# More Robust Optical Coordinate Metrology Sensors for 3D Objects

Tuan Chu, Kevin Leung, Gabriel Pizarro
Faculty Advisor: Dr. Ahmad Barari

# Project Scope

- The objective of this project was to develop a modular device that would demonstrate the principles of 3D laser scanning and photogrammetry with the end-goal of combining all sensor to operate on a drone

- Should be able to scan an area approximately the size of a boardroom

# Background

Our project is divided into two teams for different sensors:

- Laser Scanning: 3D scanning method that captures the shape of an object using a laser to obtain measurements

- Photogrammetry: the process of obtaining information about physical object by means of recording, measuring, and interpreting data from 2D images

# Existing Solution


*Figure 1: Faro Quantum Faroarm [1]*


*Figure 2: MakerBot Digitizer [2]*

# Laser Scanning - Basics

- X-Axis and Y-Axis data can be obtained in 2D
- Using the camera’s field of view (FOV), as a known parameter
- The X-Axis and Y-Axis data are collected based on the laser on camera’s FOV

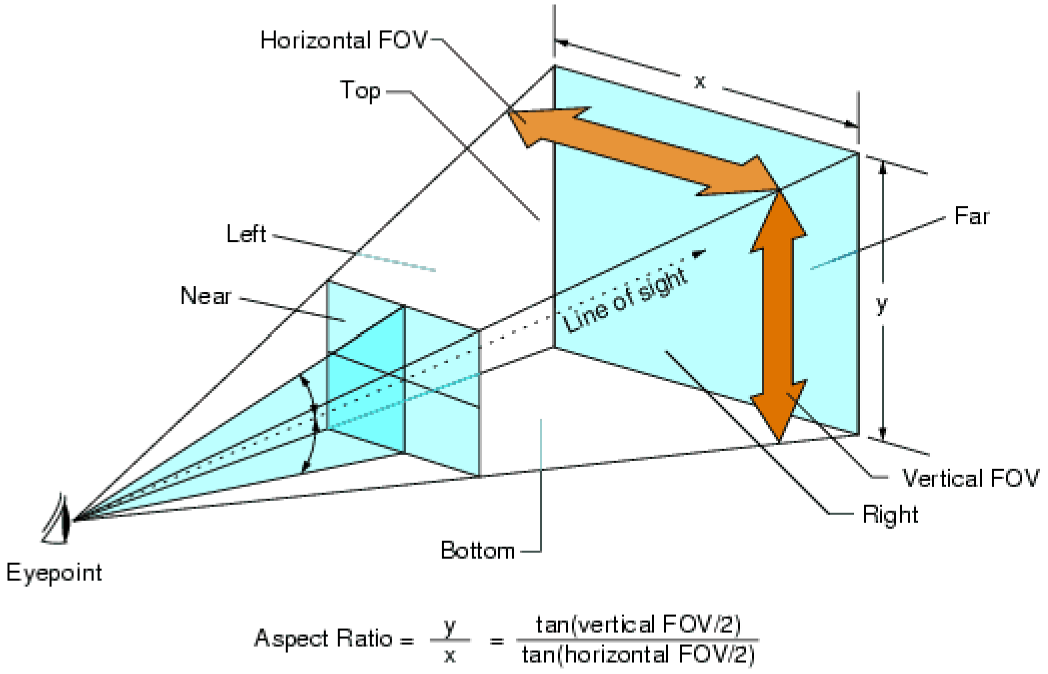


*Figure 3: Demonstration of FOV of camera [3]*

# Laser Scanning - Depth Estimation

- Z-axis data can be obtained through the depth of the object.
- Based on 3D triangulation set up to create an angle between the laser and camera
- Then using the known distance and angle between the laser and cameras, trigonometric ratios can be used to find the distance from the laser to the object

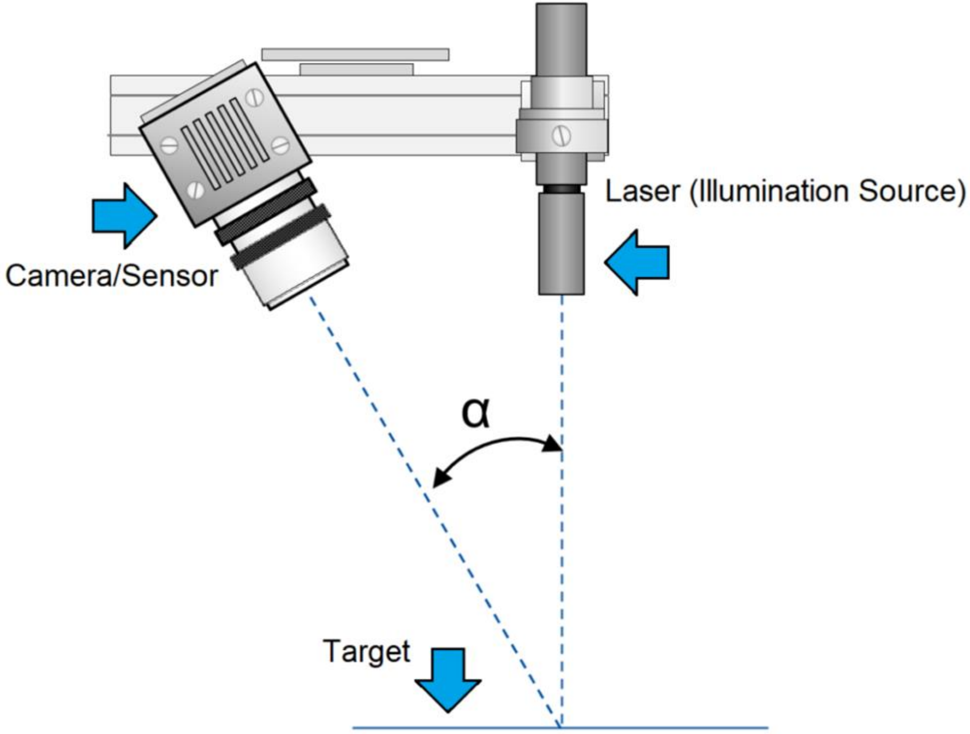


*Figure 4: Laser triangulation set-up [4]*

# Our Process

- Camera constantly recording video
- Inertial Measurement Unit (IMU) constantly tracking position and orientation
- MATLAB imports one frame
- Extracts point cloud slice
- Combines with IMU data
- Produces point cloud

