Traffic Signs Detection and Tracking using Modified Hough Transform

Pavel Yakimov[1,2] and Vladimir Fursov[1,2]
[1]*Samara State Aerospace University, 34 Moskovskoye Shosse, Samara, Russia*
[2]*Image Processing Systems Institute of Russian Academy of Sciences, 151 Molodogvardeyskaya Street, Samara, Russia*

Keywords: Traffic Signs Recognition, Advanced Driver Assistance Systems, Graphics Processing Units, Image Processing, Pattern Recognition.

Abstract: Traffic Signs Recognition (TSR) systems can not only improve safety, compensating for possible human carelessness, but also reduce tiredness, helping drivers keep an eye on the surrounding traffic conditions. This paper proposes an efficient algorithm for real-time TSR. The article considers the practicability of using HSV color space to extract the red color. An algorithm to remove noise to improve the accuracy and speed of detection was developed. A modified Generalized Hough transform is then used to detect traffic signs. The current velocity of a vehicle is then used to predict the sign's location in the adjacent frames in a video sequence. Finally, the detected objects are being classified. The developed algorithms have been tested using real scene images and the German Traffic Sign Detection Benchmark (GTSDB) dataset and showed efficient results.

1 INTRODUCTION

Traffic Sign Recognition system is designed to provide a driver with relevant information about road conditions. There are several similar systems: 'Opel Eye' of Opel, 'Speed Limit Assist' from the company Mercedes-Benz, 'Traffic Sign Recognition', Ford and others. Most of them are aimed at the detection and recognition of road signs limiting the velocity of movement (Shneier, 2005).

Traffic signs recognition is typically executed in two steps: sign detection and subsequent recognition. There are many different methods of detection: (Nikonorov et al., 2013), (Ruta et al., 2009), (Belaroussi et al., 2010). Most of them use a single frame from a video sequence to detect an object. This means they do not use the additional information about the presence of the sign in the adjacent frames. Such approaches usually have problems of operation in real time and with detection accuracy. On the other hand, several papers describe tracking algorithms, which try to predict the location of signs in a sequence images. In paper (Lafuente-Arroyo et al., 2006), the authors show that the integration of the detection and tracking improves the reliability of the whole system due to the decrease of false detections. In (Lopez and Fuentos, 2007), it is shown that tracking helps to make the detection faster. However, the described algorithms still have significant computational complexity and cannot be used in real time.

This paper describes a detection algorithm that uses the velocity obtained from the vehicle in real time. It allows predicting not only the presence of an object but also the scale and location. Thus, the accuracy will be better, while the computational complexity almost will not change.

In order to give a driver proper information about a traffic sign, the system needs to classify the found object. In fact, the recognition of a small size object does not cause any difficulties in the presence of the samples or patterns of possible traffic signs. In case of a proper detection procedure, the recognition step has accurate sign coordinates and scale. Therefore, this paper only describes a simple template matching algorithm, which shows good results combined with the detection step.

The performance of existing portable computers is not always enough for the real time detection of traffic signs. Many detection algorithms are based on Hough transform that allows you to effectively detect parameterized curves in an image, but this algorithm is very sensitive to the quality of digital images, especially in the presence of noise. The more noise in the image, the longer it will take to detect objects. Thus, the possibility of detecting traffic signs in real

Yakimov P. and Fursov V..
Traffic Signs Detection and Tracking using Modified Hough Transform.
DOI: 10.5220/0005543200220028
In *Proceedings of the 12th International Conference on Signal Processing and Multimedia Applications* (SIGMAP-2015), pages 22-28
ISBN: 978-989-758-118-2

time strongly depends on the quality of the image preparation.

This paper describes the whole technology of traffic sign detection with tracking and recognition. The section with experimental results shows processed real scene images.

2 TRAFFIC SIGNS RECOGNITION SYSTEM

Figure 1 shows a road scene model used to design the algorithms of detection and recognition. Here, α is the camera angle in the horizontal projection; W is the width of a road sign image; AC is the distance from the car to the sign. *Image width* is the width of the input image in pixels. In our case, the width of the frame is equal to the width of FullHD, i.e. 1980 pixels.

