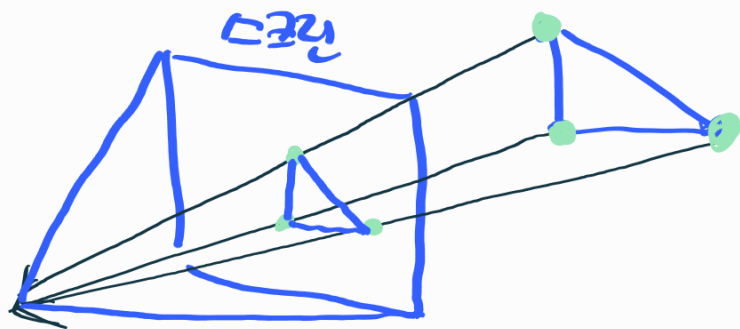


래스터화가 빠른 이유

Rasterization에서는 3차원 공간상의 물체를 표현할 때 여러개의 삼각형을 사용한다.

장점 - 여러 삼각형을 각각 따로 렌더링하기에 적당하다
래스터화

1) Project Vertices - 삼각형의 Vertices를 스크린에 투영(이동)한다
3차원 공간의 삼각형 (vertex 3개로 정의)



Raytracing은 시청에서 Ray를 쏘아서 물체나 빛의 반사지 알기 위하여 모든 스크린 Pixel에 Ray를 쏘아본다.

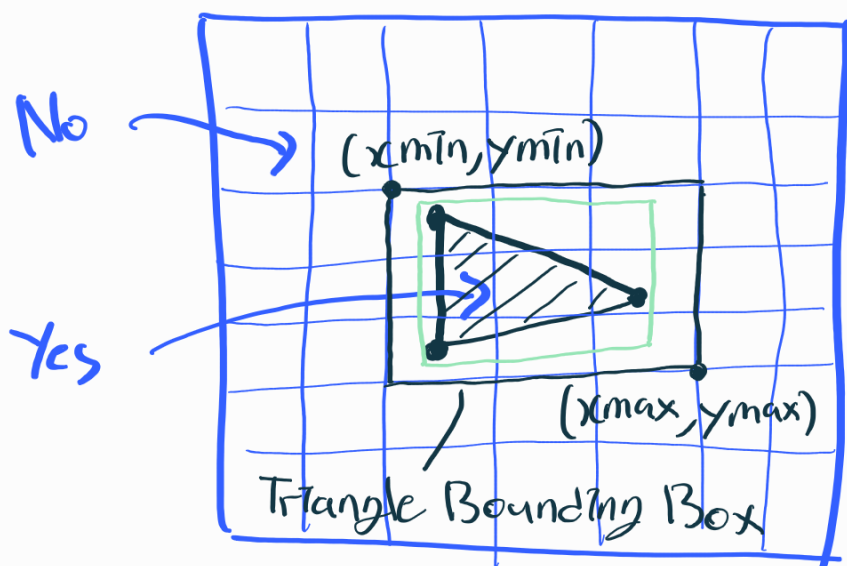
Rasterization에서는 삼각형을 스크린에 투영(Project)시킨다.

즉, 삼각형의 정점(vertices)을 스크린 좌표로 옮긴다는 의미다.

렌더링하는 스크린에 삼각형의 Vertices의 위치를 알면 삼각형을 그릴 수 있다.

2) 픽셀의 색 결정

Loop over Pixels. Does the pixel lie in the triangle?



결과적으로 Pixel의 색깔값을 정할 수 있다.

어떤 Pixel이 삼각형 안에 들어 있는지 없는지 판단하여 Pixel이 삼각형 밖이면 넘어가고 Pixel이 삼각형 안에 있다면 픽셀의 색을 삼각형에서 가져온다.

Bounding Box - 삼각형의 세 vertices를 검사한 가장 작은 직사각형을 찾아 Loop를 돌면 이 밖의 Pixel들은 삼각형 밖이므로 예외로 하면 계산의 최적화가 가능하다.

(x_{min}, y_{min}) = 좌측상단좌표, (x_{max}, y_{max}) = 우측하단좌표 \Rightarrow 박스의 1번좌표

폴리곤 (Polygon)

2D 픽셀이 아닌 3차원 모델 생성시 삼각형만 사용하거나 아닌 사각형에
오각형, 육각형까지도 사용한다. (예) 오각형으로 만든 축구공 등)

일반적으로 사각형으로 모델링할 때도 관점에서 폴리곤 모델이라고 한다.
(폴리곤 메시(mesh), 삼각형 메시, 삼각 폴리곤 메쉬라고도 한다)

Raytracing vs Rasterization

Raytracing (레이 트레이싱)

for Pixel
for Object
색 결정

Rasterization (래스터화)

for Object
for Pixel
색 결정