A Method of US Traffic Sign Detection and Recognition

Steven Haar and Mejdl S. Safran
Department of Computer Science
Southern Illinois University Carbondale
Carbondale, Illinois 62901 USA
info@stevehaar.com, mejdl.safran@siu.edu

Abstract - *A key issue in designing both autonomous vehicles and driver support systems is how to detect and recognize the traffic rules. One of these rules is the traffic signs. In this paper, we proposed a method for detecting and recognizing the United States regulatory traffic signs (stop sign, no-left-turn sign, no-right-turn sign, do-not-enter sign, and yield sign) based on color segmentation and shape analysis in real street-view images. Images that are taken as inputs by the proposed method are processed through three main phases: color segmentation, shape classification, and recognition of traffic signs. For this study, HSV color model is adopted for color segmentation which gives more accurate results even with high-lights and illumination changes. Shape signatures are used for shape classification. A ration matching technique using a decision tree is applied for the recognition phase. The proposed method is tested on real street-view images taken from Google Maps. Experimental results show that our approach is valid and can be applied for real-time applications.*

Keywords: *segmentation, image processing, traffic signs, shape detection, shape recognition.*

I. Introduction

Humans in general are not concentrated all the time while they are driving their vehicles or at least their driving attention sometimes is distracted. Additionally, since autonomous vehicles with 100 percent accuracy are not invented yet, the use of driver support systems [1] has become important to potentially improve safety by helping drivers to react to the road conditions changes. One of the important tasks of driver support systems is traffic sign recognition (TSR) which has become a research area and gained attentions widely over the world [2][3]. During vehicle travelling, TSR can detect and recognize the meaning of the traffic signs and inform drivers to take response in advance which results in reducing traffic accidents and deaths. Sign detection and sign recognition are the main tasks of TSR. Both of them are discussed in this paper.

This paper proposed an algorithm for regulatory traffic sign detection and recognition. The regulatory traffic signs that the proposed algorithm detects and recognizes are stop sign, no-left-turn sign, no-right-turn sign, do-not-enter sign, and yield sign. The common factor among these traffic signs is the red color where either the red color is the predominate color or the boundaries are red. To detect and recognize traffic signs using the proposed algorithm, real street-view images are processed through three main phases: color segmentation, shape classification, and recognition of traffic signs.

Since the pixels values in the RGB color model are highly dependent on the illumination levels in the scene and the RGB color model does not provide a better control over the variations in pixel values for the same color, the HSV color model is adopted for color segmentation [4]. Shape signatures of the traffic signs are used for classifying shapes of candidate signs [5]. A ration matching technique using a decision tree is employed to recognize the detected traffic signs [4] [6].

The rest of the paper is organized as follows. Section II reviews related work. Section III describes the proposed approach to detect and recognize the US regulatory traffic signs in real street-view images. Section IV shows our experimental results, and finally Section V concludes our work.

II. Related Work

Many approaches and techniques for traffic signs detection and recognition have been developed in the literature [3] [4] [6-15]. The literature of traffic sign detection and recognition can be divided into two groups: traffic sign detection and traffic sign recognition. Traffic sign detection generally extracts the candidate traffic sign using color segmentation by employing one of the color models such as HSV and RGB. One of the popular segmentation algorithms is thresholding which results in binary image where pixels that are above threshold values are treated as background (black pixels) and others are white pixels [9]. Instead of using color information, shape based approaches are used in segmentation of traffic sign. Also, it is found in the literature that color information is used as a preprocessor step. Shape based approaches usually use geometric properties of different traffic sign shapes [13], Hopfield neural networks [14], or Fast-Fourier transform to retrieve and compare shape signatures [15].

Traffic sign recognition generally usually uses a template matching techniques [10], neural networks [11][14], Support Vector Machine Gaussian kernel [12], Fuzzy shape recognizer [13], or color ration matching [4].

Systems proposed in the literature vary from each other but try to solve the same problem. Some limited their scope of detection and recognition to a specific group of traffic signs and some presented general approaches. Moreover, some limited their scope for only traffic sign detection while others presented complete approaches for detection and recognition. The proposed method in this paper is a complete approach for a specific group of US traffic signs. However, the scope of the proposed method can be easily extended by taking the other color components and following the same steps of the proposed method.

