NATIONAL INSTITUTE OF BUSINESS MANAGEMENT

HIGHER DIPLOMA IN SOFTWARE ENGINEERING 23.1

DIGITAL IMAGE PROCESSING

MAKING A 3D SCANNER – ENHANCING 3D PRINTER

SUBMITTED BY

KAHDSE231F-011	Geeneth Madagedara
KAHDSE231F-013	Chanuka Wanigasekara
KAHDSE231F-014	KanchanaWijesooriya
KAHDSE231F-015	Resandu Udagedara
KAHDSE231F-018	W.G.S.M Bandara
KAHDSE231F-019	Kavinda Dissanayake
KAHDSE231F-020	H.H.D Brian
KAHDSE231F-021	Lahiru Rathnayake

Date of Submission: 18/11/2023

Summary

The 3D Scanner project uses inexpensive parts like motors, IR sensors, and Arduino to build a 3D scanning solution that is economical. By making 3D scanning more widely available, the initiative hopes to provide aficionados of electronics and programming a practical introduction to these subjects. Although its open-source nature and ability to be customized, limitations include the complexity of calibration and possible accuracy limits. The initiative enables users to digitize real-world objects for a variety of uses, fosters creativity and community participation, and functions as an educational tool. It's a good starting point to learn more about the field of 3D scanning because of its versatility and price.

Declaration

We certify that everything we have written for this report is completely unique and our own work. All facts, concepts, and material collected from other sources have been properly referenced and mentioned according to our institution's academic standards and rules.

Name	Index No.
Geeneth Madagedara	KAHDSE231F-011
Chanuka Wanigasekara	KAHDSE231F-013
Kanchana Wijesooriya	KAHDSE231F-014
Resandu Udagedara	KAHDSE231F-015
W.G.S.M Bandara	KAHDSE231F-018
Kavinda Dissanayake	KAHDSE231F-019
H.H.D Brian	KAHDSE231F-020
Lahiru Rathnayake	KAHDSE231F-021

Supervisor: Mr. Sandaruwan Herath

Date: 2023/12/18

Institute: National Institute of Business Management

Project Name: Digital Image Processing (Making a 3D Scanner - Enhancing 3D Printer)

Acknowledgement

Our sincere gratitude goes out to everyone who helped make the 3D Scanner -Enhancing 3D printer idea a reality. Our great thanks go out to NIBM for providing the required resources and support, our supervisor(s) for their priceless advice, and our committed team members for their diligence and inventiveness. We express our gratitude to our family and friends for their consistent support and faith in this project. Without the cooperation and assistance of all parties involved—individuals and Scanner printer achievable.

Table of Contents

CHAPTER 01

Introduction

The "3D Scanner with IR Sensor" project offers users an easy and affordable way to build 3D scanning devices. With the use of common parts like stepper motors, an Arduino, and a Sharp infrared sensor, this project makes it easier to investigate 3D scanning technologies. The technology, which is aimed toward education, developing prototypes, and personal usage, records spatial data and offers an affordable option for individuals who want to explore the field of 3D scanning.

3D Object Scanner

The "3D Scanner with IR Sensor" is designed to make 3D scanning more accessible and provides consumers with an affordable, educational option. This project makes studying electronics and programming more practical. Its affordability, which opens up 3D scanning technology to a wider audience, and its versatility for prototyping, which encourages creativity inside projects, are its two main advantages. By enabling users to delve further into the intriguing realm of 3D scanning, the scanner fosters skill development and unleashes creative energy among the maker community.

Problem Statement and Objectives

The "3D Scanner with IR Sensor" project aims to provide users with an affordable and useful 3D scanning option. Stepper motors, an Arduino board, and Sharp Infrared distance sensor 2Y0A710K(1f) are combined in this project to create an easy learning environment for electronics and programming. The main objective of the system is to collect spatial data so that users can build 3D models for educational purposes, prototypes, and personal projects.

CHAPTER 02

Main Components and Methods

Infrared distance sensor 2Y0A710K

Purpose: Measures distance during scanning, giving important data for creating a 3D point cloud.

Figure 1| IR Distance sensor

Nano V3 (Arduino)

Purpose: Works as the main control unit, coordinating the motor motions, processing data from the IR sensor, and controlling the scanning procedure as an entire process.

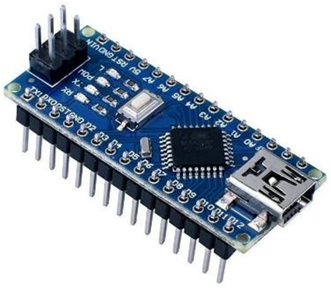


Figure 2| Nano V3 (Arduino)

Stepper Motor Driver A4988 and Nema 17 Stepper Motor

Purpose: Use one motor to control the scanning platform's horizontal movement and another to control its vertical movement.

