INTERNATIONAL JOURNAL OF COMPUTERS
Issue 1, Volume 7, 2013

Reverse Engineering of the Digital Curve Outlines using Genetic Algorithm

Muhammad Sarfraz, Malik Zawwar Hussain, and Misbah Irshad

Abstract**—**A scheme, which consists of an iterative approach for the recovery of digitized, hand printed and electronic planar objects, is proposed. It vectorizes the generic shapes by recovering their outlines. The rational quadratic functions are used for curve fitting and a heuristic technique of genetic algorithm is applied to find optimal values of shape parameters in the description of rational functions. The proposed scheme is fully automated and vectorizes the outlines of planar images in a reverse engineering way.

Keywords**—**Rational function, reverse engineering, genetic algorithm, images.

I. Introduction

IN the last two decades reorientation of traditional artificial intelligence methods has been noticed toward the soft computing techniques. This development allows us to solve difficult problems related to robotics, computer vision, speech recognition and machine translation. According to Zadeh [31], soft computing techniques are characterized by tolerance of imprecision, uncertainty and parallel truth to achieve tractability, robustness and low solution cost. Soft computing techniques such as fuzzy logic (FL), neural networks (NN), genetic algorithm (GA), simulated annealing (SA), ant colony optimization (ACO), and particle swarm optimization (PSO) have received a lot of attention of researchers due to their potentials to deal with highly nonlinear, multidimensional, and ill-behaved complex engineering problems [4].

Genetic Algorithm (GA) [8], an evolutionary technique, gives us a method to perform randomized global search in a solution space. In this space, a population of candidate solutions, called chromosomes, is evaluated by a fitness function in terms of its performance. The best candidates evolve and pass some of their genetic characteristics to their offsprings, who form the next generation of potential solutions. The process of reverse engineering of planar objects comprises of the steps like: extracting data from boundary of the shapes, finding the corner points using some technique and finally fitting curve to these corner points using rational quadratic functions and GA. This paper utilizes genetic algorithm technique for recovering the outlines of planar images in a reverse engineering way.

M. Sarfraz is with the Department of Information Science, Kuwait University, Adailiya Campus, P.O. Box 5969, Safat 13060, KUWAIT (phone: +965 2498 3109, e-mail:prof.m.sarfraz@gmail.com).

M. Z. Hussain is with the Department of Mathematics, University of the Punjab, Lahore, Pakistan (e-mail: malikzawwar.math @pu.edu.pk).

M. Irshad is with the Department of Mathematics, University of the Punjab, Lahore, Pakistan (e-mail: misbah1109@gmail.com).

Reverse engineering is quite a modern research field which deals with diverse activities. Its scientific perspective is generally related to computer science and herein to computer aided geometric design (CAGD). Reverse engineering of shapes is the process of representing an existing object geometrically in form of computer aided design (CAD) model. A good reverse engineering system not only creates a CAD model of the object, but it also helps exploring and understating the structure of the object. Generating computer aided design (CAD) model from scanned digital data is used in contour styling which needs to adopt some curve or surface approximation scheme.

Reverse engineering of planar objects is referred to the process of fitting an optimal curve to the data extracted from the boundary of the bitmap image [10, 12, 14, 15, 16]. Curve fitting is frequently used in reverse engineering to reproduce curves from measured points. It is always essential to provide new curve-fitting algorithms to acquire curves that satisfy different conditions. Fitting curves to the data extracted by generic planar shapes is the problem which is greatly worked on during last two decades. Still there is a room for researchers in this field due to its applications in diverse fields and its demand in the industry. There are several advantages of curved representation of planar objects. For example, transformations like scaling, shearing, translation, rotation and clipping can be applied on the objects very easily.

Various outline approximation techniques can be found in the literature in which different spline models have been used by the researchers like Be´zier splines [16], B-splines [9], Hermite interpolation [18] and rational cubic interpolation [19]. There are several other outline capturing techniques [3, 11, 13, 20-23, 29, 6, 24, 25, 27, 28] available in the current literature and most of them are based on least-squares fit [11, 13, 20] and error minimization [3, 19, 21]. Sarfraz et al. [22] in their outline capturing scheme, calculated the ratio between two intermediate control points and used this to estimate their position. This caused reduction of computation in subsequent phases of approximation. Few other techniques include use of control parameters [18], genetic algorithms [23], and wavelets [29]. In this work rational quadratic functions (conics) are used for curve fitting using genetic algorithm.

The paper is organized in a way that the first and second step of the proposed scheme, outline estimation and corner detection method, are described in Section 2. Review of rational cubic and rational quadratic functions is given in Section 3. Section 4 explains the proposed scheme which is demonstrated with examples in Section 5. The paper is concluded in Section 6.

II. Contour Extraction and Segmentation

First step in reverse engineering of planar objects is to extract data from the boundary of the bitmap image or a generic shape, shown in Fig. 1(a). Capturing boundary or outline representation of an object is a way to preserve the complete shape of an object. The objects in an image can also be represented by the interior of shape. Chain coding for boundary approximation and encoding was initially proposed by Freeman [7], which has drawn significant attention over last three decades. Chain codes represent the direction of the image and help to attain the geometric data from outline of the image. Extracted boundary of the bitmap image given in Fig. 1(a) is given in Fig. 1(b).

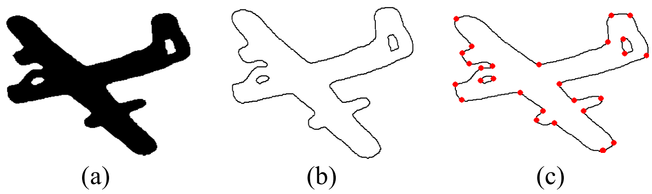
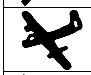


Fig. 1 (a) bitmap image of a plane, (b) detected boundary of the image (c) detected corner points from the boundary

Table 1. Details of Digital Contours and Corner points.

Image	Name	# of contours	# of contour points	# of initial corner points
	Fork	1	673	15
	Plane	3	915+36+54	28
	Fish	1	975	31

Segmentation of object boundary before curve fitting is very important for two reasons. Firstly, it reduces boundary's complexity and simplifies the fitting process. Secondly, each shape consists of natural break points (like four corners of a rectangle) and quality of approximation can be improved if boundary is subdivided into smaller pieces at these points. These are normally the discontinuous points to which we do not want to apply any smoothing and like to capture them as such. These points can be determined by a suitable corner detector. Researchers have used various corner detection algorithms for outline capturing [1, 2, 5, 27, 30]. The method proposed in [5] is used in this paper. Number of contour points and detected boundary points for different images is given in Table 1. Detected corners of the boundary shown in Fig. 1(b), can be seen in Fig. 1(c).

