Efficient Logo Retrieval Through Hashing Shape Context Descriptors

Marçal Rusiñol
Computer Vision Center
Dept. Ciències de la Computació
Edifici O, Univ. Autònoma de Barcelona
08193 Bellaterra (Barcelona), Spain
marcal@cvc.uab.es

Josep Lladós
Computer Vision Center
Dept. Ciències de la Computació
Edifici O, Univ. Autònoma de Barcelona
08193 Bellaterra (Barcelona), Spain
josep@cvc.uab.es

ABSTRACT
In this paper we present a method for organizing and indexing logo digital libraries like the ones of the patent and trademark offices. We propose an efficient queried-by-example retrieval system which is able to retrieve logos by similarity from large databases of logo images. Logos are compactly described by a variant of the shape context descriptor. These descriptors are then indexed by a locality-sensitive hashing data structure aiming to perform approximate k-NN search in high dimensional spaces in sub-linear time. The experiments demonstrate the effectiveness and efficiency of this system on realistic datasets as the Tobacco-800 logo database.

Categories and Subject Descriptors
I.7.5 [**Document and Text Processing**]: Document Capture—*Graphics recognition and interpretation*; H.3.7 [**Information Storage and Retrieval**]: Digital Libraries; H.2.8 [**Database Management**]: Database Applications—*Image databases*

General Terms
Algorithms

Keywords
Logo retrieval, Graphics recognition

1. INTRODUCTION
The Document Image Analysis and Recognition (DIAR) field has devoted, since its early years, many research efforts to extract information from business documents such as invoices, receipts, faxes, etc. Usually, the contributions in this specific topic just relied on the analysis of the layout structure of the documents and on the extraction and recognition of the text. However, in many cases, the graphic elements that might be present in these documents also convey important information. In this context, the problem of logo recognition is one of the most interesting and with more promising applications within the field of graphics recognition.

Given that logos are carefully designed to unequivocally represent and to easily identify a given corporate image, the task of recognizing logos appearing in documents is an important step forward in the identification of the document's source. For instance, if a company receives a document containing the logo of a bank, usually this document should be forwarded to the accounting department, whereas if the document contains the logo of a computer supplier, it is quite probable that the document should be addressed to the IT department. The recognition of logos can help to introduce contextual information in order to overcome the semantic gap between the simple recognition of characters and the semantic document understanding.

One of the most interesting applications of the recognition of logos is the organization and indexing of large trademark databases as the ones they have in the different patent and trademark offices. These digital libraries might contain millions of logo images and it is interesting when registering a new trademark to search if there exist other companies having a similar logo design. However, nowadays these immense image collections are difficult to browse by content. As an example, the UK Intellectual Property Office website[1] offers a way to navigate through the trademark collection by searching by trademark image class. Each logo is labelled with a set of previously manually harvested metadata which enable to search in the collection by a set of predefined labels. The most widely used metadata classification codes are the ones from the Vienna classification system, developed by the World Intellectual Property Organization [19]. We can see in Fig. 1 an example of retrieving logos from the UK Intellectual Property Office website by the category tree of Vienna codes. This manually annotation of the logos' contents has the advantage of clustering the logos by semantic information, so even if the logos in Fig. 1 are not visually similar they can be clustered together because they all represent an owl. However this approach also presents the drawbacks of being a tedious task, sometimes subjective and of course very expensive. In addition, the use of manual classification codes is not always distinctive, specially for abstract or artistic images. In that specific scenario it would be very interesting if we could use some of the logo descrip-

[1]http://www.ipo.gov.uk/tm/t-find/t-find-text/

DAS '10, June 9-11, 2010, Boston, MA, USA

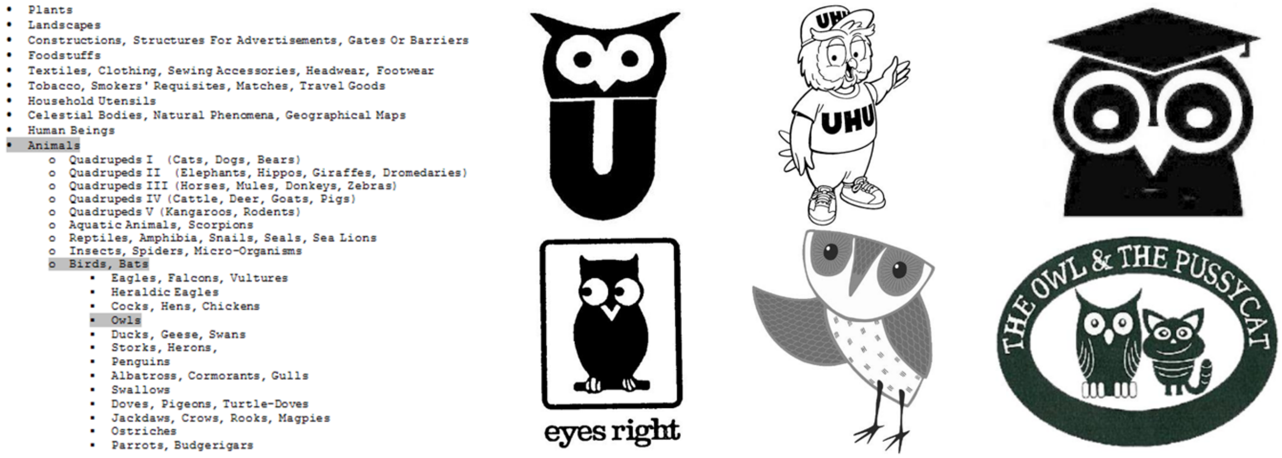


Figure 1: Example of searching logos in the UK Intellectual Property Office website by using the the category tree of Vienna codes.

tion techniques existing in the literature for the retrieval of logos by similarity in large trademark databases.

