Journal of Computer Science 6 (10): 1226-1232, 2010
ISSN 1549-3636

Retrieval of Similar Shapes Under Affine Transform Using Affine Length Parameterization

[1]A. Lakehal, [1]O. El Beqqali and [2]O. Ait Zemzami
[1]Department of Computer Science, Faculty of Sciences,
Dhar Mahraz University, 30000, Fès, Morocco
[2]Department of Computer Science, Faculty of Sciences and Technology,
Moulay Ismail University, P.O. Box 509, Boutalamine, 52000, Errachidiam, Morocco

Abstract: Problem statement: In this study, a new descriptor for shape retrieval under affine transformations had been proposed; the affine length parameterization was used in this approach. **Approach:** The zero crossing of affine curvature was used to extract a descriptor of shapes in database. For each two successive zero crossing points we extract two parameters: the first parameter was a maximal surface area of the triangle constructed by these points and a point between them. The second parameter is the average of affine curvature of the arc constructed by these two points. **Results:** The method was also evaluated objectively through a three classified databases which are a subset of the MCD database with a vast variety of shapes. The obtained results and the comparison with geometric moment and Fourier descriptors indicated a promising performance of our method. **Conclusion:** In this study, we examined the performance of the proposed descriptor under affine transformations. It was observed that the performance of the proposed method is promising even under severe deformations caused by shear.

Key words: Affine curvature, average affine curvature, affine transformation, shape similarity retrieval

INTRODUCTION

Image databases are consisted of single objects which can be simplified as shape images. In these cases, image retrieval becomes shape retrieval. Shape is also one of the important visual features for image representation. Shape descriptors can be divided into two main categories: region-based and contour-base methods (Choras, 2007). Region-based methods use the whole area of an object for shape description. Region-based shape descriptors can be applied to general applications. Some of the region based methods are grid based method, geometric moments and moments constructed orthogonal functions (Zhang, 2002), Zernike moment descriptors (Zhang and Lu, 2002a), Fourier descriptors (Sonk *et al*., 1998; Sajjanhar *et al*., 2008; Zhang and Lu, 2002b). For region based matching of shapes. Region based method, however, they usually involve more computation and its descriptors usually need more storage than contour-based descriptors. While contour-based methods use only the information present in the contour of an object, it transforms a shape image into one dimensional signature function for shape representation; it is much more efficient than the generally complex region based representation techniques which require two dimensional processing. Since they are computed using only boundary pixels, in general, their computational complexity is low and the sizes of their features are compact. Some of the contour-based methods are boundary moment descriptors (Chen, 1993; Sonk *et al*., 1998), curvature scale space descriptor (Abbasi *et al*., 2000; Mokhtarian and Bober, 2003; Mokhtarian *et al*., 1996; Mokhtarian and Mackworth, 1992) and Fourier descriptor (Zhang and Lu, 2005). A number of shape representations have been proposed to recognize shapes under affine transformation. Some of them are the extensions of well known methods, such as Fourier descriptors (Arbter *et al*., 1990), moment invariants (Flusser and Suk, 1993; Guggenheimer, 1963) and CSSD with affine length parameterization (Mokhtarian and Abbasi, 2001). In this study, a new descriptor based on a contour is proposed for 2d shape retrieval under affine transformations. The proposed method uses the smoothed contour for extracting the vectors describing a shape under affine transformations. After smoothing

Corresponding Author: A. Lakehal, Department of Computer Science, Faculty of Sciences, Dhar Mahraz University, 30000, Fès, Morocco

the curve, we calculate the surfaces areas of triangles constructed by each successive zero crossing points and a point between them so that this surface area is maximal, then we choose the n (the value of n is justified in the part of discussion) first values of the biggest surfaces areas of the triangles. These values construct the first vector of our descriptor, then we calculate the average affine curvature for each arc between two successive zero crossing points constructing the chosen triangles for constructing the second vector. Both vectors are used to charactering the shape. In this study we will examine the utility of using affine length rather than arc length to parameterize the curve prior to computing the proposed method. The proposed method is then used to find similar shapes from 280 shapes extracted of MCD database with a vast variety of shapes. Also the two classified databases consist of original as well as affine transformed shapes. This study begins by describing the proposed method, the experimental results, the discussion and finally the conclusion.

