

[CSE4170] 기초 컴퓨터 그래픽스 HW3: OpenGL 3D Viewing 과정의 이해

담당교수: 임 인 성

May 17, 2021

제출 마감: 5월 31일 (월) 오후 8시 정각 (제출 방법은 조교가 게시판에 공고)

목표: 이번 프로그래밍 숙제는 Real-time rendering pipeline의 전반부의 근간을 이루는 3D Viewing 과정에 대한 이해를 높임을 목표로 한다. 이를 위하여, OpenGL API 함수를 사용하여 Modeling Transformation, Viewing Transformation, 그리고 Projection Transformation에 대하여 기본적인 3D Viewing 과정을 구현하여 본다.

문제: 본 수업에서 제공하는 예제 프로그램에 기술된 3D 물체를 활용하여 (자신만의 3D 기하 물체를 사용해도 무방함), 다음의 기능을 가지는 OpenGL 프로그램을 작성하라. 이때, 아래의 각 기능을 구동하는 사용자 인터페이스 방법 (키보드와 마우스 등을 사용)을 사용자가 자연스럽게 받아들일 수 있도록 설계한 후, 아래의 각 기능 별 구동법을 반드시 자신의 프로그램과 같이 제출하는 README 파일에 명확히 기술하라 (파일명: README.20021234.{txt,hwp,docx}). 조교는 이 파일의 내용을 참조하여 각 기능이 구현이 되었는지 하나씩 확인할 예정이다.

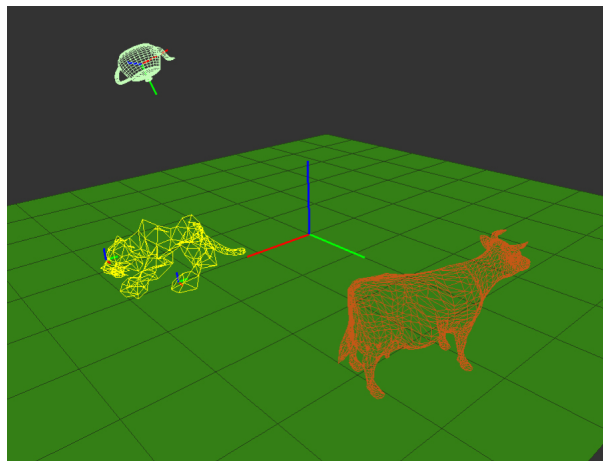


Figure 1: 가상의 3차원 세상

I. Modeling Transformation

다음의 조건을 만족하는 가상의 세상을 구축하라.

1. 먼저 가상의 3차원 세상의 바닥과 좌표의 기준이 되는 세상 좌표계를 그려라 (**최대 10점**)
2. 최대 5개까지의 서로 다른 정적인 물체를 서로 다른 모델링 변환을 사용하여 가상의 세상에 배치하라 (**물체 당 5점 최대 25점**).
3. 최대 4개까지의 서로 다른 동적인 물체를 가상의 세상에 배치하라 (**물체 당 10점 최대 40점**). 각 동적 물체는 이동 변환, 크기 변환, 그리고 회전 변환 등의 기본 기하 변환 중 최소한 두 개

이상을 사용하여 서로 다른 움직임을 표현해야 하며, 각 동적 물체는 키보드 또는 마우스 동작을 통하여 움직임과 멈춤을 조절할 수가 있어야 한다 (**자신이 선택한 최대 네 개의 동적인 물체에 대해 이 기능이 구현이 안되어 있으면 물체 당 4점 감점**).

요구 사항은 아니나 동적인 물체들 중 최소한 1개의 물체에 대해서는 뉴턴의 운동의 법칙과 같이 물리적으로 충실한 방법을 사용하여 움직임을 표현해볼 것.

II. Viewing Transformation

다음과 같은 성질을 가지는 서로 다른 5개의 카메라를 구현하라.

- 1번부터 4번까지의 카메라는 CCTV 카메라와 같이 주어진 위치에 고정하여 세상을 바라보는 카메라이다. 적절한 사용자 인터페이스 동작을 통하여 원하는 카메라에서 세상을 바라볼 수 있도록 하라.
- 5번 카메라는 동적인 카메라로서 사용자 인터페이스 동작을 통하여 다음과 같이 움직일 수 있도록 하라.
 - (a) 사용자가 원하는대로 카메라의 \vec{u} , \vec{v} , 그리고 $-\vec{n}$ 벡터에 해당하는 각 축 방향으로 이동 (translation) 할 수 있도록 하라.
 - (b) 사용자가 원하는대로 카메라의 \vec{u} , \vec{v} , 그리고 $-\vec{n}$ 벡터에 해당하는 각 축 둘레로 회전 (rotation) 할 수 있도록 하라. 즉 카메라에 대하여 pitch, yaw, 그리고 roll 기능을 구현하라.
- (추가) 1번 카메라에 대하여 고정된 위치를 중심으로 시선의 방향을 바꿀 수 있도록 하라. 어떠한 방식으로 구현할 지는 본인이 결정할 것.

배점은 다음과 같다.

- 1번부터 4번까지의 카메라를 제대로 구현하였으면, **각 카메라 당 7점 (최대 28점)**
- 5번 카메라의 이동 기능 제대로 구현하였으면, **각 축 방향마다 6점 (최대 18점)**
- 5번 카메라의 회전 기능을 제대로 구현하였으면, **각 축 둘레마다 8점 (최대 24점)**
- 1번 카메라의 시선 방향 조절 기능을 제대로 구현하였으면, **구현 내용에 따라 최대 20점**

III. Projection Transformation

적절한 사용자 인터페이스 동작을 통하여 5번 카메라에 대하여 줌 인/줌 아웃 기능을 구현하라. 이때, 최대로 줌 인/줌 아웃할 수 있도록 적절히 범위를 설정하라.

배점은 다음과 같다.

- 줌 인/줌 아웃 기능을 제대로 구현하였으면, **최대 14점**

[참고]

- 본인이 구현한 기능들을 구현하지 못한 것들과 구분하여 본 숙제의 항목 순서대로 README 파일에 정확히 기술할 것.
- 상기 항목별 배점은 모두 최대 가능 점수이며, 구현이 불충분할 경우 부분 점수를 부여할 수 있음.
- 다음 프로그래밍 숙제는 본 숙제의 틀을 사용할 예정이므로, 가급적 완성도를 높혀 구현할 것.
- 제출 화일에서 바이러스 발견 시 **최고 점수 X (-1)**/다른 사람의 숙제를 복사할 경우 복사한 사람과 복사 당한 사람 모두 **최고 점수 X (-10)**임.