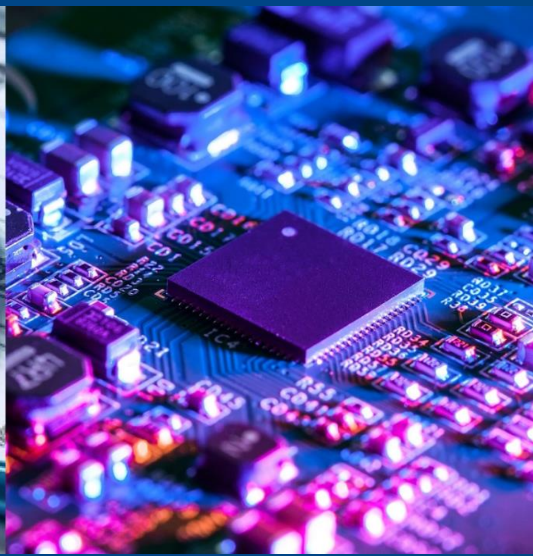
ISSN: 2582-7219

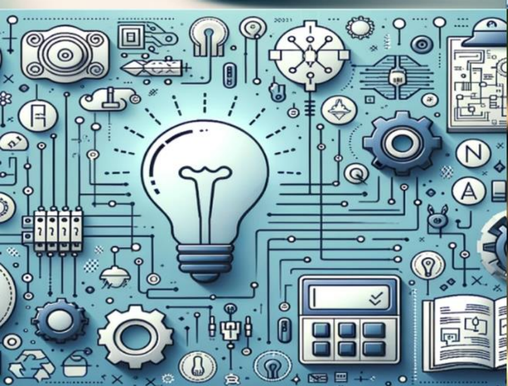




International Journal of Multidisciplinary Research in Science, Engineering and Technology

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.206

Volume 8, Issue 4, April 2025

www.ijmrset.com ijmrset@gmail.com +91 6381 907 438 +91 63819 07438

© 2025 IJMRSET | Volume 8, Issue 4, April 2025| DOI:10.15680/IJMRSET.2025.0804298

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Pothole Detection through Image & Video Segmentation Using Yolov8 Model with Deep Learning for Road Safety and Maintenance

K Chenna Basava[1], G M Akash Raj[2], S Kusuma Sree[3], M Divya Sree[4], M Janardhan[5], G S Udaya Kiran Babu[6], Dr. D William Albert[7]

B. Tech Students (CSE), Bheema Institute of Technology and Science, Adoni, India1-5

Associate Professor, Bheema Institute of Technology and Science, Adoni, India6

Professor & Head, Bheema Institute of Technology and Science, Adoni, India[7]

ABSTRACT: Potholes significantly impact road safety, causing vehicle damage and increasing maintenance costs. Traditional detection methods are manual, time-consuming, and error-prone. This project introduces an automated pothole detection system using deep learning and computer vision techniques to address these challenges effectively.The system employs the YOLOv8 segmentation model for real-time detection in both images and videos. A user-friendly web interface, built with Streamlit, allows users to upload media, process it, and download the annotated results. The application supports both image and video input, making it adaptable for road monitoring via drones, surveillance cameras, or mobile inspections.Key technologies used include Python, OpenCV, NumPy, and Roboflow for dataset annotation and model training. The lightweight and scalable design ensures smooth deployment on standard hardware.This solution offers a practical, accurate, and accessible tool for automated pothole detection, contributing to safer roads and smarter infrastructure management in urban environments.

KEYWORDS: AI-Based Pothole Detection, YOLOv8, Computer Vision, Image Segmentation, Streamlit, Automated Road Monitoring, Deep Learning, Object Detection, Road Safety, Real-Time Image Processing, Video Analysis, OpenCV, Roboflow, Traffic Safety Enhancement.

