Two Problems of Circle in Image

Zili Zhang
Shenzhen Graduate School, Harbin Institute of Technology
Shenzhen, China
zhzili@gmail.com

Xuan Wang
Shenzhen Graduate School, Harbin Institute of Technology
Shenzhen, China
wangxuan@cs.hitsz.edu.cn

[1,2]Waqas Anwar
[1]Shenzhen Graduate School, Harbin Institute of Technology
Shenzhen, China
[2]Department of Computer Science, COMSATS Institute of Information Technology
Abbottabad, Pakistan
waqas@ciit.net.pk

Usama Ijaz Bajwa
Department of Computer Science, COMSATS Institute of Information Technology
Abbottabad, Pakistan
usama@ciit.net.pk

***Abstract* - Circle is an important primitive in document images, which is contained in many scenes and shapes due to the aesthetic feeling. As it is known to all, three nonlinear points determine a circle, namely, the radius and the coordinates of circle can be calculated. However, three nonlinear points cannot determine a unique desirable circle occasionally in images. Circumferences consist of many pixels, what is the relation of radius size and number of pixels on the circumferences with and without redundancy? We mainly discuss the condition about how to make a unique desirable circle with three nonlinear pixels in images and the above-mentioned relation. Finally, the relation and condition have been found and testified by experiments.**

Index Terms— Document Image, Circle, Bresenham Algorithm,

I. Introduction

Circle has a wide application scope in computer because of its characteristics, and has been researched for many years. In research field, rough classification about circle have three categories, the first type is to find an optimal circle by means of least square to fit scattered points; the second type is to detect circle shape in images employed in computer vision and pattern recognition; the third type is to segment circular arcs or circles in document images to interpret or recognize architectural drawings and engineering drawings.

The first application is to find an optimal circle to fit those scatted points. Chan et al. [1] employed Maximum Likelihood to estimate the circle parameters. Chernov et al. [2] developed an algorithm to fit circles with correlated noise. Sharadqah et al. [3] studied several popular circle fitting methods, and analyzed the reason why different methods have different effects; at last they proposed a new algebraic algorithm which had better performance. Kanatani [4] employed high order terms to evaluate the accuracy of geometric fitting of circle fitting. Zelniker et al. [5] gave a statistical analysis about Delogne-Kasa method; and he also used convolution to estimate circle parameters [6]. Guevara et al [7] proposed a geometric method based on mean absolute error.

Researchers developed algorithms to detect circle shape in images. Rosin et al. [8] adopted a recursive algorithm to segment curves in images into circular arcs and straight lines. Chung et al. [9] presented a fast randomized algorithm, which needed less memory, to search the center of circle. Pan et al. [10] proposed probabilistic pairwise voting method to detect circular object which is based on Hough Transform. Chung et al. [11] used three evidences to speed up the randomized approach, and then employed an efficient refinement strategy to improve the accuracy.

To recognize and interpret document images which is a kind of images, primitives like straight lines and circular arcs or circles should be segmented, generally speaking, circular arcs and circles have more challenge to be recognized than straight lines, so many methods had been proposed to research them. Dori [12] used a chain of bars and a triplet of points to detect circular arcs and computed the circular arc centers by means of the intersection of the perpendicular bisectors of the chords. Liu et al. [13] presented an incremental arc segmentation algorithm to segment arcs. Hilaire [14] used the skeleton technique to segment arcs. Elliman [15] presented an approach based on vectorizing binary images, smoothing the vectors to a sequence of small straight lines, and recognize arcs. Song et al. [16] proposed an algorithm which consisted of arc seed detection, dynamic circular tracking, arc localization and arc verification to effectively segment arcs. Lamiroy et al.[17] took Random Sample Consensus minimization technique to

978-0-7695-4927-9/12 $26.00 © 2012 IEEE
DOI 10.1109/FIT.2012.25

deform the candidate circular arc to fit arcs or reject arcs if the fitting has deviation.

