EAI Endorsed Transactions
on Internet of Things

Research Article EAI.EU

Design of Intelligent Road Eye Using AI and Machine Learning for Automobiles

Chandrasekharan Nataraj[1,*], Lawrence Wong Ming Wei[2], Mukil Alagirisamy[3], and Sathish Kumar Selvaperumal[4]

[1, 2, 3, 4] School of Engineering, Asia Pacific University of Technology& Innovation (APU), Technology Park Malaysia 57000, Kuala Lumpur, Malaysia

Abstract

The project aims to design an intelligent road eye by using AI and a machine learning approach to detect speedbumps and potholes on the road. The system design utilizes a YOLOv5 custom-trained model and COCO dataset in detecting the objects on the road. The system is integrated with lane detection algorithms to achieve active steering feedback and pothole avoidance. Based on the detection results, feedback will be given in the form of visual, audio, and steering angles, allowing the driver to have sufficient response time to perform braking or steering adjustments where applicable. The trained model can achieve a mean average precision value (mAP) of up to 0.995 for all classes, and a maximum detection range of 5.77m and 34.8m for potholes and speedbump respectively. The future works of the project include integrating the algorithm into the vehicle to achieve autonomous braking and active pothole avoidance with the help of sensors and cameras on the vehicle, as well as adopting augmented reality (AR) to project the visual feedback on the vehicle windscreen.

Keywords: Artificial Intelligence, Image processing, object detection, Machine Learning, Road safety, Vehicle safety.

Received on 23 10 2024, accepted on 19 12 2024, published on 17 01 2025

doi: 10.4108/eetiot.7638

1. Introduction

According to data provided by the Malaysian Ministry of Transport, 3700 fatal cases on average occur each day, and there are over 1.35 million road accident fatalities annually. There are about 18 fatal road accidents each day in Malaysia, and among other road users, passenger cars are responsible for 21% of the fatalities [1-2].

Serious accidents can be caused by several things, including irresponsible driving, distracted drivers, weather-related low visibility, and poor road infrastructure. Some often unforeseen events are unavoidable, such as pedestrians crossing the street without noticing approaching cars or unexpected barriers to which the driver is unable to react [3,4,5].

Due to the issues that arise, car manufacturers have been developing different technologies throughout the years such as lane departure monitoring to alert the driver that the car is not staying on its respective lane, blind spot monitoring to detect vehicles at the driver's blind spots, forward collision system and more to assist the driver on the road and to prevent the unwanted accidents from happening [6-7].

Car manufacturers have developed a system called autonomous emergency braking (AEB) to further enhance the existing safety technology [8]. AEB was first introduced by the Swedish car maker Volvo in the year 2009 [9]. It uses a camera-based system and various kinds of radar to determine whether the vehicle is on a collision course and activate the braking system of the vehicle automatically when the front collision warning is given out to the driver, but no braking response is applied to the vehicle. Typically, the AEB system can prevent vehicles from getting rear-ended on the road as

[*]Corresponding author. Email: chandrasekharan@apu.edu.my

around 80% of accidents are caused by the rear-enders [10-11]. The AEB system will usually work with the anti-braking lock system (ABS) to achieve the best braking performance [12].

Furthermore, the poor maintenance of road infrastructure in Malaysia has been causing the existence of potholes and speedbumps [13]. Most of the time it is difficult for the driver to avoid potholes that are hardly visible or unmarked speedbumps on time, and it has become a major issue for road users due to the damage it may cause to the vehicle chassis, wheels, and suspension system under long run, which consequently increases the risk of road accidents and the cost of vehicle maintenance[14-16]. Therefore, the project aims to design an intelligent road eye by using AI and a machine learning approach to detect various uncertain incidences on the road.

2. Literature Survey

The proposed system in the International Journal of Electrical and Computer Engineering (IJECE), titled “Real-time traffic sign detection and recognition using Raspberry Pi” [3] can detect and recognize different traffic signs to promote safety of the road users. The system uses TensorFlow to perform object recognition with the aid of Raspberry Pi 3 as the processing unit and Raspberry Pi camera for obtaining input. Twenty types of traffic signs are considered during the test and the system has achieved an accuracy of up to 90% with a slight delay which is acceptable.

The research method started with dataset labelling using the tool called Labelling. The labelling process involves segmentation of images, followed by image annotation and interpretation. Image annotation is to apply bounding boxes onto the road signs available within the images to allow them to be recognized. The labelled images will be exported in XML file format for the use of model training. The dataset training is done by using TensorFlow, which has covered five types of traffic signs, including Speedbump, No U-Turn, Give Way, Stop, as well as Chevron Alignment. A total of 100 samples with different angles and sizes are taken for each category. TensorFlow Lite V2.5.0 is installed on the Raspberry Pi 3 for running the pre-trained model [3].

According to the research by Asad et al, titled “Pothole Detection Using Deep Learning: A Real-Time and AI-on-the-Edge Perspective” [4], the comparison of different deep learning models for pothole detection, including various versions of YOLO and SSD algorithms has been made. The test is conducted on a moving vehicle under different lighting and road conditions.

The hardware used is mainly an OAK-D camera and a Raspberry Pi module. In the data acquisition phase, 665 images which consist of up to 8000 potholes under various environments are collected. YOLO and SSD algorithm families are used as the deep learning models for training. Darknet framework is used for the training of YOLOv1 to YOLOv4, PyTorch framework is used for training YOLOv5, whereas Tensorflow framework is used for the training of SSD-Mobilenetv2. Dataset annotation is then done according to the respective model type before the training process [4].

