

Grafika Komputerowa Wirtualna Kamera

Mikołaj Kubik

27 maja 2020

1 Wstęp

Celem projektu była implementacja wirtualnej kamery, pozwalającej na obserwację trójwymiarowych obiektów. Kamera posiada możliwość poruszania się i obracania w trzech osiach oraz zmiany przybliżenia. Kod źródłowy programu oraz plik jar znajdują się w repozytorium [GitHub](#).

2 Zmiany względem wstępnego planu

Względem pierwotnego planu zmieniła się technologia - wykorzystałem język java i bibliotekę swing zamiast c++ z sfml, ponieważ okazało się, że różnica wydajnościowa między tymi narzędziami nie wpływa w widoczny sposób na działanie programu.

3 Analiza zagadnienia

Program przechowuje scenę jako listę odcinków. Aby dokonać projekcji sceny na perspektywę kamery z każdego punktu tworzona jest macierz, na której wykonywane są następujące operacje.

3.1 Translacja

Transformacja sceny odpowiadająca za pozorną zmianę położenia kamery.

$$T = \begin{bmatrix} 1 & 0 & 0 & X \\ 0 & 1 & 0 & Y \\ 0 & 0 & 1 & Z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x + X \\ y + Y \\ z + Z \\ 1 \end{bmatrix} \quad (1)$$

gdzie:

X, Y, Z - wektor translacji

x, y, z - współrzędne transformowanego punktu

3.2 Skalowanie

Transformacja sceny odpowiadająca za przybliżanie i oddalanie. Zastosowanie tego równania macierzowego pozwala na niezależne skalowanie projekcji w płaszczyznach każdej z trzech osi, jednak aby zachować zachowanie zbliżone do prawdziwej kamery ograniczyłem się do skalowania względem osi x i y jednocześnie.

$$S = \begin{bmatrix} SX & 0 & 0 & 0 \\ 0 & SY & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} SX \cdot x \\ SY \cdot y \\ z \\ 1 \end{bmatrix} \quad (2)$$

gdzie:

SX, SY - współczynniki powiększenia względem osi X i Y, SZ w tym przypadku wynosi 1

x, y, z - współrzędne transformowanego punktu

3.3 Rotacje

Transformacje sceny odpowiadające za pozorne obracanie kamery.

3.3.1 względem osi X

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ \cos\alpha \cdot y - \sin\alpha \cdot z \\ \sin\alpha \cdot y + \cos\alpha \cdot z \\ 1 \end{bmatrix} \quad (3)$$

gdzie:

α - rotacja kamery względem osi X

x, y, z - współrzędne transformowanego punktu

3.3.2 względem osi Y

$$\begin{bmatrix} \cos\beta & 0 & \sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\beta \cdot x + \sin\beta \cdot z \\ y \\ -\sin\beta \cdot x + \cos\beta \cdot z \\ 1 \end{bmatrix} \quad (4)$$

gdzie:

β - rotacja kamery względem osi Y

x, y, z - współrzędne transformowanego punktu

3.3.3 względem osi Z

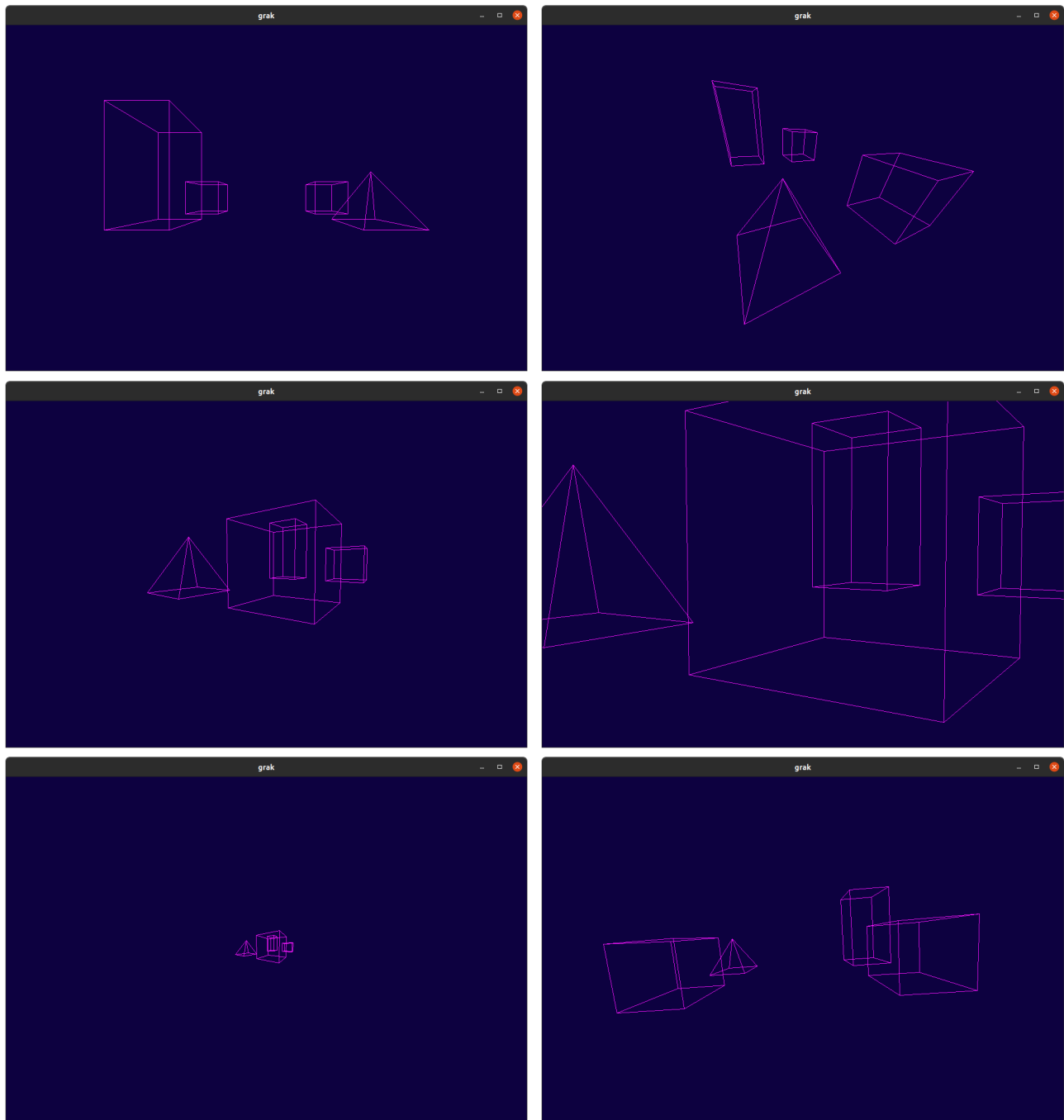
$$\begin{bmatrix} \cos\gamma & -\sin\gamma & 0 & 0 \\ \sin\gamma & \cos\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\gamma \cdot x - \sin\gamma \cdot y \\ \sin\gamma \cdot x + \cos\gamma \cdot y \\ z \\ 1 \end{bmatrix} \quad (5)$$

gdzie:

γ - rotacja kamery względem osi Z

x, y, z - współrzędne transformowanego punktu

4 Przykładowe wywołania



Nagranie działania programu [link](#)

5 Obsługa programu

5.1 Wykorzystane narzędzia

Program korzysta z javy 8 oraz The Apache Commons Mathematics Library do obliczeń macierzowych.

5.2 Plik wejściowy

5.3 Uruchomienie

```
java -jar swing_virtual_camera.jar plik_konfiguracyjny
```

Plik `swing_virtual_camera.jar` znajduje się w `out/artifacts/swing_virtual_camera_jar` względem głównego katalogu programu.

5.4 Sterowanie

5.4.1 Poruszanie kamerą

- Lewo: a
- Prawo: d
- Przód: w
- Tył: s
- Góra: space
- Dół: ctrl

5.4.2 Obracanie kamery

- Lewo: a
- Prawo: d
- Przód: w
- Tył: s
- Pochylenie w lewo: q
- Pochylenie w prawo: e

5.4.3 Zoom

- Powiększenie: 1
- Pomniejszenie: 2