

Computer Graphics Project2 Report

2019060682 정진우

1. Which requirements implemented

A. Manipulate the camera in the same way as in Project1 using your Project1 code

Project1의 코드를 그대로 사용하였고, 스케일링을 위해 grid의 간격만 조정해주었다.

B. Single mesh rendering mode

obj 파일을 드래그&드랍 시 해당 obj 파일의 vertex position, vertex normal, face 정보들을 읽어 glm.array로 파싱하여 사용하였다. 4개 이상의 vertex로 구성 된 face의 경우 triangle로 표현할 수 있도록 3개씩 나누어 vertex로 설정해주었다.

C. Animating hierarchical model rendering mode

H키를 누를 시 기존에 다운로드받은 obj 파일 3개를 불러오도록 하였고, cat obj 파일을 가장 parent node로, gun obj 파일 2개를 cat의 child node로, heart obj 파일 4개를 각 2개씩 gun의 child node로 설정해주었다.

cat에는 x축에 대해 앞뒤로 움직이도록 Translate Matrix를 설정해주었다.

gun에는 y축에 대해 90도 회전하면서 움직이도록 Rotate Matrix를 설정해주었다.

heart에는 서로 교차하면서 움직이도록 Translate Matrix를 설정해주었다.

부모의 transform matrix를 상속 받아 같이 움직일 수 있도록 구현하였다.

D. Lighting & Etc

face에 나온 정보를 토대로 vertex normal을 각 vertex에 설정하여 lighting을 구현하였다. 이 과정에서, 하나의 vertex가 여러 개의 vertex normal를 가지는 경우에 대해 Opengl에서는 지원하지 않기 때문에 light가 번지는 현상이 발생하였다. 이 경우 하나의 vertex에 대해 vertex normal들의 average를 구하여 표현할 수 있으나, 파일이 커질 경우 vertex와 vertex normal을 매핑하는 데에 시간이 너무 오래걸려 구현하지 않았다.

multiple light source를 구현하기 위해 시간별로 light position이 변하도록 설정하였다.

'z'키를 입력 할 시 glPolygonMode 함수를 이용하여 wireframe / solid 모드를 구현하였다.

유튜브 링크

<https://youtu.be/CRG5iMCdy2w>

영상 속 cat이 lighting이 없는 이유는 vertex normal이 없는 obj파일이기 때문이다.
gun의 빛번짐이 heart에 비해 심한 이유는 vertex의 개수가 적기 때문이다.