

A decorative graphic on the left side of the slide consisting of two overlapping parallelograms. The front one is blue and the back one is a light green color. They are positioned diagonally, with the blue one partially covering the green one.

Linear Transform Visualizer

MAT2240 Final Project by Avery



Related Class Topics

- Linear Transformations
 - In-Out Diagrams
 - Matrix Multiplication
 - Matrices
 - Vectors



Explanation of Connections

My project contains several examples of material we learned in class. My project is an in-out diagram for linear transformations. Linear transformations are achieved by multiplying an input matrix and a transformation matrix. The input matrix can contain a bunch of vectors which represent dots with lines drawn between, for example

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

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

Would represent a box with corners as (0,0), (0,1), (1,1), (1,0). These vectors can be multiplied by a transformation matrix, for example, to scale this box down to 50% you can use:

$$\begin{bmatrix} 0.5 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0.5 \end{bmatrix}$$



Why?

Linear Transformations have by far been the most interesting part of linear algebra for me. When we first covered linear transformations, I instantly had a real world example of this pop into my head. I do 3d printing, and sometimes a 3d model is slightly too big or small and needs to be adjusted. The program I use would use linear transformations to scale the model up or down, for example, if I need a model to be 90% original size, the program I am using would use this matrix to scale it.

$$\begin{bmatrix} 0.9 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0.9 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0.9 \end{bmatrix}$$



How

My project is made in HTML and JavaScript (JS). The HTML is used to format the page, but all the logic and the diagram is done in JS.

In the JS, I store the input diagram/shape as a 2d Array (a matrix). Each vector in the matrix has an x and y coordinate. These coordinates represent the location where a dot in screen will be drawn. The JS then draws lines between these dots and fills between the lines to create the shape.

The transformation matrix is also stored in the JS. To create the output diagram, I create a new matrix by taking every vector in the input matrix, multiplying it by the transformation matrix, and putting it in the output matrix. Then I use the same code to draw the output matrix.



Progress Downfalls

I did not get as far in this project as I had hoped. I have been feeling sick, which has taken a lot of energy out of me.

Some things I had hoped to add (and maybe I will later) were a way to enter a custom input matrix, a way to add multiple transformation matrices, and the ability use 3×3 transformation matrices for translations. Of course as well as improving the user interface and usability.



References

- Glossary 2 from AsULearn
 - Used for the Car shape
 - Author: Dr. Sarah
- ChatGPT
 - Minimal reference for learning JS (IE how to trigger a JS function from an HTML input)
 - Help fixing occasional errors I couldn't figure out
 - Author: ChatGPT/OpenAI