

심화전공실습 (CGL)

HW12_Picking



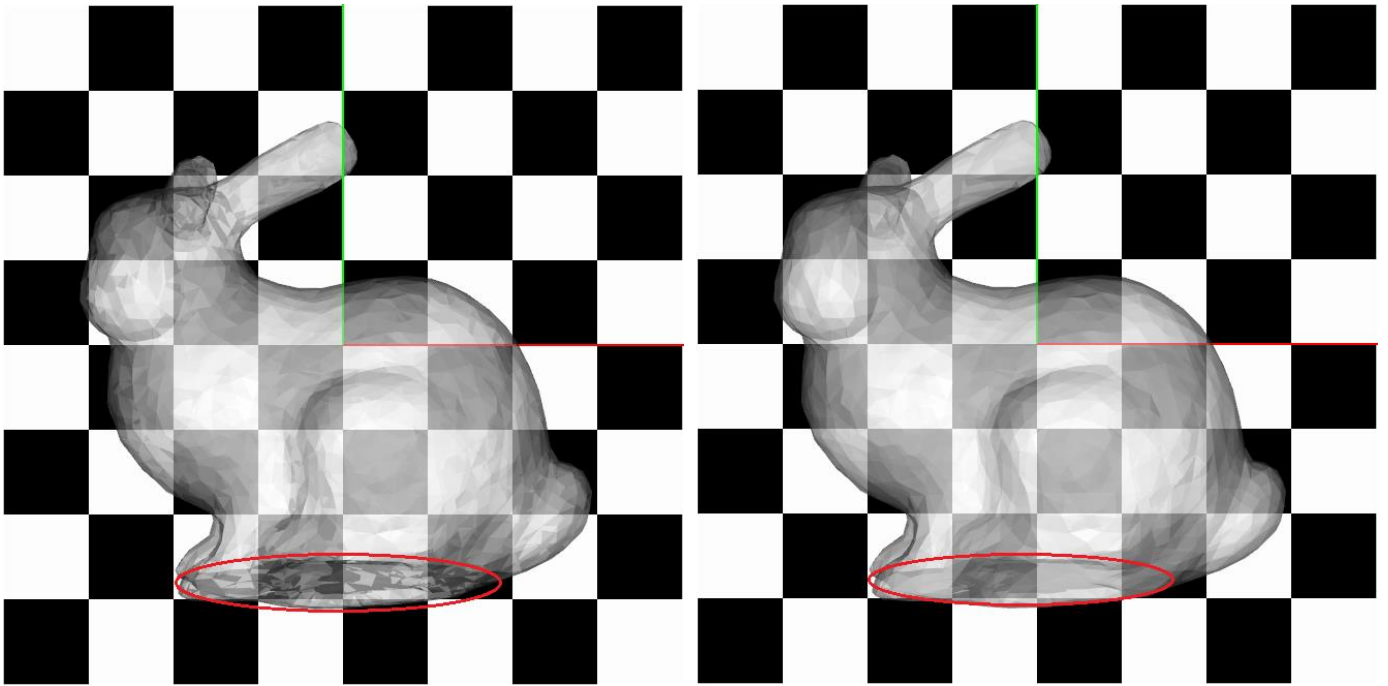
Self-scoring table

	P1	P2	P3	E1	Total
Score	1	1	1	1	4

2018707068 김경환

KwangWoon University

Practice 01. Turn on/off depth sorting to a translucent flat bunny:

