Carlos Martinez
Semester Project
Feb 27, 2017
Math 2270

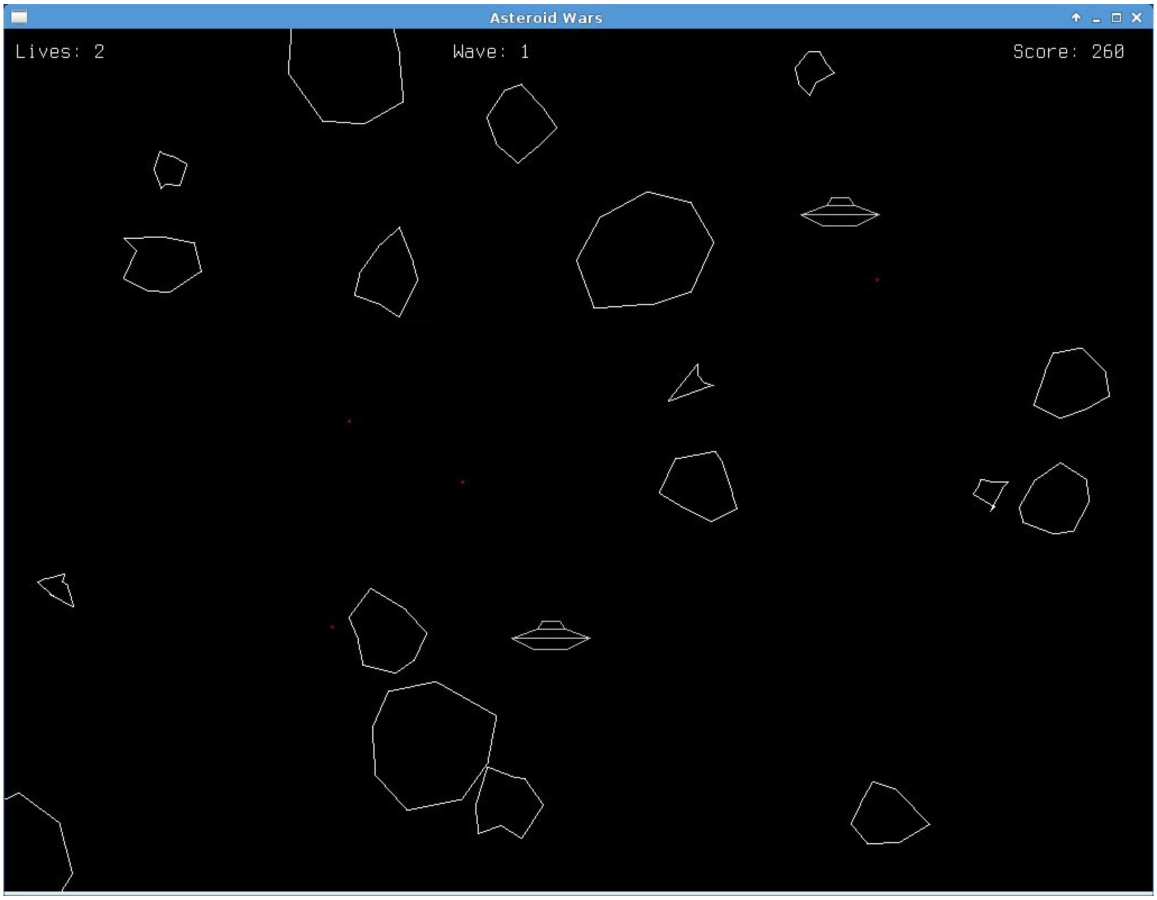
Use of Linear Algebra in Computer Science

Introduction

Computer graphics is the production of graphical image that are displayed on a computer screen. Architecture, cartoons, automotive, and video games are among the many industries where computer graphics play a huge roll in the job they do. In this paper we will discuss the basic techniques that are used to perform the many transformation the objects displayed go through to show movement with the help of linear algebra.

