Journal of the Society of Automotive Engineers Malaysia
Volume 4, Issue 2, pp 208-220, May 2020
e-ISSN 2550-2239 & ISSN 2600-8092

Lane Line Detection via Deep Learning Based-Approach Applying Two Types of Input into Network Model

N. J. Zakaria[1,2], M. I. Shapiai[*1], M. A. Abdul Rahman[2] and W. J. Yahya[2]

[1]Centre for Artificial Intelligence & Robotics, Malaysia-Japan International Institute of Technology, Universiti Teknologi Malaysia, 54100 Kuala Lumpur, Malaysia
[2]Advanced Vehicle System Research Group, Malaysia-Japan International Institute of Technology, Universiti Teknologi Malaysia, 54100 Kuala Lumpur, Malaysia

[*]*Corresponding author: md_ibrahim83@utm.my*

ORIGINAL ARTICLE ***Open Access***

Article History:

Received
15 Sep 2019

Received in revised form
11 Mac 2020

Accepted
12 Mac 2020

Available online
1 May 2020

Abstract – *Lane line detection is one of the important modules for Advanced Driver-Assistance System (ADAS) that are applied in the autonomous vehicle. This module work by exhibit the position of the road lane marking and providing the details of the geometrical features of the lane line structures into the intelligent system. This paper proposes the lane line marking detection using Fully Convolutional Neural Network (FCN) model by investigating the two types of input fed into the networks. RGB-channel (Red, Green, Blue) and Canny edge were used as the inputs to develop in the FCN model. The FCN approach has been proposed as one of the solution methods in mitigating the road lane detection issues due to its great performance in the application of objects detection in image or video. Previously, the RGB-channel is widely applied in the deep learning method meanwhile, the Canny-edge input has not been applied yet in the deep learning method. Therefore, this study investigates the further performance of this model by applying the canny edge as addition input besides applying only the RGB-channel. The data collections were acquired from real-time data collection. The result shows that the FCN model with the canny edge achieved a slight improvement with 96 % compared to FCN with the RGB-channel with 92 %.*

Keywords: Lane line detection, Fully Convolutional Network (FCN) model, RGB-channel, edge spatial

Journal homepage: www.jsaem.saemalaysia.org.my

1.0 INTRODUCTION

Lane detection is the process of presenting the position of the lane line into the system in intelligent vehicle (Kaur and Kumar, 2015). Generally, it is one of the significant modules in the Advanced Driver-Assistance System (ADAS) which help to improve the driver's performance during driving and reduce vehicle collision. However, detecting the lane line based on machine vision is uneasy task due to the existence of challenge situations such as the variety of road types and bad weather conditions. Especially in Southeast Asia, heavy rains regularly occur throughout the year that will affect the quality of the input images as the illumination changes that limit the driver's sight (Jawi et al., 2010). Therefore, the information on the lane line in the collected images is significantly important in order to train the network. It is functioning by learning the features of the target object to be detected in the images. Generally, it is heavily relying on the representation of the lane features in the image. The performance of the network training will improve with the high quality of images that will provide the well-defined dataset of lane line of the road.

Generally, there are quite a several publications (Laddha et al., 2016; Mendes et al., 2016; Oliveira et al., 2016; Simonyan and Zisserman, 2014; Mohan, 2014) of the lane detection models that have been published, nevertheless, it is noticed that the RGB-channel is widely applied in the deep learning method but lack of discussion on the other type of the inputs applied on it. For example, Laddha et al. (2016) propose a self-supervised approach which does not require any manual annotations and fine-tune an FCN based on VGG-net, and Simonyan and Zisserman (2014) using noisy labels for road detection. Meanwhile, Mendes et al. (2016) train an FCN model based on Network-in-Network (NiN) architecture which utilizes large amounts of contextual information. Next, Mohan (2014) proposes a deep deconvolutional network in combination with traditional CNNs for feature learning to road detection and Oliveira et al. (2016) propose a smaller network based on the encoder-decoder symmetric network to achieve a real-time data collection. RGB-channel based images yield powerful information of the road even though in the lack of good shape patterns and has been widely accepted as the crucial cue for detection the road lane markings. Consequently, it also requires less physical constraints on the road shape and vehicle speed, advance to a more flexible technique.

