

Mathematical Implementation of Circle Hough Transformation Theorem Model Using C# For Calculation Attribute of Circle

SUTARNO^{1*}, ABDURAHMAN², Rossi PASSARELLA², Yosef PRIHANTO³, Rudy A.G GULTOM¹

¹Department of Computer Engineering, Faculty of Computer Science, Sriwijaya University, Indonesia
²Department of Sensing Technology Defense, Faculty of Technology, Indonesia Defense University, Indonesia
³Department of Research, Geospatial Information Agency Indonesia (BIG), Indonesia
 *Corresponding author: sutarno@unsri.ac.id

Abstract. Hough transformation is a method or theorem is used in searching primitive object forms, such as lines and circles. Circular Hough transformation detection has high accuracy rate in the form of the circular object. There are several stages in performing the application of the mathematical model for circular Hough transformation, namely, pre-processing image and the process of applying the theorem. In the process of pre-processing the image, the image will change the format from RGB format to one channel image using edge detection then, the result of the image will be applied to mathematical circular Hough theorem to find the Attribute of circle like center point and the radius of the object

Keywords: Hough transformation, circle Hough transformation, pre-processing Image, mathematical model, radius

Introduction

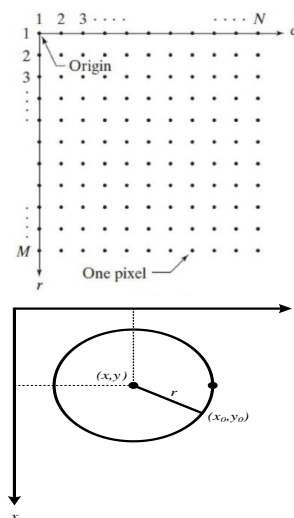
At the last year, the applications on several image processing with methods for circle detection have been studied [1]. Mathematical models are important for development algorithm on image processing because of their role and improve the system [2,3]. The method of Hough transformation is efficiently for identified lines in images [4,5]. CHT is an algorithm that is often used for the detection of circles with a high degree of accuracy[6,7], CHT recognize the object of a circle by calculating the number of accumulator with certain conditions that are assumed as the center of the circle by mapping the circle on each edge of the object [8,9]. The main purpose of this paper is to apply one of the theorems of the circular Hough transformation, from the mathematical model to the programming language to produce an application that can provide information about the image. The image for applying this theorem is a circle with a size of 320 x 240 pixels as shown in Fig. 1 below.



Figure 1. Sample of Circular Object from Pupil Eye

The application of this theorem to the sample image is expected to detect the center and radius of the circle we

input into the application, while the output of the application is the information about the position of the coordinate conventions of the center and the radius of the circle object. The Hough transformation algorithm is usually used to detect the curved objects, such as lines, circles and ellipses, Hough transformations used to detect circular objects called circular Hough transformations. The formula for circular Hough transformation is following [13]. The illustration used to describe Eq.1 can be seen on (Fig. 2).



(a) (b)

Figure 2. The Illustration Circular Hough Transformation (a) Coordinate Conventions, (b) Circular Transformation

where: x_0 = coordinate point x outer radius, x = center point of the x coordinate circle, y_0 = coordinate point y outer radius, y = center point of the y coordinate circle, r = radius.

$$(x - x_0)^2 + (y - y_0)^2 = (r)^2 \tag{1}$$

From the formulas and illustrations of Fig. 2 then the image is being processed to obtain the region that became the focus for the detection of a circle. This image processing brings out a new mathematical theorem to get the center point and the radius of a circle of an object. The illustration of this image processing can be seen in Fig. 3.