Figure 3| Stepper Motor Driver A4988

Figure 4| Nema 17 Stepper Motor

SD Module and Micro SD Card

Purpose: Saves the gathered information for simple retrieving and further analysis of the 3D scanning outcomes.

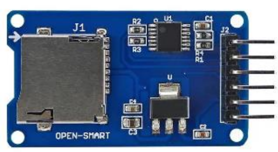


Figure 5| SD Module

Resistors and Push Button

Prototyping and Methods

1. **Mechanical Assembly:**
    - Mount NEMA17 motors on a frame for horizontal and vertical movement.
    - Attach smooth rods and bearings for stability.
    - Add a lead screw for precise vertical control.
2. **Establish Electronic Connections:**
    - For motor control, connect A4988 drivers to NEMA17 motors.
    - To measure distance, connect an Arduino board to a Sharp infrared sensor.
    - Establish connections to the SD module and power supply.
3. **Programming and Perform DIP Operations:**
    - To read data from the IR sensor, store data on the SD card, and operate motors, write the Arduino code.
    - Implement logic for the scanning sequence and **Image data acquisition.**
4. **A Combination:**
    - Combine electrical and mechanical parts to create an uninterrupted unit.
    - Establish correct alignment and connect all connections.
5. **Testing and Calibration:**
    - Check the functionality of each component separately.
    - Test the IR sensor's accuracy, the motor motions' precision, and the data storage's accuracy.
    - Optimize the system to produce accurate 3D reconstructions.
    - As necessary, change the sensor's parameters, the motor steps, and other factors.

CHAPTER 03

Working Principle of 3D Scanner

NEMA17 motors are used to coordinate the exact movement of the scanning platform in order to run the 3D Scanner with IR Sensor. The Sharp IR sensor generates a dataset of spatial information as the platform moves across the scanning area by emitting infrared light and measuring the reflection time to determine distances. This procedure is managed by the Arduino microcontroller, which logs the distance information onto a micro-SD card for further processing. Software analyses the data during the post-processing phase to create a 3D point cloud that captures the shape of the scanned scene. The scanner's operation depends on the exact synchronization of mechanical motions, the IR sensor's accurate distance measurements, and the Arduino's efficient data handling, all of which combine to provide makers and hobbyists with an accessible and affordable D3D scanning solution.

