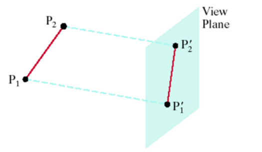
6. Projection Transformation

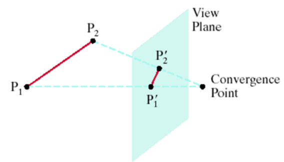
> 💡 **Projection Transformation**
> - **Orthographic Projection**
> - **Perspective Projection**

Projection Transformation

- **Parallel Projection**   수업에선 parallel = orthographic
  - **평행선 보존**됨
    - Orthographic Projection: Projection 방향이 VPN 과 평행
    - Oblique Projection: Projection 방향이 VPN 과 평행하지 않음



- **Perspective Projection**
  - **평행선 보존** 안 됨
  - 소실점 향해서 projection

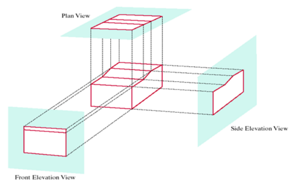


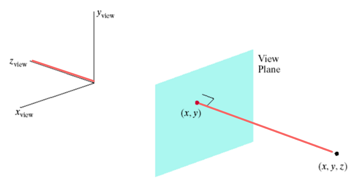
Orthographic Projection

> 💡 Viewing Coordinate System 상의 점 (x, y, z) 가 View Plane 에 Projection 되었을 때, 점 (x_p, y_p) 좌표를 구해보자

- View Plane 의 수직 벡터인 VPN = **n** = z_view 과, Projection 방향이 평행 이므로
  - $x\_p = x$
  - $y\_p = y$

  z 만 날리면 됨

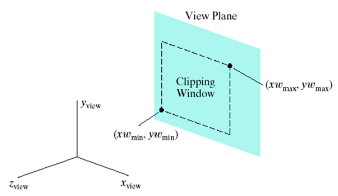




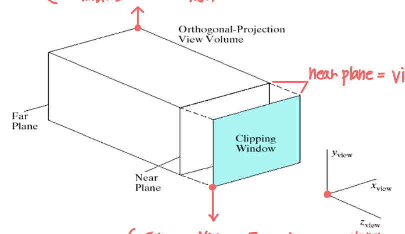
$(x, y, z) \to (x, y)$

> 💡
> - Projection Transformation 할 때, **View Volume** 정하고 **Normalize** 도 해야 한다.
>   - cf) 뒤에서 배우지만, vv 밖을 잘라주는 **clipping** 도 해야 한다. (Near clipping plane, Far clipping plane)
> - Orthogonal Projection 에서는 vv 모양이 Rectangular Parallelpipe 이다.

1. **View Volume 정의**: vv = **Rectangular Parallelpipe** 을 두 점의 좌표로 정의
   min, max



$(xw_{max}, yw_{max}, z_{far})$



near plane = view plane

오브젝트를 쳐다보는 방향이 -z 방향
z_near, z_far 둘 다 음수

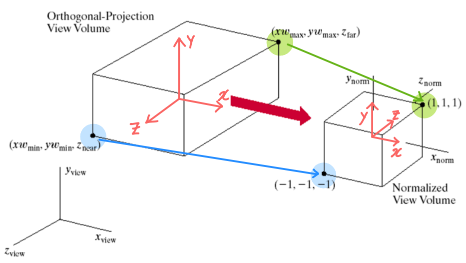
$(xw_{min}, yw_{min}, z_{near})$ eye, 카메라

2. **vv Normalization**

> 💡 Normalize 하는 이유
> 1. Clipping Algorithm 의 최적화 (정해진 규격만의 Clipping 알고리즘 짜면 되니까)
> 2. 최종적으로는 Near plane 이 Target Display 사이즈에 맞도록 Scaling 할 것이다 (Viewport Transformation)
>   이때 이미 Normalize 되어 있으면 Scaling 이 편할 것

- (이해)
1. vv 의 중앙을 원점으로 ⇒ Translation
2. vv 의 크기를 2x2×2 로 ⇒ Scaling

(1), (2) 를 만족 시키려면...



