Traffic Sign Recognition Using Deep Learning: a Better Way to Safe Driving

Rishub Kumar and Shekharesh Barik

March 4, 2024

Traffic Sign Recognition using Deep Learning: A Better Way to Safe Driving

Rishub Kumar[1] and Shekharesh Barik[2]

[1] UG-Student, Department of CSE, DRIEMS University, Cuttack, India
[2] Asst. Professor, Department of CSE, DRIEMS University, Cuttack, India
itsrishub@gmail.com

Abstract. The number of accidents caused by failure to observe traffic signs and follow traffic laws has been steadily growing. Using synthesised training data generated from road traffic sign photos, we may overcome the limitations of traffic sign detection databases, which differ between countries and regions. This technology is used to create a library of synthesised images for detecting traffic signs under various lighting situations. We can create a data-driven traffic sign identification and detection system with high detection accuracy and high-performance capabilities in training and recognition procedures using this data set and a perfect Convolutional Neural Network (CNN). This reduces the number of accidents and allows the driver to concentrate on driving rather than studying every traffic sign. The goal of this work is to present an effective approach for detecting and recognising traffic signs in India. We developed approaches such as neural networks and feature extraction to overcome the limits of existing methods, increase the effectiveness of recognising traffic signs, and minimise road accidents.

Keywords: Convolutional neural network, Feature extraction, Road accidents, Traffic sign recognition.

1 Introduction

Machine learning algorithms have grown in popularity recently. Machine learning is used in a variety of applications, including spam filtering, audio recognition, facial recognition, and traffic sign identification. Traffic Sign Recognition and Classification can be used in traffic zones to automatically recognise traffic signs. The system does this automatically when a traffic sign is recognised and the sign name is shown. As a result, even if the driver misses a sign or loses attention, it will be noticed. This allows us to warn drivers and prohibit specific actions, such as speeding. It also relieves the driver's burden, increasing his or her comfort. As a result, guaranteeing and monitoring traffic signs and adhering to them. Traffic signs, in fact, supply us with a wealth of information and direct us properly so that we can travel safely. Automatic Driver Assistance Systems benefit greatly from traffic sign classification. Convolutional neural networks are a type of deep learning network that is used to analyse and validate visual information. Because of its excellent accuracy and precision, it is used to train picture categorization and recognition models.

- Dataset of Traffic Signs The availability of a generalised dataset is critical before going on to detection or classification. This dataset is used to train a prediction model, and predictions are made for the test dataset. Sample datasets are shown in Table I below:

The GTSRB (German Traffic Sign Recognition Benchmark) dataset is the most commonly used. Its popularity stems from:

1. It consists of large number of images.
2. The traffic signs are of different variety, background, and color variation which in turn will help the model to perform accurately.

The suggested approach takes use of the GTSRB dataset since it may be utilised for both detection and classification. The dataset is divided into three parts: training, testing, and validation. The training dataset is the one from which the model is trained. In general, the validation dataset is used to test the model and update the hyper parameters. Hyper parameters, including as the number of epochs and the activation function, are used to regulate the learning process and increase accuracy. The test dataset is utilised only after the model has been trained. It is used to determine whether or not the model can generate accurate predictions. Furthermore, for the training, testing, and validation data sets, histogram graphs are generated to illustrate the number of photos in each class, where the X label specifies the "Class ID" and the Y label reflects the number of images. Plotting the graph aids in the visualisation of the dataset.

2 Literature Survey

2.1 Detection Using CNN Ensemble

The approach provided by Shustanov and P. Yakimov for Road Sign Detection and Recognition is an image processing methodology that comprises of an ensemble of (CNN) for recognition. The CNN has a very high recognition rate, making it more suitable for many computer-based vision applications. TensorFlow is the technique used to execute CNN. Using German data sets, the contributors of this research obtained more than 99 percent accuracy for circular signs.