3D Scanner PCB schematic

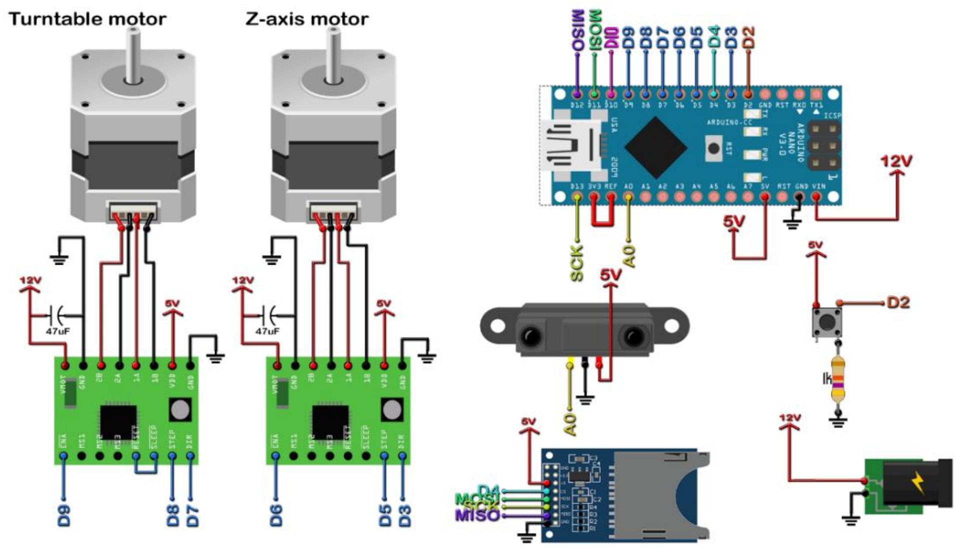


Figure 6| 3D Scanner PCB schematic

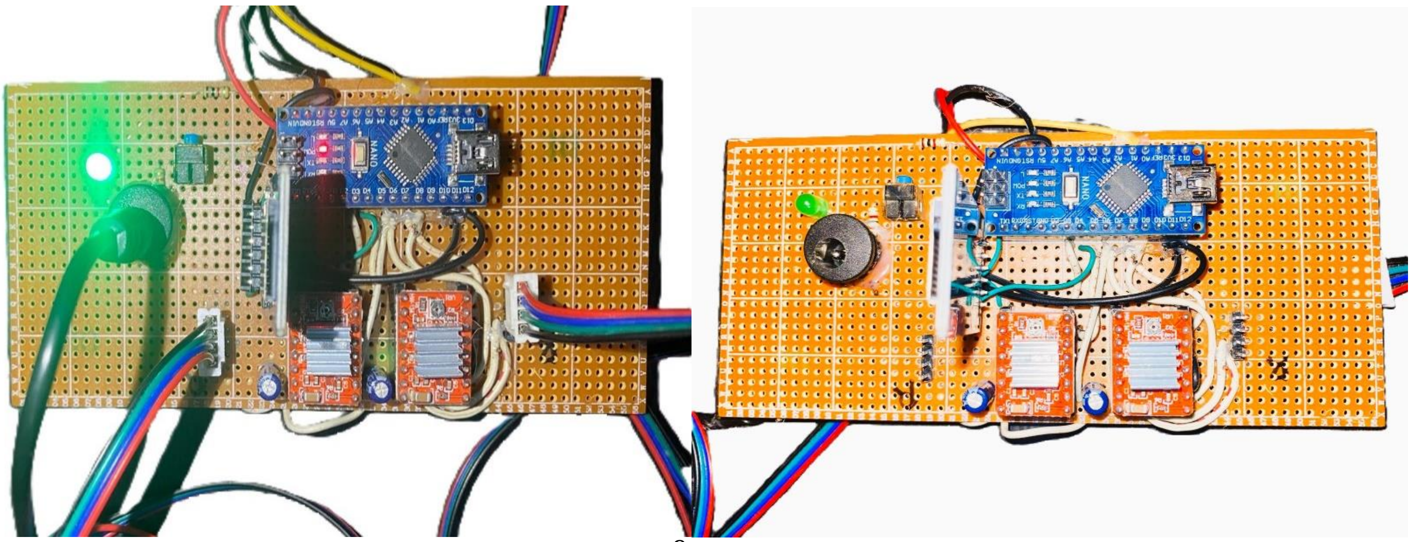
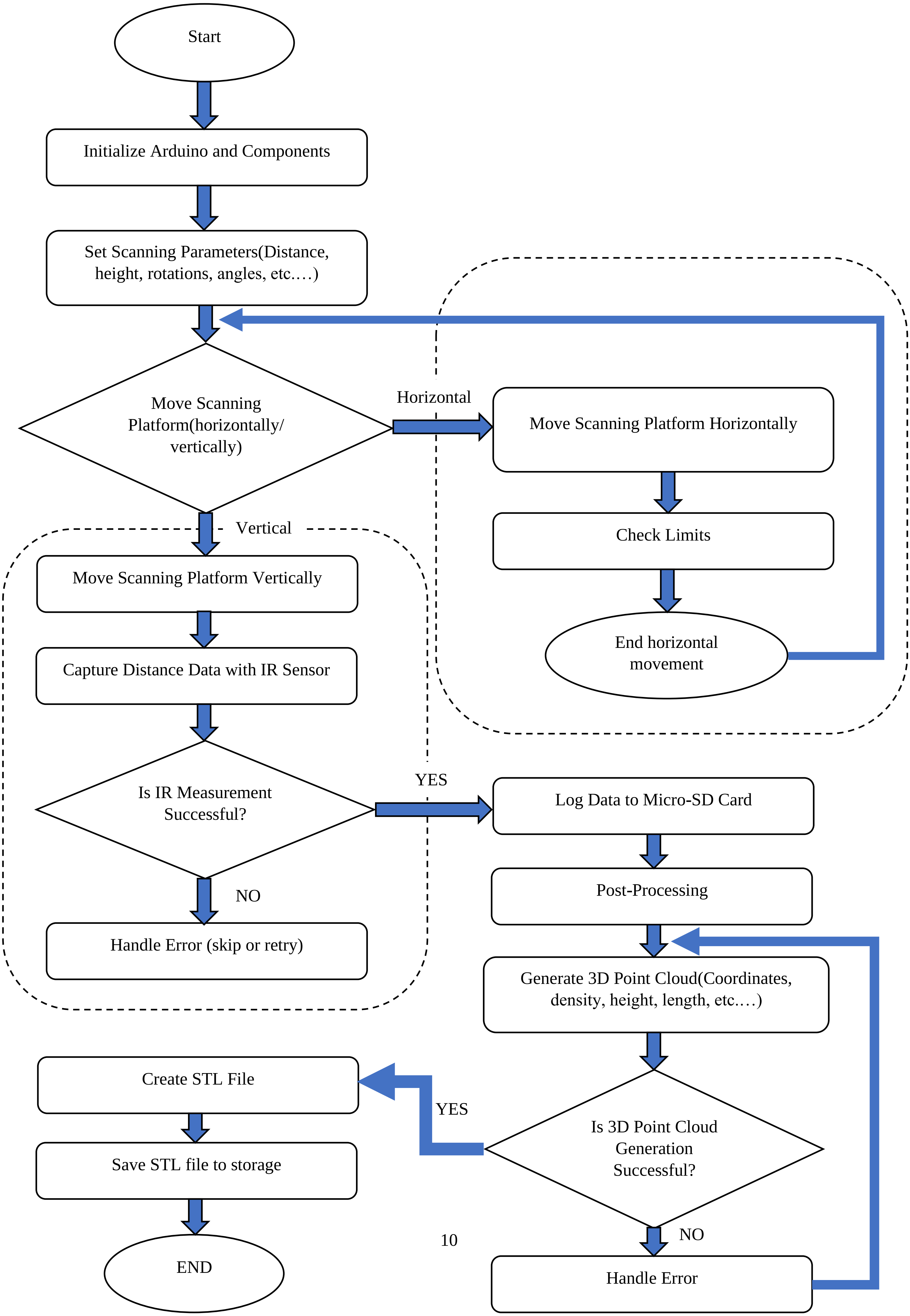


