iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** V **Month of publication:** May 2026

DOI: https://doi.org/10.22214/ijraset.2026.82445

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

YOLOv8-Based Object Detection System

Dhaval Sonawane[1], Yash Solankar[2], Prashik Wankar[3]
Department of Electronics and Telecommunication, Vishwakarma Institute of Technology Pune, India

Abstract: Automated object detection has been a significant component in intelligent monitoring systems in recent years. Monitoring of objects manually is time-consuming and could give rise to various errors. Therefore, an automated system is supposed to recognize and analyze objects correctly. The object detection system proposed in this project is a Machine Learning and Computer Vision based system. This system is aimed at identifying four major categories including: cars, JCB machines, crowd, and potholes, making it possible to monitor the areas of the population, construction, and other infrastructures. The materials were gathered as annotated datasets on an open-source site like Roboflow and Kaggle and enhanced by methods like flipping, rotation, and mosaic augmentation. YOLOv8 is an improved object detector compared to the older models such as the YOLOv3 and Faster R-CNN, hence it is more accelerated and precise. Upon training and testing, the models had an average Average Precision (mAP at 0.5) of up to 87.2% on crowd detection and 83.8% on vehicle and machinery detection with regard to the detection of the designated objects.
Keywords: Object Detection, Machine Learning, Computer Vision, YOLOv3, YOLOv8, R-CNN.

I. INTRODUCTION

The growth of urbanization and infrastructure has increased the number of cars on the roads which has created traffic jams and had led to an increased amount of construction projects and increased crowds in public places. In order to keep people safe and handle traffic well, we need to check up on cars and construction sites, crowds and the condition of the road regularly. Traditional systems are based on surveillance from CCTV cameras and viewing on site. This takes a long time, and may be delayed. It is difficult to watch large areas in real-time using these methods. In the past, computer vision mainly relied on simple image processing techniques and manually designed features. These methods often had difficulty working in low lighting conditions, with objects of different sizes, or in complex and crowded backgrounds. To address these issues, this work introduces a deep learning object detection system which can identify various objects with the help of advanced architecture YOLOv8. YOLOv8 is the one stage detector that gets up to 150 fps with a very good accuracy of 52.7 on standard tests. Many tests prove YOLO models to be faster and observe more objects compared to region-based methods. YOLO provides a reliable method for detecting cars used in the traffic flow analysis in real-time. As a single-step process YOLO has had many different model versions. It divides an image into a grid cells and predicts a certain number of boxes, and their confidence score. It also computes its likelihood that each box belongs to a class and adds to the confidence score in order to deliver the final results for detection.

For training and assessment, annotated datasets were collected from the open source platforms such as Roboflow and Kaggle. YOLOv8, which has improved feature extraction and an anchor-free detection approach and can process faster than its previous iterations, was applied to develop the main model. This enables real-time monitoring and makes the system eligible for real-world applications such as public safety, traffic control, and construction monitoring. The creation of a multi-class object detection system that can recognize vehicles, JCB machines, crowds, and potholes is the primary contribution of this work. To train and test the models, a custom annotated dataset was made using publicly accessible sources. All things considered, the project provides a practical solution that can support smart city development and infrastructure management. A methodology for creating a YOLOv8-based object detection model with the goal of identifying vehicles, potholes, JCB machines, and crowds is presented in this paper. The strategy seeks to maintain the system's stability, dependability, and suitability for real-time applications while achieving high detection speed and accuracy.

II. LITERATURE REVIEW

Object detection is a significant field in computer vision and is used in a variety of real-life applications like surveillance systems, traffic monitoring, and infrastructure inspection. In the previous years, many object detection systems were built using region-based deep learning models such as Faster R-CNN, etc. These models were able to have correct results due to the fact that they first created possible regions of objects and then they classified them. However, this two-step method took more processing time and more computational resources. Causing by this limitation, these methods were not always suitable for real-time applications, where one needs to detect something in a short amount of time.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

To overcome this problem, researchers introduced one-stage detection models such as YOLO. Unlike region-based methods, YOLO treats object detection as a single prediction task and processes the entire image in one pass. This design makes the detection process significantly faster while still maintaining a good level of accuracy. Due to its speed and efficiency, YOLO quickly became one of the most widely used object detection models in computer vision research and practical applications [1].With the further development of the model, various optimistic versions of YOLO appeared to improve the performance of detection. Indicatively, YOLOv4 has been used in road monitoring like potholes detection in different environmental factors like varying illumination and weather. Images were gathered in real-life settings on the roads and assigned labels to these images to train the model. This system recorded an accuracy of around 64.9% and this demonstrates that deep learning models can be helpful in detecting damage on the roads in real world situations [2].Another useful role that datasets will assist with object detection performance is through dataset. In one study, TRMSDN dataset was presented to detect traffic signs both in daylight and during the night. It was a set comprising of 4,386 labeled images with 12 categories. Once the dataset was changed into the format that the YOLOv8 understands and a data-enhancing strategy was applied, the model managed to obtain approximately 90 percent detectivity. This paper has shown that well planned datasets and training methods can be used to achieve a significant enhancement in the detection performance [3].