2.2 Recognition Using Color Segmentation

Wali etal show how they utilised to create a unique approach for sign recognition. They employed sophisticated ARK-2121 technology, which is a tiny computer that was mounted aboard the vehicle. SVM and HOG were the primary approaches used in the sign recognition process. They obtained 91% detection accuracy and an average classification accuracy of 98%.

2.3 The German Traffic Sign Recognition

The study and design procedure of the "German Traffic Sign Recognition Benchmark" dataset is described by R. Qian etal. The results of this experiment demonstrated that machine learning algorithms performed admirably in the recognition of traffic signals. On these datasets, the participants achieved an extremely high identification rate of 98.98%, which is equivalent to human perfection.

3 Existing System and Its Disadvantages

The previous approaches for recognising traffic signs on the road are insufficient to deal with real-world problems, and even if some of them are, they have efficiency and cost difficulties that make them unsuitable for application. The tables below demonstrate a handful of the related algorithms, along with their accuracy and flaws.

Table 1. Representation of Existing System

Ref.	Accuracy	Algorithm	Issues
[1]	78%	K-means clustering	False Detection
[2]	82%	Lidar & vision based	Redundancy (inter pixel redundancy)
[3]	88%	Video strreaming	Less efficiency
[4]	76%	Machine Learning	Low accuracy
[5]	86%	Lidar & Deep Learning	Cost related issues

4 Proposed System

The suggested framework is divided into three stages: detection, feature extraction, and recognition. The detecting stage is simply utilised to locate a traffic sign. When a car is going at a certain speed, the camera captures a road sign in nature, and our formula determines whether or not a sign is available in that perimeter. The design and colour of the traffic sign help distinguish it. The suggested computation characterises the distinct road sign during the feature extraction step. This is performed with the help of the "Convolutional Neural Network" algorithm, which divides the picture into sub-categories.

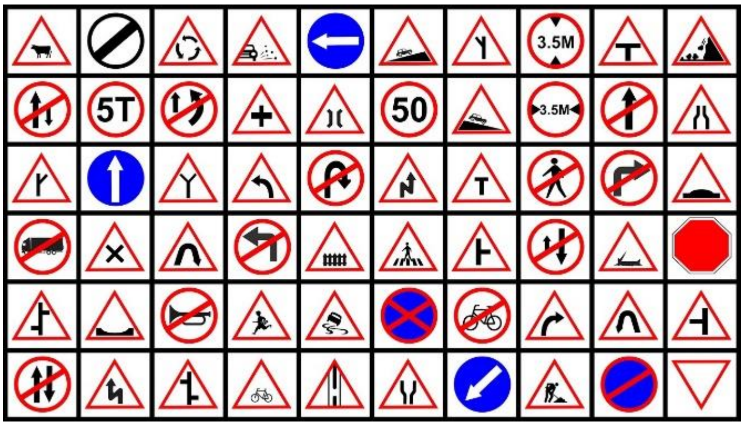

Fig. 1. Types of Traffic Signs

4.1 Flow Chart of Model

Flow chart of the model shows exactly how the algorithm is working on order to give the best accuracy possible. Below is the representation of flow chart with step wise explanation of each process:

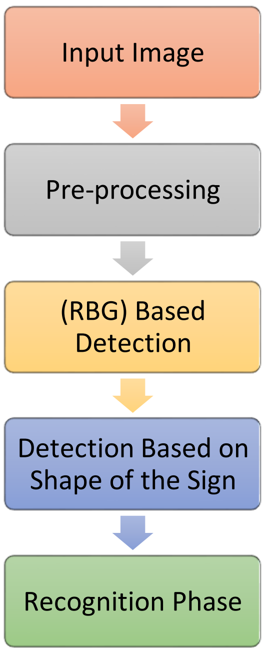

Fig. 2. Flow Chart of Model

— **Step 1:** Input Image

A camera installed inside the car records a movie that is nothing more than a collection of images. Typically, 24 frames per second are used. These photographs are processed in order to find signs. The camera's quality should be such that it clearly shows the traffic sign from the shortest distance possible. In most cases, a camera with more over 8MP is necessary for this operation.

