New Approach for Symbol Recognition via Shape Context of Interest Points and Sparse Representation

Thanh-Ha Do
Université de Lorraine-LORIA UMR 7503
Campus scientifique - BP 239, France
Email: ha-thanh.do@loria.fr

Salvatore Tabbone
Université de Lorraine-LORIA UMR 7503
Campus scientifique - BP 239, France
Email: tabbone@loria.fr

Oriol Ramos Terrades
Universitat Autònoma de Barcelona
Computer Vision Centre, Espanya
Email: oriolrt@cvc.uab.cat

***Abstract*—In this paper, we propose a new approach for symbol description. Our method is built based on the combination of shape context of interest points descriptor and sparse representation. More specifically, we first learn a dictionary describing shape context of interest point descriptors. Then, based on information retrieval techniques, we build a vector model for each symbol based on its sparse representation in a visual vocabulary whose visual words are columns in the learned dictionary. The retrieval task is performed by ranking symbols based on similarity between vector models. Evaluation of our method, using benchmark datasets, demonstrates the validity of our approach and shows that it outperforms related state-of-the-art methods.**

I. Introduction

The problem of accurate localize queried symbols on technical documents, without segmented them before, is known in the Document Analysis community as symbol spotting. Symbol spotting system are composed of two phases: *symbol description* and *symbol retrieval*. The description phase consist of defining local shape descriptors, invariant to similarity transforms, robust to local symbol distortions and robust to document noise. The retrieval phase consist of implementing matching algorithms together indexing, or hashing, techniques to effective retrieve queried symbols. This paper focus in the symbol description phase, while symbol retrieval relies on state-of-the-art information retrieval techniques that have proven to perform well in video retrieval [18] as well symbol retrieval [17].

Regarding to shape descriptors and based on objective of encoding the shape of the symbol, Marçal *et. al.* [13] divide description techniques into three different main categories. These categories are *photometric* description, *geometric* description and *syntactic and structural* description. Photometric description is suitable for recognizing complex symbols, e.g., logos. Techniques in this category include SIFT descriptor [10], moment-based descriptors [4], [5], [8], and generic Fourier descriptor (GFD) [26]. SIFT descriptor characterizes each interest point by the local edge distribution around the point. This is very useful to describe complex symbols, but it loses effectiveness when representing simpler o (see Section IV). Moment-based descriptors have some advantages in symbol description comparing to other descriptors. For instance, they are invariant to translation, scaling, and rotation transforms. In addition, we can recover original symbols from moment descriptors [5], [8]. However, they still have some shortcomings, e.g., do not allow multi-resolution analysis of a shape in radial direction [27]. GFD is extracted from spectral domain by applying 2-D Fourier transformation on polar shape symbol. Hence, it overcomes the problem of noise sensitivity while still be invariant under the affine transformations.

Geometric description techniques are primitive-based methods that have shown to be useful when the symbols are non-isolated and be affected by occlusions [15], [16], [19]. Some examples of such primitives are contours, closed regions, connected components, skeletons, etc. to enumerate some of the most popular ones. Although these descriptors can easily be computed, they are usually poorly discriminant [16], and sometimes the matching process is time consuming [15]. In addition, there are also several primitives describing symbols using geometric information [12] or vector signatures [6]. In general, they are invariant under similarity transformations, but [6] depends on a prior normalization step to achieve invariance and [12] is very sensitive to noise at vector level.

There has been a great number of works on finding good *syntactic and structural* descriptions [3], [9], [14], [24]. These descriptors aim at defining relationships between primitives. In [3], [14], the authors propose rule-based descriptions whose performance are highly affected by noisy data. In [9], [24], the authors propose structural descriptors which present symbols as one-dimensional string or by an attribute relational graph. In general, graph-based descriptors are powerful tools for symbol description but the computation time linked to them is huge since we have to deal with sub-graph isomorphism, which is NP-hard. In summary, most of *photometric* descriptors requires well-segmented symbols to satisfactory perform while *geometric* descriptors have low discriminant capacities and *structural* descriptors are computationally demanding.

In this paper, we extend the work done in [17] by providing a sparse representation of local descriptors based on keypoints. Sparse representation has been widely used in image denoising, separating, and extracting the text regions [7], [28]. But, to best of our knowledge, sparse representation has not been applied to symbol descriptors. In this way, we achieve a sparse description of invariant descriptors which will improve the performance of retrieval systems.