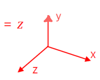
$$\mathbf{M}_{\text{ortho, norm}} \begin{bmatrix} xw_{min} \\ yw_{min} \\ z_{near} \\ 1 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \\ -1 \\ 1 \end{bmatrix}, \mathbf{M}_{\text{ortho, norm}} \begin{bmatrix} xw_{max} \\ yw_{max} \\ z_{far} \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$\mathbf{M}_{\text{ortho,norm}} = \begin{bmatrix} \overset{\text{(ㄱ)}}{\frac{2}{xw_{max}-xw_{min}}} & 0 & 0 & \overset{\text{(ㄹ)}}{-\frac{xw_{max}+xw_{min}}{xw_{max}-xw_{min}}} \\ 0 & \overset{\text{(ㄴ)}}{\frac{2}{yw_{max}-yw_{min}}} & 0 & -\frac{yw_{max}+yw_{min}}{yw_{max}-yw_{min}} \\ 0 & 0 & \overset{\text{(ㄷ)}}{\frac{-2}{z_{near}-z_{far}}} & \frac{z_{near}+z_{far}}{z_{near}-z_{far}} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

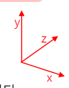
scaling   translation

(이해)
(ㄱ) : 길이가 xw_{max} - xw_{min} 인 걸 2 로 scale 한 것
(ㄴ) : 길이가 yw_{max} – yw_{min} 인 걸 2 로 scale 한 것
(ㄷ) : ~~z_{near}, z_{far} 는 음수~~ ⇒ ~~$z_{near} > z_{far}$~~ ⇒ ~~z_{near} - z_{far}~~ 는 양수
Left hand coordinate (LHC) 로 바꿔주려면 z flip 필요 →
z_{near} - z_{far} 를 길이 2 로 바꿔주고, - 붙여준다.
(ㄹ) : 앞의 두 식을 풀어서 계산한 것

viewing coordinate 까지는 right hand coordinate. $x \times y = z$

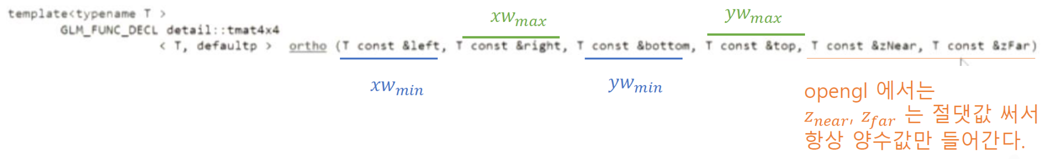


projection 하고서 normalize 할 땐 left hand coordinate. $x \times y = -z$



왜?
depth buffer 에서 눈으로부터 거리로
보이고 안 보이고를 결정한다 (depth 가 더 작은 것만 보인다)
이때 '길이' 가 나타내는 z 값이 음수가 아닌 양수인게 직관적이다.

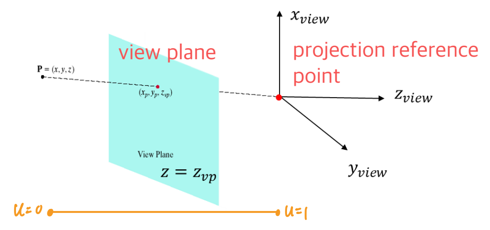
- **glm::ortho**



Perspective Projection

💡 눈으로부터 얼마나 떨어져 있냐에 따라 **물체의 크기가 다르게 보이는 원근감**을 가지는 projection

- 공간 상의 모든 물체가 원점(소실점)을 향해 projection
- **Projection reference point = 눈의 위치 = (0, 0, 0)**
- View plane = projection 할 plane



💡 P(x, y, z) 가 (0, 0, 0) 를 향해 View plane 으로 projection 될 때, 점 (x', y', z') 을 구해보자

1. 매개변수 u 를 이용하여 표현 (0 ≤ u ≤ 1)
2. ***z' = z_vp* = (1 - *u*)*z*** 이므로 u 값 구하기
3. u 대입해서 x', y' 구하기

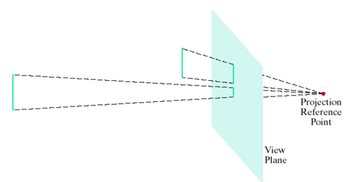
1. $\begin{cases} x' = (1-u)x \\ y' = (1-u)y \quad 0 \le u \le 1 \\ z' = (1-u)z \end{cases}$

