POTHOLE DETECTION AND PRIORITY RANKING USING DEEP LEARNING

Harsh Gupta[1], P Rajeev Siddarth[2], Dr P. S. Thanigaivelu[3]

[1]Department of Computing Technologies SRM Institute of Science and Technology Chennai, India*
hg6364@srmist.edu.in
[2]Department of Computing Technologies SRM Institute of Science and Technology Chennai, India rs6820@srmist.edu.in
[3]Department of Computing Technologies SRM Institute of Science and Technology Chennai, India
thanigap2@srmist.edu.in

Abstract: *Potholes on the road pose a great danger to transport safety and infrastructure performance. This paper presents a deep learning-based framework for pothole detection and priority ranking. Three object detection models are implemented and compared with each other in terms of precision, recall, and mAP: YOLOv8, Faster R-CNN, and RetinaNet. Experimental results show that YOLOv8 achieves the best overall performance with a balanced trade-off between accuracy and efficiency. Identified potholes are further classified into severity levels and ready to be incorporated with a context-aware prioritisation system. The suggested framework demonstrates a realistic implementation in the context of intelligent road maintenance, and the future research will be aimed at the optimisation-based ranking and practical implementation.*

Received: 04/03/2026 Revised: 10/04/2026 Acceptance: 16/04/2026 Publication: 22/04/2026

I. INTRODUCTION

One of the biggest issues that plague the contemporary transportation systems is road surface degradation. Potholes cause damage to vehicles, traffic jams and risk of accidents. The traditional road inspection is very dependent on manual monitoring, which leads to slow response to maintenance.

The latest developments in deep learning and computer vision allow automated monitoring of road conditions based on real-time image analysis [1][11]. Although the current methods are effective in identifying potholes, the majority of systems do not consider their urgency in context. This study builds on pothole detection to intelligent prioritisation by combining severity estimation with future contextual ranking systems.

II. RELATED WORK

The recent development of vision-based pothole detection has been dominated by deep learning models, especially convolutional neural networks and recent object detection models like YOLO, Faster R-CNN, and RetinaNet [1][4][5]. The current literature is mainly concerned with enhancing the detection accuracy and bounding box localisation, but there is little attention on how the detections can be used in making practical infrastructure decisions.

Unlike vision-based methods, sensor-based methods that use accelerometers and vibration analysis have also been investigated in detecting potholes. Nevertheless, they tend to have high false-positive rates because of environmental noise and do not provide spatial context, which reduces their usefulness in large-scale applications.

Initial studies in road damage detection used conventional image processing algorithms, such as edge detection, texture analysis, thresholding, and manual feature extraction algorithms, such as Histogram of Oriented Gradients (HOG) with Support Vector Machines (SVM) [6]. Although these techniques proved to be initially feasible, they were very sensitive to changes in illumination, shadows, road textures, and camera positions, which restricted their strength in the real world.

The development of deep learning greatly enhanced the detection performance. Two-stage detectors like Faster R-CNN improved the accuracy of localisation by dividing the region proposal and classification phases, leading to high recall and accurate object localisation [4]. Nevertheless, they were computationally complex and could not be used in real-time. To overcome this, single-stage detectors like YOLO variants and RetinaNet were proposed [1][2][5]. YOLO-based models are faster and more efficient, thus suitable in real-time detection, whereas RetinaNet uses focal loss to address the issue of class imbalance and detect challenging samples effectively [5].

Recent papers have discussed different enhancements to these architectures, such as improved data augmentation methods, multi-scale training, anchor optimisation, and attention mechanisms to improve detection performance in a wide range of environmental conditions. Regardless of these developments, the main emphasis of current studies is on the evaluation measures of Precision, Recall, F1-score, and mean Average Precision (mAP) [8][9].

The major weakness of existing solutions is the absence of a connection between the outputs of detection and the decision-making at the infrastructure level. Although models are effective in identifying potholes, they do not give information on the relative significance or urgency of repair. As an illustration, potholes that have the same level of detection confidence can vary greatly in their actual effect on the road depending on their size, road properties, or traffic. To fill this gap, the current work goes beyond the traditional detection by integrating various object detection models, including YOLOv8, Faster R-CNN, and RetinaNet, into a single evaluation framework. Besides detection, a contextual priority ranking mechanism is proposed, which uses geometric features based on bounding boxes as well as detection confidence to give relative priority to each pothole. This strategy will help to fill the gap between detection and actionable decision-making, which will help to make road maintenance strategies more effective and scalable.