The graphics recognition community has faced for many years the problem of symbol recognition yielding to good recognition results even with the presence of noise and other distortions. However, as pointed by Tombre and Lamiroy in [20], some challenges remain in this domain. The systems scalability is one of the main concerns. Usually, recognition schemes rely on a preliminary learning stage and a subsequent classification strategy is used to recognize the input graphics. In that scenario, it is usual that the recognition ability of the system is severely impaired as the number of considered model classes grows. In addition, the logo recognition strategies one can find in the literature usually rely on a computationally expensive matching step which hinders their application in large databases.

The main contribution of this paper is to present an efficient queried-by-example retrieval system which is able to retrieve logos by similarity from large databases of isolated trademark images. Logos are compactly described by a variant of the shape context descriptor. These descriptors are then organized by a locality-sensitive hashing indexing structure aiming to perform approximate k-NN search in high dimensional spaces in sub-linear time. The use of a hashing technique allow us to quickly index and retrieve logos by similarity. To conduct the experimental results, we will focus on the large collection of real-world complex document images Tobacco-800 [26, 14], which is public available and has a set of ground-truthed logos with their localization within the complete documents.

The remainder of this paper is organized as follows: we briefly review some related work in section 2. In section 3 the logo description scheme we use is described. Subsequently, in section 4, we present how the logo descriptors are organized in the indexing structure to efficiently retrieve logos by similarity. Section 5 provides the experimental results and finally section 6 is a summary and discussion of extensions and future work.

2. RELATED WORK

The existing literature focusing on the problem of recognizing logos is quite vast within the fields of graphics recognition and shape description at large. One of the first approaches dealing with logo recognition was the one presented by Doermann et al. in [5]. In that work, the authors decomposed the logos in a set of textual and graphical primitives such as lines, circles, rectangles, triangles, etc. and described them by a set of local and global invariants composing a logo signature which was matched against the logo database. Many other works relying on a compact description of the logos and a subsequent matching step can be found. We can cite as examples the work of Eakins et al. [6] where logos are described by a set of features extracted from the shape's boundaries. In [4], Chen et al. represented logos by line segment drawings which were then matched according to a modification of the Hausdorff distance. In [10], Hodge et al. propose a perceptual logo description, whereas in [21] Leuken et al. describe logos by the layout spatial arrangement of basic primitives. Recently, in [25], Zhu and Doermann presented a method for matching logo images based on two different logo descriptions. The shape context descriptor on the one hand, and the neighborhood graphs on the other. The experimental results are encouraging, the matching step, however, is extremely computationally expensive since it involves a point correspondence procedure. This problem is common to all the works described above, which are hardly scalable to large collections of trademark images since they all involve expensive or brute-force matching strategies. We can find in the literature some works that instead of proposing algorithms for logo matching deal with the problem of logo retrieval from image databases.

In [23], Wei et al. present a trademark image retrieval system which combines a local description of logos by means of curvature and spatial information, and a global description of the logos by using the Zernike moments. However, even if the paper is presented as a retrieval application, the logos' descriptors are matched by brute force by computing an Euclidean distance, which makes the system not scalable to large collections unless an indexing strategy is used to efficiently access to the descriptors' data. Another example of trademark retrieval can be found in [17], where Ravela and Manmatha represent logo images by a set of histograms of curvature and frequential information at different scales. Again, the matching step among logo descriptors is critical, and in this case, the authors compute the distances between pairs of logos beforehand.

We propose in this paper a method for efficient logo retrieval in large databases by using an indexing structure aiming to retrieve by similarity the feature vectors describing logo images.

Even if it is not the purpose of this paper, we can also find in the literature an interesting topic related to logo recognition which is the localization and recognition of logos appearing within complete documents without the need of a previous segmentation step. We can for instance cite the works in logo spotting of Zhu and Doermann [24] and Rusiñol and Lladós [18]. Or the ones dealing with trademark detection in video sequences of Bagdanov et al. [1] and Ballan et al. [2].

3. LOGO DESCRIPTION

We base our logo description on a variant of the shape context descriptor known as the shapeme histogram descriptor. Each logo is described by a single histogram representing the frequencies of appearance of a set of specific shape context descriptors. Let us first briefly overview the shape context descriptor, and then focus on how we describe logos by the shapeme histogram descriptor.