Fig. 3. Recognized Traffic Sign as Input Image

– **Step 2:** Pre-processing

The scanned picture or video is pre-processed using a convolutional neural network. The higher resolution image is scaled down to a lower resolution, and the RGB image is transformed to greyscale format so that it may be readily processed by "Convolution Neural Networks." The neural networks are quite similar to the neurons in our brain in that they perceive the image and send it to the processor. The picture below depicts a left or straight sign that has been transformed to greyscale and input into CNN. It is then sampled and fragmented into little parts, and the resulting output is compared to the data set provided, and the corresponding voice output is provided. Neural networks, like neurons in our brain, detect images and provide the essential information to processors. When compared to other classification techniques, the steps required for pre-processing in a Convolution neural network are less.

Fig. 4. Pre-processing of Image to Greyscale

– **Step 3:** (RBG) Based Detection

The colour is the most important aspect of a sign. When the red colour is visible, it is assumed that there is a traffic sign on the road. The similar concept is applied in our detecting technique. Based on the frames collected, our system is programmed to look for a sign based on the red colour. If a portion of the image is comparable to the threshold values of red colour, it is passed on to the next stages to determine whether or not it is a sign. The major indicator in the red area is to be recognised once the red threshold is checked.

- **Step 4:** Detection Based on Shape of the Sign

We determine the number of edges using the Douglas-Peucker algorithm based on the prior detection approach. We focus mostly on two forms, circle and triangle, because these are the most often used traffic sign shapes. The number of edges and region of interest are discovered using the Douglas-Peucker method. Now, if the number of edges detected is equal to or higher than six, and the major section meets the minimal criteria, it is termed a triangle. And if the edges are equal and bigger than six, as well as meeting the minimum criteria, the majority of the picture is recognised as a circle. Following the recognition of the forms, the next key step is to identify the bounding box. The bounding box is crucial because it separates the Region of Interest (ROI) from the surrounding environment. Typically, the box contacts the primary region's circle or triangle. A triangular sign is made up of two triangles: the outer triangle and the inside triangle. The outside triangle barely touches the bounding box, but the inner triangle does not.

Fig. 5. Pre-processing of Image to Greyscale

- **Step 5:** Recognition Phase

When a sign is detected, its classification must be completed. Convolu-tional Neural Network is created using TensorFlow, a machine learning method from Google. The initial step in this phase is to preprocess the picture from the previous steps. The most important aspect of the recognition phase is the testing and training of the CNN. We used "German Traffic Sign Benchmark and Bel-gium Traffic Signs" for testing and training the data set. CNN is termed the brain since it has the same properties and processes as a normal brain. Each neuron gets information and transmits it to the next neuron. CNN has several levels. The first layer is the input layer, while the last layer is the output layer. The hidden layer is located between the first and last layers. This strategy employs six CNN lay-ers. A properly linked concealed layer is there to prevent overfitting in the midsection. In this model, we employed a "sequential stack" designed by Keras that operates on top of TensorFlow. "Rectified Linear Unit activation" is present in all levels. ReLu is the most significant activation function in neural networks. The output of the sixth conv layer, which uses a level capacity to straighten the yield, is the input is totally connect-ed layer. The last layer consists of Softmax activation and provides the flattened output. A max pooling layer is present right after two layers to boost processing performance. To achieve more ideal outcomes, we utilise a collection or group of CNNs, in this case three. If we employ more than one CNN, the result will be more accurate. The optimizer, loss, and metric must all be supplied. Instead of percentages, the loss use numbers ranging from 0 to 1. "Stochastic Gradient Descent with Nesterov Momentum" is used by the optimizer. Epochs are used to improve training and thereby increase prediction accuracy. Finally, a text-to-speech module is added to the system, which recognises traffic signs and produces output in the form of voice, allowing the driver in the automobile to readily recognise the signs and prevent accidents.

