European Economic Letters
ISSN 2323-5233
Vol 14, Issue 2 (2024)
http://eelet.org.uk

Traffic Road Sign Recognition Using Multi-CNN Model for Autonomous Ground Vehicles

Subhranil Das
Assistant Professor, School of Business, Faculty of Business and Leadership
Dr. Vishwanath Karad MIT World Peace University, Pune
Maharashtra, India – 411038

Abstract.
Detecting and recognizing the traffic road sign plays a significant role in autonomous vehicles while processing the details regarding each road signs. This research proposed a novel Multi Convolutional Neural Network (M-CNN) for recognizing different traffic signs. In this model, four different CNN structures are applied for processing various road signs. So, the high-level and low-level features are extracted to classify the signs into a proper class in driver-assisted autonomous vehicles. Here, the developed model has produced outstanding results while classifying different real-time traffic signs. The efficiency of the proposed model has been assessed by a well-known database named German Traffic Sign Recognition Benchmark (GTSRB). Moreover, the simulation outcomes demonstrate that the M-CNN model has certain advantages while detecting various traffic signs. Simulation results indicate that the proposed M-CNN model has a higher recognition rate and reduced processing time when compared to Machine Learning (ML) and Deep Learning (DL) models.

Keywords: Convolutional Neural Networks; GTSRB; Traffic Sign, Tensor Flow

1 Introduction

In general, road signs play a vital part in regularly managing vehicles, but, the loss of information in reading the road signboards could be one of the foremost causes of road accidents. In 2015, a report published by the World Health Organization (WHO), which is stating that more than 1 million people succumbed to death due to road accidents [1]. Therefore, Advanced Driver Assistance- Systems (ADAS) [2] have been introduced to reduce human errors while scanning traffic signs in all environmental conditions, such as foggy, rain, sunny, weak light, high speeding, and destruction in road signs. This system could operate all the given conditions as mentioned above and provide correct assistance to the drivers. The ADAS's work begins with capturing all the different road sign images and placing them in the correct class displayed on the panel within a short period. Detection and classification are the two essential steps involved in the ADAS system's functioning. There are many available options for selecting classification algorithms like Support Vector Machines (SVM) [3, 4], Decision Tree (DT) [5], and Artificial Neural Networks (ANN) [6] that are considered traditional classifiers. Currently, the deep Convolutional Neural Network (CNN) is drawing much more attention from researchers as computationally more efficient [7]. The classification accuracy obtained through LeNet is much simpler in computation since it has been acquired from CNN structure. LeNet consists of a CNN structure that comprises various layers like convolutional and pooling which are followed by a set of layers, which has connected by three fully connected layers at the end of the network [8]. Moreover, the algorithms like Scale Invariant Feature Transform (SIFT) extract features and Speed Up Robust Features (SURF) are perceptive to changes in the illumination intensity. A substantial amount of computation time is required while removing the elements using the above algorithms [9]. Li et al. [10] established three feature-based algorithms, which comprise of pre-processing of the image, extraction of features, and a classifier used for classifying an image. S. Ren et al. [11] introduced the faster R-CNN method, where a target detection framework is proposed. Detecting small objects is very cumbersome while applying the R-CNN method. From 2017 to 2019, new methods are proposed for identifying different traffic signs [12,13,14,15]. The above-mentioned techniques are applied to the available databases of traffic signs, such as GTSRB and GTSDB [16,17].

The objective of this work aims to ameliorate the classification's accuracy and lessen the time for processing road signs. In our work, the GTSRB dataset has been used to validate the proposed technique. GTSRB database contains 43 classes of traffic signs containing 51839 images unequally spread over all the categories. It is challenging to construct classification models for acquiring images in road signs under various situations that are low light and motion-blurred images. Different frameworks of CNN structure are being examined for conveying these obstacles.

