

Method of model projective transformations

In Transport Fever, a mod creator can do transforms to any model, it's easy once the transformations is an affine transformations.

Written in vector and matrix:

With

$$M = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$A = (x \quad y \quad z \quad 1)^T$$

$$B = (x' \quad y' \quad z' \quad 1)^T$$

and

$$MA = B$$

Where

$$x' = x \cdot m_{11} + y \cdot m_{12} + z \cdot m_{13} + m_{14}$$

$$y' = x \cdot m_{21} + y \cdot m_{22} + z \cdot m_{23} + m_{24}$$

$$z' = x \cdot m_{31} + y \cdot m_{32} + z \cdot m_{33} + m_{34}$$

So the between the original coordinate (x, y, z)? and (x', y', z')?, it's an linear projection. For any triangle:

$$X = \begin{pmatrix} x_1 & y_1 & z_1 & 1 \\ y_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \end{pmatrix}^T$$

With the same matrix you can get

$$Y = \begin{pmatrix} x'_1 & y'_1 & z'_1 & 1 \\ y'_2 & y'_2 & z'_2 & 1 \\ x'_3 & y'_3 & z'_3 & 1 \end{pmatrix}^T$$

So

$$MX = Y$$

And

$$M = MX X^{-1} = Y X^{-1}$$

In this way we will be able to get the M we need

But since the rank(X) = 3 and rank(Y) = 3, and the rank(M) = 4, with two triangles, you can't get such M. So we need to introduce another extra point in X and Y, and these (two) points should not be coplanar with the rest three in order to have 4-rank matrices, for example

$$X = \begin{pmatrix} x_1 & y_1 & z_1 & 1 \\ y_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_1 & y_1 & z_1 - 1 & 1 \end{pmatrix}^T$$

$$Y = \begin{pmatrix} x'_1 & y'_1 & z'_1 & 1 \\ y'_2 & y'_2 & z'_2 & 1 \\ x'_3 & y'_3 & z'_3 & 1 \\ x'_1 & u'_1 & z'_1 - 1 & 1 \end{pmatrix}^T$$

Then you will get the M you want.

So back to the subject, how to do any transformation to any meshes?

In the Ultimate Station, each unit is in a rectangle mesh viewed from Z, when the station is curved, the rectangle becomes trapezium, this is NOT an affine transformation but a perspective transformation! But it doesn't matter, you can divide a rectangle into two triangles, and you will be able to get two triangles at the output side, the difference is you need two different affine transformation matrices to finish this.

This is the method I used.

Is there a better way to avoid the division?

There is, but not implemented by the game.

Note the matrix M, the last line is always (0, 0, 0, 1), and note the vector that I have written above, there are always four elements, the last is always 1.

If it's a perspective transformation, the last line of M will be no longer (0, 0, 0, 1), but something else. However, in this case, the output vector will have a non 1 element as last, we call it w

We will have

$$M \begin{pmatrix} x & y & z & 1 \end{pmatrix}^T = \begin{pmatrix} x' & y' & z' & w' \end{pmatrix}^T$$

With five non-coplanar coordinates, you will be able to get an matrix M that can transform a rectangle to any form in the space, however, you need to deal with the last element w, to make it 1

$$\begin{pmatrix} x' & y' & z' & w' \end{pmatrix}^T \rightarrow \begin{pmatrix} x'/w' & y'/w' & z'/w' & 1 \end{pmatrix}^T$$

However, this can not be done with the mod, it should be done with the compiled code of Transport Fever, and UG didn't do this step.

So the only way to do such transformation is to divide your mesh diagonally.