Experiments with a Transform based Traffic Sign Recognition System

M. Fifik, J. Turán, Ľ. Ovseník.
Department of Electronics and Multimedia Communications
Technical University of Košice
Park Komenského 13, 041 20 Košice, Slovakia
E-mail: martin.fifik@tuke.sk

K. Fazekas.
Department of Broadband Infoelectronics
Budapest University of Technology and Economics
Goldmann Gy. ut.2, 1111,Budapest, Hungary
E-mail: kfazekas@cyberspace.mht.bme.hu

Abstract— Article presents experiments with a transform based traffic sing recognition system. The traffic scene is processed with color camera and key frame is taken as input for the system "core block". The core is based on projection transforms (Trace transform etc.). The output from core block is used as input in classification block. In classification block neural network, or simple classification method is used to select proper traffic sign. Several experiments have been done with this system. Results and conclusion are discussed.

Keywords-traffic sing, projection transform, classification, color segmentation.

I. Introduction

With traffic expansion, we are dealing with situations, where human life and health is in danger. The amount of cars grows and with this also grows numbers of crashes and dangerous situations, where can be in danger driver or some other participant of traffic flow.

One choice how we can change this situation, is design of video assistance system for drivers, which can pre prevent these car accidents. Video assistance systems can just inform driver or they can make direct intervention in car driving process.

Video assistance system covers large range with different approaches. Video assistance system are used for tasks like detecting cars, traffic signs, pedestrians, obstacles, traffic lines or car tables[1-4].

Goal of this paper is present system with traffic sing detection. Projecting transforms are used for invariant feature extraction. The rules for safety traffic are displayed on traffic signs. Traffic signs are designed to show us some rule or warn us before something [1-4].

System is based on two step detection scheme (Fig. 1). Input is image, which basically contains information that we want extract. Block called Hypothesis Generation (HG) generates the promising hypotheses in form of **R**egion **O**f **I**nterest (**ROI**). After this, block called **H**ypothesis **V**erification (**HV**) verifies or rejects that previously generated hypothesis. On the output we have image with **ROI**s that are recognized or they are determine to next processing [2].

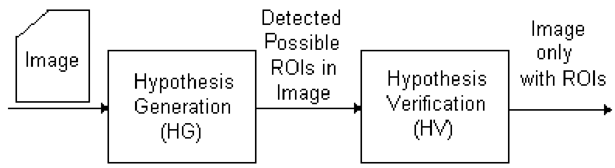

Figure 6. Two step detection scheme

II. System for Traffic Sign Detection

Based on two step detection scheme, system for traffic sign detection was designed (Fig. 2).

This system can be divided in two subsystems. One, which is called fast, composed from CSR block and Sign Class Classification block, and second slower, composed from Image Segmentation, Invariant Feature Extraction System, Feature Memory, Feature Modification and Sign Type Classification blocks.

Fast subsystem is used to promptly detection of class type and position of traffic sign. Slower subsystem is used to completely recognition of traffic sign type.

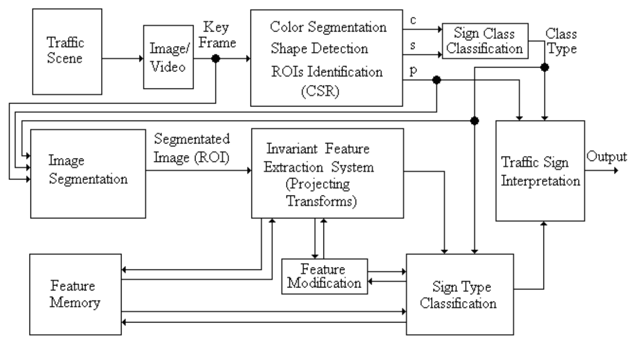

Figure 7. System for traffic sign detection

A. System description

Traffic scene is recorded with color camera. From camera created data stream, key frame is extracted. Key frame is used in next processing.

This key frame is processed in CSR block, where the base features from image are extracted. Features extracted from key frame are color, shape and ROIs position. This preprocessing is very fast, because no special transformation is there needed. From many color spaces we choose RGB and HSV color space. By extracting color, CSR block generates 3 binary maps (red, blue and yellow). In this bitmaps we are looking for shapes of traffic signs. If traffic sign was detected, then from color and shape features traffic sign class type is identified. Traffic class type goes together with position of traffic sign to block for Image Segmentation. From key frame the **ROI** is segmented, which is input to block called Invariant Feature Extraction System.

