Generic Shape Classification for Retrieval

Manuel J. Fonseca, Alfredo Ferreira, and Joaquim A. Jorge

Department of Information Systems and Computer Science
INESC-ID/IST/Technical University of Lisbon
R. Alves Redol, 9, 1000-029 Lisboa, Portugal
mjf@inesc-id.pt, alfredo.ferreira@inesc-id.pt, jorgej@acm.org

Abstract. We present a shape classification technique for structural content–based retrieval of two-dimensional vector drawings. Our method has two distinguishing features. For one, it relies on explicit hierarchical descriptions of drawing structure by means of spatial relationships and shape characterization. However, unlike other approaches which attempt rigid shape classification, our method relies on estimating the likeness of a given shape to a restricted set of simple forms. It yields for a given shape, a feature vector describing its geometric properties, which is invariant to scale, rotation and translation. This provides the advantage of being able to characterize arbitrary two–dimensional shapes with few restrictions. Moreover, our technique seemingly works well when compared to established methods for two dimensional shapes.

1 Introduction

Since shape is one of the primary low level features used in content-based image retrieval, shape representation has become a fundamental issue in these applications. The main objective of shape description is to measure geometric attributes of an object, that can be used for classifying, matching and recognizing objects. Moreover, a shape representation scheme should be affine invariant, robust, compact, easy to derive, easy to match and perceptually meaningful. Also, it is important that shape description schemes work well for practical applications. We have thus validated our work with two "real–life" applications, one to retrieve technical drawings and other to search for clip-art drawings. In this paper, after a short discussion of related work, we will briefly present our approach to drawing classification, consisting of topological and geometrical components. Then we focus on geometry extraction and describe our technique for shape classification. Next, we discuss experimental results obtained by comparing our method to five known techniques and briefly describe the two prototypes that use our method. Finally we present conclusions and future work.

2 Related Work

There is an extensive body of related work on shape representation. Mehtre et al group existing techniques into two categories: boundary-based and region-based

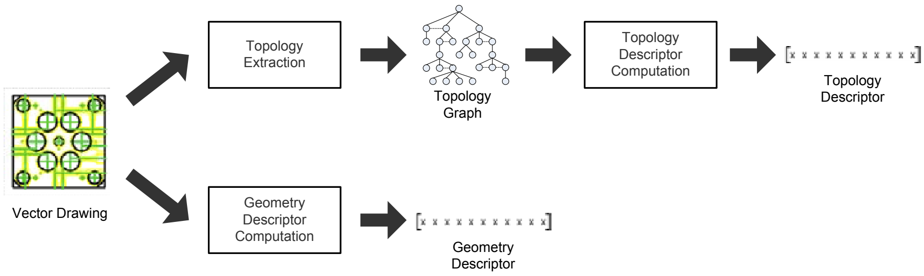


Fig. 1. Block decomposition of our approach to drawing classification.

[17]. The former use only the contour or border of an object, which is crucial to human perception in judging shape similarity, completely ignoring its interior. The latter methods exploit shape interior information, besides its boundary. More recently Safar et al presented a taxonomy [22] that complements Mehtre's classification.

As examples of boundary-based methods we have Fourier descriptors [20], chain codes [11], autoregressive models [15], polygonal approximations [13], curvature scale space [18] and shape signature [3]. In region-based methods, we encountered geometric moments [14], Zernike moments [17], grid representation [16] and area.

Although contour-based methods such as Fourier descriptors, present good results in these studies, they have limited application. For one, these methods cannot capture shape interior content or deal with disjoint shapes, where single boundaries may not be available. Also, region-based methods can be applied to more general shapes, but usually require more computational resources.

3 Drawing Classification

Content-based retrieval of pictorial data, such as digital images, drawings or graphics, uses features extracted from the corresponding picture. Typically, two kinds of features are used; visual features (such as color, texture and shape) and relationship features (topological and spatial relationships among objects in a picture). However, in the context of our work, we consider that color and texture are irrelevant features and we focus only on topology (a global feature of drawings) and geometry (a local feature).

Our feature extraction technique processes drawings via two separate stages (topology and shape) until they are mapped into geometric and topological descriptors, as depicted in Figure 1. For retrieval purposes, these descriptors may be inserted in an indexing structure, during classification, or used to query a database, when searching for similar drawings.

