

Traffic Sign Recognition using Evolutionary Adaboost detection and Forest-ECOC classification

Xavier Baró^a, Sergio Escalera^{a,b}, Jordi Vitrià^{a,b}, Oriol Pujol^{a,b} and Petia Radeva^{a,b}

Abstract—The high variability of sign appearance in uncontrolled environments has made the detection and classification of road signs a challenging problem in computer vision nowadays. In this paper, we introduce a novel approach for detection and classification of traffic signs. The detection process is based on a boosted detectors cascade, trained with a novel evolutive version of Adaboost, which allows the use of large feature spaces. The classification process is defined as a multi-class categorization problem. A battery of classifiers is trained to split classes in an Error Correcting Output Codes (ECOC) framework. We propose a ECOC design by means of a forest of optimal tree structures that are embedded in the ECOC matrix. The novel system offers high performance and better accuracy than the state-of-the-art strategies, being potentially better in case of noise, affine deformation, partial occlusions, and reduced illumination.

Index Terms—Traffic Sign Recognition, Dissociated dipoles, Evolutive boosting, Error Correcting Output Codes, Ensemble of dichotomizers.

I. INTRODUCTION

TRAFFIC sign recognition is studied for several purposes, such as autonomous driving or assisted driving [1], [2]. Recognition of traffic signs allows warning the driver for inappropriate actions and potentially dangerous situations. In the mobile mapping framework, traffic sign recognition methods are used in combination with other methods in order to compile road information and measuring position and orientation of different landmarks in movement either in an aerial or a terrestrial platform. An example of this system is given by Madeira et al. [3], where a mobile mapping system automatically processes traffic signs. In this work, a recognition accuracy over 80% on a reduced set of sign types is obtained. In [4], a vehicle based vision platform is used to detect road signs, where the main goal is mainly focused on speed signs.

In the literature, we can find two main approaches to solve the problem of road sign recognition: color-based and grey scale-based sign recognition. The first one relies on color to reduce false positive results in the recognition process [5], [6], [7], [8], [9], [10], [11], [12], [13], whereas the greyscale methods concentrate on the geometry of the object [14], [15], [16], [17]. Recent works use combination of both cues to improve the detection rates. For instance, in [18] a threshold is applied over a HSV representation of the image to find

regions with high probability of having a traffic sign. As many background objects can share colors with traffic signs, heuristics over the size and aspect ratio are used to reduce the number of false alarm regions. Once the regions are normalized to a predefined size, a linear SVM is used to classify the region in one of the possible shapes, such as circle or triangle. The color and shape information are used as a coarse classification, and finally a SVM with Gaussian kernels is used to perform the fine classification step. Since the color information is strongly related to the type of camera, illumination and sign aging, the use of color information introduces additional difficulties to the recognition process. In the work of A. de la Escalera et al. [19], these difficulties are addressed using an enhancement step previous to the use of thresholds on the color values. After applying size heuristics to remove non-sign regions, the authors use a fusion of color information, the gradient, and a distance image to remove regions with low probability of having a traffic sign. Final classification is performed by means of a Neural Network. Other recent works are focused on the final classification step. The authors of [20] propose a representation of road sign data based on extending the traditional normalized cross correlation approach to a similarity based on individual matches in a set of local image regions.

Traffic sign recognition is a straightforward application for object recognition algorithms in which previous addressing of the category detection (e.g. object location) is often required. In the last years, one of the most accepted and used approaches in the object detection field has been the one proposed by Viola & Jones in [21]. Their approach is based on a cascade of detectors, where each one is an ensemble of boosted classifiers based on the Haar-like features. Lienhart and Maydt [22] presented an extension of the original Haar-like features set, demonstrating that *Adaboost* converges faster and with better results when the features set is large. On the other hand, due to the exhaustive search over the features set, the training time grows with respect to the number of features. This fact makes unfeasible any approach that tries to extend the feature set.

Once an object (traffic sign) is located, it should be recognized from a wide set of possible classes using some kind of classification technique. Designing a machine learning multi-class technique is a hard task. In this sense, it is common to conceive algorithms to distinguish between just two classes and combine them in some way. Following the multi-class categorization problem, where a set of classifiers should learn in a natural way the features shared between categories, the Error Correcting Output Codes technique was proposed with very interesting results [23]. This technique is a very

^a Computer Vision Center, Edifici O, Campus UAB, 08193 Bellaterra, Catalonia (Spain)

^b Dept. Matemàtica Aplicada i Anàlisi, UB, Gran Via 585, 08007, Barcelona, Spain

E-mail: {xbaro,sescalera,jordi,oriol,petia}@cvc.uab.cat

successful multi-class categorization tool due to its ability to share the classifiers knowledge among classes. Recently, the embedding of a tree structure in the ECOC framework has shown to obtain high accuracy with a very small number of binary classifiers [24]. However, the ECOC design is still an open issue.

The goal of this paper is two-fold: first, at the detection step, in order to discriminate a set of traffic signs from the background, we propose a novel binary classifier by means of an evolutive version of Adaboost, which avoids the limitations of the original algorithm in terms of the dimensionality of the feature space. Second, to deal with the multi-class categorization problem, in order to distinguish among a large set of classes, a multi-class learning technique is proposed. The approach is based on embedding a forest of optimal tree structures in the ECOC framework, allowing to share features (tree nodes, base classifiers, or dichotomies) among classes in a very robust way. Finally, we develop a real traffic sign recognition system for a mobile mapping process [25], in which we validate the robustness of our approach. We show that the present strategy obtains high accuracy and outperforms the state-of-the-art strategies, being robust against a high variability of sign appearance.

The paper is organized as follows: section II presents the novel detection and multi-class categorization approaches. Section III shows the architecture of the real traffic sign recognition system and integration details. Section IV shows the experimental results, and section V concludes the paper.

II. METHODOLOGY

In this chapter, we present the new Evolutionary boosting strategy and the Forest-ECOC technique to deal with the object detection and classification stages, respectively.

A. Detection

The detection process takes an image as input and gives at the output the regions that contain the candidate object. Our detector is inspired on the face detector presented by Viola & Jones in [21]. We consider the use of the *attentional cascade* concept, boosting as the feature selection strategy, and the representation of the image in terms of the integral image. With our approach, we solve the limitation of the exhaustive search of the boosting formulation in [21]. Moreover, we present a solution to the restriction of the boosting process computation when the feature set size is large. We propose an evolutive vision of boosting, which not only drastically reduces the training time, but also allows the use of huge feature sets. In this way, we propose the use of the dissociated dipoles, a more general type of features than the Haar-like features [21], which can also be calculated using the integral image. In addition, we introduce another variation in the *Weak Classifier*, changing the decision rule from the threshold value used in the original schema to its ordinal definition, where only the sign is considered. This representation has been demonstrated to be more robust in case of noise and illumination changes [26].

1) *Detection Architecture*: Working with unbalanced problems like object detection, each time we analyze an image, the system must discard a huge number of negative regions while just few or any correspond to the object we look for. The attentional cascaded architecture allows to discard easy non-object regions at low computation cost, while more complex regions are deeply analyzed. An attentional cascade is composed by a set of classifiers or stages, where the input to each classifier corresponds to those regions classified as object-regions by the previous stages. The regions classified as object-regions by the last stage are the output of the detector.

Although any learning method can be used to learn the cascade, the usefulness of Adaboost has been widely demonstrated. Before each stage training process, a new samples set is built using the positive samples and the false positives of the previous stages of the cascade. Then a classifier is learned to achieve a given minimum hit and a maximum false alarm rates. This process is repeated until the desired number of stages or the goal false alarm rate are reached.

2) *Dissociated dipoles*: In [22], Lienhart and Maydt show that the accuracy of the detector increases with the number of available features. Therefore, in this work we use the *dissociated dipoles*, a more general type of features than the Haar-like features, and thus, dealing to a larger feature set.

The *dissociated dipoles* or *sticks* have been presented by Balas and Sinha in [27], and they are a kind of features composed by a pair of rectangular regions, named the *excitatory dipole* and the *inhibitory dipole*, respectively (see Fig. 1). The mean value of all the pixels in the *inhibitory dipole* is subtracted from the mean value of the *excitatory dipole* pixels. As in the case of the Haar-like features, the integral image is used to calculate the sum of the pixels inside the rectangular regions. Similar features are also used in other recent works, such as the one of Bay [28], where weighted dipoles with a fixed position are used in order to describe objects.

