

Development of an AI-Based System for Real-Time Pothole Detection, Severity Classification and Volume Estimation in Kenya

Kiprono Theophilus Ng'etich, Rono Brian Kiplangat, Stanly Kiprop, Wanjira Brian Ng'ang'a
Department of Civil and Structural Engineering
University of Eldoret, Kenya

Abstract: This research focuses on the development of an AI-based pothole detection system to improve road maintenance and safety in Kenya. Traditional pothole detection methods rely on manual inspection and outdated techniques, which are time-consuming, inefficient, and prone to errors. The developed AI-based system employs the YOLO (You Only Look Once) object detection framework to conduct real-time pothole detection. Data augmentation techniques such as rotation and flipping, brightness adjustment and blur effects were used on images during image preprocessing to enhance model reliability enabling it to perform well under different environmental conditions, lighting variations, and road textures. The AI system integrates the following key functionalities: Pothole detection, severity classification following the Kenyan Road Design Manual Volume 5, Part 1: Pavement Condition Survey, GPS tagging and volume estimation, and predicting the amount of materials needed for pothole repairs. The model was implemented by PyTorch on Ubuntu, and multiple performance metrics were used to evaluate the model's effectiveness, including: Mean Average Precision (mAP@50 & mAP@50-95), Intersection over Union (IoU), Precision, Recall, F1 Score and Inference Time. We built a web-based application that incorporates the trained AI model to detect and classify potholes in real time through a user-friendly system. The application allows users to both upload pictures and stream real-time video images for performing automated pothole identification.. The developed pothole detection system reduces manual work by delivering rapid and precise monitoring of road conditions. When implemented in Kenya, this system will improve road safety while lowering vehicle damage and increasing repair operation efficiency, thereby creating meaningful impacts on road management.

Keywords: AI, Image Annotation, Mean Average Precision (mAP), Potholes, YOLOv8

1. INTRODUCTION

The condition of road infrastructure is crucial to economic growth and the safety of transportation systems. Kenyan roads play a major role in moving goods and people, supporting commerce, healthcare, and education across urban and rural areas. However, potholes have become a significant issue by increasing transportation costs and leading to frequent accidents and vehicle damage. Potholes are bowl shaped depressions or cavities that form on the surface of flexible pavements, as a result of Loss of material in the wearing course and underlying pavement layers mostly in the wheel path. Potholes signify structural failure induced by traffic action [1].

Traditional methods of detecting and addressing potholes often rely on periodic manual inspections, which are resource-intensive, time-consuming, and can be inaccurate due to human error.

With advances in technology, Artificial Intelligence (AI) and machine learning have shown promising potential for improving infrastructure management by automating pothole detection. Using AI, combined with computer vision and sensor data, offers an efficient, cost-effective approach for identifying, classifying, and locating potholes. This research aims to explore an AI-based system that integrates image processing and computer vision to detect potholes on Kenyan roads, offering a scalable solution to enhance road maintenance and safety.

2. LITERATURE REVIEW

This section reviews the various techniques used currently or in the past for pothole detection and size estimation. In Kenya pothole severity is measured by assigning a pothole/patching index according to the KRDM part 5. This is done by manual inspection of the affected road and measuring the average size of potholes and the percentage area of the road affected for a 100m stretch in the case of a preliminary survey or a 50m stretch for a rehabilitation design.

The process itself must be done either from a car travelling at low speeds or on foot. This being the main method done in Kenya is time consuming at a very high operational cost.

A smartphone-based road roughness estimation system using accelerometer and gyroscope data was developed in 2014. This study focused on mapping road surface conditions using the International Roughness Index (IRI). This system achieved promising results in differentiating smooth and rough roads. It faced challenges in data accuracy which was influenced heavily by vehicle type and smartphone mounting position [2].

Another study [3] suggested the use of a low-cost Light Detection and Ranging (LiDAR) sensors that improves the detection and quantification of potholes on road surfaces. The algorithm uses curvature-based analysis to detect potholes in spatially thinned, structured LiDAR datasets and assesses their size through boundary delineation and voxelization. Although innovative and cost effective the detection accuracy dropped at low reduction levels.

