

기초 컴퓨터 그래픽스

HW5 README

20191559 강상원

1. [환경 명세]

- Windows 11 64bit, Intel® Core™ i7-1255U CPU, Intel® Iris® Xe Graphics, Visual Studio Community 2022 Release x64

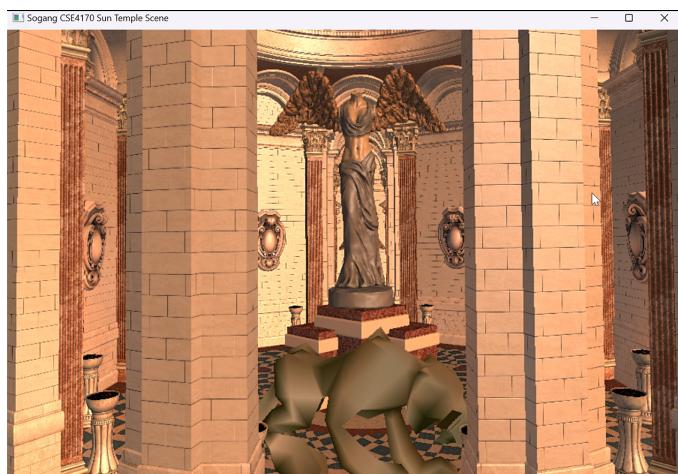
2. [요구사항]

1. 자신이 임의로 선정한 3D 물체 (텍스처 매핑 미적용)에 대하여 Gouraud shading과 Phong shading 효과를 적용하라. (어떤 물체인지를 명확히 기술한 후 '1' 키 (Gouraud shading)와 '2' 키 (Phong shading)를 사용하여 적용 방법을 선택할 수 있어야 하며, 시각적으로 분명히 두 효과의 차이를 구별할 수 있어야 한다.).

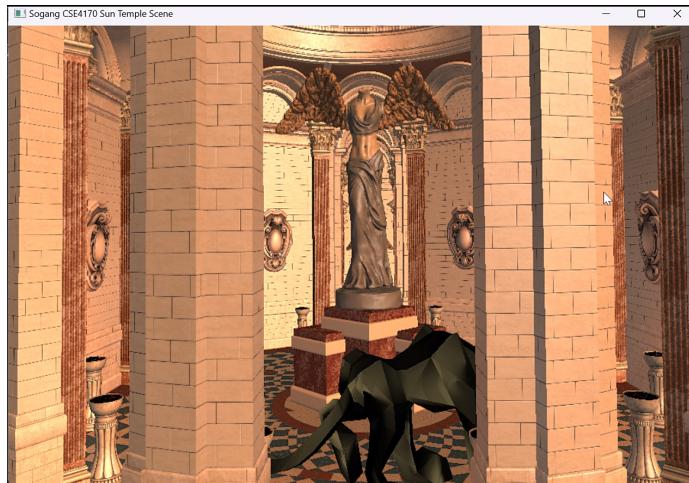
초기 카메라 (u)에서 내부를 돌고 있는 호랑이에 대하여 Gouraud shading과 Phong shading 효과를 적용하였다.

'1' 키를 누르면 텍스처 매핑은 미적용된, Gouraud shading을 적용한 호랑이가 되며, '2' 키를 누르면 Phong shading으로 변한다.