More specifically, we first compute shape context of interest points in symbols and use them as training dataset for learning a sparse dictionary by means of the K-SVD algorithm [7]. Then, we consider the learned dictionary as a visual vocabulary whose visual words are each of the entries of this dictionary. Next, we construct a vector model for every symbol based on its sparse representation in the vocabulary and adapting the *tf-idf* approach to the sparse representation. Finally, the retrieval task is performed by ranking symbols based on their similarity to the query symbol and where the similarity is computed based on the vector model approach.

We have organized the rest of this paper as follows. In Section II, we present some fundamental background on shape context and shape context of interest points descriptors. Our proposed method and retrieval model are presented in Section III and we report experimental results in Section IV. Finally, we conclude and discuss the future work in Section V.

II. Related work

In this section, we present shape context and shape context of interest point background as well as their main invariant properties under rotation and scaling transforms.

A. Shape Context

Shape context (SC) is one of the descriptors with higher accuracy rates in many shapes recognition tasks and was introduced in [2]. Shape boundaries, either internal and external, are sampled in n points. For each point p_i on the symbol contour, [2] compute its coarse histogram h_i of the relative coordinates of the remaining $n-1$ points:

$$h_i(l) = \text{card}\{c \neq p_i : (c - p_i) \in bin(l)\}, l = \overline{1, L} \quad (1)$$

where c are contours points expressed in log-polar coordinates and L is the number of bins of the SC histogram at point p_i. Thus, for each symbol S, its SC is the real matrix $H = \{h_1, ..., h_n\}$ with dimensions $L \times n$.

Since all measurements are computed with respect to all points that are sampled from internal or external contours on the symbol, SC is therefore invariant under shape translation. Invariance under scaling is obtained by normalizing all radial distances by the mean distance among all point pairs in the symbol. Moreover, it is inherently insensitive to small perturbations of symbols, and indeed it is robust to small nonlinear transforms.

B. Shape Context of Interest Points

SC defined so far show two main drawbacks when applied to symbol retrieval tasks. On the one hand, as most of *photometric* descriptors, we need to segment symbols well enough for having satisfactory retrieval performance. On the other hand, matching function is computationally time-demanding if the number of boundary points is large.

Inspired by the works of detecting efficiently an object from its key-points (also known as interest points) [1], [11], the authors in [17] proposed a new approach, named *Shape context of interest point* (SCIP). In their approach, SC is only defined in detected interest points. More specifically, they detect the interest points $IP = \{p_1, p_2, \ldots, p_r\}$ and the contour points $C = \{c_1, \ldots, c_n\}$, of a given symbol. Indeed, each of these interest points p_i is thus a reference point to compute the SC of a symbol. Because the IP set is rarely a subset of C for most of the cases [20], the same rotation normalization method for SC cannot be applied to SCIP. Instead, they use dominant orientation of interest point information for orientation normalization. In more detail, each interest point p_i is represented by its coordinates and the dominant orientation: $p_i = \{x_i, y_i, \vec{e_i}\}$. The relative log-polar coordinates of contour points $c_j \in C$ is denoted by $c^i_j = (\log r_{ij}, \theta_{ij})$ in which r_{ij} is the normalized distance from p_i to c_j, and $\theta_{ij} = \langle \overrightarrow{p_i c_j}, \vec{e_i} \rangle$. The coarse histogram at p_i is computed as below.

$$\bar{h}_i(l) = \text{card}\{c^i_j \neq p_i : (c^i_j - p_i) \in bin(l)\}, l = \overline{1, L} \quad (2)$$

Then, the SCIP descriptor is the set $\bar{H} = \{\bar{h}_1, \bar{h}_2, \ldots, \bar{h}_r\}$, where each $\bar{h}_i$ is a histogram of L bins.

III. Sparse representation and symbol retrieval

Querying symbols on a dataset using SCIP descriptor needs of accurate assign each SCIP descriptor to a visual word. In the proposed approach we avoid this step by describing each SCIP descriptor by a linear combination of *visual* words being each entry of a learned dictionary A. In this section, we will first explain how to learn a dictionary from a set of SCIP descriptors providing sparse representation and then, how to built vector models permitting to us querying symbols in a dataset.

A. Learned Dictionary of SCIPs

An over-complete dictionary A for sparse representation is a dictionary built from a family of training signals in which each signal has an optimally sparse approximation in A.