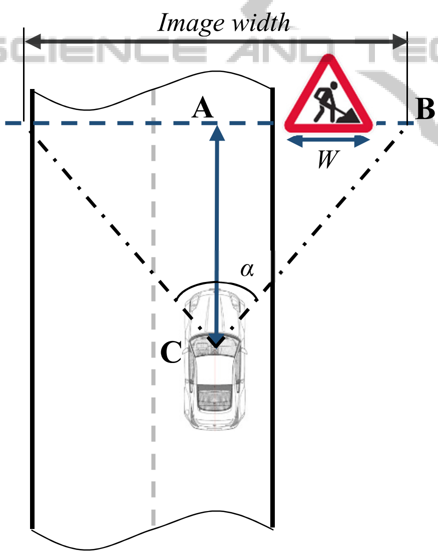


Figure 1: Road scene model.

This paper considers a whole traffic signs recognition technology with three steps: image preparation, detection and recognition. The detection is based on the color information extracted from images. Therefore, the image preparation starts with color thresholding and denoising. Then, a modification of Generalized Hough Transform is used to localize signs in images. A tracking procedure based on the vehicle's velocity is performed to verify a sign presence. Finally, the detected region is being classified.

2.1 Color Analysis and Denoising

Some specific light conditions significantly affect the ability of correct perception of the color in a scene. When taking the actual traffic situation, there are a number of different lighting conditions on the signs.

Figure 2: Example of color extraction in RGB.

The signs detection process becomes much more complicated due to such effects as direct sunlight, reflected light, shadows, the light of car headlights at night. Moreover, the various distorting effects may occur on one road sign at the same time (Figure 2).

Thus, it is not always possible to identify an area of interest in the real images by simply applying a color threshold filter directly in the RGB (Red, Green and Blue) color space. Figure 2 shows an example of applying a threshold filter to the red color channel.

To extract the red color from the input image it is necessary to use the color information of each pixel, regardless of uncontrolled light conditions. For this purpose, the color space HSV (Hue, Saturation and Value) was selected.

Most digital sensors obtain input images in the format of RGB. Conversion to HSV color space is widely described in (Koschan and Abidi, 2008). Between the three components of H, S and V there are certain dependencies. H component will not matter if the S or V components are represented by values that are close to zero. The display color will be black if V is equal to 0. Pure white color is obtained when V = 1 and S = 0.

The ideal red (R = 255, G = 0, B = 0) in the HSV color space is defined by the following values $H = 0.0°, S = 1, V = 1$. The experimental method was used to determine the optimal threshold values to extract the red color of traffic signs in the space of HSV:

$$(0.0° \le H < 23°) \vee (350° < H < 360°) \qquad (1)$$

$$0.85 < S \le 1 \qquad (2)$$

$$0.85 < V \le 1 \qquad (3)$$

Figure 3 shows the result of image processing of the road sign from Figure 2 with threshold values (1)-(3) in HSV.

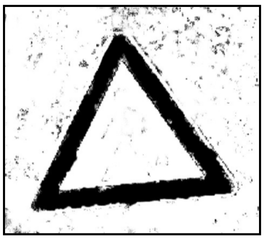


Figure 3: Threshold color filtering in HSV.

The binary image obtained using thresholding satisfies the conditions of many algorithms of traffic signs detection. However, one can easily notice the presence of noise in the image. The picture in Figure 3 is well prepared for further processing, but the situation with the frames captured from a real video sequence is completely different.

The image in Figure 2 was obtained by a camera with high resolution (8.9 megapixels), and shooting conditions were significantly better than when using a built-in car video sensor. Figure 4 shows a fragment of a frame from the video sequence obtained during the experiments containing a road sign.

Noise in the Figure 4b appears after thresholding to extract the red color. It not only reduces the performance of the system, but also affects the quality of detection. This can lead to false detection of road signs.

In order to avoid this point-like noise, a modified algorithm based on the results obtained in paper (Yakimov, 2013) was applied. This article describes the denoising algorithm based on the detection and retouching of point-like glares on the reproductions of works of art. In order to detect these glares the sliding windows algorithm was used. The main advantage of such algorithm is that the parameters can be set in such way that only point-like noise will be removed. At the same time, parts of images of signs stay unfiltered in the processed frames. The result of processing the image from Figure 4b is shown in Figure 4c.

