Proceedings of the Sixth International Conference on Inventive Computation Technologies [ICICT 2021]
IEEE Xplore Part Number: CFP21F70-ART; ISBN: 978-1-7281-8501-9

Car Traffic Sign Recognizer Using Convolutional Neural Network CNN

Abhay Lodhi
Computer Science Department
Delhi Technological University
Delhi, India
abhaylodhi3@gmail.com

Sagar Singhal
Electronics and Communication Department
Delhi Technological University
Delhi, India
sgrsinghal16@gmail.com

Massoud Massoudi
Software Engineering Department
Assistant Professor, Delhi Technological University
Delhi, India
massoud.massoudi@hotmail.com

Abstract—Acknowledgment of traffic signs vary significantly in numerous applications, for example, in self-driving vehicle/driverless vehicle, traffic planning and traffic observation. Traffic Sign Recognition (TSR) framework is a segment of Driving Assistance System (ADAS). The TSR framework helps the drivers in safe driving as street signs give significant data of the street. The car business has built up a great deal and a portion of the organizations are attempting to assemble self-sufficient vehicles and in which traffic sign acknowledgment is one of the significant factors to be thought of. To perceive the traffic signs, a model utilizing convolutional neural network is fabricated and this model will perceive the traffic signs. This model can likewise be utilized in typical vehicles to caution the driver about traffic signs through content identification.

Keywords—Convolution neural network, Adam optimizer, Traffic Sign.

I. INTRODUCTION

For the most part, traffic sign acknowledgment (TSR) frameworks comprise two phases of recognition and arrangement; for some TSR frameworks, a following stage is planned among detection and Classification for managing video successions[1].Deep learning emerges to be the most successful subset of machine learning in solving problems related to image classification and identification. The achievement of deep learning in the field of self-driving cars is unequivocal. For example, DNNs have been utilized in scene semantic division [2], traffic signal recognition [3], crosswalk arrangement [4], [5], traffic sign location [6], walker investigation [7], vehicle heading course assessment [8] and numerous different applications.

The proposed research work focuses around traffic sign detection. A traffic in alone can convey a whole lot of information to the street clients. It is significantly important to develop a reliable automatic traffic sign detection system. Apart from the advantages of Deep learning there are few concerns related to it as .Deep learning requires expensive annotation, a large balanced data set which in result requires high computational cost in training.

To achieve this task, a dataset that contains almost all the information about traffic is required as there are some traffic signs, which are rarely seen and uncommon under normal driving scenarios.

There are basically three techniques to identify traffic signs: (i) 'AdaBoost based detection', (ii) 'Support Vector Machine (SVM), and (iii) 'Neural Network (NN) based detection'.

German Traffic Sign Detection Benchmark (GTSDB) [9] is used as these images are taken from real scenarios of German street.

II. CONVOLUTION NEURAL NETWORK

A CNN is a supervised learning procedure which needs both information and target yield information to be provided. These are grouped by utilizing their marks to give a scholarly model to future information investigation.

Normally a CNN has three principles constitutes- a convolutional Layer, a Pooling Layer and a fully associated Dense Network. The Convolutional layer takes the info picture and applies m number of nxn channels to get a component map. The element map is next taken into the maximum pool layer which is basically utilized for dimensionality decrease, it picks simply the best highlights from the component map. At last, all the highlights are smoothed and sent as contribution to the completely associated thick neural organization which learns the loads utilizing backpropagation and gives the characterization yield.

The inspiration driving the CNN is that it depends in transit the visual cortex capacities, where one item in the scene is in center while the rest is obscured, comparably the CNN takes one segment/window of the info picture at once. Each time the CNN will deliver a component map for each part, in the convolutional layer. In the Pooling layer it eliminates the overabundance highlights and takes just the main highlights for that part, consequently performing highlight extraction. Subsequently, with the utilization of CNNs and don't need to play out an extra component extraction procedure.