In this paper, the Multi-CNN (M-CNN) model has been proposed, slightly deviating from previous CNN architectures. In this model, the low-level and high-level features are considered for obtaining information, which is used to classify the images. Here, the complexity of the architecture is made to detect small parts. While applying the M-CNN model, it has been observed that recognizing different traffic signs have been significantly improved as well as the time complexity has been minimized. In the second part, the simulation results are illustrated using the M-CNN technique and compared with other standard Machine Learning (ML) models such as SVM, Random Forest (RF), K- Nearest Neighbour (KNN), CNN, and ANN. The proposed M-CNN model has been also compared with Deep Learning (DL) models, such as LeNet, ResNet, VGGNet, and RetinaNet in order to prove the model's efficacy.

The structure of paper has been organized into five different sections. Section II describes the various related works in detecting numerous road signs through several methods. Section III mentions the description of the proposed M-CNN model. Section IV shows the experimental analysis about the method. Section V displays the simulation results of the proposed M-CNN model and explains the performance of the developed method by evaluating different models. Section VI discusses concluding remarks.

2 Related Works

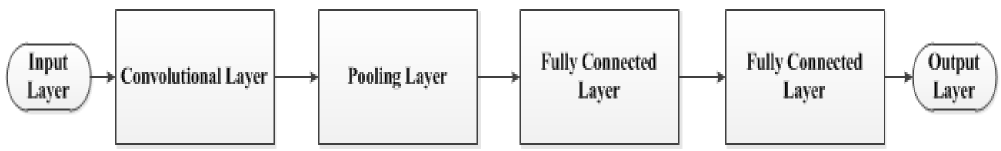
Generally, detection and classification are two significant steps in road signs. While detecting, Region of Interest (ROI) can be constructed using the segmentation of colors [18, 19, 20, 21]. Among the different algorithms of color segmentation, HSV has been considered the more effective for its robustness in the illuminations. Applying HSV helps in extracting ROI through object detection models effortlessly. In order to excerpt the characteristics from the utilized images, fuzzy methods are implemented for processing the images [18]. Gradients are estimated for the object contours formed in the image by applying algorithms like edge detection. The images are transformed into the form of binary, which is the final stage in the process of object detection. For estimating different road signs, ratio constraints are applied to large object filters in which the box is constructed on ROI. While in other works, a selective search algorithm helps to find contour applying HSV color [19]. As a result, redundant contours are discarded on background objects, whereas leftover contours are combined in an ordered manner. Therefore, a method has been selected for testing for all possible color spaces, and the best color space gives the highest detection accuracy. Moreover, road signs are successfully detected in color segmentation on HSV color space [19]. Noises in the image are sustainably getting reduced on the quantization of color. Also, ROI is measured on the basis of shape, which determines the bounding box. Subsequently, shape segmentation [20] plays an essential role after color-based segmentation while applying matched patterns. Binary templates are equated with referencing images for matching patterns.

GPU was studied to obtain ROI where MCT features are extracted [22]. In addition to that, a search algorithm is applied based on various landmarks. This algorithm's work is based on AdaBoost, where many imperfect classifiers amalgamate into a robust classifier. As a result, performance has been gained considerably with changing environmental conditions for detecting road signs. For locating certain road signs, such as speed limit where the Laplacian operator shows promising results on the performance considering distance pixels for two adjoin pixels [23]. Here, the speed limit road sign has been concentrated where red color makes the boundary, and the background is white. For particular regions, color segmentation proves more efficient as color ratios are almost similar [24]. In detecting these regions, the Hessian method is applied, while in recognizing, and the SURF algorithm is executed. A 3D model is constructed to predict the areas around traffic signs on one side of the lane. Using HSV color space, ROI is extracted on external stable regions for adjoining frames.