3.1 The Shape Context Descriptor in a Nutshell

The shape context descriptor was proposed by Belongie et al. in [3]. This descriptor allows to measure shape similarity by recovering point correspondences between the two shapes under analysis. In a first step, a set of interest points has to be selected from the logos. Usually, a Canny edge detector is used and the edge elements are sampled in order to obtain a fixed number of n points p_i per logo ℓ. Given these n points, the shape context captures the distribution of points within the plane relative to each point of the shape. A histogram using log-polar coordinates counts the number of points inside each bin. For a point p_i of the shape, a histogram h_i of the coordinates of the nearby points q is computed as:

$$h_i(k) = \#\{q \neq p_i : q \in bin_{p_i}(k)\} \quad (1)$$

In our experimental setup, we have chosen 5 bins for $\log r$ and 12 bins for θ. The descriptor offers a compact representation of the distribution of points relative to each selected point. Translational invariance comes naturally to shape context since all the histograms are computed from reference points. Scale invariance is obtained by normalizing all radial distances by the mean distance between all the point pairs in the shape. In order to provide rotation invariance in shape contexts, angles at each point can be measured relative to the direction of the tangent at that point. In our framework, however, we have chosen not to tolerate rotations in order to consider two rotated logos as different instances.

Once all the n points in a shape are described by their shape context histogram, in order to match two shapes we have to find the point correspondences. The simplest way to compute the matching among the two set of points is by using a bipartite graph matching approach that puts in correspondence points having similar shape context descriptions. An example of the shape context descriptor and the results of the point matching by using the Hungarian method can be seen in Fig. 2. In order to obtain a more robust matching, the most usual technique involves the computation the affine transform that matches the set of points from one shape to another.

Shape contexts have empirically demonstrated to be robust to deformations and noise. The shape context descriptor has been tested on different datasets as handwritten digits or silhouette shapes. It has also proven its good performance in recognizing logos. In the original paper [3], the authors use the shape context descriptor to retrieve logo images from a database, and in [25] Zhu and Doermann also used this descriptor for this purpose. However, the fact of having a local description of keypoints which entails a point correspondence matching hinders its applicability to the retrieval problem in large collections.

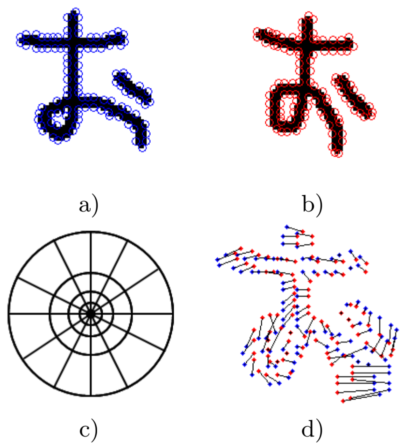
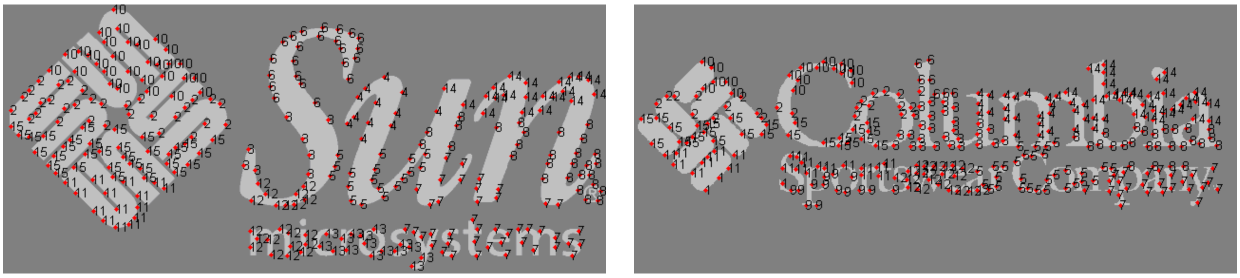


Figure 2: Example of the shape context descriptor for shape matching. (a) and (b) Original shapes to match with sampled edge points; (c) diagram of the log-polar histogram bins used in computing the shape contexts; (d) correspondences found using bipartite matching for the two shapes (a) and (b).

In order to avoid such problem, we use a variant of the shape context descriptor aiming to describe the logos globally, known as the shapeme histogram descriptor. Let us further detail in the next section its use.

3.2 From Local to Global Description: The Shapeme Histogram Descriptor

Realizing that the fact of applying the deformable matching algorithm is computationally prohibitive when dealing with large databases, Mori et al. presented in [15] the shapeme histogram descriptor. This description technique was inspired by the shape context descriptor described above, and the bag-of-words model. The main idea is to compute the shape context descriptor for all the interest points extracted



a) b)

Figure 3: Logo representation by shapemes. Note that the similar parts of the two logos are labelled with the same shapemes (10, 2, 15 and 11). Note as well that the rounded parts of the text are also labelled with the same shapeme (14) in both logos.

from a symbol and then use vector quantization in the space of shape contexts. Vector quantization involves a clustering stage of the shape context feature space. Once the clustering is computed, each shape context descriptor can be identified by the index of the cluster which it belongs to. These clusters are called shapemes. Each logo is then described by a single histogram representing the frequencies of appearance of each shapeme.

