Generating Outlines of Generic Shapes by mining Feature Points

MUHAMMAD SARFRAZ
Department of Information Science
Kuwait University
Adailiya Campus, P.O. Box 5969, Safat 13060
KUWAIT
prof.m.sarfraz@gmail.com

Abstract: - An approach, based on conic splines, is proposed here for capturing image outlines of generic shapes. It has various phases including extracting outlines of images, mining feature points from the detected outlines, and curve fitting. The idea of particle swarm optimization has been incorporated to optimize the shape parameters in the description of the conic spline. The method ultimately produces optimal results for the approximate vectorization of the digital contours obtained from the generic shapes. Demonstrations also make the essential part of the paper.

Keywords: Imaging; optimization; particle swarm optimization; generic shapes; curve fitting

1 Introduction

Capturing outlines of images is one of the important problems of computer graphics, vision, and imaging. Various mathematical and computational phases are involved in the whole process. Curve modelling [4-6] plays significant role in it. The representation of planar objects in terms of curves has many advantages. For example, scaling, shearing, translation, rotation and clipping operations can be performed without any difficulty. Although a good amount of work has been done in the area [2-3], it is still desired to proceed further to explore more advanced and interactive strategies.

This work is inspired by an optimization algorithm based on particle swarm optimization (PSO) [9-11]. It motivates the author to introduce a robust technique for the outline capture of generic shapes. The algorithm comprises of various phases to achieve the target. First of all, it finds the contour of the grey scaled bitmap image [7-8]. Secondly, it uses the idea of corner points [1, 17-23] to detect corners. These phases are considered as pre-processing steps. The next phase detects the corner points on the digital contour of the generic shape under consideration. The idea of Particle swarm optimization (PSO) is then used to fit a conic spline which passes through the corner points. It globally optimizes the shape parameters in the description of the conic spline to provide a good approximation to the digital curve.

2 Preprocessing

The proposed scheme starts with first finding the boundary of the generic shape and then using the output to find the corner points or the significant points. Forthcoming Sections 2.1 and 2.2 will explain these phases.

2.1. Finding Boundary of Generic Shapes

The image of the generic shape can be acquired either by scanning or by some other mean. The quality of scanned images is dependent upon factors such as paper quality and scanning resolution. For a better resolution and paper quality, one can achieve a better image.

The aim of boundary detection is to produce an object's shape in graphical or non-scalar representation. Chain codes [12-16] are the most widely used representations. Other well known representations are syntactic techniques, boundary approximations and scale-space techniques. The benefit of using chain code is that it gives the direction of edges. The boundary points are selected as contour points based on their corner strength and fluctuations.

Chain codes were initially proposed by Freeman [12-16]. The methodology adopted to detect the boundary is by encoding the shape boundary as a sequence of connected line segments of specified length and direction. The direction of a segment is coded using either 4-connected or 8-connected schemes. In both schemes, initially a point is

ISBN: 978-960-474-336-0

selected using either horizontal or vertical scan. After this, the 4-connected or 8-connected component algorithm is applied. Both algorithms work in intensive stack formulation. In case of 4-connected, four neighboring points are analyzed. These points are pixel positions that are right, left, above and below the current pixel. The second method is a little more complex. In this method the set of neighboring positions to be tested include the four diagonal pixels as well. The point set obtained after this step is known as contour of the object.

We simply convert a grayscale image to binary after normalizing the intensity in the range [0, 1]. The image is then converted to binary at a specified threshold. If the threshold is 0.4, pixel intensity with less than 0.4 is white pixel and others are black pixels. Outline extraction from a binary image is a simple procedure. Any pixel with a pixel value 0 (black) is a boundary point if any of its four neighbors has a pixel value 1 (white). The four neighboring pixels are upper, lower, left and right pixels. This procedure will extract all boundary points (inner or outer boundary points) from the binary image. At this point it is difficult to distinguish that which boundary point belongs to which boundary loop (in case of more than one boundary loops) and also the sequence of boundary points (clockwise) around a loop is unknown.

