Efficient and Robust Retrieval by Shape Content through Curvature Scale Space

Farzin Mokhtarian, Sadegh Abbasi and Josef Kittler

Vision Speech and Signal Processing Group
Department of Electronic & Electrical Engineering,
University of Surrey
Guildford, Surrey GU2 5XH
England
Email F.Mokhtarian@ee.surrey.ac.uk
Tel +44-1483-300800 extension 2288
Fax +44-1483-34139

Abstract. We introduce a very fast and reliable method for shape similarity retrieval in large image databases which is robust with respect to noise, scale and orientation changes of the objects. The maxima of curvature zero_crossing contours of Curvature Scale Space (CSS) image are used to represent the shapes of object boundary contours. While a complex boundary is represented by about five pairs of integer values, an effective indexing method based on the aspect ratio of the CSS image, eccentricity and circularity is used to narrow down the range of searching. Since the matching algorithm has been designed to use global information, it is sensitive to major occlusion, but some minor occlusion will not cause any problems.

We have tested and evaluated our method on a prototype database of 450 images of marine animals with a vast variety of shapes with very good results. The method can either be used in real applications or produce a reliable shape description for more complicated images when other features such as color and texture should also be considered.

Since shape similarity is a subjective issue, in order to evaluate the method, we asked a number of volunteers to perform similarity retrieval based on shape on a randomly selected small database. We then compared the results of this experiment to the outputs of our system to the same queries and on the same database. The comparison indicated a promising performance of the system.

1 Introduction

Shape representation is one of the most challenging aspects of computer vision. The problem has proven to be difficult [8][9], because shapes are often more complex than color and texture. While color and texture can be quantified by a few parameters, common shapes need hundreds of parameters to be represented explicitly.

The problem remains difficult in similarity retrieval applications in image databases. For example in [7], the authors have noted lack of reliability of their shape feature measurements. Because of the complexity and the huge variety of shapes, the problem of user interface in shape similarity retrieval has its own difficulties. While the user can specify the desired image texture or color using a menu, it is difficult to represent the same menu for shape representation.

Most proposed content based database systems aim to retrieve a small set of candidate images which include the desired image. The successful retrieval of the best candidate then relies on the final user judgement. In [1], the authors have used Polygonal approximation, while a set of features like boundary/perimeter, elongation (major axis/ minor axis), number of holes, etc, have been used in [2] for shape similarity retrieval. The authors in [3] have used a combination of heuristic shape features such as area, circularity, eccentricity, major axis orientation and a set of algebraic moment invariants. They have also used other features such as color, texture, and even sketch features.

We use a modified version of Curvature Scale Space image matching [5] for comparing shapes of objects in an image database. Our prototype database includes more than 450 colored images of marine animals, with every image containing one animal. The preprocessing step (consisting of gray-level morphology, thresholding and binary morphology) extracts the boundaries of objects. Other techniques such as active contours can also be incorporated at this stage if necessary. We compute the CSS image of every boundary and then find the maxima of CSS contours which are used as a shape descriptor to compare objects. The coordinates of these points together with the *aspect ratio* of the CSS image (number of rows / number of columns), eccentricity [10], circularity, and the name of the original image constitute a record which represents the object.

To retrieve similar images from the database, the user can either input an image and ask the system to find all images similar to it or sketch a boundary of his/her desired object using a painting package such as *xpaint*. The system computes the CSS image of the input and finds its maxima, and after comparison, assigns a matching value to every image *candidate* in the database which is similar to the input and shows the first n matched images with best values where n is determined by the user. The *candidates* are those images which their aspect ratio, eccentricity, and circularity, fall in the certain interval of the input ones. The acceptable interval can be selected by the user.

2 Curvature Scale Space image

Let Γ be a closed planar curve, and let u be the normalized arc length parameter on Γ:

$$\Gamma = \{ \, (x(u), y(u)) \; | \; u \in [0,1] \, \}$$

If each coordinate function of Γ is convolved with a 1-D Gaussian kernel of width σ , the resulting curve, Γ_σ, will be smoother than Γ . The locations of curvature zero crossings of Γ_σ can then be found [4]. As σ increases, Γ_σ becomes smoother and the number of zero crossings on it decreases. When σ becomes sufficiently high, Γ_σ will be a convex curve with no curvature zero crossings (see figure 1). The process can be terminated at this stage and the resulting points can be mapped to the (u, σ) plane. The result of this process will be a binary image called Curvature Scale Space image of the curve (see figure 2). The horizontal axis in this image represents the normalized arc length u, and the vertical axis represents σ, the width of the Gaussian kernel. The intersection of every horizontal line with the contours in this image indicates the locations of curvature zero crossings on the corresponding evolved curve Γ_σ.