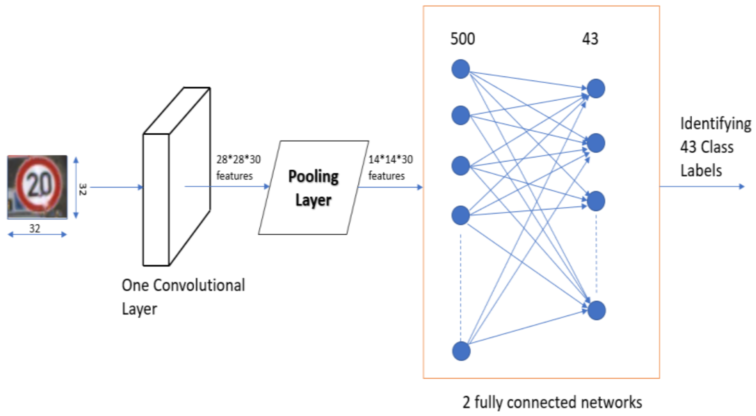
3 Proposed Multi-CNN Model (M-CNN)

The proposed structure of the M-CNN model is the combination of 4 different CNN networks where every network has specific convolutional layers, pooling, and fully connected layers. In all other CNN networks, images have been resized to 32×32 pixels, where different operations have been executed.

In the first structure of M-CNN, only one convolutional layer has been used, but two fully connected layers are applied for extracting the preliminary features from the input image. In the convolutional layer, it consists of 30 kernels where each kernel of size 5×5 pixels. Moreover, the kernel size of 2×2 pixels is kept standard while executing in the pooling layer. Subsequently, features extracted from the pooling layer are being utilized for training the layers that are fully connected. Here, 500 nodes are present in the first connected layer, followed by 43 nodes in the second fully connected layer, known as the output layer used for classifying 43 classes. The Rectified Linear Unit (ReLU) activation function is applied on the convolutional layer and 1st fully connected layer, whereas the SoftMax stimulation function is used at the output of the second fully connected layer of CNN. Here, the learning rate has been considered as 0.001 and Adam Optimizer for the above structure. The structure of the proposed first CNN network of MCNN is displayed in fig. 1.



(a)



(b)

Fig.1:(a) M-CNN Block Diagram Representation (b) Description of Layers in 1st Structure of M-CNN

4 Experimental Analysis

4.1 GTSRB Dataset

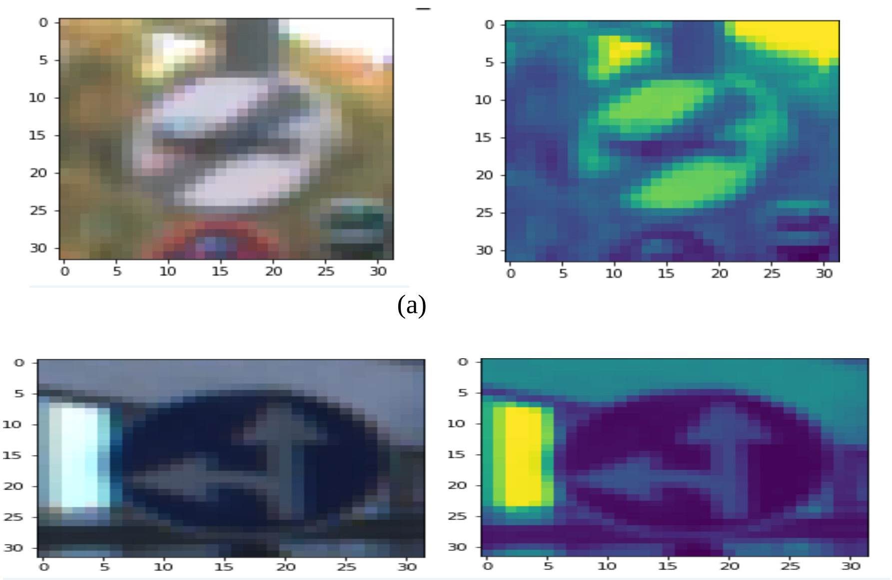
In this paper, the GTSRB database is applied which was created in 2011 by considering one of the standard databases since it contains an enormous amount of about 51,000 images of various conditions such as weak light, dark light, foggy, occlude distributed in 43 different classes. The dataset was collected by taking a video of each of the classes for a duration of 10 hours in various road conditions in Germany. Most researchers used this dataset for their own research on its variability and different sizes of each traffic sign so that verification of the designed model can be tested accordingly. The training set, validation set as well as a testing set are illustrated in given Table 1**.**