Paper (Fursov et al., 2013) shows the effective implementation of the denoising algorithm in the massively multi-threaded environment CUDA. CUDA is a parallel computing platform and programming model provided by NVidia. It enables dramatic increases in computing performance by harnessing the power of the graphics processing unit (GPU). The resulting acceleration on the GPU relative to the CPU reached 60-80 times. Frame size in the video sequence is 1920x1080 pixels. Image processing execution time on the CPU is 0.7-1 sec. Using CUDA on NVIDIA GeForce 335m has reduced the execution time to 7-10 ms, which satisfies the requirement of processing video in real time.

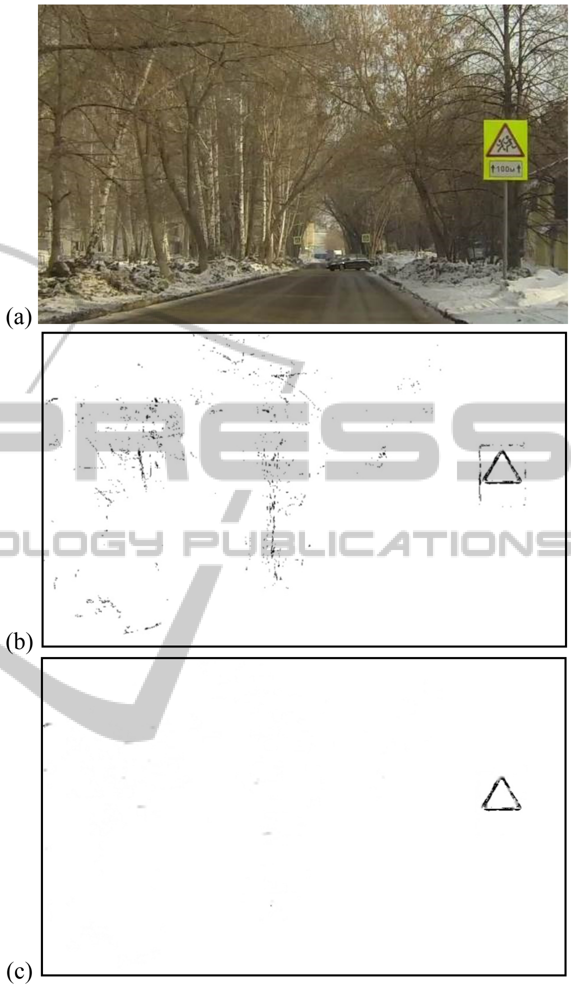


Figure 4: (a) A frame from video sequence; (b) Binary image with extracted red; (c) Result of image denoising.

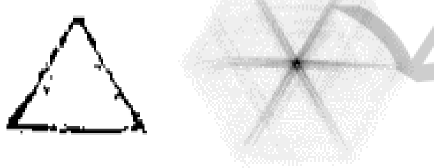
2.2 Detection and Tracking

2.2.1 Modified Hough Transform

Detecting traffic signs is implemented using a modification of Generalized Hough transform (GHT) (Ruta et al., 2008). Implementing classic GHT in Full HD 1080p images leads to enormous execution time. One of the main objectives of the TSR system is to operate in real time. Therefore, there are maximum 50 milliseconds for processing one frame on the detection step.

Many TSR systems are designed to detect only circular signs. There is no difficulty to detect circles using an implementation of Hough transform, and using CUDA makes it possible to implement it in real time. All processing takes no more than 40 ms including steps of color extraction, denoising, detection and recognition. Other systems use various machine-learning techniques such as Viola-Jones (Møgelmose et al., 2012) or Support Vector Machine (Lafuente-Arroyo et al., 2010), which do not always suit the execution time limitation.

In this paper, we consider detection and recognition of triangular signs in real time. The main difference from the original GHT is in using some other accumulator space (Figure 5b) and avoiding the R-table construction. After applying a special triangular template to the binary image in Figure 4c, the point with the maximum value in Figure 5b is the central point of the sought-for object. The case shown in Figure 5 is for equal scales of a template object and object in the real scene. The colors in the pictures are inverted in comparison to the images used in the algorithm.



