Hand Drawn Symbol Recognition by Blurred Shape Model descriptor and a Multiclass Classifier

Alicia Fornés[12], Sergio Escalera[23], Josep Lladós[12], Gemma Sánchez[12], and Joan Mas[12]

[1] Computer Vision Center, Universitat Autònoma de Barcelona,
08193, Bellaterra, Spain
E-mail: {afornes,josep,gemma,jmas}@cvc.uab.es
[2] Department of Computer Science, Universitat Autònoma de Barcelona,
Campus UAB, Edifici Q, 08193, Bellaterra, Spain
[3] Matemàtica Aplicada i Anàlisi, Universitat de Barcelona,
Gran Via 585, Barcelona, Spain
E-mail: sergio@maia.ub.es

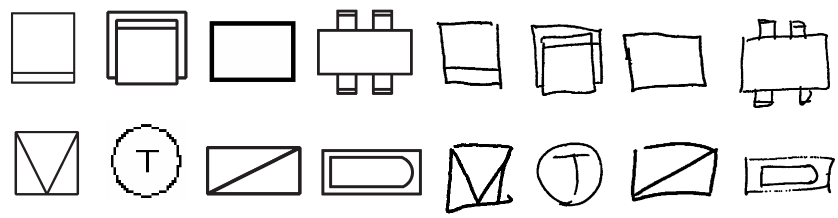
Abstract. In the document analysis field, the recognition of handwriting symbols is a difficult task because of the distortions due to hand drawings and the different writer styles. In this paper, we propose the Blurred Shape Model to describe handwritten symbols, and the use of Adaboost in an Error Correcting Codes framework to deal with multi-class categorization handwriting problems. It is a robust approach tolerant to the distortions and variability typically found in handwritten documents. This approach has been evaluated with the public GREC2005 database and an architectural symbol database extracted from a sketching interface, reaching high recognition rates compared with the state-of-the-art approaches.

1 Introduction

Symbols are a good way to express ideas. A number of graphical languages exist in different domains like engineering, architecture, software modelling, etc. These languages allow users to describe complex models with compact diagrammatic notations. On the other hand, in a technology world, freehand sketching is a very natural and powerful way of communication between humans. Sketch understanding is a research area with long history that brings together the above issues, i.e. a natural way of man-machine interaction in terms of freehand drawings and a pattern recognition ability to interpret sketches according to a diagrammatic notation. The first attempts of sketch recognition in the graphical domain can be found more than two decades ago. The early approaches were off-line systems mainly devoted to diagram beatification [1], or application cases of pattern recognition theory [2]. With the progress of digital pen and paper protocols such as Tablet PC or PDA, on-line sketch recognition systems gained in prominence. New alphabets have been developed for these devices, e.g. the

Graffitti alphabet for PDA's. The input of digital pen devices, called digital ink, consists in a sequence of points acquired at regular time intervals and grouped into basic sketch entities called strokes. A stroke is the set of points comprised between a pen down and a pen up movement. In an off-line input mode, strokes are stored in a binary image and each point has its coordinates as attributes. The advantage of on-line modes, in addition to the coordinates, is that each point may be attributed by dynamic information as the time order or the pressure. A number of applications exist that use sketches as input in areas like architecture [3], mechanics [4], logic diagrams [5], proofreading [6], retrieval [7] or iconic search in PDAs [8].

A sketch understanding system can be divided in three stages: primitive extraction, symbol recognition, and interpretation. In this paper we focus on symbol recognition. Recognizing a diagrammatic notation requires the identification of the alphabet symbols, that will subsequently be interpreted in the context of a domain-dependent graphic notation. Symbol recognition is one of the most active Graphics Recognition areas. A symbol recognition architecture requires two components, namely a shape signature able to robustly describe symbol instances, and a classification strategy. When we work with hand drawn inputs, due to the inherent distortions of strokes, the design of the descriptor is of key importance. The main kinds of distortions (see Fig.1) are: inaccuracy on junctions, on the angle between strokes, shape deformation, elastic deformation, ambiguity between line and arc, and errors like over-tracing, overlapping, gaps or missing parts. In addition, the system must cope with the variability produced by the different writer styles and different sizes.



(a) (b)

Fig. 1. a) Original shapes. b) Distorted shapes, from top to bottom and from left to right: 1) Distortion on junctions. 2) Distorion on angles 3) Overlapping. 4) Missing parts. 5) Distortion on junctions and angles. 6) and 7) Ambiguity arc-segment. 8) Gaps.