CNNs require lesser pre-preparing when contrasted with other comparative characterization calculations. While customary MLP (Multi Layer Perceptron) calculations have critical exactness for picture

2021 6th International Conference on Inventive Computation Technologies (ICICT) | 978-1-7281-8501-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICICT50816.2021.9358594

acknowledgment, they experience the ill effects of the scourge of dimensionality because of the hubs being completely associated, and subsequently can't be scaled to high goal pictures. CNNs beat these difficulties presented by MLP by abusing the spatial relationship of a picture. This is finished by implementing an example of neighborhood network between nearby neuron layers. Subsequently, CNNs end up being unrivaled at Image order, Video Analysis, Natural Language Processing and wide scope of different applications when contrasted with different methods.

III. RELATED WORK

In earlier times traffic sign detection used only shapes and colors and it was not that efficient and robust. Due to increase in computational prowess of mankind and advent of large data sets, deep convolutional datasets have demonstrated their capabilities that has led to increase in accuracy of traffic sign recognition. Traffic sign recognition mainly includes detection and recognition. Yuga hatolkar et al [10] proposed a method for traffic sign recognition in which the fuzzy classification module acts as an optimizer module for results obtained by CNN. It measured accuracy and Loss. In the last decade Convolutional Neural Networks (CNN) have shown remarkable progress in increasing the accuracy and reliability of the traffic detection system. Wu et al. [11] began employing CNNs to address traffic sign detection as a candidate region classifier. Zhu et al. [12] proposed a novel framework in which a fully convolutional network generated the region CNN for Small Traffic Sign Recognition proposals and CNNs was used for classification. Jain A et al [13] optimized detection using CNN they applied convolutional neural network domain transfer learning and genetic algorithm to calculate accuracy, precision, recall percentage. Zhang J [14] used CNN and knowledge distillation to calculate accuracy however pruning a model to 70% decreased its accuracy.S.Mehta [15] used CNN softmax activation function, RELU activation function and Adam optimizer to test and train accuracy and loss. Jin et al [16] proposed Traffic Sign Recognition implementing Hinge Loss method in Neural Networks that achieved a 99.65% accuracy on the GTSRB dataset. In recent times a new technique R-CNN is being proposed by the scholars. RBG [17] proposed an R-CNN framework. Recent developments in deep learning have increased recognition and detection capabilities of the systems using different layers in the convolutional networks. Inspired by these works we aim to develop an effective traffic sign detection framework following the similar philosophy of using different features of layers of CNN.

IV. METHODOLOGY

1) Dataset

In this paper German traffic sign dataset (GTSDB)[9] is utilized as an information base which has been widely used in the majority of the research work for ordering and identifying traffic signs.31367 samples of images are used for training purpose, 7842 for validations and 12630 for testing .The dataset contains 43 classes i.e. ,{ 1:'Speed limit (20km/h)', 2:'Speed limit (30km/h)', 3:'Speed limit (50km/h)', 4:'Speed limit (60km/h)', 5:'Speed limit (70km/h)', 6:'Speed limit (80km/h)', 7:'End of speed limit (80km/h)', 8:'Speed limit (100km/h)', 9:'Speed limit (120km/h)', 10:'No passing', 11:'No passing veh over 3.5 tons', 12:'Right-of-way at intersection', 13:'Priority road', 14:'Yield', 15:'Stop', 16:'No vehicles', 17:'Veh > 3.5 tons prohibited', 18:'No entry', 19:'General caution', 20:'Dangerous curve left', 21:'Dangerous curve right', 22:'Double curve', 23:'Bumpy road',24:'Slippery road', 25:'Road narrows on the right', 26:'Road work', 27:'Traffic signals', 28:'Pedestrians', 29:'Children crossing', 30:'Bicycles crossing', 31:'Beware of ice/snow', 32:'Wild animals crossing', 33:'End speed + passing limits', 34:'Turn right ahead', 35:'Turn left ahead', 36:'Ahead only', 37:'Go straight or right', 38:'Go straight or left', 39:'Keep right', 40:'Keep left', 41:'Roundabout mandatory', 42:'End of no passing', 43:'End no passing veh > 3.5 tons' }.