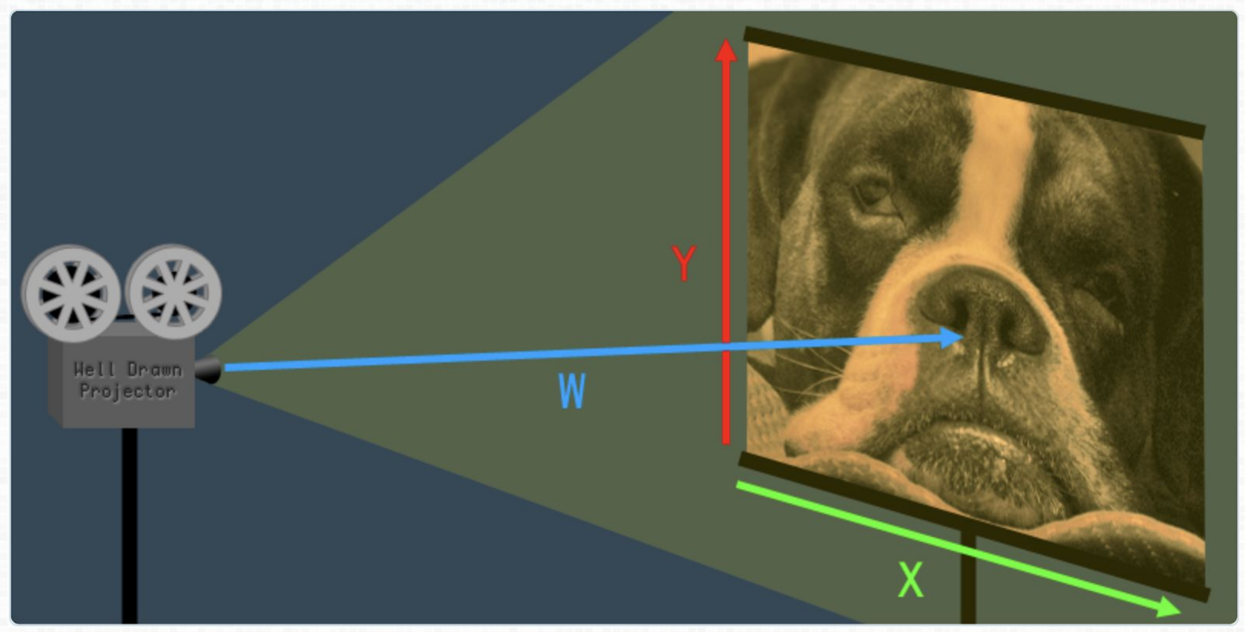
As you are probably well aware of, computers screens can hold up to millions of pixels. One would think that in order to store all the information need to display an image we would have to save the state of every pixel, but with the help of matrices we are able to represent all sort of movements and transformations on a screen with less data. The advantage of matrices is that if we make well use of points in space, we have store a great deal of information in small amount of space and then have the computer connect the lines. With the use of matrices we can represent an endless number of shapes where the only info we would need is the kind of shape it is, enough points to be able to display the image, as well as other properties of the shape, like color. To keep this topic simple enough to understand, we will leave theses extra properties out.

Asteroids Wars Computer Graphics Example



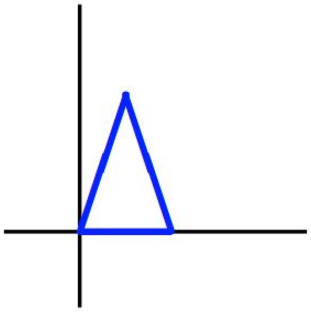
Homogeneous Coordinates

When it comes to computer graphics most would think to use (X,Y,Z) coordinates, but in reality most use (X,Y,Z,W) coordinates which contain an extra dimension. The fourth dimension is referred by many as projective space and the coordinates are known as homogeneous Coordinates.



To understand the W look at the image above. The W is the length from the projector to the screen. If you decrease W the 2D or 3D image get smaller, If you increase W the image gets bigger. The W coordinates can also be negative in which case the the image would invert upside down and left to right, but for most cases the the W is never negative. Computer graphics transformation use homogeneous coordinates and to keep the projection simple W will always be set to 1. The extra coordinate is sometimes referred as dummy coordinates because they don't add much if you always set it to 1. In computer graphics you can perform projection transformation in which the homogeneous coordinates are ideal, but due to its complexity we will not discuss such transformation, just the basics transformations.

Translating Transformation



Let this triangle above represent a ship, with the points (0,0), (2,0), (1,3).

