Robust Recognition of Targets for Underwater Docking of Autonomous Underwater Vehicle

M. F. Yahya
Underwater, Control and Robotics Group,
School of Electrical and Electronic Engineering,
Universiti Sains Malaysia, Engineering Campus,
14300, Nibong Tebal, Pulau Pinang, Malaysia.
mohdfaid85@gmail.com

M. R. Arshad
Underwater, Control and Robotics Group,
School of Electrical and Electronic Engineering,
Universiti Sains Malaysia, Engineering Campus,
14300, Nibong Tebal, Pulau Pinang, Malaysia.
eerizal@usm.my

Abstract—Underwater docking for an autonomous underwater vehicle is important in sense that the vehicle can stop at a docking station to recharge its battery, transfer data, and can be used for launch and recovery system. To perform docking, recognizing the station through vision is important. There are few researches conducted on underwater docking using vision to recognize targets as guidance for the underwater vehicle to home towards the station. In those researches, docking is unsuccessful when one or more of the targets are not detectable. Specifically, the image processing part failed to recognize the target if the number of target taken from a captured image is not the same as the number of target in a desired image. This paper proposes a robust recognition of targets algorithm using bounding box partitioning to overcome the aforementioned problem. Result shows that the algorithm is capable to recognize the targets even if some of the targets went missing.

Keywords—robust recognition, underwater docking, autonomous underwater vehicle

I. INTRODUCTION

Some of the more advanced applications of autonomous underwater vehicle (AUV) include exploring risky places like beneath ice sheets in polar regions, performing geo-hazard assessment associated with oil and gas infrastructure, and embarking on a large-scale survey in locating the position of airplane wrecks [1]. To perform some of these applications, an AUV has to be powered up by a battery supplying electrical energy to its thruster, on-board computer, camera, etc. Although battery is convenient as it offers portability to an AUV, it can only be used for a limited time. To overcome this problem, a floor standing docking station could be used for an AUV to recharge its battery back to its full power. Apart from recharging, the station would be an ideal place for the AUV to frequently store new data, can be serve as a waypoint, and for launch and recovery when the AUV needs some maintenance. In order for the AUV to dock properly, an image sensor could be used as it is suitable for close range operation which requires high degree of precision.

There are several studies regarding underwater docking using vision. In one study, a cruising AUV is optically guided by lights mounted around the entrance of a docking station [2]. 5 lights were used and a sequence of image processing steps is implemented to differentiate between the light considered as bright region and the surrounding considered as dark region. Some of the challenges faced in the study were light scattering, formation of noise during thresholding operation, and reflection of light from the surface up top. In other study, a hovering AUV docks inside a rectangular seafloor station by using 4 light-emitting diodes (LEDs) [3]. The LEDs are placed inside the docking station instead of around the entrance. This enables the AUV to land properly inside the docking station. In other interesting work, binocular vision is used to locate 4 green LEDs installed on a docking station [4]. Features of the LEDs displayed in an image are extracted and target matching technique using support vector machine (SVM) is adopted to confirm that the target LEDs are valid. Fig. 1 shows the images of light system originating from docking stations that is being captured in different studies.

(a) 5 light system [2]  (b) 4 LEDs [3]  (c) 4 light system [4]

Fig. 1 Captured docking station images

There is also a research conducted on using just a single LED for docking purposes as in [5]. However, docking using a single light source can become a complex problem because it lacks 3-dimensional (3D) visual information. For underwater 3D environment, at least 3 light sources or targets are needed. In previous research, docking is unsuccessful when one or more of the targets are missing as stated in [2]. Specifically, the image processing part failed to recognize the targets if the number of targets taken from an image is not the same as the number of targets in a desired image. Therefore, this paper proposes a robust recognition of targets algorithm to overcome the aforementioned problem. In this study, artificial object which serves as target is used instead of light sources for initial lab test.

This paper is organized as follows: Section II describes the process of extracting features from a captured image. Section III explains about the proposed robust identification of the targets placed on the docking station. Section IV is about
system design and experimental setup. Section V conveys the result and discussion. Finally, Section V concludes this paper with some future works addressed accordingly.

II. FEATURE EXTRACTION OF TARGETS

The feature that needs to be extracted from the image is the color of the artificial targets. The shape of the artificial target is that of a spherical shape and its color is chosen to be pink which from the naked eye point of view, is distinguishable from the surrounding. To extract the feature, a series of image processing steps is proposed. The series are as follow: image acquisition, color model conversion, thresholding, morphology opening, and calculating image moment. Fig. 2 shows a block diagram of the proposed method.

