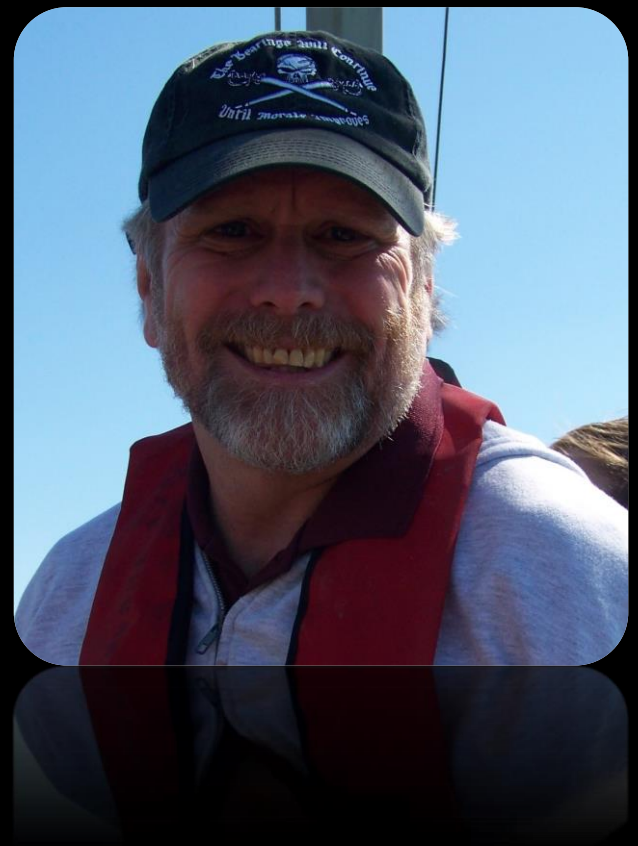
TEC-V

MILESTONE 2

By: Michael Dowling & Zealand Brennan

CLIENT

- DR. Wood
  - **Professor** | Ocean Engineering and Marine Sciences
  - **Program Chair for Ocean Engineering**

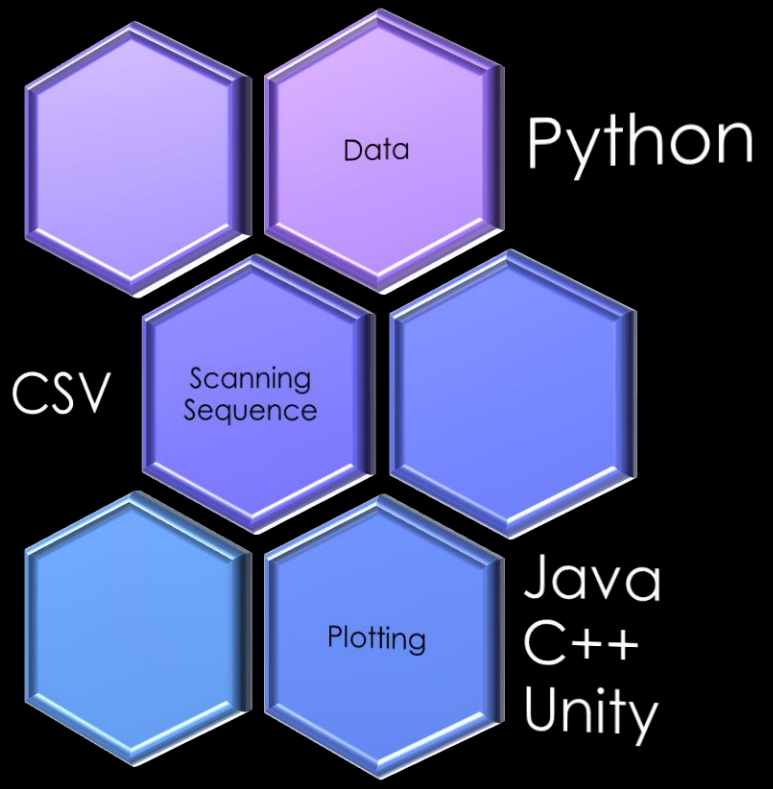


MILESTONE 1 OVERVIEW

- Sonar Data Retrieval
- Information Saving
- Testing
- Data Interpretation
- Point Cloud Plotting

