WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Real Time TSR Using CNN and OpenCV

Pritesh Narvankar[1], Rahul Bagwe[1], Siddhant Yerandkar [1], Seema Ladhe[2]

[1]Student, Dept. of I.T.,VPPCOE & VA,Mumbai University,Mumbai,India

[2]Professor, Dept. of I.T , VPPCOE & VA,Mumbai University,Mumbai,India

ABSTRACT

Traffic Sign Recognition System (TSRS) plays an important role in today's world of computerized Intelligent Transportation System (ITS). TSRS used identify sign accurately and effectively and helps to improve on autonomous driving and traffic safety. This paper proposes a Traffic Sign Recognition System which is developed on the basis of Convolutional Neural Network (ConvNet/CNN) and OpenCV Libraries. Existing TSRS can dodetection of only specific signs, colors or shapes and cannotdo real-time detection. Therefore, the main objective of this paper isto overcome the existing limitation and provide a complete and effective TSRS. Our proposed CNN Model istrained using a German Traffic Sign Recognition Benchmark (GTSRB) which consist of around 20000 images, also to check the robustness of the project it was tested and validated against 8100 and 6000 images respectively. Using our proposed system, theresult showsin around 98-100% accuracy.

Keywords— Convolutional NeuralNetwork,Traffic Sign Recognition System, ITS, GTSRB)

Introduction

TSR(Traffic Sign Recognition) is a system that allows a vehicle to detect traffic sign placed on a road, such as "speed limit", "turn ahead", "No Parking". This is part of the feature collectively called ADAS. It detects traffic sign using image processing techniques. Color-based, Shape-based and learning-based are the three types of detection methods.

The human visual perception abilities are dependent on the individual's physical and mental conditions. As a result, various things such as exhaustion and driving tension might impair these abilities [4]. As a result, having a TSRS in place to help the driver in such situations is critical. TSRS can identify traffic signs in real time and relay this information to the driver, preventing accidents, saving lives, and generally improving the user's driving ability.

Traffic signs have several unique features that may be used for their detection and identification. Colours and shapes are the most important attributes that assist and get better driving conditions[3].Colours and Shapes are the most distinct features of a traffic sign, since traffic sign are either Red, Blue ,Green and are in circular, triangular or hexagonal shaped.

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning system that can take an input image, assign relevance (learnable weights and biases) to various aspects/objects in the image, and distinguish between them. When compared to other classification methods like Speed Vector Machine (SVM), Artificial Neural Network (ANN), Optical Character Recognition (OCR) the amount of pre-processing required by a ConvNet is significantly less. While basic approaches require hand-engineering of filters, ConvNets can learn these filters/characteristics with enough training[5].

In this paper the detection of traffic sign is achieved by four modules: pre-processing, training the model, creating pickle object and recognition. In the pre-processing stage various image segmentation techniques are applied to extract the features of a traffic sign, afterwards the model is trained on the basis of these images, since CNN takes huge amount of computation cost converting the trained model into a pickle object makes the training model phases a one-time training phase resulting in reducing time required and finally in the recognition phase with the help of pickle object and OpenCV module the recognition of traffic sign takes place. Compared to the other classification methods the accuracy achieved through CNN approach is extremely high. Our CNN model is based around two parameters Loss and Accuracy. Using CNN as his classification method ourproposed system has achieved accuracy around 98-100%

The rest of the paper is organized as follows. Section II describes the literature survey of the TSRS. Section III describes the design flow of the proposed approach followed by Experimental Results in section IV. Section V concludes the paper

.

Literature Survey

Md Tarequl Islam [1] uses a British Traffic Sign Dataset as input for their proposed TSR. The British Traffic Sign Dataset consists of 40,000 images (28,000 positive images (Images of traffic Sign) & 12,000 negative images (Images that looks like a traffic sign)) are used to train the classifier and 3,600 images (2,400 positive images& 1,200 negative images) are used to test the classifier.

HSV (Hue, Saturation, Value) color space is used for detecting red and blue colors in the image as feature candidates which are then used as input for classifier.