4.2 Performance Analysis

The proposed M-CNN model is executed on the recognized database for traffic signs, i.e., GTSRB. A total of 51839 commonly used images of German Traffic Signs where 37999 images have been classified as training images, 12630 images sorted as validation, and 4410 images categorized as testing images. For measuring the validity of the proposed approach, two important measures, such as the accuracy of each structure of M-CNN and time for computation, are essential for making accurate predictions of 43 classes. For computing results on the experimentation, specifications of the computer system are Windows 10, Intel Core i7 Processor with 9th Generation, NVIDIA GEFORCE GTX Graphics Card, and 16 GB RAM with Keas Library [24, 25] imported in Python (Version 3.7.1) [26].

4.2.1 Preprocessing of Images

The most vital parameter of the M-CNN model is the shape of the input that has to be specified. Before training, all the input images are resized into 32×32 pixels to ameliorate computation efficiency for every class in GTSRB, which is given in fig.2.



(a)

(b)

Fig.2: Pre-processing of images of sample input images from given 43 classes

For testing, images are taken uniformly from each class to make an accurate prediction of 43 classes for every structure that is mentioned in the M-CNN model. Here, images acquired from traffic sign database such as speed limit (30, 50, 80 Km/hr), keep right, priority road, and yield consisting of the maximum number of images where they are trained first in priority order.

4.3 Calculation of Parameters

In the 1st Structure of MCNN, the number of kernels is 30, where each kernel size is 5×5 pixels. Therefore, the total number of weights to update are 30×5×5+ 30 (biases) = 780. The outputs of the pooling layer and the number of weights are calculated as 14×14×780 = 5080. At the output of the 1st fully connected network, the total weights are 5080×500 + 500 bias weights (Since the layer consists of 500 nodes of each weight of 1) = 2904500. For the 2nd fully connected network, the total number of weights are to be updated 500×43 +43 bias weights (Since the output layer consists of 43 nodes of each node weight taken to be 1) =21543. Hence, the total number of weights to update is 780+2904500+ 21543=

European Economic Letters
ISSN 2323-5233
Vol 14, Issue 2 (2024)
http://eelet.org.uk

