17.10.2024

CS 405 Project 1 Report

Kerem Tatari – 29208

Introduction

In this report, I will explain the steps I followed to complete CS 405 Project 1, which focuses on working with a 3D cube using WebGL. The project was divided into three main tasks: using ChatGPT to generate a transformation matrix, manually calculating the same matrix, and creating an animation that makes the cube move between two positions. Throughout the report, I will describe how I approached each task, what I learned, and the results I got from my work.

GitHub Repository Link: https://github.com/krmttr/CS-405-Projects.git

Pre-Tasks

Before starting the tasks, I made sure everything was set up properly for the project. I used the "Live Preview" add-on in VSCode to work with before-hand provided code. The first thing I did was set up the index.html, cube.js, and utils.js files, which are needed to display and modify the cube. These files had the basic code for showing the cube and placeholders where I would later add the transformations and animations.

Once I confirmed that WebGL was working with the "Live Preview" add-on and the code was displaying correctly, I took a screenshot -Figure 1- to show that everything was set up and ready.

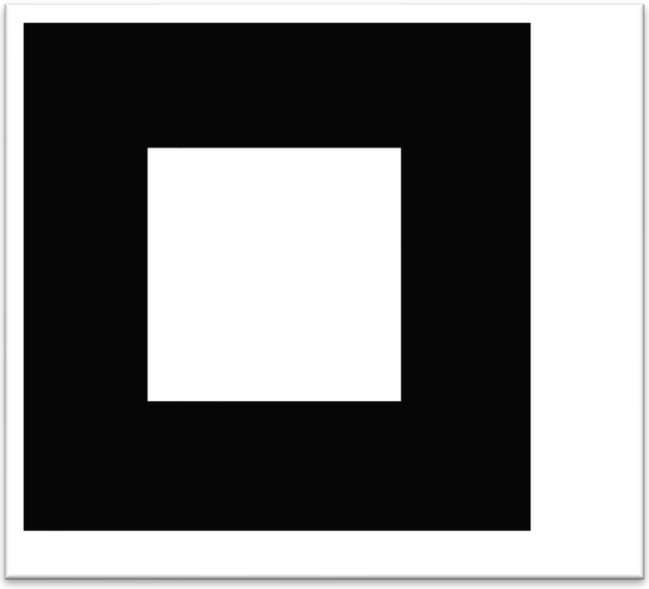

Figure 1: Pre-Tasks Screenshot

Task 1

For the Task 1, I had to use ChatGPT to calculate a ModelView matrix for a 3D cube. The specific transformations I needed to apply were provided in the transformation-prompt.txt file. These included translation, scaling, and rotation values that were to be used to generate the final matrix.

Steps I Took:

1. **Input to ChatGPT**: I pasted the text from the transformation-prompt.txt file into ChatGPT. This file described the transformations:
    - moving the cube by 0.3 units on the x-axis and -0.25 units on the y-axis (translation)
    - scaling it by 0.5 along the x and y axes
    - rotating it by 30° on the x-axis, 45° on the y-axis, and 60° on the z-axis.
2. **ChatGPT's Response**: ChatGPT responded with a Float32Array containing the calculated ModelView matrix.
3. **Implementation**: I took the generated matrix from ChatGPT and inserted it into the getChatGPTModelViewMatrix() function inside the utils.js file.
4. **Screenshot and Sharing**: After applying the transformation to the model (I think it should be cube, but it wasn't like ChatGPT provided), I rendered the updated model in my setup. I took a screenshot -Figure 2- of the transformed object -which is not cube since ChatGPT's output- and generated a shareable link to the ChatGPT conversation, as required by the task.

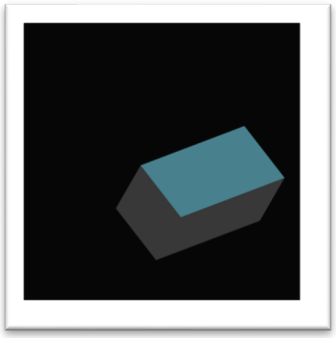

Figure 2: Screenshot after Task 1

ChatGPT Link is here: https://chatgpt.com/share/6710f30d-8148-800f-963c-e1085bab2de1

Task 2

For Task 2, I needed to manually calculate the ModelView matrix using predefined transformation functions for scaling, rotation, and translation. The goal was to recreate the same transformation as Task 1 but by coding it myself instead of using ChatGPT.

