Circle Object Recognition Based on Monocular Vision for Home Security Robot

Shih-An Li, Yi-Chun Lin, Chung-Wei Weng, Yi-Hong Chen, Chia-Hung Lo, Min-Hao Yang, Ming-Hua Hsieh,
Ching-Chang Wong
Department of Electrical Engineering,
Tamkang University,
New Taipei City, 251, Taiwan
lishyhan@gmail.com

Abstract—**In this paper, a circle object recognition method based on monocular vision for the home security robots is proposed. This vision system is able to process image and recognize a colored ball rapidly. The proposed method consists of two sub-modules, which are the object segmentation module and the circle detection module. In order to implement the object segmentation, the color feature is applied to find out the region of the object. After the region of the object is determined, a fast randomized circle detection method is used by checks if there have enough number radius which are the same in a circle in the region. Because of the double recognition, this system can improve the precision for detecting a colored ball. The proposed method is tested on a home security robot to find a ball. The experimental results illustrate the effectiveness of the proposed method.**

Keywords-object recognitio; computer vision; circle detection; digital image processing; robot;

I. INTRODUCTION

As the average age rising, more and more families need to face the issue of taking care of the elders and the children. It is becoming a very important issue. Besides, the house service is also a huge request when people are busy for working. Some of the researches about the home service in the industry area or academic area are presented recently. [1].

This paper presents a circle object recognition method for a monocular vision system of home security robots. The robot with the proposed method is able to search and pick up the colored ball autonomously. The robot is shown in Fig.1. The robot picks up the toys at home by using the propose method. A camera installed in front of robot is for capturing image. A roll style brush is used to pick up the colored ball.

Figure 1. An autonomous home security robot

Numbers of the research have been proposed for image pattern recognition. Most of them emphasize the precision of detecting and the speed of image processing. [2-3]. However, home security robot needs a real-time image processing system. The presented method in this paper without enormous algorithm but still has capability of image processing in real-time and recognizes a colored ball accurately.

This paper is organized as follows: Section 2 introduces the vision system architecture of robot, then the image process core is describing in section 3. The experiment results will be present in section 4, and section 5 is conclusions.

II. VISION SYSTEM ARCHITECTURE

The vision system architecture of the home security robot in this paper is shown in Fig.2. This system can be separated into two parts, the user interface part and the image processing part. There is a common database, which is the data center of the communication. The image capture module will store the pixel data of image to the common database. Both of color model interface and setup parameters interface will store the range of color model and parameters to database by user input data. Those data is adjusted according to the environment situation. The image processing core can use the parameters to detect object. After the image processing core module obtain and store the object information, the preview image interface module will display the result.
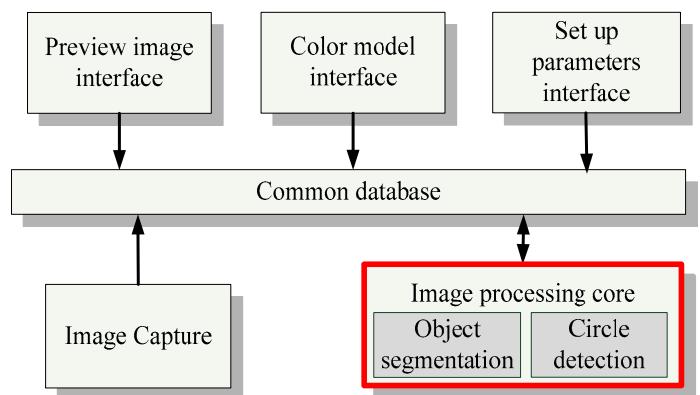

Figure 2. The visual system architecture of the home security robot

There are two main modules in image processing core; one is the object segmentation module and another is the circle

detection module. The object segmentation module is implemented to determine the colored object with target color. The circle detection module is implemented to detect the circle object according to the radius feature of the object, which is determined by the object segmentation module. The details of the image processing will be described in section 3.