In computational terms, the use of dissociated dipoles means

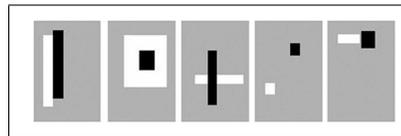


Fig. 1. Dissociated dipoles. The black region corresponds to the *inhibitory dipole* and the white one to the *excitatory dipole* [27].

to increase from the about 600.000 features in the approach of Lienhart to more than 2^{30} features in a training window size of 30×30 pixels, making the classical approach computationally unfeasible. In order to deal with this limitation, we define an evolutionary approach of Adaboost.

3) *Evolutive Adaboost*: Boosting is a powerful learning technique that allows to combine the performance of many simple classification functions or *Weak Classifiers* to produce a *Strong classifier* [29]. At each round of learning, the examples are re-weighted in order to emphasize those which were incorrectly classified by the previous *Weak Classifier*. The final

Strong Classifier is a decision stump, composed by a weighted combination of *Weak Classifiers* followed by a threshold [30]. In the classical boosting approach, an exhaustive search is used to find the best *Weak Classifier*. Therefore, with an enormous features set, this approach becomes computationally unfeasible.

Our *evolutionary weak learner* minimizes the weighted error function ε of the Adaboost scheme as:

$$\varepsilon = \sum_{i:h(x_i) \neq y_i} w_i \quad (1)$$

where $X = \{(x_i, y_i) | i = 1 : m\}$ are the pairs sample-label which compound the training set, $W = \{w_1, w_2, \dots, w_m\}$ is the Adaboost weights distribution over the training set, and $h(x_i)$ corresponds to the label predicted by the hypothesis h for the training object x_i . Although in the rest of this section, we use the original *Discrete Adaboost* algorithm for simplicity, our approach can be applied to any existing variant of Adaboost.

The *Weak Learner* can be seen as an optimization problem, where we need to find the parameters of the Weak Classifier that minimize the error function ε . As this function is defined, it seems logical to consider it as a non-derivative function full of discontinuities. Therefore, the classical approaches based on gradient descend can not be applied on this problem. An alternative solution is the use of an evolutive approach. The most well-known evolutive strategy is the genetic algorithm, which does a search over the spaces of solutions using the three basic concepts of the Darwin's theory: mutation, cross-over, and natural selection.

4) *Weak Classifier*: When we work with evolutionary algorithms, we should define two basic elements: the individuals and the evaluation function. Each individual must represent a point in the space of solutions, and the evaluation function measures the function value at the point represented by the individual. We define the evaluation function as $\mathcal{F}(I) \mapsto \mathbb{R}$, $I \in \mathbb{R}^d$, where I is an individual. In this case, the function we are optimizing is the classification error over a given labeled data set X , using a weights distribution W over the data. Then, the function \mathcal{F} corresponds to ε (eq. 1), where the individual I defines the hypothesis h . Combining both equations, we can write the evaluation function as:

$$\mathcal{F}(I, W, X) = \left(\sum_{i:h(I, x_i) \neq y_i} w_i \right) \quad (2)$$

where $h(I, x) \mapsto \{-1, +1\}$

Note that the function h depends not only on a certain data instance, but also on the individual. At this point, we formulated the evaluation function in terms of the *Weak Learner*. The following step is to determine which parameters are required to define h , or in other terms, to decide the dimension of I . An important consideration when we choose the parameters is to evaluate the relevance of each of them in contrast to the

introduced complexity.

The Viola & Jones definition of the *Weak Learner* consists of a feature and a threshold value. The feature can be parameterized by the upper-left position and size of one of the regions and their type. The size, weight, and position of the other regions that conform the Haar-like feature are fixed by the feature type (see Fig. 2). Given a feature, the threshold value must

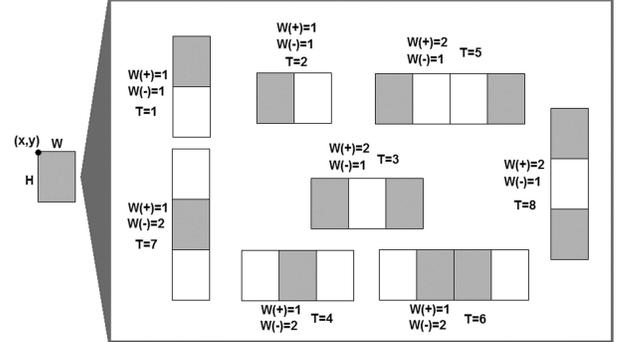


Fig. 2. Haar-like features definition from a given region and the type value.

be learnt using the training samples, applying an exhaustive search over all the possible threshold values to find the one that minimizes the error. Therefore, for the Viola approach, a *Weak Learner* can be defined as $h(I, Thr, x_i) \mapsto \{-1, +1\}$, where $I = (R_x, R_y, R_w, R_h, T)$, Thr is the threshold value, R_x, R_y, R_w , and R_h correspond to the upper-left corner (x, y) , the width, and height of one of the regions, and T is the type of the Haar-like feature (see Fig. 2). Note that the threshold value is not included in the individual I . The reason is that this value is learnt once the other parameters have been fixed. In the case of the Dissociated Dipoles, the regions have no constraints, so, both regions are independently learnt. Using the same reasoning, we define a Weak Classifier based on the Dissociated Dipoles as $h(I, Thr, x_i) \mapsto \{-1, +1\}$ where $I = (Re_x, Re_y, Re_w, Re_h, Ri_x, Ri_y, Ri_w, Ri_h)$, being Re the excitatory dipole, and the type parameter T is changed by the parameters of the inhibitory dipole Ri . This representation can be extended including extra parameters, such as the weights of the regions, which are necessary to represent some types of Haar-like features.

In the previous approaches, the difference between both regions depends on the illuminance conditions. Thus, we need to define a normalization criterion to make the representation invariant to those conditions. In [21], a method so-called *fast lighting correction* is used to deal with the illuminance variations. In addition, the Haar-like approach uses the sum of the pixels to represent the region, which introduces a scale dependence, that must be corrected to deal with the multi-scale detection. The dissociated dipoles are not affected by scale, since the mean value is used instead of the sum.

The above formulations are quantitative comparisons between the regions, not only taking in account which region has a higher value, but also the difference between these values. We can simplify by only using qualitative comparisons, therefore, the calculus of the threshold value becomes unnecessary since we use only the sign of the difference. This approach has two

main advantages: the illuminance normalization is unnecessary, and to remove the threshold learning process reduces the evaluation time. We can rewrite the previous formulations as:

$$h(I, x_i) \mapsto \{-1, +1\}$$

where (3)

$$\begin{aligned} I &= (R_x, R_y, R_w, R_h, T) \\ I &= (Re_x, Re_y, Re_w, Re_h, Ri_x, Ri_y, Ri_w, Ri_h) \end{aligned}$$

5) *Learning Algorithm*: The final approach is summarized in the Evolutionary Adaboost algorithm shown in Algorithm 1. This algorithm is used to learn all the stages of the detection cascade. Using a set $\langle (x_1, y_1), \dots, (x_m, y_m) \rangle$ of samples classified as positive samples in the previous stages of the cascade. This algorithm iteratively use a genetic algorithm to minimize the weighted error and to instantiate the parameters of a new *Weak classifier* which is added to the final ensemble.

Algorithm 1 The evolutive Discrete Adaboost.

