

Real-Time Pothole Detection on Roads Using Deep Learning and the YOLOv8(You Only Look Once, Version 8) Framework

Beena Sheril¹, Ramesh B E², Anu. H S³, Divya U⁴, Kusuma M⁵ & Madhura G M⁶

^{1&2} Associate Professor, Department of Computer Science and Engineering

SJM Institute of Technology (SJMIT), Chitradurga, Karnataka, India Email: be.ramesh@gmail.com

^{3, 4,5 & 6} B.E. Final Year Students, Department of Computer Science and Engineering, SJMIT, Chitradurga, Karnataka, India

Abstract— Potholes are one of the most common reasons behind vehicle damage, traffic jams and road accidents, and the usual practice of inspecting roads by hand is slow, costly and often inaccurate. In this paper we present an automated pothole detection system that uses deep learning, built around the YOLOv8 (You Only Look Once, Version 8) object detection model. Road images and video frames collected from vehicle-mounted cameras, roadside cameras and drones are first resized, cleaned of noise and normalized, and are then given to the trained model, which marks every pothole with a bounding box and a confidence score. The detections are stored in a database, alerts and reports are generated, and the pothole information is passed on to a road maintenance module for repair scheduling and location mapping. A feedback loop allows the system to improve with newly collected road data. The developed model achieved a precision above 92% and works at about 35 FPS (Frames Per Second), which is sufficient for real-time use. The work is meant to support smart city and ITS (Intelligent Transportation Systems) applications in maintaining roads in a proactive way.

Index Terms— Pothole detection, deep learning, YOLOv8, CNN (Convolutional Neural Network), computer vision, intelligent transportation systems, road safety.

I. INTRODUCTION

Roads are the backbone of daily life and economic growth, but damaged roads have become a common sight in both urban and rural areas. Among all road defects, potholes are the most troublesome. They are formed because of heavy traffic load, poor drainage, changing weather and delayed maintenance, and they lead to vehicle damage, traffic delays and serious accidents. Municipal authorities find it difficult to locate and repair potholes in time, which increases the maintenance cost as well as the risk to the public.

In most places pothole inspection is still done manually by road maintenance staff. This needs a lot of manpower, time and money, and the results are not always accurate or up to date. Sensor-based approaches that use accelerometers, GPS (Global Positioning System) and vibration sensors in vehicles have also been tried, but they are costly, cover only the roads the instrumented vehicles travel on, and can give false readings depending on vehicle speed and movement.

With the growth of AI (Artificial Intelligence), deep learning and computer vision, it has become possible to monitor road surfaces automatically from images and video. Object detection models based on CNN (Convolutional Neural Network) can learn the visual features of a pothole directly from data, without manual feature design. Among them,

YOLO (You Only Look Once) [4] looks at the whole image in a single pass and predicts bounding boxes and class probabilities together, which makes it very fast and suitable for real-time work.

In this work we developed a pothole detection system using YOLOv8, trained on road images that we collected and annotated ourselves. The system captures road images and videos, preprocesses them, detects potholes, and produces alerts and reports for road maintenance authorities. The main contributions of this paper are: (i) a complete pipeline from image capture to maintenance reporting, (ii) a YOLOv8 based detection model that gives precision above 92 percent at about 35 FPS, and (iii) a clear architecture that can be integrated with smart city platforms.

II. LITERATURE REVIEW

Research on automatic pothole detection has moved through three stages. The early works used classical image processing such as edge detection, thresholding, Hough transform and texture analysis to find irregularities on the road surface. These methods are simple and computationally cheap, but they work well only in controlled conditions and fail in the presence of shadows, changing light and complex road textures.

The second stage used machine learning classifiers such as SVM (Support Vector Machine), Random Forest and K-means clustering on hand-crafted features. They perform better than plain image processing but depend heavily on feature engineering and on the quality and quantity of the training data. Sensor-based systems built on accelerometers and vibration sensors belong to this stage as well.

The third stage, which is the current one, is deep learning. CNN based detectors such as Faster R-CNN (Region-based Convolutional Neural Network) [5], SSD (Single Shot MultiBox Detector) [6] and the YOLO family [4], [7] learn features directly from images and give higher accuracy with real-time detection speed. Recent studies on pothole detection using deep learning methods [1]–[3] report that such models are effective in reducing traffic accidents, and some of them also use drone imaging to cover large road networks quickly. Table I gives a comparison of the approaches.