III. IMAGE PROCESSING CORE

The flow chart of image processing is shown in Fig. 3. The pixel data of image is stored to the common database by the image capture module. The color model of the target color in the color space is accomplished the step of build color model. This will be done before starting to analyze the image. The input pixel data is compared with the target color in the step of thresholding. The pixel color which is similar to the target color is selected for object segmentation. Use these satisfied pixels to get object region by object segmentation. If there is no satisfied color object, the process will start to capture next image. The method of object segmentation will be described in section 3.1. Once the system determines the objects region by the above steps, the edge of the objects in the region will be used to calculate the radius feature by a randomized circle detection method. The method of obtain edge points and randomized circle detection will be described in section 3.2. If the system can't find any object by the randomized circle detection, the system starts to capture the next image.

A. Image segmentation module

Thresholding processing is used to create binary images. The pixel with the similar color with the target color will be marked as 1; others will be marked as 0. The pixel with or without the similar color can be separated in this step. After the thresholding processing, noise reducing processing is necessary. An Opening algorithm [4] is chosen here. This algorithm uses erosion algorithm and dilation algorithm in order. The Opening algorithm can reduce some noises of the image.
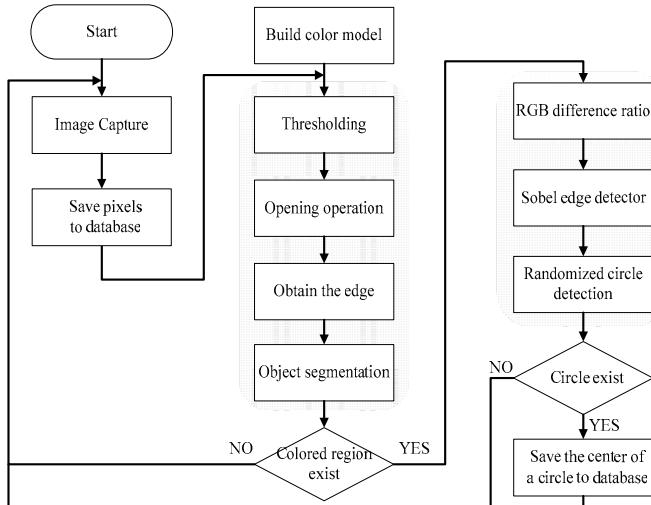

Figure 3. The flow chart of image processing

A fast edge detection method can speed up the image processing. After the image finishes the Opening Algorithm, each pixel has its own status. The status of pixel which has

similar color with the target color is 1; otherwise the status is 0. The method of the edge detection is checking a pixel with its neighbor, including right pixel, top pixel and bottom pixel. If the pixel has a different status from its neighbor, it means the edge happening.

Fig. 4. shows an example of the processing result. In the figure, there are two groups of the pixel, which has the right color. Each grid means a pixel. The grids with the status "1" mean the pixel with right color, others are not. According to the edge detection method, the yellow grid can be determined by comparing the pixel with its neighbor. It means the yellow grid is the edge point. The examples show that this method is able to find the edge of the pixel group.

Follows describe the object segmentation technology [5], which for getting the coordinates information of the color regions, so that it can choose most probable colored regions to detect if there is a circle in the area.