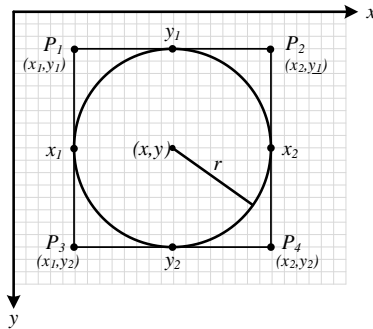


Figure 3. Circle Detection Theorem

In Fig. 3 there are some parameters of pixel coordinate points represented by coordinates; $P_1 (x_1, y_1)$ $P_2 (x_2, y_1)$; $P_3 (x_1, y_2)$; $P_4 (x_2, y_2)$; From the four coordinates points will be formed a cube so that the center can be found using the following equation:

$$x = x_1 + \frac{x_1 - x_2}{2} \tag{2}$$

$$y = y_1 + \frac{y_1 - y_2}{2} \tag{3}$$

$$r = x_2 - x \tag{4}$$

where, x = center point of the circle x -axis, x = center point of circle y -axis, x_1 = nearest x -axis that has extreme pixel values, x_2 = furthest x -axis that has extreme pixel value, y_1 = nearest y -axis that has extreme pixel value, y_2 = furthest y -axis that has extreme pixel value, r = radius. The Eq. 2 is

used to locate the center point of the circle on the x coordinate while the Eq. 3 is used to locate the center point on the y coordinate, the result of the circle center detection will find the radius of a circle like Eq. 4 which will maps the circle to the object to see if the results match the theorem.

MATHEMATIC ALGORITHM MODEL

A. Input → The input for mathematical algorithm model is an RGB image with circle image.

B. Proseses → The algorithm step for processing of mathematic algorithm model are following:

- Search for P_1 by finding the nearest pixel at x and y coordinates
- Search for P_2 by finding the nearest pixel at the farthest y and pixel coordinates in y coordinates
- Search for P_3 by finding the nearest pixel in x coordinates, and the farthest pixel in y coordinates
- Search for P_4 by finding the nearest pixel in y coordinate, and farthest pixel at x coordinate
- Looking for the center point with the Eq. 2 and Eq. 3
- Look for a radius of the formula Eq. 4
- Describes the result of center point and radius of circle with drawer image in C#

C. Output → From the process of mathematic algorithm model then it produces the output an image in the form of circled circular object with a known center point and radius.

IMPLEMENTATION OF ALGORITHM

At this stage is the application of mathematical algorithm model should be converted into a programming language. The programming language used is C# language with IDE BUILDER Microsoft Visual Studio. As for some Algorithm stages like the following are shown in Fig. 4.

A. Finding x_1 Value

The first step to seek P_1 , it is necessary to first describe the components contained in the P_1 . The first component is x_1 , drawing analogies to get the value x_1 can be seen in Fig. 4 and pseudocode to find the value x_1 can be written in the following Fig. 5 below.

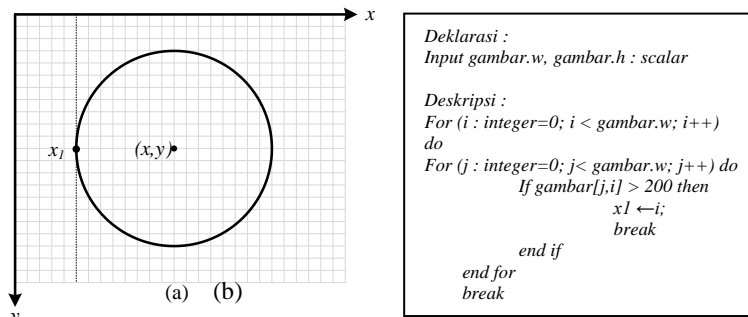


Figure 4. Finding x_1 value
(a) Coordinate Convention and (b) Pseudocode

The search for the value of x_1 starts with looping iterations from the smallest index of the component x coordinate which presents the length of the image from the input image and the smallest index of the component y presents the image width as described in the Fig. 5 above.

B. Finding x_2 Value

The value of x_2 is determined by finding the position of the pixel farthest from the maximum limit of the length of the input image that has a color with bright intensity, as illustrating in Fig 5.

