

CS480 Intro

Jakub Rodzik, Alvin Lam, and Michael O'Toole
{Jakub.Rodzik001},{Michael.OToole001},{Alvin.Lam001}@umb.edu
University of Massachusetts Boston

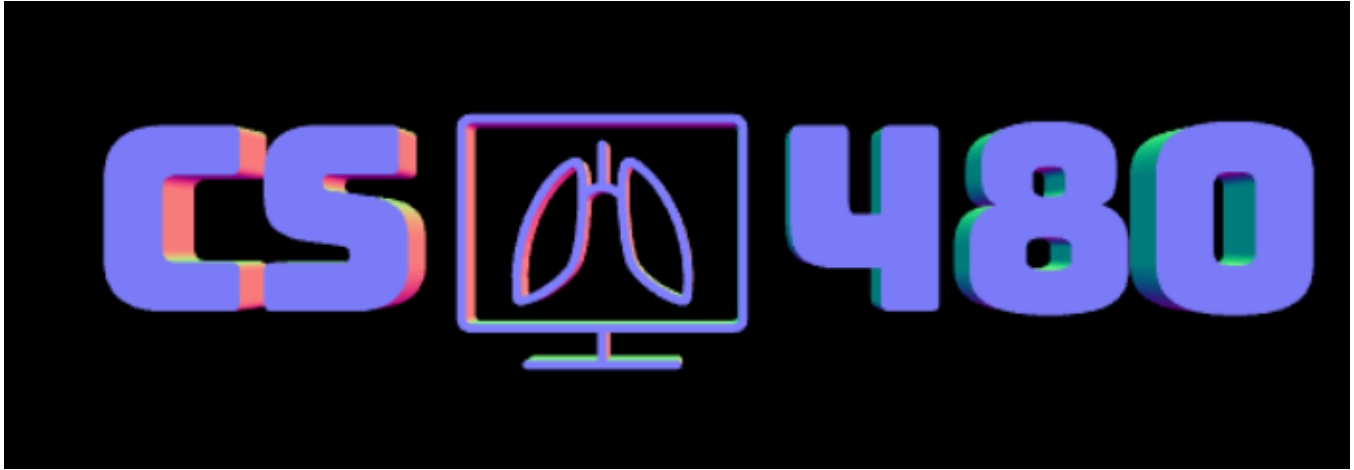


Figure 1: One of the CS 480 models used in the intro

ABSTRACT

For our final project we decided to create the introduction for Professor Haehn's new course, Biomedical Signal and Image Processing (CS480). Similar to the great intro we saw in CS460 each class, it is primarily a brief animated sequence that has some fast-paced, dramatic music aimed to hype up the students and get them excited for the topic, which a speaker describes with every different lesson.

KEYWORDS

Three.js, Tween, Imaging

ACM Reference Format:

Jakub Rodzik, Alvin Lam, and Michael O'Toole. 2020. CS480 Intro. In *CS460: Computer Graphics at UMass Boston, Fall 2020*. Boston, MA, USA, 3 pages. <https://CS460.org>

1 INTRODUCTION

This project is important because a quality introduction each class can really lift the spirits of the room and get students excited about the topic! Hearing dramatic music and cool imagery takes the class to another level in terms of attentiveness and the desire to learn will always be there after the intro. Some features for the CS480

intro include rendering of the CS480 logo and fast-paced music. Biomedical imagery and the UMass Boston logo is also prevalent.

2 RELATED WORK

Some of the related work we used during this project was Three.js [1] and Tween.js [2], a Javascript library to help with some of the animations in the intro.

3 METHOD

To create the introduction we broke it down into 3 main stages, being brainstorming, coding, and editing. The first step was to brainstorm ideas, where we essentially split the intro into different scenes, and thought up ideas for each scene, start to finish. Once we had a rough draft of what would be shown consecutively, we used video editing software to create a concrete example and tweaked it over the course of 10+ video drafts until we had a good rough draft video that fit our idea of what the intro should be like, and could be translatable into code.

With the draft video made and the ideas set up, the next step was to render the models, and begin coding the animation scenes one by one. The goal mainly was to get the baseline models of each scene set up first, then the movement, music, and image editing could be applied after.

Once the models were rendered for each scene and the chain link set up, the music, imagery, and general timing of the intro came next, and combined all the scenes into one great intro!

3.1 Implementation

As stated, the project uses Three.js to render the scene. A number of models were loaded in using OBJLoader and FontLoader. The loaders themselves were wrapped in Javascript promises to ensure

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

CS460, Fall 2020, Boston, MA

© 2020 Copyright held by the owner/author(s).

ACM ISBN 1337.

<https://CS460.org>

that each model was fully loaded before it was used in the animation. All models were initialized on page load and made invisible both for code-reading clarity and general performance.

The original UMass Boston logo and monitor icon source files were converted from SVG to OBJ using Tinkercad to provide an element of depth. Similarly, the Bungee font used for the CS480 logo was converted from TTF to typeface.json format.

We then constructed the `startAnimation` function, which primarily utilizes `Tween.js` to handle action-based timing, ensure smoothness, and access unique animation curves that would've been extremely tedious and error-prone if done manually.

