Shape Retrieval by Inexact Graph Matching

Benoit Huet, Andrew D.J. Cross and Edwin R. Hancock
Department of Computer Science
University of York, York, Y01 5DD, UK
huetb|erh@cs.york.ac.uk

abstract

This paper describes a graph-matching technique for recognising line-pattern shapes in large image databases. The methodological contribution of the paper is to develop a Bayesian matching algorithm that uses edge-consistency and node attribute similarity. This information is used to determine the a posteriori probability of a query graph for each of the candidate matches in the database. The node feature-vectors are constructed by computing normalised histograms of pairwise geometric attributes. Attribute similarity is assessed by computing the Bhattacharyya distance between the histograms. Recognition is realised by selecting the candidate from the database which has the largest a posteriori probability. We illustrate the recognition technique on a database containing 2500 line patterns extracted from real-world imagery. Here the recognition technique is shown to significantly outperform a number of algorithm alternatives.

1 Introduction

The efficient and accurate retrieval of images on the basis of content is a key enabling technology for multi-media systems. Broadly speaking there are two contrasting approaches to the problem. The first of these is to compute a compact summary of the image attributes. One of the best known examples here is the attribute histogram originally popularised by Swain and Ballard [8] for retrieving colour images from data-bases. The idea has been extended to texture [3], local feature orientation representations [1, 6] and the geometry of line-patterns [2]. The alternative to the use of attributes is to retrieve images based on their structural information content [7]. Here the basic idea is to recognise relational object descriptions by comparing graph-structure.

Although both structural indexing and attribute histograms have been separately used to recall image data from databases, they have not been used in conjunction. Here we aim to fill this gap in the literature by using graph matching as a means of retrieving the shape from a large data-base that most closely resembles a query shape. Although the indexation images in large data-bases is a problem of current topicality in the computer vision literature [8, 3], the work presented in this paper is more ambitious. Firstly, we adopt a structural abstraction of the shape recognition problem and match using attributed relational graphs. Each shape in our database is a pattern of line-segments. The structural abstraction is a nearest neighbour graph for the centre-points of the line-segments. In addition, we exploit attribute information for the line patterns. Here the geometric arrangement of the line-segments is encapsulated using a histogram of Euclidean invariant pairwise (binary) attributes. For each line-segment in turn we construct a normalised histogram of relative angle and length with the remaining line-segments in the pattern. These histograms capture the global geometric context of each line-segment. Moreover, we interpret the pairwise geometric histograms as measurement densities for the line-segments which we compare using the Bhattacharyya distance.

Once we have established the pattern representation, we realise object recognition using a Bayesian graph-matching algorithm. This is a two-step process. Firstly, we establish correspondence matches between the individual tokens in the query pattern and each of the patterns in the database. Since this is a computationally intensive process, we first filter the set of potential patterns using a rapid histogram-based recognition technique [5]. The correspondence matches are sought so as to maximise the *a posteriori* measurement probability. Here we use an extension of the graph-matching technique recently reported by Wilson and Hancock [10] in which we use the correspondence matches residing on the edges rather than the nodes of our ARG's to asses consistency. Once the MAP correspondence matches have been established, then the second step in our recognition architecture involves selecting the line-pattern from the database which has maximum matching probability.

2 Framework

Formally our recognition problem is posed as follows. We abstract the shapes to be recognised as attributed relational graphs (ARG's). Each ARG in the database is a triple, $G = (V_G, E_G, A_G)$, where V_G is the set of vertices (nodes), E_G is the edge set ($E_G \subset V_G \times V_G$), and A_G is the set of node attributes. In our experimental example, the nodes represent line-structures segmented from 2D images. The edges are established by computing the N-nearest neighbour graph for the line-centres. Each node $j \in V$ is characterised by a vector of attributes, $\underline{x}_j$ and hence $A_G = \{\underline{x}_j | j \in V\}$. In the work reported here the attribute-vector represents the contents of a normalised pairwise attribute histogram.

The data-base of line-patterns is represented by the set of ARG's $\mathcal{D} = \{G\}$. The goal is to retrieve from the data-base $\mathcal{D}$, the individual ARG that most closely resembles a query pattern $Q = (V_Q, E_Q, A_Q)$.

