Optimized lean Convolution Neural Network for Embedded System to Classify Vehicles

Kiran B. Mohan[1], Ajeesh S[2], Smita C Thomas[3]

[1]M Tech Student, APJ Abdul Kalam Technological University, Kerala, India
[2]Asst. Professor, Dept. Of CSE, Mount Zion College of Engineering, Kadammanitta, Kerala, India
[3]Asst. Professor, Dept. Of CSE, Mount Zion College of Engineering, Kadammanitta, Kerala, India

---***---

Abstract - *Vehicle classification is much important these days for traffic analysis, autonomous driving, security, among others. Convolutional Neural Network surpassed older algorithms such as Support Vector Machine (SVM) and K-Nearest Neighbour in terms of accuracy, speed, and resource management. CNN is not suitable to deploy on embedded platform, even it has better accuracy and speed, because it has heavy resource consumption. Embedded platform have minimal resources and ANN should work with better performance using minimal resources. So that ANN can use for automotive use cases in car. In this paper proposes a best accuracy capability optimized lean Convolutional Neural Network that has smaller number of parameters, which is suitable for embedded platform, which utilize minimum resources.*

Key Words*: CNN, *Artificial Intelligence, Vehicle classification

1. INTRODUCTION

Artificial intelligence functions like pattern recognition, grouping, prediction, approximation, among other applications [1] [2] are able to done by Former Artificial Neural Network, but the arrival of Deep Neural Networks has opened a vast area of applications for artificial intelligence like image classification which can be applied on medical, security, automobile, commerce and other sectors around the world [3] [4] [5]. In recent years the increasing volume of vehicles on city roads has leaded to the creation of different kind of machine vision applications like security, surveillance, traffic analysis on roads, parking slots, among others [6], where the classification of vehicles is an important task and also is a difficult challenge for image classification algorithms. Vehicle classification has been done with algorithms like K-Nearest Neighbor and Support Vector Machine, the downside of this algorithms is that they are not suitable for remote applications using hardware with limited resources like an embedded system, due to the characteristics of this methods which is the process of extracting the features in background subtraction of images and also the operations done by the classifier [6], [7], [8], however the arrival of Convolutional Neural Networks which have become a relevant study topic due to their achievements in machine vision applications have turn into the go to algorithms for object classification and even object detection with Regional Convolutional Neural Networks (RCNN). The most used algorithm for image classification is CNN, which is used for wide applications such as security, health care, traffic monitoring, autonomous driving etc [9], [10], [11].

1.1 Convolutional Neural Network

CNN achieved great results in object classification, which have many CNN structures like ALEXNET, GOOGLE-LeNet, VGG16, which are having excellent capabilities for image classification of different types of objects[9][11][12]. Though these CNN structures have their advantages, they also use a considerable amount of computers resources and are not suitable for embedded systems. Their approach to this kind of complication is to use a lean network architecture proposed. A CNN that has the minimum number of layers but also has high accuracy, compared with network architecture that requires a significant amount resources for their training. As shown in Fig. 1 CNN are comprised of multiple layers of convolution network, then followed by a pooling layer and finally becomes a fully connected Artificial Neural Network. Convolutional and pooling layers are the functions to extract the characteristics from the input images to have the best classification possible using Convolutional Neural Network.

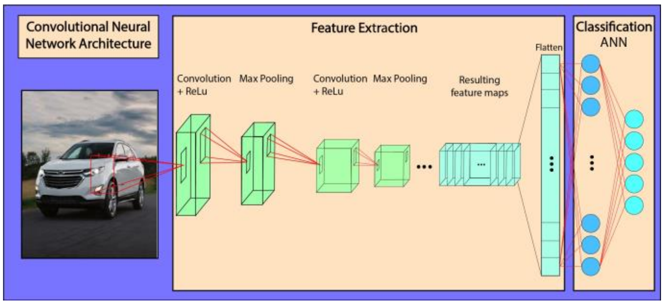

Fig -1: Convolutional neural network architecture

2. RELATED WORKS

Most of the algorithms used for vehicle classification problem, are used background subtraction with other characteristics of a vehicle image for the classification. There are implementations like the use of SVM with Histogram of Oriented Gradients (HOG) descriptors, SVM with SIFT features, KNN and virtual zones for vehicle counting, detection, among others [13], [14], [15]. Most of these implementations requires several algorithms to function like the feature extraction methods, the data conditioning and finally the classifier implemented, all of these algorithms consume considerable computer resources which can be seen as time consuming and this would not make them suitable for embedded applications. The high accuracy in classification task and implementation of CNN have made them one of the most used algorithms in machine vision applications like vehicle classification. Alexnet, VGG16 and VGG19 are the well known CNN structures, which are the evidence of the above. The structures are implemented in the way by using a pre-train model and then fine tune the network for better classification. It's not possible to introduce more classes to the network is the main disadvantage of the above networks [16], [17], [18].

