

WP Einführung in die Computergrafik

SS 2014, Hochschule für Angewandte Wissenschaften (HAW), Hamburg
Prof. Dr. Philipp Jenke, Lutz Behnke



Aufgabenblatt 7 - Portal Culling

In diesem Aufgabenblatt vervollständigen Sie eine Implementation des Portal-Culling-Algorithmus für 2D-Szenen.

Wir verwenden dazu weiter die Vektor-Klasse `Vector3` zur Repräsentation von Vektoren. Für 2D-Szenen wird die y-Koordinate aber ignoriert.

Machen Sie sich zunächst mit den vorgegebenen Klassen vertraut. Die Szene (`PortalScene2D`) ist aus Knoten (`Vector3`), Kanten (`PortalEdge`) und Zellen (`PortalCell`) zusammengesetzt. Außerdem gibt es eine Klasse zur Repräsentation des Sichtbarkeitsvolumens (`ViewVolume2D`) und eine Hilfsklasse zur Repräsentation eines Strahls (`Ray2D`). Für einige Funktionalitäten sind JUnit-Tests verfügbar (`PortalCullingSceneTest`). Außerdem gibt es einen Importer für Portal-Szenen in der Klasse `PortalSceneImporter`. Beim dem verwendeten Dateiformat handelt es sich um ein eigenes text-basiertes Format. Die Beispielszene aus Abbildung 1 ist als `scene.scene` gesichert. Sie können diese Szene so laden:

```
PortalScene2D scene = new PortalScene2D();
PortalSceneImporter importer = new
    PortalSceneImporter();
importer.importScene(scene, "scene.scene");
```

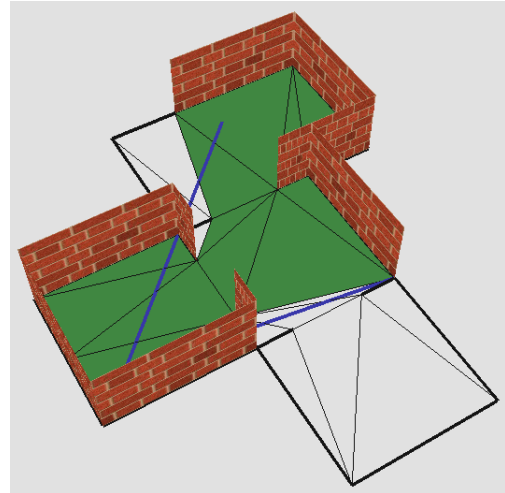


Abbildung 1: Beispielszene. Die Räume sind in dreieckige Zellen unterteilt. Jede Kante ist entweder ein Portal (dünne Linie) oder eine Wand (dicke Linie). Das Sichtvolumen ist durch blaue Linien dargestellt. Die sichtbaren Zellen (Potentially Visible Set, PVS) sind grün markiert. Zur Veranschaulichung wurden außerdem die Wände der sichtbaren Zellen als Backstein-Wände sichtbar gemacht.

Einige Methoden sind in der Vorgabe nicht implementiert, dies holen Sie in den folgenden Aufgaben nach.

a) Visualisierung

Implementieren Sie Funktionalität zum Darstellen der Szene. Mindestens die folgende Information muss sichtbar sein:

- Zellen
- Sichtbarkeitsvolumen
- Kanten (Wand vs. Portal)

Zur Umsetzung können Sie entweder einen neuen Szenengraph-Knoten implementieren oder einfach die `drawGLContent()`-Methode Ihrer Frame-Klasse verwenden.

b) Winkelberechnung

Implementieren Sie die Methode `ViewVolume2D.getAngle()`. Diese Methode liefert den Winkel zwischen den beiden Begrenzungslinien im Gradmaß (nicht Bogenmaß!) zurück. Der Winkel wird als Ganzzahl (`int`) angegeben. Sie können annehmen, dass der Winkel nie größer als 180 Grad ist.

c) Schnittpunkt zweier Strahlen

Implementieren Sie die Methode `Ray2D.intersect(Ray2D)`. Ein Strahl (engl. ray) ist dazu in der Form

$$p + \lambda \cdot \text{direction}$$

gegeben. Im 2D-Fall gibt es immer entweder einen Schnittpunkt oder die Richtungsvektoren (`direction`) der beiden Strahlen verlaufen parallel zueinander. Die Rückgabe der Methode hat den Typ `IntersectionResult`

(inneren Klasse in Ray2D). `IntersectionResult` beinhaltet die λ -Parameter der beiden Strahlen am Schnittpunkt. Falls es keinen Schnittpunkt gibt, dann soll die Methode `intersect()` null zurückliefern.

d) Punkt-Zellen-Inklusionstest

Implementieren Sie die Methode `PortalScene2D.cellContainsPoint(PortalCell cell, Vector3 p)`. Die Methode prüft, ob der Punkt p in der Zelle `cell` liegt oder nicht. Den Dreiecks-Inklusionstest für einen Punkt über baryzentrische Koordinaten haben Sie bereits in einer Hausaufgabe kennengelernt.

e) Schnitt Sichtbarkeitsvolumen-Kante

Die Methode `PortalScene2D.intersect(PortalEdge, ViewVolume2D)` berechnet den Schnitt zwischen einer Kante und einem Sichtbarkeitsvolumen. Das Ergebnis ist wieder ein Sichtbarkeitsvolumen. Die Methode ist bereits implementiert. Vollziehen Sie das Verhalten der Methode nach, verstehen Sie die Sonderfälle und seien Sie in der Lage, den Algorithmus zu erklären. Die betrachteten Fälle sind in Abbildung 2 skizziert.

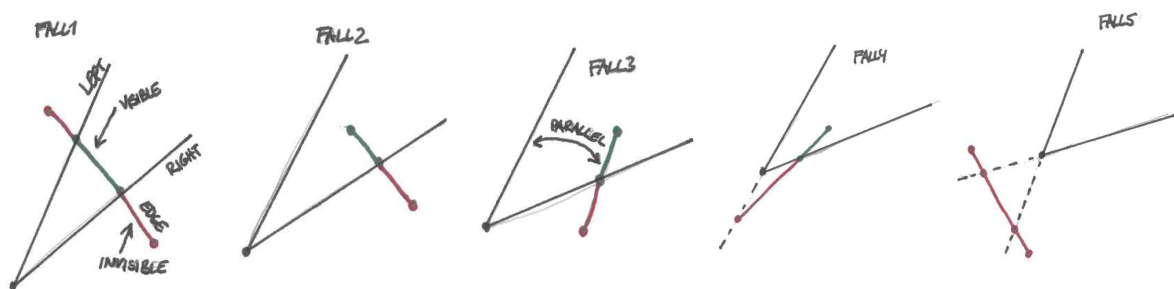


Abbildung 2: Verschiedene Fälle, die bei der Berechnung des Schnitts eines Sichtbarkeitsvolumens mit einer Kante betrachtet werden. Das Sichtbarkeitsvolumen ist mit schwarzen Linien dargestellt, der sichtbare Bereich einer Kante als grünes Liniensegment und der nicht sichtbare Bereich mit einem roten Liniensegment.

Hinweis: Sie müssen für alle Teilaufgaben auch die Herleitungen der verwendeten Algorithmen bei der Abnahme vorstellen können. Natürlich dürfen Sie dazu Notizen vorbereiten.