We now furnish some details of the shape retrieval task used in our experimental evaluation of the recognition method. In particular, we focus on the problem of recognising 2D line patterns in a manner, which is invariant to rotation, translation and scale. The raw information available for each line segment are its orientation (angle with respect to the horizontal axis) and its length (see figure 1). To illustrate how the Euclidean invariant pairwise feature attributes are computed, suppose that we denote the line segments associated with the nodes indexed a and b by the vectors $\underline{v}_a$ and $\underline{v}_b$ respectively. We use two pairwise attributes. The first these is the relative angle given by

$$\theta_{a,b} = \arccos\left[\frac{\underline{v}_a \cdot \underline{v}_b}{|\underline{v}_a||\underline{v}_b|}\right]$$

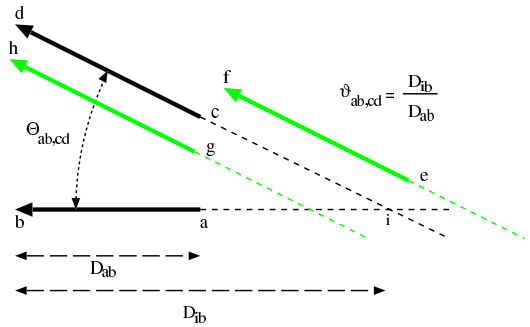


Figure 1: Geometry for shape representation

The second attribute is the normalised length ratio between the oriented baseline vector $\underline{v}_a$ and the vector $\underline{v}'$ joining the end (b) of the baseline segment (ab) to the intersection of the segment pair (cd).

$$\vartheta_{a,b} = \frac{1}{\frac{1}{2} + \frac{D_{ib}}{D_{ab}}}$$

The two attibutes are used as a feature-vector $z_{a,b} = (\theta_{a,b}, \vartheta_{a,b})^T$ for the line-segment pair.

Each node in the shape graph, i.e. each line in the pattern, is represented by the histogram of its pairwise geometric attributes to the remaining lines in the pattern. This histogram can be thought of as a local estimate of the probability distribution for the pairwise attributes. Accordingly, the angle and position attributes $\theta_{a,b}$ and $\vartheta_{a,b}$ are binned in a histogram. Suppose that $S_a(\mu,\nu) = \{(a,b)|\theta_{a,b} \in A_\mu \wedge \vartheta_{a,b} \in R_\nu \wedge b \in V_D\}$ is the set of nodes whose pairwise geometric attributes with the node a are spanned by the range of directed relative angles A_μ and the relative position attribute range R_ν. The contents of the histogram bin spanning the two attribute ranges is given by $H_a(\mu,\nu) = |S_a(\mu,\nu)|$. Each histogram contains n_A relative angle bins and n_R length ratio bins. The normalised geometric histogram bin-entries are computed as follows

$$h_a(\mu,\nu) = \frac{H_a(\mu,\nu)}{\sum_{\mu'=1}^{n_A}\sum_{\nu'=1}^{n_R} H_a(\mu,\nu)}$$

In the remainder of this paper, we use the notation h^G to denote the normalised histogram for the graph G from the database and h^Q to denote the query histogram.

In order to exploit the consistency of correspondences, we pose the retrieval process as one of associating with the query the graph from the data-base that has the largest *a posteriori* probability of match. In other words, the class identity of the graph which most closely corresponds to the query is

$$\omega_Q = \arg\max_{G' \in \mathcal{D}} P(G'|Q)$$

However, since we wish to make a detailed structural comparison of the graphs, rather than comparing their overall statistical properties, we must first establish a set of best-match correspondences between each ARG in the database and the query Q. At iteration n the set of correspondences between the query Q and the ARG G is a relation $f_G^n : V_G \mapsto V_Q$ over the vertex sets of the two graphs. The mapping function consists of a set of Cartesian pairings between the nodes of the two graphs, i.e. $f_G^n = \{(a,\alpha); a \in V_G, \alpha \in V_Q\} \subseteq V_G \times V_Q$. The retrieved pattern is the one which has the most

consistent pattern of correspondences and satisfies the condition

$$\omega_Q = \arg\max_{G' \in \mathcal{D}} \max_{f_{G'}} P(f_{G'}|G', Q)$$

The pattern of correspondences is assigned to satisfy the following *maximum a posteriori* probability condition

$$f_G^n(a) = \arg\max_{\alpha \in V_Q} p(\underline{x}_a, \underline{x}_\alpha | f_G^n(a) = \alpha) P(f_G^n | E_G, E_Q)$$

