LINEAR ALGEBRA IN COMPUTER GRAPHICS

Kundan Kumar Jha - 22MSM40052

Branch - MSC Data Science
Department of Mathematics

Chandigarh University
Mohali Punjab
November 2022

ABSTRACT

Linear algebra plays a vital role in the field of computer graphics. This paper will give a brief introduction to linear algebra in computer graphics. Matrixes are used in computer graphics to represent numerous different types of data. Games with 2D or 3D visuals rely on other matrix applications to show the game's location and characters. The linear algebra process in computer graphics is discussed further using examples from various areas of computer graphics. Linear algebra is a useful weapon for computer vision, data science, and other related fields like computer vision and natural language processing. Linear algebra has its own implementations in computer vision, such as the use of matrices, vectors, and tensors, as well as their processes such as linear transformations, matrix operations, linear correlation, and variable dependency.

Keywords: transformation, evaluation, reflection, rotation, shearing, reflection.

INTRODUCTION

Linear algebra is a field of mathematics that deals with systems of equations and their vector-space representations using matrices. Linear algebra, in other words, is the study of linear functions and vectors. It is one of mathematics' most important topics. The majority of current geometrical concepts are founded on linear algebra. Linear algebra simplifies the design of many natural phenomena and is thus an essential component of engineering and physics. The most key parts of this subject are linear equations, matrices, and vector spaces.

Elementary linear algebra means teaching the fundamentals of linear algebra. This includes basic matrix operations, various computations that can be performed on a system of linear equations, and vector aspects. The following are some key terms related to elementary linear algebra:
Scalars - The quantities that have only magnitude and no direction. It is a component used to define a vector space. Scalars are typically real numbers in linear algebra.
Vectors: A vector is a unit of measurement in vector space. It is a quantity that can describe both an element's direction and magnitude. The vector space is made up of vectors that can be added together and multiplied by scalars.
Matrix: A matrix is a rectangular array that organizes information into rows and columns. The majority of

linear algebraic properties, which can be expressed using a matrix.

Matrix Operations: These are simple arithmetic operations that can be performed on matrices, such as addition, subtraction, and multiplication.

Matrixes are used to represent many different types of data in computer graphics. Games with 2D or 3D graphics rely on matrix operations to render the game environment and characters. This paper discusses the process of linear algebra in computer graphics using examples from various areas of computer graphics.
Linear algebra's first application can be seen in the polygonal structure of 3D characters and surroundings in computer games and other 3D graphics applications. Because of their geometric properties, polygons are used to make images appear three-dimensional. Most of the time, this is accomplished by dividing the object into smaller and smaller geometric shapes, with the smallest rendered parts being triangles. The most basic application of polygons in 3D graphics is in the form of wireframe models of objects. A skeletal representation of a real-world object is a 3D wireframe. A cube, for example, can be represented as an object with eight vertices connected by lines or wires to create the illusion of three dimensions.
Animation is the name given to the second procedure in 3D graphics. This process defines the relationships between three-dimensional objects in three-dimensional space over time. This can be accomplished using a variety of techniques, such as key frames, inverse kinematics, and motion capture. Motion capture is the modeling of a 3D animation using sensors or cameras to capture the motion of an object or a person in the real world. When creating games or movies, kinematics is a useful tool for determining the precise positions of a joint system so that it can eventually reach a specific goal. This is done in movies to capture actors' facial expressions for use in animation to portray those expressions on animated characters.
3D rendering is the third process in 3D graphics. 3D rendering creates an animated scene or a 2D image from a 3D wireframe model of an object or multiple objects. This is accomplished by combining two operations. The first operation is transport, which refers to how much light is shone on the surface. being rendered, from what direction it is coming, and how intense the light source is. The second operation, scattering, includes animation as well as physics and weather animations, among other things.

A geometric transformation is a function that converts one point to another. Translation, rotation, and scaling are the most common transformations in computer graphics. Rotation and scaling can be represented in three dimensions by multiplying a 3-by-3 matrix by a 3-dimensional point. Unfortunately, translation cannot be represented in this manner, but there is a formulation that will be discussed further below that will allow us to capture all of the transformations we want to perform using matrix multiplication.

TRANSFORMATIONS

Computer graphics allow you to view an object from various angles. The architect can study the building from various perspectives, such as

1. Front Evaluation
2. Side elevation
3. Top plan

Charts and topographical maps can be resized by a cartographer. As a result, if graphic images are coded as numbers, the numbers can be stored in memory. Transformation is a mathematical operation that modifies these numbers.

The purpose of using computers for drawing is to allow the user to view the object from various angles, enlarging or reducing the scale or shape of the object, which is known as "transformation."