TOOLS

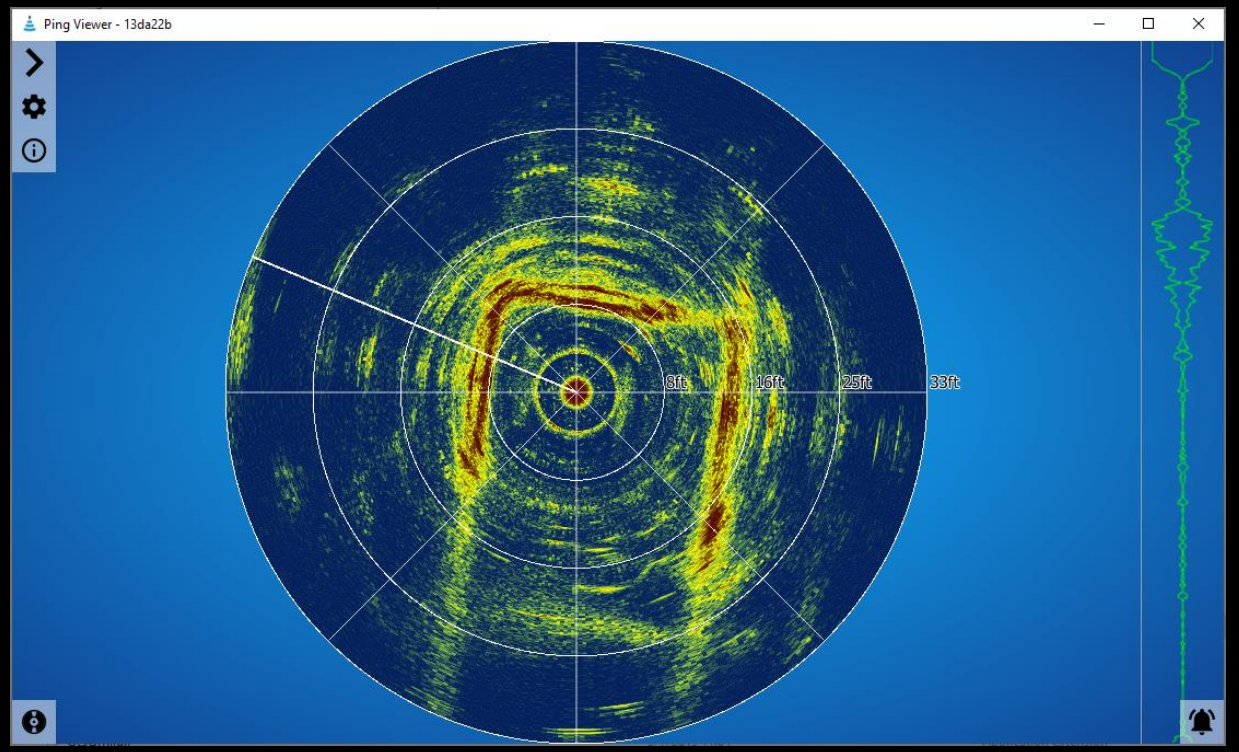
- Data: Python
  - Git Hub package that allows for simple commands
- Plotting: Unity / C++
  - Allows for better data manipulation in 3D environment



SONAR DEVICE

Ping 360

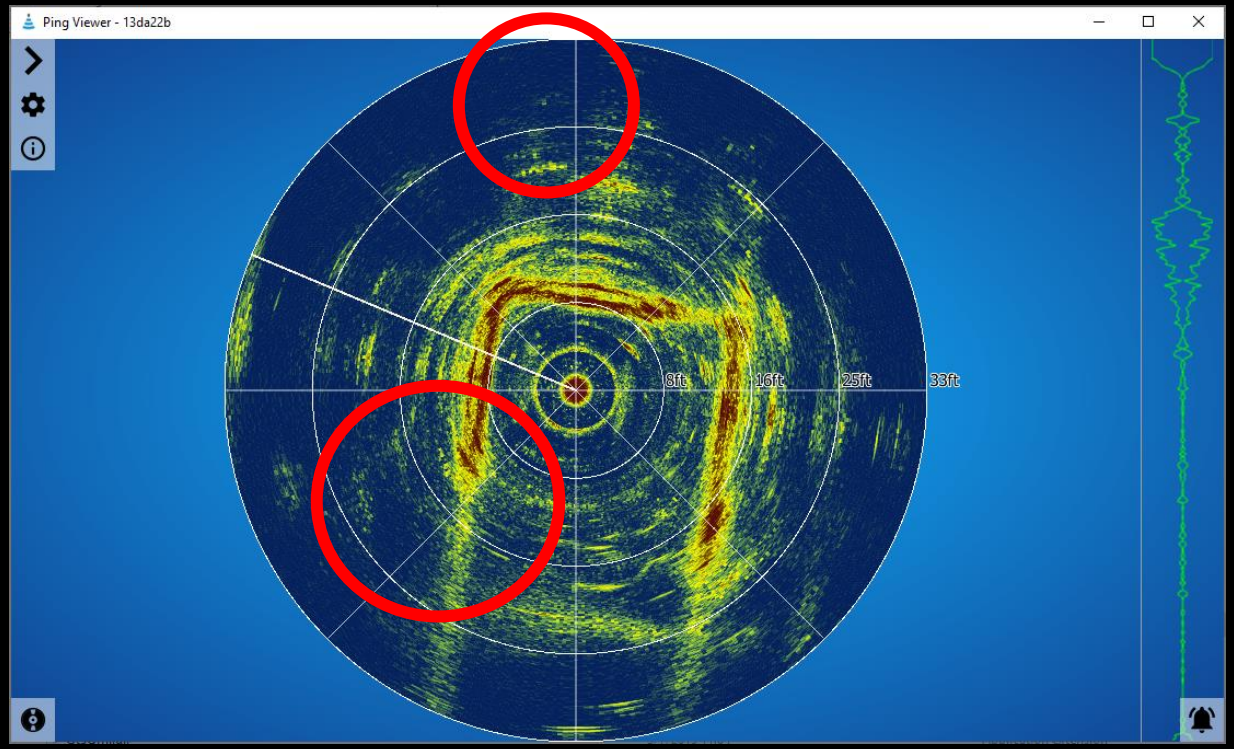
- Data is read by degrees and intensity values.
- Main view of what data you may come across