To describe the spatial organization in drawings, we use two relationships, inclusion and adjacency. While these two topological relationships are weakly

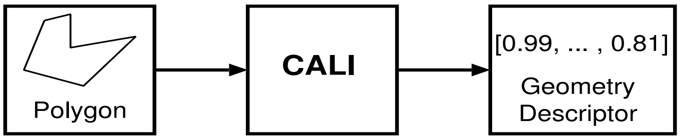


Fig. 2. Block diagram for computing the geometric descriptor.

discriminating, they do not change with rotation and translation, allowing unconstrained drawing classification. We then construct a topology graph representing the relationships among shapes. From this graph, we derive descriptors based on its spectrum [2]. We compute the graph spectrum by determining the eigenvalues of its adjacency matrix. Eigenvalues are then stored in a multidimensional vector, defining the topological descriptor. A detailed description of the topology extraction and the correspondent descriptor computation using eigenvalues can be found in [4].

4 Geometry Extraction

To describe the geometry of entities from drawings, we developed a general, simple, fast, and robust recognition approach called CALI [8, 7]. This was initially devised for recognition in calligraphic interfaces. However, since CALI performed well in recognizing hand-drawn input, we decided to generalize that approach by using it to classify more general shapes for retrieval. Thus, instead of using CALI to identify specific shapes or gestures from sketches, we compute a set of geometric attributes from which we derive features such as area and perimeter ratios from special polygons and store them in a multidimensional vector (see Figure 2). Indeed, our approach can be thought as a two–stage process. First, we evaluate a shape's geometric characteristics. Then we convert these into affine–invariant geometric features by simple arithmetic operations which

Feature	***Description***
A_{ch}	Area of the convex hull
A_{er}	Area of the (non-aligned) enclosing rectangle
A_{lq}	Area of the largest quadrilateral
A_{lt}	Area of the largest triangle
H_{er}	Height of the (non-aligned) enclosing rectangle
P_{ch}	Perimeter of the convex hull
P_{er}	Perimeter of the enclosing rectangle
P_{lq}	Perimeter of the largest quadrilateral
P_{lt}	Perimeter of the largest triangle
T_l	Total length, *i.e.* perimeter of original polygon
W_{er}	Width of the (non-aligned) enclosing rectangle

Table 1. List of relevant geometrical features.

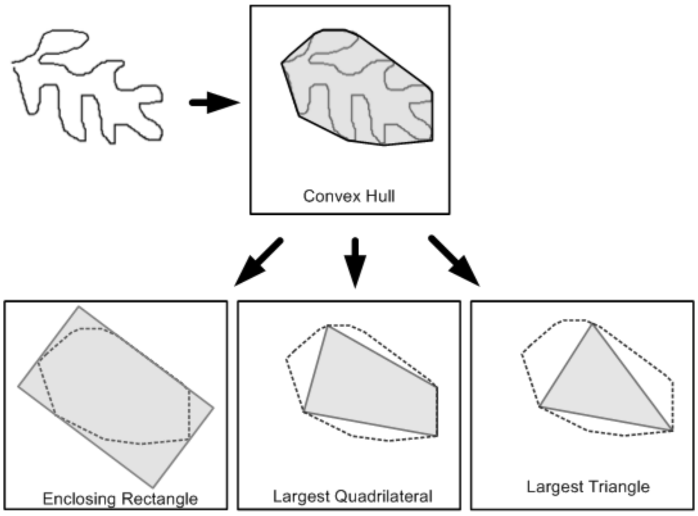


Fig. 3. Special polygons computed from shape.

combine these attributes with known commensurable values for simple convex primitives, such as quadrilaterals and triangles. What is more important, using geometric features instead of polygon classification, allows us to index and store potentially unlimited families of shapes in a scalable manner.

Our geometric description method uses a set of global geometric properties extracted from drawing entities. We start the calculation of geometric features by computing the *Convex Hull* of the provided element, using Graham's scan [19]. Then, we compute three special polygons from the convex hull: the *Largest Area Triangle* and the *Largest Area Quadrilateral* inscribed in the convex hull [1], and finally, the *Smallest Area Enclosing Rectangle* [12]. Figure 3 depicts an example of polygons extracted from a irregular shape.

Finally, we compute the ratios between area and perimeter from each special polygon. We experimentally evaluated several ratios, as described in detail in [9], before we reach the set of features listed in Table 1. This set of features allow the description of shapes independently of their size, rotation, translation or line type. This way, such features can either be used to classify drawings or hand-sketched queries. Then, we combine these geometric features to produce a feature vector that describes the shape (descriptor).

