Traffic sign shape classification evaluation II: FFT applied to the signature of Blobs

P. Gil-Jiménez, S. Lafuente-Arroyo, H. Gómez-Moreno, F. López-Ferreras and S. Maldonado-Bascón

Dpto. de Teoría de la Señal y Comunicaciones
Universidad de Alcalá
Alcalá de Henares, Madrid
e-mail: {pedro.gil, sergio.lafuente, hilario.gomez, francisco.lopez, saturnino.maldonado}@uah.es

***Abstract*— In this paper we have developed a new algorithm of artificial vision oriented to traffic sign shape classification. The classification method basically consists of a series of comparison between the FFT of the signature of a blob and the FFT of the signatures of the reference shapes used in traffic signs. The two major steps of the process are: the segmentation according to the color and the identification of the geometry of the candidate blob using its signature. The most important advances are its robustness against rotation and deformation due to camera projections.**

I. Introduction

Traffic sign detection and recognition have been an important issue for research recently: [1], [2], [3], [4], [5], [6] are some of these works. In this introduction a general overview of the state of the art of shape classification for traffic signs is given and a general description of the state of the art of segmentation and the recognition of traffic sign is given in [7].

In [8], triangular signs are localized by seeking in the image the three kinds of corner that form the triangle and by proving that they are, in fact forming an equilateral triangle. A sign will be present when three corner are found forming an equilateral triangle. For rectangular signs, the algorithm seeks the four kinds of 90 degrees corners that form the sign and that are located defining a rectangle. However, deformed signs due to projective distortions are not detected because only regular signs are considered. A similar method is used for the circular shapes.

In [9] an edge image is obtained from the segmented one, a voting for every candidate center of a circle is considered for circular signs. For rectangular shapes, four boundary line segements are detected and triangular shapes are not considered.

In [4] a configurable adaptive resoance theory neural network is used to determine the category of the input stimuli, and categories are related with the shape and color of the signs.

In [2] color and shape are the key features for the detection stage and they are used to split the problem into different subproblems. For every color the possible shapes are different: circular and triangular for red and circular and rectangular for blue. Genetic algorithm and simulated annealing are applied for traffic signs detection. Achromatic traffic signs as "end of limit" have not been analyzed in this paper.

Color	Shape	Meaning
Red Rim	Circle	Prohibition
Red Rim (Up)	Triangle	Danger
Red Rim (Down)	Triangle	Yield
Red	Octagonal	Stop
Blue	Square	Recommendation
Blue	Circle	Obligation
White	Circle	End of prohibition
Yellow	Circle	End of prohibition (construction)

TABLE I
MEANING OF TRAFFIC SIGNS ACCORDING TO ITS COLOR AND SHAPE

Many of these works show partial solutions to the general problem of traffic sign detection and recognition and none of them shows comparative results with other methods.

Two different methods for detection and classification of traffic sign according to its shape have been developed. The first method is based in Distance to Borders measurement and linear SVM and it is presented in another paper of this conference [7]. The other is based on FFT applied to the signature of the blob obtained from the segmentation and it is presented on this paper.

II. Test set categories

We have created a traffic sign image database, test set, that can be used to evaluated traffic sign detection and recognition algorithms. This test set covers the most important common problems in traffic sign detection and is available at http://roadanalysis.uah.es. Table I shows the meaning of traffic signs according to its color and shape. All the signs and properties described are for the Spanish traffic signs.

This image database test set is divided into several categories aiming to cover the most important and characteristic problems present in traffic sign recognition systems. For this reason, seven different categories has been created, according with the following list.

- Different shapes
- Different signs
- Different positions
- Rotation
- Occlusion
- Different sizes

Fig. 1. Images from category "Different shapes"

Fig. 2. Images from category "Different signs"

- Deteriorated signs

Figures 1 to 7 show a pair of images from each category. A complete description for each category can be found in [7]. This test set will be the one used to evaluate the performance of the algorithm in Results section.

