2023-2 Computer Graphics PI 3차시

Viewing Transformation
&
Projection Transformation

AM11:00에 시작됩니다.

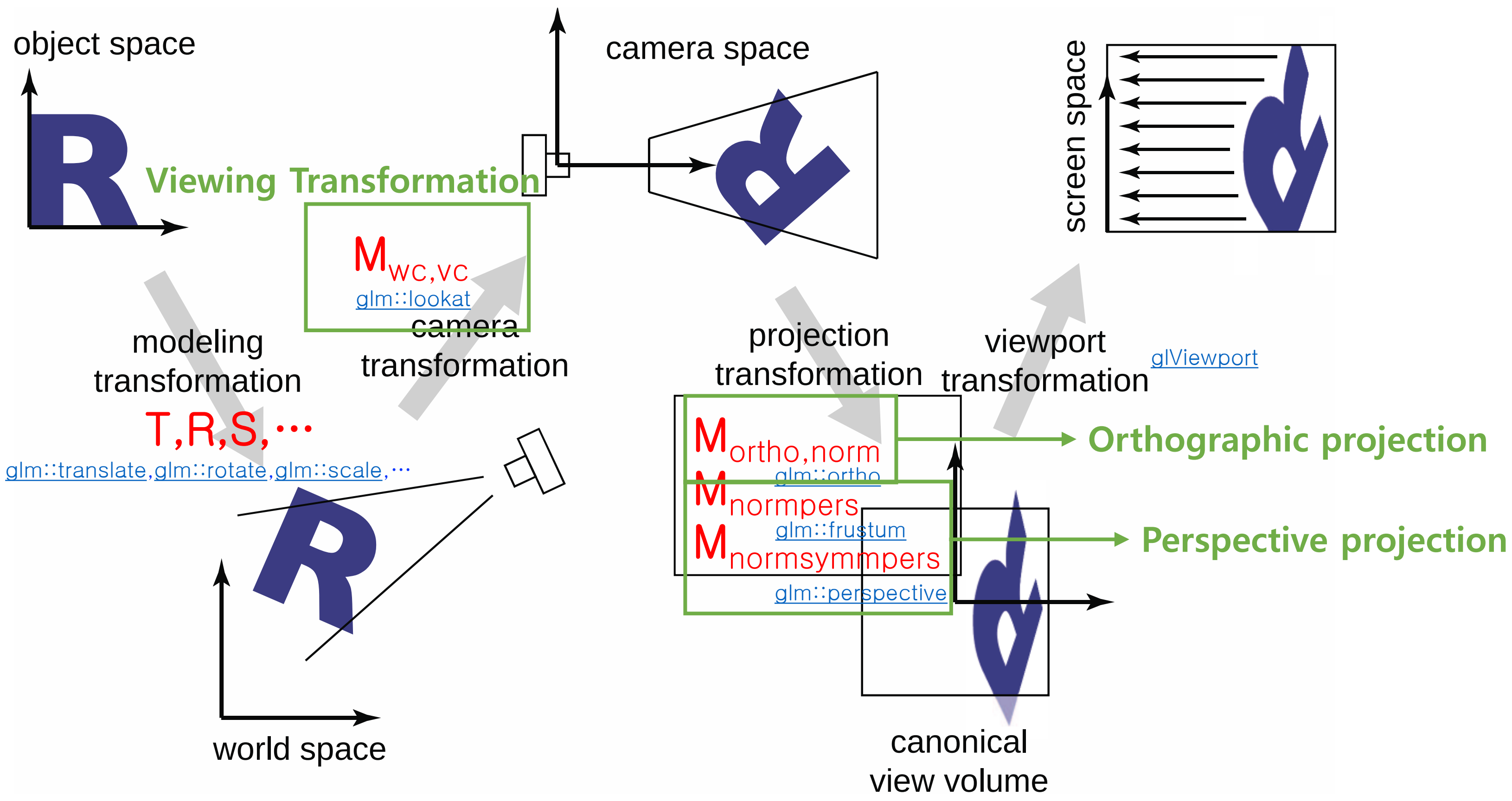
입장 후 채팅창에 학번/이름 작성 부탁드립니다.

Graphics Pipeline

- 해당 coordinate system 에 맞게 각 transformation으로 좌표체계 바꿔주는 것
- 각 transformation 은 4x4 행렬 (나중에 배움)
- 아래 표가 rendering pipeline 전부는 아님. Shading 등의 작업 남음.

Coordinate System	Transformation	한마디 정리	세부 설명
modeling coordinate system			각 모델마다 다름. 여러 개 존재.
world coordinate system	modeling transformation	모델 위치 결정	하나의 scene안에 있는 모든 모델이 하나의 wcs에 위치함.
viewing coordinate system	viewing transformation	카메라 위치 결정	카메라 위치가 vcs의 원점 & xyz축 결정
normalized coordinate system	projection transformation	3D → 2D 준비	1) Viewing volume 결정 2) Projection type 결정
screen coordinate system	viewport transformation	3D → 2D	1) Z 좌표값 날리기 2) Display screen 사이즈에 맞게 scaling

Graphics Pipeline



오늘 배우는 부분

2023-2 Computer Graphics PI 3차시

- Viewing Transformation $\mathbf{M}_{WC,VC}$

- Projection Transformation
  - Orthographic Projection $\mathbf{M}_{ortho,norm}$

  - Perspective Projection $\mathbf{M}_{normpers,}$ $\mathbf{M}_{normsymmpers}$

Viewing Transformation

Viewing Transformation – Viewing Parameter

- **viewing parameter**

1) 카메라 위치 $\mathbf{P_0 = (x_0 , y_0 , z_0)}$

2) View plane 에 수직인 vector 인 view plane normal $\boldsymbol{VPN} = \mathbf{n} = \mathbf{z}_{view}$
   카메라 방향의 z축인 $\mathbf{z}_{view}$ 을 $\mathbf{n}$ 와 동일하게 setting 한 것

3) View up $\mathbf{v} = \mathbf{y_{view}}$

4) $\mathbf{u}$ 는 $\mathbf{v}$ 와 $\mathbf{n}$ 으로 구한다. $\mathbf{u} = \mathbf{v} \times \mathbf{n},\ \mathbf{u} = \mathbf{x_{view}}$

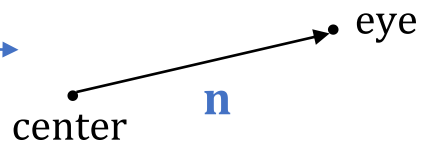
1) 2) 3) 은 사용자가 정하는 parameter. 4)는 2) 3)으로 계산하는 것.