A custom UAV dataset used in one of the research studies was obtained to monitor cars using aerial images. The experiments of R-CNN-based approaches were compared with the performance of YOLOv3. The findings demonstrated that YOLOv3 was very precise and recalling high values with processing images at an extremely high speed. It was known that the YOLO-based systems can be used on applications that may involve real-time air monitoring and require speed [4].The other research area has concerned how to enhance performance of detection of small or challenging objects. An example is that the investigators used YOLOv8 to detect potholes in the work and suggested the fusion of multiple scales using features and improved methods of data augmentation. These enhancements contributed to the model becoming more precise with the detection of smaller potholes which had been a main drawback with previous methods of detection [5].Deep learning models can be trained using large datasets, and can take significant amounts of time to train. Many research works are conducted to minimize this effort through transfer learning. In this method, the trained models that have already learned on huge datasets like COCO Dataset are reconfigured to new detection problems. Directly trained YOLO models have been deployed for live detection systems without full retraining enabling researchers to create real-time apps in less time [6], [7].

The same methods have been applied to detect drone and satellite images which in this case possess a complicated background rendering object determination more difficult [8].Despite the fact that modern systems primarily operate based on deep learning, traditional methods of computer vision have also been investigated in the past. One of their methods made use of the Scale-Invariant Feature Transform method to retrieve image features as well as a Support Vector Machine in the classification process. These systems were however not as accurate as deep learning-based systems of detection with a performance of approximately 65%. This is a clear indicator of shortcomings of traditional approaches to intricate object detection tasks [9].Researchers have also tried to enhance YOLO models by solving some of their problems of heavy memory consumption and inability to detect tiny objects. One of the solutions was offered as Stack-YOLO, which sought to minimize the memory need yet attain higher object detection accuracy of small objects. This type of improvements assists in making object detection models more viable to embedded and edge-based system [10].

The process of drone real-time detection and embedded-research is an active field of study in recent years. Other UAV systems incorporate optical flow methods and deep learning systems to enhance the ability of the system to detect while also increasing system stability [11]. Some other projects have applied detection models to drones directly on the drone onboard computers. This will enable the system to do detection without the use of cloud processing, which will enhance response time, and also the system will be more applicable in real-time monitoring activities [12].Infrastructure monitoring has also been broadly used in object detection. Indicatively, rail defect detection systems based on drones with YOLO models have demonstrated consistent performance under realistic conditions of operation [13]. On the same note, deep learning procedures have been applied in detecting potholes and cracks in road damages. These works emphasise the need to label datasets and images with good quality to get the reliable detecting performance [14]. All in all, the review articles indicate that the integration of drone technology with rapid detection systems like YOLO can offer a viable and efficient solution to numerous applications related to monitors in the real world [15].

III. METHODOLOGY

The proposed methodology starts with the acquisition of data which is shown in fig 1, where the relevant images are gathered from public datasets. To enhance the robustness of the model, data augmentation methods such as resizing, flipping, rotation, and brightness changes are applied. The resulting images are annotated with bounding boxes and class labels in YOLO format.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

Following the annotation step, data construction takes place, where the dataset is split into training, validation, and testing sets along with the necessary configuration files. Finally the YOLOv8 model is trained on the constructed dataset, and the convergence of loss is tracked during the training process. The training process continues until convergence is reached, and the final detection model is obtained.

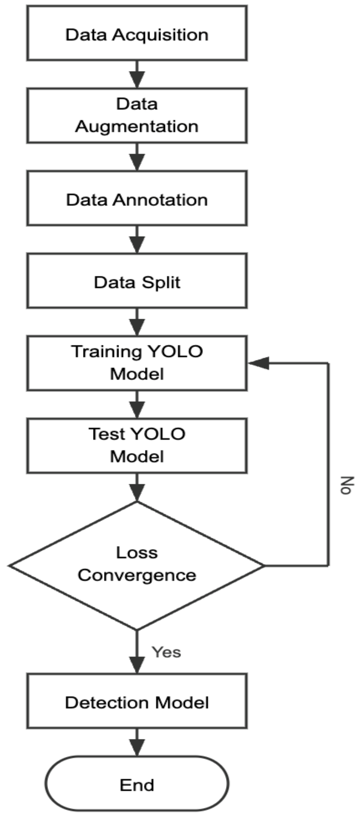


Fig.1. YOLOv8 Object Detection Workflow

A. *Data Collection/Acquisition*

For the project, public data sources such as Roboflow and Kaggle were used. These data sources contain images of scenes such as roads, vehicles, and traffic conditions which is taken by both UAVs and CCTV cameras. As shown in Fig. 1 and Fig. 2, sample images of crowd and JCB collected from the dataset are shown. Since the project focuses on a specific type of traffic and road analysis, additional data processing was required because a complete dataset dedicated to this task was not readily available. The images were re-annotated to remove inconsistencies and improve labeling accuracy. One of the major challenges was the limited availability of aerial traffic data.To solve this issue, multiple available data sources were combined, and basic data augmentation techniques were applied. As a result, a well-balanced dataset was created, which was successfully used to train the YOLOv8 model.

