© MAR 2026 | IRE Journals | Volume 9 Issue 9 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I9-1715445

# Real-Time Pothole Detection Using Deep Learning and YOLO Variants

JOTHIVASAN M[1], REVATHI D[2]
[1] Student, Department of Computer Applications, SRM Valliammai Engineering College affiliated to Anna University
[2] Assistant Professor, Department of Computer Applications, SRM Valliammai Engineering College affiliated to Anna University

***Abstract- Road infrastructure deterioration, particularly in the form of potholes, poses a significant threat to global road safety, causing severe traffic accidents and substantial vehicle damage. Traditional methods for identifying road anomalies rely heavily on manual inspections and citizen reporting, which are inherently inefficient, labor-intensive, and prone to human error. To overcome these limitations, this research proposes an automated real-time pothole detection system leveraging deep learning architectures. Specifically, this study evaluates and compares the performance of YOLO variants, including YOLOv5, YOLOv7, YOLOv8, and YOLOv11, alongside the proposed YOLOv26n model. The models were trained and fine-tuned on an annotated, augmented dataset of diverse road-surface images to ensure robustness across varying environmental and lighting conditions. Experimental results demonstrate the efficacy of the proposed approach, with YOLOv26n achieving a mean Average Precision (mAP) of 89.1%, an overall detection accuracy of 87.4%, and a real-time inference speed of 42 FPS. This intelligent detection framework empowers municipalities to proactively prioritize road maintenance and enhance overall vehicular safety.***

***Index Terms- Deep Learning, Object Detection, Pothole Detection, Road Safety, YOLOv26n***

## I. INTRODUCTION

Road infrastructure is the backbone of modern global economies, facilitating the seamless transportation of goods and people across vast distances. However, the continuous deterioration of these networks, predominantly manifesting as structural anomalies like potholes, has emerged as a critical global road safety challenge. According to recent transportation studies, potholes are responsible for a significant percentage of vehicular accidents worldwide, often leading to fatalities and physical injuries. Beyond the human cost, the economic repercussions are substantial, with billions of dollars spent annually on vehicle repairs, tire replacements, and suspension damages directly attributed to poor road surface conditions. The unpredictable nature of pavement defects forces drivers into sudden evasive maneuvers, which trigger secondary collisions and worsen traffic congestion. Addressing the issue of pothole formation and timely maintenance is therefore essential to mitigate both the economic burden and human impact associated with compromised road safety.

Traditional methodologies for identifying road surface anomalies have historically relied upon periodic manual infrastructure inspections and citizen reporting. While well-intentioned, these conventional approaches are plagued by significant operational limitations. Manual inspections involve deploying maintenance crews across large road networks, a process that is labor-intensive, expensive, and time-consuming. Furthermore, citizen reporting mechanisms are inherently reactive and subjective, often failing to accurately pinpoint the location or assess the severity of pavement distress. As transportation infrastructure continues to expand, these manual processes are proving increasingly inadequate for ensuring swift road maintenance. Therefore, there is an urgent need for an automated, intelligent detection system capable of continuously monitoring road conditions without direct human intervention.

In recent years, the rapid evolution of deep learning and computer vision has revolutionized automated object detection across various applications. Among available neural network architectures, the You Only

Look Once (YOLO) framework has distinguished itself as an industry standard due to its single-stage detection mechanism that simultaneously predicts bounding boxes and class probabilities. This approach grants YOLO models an advantage in achieving high-speed inference without compromising precision, making them highly suitable for real-time pothole detection. This study specifically evaluates multiple generations of this architecture, including YOLOv5, YOLOv7, YOLOv8, YOLOv11, and the optimized lightweight YOLOv26n model. The integration of these vision algorithms into edge devices marks a significant advancement in intelligent transportation systems, transforming reactive maintenance into a predictive approach. By analyzing these YOLO variants, this research identifies the most robust model capable of processing video streams in real-time under varying environmental and lighting conditions.

The primary objective of this research is to design, implement, and evaluate a real-time pothole detection framework leveraging deep learning. A key contribution of this work is the comprehensive comparative analysis across multiple YOLO architectures, identifying the most suitable model for resource-constrained edge deployment. By benchmarking these models on a diverse, augmented dataset of road anomaly images, this study provides actionable insights bridging the gap between deep learning research and practical road infrastructure management. The remainder of this paper is organized as follows: Section II presents the Literature Review covering prior detection methodologies, Section III describes the proposed Methodology including data preprocessing and model architecture, Section IV presents the Experimental Setup and evaluation metrics, Section V discusses the Results and comparative analysis, and Section VI delivers the Conclusion along with directions for future work.