III. Proposed Approach

The traffic signs that are used in this study are shown in Figure 1. As seen in Figure 1, there are three different shapes: circle, octagon, and upside down triangle. In this section, we discuss the proposed method which consists of three main phases illustrated in Figure 2 and described as follows:

1) *Color Segmentation*: Since we limit the method to traffic signs that either have red color as the predominate color or have red color on the boundaries, regions with red color are the regions of interest. This phase takes an RGB image as input. The output of this phase is a binary image where the regions of interest have white pixels and all other pixels are treated as the background (black pixels). Noise regions are removed before the output image is sent to the next phase. This phase is represented by the first and second components in Figure 2.

2) *Shape Classification*: Labeling and bounding the connected regions, reducing the candidate traffic signs, filling the candidates and finding their edges are the techniques that are applied on the output image from the Phase 1 to be prepared to shape classifier (to be discussed in detail). To classify the shapes in the bounding boxes of the binary image as circle, octagon, or triangle, shape signatures of the traffic signs are used to compare them with the shapes in the bounding boxes [5]. The outputs of this phase are percentages of the shapes in the bounding boxes of being circle, octagon, or triangle. This phase is represented by components 3, 4, 5, and 6 in Figure 2.

3) *Traffic Sign Recognition*: The inputs of this phase are bounding boxes with their percentages of being circle, octagon, or upside down triangle after applying thresholding technique. A decision tree using color ration matching techniques [4] is used to recognize the traffic sign of the bounding boxes as one of the traffic signs in Figure 1 using the white and black components of the same locations of the bounding boxes derived from the original image. This phase is represented by the last component, 7, in Figure 2.

Figure 1: Traffic signs used in this paper

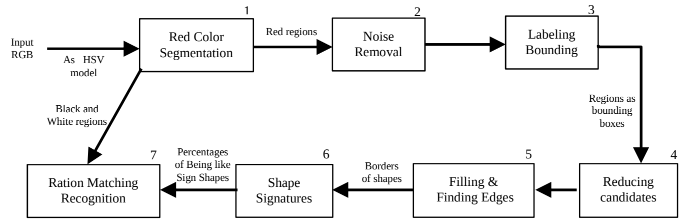

Figure 2: The proposed method

A. *Color Segmentation*

Color segmentation phase consists of two steps which are shown as the first and second components in Figure 1 and illustrated as follows:

1) *Red color segmentation*

Using the HSV color model, a red-like color can be defined by two parameters: first, the hue must be near the red component on the color wheel; second, the color must be sufficiently saturated as to no longer be considered gray. These two parameters, hue and saturation, are used to convert the input image into a binary image consisting of only the red-like pixels.

The HSV color model is used in which the hue is represented as an angle with 0° representing red, 120° representing green and 240° representing blue. The image is filtered to only allow pixels of a hue within 30° of true red to remain. Thus the condition is that the hue must be greater than 330° or less than 30°.

Depending on the time of day and ambient lighting conditions, different photographs will have different levels of saturation. For this reason, a dynamic threshold for the saturation is chosen which means the threshold changes with each image. For this dynamic threshold we use ω where $PDF(\omega) = 0.7$ and $PDF(x)$ is the probability density function over the image. Therefore, combining the conditions of the two used parameters, hue and saturation, gives the following condition:

$$Hue > 330° \ OR \ Hue < 30° \ AND \ Saturation > \ \omega$$

Filtering on hue and saturation as explained above, the result is a binary image that contains only pixels with a red hue that are sufficiently saturated compared to the rest of the image.

2) *Noise removal*

After the filtering step discussed above, the image often contains small regions of pixels that are too small or too sparse to be traffic sign. These pixels are considered as noise and an erosion technique is applied to eliminate them from the binary image. Sliding a 5x5 mask about the image pixel by pixel we find all instances in which the outer 16 pixels of the mask are black and at the same time the center pixel is white. When this occurs, all of the nine inner pixels of the mask are set to black. The result is that any regions of pixels with a height and width of less than four will be removed from the image.