(a) (b)

Figure 5: (a) An example of a triangular sign after color extraction; (b) Accumulator space after implementing the developed algorithm.

In case of different scales, we receive some more extremum points in the accumulator, three points when implementing the algorithm using a triangular template (Figure 6).

(a) (b)

Figure 6: The accumulator space in case of (a) template is smaller than the object in real scene; (b) template is bigger than the object in real scene.

Assuming that the size of a sign is up to 150 pixels, we have found that the distance between two of these points is up to 20 pixels. A sign with 150 pixels of width means that it is located in 3.5 meters from the camera. There is no opportunity and no need to detect a closer sign, because it moves out from the view of a camera with α equal to 70 ° (Figure 1). It allows computing the difference in scales of the template and an object in real scene. Using this value, we can precisely define the area of sign and then pass it to the recognition step. The middle location of these bright points in the accumulator is the coordinates of an object's center.

This paper shows the detection procedure of triangular traffic signs only. However, the described algorithm can also be used to detect rectangular and circular objects in images, thus covering almost all of the traffic sign classes. The main difference is the template images used for detection.

2.2.2 Tracking with Prediction

The above-described traffic sign detection algorithm is designed to localize objects in each two-dimensional frame of a video sequence. However, some researchers propose to use additional information to increase the reliability of traffic sign recognition. For example, in the article (Timofte et al., 2014), the authors propose to combine data obtained with a usual camera and three-dimensional scene obtained with lidar.

In this paper, we assume that a vehicle is equipped with only one FullHD camera. In this case, we can only use two-dimensional consecutive frames of a video sequence. The use of adjacent frames is described in articles (Guo et al., 2012) and (Mogelmose et al., 2012). Tracking traffic sign on adjacent frames can not only increase the confidence in the correct detection, but also reduce the computational complexity of the algorithm by reducing the search area in the adjacent frames.

In this article, tracking of traffic signs in a video sequence is performed using the information on the current vehicle speed. Most modern cars are equipped with onboard computers or GPS receivers, which can return the current vehicle speed *V*. In addition, we know the number of received frames per second *FPS*. Thus, we can get the exact difference of the distance to the sign in adjacent frames:

$$\Delta AC = \frac{V}{FPS}\,. \tag{4}$$

Consider the case of detecting a triangular traffic sign shown in Figure 5, using the model of the road scene shown in Figure 1. The distance from the vehicle to the sign can be obtained using the following equation:

$$AC = \frac{SignWidth_m \times \dfrac{FrameWidth_p}{2}}{SignWidth_p \times \tan\left(\dfrac{\alpha}{2}\right)} \tag{5}$$

Here, index $_m$ means that a value is in meters, index $_p$ means a value in pixels.

The actual size of the road signs is known and in our case is 0.7 m. We assume that the sign width in pixels is equal to 31 pixels as this size is quite suitable for the subsequent recognition step. α is equal to 70 °. In this case, the distance AC is equal to 30 m.

Using the difference (4) and the distance (5), we can calculate the exact size of the sign on the adjacent frames:

$$SignWidth'_p = \frac{AC \times SignWidth_p}{AC + \Delta AC} \tag{6}$$

Thus, the detection with tracking uses vehicle speed to predict the sizes of a detected traffic sign in the adjacent frames. This significantly increases the reliability of the correct detection and at the same time reduces the required time for detection.

2.3 Recognition

For the recognition step, the algorithm uses the specially prepared binary etalon images, which are actually inner areas of traffic signs. Figure 7 shows such etalons.

Figure 7: Etalon images for template matching.

After detection, the algorithm obtains images of detected objects, which are quite similar to etalons since they are previously resized to the constant size of etalon images. To determine the type of a found object, we can use any recognition method. However, in case of successful object detection, and due to the execution time limitation, it is expedient to use a simple image subtraction and choose the lowest value, which will point to the most similar etalon. In case of large values, the algorithm gives a false detection message, since no similar etalon images were found.

The execution time of such recognition is 1-2 ms in average with 32 types of image etalons. This performance allows using several etalon images of each type to increase the recognition efficiency and reliability.