Given: $(x_1, y_1), \dots, (x_m, y_m)$
 where $x_i \in X, y_i \in Y = \{-1, +1\}$
 Initialize $W_1(i) = 1/m$
for $t = 1, \dots, T$ **do**
 Use a genetic algorithm to minimize:
 $\epsilon_t = Pr_{i \sim W_t} [h_t(x_i) \neq y_i]$
 the given solution is taken as the hypothesis h_t
 Get the weak hypothesis $h_t : X \mapsto \{-1, +1\}$ with error ϵ_t .
 Choose $\alpha_t = \frac{1}{2} \ln \left(\frac{1-\epsilon_t}{\epsilon_t} \right)$
 Update:

$$\begin{aligned} W_{t+1}(i) &= \frac{W_t(i)}{Z_t} \times \begin{cases} e^{-\alpha_t} & \text{if } h_t(x_i) = y_i \\ e^{\alpha_t} & \text{if } h_t(x_i) \neq y_i \end{cases} \\ &= \frac{W_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t} \end{aligned}$$

where Z_t is a normalization factor (chosen so that W_{t+1} will be a distribution).

end for
 Output the final hypothesis:

$$H(x) = \text{sign} \left(\sum_{t=1}^T \alpha_t h_t(x) \right)$$

B. Classification: Forest-ECOC

Once we located an object, we need to categorize among a large set of classes. Although various systems of multiple classifiers were proposed, most of them use similar constituent classifiers, which are often called base classifiers (dichotomies from now on). In this sense, Error Correcting Output Codes represent a classification technique that allows a successful combination of base classifiers to address the multi-class problem [31], [23].

1) *Error Correcting Output Codes*: The design of an Error Correcting Output Code is based on a coding and a decoding strategy, where coding aims in assigning a codeword¹ to each of the N_c classes (up to N_c codewords), and decode aims in assigning a class label to a new test codeword. Arranging the

¹A codeword is a sequence of bits that represents a class.

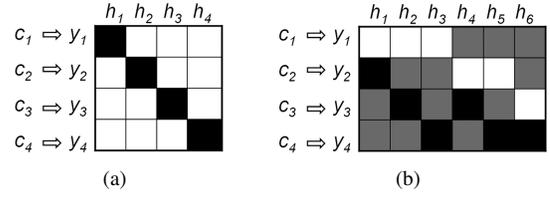


Fig. 3. Four-class ECOC designs. (a) One-versus-all ECOC codification and (b) one-versus-one ECOC codification (white: 1, black: -1, grey: 0).

codewords as rows of a matrix, we define the "coding matrix" M , where $M \in \{-1, 1\}^{N_c \times n}$, being n the code length. From the point of view of learning, the matrix M represents n binary learning problems (dichotomies), each corresponding to a column of the ECOC matrix M . Each dichotomy defines a sub-partition of classes, coded by $\{+1, -1\}$ according to their class membership. In Fig. 3(a) the codification for a four-class problem using the one-versus-all coding strategy is shown. The white and black regions correspond to $+1$ and -1 valued positions, respectively. Thus, in (a), the dichotomy h_i is trained to discriminate class c_i against the rest of classes. If we use a larger set of symbols for coding $M \in \{-1, 0, 1\}^{N_c \times n}$, some entries in the matrix M can be zero, indicating that a particular class is not considered for a given dichotomy. In Fig. 3(b), the codification for a four-class problem using one-versus-one coding strategy is shown. The grey regions correspond to the zero value (non-considered classes for the classifiers). In this strategy, all possible pairs of classes are split. For example, dichotomy h_1 classifies class c_1 versus class c_2 , etc.

As a result of the outputs of the n binary classifiers, at the decoding step a code is obtained for each data point in the test set. This code is compared to the base codewords of each class defined in the coding matrix M , and the data point is assigned to the class with the "closest" codeword. The common distances to decode are the Hamming and the Euclidean distances [32].

2) *Forest-ECOC*: Most of the discrete coding strategies up to now are pre-designed problem-independent codewords (one-versus-all [33], one-versus-one [34]). In the work of Pujol et al. [24], a method for embedding tree structures in the ECOC framework is proposed. Beginning on the root containing all classes, the nodes associated to the best partition in terms of the mutual information are found, and the process is repeated until the sets with a single class are obtained.

Taking the previous work as a baseline, we propose to use multiple trees embedding, forming a Forest-ECOC. We build an optimal tree - the one with the highest classification score at each node - and several suboptimal trees - the ones closer to the optimal one under certain conditions. Let us keep at each iteration the best k partitions of the set of classes. If the best partition is used to construct the current ECOC tree, the rest of partitions form the roots of $k - 1$ trees. We repeat iteratively this process until all nodes from the trees are decomposed into one class. Given a base classifier, the sub-optimal tree candidates are designed to have the maximum classification score at each node without repeating previous sub-partitions of classes. In the case of generating T first optimal trees, we

can create an ensemble of trees by embedding them in the ECOC matrix, as shown in Algorithm 2.

Algorithm 2 Training algorithm for the Forest-ECOC.

Given N_c classes: c_1, \dots, c_{N_c} and T trees to be embedded
 $\Omega_0 \leftarrow \emptyset$
 $i \leftarrow 1$
for $t = 1, \dots, T$ **do**
 Initialize the tree root with the set $N_i = \{c_1, \dots, c_{N_c}\}$
 Generate the best tree at iteration t :
for each node N_i **do**
 Train the best partition of its set of classes $\{P_1 P_2\} | N_i = P_1 \cup P_2, N_i \notin \Omega_{t-1}$ using a classifier h_i so that the training error is minimal
 According to the partition obtained at each node, codify each column of the matrix M as:

$$M(r, i) = \begin{cases} 0 & \text{if } c_r \notin N_i \\ +1 & \text{if } c_r \in P_1 \\ -1 & \text{if } c_r \in P_2 \end{cases}$$
 where r is the index of the corresponding class c_r .
 $\Omega_t \leftarrow \Omega_{t-1} \cup N_i$
 $i \leftarrow i + 1$
end for
end for

The proposed technique provides a sub-optimal solution because of the combination of robust classifiers obtained from a greedy search using the classification score. One of the main advantages of the proposed technique is that the trees share their information among classes in the ECOC matrix M . It is done at the decoding step by considering all the coded positions of a class jointly instead of separately. It is easy to see that each tree structure of N_c classes introduces $N_c - 1$ classifiers, that is far from the $\frac{N_c \cdot (N_c - 1)}{2}$ dichotomies required for the one-versus-one coding strategy.

An example of two optimal-trees and the Forest-ECOC matrix for a toy problem is shown in Fig. 4. The Fig. 4(a) and (b) show two examples of optimal trees. The second optimal tree is constructed based on the following optimal sub-partitions of classes. In this way, for the first initial set of classes $\{c_1, c_2, c_3, c_4\}$, the two optimal trees include the best sub-partitions of classes in terms of the classification score, that in the example corresponds to c_1, c_3 vs c_2, c_4 for the first tree, and c_1, c_2, c_3 vs c_4 for the second tree, respectively. Fig. 4(c) shows the embedding of trees into the Forest-ECOC matrix M . Note that the column h_3 corresponds to the node N_3 , and the following dichotomies correspond to the nodes of the second tree. The classes that do not belong to the sub-partitions of classes are set to zero. On the other hand, the classes belonging to each partition are set to $+1$ and -1 values, defining the subset of classes involved on each classifier.

Recent studies on the decoding steps have shown that the zero symbol introduces decoding errors in the traditional decoding distances [35]. To deal with this problem and to increase the performance of the Forest-ECOC coding design, we propose the Attenuated Euclidean decoding strategy, defined as $d_j = \sqrt{\sum_{i=1}^n |y_i^j| (x_i - y_i^j)^2}$, where d_j is the distance to row j , n is the number of dichotomies, x_i is the response of the classifier h_i over the test sample, and y_i^j is the value of the coding matrix M at i^{th} row and j^{th} column, respectively. We introduce the factor $|y_i^j|$ to avoid the error that the zero symbol introduces.

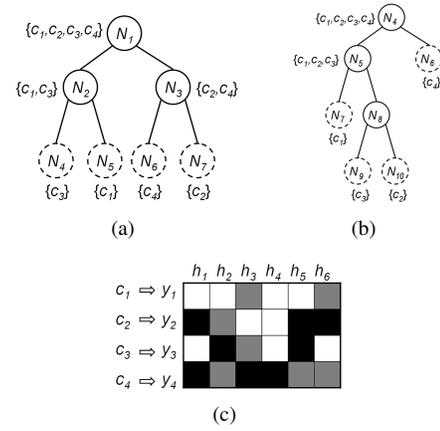


Fig. 4. Four-class optimal trees and the Forest-ECOC matrix. (a) First optimal tree for a four-class problem, (b) Second optimal tree for the same problem, and (c) Forest-ECOC matrix M for the problem, where h_1, h_2 and h_3 correspond to classifiers of N_1, N_2 and N_3 from the first tree, and h_4, h_5 and h_6 to N_4, N_5 and N_8 from the second tree.