The results show that the Tiny-YOLOv4, YOLO v4 and YOLO v5 have achieved a mean average precision of 80.04%, 85.48% and 95% respectively with Tiny YOLO v4 selected as the ideal model for pothole detection due to its 90% detection accuracy and a smooth 31.76fps frame rate. The future improvements being considered include road depression detection, road quality classification and pothole depth estimation [4].

The system proposed by Dewangan & Sahu in the paper “Deep Learning Based Speed Bump Detection Model for Intelligent Vehicle System using Raspberry Pi” [5] utilized deep learning and computer vision approach to improve speed bump detection such as enhanced accuracy and provide warning signal earlier. The system is proposed using a Raspberry Pi-based vehicle prototype, with both marked and unmarked speed bumps being considered. A total of 575 images of the speed bump were taken using the Raspberry camera module and underwent image processing to obtain a clearer image. Data augmentation is applied to increase the dataset size to 3450 samples. CNN is used as the architecture for detection with filter sizes of 5*5, 3*3 and 2*2. A batch size and learning rate of 32 and 0.001 respectively are used during the training process to avoid error.

Once the speedbump is detected within the bounding box, the distance between the camera and the bump is identified by utilizing the pixels between the left and right corners of the bounding box. By obtaining the difference between two points and substituting it into the linear equation, the gradient and intercepting point can be used for determining the actual distance between the bump and the vehicle. The proposed system can achieve an accuracy and precision level of 98.54% and 99.05% respectively. Future improvements being considered involved object detection other than speed bumps [5].

The system proposed by Anand in the paper “Intelligent Vehicle Speed Controlling and Pothole Detection System” [6] used a machine-learning approach to introduce dynamic speed limiters on vehicles to reduce accidents on the road. The system detects speed limit road signs along the way and caps the maximum speed of the vehicle accordingly. The system is also able to detect potholes by utilizing the accelerometer vibration on the phone.

The detection process can be split into three phases, which are image pre-processing, sign board detection and speed limit recognition. After the input images have undergone pre-processing for a clearer image, region detection of the speed limit sign is done using maximally stable extremal regions (MSER) detection on the pre-processed grayscale image.

A Histogram of Oriented Gradient (HOG) features from the road sign is then extracted and fed to a Support Vector Machine (SVM) for speed limit classification. Once the speed limit has been obtained. The maximum vehicle speed will be set using Raspberry Pi [6]. Whereas for pothole detection, the input will be based on the vibrations sensed by the phone accelerometer. Once the vibrations are detected, the location coordinate will be plotted on the map for respective authorities to manage using the GPS and GSM service, alternative path will be suggested to the user. The advantage of using the MSER detection method is that it's relatively robust and stable under different environmental conditions [6].

3. Proposed Methodology

3.1. Block Diagram

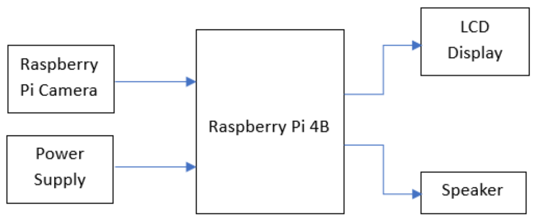

Figure 1. Block Diagram

Figure 1 shows the block diagram of the dashboard system. The Raspberry Pi 4B module is powered by an external power supply to function. Raspberry Pi camera is used as the system input to capture the images on the road. The images captured will then be processed by the Raspberry Pi 4B module through the image processing techniques and object detection algorithms to mainly detect lane marking, speedbumps and potholes on the road. Based on the detection results, feedback in the form of visual, audio, and steering angle will be given through the LCD screen, and speakers in the vehicle cabin.

3.2. System Implementation and Data Collection

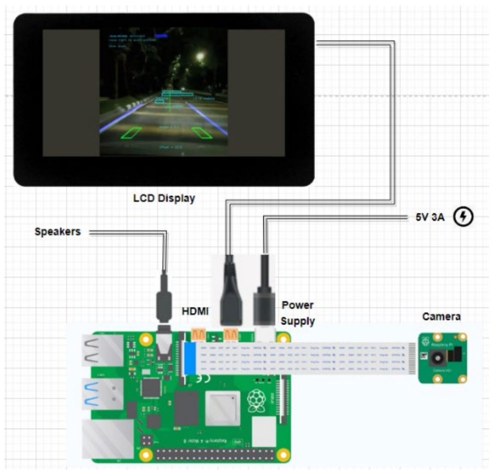

Figure 2. Dashboard System

Figure 2 shows the dashboard system for vehicle safety. The Raspberry Pi and Pi camera modules are placed on top of the vehicle dashboard to capture images of potholes and speedbumps on the road surface. RPi-Cam-Web-Interface is utilized to record the driving footage along the drive, as Raspberry Pi was not equipped with a video recording feature originally. The video resolution set for the recording is 640 x 640 to adapt to the properties of the YOLOv5 custom-trained model.

The extracted frames are then annotated using Roboflow. There's a total of 455 images that have been collected. The images are then imported into the Roboflow platform for the annotating process. During the process, bounding boxes are drawn and labelled on the potholes and speedbumps within each frame. The annotated images then underwent an augmentation process to further increase the size of the dataset. The augmentations process has produced an additional 499 images with altered saturation, brightness, and exposure by a scale of 25% from the original images, resulting in a total of 954 images for the training set.

