An algorithm for the detection of multiple concentric circles

Margarida Silveira

IST/ INESC, Lisbon, Portugal

Abstract. This paper presents a method for the detection of multiple concentric circles which is based on the Hough Transform (HT). In order to reduce time and memory space the concentric circle detection with the HT is separated in two stages, one for the center detection and another for the radius determination. A new HT algorithm is proposed for the center detection stage which is simple, fast and robust. The proposed method selects groups of three points in each of the concentric circles to solve the circle equation and vote for the center. Geometrical constraints are imposed of the sets of three points to guarantee that they in fact belong to different concentric circles. In the radius detection stage the concentric circles are validated. The proposed algorithm was compared with several other HT circle detection techniques. Experimental results show the superiority and effectiveness of the proposed technique.

1 Introduction

There are several applications that require the automatic detection of concentric circles. Some examples include iris detection, the detection of washers in industrial parts or the detection of inhibition halos of antibacterial activity in the food industry.

However, and despite the fact that circle detection with the Hough Transform (HT) was been widely studied, the specific case of the detection of concentric circles has received little attention. In fact, the only publication on this matter known to the authors is the one proposed by Cao and Deravi [9]. The Cao-Deravi method is based on scanning the image horizontally and vertically to search for groups of three edge points in order to determine the parameters of a circle. The edge gradient direction is used as a guide, together with a set of search rules, to select the three edge points. The fact that this method only performs horizontal and vertical scanning makes it inadequate when there is occlusion or deformation of the circles. In addition, and even though gradient direction is not used directly, the fact that is used in the search makes the method susceptible to noise because gradient direction is very much affected by noise.

The method we propose for concentric circle detection is also based on the search for three edge points belonging to the same circle. However, we scan the image in various directions and instead of using gradient direction to guide the search we rely on geometrical constraints. In the HT there is a tradeoff between computational effort in the edge space and computational effort in the accumulator space. The method we propose involves more work in the edge space and is therefore more effective in situations where the background is complex or the image noise is significant, because in those cases the analysis of the accumulator is too complex.

The proposed method will be described in section 2.

2 Detection of concentric circles

Circles are described by three parameters, the center coordinates (*a*,*b*) and the radius *r*:

$$(x-a)^2+(y-b)^2=r^2 \tag{1}$$

Therefore, the conventional HT [1] needs a three dimensional parameter space. In order to reduce time and memory space, the problem of circle detection was separated in two stages [2] [3]. The first stage involves a two parameter HT to find the center (*a*,*b*) of the circles and the second stage involves a one dimensional HT, that is, a simple histogram, to identify the radius of the outer circle. A new algorithm is proposed for the center detection stage. Since there are several circles in the image, there will be several peaks in the center parameter space corresponding to the different circle centers. In addition, there may be spurious peaks resulting from the interference of different circles. Therefore, the one dimensional Hough transform designed to identify the radius will also be used to validate the existence of two, or more, concentric circles.

We assume that the position and number of concentric circles halos is variable and unforeseeable and that the size of the concentric circles lies within the a known range $r_{\min}$ to $r_{\max}$.

2.1 Detection of circle centers

We propose a HT center detection method that is simple and, as results will show, is fast and reliable. The algorithm is robust to noise since it does not use gradient information and is able to detect irregular and partially occluded circles.

After edge detection, the connected components are labeled. For each point $A=(x_A,y_A)$ another two points $B=(x_B,y_B)$ and $C=(x_C,y_C)$ of the same component are randomly selected that satisfy the following expressions:

$$d_{\min}^2 \le (x_A-x_B)^2+(y_A-y_B)^2 \le 4r_{\max}{}^2 \tag{2}$$

$$d_{\min}^2 \le (x_A-x_C)^2+(y_A-y_C)^2 \le 4r_{\max}{}^2 \tag{3}$$

$$d_{\min}^2 \le (x_B-x_C)^2+(y_B-y_C)^2 \le 4r_{\max}{}^2 \tag{4}$$

where r_{max} is the maximum value allowed for the radius and d_{min} prevent selecting points too close.

The three points A, B and C are used to solve the circle equation and find a candidate circle center. Let $O = (x_O, y_O)$ denote that candidate center.

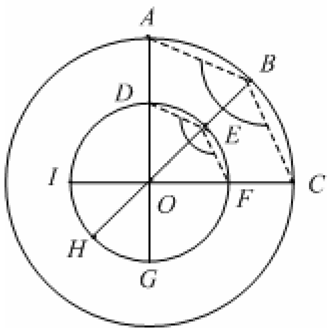


Fig. 1 The lines between each point A, B and C on the outer circle and the center should intersect the inner circle at points D, E and F respectively. The angle between line segments $\overline{BA}$ and $\overline{BC}$ equals the angle between $\overline{ED}$ and $\overline{EF}$.