Figure 4 shows the geometric features that compose the feature vector. To decide whether two shapes are similar we just compare (e.g. using dot–product) their feature vectors. This contrasts to using the feature vectors to compute a

$$\left[\frac{P_{ch}}{T_l}\;\frac{A_{ch}}{P_{ch}^2}\;\frac{H_{er}}{W_{er}}\;\frac{A_{lq}}{A_{er}}\;\frac{A_{ch}}{A_{er}}\;\frac{A_{lq}}{A_{ch}}\;\frac{A_{lt}}{A_{lq}}\;\frac{A_{lt}}{A_{ch}}\;\frac{P_{lq}}{P_{ch}}\;\frac{P_{lt}}{P_{ch}}\;\frac{P_{ch}}{P_{er}}\right]$$

Fig. 4. Geometric feature vector.

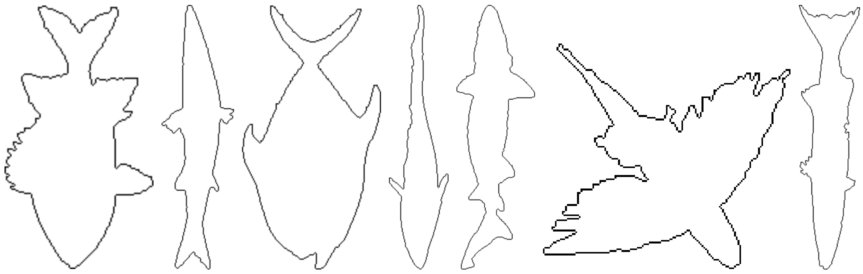


Fig. 5. Example of objects stored into the test database.

classification (e.g. rectangle or circle) and then comparing the classes ascribed to each shape. Our approach tends to work well if individual features are stable and robust, which we have found out experimentally in [8].

5 Experimental Results

In order to evaluate the retrieval capability (i.e. accuracy) of our method, we measured recall and precision performance figures using calibrated test data. Recall is the percentage of similar drawings retrieved with respect to the total number of similar drawings in the database. Conversely, precision is the percentage of similar drawings retrieved with respect to the total number of retrieved drawings.

We compared our method to describe shapes (CALI) with five other approaches, namely Zernike Moments (ZMD), Fourier descriptors (FD), grid-based (GB), Delaunay triangulation (DT) and Touch-point-vertex-angle-sequence (TPVAS). To that end we used results of an experiment previously performed by Safar [21], where he contrasted his approach (TPVAS) to the FD, GB and DT methods.

In that experiment, authors used a database containing 100 contours of fish shapes, as the ones presented in Figure 5. From the set of one hundred shapes in the database, five were selected randomly as queries. Before measuring the effectiveness of all methods, Safar performed a perception experiment where users had to select (from the database) the ten most similar to each query. This yielded the ten most perceptually similar results that each query should produce.

We repeated this experiment, on the same database and performing the same queries, using our method and an implementation of Zernike moments.

First we computed descriptors for each of the hundred shapes in the data set. Then for each query, we computed the corresponding descriptor and used it to search for the ten nearest-neighbors. For each of the five queries, we determined the positions for the 10 similar shapes in the ordered response set. Using results from our method and the values presented in Table 2 from [21] we produced the precision-recall plot shown in Figure 6.

Looking at the precision-recall chart we can see that our approach outperforms all other algorithms studied, including for the most part, the Zernike moments, which according to a previous experimental evaluation [23], were considered the best method to describe geometric shapes. Furthermore, our technique yields superior precision to all methods for all measured recall values except for recall values equal or below 20% where Zernike moments show a slight advantage.

Thus we can say that our method presents better results when drawings in the database are slightly different from the query, while Zernike moments tend work better for elements in the database which are very similar to the query. While Zernike moments tend to present better results in the topmost three to five queries, our method will likely yield more correct matches, although some of these might be ranked in lower positions. Thus, we believe that our technique to describing shape geometry is more suited to approximate queries in content-based retrieval than Zernike moments.

Although the features used by CALI were mainly selected to classify and describe geometric shapes, we can conclude from this experimental evaluation, that it can also be used to describe more general shapes, as the contours from this database. Furthermore, our geometric features were chosen to classify convex objects out of a limited vocabulary. One interesting finding is that the set is surprisingly expressive and general enough for measuring shape similarity instead of classification.