Gouraud shading



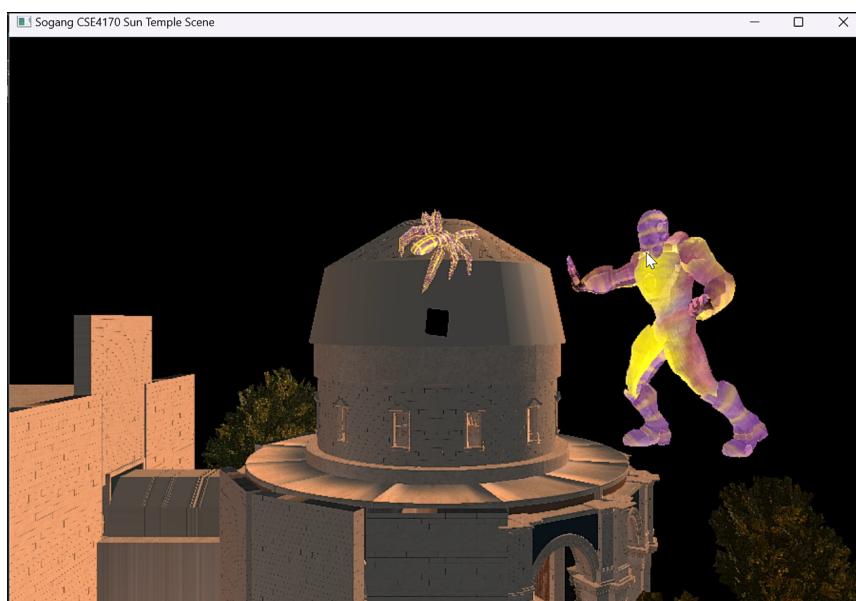
Phong Shading



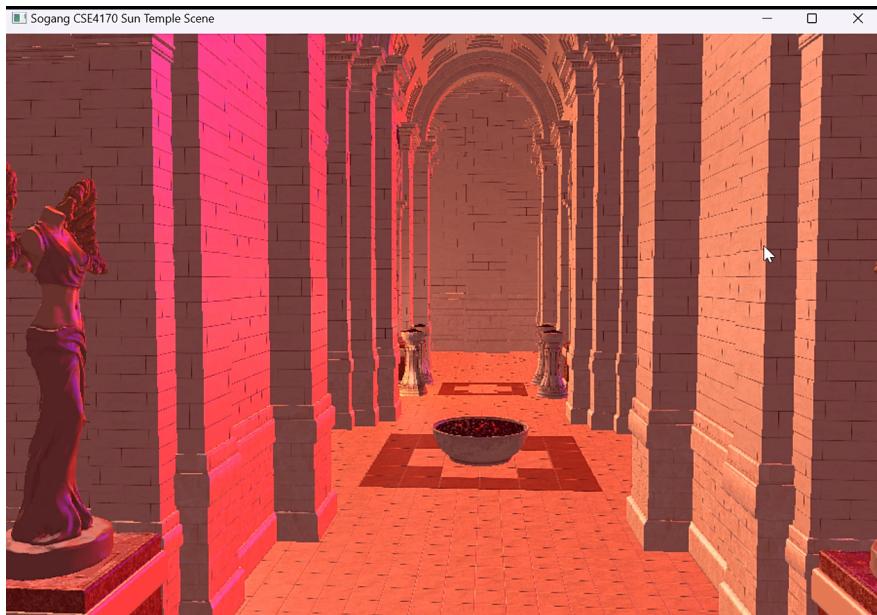
키 입력에 따른 차이를 극명화하기 위해 인자값을 조금 달리하여 색상과 반사율을 다르게 설정하였다. 위의 예시에서 보다시피 Gouraud shading을 적용하였을 때 특유의 끊어지는 그라데이션 모양을 확인할 수 있으며, Phong Shading 때 더 부드럽게 셰이딩이 적용된다.

2. 자신이 임의로 선정한 3D 물체 (정적인 물체와 동적인 물체 각각 한 개씩)에 대하여 자신이 촬영하거나 직접 만든 영상을 사용하여 텍스춰를 입혀라. (호랑이와 같이 예제 코드에서 사용한 물체는 제외함.)

'o'키를 눌러서 카메라를 전환하면 동적인 물체 spider, 정적 물체 ironman에 대하여 직접 만든 사진으로 텍스쳐를 입힌 모습을 볼 수 있다.



3. 세상 좌표계를 기준으로 최소한 한 개 이상의 광원을 배치하라. (이 광원은 '3' 키를 사용하여 동시에 on/off 할 수 있어야 한다.)

