International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Improving Driver Safety Through Automated Traffic Sign Recognition Systems

Dr. Maria Jamal[1], Shubhi Gupta[2], Saloni Gupta[3]

[1]Associate Professor, Electronics and communication engineering, Indira Gandhi Delhi Technical University for Women, Delhi, India
[2,3]Electronics and communication engineering, Indira Gandhi Delhi Technical University for Women, Delhi, India

Abstract
Machine learning is increasingly vital, with automated traffic sign recognition representing a significant application where systems identify signs in real-time and instantly inform the driver, acting as a valuable backup against missed signs due to distraction. This capability enhances road safety by issuing timely warnings, helping prevent violations like speeding, reinforcing adherence to regulations, and lessening the driver's cognitive load for improved comfort. Technologically, these systems rely heavily on Convolutional Neural Networks (CNNs), which are trained using large, diverse datasets of labeled sign images, such as the German Traffic Sign Recognition Benchmark (GTSRB), to learn visual patterns under various conditions. Processing input from cameras and sensors, the trained neural network classifies encountered signs in real-time. Beyond simply alerting the driver, this recognized sign information can potentially be used to dynamically control traffic signals, display messages, or trigger other safety mechanisms, offering substantial potential to boost road safety, alleviate traffic congestion, and aid navigation, especially on unfamiliar routes or when signs are damaged or poorly visible.

INTRODUCTION

For various practical applications, including smart transport surveillance and analysis, recognizing traffic signs is a crucial computer vision problem. Although deep neural networks have been shown to deliver state-of-the-art performance for traffic sign recognition, the high computational and memory needs of such networks provide a significant barrier to their general use for embedded traffic sign recognition. As a result, there are several advantages to researching embedded device-friendly small deep neural network designs for traffic sign identification. Thus this paper will analyze the efficient CNN model for traffic sign detection and classification.

A. Motivation

Accidents from a variety of causes frequently compromise road safety. Although driving too fast is commonly acknowledged as the primary cause, a noteworthy survey found that the second most common cause of these accidents is drivers' ignorance of the precise meanings of traffic and road signs. This knowledge gap must be filled. To address this major accident cause, this work suggests an intuitive learning strategy that will assist people in efficiently learning and remembering traffic signs. This endeavor seeks to offer a better way to improve sign recognition and contribute to safer roads, even though automated educational tools are becoming more and more common and there are other models available, they frequently have drawbacks.

