Modified Octree algorithm

This script implements a modified Octree algorithm designed to process 3D point cloud data in LAS format. The algorithm constructs a hierarchical Octree structure where each node represents a cubic region of space, and a sphere is embedded within each cube. Points within the point cloud are analyzed to determine whether they lie inside or outside the sphere. Only the points inside the sphere are retained for further processing, while the points outside the sphere are discarded.

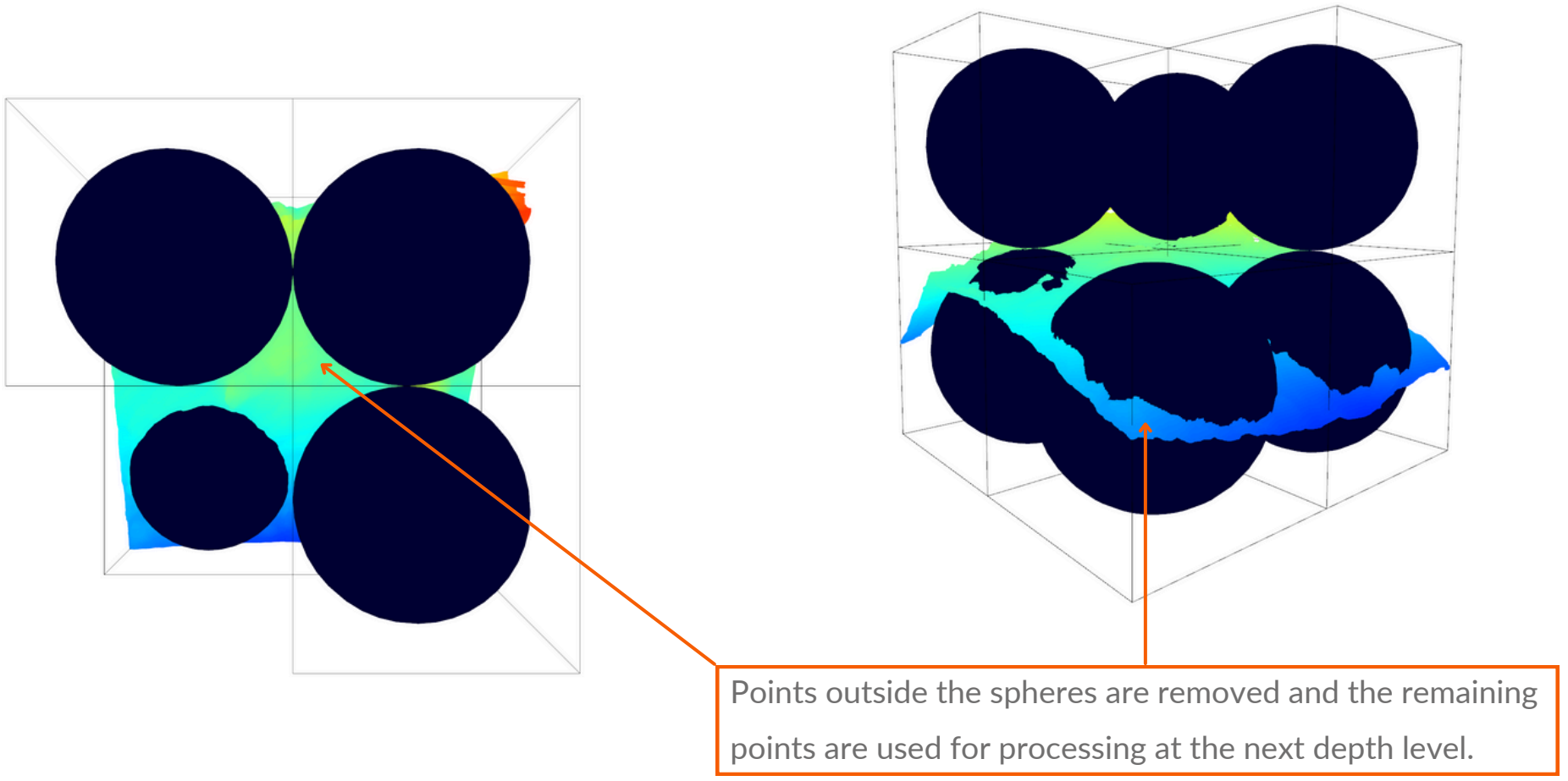

Fig. 1.1, 1,2. Explanatory schemes showing how the spheres are created and the points are filtered. These two schemes above show the creation of spheres at depth = 1.