III. Rational Quadratic Functions

In this section piecewise rational quadratic functions are presented used for curve fitting which is an alternate of the rational cubic presented in Section 3.1. The rational quadratic possesses C^1 continuity

A. C^1 Rational cubic function

A piecewise rational cubic parametric function $P \in C^1[t_i, t_{i+1}]$, with shape parameters $v_i \geq 0, i = 1, \ldots, n,$ is used for curve fitting to the corner points detected from the boundary of the bitmap image, the rational cubic function is defined for $t \in [t_i, t_{i+1}]$, $i = 1, \ldots, n,$ as follows

$$P(t) = P_i(t) = \frac{F_i(1-\theta)^3 + v_i V_i (1-\theta)^2 \theta + v_i W_i (1-\theta)\theta^2 + F_{i+1}\theta^3}{(1-\theta)^3 + v_i(1-\theta)^2\theta + v_i(1-\theta)\theta^2 + \theta^3} \tag{1}$$

where F_i and F_{i+1} are two corner points (given control points) of the i^{th} segment of the boundary with $h_i = t_{i+1} - t_i,$.

$$V_i = F_i + \frac{h_i D_i}{v_i} \text{ and } W_i = F_{i+1} - \frac{h_i D_{i+1}}{v_i} \tag{2}$$

where D_i, $i = 1, \ldots, n+1$ are the first derivative values at the knots t_i, $i = 1, \ldots, n+1$.

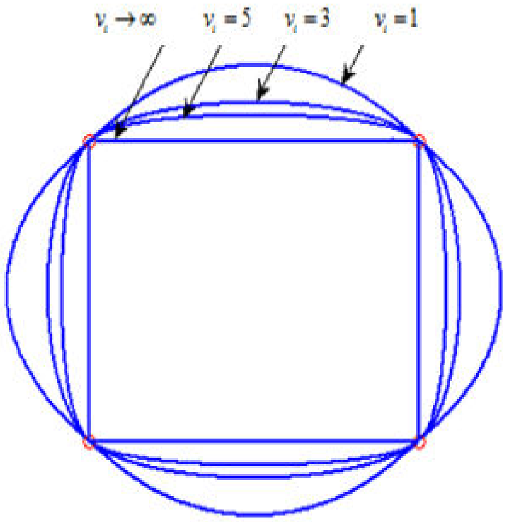


Fig. 2 Demonstration of rational cubic function (1)

It is to be noted that v_i, $i = 1, \ldots, n,$ are used to control the shape of the curve. Effect of these shape parameters on the curve is shown in Fig. 1 and Fig. 2. Moreover, for $v_i = 3$, $i = 1, \ldots, n,$ (1) represents cubic Hermite interpolation and it can be considered as default case of rational cubic (1). If

$v_i \to \infty$, then the rational cubic function (1) converges to linear interpolant given by

$$L_i(t) = (1-\theta)F_i + \theta F_{i+1} \tag{3}$$

which means that the increase in v_i pulls the curve towards F_i and F_{i+1} in the interval $[t_i, t_{i+1}]$ and the interpolant is linear as shown in Fig. 2.

For $v_i \neq 0$ Equation (1) can be written in the form

$$P_i(t_i; v_i) = R_0(\theta; v_i)F_i + R_1(\theta; v_i)V_i + R_2(\theta; v_i)W_i + R_3(\theta; v_i)F_{i+1} \tag{4}$$

where V_i and W_i are given in Equation (2) and $R_j(\theta; v_i),\ j = 0,1,2,3$ are rational Bernstein-Bezier weight functions such that $\sum_{j=0}^{3} R_j(\theta; v_i) = 1$

B. C^1 *Rational quadratic function*

Consider the general rational quadratic, given as Fig. 3.

$$P(t) = P_i(t) = \frac{V_i^*(1-\hat{\theta})^2 + r_i Z_i(1-\hat{\theta})\hat{\theta} + W_i^*\hat{\theta}^2}{(1-\hat{\theta})^2 + r_i(1-\hat{\theta})\hat{\theta} + \hat{\theta}^2} \tag{5}$$

where V_i^*, Z_i and W_i^* are the control points for i^{th} segment such that conic passes through V_i^* and W_i^* and the point Z_i affects the shape of the conic. r_i is the shape parameter for i^{th} segment.

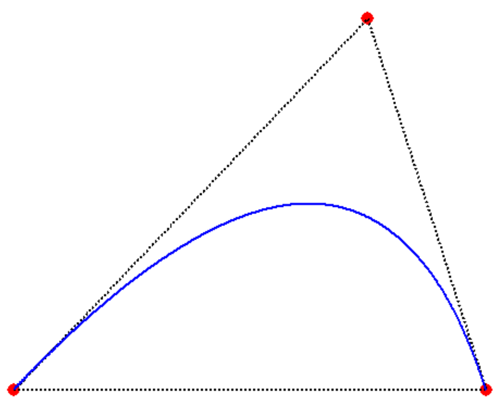


Fig. 3 Demonstration of rational quadratic function

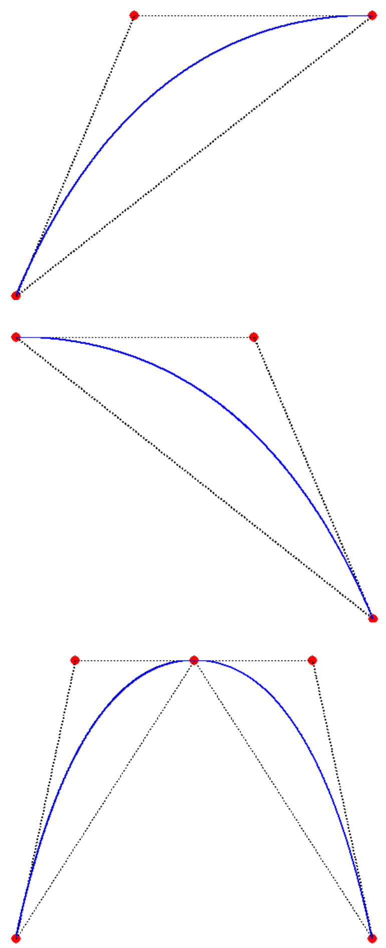
In order to have an alternate quadratic representation for cubic defined in Section 3.2, each segment of piecewise rational cubic curve should be dealt individually so that it could be represented by two segments of rational quadratic. This process will be done in a way that one conic passes through F_i and Z_i . Similarly, the other conic interpolates Z_i and F_{i+1}.