From the previous discussion, we can know that circle has a very great prospect in wide scopes, and we discover a phenomenon existing in all cases that three nonlinear points cannot determine a unique desirable circle, the better example is arc segmentation in document images, such as Song [16] detected the radius and the coordinates of circular arc in arc seed detection stage, and then used dynamic circular tracking to adjust the deforming circle or reject the pseudo-circle which shared partial circular arc with the real circle. If the circle computed by several pixels is unique in the arc seed detection stage, the following part processes can be discarded, and much time and computation could be saved. Furthermore, the radius and coordinates of center of "circle" have been computed, the "circle" which is a complete circle or partial circle can be judged by random points, if the number of pixels on circle circumference can be known, more effective algorithms can be designed, so we should find out the relation of radius size and number of pixels on circle circumference. When we scan the pixels on circle circumference, delete the redundant pixels[1], if there are no redundant pixels on circle circumference, what is the relation of radius size and number of pixels on circle circumference. So this paper mainly discusses the condition about three nonlinear points determining a unique desirable circle and the relation of radius size and number of pixels on circle circumference.

II. Two problems of circle

A. The relation of radius size and number of pixels on circle circumference

First, we discuss the relation of radius size and number of pixels on the circumference. We use the Bresenham algorithm to draw a circle, and calculate the number of pixels on the circumference. The initializing value of the radius is 10, increased by 10. We compute the number of pixels with redundancy on the circumference. We use data with radii from 10 to 200 to show the relation in Fig.1.

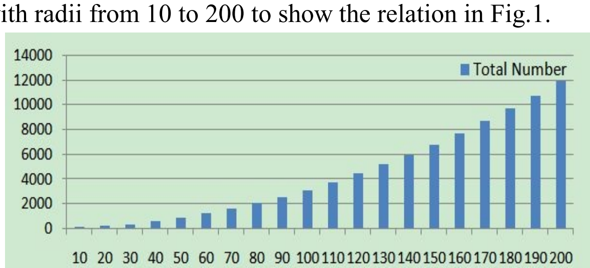

Fig.1 The relation about radii size and number of pixels with redundancy on circle circumference.

From the Fig.1 we can see that this relation seems to satisfy quadratic equation, i.e. parabola, we search the equation with all methods, but we get nothing. After we research the difference of number of pixels, we find the relation. The process is like this, Let the Ri (Ri = 10, 20, 30 …; i = 1, 2, 3…) be radius size, Ni (i = 1, 2, 3…) be the number of pixels, we use Table 1 to show the rule.

Table 1 shows the relation of number of pixels

i	1	2	3	4	5	6	7	8	9	10
R_i	10	20	30	40	50	60	70	80	90	100
N_i	56	168	336	564	848	1188	1584	2036	2544	3108
$N_i - N_{i-1}$		112	168	228	284	340	396	452	508	564
(N_i-N_{i-1}) - $(N_{i-1}-N_{i-2})$			56	60	56	56	56	56	56	56

From the last row, we discover an interesting phenomenon, called "5660 Phenomenon".

$$N_i + N_{i-2} - 2 * N_{i-1} = \begin{cases} 56; R_i \neq 40+70*n \\ 60; R_i = 40+70*n \end{cases} \quad (1)$$

When we want to get the next pixels with different angles on the circumference, we may judge several times to get the pixel. After researching this case, we find another very interesting phenomenon called "40 Phenomenon" about the circumference without redundant points. The Fig.2 can show the relation about radius size and number of pixels with no redundancy on circumferences.

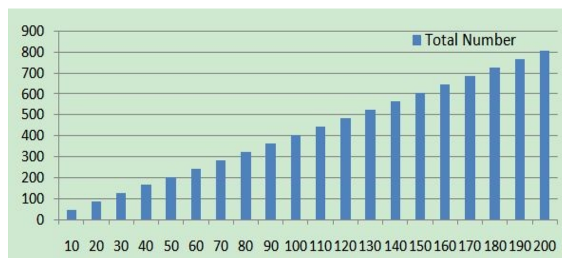

Fig.2 The relation of radii size and no redundant pixels on circle circumference.

From Fig.2, we can see that it is a linear relation of radius size and number of pixels with no redundant pixels on circumferences. Ri represents the radius; Ni represents number of pixels on circle circumference; i = 1, 2, 3…. The linear function is

$$N_i = 4 * P_i + 4; \quad (2)$$

Constrain condition: Ri is the multiples of 10.

From Fig.1 and Fig.2, we have a conclusion that no matter the shape is a complete circle or a partial circle, it is time-consuming to adopt Bresenham algorithm to "redraw" the circle because it costs much time to find the next point with different angle in Song's paper [16].

