2009 10th International Conference on Document Analysis and Recognition

Mathematical Symbol Indexing using Topologically Ordered Clusters of Shape Contexts

Simone Marinai, Beatrice Miotti, Giovanni Soda
Dipartimento di Sistemi e Informatica - Università di Firenze
Via S.Marta, 3 - 50139 Firenze - Italy
marinai@dsi.unifi.it

Abstract

This paper addresses the indexing and retrieval of mathematical symbols from digitized documents. The proposed approach exploits Shape Contexts (SC) to describe the shape of mathematical symbols. Starting from the vector space method, that is based on SC clustering, we explore the use of topological ordered clusters to improve the retrieval performance. The clustering is computed by means of Self-Organizing Maps that organize the clusters in two dimensional topologically ordered feature maps. The retrieval performance are compared with those obtained using the K-means clustering on a large collection of mathematical symbols gathered from the widely used INFTY database.

1. Introduction

The recognition of printed documents containing mathematical expressions is a challenging problem that has received a growing attention in the last few years. This interest is partially motivated by a larger number of scientific and technical documents that are nowadays available in digital libraries. Moreover, several efforts have been devoted to the building of specific mathematics libraries, such as the *Digital Library of Mathematical Functions* [8]. Even if earlier attempts to recognize mathematical expressions date back to the late 1970's [2], in the last decade a renewed interest has been pushed by digital library initiatives and by specific research projects such as the Infty project [10].

When using document image analysis techniques with documents belonging to digital libraries two main approaches can be considered: either we recognize and subsequently index all the documents or we directly index the documents at the image level [7]. In the first case a complete and accurate conversion must be performed and possible recognition errors can be hardly managed in the retrieval phase. The second approach can be advantageous when the recognition of the document content is difficult, however scalability issues must be taken into account.

Document image retrieval techniques have been seldom used to process mathematical expressions. However, several researchers envisage the utility of math search systems that would be able not only to search for text, but also for "fine-grain mathematical data" (e.g. equations and functions) [14]. Most search systems for mathematical documents rely on markup languages designed for mathematical formulae. One example is the MathWebSearch system that harvests the Web for formulae, subsequently indexed using MathML or OpenMath representations [3].

In our recent work we focused on the use of document image retrieval techniques in the field of digital libraries. For instance, in [6] we proposed a technique for efficient word retrieval from printed documents. To address the mathematical symbol indexing, in analogy with recognition-based approaches [2], we need to address two complementary aspects: how to index variable size symbols belonging to a large number of classes [12]; how to deal with the spatial arrangement of symbols in the mathematical pattern. This paper focuses on the first problem and we propose a symbol indexing method that we expect could be adopted also in other domains (e.g. to index graphical symbols).

The proposed indexing approach belongs to the family of methods based on the '*bag of visual words*' framework, that has been recently proposed in computer vision applications. These methods extend the '*bag of words*' model used in textual information retrieval, that represents documents considering the number of occurrences of words, regardless of their position in the text [13]. The overall approach involves three main steps (additional details are reported in Section 2). First, key-points are extracted from the collection of shapes to be indexed and represented with a numerical feature vector. Second, vector quantization is performed by clustering the vectors and then representing each vector by the index of the cluster it belongs to. Third, in the textual analogy each cluster index is regarded as a word and

978-0-7695-3725-2/09 $25.00 © 2009 IEEE
DOI 10.1109/ICDAR.2009.120

therefore each shape is represented by means of the frequencies of each '*visual word*' in its description. The ranking of most similar shapes is based on the classical *tf-idf* weighting scheme. The similarity is computed by means of the cosine of the angle between weight vectors, in analogy with the vector space model in Information Retrieval.

The methods proposed in the literature differ in the vectorial representation of interest points and in the similarity computation. However, in most cases partitioning clustering methods, in particular K-means, are applied.

One key feature of the method proposed in this paper is the use of a vector quantization computed with a clustering obtained by means of Self Organizing Maps (SOM). Clusters in SOMs are organized in a rectangular grid (usually in two dimensions) and most similar clusters are closer in the map. To take advantage of the topological organization of the neurons in the map we also propose an extension of the similarity computation.

The paper is organized as follows. In Section 2 we summarize the general approach of indexing by bag of visual words, whereas in Section 3 we describe the proposed extensions to this approach. Experimental results for the retrieval of mathematical symbols are reported in Section 4 and conlusions are drawn in Section 5.