- **glm::lookAt function**

```
In OpenGL function,
glm::lookAt(glm::vec3(eye.x, eye.y, eye.z),
            glm::vec3(center.x, center.y, center.z),
            glm::vec3(up.x, up.y, up.z))
```

$$\mathbf{n}=\frac{\mathbf{eye-center}}{\|\mathbf{eye-center}\|},\quad \mathbf{v}=\frac{\mathbf{up}}{\|\mathbf{up}\|}$$

$\mathbf{P_0 = (x_0 , y_0 , z_0)}$



$\mathbf{v} = \mathbf{y_{view}}$

Viewing Transformation – $M_{WC,VC}$

- (이해)

STEP1. $\mathbf{P_0}$ 이 원점 (0, 0, 0) 이 되게 하자 ⇒ Translation

STEP2. $\boldsymbol{x_{view}} = \boldsymbol{u}$, $\boldsymbol{y_{view}} = \boldsymbol{v}$, $\boldsymbol{z_{view}} = \boldsymbol{n}$ 이 각각 x, y, z 축이 되게 하자 ⇒ Rotation

$\boldsymbol{u}$ 를 $\boldsymbol{R}$ 하면 $\boldsymbol{x}$ = *(1, 0, 0)* 이 되어야 한다 ⇒ $\boldsymbol{Ru} = \boldsymbol{x}$

$\boldsymbol{v}$ 를 $\boldsymbol{R}$ 하면 $\boldsymbol{y}$ = *(0, 1, 0)* 이 되어야 한다 ⇒ $\boldsymbol{Rv} = \boldsymbol{y}$

$\boldsymbol{n}$ 를 $\boldsymbol{R}$ 하면 $\boldsymbol{z}$ = *(0, 0, 1)* 이 되어야 한다 ⇒ $\boldsymbol{Rn} = \boldsymbol{z}$

이걸 한번에 행렬로 써보면,

$$\mathbf{R}\begin{bmatrix} u_x & v_x & n_x \\ u_y & v_y & n_y \\ u_z & v_z & n_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{RU} = \mathbf{I}, \therefore \mathbf{R} = \mathbf{U}^{-1} = \mathbf{U}^{T}$$

cf) rotation 행렬과 reflection 행렬은 orthogonal matrix (직교행렬) 이라 $R^{-1} = R^{T}$

- (결과)

$$\mathbf{M}_{WC,\,VC} = \mathbf{R} \cdot \mathbf{T}$$

$$= \begin{bmatrix} u_x & u_y & u_z & -\mathbf{u} \cdot \mathbf{P}_0 \\ v_x & v_y & v_z & -\mathbf{v} \cdot \mathbf{P}_0 \\ n_x & n_y & n_z & -\mathbf{n} \cdot \mathbf{P}_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

암기!

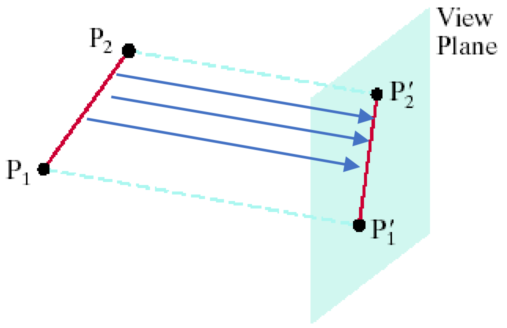
참고) 전체 graphics pipeline 에서의 이해

$$\underbrace{\boxed{(\text{아직 안 배움})}}_{\text{Projection Transformation}} \ \underbrace{\boxed{\mathbf{M}_{\mathrm{WC,VC}}}}_{\text{Viewing Transformation}} \cdot \underbrace{\boxed{\boldsymbol{T} \cdot \boldsymbol{R} \cdot \boldsymbol{T}}}_{\text{Modeling Transformation 의 예시}} \cdot \underbrace{\boxed{\begin{matrix} x \\ y \\ z \\ 1 \end{matrix}}}_{\text{Input Coordinate}}$$

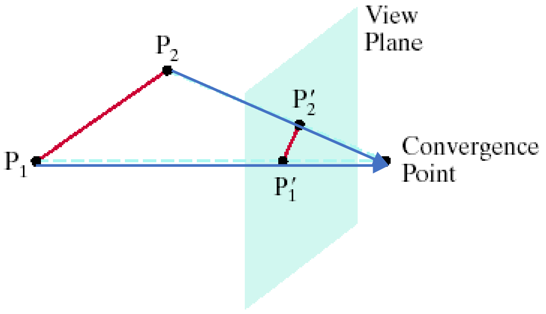
Projection Transformation

Projection Transformation 종류

- **(1) Parallel Projection** → oblique 는 잘 안 쓴다.
  수업에선 parallel = orthographic 으로 다룬다.
  - 평행선 보존됨
  - **(1-a) orthographic projection** : projection 되는 방향과 VPN 벡터 평행
  - **(1-b) oblique projection** : projection 되는 방향과 VPN 벡터 평행 X



- **(2) Perspective Projection**
  - 평행선 보존 안됨
  - 소실점 향해서 projection

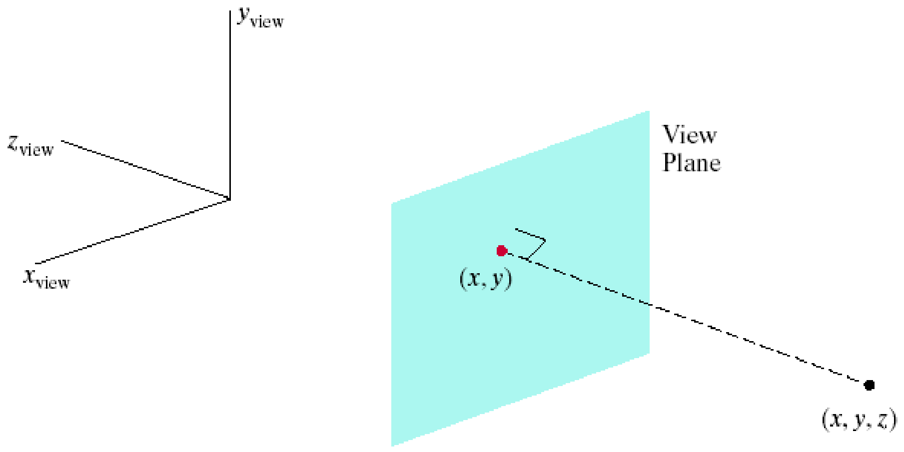


Projection Transformation – (1-a) Orthographic Projection

- view plane 의 수직 벡터인 $\boldsymbol{VPN} = \mathbf{n} = \mathbf{z}_{\boldsymbol{view}}$ 과
  projection direction 이 평행

