

Traffic Signboard Recognition and Text Translation System using Word Spotting and Machine Learning

Usha S.M.^{1*}, Mahesh Kumar D¹, and Mahesh H.B.²

¹ JSS Academy of Technical Education, Bengaluru, VTU, India, ushasm@jssateb.ac.in

² PES University, Bengaluru, VTU, India, hbmahesh@gmail.com

Abstract. This project will help the non-native people of Karnataka to easily understand the kannada boards and travel easily. The main task of this work is to recognize the kannada traffic text boards and translate that to English language. Histogram equalization is used to find the gap between the characters. K-means clustering is used to divide the characters into different clusters then the segmented characters are passed to the pre-trained model to recognize what the characters means. The model used for recognizing the traffic text is convolutional neural networks. The methodologies used here is the image augmentation, converting RGB image to grey scale and normalizing the image to reduce the noise. The validation accuracy obtained while training the model with coloured images, normalized image, grey scale image and normalized grey scale image is respectively 98.88%, 98.85%, 98.8% and 99.39% and while testing this model with kannada language, the testing accuracy obtained respectively with coloured images, normalized image, grey scale and normalized grey scale image is 95.91%, 96.58%, 95.42% and 96.98 % . In this work, word spotting method is employed for kannada language recognition. The performance of this system is faster since machine learning algorithms are used here.

1 Introduction

The road traffic safety has become increasingly important all over the world, especially in developing countries and towns, as the modernization period accelerates and the number of vehicles on the road grows, every year, a large number of people lose their lives as a result of road collisions, and the annual average number of those people seems to be increasing. Drivers also ignore signs and text boxes by the side of the road in an effort to stay on the road, which can be unsafe for everyone. This issue could be prevented if there was a quick way to warn the driver without requiring them to change the attention [1]. The flow of traffic sign and text identification and recognition are relevant here because they identify and recognize a sign/text, alerting the driver of any incoming signals. The use of image

* ushasm@jssateb.ac.in

recognition technologies in intelligent transportation systems is a crucial technological component; the importance of traffic signals in road traffic schemes cannot be overstated [2]. But recognizing/ capturing traffic signs/texts while travelling on highways/roads, and detecting what the sign is a difficult task. Because of the surrounding conditions (like weather, daylight, raining etc.) on the traffic signs and texts, it makes the image processing task a little difficult, therefore the first main task to achieve a successful traffic sign and text recognition system is image processing [3]. The lighting changes depending on the time of the day, it depends upon the atmosphere for e.g. dry, foggy, sandy, snow and other factors. Moving vehicles, motorcycles, people, store signs etc., will partly obscure a sign's appearance or create uncertainty. The appearance of the sign can change due to human vandalism or changes in height, visibility angle, and location in the picture, Because of the long exposure to sunshine and the interaction of the paint with the air, the colour of road signs fades over time. The shot taken from a moving vehicle, it is likely to include motion blur and car vibration. The proposed model to accomplish this will be examined in the report in detail in later segments. There are 45 different types of warning signals that are used to alert drivers, buses and pedestrians about dangerous targets. In traffic signs, they all play a significant part. The main Goal of this project is to overcome all the difficulties that occur in the image of traffic sign as mentioned below in Fig. 1.



Fig. 1. Warning, prohibition and mandatory signs.

2 Problem statement

The traffic sign and text recognition and detection system contains data sets with different sorts of traffic signs and text boards with annotations. The images selected for training contains clear images, blurry images, Far from the vision, have sunlight on board, Taken during night, Taken during rain, These images are selected to classify the real world scenario. Since, the data set used here not only contains the clear pictures of data, it also contains the unclear images so that our application can be used in the real world traffic system. The different cases to be considered here are

- If sign board is present, no text board then only detect the sign board
- If text board is present, no sign board then only detect the text board
- If both text and sign board are present, then detect and process them.

3 Related Work

In this paper, the work is divided the sign recognition part into 3 phrases, which consists of detecting the design, tracking the design, classification of sign. The detection of signs from an image which has many background noises is a difficult task. So, in the detection phase they were trying to assume the sign at a certain position in the image. They also detect the sign from the background by comparing the colour of the sign in the image, since the

colours of most sign boards are identical. At last, they used segmentation to separate the sign board from the image. After sign detection, the meaning of the sign identification was carried out by using classifiers. Although the proposed model is satisfying the decision part of the image does not give proper results [4]. As they are assuming a sign is present at a particular position in the image, but in certain cases, there might be no image present at the specified position, hence passing that input to the model gives incorrect output.