B. The condition of three nonlinear points determining a unique desirable circle

We all know that three nonlinear points can determine a circle; however they occasionally cannot determine a unique desirable circle. As Fig.3, Three circles, Circle 1, Circle 2,

[1] Redundancy means several near pixels have a same angle.

Circle 3, pass through three points, P1, P2 and P3. The reason is that the circular arc which three points form cannot be as a unique feature to distinguish itself from others.

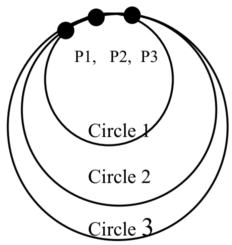

Fig. 3 Circle 1, Circle 2, Circle 3 through three points P1, P2, P3

Why this problem should be addressed? In images, if we want to know the radius and coordinates of center, we can scan the images and get more than three pixels to compute them (it is difficult to get more points, three is better), of course, we could use other methods like Hough Transform and Skeleton, but they have inherent drawbacks.

In this paper, if the desirable circle can be determined by the three points, we think that the circular arc which three points form can be considered as the only feature to distinguish itself from others. To ignore other factors, the line thickness of circle is set as one pixel so that the algorithm can get the points easily. And if the line thickness is odd and bigger than one, we can thin them to be one pixel. In the following part we will discuss the even line thickness.

In order to find the law, the image is zoomed in to the limitation, the pixels on the circumference is clear enough to be identified, so they can be considered as points due to the line thickness is one pixel. We get three points in image, the coordinates of middle point is fixed, and the coordinate of left point and right point will be changed, keeping the angle between middle point and left point or right point increasing 5 degrees every time. We determine previously the radius and the coordinates of center of a circle, and draw this circle with one pixel line thickness. The error of coordinates of center is evaluated by the absolute value between the coordinate value of real center and the center computed. The error of the radius is less than one pixel, so we do not consider it. When we compute the coordinates of center, the coordinates are double type, so if the difference of x-coordinate or y-coordinate is less than one pixel, we think that we get the right coordinates of center.

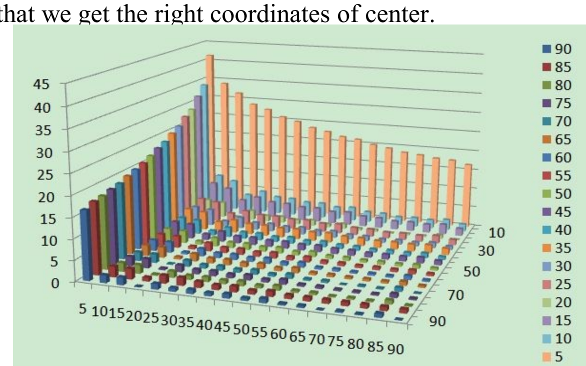

Fig.4 different angles have different values in circle with radius being 50 pixels

Here is one example: radius is fifty pixels, coordinates of center is (200, 200), change the angles to get more data, as Fig.4.

From the Fig.4, we can see that when the left angle and right angle are no more than fifteen degrees, the differences of x-coordinate and y-coordinate is more than one pixel. When the left angles and right angles are bigger, the computing results are better.

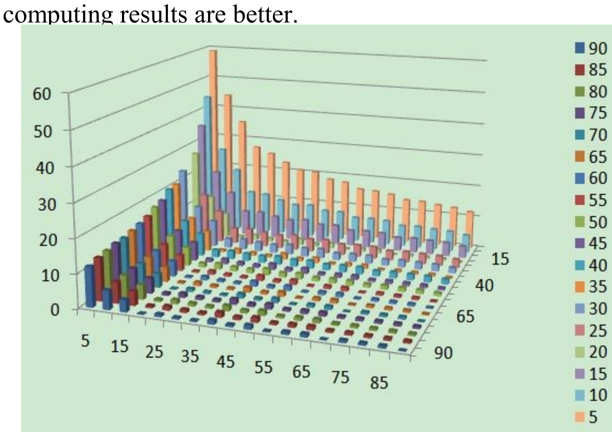

Fig.5 different angles have different values in circle with radius being 100 pixels

According to the result of Fig.5, when the left angle and right angle are no more than fifteen degrees, the differences of x-coordinate and y-coordinate are more than one pixel. But there are exceptions, i.e. when the left angle is twenty degree, or twenty-five degree, or forty degree, the error is bigger than two pixels.