Figure 7.1| Printed Circuit Board of 3D Scanner


Start
Initialize Arduino and Components
Set Scanning Parameters(Distance, height, rotations, angles, etc.…)
Move Scanning Platform(horizontally/ vertically)
Horizontal
Move Scanning Platform Horizontally
Check Limits
End horizontal movement
Vertical
Move Scanning Platform Vertically
Capture Distance Data with IR Sensor
Is IR Measurement Successful?
YES
NO
Handle Error (skip or retry)
Log Data to Micro-SD Card
Post-Processing
Generate 3D Point Cloud(Coordinates, density, height, length, etc.…)
Is 3D Point Cloud Generation Successful?
YES
NO
Handle Error
Create STL File
Save STL file to storage
END

Post Processing: Develop or configure software for post-processing, converting captured distance data into a 3D point cloud.

Digital Image Processing Operation(s)

Image Sensing and Acquisition:

The image data acquisition process revolves around the operation of the Sharp IR sensor.

IR Sensor Operation:

- The Sharp IR sensor emits infrared light towards the scanned area.

Reflection Measurement:

- The sensor measures the time it takes for the emitted infrared light to reflect off surfaces in its field of view and return to the sensor.

Distance Calculation:

- Using the measured reflection time, the Arduino microcontroller(PCB) calculates the distance between the IR sensor and the surfaces in its line of sight.

Data Logging:

- Arduino logs the distance data obtained from the IR sensor. This data may include X, Y, and Z coordinates, representing the **spatial information** of the scanned points.

Incremental Scanning:

- The scanning platform, controlled by stepper motors, moves incrementally to cover the desired scanning area. The IR sensor continues to measure distances at each position.

Keep continuing this process and recording and combining sensor's data to generate a 2D image for STL file.

How are calculations recorded?

1. **Distance Calculation:**

   The distance (D) between the IR sensor and the object's surface can be calculated using the formula:

   D = $\frac{(\text{Speed of Light} \times \text{Time of Flight})}{2}$

2. **Coordinates Conversion:**

   The Cartesian coordinates (X,Y,Z) of the scanned points can be obtained based on the trigonometric relationships with the distance and the angles of rotation or movement. For example, if you have polar coordinates (r,θ, φ) (distance, azimuth angle(z-y plane) , and altitude angle(x-y plane) respectively), the conversion to Cartesian coordinates is:

   X= $r \sin\varphi \cos\theta$

   Y= $r \sin\varphi \sin\theta$

   Z= $r \cos\varphi$

   r = $\sqrt{x^2+y^2+z^2}$

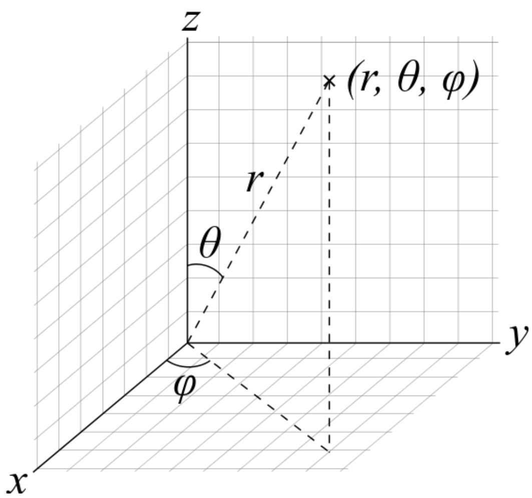


Figure 8| 3D Coordinate Systems – Spherical

3. **Incremental Scanning:**

   If the scanning platform moves incrementally along the X and Y axes, the new position X′ , Y′ after a movement can be calculated using basic arithmetic:

   X′ = $X + \Delta X$

   Y′ = $Y + \Delta Y$

   Where ΔX and ΔY are represent incremental movements in the X and Y directions, respectively.