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

The matrix above is the matrix representation of our ship. It is important to know that they are connected from left to right, with the last point connected to the first point.

$$\begin{bmatrix} 0 & 2 & 1 \\ 0 & 0 & 3 \end{bmatrix} + \begin{bmatrix} r & r & r \\ s & s & s \end{bmatrix}$$

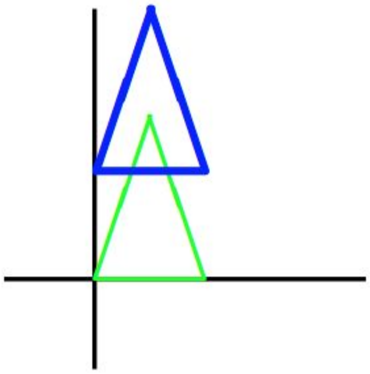
To get the ship’s new position we use the matrix above, where r is how much you move to the right, -r is how much you move to the left, s is how much you move up, and -s is how much you move down.

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

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

It is important to know that matrix multiplication is a more common way to do translation transformation. In order to be able to do matrix multiplication you have to change the matrix sometimes to hold dummy coordinates that just hold the value of one. Once you have your new matrix you can do your matrix multiplication with the equation above. r and s still stand for the same thing as before.

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 2 & 1 \\ 0 & 0 & 3 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 2 & 1 \\ 2 & 2 & 5 \\ 1 & 1 & 1 \end{bmatrix}$$

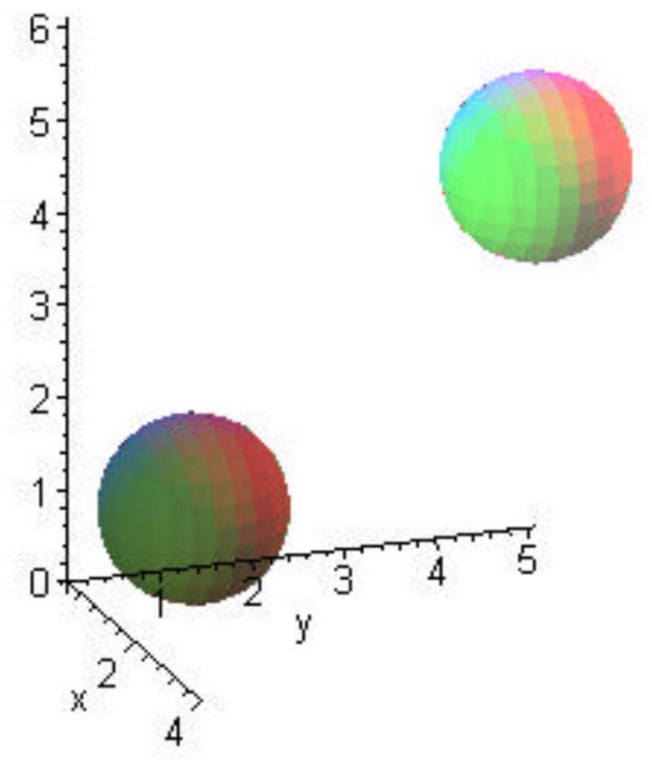


From our ship example we can see that the new points of our ship is now (0,2), (2,2), (1,5) from moving two units up.

For the 3D form of the equation is as below with a visual example.

Given any point (x, y, z) in R^3, the following will give the translated point.

$$\begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x + t_x \\ y + t_y \\ z + t_z \\ 1 \end{bmatrix}$$

