Surface 3D Scanner Using Time of Flight Ranging Sensor with Cylindrical Coordinate System

Achmad Purnomo Wijaya\textsuperscript{1,a}, Niam Tamami\textsuperscript{2,b}, Hary Oktavianto\textsuperscript{3,c}
\textsuperscript{1,2,3}Electronics Engineering Department, Politeknik Elektronika Negeri Surabaya, Indonesia
\textsuperscript{a}achmadpurnomo95@gmail.com, \textsuperscript{b}niam@pens.ac.id, \textsuperscript{c}hary@pens.ac.id

Abstract.

3D scanner that uses image sensors requires the role of a computer that includes a data generator, data acquisition, and visual display. In a simply system, it can be designed the sensory system uses non-imagery sensor so the role of the data generator can be handled by the microcontroller. This research aims to make a simple 3D scanner using inexpensive non-imagery Time of Flight VL53L0X sensor and data processing can be processed directly by the microcontroller. The results of sensor distance measurements are processed on the microcontroller and desktop application. The distance and angle values are converted into Cartesian coordinate using cylindrical coordinate system. The scan results of the cubes, prisms and cylinder are similar with the reference object, but the results of the pyramid test at the top cannot be scanned properly due to the narrow surface. The laser beam from the emitter cannot bounce back to the collector properly makes distance reading is inaccurate and causes error in the point cloud conversion. The comparison error between the side of the scan results and the reference object is between 2.54-39.8%. The surface of objects with bright color has a smaller error than those with dark color. The comparison error of the height of the scan results with the reference object is between 5-32%. The angle of the emitter exclusion cone and the collector exclusion cone sensor affects the error at the side and height of the scan results.

Keywords: VL53L0X sensor, cylindrical coordinate system, point cloud
INTRODUCTION

3D object scanning technology has been developed since 1960 using light, camera and projector [1]. Several methods for scanning 3D objects such as white light scanning, photogrammetry, machine vision, and laser scanning [2]. Laser scanning can be applied in the construction [3] and orthodontics [4]. The laser scanning method requires a camera, computer and good lighting [5].

Machine vision, laser scanning, and photogrammetry are types of 3D scanner that use image sensor and require computer roles that include data generator, data acquisition and visual display. The computer function in this system is more dominant because it compensates for the high specification of the sensor device. In a simpler system, a sensory system using a non-image sensor can be designed so the role of generating data can be handled by the microcontroller and the computer only as a visual viewer. The method used is to get the distance between the surface of the 3D object and the position of the sensor [6].

To get the 3D surface distance from the sensor, an ultrasonic sensor can be used as in [7] [8]. The results of 3D scanning using an ultrasonic sensor on 2 out of 5 bottles did not match the shape of the reference bottle, there was an indentation in the bottle so that the sensor could not accurately determine the size of the bottle [9]. Research [10] used the VL53L0X sensor to scan convex objects, but it is not suitable for scanning objects with many cavities. [11] It scans the surface of a wavy object and requires an adaptive filter to smooth the surface.

This research aims to create a 3D object surface scanner using a non-image Time of Flight VL53L0X sensor handled by a microcontroller as a controller and as a data generator. Cartesian coordinate data is displayed as a point cloud on a desktop based application. This application has control functions to control the scanner, process the scan results using filters, and convert the 3D scanned object into a stereolithographic format (.stl).

Proposed Approach

The scanner consists of hardware and software. The hardware consists of the Time of Flight VL53L0X sensor to get the distance from the 3D object, the upper limit switch and the lower limit switch as the vertical limit of the sensor, the A4988 driver to adjust the stepper motor, the NEMA17 stepper motor to rotate the 3D object 360 degrees and move the Time of Flight VL53L0X vertically, Arduino Nano as microcontroller, and SD Card to save the cartesian coordinates of the scan results in (.txt) format. Communication between the scanner and the application on the desktop uses serial communication. Figure 1 shows a block diagram for the scanner.
The following figure shows the sequence of data processing in this study. The distance reading of the VL53L0X sensor through a filter process uses a moving average filter on the microcontroller to reduce noise. The next process is the conversion of distances and angles using cylindrical coordinates to Cartesian coordinates. Cartesian coordinates data displayed on the application becomes a point cloud. In-app averaging filter is used to reduce noise outside the point cloud. The last process is to convert the point cloud to (.stl) format using a library from Matlab [12].
Method.