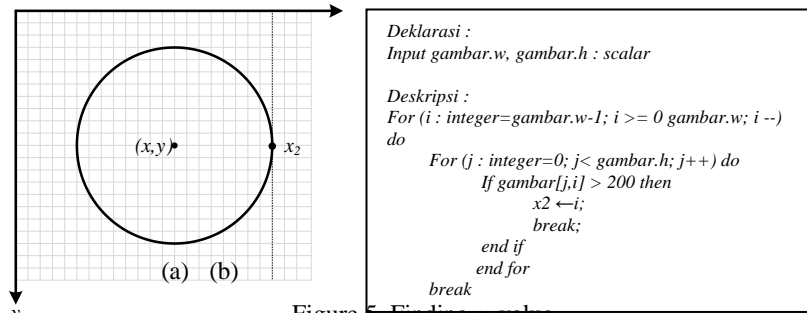


Figure 5. Finding x_2 value
(a) Coordinate Convention and (b) Pseudocode

The value of x_2 from Fig. 5(b) is obtained from conditioning the value of the intensity of the input object. If in the closest looping iteration of the maximum length to the minimum length there is a pixel intensity of more than 200, the coordinate value of the first pixel will be stored the x coordinate value and saved to a certain variable.

C. Finding y_1 Value

After the coordinate x_1 and x_2 values were found by using pseudocode Fig. 4 and Fig. 5, the next step is to found coordinate y_1 and y_2 value the illustration and the pseudocode to find y_1 and y_2 shown in Fig. 6 and Fig. 7 below.

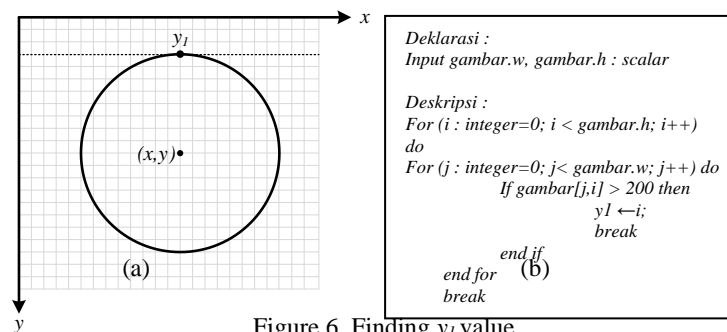


Figure 6. Finding y_1 value
(a) Coordinate Convention and (b) Pseudocode

The Difference in conditions in the detection of y_1 in Fig. 6 Is a looping iteration, looping iterations in the search value y_1 by looking for the closest value from the height of the image and then saved in variable y_1 saving the value. Break script in Fig. 6(b) works to stop the search for pixel intensity that is worth more than 200 if a pixel has been detected that meets these criteria. so that computer memory management is more optimal.

D. Finding y_2 Value

Search y_2 value has the same condition with the search value x_2 , where the difference in the search for the value of y_2 to be stored is the y coordinate value on pixels that have bright intensity as shown in Fig. 7 below.

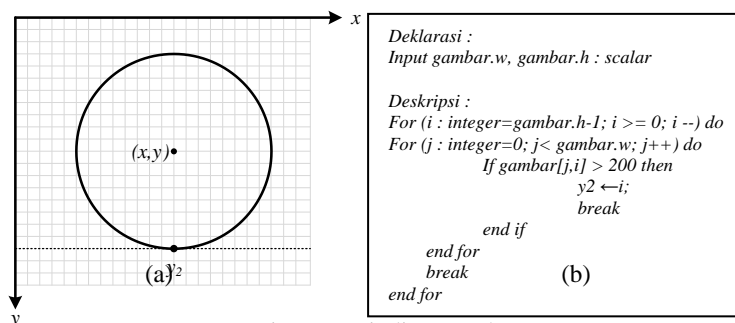


Figure 7. Finding y_2 value
(a) Coordinate Convention and (b) Pseudocode

The pseudocode from Fig. 7(a) to find the value y_2 can be written in the following Fig. 7(b)