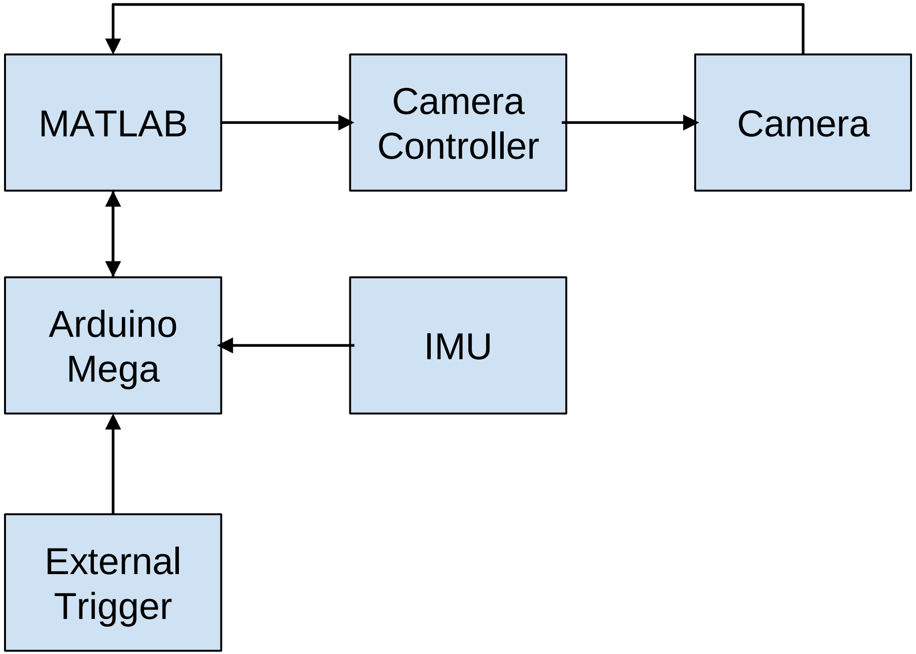


*Figure 5: Overall System Design*

# Image Processing 1 - Image Acquisition

- MATLAB instructs camera control script to record one frame
- Imports into MATLAB

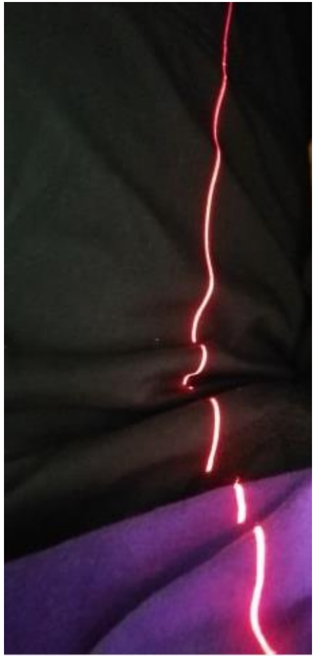

*Figure 6: Imported Test Image*

# Image Processing 2 - Distortion Correction

- Camera has a fisheye lens, causing distortion
- Determine camera intrinsic parameters
- Use to un-distort images

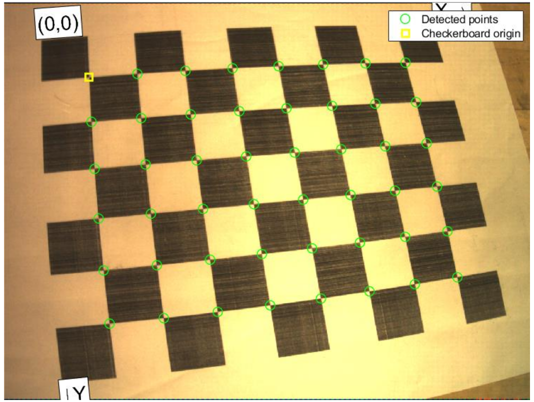


*Figure 7: Distorted Image*

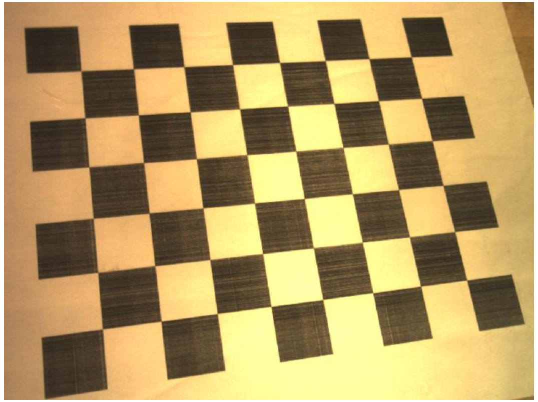

*Figure 8: Undistorted Image*

# Image Processing 3 - Noise Reduction

- Increase contrast of image
- Isolate red channel
- Convert from RGB to grayscale
- Define laser light boundary
- Find middle pixel per row