5 Conclusion & Future Scope

In this paper, we present an autonomous traffic sign segmentation to make detection easier. Convolutional neural networks aid in the quicker and more accurate identification of traffic signs. Our work contributes to the feature of automated cars. This research focuses mostly on deep learning-based automated segmentation. We achieved decent results by training with 25 photos. As the quantity of testing photos grows, so does the accuracy.

Our system detects signs continuously, which leads to detection of signs even when there are no indicators in the vicinity, resulting in a continual flow of output. As a result, incorrect or unneeded detection occurs. This might be improved by raising the sign detection threshold. More datasets from other nations can also help to enhance and customise overall performance.

References

1. Chang Failing, Huang Cui, Liu Chengdu, et al. Traffic sign detection based on Gaussian color model and SVM [J]. Chinese Journal of Scientific Instrument, 2014, 35(1): 43-49.
2. Ge Xia, Yu Feng in, Chen Ying. Ransack Algorithm with Harris angular point for detecting traffic signs [J]. Transducer and Microsystem Technologies, 2017, 36(3): 124-127.
3. Liu Hansen, Zhao Xiangmo, Li Qian, et al. Traffic sign recognition method based on graphical model and convolutional neural network [J]. Journal of Traffic and Transportation Engineering, 2016(3): 124-127.
4. Wang Xiaoping, Huang Janie, Liu Wensum. Traffic sign recognition based on optimized convolutional neural network architecture [J]. Jamal of Computer Applications, 2017, 37(2): 530-534.

5. Xin Jin, Cain Fixing, Deng Haiti, et al. Traffic Sign Classification Based on Deep Learning of Image Invariant Feature [J]. Journal of Computer-Aided Design & Computer Graphics, 2017, 29(4): 632-640.
6. Ren S, He K, Airsick R, et al. Faster R-CNN: towards realtime object detection with region proposal networks [C]// International Conference on Neural Information Processing Systems. MIT Press, 2015:91-99.
7. Krizhevsky A, Sutskever I, Hinton G E. Image Net classification with deep convolutional neural networks [C]// International Conference on Neural Information Processing Systems. Curran Associates Inc. 2012:1097-1105.
8. Zhimin xu, Si Zuo. AutoSegNet: An Automated Neural Network for Image Segmentation, IEEE Access, 2020, 92452 – 92461
9. Hoban S, Stall Kamp J, Salman J, Et al. Detection of traffic signs in real-world images: TheGerman traffic sign detection benchmark [C]// International Joint Conference on Neural Networks. IEEE, 2013:1-8.
10. Zhou Zhuhai. Machine learning [M]. Beijing: Tsinghua University Press, 2016: 23-47.
11. B. Zoph and Q. V. Le, ``Neural architecture search with rein-forcement learning," 2016, arXiv:1611.01578.
12. H. Liu, K. Simonyan, O. Vinyals, C. Fernando, and K. Kavukcuoglu, Hierarchical representations for efficient architecture search, 2017, arXiv:1711.00436.
13. H. Pham, M. Guan, B. Zoph, Q. Le, and J. Dean, Efficient neural architecture search via parameters sharing,in Proc. 35th Int. Conf. Mach. Learn., vol. 80, 2018, pp. 4095-4104.
14. H. Cai, J. Yang, W. Zhang, S. Han, and Y. Yu, Path-level network transformation for efficient architecture search," in Proc. Int. Conf. Mach. Learn., 2018, 678-687.
15. B. Baker, O. Gupta, N. Naik, and R. Raskar, Designing neural network architectures using reinforcement learning,2016, arXiv:1611.02167.
16. O. Ronneberger, P. Fischer, and T. Brox, U-Net: Convolutional networks for biomedical image segmentation, in Lect. Notes Comput. Sci., 2015, pp. 234-241.
17. L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, Encoder- decoder with atrous separable convolution for semantic image segmenta-tion, in Proc. Eur. Conf. Comput. Vis., 2018, pp. 833-851.
18. L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs, IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 4, pp. 834-848, Apr. 2018