



# Smart Pothole Detection System using Deep Learning Algorithms

Savita Chougule<sup>1</sup> · Alka Barhatte<sup>2</sup>

Received: 9 August 2022 / Revised: 3 July 2023 / Accepted: 17 July 2023 / Published online: 8 August 2023  
© The Author(s), under exclusive licence to Intelligent Transportation Systems Japan 2023

## Abstract

Potholes are a threat on roads, and their presence compromises driver, vehicle, and pedestrian safety. In developing countries, the primary reason for road accidents is bad road conditions, resulting in human life and property loss. In countries like India, Road maintenance is a challenging activity. Accidents rates are increasing year by year due to the up-surfing potholes count. This paper presents the system as "Forward View Guidance for pothole detection for Indian passenger Car." The camera captures video images, and a deep learning algorithm is used to classify the images as potholes and regular roads. The camera will also provide a view of the vehicle's front, highlighting the pothole. Deep learning YOLOv3 and YOLOv5 algorithms are used to train the model and tested with the Kaggle pothole detection datasets to predict the model's accuracy for detection. The proposed system will help monitor the road's condition, count the number of potholes on the road, and generate an alert signal. The performance of the proposed system is evaluated using precision, recall, and average precision (AP). The experimentation results show that the YOLOv5 algorithm performs best than the YOLOv3 algorithm. The precision, recall, and average precision (AP) for YOLOv5 are obtained as 0.763, 0.548, and 0.635, respectively. The system algorithm is implemented on the Raspberry Pi 4B model, which can be easily fitted as an add-on system in the vehicle.

**Keywords** ADAS · CNN · Potholes · Raspberry Pi · Yolo

## 1 Introduction

Roads have a significant role in the economy's overall growth. The majority of transportation is conducted on roads that are covered in asphalt, concrete, or both. Road conditions might contain a variety of flaws, including potholes, uneven manholes, cracks that make it difficult to slide, etc. Low-quality construction materials, a poor design that permits surface water build-up, ice in the cracks, etc., are responsible for pothole formation.

A car will shake more while going over any road surface irregularity, such as a pothole, crack, manhole, or expansion joint, than when traveling over smooth road surfaces. Earlier research mentioned that the pace of moving cars is used to identify road surface irregularities. Motion sensors pick up vibrations (e.g., accelerometers or gyroscopes)

while a vehicle moves on the potholes. The depth of the pothole is also detected using an ultrasonic sensor. In another research, Image processing analysis, such as texture extraction and comparison utilizing collected images exhibiting pavement degradation characteristics, is used in a vision-based technique. This method relies on geotagged images from a camera/video system mounted on a moving vehicle pointing downward toward the road surface. The collected geotagged video frames can automatically detect road surface distress characteristics, such as potholes and fissures. You Only Look Once (YOLO), Convolutional Neural Network (CNN), Support Vector Machine (SVM), Faster CNN, and other machine learning methods are used to identify the pothole in the images.

In India, the maintenance of roads is a significant challenge. Indian roads are full of potholes. During monsoons, Indian road conditions are even worse. Potholes are filled with water and mud, making it difficult for the driver to drive through. So, developing a system that will help drivers drive safely in potholes is necessary.

The main objective of this system is to detect potholes, count them, and provide a front view of the vehicle. To accomplish this following task has been performed.

---

✉ Savita Chougule  
1032200699@mitwpu.edu.in

<sup>1</sup> M-Tech Student, School of Electronics and Communication Engineering, MIT-WPU, Pune, India

<sup>2</sup> Assistant Professor, School of Electronics & Communication Engineering, MIT-WPU, Pune, India

- Review past research efforts on pothole detection and different techniques used to implement the same.
- Select the sources for accurate detection of potholes.
- Select the components for the hardware module.
- Select/develop the algorithm for detecting potholes using the camera.
- Select the various dataset for validation and testing of an algorithm
- Implementation of hardware.
- Validate the hardware and software

## 2 Literature Survey

Shaik Masihullah et al. [1] introduce an attention-based coupled architecture for pothole segmentation. The drivable regions in different urban and rural nations are poorly maintained. An Advanced Driver Assistant System (ADAS) is required to analyze the drivable area and warn the driver of potential potholes to prevent such scenarios and maintain vehicle safety. This data may also be used in organized systems to keep roads in good repair. This research uses a few-shot learning strategy for pothole identification to regulate accuracy even with less training data. KITTI and IDD datasets are adapted to do various experiments for road segmentation. They also presented pothole segmentation on IDD datasets. Various experiments were done to test the accuracy of the proposed frameworks. According to the IDD dataset, it segmented roads with an accuracy of 73.83% for potholes. The F1 score is used to evaluate the road segmentation on the KITTI dataset. Compared to the state-of-the-art, the F1 score is 95.21%, with an 81% reduction in run-time. This project's future work aims to finetune the model for dealing with road-scene light, such as road condition evaluation at night.

K. wang et al. [2] present a vision-based system for pothole detection using a smartphone camera. They used a deep CNN model to train the neural network and achieved an accuracy of 97% for color images and 97.5% for grayscale images. A deep convolutional neural network (CNN) based on the YOLOv2 technique was suggested by K. Lim and J. Kwon [3]. According to the author, the performance of this approach was significantly improved over YOLOv2 from 60.14% to 82.43% precision. Better outcomes were obtained with the proposed F2-Anchor and Den-F2-Anchor models.