Figure 4 shows an illustration of the scanner. The scanned object is orange. The table can rotate according to the desired angle reference, the Time of Flight VL53L0X sensor can move vertically, so that the sensor reading point is obtained in the form of cylindrical coordinates. Cylindrical coordinate data is converted into a Cartesian coordinate data.

![Figure 4 Scanning illustration](image)

The illustration in the image above can be concluded with the following equation:

\[ C = A - B \]  

(1)

Where:
A = The distance between the center of the table and the sensor
B = The distance between the sensor and the surface of the 3D object
C = The distance between the center of the table and the surface of the 3D object

![Figure 5 Scanning illustration top view](image)

Figure 5 shows a scanning illustration from top view. Turntable to rotate 3D objects so that all object surfaces can be scanned. The rotation of the table creates an angle \( \theta \) between the point being scanned and the point to be scanned. The angle \( \theta \) is affected by the stepper motor. In full step setting, the NEMA17 stepper motor delivers 200 revolutions for one full 360 ° turn. By using cylindrical coordinates, the Cartesian coordinates of each point are obtained using the following equation:
\[ x = c \cdot \cos \theta \]  \hspace{1cm} (2)
\[ y = c \cdot \sin \theta \] \hspace{1cm} (3)
\[ z = \text{rotation of the stepper motor sensor} \] \hspace{1cm} (4)

The following image shows the flowchart of the scanning process in this research.

**Figure 6** Scanning process flowchart

**Moving Average Filter.**
Moving average is a filter to find the average value of the value of an instrument at a certain time. The moving average filter can be represented as a series of numbers and a fixed subset. The first element of the moving average is obtained by averaging a subset of a series of numbers \([13]\). The following is an equation on the Moving Average.

\[ y[n] = \frac{1}{m} \sum_{k=0}^{m-1} x[n - k] \] \hspace{1cm} (5)

**Mean Filter.**
The mean filter is an average filter of the intensity of several local pixels where each pixel will be replaced by the average of the intensity value of these pixels with neighboring pixels, and the number of neighboring pixels involved depends on the filter designed. The mean filter is to replace the pixel value at position \((x, y)\) with the total pixel value in the surrounding neighbors \(\sum_{(s,t) \in S_{xy}} g(s,t)\) and divided by the amount of data \((m \times n)\) \([14]\). The following is the equation of the mean filter.

\[ f(x,y) = \frac{1}{mn} \sum_{(s,t) \in S_{xy}} g(s,t) \] \hspace{1cm} (6)
Electronics Circuit.

A. VL53L0X Sensor

The VL53L0X sensor is to measure distances using a laser with the Time of Flight method. This sensor is integrated with the VCSEL 940mm IR laser and embedded microcontroller with the sensor. It can measure absolute distances up to 2 m and using I2C communication. The working principle of this sensor is the laser beam directed at the object and then reflected back and received by the sensor. The emitter exclusion cone angle is 35° and the collector exclusion cone angle is 25° [15].

![VL53L0X sensor](image)

**Figure 7 VL53L0X sensor**

B. Regulator Circuit

12VDC is used as a power source for 2 stepper motors. To get a 5VDC voltage using the LM7805 regulator. The output from the regulator circuit is used for the microcontroller power source, stepper motor driver, VL53L0X sensor, and SD Card module.