In this paper, the work is divided the traffic sign recognition in to mainly 4 phases: pre-processing of traffic sign image, recognizing traffic sign image, extracting the region of interest from image and classifying the image. Here to enhance the traffic sign of the image and to improve the contrast of the image, an histogram equalization enhancement method is used. After recognizing the image, they classified the image using SoftMAx classifier [5,6].

In their work, they used the framework of the CNN-SVM method; they divided the traffic sign recognition into two phases, ie detecting the sign from image, implying what the image mean. In this paper, they have used a colour –based detection method to detect the traffic sign from an image with a lot of background noises. As known, the colour codes for sign boards are unique, so it can be easily distinguished from the background noises. After extracting the regions of interest, they selected a few features from ROI and gave it as input to the classification model. The classification model used here supports vector machines [7-9]. The CNN-SVM approach is successful in identifying traffic signals, according to the findings of the experiments even though it classified few images as wrong, but overall the performance of the model was good compared to others.

In this work pre-processing of images was done using Otsu’s method of binarization, then line-wise and word-wise segmentation of the image is done. Then features like aspect ratio, eccentricity etc are extracted. Then the words are classified based on their features based on a k-nearest neighbour. This method performed very well with an accuracy of over 90% for multiple data sets [10-12].

4 Proposed methodology to detect traffic sign

The first major task is to recognize the traffic sign of the image. After the image is pre-processed, the image is passed to into pre-trained model to get the desired output. Fig. 2. Shows the proposed model to detect the traffic sign.

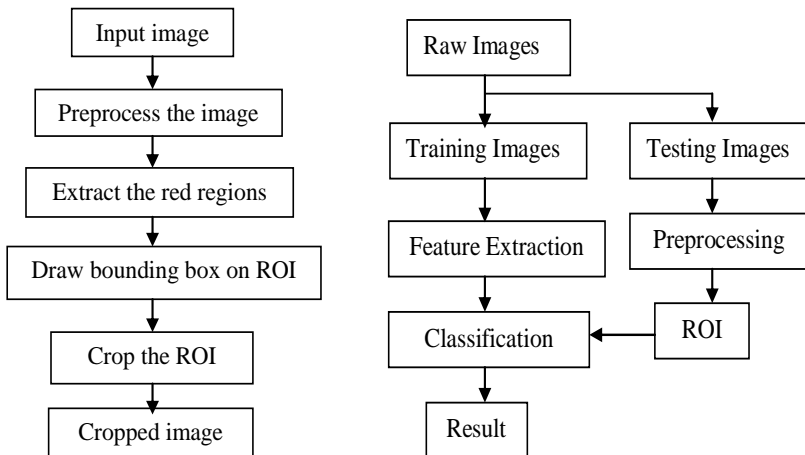


Fig.2. Proposed models to detect the traffic sign.

The augmentation process includes creation of duplicate images from the dataset, the main reason to create duplicate images from the data set is to train the model with lot of images, which increases the training accuracy and decreases the loss of the model. All the images are resized and converted them into grey scale images. Grey scale simplifies the algorithm and reduces computational requirements. The aim of traffic sign identification is to find the region of interest (ROI). After that the possible object is normalized to become an image of the given dimension, and the identification process begins. Converting the obtained RGB image to an HSV is a most common pre-processing step, the HSV(Hue, Saturation, Value) colour space is chosen for identification over the RGB (Red, Green, Blue). When compared to an RGB picture, HSV is closer to what the human eye actually sees.