Every object in our database is represented by the x and y coordinates of its boundary points. The number of these points varies from 400 to 1200 for these images. To normalize the arc length, we re-sample the boundary and represent it by 200 equally distant points. Therefore, the perimeter of all boundaries will be the same and every point on the boundary has a correspondence in the horizontal axis of the CSS image (figure 2).

Every CSS contour corresponds to a concavity or a convexity on the original boundary. For example in the first row of figure 2, there are six main contours in the CSS image, and there are six concavities or convexities in the relevant boundary. This correspondence is shown by numbering the contours on the CSS image and the regions on the boundary.

The boundary will finally be represented by the locations of the six maxima of its CSS image contours, shown in the third column of figure 2.

Also note that every re-sampled point on the boundary can be considered as the starting point. A change in the starting point only causes a circular shift in the CSS image. This can be observed by comparing the second and the third rows of figure 2.

3 CSS matching

As mentioned before, every object in the database is represented by the locations of the maxima of its CSS image. In this section we explain the basic concepts of our matching algorithm which compares two sets of maxima and assigns a matching value to them which represents the similarity between the actual boundaries of objects. For a more complete description of the CSS matching algorithm, see [6].

Consider the objects in figure 2. The regions 6 and 1 of the first object must be matched with the regions 7 and 8 of the second object respectively. Looking at the locations of the relevant maxima on the first and second row of this figure, we realize that they are in quite different positions. This is due to different starting points. If we change the starting point properly, then the locations of corresponding maxima on CSS images will be near each other. This can be observed on the third row of figure 2.

Therefore, the first step in CSS matching is to shift one of the two sets of maxima so that the effect of randomly selected starting point is compensated. Since the exact value of required shift is not available, we choose several values for it and then find the best match among them. The best choice is a value that shifts one CSS image so that its major maximum covers the major maximum of the other CSS image. Other possible choices are those values which accomplish the same with the second and possibly the third major maxima. For the two sets of maxima shown in figure 2, four choices are shown in figure 3.

Considering this figure, one can quickly realize that the first one is the best. Every maximum of the first CSS image is matched with a maximum of the second one, and two maxima remain unmatched. The matching value will be the summation of the the straight line distances between the matched pairs plus the vertical coordinates of the unmatched maxima.

4 Results and discussion and evaluation

We tested the proposed method on a database of 450 images of marine animals. Each image consisted of just one object on a uniform background. The system software was developed using the C language under Unix operating system. The response rate of the system was less than one second for every user query.

In this section we represent some of our experimental results through several examples. In these examples the inputs are images which already exist in the database. The first output of the system is always identical with the input image, with a zero match value.

In the example shown in figure 4a there is a difference in the view angle between the input and the fourth output and in figure 4b, the outputs are in different scales. This examples show that the system is robust with respect to scale and orientation changes of the objects. Other examples in this figure and figure 5 show the variety of shapes of objects in our database.

The evaluation of the performance of the system is a difficult task, because shape similarity is a subjective matter. We selected 50 images from our prototype database randomly and created a small database. We then selected 20 inputs from this database and asked a number of volunteers to find the shapes similar to every input from the database. The results of the subjective test indicated that human judgements of shape similarity noticeably differ. Interestingly though, the ranking produced by our system always agreed quite closely with, at least, a subset of the human evaluators. The short lists of the top five shapes generated by the different judges almost always included the "closest" machine selected shape. These findings indicate that the proposed approach is promising.

Four examples which can be used to compare the human judgements and the performance of the system are shown in figures 6 and 7 respectively.

We intend to test our method on another application involving a database of about 3000 varieties of chrysanthemum leaves. Each variety is represented by a sample of 10 leaves. The task is to check whether new varieties of the plant produced every year differ from all existing varieties. We believe that our method can be used to select varieties from the collection that have similarly shaped leaves to an unknown leaf and ease the process of testing a potential new variety.

Acknowledgements Sadegh Abbasi is grateful to the Ministry of Culture and Higher Education of Iran for its financial support during his research studies.