The drawbacks of the RGB-channel are involving with the luminance complex from the effect of the environmental conditions from the weather conditions and variation of road situations. Illumination is one of the critical issues as dealing with an open environment where the natural illumination may drastically alter the scene appearance. The illuminance in the input images changes based on environmental changes cause the RGB-channel image is not robust enough to overcome the brightness changes. Therefore, by utilizing colour information or pixel intensities only is not sufficient due to the information varies significantly with illumination and environmental conditions (Tapia-Espinoza & Torres-Torriti, 2013). Furthermore, there is a lack of texture and colour information under extreme situations (Alvarez et al., 2014). Colour appearance of the road varies significantly under severe lighting variations such as strong shadows and highlights conditions. Whereas, texture-based techniques are depending on well-built textures parallel to the road direction, in the form of lane markings for paved roads or tracks created by other vehicles in unpaved roads (Alvarez et al., 2014).

Hence, the additional type of the input network that can overcome the illumination issue is applied in the deep learning. Spatial structures such as shape, edge, and location of the selected objects that represent the more specific structure of the objects is proposed. In addition, numerous spatial structures alternative has not been discussed and studied, although it is supreme in order to increase the effectiveness of the network model. Therefore, in this study, the Canny edge detector is applied and utilized in the deep learning network model. Therefore, to the best of the author knowledge, the lane line segmentation in lane detection via deep learning method using the edge input type has not been proposed yet. Hence, this paper proposes a Fully Convolutional Neural Networks (FCN) based approach in detecting the lane line on the road applying two types of the input network. Therefore, the main contribution of this paper is to propose the extraction of the full-scale features for semantic segmentation using the existing FCN model and to further investigate the performance of the model by adding the type of input image which is canny edge during rainy conditions. It functions by segmenting the pixel-wise in the image that took the arbitrary size of the input lane line images. The training and testing datasets were obtained from the real-time data collection. The datasets consisted of various road conditions. The results showed that the FCN method can indeed detect the target lane on the road during rainy conditions and it is slightly increasing in term of performances when applying the canny edge as the inputs compared to RGB-channel. The real-time data collection is utilized to train, validate, and test the network model. The inputs of the network are 3-channel RGB images and 1-channel edge images fed into the same architecture. The results of these types of inputs are compared to evaluate the performance of the model which has high efficiency in detecting the lane line images in rainy conditions.

This paper is structured as follows: Section 2 explained in details the methodology applied in this study. Then, the experimental and modelling results are discussed in section 3. Finally, the conclusions and future works are summarized in Section 4.

2.0 METHODOLOGY

2.1 Configuration of Research Platform

The workflow for the configuration of the research platform is illustrated in Figure 1. Test vehicle which is Proton Exora furnished with equipment is utilized for the data acquisition purposes. The equipment used to help smooth the study such as monitor and camera sensor. The camera sensor is used to capture and recorded the road scenes and display it through the monitor. Meanwhile, the monitor is used to display and monitor the information processed by the camera sensor.

2.2 Experimental Setup and Data Collection

The information of the lane line on the road was obtained through the data collection in the form of video. Then, the video is extracted into frame form using the Python. Next, 12770 images frame were collected and then it is divided into 80 % for training, 10 % for validation, and also 10 % for testing. The data collection was conducted at the test platform (Malaysia's highways) from DUKE (Duta – Ulu Kelang) highway to LATAR (Kuala Lumpur – Kuala Selangor) highway. Therefore, at the urban area, the test platform for data collection was conducted from Sultan Yahya Petra Street to Cheras, Street, Kuala Lumpur. As of now, drivers need to perform the data collection by maintaining the vehicle in the middle of the lane to make sure that both lines on the right and left side are captured by the camera. There are some precautions and safety is taken into consideration prior to conducting the data collection on the test platform. Firstly, the vehicle speed is set at the medium speed which is around 40-110km/h only and the dataset for rainy conditions also apply only on the highway platform. The data collection for the rainy state will not be conducted in the urban area. It is to assure the safety of the driver, passengers and other surrounding vehicles during data collection are conducted especially in rainy conditions. The demographic profiles of the test platform are tabulated in Table 1. Meanwhile, the dataset of the data collection as shown in Figure 2 and the configuration of the whole process in lane detection are shown in Figure 3.