III. Shape classification

Figure 8 shows the different steps applied to each image in order to, first, get every possible blob that may be a traffic sign, and second, classify the shape of each blob into one of the reference shapes. These reference shapes are, of course, the possible shapes used to build traffic signs. The more frequently used shapes are the circle, the equilateral triangle, the rectangle and the octagon. We will also include in this set the semicircle, although not being a sign shape itself, the segmentation process can yield two semicircles

Fig. 3. Images from category "Different positions"

Fig. 4. Images from category "Rotated sign"

Fig. 5. Images from category "Occlusion"

Fig. 6. Images from category "Different sizes"

instead of a circle for some kind of signs. As we will see afterwards, we will transform every blob onto a new one with its eccentricity equal to one, so the reference shapes must also have the same value for its eccentricity [10]. Therefore, the final reference shapes will be the circle, the equilateral triangle, the square, the octagon, and a semi ellipse with the appropriate parameters.

A. Segmentation

The first step is the creation of a mask where pixels of the image that may belong to a traffic sign were marked as object pixels, whereas pixels that may not belong to a traffic sign were marked as background pixels. To achieve this task, four different color-based segmentations are performed over the whole original image, taking advance of the most frequently used colors in traffic signs. According to this, red-based, blue-based, yellow-based and white-based segmentation will result in four different masks, where possible traffic signs will appear as spots. A complete description of this process can be found in [6]. For our final purpose of detection and recognition of traffic signs, we need to have into account that each sign may

Fig. 7. Images from category "Deteriorated signs"

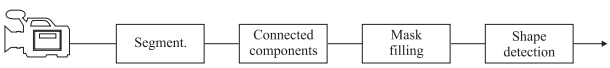


Fig. 8. Block diagram

result in more than one blob, one, or even more, for each color which the sign is painted with.

B. Connected components

The second step consists on the computation of the connected components, in order to get a initial list of all possible objects in the image. Apart from detecting objects from connected components, in this step it is also determined the bounding rectangle and the mask area for each object. After this initial list is obtained, noisy objects can be left out from the list according to its size. In this filter, objects whose bounding rectangle does not exceed a minimum size in either direction, or whose area does not exceed a minimum area, are eliminated form the list. Also, large objects will undergo a similar filter, eliminating those whose area is greater than a maximum area, or whose bounding rectangle is larger than a maximum size in either direction. Lastly, it must be taken into account that this step have to be made for each mask generated in the segmentation process, in our case, the red, blue, yellow and white mask.

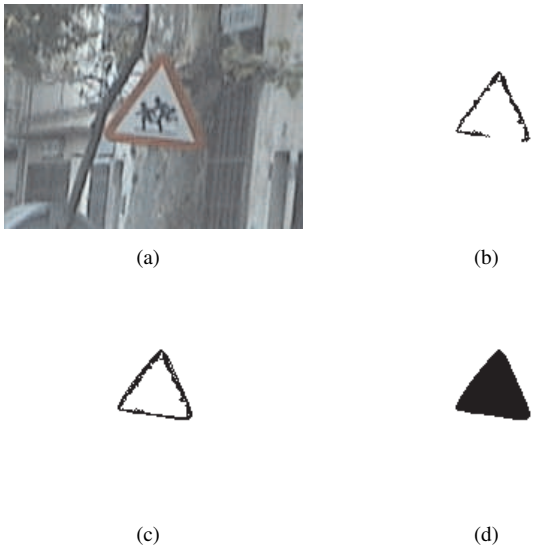


Fig. 9. Filling algorithm. (a) Original image, (b) White based segmented mask, (c) Bounded mask, (d) Filled mask

C. Mask filling

As we will see in next section, the classification algorithm requires a completely filled mask in order to get an appropriate version of the signature of the blob. For this reason, we need to fill all inner parts of the object in the mask given by the segmentation process. The main reasons for the appearance of hollows are the followings. Firstly, due to the fact that the segmentation is performed according to one color, and because traffic signs are composed of more than one color, all parts of the sign in a different color than the one used in the segmentation process would not be considered as part of the object. Fortunately, these parts of the sign belong, normally, to the inner part of the sign, and these parts are easy to fill. Secondly, the hollows can be generated for irregularities on the segmentation process, deteriorated sign or occlusions. Thanks to this, the mask filling algorithm can improve the classification results when some kind of occlusions occur to the sign, or even overcome segmentation problems produced by a non perfect segmentation process that leaves open boundaries.

