

심화전공실습 (CGL)

HW07_Shading



Self-scoring table

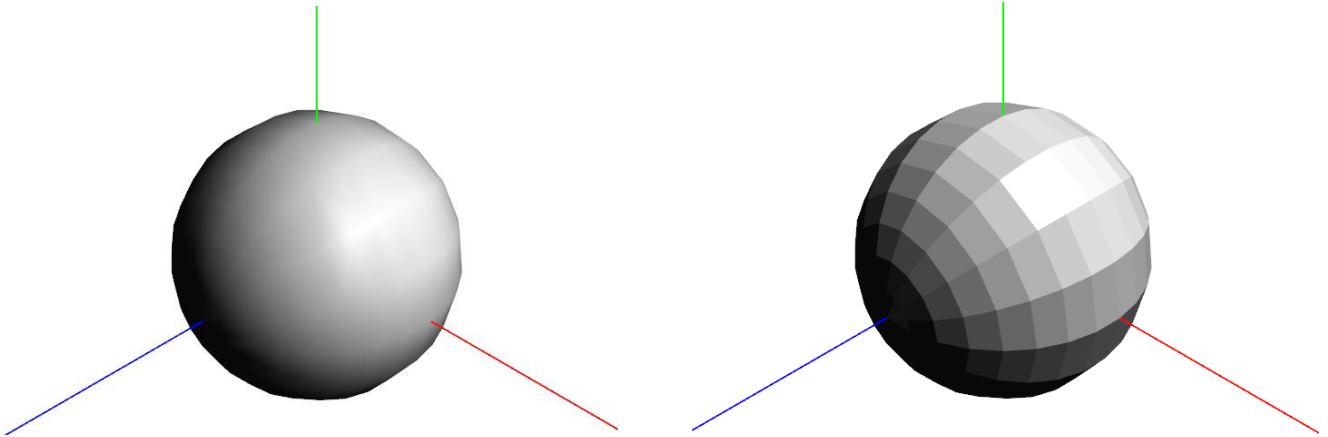
	P1	P2	E1	E2	Total
Score	1	1	1	1	4

2018707068 김경환

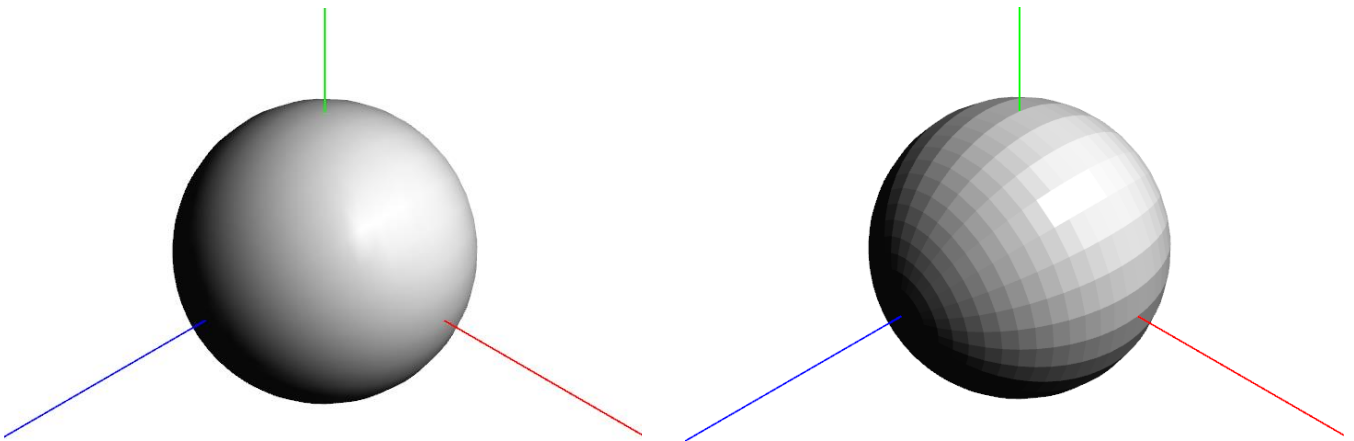
KwangWoon University

Practice01 Snapshot, Explanation:

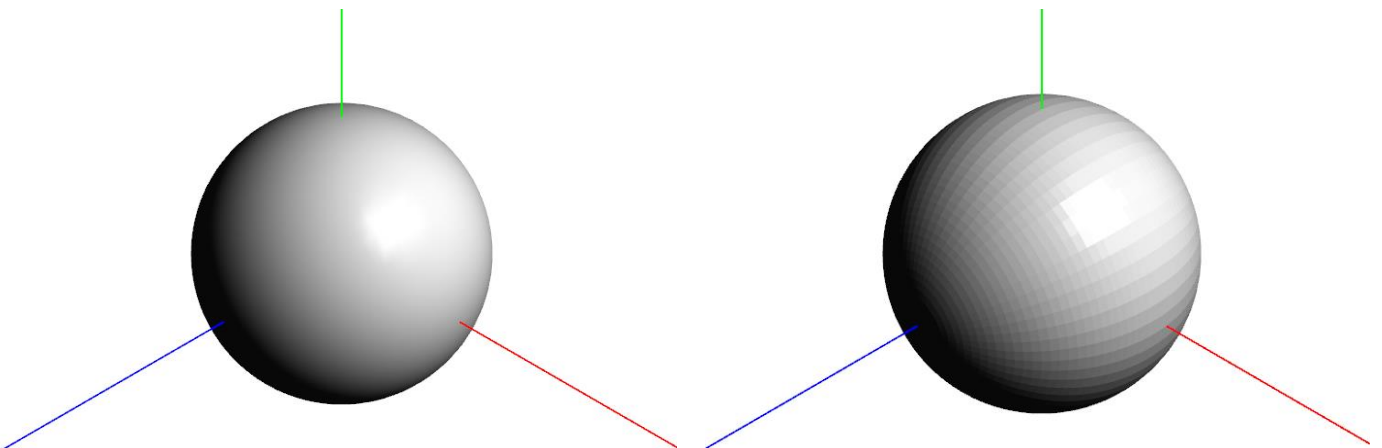
왼쪽이 flat shading으로, 오른쪽은 smooth shading으로 sphere를 그린 모습이다
glShapeModel의 인자로 GL_FLAT, GL_SMOOTH를 주어 구현할 수 있다.



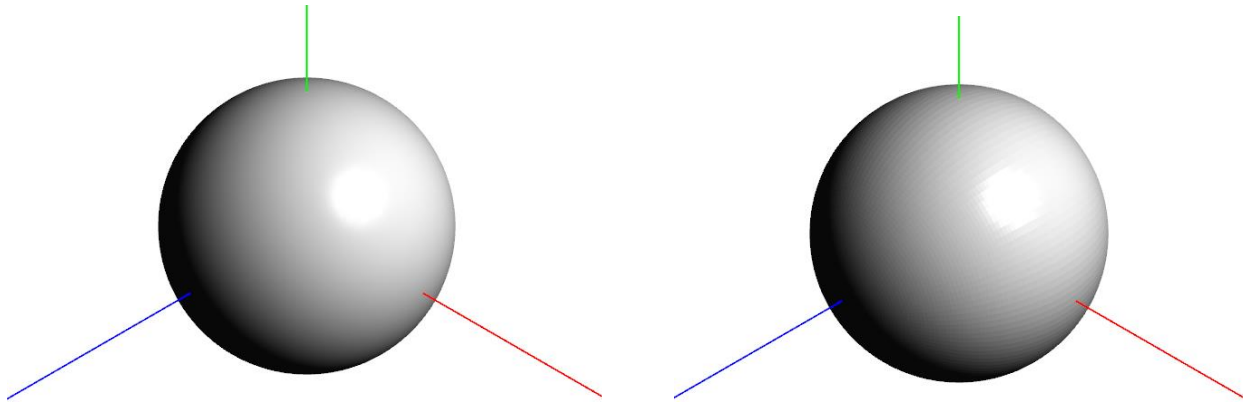
Keyboard Input으로 1을 입력했을 때 나오는 16 slices와 16 stacks를 가진 sphere이다.
해당 사진을 보면 sphere의 외곽 태두리가 많이 굴곡진 것을 볼 수 있다.