An unsupervised vision-based technique for pothole identification based on RGB color space picture segmentation was proposed by A. Akagic et al. [4]. The method is intended for daytime, fair weather settings and involves modification of the B component in RGB space. The method's accuracy was 82% when tested on 80 pothole images. A pothole detection system was created by S. Ryu et al. [5]. It contained an optical device installed on a car with data collection and storage features, Wi-Fi communication, GPS

position gathering, and a pothole identification algorithm with a classification accuracy rate of 73.5%.

Lin et al. put out a method to identify potholes [6] using a nonlinear support vector machine. This method makes use of texture measurement as a characteristic to identify potholes. A partial differential equation PDE model is utilized for image segmentation, and for classification, a nonlinear SVM tool is applied. When the recommended model is trained using nonlinear SVM, it yields good pothole identification results. Some complex situations, like mud-filled potholes, are misidentified. Additionally, the entire pothole defect region is not shown.

Soubhik Das et al. [7] employ a frequency graph to concentrate the P3De pothole detection algorithm for pothole depth calibration. Pothole identification is often made in real-time footage that has been analyzed. The original video must be optimized for cameras and other sensors. To improve eyesight and perception, it's critical to understand the depth of particular roadway areas when driving. Finding the pixel change in the location of potholes on roadways is the first step in the suggested approach. The depth is calculated using shifting pixels and shown on frequency scales, which aids in detecting potholes. The KITTI stereo dataset is used to test the model. P3De is a rapid and practical strategy that may be used to solve complex and time-consuming problems.

Roopak Rastogi et al. [8] examine modern artificial neural network techniques to see how well they function. YOLO and faster R-CNN with VGG16 and ResNet-18 architectures are examples of pothole detection algorithms. Both algorithms are more accurate and quicker. To address the issue of the classes being imbalanced between a pothole and ordinary roads, an improved YOLOv2 design was given. Precision and recall were considered when evaluating their performance against other object recognition algorithms. The YOLOv2 architecture offered better precision. The Raspberry Pi may integrate the proposed architecture with a smartphone camera. For real-time pothole recognition, the proposed architecture might be installed on the dashboard of both manual and driverless vehicles.

Potholes are abundant on Indian roadways. An example of a disturbance in the road's surface is a pothole. Indian roads are in much worse shape during the monsoon. Potholes make driving challenging for vehicles since they are covered with water and mud. When a car struck a pothole, it suffered damage such as broken wheel rims, misaligned steering, and suspension.

From the literature survey following key points are observed.

- The sensor-based system requires low storage and can analyze the data in real time.
- The disadvantage of a sensor-based system is that the sensor may get damaged due to road conditions.
- Vision-based systems are accurate, intelligent, fast, dependable, and can detect any size pothole.

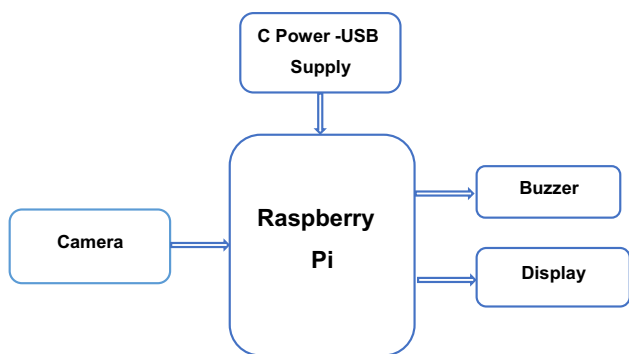


Fig. 1 Hardware Block Diagram of the pothole detection system

- For various traffic conditions, the vision-based system needs to modify several parameters. Its computational complexity is high.
- AI-based pothole systems are more precise, requiring more data and processing power.
- The AI-based system required extensive data to train the model and more training time.

### 3 Proposed Methodology

The proposed system consists of Raspberry Pi 4B as the central processor. The Camera module is used to capture the real-time video stream. The captured video stream is processed in the Raspberry Pi module. The pothole detection algorithm detects the potholes from the video stream. This system uses YOLOv3 and YOLOv5 algorithms to detect potholes. The display is provided to show the output of the detected potholes. Figure 1 shows the proposed hardware block diagram of the pothole detection system.

Fig. 2 Pothole detection using yolov3 and Yolov5

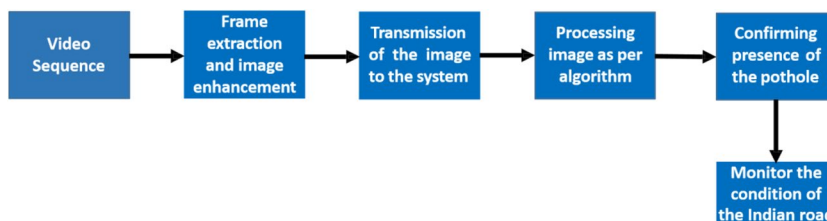


Fig. 3 Sample image of pothole Dataset



Table 1 Dataset Distribution

Total Image	Train Images	Validation Images	Test Images
665	465	133	67

In the software part of this system, YOLOv3 and YOLOv5 algorithms are used to detect the potholes from the camera. The detailed block diagram of the proposed system is presented in Fig. 2.