4.1 System Design

The image is pre-processed, which is done by noise reduction in the image and then the resized. The traffic sign boards should be detected from the pre-processed image, since edges of the traffic sign boards are red in color. This is used to get the ROI (Region of Interest). The red regions can be extracted by HSV colour thresholding. The image is converted into HSV color space. Here the hue and saturation value should be set before performing the color thresholding after that red regions are extracted from the image. Later, edges are detected and from the red regions, dilation is performed on those edges, noise will be removed by calculating its area. Fig. 3. Shows the flow chart of an entire system design. Three different techniques are used here to improve the performance

- By normalizing the images, the image data should be normalized so that the data has mean zero and equal variance.
- Convert an image in RGB format to grayscale
- Normalizing the grayscale images

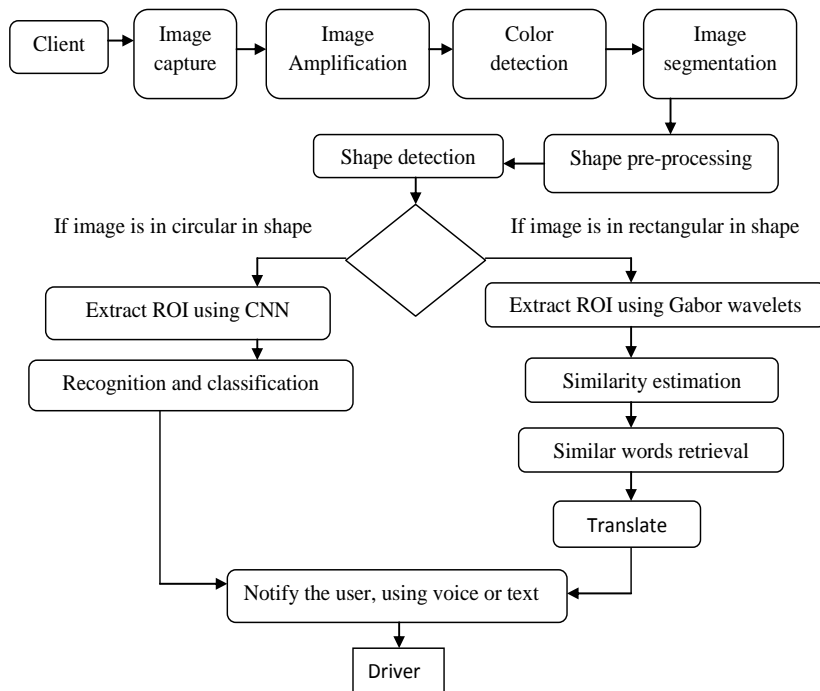


Fig. 3. Shows the flowchart of an entire system design.

Traffic sign recognition module, Sort the images in to classes, Panda’s module is used to arrange images, the signnames.csv file, provides English descriptions of the labels. MSERs and HSV color thresholding are used to detect traffic signs with temporal and structural information during the detection range. An approximate perspective is used in the identification stage to make the plane perpendicular to the camera axis, MSERS first; the individual character’s which is then shaped into text lines. Finally OCR is used to distinguish the text lines that have been identified. The recognition effects of multiple frames are merged in order to increase the recognition accuracy word by word segmentation and detection is used instead of the merging multiple frames constraints, assumptions and dependencies. The efficiency of the detection system is influenced by a variety of factors.

5 Experimental Results and Discussion

The photographs of road were taken from the Indian traffic sign dataset. Photograph includes road signs, photograph set with monitoring camera blurred images. Convolution neural network is used here. It has 9 layers including 5 layers of convolution and simplification function made up of 22 5X5 kernel filters and 2X2 max pooling filter to minimize the 32X32 input images to 16 5X5 maps. By processing the feature images in to a four layer fully connected network. The most important features to describe traffic sign class can be defined, the dropout strategy is used in neural network training to avoid over fitting by turning off certain neurons during the training process. Two different traffic recognition and detection system are shown below in Figure 4 and Figure 5.

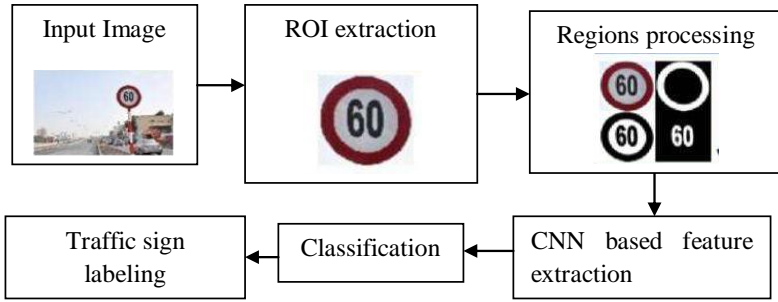


Fig. 4. An ex.1, showing traffic sign recognition and detection system.

