

3D Object Detection Based on Point Cloud Data

Dewi Mutiara Sari¹, Alfian Rizaldy Pratama², Dadet Pramadihanto³, Bayu Sandi Marta⁴

^{1,2,3,4}*Informatics and Computer Engineering Department, Politeknik Elektronika Negeri Surabaya, Indonesia*

¹dewi.mutiar@pens.ac.id (*)

²alfanrizaldy@ce.student.pens.ac.id

^{3,4}[dadet, bayu]@pens.ac.id

Received: 2021-12-26; Accepted: 2022-01-21; Published: 2022-01-24

Abstract— In the Industrial robotic, computer vision is an important part of the system. The popular object used in the industrial field is a 3D pipe. The problem that is currently being developed is how to detect an object. This research aims to estimate the object detection that is, in this case, is a 3D pipe in various lighting conditions. The camera used in this research is Time of Flight. The methods applied are Remove NaN data for Pre-processing, Random Sample Consensus (RANSAC) for Segmentation, Euclidean Distance for Clustering, and Viewpoint Feature Histogram (VFH) for the object detection. A study conducted on five different objects found that the system could detect each one with a success rate of 100% for the first object, 98.05 percent for the second object, 93.97 percent for the third object, 94 percent for the fourth object, and 99.48 percent for the fifth object. Overall, the system's accuracy in detecting the object is 97.1 percent when four different lighting conditions are applied to five different objects in total.

Keywords— Object Detection, Point Cloud, Industrial Robotic, Computer Vision.

I. INTRODUCTION

Computer vision technology has grown so fast in industrial revolution 4.0 [1]. The benefits of applying computer vision technology are reducing the cost and increasing productivity because of the high computation [2]. An example of applied technology in computer vision is the bin picking system for picking an object from the initial place and placing it to the destination.

Object detection aims to make the robot pick the object from the beginning place to the goal place. So the accuracy of object detection has a big role in the robot's success in moving the object [3]. The development of hybrid approaches between Chamfer Matching and Partial Directed Hausdorff (PDH) have been developed in object detection. The popular object that is used for the bin picking system is the pipe [4]. The common pipes that are used in Indonesian industry are T pipe, L pipe, straight pipe, drat pipe, and cover pipe [5].

The problem of estimating object detection is accuracy. One factor influencing object detection accuracy is the light intensity in the environment, which reflects the object [6]. In addition, the correct applied method also has a role in determining the accuracy of object detection.

One of the methods applied for object detection is the Viewpoint Feature Histogram. By using several cube-shaped objects and non-cylindrical bottles as the object experiments, the results of object detection accuracy obtained 94.4% [13].

There are some types of data input used for object detection. One of them is point cloud data which gives a robust object detection [7]. A camera device that can produce the point cloud data is a Time of Flight (ToF) camera. The working of the ToF camera is shooting the light source to the targeted scene/object

and reflecting back to the receiver. The ToF camera has a performance that does not affect the lighting in the environment around. So even when there is no light in the environment, the Time of Flight camera can still work to get object data [8]

II. RESEARCH METHODOLOGY

Figure 1 explains the pipeline of the research. The method is broken into two lines: dataset as the reference and testing data as the data test.

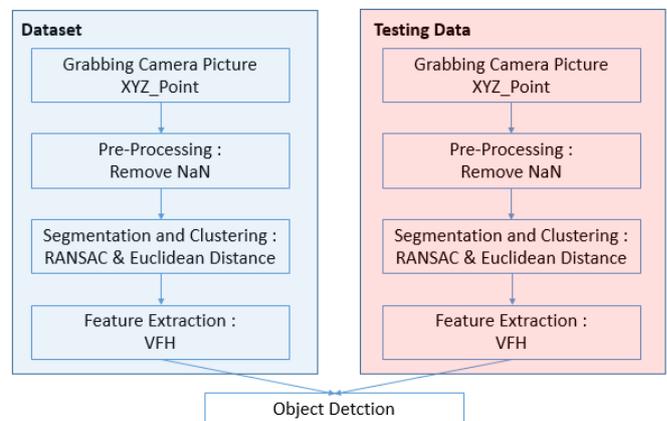


Figure 1. The Pipeline of The Research

Stages for 3d object detection based on point cloud data, among others, are:

Step 1: Grabbing a picture from a camera to get the point cloud data.

Step 2: Pre-processing data by removing point cloud data using the Remove NaN algorithm.

Step 3: The segmentation and clustering process uses the RANSAC algorithm and Euclidean Distance

Step 4: Feature Extraction uses the Viewpoint Feature Histogram (VFH) algorithm, which detects objects originating from feature extraction data from Dataset and testing data.

A. Pre-Processing

In this research, the camera used Time of Flight camera CamBoard pico flexx for taking the data. Some configuration is needed so the camera can be used, such as the installation of SDK for Linux OS and Robot Operating System (ROS) framework.