The technology includes into two steps. The first step scan the pixels from left to right, top to bottom of the image, until it find a edge point. The edge point is considered as a standard point. The second step is searching the near eight pixels based on the standard point. If any edge point can be found in this step, it will be remembered and added into a queue. Then compare the standard point with the coordinates of the boundary to update the coordinates of boundary. Once finish the search of all of the standard point's neighbors, an edge point from the queue will be chosen as a new standard point. Repeat these steps and update the coordinates of the boundary until there is no edge point in the queue, and the coordinates of the boundary is the range of the object. And then go back to the first step to search the next edge point until the whole image is searched completely. Fig.5 shows a sample of object segmentation. The red grids mean a coordinates which is the upper left and the lower right of a colored region. The two coordinates can represent the range of the object.
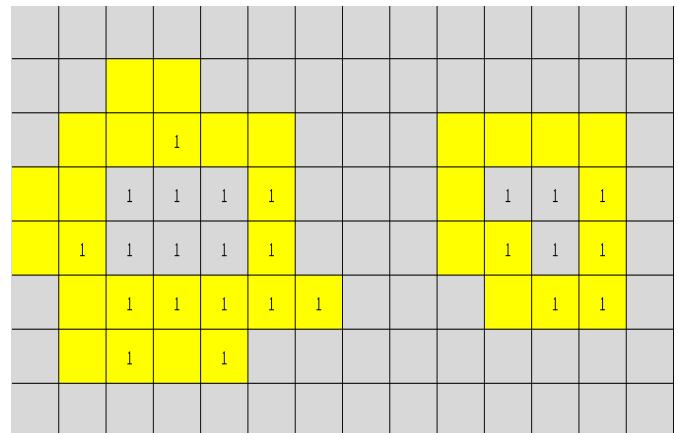

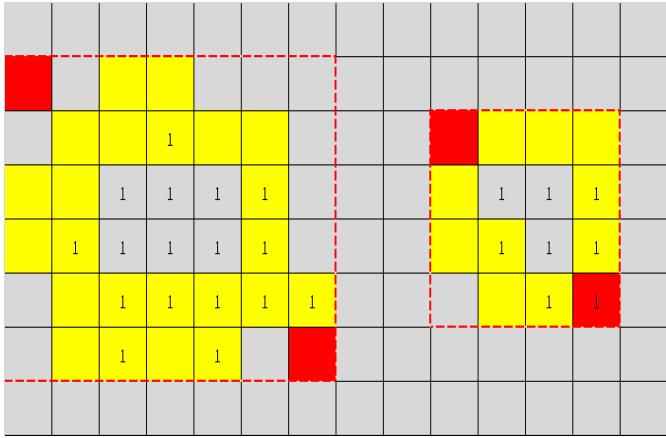Figure 4. A sample after the process of obtain the edge

Figure 5.   A sample of object segmentation

B.   Circle detection module

After obtain the boundary information of colored region by the module of object segmentation, it will conjecture that a colored ball exists in the colored regions. So this paper uses a method of randomized circle detection [6-7] to detect whether there is a circular object in colored regions.

First step is separated the object from the back ground color. Use the operation of RGB difference ratio is easier separate the more vivid target object from the duller background, and it highlights the edge of object. The equation can be expressed as follows:

$$pixel = \frac{Max(R,G,B) - Min(R,G,B)}{255} \in [0,1] \qquad (1)$$

All target pixels be computed by this equation, then normalize the data between 0~255. It's a gray image. Using Sobel operator [8] to get the edge points set in the gray image. Assume the set is $V = \{(x, y)\}$. In a Cartesian coordinate system, using Pythagorean theorem [9] can obtain the circle center coordinates (a, b) and radius r is the set of all points (x, y) such that:

$$(x-a)^2 + (y-b)^2 = r^2 \qquad (2)$$

The equation can be rewritten as follows:

$$2xa + 2yb + d = x^2 + y^2$$
$$where \ d = r^2 - a^2 - b^2 \qquad (3)$$

Let $v_i = (x_i, y_i)$, $i = 1,2,3$ are three points which is randomized selection in V. Assume $v_1 、 v_2 、 v_3$ are not collinear, and assure of the three points must greater than a distance of pixels. Substituting the three points into (3) will obtain (4):

$$\begin{cases} 2x_1 a_{123} + 2y_1 b_{123} + d_{123} = x_1^2 + y_1^2 \\ 2x_2 a_{123} + 2y_2 b_{123} + d_{123} = x_2^2 + y_2^2 \\ 2x_3 a_{123} + 2y_3 b_{123} + d_{123} = x_3^2 + y_3^2 \end{cases} \qquad (4)$$

