how the model adjusts its learning speed to more effectively minimize the loss function. This behavior suggests that the model's ability to converge and generalize well is closely tied to the choice of learning rate and the duration of training.

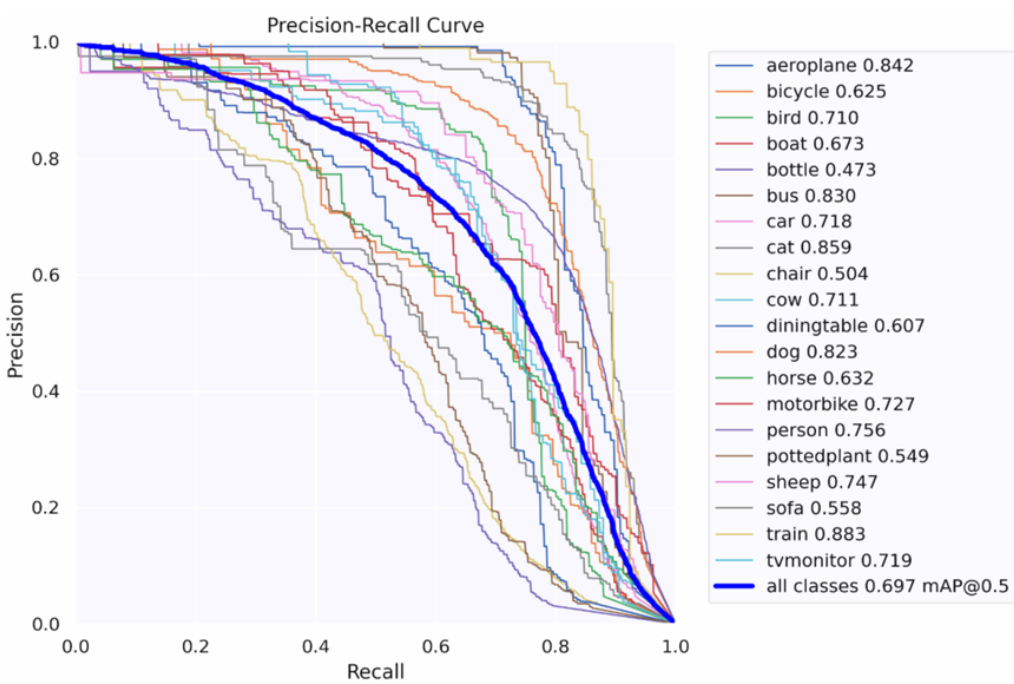


Fig. 5. Precision-recall curve for YOLOv8 model across distinct classes

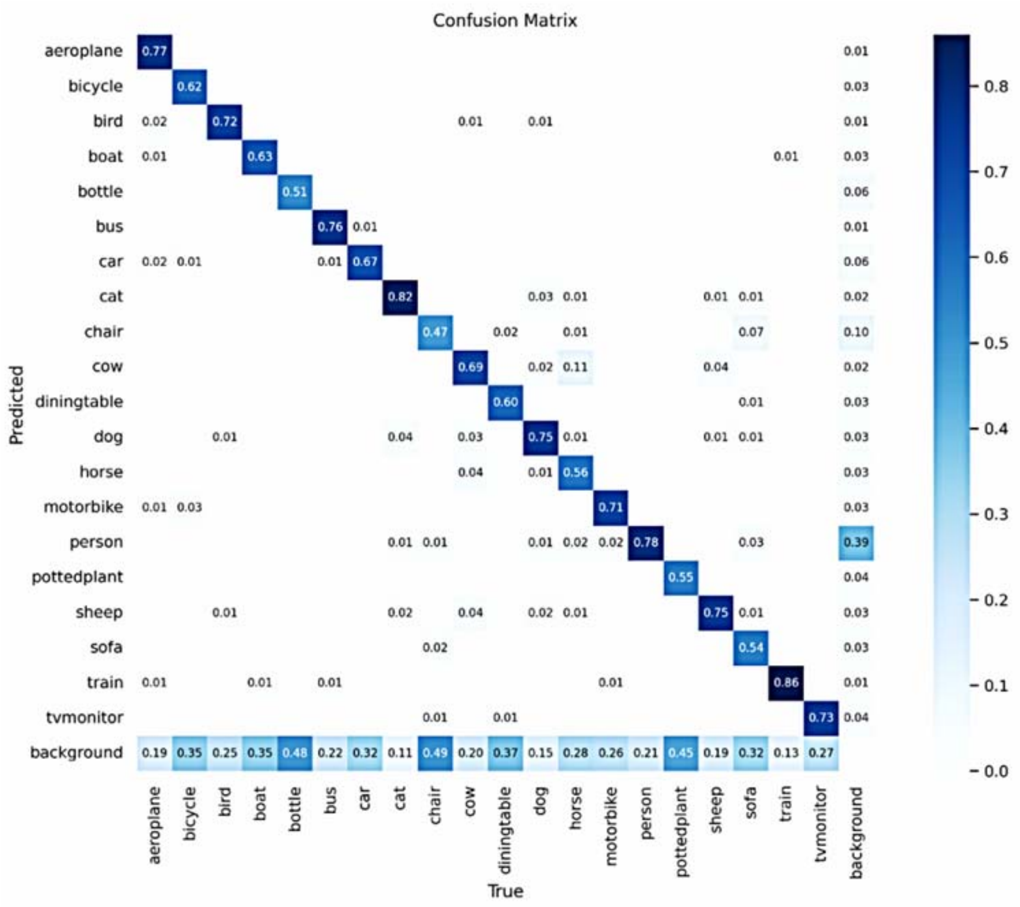
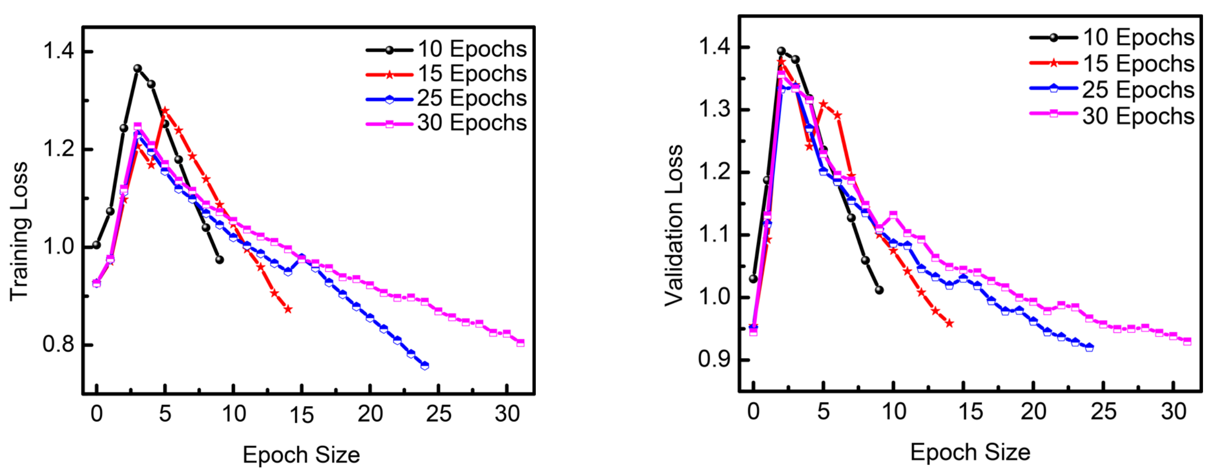


Fig. 6. Confusion matrix obtained for the YOLOv8 model



a *b*

Fig. 7. Loss curve for YOLOv8 across different epochs: *a* – training; *b* – validation loss curves

5. 4. Real time human detection system using Jetson Nano NVIDIA system

The Jetson Nano NVIDIA system is an affordable yet powerful embedded computer, capable of running advanced neural networks. It fully supports popular DL and ML frameworks like PyTorch, Caffe, Keras, and TensorFlow. Equipped with TensorRT accelerator libraries as part of the Jetpack packages, the Jetson Nano is ideal for real-time applications across various scenarios, and it can process multiple high-definition video streams at once.

