BASICS

1 VECTOR AND MATRIX

1.1 VECTOR

1.1.1 DOT

$$\vec{a} = (x_1, y_1, z_1)$$

$$\vec{b} = (x_2, y_2, z_2)$$

$$\vec{a} \cdot \vec{b} = |\vec{a}||\vec{b}| \cos\theta = x_1 x_2 + y_1 y_2 + z_1 z_2$$

1.1.2 CROSS

$$\vec{a} \times \vec{b} = \left(|\vec{a}||\vec{b}| \sin\theta\right)\vec{n}$$

$$= (y_1 z_2 - z_1 y_2, z_1 x_2 - x_1 z_2, x_1 y_2 - y_1 x_2)$$

1.1.3 INTERPOLATION

$$Lerp\left(\vec{a}, \vec{b}, t\right) = (1 - t)\vec{a} + t\vec{b}$$

$$NLerp\left(\vec{a}, \vec{b}, t\right) = \mathrm{normalize}\left(Lerp\left(\vec{a}, \vec{b}, t\right)\right)$$

$$SLerp\left(\vec{a}, \vec{b}, t\right) = \frac{\sin\left((1 - t)\theta\right)}{\sin\theta}\vec{a} + \frac{\sin t\theta}{\sin\theta}\vec{b}$$

1.2 MATRIX

$$M = \begin{bmatrix} c_{00} & c_{10} & c_{20} & c_{30} \\ c_{01} & c_{11} & c_{21} & c_{31} \\ c_{02} & c_{12} & c_{22} & c_{32} \\ c_{03} & c_{13} & c_{23} & c_{33} \end{bmatrix}$$

1.2.1 MULTIPLY WITH VECTOR

$\overrightarrow{v_0}.w = 0$ as direction, $\overrightarrow{v_0}.w = 1$ as point.

$$\vec{v} = M\overrightarrow{v_0} \rightarrow \begin{bmatrix} \vec{v}.x \\ \vec{v}.y \\ \vec{v}.z \\ \vec{v}.w \end{bmatrix} = \begin{bmatrix} c_{00} & c_{10} & c_{20} & c_{30} \\ c_{01} & c_{11} & c_{21} & c_{31} \\ c_{02} & c_{12} & c_{22} & c_{32} \\ c_{03} & c_{13} & c_{23} & c_{33} \end{bmatrix} \begin{bmatrix} \overrightarrow{v_0}.x \\ \overrightarrow{v_0}.y \\ \overrightarrow{v_0}.z \\ \overrightarrow{v_0}.w \end{bmatrix}$$

2 TRANSFORM

$$M_{TRS} = M_T M_R M_S$$

$$= \begin{bmatrix} M_R.c_{00} \times M_S.c_{00} & M_R.c_{10} \times M_S.c_{11} & M_R.c_{20} \times M_S.c_{22} & M_T.c_{30} \\ M_R.c_{01} \times M_S.c_{00} & M_R.c_{11} \times M_S.c_{11} & M_R.c_{21} \times M_S.c_{22} & M_T.c_{31} \\ M_R.c_{02} \times M_S.c_{00} & M_R.c_{12} \times M_S.c_{11} & M_R.c_{22} \times M_S.c_{22} & M_T.c_{32} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_{TRS}^{-1} = M_S^{-1} M_R^{-1} M_T^{-1}$$

2.1 TRANSLATION

$$M_T = \begin{bmatrix} 1 & 0 & 0 & x \\ 0 & 1 & 0 & y \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_T^{-1} = \begin{bmatrix} 1 & 0 & 0 & -x \\ 0 & 1 & 0 & -y \\ 0 & 0 & 1 & -z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.2 SCALE

$$M_S = \begin{bmatrix} x & 0 & 0 & 0 \\ 0 & y & 0 & 0 \\ 0 & 0 & z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_S^{-1} = \begin{bmatrix} {}^{1}/_{x} & 0 & 0 & 0 \\ 0 & {}^{1}/_{y} & 0 & 0 \\ 0 & 0 & {}^{1}/_{z} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.3 ROTATION

$$M_R = \begin{bmatrix} c_{00} & c_{10} & c_{20} & 0 \\ c_{01} & c_{11} & c_{21} & 0 \\ c_{02} & c_{12} & c_{22} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_R^{-1} = \begin{bmatrix} c_{00} & c_{01} & c_{02} & 0 \\ c_{10} & c_{11} & c_{12} & 0 \\ c_{20} & c_{21} & c_{22} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.3.1 EULER ROTATION

$$M_R = M_{Ra} M_{Rb} M_{Rc}$$

$$(a, b, c) \in \{(x, y, z), (x, z, y), (y, x, z), (y, z, x), (z, x, y), (z, y, x)\}$$

2.3.1.1 ROTATE AROUND X AXIS

$$M_{Rx} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.3.1.2 ROTATE AROUND Y AXIS

$$M_{Ry} = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.3.1.3 ROTATE AROUND Z AXIS

$$M_{Rz} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.3.2 QUATERNION

$$i^2 = j^2 = k^2 = ijk = -1$$

$$ij = k \quad ji = -k \quad jk = i \quad kj = -i \quad ki = j \quad ik = -j$$

$$\vec{a} = x_a i + y_a j + z_a k, \quad |\vec{a}| = 1$$

$$q = \vec{a}\sin\frac{\theta}{2} + \cos\frac{\theta}{2} = (\vec{v}, w) = xi + yj + zk + w$$

$$q^{-1} = \vec{a}\sin\left(-\frac{\theta}{2}\right) + \cos\left(-\frac{\theta}{2}\right) = (\overrightarrow{-v}, w) = -(xi + yj + zk) + w$$

$$|q| = 1$$

2.3.2.1 MULTIPY

$$q_1 = (\overrightarrow{v_1}, w_1) = x_1 i + y_1 j + z_1 k + w_1$$

$$q_2 = (\overrightarrow{v_2}, w_2) = x_2 i + y_2 j + z_2 k + w_2$$

$$q_1 q_2 = (w_2\overrightarrow{v_1} + w_1\overrightarrow{v_2} + \overrightarrow{v_1} \times \overrightarrow{v_2}, \quad w_1 w_2 - \overrightarrow{v_1} \cdot \overrightarrow{v_2})$$
$$= \begin{bmatrix} w_1 & -z_1 & y_1 & x_1 \\ z_1 & w_1 & -x_1 & y_1 \\ -y_1 & x_1 & w_1 & z_1 \\ -x_1 & -y_1 & -z_1 & w_1 \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ w_2 \end{bmatrix}$$

2.3.2.2 DOT

$$q_1 \cdot q_2 = |q_1||q_2|\cos\theta = \cos\theta = x_1 x_2 + y_1 y_2 + z_1 z_2 + w_1 w_2$$

2.3.2.3 INTERPOLATION

$$\Delta q = q_2 q_1^{-1}$$

$$q_t = NLerp(q_1, q_2, t) \ or \ SLerp(q_1, q_2, t)$$

2.3.2.4 ROTATE VECTOR

Rotate $\overrightarrow{v_0}$ around $\vec{a}$ axis by θ.

$$\overrightarrow{v_0} = x_0 i + y_0 j + z_0 k$$

$$\vec{v} = q\overrightarrow{v_0}q^{-1}$$

2.3.2.5 MATRIX

$$M_R = \begin{bmatrix} 1-2y^2-2z^2 & 2xy+2zw & 2xz-2yw & 0 \\ 2xy-2zw & 1-2x^2-2z^2 & 2yz+2xw & 0 \\ 2xz+2yw & 2yz-2xw & 1-2x^2-2y^2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$q = \begin{cases} x = \dfrac{c_{21}-c_{12}}{4w} \\ y = \dfrac{c_{20}-c_{02}}{4w} \\ z = \dfrac{c_{10}-c_{01}}{4w} \\ w = \dfrac{1}{2}\sqrt{1+c_{00}+c_{11}+c_{22}} \end{cases}$$

2.3.3 EULER TO QUATERNION

$$s_x = \sin\frac{\theta_x}{2} \quad s_y = \sin\frac{\theta_y}{2} \quad s_z = \sin\frac{\theta_z}{2}$$

$$c_x = \cos\frac{\theta_x}{2} \quad c_y = \cos\frac{\theta_y}{2} \quad c_z = \cos\frac{\theta_z}{2}$$

$$q_x = s_x i + c_x \quad q_y = s_y j + c_y \quad q_z = s_z k + c_z$$

$$q_a q_b q_c = \Big(\left(s_x c_y c_z + sign_1 s_y s_z c_x\right)i + \left(s_y c_x c_z + sign_2 s_x s_z\, c_y\right)j + \left(s_z c_x c_y + sign_3 s_x s_y c_z\right)k + \left(c_x c_y c_z + sign_4 s_x s_y s_z\right) \Big)$$

$$(a, b, c, sign_1, sign_2, sign_3, sign_4) \in \begin{Bmatrix} (x, y, z, -1, 1-1, 1) \\ (x, z, y, 1, 1, -1, -1) \\ (y, x, z, -1, 1, 1, -1) \\ (y, z, x, -1, -1, 1, 1) \\ (z, x, y, 1, -1, -1, 1) \\ (z, y, x, 1, -1, 1, -1) \end{Bmatrix}$$