![Regulator circuit](image)

**Figure 8 Regulator circuit**

C. Microcontroller and A4988 Stepper Driver

![Microcontroller and stepper driver](image)

**Figure 9 Microcontroller and stepper driver**
The microcontroller controls the stepper motor via the A4988 stepper driver. The step settings are on the MS1, MS2 and MS3 pins. A4988 stepper driver configuration is full step, half step, quarter step, eighth step, sixteenth step.

D. Microcontroller with VL53L0X sensor and Limit Switch

![Diagram of microcontroller with VL53L0X sensor and limit switch]

**Figure 10** Microcontroller with VL53L0X sensor and limit switch

The VL53L0X sensor is connected to the microcontroller using I2C communication. 2 limit switches are connected to the digital pins of the microcontroller.

E. Microcontroller with SD Card Module

![Diagram of microcontroller with SD card module]

**Figure 11** Microcontroller with SD card module

SD Card module is connected to the microcontroller via SPI communication. Microcontroller as master and SD Card module as slave.

The Application.

The application functions to display scan results, control scanner tool, process scan results using the mean filter, and convert them to stereolithography (.stl) format. Application created using Matlab software and have been converted into application with the extension (.exe).
The following is a flowchart of application in this research.

![Flowchart](image)

**Figure 12** The application

**Figure 13** The application flowchart

**Experimental Results**

**A4988 stepper driver for controlling the NEMA 17 stepper motor in various steps.**

The test was implemented by rotating the stepper motor for 1 full rotation with a configuration of full steps, half steps, quarter steps, eighth steps, sixteen steps. Stepper settings on the stepper driver are implemented on the microcontroller. This test aims to obtain the angle \( \theta \) generated by the stepper motor in 1 step and to determine the angle generated by the stepper motor accurately and precisely.
Table 1 Stepper motor table test result

<table>
<thead>
<tr>
<th>Stepper motor rotation</th>
<th>Configuration</th>
<th>Direction</th>
<th>Total angle value (°)</th>
<th>The theoretical value of the angle per step (°)</th>
<th>Angle measurement per one step (°)</th>
<th>Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>full step</td>
<td>CW</td>
<td>360</td>
<td>1.8</td>
<td>1.8</td>
<td>0</td>
</tr>
<tr>
<td>200</td>
<td>full step</td>
<td>CCW</td>
<td>360</td>
<td>1.8</td>
<td>1.8</td>
<td>0</td>
</tr>
<tr>
<td>400</td>
<td>half step</td>
<td>CW</td>
<td>360</td>
<td>0.9</td>
<td>0.9</td>
<td>0</td>
</tr>
<tr>
<td>400</td>
<td>half step</td>
<td>CCW</td>
<td>360</td>
<td>0.9</td>
<td>0.9</td>
<td>0</td>
</tr>
<tr>
<td>800</td>
<td>quarter step</td>
<td>CW</td>
<td>360</td>
<td>0.45</td>
<td>0.45</td>
<td>0</td>
</tr>
<tr>
<td>800</td>
<td>quarter step</td>
<td>CCW</td>
<td>360</td>
<td>0.45</td>
<td>0.45</td>
<td>0</td>
</tr>
<tr>
<td>1600</td>
<td>eighth step</td>
<td>CW</td>
<td>360</td>
<td>0.225</td>
<td>0.225</td>
<td>0</td>
</tr>
<tr>
<td>1600</td>
<td>eighth step</td>
<td>CCW</td>
<td>360</td>
<td>0.225</td>
<td>0.225</td>
<td>0</td>
</tr>
<tr>
<td>3200</td>
<td>sixteenth step</td>
<td>CW</td>
<td>360</td>
<td>0.1125</td>
<td>0.1125</td>
<td>0</td>
</tr>
<tr>
<td>3200</td>
<td>sixteenth step</td>
<td>CCW</td>
<td>360</td>
<td>0.1125</td>
<td>0.1125</td>
<td>0</td>
</tr>
</tbody>
</table>

From the test results of the stepper sensor motor and the stepper motor table with CW and CCW rotation using various stages, full step, half step, quarter step, eighth step, and sixteenth step have 0% error. The measured rotation angle of the stepper motor is in accordance with the angle of rotation in theory.

Sensor measurement with 3D object from 15 mm to 135 mm.

The distance measurement from 15 mm to 135 mm is based on the closest and farthest 3D distance to the object used in this research. The tested 3D object is a yellow cube with dimensions of 50x50x50 mm. Lighting conditions using a 20 watt fluorescent lamp with a height of 2 m. This test aims to determine the linearity of the VL53L0X sensor measurements.
Table 3 Sensor measurement testing

<table>
<thead>
<tr>
<th>Distance (mm)</th>
<th>Measurement (mm)</th>
<th>Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>14</td>
<td>6.67</td>
</tr>
<tr>
<td>25</td>
<td>28</td>
<td>12.00</td>
</tr>
<tr>
<td>35</td>
<td>35</td>
<td>0</td>
</tr>
<tr>
<td>45</td>
<td>43</td>
<td>4.44</td>
</tr>
<tr>
<td>55</td>
<td>52</td>
<td>5.45</td>
</tr>
<tr>
<td>65</td>
<td>62</td>
<td>4.62</td>
</tr>
<tr>
<td>75</td>
<td>74</td>
<td>1.33</td>
</tr>
<tr>
<td>85</td>
<td>85</td>
<td>0</td>
</tr>
<tr>
<td>95</td>
<td>96</td>
<td>1.05</td>
</tr>
<tr>
<td>105</td>
<td>106</td>
<td>0.95</td>
</tr>
<tr>
<td>115</td>
<td>116</td>
<td>0.87</td>
</tr>
<tr>
<td>125</td>
<td>126</td>
<td>0.80</td>
</tr>
<tr>
<td>135</td>
<td>135</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 14 Sensor measurement graph

From the test results above, it can be concluded that the sensor distance measurement results have linear characteristic. The maximum error is 12% and the minimum error is 0%. The biggest error occurs at a distance under 25 mm from the sensor. The closer the object is to the sensor, the bigger the error, while the farther the object is from the sensor, the smaller the error.

Scanning of cube, prism, pyramid, cylinder and simple 3D objects.

Testing with various stepper motor step configurations such as full step, half step, quarter step, eighth step, and sixteenth step. The scan results were processed using an average kernel size filter of 10x10. The vertical step in this test is 2 mm. This test aims to obtain Cartesian coordinate from 3D objects using cylindrical coordinate.
<table>
<thead>
<tr>
<th></th>
<th>Cube</th>
<th>Prism</th>
<th>Pyramid</th>
<th>Cylinder</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full step</td>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
</tr>
<tr>
<td>Half step</td>
<td><img src="image5.png" alt="Image" /></td>
<td><img src="image6.png" alt="Image" /></td>
<td><img src="image7.png" alt="Image" /></td>
<td><img src="image8.png" alt="Image" /></td>
</tr>
<tr>
<td>Quarter step</td>
<td><img src="image9.png" alt="Image" /></td>
<td><img src="image10.png" alt="Image" /></td>
<td><img src="image11.png" alt="Image" /></td>
<td><img src="image12.png" alt="Image" /></td>
</tr>
<tr>
<td>Eighth step</td>
<td><img src="image13.png" alt="Image" /></td>
<td><img src="image14.png" alt="Image" /></td>
<td><img src="image15.png" alt="Image" /></td>
<td><img src="image16.png" alt="Image" /></td>
</tr>
<tr>
<td>Sixteenth step</td>
<td><img src="image17.png" alt="Image" /></td>
<td><img src="image18.png" alt="Image" /></td>
<td><img src="image19.png" alt="Image" /></td>
<td><img src="image20.png" alt="Image" /></td>
</tr>
</tbody>
</table>