The device is powered by a QUAD-core ARM A57 CPU running at 1.43 GHz and features a 128-core Maxwell GPU. It also includes 4 GB of 64-bit LPDDR4 memory with a bandwidth of 25.6 GB/s. For connectivity, it offers a USB 2.0 Micro-B port and 4×USB 3.0 ports. In our experiments, let's connect a standard 0.3-megapixel camera module to the Jetson Nano's camera serial interface (CSI). Let's deploy a trained YOLOv8 model and a human detection algorithm, enabling the Jetson Nano to function as a standalone application. The device's power consumption measurements are detailed in Table 3.

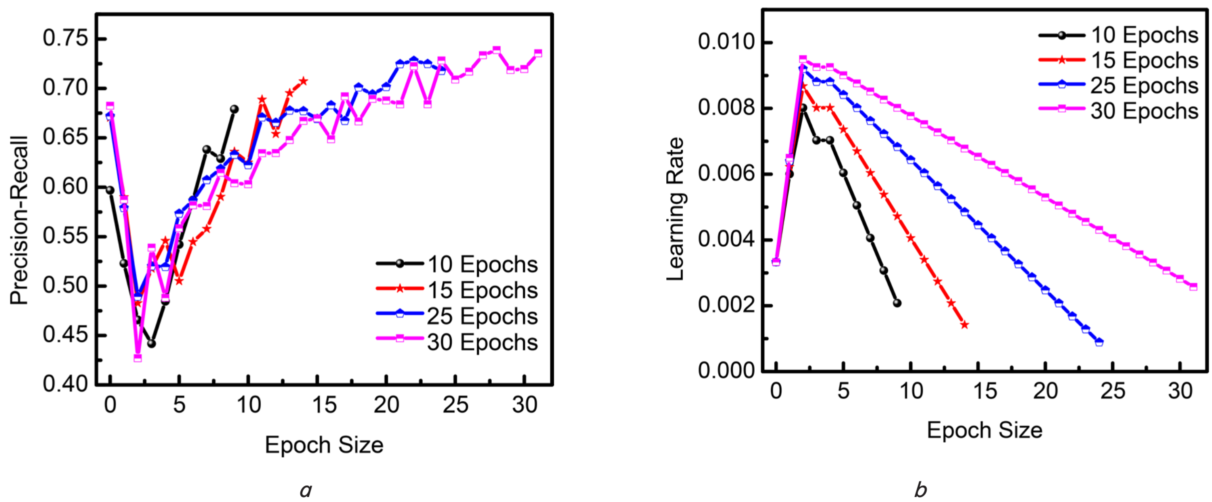


Fig. 8. Curves for YOLOv8 across different epochs: *a* — Precision-recall; *b* — Learning rate

Table 3

Power consumption measurement of Jetson Nano in different scenarios

System specifications	Algorithm position	Power measurement (W)
Standalone Jetson Nano with Web camera	Off	1.22
	Working	4.38
Jetson Nano with monitor, keyboard, mouse	Off	2.24
	Working	5.40

The algorithm's performance is influenced by both internal and external factors, including whether the camera is fixed or mobile. The method analyzes specific frames from the webcam's video feed, and ideally, the camera's placement should not significantly impact its effectiveness in detecting human presence. However, detecting stationary individuals can be challenging, and varying densities of people in different frames may lead to inconsistencies in detection. Non-human objects can also obscure human subjects, causing false positives and reducing accuracy. The algorithm's response time is affected by the specific algorithm used and the computational setup. Detection rate and accuracy are key performance metrics, as shown in Table 4 and are evaluated using following formula:

$$\text{Detection Rate} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}, \quad (1)$$

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{True Positives} + \text{True Negatives} + \text{False Positives} + \text{False Negatives}}. \quad (2)$$

Table 4

Impact of camera position on detection rate and accuracy in human detection algorithm

Camera position	Detection rate	Accuracy
Fixed frame	0.97	0.79
Moving frame	0.95	0.68

The experimental setup consists of the Jetson Nano developer kit, an Intex IT-CAM 09 Webcam (0.3MP), and an LCD monitor and is displayed in Fig. 9. The webcam captures real-time video of the laboratory environment at a resolution of 640x480, which is sufficient for recording nearby objects. The video feed is processed by the high-performance NVIDIA Jetson Nano, which operates at the edge, allowing for immediate inference and decision-making. This setup enables real-time monitoring, with the processed data being used directly for automated decisions.

