



Sabancı University

2024 – 2025 Fall Term

CS405 – Computer Graphics

Project 1 Report

Beyza Balota - 31232

Table of Contents

TASK 1:	2
TASK 2:	3
TASK 3:	5
CHATGPT LINK:	5

Task 1:

The first task's purpose was to make ChatGPT calculate some transformation operations which were given in the *transformation-prompt.txt*. The operations included some translation, rotation and scaling operations. As a result, GPT should calculate the final modelview matrix using JS. The transformation matrix that GPT calculated can be seen in the *Figure 1*.

```
function getChatGPTModelViewMatrix() {  
  const transformationMatrix = new Float32Array([  
    // you should paste the response of the chatGPT here:  
    0.1767767, -0.3061862, 0.3535534, 0.3,  
    0.3838835, 0.4330127, -0.25, -0.25,  
    -0.3061862, 0.25, 0.75, 0,  
    0, 0, 0, 1  
  ]);  
  return getTransposeMatrix(transformationMatrix);  
}
```

Figure 1: getChatGPTModelViewMatrix Function

After running index.html file in the live server, this transformation matrix outputted a 3D shape which looks like a trapezoid. (*Figure 2*)

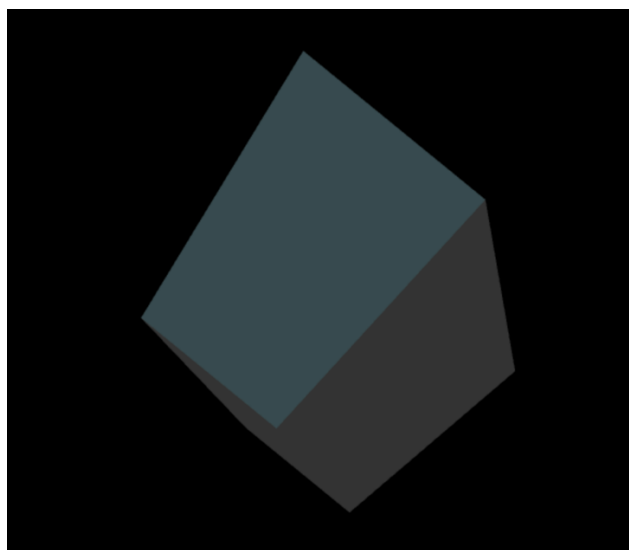


Figure 2: `modelViewMatrix = getChatGPTModelViewMatrix();`

Task 2:

The second task was more complicated than the first, which aimed generating the same transformation matrix by modifying the **getModelViewMatrix()** method, using the functions that were given in the `utils.js` file.

I first started with the translation function **createTranslationMatrix()**, for the 0.3 units in x-axis and -0.25 units in y-axis translation operations. After that, I continued with the scaling method **createScaleMatrix()**, for 0.5 unit scaling in both x and y-axes. Lastly, for the rotation by 30 degrees on x-axis, 45 degrees on y-axis, and 60 degrees on z-axis, I have used the methods **createRotationMatrix_X**, **createRotationMatrix_Y**, and **createRotationMatrix_Z**.

After using the transformation methods, the next step was to multiply the matrices in the right order to calculate `modelViewMatrix` correctly. For this part, I have checked the lecture slides and found out that the translation operations' order affects how the shape ultimately looks. After doing some research and checking my lecture notes, I found out that the recommended order was `rotation*scaling*translation` since rotation and scaling operations should be done in the origin. As a result, I first multiplied the rotation Y and rotation Z matrices, then multiplied the result with the rotation X matrix, which ended the rotation process. After that, I multiplied the result with the scaling matrix. In the end, similarly, I multiplied the last result with the translation matrix. So, the transformation process is completed.

After running the `index.html` with the new version of the **getModelViewMatrix()** method, the result is quite different than the one which ChatGPT calculated. See Figure 3 for the new image generated, and Figure 2 for comparison.

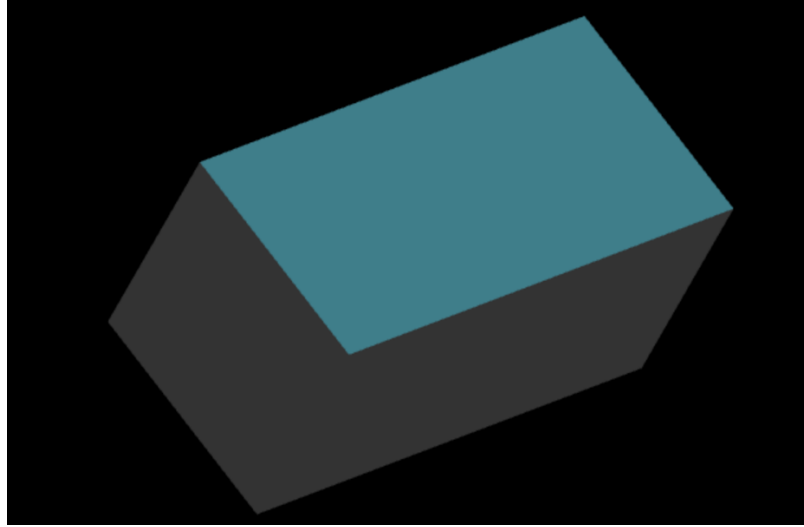


Figure 3: the image after the modification of the `getModelViewMatrix()`

The main difference between mine and the AI was the order of the transformation processes. Contrary to my research and the slides, this order of transformation was correct according to the ChatGPT. Translation at first, then rotation, and scaling in the end. (See *Figure 4*) As discussed before, the order affects everything since the rotation and scaling should be done in the origin in order not to spoil the object's structure. So, the cause of the difference between mine and the AI's depends on the order of the transformation performances.

$$M = T * R_z * R_y * R_x * S$$

Figure 4: ChatGPT's recommendation of transformation order

Task 3:

The last task was to ask ChatGPT to animate the cube object by using the transformation matrix that is found in the last task. I provided GPT with my last version of the **GetModelViewMatrix()** method and asked for proper animation, which will be 10 seconds. The first 5 seconds is the object's transition to the calculated transformation, and in the last 5 seconds, the object returns to its initial position. ChatGPT did these operations by modifying the **getPeriodicMovement()** method and did a great job.

ChatGPT Link:

<https://chatgpt.com/share/67114308-0068-800a-bcd0-4f2395baa832>