Automatic Detection and Recognition of Text in Traffic Sign Boards based on Word Recognizer

Seles Xavier  
PG Student  
Department of Computer Science & Engineering  
Federal Institute of Science and Technology

Reshmi R  
Assistant Professor  
Department of Computer Science & Engineering  
Federal Institute of Science and Technology

Abstract

While driving one passes so many text based - traffic signs that sometimes even important information such as speed limits or no overtaking traffic signs can get lost. The driver can also be sometimes distracted by traffic and does not notice some of the text based traffic signs. The automatic text detection and recognition of traffic sign boards eliminates this issue by automatically detecting and recognizing the text in traffic sign boards. The text based traffic sign boards should be detected by making use of HSV color thresholding. RANSAC algorithm and homography are used to vertically align the text characters and reduce distortion. This is followed by the recognition phase, which interprets the text in the detected traffic boards. Each letter of the text in the traffic sign board is identified and fed into the Optical character recognition (OCR) for interpretation. The recognition phase is further revised using a dictionary of words. Text recognition is also applied on images with noise. Vectorization is performed so that computational time can be reduced.

Keywords: HSV Thresholding, RANSAC, Homography, OCR, Dictionary of Words, Traffic Text Sign Recognition

I. INTRODUCTION

Have you ever been caught speeding because you were unaware of the speed limit? Ever wandered into a one way in the wrong direction? In today’s hectic driving environment, traffic signs can be easy to miss. The technology discussed in this paper is aimed at providing assistance to drivers by reading traffic sign boards and alerting drivers of key information they may have missed on the road.

Traffic sign recognition as an application of character recognition has two parts- Symbol recognition and Text recognition. This paper focuses on the detection and recognition of text on traffic signs. Text on traffic signs carries information that can be very useful to drivers like speed limits, right of way, warnings etc. Recognition of text from road signs can be particularly difficult. Challenges include font variation, orientation, shadows, low image resolution, blurring and other distortion attributed to the motion of the vehicle.

This paper has attempted to develop an algorithm to accurately detect and recognize text from traffic signs boards. It involves firstly the detection of traffic signs by their colour scheme using HSV (Hue saturation value) thresholding. Once the board is detected, the image is oriented favourably using homography for OCR (optical character recognition). The words recognized from the boards are then compared with words from a dictionary to avoid errors arising by blurring, incomplete recognition and false positives.  

The above mentioned technologies are discussed in further detail in the following sections. In section II, literature survey is briefly described. Section III describes the proposed methodology. In section IV presents the experimental results and analysis and section V summarizes the system.

II. RELATED WORKS

In [1] Hanwell et al. proposed a system for detecting and tracking the lanes of roads. This is used to warn the drivers when they change their lanes. It proposes a Hough transformation. The system detects and finds the road marking which are linear and the ones with common vanishing point. Extended Kalman filter is also used to detect and track lanes.

Greenhalgh et al. introduced a system that is used for automatic detection and recognition of the traffic signs[2]. Here candidate regions are detected using maximally stable extremal regions(MSER).Support vector machine classifier are used for recognizing the traffic signs. These classifiers are trained using histogram of oriented gradient features. In this approach all types of traffic signs are detected.

Fischler et al. proposed a new model called Random Sample Consensus (RANSAC) [3].It is a fitting model. It is capable of smoothing data that contain a significant percentage of errors. Here a subset of the original data is taken first. This is called as hypothetical inliners. Then a model is fitted into this hypothetical inliners. The remaining data are tested in this model. The point that fits these models are a part of the consensus data set. This is how RANSAC algorithm is working.

In [4] Gonzalez et al. applied color segmentation to detect the candidate regions. The classification is done using Support Vector Machines or Naïve Bayes. He introduced a new method called visual appearance categorization. He also proposed a
language model. It is based on a dynamic dictionary for a specific geographical area. It is done using reverse geocoding service. The images are taken from the Google Street View.

In [5], Neumann et al. proposed a general method for text localization and recognition in real world images. Here a hypotheses verification framework is introduced. He used various kinds of fonts to train the algorithm. Maximally stable extremal regions are used to detect the candidate regions. It is robust to the illumination changes.