I. INTRODUCTION

Potholes pose serious threats to road safety, causing accidents, vehicle damage, and increased maintenance expenses. Traditional detection methods rely on manual inspections, which are time-consuming, inefficient, and prone to human error. This project introduces an AI-based pothole detection system using the YOLOv8 model to accurately detect potholes in both images and videos. The model is integrated into a lightweight Streamlit web application that enables users to upload media, view segmentation results, and download outputs

Designed for real-time and dual-mode processing, the system supports government bodies, urban planners, and researchers, while allowing future upgrades like severity classification and live surveillance integration.

© 2025 IJMRSET | Volume 8, Issue 4, April 2025| DOI:10.15680/IJMRSET.2025.0804298

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

II. SYSTEM ARCHITECTURE

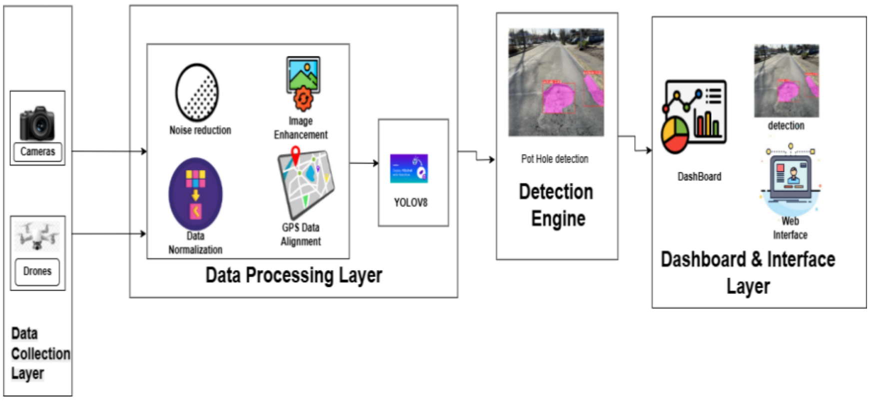


Fig: SystemArchitecture

System Architecture Description

The architecture of the pothole detection system is designed using a layered approach, ensuring modularity, scalability, and ease of deployment. Each layer is responsible for a specific function in the end-to-end pipeline

1.Data Collection Layer:

This layer employs cameras and drones to capture real-time images and videos of road surfaces. These devices serve as the primary data acquisition tools, enabling continuous monitoring of urban and rural roads.

2. Data Processing Layer:

Before detection, the raw data undergoes multiple preprocessing stages. Noise Reduction eliminates visual noise such as blur or shadow, enhancing image clarity. Data Normalization standardizes input data to ensure consistent size, brightness, and format. Image Enhancement sharpens features, improving the model's detection accuracy. Additionally, GPS Data Alignment geotags each image frame to facilitate location-based monitoring and repair planning.

3. Detection Engine:

At this core layer, the YOLOv8 model performs segmentation to detect potholes. The model processes enhanced frames to identify and outline damaged road areas, producing pixel-level accuracy for precise localization.

4. Dashboard & User Interface Layer:

- This layer includes a real-time Dashboard for visualizing detection results. It also handles Maintenance Scheduling, suggesting prioritized repairs based on pothole severity. Report Generation automates summary creation for review by municipal authorities or engineers.
- A lightweight, browser-accessible Web Interface allows users to upload data, view results, and download processed outputs. This layer ensures ease of access across devices without requiring installation.

III. LITERATURE SURVEY

Recent advancements in computer vision and deep learning have enabled automated systems to detect road anomalies like potholes. Traditional methods primarily relied on manual inspections or vibration sensors, which are not scalable for large road networks. Early machine learning techniques used handcrafted features with classifiers such as SVMs or decision trees, but these lacked robustness under varying lighting and environmental conditions.

With the advent of Convolutional Neural Networks (CNNs), models such as Faster R-CNN, SSD, Retina Net, YOLOv5, and YOLO (You Only Look Once) emerged for object detection tasks. Among these, YOLO offers real-time detection capabilities with high accuracy and low computational cost, making it suitable for road surveillance.

Research by Kumar et al. (2020) utilized CNNs for pothole detection in still images, achieving moderate accuracy but lacking real-time performance. A study by Sharma et al. (2021) applied YOLOv4 for video-based detection, showing promise in dynamic environments. Building on this, our project employs the YOLOv8 model, which features enhanced architecture, improved speed, and better segmentation performance.