Reference [4] used Machine Learning (ML) methods to carry out pothole detection and dimension estimation. The paper used K-nearest neighbour (KNN) in building the detection model. The study used both Manhattan and Euclidean distance measures to estimate the dimensions of the detected potholes. The detection model achieved 100 % accuracy in training and 89 % in testing using 150 and 50 images, respectively.

Another research [5] proposed, used the Mask Region-Based Convolutional Neural Network (Mask R-CNN) approach to detect potholes from 2D images. The Region of Interest (RoI) was identified in the pothole images which was then used to estimate the areas of potholes. After testing the results showed that the approach attained an accuracy of 90 % between the estimated areas and the actual area.

In summary the past studies have yet to suggest an accurate and cost-effective solution to the measurement of size and area of potholes especially in the case of Kenyan roads. This paper proposes the use of AI as the main evolving technology and integrating it with the rules of the Kenya Road Manual part 5 in volume estimation to generate a quick and economic alternative to pothole detection without sacrificing the accuracy.

3. METHODOLOGY

This section provides a detailed approach on the methods, tools, data, and processes used to achieve our objectives. The methodology includes data collection, data preparation and augmentation, model training, testing, system development, and evaluation phases.

3.1 Data Collection

The model is based on computer vision; it therefore needs images of high quality for training and testing purposes. The data used included images of potholes captured using cameras and mobile devices. Additional pothole images were collected from Kaggle (Online platform hosting a large collection of public datasets uploaded by researchers, organizations (accessible through the link: <https://www.kaggle.com/datasets>) this helps improve the accuracy of the model.

3.2 Data preparation and augmentation

This process involved preparing the images for training before feeding them into the YOLOv8 model. The preparation process involved image annotations, and data augmentation.

3.3 Annotation

In machine learning and deep learning, image annotation is the process of labelling or classifying an image using text, annotation tools, or both, to show the data features you want your model to recognize on its own [6]. When an image is annotated, metadata is added to a dataset. Annotating the pothole images creates ground-truth labels for training. Bounding boxes were drawn around potholes, and each bounding box was labelled with the class "pothole." This process ensured that the model would be able to learn how to detect potholes in various conditions and positions within images.

Tool Used:

CVAT: A popular image annotation tool that allows users to annotate objects within images and save the labels in YOLO format (.txt files corresponding to each image).

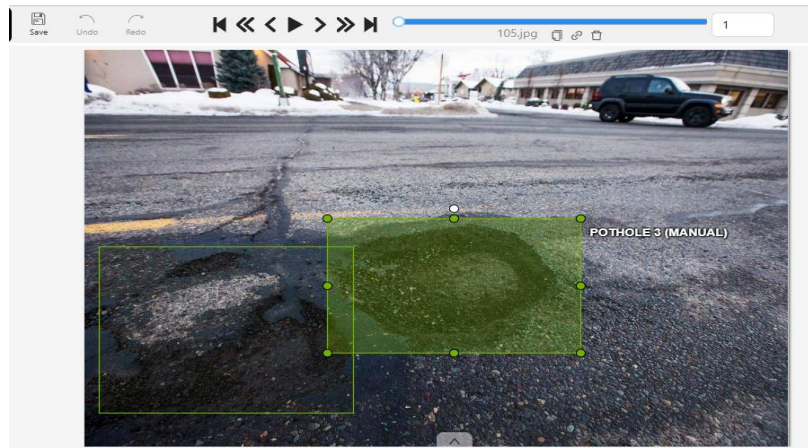


Figure 1 Annotation with CVAT

3.4 Data Augmentation

This process consists of applying different modifications to data items in order to expand the dataset and boost its diversity. The set of transformations in data augmentation encompasses multiple options such as rotations, flips, scaling and brightness adjustments alongside cropping and various others. The main purpose is to enhance generalization while protecting against overfitting through expanded data exposure before collecting additional data. The following augmentation techniques were applied

Rotation

The test images underwent random rotation by 45 degrees to simulate camera positioning changes. fig 2 shows a rotated image.



Figure 2 Rotation

Flipping

The images underwent horizontal and vertical flipping operations. fig 3 shows a flipped image.



Figure 3 Flipped Image

Brightness Adjustment

The brightness technique modifies the intensity of lighting across images. This helped the model learn to detect potholes under diverse illumination settings. fig 4 below shows a bright image.