As stated above, a symbol recognition system firstly requires the definition of expressive and compact descriptors. It has to ideally guarantee intra-class compaction and inter-class separability, even when we describe noisy and distorted symbols. A number of well-known shape signatures exist (see a review in [9]) that can be used for describing symbols in Graphics Recognition. It was proved that some descriptors, robust with some affine transformations and occlusions in printed symbols, are not efficient enough for hand drawn symbols. Secondly, the formulation of robust classification methods according to such descriptors is required. Both, the descriptor and the recognition strategy must tolerate the inherent distortions involved in hand drawn inputs. A number of symbol recognition methods have been proposed to modelize such distortions. Examples are spectral models [10], arc-length curvature signatures [11], HMMs [12], deformable models [13], or graph transformation [14]. The reader is referred to [15] for a further review.

In this paper, we present an approach to model and classify handwritten symbols. Symbols are described using the Blurred Shape Model representation. The obtained features show to be high discriminative and tolerant to the transformations produced by the different writing styles. Moreover, we present a multi-class scheme, where Adaboost and Error-Correcting Output Codes are combined to deal with multi-class handwriting recognition problems. One of the most well-known techniques in the Machine Learning domain is the Adaboost algorithm due to its ability for feature selection, detection, and classification problems [17]. The design of a single multi-classifier is a difficult task, and it is common to conceive just binary classifiers and to combine them. One-versus-one voting scheme or one-versus-all strategies are the schemes most frequently applied. In this topic, Error Correcting Output Codes (ECOC) efficiently combines binary classifiers to address the multi-class problem [18]. The results over two multi-class databases show that the present methodology obtains significant performance improvements compared to the state-of-the-art approaches.

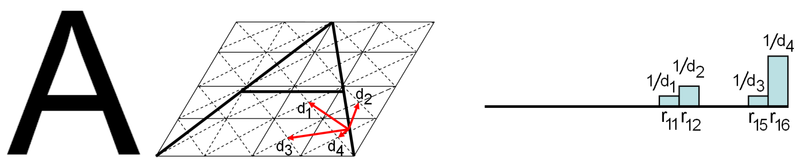
The steps of our approach are shown in Figure 2: First, the input hand drawn symbol is obtained as a binary image. Secondly, the Hotelling transform based on principal components [19] is applied to find the main axis of the object so the alignment can be performed. Third, the method defines a blurred region of the shape that makes the technique robust against elastic deformations. Afterwards, Adaboost is applied to each pair of classes to train relevant features that split better object classes. And finally, the set of binary classifiers is embedded in the framework of Error Correcting Output Codes (ECOC) to deal with multi-class categorization problems.

Fig. 2. Process scheme.

The paper is organized as follows: The proposed distortion tolerant descriptor, called Blurred Shape Model, is described in section 2. The classification stage is presented in section 3. Section 4 shows the experimental results over two multi-class databases. Finally, section 5 concludes the paper.

2 Blurred Shape Model Descriptor

The proposed descriptor describes the distribution of pixels among a predefined set of spatial regions. It is inspired in the well known zoning signature used in a number of OCR systems. Our method has also a foundation in the SIFT descriptor [16] that is one of the preferred strategies to describe image regions. The SIFT descriptor constructs a probability density function of the distribution of orientations within a region. However, in the handwriting recognition topic, orientations suffer from the variations produced by the different writing styles. Our approach consists in describing the symbol by a probability density function of Blurred Shape Model (BSM) that encodes the probability of pixel densities of image regions: The image is divided in a grid of $n \times n$ equal-sized subregions, and each bin receives votes from the shape points in it and also from the shape points in the neighboring bins. Thus, each shape point contributes to a density measure of its bin and its neighboring ones.



(a) (b) (c)

Fig. 3. (a) Input image (b) Shape pixel distances estimation respect to neighbor centroids. (c) Vector actualization of the region 16th, where $\frac{1}{d_1+d_2+d_3+d_4} = 1$.

In Fig. 3(a), an input symbol is shown. Figure 3(b) shows the distances estimation of a shape point respect to the nearest centroids. To give the same importance to each shape point, all the distances to the neighbors centroids $\{d_1, d_2, d_3, d_4\}$ are normalized to unit. The output descriptor is a vector histogram v of length $n \times n$, where each position corresponds to the amount of shape points in the context of the sub-region. The estimated normalized distances for each affected sub-region r are used to actualize their corresponding vector locations. Fig. 3 (c) shows the vector at this stage for the analyzed point of Fig. 3(b).

