A Deep Convolutional Neural Network for Traffic Sign Classification and Detection with Voice Recognition

K.Prakash [a].

[a] M-Tech student, VNR Vignana Jyothi Institute of Engineering and Technology (Autonomous), Nizampet, Affiliation to JNTUH.

Article History: Received: 10 November 2020; Revised 12 January 2021 Accepted: 27 January 2021; Published online: 5 April 2021

Abstract: Traffic Sign Detection and Classification play an important major role nowadays in our daily life. Anyway, Traffic Signs are present on roads. Even though, often the drivers do mistakes. It's very hard to recognize and detect traffic signs while travelling on roads. Drivers may misinterpret traffic signs, this leads to Accidents and results in damage to the vehicle. To overcome this problem this project introduces a concept named Traffic Sign Detection and Classification with Voice Recognition. This model is built by using CNN to extract the images and classify the traffic signs. Here DCCNN model is built to improve overall accuracy and speed. Here Classification process is also tested with AlexNet, VGG16, VGG19. This system reveals output that recognizes the traffic signs automatically that helps to detect the street condition and alerts driver soon and this enables to build a smart vehicle.

Keywords: DCCNN, VGG16, Deep Learning, Traffic Sign Detection, Automation.

1. Introduction

Now a day's various modes of Transport facilities are very important in our daily life [1]. At the same time safety also plays a very important role. For Safety Purpose in Roads already traffic signs are placed on roads to overcome accidents and to follow the traffic rules [2]. Traffic Signs guide us to be in our safety and they help us to travel in our vehicle in the desired path. Even though the traffic signs are present on roads, the drivers often do mistakes [3]. It's very hard to detect and recognize the traffic signs while driving the vehicle. With the rise in a huge number of road accidents happening every year there has been needing to develop a safety system that helps in contributing to the pedestrian vehicle and also driver's safety [4].

Traffic signs play a very important role and crucial role while driving vehicles. A traffic sign is also a part of road infrastructure [5]. Here is an automated system that helps us to notify and identify and can help a lot. This aims to recognize and detect the traffic sign much faster and enables to build of an autonomous vehicle [6].

2. Background Theory

The traffic signs are placed on the road with different sizes and shape some traffic signs are very blur and small in size, which is covered by snow and heavy rainfall. Even though it's in any condition, Our aim to detect traffic signs [7]. The traffic sign classification is a process that receives the input of the image to detect the traffic sign ie.., U-Turn Ahead, Speed Limit 40, Caution Signs, etc… These signs recommend drivers to get by giving input commands through programming the data in such to classify the traffic sign automatically in vehicles [8]. Traffic Sign Detection is a process that detects the traffic sign and performs the feature extraction process, clustering, and filtering process and then the labeling process and testing process has to be taken place then detected result is to be displayed on the dashboard [9].

The dataset consists of various formats. With respect to size, shape structure. Images preprocessing enables the improvement of images and reduces the distortion effect [10]. Here I also used the Gaussian effect to eliminate the image noise [11]. The dataset processing is to be done to ensure the data is arranged suitably to detect the traffic signs much faster.

3. Challenges of the Study

- Can build a smart autonomous smart vehicle further to improve the safety mechanism for drivers, Pedestrians, and Vehicles.
- To Implement Navigating Mechanism for drivers safety.

4. Objectives Of the Study

- To implement Traffic sign Classification in order to detect the Traffic sign much faster.
- To find Traffic Sign Detection to detect the traffic sign when traffic signal approaches near to us and alerts us and notifies us.
- To find out Traffic sign Recognition in order to detect traffic signals from Real-time Webcams.

5. Methodology

Various Methods are tested to detect which methods will perform the task much faster with more accuracy and to test which method is more efficient.

