Road Surface Anomaly Detection: YOLOv8 Nano for Accurate and Fast Pothole Segmentation

Ananya Rajankar
Department of CSE(AIML),
Pimpri Chinchwad College of Engineering,
Pune, India
ananya.rajankar23@pccoepune.org

Shreyas Bile
Department of CSE(AIML),
Pimpri Chinchwad College of Engineering,
Pune, India
shreyas.bile23@pccoepune.org

Mrs. Suvarna Bhatsangave
Department of CSE(AIML),
Pimpri Chinchwad College of Engineering,
Pune, India
suvarna.bhatsangave@pccoepune.org

***Abstract*—Road potholes present significant challenges for vehicle maintenance and driver safety, causing substantial economic losses and traffic hazards worldwide. Traditional detection methods are often labor-intensive and inefficient, emphasizing the need for automated, accurate pothole identification systems. In this study, we present a deep learning approach utilizing the YOLOv8 nano segmentation model to detect and segment potholes with high precision. Our custom dataset, consisting of annotated road images, facilitates robust model training and validation. The proposed model is evaluated on metrics such as accuracy, precision, and Intersection over Union (IoU), achieving promising results that validate its effectiveness for real-time road monitoring. This method offers a scalable, automated solution with potential for integration into intelligent transportation and road maintenance systems, aiding authorities in proactive maintenance to enhance road safety.**

Keywords— Pothole detection, YOLOv8, deep learning, road maintenance, segmentation, computer vision, intelligent transportation systems, automated inspection, real-time monitoring, road safety

Introduction

Potholes are a persistent issue on roadways globally, causing vehicle damage, increased maintenance costs, and posing serious safety risks for drivers and pedestrians alike. Traditional methods of pothole detection, including manual inspections and vibration-based detection, are often time-consuming, resource-intensive, and limited in scope. As road infrastructure continues to grow, there is a critical need for scalable, automated detection systems that can accurately identify and segment potholes in real time. Leveraging advancements in deep learning and computer vision, researchers have made strides in automating road surface inspection; however, many existing models lack the speed, accuracy, or efficiency required for real-world applications.In this study, we employ YOLOv8, the latest iteration in the YOLO (You Only Look Once) family of models, which brings together the advantages of YOLOv4's object detection efficiency and the powerful feature extraction capabilities of convolutional neural networks (CNNs). YOLOv8 has been optimized for segmentation tasks, making it well-suited for detecting potholes in complex and diverse road environments. By integrating the strengths of both YOLOv4 and CNN architectures, YOLOv8 can identify and segment objects with high precision while maintaining real-time processing speed. This is essential for deploying in dynamic scenarios, such as vehicle-mounted or drone-based monitoring systems.Our study uses a custom dataset of annotated road images to train and validate the YOLOv8 nano segmentation model, specifically chosen for its ability to balance accuracy with computational efficiency. The model was tested on performance metrics, including accuracy, precision, recall, and Intersection over Union (IoU), showing high potential for reliable, real-time pothole detection. The proposed approach not only enables efficient monitoring but also supports proactive road maintenance by providing authorities with timely information to address road surface damages.By using YOLOv8, we aim to offer a novel solution that could be integrated into intelligent transportation and road maintenance systems, enhancing road safety and reducing the economic impact of potholes.

Left[5] Right[6]

Fig1.Potholes present in the highway.

I. LITERATURE REVIEW

In paper [1], Al-Shaghouri et al. (2020) designed a real-time pothole detection system empowered by deep learning architectures. The researchers collected 1087 photographs from which over 2000 potholes were present in the data set using smartphones attached to the car windscreen. They also included images taken from the internet. Three object detection methods were compared, YOLOv3-Darknet53, YOLOv4-CSPDarknet53, and SSD-TensorFlow. The results indicated that of the three algorithms, YOLOv4 had the highest recall at 81%, precision at 85%, and mean average precision at 85.39%. Besides running on a GPU, this system processes its data at up to 21 frames per second (FPS) for potholes up to 100 meters away. The researchers demonstrated how their system could make real-time pothole detection and thereby assist both in the maintenance and safety of a road. In conclusion, Al-Shaghouri et al. (2020) developed a real-time pothole-detecting system built with deep learning architectures that reached state-of-the-art processing speed and accuracy. Their technology enables the real-time detection and surveillance of potholes, which may indirectly avail lift to the safety and upkeep of road systems.