The resulting vector histogram, obtained by processing all feature points, is normalized in the range [0..1] to obtain the probability density function (pdf) of $n \times n$ bins. In this way, the output descriptor represents a distribution of probabilities of the object shape considering spatial distortions. In Fig. 4, an input shape is processed. Fig. 4b) to e) are the blurred parameterizations considering 48×48, 32×32, 16×16, and 8×8 sub-regions, respectively. In one hand, one can see that the less number of sub-regions, the less clear is the shape, and consequently, more tolerant to distortions. In the other hand, the more blurring effect, the more probability to having confusion between classes. Thus, it is important

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

Fig. 4. (a) Input image. (b) 48 regions blurred shape. (c) 32 regions blurred shape. (d) 16 regions blurred shape. (e) 8 regions blurred shape.

to find the suitable number of sub-regions in a problem-dependent way, reaching a balance between these two aspects.

Referring the computational complexity, for a region of $n \times n$ pixels, the $k \leq n \times n$ pixel points are considered to obtain the BSM with a cost of $O(k)$ simple operations. The whole algorithm is summarized in Figure 5.

Given a binary image I,
  Obtain the *shape* S contained in I
  Divide I in $n \times n$ equal size sub-regions $R = \{r_i, ..., r_{n \times n}\}$, with c_i the center of coordinates for each region r_i.
  Let $N(r_i)$ be the neighbor regions of region r_i, defined as $N(r_i) = \{r_k | r \in R, ||c_k - c_i||^2 \leq 2 \times g^2\}$, where g is the cell size.

  For each point $\mathbf{x} \in S$,
    For each $r_i \in N(r_{\mathbf{x}})$,
      $d_i = d(\mathbf{x}, r_i) = ||\mathbf{x} - c_i||^2$
    End_For
    Update the probabilities vector v positions as:
    $v(r_i) = v(r_i) + \frac{1/d_i}{D_i}, \quad D_i = \sum_{c_k \in N(r_i)} \frac{1}{||\mathbf{x} - c_k||^2}$
  End_For
  Normalize the vector v as: $v = \frac{v(i)}{\sum_{j=1}^{n^2} v(j)} \forall i \in [1, ..., n^2]$

Fig. 5. Blurred Shape Model algorithm.

3 Classification

Concerning the classification step, the Adaboost algorithm is proposed to learn the descriptor features that best split classes, training the classifier from Blurred Shape Model descriptors. The BSM has a probabilistic parametrization on the object shape considering its possible shape distortions. Due to the fact that different types of objects may share local features, Adaboost has been chosen to boost the BSM model in order to define a classifier based on the features that best discriminate one class against the others.

To extend the binary behavior of Adaboost to the multi-class case, we embed the binary classifiers in the framework of Error Correcting Output Codes [18]. The basis of the ECOC framework is to create a codeword for each of the L_c classes. Arranging the codewords as rows of a matrix, a "coding matrix" M is defined, where $M \in \{-1, 0, 1\}^{L_c \times z}$, being z the code length. From the point of view of learning, M is constructed by considering n binary problems, each corresponding to a matrix column. Joining classes in sets, each classifier defines a partition of classes (coded by +1, -1, according to their class set membership, or 0 if the class is not considered by the classifier).

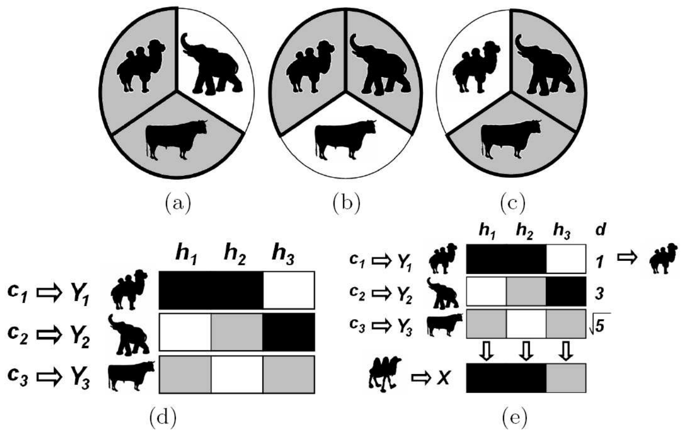


Fig. 6. One-versus-one ECOC design for a 3-class problem. a)b)c) Three bi-partitions of classes. d) ECOC coding. e) ECOC decoding.

In Fig. 6 an example of a matrix M is shown. The matrix is coded using 3 classifiers $\{h_1, ..., h_3\}$ for a 3-class problem. The white regions are coded by 1 (considered as positive for its respective dichotomy, h_i), the dark regions by -1 (considered as negative), and the grey regions correspond to the zero symbol (not considered classes for the current classifier). 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. In the right side of Fig. 6, an input test sample x is shown (a camel). This input shape is tested using the three classifiers, and assigning the outputs to each codeword position (down of the figure). Finally, the Hamming decoding distance is applied between each class codeword, classifying the test sample by the first class, because it has the minimum distance.

4 Results

Before the experimental results are presented, we discuss the data, methods, and evaluation measurements:

• *Data*: The presented approach has been evaluated using two databases. The first one is a architectural symbol database extracted from a sketching interface. The second one is the GREC2005 database, a public printed symbols database.

• *Methods*: We compare our methodology with the kernel density matching method (KDM) proposed in [22], ART, Zoning, and Zernike descriptors [9],[21]. To test our system, we use the Discrete Adaboost version [17] with 50 iterations of decision stumps, and the one-versus-one ECOC coding with Euclidean distance decoding [18],[19].

• *Evaluation measurements*: The performances are obtained by means of stratified ten-fold cross-validation with a two-tailed t-test at 95% of the confidence interval.

4.1 Architectural hand-drawn Categorization

The architectural symbol database is a benchmark database that has been created with the logitech io digital pen [24]. This database, which has been used in a sketch CAD framework [25], is composed of on-line and off-line instances from a set of 50 symbols drawn by a total of 21 users. Each user has drawn a total of 25 symbols and over 11 instances per symbol. The database consists on more than 5000 instances. To capture the data the following protocol has defined: The authors give to each user a set of 25 dot papers, which are paper containing the special pattern from anoto. Each paper is divided into 24 different spaces where the user has to draw in. The first space is filled with the ideal model of the symbol to guide the users on their draw due to they are not experts on the field of Architectural design.

Although the database is composed of 50 symbols, in our experiments we have chosen the 14 architectural symbols most representative from this database. Our experimental set consists in 2762 total samples organized in the 14 classes shown in Fig. 7. Each class consists of an average of 200 samples drawn by 13 different authors. In this experiment, the architectural symbol database has been used to test the performance of different descriptors for different number of classes.

The results obtained from BSM are compared with the ART, Zoning, and Zernike state-of-the-art descriptors [9][21]. The compared descriptors are also introduced in the classification framework to quantify the robustness of each descriptor at the same conditions. The parameters for ART are radial order with value 2 and angular order with value 11; and for the Zernike descriptor, 7 Zernike moments are used. The descriptors for BSM and Zoning techniques are of length 8×8, from the considered sub-regions. This optimum grid size has

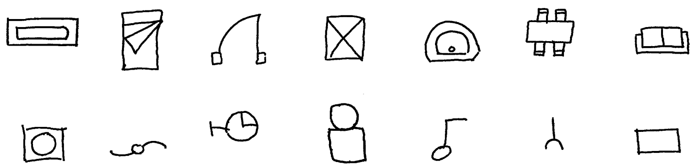


Fig. 7. Architectural handwriting classes.

been estimated applying cross-validation over the training set using a 10% of the samples to validate the different sizes of n, being 8×8 the size with the highest performance in the training set.

The classification starts using the first 3 classes. Iteratively, one class was added at each step and the classification is repeated. The higher number of classes, the higher confusion degree among them because of the elastic deformations inherent to hand drawn strokes, and the higher number of objects to distinguish. The results of accuracy recognition in terms of an increasing number of classes are shown in Fig. 8. The performance of the ART and Zernike descriptors decreases dramatically when we increase the confusion in terms of the number of classes, while Zoning obtains higher performance. Finally, the accuracy of the BSM outperforms the other descriptors results, and its confidence interval only intersects with Zoning in few cases. This behavior is quite important since the accuracy of the latter descriptors remains stable, and BSM can distinguish the 14 classes with an accuracy upon 90%.

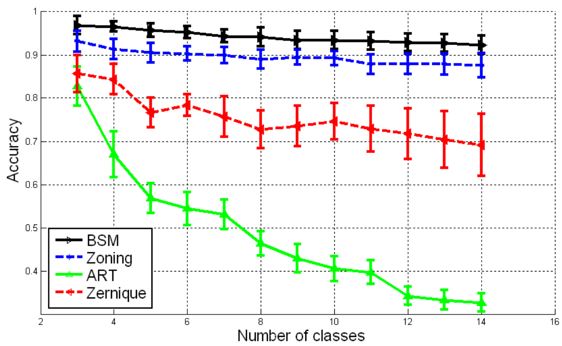


Fig. 8. Descriptors classification accuracy increasing the number of architectural symbol classes.