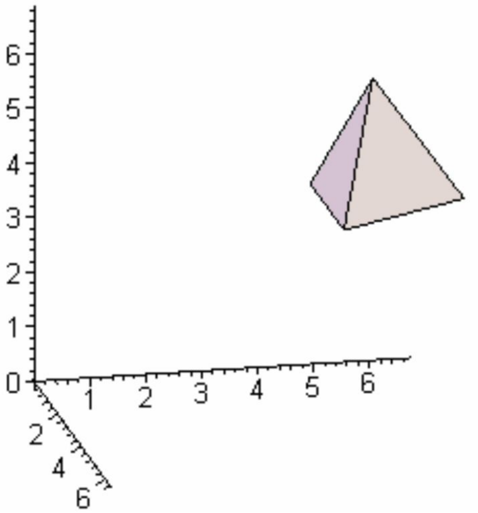


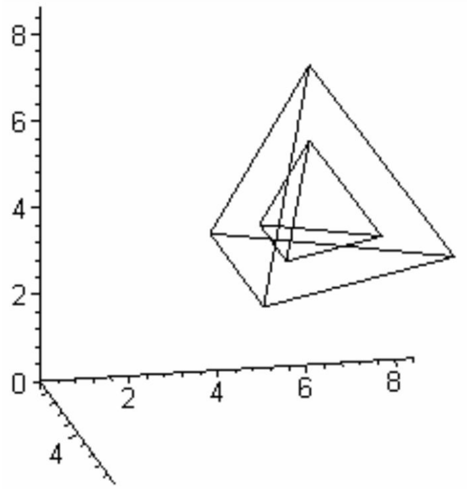
Scaling Transformation

To Scale an image the matrix transformation is even easier, because the only thing you have to do is multiply the image by a scale constant S. The S controls how the image changes, a bigger S makes the image larger, a smaller S shrink the image.

Given any point (x, y, z) in R^3, the following will give the scaled point.

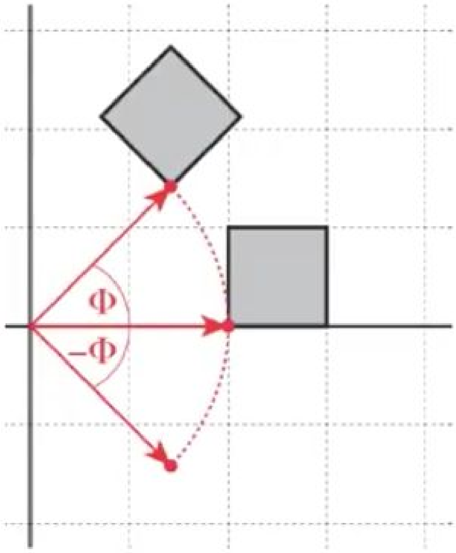
$$\begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} s_x x \\ s_y y \\ s_z z \\ 1 \end{bmatrix}$$





Rotation Transformation

When it comes rotating an object, both 2D and 3D object work the same. To rotate an object we rotate around the X, Y, and Z axis using trigonometry.



From trigonometry we get the equation below to rotate around the the X axis. Also know that in a 2D rotation the left matrix leaves out the top row, and the right matrix leaves out the z. The object, using this equation, will rotate in a counterclockwise direction. If you switch the signs on the sins below, you can get an equation that rotates clockwise, but the one equation is enough to get ever rotation you need in 2D.

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Equation to rotate around the Y axis in 3D below.

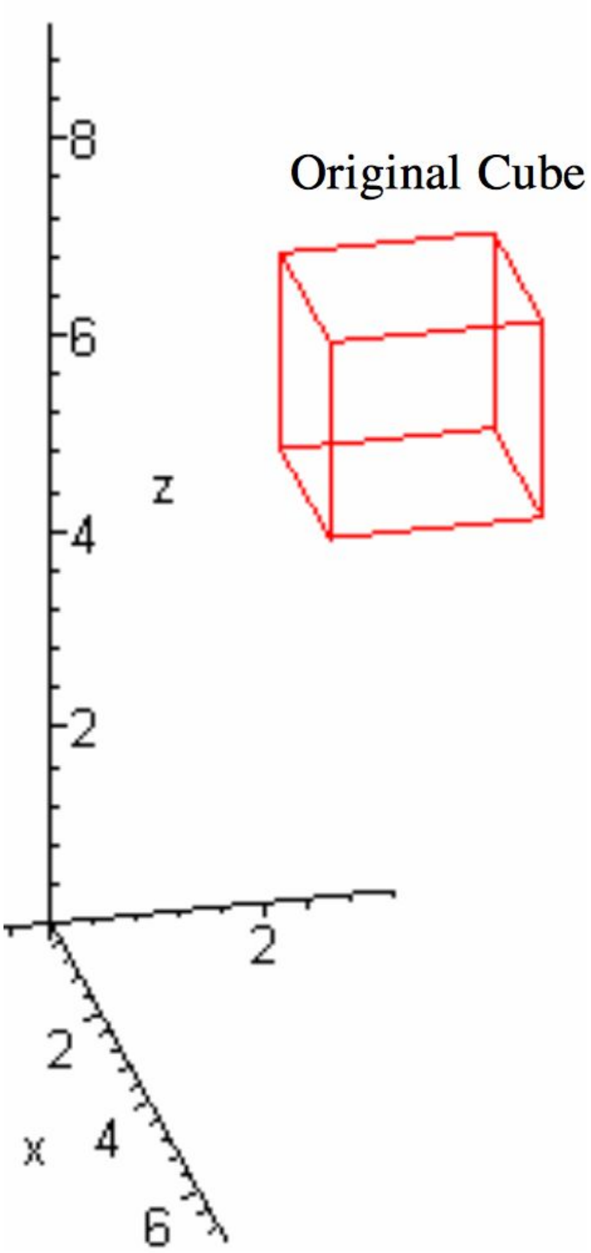
$$\begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

