Transformaciones

domingo, 25 de octubre de 2020 07:55 p. m.

Vector: Mathematical object that has a direction and a magnitude.

Vectors can have any dimension.

When describing vectors we describe them as character symbols like $\bar{v}$. Also, when displaying vectors in formulas are displayed as follows:

$\bar{v} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$ Using vectors we describe directions and positions in 2D and 3D

Scalar operation:

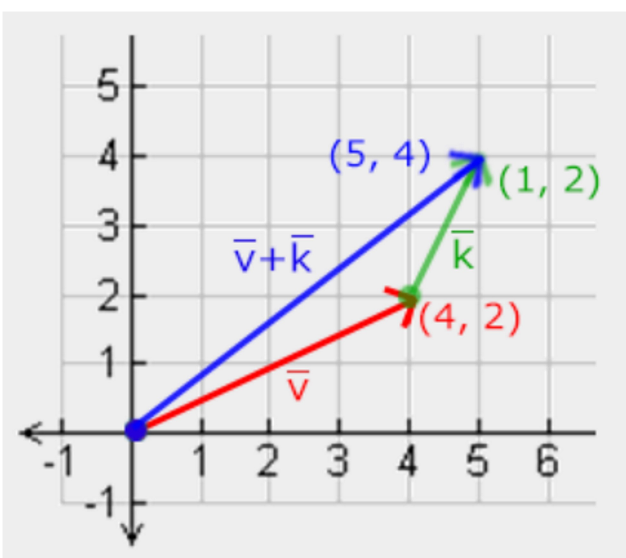
$$\begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} + x = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} + \begin{pmatrix} x \\ x \\ x \end{pmatrix} = \begin{pmatrix} 1+x \\ 2+x \\ 3+x \end{pmatrix}$$

Vector Negation:

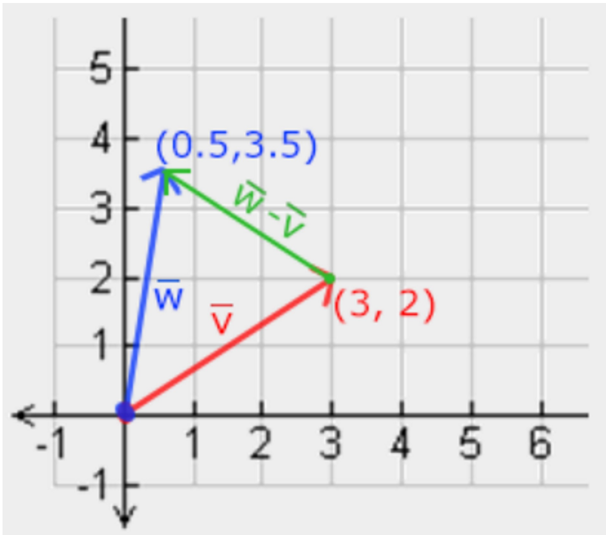
$$-\bar{v} = -\begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} = \begin{pmatrix} -v_x \\ -v_y \\ -v_z \end{pmatrix}$$

Addition and substraction

$$\bar{v} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}, \bar{k} = \begin{pmatrix} 4 \\ 5 \\ 6 \end{pmatrix} \rightarrow \bar{v} + \bar{k} = \begin{pmatrix} 1+4 \\ 2+5 \\ 3+6 \end{pmatrix} = \begin{pmatrix} 5 \\ 7 \\ 9 \end{pmatrix}$$



$$v = \begin{pmatrix}1\\2\\3\end{pmatrix},\ K = \begin{pmatrix}4\\5\\6\end{pmatrix} \to v - K = \begin{pmatrix}1-4\\2-5\\3-6\end{pmatrix} = \begin{pmatrix}-3\\-3\\-3\end{pmatrix}$$



length: $\|v\| = \sqrt{x^2+y^2}$

Unit vector: Length is 1. $\hat{n} = \frac{\tilde{v}}{\|v\|}$ has the same direction of the original vector. We call this normalize