In this paper, we use SCIPs $\{\bar{H}_n\}_{n=1}^N$ extracted from a set of N training symbols as a family of training signals. Each training signal $\bar{h}_i \in \mathbb{R}^L$ has an optimally sparse approximation x_i satisfying $\|\bar{h}^*_i - \bar{h}_i\|_2 \leq \epsilon$ with $\bar{h}^*_i = Ax_i$.

$$\min_{x_i, A} \sum_i \|x_i\|_1 \text{ subject to } \|\bar{h_i} - Ax_i\|_2^2 \leq \epsilon \quad (3)$$

Such a dictionary can be obtained by solving the problem defined in Equation 3. To do this, Aharon *et al* [7] proposed a 2 step iterative algorithm called K-SVD. In this algorithm, they iteratively adjust A via two main stages: *sparse coding* stage and *update dictionary* stage. In the sparse coding stage, all sparse representations $X = \{x_i\}_i$ of $Y = \{\bar{h_i}\}_i$ are computed while keeping A fixed. These sparse representations can be computed by an algorithm that approximates the solution of Equation 4. The algorithm used by the authors, and also by us in this approach, is the *orthogonal matching pursuit* (OMP) algorithm [23].

$$x_i = \min_x \|x\|_0 \text{ subject to } \|\bar{h_i} - Ax\|_2^2 \leq \epsilon \quad (4)$$

In the *update dictionary* stage, an updating rule is used to optimize the sparse representations of the training signals. In general, the way to update the dictionary is different from one learning algorithm to another. In K-SVD algorithm, the updating rule is applied column-wise on the dictionary's matrix A. Thus, each column a_{i_0} of A is updated sequentially such as minimizing the residually error in Equation 5:

$$\begin{aligned} \|Y - AX\|_F^2 &= \|(Y - \sum_{i \neq i_0} a_i x_i^T) - a_{i_0} x_{i_0}^T\|_F^2 \\ &= \|E_{i_0} - a_{i_0} x_{i_0}^T\|_F^2 \end{aligned} \quad (5)$$

since all columns in A other than i_0-th column are fixed, E_{i_0} is also fixed. Thus, the minimization of the Equation 5 depends only on the optimal a_{i_0} and $x_{i_0}^T$, where $x_{i_0}^T$ refers to the i_0-th row of X. This problem is therefore converted to a problem of approximating a matrix E_{i_0} by a rank 1 matrix by minimizing the Frobenius norm. Moreover, to ensure the sparsity in vector x, [7] defined a group of indexes Ω_{i_0} where $x_{i_0}^T$ is different to zeros, and $E_{i_0}^R$ is the E_{i_0} matrix restricted to those indexes. Then, the minimization of Equation 5 is equivalent to the minimization with respect to x_{i_0} of the next equation:

$$\|E_{i_0}\Omega_{i_0} - a_{i_0}x_{i_0}^T\Omega_{i_0}\|_F^2 = \|E_{i_0}^R - a_{i_0}x_{i_0}^R\|_F^2 \tag{6}$$

The optimal solution $x_{i_0}^R$ in Equation 6 has the same support as the original $x_{i_0}^T$, and can be find it by calculating the singular value decomposition (SVD) of the error matrix $E_{i_0}^R$.

B. Visual vector model

After applying the K-SVD algorithm on the set of SCIPs descriptors $\{\bar{H}_n\}_{n=1}^N$ used as training dataset, we have learned a dictionary $A \in \mathbb{R}^{L\times K}$ as well as the sparse representations of all SCIPs in that dataset. In the remainder of this section we will explain how we can built up a visual vector model from the sparse representation of SCIP descriptor that we will use for retrieval tasks in the experiments sections. Thus, the columns of the dictionary matrix A will play the role of words in a visual word vocabulary framework.

Without loss of generality, we can assume that $\bar{h} \in \mathbb{R}^L$ is one SCIP descriptor in $\{\bar{H}_n\}_{n=1}^N$ and $x \in \mathbb{R}^K$ is the sparse representation of $\bar{h}$ given the dictionary A. Instead of assigning a single visual word, typically the nearest centroid of a cluster given by a k-means algorithm like in [17], [18], $\bar{h}$ can be seen as a linear combination of visual words. Therefore, we have to adapt the vector model construction to the sparse representation of SCIP descriptors.