5.1. *CNN (Convolutional Neural Network)*

Traffic Signs are captured and detected and then pre-processed by CNN. After the Detection of traffic signs then Feature Extraction is taken place. In Feature Extraction, the traffic sign is classified and segmented into multiple pixels [12]. This feature extraction results in the extraction of multiple features of the image. Then the training process is carried on by input image pixels which can differentiate from each other [13]. Clustering is a process that segregates different objects in such a way similar objects are placed into similar groups. The Classification of traffic signs is filtered based upon the size and shape of the image [14]. Filtration is a process to detect the edge of image blur image and a shaded image is detected with the kernel sliding in the extracted pixel of the image [15]. Normally a Convolution Neural Network recognizes this field and learns everything by the basic shapes by evolving the many features in the training process. The CNN learns everything in such a way that this can distinguish from one sign to the other sign. This Max-pooling is a technique that decreases the density and to classify the respective sign. In the detection of traffic Sign Video, the Gaussian technique is introduced to reduce the noise. So Here the Rotating kernel filter is calculated as follows

$$g(a,b) = \sum_{ab=-x}^{x} \sum_{aa=-y}^{y} \rho(dx+dy) f(a+dx, b+dy) \quad \ldots\ldots (1)$$

Where $-x < dx < x$ and $y < dy < y$

Here $g\ (a,b)$ is a filtering image ρ is filter Kernel. By this Equation kernel level of the filter is calculated as above.

Under CNN this model also tested with AlexNet, VGG16, VGG19

VGG16:

Here the input is fed to filters that use convolutional layers where filters are with tiny receptive fields of 3 X 3 size. This uses mostly small size layers. The Very first layer is big. The next layers are mostly very small. It supports only a 1X1 filter. So, it takes more time to compute time more.

VGG19:

This VGG19 contains 16 layers of kernel 3X3 size of each pixel to cover an entire whole image. It is a Good Classification compared to VGG16. This contains activation function RELU that produces less computation. It is implemented with the modification of method VGG16.

AlexNet:

AlexNet can deal with big datasets and achieves high accuracy, The AlexNet is implemented to reduce overfitting problems. To reduce the overfitting model This is implemented with Technique Data Augmentation and Dropout.

5.2. *Dual Channel Convolutional Neural Network (DCCNN)*

In the DCCNN model additionally, one individual channel is present. This includes additional processing of the image. First, the drawbacks present in CNN are as follows. The entire images which are present on roads are not unique. Some will blur, shadow, snow, shaded half of the part of the image will be lost due to some circumstances and some images are very small on roads. This results in very low frequency and unbalanced data. This results in very low accuracy and very poor data insufficiency even. To overcome these issues I introduce DCCNN here [16]. Here in these two channels of input is taken and output is generated by one entity. Two channels are needed to be get trained with datasets. Here the data which is inconsistent and inadequate gets fed with two inputs of data in different channels. This results in an increased detection speed of data. Finally, the output can be traced with the help of the final entity.

The main Contributions of DCCNN is as follows:

1. This involves two channels that solve issues of complicated tracing of the image, Poor resolution of images, etc.
2. This involves two Convolutional Neural Networks; quantities are adjusted automatically between the two different channels until the desired is satisfactory. By this model, inadequate training problem can easily overcome and it can trace very fast compared to CNN.
3. Here the decision is taken by using two channels, the outputs of two channels are fed to the fusion point, and then the decision can be taken at the classification stage.

Here DCCNN Leaning Mechanism is mentioned as follows as shown in Fig 1.