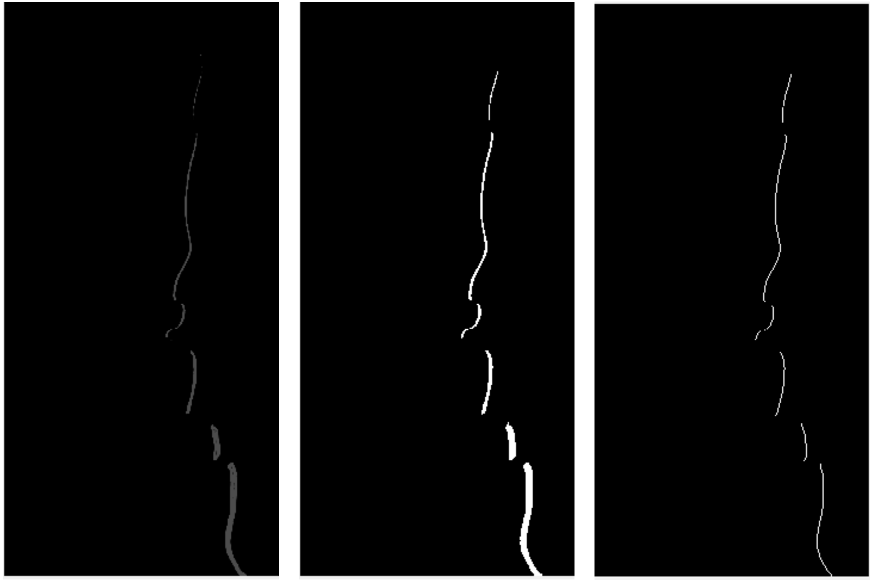

*Figure 9: Noise Reduction*

# Image Processing 4 - Depth Estimation

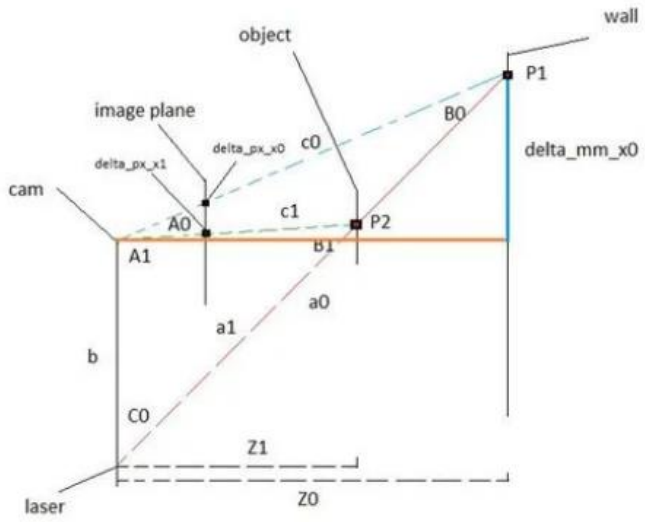


*Figure 10: Depth Estimation [5]*

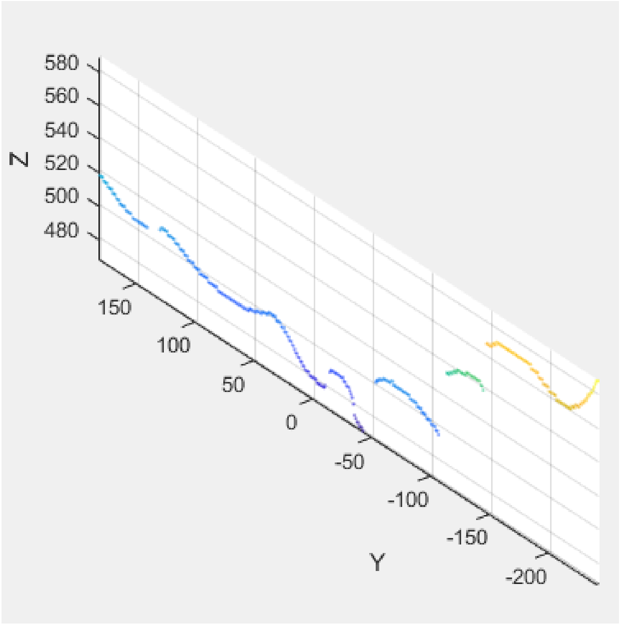


*Figure 11: Point Cloud Slice*

# Electrical Hardware

- Inertial Measurement Unit (IMU) selected is the LSM6DS0 from STMicroelectronics featuring 3-axis accelerometer and 3-axis gyroscope.
- All schematic, and Printed Circuit Board (PCB) designs are original from the capstone team.
- Software used was Autodesk Eagle.
- Main communication method used for the accelerometer and gyroscope data is I2C communication.