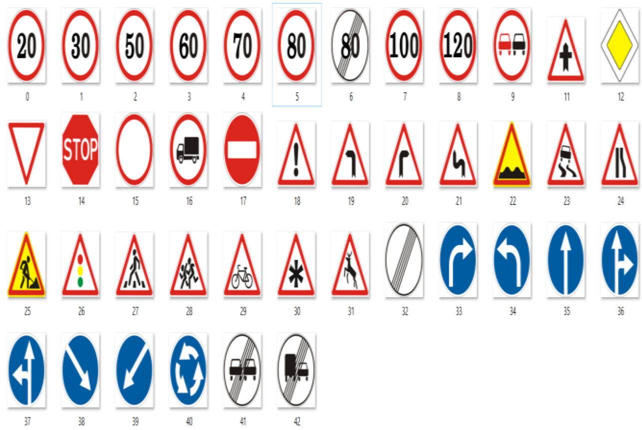

Figure 1: Preview of German Dataset

Below are the three histograms representing the class wise samples of images taken for (i) training, (ii) validation and (iii) testing. We had taken 80% of the dataset for training purpose and 20% for testing.

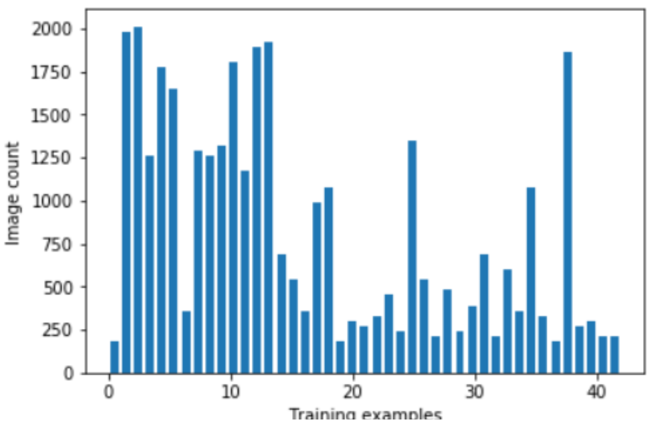

Figure 2: Class wise visualization of samples taken for Training

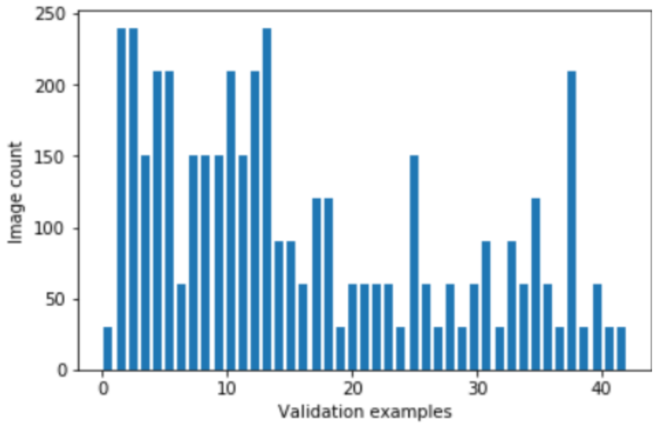

Figure 3: Class wise visualization of samples taken for Validation

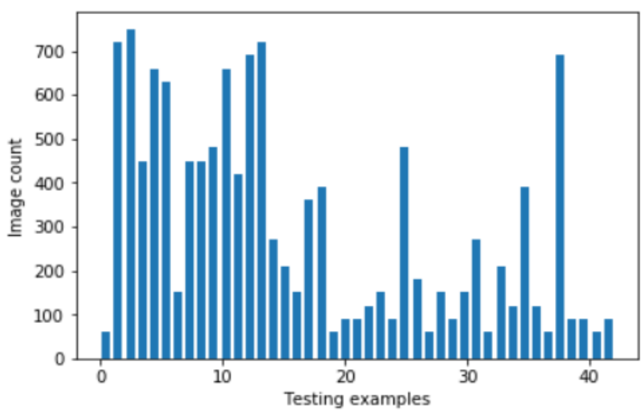

Figure 4: Class wise visualization of samples taken for Training

2. *Experimental Setup*

Proposed work is implemented with the help of tensorflow and keras library. The system is backed with Intel core i7 7th Gen, Nvidia Geforce GTX 1050Ti, 8 GB ram and 64 bit operating system. We had used pillow library for opening, manipulating and saving images and had used OS module for iterating the dataset because of its redundancy, portability, safety. It provides faster execution of images with respect to other module.

3. *CNN Architecture*

Figure 6, showed the proposed CNN architecture which we had developed for classification and recognition of German Traffic Sign. The proper number of filter size is subjected to experimental tries. The number of filter size must be chosen considering calculation costs. The first three layers of CNN network is constructed with 32 filters and a kernel size of (5x5).Kernel is slided over the input dataset performing the dot product with the sub area of input images and gives matrix representation of the dot product as a output. ReLu Activation function is used in each layer. The output of the convolutional layer was made to go through the ReLU initiation function each time. This function basically chooses the output that must be shipped to the neurons of the next layer. This function gives zero as an output for all the values less than zero. Graphical representation of ReLU function is given below.

