Dokumentation zum OpenGL-Praktikum „Swimmingpool“

Tino Schwarze

18. Februar 2000

1 Überblick über das Modell

Das Modell besteht aus einer Grundfläche („Wiese“), den Swimmingpoolwänden und der Wasseroberfläche (siehe Abb. 1).

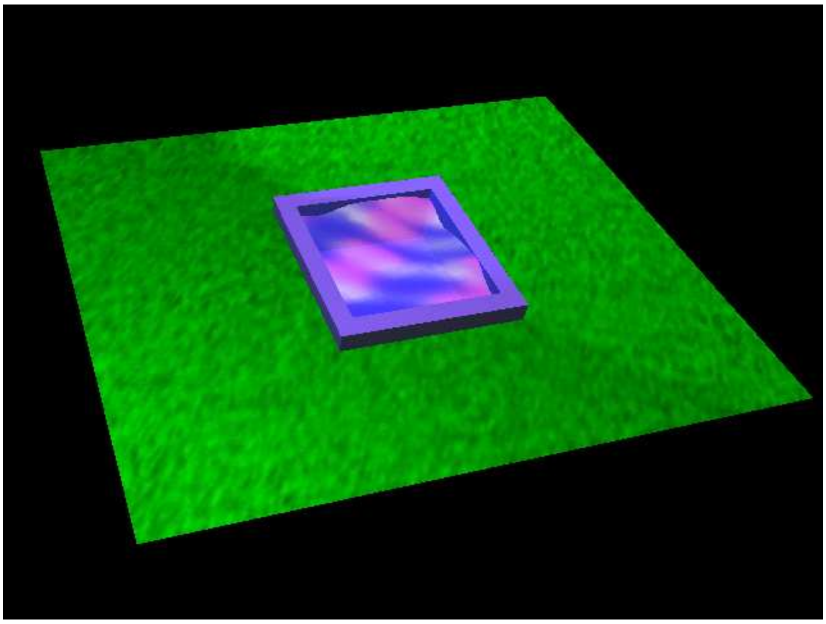

Abbildung 1: Screenshot des gerenderten Modells

2 Koordinatensysteme / Transformationen

Das Modell ist dreistufig aufgebaut (im Szenegraphen des Menüs sind mehr Stufen, weil Lichtquellen und Materialien jeweils eine eigene Stufe beanspruchen).

Für die Repräsentation der Rotationen werden Quaternionen verwendet (siehe `cQuaternion.hh`). Die Anwendung der Transformationen (Rotation, Verschiebung) erfolgt dann in `cVisibleObject::Transform ()` (siehe `cVisibleObject.cc`, ca. Zeile 160) - es wird eine Transformationsmatrix errechnet und mit der derzeitigen OpenGL Modelview-Matrix multipliziert.

Der Pool setzt sich aus Außen- und Innenwand zusammen. An die Außenwand schließt sich die Deckfläche und daran die Wasseroberfläche an (siehe Abb. 2). Als Primitive kommen vorwiegend die selbstimplementierte Ebene (`cPlane`) und „Ebene mit Loch“ (`cHoledPlane`) zum Einsatz, die wiederum `GL_TRIANGLE_STRIP` verwenden. Weiterhin werden für die Punktlichtquelle `gluSphere` und für die Spotlichtquelle `glutSphere` eingesetzt.

Die Wasseroberfläche, die in `cWater` implementiert ist, wird ebenfalls mittels `GL_TRIANGLE_STRIP` erzeugt.

Die verwendeten Transformationen sind auch in Abb. 2 verzeichnet.

3 Material- und Lichtdefinitionen

Materialien und Lichtquellen wurden jeweils in eigene Klassen gekapselt (`cMaterial`, `cTextureMaterial`, `cLight`, `cDirectionLight`, `cSpotLight`). Deshalb sind die einzelnen OpenGL-Aufrufe nicht direkt zu sehen. (siehe `cLight.cc`, Zeilen 140 ff. und `cMaterial.cc`, Zeilen 46 ff.)

