CIRCULAR BLURRED SHAPE MODEL FOR SYMBOL SPOTTING IN DOCUMENTS

Sergio Escalera[1,2], *Alicia Fornés*[1,3], *Oriol Pujol*[1,2], *Alberto Escudero*[2], *and Petia Radeva*[1,2]

[1]Computer Vision Center, Campus UAB, edifici O, 08193, Bellaterra, Spain
[2]Dept. Matemàtica Aplicada i Anàlisi, UB, Gran Via 585, 08007, Barcelona, Spain
[3]Dept. Computer Science, Campus UAB, Edifici Q, 08193, Bellaterra, Spain

ABSTRACT

Symbol spotting problem requires feature extraction strategies able to generalize from training samples and to localize the target object while discarding most part of the image. In the case of document analysis, symbol spotting techniques have to deal with a high variability of symbols' appearance. In this paper, we propose the Circular Blurred Shape Model descriptor. Feature extraction is performed capturing the spatial arrangement of significant object characteristics in a correlogram structure. *Shape* information from objects is shared among correlogram regions, being tolerant to the irregular deformations. Descriptors are learnt using a cascade of classifiers and Abadoost as the base classifier. Finally, symbol spotting is performed by means of a windowing strategy using the learnt cascade over plan and old musical score documents. Spotting and multi-class categorization results show better performance comparing with the state-of-the-art descriptors.

Index Terms— Image Shape Analysis, Document Image Processing.

1. INTRODUCTION

Document Image Analysis and Recognition (DIAR) is one of the most active areas in the field of pattern recognition, including the recognition of text documents, page layout analysis and graphics recognition. Symbol recognition is one of the main topics of Graphics Recognition, which has been intensivly researched in the last decades. Symbols are synthetic visual entities that can be used for describing complex models with compact diagrammatic notations, allowing the definition of intuitive graphical languages. Graphical languages have been effectively used in a large number of domains like engineering, architecture, software modeling, music, cartography, etc. The automatic interpretation of such graphic documents requires two main processes: the description of symbols and their localization in the image (symbol spotting).

Symbol recognition in document images can be seen as a particular case of *Shape* Recognition. A good symbol recognition method requires a symbol descriptor that guarantees intra-class compactness and inter-class separability. It should be tolerant to noise, degradation, occlusions and distortion. Often symbols in graphical documents appear isolated, and thus, their descriptors should be invariant to rotation, scaling, and translation. Numerous *shape* descriptors, tolerant to such distortions, have been proposed based on contours (silhouettes) and regions [1]. They can also be divided in continuous approaches (which use a feature vector) and structural approaches.

The aim of Symbol Spotting is the localization of specific important information instead of analyzing the whole content of the document. First, the recognition of the whole document can be a very complex task (i.e. the analysis of historical documents). Secondly, a fast symbol detection technique (avoiding segmentation) is required for localizing symbols in large data sets. Symbol Spotting is related to indexing and retrieval, and it has been a very emerging topic of interest, applied to technical drawings or maps [2]. The spotting techniques can rely on different pattern recognition methods, such as geometric features, region-based approaches using connected components, or structural symbol representation [3].

In order to describe an object that can suffer from irregular deformations, the authors of [4] proposed a description strategy in which spatial arrangement of object parts is captured in a rectangular grid. Contiguous regions share information about their containing object points, and thus, the descriptor is tolerant to irregular deformations. The authors validated that the descriptor is suitable for the multi-class categorization of aligned symbols, outperforming state-of-the-art strategies [4]. In this paper, we present a method for symbol spotting which uses a new symbol descriptor, the Circular Blurred Shape Model (CBSM), which is rotationally invariant, able to cope with irregular deformations, and fast to compute. The method, as an extension of [4], codifies the spatial arrangement of object characteristics based on a prior blurring degree, which determines the *shape* deformation allowed to the object. Regions in a correlogram are used to vote object characteristics from neighbor regions. The new descriptor is learnt in a cascade of classifiers with Adaboost, and tested with a windowing strategy to perform symbol spotting. The validation of the method over plan and old musical score documents show the robustness of the presented approach. Moreover, state-of-the-art descriptors are compared in multi-class categorization problems, showing better performance of the new descriptor.

The paper is organized as follows: Section 2 presents the rotationally invariant CBSM descriptor and its spotting extension. Experiments are presented in Section 3. Finally, Section 5 concludes the paper.

