An Object Recognition and Identification System Using the Harris Corner Detection Method

T. Kitti, T. Jaruwan, and T. Chaiyapon

Abstract—This paper presents an object recognition and identification system using the Harris Corner Detection method. The process starts from imported images into the system by a webcam, detected image edge by canny edge detection, recognized the object by Harris Corner Detection, and separated the objects by the robot arm. Three object types; triangle, rectangular and, rigid circle are used. The results showed that the objects can be isolated 94%, 95%, and 99% for triangular, rectangular, and rigid circle respectively.

Index Terms—Harris Corner Detection, canny edge detection, object recognition.

I. INTRODUCTION

Nowadays, the robots are used widely in many applications such as automotive industry, rescue and security areas, medical exploration robotic, etc. Manipulator arms are employed in the industry as the human arm for welding, lifting, separating, etc. Sometimes referred to as robot arm also means that the robot in industry. The robots will be played greater roles in the industry as much as the robot control algorithms are developed. They will work in various hazardous jobs such as lifting steel into furnace, job-related chemicals, repetitive monotonous work such as lifting or packing the objects in production line, desired quality such as welding and cutting, or the jobs that required highly skills such as line welding, laser welding, etc.

In this research, we are presented an object recognition and identification system using the Harris Corner Detection method. There are many researches that used the Corner Detection method. Honda presents methods for enhancing Corner detection by comparison with Harris Corner Detection Method and other method [1]. Nain and others use first order difference of chain codes called shape numbers for corner detection [2]. Alshennawy and Aly introduce the fuzzy logic reasoning to detect the digital images edge [3]. Banerjee and other present a support vector machine based algorithm for corner detection [4]. Saeedi, Lawrence and Lowe introduced a binary corner detector and comparison with Harris method and SUSAN method [5]. Robert presents a variant of the morphological closing operator for corner detection [6]. Teixeira and other applied the Harris corner-detector algorithms make use of graphics processing units (GPU) provided by commodity hardware [7]. A curvature-based corner detector detects both fine and coarse features accurately at low computational cost which was presented by Chen He and Yung [8].

In our work, a robot arm was designed to use a canny edge detection method to detect the image edge [9] and to use Harris corner detection method to recognize, identify, and separate the objects. The triangle, rectangular, and rigid circle objects are used to verify the system.

II. IMAGE PREPROCESSING AND ENHANCE

The images of the object that used in our experimental have different quality such as brightness, contrast, or noise. The fuzzy image processing (FIP) is used to adjust brightness of the image [3]. The process includes image fuzzification, membership modification, and image defuzzification which can be shown in Fig. 1.

To adjust the brightness of the image, the INT operator with the following methods is used. The original image matrix X of N x M is the array of fuzzy singleton of each member represented by the degree of brightness levels p, p = 0,1,2,3,...P-1 with p values between 0 and 255 can be written as follows.

\[
X = \left( \begin{array}{cccc}
\frac{\mu_{x1}/x1}{1} & \frac{\mu_{x2}/x2}{1} & \cdots & \frac{\mu_{xM}/xM}{1} \\
\frac{\mu_{x1}/x1}{2} & \frac{\mu_{x2}/x2}{2} & \cdots & \frac{\mu_{xM}/xM}{2} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{\mu_{x1}/x1}{N} & \frac{\mu_{x2}/x2}{N} & \cdots & \frac{\mu_{xM}/xM}{N}
\end{array} \right)
\]

where \(0 \leq \mu_{xm} \leq 1, m = 1,2,..M, n = 1,2,..N\). The equation (1) is used to calculate.

\[
T_i(\mu_{xm}) = T_i(\mu_{xm}) = 2\mu_{xm}^2, \quad 0 \leq \mu_{xm} \leq 0.5
\]

\[
T_i(\mu_{xm}) = 1 - 2(1 - \mu_{xm})^2, \quad 0.5 \leq \mu_{xm} \leq 1
\]

Adjust each pixel of image brightness as follows, the values greater than 0.5 increases the brightness of the pixels, the values less than 0.5 decreases the brightness of the pixels. The procedure of the fuzzy image enhancement is shown in Fig. 2.
III. EASE OF EDGE DETECTION USING CANNY METHOD[9]

The Canny method is used for the edge detection. The steps are described as follows.