3. EXISTING METHOD

Sensor-based and image-based methods are the two general categories of existing methods for vehicle classification.

Pneumatic pressure sensor, induction loop, magnetic sensor, and fiber-optic sensors are the sensors used for Sensor based classification to differentiate one class of vehicle from another. These sensors measure the passing vehicle's physical attributes such as weight, number axles, number of wheels, height, and even magnetic field in order to obtain unique features of each class of vehicle. For example, trucks will typically weigh more than cars while buses have more axles than cars. Although sensor-based methods are accurate in classifying the vehicles, they are usually more difficult to install and maintain, oftentimes requiring the temporary closure of roads. Furthermore, sensors that have to be embedded in roads have to be replaced if the road is repaved, thus increasing costs and complexity.

Image-based methods are usually easier to install and a single camera can capture more than just one lane of passing vehicle, thus making it more cost effective. Z.Chin presented the method to classify vehicle by using size and shape, they choose four classes of vehicle (motorcycle, van, car and bus) for classification with two algorithms SVM and random forests. SVM provide the satisfactory results as compared to the random forest, and misclassification occurs in between Car and Van because of some similarity in size and shape.

Another proposed system based on Linear SVM for classification task. They used the Scale Invariant Feature Transform (SIFT) algorithm for to extract local features and for model classification they used "bag of word". Model gave the accuracy of 89% on NTOUMMR data-set but the drawback with this data-set is all the images contain only front view of vehicles. Another framework in a distributed architecture for real-time vehicle detection and classification. In that system, the vehicles are classified with the combination of different techniques such as background subtraction, foreground detection, and feature extraction. The system showed best results with SVM in the daytime but the result is not satisfactory in the night and bad weather conditions.

4. PROPOSED MODEL ARCHITECTURE

CNN is comprised by convolution and max pooling feature extraction layers and finally a fully connected Artificial Neural Network as show in Fig. 1. Figure 2 presents the architecture of their CNN, the number of layers was decided by a previous research where it was found that with small structure, CNN still have a good performance [19] [20]. Their structure as seen in Fig. 2 is comprised buy five layers of convolution and five layers of Max Pooling, in the classification section they have an ANN of two layers which classifies 5 types of vehicles: Sedan, Buses, SUV, Pickup trucks and Motorcycles.

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 298, 298, 16)	448
max_pooling2d (MaxPooling2D)	(None, 149, 149, 16)	0
conv2d_1 (Conv2D)	(None, 147, 147, 32)	4640
max_pooling2d_1 (MaxPooling2	(None, 73, 73, 32)	0
conv2d_2 (Conv2D)	(None, 71, 71, 64)	18496
max_pooling2d_2 (MaxPooling2	(None, 35, 35, 64)	0
conv2d_3 (Conv2D)	(None, 33, 33, 64)	36928
max_pooling2d_3 (MaxPooling2	(None, 16, 16, 64)	0
conv2d_4 (Conv2D)	(None, 14, 14, 64)	36928
max_pooling2d_4 (MaxPooling2	(None, 7, 7, 64)	0
flatten (Flatten)	(None, 3136)	0
dense (Dense)	(None, 512)	1606144
dense_1 (Dense)	(None, 5)	2565