- 이런 조건에서, VCS 상의 점 $(x, \mathrm{y}, \mathrm{z})$ 이
  View plane 에 projection 되었을 때 점 $(\mathrm{x}_{\mathrm{p}}, y_{\mathrm{p}})$ 의 좌표가 궁금한 것

- (결론) orthographic projection 에서는
  $\boldsymbol{VPN} = \mathbf{n} = \mathbf{z}_{\boldsymbol{view}}$ 이므로
  $x_p = x,\ y_p = y$

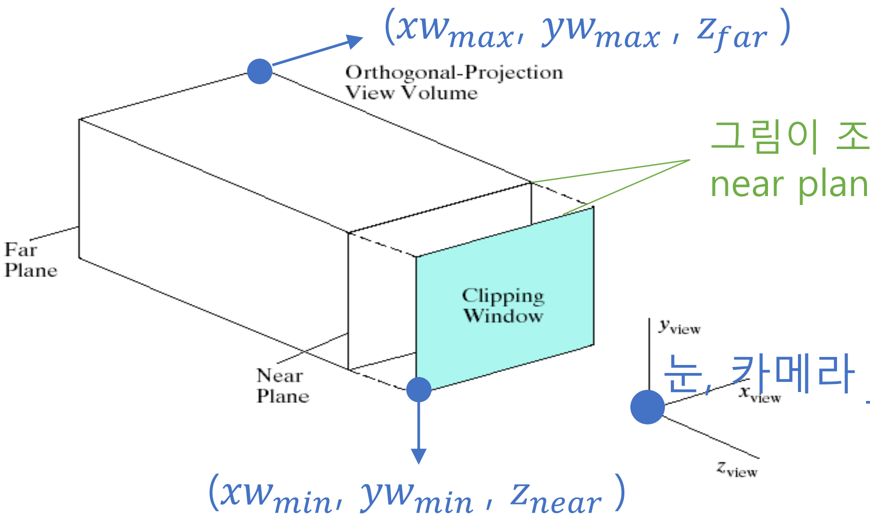


Projection Transformation – (1-a) Orthographic Projection

- projection transformation 할 때, view volume 도 정하고 normalize 해야 한다.
  cf) 뒤에서 배우지만, vv 밖은 잘라주는 clipping 도 해야 한다.

- Orthogonal projection 에서는
  vv모양이 rectangular parallelpipe 이다.

- **STEP1. vv 정의** : vv = rectangular parallelpipe 를 두 점의 좌표로 정의



그림이 조금 이상함.
near plane = view plane

쳐다보는 방향이 –z 방향
z_{near}, z_{far} 는 둘 다 음수

Projection Transformation – (1-a) Orthographic Projection

- **STEP2. vv normalization**

cf) 왜 normalize 하냐?

이유1) clipping algorithm 의 최적화

이유2) 최종적으로는 near plane 이 target display 사이즈에 맞도록 scaling 할 것이다 (viewport transformation)

이때 이미 normalize 되어 있으면 scaling 이 편할 것

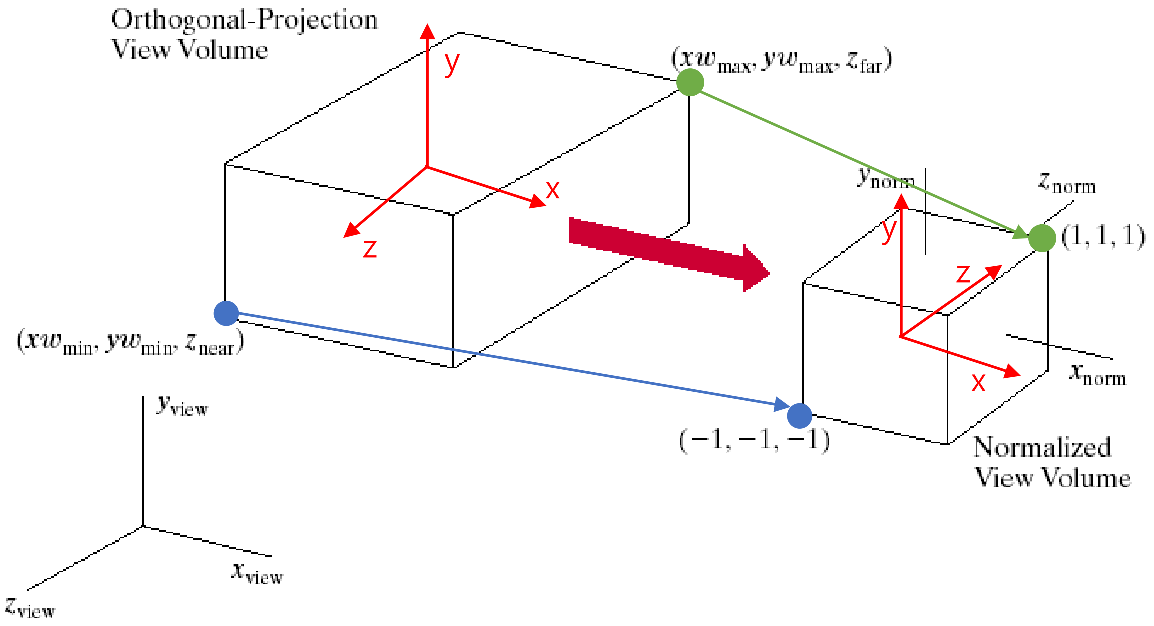
- (이해)

(1) vv의 중앙을 원점으로 ⇒ translation

(2) vv의 크기를 2x2x2 로 ⇒ scaling

(1) (2) 를 만족 시키려면,

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



Projection Transformation – (1-a) Orthographic Projection

- **STEP2. vv normalization (cont.)**

(결론)

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

(ㄹ) translation

scaling

(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

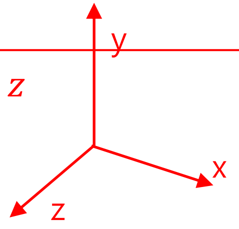
(ㄷ) : <u>z_{near}, z_{far} 는 음수</u> ⇒ <u>$z_{near} > z_{far}$</u> ⇒ <u>$z_{near} - z_{far}$</u> 는 양수

Left hand coordinate (LHC) 로 바꿔주려면 z flip 필요

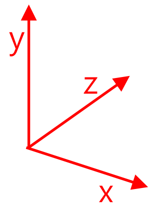
z_{near} - z_{far} 를 길이 2 로 바꿔주고, - 붙여준다.