Consider the first conic which passes through F_i and Z_i and it lies in the convex hull of F_i , V_i^* and Z_i .

$$P(t) = P_i(t) = \frac{F^i(1-\theta)^2 + r_i V_i^*(1-\theta)\theta + Z_i\theta^2}{(1-\theta)^2 + r_i(1-\theta)\theta + \theta^2} \tag{6}$$

The other conic which passes through Z_i and F_{i+1} and it lies in the convex hull of F_{i+1}, W_i^* and Z_i is given by

$$P^*(t) = P_i^*(t) = \frac{Z^i(1-\theta^*)^2 + r_i W_i^*(1-\theta^*)\theta^* + F_{i+1}\theta^{*2}}{(1-\theta^*)^2 + r_i(1-\theta^*)\theta^* + \theta^{*2}} \tag{7}$$



(a)

(b)

(c)

Fig. 4 (a) demonstration of conic (6), (b) demonstration of conic (7), (c) combined view of both the conics (6) and (7).

Following properties should be satisfied by conics (6) and (7) to be C^1.

$$V_i^* = F_i + \frac{h_i D_i}{2r_i}$$

$$W_i^* = F_{i+1} - \frac{h_i D_{i+1}}{2r_i}$$

$$Z_i = \frac{V_i^* + W_i^*}{2} = \frac{F_i + F_{i+1}}{2} + \frac{h_i}{4r_i}\left(D_i - D_{i+1}\right)$$

It is to be noted that, for $r_i = 2,\ i = 1,\ldots,n,$ denominators for rational quadratic (6) and (7) become (1) and rational quadratic can be written as:

$$P(t) = P_i(t) = F^i(1-\theta)^2 + 2V_i^*(1-\theta)\theta + Z_i\theta^2 \quad (8)$$

$$P^*(t) = P_i^*(t) = Z^i(1-\theta^*)^2 + 2W_i^*(1-\theta^*)\theta^* + F_{i+1}\theta^{*2} \quad (9)$$

This is the default case of rational quadratic.

Both the conics (6) and (7) are demonstrated in Fig. 4(a) and Fig. 4(b) respectively whereas Fig. 4(c) depicts combine view of both the conics (6) and (7) . Fig. 5 represents combined view of rational cubic (1) and the conics (6), (7) for different values of r_i and v_i, it can be noted that the rational cubic (1) coincides with conics (6) and (7) if $v_i = \frac{3r_i}{2}$. Figs. 5(a) and 5(b) show default view of rational quadratic and rational cubic respectively. Fig. 5(c) gives the combine view of both the default rational functions. Fig. 5(d) , (e) and (f) represent combined view of conics for r_i = 4 and rational cubic for v_i = 6, combined view of conics for r_i = 6 and rational cubic for v_i = 9 and combined view of conics for r_i = 8 and rational cubic for v_i = 12 respectively.

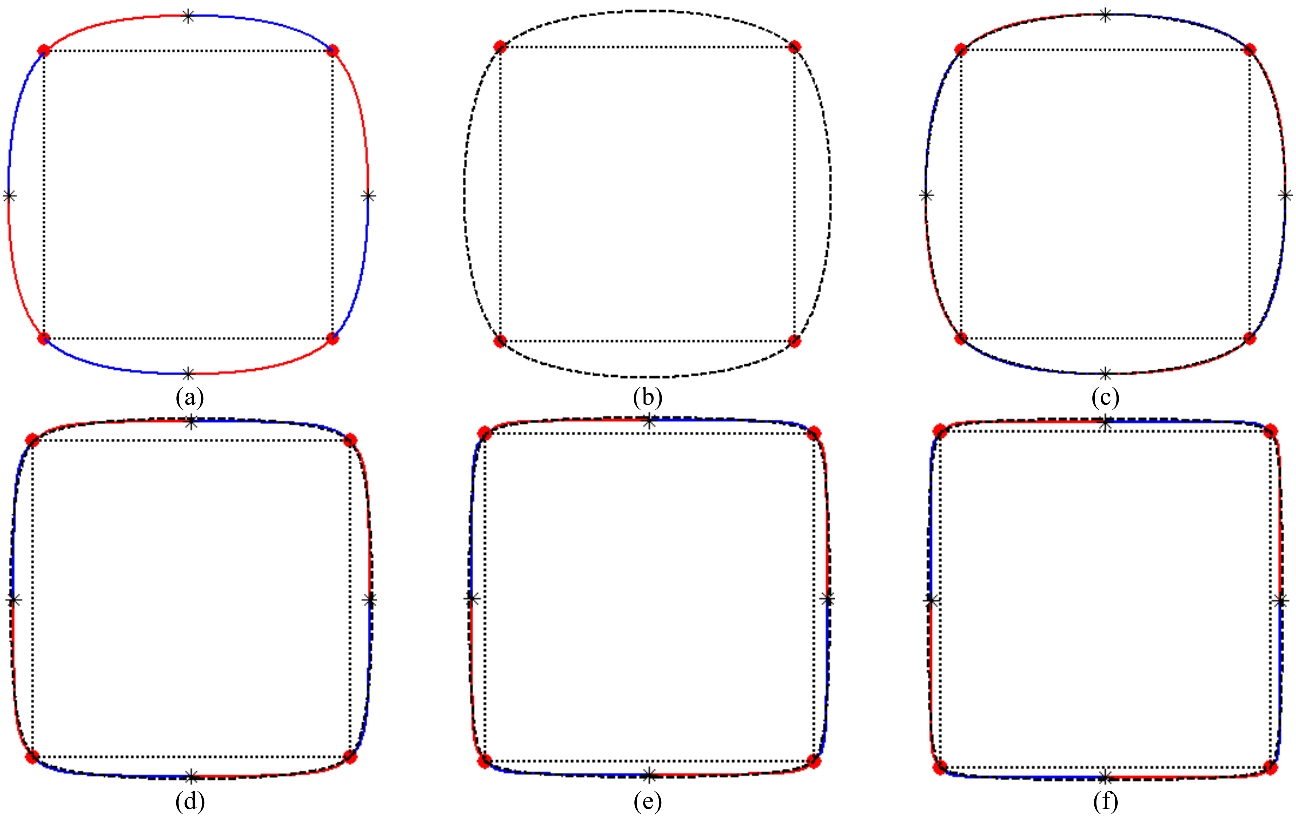


Fig. 5 (a) default view of conis, (b) default view of rational cubic, (c) combined view of conics and rational cubic for default cases, (d) combined view of conics for r_i = 4 and rational cubic for v_i = 6, (e) combined view of conics for r_i = 6 and rational cubic for v_i = 9, (f) combined view of conics for r_i = 8 and rational cubic for v_i = 12

A. Parameterization

Number of parameterization techniques can be found in literature for instance uniform parameterization, linear or chord length parameterization, parabolic parameterization and cubic parameterization. In this paper, chord length parameterization is used to estimate the parametric value t associated with each point. It can be observed that θ_i is in normalized form and varies from 0 to 1. Consequently, in our case, h_i is always equal to 1.