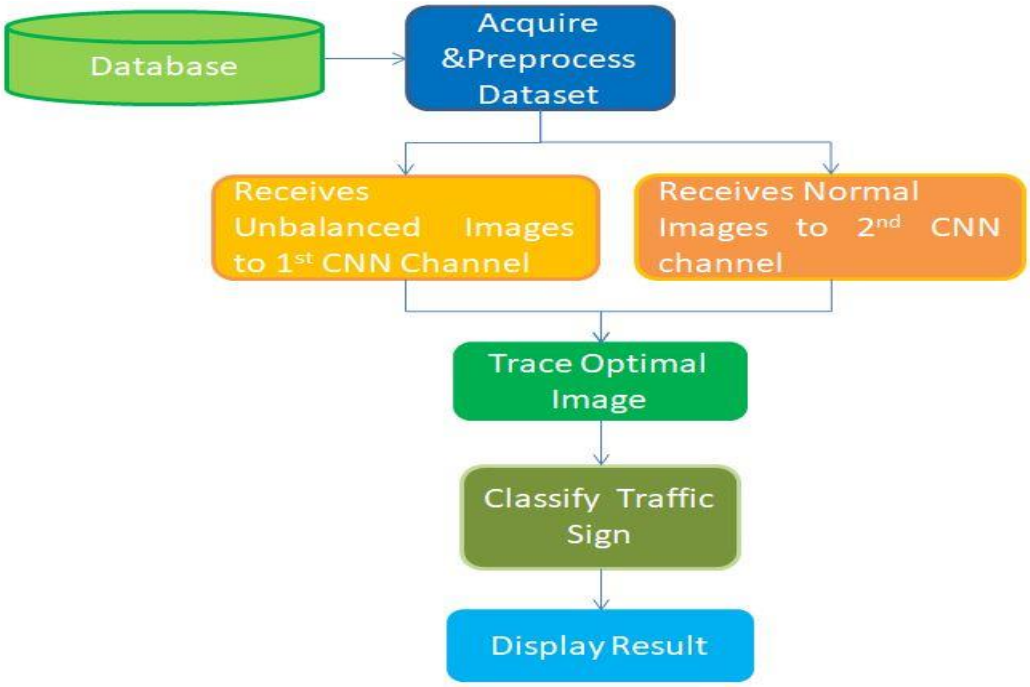

Figure.1: Proposed Learning DCCNN Algorithm

The DCCNN is implemented as follows; it consists of two channels of two networks. Each channel is implemented by its features. Here every channel is implemented by its own 4 fully connected layers.

Here Layers in DCCNN is get trained and implemented as follows:

- Input Layer: 30 x30 input Number of filters are 48
- Hidden Layers
  - Channel 1:Fork11,
  - Channel 2:Fork12, Both Channels performs convolutional on input.
  - Merge: This layer combines the output of channel1 and channel2.
  - Flatten: It converts its input to a one-dimensional array
- Output Layer: Dense Layer uses SoftMax activation function, the shape of output is equal to several classes 43.

Here Dropout is implemented in 50% in the model with fully connected layers of output nodes is 20. By considering Activation SoftMax function is present. This enables the convergence function to maintain stability in all activations.

This dataset consists of low unbalanced data, normalized samples. The unbalanced trained data gets processed separately with the help of dual channels. Here I train every image of every class will be in a cycle until the image will match its suitable respective labels. Here the images are very huge in number. Every Image is segregated to its respective class associated with it. In the Classification process, the dataset consists of various classes of images it consists of 43 classes and each class has up to 400 images overall 8600 images are present in dataset. The dataset of Traffic signs is shown here is as follows in Fig 2.

Figure.1: Dataset of traffic signs

6. Traffic Sign Classification

Traffic Sign Classification is a process that takes place to detect the traffic sign with the help of an image only. The flow diagram of Traffic sign Classification is shown as follows in Fig 3.

