http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (20): 5853-5858, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.5853.5858

A Circle Fitting-based Method for Computing Parameters of Circular Arc in Scanning Engineering Drawing

Zili Zhang, Xuan Wang and Muhammad Waqas Anwar
Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen, 518055, China

Abstract: Arc segmentation is quite a challenging field in Graphics Recognition and the computation of the coordinates of centers and the radii of circular arcs is a crucial problem which has drawn much attention. Therefore this paper mainly discusses the application of circle fitting skill in scanning engineering drawings to determine correct coordinates of centers and radii. At first we should choose appropriate seed points and improve circle fitting algorithm of Instrumental Variable Estimator (IVE). Then we combine seed points and the improved IVE (IIVE) algorithm to calculate coordinates of centers and radii of circular arcs. In experimental section, the performance of IIVE and other two methods are compared by using classical experimental data and the coordinates of centers and radii are computed by employing the Arc Segmentation contest data. The results show that the proposed algorithm is very effective and efficient and the causes of the unsatisfactory results are analyzed.

Key words: Scanning engineering drawing, circle fitting, instrumental variable estimator, improved ive, seed points, line width

INTRODCUTION

The common primitives in scanning engineering drawings are lines and circular arcs, among which the circular arc is one of the most difficult elements to recognize. Much attention is focused on this field. Many approaches, e.g. Hough Transform and Thinning, etc. are proposed. Efficient techniques in recent decades were presented by Dori (1995), Liu and Dori (1998), Song *et al.* (2004) and Hilaire and Tombre (2006). Dori (1995) developed in his study a Machine Drawing Understanding System and computed the circular arc center by the intersection of the perpendicular bisectors of the chords. He could not handle circular arcs sharing vectors with other graphic objects, neither could Liu and Dori (1998). Hilaire and Tombre (2006) employed a skeleton method but this method could not avoid the inherent drawbacks of skeleton. The serious situation tells us that we should propose some new techniques to calculate the parameters (coordinates of centers and radii) of circular arcs in scanned engineering drawings, since this is the first and important step.

Circle fitting, which is widely used in many fields such as pattern recognition, computer vision, nuclear physics etc, is a classical method and has been presented as early as 1970s (Robinson, 1961). With several scattered points we can yield an approximate fitting circle having center and radius, as Fig. 1.

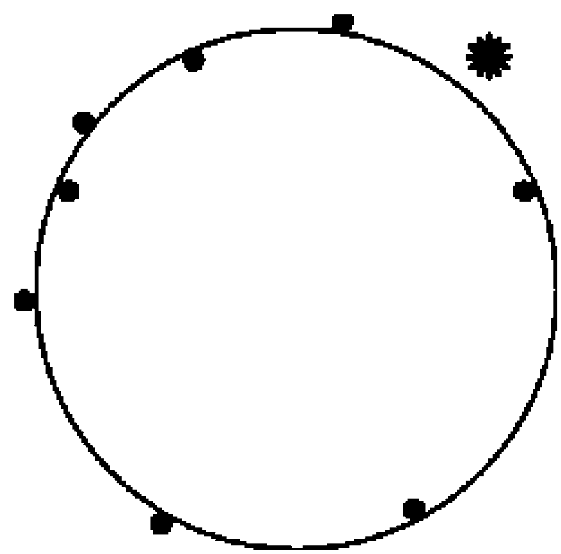

Fig. 1: Circle fitting (*denoting noise data, •denoting experimental data, circle denoting the optimal circle)

There are many variants to calculate parameters of a circle among circle fitting approaches. They can be classified as geometric fitting Eg and algebraic fitting Ea. The difference between them is the method of measuring the error distance. They are all denoted by the implicit equation F (X; U) = 0. The experimental data

Corresponding Author: Zili Zhang, Shenzhen Graduate School, Harbin Institute of Technology, 518055, China

