Point Clouds and 3D Modelisation
TP1: Basic operations and structures on point clouds

Thibaud Ardoin
Alain Riou

January 24, 2020

Question 1 See figure 1.

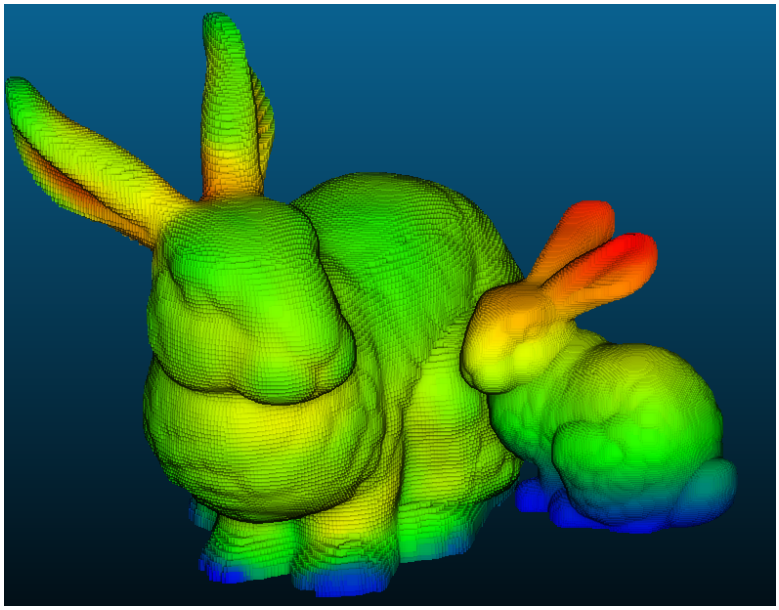

Figure 1: 3D representation of the original and transformed bunnies

Question 2 The table below indicates the time needed to search the neighborhoods with both methods for 10 queries, and the estimated time needed to compute on the whole point cloud.

	spherical	**KNN**
10 queries	0.480 seconds	0.615 seconds
whole point cloud	41 hours	52 hours

Question 3 The fastest spherical neighborhood search is reached when `leaf_size` = 84. In fact, the leaf size corresponds to the number of elements in each voxel. For a leaf size of 1, finding the right element of the voxel one is looking into is thus immediate (there is only one element) but the tree construction is very slow, and so is the research of the right voxel. On the contrary, if the leaf size is too big, finding the right voxel is fast but intra-voxel search can be too slow. Therefore a good balance has to be found.-

Question 4 According to figure 2, the computation time of the spherical neighborhoods search seems to be proportional to the radius. For $r = 20$cm, search could be computed within 2 minutes and 27 seconds, which is much faster than brute force method.

Question 5 Both subsampling results are shown on figure 3 (it is strongly advised to zoom in order to see anything).

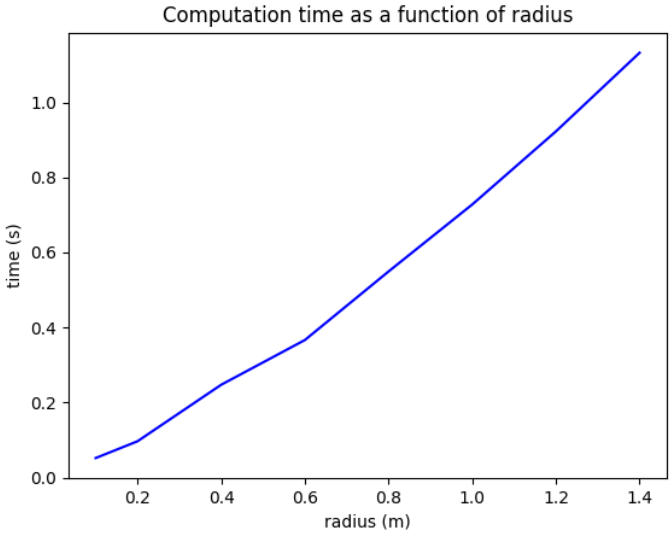

Figure 2: Computation time of spherical neighborhoods search for different radii.

One clear advantage of grid subsampling on decimated one is that it produces a more regular point cloud, and thus avoids meaningless accumulations of many points in small regions. However, with fixed-size voxels, details of objects are erased.

On the other hand, decimated subsampling *a priori* conserves the same ratio of points as in the original point clouds, and thus keeps details if textured zones contains more points than flat ones in the original point cloud. Nevertheless, there are several drawbacks: one cannot control the density of point clouds, and usually the number of points per region depends on the distance between the sensor and the environment. Above all, one big problem with decimated sampling is that it is a non transposition-invariant subsampling, i.e. the resulting point cloud depends on the order the points coordinates are written in the `.ply` file.

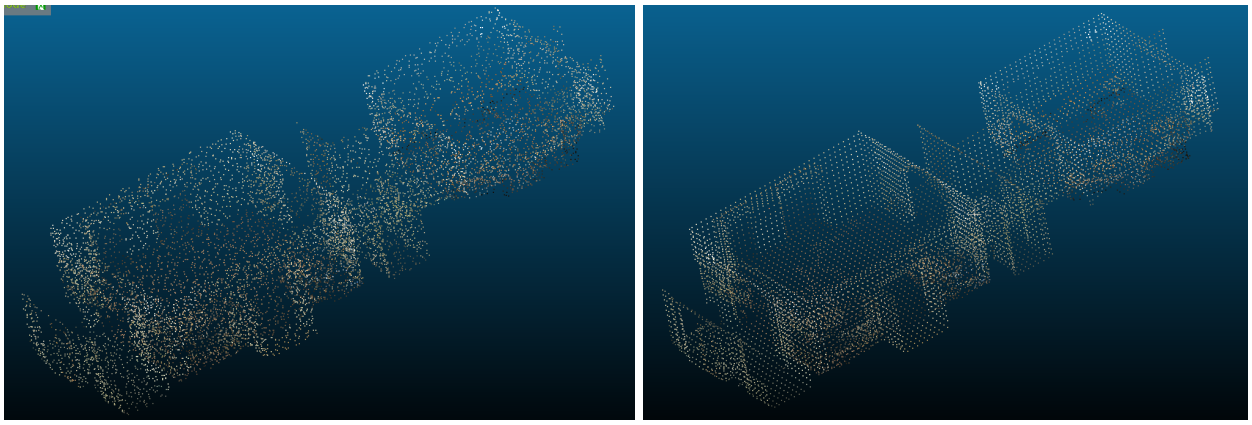

Figure 3: Subsamplings of the indoor scan. Left: decimated. Right: grid subsampling

Question bonus 1 See figure 4.

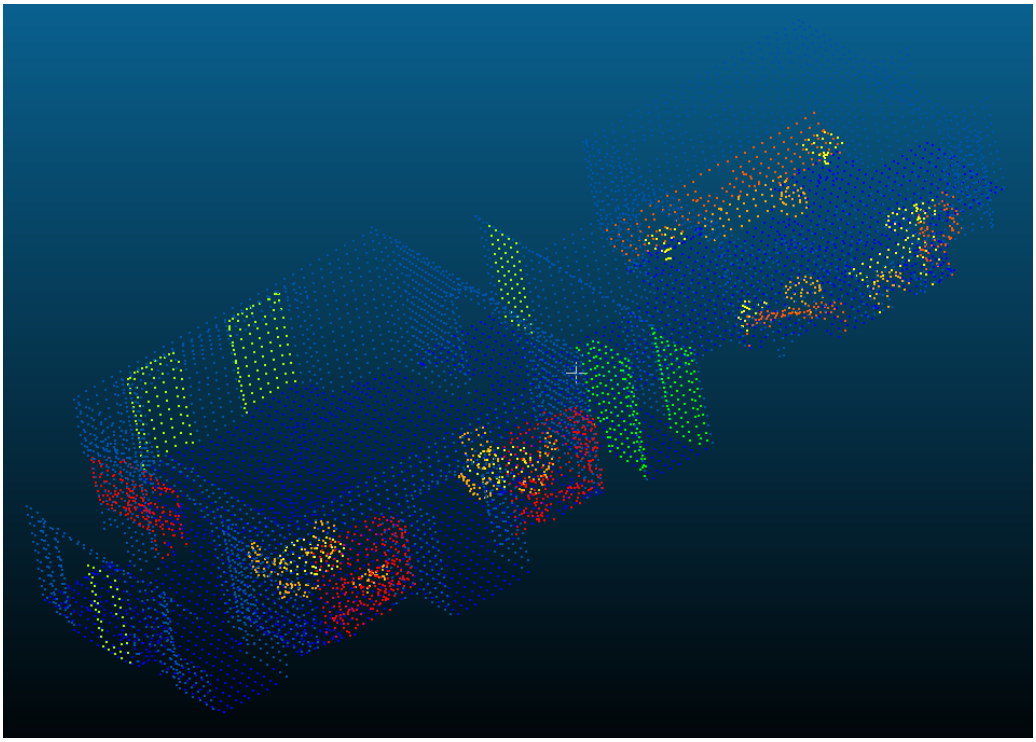

Figure 4: Grid subsampling of the indoor scan with labels