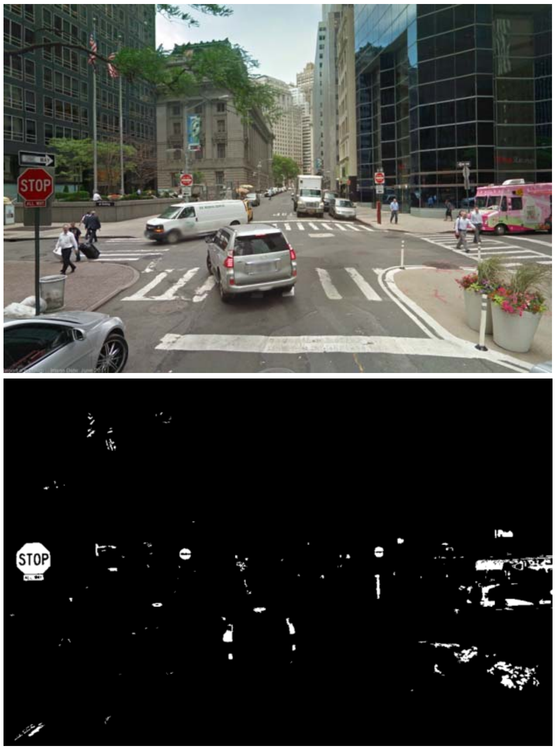

Figure 3: Input to color segmentation and output of color segmentation (bottom)

Figure 3 shows an input image, as example, to Phase 1 and the binary image resulted after applying the color segmentation and noise removal, as discussed above. The resulted image is now ready to be sent as input image to the next phase which is shape classification.

B. *Shape Classification*

Shape classification phase is represented by components 3, 4, 5 and 6 in Figure 2 and illustrated as follows.

1) *Labeling and bounding the connected regions*

A labeling approach that assigns a label to every white pixel in the binary image is employed [16]. Pixels connected by one of their eight nearest neighbors will have the same label. Thus each group of pixels with the same label represents a distinct region in the image. At this point the input binary image is segmented into a number of different regions. We find a bounding box for each region which is the smallest square that contains every pixel in the region. Therefore, the results of this step are bounding boxes containing the regions of interest.

2) *Reducing and combining regions*

The number of regions can be reduced by eliminating any regions whose bounding box is completely contained inside another bounding box. Making an assumption allows us to further reduce the number of regions. The assumption is that the traffic sign in the photograph would not take up a very large or a very small portion of the image. Therefore, any regions whose bounding box has a width that is smaller than $\frac{1}{85}$ of the width of the image or larger than $\frac{1}{10}$ of the width of the image is eliminated.

One problem that was realized is that the image segmentation would often leave the top and bottom halves of do-not-enter traffic signs disconnected. This is not a surprise because the do-not-enter sign has a large horizontal white stripe that runs through the center of the sign. To resolve this problem, this step finds and combines any bounding boxes that are horizontal rectangles of nearly the same size where one is directly above the other.

Finally, we further reduce the regions by eliminating any regions whose width to height ratio is too far from one. Specifically, we eliminate any regions whose bounding box matches the following condition:

$$|Width - Height| \geq 0.3\ Max(Width, Height)$$

Figure 4 shows the resulted image after applying the two steps that are discussed above (labeling, bounding, reducing and combining regions). Only 11 regions out of 222 connected regions are left to be processed in the next step.