III. DATASET DESCRIPTION

The data in this paper is a set of annotated road surface images that have been filtered to detect potholes with deep learning-based object detection models. It has a total of 1,977 images, 1,581 images of which are used in training and 396 images in validation. The pictures are gathered in various publicly available sources and reflect a variety of real-life road conditions, which allows effective model learning and testing [8][9].

A. Data Composition

The images are accompanied by an annotation file in YOLO format, which includes the information about the bounding box that indicates the position of potholes in the image. The data is in a single-class detection format, with all potholes being classified as a single class. This design option enables the detection model to concentrate on correct localisation, and severity classification (Minor, Medium, Major) is addressed independently in the proposed prioritisation framework [3].

B. Image Characteristics

The dataset contains images that were taken in a very diverse environment and visual conditions. These are changes in lighting (bright daylight, low light, and shadows), weather conditions (dry and wet roads), and various road surface textures. Also, the dataset includes images with occlusions due to vehicles, pedestrians, and road markings, and different camera angles and distances. This variety guarantees that models that are trained on this dataset can be generalised to the real-world deployment conditions.

C. Annotation Format

The annotations are given in the YOLO format, in which each bounding box is defined as:
class_id x center y center width height
All coordinates are normalised to the range [0, 1] with respect to the image dimensions. This format facilitates effective training with YOLO-based architectures and can be readily modified to be used with other object detection models like Faster R-CNN and RetinaNet with proper preprocessing.

D. Dataset splitting

The data is already split into training and validation sets, with an approximate 80:20 split. The training set is composed of 1,581 images that are used to learn the model, and the validation set is composed of 396 images that are used to evaluate the performance of the model. This division guarantees the equal distribution of data and allows the assessment of model generalisation to be reliable.

E. Dataset Challenges

The dataset has a number of practical challenges that make pothole detection more complex. They are high variability in pothole size, shape, and appearance, inability to detect small or partially visible potholes, and similarity of the background between potholes and other irregularities in the road surface like cracks or patches. Also, the lighting differences, shadows, and motion blur also make detection more difficult. These aspects render the dataset appropriate to assess the strength of sophisticated deep learning models.
ches is the absence of integration between the outputs of detection and infrastructure-level decision-making. Although models are effective in identifying potholes, they do not give information on the relative significance or urgency of repair. To illustrate, potholes of similar detection confidence can vary widely in their actual effect on the road depending on size, road properties, or traffic.
To fill this gap, the current study goes beyond traditional detection and includes several object detection models, including YOLOv8, Faster R-CNN, and RetinaNet, in a single evaluation framework. Besides detection, a contextual priority ranking mechanism is proposed, which uses both geometric features based on bounding boxes and detection confidence to provide relative importance to each pothole. This methodology fills the gap between detection and actionable decision-making, which is part of more effective and scalable road maintenance strategies.

IV. METHODOLOGY

The suggested system provides a common and standardized experimental platform of pothole detection with deep learning-based object detection models. Three architectures—YOLOv8, Faster R-CNN, and RetinaNet—are implemented and evaluated under identical conditions. Each model is trained and evaluated on the same dataset, preprocessing pipeline, and evaluation metrics to provide a fair and unbiased comparison.

A. Dataset and Experimental Setup

The data set is comprised of 1,977 annotated road images, which are split into 1,581 training images and 396 validation images. The images are annotated with bounding box annotations in the YOLO format.
All models are trained using the same data split and evaluated on the validation set using standard detection metrics. The Adam optimizer is used in the training process to achieve efficient convergence and adaptive learning rate adjustment [10].

B. Preprocessing and Data Preparation

The input images are all resized to a constant resolution to make them consistent across models and to have the same input dimensions when training. This is necessary to ensure stable learning particularly when comparing different architectures in the same conditions.

The following transformation [11] is used to normalise pixel values to the range [0,1]:

$$x' = x / 255$$

This normalisation makes sure that the input data is in a similar scale, enhancing numerical stability and speeding up convergence in training.

All models have the same preprocessing pipeline to provide a fair and unbiased comparison. Although simple transformations are used, more intensive data augmentation methods like rotation and translation are not intensively used in the current implementation and are viewed as a part of the future work to further improve the model generalisation.