2. $u = 1 - \dfrac{z_{vp}}{z}$

3. $x_p = \dfrac{z_{vp}}{z}x$

$y_p = \dfrac{z_{vp}}{z}y$

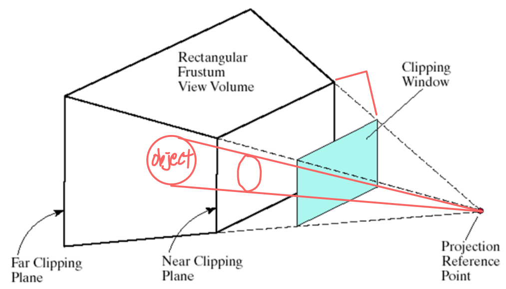
💡 식의 의미
원점으로부터 거리가 **멀수록 (z 가 클수록) *x_p, y_p* 값이 줄어들고,**
거리가 **가까울수록 (z 가 작을수록) *x_p, y_p* 값이 커짐**
⇒ perspectivity



View Volume

(OpenGL 경우, Clipping Window 가 Near Plane 과 일치하기 때문에 Clipping Window 는 무시)

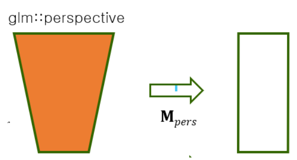
- **View frustum**
  - View volume 의 모양이 피라미드 형태
  - 필요한 파라미터 : ***z_near, z_far*, FOV**
- 공간 상 물체들은 near plane 으로 projection
  - near plane = view plane
- ***z = z_vp = z_near* < 0**
  - Normalize 전에는 **right hand coordinate** 이므로, z 가 증가하는 방향(+)이 쳐다보는 방향(-)의 반대
- View volume 을 Normalize 하며 정육면체 형태로 바꾸어야 함



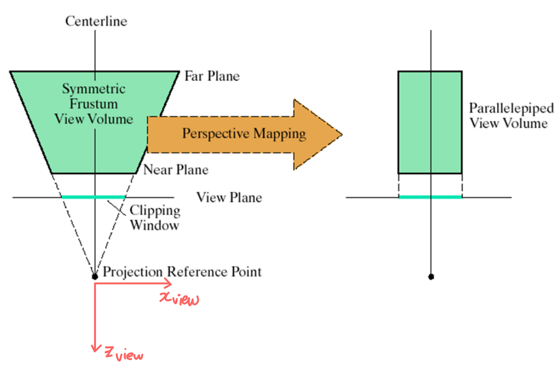
💡
- Parallel Projection 에서는 **깊이와 관계없이 객체 크기가 일정하게 유지**되기 때문에, **z 에 대한 의존성이 없음**
- Perspective Projection 에서는 깊이값 z 에 따라 객체의 크기가 달라지므로, **z 에 의존적인 계산**이 필요

Perspective Projection 을 **Parallel Projection** 과 같은 방식으로 **다룰 수 있게 하여 계산을 단순화**
⇒ **Perspective Mapping**

- Fruastum → Parallelpiped → 정육면체
  - 결국, Fraustum shape 을 정육면체 형태로 바꿔야 함
  - **Fraustum → Parallelpiped** (**Perspective Mapping**)



    - 위에서 다룬, Perspecive projection 에서 view plane 에 projection 한 점을 구하는 방법 이용
  - Parallelpiped → 정육면체
    - 위에서 다룬, Orthographic projection 에서 Normalization 하는 방법 이용



Perspective division: 원근감을 보정해주는 h 로 (x_h, y_h, z_h) 를 나누는 것

$$x_p = \frac{x z_{near}}{z} = \frac{x_h}{h}$$
$$y_p = \frac{y z_{near}}{z} = \frac{y_h}{h}$$

$$\xrightarrow{\text{행렬형태}}$$

$$\mathbf{M}_{\text{pers}} = \begin{bmatrix} -z_{near} & 0 & 0 & 0 \\ 0 & -z_{near} & 0 & 0 \\ 0 & 0 & s_z & t_z \\ 0 & 0 & -1 & 0 \end{bmatrix} \begin{matrix} x \\ y \\ z \\ 1 \end{matrix} \qquad \begin{matrix} x_h \\ y_h \\ z_h \\ h \end{matrix}$$