The following are two critical aspects of transformation:
Each transformation exists as a separate entity. It can be identified by a distinct name or symbol.
It is possible to connect two transformations to obtain a single transformation; for example, A is a translation transformation. Scaling is accomplished by the B transformation. C=AB is the product of two. As a result of the concatenation property, C is obtained.

There are two complementary perspectives on object transformation.

1. Geometric Transformation: The object is transformed with respect to the coordinate system or background. This viewpoint's mathematical statement is defined by geometric transformations applied to each point of the object.
2. Coordinate Transformation: The object is kept stationary while the coordinate system is shifted relative to it. This effect is achieved through the use of coordinate transformations.

An illustration of how these two points of view differ:

The movement of a car against a scenic backdrop This can be simulated by:

1. Moving the car while keeping the background static- (Geometric Transformation)
2. We can keep the car fixed while changing the scenery- (Coordinate Transformation)

VARIOUS TYPES OF TRANSFORMATIONS

Now, let's see the various types of transformations:

1. Translation
2. Reflection
3. Shear
4. Scaling
5. Rotation

Translation, rotation, and scaling are basic transformations.

TRANSLATION

Translation is the straight-line motion of an object from one point to another. In this case, the object is moved between two coordinate locations.

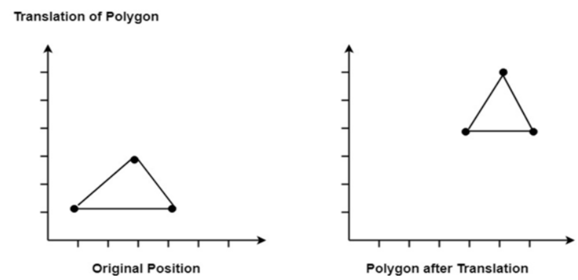
The translation distances Tx and Ty are added algebraically to the original coordinate to translate a point from the coordinate positions (x, y) to (x1, y1).

x1=x+Tx

y1=y+Ty

Here, Tx and Ty are known as shift vectors.

The percentage of translation is the same for whatever position or point. The straight line will be drawn using endpoints after being translated. Each vertex of the polygon is moved in order to translate it to a new place. Curved objects are translated in a similar manner. The center coordinates of a circle or ellipse are shifted to shift its position, and the object is then rendered using the new coordinates.



Matrix of translation:-

2D form:

Xnew= Xold   Tx

Ynew= Yold   Ty

3D form:

Xnew = 1	0	Tx		Xold
Ynew = 0	1	Ty		Yold
1 = 0	0	1		1

REFLECTION

It is a transformation that creates an object's mirror image. Either the x-axis or the y-axis can correspond to the mirror image. The object is rotated 180 degrees.

Types of reflection**:**

1. Reflection about the x-axis
2. Reflection about the y-axis
3. Reflection about an axis perpendicular to xy plane and passing through the origin
4. Reflection about line y=x

1. **Reflection about the x-axis:** The following matrix can be used to reflect the item about the x-axis.

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

The value of x will remain unchanged in this transformation, but the value of y will turn negative. The object axis reflection is depicted in the following figures: It will be on the other side of the x-axis.

2. **Reflection about the y-axis:** Using the following transformation matrix, the object can be reflected about the y-axis.

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

In this case, the values of x will be inverted, while the value of y will not change. It will be on the other side of the y axis.

3. **Reflection about an axis perpendicular to xy plane and passing through origin:** Below is the matrix for this transformation.

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

In this value of x and y both will be reversed. This is also called the "half revolution" about the origin.

4. Reflection about line y=x: The following transformation matrix can be used to reflect the object around the line y = x.

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

The object is first rotated by 45 degrees. The rotation is done in a clockwise manner. After that, the x-axis reflection is finished. The final stage is the rotation of y = x back to its original position, which is 45 degrees counterclockwise.

SHEAR

Transformation is what causes an object's shape to alter. Layers of an object slide off of one another. One direction or two directions may be sheared.

Shearing that occurs in the X direction involves the sliding of layers. Below is a diagram of the homogeneous matrix for shearing in the x-direction:

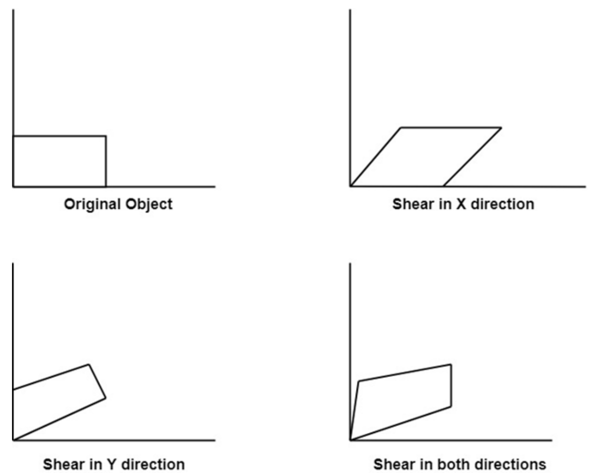
$$\begin{bmatrix} 1 & 0 & 0 \\ \mathrm{Sh}_x & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Shearing along the Y-axis: In this case, shearing is accomplished by sliding along the y-axis.