#### 3.1 Input Video

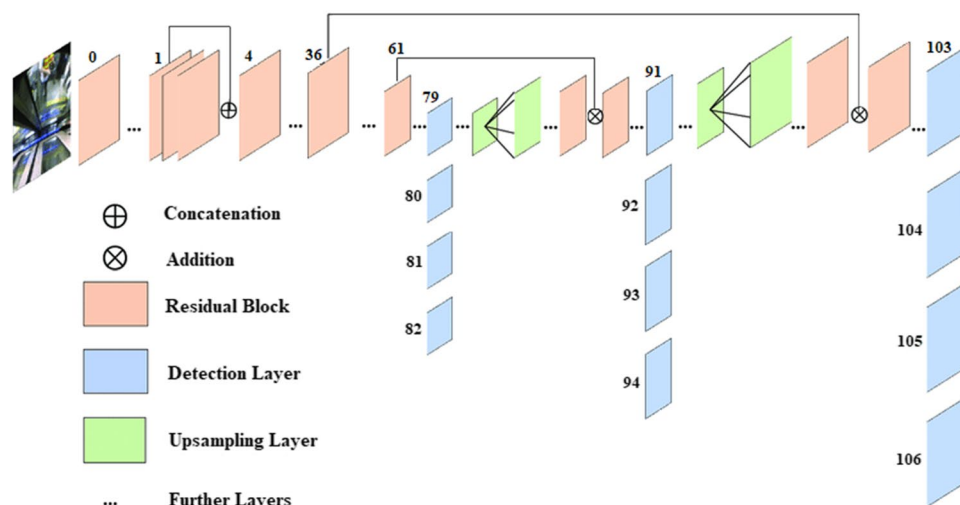
This system collects the pothole dataset from the Kaggle website [9]. Atikur Rahman Chitholian collected and shared the dataset as part of his undergraduate thesis, and it was initially posted on Kaggle. They captured a portion of the images using an Android phone camera from some damaged roads. Another portion is web-scraped (from Google Image Search). To generate annotations (XML files), we used label images from the GitHub source [10]. Sample images of the dataset are shown in Fig. 3.

The whole dataset is split into training, validation, and testing. The dataset distribution is shown in Table 1.

#### 3.2 Frame Extraction and Video Enhancement

The video is nothing but a sequence of the frame. The image or computer vision algorithms are working on the frames; hence need to extract the frame from the video sequences. The number of extracted frames is depended on the video frame rate. Most video captured in real-time has a frame rate of 30 FPS. The salt and pepper sound significantly impacted the footage that was shot. Therefore, the salt and pepper noise is eliminated using a median filter. Median

Fig. 4 Yolo V3 architecture



filter smoothing out the input frame to help further the classification process.

### 3.3 Training and Classification

This approach uses YOLOv3 and YOLOv5 algorithms to train and test the system for pothole detection. A detailed explanation of YOLOv3 and YOLOv5 algorithms is present in this section.

#### a) YOLOv3

YOLOv3 is a real-time object identification system that detects particular objects in movies, live streams, and images. YOLO uses characteristics learned by a deep CNN to detect an object. Joseph Redmon and Ali Farhadi created the first three versions of YOLO. To begin, YOLOv3 employed a modified version of Darknet, which was trained on ImageNet and contained a 53-layer network. YOLOv3 has a 106-layer fully convolutional base architecture, with 53 extra layers added for detection. This is due to the slower performance of YOLOv3 compared to YOLOv2.

The YOLOv3 architecture's convolutional layers offer a detection prediction after delivering the learned features to a classifier or regressor. These characteristics include the class label, bounding box locations, bounding box sizes, and more.

YOLOv3 and later versions can grasp this prediction map since each cell predicts a certain number of bounding boxes. The cell in charge of predicting the item is then identified as the cell containing the core of the ground truth box for the item. The underlying workings of the prediction architecture are primarily dependent on mathematics. The YOLOv3 architecture is shown in Fig. 4.