To arrange the extracted boundary points in a sequence (clockwise direction), a boundary tracing is performed, starting from the left-top boundary pixel. The algorithm [25] for boundary tracing is as follows:

1. Search the top left boundary point; this point P_0 has the minimum column and row value of all the boundary points. Point P_0 is the starting point of the boundary tracing. Define a variable *dir* using Freeman's chain code [7] as shown in Fig. 1. It stores the direction of the previous move along the boundary from previous point to the current point. Assign *dir* = 3.
2. Search the 3x3 neighborhood of the current point in an anti-clockwise direction as shown in Fig. 2, Starting the neighborhood search in the direction given below:

   a. (*dir* + 7) mod 8 if *dir* is even.
   b. (*dir* + 6) mod 8 if *dir* is odd.

   Update the *dir* value as per new point found.
3. If the current boundary point P_n is equal to the point P_0 then stop. Otherwise repeat step 2.
4. The detected boundary points are represented by points P_0 . . . . P_{n-1}. This makes a one loop of boundary points.
5. Delete the detected boundary points P_0 . . . . P_{n-1} from the list of extracted boundary points and repeat Steps 1 to 4 for other boundary loops till all boundary loops have been traced.

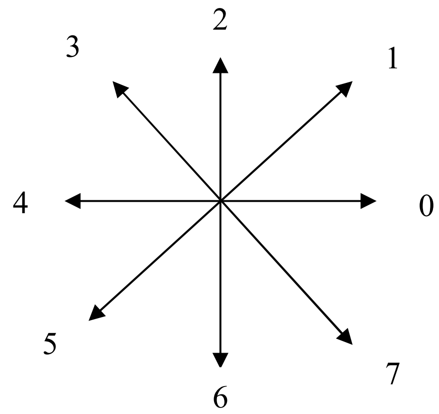


Figure 1.Freeman's chain code.

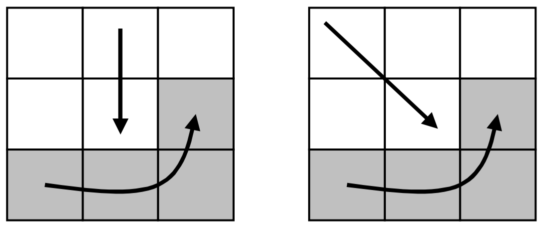


Figure 2.Neighborhood search sequence.

Demonstration of the method can be seen in Fig. 3(b) which is the contour of the bitmap image shown in Fig. 3(a).

2.2. Detecting Corner Points

Corners in digital images give important clues for the shape representation and analysis. Generally objects information can be represented in terms of its corners, which play a very vital role in object recognition, shape representation and image interpretation [1, 16]. These are the points that partition the boundary into various segments. The strategy of getting these points is based on the

ISBN: 978-960-474-336-0

method proposed in [1, 16]. The details of this procedure are as follows.