DETAILS

Main Problems

- Data can be missed by the sonar
- Reflections can cause higher intensity values than actual readings
- .8 meters from center is not viable



DATA RETRIEVED

Message

- Loop that asks for the intensity values at x degree to be sent back
- Intensity values 0-255
- Range 1200 in array

EXAMPLE: Intensity array

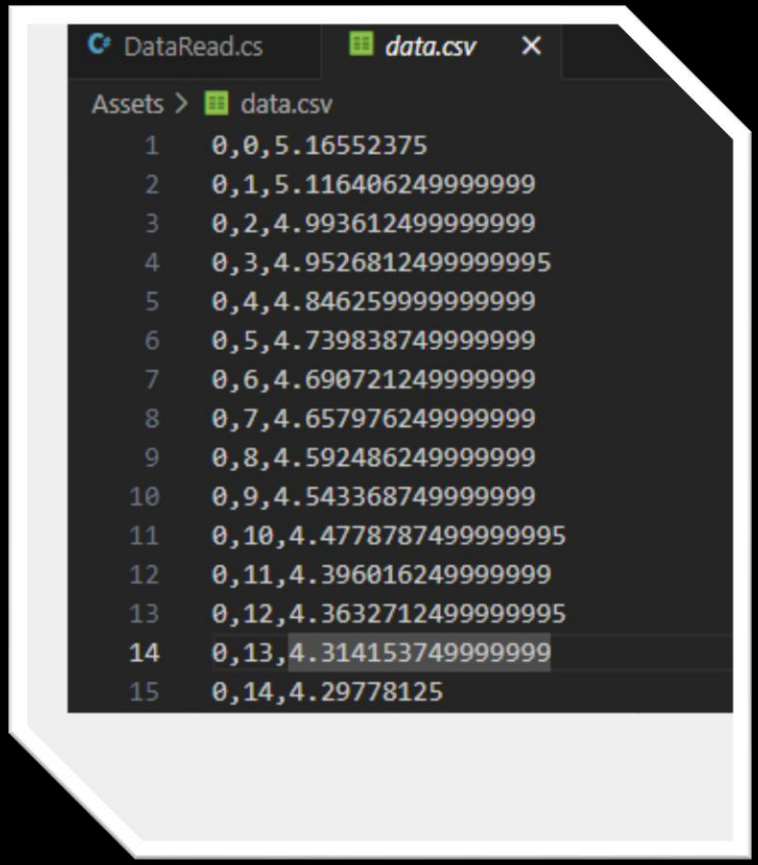
[0, 4, 134, 55, 20, 100, 160, 255, 240…….]

```
79  for currentAngle in range(400):
80      # Read a single iteration of intensity data
81      ping_data = ping360.transmitAngle(currentAngle)
82
83      # Extracting intensity as integer values
84      intensity_data = [(struct.unpack('!H', int(data).to_bytes(2, byteorder='big'))[0], i)
85                        for i, data in enumerate(ping_data.msg_data)]
86
87      # Store the raw data for the current angle
        raw_data[currentAngle] = intensity_data
```

DATA SAVING

Data.csv

- Three categories
  - Depth (in progress)
  - Angle
  - Most likely distance to object