To assess the applicability of our approach for content-based retrieval in real-life settings, we developed two prototypes, one to retrieve technical drawings (SIBR) [10] and other for clip-art drawings (BajaVista) [5]. The SIBR prototype allows retrieving sets of drawings similar to a hand-sketched query or a digitized drawing. Figure 7(a) depicts a screen-shot of the calligraphic interface of the SIBR application. On the left we can see the sketch of a part and on the right

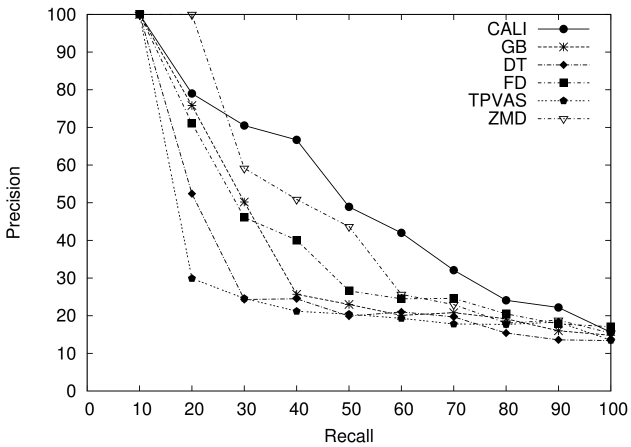
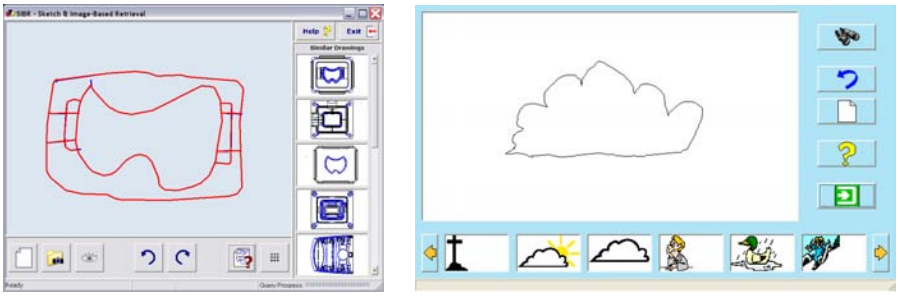


Fig. 6. Precision-recall comparison



(a) SIBR (b) Bajavista

Fig. 7. Screen-shots of prototypes.

the results returned by the implied query. These results are ordered from top to bottom and from left to right, with the most similar on top. On the other hand, the BajaVista prototype can index and retrieve clip-art drawings by content, either using sketches or querying by example. Figure 7(b) depicts a screen-shot of this application. On the top-left we can see the sketch of a cloud and on the bottom results returned by the implied query. These results are ordered from left to right, with the most similar on the left. It is also possible to perform query-by-example, thus allowing the user to select one of the results and using it to specify the next query, since our classification scheme handles graphics and sketches in the same manner.

These two prototypes were evaluated using medium–size databases. The SIBR prototype was tested on a database containing one hundred elements, while the database used to test BajaVista indexed 968 drawings. Tests with both prototypes showed effective results when searching for both technical or clip-art drawings. Furthermore users were satisfied that returned results matched their expectations. Indeed, while in first instances we presented the topmost five drawings in each case, feedback from tests convinced us to increase the displayed set to ten or twenty drawings. Surprisingly to us, users assigned greater importance to being able to retrieve the desired result among the top 10 or 20 elements, rather than finding the two "best" candidates. Indeed, we were told by users that they preferred recall over precision (at least in this limited sense) which empirically supports our claims about the precision–versus–recall performance of our technique against Zernike moments.

6 Conclusions

We presented a shape classification method which can be applied to content–based retrieval of two–dimensional vector graphics. Unlike other approaches, our method works not by a-priori classifications but by estimating the resemblance of a given shape to each of the forms in a restricted set. In this manner we are able to characterize many different two–dimensional shapes with few restrictions. Experimental evaluation of our method seems to indicate superior performance against other known sound approaches. However, we have not tested it with strongly concave shapes, for which it is not clear whether convex geometrical features will work well. This is the subject of ongoing work.

From an analysis of experimental results, our approach on shape classification for retrieval proved well on both theoretical and practical grounds.

We have developed successful applications for retrieving two–dimensional CAD drawings and clip–art images. One area for future work lies in extending our approach to three–dimensional vector drawings, where preliminary findings seem to yield promising results [6]. We strongly believe that an approach based on explicit structural descriptions has the potential to find a wide range of applications for human–made vector drawings.

