Proceedings of the
2007 IEEE Intelligent Vehicles Symposium
Istanbul, Turkey, June 13-15, 2007

Real Time Road Signs Recognition

Alberto Broggi, Pietro Cerri, Paolo Medici, Pier Paolo Porta
VisLab – Dipartimento di Ingegneria dell'Informazione
Università degli Studi di Parma, ITALY
WWW home page: http://vislab.it
{broggi,cerri,medici,portap}@vislab.it

Guido Ghisio
Magneti Marelli Electronics Systems
Strategic Innovation & IP Mgt.
Venaria Reale (Torino), ITALY
guido.ghisio@mmarelli-se.com

***Abstract*— This paper presents a road signs detection and classification system based on a three-step algorithm composed of color segmentation, shape recognition, and a neural network. The final goal of this algorithm is to detect and classify almost all road signs present along Italian roads.**

Color segmentation was suggested by the aim to achieve real time execution, since color-based segmentation is faster than the one based on shape. In order to save computational time, only the RGB color space, directly supplied by the chosen camera, or color spaces that can be obtained with linear transformations, are considered.

Two different methods are used for shape detection, one is based on pattern matching with simple models and the other one is based on edge detection and geometrical cues.

The complete set of signs taken in account has been divided in several categories according to their shape and color. Finally for each road signs set a neural network is built and trained.

I. Introduction

Automatic traffic signs detection and classification is a very important issue for Advanced Driver Assistance Systems (ADAS) and road safety: different road signs detectors were developed in the last 10 years [1]. Most of the industrial systems developed are based only on speed limit signs recognition, on the contrary the system proposed here can detect a large scope of road signs. Moreover the system works with a camera already mounted on-board for other purpose such as lane departure warning (LDW). Another advanced feature introduced here is the low dependence from illumination conditions: this is of paramount importance for good performance in early mornings and late afternoons where sunlight usually presents an appreciable deviation towards red.

Missed signs can cause dangerous situations or even accidents. An automatic road sign detection system can be used both to warn drivers in these situations, and to supply additional environmental information to other on-board systems such as ACC[1].

Both gray-scale and color cameras can be used for this purpose; in the first case the search is mainly based on shape and can be quite expensive in terms of computational time [2], [3]. Using a color camera, the search can be based mainly on color: color segmentation is faster than a shape detection, althrough requiring additional filterings. Images acquired by an inexpensive color camera can suffer from bayer conversion artifacts and other problems such as color balance, but anyway, the developed system is more robust and definitely faster. Some research groups have already used color images for traffic signs detection. Most of these methods have been developed using color base trasformations; HSV/HSI color space is the most used [4], [5] but other color spaces, such as CIECAM97 [6], can be used as well. These spaces are used because chromatic information can be easily separated from the lighting information: this is used to detect a specified color in almost all light conditions. Anyway traffic signs can be detected in RGB [7] or YUV [8] color space with the advantage that no trasformation, or just a very simple one, is required. The segmentation and thresholding algorithms are anyway more complex, but a lot of computational time can be saved. In order to make the detection more robust, both color segmentation and shape recognition can be used in cooperation [6]. Anyway the processing described so far has to be computationally light to keep the advantages of the selected color space.

Many different approaches are used for the subsequent classification: most of them are based on artificial intelligence techniques; the most used are neural networks [9], [15], bayesian networks, and fuzzy logic [10].

The proposed approach is based on 3 steps (see figure 1): color segmentation, that is presented in the following section, shape detection (section III) and classification, (section IV) based on several neural networks. Section V concludes the paper with a discussion on results and performance.

This paper is mainly focused on the first two steps; the last one has been implemented to test the behavior of the

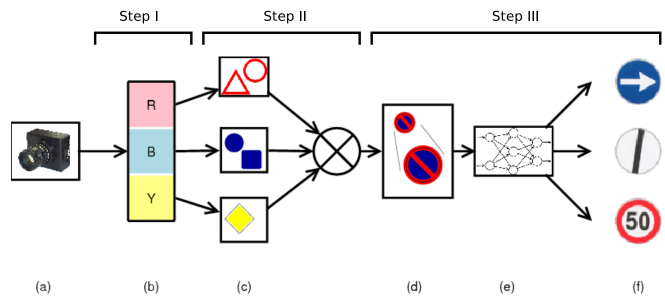

Fig. 1. Algorithm flowchart. (a) Image acquisition; (b) color segmentation; (c) shape detection; (d) bounding box scaling; (e) classification; (f) output.