References

1. E. Gary and R. Mehrorta, " Shape similarity-based retrieval in image databases " in SPIE, Vol. 1662, 1992.
2. P. Eggleston, " Content based feature indexing and retrieval for image databases " in SPIE, Vol. 1819, 1992.
3. W. Niblack *et al* , " The QBIC project; querying images by content using color , texture and shape " in SPIE, VOL. 1908, 1993.
4. Mokhtarian, F. and Mackworth, A. K. " A theory of multiscale, curvature-based shape representation for planar curves " in IEEE Trans. on Pattern Anal. Mach. Intell. , VOL. PAMI-14, No. 8, August 1992 pp789-805.
5. Mokhtarian, F. " Silhouette-based isolated object recognition through curvature scale space " in IEEE Transactions on Pattern Analysis and Machine Intelligence, May 1995, Vol.17, No.5, pp.539-544
6. Mokhtarian, F., Abbasi S. and Kittler J. " Robust and Efficient Shape Indexing through Curvature Scale Space " in Proceedings of the 1996 British Machine and Vision Conference BMVC'96, September 1996, Edinburgh, Scotland.
7. Y. Gong, H. Zhang H.C. Chuan and M. Sakauchi , " An image database system with content capturing and fast image indexing abilities " in Proceedings of the International Conference on Multimedia Computing and Systems, 1994, pp.121-130.
8. Mumford, D. " Mathematical theories of shape: Do they model perception? " in Geometric Methods in Computer Vision, VOL. 1570, pages 2-10, SPIE, 1991.
9. Mumford, D. " The problem of robust shape descriptions " in First International Conference on Computer Vision, pages 602-606, London, England, June 1987. IEEE.
10. Heijden, F. " Image based measurement systems ", John Wiley & Sons, 1995.

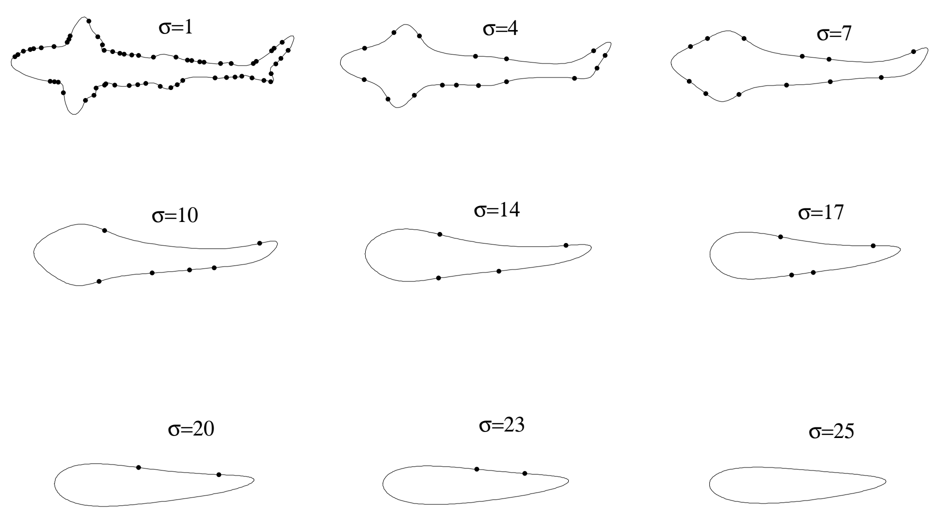


Fig. 1. Curve evolution.

