A Comparative Study of Three Region Shape Descriptors

Dengsheng Zhang and Guojun Lu
Gippsland School of Computing and Information Technology
Monash University
Churchill, Victoria 3842, Australia
Email: dengsheng.zhang, guojun.lu@infotech.monash.edu.au

Abstract

In Content Based Image Retrieval (CBIR), shape is one of the primary low level image features. Many shape representations have been proposed. However, most of them assume the knowledge of shape boundary information which is not available in general situations. Among them, region-based shape descriptors are not only applicable to generic shapes, but also robust to noise and distortions. In this paper we study and compare three region shape descriptors: Zernike moment descriptors (ZMD), grid descriptors (GD) and geometric moments descriptors (GMD). The strengths and limitations of these methods are analyzed and clarified. A Java frame retrieval framework is implemented to test the retrieval performance. The study and retrieval experiments on standard shape databases show that ZMD is the most suitable for shape retrieval in terms of computation complexity, compact representation, robustness, hierarchical coarse to fine representation and retrieval performance.

Keywords: *Zernike moments, grid, CBIR, shape*

1. Introduction

Since shape is a fundamental property of an object, an effective shape descriptor is a key component multimedia content description. Applications of shape description/ representation can also be found in many areas, such as meteorology, medicine, space exploration, manufacturing, entertainment, education, law enforcement and defense. There are generally two types of shape descriptors: contour-based shape descriptors and region-based shape descriptors.

Contour-based shape descriptors such as Fourier descriptors [5][15][21][9][20], curvature scale space [16] and shape signatures [3] exploit only boundary information, they cannot capture shape interior content. Besides, these methods cannot deal with disjoint shapes where boundary may not be available, therefore, they have limited applications.

In region based techniques, all the pixels within a shape region are taken into account to obtain the shape representation. Common region based methods use moment descriptors to describe shape [6][19][10][18][14]. These include geometric moments, Legendre moments, Zernike moments and pseudo Zernike moments. Recently, several researchers also use the grid method to describe shape [11] [17][2]. The grid-based method attracts interest for its simplicity in representation, conforms to intuition, and also agrees with shape coding method in MPEG-4. Since region-based shape representations combine information across an entire object rather than exploiting information just at boundary points, they can capture interior information in a shape. Other advantages of region-based methods are that they can be employed to describe disjoint shape and robust to shape distortions.

In these paper we study and compare three region-based shape descriptors: Zernike moments descriptors, grid descriptors and geometric moments descriptors. The principles we use for the comparison are the six requirements set by MPEG-7 [8], i.e. good retrieval accuracy, compact features, general application, low computation complexity, robust retrieval performance and hierarchical representation. The rest of the paper is organized as following. In Section 2, we describe the three region-based shape descriptors. Section 3 shows the results of the retrieval experiments. Section 4 concludes the paper.

2. Region-based Shape Descriptors

In this section, the three region-based shape descriptors to be compared are described in details.

2.1 Zernike Moments Descriptors

Teague [18] has proposed the use of orthogonal moments to recover the image from moments based on the theory of orthogonal polynomials, and has introduced Zernike moments, which allow independent moment invariants to be constructed to an arbitrarily high order.

The complex Zernike moments are derived from Zernike polynomials:

$$V_{nm}(x, y) = V_{nm}(\rho\cos\theta, \rho\sin\theta) = R_{nm}(\rho)\exp(jm\theta)$$

and

$$R_{nm}(\rho) = \sum_{s=0}^{(n-|m|)/2} (-1)^s \frac{(n-s)!}{s!(\frac{n+|m|}{2}-s)!(\frac{n-|m|}{2}-s)!} \rho^{n-2s}$$

where ρ is the radius from (x, y) to the shape centroid, θ is the angle between ρ and x axis, n and m are integers and subject to n-$|m|$ = *even*, $|m| \le n$. Zernike polynomials are a complete set of complex-valued function orthogonal over the unit disk, i.e., $x^2 + y^2 = 1$. Then the complex Zernike moments of order n with repetition m are defined as:

$$A_{nm} = \frac{n+1}{\pi}\sum_x \sum_y f(x, y)V_{nm}^*(x, y), \qquad x^2 + y^2 \le 1$$