Total params: 1,706,149
Trainable params: 1,706,149
Non-trainable params: 0

Fig -2: Proposed model architecture and parameters

Basic process of feature extraction is feed initial image with resolution of 300x300 into the convolution layer with 16 filters of 3x3 and then reduce the size of the generated feature using 2x2 array by passes through the Max pooling layer. This process repeated 4 times with filters 32, 64, 64 and 64 per convolution layer. The max pooling layer stays with the same array. Finally the resulting feature maps of the last pooling layer are flattened and fed to ANN for classification as shown in the Fig. 2. A high resolution image brings more fine details for the CNN to extract and use for the classification, using high resolutions also brings some drawbacks like a bigger demand of memory usage and more computational operations being done to the image, so with a comparison between different researches with architectures like VGG16, VGG19, Alexnet and ResNet50 where there would be input resolutions of 224x224 pixels, but also there are cases were the datasets use images below 100x100 pixels, so it was decided on a resolution of 300x300 aiming to have more details on their images. The number of filters and their sizes was also selected by the comparison of different architectures like VGG16, Alexnet, ResNet50, where the most common use filter size would be 3x3, but the quantity would vary from 64 to 512 filters per convolution, and as mentioned before filters over each convolutional layer would increase(16, 32, 64, 128), as they are looking for lean network architecture.

Network layer function as described below.

A) Convolution Layers

The convolution layers are the core layers of this algorithm, because these are responsible for the feature extraction of the input image, each convolution layer has their own kernel and parameters. The eq. 1 shows the mathematical operation for this procedure,

$S = \max(0, X * K)$---- (1)

Where,

S is the resulting feature maps,

K is the filter bank used as kernel,

X is the pixels of the entered image.

The use of ReLu activation (max(,)) function is to gain non-linearity on the convolution layers.

B) Activation Function

The activation function indicates when the neuron will activate, there are different activation functions like Leaky ReLu, SoftMax, Sigmoid, Maxout, Tanh among others with different characteristics, in this model they use Relu for the first layer and for the last layer they use SoftMax for the ANN.

C) Max Pooling Layer

The Max Pooling layers serve as a robustness enhancer of the CNN to handle translations, usually the pooling layer is after the convolution layer. The pooling layer can be Max or Average Pooling, in their architecture they use Max Pooling to handle any existing translations.

D) Fully Connected Layer

The Fully Connected Layer is the neural network responsible for the learning of the different objects of interest, in their network they have a two layer Neural Network where the first layer has 512 neurons and the second layer has five neurons, the use of five neurons in the last layer is for the five objects to be classified.

5. ANALYSIS

We have done all the experiments using python with libraries like Keras and Tensor flow on the server with 12 physical cores and 100 GB of RAM. We have a callback to stop the training process once it reached an accuracy equal or above 95% in training, which to prevent over fitting of the network. The proposed CNN is trained with an image dataset that was created, with the use of existing free image sets on the internet, the images were selected manually to create the training set and validation set, used to feed the proposed algorithm. This dataset has 14144 in total images, 10880 for training and 3264 for validation, in which they are divided in 5 different classes Sedan, Buses, SUV, Pickup trucks and Motorcycles, which can be seen in Fig. 3. Data Augmentation was implemented for robustness and to create more training and validation samples without the necessity to find new images for the image dataset, also the images in this dataset have various resolutions. Over fitting may affect the performance of a network when classifying new data, So that in the tests networks were not allowed to surpass 95% limit in training accuracy.

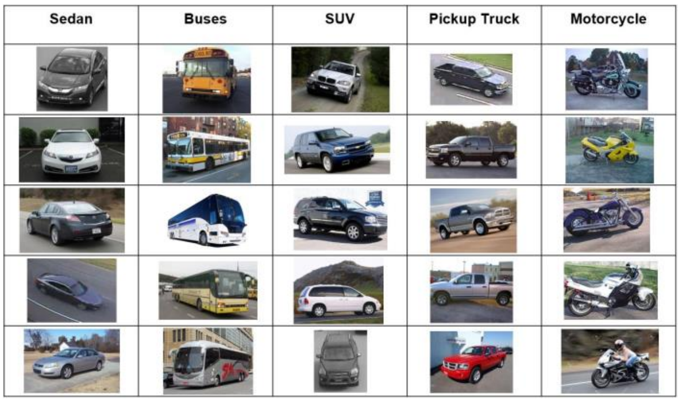

Fig-3: Samples of image dataset

In the proposed method section, model achieves a 96% in training accuracy and 76% in validation accuracy in ten epochs, able to complete the process in not more than 3 hours without Data Augmentation. To analyse the difference in time of training we taken this network as reference and perform modification on the proposed structure, implications on having more neurons in the first layer of the ANN and also the impact of having more training parameters with implementing a second convolution layer, Table 1 shown the results of the models trained. We were able to achieve 95% in training and 76% on validation of the CNN with just 10 epochs on Model 1, without Data Augmentation. Model 4, 5, and 6 have the better result in comparison, in validation accuracy almost 7% approximately. But the downside is, those models required more training time because of Data Augmentation is used and also have more training parameters by the extra convolution in the first layer for both models, Double number of neurons in the first input layer and also double number of filters in the last convolutional layer in Model 6.