3.3. Training of datasets

The training of datasets is done on the Google Colab platform. Due to the large number of images, the annotated datasets are imported into Google Drive before the training process. The Google Drive directory as well as the libraries required such as PyTorch and Torchvision are then imported into Google Colab. The number of training cycles (epochs), batch size, directory to the dataset YAML file and the training size of the model is then defined accordingly in the YOLOv5 training command.

For lane detection, input images will be resized to the resolution of 640 x 640. The resized frames will undergo canny edge detection for the detection of the edges available in the frame. Since the only interest is to detect the lane marking on the road, a region of interest (ROI) is set with the aid of coordinates, hence, canny edge detection will only be applied on the road surface region within the frame, which can improve the overall efficiency of the detection as the unwanted parts have been filtered out. Next, the Hough transform is applied to the cropped frames with Canny edge detection to obtain the ideal or best fit for the straight line within the frame. Object detection algorithms are then applied to detect potholes and speedbumps on the road using the custom trained YOLOv5 model. The respective visual and audio feedback for the detections is defined accordingly.

3.4. Python Programming (Algorithm) and Flowchart

For lane detection, input images will be resized to the resolution of 640 x 640. The resized frames will undergo canny edge detection for the detection of the edges available in the frame. Since that the only interest is to detect the lane marking on the road, a region of interest (ROI) is set with the aid of coordinates, hence, canny edge detection will only be applied on the road surface region within the frame, which can improve the overall efficiency of the detection as the unwanted parts have been filtered out. Next, Hough transform is applied onto the cropped frames with Canny edge detection applied to obtain the ideal or best fit for the straight line within the frame. Object detection algorithms are then applied to detect potholes and speedbumps on the road using the custom trained YOLOv5 model. The respective visual and audio feedback for the detections is defined accordingly.

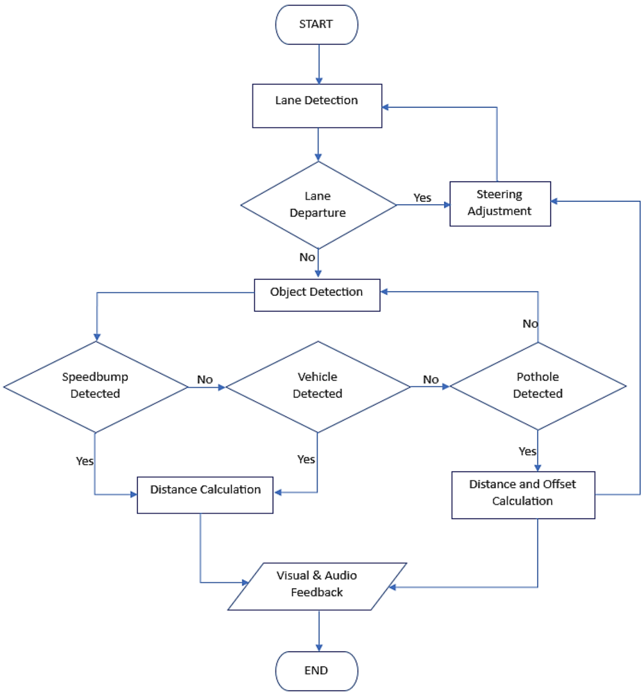

Figure 3. Methodology Flowchart

Figure 3 above shows the flowchart of the intelligent road eye. The working process started with the lane detection algorithm. The algorithm will constantly detect the lane marking on the road to ensure that the vehicle is staying in the middle of the lane. Once lane departure is detected, the steering adjustment required will be calculated through trigonometry with the aid of the returning values from the lane detection algorithm. When the vehicle is travelling on the lane, object detection will be executed to detect the presence of speedbumps, vehicles, and potholes. When a speedbump or vehicle is detected, the distance between the object detected and the vehicle will be calculated, whereas when a pothole is detected, the object distance and the offset of the pothole from the middle line will be calculated, and the parameters obtained will be considered in the steering adjustment calculation. The feedback will then be given in the form of visual and audio through the LCD screen and speaker respectively.

4. Results and Discussion

Figure 4 shows the hardware setup of the intelligent road eye. The system will constantly capture the road situation ahead of the vehicle and provide feedback in the form of visual and audio to the driver. The integrated system is attached on the dashboard of the car, which is powered by the car battery. The system consists of high sensitivity camera, microcontroller and display system. The camera is very sensitive to the images captured in both day and night. The images are

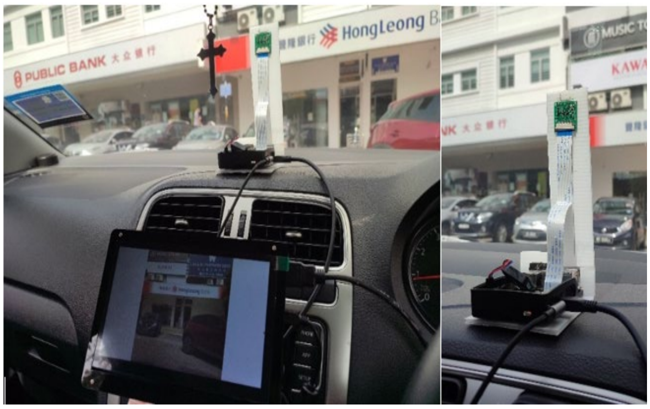

Figure 4. Dashboard Hardware Setup

processed by using microcontroller and provides the relevant steering angle signals to the IOT platform. Then, the IOT will activate the actuators connected with the car braking system. To evaluate the performance of the developed hardware, significant analysis is conducted through the following tests.