- gerichtetes Licht

Komponente	Rot	Grün	Blau
ambient	0.1	0.1	0.1
diffus	1.0	0.2	0.2
spekular	1.0	0.2	0.2

- Spotlicht

Komponente	Rot	Grün	Blau
ambient	0.0	0.0	0.0
diffus	0.8	0.8	1.0
spekular	0.8	0.8	1.0

Wiese (`cHoledPlane`) - keine Translation oder Rotation

- **Poolaußenboden (`cPlane`)**
  Transl.: $(0, 0, -\texttt{POOL\_DEPTH\_Z})$
  Rot.: 180° um x-Achse
  - **Vorderseite (`cPlane`)**
    Transl.: $(0, -\texttt{POOL\_SIZE\_Y}/2, -\texttt{POOL\_SIZE\_Z}/2)$
    Rot.: 90° um x-Achse
  - **Rückseite (`cPlane`)**
    Transl.: $(0, \texttt{POOL\_SIZE\_Y}/2, -\texttt{POOL\_SIZE\_Z}/2)$
    Rot.: -90° um x-Achse
  - **linke Seite (`cPlane`)**
    Transl.: $(-\texttt{POOL\_SIZE\_X}/2, 0, -\texttt{POOL\_SIZE\_Z}/2)$
    Rot.: -90° um y-Achse
  - **rechte Seite (`cPlane`)**
    Transl.: $(\texttt{POOL\_SIZE\_X}/2, 0, -\texttt{POOL\_SIZE\_Z}/2)$
    Rot.: 90° um y-Achse
  - **Deckfläche (`cHoledPlane`)**
    Transl.: $(0, 0, -\texttt{POOL\_SIZE\_Z})$
    Rot.: 180° um x-Achse
    - **Wasseroberfläche (`cWater`)**
      Transl.: $(0, 0, -(\text{max. Amplitude}))$
      Rot.: -
- **Poolinnenboden (`cPlane`)**
  Transl.: $(0, 0, \texttt{POOL\_IN\_POS\_Z})$
  Rot.: -
  - **Vorderseite (`cPlane`)**
    Transl.: $(0, \texttt{POOL\_IN\_SIZE\_Y}/2, \texttt{POOL\_IN\_SIZE\_Z}/2)$
    Rot.: 90° um x-Achse
  - **Rückseite (`cPlane`)**
    Transl.: $(0, -\texttt{POOL\_IN\_SIZE\_Y}/2, \texttt{POOL\_IN\_SIZE\_Z}/2)$
    Rot.: -90° um x-Achse
  - **linke Seite (`cPlane`)**
    Transl.: $(\texttt{POOL\_IN\_SIZE\_X}/2, 0, \texttt{POOL\_IN\_SIZE\_Z}/2)$
    Rot.: -90° um y-Achse
  - **rechte Seite (`cPlane`)**
    Transl.: $(-\texttt{POOL\_IN\_SIZE\_X}/2, 0, \texttt{POOL\_IN\_SIZE\_Z}/2)$
    Rot.: 90° um y-Achse

Abbildung 2: Szenegraph mit Transformationen

- Punktlicht

Komponente	Rot	Grün	Blau
ambient	0.0	0.0	0.0
diffus	1.0	1.0	1.0
spekular	1.0	1.0	1.0

- Wiese

  Die „Wiese“ ist texturiert, wobei die Texturparameter je nach Qualitätseinstellung variieren (siehe `finalpool.cc`, Zeilen 680 ff.). Als Textur wird die selbst erstellte Datei `wiese2.jpg` verwendet.

- Poolaußenseiten

Komponente	Rot	Grün	Blau
ambient	0.1	0.1	0.2
diffus	0.2	0.2	0.7
spekular	0.9	0.9	0.9

Shininess: 32

- Poolinnenseiten

Komponente	Rot	Grün	Blau
ambient	0.1	0.1	0.3
diffus	0.1	0.1	0.8
spekular	1.0	1.0	1.0

Shininess: 48

- Wasseroberfläche

  Je nach Einstellung wird die Wasseroberfläche entweder mit einem Material oder einer Environment-Map versehen. Die Wasseroberfläche ist in jedem Fall transparent. Die Environment-Map wird so berechnet, daß in etwa die gleichen Farbtöne wie bei normaler Schattierung erzeugt werden.

  Die zugrunde liegende Materialdefinition:

Komponente	Rot	Grün	Blau	Transparenz
ambient	0.1	0.1	0.3	0.7
diffus	0.1	0.1	0.9	0.7
spekular	1.0	1.0	1.0	0.7

Shininess: 16

4 Interaktion

Es gibt zwei Arten von Interaktion: globale und lokale. Globale Interkation ist immer verfügbar während sich lokale Interaktion nur auf das selektierte Objekt bezieht.