Gonzalez et al. introduced a method to extract the text in the road panels [6]. Using Hidden Markov Models and a Web Map Service, word recognition is performed. Distance between the vehicle and the panel is calculated. Then world coordinates are computed for each panel. Here Web Map service is used to reduce the size of the dictionary. It makes use of the Cartociudad server that gives position of the input image.

Reina presented a method to extract text from the traffic sign boards [7]. To detect the rectangular panels, color segmentation and shape classification is done. Then it is reoriented. Chrominance and histogram analysis and adaptive segmentation is performed. Afterwards the connected components are labelled. Finally position labelling is done for arranging each letter of the traffic sign board.

Gonzalez introduced a system based on a automatic inspection system called VISUAL Inspection of Signs and panElS (VISUALISE) [8]. It is fixed on the vehicle. This will perform various tasks so that the inspection can be done. It is done at various speeds. It is used to classify sign or panel to a specific class by making use of the luminance measurements.

III. PROPOSED METHODOLOGY

In the automatic text recognition of the traffic sign board, there are mainly two stages. They are the detection of the traffic sign board and the recognition of the text from the detected traffic sign board. The Fig.1 shows the block diagram of the entire system.

![Block Diagram of Detection and Recognition of Text in Traffic Sign Board](image)

**A. Detection of Text-Based Traffic Signs**

The first stage in the automatic detection and recognition of text-based traffic sign board is the detection of the traffic sign board from the input image. There are three steps involved in this stage. They are edge detection, Hough transformation, finding the candidate region from the provided input image and the detection of text-based traffic sign board.

In order to determine the text-based traffic signs boards, the first step is to find the edges of the input image. It is done by the Canny Edge detector. Next, the Hough transformation should be applied. It is used to find the straight lines from the edges found out from the Canny Edge detector.

There are three search regions where traffic sign boards can be found. They are one to the left-hand side of the road, one to the right, and one above. The top search region is defined to be the width of the road, given that overhead gantries, which appear in this region, never extend beyond the sides of the road. In this project all the text-based traffic signs are detected, despite of these three. For that, all the hough transformed straight lines are taken. The end points of these hough lines are considered. Then find the least point from these end points. The candidate region wills the region from the top to that very point. This is how the candidate region is being detected. Most of the unnecessary regions are being removed by making use of this technique.

Next, the text-based traffic sign boards should be detected from the obtained candidate regions. Text traffic sign candidates can be detected using HSV color thresholding. Firstly, all the frames are converted into the HSV color space. Then a threshold is applied to hue and saturation channels. The value channel is not changed so that the system will be invariant to changes in the brightness. [8]

Threshold values are determined using the images and videos provided in the dataset of this work. The dataset contains traffic sign panels which are green in color. And the writing on the panel is white. The hue and saturation range of green color should be set before performing the HSV color thresholding. The hue range for green color is from 110 to 135. And the saturation value should be above 50%. Thus the text-based traffic sign candidates are found by HSV thresholding to detect candidate regions for green traffic signs panels.
After performing the HSV color thresholding, some operations like masking, dilating, closing and filling the image is done. Then the area of all filled portions is considered. The maximum area from those are taken. That portion will contain the text-based traffic sign board. Finally, the traffic panel is cropped out from the image. In the next stage, text in traffic panel is recognized.

### B. Recognition of Text from the Traffic Sign Panel

Recognition of text within the detected candidate regions is found out next. First apply an approximate perspective transform to the rectangular candidate regions to vertically align them and their text characters. Each letter of the traffic sign board is then recognized and then sent to OCR.

1) **Correction of Traffic Sign Board**

After the detection of the text based traffic sign board, the recognition should be done. For recognizing the text in the traffic sign board, firstly check if the text in the traffic sign boards is aligned vertically. If not, it should be vertically aligned by making use of Random Sampling Consensus (RANSAC) algorithm and homography. RANSAC algorithm is used to find the lines that show the top and bottom edges. After doing the RANSAC algorithm, homography should be done so that the text in the traffic sign board can be vertically aligned. Example of this result can be seen in Fig. 2.