In the learning stage, given a set of model logos, we can compute their shape context descriptors and cluster this space by means of the k-means algorithm identifying a set of k cluster centers and assigning to them a given integer index $I \in [1, k]$. Then, in the recognition stage, given a logo ℓ, and its n sampled points from its edge map, we compute their shape context descriptors $h_i, \forall i \in [0, n]$. Each shape context descriptor of the points p_i is then projected to the clustered space and can be identified by a single index I_i. The logo ℓ can thus be represented by a histogram coding the frequency of appearance of each of the k shapeme indices. By this means, we globally describe by a unique histogram SH each logo by applying the following equation:

$$SH(x) = \#\{I_i == x : I_i \in [0, k]\} \qquad (2)$$

We can see an example of the shapeme histogram descriptor in Fig. 3. Here, the shape context space has been clustered by the k-means algorithm with $k = 15$. We can see at each of the points p_i its corresponding index I_i, and how the similar parts shared by both logos are identified by the same indices.

By using the shapeme histogram descriptor, the matching of two logos is reduced to find the k-NN in the space of shapeme histograms, avoiding much more complex matching strategies. However, in order to efficiently retrieve those histograms when dealing with large image datasets, we propose to use an indexing structure which aims to perform an approximate k-NN search in sub-linear time. Let us detail in the next section the use of this structure.

4. APPROXIMATE K-NN SEARCH BY LSH

In order to avoid a one-to-one logo matching, we propose to use an algorithm for approximate k-NN search that efficiently results in a set of candidates that probably lie nearby the queried point. The method we use is the locality-sensitive hashing (LSH), first introduced by Indyk and Motwani in [11], and then revised by Gionis et al. in [9]. The LSH algorithm has been proven to perform approximate k-NN search in sub-linear time and has been used in several applications. As example, Frome and Malik used the LSH algorithm in [8] to index shape context descriptors of three-dimensional objects. Within the DIAR field, Kumar et al. used in [12] the LSH algorithm for a word spotting problem, and in [16], Rasagna et al. use the LSH technique for an efficient clustering of word images.

The basic idea of the LSH method is to index points from a database by using several naïve hash functions, in order to ensure that objects that are close in the feature space have a high probability of provoking collisions in the hash tables.

Let us consider a shapeme histogram description of a logo $SH = (x_1, ..., x_k)$, this point in the k-dimensional space is transformed in a binary vector by using the following equation:

$$v(SH) = (Unary_C(x_1), ..., Unary_C(x_k)) \qquad (3)$$

Being C the largest coordinate of the shapeme histograms space, and $Unary_C(x_i)$ representing the unary representation of x_i, i.e. a sequence of x_i ones followed by $C - x_i$ zeros. The distance between two shapeme histograms can then be computed by the Hamming distance among their binary representations $v(SH)$. We define the hash function family $\mathcal{H}$ describing all possible hash functions $g(x)$ that project binary points to one of their d coordinates as follows:

$$\mathcal{H} = \{g : \{0,1\}^d \rightarrow \{0,1\} | g(x) = x_i, i = 1..d\} \qquad (4)$$

where x_i is the value of the ith coordinate of x. A hash function G is then defined by randomly selecting m hash functions $g(x)$ from $\mathcal{H}$ and concatenating them.

The LSH algorithm then constructs L hash tables $\mathcal{T}$, each corresponding to a different randomly chosen hash function G. Given a query point, the algorithm iterates over all the L hash tables $\mathcal{T}$ retrieving the data points that are hashed into the same bucket as the query.

Let us follow a simple example. Consider three different descriptors:

$$SH_1 = \{5, 3, 2\}$$
$$SH_2 = \{4, 1, 6\}$$
$$SH_3 = \{2, 5, 4\}$$

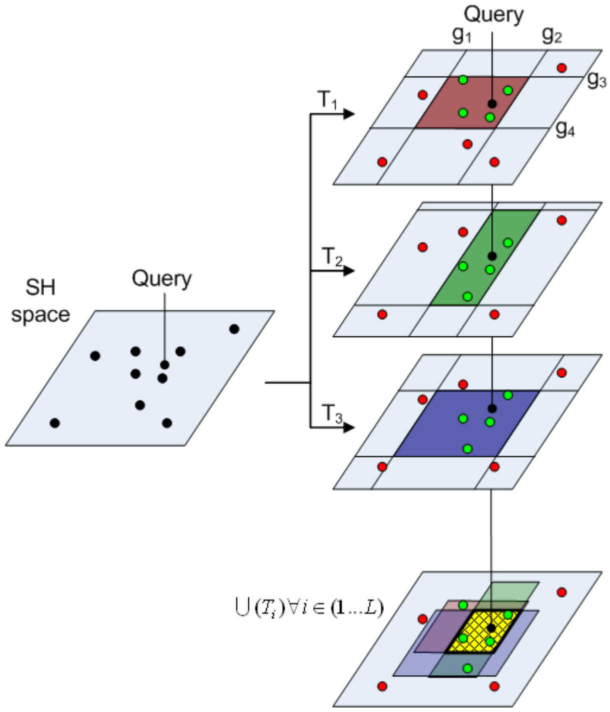


Figure 4: Example of the LSH algorithm. Each hash table $\mathcal{T}$ defines a set of buckets by applying the hash function G. The resulting approximate NN are given by the union of the activated buckets.

