Aiamg

Tommy Petersen
tp@aia.dk

August 22, 2021

1 Introduction

These notes are about a 3D system called **Aiamg**.

2 The transformation matrix

The transformation matrix is a four by four matrix which contains a three by three rotation matrix and a three by one translation vector:

$$\begin{pmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Let R be the rotation matrix and let T be the translation vector, then we can write the transformation matrix as

$$\begin{pmatrix} & R & & T \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

2.1 Multiplying by a rotation matrix

Each rotation matrix is on the form

$$\begin{pmatrix} r'_{11} & r'_{12} & r'_{13} & 0 \\ r'_{21} & r'_{22} & r'_{23} & 0 \\ r'_{31} & r'_{32} & r'_{33} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Multiplying the transformation matrix by the rotation matrix on the right, we have the following:

$$\begin{pmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} r'_{11} & r'_{12} & r'_{13} & 0 \\ r'_{21} & r'_{22} & r'_{23} & 0 \\ r'_{31} & r'_{32} & r'_{33} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} \sum_{j=1}^3 r_{1j}r'_{j1} & \sum_{j=1}^3 r_{1j}r'_{j2} & \sum_{j=1}^3 r_{1j}r'_{j3} & t_1 \\ \sum_{j=1}^3 r_{2j}r'_{j1} & \sum_{j=1}^3 r_{2j}r'_{j2} & \sum_{j=1}^3 r_{2j}r'_{j3} & t_2 \\ \sum_{j=1}^3 r_{3j}r'_{j1} & \sum_{j=1}^3 r_{3j}r'_{j2} & \sum_{j=1}^3 r_{3j}r'_{j3} & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

As we can see, multiplying the transformation matrix by a rotation matrix on the right gives a four by four matix with only the original rotation submatrix affected.

2.2 Multiplying by a translation matrix

Each translation matrix is on the form

$$\begin{pmatrix} 1 & 0 & 0 & t'_1 \\ 0 & 1 & 0 & t'_2 \\ 0 & 0 & 1 & t'_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Multiplying the transformation matrix by the translation matrix on the left, we have the following:

$$\begin{pmatrix}1 & 0 & 0 & t_1'\\ 0 & 1 & 0 & t_2'\\ 0 & 0 & 1 & t_3'\\ 0 & 0 & 0 & 1\end{pmatrix} \cdot \begin{pmatrix}r_{11} & r_{12} & r_{13} & t_1\\ r_{21} & r_{22} & r_{23} & t_2\\ r_{31} & r_{32} & r_{33} & t_3\\ 0 & 0 & 0 & 1\end{pmatrix}$$

$$= \begin{pmatrix}r_{11} & r_{12} & r_{13} & t_1' + t_1\\ r_{21} & r_{22} & r_{23} & t_2' + t_2\\ r_{31} & r_{32} & r_{33} & t_3' + t_3\\ 0 & 0 & 0 & 1\end{pmatrix}$$

As we can see, multiplying the transformation matrix by a translation matrix on the left gives a four by four matix with only the original translation subvector affected.

2.3 Multiplying by a point vector

Each point vector is on the form

$$\begin{pmatrix}p_1\\ p_2\\ p_3\\ 1\end{pmatrix}$$

Multiplying the transformation matrix by the point vector on the right, we have the following:

$$\begin{pmatrix}r_{11} & r_{12} & r_{13} & t_1\\ r_{21} & r_{22} & r_{23} & t_2\\ r_{31} & r_{32} & r_{33} & t_3\\ 0 & 0 & 0 & 1\end{pmatrix} \cdot \begin{pmatrix}p_1\\ p_2\\ p_3\\ 1\end{pmatrix}$$

$$= \begin{pmatrix}\sum_{j=1}^{3} r_{1j}p_j + t_1\\ \sum_{j=1}^{3} r_{2j}p_j + t_2\\ \sum_{j=1}^{3} r_{3j}p_j + t_3\\ 1\end{pmatrix}$$

As we can see, multiplying the transformation matrix by a point vector on the right gives a four by one vector where the original point has been transformed by the rotation submatrix and then translated by the translation submatrix.

3 Lines i 3D

Proposition 3.1 (Existence of scaled point on line) *Given the point $P_0 := (x_0, y_0, z_0)$ not equal to $(0, 0, 0)$. Then for every $t \in \mathbb{R}$, the point (tx_0, ty_0, tz_0) is on the line through $(0, 0, 0)$ and P_0.*

Proof

Case when x_0 is not equal to zero
If x_0 is not equal to zero, then for L's coordinate projection on the z plane, $z = 0$, the variable y is a linear function of x:

$$y = x\frac{y_0}{x_0}$$

This function is defined on all of $\mathbb{R}$ and especially in the point $x = tx_0$. We can conclude, that a point with x-value tx_0 is on the line.

Case when x_0 is equal to zero
If x_0 is equal to zero, then the line is fully contained in the plane $x = 0$ so specifically there exists at least one point on the line with x-value equal to tx_0 (which has the value zero).

The Cases for y_0 and z_0 are shown similarly. □

4 Triangulation

4.1 Split longest line

In figure 1 we want to split the longest line c in two equal halves:

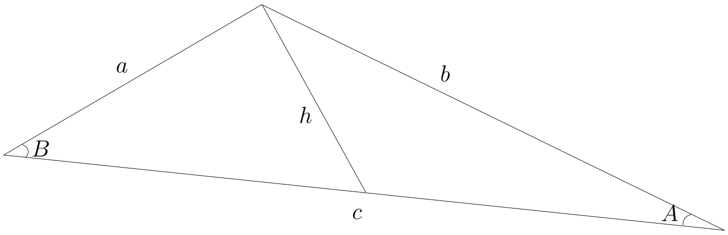

Figure 1: Split longest line c in two equal halves

But in order to guarantee algorithm termination, the split must not add another line of the same length or greater than that of c. The line used for the split is h so we must prove, that h is strictly less than c.