MATERIALS AND METHODS

Affine length: In computer vision, we always look for those descriptors of a shape which are invariant under certain transformations. For example, curvature is invariant under similarity transforms; i.e. rotation, translation and uniform scaling. However, under general affine transforms, the change in curvature is not a linear function of the transformation matrix. Affine curvature has been defined as an alternative for curvature which changes linearly under affine transforms. Therefore the normalized arc length parameter Eq. 1 is replaced by the normalized affine length parameter (Zhao and Chen, 1997) Eq. 2:

$$s(u) = \frac{\int_0^u (\dot{x}^2 + \dot{y}^2)^{\frac{1}{2}}}{\int_0^1 (\dot{x}^2 + \dot{y}^2)^{\frac{1}{2}}} \tag{1}$$

$$\tau(u) = \frac{\int_0^u (\dot{x}\ddot{y} + \ddot{x}\dot{y})^{\frac{1}{3}}}{\int_0^1 (\dot{x}\ddot{y} - \ddot{x}\dot{y})^{\frac{1}{3}}} \tag{2}$$

Affine curvature: The definition of affine curvature is based on affine length parameterization τ Eq. 2, which is a replacement for arc length parameterization s. The main disadvantage of the affine length is that its computation requires higher order derivatives. However, by using the method described in (Mokhtarian and Abbasi, 2001), we can parameterize the curve. The definition of affine curvature is based on affine length parameterization, as follows:

$$\tau(u) = \int_0^u (\dot{x}(\tau)\ddot{y}(\tau) - \ddot{x}(t)\dot{y}(\tau))^{\frac{1}{3}} dt$$

Consider a parametric vector equation for a continuous curve Γ:

$$\Gamma(\tau) = \Gamma\big(x(\tau), y(\tau)\big) \tau \in [0,1]$$

where, τ is the normalized affine length parameter for the original curve Γ. Affine curvature is then defined as the following:

$$v(\tau) = (\ddot{x}(\tau)\dddot{y}(\tau) - \dddot{x}(\tau)\ddot{y}(\tau) \tag{3}$$

It is much easier to compute affine curvature if it is expressed as a function of arbitrary parameter u. The final formula for affine curvature as a function of an arbitrary parameter u is obtained as follows (Mokhtarian and Abbasi, 2001; Matusiak, 1999):

$$\nu = \frac{\ddot{x}\tilde{y} - \tilde{x}\ddot{y}}{(\dot{x}\ddot{y} - \ddot{x}\dot{y})^{\frac{5}{3}}} - \frac{\begin{array}{c} 3(\tilde{x}\dot{y} - \dot{x}\tilde{y} + \ddot{x}\dddot{y})(\dot{x}\ddot{y} - \ddot{x}\dot{y}) \\ -5(\dot{x}\dddot{y} - \dddot{x}\dot{y})^2 \end{array}}{9(\dot{x}\ddot{y} - \ddot{x}\dot{y})^{\frac{8}{3}}} \tag{4}$$

Equation 4 presents an explicit formula for affine curvature as a function of an arbitrary parameter u, where $\dot{x}$, $\ddot{x}$, $\dddot{x}$ and $\tilde{x}$ denote the first, second, third and fourth derivatives of x with respect to u respectively. Derivatives of y are defined similarly. The main advantage of this descriptor is expected to be invariance under general affine transformations (Mokhtarian and Abbasi, 2001).

Curve smoothing prior to affine curvature measurement reduces the effects of noise. The computation starts from convolving each coordinate x(u) and y(u) of the curve with a Gaussian function g(u,σ). In continuous form we have:

$$X(u,\sigma) = x(u) * g(u,\sigma)$$

$$Y(u,\sigma) = y(u) * g(u,\sigma)$$

The 1D Gaussian function g(u,σ) of width σ is defined as follow:

$$g(u,\sigma)=\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{u^2}{2\sigma^2}}$$

where, "*" denotes the convolution operation, according to the properties of convolution, the derivatives of every component can be calculated easily:

$$\dot{X}(u,\sigma)=x(u)*\dot{g}(u,\sigma)$$

$$\ddot{X}(u,\sigma)=x(u)*\ddot{g}(u,\sigma)$$

The same for $\ddot{X}$ and $\tilde{X}$. We will have similar formulas for Y. As the exact form derivative of each order of g(u,σ) is known, the affine curvature of an evolved digital curve can be computed easily.