#### b) YOLOv5

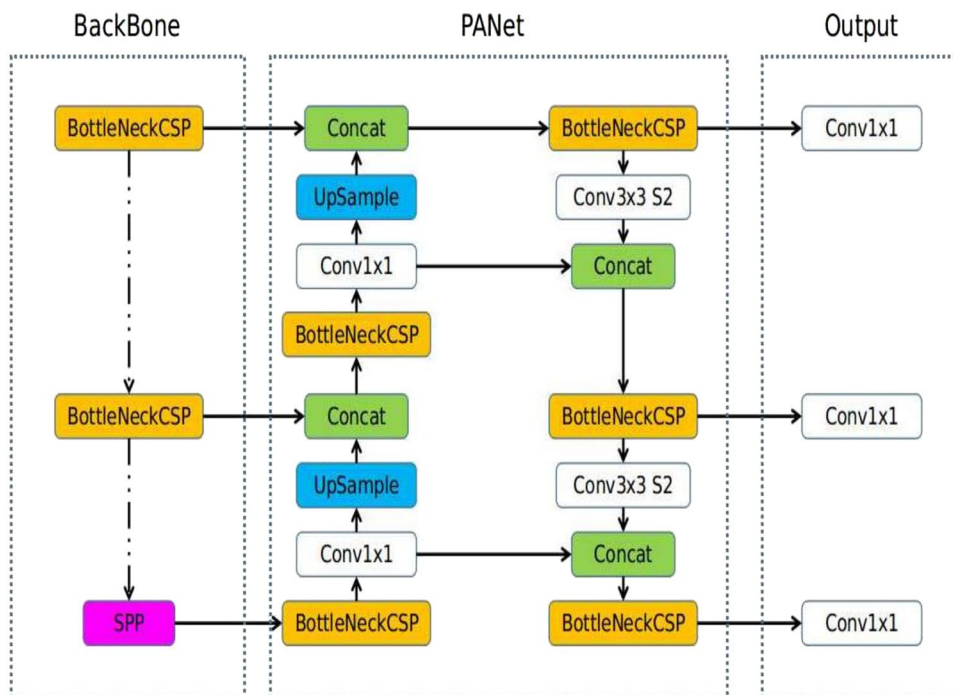
In the year 2020, a firm named Ultralytics launched YOLOv5. Glenn Jocher, Founder & CEO of Ultralytics, put it on a GitHub repository and quickly gained popularity. The YOLOv5 object detection model was also released on the iOS App Store as "iDetection" and "Ultralytics LLC," respectively. Even though the Ultralytics website on GitHub claims to be the most up-to-date of all known YOLO implementations, the version's credibility has been questioned in the community. The YOLOv5 object identification model's architecture is shown in Fig. 5.

Because YOLOv5 is a single-stage object detector, it includes three key components that any other single-stage object detector would have.

- **YOLOv5 Backbone:** To extract features from images made out of cross-stage partial networks, it employs CSPDarknet as the framework.
- **YOLOv5 Neck:** It uses PANet to construct a network of feature pyramids for feature aggregation and forwarding to the Head for prediction.
- **YOLOv5 Head:** layers for object detection that use anchor boxes to generate predictions.
- **Activation and Optimization:** In YOLOv5, leaky ReLU, and sigmoid activation are employed, with SGD and ADAM potential optimizers.
- **Loss Function:** Binary cross-entropy with logits loss is used.

The model trained by YOLOv3 and YOLOv5 is used for real-time testing. In testing, the input videos are captured from the video camera. The trained model will extract, preprocess, and test the frame. The pothole will be detected from the input frame and shown by bounding boxes.

Fig. 5 Yolov5 architecture



### 4 Hardware and Software Specifications

This system is built on the Raspberry Pi hardware platform. The camera is attached to the USB port and placed in front of the car to detect the pothole. The display is mounted inside the car to visualize the results and activity. The algorithms of the proposed system are developed in python language. The hardware and software specification of the proposed system is explained in this section.

#### 4.1 Hardware Specifications

a) Raspberry Pi 4 Model B

A screen, keyboard, and mouse are all included in the little Raspberry Pi computer. People of various ages may use this little yet functional device to learn about programming languages like Python. The hard drive for the Raspberry Pi is an SD card inserted into a slot on the device. The graphical output may be watched on a monitor linked through an HDMI connector. A Raspberry Pi 4 is a small computer simulating hard drives using USB storage and very little electricity.

b) USB Camera

The video feed is captured using this Logitech 310 USB camera module. It supports Full HD Widescreen video calling at 30 frames per second in HD 720p. The 60-degree

field of view lens captures all of the action. It automatically improves your image's warmth and balance for any setting, so it appears best even in low-light situations. It features a built-in noise-canceling microphone.

c) Raspberry Pi Touch Display

With the 7-inch Raspberry Pi Touch Display, users may create integrated, all-in-one projects like tablets, infotainment systems, and embedded software. An adapter board transforms power and signals for the 800 × 480 display.

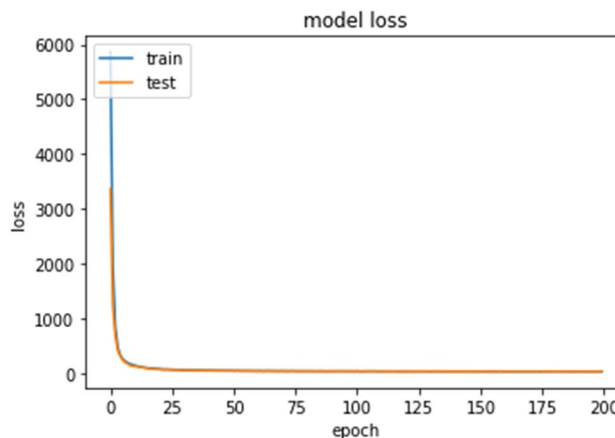
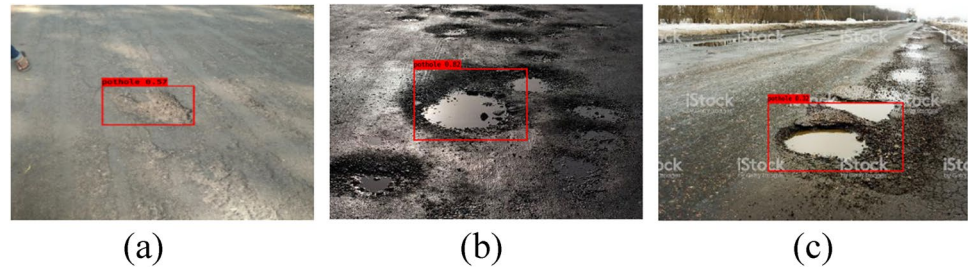


