A Smart App for Pothole Detection Using Yolo Model

Rajshekar Hiremath, Komal Malshikare, Manish Mahajan, and Radhika V. Kulkarni

Abstract Pothole is the structural failure on the road, which causes accidents. In India, due to an increase in transportation, the number of mishaps because of potholes has additionally expanded. In this way, for diminishing the loss of human life because of potholes a few techniques has been conceived to identify the potholes utilizing sensors. These techniques are exorbitant and inefficient. So we have structured a savvy approach which utilizes cell phones with camera and GPS sensors. Here, we are using “YOLO object detection” algorithm to detect potholes. The application detects the location of a pothole. The users can upload images of potholes in their area. After uploading, the YOLO algorithm validates the given image. Then, the location of the pothole is displayed on the map. Civic authority in that area can repair the potholes. So in this strategy, we are executing the tech-savvy and sustainable answer for pothole identification, and this technique can successfully identify street road conditions utilizing the cell phone.

Keywords Convolution neural network · Global positioning system · Image classification · Object detection · Roads safety · Pothole detection · YOLO model

R. Hiremath (✉) · K. Malshikare · M. Mahajan · R. V. Kulkarni
Department of Information Technology, Pune Institute of Computer Technology, Pune, India
e-mail: rajshekar2198@gmail.com

K. Malshikare
e-mail: kg.malshikare@gmail.com

M. Mahajan
e-mail: manishsm11@gmail.com

R. V. Kulkarni
e-mail: radhikavikaskulkarni@gmail.com

S. Fong et al. (eds.), *ICT Analysis and Applications*, Lecture Notes in Networks and Systems 154, https://doi.org/10.1007/978-981-15-8354-4_16

1 Introduction

Transportation is the leading sector in every country. Due to the increasing number of vehicles, the likelihood of mishaps in India has expanded. Road surface conditions affect transport safety and driving comfort. The user should be aware of the road conditions for safety purposes. A few strategies [1–6] have been proposed, however, are inefficient and expensive. There is no computerized framework to recognize the potholes. Hence, civic authority and citizens both have to face the challenges [1]. The severity caused by the potholes is largely unnoticed. Daily, death of ten people is reported due to accidents caused by potholes in India [7]. In 2017, about 3597 deaths and 25,000 injured were reported in accidents due to the bad road conditions [8]. Over 9300 deaths have been reported due to potholes in the last three years [8]. This causes loss of human life and damage cost to the roads and the vehicles involved.

Hence, we have thought of a sustainable solution to counter this problem by developing a smart user-friendly app for people to detect the road surface conditions. Our solution is fast, accurate, inexpensive to operate, and utilizes a smartphone's camera and GPS. By applying the YOLO algorithm [9], the proposed method efficiently detects multiple potholes in the image and accordingly defines the road surface conditions.

2 Related Work

Many researchers have contemplated different techniques to distinguish the potholes. Some methods [1, 2] are very difficult to use, some [3, 4] have very low accuracy, and some [5] perform only in favorable conditions. Some methods [10, 11] give good accuracy but are inefficient due to high computation and expensive techniques.

The study in [1] proposes a low-cost sensor and CNN-based method for automatic pothole detection. This method is unable to detect the pothole image under the condition of illumination variation [1]. Some researchers describe sensor-based methodologies to detect the potholes [2, 5]. The study in [2] introduces road damage detection using ANN and accelerometer, with images captured through a smartphone mounted on the car with its GPS and accelerometer activated. This method detected and classified the three types of roadway anomalies and showed a high degree of accuracy, but did not detect potholes. Piao and Aihara [5] use sensors which present a vibration-based system for pothole detection. They are using mobile sensors that contain both accelerometer and GPS. Here, high error rate is noticed because this method is more physical than the digital one.

Some methods determined the road conditions according to the roughness of the road. The research work in [3] shows that the root mean square of the vertical component of the acceleration has a high correlation with IRI. Using this, approximate roughness of the road can be detected. Here, the limitation is that the parameter must be manually adjusted for each vehicle. Yagi [4] describes a spring and damper model. It can automatically estimate vehicle parameters including the damping ratio and resonant frequency which can be used to detect the road roughness. This study estimates the road roughness index and identifies changing road conditions.