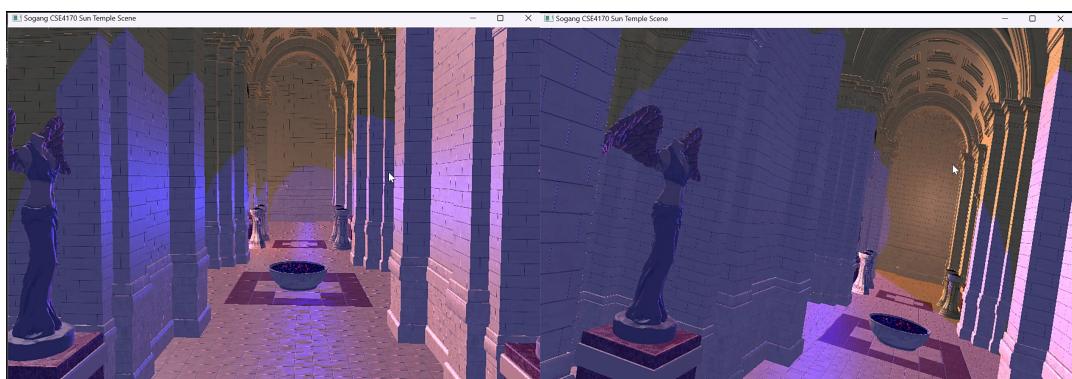


'1'키를 눌러 카메라를 전환하면 조명들을 관측할 수 있다.

'3' 키를 세상 좌표계에 고정되어 있는 붉은색 drop light가 떨어진다.

4. 움직이는 카메라 (카메라는 마우스 또는 키보드를 사용하여 조작)에 대하여 눈 좌표계를 기준으로 (즉 카메라에 상대적으로 고정되어 움직이는) 광원을 한 개 구현하라. (이 광원은 '4' 키를 사용하여 on/off 하며, 눈 좌표계에서 설정한 광원임을 분명히 알 수 있어야 한다.)

'4' 키를 누르면 파란색 스팟 라이트를 킬 수 있다. 화면에 마우스를 누른채로 드래그 (상하좌우)하여 회전하거나 화살표 키로 위치를 이동하면 스팟 라이트가 카메라 눈 좌표계를 따라 움직인다.



5. 움직이는 물체에 고정된 광원, 즉 그 물체의 모델링 좌표계에 배치한 광원을 한 개 구현하라. (이 광원은 '5' 키를 사용하여 on/off 하며, 모델링 좌표계에서 설정한 광원임을 분명히 알 수 있어야 한다.)

'o' 키를 눌러 움직이는 거미를 관측한다. '5' 키를 누르면 거미에 고정된 초록색 조명이 나온다. (거미의 등에서 나온다). 거미가 회전함에 따라 벽면, 아이언맨에 조명이 비침을 확인할 수 있다.



6. 육면체와 같이 다각형의 전후 관계가 알기 쉬운 닫힌 물체에 대하여 투명한 효과를 생성하라. (이 물체는 '6' 키를 사용하여 투명함을 on/off 할 수 있어야 하며, 투명한 상태에서 자신이 설정한 키를 사용하여 불투명도를 조절할 수 있어야 한다.) 이 물체는 자신의 무게 중심 둘레로 적절히 회전하고 있어야 하며, 불투명도를 조절할 때 그 효과가 분명히 보여야 한다. (즉 육면체 뒤쪽의 모습이 의도한 대로 보여야 한다.)

마찬가지로 'o' 카메라의 가운데에서 회전하고 있는 검은색 육면체를 관측할 수 있다.

