



AUFGABE 1: Texturierung

Unter http://www.cg.tu-berlin.de/cg1_07.html steht ein C Programmskelett zur Verfügung, das zu ergänzen ist:

- a) Modelle in Form von Dreiecksnetzen sollen geladen und angezeigt werden können. Als Modellformat soll OFF verwendet werden, sie können also ihren Loader vom dritten Übungsblatt weiterverwenden. Legen sie die Modellinformation in einer geeigneten Datenstruktur ab und generieren Sie aus den Knoten Normalen für jede Facette und zusätzlich für jeden Eckpunkt das normierte arithmetische Mittel der anliegenden Flächennormalen, um von der Smooth-Shading Funktionalität Gebrauch machen zu können. (1 Punkt)
- b) Nun sollen mittels des Zweischritt-Kugelmappings Texturkoordinaten für die Eckpunkte berechnet werden. Implementieren sie dieses in `calculateTextureCoordinates(...)`. Erweitern Sie das Programm, so dass die Dreiecke auf Wunsch texturiert angezeigt werden. Die Texturdaten befinden sich an der Adresse `_textureImage` und haben die Ausmaße `_texX`, `_texY`. (2 Punkt)
- c) Stellen Sie Funktionalität zum Editieren der Textur zur Verfügung. Die Modifikation soll interaktiv auf der Textur sichtbar werden. Als Mindestfunktionalität sollen das Setzen von Punkten und das Löschen einer Textur ermöglicht werden. Es kann dabei von der Funktion `setPointInTexture(...)` Gebrauch gemacht werden. Alternativ dazu kann auch der Framebuffer ausgelesen werden. Um die neue Textur an das graphische System zu übergeben stehen `glTexImage2D(...)`, `glTexSubImage2D(...)`, `glCopyTexSubImage2D(...)` zur Verfügung. Mittels `glTexSubImage2D(...)` muss nur der veränderte Ausschnitt der Textur übermittelt werden und `glCopyTexSubImage2D(...)` kopiert einen Ausschnitt direkt aus dem Framebuffer. Implementieren sie die Editierfunktionalität in `textureMouse(...)`, `textureMotion(...)` und `texturePassiveMotion(...)`. Um dem Benutzer die Orientierung zu erleichtern, soll in der 3D Ansicht zusätzlich ein Cursor in Form einer Linie angezeigt werden. Implementieren und verwenden Sie die Funktion `updatePointer(...)`. (2 Punkte)

AUFGABE 2: Theoriefragen

1. Da die Rasterisierung eines texturierten Dreiecks im (2D) Bildraum durchgeführt wird, kann es zu Abbildungsfehlern kommen wenn zwischen den Eckpunkt-Texturkoordinaten linear interpoliert wird. Beschreiben Sie (in Worten) wie und wann dieser Fehler entsteht. (1 Punkt)
2. Unter welchen Umständen ist der Einsatz einer MipMap zur Texturierung sinnvoll? (1 Punkt)
3. Die Zweischritt-Verfahren zur Bestimmung von Texturkoordinaten bilden das Modell zumeist nicht bijektiv auf den umschreibenden Körper ab. Beschreiben Sie die Familie von Objekten, die sich durch Zentralprojektion bijektiv auf einen Kugel abbilden lassen! (1 Punkt)
4. Meist wird für die Rasterisierung von texturierten Dreiecken (pro Pixel) bilinear zwischen den vier benachbarten Texeln interpoliert (siehe Vorlesungsfolien). Dieser Vorgang stellt eine *diskrete Faltung* im Ortsraum dar, und zwar mit einem bilinearen Filter (\wedge). Wie sieht der optimale Filter für die diskrete Faltung im Ortsraum aus? Begründen Sie Ihre Antwort. (Tip: Mit welchem Filter wird, unter der Voraussetzung optimaler Abtastung, im *Frequenzraum* multipliziert, um eine einzige Kopie des punktabgetasteten Signals zu erhalten?) (1 Punkt)
5. Warum kommt es bei der Parametrisierung einer Environment Map in planaren (u,v) Koordinaten zwangsläufig zu Verzerrungen? (es kommt auch zu Singularitäten, danach ist aber nicht gefragt) (1 Punkt)
6. Wie könnte man mit Hilfe von Environment Mapping eine interaktive Darstellung der Silhouetten eines Objektes erreichen? (1 Punkt)