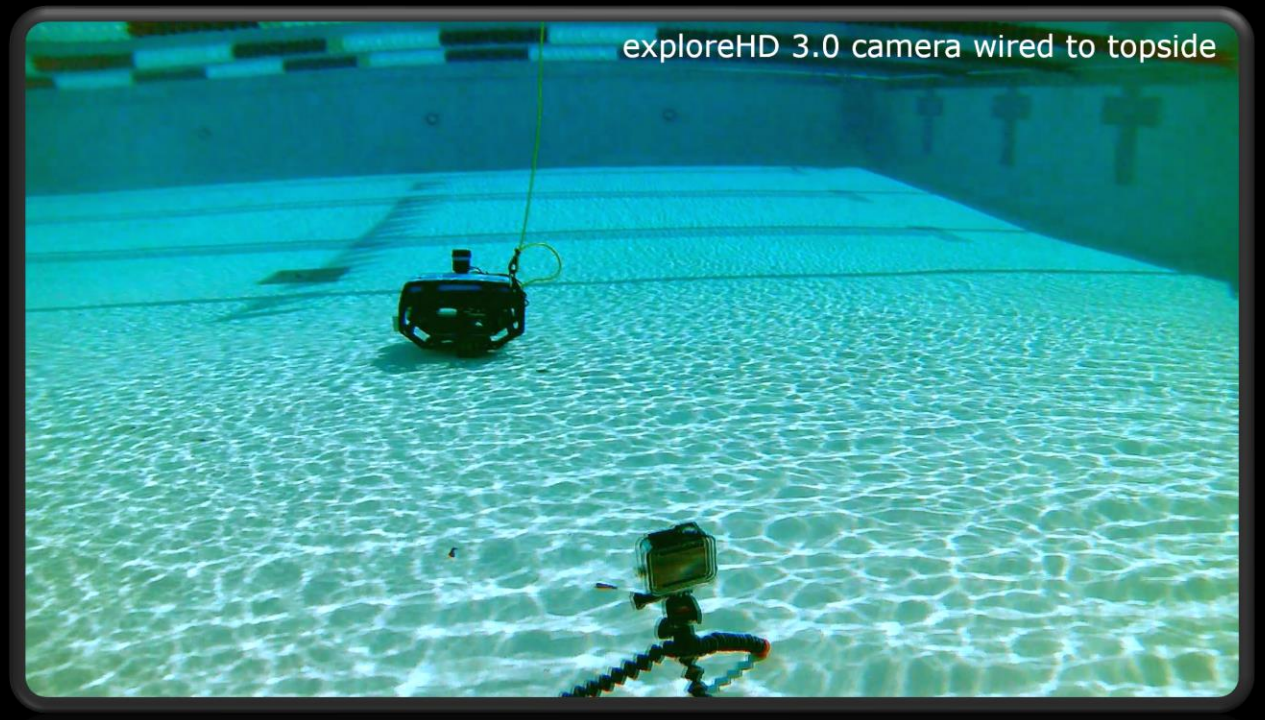


TESTING

10-21-23

- Clemente Pool 10 a.m. to 1 p.m.

- Goal:
  - Test sonar data retrieval
  - Collect Data for Cloud Plotting
  - Have a real-world test to see accuracy