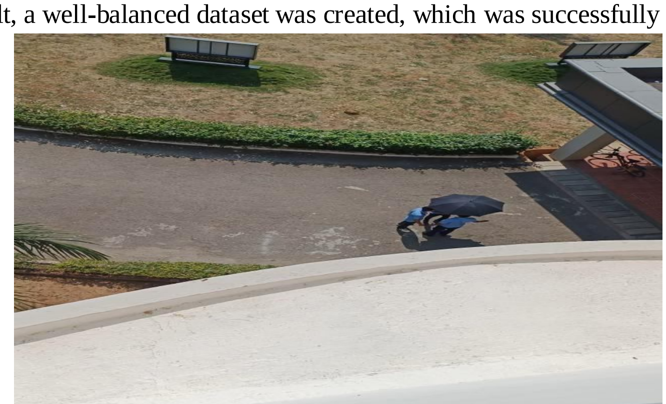


Fig.2.Sample image of crowd from dataset

Fig.3.Sample image of JCB

B. Data Augmentation

Images are subject to simple operations such as resizing, flipping, rotating and adjusting of brightness. There are also sophisticated features applied in YOLOv8, such as mosaic and mixup, as a combination of several images.

C. Data Annotation

The data were annotated and gathered with the help of the CVAT (Computer Vision Annotation Tool). Before the annotation, every image was inspected by hand, and the objects that were required like a car, a pothole, JCB machines, and crowds were identified by carefully putting the required objects into bounding boxes. The annotator chose the correct label of the classes and resized the bounding boxes to make sure that every object was sufficient. The semi-automatic services offered with CVAT such as interpolation, auto-annotation were also helpful and helped save time. Since the tool is easily usable, and allows the export of data under different forms, it was chosen in this project.Upon labeling all the images, the labeled data were exported to Training format in YOLO. Particular caution was applied when identifying very small or overlapping objects particularly on aerial images where objects might be hard to find.. Proper annotation helps the YOLOv8 model understand object features clearly and improves the overall detection accuracy.

D. Data Split

After the images are annotated, the dataset is divided into three parts- training, validation, and testing sets. The training set is used to teach the model how to detect objects. The validation set is used during training to check how well the model is learning . Finally, the testing set is used to evaluate the overall performance of the trained model on completely different data.

E. Train and Test YOLO Model

The YOLOv8 object detection model was trained using a transfer learning approach on a custom aerial image dataset represented as in equation (1):

$$D = \{(x_i, y_i)\}_{i=1}^{N} \qquad (1)$$

where x_i denotes the input image and y_i represents the corresponding ground-truth annotations, including bounding boxes and class labels.

1) Model Initialization

The training process was initialized using YOLOv8-S pretrained weights that were originally trained on the COCO dataset. The model can be mathematically represented as $M(x;\theta)$, where θ denotes the complete set of trainable parameters. For efficient learning, these parameters were divided into two subsets, θ_f and θ_t. The parameters θ_f correspond to the backbone layers responsible for extracting low-level visual features, while θ_t represent the detection layers that perform object localization and classification. In this work, the backbone layers were kept frozen to preserve previously learned generic features such as edges, shapes, and textures. Fine-tuning was done on the detection layers only using the custom dataset. The strategy minimizes the training time, minimizes the computational cost, and is specifically quite useful whenever the size of the available dataset is small.

2) Training Phase

All the input images were scaled to 640x640 during training in order to be similar. In a bid to enhance the ability of the model to generalize, the data augmentation methods of horizontal flipping and brightness adjustments were utilized according to equation (2):

$$x_i' = T(x_i) \qquad (2)$$

where T(.)is the augmentation function. In both training runs, the model estimated the bounding boxes, the probability of a class, and the objectness. The entire loss was formulated as in equation (3):

$$L_{\text{total}} = \lambda_1 L_{\text{box}} + \lambda_2 L_{\text{cls}} + \lambda_3 L_{\text{obj}} \qquad (3)$$

where L_{box}represents the IoU-based bounding box regression loss, L_{cls}denotes the classification loss (Focal Loss), and L_{obj}corresponds to the objectness loss. The weighting factors λ_1, λ_2, and $\lambda_3$balance the contribution of each loss component. Model parameters were then updated using gradient descent optimization as expressed in equation (4):

$$\theta_t^{(k+1)} = \theta_t^{(k)} - \eta \nabla_{\theta_t} L_{\text{total}} \qquad (4)$$

where $\eta$is the learning rate. Training performance was evaluated using Mean Average Precision (mAP) as defined in equation (5):

$$\text{mAP} = \frac{1}{C} \sum_{c=1}^{C} AP_c \qquad (5)$$

where Crepresents the number of object classes.