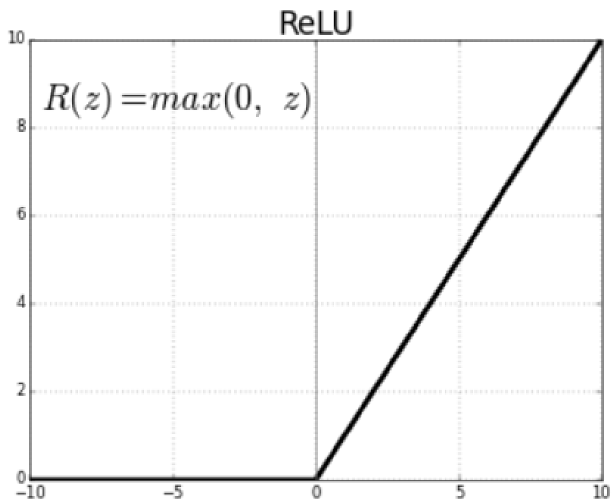

Figure 5: Graphical representation of ReLU Function

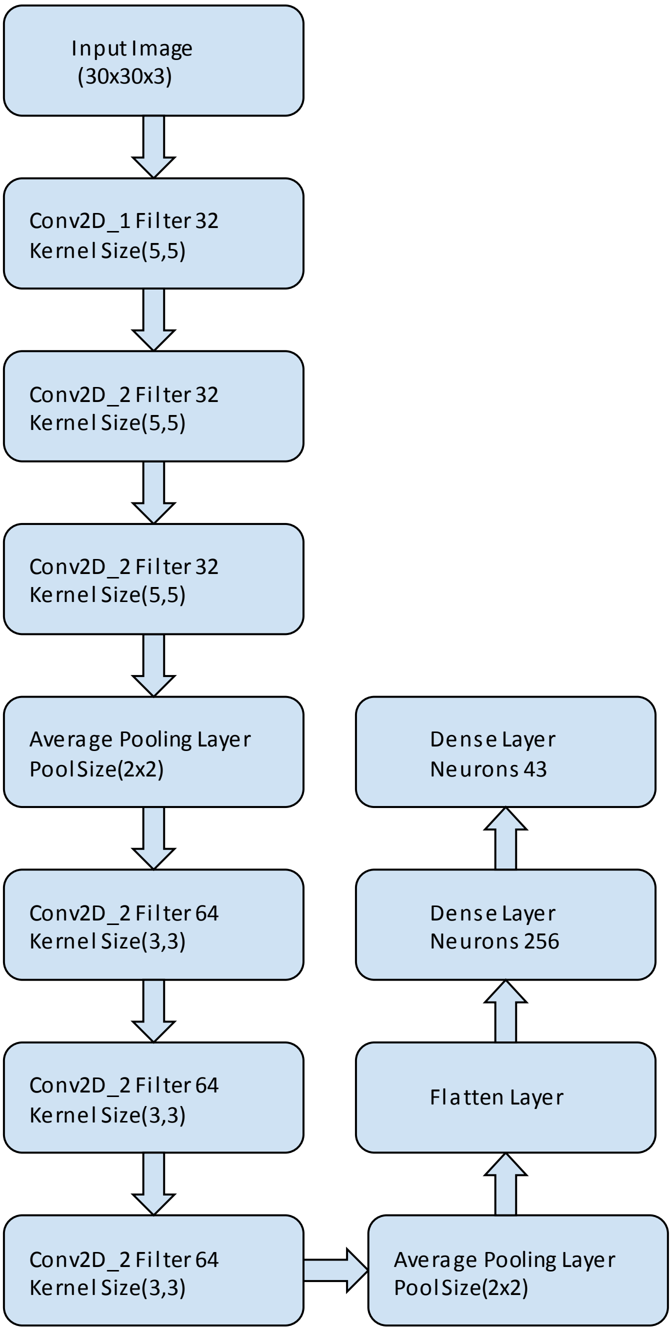

Figure 6: Flow Diagram Of CNN Architecture.

To choose the correct pooling layer is next big step to build a successful neural network, as it help us in reducing the spatial size and reduce the complexity of the input images[18],deducting some parameters to reduce the computational complexity of the input images. The sizes of images are controlled by three parameters (i) how many filters we have used, (ii) stride, (iii) zero padding. The proposed model is tested with average pooling layer and max pooling layer results were prominent from both side. The average pooling layer help us in bringing the smoothness in image, so finally we build our model with average pooling layer.

We first thought of using dropout layer, but after experimental result we were motivated not use dropout layer as dataset we had used was less prone to overfitting and we were getting better training and validation accuracy by dropping dropout layer. Training accuracy which was around 96% jumps to touching 100% (99.06) and there was also slight increase in the validation accuracy but the final testing accuracy were more and less similar. Table I shows the experimental result of our model with and without dropout layer.

Table I shows the comparison of the model with and without dropout layer.

With And Without Dropout layer	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss
With Dropout layer	0.9602	0.1462	0.988	0.0477
Without Dropout Layer	0.9906	0.0366	0.987	0.0637