B. Estimation of Tangent Vectors

A distance based choice of tangent vectors D_i's at F_i's is defined as:

For open curves:

$$\left.\begin{aligned} D_0 &= 2(F_1 - F_0) - (F_2 - F_0)/2 \\ D_n &= 2(F_n - F_{n-1}) - (F_n - F_{n-2})/2 \\ D_i &= a_i(F_i - F_{i-1}) - (1-a_i)(F_{i+1} - F_i),\ i = 1,2,\ldots,n-1 \end{aligned}\right\}$$

For close curves:

$$\left.\begin{aligned} &F_{-1} = F_{n-1}, F_{n+1} = F_1 \\ &D_i = a_i(F_i - F_{i-1}) - (1-a_i)(F_{i+1} - F_i),\ i = 0,1,\ldots,n \end{aligned}\right\}$$

where

$$a_i = \frac{|F_{i+1} - F_i|}{|F_{i+1} - F_i| + |F_i - F_{i-1}|},\ i = 0,1,\ldots,n.$$

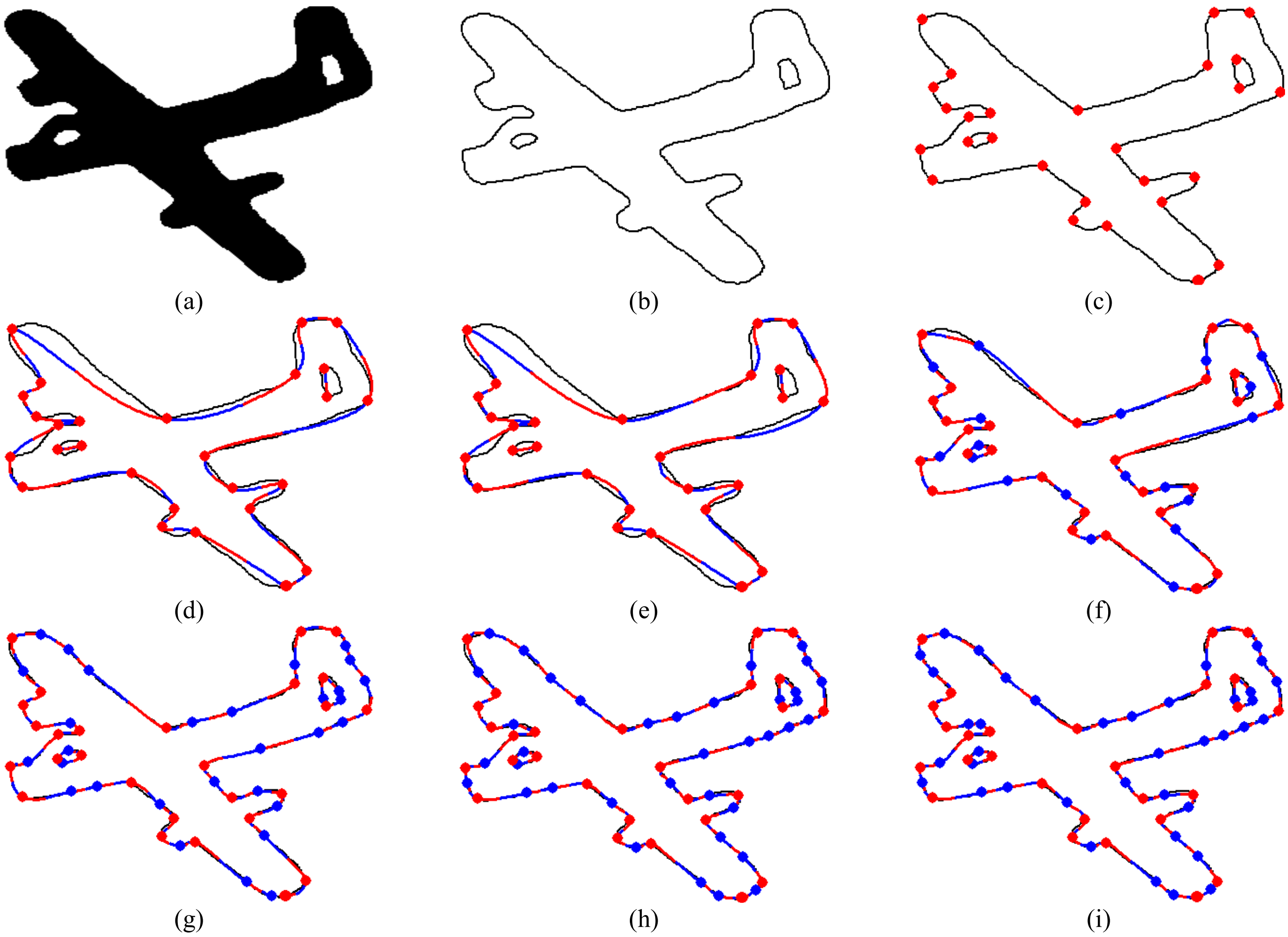


Fig. 6 Demonstration of proposed scheme (a) bitmap image of plane, (b) detected boundary of the image in (a), (c) corners detected from the boundary, (d) default conics fitted to the corners along with boundary, (e) 1st iteration of conics fitted through GA, (f) 2nd iteration of conics fitted through GA, (g) 3rd iteration of conics fitted through GA, (h) 4th iteration of conics fitted through GA, (i) 5th iteration of conics fitted through GA.

IV. Optimal Rational Quadratic Functions

This Section describes the process of evaluating optimal quadratic functions using GA [8]. Genetic Algorithm formulation of the curve fitting problem discussed in this paper is also elaborated here.

Suppose, for $i = 1,\ldots,n$, the data segments $P_{i,j} = (x_{i,j}, y_{i,j})$, $j = 1,2,\ldots,m_i$ be the given data segments. Then the squared sums S_i's of distance between $P_{i,j}$'s and their corresponding parametric points $P(t_j)$'s on the curve are determined as $S_i = \sum_{j=1}^{m_i}\left[P_i(u_{i,j}) - P_{i,j}\right]^2$, $i = 1,\ldots,n$ where u's are parameterized in reference to chord length parameterization.

Conic 1:

When conic represented by the rational quadratic (6) is considered, the squared sum S_i would be defined as

$$S_i = \sum_{j=1}^{m_i}\left[P_i(u_{i,j}) - P_{i,j}\right]^2, \quad i = 1,\ldots,n$$

where $P_i(u_{i,j})$ is defined as in (6).

Conic 2:

Similarly for conic represented by the rational quadratic (7), the squared sum S_i would be defined as:

$$S_i^* = \sum_{j=1}^{m_i} \left[P_i^*(u_{i,j}) - P_{i,j} \right]^2, \quad i = 1, \ldots, n$$

where $P_i^*(u_{i,j})$ is defined as in (7).

