

## Fast Region Convolutional Neural Network Lane and Road Defect Detection for Autonomous Vehicle Application

Nazirah Zulkifli, Muhammad Amirul Idham Laili, Siti Nurshaline Mohd Saadon, Adizul Ahmad, Azlee Zabidi, Ihsan Mohd Yassin\*, Fadhlan Hafizhelmi Kamaru Zaman  
Megat Syahirul Amin Megat Ali, Mohd Nasir Taib

Faculty of Electrical Engineering  
Universiti Teknologi MARA Shah Alam  
Selangor, Malaysia  
ihsan.yassin@gmail.com



### ABSTRACT

Road markers are the main landmarks and provide critical information for traffic guidance and safety for drivers. These markers are especially crucial for an autonomous vehicle as this type of vehicle needs to make automated decisions to ensure the safety of other road users. Therefore, for an autonomous vehicle to avoid traffic accidents, these markers should be detected and localized accurately. Additionally, road defects may also present a hazard to autonomous vehicles. This paper proposes a Faster Region Convolutional Neural Network (FRCNN) to detect and localize road markers and potholes on the bird's eye view images of roads. Data were collected using the Point Grey Blackfly camera mounted on the roof of a car. The Inverse Perspective Mapping (IPM) algorithm was used to transform the images into a bird's eye view perspective. The bird's eye view perspective is particularly important as it is easier to detect objects of interest from the top view. The data was then used to train an FRCNN on MATLAB R2018a. The results indicate that the FRCNN was successful in the detection of the objects of interest with some overlapping issues, which will be addressed in future works.

**Key words:** Artificial Intelligence, Deep Learning Faster Region Convolutional Neural Network, Autonomous Vehicle, Object Detection

### 1. INTRODUCTION

In the era of Industrial Revolution 4.0 (IR4.0), there is an increasing shift towards computing in contrast to human-dependent manual processes. In the automotive industry, recent interests by large corporations (such as Google, NVidia, and others) in automatic navigation and driving vehicles demonstrate the shift mentioned above.

An Autonomous Vehicle (AV) is a vehicle able to transport

objects or humans which relies on automation equipment that absent from human handling [1]. Autonomous driving is not a new concept in the automobile industry. The first commercial car using this technology was invented on the 1920s [2] with some basic autonomous features. With recent technological advancements, active research has been done in this area to increase the level of driving automation. AV research integrates many fields, namely sensors, computer vision, artificial intelligence, pattern recognition, and precision intelligent control [3].

Since the past decade, many types of research have been carried out to enhance AV technology. Currently, several industry players (such as Audi, Ford, Toyota, and Honda) are extensively researching AVs [4]. A large variety of sensors enable the AV to perceive and analyze its environment [5][6]. These include cameras, Light Detection and Ranging (LiDAR), Radio Detection and Ranging (RADAR), odometry, and Global Positioning System (GPS) [1][7]. These sensors can be divided into two general categories: object and pose sensors. Object detection sensors detect and identify objects of interest (such as pedestrians, road curvature, lanes, etc.), while pose sensors (such as GPS) detect the position of the car. A common problem in AV is to detect objects of interest (such as lanes, hazards, and obstacles) while driving to ensure safe and accurate navigation. Though this is a natural process of human driving, the computer needs to be trained to understand the sensor inputs to make the same decisions equivalent to its human counterpart. For this project, a Faster Region Convolutional Neural Network (FRCNN) model was used to detect and localize two principal objects of interest, namely lanes and potholes. These two objects of interest are particularly important for the safe handling and navigation of autonomous vehicles.

The FRCNN is a part of a new artificial intelligence paradigm called Deep Learning, which mimics the learning ability of humans through a complex neural network structure. A clearer view of the road is achieved using the Inverse

Perspective Mapping (IPM) algorithm to transform the image into a bird's eye view. The FRCNN was then trained on sample regions containing the object of interest and tested on sample images of an independent dataset to evaluate its performance.

This paper demonstrates a Faster Region Convolutional Neural Network (FRCNN) to detect and localize road markers and potholes on the bird's eye view images of roads. Data were collected using the Point Grey Blackfly camera mounted on a car roof. Then, Inverse Perspective Mapping (IPM) algorithm was used to obtain bird's eye view images from the collected images.

The remainder of this paper is organized as follows: Section 2 presents a review of related literature, followed by the research methodology in Section 3. Results and discussions are presented in Section 4. Finally, concluding remarks are presented in Section 5.

## 2. RELATED WORKS

### A. Neural Network and Deep Learning

Neural networks simulate the structure and learning capabilities of the human brain [8]. Popularized in the 1980s and 1990s, the neural network became widely used due to its flexibility to learn in solving many practical problems [9]-[11]. Among its potential application is in the area of image classification. The task involves training the neural network to recognize objects of interest [8]. Earlier versions of the neural networks derive an offline feature set, which is used to train the neural network. However, there was a tradeoff between the richness of the feature set and the complexity of the neural network which is commonly referred to as the "curse of dimensionality".

Recent advancements in the field of computation have ushered in a new era of neural networks called Deep Learning neural networks. These networks are capable of extracting sophisticated features autonomously through the use of convolutional layers. The convolutional layers are then fed to a series of complex layers to learn to classify the features. Since their structures are relatively more complex, the training of such networks involves the use of Graphics Processing Units (GPUs) that are capable of performing massive computations in parallel, significantly speeding up the training process of these networks.