In the case of two concentric circles, the lines between AO, BO and CO should intersect a different connected component at points D, E and F, respectively or even at points G, H and I. Moreover the angle between $\overline{BA}$ and $\overline{BC}$ should be the same as the angle between $\overline{ED}$ and $\overline{EF}$ or $\overline{HI}$ and $\overline{HG}$. This is illustrated in Fig. 1.

Therefore, the proposed algorithm will draw three lines that go through each of the points A, B, C and the candidate center O and conduct a search for edge points that lie on each of the three lines and belong to a given connected component which is different from the one that points A, B and C belong to. If such three points D, E and F are found and they verify the angle requirement, $\phi\left(\overline{BA},\overline{BC}\right) = \phi\left(\overline{ED},\overline{EF}\right)$, then the center coordinates O are incremented in the two dimensional Hough space. Additionally, the three new points are used to find another estimate of the center O and this new estimate is also incremented in the Hough accumulator in order to increase the center detection accuracy.

The Bresenham line drawing algorithm [4] was used to draw the lines and the Cramer's rule was used to solve the circle equation from the sets of three points. The angle between two line segments, for instance $\overline{BA}$ and $\overline{BC}$ is calculated by the following expression:

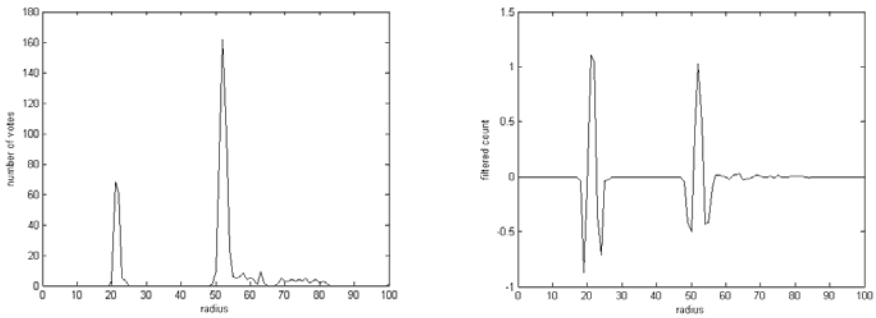
$$\phi\left(\overline{BA},\overline{BC}\right) = arc\cos\frac{\overline{BA}\ .\ \overline{BC}}{\|BA\|\|BC\|} \tag{5}$$

The number of concentric circles is variable and unforeseeable, and consequently the centers accumulator will have several peaks. In the radius detection stage described in section 2.2 each candidate center will be validated. After a circle has been analyzed, in the centers accumulator a small region of cells around that center are zeroed and then the accumulator is searched for the next peak.

2.2 Detection of circle radius

After a candidate circle center has been detected, edge points in the $2r_{\max}$ square region around the candidate center vote for the corresponding circle radius using expression (1).

Concentric circles will originate several peaks in the radius accumulator corresponding to the different radii. Since larger concentric circles will also originate higher peaks in the radius accumulator, the count is normalized. In addition, the accumulator is filtered to enable the detection of the more diffuse or deformed circles. The filter proposed in [9] was used. We require that circles have some percentage of their circumference appearing in the image. That percentage may be different for the inner circles than for the outer ones depending on the application.



a) b)

Fig. 2 Radius histogram. Two maxima appear corresponding to two concentric circles. The identified circles were at $r = 22$ and $r = 52$ a) Number of votes b) Filtered count.

A group of two concentric circles is identified as two peaks in the radius accumulator that verify the percentage requirement. Fig. 2 shows an example of the radius histogram corresponding to an accepted concentric circle. In case there is prior knowledge about the expected ring width, that information can be incorporated to check the separation between peaks. Naturally, the method can be extended to deal with any number of concentric circles.

4 Experimental results

In this section we will present results of the proposed method using real and synthetic images. We will study the performance of the technique with increasing additive Gaussian noise and compare it with the Ioannou method proposed in [9] and the Cao-Deravi method [9]. The Cao-Deravi method was described in the introduction.

The Iaonnou method, although it was not specifically designed for the detection of concentric circles, it can be utilized for that purpose if the radius validation suggested in section 2.2 is used. This method exploits the property that every line that perpendicularly bisects any chord of a circle passes through its centre. Therefore, it selects pairs of points of the same connected component and finds the line that perpendicularly bisects the two points. All the points on this line that belong to the parameter space are incremented. This method is very accurate and robust when compared to methods that rely on edge gradient direction. The Bresenham line drawing algorithm [4] was used to find the bisection line.