$C = \{ X_i = (x_i, y_i), R^2, i = 1, 2... n\}$. U denotes three unknown parameters, i.e. (a, b) denotes coordinates of center of circular arc in X-Y Cartesian Coordinate System and r denotes the radius of circular arc:

$$E_g = \sum_{i=1}^{n} (\sqrt{(x_i - a)^2 + (y_i - b)^2} - r)^2 \quad (1)$$

And:

$$E_a = \sum_{i=1}^{n} ((x_i - a)^2 + (y_i - b)^2 - r^2)^2 \quad (2)$$

Geometric fitting is known for its accuracy. But it can be achieved by iteration process which is time-consuming. Algebraic fitting is regarded as fast speed and preciseness. Kanatani (2008) analyzed geometric fitting algorithms from theory and gave detailed and powerful argument about the differences between them. De Guevara et al. (2011) proposed a new algorithm called Absolute Geometric Error (AGE) which was the variation of mean absolute error. Although, it was a robust algorithm at noise and outlier points, it could handle neither n points which were collinear nor sensitivity of initial points.

COMPUTATION OF SEED POINTS

As we all know, circle fitting approaches work on the premise that there are at least three points which are used to determine the parameters. Song *et al.* (2004) have given some constructive suggestions to get seed points in scanning engineering drawing.

Motivation: Multi-resolution Arc Segmentation (MAS) algorithm presented by Song *et al.* (2004) consisted of four steps. Whereas we believe that the arc seed detection step is very important. If we can get accurate coordinates of centers and radii, the dynamic circular tracking step will need less computation since the iteration of this process is inefficient. This idea will be verified in experiment section. Sequentially the arc localization step and arc verification step take less time, so we should obtain exact coordinates of centers and radii as soon as possible. Song *et al.*, have not utilized the line width to compute the coordinates of seed points in Song *et al.* (2004), so we add this skill to compute the coordinates of seed points which is employed in circle fitting method to calculate parameters of circular arc. We also improve a circle fitting method to compute circle parameters. In order to get appropriate seed points we should take advantage of line width skill. Therefore line width will be introduced firstly and then state the process of computing seed points.

Line width: Before we manage to get several points called seed points, we firstly discuss line width. Line width is a particular feature of different components (lines, circle, circular arc and so on) that can be easily identified by readers. So we should compute accurate line width to contribute the process of choosing seed points. The scanning engineering drawing is a binary image, i.e. its background is white and foreground is black. Therefore when we encounter a black pixel P, in order to get a precise line width we will compute four directions lengths of this component in four different directions: vertical, horizontal, left diagonal and right diagonal. In order to make sure line width as accurate as possible, the line widths of four directions are calculated. Let the line width as the weight of corresponding direction, all the line width include the encountered pixel P. Then the number of same weight is counted as frequency. At last let the line width of corresponding the biggest frequency as that of this segment (part of a component). In ideal case, the line widths of four directions are same, usually at least that of two directions are same. With the same operation, the line width of other seed points will be computed. Let the biggest frequency line width of all see points as that of this component.

Seed points: In order to find appropriate seed points, the line width has twofold effects. One is that it can adjust whether all seed points belong to one component (the word “adjust”, not “judge”, because sometimes different components adjacent to each other have same line width). Secondly as line width is either odd or even. So as to compute a proper seed point we should distinguish odd width and even width carefully by line width knack. Line width is odd, as one pixel, three pixels, five pixels, etc. We take the component owning three pixels line width as an example. As shown in Fig. 2, the black pixels are a part of a component, the white square pixels are proper seed