TABLE I. COMPARISON OF POTHOLE DETECTION APPROACHES

Approach	Accuracy	Speed	Cost	Limitation
Image processing	Moderate	Fast	Low	Sensitive to environment
Machine learning (SVM, RF)	Good	Moderate	Mode rate	Needs feature engineering
Deep learning (CNN)	High	Moderate	High	Computationally expensive
YOLO based detection	Very high	Real-time	Mode rate	Needs large annotated dataset

From the table it is clear that YOLO based detection gives the best balance between accuracy and speed, and this is the reason we selected it for our work. The literature also shows some gaps that are still open: most datasets are small and limited to one region, detection becomes weak at night, in rain and in the presence of shadows, very few systems are linked with municipal maintenance departments or with GPS and IoT (Internet of Things) platforms, and many models need GPU (Graphics Processing Unit) hardware that raises the deployment cost. Our work tries to address these gaps with a lightweight and integrated pipeline.

III. PROBLEM STATEMENT AND OBJECTIVES

Manual pothole detection is slow and prone to human error, sensor based methods are expensive and limited in coverage, and the lack of real-time detection delays repairs and causes accidents. A usable system must detect potholes of different shapes, sizes and depths under different lighting and weather, must work in real time on moving vehicles, must keep the computational load low enough for resource-constrained devices, and must give useful information to the authorities so that repairs can be prioritized. Hence there is a need for an automated, cost-effective and accurate pothole detection system based on deep learning.

The objectives of this work are: (1) to train and optimize a YOLO based model that detects potholes under varying road, light and weather conditions; (2) to create a custom dataset of annotated road surface images; (3) to achieve fast detection suitable for vehicles in motion; (4) to evaluate the model using precision, recall and mAP (mean Average Precision); (5) to provide a scalable solution that can be fitted into intelligent transportation and smart city systems; and (6) to improve road safety through early detection and timely repair.

IV. PROPOSED SYSTEM ARCHITECTURE

The proposed system captures road images and videos, preprocesses them, detects potholes using YOLOv8, and generates alerts and reports for the road maintenance authorities. The overall architecture is shown in Fig. 1, and its components are described below.

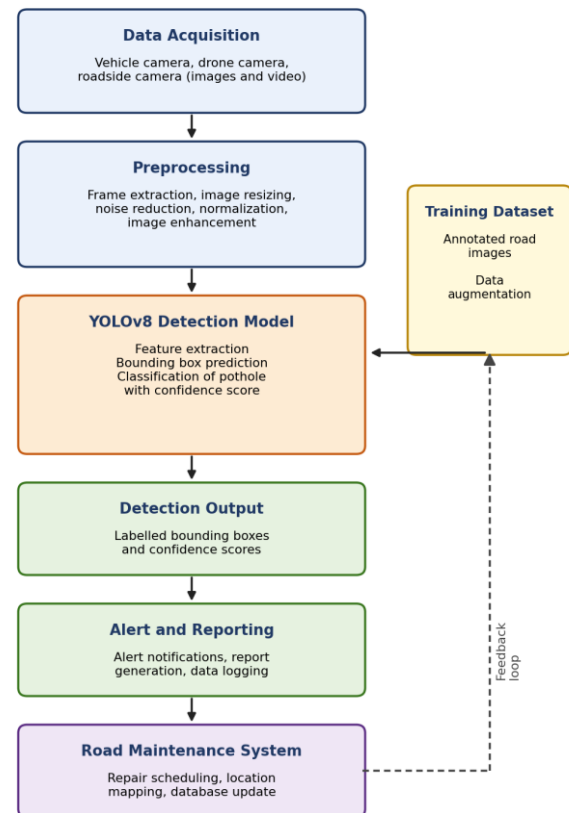


Fig. 1. Architecture of the proposed pothole detection system.

A. Input Module

The process begins with the capture of road images and video streams using vehicle-mounted cameras, roadside cameras and drone cameras. These devices keep monitoring the road surface and provide the visual data for detection. Video is converted into individual frames before further processing.