in a three-dimensional space where $C = 6$. Their binary representation by applying the unary function is:

$$v(SH_1) = 111110\ 111000\ 110000$$
$$v(SH_2) = 111100\ 100000\ 111111$$
$$v(SH_3) = 110000\ 111110\ 111100$$

We then randomly define three different hash functions:

$$G_1 = \{g_2, g_9, g_{15}\}$$
$$G_2 = \{g_1, g_3, g_{11}, g_{17}\}$$
$$G_3 = \{g_5, g_{14}, g_{16}, g_{18}\}$$

that define the coordinates of the data to take into account. Applying G_1 to a binary vector results in the bucket index identified by the binary number formed by the second, the ninth and the fifteenth value of the original vector. If we apply these hash functions to our data we obtain the following buckets:

$$G_1(SH_1) = 110, \quad G_2(SH_1) = 1100, \quad G_3(SH_1) = 1100$$
$$G_1(SH_2) = 101, \quad G_2(SH_2) = 1101, \quad G_3(SH_2) = 0111$$
$$G_1(SH_3) = 111, \quad G_2(SH_3) = 1010, \quad G_3(SH_3) = 0110$$

Then, given a query $SH_q = \{3, 2, 6\}$ we have:

$$v(SH_q) = 111000\ 110000\ 111111$$
$$G_1(SH_q) = 101, \quad G_2(SH_q) = 1101, \quad G_3(SH_q) = 0111$$

We obtain $SH_2 = \{4, 1, 6\}$ as the nearest descriptor to the query since it provokes a collision in each of the $\mathcal{T}$ tables. If we make the analogy to a two-dimensional space, we can see an illustrative example of the LSH algorithm in Fig. 4. Given our data space and a query, each hash table $\mathcal{T}$ defines a set of buckets by applying a naïve hash function G. The resulting approximate nearest-neighbors are defined by the union of all the data points in each of the buckets identified by the query (green points in the figure). In our experiments setup, we set the width parameter m to 10 bits and we used $L = 50$ tables.

5. EXPERIMENTAL RESULTS

Let us first introduce the logo dataset we used in our experiments and the performance evaluation measures. We will then present the experimental results.

5.1 Dataset and Performance Evaluation

To conduct the experimental results, we will focus on the large collection of real-world complex document images Tobacco-800 [26, 14], which is public available and has a set of ground-truthed logos with their localization within the complete documents. This dataset has labelled 432 logos belonging to 35 different classes. In this case, we consider two logos as similar if they belong to the same class. In order to validate the proposed method, we use a repeated random sub-sampling validation scheme. A 10% of logos are randomly taken as training set to run the k-means clustering and the other 90% are taken as test. The random splitting is repeated ten times and the results are finally averaged for the sake of stability. Note that, usually, in most of the recognition techniques where a learning stage is needed, the training set is much larger than the test set. One of the advantages of the proposed method is that it can perform well with a reduced amount of training data.