4.2 GREC05 Categorization

The GREC2005 database [23] is not a hand drawn symbol database, but it has been chosen in order to evaluate the performance of our method on a standard, public and big database. It must be said that our initial tests are applied on the first level of distortions (see Fig. 9). We generated 140 artificial images per model (thus, for each of the 25 classes) applying different distortions such as morphological operations, noise addition, and partial occlusions. In this way, the ECOC Adaboost is able to learn a high space of transformations for each class. The BSM descriptor uses a grid of 30×30 bins. In this sense, 900 features are extracted from every image, from which Adaboost selects a maximum of 50. For this experiment, we compare our results with the reported [22] using the kernel density matching method (KDM). The results are shown in Table 1. One can see that the performances obtained with our methodology are very promising, outperforming for some levels of distortions the KDM results.

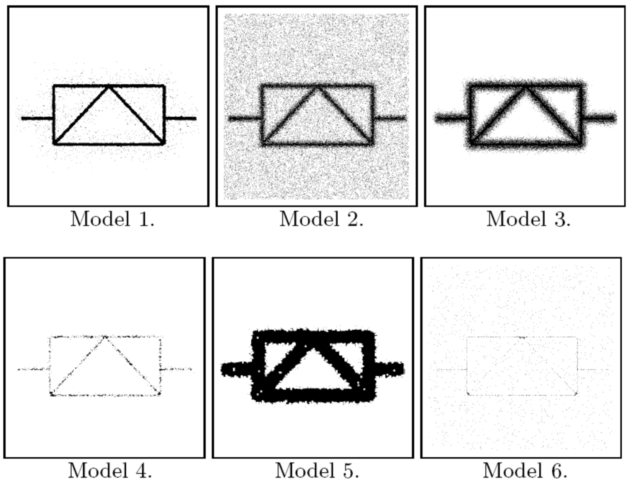


Fig. 9. An example of the distortion levels used in the GREC2005 database.

5 Conclusions and future work

In this paper, we have proposed the Blurred Shape Model descriptor and the use of Adaboost in the Error-Correcting Output Codes framework to deal with multi-class handwriting recognition problems. This methodology was evaluated on two multi-class databases, showing promising results in comparison to the

Method	Distortion Level 1	Distortion Level 2	Distortion Level 3	Distortion Level 4	Distortion Level 5	Distortion Level 6
KDM	100	100	100	96	88	76
BSM	100	100	100	100	96	92

Table 1. Descriptors classification accuracy increasing the distortion level of GREC2005 database using 25 models and 50 test images.

state-of-the-art approaches, being robust against noise, scale, and the elastic deformations produced by the different writing styles. Moreover, the complexity of the present methodology shows to be very suitable for real-time multi-class classification problems.

Nowadays, we are extending the experiments on different printed databases, such as the MPEG7 or GREC07, increasing the set of distortions to evaluate the robustness of the present descriptor. Moreover, we are also testing several handwritten symbol databases to show the suitability of the present multi-classification scheme.

Acknowledgements

This work has been partially supported by the spanish projects TIN2006-15694-C02-02, TIN2006-15308-C02-01 and CONSOLIDER-INGENIO 2010 (CSD2007-00018).

References

1. Pavlidis, T., Van Wyk, C.J.: An automatic beautifier for drawings and illustrations. J. SIGGRAPH Comput. Graph. 19, n.3, 225–234 (1985).
2. Bunke, H.: Attributed programmed graph grammars and their application to schematic diagram interpretation, IEEE Trans. on PAMI 4, 574–582 (1982).
3. Leclercq, P.: Absent Sketch Interface in Architectual Engineering. In: Lladós, J., Kwon, Y.B. (eds.) Graphics Recognition: Recent Advances and Perspectives. LNCS, vol. 3088, pp. 351–362, Springer-Verlag (2004).
4. Davis, R.: Understanding in Design: Overview of Work at the MIT Lab. In: AAAI Spring Symposium Sketch Understanding, pp.24–31. AIII Press, Stanford, California (2002).
5. Gennari, L., Burak Kara, L., Stahovich, T.F., Shimada, K.: Combining geometry and domain knowledge to interpret hand-drawn diagrams. In: Computers & Graphics, vol. 29, n.44, pp.547–562. Elsevier (2005).
6. Guimbretière, F.: Paper augmented digital documents. In: 16th annual ACM symposium on User interface software and technology, pp.51–60. ACM Press, Vancouver, Canada Edition (2003).
7. Rigoll, G., Muller, S.: Graphics-Based Retrieval of Color Image Databases Using Hand-Drawn Query Sketches. In: Chhabra, A.K., Dori, D. (eds.) Graphics Recognition: Recent Advances. LNCS, vol. 1941, pp. 256–265, Springer-Verlag, Berlin (2000).