B. Data Preprocessing

The captured data is processed to improve image quality and detection accuracy. Images are resized to a fixed dimension for faster processing, noise is reduced to remove unwanted disturbances, and pixel values are normalized so that the model receives standardized input. Image enhancement is also applied to improve clarity.

C. YOLOv8 Detection Model

The preprocessed images are passed to the YOLOv8 model. The model extracts features related to potholes, predicts the bounding box that locates each pothole, and classifies the detected object as a pothole with a confidence score. As the whole image is processed in a single forward pass, detection is fast enough for real-time use. The model is trained on the annotated training dataset, which is enlarged with data augmentation so that it can handle different shapes, sizes and lighting conditions.

D. Detection Output

After detection, the system displays the pothole with a labelled bounding box together with its confidence score. Only the detections above a defined confidence threshold, such as 90 percent, are treated as confirmed, which helps in reducing false positives.

E. Alert and Reporting

When a pothole is detected, the system sends alert notifications, generates pothole detection reports and logs the detection records and road condition data for the authorities and users.

F. Road Maintenance System

The detected pothole information is integrated with the road maintenance system. This supports repair scheduling, location mapping using GPS or map services, and updating of the pothole and maintenance databases.

G. Feedback Loop

The feedback loop continuously updates the system with new road data and maintenance information. This helps in improving detection accuracy, monitoring efficiency and the performance of future models.

H. Data Flow and Interfaces

At Level 0 of the data flow, the user gives input through a camera to the system and receives the output. At Level 1, the input (image or video) goes through preprocessing and YOLO detection, the result (pothole or no pothole) is stored in the database, and maintenance alerts are produced. The system has four interfaces. The user interface is a dashboard that shows road images, bounding boxes, confidence scores, alerts and reports in real time. The system interface manages the communication between modules and handles the image data flow, preprocessing and model processing. The database interface stores detection records, images, timestamps, confidence scores and maintenance information in MongoDB or PostgreSQL. The output interface produces reports, alerts and visual outputs for users and municipal authorities, and connects to repair scheduling and smart city systems.

V. IMPLEMENTATION AND SYSTEM REQUIREMENTS

The system is developed in Python because of its flexibility and the wide range of libraries available for deep learning. PyTorch is used for model training and inference, TensorFlow Lite for lightweight mobile and edge inference, OpenCV (Open Source Computer Vision Library) for image processing, and NumPy and Pandas for data handling. Labellmg and Roboflow are used to annotate the pothole regions with bounding boxes. Model training is done on a GPU enabled workstation or on platforms such as Google Colab and Kaggle. Table II summarizes the main requirements.

TABLE II. SUMMARY OF SYSTEM REQUIREMENTS

Category	Requirement
Hardware (minimum)	Intel i5 processor (8th generation or above), 8 GB RAM, 256 GB SSD, NVIDIA GTX 1050 GPU with 4 GB VRAM
Hardware (recommended)	Intel i7 or i9 processor, 16 to 32 GB RAM, 512 GB SSD, NVIDIA RTX 3060 or 3080 GPU with 8 to 12 GB VRAM
Input devices	Vehicle-mounted cameras, roadside cameras, drones and mobile cameras
Operating system	Windows 10 or 11 (64-bit), or Ubuntu 20.04 LTS (Long Term Support) and later
Frameworks	Python 3.10 or above, PyTorch 2.1, TensorFlow Lite 2.x, OpenCV

Database	MongoDB Atlas and PostgreSQL
Backend and frontend	FastAPI and Node.js with Express.js; React.js with Tailwind CSS and Mapbox API for map display
Security	Firebase Authentication and JWT (JSON Web Token); HTTPS (Hypertext Transfer Protocol Secure) and SSL (Secure Sockets Layer)
Deployment	Git, GitHub Actions and Docker; AWS (Amazon Web Services) IoT Core, S3 and Lambda for cloud integration

The database holds three kinds of records: the detection log (image identity, timestamp, confidence score and location), the maintenance record (pothole identity, repair status and assigned personnel) and the user information (operator credentials and access levels). For security, the system uses user authentication, encryption of sensitive data, secure storage and transmission of detection logs, and regular updates to avoid vulnerabilities.