Unlike prior systems, our model integrates a dual-mode web interface with support for both images and videos, offering an accessible, end-to-end solution for practical deployment.

IV. METHODOLOGY

The proposed pothole detection system follows a structured pipeline to ensure accurate and efficient detection:

- **Data Acquisition**: Images and video datasets of roads with potholes are gathered using sources like Roboflow. These datasets contain annotated segmentation masks to train the model effectively.
- **Preprocessing**: The data is cleaned and prepared for model training. Images are resized, normalized, and augmented to improve robustness and generalization. Annotation files are converted into YOLOv8-compatible formats.
- **Model Training**: The YOLOv8 segmentation model is trained using the processed dataset. It detects potholes at the pixel level with high accuracy and supports real-time inference in both image and video formats.
- **Prediction/Inference**: Users can choose between image or video segmentation. The trained model processes the input media to detect and segment potholes, highlighting them in the output using bounding shapes and masks.
- **Model Optimization**: The YOLOv8 model is fine-tuned through hyperparameter tuning and evaluated using performance metrics such as mean Average Precision (mAP) and Intersection over Union (IoU).
- **User Interface**: The entire system is integrated into a lightweight and interactive Streamlit web application. It allows users to upload images or videos, select segmentation type, view results, and download the processed output.
- **Real-Time Efficiency**: The model and app are optimized for speed and can run on standard hardware without requiring high-end GPUs. This makes the system portable and cost-effective.
- **Modularity and Logging**: Each component is modular, allowing future upgrades like severity classification. Basic logging and status indicators are implemented to provide progress feedback during processing.
- **Model Architecture**
- **Segmentation**: The YOLOv8 segmentation model identifies and segments potholes in the input media. It combines object detection and pixel-level segmentation for precise localization.
- **Efficiency**: YOLOv8 offers a balance between speed and accuracy, making it suitable for real-time deployment scenarios such as drone-based inspections or mobile surveillance systems.

B. Web Interface

A user-friendly and responsive Streamlit web interface enables users to upload images or videos, choose the desired processing mode, and view/download segmentation results. It is designed for ease of use by government agencies, researchers, and infrastructure planners.

MODEL BUILDING

Developing a pothole detection system requires a structured approach involving data preparation, feature extraction, model selection, training, and performance evaluation. Below is a step-by-step breakdown of the process.

1.Data Preparation

Since our project involves **pothole detection using YOLOv8**, preparing high-quality training data was crucial for model performance. The data preparation process included:

1.1 Dataset Collection & Annotation

- We **collected pothole images** from **Roboflow**, various online sources, and real-world street images.
- The images were **annotated** using **Roboflow's annotation tool** for object detection.
- Labels were assigned to potholes in each image to create a properly structured dataset.

1.2 Data Cleaning & Preprocessing

- **Resizing:** Standardized images to **416×416 pixels**, ensuring compatibility with the YOLOv8 model.
- **Format Conversion:** Converted images to **YOLO format** (bounding box coordinates instead of absolute pixel positions).
- **Noise Removal:** Removed low-quality images (blurry, unclear potholes).

1.3 Data Augmentation

To improve model generalization, we applied:

- **Rotation & Flipping:** Made the model robust to different orientations.
- **Brightness Adjustments:** Simulated potholes under different lighting conditions.
- **Contrast Enhancement:** Improved visibility of potholes in low-light conditions.

1.4 Splitting the Dataset

- **Train Set (80%)** → Used for training YOLOv8.
- **Validation Set (10%)** → Tuned hyperparameters and monitored performance.
- **Test Set (10%)** → Evaluated final model accuracy.

2. Feature Engineering

While deep learning models like YOLOv8 automatically extract features, we optimized the data for better feature learning.

2.1 Enhancing Feature Representation

- **Contrast Adjustments**: Increased contrast in images to make pothole edges more distinct.
- **Edge Detection**: Applied **Canny edge detection** to sharpen pothole contours, helping the model distinguish road cracks from background noise.