Dot Product: Dot product of two vectors is equal to the scalar product of their lengths times cosine of the angle

$\tilde{v} \cdot \bar{K} = \|\tilde{v}\| \cdot \|K\| \cdot \cos\theta$

if $\tilde{v}$ and $\bar{K}$ are unit vectors: $\tilde{v} \cdot \bar{K} = 1 \cdot 1 \cdot \cos\theta = \cos\theta$. Now the dot product only defines the angle between two vectors

If dot Product $= 1$ is Ortogonal

if dot Product $= 0$ is parallel

$$\begin{pmatrix}0.6\\-0.8\\0\end{pmatrix} \cdot \begin{pmatrix}0\\1\\0\end{pmatrix} = (0.6 * 0) + (-0.8 * 1) + (0 * 0) = -0.8$$

to calculate the angle: $\cos^{-1}(-0.8) = 143.1^{0}$

Cross Product: 3D space takes two non-parallel vectors as input and produces a third vector to both the input vectors.

$$\begin{pmatrix}A_x\\ \ldots\end{pmatrix} \times \begin{pmatrix}B_x\\ \ldots\end{pmatrix} = \begin{pmatrix}A_y \cdot B_z - A_z \cdot B_y\\ \ldots\end{pmatrix}$$

$$\begin{pmatrix} A_x \\ A_y \\ A_z \end{pmatrix} \times \begin{pmatrix} B_x \\ B_y \\ B_z \end{pmatrix} = \begin{pmatrix} A_y \cdot B_z - A_z \cdot B_y \\ A_z \cdot B_x - A_x \cdot B_z \\ A_x \cdot B_y - A_y \cdot B_x \end{pmatrix}$$

Matrices: Is an rectangular array of numbers, symbols and/or mathematical expressions. Matrices are indexed by (i,J) where i is the row and J is the column.

Addition and Subtraction:

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} + \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = \begin{bmatrix} 1+5 & 2+6 \\ 3+7 & 4+8 \end{bmatrix} = \begin{bmatrix} 6 & 8 \\ 10 & 12 \end{bmatrix}$$

Matrix scalar products:

$$2 \cdot \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 2 \cdot 1 & 2 \cdot 2 \\ 3 \cdot 2 & 4 \cdot 2 \end{bmatrix} = \begin{bmatrix} 2 & 4 \\ 6 & 8 \end{bmatrix}$$

Matrix Multiplication:

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \cdot \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = \begin{bmatrix} 1 \cdot 5 + 2 \cdot 7 & 1 \cdot 6 + 2 \cdot 8 \\ 3 \cdot 5 + 4 \cdot 7 & 3 \cdot 6 + 4 \cdot 8 \end{bmatrix} = \begin{bmatrix} 19 & 22 \\ 43 & 50 \end{bmatrix}$$

Matrix-Vector Multiplication:

A vector is basically a $N \times 1$ matrix where N is the vector's number of component.

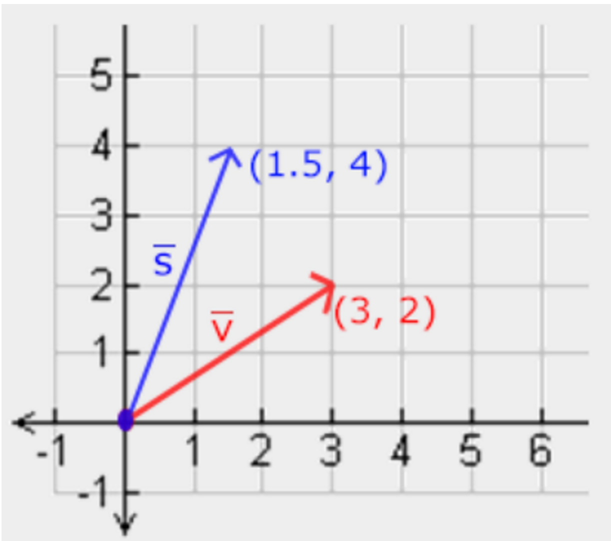
In OpenGL, we usually work 4×4 transformation matrices. One of the reasons is that most vectors are size 4.