4.1. Model Performance Test

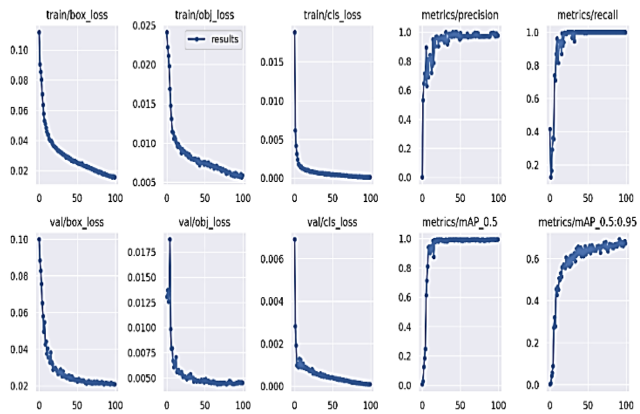

Figure 5. Custom model training results

Figure 5 shows the results of custom model training. The epoch set used for the training is 100. The entire training process took approximately 1,048 hours to complete. It can be observed that the precision and recall value of the trained model have achieved a steady state after surpassing the 50th training cycle. The mean Average Precision value (mAP) at the Intersection over Union (IoU) with a threshold of 0.5 (mAP_0.5) and a threshold range between 0.5 to 0.95 (mAP_0.5:0.95) have been plotted. For the threshold of 0.5, the mAP_0.5 value reached its peak at 0.995 for all classes, including pothole and speedbump.

As for the comparison with the existing works, the system proposed by Asad et al [4] has achieved a mean average precision of 0.95 for YOLOv5 model, the proposed system in the existing work can achieve an accuracy of up to 90% with a slight delay, indicating that the training results is considered decent and reasonable. On the other hand, the mAP-0.5:0.95 reached a peak of 0.692, 0.631 and 0.752 for the class of all, pothole, and speedbump respectively. The trained model seems to have the perfect skill as it is depicted as a point that is close to the coordinate (1,1) in the precision-recall curve as shown in Figure 6.

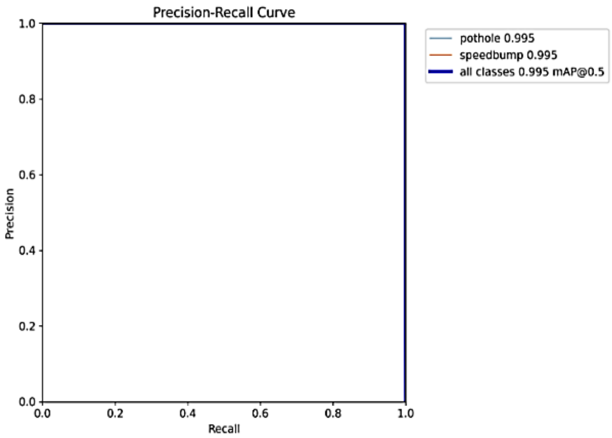

Figure 6. Precision-recall curve

4.2. Navigation test for pothole detection in daytime

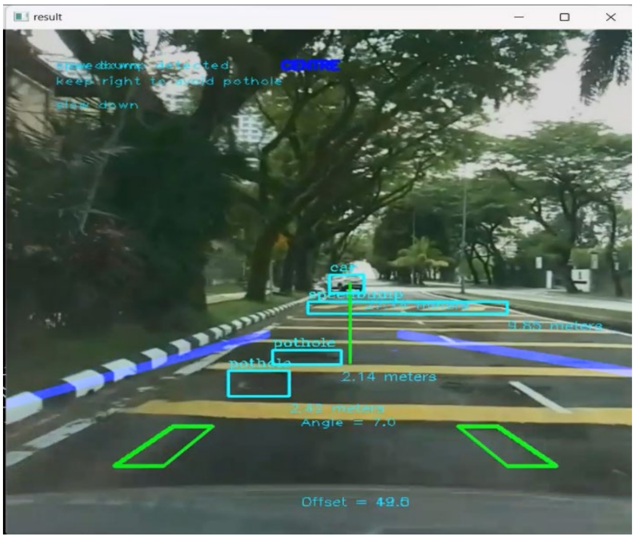

Figure 7. Navigation test for potholes (daytime)

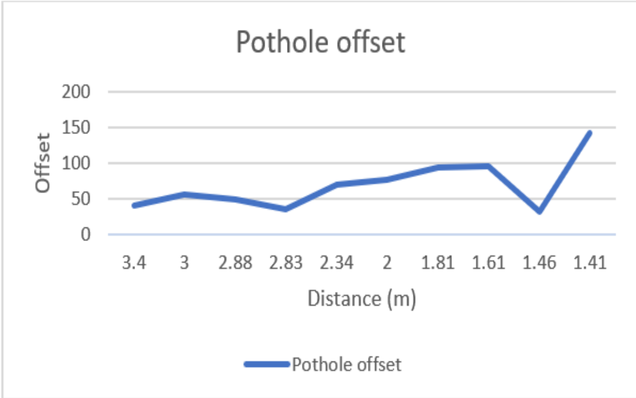

Figure 8. Pothole offset against distance

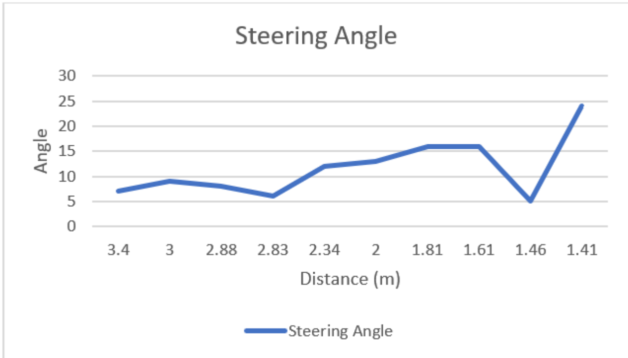

Figure 9. Steering angle generated against distance

Figures 7, 8 and 9 show that the pothole can be detected from a maximum distance of 3.4 meters. The short detection range could be due to the damp road surface which affects the detection accuracy. As the vehicle approaches the potholes, the pothole offset from the middle line set is increased accordingly due to the increase in object size within the frame and the pixels detected within the offset. The concept is utilized for generating the steering angle needed to avoid potholes.