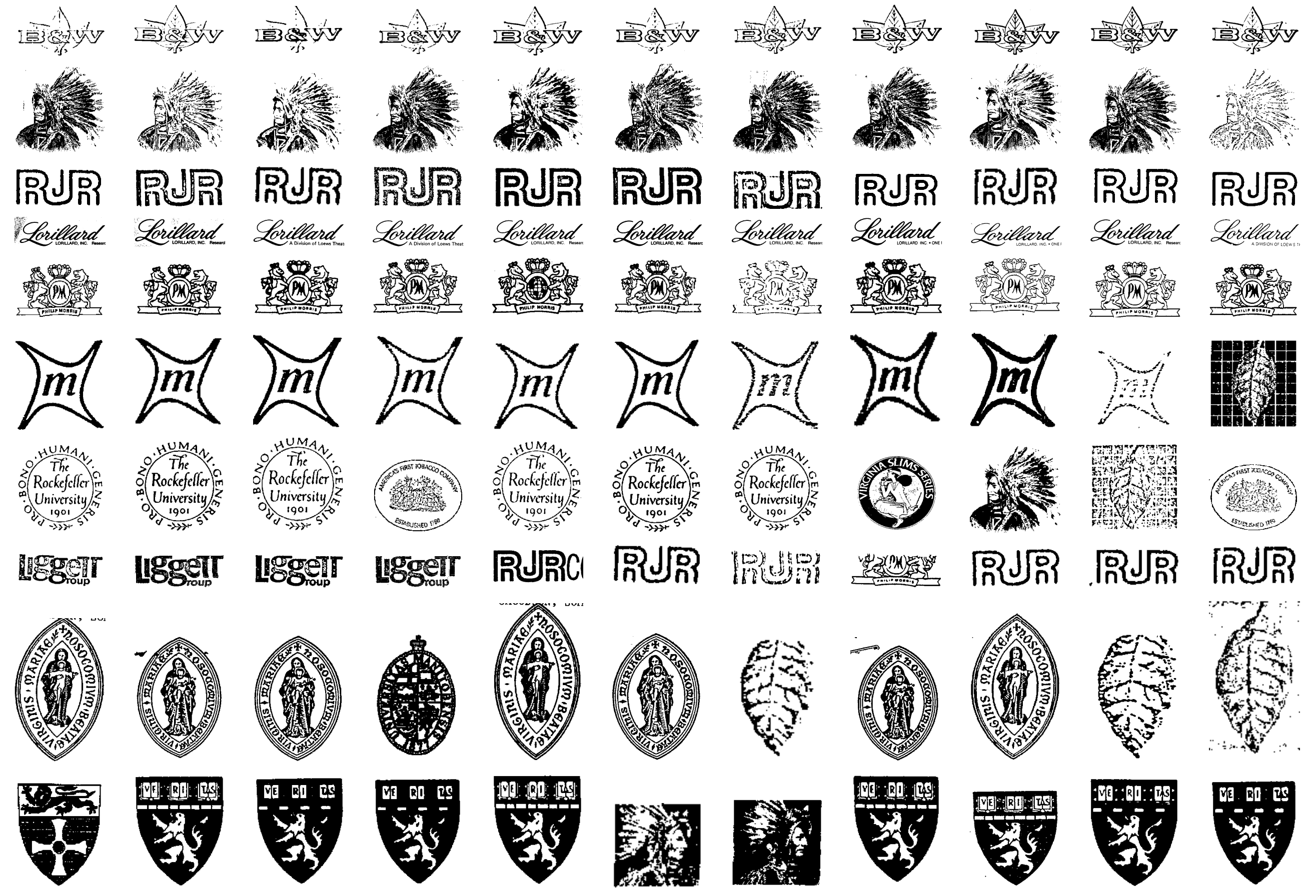
To evaluate the performance of the retrieval system, we will use different measures. First, we will evaluate the retrieval performance by using the ROC curves [7] which plot the false positive rate (FPR) against the true positive rate (TPR). These ratios are derived from the contingency table and defined in terms of the amount of true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN):

$$TPR = \frac{TP}{(TP + FN)} \ ; \quad FPR = \frac{FP}{(FP + TN)} \qquad (5)$$

The TPR ratio measures the effectiveness of the system in retrieving the relevant items. Whereas the FPR ratio measures the probability that a non-relevant document is retrieved by the query. In order to quantitative evaluate the different ROC curves we use the area under curve (AUC) measure. In addition, usually, the user of a retrieval system is mainly interested in retrieving relevant items in the first positions of the results. In order to evaluate the goodness of the retrieval, we also use the mean average precision ($mAveP$) measure. The mean average precision is a measure of quality which rewards the earliest return of relevant items [22]. Retrieving all relevant items in the collection and ranking them perfectly will lead to a mean average precision of 100%. We also use the Bull's eye measure (BE) to evaluate the early retrieval of relevant items. If in the database we have p positive examples for a given query, the Bull's eye measure is computed as the amount of correctly retrieved items in the first $2 \times p$ results [13].

Table 1: Qualitative retrieval results. Query logo and first ten retrieved logos by similarity.

query	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th



All these measures are averaged for all the random splits, and we give the mean $\overline{x}$ and the standard deviation σ for each of them. In order to compare performance of the LSH indexing technique, we will compare the retrieval results with the results of an exhaustive search with a k-NN algorithm by computing the logo similarity between two shapeme histogram descriptors with the χ^2 distance defined as:

$$\chi^2(i,j) = \frac{1}{2}\sum_{m=1}^{k}\frac{[SH_i(m) - SH_j(m)]^2}{SH_i(m) + SH_j(m)} \qquad (6)$$

5.2 Results

Let us first take a look at the qualitative results. We can see in Table 1 some retrieval examples. This experiment was performed with $k = 200$ in one of the random splits. We can appreciate that the retrieval results are quite good since in most cases just correct logos are retrieved in the first 10 ranks. When querying a symbol with few positive samples in the database, the system returns several false alarms which in most cases are visually similar to the query.

Tables 2 and 3 show respectively the average time[2] in retrieving all the logos and the evaluation measures for different values of k. Here we can appreciate a common tradeoff between the time taken to execute a query and the final performance of the system. The more dimensions we add to the shapeme histogram, the more it performs well, however, the more time it takes to answer a query. Fig 5 plots the ROC curves for these different values of k showing the performance gain as long as we increase the dimensions of the shapeme histogram descriptor.

Table 2: Average time in retrieving all the logos for different values of k.

	k				
	25	50	100	200	300
Time (ms.)	20.33	21.25	25.24	26.98	28.04

In Table 3 we compare as well the performance of the presented method with an exhaustive search with a k-NN al-

[2]The times given in Table 2 correspond to a prototype programmed in Matlab without any code optimization, and are given just to compare the effect of the parameter k.