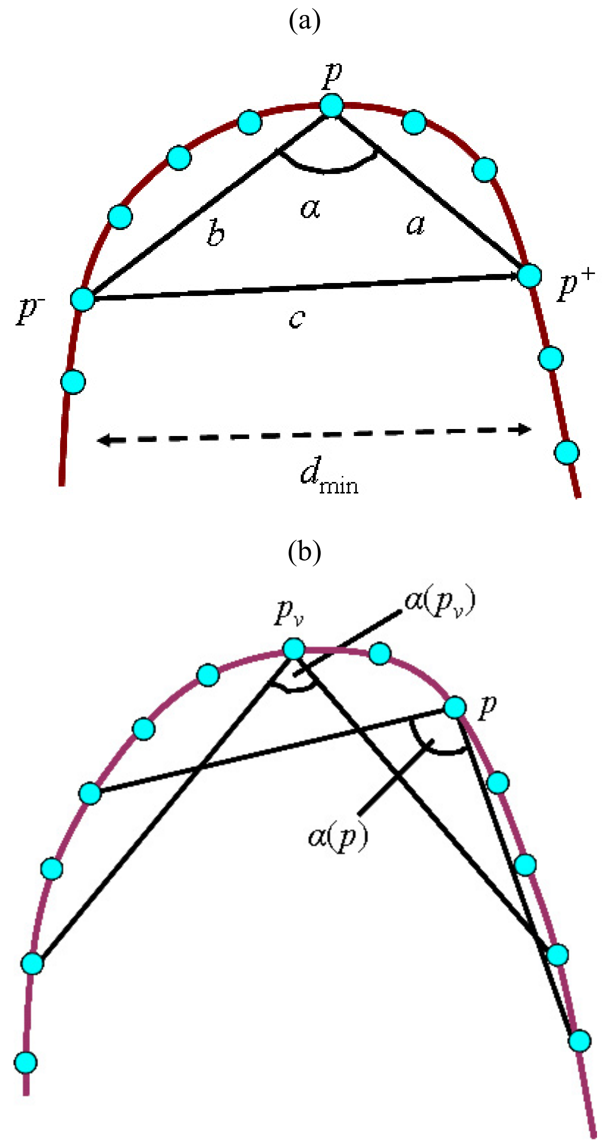


Figure 3. Corner point detection demonstration: (a) First pass: test p for candidate corner point, (b) Second pass: discard non-maxima points.

The corner detector used, assesses the curvature of points and picks those with high values. The algorithm has two passes. A variable triangle (p^-, p, p^+) is inscribed in the first pass as shown in Fig.3(a). This is constrained by the following rules:

$$d_{\min}^2 \le \left|p - p^+\right|^2 \le d_{\max}^2 , \tag{1}$$

$$d_{\min}^2 \le \left|p - p^-\right|^2 \le d_{\max}^2 , \tag{2}$$

$$\alpha \le \alpha_{\max} , \tag{3}$$

where $\left|p - p^+\right| = |a| = a$ is the distance between p and $p+$ and similarly $\left|p - p^-\right| = |b| = b$ is the distance between p and p^-. If $\left|p^+ - p^-\right| = |c| = c$ is the distance between p^+ and p^-. The opening angle of the triangle α is computed as follows:

$$\alpha = \cos^{-1}\left(\frac{a^2 + b^2 - c^2}{2ab}\right).$$

Adjacent points can be detected in the first pass. The second pass is used to discard the non-maxima points. A candidate point p is discarded if it has a sharper valid neighbor p_v such that $\alpha(p) > \alpha(p_v)$. The validity of the candidate point in the neighborhood of p is legitimate if $\left|p - p_v\right|^2 \le d_{\min}^2$ or if it is adjacent to p. This is depicted in Fig.3(b). The demonstration of the algorithm is made on Fig.4(b). The corner points of the image are shown in Figure 4(c).

3 Curve Fitting and Spline

The curve fitted, to the corner points, by a conic spline is a candidate of best fit, but it may not be a desired fit. This leads to the need of introducing some shape parameters in the description of the conic spline. This section deals with a form of conic spline. It introduces shape parameters u's in the description of conic spline defined as follows:

$$P(t) = \frac{P_i(1-\theta)^2 + u_i U_i 2\theta(1-\theta) + P_{i+1}\theta^2}{(1-\theta)^2 + 2u_i\theta(1-\theta) + \theta^2},$$

where

$$\theta\big|_{[t_i,t_{i+1})}(t) = \frac{(t - t_i)}{h_i},\ h_i = t_{i+1} - t_i,\ U_i = \frac{(V_i + W_i)}{2},$$

and

$$V_i = P_i + \frac{h_i D_i}{u_i},\ W_i = P_{i+1} - \frac{h_i D_{i+1}}{u_i}.$$

Here P_i and P_{i+1} are feature corner points of the i^{th} piece of the digital contour. D_i and D_{i+1} are the corresponding tangents at featurecorner points.