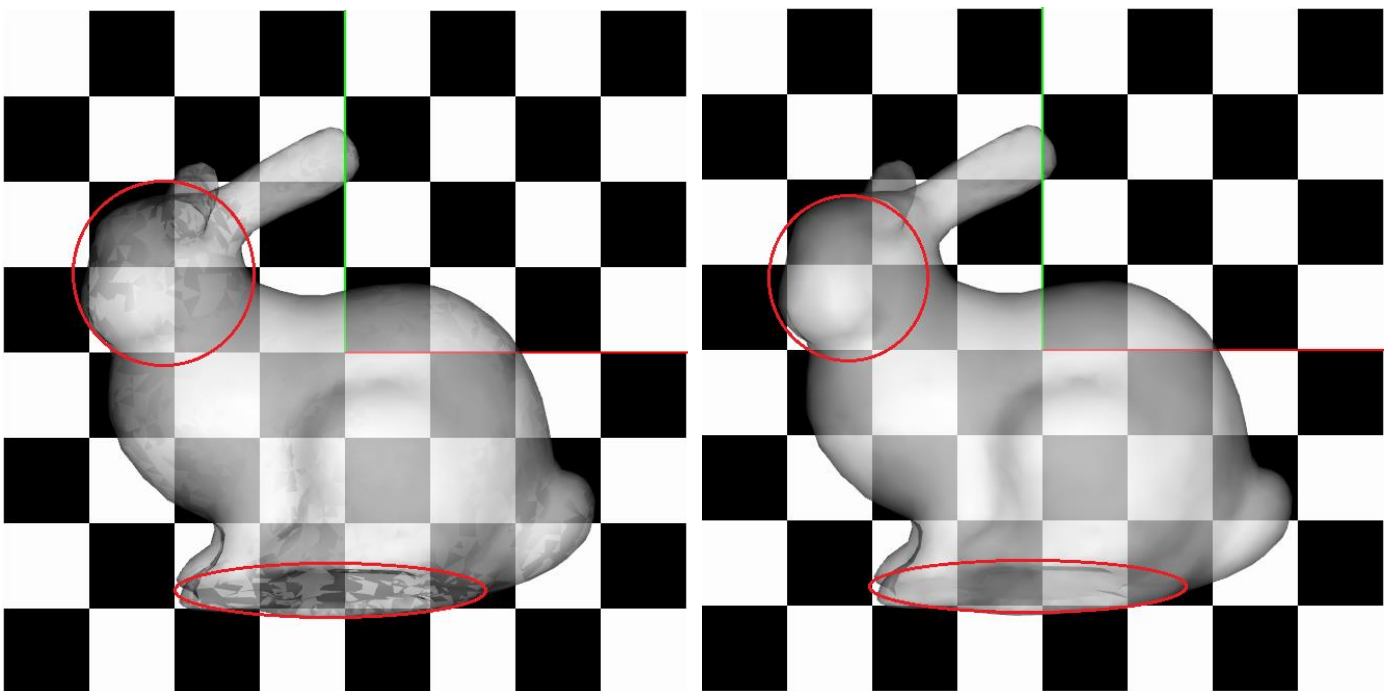


왼쪽은 Depth sorting 적용된, 오른쪽은 Depth sorting이 적용되지 않은 translucent flat bunny이다.

위 사진에서 빨간색 동그라미로 표시되어 있는 부분에서 큰 차이가 나는 것을 볼 수 있다.

이는 Depth sorting을 하지 않으면 translucent polygon(여기에서는 triangle)이 무작위의 순서로 그려지기 때문에 이미지가 깨진 듯한 모습을 보인다. 하지만 Depth sorting을 적용하면 translucent polygon을 먼 곳부터 가까운 곳까지 순서대로 그리기 때문에 그러한 현상은 일어나지 않는 것을 알 수 있다.

Practice 02. Turn on/off depth sorting to a translucent smooth bunny:



왼쪽은 Depth sorting 적용된, 오른쪽은 Depth sorting이 적용되지 않은 translucent smooth bunny이다.

위에서 설명한 것과 같이 빨간색 동그라미로 표시되어 있는 부분에서 큰 차이가 나는 것을 볼 수 있다.

Smooth bunny의 경우 flat bunny와 다르게 polygon의 모양이 보이지 않아야 하기 때문에 더 큰 차이가 나는 것처럼 보인다.