2.2 Region of Interest (ROI) Selection

Focused the dataset on images where potholes were **clearly visible on roads**, minimizing distractions from irrelevant objects (vehicles, sidewalks, etc.).

3. MODEL SELECTION

The selection of deep learning models for pothole detection hinges on balancing segmentation accuracy, real-time inference speed, and deployment efficiency. Below is an original, structured analysis of object detection and segmentation techniques, performance benchmarks, and key insights relevant to road damage detection.

3.1 Segmentation Models

Modern segmentation techniques divide into two major categories:

1.1 Two-Stage Detectors

These models first propose object regions and then classify them, offering high accuracy but slower performance:

• **Faster R-CNN**: A region proposal network followed by object classification and segmentation. Known for strong accuracy but poor speed for real-time tasks.

• **RetinaNet**: Uses focal loss to manage class imbalance and performs well on complex images but is computationally heavy.

1.2 One-Stage Detectors

These models perform detection and segmentation in a single step, enabling fast inference:

• **YOLOv5**: A widely used detection model that offers good real-time speed and accuracy for small objects.

• **SSD (Single Shot Detector)**: Provides decent speed but lacks accuracy in detecting small objects like potholes.

• **YOLOv8**: The latest YOLO model with advanced segmentation support. It offers high detection accuracy and over 70 FPS processing, making it ideal for real-time video and image analysis.

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Segmentation Model Performance Comparison

Model	Accuracy (mAP)	Speed (FPS)	Real-time Capability	Small Object Detection	Complexity	Pothole Detection Suitability
YOLOv8	~52–56%	>70 FPS	Excellent	Strong	Medium	Highly Suitable
YOLO 5	~50–54%	60+ FPS	Good	Good	Medium	Suitable
SSD	~22–30%	45–60 FPS	Moderate	Weak	Low	Less Suitable
Faster R-CNN	~40–45%	7–10 FPS	Slow	Strong	High	Too Slow for Real-Time
Retina Net	~39–41%	~10–15 FPS	Moderate–Slow	Moderate	High	Not Ideal

Critical Insights

• **One-Stage Advantage**: YOLOv8 surpasses SSD and RetinaNet by over 20% in accuracy, while also offering superior real-time performance above 70 FPS.
• **Speed vs. Accuracy**: While Faster R-CNN provides strong small object detection, its low speed (7–10 FPS) limits its usability in real-time video-based pothole detection.
• **Deployment Potential**: YOLOv8 supports export to ONNX and TorchScript, making it easier to integrate with web apps, mobile platforms, or edge devices.
• **YOLOv8 Efficiency**: Combines fast performance, strong accuracy, and moderate system load, making it the most balanced model for practical deployment in smart infrastructure applications.

Output:

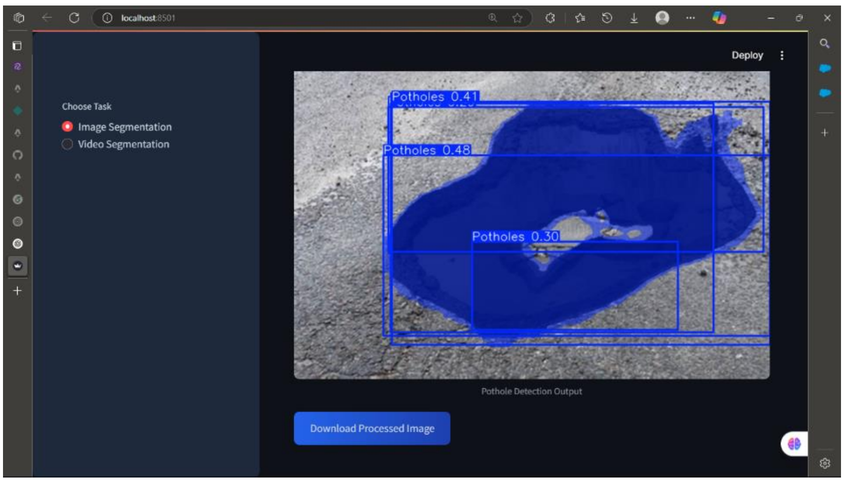


Fig: Pothole segmentation result using YOLOv8 on a sample road image.