The camera produced a point cloud in the x,y, and z-axis. In the so-called point PointXYZ, not all the point cloud data have defined values. Some of the point cloud data are stored with Not a Number or NaN value. The point cloud data with NaN value disrupt to be processed. So it is important to eliminate the points with the NaN value. Remove NaN algorithm to remove the point cloud data with NaN value. The *Remove NaN* algorithm is described in Equation (1) and (2),

$$P_{i.(x|y|z)} \in S \quad (1)$$

$$P_{i.(x|y|z)} = \begin{cases} P_{i.(x|y|z)} \neq NaN, & P_{i.(x|y|z)} \\ NaN, & null \end{cases} \quad (2)$$

Where, $P_{i.(x|y|z)}$ Is point data in scene S . Meanwhile, NaN is not a number, and $null$ is empty.

B. Segmentation and Clustering

After the NaN data are eliminated, it remains the object and background data. Segmentation separates background and object target because the background is not needed for future calculation.

The segmentation algorithm used plane segmentation based on Random Sample Consensus (RANSAC) method. RANSAC is used to get the 3D key point contained corner point and spin image [9].

RANSAC is a method by using the resampling technique to produce a point candidate as the iterative model. The workflow of the RANSAC algorithm for segmentation is described in Figure 2.

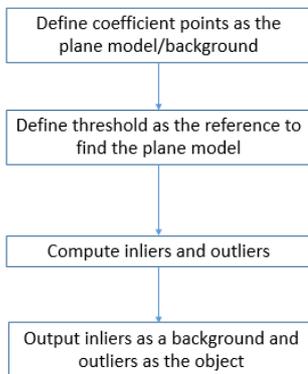


Figure 2. RANSAC workflow

Start by choosing 3 random points to make a plane, with the vertical axis being the normal axis. Then continued calculating the plane equation formulated in Equation (3).

$$ix + jy + kz + l = 0 \quad (3)$$

The $i, j,$ and k variable are the constant, l variable is the normal vector, and $x, y,$ and z are the random points. The value of each constant can be obtained by Equation (4), (5), and (6).

$$[(y_3 - y_1)(x_2 - x_1) - (x_3 - x_2)(y_2 - y_1)] = k \quad (4)$$

$$[(x_3 - x_1)(z_2 - z_1) - (z_3 - z_2)(x_2 - x_1)] = j \quad (5)$$

$$[(z_3 - z_1)(y_2 - y_1) - (y_3 - y_2)(z_2 - z_1)] = i \quad (6)$$

Where, (x_3, y_3, z_3) variable is random point-3, (x_2, y_2, z_2) variable is random point-2, and (x_1, y_1, z_1) is random point-1. Meanwhile, the value of l in Equation (3) can be obtained using Equation (7),

$$-(ix_1, jy_1, kz_1) = l \quad (7)$$

The next computation determines the distance between all random points concerning the plane. Equation (8) is a formulation to test whether the new point belongs to plane or not,

$$Dist = (ix_4 + jy_4 + kz_4 + l) * (i^2 + j^2 + k^2)^{-1/2} \quad (8)$$

$Dist$ is the distance, and (x_4, y_4, z_4) is the new point. A threshold must be measured to determine whether or not the new point belongs to the plane.

The next process is clustering. The input is the obtained object data. The clustering process applied the Euclidean Cluster Extraction method. Euclidean Cluster Method's main task is to define a measurable character and make a group of points by using the closest distance of two points obtained from Euclidean Distance. Equation (9) describes the distance between two points a and b in the n -points,

$$D = (\sum_{i=1}^n a_i - b_i)^{-1/2} \quad (9)$$

Where i variable is the i -th point and D variable is the distance between point a and points b .

After the *Euclidean Distance* between two points is acquired, then the point will be grouped into the cluster. The clusterization is described in Equation (10).

$$P_{i.(x|y|z)} \begin{cases} P_{i.(x|y|z)} = & \begin{matrix} C_size_{max} > CO_size \\ CO_size > C_size_{min} \\ P_{i.(x|y|z)} < C_Threshold \end{matrix} , P_{i.(x|y|z)} \\ & otherwise, null \end{cases} \quad (10)$$

In which $P_{i.(x|y|z)}$ is a scene point, a clustered object variable (CO variable). The point belongs to clustered object if it has

points between $C_{size_{min}}$ and $C_{size_{max}}$. In addition, the point also should satisfy the minimal threshold of $C_{Threshold}$. If all the conditions are satisfied, then the point belongs to the scene.

C. Object Detection

After the object and background points are well separated by using RANSAC, the objects are clustered using Euclidean Distance. The next step is extracting the object feature such that the object can be detected. In this research, the used feature is *Viewpoint Feature Histogram* (VFH) [10][11][12]. VFH is one of the global descriptors defining the point cloud data in a histogram.

VFH consists of viewpoint direction component and extended Fast Point Feature Histogram (FPFH). The viewpoint feature is formulated in Equation (11).

$$\gamma = \arccos(n_v \cdot \frac{p_v - p_c}{\|p_v - p_c\|}) \quad (11)$$

Where p_c is the center of gravity of i -th point p_i , p_v is viewpoint position, and n_v is the normal vector.