The research work in [11] suggests creating 3-D images from two 2-D images captured by two different cameras aligned in a specific manner. Hence, potholes can be detected based on their geometric shape. It provides high accuracy but it is highly dependent on the orientation of cameras on vehicles. Also, a lot of computational power is required to create initial 3-D images [11].

3 Proposed Methodology

3.1 Android Application and Google Maps API

In the proposed methodology, we are using an android application along with Google maps API. Android applications are widely used by people all over the world due to its easily accessible platform. This app is used in our method, as it is the most easily accessible device in one's pocket and can be used by a number of people widely over the world [2].

Google APIs permit the use of Google services. One such API is an embedded Google map. It can be accomplished by utilizing the Static maps API, Places API, or Google Earth API. This map API is utilized in our method. Google maps API helps us to integrate the maps into our system.

3.2 YOLO Model

YOLO stands for "you only look once." This is the fastest algorithm that even can be used for object detection in live stream video [9]. Due to its good accuracy and fast rate of image recognition, we deployed this algorithm to detect the potholes [9]. It is faster than other detection systems across a variety of detection datasets. It is famous because it achieves good accuracy while also being able to work in real-time and provides a frame rate of 45 fps [12]. It has a smooth trade-off between speed and accuracy [13]. The algorithm "only looks once" at the image in the sense that it requires only one forward propagation pass through the neural networks to make predictions [14].

3.3 Methodology

The proposed methodology consists of a smartphone application useful for citizens and civic authority to detect the road surface condition. It is a simple user-friendly app that enables its user to tap pictures of potholes and upload them. This application is used to detect the potholes as well as view the locations of the potholes. Citizens register on the app using their email and username–password. Access permissions, like the location of the user (i.e., GPS), camera permissions are required. User logs into his account and taps the photos of the potholes in their area. When one submits the image, our application processes the request and sends it to our server. Within a few seconds, he is notified, the number of potholes detected in the image. If potholes have been detected, then it is stored in our database and displayed on the map for everyone to view. At the backend, to detect the potholes in the image, neural networks are being used. Specifically, we are using the YOLO algorithm to detect the potholes in the image. Civic authority registers on the app and he logs in using his credentials. Civic authority can view the potholes in his locality that have been reported by the users on the map. Accordingly, he repairs the potholes and updates the status of the work in our app. In this way, our app effectively works to detect potholes and easily repairs them.

For more effective classification, the potholes are categorized and given more priority according to their severity (or location), i.e., national highway, state highway, city rods, local roads, and number of potholes detected in one image or area. More the user base, more effective our app works. To increase our user base, we are providing credit points to citizens for each upload. Likewise, the civic authority will be rated and rewarded according to his work completed in a stipulated time. Our methodology consists of an android app, which is easy to use smart way to detect the potholes. This method has high speed and good accuracy in detecting potholes in the image. It also can efficiently detect multiple potholes in a single image.

3.4 Workflow of the App

1. Registration: Users can register on the app as a citizen or civic authority with their email and password.
2. Login: Users can log into the app by the credentials provided to him.
3. Access to permission: User then has to give access to permissions of GPS and camera.
4. Capture Image: User then has to open the camera, click a picture of the pothole, and submit it.
5. Processing and result: After submitting the image, within seconds the image is processed, and then, the user is notified the number of potholes detected.
6. Google Map: Users can access the map, integrated into our app, and view the markings/locations of the pothole in the area.
7. Logout.