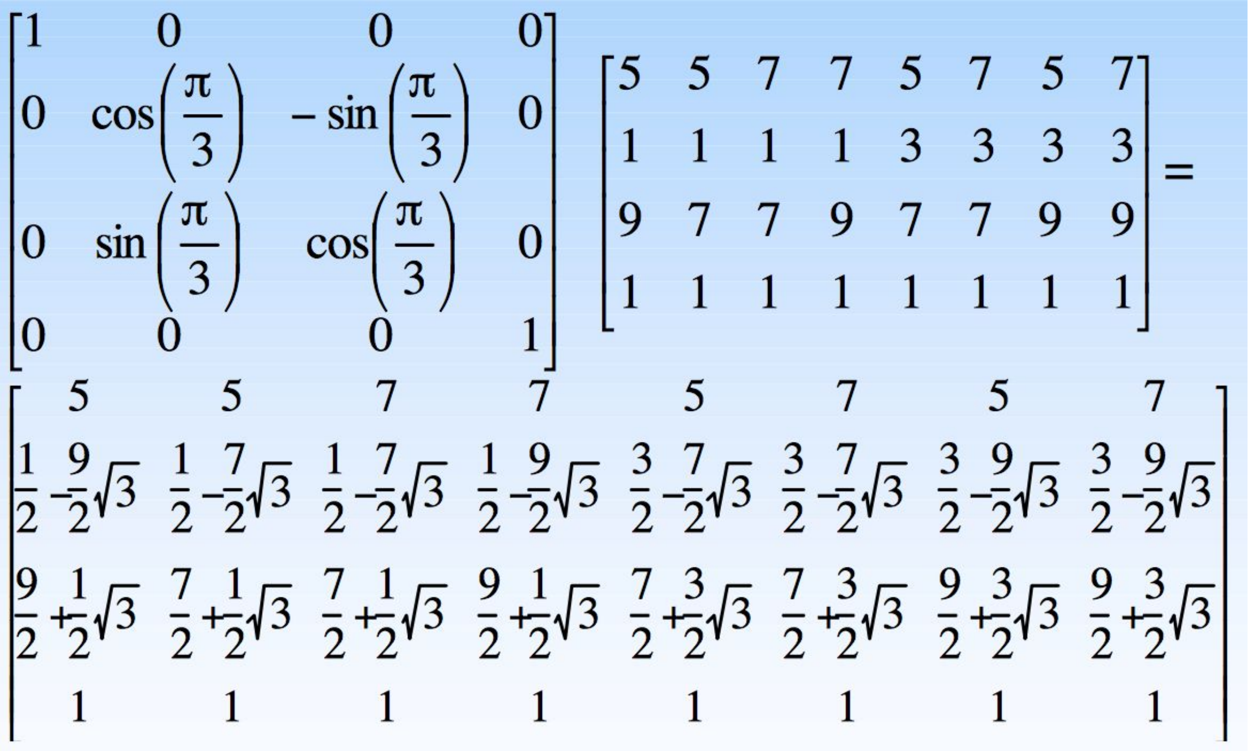
Equation to rotate around the Z axis in 3D below.

$$\begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

To ensure that we understand what we are doing when rotating, take a look at the example below. We are going to rotate a cube $\pi/3$ around the X axis. The matrix below will represent the cube, with the bottom row being a row of dummy coordinates to be able to do the matrix multiplication.

