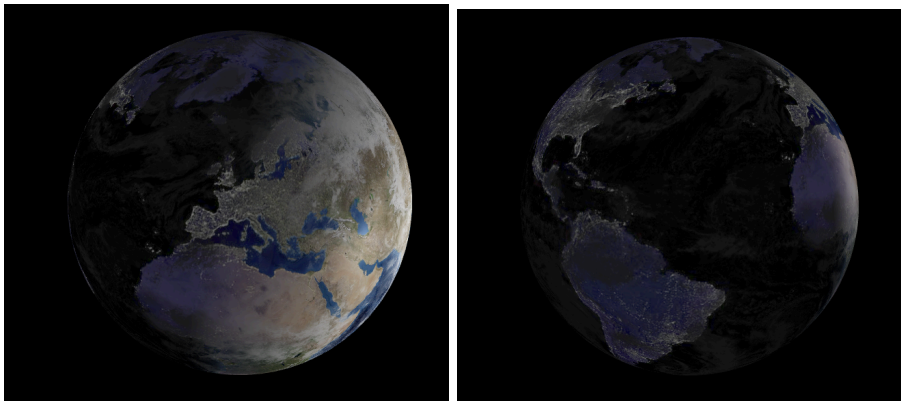
Prof. Dr.-Ing. Hartmut Schirmacher
Fachbereich Informatik und Medien

Computergrafik 2 / Aufgabe 3: *Weltanschauung*



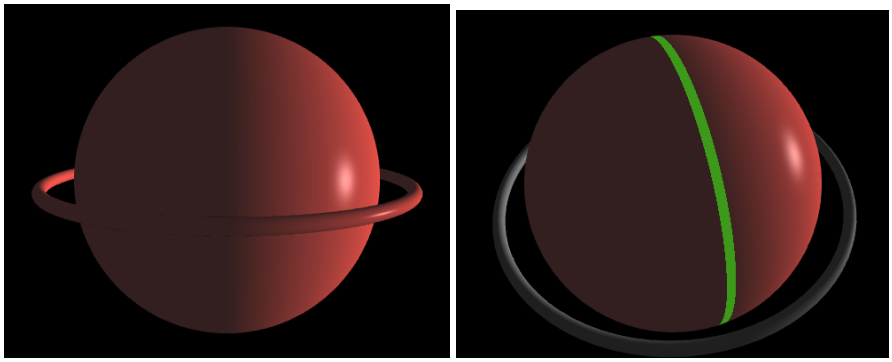
In dieser Aufgabe starten Sie mit einer Kugel, einer direktionalen Lichtquelle und einem Phong-Shader. Sie modifizieren den Shader so, dass sie einen Erdglobus unter beliebigen Sonnenständen und mit verschiedenen Oberflächeneigenschaften darstellen können. Dabei können Sie sich verschiedenster NASA-Bilder als Texturen bedienen: Topographie, Höheninformation, Wolkendichte etc.

Framework / Vorbereitung

Laden Sie das Framework zu Aufgabe 3 von Moodle herunter und legen Sie es so ab, dass der Ordner auf der gleichen Verzeichnisebene wie das bereit vorhandene Verzeichnis `lib/` liegt. Es handelt sich um eine erweiterte Version des Frameworks von Aufgabe 2 mit Licht, Material, Texturen und Phong-Shading. Shader werden nun via `shaders.js` aus separaten Dateien in dem Unterverzeichnis `shaders/` geladen, was die Bearbeitung der Shader deutlich vereinfacht. Der `SceneNode` wurde um die Berechnung der Normalen-Matrix erweitert. In `models/` findet sich eine Datei `parametric.js`, mit welcher beliebige parametrische Flächen erzeugt werden können; als Beispiel sind ein Torus und eine Kugel implementiert. Diese Modelle stellen die folgenden Vertex-Attribute bereit: Position, Normalenrichtung, Texturkoordinaten. Das Modul `scene_explorer.js` wurde integriert, und erlaubt die interaktive Navigation der Szene mittels der Maus (Drag = Rotation; Shift-Drag = Translation; Alt-Drag oder Wheel = Zoom). Wenn die Maus hakt, hilf i.d.R. die Escape-Taste.