Since Zernike basis functions take the unit disk as their domain, this disk must be specified before moments can be calculated. In our implementation, all the shapes are normalized into a unit circle of fixed radius of 64 pixels. The unit disk is then centered on the shape centroid. This makes the obtained moments scale and translation invariant. Rotation invariance is achieved by only using magnitudes of the moments. The magnitudes are then normalized into [0, 1] by dividing them by the mass of the shape. The similarity between two shapes indexed with Zernike moments descriptors is determined by the Euclidean distance between the two Zernike moments vectors. The block diagram of the whole process of computing ZMD is shown in Figure 1.

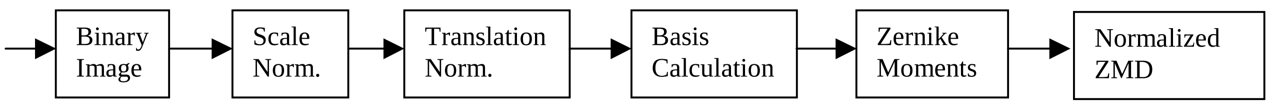


Figure 1. Block diagram of computing ZMD.

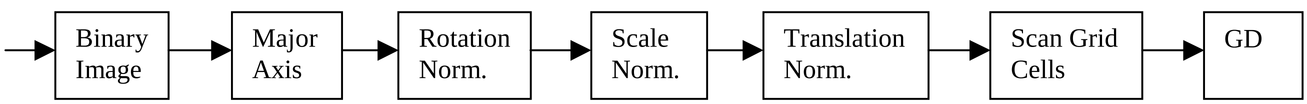


Figure 2. Block diagram of computing GD.

The theory of Zernike moments is similar to that of Fourier transform, to expand a signal into series of orthogonal basis. However, the computation of Zernike moments descriptors does not need to know boundary information, making it suitable for more complex shape representation. Like Fourier descriptors, Zernike moments descriptors can be constructed to arbitrary order, this overcomes the drawback of geometric moments in which higher order moments are difficult to construct. The precision of shape representation depends on the number of moments truncated from the expansion, the first 36 moments of orders up to 10 are used in our implementation in accordance to [ISO00].

2.2 Grid descriptors

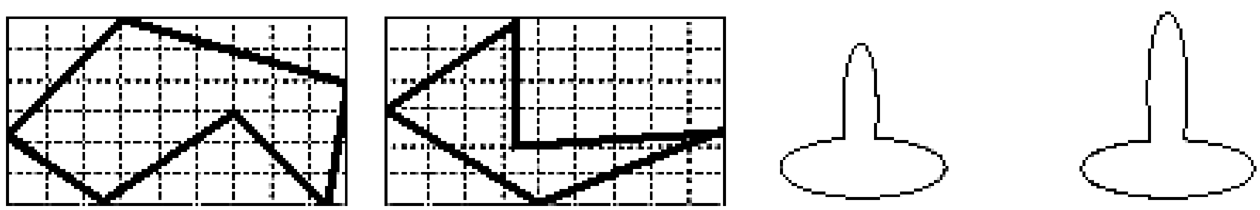
Several authors use grid-based method to describe shape [2][17][11]. In grid shape representation [11], a shape is projected onto a grid of fixed size, 16×16 grid cells for example. The grid cells are assigned the value of 1 if they are covered by the shape (or covered beyond a threshold) and 0 if they are outside the shape. A shape number consisting of a binary sequence is created by scanning the grid in left-right and top-bottom order, and this binary sequence is used as shape descriptors to index the shape.

For two shapes to be comparable using grid descriptors, several normalization processes have to be done to achieve scale, rotation and translation invariance. The block diagram of computing grid descriptors for a contour-based shape is given in Figure 2.