Vectors descriptor extracting: before extracting the vectors descriptor, we must smooth the curve to reduce the effects of noise. The standard deviation parameter σ must be properly chosen, the information imported by the curve must be kept and the noise must be eliminated. For this reason, we have fixed σ at $\sigma_0 = 5$ because experiments showed that σ = 5 gave good results with almost all shapes. The affine curvature zero crossing points should be determined at this level of smoothing. Due to the denominator in Eq. 4 affine curvature has not been defined at inflection points. As a result, whenever conventional curvature approaches to zero, affine curvature goes to infinity. The affine curvature zero crossings of the curve $\Gamma_{\sigma o}$ will be obtained by using method (Mokhtarian and Abbasi, 2001). For every curvature zero crossing point, we associated a point P_i of coordinates (x(u),y(u)), 1≤i≤N where N is the number of inflections points of the curve $\Gamma_{\sigma o}$. For each two successive zero-crossing points P_i and P_{i+1}, we search a point S_i of the arc $P_t\widehat{P}_{t+1}$ of the curve $\Gamma_{\sigma o}$ where the surface of the triangle $P_iS_iP_{i+1}$ is maximal as illustrated in the Fig. 1, then we choose the n first maximal value of the surfaces areas calculated. Denote V^S the vector descriptor constructed by n maximal value of the surfaces areas of the triangles called Surface-Vector. Then we calculated the average affine curvature between each successive zero-crossing points P_i and P_{i+1} where 1≤i≤n for constructing the second vector called Curvature-Vector noted V^C.

Similarity measurements: The process of the feature extraction is applied to each curve in the database. At the end, each curve in the database is represented and indexed using its feature vectors.

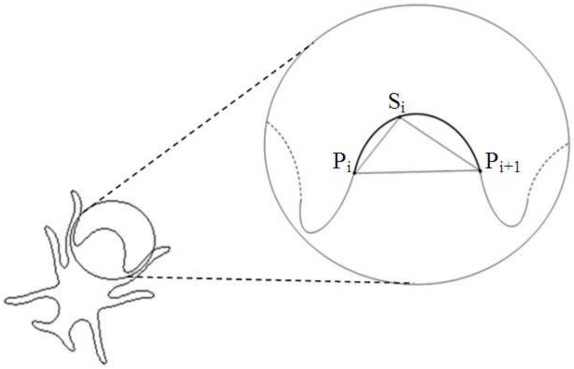

Fig. 1: Part of the boundary curve of the shape

Given a query feature vectors V^{CQ} and V^{sQ}, respectively the Curvature-Vector and Surface-Vector and the target feature vectors V^{CT} and V^{ST}. For measuring the similarity between the query and the database shape, we have calculated the distance between the feature vectors of them, the City-block distance is used since it is very easy to compute. If we suppose two points p = (x,y) and q = (u, v), the distance function is defined as follows:

$$d_1(p,q)=|x-u|+|y-\nu|$$

Then the similarity between the query shape Q and the shape of the database (target) T is measured as distance between the query feature vectors and target feature vectors which defined as:

$$d(Q,T)=\alpha\cdot d_1(V^{SQ},V^{ST})+(1-\alpha)\cdot d_1(V^{CQ},V^{CT}) \qquad (5)$$

where, $d_1(V^{SQ},V^{ST})=\sum_{i=1}^{n}|V_i^{SQ}-V_i^{ST}|$ the same for $d_1(V^{CQ},V^{CT})$.

General affine transformation: A general affine transformation T in R^2 is defined as

$$T(X)=AX+B \qquad (6)$$

where, $X\in R^2$, A is a nonsingular square matrix (a real matrix 2×2 with positive determinant) representing the rotation, scaling and shearing transformations and the vector $T\in R^2$ represents the translation vector. The Eq. 6 can be represented as follows:

$$\begin{pmatrix}x_a(\tau)\\ y_a(\tau)\end{pmatrix}=\begin{pmatrix}a_{11} & a_{12}\\ a_{21} & a_{22}\end{pmatrix}\begin{pmatrix}x(\tau)\\ y(\tau)\end{pmatrix}+\begin{pmatrix}b_1\\ b_2\end{pmatrix}=A\begin{pmatrix}x(\tau)\\ y(\tau)\end{pmatrix}+B \qquad (7)$$

where, (x (τ),y (τ)) represents a point on the contour parameterized by the affine length parameter τ, (x_a (τ),y_a (τ)) is the corresponding point after the affine transformation. The affine transformation matrix A can be expressed by several parameters as follows:

$$A = \begin{pmatrix} S_x & 0 \\ 0 & S_y \end{pmatrix} \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} 1 & k \\ 0 & 1 \end{pmatrix}$$