C. Model Implementation

Three models are used to capture various detection paradigms:

YOLOv8 Model

YOLOv8 is a single-stage object detector that localises and classifies objects in a single forward pass, based on the design principles of the YOLO family of detectors [1], [3]. The model estimates bounding boxes and confidence scores and class probabilities. It employs multi-scale feature extraction and anchor-free detection mechanism to enhance performance with different object sizes [3].

The overall loss function is a combination of localisation, classification, and objectness losses:

L = L_box + L_cls + L_obj

Intersection over Union (IoU) is used to optimise bounding box regression:

$$IoU = \frac{|B_p \cap B_{gt}|}{|B_p \cup B_{gt}|}$$

YOLOv8 employs multi-scale feature extraction and anchor-free detection to enhance performance with different object sizes.

1) Faster RCNN

Faster R-CNN is a two-stage object detector that extends the previous region-based detectors like Fast R-CNN by including a Region Proposal Network (RPN) to be trained end-to-end, and then a classification and regression head [4][7]. The RPN produces candidate object regions that are refined and classified in the second stage. This architecture enhances the localisation accuracy and detection performance, especially on small objects, but it comes with increased computational complexity than single-stage detectors [4].

The RPN produces candidate regions and each region is classified and narrowed down. The total loss is given by:

$$L = [L_{cls} + L_{reg}]$$

Where:

- L_{cls} : $classification\ loss$Lreg
- L_{reg} : $bounding\ box\ regression\ loss$

Bounding box regression is computed as:

$$L_{reg} = \sum_i SmoothL1\left(t_i - t_i^*\right)$$

This two-step method enhances the accuracy of localisation but adds complexity to computation.

2) RetinaNet

RetinaNet is a single-stage object detector that uses Focal Loss to solve the problem of class imbalance in training [5]. It uses a Feature Pyramid Network (FPN) to represent features in a multi-scale way, which allows the successful detection of objects at various scales. The focal loss function minimizes the impact of easy examples and concentrates training on more difficult, misclassified samples, which enhances the overall detection performance [5].

The focal loss is defined as:

$$FL\left(p_t\right) = -\alpha\left(1 - p_t\right)^{\gamma} \log log\left(p_t\right)$$

Where:

- p_t : $predicted\ probability$
- p_t : $predicted\ probability$
- p_t : $predicted\ probability$

This loss function reduces the contribution of easy examples and focuses training on harder samples.

D. Performance Evaluation

Model performance was evaluated using standard object detection metrics:

$$Precision\ (P):\ P = TP / (TP + FP)$$
$$Recall\ (R):\ R = TP / (TP + FN)$$
$$F1 - Score:\ F1 = 2PR / (P + R)$$
$$Mean\ Average\ Precision\ (mAP):\ mAP = (1/N) \times \Sigma\ AP\_i$$

Where AP_i is the Average Precision for class i, and N = 3 in this study. Both mAP@0.5 and mAP@0.5:0.95 were used to assess the localisation accuracy at different IoU thresholds [8][9].

E. Experimental Accuracy

All models are trained with the same experimental conditions, such as dataset splits, pre-processing steps, and evaluation metrics. Hyperparameters are optimized in a controlled environment to make sure that performance differences are due to model architecture and not experimental variations.

V. PRIORITY RANKING

To further expand pothole detection to actionable infrastructure management, a weighted priority scoring system is presented to prioritize detected potholes in terms of their contextual severity. Although object detection models like YOLOv8 can offer precise localisation and confidence scores [3], they do not necessarily measure the urgency of repair. The proposed scoring model overcomes this shortcoming by combining geometric, contextual, and confidence-based features into a single priority measure.

The priority score is developed as a weighted average of four important factors: pothole size, traffic relevance, detection confidence, and location importance. All these factors add to the general urgency of a pothole in the real world. The mathematical formulation of the priority score is given by:

$$Priority\ Score = w_1 \cdot S + w_2 \cdot T + w_3 \cdot C + w_4 \cdot L$$

Where S is the normalised size of the pothole based on the size of the bounding box, T is the traffic score of the road segment, C is the confidence score of the detection model [1][4], [5], and L is the relative significance of the location based on road hierarchy. The weights w 1, w 2, w 3 and w 4 are determined empirically in such a way that:

$$w_1 + w_2 + w_3 + w_4 = 1$$

Geometric severity and traffic conditions are given more weight in this study because they are the direct determinants of safety and infrastructure impact. Common weightings are w 1 = 0.4, w 2 = 0.3, w 3 = 0.2, and w 4 = 0.1, but these can be adjusted depending on the needs of the application.