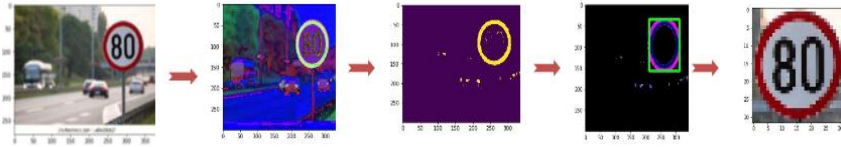


Fig. 5. An ex 2, showing traffic sign recognition and detection system.

5.1 Building and compiling the model

Sequential model is used here; layer by layer model can be built using this model. In neural networks this activation function is found to be effective. The output layer used here is the dense layer and has 43 nodes. Each node is for one potential outcome. The next step is compiling the model; this model is built using three parameters: optimizer, loss and metrics. The learning rate is controlled by the optimizer. Adam is a nice optimizer to employ. Throughout training, the Adam optimizer adjusts the learning rate. The learning rate controls how quickly the model's optimum weights are computed. A slower learning rate is more accurate weights but it will take longer to compute the weights. Categorical cross entropy is the popular categorization method. A lower score indicates that the model is performing better. Accuracy metric is used here to see the accuracy score on validation set during training the model.

5.2 Training the model

To train the models, the model's fit function is used with the following parameters. Training data (train X), target data (train y), validation data and epochs. All the three types of pre-processed images are passed to this model separately and compared the training and test accuracies obtained as depicted in Fig. 6. below.

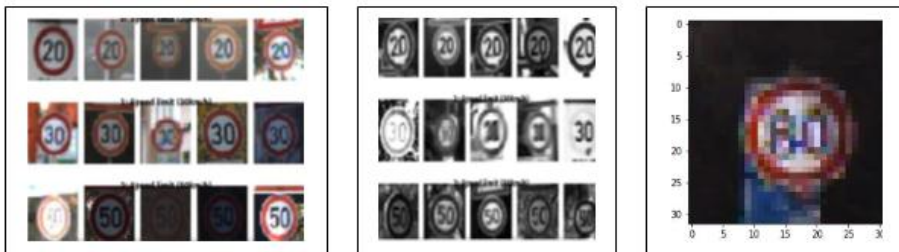


Fig. 6. Training model.

5.3 Line wise segmentation

The horizontal projection profile of the document image is used to distinguish the text lines from the document image. Text lines are segmented using white space between them along with its horizontal projection. Along the text row, the projection profile would have valleys of zero height. At these stages, line segmentation is performed as represented in Fig. 7.



Fig. 7. Line wise segmentation of kannada text.

5.4 Character wise segmentation

The characters are segmented properly using the space between the words as depicted in Fig. 8. As it can be seen from the above result, the results are really good compared to the first method.

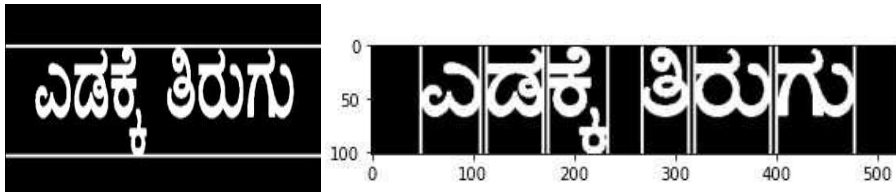


Fig. 8. Character wise segmentation of kannada text.

K-means clustering is performed to divide the gaps between words into characters and words as it can be seen, two clusters are formed correctly which will further help in constructing the characters as depicted in Fig. 9.

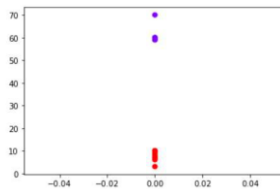


Fig. 9. K-means clustering to divide the gap between the words.

5.5 Data pre-processing

The dataset contains a total of 240 classes. It contains a total of 4352 images, the images are in the text format and few are in handwritten format. All images are different sizes. As the images are of different shapes, therefore the original irregular shaped images are converted into sizes of 32, 32. After reshaping, the images are normalized to remove the noise in the images and at last converted it to greyscale image for proper recognition.