(ㄹ) : 앞의 두 식을 풀어서 계산한 것

viewing coordinate 까지는 right hand coordinate. x x y = z



projection 하고서 normalize 할 땐 left hand coordinate. x x 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.

Projection Transformation – (1-a) Orthographic Projection

- glm::ortho

```
template<typename T >
        GLM_FUNC_DECL detail::tmat4x4
                < T, defaultp >  ortho (T const &left, T const &right, T const &bottom, T const &top, T const &zNear, T const &zFar)
```

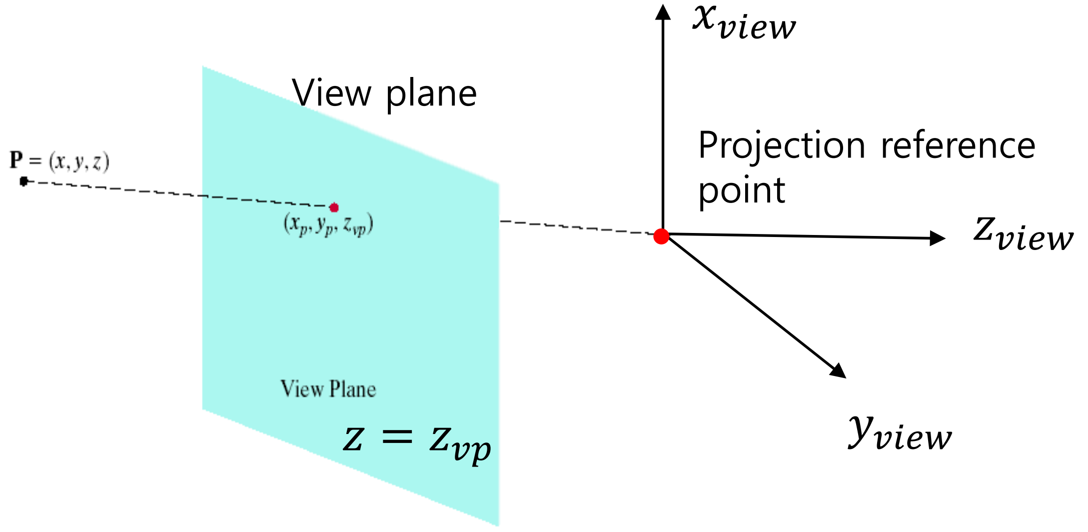
xw_{max} yw_{max}

xw_{min} yw_{min}

opengl 에서는
z_{near}, z_{far} 는 절댓값 써서
항상 양수값만 들어간다.
(xw_{max}, yw_{max} , z_{far})

Projection Transformation – (2) Perspective Projection

- **Perspective projection**
  - 눈으로부터 얼마나 떨어져 있냐에 따라 물체의 크기 다르게 보이는 원근감을 가지는 projection
  - 공간 상의 모든 물체가 원점(소실점)을 향해 projection
- projection reference point = 눈의 위치 = (0,0,0)
- view plane = projection할 plane
- P(x,y,z)가 (0,0,0)를 향해 평면 상으로 projection될 때 점(x',y',z')을 구해보자
  1. 매개변수 u를 이용하여 표현 (0<=u<=1)
  2. $z' = z_{VP} = (1-u)z$ 이므로 u값 구하기
  3. u 대입해서 x',y' 구하기



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

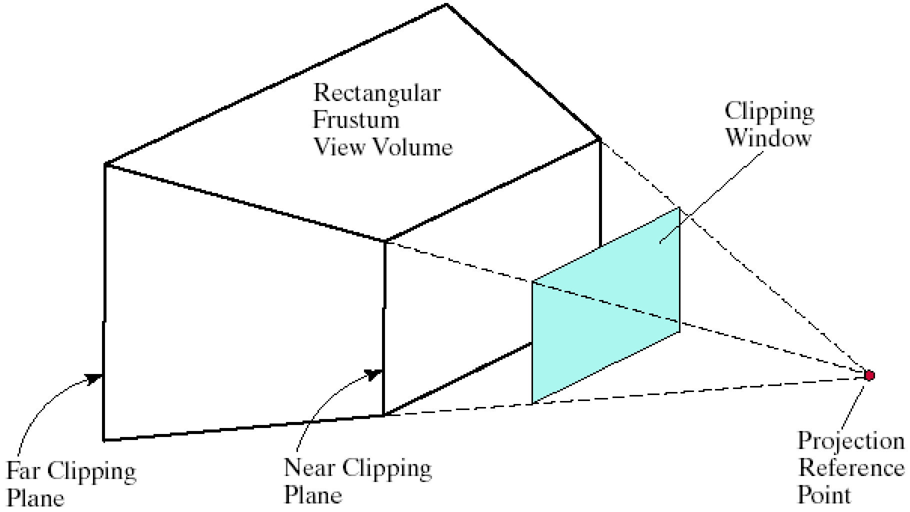
2 $$u = 1 - \frac{z_{vp}}{z}$$

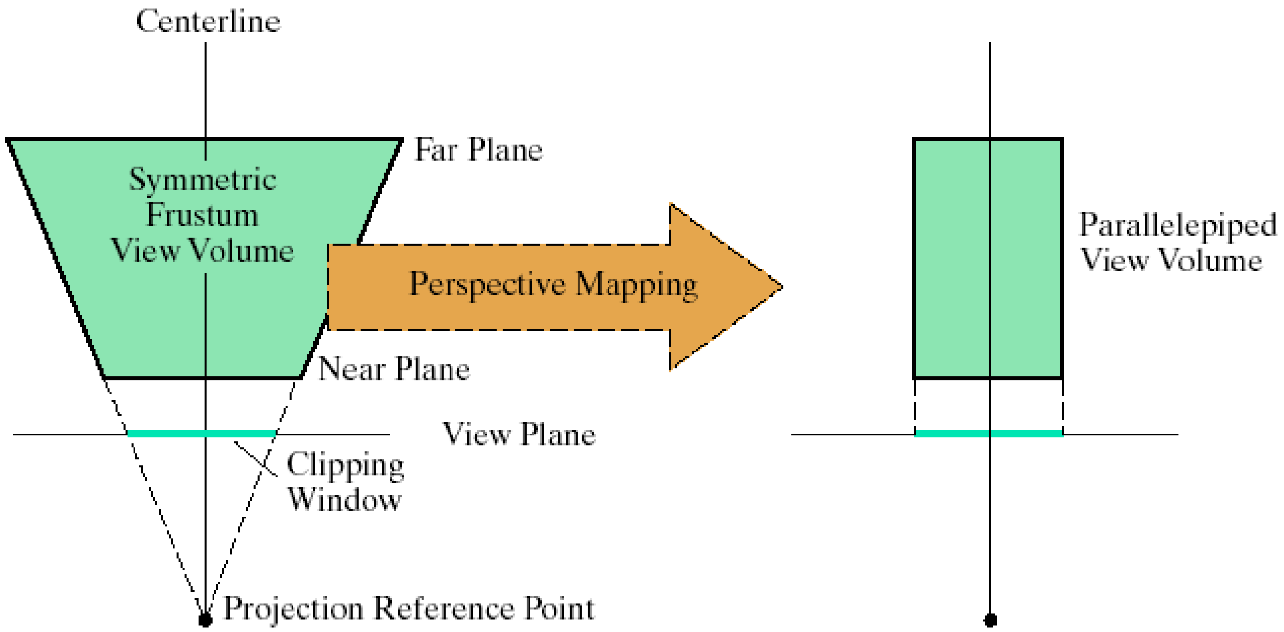
3 $$x_p = \frac{z_{vp}}{z}x \qquad y_p = \frac{z_{vp}}{z}y$$