Two CNNs are implemented. One CNN is used to detect the shape of the contour i.e., Circle, triangle, irregular, random, etc. Second CNN is used to detect whether the contour contains any traffic sign or not, and if present then it detects the traffic sign.

The Major disadvantage of their proposed system is that it detects only red and blue colored signs, Also the U.K. traffic sign dataset covers only 28 signs which are globally used. Thus, their proposed system can detect only 28 traffic signs Also their proposed system cannot detect signs in real-time.

Traffic Sign Detection and Recognition Based on Convolutional Neural Network[2] proposed by Ying Sun *et al.*, uses the German Traffic Sign Recognition Dataset (GTSRB). The images are classified into 43 classes. The training set contains 51,839 labeled images. The image dataset is divided into two sets, 90% is used as training sets whereas the remaining 10% is used as test sets.

Firstly, image undergoes preprocessing to highlight important information. Then, Hough Transform is used for detecting circular shapes in the input image.

Afterwards, a Convolutional Neural Network (CNN) is implemented to classify and recognize whether traffic sign is present or not. Here, CNN is implemented using TensorFlow which is an open-source software library used for implementing ML tools.However their proposed system is able to detect only those traffic signs which are present on a circular-shaped contour. Also, the proposed system is unable to detect signs of any other shapes for example, signs which are having triangular contour.

Rubel Biswas *et al.*, have proposed a TSR for Detecting and Classification of Speed Limit Traffic Signs [4]. Their proposed system uses a training set of 270 property curves of digits to train SVM classifier. 210 images were used for testing which included 213 speed limit signs and 288 non-speed signs. The authorsperformed the Segmentation on input image to remove all undesired objects. Their proposed systemuse Canny edge detection algorithm to extract edges from the image. Afterwards, Circular Hough Transform (CHT) is used to detect circular shapes. The output of this step acts as candidate area from the image where digit segmentation will be carried out in the next step. Otsu thresholding algorithm [8] is used to binarize cropped image and perform digit segmentation. Finally, SVM classifier is used to recognize the segmented digits using training set consisting of property curves for each digit from 0 to 9.

However their proposed system can detect only speed limit signs i.e., it cannot detect general traffic signs like No Entry, Stop, etc. Since it uses Hough Transform for candidate selection, thus it is restricted to detect only signs which are present on circular contours, meaning that signs of other shapes cannot be detected.

Md. Abdul Alim Sheikh *et al.* [3] uses, the database which consists of 4 traffic signs: Stop sign, No Entry sign, Give Way sign and Speed Limit Sign. They have used 300 images for training purpose, out of which 75 images used for each of 4 sign type. The authors have used 200 images for testing the network i.e., 50 images per sign type.

The authors created two modules: road sign detection module and classification + recognition module.Initially, color space conversion and segmentation are done to find out if a traffic sign is present. If the Region of Interest (ROI) is between 100 to 3000 pixel it is considered as a traffic sign hence neglecting all other noise. If it is present, then the sign will be highlighted and normalized in size. Then, it is classified using Neural Network (Multi-Layer Perceptron (MLP) is used here).Then, it is tested against the testing dataset and then evaluated for accuracy of detection and recognition phase respectively. Since their proposed system focuses only on four particular signs (i.e., Give Way, No Entry, Stop, Speed Limit) hence resulting in high accuracy of detection of these signs. Since only 500 images are used in training the MLP here, processing power is required less and thus it can run on low end PC configurations as well.However, their proposed system can detect only four types of traffic signs i.e., Stop sign, No Entry sign, Give Way sign and Speed Limit Sign. Thus, its scope of identifying traffic signs is very limited. Also, the proposed system cannot detect traffic signs which are present in a cluster i.e., if multiple signs are present in an input image, then the system fails to recognize all of them and will detect only one of the four types of traffic sign said above.

