

TRAFFIC LIGHTS RECOGNITION FOR ROBOTIC COMPETITION

J. Zavadil¹, V. Santos², J. Tuma¹

¹VSB – Technical University of Ostrava, Czech Republic

²University of Aveiro, Portugal

Abstract

Recognition of traffic lights and signs is an important task for advanced driver assistance systems (ADAS), as well as for autonomous driving systems of driverless cars. This paper deals with the challenge of perceiving traffic signs for autonomous driving. It is concerned with the competition of fully autonomous robots that takes place in a track with the shape of a traffic road. A specific set of signs has been addressed and solved with very good results. A combination of two techniques based on blob analysis and pattern recognition has been used and selected results of the experiments will be presented along with the description of the main algorithms.

1 Competition

The problem to be solved concerns recognition of the light signaling for a robot during Autonomous Driving Competition ROBOTICA 2011. ROBOTICA is the Portuguese Robotics Open that takes place since 2001 and includes a competition for fully autonomous robots running in a track with the shape of a traffic road. Full text of the rules and technical specifications can be found on the website of the competition (<http://robotica2011.ist.utl.pt/>).

To fulfill all the challenges, the task covers the need to recognize five different symbols, which are presented on the signaling panel (17" TFT panel with resolution 1024x768 pixels). The colors used for symbols on images are red, green and yellow on a black background. Dimensions of symbol image area are 600x600 pixels. A scheme of signaling panels as well as images of the symbols presented in these panels can be seen below (Figure 4.1 and Figure 4.2).

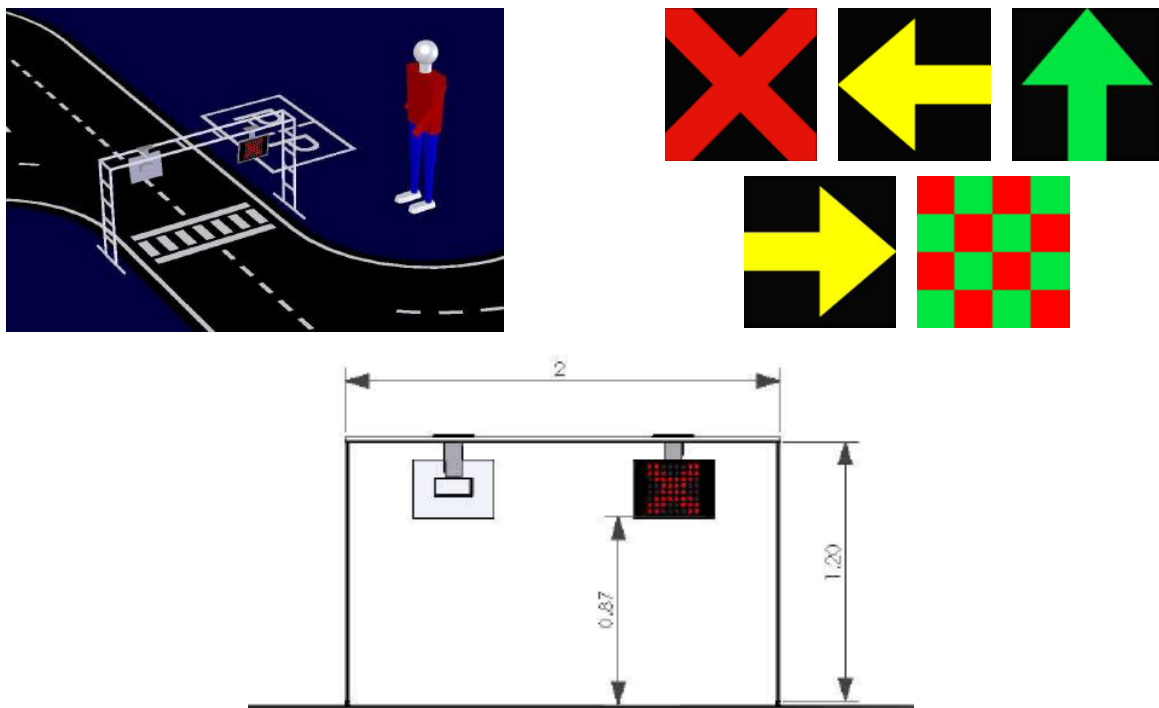


Figure 1 – Overview of the track and symbols presented on the signaling panels [3]

The aim of the signaling panels is to give robots instructions which must be obeyed and affect the final robot performance and assessment. The description of these commands can be seen in Table 1.

Table 1 – Information presented in the signaling panels

Action	Signal
Stop	A red colored cross.
Follow straight ahead	A green colored vertical arrow.
Follow to the left	A left pointing yellow horizontal arrow.
End of trial	Red and green checkers flag.
Follow to parking area	A right pointing yellow horizontal arrow.