Output from the dropout layer is sent as an input to the flatten layer which is useful in fast feeding the forward execution,reducing parameters especially weights of neural network is taken into consideration at high exent.

After passing images through linear layers (y=wx+c) , we passed our model to the nonlinear layer(Dense layer) to modulate the nonlinear properties of our model and we finally aggregate this model with Adam optimizer which performs well in categorical cross entropy as we had various classes of traffic signs to classify.To achieve high accuracy it is important to choose correct optimizer with right parameter.

V. EXPERIMENTAL RESULT

We first preprocess the data set to improve the training and testing results.we resize them to 30x30x3 format.After that we observe that data augmentation done improves the accuracy.

Table II shows the accuracy of the model with different optimizers.

Optimizer	Training Accuracy	Training Loss	Validation Accuracy	Validation Loss
SGD	0.020	0.022	0.022	0.022
AdaGra	0.038	0.024	0.061	0.022

d				
Adam	0.958	0.150	0.986	0.055

We tested our model with different optimizer results depicted in table I.Model was performing poorly with SGD(Stochastic Gradient Descent), AdaGrad as these optimizers were not able to learn weights adaptively.Most prominent results were with Adam optimizer.

Taking into account this comparison Table I we finally decided to use Adam optimizer as its training cost was much lower than others optimizer and we were able to achieve higher accuracy.There were several pros for using this optimizer , as it not only reduce the computational cost but was also faster than the other optimizer [19].

