

Aufgabenblatt 2

Aufgabe 1 - Multiplikation mit homogenen Koordinaten

Der zentrale Inhalt dieser Aufgabenserie wird sein, Transformationen für die Modelle zu ermöglichen. Um dies zu implementieren sind die folgenden Vorbereitungen nötig:

1. Informieren Sie sich zur Anwendung homogener Koordinaten in der Computergrafik.
2. Implementieren Sie in der *math.hpp* die Multiplikation einer *GLMatrix* ...
 - mit einem *GLVector*.
 - mit einem *GLPoint*.
 - mit einer *GLMatrix*.

Aufgabe 2 - Transformationen für die Modelle

Fügen Sie in *WireframeRenderer::renderScene* eine Matrixmultiplikation zur Transformation der Vertices hinzu. Die entsprechende Matrix sollte die Rotation, die Translation sowie die Skalierung beinhalten und für jedes Modell der Szene (*mModels*) unabhängig veränderbar sein. Sie soll dementsprechend Attribut der Klasse *Model* sein.

Default-Werte bei der Instanziierung eines Modells sollen sein: Keine Rotation, keine Translation, keine Skalierung. Schreiben Sie zusätzlich drei Methoden, mit denen sich die Rotation, die Translation und die Skalierung eines Modells setzen lassen und eine Methode, die die Matrix eines Modells bei Veränderungen neu berechnet:

- `Modell::setRotation(GLVector rotation)`
- `Modell::setTranslation(GLVector translation)`
- `Modell::setScale(GLVector scale)`
- `Modell::updateMatrix()`

Anmerkung: Entscheiden Sie selbst, ob mehrfache Aufrufe von Transformations-Methoden gleicher Art (mehrfache Translationen, mehrfache Rotationen oder mehrfache Skalierungen) zur Verkettung mit vorher gesetzten Transformationen gleicher Art führen oder ob lediglich die zuletzt gesetzten drei Transformationen (Translation, Rotation und Skalierung) angewendet werden. Machen Sie sich zudem Gedanken dazu, in welcher Reihenfolge Sie die Transformationen auf das Modell anwenden.

Verwenden Sie die von Ihnen erstellten Setter in der Main-Methode in *Main.cpp* um das Bunny-Modell (*bunny_scaled.ply*) und den Cube (*cube_scaled.ply*) in einem Bild ungefähr gleich groß nebeneinander zu rendern. Fügen Sie zudem für mindestens eines der Modelle eine Rotation hinzu. Ein Beispiel für ein mögliches Ergebnis dieser Aufgabe finden Sie in Abbildung 1.

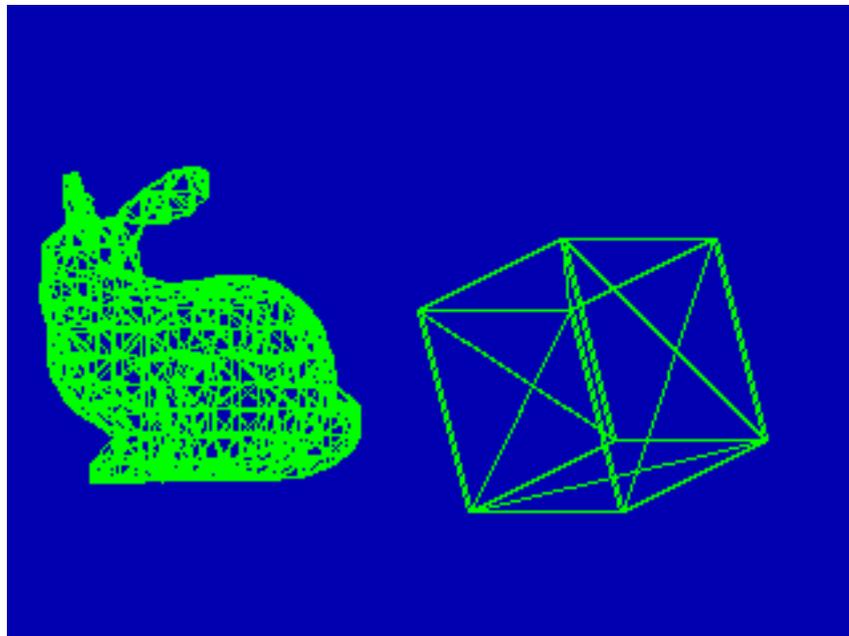


Abbildung 1: Zielsetzung der angewendeten Transformationen (Beispiel)

Anmerkung: Für dieses Rendering wurde der Cube um jeweils 100 Einheiten in x und y Richtung verschoben, auf 90 Prozent skaliert und um 20 Grad um die x-Achse sowie 45 Grad um die y-Achse rotiert. Das Bunny-Modell wurde um 250 Einheiten auf der x-Achse und um 100 Einheiten auf der y-Achse verschoben, auf 80 Prozent skaliert und um 5 Grad um die y-Achse rotiert.