2. Indexing by bag of visual words

Similarly to words in a text document, an image contains local interest points that concentrate most of its information. Structural shape representations identify the interest points by finding, for instance, local maxima in the contour curvature or crossing points in the skeleton. These methods provide a semantical meaningful representation, but are in general sensitive to noise. To reduce this sensitivity, some approaches sample the shape contour and compute a local descriptor in each sampled point. Shape Contexts (SCs) are descriptors of this type that have been used also for document image retrieval [5]. The comparison between Shape Contexts in a query and in an indexed object can provide a very accurate similarity evaluation among the two objects. However, this process can be time consuming and cannot be considered when dealing with large data-sets.

To address the latter problem the shape representation is transformed adopting techniques used in the vector space method of Information Retrieval. The vector quantization is performed by clustering the vector representations and then identifying each vector by the index of the cluster it belongs to. Going on with the textual analogy, each cluster is considered as a word and each symbol can be represented by the frequencies of each 'visual-word' in its description [13].

Shape descriptors have been widely used in several fields in computer vision. In the area of document image analysis most work has been performed for graphical symbol recognition/indexing. In this domain, a recent paper compared various shape descriptors for symbol representation [11]. An extension of the *bag of visual words* method based on Shape Context has been proposed in [9] for symbol retrieval. The peculiarity of the method described in [9] is the introduction of Shape Context descriptors computed on interest points.

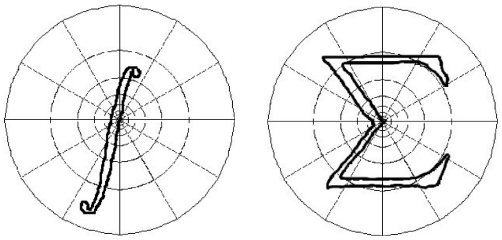

Figure 1. Two examples of grid overlapped to the mathematical symbol used to compute the Shape Context.

The Shape Context method is based on a simple and robust algorithm that uses silhouettes of the objects as shape descriptors [1]. The shape of the symbol is treated as a set of contour points and the symbol shape is described by a subset of its contour points.

Let $P = \{p_1, \ldots, p_m\}$, $p_i \in \Re^2$, be the set of m contour points. Unlike the method described in [9], we do not require that these points are key-points such as maximum or corner points. In our approach, the points in P are sampled from the contour by a roughly uniform spacing so that the sample points are separated by a minimum distance.

One Shape Context can be computed for each point p_i by taking in account the relative position of the other points in P. The SC gives an idea of the spatial distribution of points in the proximity of p_i. Since in general similar shapes have similar descriptors, various symbols can be compared considering the SCs. The SC for the point p_i is obtained by computing a coarse histogram h_i whose bins are uniform in log-polar space (Figure 1). The remaining $m - 1$ points of the symbol contour are assigned to the appropriate bin according to the logarithm of the Euclidean distance between p_i and the point itself and to the angle formed by the link between the two points and the horizontal axis of the diagram (Figure 1). The histogram h_i is defined to be the Shape Context of p_i. The m shape context vectors describe the configuration of the entire symbol. Shape contexts are invariant to translations since all the measures are taken with respect to points on the object. SCs can be adapted to be scale invariant as well as rotation invariant [1]. However, in our work we did not implement the rotation invariance of shape contexts to avoid ambiguities in the retrieval process

that could bring to confuse, for instance, 6 with 9. We use a log-polar diagram with eight bins for $\log r$ and 12 bins for the angle θ. Therefore, the size of the SC, corresponding to the number of bins in the histogram, is $n = 96$.

By clustering the SCs it is possible to build a visual vocabulary that is subsequently used to describe each symbol into a vector of size C (the number of clusters). Each symbol s_j is represented by a weight vector $\vec{W}_j = (w_{1,j}, w_{2,j}, \ldots, w_{C,j})$ that is computed starting from the frequency of SCs belonging to each cluster c_i in the symbol s_j. One of the most common weighting schemes in Information Retrieval is the *tf.idf* one where the weight $w_{i,j}$ is computed as follows:

$$w_{i,j} = tf_{i,j} \cdot idf_i = \frac{freq_{i,j}}{max_k(freq_{k,j})} \cdot log(N/n_i) \quad (1)$$

$freq_{i,j}$ is the *term frequency*: the number of occurrences of Shape Contexts belonging to cluster c_i in the symbol s_j and $log(N/n_i)$ is the *inverse of the document frequency* (N is the total number of symbols, n_i is the number of documents containing SCs clustered into c_i). A simple interpretation of the *tf.idf* weighting is that the weight is zero in two cases: when the term is missing in the document and when the term occurs in all the documents and therefore it does not bring any information.

