

Blurred Shape Model for Binary and Grey-level Symbol Recognition

Sergio Escalera^{ac}, Alicia Fornés^{ab}, Oriol Pujol^{ac},
Petia Radeva^{ac}, Gemma Sánchez^{ab}, and Josep Lladós^{ab}

^a*Computer Vision Center, Universitat Autònoma de Barcelona, Edifici O,
Bellaterra 08193, Spain.*

^b*Dept. Computer Science, Universitat Autònoma de Barcelona, Edifici Q,
Bellaterra 08193, Spain.*

^c*Universitat de Barcelona, Dept. Matemàtica Aplicada i Anàlisi, Gran Via 585,
08007 Barcelona, Spain.*

{sescalera,afornes,oriol,petia,gemma,josep}@cvc.uab.es

Abstract

Many symbol recognition problems require the use of robust descriptors in order to obtain rich information of the data. However, the research of a good descriptor is still an open issue due to the high variability of symbols appearance. Rotation, partial occlusions, elastic deformations, intra-class and inter-class variations, or high variability among symbols due to different writing styles, are just a few problems. In this paper, we introduce a symbol *shape* description to deal with the changes in appearance that these types of symbols suffer. The *shape* of the symbol is aligned based on principal components to make the recognition invariant to rotation and reflection. Then, we present the Blurred Shape Model descriptor (BSM), where new features encode the probability of appearance of each pixel that outlines the symbols *shape*. Moreover, we include the new descriptor in a system to deal with multi-class symbol categorization problems. Adaboost is used to train the binary classifiers, learning the BSM features that better split symbol classes. Then, the binary problems are embedded in an Error-Correcting Output Codes framework (ECOC) to deal with the multi-class case. The methodology is evaluated on different synthetic and real data sets. State-of-the-art descriptors and classifiers are compared, showing the robustness and better performance of the present scheme to classify symbols with high variability of appearance.

Key words: Symbol Description, Symbol Recognition, Error Correcting Output Codes, Adaboost, Multi-class classification.

1 Introduction

Symbol recognition is a particular case of object recognition and one of the main topics of Graphics Recognition. Symbols are synthetic visual entities made by humans to be understood by humans. They can appear in scanned document images or in natural scenes captured by a camera. Typical applications are: analysis and recognition of logical circuit diagrams, engineering drawings, maps, architectural drawings, musical scores, and logo recognition [10]. Concerning natural scenes, the main applications are the recognition of logos with PDA cameras and smartphones, driving assistance (traffic signs) and blind people aid systems.



Fig. 1. Examples of symbols in scanned documents and natural scenes.

The most typical visual cues for recognizing symbols are texture, color, and shape, being the last one the most widely considered. Still, the definition of expressive and compact shape descriptors is required. A symbol recognition system consists of two main steps: the description and the classification step.

From the point of view of shape signature, the descriptor should ideally guarantee intra-class compactness and inter-class separability. It should be tolerant to noise, degradation, occlusions and distortion (including shear). And due to isolated symbols which are present in graphical documents, we must also take into account the variations in rotation, scaling and translation. The particular case of handwritten symbols deserves special attention. The main kinds of distortions in this case are: elastic deformation, inaccuracy on junctions or on the angle between strokes, line-arc ambiguity, errors like over-tracing, overlapping, gaps, or missing parts. Moreover, the system must cope with the variability produced by the different writer styles, with variations in sizes, intensities and the increase in the number of touching and broken symbols. On the contrary, in the camera-based symbol recognition domain, the system should cope with a totally different problematic: uncontrolled environments, illumination changes, and changes in the point of view (perspective).

According to [9], numerous shape descriptors, tolerant to such distortions, have been proposed. They can be classified in two strategies: continuous and structural approach (the reader is referred to [3][9] and [10] for surveys on shape recognition and symbol recognition, respectively). Continuous approach uses a feature vector derived from the image photometry to describe the shape. R-Signature [1], Angular Radial Transform (ART) [2] and Zernike moments [5] are some examples of Region-based continuous approaches. ART has good

performance for general shapes and uses few features by descriptor, whereas R-Signature is based on Radon transform, and includes a Fourier transform to reach invariance to rotation. Zernike moments are invariant to rotation and maintain properties of the shape, being invariant in presence of deformations. The curvature scale space (CSS) descriptor [6] uses the external contour (silhouette) for coding the shape, and is one of the MPEG7 standards [19,21]. It can only be used for closed curves, but it is tolerant to rotation. Shape context [7] is tolerant to deformations, and it does not impose restriction of being closed regions, as well as it is also used in handwritten symbols. As far as we know, directional descriptors (such as the SIFT descriptor [24]), which are widely applied for detecting objects in real scene images, have rarely been applied to symbol recognition so far. The SIFT descriptor is based on determining the significant orientations within a region taking into account their spatial arrangement. The second group corresponds to structural approaches, which tend to represent the shape (and the relations between parts of the shape) using structures like string, tree, graph or grammar, where the similarity measure is done by string, tree, graph matching or parsing respectively [4] [8]. These approaches capture the spatial arrangement of symbol parts, which usually may suffer from complex distortions.

Concerning the classifier, numerous techniques have been investigated based on statistical or structural approaches [10]. Elastic deformations of shapes modeled by probabilities tend to be learnt using statistical classifiers. The goal is to establish decision boundaries in the feature space which split patterns belonging to different classes. In the statistical decision theoretic approach, the decision boundaries are determined by the probability distributions of the patterns belonging to each class, which must be either specified or learnt [11,12]. One of the most well-known techniques in this domain is the Adaboost algorithm due to its ability for feature selection, detection, and classification [13]. On the other hand, though many classification algorithms are designed for multi-class problems, this extension is normally difficult. In such cases, the usual way to proceed is to reduce the complexity of the problem into a set of simpler binary classifiers and combine them. A usual way to combine these simple classifiers is the voting scheme (one-versus-one or one-versus-all strategies are the most frequently applied). In this field, Dietterich et. al. [14] proposed a framework inspired in the signal processing coding and decoding techniques in order to deal with multi-class categorization problems. The method is based on combining the binary classifiers as codified columns of a matrix and generating a codeword for each class. Thus, a test sample is evaluated with all the binary classifiers, and codewords are compared in the classification stage [14].

