

## Übungsblatt 7: Licht und Material

### Abgabe:

Dieses Übungsblatt ist einzeln zu lösen. Die Lösung ist bis **Donnerstag, den 12. Juni 2014, 12:00 Uhr s.t.** über UniWorx (<https://uniworx.ifl.lmu.de/>) abzugeben.

### Aufgabe 1: Licht und Material

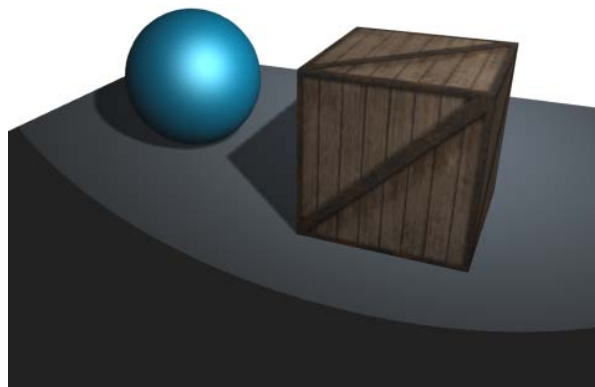
In dieser Aufgabe sollen Sie eine einfache Szene in three.js erstellen, die folgende Elemente beinhaltet:

- Eine ambiente Lichtquelle
- Ein Spotlight
- Einen texturierten Würfel
- Eine Kugel mit einem Material, das Glanzlichter darstellen kann
- Eine Ebene mit diffusem Material

Kugel und Würfel sollen zudem Schatten auf die Ebene werfen (siehe Beispiel Aufgabe 1).

Tipps:

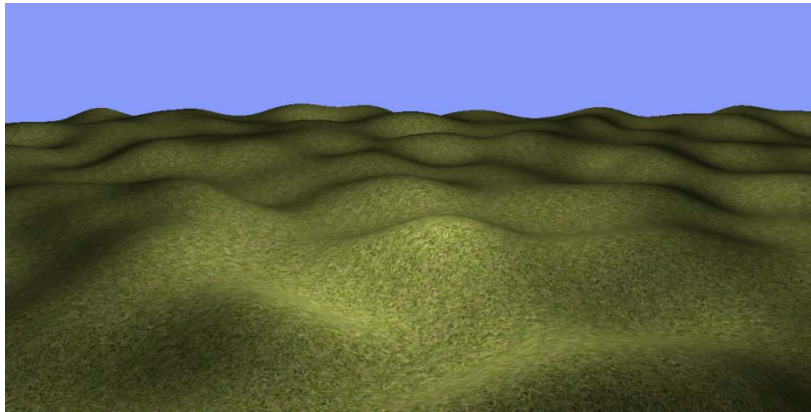
- Eine große Auswahl an freien Texturdateien finden Sie z.B. unter <http://www.cgtextures.com/> (Hinweis: das Format der Bilddateien spielt eine wichtige Rolle)
- Die Dokumentation von three.js kann sehr behilflich sein



*Abbildung 1* Beispielhafte Ansicht der Szene

## Aufgabe 2: Landschaft (Teil 2)

Auf der Vorlesungswebsite finden Sie die Datei terrain.html. Erweitern Sie das Programm, indem Sie Licht in die Szene einfügen und die Oberfläche der Landschaft beschreiben (siehe Beispiel in Abbildung 2). Definieren Sie ein Material und optional eine Textur für die Landschaft (Deaktivieren Sie zudem den Wireframe-Modus).



*Abbildung 2* Beispielsicht der Landschaft

## Aufgabe 3: Erste Schritte mit GLSL

Ziel dieser Aufgabe ist es, die Grundzüge der OpenGL Shading Language (GLSL)

kennenzulernen. Lesen Sie hierzu z.B. folgenden Artikel: <http://aerotwist.com/tutorials/an-introduction-to-shaders-part-1/>.

- Laden Sie die Datei start\_here.html von der Vorlesungswebsite herunter und öffnen Sie die Datei in einem Browser. Sie sollten einen Kreis sehen.
- Was machen Vertex- und Fragment-Shader? Was ist der Unterschied zwischen Variablen vom Typ attribute, uniform oder varying? Wofür werden sie jeweils eingesetzt? Was wird in den Variablen gl\_FragColor und gl\_Position gespeichert?
- Informieren Sie sich über die verschiedenen Variablentypen in GLSL (matX, vecX, etc.) und die verschiedenen vordefinierten Funktionen (z.B. dot, cross). Lesen Sie dazu z.B. dieses Tutorial: [http://wiki.delphigl.com/index.php/Tutorial\\_glsl](http://wiki.delphigl.com/index.php/Tutorial_glsl).
- Nehmen Sie jetzt das Codegerüst von der Homepage das den Kreis anzeigt. Verändern Sie den Fragment Shader so, dass der Kreis in Rot gezeichnet wird. Verändern Sie den Vertex Shader so, dass alle x-Koordinaten auf 0.5 Einheiten skaliert werden (ein Vertex an der Position (-2, 0, -1) soll also z.B. auf der Position (-1, 0, -1) landen).

*Viel Erfolg.*