According to our experiments, we conclude that the error is bigger than one pixel if the left angle and right angle are smaller than fifteen degrees. We can obtain the very accurate result on the premise that the left angle and right angle are ninety degrees respectively, but this case is rare in applications. So when we compute the parameters of circle in images, we should employ the surrounding information to adjust the parameters.

If the line thickness is even and bigger than two, we can thin the line to get the middle two pixels. To save the computation, we compute the average value of the intersection. If the line thickness is odd and bigger than one, we can thin the line to get the middle pixel. Or we can compute the average values of intersects directly to save computation at the cost of low accuracy.

III. EXPERIMENTS

TABLE 2 SHOWS THE RESULT HAVING REDUNDANT POINTS

i	**3**	**4**	**5**	**6**	**7**	**8**	**9**
R_i	2020	2030	2040	2050	2060	2070	2080
N_i	34112	45596	57136	68732	80384	92092	103860
$N_i - N_{i-1}$		11484	11540	11596	11652	11708	11768
$(N_i-N_{i-1})-(N_{i-1}-N_{i-2})$			56	56	56	56	60

We use the IBM cluster to compute the number of pixels on circumferences, Table 2 gives the results with redundant points.

Table 3 gives the results having no redundancy, and we use the equation to validate the results.

TABLE 3 SHOWS THE RESULT HAVING NO REDUNDANCY

i	1	2	3	4	5	6	7	8
R_i	1000	2000	3000	4000	5000	6000	7000	8000
N_i	4004	8004	12004	16004	20004	24004	28004	32004
$N_i - N_{i-1}$		4000	4000	4000	4000	4000	4000	4000
		100*40	100*40	100*40	100*40	100*40	100*40	100*40

In Fig.2 the radius is the multiples of 10, now the radius is the multiples of 1000, so the difference of neighbors is 100 * 40.

Arc segmentation contest has already been held by International Association of Pattern Recognition since 1995. Herewith we borrow Fig.6 from GREC 2005 to confirm the condition.

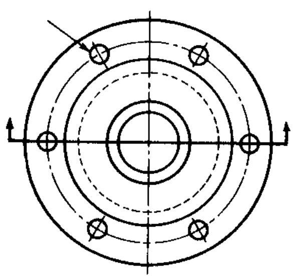

Fig.6 Image from GREC 2005.

TABLE 4 SOME DATA TO TEST THE CONDITION

L_A	R_A	L_Point	M_Point	R_Point	Center	True-Center	Error	Verification
90	90	(74, 211)	(242, 44.5)	(410, 211)	(242, 212.507)	(242, 211)	1.507	(242, 210)
20	20	(185, 53)	(242, 44.5)	(299, 53)	(242, 239.868)	(242, 211)	28.868	(242, 210)
15	15	(199, 49)	(242, 44.5)	(285, 49)	(242, 252.194)	(242, 211)	41.194	(242, 212)
10	10	(213, 46)	(242, 44.5)	(271, 46)	(242, 325.583)	(242, 211)	114.583	(242, 212)
5	5	(227, 44)	(242, 44.5)	(257, 44)	(242, -180.75)	(242, 211)	[2]	(242, 212)

From Table 4 we can see that when the left angle and right angle are ninety degree respectively, the error is very small, while when the left angle and right angle are about fifteen degrees respectively, the error is bigger. If we employ the surrounding information i.e. symmetry axis, to adust them, the verification column shows that the verified result are better.

In Song et al.'s paper [16], they compute the coordinates of center by means of four or five midpoints of the segments in arc seed detection step, from the previous analysis we can see that it is impossible to get correct center, their dynamic circular tracking step which adjust the coordinates of center can prove this point. We use the environmental information e.g. symmetry axis to adjust coordinates of center, this process utilize simply arithmetic, so it does not need much time; while Song et al. use the Bresenham algorithm to dynamic circular track, from the "**5660 Phenomenon**" and "**40 Phenomenon**" we know that Song et al.'s process may not meet people' requirement.

In Lamiroy et al.'s paper[17], the arc candinate generation part uses three points to compute the coordinates of center, then exclude the pesudo-circle and save the real-circle by testing. They need to exclude the pesudo-circle due to incorrect coordinates of center, consequently it is time-consuming. Using the previous analysis, we verify the coordinates of center by employing the environmental information, which is simple and fast.