[1] Automatic Cruise Control

1-4244-1068-1/07/$25.00 ©2007 IEEE.

Fig. 2. Examples of different illumination conditions.

formers with the result of a generic classification stage.

II. Color analysis

In this section the first step of road signs detection based on color information is presented. One of the main road sign feature is their immediate identification by a human driver: this is due to a limited number and very specific set of shapes and colors. In particular, the colors set used for road signs is composed of: white, grey, red, yellow, and blue; green used on highway signs (in Italy) is not considered in this analysis.

This section presents a discussion on the choice of the color space to use, a robust color segmentation, and a solution for the problem of cromatic predominance of the light source.

A. *Color segmentation*

The specific illumination condition deeply affects the road signs color perception. Common enviromental conditions are usually characterized by a wide range of different illuminations: direct sunlight, reflected sunlight, shadows, and sometimes even different illuminations can coexist on the surface of the same sign as shown in figure 2.

The objective of this work is to identify a sign of a given color (for example red) regardless of its illumination. As already mentioned, in literature most approaches are based on HSV or HLS color space [5], [8], [10] but the camera used for this application has a raw Bayer output and a conversion would be too computationally expensive because of the non-linearity introduced. Therefore RGB space is too dependent on brightness, so different color bases were tested to find one that is less dependent on brightness. All the color base conversions evaluated can be obtained with linear transformations; this kind of approach has been followed also by [7]–[9], [11], [12].

First YUV color space has been tested, using the matrix shown below:

$$\begin{pmatrix} Y \\ U \\ V \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.147 & -0.289 & 0.436 \\ 0.615 & -0.515 & -0.100 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad (1)$$

The Y coordinate is strongly dependent on brightness thus, considering only the UV plane, it is possible to bound regions mainly based on hue. Figure 3 shows the UV plane with Y=127: the grey color is visible in the center of the image.

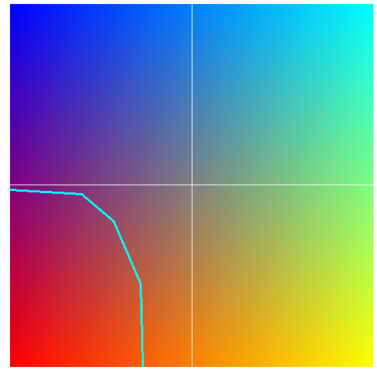

Fig. 3. The UV plane in the YUV color space with Y=127. The blue line bounds the region selected to identify the red color.

Empiric tests demonstrated that this kind of bounding is too simple to collect all the cases of different illumination and to cover most of the case study.

Our second attempt focused on RGB values: what we bound here is the ratio between different channels and not only the channel itself. Eq. 2 shows the expression of this kind of thresholding:

$$\begin{cases} \alpha_{min} * G & < & R & < & \alpha_{max} * G \\ \beta_{min} * B & < & R & < & \beta_{max} * B \\ \gamma_{min} * B & < & G & < & \gamma_{max} * B \end{cases} \quad (2)$$

The values of the parameters involved have been obtained tuning the algorithm on real images in different conditions.

Both these methods have produced good results but we choose the second one for several reason:

- lower number of false positives,
- lower computational load: indeed no color space conversion only a thresholding is needed,
- easier to tune: because it is less sensible to small parameters variations.

After an analysis on colors present in italian traffic signs the research has been focused on 3 colors: red, blue, and yellow; the described algorithm is applied on all these 3 colors to generate 3 binary images containing only the pixels referred to that color; figure 4 shows the results for red and blue.

All the pixels referred to a same connected region are labeled together. Regions smaller than a fixed threshold are discarded because usually do not represent a road sign or,

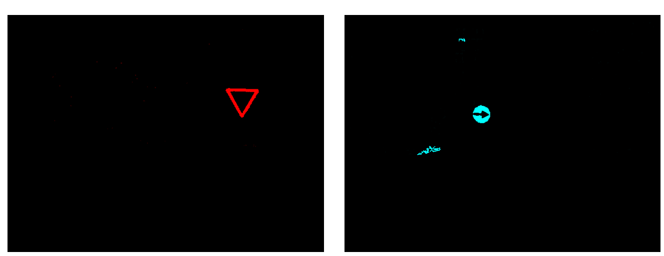

Fig. 4. Color segmentation of red and blue channel.