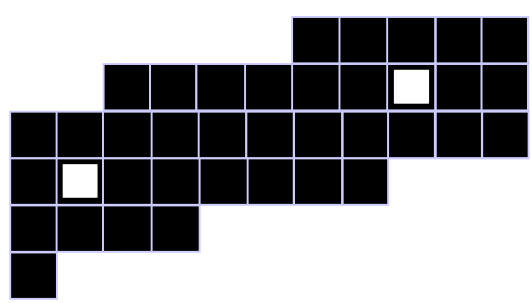

Fig. 2: Black component owning three pixels line width

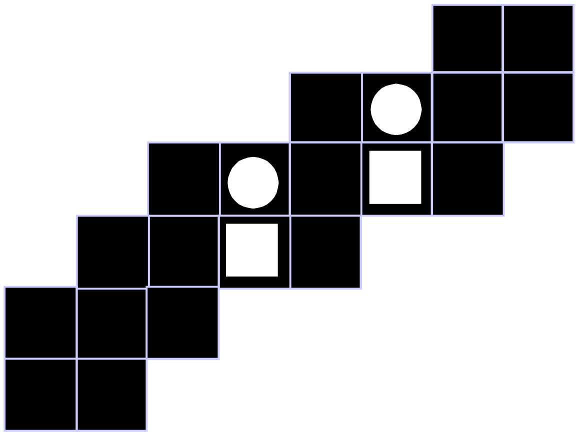

Fig. 3: Black component having two pixels line width

points. According to the algorithm presented in Song *et al.* (2004), we employ two concentric circles with different radii to calculate the seed points. There are several intersecting points where the windows and black pixels intersect and we always take the middle pixels as seed points, just as the white square pixels in Fig. 2. Line width is even, such as two pixels, four pixels and so on. We take the component of two pixels line width as an example, as shown in Fig. 3. The black pixels are a part of a component. The "circle" type or "square" type pixels are all proper seed points. According to the algorithm proposed in (Song *et al.*, 2004) when we calculate the seed points we always choose the inside most or outside most pixels of the middle two row pixels as candidate seed points. If we take a up (outside) white "circle" type pixel, all seed points taken are white "circle" type pixels. If we take a down (inside) white "square" type pixel, all seed points taken are the "square" type pixels. When the line width of the component is even and bigger than two pixels, the outside pixels will be removed and the inside two row pixels will be remained.

IMPROVED INSTRUMENTAL VARIABLE ESTIMATION (IIVE)

The major drawback of processing images is inefficient, especially about pixels. When the parameters of circular arcs or circle are computed; the accuracy and speed of these methods should be taken into consideration. According to previous discussion, the algebraic fitting is a proper choice. The algorithm presented by Chan *et al.* (2000) has poor performance when the amount of data is few, so the Instrumental Variable Estimator (IVE) algorithm is improved to handle a small quantity of data. The cost function is below:

$$J_a = \sum_{i=1}^{n}[(x_i - x)^2 + (y_i - y)^2 - R]^2 \tag{3}$$

And

$$J_a = \sum_{i=1}^{n}[x_i^2 + y_i^2 - (2\times x_i \times x + 2\times y_i \times y + R^2 - x^2 - y^2)]^2 \tag{4}$$

To give a brief description of this technique, we just build the key matrices A and b; the following processes are the same as the method of Chan *et al.* (). There is N experiment data, (x_i, y_i), i = 1, 2... N. To construct rows of matrices A and b, we subtract equation i+1 when the variation is i+1 from equation i when the variation is i in Eq. 4. And the resulting matrices A and b are below:

$$A = \begin{bmatrix} 2(x_2 - x_1) & 2(y_2 - y_1) \\ \vdots & \vdots \\ 2(x_N - x_{N-1}) & 2(y_N - y_{N-1}) \end{bmatrix} = [A_1 \mid A_2]$$

The size of A is (N-1)×2.

$$b = \begin{bmatrix} x_2^2 + y_2^2 - x_1^2 - y_1^2 \\ \vdots \\ x_N^2 + y_N^2 + x_{N-1}^2 + y_{N-1}^2 \end{bmatrix}$$

The size of b is (N-1)×1

In present experiments, there is usually not much data and some of them are noise. Therefore, we should add rows in matrices A and b to compute the parameters of circular arc to reduce errors.