The Jetson Nano is a powerful development kit designed to run multiple neural network algorithms simultaneously, making it ideal for a range of applications. It boasts superior processing capabilities compared to the Raspberry Pi, thanks to its more powerful GPU. While the Jetson Nano is more compact and slightly slower than the Jetson TX2, it consumes less power (5–10 W) and is more cost-effective. It efficiently runs the trained YOLOv8 algorithm, which is used to detect and classify objects in the video feed as either human or non-human. The object detection process involves two stages: feature extraction and feature classification, with YOLOv8 handling both tasks effectively.

This proposal introduces a real-time object detection system using the YOLOv8 algorithm integrated with the NVIDIA Jetson Nano platform. The YOLOv8 algorithm is well-known for its speed and accuracy in object detection and is employed here to identify and classify objects in video feeds, with a particular focus on detecting human figures. The results from this experimental setup are shown in the accompanying images.

Fig. 10 illustrates the system's ability to detect humans in both outdoor and indoor environments. In Fig. 10, *a*, the system successfully identifies a person walking outdoors with a confidence score of 0.86, demonstrating its effectiveness even in outdoor settings where lighting and background complexity can vary widely. Fig. 10, *b* shows the detection of a person in a gym, where the algorithm assigns a confidence score of 0.87 to the human figure. This result highlights the system's robustness in indoor environments, where factors like lighting and object proximity differ from that outdoors. Another complex scenario is depicted in Fig. 10, *c*, and Fig. 10, *d*, this time in the foggy and rainy environments, where the system detects a person with a confidence score of 0.83 and 0.86, respectively. This further emphasizes the algorithm's adaptability to poor lightning and complex backgrounds and its ability to maintain high detection accuracy across different contexts. These experimental results demonstrate the effectiveness of the YOLOv8 algorithm when deployed on a Jetson Nano for real-time object detection. The system's ability to operate reliably in diverse environments and consistently detect human figures with high confidence underscores its potential for applications in surveillance, security, and other areas where real-time monitoring is essential. The integration of feature extraction and classification within the YOLOv8 pipeline ensures not only that objects are detected but that this detection is carried out with a high degree of accuracy and reliability.

Fig. 11 showcases the results of real-time human detection using the YOLOv8 algorithm on an NVIDIA Jetson Nano platform. In Fig. 11, *a*, the results are displayed for a scenario where the camera is stationary. In this setup, the algorithm successfully identifies two individuals with confidence scores of 0.65 and 0.79. It also detects a dining table with a confidence score of 0.67 and a bottle with a confidence score of 0.73. The fixed frame provides a stable environment, allowing the algorithm to maintain consistent accuracy as the objects remain clearly visible and unobstructed. In Fig. 11, *b*, the results from a moving frame scenario are shown, where either the camera or the scene is in motion. Despite the added challenges of movement, the algorithm continues to detect multiple objects, including chairs, a potted plant, and two individuals. The chairs are detected with confidence levels ranging from 0.34 to 0.84, reflecting variations in visibility and orientation due to motion. The potted plant is identified with a high confidence score of 0.80, while the human figures are detected with confidence scores around 0.68. This moving frame scenario tests the algorithm's robustness in dynamic environments, where objects might enter or exit the frame or appear at different angles.