In the filling algorithm, we consider that all the objects of interest are not concave, as we can assert at least for the reference shapes. The filling algorithm consists of two steps. The first step is designed to close all possible open boundaries and eliminate the concave parts of the mask. Once we have a perfectly close boundary, the second step finally fills all the area defined by the boundary created on the previous step. Figure 9 shows an example of this algorithm. In *a* we can see the original image, corresponding in this case to a triangular sign. This image has been extracted from one image of the database described previously. In *b* it is shown a not completely filled mask, in this case with an open boundary. In this particular case, the mask corresponds to the white segmentation. In *c*, the same mask has a well defined boundary, as it is the function of the first step. Finally, in *d* we have the mask completely filled, and ready to be processed by the shape classification algorithm.

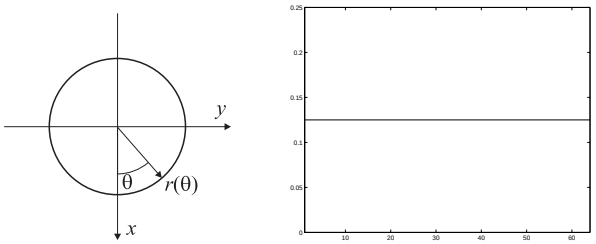
D. Shape classification

The final goal of this paper is shape classification of traffic signs, and that decision is performed on this last step. In this case, the decision is made in base of comparisons between the signature of the blob and the signature of the theoretical shapes we are looking for. Figure 10 shows all the reference shapes mentioned above, and the signature for each, extracted from the mass center (μ_{10}, μ_{01}) of the object [11]. As it will be described later, all signature signals have a total of 64 samples, starting at -π radians and ending at π radians. To make the algorithm robust to object rotations, which become circular shifts in the signature, we compare the absolute value of the FFT of the signatures instead of the signatures themselves, taking advance of the property of the DFT in the presence of shifts.

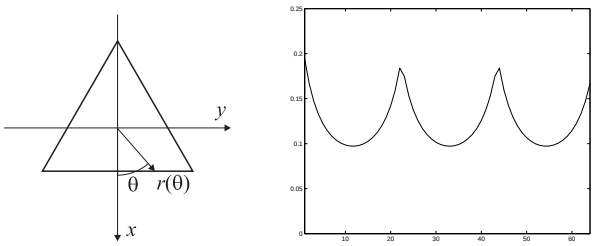
To be able to evaluate the actual sign independently of projection dirtortions, we have to deal with object deformations. So, before we can compute the blob signature, and hence, its FFT, we have to undo every possible deformation. In order to correct the deformation, the object orientation defined as the angle of axis of the least moment of inertia is computed, according to [10]:

$$\theta = \frac{1}{2}\arctan\left(\frac{2\cdot\mu_{11}}{\mu_{20}-\mu_{02}}\right)$$

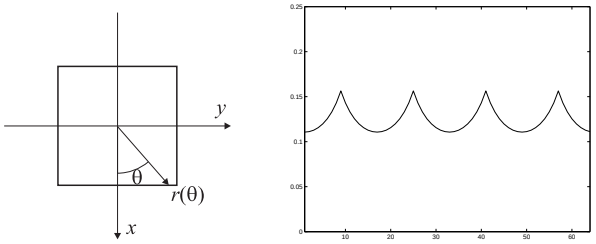
where μ_{11}, μ_{20} and μ_{02} are the central moments of the object. Once the orientation is obtained, a new coordinates system is created, being *u* the coordinate in the orientation direction, and *v* the coordinate perpendicular to coordinate *u*, as can be seen in figure 11 *a*. From now on, we denote μ^{uv} as a moment in the *uv* space, and so on. The transformation matrix in this case is:



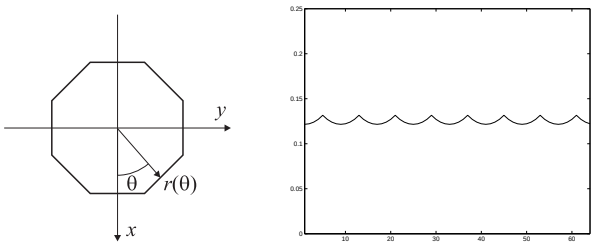
(a) Circle



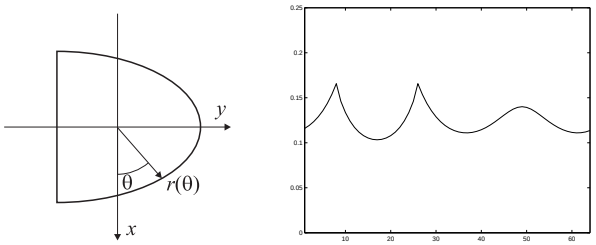
(b) Triangle



(c) Square

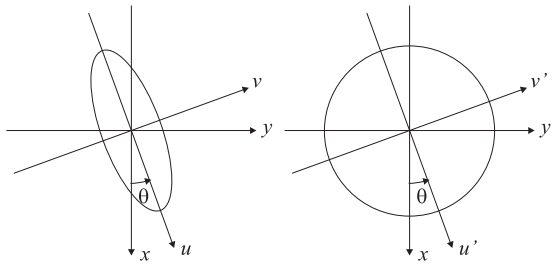


(d) Octagon



(e) Semi ellipse

Fig. 10. Reference shapes.



(a) (b)

Fig. 11. Coordinates systems.

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}$$

In this new coordinates system the second order moments μ_{20} and μ_{02} can be computed from the moments in the *xy* coordinates system according to these expressions:

$$\mu_{20}^{uv} = m_{20}^{uv} - m_{00} \cdot (\mu_{10}^{uv})^2$$
$$\mu_{02}^{uv} = m_{02}^{uv} - m_{00} \cdot (\mu_{01}^{uv})^2$$

where

$$m_{20}^{uv} = m_{20}^{xy} \cdot \cos^2\theta + m_{02}^{xy} \cdot \sin^2\theta + 2 \cdot m_{11}^{xy} \cdot \cos\theta \cdot \sin\theta$$
$$m_{02}^{uv} = m_{20}^{xy} \cdot \sin^2\theta + m_{02}^{xy} \cdot \cos^2\theta - 2 \cdot m_{11}^{xy} \cdot \cos\theta \cdot \sin\theta$$

and

$$\mu_{10}^{uv} = \mu_{10}^{xy} \cdot \cos\theta + \mu_{01}^{xy} \cdot \sin\theta$$
$$\mu_{01}^{uv} = -\mu_{10}^{xy} \cdot \sin\theta + \mu_{01}^{xy} \cdot \cos\theta$$

Finally, these three values lead us to a new coordinates system *u'v'*, being *u'* the same as *u* and *v'* equal to *kv*, where *k* comes from the relationship between the second order moments in the *uv* coordinates system, that is, its eccentricity, according to this expression:

$$k = \sqrt{\frac{\mu_{20}^{uv}}{\mu_{02}^{uv}}}$$

This expression will always give a value of *k* greater than 1, since this is imposed by the orientation angle examined before. This value allows a stretch in the *v* direction, so that the eccentricity of the blob disappears and the deformation done by camera projections can be, in a great amount, corrected, as can be seen in figure 11 *b*.

Finally, the relationship between this coordinates system and the image coordinates system is given by the following matrix transformation:

$$\begin{pmatrix} u' \\ v' \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta \\ \frac{-\sin\theta}{k} & \frac{\cos\theta}{k} \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}$$

Before we can compare the FFT of the blob signature with all the reference FFT, and since the ratio between the second order moments of the blob in the *u'v'* coordinates

system is equal to one always, due to the previous steps, we need to modify the signatures for each reference shape whose eccentricity does not equal 1. This happens solely to the semicircle, since the rest of the figures all have a ratio equal to 1. So, in figure 10 the fifth signature actually corresponds to a semi ellipse instead of a a semicircle, according with the previous discussion.

We also need to normalize all the reference signatures and the blob signature before we can compare the last signal with the reference signals. For the normalization process, we compute the total energy of each signature and divide each sample by the square root of this energy, so that the new signal has a total energy of 1. Different energies in the signature of the blob are due exclusively to different sizes of the blobs, therefore making the system invariant to scaling.

The final step is the comparison of the FFT of the blob signature with all the reference FFTs. In order to increase the speed of the process, a power of 2 number of samples of the signature is chosen, in this case 64. We have to take into account that the signature must be obtained in the new coordinates system, in order to avoid deformation problems. Once the signature is obtained, the absolute value of the FFT computation is done to that signal, and finally compared with all the reference FFT, previously computed. The comparison can be done simply by computing the sum of the absolute difference for each FFT sample. However, the continuous component must not enter in this comparison, since its value is greater enough compared with the rest of the FFT. Also, only half of the spectrum is needed, since all signals are real, and hence, the absolute value of their FFT will be always symmetrical. Logically, the reference signal with a lower value on the comparison is chosen as the candidate shape for that blob.