As figure 2 shows, there are three cases for splitting c based on the angle θ between h and c:

- Case I: $\theta = \pi/2$: Then $\cos\theta = 0$.
- Cases II_a and II_b: $\pi/2 < \theta < \pi$: Then $\cos\theta < 0$.

In all three cases, we have that $\cos\theta \le 0$. Using the law of cosines, we have that

$$\begin{aligned} b^2 &= h^2 + (c/2)^2 - 2h(c/2)\cos\theta \\ &= h^2 + (c/2)^2 - hc\cos\theta \\ &\ge h^2 + (c/2)^2 && (\cos\theta \le 0) \end{aligned}$$

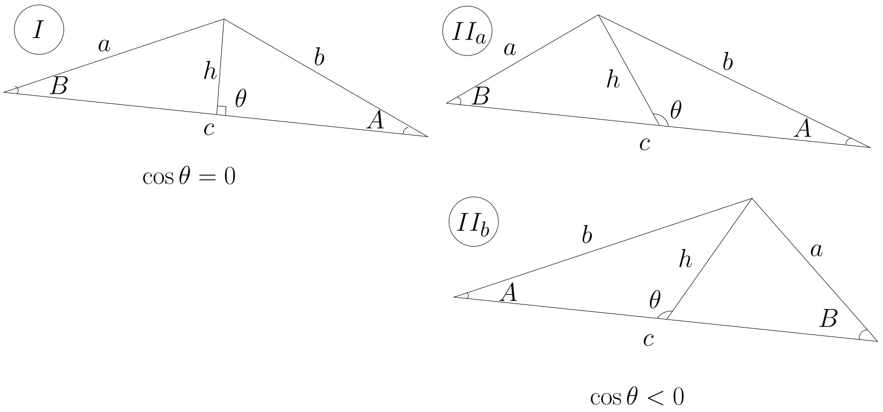

Figure 2: Cases for the split

Hence

$$\begin{aligned} h^2 &\leq b^2 - (c/2)^2 \\ &< b^2 \end{aligned}$$

So that for h we have

$$h < b$$

Since c has longest length, we have, that $b \leq c$. Hence we can conclude, that $h < c$.

5 Color fading

In color fading, a color of highest intensity, say 1.0, has the intensity fade away to 0.0 resembling some fogginess in the distance. A given point in the camera's view volume then has the intensity fade by linear interpolation on the point's z-coordinate relative to some fixed background z-value. But how about a line? Could we calculate the faded intensity for each of the endpoints and then interpolate the intensity along the line? The following example shows that this is not the correct way to fade a line.

Consider figure 3. Here, the singular point is P_1 with a z-value of 100 and the line has endpoints Q_0 with a z-value of -1200 and Q_2 with a z-value of 1200. The projection plane has z-value 100 and the background has z-value is 1000.

When fading the line, Q_0 gets highest intensity 1.0 and Q_2 gets lowest intensity 0.0. Calculating the intensity of Q_1 via linear interpolation of the intensities of Q_0 and Q_2, we get the value

$$(1 - \frac{1300}{2400}) \cdot 1,0 + \frac{1300}{2400} \cdot 0,0 \approx 1,0 - 0,5417 = 0,4583$$

When fading the point P_1, we get the value

$$(1 - \frac{100}{1000}) \cdot 1,0 + \frac{100}{1000} \cdot 0,0 \approx 1,0 - 0,10 = 0,90$$

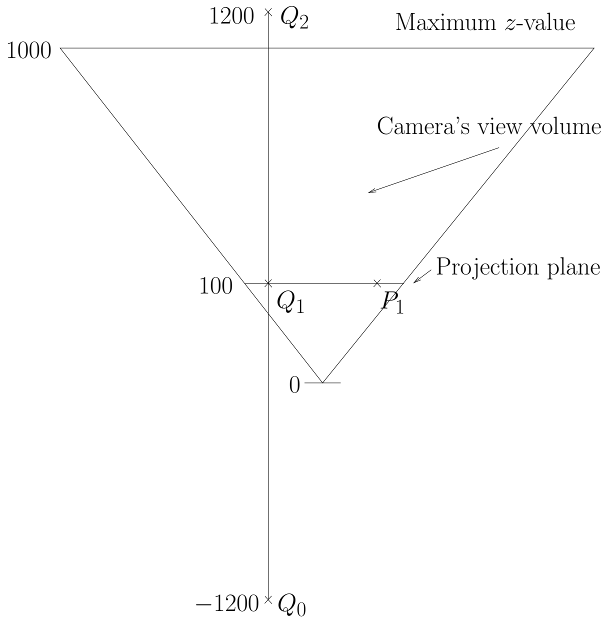

Figure 3: Fading a line

So Q_1 gets the intensity $0,4583$ and P_1 gets the intensity $0,90$. But they have the same z-values and hence should have the same intensity, so fading a line should not be done by interpolation of faded endpoints.

A correct way to fade a line, and all other types of objects, is to fade the individual pixels when they are rendered. This can be made to ensure that pixels rendered from the same z-values get the same intensity.