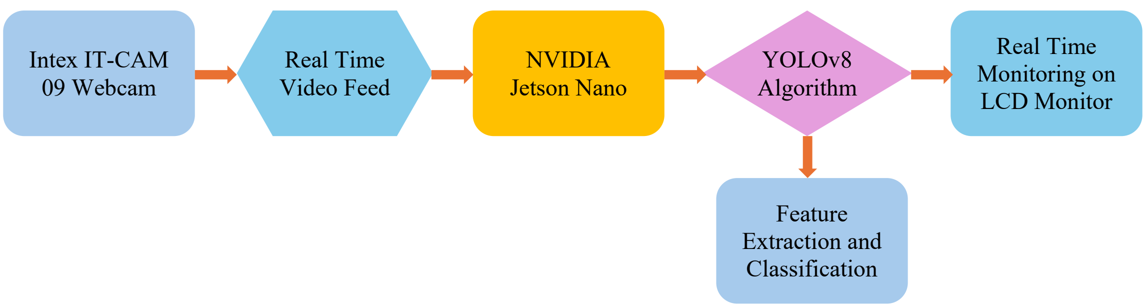
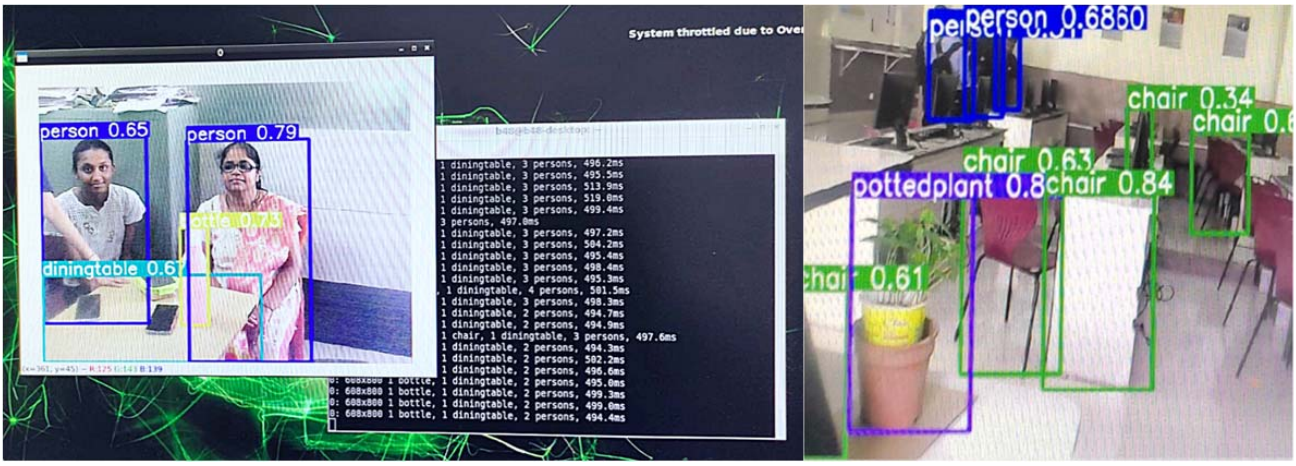


Fig. 9. Flowchart representing the experimental setup

Fig. 10. Human detection: *a* – outdoor environment; *b* – indoor environment; *c* – foggy environment; *d* – rainy environment



a *b*

Fig. 11. Real time human detection: *a* — fixed frame; *b* — moving frame

These results demonstrate the YOLOv8 algorithm's effectiveness in performing real-time human detection and object classification in both static and dynamic scenarios. The system's ability to maintain consistent accuracy, even when faced with the challenges of a moving frame, is particularly important for applications like surveillance, where environments are often in flux and real-time detection is critical. The algorithm's strong performance in both fixed and moving frames highlights its potential for reliable and efficient object detection across a variety of real-world settings.

6. Discussion of real-time human detection using the proposed YOLOv8 model in CCTV cameras

The results obtained in this study can be explained by the combination of advanced DL algorithms and optimized hardware implementations. The YOLOv8 model's integration with an energy-efficient NVIDIA Jetson Nano platform demonstrates a highly effective system for real-time human detection in surveillance applications. The system's advanced architecture, enables continuous processing of video streams, achieving accurate detection of human figures in various poses and lighting conditions (Fig. 2). This highlights its adaptability, overcoming challenges like poor lighting and cluttered backgrounds, addressing key limitations of traditional surveillance systems. Furthermore, the streamlined workflow emphasizes the systematic approach, from data preparation to deployment, ensuring optimal performance under diverse surveillance scenarios (Fig. 3). The combination of advanced model architecture and efficient hardware integration underscores the system's potential for transforming real-time monitoring applications.