CHAPTER 04

Results and Troubleshooting

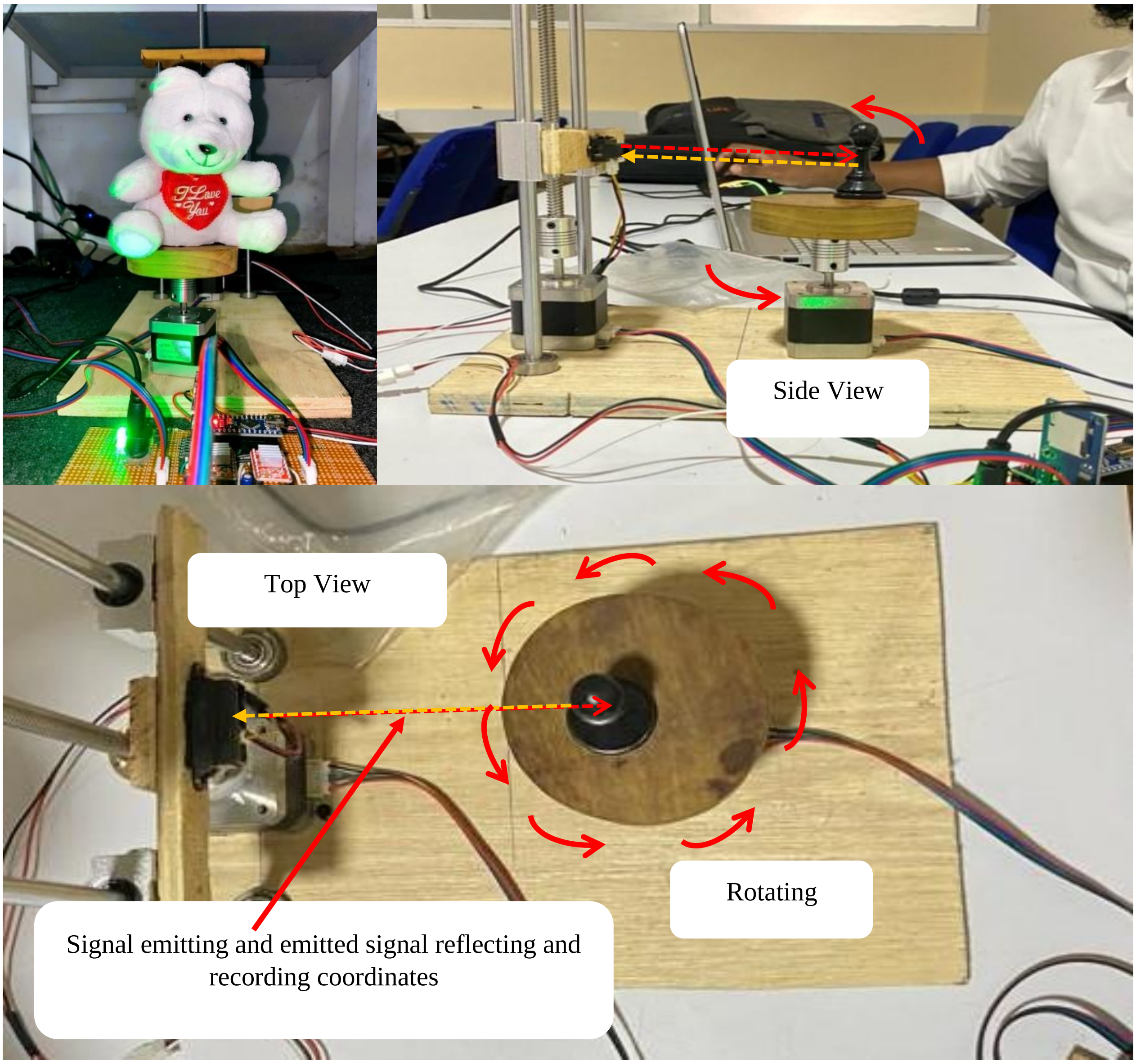
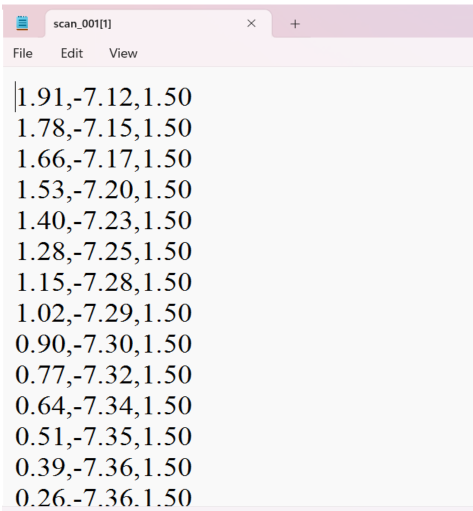


Figure 9| Scanned Objects

- The object's platform rotates while the sensor's platform moves vertically, recording positional coordinates. The code allows customization for the frequency of coordinate recording in a single position. Subsequently, employing MeshLab software facilitates tasks such as meshing, image filtering, and noise removal. The recorded coordinates can then be converted to an STL file. This structured approach enhances precision and versatility in the 3D scanning process, offering comprehensive control over the acquisition of data and subsequent meshing operations.