8. Liu, W., Xiangyu, J., Zhengxing, S.: Sketch-Based User Interface for Inputting Graphic Objects on Small Screen Device. In: Blostein, D., Kwon, Y.B. (eds.) Graphics Recognition: Algorithms and Applications. LNCS, vol. 2390, pp.67–80, Springer, Berlin (2002).
9. Zhang, D., Lu, G.: Review of shape representation and description techniques. Pattern Recognition 37, 1–19 (2004).
10. Liang, S., Sun, Z., Li, B.: Sketch Retrieval Based on Spatial Relations. In: International Conference on Computer Graphics, Imaging and Visualization, pp.24–29. Beijing, China (2005).
11. Wilfong, G., Sinden, F., Ruedisueli, L.: On-line recognition of handwritten symbols. IEEE Transactions on Pattern Analysis and Machine Intelligence 18, n.9, 935–940 (1996).
12. Muller, S., Rigoll, G.: Engineering drawing database retrieval using statistical pattern spotting techniques, In: Chhabra, A.K., Dori, D. (eds.) Graphics Recognition: Recent Advances. LNCS, vol. 1941, pp. 246–255, Springer-Verlag, Berlin (2000).
13. Valveny, E., Martí,E.: Hand-drawn symbol recognition in graphic documents using deformable template matching and a bayesian framework, In: 15th. Int. Conf. on Pattern Recognition, vol. 2, pp.239–242. Barcelona, Spain (2000).
14. Lladós, J., Martí, E., Villanueva, J.J.: Symbol Recognition by Error-Tolerant Subgraph Matching between Region Adjacency Graphs. IEEE Transactions on Pattern Analysis and Machine Intelligence 23, n.10, pp.1137–1143 (2001).
15. Lladós, J., Valveny, E., Sánchez, G., Martí, E.: Symbol Recognition: Current Advances and Perspectives. In: Blostein, D., Kwon, Y.B. (eds.) Graphics Recognition: Algorithms and Applications. LNCS, vol. 2390, pp.104–127, Springer, Berlin (2002).
16. Lowe, D. G.: "Distinctive Image Features from Scale-Invariant Keypoints", International Journal of Computer Vision 60, n.2, 91–110 (2004).
17. Friedman, J., Hastie, T., Tibshirani, R.: Additive logistic regression: a statistical view of boosting, The Annals of Statistics 8, n.2, 337–374 (1998).
18. Dietterich T., Bakiri, G.: Solving multiclass learning problems via error-correcting output codes, J. Artificial Intelligence Research 2, 263–286 (1995).
19. Dunteman, G. H.: Principal components analysis (Quantitative Applications in the Social Sciences), Sage Publications (1989).
20. Escalera, S., Fornés, A., Pujol, O., Lladós, J., Radeva, P.: Multi-class Binary Object Categorization using Blurred Shape Models, In: Rueda, L., Mery, D., Kittler, J. (eds) Progress in Pattern Recognition, Image Analysis and Applications, 12th Iberoamericann Congress on Pattern Recognition. LNCS, vol. 4756, pp. 773–782, Springer (2008).
21. Kim, W., "A new region-based shape descriptor". Technical report, Hanyang University and Konan Technology (1999).
22. Zhang W., Wenyin, L., Zhang, K.: Symbol recognition with kernel density matching, Trans. on PAMI 28, n.12, 2020–2024 (2006).
23. GREC2005 database, http://symbcontestgrec05.loria.fr/formatgd.php
24. Logitech,I(O) digital pen, http://www.logitech.com (2004).
25. Sáchez, G., Valveny, E., Lladós, J., Mas, J., Lozano, N.: A platform to extract knowledge from graphic documents. Application to an architectural sketch understanding scenario, In: Marinai, S., Dengel, A. (eds.), Document Analysis Systems VI. LNCS, vol. 3163, pp.389–400, Springer-Verlag (2004).