Data Extraction Module Docs

Overview

This module, primarily written in Python, contains methods to obtain measurements from a 3D point cloud. The measurements extracted from this module are wingspan, height, waist, and chest.

Prerequisites

Python 3.11

Open3D 0.18.0

NumPy

Pandas

Supported operating systems:

- Ubuntu 18.04+
- macOS 10.15+
- Windows 10+ (64-bit)

Before calling the methods in this module, point clouds **must be denoised to only be that of the individual person** to ensure accurate measurements.

Methods

dataExtract(pointcloud, threshold=0.001)	Extracts height, width, waist, and chest from pointcloud.
drawMeasurements(pointcloud, height_points, width_points)	Draws point cloud with positions of height points and width points to new window.

dataExtract(pointcloud, threshold=0.001)

Parameters:

- **pointcloud** : (open3d.geometry.PointCloud) – Must be denoised.
- **threshold** : (float) – How wide the area of points it grabs for chest and waist calculations.

Returns:

- **measurements** : (pandas.DataFrame)
- **height_points** : (list) – The top and bottom most point on the point cloud.
- **width_points** : (list) – The points from one hand to the other on the point cloud.

Example:

```
import open3d as o3d

# Used to denoise point cloud
from ModelGenFunctions import cut_floor
from ModelGenFunctions import cloud_denoise

from data_extraction import dataExtract

# Reads point cloud from location and assigns to pc
pc = o3d.io.read_point_cloud("point-cloud-example/brandon-5_8_slow.ply")

# Denoises point cloud
pc = cut_floor(pc)
pc = cloud_denoise(pc)

# Extracts measurements and the points it grabs
measurements, height_points, wingspan_points = dataExtract(pc)

measurements = measurements.loc[0]
height = measurements["height"]
wingspan = measurements["wingspan"]
waistCir = measurements["waist-cir"]
chestCir = measurements["chest-cir"]

print(
    f"Height: {height} units\n" +
    f"Wingspan: {wingspan} units\n" +
    f"Waist Circumference: {waistCir} units\n" +
    f"Chest Circumference: {chestCir} units\n"
    )
```

drawMeasurements(pointcloud, height_points, width_points)

Parameters:

- **pointcloud** : (open3d.geometry.PointCloud)
- **height_points** : (list) – The top and bottom most point on the point cloud.
- **width_points** : (list) – The points from one hand to the other on the point cloud.

Returns:

- **None**

Example:

```
import open3d as o3d

# Used to denoise point cloud
from ModelGenFunctions import cut_floor
from ModelGenFunctions import cloud_denoise

# Used to extract height and width points
from data_extraction import dataExtract

# Used to draw pointcloud with points in separate window
from data_extraction import drawMeasurements

# Reads point cloud from location and assigns to pc
pc = o3d.io.read_point_cloud("point-cloud-example/brandon-5_8_slow.ply")

# Denoises point cloud
pc = cut_floor(pc)
pc = cloud_denoise(pc)

# Extracts measurements and the points it grabs
measurements, height_points, width_points = dataExtract(pc)

drawMeasurements(pc, height_points, width_points)
```

Output: