TP 1: Basic operations and structures on point clouds

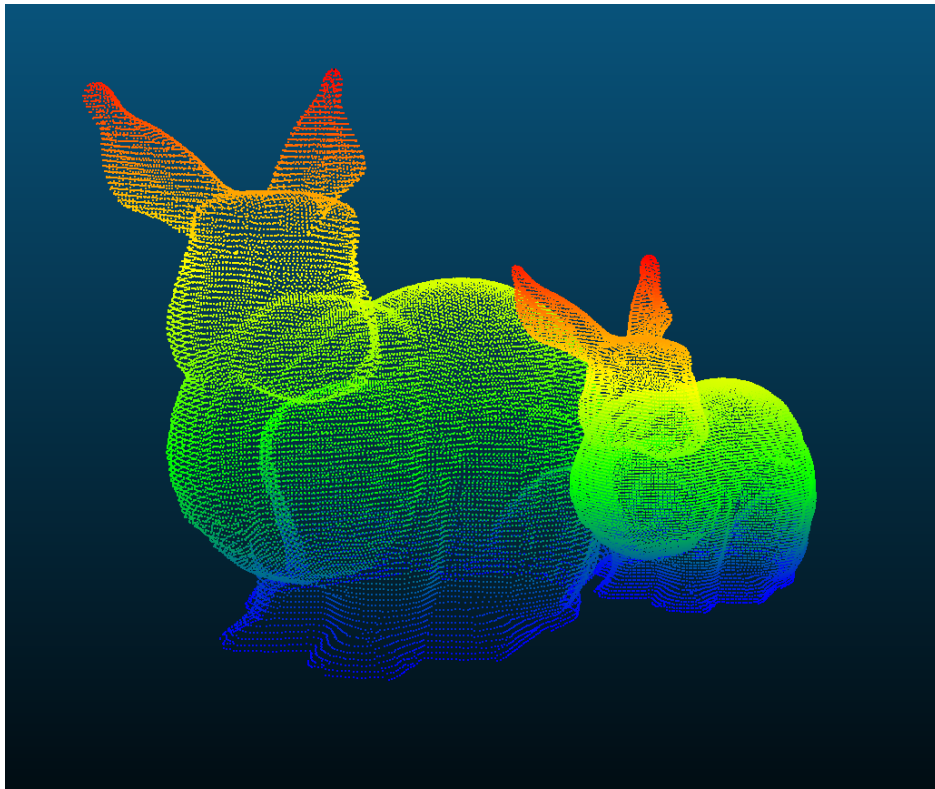Martin Jolif

January 11, 2025

1 Question 1

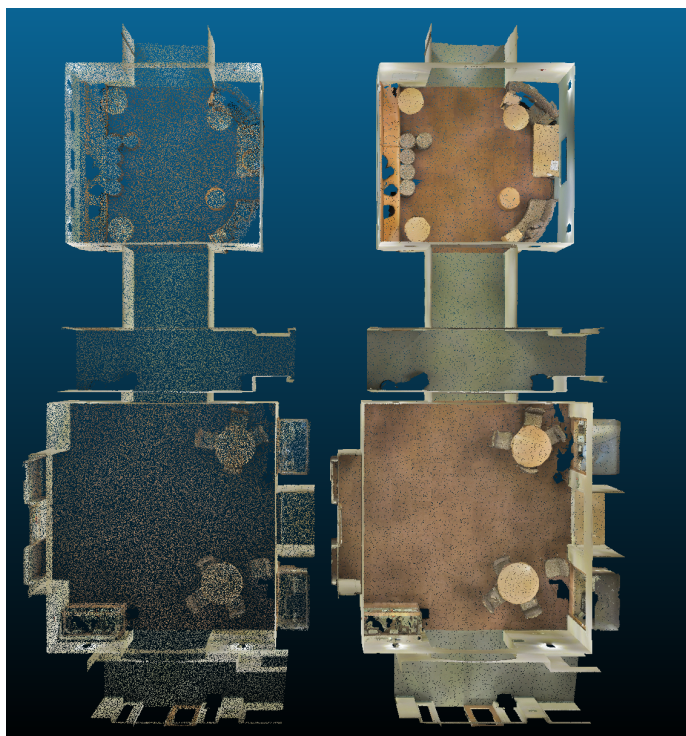Figure 1: Original and transformed bunny

2 Question 2

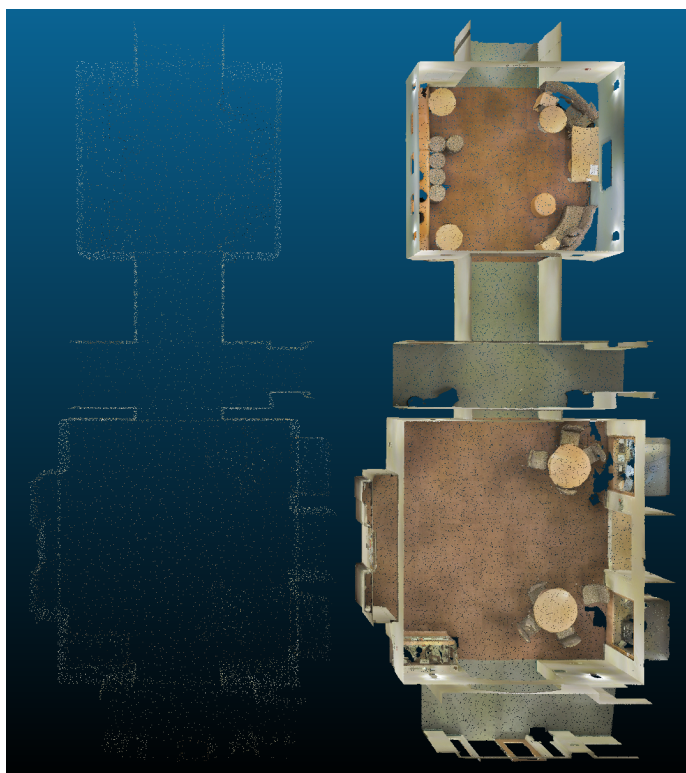Figure 2: Decimated and original "indoor_scan" point cloud (k=10)

Figure 3: Decimated and original "indoor_scan" point cloud (k=300)

3 Question 3

10 spherical neighborhoods with a radius of 20cm computed in 47.126 seconds. 