The size component S is computed using the area of the predicted bounding box, normalised with respect to the image dimensions. This expression is in line with conventional object detection representations and assessment routines [8][9]. The bigger the potholes, the more severe they are because they can cause more damage to the vehicle and pose safety risks. Traffic score T is calculated based on geospatial data collected by GPS tagging. Roads are classified into hierarchical groups including highways, arterial roads, and residential streets, with roads with higher traffic having higher weights because they have a greater influence on mobility and congestion.

The confidence score C, which is a direct result of the object detection model, is an indication of the reliability of the prediction [1][4][5]. The inclusion of this factor will make sure that the uncertain detections will have less impact on the final priority and thus the impact of false positives will be minimized. The location importance factor L further narrows down prioritisation by taking into account strategic relevance, e.g. proximity to intersections, urban centres, or critical infrastructure areas.

After calculating the composite priority score, potholes are classified into three levels of severity: Minor, Medium, and Major, according to predetermined threshold values. As an example, a score of less than 0.3 is considered Minor, a score between 0.3 and 0.7 is considered Medium, and a score of more than 0.7 is considered Major. This categorization allows easy interpretation and can be integrated with maintenance scheduling systems.

The weighted scoring mechanism proposed offers a scalable and flexible method of pothole prioritisation. This framework uses real-world contextual parameters unlike the traditional methods which only use geometric features, which can be used to make more informed decisions regarding road maintenance. Moreover, the modular design enables the incorporation of other factors, including pothole density or time-dependent traffic changes, in subsequent improvements. This strategy is consistent with the goal of shifting detection-based systems to smart, context-sensitive infrastructure management systems.

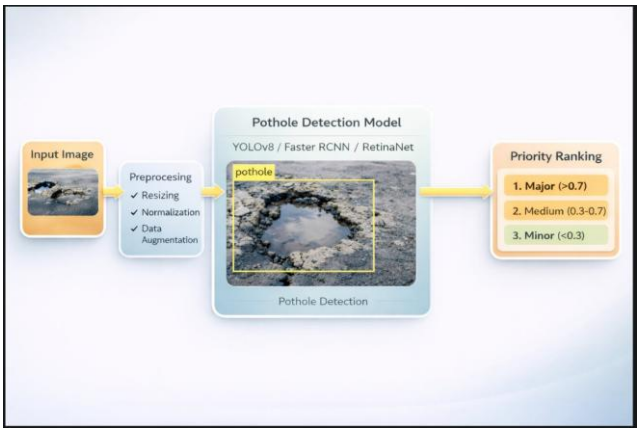

Fig 1. Simplified Pothole Detection and Priority Ranking Pipeline

VI. RESULTS AND DISCUSSION

The performance of the three implemented object detection models—YOLOv8, Faster R-CNN, and RetinaNet—was evaluated using a consistent dataset, preprocessing pipeline, and training-validation split. This makes the comparison fair and the differences in performance due to architectural and methodological differences only. The metrics of evaluation that will be taken into account are Precision, Recall, F1-score, Accuracy, and mAP@0.5.

A. Training Convergence and Learning Behavior

The mAP and precision-recall trends across epochs were used to analyse the training behaviour of the models. It is noted that all models show a rapid improvement in performance in the first training phase, and then a gradual stabilisation as training continues.

The mAP@0.5 curve exhibits a steep rise in the initial epochs, which means that the features are learned successfully, and reaches the plateau after about 40-60 epochs. Likewise, the values of precision and recall stabilise with time, indicating that the models achieve an optimal compromise between detection accuracy and generalisation.