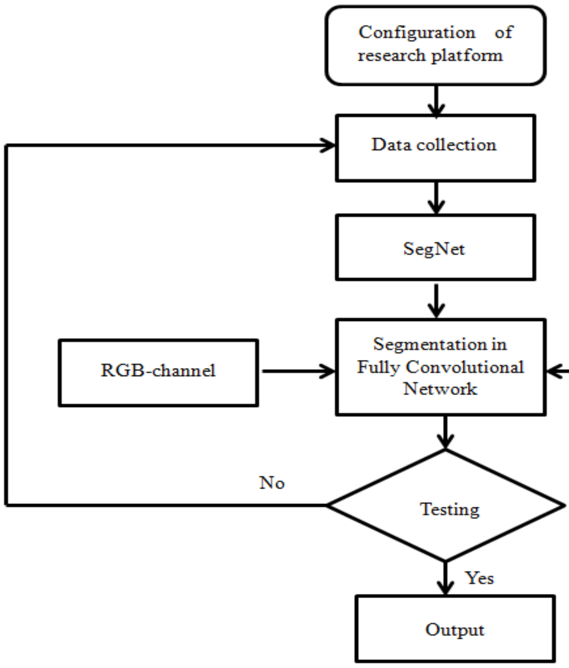


Figure 1: Work flow chart

Table 1: The demographic profiles of the test platform

Test platform	Conditions	Speed (km/h)
Highway state	Sunny	40-110
	Rainy	40-60
Urban State	Sunny	40-90

Figure 2: Images of the lane line

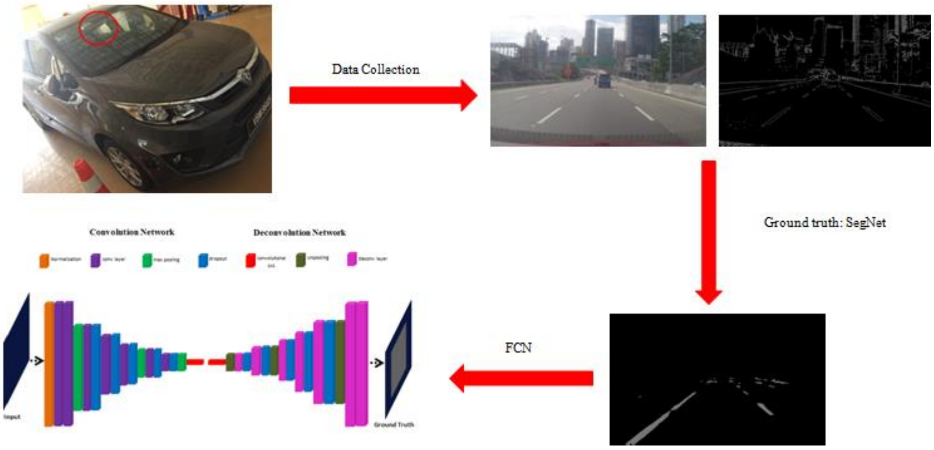


Figure 3: Lane detection process

2.2.1 RGB (Red, Blue, Green)- channel

The camera is mounted at the rear-view mirror inside the vehicle so that it can capture the highway and urban areas with variability in the environmental conditions. The variability of the road conditions in the dataset were varied in term of time, places, and weather. The total numbers of the dataset are 12770 which the FCN model is able to train and learn efficiently. The examples of the RGB-channel image as shown in Figure 2.

2.2.2 Edge spatial

The dataset of the edges is generated automatically from the RGB-channel images. Edges yield a details information of an image. Literally, the edge line of an image can bring high-level information for the neural network to train and learn specifically (Zitnick and Dollár, 2014). In this paper, the Canny algorithm is used to generate edges because it can accurately position the edge of the images and lessen the sensitivity of the edge (Song et al., 2016). Figure 4 shows an example of the image in gradient-based edges information dataset.

Figure 4: Gradient-based edges images

2.2.3 Ground Truth

The ground truths for the training datasets are annotated using the SegNet. It is presented by the members of Computer Vision and Robotics Group at the University of Cambridge, UK. It is used to annotate the lane line images of the training dataset. Firstly, every pixel in the images is annotated to present the classes for each class such as lane line, vehicle, sky, pole, tree, signboard, fence, bike, pavement and building. However, the focuses on this work is the orange colour in the images which represents the annotation for lane classes. The example of the ground truths is shown in Figure 5.

