

# Masterarbeit

## Inhaltsverzeichnis

### I. Überblick

1. Punktwolke .....	2
2. Daten .....	2
3. ... ..	2

### II. Berechnung

1. Ablauf .....	2
2. Separierung in Bäume .....	2
3. Segmentierung von einem Baum .....	2
4. Baumeigenschaften .....	2
4.1. Krümmung .....	2
4.2. Punkthöhe .....	3
4.3. Varianz in Scheibe .....	3
4.4. ... ..	3
5. Eigenschaften für Visualisierung .....	3
5.1. Normale .....	3
5.2. Punktgröße .....	3
5.2.1. Detailstufe .....	3
6. Baumart .....	3

### III. Meshing

### IV. Visualisierung

1. Technik .....	3
2. Punkt .....	4
3. Dynamische Eigenschaft .....	4
3.1. Lookup Table .....	4
4. Subpunktwolken (Bäume) .....	4
4.1. Selektion (Raycast) .....	4
5. Eye Dome .....	4
6. LOD Octree .....	4
6.1. Kostenbudget? .....	4
7. Kamera/Projektion .....	4
7.1. Kontroller .....	4
7.1.1. Orbital .....	5
7.1.2. First person .....	5
7.2. Projektion .....	5
7.2.1. Perspektive .....	5
7.2.2. Orthogonal? .....	5
8. Bedienung/Interface .....	5

# I. Überblick

## 1. Punktwolke

- Menge von Punkten
- mindestens Position

## 2. Daten

- Waldstücke
- Deutschland
- terrestrial und arial
- zusätzlich manuelle Datenbestimmung
- nur Position bekannt

## 3. ...

# II. Berechnung

## 1. Ablauf

Als Bild

1. Eingabedateien
  - Dateien laden
2. Punktmenge
  - Segmentierung in Bäume, Boden...
3. Liste von Bäumen
  - Analyse der Bäume
4. Liste von analysierten Bäumen
  - Generierung von Octree
5. Octree + LOD für Visualisierung

getrennte Phasen (Phase ist in sich parallelisiert)

1. Laden der Dateien
2. Segmentierung
3. Analyse + Generierung

## 2. Separierung in Bäume

- ?

## 3. Segmentierung von einem Baum

- ?

## 4. Baumeigenschaften

### 4.1. Krümmung

1. Hauptkomponentenanalyse
  - $\lambda_i$  mit  $i \in \mathbb{N}_0^2$  und  $\lambda_i > \lambda_j$  wenn  $i > j$

$$2. \ c = \frac{3\lambda_2}{\lambda_0 + \lambda_1 + \lambda_2}$$

- $c \in [0, 1]$

## 4.2. Punkthöhe

$$1. \ h = \frac{p_y - y_{\min}}{y_{\max} - y_{\min}}$$

- $h \in [0, 1]$

## 4.3. Varianz in Scheibe

1. 5 cm Scheiben
2. geometrischen Schwerpunkt berechnen
3. Varianz  $v$  berechnen
4.  $x = \frac{v_i}{v_{\max}}$ 
  - $x \in [0, 1]$

## 4.4. ...

# 5. Eigenschaften für Visualisierung

## 5.1. Normale

1. Hauptkomponentenanalyse
2. Eigenvektor für  $\lambda_2$

## 5.2. Punktgröße

1. Durchschnittliche Abstand zu umliegenden Punkten
2. Ausgleichsfaktor?

### 5.2.1. Detailstufe

1. Grid
  - Größe abhängig von Leafgröße, wird größer für größere Blätter
2. Kombination von Punkten
  - Größe als Fläche addieren
  - Normale Durchschnitt
  - Position durchschnitt
  - Eigenschaften?

## 6. Baumart

- ?

# III. Meshing

- ?

# IV. Visualisierung

## 1. Technik

- Rust
- WebGPU (wgpu)

- native Window (website?)
- LAS/LAZ

## 2. Punkt

- Instancing
- quad rect
- Ausdehnung mit Normale
- Discard mit Distanz für Kreis (Kreisfläche)

## 3. Dynamische Eigenschaft

### 3.1. Lookup Table

## 4. Subpunktvolken (Bäume)

### 4.1. Selektion (Raycast)

## 5. Eye Dome

1. Post processing
2. depth image
3. anliegender Pixel mit maximalem Abstand
  1.  $(-1, 0), (0, -1), (1, 0), (0, 1)$
4. Parameter  $m$
5.  $x = \frac{\text{maximaler abstand}}{m}$
6. auf  $[0, 1]$  beschränken
7. Parameter color?
8. Pixel mit color und  $x$  als  $\alpha$  überlagern

## 6. LOD Octree

1. (Octree begriffe in English)
2. Octree mit maximaler Blattgröße  $1 \ll 15?$  (32k)
3. Blätter mit mehr Punkten werden in 8 Kinderknoten geteilt
  - Punkte auf Kinder verteilen
4. non Leaf Knoten wird LOD aus Kindern berechnet
  1. Punkte kombinieren
  2. Für Eigenschaften wert von einem Punkt übernehmen
5. rekursiv von Kindern bis zum Root
6. Beim rendern für entferne Punkte nur Lod Stufe verwenden
  1. je näher so genauere LOD Stufe

### 6.1. Kostenbudget?

- Anpassung der Genauigkeit
  - Verringerung des Aufwands

## 7. Kamera/Projektion

### 7.1. Kontroller

**7.1.1. Orbital**

**7.1.2. First person**

**7.2. Projektion**

**7.2.1. Perspektive**

- FOV

**7.2.2. Orthogonal?**

**8. Bedienung/Interface**

- ?