ISBN: 978-960-474-336-0

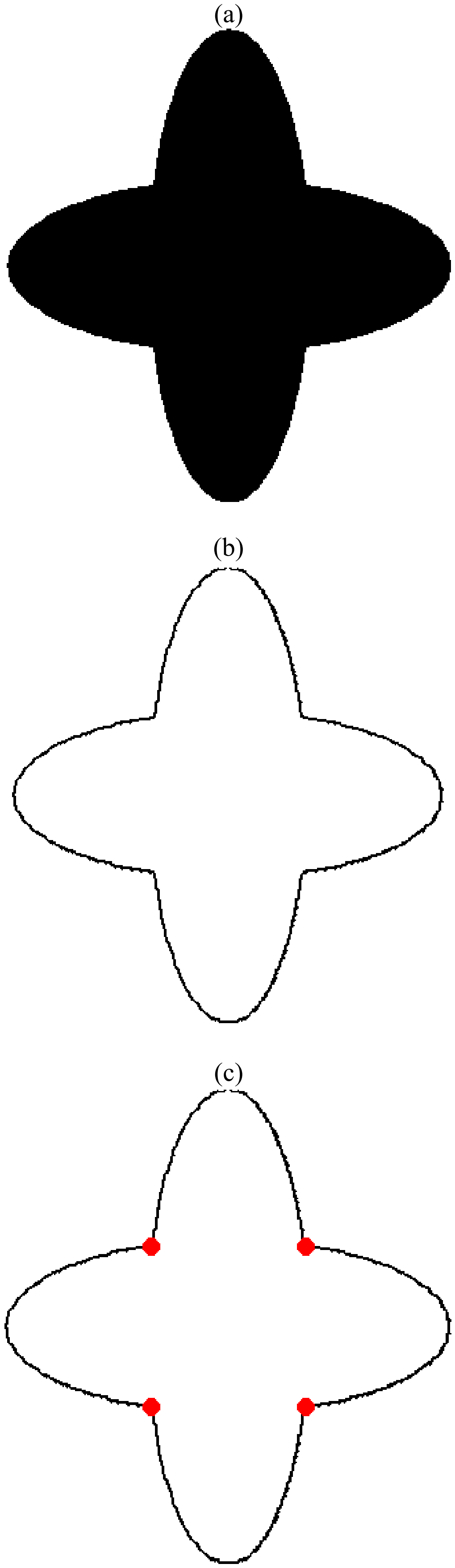


Figure 4. Pre-processing Steps: (a) Original Image, (b) Outline of the image, (c) Feature points mined.

Obviously, the parameters u_i's, when equal to 1, provide the special case of quadraticic spline. Otherwise, these parameters can be used to loose or tight the curve. This paper proposes an evolutionary technique, namely particle swarm optimization (PSO), to optimize these parameters so that the curve fitted is optimal.

To construct the parametric conic spline interpolant on the interval $[t_0, t_n]$, we have $F_i \in R^m$, $i = 0,1,......,n$, as interpolation data, at knots t_i, $i = 0,1,......,n$. The tangent vectors D_i's can be calculated using some appropriate method like arithmetic means.

Since, the objective of the paper is to come up with an optimal technique which can provide a decent curve fit to the digital data. Therefore, the interest would be to compute the curve in such a way that the sum square error of the computed curve with the actual curve (digitized contour) is minimized.

4 Particle Swarm Optimization

A novel population based optimization approach, called Particle Swarm Optimization (PSO) approach, has been used in this paper. PSO was introduced first in 1995 by Eberhart and Kennedy [9-11]. This new approach features many advantages; it is simple, fast and can be coded in few lines. Also, its storage requirement is minimal. Our interest is to optimize the values of curve parameters u such that the defined curve fits as close to the original contour segments as possible. Note that we apply PSO independently for each segment of a contour that we have identified using corner points. PSO is applied sequentially on each of the segments, generating an optimized fitted curve for each segment. The algorithm is run until the maximum allowed time is reached, or an optimal curve fitting is attained. Derivation of the PSO algorithm and underlying theory can be found in [9-11].

5 Proposed Approach

Once we have the bitmap image of a generic shape, 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 as explained in Section 2. This corner detection technique assigns a measure of 'corner strength' to each of the points on the boundary of the image. This step helps to divide the boundary of the image into n segments. Each of these segments is then approximated by interpolating spline described in Section 3. The initial solution of spline parameters (u) are randomly selected within the range [-1, 1].