- 식의 의미
  - 원점으로부터 거리가 멀수록(z가 클수록) x_p, y_p값이 줄어들고 거리가 가까우면 x_p, y_p 값이 커짐
  - -> perspectivity

Projection Transformation – (2) Perspective Projection

- **View frustum**
  - View volume의 모양이 피라미드 형태
  - 필요한 파라미터 : z_{near}, z_{far}, FOV
- 공간 상 물체들은 near plane으로 projection
  - near plane = view plane
- $z = z_{vp} = z_{near} < 0$
  - right hand coordinat을 사용하므로 z가 증가하는 방향(+)이 쳐다보는 방향(-)의 반대
- View volume을 normalize 하며 정육면체 형태로 바꾸어야 함

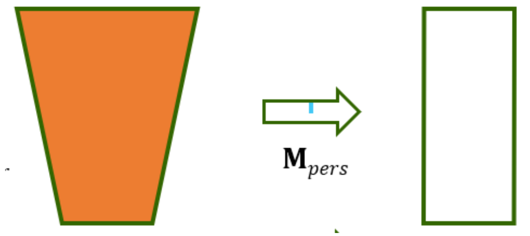




Projection Transformation – (2) Perspective Projection

- Fruastum -> parallelpiped -> 정육면체 순서로 fraustum shape을 결국 정육면체 형태로 바꿔야 함
- Parallel -> 정육면체
  - Orthographic projection에서 다뤄 이미 알고 있음
- Fraustum -> parallel **(perspective mapping)**
  - Perspecive projection에서 view plane에 projection 한 점을 구하는 방법 이용

glm::perspective



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

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

$$\xrightarrow{\text{행렬 형태} \quad \neq}$$

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

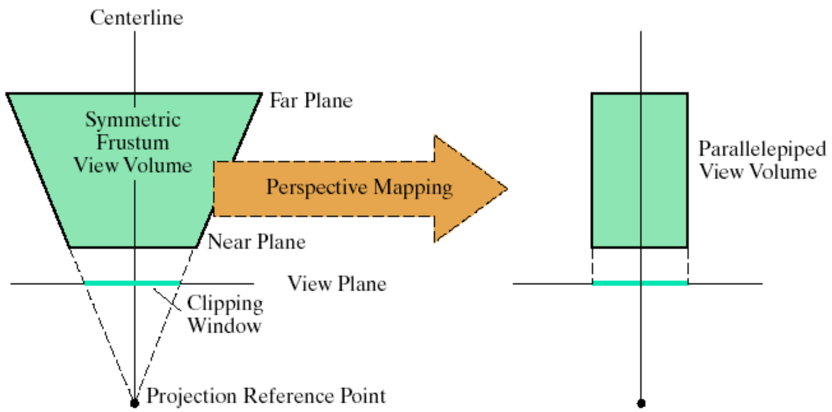
h≠1

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

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

$$h = -z$$

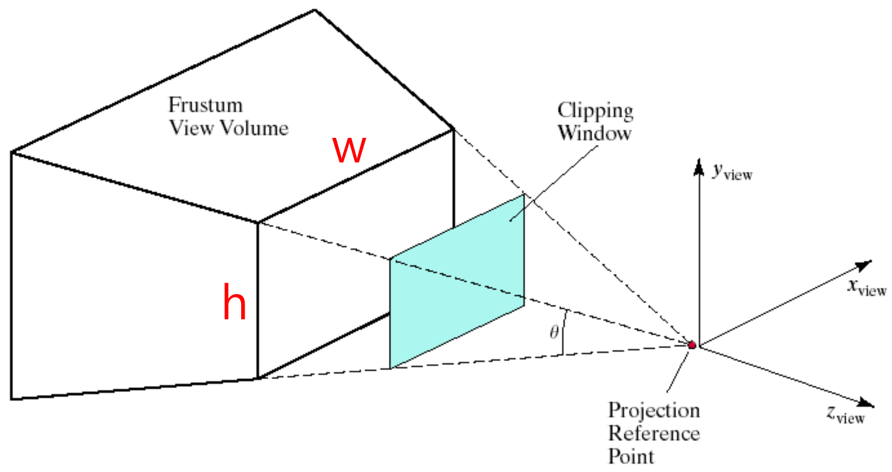
- $\mathrm{M_{pers}}$는 projection 되는 점을 구하는 행렬
  - s_z는 z축 scaling, t_z는 z축 translation에 사용
  - Affine transformation이 아님
- Perspective division: 원근감을 보정해주는 h로 (x_h, y_h, z_h)를 나누는 것

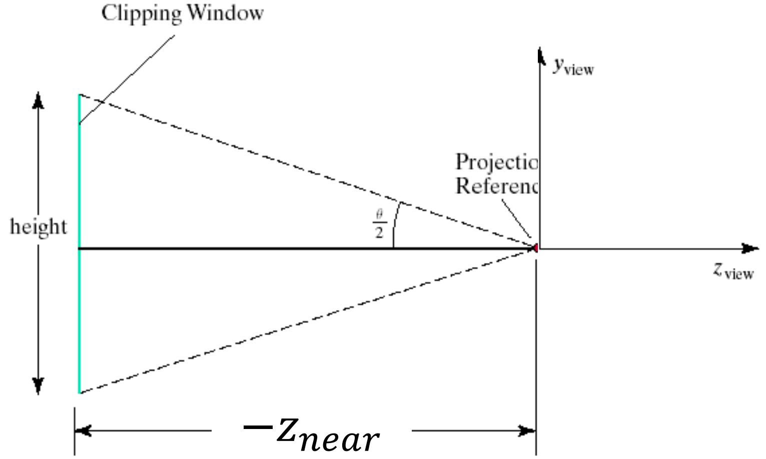


Projection Transformation – (2) Perspective Projection

(1) View Volume - symmetric case

- View volume이 x,y축 방향으로 symmetric한 형태
- glm::perspective(fovy, aspect, |znear|, |zfar|), aspect=width/height
  - fovy: view volume의 y방향 각도
  - aspect: 너비와 높이의 비율
  - znear: near plane까지의 거리
  - zfar: far plane까지의 거리