In Invariant Feature Extraction System projecting transform are used (Trace transform, Hough transform) [6,7,9]. Invariant features are features that have equal values in cases when the image is translated, rotated, resized or have some other modification. The block of Invariant Feature Extraction System with cooperation with blocks Feature Memory and Feature Modification block, gives information for Sign Type Classification block. In this block are brought information about class type of detected traffic sign. An output of this block gives us completely recognized traffic sign for block called Traffic Sign Interpretation.

B. CSR Block (Preprocessing)

This block is preprocessing block. In experiments two color spaces are used: HSV and RGB.

1) HSV color space

Input is converted in to the HSV color space. Every traffic sign has his dominant color. On Slovak roads most often yellow, red and blue color are used. This means we need to create three binary maps, one for each of these colors.

By analyzing hue component (H), we can identify blue, yellow and red regions in our detected image. For each image pixel, hue-based detection of blue, red and yellow colors is done. For each color one passes one of following equation[1]

$$Y = e^{\frac{-(x-42)^2}{30^2}} \quad (1)$$

$$R = e^{\frac{-x^2}{20^2}} + e^{\frac{-(x-255)^2}{20^2}} \quad (2)$$

$$B = e^{\frac{-(x-170)^2}{30^2}} \quad (3)$$

Equations Y gives values close to 1 for yellow regions, R gives values close to 1 for red regions and B gives values close to 1 for blue regions. In this equations we can see, that H can be from range 0-255. Yellow can be detected near value 42, red near values 0 and 255 and blue value is 170. These equations can be tuned for every color.

Now we need saturation detection value, by exploring the S component. This is described by following equation

$$S = e^{\frac{-(x-255)^2}{115^2}} \quad (4)$$

From equations (1), (2) and (3) we got 3 values. Every value is multiplied with S value. This value will be called D. From D we create D_n, which means D normalized. Values close to zero will be discarded. Other will follow next equation (5).

$$D_n = \begin{cases} 0, if \quad D < 0{,}3 \\ \left(\frac{D-0{,}3}{0{,}7-0{,}3}\right)^2 \\ 1, if \quad D \geq 0{,}7 \end{cases} \quad (5)$$

Now the threshold can be created. For this we use Otsu's algorithm on D_n [8]. Then we can create three binary maps[1,8].

Now every binary map must be cleared. Too small regions and too big regions are discarded. In first step we must find first white point in image. Searching is done by rows. After finding first white point, method called seed-fill is used. With this method we are finding regions, and these regions are valued. If valued region has value lower than 400, then is discarded. If region has value more than 16000 then is also discarded. The regions that are left are potential candidates for next processing.

Method seed-fill we can apply only, if all white areas in image have their boundaries. Next condition is that there must be at least one white point. After successful finding white point A (seed), its neighbors are tested (Fig. 3).

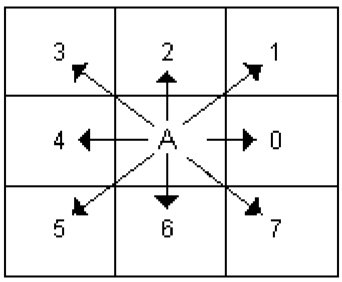

Figure 8. Values 0-7 for neighbors of seed A

This white point is filled and its neighbors' relations are checked. These points are now stored in seed vector. In next step we continue with last filled point. If this point has no unfilled neighbor, then point from seed vector is choosing for next checking. After using all points from seed vector, we have our region that has value. On the end we have selected **ROI**(s).

2) RGB color space

Input is in RGB format. First step is brightness correction of input image. Brightness correction was done manually.

Next step is color segmentation. This has been done by using following equations (6):

$$\begin{aligned} \alpha_{\min} * G < R < \alpha_{\max} * G \\ \beta_{\min} * B < R < \beta_{\max} * B \\ \gamma_{\min} * B < G < \gamma_{\max} * B \end{aligned} \qquad (6)$$

Three binary maps have been generated; bitmaps are cleared with the same way as it was at HSV. Those bitmaps are used for ROI detection. ROI detection is done by same method at HSV detection [1].

3) Shape detection

Figure 9. Four perfect shapes

Now with pattern matching method, on that size of ROI are created perfect shapes, like it is show on Fig.10. Here every **ROI** is tested pixel by pixel with every perfect shape, with circle, rhomb, triangle, reverse triangle, filled circle-STOP sign, square /rectangle [5].

C. Invariant Feature Extraction System Block (Projecting Transforms)

This block is used for image detection inside traffic signs. The major problem of this image is that it is not perfect. For example: it is resized, rotated or moved.

For invariant feature extraction Trace transforms is used here. Trace transform can be understood as generalization of the well-know Radon transformation. Trace Transform is detail described in [6,7, 9].