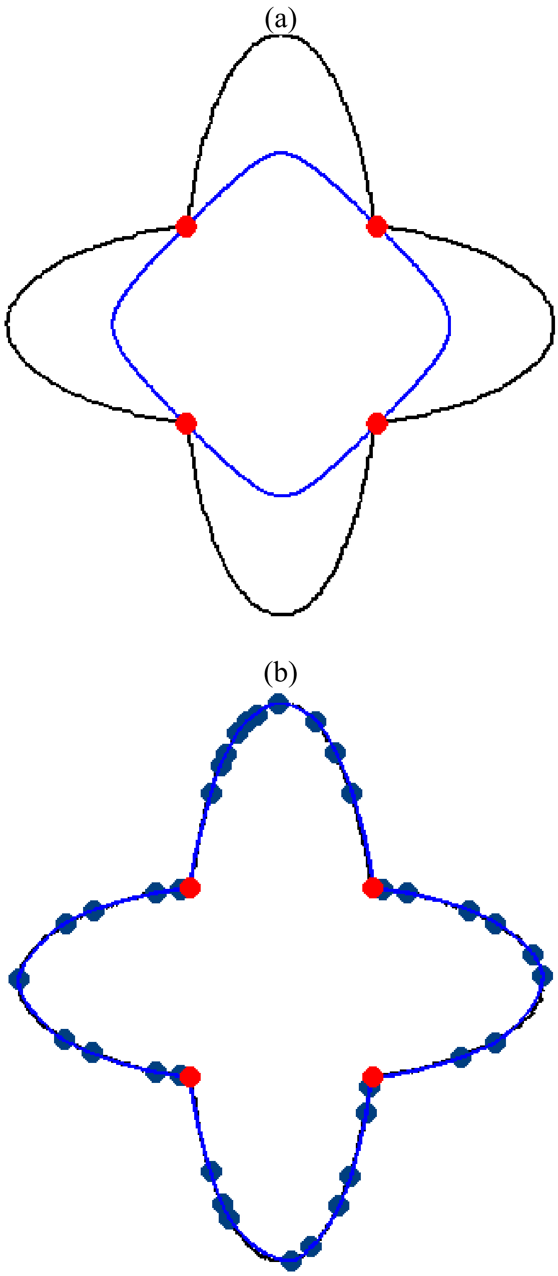


Figure 5. Results of conic scheme: (a) Fitted Outline of the image, (b) Fitted Outline of the image with intermediate points.

After an initial approximation for the segment is obtained, better approximations are obtained through PSO to reach the optimal solution. We experiment with our system by approximating each segment of the boundary using the generalized conic spline of Section 3. Each boundary segment is approximated by the spline. The shape parameters (u) in the conic spline provide greater flexibility over the shape of the curve. These parameters are adjusted using PSO to get the optimal fit.

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.

