Robotic arm grasping coloured balls using MATLAB

I. Introduction:

The application of machine vision in robotic arms has been increasing in the industries to make the processes more and more autonomous. By the use of image processing tools and various algorithms, we enhance the the ability of the robot to perform desired tasks on its own with increasing accuracy.

This report shows the machine vision's role before manipulating the robotic arm to grasp and place colured balls in a box with the help MATLAB and its image processing toolbox.

II. Methodology

There are five steps in identifying a ball of a specific color among red, blue and green balls. We input the image for the image processing and then shape matching check if there is a ball. If yes, the robotic arm can move the camera lens pointing to the center of the ball and then grasp it. Finally, the ball can be placed the assigned box according to the ball color.

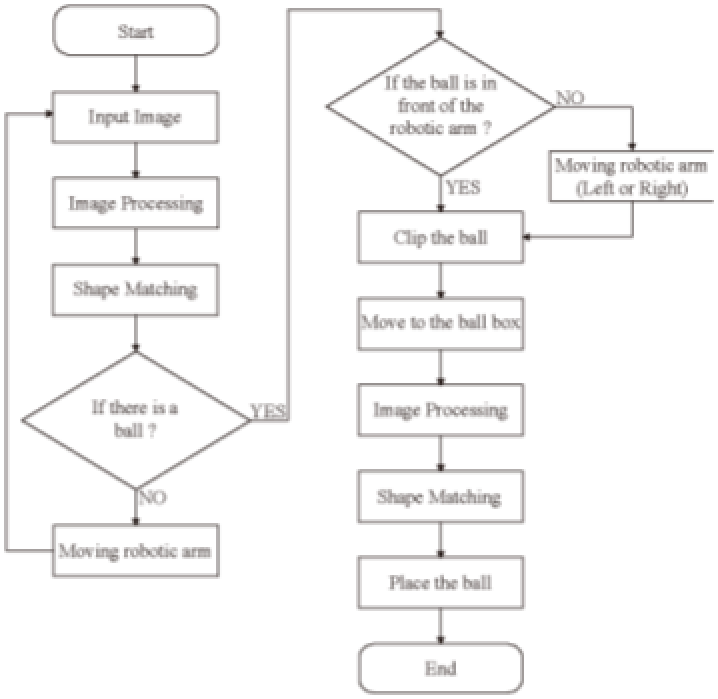


Figure 1. Flowchart of the system structure

If the robotic arm fails to locate a ball at a certain place, it moves from that position and searches at the current position by scanning it. This process is repeated continuously till all the balls are grabbed and placed.

III.Algorithm

In the image processing, the noise, light and other interferences should be reduced in order to improve the shape matching error after capturing images. The following algorithms are used, in the the given sequence to complete the task:

1.Light compensation:

By increasing the values of the intensities in an image we wil be able to get a brighter image and the darker objects can be viewed clearly. But, by this method the already visible parts of the image will be overexposed. Hence, we can decrease or increase the intensities of elements in an image in a desired way by light compensation method as shown in equation (1) and (2). Light compensation is necessary only in the cases where 0.95*gray_max is greater than 100, gray_max being the maximum gray value or the value of the pixel whose intensity is the highest.

$$\begin{aligned} AveR &= \frac{1}{N}\sum_{i,j\in RW} I_R(i,j) \\ AveG &= \frac{1}{N}\sum_{i,j\in RW} I_G(i,j) \\ AveB &= \frac{1}{N}\sum_{i,j\in RW} I_B(i,j) \end{aligned} \tag{1}$$

$$\begin{aligned} R_O(i,j) &= I_R(i,j)\times\frac{255}{AveR} \\ G_O(i,j) &= I_G(i,j)\times\frac{255}{AveG} \\ B_O(i,j) &= I_B(i,j)\times\frac{255}{AveB} \end{aligned} \tag{2}$$