Fig. 5. Original image and result of chromatic equalization. In the right image the region supposed to frame the pave is shown.

otherwise, the target is so far that a detection whould not be followed by a succesful classification due to reduced size.

Considering for example the red color segmentation, all the red objects present will be detected (e.g. cars, buildings, placards), but these detections will be discarded by the following further steps.

B. Chromatic equalization

One of the main problems experienced in this stage is the dependence on the color of the light source. For example during sunset or dawn a red color predominance is present and this deeply affects the color segmentation step; see figure 5.

To solve this problem we developed a chromatic equalization based on two steps:

- light source color identification,
- chromatic correction.

The easiest way to find the light source color is to find an object supposed to be white and then compute the aberration from theoretical white (255, 255, 255 in the RGB color space). Unfortunately on a dynamic enviroment such as roads, it is difficult to have a white reference point. Thus we searched for the color of the road, as suggested in [13] that is supposed to be grey. In [13] the chromatic response of several materials has been analyzed in different illumination conditions. Moreover in most cases in our vehicular application a specific region of the image frames the pave as shown in figure 5.

Through the use of a temporal window in which we integrate the light source color, it is possible to avoid fast changings of the result and keep it stable. In case of a tracking has to be introduced in the processing chain, it is very important to have stable conditions for a reasonable number of frames.

Once the light source color has been evaluated, chromatic equalization can be applied. This step is very similar to a gamma-correction process: to reduce the computational time a linearization of the gamma correction function has been used, as shown in figure 6 and described below.

line A: $y = \alpha \cdot x$
line B: $y = \beta \cdot x + \gamma$
line C: $y = -x + k$
D: point of coordinates (255, 255).

Now suitable values for the parameters α, β, γ and k have to be found. The α value for the 3 channels can be computed in 3 steps:

- consider the RGB value of the light source color,
- set $\alpha = 1$ to the channel that has the intermediate value,
- set $\alpha = \frac{\textit{intermediate channel value}}{\textit{channel considered value}}$ for the other 2 channels.

The parameter k can be set once with empiric tests, to avoid saturations and β can be obtained forcing the curve to be continue and to reach D.

III. Shape Detection

After color segmentation, a first sorting based on shape is performed. This sorting is developed in order to reduce the complexity of the final classification. Two different methods are used to determine the correct shape with high reliability; the first one is based on pattern matching, the other one is based on remarks about edges. Before this sorting, a method to merge and split the bounding boxes generated by color segmentation is applied together with a filtering based on aspect ratio.

A. Bounding Boxes merge and split

Color segmentation can sometimes provide a bounding box that contains two or more signs or only a part of a sign; the use of such a bounding box in the subsequent classification may cause an error: this merge and split step is developed in order to solve this problem.

When two signs with the same color are hanged on a single pole it may happen that the segmentation identifies the two signs as a single one because of the weak separation between the signs. All the boxes with height almost double as width are checked considering the vertical histogram of the binarized image. If a very low value is identified around the middle of the histogram, the bounding box is splitted, see fig. 7.

On the other hand, bounding boxes of the same color, that overlap more than a threshold, are merged together as a single sign. This process is useful to merge in a single box different parts of a same sign that may have been divided by the labeling step.

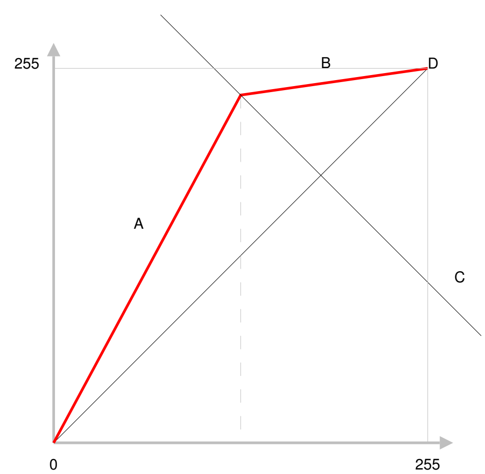

Fig. 6. Gamma correction curve.