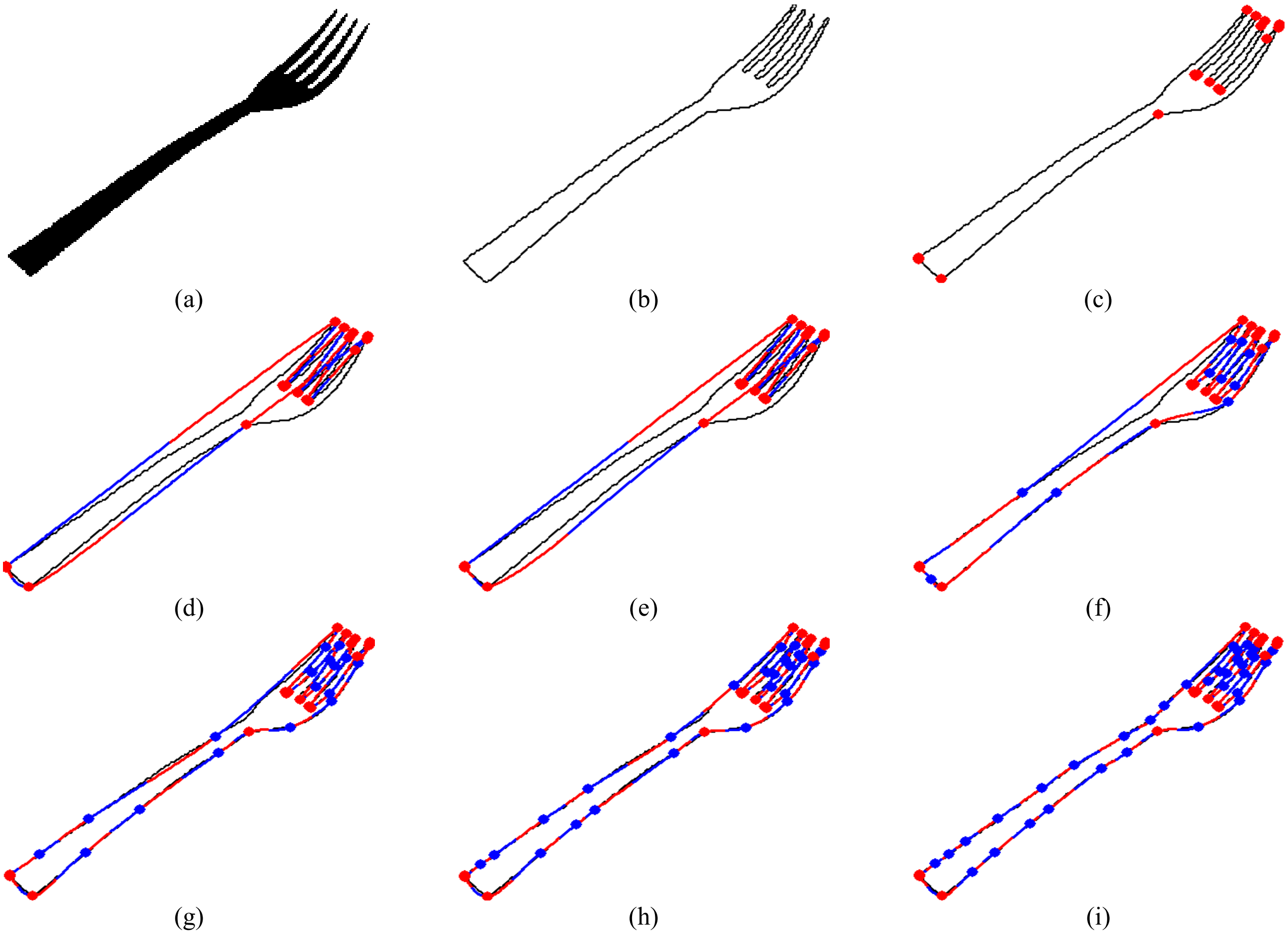


Fig. 7 Demonstration of proposed scheme (a) bitmap image of fork, (b) detected boundary of the image in (a), (c) corners detected from the boundary, (d) default conics fitted to the corners along with boundary, (e) 1st iteration of conics fitted through GA, (f) 2nd iteration of conics fitted through GA, (g) 3rd iteration of conics fitted through GA, (h) 4th iteration of conics fitted through GA, (i) 6th iteration of conics fitted through GA.

Table 2. Number of corner points together with number of intermediate points for iterations 1, 2 and 3 of GA.

Image Name	**# of initial corner points**	**# of intermediate points in cubic interpolation with threshold value 3**				**Total time elapsed**
		Itr.1	Itr.2	Itr.3	Final itr.	
Fork	15	0	10	19	35	6.075 sec
Plane	28	0	19	30	39	6.7 sec
Fish	31	0	17	29	35	9.395 sec

Now to find optimal curve to given data, such values of parameters r_i's, are required so that the sums S_i's are minimal. Genetic Algorithm is used to optimize this value for the fitted curve. Randomly chosen values of r_i are needed to initialize the process. Successive application of search operations like selection, crossover and mutation to this population gives optimal values of r_i's.

A. Initializtion

Once we have the bitmap image of a character, the boundary of the image can be extracted using the method described in Section 2. After the boundary points of the image are found, the next step is to detect corner points to divide the boundary of the image into n segments as explained in Section 2. Each of these segments is then approximated by interpolating quadratic functions (6) and (7) described in Section 3.2.