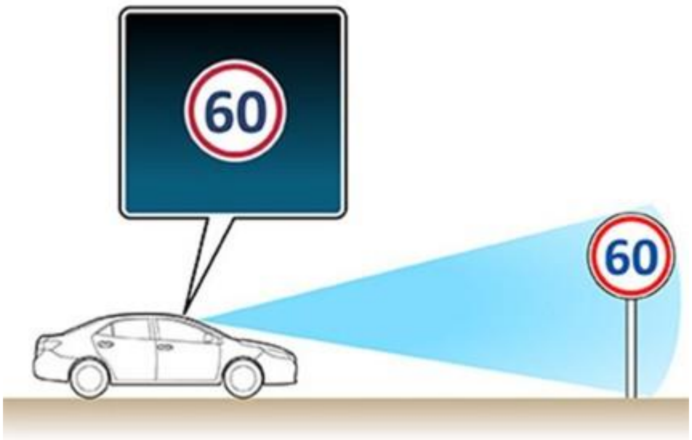

Fig. 1. A car recognizing speed sign

LITERATURE SURVEY

To recognize traffic signs more quickly, efficient algorithms require real-time traffic sign identification leveraging dataset resources. Additionally, manual feature extraction is not possible with convolutional neural networks. With systems of moving, learning, and reacting to continually improve power on the road through traffic sign recognition, the sensory activities of the human brain are frequently well imitated. In this section, we look at the drawbacks of the traditional LeNet-5 and pair them with Conv.NN's benefits for graphical identification. The LeNet-5 model performs well for single targets when recognition and characterization are used. Character recognition preparation struggles to provide adequately accurate recognition rates during peak periods, and the lack of an integrated preparation network considerably lowers the organization's recognition performance. Road traffic sign recognition depends on gathered dataset assets and makes use of potent grouping calculations to recognize distinct and criticize cunning cars exactly and gradually. Conv.NN produces the characterization findings through the constructed classifier based on picture highlights after clearly separating the includes from the information recognition picture. This circumstance shows that Conv.NN has excellent realistic recognition performance. The organization, recognition, and detection of road traffic signs can be gradually improved by using a forward-looking learning and critique tool to imitate the tactile cycle of the human mind. The shortcomings of the conventional LeNet-5 organizational design are examined in this area, and a model that extends the exceptional strengths of Conv.NN in image recognition is significantly improved. The training image for traffic signs contains some reference data on edges and regions of interest. Consequently, pre-processing the image is necessary. An image's cropping can be a crucial stage in the processing of images. The backdrop portion of a road sign, which makes up 10 percent of the overall image, is not used for recognition. For proportionate framing, the bounding box coordinates that the ROI recovered are used. Unwanted areas can be removed to lower the amount of information and increase the network's capacity for learning. The same road sign might be read differently depending on the illumination. Various image enhancement techniques minimize image noise. To modify grey values, conventional conversion techniques are applied. This technique enhances image quality while lightening the network's computational burden. Different sizes of the same road signs are available. Convolutional neural networks can be trained with training images of varying sizes due to varied feature sizes, which can make detection and recognition difficult.

A template is a processing object used in image recogni- tion. These patterns are representations of traffic signs. Simply enough, classifying and recognizing objects is pattern recog- nition. People are always looking for patterns. When someone sees something, whether it be a picture, a language, or text, their senses are stimulated, and memories are automatically triggered in their brains. The template library for this project shares the same memory. However, you can also refer to this design as picture recognition since it is intended for image recognition. We must build a pattern library since pattern matching is how we identify objects. The template collection only contains RGB graphics. Before utilizing the templates, compare them to the binary template library. The method was discovered to better detect RGB photos. So, RGB images are utilized in the template library. 164 traffic sign template pictures, which includes 51 prohibition signs, 77 warning signs, and 36 signs that include practically all three types of signs, have been added to this template library for maximum contrast. Pattern matching is the process in which the first step is to compare templates from the template library to each component of the image. Then it finds whether the image and the template have the same components. Finally, the region of interest's exact position inside the input image is established. First, we can overlay the image f (x, y) with the detection object template t(x, y). Second, we look for connections between the input photos and patterns in the pattern library and the patterns in the pattern library themselves. To find the exact location of the object, we must first identify whether there is an object in the image based on maximal similarity or whether it exceeds a predefined threshold.

Feature Extraction: Extracting features from input photos, which may come from complicated natural surroundings, creates difficulties since several locations may share colors comparable to traffic signs. To begin processing, edge detection and HSV color segmentation are used to build the necessary edge and contour maps. Because of the visually identical backdrop features, relying merely on color attributes for identification might result in considerable mistakes. As a result, shape segmentation is an important next step since it helps eliminate these interfering features while also improving overall recognition performance. Although traffic signs might have elaborate embellishments, their basic shapes are usually identified as triangles, circles, or rectangles. Potential regions of interest in the image are identified by examining three comprehensive shape descriptors: circularity (Y), rectangularity (J), and elongation (S).

The training process for the target neural network developed in this study involves initially training on a dedicated dataset and subsequently assessing its recognition accuracy on a separate validation set. The training regimen is adjusted and potentially continued based on performance observed during evaluation, with the final measure of the network's precision determined using an independent test set. This methodology is designed to cultivate superior generalization and classification performance from varied perspectives. To bolster the network's ability to handle variations and effectively increase the dataset size, data augmentation is employed using the Imgaug library. Techniques applied include transformations like rotation, blurring, and grayscale conversion, along with methods specified in this research, such as 50% image shading, random cropping, 50% color conversion, and filling designated pixel values.

This expansion of the GTSRB dataset via augmentation, coupled with processing it in smaller batches, serves to enhance the network's generalization while also mitigating computational resource requirements. The architecture itself, termed TS-CNN, is a 10-layer convolutional neural network. It predominantly features convolutional layers for progressive feature map extraction from inputs and max-pooling layers to systematically reduce feature map dimensionality. By incorporating additional layers, features are captured across different scales. Following necessary dimensional adjustments, a fully connected layer utilizes a Softmax activation function to perform the final traffic sign classification. To prevent overfitting and further improve generalization, dropout is incorporated; this technique randomly disables a portion of neurons during the forward pass of training, thereby reducing parameter interdependence. To locate the precise location of the object, we must first determine whether there is an object in the image based on the maximum similarity or whether it exceeds a predetermined threshold.