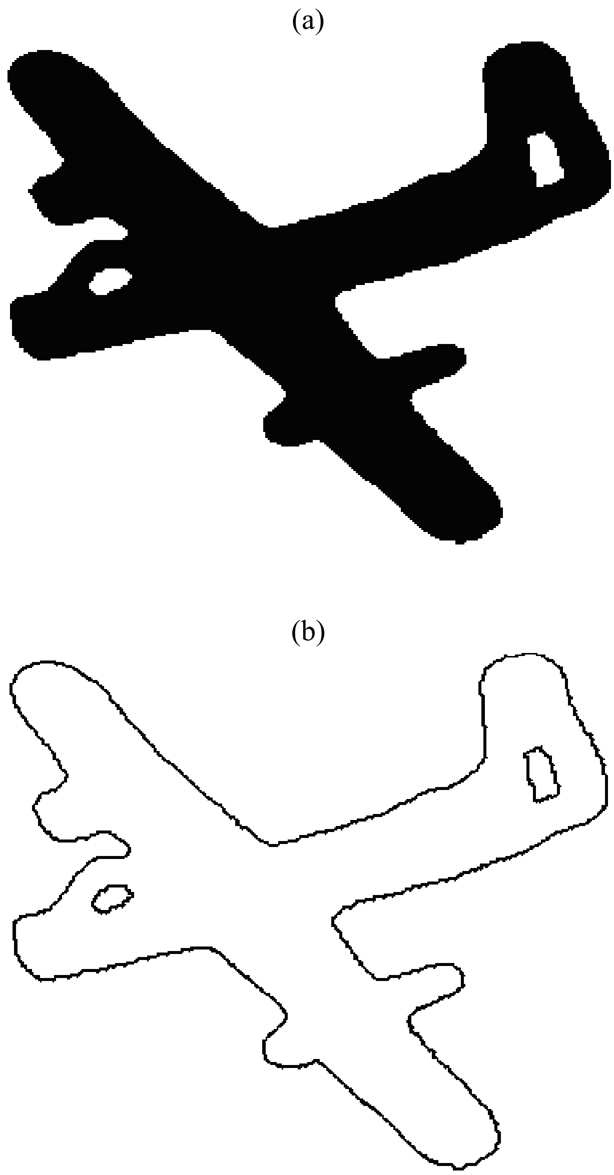


Figure 6. Pre-processing Steps: (a) Original Image, (b) Outline of the image, (c) Corner points achieved, (d) Fitted Outline of the image.

ISBN: 978-960-474-336-0

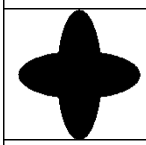
6 Demonstration

The methodology, in Section 5, has been implemented practically and the proposed curve scheme has been implemented successfully. We evaluate the performance of the system by fitting parametric curves to different binary images. Two images “ellipses” and “airplane” have been chosen to test the proposed scheme based upon the PSO. The image of “ellipses” is relatively simple, whereas the image of “airplane” is somewhat complex.

Fig.5(a) shows the implementation results of the algorithm with PSO for the original image of an image “ellipses” in Fig.4. One can see that the approximation is not satisfactory, this is specifically due to those segments which are bigger in size and highly curvy in nature. Thus, some more treatment is required for such outlines. One of the idea is to insert some intermediate points, this is demonstrated in Fig.5(b) where excellent result has been achieved. The idea of how to insert intermediate points is not explained here due to limitation of space. It will be explained in a subsequent paper.

Fig.6(a) shows another image of an airplane. Its outline is shown in Fig.6(b). Fig.7(a) demonstrates the cornerdetection of feature points. The implementation result of the algorithm with PSO can be seen in Fig.7(b), this is the fitted outline at the final iteration.

Table 1.Names and contour details of images.

Image	Name	# of Contours	# of Contour Points
	Ellipses.bmp	1	[831]
	Plane.bmp	3	[1106+61 +83]

Tables 1 to 2 summarize the experimental results for different bitmap images. These results highlight various informations. As seen in Figs. 4 and 2, two images have been chosen to test the proposed scheme based upon the PSO. Table 1 shows the two experimental images, their file names, number of contours in each image, and number of contour points in each image.

Table 2 describes some more detail analysis of the proposed technique. In Table 2, column 1 describes file names of the test images, column 2 depicts the number of initial feature (corner) points, Column 3 represents intermediate feature points during the iterative process of PSO algorithm, columns 4 and 5 mention about the total time of execution of the algorithm for each image, without and with inserting intermediate features, respectively.

(a)

(b)