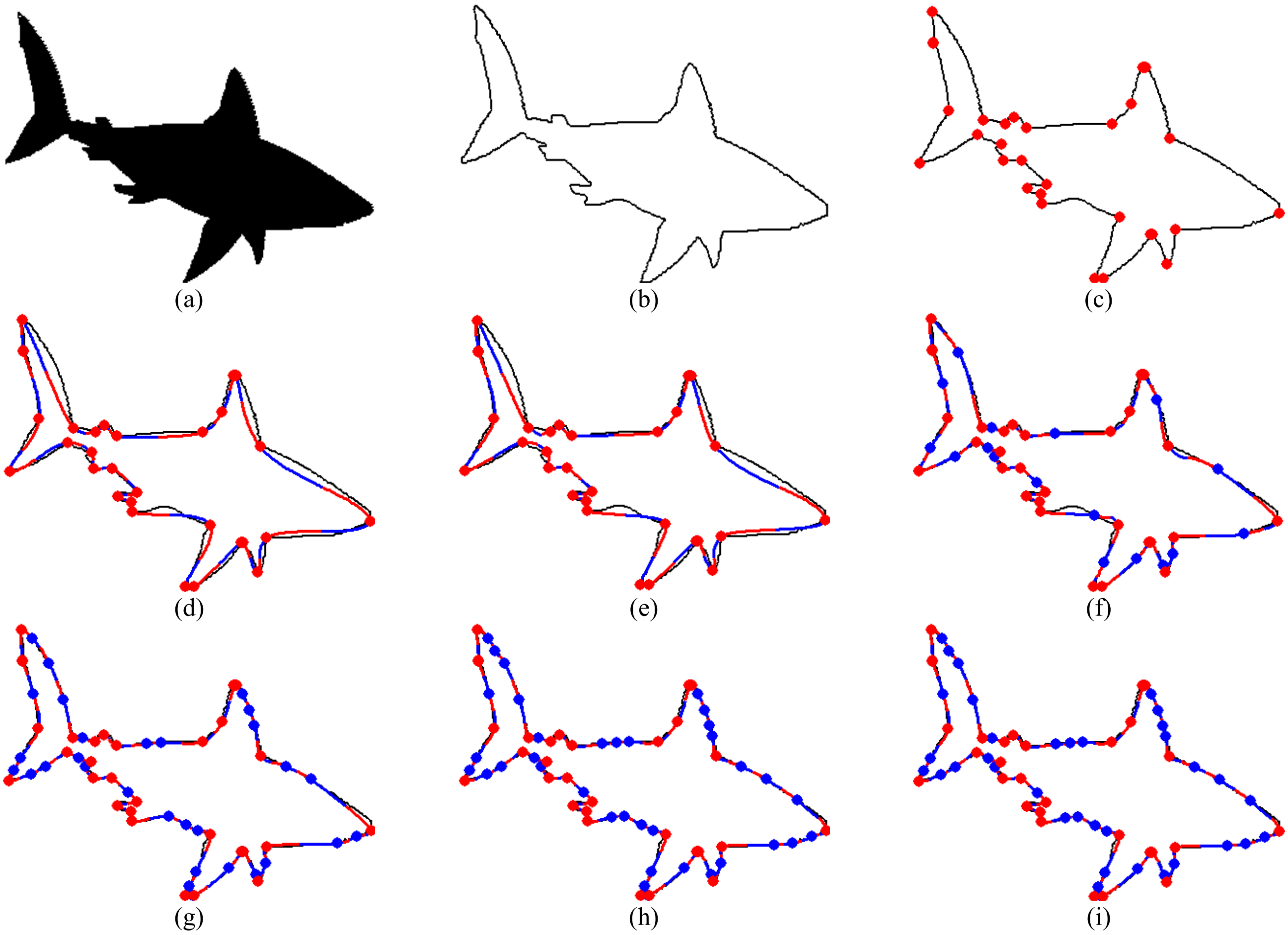


Fig. 8 Demonstration of proposed scheme (a) bitmap image of fish, (b) detected boundary of the image in (a), (c) corners detected from the boundary, (d) default conics fitted to the corners along with boundary, (e) 1st iteration of conics fitted through GA, (f) 2nd iteration of conics fitted through GA, (g) 3rd iteration of conics fitted through GA, (h) 4th iteration of conics fitted through GA, (i) 5th iteration of conics fitted through GA.

B. Curve Fitting

After an initial approximation for the segment is obtained, Genetic Algorithm helps to obtain better approximations to achieve optimal solution. The tangent vectors at knots are estimated by the method described in Section 3.4.

C. Breaking Segment

For some segments, the best fit obtained through iterative improvement, may not be satisfactory. In that case, we subdivide the segment into smaller segments at points where the distance between the boundary and parametric curve exceeds some predefined threshold. Such points are termed as *intermediate points*. A new parametric curve is fitted for each new segment as shown in Figs. 6(f-h), Figs. 7(f-h) and Figs. 8(f-h). Table 2 gives details of number of intermediate points achieved during different iteration of Genetic Algorithm applied in process of curve fitting.

D. Algorithm

All the steps of computing the desired outline curve manipulation can be summarized into the following algorithm:

Step 1. Input the data points.
Step 2. Subdivide the data, by detecting corner points using the method mentioned in Section 2.
Step 3. Compute the derivative values at the corner points by using formula given in Section 3.4.
Step 4. Fit the rational quadratic functions, of Section 3, to the corner points found in Step 2.
Step 5. If the curve, achieved in Step 4, is optimal then go to Step 7, else find the appropriate break/intermediate points (points with highest deviation) in the undesired curve pieces. Compute

the best optimal values of the shape parameters r_i's. Fit rational quadratic functions in Section 3 to these intermediate points.

Step 6. If the curve, achieved in Step 5, is optimal then go to Step 7, else add more intermediate points (points with highest deviation) and go to Step 3.

Step 7. Stop.

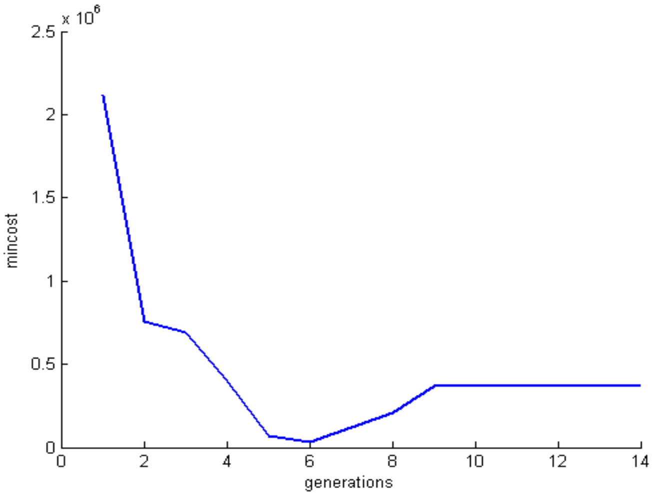


Fig. 9 Graph of minimum cost for 'plane' showing mixed behaviour .

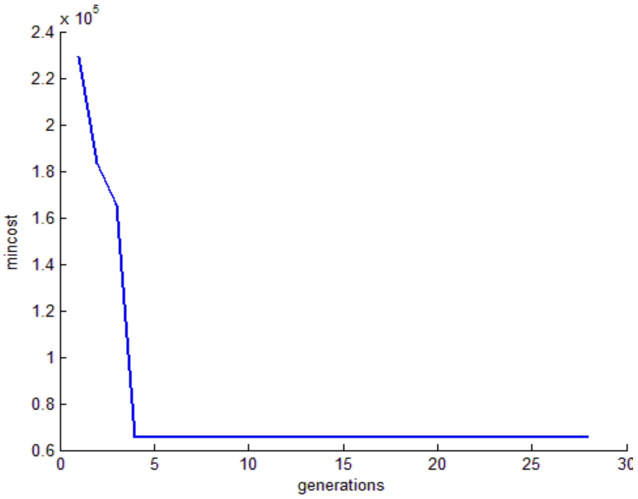


Fig. 10 Graph of minimum cost for 'plane' showing decreasing behaviour.