Figure 12, 13 and 14 show three successful examples of blob signatures for three different shapes. In all examples, we tray to show the robustness of the system to projection deformations, since all signs are clearly deformed. In *a* it is shown the original image, in *b* it is shown the filled mask obtained for the red segmentation, and in *c* is the signature for the blob shown in *b*.

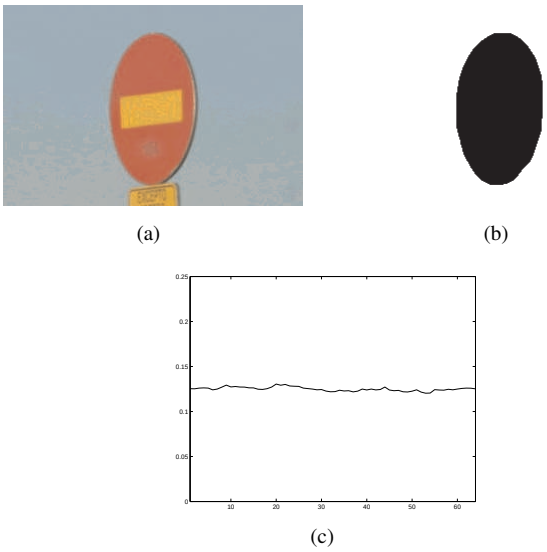


Fig. 12. Circle example: (a) Original image, (b): Red mask, (c) Blob signature

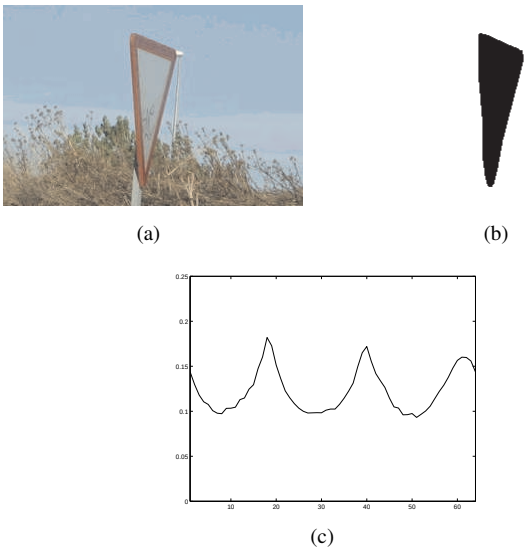


Fig. 13. Triangle example: (a) Original image, (b): Red mask, (c) Blob signature

IV. Results

Table II shows the results for each category in the test set. The parameters evaluated are, first, the classification success probability as the ratio between the signs whose shapes are correctly detected and all the detected signs in the segmentation process. The second parameter is the number of false alarms yielded by the system. And the third parameter expresses the loss probability, as the ratio between the number of signs not detected by the algorithm, and the total amount of signs actually present in the images. This last number was determined from manual inspection, and includes all signs visible in the scene, not caring wether it is small, occluded or deteriorated. The image database test set consists on about 300 images.

Num. Img.	Category	Subcateg.	Success Prob.	False Alarms	Loss Prob.
30	Dif. shapes	Circular	100 %	72	26 %
30	Dif. shapes	Octagonal	97 %	111	0 %
30	Dif. shapes	Rectang.	97 %	188	3 %
30	Dif. shapes	Triangular	96 %	175	10 %
40	Dif. positions	-	95 %	202	11 %
30	Rotation	-	92 %	207	21 %
37	Occlusion	-	63 %	239	24 %
40	Dif. sizes	-	90 %	141	37 %
23	Deter. signs	-	83 %	164	8 %