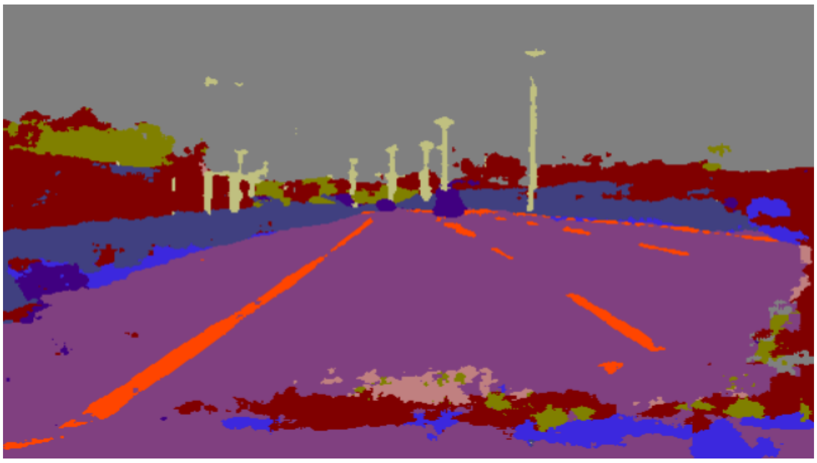


Figure 5: Ground truth using SegNet

After that, the ground truth is obtained by drawing the boundary lines along the long edges of lane markings for each frame. Meanwhile, the background and foreground of the images transformed to black colour and the lane line is converted to white as shown in Figure 6. Prior fit the labels together with the training datasets, the labels are normalized through dividing by 255 so labels were from 0 to 1 for 'G' pixel values. Next, the datasets are fed into the Fully Convolutional Network architecture.

Figure 6: Ground truth of the training dataset with blank (black colour) background and foreground

2.3 Model Network Detection

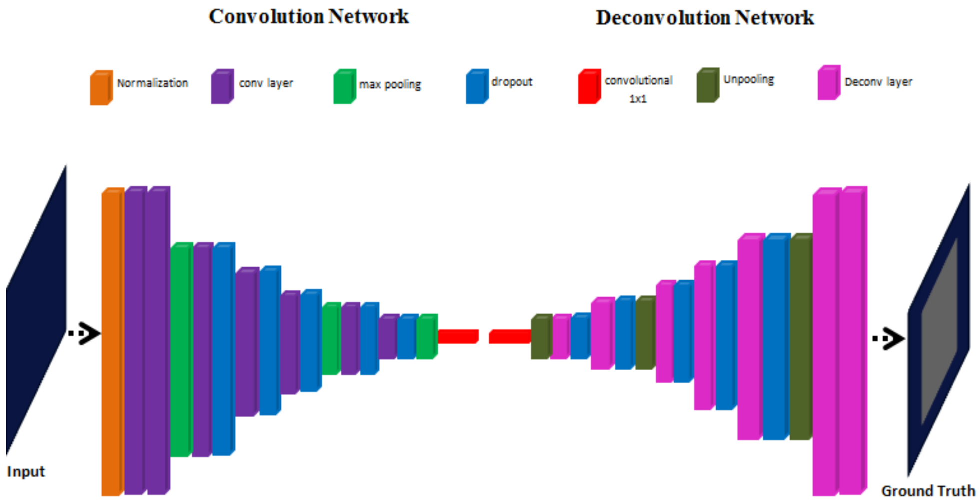


Figure 7: Fully Convolutional Network architecture

A Fully Convolutional Network (FCN) model involves utilizing multiple layered neural networks, which use mathematical properties to decrease error from predictions against actual to converge toward a final model, effectively learning as trained using the data images. The parameter value for the network architecture such as the number of epochs, batch size, number of classes and image shape is set as listed in Table 2. In order to learning and train the network model, the architecture of the network is designed which made up of one normalization layer, seven convolutional layers, three max-pooling layers, ten drops out layers, seven deconvolutional layers, and four unpooling layers. The architecture of the FCN is illustrated in Figure 7. The first layer of the network is the input image with pixel size 80 (height) x 160 (weight) x 3 for RGB while, 1 for edge spatial (colour channels. The input training images have its annotations as briefly discussed in the previous section. In addition, the training data and labels are patches into arrays and then the data is shuffled to make sure that the different images are presented rather than fit only on certain images. It is divided into two classes, lane and non-lane. The details of network architecture are as follows:

(1) At the first layer, a 3x3 kernel filter is utilized to filter the image that consists of 64 filters with 7 strides.
(2) Then, in order to increase the accuracy of the network, the normalization is applied at the next layer followed up by the convolution layer that functions to extract the images features.
(3) After that, the image size is reduced by half as the 2x2 max pooling layers are applied. Thus, reducing the FCN's memory requirements (Mendes et al., 2016). Max pooling is used after the second, fifth, and seventh convolutional layers to reduce the spatial size of the feature maps.
(4) Other than that, to prevent the over fitting problem, the dropout layer is added. The value of the dropout layers is set to 0.2.
(5) Therefore, to establish the size of the images to its original input size, the unpooling and deconvolutional layers are required. FCN model is trained and then generate 1x1 convolution of these layers. Therefore, the additional 1x1 convolution layers are present at the middle of the network architecture for the reason of:

(i) It increases the network depth enable it to learn high-level features of the images.
(ii) It decreases the number of channels by a factor of a half which is speeding up the training phase.