### B. Lane Detection in Autonomous Vehicles

It is essential to have a stable and reliable line detection image while an AV is moving in a traffic environment to ensure the safety of other road users. Any abrupt lane changes could

result in catastrophic accidents. Deep Learning networks are well suited for this task. However, to develop an accurate model, the network needs to be trained on a massive amount of data.

Lane marking detection has been investigated for more than two decades [12]. The road or lane detection problem has been studied using multi-modal data sources such as monocular vision [13], stereo vision [14], and LiDAR readings [15]. These methods were investigated in works such as in [13], [14], [15] and [16].

Research from [13] analyzed rough road region from complex background noise for dark channel-based image segmentation intending to decrease the processing workload for road detection. A new voting strategy was introduced to minimize the processing time of the road extraction stage. This method is based on the vanishing point of properties of the segmented region. Additionally, it is an alternate technique instead of using conventional pixel-wise soft voting. The segmentation region with maximum voting value will be selected as the road region. The interference from pedestrians, vehicles, and other obstacles can be eliminated from the road detection algorithm. The method has proven to be relatively fast, scoring approximately 40 times more quickly compared to different approaches.

Works by [17] implemented a GPU-based parallel lane detection algorithm. Three algorithms (median filter, parallel differential excitation, and parallel Hough transform) were combined to produce a lane detection algorithm computed on the Compute Unified Device Architecture (CUDA). The results suggest a speedup of 35.35 times faster by using GPU compared to serial-based computation.

Chen et al. [18] proposed an end-to-end learning method for self-driving cars that can control the proper steering angles automatically based on image frames captured by the front-view camera. The convolutional neural network (CNN) model was developed using the Caffe framework trained on the comma.ai dataset. The output layer of steering angles depended on the input layer raw image frame, which follows selected criteria such as speed, lane changes or turns, and road curvature. In this study, the RGB image was fed to the input layer and the predicted steering angle for the input image as the output layer. Euclidean loss was introduced as a loss layer during the training to compute the sum of the square difference between the prediction steering angle and the actual steering angle. The results suggest that the developed model could offer relatively precise steering of the vehicle.

To minimize the higher level of noise and computation time, research by [19] explored a durable and real-time vision-based lane perception algorithm with an effective

region of interest (RoI). The work suggested an algorithm consisting initialization, lane detection, and lane tracking that can process a gradient cue and a color cue together. A line clustering with scan-line tests to verify the characteristics of the lane markings. The Kalman filter was used to collect and track vanishing points (VPs). VP was applied during the initialization stage to find a rectangular RoI. The lane detection stage finds the A-RoI, which is a distorted trapezoidal region surrounding the lane markings. Combination of the proposed algorithm and the method that introduce such as color cues, line clustering, scan-line test, lane verification, and variable A-RoI gave accurate result while successfully removed false lane marking. Additionally, the system was able to perform in real time by the accumulation of statistical data.

In [20], a CNN real-time and robust leading vehicle detection system was presented. The proposed system consisted of a combination of lane detection and tracking, sky and road region recognition, and vehicle detection and tracking based on Multibox pipeline. The system establishes the lead vehicle, in which the RoI can be limited based on the lane detection and road and sky region recognition. Thus, vehicle detection can be applied to localize entire vehicles' regions per image accurately. The Hough transform was applied for lane detection to detect straight lines on the road. Inter-frame information was implemented for real-time and stable lane tracking. The system showed robust and accurate detection compared to baseline object detection methods.

Another technique to improve lane detection is by modifying the framework of deep neural networks to study its structure for visual analytics [21]. The multitasking CNN and Recurrent Neural Network (RNN) was used to simultaneously detect the presence and geometric attributes (location and orientation) of an object in multiple prediction tasks. This multitasking structure shares low-level features when adapted with the LeNet network structure. Furthermore, RNN was useful for visual detection when dealing with the spatial distribution of visible cues belonging to an object when the shape or structure is difficult to define. The RNN was designed as a lane boundary detector, which can be applied either for higher level models or not of traffic lanes. RNN was enabled when CNN was unable to determine a global model. The results suggest that the proposed framework is not only restricted for lane detection purposes but can also be applied to other visual perception tasks.

Another method for lane detection, which was proposed by [22], implemented the parallel detection mechanism using a spiking neural network (SNN). The input image for the neural network was obtained from a camera mounted on the vehicle. Non-prospective regions outside the road area were excluded to reduce the processing time of the algorithm. After

performing some image processing to remove noise, the spiking neural network with Hodgkin and Huxley neuron model was used for edge detection. For lane detection, The Hough transform was implemented. When the algorithm was implemented in parallel, the processing time was approximately 100ms on a computer with modest specifications. This edge detection algorithm was also able to detect weak edges better than compared methods.

Research from [23] utilized CNN to able to estimate the position of lane. The CNN was able to perform detection very accurately (up to centimetre precision accuracy). The facing-down camera captures the input image of the lane markings. Image of multiple parallel lanes and different colour of the lane are the example of models being trained in this paper. On the real-time embedded automotive platform, 99% of cases of lane position can be estimated with less than five-pixel error. The system has the potential to not only be used for lane detection but also applicable to real-time navigation as well as robustness testing of driver-assist or automated driving feature.