D. Sign Type Classification Block

TABLE I. EXAMPLE OF TRAFFIC SIGN CLASSIFICATION

Color / Shape	Red	Blue	Yellow
Square/Rectangle	-	Information	-
Circle	Obligation	Prohibition	-
Rhomb	-	-	Highway
Triangle	Danger	-	-
Reverse triangle	Yield sign	-	-
Cut square	Stop sign	-	-

Now when we got shape and color we can classify traffic signs in classes. This is shown in table I.

All possibilities are not obtained in table. When we know type and image as output from feature extraction system, then we can classify sign in to the type. For example, it's red triangle with image symbolizing kids, so this sign is classified and recognized.

III. EXPERIMENTS

Experiments were done for HSV and RGB color space. Input images were in resolution 640x480 pixels. Tests were done on Pentium PC with dual core processor 2 x 2 GHz.

In HSV color space, the number of tested images was 314; number of false positive tested was 43, which means 86,31 % successful rate. Details are shown in table II.

In RGB color space, the number of tested images was 314; number of false positive tested was 35, which means 88,85 successful rate. Details are shown in table III.

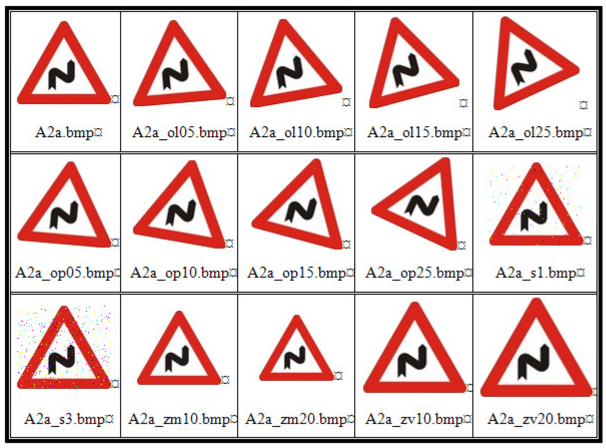

Figure 10. Modified traffic sign for Trace transform experiment

TABLE II. RESULTS FOR HSV COLOR SPACE

Brightness condition	Number of sings	Detected signs	Rate[%]	Average rate[%]
low	93	77	82,796	
normal	196	191	97,449	
extreme	25	3	12	86,31

TABLE III. RESULTS FOR RGB COLOR SPACE

Brightness condition	Number of sings	Detected signs	Rate[%]	Average rate[%]
low	93	81	87,0968	
normal	196	192	97,959	
extreme	25	5	20	88,54

Preprocessing time in both color spaces was average from 1~2 seconds. Average processing time in invariant feature extraction system was 2,2 seconds. Table classification average time was 0,2 seconds.

In table IV., K1, K2, and K3 are combination of funcionals, which gives us significant values. Rate is compared to A2a.bmp.

TABLE IV. VALLUES FOR MODIFIED SIGN A2A

Image	K1	K2	K3	Time [s]	Rate [%]
A2a.bmp	33120.000	64800.000	1450554.923	2.250	-
A2a_ol05.bmp	32752.000	64080.000	1413185.662	2.391	**98.40**
A2a_ol10.bmp	34408.000	67320.000	1524072.932	2.359	**95.72**
A2a_ol15.bmp	35052.000	68580.000	1569528.796	2.390	**93.38**
A2a_ol25.bmp	32844.000	64260.000	1423137.122	2.421	**98.81**
A2a_op05.bmp	33488.000	65520.000	1459762.642	2.422	**99.05**
A2a_op10.bmp	34454.000	67410.000	1528531.930	2.437	**95.52**
A2a_op15.bmp	35558.000	69570.000	1603138.119	2.516	**91.59**
A2a_op25.bmp	33810.000	66150.000	1485780.200	2.516	**97.80**
A2a_s1.bmp	32982.000	64530.000	1437400.945	2.562	**99.42**
A2a_s3.bmp	32936.000	64440.000	1434211.937	2.375	**99.25**
A2a_zm10.bmp	34132.000	66780.000	1516985.554	2.407	**96.44**
A2a_zm20.bmp	32982.000	64530.000	1427542.574	2.312	**99.19**
A2a_zv10.bmp	33166.000	64890.000	1445871.846	2.453	**99.98**
A2a_zv20.bmp	32522.000	63630.000	1403781.816	2.453	**97.72**

IV. CONCLUSION

Experiments show us average system recognition rate 86% at HSV color space and 89% at RGB color space. Factors which affect images in preprocessing stage was darkness and sun lightning.

Trace transform based invariant image recognition system depends on close functional combination. For best result were used three combinations of these functionals. With less number of these combinations, it's harder to determine the completed image.