$$\begin{bmatrix} 1 & \mathrm{Sh}_y & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Shearing in the x-y direction involves sliding the layers in both the x and y directions. Both a horizontal and a vertical slide will be made. The object's shape will be altered. This formula yields the matrix of shear in both

$$\begin{bmatrix} 1 & \mathrm{Sh}_y & 0 \\ \mathrm{Sh}_x & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



SCALING

Scaling means resizing the object. It is done by using scaling factors that are Sx, Sy, and Sz for the x, y, and z directions, respectively. The scaling factor determines how the size of the object changes.

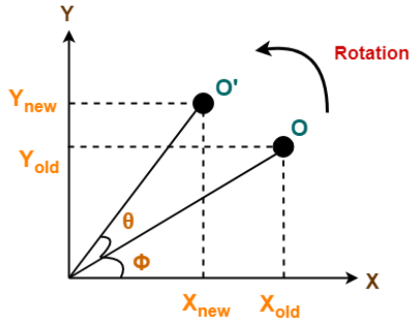
If scaling factor >1 size increases, if scaling factor <1 size decreases.

The 3D scaling matrix is given by:

$$\begin{pmatrix} sx & 0 & 0 & 0 \\ 0 & sy & 0 & 0 \\ 0 & 0 & sz & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} = \begin{bmatrix} x * sx \\ y * sy \\ z * sz \\ w \end{bmatrix}$$

ROTATION

Three-dimensional rotation is a challenging subject. Fortunately, we don't have to fully comprehend them in order to use them. We first think about rotation around the Z axis before making a generalization about rotation about any axis.



2D Rotation in Computer Graphics

Rotate Around Any Axis

Imagine that we wish to rotate around any axis.

The following matrix will rotate points around an axis in space if "u" is a unit vector that represents that axis:

$$R_u = \begin{bmatrix} cos\theta + u_x^2(1-cos\theta) & u_x u_y(1-cos\theta) - u_z sin\theta & u_x u_z(1-cos\theta) + u_y sin\theta \\ u_y u_x(1-cos\theta) + u_z sin\theta & cos\theta + u_y^2(1-cos\theta) & u_y u_z(1-cos\theta) - u_x sin\theta \\ u_z u_x(1-cos\theta) - u_y sin\theta & u_z u_y(1-cos\theta) + u_x sin\theta & cos\theta + u_z^2(1-cos\theta) \end{bmatrix}$$

$\boldsymbol{\theta}$ is measured in radians once more, and a positive angle causes the axis to rotate counterclockwise while it is oriented in your direction.

CONCLUSION

I have included information on linear algebra and a few of its applications in this project. A linear algebraic approach to computer graphics is provided. Animated films, motion pictures, and video games are only a few examples of major applications of computer graphics on a computer

How a matrix is used in graphics to convert geometric data into several coordinate systems In layman's terms, we state that a matrix's elements go through a transformation. Therefore, we attempt to comprehend how a matrix transformation is used to scale, translate, and rotate images on a computer screen.

We came to the conclusion that a large number of computer graphics applications utilize matrix multiplication in linear algebra.

REFERENCES

[1] Introduction to Linear Algebra -Fifth edition-Gilbert Strang

[2] Matrix and linear algebra in computer graphics-Kanti Bhushan Datta

[3]https://sites.math.washington.edu/~king/coursedir/m308a01/Projects/m308a01-pdf/yip.pdf

[4] Atkinson K.E ,An Introduction on Linear Algebra in Computer Graphics, Third Edition, Prentice hall India, New Delhi, 1988.

[5] W. Moore, "Linear Algebra for Graphics Programming," Metal by Example, 2014. [Online]. Available: https://metalbyexample.com/linear-algebra/

[6] https://metalbyexample.com/linear-algebra/

[7]https://gabrielgambetta.com/computer-graphics-from-scratch/A0-linear-algebra.html

[8] https://cs.colby.edu/maxwell/courses/guest/math16.htm

[9]https://www.cfm.brown.edu/people/dobrush/cs52/Mathematica/Part7/graphics.html

[10]https://eevibes.com/mathematics/linear-algebra/application-of-linear-algebra-in-computer-graphics/

[11]https://www.cs.bu.edu/fac/crovella/cs132-book/L13ComputerGraphics-Spring2021.html

[12]https://medium.com/@musadiq.iba/linear-algebra-in-computer-vision-9c5e0be73ae7