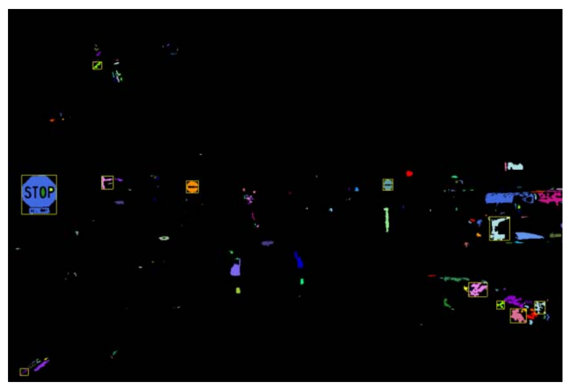

Figure 4: Labeling, bounding, reducing and combing regions

3) *Filling the candidate shapes*

In an effort to combat noise in the image and to reconstruct the original shape, we fill in the regions by performing a horizontal (vertical) scan of each row (column) of pixels in the image. We note the first and the last white pixel in the scan and set all other pixels that lie in between these two pixels as white.

4) *Shape signatures*

To recognize the shape of each of the regions, a method that extracts a signature from the shape and compares it to a database of known shape signatures is employed. Traffic signs in images can vary in size but will always have the same shape, and furthermore will always have the same rotational orientation. For instance, when detecting a yield sign we are only interested in the upside triangle and never the rotated, right side up triangle. For these reasons, a signature method that is size-invariant but not rotation-invariant is used.

An angle vs. distance signature method is employed. The signature is taken from the border of the shape which is extracted using a border following algorithm [17]. To take the signature, first the centroid of the region is found and then a set of points (θ, y) along the function $f(\theta) = y$ where θ is the angle and y is the distance of the line from the centroid to the border of the region along the angle θ is constructed [5]. These set of points at a specific interval will represent the signature of the function. Because the angle θ is always defined as the angle around the unit circle between 0 and 2π, the shape signature will not be rotation-invariant. To make the signature size-invariant, all the y values are normalized to values between 0 and 1.

After extracting the shape signature, the signature is compared to six known signatures in our database, Figure 5. Three of the signatures come from a perfect circle, octagon, and an upside down triangle. In some of the images the photograph was taken at a horizontal angle to the traffic sign rather than head on. The result is that the traffic sign's shape border appears to be stretched vertically. Such shape borders result in signatures that are not so similar to the perfect shapes in the database. Therefore, to overcome this issue, the elongated, or vertically stretched shapes, are also included as part of our database.

Figure 5: Shape used to populate the shape signature database. The bottom row consists of the elongated shapes.

The signature in question is compared to the known signatures one at a time by taking the sum of the square of the differences of the points in the signatures. We then select the shape corresponding to the known signature with the least sum of squared differences and assign a percentage to the signature and shape pair. The percentage is derived from the maximum possible sum of squared differences and represents the likelihood that the shape is correctly classified. Regions with a percentage of less than 80% are removed from the results before being passed on to the next phase.

Figure 6 shows the resulted image after filling the candidate shapes, following the shapes' boarders and comparing the shape signatures. Only three shapes out of 11 satisfy the aforementioned condition of the shape signature. The bounding boxes whose shapes satisfy the condition are sent to the recognition step which is discussed next.

Figure 6: Shape classification result

C. Traffic Sign Recognition

1) Red, white and black components

The five traffic signs in Figure 1 contain only three colors: red, white and black. We analyze these three components from the image to classify the region as one of the five street signs. For the red component of the image we use the previously defined red component resulting from the color segmentation phase. We define the black component as the pixels not contained in the red component whose value or brightness, according to the HSV color model, is less than 0.2 on a 1.0 scale. Consequently, the white component is defined as any pixel not in the red or white component, thus having a brightness greater than 0.8.

2) Sign Recognition through a Decision Tree

After the red, white and black components are obtained, a decision tree is used to classify each region as one of the five street signs. The first branch in the decision tree looks at the detected shape. If the detected shape is the upside down triangle, then the region is identified as a yield sign. Otherwise we progress to the second stage of the decision tree.

At this stage, the four remaining signs are divided into two groups, A and B. Group A consists of the stop sign and the do-not-enter sign. These signs are mostly red and contain no black. Group B consists of the no-left-turn and no-right-turn signs. These signs are mostly white and do contain some black. To determine which group the suspect region belongs to and to progress down the decision tree, we ask if the white component of the region is greater than 50% or if the black component of the region is greater than 10%. If either of these is true, then we classify the suspect region as belonging to Group B.