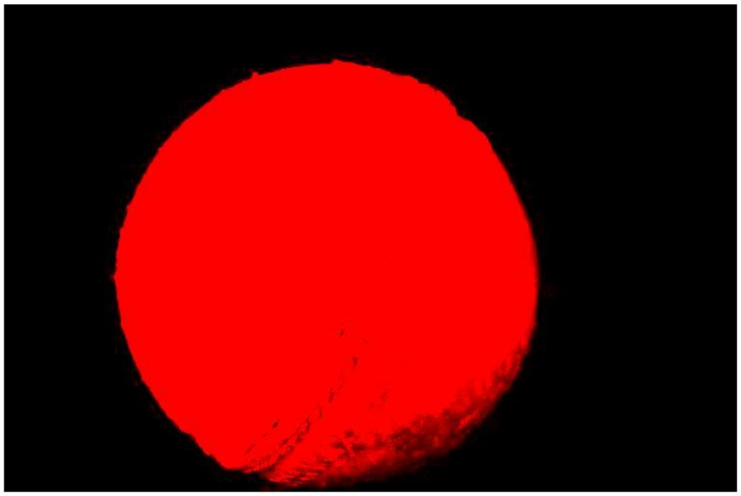


Figure 2.Light compensated red matrix

2.Noise filter:

The smoothing out of the image or elimination of noise helps us by giving more accurate results in the further steps. We perform convolution with the averaging filter which here is a 7x7 mask.

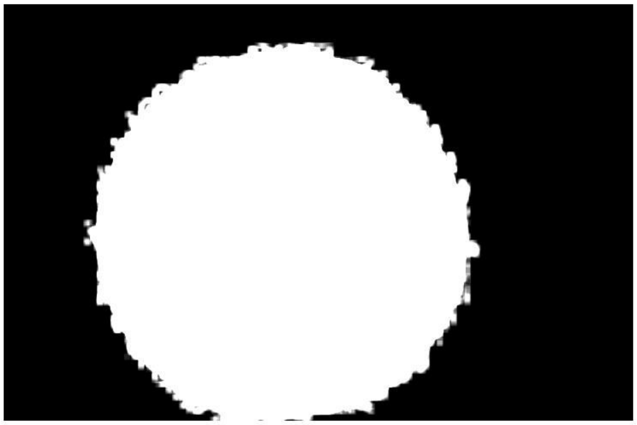


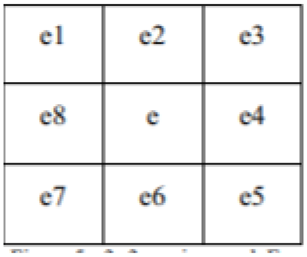
Figure 3.Smoothing the grayscale image

3.Morphology:

The erosion and dilation operations are performed on images to get nearly perfect circles by eliminating noisy parts.

a) Erosion operation: In this operation, after referring to the mask, firstly we justify if the mask element e is at 1 status in the binary image. If yes, continue to check if at least one of the surrounding elements (e1~e8) has a value in its respective location of A matrix. If the condition is satisfied, “e” is set as 1. Otherwise “e” is set as 0.

