

Wintersemester 2011/12

## Übungen zu Virtuelle Realität und Simulation - Blatt 2

Abgabe am 21. 10. 2011

### Aufgabe 1 (Magic Eye, 3 Punkte)

Erklären Sie, wie die sog. "Magic Eye"-Bilder konstruiert werden können, ausgehend von einem gegebenen Tiefenbild.

### Aufgabe 2 (User Interface, 2 Punkte)

Angenommen, Ihr User Interface (evtl. auch in VR) enthält einen Button, den ein User mit Hilfe eines Pointers typischerweise aus einer Entfernung von 100 Einheiten (z.B. Pixel) anklicken (= akquirieren) muss. Der Button sei 10 Einheiten groß.

Um die User-Effizienz zu steigern, haben Sie nun die Möglichkeit, entweder die Distanz um 10 Einheiten zu verringern, oder den Durchmesser des Buttons um 10 Einheiten zu vergrößern — welche wählen Sie? Begründen Sie kurz ihre Antwort.

### Aufgabe 3 (Image-Warping, 3 Punkte)

Beim einfachen Image-Warping für Stereo-Rendering ohne zweimaligem vollem Rendern des Bildes (Folie 52 im Real-Time-Rendering-Kapitel) wurde eine Formel für das Verschieben der Pixel angegeben. Leiten Sie diese her.

### Aufgabe 4 (Rollercoaster – Pendel, 10 Punkte)

- a) Abbildung 1 zeigt ein einfaches Pendel. Ihre Aufgabe besteht zuerst einmal darin, ein solches Pendel in VRML zu animieren. Eine Vorlage ist in der Datei `Rollercoaster.wrl` ab Zeile 3308 zu sehen. Dort ist ein einfaches Pendel integriert. Nutzen Sie eine Sprache ihrer Wahl (Java / JavaScript) um die Funktion für die Pendelbewegung zu berechnen:

$$\varphi = \varphi_{max} \cos(\sqrt{\frac{g}{l}} \cdot t)$$

$\varphi$  gibt die Auslenkung des Pendels wieder,  $\varphi_{max}$  somit den maximalen Pendelausschlag (hier  $40^\circ$ ).  $g$  ist die Erdbeschleunigung (9.78033) und  $l$  die Länge des Pendels mit 0.24. Der Wert für  $t$  kann durch die Abfrage des `TimeSensor TimerPendulum` aus `Rollercoaster.wrl` bestimmt werden. Ihre Aufgabe ist die korrekte Implementierung des Scripts `rotationFunction` in `Rollercoaster.wrl` Zeile 3362. Desweiteren müssen noch die entsprechenden ROUTEs erstellt werden.

- b) Der zweite Teil der Aufgabe besteht darin, das Pendel im richtigen Moment zu aktivieren. Hierfür ist zu beachten, dass momentan der `Timer TimerPendulum` immer läuft. Dies muss nun über ein Script oder eine ROUTE geändert werden! In der Szene wurde ein `ProximitySensor`

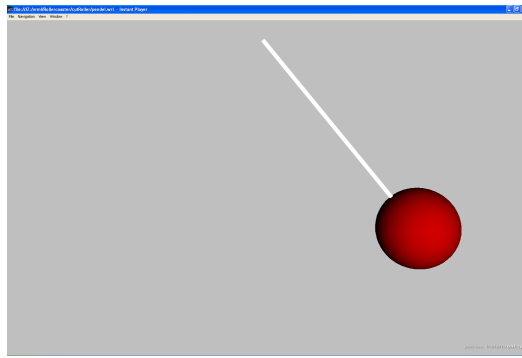


Abbildung 1: Pendel

PSPendulum platziert (in der Datei Rollercoaster.wrl Zeile 3310). Betritt der Betrachter den vorgegebenen Bereich, liefert dieser TRUE als `outEvent` für das Feld `isActive` und FALSE wenn der Betrachter den Bereich wieder verlässt beziehungsweise sich nicht in diesem befindet. Ziehen Sie die entsprechenden ROUTES, damit ihr Pendel nur dann aktiv ist, wenn sich der Betrachter in dem Bereich des *ProximitySensors* befindet.