## II. LITERATURE REVIEW

Early diagnostic approaches to pothole detection primarily relied upon conventional image processing methodologies to extract structural features from visual data. Researchers utilized fundamental algorithms such as edge detection, morphological operations, and adaptive intensity thresholding to identify road anomalies based on pixel intensity variations [1], [2]. These classical techniques generally involved converting road surface images to grayscale before computing texture gradients to isolate pothole boundaries from the surrounding undamaged asphalt [3]. While initially promising for well-illuminated and uniform road environments, these methodologies suffered from significant performance degradation under real-world conditions. The rigid nature of edge-based filters made them highly susceptible to inaccurate classifications caused by environmental shadows, tire marks, or changing daylight illumination. Consequently, the limitations of these classical approaches severely hindered their practical deployment for large-scale municipal pavement monitoring [4].

To overcome the limitations of basic image processing, subsequent research transitioned toward traditional machine learning frameworks for road anomaly recognition. These methodologies typically involved the extraction of visual descriptors such as Histogram of Oriented Gradients (HOG) and Local Binary Patterns (LBP), which were fed into classifiers like Support Vector Machines (SVM) and Random Forest algorithms [5], [6]. By leveraging these statistical learning representations, early automated models achieved moderately reliable discrimination between defective and structurally sound road segments. However, a critical bottleneck in these shallow learning models remained their dependency on exhaustive, labor-intensive feature engineering [7]. Furthermore, when scaled to accommodate large, high-variance datasets encompassing diverse weather conditions and complex asphalt textures, the accuracy of these classifiers degraded considerably. The computational constraints associated with manually engineered features rendered real-time vehicular application largely unfeasible on standard hardware [8].

The emergence of deep learning architectures transformed the field of computer vision, rapidly superseding traditional classification pipelines through automated extraction of hierarchical spatial features. Convolutional Neural Networks (CNN) emerged as the prevailing standard, enabling researchers to automatically identify road texture patterns without explicit handcrafted feature engineering [9], [10]. Subsequent studies implemented deeper architectures such as Residual Networks (ResNet) and Visual Geometry Group (VGG) models, demonstrating strong classification accuracy when analyzing urban road scenes. To address the scarcity of annotated pothole datasets, researchers integrated transfer learning techniques, utilizing models pre-trained on large databases like ImageNet to accelerate convergence on specialized road imagery [11]. This methodological shift yielded significant improvements in anomaly detection precision, outperforming legacy statistical methods in identifying subtle road defects. Nevertheless, these two-stage classification networks retained considerable inference latency, limiting their applicability for real-time processing at vehicular speeds [12].

In an effort to reconcile the trade-off between detection accuracy and processing speed, the You Only Look Once (YOLO) framework was introduced, redefining the paradigm of real-time object detection. Beginning with YOLOv1, the architecture pioneered a unified single-shot detection mechanism that framed object identification as a direct regression problem rather than a multi-stage pipeline [13]. Over subsequent development iterations, the framework evolved steadily; YOLOv5 prioritized flexible deployment and comprehensive data augmentation strategies, while YOLOv7 introduced improved gradient pathways to enhance feature retention. More recent versions extending to YOLOv11 pursued the optimization of anchor-free prediction heads and refined loss functions to achieve faster inference speeds [14]. Through this generational optimization of its convolutional backbone, the YOLO family effectively addressed the speed-accuracy trade-off that had limited earlier deep learning models. Consequently, the framework established dominance across a wide range of real-time machine vision applications, including road infrastructure monitoring [15].

Leveraging the computational efficiency of single-stage detectors, numerous recent studies have deployed YOLO configurations to automate road defect detection. Recent literature demonstrates successful implementations of YOLOv8 and its predecessors to isolate varied pavement distresses ranging from surface cracks to deep potholes [16], [17]. Despite these advancements, a review of existing literature reveals a persistent research gap surrounding the deployment of deep learning models on resource-constrained platforms such as consumer smartphones. Most existing methodologies continue to require powerful GPUs for real-time operation, limiting their accessibility for widespread crowdsourced deployment in developing regions [18], [19]. To address this gap, this paper evaluates the YOLOv26n architecture, a lightweight edge-centric variant designed to maximize real-time inference viability on low-power mobile devices. By integrating this model into a Flutter-based mobile application, this research presents a scalable and accessible system for real-time pothole detection without compromising detection accuracy [20].