$\neq 1$ $\qquad$ $h \neq 1$

$$\mathbf{P}_h = \mathbf{M}_{\text{pers}}\mathbf{P}$$

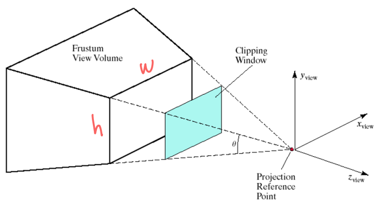
$$x_h = x(-z_{near}), \quad y_h = y(-z_{near})$$
$$h = -z$$

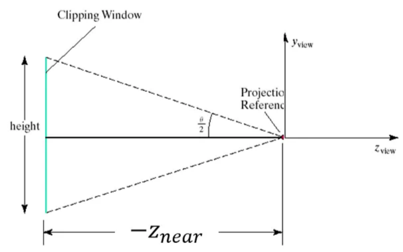
즉, **M_pers** 는 projection 되는 점을 구하는 행렬

- s_z 는 z 축 scaling, t_z 는 z 축 translation 에 사용 (Normalize 때 필요, 밑에서 배움)
- Affine transformation 아님

View volume 이 **symmetric** 인 경우

- View volume 이 x, y 축 방향으로 **symmetric 한 형태**
- **glm::perspective(fovy, aspect, |znear|, |zfar|), aspect=width/height**
  - fovy: view volume 의 y 방향 각도
  - aspect: 너비와 높이의 비율
  - znear: near plane 까지의 거리
  - zfar: far plane 까지의 거리



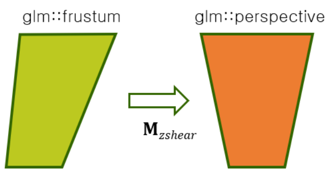


$$\tan\left(\frac{\theta}{2}\right) = \frac{\text{height}/2}{-z_{near}} \qquad \text{height} = -2z_{near}\tan\left(\frac{\theta}{2}\right)$$

$$-z_{near} = \frac{\text{height}}{2}\cot\frac{\theta}{2} = \frac{\text{width}}{2\text{aspect}}\cot\frac{\theta}{2}$$

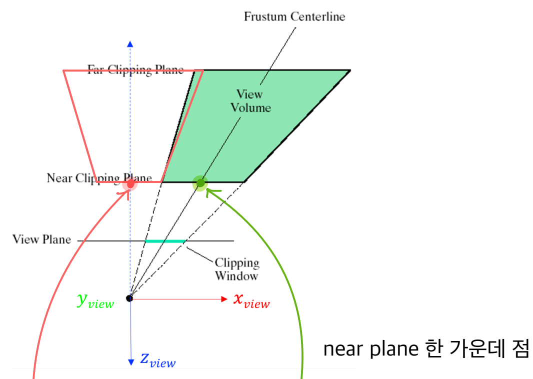
View volume 이 **oblique** 인 경우

- View volume 이 x, y 축 방향으로 **symmetric 하지 않은 형태**
- **glm::frustum(xwmin, xwmax, ywmin, ywmax, |znear|, |zfar|)**
  - xwmin, xwmax: x 방향 최소, 최댓값
  - ywmin, ywmax: y 방향 최소, 최댓값
  - znear, zfar 는 동일
- **Frustum → symmertric**



  - shearing transformation(왜곡) 사용
  - 이 frustum 은 x, y 방향으로만 왜곡되어 있음

$$\mathbf{M}_{z\,\text{shear}} = \begin{bmatrix} 1 & 0 & \text{sh}_{zx} & 0 \\ 0 & 1 & \text{sh}_{zy} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