5.6 Model building and text translation

The model building and training is shown in Fig. 10. Recognized kannada text translation and producing the output, after kannada text is being recognized, the next main task is to convert the recognized kannada text in to English language such that it can be understood by everyone. Here, google API is used to translate the kannada text, the input to google API is recognized kannada text and the output expecting is the meaning of our recognized kannada text translated in English. Fig. 11. shows the input to the model to recognize the kannada text. Fig. 12. Depicts the translated English text to the corresponding kannada text.



Fig. 10. Model building.

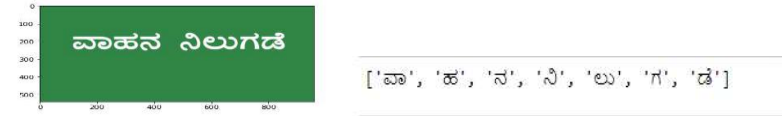


Fig. 11. Input to the training model to recognize kannada text.

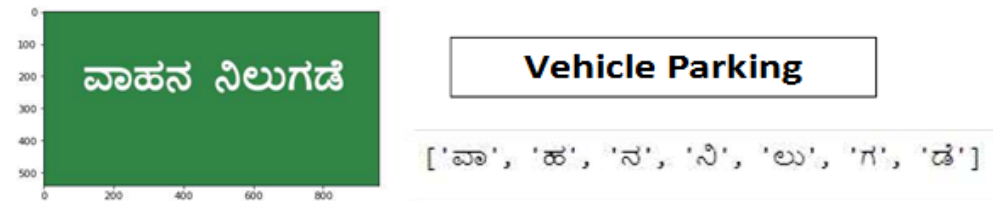


Fig. 12. Output showing translation of kannada to English.

5.7 User Interface for sign detection, recognition, and translation

The process involved is traffic sign detection, recognition and translation. Tkinter is used to develop a user interface. It's a standard python interface to the Python-supplied Tk GUI toolkit. The fastest and simplest approach to construct GUI apps is with Python and Tkinter. The main buttons in this Gui are recognize sign, detect sign and translate sign as shown in Fig. 13.



Fig. 13. GUI showing traffic sign detection, recognition and translation buttons.

5.8 User Interface for sign detection, recognition, and translation

When a kannada text is uploaded, the function segments the characters from the word. At last, the segmented characters are passed into pre-trained model to get the translated kannada text into English. The output is in the form of voice and text as shown in Fig.14.

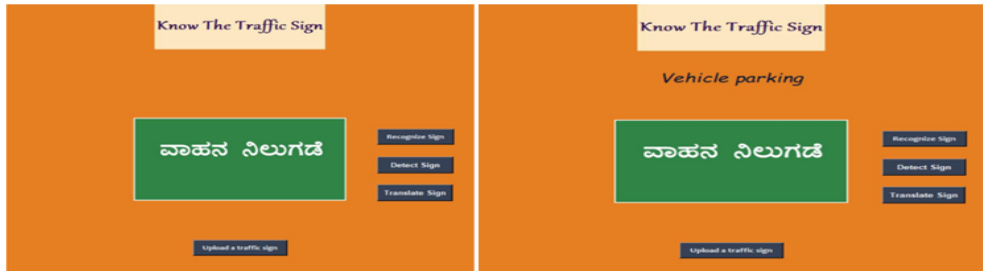


Fig. 14. showing results for applied input image.



Fig. 15. ROI created and detecting red colour object.

5.9 Training validation accuracy and loss

Accuracy and loss obtained from coloured images as shown in Fig. 16 & Loss :0.2093, accuracy: 0.9514 val_loss:0.0411, val_acc:0.9888