3) Validation and Loss Convergence

To ensure proper generalization, the model was evaluated on a separate validation dataset D_{val}. Training and validation losses were compared to detect possible overfitting. Loss convergence was assumed when the change in total loss between consecutive epochs became negligible, as expressed in equation (6):

$$| L_{\text{total}}^{(e)} - L_{\text{total}}^{(e-1)} | < \epsilon \qquad (6)$$

where ϵ is a small predefined threshold.

4) Testing Phase

All resulting model M* was optimized to come up with the final optimized version M. Measures of performance were also done through standard object detection measures, such as, mAP@0.5, Precision and Recall. Precision and Recall have been calculated as in equations (7), (8):

$$\text{Precision} = \frac{TP}{TP + FP} \qquad (7)$$

where TPdenotes true positives and FPdenotes false positives.

$$\text{Recall} = \frac{TP}{TP + FN} \qquad (8)$$

where TP, FP and FN denote true positives, false positives and false negatives respectively. The experiment outcomes prove that the trained YOLOv8 architecture is effective to identify various object classes including cars, potholes, construction equipment, and in crowd regions in aerial and traffic image data. The combination of transfer learning, balanced loss optimization, and systematic validation allowed achieving the robust and computationally efficient YOLOv8-based multi-class object detector. The approach suggested has a good detection performance, as well as lower training complexity.

F. Detection Model

Once the training and convergence of the loss occur successfully, the final detection model is received. It is possible to apply this trained model to detect and classify objects in real time aerial and traffic images.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

IV. RESULTS AND DISCUSSION

Both quantitative and qualitative evaluation was used to test the performance of the proposed object detection system based on YOLOv8, through a visual examination. The analysis is based on convergence behavior training, accuracy in detection, localization quality and inference . The findings prove that the model can acquire complex visual features and can also generalize well to unknown infrastructure environments.

A. Figures and Tables

Object Class	Threat Category	Final Epoch	mAP@0.5 (Detection Accuracy)	mAP@0.5:0.95 (Localization Quality)	Precision (P)	Recall (R)
Car	High-Risk TPI	40	0.838	0.678	0.786	0.941
JCB	High-Risk TPI	40	0.838	0.678	0.786	0.941
Crowd	TPI/Access	40	0.872	0.500	0.832	0.834
Pothole	Maintenance/Defect	61	0.728	0.428	0.769	0.658

