

Übungen - Bildgenierung

Übung 11.

Jose Jimenez

Angewandte Informatik
Bergische Universität Wuppertal

January 22, 2025



Table of Contents

1 Aufgabe 34: Raytracing nach Whitted



Raytracing nach Whitted

- Die Struktur Ray
- Die Struktur HitInfo
- Die Struktur VisibleInfo



Raytracing nach Whitted

- **Die Struktur Ray**
- Die Struktur HitInfo
- Die Struktur VisibleInfo

```
typedef struct {  
    Vector3d origin;  
    Vector3d direction;  
    Object* insideObject {nullptr};    //Dieses Objekt wird Ignoriert.  
} Ray;
```

Die Struktur Ray stellt einen Strahl mit Ausgangspunkt und Richtung dar. Alle Informationen dazu finden Sie in der Datei ray.h.



Raytracing nach Whitted

- Die Struktur Ray
- **Die Struktur HitInfo**
- Die Struktur VisibleInfo

```
typedef struct {  
    bool hit {false};  
    // rayparam:    position = origin + rayparam*direction : linie(t)  
    // -> für Entfernungsbestimmung.  
    double rayparam {std::numeric_limits<double>::max()};  
    Vector3d position {0,0,0};  
    Vector3d normal {0,0,0};  
    Object *object {nullptr};  
} HitInfo;
```

Die Struktur HitInfo bündelt die Schnittpunkt-Informationen eines Strahls mit einem Objekt. Alle Informationen dazu finden Sie in der Datei hitinfo.h.



Raytracing nach Whitted

- Die Struktur Ray
- Die Struktur HitInfo
- **Die Struktur VisibleInfo**

```
typedef struct {  
    bool visible {false};  
    Vector3d direction {0,0,0};    /* Richtung vom Punkt  
                                   zur Lichtquelle, normiert*/  
  
    double distance {0};  
    Light *light {nullptr};  
} VisibleInfo;
```

Die Struktur VisibleInfo bündelt die Informationen, ob eine Lichtquelle von einem Punkt aus sichtbar ist. Alle Informationen dazu finden Sie in der Datei visibleinfo.h.



Raytracing nach Whitted

Zunächst müssen wir alle Eigenschaften der Szene und des Bildes abrufen. d.h.

- Ecke und spannende Vektoren der Projektionsebene bestimmen.
 - **Alle Kamerawerte.**
 - Alle Ebenes.

```
Vector3d campos = scene.getCamera().getPosition();  
Vector3d camdir = scene.getCamera().getDirection();  
Vector3d up     = scene.getCamera().getUp(); // VUP = Aufwärtsrichtung  
double horangle = scene.getCamera().getHorAngle();  
double aspect   = static_cast<double>(image.getWidth())  
                  /image.getHeight();
```

Lass uns sehen was wir machen, d.h. wie erzeugen wir die Ebenen?