4.1 Überblick

Eine Maus mit drei Knöpfen ist zu empfehlen. Mausknopf Nummer 1 entspricht dem linken Knopf, Nummer 2 dem mittleren und Nummer 3 dem rechten. Der rechte Mausknopf ist mit dem Menü belegt.

Interaktion erfolgt grundsätzlich mit mindestens einer gedrückten Maustaste. Es wird hierbei die Notation „M+1“ für „Maus bewegen mit Knopf 1 gedrückt“ und „M+1+2“ für „Maus bewegen mit Knöpfen 1 und 2 gedrückt“ verwendet. Danach wird in Kurzform angegeben, was eine Mausbewegung nach rechts/links bzw. oben/unten bewirkt.

4.2 globale Interaktion

Das Wechseln zwischen mehreren Interaktionsmodi erfolgt global mit dem Tasten „w“ für den „Walk-Mode“, „o“ für den „Orbit Mode“, „m“ für den „Move Mode“ und „r“ für den „Rotation Mode“. Der aktuell ausgewählte Modus beeinflußt die Ereignisse, die durch Mausbewegungen ausgelöst werden.

Die Maus bewirkt in den einzelnen Modis folgendes:

- Walk Mode

  Bewegt die Kamera wie folgt (aktuell selektiertes Objekt wird nicht verändert, da die Kamera kein eigenständiges Objekt ist):

  M+1 links/rechts, vor/zurück (Ich-Perspektive: Gehen)

  M+2 links/rechts, hoch/runter (Ich-Perspektive: Fliegen)

  M+1+2 Kamera schwenken

- Orbit Mode

  M+1 um Look-At kreisen (Look-At ist nicht sichtbar)

  M+2 Look-At bewegen in X-Y-Ebene

  M+1+2 Look-At bewegen in X-Z-Ebene

- Move Mode

  Bewegt das zur Zeit selektierte Objekt. (Achtung: Wählt man das selektierte Objekt im Menü an, so wird es deselektiert.)

M+1 Objekt in X-Y-Ebene bewegen

M+2 Objekt in X-Z-Ebene bewegen

M+1+2 wie „M+2“

- Rotation Mode

  Rotiert das aktuell selektierte Objekt.

  M+1 Objekt um Y- und X-Achse drehen

  M+2 Objekt um Y- und Z-Achse drehen

  M+1+2 wie „M+2“

Für „Move Mode“ und „Rotation Mode“ kann weiterhin noch global eingestellt werden, ob die Bewegungen bzw. Rotationen im Welt- oder Objektkoordinatensystem erfolgen soll. Dazu dienen die Tasten „G“ und „g“.

Weiterhin sind folgende Tasten global verfügbar:

z Z-Buffer aus

Z Z-Buffer an

a Animation aus

A Animation ein

+ Animationsgeschwindigkeit erhöhen

- Animationsgeschwindigkeit verringern

c vordere Clipebene weiter nach vorn verlagern

C vordere Clipebene weiter nach hinten verlagern

q Programm verlassen

ESC Programm verlassen

4.3 lokale Interaktion

Lokale Interaktion bezieht sich auf das spezielle, gerade ausgewählte Objekt. Objekte können über das Menü selektiert werden (außer das Weltobjekt und Materialien). Folgende Tastenkombinationen stehen für alle interaktiven Objekte zur Verfügung:

h Objekt nicht anzeigen (hide)

s Objekt anzeigen (show)

Für Lichtquellen gibt es zusätzlich:

l Lichtquelle deaktivieren

L Lichtquelle aktivieren

Für Spotlichtquellen gibt es zusätzlich:

t Cutoff verkleinern um $1°$

T Cutoff vergrößern um $1°$

y Lichtverteilungsexponenten um 1 vermindern

Y Lichtverteilungsexponenten um 1 erhöhen

5 Programmlisting

An dieser Stelle soll auf das Listing von ca. 13500 Zeilen C++-Code verzichtet werden. Zu Referenzzwecken steht die mit `ccdoc` generierte Klassendokumentation im Verzeichnis `ccdocs/` zur Verfügung.