Table 1.final metrics for Car, JCB, Crowd, and Pothole

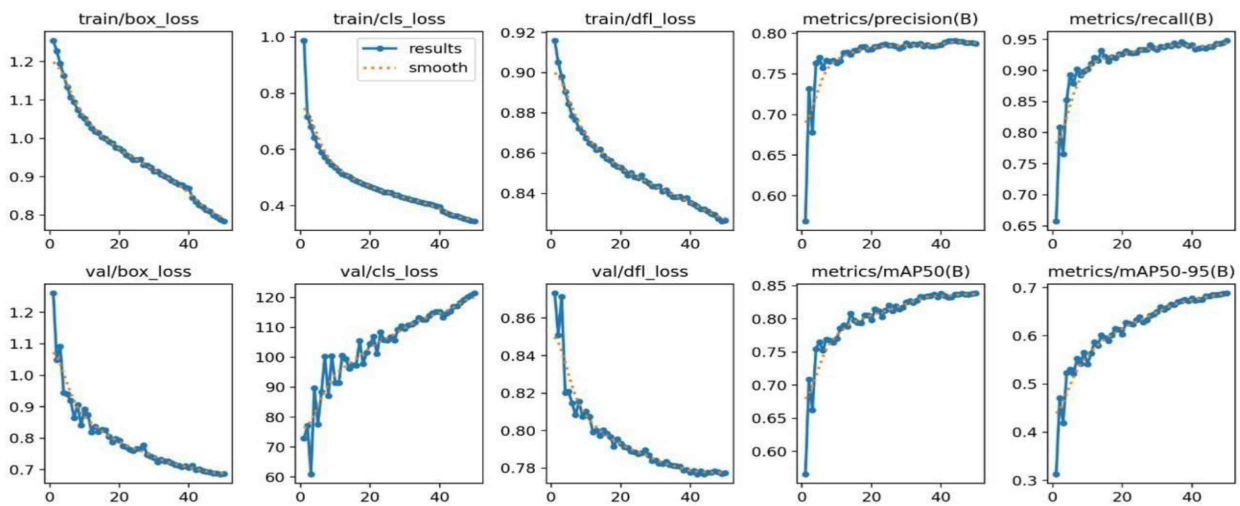


Fig.4.. Model Convergence Plot

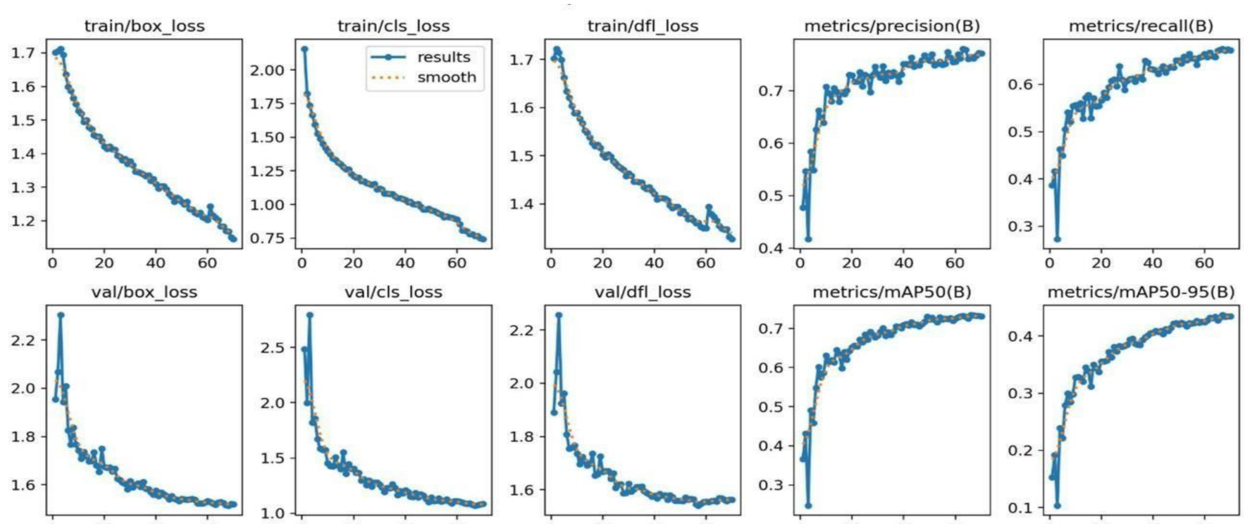


Fig.5. Model Convergence Plot

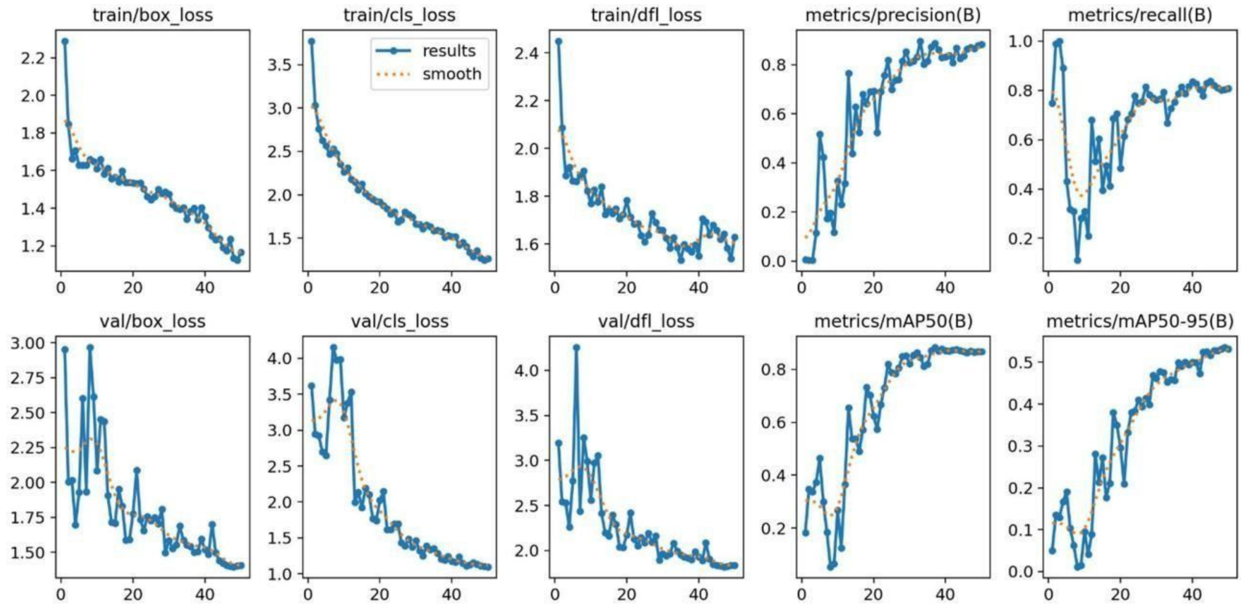


Fig.6. Model Convergence Plot

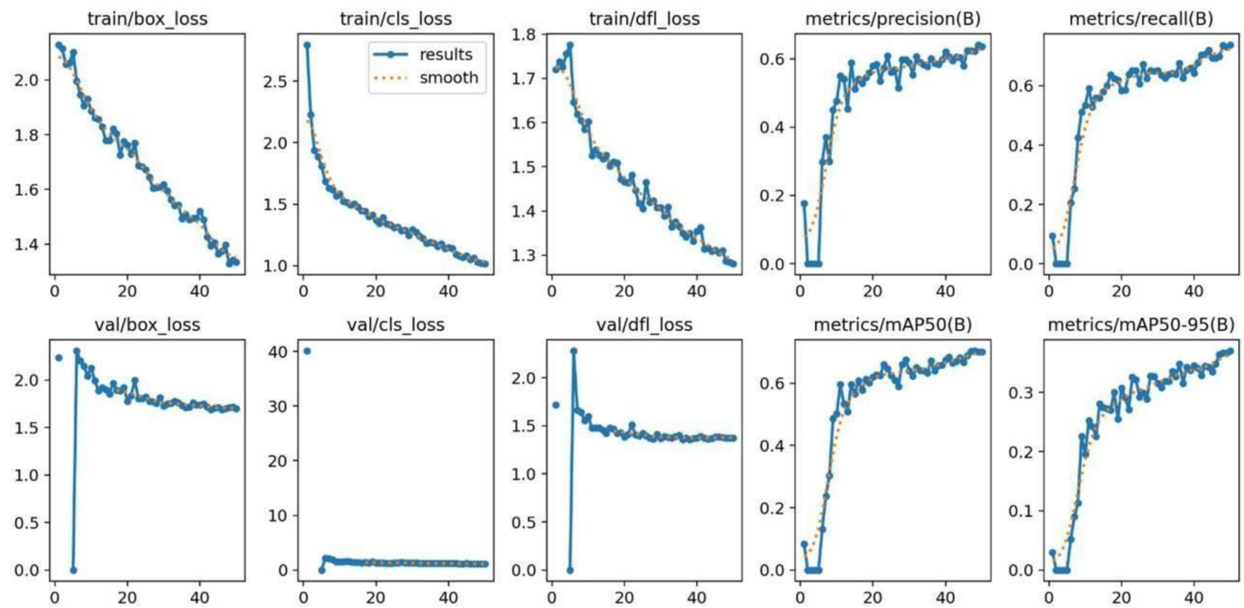


Fig.7. Model Convergence Plot

B. Model Convergence and Training Behavior

The YOLOv8-based model was trained by minimizing a complex loss function using gradient descent optimization. Mathematically, convergence occurs when the gradient of the loss function approaches zero:

$$\|\nabla L(\theta)\| \approx 0$$

where $L(\theta)$represents the loss function and θ denotes the model parameters. Figures 4-7 indicate that the model revived the model through numerous cycles to ensure the learning was stable. At the beginning the loss decreased rapidly when the model began to pick the basic things up such as object edges and bounding box spots. At a later point the loss ceased to reduce implying that the model had become converged. The almost similarity of training and validation loss indicates that the model is not overfitted and it remains effective even with novel data.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

C. Quantitative Performance Analysis

The prepared dataset with cars, JCB machines, crowds, and potholes was used to test the detection accuracy. The system recorded the highest accuracy 77% and real time processing speed of 150 Fps. The extent of the metrics of classes is summarized in Table I. Crowd class was the most accurate at 87.2%. There is also the guarantee of high Cars and JCB recall rate (94.24)% so that the vital objects are unlikely to be overlooked in the course of surveillance. Although Potholes had worse measures (72.8%) owing to their irregular geometry, the model was able to pick them after the long training of 61 epochs.

D. Qualitative Detection Outputs

The feasibility of the developed system is proved practically with reference to visual analysis of inference results in different operation scenarios. Through experimenting the model in various and other complex environments, we confirm the capability of the model to shift an experimental training environment to the real world.

1) Car Detection

Fig.8. Car Detection

As shown in Fig. 8, the initial assessment shows the possibility of the model of a drone to spot vehicles with great accuracy on its high-angle view. Although the objects required by the high-altitude camera angle are small, the system experiences limited performance variability and localizations the position of a number of cars with high accuracy among various frames. This demonstration shows that the model is reliable in automated traffic monitoring and management of smart infrastructure in cities.

2) Crowd Detection

Fig.9. Crowd Detection

On Public Safety and Crowd Monitoring, Fig. 9.shows the effective detection of people in high-density crowds. The result is a demonstration of the usefulness of the system to urban security in that there are actual people correctly identified by the system even when they are crowded . The high accuracy of this category is crucial to ensuring a high level of the management of the safety of the population in places with large traffic like transport centers and squares.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

3) Infrastructure and Construction Management

Fig.10. JCB Detection

Fig. 10 indicates that the model can only identify JCB machines and other construction machinery even in a complicated scene. The ability to detect such objects is one of the major attributes of this project that will enable more adequate monitoring of busy building works, and may facilitate the process of applying robotization to the road maintenance. Although the size of these machines is great, the system can easily tell those machines and the background that is cluttered and messy.

4) Pothole Detection Results

Fig.11 Potholes Detection

Equally, as it is indicated in Fig. 11, the model is able to identify surface cracks and structural damage on various parts of highways. This is a significant aspect since it assists in the accurate identification of the areas where roads are damaged enabling the repair people to repair such damages more effectively. The shapes of potholes are hard to notice because they are irregular and have low contrast at high speed, however the system can identify them and then sketch a bounding box surrounding the pothole. Big potholes with high contrast are being detected with high confidence (0.7) and some smaller or barely noticeable damage is being detected as well, with lower but still useful confidence (0.3).