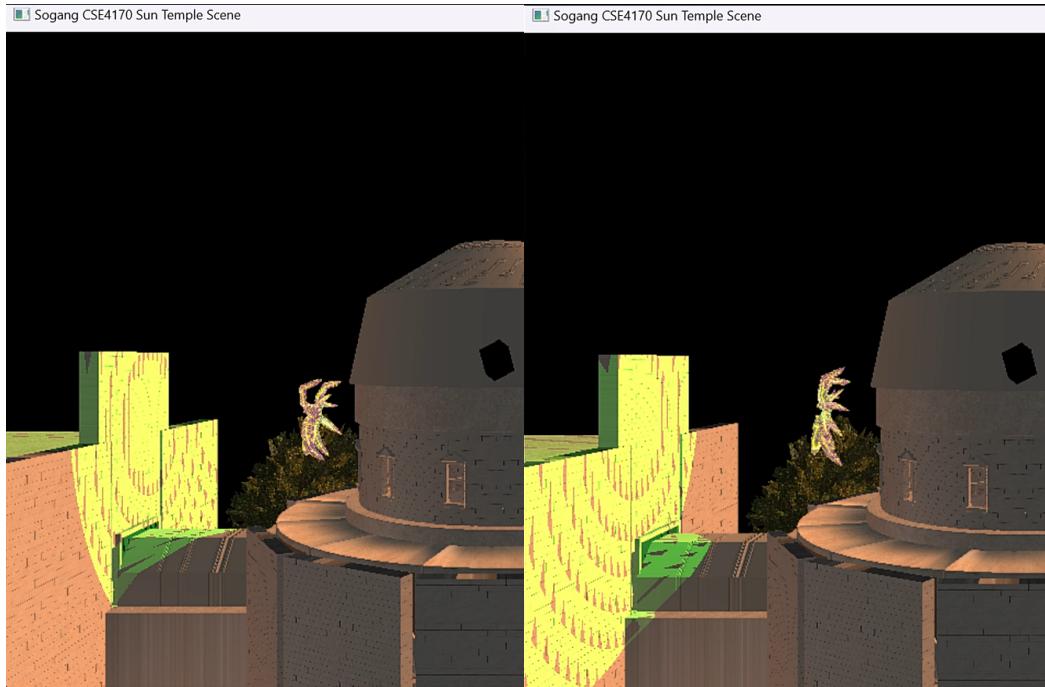
키를 사용한 불투명도 조절, 투명함 on/off는 시간 관계상 구현하지 못하였다.

7. '스크린 효과' 또는 '블라인드 광원'과 같은 재미있는 쉐이더 효과를 최대 3개까지 구현하라 (각각의 효과는 '7', '8', 그리고 '9' 키를 사용하여 on/off 할 수 있어야 함). 당연히 이 두효과와는 전혀 다른 자신만의 창의력을 발휘한 3D 효과이어야 하며, 무엇을 대략적으로 어떻게 구현했는지 README 파일에 기술하라. (주의: 이 쉐이더 효과는 이번 숙제에서 제공하는 기반 코드 문맥에서 생성을 해야 하며, 그렇지 않을 경우 점수를 부여하지 않을 수도 있음.)

마찬가지로 'o' 카메라에서, '5' 키로 거미의 등에서 나오는 초록색 공원을 켜 상태에서 관

측하는 것이 가장 편리하다.

- 체크 효과

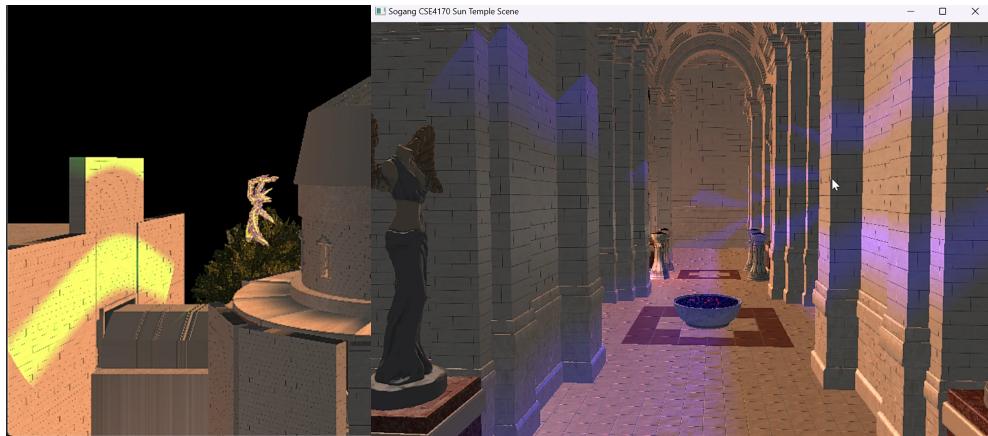


'7' 키를 눌러 해당 쉐이딩 효과를 키면 위와 같이 스팟 라이트에 대하여 원형 체크 무늬 효과를 적용한다.

```
// check effect
vec2 checkPos = fract(L_EC.xz * 20.0); // 10.0 is the scale of
the checkerboard
float checker = step(0.5, mod(checkPos.x + checkPos.y, 10.0));
tmp_float *= checker;
```

조명 방향의 x와 z 성분을 사용하여 체크 패턴을 생성한다. `fract` 함수는 입력 값의 소수부분을 반환하므로, `L_EC.xz * 20.0` 계산 결과는 0과 1 사이의 값이 된다. 이 값이 체크 패턴의 스케일을 제어한다. `mod` 함수를 사용하여 이 값들의 합을 10으로 나눈 나머지를 구하고, `step` 함수를 사용하여 이 값이 0.5보다 크면 1, 그렇지 않으면 0을 반환하게 한다. 이는 체크 패턴의 각 체크에 해당하는 값이다.