The model's performance is improved significantly with hyperparameter tuning (Table 1), where adjusting the number of epochs yields lower training and validation losses, with an optimal configuration observed at 25 epochs. This tuning enhances the model's overall accuracy to 87 %, confirming that targeted adjustments to hyperparameters can significantly improve performance while avoiding overfitting. These improvements are attributed to the efficient design of the YOLOv8 architecture, which processes images in a single pass, resulting in faster detection with fewer computational resources. Additionally, the use of the PASCAL VOC 2012 dataset allowed for robust training in diverse conditions, enhancing the model's adaptability to various surveillance environments. When comparing the results of this study with existing methods (Table 2), it is clear that the proposed system offers several improvements. The YOLOv8 model implemented in this study, with a mAP of 79 %, outperforms earlier versions like YOLOv3, which achieved 70.5 % mAP with slightly faster frame rates (47.22 FPS). This demonstrates a better balance between precision and speed, as the current study maintains high accuracy while operating at a competitive FPS (39.37 FPS).

Furthermore, the performance of the model is evaluated by varying the performance metrics such as precision, recall, and F1 score across different confidence levels (Fig. 4). At higher confidence levels, the precision increases sharply, which is critical for high-stakes surveillance settings where false positives need to be minimized. However, the recall curve shows a slight decline as confidence increases, reflecting the typical trade-off where improved precision comes at the expense of detecting all relevant instances. The Precision-recall curves further delineate the YOLOv8 model's strengths and limitations across specific object classes (Fig. 5). Higher mAP values for classes like "train" and lower for "bottle" indicate that object size, shape, and distinctiveness in visual data impact detection performance. The confusion matrix highlights instances of misclassification, particularly with objects that share visual similarities or occur frequently as background elements (Fig. 6). These misclassifications suggest areas for refinement to increase detection reliability. The training and validation loss curves (Fig. 7) demonstrate that 25 epochs yield optimal results, with both losses stabilizing, ensuring effective learning and robustness in the model. The different loss trends observed across various training configurations underscore the importance of selecting the right training parameters, such as the number of epochs and learning rates, to achieve a good balance between precision and recall while ensuring the model converges effectively (Fig. 8).

Moreover, the energy-efficient implementation on the Jetson Nano platform (Table 3) makes this system ideal for continuous surveillance applications, where power consumption is a critical factor. The YOLOv8 system demonstrates its robustness in detecting human figures across varied settings, achieving high confidence scores of 0.86 in outdoor environments and 0.87 indoors (Fig. 10). This highlights the model's adaptability, accurately identifying human presence despite differences in lighting and background complexity. To further supports these results, the system's performance in both fixed and moving frames has been investigated (Fig. 11). In the stationary setup, human figures and surrounding objects are identified with consistent accuracy, achieving confidence scores up to 0.79. Even in dynamic frames, where movement poses additional challenges, the model reliably detects multiple objects, maintaining stable detection scores. The obtained inferences suggests that YOLOv8's capability to deliver real-time, high-accuracy human detection in complex environments, validating its potential as a dependable solution for modern CCTV surveillance needs. Overall, this discussion underscores that the YOLOv8 model, with appropriate tuning and the NVIDIA Jetson Nano platform, effectively addresses challenges in real-time human detection, offering a reliable, efficient solution adaptable to dynamic environments, yet suggesting further optimizations for higher resilience in complex scenarios.

7. Conclusions

1. The implementation of the YOLOv8 model into the proposed system has significantly enhanced its ability to deliver high-accuracy, real-time human detection in various environmental conditions. By employing its advanced CSPNet architecture and training on the diverse PASCAL VOC dataset, the system achieved consistent detection accuracy and responsiveness. This implementation ensures effective performance, addressing the key challenges of traditional surveillance systems, such as adaptability to poor lighting and complex backgrounds, while maintaining energy efficiency.

2. The optimization of hyperparameters significantly enhanced the system's performance. After tuning the YOLOv8 model through multiple epochs, the system achieved training loss of 13.78 and validation loss of 12.94 after 25 epochs, alongside test accuracy of 87 %. These results indicate a well-calibrated model capable of effectively learning from data while minimizing errors. The steady reduction in losses over time highlights the impact of a carefully structured training process, allowing the system to perform reliably in varied scenarios. This refinement distinguishes the system from less optimized models, offering a better balance between detection accuracy and efficiency.