2962823 for each epoch. Therefore, the total number of epochs executed for ten epochs is validated in 4410 samples, as shown in Fig.3.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 28, 28, 30)        780
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 14, 14, 30)        0
_________________________________________________________________
flatten_1 (Flatten)          (None, 5880)              0
_________________________________________________________________
dense_1 (Dense)              (None, 500)               2940500
_________________________________________________________________
dropout_1 (Dropout)          (None, 500)               0
_________________________________________________________________
dense_2 (Dense)              (None, 43)                21543
=================================================================
Total params: 2,962,823
Trainable params: 2,962,823
Non-trainable params: 0
```

Fig.3: Weight Parameters for 1st Structure of M-CNN model

5 Simulation Results and Discussion

In this paper, the GTSRB data set has been used for recognizing the different traffic signs. Four different architectures of the M-CNN model have been applied in order to get optimized performance on the large dataset for differentiating all 43 classes of GTSRB. For building different architectures of CNN, TensorFlow and Keras libraries have been used. For different structures of the M-CNN model, accuracy for training and evaluation, as well as the loss of training and evaluation graphs, are plotted, which are shown from Fig.14 to Fig.18. Hence, the combination of the attained accuracy has been plotted, displayed in Fig. 18.

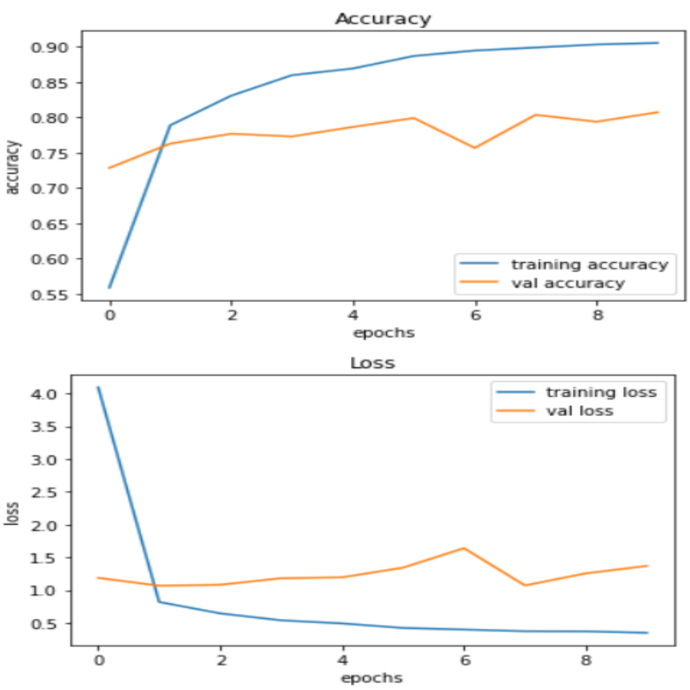


Fig.14: (a) Accuracy (b) Loss for 1st Structure of M-CNN

5.1 Performance Evaluation

Road sign detection for advanced driving systems participates in controlling the vehicle speed, direction, and orientation, mainly in dynamic and complex environments. Hence, the extensive understanding of these types of highly changing oriented environments for predicting road signs from a certain distance, and then executing accordingly to ensure safety for the vehicle. For evaluating the performance of the proposed M-CNN model, a numerical simulation has been implemented to avoid the collision of vehicles, including all 43 classes of road signs demonstrated in the GTSRB dataset. The accuracy percentage of each structure of M-CNN is described in Table 2.

Table2: Summary of Accuracy and loss percentages used in all Structures of M-CNN

S. No.	Structure of M-CNN	Accuracy		Loss	
		Validation	Training	Validation	Training
1	Structure-I	0.80	0.91	1.48	0.49
2	Structure-II	0.88	0.93	0.92	0.47
3	Structure-III	0.92	0.96	0.98	0.31
4	Structure-IV	0.965	0.961	0.225	0.220

Training images consist of 70%, whereas 10% are considered validation, and the remaining percentage of images are put into a testing stage from the GTSRB dataset consisting of 51839 images.

5.2 Advantages of Proposed M-CNN with other methods

The classical methods such as KNN, RF, and CART are simple and more comfortable to implement in machine learning algorithms [27,28,29,30,31]. Moreover, KNN has been considered the ideal algorithm that takes extra time to neglect data or read current data for which distances are being updated for every prediction in training data. Regarding decision trees, it is difficult to measure depth and splitting criteria. Hence, the overfitting problem arises as proper numbers of splits have not taken place. For the RF method, tunning is more costly according to the structure since there is a considerable accumulation of trees. For Deep Learning (DL) algorithms, such as ResNet, LeNet, VGGNet, and Retina Net, where the features are automatically extracted. Here, different structures are proposed for these algorithms in order to get the required classification accuracy. Feature extraction is a separate step before the classifier is applied, mainly in SVM methods. Feature descriptors such as SURF and HOG are applied primarily for SVM for converting features contained in images into a useful form for the input to the classifier. In