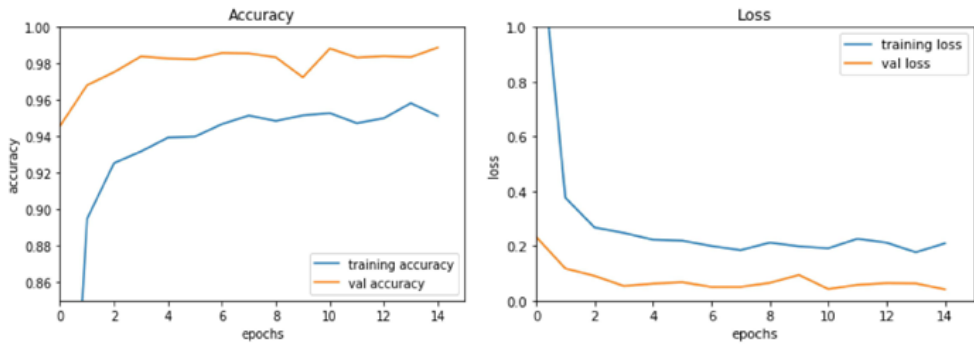


Fig. 16. Accuracy and loss for coloured images.

Accuracy and loss obtained from normalized images as shown in Figure 17 & Loss:0.0892, accuracy: 0.9768 val_loss:0.0513, val_acc:0.9885

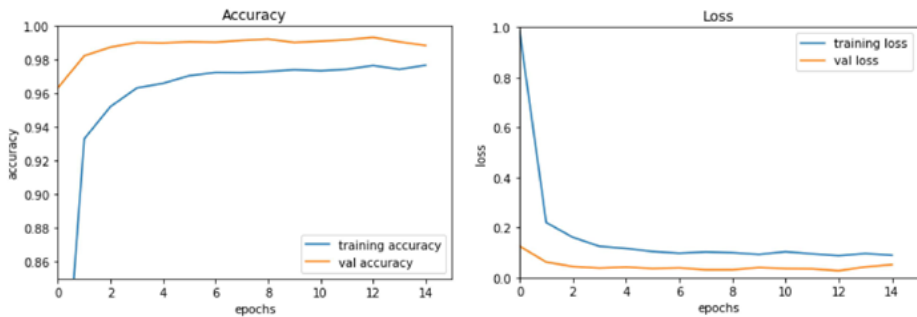


Fig 17. Accuracy and loss for normalized images.

- Accuracy and loss obtained from grayscale images & Loss: 0.1977, accuracy: 0.9432 val_loss:0.0391, val_acc:0.9880
- Accuracy and loss obtained from normalizing the grey scale images as shown in Fig. 18 and Fig. 19 respectively. Loss: 0.0593, accuracy: 0.9835 val_loss:0.0213, val_acc:0.9939

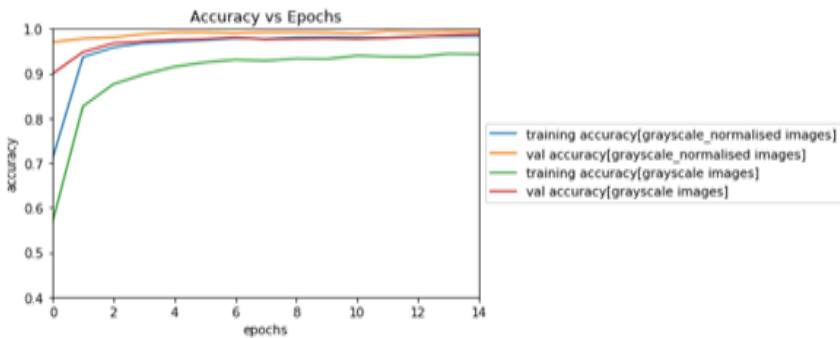


Fig.18. Accuracy of normalized grey scaled images.

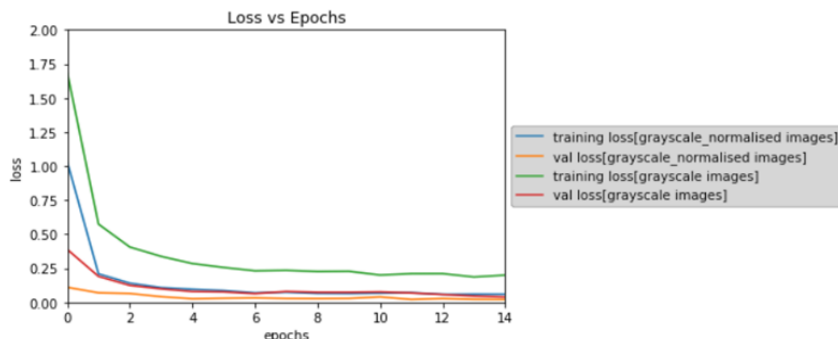


Fig. 19. loss of normalized grey scale images.