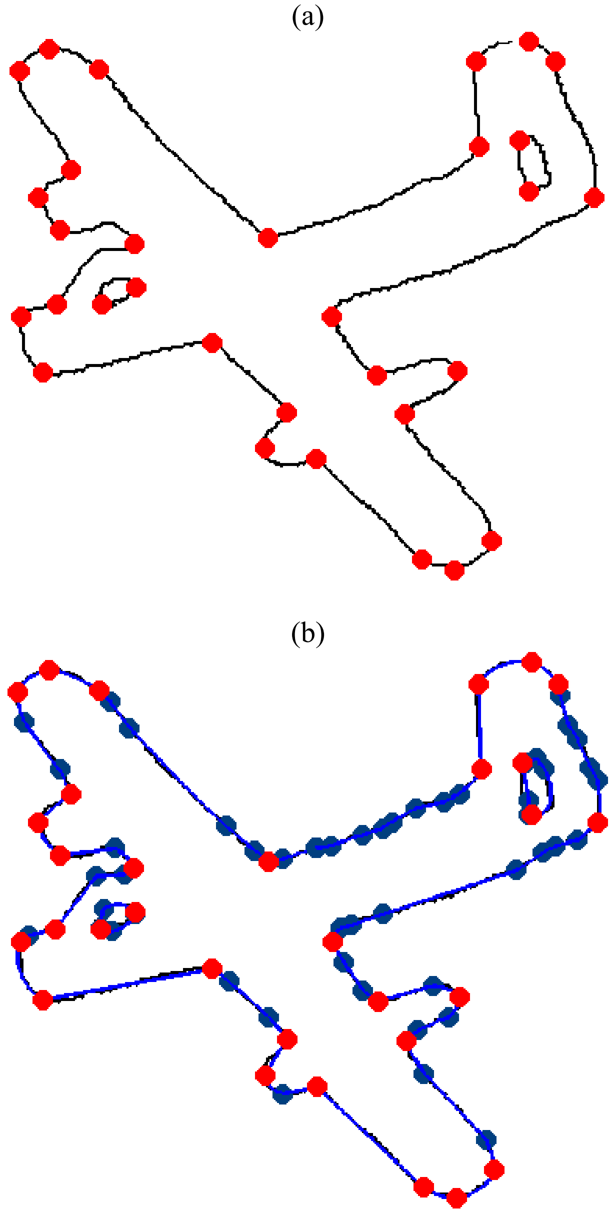


Figure 7. Further processing steps: (a) Corner points achieved, (b) Fitted Outline of the image.

Table 2.Comparison of number of initial corner points, intermediate points and total time taken (in seconds) for conic interpolation approaches.

Image	# of Initial Feature Points	# of Intermediate Points in Conic Interpolation	Total Time Taken for Conic Interpolation	
			Without Intermediate Points	With Intermediate Points
Elipses. bmp	4	24	85.735	660.238
Plane.b mp	31	26	78.175	368.242

7 Conclusion and Future Work

A global optimization technique, based on particle swarm optimization, is proposed for the outline capture of planar images. The proposed technique uses the particle swarm optimization to optimize a conic spline to the digital outline of planar images. By starting a search from certain good points (initially detected corner points), an improved convergence result is obtained. The overall technique has various phases including extracting outlines of images, detecting corner points from the detected outline, curve fitting, and addition of extra knot points if needed. The idea of particle swarm optimization has been used to optimize the shape parameters in the description of a conic spline introduced. The spline method ultimately produces optimal results for the approximate vectorization of the digital contour obtained from the generic shapes. It provides an optimal fit with an efficient computation cost as far as curve fitting is concerned. The proposed algorithm is fully automatic and requires no human intervention. The author is also thinking to apply the proposed methodology for another model curve namely cubic. It might improve the approximation process. This work is in progress to be published as a subsequent work.

Acknowledgment

This work was supported by Kuwait University, Research Grant No. [WI 05/12].

References:

[1] Chetrikov, D., Zsabo, S.: A Simple and Efficient Algorithm for Detection of High Curvature Points in Planar Curves, In Proceedings of the 23rd Workshop of the Australian Pattern Recognition Group, pp. 1751-2184 (1999).

[2] Sarfraz, M.: Computer-Aided Reverse Engineering using Simulated Evolution on NURBS, International Journal of Virtual & Physical Prototyping, Taylor & Francis, vol. 1(4), pp. 243 – 257 (2006).

[3] Lavoue, G., Dupont, F., Baskurt, A.: A New Subdivision Based Approach for Piecewise Smooth Approximation of 3D Polygonal Curves, Pattern Recognition, vol. 38, pp. 1139-1151 (2005).

[4] Horng, J. H.: An Adaptive Smoothing Approach for Fitting Digital Planar Curves with Line Segments and Circular Arcs, Pattern Recognition Letters, vol. 24(1-3), pp. 565-577 (2003).