The extended FPFH consists of 3 values: pan, tilt, and yaw. Every value of extended FPFH has 45 bins in the histogram. Then 45 bins for the distance between p_i and p_c and 128 bins for viewpoint feature. So the total bins for each object are 308 bins. The illustration is figured in Figure 3 [13].

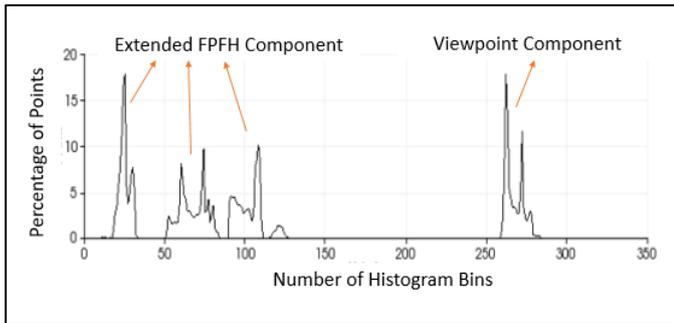


Figure 3. Example of Viewpoint Feature Histogram

With p_i is i -th point, n_i is i -th normal point, and then i -th pan (α_i), tilt (β_i), and yaw (ω_i) can be obtained from Equation (12), (13), and (14)

$$\cos(\alpha_i) = \frac{p_i - p_c}{\|p_i - p_c\|} \cdot n_v \cdot n_i \quad (12)$$

$$\cos \beta_i = n_i \cdot \left(\frac{p_i - p_c}{\|p_i - p_c\|} \right) \quad (13)$$

$$\omega_i = \tan^{-1} \left(\frac{w_i \cdot n_i}{u_i \cdot n_i} \right) \quad (14)$$

Where, w_i variable is a derivation from the local coordinate of a point p_c . The formulation is depicted in Equation (15)(16)(17).

$$w_i = v_i \cdot u_i \quad (15)$$

$$v_i = n_v \cdot \left(\frac{p_i - p_c}{\|p_i - p_c\|} \right) \quad (16)$$

$$u_i = n_v \quad (17)$$

The object detection process obtains the required input from the clustering output. The output from the clustering step is converted into a histogram by using VFH. The VFH of the data set and the VFH of object testing are compared. Chi-Squared Distance is used to determine the histogram similarity between the object dataset and the test object. Then by applying a threshold, the system can detect the object by distinguishing whether the testing object is the same as the object in the data set or not. The formulation of *Chi-Squared Distance* is explained in Equation (18).

$$\frac{1}{2} \sum_{i=1}^n (x_i - y_i)^2 (x_i + y_i)^{-1} \quad (18)$$

Where n variable is the bins value in the histogram, then x_i and y_i the i -th value of histogram for x -axis and y -axis.

III. RESULT AND DISCUSSION

This section will be divided into experimental setup, Segmentation and Clustering, and Object Detection.

A. Experimental Setup

The specification for the experimental setup of this research regarding hardware and software is summarized in Table I.

Scope	Description	Specification
Hardware	Processor	Intel(R) Core i3-2370M CPU @2.40GHz x 4
	Storage	93 GB
	RAM	8 GB
	VGA	NVIDIA Geforce 610M 2GB
Software	OS	Ubuntu 18.04 LTS 64-bit
	Software	ROS Melodic Morenia
	Library	<ul style="list-style-type: none"> Chambord pico flex SDK for Linux 3.23.0.86 Point Cloud Library 1.9.1

Meanwhile for environmental setup is illustrated in Figure 4.



Figure 4. Environmental setup

Based on Figure 4, the camera used in this research is Time of Flight (ToF). The distance between the camera and the object plane is 45 cm, and the camera position is fixed. The objects used in this research are shown in Figure 5.

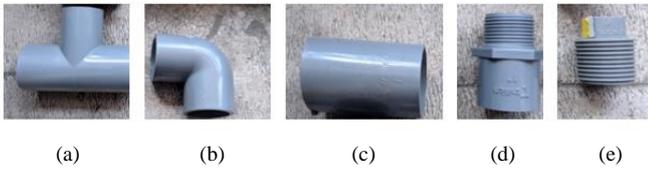


Figure 5. Pipe Object : (a) Object 1 (b) Object 2 (c) Object 3 (d) Object 4 (e) Object 5

Figure 5 shows the 3D object used in the research. The objects are pipes which are commonly used in the industrial field. Then for the lighting used are environmental lighting, 120 Lumens, 400 Lumens, and 800 Lumens.

For the object detection test, the system captured the object for 2 minutes for each test. Meanwhile, the number of test sets is 50 experiments for 5 objects (10 experiments for each object) with the number of frames explained in The Object Performance Table. Meanwhile, the accuracy is obtained by comparing the test result for the real condition (as the ground truth). Then converted into a percentage to express the accuracy.

B. Segmentation and Clustering

The sampling data acquisition or raw data captured by ToF and the segmentation result are shown in Table II.