Neben geometrischen Objekten kann der Szenengraph nun auch Materialknoten enthalten (siehe `material.js` und Handout). Diese Knoten setzen in Ihrer `draw()`-Methode lediglich einige globale Variablen, die vom Phong-Shader erwartet werden. So setzt ein Phong-Material z.B. die `uniform`-Variablen `material.ambient` und `material.diffuse`, die der Phong-Shader dann zur Beleuchtungsberechnung verwendet. Die Materialdefinition gilt für alle Objekte, die im Szenengraphen nach dem Material gezeichnet werden, so lange bis ein anderes Material oder ein anderer Shader zum Einsatz kommen.

Aufgabe 3.1: Phong-Shader und Tag-Nachtgrenze

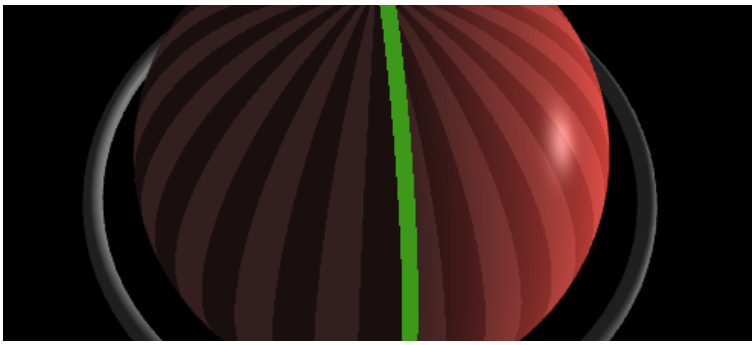


Die erste Aufgabe dient der Einarbeitung und soll helfen, den Phong-Shader sowie den Zusammenhang zwischen JS-Code, Vertex-Shader und Fragment-Shader zu verstehen.

Wenn Sie die `index.html` anschauen, sehen Sie einen roten Planeten mit einem roten Ring (Torus). Die Szene wird von einer "Sonne" beleuchtet. Das Einschalten der Animation bewirkt, dass diese Lichtquelle um die Szene herum rotiert. Außerdem können Sie die Szene wie gewohnt mit den Tasten X und Y manipulieren, und Sie können wie oben beschrieben die Maus verwenden, um die Szene zu rotieren, zu verschieben und zu zoomen.

- Planet und Ring verwenden das gleiche Material. Legen Sie stattdessen für den Ring ein neues Materialobjekt an und weisen Sie ihm eine andere Farbe als dem Planeten zu.
- Die Beleuchtungsberechnung auf dem Planeten soll über das Phong-Modell hinausgehen. Kopieren Sie die beiden Phong-Shader-Dateien, nennen Sie die Kopien `planet.vs` und `planet.fs`, fügen Sie diese zur Liste in `shaders.js` hinzu. Verwenden Sie in Ihrer Szene ab sofort ein neues `planet`-Programm für den Planeten und das alte Phong-Programm für den Ring.
- Damit die Beleuchtung auch auf dem Planeten funktioniert, müssen Sie dem Licht bei seiner Erzeugung mitteilen, dass es auch in ihrem neuen `planet`-Programm wirken soll.
- Fügen Sie der Szene eine `drawOption` "Debug" hinzu, und reichen Sie den Wert dieser Option über eine geeignete uniform-Variable an die Shader im `planet`-Programm weiter.
- Visualisieren Sie auf dem Planeten den Bereich, in welchem der Übergang zwischen Tag und Nacht stattfindet. Wenn der "Debug-Schalter" an ist, malen Sie im `planet`-Shader diejenigen Fragmente grün, bei denen der Winkel zwischen Sonne und Oberfläche zwischen 0° und 3° beträgt. Alle für die Berechnung benötigten Vektoren liegen bereits im Phong-Shader vor.
- Vergewissern Sie sich, dass die Tag-Nacht-Linie wie gewünscht mit der Sonne wandert, wenn die Animation eingeschaltet ist.

Aufgabe 3.2: Texturkoordinaten



Die Modelle `Sphere()` und `Torus()` in `models/parametric.js` stellen pro Vertex die 3D-Position, die Normalenrichtung sowie 2D-Texturkoordinaten zu Verfügung. Letztere sollen nun im Fragment-Shader verwendet und zunächst visuell überprüft werden.