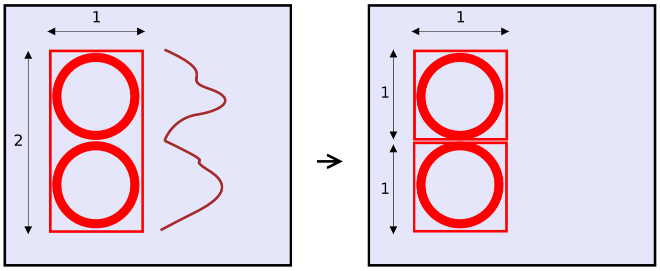

Fig. 7. Bounding boxes split.

A single sign can also contain two colors that can be identified in the two corresponding images, e.g. work in progress signs contain both red and yellow. All the three color segmentation images are checked for overlapping bounding boxes and, if overlapping bounding boxes are found, they are merged in a single bounding box. Each bounding box, can handle up to two colors, a primary and a secondary one; if a bounding box is the result of a merging, its pimary color will be set to the color of the larger bounding box while the other bounding box will set the secondary color. If a bounding box does not overlap any other box, it will have only a primary color.

B. Pattern matching

Bounding boxes with primary color red or yellow are supposed to have specific shapes that can be easily detected using a pattern matching. A reference pattern is built for each shape that have to be detected. The patterns are built growing the shape of a signs, with the aim of detecting also rotated or misaligned signs; these patterns are shown in figure 8. Triangle, reversed triangle, circle and filled circle are searched in red bounding boxes, while only rhomb is searched in yellow ones. Filled circles are used to detect stops and no thoroughfare signs.

All the detected bounding boxes are resampled to a fixed size (50×50 pixels) equal to the pattern size. A very simple pattern matching is used in order to reduce the complexity and the computational time: the matching function counts the number of pixels that are positive both in the pattern, that is a binary image, and in the image of the corresponding color. The ratio of matched pixels to white pixels in the considered pattern is computed. When all the patterns have been checked, the shape that provides the best ratio is choosen as the correct shape of the sign, only if the ratio exceeds a fixed threshold. If the choosen shape is a circle, the ratio of matched pixels to white pixels in the image of the corresponding color is considered as well, in order to distinguish between circles and filled circles.

(a) (b) (c) (d) (e)

Fig. 8. Patterns of searched shapes: (*a*) circle, (*b*) filled circle, (*c*) triangle, (*d*) reversed triangle, (*e*) rhomb.

C. Remarks about edges

A double check is extremely useful to be sure that the choosen shape is the correct one. Starting from the center of each bounding box in the segmented image and expanding to the borders in several directions, the last pixel of primary color is searched for in order to find the external edges of the labeled region inside the bounding box. The set of points found is used to generate lines or a circle that best fits the set. Appropriate geometrical formulas are used to determine if the lines are composing one of the searched shapes. The last step is to find the best match between all this possible cases.

The result of this processing is used to dinamically modify the threshold of the pattern matching previously described: if the two methods agree the pattern matching threshold is reduced, otherwise it is raised.

Since blue signs are not suitable to be detected with pattern matching because their shape fills most of the bounding box, an ad-hoc method has been developed. It consists in searching for the blue pixel nearest to each corner. According to the distances found we can discriminate between squares, circles and other irregular shapes.

IV. Classification

As already mentioned a specific neural network has been developed for each sign category. A single neural network would have been too complex and heavy; on the other hand our real-time approach direct searches signs in the correct network. The categories considered are:

- obligation signs (blue circles),
- prohibition signs (red circles),
- danger signs (red triangles),
- indication signs (blue squares),
- no parking, no waiting signs (red and blue circles),
- stop, no access (red filled circles),
- work in progress signs (red and yellow triangles)
- priority signs (yellow rhombs),
- yeld signs (red reverse triangles).

Concerning the last two categories, no neural network is needed since they are the only signs with a reverse triangle or a rhomb as shape.

The network choosen is the LWN++ [14] that is an open source implementation of a backpropagation neural net with some examples on images classification.

The net has been designed as follows: the input stage is composed by 2500 neurons, corresponding to the number of pixels both of the models and the resized bounding boxes (50×50). The number of neurons in the output stage is equal to the number of signs included in the category considered and corresponds to the probability (from 0 to 1) that the considered sign is the one corresponding to the output. Finally there is only one hidden stage and its neurons number is computed trying different solutions; the values used for the considered cases vary between 30 and 100.