all methods mentioned above, the tunning problem is a common problem that results in more computation time for selecting the right feature from a large set of features. In M-CNN, the size of filters is changeable after installing programming packages that play certain advantages when extracting the best features in the minimum possible time. For doing the job much more comfortably, CNN is flexible in choosing filters of different sizes to reduce the time and effort for feature extraction. Here, different filter sizes are applied to the input to different convolutional layers for optimizing each feature extraction resulting in higher accuracy in classification for a proper layout of layers in a hierarchical order. Computational time is significantly reduced since each structure of CNN carries its feature descriptor.

Conclusion and Future Scope

In this proposed system, the M-CNN exploits all the architectures of CNN, where different input images are acquired from the GTSRB dataset. With different kinds of noises and occlusions in input images, the impacts of an algorithm are beneficial while classifying. The accuracy obtained from M-CNN is higher than standard CNN. The execution time for running each epoch plays a vital parameter while processing each image. Compared with LeNet, ResNet, VGGNet, and RetinaNet, the time required for processing for every epoch has been reduced significantly amount. Our work has led to higher accuracy in classification than other methods, such as CART, KNN, and RF.

Declarations

Funding

There is no funding required for this project.

Conflict of Interest

The authors declare that they have no conflict of interest.

References

1. Global Status Report on Road Safety 2015 published by World Health Organization(WHO)..
2. Piccioli, Giulia, Enrico De Micheli, Pietro Parodi, and Marco Campani. "Robust method for road sign detection and recognition." Image and Vision Computing 14, no. 3 (1996): 209-223.
3. Maldonado-Bascón, Saturnino, Sergio Lafuente-Arroyo, Pedro Gil-Jimenez, Hilario Gómez-Moreno, and Francisco López-Ferreras. "Road-sign detection and recognition based on support vector machines." IEEE transactions on intelligent transportation systems 8, no. 2 (2007): 264-278.
4. Hechri, Ahmed, and Abdellatif Mtibaa. "Two-stage traffic sign detection and recognition based on SVM and convolutional neural networks." IET Image Processing 14, no. 5 (2019): 939-946.
5. Klęsk, Przemysław. "From Synthetic Images Towards Detectors of Real Objects: A Case Study on Road Sign Detection." In International Conference on Artificial Intelligence and Soft Computing, pp. 14-26. Springer, Cham, 2019.
6. Islam, Kh Tohidul, and Ram Gopal Raj. "Real-time (vision-based) road sign recognition using an artificial neural network." Sensors 17, no. 4 (2017): 853.
7. Shustanov, Alexander, and Pavel Yakimov. "CNN design for real-time traffic sign recognition." Procedia engineering 201 (2017): 718-725.
8. S.Das, CNN architectures: LeNet, AlexNet, VGG, GoogLeNet, ResNet, <https://medium.com/analytics-vidhya/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5>,2017.
9. Abedin, Zainal, Prashengit Dhar, Muhammad Kamal Hossenand, and Kaushik Deb. "Traffic sign detection and recognition using fuzzy segmentation approach and artificial neural network classifier respectively." In 2017 International Conference on Electrical, Computer and Communication Engineering (ECCE), pp. 518-523. IEEE, 2017.
10. Li, Chen, and Cheng Yang. "The research on traffic sign recognition based on deep learning." In 2016 16th International Symposium on Communications and Information Technologies (ISCIT), pp. 156-161. IEEE, 2016.
11. Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. "Faster r-cnn: Towards real-time object detection with region proposal networks In IEEE Transactions on Pattern Analysis and Machine Intelligence (pp. 1137–1149)." Los Alamitos, CA: IEEE 10 (2017).
12. M. A. B. Mahmoud and P. Guo, ‘‘A novel method for traffic sign recognition based on DCGAN and MLP with PILAE algorithm,’’ IEEE Access, vol. 7, pp. 74602–74611, 2019.