For nodes containing points within the sphere, the retained points are further subdivided into eight smaller cubes (child nodes), each with its own embedded sphere. This recursive process continues down the hierarchy, allowing the algorithm to progressively filter and refine the point cloud data at increasing levels of detail. The spherical filtering within each cube ensures that only points of interest are retained and the process continues until the maximum depth of the Octree (max_depth) defined by the user is achieved. Once the filtering and subdivision are complete, the script aggregates the filtered points and visualizes the resulting point cloud.

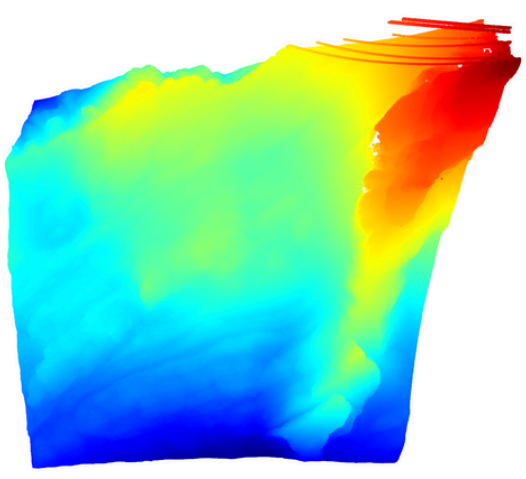

Fig. 2.1 - The Original point cloud

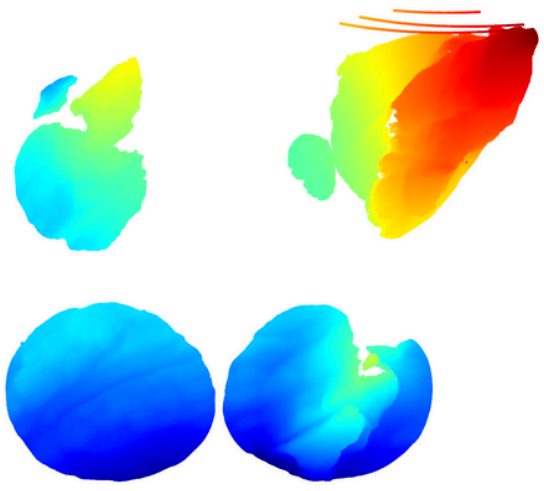

Fig. 2.2 - Results at depth level 1

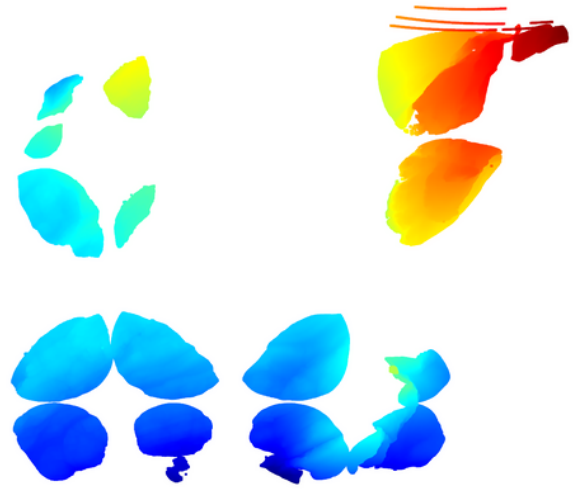

Fig. 2.3 - Results at depth level 2

User-Configurable Parameters

The script allows users to configure several parameters to customize the processing of the point cloud. These parameters include the path to the LAS file, the maximum depth of the octree, the voxel size for downsampling, and whether to filter the points in the initial octree cube. Below is a detailed description of each parameter:

1.las_file_path:
The path to the LAS file containing the point cloud data. This file should be in LAS format, which is commonly used for storing LiDAR point cloud data.

Example: "C:/Users/karol/Desktop/Coding/Matom AI task/2743_1234.las"

2. Max_depth:
The maximum depth of the octree. The octree is a hierarchical spatial structure that divides the 3D space into smaller cubic regions (nodes). The max_depth parameter controls how many levels of subdivision are performed. A higher value results in a more detailed octree.

3. downsampling_voxel_size:
Downsampling reduces the density of the point cloud by averaging points within each voxel. This can speed up processing and reduce memory usage. It is useful for testing the algorithm and making the processing faster. The default value is set to 0 and the original cloud density is retained unless specified otherwise by the user.

Example:

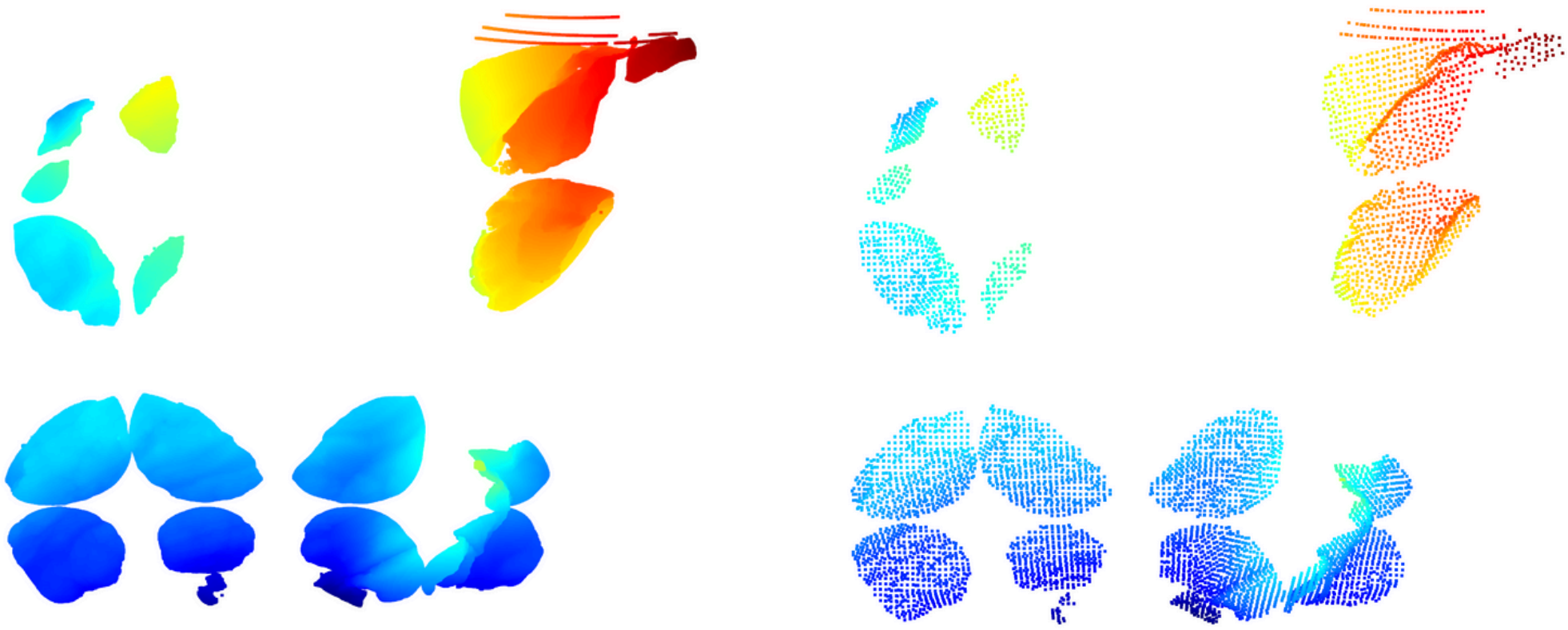

Fig. 3.1, 3.2. Comparison of results with downsampling_voxel_size set to 0 (left) and downsampling_voxel_size set to 10 (right)

4. filter_initial_cube

A parameter indicating whether to filter the first initial cube of the octree encompassing the whole cloud object at depth = 0. The parameter is set to False by default and the filtering will only occur from depth = 1

Example:

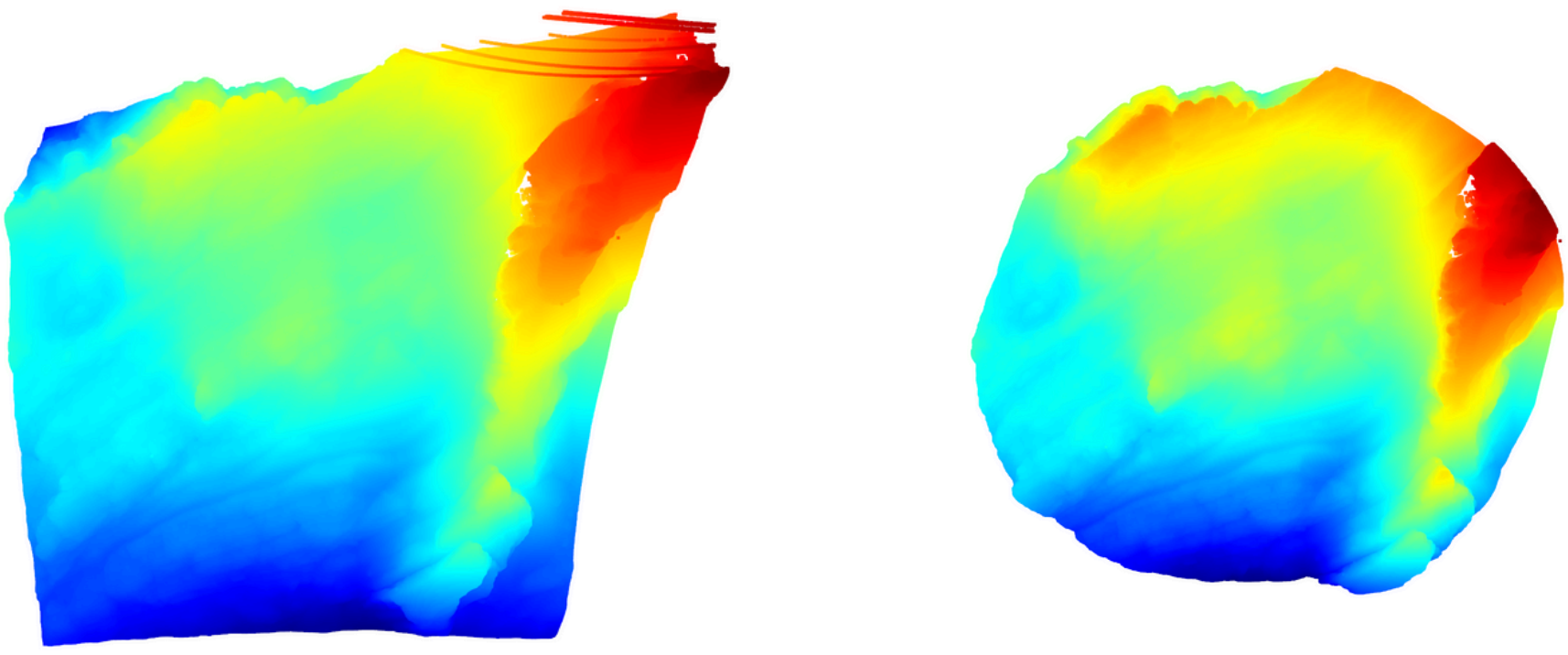

Fig. 4.1, 4.2. Comparison of results at depth = 0 with filter_initial_cube set to False (left) and filter_initial_cube set to True (right).