Another CNN based lane detection method was presented in [24]. A unified encoder-decoder approach was used for image segmentation, and the development was done using the Tensorflow framework. The method was tested on 181 images from the CaRINA dataset. Compared to other approaches based on the local gradient; the encoder-decoder structure was more durable and more straightforward for road marking detection without having to filter the outliers.

Common challenges in road marking detection are lighting and shadows. Addressing the issue, Bailo et al. [25] proposed a two-part algorithm based on Contrast Limited Adaptive Histogram Equalization (CLAHE), Inverse Perspective Mapping (IPM) and Maximally Stable Extremal Regions (MSER) for the detection and classification of road markings. CLAHE was used to adjust the road image to remove texture and histogram equalization. The method has been shown to be robust in detecting road markings in parallel with no increase in computational overhead. Density-based clustering was applied to RoIs to cluster detections. Besides, the MSER algorithm extracted a set of images generated from IPM and corresponding edge map to remove outliers. The proposed algorithm was found to be more effective compared to existing baseline methods.

### C. Pothole Detection

A pothole is a road defect and is one of the many obstacles during AV navigation. If the AV crashes into a pothole, it may cause severe damage to the AV and thus, the vehicle will lose control. In a study by [26], light sensors were used to detect potholes. Potholes emanate different Light Emitting

Diode (LED) brightness characteristics than its surroundings when sensed by the Light Dependent Resistor (LDR) sensor. The concept applied in this study successfully detected the obstruction for potholes and can be implemented to the AV.

Research by [27] suggested that Infrared Light (IR) also can act as a sensor for surface detection and recognition in different object surface types. However, the researchers found that distance and voltage might impact the sensitivity of the sensor during the detection of IR light. Therefore, several measurements were collected from different material to investigate various voltage characteristics for different distances on different materials. From this research, it was found that the proposed low-cost sensor was potentially applicable for surface detection and recognition such as AV, defect detection in industry, and etc.

Two-dimensional LiDAR and camera were proposed by [28] for pothole detection. Two sets of LiDARs were used to scan the road surface area accurately. The Raspberry Pi 3 was connected to 2D LiDAR and the camera to obtain the pothole information, such as its shape. From this study, the accuracy of the performance of pothole detection can be obtained by combining heterogeneous sensor data with 2D LiDAR.

Image processing and edge detection were proposed by [29] for pothole detection. Pothole images acquired were treated with Gaussian filtering to remove noise. Clustering techniques such as edge detection and thresholding, K-Mean and Fuzzy C-Means were also tested in the study. Based on the accuracy and detection performance, it is found that the clustering method using the k-mean technique had the fastest computation time and better edge detection compared to other tested methods.

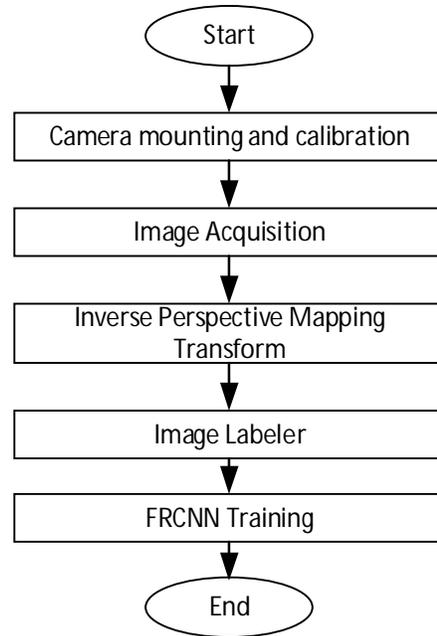
Works by [30] analyzed numerous potholes captured from video sequences. With the proposed algorithm, a threshold was used to select the off-line area of the image that contains pixel with high-intensity value as an RoI. The selected pothole RoI and several features such as size, shape, and length of the contour, the intensity of the surface and contrast of the image concerning its background were then analyzed. This study found that the proposed algorithm could detect and tracking the potholes with high precision and fast computation time.

### 3. METHODOLOGY

The research method is summarized in Figure 1. Each of the five steps is outlined in Section 3-A to 3-E.

#### A. Camera Calibration

The Point Grey camera was mounted on the car roof. While doing the camera setup, the camera pitch was to 8°, and the height from the ground was measured to be around 1.4 meters. The camera needs to be calibrated before data acquisition as the calibration results are essential for the IPM transformation. The calibration process is the most critical step, not only to make sure the position of the camera is on the right angle, but it also helps to minimize error caused by the camera during image capture.



**Figure 1:** Research methodology

The calibration process used a 6 x 8 inches checkerboard pattern, with equal checkerboard sizes. During the calibration process, the checkerboard was moved at various angles and tilted while the camera was acquiring the images (Figure 2). The calibration software then attempts to detect the unique edges present between the checkerboard boxes. Using this information, the calibration software calculates the camera parameters, which are then used to re-project the detection results back onto the original images. The reprojection error was used to determine whether the camera calibration was accurate.