Figure 4 Bright image

Gaussian Blur

By blurring, this reduces the sharpness of an image by softening the transitions between pixel colours. As a result, the edges become less defined, making the image appear unclear or out of focus. fig 5 shows a blurred image.



Figure 5 Blurred Image

Gray Scale

The images underwent grayscale transformation to remove colour information while retaining intensity values thus creating a monochromatic representation. fig 6 shows a grayscale image.



Figure 6 Gray Scaled

3.5 Model Development and Training

This section outlines the development and training of the system for real-time pothole detection and volume estimation, integrating advanced computer vision techniques. The system uses the YOLOv8 architecture and PyTorch framework to identify, classify, and estimate the dimensions and volume of potholes using image data.

3.6 Model Architecture

YOLOv8 (You Only Look Once v8) was chosen for its superior object detection capabilities and real-time processing performance. It operates using a Convolutional Neural Network (CNN) architecture with 24 convolutional layers followed by head layers for detection. It features spatial pyramid pooling for handling flexible input dimensions, making it adaptable to various real-world images. To avoid the tedious work of developing a model from scratch a pre-trained YOLOv8 model from GitHub was obtained. By selecting a pre-trained YOLOv8 model, it reduced training time and it provided access to its extraction features. We utilized our custom pothole dataset including images from Kenyan roads to refine the obtained YOLOv8 model.

3.7 PyTorch Framework

The model was developed through PyTorch which stands as an open-source deep learning framework created by Facebook's AI Research lab (FAIR). It was chosen for its dynamic computation graphs, ease of debugging, and integration with YOLOv8 by Ultralytics. All training, evaluation, and model management tasks were executed within the PyTorch environment.

3.8 Model Training

This section outlines the tools, techniques, configurations, and environment used during model training. It was conducted on Ubuntu 24.04 LTS using Python 3.13 and PyTorch. Despite the absence of a GPU, training was completed using small batches and a moderate number of epochs. The dataset comprised annotated pothole images which were split: 80% for training and 20% for validation/testing. This included images without potholes to minimize false positives and false negatives. Training used a custom configuration file (dataset.yaml) for paths and class labels. Throughout training, real-time feedback was displayed in the terminal, including loss values, mAP (mean Average Precision), and classification metrics. YOLOv8 automatically handled early stopping, validation, and saving the best model checkpoint as best.pt.

Table 1 Hyper parameters and Configuration

| Parameter | Value |
|--------------------|------------------|
| Epochs | 50 |
| Batch size | 8 |
| Input image size | 640 x 640 pixels |
| Anchor box setting | Dynamic |
| Training duration | ~ 3 hours |

3.9 Pothole Severity Classification

The second objective of the project was to classify the potholes based on severity ranging from small potholes which are of low concern to the public, to large or severe potholes which are of a huge nuisance. The project integrates both OpenCV and YOLO which is used to classify potholes and assess their severity based on visual data. The process will involve analysing the shape, size, and depth of potholes in images or video streams.

3.9.1 Detection and Bounding Box Analysis

YOLO is used for pothole detection and one of the key features of YOLO is the ability to draw bounding boxes once it identifies a pothole. YOLO, using the trained model, draws bounding boxes around detected potholes and provides its dimensions inform of coordinates.(x_min, y_min, x_max, y_max). From these coordinates:

- width = x_max - x_min
- height = y_max - y_min

Next the bounding box dimensions are used for the initial area estimation and are used in preliminary classification.

Bounding box area = width× height

Bounding boxes are helpful but may overestimate the actual size of irregularly shaped potholes.

3.9.2 Refinement with OpenCV

Since YOLO only provides bounding boxes the shape and size of the potholes will often be irregular. To improve the accuracy of the size estimation, OpenCV's image processing techniques were applied on the regions identified by

YOLO. Contour detection **is used to** isolate the exact pothole shape. OpenCV detects the contours within the bounding box using edge detection and contouring techniques in its library. Afterwards is calculating the area of the contour as a more precise measure of the pothole's size using the following formula:

The number of pixels enclosed by the contour: $\text{area_pixels} = \text{cv2.contourArea}(\text{max_contour})$