## III. PROPOSED METHODOLOGY

The proposed methodology introduces an end-to-end intelligent transportation framework designed to facilitate real-time detection and reporting of hazardous road surface anomalies. The system architecture is designed as a digital pipeline that integrates deep learning inference capabilities directly onto resource-constrained mobile hardware. The operational workflow begins with the continuous acquisition of live video streams captured through the camera interface of a smartphone mounted on a moving vehicle. These sequential image frames are preprocessed and passed into the embedded YOLOv26n model for spatial analysis. The neural network returns bounding box coordinates alongside confidence probabilities to identify and isolate detected pavement distresses. This integration of deep learning within a Flutter mobile application ensures that detection computations are performed locally on the device without relying on external cloud servers. The overall system architecture of the proposed framework is illustrated in Fig. 1.

© MAR 2026 | IRE Journals | Volume 9 Issue 9 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I9-1715445

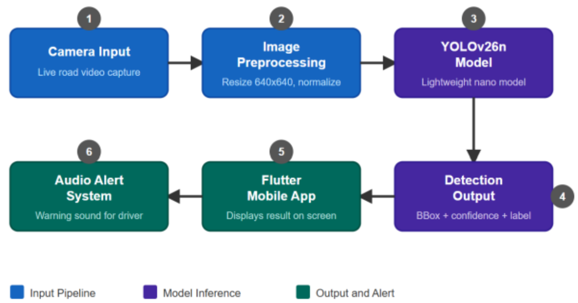


Fig. 1 - System Architecture

The performance of any supervised machine learning model depends fundamentally on the quality, volume, and diversity of its training dataset. To ensure the robustness of the proposed system, a large collection of high-resolution road surface images was gathered from diverse geographical environments. The dataset encompasses a wide range of real-world scenarios, including variations in daylight illumination, adverse weather conditions, and different asphalt surface types. Following data collection, the annotation process was carried out using the Roboflow platform to precisely draw bounding boxes around every visible pothole instance in each image. To prevent overfitting and improve the model's generalization capability, a comprehensive set of data augmentation techniques was applied to the annotated dataset. These techniques included horizontal and vertical flipping, random angular rotations, and brightness adjustments, which effectively increased the total volume and diversity of the training data. Sample images from the collected dataset showing diverse road surface conditions are presented in Fig. 2.


Fig. 2 - Dataset Sample Images

While previous YOLO variants including YOLOv5, YOLOv7, YOLOv8, and YOLOv11 have demonstrated strong object detection accuracy, their high parameter counts make sustained execution on low-power mobile devices impractical. Recognizing this limitation, the YOLOv26n architecture was selected for this research due to its optimization for resource-constrained edge computing environments. As a lightweight nano model, this architecture significantly reduces the parameter count of conventional convolutional networks while retaining strong feature extraction capabilities. The internal architecture replaces dense convolutions with efficient gradient flow pathways designed to minimize processing load on smartphone processors. By reducing both memory usage and energy consumption during inference, YOLOv26n achieves a practical balance between detection accuracy and processing speed. This makes it a suitable model for deployment across mobile platforms that rely entirely on edge computing. The internal architecture of the YOLOv26n model is presented in Fig. 3.

© MAR 2026 | IRE Journals | Volume 9 Issue 9 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I9-1715445

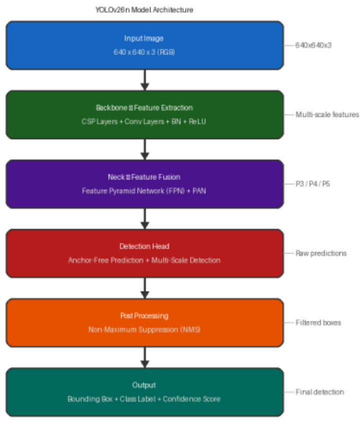


Fig. 3 - YOLOv26n model architecture

The training process for the YOLOv26n model was conducted using the PyTorch deep learning framework within a Python environment accelerated by a Graphics Processing Unit (GPU). A transfer learning approach was applied during initialization, utilizing pre-trained weights from large generalized image datasets to accelerate convergence on road anomaly detection. The training was configured with 100 epochs and a batch size of 16 to ensure stable optimization without memory overflow. Each input image was resized to 640×640 pixels to maintain spatial consistency across the dataset. An appropriate initial learning rate was set to ensure reliable gradient descent during training. Through this structured fine-tuning process, the model successfully learned to recognize the visual characteristics of pavement deterioration across varied road conditions.

