

cmd로 도넛만들기

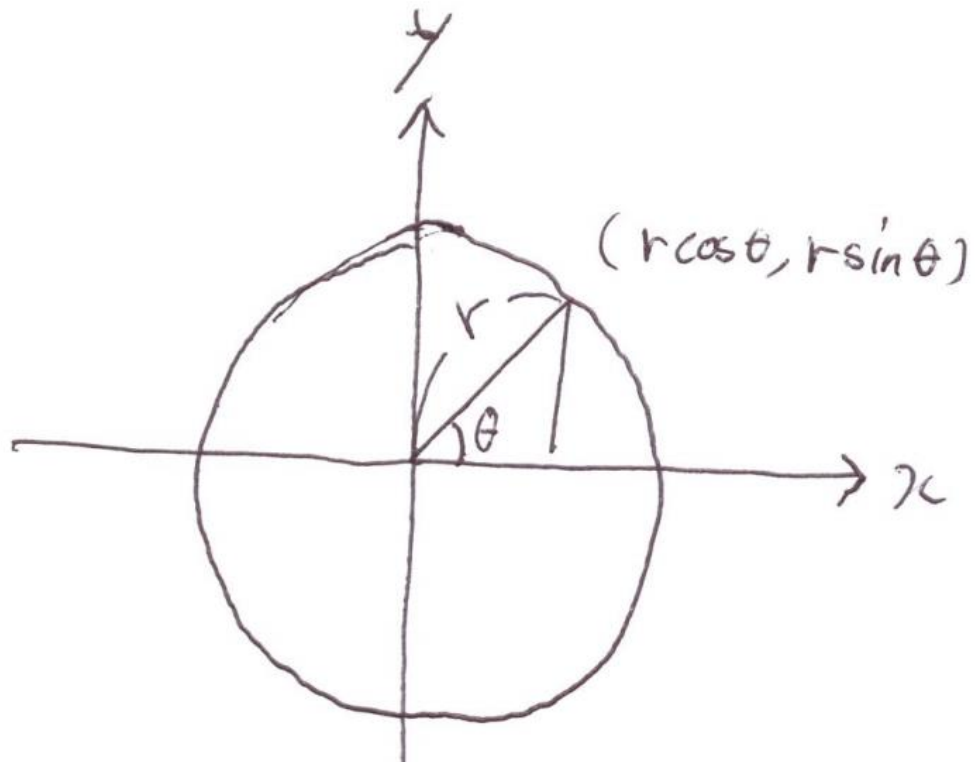
본 프로젝트의 목적은 기본적인 수치해석의 공식을 이용하여 도형을 출력하는 것 입니다.

프로젝트의 동작방식은 다음과 같습니다.

1. 3차원 공간에 시작원을 하나 그립니다.
2. 원을 회전시켜 도넛을 얻습니다.
3. 도넛을 스크린으로 옮깁니다. (3D->2D)
4. 스크린 상에서 해당 위치의 밝기를 계산하여 적용합니다.

1. 3차원 공간에 시작원을 하나 그립니다.

원의 방정식을 알고있다면 이것은 전혀 어렵지 않습니다.