# LSM6DS0 IMU Testing Board - Schematic

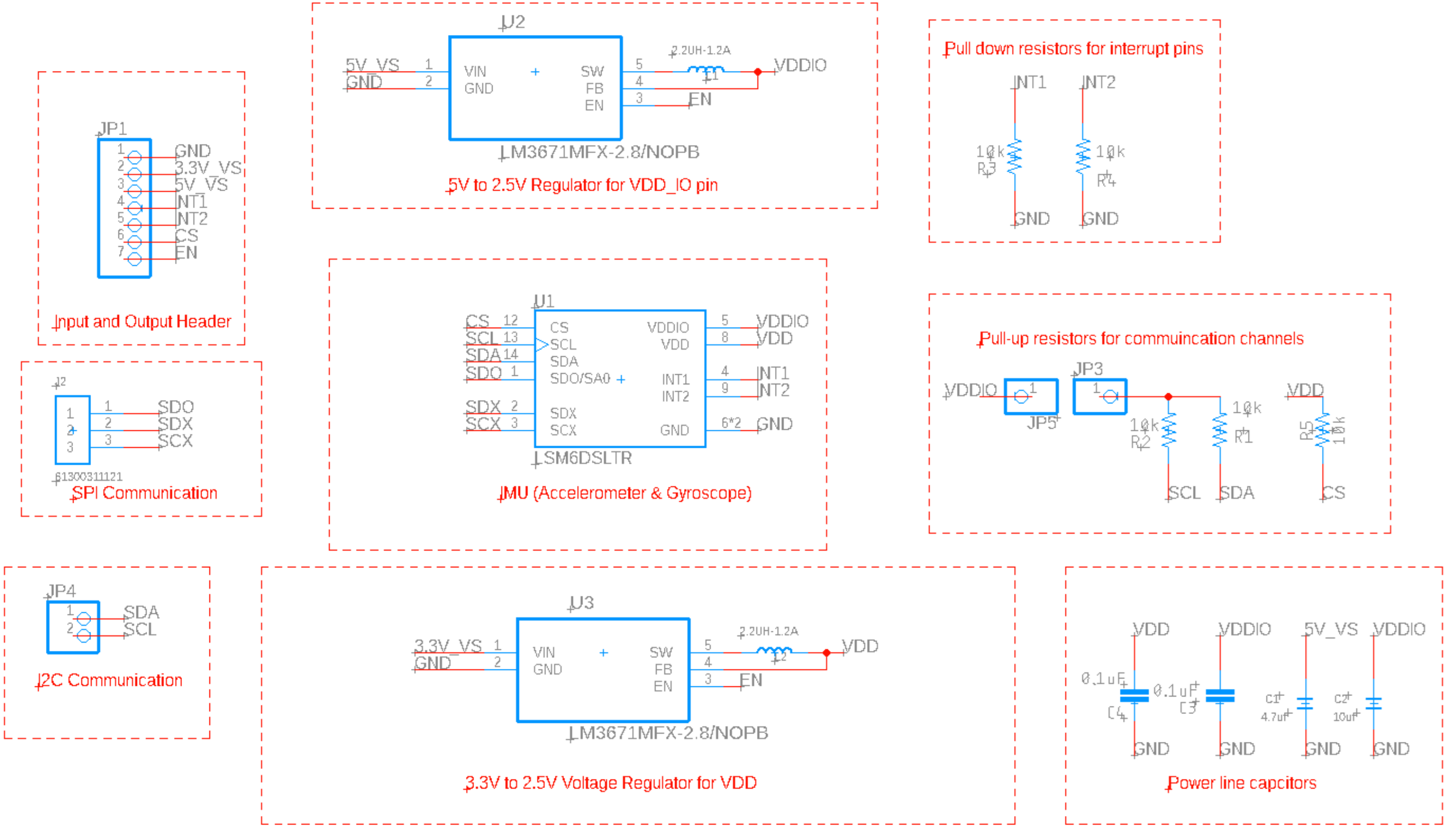


*Figure 12: Testing board schematic*

# LSM6DS0 IMU Testing Board - PCB Layout

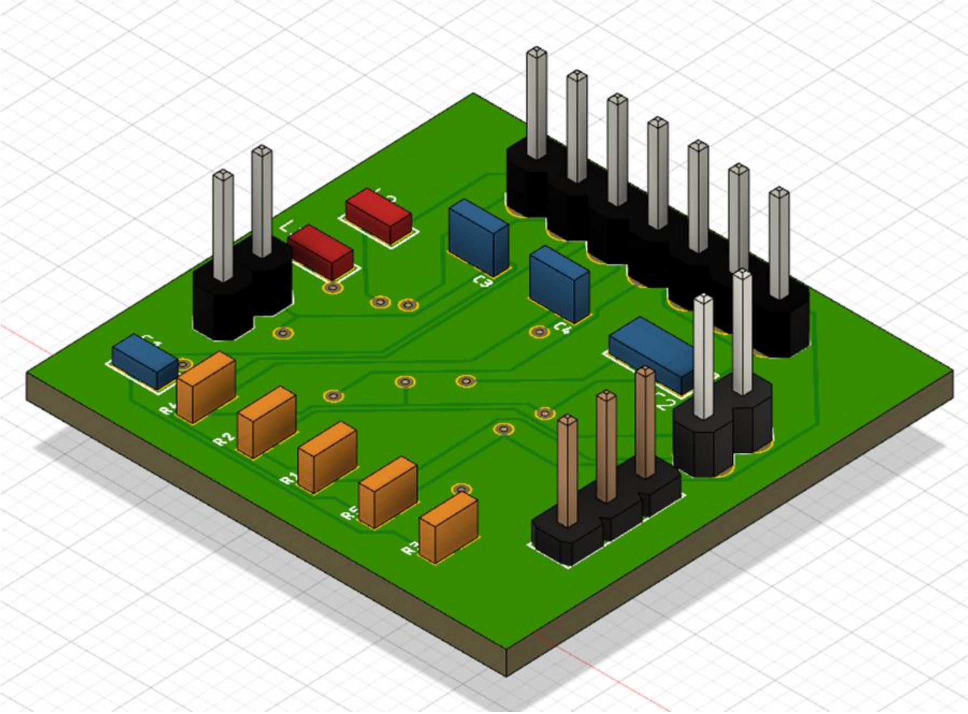

*Figure 13: Testing board render*

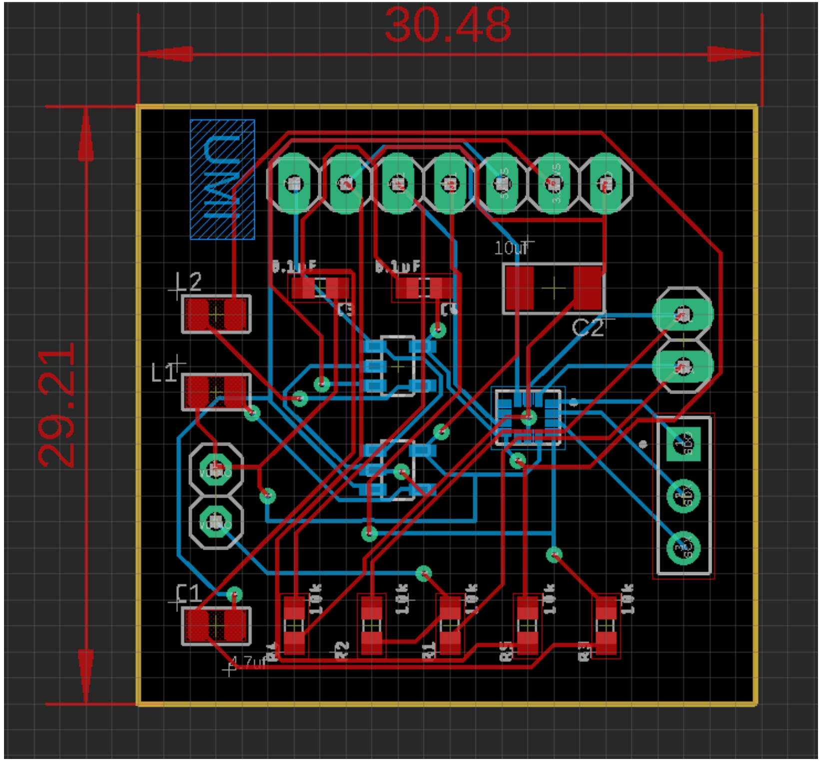


*Figure 14: Testing board layout*

# IMU Testing Board

- The inertial measurement unit (IMU) testing board was design with the ability of cascading for a multiple IMU setup. This ability is structured with the removal connection to the pull-up resistors for the communication lines. If you connect two or more IMUs together only one will need the pair of pull-up resistors. Our design uses a two-connection removal jumper.
- We use I2C communication but this board also has the ability to support SPI Comm.

# IMU Main Board

- For our capstone we had the idea of designing and manufacturing a board that integrated 4 specific features such as: 5 IMU setup for improved accuracy, fault-switching protocol to detect and disable IMUs that present a fault at real-time, modular design for easy plug-and-play system between the hand-held device and drone, and a level of memory storage for scanning data.
- Due to situations outside of our control this could not be accomplished.