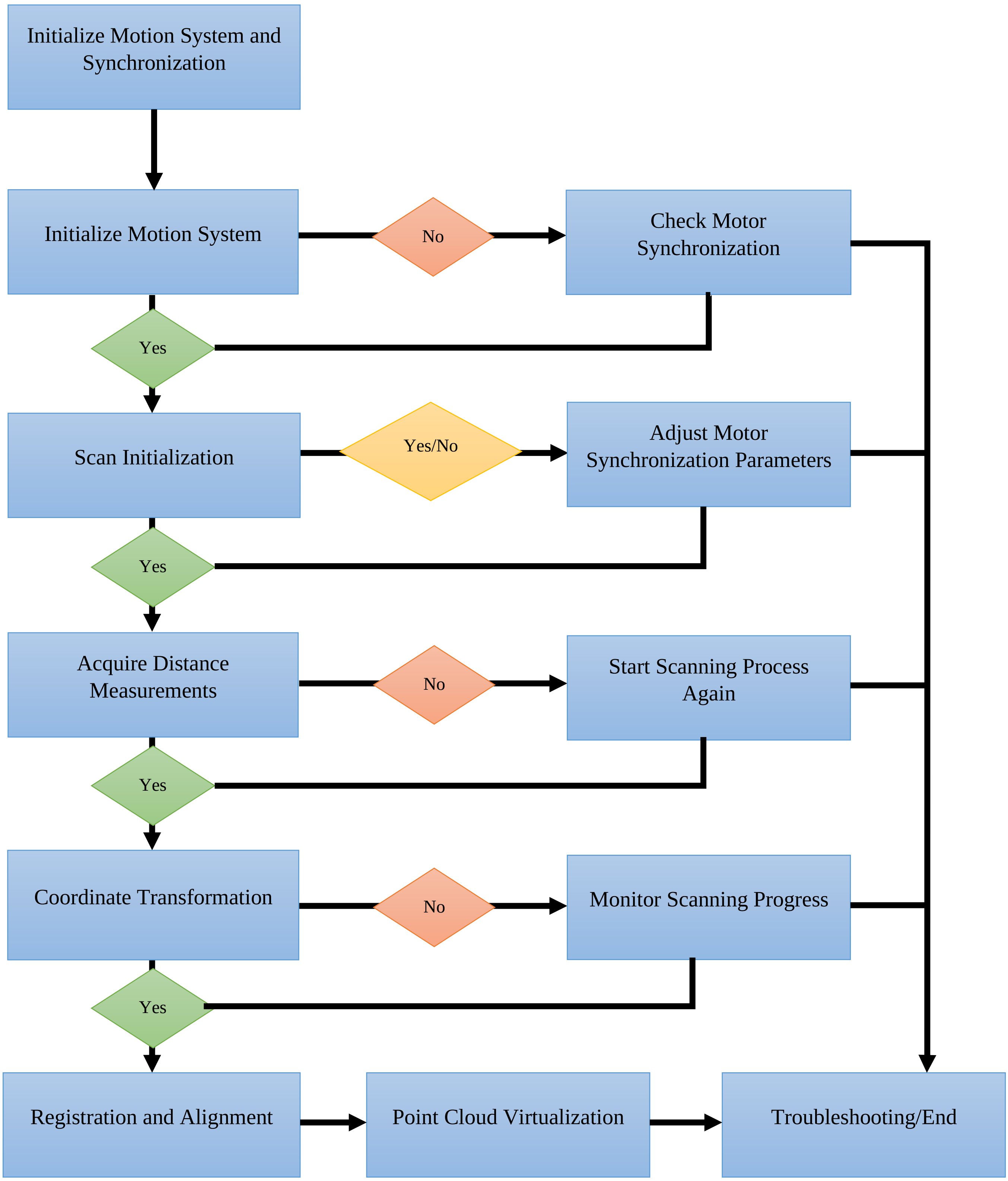

scan_001[1]

File Edit View

1.91,-7.12,1.50
1.78,-7.15,1.50
1.66,-7.17,1.50
1.53,-7.20,1.50
1.40,-7.23,1.50
1.28,-7.25,1.50
1.15,-7.28,1.50
1.02,-7.29,1.50
0.90,-7.30,1.50
0.77,-7.32,1.50
0.64,-7.34,1.50
0.51,-7.35,1.50
0.39,-7.36,1.50
0.26,-7.36,1.50