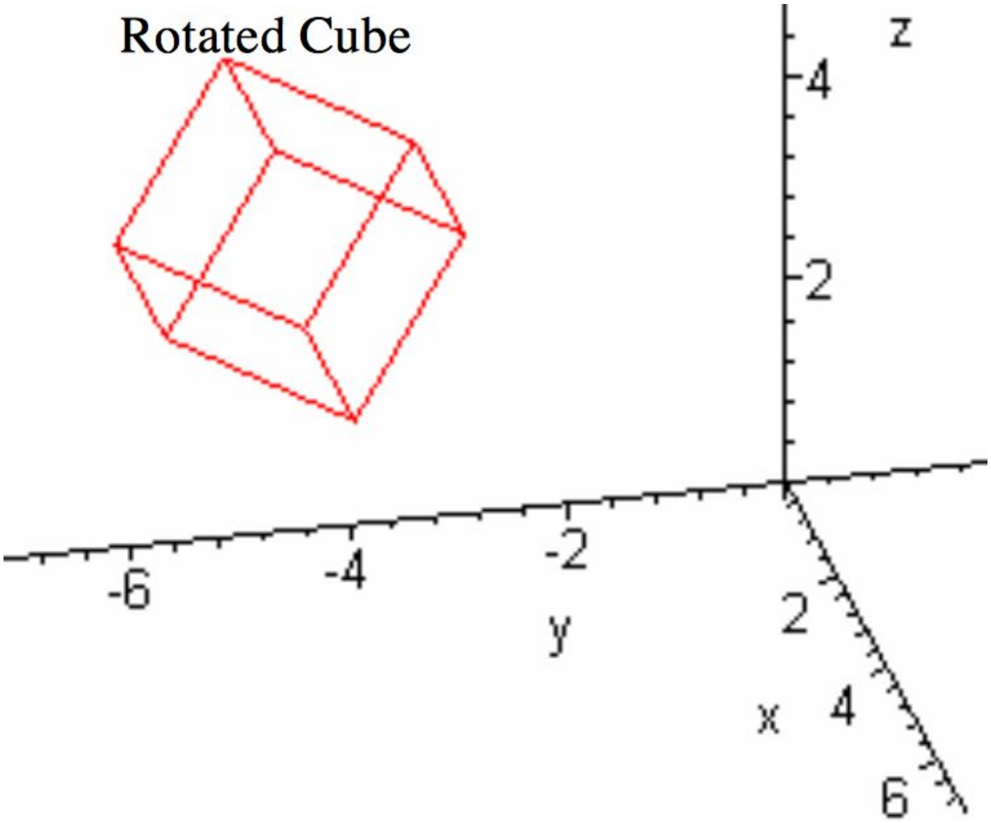
$$\begin{bmatrix} 5 & 5 & 7 & 7 & 5 & 7 & 5 & 7 \\ 1 & 1 & 1 & 1 & 3 & 3 & 3 & 3 \\ 9 & 7 & 7 & 9 & 7 & 7 & 9 & 9 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$