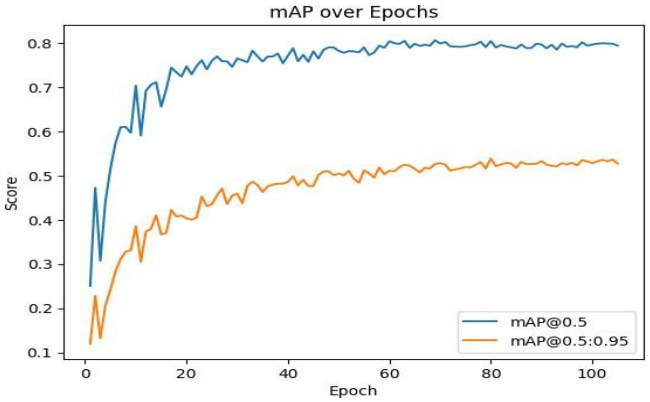

Fig 2. mAP over Epochs for YOLOv8

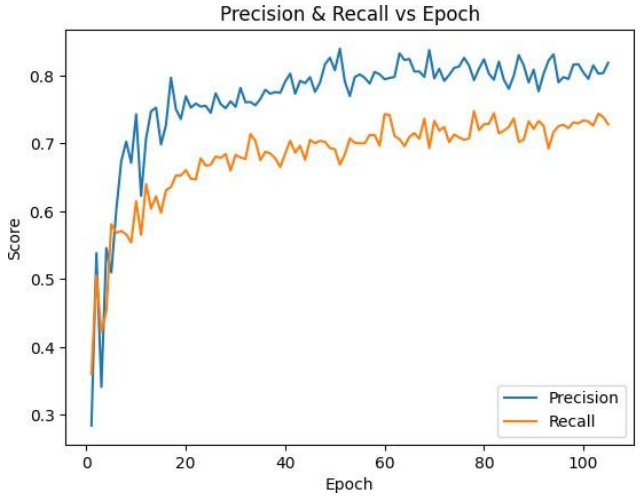

Fig 3. Precision and Recall vs Epoch for YOLOv8

The convergence behaviour shows that the models are well-trained with no major overfitting, since there are no drastic changes in subsequent epochs. The difference between accuracy and recall underscores the trade-off nature of object detection tasks.

B. Performance of Faster R-CNN

The Faster R-CNN model has a high recall of 0.9169, which means that it is able to detect most of the potholes in the dataset. The accuracy of 0.8278 indicates that the majority of the predictions of the model are accurate, but there are still a sufficient number of false positives. The obtained F1-score of 0.8701 indicates a balanced performance in terms of precision and recall.

Nevertheless, the detection accuracy of 0.7700 indicates that although the model is very effective in detecting potholes, it generates redundant or overlapping detections, which slightly influence the overall accuracy. This is typical of two-stage detectors, which emphasize completeness of detection.

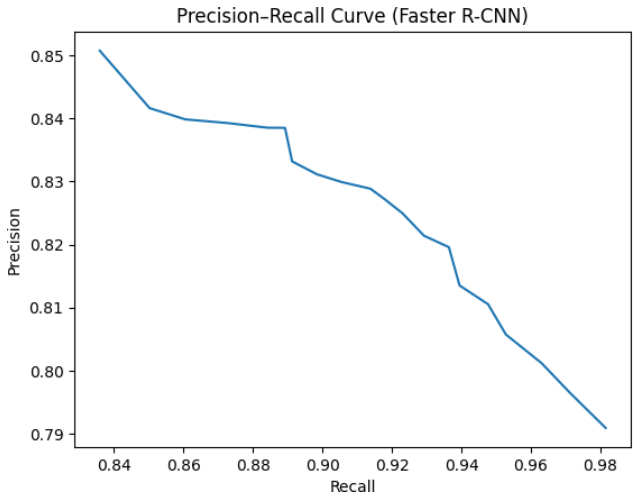

Fig 4.Precision-Recall Curve for Faster R-CNN

C. Performance of YOLOv8

YOLOv8 has a precision of 0.8375 and a recall of 0.6930, which means that it has a more conservative detection strategy than Faster R-CNN. The model generates fewer false positives, but misses more potholes, especially smaller or less pronounced ones. This trade-off is reflected in the F1-score of 0.7584.

Nevertheless, YOLOv8 has the best mAP at 0.5 of 0.8066 and mAP at 0.5:0.95 of 0.5266, and has better localization accuracy at different IoU thresholds. This underscores the fact that the model is strong in producing accurate bounding boxes and uniform detections.

The reduced accuracy of 0.5803 is explained by its strict prediction filtering, which decreases the detection coverage but increases prediction confidence.