TABLE II
SEGMENTATION RESULT

Raw Data	Segmentation

Raw Data	Segmentation

Based on Table II, the data in the second column of Table II gives the information about point cloud, representing the position of points in the x, y , and z -axis. The camera doesn't provide information regarding the RGB color.

For the data to be segmented between the object's points and the unnecessary points, the system conducted NaN removal priorly then continued with segmentation. The NaN data emerged from the raw data captured by the ToF camera. This data includes built-in noise data from ToF camera captures. So the NaN data needs to be removed to proceed to the segmentation process.

TABLE III
NaN POINTS DATA

i -th Sampling Experiment	Total Input Points	Total NaN Data points	% NaN
1	38304	1009	2,63
2	38304	1001	2,61
3	38304	905	2,36
4	38304	1019	2,66
5	38304	979	2,55
6	38304	997	2,60
7	38304	1007	2,63
8	38304	1028	2,68
9	38304	997	2,60
10	38304	996	2,60

Table III describes that the percentage of NaN data that can not be processed in the next step should be removed. For the programming output of NaN, removal data is shown in Figure 6. As described in Figure 6, the left part is original data which still contains the NaN value. After applying the NaN removal algorithm in Equation (1) and (2), the NaN values are removed, as shown on the right side of Figure 6. After the NaN value is removed, the system applies the segmentation algorithm by removing the plane. The result is shown in the Segmentation column of Table II. Qualitatively, the system successfully removes the plane. The system remains the object's point which is assigned with green color.

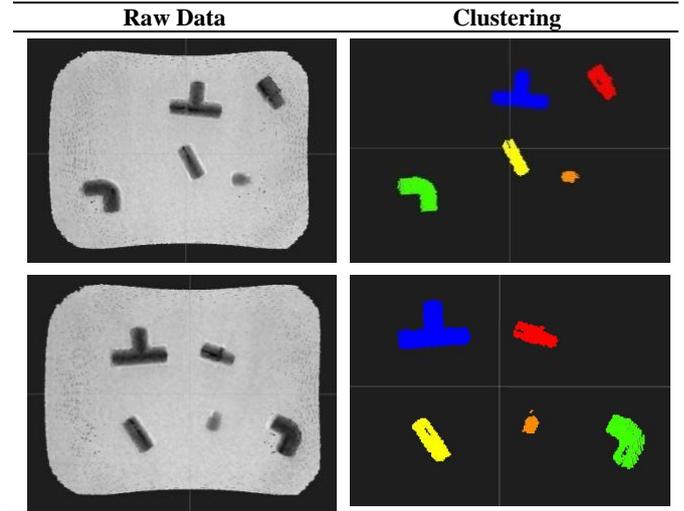
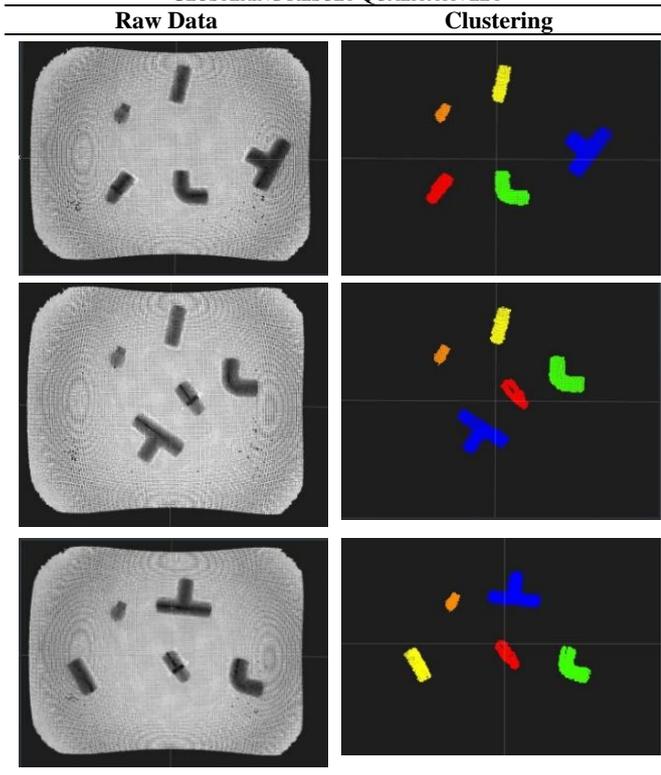
```

1 # .PCD v0.7 - Point Cloud Data file format
2 VERSION 0.7
3 FIELDS x y z
4 SIZE 4 4 4
5 TYPE F F F
6 COUNT 1 1 1
7 WIDTH 224
8 HEIGHT 171
9 VIEWPOINT 0 0 0 1 0 0 0
10 POINTS 30304
11 DATA ascii
12 nan nan nan
13 nan nan nan
14 nan nan nan
15 nan nan nan
16 nan nan nan
17 nan nan nan
18 nan nan nan
19 nan nan nan
20 nan nan nan
21 nan nan nan
22 nan nan nan
23 nan nan nan
24 nan nan nan
25 nan nan nan
26 nan nan nan
27 nan nan nan
28 nan nan nan
29 nan nan nan
30 nan nan nan
31 nan nan nan
32 nan nan nan
33 nan nan nan
34 nan nan nan
35 nan nan nan
36 nan nan nan
37 nan nan nan
38 nan nan nan
39 -0.20717666 -0.19839217 0.45623401
1 # .PCD v0.7 - Point Cloud Data file format
2 VERSION 0.7
3 FIELDS x y z
4 SIZE 4 4 4
5 TYPE F F F
6 COUNT 1 1 1
7 WIDTH 37300
8 HEIGHT 1
9 VIEWPOINT 0 0 0 1 0 0 0
10 POINTS 37300
11 DATA ascii
12 -0.20717666 -0.19839217 0.45623401
13 -0.19616874 -0.19868162 0.45337883
14 -0.1934008 -0.19816323 0.45160595
15 -0.19136782 -0.19839458 0.45150184
16 -0.19041114 -0.19976059 0.4541761
17 -0.18738864 -0.19896621 0.45189759
18 -0.18581359 -0.19970803 0.45314965
19 -0.18251838 -0.19859654 0.45023888
20 -0.18157558 -0.20064894 0.45318234
21 -0.17871092 -0.19939384 0.45139292
22 -0.17922556 -0.20254099 0.45025022
23 -0.17291515 -0.19795661 0.44765949
24 -0.17397138 -0.20179603 0.45616257
25 -0.1690858 -0.19875382 0.44915166
26 -0.16662069 -0.19851285 0.44851539
27 -0.16775736 -0.20261547 0.45773393
28 -0.16463917 -0.20162185 0.45548066
29 -0.15935528 -0.19791043 0.44712836
30 -0.15808611 -0.1991501 0.45000142
31 -0.15611772 -0.19953226 0.45097631
32 -0.15421924 -0.2000163 0.45222005
33 -0.15183828 -0.19987252 0.45200114
34 -0.1506751 -0.20136657 0.45568347
35 -0.14722347 -0.19979365 0.45237899
36 -0.14487524 -0.19969229 0.45243627
37 -0.1419041 -0.19871539 0.45053028
38 -0.14009537 -0.19936101 0.45234677
    
```