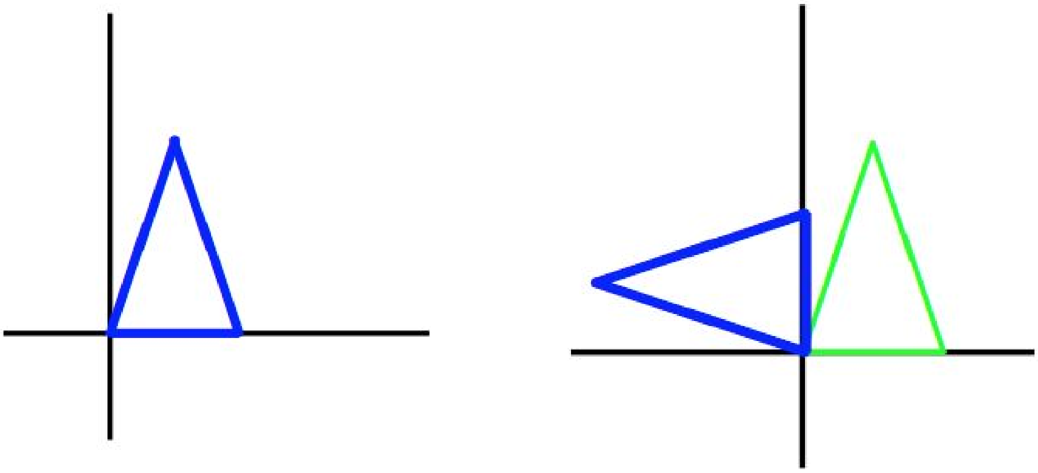


$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\left(\frac{\pi}{3}\right) & -\sin\left(\frac{\pi}{3}\right) & 0 \\ 0 & \sin\left(\frac{\pi}{3}\right) & \cos\left(\frac{\pi}{3}\right) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 5 & 5 & 7 & 7 & 5 & 7 & 5 & 7 \\ 1 & 1 & 1 & 1 & 3 & 3 & 3 & 3 \\ 9 & 7 & 7 & 9 & 7 & 7 & 9 & 9 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} =$$

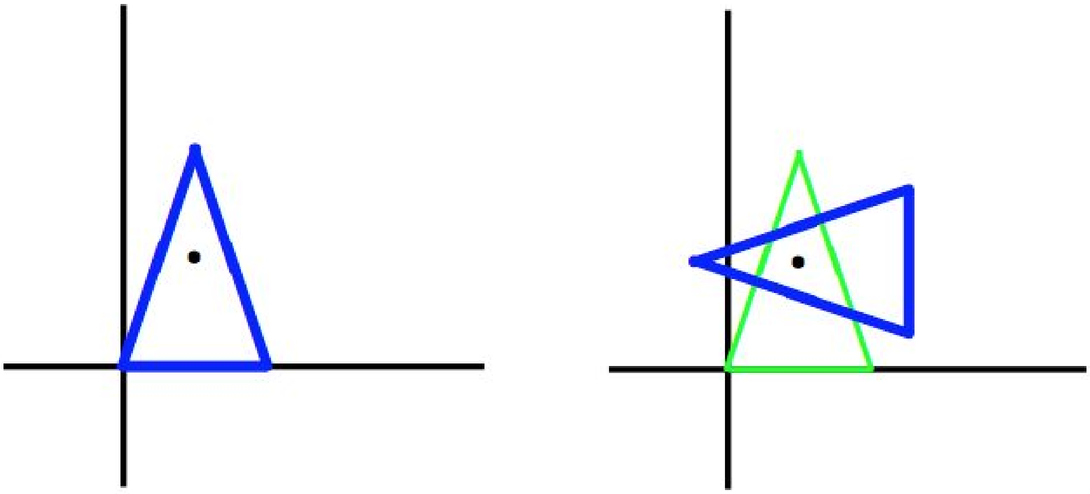
$$\begin{bmatrix} 5 & 5 & 7 & 7 & 5 & 7 & 5 & 7 \\ \frac{1}{2}-\frac{9}{2}\sqrt{3} & \frac{1}{2}-\frac{7}{2}\sqrt{3} & \frac{1}{2}-\frac{7}{2}\sqrt{3} & \frac{1}{2}-\frac{9}{2}\sqrt{3} & \frac{3}{2}-\frac{7}{2}\sqrt{3} & \frac{3}{2}-\frac{7}{2}\sqrt{3} & \frac{3}{2}-\frac{9}{2}\sqrt{3} & \frac{3}{2}-\frac{9}{2}\sqrt{3} \\ \frac{9}{2}+\frac{1}{2}\sqrt{3} & \frac{7}{2}+\frac{1}{2}\sqrt{3} & \frac{7}{2}+\frac{1}{2}\sqrt{3} & \frac{9}{2}+\frac{1}{2}\sqrt{3} & \frac{7}{2}+\frac{3}{2}\sqrt{3} & \frac{7}{2}+\frac{3}{2}\sqrt{3} & \frac{9}{2}+\frac{3}{2}\sqrt{3} & \frac{9}{2}+\frac{3}{2}\sqrt{3} \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$



From the above example you can see that the object is rotated around an axis, which means that the object does not just rotate but also change position.