Suppose that we use the notation

$$s_{a,\alpha}^n = \begin{cases} 1 & \text{if } f_G^n(a) = \alpha \\ 0 & \text{otherwise} \end{cases}$$

to represent the correspondence assignment. The consistency of global match against the query pattern can be improved by iterating the assignment condition

$$f_G^n(a) = \arg\max_{\alpha \in V_Q} \left[\ln p(\underline{x}_a, \underline{x}_\alpha | f_G^n(a) = \alpha) + \sum_{(a,b) \in E_G} \sum_{(\alpha,\beta) \in E_Q} \left\{ \ln(1 - P_e) s_{a,\alpha}^{n-1} s_{b,\beta}^{n-1} + \ln P_e (1 - s_{a,\alpha}^{n-1} s_{b,\beta}^{n-1}) \right\} \right]$$

The probability of match between the pattern-vectors is computed using the Bhattacharyya coefficient $B_{a,\alpha}$ between the normalised histograms.

$$P(f_G^n(a) = \alpha | \underline{x}_a, \underline{x}_\alpha) = \exp[-B_{a,\alpha}] = \frac{\sum_{\mu=1}^{n_A} \sum_{\nu=1}^{n_R} \sqrt{h_a(\mu,\nu) h_\alpha(\mu,\nu)}}{\sum_{\alpha \in V_Q} \sum_{\mu'=1}^{n_A} \sum_{\nu'=1}^{n_R} \sqrt{h_a(\mu,\nu) h_\alpha(\mu,\nu)}}$$

With this modelling ingredient, and using the correspondence matches delivered by the graph-matching scheme, the condition for recognition is

$$\omega_Q = \arg\max_{G' \in \mathcal{D}} \sum_{(a,b) \in E'_G} \sum_{(\alpha,\beta) \in E_Q} \left\{ -B_{a,\alpha} - B_{b,\beta} + \ln(1 - P_e) s_{a,\alpha}^n s_{b,\beta}^n + \ln P_e (1 - s_{a,\alpha}^n s_{b,\beta}^n) \right\}$$

3 Recognition Experiments

The aim in this section is to evaluate the graph-based recognition scheme on a database of real-world line-patterns. We have conducted our recognition experiments with a database of 2500 line-patterns each containing over a hundred lines. The line-patterns have been obtained by applying line and edge detection algorithms to the raw grey-scale images followed by polygonisation to identify straight-line segments. For each line-pattern in the database, we construct the adjacency graph of the line centre-points; the choice of structure is one of the issues discussed later on in this section. The feature extraction process together with other details of the data used in our study are described in recent papers where we have focussed on the issues of histogram representation [5] and the optimal choice of the relational structure [4] for the purposes of recognition.

In order to prune the set of line-patterns for detailed graph-matching we select about 10% of the database using a two-step process. This consists of first refining the data-base using a global histogram of pairwise attributes [5]. The top quartile of matches selected in this way are then further refined using a variant of the Haussdorff distance to select the set of pairwise attributes that best match against the query [4].

The recognition task is posed as one of recovering the line-pattern which most closely resembles a digital map. The original images from which our line-patterns have been obtained come from a number of diverse sources. However, a subset of the images are aerial infra-red line-scan views of southern England. Two of these infra-red images correspond to different views of the area covered by the digital map. These views are obtained when the line-scan device is flying at different altitudes. The line-scan device used to obtain the aerial images introduces severe barrel distortions and hence the map and aerial images are not simply related via a Euclidean or affine transformation. The remaining line-patterns in the database have been extracted from trademarks and logos. It is important to stress that although the raw images are obtained from different sources, there is nothing salient about their associated line-pattern representations that allows us to distinguish them from one-another. Moreover, since it is derived from a digital map rather than one of the images in the database, the query is not identical to any of the line-patterns in the model library.

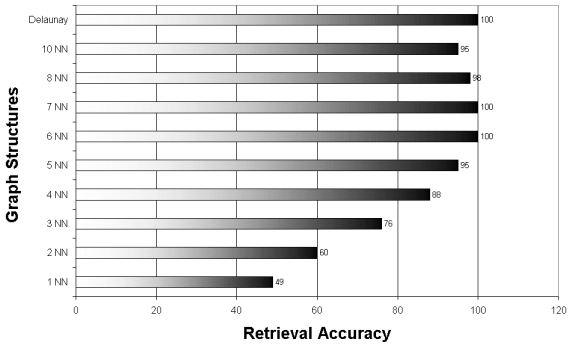


Figure 2: Effect of the choice of structural representation on retrieval accuracy

We commence by examining the role of structural representation. Here we investigate the effect of the order of the N-nearest neighbour graph. We also compare the N-nearest neighbour graph with the Delaunay graph. In figure 2 we show a plot of the recognition accuracy of a database of 260 line-patterns. The main effects to note are that the optimal order of the nearest neighbour graph is 6. Although the recent study of Tuceryan and Chorzempa [9] indicates that the Delaunay graph offers greater robustness to noise and clutter, in the experiments reported here the performance advantage is only marginal.