10 KNN computed in 5.864 seconds. Computing spherical neighborhoods on the whole cloud would have taken 3978 hours. Computing KNN on the whole cloud would have taken 495 hours.

4 Question 4a

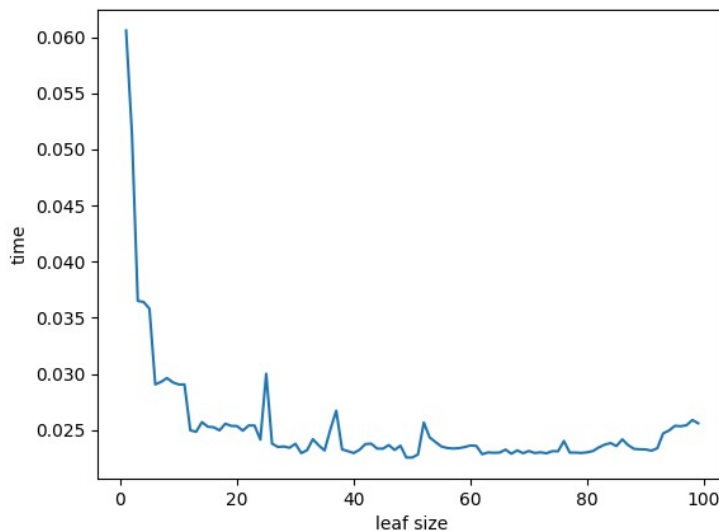Since the computing time, the fastest spherical neighborhoods search is for a leaf size of 50:

Figure 4: Evolution of the computing time (seconds) depending on the leaf size of the Kdtree

The optimal leaf size balances two factors: minimizing the number of nodes to traverse while keeping the leaf node size small enough so that the cost of searching within a leaf is not too high. Thus, the optimal size is a trade-off.

A leaf size of 1 is suboptimal because it leads to higher traversal overhead.

5 Question 4b

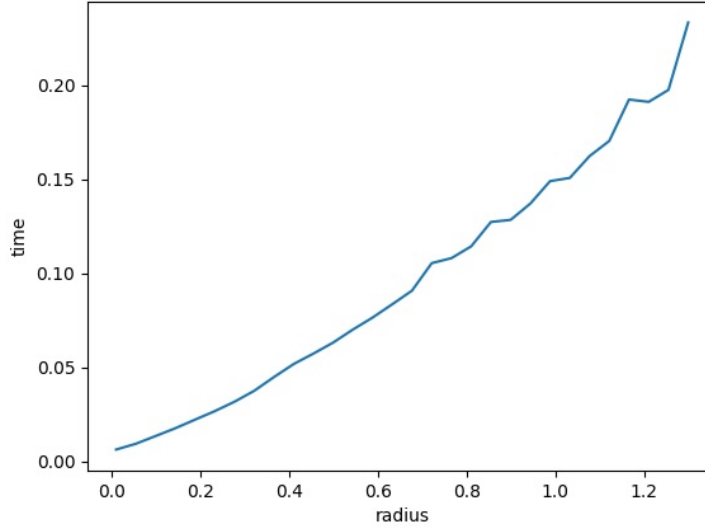Here are the timings obtained with KDTree as a function of radius:

Figure 5: Evolution of the computing time (seconds) depending on the radius (optimal leaf size for Kdtree)

It would take 158 seconds to compute the spherical neighborhoods on the whole cloud.

6   Question 5

Grid subsampling maintains a more structured and uniform downsampling, preserving geometry and avoiding bias in dense areas, while decimated subsampling is simpler but may result in uneven and less accurate representations.
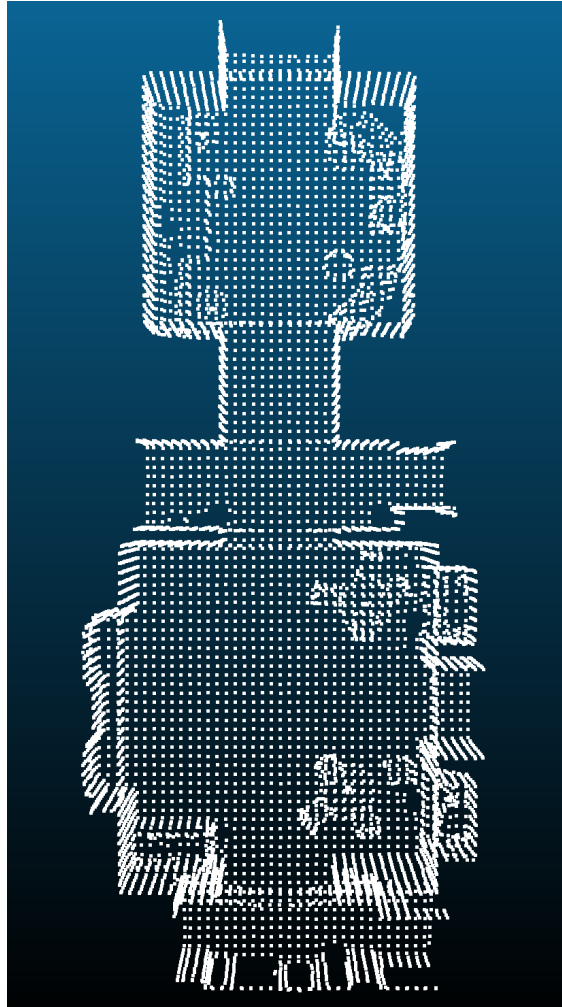
Figure 6: Grid subsampling of the "indoor_scan" point cloud