# Kamera wirtualna

Michał Janaszewski

Cel

Celem projektu jest stworzenie systemu wirtualnej kamery w 3D, kamera ma mieć możliowść poruszanai się w 6 zwrotach i możliwość wykonywania rotacji wokół trzech osi w obu zwrotach, dodatko 2 operacje zoom.

Program stworzony w języku Python, z wykorzystaniem wirtualnego środowiska (virtual environment).

Opis wykonania

* 1. Układ współrzędnych

Wybrano układ współrzędnych w którym kamera jest ustawiona w centum układu (0,0,0), a operacje wykonywane są obiektach sceny - prostopadłościanach.

* 1. Rotacja kamery

Rotacja kamery jest realizowana za pomocą **kwaternionów**, które pozwalają na obrót wokół dowolnej osi bez problemów związanych z klasycznymi kątami Eulera (np. gimbal lock). Kwaterniony reprezentują obrót, który jest wykonywany wokół określonej osi (X, Y lub Z) w przestrzeni 3D.

* 1. Translacja kamery

Translacja kamery została zrealizowana przez odwrócenie rotacji kamery, a następnie przesunięcie jej w przestrzeni. Oznacza to, że zanim przekształcimy wektor przesunięcia, należy uwzględnić obecną rotację kamery. Tylko wtedy, po odwróceniu rotacji, translacja (przesunięcie) odbywa się w kierunkach zgodnych z układem współrzędnych kamery, a nie globalnym.

* 1. **Zoomowanie**

Zoomowanie kamery polegało na zmianie **kąta widzenia (FOV)**, co wpływało na sposób rzutowania obiektów 3D na ekran, dając efekt przybliżania lub oddalania widoku.

* 1. **Rzutowanie 3D na 2D**

W projekcie użyto **projekcji perspektywicznej**, który przekształca współrzędne 3D na 2D, a problem przycinania śćan prostopadłościanów rozwiązany algotytem Cohena-Sutherlanda.

Efekt działania programu