Table 1. depicts the trained model and test accuracy data.

Table 1. Trained model and test accuracy.

Model trained	Test Accuracy
Coloured images	0.959144893111639
Normalized images	0.965795724465582
Greyscale images	0.9542359461599367
Normalized grey scale images	0.9697545526524148

Table 2. Depicts the training validation accuracy and loss.

Table 2. Training validation accuracy and loss.

Model trained	Loss	Accuracy	val_loss	val_accuracy
Coloured images	0.2093	0.9514	0.0411	0.9888
Normalized images	0.0892	0.9768	0.0513	0.9885
Greyscale images	0.1977	0.9432	0.0391	0.9880
Normalized grey scale images	0.0593	0.9835	0.0213	0.9939

6 Conclusion & Future Work

In order to improve traffic safety, it is necessary to conduct an in-depth study on the detection algorithms of traffic signs. As this work is research based, exhaustive literature surveys and came across many algorithms to solve this problem statement. Not only is a

literature survey, data also collected to solve the problem statement. For traffic sign detection and recognition, the standard dataset GTSRB dataset is used. The dataset is collected for kannada text recognition also and the chars74k dataset is used for character recognition. For traffic sign recognition, an accuracy of 98% is obtained on the GTSRB test dataset, also, able to correctly recognize the traffic sign on a few real-world samples. The accuracy of above 90% is obtained for kannada test recognition on both test and train dataset and able to identify the small variations on kannada characters.

In traffic sign detection, In this work, the sign board detection is carried out using the red color property of the traffic sign, but this approach fails when other big red objects in the image is detected. For kannada character recognition, this model is able to recognize printed kannada characters efficiently but it struggles in case of handwritten kannada characters, so further work on recognizing handwritten kannada characters. Further work on recognizing the traffic boards in real time will help the drivers immensely.

References

1. Hee Seok Lee and Kang Kim 2018 *IEEE Transactions on Intelligent Transportation Systems* **19** 1652-63F. De Lillo, F. Cecconi, G. Lacorata, A. Vulpiani, EPL, **84** (2008)
2. Yi Yang, Hengliang Luo, Huarong Xu and Fuchao Wu 2016 *IEEE Transactions on Intelligent Transportation Systems* 17 2022-31
3. Md. Safaet Hossain, Zakir Hyder, I. S. Jacobs and C. P. Bean 2015 *International Journal of Computer Applications* 120 10-15
4. L. Fletcher, N. Apostoloff, L. Petersson and A. Zelinsky 2003 *IEEE Intell. Syst.* 18 12-17
5. S. Lafuente-Arroyo, P. Gil-Jimnez, R. Maldonado-Bascn, F. Lpez-Ferreras and S. Maldonado-Bascn 2005 *Proc. IEEE Intell. Veh. Symp.* 5 57-62
6. P. Gil-Jimnez, S. Lafuente-Arroyo, H. Gomez-Moreno, F. Lpez-Ferreras and S. Maldonado-Bascn 2005 *Proc. IEEE Intell. Veh. Symp* 60 7-12
7. J. Miura, T. Kanda and Y. Shirai 2000 *Proc. IEEE Intell. Transp. Syst.* 52-7
8. C. Fang, S. Chen and C. Fuh 2003 *IEEE Trans. Veh. Technol* 52 1329-41
9. Tananon Klinkaew, Kriengsak Yothapakdee, Nittaya Kaewsuwan and Tanunchai Boonnuk 2021 9th International Electrical Engineering Congress (IEECON). 393-96
10. Aklima Akter Lima, Md. Mohsin Kabir, Sujoy Chandra Das, Md. Nahid Hasan and M. F Mridha, 2022 *Proceedings of the International Conference on Big Data, IoT, and Machine Learning*, 95 555
11. Jack Greenhalgh and Majid Mirmehdi 2012 *IEEE Trans. Intell. Ttransp. Syst.* 13
12. S. Maldonado-Bascón, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gómez-Moreno and F. López-Ferreras 2007 *IEEE Trans. Intell. Trans. Sys* 8 264-78