이러한 모습이 보이는 이유는 위에서 설명한 것과 같이 그려지는 순서로 인한 것이다.

Depth sorting을 구현하는 방법으로는 우선 Depth Sort Data를 갖고 있는 구조체를 선언한다.

그리고 해당 구조체를 그리고자 하는 물체의 face 개수만큼 동적할당을 통해 선언한다.

또, 구조체 멤버 변수에 face의 중심이 있으므로 물체의 모든 face 중심의 좌표를 얻어낸다.

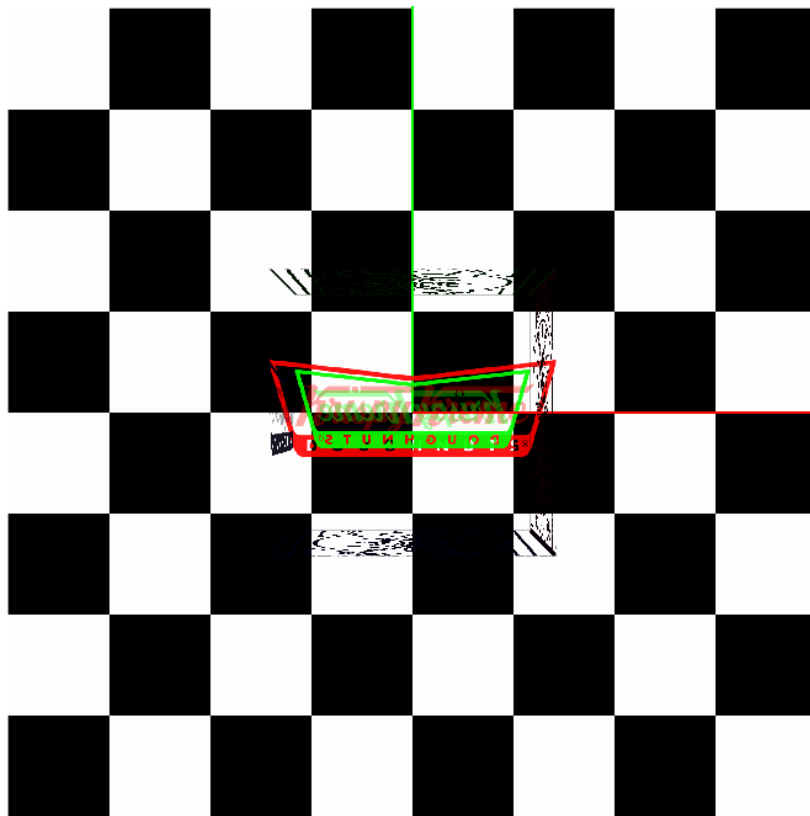
우선은 구조체 멤버 변수의 index를 임의로 설정해두고, face 중심의 좌표도 동차좌표로 저장한다.

Depth sorting을 위해 현재 model view matrix를 구해 face 중심의 좌표에 곱해 eye coordinate system상의 face center 좌표를 얻어낸다.

여기서 해당 구조체의 depth는 OpenGL에서 camera가 -z 방향을 바라보고 있으므로 eye coordinate system상의 face center 좌표의 -z값이 depth가 된다.

마지막으로 c++에서 지원하는 qsort와 depth 값이 크면 -1, depth 값이 작으면 1, depth 값이 같으면 0을 return하는 compare 함수를 사용하여 가장 먼 곳의 face가 가장 앞에 저장되어 sort되도록 한다.

Practice 03. Alpha textured cube without depth sorting:



Depth sorting을 적용하지 않은 alpha textured cube이다.

해당 그림의 경우 depth sorting이 적용되지 않았기 때문에 더 먼 곳에 있는 초록색 logo가 더 가까운 곳에 있는 빨간색 logo를 가리는 것을 볼 수 있다.