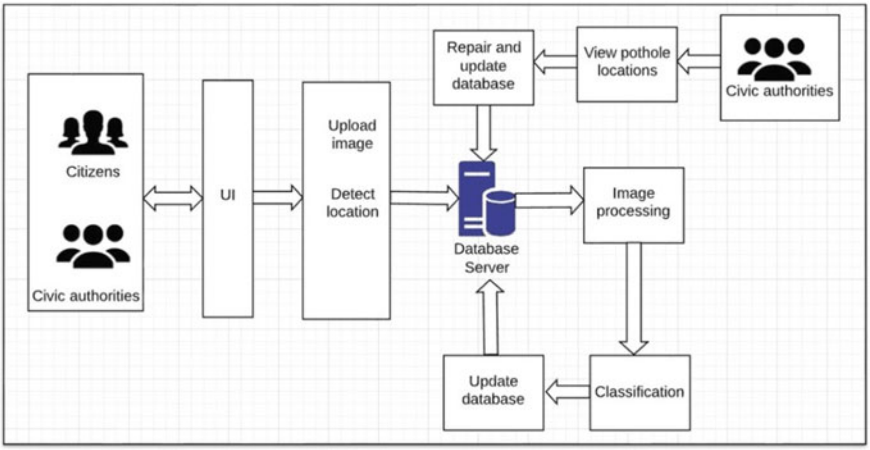

Fig. 1 System architecture

3.5 Architecture

The proposed system architecture consists of an android application, image processing model, and server to process in request and store data. We are trying to automate the system by providing real-time pothole locations (Fig. 1).

3.6 Pothole Detection Using YOLO Algorithm

Initially, we take a pre-processed pothole picture, and YOLO [9] is applied. The picture is isolated as frameworks of grids. We then isolate the picture into any number matrices, contingent upon the picture. Each grid undergoes classification and localization. The objectness or the certainty score of every matrix is found. In the event that there is no any pothole found in the framework, at that point the objectness and bounding box estimation of the network will be zero, or in the event that we have found a pothole in the lattice, at that point the objectness will be one, and the bounding box value will be its bounding value of the discovered object. To comprehend the YOLO calculation, it is important to build up what we need to foresee [14]. At last, we aim to predict a class of an object and the bounding box indicating object area. Each bounding box has four descriptors: Center of a bounding box (**bx**, **by**), width (**bw**), height (**bh**), value **pc** is corresponding to a class of an object (as: pothole or no pothole) [15].

Also, we need to anticipate the pc value, which is the likelihood that there is an object in the bounding box. As mentioned above, when working with the YOLO algorithm we are not searching for interesting regions in our image that could potentially contain an object. Rather, we are parting our picture into cells, utilizing a 19 ×

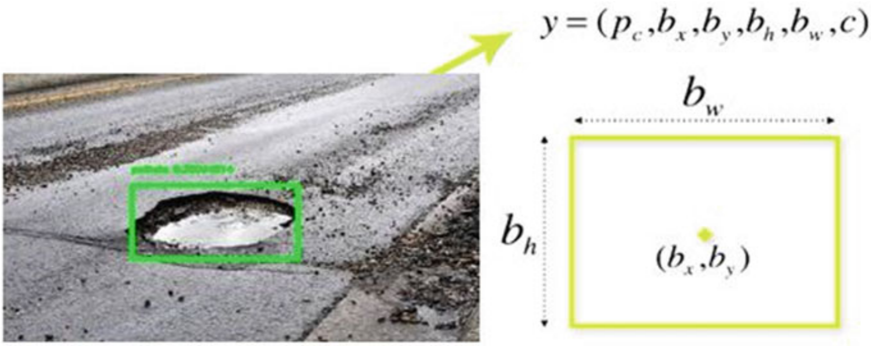

Fig. 2 Bounding box description

19 lattice. Every cell is liable for foreseeing five bounding boxes (in the event that there is more than one object in this cell). In this manner, we show up at an enormous number of 1805 bounding boxes for one picture. A large portion of these cells and bounding boxes won't contain an object. Most of these cells and bounding boxes will not contain an object. Therefore, we predict the value pc, which serves to remove boxes with low pothole probability and bounding boxes with the highest shared area in a process called non-max suppression [14] (Fig. 2).

4 Experimentation and Results

4.1 Pre-processing

Pothole images were collected from the kaggle pothole dataset [16]. These images are used for training and testing of our YOLO model. One thousand five hundred images were used with five-fold cross-validation. Images were pre-processed before training. Following are processing methods being used like checking annotations and labels of the data, checking bounding boxes, scale images to 80–120%, converting images to superpixel representation, horizontal and vertical flip if required, adding Gaussian blur to images, detect the edges, scale, and resize images if required.