It begins with finding out the major axis, i.e., the line joining the two furthest points on the boundary. Rotation normalization is achieved by turning the shape so that the major axis is parallel with x-axis. To avoid multi normalization results for mirrored shape and flipped shape, the centroid of the rotated shape may be restricted to the lower-left part, or a mirror and a flip operation on the shape number are applied in the matching stage. Scale normalization can be done by resizing the shape so that the length of the major axis is equal to the preset grid width, and by shift the shape to the upper-left of the grid, the representation is translation invariant. The next step is scanning the grid cells so that a binary value is calculated for each cell based on the coverage of the cell by the shape boundary. Finally, a binary sequence is generated as shape descriptors. The distance between two set of grid descriptors is simply the number of elements having different values. For example, the grid descriptors for the two shapes in Figure 3 (a) and (b) are 001111000 011111111 111111111 111111111 111110011 001100011 and 001100000 011100000 111100000 111100000 011111100 000111000 respectively, and the distance between the two shapes will be 27 by XOR operation on the two sets. Since horizontally flipped and vertically flipped shapes will have different representations with the original shape even after normalization, the matching has to take into consideration of the two types of flipped shapes.

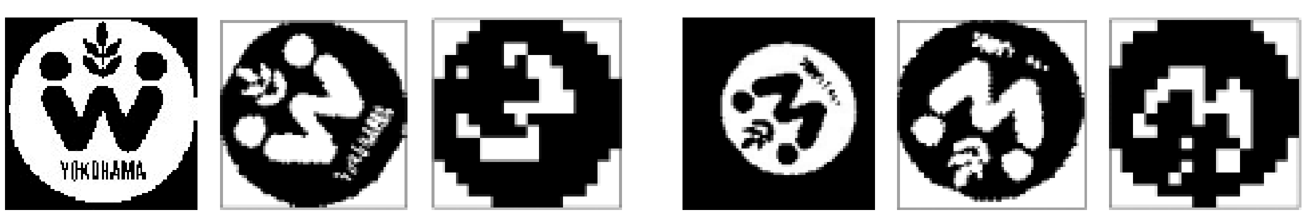
The above GDs computing algorithm is for contour-based shape. For region-based shape, the GD generation process is more complex. In the major axis computation step, it is not possible to find the major axis of a region shape using point by point computation, the computation would be prohibitive. Therefore, an algorithm of finding approximated major axis is used. The major axis for a region shape is found by finding the outer border point pairs on the shape boundary in a number of directions, for example, 180 directions. The pair with the furthest distance defines the major axis. A interpolation process is followed the rotation normalization, because after rotation, the region points are scattered. A similar interpolation is also needed for the scale normalization. Two examples of region-based shape normalization and grid representation are given in Figure 4.

Grid representation attracts interests because it agrees to human intuition and the shape coding method in MPEG-4. However, it is not robust, because a slight shape distortion, such as shear affine transform, skew and stretching can cause very big difference in the similarity measurement. Because the normalizations are mainly based on major axis (which is unreliable in essence) and eccentricity (which is only reliable for convex shapes or compact shapes), shapes otherwise similar may be treated as different due to this normalization. For example, the two shapes in Figure 3 (c) and (d) are perceptually similar, but are very different under grid representation, for the major axis of shape (c) is horizontal while the major axis of shape (d) will be vertical. Due to the negligence of interior content in the rotation normalization, grid representation is not rotation invariant for some region-based shapes, or shapes with interior content. For example, the rotation normalization does not work for the two shapes in Figure 4. The accuracy of shape representation also depends on the cell size and the threshold to determine the binary value of a cell based on its coverage by the shape. The online retrieval usually involves high computation due to the high dimensionality of the feature vectors (for a shape of 192×192 pixels using cell size of 12×12 pixels, the dimension is 196) and two extra matching of horizontally flipped and vertically flipped shapes.