Pranjali Pandey *et al* [11] uses Template Matching technique specifically CCORR_NORMED(A Template Matching technique) to identify traffic sign.Their proposed technique is divided into two parts mainly into Training phase and Testing phase. During training phase signs are imported from IRRS(Indian Regulatory Road Signs) then these signs are used to train the system. Afterwards a template is obtained, by manually selecting the road sign andcapturing it as a ROI. Once the template is selected it is given its name and saved in the database. Also during template creation if the image captured is dull due to lighting condition, Histogram equalization is used for color correction of that image.In the testing phase the captured source image is slided over the template image, sliding means matching both the image pixel by pixel from left to right and top to down approach. All the matched pixel values are stored in a matrix and once the highest intensity matrix is obtained it gives an output message by a voice trigger and a display message.

Though their proposed system can detect traffic sign in any lighting conditions however,it is unable to detect traffic signs from low-quality blurred images as it is difficult to match the intensities of blurred images during template matching. Since it is just Template Matching, false positives (showing presence of a sign which isn't actually present in the image) are also reported which hinders the efficiency of the system.

Richa Jain *et al.* [12] proposes a hybrid approach for detecting and recognizing traffic text signs based on MSER (**Maximally Stable Extremal Regions**) and OCR(**Optical Character Recognition**).The dataset used is movement content sign information from Jaguar Land Rover Research. The system is divided into two steps mainly text area detection and text recognition. Noise removal and de-blurring of image is done using Lucy Richardson (LR) algorithm after detection of the text. Afterwards the contrast of the image is adjusted and then the image is converted into a binary grayscale image using rgb2gray function. Canny edge detection algorithm is used for image edge detection and Maximally Stable Extremal Region (MSER) region detection algorithm is applied to the edge enhanced image. Finally, the image undergoes morphological segmentation and geometric filtering. OCR recognizes text found inside the bounding box.

The major drawback of their proposed system is that it provides a partial solution for traffic sign recognition as only text signs (e.g.- Left Turn on Green) are detected. Also, terribly massive and extremely small objects are rejected in the geometric filtering phase, so if a sign is written in very small text, then it may get rejected depending on circumstances.

Considering the drawbacks of existing systems like detection of only specific signs, specific colors, specific shapes and no real-time detection, we aimed to provide improvement in Autonomous Driver Assistant System (ADAS), to provide a system which can detect traffic signs in real-time and to provide a system which can detect signs not limited to a single shape or color and also provide high accuracy at the same time.

In the next section we detailed our proposed system..

Proposed System

In the above section we discussed about the existing work on TSR. This paper is aimed to recognise all types of traffic sign irrespective of shapes and colors. The details of our proposed system are as given below.

Libraries to be used: - NumPy, Matplotlib, Keras, OpenCV, Scikit-learn, Pickle, OS, Pandas, Random.

Input Data: - We used German Traffic Sign Recognition Benchmark (GTSRB)[6]. The input dataset contains 34,799 labelled images which are classified into 43 classes. The details of 43 classes are given in below Fig 1.

For training, testing and validating sets, we split total images into 80:20 for training to testing sets ratio. From this training set, we again split total images into 80:20 for training to validating sets ratio.Thus, dataset is divided into three parts, 64% of the dataset is used as training sets, 20% of the dataset is used as testing sets whereas the remaining 16% of the dataset is used as validation sets.

Fig 1: Set of traffic sign images

Therefore, 22,271 images are used for training, 6960 images are used for testing, and 5568 images are used for validating.

Fig. 2 shows the distribution of input dataset (34,699 images) into 43 classes. This distribution of input dataset into 43 classes is as per the input dataset used.

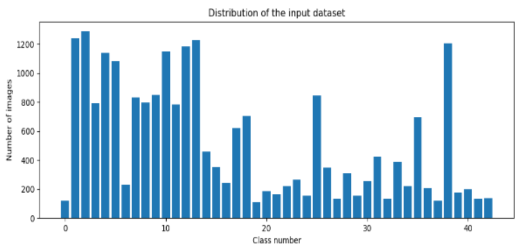

Fig 2: Distribution of Input Dataset (34,799 images)

Pre-processing: -Data preprocessing is a process of preparing the raw data and making it suitable for Machine Learning.

