DBSCAN.py

```python
import numpy as np
import open3d as o3d
import matplotlib.pyplot as plt

# ----------------------------------------------------------------------------
# 1. Load and combine six point cloud files
# ----------------------------------------------------------------------------
for i in range(1,7,1):
    file_name = ".\\dataset\\ny" + str(i) + ".xyz"
    point_cloud_object_temp = np.loadtxt(file_name, skiprows = 1,
                                usecols = (0,1,2,3))
    if i != 1:
        point_cloud_object = np.concatenate((point_cloud_object,
                                             point_cloud_object_temp), axis=0)
    else:
        point_cloud_object = point_cloud_object_temp

# ----------------------------------------------------------------------------
# 2. Crop combined cloud and save intermediate file
# - Use intensity as RGB
# - Normalize coordinates
# - Remove low x-values (mostly water)
# ----------------------------------------------------------------------------
point_cloud_object = np.concatenate((point_cloud_object, np.transpose(np.tile(
                        np.copy(point_cloud_object[:,3]), (2,1)))), axis = 1)
maxElements = np.max(point_cloud_object, 0)
minElements = np.amin(point_cloud_object, 0)
for i in range(3):
    point_cloud_object[:,i] = point_cloud_object[:,i] - minElements[i]
maxElements = np.max(point_cloud_object, 0)
minElements = np.amin(point_cloud_object, 0)
point_cloud_object = np.delete(point_cloud_object,
                np.where((point_cloud_object[:, 0] < 750))[0], axis=0)
pcd = o3d.geometry.PointCloud()
pcd.points = o3d.utility.Vector3dVector(point_cloud_object[:,:3])
pcd.colors = o3d.utility.Vector3dVector(point_cloud_object[:,3:6])
o3d.io.write_point_cloud(".\\results\\beforeDBSCAN.pcd", pcd)

# Open3D tip: window may look blank, press '4' to switch to ortho view
o3d.visualization.draw_geometries([pcd])

# ----------------------------------------------------------------------------
# 3. Run DBSCAN clustering
# ----------------------------------------------------------------------------
with o3d.utility.VerbosityContextManager(
    o3d.utility.VerbosityLevel.Debug) as cm:
    labels = np.array(pcd.cluster_dbscan(
        eps=1.4, min_points=10, print_progress=True))
```

```
49
50  max_label = labels.max()
51  print(f"point cloud has {max_label + 1} clusters")
52
53  # Assign colors to clusters
54  colors = plt.get_cmap("tab20")(labels / (max_label if max_label > 0 else 1))
55  colors[labels < 0] = 0
56  pcd.colors = o3d.utility.Vector3dVector(colors[:, :3])
57
58  # ------------------------------------------------------------------------
59  # 4. Save and display clustered results
60  # ------------------------------------------------------------------------
61  o3d.io.write_point_cloud(".\\results\\DBSCAN.pcd", pcd)
62  o3d.visualization.draw_geometries([pcd])
```