Faster R-CNN has two modules, and Mask R-CNN is an extension of it. The first module, known as the Local Proposal Network (RPN), is a deep fully convolutional network that receives an input image and produces a set of rectangular feature proposals, each with a score for objectivity. The second module divides the suggested regions into several predetermined categories using a regional CNN dubbed Fast R-CNN. Because it distinguishes between convolutions be- tween different sentences, fast R-CNN is particularly effective. To further enhance the quality of the suggested regions, we additionally perform bounding box regression. Using a shared convolutional function, the entire system is one integrated net-work that combines RPN and Fast R-CNN. The RPN module directs the Fast R-CNN module where to look, using the phrase ”attention” neural network, which has lately gained popularity. Then, Mask R-CNN improves this system by fusing a Feature Pyramid Network (FPN) with the fundamental network de- sign. Because FPN pulls characteristics from lower layers of the network before down sampling eliminates crucial details from small objects, it enables detectors to perform better on small objects. Residual Network (ResNet) of Mask R-CNN replaces the fundamental network design, VGG16 of Faster R-CNN. Faster and Mask R-CNN is trained on classification and domain proposal problems. Stochastic gradient descent is used in this process. Mask R-CNN uses end-to-end learning to train two networks at once. This was accomplished by the original Faster R-CNN implementation using a four-step interleaved optimization procedure. Faster R-CNN is, however, also applied the innovative end-to-end learning scheme of Mask R-CNN. Both networks are often trained on certain domains before being initialized using pre-trained ImageNet models.

Proposed Model

The design of our suggested CNN model includes the following layers: • Conv2D layer - Two convolutional layers with 32 filters each, measuring 5x5, and by considering ReLU activation. • MaxPool2D - Max Pooling layer with two layers of two by two. • A dropout layer with a 0.25 rate. • Two 3x3 convolutional layers, each with 64 filters. • A dropout layer with a 0.25 rate. • Flatten layers to compress them to a single dimension. • A dense feed-forward neural network with 256 nodes and ReLU activation. • Layer Dropout (0.5).

Dense layer (46 nodes, SoftMax activation). • MaxPool2D - Maxpooling layer is used for feature selection and to compress images. • Dropout - The implementation of a dropout layer forces every neuron in the network to take part in the training procedure. Additionally, it is another regularisation technique that helps to lessen overfitting. • Compress the layers by flattening the parallel layers. • SoftMax activation function for multi-class classification will be present in the top layer.

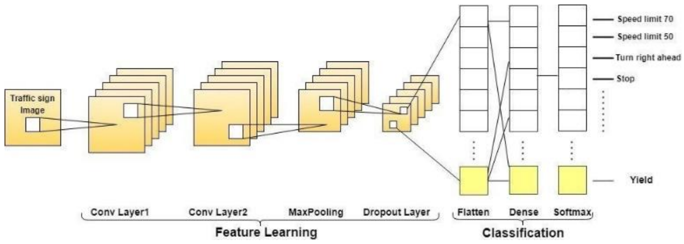

Fig. 2. Model Architecture

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 26, 26, 32)        2432

 conv2d_1 (Conv2D)           (None, 22, 22, 32)        25632

 max_pooling2d (MaxPooling2D  (None, 11, 11, 32)       0
 )

 dropout (Dropout)           (None, 11, 11, 32)        0

 conv2d_2 (Conv2D)           (None, 9, 9, 64)          18496

 conv2d_3 (Conv2D)           (None, 7, 7, 64)          36928

 max_pooling2d_1 (MaxPooling  (None, 3, 3, 64)         0
 2D)

 dropout_1 (Dropout)         (None, 3, 3, 64)          0

 flatten (Flatten)           (None, 576)               0

 dense (Dense)               (None, 256)               147712

 dropout_2 (Dropout)         (None, 256)               0

 dense_1 (Dense)             (None, 43)                11051

=================================================================
Total params: 242,251
Trainable params: 242,251
Non-trainable params: 0
_________________________________________________________________
```

Fig. 3. Model Parameters

SYSTEM DESIGN

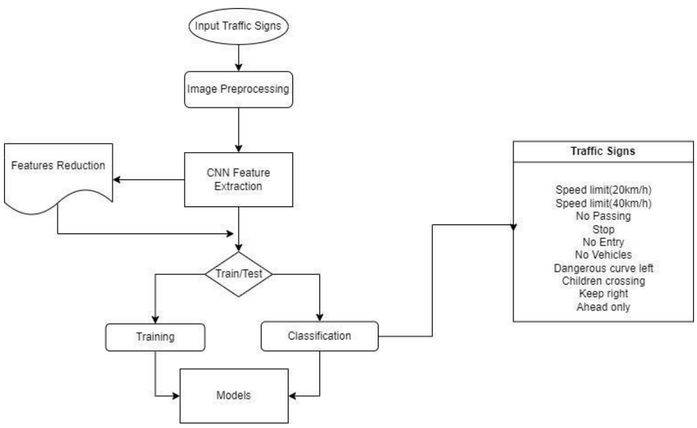

Fig. 4. Model Design

Images of traffic signs captured using high resolution camera is given as input to the system. Preprocessing of the given picture is done which also includes image resizing. Preprocessed im- age is then passed to further layers for Feature extraction using appropriate convolutional neural networks. Features selection techniques are applied using max pooling layers and drop out layers to train the model efficiently.

International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Further the SoftMax function is applied for multiclass classification as it provides probability for different classes so that the highest probable traffic sign is predicted.

DATASET DESCRIPTION

Fig. 5. Different categories of traffic signs

The dataset in consideration is the German Traffic Sign Recognition Benchmark (GTSRB), which was presented at the 2011 International Joint Conference on Neural Networks (IJCNN) [7]. The internal traffic signs were gathered from the actual German road environment, and they have since been a widely used dataset for traffic signs by professionals and academics working in the domains of computer vision, self- driving, and other related fields. The greatest and smallest photos are, respectively, 250 x 250 and 15 x 15, indicating an uneven distribution of image sizes. Dataset comprises of, Total: 51,389 images. Training: 39,209 images [75 percent]. Test: 12,630 images [25 percent]. The are 43 classes of traffic signs in the GTSRB. Which itself includes a wide range of variations [9]. Each form of traffic sign has a catalogue that includes an image of numerous tracks and a CSV file annotated with a class label for each type. As indicated in Figure 5, there are six major categories within the GTSRB: danger, speed limit, prohibitory, required, derestriction, and various special traffic signs. The dataset is more representative of genuine road scenes because the same type of traffic signs have been captured under various resolutions, weather, illumination circumstances, occlusion degrees, tilt levels, and other variables.

IMPLEMENTATION

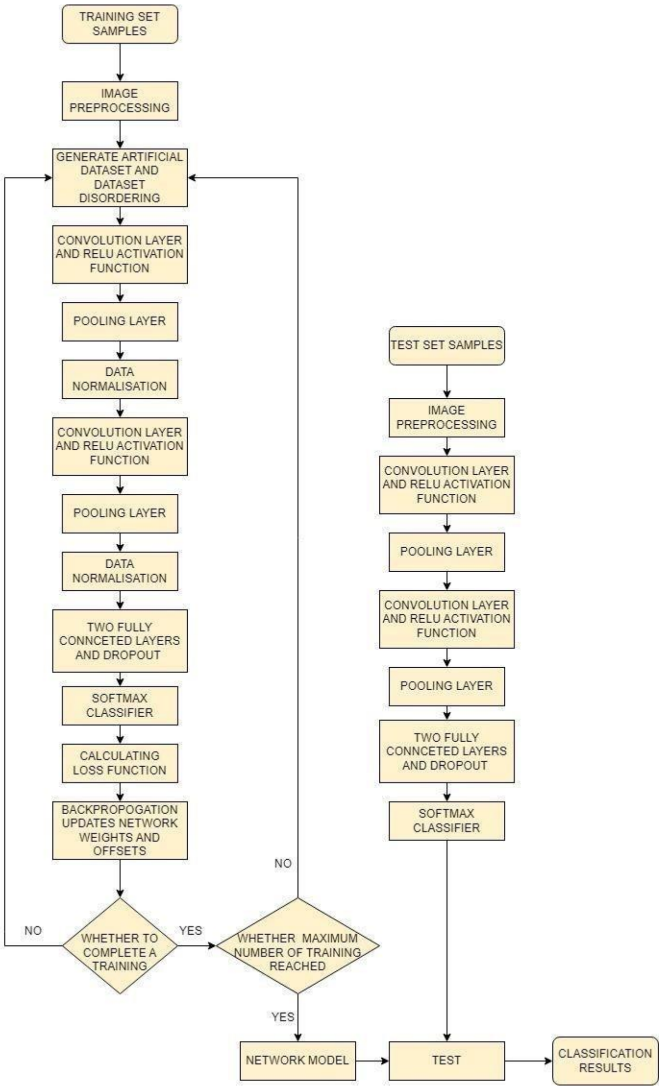

Fig. 6. Flowchart of traffic sign recognition

There are two parts to the traffic sign classification and recognition experiment: network training and testing. The training set samples from the dataset [GTSRB] are used as input in the training network stage. Traffic signs can be classified and predicted by conducting a certain number of network iterations, which continuously update parameters like network weights and offset values based on forward and backward propagation mechanisms [8]. The trained network model is fed testing samples from the dataset during the testing network stage to check the correctness of the recognition rate. The basic steps of network training stage as shown in the above Fig 6:

- The artificial intermediate dataset is being constructed, the training data set samples are being preprocessed, and the dataset order is disordered.