SONARS ACCURACY

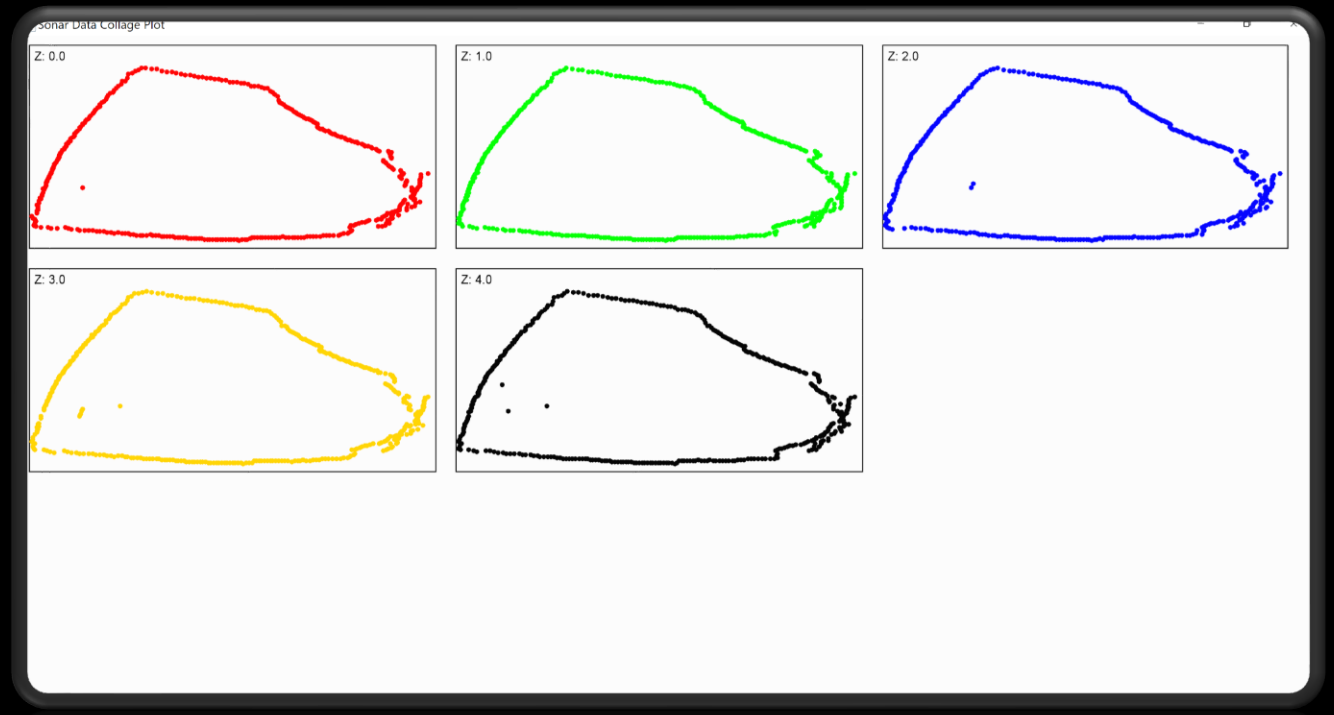
POOL TEST

Transcription

- Idea 1: Using Java
- Original Formula was incorrect

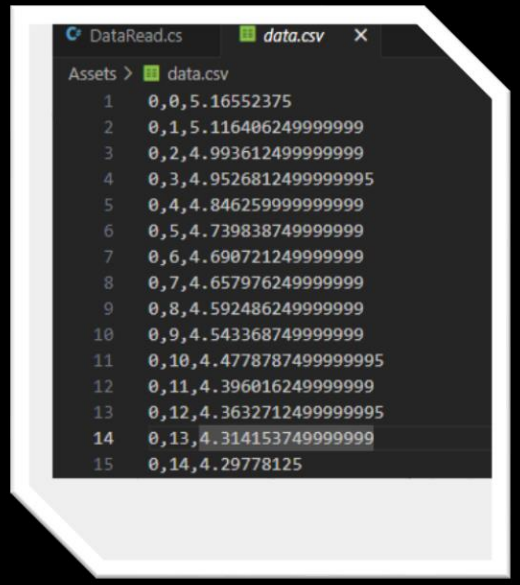
```
// Normalize the angle
double angleDegreesNormalized = angleDegrees % 360;
double angleRadians = Math.toRadians(angleDegreesNormalized);

double x = distance * Math.cos(angleRadians);
double y = distance * Math.sin(angleRadians);
```