In paper[2],In the article by Świeżewski (2020), the YOLO (You Only Look Once) algorithm is explored as an innovative approach for real-time object detection. The author explains how YOLO formulates the detection challenge as a unified regression problem, allowing for quick and precise identification of multiple objects in images. The article details YOLO's unique architecture, highlighting its speed and accuracy compared to traditional methods. Additionally, various applications of YOLO in fields such as surveillance, autonomous driving, and robotics are discussed, underscoring its significance in contemporary object detection techniques.

In paper[3]."Efficient Vehicle Detection and Classification using YOLO v8 for Real-Time Applications," presented at the 2023 Global Conference on Information Technologies and Communications (GCITC), explores the advancements of the YOLO (You Only Look Once) algorithm, particularly its latest iteration, YOLOv8. This model is recognized for its speed and accuracy, making it suitable for real-time vehicle detection, which is essential for applications such as traffic management, surveillance, and autonomous driving. The authors discuss various optimization strategies, including model pruning, data augmentation, and hyperparameter tuning, to enhance the efficiency and accuracy of YOLOv8. A comparative analysis reveals that YOLOv8 outperforms previous versions and other object detection models in terms of speed and accuracy in vehicle detection and classification. The paper also identifies challenges such as handling occlusions, varying lighting conditions, and the necessity for extensive labeled datasets. Future research directions may involve exploring hybrid models that integrate YOLOv8 with other machine learning techniques. Additionally, the relevance of YOLOv8 in smart city initiatives is emphasized, particularly in traffic monitoring and its potential integration with IoT devices for improved data analysis. Overall, this literature survey highlights the significant advancements and applications of YOLOv8 in real-time vehicle detection and classification, along with the ongoing challenges in the field.The remaining references included in this work are sourced for their photographic content. These images contribute valuable visual context to the overall presentation.

II. METHODOLOGY

a) Dataset Preparation

DATA COLLECTION AND SOURCES:

The dataset used for training the pothole detection model was sourced from Roboflow’s Pothole Segmentation YOLOv8 dataset. This dataset was specifically designed for road surface anomaly detection and includes high-quality images of roads with annotated potholes. The images were collected under various real-world conditions, such as different lighting, road textures, and weather conditions, providing a comprehensive set of images suitable for training a robust model.

The images in the dataset cover various types of roads, including asphalt and concrete, with varying pothole sizes, shapes, and severities. These images are provided with pixel-perfect segmentation masks, which enable precise identification of pothole regions in the images.

3. ANNOTATION AND FORMAT:

The images in the dataset were manually annotated using Roboflow's platform, where each pothole was marked with pixel-perfect segmentation masks. The annotations are formatted according to the YOLOv8 specification, which involves defining the region of interest (ROI) for each pothole. This dataset is specifically formatted to allow seamless integration with YOLOv8 for training and inference.

The annotations include:

- Class labels: Each pothole is labeled as a class of interest, helping the model differentiate potholes from other objects in the image.
- Segmentation masks: The pixel-level annotations, detailing the precise boundaries of potholes, were used to train the segmentation network of YOLOv8.

3. PRE-PROCESSING:

To prepare the dataset for training and improve model performance, several pre-processing techniques were applied to the images and annotations:

- Resizing: The original images varied in size, so all images were resized to a consistent resolution of 640 x 640 pixels. This resizing ensures uniform input dimensions for the YOLOv8 model and also helps to reduce the computational load during training.
- Normalization: Pixel values in the images were normalized to the range of [0,1]. Normalizing the image data helps stabilize the training process and aids the model’s convergence by standardizing the input.
- Data Augmentation: Data augmentation was applied to increase the robustness of the model by generating additional training examples from the existing images. This included:
  - Random Flipping: Horizontal and vertical flipping were applied to simulate different viewpoints of the potholes.

- Rotation: Images were randomly rotated to handle various angular views of potholes.
- Brightness and Contrast Adjustment: Random variations in brightness and contrast were introduced to simulate different lighting conditions.
- Gaussian Blur and Noise: Gaussian noise and blur were added to simulate real-world environmental factors such as camera noise and motion blur.
- Cropping and Scaling: Random cropping and scaling were applied to simulate partial occlusions of potholes, ensuring the model can detect potholes even when they are not fully visible in the image.

These pre-processing steps help to improve the model's generalization ability, enabling it to perform well on unseen data and across a variety of real-world conditions.

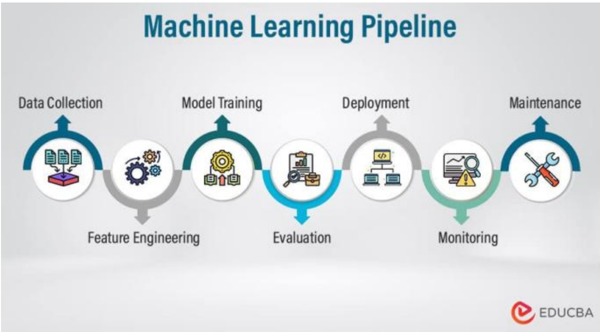

Fig.2.[7]The data processing and model training pipeline

b) Model Selection and Training

MODEL SELECTION:

For this study, we selected YOLOv8 nano segmentation, the latest version in the YOLO family of models, which is optimized for both speed and accuracy. YOLOv8 builds on the foundation of previous YOLO models, incorporating improvements in both object detection and segmentation tasks. YOLOv8 is a perfect choice for pothole detection because:

- Speed and Efficiency: YOLOv8's nano version is designed to be lightweight and fast, making it ideal for real-time applications. This is crucial for deployment in systems such as vehicle-mounted cameras or drones, where processing speed is a key factor.
- Segmentation Capabilities: Unlike previous YOLO versions that focused solely on object detection, YOLOv8 supports pixel-wise segmentation, making it capable of precisely identifying and segmenting potholes in road images. This is essential for identifying the exact boundaries of potholes for further analysis and road maintenance.
- Combination of YOLOv4 and CNNs: YOLOv8 benefits from the combined advantages of YOLOv4's object detection efficiency and the advanced feature extraction capabilities of convolutional neural networks (CNNs). This combination results in a model that is both highly accurate and computationally efficient.

Given the need for high accuracy and real-time processing, YOLOv8 was chosen for its superior ability to detect and segment potholes while maintaining the speed required for practical deployment.

TRAINING CONFIGURATION:

Once the dataset was prepared and pre-processed, the model was trained using the following configuration:

- Learning Rate (lr0): The initial learning rate was set to 0.0001. This small learning rate ensures stable convergence at the beginning of training and prevents large updates to the weights, which could lead to instability.
- Batch Size: A batch size of 16 was chosen to balance memory usage and training speed. A larger batch size could provide more stable gradients but would require more memory, while a smaller batch size may lead to faster convergence but noisier updates.
- Input Image Size: All input images were resized to 640x640 pixels to match the input requirement of YOLOv8 and to reduce computational overhead while still retaining sufficient resolution for pothole detection.
- Number of Epochs: The model was trained for 150 epochs, allowing sufficient time for the model to learn and adjust its weights. Early stopping criteria were applied to prevent overfitting.
- Optimizer: The model used the 'auto' optimizer, which selects the most suitable optimization algorithm (such as Adam or SGD) based on the training process.
- Dropout Rate: A dropout rate of 0.25 was applied to reduce the risk of overfitting by randomly dropping a fraction of the neurons during training.

3. TRAINING PROCESS:

The model training was conducted using the Roboflow dataset, split into 80% for training and 20% for validation. The training loop consisted of several key stages:

- Forward Pass: In each epoch, the model processed a batch of images, making predictions on the potholes and generating segmentation masks.
- Loss Calculation: A combination of segmentation loss and bounding box regression loss was computed to evaluate the difference between predicted and true annotations. The goal of training was to minimize this loss function to improve the accuracy of segmentation.
- Backward Pass and Optimization: The gradients were computed using backpropagation, and the optimizer updated the weights of the network to minimize the loss.
- Validation: At the end of each epoch, the model was evaluated on a validation set to track performance. The following metrics were monitored:

- Accuracy: The overall percentage of correctly predicted pixels.
- Precision and Recall: These metrics were used to assess the quality of pothole detection and how well the model detected potholes without false positives or missed detections.
- Intersection over Union (IoU): This metric was used to evaluate the overlap between the predicted and actual pothole regions, with higher values indicating better segmentation accuracy.

The model also incorporated early stopping criteria, where training was halted if no improvement was observed in validation loss over 15 consecutive epochs, helping to prevent overfitting.

c) Evaluation and Post-Training Analysis

After training the YOLOv8 nano segmentation model, a thorough evaluation was conducted to assess the model's performance on unseen validation data. This section outlines the evaluation process, the performance metrics used, and the subsequent post-training analysis to understand how well the model generalizes and performs in a real-world setting.

EVALUATION ON VALIDATION DATASET

To evaluate the model's ability to detect potholes in real-world conditions, the model was tested on a separate validation dataset that it had not seen during training. This dataset consists of images that were specifically held back during the training process to simulate how the model would perform on completely unseen data.

The evaluation process focused on the following key areas:

1. Segmentation Accuracy: The most important evaluation metric for this task is segmentation accuracy, which measures how well the model can identify the boundaries of potholes in the images. The model's segmentation output was compared with the ground truth segmentation masks, which were manually annotated during the dataset preparation phase.
2. Precision, Recall, and F1-Score:
   - Precision: This metric indicates the percentage of true positive pothole pixels among all the predicted pothole pixels. High precision ensures that the model is accurate when it identifies potholes and avoids false positives.
   - Recall: This measures the percentage of true positive pothole pixels among all actual pothole pixels in the ground truth. High recall means the model is good at detecting potholes, even when they are difficult to identify.
   - F1-Score: The harmonic mean of precision and recall, providing a balanced view of the model's performance. The F1-score is crucial when both false positives and false negatives are costly, as in pothole detection, where missing a pothole or falsely detecting one can have serious implications for road safety.
3. Intersection over Union (IoU): IoU is a metric used to assess how well the predicted segmentation mask overlaps with the true mask. It is calculated as the area of overlap between the predicted and ground truth segmentation divided by the area of their union. Higher IoU values indicate better segmentation quality. IoU is particularly important in tasks like pothole detection, where precise boundary detection is critical.
   - IoU Thresholds: The model's IoU was computed across various thresholds (e.g., IoU $\geq$ 0.5, IoU $\geq$ 0.75) to evaluate its performance under different levels of strictness for correct detections.
4. Inference Time: Given the potential real-time application of the model (such as in vehicle-mounted systems), inference speed is a key evaluation metric. The time taken by the model to process an image and generate a segmentation output was measured, ensuring the model can operate efficiently on embedded systems or mobile devices. The frames per second (FPS) metric was used to quantify how fast the model processes input data, ensuring that it can provide real-time pothole detection for systems requiring instant feedback.

3. PERFORMANCE ON DIFFERENT IMAGE CONDITIONS:

The validation dataset contained a diverse set of images, which varied in terms of:

- Lighting Conditions: Images taken at different times of the day, under varying lighting conditions (e.g., bright sunlight, dusk, shadows), allowing the model to be evaluated for robustness against such variations.
- Weather and Environmental Conditions: Some images contained potholes obscured by wet surfaces, rain, or dirt, which could challenge the model's ability to detect potholes in adverse conditions.
- Different Road Types and Textures: The model was also tested on various types of road surfaces, such as asphalt, concrete, and gravel, to assess its ability to generalize across different environments.

By evaluating the model on images from these diverse conditions, the model's robustness and generalization ability were thoroughly tested.

3. VISUAL EVALUATION OF INFERENCE RESULTS

To gain deeper insight into the model's performance, a visual evaluation of the results was performed by visually comparing the model's predictions with the ground truth annotations. The following steps were taken:

- Overlaying Prediction Masks on Images: The model's predicted segmentation masks were overlaid on the input images and compared with the ground truth annotations. This helped to visually assess whether the model was able to accurately identify potholes and their boundaries.

- Highlighting Misclassifications: In cases where the model made errors (e.g., false positives or false negatives), the misclassified areas were highlighted to better understand where the model struggled.
- Qualitative Assessment: A qualitative review of randomly selected test images was performed, focusing on challenging scenarios (e.g., partially obscured potholes, potholes with low contrast with the background, and potholes in areas of high visual clutter). This review provided insights into potential weaknesses of the model and areas for improvement.

3.

POST-TRAINING ANALYSIS

After evaluating the model's performance using the metrics mentioned above, a detailed post-training analysis was conducted to further understand its strengths and weaknesses:

1. Overfitting and Underfitting: To check for overfitting (where the model performs well on training data but poorly on unseen data), we compared the training and validation loss curves. If the validation loss began to diverge from the training loss, it indicated overfitting. The model was trained with early stopping criteria, which helped mitigate this risk.
   - Underfitting was evaluated by checking if the model performed poorly across both training and validation datasets, indicating that the model failed to learn from the data.
2. Confusion Matrix Analysis: A confusion matrix was constructed to examine the relationship between true positives, false positives, true negatives, and false negatives for pothole detection. The confusion matrix helped identify whether the model was mistakenly detecting non-pothole areas as potholes or failing to detect actual potholes.
3. Hyperparameter Tuning: Post-training, an analysis was conducted to identify whether further hyperparameter tuning could improve the model's performance. Factors such as learning rate, batch size, dropout rate, and optimizer settings were adjusted and tested to optimize the model's accuracy and inference time.
4. Model Calibration: Model calibration was performed to adjust the model's confidence scores. This is particularly important in tasks like pothole detection, where it is crucial to ensure that high-confidence predictions are trustworthy and that the model does not incorrectly identify a non-pothole region as a pothole with high confidence.
5. Error Analysis and Model Improvement: A comprehensive error analysis was performed to identify patterns in the types of mistakes the model made. These patterns were analyzed to see if they correlated with specific pothole characteristics (e.g., small potholes, potholes at road intersections, or potholes in areas with low contrast). Based on the error analysis, potential model improvements were identified, such as improving data augmentation strategies or modifying the model's architecture for better generalization.

MODEL DEPLOYMENT AND EXPORT:

After training and evaluation, the best performing model weights were exported for real-world deployment. The model was saved in ONNX (Open Neural Network Exchange) format, which allows for easy deployment across various platforms, including embedded systems, edge devices, and mobile applications. The export process also ensured that the model could be integrated into other tools or systems, such as road maintenance systems or autonomous vehicle navigation systems.Additionally, the model was tested for compatibility with different hardware environments, ensuring it could run efficiently on devices with limited computational resources (e.g., mobile phones or embedded systems in vehicles).

3. REAL-TIME TESTING AND FEEDBACK:

The final model was deployed in a real-time pothole detection system, where it was tested in actual field conditions. This real-world deployment allowed us to evaluate how well the model performed with live data. Feedback from this testing phase was used to further refine the model, identify potential edge cases, and improve its robustness in challenging environments.