$$\overrightarrow{nr} = \text{normalize}\left(\overrightarrow{right}\right)$$

$$\overrightarrow{nu} = \text{normalize}(\overrightarrow{up})$$

$$\overrightarrow{nf} = \text{normalize}\left(\overrightarrow{forward}\right)$$

$$M_R = \begin{bmatrix} \overrightarrow{nr}.x & \overrightarrow{nu}.x & \overrightarrow{nf}.x & 0 \\ \overrightarrow{nr}.y & \overrightarrow{nu}.y & \overrightarrow{nf}.y & 0 \\ \overrightarrow{nr}.z & \overrightarrow{nu}.z & \overrightarrow{nf}.z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3 RENDER PIPELINE

3.1 CAMERA PARAMETERS

$$n = \text{near clip plane distance value from camera}$$

$$f = \text{far clip plane distance value from camera}$$

$$l = \text{min x coordinate value at near plane}$$

$$r = \text{max x coordinate value at near plane}$$

$$b = \text{min y coordinate value at near plane}$$

$$t = \text{max y coordinate value at near plane}$$

$$a = \text{aspect} = \frac{r-l}{t-b} \cap \{t = -b, r = -l\}$$

$$\theta = \text{field of view} = 2\arctan\frac{(t-b)}{2n} \cap \{t = -b, r = -l\}$$

3.2 OBJECT SPACE TO WORLD SPACE

$$\begin{bmatrix} x_{world} \\ y_{world} \\ z_{world} \\ 1 \end{bmatrix} = M_M \begin{bmatrix} x_{object} \\ y_{object} \\ z_{object} \\ 1 \end{bmatrix}$$

$$M_M = M_{TRS_{object}}$$

3.3 WORLD SPACE TO VIEW SPACE

$$\begin{bmatrix} x_{view} \\ y_{view} \\ z_{view} \\ 1 \end{bmatrix} \cap \{-n \geq z_{view} \geq -f\} = M_V \begin{bmatrix} x_{world} \\ y_{world} \\ z_{world} \\ 1 \end{bmatrix}$$

$$M_V = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} M_R^{-1}{}_{camera} M_T^{-1}{}_{camera}$$

3.4 VIEW SPACE TO CLIP SPACE (PROJECTION)

In DirectX platform, Unity remap OpenGL z and reverse z depth into DirectX z

$$M_{GL2DX} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3.4.1 ORTHOGRAPHIC

3.4.1.1 OPENGL

$$\begin{bmatrix} x_{clip} \\ y_{clip} \\ z_{clip} \\ w_{clip} = 1 \end{bmatrix} \cap \begin{cases} -1 \leq x_{clip} \leq 1 \\ -1 \leq y_{clip} \leq 1 \\ -1 \leq z_{clip} \leq 1 \end{cases} = M_P \begin{bmatrix} x_{view} \\ y_{view} \\ z_{view} \\ 1 \end{bmatrix}$$

$$M_P = M_{ortho_{GL}} = \begin{bmatrix} \frac{2}{r-l} & 0 & 0 & -\frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & -\frac{t+b}{t-b} \\ 0 & 0 & \frac{-2}{f-n} & -\frac{f+n}{f-n} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3.4.1.2 DIRECTX

$$\begin{bmatrix} x_{clip} \\ y_{clip} \\ z_{clip} \\ w_{clip} = 1 \end{bmatrix} \cap \begin{cases} -1 \le x_{clip} \le 1 \\ -1 \le y_{clip} \le 1 \\ 1 \ge z_{clip} \ge 0 \end{cases} = M_P \begin{bmatrix} x_{view} \\ y_{view} \\ z_{view} \\ 1 \end{bmatrix}$$

$$\text{Original } M_{ortho_{DX}} = \begin{bmatrix} \frac{2}{r-l} & 0 & 0 & -\frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & -\frac{t+b}{t-b} \\ 0 & 0 & \frac{1}{f-n} & -\frac{n}{f-n} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\text{But in Unity } M_P = M_{GL2DX} M_{ortho_{GL}} = \begin{bmatrix} \frac{2}{r-l} & 0 & 0 & -\frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & -\frac{t+b}{t-b} \\ 0 & 0 & \frac{1}{f-n} & \frac{f}{f-n} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3.4.2 PERSPECTIVE

3.4.2.1 OPENGL

$$\begin{bmatrix} x_{clip} \\ y_{clip} \\ z_{clip} \\ w_{clip} = -z_{view} = z_{from\ camera} \end{bmatrix} \cap \begin{cases} -n \le x_{clip} \le f \\ -n \le y_{clip} \le f \\ -n \le z_{clip} \le f \end{cases} = M_P \begin{bmatrix} x_{view} \\ y_{view} \\ z_{view} \\ 1 \end{bmatrix}$$

$$M_P = M_{persp_{GL}} = \begin{bmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{-(f+n)}{f-n} & \frac{-2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