E. Discussion

The experimental outcomes show that the model using YOLOv8 is very effective in tracking various settings. The great design of this architecture is one of its greatest benefits over the previous models. It can identify objects of extremely dissimilar scales like large JCB machines and small road potholes with much more precision than earlier architectures like YOLOv3. Besides this, the model is much faster at detection compared to other methods such as Faster R-CNN, which tend to take a longer time to generate the same.The system recorded a high success rate especially in vehicles and heavy machinery detection, which is a critical aspect of automated infrastructure monitoring. The model was capable of identifying smaller or irregular objects like potholes although this may be very difficult to detect. The more the epochs of training, the more the model can learn the peculiarities of such objects and the better it will detect..

V. CONCLUSION AND FUTURE SCOPE

The project was able to create an effective tracking system that is built on the YOLOv8 architecture to detect vehicles, crowds, heavy machinery, and road damage. This model proved to be very suitable and rapid in the localization and detection of these objects in a wide scope of real life situations.

As the experimental results show, YOLOv8 is one of the most suitable models in this application that beats the older architecture like Faster R-CNN. The system has a maximum processing rate of 150 FPS; it is optimal in a live application in smart cities and highway surveillance. Although the model was most accurate with large and distinct objects, e.g. cars and JCB machines, its capability of detecting smaller road faults such as a pothole even with a lower confidence level is vital in avoiding failures of infrastructure.

The system has several ways into which it can be enhanced in future. To amplify the dataset with a greater variety of images i.e. those that were taken during heavy rain, fog or night, etc. would make the model more accurate in the challenging situations. The system could also be expanded to a more sophisticated traffic analysis system that would be able to estimate the speed of the vehicles and automatically identify license plates. Lastly, an alarm system might be adopted to give real-time information about the precise positions of the potholes to the repair crews so that they might undertake instant fixes and enhance the roads.

VI. ACKNOWLEDGMENT

We would like to express our sincere gratitude to Remote Data Exchange which entrusted us with this significant project. Their experience in handling the underground fuel pipeline systems of BPCL and BORL. We thank their guidance, too.

REFERENCES

[1] A. B. Amjoud and M. Amrouch, "Object Detection Using Deep Learning, CNNs and Vision Transformers: A Review," in IEEE Access, vol. 11, pp. 35479-35516, 2023, doi: 10.1109/ACCESS.2023.3266093.

[2] C. Dewi, R. -C. Chen, Y –C. Zhuang and W. E. Manongga, “Image Enhancement Method Utilizing YOLO models to Recongnize Road Markings at Night,” in IEEE Access, vol. 12, pp. 131065-131081, 2024, doi: 10.1109/ACCESS.2024.3440253.

[3] Ms. Shaik Ishrath Anjum, Ms.Syed Roohi, “Real-Time Object Detection using The YOLO Algorithm With the Opencv framework,” in IJCRT.ORG, vol 11, ISSN: 2320-2882, 2023 IJCRT.

[4] Thomas Moranduzzon, Farid Melgani, “An object Detection Technique for very high resolution remote sensing images,” in WoSSPA 2013.

[5] Yaru Cao, Zhijian He, Lujia Wang, Wenguan Wang, “VisDrone- DET2021: The Vision Meets Drone Object detection Challenge Result,” in IEEE/CVF (ICCVW) 2021.

[6] E. Arkin, N. Yadikar, Y. Muhtar and K. Ubul, "A Survey of Object Detection Based on CNN and Transformer," 2021 IEEE 2nd International Conference on Pattern Recognition and Machine Learning (PRML), Chengdu, China, 2021, pp. 99-108, doi: 10.1109/PRML52754.2021.9520732.

[7] R. Varghese and S. M., "YOLOv8: A Novel Object Detection Algorithm with Enhanced Performance and Robustness," 2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS), Chennai, India, 2024, pp. 1- 6, doi: 10.1109/ADICS58448.2024.10533619.

[8] Z. Zou, K. Chen, Z. Shi, Y. Guo and J. Ye, "Object Detection in 20 Years: A Survey," in Proceedings of the IEEE, vol. 111, no. 3, pp. 257- 276, March 2023, doi: 10.1109/JPROC.2023.3238524.