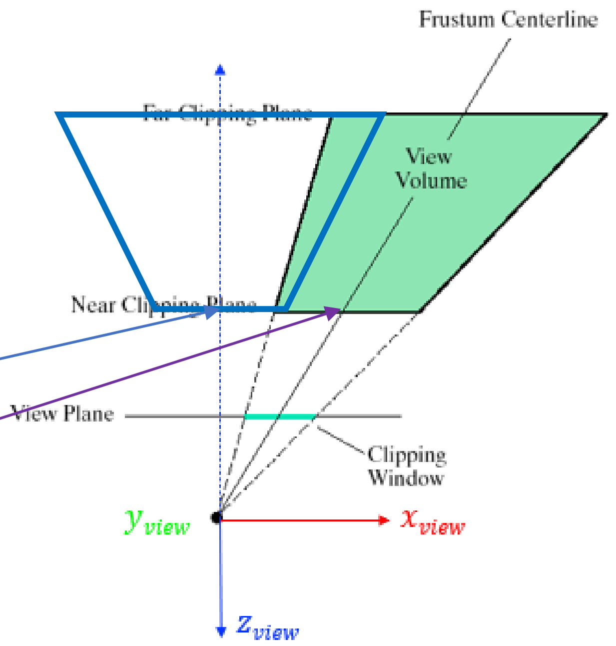
$$\tan\left(\frac{\theta}{2}\right) = \frac{\text{height}/2}{-z_{near}} \qquad \text{height} = -2z_{near}\tan\left(\frac{\theta}{2}\right)$$

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

Projection Transformation – (2) Perspective Projection

(1) View Volume - oblique case

- View volume이 x,y축 방향으로 symmetric하지 않은 형태
- glm::frustum(xwmin, xwmax, ywmin, ywmax, |znear|, |zfar|)
  - xwmin, xwmax: x방향 최소, 최댓값
  - ywmin, ywmax: y방향 최소, 최댓값
  - znear, zfar는 동일
- frustum -> summertric
  - shearing transformation(왜곡) 사용
  - 이 frustum은 x,y 방향으로만 왜곡되어 있음

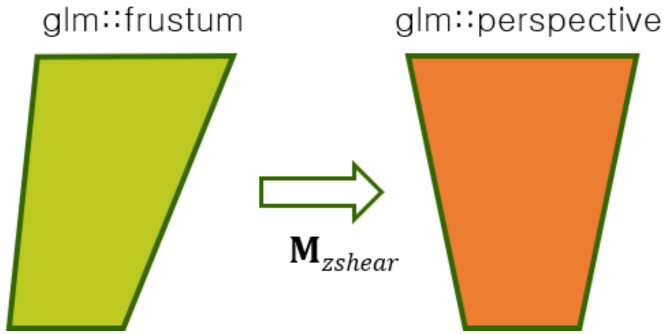


$$\mathbf{M}_{z\,\text{shear}} = \begin{bmatrix} 1 & 0 & \text{sh}_{zx} & 0 \\ 0 & 1 & \text{sh}_{zy} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

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

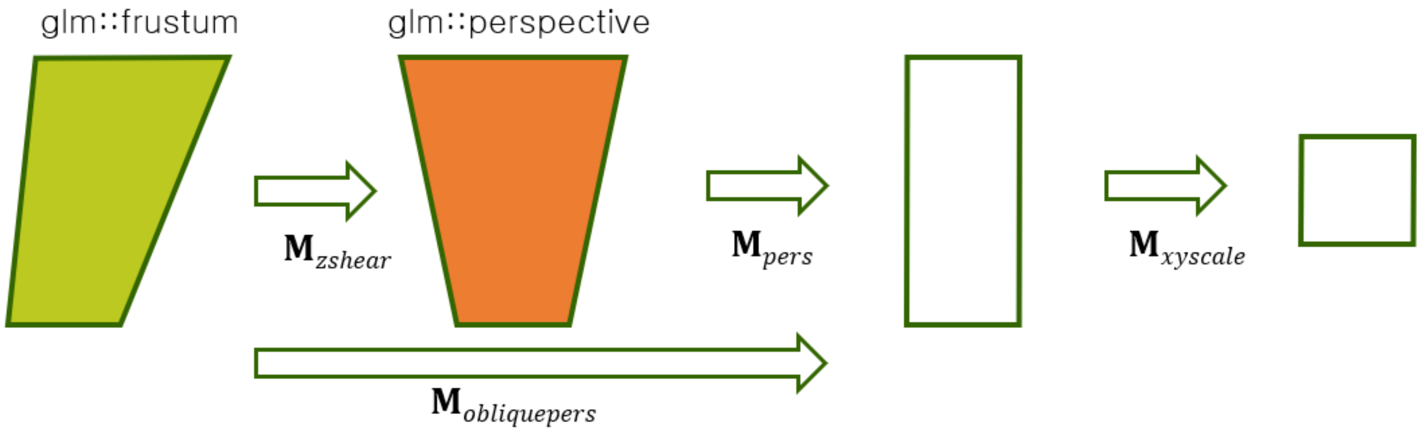
near plane 한 가운데 있는 점

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



Projection Transformation – (2) Perspective Projection

(1) View Volume - oblique case



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

$$\mathbf{M}_{\text{pers}} = \begin{bmatrix} -z_{\text{near}} & 0 & 0 & 0 \\ 0 & -z_{\text{near}} & 0 & 0 \\ 0 & 0 & s_z & t_z \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

