

Computergrafik 2 / Aufgabe 2.1

Parametrische Flächen

WiSe 2015/2016

Bei dieser Aufgabe erzeugen Sie eine parametrische 3D Oberfläche auf Basis einer Punktwolke. Zudem üben Sie die Darstellung allgemeiner parametrischer Oberflächen und stellen diese mit Hilfe des WebGL und Three.js Übungsframeworks dar.

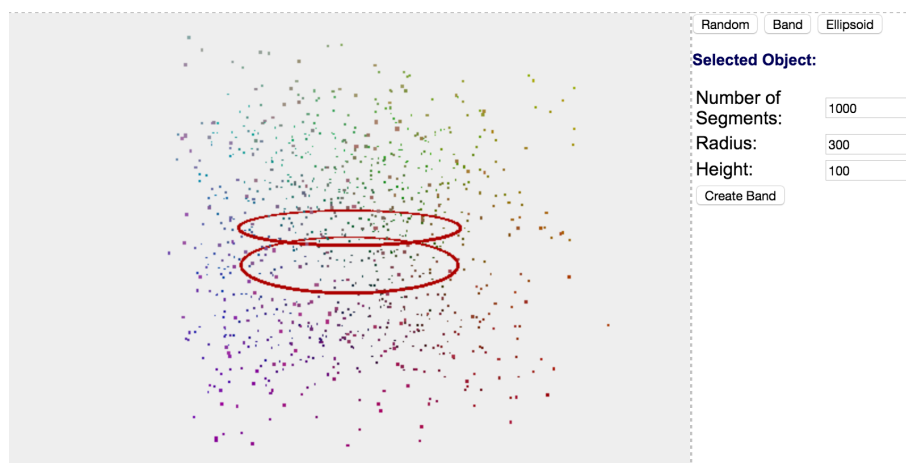


Abbildung 1: Beispielhafte aber unvollständige Ergebnisausgabe

Aufgabe 1: Einarbeitung in *three.js*

Für die Bearbeitung der folgenden Übungsaufgaben starten wir mit einem neuen Übungsframework. Dieses ist ähnlich strukturiert wie das Vorherige. Bitte lesen Sie sich in den vorgegebenen Sourcecode ein. Wir werden ab jetzt mit dem WebGL Framework *three.js* arbeiten. Machen Sie sich mit dem Framework vertraut:

- Beachten Sie, dass sich ab jetzt im `./lib` Verzeichnis zusätzlich noch die Datei *three.min.js* befindet. Sie können also nicht das vorherige Verzeichnis wiederverwenden bzw. müssen die fehlende Datei dort hinein kopieren.
- Lesen Sie die Dokumentation auf <http://threejs.org/>, probieren Sie die Beispiele und schauen Sie sich den Sourcecode dazu an.
- Viele Dinge werden Ihnen unbekannt sein und sich erst im Laufe der folgenden Vorlesungen erschließen. **Geben Sie bitte bis 18.11. im Moodle eine Textdatei pro Gruppe ab, die drei sinnvolle Fragen enthält, die Ihnen im Zusammenhang mit dem Vorlesungsstoff und dem Framework *three.js* einfallen.** Diese sollten sich auf die Umsetzung der Grafikpipeline-Konzepte und wie diese in *three.js* umgesetzt sind beziehen. Bei einer Nichtabgabe werden Ihnen 0.3 Notepunkte auf die nächste Übungsnote abgezogen.

Hinweis: *Three.js* bietet Ihnen weitreichende Möglichkeiten, diese und die folgenden Aufgaben z.B. mit Hilfe von anderen Shadern oder Materialien besser und umfangreicher zu gestalten. Nutzen Sie diese

Möglichkeiten und lassen Ihrer Kreativität freien Lauf. Setzen Sie jedoch nur Sachen ein, die Sie auch verstanden haben.

Aufgabe 2: Parametrische Flächen - Ellipsoid als 3D Punktwolke darstellen

Wir konzentrieren uns zunächst nur auf das Erzeugen einer parametrischen Geometrie. Bitte beachten Sie, dass Sie die Dateien über einen lokalen Webserver abzurufen und nicht direkt aus dem Filesystem (siehe Warmup-Aufgabe).

Stellen Sie `index.html` im Browser dar, erzeugen Sie ein zufällige Punktwolke oder das Band-Objekt und drücken Sie die Pfeiltasten.

Studieren Sie das Modul `models/band.js`. Es besteht aus

- einem Konstruktor, der ein Array mit den Koordinaten von 3D-Punkten füllt. Aus diesem Array wird dann eine `BufferGeometry` (in OpenGL auf Vertex Buffer Object (VBO)) erzeugt;
- zwei getter Methoden, die jeweils Farbe oder Position zurückgeben.

Studieren Sie das Modul `scene.js`. Auch die Szene besteht im wesentlichen aus einem Konstruktor, einem Keyboard-Handler und einer `draw()`-Funktion. Im Konstruktor werden Kamera und Szenenobjekte angelegt, und in der `draw()`-Methode werden diese dann immer wieder gezeichnet. Mit den Pfeiltasten können Sie das aktuell erzeugte Objekt rotieren.

Im `HtmlController` sollen Objekte angelegt werden, ähnlich wie im vorherigen Framework. Dort können im Moment eine zufällige Punktwolke, das Band-Objekt, sowie ein `ParametricSurface`-Objekt angelegt werden.

Der Konstruktor des `ParametricSurface`-Objekts erhält dabei eine Funktion, welche zu einem gegebenen Paar (u,v) Werten jeweils ein Array mit drei Koordinaten $[x,y,z]$ zurückliefert. Die übergebene Funktion definiert ein Ellipsoid, siehe z.B. <http://en.wikipedia.org/wiki/Ellipsoid>. Als weiteres Argument übergibt die Szene dem `ParametricSurface`-Konstruktor ein `config`-Objekt mit den Wertebereichen der Parameter u und v (`umin`, `umax`, `vmin`, `vmax`) sowie der gewünschten Anzahl von Segmenten in u - und v -Richtung.

Implementieren Sie nun das Modul `models/parametric.js`, welches Sie bereits als Skelett vorfinden. Orientieren Sie sich dabei an `band.js`. Werten Sie auf dem angegebenen u - v -Gitter die Fläche durch Aufruf der angegebenen Funktion aus, füllen Sie ein Array `positions` mit diesen Punkten und weisen jedem Punkt eine Farbe zu (`colors`). Implementieren Sie alle weiteren notwendigen Dinge, um das Objekt über die `index.html` anlegen und konfigurieren können.

Implementieren Sie eine Animations-Funktion, die das aktuell ausgewählte Objekt in einer bestimmten Geschwindigkeit rotieren lässt (Checkbox mit Animation ein/aus).

Zusatzaufgabe (nur für das Erreichen einer 1.0) :

Verwenden Sie Ihre `ParametricSurface`, um noch zwei weitere parametrische Flächen zur Szene hinzuzufügen, z.B. einen Torus und eine beliebige weitere Fläche (siehe z.B. <http://www.3d-meier.de/tut3/Seite0.html>)

Abgabe Dies ist *Teilaufgabe 2.1*; die Bearbeitungszeit der Teilaufgabe ist für ca. eine Woche ausgelegt. Die Abgabe der gesamten Aufgabe 2 soll via Moodle bis zu dem dort angegebenen Termin erfolgen. Verspätete Abgaben werden 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.

Demonstrieren und erläutern Sie 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.