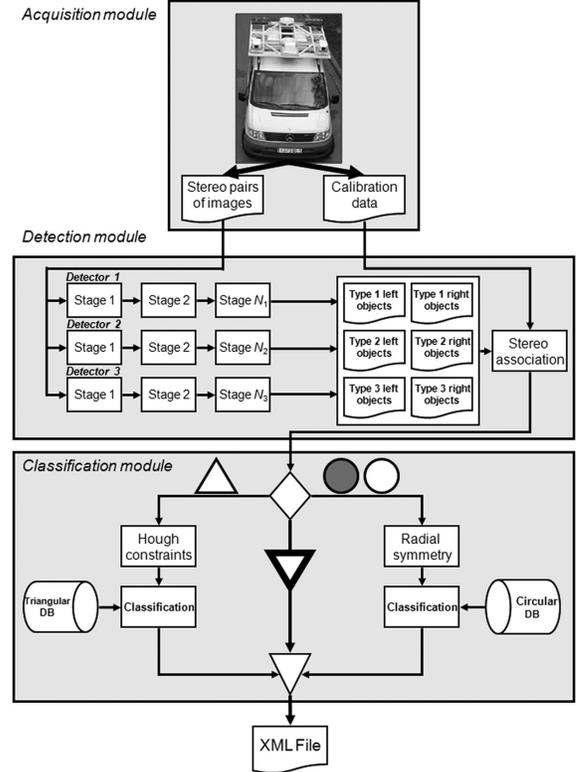


Fig. 5. Scheme of the whole traffic sign recognition system.

III. TRAFFIC SIGN RECOGNITION SYSTEM

This chapter presents the details of the system scheme shown in Fig. 5. We explain the relationship between each of the methods explained at the previous chapter and their integration in a real traffic sign recognition system.

A. Acquisition module

The mobile mapping system has a stereo pair of calibrated cameras, which are synchronized with a GPS/INS system. Therefore, the result of the acquisition step is a set of stereo-pairs of images with their position and orientation information.

This information allows the use of epipolar geometry in order to change from one camera to the other one and to obtain the real position in world coordinates of a point. In order to detect all signs that appear in the input image independently of their size and position, we scan the image using windows at different scales. All these windows are used as the detector input. In this sense, the detector can be defined as a classification function $h : X \mapsto \{-1, 1\}$, where X is a window from the input image. The result of the detection process is a set of windows that corresponds to the valid objects, that is, all the windows X where $h(X) = 1$.

B. Detection module

All the data generated in the acquisition process is given to the detector. We organize the trained detectors as an attentional cascade [21]. The *attentional cascade* is a degenerated decision tree where at each stage a detector is trained to detect almost all objects of interest while rejecting a certain fraction of the non-signs patterns. Because of the huge number of different traffic signs types, we group them using a similarity criterion, and we train a different cascade for each group. Each window in the input images is analyzed by all the cascades, and the detected objects from each cascade are given as output of the detector. Since each cascade detects only a certain category of signs, the output objects have a first classification information.

1) *Stereo association*: Since we work with a stereo system, all signs appear on both cameras at each frame (except in case of occlusions or if one of the signs is out of the field of view). This redundant information is used in order to improve the detection ratio. Using the epipolar geometry, given an instance of a sign in one of the sources, we estimate the region where it must appear in the other source. Once we have a reduced search window in the other source, we apply similarity criterion based on normalized correlation. The point with the highest similarity value gives us the position of the target object. This information is used to link the object of a source with its stereo object to recover it. Using this information, we only lose the objects that have been lost in both cameras. Using the calibration data, the position and orientation information of the GPS/INS system, and the coordinates of an object in each camera, we compute the object position in world coordinates.

C. Classification module

Using the Evolutive Adaboost, a region of interest (ROI) that contains a sign is determined. Depending on the type of the detected sign, a different model fitting is applied before classification, looking for affine transformations that perform the spatial normalization of the object.

1) *Model fitting*: Because of the few changes on the point of view of the captured signs, we apply the fast radial symmetry [36] for the circular signs, which offers high robustness against image noise. As it is shown in Fig. 6, the fast radial

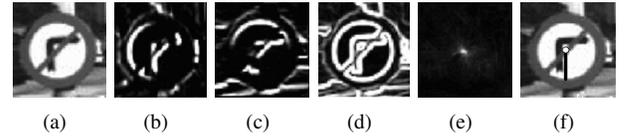


Fig. 6. (a) Input image, (b) X -derivative, (c) Y -derivative, (d) image gradient, (e) accumulator of orientations, (f) center and radius of the sign.

symmetry provides an approximation to the center and the radius of the circular sign.

On the other hand, for the case of triangular signs, the method that allows a successful model fitting is based on the Hough transform [37]. Nevertheless, we need to consider additional constraints to obtain the three representative border lines of a triangular traffic sign. Each line has associated a position in relation to the others. In Fig. 7(a) a false horizontal line is shown. Since this line does not fulfil the expected spatial constraints of the object, we iterate the Hough procedure to detect the next representative line in the allowed range of degrees. The corrected image is shown in Fig. 7(b). Once we have the three detected lines, we calculate their intersection, as shown in Fig. 7(c). To assure that the lines are the expected ones, we complement the procedure looking for a corner at the circular region of each intersection surroundings (as shown in Fig. 7(d) and (e)) $S = \{(x_i, y_i) \mid \exists p < ((x - x_i)^2 + (y - y_i)^2 - r^2)\} \mid i \in [1, \dots, 3]$, where S is the set of valid intersection points, and p corresponds to a corner point to be located in a neighborhood of the intersection point. Since at each lines intersection a corner should be determined, we apply a corner detector at surroundings of the triangular vertices to increase the confidence of determining a sign.

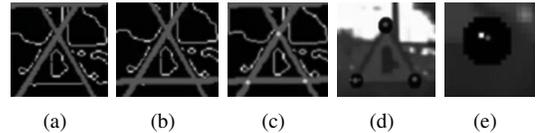


Fig. 7. (a) Detected lines, (b) corrected line, (c) intersections, (d) corner region, (e) corner found.

2) *Spatial normalization*: Once the sign model is fitted using the previous methods, the next step is the spatial normalization of the object before classification. The steps are: a) transform the image to make the recognition invariant to small affine deformations, b) resize the object to the signs database size, c) filter using the Weickert anisotropic filter [38], and d) mask the image to exclude the background pixels at the classification step. To prevent the effects of illumination changes, the histogram equalization improves image contrast and yields an uniform histogram.

3) *Forest-ECOC*: Once the signs are extracted, they are classified by means of the Forest-ECOC strategy. The optimal trees consider the best sub-partitions of classes to obtain robust classifiers based on the gray-level pixel-values of the object.

D. System outputs

At the end of the system, a XML file is generated, containing the position, size, and class of each of the detected

traffic signs. This information is used to obtain the real world position of each detected and recognized sign. An execution example of the whole process is shown in Fig. 8.

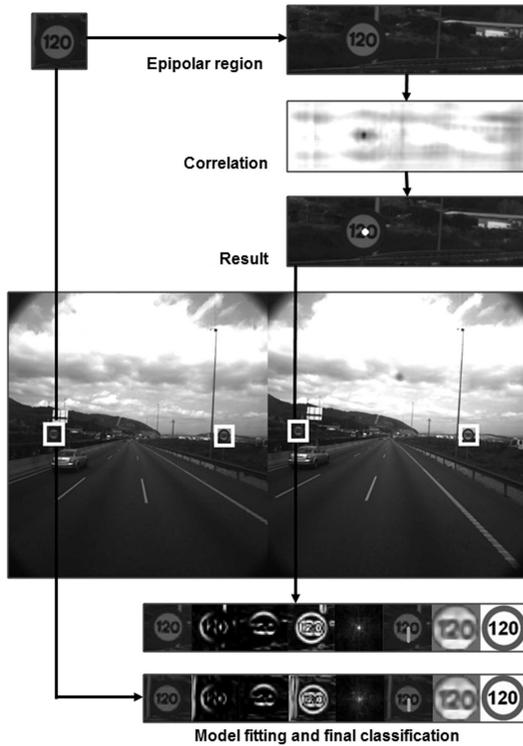


Fig. 8. Example of the system execution: image acquisition, detection, and classification.