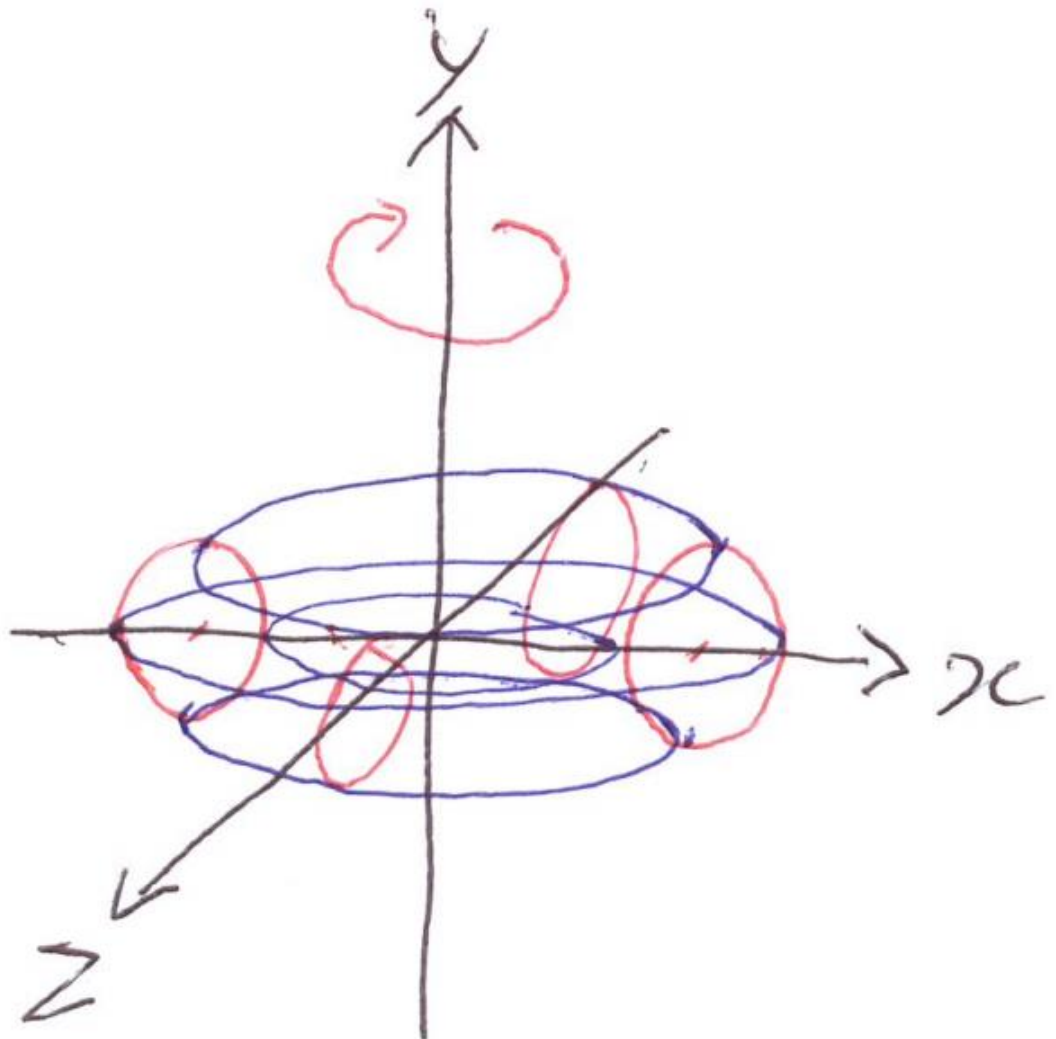


저는

이 원의 중심을 (100, 0, 0)에 두고 xy평면위로 r(반지름)을 40으로 설정한 후 θ 를 0부터 0.07씩 6.28(약 2π)까지 증가시키며 연속된 점으로 원을 표현하였습니다.

물론 xy평면에 시작원을 두지 않아도 괜찮습니다. 하지만 저는 2번작업의 편의성과 연산의 직관성을 위해 이렇게 설정하였습니다.

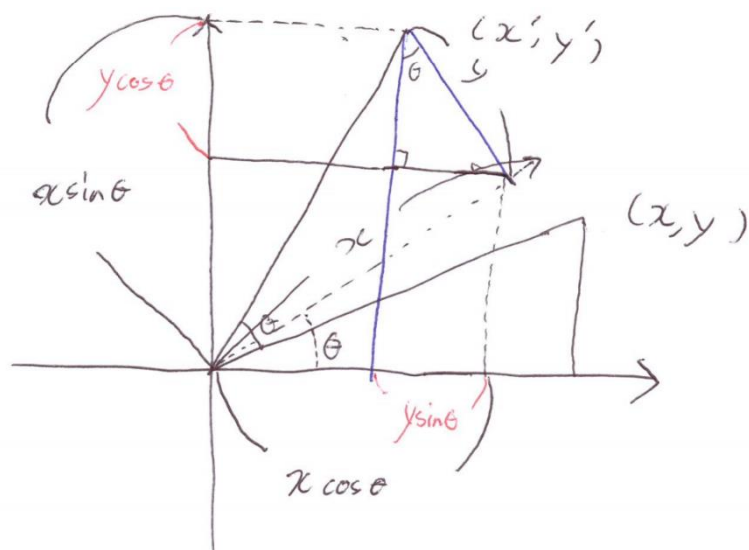
2. 원의 중심을 회전시켜 도넛을 연습니다.



그림이 조금 지저분하지만 위의 그림처럼 y 축을 기준으로 표현된 원을 회전하면 3차원 공간위로 도넛을 그릴 수 있습니다.

축회전은 행렬 곱셈을 통해 구현할 수 있습니다.

다음은 z 축회전을 위해 xy 평면의 요소 (x, y) 를 θ 만큼 회전할 때 (x', y') 를 구하는 원리입니다.