- A number of parameters are established, and the training set samples are transmitted forward in the network model. It employs Adam optimizer [15]. The output of the SoftMax activation function serves as a classifier, and the dropout parameter is set to 0.25 in fully linked layers and 0.5 overall.
- The parameters, such as offset values and network weights, are being updated based on a back-propagation method, and the gradient of loss is determined.

- The difference between the sample's actual value and its estimated value is calculated. The next step is carried out when the obtained error falls below the specified threshold error or exceeds the maximum number of recommended trainings; otherwise, the first step is repeated for the following network iteration (epoch).
- The network model is being used to run the classification test [12]. The dataset's projected traffic sign categories are con- trasted with the actual subordinate groups. The classification prediction results of the traffic signs are counted to determine the right prediction rate.
- The basic steps of the network testing include;
- After preprocessing, a number of photos are fed into the trained network model by randomly selecting samples from the testing set.
- The network model outputs recognition results, thereby displaying the meaning of traffic signs.
- The output findings are compared to the real reference meanings, with statistical analysis providing the outcomes.

RESULTS

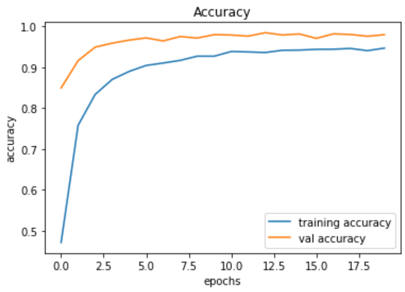

Fig. 7. Accuracy vs Epochs

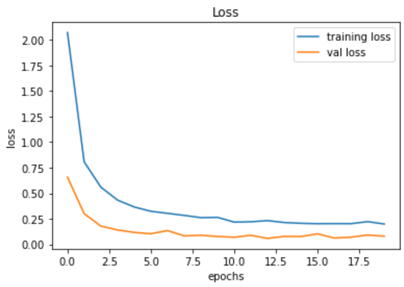

Fig. 8. Loss vs Epochs

With increase in the number of epochs (one epoch means whole forward and backward propagation) both training and validation accuracy increases. Whereas, both training and validation loss decrease with an increase in the number of epochs.

A. Outputs

Predicted traffic sign is: Speed limit (50km/h)

Predicted traffic sign is: Turn right ahead

Predicted traffic sign is: Slippery road

Predicted traffic sign is: Veh > 3.5 tons prohibited

CONLCUSION

For automotive and driver-assistance vehicles, a better traffic sign classification and recognition system is proposed in this study. First, a high-quality webcam image of a traffic sign is provided as input to the model. After preprocessing, the image is then sent to the robust model, which is being trained using the GTSRB dataset in various environmental conditions [14]. The model then accurately predicts the name of the traffic sign with 95 percent accuracy. The network model is regularly tested and trained to produce accurate forecasts and traffic sign recognition. Processing takes 50ms per frame on average. In comparison to previous ways, the suggested methodology offers a better training efficiency. But • High resolution cameras must be used in cars to recognize the traffic signs more accurately. • Efficiency of model might reduce if the traffic sign is distorted more than 35 percent. • Additional driving factors, such as lighting, camera angle, obstacles, and speed, could hinder the performance of our model. Although our suggested methodology offers a number of noteworthy benefits in terms of sign recognition accuracy and processing speed. It excels at enhancing driving safety, operates efficiently in real-world driving situations, and satisfies real-time objec- tive criteria. Future work can further refine and optimize the ’Automated Traffic Sign Recognition’ system’s performance to improve the model’s overall functionality.