V. DEMONSTRATION

Curve fitting scheme, proposed in Section 4, has been implemented on different images of a plane (Fig. 6(a)), a fork (Fig. 7(a)), and a Fish (Fig. 8(a)). In Fig. 6 ((a) represents original image, (b) shows outline of the image, (c) demonstrates corner points, (d) presents fitted Hermite curve to the corners along with boundary of the image, (e) gives fitted outline to the corners for 1st iteration using Genetic Algorithm together with corner points and boundary, (f), (g), (h) and (i) depict fitted outline for 2nd , 3rd, 4th and final (5th) iterations respectively using Genetic Algorithm together with corner points, breakpoints and boundary.

Similarly, the automatic algorithm has been implemented on the Fork Fig. 7(a) to produce Figs. 7(b-i) in a similar manner as those in Fig. 6. However, the last iteration in Fig.7(i) has appeared to be the 6th one in this case.

The Fish image, in Fig, 8(a), has also been attempted for the algorithm implementation. The output appeared, in Figs. 8(b-i), in a similar way as in Figs. 6(b-i).

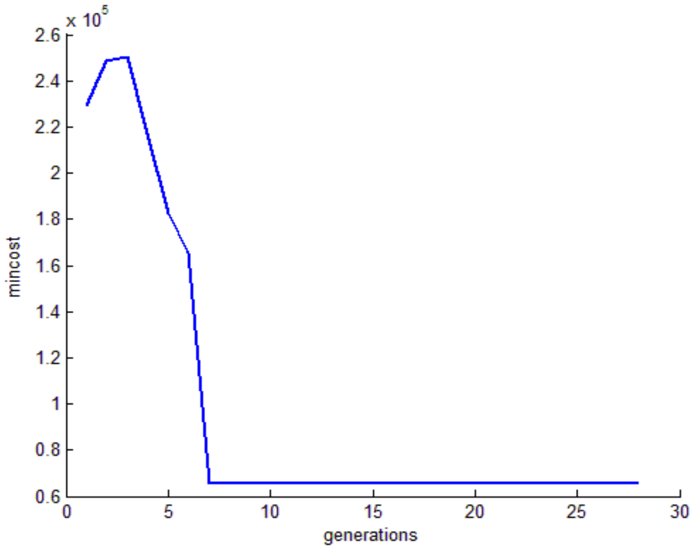


Fig. 11 Graph of minimum cost for 'plane' showing mix behaviour.

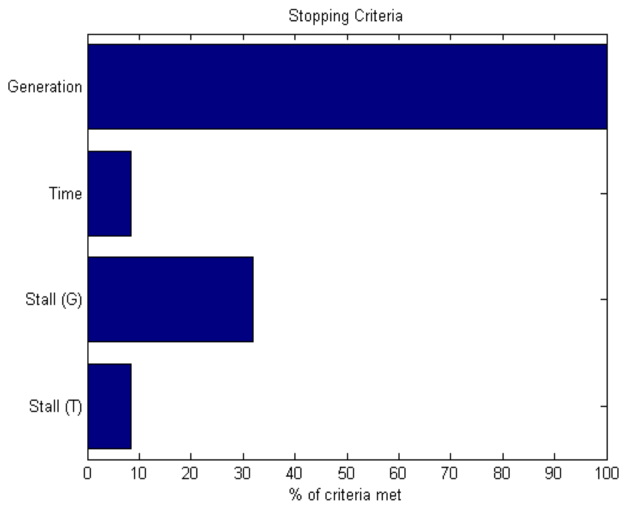


Fig. 12 Stopping crieteria met by GA for image of plane.

Some analytical study has also been made for the performance of the devised algorithm. Fig. 9, Fig. 10 and Fig. 11 represent minimum cost during different generations of GA for the image of plane. Figs. 12-13 give the percentage of stopping criteria met by GA for the images of plane and fish respectively and the parameters used while applying GA are given in Table 3.

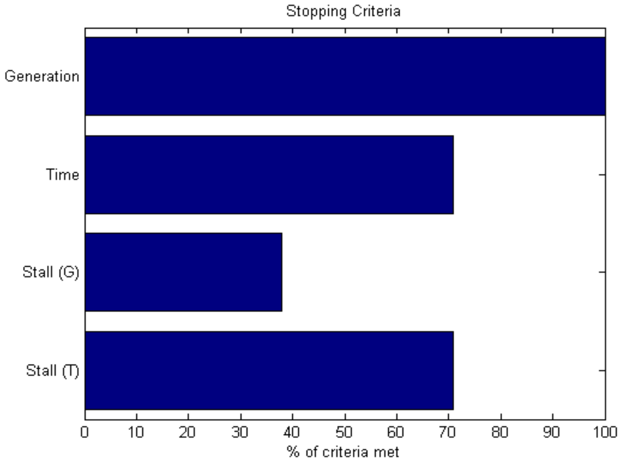


Fig. 13 Stopping crieteria met by GA for image of fish.

Table 3. Parameters of GA.

Sr. No.	Name	Values
1	Population size	25
2	Genome length	15
3	Selection rate	0.5
4	Mutation rate	0.01

VI. Conclusion

The reverse engineering technique of planar objects is presented which uses conics for curve fitting and genetic algorithm to find the optimal values of parameters in the description of conics. Two rational quadratic functions are implemented in the replacement of a rational cubic. Initial random population of parameters is required for the proposed scheme to get started and then the algorithm assures the values of parameters which provide the optimal fit to the boundary of the bitmap images of planar shapes.

The authors are interested to proceed further and extend the scheme to vectorize 3D shapes. This work is in progress with the authors.