Scaling, rotation and shear may be represented by the following matrices:

$$A_{scaling} = \begin{pmatrix} S_x & 0 \\ 0 & S_y \end{pmatrix}; A_{rotation} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}; A_{shear} \begin{pmatrix} 1 & k \\ 0 & 1 \end{pmatrix}$$

If T is an affine transformation and A its matrix, the surfaces areas of the original and affine transformed curve respectively is related by this equation:

$$\text{Area}(T(\Delta)) = \det(A) \cdot \text{Area}(\Delta) \qquad (8)$$

By substituting x and y in Eq. 4 by x_a and y_a from Eq. 7, we obtain the following result:

$$v_a = \frac{1}{[\det(A)]^{\frac{2}{3}}} v \qquad (9)$$

where, x and x_a denote affine curvature of the original and affine transformed curves respectively.

Invariant to translation and rotation: since we use the affine length parameterization, the vectors V^S and V^C are the same for both the original shape and the transformed shape under translation and rotation.

Invariant to scaling: the general affine transformation becomes the scaling transformation when we substitute the matrix A by the matrix $A_{scaling}$, the Eq. 8 is rewritten as:

$$\text{Area}(T(\Delta)) = \alpha \cdot \text{Area}(\Delta) \qquad (10)$$

where, α = $S_x.S_y$ and Area(Δ) represents the surface area constructing the vector V^S, we normalized this vector by dividing each component of this vector by the maximal value of these components.

By substituting the matrix A by $A_{scaling}$ in Eq. 9 the relation between the affine curvature of the original shape and the transformed shape under scaling is given by:

$$v_a = \frac{1}{\alpha^{\frac{2}{3}}} v \qquad (11)$$

The vector V^C is constructed now, by normalizing each component of this vector by a maximal value of these components. Therefore, our descriptor becomes invariant to scaling transformation.

Invariant to shear: the shear transformation is represent mathematically by the Eq.7 if we substitute the matrix A by A_{shear}, as the determinant of this matrix is unit ($\det(A_{shear}) = 1$), Eq. 8 and 9 becomes:

$$\text{Area}(T(\Delta)) = \text{Area}(\Delta) \qquad (12)$$

$$v_a = v \qquad (13)$$

The Eq. 12 and 13 show that the surface triangle area and the affine curvature are the same for both original shape and the transformed shape under shearing transformation, by following the vectors V^S and V^C are invariant to shear transformation.

RESULTS

Retrieval performance test: in order to test the retrieval performance and the measurement of the proposed method using MCD database. The retrieval performance of our method is also compared with widely used shape descriptors: Geometric Moment Descriptor (GMD) and Fourier Descriptor using centroid distance (FD).

Database of test: The MCD database comprises 40 curve categories, each corresponding to a curve drawn from an MPEG-7 curve class. Each category in the dataset contains 14 curve samples that correspond to different perspective distortions of the original curve. We use 280 images as our test set of the MCD database, these shapes have been classified into 20 class as mentioned in the Fig. 2 (14 similar shapes in each class). The 20 class of shapes are used to test the overall robustness of a shape descriptor. In our experiment, all the 280 shapes from the 20 groups are used as queries to test the retrieval.

Performance measurement and test dataset: There are a number of ways to evaluate a shape descriptor. In the field of shape retrieval, precision and recall are the most widely used methods. Basically, recall measures the capacity to retrieve relevant items from the database and precision measures the retrieval accuracy:

$$\text{Precision} = \frac{\text{Number of relevant returned images}}{\text{Number of returned images}}$$

$$\text{Re call} = \frac{\text{Number of relevant returned images}}{\text{Number of images in the class of the query}}$$

The subset of the MCD database mentioned above are used to compare the retrieval of the three methods using the rappel-precision curve, all the 280 shapes in the subset are used as queries. The common retrieval performance measure precision and the recall are used as the evaluation of the query results. For each query, the precision of the retrieval at each level of the recall is obtained. The result precision of retrieval is the average precision of all the queries retrievals. The average precision and recall of the retrieval using the three descriptors are shown in Fig. 3.