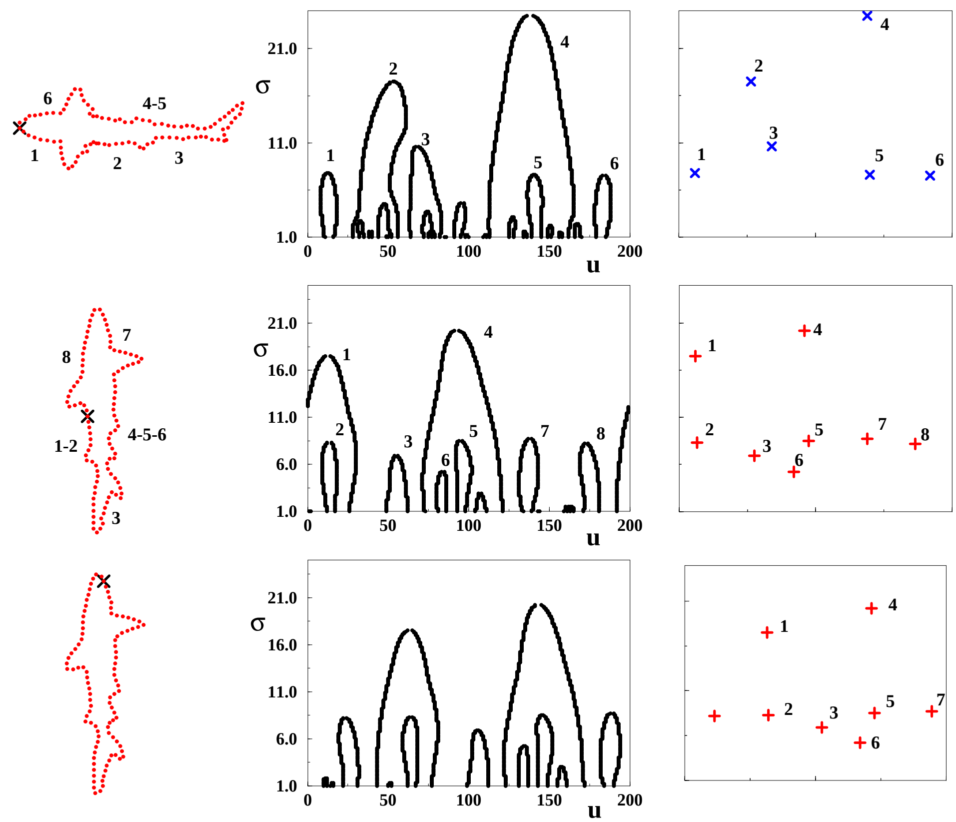


Fig. 2. CSS image and its maxima, left: re-sampled boundary with the marked starting point, middle: CSS image, right: normalized maxima of CSS images.

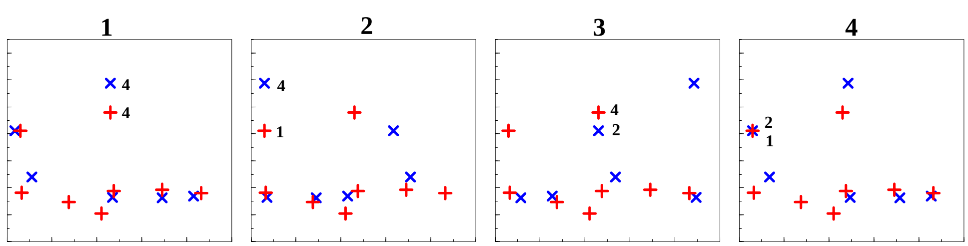
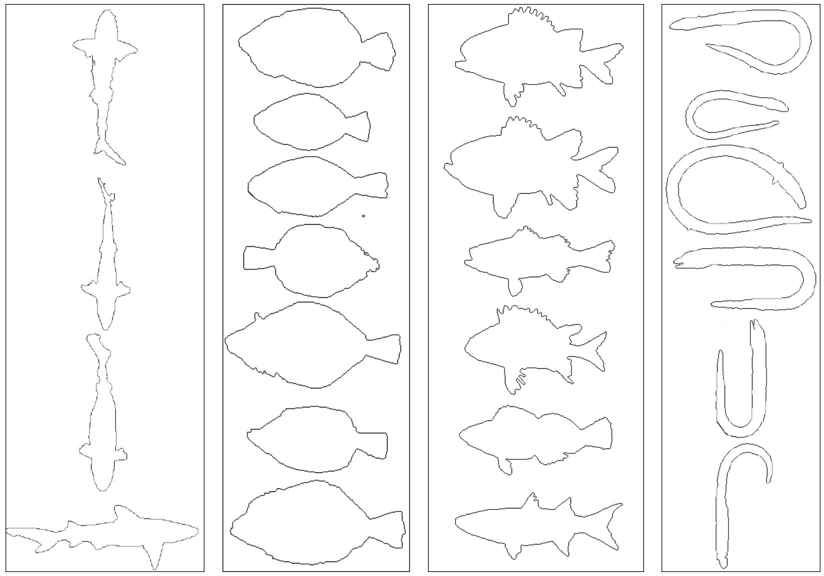
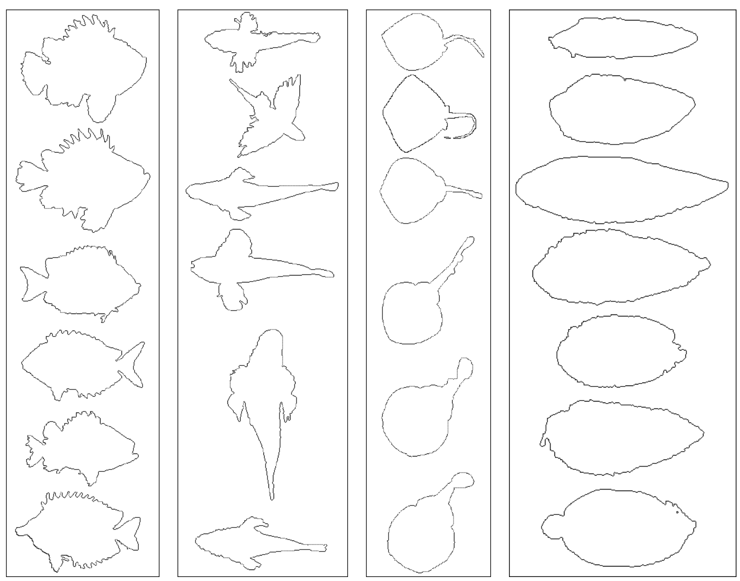


Fig. 3. Four possible choices for matching of the two sets of maxima related to first and second rows of figure 2.