IV. RESULTS

The different experiments of this section are focused on evaluating each individual part of the framework separately (detection and classification), and finally, performing the whole system evaluation. The validation of the methodology is carried out using real images from the mobile mapping system Geomobil [25]. This system captures georeferenced stereo-pairs of images, which are the input of the recognition system. For these experiments, the system was configured to acquire one frame each ten meters or when the orientation variation is greater than 60° . These parameters are hardly tested to assure a correct acquisition in road scenarios, which means that all the interesting objects appear at least in two or three stereo-pairs of images with a minimum size of 30×30 pixels resolution to be processed.

To assure a high diversity of road conditions, we selected three recording sessions, each one carried out different days and with different weather conditions. It represents a total of 9.510 stereo-pairs road images. To avoid using different images containing the same traffic sign in the training and test sets, instead of using random sequences, we divide the sessions in four subsequences of similar number of frames, without sharing signs. The first and third parts are used to train, and the second and fourth to test. The reason of using a larger test set is because there are a lot of frames that do not contain objects, and it is interesting to extract the

false alarm ratio in normal conditions, assuring to test the system under different illuminations conditions, road types, and traffic intensities.

A. Detection results

1) *Evolutionary strategy*: To perform all the tests with the evolutionary Adaboost approach, we use a Genetic Algorithm with a population size of 100 individuals, Gaussian based mutation probability (the Gaussian is centered at zero with a variance of the half of the variable range, decreasing the variance along the generations), and scattered cross-over strategy with a cross-over fraction of 0.8. When we use a genetic algorithm instead of an exhaustive search, different initializations of the algorithm with the same training data give rise to different weak classifiers. Since the dissociated dipoles can not be learned by the classical approach, in this experiment we use the Haar-like features. An one-stage detector is learnt using fixed training and test sets, comparing the error evolution for both strategies, and the variance in the evolutive approach over different runs.

We run the learning process over the same training and test sets 50 times, using 50 iterations of the evolutive Adaboost. In the case of the classic Adaboost, as the *Weak Learner* does an exhaustive search over the features, at each round the selected features are the same. In the case of the Genetic Algorithm, we calculate the mean error value over all the rounds for each iteration.

In Fig. 9 the train and test mean error values at each iteration are shown. Note that both methods converge with the same number of iterations. To analyze the error variability, in Fig. 10 we show the mean and standard deviation for the error at each iteration. The confidence interval shows that the variance is very closed. Therefore, though the evolutive Adaboost has a random component, the goodness of the given solution is similar.

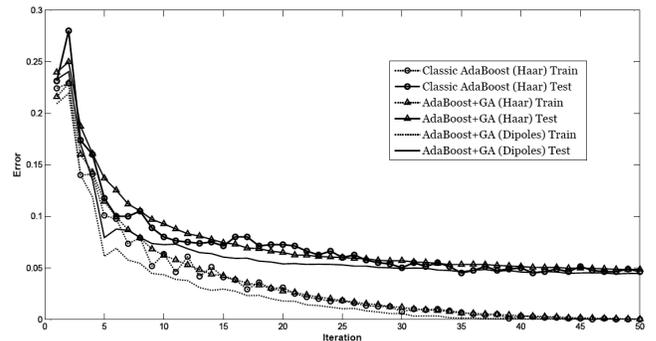


Fig. 9. Error evolution using the classic Adaboost approach and the genetic WeakLearner

2) *Detector performance*: To evaluate the detector performance, we train a cascade of detectors using the evolutive method with ordinal dissociated dipoles. In Fig. 11 we show the most relevant features selected by the evolutive method at the first stage of the cascade. Note that only few of them correspond to Haar-like features.

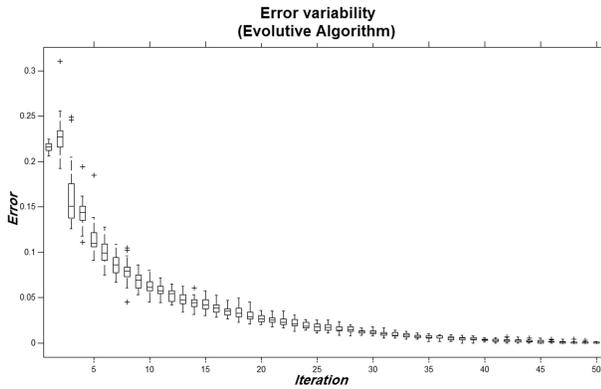


Fig. 10. Genetic approach. Error variability on the training process.



Fig. 11. Selected dipoles obtained over the danger signs.

Due to the different appearance frequency of each type of sign and the high intra-class variability, we trained a detection cascade for each group of similar signs. In table I we show the groups of signs and the number of positive samples used to train each cascade. The number of negative samples on the train process is automatically selected at each stage with a proportion of 3 : 1 (three negative examples for each positive example). Most part of the captured images are from main roads, and consequently, some types of signs do not appear enough times to train a detector. Due to this reason, we only trained the four detectors shown in table I.

Sign	Danger	Yield	Command	Prohibition
#Samples	545	425	256	993

TABLE I

NUMBER OF POSITIVE SAMPLES USED TO TRAIN THE CASCADE FOR EACH CONSIDERED SIGN.

The results are analyzed using two configurations. The first uses the stereo association to take advantage of the stereo information. The second considers each stereo-pair of images as two independent images. For each configuration, the obtained results with and without using sequential information are extracted. When the sequential information is used, different instances of the same real traffic sign are considered as the same object. In case of not using this information, each instance is considered as an independent object. In Fig. 12, we show the hit ratio of the detector trained for each type of sign. In general, we can see that the accuracy of the detectors

depends on the variability of sign appearance and the size of the training set. The First and the third columns correspond to the results considering each appearance of a traffic sign as a different sign. And the second and the fourth columns only take into account the real traffic signs, considering that a sign is detected if we can detect it in one or more frames where it appears. The first two columns do not take into account stereo redundancy, whereas the two last columns take it into account.

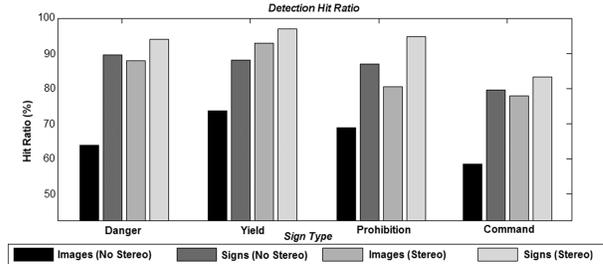


Fig. 12. Hit ratio for each sign type, using dissociated dipoles.

The other measure to evaluate the performance of the system is the false alarm rate. As we work with a mobile mapping system, an important point is which percentage of the detected objects corresponds to traffic signs. Therefore, our false alarm value is referred to the detected signs instead of the number of analyzed windows, which is of order of 5.000.000 per stereo-pair. Nevertheless, the number of false positives with respect to the number of stereo-pairs images has been included to make easier the analysis of the results. Both false alarm rates for each type of sign are detailed in table II. Some samples of detected objects and false alarms are shown in Fig. 13. One can see that the system is able to detect the signs in a very extreme lighting conditions. In the false positive images, one can see that frequently, other road elements look similar to traffic signs.

Sign	Danger	Yield	Command	Prohibition
FA/Sign	2.140	4.549	8.551	0.696
FA/Frame	0,045	0,056	0,073	0,019

TABLE II

FALSE ALARM RATES FOR EACH SIGN TYPE.



Fig. 13. Some samples of detected objects and false positives.

B. Classification results

Four types of experiments are performed to evaluate the classification scheme: traffic sign classification using different state-of-the-art classifiers, tree embedding analysis using Forest-ECOC, public UCI Machine Learning Repository classification, and model fitting classification. First of all, we comment the generation of the database to train the classifiers and to perform the experiments.