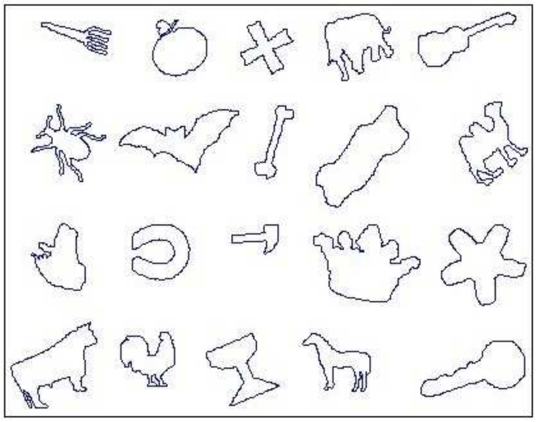

Fig. 2: Each curve represents a class of MCD database

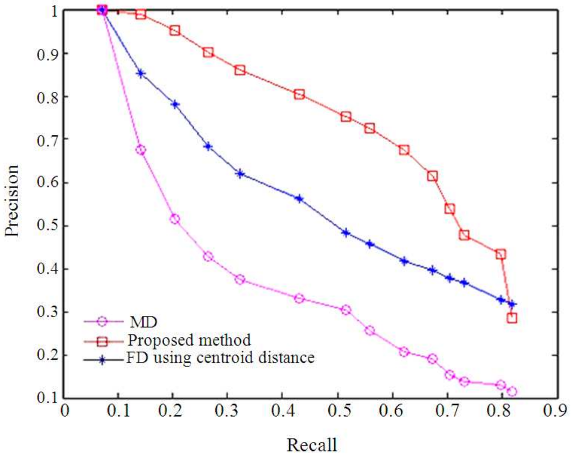

Fig. 3: Average retrieval performance of different method

To test the retrieval performance of the proposed method, we have used the subset of the MCD database as mentioned below. After making a sequence of experiments, we concluded that by giving the parameter α in Eq. 5 the value 0.7, the best results can be achieved. The retrievals of some queries are shown in the Fig. 4.

Invariance to shear: we will examine the performance of the proposed method under shear transformation with the constructed database as mentioned under. The measure of shape deformation depends upon the parameter k called shear ratio in the matrix A_{shear}. In the present form of the matrix A_{shear}, x axis is called shear axis, as the shape is pulled toward this direction. Figure 5 shows the effects of shear transformation on the shapes, the deformation is much more severe for values k is superior than 1.

To create a database, we choose nine different values for shear ratio parameter k, 0.25, 0.5, 0.75, 1, 1.25, 1.5, 1.75, 2 and 2.5. We then apply the transformation on a database of 40 original object shapes, each class of MCD database is represented by one shape from this class. From every original object, we obtain 9 transformed shapes with different values of k as mentioned before.

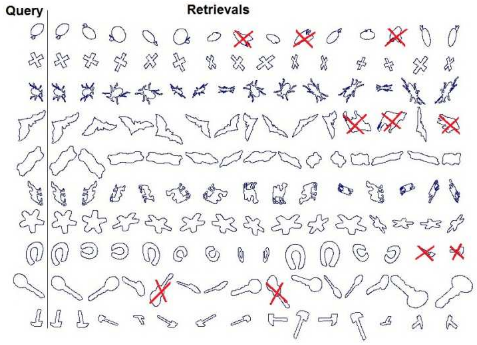

Fig. 4: Retrieval results obtained using 10 random queries from the MCD. The crosses mark retrievals that do not belong to the correct category

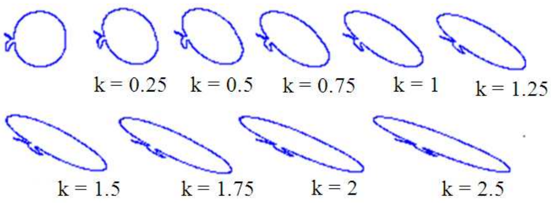

Fig. 5: The original shape presented in top left. Others represent the transformation with different value of the parameter shear ratio k