Table -1: Table of model results

	% Ac. Training	% Ac. Validation	D. A.	Epochs	T. time
Model1	0.9662	0.7647	no	10	<3hrs
Model2	0.9339	0.8199	yes	200	40hrs
Model3	0.9507	0.8137	yes	50	<4hrs
Model4	0.9524	0.8327	yes	172	33hrs
Model5	0.9511	0.8392	yes	187	62hrs
Model6	0.9512	0.8382	yes	172	58hrs

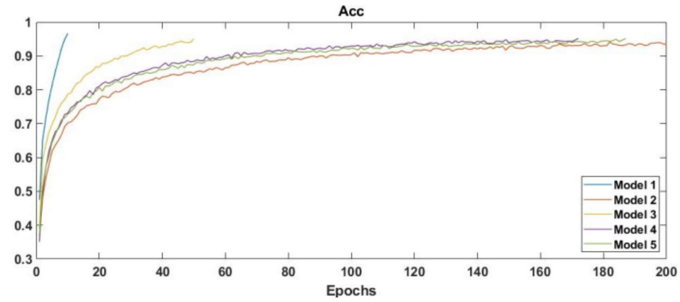

Fig-4: Models training accuracy

By double the number of neuron in the initial layer of ANN, Model 4 performs nearly as equal to mode 5 and 6, Also it has the half of the training time which can be because of the smaller number of parameters by not having a second convolution layer. Model 2 still did not reach the level of others because of it has smaller number of neurons on the input of the neural network takes longer time to reach 95% in training accuracy. With the test it's having a limit of 200 epochs in the training process.

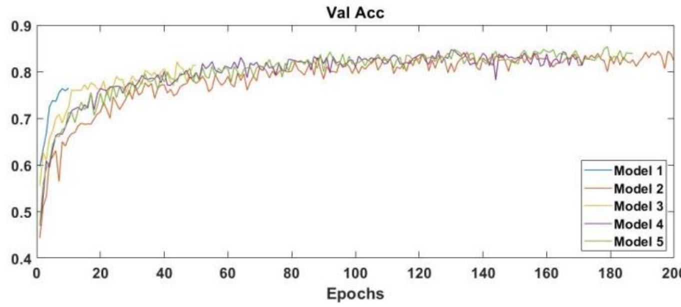

Fig-5: Models Validation accuracy

Model 3 has shown nearly as good result as Model 4, 5 and 6 but in less training time. As shown in fig. 4 the value of accuracy over 200 epochs of training where Model 1 is the first one passed above the 95% accuracy mark in less than 10 epochs, Also this model does not have Data augmentation. Model 3 took 5 timed more to epochs to pass 95% in training accuracy, due to data augmentation being used. From a time perspective it shows that it does not take five times more than Model 1, but it takes ten times less to reach stop mark in direct comparison with rest of the models (Model 2, Model 4, Model 5 and Model 6).

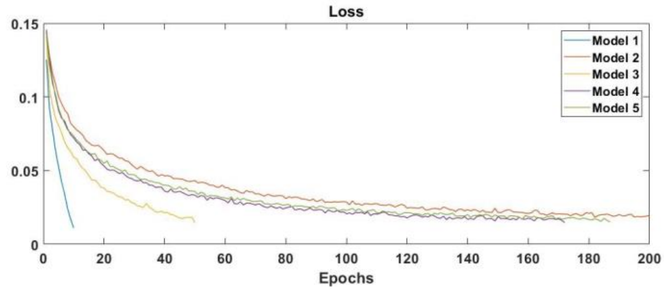

Fig-6: Models training loss

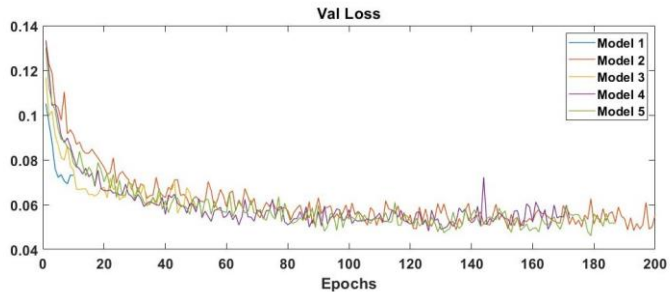

Fig-7: Model validation loss

From the Fig. 5, it shows similar behaviour in the validation accuracy during the 200 epochs in training process, But as per the results models do not surpass an 84% of accuracy on validation set, which means network still needs improvement, From Fig. 5 there is a notable stagnation of the validation results which is below 84%, there is also something notable in the trace of the result which is that the values obtained after each epoch varies more than those in Fig. 4, also when surpassing 80% in validation accuracy, there seems that almost all the models perform similar.