The similarity between two symbols s_a and s_b is evaluated by computing the cosine of the angle between the two vectors $\vec{W}_a$ and $\vec{W}_b$ that belong to the vector space defined by the set of clusters:

$$sim(s_a, s_b) = \frac{\vec{W}_a \bullet \vec{W}_b}{|\vec{W}_a| \cdot |\vec{W}_b|} \quad (2)$$

where $\vec{W}_a \bullet \vec{W}_b$ denotes the scalar product between $\vec{W}_a$ and $\vec{W}_b$. In the next section we describe the main changes to this approach that are proposed in this paper.

3. Shape context vector quantization

In this section we focus on the Shape Context vector quantization that is used to build the visual dictionary and to label symbols on the basis of this dictionary. In most cases the K-means clustering algorithm is used for vector quantization. This algorithm is simple to implement and performs well in most cases. However, as with other partitioning methods, there can be problems when trying to cluster objects that smoothly change and clusters are not clearly separable. In this case the objects laying close to the frontier between two clusters can be assigned to either clusters depending on a small change in the numerical values of their feature vectors. This feature can be problematic in the *bag of visual words* approach, since two very similar shape contexts could be assigned to different clusters. As a consequence, the contribution of these visual words to the similarity between the corresponding symbols will be zero, even if the original shape contexts are rather similar.

To address this problem we propose to use the Self Organizing Map (SOM) to perform the vector quantization and we modify the similarity computation so as to take into account the peculiarity of SOM.

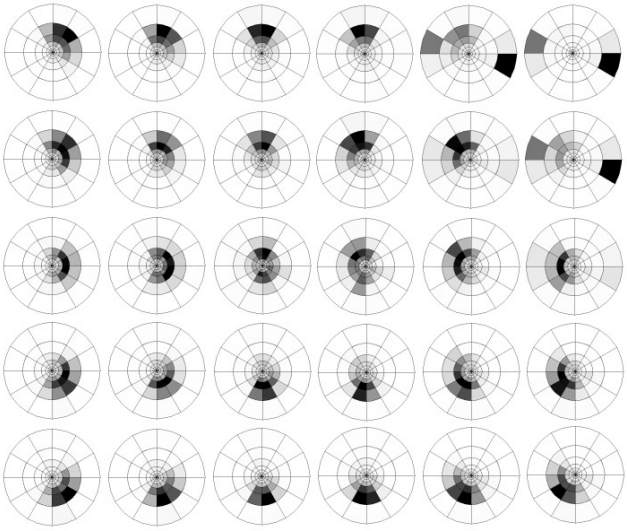

Figure 2. One SOM of Shape Contexts. The neurons are displayed with a graphical representation of Shape Contexts corresponding to cluster centroids.

The Self-Organizing Map (SOM) is a type of artificial neural network that is based on unsupervised learning [4]. The SOM neurons are typically arranged in a two dimensional lattice. Each neuron is associated with a weight vector $\vec{m}_i = [m_{i,1}, m_{i,2}, ..., m_{i,n}]^T \in \Re^n$. During the learning, each input vector $x \in \Re^n$ is compared with all the weight vectors of the SOM and the Best Match Unit (BMU) is found. The BMU is the neuron closest to the input vector according to a given distance (in most cases the Euclidean distance). The BMU c is therefore defined by

$$c = arg \min_i \parallel \vec{x} - \vec{m}_i \parallel \quad (3)$$

where $\vec{x}$ is an input vector and $\vec{m}_i$ is the weight vector associated with neuron i, named *centroid* or *codebook vector* in SOM's terminology. The input vector $\vec{x}$ is then mapped into the cluster represented by neuron c, and this neuron and its neighbors are updated. In the update process, two types of rules may be used: the *incremental* learning rule and the *batch* learning rule. In the incremental learning, that we used in our experiments, the update process is performed using one data vector at a time, and the centroid is updated with the following rule:

$$m_i(t+1) = m_i(t) + h_{ci}[x(t) - m_i(t)] \quad (4)$$

where t is a step in the learning process, $x(t)$ is the input vector, $m_i(t)$ is the centroid to update and h_{ci} is the neighborhood function [4].