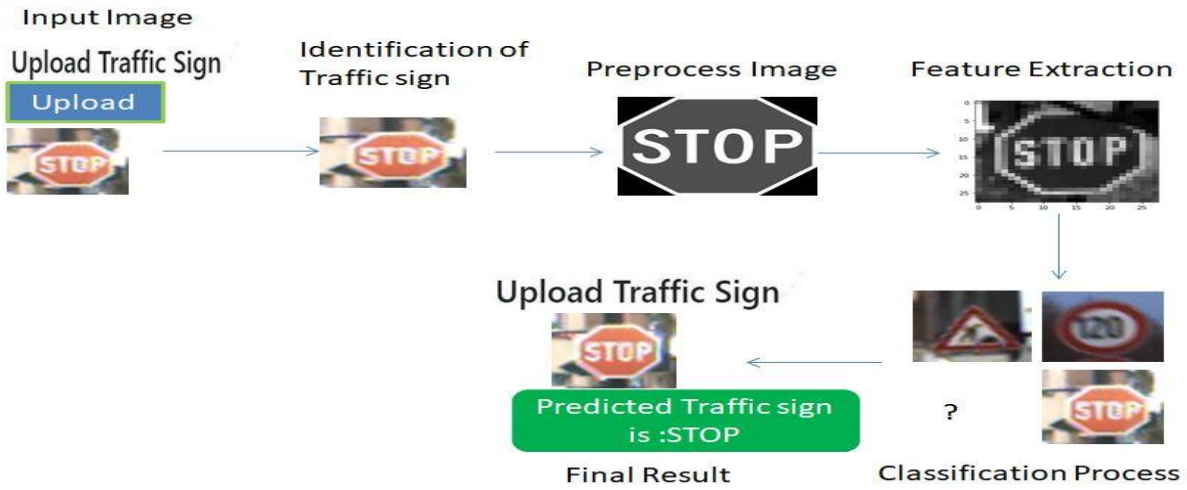

Figure. 2: Flow Diagram of Traffic Sign Classification

The above diagram indicates the proposed model of the Traffic Sign Classification System.

The Procedure of this work is implemented as follows:

- Traffic signs will be in different sizes and shapes. Resizing has to take place to achieve fast detection of traffic signs.
- In this process, the image is captured through the network. Identification of images is taken place with the help of Object Detection API.
- The pre-processing includes tracing the path of the traffic sign and assigning the labels for it. The traffic Sign classes are arranged in the form of an array with the help of its respective data associated with it to retrieve the data easily. This paper also implements a loop to trace the data until the data associated with the respective image is traced.
- Now building the model has been taken place to train by using an Algorithm. Then training and testing process is carried for the training model.
- After the training and testing process is carried on then we can check its accuracy and confusion matrix then test with the desired image so the resultant traffic sign information is obtained.

7. Traffic Sign Detection

Traffic Sign detection is the detection of traffic signs that takes place within the autonomous smart vehicle itself. The detection of traffic sign is taken place as follows as shown in Fig 4.

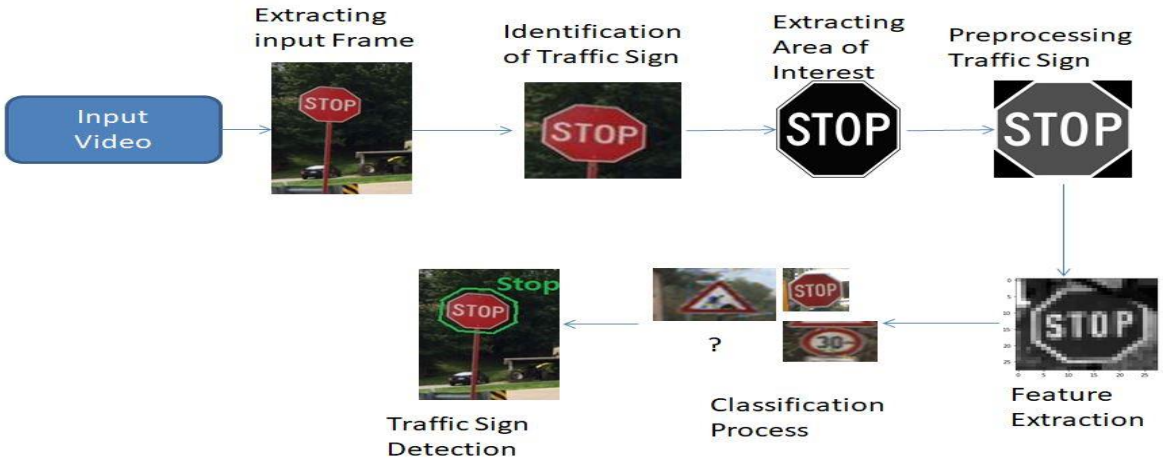

Figure. 3: Flow Diagram of Traffic Sign detection

The smart vehicle can be equipped with the software. The above diagram indicates the proposed work of the Traffic sign detection system.

The procedure of this work is implemented as follows:

- First, we take a sample video as an input here, then the first detection of traffic sign in a video is taken place with the help of object detection Even though it's in any shape and size no matter it detects and this fed to Again traffic sign classification Process
- The detect traffic sign is fed to the Feature Engineering Process that divides the whole image into a huge number of pixels and then extracts the Features and then clustering and Filtration operation process is also performed.
- By the above process, the Image is then extracted and detected and then the image is compared with the testing process which is performed in the traffic sign Classification.
- The label map is done to check with the testing feature to verify the predicted result is correct and after verification, this displays the result.
- The Traffic Sign Detection Process is performed with the help of a dataset that gets trained and tested by the CNN in the Traffic sign Classification Process.