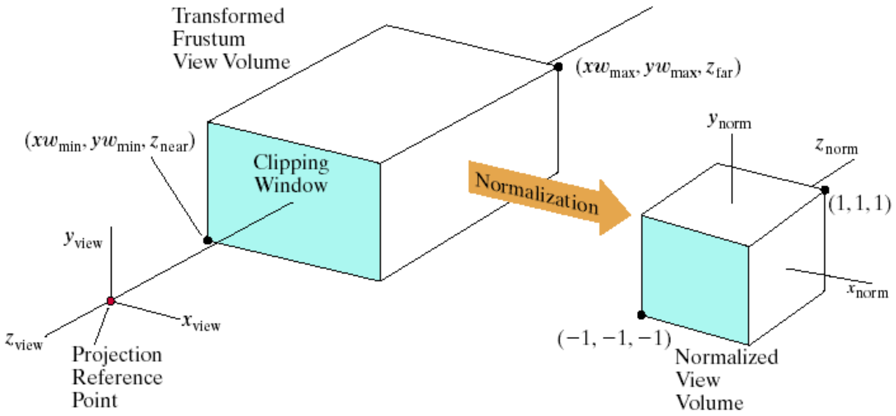
$$\mathbf{M}_{z\,\text{shear}} = \begin{bmatrix} 1 & 0 & \text{sh}_{zx} & 0 \\ 0 & 1 & \text{sh}_{zy} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Projection Transformation – (2) Perspective Projection

Normalization

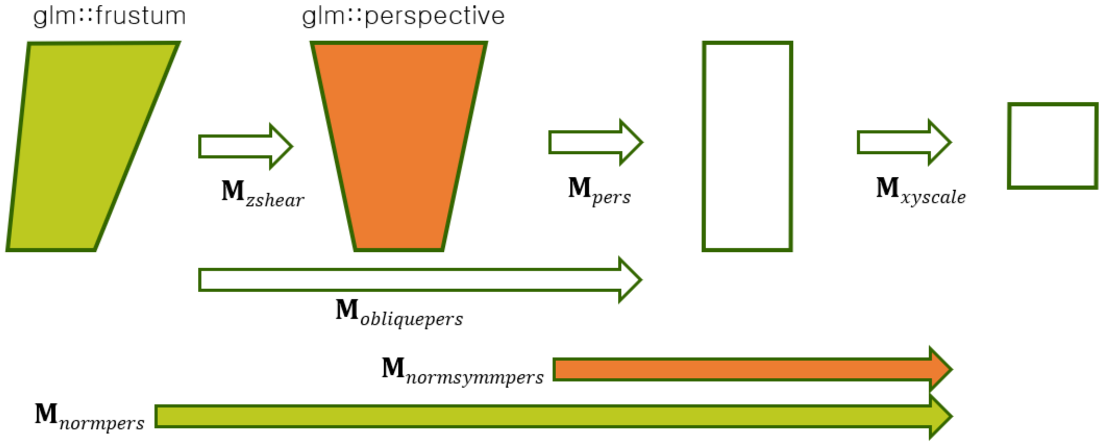
- parallel -> 정육면체
  - normalization 행렬 = $\mathrm{M_{xyscale}}$
  - $\mathrm{M_{xyscale}}$ 행렬에는 x,y축 scaling만 존재

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



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

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



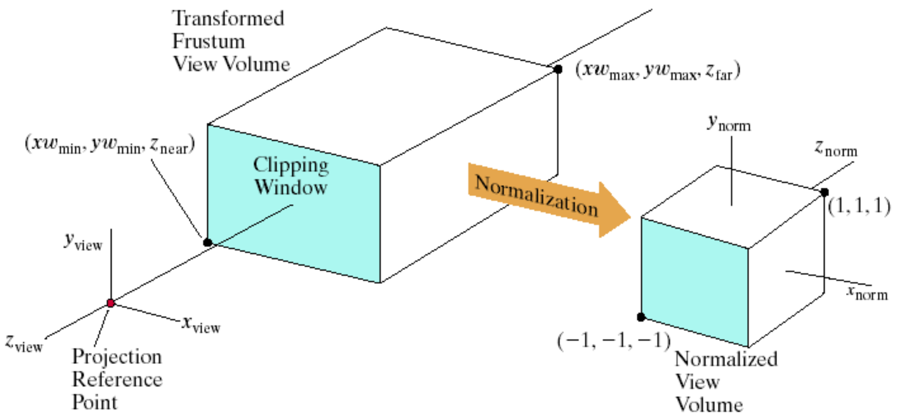
$$\mathbf{M}_{\mathrm{normpers}} = \mathbf{M}_{xy\,\mathrm{scale}} \cdot \mathbf{M}_{\mathrm{obliquepers}}$$

$$= \begin{bmatrix} -z_{\mathrm{near}}s_x & 0 & s_x\dfrac{xw_{\min}+xw_{\max}}{2} & 0 \\ 0 & -z_{\mathrm{near}}s_y & s_y\dfrac{yw_{\min}+yw_{\max}}{2} & 0 \\ 0 & 0 & s_z & t_z \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

Projection Transformation – (2) Perspective Projection

Normalization

- parallel -> 정육면체



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

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

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

$$\mathbf{M}_{\text{normpers}} = \mathbf{M}_{xy\,\text{scale}} \cdot \mathbf{M}_{\text{obliquepers}}$$

$$= \begin{bmatrix} -z_{\text{near}} s_x & 0 & s_x \dfrac{xw_{\min} + xw_{\max}}{2} & 0 \\ 0 & -z_{\text{near}} s_y & s_y \dfrac{yw_{\min} + yw_{\max}}{2} & 0 \\ 0 & 0 & s_z & t_z \\ 0 & 0 & -1 & 0 \end{bmatrix} \quad \begin{matrix} xw_{min} \\ yw_{min} \\ zw_{min} \\ 1 \end{matrix} \qquad \begin{matrix} x_h \\ y_h \\ z_h \\ h \end{matrix}$$

$x_h/h=-1,\ y_h/h=-1,\ z_h/h=-1$

$$s_x = \frac{2}{xw_{\max} - xw_{\min}}, \qquad s_y = \frac{2}{yw_{\max} - yw_{\min}}$$

$$s_z = \frac{z_{\text{near}} + z_{\text{far}}}{z_{\text{near}} - z_{\text{far}}}, \qquad t_z = \frac{2\, z_{\text{near}}\, z_{\text{far}}}{z_{\text{near}} - z_{\text{far}}}$$