**Figure 2:** Calibration process

### B. Image Acquisition

For the image acquisition, the road images were captured during daylight under sunny weather in Puncak Alam and Meru, Malaysia. The images were captured at approximately 30 frames per second, resulting in 10,000 still images in JPEG format.

### C. Inverse Perspective Mapping (IPM)

IPM is a mathematical transform that adjusts the original image to produce a bird's eye view. In IPM, the image is represented as a shifted coordinate system, with a rotation of the image and 2-dimensional matrix. This technique was used to achieve a more unobstructed view of the objects of interest. Figure 3 and Figure 4 show a sample of the IPM transformation.



Figure 3: Image before IPM

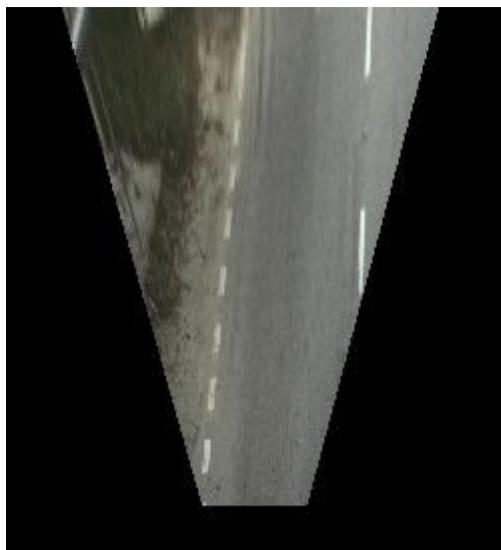


Figure 4: Result of IPM transformation

### D. Image Labeler

Image labeler was used to select regions of interest from the bird's eye view images. The labelling process was performed by manually marking the regions of interest using the MATLAB Image Labeler program.

### E. FRCNN Training

Deep Learning (DL) is a supervised machine learning technique extending the concept of Multi-Layer Perceptron (MLP), with the addition of feature extractors in the form of convolutional layers [31]. The retina-like convolutional layers extract simple edges and gradients and combine them to form more meaningful features for object recognition and detection. DL has found its way into the automotive industry, among them in Advanced Driver Assistance System (ADAS) [32] and can be represented in many architectures [33]. In this project, FRCNN was applied because of its ability to identify and localize lanes, arrow, and potholes based on the test image automatically.

The structure of the FRCNN, together with its parameters, is shown in Table 1. The structure of CNN consists of convolutional, normalization, Rectifier Linear Units (ReLU), pooling and fully connected layers. Several convolutional layers are placed alternately in between pooling and normalization layers. The training was performed in a supervised manner, using the gradient descent algorithm. The training method used a set of labelled examples with the desired outputs.

Table 1: FRCNN structure and parameters

Layer	Filter Size D × H × W	Other Parameters
Image Input	32 × 32 × 3	-
Convolutional	5 × 3 × 3	stride=1, padding=2
Max Pooling	3 × 3	stride=2, padding=0
ReLU	-	-
Convolutional	5 × 3 × 32	stride=1, padding=2
ReLU	-	-
Average Pooling	3 × 3	stride=2, padding=0
Convolutional	5 × 3 × 32	stride=1, padding=2
ReLU	-	-
Average Pooling	3 × 3	stride=2, padding=0
Fully Connected	64	-
ReLU	-	-
Fully Connected	2	-
Softmax	-	-
Classification Output	-	-

## 4. RESULT AND DISCUSSION

### A. Camera Calibration

The checkerboard detection results are shown in Figure 5. The checkerboard edges are highlighted in green circles, while the yellow square indicates the origin of the pattern.

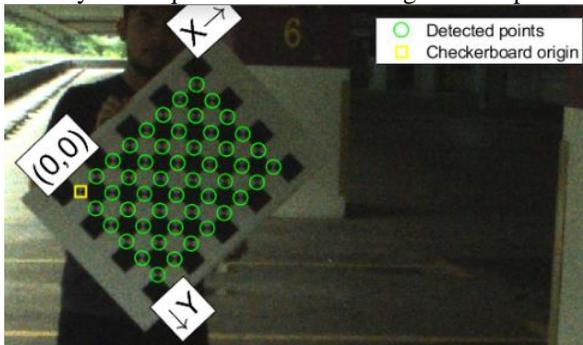


Figure 5: Checkerboard edge detection results

During the calibration process, the origin of the pattern was not detected correctly (Figure 6). These images were considered as outliers and removed before calibration.

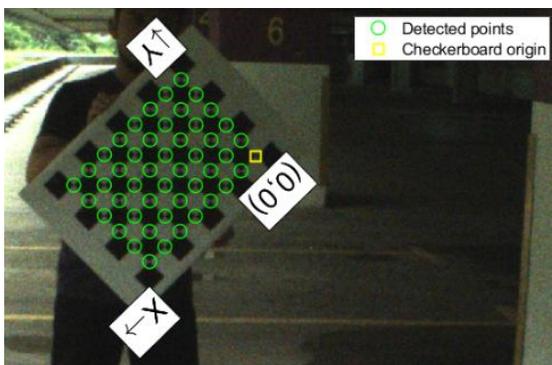


Figure 6: Example of outlier during the detection process. Note the position of the origin was rotated compared to Figure 5.