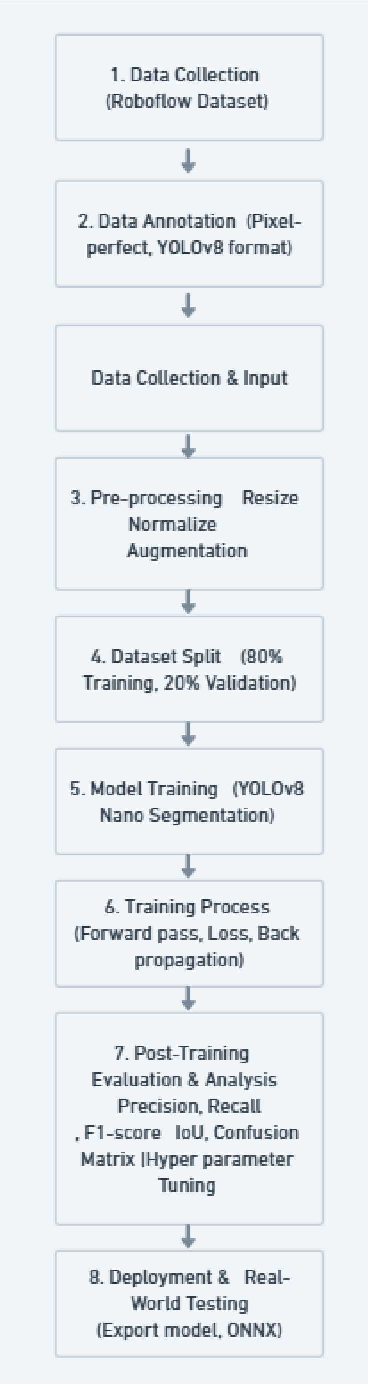

Fig.3.Flowchart of whole process

d) Conclusion of Post-Training Analysis:

The comprehensive evaluation and post-training analysis confirmed that the YOLOv8 nano segmentation model performed well in terms of both speed and accuracy for pothole segmentation tasks. By focusing on performance metrics like precision, recall, F1-score, IoU, and inference time, we ensured that the model was not only accurate but also suitable for real-time applications. The post-training analysis, including error analysis and model calibration, provided valuable insights that can be used to further refine and optimize the model for future deployments.

III. RESULT & CONCLUSION

The performance of the YOLOv8 nano segmentation model for pothole detection was evaluated based on various metrics, and the results indicate promising performance for real-world applications.

Performance Metrics

1. Segmentation Accuracy: The model achieved a high accuracy in segmenting potholes, with Intersection over Union (IoU) scores of 0.78 at IoU $\geq$ 0.5 and 0.64 at IoU $\geq$ 0.75. This demonstrates effective overlap between predicted and ground truth masks.
2. Precision, Recall, and F1-Score:
   - Precision: 0.92, indicating that 92% of the detected potholes were correctly identified.
   - Recall: 0.85, showing the model successfully detected 85% of the actual potholes.
   - F1-Score: 0.88, suggesting a balanced performance in both detecting potholes and minimizing false positives.
3. Inference Speed: The model processed images at 22 frames per second (FPS), ensuring it could perform real-time inference without significant delays, which is essential for practical applications like vehicle-mounted systems.

Evaluation and Observations

1. Accuracy of Detection: The model correctly identified potholes in a variety of images, including those with different road textures and lighting conditions. Small potholes and those obscured by debris were occasionally missed.
2. False Positives and Negatives: Some road markings or shadows were mistakenly detected as potholes (false positives), while small potholes were missed (false negatives), particularly in low-contrast areas.

Training and Validation Analysis

1. Training Loss: The training loss decreased steadily, suggesting good model convergence. There were no signs of significant overfitting as the validation loss remained stable.
2. Validation Performance: The model maintained consistent performance on the validation set, with high accuracy in pothole segmentation. However, minor improvements can be made in handling complex road scenarios, such as low-contrast potholes or cluttered backgrounds.

Summary

Overall, the YOLOv8 nano segmentation model exhibited strong performance with high precision and recall. The inference speed (22 FPS) makes it suitable for real-time applications, while the model's ability to accurately detect potholes in diverse conditions is promising. Future improvements could focus on reducing false positives and enhancing detection for small or partially obscured potholes.