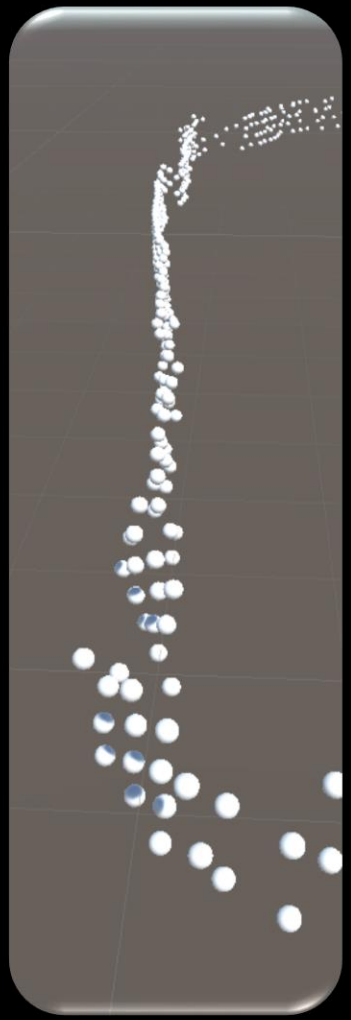
UNITY

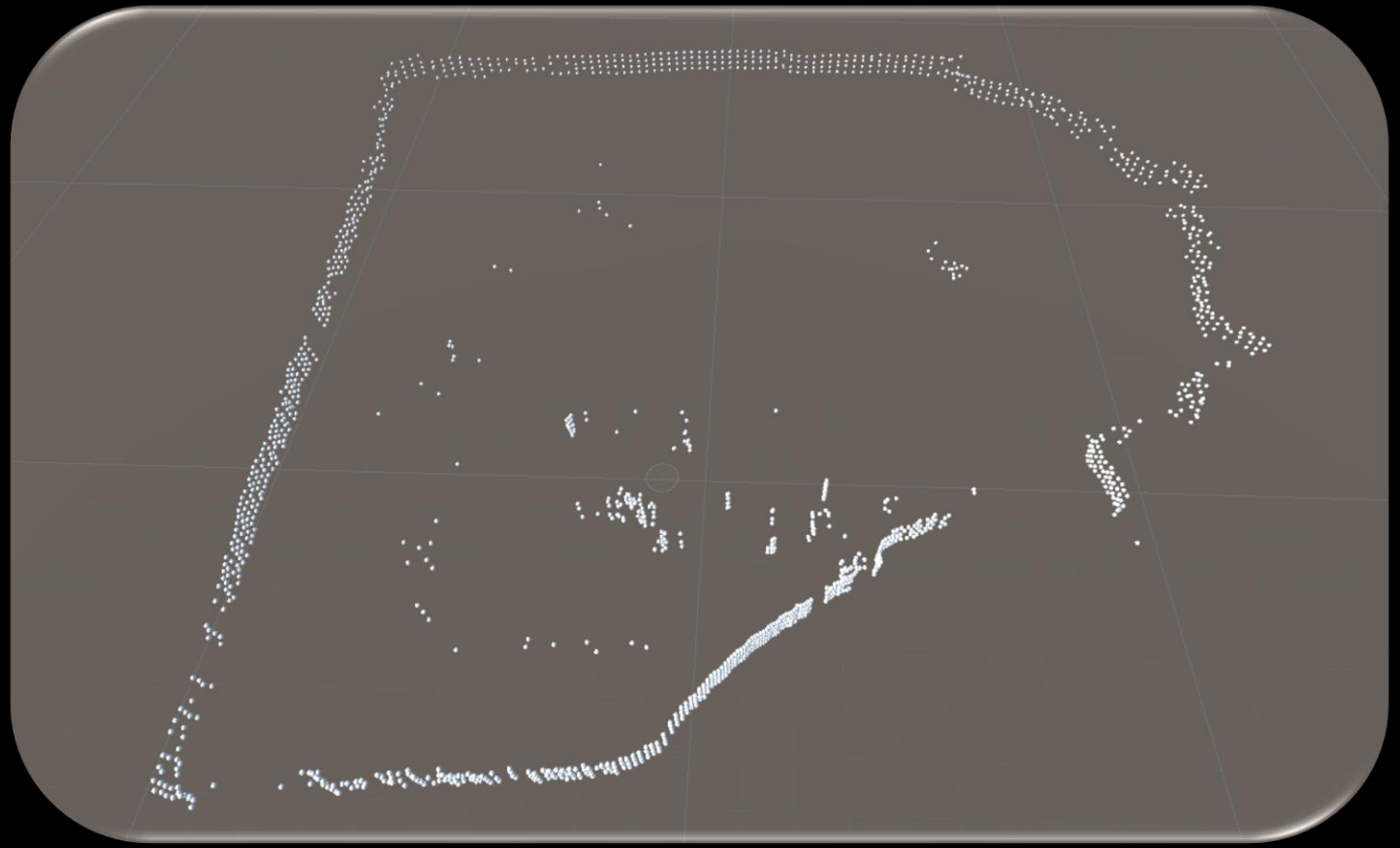
- Idea 2: Unity
  - Secondary formula corrected
  - Better data manipulation




```
while (!endoffile)
{
    string Data_String = stReader.ReadLine();
    if (Data_String == null)
    {
        endoffile = true;
        break;
    }
    var datavalues= Data_String.Split(',');
    Debug.Log(datavalues[0].ToString() +','+ datavalues[1].ToString() + ','+datavalues[2].ToString());
    Instantiate(sphere, new Vector3(Mathf.Cos(float.Parse(datavalues[1])*
                (Mathf.PI/200))*scale*-1* float.Parse(datavalues[2]),
                float.Parse(datavalues[0])*scale/5, Mathf.Sin(float.Parse(datavalues[1])
                * (Mathf.PI / 200)) * scale *float.Parse(datavalues[2])),
                new Quaternion(1, 1, 1,1));
}
```