8. Experimentation Results:

The Experiment is done with help of a flask that makes use of even HTML CSS JS along with python etc. The Traffic Sign Classification is proved as best in automatically recognizing the traffic signs and helps in very fast detection of traffic signs. The interface of his project of traffic sign classification is shown as follows in Fig 5.

Figure. 4: Result of Traffic Sign Classification (on left) Input Image (on right) Predicted Traffic sign Image.

The above diagrams indicate the output of Traffic Sign Classification. First Fig 5(on left) indicates the input of the image and another one when I upload traffic this shows the desired output of an image. Traffic Sign Classification is very necessary to identify traffic signs much faster even though the traffic sign is in any size or shape. Even though the traffic sign is a blur and faded then also with this we can detect it very easily as shown in Fig 5(on right). Traffic sign detection is very necessary to take place detection automatically when we are in travelling in a vehicle. This helps to detect the traffic signal before the traffic sign is arrived. This alerts the when the person is in sleepy mode. This also guides the way of road automatically. The following fig of traffic sign detection is shown in Fig 6 as follows. Fig 6(on left) indicates here the input video frame without detection of any traffic sign signal. In Figure 6(on right) The detection of Traffic sign detection has been taken place.

Figure. 5: Result of Traffic Sign Detection (on left) Input video (on right) Traffic Sign detection

Here the output of Traffic sign Classification and Detection is fed to the input of GTTS library in python is used to implement the same result with speech. The Traffic Sign is Recognition is performed from a Real-time Webcam camera with voice software i.e., My Groovy music. This involves the same training method as Traffic sign detection but the procedure is followed with OpenCV python. The Traffic Sign Recognition is performed as follows as shown in Fig 7.

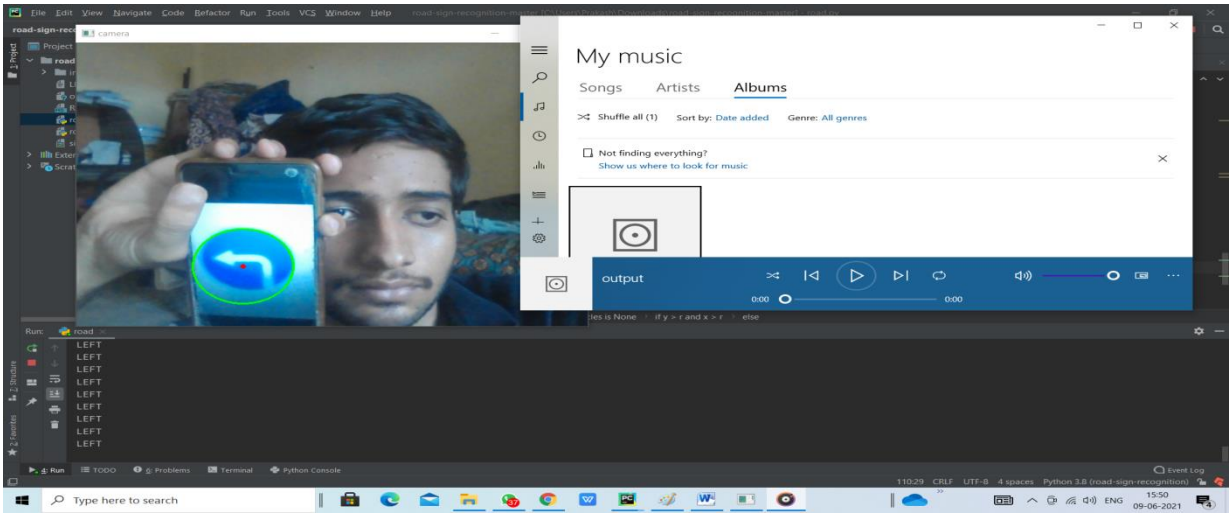

Figure. 6: Traffic Sign Recognition

9. Comparative Results

Here the Model is implemented with VGG16, VGG19, AlexNet. The performance of various algorithms is implemented as follows:

The VGG16 is implemented for the traffic sign classification of images. VGG16 holds 82.77% Accuracy, but this is used to handle a huge amount of datasets. Here the VGG 19 is implemented the Results are somewhat better compared to VGG16. VGG19 holds an accuracy of 88.39%. Here also input images of the dataset are fed to VGG19. To improve this model I use DCCNN (Dual Channel Convolutional Neural Network). Here the DCCNN achieves high accuracy compared to the Normal CNN method. The DCCNN

contains two CNN Channels this so here with the help of DCCNN anyway gains high accuracy compared to CNN.