The steering angle graph possessed a similar characteristic as the graph of pothole offset against distance. Theoretically, the steering angle is supposed to be increased as the distance decreases, since the shorter the distance, the nearer the vehicle is to the pothole, hence larger angle is needed to avoid the pothole under moving condition. However, the graph shows a slightly fluctuating result due to the presence of yellow stripes on the road that interrupting the lane detection algorithm.

4.3. Navigation test for pothole detection at night

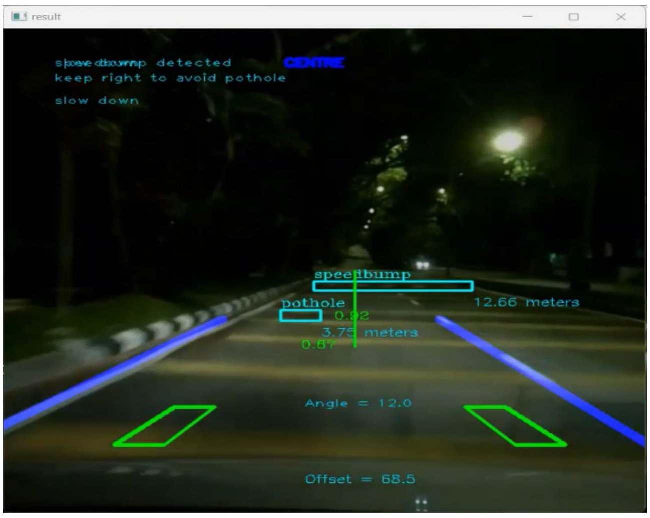

Figure 10. Navigation test for potholes (night)

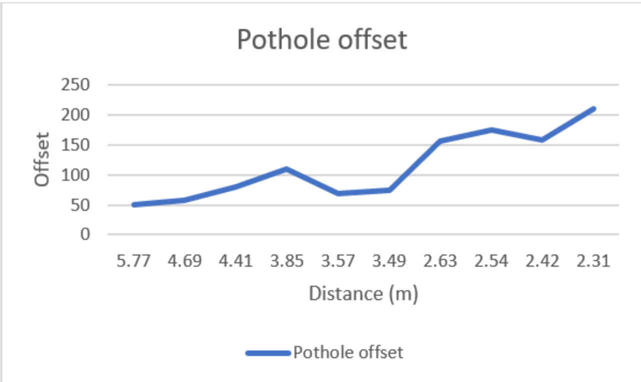

Figure 11. Pothole offset against distance

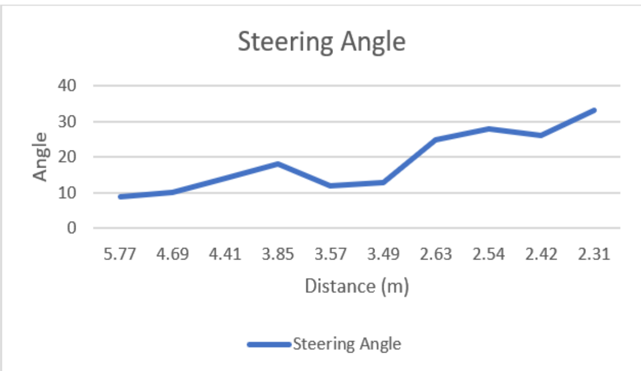

Figure 12. Steering angle generated against distance

Figures 10, 11 and 12 show that the pothole can be detected from a maximum distance of 5.77 meters at night, which is significantly further than the detection range during daytime. This could be due to less environmental disruptions as it is dark at night. Moreover, the vehicle headlight which points towards the road ahead allows the pothole to be detected more easily.

4.4. Confidence test for speedbumps and potholes detection (daytime)

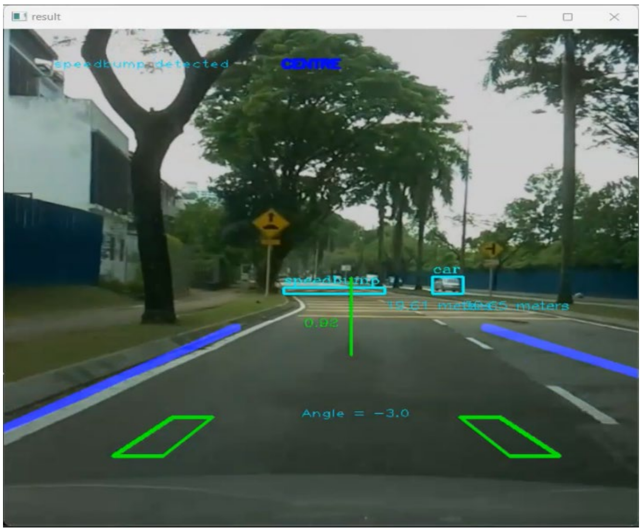

Figure 13. Confidence test (daytime)