# IMU Breakout Board

- Using the MPU-9250 9DOF IMU for development of the Matlab script and arduino code to process the accelerometer, gyroscope, and magnetometer data to get real-time position in 3D space.
- This IMU unlike the one in the testing board has a magnetometer that helps us correct for drift.
- 3D orientation visualization was attempted but did not had nearly the accuracy needed for integration.

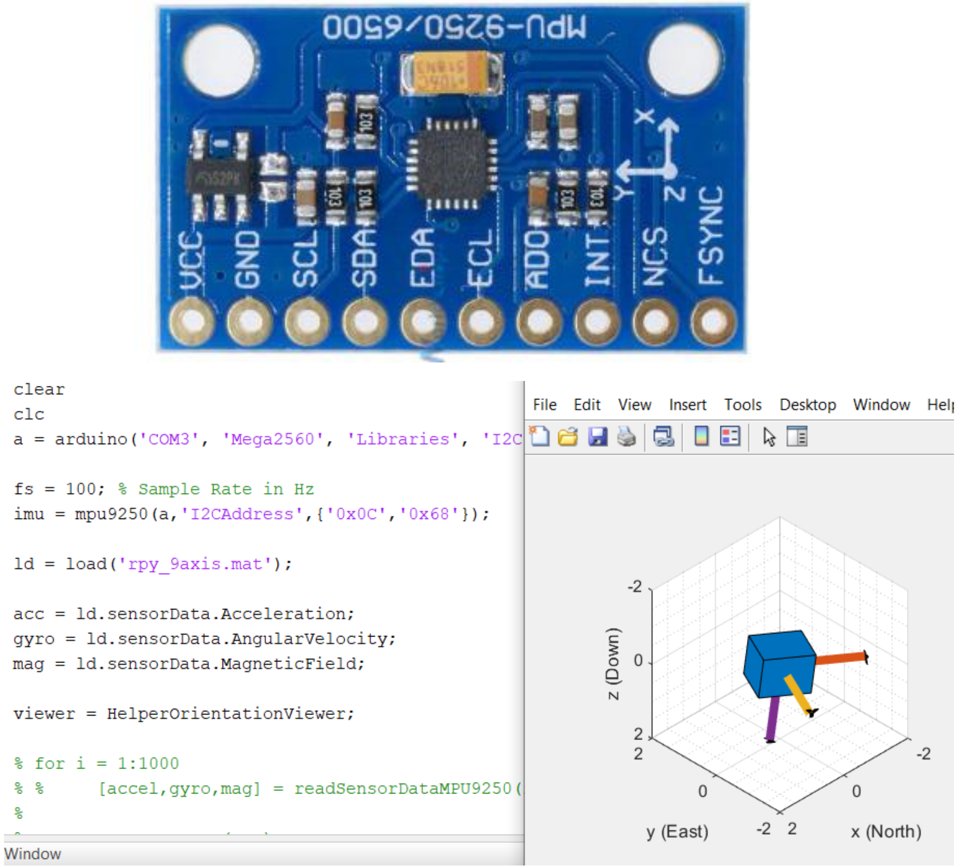


*Figure 15: IMU Breakout Board Testing*

# Integration IMU Data

- To create full point cloud data, IMU data must be incorporated to each point cloud slice through translational and rotational transformation.
- Translation transformation are based on the IMU data change in position, to calculated the displacement of the new slice point from previous point
- Rotation transformation required the IMU data to measure change in angle of X, Y, Z-axis

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ t_x & t_y & t_z & 1 \end{bmatrix}$$

*Figure 16: 3D translation [6]*

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(a) & \sin(a) & 0 \\ 0 & -\sin(a) & \cos(a) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

*Figure 17: 3D rotation at X-axis [6]*

$$\begin{bmatrix} \cos(a) & 0 & -\sin(a) & 0 \\ 0 & 1 & 0 & 0 \\ \sin(a) & 0 & \cos(a) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