Table 1. Overall Performance of Algorithms

S.No	Methods	Accuracy	Performance
1	AlexNet	66.53%	Poor
2	VGG16	82.77%	Good
3	VGG19	88.39%	Better
4	DCCNN	94.15%	Very Fast

The accuracy gained by the DCCNN algorithm for this proposed model is 94.15%. So the accuracy is gained by DCCNN is finally 6% higher than the CNN. During the training process, the images which are present with unbalanced data are fed to the DCCNN in order add setup a high training frequency of images and to improve training speed and accuracy. So Here Confusion Matrix for the final DCCNN Algorithm is as shown in Fig 8.

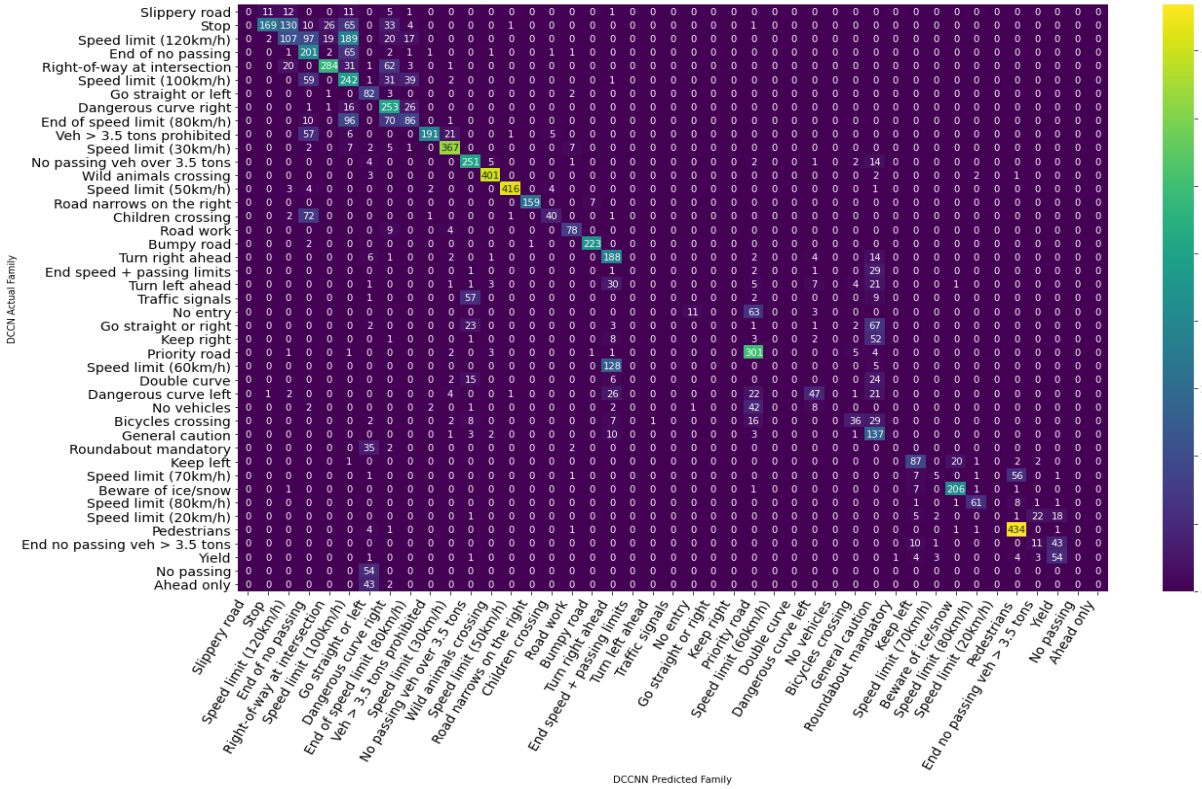

Figure.8: Confusion Matrix of DCCNN

10. Conclusion

The Proposed system is an effective method for performing the traffic sign Classification and detection system. Even though the traffic sign is any condition, whether it is to blur small, or it is small size or improper display, This mechanism helps to detect traffic signs much faster and alerts us. Among all the above algorithms, the DCCNN is only the best technique that can detect the traffic Sign much faster with better accuracy. This project is well suits for quotation "one-stop solution to all traffic issues". Traffic sign detection and classification are very important to have the safety to drivers pedestrians and vehicles. These results can guide the way present on-road and provides information related to roads and improves safety mechanisms. With help of this Mechanism in the future, we can build a smart autonomous vehicle.

11. Acknowledgment

The author gratefully acknowledges the support provided by the department of information technology, VNRVJIET, Hyderabad, India.

References

1. Ryley, Tim J., et al. "Investigating the contribution of Demand Responsive Transport to a sustainable local public transport system." *Research in Transportation Economics* 48 (2014): 364-372.
2. Greenblatt, Jeffery B., and Susan Shaheen. "Automated vehicles, on-demand mobility, and environmental impacts." *Current sustainable/renewable energy reports* 2.3 (2015): 74-81.
   Austin, Kevin. "The identification of mistakes in road accident records: Part 1, locational variables." *Accident Analysis & Prevention* 27.2 (2015): 261-276.https://timesofindia.indiatimes.com/city/mumbai/sleepy-driver-crashes-suv-on-mumbai-pune-expressway-7-passengers-killed/articleshow/18801253.cms.