The tests were performed in the following conditions. For each value of sigma 50 test images of size 256x256 were created. Each image consisted of 5 black rings on a white background. The position, size and width of the rings were all randomly selected meaning there may be overlap. An example is show in Fig. 3. The Canny operator was used for edge detection because of its good localization and its robustness in the presence of noise, and also because Canny´s gradient magnitude is not as sensitive to contour orientation as other detectors [7]. In the Cao-Deravi algorithm edge direction was obtained with Wilson and Bhalerao's operator [5]. The radius threshold was set to 0.3 for both the outer and the inner circle. All the angles calculated with expression (**5**) used PI/60 accuracy.

The different techniques were compared as to their false positive rate, miss detection rate and accuracy. The radius threshold value has an influence on these statistics. In fact, increasing this threshold would increase the miss detection rate and simultaneously decrease the false positive rate.

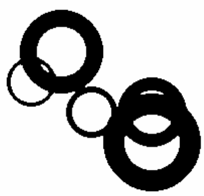


Fig. 3 Test image example

The results are presented in Fig. 4. As it would be expected, the performance of all the methods decreases with increasing noise sigma. It can be seen that the method has much lower miss detection rate than the other methods and also less false positive rate, although with slightly worse accuracy when there is little noise. The Cao-Deravi method has the best accuracy for low amounts of noise but is also the less resistant to increasing noise.

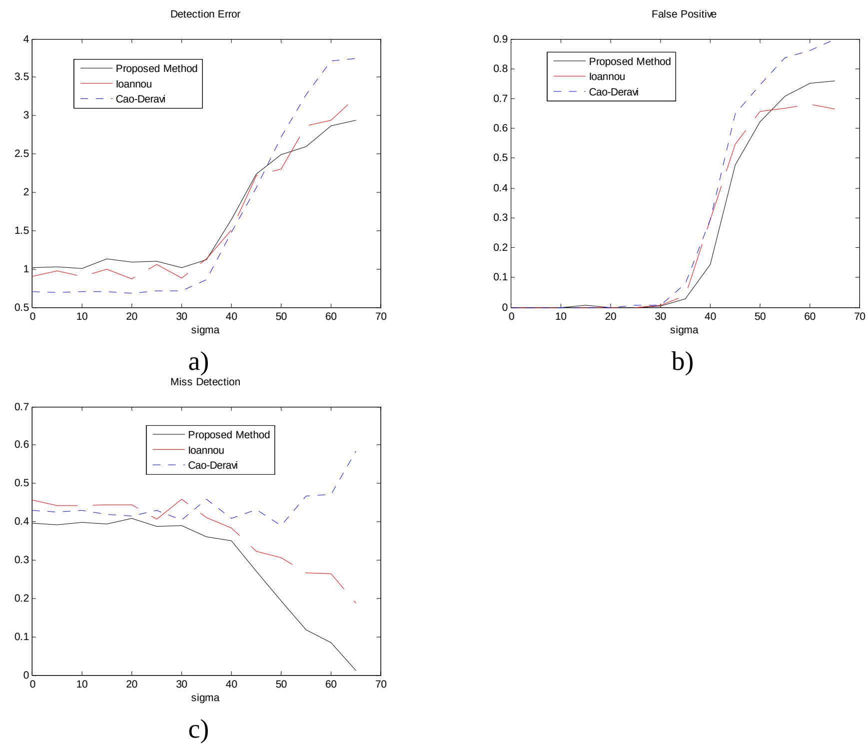


Fig. 4 Comparison of the methods performance with increasing noise. a) Detection error b) False positive rate c) Miss detection rate

Fig. 5 shows an example of the application of the different methods on a real image with a circular traffic sign and a complicated background. The images show that the proposed method was the best in detecting the traffic sign. The Cao-Deravi method was also able to detect the traffic sign although slightly skewed. The Ioannou method failed to detect the sign and produced two false concentric circles. Another example is shown in Fig. 6 where the test image is a motorcycle. In this example both the proposed and the Ioannou methods detected the front tire and the results are quite similar. The Cao-Deravi method however, missed the front tire and detected the fender and also originated a false detection.