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

Figure 3. (a)(b) Grid representation of two contour shapes; (c)(d) two similar shapes with very different grid representation.



(a) (b) (c) (d) (e) (f)

Figure 4. (a) original shape; (b) normalized shape of (a); (c) grid shape of (a); (d) original shape; (e) normalized shape of (d); (f) grid shape of (d);

2.3 Geometric Moments Descriptors

The technique based on geometric moment invariants for shape representation and similarity measure is extensively used in shape recognition. Moment invariants are derived from moments of shapes and are invariant to 2D geometric transformations of shapes. The central moments of order $p+q$ of a two dimensional shape represented by function $f(x, y)$ are given by

$$\mu_{pq} = \sum_x \sum_y (x-\bar{x})^p (y-\bar{y})^q f(x,y) \qquad p, q = 0, 1, 2...$$

where $\bar{x} = \mu_{10}/m$, $\bar{y} = \mu_{01}/m$ and m is the mass of the shape region. μ_{pq} are invariant to translation. The first 7 normalized geometric moments which are invariant under translation, rotation and scaling are given by Hu [Hu62]:

$$\Phi_1 = \eta_{20} + \eta_{02}$$
$$\Phi_2 = (\eta_{20} - \eta_{02})^2 + 4(\eta_{11})^2$$
$$\Phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$
$$\Phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$
$$\Phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$
$$\Phi_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$
$$\Phi_7 = (3\eta_{21} - \eta_{30})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{12} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$

where $\eta_{pq} = \mu_{pq}/(\mu_{00})^{\gamma}$ and $\gamma = 1+(p+q)/2$ for $p+q = 2, 3, ….$

A feature vector consists of the seven moment invariants: $\mathbf{f} = (\Phi_1, \Phi_2, \ldots \Phi_7)$ is used to index each shape in the database. The values of the computed moment invariants are usually small, values of higher order moment invariants are close to zero in some cases, therefore, all the invariants are further normalized into [0, 1] by the limit values of each dimension. The advantage of using GMD is it is a very compact shape representation and the computation is low, however, it is difficult to obtain higher order moment invariants.

3. Retrieval Experiments

To test the retrieval performance of the three region-based shape descriptors described above, a Java indexing and retrieval framework which runs on Windows platform is implemented. The retrieval performance of region-based shape descriptors are tested on both the contour-based shape database and the region-based shape database used by MPEG-7. The contour shape databases consists of Set B of MPEG-7 contour shape database. Set B has 1400 shapes of 70 classes. All the 1400 shapes in the database are used as queries. The region-based shape database used by MPEG-7 consists of 3,621 shapes of over 500 varieties. 31 classes of shapes (651) are selected as queries. Each class has 21 members.

The common retrieval performance measure – precision and recall [1] – are used as the evaluation of the query results. Precision P is defined as the ratio of the number of retrieved relevant shapes r to the total number of retrieved shapes n, i.e. $P = r/n$. Precision P indicates accuracy of the retrieval. Recall R is defined as the ratio of the number of retrieved relevant images r to the total number m of relevant shapes in the whole database, i.e. $R = r/m$. Recall R indicates the robustness of the retrieval performance. The average precision and recall of the 70 classes of shapes from the contour shape database is given in Figure 5(a), and the average precision and recall of the 31 classes of shapes from the region shape database is given in Figure 5(b). Some example screen shots of retrieval using the three shape descriptors are shown in Figure 6.

From the retrieval performance, it is clear that all the three region based descriptors have better retrieval performance on region based shape. In both cases, ZMD outperforms the other two descriptors and GMD performs poorly in both cases. Compared to ZMD, GD is less robust to boundary variations and its rotation normalization does not consider interior content of region-based shape. However, GD outperforms GMD significantly. It has been found that geometric moments' scale invariance is limited for region-based shapes, due to the use of shape mass for the scale normalization [6]. The shape mass does not change squarely proportional to scale in general situations, that is, when the shape is scaled to α times for generic shapes, the mass is not scaled to α^2 times. The low retrieval performance of GMD indicates it is a inaccurate shape representation.