A. Change the Image Color

The image colors are changed to gray scale.

B. Gaussian Filter

Gaussian Filter is used for noise reduction and blurs the image by applying the Gaussian Filter convolution with the original images. The graph of Gaussian filter is shown in Fig. 3. Algorithms for noise reduction are derived from the calculation of the mask size. If the mask size is large, the noise is significantly reduced. If the mask size is small, the noise is less reduced. If the mask size is too large, the edge of the small details is missing. The smooth image can be calculated by the equation (2)

\[ S[i,j] = G[i,j,\sigma] \times I[i,j] \]  

When \( I[i,j] \) the position of the pixel is, \( G[i,j,\sigma] \) is Gaussian Smooth filter, \( \sigma \) is spread of Gaussian and \( S[i,j] \) is Smooth image.

C. Gradient Calculation

Divide the Gradient Calculation into two parts: \( x, y \) partial derivatives which can be calculated by rectangular to polar conversion equation (5)-(6).

\[ P[i,j] \approx (S[i,j+1]-S[i,j]+S[i+1,j+1]-S[i+1,j])/2 \]  

\[ Q[i,j] \approx (S[i,j]-S[i+1,j]+S[i,j+1]-S[i+1,j+1])/2 \]

Magnitude and direction of gradient was built from \( x, y \) partial derivatives which can be calculated by rectangular to polar conversion equation (5)-(6).

\[ M[i,j] = \sqrt{P[i,j]^2+Q[i,j]^2} \]  

\[ \theta[i,j] = \arctan(Q[i,j],P[i,j]) \]

From the equation (6), it can be replaced the function \( \arctan (x, y) \) to find \( \theta \).

D. Nonmaxima Suppression

The pixels of edge detection from canny method are the highest values of gray scale level and the same direction of gradient. Such a method has a thin edge only 1 pixel. The Image pixels after making Nonmaxima Suppression are zero at all points except the local maxima points which are still the old value.

E. Thresholding

This paper used the concept of threshold in the final process of the edge detection for reduced noise or the patterned surface of the object in the image. The threshold value consists of 2 characters, \( T1 = \) High Threshold, \( T2 = \) Low Threshold. If the pixels have values of gray scale level higher than \( T1 \), the result is a edge region. If the pixels have values of gray scale level less than \( T2 \), the result is not edge region. If the pixels have values of gray scale level between \( T1 \) and \( T2 \), the result is depending on The pixels neighboring of the surrounding.

IV. HARRIS CORNER DETECTION[10][11]

The Harris Method is used for object corner detection. This method does not change the value of scale, noise, rotation, the Variance of light by calculating the local auto-correlation function of a signal from measures the local changes of the signal with patches shifted by a small amount in different directions as given in equation (7).

\[ c(x,y) = \sum [I(x_i,y_j) - I(x_i + \Delta x, y_j + \Delta y)]^2 \]  

When \( w(x,y) \) is a window function, \( I(x+u,y+v) \) is a shifted intensity and \( I(x+y) \) is an intensity. The small shifted windows are approximated by equation (8).

\[ E(u,v) = M \begin{bmatrix} u \\ v \end{bmatrix} \]  

When \( M \) is matrix 2*2 can be calculated by the image derivatives as shown in equation (9).

\[ M = \sum_{x,y} w(x,y) \begin{bmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix} \]  

If window was shifted, the intensity value and the eigen value of \( \lambda_1, \lambda_2 \) are changed. Three cases are considered as follows:

- If both \( \lambda_1, \lambda_2 \) are small, so that the local auto-correlation function is flat, the windowed image region is approximately constant intensity.
- If one eigenvalue is high and the other low, so the local auto-correlation function is ridge shaped, then only local shifts in one direction (along the ridge) cause little change in \( C(x,y) \) and significant change in the orthogonal direction, this indicates an edge.
If both eigenvalues are high, so the local auto-correlation function is sharply peaked, then shifts in any direction will result in a significant increase, this indicates a corner.

V. VASCULAR SYSTEM AND ARM

The conveyor system simulation was designed using a servo motor driving the conveyor belt width 10 cm, 62 cm long, as shown in Fig. 4. The robot arm is a revolute robot type (RRR) and controlled by the microcontroller as shown in Fig. 5 and 6.