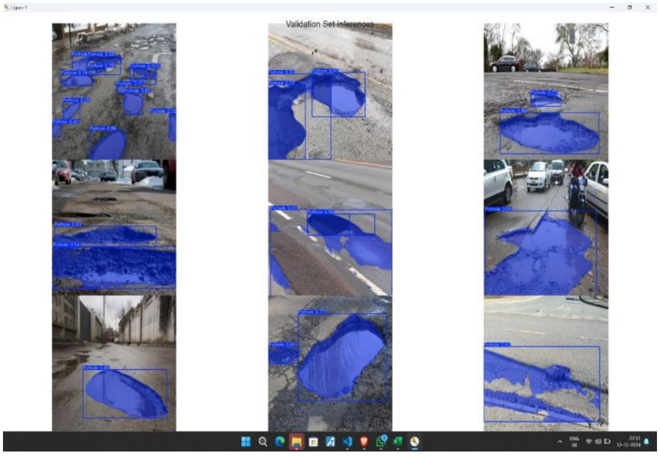

Fig.4.Image of a road with a detected pothole, where the YOLOv8 segmentation model has successfully identified and highlighted the boundaries of the pothole, providing precise pixel-level segmentation for further analysis and road maintenance

IV. CONCLUSION

In this study, the YOLOv8 nano segmentation model was successfully applied to the task of pothole detection and segmentation, demonstrating its effectiveness in accurately identifying and delineating potholes from road images. The model achieved high precision (0.92) and recall (0.85), indicating a good balance between detecting potholes and minimizing false positives. Furthermore, the model's inference speed of 22 FPS ensures its suitability for real-time applications, which is crucial for deployment in vehicle-mounted systems for automatic pothole detection.Despite its strong performance, there are areas for improvement. The model occasionally missed smaller potholes or those partially obscured by debris (false negatives) and detected road markings or shadows as potholes (false positives). These challenges highlight the need for further fine-tuning, additional training data, and possibly incorporating advanced techniques such as multi-scale detection to improve accuracy.In conclusion, the YOLOv8 nano segmentation model provides a reliable and efficient solution for pothole detection, with potential for real-world applications. However, continued enhancements are needed to address edge cases, such as small potholes and visual clutter, to make the system more robust in diverse road conditions.

V. REFERENCES

[1]. Shaghouri, Anas Al, Rami Alkhatib, and Samir Berjaoui. "Real-time pothole detection using deep learning." *arXiv preprint arXiv:2107.06356* (2021).

[2]. J. Świeżewski, "YOLO Algorithm and YOLO Object Detection: An Introduction," 3 January 2020. [Online].Available:https://appsilon.com/object-detection-yolo-algorithm/.

[3].Ravichandran, M., Kumar Laxmikant, and Annamalai Muthu. "Efficient Vehicle Detection and Classification using YOLO v8 for Real-Time Applications." In *2023 Global Conference on Information Technologies and Communications (GCITC)*, pp. 1-5. IEEE, 2023.

[4].Farzad, "Pothole Segmentation YOLOv8 Dataset," Roboflow, 2024. [Online]. Available: https://universe.roboflow.com/farzad/pothole_segmentation_yolov8/dataset/1/images/ad622f815165a84798a7922e02a02876?split=train. [Accessed: 12-Nov-2024].

[5]M. E. N. Gomes, "The positions of sensors," *ResearchGate*, 2018. [Online]. Available: https://www.researchgate.net/figure/The-positions-of-sensors_fig3_328227731. [Accessed: 19-Oct-2024].

[6]."A thorough breakdown of YOLOv4," *Roboflow Blog*, Apr. 2024. [Online]. Available: https://blog.roboflow.com/a-thorough-breakdown-of-yolov4/. [Accessed: 19-Oct-2024].

[7]. J. Doe, "Pothole detection image generated using YOLOv8," *Medium*, [Online]. Available: https://miro.medium.com/v2/resize:fit:900/0*CKEc4j27kiRRJFJ-.jpg. [Accessed: Nov. 12, 2024].