From the previous discussion, we can observe that the proposal of an universal descriptor to cope with the symbol taxonomy is still an open issue. In this paper, we introduce a novel symbol descriptor, the Blurred Shape Model. The

descriptor encodes the spatial probability of appearance of the *shape* pixels and their context information. As a result, a robust technique to deal with noise and elastic deformations is obtained. Moreover, we present a successful scheme using the Blurred Shape Model to deal with multi-class symbol recognition problems. The method aligns symbols *shape* by means of the Hotelling transform and an area density adjustment. Then, the BSM is used for obtaining the *shape* description. The Adaboost algorithm [13] is proposed to learn the descriptor features that best split each pair of classes, and the one-versus-one scheme of Error Correcting Output Codes [14] combines the Adaboost classifiers to deal with the multi-class case. With this classification strategy, the system can also cope with scalability variations. Moreover, a cross-validation procedure over different Blurred Shape Model parameters increases the system generalization capability. The present methodology is evaluated on synthetic, hand-drawn, and real data sets. Different state-of-the-art descriptors and classification strategies are compared, showing the robustness and better performance of the proposed scheme when classifying large number of symbols with high variability of appearance.

This paper is organized as follows: Section 2 introduces the Blurred Shape Model descriptor. Section 3 presents the multi-class symbol recognition system. Experimental results and discussions are presented in section 4. Finally, section 5 concludes the paper.

2 Blurred Shape Model

In order to describe a symbol that can suffer from irregular deformations, we propose to codify its *shape* in terms of a set of keypoints, which are defined by high gradient magnitude pixels. Taking into account these pixels, the Blurred Shape Model descriptor defines a set of spatial regions by means of a grid. Then, spatial relations among keypoints from neighbor regions are computed, and descriptor features are obtained.

Given a set of points forming shape $S = \{x_1, \dots, x_m\}$ of a particular symbol, each point $x_i \in S$, called from now *SP*, is treated as a feature to compute the BSM descriptor. The image region is divided into a grid of $n \times n$ equal-sized sub-regions (cells), r_i . Each cell receives votes from the *SPs* in it and also from the *SPs* in the neighboring sub-regions. Thus, each *SP* contributes to a density measure of its cell and its neighbors, and thus, the grid size identifies the blurring level allowed for the shape. This contribution is weighted according to the distance between the point and the centroid c_i of the region r_i . The algorithm is summarized in table 1.

In Fig. 2, a shape description is shown for an apple data sample. Fig. 2(a)

<p>Given an image I:</p> <ol style="list-style-type: none"> 1. Obtain the <i>shape</i> S contained in I. 2. Divide I in $n \times n$ equal size sub-regions $R = \{r_1, \dots, r_{n^2}\}$, with c_i the center of points for each region r_i, $i \in [1, \dots, n^2]$. 3. Let $N(r_i)$ be the neighbor regions of region r_i, defined as $N(r_i) = \{r_k r_k \in R, \ c_k - c_i\ < 2 g \}$, where g is the cell size. 4. Let r_i^x be the region which contains the point x. 5. Initialize the probability vector v as $v(i) = 0, \forall i \in [1, \dots, n^2]$. 6. <ul style="list-style-type: none"> For each point $\mathbf{x} \in S$, <li style="padding-left: 20px;">$D = 0$ <li style="padding-left: 20px;">For each $r_i \in N(r_i^x)$, <li style="padding-left: 40px;">$d_i = d(\mathbf{x}, r_i) = \ \mathbf{x} - c_i\ ^2$ <li style="padding-left: 40px;">$D = D + \frac{1}{d_i}$ <li style="padding-left: 20px;">End_For <li style="padding-left: 20px;">Update the probability vector v as $v(r_i) = v(r_i) + \frac{1}{d_i D}$ <li style="padding-left: 20px;">End_For 7. Normalize v as: $v(i) = \frac{v(i)}{\sum_{j=1}^{n^2} v(j)} \forall i \in [1, \dots, n^2]$

Table 1
Blurred Shape Model description algorithm.

shows the distances d_i of a SP to the nearest sub-regions centers. To give the same importance to each SP , all the distances to the neighbor centers are normalized. The output descriptor is a vector histogram v of length n^2 , where each position corresponds to the spatial distribution of SPs in the context of the sub-region and its neighbors. Fig. 2(b) shows the vector descriptor update once the distances of the first point in Fig. 2(a) are computed. Observe that the position of the descriptor corresponding to the affected sub-region r_{15} , the region with centroid is nearer to the analyzed SP , obtains the highest value.

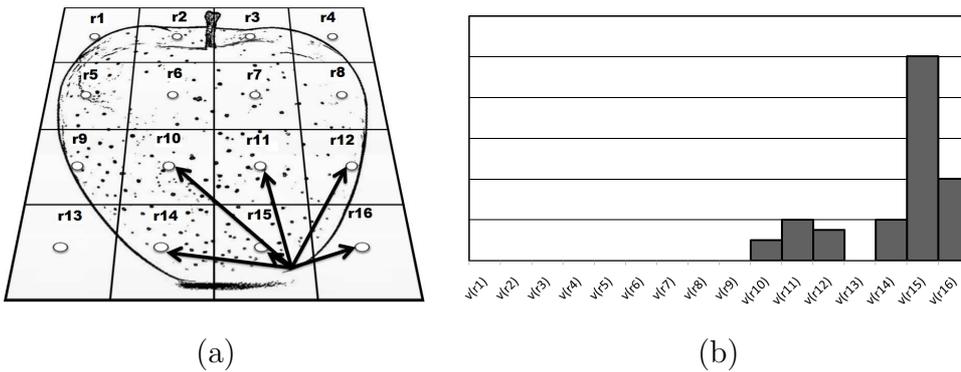


Fig. 2. BSM density estimation example. (a) Distances of a contour point SP to its neighbor centroids. (b) Vector descriptor update using the distances of (a).

The resulting vector histogram, obtained by processing all SPs , is normalized

in the range $[0, 1]$. In this way, the output descriptor represents a distribution of probabilities of the symbol structure considering spatial distortions, where the distortion level allowed is determined by the grid size. The BSM descriptors for different grid sizes applied to the previous example of Fig. 2 are shown in Fig. 3. Concerning the computational complexity, for a region of $n \times n$ pixels, the k SPs considered for obtaining the BSM descriptor require a cost of $O(k)$ simple operations. In Fig. 4(a) four BSM descriptors of apple samples of size 10×10 are shown. Fig. 4(b) shows the correlation of the four previous descriptors. Note that though variations on the shape of the symbols exists, the four descriptors remain closely superposed.