However, if both conditions are false, we classify the region as belonging to Group A.

If the region is determined to belong to Group A, then we must decide if it is either a stop sign or a do-not-enter sign. In order to do this, we exploit the fact that the do-not-enter sign has a large horizontal white stripe running through its center. We examine the pixels in the center row of the region and if at least 70% of the pixels belong to the white component, then we classify the region as a do-not-enter sign, otherwise we classify the region as a stop sign.

If the region is determined to belong to Group B, then we must decide if it is either a no-left-turn sign or a no-right-turn sign. In order to do this, we examine the lower left quadrant of the region. If the sign is a no-right-turn sign, there should be black pixels in this quadrant, while if the sign is a no-left-turn sign, there should not be any black pixels in this quadrant. Thus, we examine all of the pixels in the lower left quadrant and compare the number of pixels belonging to the white component to the number of pixels belonging to the black component. If there are more black pixels than white, then we classify the region as a no-right-turn sign, otherwise we classify the region as a no-left-turn sign.

Figure 7 concludes the decision tree discussed above. Figure 8 shows the results of applying the decision tree on the output image from shape classification phase which is shown in Figure 6.

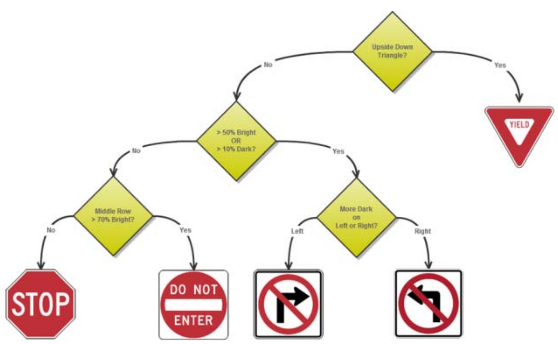

Figure 7: The decision tree

Figure 8: Traffic sign recognition result

IV. Experimental Results

To test the validity of our approach and the ability of being a real-time system, we used Google Maps to collect 92 real-time street view images containing 151 traffic signs that are considered in this study. While we were collecting the image, we simulated the reality of taking images from an installed camera on a vehicle for the same purpose. Moreover, we built a GUI application using C# to take an image as input (Figure 3, top) and show all the steps done on the image to get the resulted image in Figure 8. A screenshot of the developed application is shown in Figure 9. This application is more educational than practical since it is built to demonstrate all the steps of the proposed method.

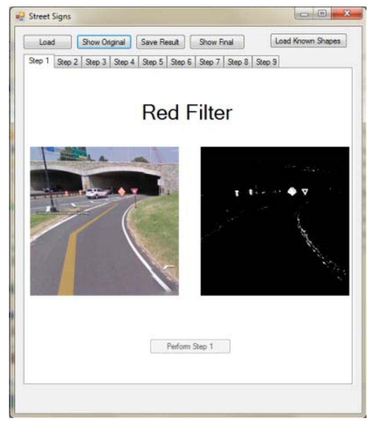

Figure 9: The application

Table 1 shows the experimental results of the proposed method. The third column, “Correctly detected”, means the number of signs that are correctly detected by the shape classification regardless the results of the recognition phase. The last column, “Correctly detected & recognized”, means the number of signs that are correctly detected by the shape classification and also correctly recognized by the recognition phase.

Not all the images used in this experiment have high quality colors in the regions of the traffic signs. However, some of the images used in the experiment have low quality colors which cause the failures in sign recognition phase (to be discussed in detail). Consequently, this is why the recognition results, last column, are lower than the detection results, the third column.

Table 1: Experimental results

Sign	Number of signs	Correctly detected	Correctly detected & recognized
Stop	41	34 (83%)	27 (66%)
Yield	10	8 (80%)	8 (80%)
Do Not Enter	39	34 (87%)	28 (72%)
No Left Turn	47	39 (83%)	28 (60%)
No Right Turn	14	12 (86%)	7 (50%)

The failures of our method can occur in any one of the three different phases of our method. Therefore, the failures occur in each phase can be discussed separately as follows.