4. Discussions and Conclusions

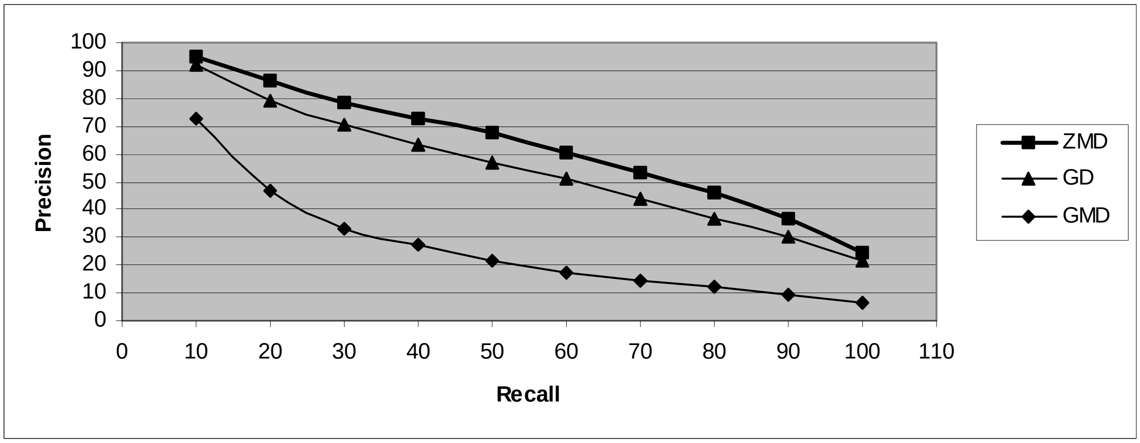
In this paper we have implemented and studied three region-based shape descriptors. The retrieval performance of the three region-based shape descriptors are obtained based on two standard shape databases. The comparison of the three region-based shape descriptors is given in the following.

- All the three shape descriptors studied can be applied to general applications.
- Feature domains. ZMD is extracted from spectral domain while GD and GMD are extracted from spatial domain.
- Compactness. The dimension of GMD and ZMD is low while that of GD is high.
- Robustness. ZMD is the most robust to shape variations among the three region-based shape descriptors, GD is more robust than GMD.
- Computation complexity. The extraction of GD and ZMD involves expensive computation while it is simple to extract GMD. The computation of GD is the most expensive both in the feature extraction and the online matching.
- Accuracy. At the same level of recall, the retrieval precision of ZMD is higher than both that of GD and that of GMD. However, the retrieval precision of GD is significantly higher than GMD. It indicates that

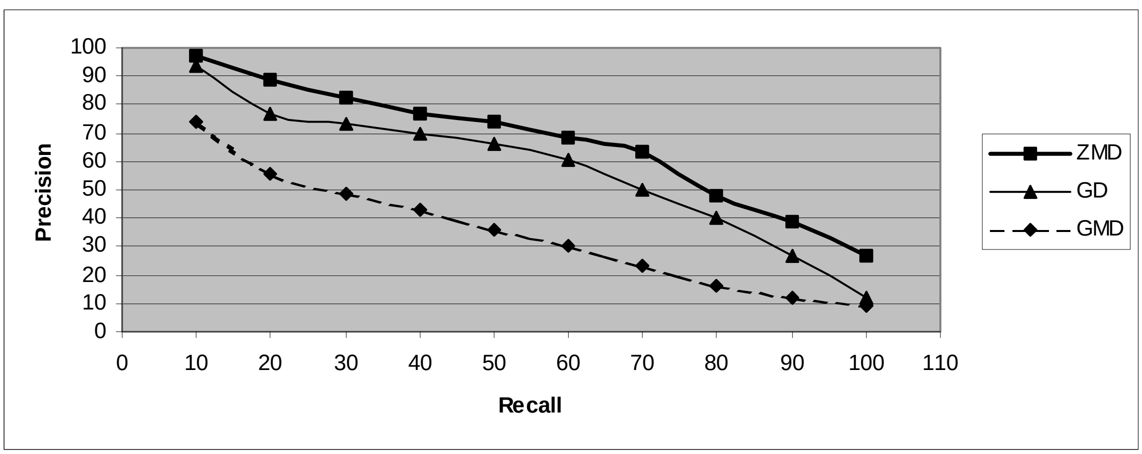
GMD is the most inaccurate shape descriptors among the three region-based descriptors.