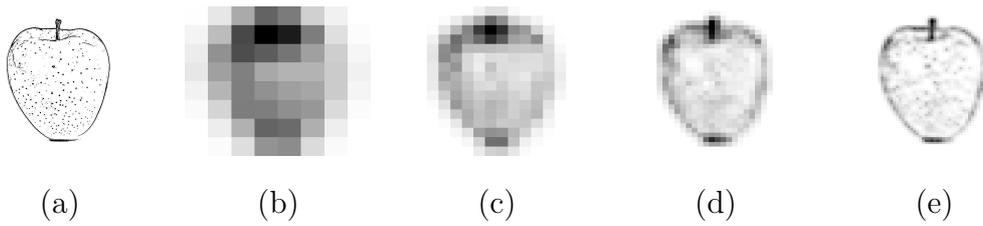
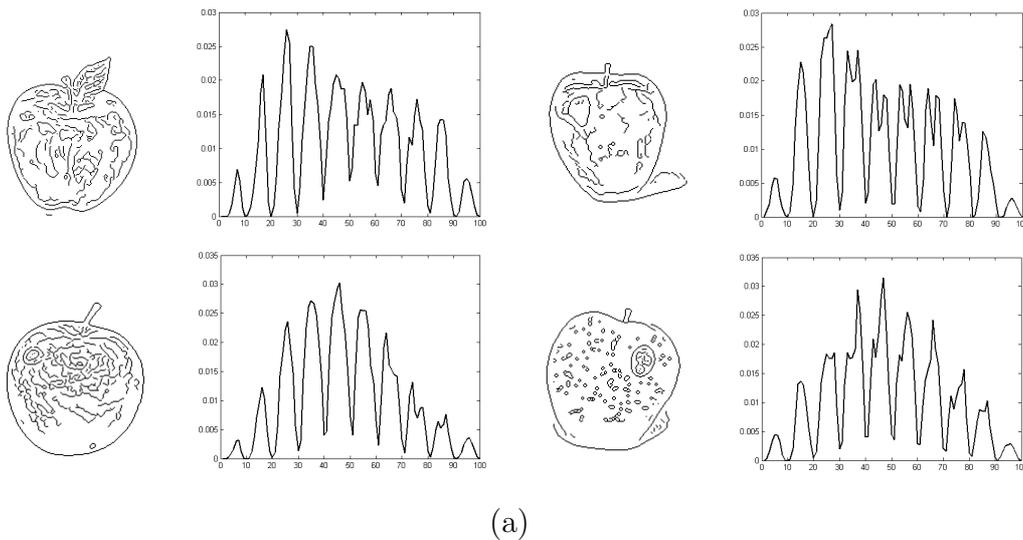
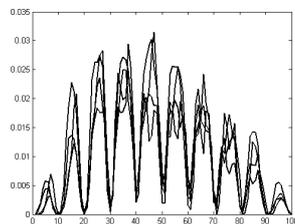


Fig. 3. (a) Input shape. BSM for (b) 8×8 , (c) 16×16 , (d) 32×32 , and (e) 64×64 grid sizes.



(a)



(b)

Fig. 4. (a) Plots of BSM descriptors of length 10×10 for four apple samples. (b) Superposition of previous BSM descriptors.

3 Multi-class recognition system using BSM

The process of symbol recognition usually involves a multi-class categorization problem. When different classes of symbols are described, a classification strategy is used to split among different categories. Based on the presented BSM features, Fig. 5 summarizes the training and testing steps of the proposed multi-class BSM system. For the training step, an input sample is processed to obtain its shape points in a binary map. This map is oriented to be rotational invariant, and then, it is described using the BSM descriptor. Adaboost is used to train an one-versus-one scheme of an Error-Correcting Output Codes (ECOC) design with Euclidean decoding. A cross-validation is applied considering different BSM sizes in order to look for the optimal grid size. The testing step of the system takes as input a region, and the structure processing, alignment, and BSM description optimizing the grid size is applied. Then, the ECOC scheme based on Gentle Adaboost is used to obtain a classification decision for the new test sample.

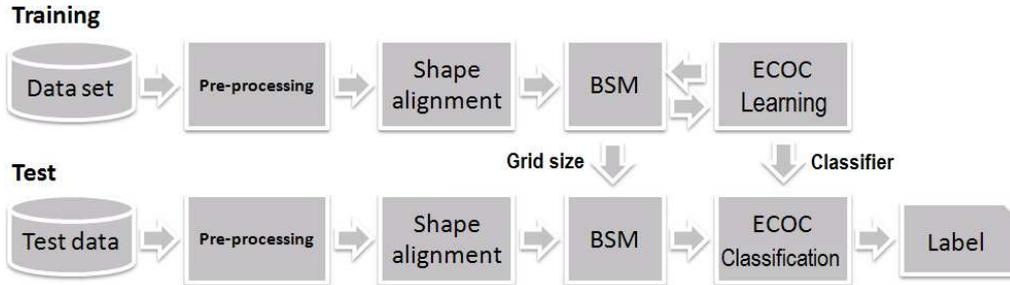


Fig. 5. Training and testing scheme of the multi-class symbol recognition scheme.

In the following sections, we explain in detail each of the steps of the system. Firstly, we describe the pre-processing step which is used for obtaining the structure map. Secondly, we describe the Hotelling transform based on principal components and the area density readjustment for aligning the symbols shape. Finally, we describe the ECOC scheme used to extend the binary classification to the multi-class case.

3.1 Pre-processing

To process the image and obtain the symbol structure, different pre-processing techniques can be applied depending on each particular problem domain. For instance, in the case of handwritten symbols, the skeleton is a good choice since it maintains the structure of the symbols for different author strokes. In the case of grey-level or binary symbols, a contour map is more suitable to

obtain the structure map¹.

3.2 Shape alignment

A shape alignment process is performed before computing the BSM descriptor. This process consists of two steps: the first step, provides invariance to rotation by means of the Hotelling transform. The second step deals with the mirroring effect.

The Hotelling transform finds a new coordinate system equivalent to locating the main axis of the symbol. Given a set of representative symbol points defined as pairs of coordinates $\mathbf{x} = (x_i, y_i)$, where $i \in [1, \dots, n]$, the center of mass of the symbol $m_{\mathbf{x}}$ computed as the average vector of points, and the eigenvectors V of the covariance matrix, the new transformation is obtained by means of the projection of the centered points of the symbol in the following way:

$$\mathbf{x}'_i = V(\mathbf{x}_i - m_{\mathbf{x}}), i \in [1, \dots, |S|] \quad (1)$$

Using this transform, we find the common axes for the different symbol instances. In Fig. 6(a), the mean shape for the samples of a MPEG7 category after applying the Hotelling transform is shown. One can observe that the shapes are not properly aligned. For this reason, a second step, consisting of an area density estimation process is used (see Fig. 6(b)). Horizontal and vertical projections are applied to obtain the area of the symbol. The final alignment is obtained by horizontal and vertical reflection of the symbol in the direction of the higher area projections. The result of adjusting the alignment is shown in Fig. 6(c). Another example of alignment of two MPEG7 symbol categories is shown in Fig. 7.

