

Übung zu Computergraphik II

– Übungsblatt 9 –

Lehrstuhl für Computergraphik
und Multimediasysteme

Peter Marchel, Julian Bader

Hinweis: Für die folgenden Programmieraufgaben benötigen Sie die GLEW-Bibliothek, die Sie unter <http://glew.sourceforge.net> herunterladen können. Damit CMake diese Bibliothek finden kann, muss zudem noch die Umgebungsvariable 'GLEW_DIR' auf Ihrem System angelegt sein (analog zu 'GLUT_DIR'). Zusätzlich müssen Sie noch Ihre PATH Umgebungsvariable auf den bin-Ordner von der GLEW-Bibliothek erweitern, sodass die dynamische Library (glew32.dll) zur Laufzeit gefunden wird.

Aufgabe 1 [1 Punkt] Vertex Shader - Bezier-Kurven

Ziel dieser Aufgabe ist es eine kubische Bezierkurve auf der GPU zu implementieren. Laden Sie hierzu das Programmgerüst (ueb9_1.zip) herunter. Innerhalb der Datei main.cpp finden sie in der Funktion `drawCPUCurve()` bereits die Implementierung für die CPU (linke Kurve). Die Implementierung für die GPU (rechte Kurve) ist dagegen unvollständig:

- Überlegen Sie sich zunächst wie die Datenübertragung zwischen CPU und GPU aussehen kann. Ändern Sie dann den `glVertex3f`-Aufruf in der Funktion `drawGPUCurve()` so ab, dass für jeden Vertex der entsprechende Kurvenparameter übertragen wird.
- Wechseln Sie nun in den Vertex-Shader (`vs_bezier_curve.glsl`). Verschieben Sie nun jeden einzelnen Vertex auf die entsprechende Position auf der Bezierkurve.

Das Ergebnis sollte wie in der folgenden Abbildung aussehen.

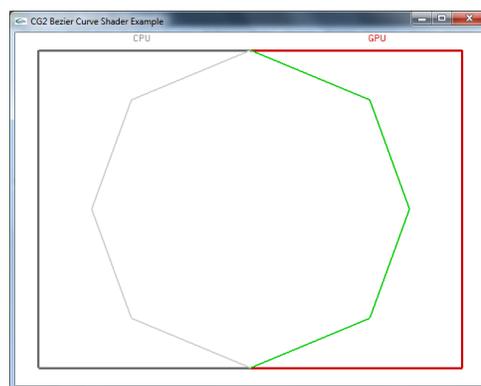


Abbildung 1: Ergebnisbild der Bezier-Kurven

Aufgabe 2 [1 Punkt] Fragment Shader - Prozedurale Textur

Das Ziel dieser Übungsaufgabe ist es, eine prozedurale Textur im Fragment-Shader zu erzeugen. Diese prozedurale Textur soll hierbei ein Kreuz enthalten, dessen Größe durch die Variable `g_crossSize` dynamisch mittels der Tasten +/- verändert werden kann. Zudem soll die Textur mit dem diffusen Anteil des Phong-Modells beleuchtet werden (vergleiche Abb. 2). Laden Sie hierzu zunächst das Programmgerüst (ueb9_2.zip) herunter und vervollständigen Sie anschließend den Shader `fs_cross.glsl`.

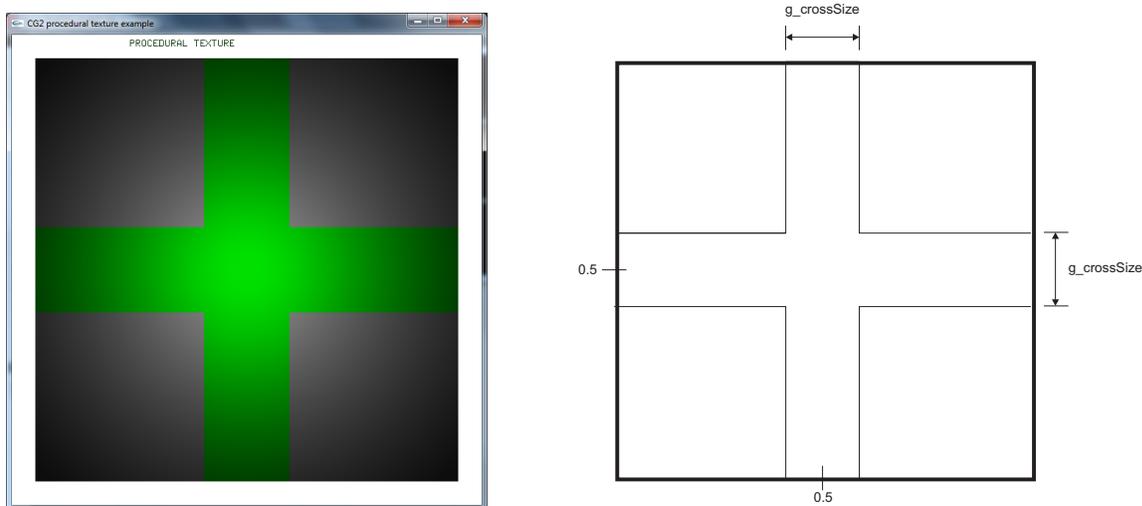


Abbildung 2: Die prozedurale Kreuz-Textur (links) und zugehöriges Schema (rechts).

Die Parameter (`g_crossSize` und `g_crossColor`) sind dabei bereits vordefiniert.

- Zunächst versuchen Sie bitte das prozedurale Kreuz zu erstellen. Hierzu soll die Variable `crossFactor` genau dann dem Wert 1 entsprechen, wenn die zugehörige Textur-Koordinate in x- und y-Richtung innerhalb des Kreuzes liegt. Andernfalls soll 0 gesetzt werden. Die 2D Textur-Koordinaten `v_texcoord` liegen dabei im Bereich $[0, 1]^2$ vor. Um die Variable zu setzen können Sie die builtin-Funktion `step` verwenden.

$$\text{step}(\text{edge}, x) = \begin{cases} 1 & \text{if } x > \text{edge} \\ 0 & \text{else} \end{cases} \quad (1)$$

- Weitergehend soll nun der diffuse Anteil des Phong-Modells für die Variable `diffuse` berechnet werden.

$$d_{\text{phong}} = \max(0, \hat{n} \cdot \hat{l}) \quad (2)$$

Hierzu stehen Ihnen die Variablen `v_normal` und `v_lightDir` zur Verfügung.

Abgabe: 11.12.2012, zu Beginn der Übung oder bis 8:30 Uhr im Postkasten des Lehrstuhls (gegenüber Raum H-A 7107)

Senden Sie uns für Aufgabe 1 und 2 Ihre Lösungen zu.

→ Email an: peter.marchel@student.uni-siegen.de **und** julian.bader@uni-siegen.de

E-Mails werden nur bis Dienstag 8:30 Uhr angenommen. Wenn wir Ihre Lösung der Aufgaben 1 und 2 per Email erhalten haben, senden wir Ihnen so bald wie möglich eine Bestätigung.