Let I be the indexes set where x is different to 0. We define the characteristic vector $v \in \mathbb{R}^K$ as being the 0-1 valued vector $v(k) = 1$ if $i \in I$ and 0 otherwise. For example, if $I = \{1, p-1, p, q, q+1\}$ then $v = \{1_1, 0_2, \dots, 1_{p-1}, 1_p, 0_{p+1}, \dots, 1_q, 1_{q+1}, 0_{q+2}, \dots, 0_K\}$. For each training symbol n, $\bar{H}_n = \{\bar{h}_1^n, \bar{h}_2^n, \dots, \bar{h}_{r_n}^n\}$ is its SCIP descriptor set and r_n is number of interest points detected for such symbol. Then, v_i^n denote the characteristic vector of $\bar{h}_i^n$, which is contributed by the sparse representation x_i^n of $\bar{h}_i^n$ given the learned dictionary A.

Similarly to the *tf-idf* approach used in information retrieval for building vector models, we define *tf* and *idf* factors to describe, respectively, the document contents and the importance degree of terms. Herein, documents and symbols are identified. Thus, f_k^n is the frequency of the word k in a symbol n and $tf_{k,n} = \frac{f_k^n}{\max_s f_s^n}$. Observe that we can easily computed these frequencies through the characteristic vector: $f_k^n = \sum_i^{r_n} v_i^n(k)$.

The *idf* factor is similarly defined as usual information retrieval systems but also adapting its definition to the sparse representation of SCIP descriptors. The importance in distinguishing a relevant symbol from non-relevant one in a database is measured by $\log \frac{N}{l_k}$, where l_k is the number of symbols in which the word k appears.

$$l_k = \text{card}\{n = \overline{1, N} | f_k^n \neq 0\}$$

Therefore, the vector model for a given symbol is defined by the weighted frequency for all words k in our visual vocabulary:

$$w^n(k) = tf_{k,n} \times idf_k = \frac{f_k^n}{\max_s f_s^n} \times \log \frac{N}{l_k}, \tag{7}$$

C. Retrieval Symbol

For each query symbol s_i^q in the set of query symbols $S_q = \{s_1^q, s_2^q, \dots, s_m^q\}$, its vector model is computed in the same way described in section III-B. We first compute its SCIP descriptor $\bar{H}_i^q = \{\bar{h}_1^q, \dots, \bar{h}_{r_i}^q\}$. Then, using the learned dictionary A, we find the sparse representation of each element in $\bar{H}_i^q$ by applying the OMP algorithm [23] to solve Equation 4. Finally, we compute the vector model, named w_i^q, as summarized in Equation 7.

Next, the similarity of a query symbol $s_i^q \subseteq S_q$ and symbols in a database $s^n \subseteq S$ is computed as the cosine distance between two vectors w_i^q and w^n:

$$\text{distance}(w_i^q, w^n) = \frac{\langle w_i^q, w^n\rangle}{|w_i^q| \times |w^n|} \tag{8}$$

where $\langle \cdot, \cdot \rangle$ is the dot product. Finally, symbols in the database are ranked based on their similarity to the query symbol s_i^q.

IV. Experimental Results

We have evaluated our proposed method in a benchmark synthetic dataset like GREC[1] dataset. This dataset contains the occurrences of 50 different symbols (the group A) obtained by linear transforms (rotation and scaling) and by applying deformation and degradations processes. We have used symbol in group A as queries to retrieve similar symbols to them in the following subsets: dataset D_1 includes 250 symbols generated by linear transforms (rotation and scaling); dataset D_2 includes 75 symbols generated by strong non-rigid transforms; and dataset D_3 includes 75 symbols generated by strong non-rigid transforms and Kanungo noise. The number of occurrences for each class is not the same for all of them: ranging from 1 to 10 times in dataset 1; from 1 to 11 in dataset 2 and ranging 0 to 11 in dataset 3.

We use the set $D_i, i = 1, \dots, 3$ to construct the learned dictionary using K-SVD algorithm with number of columns K is 512.