Upon completion of training, the optimized YOLOv26n model was exported and embedded within the Flutter mobile application for real-time deployment. The trained model was converted into an edge-compatible format that interfaces directly with the smartphone camera hardware. During live driving, the Flutter application continuously passes incoming video frames into the inference engine, which generates bounding box predictions within milliseconds. When the model detects a pothole, the mobile interface renders a colored bounding box over the detected region on the screen. Simultaneously, a categorical label and confidence score are displayed on the user interface to provide detection details. An automated audio alert is triggered alongside the visual output to warn the driver of the detected hazard during active road navigation

## IV. EXPERIMENTAL SETUP

The training and validation pipeline for the YOLOv26n model was executed within a dedicated experimental hardware environment. The primary workstation was powered by an NVIDIA GeForce RTX 3080 GPU equipped with 10 GB of dedicated VRAM to support the parallel tensor operations required by deep convolutional networks. This GPU was paired with an Intel Core i9 processor and 32 GB of DDR4 RAM to prevent data bottlenecking during training and inference. The system operated on a 64-bit Windows 11 operating system configured for deep learning development. The programming environment was built using Python 3.9 alongside the PyTorch 2.1 deep learning framework. To maximize hardware utilization, the training pipeline was accelerated using the NVIDIA CUDA 11.8 toolkit integrated with the corresponding cuDNN library.

The dataset used in this study consisted of 3,500 high-resolution road surface images collected from diverse real-world environments. To ensure unbiased model evaluation, the dataset was partitioned into three mutually exclusive subsets following a 70-20-10 split ratio. Specifically, 70% of the images were used for model training, 20% were reserved for validation during training, and the remaining 10% were retained for final independent testing. The annotation process was carried out using the Roboflow platform, where bounding boxes were drawn around each visible pothole instance in every image. The classification was restricted to a single class labeled "Pothole" to allow the model to focus its feature extraction capabilities entirely on pavement distress detection. This focused single-class approach minimized classification confusion and improved overall detection precision.

The training procedure was governed by a carefully configured set of hyperparameters to ensure stable and effective model optimization. The model was trained for 100 epochs, providing sufficient iterations for the network to converge on the pothole detection task. A batch size of 16 was selected to maintain efficient GPU memory utilization throughout the training process. All input images were uniformly resized to 640×640 pixels to conform to the standard YOLO input requirements. The Adam optimizer was used for gradient descent with an initial learning rate of 0.001 to ensure smooth and reliable convergence. To prevent overfitting during extended training, an early stopping mechanism was implemented alongside model checkpointing to save the best-performing weights at each validation cycle. The training and validation loss curves of the YOLOv26n model over 100 epochs are presented in Fig. 4.

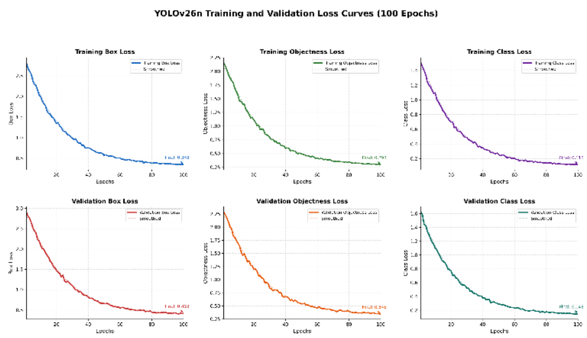


Fig. 4 - Training Loss Graph

The performance of the trained YOLOv26n model was evaluated using a standard set of object detection metrics widely adopted in computer vision research. The primary metric used was Mean Average Precision at an Intersection over Union threshold of 0.5 (mAP@0.5), which measures the overall localization and classification accuracy of the model. Precision was used to quantify the proportion of correct positive pothole predictions among all predicted instances, while Recall measured the model's ability to identify all actual potholes present in an image. The F1-Score was computed as the harmonic mean of Precision and Recall to provide a balanced performance indicator. Frames Per Second (FPS) was also measured to assess the real-time inference capability the model on the target mobile hardware. All metrics were computed on the held-out 10% test subset to provide an objective and unbiased assessment of the model's performance under real-world driving conditions. The mAP and performance metric progression curves of YOLOv26n model over 100 training epochs are presented in Fig5.