3. The integration of the YOLOv8 model into the system provided the necessary real-time human detection capabilities. The system demonstrated a mAP of 79 % and operated at 39.37 FPS, showcasing its effectiveness in terms of speed and accuracy. These results reflect the system's ability to quickly and reliably identify human figures in both indoor and outdoor environments, surpassing traditional methods that lacked such real-time capabilities. The high precision is explained by YOLOv8's CSPNet architecture, which optimizes detection without sacrificing speed.

4. Testing under non-ideal conditions, such as varying lighting, background complexity, and motion, demonstrated the robustness of the system. The system maintained high detection accuracy, with confidence scores ranging from 0.65 to 0.87 across different settings. These results emphasize the adaptability of the system in real-world environments, where conditions are rarely optimal. The success in these challenging conditions can be attributed to the YOLOv8 model's advanced object detection algorithms, which handle noise, low-resolution inputs, and environmental variability more effectively than traditional CCTV systems.

In conclusion, the developed smart CCTV system not only met but exceeded the initial objectives by delivering a highly accurate, real-time human detection solution. Its energy-efficient design and adaptability in non-ideal conditions offer a significant advancement over existing surveillance technologies, making it a robust and scalable solution for modern security needs.

Conflict of interest

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.

Financing

The author received no funding support from any agencies or firms.

Data availability

The corresponding author can provide the data from this study upon request.

Use of artificial intelligence statement

The authors confirm that they did not use artificial intelligence technologies when creating the current work.

References

1. Sreenu, G., Saleem Durai, M. A. (2019). Intelligent video surveillance: a review through deep learning techniques for crowd analysis. Journal of Big Data, 6 (1). https://doi.org/10.1186/s40537-019-0212-5
2. Dimou, A., Medentzidou, P., Garcia, F. A., Daras, P. (2016). Multi-target detection in CCTV footage for tracking applications using deep learning techniques. 2016 IEEE International Conference on Image Processing (ICIP), 928–932. https://doi.org/10.1109/icip.2016.7532493
3. Amrutha, C. V., Jyotsna, C., Amudha, J. (2020). Deep Learning Approach for Suspicious Activity Detection from Surveillance Video. 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), 335–339. https://doi.org/10.1109/icimia48430.2020.9074920
4. Ragedhaksha, Darshini, Shahil, Arunnehru, J. (2023). Deep learning-based real-world object detection and improved anomaly detection for surveillance videos. Materials Today: Proceedings, 80, 2911–2916. https://doi.org/10.1016/j.matpr.2021.07.064