The input images in the dataset are of size 32x32 with 3 channels i.e., Red-Green-Blue (RGB). These images and their classes are loaded into a matrix using NumPy. The input images in the dataset are of size 32*32*3 (where, 3 denotes number of channels i.e., 3 channels = RGB).

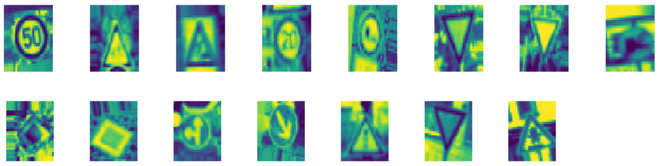

Fig 3: Image Augmentation Technique used on training dataset

First,we converted all training images into grayscale, then we improved the contrast by equalizing histogram and normalizing values.Then, images are augmented (i.e., zoom,rotation, width shifting, height shifting, shearing, etc.) as shown in Fig. 3.

Classification: - We implemented a Convolutional Neural Network (CNN) based on LeNet-5 architecture developed by Yann LeCun [6]. The system architecture of LeNet-5 is shown in Fig. 4.

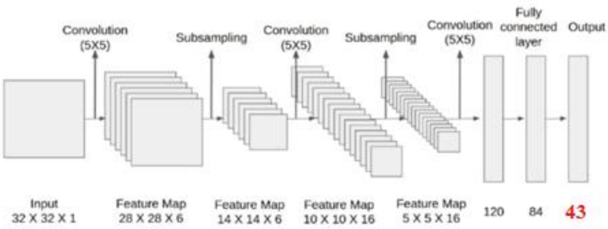

Fig 4: LeNet 5 CNN model [6]

The Fig. 4 is taken from the source [6], however in this figure we are showing 43 output classes as there are 43 traffic signs present in our dataset which are to be classified. First, this model is implemented, and then it is trained using training dataset and validated using validation dataset.

This trained model is stored as a pickle object which is then used in OpenCV. We used OpenCV to test our trained model for detecting signs in real-time.

Fig. 5 shows the layout of our proposed system.

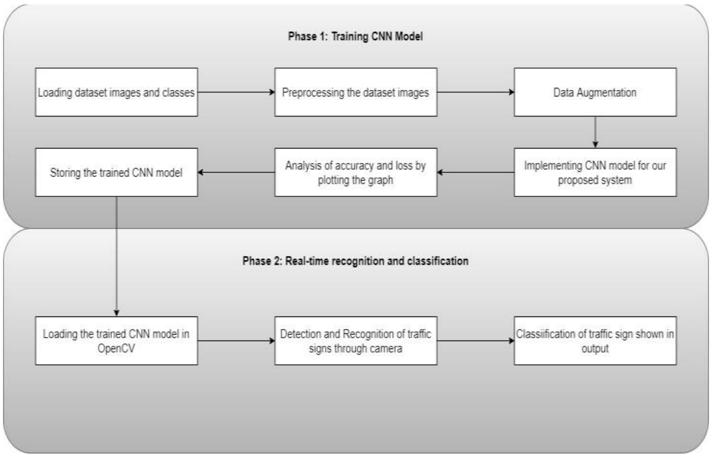

Fig 5: Layout of our proposed solution

In this section we elaborated our proposed system. In next section we discuss the results.

Result

We consider the following metrics to judge our proposed system: Loss [10] and Accuracy[8]

Accuracy – Accuracy is a metric that measures how many of our model's predictions about traffic signs were correct [8]. The accuracy is given below:

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \quad \ldots\ldots\ldots\ [8]$$

Loss - A faulty guess results in a loss**.** To put it another way, loss is a metric that indicates how inaccurate the model's forecast was for a single case [10]. In our model, we have used categorical cross-entropy loss function available in TensorFlow.

We analyze our proposed system by computing and plotting the accuracy and loss as shown in Fig. 6 & 7 respectively.