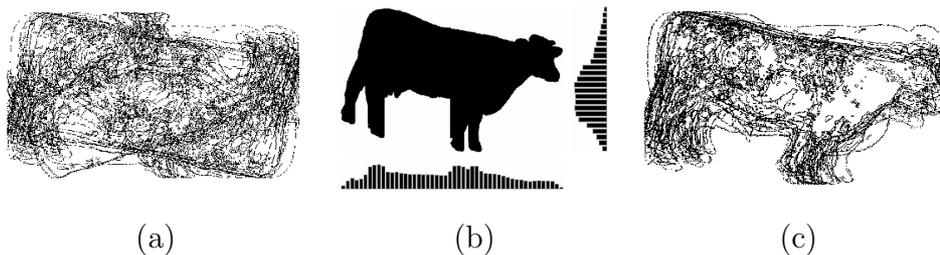


Fig. 6. (a) Mean aligned shape based on principal components. (b) Horizontal and vertical area estimation. (c) Readjusted alignment.

¹ Different alternatives for grey-scale symbols are presented in the experimental results section.



Fig. 7. Mean aligned shapes for two MPEG7 categories.

3.3 Error Correcting Output Codes

The ECOC framework is a simple but powerful framework to deal with the multi-class categorization problem based on the embedding of binary classifiers. Given a set of N_c classes, the basis of the ECOC framework consists of designing a codeword for each of the classes. These codewords encode the membership information of each binary problem for a given class. Arranging the codewords as rows of a matrix, we obtain a "coding matrix" M , where $M \in \{-1, 0, 1\}^{N_c \times n}$, with n being the length of the codewords codifying each class. From the point of view of learning, M is constructed by considering n binary problems, each one corresponding to a column of the matrix M . Each of these binary problems (or dichotomizers) splits the set of classes in two partitions (coded by +1 or -1 in M according to their class set membership, or 0 if the class is not considered by the current binary problem).

In Fig. 8(d), an example of a coding matrix M design is shown. The matrix is coded using 3 dichotomies $\{h_1, h_2, h_3\}$ for a 3-class problem (c_1, c_2 , and c_3). In Fig. 8(a)-(c), three different sub-partitions of classes are formed, corresponding to all possible pairs of classes, called one-versus-one strategy. Once we define the partitions of classes, each one is coded as a column of the coding matrix M , as shown in Fig. 8(d). The dark regions are coded as +1 (first partition of classes), and the white regions are coded as -1 (second partition of classes). The grey regions correspond to the non-considered classes for their respective classifiers. Now, the rows of the matrix M define the codewords $\{Y_1, Y_2, Y_3\}$ for their corresponding classes $\{c_1, c_2, c_3\}$.

At the decoding step, applying the n trained binary classifiers, 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 matrix M , and the data point is assigned to the class with the "closest" codeword.

As different types of symbols may share local features [15] (see Fig. 9), we make use of Adaboost [13] as base classifier applied in the previous ECOC scheme. Adaboost is used to learn the BSM models from different classes in order to define a classifier based on the features that best discriminate one class against another. Note that when we compare symbol descriptors, traditional matching distances take into account all symbol features for the final classification decision. When symbols are very similar, slight deformations

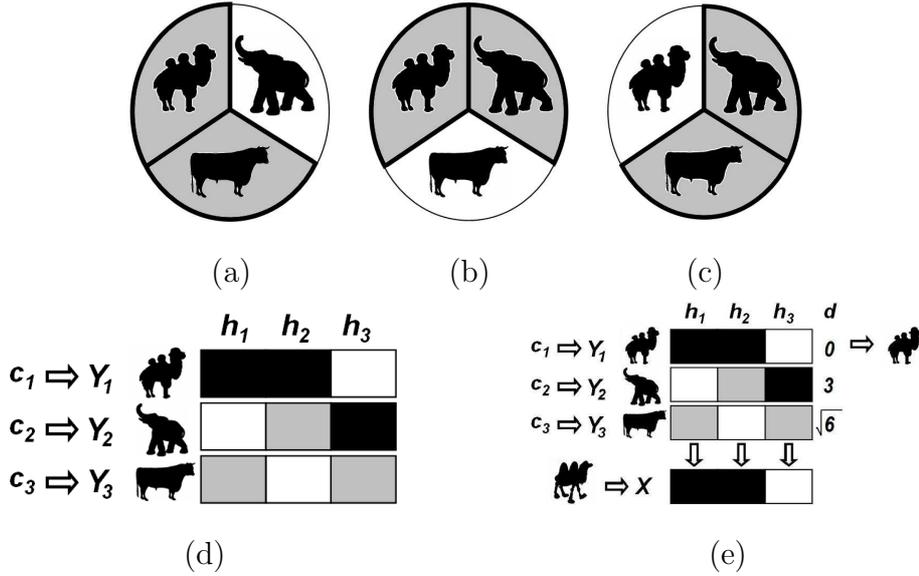


Fig. 8. (a)(b)(c) Three bi-partitions of classes for a three multi-class problem. (d) ECOC coding and (e) decoding for the problem.

in the shared parts may include significant distance errors that finally can lead to a miss-classification of the symbols. In Fig. 9 the two symbols have a discriminative region that splits both categories (marked with a rectangle). Adaboost focuses on these regions by selecting the highest splitting features. In particular, we use the Gentle version of Adaboost since it has been shown to be dominant to the rest of variants when applied to real categorization problems [13].

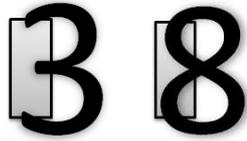


Fig. 9. Discriminant symbol features.

In Fig. 8(e), an input test sample classification is shown. This input is tested using the three binary classifiers, and a codeword X is obtained. Finally, the Euclidean distance is applied between each class codeword and the test codeword X in the form $d(X, Y_i) = \sqrt{\sum_{j=1}^n (X(j) - Y_i(j))^2}$, where $i \in [1, \dots, 3]$. Finally, the test input X is assigned to the class with minimum distance c_1 .

4 Experimental Results

In order to validate the proposed methodology, first, we describe our performance evaluation protocol in terms of the data used, metrics, comparatives, and experiments.

Data: To test the multi-class symbol recognition system, we used three different scenarios: a 7-class handwritten symbols data set², namely clefs and accidentals from old musical scores [18]. The main difficulty in this problem consists in the elastic deformations due to the different writing styles; the public 70-class MPEG7 repository data set [19]³, which contains a high number of classes with different appearance of symbols from the same class, including rotation. And finally, a 17-class data set of grey-level symbols², which contains the common distortions from real-environments, such as illumination changes, partial occlusions, or changes in the point of view. Some samples from each data set are shown in Fig. 10.

Metrics: To analyze the performance of the techniques, the descriptors are trained using 50 runs of Gentle Adaboost with decision stumps [13], and the one-versus-one ECOC design with the Euclidean distance decoding [16]. The classification score is computed by means of stratified ten-fold cross-validation, testing for the 95% of the confidence interval with a two-tailed t-test. Moreover, to look for statistical significance of the results, the performances are analyzed using the statistical Friedman and Nemenyi tests [22].