E. Finding the Center and the Radius of Circle

The last step is to find the center point of the circle and the radius of the circle, parameters $(x_1, x_2, y_1, \text{ and } y_2)$ are used into the equation by using Eq. 2 and Eq. 3 to determine the center point and Eq. 4 to determine the radius. Pseudocode as shown in Fig. 8 as below.

```

Deklarasi :
Input gambar.w, gambar.h : scalar
Input hasilx,hasily,hasilx1,hasilx2 : integer
Input hasily1,hasily2,jari : integer

Deskripsi :
hasilx=hasilx1 + ((hasilx2 - hasilx1)/2);
hasily=hasily1 + ((hasily2 - hasily1)/2);
jari = hasilx2 - hasilx;

print hasilx
print hasily
print jari
    
```

Figure 8. Pseudocode Center Point and Radius

The pseudocode in Fig. 8 is the final part of the structure scheme of simple mathematical applications of the CHT algorithm. all pseudocode schemes are structural, the pseudocode above will run if all four pseudocodes have been executed before.

RESULTS AND DISCUSSION

The pseudocode will be implemented and applied in C# programming, and the results of the software design program for Pre-processing image and implemented mathematic process are shown in Fig. 9.

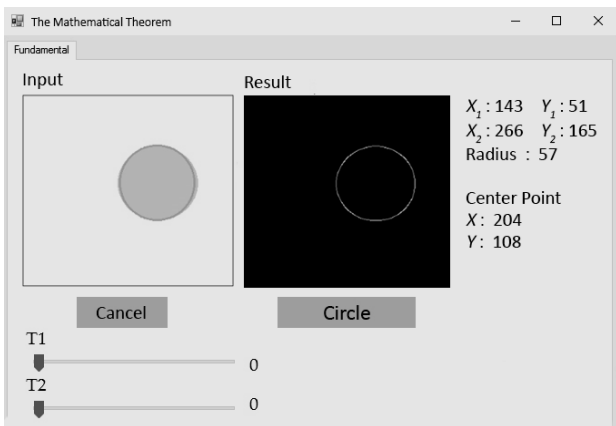


Figure 9. Screenshot Program

A. Pre-processing Image Result

At this stage, the RGB image as input will be changed into grayscale image and then edge detection object will take place. The result is detected by the mathematical theorem as described above. As shown in Fig. 10.

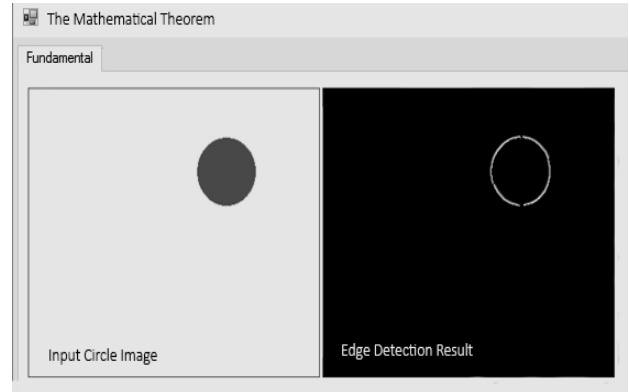


Figure 10. Pre-processing Image Result

edge detection in Fig. 10 to ensure there are no pixel noise in the search parameters P_1, P_2, P_3, P_4 as described in Fig. 3 previously.

B. Result of Mathematic Implementation

The results of Fig. 10 that has been applied to edge detection will be calculated according to the algorithm and pseudocode above as shown in Fig. 11.

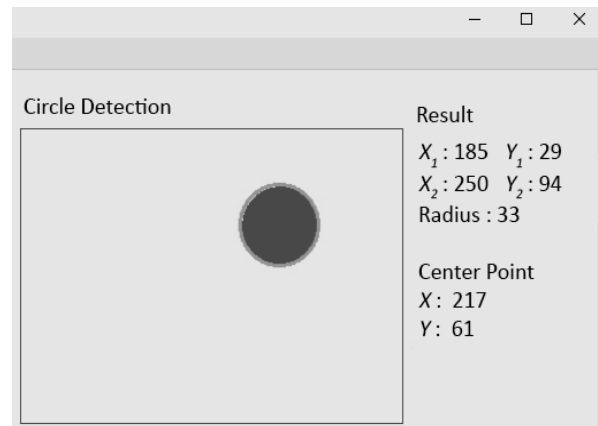






Figure 11. Result of Circle Detection

Following are the results of 4 times the detection of 4 circles on different colors and position. As shown in Table 1 as below