After the outliers have been removed, the calibration program attempted to calculate the camera parameters and re-project a set of new points based on the new parameters (Figure 8). As this was an estimation, there are slight differences between the re-projected and original positions.

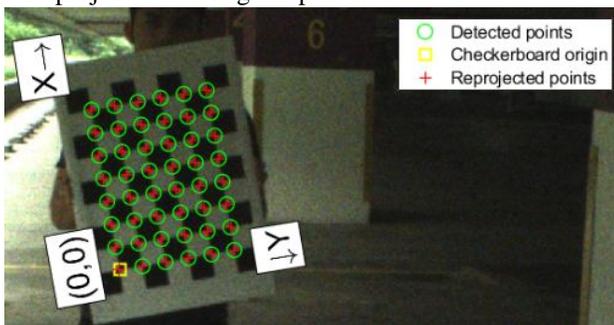


Figure 8: Sample reprojection results

The first-stage reprojection errors are summarized in Figure

9. For the second-stage recalibration, the images with the lowest mean projection errors were selected.

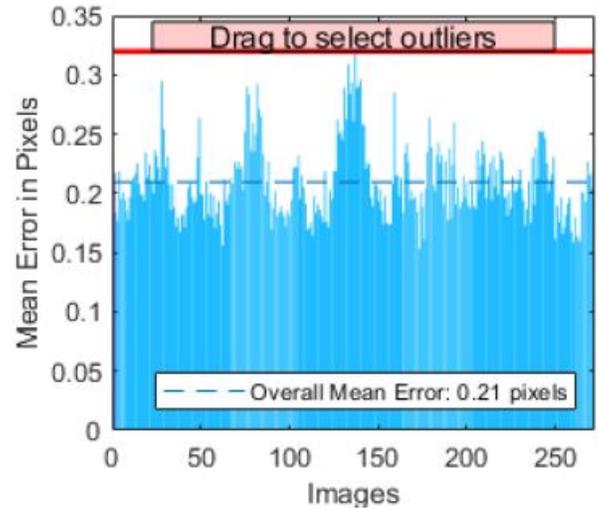


Figure 9: First stage reprojection error

The images with the least mean projection error was set to below 0.22. After the outliers have been removed, the second-stage calibration showed significantly fewer errors compared to the first-stage reprojection (Figure 10). Once the calibration was successful, MATLAB was configured to acquire continuously during data collection.

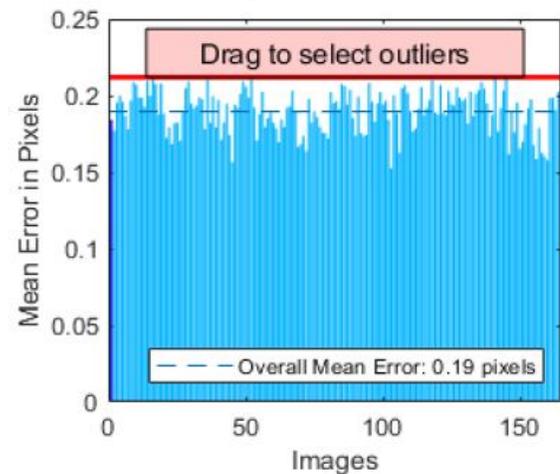


Figure 10: Second stage reprojection error

### B. Image Acquisition

From the 10,000 images captured, some selection was performed to remove unsuitable images. Initial frames captured were discarded as they contained a yellowish tint due to initialization of the Color Correction Matrix (CCM) of the Point Grey camera (Figure 11). After the CCM had stabilized, the images no longer exhibited a yellowish tint (Figure 12). Further selection was performed to select only images containing objects of interest.



**Figure 11:** Sample image pre-CCM stabilization



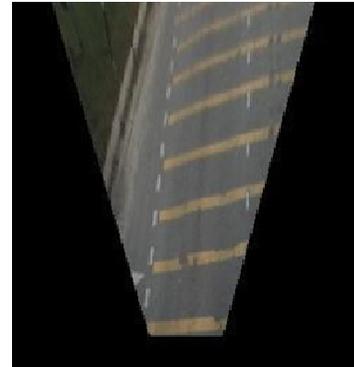
**Figure 12:** Sample image post-CCM stabilization

### C. IPM Results

The IPM was used to obtain a clearer view of objects of interest in labelling and classification. The images for lane detection and pothole detection are shown in Figure 13 (a-b) and Figure 14 (a-b). Both results show the IPM transforming the original images into a top-view, where objects of interest (lanes and potholes) can be seen. Apart from details of the lanes or potholes information that had been gained from the top view image, the distance of the lane or potholes also can be estimated.



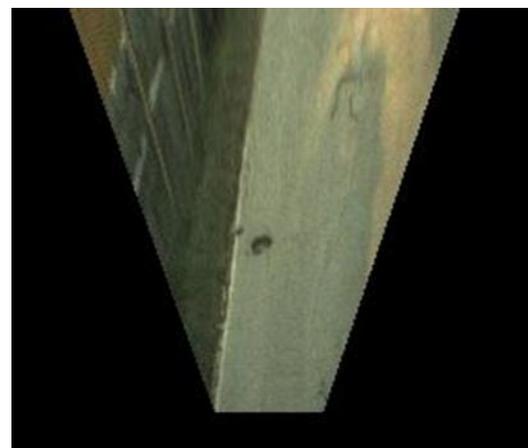
**Figure 13(a):** Sample lane image



**Figure 13(b):** IPM results for lane



**Figure 14(a):** Sample pothole image



**Figure 14(b):** IPM results for pothole

#### D. Ground Truth Labeling

Ground truth labelling marks regions of interest as the ground truth for training the FRCNN. The labelled regions of interest were lane, arrow and potholes (samples are shown in Figure 15(a) and Figure 15(b)).

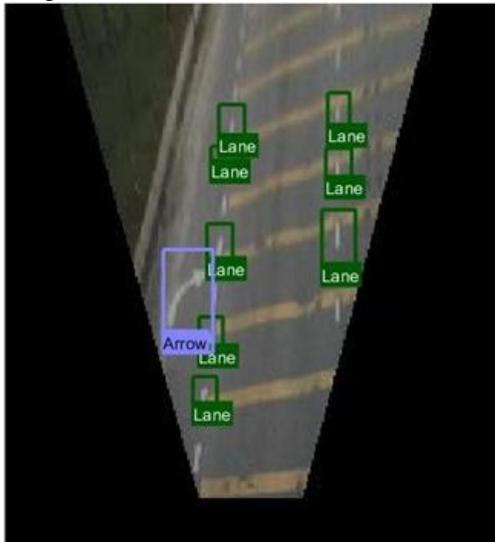


Figure 15(a): Sample ROI labelling (lane and arrow)

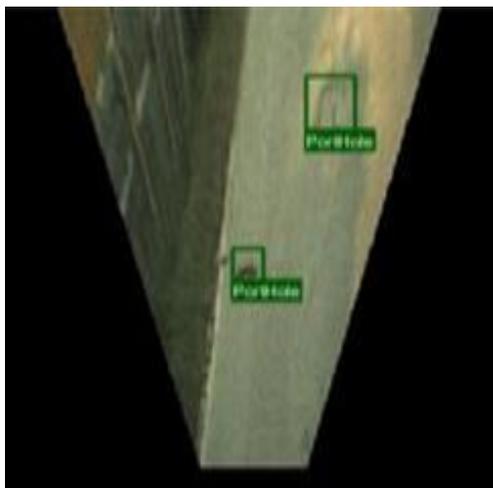


Figure 15(b): Sample ROI labelling (potholes)

#### E. Deep learning process

After FRCNN training was completed, the network was put to the test on a set of unused independent images which were not selected during training. Sample results are shown in Figure 16 and Figure 17. The FRCNN was set to mark areas where the detection confidence was above 85%. The detection box appears to mark the lanes and potholes, which indicates that the constructed FRCNN was successful. Some overlapping detections occurred for continuous-line lanes, and we will attempt to address this with additional detection class (solid line) in future.