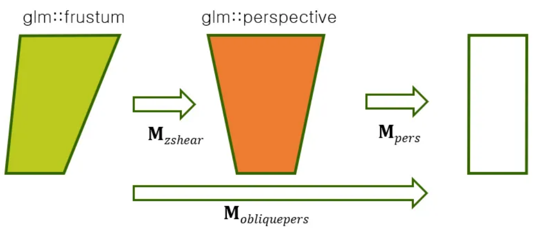


$$\begin{bmatrix} 0 \\ 0 \\ z_{\text{near}} \\ 1 \end{bmatrix} = \mathbf{M}_{z\,\text{shear}} \cdot \begin{bmatrix} \frac{xw_{\min} + xw_{\max}}{2} \\ \frac{yw_{\min} + yw_{\max}}{2} \\ z_{\text{near}} \\ 1 \end{bmatrix}$$

$$\downarrow$$

$$sh_{zx} = \frac{xw_{min} + xw_{max}}{-2z_{near}}, \quad sh_{zy} = \frac{yw_{min} + yw_{max}}{-2z_{near}}$$

- (정리) frustum → symmertric (**M_zshear**) ⇒ symmertric → Parallelpiped (**M_pers**)
- 이걸 한 번에, **M_obliquepers**



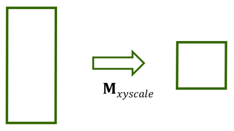
$$\mathbf{M}_{\text{obliquepers}} = \mathbf{M}_{\text{pers}} \cdot \mathbf{M}_{z\,\text{shear}}$$
$$= \begin{bmatrix} -z_{\text{near}} & 0 & \frac{xw_{\min} + xw_{\max}}{2} & 0 \\ 0 & -z_{\text{near}} & \frac{yw_{\min} + yw_{\max}}{2} & 0 \\ 0 & 0 & s_z & t_z \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

$$\mathbf{M}_{\text{pers}} = \begin{bmatrix} -z_{\text{near}} & 0 & 0 & 0 \\ 0 & -z_{\text{near}} & 0 & 0 \\ 0 & 0 & s_z & t_z \\ 0 & 0 & -1 & 0 \end{bmatrix} \qquad \mathbf{M}_{z\,\text{shear}} = \begin{bmatrix} 1 & 0 & \text{sh}_{zx} & 0 \\ 0 & 1 & \text{sh}_{zy} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Normalization

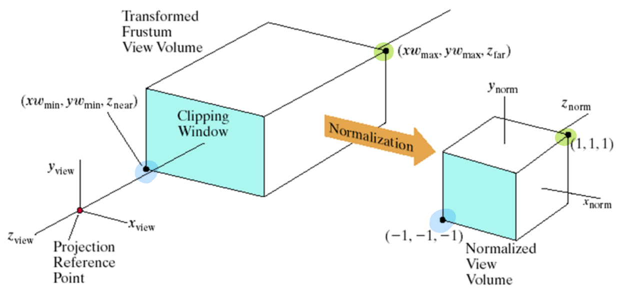
> 💡 할 일: 크기 **2x2×2**, 원점 **(0, 0, 0)**, z축 **left hand coordinate**
> - s_x, s_y, s_z, t_z 정의

- Parallelpiped → 정육면체



$$\mathbf{M}_{xy\,\text{scale}} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

  - Normalization 행렬 = **M_xyscale**
  - **M_xyscale** 행렬에는 x, y 축 scaling 만 존재



$(xw_{min}, yw_{min}, z_{near}) \to$ (-1,-1,-1)

$(xw_{max}, yw_{max}, z_{far}) \to$ (1,1,1)

$$\begin{bmatrix} x_h \\ y_h \\ z_h \\ h \end{bmatrix} = \mathbf{M}_{\text{normpers}} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$\mathbf{M}_{\text{normpers}} = \mathbf{M}_{xy\,\text{scale}} \cdot \mathbf{M}_{\text{obliquepers}}$$
$$= \begin{bmatrix} -z_{\text{near}}s_x & 0 & s_x\dfrac{xw_{\min}+xw_{\max}}{2} & 0 \\ 0 & -z_{\text{near}}s_y & s_y\dfrac{yw_{\min}+yw_{\max}}{2} & 0 \\ 0 & 0 & s_z & t_z \\ 0 & 0 & -1 & 0 \end{bmatrix} \begin{array}{cc} xw_{min} & x_h \\ yw_{min} & y_h \\ zw_{min} & z_h \\ 1 & h \end{array}$$