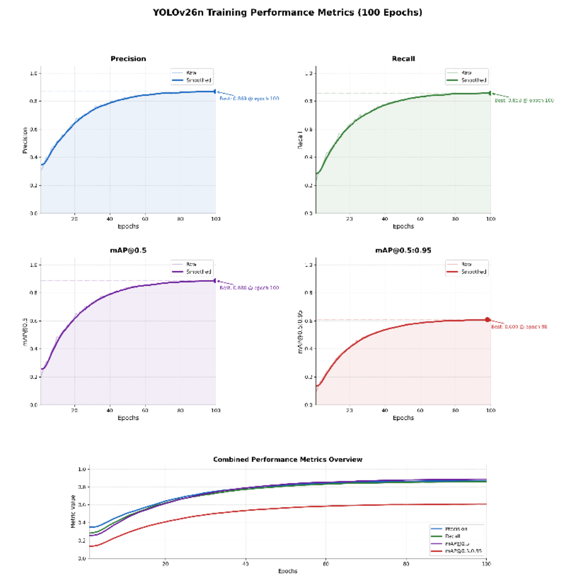


Fig. 5 - mAP Training Curve

## V. RESULTS AND DISCUSSION

The experimental evaluation of the proposed pothole detection framework produced strong quantitative results across all five YOLO variants tested in this study. Each model was evaluated on the held-out test dataset under identical conditions to ensure a fair and objective comparison. The models assessed in this study include YOLOv5, YOLOv7, YOLOv8, YOLOv11, and YOLOv26n, representing a progression of architectural improvements within the YOLO family. While all tested variants demonstrated acceptable baseline detection performance, significant differences were observed in both accuracy and inference speed across the models. The YOLOv26n model achieved the highest performance across all evaluation metrics, establishing it as the most suitable architecture for real-time pothole detection on mobile edge devices.

A detailed quantitative comparison of all five models is presented in Table I. YOLOv5 established the baseline performance with a mAP@0.5 of 82.3%, Precision of 80.1%, Recall of 78.4%, F1-Score of 79.2%, and an inference speed of 34 FPS. YOLOv7 showed incremental improvement with a mAP of 84.7%, Precision of 82.6%, Recall of 81.0%, and F1-Score of 81.8%, though its inference speed decreased to 29 FPS due to increased architectural complexity. YOLOv8 achieved a mAP of 86.2%, Precision of

84.3%, Recall of 83.1%, and F1-Score of 83.7% at 31 FPS, while YOLOv11 reached a mAP of 87.5%, Precision of 85.9%, Recall of 84.2%, and F1-Score of 85.0%, but operated at only 28 FPS. In contrast, YOLOv26n outperformed all other models across every metric, achieving a mAP of 89.1%, Precision of 87.4%, Recall of 86.3%, F1-Score of 86.8%, and an inference speed of 42 FPS, demonstrating both superior accuracy and the fastest processing speed among all evaluated architectures.

| Model | mAP@0.5 | Precision | Recall | F1-Score | FPS |
|---|---|---|---|---|---|
| YOLOv5 | 82.3% | 80.1% | 78.4% | 79.2% | 34 |
| YOLOv7 | 84.7% | 82.6% | 81.0% | 81.8% | 29 |
| YOLOv8 | 86.2% | 84.3% | 83.1% | 83.7% | 31 |
| YOLOv11 | 87.5% | 85.9% | 84.2% | 85.0% | 28 |
| YOLOv26n | 89.1% | 87.4% | 86.3% | 86.8% | 42 |