Robots used for competition were created by the Group of Automation and Robotics at the Department of Mechanical Engineering of the University of Aveiro. The project is called ATLAS (see atlas.web.ua.pt).

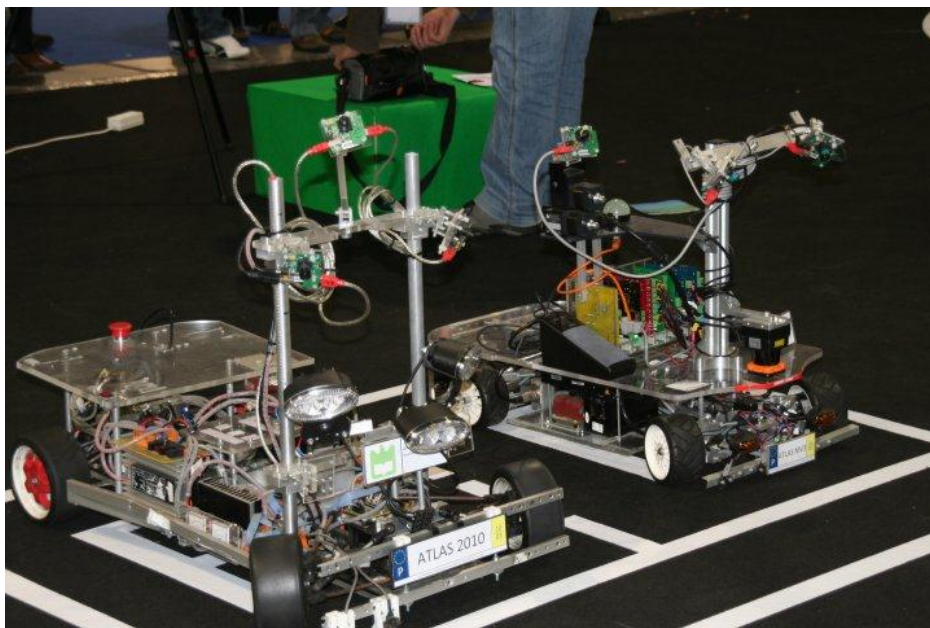


Figure 2 – Robots used for competition [4]

2 Solution

There are a lot of techniques to detect and recognize traffic lights. Proposed solution combines two techniques, namely blob analysis and pattern recognition. First a color pre-segmentation in HSV color space is done. After color segmentation the blob analysis is used to find the symbol candidates. Finally, the minimum distance classifier is used for classification of these candidates. This chapter describes the used methods.



Figure 3 – Examples of images taken by robot during the competition [ATLAS]

As mentioned, the first step after acquisition of an image is the conversion to HSV color space and color segmentation. Used ranges for each color component are listed in Table 2. The values were reached by testing of several sets of images (2000 images approximately). HSV parameters are in the range [0, 1]. The dependence to illumination conditions impose that large ranges in the parameters have to be used. This, of course, creates many candidates that have to be further filtered and processed.

Table 2 – Used color ranges values (obtained by testing approximately 2000 images)

	H	S	V
red	0 - 0.05 & 0.96 - 1	> 0.5	> 0.5
green	0.2 - 0.54	> 0.4	> 0.4
yellow	0.11 - 0.2	> 0.5	> 0.4

After color segmentation follows the blob analysis or, more precisely, analysis of region descriptors. This analysis has been done in MATLAB using the `regionprops()` function to calculate properties of the pre-segmented image regions. The `regionprops()` functions yields a large set of properties. From these, a reduced set of properties was chosen to describe the symbol candidates. The used properties can be seen in Table 3.

Table 3 – Used properties of regionprops()

Area	Eccentricity	FilledArea
BoundingBox	EulerNumber	Orientation
Centroid	Extent	Solidity

The main concern to select these properties was to have an algorithm robust to scale and rotation variance. Not all descriptors turned out as useful as expected. Also, some descriptors not available directly from the Matlab `regionprops` function were calculated manually; one of these cases the Form Factor.

To perform an adequate detection of signs, a comparison with some reference symbols was used. Therefore, a “pattern” of properties for the symbols (regions) has been defined and the Mahalanobis distance is computed for each symbol candidate. Contrary to the Euclidean distance the Mahalanobis distance takes into account the correlations of the data sets and is therefore independent to dynamic range variations of the descriptors. The Mahalanobis distance is defined as:

$$d_M(Y, X) = \sqrt{(Y - X)^T \Sigma^{-1} (Y - X)} \quad \Sigma = \begin{bmatrix} \sigma_{11}^2 & \sigma_{12}^2 & \cdots & \sigma_{1n}^2 \\ \sigma_{21}^2 & \sigma_{22}^2 & \cdots & \sigma_{2n}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1}^2 & \sigma_{n2}^2 & \cdots & \sigma_{nn}^2 \end{bmatrix} \quad (1)$$

where X represents the reference set of samples, Y is an object to be classified and Σ is the covariance matrix among the several parts of the pattern (descriptors).. Along the main diagonal of this matrix are the variances of the data set.