Table 1. Calculation of Radius Object

No	Experiment of Circle Sample		
	Image Result	Center Point (x, y)	Radius (pixel)
1		(148,126)	77
2		(77,75)	37
3		(204,108)	57
4		(217,61)	33

From table 1 the results obtained that calculation of radius of an object with a circular Hough transformation mathematical theorem can be applied to circle object and different colors and position.

SUMMARY

From the results of the study of the mathematical implementation of circle Hough transformation theorem model using c#.

1. The system results obtained that calculation of radius of an object with a circular Hough transformation mathematical theorem can be applied to circle object and different colors.
2. The System can calculate the radius of the object provided that it has gone through the pre-processing edge detection with canny operator and
3. The calculation of radius circular object with a circular Hough transformation mathematical theorem can be applied only to a perfectly circular object if the sample object is not a perfect circle, then calculations radius object will not be optimal.

ACKNOWLEDGMENT

Many thanks to the Department of Computer System, Computer Science Faculty, Sriwijaya University and all of the peoples who supports this article.

REFERENCES

[1] T. D. Orazio, C. Guaragnella, M. Leo, and A. Distanto, "A New Algorithm for Ball Recognition using Circle Hough Transform and Neural Classier," vol. 37, pp. 393–408, 2004.

[2] Passarella, Rossi, Sutarno and Ahmad Rezqy FF, "Implementation of Image Segmentation Using RGB Color Detection on Virtual Reality Robot as Spy Tool." *Proceeding of the Electrical Engineering Computer Science and Informatics 1.1* (2014): 374-379.

[3] Passarella, Rossi, Muhammad Fadli, and Sutarno, "Design Concept of Convexity Defect Method on Hand Gestures as Password Door Lock."

Computer Engineering and Applications Journal 2.3 (2013).

[4] Passarella, Rossi, and Osvari Arsalan, "Object Reconstruction from 2D Drawing sketch to 3D Object." *Computer Engineering and Applications Journal* 5.3 (2016): 119-126.

[5] I. Introduction, "Advances in Mathematical Models," vol. 69, no. 5, 1981.

[6] W. Lip, J. Shiun, W. Shin, H. Hashim, and C. Tin, "Review of Distributed Generation (DG) System Planning and Optimization Techniques : Comparison of Numerical and Mathematical Modeling Methods," *Renew Sustain Energy Rev.*, vol. 67, pp. 531–573, 2017.

[7] P. Mukhopadhyay and B. B. Chaudhuri, "Author Accepted Manuscript A Survey of Hough Transform," *Pattern Recognition.*, 2014.

[8] M. W. Spratling, "A neural implementation of the Hough transform and the advantages of explaining away," *IMAVIS*, 2016.

[9] A. O. Djekoune, K. Messaoudi, and K. Amara, "Incremental Circle Hough Transform : An Improved Method for Circle Detection," *Opt. - Int. J. Light Electron Opt.*, 2015.

[10] H. Yazid, M. Harun, S. Mohd, and A. A. Mohamed, "Circular Discontinuities Detection in Welded Joints using Circular Hough Transform," vol. 40, pp. 594–601, 2007.

[11] M. Mirzaei and H. K. Rafsanjani, "An Automatic Algorithm for Determination of the Nanoparticles from TEM Images Using Circular Hough Transform," *Micron*, 2017.

[12] A. Jalil, R. Sahak, A. Saparon, and U. T. Mara, "Iris Localization Using Color Segmentation and Circular Hough Transform," no. December, pp. 784–788, 2012.

[13] D. Putra, "Pengolahan Citra Digital," Penerbit Andi, Yogyakarta, 2010