- Sorgen Sie dafür, dass die Texturkoordinaten, die `Sphere()` unter dem Namen `vertexTexCoords` bereitstellt, vom Planet-Vertex-Shader entgegengenommen und an den Planet-Fragment-Shader unter einem Namen wie z.B. `texCoords` weitergereicht werden (siehe Handout).
- Die erzeugten Texturkoordinaten liegen im Intervall [0 ... 1]. Visualisieren Sie den Verlauf der ersten Texturkoordinate (z.B. `texCoords.s`) im Debug-Modus, indem Sie immer abwechselnd einen Streifen von Fragmenten dunkler machen und einen Streifen unangetastet lassen. Am einfachsten multiplizieren Sie den ambienten und den diffusen Term um einen "Debugging-Abdunkel-Faktor", der abhängig von der Texturkoordinate berechnet wird.
- Tipp: Modulo ist Ihr Freund und heisst in der Shading Language `mod()`.
- Vergewissern Sie sich, dass bei Einschalten der Animation die Tag-Nacht-Linie wandert, aber die Texturstreifen stehen bleiben

Aufgabe 3.3: Tag- und Nachttextur

Nun soll das Erscheinungsbild des Planeten mit Texturen aufgewertet werden. Siehe auch Beispiel-Bilder auf der nächsten Seite.

- Bedienen Sie sich im Konstruktor der Szene der Funktion `texture.Texture2D`, um `textures/earth_month04.jpg` als Textur zu laden.
- Binden Sie die Textur in Ihrem `planet`-Programm mittels der Methode `setTexture()` an die Textureinheit 0 und z.B. an den `uniform`-Bezeichner "`daylightTexture`". Diesen Schritt sollten Sie nur dann ausführen, wenn die Textur auch wirklich geladen wurde (`onAllTexturesLoaded()`, siehe SU-Folien).
- Greifen Sie nun im Planet-Fragment-Shader auf die Textur zu. Erweitern Sie den Shader so, dass Sie den Wert der Textur anstelle des diffusen Material-Koeffizienten verwenden. Ersetzen Sie nicht den gesamten diffusen Term! Trotz der Werte aus der Textur soll die Abdunkelung zum Abend immer noch gut zu erkennen sein. Hellen Sie die Werte aus der Textur ggf. nach Geschmack auf (z.B. mittels * und/oder `pow()`).
- Führen Sie eine Checkbox "Daytime Texture" ein, welche es ermöglicht, zwischen der texturierten Darstellung der Tagseite und der roten Phong-Darstellung zu wechseln.
- Da die Textur nur in den diffusen Term eingeht, wie sie automatisch auf der „Tagseite“ der Erde angezeigt, während aufgrund des Einfallswinkels bei Phong die Nachtseite der Erdkugel nur mit der ambienten Farbe dargestellt wird. Verändern Sie die Beleuchtungsfunktion so, dass auf der Nachtseite der Erde die Textur `earth_at_night_2048.jpg` angezeigt wird. Interpolieren Sie dazu auf geeignete Weise zwischen Tag- und Nachttextur.
    - Jenseits der Tag-Nachtgrenze sollten nur die Lichter zu sehen sein (nur ambienter Term); dort wo die Sonne steil einfällt (mittags) sollen die Lichter nicht zu sehen sein (ambienter Term = 0).
    - In der Dämmerung nahe der Tag-Nacht-Linie sollte man sehen, dass die Menschen am Anfang/Ende des Tages das Licht einschalten. Der Übergang von Tag und Nacht sollte glatt ohne sichtbaren "Bruch" erfolgen.
- Führen Sie eine Checkbox "Night Lights" ein, welche es ermöglicht, die zusätzliche Nachttextur ein- und auszuschalten. Wenn Sie ausgeschaltet ist, sollte die "Rückseite" der Erde schwarz oder zumindest sehr dunkel sein. Verwenden Sie den Schalter, um sicherzustellen, dass in der Mittags-Zeit auf der Erde keine Lichter angehen.
- Trotz der nachfolgenden Veränderungen soll der Debugging-Modus nach wie vor die Tag-Nacht-Linie zeichnen, damit Sie die Sinnhaftigkeit der Überblendung prüfen können.