Comparatives: The methods used in the comparative are: SIFT [24], Zoning, Zernike, and CSS curvature descriptors from the standard MPEG [2,9,20]. The details of the descriptors used for the comparatives are the followings: the optimum grid size of the BSM descriptors is estimated applying cross-validation over the training set using a 10% of the samples to validate the different sizes of $n \in \{8, 12, 16, 20, 24, 28, 32\}$. For a fair comparison among descriptors, the Zoning descriptor is of the same length as BSM. Concerning the Zernike technique, 7 moments are used. The length of the curve for the CSS descriptor is normalized to 200, where the sigma parameter takes an initial value of 1 and increases by 1 unit at each step.

In order to compare the learning of the descriptors, different base classifiers are used in the ECOC scheme: OSU implementation of Linear Support Vector Machines with the regularization parameter C set to 1 [26], OSU implementation of Support Vector Machines with Radial Basis Function kernel with the default values of the regularization parameter C and the gamma parameter set to 1 [26]⁴, and Linear Discriminant Analysis implementation of the PR Tools using the default values [25].

² These data sets and ground truths are publicly available under request to the authors of this paper.

³ MPEG7 Repository Database: <http://www.cis.temple.edu/~latecki/research.html>

⁴ The regularization parameter C and the gamma parameter are set to 1 for all the experiments. We selected this parameter after a preliminary set of experiments. We decided to keep the parameter fixed for the sake of simplicity and easiness of replication of the experiments, though we are aware that this parameter might not be optimal for all data sets.

Experiments: To test and compare the performance of the different descriptors, we classify the set of clefs and accidentals data set classes using the different descriptors and base classifiers. The results are analyzed using statistical tests. Then, we classify the 70 MPEG7 classes. And finally, we deal with the 17-class problem of grey-level symbols from real environments. Moreover, we discuss implementation details to adapt the BSM descriptor and the classification scheme to different applications where the proposed system could be useful, such as symbol spotting from whole images. In the following subsections we further describe the experiments on the different benchmarking data sets.

4.1 Clefs and accidental data set

The data set of clefs and accidental is obtained from a collection of modern and old musical scores (19th century) of the Archive of the Seminar of Barcelona. The data set contains a total of 4098 samples among seven different types of clefs and accidental from 24 different authors. The images have been obtained from original image documents using a semi-supervised segmentation approach [18]. The main difficulty of this data set is the lack of a clear class separability because of the variation of writer styles and the absence of a standard notation. A pair of segmented samples for each of the seven classes showing the high variability of clefs and accidental appearance from different authors can be observed in Fig. 10(a). An example of an old musical score used to obtain the data samples are shown in Fig. 11.

The objective of this experiment is to classify the clefs and accidental data set comparing different state-of-the-art descriptors and classifiers. We compare the BSM, Zoning, SIFT, CSS, and Zernike descriptors using the parameters defined above. The optimum grid size used for the BSM and Zoning descriptors is of length 12×12 . Each feature set is learnt using the one-versus-one scheme with the previous commented base classifiers: FLDA, Linear SVM, RBF SVM, and Gentle Adaboost. Moreover, we include a comparative with a 3-Nearest Neighbor classifier to show the reliability of the present classification system. The performance and confidence interval obtained for each descriptor and classifier are shown in table 2. Looking at table 2, one can see that for each column corresponding to a different classifier, the descriptor that attains the best performance is the BSM. Moreover, looking at the performances of each row corresponding to the results of each base classifier applied over each feature set, one can see that the base classifier that obtains the best performance is the one-versus-one ECOC design with Gentle Adaboost as the base classifier. The only different case corresponds to the SIFT descriptor, which obtains its best performance with Linear SVM as base ECOC classifier. Finally, note that the results obtained by the 3NN classifier correspond to the lowest performance

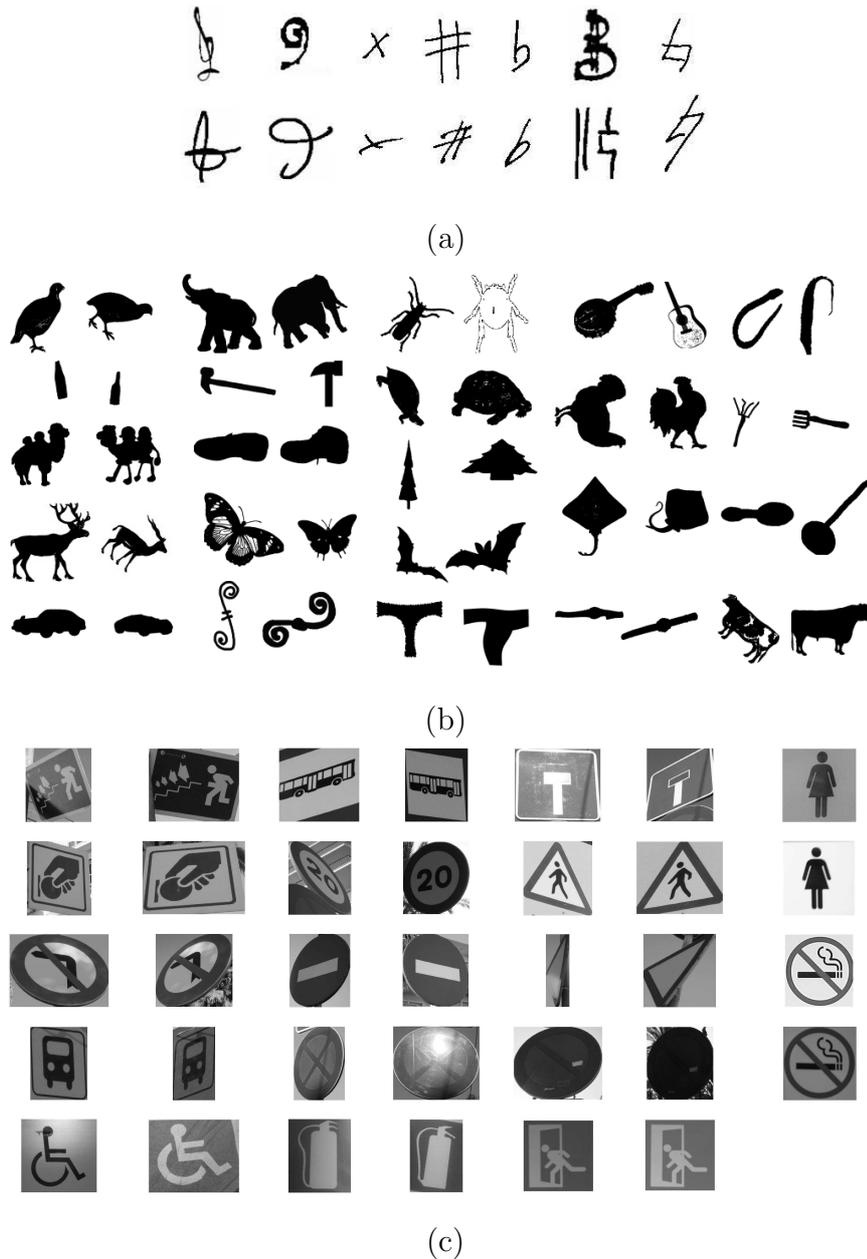


Fig. 10. Symbol data sets: (a) clefs and accidental data set, (b) MPEG7 data set, and (c) real symbols data set.

of each feature space.