3 EXPERIMENTAL RESULTS

The developed algorithm was tested on the video frames obtained on the streets of the city of Samara using a camera GoPro Hero 3 Black Edition built in to a car.

Figure 8 shows the fragments of the original images with marked road signs on them.

Figure 9 shows the result image of the detection algorithm without the prior application of the noise reduction algorithm. It shows a case of false detection. In this noisy image, the accumulator space collected more votes for noisy part of the image than for road sign is in the shade.

Figure 8: Frames with detected signs.

Figure 9: False detection.

Note that it took 80 ms to apply the detection algorithm on the noisy image. While it took almost half as much time (41 ms) to process the denoised image.

In order to evaluate the detection and recognition algorithms accuracy, we used the German Traffic Sign Detection Benchmark (GTSDB) (Houben et al., 2013) and the German Traffic Sign Recognition Benchmark (GTSRB) (Stallkamp et al., 2012). They contain more than 50,000 images with traffic signs registered in various conditions. To assess the quality of the sign detection, we counted number of images with correctly recognized traffic signs. When testing the developed algorithms, we used only 9,987 images containing traffic signs of the required shape and with red contours. The experiments showed 97,3% of correctly detected and recognized prohibitory and danger traffic signs.

The remaining 2.7% of traffic signs were rejected on the recognition step with the message “There is no sign in the detected region”. When lowing the thresholds of recognition, we can increase the accuracy of the whole procedure. Still, it is much more important to avoid false detection cases in real situation than to miss some traffic signs.

While several teams-participants of both benchmarks reached 100% of detection and recognition accuracy, they did not provide any information about the algorithms performance. In this article, we describe the implementation that can operate in real time using hardware of limited power consumption and comparatively low performance.

Table 1 shows the execution time of the proposed traffic signs detection and recognition algorithms. CPU results were obtained using Intel Core i5 3210m; GPU results were obtained using CUDA-enabled NVIDIA GeForce GT 750m.

Table 1: Traffic signs detection and recognition algorithms performance.

	Time	FPS
CPU, 1920x1080	283 ms	3.53
CPU, 1280x720	142 ms	7.04
GPU, 1920x1080	23 ms	43.47
GPU, 1280x720	14 ms	71.42

Figure 10 shows some examples of detected traffic signs in the images from the German Traffic Sign Detection Benchmark dataset. The detected objects are marked with green rectangles.

In Figure 11, there are examples of detected and recognized traffic signs from the German Traffic Sign Recognition Benchmark dataset. Despite the bad registration quality of some images, most of the detected objects are recognized correctly, because the coordinates of the inner area of a traffic sign are detected quite precisely.

Figure 10: Traffic signs detection in images from the GTSDB dataset.

Figure 11: Successfully recognized traffic signs from the GTSRB dataset.

In future research, we plan to create a new traffic sign detection and recognition dataset with annotated videos instead of images, like in GTSDB and GTSRB.

4 CONCLUSIONS

This paper proposes a whole technology for traffic signs recognition, including image preprocessing, detecting with tracking, and recognition of traffic signs. The HSV color model was approved as the most suitable one for the extraction of red color in the images. The modified algorithm for removing noise helped not only to avoid false detection of signs, but also accelerated the processing of images. The developed algorithm can improve the quality and increase the reliability of automotive traffic sign recognition systems, and reduce the time required to process one frame, which brings the possibility to carry out the detection and recognition of signs in Full HD 1920x1080 images from the video sequence in real time.

An algorithm for detection of triangular signs is considered in the paper. It is based on the Generalized Hough Transform and is optimized to suit the time limitation. The developed algorithm shows efficient results and works well with the preprocessed images. Tracking using a vehicle's current velocity helped to improve the performance. In addition, the presence of a sign in a sequence of adjacent frames in predicted areas dramatically improves the confidence in correct detection. Recognition of detected signs makes sure that the whole procedure of TSR is successful.

In this paper, we consider triangular traffic signs. The developed detection algorithm makes it possible to detect signs of any shape. It is only needed to replace the template image with a sought-for shape.

The use of our TSR algorithms allows processing of video streams in real-time with high resolution, and therefore at greater distances and with better quality than similar TSR systems have.

CUDA was used to accelerate the performance of the described methods. In future research, we plan to move all the designed algorithms to the mobile processor Nvidia Tegra X1.

