Exercise 3D Maschine Vision

Prof. Dr.-Ing. Volker Willert

thws
Technical University
of Applied Sciences
Würzburg-Schweinfurt

Sheet 2

In this exercise we will cover *3D data formats*, as well as *data structures*. The questions are small-part and can be seen as examples for potential exam questions.

Task 2.1: 3D Datenformate

2.1a)

Name three attributes that are typically assigned to elements in PLY format.

2.1b)

What are the advantages of the PCD data format compared to the PLY data format?

2.1c)

Explain what a triangular fan is and how it is encoded in PLY format.

2.1d)

Explain what the individual lines of the following text file in PLY format mean and reconstruct the surface with its properties. Note: Another object `element edge` is introduced here to assign properties to edges between two points (analogous to a point, the definition of properties is arbitrary).

```
ply
format ascii 1.0
comment author: Greg Turk
comment object:
element vertex 8
property float x
property float y
property float z
property uchar red
property uchar green
property uchar blue
element face 7
property list uchar int vertex_index
element edge 5
property int vertex1
property int vertex2
property uchar red
property uchar green
property uchar blue
end_header
0 0 0 255 0 0
0 0 1 255 0 0
0 1 1 255 0 0
0 1 0 255 0 0
1 0 0 0 0 255
1 0 1 0 0 255
1 1 1 0 0 255
1 1 0 0 0 255
3 0 1 2
3 0 2 3
4 7 6 5 4
4 0 4 5 1
4 1 5 6 2
4 2 6 7 3
4 3 7 4 0
0 1 255 255 255
1 2 255 255 255
2 3 255 255 255
3 0 255 255 255
2 0 0 0 0
```

Task 2.2: 3D data structures

2.2a)

What are the properties of structured representations of 3D data?

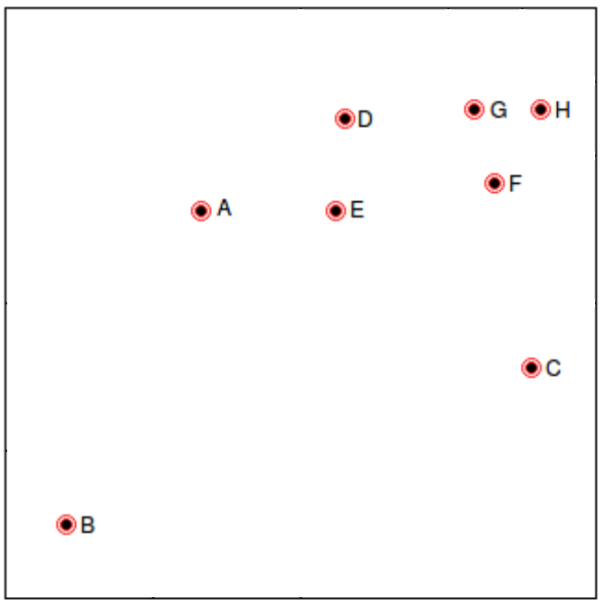
- ☐ cells can be uniquely indexed by integer numbers.
- ☐ no fixed topology
- ☐ large memory requirement
- ☐ cells are present in the grid
- ☐ no discretization of data possible
- ☐ the 3D area is not limited
- ☐ low computational effort when calculating neighborhoods

2.2b)

How does cell mapping of a point work for an octree if you define a local coordinate system for each cell with origin at the center of the cell?

2.2c)

Perform a quadtree decomposition of a 2D square surface containing a 2D point cloud consisting of 8 points (A-G), so that at the end each cell contains only one data point (so-called point-region quadtree). Draw the corresponding quadtree as a graph next to the square and specify the coordinates of the points.



2.2d)

Perform a kD tree decomposition of the following 2D point cloud and create the corresponding 2D tree as a graph, where each node contains a 2D coordinate. The order of decomposition of each dimension should be

- start with the horizontal axis at the first decomposition (left),
- then perform another decomposition, this time starting with the vertical axis (right).

Note: the first coordinate corresponds to the horizontal axis, the second to the vertical axis of the 2D space.