Figure 6. NaN Removal Data: Program Output

The next step is clustering. The goal of this step is the system can distinguish between one object and other objects. With the same sampling case in acquired data in Table II, the result of the clustering step is qualitatively shown in Table IV.

TABLE IV
 CLUSTERING RESULT QUALITATIVELY



Based on Table IV, it can be analyzed qualitatively to distinguish the objects. It is assigned by every type of object (pipe) with different colors. The system has 5 clusters in the experiment of Table IV. The first cluster is the blue color. The second cluster is green, the third cluster is yellow, the fourth cluster is red, and the fifth cluster is orange color.

Quantitatively, the sampling of clustering result summarized in Table V. Table V tells about the quantity points of the clustered object. The quantity also depends on the dimension of the object.

TABLE V
 CLUSTERING RESULT QUANTITATIVELY

<i>i</i> -th experiment	Clustering Data				
	Object 1 (points)	Object 2 (points)	Object 3 (points)	Object 4 (points)	Object 5 (points)
1	789	534	343	325	113
2	832	532	342	329	113
3	768	532	343	326	110
4	834	538	344	326	116
5	798	541	377	325	110
6	799	541	378	328	111
7	789	556	378	328	115
8	779	476	379	326	81
9	784	514	381	328	123
10	787	555	377	325	123

B. Object Detection

Two minutes were allocated for each test to examine object detecting techniques. If the value of object detection is less than the threshold value, then the system assigns a bounding box to the object. The same case of the sampling test in the prior test can be shown in Figure 7 for the first sampling test. Based on Figure 7, the first object detection (T pipe) qualitatively can be detected. The system successfully could detect the T pipe by giving the bounding box to the object target: the T pipe.



(a). Real Scene (b). Test Result
 Figure 7. The First Object Detection Test

Quantitatively, the result can be shown in Table VI with various lighting conditions. Based on Table VI, it can be analyzed that along two minutes, and the system consistently could detect the first object (T Pipe) in all frames and in all various lighting.

TABLE VI
 OBJECT DETECTION OF FIRST OBJECT WITH VARIOUS LIGHTING

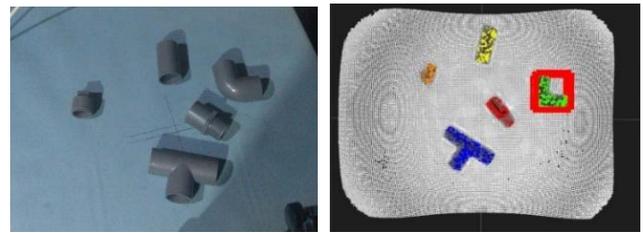
<i>i-th</i> Experiment	The success of Detection (%)			
	Environment Lighting	Lighting 120 Lumens	Lighting 400 Lumens	Lighting 800 Lumens
1	100	100	100	100
2	100	100	100	100
3	100	100	100	100
4	100	100	100	100
5	100	100	100	100
6	100	100	100	100
7	100	100	100	100
8	100	100	100	100
9	100	100	100	100
10	100	100	100	100

This test summarizes the number of frames and Frame per Second (FPS) values in Table VII. Based on Table VI and VII, in all lighting treatments, the system 100% could detect the first object (T pipe) in 10 times experiment.