1) Failures in color segmentation

One of the failures in color segmentation is caused by the red background of the traffic sign. If the background of the traffic sign is red, then the red component will have the traffic sign connected to the background. Therefore, the traffic sign and the red background will be assumed as one connected region. Figure 10 shows an example of detection failure caused by red background.

Figure 10: Failure of red background

Another type of failures in color segmentation is caused by the darker color of the traffic sign. Pixels' colors of some of the traffic signs are not really red; however, they are analyzed as more dark red pixels (it is considered as black more than it is considered red). Therefore, the quality of the traffic signs plays an important role in building high quality system for traffic sign detection and recognition. Figure 11 shows an example of low quality no-left-turn sign that was not filtered from the color segmentation phase.

2) Failures in shape classification

A failure in shape classification is caused by traffic signs whose shapes are not complete. Some of the traffic signs are overlapped with other objects in the road environment. Therefore, the shapes of these traffic signs would not satisfy the condition of the shape recognition to be correctly classified as one of the shape in the database as shown in Figure 5. Figure 12 shows an example of broken octagon shape (the right stop sign). The percentage of this shape of being an octagon is 78%, so the right stop sign will be removed from the candidate signs since it does not satisfy the condition of the shape signature.

Figure 11: Failure of low quality colors

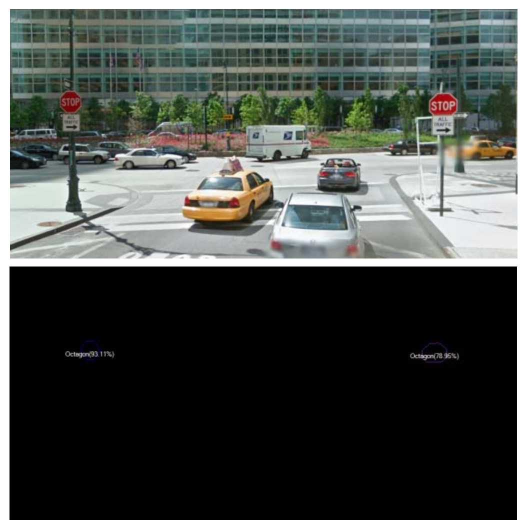

Figure 12: Failure of broken octagon

3) Failures in traffic sign recognition

All the failures in this phase are caused by the low quality of the colors in the traffic signs which control the number of the white, black and red pixels in the traffic sign. The failures in this phase are only recognition failures. For example, a do-not-enter is recognized as a stop sign. Figure 13 shows an example of low quality stop sign (the left sign) recognized as a do-not-enter sign.

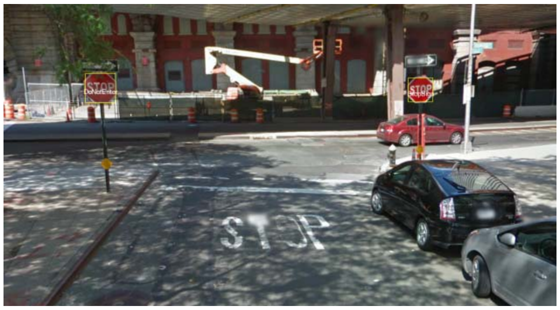

Figure 13: Failure of low-quality colors

V. Conclusion and Future Work

A method for detecting and recognizing the United States regulatory traffic signs in real street-view images is proposed. The regulatory traffic signs that are considered in this study are stop sign, no-left-turn sign, no-right-turn sign, do-not-enter sign, and yield sign. The proposed method is based on color segmentation and shape analysis. The system of the proposed method is divided into three main phases: color segmentation, shape classification, and recognition of traffic signs. A thresholding technique is used for color segmentation on the HSV color model. Shape signatures are used for shape classification. A ration matching technique using a decision tree is applied for the recognition phase. The method is tested using real street-view image taken from Google Maps. The experimental results show that our approach is valid and efficient for good and high quality colors in the regions of the traffic signs. The method needs some improvements to deal with the low quality colors in the regions of interest.

