

Übung 3D Maschinelles Sehen

Musterlösung

Prof. Dr.-Ing. Volker Willert



Übungsblatt 2

In dieser Übung behandeln wir *3D Datenformate*, sowie *Datenstrukturen*. Die Fragen sind kleinteilig und können als Beispiele für potentielle Klausuraufgaben gesehen werden.

Aufgabe 2.1: 3D Datenformate

2.1a)

Nennen Sie drei Attribute, die den Elementen im PLY Format typischerweise zugeordnet werden.

Antwort: Koordinaten, Farbe, Normale.

2.1b)

Welche Vorteile besitzt das PCD Datenformat gegenüber dem PLY Datenformat?

Antwort: Im PCD Datenformat kann man die Struktur von RGB-D-Bildern abbilden und binäre Datentypen definieren. Dies ermöglicht einen schnelleren Zugriff auf Daten! Außerdem können n-D Histogramme als Deskriptoren gespeichert werden.

2.1c)

Erklären Sie was ein Dreiecksfächer ist und wie dieser im PLY Format kodiert wird.

Antwort: Ein Dreiecksfächer ist ein Objekt in der 3D-Computergrafik. Er beschreibt eine Reihe von N verbundenen Dreiecken mit $N + 2$ Punkten, die sich einen zentralen Punkt teilen und $N + 5$ Dreiecksseiten. Im PLY Format wird ein Dreiecksfächer folgendermaßen kodiert. Die erste Ziffer bezieht die Anzahl der Punkte. Die zweite Ziffer den Index des zentralen Punktes. Alle weiteren Ziffern die Indizes der anderen Punkte.

2.1d)

Erklären Sie, was die einzelnen Zeilen der folgenden Textdatei in PLY-Format bedeuten und rekonstruieren Sie die Oberfläche mit ihren Eigenschaften. Hinweis: Hier wird ein weiteres Objekt `element edge` eingeführt, um Kanten zwischen zwei Punkten mit Eigenschaften zu belegen (analog zu einem Punkt ist die Definition der Eigenschaften beliebig).

```
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
```

Antwort: Das Objekt besteht aus 8 Punkten (vertex), 7 Flächen (faces) und 5 Kanten (edges). Die 8 Punkte entsprechen den Eckpunkten eines Einheitswürfels mit der Kantenlänge 1, wobei der Ursprung des Koordinatensystems bei $(0, 0, 0)$ liegt und der Würfel im ersten Oktanten liegt ($X \geq 0, Y \geq 0, Z \geq 0$). Vier der Eckpunkte sind rot, die anderen vier sind blau. 2 der Flächen sind Dreiecke. 4 der Flächen sind Dreiecksfächer bestehend aus 2 Dreiecken. 4 der 5 Kanten sind weiß, eine Kante ist schwarz.

Aufgabe 2.2: 3D Datenstrukturen

2.2a)

Welche Eigenschaften besitzen strukturierte Repräsentationen von 3D Daten?

- ☒ Zellen lassen sich eindeutig durch ganzzahlige Zahlen indizieren
- ☐ keine festgelegte Topologie
- ☐ großer Speicherbedarf
- ☒ Zellen liegen im Raster vor
- ☐ keine Diskretisierung der Daten möglich
- ☐ der 3D Bereich ist nicht beschränkt
- ☒ geringer Rechenaufwand bei der Berechnung von Nachbarschaften

2.2b)

Wie funktioniert die Zellenzuordnung bei einem Octree, wenn man für jede Zelle ein lokales Koordinatensystem mit Ursprung im Mittelpunkt der Zelle definiert?