ACKNOWLEDGEMENTS

This work was partially supported by Project #RFMEFI57514X0083 by the Ministry of Education and Science of the Russian Federation.

REFERENCES

Shneier, M., 2005. Road sign detection and recognition. *Proc. IEEE Computer Society Int. Conf. on Computer Vision and Pattern Recognition*, pp. 215–222.

Nikonorov, A., Yakimov, P., Petrov, M., 2013. Traffic sign detection on GPU using color shape regular expressions. *VISIGRAPP IMTA-4*, Paper Nr 8.

Ruta, A., Porikli, F., Li, Y., Watanabe, S., Kage, H., Sumi, K., 2009. A New Approach for In-Vehicle Camea Traffic Sign Detection and Recognition. *IAPR Conference on Machine vision Applications (MVA)*, Session 15: Machine Vision for Transportation.

Belaroussi, R., Foucher, P., Tarel, J. P., Soheilian, B., Charbonnier, P., Paparoditis, N., 2010. Road Sign Detection in Images. *A Case Study*, 20th International Conference on Pattern Recognition (ICPR), pp. 484-488.

Lafuente-Arroyo, S., Maldonado-Bascon, S., Gil-Jimenez, P., Gomez-Moreno, H., Lopez-Ferreras, F., 2006. Road sign tracking with a predictive filter solution. *IEEE Industrial Electronics,* IECON 2006 - 32nd Annual Conference on, vol., no., pp.3314-3319.

Lopez, L.D. and Fuentes, O., 2007. Color-based road sign detection and tracking. *Image Analysis and Recognition, Lecture Notes in Computer Science*. Springer.

Koschan, A., Abidi, M. A., 2008. Digital Color Image Processing. *ISBN 978-0-470-14708-5*, p. 376.

Yakimov, P., 2013. Preprocessing of digital images in systems of location and recognition of road signs. *Computer optics,* vol. 37 (3), pp. 401-405.

Fursov, V.A., Bibikov, S.A., Yakimov, P.Y., 2013. Localization of objects contours with different scales in images using Hough transform. *Computer optics*. Vol. 37(4), pp. 496-502.

Ruta, A., Li, Y., Liu, X., 2008. Detection, Tracking and Recognition of Traffic Signs from Video Input. *Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems.* Beijing, China.

Møgelmose, A., Trivedi, M., Moeslund, M., 2012. Learning to Detect Traffic Signs: ComparativeEvaluation of Synthetic and Real-World Datasets. *21st International Conference on Pattern Recognition*, pp. 3452-3455, IEEE.

Lafuente-Arroyo, S., Salcedo-Sanz, S., Maldonado-Basc´on, S., Portilla-Figueras, J. A., Lopez-Sastre, R. J. 2010. A decision support system for the automatic management of keep-clear signs based on support vector machines and geographic information systems. *Expert Syst. Appl.*, vol. 37, pp. 767–773.

Timofte, R., Zimmermann, K.,Van Gool, L., 2014. Multi-view traffic sign detection, recognition, and 3D localisation. *Machine Vision and Applications,* vol. 25, pp. 633-647, Springer Berlin Heidelberg.

Guo, C., Mita, S., McAllester, D., 2012. Robust Road Detection and Tracking in Challenging Scenarios Based on Markov Random Fields With Unsupervised Learning. *Intelligent Transportation Systems, IEEE Transactions on*, vol.13, no.3, pp.1338-1354.

Mogelmose, A., Trivedi, M.M., Moeslund, T.B., 2012. Vision-Based Traffic Sign Detection and Analysis for Intelligent Driver Assistance Systems: Perspectives and Survey. *Intelligent Transportation Systems, IEEE Transactions on*, vol.13, no.4, pp.1484-1497.

Houben, S., Stallkamp, J., Salmen, J., Schlipsing, M., Igel, C., 2013. Detection of Traffic Signs in Real-World Images: The {G}erman {T}raffic {S}ign {D}etection {B}enchmark. *International Joint Conference on Neural Networks*.

Stallkamp J., Schlipsing M., Salmen J., Igel C., 2012. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural networks*, vol. 32, pp. 323-332.