- Hierarchical representation. Both ZMD and GD support hierarchical representation. The number of ZMDs can be adjusted to meet hierarchical requirement. For GD, hierarchical representation can be achieved by adjusting the cell size or combined with eccentricity and circularity. GMD does not support hierarchical representation because higher geometric moment invariants are difficult to obtain.
- Agree with human intuition. GD is a more intuitive shape representation than GMD and ZMD.

Based on the study, it has been found that ZMD extracted from spectral domain outperforms GD in terms of compactness, robustness, accuracy and low computation complexity; ZMD outperforms GMD in terms of robustness, accuracy, and hierarchical representation. Therefore, we conclude that ZMD is the most suitable for effective and efficient shape retrieval among the three region-based shape descriptors studied.



(a)



(b)

Figure 5. (a) Average retrieval performance of the three shape descriptors on contour-based shapes; (b) Average retrieval performance of the three shape descriptors on region-based shapes.

Reference:

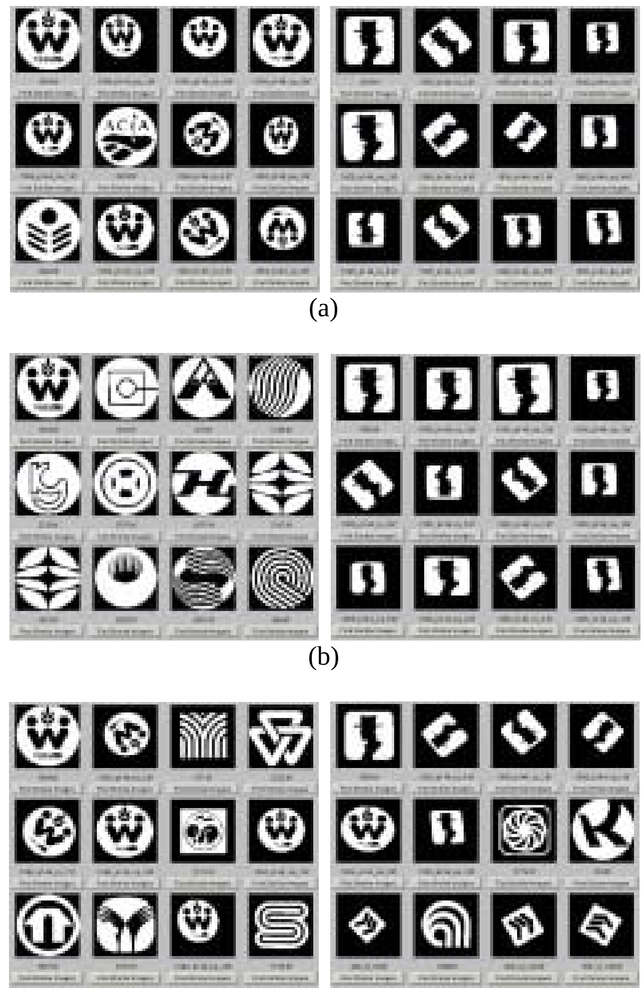
[1] A. Del Bimbo. Visual Information Retrieval. pp.56-57, Morgan Kaufmann Publishers, Inc. San Francisco, USA, 1999.