![Fig. 2 Proposed feature extraction method](image)

Firstly, an image is acquired using an image capture device. The image capture device used in this study is Logitech C920 as it is affordable and can display good quality image. It is equipped with charge-coupled devices (CCD) image sensor to capture light and convert it into electrical signal. The size of captured image is set to 680 pixels wide times 480 pixels high. This size allows the device to display smooth image sequence of about 30 frames per second. Also, this size is selected because bigger size causes frame drops and smaller size reduces details from an image. Table I shows the camera property values for image capture.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
<th>Selected</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brightness</td>
<td>128</td>
<td>0</td>
<td>255</td>
<td></td>
</tr>
<tr>
<td>Contrast</td>
<td>128</td>
<td>0</td>
<td>255</td>
<td></td>
</tr>
<tr>
<td>Exposure</td>
<td>-5</td>
<td>-7</td>
<td>-1</td>
<td></td>
</tr>
<tr>
<td>Focus</td>
<td>0</td>
<td>0</td>
<td>250</td>
<td></td>
</tr>
<tr>
<td>Frame rate</td>
<td>30.0000</td>
<td>5.0000</td>
<td>30.0000</td>
<td></td>
</tr>
<tr>
<td>Gain</td>
<td>100</td>
<td>0</td>
<td>255</td>
<td></td>
</tr>
<tr>
<td>Saturation</td>
<td>128</td>
<td>0</td>
<td>255</td>
<td></td>
</tr>
<tr>
<td>Sharpness</td>
<td>128</td>
<td>0</td>
<td>255</td>
<td></td>
</tr>
<tr>
<td>White Balance</td>
<td>6500</td>
<td>2000</td>
<td>6500</td>
<td></td>
</tr>
<tr>
<td>Zoom</td>
<td>100</td>
<td>100</td>
<td>500</td>
<td></td>
</tr>
</tbody>
</table>

The image captured from the camera is of Red-Green-Blue (RGB) color model. This color model is the standard for visual display in computer and television. Basically, the image is formed from the additive of three primary colors of red, green, and blue. The additive property makes color separation difficult. Alternatively, Hue-Saturation-Value (HSV) color model is one of the preferable ways to extract specific color from an image. HSV color model can be obtained from RGB color model by firstly normalizing the RGB values to only 0 to 1 by

\[ R' = R / 255 \]
\[ G' = G / 255 \]
\[ B' = B / 255 \]

then use the following formula

\[ V = \max(R', G', B') \]
\[ S = \begin{cases} 
V - \min(R', G', B') & \text{if } V \neq 0 \\
0 & \text{otherwise} 
\end{cases} \]
\[ H = \begin{cases} 
60(G' - B') / V & \text{if } V = R' \\
60(B' - R') / V & \text{if } V = G' \\
60(R' - G') / V & \text{if } V = B' 
\end{cases} \]

Once the image had been converted to a HSV color model, the target can now be properly separated from the background using thresholding operation. This thresholding operation will convert the target into white region while the background into black region. In order to separate the target from the background, the thresholded pixel value is chosen as

\[ P_T = \begin{cases} 
0.795 < H < 0.889 \text{ and } 0.451 < S < 0.732 \text{ and } 0.991 < V < 1.000 & \text{if } 1 \\
0 & \text{otherwise} 
\end{cases} \]

The threshold values are obtained based from the average values by thresholding 10 image samples. Fig. 3 shows the image color property and histograms for thresholded value selection of HSV color model.

![Fig. 3 HSV color property and histograms](image)
Due to the nature of thresholding operation, the produced image may contain some noises and the target feature may have deformed quite a bit. In order to remove these noises and reconstruct the image to have a proper shape back, morphology opening is adopted. Morphology opening is the dilation of erosion of an image \( I \) by a structuring element \( S \):

\[
I \circ S = (I \otimes S) \oplus S
\]  

(8)

where \( \otimes \) and \( \oplus \) denote erosion and dilation respectively. The structuring element used is of disk shape. Radius of the disk is chosen to be 3 and line structuring element used to approximate the disk shape is chosen as 4. These values are chosen based from observation of 10 sample images that during erosion operation, they do not completely eliminate the target. Fig. 4 shows the structuring element.