Keyboard Input으로 2을 입력했을 때 나오는 32 slices와 32 stacks를 가진 sphere이다.
해당 사진은 이전 16, 16 보다는 낫지만 아직도 외곽 태두리가 굴곡진 것을 볼 수 있다.

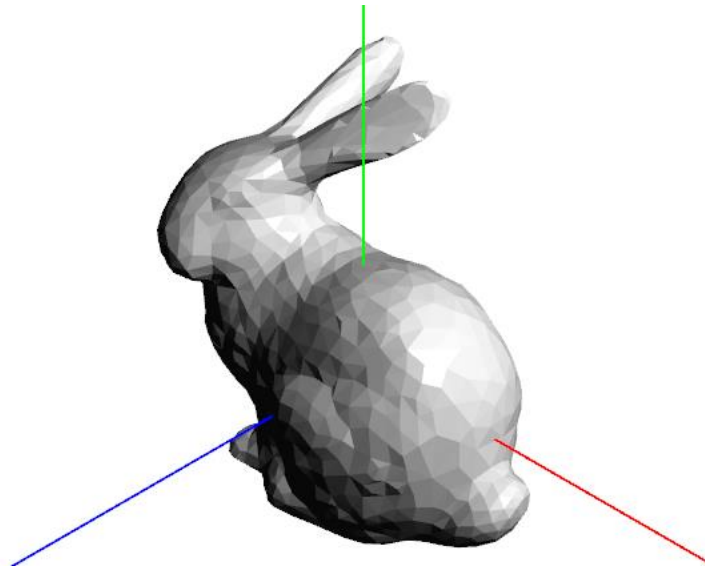


Keyboard Input으로 3을 입력했을 때 나오는 64 slices와 64 stacks를 가진 sphere이다.
해당 사진에서는 외곽 태두리가 굴곡진 것은 많이 좋아졌으나 아직까지도 하이라이트 부분에서 좋지 못한 퀄리티를 보인다.



Keyboard Input으로 4를 입력했을 때 나오는 128 slices와 128 stacks를 가진 sphere이다.
해당 사진을 보면 외곽 태두리와 하이라이트 부분이 많이 좋아진 것을 볼 수 있다.

Practice02 Snapshot, Explanation:

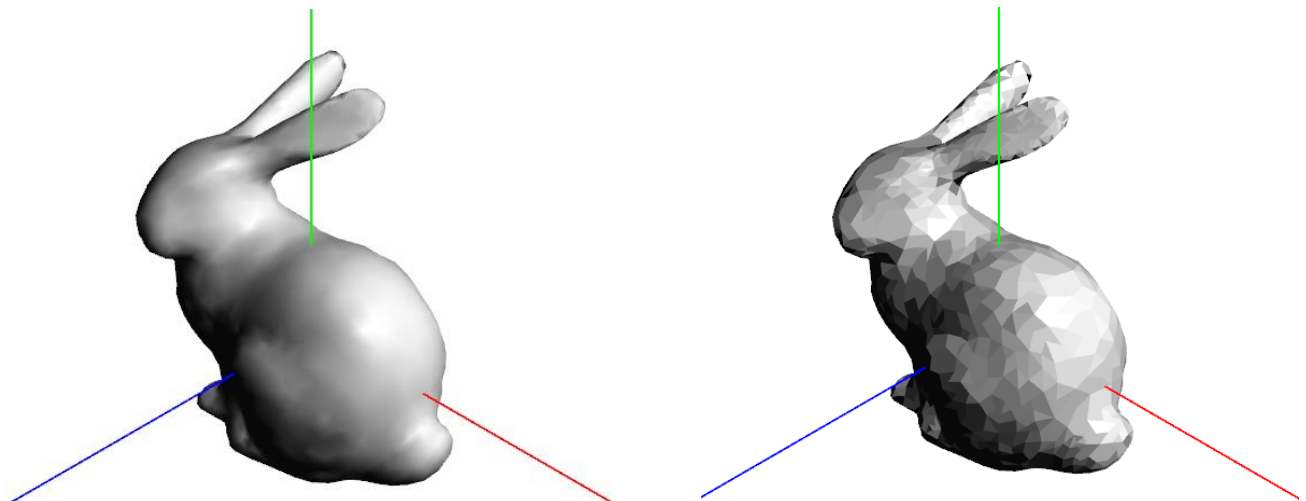


Keyboard Input으로 5을 입력했을 때 나오는 flat approximation of a bunny이다.