1) *Classification Database*: The database used to train the classifiers was designed using the regions of interest obtained from the detection step and the model fitting methods presented in the previous sections. We defined three groups of classes using the most common types of signs. The considered classes are shown in Fig. 14. Speed signs need special attention. These types of signs are less discriminative, being some of them only differentiated by a few pixels. With this type of signs it is better to work on binary images to avoid the errors that can be accumulated because of the grey levels of the signs. For the twelve classes of circular signs and twelve of triangular signs we have 750 training images in both cases. For the seven speed classes we use 500 training samples. Finally, the resolution of each database is: 35×35 pixels for the circular group, 44×39 pixels for the triangular group, and 41×41 pixels for the speed group, respectively.

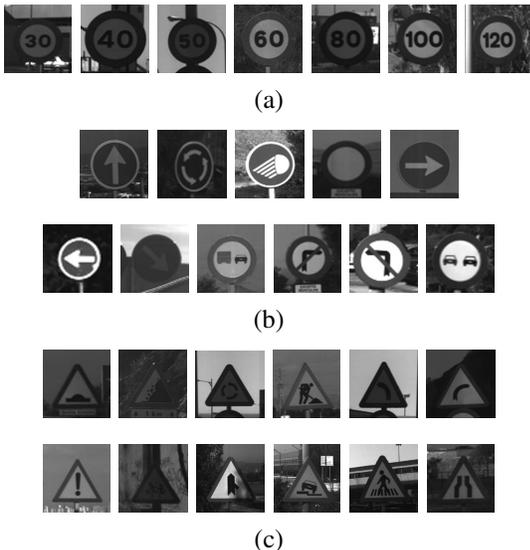


Fig. 14. Set of classes considered in the classification module. (a) Speed classes, (b) circular classes, and (c) triangular classes.

2) *State-of-the-art comparison*: To evaluate the Forest-ECOC performance, we compare it with the state-of-the-art classifiers. The details for each strategy are: 3-Euclidean distance Nearest neighbors (K-NN), Tangent Distance (TD) [39] with invariant tangent vector with respect to translation, rotation, and scaling, 99.98% of Principal Components Analysis followed by 3-Nearest neighbors (PCA K-NN) [40], Fisher Linear Discriminant Analysis with a previous 99.98% PCA (FLDA) [40], Support Vector Machine with projection kernel Radial Basis Function and the parameter $\gamma = 1$ (SVM) [41], Gentle Adaboost with decision stumps using the Haar-like features (BR) [22][42], multiclass Joint Boosting with decision

stumps (JB) [43], Gentle Adaboost [44] Sampling with FLDA (BS), statistical Gentle Naive Boosting with decision stumps (NB) [42], and our Forest-ECOC (F-ECOC) with 3-embedded optimal trees. In the different variants of boosting we apply 50 iterations. We use Gentle Adaboost since it shown to outperform the other Adaboost variants in real applications [44]. Concerning the Forest-ECOC base classifier, we apply FLDA in the experiments classifying traffic signs and 50 runs of Gentle Adaboost with Decision Stumps on the UCI data sets. This last choice was selected so that all strategies share the same base classifier.

Table III shows the characteristics of the data used for the classification experiments, where #Training, #Test, #Features, and #Classes correspond to the number of training and test samples, number of features, and number of classes, respectively.

Dataset	#Training examples	#Test examples	#Features	#Classes
Circular	750	200	1225	12
Speed	500	200	1681	7
Triangular	750	200	1716	12

TABLE III
CHARACTERISTICS OF THE DATABASES USED FOR CLASSIFICATION. THE TRAINING AND TEST EXAMPLES ARE EQUALLY DISTRIBUTED AMONG THE GROUP CLASSES.

The classification results and confidence intervals are shown graphically in Fig. 15 for the different groups. One can see that the Forest-ECOC using FLDA as a base classifier attains the highest accuracy in all cases. Nevertheless, for the circular and triangular signs the differences among classifiers are significantly different because of the high discriminability of these two groups. The speed group is a more difficult classification problem. In this case, the Forest-ECOC strategy obtains an accuracy upon 90%, outperforming the rest of classifiers.

3) *Tree embedding analysis*: The training evolution of the Forest-ECOC at the previous experiment is shown in Fig. 16 for the speed group. Each iteration of the figure shows the classification accuracy by embedding a new node (binary classifier) from each optimal tree in the Forest-ECOC matrix M . The three optimal trees are split by the dark vertical lines. The respective trees are shown in Fig. 17. In the first generated tree of Fig. 17, one can see that the most difficult partitions are reserved to the final classifiers of the tree. The next trees select the following best partitions of classifiers to avoid repeating classifiers. These classifiers learn sub-groups of classes from the same data, improving the classification results (Fig. 16) by sharing their knowledge among classes.

4) *UCI Evaluation*: In order to validate the accuracy of the Forest-ECOC strategy, we tested it on the public UCI Machine Learning Repository [45]. The characteristics of the UCI datasets considered are shown in table IV. In this case, to observe the benefits of using multiple trees embedding, we compared the performance of the Forest-ECOC strategy with the DECOC approach [24]. Both methods use 50 runs of

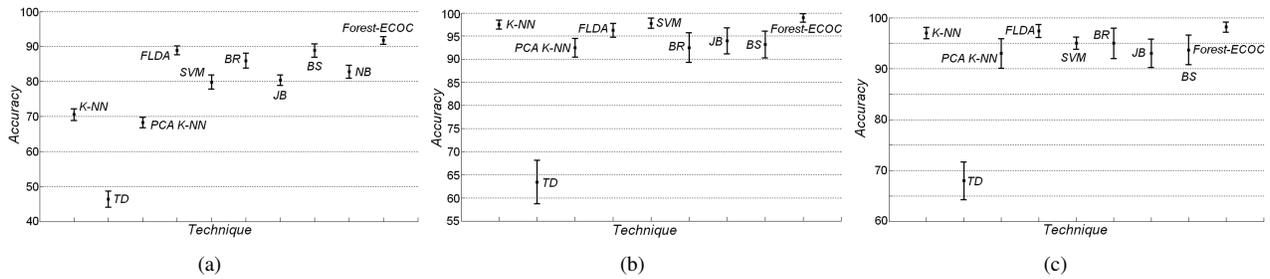


Fig. 15. Classification results for the (a) Speed, (b) Circular, and (c) Triangular problems.

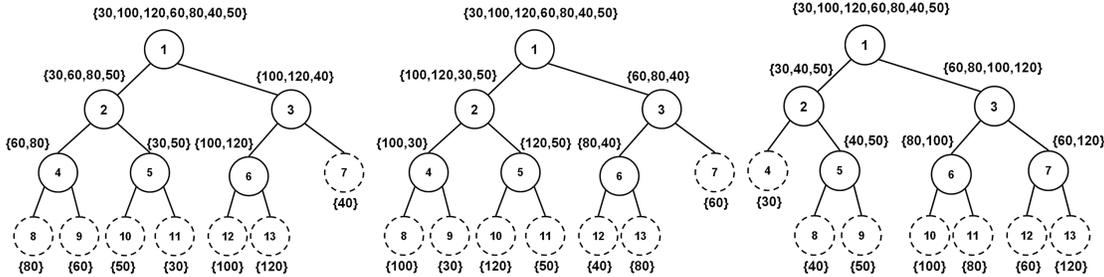


Fig. 17. Three optimal trees generated by the Forest-ECOC for the speed group.

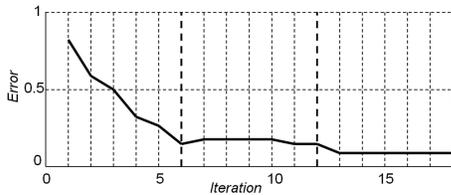


Fig. 16. Training process of Forest-ECOC embedding the first three optimal trees for the speed group.

UCI	Forest ECOC	DECOC
Yeast	53.85±1.64	51.15±2.14
Dermathology	95.32±1.31	93.10±1.81
Ecoli	83.98±1.13	78.15±1.42
Segmentation	94.98±0.66	92.30±0.89
Satimage	73.91±1.11	73.91±1.11
Vowel	77.67±1.81	73.84±1.90
Pendigits	81.42±1.93	81.42±1.93

TABLE V
CLASSIFICATION RESULTS ON THE UCI DATA SETS.