To examine the effectiveness of the proposed descriptor, we have compared it to 6 state-of-the-art methods for symbol recognition, namely R-signature [21], GFD [25], Zernike moments [22], SIFT [11], SC [2] and SCIP [17]. In addition, for Zernike moments descriptors, we have built two descriptors. The first descriptor, G_1 includes 32 low-order moments while the second descriptor, G_2, includes 32 hight-order moment. We have only considered the magnitude of Zernike moments for both descriptors G_1 and G_2 and for each one, the computed moments satisfy the following conditions:

[1] http://www.cvc.uab.es/grec2003/SymRecContest/

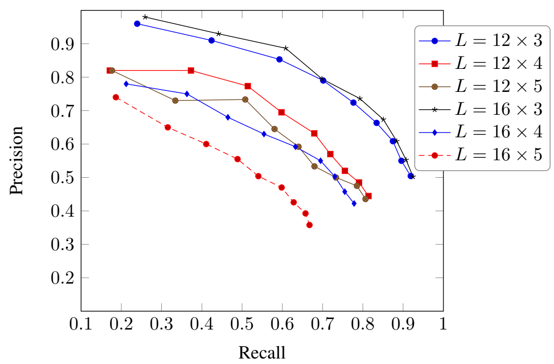

Fig. 1. The effect of L on the precision and recall rate

$$G_1 = \begin{cases} 3 \leq n \leq 10 \\ |m| \leq n \\ n - |m| = 2k \\ k \in N \end{cases} \qquad G_2 = \begin{cases} 10 \leq n \leq 17 \\ |m| \leq n \\ n - |m| = 4k \\ k \in N \end{cases}$$

For SIFT, SC and SCIP descriptors we have used k-means algorithm to build the visual dictionary, being each cluster centroid a visual word. The number of clusters is 175 for the three descriptors. The similarity measure for retrieval tasks is always the cosine distance, as defined in Equation 8. We have used *precision-recall* rate, denoted by R and P respectively, to evaluate the retrieval task:

$$\mathrm{R} = \frac{\text{the relevant symbols retrieved}}{\text{the relevant symbols existing in the database}}$$
$$\mathrm{P} = \frac{\text{the relevant symbols retrieved}}{\text{the number of retrieved symbols}}$$

Finally, we have conducted two experiments to asses the goodness of the proposed approach. The first experiment aims at finding the best size of SCIP descriptor (L) while the second experiment aims at evaluating the retrieval performance.

For the first experiment, devoted to find the best dictionary corresponding with the size of SCIP (L value), we have compared the averages of precision and recall rates performed on dataset D_1 for the number of bins for $\log r$ ranges from 3 to 5 and the number of bins for θ are 12, 16, respectively. So the descriptor dimension L belongs to $\{36, 48, 60, 64, 80\}$. Figure 1 shows that we have obtained the best precision and recall rates when $L = 36$ or $L = 64$.

We show results for the second experiment in figure 2. As it was also observe in [17], we can observe that SIFT descriptor loses its effectiveness when applied to symbols. SCIP descriptor have proved to be more suitable for symbol retrieval than SIFT in technical documents. Results also show that our proposed approach outperform the compared methods in datasets D_1 and D_2 while for the D_3 dataset GFD show better performance. Causes of this result can be explained by two facts. On the one hand, the key-point detection step is sensitive to noises. On the other hand, we have applied GFD to the whole image, since symbols in this datasets are fully segmented.

Nevertheless, comparing our proposed method to the others based on key-points extraction, SIFT and SCIP, which we can later applied on complex documents without pre-segmenting steps, we achieve better retrieval performance than them. It therefore shows, that building a visual vocabulary based on sparse representation provides better results than using cluster algorithms like k-means and assigning just one visual word to each local descriptor.

We give some retrieval example using our descriptor in Figure 3 for the reader to have a better qualitative assessment.

V. Conclusions and Future Work

In this paper, we have proposed a new approach for symbol description based on SCIP descriptors and sparse representation. This new approach is invariant under rotation, scaling, and distortion, since SCIP descriptor is. Also, it is well-adapted degraded and noise symbols since sparse approaches are robust to this kind of degradations.

Obtained results in a benchmark dataset have proven that our proposed descriptor is suitable for symbol description in retrieval task and improve related state-of-the-art methods. Indeed, by describing each SCIP descriptor as a linear combination of visual words, instead of only one 'visual word' for each shape context of interest points, we have achieve better system performance.

In the future, we would like to apply this approach to symbol spotting problem in the large graphical documents where symbols can not be easily segmented.

References

[1] S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11):1475–1490, November 2004.

[2] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(24):509–522, April 2002.

[3] H. Bunke. Attributed programmed graph grammars and their application to schematic diagram interpretation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 4(6):574–582, 1982.

[4] T. Cheng, J. Khan, H. Liu, and D. Yun. A symbol recognition system. In *Proceedings of the Seventh International Conference on Document Analysis and Recognition*, pages 918–921, 1993.