$$where \ d_{123} = r_{123}^2 - a_{123}^2 - b_{123}^2$$

Where r_{123} is radius, (a_{123}, b_{123}) is the circle center coordinates. Equation (4) can be rewritten a linear system and Using Gaussian elimination will obtain equation such as (5):

$$\begin{bmatrix} 2x_1 & 2y_1 & 1 \\ 2(x_2 - x_1) & 2(y_2 - y_1) & 0 \\ 2(x_3 - x_1) & 2(y_3 - y_1) & 0 \end{bmatrix} \begin{bmatrix} a_{123} \\ b_{123} \\ d_{123} \end{bmatrix} = \begin{bmatrix} x_1^2 + y_1^2 \\ x_2^2 + y_2^2 - (x_1^2 + y_1^2) \\ x_3^2 + y_3^2 - (x_1^2 + y_1^2) \end{bmatrix} \qquad (5)$$

It has known that $v_1 、 v_2 、 v_3$ are not collinear from assumption, by using Cramer's rule and (6) will get the solution of circle center:

$$a_{123} = \frac{\begin{vmatrix} x_2^2 + y_2^2 - (x_1^2 + y_1^2) & 2(y_2 - y_1) \\ x_3^2 + y_3^2 - (x_1^2 + y_1^2) & 2(y_3 - y_1) \end{vmatrix}}{4((x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1))}$$

$$b_{123} = \frac{\begin{vmatrix} 2(x_2 - x_1) & x_2^2 + y_2^2 - (x_1^2 + y_1^2) \\ 2(x_3 - x_1) & x_3^2 + y_3^2 - (x_1^2 + y_1^2) \end{vmatrix}}{4((x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1))} \qquad (6)$$

By using the solution of circle center and (2) will get the circle radius, as describes in (7):

$$r_{123} = \sqrt{(x_i - a_{123})^2 + (y_i - b_{123})^2} , i = 1,2,3 \qquad (7)$$

Determining the distance from circle center to other edge points, it will be considered the circle get a vote if the distance less than the setting threshold. It will be considered a real circle if the numbers of vote large than the setting threshold, otherwise delete $v_1 、 v_2 、 v_3$ from the edge set, and repeat above steps. Finally, the circle center and circle radius information of a circle are stored to database. Otherwise, if the remaining edge points less than a setting threshold, the circle detection will be terminated. It means there is no circular object in the colored region.

IV.   EXPERIMENTS

The experimental results of object recognition method are shown in TABLE I. In the image, the red regions will be segmented. The object with the red color is framed by the red rectangular. The circle detection can find the circle object correctly. The object with circle feature is determined by the

proposed method. Green circle shows the results. Besides, TABLE II presents the executions time of this method, it shown that this method has well recognize ability in real time. So, it can prove the proposed circle object recognition method in this paper is effective.