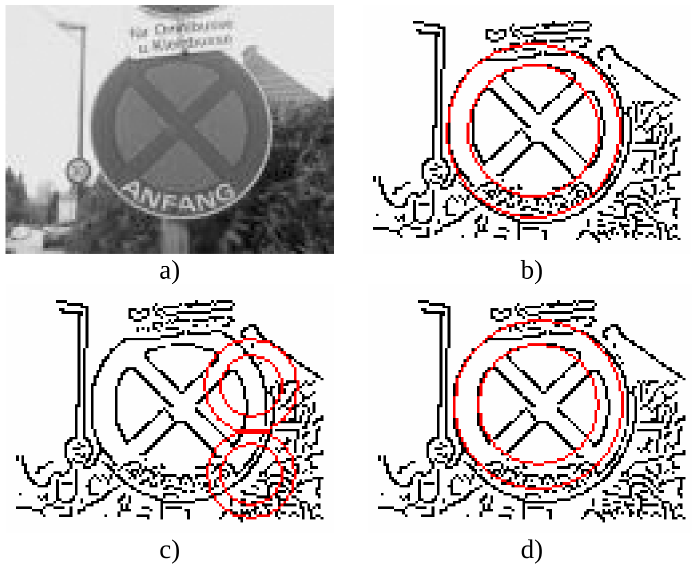


Fig. 5 Example of the methods performance using a real image. a) Original image b) Results of the proposed method superimposed on the edge map c) Results of the Ioannou method superimposed on the edge map d) Results of the Cao-Deravi method superimposed on the edge map

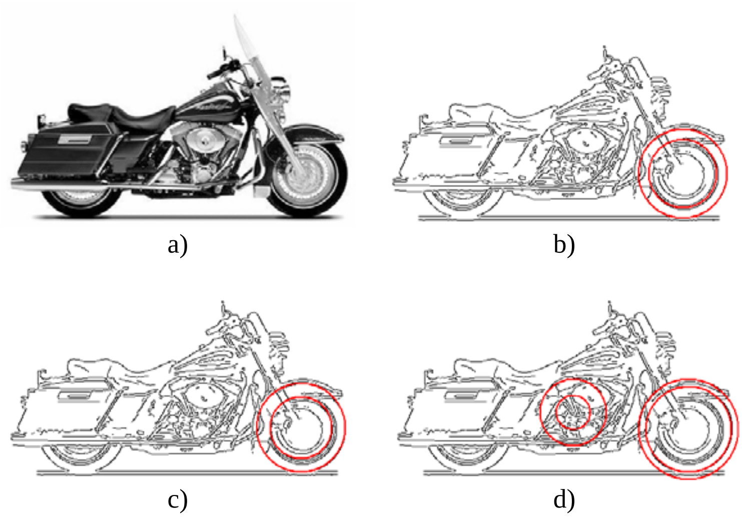


Fig. 6 Example of the methods performance using a real image. a) Original image b) Results of the proposed method superimposed on the edge map c) Results of the Ioannou method superimposed on the edge map d) Results of the Cao-Deravi method superimposed on the edge map

5 Conclusions

This paper proposed a method for the detection of multiple concentric circles. In order to reduce time and memory space the circle detection with the HT was separated in two stages, one for the center detection and another for the radius determination. A new algorithm was proposed for the detection of the circle centers. The proposed method selects groups of three points in each of the concentric circles to solve the circle equation and vote for the center. Geometrical constraints are imposed of the sets of three points to guarantee that they in fact belong to different concentric circles. The search is performed in several directions and doesn't rely on the use of edge direction. Therefore the algorithm is robust to noise and occlusion. The radius determination stage also performs a validation of the concentric circles.

Examples were provided to illustrate the performance of the algorithm. The proposed algorithm was favorably compared with other HT center detection methods. Experiments showed that the method is more robust to noise.

References

1. Duda, R., Hart, P., Use of the Hough Transformation to Detect Lines and curves in Pictures. Communications of the ACM, 15, 1, (1972), 11-15
2. Illingworth, J., Kittler, J.: The Adaptive Hough Transform. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-9, 5 (1987) 690–698
3. Kimme, C., Ballard, D.H., Sklansky, J.: Finding Circles by an Array of Accumulators. Communications of the ACM, 2, (1975), 120-122
4. Foley D., J., van Dam, A., Feiner, S., K., Hughes, J.,F.: Computer Graphics: Principles and Practice. 2nd edn., Addison-Wesley, Reading, MA (1997)
5. Wilson, R., Bhalerao, H.: Kernel Designs for Efficient Multiresolution Edge Detection and Orientation Estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 14, 3 (1992) 384-389

a. Haralick, R., M., Shapiro, L.G.: Computer and Robot Vision. Addison-Wesley (1992) 80-81

7. Ziou, D.: The Influence of Edge Direction on the Estimation of Edge Contrast and Orientation. Pattern Recognition, 34, 4 (2001) 855–863
8. Ioannou, D., Huda, W., Laine, A.F.: Circle Recognition through a 2D Hough Transform and Radius Histogramming. Image and Vision Computing, 17, 1 (1999) 15–26
9. Cao, X., Deravi, F.: An Efficient Method for the Detection of Multiple Concentric Circles. Proc. International Conference on Acoustic, Speech and Signal Processing, ICASSP'92, San Francisco, (1992) III-137-III-140