

Wintersemester 2018/19

Übungen zu Computergraphik - Blatt 9

Abgabe am 13.1.2019

Aufgabe 1 (Transformation — in Worten, 1 Punkte)

Beschreibe ein Verfahren (in Worten), mit dem man eine Transformationsmatrix berechnen kann, die ein Objekt um eine bestimmte, beliebige Achse im Raum rotieren kann.

Diese Rotation ist gegeben durch einen Punkt A , einen Vektor \mathbf{r} , und einen Winkel ϕ .

Aufgabe 2 (Transformation — in Zahlen, 1+1 Punkte)

a) Zeigen Sie, dass M_1 und M_2 die selbe Skalierung beschreiben.

$$M_1 = \begin{pmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & s & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (1)$$

$$M_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{1}{s} \end{pmatrix} \quad (2)$$

b) Sei folgende Transformation gegeben:

$$M = S(s_x, s_y, s_z) \cdot T(t_x, t_y, t_z) \cdot R(\phi, r_x, r_y, r_z) \quad (3)$$

Geben Sie M^{-1} an.

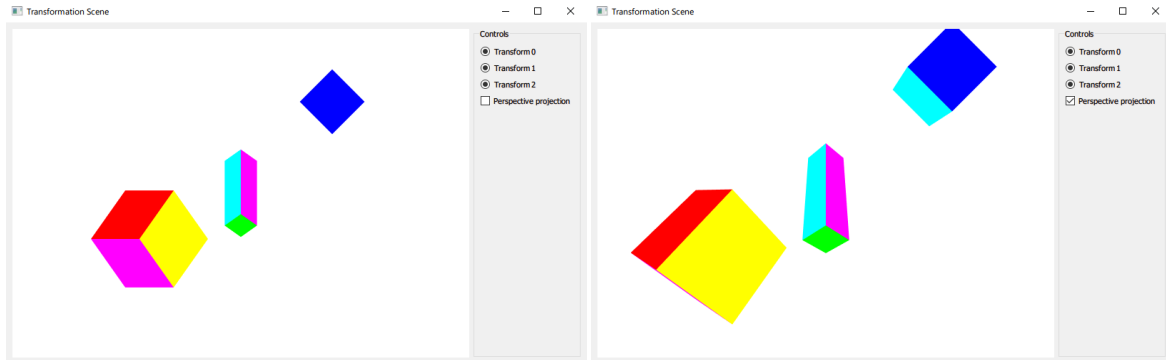
Aufgabe 3 (Transformation — in C++, 3+1,5+3,5 Punkte)

a) Erweitern Sie das Framework `Transformation` um Funktionen die elementare Rotation, Skalierung und Translation als Matrix abbilden. Implementieren Sie dazu folgende statische Funktionen in `Matrix4x4`:

- `Identity`: Einheitsmatrix.
- `RotationX`: Rotation um die X-Achse um einen Winkel in Grad gegeben.
- `RotationY`: Rotation um die Y-Achse um einen Winkel in Grad gegeben.
- `RotationZ`: Rotation um die Z-Achse um einen Winkel in Grad gegeben.
- `Scale`: Skalierung entlang der X-, Y- und Z-Achsen um die gegebenen Faktoren.
- `Translation`: Verschiebung um den gegebenen Vektor durch homogene Koordinaten.

Es sollen keine externen Bibliotheken, insbesondere Qt, zur Hilfe dieser Funktionen benutzt werden.

- b) Erzeugen Sie drei verschiedene Würfel durch Kombination von elementaren Transformationen, zu sehen in Abbildung 1. Nutzen Sie die von Ihnen implementierten Funktionen in `Matrix4x4.cpp`. Dabei gibt es eine Ausnahme: Die Matrix darf (und sollte) mit `m_trackball` initialisiert werden. Sollten Sie die Aufgabe 3 a) nicht gelöst haben, dann können Sie diese Unteraufgabe auch mit Hilfe von `QMatrix4x4` lösen.



(a) Oerthografische Ansicht der drei Objekte.

(b) Perspektivische Ansicht der drei Objekte.

Abbildung 1: Die drei Objekte, die durch `setupTransform0-3` vorbereitet werden.

Implementieren Sie dazu die Funktionen `setupTransform0`, `setupTransform1` und `setupTransform2` in `glWidget.cpp`, wobei jede dieser Funktionen eines der Objekte jeweils vorbereitet.

- c) Implementieren Sie eine Steuerung des Objektes mittels virtuellem Trackball wie in der Vorlesung vorgestellt. Vervollständigen Sie dazu die Funktion `mouseMovement` und `mouseMoveEvent` in der Datei `glWidget.cpp`.

Stellen Sie dafür sicher, dass die Transformationen in `setupTransform0-2` mit `m_trackball` initialisiert werden, einen entsprechenden Konstruktor gibt es in `Matrix4x4` bereits.

Sie dürfen gerne Teile der Qt-Bibliothek benutzen, insbesondere die Klassen `QVector3D` und `Quaternion` inklusive aller Funktionen.