Once the results are obtained, we analyze their statistical significance. For this purpose, we consider the performances obtained by the five different classifiers as five different experiments performed over the same data set, and we use the Friedman and Nemenyi tests [22] to look for statistical difference among the descriptors performances. Table 3 shows the mean rank of each descriptor considering the five different classifiers. The ranks are obtained estimating each particular rank r_i^j for each experiment i and each descriptor j , and then,

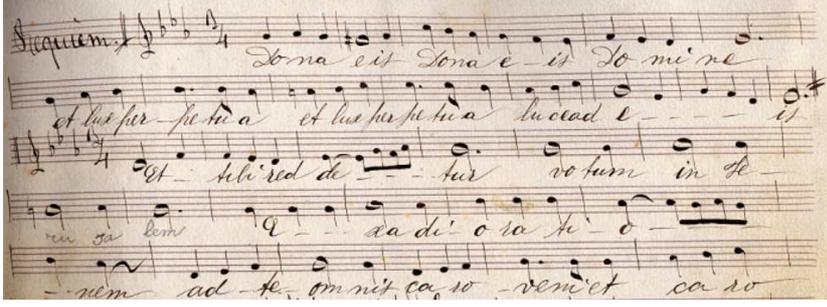


Fig. 11. Old musical score.

	FLDA	Linear SVM	RBF SVM	G. Adaboost	3NN
BSM	83.53(7.52)	80.51(7.31)	81.54(7.52)	88.99(5.00)	73.92(8.21)
Zoning	78.62(7.28)	79.45(6.30)	80.43(6.17)	83.61(5.24)	69.29(10.12)
SIFT	71.35(9.04)	76.45(6.73)	54.77(9.76)	74.95(9.77)	57.39(9.18)
CSS	68.76(11.02)	66.87(8.19)	69.87(9.18)	71.33(8.44)	61.28(8.92)
Zernike	69.09(6.01)	71.66(8.29)	59.21(9.00)	72.05(7.76)	54.12(9.10)

Table 2

Classification accuracy and confidence interval on the clefs and accidental categories for the different descriptors and classifiers.

BSM	Zoning	SIFT	CSS	Zernike
1	2	3.6	4.2	4.2

Table 3

Ranking of the different descriptors for the different experiments performed on the music dataset.

computing the mean rank R for each descriptor as $R_j = \frac{1}{P} \sum_i r_i^j$, where P is the number of experiments.⁵ One can see that the BSM descriptor, as commented, attains the best position for all experiments. In order to reject the null hypothesis that the measured ranks differ from the mean rank and that the ranks are affected by randomness in the results, first, we use the Friedman test. The Friedman statistic value is computed as follows:

$$X_F^2 = \frac{12P}{k(k+1)} \left[\sum_j R_j^2 - \frac{k(k+1)^2}{4} \right] \quad (2)$$

In our case, with $k = 5$ descriptors to compare, $X_F^2 = 16.48$. Since this value is undesirable conservative, Iman and Davenport proposed a corrected statistic [22]:

$$F_F = \frac{(P-1)X_F^2}{P(k-1) - X_F^2} \quad (3)$$

⁵ We realize that averaging over data sets has a very limited meaning as it entirely depends on the selected set of problems.

Applying this correction we obtain $F_F = 18.73$. With five methods and five experiments, F_F is distributed according to the F distribution with 4 and 16 degrees of freedom. The critical value of $F(4, 16)$ for 0.05 is 3.00. As the value of F_F is higher than 3.00, we can reject the null hypothesis. Once we have checked for the non-randomness of the results, we can perform a post hoc test to check if one of the techniques can be singled out. For this purpose we use the Nemenyi test - two techniques are significantly different if the corresponding average rankings differ by at least the critical difference value (CD):

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6P}} \quad (4)$$

where q_α is based on the Studentized range statistic divided by $\sqrt{2}$. In our case, when comparing five methods with a confidence value $\alpha = 0.05$, $q_{0.05} = 2.01$. Substituting in eq.(4), we obtain a critical difference value of 2.01. Since the difference of the SIFT, CSS, and Zernike ranks with the BSM rank is higher than the CD , we can infer that the BSM descriptor is significantly better than these descriptors with a confidence of 90% in the present experiment, being only comparable with the zoning descriptor, which also obtains inferior results.

4.2 MPEG7 data set

The MPEG7 data set has been chosen because it contains samples with high intra-class variability in terms of scale, rotation, rigid and elastic deformations, as well as a low inter-class variability. A pair of samples for some categories of the data set are shown in Fig. 10(b). Each of the classes contains 20 instances, which represents a total of 1400 symbol samples for the 70 classes.

The objective of this experiment is to classify a high number of candidates that suffer from rigid and elastic deformations. In this case, we compare different state-of-the-art descriptors: BSM, Zoning, SIFT, CSS, and Zernike with the previous defined parameters. In this experiment, the optimum grid size of the BSM descriptor is of length 16×16 . The scores obtained using cross-validation to look for the optimum grid size are shown in Fig. 12. In this case, the Hotelling alignment is applied to all the data set samples before computing the different descriptors. Then, each feature set is learnt using the one-versus-one scheme with Gentle Adaboost. Moreover, we include a comparison with a 3-Nearest Neighbor classifier to show reliability of the present classification system. The performance and confidence interval obtained by each descriptor and classifier are shown in table 4.

Looking at the results of table 4, one can see that for each descriptor, the

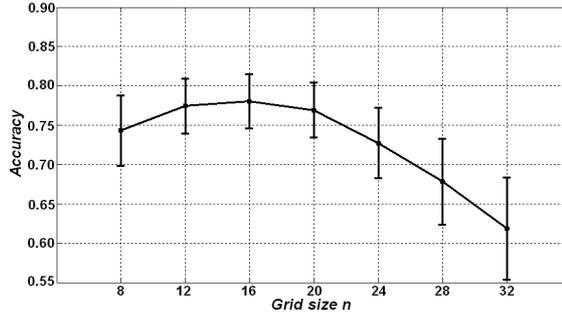


Fig. 12. Cross-validation on the training and validation sub sets for the MPEG7 data set using different BSM grid sizes.