EXPERIMENTS

Firstly, experiment is used to verify that the dynamic circular tracking step is time-consuming. Then the fitting algorithm is evaluated. At last parameters of circular arcs are calculated in scanning engineering drawings.

Verification of dynamic circular tracking step: Bresenham Algorithm (Bresenham, 1965), which is a representative drawing circle algorithm, has been used in dynamic circular tracking step. From (Song *et al.*, 2004) we can find that one should check several times to find next point with different angle. In our experiment we detail this process, first let the radii be multiple of 10 pixels (points), then compute the number of points on circle circumference by employing Bresenham Algorithm. As shown in Figure 4 all the points on circle circumference are counted, on the other hand the redundant points are removed as shown in Figure 5. Redundant point means that several points have same angle. For example, when

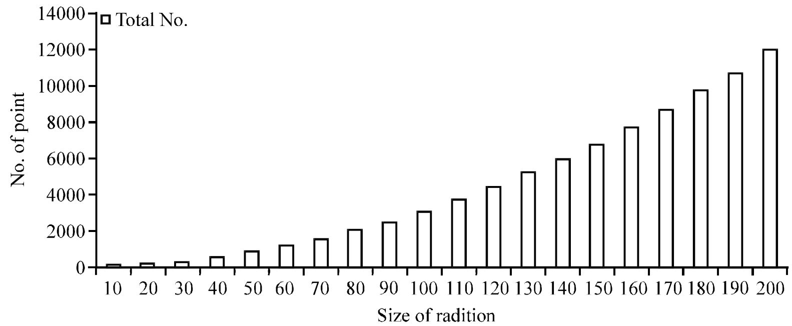

Fig. 4: Radii and the number of redundant points on circle circumference

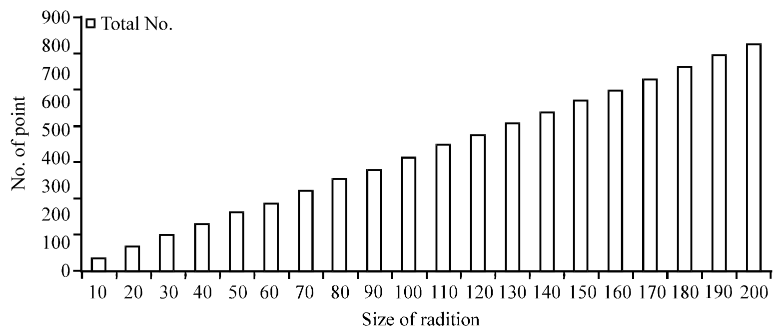

Fig. 5: Radii and the number of points with no redundancy on circle circumference

radius is 100 pixels (points), all points on circle circumference is 3108. If the redundant points are deleted, the number of points on circle circumference with no redundancy is 564. It is obvious that the algorithm should judge 5.5 times averagely to find next point with different angle.

Evaluation of fitting algorithms: Firstly, in order to evaluate the performances of IIVE, experiment chooses two classic sets of data which have been used for many times. We compare the improved circle fitting algorithm IIVE with the original algorithm IVE and the new algorithms AGE. The first set of data was used in (Umbach and Jones 2003) and shown as Table 1. The center of AGE is (1, 1) as IIVE's but the center of IVE method is not, therefore from this point we can see that the IIVE method is better than the original method. The visual image is Fig. 6.

The second set of data came from (De Guevara *et al*., 2011) and the data are as Table 2. The resulting image is shown in Fig. 7. We can see that IIVE algorithm has a better result than IVE and AGE.

From two visual images we can conclude that the IIVE algorithm is better than the other two algorithms.

Computation of parameters of circular arc in scanning engineering drawing.