$$e = e1 \cap e2 \cap e3 \cap e4 \cap e5 \cap e6 \cap e7 \cap e8$$



e1	e2	e3
e8	e	e4
e7	e6	e5

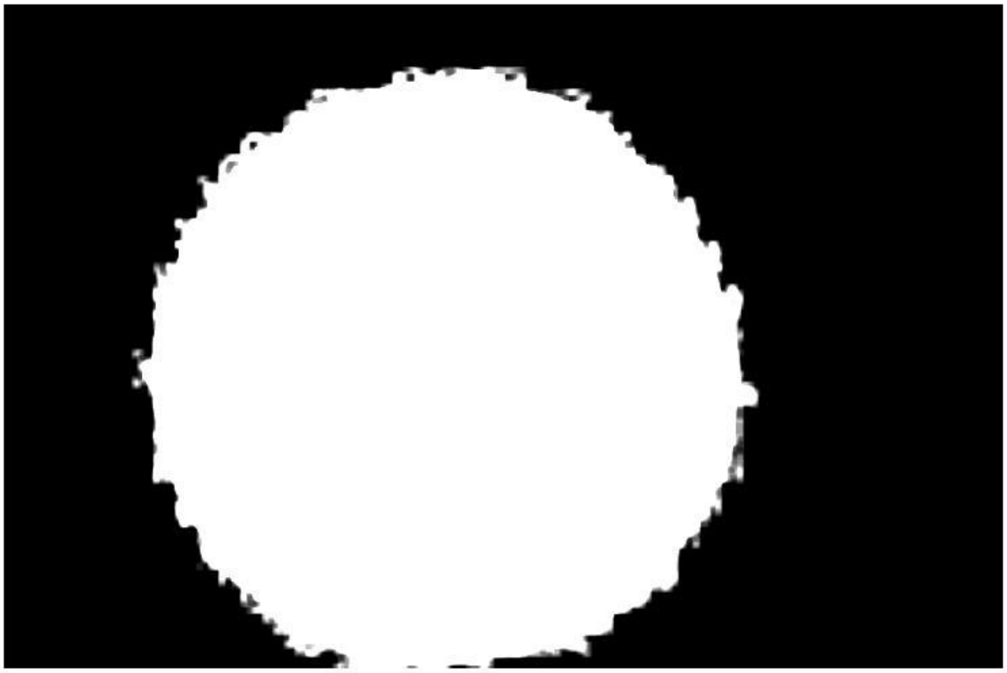
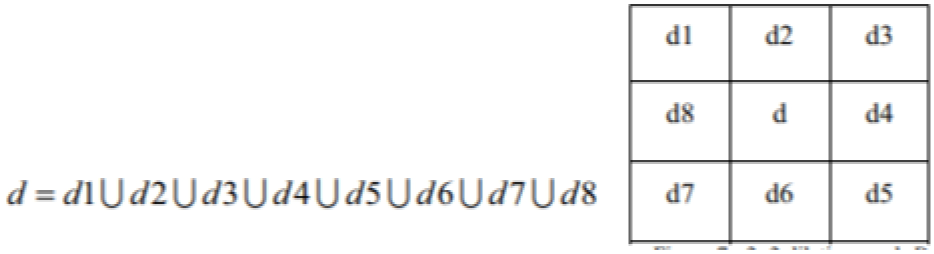


Figure 4. Eroded image

b) Dilation operation:In this operation, after referring to the mask, firstly we justify if the mask element d is at 1 status in the binary image. If yes, continue to check if at least one of the surrounding elements (d1~d8) has a value in its respective location of A matrix. If the condition is satisfied, “d” is set as 1. Otherwise “d” is set as 0.



$$d = d1 \cup d2 \cup d3 \cup d4 \cup d5 \cup d6 \cup d7 \cup d8$$

d1	d2	d3
d8	d	d4
d7	d6	d5

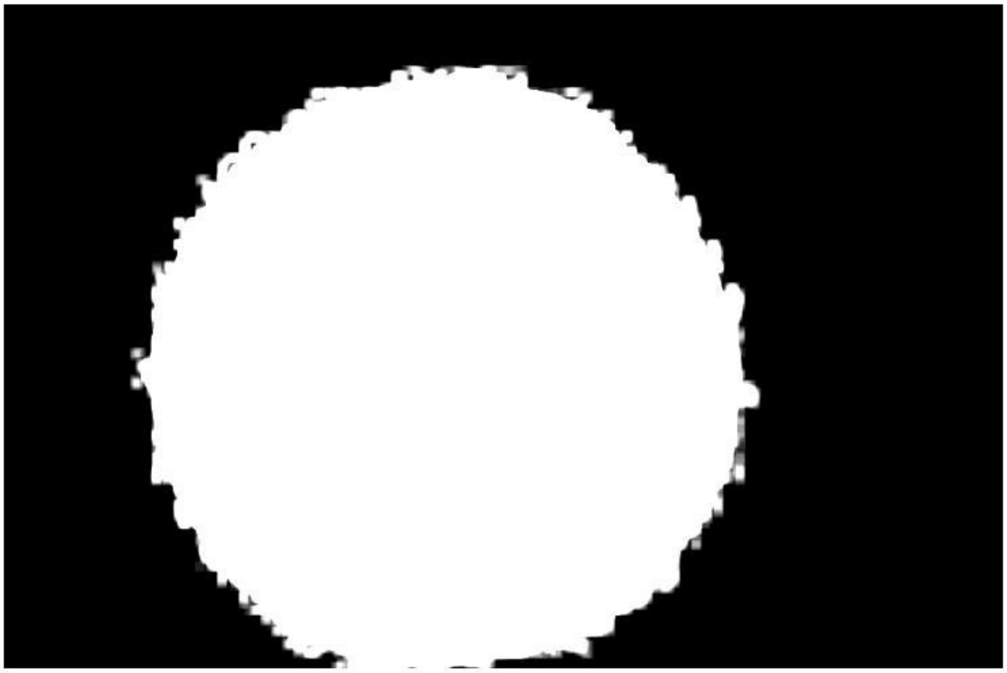
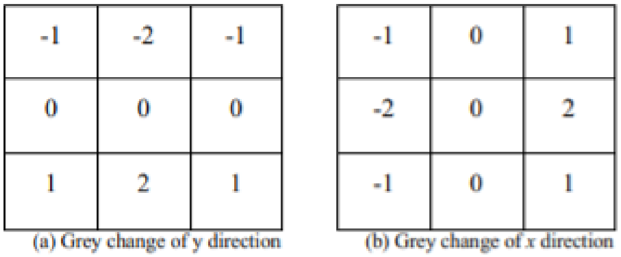