Fig-8: Vehicle classification test

Fig. 6 shows the results of the training loss function, which it shows the behavior of different models results is maintained. In accord to the accuracy obtained the value of loss function to decrease rapidly for Model 1, which the first model achieved the specified limit of training accuracy in less amount of time as shown in Fig 4. In Fig. 7 they have the validation loss function values, which fallows the same behavior found with Fig. 4 and Fig. 6, while the accuracy rises the loss function decreases and also in Fig. 7, like with models validation accuracy in Fig. 5 the same behavior of how the values vary more. Used Model 5 to analyze 24 new vehicle images to see if the relation of 80% on accuracy is maintained to prove that the proposed models are close to 80% of accuracy , Fig 8 shows the set of vehicle images correctly classified and incorrectly classified with green and red background respectively.

6. CONCLUSIONS

For vehicle classification a CNN was developed, and performed several tests to show the impact of the network performance by modify the network parameters and structures. Different categories of vehicle image dataset was created by own such as Sedan, SUV, Bus, Pickup Truck and Motorcycle. With the proposed CNN model and the image dataset build the obtained validation accuracy was above 80% using Data Augmentation and a lean network, these results showed that they could obtain a better classification accuracy by improving the image dataset for the network. Model 3 is the best model with a score 81.37% of validation accuracy, which in less than four hours of training time, as per the results shown in Table 1. Model 5 has the validation accuracy of 83.92%, but it took around 62 hours of training time, which not fit for embedded devices.

There is scope for the improvement in the network to increase the validation accuracy above 92% by increasing the image dataset. The exploitation of different innovative ideas in lean CNN architectural design has changed the direction of research, especially in image processing and classification using minimal resources. Architectural design of lean CNN is a promising research field and in future, it is likely to be one of the most widely used AI techniques for embedded platform

REFERENCES

[1] J. C. Rodriguez-Quinonez, O. Sergiyenko, F. F. Gonzalez-Navarro, L. Basaca-Preciado, and V. Tyrsa, “Surface recognition improvement in 3d medical laser scanner using levenberg–marquardt method,” Signal Processing, vol. 93, no. 2, pp. 378–386, 2013.

[2] G. Trujillo-Hern´andez, J. C. Rodr´ıguez-Qui˜nonez, L. R. Ram´ırez􏰀Hern´andez, M. J. Castro-Toscano, D. Hern´andez-Balbuena, W. Flores􏰀Fuentes, O. Sergiyenko, L. Lindner, and P. Mercorelli, “Accuracy improvement by artificial neural networks in technical vision system,” in IECON 2019-45th Annual Conference of the IEEE Industrial Electronics Society, vol. 1, pp. 5572–5577, IEEE, 2019.

[3] K. Makantasis, K. Karantzalos, A. Doulamis, and N. Doulamis, “Deep supervised learning for hyperspectral data classification through convo􏰀lutional neural networks,” in 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), pp. 4959–4962, IEEE, 2015. [4] Y. Seo and K.-s. Shin, “Image classification of fine-grained fashion image based on style using pre-trained convolutional neural network,” in 2018 IEEE 3rd International Conference on Big Data Analysis (ICBDA), pp. 387–390, IEEE, 2018.

[4] Y. Seo and K.-s. Shin, “Image classification of fine-grained fashion image based on style using pre-trained convolutional neural network,” in 2018 IEEE 3rd International Conference on Big Data Analysis (ICBDA), pp. 387–390, IEEE, 2018.

[5] P. Xia, J. Hu, and Y. Peng, “Emg-based estimation of limb movement using deep learning with recurrent convolutional neural networks,” Artificial organs, vol. 42, no. 5, pp. E67–E77, 2018.

[6] A. Arinaldi, J. A. Pradana, and A. A. Gurusinga, “Detection and classification of vehicles for traffic video analytics,” Procedia computer science, vol. 144, pp. 259–268, 2018.