13. A. Gudigar, S. Chokkadi, U. Raghavendra, and U. R. Acharya, ‘‘An efficient traffic sign recognition based on graph embedding features,’’ Neural Comput. Appl., vol. 31, no. 2, pp. 395–407, Jul. 2017.
14. A. Gudigar, S. Chokkadi, U. Raghavendra, and U. R. Acharya, ‘‘Local texture patterns for traffic sign recognition using higher order spectra,’’ Pattern Recognit. Lett., vol. 94, pp. 202–210, Jul. 2017.

15. Sanyal, Banhi, Ramesh Kumar Mohapatra, and Ratnakar Dash. "Traffic Sign Recognition: A Survey." In 2020 International Conference on Artificial Intelligence and Signal Processing (AISP), pp. 1-6. IEEE, 2020.
16. Stallkamp, Johannes, Marc Schlipsing, Jan Salmen, and Christian Igel. "The German traffic sign recognition benchmark: a multi-class classification competition." In The 2011 international joint conference on neural networks, pp. 1453-1460. IEEE, 2011.
17. Wang, Gangyi, Guanghui Ren, Zhilu Wu, Yaqin Zhao, and Lihui Jiang. "A robust, coarse-to-fine traffic sign detection method." In The 2013 international joint conference on neural networks (IJCNN), pp. 1-5. IEEE, 2013.
18. Z. Abedin, P. Dhar, M.K. Hossen and, K. Deb, Traffic sign detection and recognition using fuzzy segmentation approach and artificial neural network classifier respectively, in 2017 International Conference on Electrical, Computer and Communication Engineering (ECCE), 2017, pp. 518523.
19. S. Huang, H. Lin, C. Chang, An in-car camera system for traffic sign detection and recognition, in 2017 Joint 17th World Congress of International Fuzzy Systems Association and 9th International Conference on Soft Computing and Intelligent Systems (IFSA-SCIS), 2017, pp. 16.
20. A. Broggi, P. Cerri, P. Medici, P.P. Porta, G. Ghisio, Real time road signs recognition, in 2007 IEEE Intelligent Vehicles Symposium, 2007, pp. 981986.
21. L. Chen, Q. Li, M. Li, Q. Mao, Traffic sign detection and recognition for intelligent vehicle, in 2011 IEEE Intelligent Vehicles Symposium (IV), 2011, pp. 908913.
22. Wang, Dongdong, Xinwen Hou, Jiawei Xu, Shigang Yue, and Cheng-Lin Liu. "Traffic sign detection using a cascade method with fast feature extraction and saliency test." IEEE Transactions on Intelligent Transportation Systems 18, no. 12 (2017): 3290-3302.
23. Kageyama, Yoichi, Kazuto Suzuki, Chikako Ishizawa, and Takuya Suzuki. "Extraction and Recognition of Speed Limit Signs in Night-scene Videos." Journal of the Institute of Industrial Applications Engineers 6, no. 1 (2018): 29-33.
24. Alam, Altaf, and Zainul Abdin Jaffery. "Indian Traffic Sign Detection and Recognition." International Journal of Intelligent Transportation Systems Research 18, no. 1 (2020): 98-112.
25. Ketkar, Nikhil. "Introduction to keras." In Deep learning with Python, pp. 97-111. Apress, Berkeley, CA, 2017.
26. Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, et al. "Scikit-learn: Machine learning in Python." the Journal of machine Learning research 12 (2011): 2825-2830.
27. Das, Subhranil, Rashmi Kumari, and S. Deepak Kumar. "A review on applications of simultaneous localization and mapping method in autonomous vehicles." Advances in Interdisciplinary Engineering: Select Proceedings of FLAME 2020 (2021): 367-375.
28. Kundu, R., Das, S., & Mishra, S. K. (2022, April). A comprehensive review of single and double stage object detection algorithms in self driving cars. In 2022 IEEE 7th International conference for Convergence in Technology (I2CT) (pp. 1-8). IEEE.
29. Kumari, R., Das, S., & Singh, R. K. (2024). Agglomeration of deep learning networks for classifying binary and multiclass classifications using 3D MRI images for early diagnosis of Alzheimer's disease: a feature-node approach. International Journal of System Assurance Engineering and Management, 15(3), 931-949.
30. Das, S., & Mishra, S. K. (2023). Collision avoidance and path planning for mobile robots based on state estimation approach. Journal of Intelligent & Fuzzy Systems, 44(4), 5991-6002.
31. Gadkar, N., Das, S., Chakraborty, S., & Mishra, S. K. (2022, May). Static obstacle avoidance for rover vehicles using model predictive controller. In 2022 International Conference on IoT and Blockchain Technology (ICIBT) (pp. 1-6). IEEE.