2. CIRCULAR BLURRED SHAPE MODEL SPOTTING

In this section, we present a circular formulation of the Blurred Shape Model descriptor (CBSM). By defining a correlogram structure from the center of the object region, spatial arrangement of object parts is shared among regions defined by circles and sections. The method also allows a rotationally invariant description, rotating the correlogram by the predominant region densities. We divide the description of the algorithm into three main steps: the definition of the correlogram parameters, the descriptor computation, and the rotationally invariant procedure. We include a fourth step to extend the CBSM methodology to solve symbol spotting problems.

Correlogram definition: Given a number of circles C, number of sections S, and an image region I, a correlogram $B = \{b_{\{1,1\}}, .., b_{\{C,S\}}\}$ is defined as a radial distribution of sub-regions of the image, as shown in Figure 1(a) and (b). Each region b has centroid coordinates defined by b^*. Then, the regions around b are defined as the neighbors of b. Note that depending of the spatial location of the analyzed region, different number of neighbors can be defined (Fig. 1(c)).

Descriptor computation: In order to compute the descriptor, first, a pre-process of the input region I to obtain the shape features is required. Working with document images, relevant $shape$ information can be obtained by means of a contour map (although based on the object properties we can define other initial properties). In this paper, we use a Canny edge detector procedure.

Given the object contour map, each point from the image belonging to a contour is taken into account in the description process (Fig. 1(d)). First of all, the distances from the contour point **x** to the centroids of its corresponding region and neighbor regions are computed. The inverse of these distances are computed and normalized by the sum of total distances. These values are then added to the corresponding positions of the descriptor vector ν, including higher values to that positions corresponding to the nearest regions to **x** (Figure 1(e) bottom). This makes the description tolerant to irregular deformations. Note that for a map of k relevant contour points, the computation of the descriptor just requires k simple operations.

At this point we have a description ν for an input image I, where the length of ν, defined by parameters C and S, defines the degree of spatial information taken into account in the description process. In Figure 2, a bat instance from the public MPEG7 data set [5] is described with different $C \times S$ correlogram sizes. In the way that we increase the number of regions, the description becomes more local. Thus, an optimal parameters of C and S should be obtained for each particular problem (i.e. via cross-validation).

Rotationally invariant descriptor: In order to make the description rotationally invariant, a second step is included in the description process. We look for the main diagonal G_i of correlogram B with the highest density. This diagonal is then the reference to rotate the descriptor. The orientation of the rotational process, so that G_i is aligned with the x-axis, is that corresponding to the highest description density at both sides of G_i. This procedure is detailed in Algorithm 2. A visual result of the rotationally invariant process can be observed in Figure 2.

Algorithm 1 Circular Blurred Shape Model Description Algorithm.

Require: a binary image I, the number of circles C, and the number of sections S
Ensure: descriptor vector ν

Define $d = R/C$ and $g = S/360$, where R is the radius of the correlogram, as the distance between consecutive circles and the degrees between consecutive sectors, respectively (Figure 1(a)).

Define $B = \{b_{\{1,1\}}, .., b_{\{C,S\}}\}$ as the set of bins for the circular description of I, where $b_{c,s}$ is the bin of B between distance $[(c-1)d, cd)$ with respect to the origin of coordinates o, and between angles $[(s-1)g, sg)$ to the origin of coordinates o and x-axis (Figure 1(b)).

Define $b^*_{\{c,s\}} = (d \sin\alpha, d\cos\alpha)$, the centroid coordinates of bin $b_{\{c,s\}}$, where α is the angle between the centroid and the x-axis, and $B^* = \{b^*_{\{1,1\}}, .., b^*_{\{C,S\}}\}$ the set of centroids in B (Figure 1(e)).

Define $X_{b_{\{c,s\}}} = \{b_1, .., b_{cs}\}$ as the sorted set of the elements in B^* so that $d(b^*_{\{c,s\}}, b^*_i) \leq d(b^*_{\{c,s\}}, b^*_j), i < j$.

Define $N(b_{\{c,s\}})$ as the neighbor regions of $b_{\{c,s\}}$, defined by the initial elements of $X_{b_{\{c,s\}}}$:

$$N(b_{\{c,s\}}) = \begin{cases} X', |X'| = S+3 & \text{if } b_{\{c,s\}} \in IN \\ X', |X'| = 9 & \text{if } b_{\{c,s\}} \in MI \\ X', |X'| = 6 & \text{if } b_{\{c,s\}} \in EX \end{cases}$$

being X' the first elements of X, and IN, MI, and EX, the inner, middle, and extern regions of B, respectively (Figure 1(c)). Note that different number of neighbor regions appears depending of the location of the region in the correlogram. We consider the own region as the first neighbor.