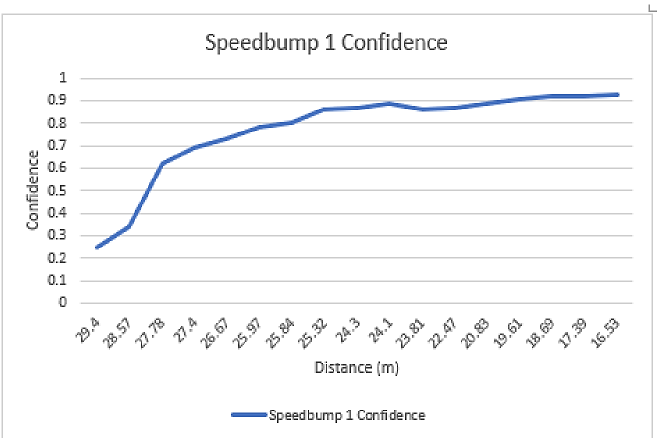

Figure 14. Confidence score of speedbump 1 (daytime)

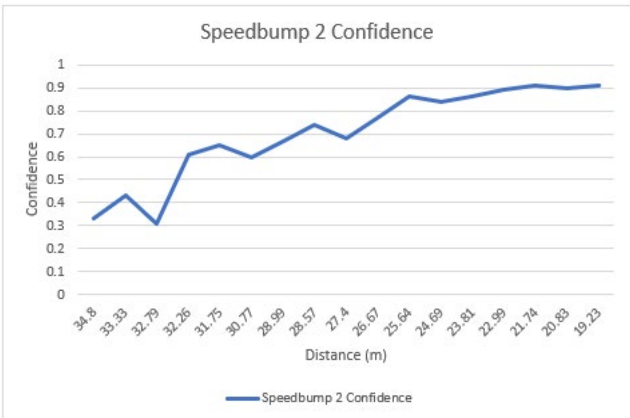

Figure 15. Confidence score of speedbump 2 (daytime)

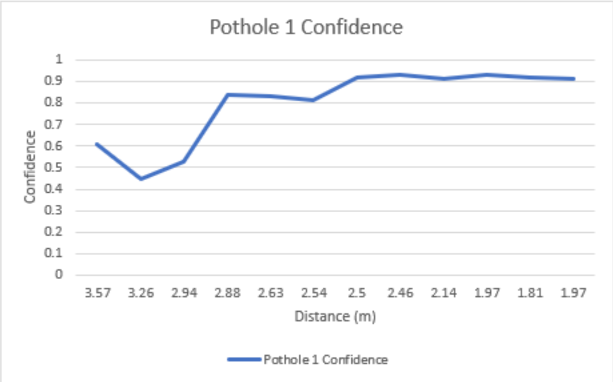

Figure 16. Confidence score of pothole 1 (daytime)

Figures 13, 14, and 15 show that the confidence score increased as the distance decreased. During the daytime, speedbump 1 can be detected from as far as 29.4 meters away, the confidence score increased drastically from 0.25 at 29.4 m to 0.62 at 27.72m, and gradually achieved its peak of 0.93 at 16.53m and maintained the confidence score at above 0.9 afterwards. On the other hand, speedbump 2 can be detected from as far as 34.8 meters away, which is relatively further compared to speedbump 1.

This could be due to the reason that Speedbump 1 is located within a corner, whereas Speedbump 2 is located on a straight road. The confidence score increased gradually from 0.33 at 34.8m to the peak of 0.91 at 19.23m and maintained the confidence score above 0.9 afterwards. Fluctuation can be seen throughout the graph, which could be due to the presence of vehicles on the road. However, the graph is still showing an overall increment characteristic of the confidence score throughout the dataset.

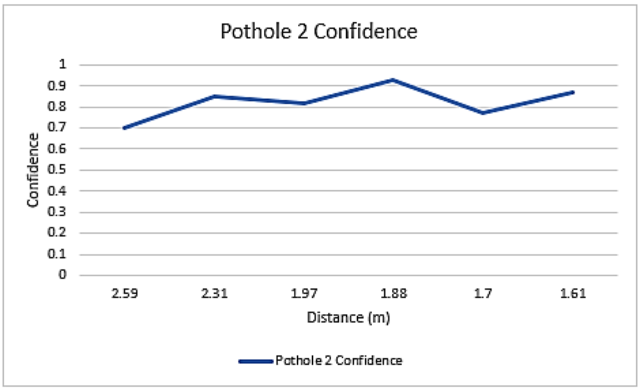

Figure 17. Confidence score of pothole 2 (daytime)

During daytime, pothole 1 can be detected from as far as 3.57 m away. Figure 16 shows that the confidence score dropped from 0.61 at 3.57 m to 0.45 at 3.26 m, and then increased again drastically to 0.84 at 2.88 m and gradually towards it peak of 0.93 at 2.46 m. Figure 17 shows little increment in confidence score for pothole 2, since that it can only be detected from 2.59 m away during daytime The fluctuations in between the data could be due to the presence of yellow stripes or the damp road surface conditions that affect the detection.