Steps I Took:

1. **Setting Up the Transformation Matrices**: I used functions from utils.js to create the necessary transformation matrices:
    - First, I made the scaleMatrix() to resize the cube by 0.5 along the x and y axes.
    - Then, I created rotation matrices for the x, y, and z axes with the provided angles (30° on the x-axis, 45° on the y-axis, and 60° on the z-axis).
    - Finally, I made a translation matrix to move the cube by 0.3 units along the x-axis and -0.25 units along the y-axis.
2. **Applying the Transformations**: I combined the transformations by multiplying the matrices in this specific order: **scaling first**, followed by **rotation on the x, y, and z axes**, and **translation** last. This is important because it ensures the transformations are applied correctly in 3D space. I applied the transformations in the following order:
    - First, I applied **translation**, which moves the cube to the new position.
    - Then, I applied the **rotation on the z-axis**, followed by the **rotation on the y-axis**, and finally, the **rotation on the x-axis**.
    - Lastly, I applied **scaling** to resize the cube along the x and y axes.
3. **Generating the Final Matrix**: After multiplying the matrices in the right order using the multiplyMatrices()function, I returned the final ModelView matrix.
4. **Comparison**: After calculating the matrix manually, I compared it with the matrix from Task 1 (calculated by ChatGPT). The results were identical, confirming that both matrices correctly transformed the cube. Unlike my previous assumption, the matrix ChatGPT provided in Task 1 was correct.
5. **Screenshot**: I took a screenshot -Figure 3- to show that the cube was transformed as expected, and both Task 1 and Task 2 produced the same result.

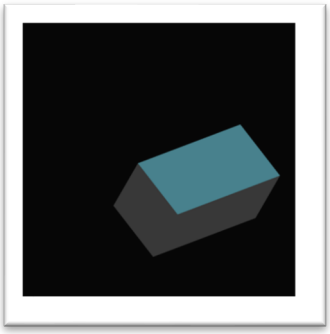

Figure 3: Screenshot after Task 2

Task 3

The last task involved creating an animation for the cube, where it moves between two positions in a 10-second cycle. For the first 5 seconds, the cube moves to its transformed position (from Task 2), and in the next 5 seconds, it moves back to its original position. This animation keeps repeating. I used ChatGPT as allowed.

Steps I Took:

1. **Modifying the getPeriodicMovement() Function**: I updated the getPeriodicMovement() function in utils.js. This function had to smoothly switch between the cube's initial and transformed positions using an interpolation method.
2. **Linear Interpolation**: I used linear interpolation (LERP) to calculate the movement between the two positions. The t value controls the animation, where:
    - During the first 5 seconds, t changes from 0 to 1, moving the cube to its transformed position.
    - During the next 5 seconds, t moves from 1 back to 0, returning the cube to its original position.
3. **Render Loop**: I set up a loop that keeps updating the ModelView matrix every frame, depending on the current time. This makes the cube move back and forth smoothly.
4. **Testing and Sharing**: After implementing the animation, I tested it to make sure the transitions were smooth and happened as expected. I also generated a shareable link to the ChatGPT conversation I used in this task, as required.

ChatGPT Link is here, I've used the same chat with the Task 1:
https://chatgpt.com/share/6710f30d-8148-800f-963c-e1085bab2de1

Conclusion

Throughout this project, I got hands-on experience with 3D transformations like translation, scaling, and rotation. Using ChatGPT in Task 1 to automatically calculate the matrix was really helpful and gave me something to compare with when I manually did the same thing in Task 2. Task 3 really helped me understand how to animate objects in 3D by making the cube move back and forth using interpolation. Overall, this project gave me a much better understanding of how transformations and animations work in WebGL, and it was a good mix of using both automated tools and manual coding.

GitHub Repository Link for the codes: https://github.com/krmttr/CS-405-Projects.git

Submitted by:
Kerem Tatari – 29208