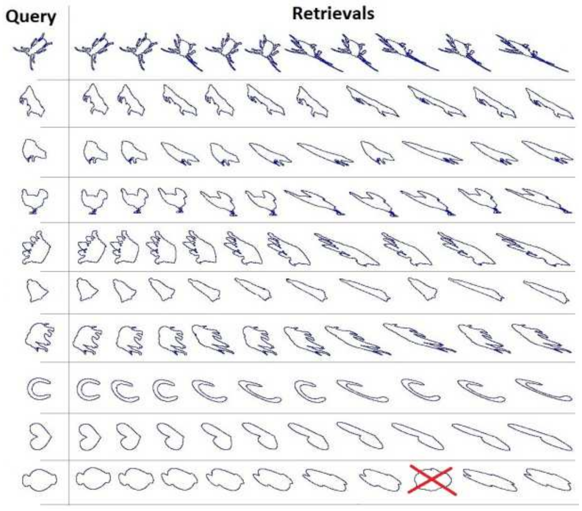

Fig. 6: Shape retrieval results under shear transformation

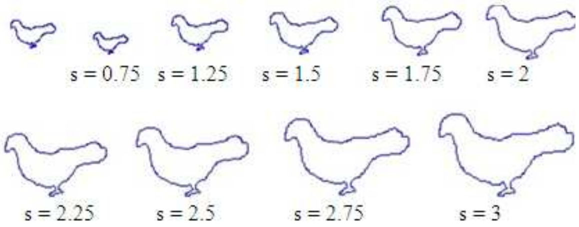

Fig. 7: The original shape presented in top left. Others represent the transformation with different value of scale parameter s

Therefore the database consists of 40 original and 360 transformed shapes. Therefore, the database consists of 400 shapes. We then carry out a series of experiments on this database to verify the robustness of the proposed method under shear transformation. Every original shape was selected as the input query and the first 10 outputs of the system were observed to see if the transformed versions of the query are retrieved by the system. We found out that for 99% of queries, all transformed versions of a shape appear in the first 10 outputs of the system. The results for some queries and the shapes retrievals by system are presented in Fig. 6.

Invariance to scale transformation: to examine the performance of our descriptor under uniform scaling transformation (S_x= S_y= s), we create a database by choosing nine different values for scaling parameter s, 0.75; 1.25; 1.50… 3 as shows in Fig. 7. We then apply the scaling transformation as we made for shear examine.

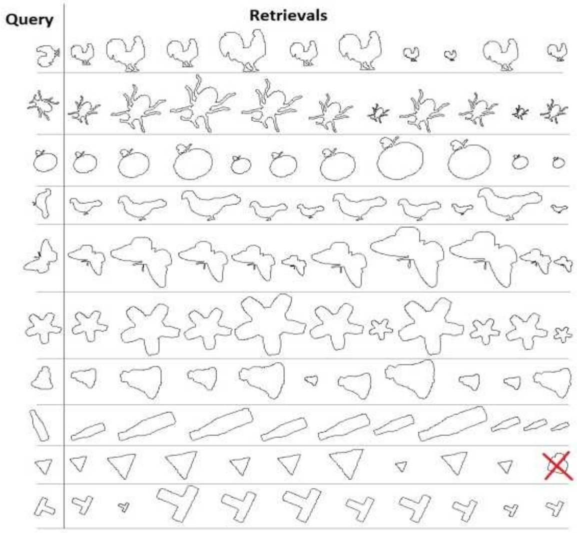

Fig. 8: Shapes retrievals results under scale transformation

The results for some queries and the shapes retrievals by system are presented in Fig. 8.

DISCUSSION

The choice of n, the size of the vectors V^S and V^C must takes in consideration the following reasons:

- The number n must be big enough so that the characteristic vectors represent well the shape, but still take into consideration the time of calculation
- It must not depend on a threshold which is related to the triangles areas calculated because we can lose important information concerning the shapes whose surfaces areas decrease sharply
- Some shapes after smoothing the number of triangles found very few, sometimes it does not exceed 10 triangles, then the value of n should not be very big

For these reasons we fixed the value of n in 20.

The boundary curve of some shapes after smoothing doesn't contain the sufficient triangles (n<20), so the vectors V^S and V^C are not completed. Therefore, for completing these vectors, we completing their components by zero.