3. Sonkin, Beth, et al. "Walking, cycling and transport safety: an analysis of child road deaths." *Journal of the Royal Society of Medicine* 99.8 (2016): 402-405. http://archive.indianexpress.com/news/accidents-on-expressway-prompt-fresh-survey-of-state-highways/1083181/
4. Khadaroo, Jameel, and Boopen Seetanah. "Transport infrastructure and tourism development." *Annals of tourism research* 34.4 (2017): 1021-1032. https://timesofindia.indiatimes.com/india/traffic-junctions-account-for-half-of-road-deaths-in-india/articleshow/40956759.cms
5. Haboucha, Chana J., Robert Ishaq, and Yoram Shiftan. "User preferences regarding autonomous vehicles." *Transportation Research Part C: Emerging Technologies* 78 (2017): 37-49.
6. De La Escalera, Arturo, et al. "Road traffic sign detection and classification." *IEEE transactions on industrial electronics* 44.6 (2015): 848-859.
7. Zhu, Zhe, et al. "Traffic-sign detection and classification in the wild." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016..
8. Gavrila, Dariu M. "Traffic sign recognition revisited." *Mustererkennung 1999*. Springer, Berlin, Heidelberg, 2018. 86-93.
9. Houben, Sebastian, et al. "Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark." *The 2013 international joint conference on neural networks (IJCNN)*. Ieee, 2013.
10. Guo, J., Zhang, H., Zhen, D., Shi, Z., Gu, F., & Ball, A. D. (2020). An enhanced modulation signal bispectrum analysis for bearing fault detection based on non-Gaussian noise suppression. *Measurement*, *151*, 107240.
11. Jogin, Manjunath, et al. "Feature extraction using convolution neural networks (CNN) and deep learning." 2018 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT). IEEE, 2018.
12. Chua, Leon O., and Tamas Roska. "The CNN paradigm." IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications 40.3 (1993): 147-156.
13. Zheng, Yufeng, et al. "CNN classification based on global and local features." *Real-Time Image Processing and Deep Learning 2019*. Vol. 10996. International Society for Optics and Photonics, 2019.
14. Wei, Wang, et al. "Image object recognition via deep feature-based adaptive joint sparse representation." *Computational Intelligence and Neuroscience* 2019 (2019).
15. Cao, Jianfang, et al. "An improved convolutional neural network algorithm and its application in multilabel image labeling." *Computational Intelligence and Neuroscience* 2019 (2019).
16. Yu, Wei, et al. "Visualizing and comparing AlexNet and VGG using deconvolutional layers." *Proceedings of the 33 rd International Conference on Machine Learning*. 2016.