The area in square meters was calculated using a known scaling factor: $\text{area_m2} = \text{area_pixels} * \text{PIXEL_TO_M2}$

3.9.3 Classifying Pothole Size

For classification, the diagonal length of each YOLO-detected bounding box was calculated using OpenCV. This diagonal is treated as a proxy for the pothole's diameter. This is based on Kenya Road Design Manual Volume 5 Part 1: Pavement Condition Survey

Table 2 Severity Rating based on Diameter

| Severity level | Diagonal/ Diameter |
|----------------|--------------------|
| Minor | $\leq 250\text{m}$ |
| Moderate | 251-500mm |
| Severe | $> 500\text{mm}$ |

3.9.4 Volume estimation

Once the size of a pothole is estimated and classified based on their severity (using bounding box diagonals), the next step is to estimate the volume of each pothole. This will involve combining the estimated size of the potholes with depth measurements. Here's how the process will work: After severity classification (based on bounding box diagonal), each pothole is assigned a standard depth:

Table 3: Severity Rating based on Depth

| Severity level | Assigned Depth (mm) |
|----------------|---------------------|
| Minor | 30 |
| Moderate | 75 |
| Severe | 100 |

The volume of the pothole was calculated using the formula:

$$\text{volume_m3} = \text{area_m2} * (\text{depth (mm)} / 1000)$$

This enables accurate material estimation for road repairs.

4. RESULTS AND DISCUSSION

This study aimed at coming up with a robust, reliable and real-time system for detecting potholes on Kenyan roads. YOLO, OpenCV, CVAT were utilized in developing the system.

In this section, the results and performance of the proposed AI-based pothole detection system are provided. The performance of the model was evaluated using several performance metrics such as precision, recall and mean average Precision (maP).

4.1 Performance metrics

4.1.1 Precision

Precision is the ratio of true positive predictions to the total positive predictions made by the model [7]. It shows the number of the potholes detected by the system that were actually potholes.

A high precision value indicates that the false positives are low therefore the model is more accurate in identifying a pothole correctly

4.1.2 Recall

Recall is a parameter that is used to determine the ability of the model to identify all the potholes correctly [8]. It is the ratio of the true positives to the sum of the true positives and false negatives.

A high recall value shows that the model correctly identifies most of the potholes available.

4.1.3 Mean Average Precision (mAP)

Mean Average Precision (mAP) is a metric used to evaluate object detection models. It is based on other metrics such as precision, recall and Intersection Over Union (IoU). Average Precision (AP) is computed as the weighted mean of precisions at each threshold. mAP is then calculated as the average of average precisions of each class [9].

The performance metrics of our model upon development and training are provided in the table below.

Table 3 Models Performance Metrics

| Performance metric | Value |
|--------------------|-------|
| Precision | 86.4% |
| Recall | 72.4% |
| maP@50 | 84.7% |

The precision-recall curve, as shown in fig 7 below, demonstrated that the model has high precision-recall rates. The highest value obtained was 84.7%. This further shows the model's excellent performance and effectiveness.

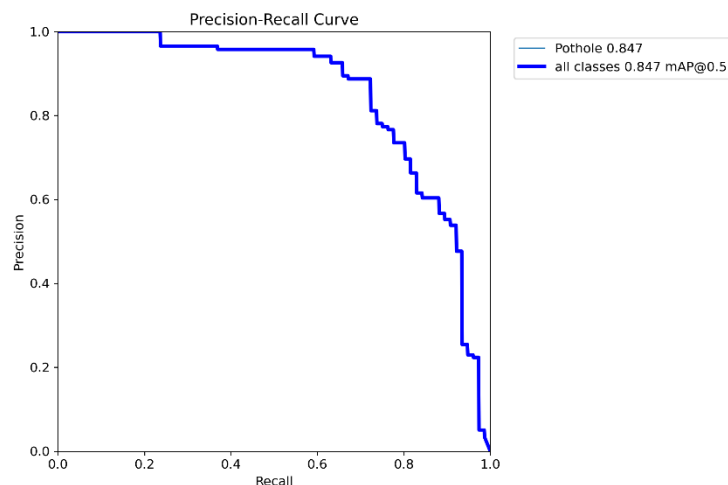


Figure 7 precision-recall curve

4.2 Detailed Pothole Classification

The model was able to provide more insights about the potholes such as their area. It detected the contours within the bounding box using edge detection and contouring techniques. The largest contour was selected as the likely pothole boundary. The model then proceeded to calculate the area of the contour as a more precise measure of the pothole's size.

