Assignement: Playing with a scene point cloud

avataar

Rohit Jagannath Dhote **Email:** rohitdhote111@gmail.com

May 26, 2024

Overview: This report includes my approach toward the three tasks given in the assignment. I have explained my thought process while solving the tasks and added visual results, advantages, and limitations of the work.

---

1. Task 1: Re-orient the point cloud

To accomplish this task, I followed these steps:

1.1. Understanding the point cloud:

First, I analyzed the point cloud data (shoe pc) by visualizing it in open3D.

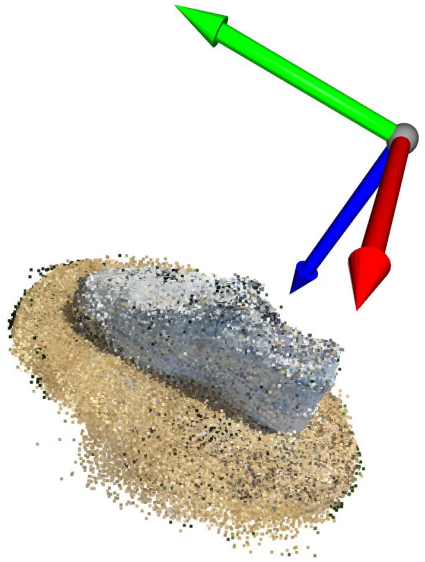


Figure 1: Initial PC and reference axes

1.2. Fitting a Plane:

Ground points in PC are spread on a plane so in order to detect them I need to use any plane detection algorithm. so I used the RANSAC plane fitting method from the open3d library.

1.3. Aligning with YZ Plane:

With the equation of the floor plane determined, I calculated the necessary rotation to align the floor plane with the YZ plane (where y = 0). This rotation ensured that the floor would lie flat on the YZ plane with its center at the origin.

1.4. Transforming the Point Cloud:

Finally, I applied the calculated rotation to the entire point cloud dataset. This transformation re-oriented the entire scene so that the floor was aligned as specified in the task, with the shoe and other objects adjusted accordingly.

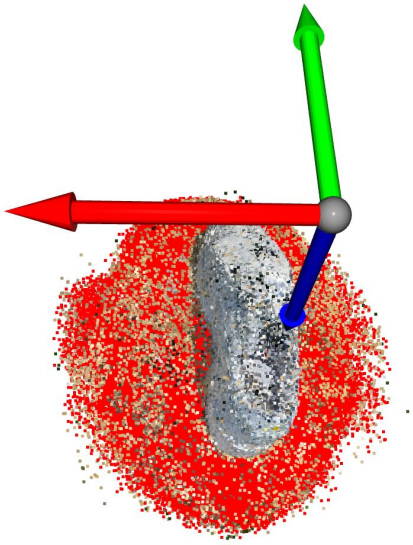


Figure 2: PC with plane fitted on ground points shown with red colour

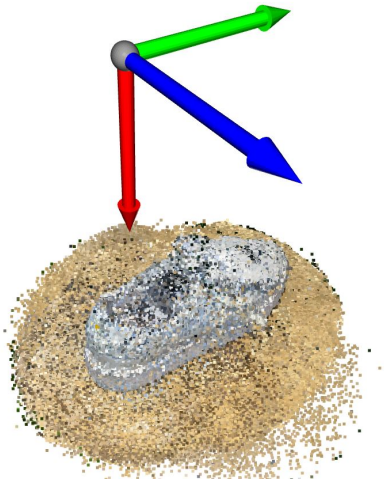


Figure 3: PC reoriented to align with the YZ plane

2. Task 2: Creating mesh from point cloud

2.1. Pre-processing point cloud

This involves removing noise points and removing the unwanted part of the point cloud. This is required in order to get the good-quality mesh. I have implemented SOR noise removal technique and also checked with various point density but didn't observe much improvement in mesh quality. So skipped this step in my final submission file.