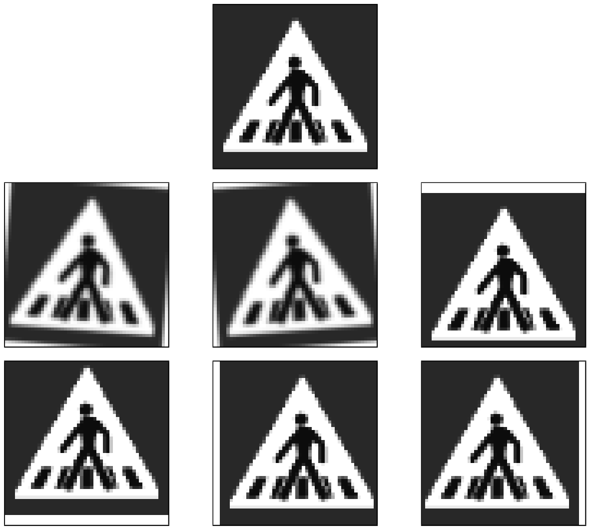

Fig. 9. Examples of variations, rotations and traslations, of each model included in the training set. The Original model is shown in the image at the top.

The net training is performed sepratately for each net using a set of synthetic images for each sign present in the categories; a classifier based on synthetic images is presented also in [15]. The set of images are built starting from a strandard image, provided by Italian department of trasportation; the image is converted to grey-scale and then rotated and traslated in the bounding boxes, as shown in figure 9, simulating rotated signs or inaccurate localizations.

When training the net to identify a specific sign, the output corresponding to that sign is fixed to 1 and the other ones to 0. We choosed this approach even if in the litterature it is warn of produce overfitting, and we obtain correct results.

The net obatined by the training is copied in the algorithm branch and used in the classification step. Before the bounding box is passed to the neural network, it has to be modified in order to become the most similiar as possible to the images used for the training. The steps needed to achieve this goal are:

- border deletion, for the signs that have a fixed border, as for example prohibition signs;
- gray-scale conversion;
- 50×50 resampling;
- contrast stretching.

The last operation is the most critical one since there are signs with two or more colors. In particular contrast stretching is performed searching for the two or three peaks (depending on the number of colors of the sign) present on the image histogram and moving them in fixed positions estracted from the models analysis.

V. Results and Conclusions

In this section some results of several tests are presented. The development of neural networks for sign classification is not complete, and only some categories are recognized: obligation signs, prohibition signs, yeld signs.

Tests have been peformed in several situatons, with different illumination conditions. Figure 10 shows some examples of different types of signs. The black stripe placed below each frame is divided in two lines: the top line shows all the signs detected in that frame scaled to a 50×50 pixels image, while the bottom line shows all the corresponding models for the signs that have been classified.

All kinds of sign are correctly detected, even in some ambiguous cases such as figure 10.f. Generally, signs are recognized when they are relatively close to the vehicle (e.g. 20 m) and appear not too misaligned with the camera; for example the perspective deformation of the yeld sign in figure 10.l does not allow the system to detect it, while the same sign in figure 10.j is correctly detected. Empirical tests demonstrated that signs can be detected up to 30 meters ahead. However this distance can be increased reducing camera focal lenght but dropping the possibility to detect signs that are close and at the side of the vehicle.

All the categories of signs are correctly classified except danger signs, because the neural net devoted to this class has still to be developed. Since the development of all the parts is not completely finished, precise benchmark are not available at the moment, however cases of misclassification or missed detection are rare. Final results and benchmarks will be added in the final version of this paper.

During all the development process, low computational time has been one of the issues to follow. Even if code optimization has still to be completed at the time of writing the paper, on a Pentium 4 at 3 GHz the algorithm runs faster than 10 Hz.

One of the main problems encountered is the non persistent detection of signs in subsequent frames: this is mainly due to the background and illumination changes. As a future improvement a tracking step is under evaluation. Other possible research fields can be the use of other techniques such as bayesian networks or different implementation of the neural network in order to make the current classification more robust.

Another open problem is to understand wether a detected sign refers to the driver or not: it can happen, especially in junctions, that a sign is seen by drivers running on another road. This problem can be solved only if the system can perceive the road and junction structure. It is not unusual that a sign is mounted in a wrong way as, for example, shown in figure 10.j.

References

[1] Y. Nguwi and A. Kouzani, “A Study on Automatic Recognition of Road Signs,” in *Procs. IEEE Conference on Cybernetics and Intelligent Systems*, Bangkok, Thailand, June 2006, pp. 1–6.