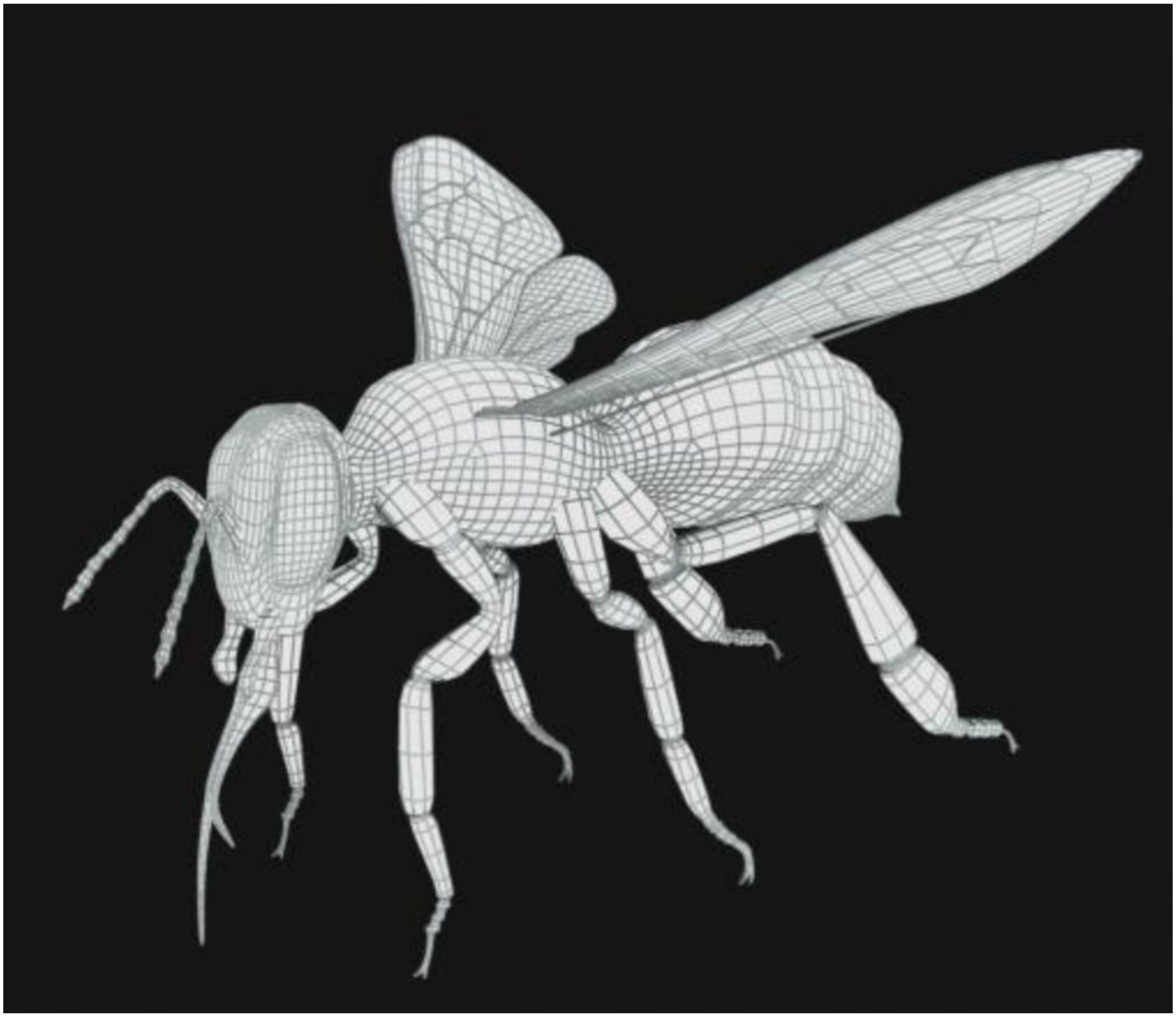


When rotating an object many people want the object to be in the same position as the original. To get the rotated object back to the original position all you need to do is a translation transformation. This usually require the user to keep track of the center of the object to know where to move it back.



Conclusion

When it comes to computer graphics, linear algebra makes up an important part of representing and manipulating computer data. There are a number of different transformation, but translating, scaling, and rotation are the three most commonly used. While this report mostly covers very basic examples to help you understand the concept, you must know that the material covered here can get extremely complex.



To do more complex transformation all you need is more points and objects, but the transformations are still done the same way.

References

Yip, Ting. *Matrices in Computer Graphics*. 2 Dec. 2001.

Lintlop, Matthew. *Computer Graphics and Linear Algebra*. 22 Aug. 2007.