Table 3: **Quantitative evaluation measures. Comparison between LSH and k-NN search.**

k	*AUC* (%)				*mAveP* (%)				*BE* (%)			
	LSH		NN		LSH		NN		LSH		NN	
	$\overline{x}$	σ	$\overline{x}$	σ	$\overline{x}$	σ	$\overline{x}$	σ	$\overline{x}$	σ	$\overline{x}$	σ
25	93.78	0.33	93.87	0.34	71.84	23.79	72.16	23.56	76.64	17.74	83.26	1.62
50	94.21	0.19	94.36	0.15	74.85	23.61	75.46	23.3	76.64	19.57	81.82	12.66
100	94.78	0.26	94.74	0.31	77.59	23.16	77.88	22.74	75.33	23.89	82.12	17.16
200	95.38	0.26	95.05	0.31	80.37	22.43	79.95	22.33	78.91	21.14	85.34	10.99
300	95.77	0.18	95.12	0.19	82.6	22.08	81.71	22.33	85.48	19.74	91.14	3.16

gorithm. We can appreciate that even if LSH perform an approximate k-NN search the performance loss is not significative, and, in some of the cases in the highest dimensional spaces even perform better.

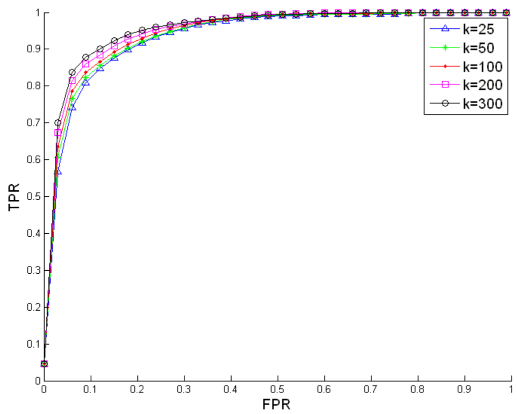


Figure 5: ROC curve for different values of k.

Finally, Fig. 6 we show the results of the last experiment aiming to see the scalability of the proposed method. In this case, besides the logos from the Tobacco-800 database we build a database of 100000 heterogenous objects. We have added architectural and electrical symbols from the GREC databse, logos from the UMD database, silhouettes from the MPEG-7 database, objects from the ALOI database and random generated shapeme histograms. In this experiment we are not interested in the retrieval results but in seeing how scalable the system is regarding the growth of the database. We can appreciate the immense difference between using LSH for accessing large data collections and using a brute-force k-NN search. Although even in the case of using LSH, the consumed time depends as well on how much we increase the collection, the obtained times are still affordable.

6. CONCLUSIONS

In this paper we have presented a method for logo retrieval in large digital libraries. We have proposed an efficient queried-by-example retrieval system which is able to retrieve logos by similarity from large databases of isolated logos. Logos have been compactly described by the shapeme histogram descriptor. These descriptors are then organized by the LSH

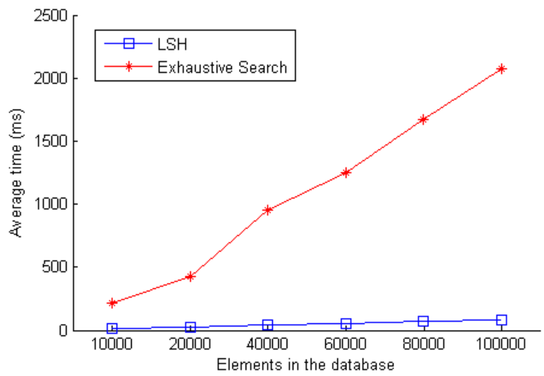


Figure 6: Average time to retrieve the 50-NN for different database sizes.

indexing structure aiming to perform approximate k-NN search in high dimensional spaces in sub-linear time. The experiments demonstrate the effectiveness and efficiency of this system on realistic datasets. We have pointed in this paper the importance of proposing scalable methods in the document image analysis community by means of indexing structures when it comes to the problem of retrieval in digital libraries. The use of LSH aims to achieve retrieval by content similarity in sub-linear time with a negligible accuracy loss regarding the exhaustive search. In addition, the proposed method is able to perform well with a low amount of training samples which is also an important point when dealing with heterogenous image collections as logos.

One of the points to take into account is that the shapeme histogram descriptor does not involve any spatial information among the interest points. It would be an interesting idea to try to add relational information to the proposed description technique in order to make the method much more robust. Regarding the future work, we believe that the retrieval of logos present in complete documents is still a promising research field. In this paper we just focused in isolated logos, but it would be very interesting to extend the presented techniques to tackle with complete documents for focused retrieval of logos in documents. Logo spotting will be useful for a great number of applications such as document classification, indexation and browsing. The use of local description techniques such as shape contexts and efficient indexing structures as LSH should be further investigated in relation with logo spotting and focused retrieval.

7. ACKNOWLEDGMENTS

This work has been partially supported by the Spanish projects TIN2006-15694-C02-02, TIN2009-14633-C03-03 and CONSOLIDER - INGENIO 2010 (CSD2007-00018). We would also thank Dr. R. Manmatha to give us advice during ICDAR09 regarding the LSH technique.