$$\mathbf{M}_{\text{obliquepers}} = \mathbf{M}_{\text{pers}} \cdot \mathbf{M}_{z\,\text{shear}}$$
$$= \begin{bmatrix} -z_{\text{near}} & 0 & \dfrac{xw_{\min}+xw_{\max}}{2} & 0 \\ 0 & -z_{\text{near}} & \dfrac{yw_{\min}+yw_{\max}}{2} & 0 \\ 0 & 0 & s_z & t_z \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

$$s_x = \frac{2}{xw_{\max}-xw_{\min}}, \qquad s_y = \frac{2}{yw_{\max}-yw_{\min}}$$
$$s_z = \frac{z_{\text{near}}+z_{\text{far}}}{z_{\text{near}}-z_{\text{far}}}, \qquad t_z = \frac{2\,z_{\text{near}}\,z_{\text{far}}}{z_{\text{near}}-z_{\text{far}}}$$

Final Perspective Transformation Matrix

- symmetric하지 않은 경우

$$\mathbf{M}_{\text{normpers}} = \begin{bmatrix} \dfrac{-2z_{\text{near}}}{xw_{\max}-xw_{\min}} & 0 & \dfrac{xw_{\max}+xw_{\min}}{xw_{\max}-xw_{\min}} & 0 \\ 0 & \dfrac{-2z_{\text{near}}}{yw_{\max}-yw_{\min}} & \dfrac{yw_{\max}+yw_{\min}}{yw_{\max}-yw_{\min}} & 0 \\ 0 & 0 & \dfrac{z_{\text{near}}+z_{\text{far}}}{z_{\text{near}}-z_{\text{far}}} & -\dfrac{2z_{\text{near}}z_{\text{far}}}{z_{\text{near}}-z_{\text{far}}} \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

- symmetric한 경우

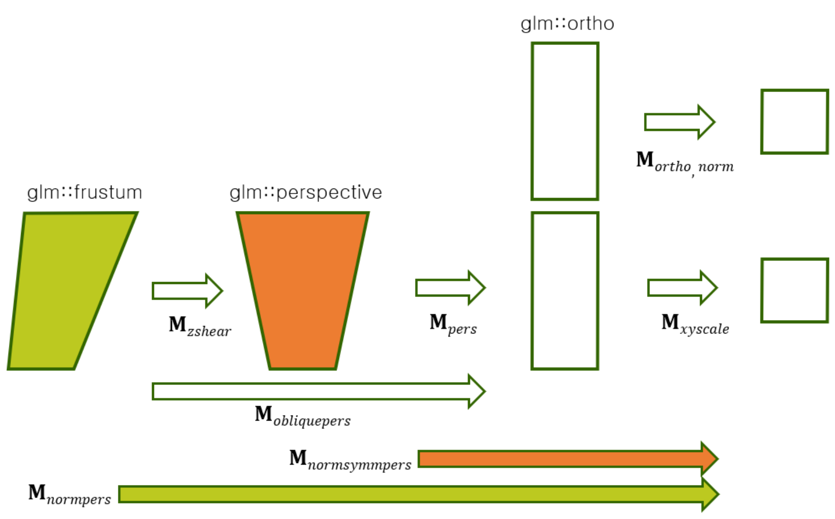
$$\mathbf{M}_{\text{normsymmpers}} = \begin{bmatrix} \dfrac{\cot\left(\frac{\theta}{2}\right)}{\text{aspect}} & 0 & 0 & 0 \\ 0 & \cot\left(\dfrac{\theta}{2}\right) & 0 & 0 \\ 0 & 0 & \dfrac{z_{\text{near}}+z_{\text{far}}}{z_{\text{near}}-z_{\text{far}}} & -\dfrac{2\,z_{\text{near}}\,z_{\text{far}}}{z_{\text{near}}-z_{\text{far}}} \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

$$-z_{near} = \frac{\text{height}}{2}\cot\frac{\theta}{2} = \frac{\text{width}}{2\text{aspect}}\cot\frac{\theta}{2}$$

$$\frac{xw_{min}+xw_{max}}{2} = 0, \quad \frac{yw_{min}+yw_{max}}{2} = 0$$

Summary

> 💡 frustum(symmetric x) → symmetric → parallel → normalized 정육면체



QUIZ) Orthographic Projection

The view volume for orthographic projection is defined by two corner points:

- $(xw_{min}, yw_{min}, z_{near})=(2, 2, -1)$
- $(xw_{max}, yw_{max}, z_{far})=(4, 4, -3)$

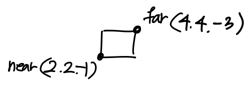
Find the corresponding orthographic projection matrix $\mathbf{M}_{ortho,norm}$.