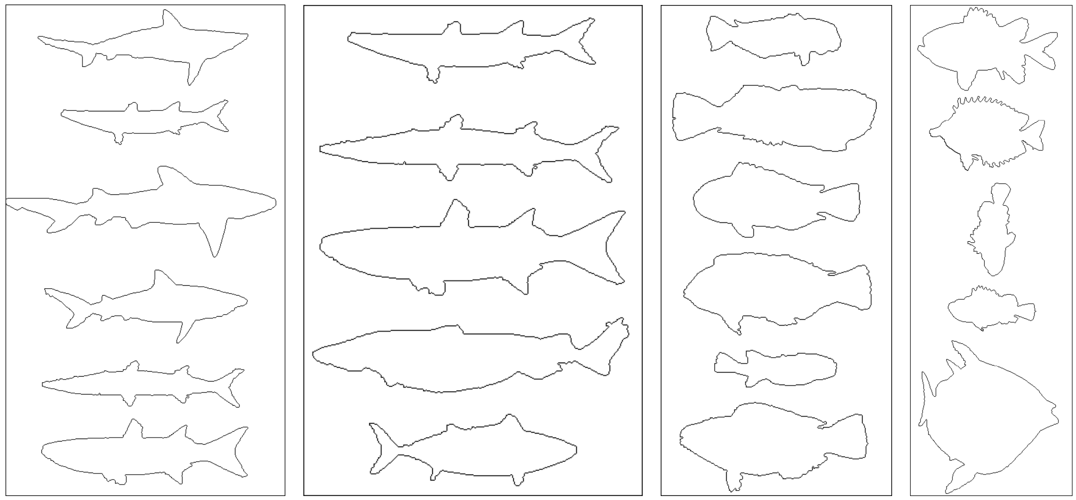
(a) **(b)** **(c)** **(d)**

Fig. 4. query results



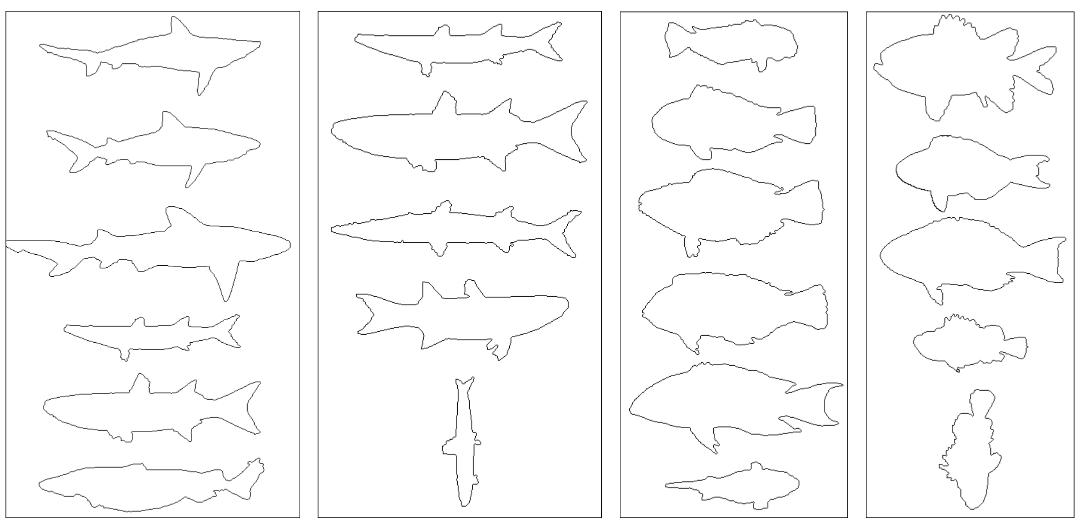
(a) **(b)** **(c)** **(d)**

Fig. 5. query results



(a) (b) (c) (d)

Fig. 6. Results of evaluation, human judgements. The first image is the input and the others are the most similar shapes found by volunteers.



(a) (b) (c) (d)

Fig. 7. Results of evaluation, system response to the same queries as above.