TABLE VII
 PERFORMANCE OF FIRST OBJECT DETECTION WITH VARIOUS LIGHTING

<i>i-th</i> Experiment	n-Frame / FPS (Environment Ligthing)	n-Frame / FPS (120 Lumens)	n-Frame / FPS (400 Lumens)	n-Frame / FPS (800 Lumens)
1	78 / 0.65	55 / 0.45	57 / 0.55	69 / 0.58
2	104 / 0.86	99 / 0.83	98 / 0.81	107 / 0.89
3	96 / 0.8	97 / 0.81	91 / 0.75	96 / 0.8
4	98 / 0.81	97 / 0.81	96 / 0.8	91 / 0.75
5	94 / 0.78	90 / 0.75	94 / 0.78	94 / 0.78
6	109 / 0.9	108 / 0.9	106 / 0.88	111 / 0.93
7	108 / 0.9	106 / 0.88	108 / 0.9	107 / 0.89
8	108 / 0.9	108 / 0.9	107 / 0.89	112 / 0.93
9	107 / 0.89	109 / 0.9	108 / 0.9	108 / 0.9
10	109 / 0.9	110 / 0.92	107 / 0.89	107 / 0.89

The experiments are carried out over 2 minutes. Figure 8 depicts the second object (L pipe) in the object detection test, which is now in progress. Based on Figure 8, the second object detection (L pipe) qualitatively can be detected. The system successfully could detect the L pipe by giving the bounding box to the object target: the L pipe.



(a). Real Scene (b). Test Result
 Figure 8. The Second Object Detection Test

Quantitatively, the result can be shown in Table VIII with various lighting conditions. Based on Table VIII, it can be concluded that during two minutes, the system's success rate in detecting the second object is: 97.94% for the environment lighting, 98.59% for lighting 120 Lumens, 97.66% for lighting 400 lumens, and 97.99% for lighting 800 Lumens.

TABLE VIII
 OBJECT DETECTION OF SECOND OBJECT WITH VARIOUS LIGHTING

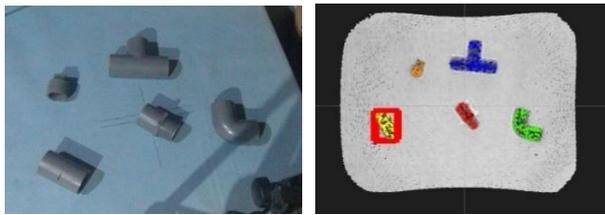
<i>i-th</i> Experiment	The success of Detection (%)			
	Environment Lighting	Lighting 120 Lumens	Lighting 400 Lumens	Lighting 800 Lumens
1	100	100	100	100
2	100	100	100	100
3	100	100	100	100
4	81.25	85.92	77.6	80.9
5	100	100	100	100
6	98,17	100	99	99
7	100	100	100	100
8	100	100	100	100
9	100	100	100	100
10	100	100	100	100

Those results are conducted in 10 experiments for each condition. In this test, the number of frames and the value of Frame per Second (FPS) are summarized in Table IX.

TABLE IX
 PERFORMANCE OF SECOND OBJECT DETECTION WITH VARIOUS LIGHTING

<i>i-th</i> Experiment	n-Frame / FPS (Environment Ligthing)	n-Frame / FPS (120 Lumens)	n-Frame / FPS (400 Lumens)	n-Frame / FPS (800 Lumens)
1	97 / 0.81	101 / 0.84	94 / 0.78	96 / 0.8
2	93 / 0.77	94 / 0.78	94 / 0.78	91 / 0.75
3	129 / 1.08	118 / 0.98	111 / 0.93	132 / 1.1
4	96 / 0.8	71 / 0.6	89 / 0.74	89 / 0.74
5	84 / 0.7	71 / 0.6	84 / 0.7	81 / 0.66
6	109 / 0.9	107 / 0.89	107 / 0.89	108 / 0.9
7	107 / 0.89	107 / 0.89	108 / 0.9	107 / 0.89
8	106 / 0.88	108 / 0.9	107 / 0.89	109 / 0.9
9	107 / 0.89	108 / 0.9	108 / 0.9	107 / 0.89
10	108 / 0.9	108 / 0.9	107 / 0.89	107 / 0.89

Then the third object of the object detection test can be shown in Figure 9. Based on Figure 9, the third object (straight pipe) detection qualitatively can be detected. The system successfully could detect the pipe by giving the bounding box to the third object target. Meanwhile, the third object detection success rate is summarized in Table X with various lighting conditions.