References

1. J. E. Boyce and D. P. Dobkin. Finding Extremal Polygons. *SIAM Journal on Computing*, 14(1):134–147, February 1985.
2. Dragos Cvetkovic, Peter Rowlinson, and Slobodan Simic. *Eigenspaces of Graphs*. Cambridge University Press, United Kingdom, 1997.
3. E. R. Davies. *Machine Vision: Theory, Algorithms, Practicalities*. Academic Press, 1997.
4. Manuel J. Fonseca. *Sketch-Based Retrieval in Large Sets of Drawings*. PhD thesis, Instituto Superior Técnico / Universidade Técnica de Lisboa, July 2004.
5. Manuel J. Fonseca, Bruno Barroso, Pedro Ribeiro, and Joaquim A. Jorge. Retrieving ClipArt Images by Content. In *Proceedings of the 3rd International Conference on Image and Video Retrieval (CIVR'04)*, LNCS. Springer-Verlag, 2004.
6. Manuel J. Fonseca, Alfredo Ferreira, and Joaquim A. Jorge. Towards 3D Modeling using Sketches and Retrieval. In *Proceedings of the first Eurographics Workshop on Sketch–Based Interfaces and Modeling*, Grenoble, France, 2004. EG Publishing.
7. Manuel J. Fonseca and Joaquim A. Jorge. CALI : A Software Library for Calligraphic Interfaces. INESC-ID, available at `http://immi.inesc-id.pt/cali/`, 2000.
8. Manuel J. Fonseca and Joaquim A. Jorge. Experimental Evaluation of an on-line Scribble Recognizer. *Pattern Recognition Letters*, 22(12):1311–1319, 2001.
9. Manuel J. Fonseca and Joaquim A. Jorge. Experimental evaluation of an on-line scribble recognizer. *Pattern Recognition Letters*, 22(12):1311–1319, Jan 2001.
10. Manuel J. Fonseca, Alfredo Ferreira Jr., and Joaquim A. Jorge. Content-Based Retrieval of Technical Drawings. *International Journal of Computer Applications in Technology (IJCAT)*, 23(2/3/4):86–100, 2005.
11. H. Freeman and A. Saghri. Generalized Chain Codes for Planar Curves. In *Proceedings of the International Joint Conference on Pattern Recognition*, pages 701–703, Kyoto, Japan, November 1978.
12. Herbert Freeman and Ruth Shapira. Determining the Minimum-area Encasing Rectangle for an Arbitrary Closed Curve. *Communications of the ACM*, 18(7):409–413, July 1975.
13. Chuang Gu. *Multivalued Morphology and Segmentation-based Coding*. Phd thesis, Signal Processing Lab. of Swiss Federal Institute of Technology at Lausanne, 1995.
14. Ming-Kuei Hu. Visual Pattern Recognition by Moment Invariants. *IRE Transactions on Information Theory*, 8:179–187, 1962.

15. Hannu Kauppinen, Tapio Seppanen, and Matti Pietikainen. An Experimental Comparison of Autoregressive and Fourier-Based Descriptors in 2D Shape Classification. *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*, 17(2):201–207, 1995.
16. G. J. Lu and A. Sajjanhar. Region-Based Shape Representation and Similarity Measure Suitable for Content-Based Image Retrieval. *Multimedia Systems*, 7:165–174, 1999.
17. B. M. Mehtre, M. S. Kankanhali, and W. F. Lee. Shape Measures for Content Based Image Retrieval: A Comparison. *Information Processing and Management*, 33(3):319–337, 1997.
18. F. Mokhtarian, S. Abbasi, and J. Kittler. Efficient and Robust Retrieval by Shape Content through Curvature Scale Space. In *International Workshop on Image Databases and Nultimedia Search*, pages 35–42, Amsterdam, The Netherlands, 1996.
19. Joseph O'Rourke. *Computational Geometry in C.* Cambridge University Press, 2nd edition, 1998.
20. Eric Persoon and King-Sun Fu. Shape Discrimination Using Fourier Descriptors. *IEEE Trans. on Systems, Man and Cybernetics*, 7(3):170–179, 1977.
21. Maytham Safar, Cyrus Shahabi, and Chung hao Tan. Resiliency and Robustness of Alternative Shape-Based Image Retrieval Techniques. In *Proceedings of IEEE International Database Engineering and Applications Symposium (IDEAS'00)*, pages 337–348, 2000.
22. Maytham Safar, Cyrus Shahabi, and Xiaoming Sun. Image Retrieval By Shape: A Comparative Study. In *Proceedings of the IEEE International Conference on Multimedia and Exposition. (ICME'00)*, pages 141–144, 2000.
23. D. S. Zhang and G. Lu. A comparative study of curvature scale space and fourier descriptors. *Journal of Visual Communication and Image Representation*, 14(1):41–60, 2003.