5. Piza, E. L., Caplan, J. M., Kennedy, L. W. (2013). Analyzing the Influence of Micro-Level Factors on CCTV Camera Effect. Journal of Quantitative Criminology, 30 (2), 237–264. https://doi.org/10.1007/s10940-013-9202-5
6. Sathyadevan, S., Balakrishnan, A. K., Arya, S., Athira Raghunath, S. (2014). Identifying moving bodies from CCTV videos using machine learning techniques. 2014 First International Conference on Networks & Soft Computing (ICNSC2014), 151–157. https://doi.org/10.1109/cnsc.2014.6906721
7. Gelana, F., Yadav, A. (2018). Firearm Detection from Surveillance Cameras Using Image Processing and Machine Learning Techniques. Smart Innovations in Communication and Computational Sciences, 25–34. https://doi.org/10.1007/978-981-13-2414-7_3
8. Anishchenko, L. (2018). Machine learning in video surveillance for fall detection. 2018 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT), 99–102. https://doi.org/10.1109/usbereit.2018.8384560
9. Bhatti, M. T., Khan, M. G., Aslam, M., Fiaz, M. J. (2021). Weapon Detection in Real-Time CCTV Videos Using Deep Learning. IEEE Access, 9, 34366–34382. https://doi.org/10.1109/access.2021.3059170
10. Lestari, D. P., Kosasih, R., Handhika, T., Murni, Sari, I., Fahrurozi, A. (2019). Fire Hotspots Detection System on CCTV Videos Using You Only Look Once (YOLO) Method and Tiny YOLO Model for High Buildings Evacuation. 2019 2nd International Conference of Computer and Informatics Engineering (IC2IE), 87–92. https://doi.org/10.1109/ic2ie47452.2019.8940842
11. Pan, S.-H., Wang, S.-C. (2021). Identifying Vehicles Dynamically on Freeway CCTV Images through the YOLO Deep Learning Model. Sensors and Materials, 33 (5), 1517. https://doi.org/10.18494/sam.2021.3236
12. Pillai, M. S., Chaudhary, G., Khari, M., Crespo, R. G. (2021). Real-time image enhancement for an automatic automobile accident detection through CCTV using deep learning. Soft Computing, 25 (18), 11929–11940. https://doi.org/10.1007/s00500-021-05576-w
13. Rizk, M., Bayad, I. (2023). Human Detection in Thermal Images Using YOLOv8 for Search and Rescue Missions. 2023 Seventh International Conference on Advances in Biomedical Engineering (ICABME), 210–215. https://doi.org/10.1109/icabme59496.2023.10293139
14. Shinde, S., Kothari, A., Gupta, V. (2018). YOLO based Human Action Recognition and Localization. Procedia Computer Science, 133, 831–838. https://doi.org/10.1016/j.procs.2018.07.112
15. Lu, J., Yan, W. Q., Nguyen, M. (2018). Human Behaviour Recognition Using Deep Learning. 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 1–6. https://doi.org/10.1109/avss.2018.8639413
16. Ghosh, S., Das, D. (2019). Comparative Analysis and Implementation of Different Human Detection Techniques. 2019 Fifth International Conference on Image Information Processing (ICIIP), 443–447. https://doi.org/10.1109/iciip47207.2019.8985977
17. Ivašić-Kos, M., Krišto, M., Pobar, M. (2019). Human Detection in Thermal Imaging Using YOLO. Proceedings of the 2019 5th International Conference on Computer and Technology Applications. https://doi.org/10.1145/3323933.3324076
18. Ahmad, M., Ahmed, I., Adnan, A. (2019). Overhead View Person Detection Using YOLO. 2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), 0627–0633. https://doi.org/10.1109/uemcon47517.2019.8992980
19. Bochkovskiy, A., Wang, C.-Y., Liao, H.-Y. M. (2020). YOLOv4: optimal speed and accuracy of object detection. Computer Vision and Pattern Recognition. https://doi.org/10.48550/arXiv.2004.10934
20. 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). https://doi.org/10.1109/iccsce50387.2020.9204925
21. Linder, T., Pfeiffer, K. Y., Vaskevicius, N., Schirmer, R., Arras, K. O. (2020). Accurate detection and 3D localization of humans using a novel YOLO-based RGB-D fusion approach and synthetic training data. 2020 IEEE International Conference on Robotics and Automation (ICRA), 1000–1006. https://doi.org/10.1109/icra40945.2020.9196899
22. Baig, M. A., Pasha, M. A., Masud, S. (2020). An Optimized Hardware/Software Co-Design Framework for Real-Time Pedestrian Detection. 2020 International Conference on Electrical, Communication, and Computer Engineering (ICECCE), 1–6. https://doi.org/10.1109/icecce49384.2020.9179465
23. Jin, F., Zhang, Z., Ning, Y., Lu, Y., Song, W., Qin, X., Chen, J. (2024). Crowd Counting and People Density Detection: An Overview. Proceedings of the 2024 3rd International Conference on Engineering Management and Information Science (EMIS 2024), 434–454. https://doi.org/10.2991/978-94-6463-447-1_46
24. ultralytics/yolov5. Available at: https://github.com/ultralytics/yolov5
25. Gündüz, M. Ş., Işık, G. (2023). A new YOLO-based method for real-time crowd detection from video and performance analysis of YOLO models. Journal of Real-Time Image Processing, 20 (1). https://doi.org/10.1007/s11554-023-01276-w
26. Redmon, J., Farhadi, A. (2017). YOLO9000: Better, Faster, Stronger. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 6517–6525. https://doi.org/10.1109/cvpr.2017.690