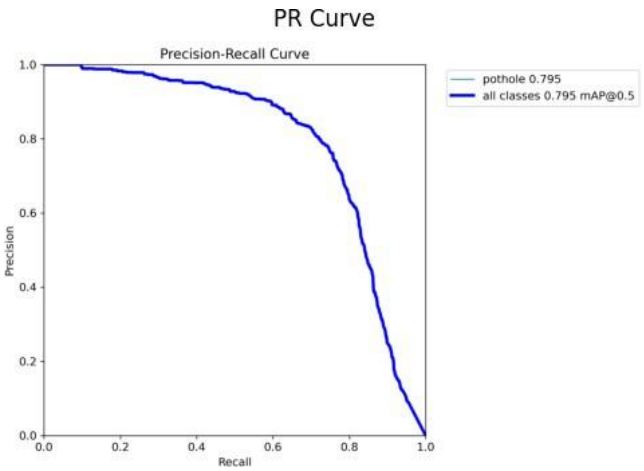

Fig 5.Precision-Recall Curve for YOLOv8

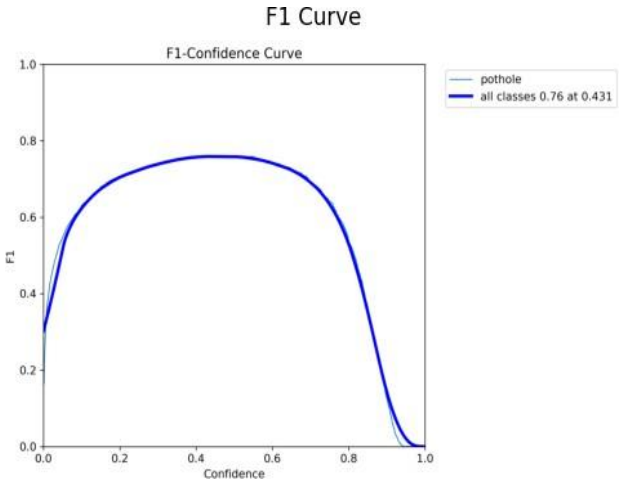

Fig 6.F1 curve for YOLOv8

D. Performance of RetinaNet

RetinaNet has a balanced performance with a precision of 0.8109 and a recall of 0.8839. The model is able to capture many potholes and still have a reasonable prediction accuracy. The F1-score of 0.8458 is a good trade-off between precision and recall.

This balance is further supported by the detection accuracy of 0.7329. The estimated mAP at 0.5 of 0.7168 indicates that it is competitive in localization, albeit slightly less than YOLOv8.

This is explained by the fact that the focal loss mechanism can be used to correct the imbalance in the classes, but it can also lead to more false positives at lower confidence levels.

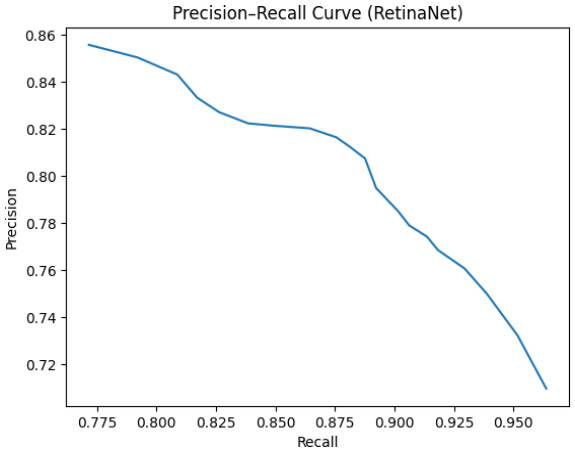

Fig 7.Precision-Recall curve for RetinaNet

E. Error Analysis

A confusion matrix is examined to gain a more insight into the performance of the model. The matrix shows the distribution of true positives, false positives, and false negatives among predictions.