Initialize $\nu_i = 0, i \in [1, .., CS]$, where the order of indexes in ν are:
$\nu = \{b_{\{1,1\}}, .., b_{\{1,S\}}, b_{\{2,1\}}, ..b_{\{2,S\}}, .., b_{\{C,1\}}, ..b_{\{C,S\}}\}$

for each point $\mathbf{x} \in I$, $I(\mathbf{x}) = 1$ (Figure 1(d)) **do**
  for each $b_{\{i,j\}} \in N(b_{\mathbf{x}})$ **do**
    $d_{\{i,j\}} = d(\mathbf{x}, b_{\{i,j\}}) = ||\mathbf{x} - b^*_{\{i,j\}}||^2$
  end for
  Update the probabilities vector ν positions as follows (Figure 1(f)):
  $\nu(b_{\{i,j\}}) = \nu(b_{\{i,j\}}) + \frac{1/d_{\{i,j\}}}{D_{\{i,j\}}}$,
  $D_{\{i,j\}} = \sum_{b_{\{m,n\}} \in N(b_{\{i,j\}})} \frac{1}{||\mathbf{x} - b^*_{\{m,n\}}||^2}$
end for
Normalize the vector ν as follows:
$d' = \sum_{i=1}^{CS} \nu_i, \nu_i = \frac{\nu_i}{d'}, \forall i \in [1, .., CS]$

Spotting CBSM extension: Once we have the rotationally invariant CBSM descriptor, we need to define two stages in order to design a symbol spotting methodology. A first stage should learn to distinguish among the target object and *background* (i.e. learning a binary classifier). A second stage should perform a search over the whole image using the learnt classifier in order to locate those regions containing the target object.

For the first step, we propose to learn a binary classifier using Adaboost [6] with a set of positive and negative object instances. Since we need to apply this classifier to a huge number of regions, the final detection time for an image is very expensive. In order to address this limitation, we learn the classifier using a cascade methodology [7]. Afterwards,

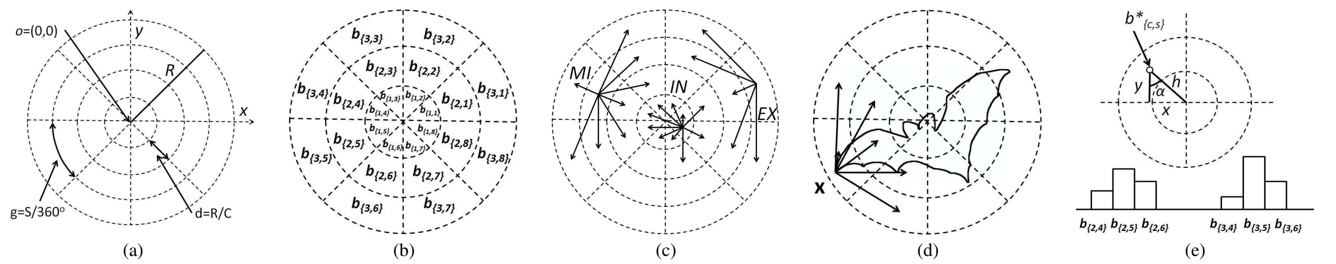

Fig. 1. (a) CBSM correlogram parameters, (b) regions distribution, (c) region neighbors, (d) object point analysis, (e) region centroid definition, and (e) descriptor vector update after point analysis.

Algorithm 2 Rotationally invariant ν description.

Require: ν, S, C
Ensure: Rotationally invariant descriptor vector ν^{ROT}

Define $G = \{G_1, .., G_{S/2}\}$ the $S/2$ diagonals of B, where $G_i = \{\nu(b_{\{1,i\}}), .., \nu(b_{\{C,i\}}), .., \nu(b_{\{1,i+S/2\}}), .., \nu(b_{\{C,i+S/2\}})\}$

Select G_i so that $\sum_{j=1}^{2C} G_i(j) \geq \sum_{j=1}^{2C} G_k(j), \forall k \in [1, .., S/2]$