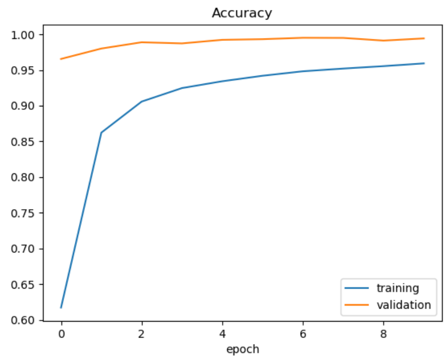

Fig 6: Training and Validation Accuracy

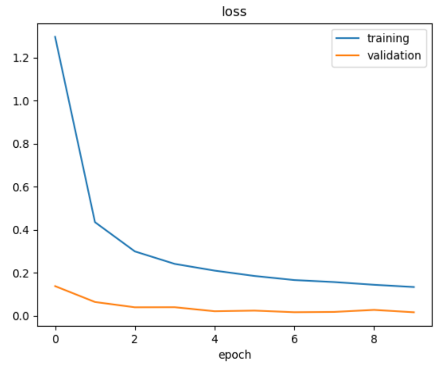

Fig 7: Training and Validation Loss

During the initial epochs of training and validation phase, the accuracy of our model was very low around 62%. However, with increase in number of epochs, the accuracy rises very quickly by 2nd epoch and then there was gradual increase in accuracy up to 99% by the 10th epoch. The accuracy increases because the number of images increases with each epoch during training the model, with increase in number of images the model understands the details and features of a traffic sign hence yielding a highly accurate result.Simuntaneously during the initial epoch when the model has started its training it results in high loss but with enough training the Loss factor of the model decreases drastically.

Here are some of the outputs for real time usage trial using OpenCV:

Fig 8: Image outcome of our proposed system

The above Fig 8 displays the recognition of the traffic sign in real time of our TSR system. It displays the name of the detected sign along with the probability percentage of accurate detection of sign. Higher the probability percentage defines the correct recognition of the traffic sign.

CONCLUSION

Our proposed system can detect and classify a set of 43 traffic signs in different environments likedim lights, rainy weather, etc.Using our proposed system. we observed accuracy around 99% and loss around 1.3%,The probability of accurate recognition of traffic sign is 100% for most of the sign but not all sign results in such high probability which can be improved by providing an equally distributed dataset. Our proposed system overcomes many limitations of the existing TSR like detecting any traffic signs irrespective of their shape & size in any lightning condition, and thus providing a better solution.

References

[1] Md Tarequl Islam, **“Traffic sign detection and recognition based on convolutional neural networks”**, 2019 International Conference on Advances in Computing, Communication and Control (ICAC3)

[2] Ying Sun, Pingshu Ge and Dequan Liu, **“Traffic Sign Detection and Recognition Based on Convolutional Neural Network”,**2019 Chinese Automation Congress (CAC)

[3] Md. Abdul Alim Sheikh, Alok Kole and Tanmoy Maity, "**Traffic sign detection and classification using colour feature and neural network**", 2016 International Conference on Intelligent Control Power and Instrumentation (ICICPI)

[4] Rubel Biswas, Hasan Fleyeh and MoinMostakim, "**Detection and classification of speed limit traffic signs**", 2014 World Congress on Computer Applications and Information Systems (WCCAIS)

[5] https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

[6] https://sid.erda.dk/public/archives/daaeac0d7ce1152aea9b61d9f1e19370/published-archive.html

[7] https://www.analyticsvidhya.com/blog/2021/03/the-architecture-of-lenet-5/

[8] N. Otsu, "A threshold selection method from gray level histogram," IEEE Trans. Syst.ManCybern, vol. SMC-9, pp. 62-66, 1979

[9] https://developers.google.com/machine-learning/crash-course/classification/accuracy

[10] https://towardsdatascience.com/common-loss-functions-in-machine-learning-46af0ffc4d23

[11] Pranjali Pandey and Ramesh Kulkarni, "**Traffic Sign Detection Using Template Matching Technique**", 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)

[12] Richa Jain and Deepa Gianchandani, "**A Hybrid Approach for Detection and Recognition of Traffic Text Sign using MSER and OCR**", 2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)