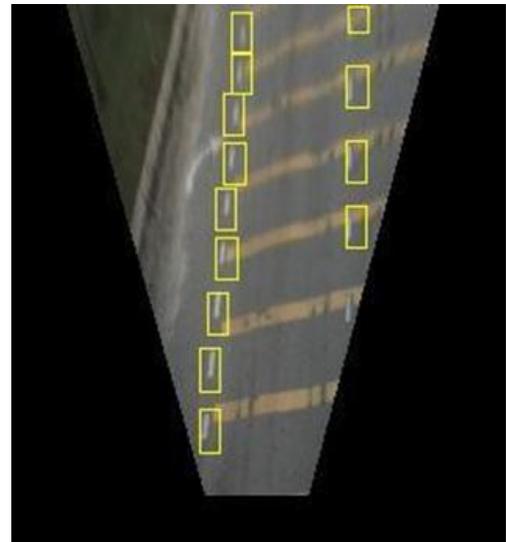


Figure 16(a): Result of FRCNN for lane detection with straight lane

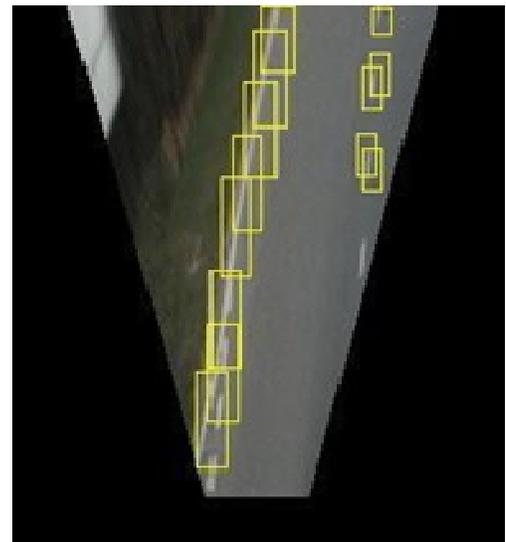


Figure 16(b): Result of FRCNN for lane detection with curved lane

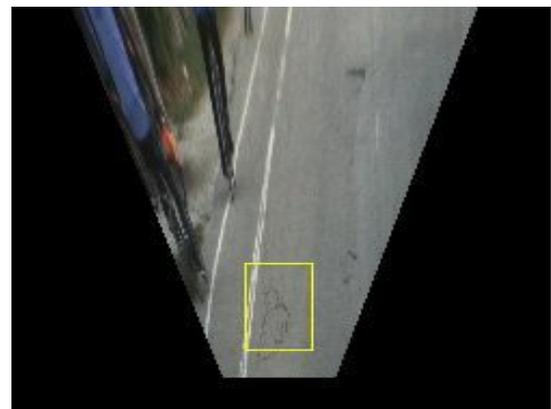


Figure 17: Result of FRCNN road with a pothole

## 5. CONCLUSION

In this paper, we demonstrate that FRCNN has successfully applied for detecting lanes and potholes. Transformation of the collected images into bird's eye view images was performed using Inverse Perspective Mapping (IPM) algorithm. The data were then used to train an FRCNN, and the results indicate that the FRCNN successfully detected the objects. Autonomous vehicles need to identify and recognize objects on the road to ensure the safety and navigation of the vehicle. With the advancement of Artificial Intelligence, the techniques proposed in this study can be used for high-level detection and localization of objects in AVs.

## ACKNOWLEDGEMENT

The author would like to thank Ministry of Education for the FRGS grant (600-IRMI/FRGS 5/3/ (081/2019)) and Faculty of Electrical Engineering, Universiti Teknologi MARA for the support.

## REFERENCES

1. H. J. Vishnukumar, B. Butting, C. Müller and E. Sax. **Machine learning and deep neural network - Artificial intelligence core for lab and real-world test and validation for ADAS and autonomous vehicles: AI for efficient and quality test and validation**, 2017 *Intelligent Systems Conference (IntelliSys)*, London, 2017, pp. 714-721  
<https://doi.org/10.1109/IntelliSys.2017.8324372>
2. V. Butakov and P. Ioannou. **Driving Autopilot with Personalization Feature for Improved Safety and Comfort**, 2015 *IEEE 18th International Conference on Intelligent Transportation Systems*, 2015  
<https://doi.org/10.1109/ITSC.2015.72>
3. J. Zhao, B. Liang, and Q. Chen. **The key technology toward the self-driving car**, *International Journal of Intelligent Unmanned Systems*, Vol. 6, No. 1, pp. 2–20, February 2018.  
<https://doi.org/10.1108/IJIUS-08-2017-0008>
4. A. Rasouli, I. Kotseruba, and J. K. Tsotsos. **Understanding Pedestrian Behavior in Complex Traffic Scenes**, *IEEE Transactions on Intelligent Vehicles*, Vol. 3, No. 1, 2018, pp. 61–70.  
<https://doi.org/10.1109/TIV.2017.2788193>
5. S. Lange, F. Ulbrich and D. Goehring. **Online vehicle detection using deep neural networks and lidar based preselected image patches**, 2016 *IEEE Intelligent Vehicles Symposium (IV)*, Gothenburg, 2016. Pp. 954-959.  
<https://doi.org/10.1109/IVS.2016.7535503>
6. B. C. Z. Blaga and S. Nedeveschi. **Online cross-calibration of camera and LIDAR**, 2017 *13th IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, Cluj-Napoca, 2017, pp. 295-301.
7. H. Yamada, T. Kobayashi, Y. Yamaguchi and Y. Sugiyama. **High-resolution 2D SAR imaging by the millimetre-wave automobile radar**, 2017 *IEEE Conference on Antenna Measurements & Applications (CAMA)*, Tsukuba, 2017, pp. 149-150.  
<https://doi.org/10.1109/CAMA.2017.8273384>
8. Z. Zhang, C. Xu, and W. Feng. **Road vehicle detection and classification based on Deep Neural Network**, 2016 *7th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, 2016.
9. Anon, (n.d.). **MIROS (Malaysia Institute of Road Safety Research)**. [online] Available at: <https://www.miros.gov.my/page.php?id=17>.
10. A. Deshpande, R. Sharma. **Multilevel Ensemble Classifier using Normalized Feature based Intrusion Detection System**, *International Journal of Advanced Trends in Computer Science and Engineering* Volume 7, No.5, September - October 2018  
<https://doi.org/10.30534/ijatcse/2018/02752018>
11. V. Diachenko, O. Liashenko, B. Fareed Ibrahim, Oleg Mikhal, Y. Koltun. **Kohonen Network with Parallel Training: Operation Structure and Algorithm**, *International Journal of Advanced Trends in Computer Science and Engineering*, Volume 8, No.1.2, 2019.  
<https://doi.org/10.30534/ijatcse/2019/0681.22019>
12. X. Liu, Z. Deng, H. Lu and L. Cao. **Benchmark for road marking detection: Dataset specification and performance baseline**, 2017 *IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, Yokohama, 2017, pp. 1-6.  
<https://doi.org/10.1109/ITSC.2017.8317749>
13. Y. Li, W. Ding, X. Zhang, and Z. Ju. **Road detection algorithm for autonomous navigation systems based on dark channel prior and vanishing point in complex road scenes**, *Robotics and Autonomous Systems*, Vol. 85, pp. 1-11, November 2016.  
<https://doi.org/10.1016/j.robot.2016.08.003>
14. L. Wang, T. Wu, Z. Xiao, L. Xiao, D. Zhao, and J. Han. **Multi-cue road boundary detection using stereo vision**, *In Vehicular Electronics and Safety (ICVES)*, pp. 1-6, July 2016.  
<https://doi.org/10.1109/ICVES.2016.7548169>
15. A. Ferraz, C. Mallet, and N. Chehata. **Large-scale road detection in forested mountainous areas using airborne topographic lidar data**, *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 112, pp. 23-36, February 2016.  
<https://doi.org/10.1016/j.isprsjprs.2015.12.002>