(6) The final deconvolution layer ends with one filter, because only necessary returned image in the ‘G’ colour channel. It is because the predicted lanes are drawn in green then is stacked up with zeroed-out red (‘R’) and blue (‘B’) channels to merge with the original road image.

The input images propagate through every layer and compared with the expected output to get the actual result. The padding and resize operations after the model are used to compensate for the effects of the convolutional and pooling layers respectively. Moreover, there are optimizers applied in this network. This step is needed to optimize the weights and minimize the error between actual output and expected output. The weights are updated to the optimum values required by the network. By reducing the error, the computational time of the network will decrease and it speeds up the process. The network can process arbitrary sized input images and the output of the network is corresponding to lane and non-lane, respectively.

Table 2: Parameters setting

Parameters	Values
Number of Epochs	10
Batch Size	8
strides	(1,1)
padding	‘valid’
Number of Classes	2
Image Shape	80 x 160 x 3
Activation Function	RELU
Kernel Filter	3x3
Max pooling	2x2
Dropout	0.2
Optimizers	Cross entropy and ‘Adam’

Next, the network is evaluated through the performance of the accuracy and loss of training and validation. Accuracy is the value of how accurate for the network in detecting the road lane. While the loss is the difference between predicted pixel values of the output lane image and what the lane image label was. Therefore, the network calculates loss through the Mean Square Error (MSE). It works by uses the mean of all the squared differences to calculate the loss. Lastly, the testing process is conducted using the dataset that is not included in the training dataset. In this study, the FCN model is implementing using Keras (“keras. models. Sequential”) in Python 3.5 along with Numpy, Sklearn and Scipy libraries. The neural network is created with convolutional layers (keras. layers. Convolutional2D) and fully connected layers (keras. layers. Dense).

3.0 RESULTS AND DISCUSSION

3.1 Training and Validation Results

For comparison purposes, the same architecture of FCN network model with different types of training input dataset is performed. In this study, an FCN model was not only fed with RGB-channel input but, the edge spatial was also applied to compare which types of input will achieve better performance during lane line detection especially in the rainy state. Figure 8 and Figure 9 shows the plots of the accuracy and loss for the training performances for non-rainy conditions. Meanwhile, Figure 10 and Figure 11 show the performance of the network in rainy conditions. In addition, Table 3 illustrated the overall accuracies and losses of the network model. The trained network has demonstrated a promising detection performance.