All reached results was in 3-4 seconds per image, which means that this system can be used in real time.

ACKNOWLEDGMENT

This work was partially supported from the grants VEGA, project COST IC0802 and Centre of excellence FEI TUKE.

REFERENCES

[1] F. P. Paulo, L. P. Correia, "Automatic detection and classification of traffic signs," Image Analysis for Multimedia Interactive Services, 2007, WIAMIS, 6-8 June 2007.

[2] A. Laika, W. Stechele, "A review of different object recognition methods for the application in driver assistance systems, " Image Analysis for Multimedia Interactive Services, 2007, WIAMIS, 6-8 June 2007.

[3] J. Turán, M. Fifik, Ľ. Ovseník and J. Turán Jr., "Transform based system for traffic sign recognition" 15th International Conference on Systems, Signals & Image Processing IWSSIP'08, Bratislava, Slovakia, 2008

[4] A. Broggi, P. Cerri, P. Medici, P. P. Porta, "Real Time Road Signs Recognition", Intelligent Vehicles Symposium, 2007 IEEE Volume , Issue , 13-15, 981 – 986, June 2007

[5] M. Nixon, A. Aguado "Feature extraction and image processing", Academic Press, 2008

[6] J. Turán, "Fast translation Invariant Transforms and their Application," Elfa, Košice, 1999.

[7] J. Turán, Ľ. Ovseník, J. Turán, Jr., "Transform Based Invariant Feature Extraction", 13th International Conference on Systems, Signals & Image Processing IWSSIP'06, Budapest, Hungary, September 21-23, 79-82, 2006

[8] N. Otsu. "A threshold selection method from gray-level histogram" IEEE Transactions on Pattern Analysis and machine Intelligence, vol. 25, n°8, August 2003, pp 959- 973

[9] J. Turán, D. Šiškovičová, Ľ. Ovseník and J. Turán Jr., P.Filo, "Trace transform and KLT Based Invariant Feature and Image Recognition System", Acta Electrotechnika et Informatica, no.3, vol. 6, 2006.

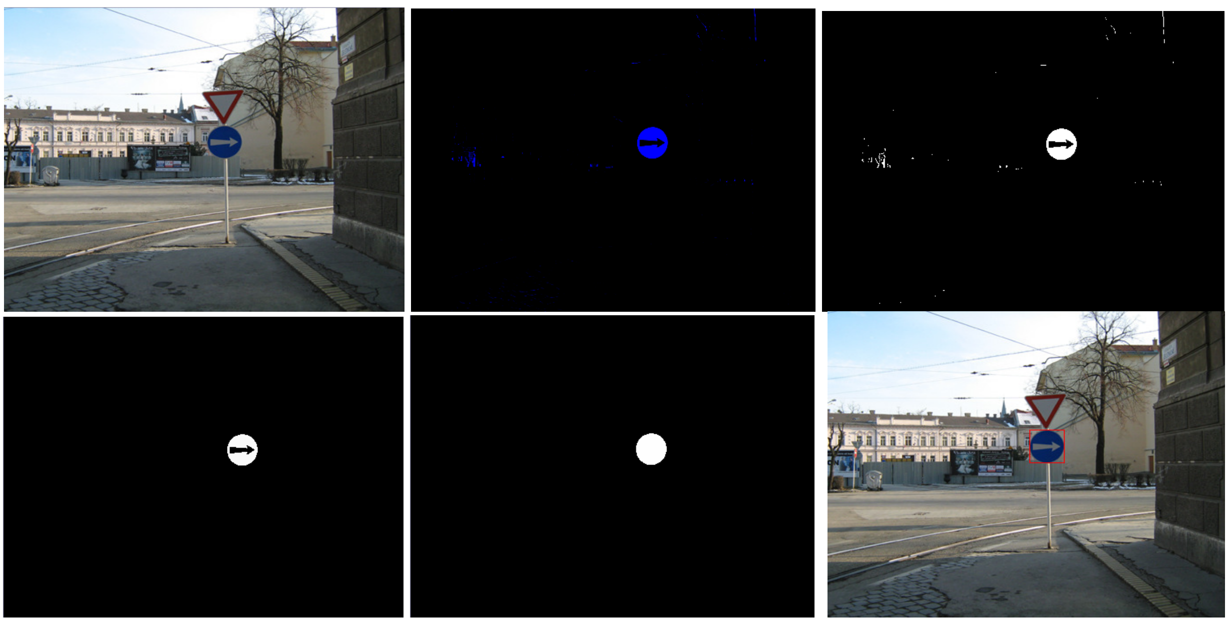

Figure 6. Samples of images for preprocessing and ROI extraction from blue color in HSV color space