4.2 Experimentation

For carrying this experiment, we have used pothole kaggle dataset [16]. About 1500 images are used for training our YOLO model. The model is trained with five-fold cross-validations. Table 1 shows the confusion matrix for the training and testing of 1500 images. By applying five-fold cross-validation, the accuracy of the proposed

Table 1 Confusion matrix of five-fold cross-validation

	Fold 1		Fold 2		Fold 3		Fold 4		Fold 5	
Class	Class 1 predicted	Class 2 predicted	Class 1 predicted	Class 2 predicted	Class 1 predicted	Class 2 predicted	Class 1 predicted	Class 2 predicted	Class 1 predicted	Class 2 predicted
Class 1 actual	160 (TP)	40 (FN)	164 (TP)	36 (FN)	165 (TP)	35 (FN)	172 (TP)	28 (FN)	176 (TP)	24 (FN)
Class 2 actual	40 (FP)	60 (TN)	35 (FP)	65 (TN)	33 (FP)	67 (TN)	26 (FP)	74 (TN)	19 (FP)	81 (TN)
Accuracy (%)	69.1		74.3		75.66		79.21		85.45	
Overall accuracy (%)					76.4					

Here,
Class 1: Pothole. *Class 2*: No Pothole
True Positive (*TP*): Observation is that there is a pothole and is predicted as a pothole
False Negative (*FN*): Observation is that there is a pothole but is predicted as not a pothole
True Negative (*TN*): Observation is that there is no pothole and is predicted as not a pothole
False Positive (*FP*): Observation is that there is no pothole, but is predicted as a pothole

Table 2 Number of images which detected the exact number of potholes

	Images detected with the accurate number of potholes	Images detected with inaccurate number of potholes than actual	Accuracy (%)
Fold 1	130	30	81.2
Fold 2	135	29	82.3
Fold 3	138	27	83.6
Fold 4	148	24	86.1
Fold 5	150	26	85.2
Total	701	136	83.6 (Average)

method comes out to be 76.4%. With an increase in training data, the model can improve its accuracy. Further, the proposed method also identifies the number of potholes in an image. It detects the exact number of potholes in an image with the accuracy of 83.6% (Table 2). Thus, along with good accuracy to detect potholes, it also shows better accuracy in detecting multiple potholes in an image. More number of potholes in an image describes the worse road condition.

The input image size for our system is set to 416 × 416 pixels with three channels of color as red, blue, and green. Our model is trained with object threshold of 0.5, and non-max suppression is set to 0.45. Maximum boxes per image are set to 18. After exceeding 18, the pothole is set on the high priority list. Confidence score is set above 80 to accurately detect the potholes in the image. The output of our model is presented with four parameters. (1) Location: Multidimensional array representing bounding boxes (2) Class: Indicating the class (Pothole or No pothole). (3) Score: Representing the probability that a class was detected. (4) Number of detections: Array of length 1 containing a value expressing the total number of potholes detected in an image. Refer (Fig. 3).

Various algorithms were studied and compared for object detection in the image. It is found that the YOLO algorithm has quicker handling time than any other algorithm.

Fig. 3 Multiple potholes detected in an image

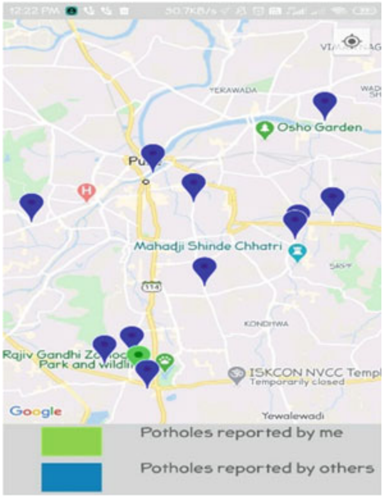

Fig. 4 Potholes location on the map

It could also be used for real-time object detection [13]. It is accurate in detecting the multiple potholes in the single images which is poor in the case of other algorithms. Our model gives an accuracy of 76% [14]. Hence, due to multiple pothole detection and low processing time, this is effective for real-time pothole detection. Utilizing results from gadgets running the application, we have delivered rich maps of the city with potholes and street conditions, as appeared in Fig. 4.