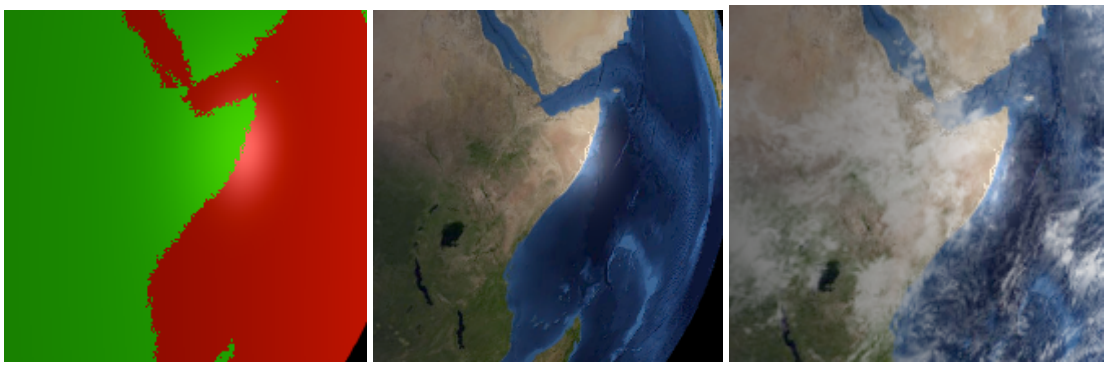
ohne Texturen, Tag-Nacht-Grenze im Debug-Modus

nur Tag-Textur (diffuser Term)

Nur Nacht-Textur, man sieht die Überblendung auf der Tagseite

Tag-und Nachttextur mit Überblendung

Aufgabe 3.4 (optional, nur für eine sehr gute Note): Wasser und Wolken



Durch die Natur des Phong-Shaders sieht die Erdkugel noch eher wie eine Plastik-Kugel aus, da Erde und Wasser gleichermaßen spekular reflektieren. Es erscheint jedoch realistischer, wenn Wasser stärker spiegelt als die Landflächen.

- Führen Sie zunächst einen Schalter "Red/Green" ein, der bewirkt, dass sowohl auf Tag- als auch auf Nachtseite Landfragmente grün und Wasserfragmente rot gezeichnet werden.
- Verwenden Sie dazu z.B. die Information aus der topografischen oder bathymetrischen NASA-Textur, die die Höhen- bzw. Tiefeninformation kodiert. Ermitteln Sie, welche Texturwerte welcher Höhe entsprechen, und setzen Sie im Shader abhängig von dieser Information die Farbe.
- Fügen Sie einen Schalter "Gloss Map" hinzu, der bewirkt, dass sich der spekulare Term Ihrer Beleuchtungsfunktion abhängig von Wasser vs. Land verändert. Manipulieren Sie wahlweise den spekularen Materialkoeffizienten, den Phong-Exponenten, oder beides.
- Die bisher verwendeten Texturen der Erde sind wolkenfrei. Fügen Sie der Erde abhängig von einer Checkbox "Clouds" eine Wolkenschicht auf Basis der Textur `earth_clouds_2048.jpg` hinzu. Der Rot-Kanal der Textur kann wie eine "Wolkendichte" interpretiert werden. Je dichter die Wolken sind, desto mehr sollte die Wolkenfarbe die Erdfarbe überdecken (Interpolation). Die Wolken sollten auch eine sinnvolle Auswirkung auf die Sichtbarkeit der Lichter in der Nacht haben (Abdunklung).

Abgabe

Die Abgabe soll via Moodle bis zu dem dort angegebenen Termin erfolgen. Verspätete Abgaben werden wie in den Handouts beschrieben mit einem Abschlag von 2/3-Note je angefangener Woche Verspätung belegt. Geben Sie bitte pro Gruppe jeweils nur eine einzige `.zip`-Datei mit den Quellen Ihrer Lösung ab (das Texturverzeichnis können Sie dabei auslassen).

Demonstrieren und erläutern Sie dem Übungsleiter Ihre Lösung *in der nächsten Übung nach dem Abgabetag*. Die Qualität Ihrer Demonstration ist, neben dem abgegebenen Code, ausschlaggebend für die Bewertung! Es wird erwartet, dass alle Mitglieder einer Gruppe anwesend sind und Fragen beantworten können.