(a). Real Scene (b). Test Result
 Figure 9. The Third Object Detection Test

TABLE X
 OBJECT DETECTION OF THIRD OBJECT WITH VARIOUS LIGHTING

<i>i-th</i> Experiment	The success of Detection (%)			
	Environment Lighting	Lighting 120 Lumens	Lighting 400 Lumens	Lighting 800 Lumens
1	98.9	98.8	100	97.83
2	95.4	92.96	95.6	94.4
3	96.63	92.64	92.4	93.4
4	100	100	100	100
5	87.26	98.5	93.2	90.2
6	96.26	90.74	92.66	89.81
7	89.81	89.81	94.34	88.68
8	95.37	95.37	95.37	94.44
9	89.72	89.72	91.59	90.65
10	94.34	88.68	90.74	92.66

By learning the data in Table X, the system's success rate in detecting the third object are 94.37% for the environment lighting, 93.72% for lighting 120 Lumens, 94.59% for lighting 400 lumens, and 93.21% for lighting 800 Lumens. Those results are conducted in 10 experiments for each lighting condition. In this test, the number of frames and Frame per Second (FPS) values are shown in Table XI.

TABLE XI
 PERFORMANCE OF THIRD OBJECT DETECTION WITH VARIOUS LIGHTING

<i>i-th</i> Experiment	n-Frame / FPS (Environment Ligthing)	n-Frame / FPS (120 Lumens)	n-Frame / FPS (400 Lumens)	n-Frame / FPS (800 Lumens)
1	97 / 0.81	101 / 0.84	94 / 0.78	96 / 0.8
2	93 / 0.77	94 / 0.78	94 / 0.78	91 / 0.75
3	129 / 1.08	118 / 0.98	111 / 0.93	132 / 1.1
4	96 / 0.8	71 / 0.6	89 / 0.74	89 / 0.74
5	84 / 0.7	71 / 0.6	84 / 0.7	81 / 0.66
6	109 / 0.9	107 / 0.89	107 / 0.89	108 / 0.9
7	107 / 0.89	107 / 0.89	108 / 0.9	107 / 0.89
8	106 / 0.88	108 / 0.9	107 / 0.89	109 / 0.9
9	107 / 0.89	108 / 0.9	108 / 0.9	107 / 0.89
10	108 / 0.9	108 / 0.9	107 / 0.89	107 / 0.89

The next is for the fourth object. The result of the object detection test for the fourth object is shown in Figure 10.



(a). Real scene (b). Test result
 Figure 10. The Fourth Object Detection Test

By analyzing Figure 10, the fourth object detection qualitatively can be detected. The system could detect the fourth object and give the red bounding box. For the quantitative analysis with various lighting conditions, the success rate of the fourth object is served in Table XII.

TABLE XII
 OBJECT DETECTION OF FOURTH OBJECT WITH VARIOUS LIGHTING

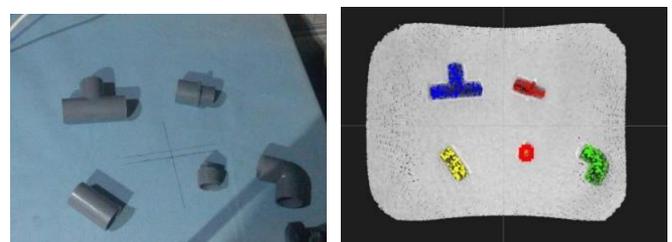
<i>i-th</i> Experiment	The success of Detection (%)			
	Environment Lighting	Lighting 120 Lumens	Lighting 400 Lumens	Lighting 800 Lumens
1	98.8	95.8	97.2	98.63
2	91.4	94.2	85.8	88.8
3	93.75	97.19	94.4	97.06
4	81.5	81.7	82.36	90.48
5	97.2	85.92	90.15	97.02
6	97.22	95.45	99.06	98.17
7	96.26	99.07	96.3	99.26
8	98.13	97.2	99.07	98.15
9	91.51	96.26	92.01	93.46
10	89.62	93.54	95.45	95.33

The experiment is conducted ten times along 2 minutes based on Table XII. The system's success rate in detecting the fourth object is 93.54% for the environment lighting, 93.18% for lighting 120 Lumens, 93.18% for lighting 400 lumens, and 95.64% for lighting 800 Lumens. Then the number of frames and the value of Frame per Second (FPS) are summarized in Table XIII.

TABLE XIII
 PERFORMANCE OF SECOND OBJECT DETECTION WITH VARIOUS LIGHTING

<i>i-th</i> Experiment	n-Frame / FPS (Environment Ligthing)	n-Frame / FPS (120 Lumens)	n-Frame / FPS (400 Lumens)	n-Frame / FPS (800 Lumens)
1	78 / 0.65	70 / 0.58	71 / 0.59	73 / 0.61
2	69 / 0.58	69 / 0.58	70 / 0.58	71 / 0.59
3	64 / 0.53	71 / 0.59	71 / 0.59	68 / 0.56
4	70 / 0.58	71 / 0.59	68 / 0.56	63 / 0.53
5	70 / 0.58	71 / 0.59	71 / 0.59	67 / 0.55
6	108 / 0.9	110 / 0.92	106 / 0.88	109 / 0.9
7	107 / 0.89	108 / 0.9	108 / 0.9	107 / 0.89
8	107 / 0.89	107 / 0.89	107 / 0.89	108 / 0.9
9	106 / 0.88	107 / 0.89	107 / 0.89	107 / 0.89
10	106 / 0.88	106 / 0.88	107 / 0.89	107 / 0.89

The last experiment is for the fifth object : pipe cover. The result of the fifth object detection is illustrated in Figure 11.