5 Conclusion and Future Scope

We studied different approaches for pothole detection. We have selected the best suitable YOLO algorithm for pothole detection. It detects the presence of potholes in an image with around 76% accuracy; in these correctly detected images, it counts the exact number of potholes with the accuracy of 83.6%. This app helps to detect fake images by making use of machine learning. Since the users directly feed the pothole images to the app, it reduces the burden of redundant storage and pre-processing. This mobile app also categorizes the road conditions depending on the presence of number of potholes. It helps the civic authorities to take remedial actions on priority basis.

This project can be further extended for other activities like comparing different road conditions and selecting the shortest route to the destination with the better road conditions. Further, we can add the feature to produce color-coded maps based on the severity of the road surface conditions. The efficiency of the model for recognizing potholes can be further enhanced by applying backward learning to make the system smarter with time. In future, this work can be extended to detect road conditions in all other cities.

References

1. Kumar, A., & Kumar, S. (2018). Road quality management using mobile sensing. In *International Conference on Innovations in information Embedded and Communication Systems*.
2. Mendis, A., Starzadnis, G., & Anoris, G. K. (2019). Real time pothole detection using android smartphones and accelerometers. In *2019 12th International Conference on ITS Telecommunications (ITST)* (pp. 668–672). IEEE.
3. Mohan, P., Padmanabhan, V. N., & Ramjee R. (2008). Nericell: Rich monitoring of road and traffic conditions using mobile smartphones. In *Proceedings of 6th ACM Conference on Embedded Networked Sensor Systems* (pp. 323–336).
4. Yagi, K. (2016, September). A measuring method of road surface with road profiler. In *Proceedings of IEEE Transaction on Intelligent Transportation System* (Vol. 13).
5. Piao, B., & Aihara, K. (2019). Detecting the road surface condition by using mobile crowd sensing with drive recorder. In *IEEE 20th International Conference on Intelligent Transportation Systems.*
6. Kyriakou, C., Christodoulou, S. E., & Dimitriou, L. (2016). Roadway pavement anomaly classification utilizing smartphones and artificial intelligence. *Journal of the Transportation Research Board*.
7. https://www.theguardian.com/world/2018/jul/25/more-deadly-than-terrorism-potholes-responsible-for-killing-10-people-a-day-in-india.
8. https://www.indiatoday.in/india/story/over-9300-deaths-25000-injured-in-3years-due-to-potholes-1294147-2018-07-24.
9. Redmon, J., & Farhadi, A. (2020). YOLO9000: Better, faster, stronger. In *IEEE Conference on Computer Vision and Pattern Recognition.*
10. Lee, S. W., & An, K. E. (2018). Image processing based pothole detection system for driving environment. In *IEEE International Conference on Consumer Electronics (ICCE).*
11. Zhang, Z., Ai, X., & Chan, C. K. (2018). An efficient algorithm for pothole detection using stereo vision. In *IEEE International Conference on Acoustic, Speech and Signal Processing.*
12. Comparison: https://medium.com/@jonathan_hui/object-detection-speed-andaccuracy-comparison-faster-r-cnn-r-fcn-ssd-and-yolo-5425656ae359.
13. Divvala, S., Girshick, R., & Farhadi, A. (2017). You only look once: Unified, real-time object detection. In *Computer Vision-ECCV 2014 Workshops* (pp. 101–116). Springer.
14. Geethapriya, S., Duraimurugan, N., & Chokkalingam, S. P. (2019, February). Real-time object detection with Yolo. *International Journal of Engineering and Advanced Technology (IJEAT), 8*(3S). ISSN: 2249 – 8958.
15. YOLO: Real-Time Object Detection. https://pjreddie.com/darknet/yolov2/.
16. Kumar, A. Dataset for potholes. https://www.Kaggle.com/atulyakumar98/pothole-detection-dataset.