8. REFERENCES

[1] A. Bagdanov, L. Ballan, M. Bertini, and A. D. Bimbo. Trademark matching and retrieval in sports video databases. In *Proceedings of the International Workshop on Multimedia Information Retrieval*, pages 79–86. ACM, 2007.

[2] L. Ballan, M. Bertini, and A. Jain. A system for automatic detection and recognition of advertising trademarks in sports videos. In *Proceeding of the Sixteenth ACM International Conference on Multimedia*, pages 991–992. ACM, 2008.

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

[4] J. Chen, M. Leung, and Y. Gao. Noisy logo recognition using line segment Hausdorff distance. *Pattern Recognition*, 36(4):943–955, 2003.

[5] D. Doermann, E. Rivlin, and I. Weiss. Applying algebraic and differential invariants for logo recognition. *Machine Vision and Applications*, 9(2):73–86, 1996.

[6] J. Eakins, K. Shields, and J. Boardman. ARTISAN - A shape retrieval system based on boundary family indexing. In *Proceedings of SPIE: Storage and Retrieval for Image and Video Databases*, pages 17–28, 1996.

[7] T. Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

[8] A. Frome and J. Malik. Object recognition using locality sensitive hashing of shape contexts. In G. Shakhnarovich, T. Darrell, and P. Indyk, editors, *Nearest-Neighbor Methods in Learning and Vision*. MIT Press, 2005.

[9] A. Gionis, P. Indyk, and R. Motwani. Similarity search in high dimensions via hashing. In *Proceedings of the Twentyfifth International Conference on Very Large Data Bases*, pages 518–529. Morgan Kaufmann, 1999.

[10] V. Hodge, G. Hollier, J. Austin, and J. Eakins. Identifying perceptual structures in trademark images. In *Proceedings Fifth IASTED International Conference on Signal Processing, Pattern Recognition, and Applications*. ACTA Press, 2008.

[11] P. Indyk and R. Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, pages 604–613. ACM, 1998.

[12] A. Kumar, C. Jawahar, and R. Manmatha. Efficient search in document image collections. In *Computer Vision, ACCV*, volume 4843 of *Lecture Notes on Computer Science*, pages 586–595. Springer-Verlag, 2007.

[13] L. Latecki, R. Lakämper, and T. Eckhardt. Shape descriptors for non-rigid shapes with a single closed contour. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 424–429. IEEE Computer Society, 2000.

[14] D. Lewis, G. Agam, S. Argamon, O. Frieder, D. Grossman, and J.Heard. Building a test collection for complex document information processing. In *Proceedings of the Twentyninth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 665–666. ACM, 2006.

[15] G. Mori, S. Belongie, and J. Malik. Efficient shape matching using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(11):1832–1837, 2005.

[16] V. Rasagna, A. Kumar, C. Jawahar, and R. Manmatha. Robust recognition of documents by fusing results of word clusters. In *Proceedings of the Tenth International Conference on Document Analysis and Recognition*, pages 566–570. IEEE Computer Society, 2009.

[17] S. Ravela and R. Manmatha. Multi-modal retrieval of trademark images using global similarity. Technical Report Technical Report: UM-CS-1999-032, University of Massachusetts, 1999.

[18] M. Rusiñol and J. Lladós. Logo spotting by a bag-of-words approach for document categorization. In *Proceedings of the Tenth International Conference on Document Analysis and Recognition*, pages 111–115. IEEE Computer Society, 2009.

[19] J. Schietse, J. Eakins, and R. Veltkamp. Practice and challenges in trademark image retrieval. In *Proceedings of the Sixth ACM International Conference on Image and Video Retrieval*, pages 518–524. ACM, 2007.

[20] K. Tombre and B. Lamiroy. Graphics recognition - from re-engineering to retrieval. In *Proceedings of the Seventh International Conference on Document Analysis and Recognition*, pages 148–155. IEEE Computer Society, 2003.

[21] R. van Leuken, O. Symonova, and R. Veltkamp. Topological and directional logo layout indexing using hermitian spectra. In *Proceedings Fifth IASTED International Conference on Signal Processing, Pattern Recognition, and Applications*. ACTA Press, 2008.

[22] C. van Rijsbergen. *Information retrieval*. Butterworth-Heinemann Newton, MA, USA, 1979.

[23] C. Wei, Y. Li, W. Chau, and C. Li. Trademark image retrieval using synthetic features for describing global shape and interior structure. *Pattern Recognition*, 42(3):386–394, 2009.

[24] G. Zhu and D. Doerman. Automatic document logo detection. In *Proceedings of the Ninth International Conference on Document Analysis and Recognition*, pages 864–868. IEEE Computer Society, 2007.

[25] G. Zhu and D. Doerman. Logo matching for document image retrieval. In *Proceedings of the Tenth International Conference on Document Analysis and Recognition*, pages 606–610. IEEE Computer Society, 2009.

[26] G. Zhu and D. Doermann. Tobacco-800 complex document image database and groundtruth. Online, 2008.