[7] N. Seenouvong, U. Watchareeruetai, C. Nuthong, K. Khongsomboon, and N. Ohnishi, “Vehicle detection and classification system based on virtual detection zone,” in 2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE), pp. 1–5, July 2016.

[8] M. A. Manzoor and Y. Morgan, “Vehicle make and model classification system using bag of sift features,” in 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), pp. 1–5, IEEE, 2017.

[9] Q. Fan, L. Brown, and J. Smith, “A closer look at faster r-cnn for vehicle detection,” in 2016 IEEE intelligent vehicles symposium (IV), pp. 124–129, IEEE, 2016.

[10] B. B. Traore, B. Kamsu-Foguem, and F. Tangara, “Deep convolution neural network for image recognition,” Ecological Informatics, vol. 48, pp. 257–268, 2018.

[11] H. Gao, B. Cheng, J. Wang, K. Li, J. Zhao, and D. Li, “Object classification using cnn-based fusion of vision and lidar in autonomous vehicle environment,” IEEE Transactions on Industrial Informatics, vol. 14, no. 9, pp. 4224–4231, 2018.

[12] H. Gao, B. Cheng, J. Wang, K. Li, J. Zhao, and D. Li, “Object classification using cnn-based fusion of vision and lidar in autonomous vehicle environment,” IEEE Transactions on Industrial Informatics, vol. 14, no. 9, pp. 4224–4231, 2018.

[13] S.-H. Lee, M. Bang, K.-H. Jung, and K. Yi, “An efficient selection of hog feature for svm classification of vehicle,” in 2015 International Symposium on Consumer Electronics (ISCE), pp. 1–2, IEEE, 2015.

[14] M. A. Manzoor and Y. Morgan, “Vehicle make and model classification system using bag of sift features,” in 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), pp. 1–5, IEEE, 2017.

[15] N. Seenouvong, U. Watchareeruetai, C. Nuthong, K. Khongsomboon, and N. Ohnishi, “Vehicle detection and classification system based on virtual detection zone,” in 2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE), pp. 1–5, IEEE, 2016.

[16] R. Mehra et al., “Breast cancer histology images classification: Training from scratch or transfer learning?,” ICT Express, vol. 4, no. 4, pp. 247–254, 2018.

[17] Q. Fan, L. Brown, and J. Smith, “A closer look at faster r-cnn for vehicle detection,” in 2016 IEEE intelligent vehicles symposium (IV), pp. 124–129, IEEE, 2016.

[18] A. K. Rangarajan, R. Purushothaman, and A. Ramesh, “Tomato crop disease classification using pre-trained deep learning algorithm,” Proce�dia computer science, vol. 133, pp. 1040–1047, 2018.

[19] Y. Gao and H. J. Lee, “Vehicle make recognition based on convolutional neural network,” in 2015 2nd International Conference on Information Science and Security (ICISS), pp. 1–4, IEEE, 2015.

[20] X. Wu, R. He, Z. Sun, and T. Tan, “A light cnn for deep face representation with noisy labels,” IEEE Transactions on Information Forensics and Security, vol. 13, no. 11, pp. 2884–2896, 2018.

BIOGRAPHIES

Kiran B. Mohan received the B.Tech degree in Computer Science and Engineering from APJ Abdul Kalam Technological University, Kerala, India in 2012. He is currently pursuing M.Tech degree in Computer Science and Engineering from APJ Abdul Kalam Technological University, Kerala, India at Mount Zion College of Engineering, Kadammanitta, Kerala, India. His primary research interests are in Artificial Intelligence (Machine Learning oriented programming), Automotive and embedded system.

Ajeesh S received the M.Tech degree in Communication Engineering from MG University, Kottayam, India in 2016. He is currently working as Assistant Professor in the Department of Computer science and Engineering at Mount Zion College of Engineering, Kadammanitta, Kerala, India. His primary research interests are in Computer Networking, Data Communication, Logical system design etc.

Dr. Smita C Thomas. received the Ph. D degree in Computer Science and Engineering from Vels University, India in 2020. She is currently working as Associate Professor in the Department of Computer science and Engineering at Mount Zion College of Engineering, Kadammanitta, Kerala, India. Her primary research interests are in Data mining, Data analysis and Artificial Intelligence (Machine Learning oriented programming).