[5] C. Chong, P. Raveendran, and R. Mukundan. Translation and scale invariants of legendre moments. *Pattern Recognition*, 37(1):119–129, 2004.

[6] P. Dosch and J. Lladós. Vectorial signatures for symbol discrimination. In *Graphics Recognition: Recent Advances and Perspectives, Lecture Notes on Computer Science*, volume 3088, pages 154–165, 2004.

[7] M. Elad, M. Aharon, and A. M. Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on signal processing*, 54(11):4311–4322, 2006.

[8] A. Khotanzad and Y. Hong. Invariant image recognition by zernike moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(5):489–497, 1990.

[9] J. Lladós, E. Martí, and J. Villanueva. Symbol recognition by error-tolerant subgraph matching between region adjacency graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):1137–1143, 2001.

[10] D. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, pages 1150–1157, 1999.

[11] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[12] G. Lu and A. Sajjanhar. Region-based shape representation and similarity measure suitable for content-based image retrieval. *Multimedia Systems*, 7(2):165–174, 1999.

[13] R. Marçal and J. Lladós. *Symbol Spotting in Digital Libraries*. Springer, 2010.

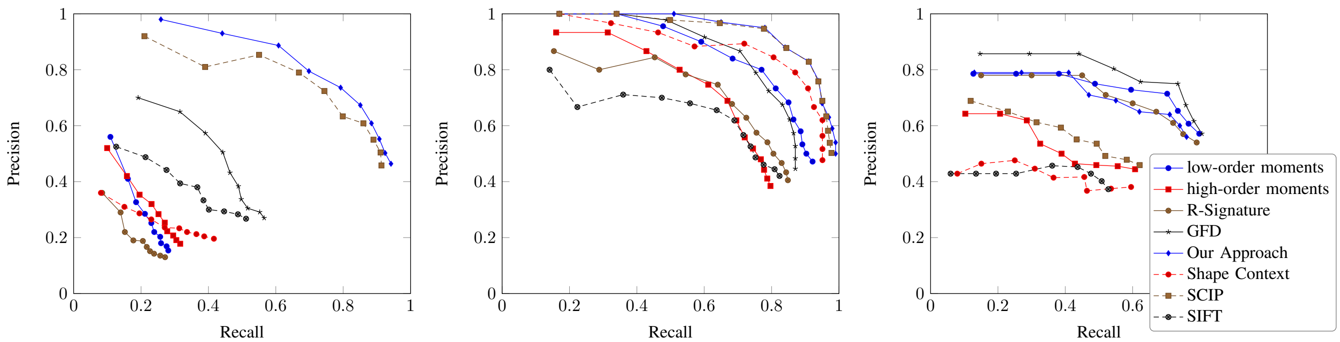

Fig. 2. Retrieval effectiveness with recall rate and precision rate in datasets: D_1 (left), D_2 (center) and D_3 (right). The legend is the same for all plots.