To choose the correct pooling layer is an important decision to build a better CNN architecture , we were getting slightly better results with the average pooling layer,so we finally decided to use it.We finally used 3 convolution layer with ReLU as an activation function in each layer,followed by average pooling layer , repeating this pattern one more time and finally passing it to flatten layer followed by two dense layers.

Table III shows the performance measure of the model with different pooling layers.

Performance Measure	Score (Max_Pooling_layer)	Score (Average_Pooling_layer)
Training Accuracy	95.87%	99.06%
Validation Accuracy	98.62%	98.71%
Testing Accuracy	96.00%	96.19%
F1 Score	0.9600	0.9619
Precision Score	0.9487	0.9619
Cohen Kappa Score	0.9584	0.9605
Training Loss	0.1501	0.1462
Validation Loss	0.0552	0.0477

This paper presents a good approach to recognise traffic signs , on training the model on 15 epochs with a batch size of 62 we were able to get an accuracy of 99.06% on training model.The accuracy on validation was about 98.71% and the final testing accuracy was 96.19%. The accuracy and loss on training and validation dataset are shown in fig 6 and fig 7.

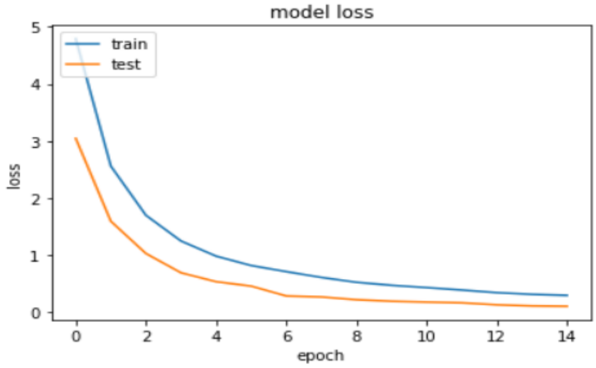

Figure 6: Model loss vs epoch

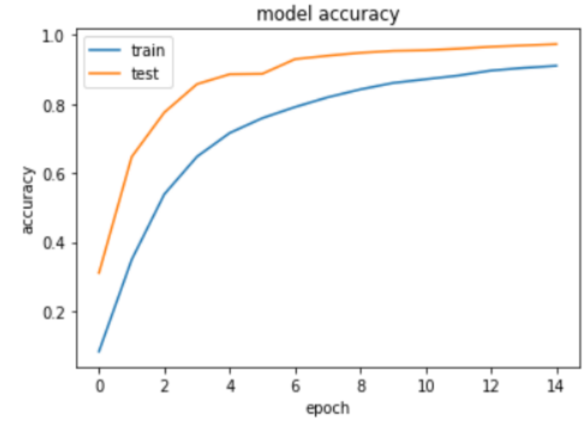

Fig 7. Accuracy vs epoch

VI. CONCLUSION

In this paper we proposed a Traffic Sign recognition system using Convolutional Neural Networks(CNN) with State of art results having 96.19% accuracy on the German dataset(GTSDB) used. The CNN framework integrates multiple levels of convolution feature and multiple levels of contextual information. At the detection stage, the region proposals are generated from the fused feature map with sufficient information. ADAM optimizer was used to decrease the computational and training cost which helped in achieving the given accuracy.

In the future we would like to increase the accuracy and efficiency of our project by visualizing remaining errors and using further optimization techniques.

REFERENCES

[1] C. Liu, S. Li, F. Chang and Y. Wang, "Machine Vision Based Traffic Sign Detection Methods: Review, Analyses and Perspectives," in *IEEE Access*, vol. 7, pp. 86578-86596, 2019, doi: 10.1109/ACCESS.2019.2924947.

[2] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff and H. Adam, "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation", *European Conference on Computer Vision (ECCV)*, 2018.