- 웨이브 효과

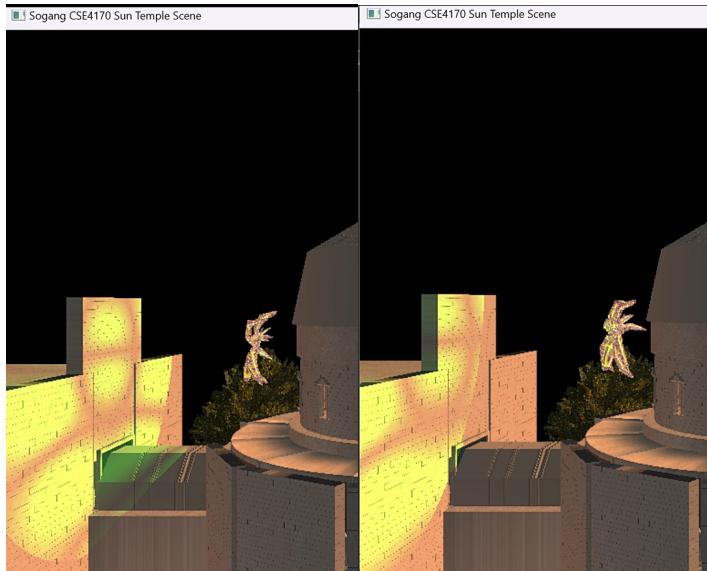


'8' 키를 눌러 해당 쉐이딩 효과를 키면 위와 같이 스팟 라이트에 대하여 웨이브 무늬 효과를 적용한다.

```
// wave effect
float angle = atan(L_EC.y, L_EC.x); // Compute angle in [0, 2pi]
float radius = length(L_EC.xy); // Compute distance to the center
float pattern = cos(radius*10.0 + angle*10.0); // 10.0 control the
frequency of the pattern
tmp_float *= pattern;
```

`L_EC`의 `x`와 `y` 성분을 사용하여 각도와 거리를 계산하고, 코사인 함수를 사용하여 웨이브 패턴을 생성한다. 코사인 함수에 더해지는 각도와 거리를 조절함으로써 패턴의 빈도와 위상을 제어한다.

- 물빛 효과 (수영장 빛 반사 느낌)



'9' 키를 눌러 해당 쉐이딩 효과를 적용하면 위와 같이 스팟 라이트에 대하여 수영장

바닥에 비친 물빛 무늬와 같은 빛 무늬가 생성된다.



```
// water effect
float gridFreqX = 10.0; // Controls the frequency of the grid in
the x direction
float gridFreqY = 10.0; // Controls the frequency of the grid in
the y direction
float gridPattern = sin(L_EC.x * gridFreqX) * sin(L_EC.y *
gridFreqY);
tmp_float *= abs(gridPattern);
if (tmp_float < zero_f)
    tmp_float = zero_f;
```

격자 패턴의 웨이브를 생성한다. 각 격자는 L_EC의 x와 y 성분에 대해 사인 함수를 적용하여 생성한다. 이 함수의 빈도는 gridFreqX와 gridFreqY로 제어된다. 이 두 값이 커지면 격자는 더 밀집하게 된다. 사인 함수는 -1과 1 사이의 값을 반환하므로, 이를 절댓값으로 변환하여 최종적인 빛의 강도를 양수로 유지한다.

각 이펙트는 tmp_float에 곱해져 결과적으로 빛의 강도를 변화시킨다. 이 코드는 각 픽셀에 적용되므로, 효과는 조명 아래의 모든 픽셀에 영향을 미치게 된다.