Define L_G and R_G as the left and right areas of the selected G_i as follows:
$L_G = \sum_{j,k} \nu(b_{\{j,k\}}), j \in [1, .., C], k \in [i+1, .., i+S/2-1]$
$R_G = \sum_{j,k} \nu(b_{\{j,k\}}), j \in [1, .., C], k \in [i+S/2+1, .., i+S-1]$

if $L_G > R_G$ **then**
B is rotated $k = i + S/2 - 1$ positions to the left:
$\nu^{ROT} = \{\nu(b_{\{1,k+1\}}), .., \nu(b_{\{1,S\}}), \nu(b_{\{1,1\}}), .., \nu(b_{\{1,k\}}), ..,$
$, .., \nu(b_{\{C,k+1\}}), .., \nu(b_{\{C,S\}}), \nu(b_{\{C,1\}}), .., \nu(b_{\{C,k\}})\}$
else
B is rotated $k = i - 1$ positions to the right:
$\nu^{ROT} = \{\nu(b_{\{1,S\}}), .., \nu(b_{\{1,S-k+1\}}), \nu(b_{\{1,1\}}), .., \nu(b_{\{1,S-k\}}), ..,$
$, .., \nu(b_{\{C,S\}}), .., \nu(b_{\{C,S-k+1\}}), \nu(b_{\{C,1\}}), .., \nu(b_{\{C,S-k\}})\}$
end if

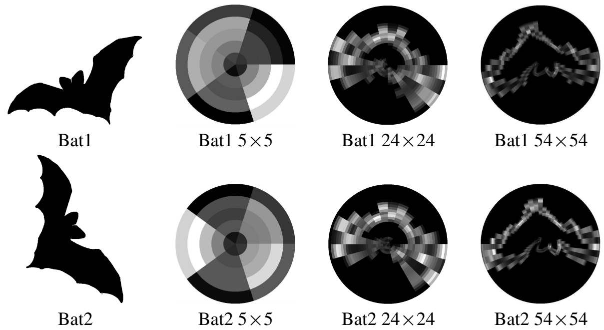

Fig. 2. Examples of image descriptions at different sizes for two object instances.

a windowing strategy is applied over the whole test images. Between a range of windows sizes, a window is tested around the image, and the cascade of classifiers is applied, detecting target symbols.

3. EXPERIMENTAL EVALUATION

Before the presentation the results of the CBSM methodology, first we discuss the data, methods, and validation of the experiments.

• *Data*: In order to test the object spotting methodology, we selected 20 predefined plan files of the Smart Draw software [8], and 20 old musical scores from a collection of modern and old musical scores (19th century) of the Archive of the Seminar of Barcelona. We also used the 70 object categories from the public MPEG7 binary object data set [5].

• *Methods*: we used the proposed CBSM descriptor, learning the C and S parameters via cross-validation, learning 10 levels of the cascade with Gentle Adaboost classifier with 50 decision stumps [6], and using as negative set 5000 random background images from Google. We also compare our descriptor with SIFT [9], Zoning, Zernique, CSS curvature descriptors from the standard MPEG [10] [1] [11], and the original rectangular BSM [4]. The optimum grid size of the CBSM descriptors is estimated applying cross-validation over the training set using a 10% of the samples to validate the different sizes of $S, C = \{8, 12, 16, 20, 24, 28, 32\}$. The Zoning and BSM descriptors are set to the same number of regions than the CBSM descriptor. Concerning the Zernique technique, 7 moments are used. The length of the curve for the CSS descriptor is normalized to 200, where the σ parameter takes an initial value of 1 and increases by 1 unit at each step. We also used a simple 3-Nearest Neighbor to compare multi-class categorization results among descriptors.

• *Validation*: 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. In the case of the object spotting problems, we apply the evaluation framework of [12] for the detection rate criterion. The accuracy is measured by the amount of overlap between the detected region and the labelled one. We consider that two regions are matched if they satisfy $1 - \frac{R_d \cap R_o}{R_d \cup R_o} < \epsilon$, where R_d is the detected region and R_o is the original one. We set the maximum overlap error ϵ to 40%, as in [12]. Moreover, we introduce the false alarm rate criterion, defined as the ratio between the number of detected regions that do not match which the original labelled ones and the total number of detected regions.

3.1. Symbol spotting in plan documents

In this experiment, we learnt a cascade of classifiers using 30 positive door symbols. Some results testing the spotting procedure with a windows displacement of five pixels are shown in Figure 3. Note that all the doors are detected even when connected with different wall types and over different rotation degrees. From the total number of doors in the 20 plan images, the 32 test doors were successfully detected using the previous accuracy measure, obtaining a hit ratio of 100%. Moreover, only one false positive region was detected, corresponding to a 3% of the positive detections, taking into account that the cascade has analyzed thousands of regions per image.