Gentle Adaboost with decision stumps as the base classifier, using 2-optimal trees for the Forest-ECOC strategy.

Problem	#Train	#Attributes	#Classes	Data types	Attribute types	Year
Dermathology	366	34	6	Multivariate	Categorical, Integer	1998
Ecoli	336	8	8	Multivariate	Real	1996
Vowel	990	10	11	Multivariate	Real	-
Yeast	1484	8	10	Multivariate	Real	1996
Satimage	6435	36	7	Multivariate	Integer	1993
Pendigits	10992	16	10	Multivariate	Integer	1998
Segmentation	2310	19	7	Multivariate	Real	1990

TABLE IV
UCI REPOSITORY DATA SETS CHARACTERISTICS.

The classification results of the two strategies over the UCI datasets are shown in table V. One can observe that the Forest-ECOC strategy outperforms in most cases the results obtained by the DECOC strategy for a same base classifier. Moreover, in the worst case, the Forest-ECOC methodology obtains the same results than the DECOC technique.

5) *Model fitting classification*: Finally, to test the performance of the classification step of the system, model fitting and Forest-ECOC classification are applied in a set of 200 regions of interests for each group. The regions of interest are obtained from the detection step. The results are shown in table VI. One can see that for circular and speed signs the results are practically maintained from the previous experiments. For triangular signs, the accuracy is slightly decreased

because of the effect of noise, variability of sign appearance, and resolution, that makes the Hough transform lose some sides of the triangular signs. Nevertheless, the final results are upon 90% in all cases.

Recognition problem	Accuracy
Circular	98.00
Speed	91.50
Triangular	92.50

TABLE VI
MODEL FITTING AND CLASSIFICATION RESULTS.

6) *System results*: We calculate the performance of the whole system over a test set of 10.000 stereo-pairs of images, which correspond to 100Km of road. The accuracy of the real traffic sign recognition system applying the detection and classification approaches jointly obtains a mean triangular sign reliability of $90.87 \pm 0.87\%$, and a circular sign reliability of $90.16 \pm 1.01\%$. In the detection stage, recognition fails are caused because of the background confusion (see Fig. 13) and the high inter-class variability, whereas in the classification stage the errors are produced because of the poor resolution of the images.

7) *Discussions*: Classical approaches to traffic sign detection are based on a segmentation of the image using

thresholding over a color space. It is useful by the fact that traffic signs use a color code to be easily identifiable by humans. These colors are used to be distinguished in natural scenes. However, some objects, such as vehicles, buildings, or advertising near the road, share similar colors. In addition, the acquired color information is related to the used camera, and it is far away from the real object color. Although in some works this problem is attenuated using some heuristics as the size and position of the segmented regions or shape, it can be a source of false positives that can interfere on the recognition of the signs.

Our work avoids these problems using appearance information instead of directly using the gray-scale information. The presented system uses relations between gray values inside the image, and thus, the changes produced by different acquisitions systems are smoothed². However, the appearance based methods improve the results at not free cost. The presented methods introduces an extra computationally cost to the detection stage, which disable them to be used in driver support applications, where the real-time requirement is harder than in the case of the mobile mapping problem. To face those restrictions, the detection stage must be optimized. There are promising works focused on cascade optimizations [46] and multiple detectors organization [47], which can be used to improve not only the results, but also the computational time. In addition, the original detection scheme of Viola & Jones [21] was successfully embedded in Hardware devices, allowing high detection frame rates. Similar strategies can be adopted for our method.

The performed experiments for the Forest-ECOC classification technique used a Greedy search to look for the optimal sub-groups of classes that form the tree structures. In our case, the exhaustive search was computationally feasible with the number of classes. If necessary, different strategies can be applied instead. In case of having a huge number of classes to learn, sub-optimal solutions can be found using faster approaches, such as the Sequential Forward Floating Search [48], in order to speed up the method.

In the same way, after a preliminary set of experiments, we fixed FLDA as the base classifier for the Forest-ECOC strategy in order to learn the binary problems. It shown to be a suitable choice for traffic sign classification. In other type of classification problems, the Forest-ECOC approach can be applied with other base classifier that better adapts to learn a particular distribution of the data.

The detection and recognition system has been implemented to automatically process hundreds of video sequences obtained from the Mobile Mapping Process of [25]. This data is processed by the Institut Cartogràfic de Catalunya to extract cartographic information. Before the system was implemented, the labeling of the traffic signs in image sequences was done manually by an user. With the automatic system, the recompilation of the information is about three times faster. Furthermore, comparing the results of labeling with the automatic and manual processes, we can argue that the automatic

performance obtains better results. It is done by the fact that the performance of the manual labeling is affected by the eyestrain of the user that processes thousand of frames and the extreme illumination conditions of the video sequences.

Concerning the computational cost of the system, the final cascade of detectors learnt with the evolutionary Adaboost is comparable in complexity to the Viola & Jones [21] face detector. Moreover, the same optimization technique to calculate the Haar-like features by means of the integral image can be used in the case of the dissociated dipoles. Using ordinal features, the comparison with a float value has been replaced with the use of the sign, which reduces the computational cost compared to the Viola & Jones real-time detector. Regarding the classification stage, whereas the training time could be expensive depending on the number of classes, training examples, and number of features per data sample, once the Forest-ECOC hypotheses are learnt, the classification time depends on the nature of the applied base classifier. In particular, considering FLDA or Adaboost, the classification decision only requires a simple matrix product or an additive model estimation, which can be computed in real-time. Nevertheless, though the detection and classification techniques can be optimized to be real-time approaches, the use of correlation on the stereo-association stage, the model fitting, and normalization methodologies prevent the whole system to be real-time.

Concerning to the memory usage, the final detection system needs eight integers per feature to represent the dissociated dipoles. During the detection process, at least the integral image of the processed image must be in memory. Finally, the classification parameters and the ECOC matrix must also be available in memory. Summarizing, the stronger restriction in memory usage is the size of processed images, not the system itself.”

V. CONCLUSIONS

We presented a mobile mapping detection and classification system to deal with the traffic sign recognition problem. We introduced a computationally feasible method for feature selection based on an evolutionary strategy. Since the exhaustive search over all the combinations between feature and threshold are replaced with the evolutionary approach and ordinal features, the final approach has two main advantages: it speeds up the learning process and allows to work with large feature sets to distinguish between object and background, which is computationally unfeasible using traditional methods because of the huge number of features. Moreover, we proposed the Forest-ECOC classification strategy to lead with the multi-class categorization problem. The method is based on the embedding of multiple optimal trees structures in the Error Correcting Output Codes framework, sharing their knowledge among classes, and constructing an ensemble of trees until the necessary performance is achieved. A wide set of traffic signs are recognized under high variance of appearance, such as noise, affine deformation, partial occlusions, or reduced visibility. The validation of the mobile mapping system shows that the presented methodology offers high robustness and better

²Note that thought color is not required in any stage of the present system, all the feature sets used on the detection stage can be extended to any color space.

performance than the state-of-the-art strategies in uncontrolled environments.

ACKNOWLEDGES

This work has been developed in collaboration with the *Institut Cartogràfic de Catalunya* under the supervision of Maria Pla. This work was supported in part by a research grant from projects TIN2006-15308-C02, FISPI061290, and CONSOLIDER-INGENIO 2010 (CSD2007-00018).