One important feature of SOM maps is that more similar clusters are usually closer in the feature map. To explain this concept, we report in Figure 2 one SOM that is obtained when clustering the Shape Contexts extracted from the mathematical symbols indexed in the experiments described in Section 4. The map contains 30 neurons (clusters) that are organized into a 6 x 5 grid. Each cluster is represented by its centroid that is stored in the corresponding weight vector. To have an idea of the clustering achieved, we depict in the figure one graphical representation for each cluster center. Basically, we fill the bins in the Shape Context with a gray level that is darker for higher values in the feature vector, that correspond to higher values in these bins for the SCs in the corresponding cluster. Each cluster can be considered as a term in the visual dictionary and SCs in a symbol are labeled according to this dictionary. By checking the cluster centers we can notice that closer neurons are in general more similar and there is also a smooth change between neighboring neurons. For instance, the last row in the map represents in some way the SCs that correspond to upper corners (such as those occurring in the top part of an 'M'). Browsing the row from left to right we can notice a smooth rotation of the represented corner in the clockwise direction.

Considering the definition of scalar product we can rewrite Eq. (2) as:

$$sim(s_a, s_b) = \frac{\sum_{i=1}^{C} w_{i,a} \cdot w_{i,b}}{|\vec{W_a}| \cdot |\vec{W_b}|} \quad (5)$$

The components in the summation are different from zero only when both $w_{i,a}$ and $w_{i,b}$ are different from zero. However, when two similar shape contexts are not mapped to the same cluster, their contribution to $sim(s_a, s_b)$ is zero. We therefore modified Eq. (2) as follows:

$$sim'(s_a, s_b) = sim(s_a, s_b) + \\ + \frac{\sum_{(i|w_{i,b}=0)} w_{i,a} \cdot \frac{Max(Neigh(w_{i,b}))}{4}}{|\vec{W_a}| \cdot |\vec{W_b}|} \quad (6)$$

In the second part we compute a value in the summation for the terms in $\vec{w_a}$ having no counterpart in $\vec{w_b}$, but with at least one neighborhood in the SOM map that is different from zero; $Max(Neigh(w_{i,b}))$ is the maximum value in $\vec{W}$ among the four SOM neighborhoods of $w_{i,b}$.

In the next section we briefly analyze the experiments that we performed to numerically evaluate the proposed approach.

training pages	number of clusters	K-mean	SOM (sim)	SOM (sim')
10	25	63.38	70.06	**70.81**
20	30	62.19	69.57	69.33
30	30	61.28	65.14	66.09

Table 1. Percentage of correct hits among the first 50 retrieved symbols for three compared methods.

training pages	number of clusters	K-mean	SOM (sim)	SOM (sim')
10	25	74.38	79.90	**80.00**
20	30	69.81	78.95	78.76
30	30	72.10	72.95	72.86

Table 2. Percentage of correct hits among the first 25 retrieved symbols for three compared methods.

4. Experimental Results

In our experiments we used one dataset collected by of the INFTY project [10]. It contains 26 scientific papers and 25 documents in English extracted by several math journals. The total number of pages is 476. The dataset is supplied with a set of files that provide groundtruth information about the symbols in the pages. The dataset includes, in addition to standard characters from the English alphabet, several typologies of characters: Greek, Roman and German and math symbols like integrals, summations, logical operators etc.

To reduce the computational burden in the experiments, we used 394 documents randomly selected from the INFTY dataset, for a total of 628,652 symbols.

We performed several tests to compare the results that can be achieved when using the K-means clustering and the proposed SOM-based clustering (see Tables 1 and 2). The values in the tables are computed considering the percentage of correct hits in response of 42 queries made with various symbols (some examples are shown in Figure 3). In most cases all the top 50 symbols are correct. However, for some symbols we have few answers that are probably due to a low number of data in the dataset.

From these experiments it is clear that the SOM clustering shows better performance with respect to K-means. The extended similarity computation (sim') is slightly better than sim but additional experiments are needed to confirm this finding.

The number of training patterns used for clustering is

comprised between 282,450 (10 pages) and 806,760 (30 pages). It is interesting to notice that methods based on a reduced number of clusters which are computed with fewer training samples, perform better than the others. The clustering time for K-mean is comprised between 60 minutes for 10 pages and 150 minutes for 30 pages (51 to 110 minutes with the SOM). The retrieval time per query is 37 seconds. Most of this time is needed to extract the images of the retrieved symbols from the pages in the database.

5. Conclusions

In this paper we proposed a technique for image-based symbol retrieval based on Shape Context representation encoded with a *bag of visual words* method. The peculiarity of the proposed approach is the use of Self Organizing Maps for clustering the Shape Contexts into a suitable visual dictionary. Experiments performed on a large data set containing alphabetical and mathematical symbols allows us to positively evaluate the proposed approach.