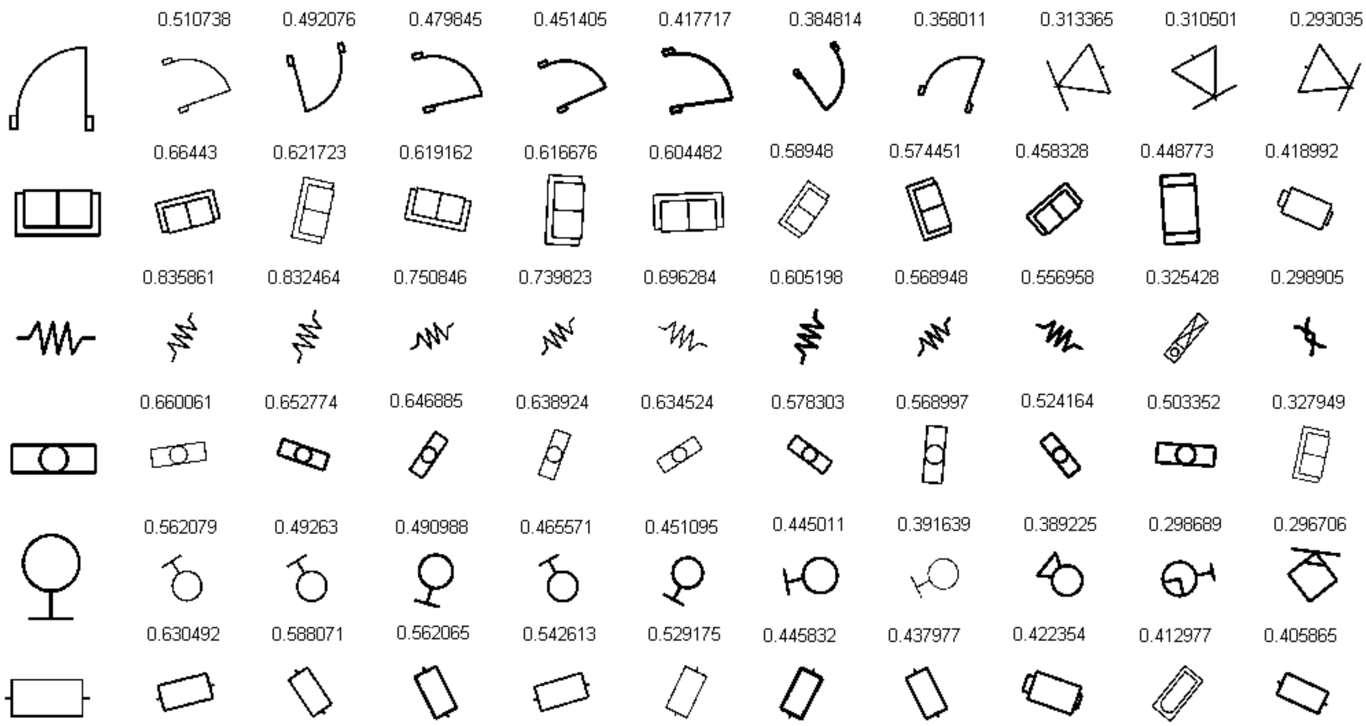

Fig. 3. Some retrieval examples in D_1: the query symbol is in the first column; other columns are the nearest matches ranked from left to right.

[14] J. Mas, J. Jorge, G. Sánchez, and J. Lladós. Representing and parsing sketched symbols using adjacency grammars and a grid-directed parser. In *Graphics Recognition. Recent Advances and New Opportunities, Lecture Notes on Computer Science*, volume 5046, pages 169–180, 2008.

[15] F. Mokhtarian, S. Abbasi, and J. Kittler. Robust and efficient shape indexing through curvature scale space. In *Proceedings of the British Machine Vision Conference*, pages 53–62, 1996.

[16] J. Neumann, H. Samet, and A. Soffer. Integration of local and global shape analysis for logo classification. *Pattern Recognition Letters*, 23(12):1449–1457, 2002.

[17] T. O. Nguyen, S. Tabbone, and O. R. Terrades. Symbol descriptor based on shape context and vector model of information retrieval. In *The 8th IAPR International Workshop on Document Analysis Systems*, Nara, Japan, 2008.

[18] J. Sivic and A. Zisserman. Video google: a text retrieval approach to object matching in videos. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 1470 –1477 vol.2, oct. 2003.

[19] F. Stein and G. Medioni. Structural indexing: Efficient 2d object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(12):1198–1204, 1992.

[20] S. Tabbone, L. Alonso, and D. Ziou. Behavior of the laplacian of gaussian extrema. *Journal of Mathematical Imaging and Vision*, 23(1):107–128, July 2005.

[21] S. Tabbone, L. Wendling, and J.-P. Salmon. A new shape descriptor defined on the radon transform. *Computer Vision and Image Understanding*, 102(1):42–51, , April 2006.

[22] A. Tahmasbi, F. Saki, and S. B. Shokouhi. Classification of benign and malignant masses based on zernike moments. *Computers in Biology and Medicine*, 41(8):726–735, August 2011.

[23] J. Tropp and A. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *Information Theory, IEEE Transactions on*, 53(12):4655 –4666, dec. 2007.

[24] H. Wolfson. On curve matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(5):483–489, 1990.

[25] D. Zhang. Generic fourier descriptor for shape-based image retrieval. In *International Conference on Multimedia and Expo*, volume 1, pages 425–428, 2002.

[26] D. Zhang and G. Lu. Shape-based image retrieval using generic fourier

descriptor. *Signal Processing*, 17:825–848, 2002.
[27] D. Zhang and G. Lu. Review of shape representation and description techniques. *Pattern Recognition*, 37:1–19, 2004.
[28] M. Zhao, S. Li, and J. Kwok. Text detection in images using sparse representation with discriminative dictionaries. *Image and Vision Computing*, 28:1590–1599, 2010.