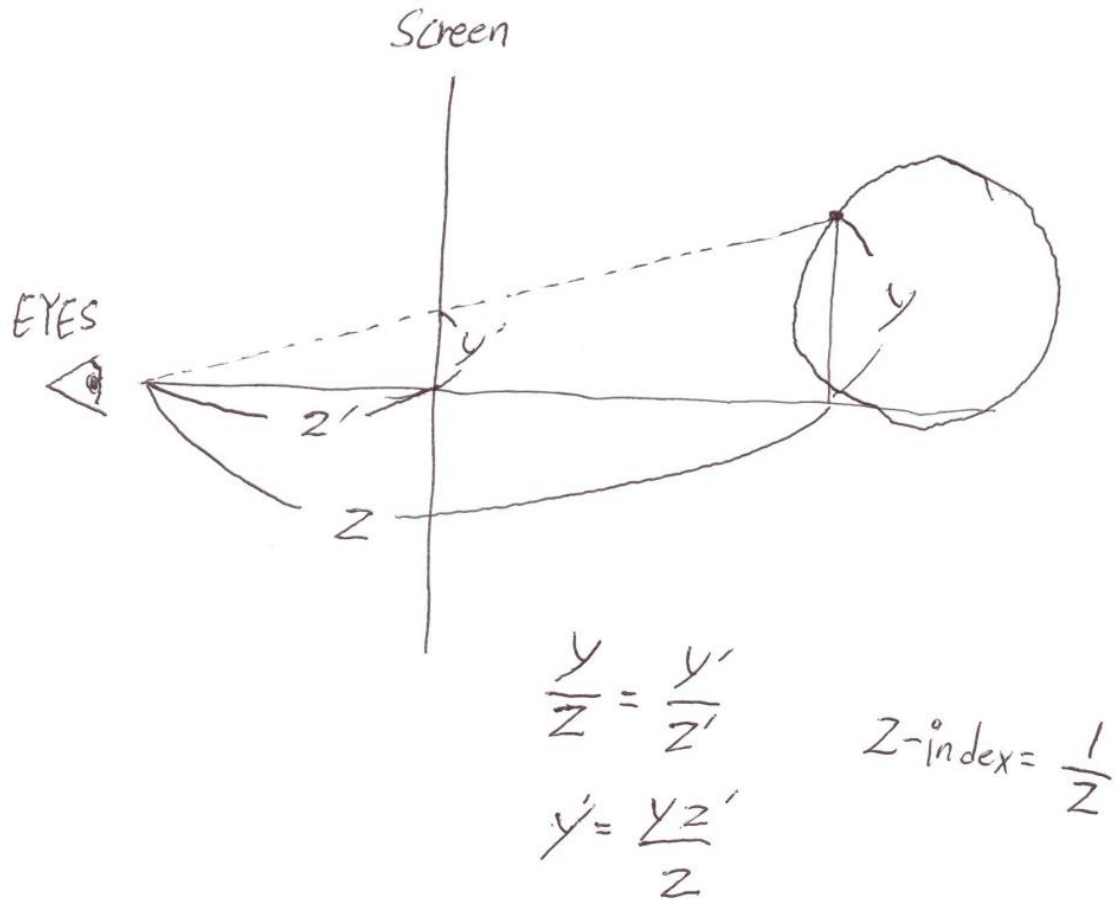
$$x' = x \cos \theta - y \sin \theta$$

$$y' = x \sin \theta + y \cos \theta$$

이를 이용하면 원의 좌표들을 이용해 3D도넛형태의 물체를 만들 수 있습니다.

3. 도넛을 스크린으로 옮깁니다. (3D->2D)

3D 프로젝션을 이용합니다 원리는 다음과 같습니다.



실제 물체의 위치 y 를 이용하여 스크린상의 표시위치 y' 로 바꿉니다.

Z-index는 $1/z$ 로 이를 이용하여 스크린 상으로는 물체의 정면을 인식하게 됩니다.

좌표평면 상에서 시점의 위치는 -300으로 설정했습니다.

4. 스크린 상에서 해당 위치의 밝기를 계산하여 적용합니다.

