Exercise 3D Machine Vision

Sample solution

Prof. Dr.-Ing. Volker Willert

thws
Technical University
of Applied Sciences
Würzburg-Schweinfurt

Exercise sheet 2

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

Task 2.1: 3D data formats

2.1a)

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

Answer: coordinates, color, normal.

2.1b)

What advantages does the PCD data format have over the PLY data format?

Answer: In the PCD data format, you can map the structure of RGB-D images and define binary data types. This enables faster access to data! In addition, nD histograms can be stored as descriptors.

2.1c)

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

Answer: A triangular fan is an object in 3D computer graphics. It describes a series of N connected triangles with N + 2 points that share a central point and N + 5 triangle sides. In the PLY format, a triangular fan is coded as follows. The first digit indicates the number of points. The second digit is the index of the central point. All other digits are the indices of the other points.

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 the 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
```

Answer: The object consists of 8 vertices, 7 faces and 5 edges. The 8 points correspond to the corner points of a unit cube with edge length 1, where the origin of the coordinate system is at (0, 0, 0) and the cube is in the first octant (X >= 0, Y >= 0, Z >= 0). Four of the vertices are red, the other four are blue. 2 of the faces are triangles. 4 of the faces are triangular fans consisting of 2 triangles. 4 of the 5 edges are white, one edge is black.

Task 2.2: 3D data structures

2.2a)

What properties do structured representations of 3D data have?

- × Cells can be uniquely indexed by integer numbers
  - no fixed topology
  - large storage requirements
- × Cells are present in the grid
  - no discretization of the data possible
  - the 3D area is not limited
- × low computational effort when calculating neighborhoods

2.2b)

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

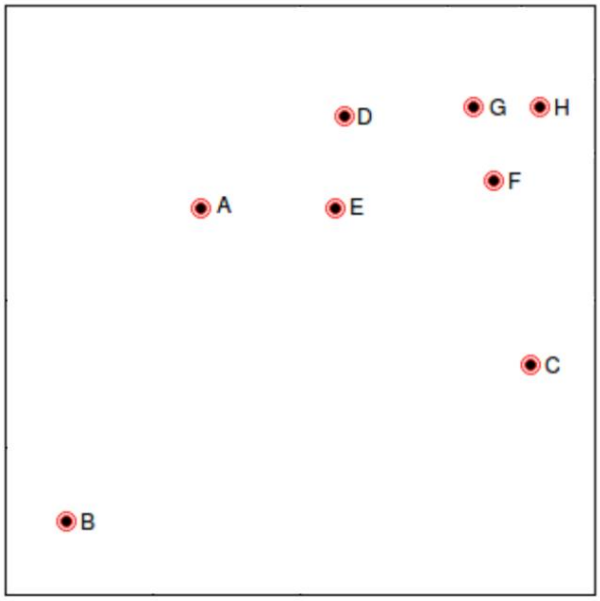
Answer: If you form the difference vector between the local coordinate origin and a point, then the cell assignment to the 8 different neighboring cells (octants) can only be done using the signs of the three coordinates x, y and z. The sign comparison x >= 0, y >= 0, z >= 0 can, for example, correspond to the 1st octant, whereby the numbering of the octants must be specified accordingly.

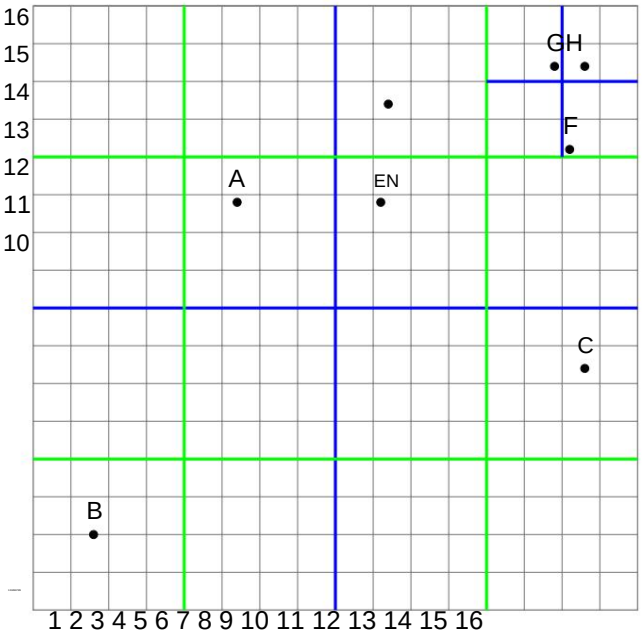
2.2c)

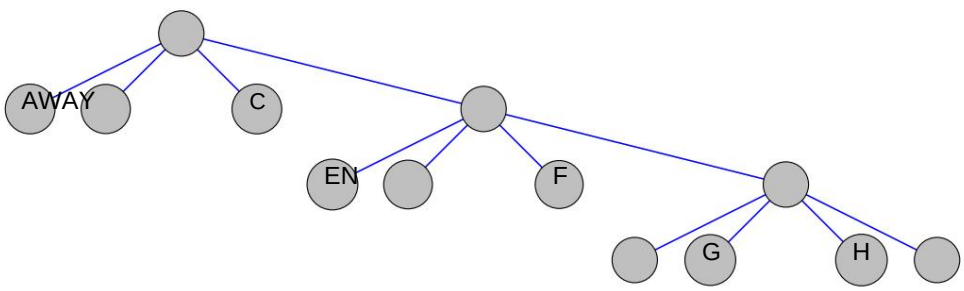
Perform a quadtree decomposition of a square 2D area containing a 2D point cloud consisting of 8 points (AG), so that in the end each cell contains only one data point (so-called point-region quadtree).
Draw the corresponding quaternary tree as a graph next to the square and indicate the coordinates of the points.

The coordinates of the points are: A = (5.4, 10.8), B = (1.6, 2), C = (14.6, 6.4), D = (9.4, 13.4), E = (9.2, 10.8), F = (14.2, 12.2), G = (13.8, 14.4) and H = (14.6, 14.4).

Quaternary tree: After each subdivision, we start with the upper left quadrant and move counterclockwise . The starting point and the direction of travel can be chosen arbitrarily, but must be clearly defined.





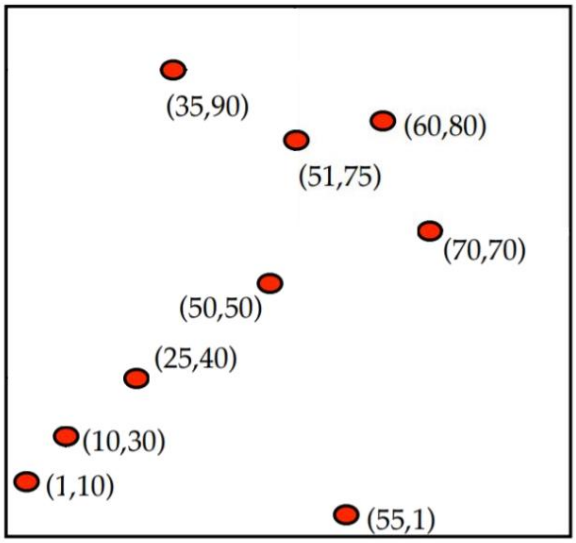


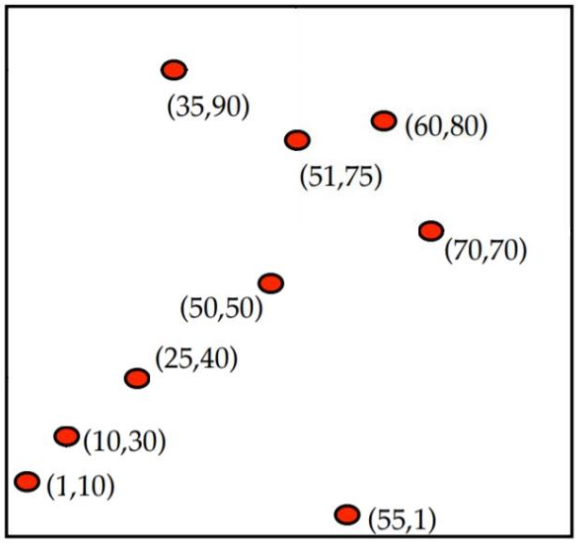
2.2d)

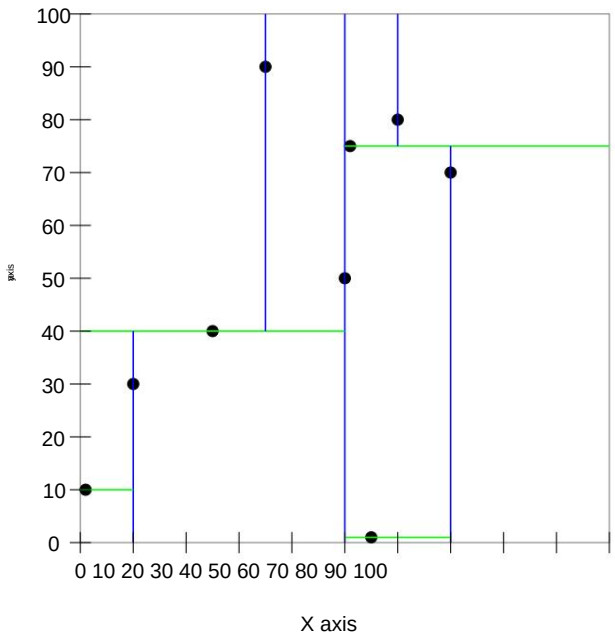
Perform a kD tree decomposition of the 2D point cloud and create the corresponding 2D tree as a graph, where each node contains a 2D coordinate. The order of the decomposition of the individual dimensions should

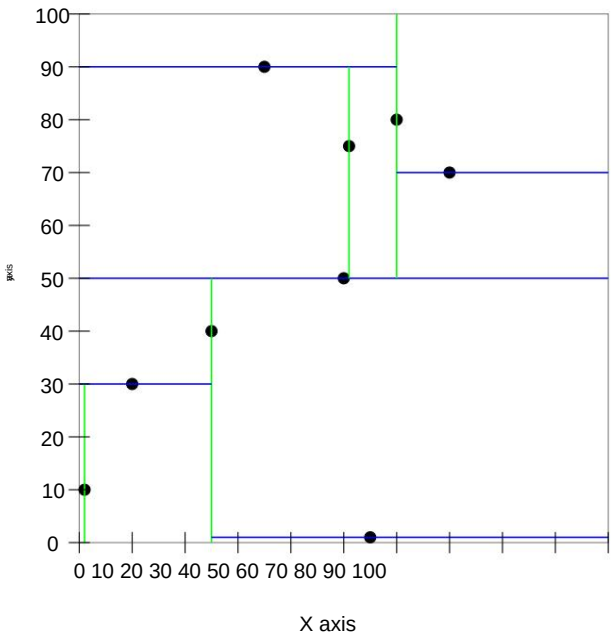
- start with the horizontal axis for the first decomposition (left),
- Then decompose again, starting this time with the vertical axis (right).

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









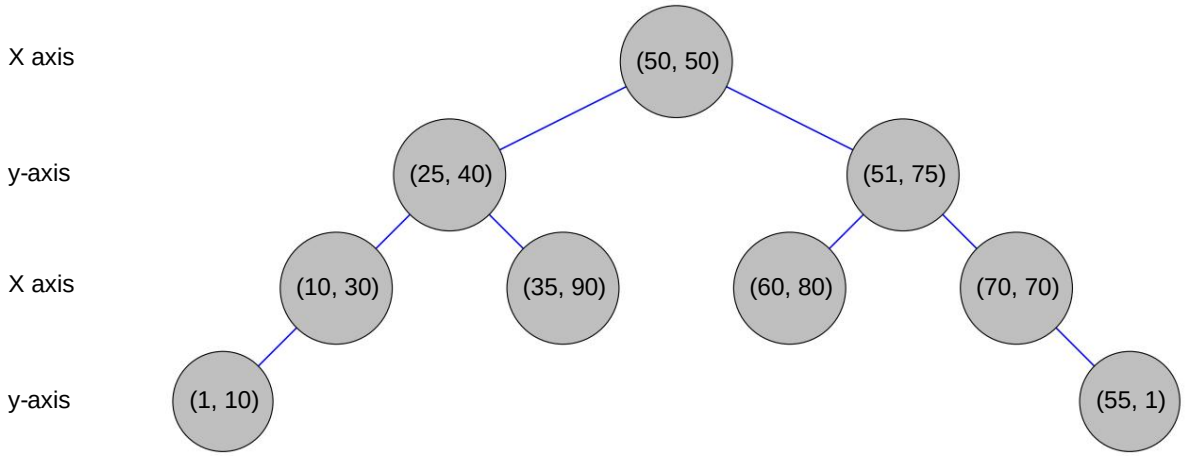


Figure 1: 2D tree decomposition starting with the x-axis.

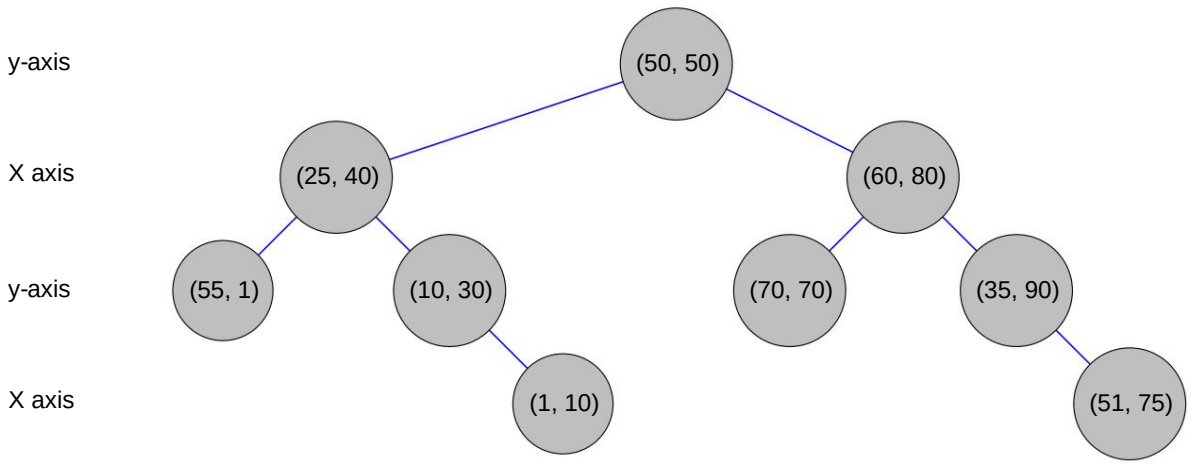


Figure 2: 2D tree decomposition starting with the y-axis.