Figure 9.1| Recorded Coordinates

Troubleshooting of Motion Control and Synchronization



CHAPTER 05

Conclusion and Discussions

Advantages:

- **Cost Effective Solution:** Makes use of commonly available and reasonably priced components, increasing the accessibility of 3D scanning technologies. (Ex: Do not need to use laser camera, easily to manage IR sensor)
- **Customization and Flexibility:** Allows creativity and adaptability to the demands of particular projects by enabling users to customize and customize the system according to their specifications.
- **Scalability:** Allowing users to start with a simple configuration and expand the project slowly as their needs and knowledge increase by adding more features, sensors, or improvements.

Limitations:

- **Limited Scanning volume:** The size of objects that can be scanned may be limited due to restrictions on the scanning volume. The size of objects that can be scanned may be limited due to restrictions on the scanning volume.
- **Limited Compatibility:** The kinds of surfaces that the project can efficiently scan may be limited, especially if such surfaces have transparent or highly reflecting qualities.
- **Hardware Constraints:** When compared to industrial 3D scanners, the usage of standard, low-cost parts may result in **restrictions regarding motor vibration, sensor range**, and overall hardware capabilities.
- **Complex Post-Processing:** The process becomes more complex when creating a useable 3D model and post-processing the data, which may call for more software tools and knowledge. (Used MeshLab Software and calibrate additionally)

Conclusion:

The DIY 3D Scanner project offers an affordable way to start and valuable instructional resources by improving the 3D scanning process. Although adaptable and encouraging community involvement, disadvantages include possible errors and depend on user knowledge. The benefits are in being scalable, easily accessible, and enabling people to experiment with digital fabrication. This project is a good starting point for enthusiasts because it combines real-world use with an educational resource for people interested in 3D modeling and electronics.

References

Dahake, T. K. (2022). *Amazing 3D Scanning | Arduino Project*. Retrieved from https://www.youtube.com/watch?v=rpnSG3B8p9U&list=TLPQMDgxMjIwMjOAo4nKlCb_Xw

DIY Arduino based 3D scanning machine | IR 3D scanner. (n.d.). Retrieved from https://www.youtube.com/watch?v=diuOMDRtIMI&list=TLPQMDgxMjIwMjOAo4nKlCb_Xw&index=5

electronoobs. (2018). *3D Scanner - Arduino + Step motors*. Retrieved from electronoobs: https://electronoobs.com/eng_arduino_tut30.php

hackaday. (2023). *3D-Scanner-based-on-IR-sensor*. Retrieved from hackaday: https://hackaday.io/project/186134-3d-scanner-based-on-ir-sensor

Lidar News Today . (2020, August 01). *3D Coordinate Systems for Lidar Sensor Explained*. Retrieved from lidarnews: https://blog.lidarnews.com/3d-coordinate-systems-explained/

User Manual

Use MESHLAB

Step 01: Download and install MeshLab with the default settings.

Step 02: Open MeshLab and go to "File," then select "Import Mesh." Open the scanned file.

Step 03: In the next window, choose XYZ format, and as a separator, select ','.

Step 04: Now the point cloud is open. We have to give normal to the points.

Step 05: For that, go to "Filter," then "Normal Curvature and Orientation," choose "Compute Normal for Point Set," and in this window, adjust the settings.

Step 06: A value of 10 works well. Click "Apply" and close the window.