References

[1] G. Avrahami, V. Pratt, "Sub-pixel edge detection in character digitisation," *Raster Imaging and Digital Typography II*, 1991, pp. 54–64.
[2] H. L. Beus, S. S. H. Tiu, "An improved corner detection algorithm based on chain coded plane curves," *Pattern Recognition*, vol. 20, 1987, pp. 291-296.
[3] A. C. Cabrelli, U. M. Molter, "Automatic representation of binary images," *IEEE Transaction on Pattern Analysis and Machine Intelligence*". vol. 12, no. 12, 1990, pp. 1190–1196.
[4] M. Chandrasekaran, M. Muralidhar, M. Krishna, U. S. Dixit, "Application of soft computing techniques in machining performance prediction and optimization: a literature review," *The International Journal of Advanced Manufacturing Technology*, vol. 46 , 2010, pp. 445–464
[5] D. Chetrikov, S. Zsabo, "A simple and efficient algorithm for detection of high curvature points in planar curves," Proceedings of the 23rd Workshop of Austrian Pattern Recognition Group, 1999, pp. 175–184.
[6] L. Davis, "Shape matching using relaxation techniques," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 1, No. 1, 1979, p. 60–72.
[7] H. Freeman, "On the encoding of arbitrary geometric configurations," *IEEE Transactions on Electronic Computer*, vol. 10, no. 2, 1961,pp. 260–268.
[8] D. E. Goldberg, "Genetic algorithms in search optimization and machine learning," *Addison Wesley, Reading, MA.*, 1989.
[9] G. E. Hölzle, "Knot placement for piecewise polynomial approximation of curves," *Computer Aided Design*, vol. 15, no. 5, 1983, pp. 295–296.
[10] Z. J. Hou, G. W. Wei, "A new approach to edge detection," *Pattern Recognition*, vol. 35, no. 7, 2002, pp. 1559–1570.
[11] K. Itoh, Y. Ohno, "A curve fitting algorithm for character fonts," *Electronic Publishing*, vol. 6, no. 3, 1993, pp. 195–198.
[12] S. Kirkpatrick, Jr. C. D. Gelatt, M. P. Vecchi, "Optimization by Simulated Annealing," *Science,* vol. 220, no. 4598, 1983, pp. 671-680.
[13] M. Plass, M. Stone, "Curve-fitting with piecewise parametric cubics," *Computer Graphics*, vol. 17, no. 3, 1983, pp. 229–239.
[14] M. Sarfraz, M. A. Khan, "An automatic algorithm for approximating boundary of bitmap characters," *Future Generation Computing Systems*, vol. 20, no. 8, (2004), pp. 1327–1336.
[15] M. Sarfraz, "Some algorithms for curve design and automatic outline capturing of images," *International Journal of Image and Graphics* vol. 4, no. 2, 2004, pp. 301–324.
[16] M. Sarfraz, A. Rasheed, "A Randomized Knot Insertion Algorithm for Outline Capture of Planar Images using Cubic Spline," The Proceedings of The 22th ACM Symposium on Applied Computing, Seoul, Korea, March 11 – 15, 2007, pp. 71 – 75.
[17] M. Sarfraz, M. N. Haque, M. A. Khan, "Capturing outlines of 2D Images," Proceedings of international conference on imaging science, systems, and technology, Las Vegas, Nevada, USA, June 26-29, 2000, pp. 87–93.
[18] M. Sarfraz, M. F. A. Razzak, "An algorithm for automatic capturing of font outlines," *Computers and Graphics*, vol. 26, no. 5, 2002, pp. 795–804.
[19] M. Sarfraz, M. Khan, "Towards automation of capturing outlines of Arabic fonts," Proceedings of the third KFUPM workshop on information and computer science: software development for the new millennium, Saudi Arabia, October 22 – 23, 2000, pp. 83–98.
[20] M. Sarfraz, M. A. Khan, "Automatic outline capture of Arabic fonts," *Information Sciences*, vol. 140, no. 3–4, 2002, pp. 269–281.
[21] M. Sarfraz, M. A. Khan, "An automatic algorithm for approximating boundary of bitmap characters," *Future Generation Computer Systems*, vol. 20, 2004, pp. 1327–1336.
[22] M. Sarfraz, M. R. Asim, A. Masood, "Capturing outlines using cubic Be´zier curves," Proceedings of the IEEE international conference on information and communication technologies: from theory to application, April 19-23, 2004, pp. 539–540.
[23] M. Sarfraz, S. A. Raza, "Capturing outline of fonts using genetic algorithm and splines," The proceedings of IEEE international conference on information visualization-IV-UK, July 25-27, 2001, pp. 738–743.
[24] M. Sarfraz, M. Riyazuddin, M. H. Baig, "Capturing planar shapes by approximating their outlines," *Journal of Computational and Applied Mathematics*, vol. 189, 2005, pp. 494–512.
[25] M. Sarfraz, S. A. Raza, "Visualization of data with spline fitting: a tool with a genetic approach," Proceedings of international conference on imaging science, systems, and technology, Las Vegas, Nevada, USA, June 24-27, 2002, p. 99–105.
[26] M. Sarfraz, A. Masood, M. R. Asim, "A new approach to corner detection," *Computer vision and graphics*, vol. 32, 2006, p. 528–533.
[27] P. J. Schneider, "An algorithm for automatically fitting digitized curves," *Graphics Gems,* 1990, pp. 612–626.

[28] F. A. Sohel, G. C. Karmakar, L. S. Dooley, “Arkinstall J. Enhanced Be´zier curve models incorporating local information,” Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, USA, March 18-23, 2005, pp. 253-256.

[29] Y. Y. Tang, F. Yang, J. Liu, “Basic processes of chinese character based on cubic B-spline wavelet transform,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 12, 2001, p. 1443-1448.

[30] C. H. Teh, R. T. Chin, “On the detection of dominant points on digital curves,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 8, 1989, pp. 859–873.

[31] L. A. Zadeh, “Fuzzy Logic and Soft Computing: Issues, Contentions and Perspectives,” International Conference on Fuzzy Logic, Neural Nets and Soft Computing, Iizuka, Japan, 1994, pp. 1-2.

M. Sarfraz is a Professor in Kuwait University, Kuwait. He received his Ph.D. from Brunel University, UK, in 1990. His research interests include Computer Graphics, CAD/CAM, Pattern Recognition, Computer Vision, Image Processing, and Soft Computing. He is currently working on various projects related to academia and industry.

He has been keynote/invited speaker at various platforms around the globe. He has advised/supervised more than 50 students for their MSc and PhD theses. He is the Chair of the Information Science Department, Kuwait University. He has published more than 250 publications in the form of various Books, Book Chapters, journal papers and conference papers.

He is member of various professional societies including IEEE, ACM, IVS, IACSIT, and ISOSS. He is a Chair, member of the International Advisory Committees and Organizing Committees of various international conferences, Symposiums and Workshops. He is the reviewer, for many international Journals, Conferences, meetings, and workshops around the world. He is the Editor-in-Chief of the International Journal of Computer Vision and Image Processing. He is also Editor/Guest Editor of various International Conference Proceedings, Books, and Journals. He has achieved various awards in education, research, and administrative services.

M. Z. Hussain is a Professor in Punjab University, Lahore, Pakistan. He received his Ph.D. from Punjab University, Pakistan, in 2002. His research interests include Computational Mathematics, Computer Graphics, CAGD, CAD/CAM, Image Processing and Soft Computing. He has advised/supervised more than 30 students for their M.Phil. and PhD theses. He has published more than 50 publications in the form of journal and conference papers.

M. Irshad is a Ph.D. student in Punjab University, Lahore, Pakistan. She received her M.Phil. from Punjab University, Pakistan, in 2006. Her research interests include Computational Mathematics, Computer Graphics, CAGD, CAD/CAM, Image Processing, and Soft Computing. She has published more than 6 publications in the form of journal and conference papers.