16. H. Zhou, H. Kong, L. Wei, D. Creighton, and S. Nahavandi. **Efficient road detection and tracking for unmanned aerial vehicle**, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 16, pp. 297-309, February 2015.  
<https://doi.org/10.1109/TITS.2014.2331353>
17. Y. Xu, B. Fang, X. Wu and W. Yang. **Research and implementation of parallel Lane detection algorithm based on GPU**, *2017 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)*, Shenzhen, 2017, pp. 351-355.
18. Z. Chen, X. Huang. *End-to-end learning for lane keeping of self-driving car*, *Intelligent Vehicles Symposium (IV) 2017 IEEE*, pp. 1856-1860, 2017.  
<https://doi.org/10.1109/IVS.2017.7995975>
19. C. Lee and J. H. Moon. **Robust Lane Detection and Tracking for Real-Time Applications**, in *IEEE Transactions on Intelligent Transportation Systems*.
20. H. Guan, W. Xingang, W. Wenqi, Z. Han and W. Yuanyuan. **Real-time lane-vehicle detection and tracking system**, *2016 Chinese Control and Decision Conference (CCDC)*, Yinchuan, 2016, pp. 4438-4443.  
<https://doi.org/10.1109/CCDC.2016.7531784>
21. J. Li, X. Mei, D. Prokhorov and D. Tao. **Deep Neural Network for Structural Prediction and Lane Detection in Traffic Scene**, *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 28, no. 3, pp. 690-703, March 2017.  
<https://doi.org/10.1109/TNNLS.2016.2522428>
22. X. Li, Q. Wu, Y. Kou, L. Hou and H. Yang. **Lane detection based on spiking neural network and hough transform**, *2015 8th International Congress on Image and Signal Processing (CISP)*, Shenyang, 2015, pp. 626-630.
23. A. Gurghian, T. Koduri, S.V. Bailur, K.J. Carey, V.N. Murali. **Deeplanes: End-to-end lane position estimation using deep neural networks**, *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2016.  
<https://doi.org/10.1109/CVPRW.2016.12>
24. L. R. T. Horita and V. Grassi. **Employing a fully convolutional neural network for road marking detection**, *2017 Latin American Robotics Symposium (LARS) and 2017 Brazilian Symposium on Robotics (SBR)*, Curitiba, 2017, pp. 1-6.  
<https://doi.org/10.1109/SBR-LARS-R.2017.8215297>
25. O. Bailo, S. Lee, F. Rameau, J. S. Yoon and I. S. Kweon. **Robust Road Marking Detection and Recognition Using Density-Based Grouping and Machine Learning Techniques**, *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Santa Rosa, CA, 2017, pp. 760-768.  
<https://doi.org/10.1109/WACV.2017.90>
26. C. W. Kit, P. J. Ling, and A. K. Chowdhury. **Detection mechanism of an autonomous vehicle: Self recognising obstructions**, *2016 Int. Conf. Adv. Electr. Electron. Syst. Eng. ICAEES 2016*, pp. 134-137, 2017.
27. L. Mikulas and S. Republic. **Surface Detection and Recognition Using Infrared**, *2014 Elektro*, pp. 159-164.
28. B.-H. Kang and S.-I. Choi. **Pothole detection system using 2D LiDAR and camera**, *2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN)*, 2017.
29. K. Vigneshwar and B. H. Kumar. **Detection and counting of pothole using image processing techniques**, *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICIC)*, 2016.  
<https://doi.org/10.1109/ICIC.2016.7919622>
30. I. Schiopu, J. P. Saarinen, L. Kettunen, and I. Tabus. **Pothole detection and tracking in car video sequence**, *2016 39th International Conference on Telecommunications and Signal Processing (TSP)*, 2016.  
<https://doi.org/10.1109/TSP.2016.7760975>
31. E. Nishani and B. Cico. **Computer vision approaches based on deep learning and neural networks: Deep neural networks for video analysis of human pose estimation**, *2017 6th Mediterr. Conf. Embed. Computer*, no. June, pp. 1-4, 2017.  
<https://doi.org/10.1109/MECO.2017.7977207>
32. H. J. Vishnukumar, B. Butting, C. Muller, and E. Sax. **Machine learning and deep neural network — Artificial for ADAS and autonomous vehicles: AI for efficient and quality intelligence core for lab and real-world test and validation test and validation**, *2017 Intelligent Systems Conference (IntelliSys)*, 2017.
33. P. Ongsulee. **Artificial intelligence machine learning and deep learning**, *2017 15th International Conference on ICT and Knowledge Engineering (ICT&KE)*, pp. 1-6, 2017.  
<https://doi.org/10.1109/ICTKE.2017.8259629>