To enable classification of the potholes based on their severity, the model was able to determine the diagonal of the drawn bounding box. It then compared the diagonals with the diameters as provided in Kenya Road Design Manual Volume 5 Part 1: Pavement Condition Survey and assigned severity ratings to the potholes as stated earlier.

To enable computation of the approximate volume of materials required for the repair of the potholes, the model assigned depths to the potholes while taking into consideration the severity of the potholes. The model then proceeded to calculate the volume of the potholes by utilising the area of the potholes determined earlier and the assigned depths. Fig8 shows potholes detected using the model.

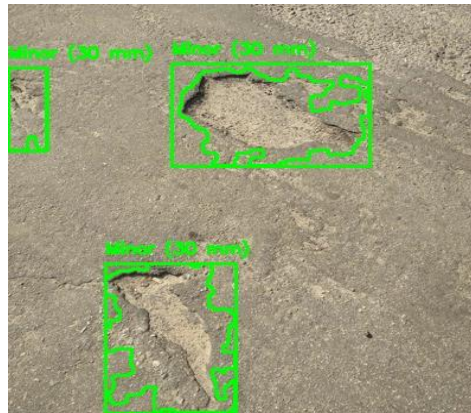


Figure 8 Detected Potholes

4.3 GPS Tagging for Pothole Localization

The model was also able to obtain the Global Positioning System (GPS) coordinates of the potholes so as to enable mapping of pothole locations for efficient road maintenance. By using GPS-enabled cameras and python geolocation libraries, the model was able to extract GPS data from pothole images and also during the live detection process. The model then proceeded to save the severity of the pothole, area (m^2), depth (mm), volume (m^3), latitudes and longitudes in a CSV file which can be easily downloaded for use.

5. CONCLUSION

An AI-based system for real-time pothole detection, severity-based classification, GPS extraction, and volume estimation of potholes was developed with the help of the YOLOv8 model and OpenCV. This solution fixes the problem of manual road inspections by using an automated system that is easy to use and saves costs.

6. REFERENCES

- [1] Ministry of Roads and Transport, Road Design Manual, Vol. 5 – Pavement Rehabilitation and Maintenance, Part 1 – Pavement Condition Surveys, RDM 5.1, Aug. 2023.
- [2] V. Douangphachanh and H. Oneyama, "A study on the use of smartphones for road roughness condition estimation," 2014.
- [3] A. Faisal and S. Gargoum, "Cost-effective LiDAR for pothole detection and quantification using a low-point-density approach," *Automation in Construction*, vol. 172, 2025.
- [4] P. Motwani and R. Sharma, "Comparative study of pothole dimension using machine learning, Manhattan and Euclidean algorithm," *Int. J. Innov. Sci. Res. Technol.*, 2020.
- [5] S. Arjapure and D. R. Kalbande, "Deep learning model for pothole detection and area computation," in *Proc. 2021 Int. Conf. Commun. Inf. Comput. Technol. (ICCICT)*, pp. 1–6, IEEE, 2021.
- [6] G. Boesch, "Labellmg for image annotation," VISO.AI, Feb. 11, 2022. [Online]. Available: <https://viso.ai/computer-vision/labelimg-for-image-annotation/>
- [7] Y.-M. Kim, S. Y. Son, S. Y. Lim, B. Y. Choi, D. H. Choi, et al., "Review of recent automated pothole-detection methods," *Applied Sciences*, vol. 12, no. 11, p. 5320, 2022. doi: 10.3390/app12115320.
- [8] A. K. Pandey, R. Iqbal, T. Maniak, C. Karyotis, S. Akuma, and V. Palade, "Convolution neural networks for pothole detection of critical road infrastructure," *Comput. Electr. Eng.*, vol. 99, p. 107725, 2022. doi:10.1016/j.compeleceng.2022.107725
- [9] Shah, D. (2022, March 7). Mean Average Precision (mAP) Explained: Everything You Need to Know. V7 Labs. <https://www.v7labs.com/blog/mean-average-precision>