[3] K. Behrendt, L. Novak and R. Botros, "A deep learning approach to traffic lights: Detection tracking and classification",

International Conference on Robotics and Automation (ICRA), pp. 1370-1377, 2017.

[4] R. F. Berriel, F. S. Rossi, A. F. de Souza and T. Oliveira-Santos, "Automatic Large-Scale Data Acquisition via Crowdsourcing for Crosswalk Classification: A Deep Learning Approach", *Computers & Graphics*, vol. 68, pp. 32-42, 2017

[5] R. F. Berriel, A. T. Lopes, A. F. de Souza and T. Oliveira-Santos, "Deep Learning-Based Large-Scale Automatic Satellite Crosswalk Classification", *Geoscience and Remote Sensing Letters*, vol. 14, no. 9, pp. 1513-1517, 2017.

[6] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li and S. Hu, "Traffic-Sign Detection and Classification in the Wild", *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016

[7] R. Guidolini, L. G. Scart, L. F. Jesus, V. B. Cardoso, C. Badue and T. Oliveira-Santos, "Handling Pedestrians in Crosswalks Using Deep Neural Networks in the IARA Autonomous Car", *International Joint Conference on Neural Networks (IJCNN)*, 2018.

[8] R. F. Berriel, L. T. Torres, V. B. Cardoso, R. Guidolini, C. Badue, A. F. D. Souza, et al., "Heading direction estimation using deep learning with automatic large-scale data acquisition", *International Joint Conference on Neural Networks IJCNN*, 2018.

[9] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing and C. Igel, "Detection of Traffic Signs in Real-World Images: The German Traffic Sign Detection Benchmark", *International Joint Conference on Neural Networks (IJCNN)*, 2013.

[10] Hatolkar, Yuga & Agrawal, Poorva & Patil, Seema. (2018). A Survey on Road Traffic Sign Recognition System using Convolution Neural Network. International Journal of Current Engineering and Technology. 8. 10.14741/ijcet/v.8.1.21.

[11] Y. Wu, Y. Liu, J. Li, H. Liu and X. Hu, "Traffic sign detection based on convolutional neural networks," The 2013 International Joint Conference on Neural Networks (IJCNN), Dallas, TX, 2013, pp. 1-7, doi: 10.1109/IJCNN.2013.6706811.

[12] Y zhu,C zhang "Traffic sign detection and recognition using fully convolutional network".

[13] Jain, A., Mishra, A., Shukla, A. et al. A Novel Genetically Optimized Convolutional Neural Network for Traffic Sign Recognition: A New Benchmark on Belgium and Chinese Traffic Sign Datasets. Neural Process Lett 50, 3019–3043 (2019).

[14] Zhang, J., Wang, W., Lu, C. et al. Lightweight deep network for traffic sign classification. Ann. Telecommun. 75, 369–379 (2020).

[15] 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), Madurai, India, 2019, pp. 1293-1298, doi: 10.1109/ICCS45141.2019.9065537.

[16] J. Jin, K. Fu and C. Zhang, "Traffic Sign Recognition With Hinge Loss Trained Convolutional Neural Networks," in IEEE Transactions on Intelligent Transportation Systems, vol. 15, no. 5, pp. 1991-2000, Oct. 2014, doi: 10.1109/TITS.2014.2308281.

[17] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Region-Based Convolutional Networks for Accurate Object Detection and Segmentation," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 1, pp. 142-158, 1 Jan. 2016, doi: 10.1109/TPAMI.2015.2437384.

[18] C.Martin Navas, F. J. Navas Gonzalez, V. Castillo Lopez, L.Payers Capella, M. Gomez Fernandez , and J. V. Delgado Bernejo , "Impact of breeding for coat and spotting patterns on the population structure and genetic diversity of an islander endangered dog breeds' ', Res. vet. Sci.,vol.131,no.April,pp.117-130,2020,doi:10.1016/j.rvsc.2020.04.003.

[19] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, 2014, pp. 580-587, doi: 10.1109/CVPR.2014.81.