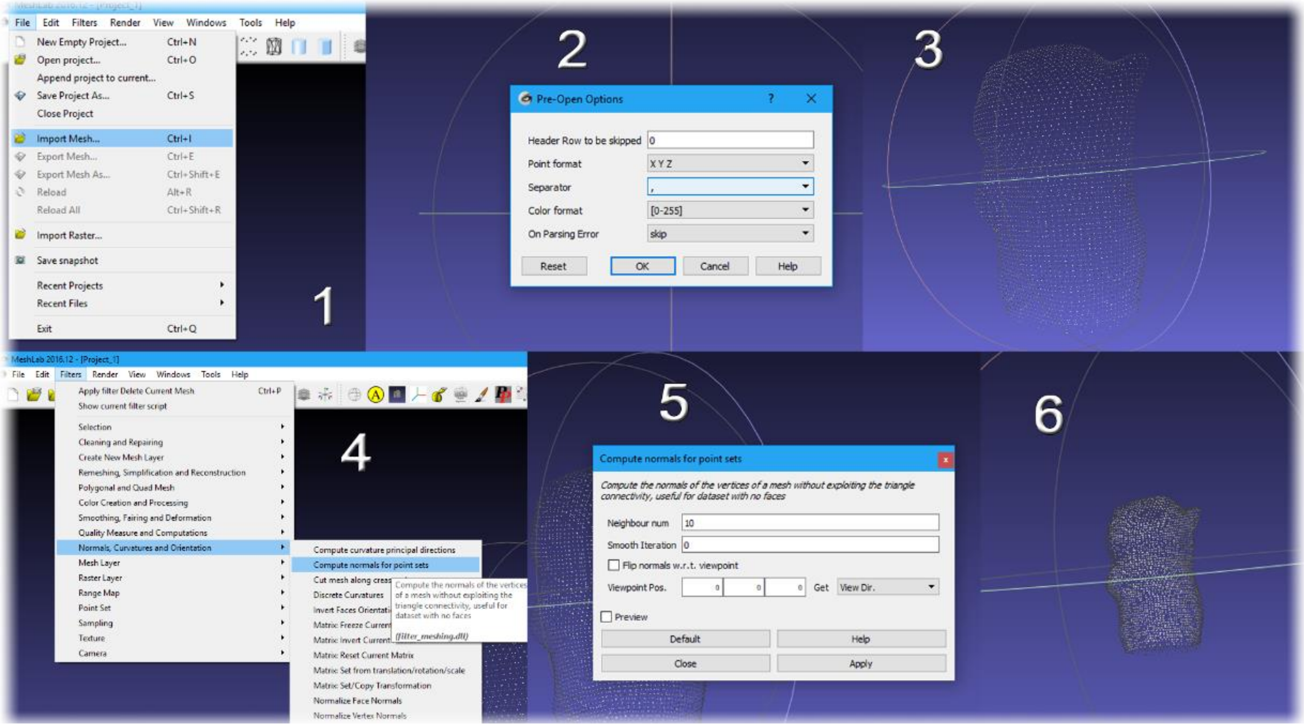


Figure 10| Using MESHLAB step by step 1-6

Step 07: Proceed to the filter section, then pick "screened Poisson surface reconstruction" under "remeshing simplification and reconstruction. Click apply afterward.

Once the window is closed, the STL file is there.

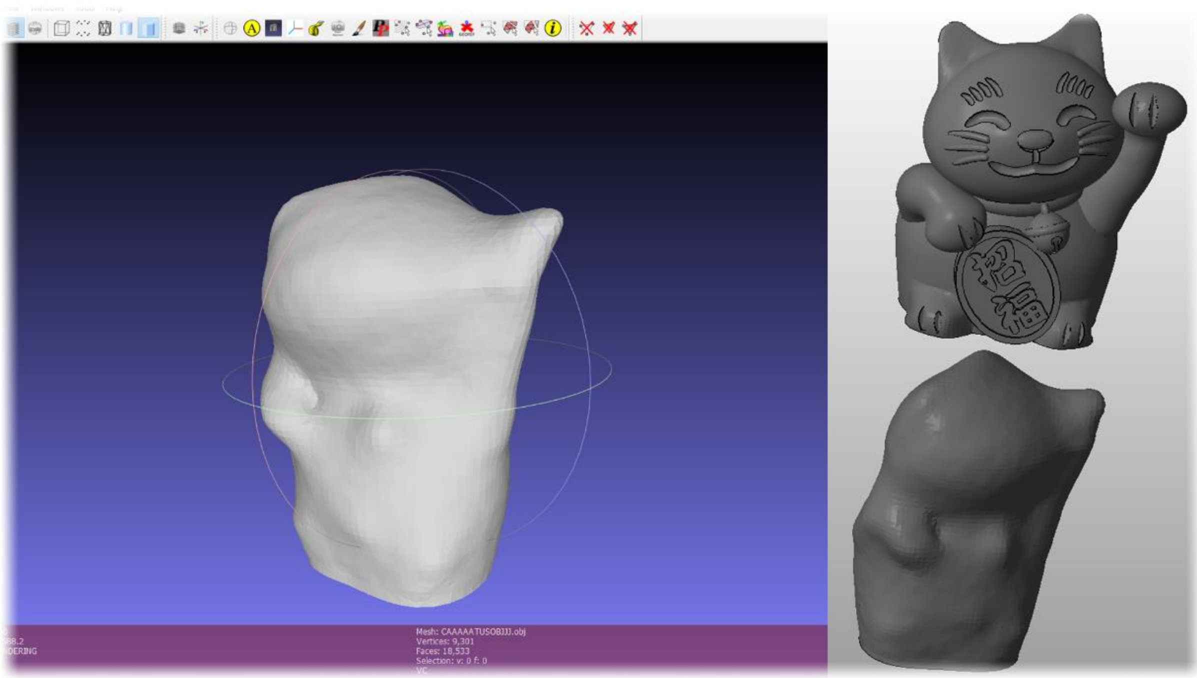


Figure 1110|Using MESHLAB step by step 7