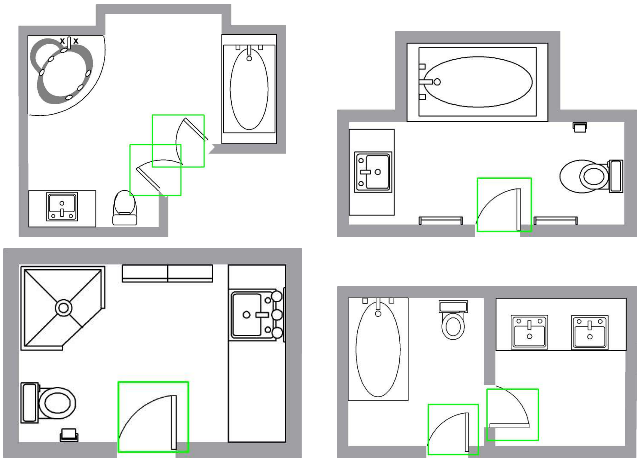

Fig. 3. Smartdraw door symbol detection results.

3.2. Symbol spotting in old musical scores

In this experiment, we learnt a cascade of classifiers using 144 positive clefs samples. Some results testing the spotting procedure with a windows displacement of five pixels over different staves are shown in Figure 4. Note that all the clefs are detected. At the bottom one false positive is detected. Note that under this false positive a rotation of the region appears, looking as the beginning of a stave, where the clefs appear. In this case, the degradation of the images reduces the accuracy in comparison to the previous case. In particular, from the total number of 30 test clefs in the images, 28 were successfully detected using the previous accuracy measure, which corresponds to a hit ratio of 93.33%. Regarding the false positives, 7 regions were detected.

3.3. CBSM for multi-class classification

In this experiment, we used the 70 object categories from the public MPEG7 binary object data set [5] to compare the descriptors in a multi-class categorization problem. The classification results and confidence interval using a 3-NN are shown in Table 1. Note that the best performance is obtained by our CBSM descriptor, followed by the original BSM.

CBSM	BSM	Zernique
71.84 (6.73)	65.79 (8.03)	43.64 (7.66)
Zoning	CSS	SIFT
58.64 (10.97)	37.01 (10.76)	29.14 (5.68)

Table 1. Classification results of state-of-the-art descriptors and 3-NN for the 70 classes of the public MPEG7 data set.

4. CONCLUSION

We presented the Circular Blurred Shape Model descriptor. The descriptor codifies the spatial arrangement of object parts based on a prior blurring degree. The descriptor has shown to be potentially useful to describe objects that may suffer from irregular deformations, such as the symbols that appear in document analysis. The descriptor is learnt using a cascade of classifiers with Adaboost to discard non-object regions, and tested over whole images, localizing the target objects. The symbol spotting procedure presented in this paper shown to robustly locate object instances in documents, such as symbols in plans and old musical scores. Moreover, the presented descriptor also outperforms the state-of-the-art descriptors when compared in multi-class object categorization problems.

Fig. 4. Clef detection in old stave images. A false positive is shown at the bottom of the figure.

5. REFERENCES

[1] D. Zhang and G. Lu, "Review of shape representation and description techniques," *Pattern Recognition*, vol. 37, no. 1, pp. 1–19, 2004.

[2] H. Samet and A. Soffer, "MARCO: MAp retrieval by COntent," *Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 783–798, 1996.

[3] D. Zuwala and S. Tabbone, "A Method for Symbol Spotting in Graphical Documents," *Lecture Notes in Computer Science*, vol. 3872, pp. 518.

[4] A. Fornes, S. Escalera, J. LLados, G. Sanchez, P. Radeva, and O. Pujol, "Handwritten symbol recognition by a boosted blurred shape model with error correction," in *Lecture Notes in Computer Science*. IBPRIA, 2007, vol. 4477, pp. 13–21.

[5] MPEG7 Repository, ," in *http://www.cis.temple.edu/latecki/research.html*.

[6] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of boosting," *Annals of Statistics*, 2000.

[7] P. Viola and M. Jones, "Robust real-time object detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2002.

[8] SmartDraw software, ," in *www.SmartDraw.com*.

[9] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[10] W.Y.Kim and Y.S.Kim, "A new region-based shape descriptor," Hanyang University and Konan Technology, 1999.

[11] Standard MPEG ISO/IEC 15938-5:2003(E), ," .

[12] K.Mikolajczyk, T. Tuytelaars, and C. Schmid, "A comparison of affine region detectors," *International Journal of Computer Vision*, vol. 65, no. 1-2, pp. 43–72, 2005.