이는 GL_TRIANGLES와 face normal vector를 사용하여 flat mesh로 bunny model을 그린 모습이다.

여기서 face normal vector는 삼각형을 이루는 세 vertex를 이용하여 한 vertex에서 나머지 둘 vertex로 향하는 vector를 만들고, 이를 오른손 법칙에 따라 cross product하여 구하였다.

해당 flat mesh로 그린 bunny model은 face normal vector를 사용하기 때문에 세 vertex에 동일한 normal vector가 갖게 되고 따라서, flat shading과 smooth shading 둘 다 같은 모습을 보인다.



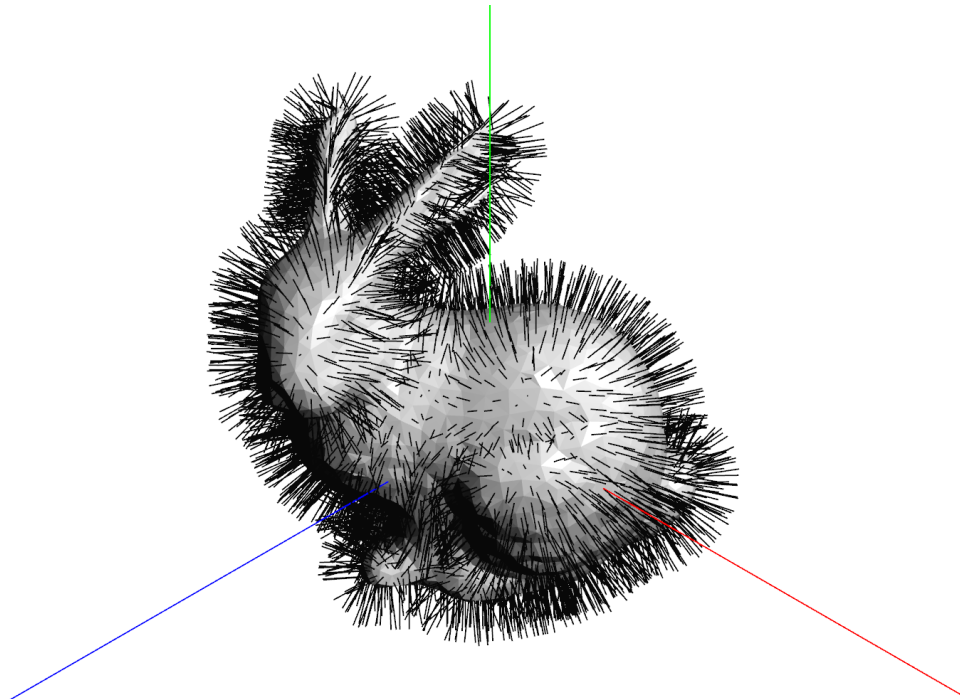
Keyboard Input으로 5을 입력했을 때 나오는 smooth approximation of a bunny이다.

이는 GL_TRIANGLES와 vertex normal vector를 사용하여 smooth mesh로 bunny model을 그린 모습이다
왼쪽은 smooth mesh에 smooth shading을 적용한 그림이고, 오른쪽은 smooth mesh에 flat shading을 적용한 그림이다. 위 flat mesh로 그린 bunny model과 달리 flat, smooth shading이 서로 다른 모습을 보이는 이유는 삼각형을 이루는 세 vertex가 서로 다른 normal vector를 갖기 때문이다.