3.4.2.2 DIRECTX

$$\begin{bmatrix} x_{clip} \\ y_{clip} \\ z_{clip} \\ w_{clip} = -z_{view} = z_{from\ camera} \end{bmatrix} \cap \begin{cases} -n \le x_{clip} \le f \\ -n \le y_{clip} \le f \\ n \ge z_{clip} \ge 0 \end{cases} = M_P \begin{bmatrix} x_{view} \\ y_{view} \\ z_{view} \\ 1 \end{bmatrix}$$

$$\text{Original } M_{persp_{DX}} = \begin{bmatrix} \frac{2n}{r-l} & 0 & -\frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & -\frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{f}{f-n} & \frac{-fn}{f-n} \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\text{But in Unity } M_P = M_{GL2DX} M_{persp_{GL}} = \begin{bmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{n}{f-n} & \frac{fn}{f-n} \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

3.5 CLIP POSITION TO NORMALIZED DEVICE COORDINATES (NDC)

$$\begin{bmatrix} x_{NDC} = \frac{x_{clip}}{w_{clip}} \\ y_{NDC} = \frac{y_{clip}}{w_{clip}} \\ z_{NDC} = \frac{z_{clip}}{w_{clip}} \\ 1 \end{bmatrix} \cap \left\{ \begin{array}{l} GL \begin{cases} -1 \le x_{NDC} \le 1 \\ -1 \le y_{NDC} \le 1 \\ -1 \le z_{NDC} \le 1 \end{cases} \\ \quad or \\ DX \begin{cases} -1 \le x_{NDC} \le 1 \\ -1 \le y_{NDC} \le 1 \\ 1 \ge z_{NDC} \ge 0 \end{cases} \end{array} \right.$$

3.6 POSITION IN SHADER

3.6.1 VERTEX OUTPUT

$$SV\_POSITION = \begin{bmatrix} x_{clip} \\ y_{clip} \\ z_{clip} \\ w_{clip} \end{bmatrix} = M_P M_V M_M \begin{bmatrix} x_{object} \\ y_{object} \\ z_{object} \\ 1 \end{bmatrix}$$

3.6.2 FRAGMENT INPUT

$$SV\_POSITION = \begin{bmatrix} \left(\frac{x_{NDC}}{2} + \frac{1}{2}\right) \times output\ texture\ width + \frac{1}{2} \\ \left(\frac{y_{NDC}}{2} + \frac{1}{2}\right) \times output\ texture\ height + \frac{1}{2} \\ z_{NDC} \\ w_{clip} \end{bmatrix}$$

3.7 DEPTH

3.7.1 WRITE INTO DEPTH TEXTURE

$$r = \begin{cases} GL{:}\ \frac{z_{NDC}}{2} + \frac{1}{2} \\ or \\ DX{:}\ z_{NDC} \end{cases}$$

3.7.2 READ FROM DEPTH TEXTURE

$$zP = GL \begin{cases} x = 1 - \frac{f}{n} \\ y = \frac{f}{n} \\ z = \frac{1}{f} - \frac{1}{n} \\ w = \frac{1}{n} \end{cases} \quad or \quad DX \begin{cases} x = -1 + \frac{f}{n} \\ y = 1 \\ z = -\frac{1}{f} + \frac{1}{n} \\ w = \frac{1}{f} \end{cases}$$

$$\mathrm{RevZ}(r) = \begin{cases} GL{:}\ r \\ or \\ DX{:}\ 1 - r \end{cases}$$

$$depth = \begin{cases} ortho \begin{cases} 01 \begin{cases} from\ 0\text{: } \dfrac{(f-n)\times \mathrm{RevZ}(r) + n}{f} \\ from\ n\text{: } \mathrm{RevZ}(r) \end{cases} \\ eye \begin{cases} from\ 0\text{: } (f-n)\times \mathrm{RevZ}(r) + n \\ from\ n\text{: } (f-n)\times \mathrm{RevZ}(r) \end{cases} \end{cases} \\ persp \begin{cases} 01 \begin{cases} from\ 0\text{: } \dfrac{1}{zP.x \times r + zP.y} \\ from\ n\text{: } \dfrac{1}{zP.x + \dfrac{zP.y}{r}} \end{cases} \\ eye \begin{cases} from\ 0\text{: } \dfrac{1}{zP.z \times r + zP.w} \\ from\ n\text{: } \dfrac{1}{P.z \times r + zP.w} - n \end{cases} \end{cases} \end{cases}$$