[2] D. Gavrila, “Traffic Sign Recognition Revisited,” in *Procs. of the 21st DAGM Symposium fr Mustererkennung*, Bonn, Germany, 1999, pp. 86–93.

[3] G. Loy and N. Barnes, “Fast Shape-based Road Signs Detection for a Driver Assistance System,” in *Procs. IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, Sendai, Japan, Sept. 2004, pp. 70–75.

[4] S. Vitabile, G. Pollaccia, and G. Pilato, “”Road signs recognition using a dynamic pixel aggregation technique in the HSV color space,” in *Procs. of Intl. Conf. on Image Analysis and Processing*, Palermo, Italy, Sept. 2001, pp. 572–577.

Fig. 10. Results on several conditions and different signs. Different bounding box colors indicate primary and secondary color assigned with color segmentation. (*a*) and (*d*) pool of two signs; (*b*) prohibition sign detected at high distance; (*c*) and (*d*) prohibition sign with two colors; (*e*) red sign with red building on the background; (*f*) sign on a placard; (*g*) indication sign; (*h*) danger signs correctly detected but not classified; (*i*) yeld sign with a small deformation; (*j*) yeld sign in a saturated image and slightly rotated; (*k*) obligation sign; (*l*) yeld sign is not detected because of its high rotation, the blue sign instead, has been recognized even if it is mounted upside-down.

[5] A. de la Escalera, J. M. Armignol, and M. Mata, “Traffic sign recognition and analysis for intelligent vehicles,” *Image and Vision Computing*, vol. 21, no. 3, pp. 247–258, Mar. 2003.

[6] X. Gao, N. Shevtsova, K. Hong, S. Batty, L. Podladchikova, A. Golovan, D. Shaposhnikov, and V. Gusakova, “Vision Models Based Identification of Traffic Signs,” in *Procs. of Europ. Conf. on Color in Graphics Image and Vision*, Poitiers,France, Apr. 2002, pp. 47–51.

[7] A. Soetedjo and K. Yamada, “Fast and Robust Traffic Sign Detection,” in *Procs. IEEE Intl. Conf. on Systems, Man and Cybernetics 2005*, vol. 2, Oct. 2005, pp. 1341–1346.

[8] W. G. Shadeed, D. I. Abu-Al-Nadi, and M. J. Mismar, “Road traffic sign detection in color images,” in *Procs. IEEE 10th Intl. Conf. on Electronics, Circuits and Systems, 2003*, vol. 2, Dec. 2003, pp. 890–893.

[9] A. de la Escalera, L. E. Moreno, E. A. Puente, and M. A. Salichs, “Neural Traffic Sign Recognition for Autonomous Vehicles,” in *Procs. IEEE 20th Intl. Conf. on Industrial Electronics, Control and Instrumentation*, vol. 2, Sept. 1994, pp. 841–846.

[10] G.-Y. Jiang and T. Y. Choi, “Robust Detection of Landmarks in Color Image Based on Fuzzy Set Theory,” in *Procs. IEEE 4th Intl. Conf. on Signal Processing*, vol. 2, Oct. 1998, pp. 968–971.

[11] A. de la Escalera, L. E. Moreno, M. A. Salichs, and J. M. Armignol, “Road Traffic Sign Detection and Classification,” *Industrial Electronics, IEEE Transactions on*, vol. 44, pp. 848–859, Dec. 1997.

[12] F. Jeun-Haii and L. Gang, “A Vision-Aided Vehicle Driving System: Establishment of a Sign Finder System,” in *Procs. Conf. on Vehicle Navigation and Information Systems, 1994*, Aug.–Sept. 1994, pp. 33–38.

[13] S. D. Buluswar, “Color-Based Models for Outdoor Machine Vision,” Ph.D. dissertation, University of Massachusetts Amherst, 2002.

[14] L. Cinti and L. Masetti, “Il progetto lightweight neural network ++: un’implementazione c++ di una rete feed forward con backprpagation,” apr 2004, http://lwneuralnetplus.sourceforge.net.

[15] H. Hoessler, C. Wöhler, F. Lindner, and U. Kreßel, “Classifier training based on synthetically generated samples,” in *Procs. 5th Intl. Conf. on Computer Vision Systems*, Bielefeld, Germany, Mar. 2007.