4.5. Confidence test for speedbumps and potholes detection (night-time)

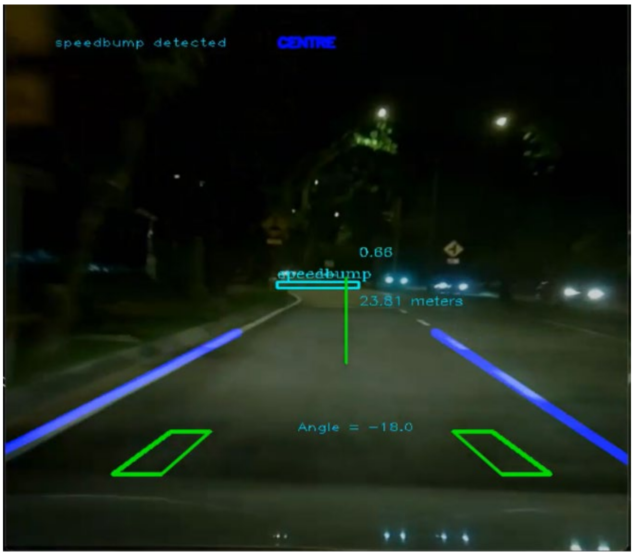

Figure 18. Confidence test (night)

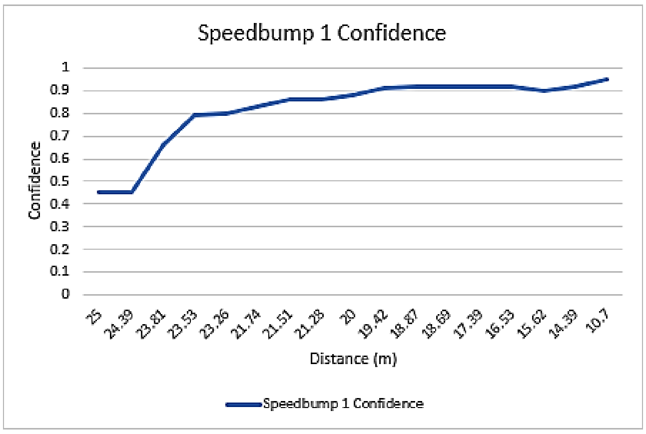

Figure 19. Confidence score of speedbump 1 (night)

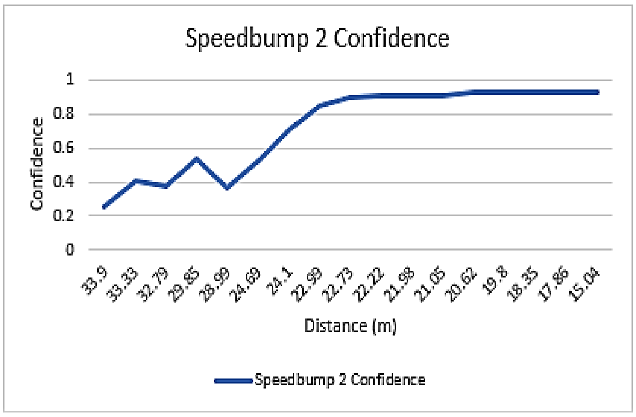

Figure 20. Confidence score of speedbump 2 (night)

Figures 18, 19, and 20 show the results of the confidence test conducted at night-time. During night-time, speed bump 1 can be detected from as far as 25 meters away, slightly shorter than 29.4 meters during daytime. Figure 19 shows that the confidence score increased drastically from 0.45 at 24.39 m to 0.79 at 23.53m, and gradually achieved its peak of 0.95 at 10.7m and maintained the confidence score at above 0.9 afterwards.

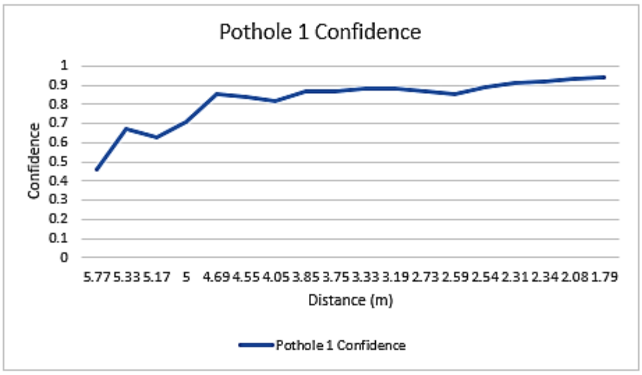

Figure 21. Confidence score of pothole 1 (night)

On the other hand, speedbump 2 can be detected from as far as 33.9 meters away, which is slightly shorter than 34.8 meters during daytime. This could be due to a lack of light source that caused relatively poor visibility at night. Figure 20 shows that the confidence score increased gradually from 0.26 at 33.9m to the peak of 0.93 at 20.62m. Fluctuation can be seen at the beginning of the graph, which could be due to the presence of vehicles on the road. However, the graph is still showing an increment characteristic in the confidence score across the dataset.

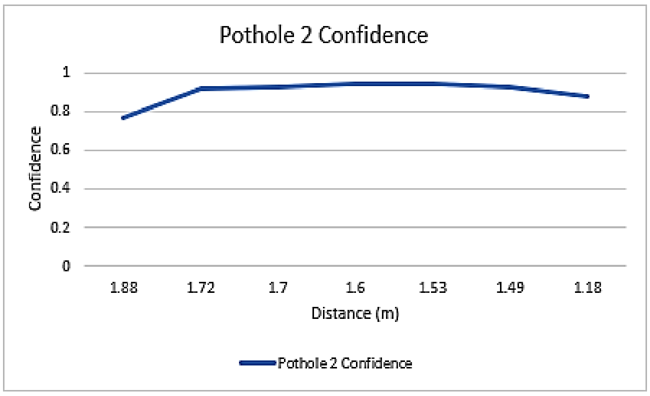

Figure 22. Confidence score of pothole 2 (night)

During night-time, pothole 1 can be detected from as far as 5.77 m away, which is significantly higher than the detected distance during the daytime.

