

Normal Vector 변환

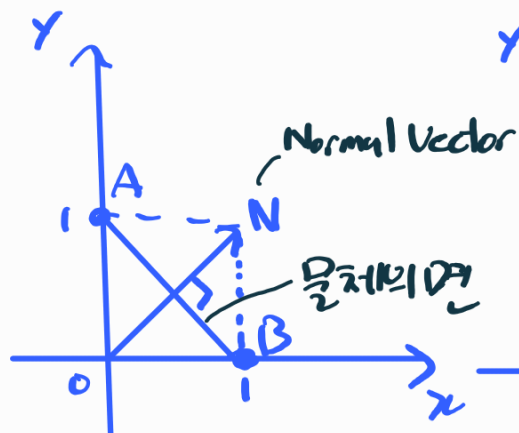
물체가 회전하면 물체 표면이 어떤 방향으로 향하고 있는지를 가리키는 Normal Vector로 회전시켜줘야 한다.

하지만 Non-Uniform scale을 하면 Normal Vector의 의미가 훼손된다.

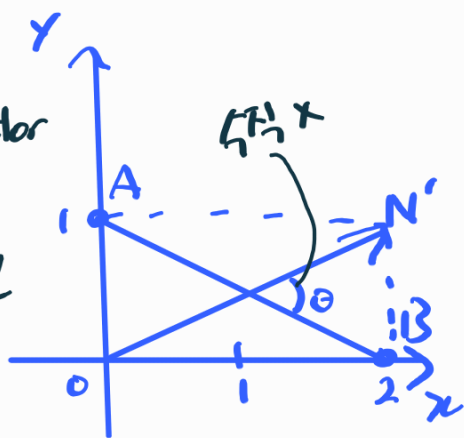
(Uniform Scale은 3D 모델을 기준으로 x, y, z 값이 같은 것을 의미한다.)

Ex: (1, 1, 1), (5, 5, 5) 등

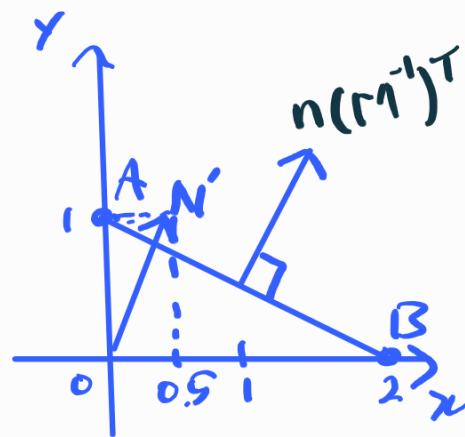
그래서 Normal Vector를 올바른 방향으로 맞추고 이해한다.



a)



b) Non-Uniform Scaling
(Normal Vector가 수직x)



c) 실제 Scaling 후
필요한
Normal Vector

선분 AB = 물체의 면
N = Normal Vector

a에 Non-Uniform Scaling인 $M = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ 가만 2배로 하면
b가 된다.

똑같은 M을 Normal Vector에 적용하면 b의 N'이 된다.

그러면 더 이상 Normal Vector가 물체의 면 AB에 수직이 아니게 된다.

c는 a에 M을 적용한 뒤 실제로 필요한 Normal Vector다.

변환된 물체의 Normal Vector는 원래 Normal Vector에 $(M^{-1})^T$ 를
적용시켜 주어야 한다. (역행렬의 Transpose 적용)

(Normal Vector를 알아야 조명효과를 제대로 적용 가능해진다, shading)

이때 M의 Translation(이동)을 $vec4(0.f, 0.f, 0.f, 1.f)$ 즉
0으로 만들어준다. Normal Vector는 벡터이기때문에 이동하지 않으나
0으로 바꿔주면 수치적 ERROR를 줄일 수 있다.

Normal Vector 변환 유즈

$u \cdot n = 0$, 원래 Normal Vector인 물체의 면을 의미하는 u 는 스칼라

$u n^T = 0$, dot product를 행렬 형태로 변환

$u(AA^{-1})n^T = 0$, $AA^{-1} = I$, Identity Matrix

$(uA)(A^{-1}n^T) = 0$, 행렬 연산에서 () 바꾸리기

$(uA)((A^{-1}n^T)^T)^T = 0$, $(A^T)^T = A$

$(uA)(n(A^{-1})^T)^T = 0$, $(AB^T)^T = B^T A^T$

$uA \cdot n(A^{-1})^T = 0$, 행렬 연산을 Dot Product로 변환

$uA \cdot nB = 0$, 원래 u 에 A 변환을 적용한 벡터와

normal vector에 $(A^{-1})^T = B$ 를 적용하면 서로 수직이다.