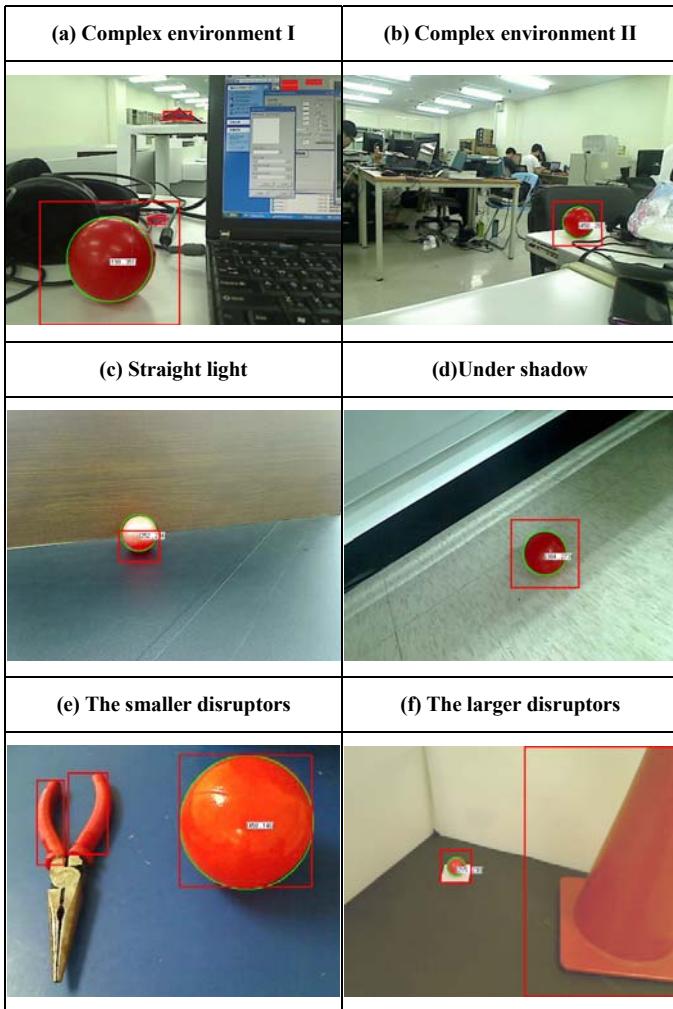TABLE I.   THE EXPERIMENT RESULTS AT DIFFERENT ENVIRONMENT

(a) Complex environment I	(b) Complex environment II
(c) Straight light	**(d)Under shadow**
(e) The smaller disruptors	**(f) The larger disruptors**

TABLE II.   THE EXECUTIONS TIME AT DIFFERENT ENVIRONMENT

Experimental results numbers	Executions times (times/sec)	
	Capture a frame times	*Execute recognition times*
(a)	30	14-18
(b)	30	20-22
(c)	30	20-22
(d)	30	18-20
(e)	30	16-18
(f)	30	14-16

V.   CONCLUSIONS

In this paper, a circle object recognition method based on monocular vision for the home security robots is proposed. This proposed system is able to find a colored ball rapidly. Two sub-modules are implemented in this system. The proposed method checks the objects with color and radius feature in a circle - all the radius are the same. The experimental results illustrate the effectiveness of the proposed method.

REFERENCES

[1] Department of Investment Services, Ministry of Economic Affairs - Intelligent Robotics Industry,
http://www.dois.moea.gov.tw/asp/iaaio.asp

[2] L.Q. Jia., H.M. Liu., Z.H. Wang.; H. Chen.: An effective non-HT circle detection for centers and radii. In: 2011 International Conference on Machine Learning and Cybernetics (ICMLC), pp814-818. IEEE Press, China.

[3] Ceccarelli, M., Petrosino, A.: Circle detection based on orientation matching. In: 2001. Proceedings. 11th International Conference on Image Analysis and Processing, pp119-124. IEEE press, Italy.

[4] Rafael C., Richard E.: Digital Image Processing 3/e. Prentice Hall, United States (2008)

[5] C.J. Yu.: Hardware/Software Co-design of Real-time Human Face Detection. Master's thesis. Department of Electrical Engineering, Tamkang University. (2010)

[6] T.C. Chen, K.L. Chung.: An efficient randomized algorithm for detecting circles. Computer. Vision and Image Understanding 83, 172–191 (2011)

[7] K.L. Chung.: Image Processing and Computer Vision. Tung Hua, Taiwan. (2008)

[8] Wikipedia-Sobel operator,
URL=http://en.wikipedia.org/wiki/Sobel_operator

[9] Wikipedia- Pythagorean theorem,
URL=http://en.wikipedia.org/wiki/Pythagorean_theorem

[10] Wikipedia-RGB color model,
URL=http://en.wikipedia.org/wiki/RGB_color_model