Fig. 6 Training loss of YOLOv3 model

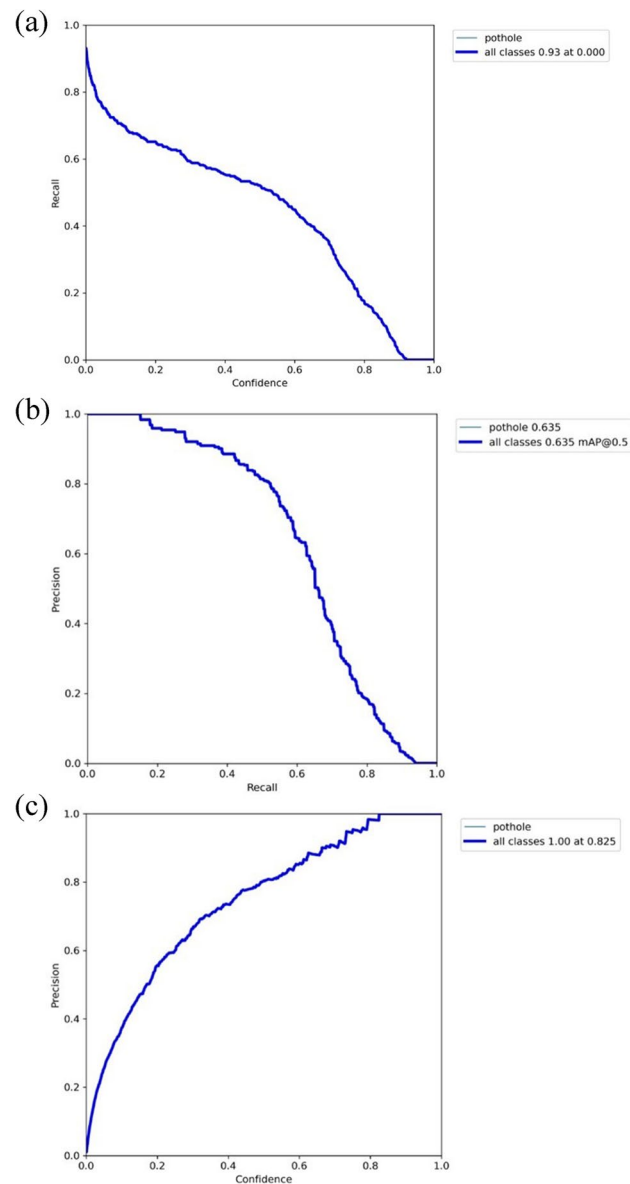
**Fig. 7** Qualitative analysis of the proposed system using the Yolo V3 model



## 4.2 Software Specifications

Python is a high-level programming language with a wide range of applications. Python is an interpreted language that supports

a variety of programming scripts as well as a syntax that allows you to utilize programs written in various languages, including C++ and Java. The language has constructs that allow for explicit programming at all scales. Python is simple to learn and use, and writing Python code is much easier than writing code in other languages. In addition, the TensorFlow and PyTorch library is used to train the dataset image for pothole detection.



**Fig. 8** Performance plot of the YoloV5 model (a) Confidence Vs. Recall (b) Precision Vs. Recall (ROC) (c) Confidence Vs. Precision

## 5 Results

The results of the proposed system are presented in this section. In this system, two algorithms, i.e., YOLOv3 and YOLOv5, are used to train and test the pothole images. The loss function categorizes input data points in a dataset and calculates a predictor's performance. The classifier predicts the link between the input data and the output target more accurately, the smaller the loss value is. The progressive learning process of YOLOv3 is represented by the continuous drop of the loss value following each epoch in Fig. 6. After five epochs, the curves produced by the YOLOv3 loss function become relatively stable.

Figure 6 shows that the training loss of the YOLOv3 model is lower at every progressing epoch. The loss of the model is calculated using Mean Square Error (Error). From the graph, we can conclude that the YOLOv3 model achieved better results. The testing results of the YOLOv3 model on the testing image are shown in Fig. 7

From Fig. 7, it is observed that the YOLOv3 algorithm detects the nearer potholes correctly. But in some cases, the algorithm failed to detect the pothole; hence we used the YoloV5 algorithm to detect the potholes. The YOLOv5 algorithm is trained on the same dataset for training the YOLOv5 algorithm. The performance plot of the YOLOv5 model is present in Fig. 8

The full-topology confidence score matches quite well with the observed recall for the pothole detection in the dataset, as shown in Fig. 8(a). Like the confidence scores, the whole confidence score should be read with the premise that the pothole was detected accurately.