TABLE II
RESULTS FOR EVERY CATEGORY

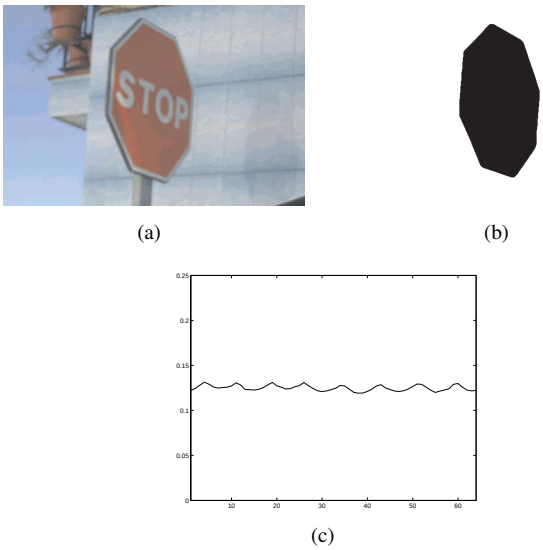


Fig. 14. Octagon example: (a) Original image, (b): Red mask, (c) Blob signature

From table II, we can conclude that the results shows a good performance in all categories except for those sign affected by occlusions and for deteriorated signs. This can be expected since occluded signs yield a completely different shape and, although being correctly classified the occluded shape, this shape does not correspond to the actual sign shape. The same can be said for the deteriorated signs category. We also have to take into account that the images used in the experiments are realistic ones, that includes different illuminations, occlusions and other kind of problems, that makes the success probability decrease.

V. Conclusions and future work

Shape classification is one of the most important steps in the process of traffic signs recognition. In fact, shape classification can be considered as the step where traffic signs are located in the image. Appropriate shape classification is essential in a traffic signs recognition system, since an error on this step leads to subsequent errors in the following steps, an a global error in the recognition process. In this paper we have proposed a method that is, in a great amount, invariant to any kind of distortion projections, like rotation, scaling and deformation, taking advance of the invariance to rotation of blob signatures. Furthermore, the proposed method does not need training set, since only theoretical signatures are considered. The main drawback of this method is the high false alarm probability, that has not been taken into account in this paper.

The direction of our future work will be, first, the reduction of the false alarm probability, extracting more parameters from the blob than the one used here. The second future work is the recognition of the sign, according with its shape, its colors and its contents. Another important issue is the reduction of loss probability, that is caused essentially in the segmentation process. So, it is also important the improvement of the segmentation step, especially in the presence of noise, shadows, different illuminations, and some other kind of situations. Finally, inter frame information can be used to improve the results in a sequence of consecutive images, especially the false alarm and loss probability.

Acknowledgment

This work was supported by the project of the Ministerio de Educación y Ciencia de España number TEC2004/03511/TCM.

References

[1] A. de la Escalera, J. M. Armingol, and Mata M. Traffic sign recognition and analysis for intelligent vehicles. *Image and vision computing*, 21:247–258, 2003.

[2] A. de la Escalera, J. M. Armingol, J. M. Pastor, and F. J.. Rodríguez. Visual sign information extraction and identification by deformable models for intelligent vehicles. *IEEE trans. on intelligent transportation systems*, 5(2):57–68, 2004.

[3] C. Fang, S. Chen, and Fuh C. Road sign detection and tracking. *IEEE trans. on vehicular technology*, 52(5):1329–1341, September 2003.

[4] C. Y. Fang, C. S. Fuh, P. S. Yen, S. Cherng, and Chen S. W. An automatic road sign recognition system based on a computational model of human recognition processing. *Computer Vision and Image Understanding*, 96:237–268, August 2004.

[5] A. Farag and A. E. Abdel-Hakim. Detection, categorization and recognition of road signs form autonomous navigation. *Proc. of ACIVS*, pages 125–130, September 2004.

[6] S. Lafuente-Arroyo, P. García-Díaz, F. J. Acevedo-Rodríguez, P. Gil-Jiménez, and S. Maldonado-Bascón. Traffic sign classification invariant to rotations using support vector machines. *ACIVS'04*, August 2004.

[7] S. Lafuente-Arroyo, P. Gil-Jiménez, R. Maldonado-Bascón, F. López-Ferreras, and S. Maldonado-Bascón. Traffic sign shape classification evaluation I: SVM using distance to borders. *2005 IEEE Intelligent Vehicles Symposium*, June 2005. Preprinted.

[8] A. de la Escalera, L.E. Moreno, M. A. Salichs, and J.M. Armingol. Road traffic sign detection and classification. *IEEE trans. on industrial electronics*, 44(6):848–859, December 1997.

[9] J. Miura, T. Kanda, and Shirai Y. An active vision system for real-time traffic sign recognition. *Proc. IEEE Inteligent transportation systems*, pages 52–57, october 2000.

[10] A. K. Jain. *Fundamentals of Digital Image Processing*. Prentice Hall, 1989.

[11] R.C. González and R.E. Woods. *Digital Image proccesing*. Addison-Wesley, 1993.