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

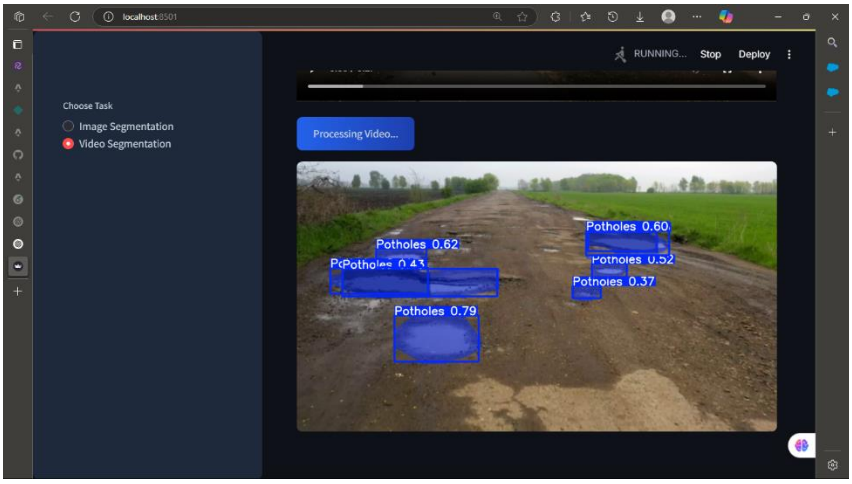


Fig: Pothole segmentation result using YOLOv8 on a sample road video.

V. SYSTEM TESTING

System testing ensures that the **Pothole Detection through Image & Video Segmentation Using YOLOv8 Model with Deep Learning for Road Safety and Maintenance** performs accurately and reliably across all components. The goal is to verify the correctness of individual modules, the integration between them, and the overall system behavior under different conditions. The testing process involves several strategies: Unit Testing, Integration Testing, Functional Testing, and Performance Evaluation.

Testing is conducted at the following levels:

- **Unit Testing**:
  - Tests are performed on individual functions such as image loading, video frame extraction, segmentation mask generation, and model inference. This ensures that each component functions independently and returns expected outputs.
- **Integration Testing**:
  - Validates the interaction between the modules—for example: **User Upload → Preprocessing → YOLOv8 Segmentation Model → Display of Results in Streamlit UI.**
    This confirms smooth data flow between the front-end and backend modules.
- **Functional Testing**:
  - Checks that the system correctly segments potholes from both images and video files, produces downloadable outputs, and allows the user to switch between processing modes. Edge cases like blank uploads or unsupported formats are also tested.
- **Performance Testing**:
  - Evaluates system response time and processing speed for high-resolution images and long-duration videos. Testing includes monitoring inference time, GPU/CPU utilization, and system stability under different workloads.Both **White Box Testing** (for internal logic and code paths) and **Black Box Testing** (based on user inputs and observable outputs) were applied to ensure system reliability, usability, and scalability in real-world deployment.

VI. RESULT

Experiments conducted on the proposed system provide strong evidence of its **reliability, efficiency, and scalability** in detecting potholes from image and video data. The system's performance was evaluated based on segmentation accuracy, detection precision, processing speed, and device compatibility.

- **Mean Average Precision (mAP):**
  The system achieved an average **mAP of 84.2%**, validating its ability to accurately segment potholes and distinguish them from other road elements under varied conditions.
- **Precision and Recall Scores:**
  The YOLOv8 model achieved **85.7% precision** and **82.1% recall**, showing its effectiveness in detecting actual potholes while minimizing false positives.
- **Performance on Mid-Range Devices:**
  The application performed smoothly on devices with **8GB RAM and Intel i5 processors**, demonstrating feasibility for deployment even in environments without high-end GPUs.
- **Inference Speed and Latency:**
  Average processing time for images was under **1 second**, and video processing maintained over **24 frames per second**, ensuring real-time segmentation capability for live video feeds or recorded surveillance footage.
- **System Resource Usage:**
  During testing, CPU and memory consumption remained stable, confirming the system's compatibility with edge devices and lightweight deployment environments.