*Figure 18: 3D rotation at Y-axis [6]*

# Integration IMU Data

$$\begin{bmatrix} \cos(a) & \sin(a) & 0 & 0 \\ -sin(a) & \cos(a) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

*Figure 19: 3D rotation at Z-axis [6]*

```
% Rotation along X-axis, angle = ax
x = x;
y = y.*cos(ax) - z.*sin(ax);
z = y.*sin(ax) + z.*cos(ax);
% Rotation along Y-axis, angle = ay
x = x.*cos(ay) + z.*sin(ay);
y = y;
z = z.*cos(ay) - x.*sin(ay);
% Rotation along Z-axis, angle = az
x = x.*cos(az) - y.*sin(az);
y = x.*sin(az) + y.*cos(az);
z = z;
```

*Figure 20: Calculation of coordinate at different rotation axis*

# Integration IMU Data (Translation)

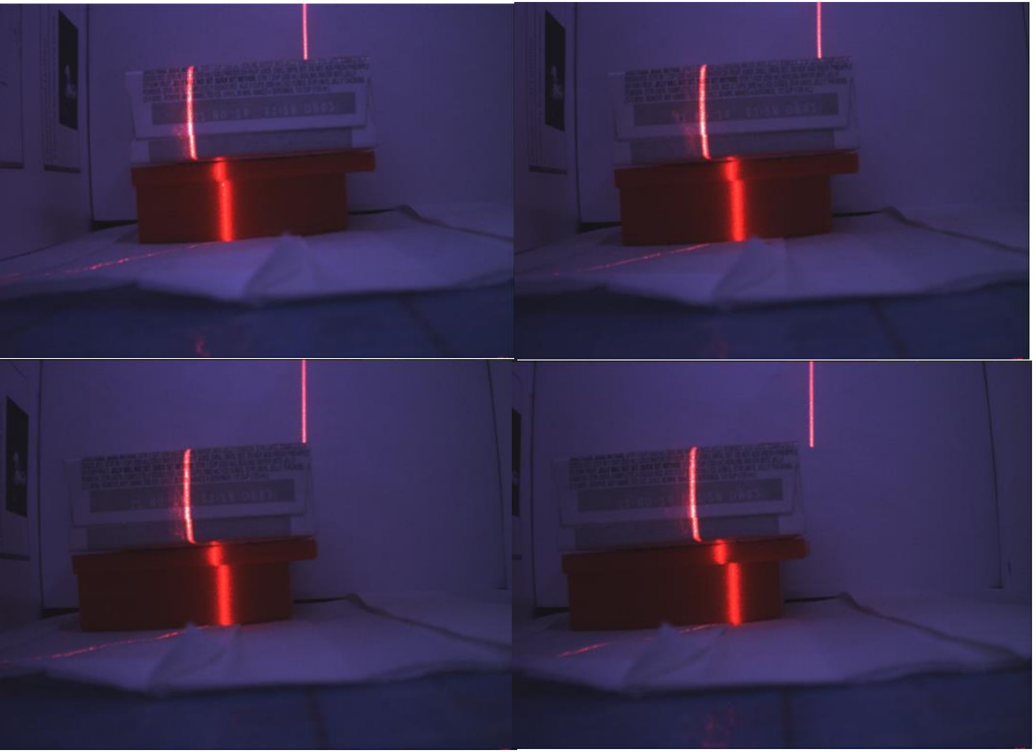

Figure 21: Laser scan of object at different distance

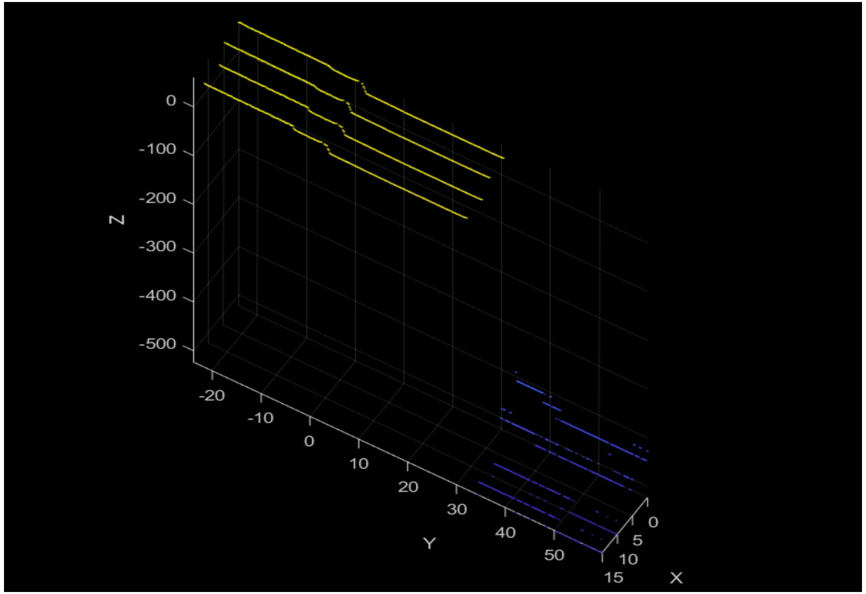


Figure 22:  Laser scan of object at translation transformation

# Integration IMU Data (Rotation)

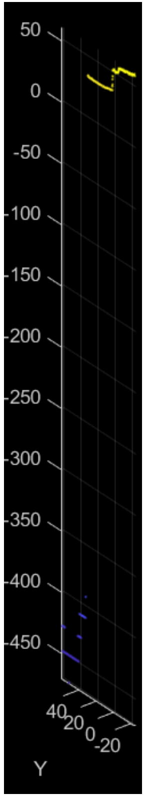


Figure 23: Original position

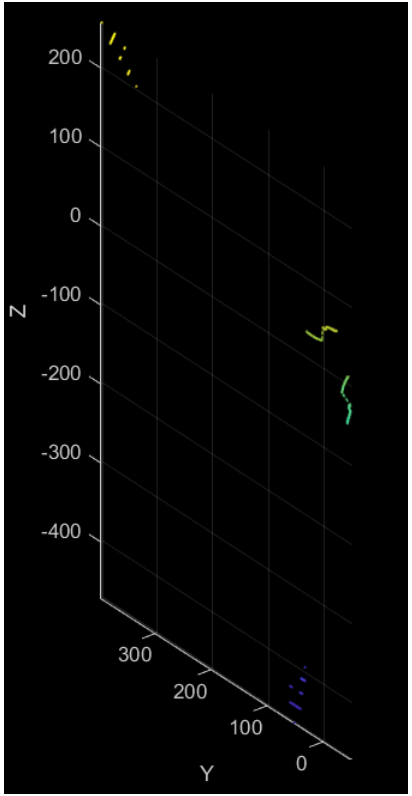


Figure 24: Original position and rotation at X-axis

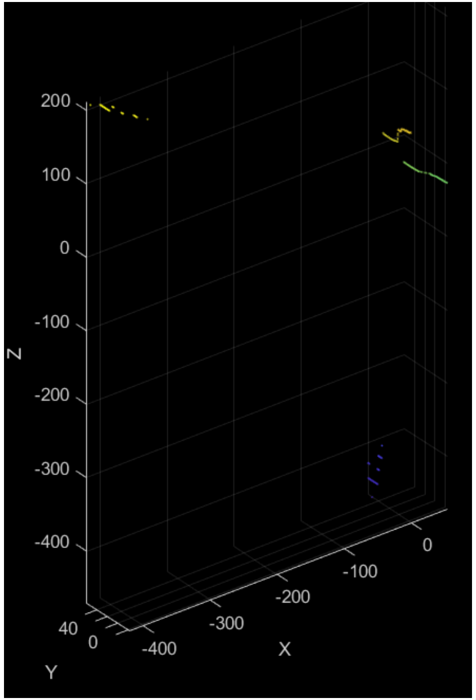


Figure 25: Original position and rotation at Y-axis

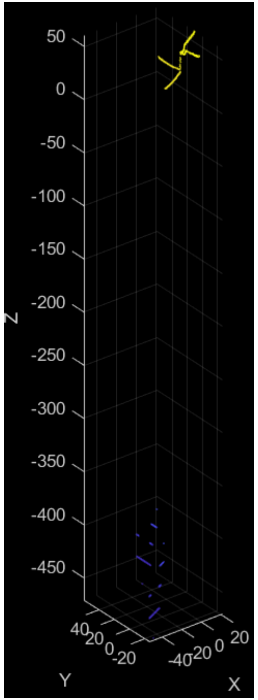


Figure 26: Original position and rotation at Z-axis

# Simulated IMU Environment

- Missing hardware due to shipping delays
- Inaccurate IMU position
- Accurate IMU Orientation

Solution:

- Missing axis can be simulated
- Assume unit is moving at a constant velocity

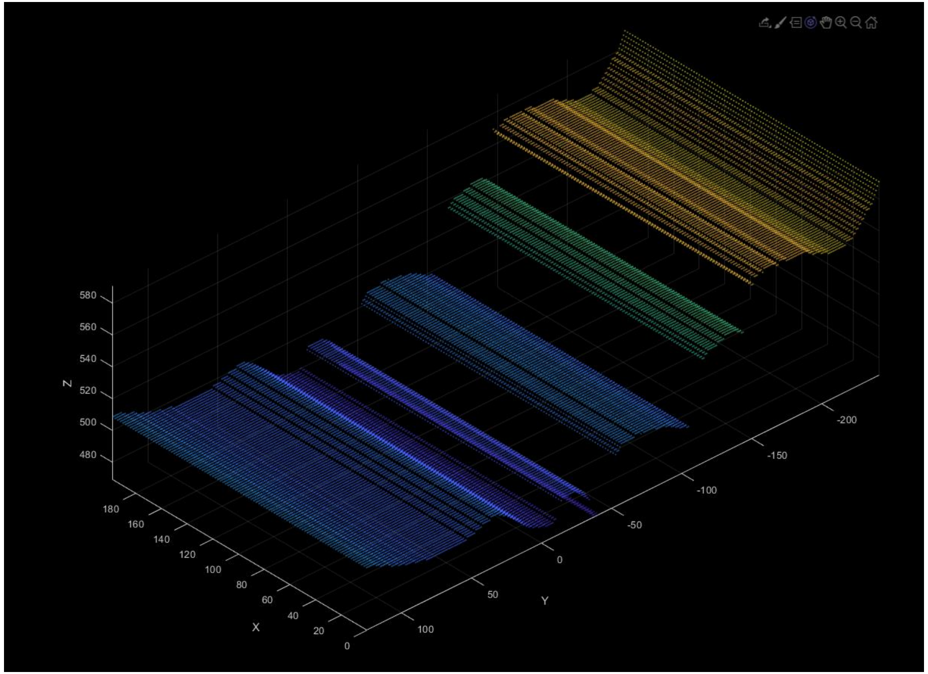


*Figure 27: Simulated IMU Integration*

# Results

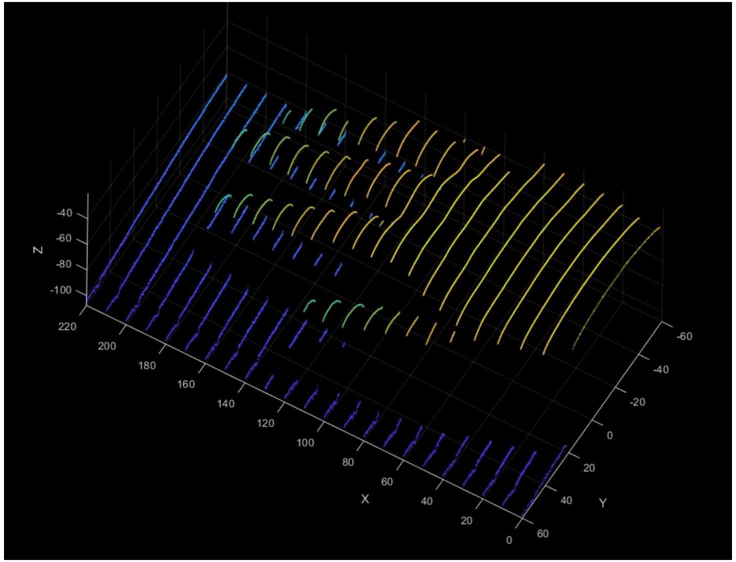


*Figure 28: Full Point Cloud*

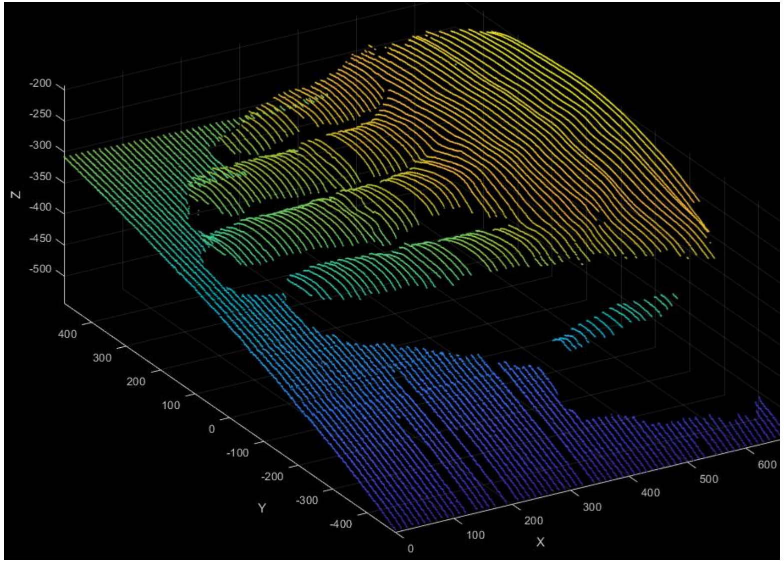


*Figure 29: Full Point Cloud, 2x IMU Sim Speed*

# Handheld Device

- System Design to hold laser diode at consistent angle and distance from camera
- Store all electrical components within the unit
- Designed to switch between handheld to drone attachment

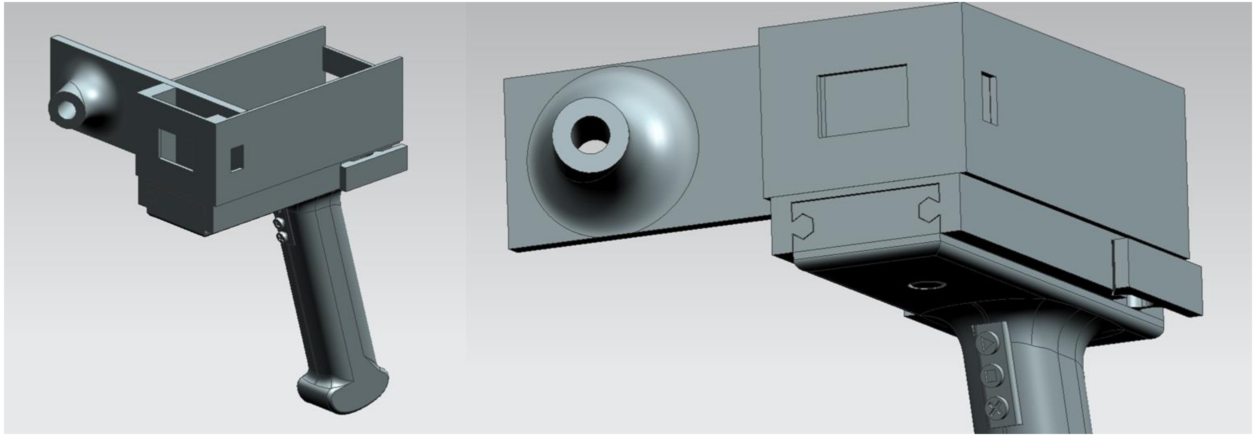

*Figure 30: Handheld Device*

# References

[1]2020. [Online]. Available: https://www.faro.com/. [Accessed: 31- Mar- 2020].

[2]"MakerBot Digitizer Desktop 3D Scanner | Order Today! | MakerBot", *MakerBot*, 2020. [Online]. Available: https://www.makerbot.com/stories/news/makerbot-digitizer-desktop-3d-scanner-order-today/. [Accessed: 31- Mar- 2020].

[3]J. Atwood, "Widescreen and FOV", *Coding Horror*, 2020. [Online]. Available: https://blog.codinghorror.com/widescreen-and-fov/. [Accessed: 31- Mar- 2020].

[4]2020. [Online]. Available: https://www.edge3d.io/what-3d-scanning-types-applications-advantages-and-methods-laser-lidar-contact-based-photogrammetry-triangulation-structured-pulse-light/. [Accessed: 31- Mar- 2020].

[5] "DIY: How to Make a Low Cost 3D Scanner", *Instructables*, 2019. [Online]. Available: https://www.instructables.com/id/3D-Laser-Scanning-DIY/?fbclid=IwAR25XCbzf-pwCbKNXxFtuzTl72uhfZGlw3hvVH3QJ8vU9LavszUAXPxQ29c. [Accessed: 25- Oct- 2019].

[6]"Matrix Representation of Geometric Transformations- MATLAB & Simulink- MathWorks United Kingdom", *Uk.mathworks.com*, 2020. [Online]. Available: https://uk.mathworks.com/help/images/matrix-representation-of-geometric-transformations.html. [Accessed: 31- Mar- 2020].