The accuracy vs. recall graph of YOLOv5 training is shown in Fig. 8(b). Curves represent the trade-off between true positive and false positive rates for a statistical model with variable probability thresholds. Evaluate the area under the curve as accuracy during training. To capture

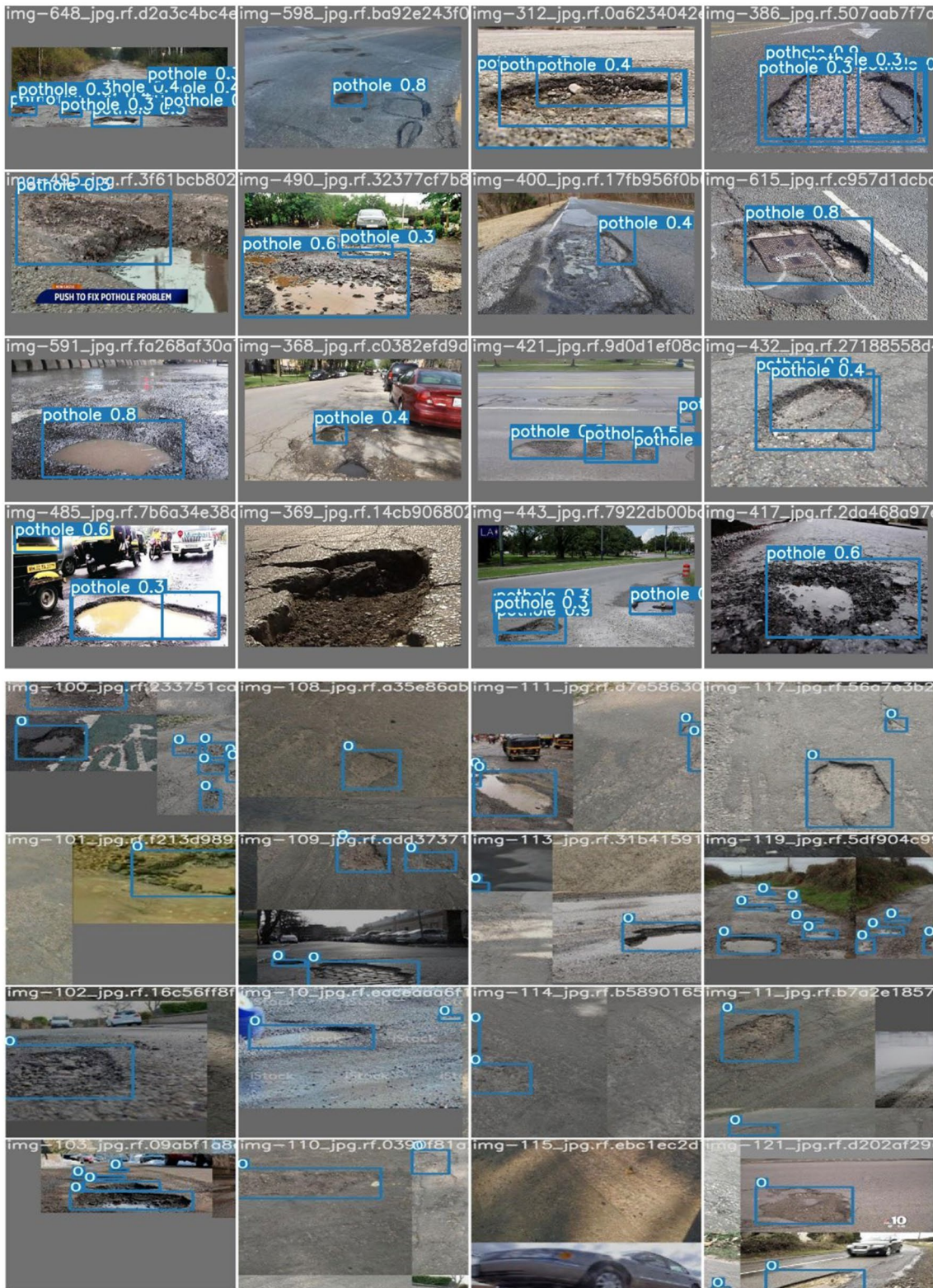
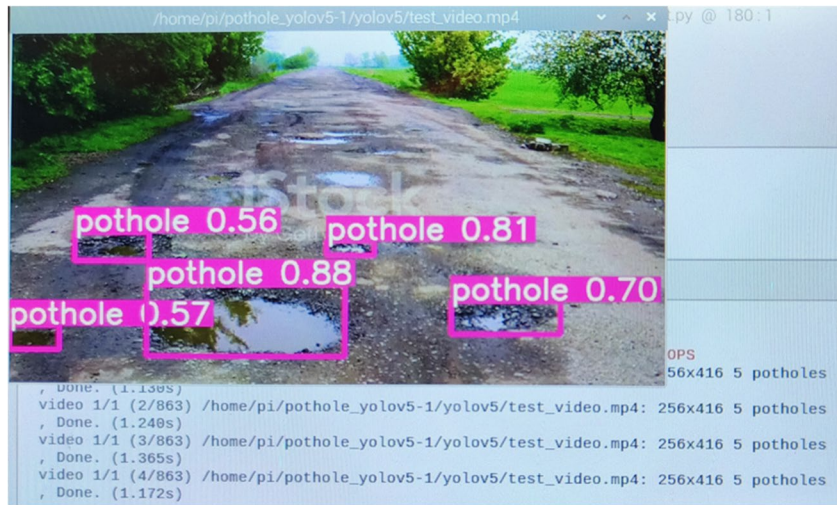


Fig. 9 Testing results of the Yolo V5 model

**Fig. 10** Detection time of potholes



**Fig. 11** Camera to Potholes distance

the whole AP, the curve should ideally travel from  $P = 1$ ,  $R = 0$  at the top left to  $P = 0$ ,  $R = 1$  on the bottom right (area under the curve). The model can be run by adjusting the threshold. It is observed that the area under the PR curve is large; hence it will show better accuracy.