2.2. Computing normals :

I checked for various mesh creation algorithms and found out that Poisson's method works well. Poisson's surface reconstruction requires normals for each point in the point cloud. Since the input point cloud only has XYZ and RGB data, you need to estimate the normals. This can be done using Open3D's estimate normals function.

2.3. Implementing Poisson's method

Algorithm parameters are selected after multiple trials in CloudCompare and best parameters are selected.

2.4. Post-processing of mesh to make more smoother

We can use Meshlab software for this task

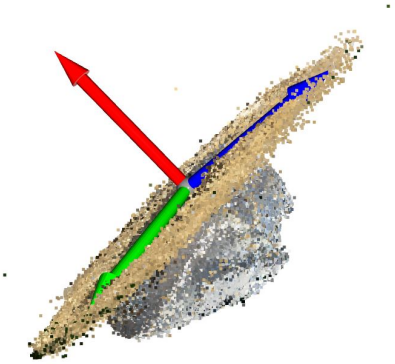
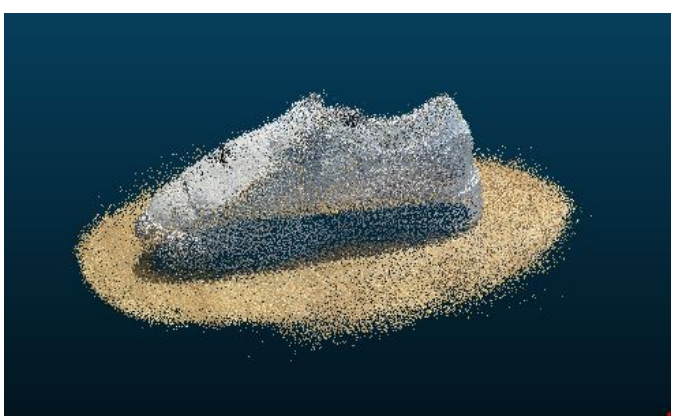
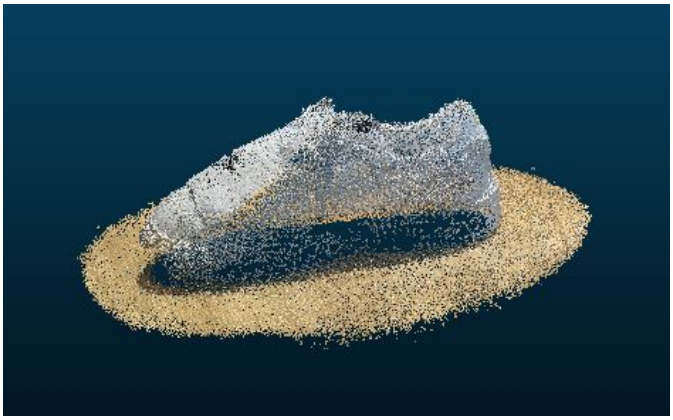


Figure 4: Reoriented PC with origin at the plane center and floor plane aligned with YZ plane



(a) Initial PC with noise points



(b) PC after application SOR

Figure 5: PC noise removal

- **Remove Duplicates and Isolated Vertices:** From Meshlab use filters like ”Remove Duplicated Faces” and ”Remove Duplicated Vertices”.
- **Noise Reduction and Smoothing:** In Meshlab, Apply ”Laplacian Smooth” under ”Filters” -¿ ”Smoothing, Fairing and Deformation”.
- **Mesh Decimation and Simplification** In Meshlab, Use ”Quadric Edge Collapse Decimation” under ”Filters” -¿ ”Remeshing, Simplification and Reconstruction”.
- **Hole Filling and Mesh Repair** In Meshlab, Use ”Close Holes” under ”Filters” -¿ ”Remeshing, Simplification and Reconstruction”.

3. Task 3: Unit testing for task 1

3.1. Apply transformations to input PC

Applied random rotation and translation transformation to input point cloud using open3d library

3.2. Check

The floor plane should be on the YZ plane, hence a should be close to 1, and b, c should be close to 0. where a, b and c are plane parameters.