REFERENCES

- [1] U. Handmann, T. Kalinke, C. Tzomakas, M. Werner, and W. von Seelen, "An image processing system for driver assistance," in *IV'98, IEEE International Conf. on Intelligent Vehicles*. Stuttgart, Germany: IEEE, 1998, pp. 481–486.
- [2] U. Franke, D. Gavrilă, S. Gorzig, F. Lindner, F. Paetzold, and C. Wohler, "Autonomous driving goes downtown," *IEEE Intelligent Systems*, vol. 13, no. 6, pp. 40–48, 1998.
- [3] S. Madeira, L. Bastos, A. Sousa, J. Sobral, and L. Santos, "Automatic traffic signs inventory using a mobile mapping system," in *GIS PLANET 2005, International Conference and Exhibition on Geographic Information*, Estoril, Lisboa, Portugal, 2005.
- [4] N. Barnes and A. Zelinsky, "Real-time radial symmetry for speed sign detection," *Proc IEEE Intelligent Vehicles Symposium*, 2004.
- [5] H. Akatsuka and S. Imai, "Road signposts recognition system," in *SAE vehicle highway infrastructure: safety compatibility*, 1987, pp. 189–196.
- [6] D. Kellmeyer and H. Zwahlen, "Detection of highway warning signs in natural video images using color image processing and neural networks," in *Int. Conf. Neural Networks 1994*, vol. 7. IEEE, 1994, pp. 4226–4231.
- [7] M. de Saint Blancard, "Road sign recognition: A study of vision-based decision making for road environment recognition," in *Vision-based Vehicle Guidance*, ser. Springer Series in Perception Engineering. New York, Berlin, Heidelberg: Springer Verlag, 1992, pp. 162–172.
- [8] A. D. L. Escalera, L. E. Moreno, M. A. Salichs, and J. M. Armingol, "Road traffic sign detection and classification," in *IEEE: Transactions on Industrial Electronics*, vol. 44. IEEE, Dec 1997, pp. 848–859.
- [9] D. Ghica, S. W. Lu, and X. Yuan, "Recognition of traffic signs using a multilayer neural network," in *Proceedings of the 1994 Canadian Conference on Electrical and Computer Engineering*, vol. 44, Halifax, Canada, 1994, pp. 848–859.
- [10] A. de la Escalera, J. M. Armingol, and M. Salichs, "Recognition of traffic signs using a multilayer neural network," in *3rd International Conference on Field and Service Robotics*, Espoo, Finland, June 2001.
- [11] J. Miura, T. Kanda, and Y. Shirail, "An active vision system for real-time traffic sign recognition," in *IEEE: International Conference on Intelligent Transportation System 2000 (ITSC '00)*, 2000, pp. 52–57.
- [12] D. Shaposhnikov, L. Podladchikova, A. Golovan, N. Shevtsova, K. Hong, and X. Gao, "Road sign recognition by single positioning of space-variant sensor," 2002. [Online]. Available: citeseer.csail.mit.edu/657126.html
- [13] S. Hsu and C. Huang, "Road sign detection and recognition using matching pursuit method," in *Image and Vision Computing*, vol. 19, 2001, pp. 119–129.
- [14] G. Piccioli, E. D. Micheli, and M. Campani, "A robust method for road sign detection and recognition," in *ECCV (1)*, 1996, pp. 495–500. [Online]. Available: citeseer.csail.mit.edu/piccioli96robust.html
- [15] G. Piccioli, E. D. Micheli, P. Parodi, and M. Campani, "Robust road sign detection and recognition from image sequences," *Proc. Intelligent Vehicles*, pp. 278–283, 1994.
- [16] S. Escalera and P. Radeva, "Fast greyscale road sign model matching and recognition," *Recent Advances in Artificial Intelligence Research and Development*, IOS Press, Amsterdam, October 2004.
- [17] G. Loy and A. Zelinsky, "Fast radial symmetry for detecting points of interest," *IEEE Trans Pattern Analysis and Machine Intelligence*, no. 8, 2003.
- [18] S. Maldonado-Bascon, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gomez-Moreno, and F. Lopez-Ferreras, "Road-sign detection and recognition based on support vector machines," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 8, no. 2, pp. 264–278, June 2007.
- [19] A. de la Escalera, J. Armingol, J. Pastor, and F. Rodriguez, "Visual sign information extraction and identification by deformable models for intelligent vehicles," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 5, no. 2, pp. 57–68, June 2004.
- [20] P. Paclik, J. Novovicova, and R. Duin, "Building road-sign classifiers using a trainable similarity measure," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 7, no. 3, pp. 309–321, Sept. 2006.
- [21] P. Viola and M. Jones, "Robust real-time object detection," *International Journal of Computer Vision*, 2002.
- [22] R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," *Proc. of IEEE Conf. Image Processing*, pp. 155–162, 2002.
- [23] T. Dietterich and G. Bakiri, "Solving multiclass learning problems via error-correcting output codes," *Journal of Artificial Intelligence Research*, vol. 2, pp. 263–286, 2005.
- [24] O. Pujol, P. Radeva, and J. Vitrià, "Discriminant ECOC: A heuristic method for application dependent design of error correcting output codes," *Transactions on PAMI*, vol. 28, no. 6, pp. 1001–1007, 2006.
- [25] R. Alamús, A. Baron, E. Bosch, J. Casacuberta, J. Miranda, M. Pla, S. Sanchez, A. Serra, and J. Talaya, "On the accuracy and performance of the geomobil system," in *International Society for Photogrammetry and Remote Sensing (ISPRS '04)*, "Istanbul, Turkey", "July" 2004.
- [26] K. Thoresz and P. Sinha, "Qualitative representations for recognition," *Journal of Vision*, vol. 1, no. 3, pp. 298–298, 12 2001. [Online]. Available: <http://journalofvision.org/1/3/298/>
- [27] Balas and Sinha, "STICKS: Image-representation via non-local comparisons," *Journal of Vision*, vol. 3, no. 9, pp. 12–12, 10 2003.
- [28] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded Up Robust Features," in *Proceedings of the ninth European Conference on Computer Vision*, May 2006.
- [29] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *International Conference on Machine Learning*, 1996, pp. 148–156.
- [30] Y. Freund and R. Schapire, "A short introduction to boosting," in *J. Japan. Soc. for Artif. Intel.*, vol. 14, October 1999, pp. 771–780.
- [31] T. Dietterich and E. Kong, "Error-correcting output coding corrects bias and variance," *S. Prieditis and S. Russell*, 1995.
- [32] E. Allwein, R. Schapire, and Y. Singer, "Reducing multiclass to binary: A unifying approach for margin classifiers," *Journal of Machine Learning Research*, vol. 1, pp. 113–141, 2002.
- [33] N. J. Nilsson, *Learning Machines*. McGraw-Hill, 1965.
- [34] T. Hastie and R. Tibshirani, "Classification by pairwise grouping," in *proc. NIPS*, vol. 26, 1998, pp. 451–471.
- [35] S. Escalera, O. Pujol, and P. Radeva, "Decoding of ternary error correcting output codes," *CIARP*, 2006.
- [36] G. Loy and A. Zelinsky, "A fast radial symmetry transform for detecting points of interest," *TPAMI*, vol. 25, pp. 959–973, 2003.
- [37] B. S. Morse, "Segmentation (edge based, hough transform)," *technical report*, 2000.
- [38] J. Weickert, *Anisotropic diffusion in image processing*, ser. ECMI Series. Stuttgart: Teubner-Verlag, 1998.
- [39] P. Simard, Y. LeCun, J. Denker, and B. Victorri, "Transformation invariance in pattern recognition, tangent distance and tangent propagation," *Neural Networks: Tricks of the Trade*, vol. 1524, pp. 239–274, 1998.
- [40] S. Dudoit, J. Fridlyand, and T. Speed, "Comparison of discrimination methods for the classification of tumors using gene expression data," *Technical Report*, June 2000.
- [41] C. Hsu, C. Chang, and C. Lin, "A practical guide to support vector classification," *Department of CSIE, technical report*, 2002.
- [42] Y. Kim, S. Hahn, and B. Zhang, "Text filtering by boosting naive bayes classifiers," *SIGIR Conference on Research and Development*, 2000.
- [43] A. Torralba and K. Murphy, "Sharing visual features for multiclass and multiview object detection," in *TPAMI*, 2004.
- [44] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of boosting," *Technical Report*, 1998.
- [45] A. Asuncion and D. Newman, "UCI machine learning repository," 2007.
- [46] H. Luo, "Optimization design of cascaded classifiers," *Computer Vision and Pattern Recognition*, vol. 1, pp. 480–485, 2005.
- [47] C. Huang, H. Ai, Y. Li, and S. Lao, "Vector boosting for rotation invariant multi-view face detection," in *International Conference on Computer Vision*, 2005.
- [48] P. Pudil, F. Ferri, J. Novovicova, and J. Kittler, "Floating search methods for feature selection with nonmonotonic criterion functions," *Proc. Int. Conf. Pattern Recognition*, pp. 279–283, 1994.