Descriptor	3NN	Gentle Adaboost
BSM	65.79(8.03)	77.93(7.25)
Zernike	43.64(7.66)	51.29(5.48)
Zoning	58.64(10.97)	65.50(6.64)
CSS	37.01(10.76)	44.54(7.11)
SIFT	29.14(5.68)	32.57(4.04)

Table 4

Classification accuracy on the 70 MPEG7 symbol categories for the different descriptors using 3-Nearest Neighbor and the one-versus-one ECOC scheme with Gentle Adaboost as the base classifier.

one-versus-one scheme of ECOC with Gentle Adaboost as the base classifier obtains the best performance. Note that in this case, for the same classifier, the difference among descriptor performances is more significant. This is produced because of the high number of classes and the high variability of appearance of the symbol shape. The BSM descriptor obtains the best performance with an accuracy near 80%, followed by Zoning and Zernike, and finally by the CSS and SIFT descriptors. The performance of the two last descriptors was expected since they focus on the points of curvature from the symbols shape and the degrees of orientation from the image derivatives, which significantly change in this data set for the samples of the same class.

4.3 Camera-based grey-level symbols data set

The third data set of symbols is composed by grey-level samples from 17 different classes, with a total of 550 samples acquired with a digital camera from real environments. The samples are taken so that there are high affine transformations, partial occlusions, background influence, and high illumination changes. A pair of samples for each of the 17 classes are shown in Fig. 10(c). The SIFT descriptor is nowadays a widely-used strategy for this type of images, yielding good results [27]. For this reason, we compare the BSM and SIFT descriptors in this experiment.

BSM	SIFT
75.23(7.18)	62.12(9.08)

Table 5

Performance of the BSM and SIFT descriptors on the grey-scale symbols data set using an one-versus-one ECOC scheme with Gentle Adaboost as the base classifier.

To extend the use of the BSM descriptor from binary to grey-scale images, we estimate an adaptive orientation threshold for each particular problem. For a given image, our method computes the gradient module and normalizes it to unit. Then, the histogram of gradient magnitudes is estimated, and the Otsu method is applied in order to obtain an adaptive threshold for significant gradient modules. The points in the image with a higher gradient module than the computed threshold use to correspond to relevant symbol shape points. The optimum grid size used for the BSM descriptor is of length 8×8 . Some examples of the data set of this experiment and their corresponding BSM descriptors are shown in Fig. 13.

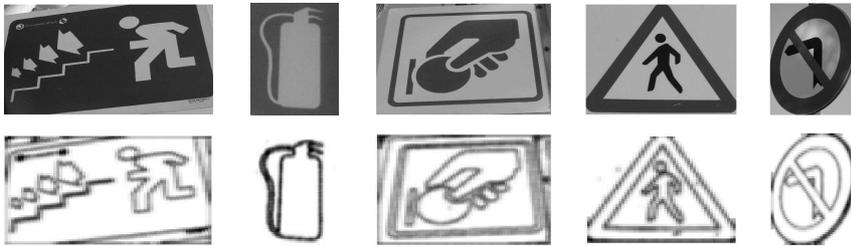


Fig. 13. BSM descriptors from samples of the grey-level symbols data set.

The performance and confidence interval obtained in this experiment from a ten-fold cross-validation using the BSM and SIFT descriptors in an one-versus-one ECOC scheme with Gentle Adaboost as the base classifier is shown in table 5. One can see that the results obtained by the BSM descriptor adapted to grey-scale symbols significantly outperform the results obtained by the SIFT descriptor. This difference is produced in this data set because of the large changes in the point of view of the symbols and the background influence, which produces significant distortions in the SIFT orientations.

4.4 Discussions

BSM are highly suitable to deal with multi-class symbol categorization problems due to the fact that our method is rotationally invariant because of the use of the Hottelling transform and the area density adjustment. The method is also scaling and stretching invariant taking into account the use of the $n \times n$ BSM grid. Moreover, the BSM descriptor is robust against symbols with rigid and elastic deformations since the size of the BSM grid defines the region of activity of the symbol shape points. The use of Adaboost as base classi-

fier allows to learn difficult classes which may share several symbol features. Besides, the ECOC framework has the property of correcting possible classification errors produced by the binary classifiers, and allows the system to deal with multi-class categorization problems. When the classifiers are trained, only few features are selected, and when classifying a new test sample, only these features are computed. This makes the approach very fast and suitable for real-time categorization problems.

An important point of the BSM description is the selection of the grid size. The optimum size defines the optimum grid encoding the blurring degree based on a particular data set distortions. For this reason, a common way to look for the optimum grid size is applying cross-validation over the data for different descriptor parameters, in particular, ten-fold cross-validation has been applied. The selected grid is the one which attains the highest performance on the validation subset, defining the optimum grid encoding the different distortions over each particular problem, and offering the required tradeoff between inter-class and intra-class variabilities in a problem-dependent way.

It is important to make clear that though Adaboost has been chosen as the base classifier in the presented system, depending on the problem we are working on, different alternatives of classifiers could be used instead, basing the selection of the base classifier on the type of distribution of the data and the behavior of each particular learning technique. Although at the previous experiments the comparison between Gentle Adaboost and other state-of-the-art classifiers showed higher performance improvements of Adaboost, different results could be obtained over different data sets or with an exhaustive tuning of the parameters of the classifiers. In the same way, the ECOC framework offers several advantages to deal with multi-class problems. Recent advances of the ECOC designs showed different alternatives for coding and decoding [16][17], which could be also applied in combination with the present methodology.

Moreover, it is important to mention different applications where the multi-class BSM scheme could also be useful. A possible application consists in symbol spotting. Because of the good shape encoding and fast computation of the BSM descriptor, it can be applied to this type of applications. It only requires to change the classification rule from symbols instances to the detection rule in whole images [23]. To show this behavior, we developed a simple experiment. We designed several images containing MPEG7 symbols, as the one shown in Fig. 14(a). The negative samples are those that do not contain the target symbol, and the positive test samples contain target instances (bats in this experiment). Random regions are selected from the negative set of images, and described using the BSM descriptor with a fixed grid size. The positive set is selected and described directly from the bat category of the MPEG7 data set. Then, the Gentle Adaboost classifier is trained with the positive and negative instances (500 negatives and 20 positives samples), and

a windowing procedure is applied over the test image. In Fig. 14(a) one can see that different image sub-windows are described using the BSM descriptor. The classifier learnt is applied over all sub-windows, and the response of the detector defines the regions containing a bat instance, as shown in Fig. 14(b). This process is speeded up by computing at the detection step only the features selected by Adaboost at the learning step, processing medium resolution image of 800×640 pixels in less than 1s.