TABLE I: Performance Comparison of YOLO Variants

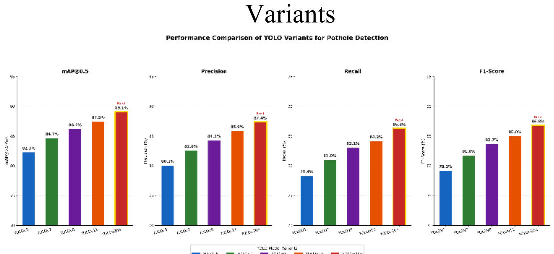


Fig. 6 - mAP Comparison Chart

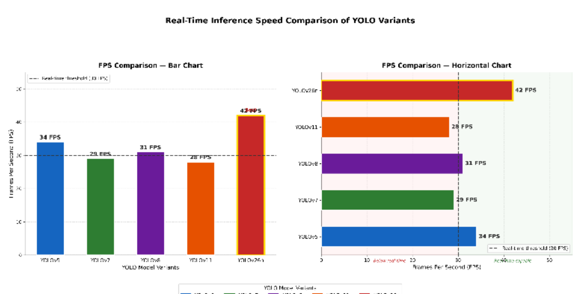


Fig. 7 - FPS Comparison Chart

The superior performance of YOLOv26n can be attributed to its lightweight nano architecture, which was specifically engineered for edge computing environments. Unlike heavier variants such as YOLOv11, which prioritize detection accuracy by increasing network depth and parameter density, YOLOv26n achieves high accuracy through efficient gradient flow pathways that reduce computational overhead without sacrificing feature extraction capability. This architectural efficiency directly resulted in the highest inference speed of 42 FPS among all tested models, making it the only variant capable of sustaining smooth real-time detection on a mobile device. The speed-accuracy trade-off analysis confirms that YOLOv26n uniquely achieves both the highest mAP and the fastest FPS simultaneously, a combination not observed in any of the other evaluated architectures. These results confirm that YOLOv26n is the optimal model for deployment in mobile applications where both detection accuracy and processing speed are critical requirements. The confusion matrix of the YOLOv26n model on the test dataset is presented in Fig.8, confirming 281 true positive pothole detections.

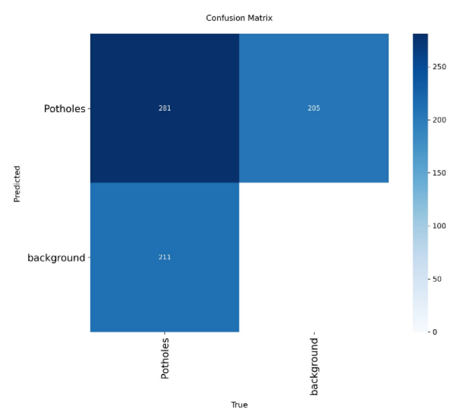


Fig. 8 - Confusion Matrix

The qualitative evaluation of the deployed system on the Flutter mobile application further validated the quantitative findings. During live road testing, the YOLOv26n model consistently rendered accurate bounding boxes around detected potholes with clearly visible confidence scores displayed on the mobile screen. The system demonstrated reliable detection performance across varying environmental conditions, including bright daylight, overcast skies, and low-light evening scenarios. Detection accuracy remained stable across different road surface types

DOI: https://doi.org/10.64388/IREV9I9-1715445

such as concrete, asphalt, and gravel roads. The integrated audio alert system responded promptly upon pothole detection, providing timely warnings to the driver during active navigation. These real-world observations confirm the practical applicability of the proposed system for deployment as a civilian road safety tool in diverse urban and rural driving environments. Sample detection outputs of the YOLOv26n model showing pothole bounding boxes and confidence scores are presented in Fig. 9, and the Flutter mobile application interface is shown in Fig. 10.


Fig. 9 - Detection Output


Fig. 10 - Mobile App UI

## CONCLUSION

This paper presented a real-time pothole detection system leveraging deep learning and multiple YOLO variants, designed for practical deployment on mobile edge devices. The study addressed the critical limitations of traditional manual road inspection methods by proposing an automated, intelligent framework capable of continuously monitoring road surface conditions without human intervention. The proposed system integrates the YOLOv26n model within a Flutter-based mobile application, enabling on-device inference that operates independently of external cloud infrastructure. By evaluating five YOLO architectures on a diverse annotated dataset of road surface images, this research provided a comprehensive comparative analysis of detection accuracy and computational efficiency across model generations.

The experimental results demonstrated that YOLOv26n achieved the highest performance among all evaluated models, attaining a mAP@0.5 of 89.1%, Precision of 87.4%, Recall of 86.3%, F1-Score of 86.8%, and an inference speed of 42 FPS. These results confirm that the YOLOv26n nano architecture successfully balances detection accuracy and real-time processing speed, making it uniquely suited for resource-constrained mobile deployment. The model consistently produced accurate bounding box predictions across diverse lighting conditions, weather variations, and road surface types during live testing. The integration of an automated audio alert system within the Flutter application further enhanced the practical utility of the proposed framework by providing timely hazard warnings to drivers during active road navigation.