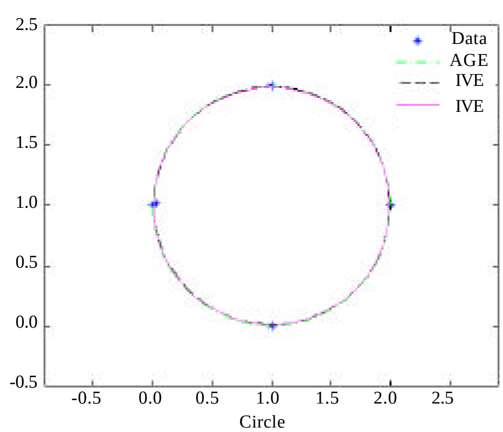

Fig. 6: Comparing IIVE, IVE and AGEc

Table 1: Data from Umbach and Jones (2003)

x	0	2	1	1	0.03
y	1	1	0	2	1.02

Table 2: Data from De Guevara *et al.* (2011)

x	0.7	3.3	5.6	7.5	6.4	4.4	0.3	-1.1
y	4.0	4.7	4.0	1.3	-1.1	-3.0	-2.5	1.3

According to the introduced technique, some seed points are computed from the scanning engineering

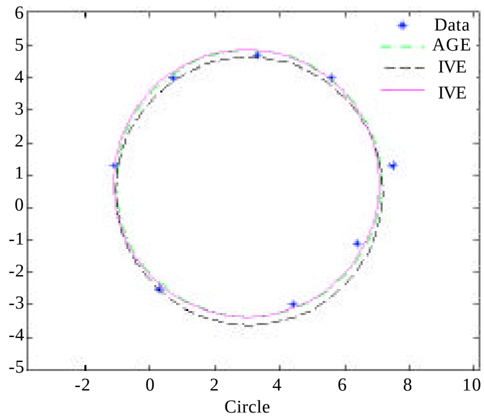

Fig. 7: Fitting results of three algorithms

Table 3: Computing parameters without line width skill

x	210.6	196.5	216.6	189.75	204
y	32.6	26	36.6	24.25	28.5
Algorithm	Center	Radius			
Ground Truth	(175,91)	R = 69			
Not using line width	(193,53.6)	R = 28.256			

Table 4: Computing parameters with line width skill

x	211	196	217	190	204
y	32	26	36	24	28.5
Algorithm	Center	Radius			
Ground Truth	(175,91)	R = 69			
Using line width	(177.2,90.85)	R = 67.82			

Table 5: Computing parameters without line width skill

x	311	295.5	318	290.333	389
y	389	392	390.5	396.667	303
Algorithm	Center	Radius			
Ground Truth	(415,309)	R = 27			
Not using line width	(412.1, 308.1)	R = 23.61			

Table 6: Computing parameters with line width skill

x	311	295	318	290	303
y	389	392	390	396	389
Algorithm	Center	Radius			
Ground Truth	(415,309)	R = 27			
Using line width	(414.6, 308.5)	R = 26.13			

drawings which have hardly noise data (we utilize some tricks to eliminate the noise data). The Figure 8 came from Arc Segmentation of GREC'2003 (Liu, 2004) which is the main activity of International Association of Pattern Recognition Technical Committee 10 (TC 10) on Graphics Recognition. We get the seed points by means of the algorithm of section two, i.e. the points where two concentric circles with different radii encountered the black pixels, the coordinates of all points are recorded and proper seed points from all points are chosen. Some noise data might be included but they will be removed in advance. Comparing the Table 3 with Table 4 above, we can see that we a get very approximate parameters by the