VI. EXPERIMENT AND RESULT

The aim of this study was to determine the efficiency of inspection triangle, squares, and circle objects by using The Harris Corner Detection. The software was developed by using the Opencv library and Microsoft Visual C++. The results are collected by testing three different objects. The experimental sequence was shown as follows.

A. Preparation and Improvement of Image

The first process is to import the images from webcam and improve the image as shown in Fig. 7 (a)-(b).

B. Object Edges Detection

The object edges were detected by using canny edge detection algorithm. The results are shown in Fig.7(c).

C. Detect Objects

The Harris Corner Detection was used to detect the objects and the results are shown in Fig. 7 (d)

D. Separate Objects

In this step, the robotic revolute robot arm was used to catch the object for separating. There are two experiments were implemented.

Fig. 4. Conveyor model

Fig. 5. A revolute robot arm

Fig. 6. Microcontroller for the control the robot arm

---

VI. EXPERIMENT AND RESULT

The aim of this study was to determine the efficiency of inspection triangle, squares, and circle objects by using The Harris Corner Detection. The software was developed by using the Opencv library and Microsoft Visual C++. The results are collected by testing three different objects. The experimental sequence was shown as follows.

A. Preparation and Improvement of Image

The first process is to import the images from webcam and improve the image as shown in Fig. 7 (a)-(b).

B. Object Edges Detection

The object edges were detected by using canny edge detection algorithm. The results are shown in Fig.7(c).

C. Detect Objects

The Harris Corner Detection was used to detect the objects and the results are shown in Fig. 7 (d)

D. Separate Objects

In this step, the robotic revolute robot arm was used to catch the object for separating. There are two experiments were implemented.

Fig. 7. Experiments of three object type (a) original image [RGB] (b) Through improved image [Gray level] (c) Canny Edge Detection (d) Conventional Hough Transform

- Each object type is implemented for robot to detect and separate 100 times. The result is shown in Table I.
- Three object types are mixed for robot to detect and separate. This experiment consists of 100 pieces of each object type. The result is shown in Table II

<table>
<thead>
<tr>
<th>Objects</th>
<th>Number of Objects Counted</th>
<th>Error Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rigid circle</td>
<td>100</td>
<td>99</td>
</tr>
<tr>
<td>Rectangular</td>
<td>100</td>
<td>99</td>
</tr>
<tr>
<td>Triangular</td>
<td>100</td>
<td>95</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Objects Types</th>
<th>Number of Objects Counted</th>
<th>Error Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rigid circle</td>
<td>100</td>
<td>99</td>
</tr>
<tr>
<td>Rectangular</td>
<td>100</td>
<td>95</td>
</tr>
<tr>
<td>Triangular</td>
<td>100</td>
<td>94</td>
</tr>
</tbody>
</table>

VII. CONCLUSION

The conventional Harris Corner Detection can be used for the object identification and shape recognition. The algorithm can be implemented with the robot arm for separate the objects. The experimental results have an error occurs form the light affect and the angle of the camera. However, the error maximum was 6%.

REFERENCES


Kitti Toontham received his B.S.I.Ed. from Rajamungala University of Technology, Thailand, in 1988. He received her M.S.Ed from the King Mongkut's University of Technology North Bangkok (KMUTNB), Thailand, in 2010. He joined RMUTI as a lecture of Teacher Training in Computer Engineering Department in 1990. Now he is an assistant professor in the department.

Jaruwon Toontham received her B.S.I.Ed. from Rajamungala University of Technology, Thailand, in 1994. She received her M.S.Ed from the King Mongkut's University of Technology North Bangkok (KMUTNB), Thailand, in 1999. Now, she is a Ph.D student in King Mongkut's University of Technology North Bangkok (KMUTNB), Thailand.

Chaiyapon Thongchaisaratkrul received his B.S.I.Ed. and M.Eng from the King Mongkut's University of Technology North Bangkok (KMUTNB), Thailand, in 1989 and 1992 respectively. He received his Ph.D. from the Vanderbilt University, USA in 2001. He joined KMUTNB as a lecture of Teacher Training in Electrical Engineering Department in 1989. Now he is an assistant professor in the department. He served as the vice president for network and conference of the Thai Robotics Society in 2008. His research interests include intelligent control systems, robotics, energy conservation, and energy resources.