(a). Real Scene (b). Test Result
 Figure 11. The Fifth Object Detection Test

REFERENCE

Based on Figure 11, the fifth object detection qualitatively can be detected, and the system gave the bounding box. Meanwhile, Table XIV summarizes the success rate of the fifth object with various lighting conditions.

TABLE XIV
 OBJECT DETECTION OF FOURTH OBJECT WITH VARIOUS LIGHTING

<i>i-th</i> Experiment	The success of Detection (%)			
	Environment Lighting	Lighting 120 Lumens	Lighting 400 Lumens	Lighting 800 Lumens
1	100	97.15	94.55	100
2	100	100	100	100
3	100	100	100	100
4	100	100	100	100
5	95.84	97.3	94.55	100
6	100	100	100	100
7	100	100	100	100
8	100	100	100	100
9	100	100	100	100
10	100	100	100	100

The experiment is conducted ten times along 2 minutes based on Table XIV. The system's success rate in detecting the fifth object is 99.58% for the environment lighting, 99.45% for lighting 120 Lumens, 98.91% for lighting 400 lumens, and 100% for lighting 800 lumens.

IV. CONCLUSION

One of the issues in the computer vision field is object detection. This research aims to estimate 3D object detection in various lighting conditions. Based on the experiment, the system could detect the object with a success rate of 100% for the first object, 98.05% for the second object, 93.97% for the third object, 94% for the fourth object, and 99.48% for the fifth object. All in all, by applying four various lightings in five objects, the total accuracy of the system to detect the object is 97.1%.

ACKNOWLEDGMENT

The research is granted by Politeknik Elektronika Negeri (PENS) Surabaya and Robotic and Intelligent Systems Center (RoISC) Lab. Thanks to them.

This is an open-access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



[1] Hsu, C.-Y., Kang, L.-W., You, T.-Y., & Jhong, W.-C., "Vision-Based Automatic Identification Tracking of Steel Products for Intelligent Steel Manufacturing", *IEEE International Symposium on Multimedia (ISM)*, 2017.

[2] Adamo, F., Attivissimo, F., Di Nisio, A., & Savino, M., "An online defects inspection system for satin glass based on machine vision", *IEEE Instrumentation and Measurement Technology Conference*, 2009.

[3] Kim, J., Lee, M., Yoo, K., Barajas, L. G., & Menassa, R., "3D visual perception system for bin picking in automotive sub-assembly automation", *IEEE International Conference on Automation Science and Engineering (CASE)*, 2012.

[4] Hsu, C.-Y., Kang, L.-W., You, T.-Y., & Jhong, W.-C., "Vision-Based Automatic Identification Tracking of Steel Products for Intelligent Steel Manufacturing", *IEEE International Symposium on Multimedia (ISM)*, 2017.

[5] Bahaghighat, M., Mirfattahi, M., Akbari, L., & Babaie, M., "Designing quality control system based on vision inspection in pharmaceutical product lines", *International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, 2018.

[6] He, Ying & Liang, Bin & Yang, Jun & Li, Shunzhi & He, Jin., "An Iterative Closest Points Algorithm for Registration of 3D Laser Scanner Point Clouds with Geometric Features", *Sensors*. 17. 1862. 10.3390/s17081862., 2017.

[7] Aranjuelo N, et al. "Robust 3D Object from LiDAR Point Cloud Data with Spatial Information Aggregation", DOI : 10.1007/978-3-030-57802-2_78, 2021.

[8] Ringbeck, Dr.-Ing. Thorsten, "A 3 D TIME OF FLIGHT CAMERA FOR OBJECT DETECTION", 2007.

[9] Kuo, H.-Y., Su, H.-R., Lai, S.-H., & Wu, C.-C., "3D object detection and pose estimation from depth image for robotic bin picking", *IEEE International Conference on Automation Science and Engineering (CASE)*, 2014.

[10] Han, Xian-Feng & Jin, Jesse & Xie, Juan & Wang, Ming-Jie & Jiang, Wei." A comprehensive review of 3D point cloud descriptors", 2018.

[11] Chen, Chin-Sheng, Po-Chun Chen and Chih-Ming Hsu., "Three-Dimensional Object Recognition and Registration for Robotic Grasping Systems Using a Modified Viewpoint Feature Histogram.", *Sensors (Basel, Switzerland)* 16, 2016.

[12] Wang, Fei & Liang, Chen & Ru, Changlei & Cheng, Hongtai., "The Iterative Closest Point Registration Algorithm Based on the Normal Distribution Transformation", 2018.

[13] F. Wang, et al, "An Improved Point Cloud Descriptor for Vision Based Robotic Grasping System", 2019, DOI : 10.3390/s19102225.