As future work, we plan to use an edge detector in the color segmentation phase to overcome the red background failures, to do improvements on the shape classification phase, especially on the shape signatures step and to include other traffic signs in addition to those signs we considered in this study.

References

[1] Road Sign Recognition Survey,
URL: http://euler.fd.cvut.cz/research/rs2/

[2] V. Kastrinak, M. Zervakis, K. Kalaitzakis. A Survey of Video Processing Techniques for Traffic Applications. Image and Vision Computing, vol. 21, no. 4, pp. 359-381, 2003.

[3] P. Shenghui, Z. Fan, L. Menghe, K. Baozhong. A Method of Traffic Sign Detecting Based on Color Similarity. In Proceedings of 3th International Conference on Measuring Technology and Mechatronics Automation *(ICMTMA)*, no. 1, pp. 123-126, 2011.

[4] M. Zadeh, T. Kasvand and C. Suen. Localization and Recognition of Traffic Signs for Automated Vehicle Control Systems. In Proceedings of SPIE'S Intelligent System and Automated Manufacturing, vol. 3207, no. 272, 1998.

[5] R. Gonzalez, R. Woods. *Digital Image processing*. 3rd, 2008. Prentice Hall.

[6] S. Sallah, F. Hussin, M. Yusoff. Road Sign Detection and Recognition System for Real-Time Embedded Applications. In Proceedings of the International Conference on Electrical, Control and Computer Engineering (INECCE), pp. 213-218, 2011.

[7] W. Kuo and C. Lin, Two-Stage Road Sign Detection and Recognition. IEEE International Conference on Multimedia and Expo, Beijing, pp. 1427-1430, 2007.

[8] L. E. Moreno and M. A. Salichs, J. M. Armingol. Road Traffic Sign Detection and Classification. IEEE Transactions on Industrial Electronics, vol. 44, pp. 848-859, 1997.

[9] U. Zakir, A. N. J. Leonce , E. A. Edirisinghe. Road Sign Segmentation Based on Colour Spaces: A Comparative Study. In Proceedings of the 11th Iasted International Conference on Computer Graphics and Imgaing, Innsbruck, Austria, 2010.

[10] R. Malik, J. Khurshid, S. N. Ahmad. Road Sign Detection and Recognition Using Colour Segmentation, Shape Analysis and Template Matching. In Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, Hong Kong, vol. 6, pp. 3556-3560, 2007.

[11] H. Ohara, I. Nishikawa, S. Miki, N. Yabuki. Detection and Recognition of Road Signs Using Simple Layered Neural Networks. Proceedings of the 9th International Conference on Neural Information Processing, vol. 2, pp. 626–630, 2002.

[12] S. Lafuente-Arroyo, S. Maldonado-Bascon, P. Gil-Jimenez, J. Acevedo-Rodriguez, R. Lopez-Sastre. A Tracking System for Automated Inventory of Road Signs. IEEE Intelligent Vehicles Symposium, pp.166-171, 2007

[13] H. Fleyeh. Traffic Sign Recognition by Fuzzy Sets. IEEE Intelligent Vehicles Symposium, Eindhoven University of Technology Eindhoven, Netherlands, PP. 422–427, 2008.

[14] G. H. Kim, H. G. Sohn, Y. S. Song. Road Infrastructure Data Acquisition Using a Vehicle-Based Mobile Mapping System. Computer-Aided Civil and Infrastructure Engineering, vol. 21, no. 5, pp. 346–356, 2006.

[15] P. Gil-Jimenez, S. Lafuente-Arroyo, H. Gomez-Moreno, F. Lopez-Ferreras, S. Maldonado-Bascon. Traffic Sign Shape Classification Evaluation II: FFT Applied to the Signature of Blobs, pp.607- 612, 2005.

[16] L. Shapiro, G. Stockman.. *Computer Vision*. 2002. Prentice Hall.

[17] G. A. Moore. Automatic Sanning and Computer Processes for the Quantitative Analysis of Micrographs and Equivalent Subjects. Chapter in *Pictorial Pattern Recognition 1* (G.C. Cheng et al.), pp. 275-326, 1968. Thomson, Washington D. C.