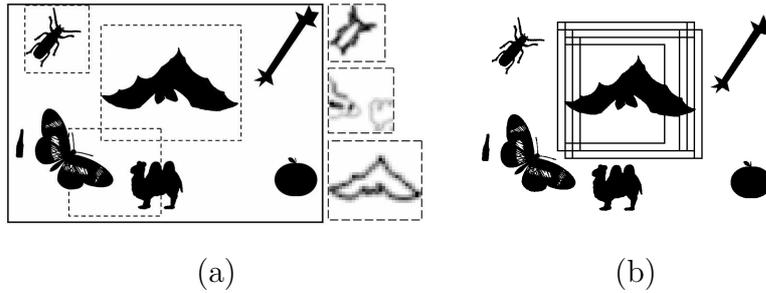


Fig. 14. (a) Some sub-windows BSM descriptions. (b) Selected sub-windows detecting bat symbols.

5 Conclusions

In this paper, we presented a multi-class symbol categorization system based on Blurred *Shape* Model descriptors. The Blurred Shape Model is a simple descriptor that in a fast way defines a probability density function of the *shape* of a symbol. The *shape* is described with a set of probabilities that encodes the spatial variability of the symbol, being robust to several symbol distortions. Besides, a system to improve the performance of the descriptor dealing with multi-class categorization problems has been proposed. Adaboost learns the discriminative features that better split symbol categories, and the binary classifiers are embedded in an ECOC framework. The present methodology has been evaluated on the public MPEG7 data set, in a handwritten data set of old musical scores, and on a real grey-level symbol data set. Different state-of-the-art descriptors and classifiers have been compared, showing the robustness and better performance of the proposed scheme when classifying symbols with high variability of appearance, such as irregular deformations induced by handwritten strokes, low inter-class and high inter-class variabilities, partial oclusions, illumination changes, or changes in the point of view. Moreover, we have shown that the novel BSM strategy could also be applied to symbol spotting problems.

Acknowledgements

This work has been partially supported by the projects TIN2006-15694-C02-02, TIN2006-15308-C02, and CONSOLIDER-INGENIO 2010 (CSD2007-00018).

References

- [1] S.Tabbone, L.Wendling, J.P.Salmon, "A new shape descriptor defined on the radon transform", *Computer Vision and Image Understanding* 102, vol. 1, pp. 42-51, 2006.
- [2] W.Y.Kim, Y.S.Kim, "A new region-based shape descriptor", *Technical Report*, Hanyang University and Konan Technology, December, 1999.
- [3] O.Ramos, S.Tabbone, E.Valveny, "A Review of Shape Descriptors for Document Analysis", *9th International Conference on Document Analysis and Recognition (ICDAR 2007)*, vol. 1, pp. 227-231, Curitiba (Brazil), September, 2007.
- [4] H. Bunke, "Attributed programmed graph grammars and their application to schematic diagram interpretation", *IEEE TPAMI*, Vol. 4, issue 6, pp. 574-582, 1982.
- [5] A.Khotanzad, Y.H. Hong, "Invariant Image Recognition by Zernike Moments", in *IEEE TPAMI*, vol. 12, issue 5, pp. 489-497, May 1990.
- [6] F. Mokhtarian, A. Mackworth, "Scale-based description and recognition of planar curves and two-dimensional shapes", in *IEEE TPAMI*, vol. 8, issue 1, pp. 34-43, 1986.
- [7] S. Belongie, J. Malik, and J. Puzicha, "Matching and Object Recognition Using Shape Contexts", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, issue 4, pp. 509-522, 2002.
- [8] J.Lladós, E. Martí, J.J. Villanueva, "Symbol Recognition by Error-Tolerant Subgraph Matching between Region Adjacency Graphs", *IEEE TPAMI*, vol. 23, issue 10, pp. 1137-1143, 2001.
- [9] D. Zhang and G. Lu, "Review of shape representation and description techniques", *Pattern Recognition*, vol. 37, pp. 1-19, 2004.
- [10] J. Lladós, E. Valveny, G. Sánchez, E. Martí, "Symbol Recognition: Current Advances and Perspectives", in *Graphics Recognition: Algorithms and Applications*, ed. D. Blostein and Y.B. Kwon, vol. 2390 of LNCS, Springer, Berlin, pp. 104-127, 2002.
- [11] L. Devroye, L. Györfi, and G. Lugosi, "A Probabilistic Theory of Pattern Recognition", Berlin: Springer-Verlag, 1996.

- [12] R.O. Duda and P.E. Hart, "Pattern Classification and Scene Analysis", New York: John Wiley & Sons, 1973.
- [13] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of boosting", *The Annals of Statistics*, vol. 8, issue 2, pp. 337-374, 1998.
- [14] T. Dietterich and G. Bakiri, "Solving multiclass learning problems via error-correcting output codes", *Artificial Intelligence Research*, vol. 2, pp. 263-286, 1995.
- [15] A. Torralba, K. Murphy, and W. Freeman, "Sharing visual features for multiclass and multiview object detection", *Technical Report, Massachusetts Institute of Technology Computer Science and Artificial Intelligence*, 2004.
- [16] S. Escalera, O. Pujol, P. Radeva, "An Incremental Node Embedding Technique for Error Correcting Output Codes", *Pattern Recognition*, vol. 14, issue 2, 2008, pp.713-725.
- [17] S. Escalera, D. Tax, O. Pujol, P. Radeva, and R. Duin, "Subclass Problem-dependent Design of Error-Correcting Output Codes", *IEEE TPAMI*, vol. 30, issue 6, pp. 1-14, 2008.
- [18] A. Fornés, J. Lladós, and G. Sánchez, "Primitive segmentation in old handwritten music scores", in *Graphics Recognition: Ten Years Review and Future Perspectives*, ed. W. Liu and J. Lladós, vol. 3926 of LNCS, Springer-Verlag, pp. 279-290, 2006.
- [19] MPEG7 Repository Database
[<http://www.cis.temple.edu/~latecki/research.html>].
- [20] Standard MPEG ISO/IEC 15938-5:2003(E).
- [21] B. Manjunath, P. Salembier, and T. Sikora, "Introduction to mpeg-7", *Multimedia content description interface*, John Wiley and Sons, 2002.
- [22] J. Demsar, "Statistical Comparisons of Classifiers over Multiple Data Sets", *Journal of Machine Learning Research*, vol. 7, pp. 1-30, 2006.
- [23] S. Escalera, O. Pujol, and P. Radeva, "Boosted Landmarks and Forest ECOC: A Novel Framework to Detect and Classify Objects in Clutter Scenes", *Pattern Recognition Letters*, vol. 28, issue 13, pp. 1759-1768, 2007
- [24] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", *Journal of Computer Vision*, vol. 60, issue 2, 91-110, 2004.
- [25] PRTools toolbox (Faculty of Applied Physics, Delft University of Technology, The Netherlands) [<http://www.prtools.org/>].
- [26] OSU-SVM-TOOLBOX [<http://svm.sourceforge.net>].
- [27] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A Comparison of Affine Region Detectors", *Journal of Computer Vision*, vol. 65, issue 1-2, pp. 43-72, 2005.