REFERENCES

1. Suriya Prakash, D. Vigneshwaran, R. Seenivasaga Ayyalu and S. Jayanthi Sree, ”Traffic Sign Recognition us- ing Deeplearning for Autonomous Driverless Vehicles,” 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), 2021.
2. L. Jia, X. Shi and L. Wei, ”Design of Traffic Sign Detection and Recognition Algorithm Based on Template Matching,” 2020 IEEE 2nd International Conference on Civil Aviation Safety and

Information Technology (ICCASIT, 2020.
3. Z. He, Z. Xiao and Z. Yan, ”Traffic Sign Recognition Based on Convolutional Neural Network Model,” 2020 Chi- nese Automation Congress (CAC), 2020.
4. D. Tabernik and D. Skocˇaj, ”Deep Learning for Large- Scale Traffic-Sign Detection and Recognition,” in IEEE Trans- actions on Intelligent Transportation Systems, vol. 21, no. 4.
5. R. Ravindran, M. J. Santora, M. Faied and M. Fanaei, ”Traffic Sign Identification Using Deep Learning,” 2019 International Conference on Computational Science and Computational Intelligence (CSCI), 2019.
6. S. Mehta, C. Paunwala and B. Vaidya, ”CNN based Traffic Sign Classification using Adam Optimizer,” 2019 International Conference on Intelligent Computing and Control Systems (ICCS).
7. A. S. Alturki, ”Traffic Sign Detection and Recognition Using Adaptive Threshold Segmentation with Fuzzy Neural Network Classification,” 2018 International Symposium on Networks, Computers and Communications (ISNCC), 2018.
8. R. Biswas, H. Fleyeh and M. Mostakim, ”Detec- tion and classification of speed limit traffic signs,” 2014 World Congress on Computer Applications and Informa- tion Systems (WCCAIS).
9. S. Santaji, S. Santaji and S. Hallur, ”Detection and Classification of Occluded Traffic Sign Boards,” 2018 Interna- tional Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT), 2018.
10. Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li and S. Hu, ”Traffic-Sign Detection and Classification in the Wild,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
11. C. G. Kiran, L. V. Prabhu, V. Abdu Rahiman, K. Rajeev and A. Sreekumar, ”support vector machine learning based traffic sign detection and shape classification using Distance to Borders and Distance from Center features,” TENCON 2008 - 2008 IEEE Region 10 Conference, 2008.
12. Y. Sun et al., ”Research on Video-based Traffic Sign Recognition,” 2017 IEEE 7th Annual International Confer- ence on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), 2017.
13. X. Bangquan and W. Xiao Xiong, ”Real-Time Embed- ded Traffic Sign Recognition Using Efficient Convolutional Neural Network,” in IEEE Access, vol. 7.
14. B. Smitha Shekar and G. Harish, ”A Ma- chine Learning Model For Detection and Recognition Of Traffic Signs,” 2021 International Conference on In- telligent Technologies (CONIT), 2021.
15. A. Wong, M. J. Shafiee and M. St. Jules, ”Micron- Net: A Highly Compact Deep Convolutional Neural Network Architecture for Real-Time Embedded Traffic Sign Classifi- cation,” in IEEE Access, vol. 6, pp. 59803-59810, 2018.
16. M. Y Radzak, Illanggo Muniandy, “Study on Traffic Sign Recognition”, International Journal of Research Studies in Computer Science and Engineering (IJRSCSE) Volume 2, Issue 6, June 2015.
17. Ayoub Ellahyani, Ilyas El Jaafari, Said Charfi, “Traf- fic Sign Detection for Intelligent Transportation Systems: A Survey” in E3S Web of Conferences 229, 01006 (2021).
18. Mrinal Haloi, “Traffic Sign Classification Using Deep Inception Based Convolutional Networks” in IIT Guwahati, 2019.