![Fig. 2: Correction using Homography](image)

2) **OCR Recognition**

OCR is used for recognizing the text in the traffic sign panels. OCR recognizes all the alphabets and numbers in the traffic sign boards. Alphabets include both the lower and upper case letters. The input for the OCR should be a binary image. The input of the OCR should be categorized in two ways. First is to detect all possible letters in the traffic sign board using the blob detection method. The other way is to directly feed the traffic sign board to OCR. Blob detection is done by making use of many features like area, centroid etc. In this system, both the techniques are done. And the better results are taken, so that the accuracy of recognition can be improved.

3) **Word Recognizer**

The automatic text detection and recognition proposes a language model based on a dictionary. This dictionary contains all the words in the traffic sign boards. Each letter from the OCR is compared with the words in the dictionary. The best matched word is considered as the output. This method recognizes the text in the traffic sign boards more accurately than other systems by making use of the dictionary. This system can be limited to a particular geographical area.

Next text detection and recognition of noisy traffic sign board is taken into consideration. First of all the noise is being removed from the system. After that, the detection and recognition is done. This is to ensure that the noisy inputs will be working well with the proposed system.

Vectorization is also done on the entire code such that the output should reduce the time taken.

### IV. EXPERIMENTAL RESULTS AND ANALYSIS

This Automatic Detection and Recognition of Text in Traffic Sign Boards Based on Word Recognizer was implemented using MATLAB 2009 on Intel i3, 1.70GHz processor with 4GB RAM, running on Windows 8 operating system. The data base was built with images and videos from Google and from the roads of Thrissur, India. All the images are having green background and the text is white in color. The example demonstrates both the detection and recognition of the text from the traffic sign panels. In preprocessing first we convert the input image (Fig. 3) into grayscale image to reduce the complexity.
Next apply the Canny edge detector so that all the edges can be extracted from it. The output by applying the Canny edge detector is shown in Fig. 4.

Next the hough transformation is done on the output image that is obtained from the Canny edge detector. That image will be a binary image. Hough transformation is done to obtain the straight lines from the edges obtained from the Canny edge detector. The output of Hough transformation is given in the Fig.5.
The candidate regions are then extracted from these Hough lines obtained. The portion from the top of the image to the least endpoint of the Hough line is the expected candidate region. This is shown in the Fig.6.

![Candidate Region](image)

**Fig. 6: Candidate Region**

Next the text based traffic sign should be extracted from this candidate region. And it is done by making use of the HSV color thresholding technique. The output after applying the HSV color thresholding is given in the Fig7.

![HSV Color Thresholding](image)

**Fig. 7: HSV Color Thresholding**

Now some operations should be done so as to detect the text-based traffic sign boards. Firstly, masking is done. That is the images detected using HSV color thresholding will be white and all others black. Thus a binary image. This is shown in the Fig8.

![Masked Image](image)

**Fig. 8: Masked Image**

After masking, the small objects in the image should be removed using the bwareopen function. And the output this is shown in the Fig.9.
Afterwards, the image has to dilated. That is the white region should be dilated. So that the traffic sign board should be easily detected. This is shown in the Fig.10.

Closing of the image is done using the close function. This is shown in Fig.11.

After closing the image, the image should be filled. Fig.12 shows the filled image that is obtained using the fill function.
The area of all the filled portions is taken into consideration. And the region with maximum area is taken as the candidate region. This region is cropped from the candidate region as in Fig.13.

![Sign Board Detection](image)

Fig. 13: Sign Board Detection

After detecting the text based traffic sign board from the input image. Next recognition of text in the traffic sign boards should be obtained. For that check if the detected traffic sign board is aligned properly. If not, an approximate transform is applied to vertically align the text characters and to reduce perspective distortion. RANSAC algorithm is performed to estimate the lines representing the bottom and top edges. The RANSAC algorithm is applied as in Fig.14

![Applying RANSAC Algorithm](image)

Fig. 14: Applying RANSAC Algorithm

After applying the RANSAC algorithm, the homography transformation should be done so as to correct the alignment of the image. The output after applying the homography is shown in the Fig.15.

![Corrected using Homography](image)

Fig. 15: Corrected using Homography

Next the image is converted into binary image. So that the blob detection can be easily done. Here there are two cases. First one is to detect all the blobs for each individual characters in the image and fed into the OCR as in Fig.16. The other is to directly feed in to the OCR.
The results from the OCR are compared with the dictionary of words and then the best match is obtained. The final output is shown in Fig.17.