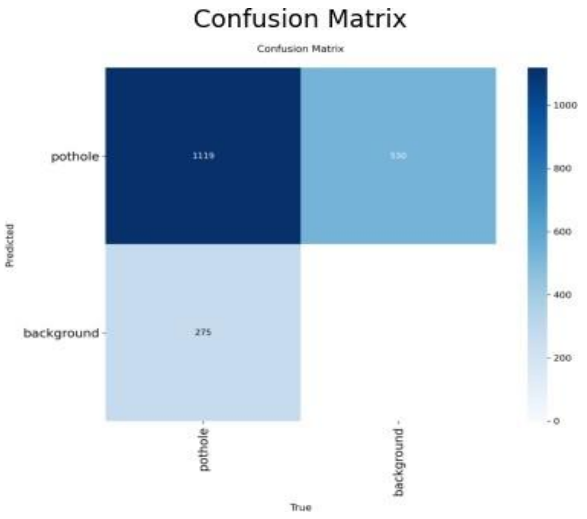

Fig 8.Confusion Matrix for YOLOv8

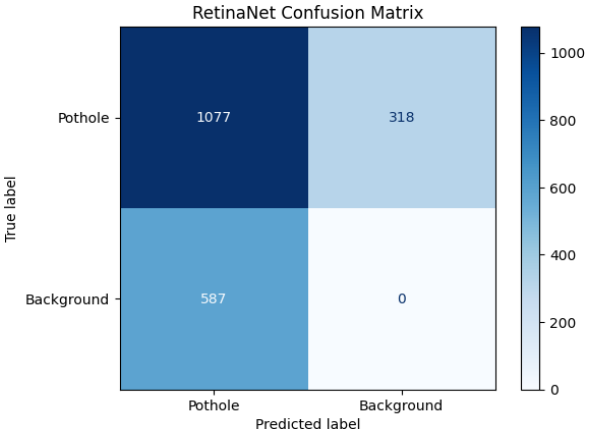

Fig 9.Confusion Matrix for RetinaNet

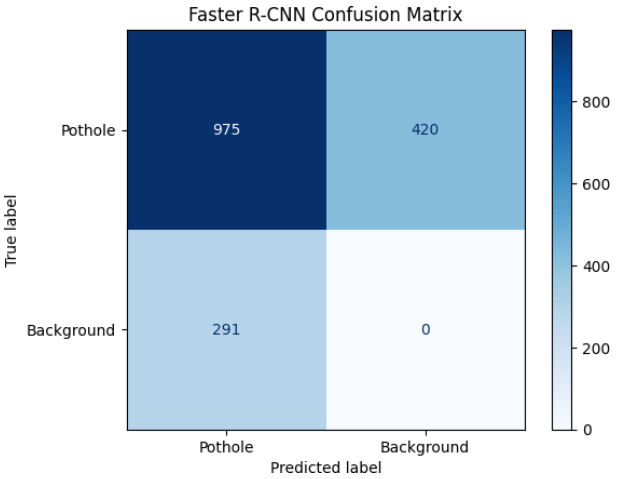

Fig 10.Confusion Matrix for Faster R-CNN

The findings suggest that false positives are caused by road textures and cracks being mistaken as potholes, whereas false negatives are mostly related to small or partially covered potholes. This is in line with the trends in recall and precision observed in models.

F. Model Reliability and Confidence Analysis.

The confidence distribution of predictions gives an understanding of the reliability of the model. The high density of predictions at the high confidence levels is a sign of a good detection ability.

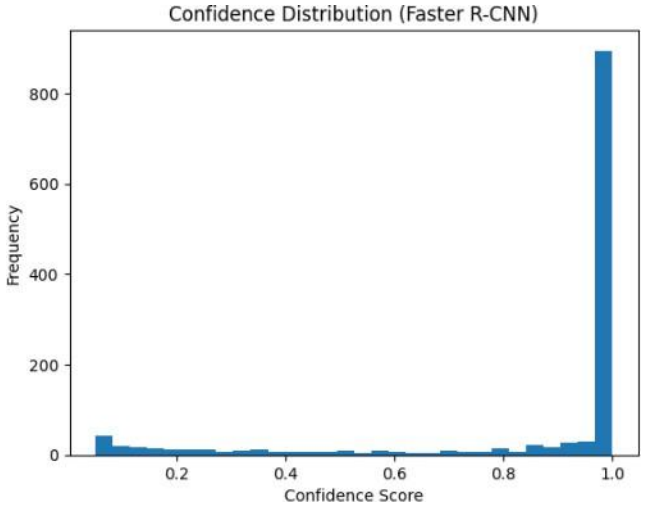

Fig 11.Confidence Score for Faster R-CNN

The distribution shows that most detections occur at high confidence values, suggesting that the models are confident in their predictions. However, the presence of lower confidence detections highlights uncertain cases, which may correspond to ambiguous or borderline pothole instances.

G. Comparative Evaluation

Table I presents the consolidated comparison of all three models.