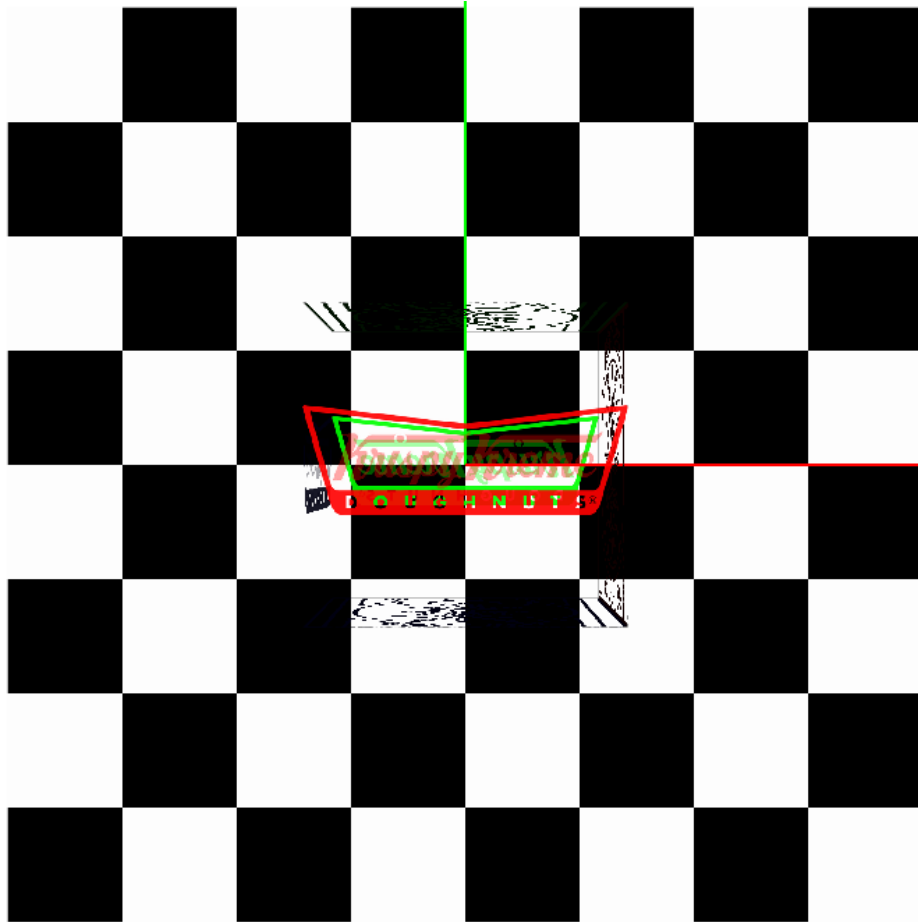
또한 alpha texture를 적용하는 방법으로는 GL_TEXTURE_2D와 GL_BLEND를 enable하고 glBlendFunc()을 사용해 불투명도인 alpha값으로 나타나는 $c = aC_f + (1-a)C_b$ 을 적용할 수 있다.

여기서 C_f 는 translucent foreground color, C_b 는 translucent background color이다.

그리고 glColor4f()를 사용해 RGB 외에도 alpha값을 지정해줘야 한다.

해당 예제에서는 texture가 luminance format이기 때문에 색상을 지정하기 위해 glPixelTransferf()를 사용해 GL_RED 또는 GL_GREEN, GL_BLUE의 BIAS를 0~1의 값으로 설정한다.

Exercise 01. Alpha textured cube with/without depth sorting:



Depth sorting이 적용된 alpha textured cube이다.

해당 예제에서 사용된 alpha texture를 적용하는 방법과 depth sorting을 적용하는 방법은 위에서 설명했으므로 생략하겠다.

위 그림을 보면 직전의 depth sorting이 적용되지 않은 예제와는 다르게 가까운 곳에 있는 빨간색의 logo가 더 먼 곳에 있는 초록색의 logo를 가리는 것을 볼 수 있다. 또한, 가까운 빨간색 logo를 투과하여 초록색 logo가 보이는 것도 확인할 수 있다.