CONCLUSION

In this study, we have proposed a method which uses the shape contour for indexing the shape.

Experiments have been performed on the MCD database. Experiments performed show the effectiveness of the proposed method and it is better than other methods using a subset of MCD database. We also show the robustness of our method under affine transformations.

REFERENCES

Abbasi, S., F. Mokhtarian and J. Kittler, 2000. Enhancing CSS-based shape retrieval for objects with shallow concavities. Image Vis. Comput., 18: 199-210. DOI: 10.1016/S0262-8856(99)00019-0

Arbter, K., W.E. Snyder, H. Burkhardt and G. Hirzinger, 1990. Applications of affine-invariant Fourier descriptors to recognition of 3-D objects. IEEE Trans. Patt. Anal. Mach. Intell., 12: 640-646. DOI: 10.1109/34.56206

Chen, C.C., 1993. Improved moment invariants for shape discrimination. Patt. Recog., 26: 683-686. DOI: 10.1016/0031-3203(93)90121-C

Choras, R.S., 2007. Image feature extraction techniques and their applications for CBIR and biometrics systems. Int. J. Biol. Biomed. Eng., 1: 6-16. http://www.naun.org/journals/bio/bio-2.pdf

Flusser, J. and T. Suk, 1993. Pattern recognition by affine moment invariants. Patt. Recog., 26: 167-174. DOI: 10.1016/0031-3203(93)90098-H

Guggenheimer, H.W., 1963. Differential Geometry. 1st Edn., McGraw-Hill, New York, pp: 230.

Matusiak, S., 1999. Local and invariant description of the planar shapes for indexing of the images database. Ph.D. Thesis, Valenciennes University. http://www.tele.ucl.ac.be/PEOPLE/Matusiak/Publi/These/

Mokhtarian, F. and A.K. Mackworth, 1992. A theory of multiscale, curvature-based shape representation for planar curves. IEEE Trans. Patt. Anal. Mach. Intell., 14: 789-805. DOI: 10.1109/34.149591

Mokhtarian, F. and M. Bober, 2003. Curvature Scale Space Representation: Theory, Applications and MPEG-7 Standardization (Computational Imaging and Vision). 1st Edn., Springer, USA., pp: 450.

Mokhtarian, F., S. Abbasi and J. Kittler, 2001. Affine curvature scale space with affine length parameterization. Patt. Anal. Appli., 4: 1-8. DOI: 10.1007/PL00010984

Mokhtarian, F. and S. Abasi, 1996. Efficient and Robust Retrieval by Shape Content through Curvature Scale Space. In: Image Databases and Multi-Media Search, Smeulders, A. and R. Jain (Eds.). World Scientific Publishing Co., Arnold WM., pp: 51-58.

Sajjanhar, A., G. Lu, D. Zhang and W. Zhou, 2008. Corners-based composite descriptor for shapes. Proceeding of the International Conference on Computer and Information Science, May 27-30, IEEE Xplore Press, Sanya, China, pp: 714-718. DOI: 10.1109/CISP.2008.698

Sonk, M., V. Hlavac and R. Boyle, 1998. Image Processing: Analysis and Machine Vision. 2nd Edn., CL-Engineering, USA., pp: 800.

Zhang, D. and G. Lu, 2002a. Improving retrieval performance of Zernike moment descriptor on affined shapes. GSCIT. http://personal.gscit.monash.edu.au/~dengs/resource/papers/icme02a.pdf

Zhang, D. and G. Lu, 2002b. A comparative study of Fourier descriptors for shape representation and retrieval. Proceeding of the 5th Asian Conference on Computer Vision, (CV'02), Springer, Melbourne, Australia, pp: 646-651. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.73.5993

Zhang, D. and G. Lu, 2005. Study and evaluation of different Fourier methods for image retrieval. Image Vis. Comput., 23: 33-49. DOI: 10.1016/j.imavis.2004.09.001

Zhang, D., 2002. Image retrieval based on shape. Ph.D. Thesis, Monash University. http://personal.gscit.monash.edu.au/~dengs/resource/papers/thesis.zip

Zhao, D. and J. Chen, 1997. Affine curve moment invariants for shape recognition. Patt. Recog., 30: 895-901. DOI: 10.1016/S0031-3203(96)00126-4