The functional requirements of the system are: capturing road images and videos, preprocessing the input, detecting potholes with YOLOv8, displaying bounding boxes with confidence scores, generating alerts and reports, storing the results in a database, and integrating with road maintenance systems. The non-functional requirements are real-time performance with minimum latency, high accuracy, scalability to different regions and road conditions, a simple and usable interface, reliability under daylight, night, rain and shadows, and security of data and access.

From the feasibility point of view, the system is technically feasible because YOLO is a proven real-time detector, strong Python libraries and GPU enabled systems are available, cameras and annotated data can be obtained, and the architecture can easily be upgraded with newer models and larger datasets in future.

VI. RESULTS AND DISCUSSION

The trained YOLOv8 model was tested on road images and videos covering different road conditions. The model was able to locate potholes and to display them with bounding boxes and confidence scores in real time. The main results obtained are given in Table III.

TABLE III. PERFORMANCE OF THE DEVELOPED SYSTEM

Parameter	Result
Detection model	YOLOv8
Precision	Above 92 %
Detection speed	About 35 FPS (Frames Per Second)
Output	Bounding box with confidence score, alerts, reports

A precision above 92% shows that most of the regions marked as potholes by the model are real potholes, so the number of false alarms is low. The speed of about 35 FPS is higher than what is needed to process a live video stream from a moving vehicle, so the system is practical for real-time monitoring. The integration of preprocessing, detection and

reporting modules in one pipeline also means that the output can be passed directly to the maintenance staff without any manual step.

The system, however, has some limitations. The detection accuracy goes down in poor lighting and in rainy conditions, where shadows, puddles and wet surfaces look similar to potholes. The model needs a large annotated dataset for the best training, and training is computationally heavy and demands GPU resources. The integration with real-time municipal repair scheduling systems is also limited at present.

VII. CONCLUSION AND FUTURE SCOPE

In this paper we presented a pothole detection system based on deep learning and the YOLOv8 model. The system captures road images and videos, preprocesses them, detects potholes in real time with bounding boxes and confidence scores, and passes the information to alert, reporting and road maintenance modules. The developed model reached a precision above 92 percent at about 35 FPS, which shows that it is suitable for smart transportation systems. We believe that such automated detection can reduce accidents and vehicle damage and help authorities to plan the road maintenance in advance.

In the next phase we plan to use newer YOLO versions and optimized techniques to improve accuracy, to add GPS based tracking of pothole locations, to build a mobile application for real-time monitoring and reporting, and to improve the performance in low light, rain and shadow conditions. Lightweight models for mobile and embedded devices, map-based visualization, automatic alert generation and repair scheduling, a larger dataset from different road environments and weather, and adaptive learning that retrains the model with new data are also part of our future work.

ACKNOWLEDGMENT

The authors thank the Principal, HOD and Project coordinator Department of Computer Science and Engineering, SJM Institute of Technology, Chitradurga, for the support and facilities provided during this project work.

REFERENCES

- [1] M. Y. M, P. K. M. J, A. K, and S. S. V, "Pothole detection using deep learning methods," in Proc. 2025 IEEE Bangalore Humanitarian Technology Conf. (B-HTC), 2025, pp. 1–6, doi: 10.1109/B-HTC64616.2025.11116474.
- [2] M. Senbagavalli, S. Mahapatra, A. F. M. K, and R. L, "Revolutionizing road safety: An innovative approach to pothole detection and prevention using XAI and deep learning," in Proc. 4th Int. Conf. Sentiment Analysis and Deep Learning (ICSADL), 2025, pp. 878–884, doi: 10.1109/ICSADL65848.2025.10933246.
- [3] U. Khusni, R. Hidayat, H. Susanto, and A. Insani, "Automatic detection of potholes uses deep learning to reduce traffic accidents," in Proc. 2024 IEEE Int. Conf. Internet of Things and Intelligence Systems (IoTaIS), Nov. 2024, pp. 8–13, doi: 10.1109/IoTaIS64014.2024.10799472.
- [4] 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, pp. 779–788.
- [5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
- [6] W. Liu et al., "SSD: Single shot multibox detector," in Proc. European Conf. Computer Vision (ECCV), 2016, pp. 21–37.
- [7] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLOv8," 2023. [Online]. Available: <https://github.com/ultralytics/ultralytics>