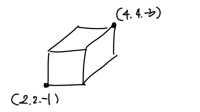
답이 맞습니다.

$$\mathbf{M}_{\text{ortho,norm}} = \begin{bmatrix} \frac{2}{xw_{\max}-xw_{\min}} & 0 & 0 & -\frac{xw_{\max}+xw_{\min}}{xw_{\max}-xw_{\min}} \\ 0 & \frac{2}{yw_{\max}-yw_{\min}} & 0 & -\frac{yw_{\max}+yw_{\min}}{yw_{\max}-yw_{\min}} \\ 0 & 0 & \frac{-2}{z_{\text{near}}-z_{\text{far}}} & \frac{z_{\text{near}}+z_{\text{far}}}{z_{\text{near}}-z_{\text{far}}} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

정답 : $\begin{bmatrix} 1 & 0 & 0 & -3 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & -1 & -2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

$$\mathbf{M}_{\text{ortho,norm}} = \begin{bmatrix} \frac{2}{xw_{\max}-xw_{\min}} & 0 & 0 & -\frac{xw_{\max}+xw_{\min}}{xw_{\max}-xw_{\min}} \\ 0 & \frac{2}{yw_{\max}-yw_{\min}} & 0 & -\frac{yw_{\max}+yw_{\min}}{yw_{\max}-yw_{\min}} \\ 0 & 0 & \frac{-2}{z_{\text{near}}-z_{\text{far}}} & \frac{z_{\text{near}}+z_{\text{far}}}{z_{\text{near}}-z_{\text{far}}} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$





$$\begin{bmatrix} \frac{2}{4-2} & 0 & 0 & -\frac{4+2}{4-2} \\ 0 & \frac{2}{4-2} & 0 & -\frac{4+2}{4-2} \\ 0 & 0 & \frac{-2}{-1-(-3)} & \frac{-1-3}{-1-(-3)} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -3 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & -1 & -2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Given a point $\mathbf{p}$=(3, 3, -2) defined in viewing coordinating system, use question #4 to find its corresponding point $\mathbf{p}$' in normalized device coordinate (i.e. $\mathbf{p}'=\mathbf{M}_{ortho,norm}\,\mathbf{p}$)

하나를 선택하세요.

- a. (3, 3, -2)
- b. (1, 0, 0)
- c. (0, 0, 0) ✔
- d. (1, 1, 1)

答
답이 맞습니다.

$$\begin{bmatrix} 1 & 0 & 0 & -3 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & -1 & -2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 3 \\ 3 \\ -2 \\ 1 \end{bmatrix}$$

정답 : (0, 0, 0)

$$\begin{bmatrix} 1 & 0 & 0 & -3 \\ 0 & 1 & 0 & -3 \\ 0 & 0 & -1 & -2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 3 \\ 3 \\ -2 \\ 1 \end{bmatrix} = \begin{bmatrix} 3-3 \\ 3-3 \\ -2+2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Find the projected point $\mathbf{p}$'' on the 2D view plane (near plane) of $\mathbf{p}$' from question #5.

하나를 선택하세요.

- a. (0, 1)
- b. (1, 1)
- c. (0, 0) ✔
- d. (1, 0)

답이 맞습니다.

Just take the x and y coordinates from (0, 0, 0), the result of question #5.

정답 : (0, 0)

z 날림

(0, 0, 0) → (0, 0)

QUIZ) Perspective Projection

With the following GLM function, what would be the corresponding projective transformation matrix $\mathbf{M}_{normsymmpers}$?