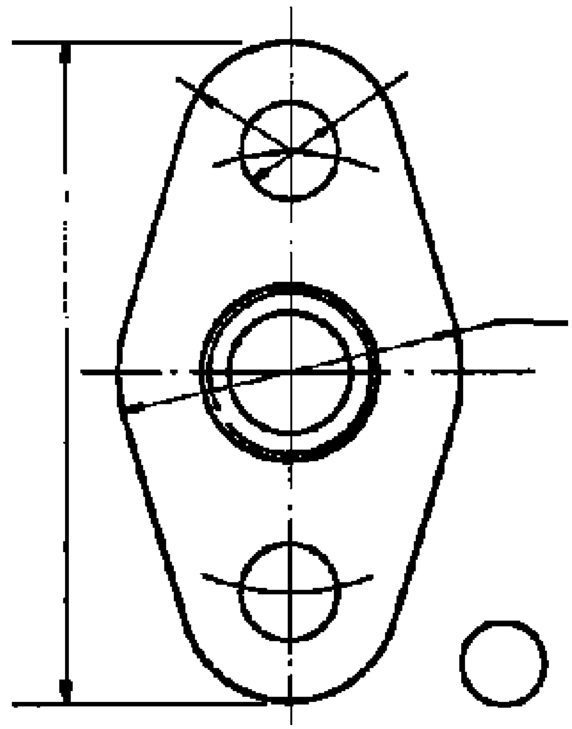

Fig. 8: A scanning engineering drawing from

improved method. In Table 3 we have not used line width skill to compute the parameters, while in Table 4 we used line width skill to calculate the parameters. It is obvious that the result of Table 4 is better than that of Table 3.

In Table 5 and 6, we employ another data to compare the results. The result of using line width technique in Table 5 is also better than that of unutilized line width technique in Table 6. Table 4 and 6 show that the resulting data are better than that of Table 3 and 5 but there is a small gap between the Ground Truth data and resulting data, that is because suboptimal seed points are sometimes chosen due to the circular arc intersecting with other arcs and lines. So the algorithm should be improved in the future.

CONCLUSIONS

From above experiments, it can be concluded that the accurate seed points are essential to compute parameters of circular arc or circle. Various cases are always encountered, e.g. tangency with a line or a circle, a circle with same line width as target circular arc, being disturbed by the auxiliary dimension information or signs. A good circle fitting algorithm is also very important, consequently in next step we should propose robust algorithms to handle all those problems.

ACKNOWLEDGMENTS

The authors wish to express their gratitude to the reviewers for the useful comments and suggestions.

REFERENCES

Bresenham, J.E., 1965. Algorithm for computer control of a digital plotter. IBM Syst. J., 4: 25-30.

Chan, Y.T., B.H. Lee and S.M. Thomas, 2000. Unbiased estimates of circle parameters. J. Optim. Theory Appl., 106: 49-60.

Dori, D., 1995. Vector-based arc segmentation in the machine drawing understanding system environment. IEEE Trans. Pattern Anal. Mach. Intell., 17: 1057-1068.

De Guevara, I.L., J. Munoz, O.D. de Cozar and E.B. Blazquez, 2011. Robust fitting of circle arcs. J. Math. Imaging Vision, 40: 147-161.

Hilaire, X. and K. Tombre, 2006. Robust and accurate vectorization of line drawings. IEEE Trans. Pattern Anal. Mach. Intell., 28: 890-904.

Kanatani, K., 2008. Statistical optimization for geometric fitting: Theoretical accuracy bound and high order error analysis. Int. J. Comp. Vision, 80: 167-188.

Liu, W.Y., 2004. Report of the arc segmentation contest. Lect. Notes Comput. Sci., 3088: 364-367.

Liu, W.Y. and D. Dori, 1998. Incremental arc segmentation algorithm and its evaluation. IEEE Trans. Pattern Anal. Mach. Intell., 20: 424-431.

Robinson, S.M., 1961. Fitting spheres by the method of least squares. Commun. Assoc. Compute. Mach., 4: 491-491.

Song, J.Q., M.R. Lyu and S.J. Cai, 2004. Effective multiresolution arc segmentation: Algorithms and performance evaluation. IEEE Trans. Pattern Anal. Mach. Intell., 26: 1491-1506.

Umbach, D. and K.N. Jones, 2003. A few methods for fitting circles to data. IEEE Trans. Instrum. Meas., 523: 1881-1885.