At the beginning of the graph in Figure 21, the confidence score increased from 0.46 at 5.57 m to 0.67 at 5.33 m and dropped to 0.63 at 5.17 m. The score is then gradually increased towards the 0.94 peak at 1.79 m. The fluctuations in between the data could be due to the detection range, the presence of yellow stripes or the damp road surface conditions that affect the detection.

Figure 22 shows that pothole 2 can only be detected from 1.88 m away during night-time, which is relatively lower than the 2.59m range during daytime. The confidence score increased from 0.77 at 1.88m to the 0.94 peak at 1.60 m and dropped slightly to 0.88 at 1.18m. The dropping of the confidence score at the end of the data could be due to the close detection range and partially covered potholes, which are not included in the training dataset.

5. Conclusion

The intelligent road eye is designed using AI and machine learning approaches to detect the existence of speedbumps and potholes on the road. Based on the detection results, feedback in the form of audio, visual and steering angle will be given to the driver for performing braking or steering adjustments where applicable. In this work, an AI algorithm was developed for detecting potholes and speedbumps using the YOLOv5 model on the Google Colab platform. In addition, a Raspberry Pi-based system was constructed for executing relevant feedback responses such as audible, and visible alert signals, and steering angle calculation for pothole avoidance with the aid of Python programming.

Further, the prediction accuracy of the developed road eye was analysed through a model performance test, navigation test and confidence test in both day and night conditions. The trained model could produce a mean average precision value (mAP) of up to 0.995 for all classes and a maximum detection range of 5.77m and 34.8m for potholes and speedbumps respectively. The developed dashboard with an object detection algorithm will enable us to reduce the risks of road accidents and create a new market opportunity.

References

[1] Ministry of Transport Malaysia, https: //www.mot.gov.my /en/land/safety/malaysia-road fatalities index.2019.

[2] S. Bhandari, A. Mhaske, S. Bhandari, A. Memon and B. Dixit. Road Conditions and Obstacles Indication and Autonomous Braking System. 3rd International Conference on Intelligent Sustainable Systems (ICISS), India. 2020; 119-124.

[3] A, P. Y., Salim, A., Dileep, A., & S, A. Autonomous Car using Raspberry PI and Ml. International Journal of Recent Technology and Engineering. (2020); 9(2): 1067–1071.

[4] Asad, M. H., Khaliq, S., Yousaf, M. H., Ullah, M. O., & Ahmad, A. Pothole Detection Using Deep Learning: A Real-Time and AI-on-the-Edge Perspective. Advances in Civil Engineering. 2022: 1–13.

[5] Dewangan, D. K., and Sahu, S. P. Deep Learning-Based Speed Bump Detection Model for Intelligent Vehicle System Using Raspberry Pi. IEEE Sensors Journal, 2021; 21(3): 3570–3578.

[6] Shahane V, Jadhav H, Sansare M and Gunjgur P. A Self-Driving Car Platform Using Raspberry Pi and Arduino. 2022 6th International Conference On Computing, Communication, Control And Automation (ICCUBEA). 2022.

[7] U. Karni, S. S. Ramachandran, K. Sivaraman and A. K. Veeraraghavan. Development Of Autonomous Downscaled Model Car Using Neural Networks And Machine Learning. 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), India. 2019; 1089-1094.

[8] Tiwari, R., Singh, D.K. Vehicle Control Using Raspberry pi and Image Processing. In: Singh, R., Choudhury, S., Gehlot, A. (eds) Intelligent Communication, Control and Devices. Advances in Intelligent Systems and Computing. 2018; Vol. 624, Springer, Singapore.

[9] B. Premalatha and C. Nataraj, “Linear Regression Based Machine Learning Model for Cataract Disease Prediction”, EAI Endorsed Trans Perv Health Tech, vol. 10, Jun. 2024.

[10] Lee, D.H., Kim, S.K., Kim, C.S. et al. Development of an autonomous braking system using the predicted stopping distance. International Journal of Automotive Technology. 2014; 15: 341–346.

[11] D. F. Llorca, V. M. Montero, M. Sotelo. Autonomous Pedestrian Collision Avoidance Using a Fuzzy Steering Controller. IEEE Transactions on Intelligent Transportation Systems. IEEE Journals & Magazine. 2011.

[12] Philip Koopman, Uma Ferrell, Frank Fratrik, Michael Wagner. A Safety Standard Approach for Fully Autonomous Vehicles. Semantic Scholar. https://www.semanticscholar.org /paper/A-Safety. 2019.

[13] Wong, A., Strous, L., Johnson, R., Grier, D.A., Swade, D. (eds). The Laws and Regulation of AI and Autonomous Systems. Unimagined Futures – ICT Opportunities and Challenges. IFIP Advances in Information and Communication Technology. 2020; 555.

[14] Nataraj, C., Karthikeyan, G., Bharathi, G.J. et al. Comparative analysis of direct coupling and MPPT control in standalone PV systems for solar energy optimization to meet sustainable building energy demands. Sci Rep 14, 22924, 2024.

[15] Ida Syafiza Binti Md Isa, Choy Ja Yeong, Nur Latif Azyze bin Mohd Shaari Azyze. Real-time traffic sign detection and recognition using Raspberry Pi. International Journal of Electrical and Computer Engineering (IJECE). 2022; 12(1):331-338.

[16] Aditya Anand, Rushiraj Gawande, Prathamesh Jadhav, Rima Shahapurkar, Anjali Devi and Nikhil Kumar Anand, A. Intelligent Vehicle Speed Controlling and Pothole Detection System, 6th International Conference on Energy and City of the Future (EVF’2019). 2020.