[9] M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

[10] X. Han, Y. Zhong, R. Feng and L. Zhang, "Robust geospatial object detection based on pre-trained faster R-CNN framework for high spatial resolution imagery," 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 2017, pp. 3353-3356, doi: 10.1109/IGARSS.2017.8127716.B. Hou, B. Li, Y. Li and J. Ma

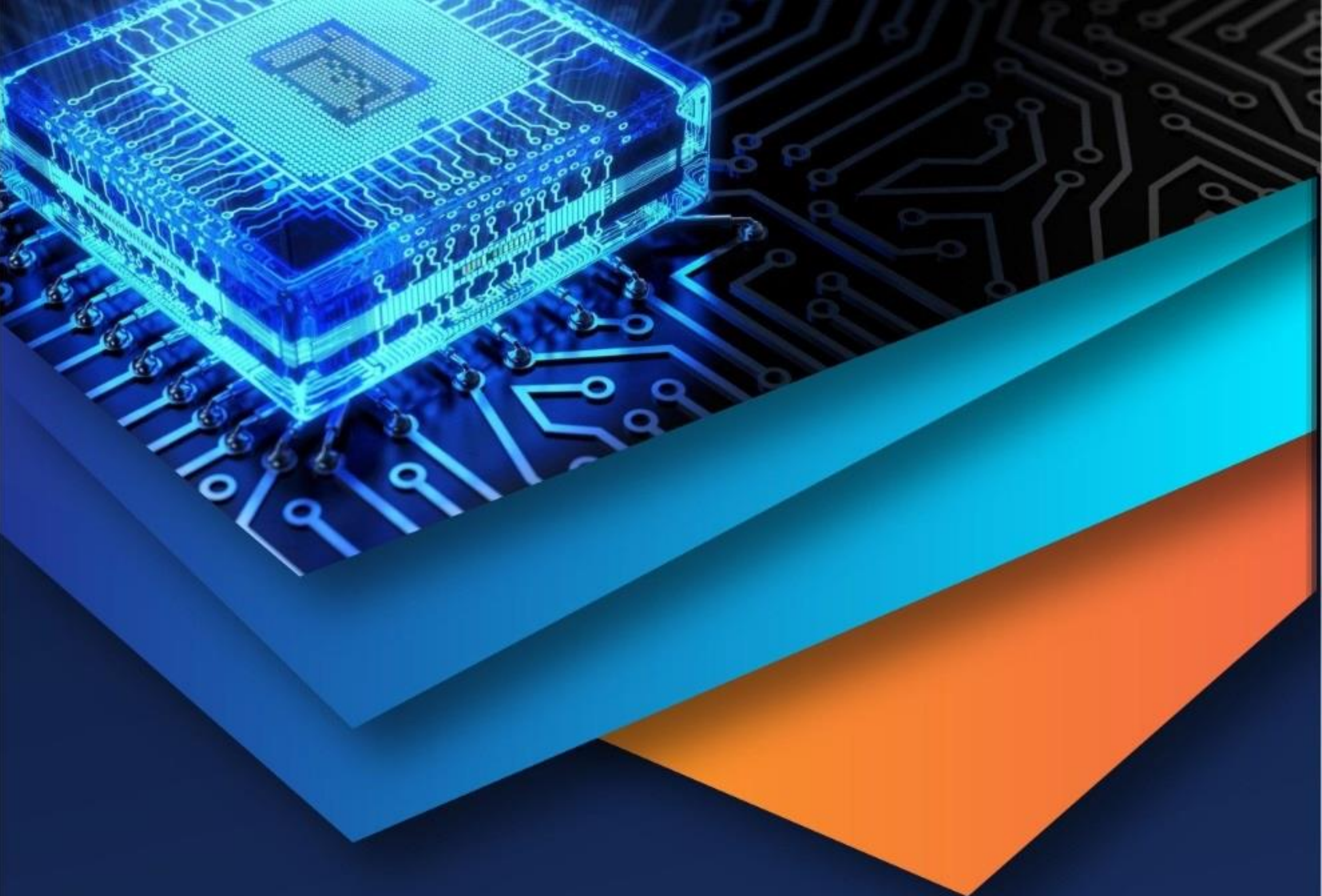
[11] Q. Wu, Y. Zhou, X. Wu, G. Liang, Y. Ou, and T. Sun, “Real-time running detection system for UAV imagery based on optical flow and deep convolutional networks,” IET Intelligent Transport Systems, vol. 14, no. 5, pp. 278–287, Mar. 2020, doi: 10.1049/iet-its.2019.0455.

[12] I. Aydin, M. Sevi, K. Sahbaz, and M. Karakose, “Detection of rail defects with deep learning controlled autonomous UAV,” Conference Paper, 2019.

[13] A. Lekidis, A. G. Anastasiadis, and G. A. Vokas, “Electricity infrastructure inspection using AI and edge platform-based UAVs,” Energy Reports, vol. 8, pp. 1394–1411, Aug. 2022, doi: 10.1016/j.egyr.2022.07.115.

[14] S. Guan, H. Liu, H. R. Pourreza, and H. Mahyar, “Deep learning approaches in pavement distress identification: A review,” arXiv preprint arXiv:2308.00828, Aug. 2023

[15] L. P. Osco et al., “A review on deep learning in UAV remote sensing,” International Journal of Applied Earth Observation and Geoinformation, vol. 102, pp. 102456, July 2021, doi: 10.1016/j.jag.2021.102456.



10.22214/IJRASET

INDEX COPERNICUS 45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

TOGETHER WE REACH THE GOAL

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY