



## 1. Themenvorschläge

### **Virtual Reality Anwendung (z.B. Spiel)**

Mit dem Google Cardboard, einem einigermaßen aktuellen Smartphone und der Game-Engine Unity 3D lassen sich mit wenig Aufwand VR-Anwendungen entwickeln. Inhalt und Form dürfen nach Rücksprache frei gewählt werden. Hinweis: ein eigenes Android oder iOS Smartphone/Tablet ist erforderlich.

Beispiele: einfaches Computerspiel, Bewegung durch eine selbst generierte Szene (Inspiration: <https://cgi.tutsplus.com/tutorials/secrets-to-creating-low-poly-illustrations-in-blender--cg-31770>, abgerufen am 02.11.2016), eigener Vorschlag nach Absprache

### **Augmented Reality Anwendung (z.B. Spiel)**

Das AR Toolkit und OpenCV bieten Schnittstellen für Android und iOS-Geräte. Darauf basierende Ergebnisse eröffnen Möglichkeiten für ein breites Spektrum an Augmented Reality Anwendungen. Inhalt und Form dürfen nach Rücksprache frei gewählt werden. Hinweis: ein eigenes Android oder iOS Smartphone/Tablet ist erforderlich.

Beispiel: Darstellung eines 3D-Modells, Pose durch Marker bestimmt.

### **Partikelsysteme**

Partikelsysteme erlauben es, natürliche Phänomene wie Feuer zu simulieren.

Einstiegspunkt: <https://www.khanacademy.org/computing/computer-programming/programming-natural-simulations/programming-particle-systems/a/intro-to-particle-systems>, abgerufen am 02.11.2016

### **Prozedurale Generierung von Pflanzen**

Zur automatisierten Generierung von Pflanzen kann eine Grammatik (L-Regel-System) verwendet werden. Durch Ableiten von Regeln mittels des Systems lassen sich unterschiedliche Pflanzen generieren.

Einstiegspunkt: Przemysław Prusinkiewicz, Aristid Lindenmayer: The Algorithmic Beauty of Plants. Springer Verlag, New York NY u. a. 1990, ISBN 3-540-97297-8, (The virtual Laboratory).

### **BSP-Bäume zur Darstellung semi-transparenter Szenen**

BSP-Bäume erlauben es, eine Tiefensortierung einer Szene zu generieren. Das kann dazu verwendet werden, mit Transparenzen zu arbeiten, indem die Objekte in der Reihenfolge von hinten nach vorne gezeichnet werden. Möglicherweise in Kombination mit: <https://cgi.tutsplus.com/tutorials/secrets-to-creating-low-poly-illustrations-in-blender--cg-31770>, abgerufen am 02.11.2016),

Einstiegspunkt: Vorlesungsfolien zu BSP-Bäumen

### **Raytracing für Dreiecksnetze**

Raytracing ist im Kern ein einfaches Konzept, wird aber schnell sehr Rechenzeit-aufwändig. Insbesondere, wenn man Dreiecksnetze unterstützen will, dann muss man sich über Beschleunigungsdatenstrukturen Gedanken machen.

Einstiegspunkt: Vorlesungsfolien zu Raytracing, Strahl-Dreiecks-Schnitt, Datenstrukturen

### **Raytracing für Bezierflächen**

Raytracing ist im Kern ein einfaches Konzept, wird aber schnell sehr Rechenzeit-aufwändig. Bei CAD-Anwendungen werden häufig Bezierflächen verwendet, für die in einem Raytracing-System Schnittpunkte bestimmt werden müssen.

Einstiegspunkt: Vorlesungsfolien zu Raytracing, Strahl-Dreiecks-Schnitt, Datenstrukturen, Ansatzidee: <https://core.ac.uk/download/pdf/4891798.pdf>, abgerufen am 03.11.2016

## Darstellung von Volumendatensätzen

Volumendaten können durch die Überlagerungen von mehreren Texturschichten dargestellt werden. Die Texturen werden in der Reihenfolge von hinten nach vorne gezeichnet und miteinander überblendet.

Einstiegspunkt: Vorlesungsfolien zu Volumenvisualisierung (auf Anfrage) und

[http://http.developer.nvidia.com/GPUGems/gpugems\\_ch39.html](http://http.developer.nvidia.com/GPUGems/gpugems_ch39.html), abgerufen am 02.11.2016

## Mobile

Moderne Smartphones bieten seit einiger Zeit ausreichend Rechenleistung, um 3D-Anwendungen laufen lassen zu können. Da Android-Anwendungen (primär) in Java entwickelt werden, kann das Framework der Vorlesung übertragen werden. Anpassungen sind insbesondere bei der Steuerung (Touch statt Maus) notwendig. Als Ergebnis sollte ein funktionales Projekt stehen (z.B. ein Viewer für 3D Modelle oder  
Einstiegspunkt: <https://developer.android.com/guide/topics/graphics/opengl.html>, abgerufen am 02.11.2016

*Weitere Themen bei größerer Nachfrage möglich.*

## 2. Vorgehen

- bis 20.12.2016:
  - verbindliche Entscheidung, dass Abschlussprojekt als Prüfungsleistung verwendet werden soll
  - verbindliche Themenwahl
- bis 27.01.2017
  - Erarbeiten eines Exposees (1 Seite Din A4), Inhalt:  $\geq 5$  Basisquellen, konkretes Ziel (Ergebnis), notwendige Schritte zum Ziel
- in der letzten Woche der Semesterferien vor Beginn der Informatik-Vorlesungen (voraussichtlich KW 11, 2017)
  - Kurzvortrag ca. 10 Minuten mit Demo des Ergebnisses
  - Abgabe einer Kurzausarbeitung zum Thema (ca. 10 Seiten), Inhalt: Grundlagen (Literatur, theoretische Beschreibung des Ansatzes, praktische Umsetzung, Erfahrungen bei der Umsetzung, kritische Betrachtung des Ergebnisses)
  - Abgabe des Quellcodes (gerne per GitHub o.ä.)

## 3. Team-Bearbeitung

Es ist möglich, dass Sie im Team (zwei Personen) das Abschlussprojekt bearbeiten. In dem Fall muss der Umfang entsprechend größer sein. Außerdem ist eine sehr konkrete schriftliche Darstellung der Beiträge (Konzeption, Implementierung, Ausarbeitung) der Beteiligten notwendig.