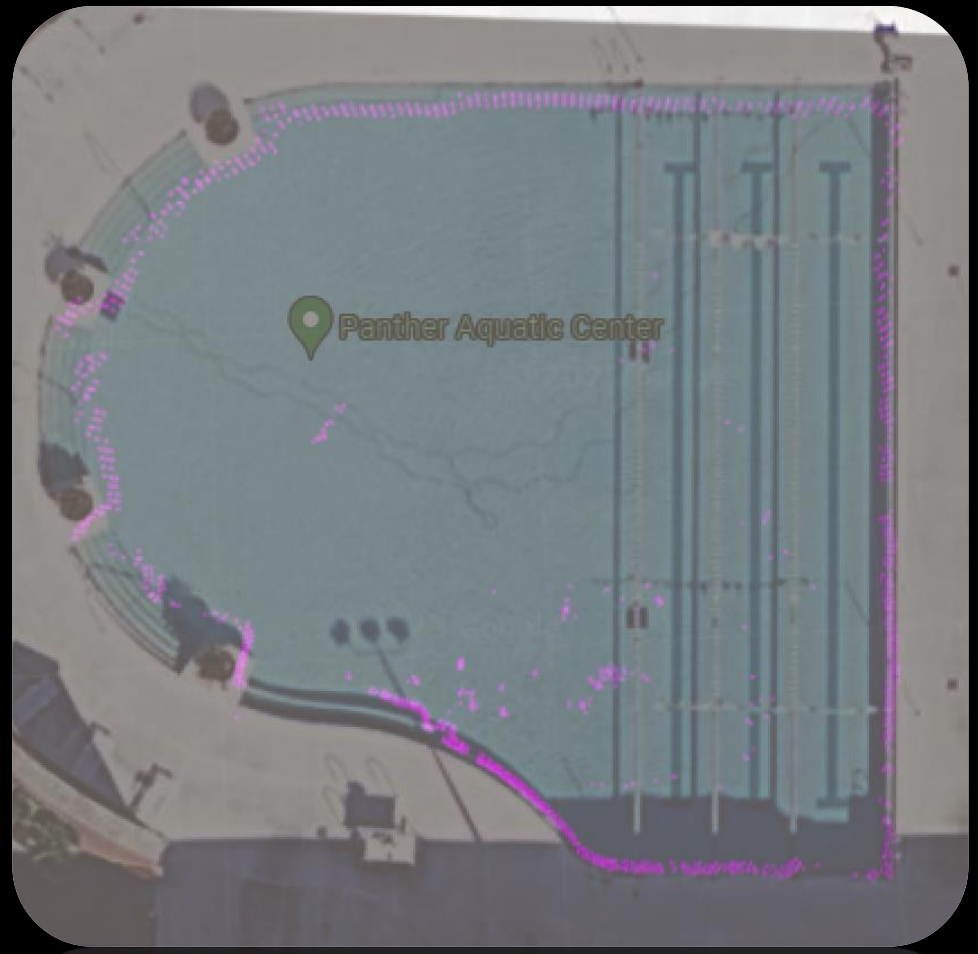
UNITY





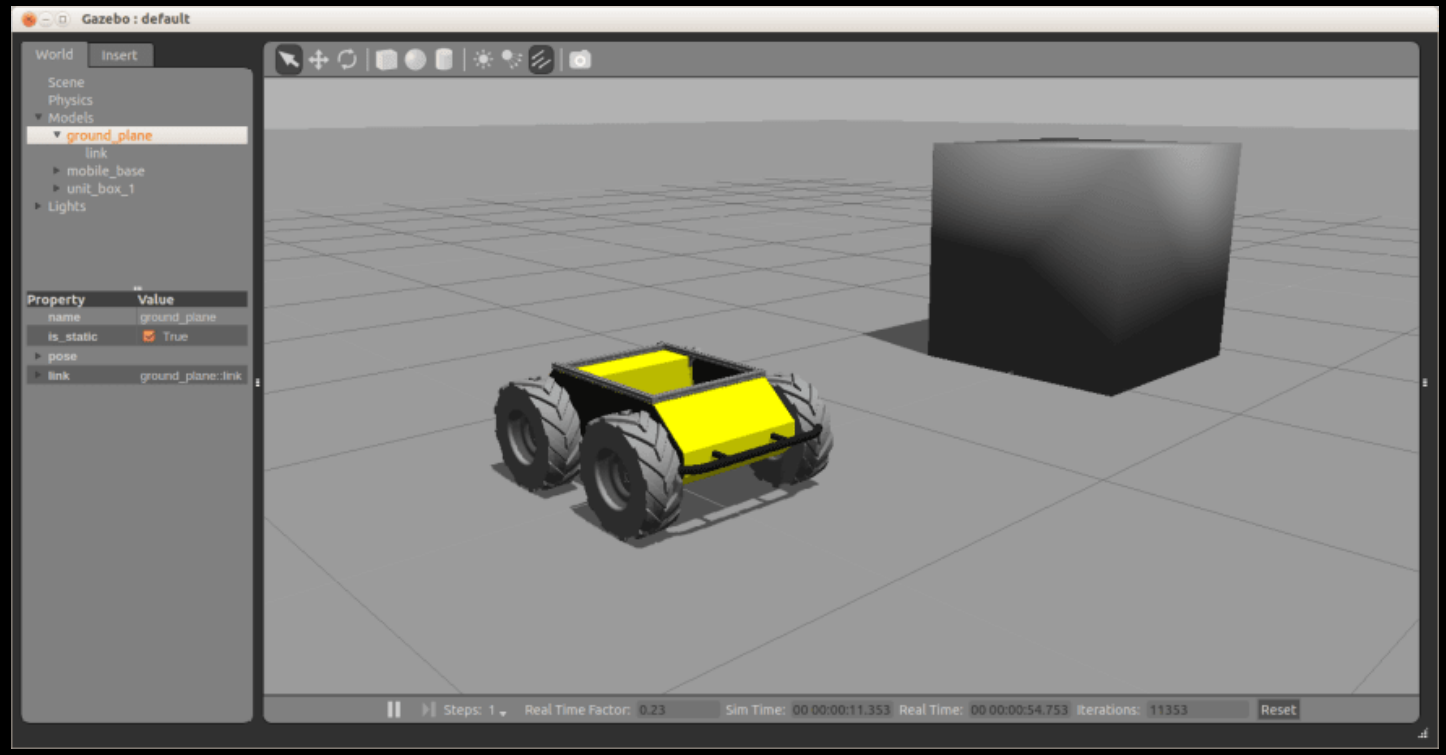
TRANSPOSE

- Data shows accuracy along flat edges
- Slight difficulty along the shallow end
- Shadows from where the sonar was unable to see