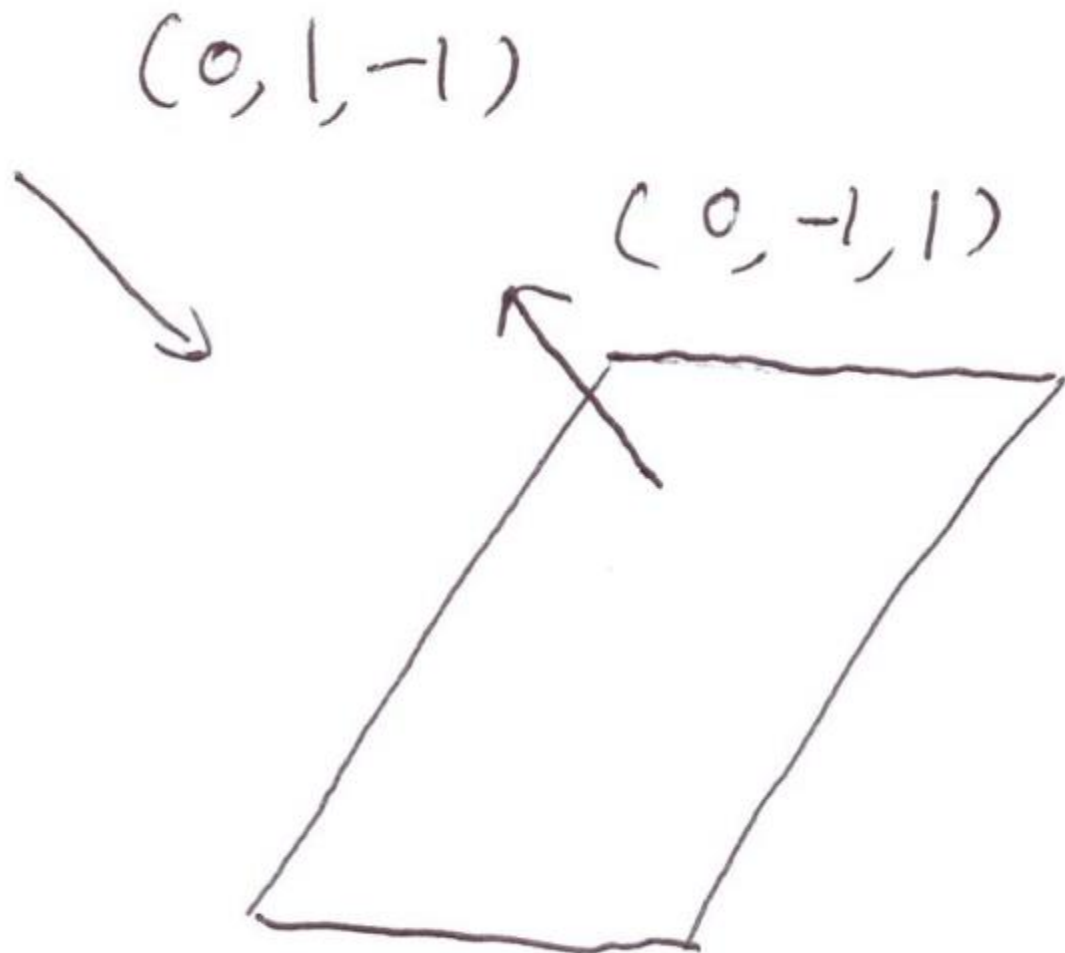
위 프로젝트에서 밝기란 빛의 반사 정도를 나타내는 것으로 계산하였습니다.

이를 광원의 벡터((0, 1, -1)빛의 방향)와 도넛 표면의 법선벡터의 사이각을 통해 계산합니다.

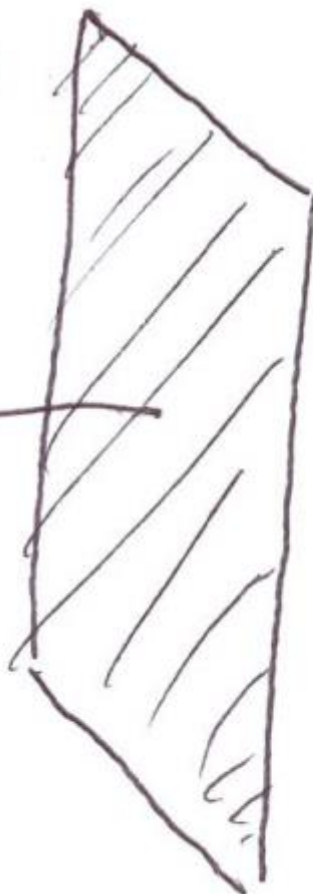
이는 벡터의 Dot Product를 이용해 계산할 수 있습니다.

출력시에 타일의 밀도가 높아질수록 밝게 표현되므로 Dot Product의 결과값에 비례하여 총 12단계의 밝기로 구분합니다.

```
Bright[13] = ".,-~:;!*=#$@";
```



$(0, 1, -1)$



$(0, 1, 0)$

용어

Frame-buffer: 스크린에 그려질 정보를 담는 버퍼

Z-buffer : 3차원 그래픽의 심도좌표 관리방식, z축의 위치값, 픽셀의 깊이정보

수치해석

휘도(illuminance) : 물체가 갖는 빛의 반사 세기, 이것을 통해 물체를 인식한다. 본 프로젝트에서는 광원의 벡터와 물체의 법선벡터의 기울기차이로 반사량을 측정하였음.

축회전 : 축을 중심으로 회전하는 방식, 편의성을 위해 모든회전은 (0,0,0) 원점을 기준으로 실행하였다.