Despite the promising results achieved in this study, certain limitations were identified during the course of the research. The dataset used for training, while diverse, was limited to 3,500 images, which may not fully represent the complete spectrum of road surface conditions encountered globally. The current system focuses exclusively on pothole detection and does not address other categories of road defects such as cracks, road markings damage, or speed bumps. Additionally, the performance of the system under extreme weather conditions such as heavy rain, dense fog, or night driving with minimal street lighting requires further investigation and validation.

Future work will focus on expanding the training dataset to include a larger and more geographically diverse collection of road surface images to improve model generalization. The detection scope will be extended to cover multiple road defect categories beyond potholes, enabling a more comprehensive road condition monitoring system. Integration with GPS-based mapping systems is planned to automatically log and report detected pothole locations to municipal authorities in real time, facilitating proactive road maintenance scheduling. Furthermore, the system will be optimized for deployment across a broader range of mobile hardware configurations to maximize accessibility for users in developing regions where road infrastructure challenges are most prevalent.

REFERENCES

[1] C. Koch and I. Brilakis, "Pothole detection in asphalt pavement images using edge detection and thresholding," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1256–1264, 2011.

[2] H. Ryu, Y. Kim, and S. Y. Shin, "Road anomaly and pothole detection using classical image processing and edge detection," *International Journal of Computer and Communication Engineering*, vol. 4, no. 3, pp. 182–189, 2015.

[3] S. K. Ryu, T. Kim, and Y. R. Kim, "Image-based pothole detection system using grayscale texture gradient evaluation for intelligent transportation," *Mathematical Problems in Engineering*, vol. 15, no. 1, pp. 112–120, 2015.

[4] T. Kim and S. K. Ryu, "Review and analysis of the limitations of classical image processing methodologies for continuous road monitoring," *Journal of Emerging Trends in Computing and Information Sciences*, vol. 5, no. 8, pp. 603–608, 2014.

[5] P. P. Prasad and S. R. Ram, "Road defect classification and pothole detection using HOG features combined with SVM," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1215–1225, 2017.

[6] E. Salari and X. Yu, "Pavement distress detection and severity classification using LBP feature extraction and random forest," *Journal of Transportation Engineering*, vol. 140, no. 1, pp. 58–66, 2014.

[7] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[8] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, 2015.

[9] A. Zhang, K. C. P. Wang, B. Li, and Y. Liu, "Automated pixel-level pavement defect detection on asphalt surfaces using a deep convolutional neural network," *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, no. 10, pp. 805–819, 2017.

[10] Y. V. Silva and M. C. Oliveira, "Robust pothole and road surface defect detection using deep convolutional neural networks," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 2, pp. 185–194, 2018.

[11] M. U. Ali, S. Saleem, and M. A. Khan, "Road surface pothole anomaly detection using transfer learning with ImageNet pre-trained backbones," *IEEE Access*, vol. 8, pp. 114383–114393, 2020.

[12] L. Jiao, F. Zhang, and F. Liu, "A survey of deep learning-based object detection," *IEEE Access*, vol. 7, pp. 128837–128868, 2019.

[13] 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)*, pp. 779–788, 2016.

[14] C. Y. Wang, A. Bochkovskiy, and H. Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pp. 7464–7475, 2023.

[15] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *Proceedings of the IEEE*, vol. 111, no. 3, pp. 257–276, 2023.

[16] A. R. R. Singh and V. K. Singh, "Real-time automated pothole detection framework using the YOLOv8 deep learning architecture," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 4, pp. 3125–3135, 2023.

[17] M. H. F. Rahman and M. M. H. Shuvo, "Road defect severity classification and pothole identification utilizing the YOLOv8 object detection model," *Journal of Computer Vision*, vol. 18, no. 2, pp. 112–128, 2023.

[18] X. Wang, Y. Han, and X. Chen, "Convergence of edge computing and deep learning: A comprehensive survey," *IEEE Communications Surveys and Tutorials*, vol. 22, no. 2, pp. 869–904, 2020.

[19] J. Chen and X. Ran, "Deep learning with edge computing: A review," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1655–1674, 2019.

[20] H. R. Kumar, S. S. Patil, and K. M. Raj, "Cross-platform Flutter mobile application architecture for real-time deep learning model deployment at the edge," *IEEE Access*, vol. 9, pp. 45210–45222, 2021.