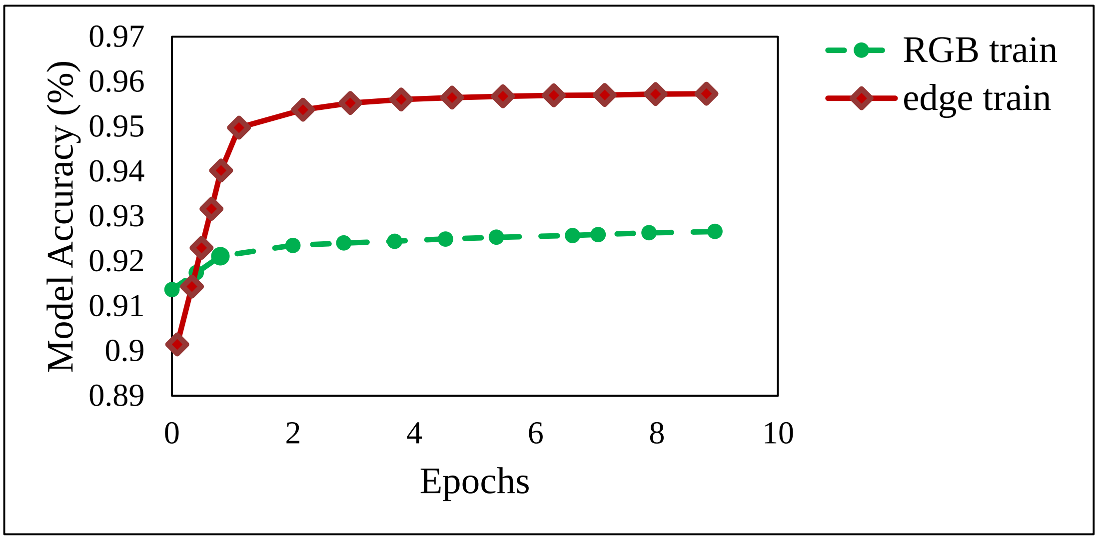


Figure 8: The plots of model accuracy in non-rainy state

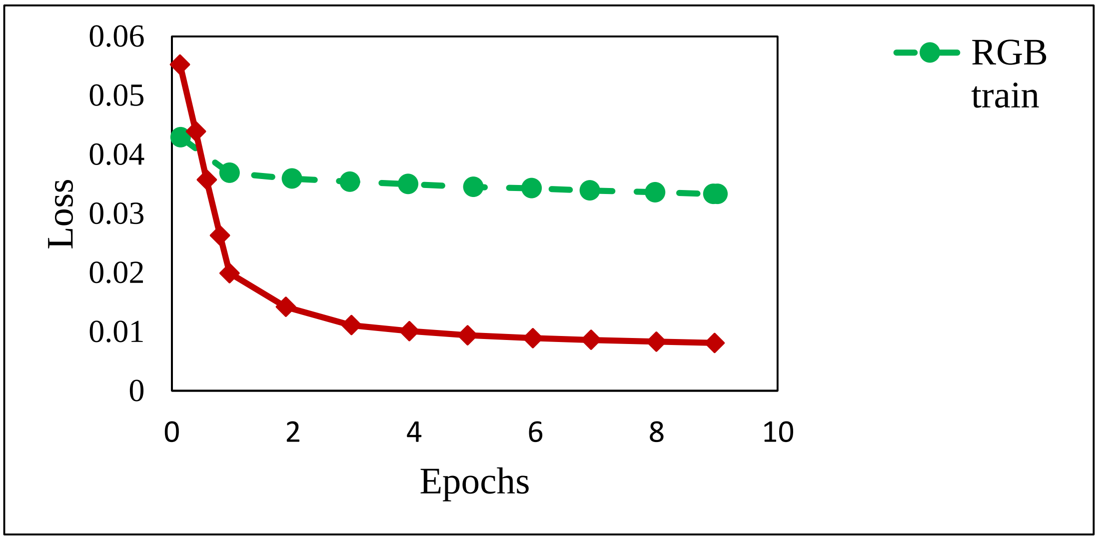


Figure 9: The plots of model loss in non-rainy state

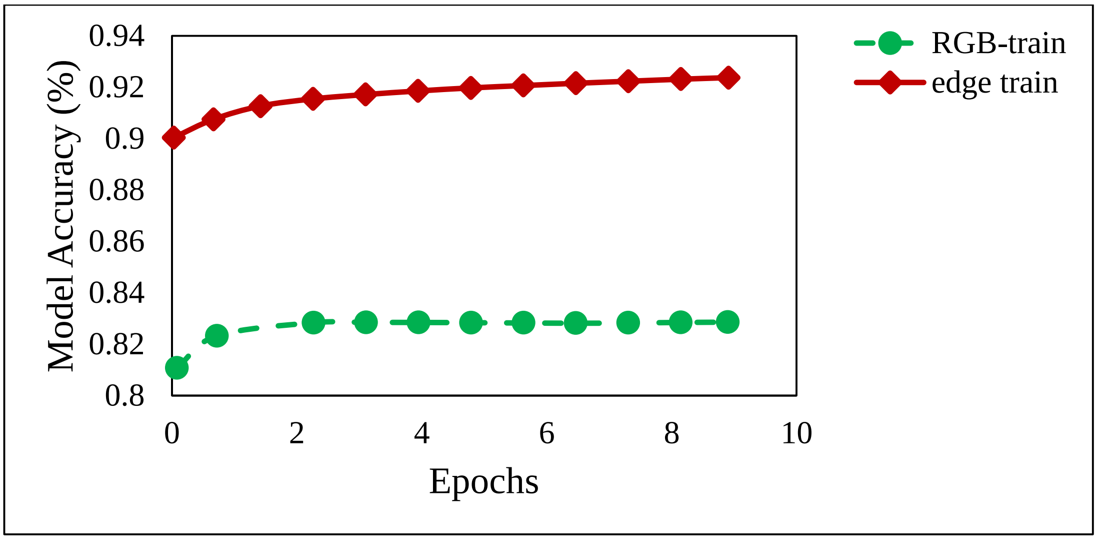


Figure 10: The plots of model accuracy in rainy state

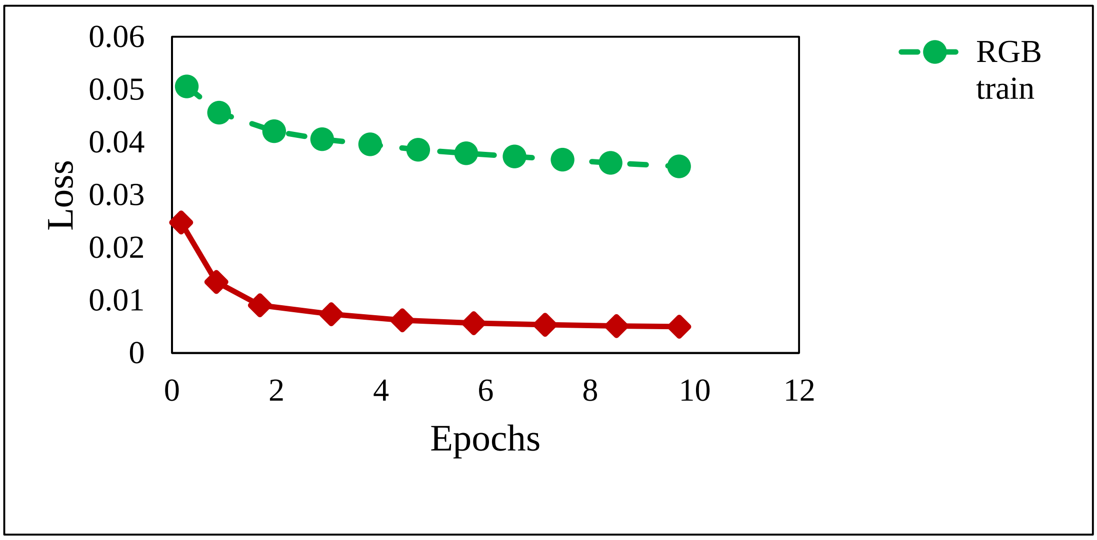


Figure 11: The plots of model loss in rainy state

Table 3: Model accuracies and losses

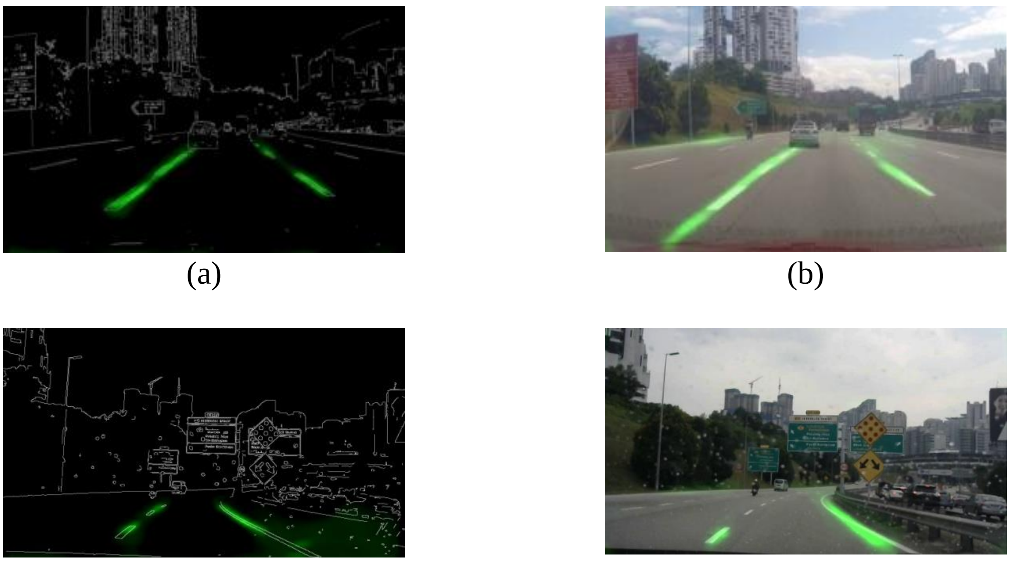
Cases	States	Accuracy (%)	Loss
FCN (RGB-channel)	Non-rainy	92.65	0.0334
	Rainy	82.88	0.0354
FCN (edge spatial)	Non-rainy	95.74	0.0080
	Rainy	92.40	0.0054

Model accuracy FCN (edge spatial) achieves a higher accuracy with 95.74 % in non-rainy state and 92.40 % in rainy state compare to FCN (RGB-channel) that achieve lower accuracy than FCN (edge spatial) which is 92.65 % during non-rainy state and 82.88 % in the rainy state. In terms of losses, the FCN (edge spatial) achieves the lowest number of loss which is 0.0080 for non-rainy state and 0.0054 for rainy. Meanwhile, the FCN (RGB-channel) get more errors which are 0.0334 in non-rainy state and 0.0354 in the rainy state. In addition, during the non-rainy state, regarding a trained data, the model of FCN (edge spatial) has practically achieved higher performance than FCN (RGB-channel) with differences about 10 % rate of accuracy percentage. Therefore, it also has a slightly low error compare to RGB-channel with different about 0.002. In the rainy state, FCN (edge spatial) also outperforms by slightly achieved the accuracy higher than FCN (RGB-channel) with very low error value. The differences in the accuracy and loss between these two are 4 % for accuracy percentage rate and 0.0026 differences in loss.