Figure 5. Dilated image

4.Edge detection:

By edge detection of objects, we can find out the approximate shape and reduce the computational procedure. Here,we applied Sobel Operator for x and y differential calculation with mask method and then, ∇f can be obtained.

$$\nabla f = \sqrt{(\nabla_x f)^2 + (\nabla_y f)^2}$$



-1	-2	-1
0	0	0
1	2	1

(a) Grey change of y direction

-1	0	1
-2	0	2
-1	0	1

(b) Grey change of x direction

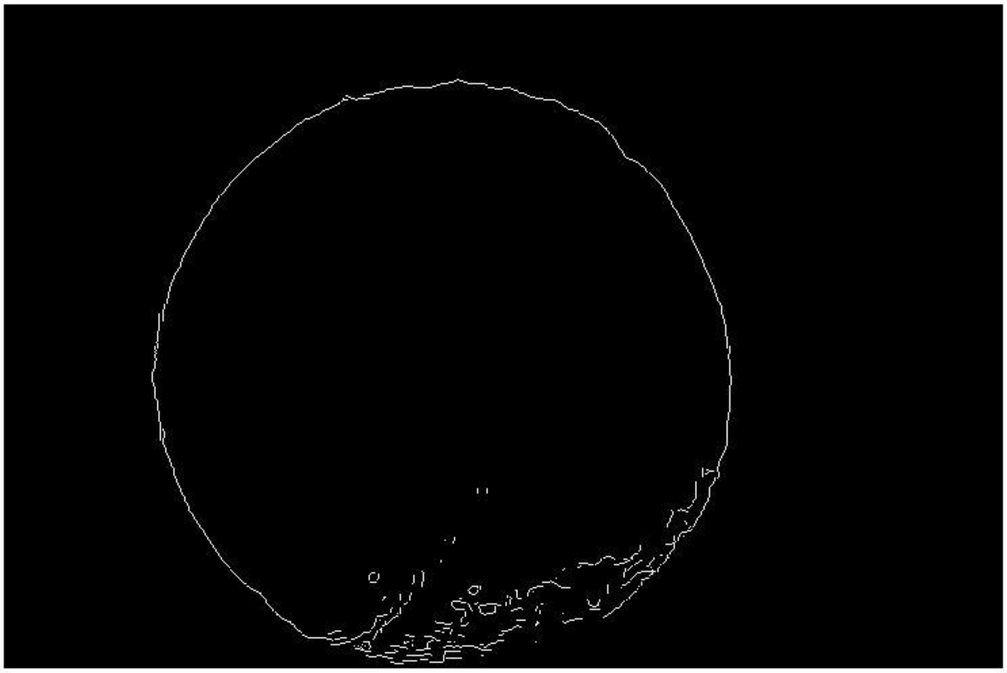


Figure 6. Edges of the scanned object

5.Convex hull algorithm

As we can see from figure (6), the detected edge is not a perfect circle. Hence, we search for a boundary point and the clockwise or counterclockwise around the circle outward. When the search point is covered, all points on the plane are the most peripheral points covered. We finally get the outline of the circle.