4. Advantages and limitations of the work

4.1. Advantages for Task 1 approach

- This approach is unsupervised and doesn’t need any training data.

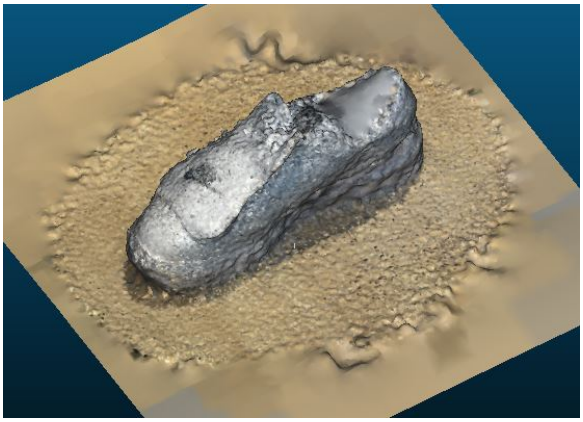


Figure 6: Generated mesh using Poisson's method

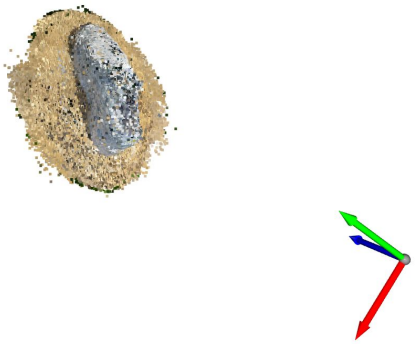


Figure 7: Transformed PC (rotation + translation)

- After re-orientation of the objects, it is easier to compare them.

4.2. Limitations for Task 1 approach

- If the object is placed on the ground in some inclined manner then the object will not get re-oriented properly.
- If the object is shaped like a box where multiple planes are present then RANSAC will detect a plane that may not be the ground plane and the entire process will fail.
- **Solution to this limitation** We need to set some elimination criteria for detected planes based on the plane orientation, plane size, plane shape, etc., and using these criteria we can eliminate the detected plane that does not belong to the ground.

4.3. Advantages of task 2 approach

- Noise Reduction: Pre-processing steps like noise removal improve the quality and accuracy of the resulting mesh.
- Customizable Output: Allows for post-processing and refinement in Meshlab, providing control over the final mesh quality.

4.4. Limitation of task 2 approach

- Parameter Sensitivity: Requires careful selection of algorithm parameters to balance between detail and smoothness, which can be time-consuming.
- Quality Dependence on Input Data: The quality of the resulting mesh is highly dependent on the initial point cloud's density and noise levels.
- **Idea** Only for visualization purposes as the mesh quality is not good we can check for 3D model rendering algorithms such as NeRF, etc..

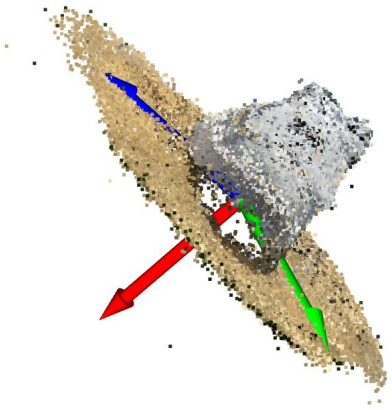


Figure 8: PC is oriented as per requirements

4.5. Advantages of task 3 approach

- Validation of Transformations: Ensures that the re-orientation task was performed correctly, verifying that the floor plane aligns with the YZ plane.

4.6. Limitations of task 3 approach

- Simplified Checks: Tests primarily for plane alignment and might not catch all potential issues in the point cloud re-orientation.
- Dependence on Accurate Initial Transformations: Effectiveness of the test is contingent on the initial random transformations being correctly applied, which adds an additional layer of complexity.