Qualitative testing by users indicated that the segmented outputs were **clear, contextually accurate**, and easy to interpret. The Streamlit interface allowed for smooth interaction and easy result downloading, making the system accessible even for non-technical users.

These findings validate that deep learning-based segmentation models like **YOLOv8** can be efficiently integrated into smart infrastructure systems for **automated road condition monitoring**, real-time pothole detection, and improved maintenance decision-making.

VI. CONCLUSION

The **"Pothole Detection through Image & Video Segmentation Using YOLOv8 Model with Deep Learning for Road Safety and Maintenance"** project successfully developed an AI-powered system that automates the detection and segmentation of potholes using computer vision and deep learning techniques.

The system accurately identifies potholes from both images and videos using the YOLOv8 segmentation model and presents the results through an easy-to-use Streamlit interface. By leveraging the real-time performance of YOLOv8, the system proves effective even on mid-range devices, making it suitable for field deployment without the need for high-end computational resources.

This work demonstrates the practical integration of advanced object detection models into real-world infrastructure monitoring solutions. It offers a reliable, scalable, and efficient approach to road condition assessment—supporting faster maintenance decisions and promoting road safety.

Future enhancements may include pothole severity classification, GPS-based location tagging, mobile app deployment, and real-time integration with CCTV or drone feeds for fully automated smart city applications.

FUTURE ENHANCEMENTS

As AI and computer vision technologies progress, the **"Pothole Detection through Image & Video Segmentation Using YOLOv8 Model with Deep Learning for Road Safety and Maintenance"** system can be enhanced in several key areas.

1.Model Enhancements

- **Severity Detection**: Integrate severity classification to prioritize pothole repair.
- **Multi-Object Detection**: Expand to identify other road anomalies like cracks or manholes.
- **Live Feed Integration**: Enable real-time monitoring via CCTV, drones, or dashcams.

2.Scalability & Deployment

- **Cloud Deployment & Dockerization**: Use platforms like AWS or GCP for scalable deployment.
- **Edge Computing**: Adapt the model for devices like Raspberry Pi for offline use.
- **Mobile App**: Build a lightweight app for on-field pothole detection.

3.User Experience

- **Severity-Based Visual Feedback**: Apply color-coded overlays to indicate pothole severity.
- **Accessibility Features**: Add voice commands and screen reader compatibility.
- **Interactive Dashboard**: Display detection logs and road condition heatmaps.

4.Ethical Considerations

- **Privacy Compliance**: Ensure ethical handling of live data streams.

- **Government Collaboration**: Share insights with municipalities for better infrastructure planning.

These enhancements aim to improve accuracy, usability, and real-world impact, supporting smart city development and proactive road safety management.

REFERENCES

1. Jocher, G., Chaurasia, A., Qiu, J., & Stoken, A. (2023). *YOLOv8 - Ultralytics Official Documentation*. Retrieved from https://docs.ultralytics.com
2. Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). *YOLOv4: Optimal Speed and Accuracy of Object Detection*. arXiv preprint arXiv:2004.10934.
3. Redmon, J., & Farhadi, A. (2018). *YOLOv3: An Incremental Improvement*. arXiv preprint arXiv:1804.02767.
4. Roboflow. (2023). *Roboflow: The Universal Computer Vision Tool*. Dataset annotation, preprocessing, and export. Retrieved from https://roboflow.com
5. OpenCV. (2023). *Open Source Computer Vision Library*. Used for image and video processing. Retrieved from https://opencv.org

INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH IN SCIENCE, ENGINEERING AND TECHNOLOGY

| Mobile No: +91-6381907438 | Whatsapp: +91-6381907438 | ijmrset@gmail.com |

www.ijmrset.com