![Fig. 4 Structuring element of disk shape](image)

After the image had undergone the morphology operation, information about all of the target features in the image can be extracted for further processing. The feature information in term of pixel’s intensity is also known as image moment. In order to find this image moment, the image is scanned from top left to top right and moving downwards row by row to accumulate the grouping of white pixels. Their moment can be evaluated using the following formula

\[
m_{j} = \sum_{x,y}\text{array}(x,y) \cdot x^{j} \cdot y^{j}
\]

(9)

where \( m_{00} \) represents area, and its centroid can be found from

\[
x_{m} = \frac{m_{10}}{m_{00}}
\]

(10)

\[
y_{m} = \frac{m_{01}}{m_{00}}
\]

(11)

The image moment formula is derived from Green’s formula. Fig. 5 shows the feature extraction process of a single image containing the artificial objects.

III. ROBUST CLASSIFICATION OF ARTIFICIAL TARGET

The algorithm proposed in this paper is on identifying the configuration of the artificial targets correctly even if some of those targets went missing. There are 4 major steps in the proposed algorithm: determining number of detected features, verifying the area of the features, measuring the closeness of the features, and classifying the targets configuration. Each of the steps is explained as follow.

A. Determine the number of features

The maximum number of targets is set to 5 and the minimum is set to 3. If the number of detected features is greater than 5, this signifies too much noise while if the number of features less than 3, this represents the number of detected features are too low. In either case, thresholding values need to be re-adjusted so that the number of detected features is within 3 and 5.

B. Verifying the area of the features

Although the number of extracted features is acceptable this does not mean that each of the features correctly represent the target on the docking station. So, the areas of the features need to be verified. Initially, the average value of the area of all features can be calculated from

\[
\overline{m}_{ao} = \frac{\sum_{j=1}^{n} m_{aoj}}{n}
\]

(12)

where \( n \) is the total number of features. The condition imposed that all area of the features must not vary so much from the calculated average value. All areas will be divided by \( \overline{m}_{ao} \) and must be within 0.7 to 1.3. If it is lower than 0.7, then it might be some random noise and not the target. If it is greater than 1.3, then it might just be another large object which size is greater than the average size of the features.
C. Measuring the closeness of the features

The distance between each feature had to be measured to signify that all of the targets are close to one another. The Euclidean distance between each centroid of the features can be calculated from

$$d_k = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

(13)

Then, the minimum distance between any two features is found from

$$d_{min} = \min(d_k)$$

(14)

The rule is imposed that the maximum distance between any two features must be within 3 times of the minimum distance of any two features. This should confirm that each feature is close to each other which represent a proper group of targets placed on the docking station.

D. Classifying the target configurations by partitioning of bounding box

The configuration of the targets is important in sense that it will serve as the desired image used for a visual servo system in docking operation. The underlying concept of this algorithm is based from classifying the centroid of detected features within a constructed bounding box. The bounding box has a dimension of size $(x_{max} - x_{min}) \times (y_{max} - y_{min})$ and located at point $(x_{min}, y_{min})$. The bounding box is partitioned into 9 regions or 3 by 3. Each horizontal partition length is $(x_{max} - x_{min})/3$ and each vertical partition length is $(y_{max} - y_{min})/3$. All possible target configurations are as listed in Table II. In the table, the black dot represents target placed on the docking station.

<table>
<thead>
<tr>
<th>Number of Targets</th>
<th>Target Configurations</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td><img src="image1" alt="Target Configurations" /></td>
</tr>
<tr>
<td>4</td>
<td><img src="image2" alt="Target Configurations" /></td>
</tr>
</tbody>
</table>

TABLE II. VALID CONFIGURATION OF TARGETS

IV. SYSTEM DESIGN AND EXPERIMENTAL SETUP

Basically, the design of the AUV and the docking station are based from [6]. Fig. 5 shows the frontal part of the AUV and the entrance part of the underwater docking station. The AUV is expected to dock properly inside the docking station since the AUV has smaller size when compared to the docking station entrance size. Also shown in the figure is the placement of the artificial targets on the docking station.

Fig. 5. Frontal views of designed system [6]

The developed algorithm is first tested by just using a camera which represents an AUV and artificial objects which represents the artificial targets on the underwater docking station. The system setup is as shown in Fig. 6. The camera is placed at 5 meters away from the artificial targets. Then, the camera starts recording and surges forward for 4 meters and stop. This process is repeated for 10 times to get 10 sample videos. These sample videos are acquired to test for the proposed developed algorithm.
For each sample video, only a certain range of frames are acquired. The frames that are cut out refer to the frames when the camera is not moving by the operator. Table III shows the range of frames used for experimentation.