Future work includes a more extensive evaluation of the retrieval performance on the whole INFTY dataset and a deeper comparison of various retrieval approaches. We aim also to extend the retrieval mechanism to incorporate the structural information of formulae in the retrieval algorithm.

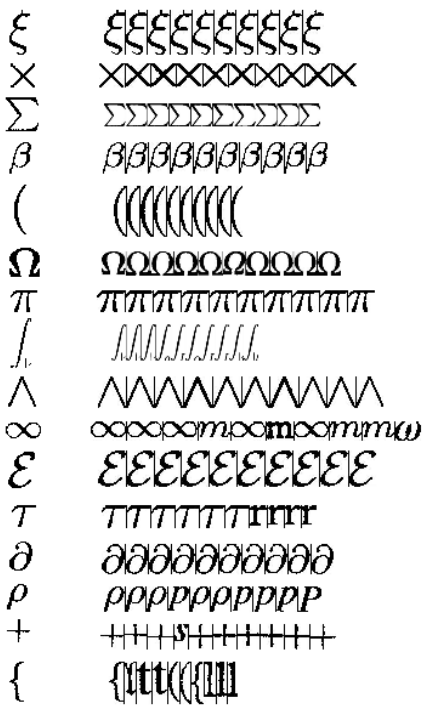

Figure 3. Example of retrieval results for some query symbols.

References

[1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(4):509–522, Apr 2002.

[2] K.-F. Chan and D.-Y. Yeung. Mathematical expression recognition: a survey. *IJDAR*, 3(1):3–15, 2000.

[3] M. Kohlhase and I. Sucan. A search engine for mathematical formulae. In *Artificial Intelligence and Symbolic Computation*, pages 241–253. Springer Verlag- LNCS 4120, 2006.

[4] T. Kohonen. *Self-organizing maps*. Springer Series in Information Sciences, 2001.

[5] J. Llados and G. Sanchez. Indexing historical documents by word shape signatures. In *ICDAR '07: Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007) Vol 1*, pages 362–366, Washington, DC, USA, 2007. IEEE Computer Society.

[6] S. Marinai, E. Marino, and G. Soda. Font adaptive word indexing of modern printed documents. *IEEE Transactions on PAMI*, 28(8):1187–1199, 2006.

[7] S. Marinai, E. Marino, and G. Soda. Exploring digital libraries with document image retrieval. In *ECDL*, pages 368–379, 2007.

[8] B. R. Miller and A. Youssef. Technical aspects of the digital library of mathematical functions. *Annals of Mathematics and Artificial Intelligence*, 38:121–136, 2003.

[9] T.-O. Nguyen, S. Tabbone, and O. R. Terrades. Symbol descriptor based on shape context and vector model of information retrieval. In *DAS '08: Proceedings of the 2008 The Eighth IAPR International Workshop on Document Analysis Systems*, pages 191–197, Washington, DC, USA, 2008. IEEE Computer Society.

[10] M. Suzuki, F. Tamari, R. Fukuda, S. Uchida, and T. Kanahori. Infty: an integrated ocr system for mathematical documents. In *DocEng 03: Proceedings of the 2003 ACM symposium on Document engineering*, pages 95–104, New York, NY, USA, 2003. ACM.

[11] E. Valveny, S. Tabbone, O. Ramos, and E. Philippot. Performance characterization of shape descriptors for symbol representation. In *Graphics Recognition. Recent Advances and New Opportunities: 7th International Workshop, GREC 2007, Curitiba, Brazil, September 20-21, 2007. Selected Papers*, pages 278–287, Berlin, Heidelberg, 2008. Springer-Verlag.

[12] S. M. Watt. An empirical measure on the set of symbols occurring in engineering mathematics texts. *Document Analysis Systems, 2008. DAS '08. The Eighth IAPR International Workshop on*, pages 557–564, Sept. 2008.

[13] J. Yang, Y.-G. Jiang, A. G. Hauptmann, and C.-W. Ngo. Evaluating bag-of-visual-words representations in scene classification. In *MIR '07: Proceedings of the international workshop on Workshop on multimedia information retrieval*, pages 197–206, New York, NY, USA, 2007. ACM.

[14] A. Youssef. Roles of math search in mathematics. In *Mathematical Knowledge Management MKM 2006*, pages 2–16. Springer Verlag- LNCS 4108, 2006.