Projection Transformation – (2) Perspective Projection

Final Perspective Transformation Matrix

- symmetric하지 않은 경우

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

- symmetric한 경우

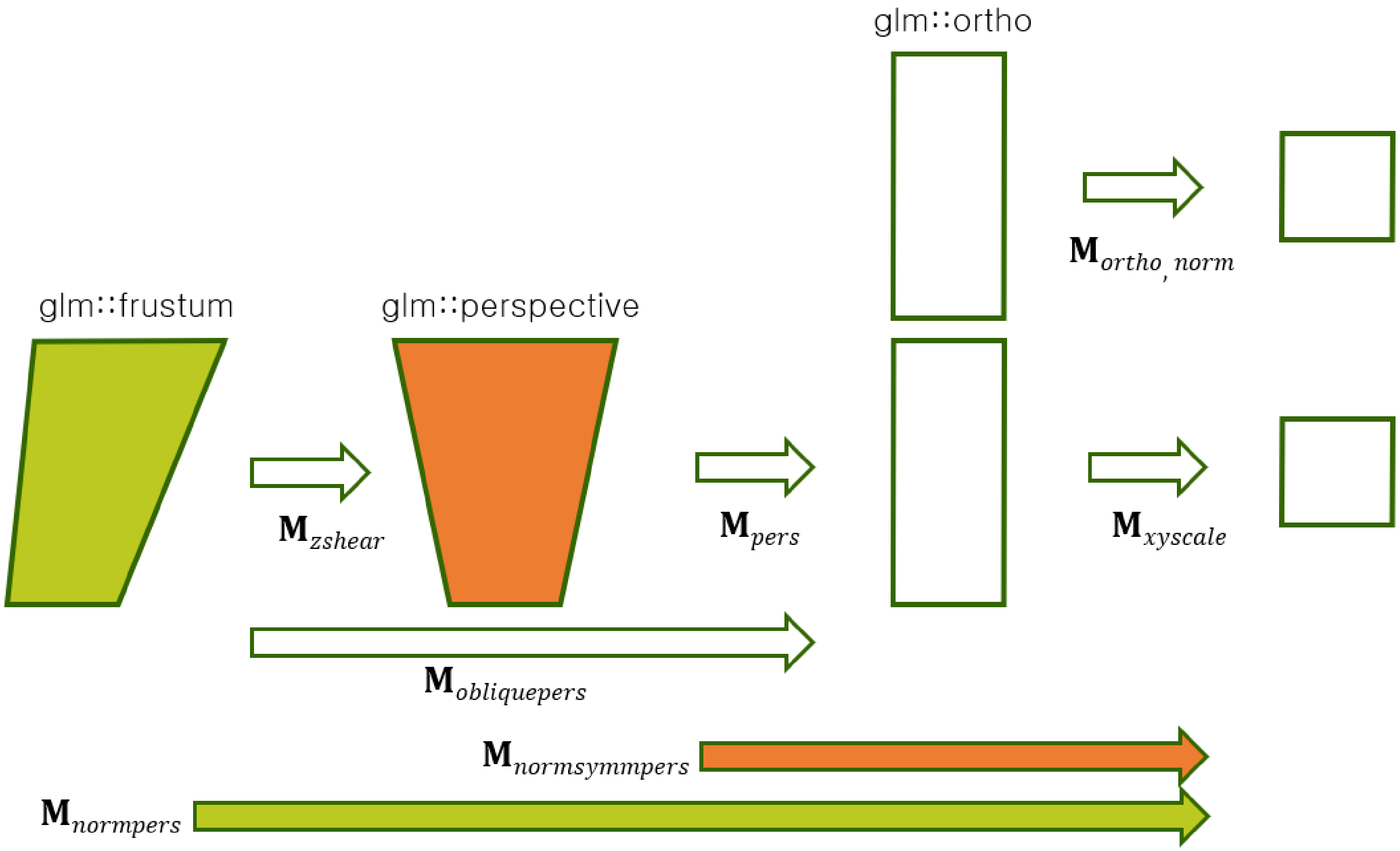
$$\mathbf{M}_{\text{normsymmpers}} = \begin{bmatrix} \frac{\cot(\frac{\theta}{2})}{\text{aspect}} & 0 & 0 & 0 \\ 0 & \cot\left(\frac{\theta}{2}\right) & 0 & 0 \\ 0 & 0 & \frac{z_{\text{near}} + z_{\text{far}}}{z_{\text{near}} - z_{\text{far}}} & -\frac{2z_{\text{near}}\, z_{\text{far}}}{z_{\text{near}}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

$$-z_{near} = \frac{\text{height}}{2}\cot\frac{\theta}{2} = \frac{\text{width}}{2\text{aspect}}\cot\frac{\theta}{2} \qquad \frac{xw_{min}+xw_{max}}{2} = 0, \frac{yw_{min}+yw_{max}}{2} = 0$$

Projection Transformation – (2) Perspective Projection

Summary

- frustum(symmetric x) -> symmetric -> parallel -> normalized 정육면체



QUIZ 1 – Viewing Transformation

The viewing parameters are given as below:

- View-up vector $\mathbf{v} = (0, -1, 0)$
- View plane normal (VPN) $\mathbf{n} = (0, 0, -1)$
- Viewing coordinate origin $\mathbf{o} = (-1, -2, -3)$

Find $\mathbf{u}$ to complete the **uvn** viewig coordinate system.

하나를 선택하세요.

a. (0, 1, 0)

b. (-1, 0, 0)

c. (0, 0, 1)

d. (1, 0, 0) ✔

$\mathbf{u} = \mathbf{v} \times \mathbf{n} = (1, 0, 0)$

정답 : (1, 0, 0)

Find the 4X4 homogenous matrix to perform viewing transformation, given as #1 question.

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

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

QUIZ 1 – Viewing Transformation

Point A(4, 5, 6) is defined in the world coordinate system. How is A viewed from the viewing coordinate system as defined in question #2? In other words, apply the viewing transformation to A.

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & -1 & 0 & -2 \\ 0 & 0 & -1 & -3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 \\ 5 \\ 6 \\ 1 \end{bmatrix}$$

정답 : (5, -7, -9)

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

QUIZ 2 – Orthographic Projection

Given a point **p**=(3, 3, -2) defined in viewing coordinating system, use question #1 to find its corresponding point **p'** in normalized device coordinate (i.e. **p'**=$M_{ortho,norm}$ **p**)

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

정답 : (0, 0, 0)

Find the projected point **p**" on 2D view plane (near plane) of **p**' from question #2.

Just take the x and y coordinate from (0, 0, 0), the result of question #2.

정답 : (0, 0)

QUIZ 3 – Perspective Projection

문제 1
정답
총 1.00 점에서

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

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

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

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

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

QUIZ 3 – Perspective Projection

문제 2
정답
총 1.00 점에서
1.00 점 할당
문제 표시

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

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

정답 : (0, 0, 2, 2)

QUIZ 3 – Perspective Projection

문제 3
정답
총 1.00 점에서 1.00 점 할당
문제 표시

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

하나를 선택하세요.

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

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

정답 : (0, 0, 1)

문제 4
정답
총 1.00 점에서 1.00 점 할당
문제 표시

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

하나를 선택하세요.

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

Just take x and y from question #3.

정답 : (0, 0)