Table I. PERFORMANCE COMPARISON

Model	Precision	Recall	F1-Score	Accuracy		mAP@0.5
Faster R-CNN	0.8278	0.9169	0.8701	0.7700		0.6165
YOLOv8	0.8375	0.6930	0.7584	0.5803		0.8066
RetinaNet	0.8109	0.8839	0.8458	0.7329		0.7168

From the results, Faster R-CNN achieves the highest recall and F1-score, making it highly effective in detecting potholes comprehensively. RetinaNet provides a balanced trade-off between detection coverage and prediction correctness. YOLOv8 achieves the highest mAP, indicating superior localisation and bounding box precision.

H. Analysis and Observations

The evaluation highlights a fundamental trade-off between precision and recall. Faster R-CNN prioritises detection completeness, YOLOv8 emphasises precision and localisation, while RetinaNet balances both aspects.

Additionally, models with higher recall tend to generate more predictions, increasing false positives and slightly reducing accuracy. Conversely, stricter models improve precision but may miss detections.

I. Conclusion of Results

Based on the comparative analysis, YOLOv8 emerges as the most suitable model for practical deployment due to its superior mAP and efficient detection performance. Faster R-CNN is more suitable for safety-critical applications where high recall is essential, while RetinaNet offers a balanced alternative.

Overall, the results demonstrate that model selection should be guided by application requirements rather than a single performance metric.

VII. FUTURE SCOPE

The proposed pothole detection framework can be further enhanced in multiple directions to improve its practical applicability and performance. One key extension involves incorporating **multi-class severity classification**, where detected potholes are categorised into Minor, Medium, and Major levels. This would enable more effective prioritisation of road maintenance activities.

Another important improvement is the integration of **geospatial information** through GPS-based geotagging, allowing the system to generate real-time pothole maps for smart city and infrastructure management applications. Additionally, optimising the model for **real-time deployment on edge devices** using techniques such as pruning, quantisation, or hardware acceleration can make the system suitable for in-vehicle or roadside deployment.

Future work can also explore **video-based detection and tracking**, which would improve detection stability across frames and reduce false positives. Expanding the dataset to include more diverse road conditions, lighting scenarios, and geographical variations will further improve model robustness. Finally, advanced architectures or ensemble approaches combining multiple models may be investigated to achieve higher detection accuracy and reliability.

VIII. CONCLUSION

This paper presented a deep learning-based framework for pothole detection and analysis using three object detection models—YOLOv8, Faster R-CNN, and RetinaNet. A consistent experimental setup was employed to ensure a fair comparison across all models using a unified dataset and evaluation metrics.

The results demonstrate that each model exhibits distinct strengths. Faster R-CNN achieves the highest recall, making it highly effective in detecting a large number of potholes. RetinaNet provides a balanced performance between precision and recall, benefiting from its focal loss mechanism. YOLOv8, however, delivers the best overall performance in terms of mAP and computational efficiency, making it the most suitable model for practical deployment.

The study also highlights the inherent trade-off between precision and recall in object detection tasks, emphasizing the importance of selecting an appropriate model based on application requirements. While high recall is crucial for safety-critical detection, high precision and efficiency are essential for real-time deployment.

Overall, YOLOv8 is identified as the most effective model for real-world pothole detection systems, particularly in intelligent transportation applications. The proposed framework lays a strong foundation for further extensions such as severity-based classification and context-aware prioritization of road maintenance.

REFERENCES

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2016.

[2] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection,” arXiv preprint arXiv:2004.10934, 2020.

[3] G. Jocher et al., “Ultralytics YOLOv8,” 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[4] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137–1149, 2017.

[5] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal Loss for Dense Object Detection,” in Proc. IEEE Int. Conf. Computer Vision (ICCV), 2017.

[6] N. Dalal and B. Triggs, “HOGs of Oriented Gradients for Human Detection,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 886–893.

[7] R. Girshick, “Fast R-CNN,” in Proc. IEEE Int. Conf. Computer Vision (ICCV), 2015.

[8] M. Everingham et al., “The Pascal Visual Object Classes (VOC) Challenge,” Int. Journal of Computer Vision, vol. 88, no. 2, pp. 303–338, 2010.

[9] T.-Y. Lin et al., “Microsoft COCO: Common Objects in Context,” in Proc. European Conf. Computer Vision (ECCV), 2014.

[10] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” arXiv preprint arXiv:1412.6980, 2014.

[11] I. Goodfellow, Y. Bengio, and A. Courville, “Deep Learning,” MIT Press, 2016.