From Table 4, our method can get accurate result. Compared with the two methods of Song et al. and Lamiroy et al., we have superiority at speed and accuracy.

IV. CONCLUSION

In this paper, we find two interesting phenomena, namely, "**5660 Phenomenon**" and "**40 Phenomenon**". From the phenomena we can conclude that scanning the circumference with redundant points is slower than the circumference with no redundancy. And in image when we want to compute the coordinates of center and radius through three nonlinear points, the result probably has deviation except the left angle and right angle are ninety degrees but this case is very rare, so we should employ surrounding information to verify the parameters.

REFERENCES

[1] Y.T.Chan, B.H.Lee, S.M.Thomas. Approximate Maximum Likelihood Estimation of Circle Parameters. Journal of Optimization Theory and Applications.Vol.125, No.3.pp.723-734, June 2005

[2] N.Chernov, P.N.Sapirstein. Fitting circles to data with correlated noise. Computational Statistics and Data Analysis.Vol.52, pp: 5328-5337. 2008

[3] A.Al-Sharadqah, N.Chernov. Error analysis for circle fitting algorithms. Electronic Journal of Statistics.Vol.3, pp: 886-911.2009

[4] K.Kanatani. Statistical Optimization for Geometric Fitting: Theoretical Accuracy Bound and High Order Error Analysis. International Journal of Computer Vision.Vol.80, pp:167-188.2008

[5] E.E.Zelniker, I.Vaughan L.Clarkson. a statistical analysis of the DELOGNE-KASA method for fitting circles. Digital Signal Processing.Vol.16, pp: 498-522.2006

[6] E.E.Zelniker, I.V. L.Clarkson. Maximum-Likelihood Estimation of Circle Parameters via Convolution. IEEE Transactions on Image Processing, Vol.15, No.4, APRIL 2006.

[7] I.L. de Guevara, J.Munoz, O.D. de Cozar, E.B.Blazquez. Robust Fitting of Circle Arcs.J. Math. Imaging Vis.Vol.40, pp: 147-161, 2011

[8] R.L Rosin, G. A W West. Segmentation of edges into lines and arcs.Vol.7, No.2, pp: 109-114.1989

[2] Three points are collinear.

[9] K.L.Chung, Y.H.Huang, J.P.Wang, T.C.Chang, H.Y.M.Liao. Fast randomized algorithm for center-detection.Vol.43, pp: 2659-2665.2010

[10] L. Pan, W.S. Chu, J.M.Saragih, F.De La Torre. Fast and Robust Circular Object Detection with Probabilistic Pairwise Voting (PPV). IEEE Signal Processing Letters.Vol.18, No.11, November 2011

[11] K.L.Chung, Y.H.Huang, S.M.Shen, A.S.Krylov, D.V.Yurin, E.V.Semeikina. Efficient sampling strategy and refinement strategy for randomized circle detection. Pattern Recognition.Vol.45, pp: 252-263.2012

[12] D. Dori. Vector-based arc segmentation in the machine drawing understanding system environment. IEEE Transaction on pattern analysis and machine intelligence, Vol.17, No.11, pp: 1057-1068.November 1995

[13] W. Liu, D.Dori. Incremental arc segmentation algorithm and its evaluation. IEEE Transaction on pattern analysis and machine intelligence. Vol.20, No.4, pp: 424-431.APRIL 1998

[14] X.Hilaire. RANVEC and the Arc Segmentation Contest.GERC 2002, Lecture Notes in Computer Science 2390, pp: 359-364.2002

[15] D.Elliman.TIF2VEC, an algorithm for arc segmentation in engineering drawings. GREC 2002, Lecture Notes in Computer Science 2390, pp: 350-358, 2002

[16] J.Song, M.R. Lyu, S.Cai. Effective multiresolution arc segmentation: algorithms and performance evaluation. IEEE Transactions on Pattern Analysis and Machine Intelligence.Vol.26, No.11, pp: 1491-1506.November 2004.

[17] B.Lamiroy, Y.Guebbas. Robust and precise circular arc detection. GREC 2009, Lecture Notes in Computer Sciences 6020, pp: 49-60, 2010