The following code snippet renders the very first scene of the animation (see Figure 2):

```
(function umbLogoAnimation() {
  let start = { opacity: 1, spinSpeed: 0 };
  let end = { opacity: 0, spinSpeed: 1 };
  umbLogoTween = new TWEEN.Tween(start).to(end, 2000);
  umbLogo.then(mesh => { scene.add(mesh) });
  umbLogoTween.delay(1500);
  umbLogoTween.onUpdate(function() {
    umbLogo.then(mesh => {
      mesh.material.opacity = start.opacity;
      mesh.rotation.y += start.spinSpeed * 0.15;
    });
  });
  umbLogoTween.onComplete(function() {
    umbLogo.then(mesh => {
      mesh.visible = false;
    });
  });
})();
```

The rather different handling of meshes arises from the fact that they are Promise objects and thus cannot be accessed directly.

The library handles the stepping up and down of the numerical variables that act as modifiers of the scene. Over the length of 2000 ms, opacity and spinSpeed are accordingly changed from their initial values in `start` to match their targets set by `end`.

Given that the models are initialized to be invisible, it is necessary that any relevant meshes are made visible in beginning and then hidden once again afterwards. This is advisable, as permanently unloading and removing objects from the main `Three.js` scene is a costly operation.

Each following animation is chained to the previous one and structured exactly like this "tween." At the very end, `umbLogoTween.start()` is called to begin the intro.

3.2 Milestones

We structured the development in 3 main stages, involving the brainstorming of the project, the coding and rendering of the models, and the final editing process of music, images, and chaining them all together.

3.2.1 Milestone 1. The team brainstormed each section of the intro, part by part.

3.2.2 Milestone 2. After a rough draft, we then went through a process of making an video of it using editing software to see concretely what it would look like before coding.

3.2.3 Milestone 3. After the rough draft video was agreed upon and we had a solid idea on how it should look, we then began coding the models and renders.

3.2.4 Milestone 4. The next main milestone was to create the "spine" of the intro, or to make the base models that would be used in each section, with the editing and post-production to be done after.

3.2.5 Milestone 5. Finally, after all the models were rendered and the animation sequences strung together, the music was added and editing was done to make it flow smoothly one after another, like a chain of animations.

3.3 Challenges

Some challenges we faced included:

- Challenge 1: Finding and utilizing a library to help with animation, solved by Tween
- Challenge 2: Turning our video editing draft into code
- Challenge 3: Rendering all the models correctly, such as the UMB logo, the CS480 text, and the brain
- Challenge 4: Timing all the animations and chaining them together in a smooth fashion.

4 RESULTS

The final result came out great, and is mainly a group of scenes chained together into one intro. It starts with the UMB logo (Figure 2) rotating and fading out.

Next, the CS480 text drops down and is revealed, shown in figure 1. After showing the course number, some rapid images (Figure 3) are shown in tune with the dramatic music in the background.

Finally, the last intro is revealed, with many of the previous models and a rotating brain (Figure 4). This would be the end of the intro, and configurable text was added and an example audio file to show how the professor can add the topic of the class during the intro background.

5 CONCLUSIONS

This was a very interesting project that allowed us to utilize skills and topics that we had learned throughout the semester as well as gave us a chance to contribute to a future course taught by the professor. Being able to work as a team was very helpful as we acquired good team building experience by working together to finish a project by the deadline. There were many rough drafts and after putting it all together getting to see the result was very rewarding after all the hard effort we put into the project. The semester prepared us well for this project and we all agreed that this was one of the most organized and interesting courses we've had in CS, and with COVID making it hard to interact the soundboard and general upbeat tone of the class was very much needed during this tough time for everyone.



Figure 4: The final reveal at the end of the intro



Figure 2: The UMB logo in the beginning

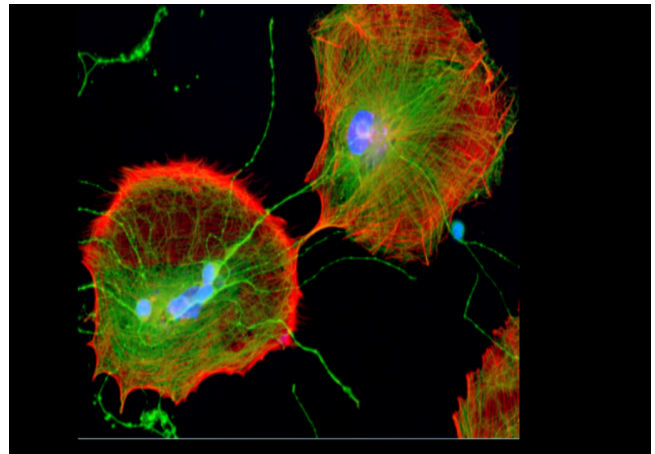


Figure 3: One of the images during the fast paced reveal

REFERENCES

- [1] Ricardo Cabello et al. 2010. Three.js. URL: <https://github.com/mrdoob/three.js> (2010).
- [2] Mike Casebolt Joe Pea et al. 2010. Tween.js. URL: <https://github.com/tweenjs/tween.js/> (2010).