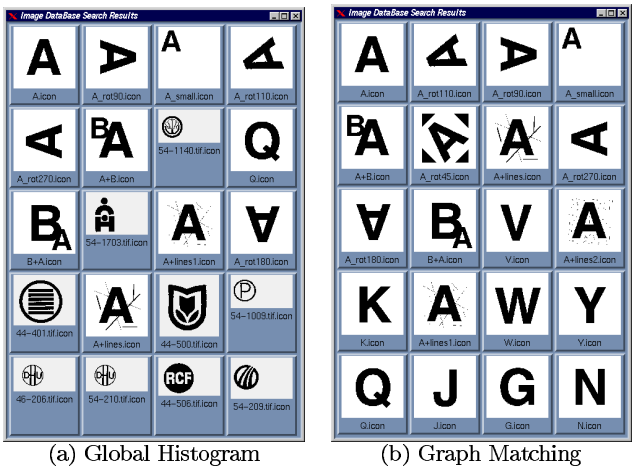
In order to illustrate the advantages of using a graph-matching technique, we have compared the results with those obtained using a conventional histogram retrieval method [5]. This retrieval method uses a global histogram of pairwise attributes. Using the notation developed in section 2, for the graph indexed G, the histogram bin-contents is computed by summing the bin-contents for the individual node histograms, i.e.

$$\gamma_G(\mu,\nu) = \sum_{a \in V_G} H_a(\mu,\nu)$$

Recognition is effected by recalling the line-pattern which has the histogram of minimum Bhattacharyya distance to the query. In other words, the class identity of the retrieved pattern is

$$\omega_Q = \arg\max_{G \in \mathcal{D}} \sum_{\mu=1}^{n_A} \sum_{\nu=1}^{n_R} \sqrt{\Gamma_G(\mu,\nu)\Gamma_Q(\mu,\nu)}$$

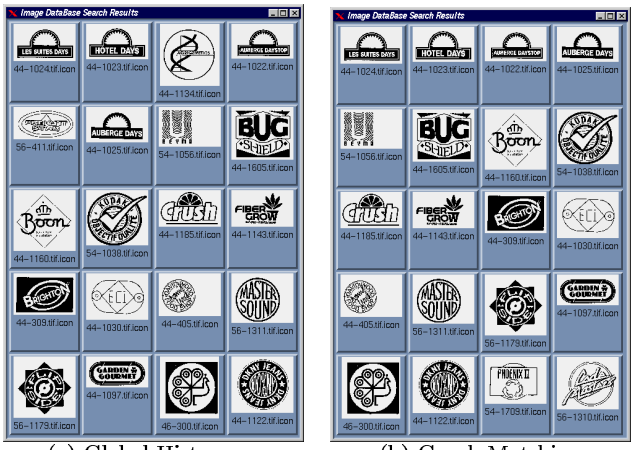
The results of our comparative study are presented in the form of panels of "thumbnail" images from the database used in our recognition experiments. The "thumbnails" are ordered (from left-to-right and top-to-bottom) according their distance from the query image. The first experiment involves querying the database with a letter "A". In this example the database is augmented with 11 images of the letter "A" subjected to various rotations and scalings. In some of these images the letter "A" appears alongside another letter or has random clutter lines superimposed. In Figure 3, we compare the recognition results obtained with the graph-matching and global histogram comparison. In the case of the graph-matching, the first error occurs at rank position 11. Of the 12 A's in the database the worst recall appears at rank position 14. By contrast, in the case of the global histogram, the first error is at rank position 7 and the worst recall is at rank position 47. In the second example, we experiment with the "Le



(a) Global Histogram (b) Graph Matching

Figure 3: *The result of querying the data-base with the letter "A": The images are ordered from left-to-right and top-to-bottom in increasing distance from the query image, which is shown at the top of the figure.*