ADVISOR FEEDBACK

- Remove False data
- Use Gazebo to test AI pathing



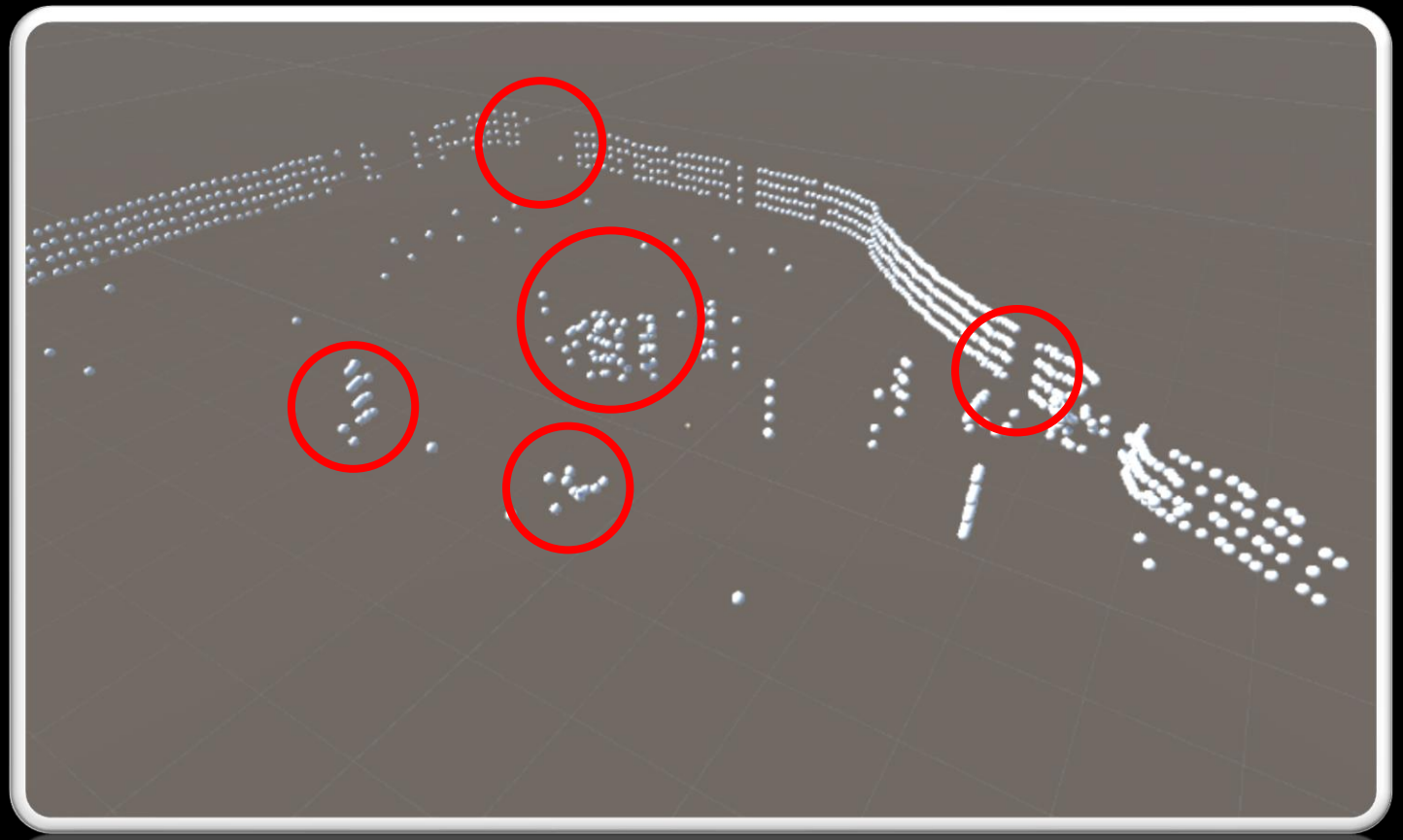
MILESTONE 3:

Task	Michael	Zealand
False Data	Create an algorithm to remove false data points / fill in the shadows within the data to create a cleaner image.	
Depth Finder	Identify the protocols to find and retrieve this data, may need to be done through Arduino. The goal for this is to have accurate measurements of the current depth.	
Compass and Telemetry	Identify the protocols to find, retrieve, and save the information. This is so that once we start rotating the AUV we can track the current heading to assist with data transcription.	
Cloud Plot Application		Work on creating an environment that will transpose the data and allow for Autonomous testing in a virtual environment.

MILESTONE 3: TASKS

1. ❑ **Improve False Data:**
   - ❑ Create a sorting algorithm to remove false data
2. ❑ **Telemetry Data:**
   - ❑ **Gain access to accurate depth and positioning instruments**
3. ❑ **Cloud Plotting / Testing**
   - ❑ **Use Gazebo to plot and test AI pathing**

FALSE DATA



Demo:

https://www.youtube.com/watch?v=VTigK4eMFWs

WEBPAGE LINK

TEC-V

https://bluecodehydra.github.io/FIT_Project-TEC_V/data.html

QUESTIONS?