Identity Matrix: Is a $N \times N$ matrix with only 0s except on its diagonal

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 1 \cdot 1 \\ 1 \cdot 2 \\ 1 \cdot 3 \\ 1 \cdot 4 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$$

Scaling: When we're scaling a vector we are increasing the length of the arrow by the amount we'd like to scale, keeping the direction the same. Since we're working in either 2 or 3 dimension we can define scaling by a vector of 2 or 3 scaling variables, each scaling one axis (x, y or z).

each Scaling one axis (x, y or z).
Let's try scaling $\bar{v}=(3,2)$, we will scale the vector along the x-axis by 0.5, thus making it twice as narrow; and we'll scale the vector by 2 along the y-axis. $\bar{S}=(0.5, 2)$



The scaling operation we just performed is a non-uniform scale because the scaling factor is not the same for each axis.
If the scalar would be equal on all axes it would be called a uniform scale

Lets start building a transformation matrix that does the scale for us. What if we change the 1s of the identity matrix to a variable S. If we represent the scaling variables as (S_1, S_2, S_3) we can define a scaling matrix of any vector (x, y, z) as:

$$\begin{bmatrix} S_1 & 0 & 0 & 0 \\ 0 & S_2 & 0 & 0 \\ 0 & 0 & S_3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} S_1 \cdot x \\ S_2 \cdot y \\ S_3 \cdot z \\ 1 \end{pmatrix}$$

The last element is the 'W' component

Translation: Is the process of adding another vector on top of the original vector to return a new vector with a different position, thus moving the vector based on a translation vector.
Just like the scaling matrix there are several locations on a 4-by-4 matrix that we can use to perform certain operations and for translation these are the 3 values of the 4th column. If we represent the translation vector as (T_x, T_y, T_z) we can define the translation matrix

$$\begin{bmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \end{bmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x + T_x \\ y + T_y \end{pmatrix}$$

this works because all the translation values are multiplied by the vector's w column and

$$\begin{bmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x + T_x \\ y + T_y \\ z + T_z \\ 1 \end{pmatrix}$$

This is because all the translation values are multiplied by the vector's w column and added to the vector's original values.

W component: homogeneous coordinate. To get the 3D vector from an homogeneous vector we divide the x, y, z coordinate by its w. w component allows us to do matrix translation on 3D

Rotation: Is represented with an angle. Rotating half a circle rotates us 360/2 = 180° and rotating 1/5th to the right means we rotate 360/5 = 72° to the right

Rotations in 3D are specified with an angle and a rotation axis. The angle specified will rotate the object along the rotation axis given. This is usually done via smart combination of sine and/or cosine

Rotation around the X-axis:

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

Rotation around the Y-axis:

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

Rotation around the Z-axis:

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

Combining Matrices: The true power of transformations in a single matrix is thanks to matrix-matrix multiplication. Say we have a vector $\begin{pmatrix} x \\ y \\ z \end{pmatrix}$ and we want to scale it by 2 and then translate by (1, 2, 3).

$$Trans . Scale = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 & 1 \\ 0 & 2 & 0 & 2 \\ 0 & 0 & 2 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Matrix multiplication is not commutative. It is advised to do the first the scaling operations, then rotations and lastly translations

$$\begin{bmatrix} 2 & 0 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} x \end{bmatrix} \quad \begin{bmatrix} 2x + 1 \end{bmatrix}$$

The vector is first scaled by two

$$\begin{bmatrix} 2 & 0 & 0 & 1 \\ 0 & 2 & 0 & 2 \\ 0 & 0 & 2 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} 2x + 1 \\ 2y + 2 \\ 2z + 3 \\ 1 \end{bmatrix}$$

The vector is first scaled by two and then translated by (1,2,3)