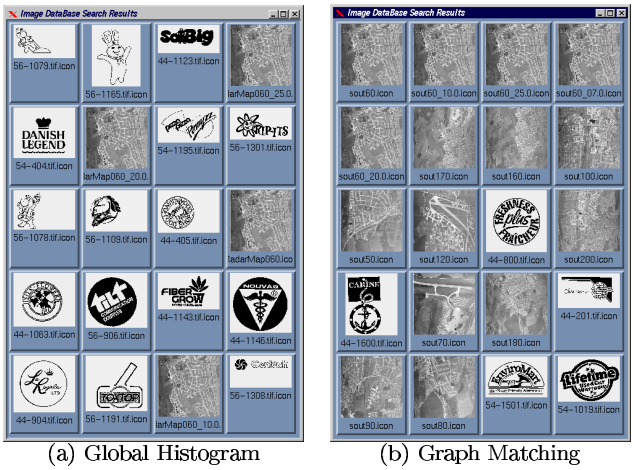
Suites Days" logo. The results are shown in figure 4. There are four logos in the database that have the same shape, but which carry a different legend (Hotel Days, Auberge Daystop and Auberge Days). In the case of both the graph-matching and global histogram comparison, the correct logo is correctly recognised in the top ranked position. However, in the case of graph-matching the three logos with the same shape but different legends are ranked from second to fourth. When the global histogram is used, three logos are more dispersed and appear at positions 2, 4 and 6. The third example draws on the cartographic section of the database. The line-patterns contained in the cartographic section of the database are segmented from infra-red aerial images of urban and semi-urban areas. The *Digital Map* (or cartographic model) used to query the database represents a road network known to appear in a subset of the images contained within the database. Two different images contain the road network area depicted in the cartographic model, but are acquired when the imaging apparatus is making different altitude passes over the scene. However, there is known to be a significant barrel distortion in the horizontal direction of the infra-red images caused by the rotating mirror optics underlying the line-scan process used during sensing. Additionally, the viewing angle, altitude and direction are not similar to the *Digital Map*. Ideally the result of a database query using the *Digital Map* is a retrieval where both the target images are recalled as the first and second best matches. Since we also aim to investigate the sensitivity of recall to the line segmentation process, we have



(a) Global Histogram (b) Graph Matching

Figure 4: *The result of querying the data-base with the "Le Suites Days" logo*

augmented the database with four additional sets of over and under segmented line-segments for the target infra-red image. The ranked "thumbnail" images



(a) Global Histogram (b) Graph Matching

Figure 5: *The result of querying the data-base with the digital map*

are compared for the graph-matching method and the global histogram in Figure 5. In the case of the relational histograms, the five different segmentations of the low-altitude target image are top-ranked. The result of controlled over and under segmentation of the image results in significant variation in the number of segments used to construct the histogram representation. The retrieval is not specifically affected by these segmentation systematics as can be seen in figure 5. The high altitude image of the same area appears at rank 6.

In the case of the global histogram the different segmentations of the target image are badly dispersed. Four of the five different segmentations appear at rank positions 4, 6, 12 and 19. In other words, the recognition performance is very poor.

4 Conclusions

We have presented a practical graph-matching algorithm for data-mining in large structural libraries. The main conclusion to be drawn from this study is that the combined use of structural and histogram information improves both recognition performance and recall speed. There are a number of ways in which the ideas presented in this paper can be extended. Firstly, we intend to explore more a perceptually meaningful representation of the line patterns, using grouping principal derived from Gestalt psychology. Secondly, we are exploring the possibility of formulating the filtering of line-patterns prior to graph matching using Bayes decision trees.

References

[1] C. Dorai and A. K. Jain. View organisation and matching of free-form objects. *IEEE ISCV*, pages 25–30, 1995.

[2] A. C. Evans, N. A. Thacker, and J. W. E. Mayhew. The use of geometric histograms for model-based object recognition. *BMVC*, pages 429–438, Sept 1993.

[3] G. L. Gimelfarb and A. K. Jain. On retrieving textured images from an image database. *Pattern Recognition*, 29(9):1461–1483, 1996.

[4] B. Huet and E. R. Hancock. Fuzzy relational distance for large-scale object recognition. *IEEE CVPR*, pages 138–143, June 1998.

[5] B. Huet and E. R. Hancock. Relational histograms for shape indexing. *ICCV*, pages 563–569, Jan 1998.

[6] I. Rigoutsos and R. Hummel. A bayesian approach to model matching with geometric hashing. *CVIU*, 62(1):11, 26 1995.

[7] K. Sengupta and K. L. Boyer. Organising large structural databases. *IEEE PAMI*, 17(4):321–332, 1995.

[8] M. J. Swain and D. H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[9] M. Tuceryan and T. Chorzempa. Relative sensitivity of a family of closest-point graphs in computer vision applications. *Pattern Recognition*, 24(5):361–373, 1991.

[10] R. Wilson and E. R. Hancock. Structural matching by discrete relaxation. *IEEE PAMI*, 19(6):634–648, June 1997.