In MATLAB there is the `mahal()` function for computing the Mahalanobis distance. The syntax of this function is following

```
>> d = mahal(X, Y);
```

X and Y must have the same number of columns, but can differ in numbers of rows. X must have more rows than columns. For more information see MATLAB Help.

To create the reference set, 12 images of each symbol were chosen. Following descriptors were used to define pattern:

- Solidity (S)
- Eccentricity (Ecc)
- Extent (Ex)
- Form Factor (F)
- Axis Proportion (A)

$$X = \begin{bmatrix} S_1 & Ecc_1 & Ex_1 & F_1 & A_1 \\ S_2 & Ecc_2 & Ex_2 & F_2 & A_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ S_{12} & Ecc_{12} & Ex_{12} & F_{12} & A_{12} \end{bmatrix} \quad (2)$$

The aim of the created program is to capture and process images from a connected camera. A standard FireWire color digital camera was used for experiments (video mode 320x240 YUV). The program was running on a standard 2.26GHz dual-core laptop with 4GB RAM. Windows 7 64bit operating system and MATLAB R2010b were used.

3 Results

The final test using images from the competition in 2010 was done. These images were taken in different lights conditions then images used as a reference set. Despite of these difficult conditions, the success of recognition was very good. But still the results depend on surrounding light conditions. Majority of images in this test did not contain any symbol. However, there were no false positives. Examples of used images can be seen on Figure 4.



Figure 4 – Example of well recognized images

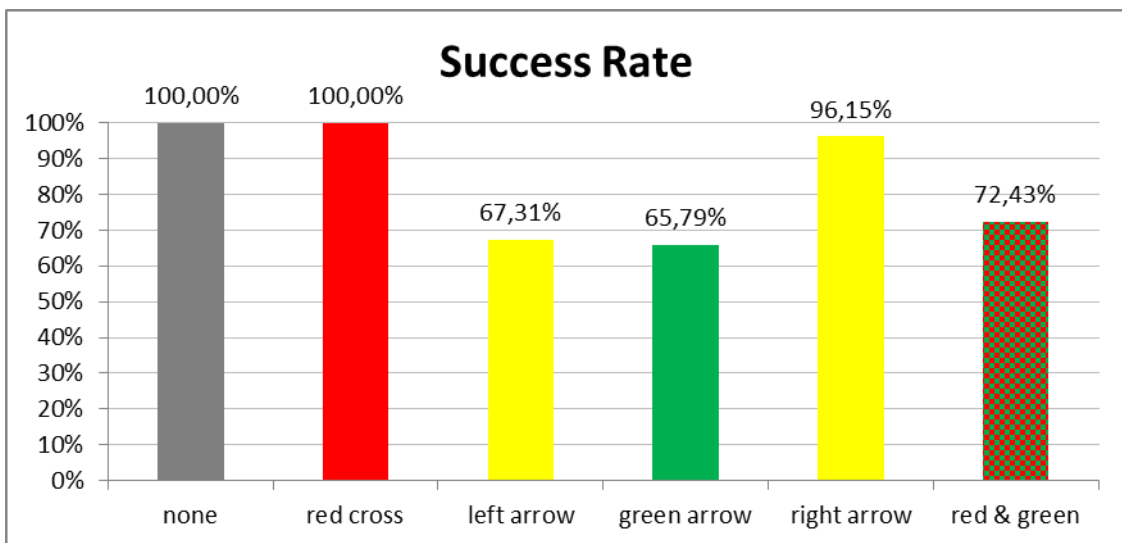


Figure 5 – Results of the test

The poorer results of some symbols recognition were caused by large distance of the camera from the Signaling Panel. Maximum distance for good recognition of symbols is approximately 2,5m. Time taken by processing one image was approximately 0.7 s.

Table 4 – Results of the test

Tested Images							Results		
red cross	left arrow	green arrow	right arrow	red & green	without light	total	missed	wrong	
18	104	38	26	243	2885	3314	115	0	

To compare the results of symbols recognition better, the last test with images only taken approximately up to distance of 2 meters from the signaling panel was done. When only images taken from shorter distance are included, the results are significantly better. Only one yellow arrow and 14 Red & Green symbols were missed. It was caused by bad taken images.

