Themes of Interaction and Interconnectivity

Our projects idea was conceived to incorporate to the overall topic of the course – interconnectivity and interaction. This is done through the two intensive codes and the new recent processing software release, accompanied with sound. When run, the first scene of our code represents interconnectivity. In this scene, the cubes represent an aspect of the universe. This aspect could be anything, such as an idea, status, animal, or person; in other words, the cubes represent both abstraction and tangibility. The lines which shoot from cube to cube represent the interaction of one entity upon other; these interactions upon an entity lead to a fate – a cause and effect. To clarify, our group conceived an example of how real life actions are symbolically represented in this scene. A mother is on her way to work, but is suddenly halted by her daughter, delaying her from getting to work. When she is finally able to head to work, she hits a man on the way. Had her daughter not delayed her mother from getting to work, the mother would not have hit the man. This interaction from daughter to mother, and mother to the man, shows how something so simple can change the fate of an entity without the entity which caused it even knowing. The motion of the cubes coming together and moving away also emphasizes the connectivity of the universe; while we are connected by the interaction we impose upon each other, we are also separate as entities alone.

The distortion of the webcam scene represents how we cannot see or predict the fate of entities interacting. This is because we only see through one point of view, which contrasts from the point of view of the all existing entities interacting (the cube scene). But since we are separate as entities, the cube scene's universal life cannot be seen be you or I. Therefore, our vision is distorted; using live video feed gave this symbolism personal depth, as well as interaction.

While this project idea is our submission, it was not the only idea my group and I came up with. Our initial idea was to create a music visualizer. However, we felt that it was the most basic and popular idea to code, and considering we were a group of four members, we had to go beyond expectations with this project. Our second idea was to create an abstract method of linking the smaller things in the universe with bigger things. This idea involved transitions or transformations from one object to another which was completely different in concept and scale. Going from an the oxygen atom in an H_2O molecule to a planet in a solar system is an example, and was one of our ideas. However, after seeing several groups presenting this idea, we decided to think outside the box when it came to using images. That is when we discovered that the new processing software allowed coders to take in live webcam video feed into a program. Because the release of the new software came only recently, other groups had yet to discovered it because they were still using the old processing from the previous projects. We used this discovery to our advantage, and created a distortion effect which tied well with our final concept.

However, we encountered technical problems with the final idea. One of the most prevalent problems was trying to fit our code under the 32kb limit. We had incorporated a great deal of libraries, which took most of our 32kb. We overcame this problem by omitting javadoc comments and any part of the effect that, while amazing, did not fit with our concept. This included making the video feed black and white, to be consistent with the black and white cubes

of the other animation. Another problem was combining the codes. At first, we were not sure how to make our code switch from the cube animation to the video feed. However, Danielle's ingenuity helped bring in our separate, working code to become one uniformed presentation.

Roles and Responsibilities

Rohan Karnik: Interacting cubes coder
    Coded the cube animation and randomized the animation and color change
Carlos Rios: Distorted webcam coder
    Coded the live video feed to be distorted in a gradient-like design and allowed interaction
Gabriel Arronte: Sound coder
    Coded three melodies and randomized them. The sound was coded to fit the rhythm of the animations
Danielle Arabov: Code compiler and debugger
    Combined our three codes together, randomized the visual scenes and timing, and debugged the compilation