<table>
<thead>
<tr>
<th>Videos</th>
<th>Frames</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Starts at</td>
<td>Ends at</td>
</tr>
<tr>
<td>1</td>
<td>567</td>
<td>1757</td>
</tr>
<tr>
<td>2</td>
<td>450</td>
<td>1038</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>875</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>732</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>627</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>583</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>556</td>
</tr>
<tr>
<td>8</td>
<td>280</td>
<td>916</td>
</tr>
<tr>
<td>9</td>
<td>399</td>
<td>954</td>
</tr>
<tr>
<td>10</td>
<td>336</td>
<td>907</td>
</tr>
</tbody>
</table>

V. RESULT AND DISCUSSION

There are four criteria to measure the robustness of the developed algorithm. These four criteria are in term of invalid number of features, invalid area of features, invalid distance between features, and correct configuration of the features.

To describe further, invalid number of features represent the occurrence of the total number of features detected in the video when less than 3 or more than 5. Then, invalid area of features denotes the total number of feature when the feature’s area (divided by average of all area) is out from the range of a specified limit. The specified limit is between 0.7 to 1.3. Next, invalid distances between features signify the total number of events when distance between features might go beyond 3 times of the minimum distance between any two features. Finally, the correct configuration of the features must confirm to one of the configurations listed in Table II. Otherwise, the configuration is not valid.

For this experiment, the proposed algorithm is tested for when the number of artificial targets is at 5, 4, and 3.

A. Test for 5 targets

Fig. 7 shows an image with 5 artificial targets. Additionally, Fig. 8 displays the bounding box partitioning of the targets from Fig. 7. The image is smaller in size since it has been cropped for better display.

Fig. 7 One of the sample image taken from a video shows 5 pink artificial objects

Fig. 8 Bounding box partitioning for 5 targets

The result obtained when testing the algorithm by using all of the sample videos is listed in Table IV. From the table, the developed algorithm is capable to identify targets from each frame in the videos correctly.

<table>
<thead>
<tr>
<th>Videos (Number of Frames)</th>
<th>Invalid number of targets</th>
<th>Invalid target area</th>
<th>Invalid distance between targets</th>
<th>Correct configuration of targets</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (1191)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1191</td>
</tr>
<tr>
<td>2 (589)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>589</td>
</tr>
<tr>
<td>3 (875)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>875</td>
</tr>
<tr>
<td>4 (732)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>732</td>
</tr>
<tr>
<td>5 (627)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>627</td>
</tr>
<tr>
<td>6 (583)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>583</td>
</tr>
<tr>
<td>7 (556)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>556</td>
</tr>
<tr>
<td>8 (637)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>637</td>
</tr>
<tr>
<td>9 (556)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>556</td>
</tr>
<tr>
<td>10 (572)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>572</td>
</tr>
</tbody>
</table>

B. Test for 4 targets

In order to make 4 artificial targets, an occlusion is introduced to filter out one of the 5 targets. The occlusion is made to be randomly occurred to one of the targets when testing is conducted. Fig. 9 shows an image with one of the 5 targets being occluded by a black imaginary block.
Fig. 9 Only 4 targets seen and another 1 is blocked by an occlusion

Fig. 10 shows the entire correct configuration of the targets using bounding box partitioning when there are only 4 targets remaining. Furthermore, the result obtained for the test for 4 artificial targets is the same as listed in Table IV.

C. Test for 3 targets

Similar to the way imaginary occlusion was introduced to test for 4 artificial targets, in the case for 3 artificial targets, there are 2 occlusions drawn. Fig. 11 shows an image with 2 imaginary occlusions blocking the artificial targets.

Fig. 11 Only 3 artificial targets seen and another 2 is blocked by imaginary occlusions

Fig. 12 shows the entire correct configuration for 3 artificial targets. Additionally, the result obtained for the test for 3 artificial targets is the same as listed in Table IV. This shows that the developed algorithm is robust even when there are only 3 targets out of total 5 placed on the board.

VI. CONCLUSION AND FUTURE WORKS

In conclusion, an algorithm to robustly recognized targets placed on a docking station model had been proposed. The proposed algorithm has 2 major sections. First major section is on extracting features of the targets in an image. The section contains methods of image acquisition, color model conversion, thresholding, morphology opening, and calculating image moments. Second major section is on determining the validity of the targets. This section contains the methods to validate the number of detected targets, verifying the area of the targets, measuring the closeness of the targets, and classifying the configuration of the targets by bounding box partitioning. Result shows that the algorithm is capable to recognize the targets even if some of the targets are missing.

As for future works, the developed algorithm is going to be implemented in a real AUV system to perform underwater docking. With this, further robust test can be conducted in a pool environment.
ACKNOWLEDGMENT

Author 1 would like to express gratitude to Universiti Teknikal Malaysia Melaka (UTeM) and Kementerian Pendidikan Malaysia (KPM) for providing scholarship.

REFERENCES