To measure the performance of the proposed method three metrics were used: precision, recall and $F_{\text{measure}}$. They are defined below.

$$\text{Precision} = \frac{TP}{TP + FP}$$
$$\text{Recall} = \frac{TP}{TP + FN}$$
$$F_{\text{measure}} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

In the equations, 
TP stands for True Positive (correctly detected),
FP stands for False Positive (incorrectly detected),
TN stands for True Negative (correctly not detected)
FN stands for False Negative (incorrectly not detected).

While considering the detection stage, TPs are correctly detected text-based traffic signs, FPs are regions incorrectly identified as text-based traffic signs, and false negatives (FNs) are text-based traffic signs that the system failed to detect.

The Table 1 shows the Precision, Recall and $F_{\text{measure}}$ of the proposed system and the previous works. It can be seen from these results that the proposed method achieves an $F_{\text{measure}}$ of 0.947, whereas Gonzalez et al. [6] and Reina et al. [7] reach considerably lower values of 0.60 and 0.61 respectively. From this it is clearly seen that all the three performance parameters shows high values for the proposed system.

<table>
<thead>
<tr>
<th>Method</th>
<th>Precision</th>
<th>Recall</th>
<th>$F_{\text{measure}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reina et al.</td>
<td>0.58</td>
<td>0.64</td>
<td>0.61</td>
</tr>
<tr>
<td>Gonzalez et al.</td>
<td>0.54</td>
<td>0.68</td>
<td>0.60</td>
</tr>
<tr>
<td>Proposed System</td>
<td>0.9</td>
<td>0.95</td>
<td>0.94</td>
</tr>
</tbody>
</table>

To evaluate the performance of the recognition stage, Precision, Recall and $F_{\text{measure}}$ were computed based on the number of individual words correctly classified. For a word to be considered a TP, all characters must be correctly recognized in the correct letter case. If a single character is recognized incorrectly, then the entire word is considered to be a FP. Symbols such as 'arrow mark' when recognized have no effect on the result. And therefore classified as true negatives (TNs). The precision for recognition is 0.96. The recall parameter shows 0.96 for recognition of text. The $F_{\text{measure}}$ shows 0.952.
The vectorization is done so that the time complexity is reduced considerably. Image code that is vectorized is shown in Table 2 and on video code is in Table 3. In both these tables time taken for the non-vectorized code and vectorized code is considered. From these tables, it is clear that the computational time is decreased after performing the vectorization.

<table>
<thead>
<tr>
<th>Image</th>
<th>Non-Vectorized Code (Time in sec)</th>
<th>Vectorized Code (Time in sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image01</td>
<td>253.74</td>
<td>55.76</td>
</tr>
<tr>
<td>Image02</td>
<td>9.68</td>
<td>7.537</td>
</tr>
<tr>
<td>Image03</td>
<td>419.27</td>
<td>94.91</td>
</tr>
<tr>
<td>Image04</td>
<td>398.04</td>
<td>102.61</td>
</tr>
<tr>
<td>Image05</td>
<td>88.39</td>
<td>29.73</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Video</th>
<th>Non-Vectorized Code (Time in sec)</th>
<th>Vectorized Code (Time in sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Video01</td>
<td>251.280</td>
<td>248.955</td>
</tr>
<tr>
<td>Video02</td>
<td>266.62</td>
<td>261.702</td>
</tr>
<tr>
<td>Video03</td>
<td>123.066</td>
<td>112.84</td>
</tr>
</tbody>
</table>

In short, it is clear that both the detection and recognition shows better results for the proposed system. By making use of vectorization the time consumption is also reduced.

V. CONCLUSION

The work proposes an automatic detection and recognition of text in traffic sign boards which is based on a word recognizer. Traffic sign boards are detected using HSV color thresholding. RANSAC algorithm and homography are used to vertically align the text characters. In the recognition phase, it interprets the text in the detected traffic boards. Each letter of the text in the traffic sign board is identified and fed into the Optical character recognition (OCR) for interpretation. Recognition accuracy is vastly improved using a dictionary of words. Vectorization is performed so that computational time can be reduced. Both the detection and recognition stages of the system were validated through comparative analysis achieving the F measure of 0.947 for detection, 0.952 for recognition.

REFERENCES