- glm::perspective($\frac{\pi}{2}$, 2, 1, 2)

답이 맞습니다.

$$\mathbf{M}_{\text{normsymmpers}} = \begin{bmatrix} \frac{\cot\left(\frac{\theta}{2}\right)}{\text{aspect}} & 0 & 0 & 0 \\ 0 & \cot\left(\frac{\theta}{2}\right) & 0 & 0 \\ 0 & 0 & \frac{z_{\text{near}}+z_{\text{far}}}{z_{\text{near}}-z_{\text{far}}} & -\frac{2z_{\text{near}}z_{\text{far}}}{z_{\text{near}}-z_{\text{far}}} \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

$\theta = \frac{\pi}{2}, aspect = 2, z_{near} = -1, z_{far} = -2$

정답 : $\begin{bmatrix} \frac{1}{2} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -3 & -4 \\ 0 & 0 & -1 & 0 \end{bmatrix}$

$$\mathbf{M}_{\text{normsymmpers}} = \begin{bmatrix} \frac{\cot\left(\frac{\theta}{2}\right)}{\text{aspect}} & 0 & 0 & 0 \\ 0 & \cot\left(\frac{\theta}{2}\right) & 0 & 0 \\ 0 & 0 & \frac{z_{\text{near}}+z_{\text{far}}}{z_{\text{near}}-z_{\text{far}}} & -\frac{2z_{\text{near}}z_{\text{far}}}{z_{\text{near}}-z_{\text{far}}} \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

symmetric. normalize

$\theta=\frac{\pi}{2}$, aspect $=\frac{w}{h}=2$, $|z_{near}|=1$. $|z_{far}|=2$

☆ ↳ $z_{near}=-1$, $z_{far}=-2$

$\cot=\frac{1}{\tan}$

$\tan(\frac{\pi}{4})=1$

$$\begin{bmatrix} \frac{1}{2} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{-1-2}{-1+2} & -\frac{2(-1)(-2)}{-1+2} \\ 0 & 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -3 & -4 \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

Given a point A=(0, 0, -2) in the viewing coordinate system, what would be the corresponding projected point A' in clip coordinate of the homogeneous form (before perspective division) according to the question #1?

하나를 선택하세요.

- a. (0, 0, 2, 2) ✔
- b. (0, 0, 0, 2)
- c. (0, 0, 1, 2)
- d. (0, 0, 0, 1)

$$\begin{bmatrix} \frac{1}{2} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -3 & -4 \\ 0 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ -2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 2 \\ 2 \end{bmatrix}$$

정답 : (0, 0, 2, 2)

open GL에서, h까지 포함한 걸 $\begin{bmatrix} x_h \\ y_h \\ z_h \\ h \end{bmatrix}$ clip cordinate라고 함

perspective division 이전이니까 z와 h로 나누기 전.

→ Mnormsymmpers 곱해주기

$$\begin{bmatrix} \frac{1}{2} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -3 & -4 \\ 0 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ -2 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 \\ 0 \\ 6-4 \\ 2 \end{bmatrix}$$

$$= \begin{bmatrix} 0 \\ 0 \\ 2 \\ 2 \end{bmatrix}$$

What is the corresponding normalized device coordinate (after perspective division) of question #2?

하나를 선택하세요.

- a. (0, 0, 1) ✔
- b. (0, 0, 0)
- c. $(0, 0, \frac{1}{2})$
- d. (0, 0, 2)

$$\begin{bmatrix} \frac{0}{2} \\ \frac{0}{2} \\ \frac{2}{2} \end{bmatrix}$$

정답 : (0, 0, 1)

h로 나눈거

normalized device coordinate $\begin{bmatrix} \frac{x_h}{h} \\ \frac{y_h}{h} \\ \frac{z_h}{h} \end{bmatrix}$

perspective division 후에ㅏ h로 나눈 후

$$\begin{bmatrix} \frac{0}{2} \\ \frac{0}{2} \\ \frac{2}{2} \\ \frac{2}{2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

h=1 이어야 함

What is the projected point in 2D of point A, from question #2, on the near plane?

하나를 선택하세요.

- a. (1, 1)
- b. (0, 1)
- c. (0, 0) ✔
- d. (1, 0)

Just take x and y from question #3.

정답 : (0, 0)

Projection할 준비 완료

z날림

(0, 0, 1) → (0, 0)