[2] K. Chakrabarti, M. O. Binderberger, K. Porkaew and S. Mehrotra, Similar Shape Retrieval in MARs, IEEE Int. Conf. On Multimedia and Expo, New York, USA, 2000, (CD-ROM Proceeding).

[3] E. R. Davies. Machine Vision: Theory, Algorithms, Practicalities. Academic Press, 1997, pp.171-190.

[4] A. S. Dudani, K. J. Breeding, R. B. McGhee. Aircraft Identification by Moment Invariants. IEEE

transaction on Computers, Vol. C-26, No. 1, pp. 39-46, January 1977.
[5] G. Granlund. Fourier Preprocessing for hand print character recognition. IEEE Trans. Computers, Vol. 21, 1972, pp.195-201.
[6] Ming-Kuei Hu. Visual pattern Recognition by Moment Invariants. IRE Transactions on Information Theory, IT-8:179-187, 1962.
[7] S. Jeannin Ed. MPEG-7 Visual part of experimentation Model Version 5.0. ISO/IEC JTC1/SC29/WG11/ N3321, Nordwijkerhout, March, 2000.
[8] H. Kim and J. Kim. Region-based shape descriptor invariant to rotation, scale and translation. Signal Processing: Image Communication 16:87-93, 2000.
[9] Hannu Kauppinen, Tapio Seppanen and Matti Pietikainen. An Experimental Comparison of Autoregressive and Fourier-Based Descriptors in 2D Shape Classification. IEEE Trans. PAMI-17(2):201-207, 1995
[10] Simon X. Liao and Miroslaw Pawlak. On Image Analysis by Moments. IEEE Trans. On Pattern Analysis and Machine Intelligence. 18(3):254-266, 1996.
[11] G. J. Lu and A. Sajjanhar. Region-based shape representation and similarity measure suitable for content-based image retrieval. Multimedia System 7:165-174, 1999.
[12] F. Mokhtarian, S. Abbasi and J. Kittler. Robust and Efficient Shape Indexing through Curvature ScaleSpace. Proc. British Machine Vision Conference, pp.53-62, Edinburgh, UK, 1996.
[13] D. Mohamad, G. Sulong and S. S. Ipson. Trademark Matching Using Invariant Moments. Proceedings of the Second Asian Conference on Computer Vision, pp. I439-III444, Singapore, 5-8 December 1995.
[14] W. Niblack et al. The QBIC Project: Querying Images By Content Using Color, Texture and Shape. SPIE Conf. On Storage and Retrieval for Image and Video Databases, vol 1908, San Jose, CA, pp.173-187, 1993.
[15] Eric Persoon and King-sun Fu. Shape Discrimination Using Fourier Descriptors. IEEE Trans. On Systems, Man and Cybernetics, Vol.SMC-7(3):170-179, 1977.
[16] A. Sajjanhar. A Technique for Similarity Retrieval of Shapes. Master thesis, Monash University, Australia, 1997.
[17] M. Safar, C. Shahabi and X. Sun. Image Retrieval by Shape: A Comparative Study, IEEE Int. Conf. On Multimedia and Expo, New York, USA, 2000, pp.
[18] Michael Reed Teague. Image Analysis Via the General theory of Moments. Journal of Optical Society of America, 70(8):920-930, 1980.
[19] C.-H. Teh and R. T. Chin. On image analysis by the methods of moments. IEEE Trans. On Pattern Analysis and Machine Intelligence, 10(4):496-513, 1988
[20] D. S. Zhang and G. J. Lu. Shape Retrieval Using Fourier Descriptors. In Proc. Int. Conference on Multimedia and Distance Education. Fargo, ND, USA, June 2001, pp.1-9.
[21] Charles T. Zahn and Ralph Z. Roskies. Fourier Descriptors for Plane closed Curves. IEEE Trans. On Computer, c-21(3):269-281, 1972.



(a)

(b)

(c)

Figure 6. Screen shots of shape retrieval using (a) ZMD; (b) GD; (c) GMD. In all the screen shots, the top left shape is the query shape.