[5] Yang, X.: Curve Fitting and Fairing using Conic Spines, Computer Aided Design, vol. 6(5), pp. 461-472 (2004).

[6] Sarfraz, M.: Designing Objects with a Spline, International Journal of Computer Mathematics, Taylor & Francis, vol. 85(7) (2008).

[7] Gonzalez, R.C., Woods, R.E., Eddins, S.L.: Digital Image Processing Using MATLAB, 2nd Ed., Gatesmark Publishing (2009).

[8] Nixon, M.S., Aguado, A.S.: Feature extraction and image processing, Elsevier (2008).

[9] J. Kennedy, R. Eberhart, Particle swarm optimization, Proc. IEEE Intl. Conf. Neural Networks, 4, Nov/Dec 1995, pp. 1942 –1948.

[10] R. Eberhart and J. Kennedy, A new optimizer using particle swarm theory, Proc. the Sixth Intl. Symposium on Micro Machine and Human Science, MHS '95, 4-6 Oct 1995, pp. 39 -43.

[11] Y. Shi, R. Eberhart, A modified particle swarm optimizer, The 1998 IEEE Intl. Conf. on Evolutionary Computation Proc., IEEE World Congress on Computational Intelligence, 4-9 May 1998, pp. 69 – 73.

[12] P. Reche, C. Urdiales, A. Bandera, C. Trazegnies, and F. Sandoval, Corner detection by means of contour local vectors, Electron. Lett. 38(14) (2002), pp. 699 - 701

[13] M. Marji and P. Siv, A new algorithm for dominant points detection and polygonization of digital curves, Pattern Recognition 36(10)(2003), pp. 2239–2251.

[14] M. Sonka, V. Hlavac, and R.D. Boyle. Image Processing, Analysis and Machine Vision. Brooks Cole, Pacific Grove, CA, 3rd edition, 2008.

[15] N. Richard, T. Gilbert, Extraction of Dominant Points by estimation of the contour fluctuations, Pattern Recognition, Vol. (35), pp. 1447-1462, 2002.

[16] Sarfraz, M. (2010), Vectorizing Outlines of Generic Shapes by Cubic Spline using Simulated Annealing, International Journal of Computer Mathematics, Taylor & Francis, Vol. 87(8), pp. 1736 – 1751.

[17] F. Attneave, "Some informational aspects of visual perception", Psychological Review, Vo. 61, 1954, pp. 183–93.

[18] A. Rattarangsi and R. T. Chin, "Scale-based detection of corners of planar curves," Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, 1992, pp. 430–4.

[19] M. Sarfraz, A. Rasheed and Z. Muzaffar, "A Novel Linear Time Corner Detection Algorithm, Computer Graphics," Imaging and Visualization – New Trends, Sarfraz, M., Wang, Y., and Banissi, E. (Eds.), ISBN: 3-7695-2392-7, IEEE Computer Society, USA, 2005, pp. 191-196.

[20] L. Dreschler and H. H. Nagel, "On the selection of critical points and local curvature extrema of region boundaries for inter-frame matching," Proceedings of ICPR, 1982, pp. 542–44.

[21] A. J. Pritchard, S. J. Sangwine and R. E. N. Horne, "Corner and curve detection along a boundary using line segment triangles," Electronics Division Colloquium on Hough Transforms Digest, No. 1993/106, 1993, pp. 1–4.

[22] Z. N. K. Swati, S. Zaman, and M. Sarfraz, "A Novel Corner Detector Approach using Sliding two Ellipses and one Rectangle," The Proceedings of International Conference on Frontiers of Information Technology (FIT 2009), December 16-18, 2009, COMSATS Institute of Information Technology, Pakistan, Article # 73, ISBN: 978-1-60558-642-7, ACM Press, 2009.

[23] Sarfraz, M., Swati, Z.N.K., (2013), Mining Corner Points on the Generic Shapes, Open Journal of Aplied Sciences, Vol. 3(1B), pp. 10 – 15, DOI:10.4236/ojapps.2013.31B003.