FCN (edge spatial) slightly has better performance in both state may due to its ability to provide the additional information for the model in the learning process as it has additional advantageous in feature extraction of the lane line images. Therefore, FCN (RGB-channel) contains more high number of loss may due to the RGB input consists more noise from the background meanwhile, the noise is reducing in edge input that it can provide additional higher-level feature than RGB-channel. The edge has provided an advantage and improve the detection of the road lane. It is important to get adequate accuracy performance and low error during training mode for the next process which is in the testing stage.

3.2 Lane Line Detection Result

The testing process is conducted after the training process. About 10 % of the input dataset is utilized in this stage. As the accuracy performance of the FCN model with edge spatial is much higher than the accuracy performance of the FCN model with RGB channel and have the low value of the loss value, it is clearly shown that the FCN with edge spatial input is more general than the FCN model with the RGB-channel. The limitation faced during the study which is challenging in data collection that was conducted at the test platform as there are many existing cars and the vehicle has a limited average speed for safety precautions. Figure 12 shows the results of the visualization for the lane line detection FCN model with two types of input sets.



(a) (b) (c) (d)

Figure 12: Results in the visualization. (a) FCN-edge (non-rainy); (b) FCN-RGB (non-rainy); (c) FCN-edge (rainy); (d) FCN-RGB (rainy)

4.0 CONCLUSION

In this paper, a lane detection model developed using Fully Convolutional Neural Network (FCN) model is presented. Input data for the model were obtained from data acquisition performed at real-time test platform using the testbed that is fully furnished with the equipment. The model is performed by fed with two types of lane inputs that are RGB-channel and edge spatial. The lane detection outputs had shown a decent performance which is the accuracy percentage for the detection of both types of inputs is more than 80 %. Furthermore, edge spatial input has slightly achieved better performance than RGB-channel as the percentage is more than 90 %. It works well even in challenge condition which is in a rainy state. It is may due to its ability to learn more discriminative features of the road lane than RGB-channel input. This demonstrates that the proposed FCN model is applicable and has a promising potential in detecting the lane line of the road. Next, the future work will have a focal point on other types of deep learning based-approach that will increase the network performances.

ACKNOWLEDGEMENTS

This work was supported in part by the CRG 8.1: AI Predictive Model for HVAC Operation under Grant Q.K130000.2443.04G73, and in part by the ASEAN NCAP Collaborative Holistic Research (ANCHOR) II under Grant 4B386.

REFERENCES

Alvarez, J.M., Lopez, A.M., Gevers, T., & Lumbreras, F. (2014). Combining priors, appearance, and context for road detection. *IEEE Transactions on Intelligent Transportation Systems, 15*(3), 1168-1178.

Jawi, Z.M., Isa, M.H.M., Sarani, R., & Wong, S.V. (2010). An exploration of weather threats to road safety in tropical country. In *4th International Conference – Expert Symposium on Accident Research (ESAR) 2010*, Hannover, Germany.

Kaur, G., & Kumar, D. (2015). Lane detection techniques: A review. *International Journal of Computer Applications, 112*(10).

Laddha, A., Kocamaz, M.K., Navarro-Serment, L.E., & Hebert, M. (2016). Map-supervised road detection. In *2016 IEEE Intelligent Vehicles Symposium (IV)* (pp. 118-123). IEEE.

Mendes, C.C.T., Frémont, V., & Wolf, D.F. (2016). Exploiting fully convolutional neural networks for fast road detection. In *2016 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 3174-3179). IEEE.

Mohan, R. (2014). Deep deconvolutional networks for scene parsing. *arXiv preprint arXiv*:1411.4101.

Oliveira, G. L., Burgard, W., & Brox, T. (2016). Efficient deep models for monocular road segmentation. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 4885-4891). IEEE.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Song, W., Liu, L., Zhou, X., & Wang, C. (2016). Road detection algorithm of integrating region and edge information. In *Proceedings of the International Conference on Artificial Intelligence and Robotics and the International Conference on Automation, Control and Robotics Engineering* (pp. 1-6).

Tapia-Espinoza & Torres-Torriti (2013). Robust lane sensing and departure warning under shadows and occlusions. *Sensors*, *13*(3), 3270-3298.

Zitnick, C.L., & Dollár, P. (2014). Edge boxes: Locating object proposals from edges. In *European Conference on Computer Vision* (pp. 391-405). Springer, Cham.