**Table 4 3D object scanning**
Table 5 Simple 3D object scanning

<table>
<thead>
<tr>
<th>Object 1</th>
<th>Object 2</th>
<th>Object 3</th>
<th>Object 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full step</td>
<td>Full step</td>
<td>Full step</td>
<td>Full step</td>
</tr>
<tr>
<td>Half step</td>
<td>Half step</td>
<td>Half step</td>
<td>Half step</td>
</tr>
<tr>
<td>Quarter step</td>
<td>Quarter step</td>
<td>Quarter step</td>
<td>Quarter step</td>
</tr>
<tr>
<td>Eighth step</td>
<td>Eighth step</td>
<td>Eighth step</td>
<td>Eighth step</td>
</tr>
</tbody>
</table>
From the scan results of the cube object with the reference object, there are dissimilarities on the sides of the cube. This is because the side of the cube has a narrow surface area, when the light from the emitter reaches the side of the cube it cannot return to the collector properly so that the distance reading is inaccurate and causes errors in the conversion to point cloud. This also applies to the blunt sides of the prism. The surface of the cube and prism is slightly wavy due to less noise in the point cloud. In the scan results of the pyramid, the surface of the pyramid is wavy, this is because the side surface of the object is slanted. At the peak of the scan, the pyramid is truncated because at the top of the object the pyramid has a small surface area, making it difficult for the sensor to detect that part. In the object the tube has sides with different areas because when the scanning process the object is shifted from its original place. The side of the tube scan results in accordance with the original object.

The 3D scan results of object 1 on the front surface are not similar to the reference object because the VL53L0X sensor has an accuracy of 1-3 mm so it cannot measure the distance of the object's surface accurately. The results of the scanning of objects 2, 3, and 4 have an error equation on the side of the object there is an uneven curve, but overall the scan results are in accordance with the shape of the object.

**Compare the dimensions of the scan results with the reference object.**

The objects in this test are cube, prism, and pyramid. Each object is black and white with dimensions of 80 mm, 65 mm, and 50 mm. Tests were carried out 10 times on each object. To find out the dimensions of the scan using the SketchUp software. In this test using the scan results with the format (.stl). This test aims to determine the dimensions of the 3D scan results of the reference object.
From the comparison of the dimensions of the scan results with the 3D reference object, cube, prism, and pyramid object has varying errors on the side which is influenced by the accuracy of the sensor distance measurement. Scanning results on the surface of an object with a white or bright surface produce smaller errors when compared to a black or dark surface. A reference object size that is too small will cause a large error. The incompatibility of the distance measurement results by the sensor with the reference distance, thus affecting the conversion results into point cloud. The height error is influenced by the vertical step setting in the application, the smaller the vertical step value
used, the scanner can scan each side, but if the larger the vertical step value is used, the scanner can only scan a certain side.

**Conclusion and Recommendation**

The scan results of the cube, prism, and cylinder objects match the reference object, the scan results of the pyramid are truncated because the peak of the pyramid object has a small surface area, making it difficult for the sensor to detect that part. 3D scanning objects have indentations on both sides that do not match the reference object. The comparison error between the side of the scan results and the reference object is between 2.54-39.8%. The surface of an object with a light color has a smaller error than that of a dark color. The comparison error of the height of the scan results with the reference object is between 5-32%. The smaller the 3D object size, the bigger the error.

**Recommendation.**

The VL53L0X sensor has a weakness in measuring distance with a small and light surface. To improve the scan results it is necessary to add a sensor at a certain angle so that it can measure a small surface and overcome the lack of sensors on the 3D surface of an object that has a slope. It is necessary to add a filter algorithm to remove unwanted noise in certain parts of the scan results so that the details of the desired object are not lost. In addition, it requires further processing in certain parts of the scan that do not match the actual object using 3D editing software.

**REFERENCES**