GL_FLAT -> 3개의 vertex 중 하나로 내부를 칠함. (3개의 vertex가 서로 다른 normal vector)

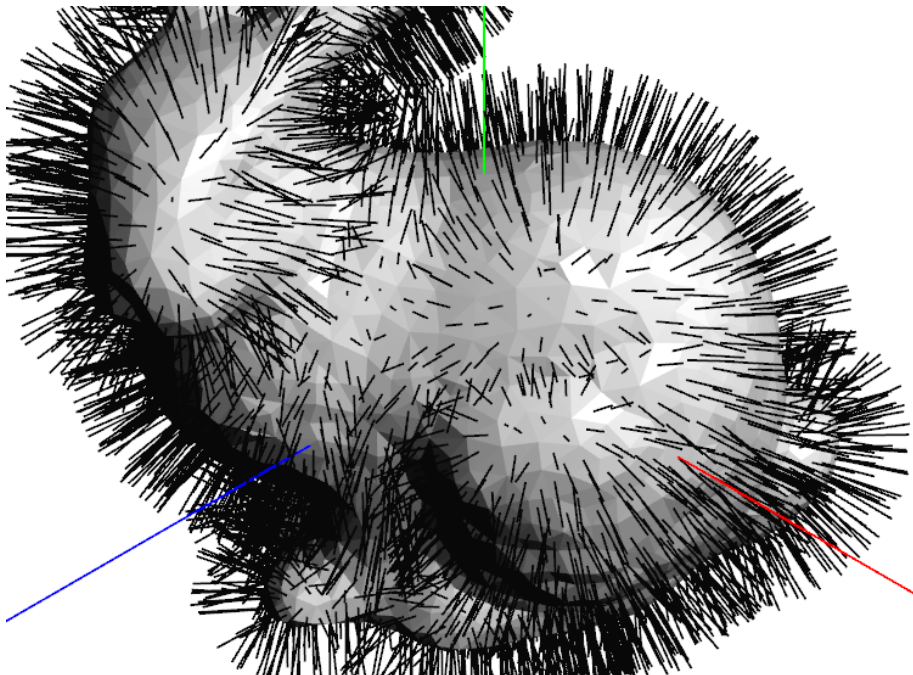
GL_SMOOTH -> 3개의 vertex를 부드럽게 연결하여 칠함. (3개의 vertex가 동일한 normal vector)

Exercise01 SnapShot, Explanation:



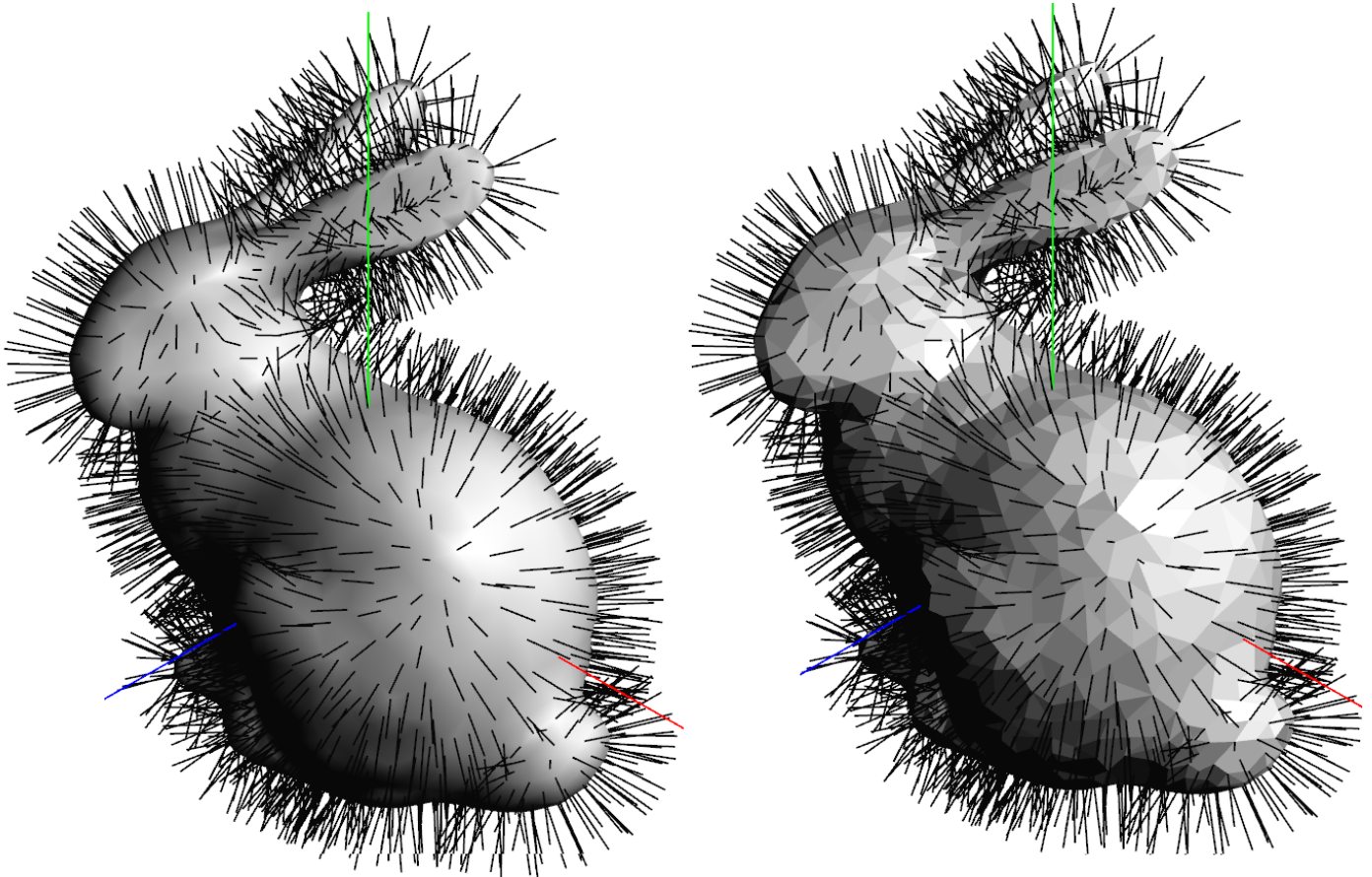
Bunny model의 삼각형에 대한 normal vector를 그린 모습이다.

해당 사진에서는 normal vector의 시작지점, 방향 등이 잘 보이지 않으므로 아래 사진에서 설명하겠다.



위 사진을 보면 삼각형에 수직인 normal vector가 삼각형의 무게중심에서 시작되는 것을 볼 수 있다. 이는 삼각형을 이루는 세 vertex의 x, y, z 좌표를 각각 합한 다음 이에 대한 평균값을 계산해 무게 중심을 구하고, 여기서 normal vector를 더해줬기 이러한 모습이 보인다. 또한, 여기서 사용된 face normal vector는 위에서 이야기한 것과 같이 삼각형을 이루는 세 vertex로 한 점에서 다른 점으로 나가는 vector를 구한 후 오른손 법칙에 따라 cross product를 해줘 구할 수 있었다..

Exercise02 SnapShot, Explanation:



Bunny model의 vertex에 대한 normal vector를 그린 모습이다.

해당 bunny model을 smooth shading 하였을 땐 vertex normal vector의 시작지점, 방향 등이 잘 보이지 않아 flat shading과 나란히 놓고 비교해가며 보겠다.

오른쪽 flat shading을 보면 삼각형의 세 vertex에서 normal vector가 나오고, 세 vertex의 normal vector가 서로 다를 수 있다. 그리고 해당 bunny model은 vertex를 총 2162개, face를 총 4320개를 가졌기 때문에 normal vector가 위 Exercise1보다 Exercise2가 더 sparse한 것을 볼 수 있다.

마지막으로 vertex normal vector를 구할 때는 face normal vector를 먼저 구하고, 이러한 face normal vector를 해당 face를 이루는 세 vertex의 0으로 초기화된 normal vector에 더해준다. 그리고 각각의 vertex normal vector를 normalize하여 구하였다. 그리고 vertex의 좌표와 vertex normal vector의 합을 통해 vertex에서 시작되는 vertex normal vector를 그릴 수 있었다.