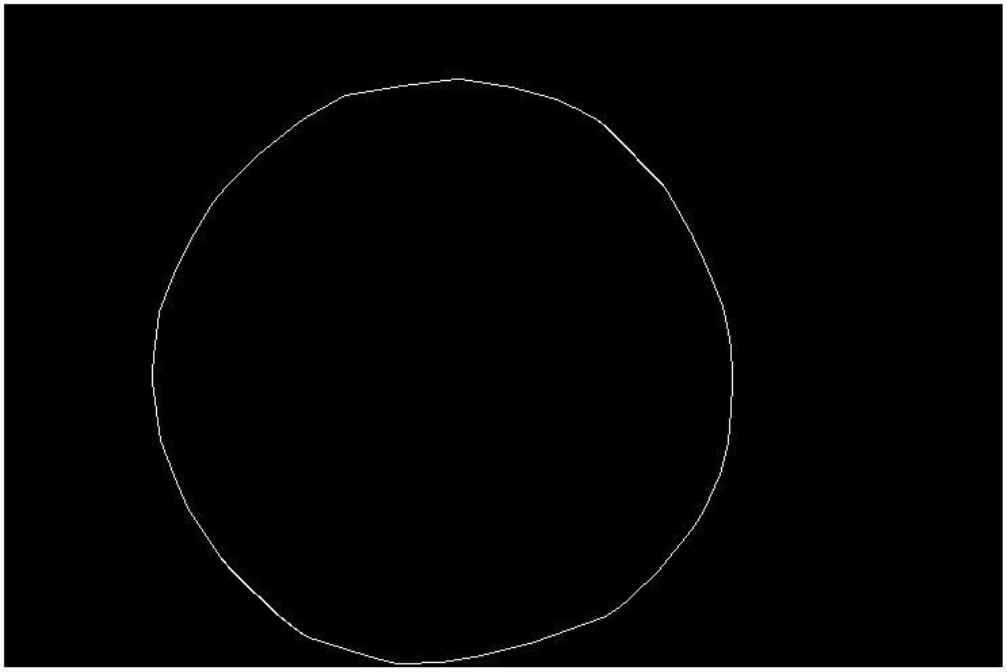


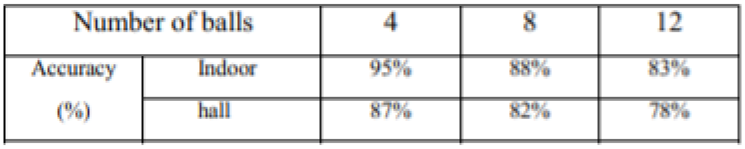
Figure 7. Outline of the object

IV.Experiments Done

The digital images of different coloured balls were tested using the MATLAB code.

V.Evaluation Metrics

The success rate of correct grabbing the balls under different number of color balls and environment is as given below:



Number of balls		4	8	12
Accuracy	Indoor	95%	88%	83%
(%)	hall	87%	82%	78%

VI.Analysis

When the number of balls increases, light reflection will affect the recognition rate of color ball between the ball and ball. Although it can be obtained higher success rate under plenty of lighting indoor than less bright hall, in general speech, the less bright hall has also a good success rate.

VII.Conclusion

The process of grabbing and placing is affected by surroundings, such as lighting or rough platforms. A more robust code which considers practical aspects and which works uniquely for different positions of the balls will increase the accuracy. On the whole, this method certainly is useful for performing low-cost and trivial tasks.

VIII.References

Robotic Arm Grasping and Placing Using Edge Visual Detection System;Guo-Shing Huang, Hsiung-Cheng Lin, and Po-Cheng Chen Department of Electronic Engineering, National Chin-Yi University of Technology Taichung 41101, Taiwan, ROC;Proceedings of 2011 8th Asian Control Conference (ASCC)

Automatic Edge Detection Method for the Mobile Robot Application Wangbeon Lee, Dongsu Kim and Inso Kweon;Proceedings of the 2003 IEEWRSJ Intl. Conference on Intelligent Robots and Systems

LIGHT COMPENSATION Chih-Fong Chang (張致豐), Chiou-Shann Fuh (傅楸善) Dept. of Computer Science and Information Engineering, National Taiwan University

IX.Topic Suggestion

The use of light sensors, proximity sensors etc. on the robotic arm to help in conditional execution of algorithm and methods of pick and scan to avoid reflections of other objects on the prime object.