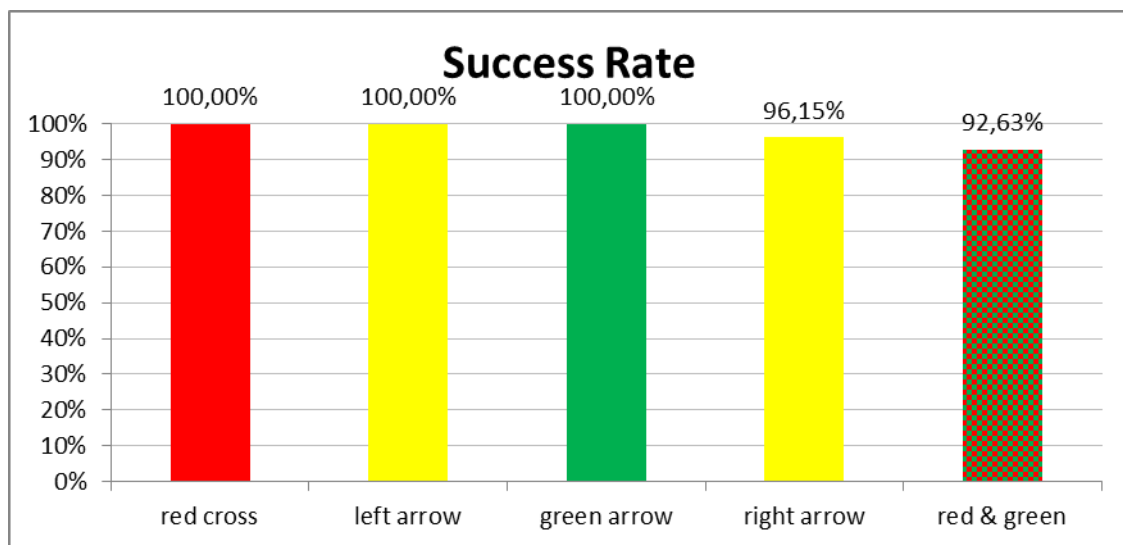


Figure 6 – Results of the test with images taken till 2m distance

Results of the tests show that even using relatively small reference set is possible to reach very good success rate.

Table 5 – Results of the test with images taken till 2m distance

Tested Images						Results	
red cross	left arrow	green arrow	right arrow	red & green	total	missed	wrong
18	70	17	26	190	321	15	0

4 Conclusion

The developed program solves recognition of symbols of traffic lights which are used for a robotic competition. It was created especially for the competition ROBOTICA 2011 Festival National de Robótica.

This work shows a possibility of the MATLAB environment in the field of digital image processing. The accuracy of used approach depends on surrounding light conditions. However, even using relatively small reference set the algorithm reached very good success rate. Recognition of symbols is very good for distances up to 2m reaching almost 100%. Time taken by processing one image was approximately 0.7 s.

References

- [1] Gonzalez, Woods, Eddins, *Digital Image Processing Using MATLAB*, Prentice Hall, 2003, 728 p., ISBN: 978-0130085191
- [2] C. Solomon, T. Breckon, *Fundamentals of Digital Image Processing, A Practical Approach with Examples in MATLAB*, Willey-Blackwell, 2011, 344 p., ISBN: 978 0 470 84472 4
- [3] Robotica 2011, Festival Nacional de Robotica, *Competition, Rules and Technical Specifications*, 2011, 27 p., Accesible on www: <http://robotica2011.ist.utl.pt/docs/2011_Conducao_Autonomia-regras-en.pdf>
- [4] ATLAS project website, Accesible on www: <atlas.web.ua.pt>
- [5] S. Theodoridis, K. Koutroumbas, *An Introduction to Pattern Recognition: A MATLAB Approach*, Elsevier Inc., 2010, 219 p.
- [6] S. Theodoridis, K. Koutroumbas, *Pattern Recognition, Second Edition*, Elsevier, 2003, 689 p., ISBN: 0-12-685875-6
- [7] M. N. Murty, V. S. Devi, *Pattern Recognition, An Algorithmic Approach*, Springer, 263p., ISBN: 978-0-85729-494-4
- [8] Tůma, J., *Signal Processing*, VSB – Technical University of Ostrava, 17. listopadu 15/2172, Ostrava – Poruba, 2009, ISBN 978-80-248-2114-6

Jaromir Zavadil – jaromir.zavadil@vsb.cz
VSB – Technical University of Ostrava,
Faculty of Mechanical Engineering, Department of Control Systems and Instrumentation,
17. listopadu 15, 708 33 Ostrava – Poruba, Czech Republic

Vitor Manuel Ferreira dos Santos – vitor@ua.pt
University of Aveiro,
Department of Mechanical Engineering,
Campus de Santiago, 3810-193 Aveiro, Portugal

Jiri Tuma – jiri.tuma@vsb.cz
VSB – Technical University of Ostrava,
Faculty of Mechanical Engineering, Department of Control Systems and Instrumentation,
17. listopadu 15, 708 33 Ostrava – Poruba, Czech Republic