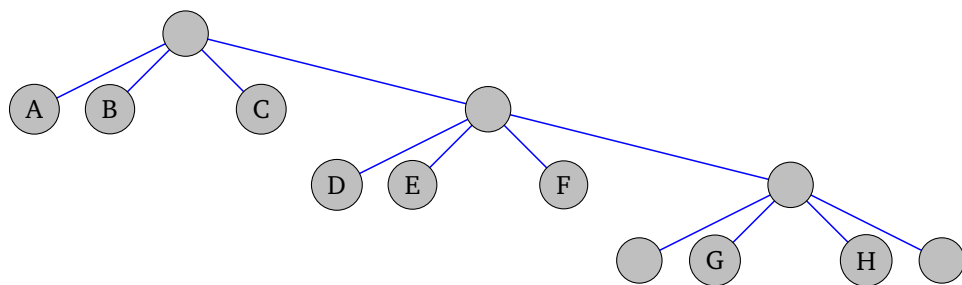
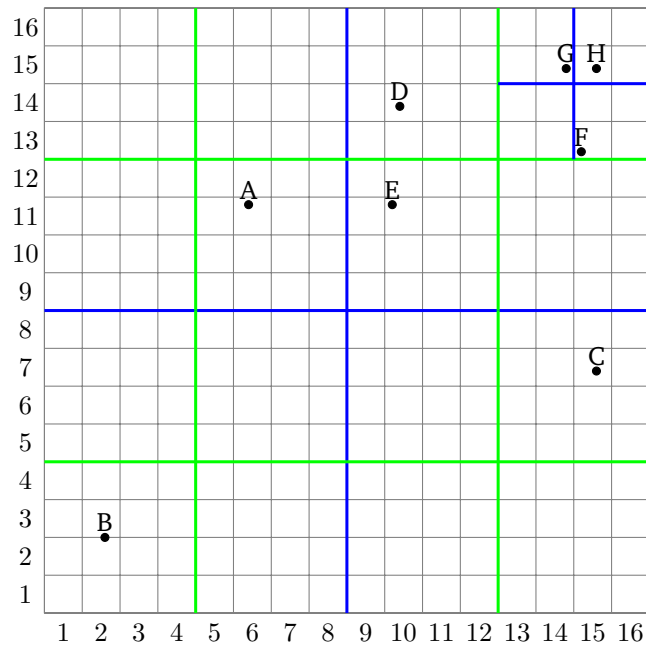
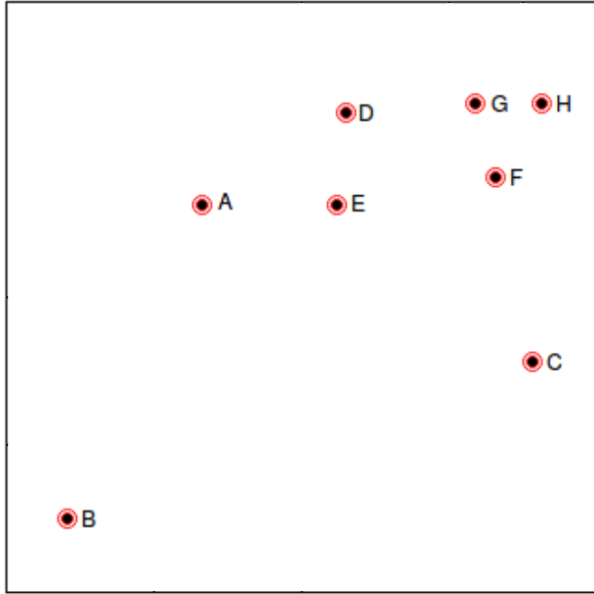
Antwort: Bildet man den Differenzvektor zwischen lokalem Koordinatenursprung und einem Punkt, dann kann die Zellzuordnung zu den 8 unterschiedlichen Nachbarzellen (Octanten) nur über die Vorzeichen der drei Koordinaten x, y und z erfolgen. Der Vorzeichenvergleich $x \geq 0, y \geq 0, z \geq 0$ kann beispielsweise dem 1. Octanten entsprechen, wobei die Nummerierung der Octanten entsprechend vorgegeben werden muss.

2.2c)

Führen Sie eine Quadtreezerlegung einer quadratischen 2D Fläche durch, in der sich eine 2D Punktwolke bestehend aus 8 Punkten (A-G) befindet, so dass am Ende jede Zelle nur noch einen Datenpunkt enthält (sog. Point-Region-Quadtree). Zeichnen Sie den dazugehörigen Quaternärbaum als Graph neben das Quadrat und geben Sie die Koordinaten der Punkte an.

Die Koordinaten der Punkte lauten: $A = (5.4, 10.8)$, $B = (1.6, 2)$, $C = (14.6, 6.4)$, $D = (9.4, 13.4)$, $E = (9.2, 10.8)$, $F = (14.2, 12.2)$, $G = (13.8, 14.4)$ und $H = (14.6, 14.4)$.

Quaternärbaum: Nach jeder Unterteilung beginnen wir mit dem linken oberen Quadranten und laufen gegen den Uhrzeigersinn. Der Startpunkt und die Durchlaufrichtung können beliebig gewählt werden, müssen jedoch eindeutig festgelegt werden.

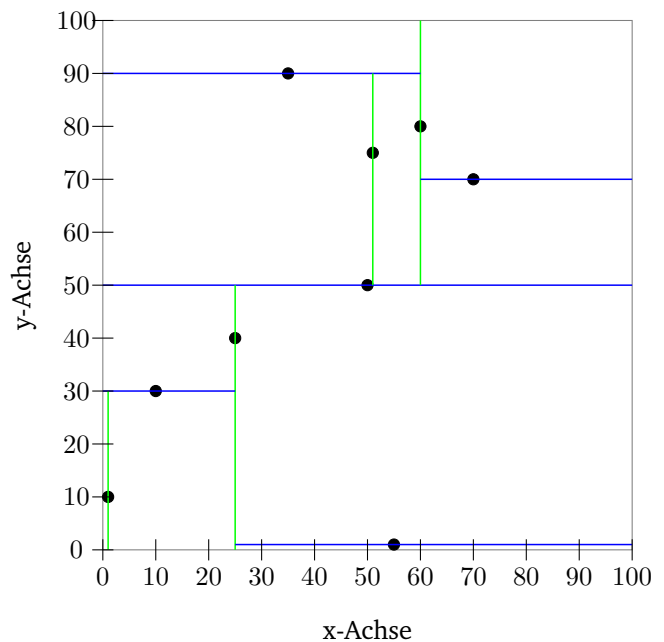
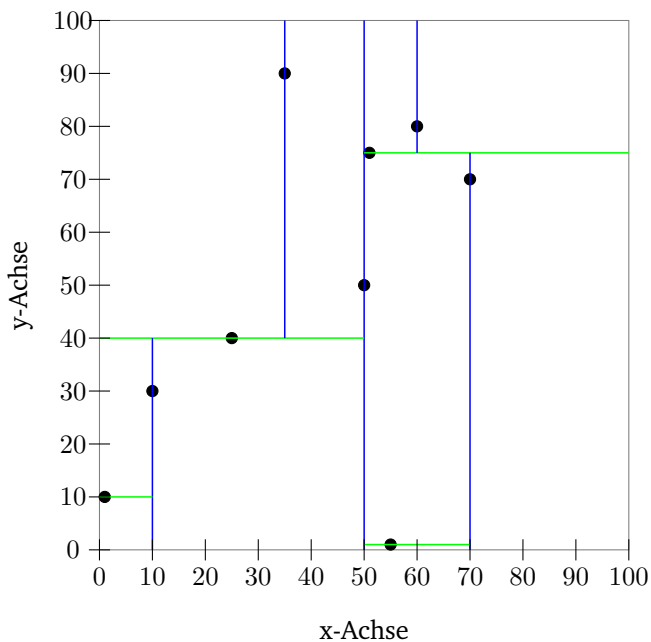
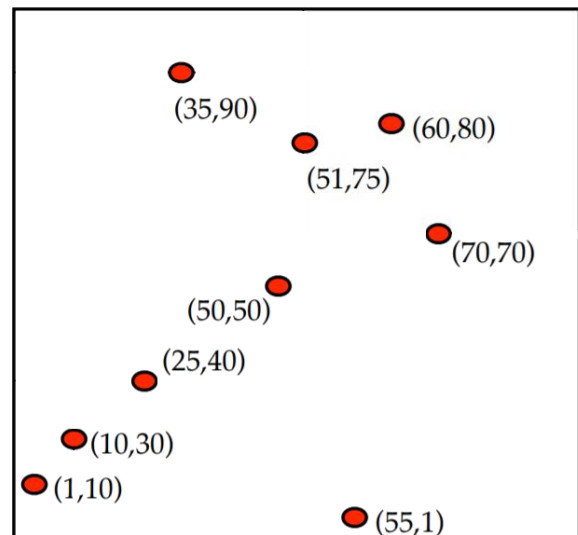
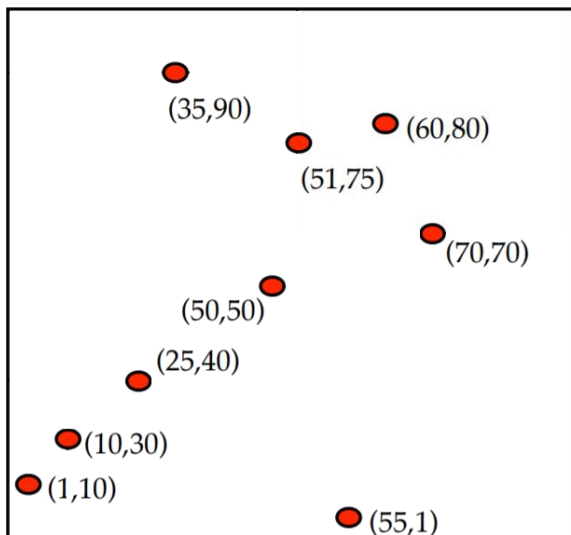


2.2d)

Führen Sie eine kD-Baumzerlegung (kD-Tree) der 2D Punktwolke durch und erstellen Sie den entsprechenden 2D-Baum als Graph, wobei jeder Knoten eine 2D Koordinate enthält. Die Reihenfolge der Zerlegung der einzelnen Dimensionen soll

- bei der ersten Zerlegung mit der horizontalen Achse beginnen (links),
- danach führen Sie erneut eine Zerlegung durch und beginnen diesmal mit der vertikalen Achse (rechts).

Hinweis: die erste Koordinate entspricht der horizontalen Achse, die zweite der vertikalen Achse des 2D Raums.



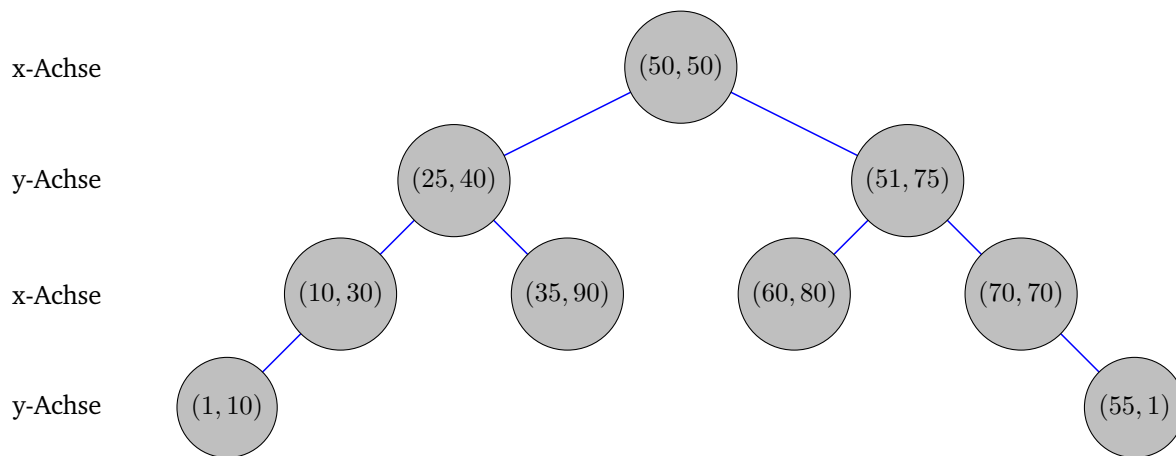


Abbildung 1: 2D-Baum Zerlegung beginnend mit der x-Achse.

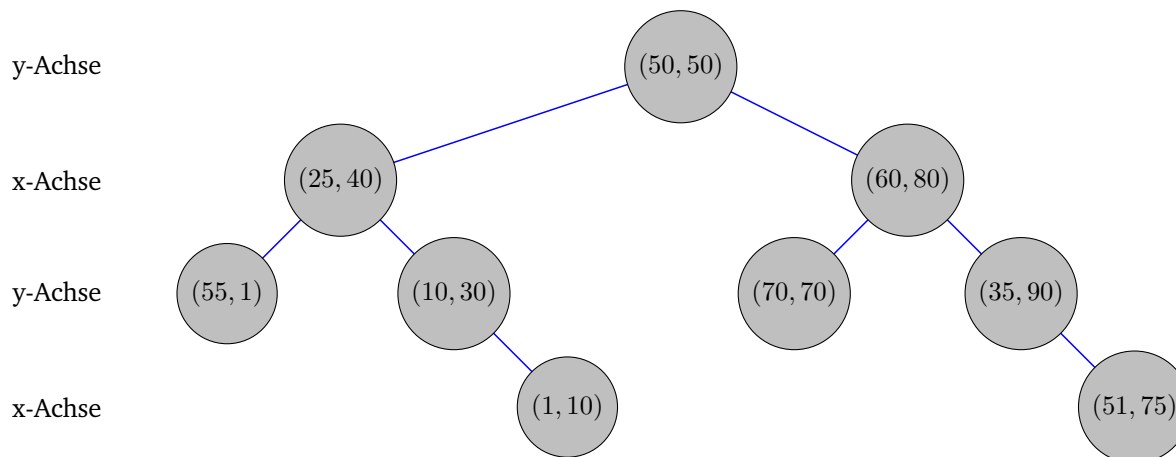


Abbildung 2: 2D-Baum Zerlegung beginnend mit der y-Achse.