The full-topology confidence score matches quite well with the dataset's observed accuracy for pothole identification, as

**Table 2** Results Of Real-Time Video

Model	No. of potholes	No. of correctly detected potholes	No. of incorrectly detected potholes	Accuracy
YOLOv3	36	11	25	0.3055
YOLOv5	36	35	1	0.9722

shown in Fig. 8(c). Like the confidence scores, the whole confidence score should be read with the premise that the pothole was detected accurately. The testing results of the YOLOv5 model on the testing image are shown in Fig. 9.

As soon as the pothole has been detected, the buzzer connected to the raspberry pi generates the alert sound for the driver. The distance between the pothole and vehicle can be adjusted by proper positioning/mounting of the camera. Figure 10 represents the potholes detected and time taken to detect these potholes is varying between 1 s to 1.5 s.

Consider the scenario where the camera mounted on a vehicle can capture the video at a distance of 30 m as shown in Fig. 11.

Assuming a vehicle running at a speed of 40 km/hr. The instance the pothole detected and alert is generated the time elapsed will be 1.5 s where the vehicle has already covered approximately 16 m. The driver still gets 1 to 1.2 s to control the vehicle speed. The further improvement in this scenario can be easily made by bringing down the pothole detection time to millisecond so that driver gets alert immediately after the pothole detection. This is possible by replacing the Raspberry Pi controller with NVIDIA Jetson Nano which has the performance and capabilities needed to run modern AI workloads and includes huge AI and Computer Vision libraries and APIs.

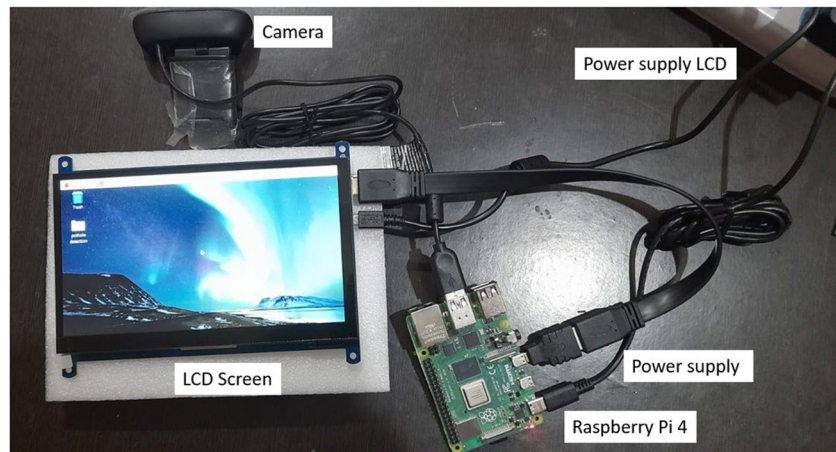
The training was done using 200 epochs for 665 images on YOLOv3 and YOLOv5. The Tensor board module was used to get the metrics for each method. The yolov3 and yolov5

**Table 3** Comparative Analysis of Proposed Yolov5 with Existing Algorithm

Algorithm	Precision	AP_0.5
YOLOv5 [proposed]	0.763	0.635
YOLOv5 Large [11]	0.8643	0.589
YOLOv5 medium [11]	0.8696	0.6154
YOLOv5 Small [11]	0.7673	0.589



**Fig. 12** Hardware module of the proposed system



algorithms are performed on the real-time video, and their results are tabulated in Table 2. YOLOv3 algorithms failed to detect small potholes hence the average accuracy is 0.3055, which is improved to 0.9722 using the YOLOv5 algorithm (proposed algorithm). Thus, yielding an effective method in the detection of the small potholes with greater accuracy.

The comparative analysis of the proposed system with the existing system is tabulated in Table 3.

Table 3 gives the comparison of the proposed algorithm with the existing algorithms. The comparative analysis shows that the proposed YOLOv5 algorithm shows better average precision (AP) than the existing YOLOv5 algorithm presented in [11] for pothole detection. The improved AP means the excellent prediction for all Intersection over Union (IoU) thresholds resulting in a robust system.

The proposed system is built on the raspberry pi module and tested in a real-time environment. The proposed system shows promising results. The raspberry pi hardware module becomes relatively slow because of the complexity of the algorithms. The picture of the hardware module of the proposed system is shown in Fig. 12.

## 6 Conclusion

Pothole guidance is a growing research field to improve vehicle safety and reduce road accidents. The proposed system is developed to monitor Indian road conditions by detecting the number of potholes on Indian roads. This will help the government to make a plan for road maintenance. Deep learning algorithms like YOLOv3 and YOLOv5 will improve object detection and classification accuracy. Major OEMs like Land Rover, Mercedes Benz, etc., are working on this system. The proposed system uses the YOLOv5 algorithm to train the network resulting in an accuracy of 97.22% for pothole detection. The qualitative and quantitative analysis shows

that the YOLOv5 model outperforms the YOLOv3 model for pothole detection in Indian road scenarios.

The main limitation of the proposed system is the processing speed of the proposed system is moderate because the processor of raspberry pi is slow for the deep learning algorithm. The processing speed can be improved more widely using the most powerful computer, like JETSON Nano, specifically designed for AI applications. Further proposed system's performance can be increased by increasing the dataset images of different road conditions on Indian roads. Also, the system can be trained by hyper tuning the training parameter of deep learning algorithms like Faster RCNN.

**Abbreviations** AI: Artificial Intelligence; ADAS: Advanced Driver Assistance System; ML: Machine Learning; IoT: Internet of Things; SVM: Support Vector Machine; KNN: K Nearest Neighbor; CNN: Convolutional Neural Network; R-CNN: Region-based Convolutional Neural Network.; YOLO: You Only Look Once

**Authors' Contribution** SC carried out literature review, contributed in data collection, analysis and interpretation of results, and was major contributor in manuscript preparation. AB helped in hardware implementation and final drafting of manuscript. All authors read and approved the final manuscript.

**Funding** N/A.

**Data availability** N/A.

## Declarations

**Ethics Approval** No approval required. Manuscript doesn't include any data related human participants, human data and human tissue.

**Consent for Publication** I, the undersigned, give my consent for the publication of identifiable details, which can include photograph(s) and/or videos and/or case history and/or details within the text ("Material") to be published in the above Journal and Article.

**Competing Interest** No conflict of interest

## References

- Masihullah, S., Garg, R., Mukherjee, P., Ray, A.: Attention based coupled framework for road and pothole segmentation. In: 25th International Conference on Pattern Recognition (ICPR) (Milan, 2021), pp. 5812–5819. <https://doi.org/10.1109/ICPR48806.2021.9412368>
- Kwang, A., Sung, L., Seung, R., Dongmahn, S.: Detecting a pothole using deep convolutional neural network models for an adaptive shock observing in a vehicle driving. In: 2018 IEEE International Conference on Consumer Electronics (ICCE) (Las Vegas, 2018), pp. 1–2. <https://doi.org/10.1109/ICCE.2018.8326142>
- Lim, S., Kwon, J.: Detection of potholes using a deep convolutional neural network. *J Univers Comput Sci.* **24**(9), 1244–1257 (2018)
- Akagic, A., Buza, E., Omanovic, S.: Pothole detection: An efficient vision based method using RGB color space image segmentation. 40th international convention on information and communication technology, electronics and microelectronics (MIPRO) (Opatija, 2017), pp. 1104–1109. <https://doi.org/10.23919/MIPRO.2017.7973589>
- Seung, R., Taehyeong, K., Young-Ro, K.: Image-based pothole detection system for its service and road management system. *Math Probl Eng.* **2015**, 968361, 10 pages (2015). <https://doi.org/10.1155/2015/968361>
- Jin, L., Yayu, L.: Potholes detection based on SVM in the pavement distress image. In: DCABES '10: Proceedings, ninth international symposium on distributed computing and applications to business, engineering and science (2010), pp. 544–547. <https://doi.org/10.1109/DCABES.2010.115>
- Das, S., Kale, A.: P3De - a novel pothole detection algorithm using 3D depth estimation. In: 2nd International Conference for Emerging Technology (INCET) (Belagavi, 2021), pp. 1–5. <https://doi.org/10.1109/INCET51464.2021.9456343>
- Rastogi, R., Kumar, U., Kashyap, A., Jindal, S., Pahwa, S.: A comparative evaluation of the deep learning algorithms for pothole detection. In: IEEE 17th India Council International Conference (INDICON), vol. 2020 (New Delhi, 2020), pp. 1–6. <https://doi.org/10.1109/INDICON49873.2020.9342558>
- Makeml.app. Available at: <https://makeml.app/datasets/potholes> and <https://www.kaggle.com/datasets/andrewmvd>. Database Contents License (DbCL) v1.0. Available online
- Fan, R., Ai, X., Dahnoun, N.: Road surface 3D reconstruction based on dense subpixel disparity map estimation. *IEEE Trans. Image Process.* **27**(6), 3025–3035 (2018). <https://doi.org/10.1109/TIP.2018.2808770>
- Ahmed, K.: Smart pothole detection using deep learning based on dilated convolution. *Sensors.* **21**(24), 8406 (2021)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.



**Savita Chougule** is a post-graduate in VLSI and embedded system from School of Electronics and Communication Engineering, MIT World Peace University, Pune, India. She is working professional and have 20 years of industry experience in the field of automotive development and customer care. Her areas of interest are automotive electronics, artificial intelligence and software defined vehicle.



**Dr. Alka Surendra Barhatte** is a post-graduate in Electronics-VLSI Design from Bharati Vidyapeeth Deemed University, Pune, India and Doctorate in Electronics and Telecommunication from Savitribai Phule Pune University. She has been working with School of Electronics and Communication Engineering, MIT World Peace University, Pune as Assistant Professor. She has 16 years of teaching experience and eight years of research experience in the field of soft computing, Artificial Neural Network and biomedical signal processing. Her areas of interest are automotive electronics and AI-ML.