C^3: COILS Clip Cut Visualization for Classroom Video Analysis

Major Qualifying Project

Written By:

Syreneti DeLaCruz
Sean Morrissey
Johvanni Perez

Advisor:

Lane Harrison

A Major Qualifying Project
Worcester Polytechnic Institute

Submitted to the Faculty of the Worcester Polytechnic Institute in partial fulfillment of the requirements for the Degree of Bachelor of Science in Computer Science.

August 25, 2021 - May 3, 2022

Abstract

Teacher training through data visualization to help improve classroom evironments has currently been an up and coming tool for teachers. The computer science and data science department at Worcester Polytechnic Institute (WPI) seek to improve the quality of teacher training through different data visualization techniques as we improve upon the current Classroom Observation Interactive Learning System (COILS) demonstration user interface (UI). We implemented a new data visualization technique of video-clips where the UI can filter out the top emotions represented in the videos shown in the demo through smaller video clips. The implementation of this new data visualization technique and our recommendations will ease teacher's experience through their training on the COILS platform.

Table of Contents

List of Figures

Figure 0.1: *Syreneti DeLaCruz - WPI 2022 Bachelor of Science in Computer Science*
Syreneti is from Long Island, New York and is a senior at WPI studying computer science. On campus, she is involved as a Resident Advisor (RA), National Society of Black Engineers (NSBE), and works as a student supervisor at the WPI George C. Gordon library.

Figure 0.2: *Sean Morrissey - WPI 2022 Bachelor of Science in Computer Science*
Sean is from Foster City, California and is a senior at WPI studying computer science. On campus, he is a member of the Massachusetts Delta Chapter of Sigma Alpha Epsilon, WPI esports, and works as a student supervisor at the WPI George C. Gordon Library.

Figure 0.3: *Johvanni Perez - WPI 2022 Bachelor of Science in Computer Science*
Johvanni is from Lawrence, Massachusetts and is a senior at WPI studying computer science. On campus, he is a member of the National Society of Black Engineers (NSBE), a mentee in the Board of Trustees Mentorship Program, and a student worker at the Academic Advising Office.

Executive Summary

The implementation of data visualization has brought a new and easier way to understand raw data. Currently, the Classroom Observation Interactive Learning System (COILS) has found a way to develop a system where teachers can go through demonstrations that provide data visualization techniques in classrooms and analyze students' emotions and interactions. The COILS project is a combination of machine learning, natural language processing, psychology, and educational theory to provide feedback and emotions seen in the classroom from teacher-student interactions. For our research project, we took on the challenge of creating a new user interface (UI) for the system and explored a new design of visualizing data.

0.1 Project Goals

The COILS demonstration focuses on implementing different data visualization techniques that are easy to understand so teachers can improve the classroom environment. In order to help create a new way to visualize the data on the current system, the use of video clips will be beneficial for teachers in order to understand what types of actions trigger positive emotions and what types of actions trigger negative emotions. Additionally, the clips can help identify different gaze interactions between different students.

Our overall goal for this project is to drive the use of clip cutting/sub-clips and a new design to visualize the data provided. The way that we can help achieve this main goal is by creating a new UI for the system that would be able to present this new design. In order to reach these project goals, we aim to achieve three main objectives:

1. Explore alternative recommendations based on our takes on the current user interface design
2. Manipulate/filter the data with specific information needed to pass in to the front-end
3. Implement the new visualized data to a more user-friendly interface

With these three main project objectives, we plan on discussing various designs within our team. We also plan on working with the raw data given from the current COILS demonstration and figure out a way to filter that data in a way where we can separate specific peak values which are the highest values in each of the emotions columns. With those extracted data values, we can then get time stamps to create clips. Once those clips are created, we will create a new design for the system and place clips as the form of visualizing what is happening. We hope that with the new UI, teachers will find it easier to navigate through the system.

0.2 Methods

By achieving our three objectives, we cultivated various data visualization techniques, different ways of working with raw data when it comes to coding it, and envision a new way the we can create a new UI design and explore unique software tools in order to create it. With the various data visualization techniques, we took a look at what the current methods were. Currently, there are emotional timelines and gaze interactions. The emotional timelines provide different layers of emotions and obtain the times and a curve that provides the emotion being shown at that time. The gaze interaction shows a dark gray and light gray bar that represents if they are being looked at by someone and if they are looking at someone else. After analyzing these visualization techniques, we then looked at the videos of teachers going through the system itself. While going through the demonstration, they gave continuous feedback on what they did and did not understand throughout the process. After reviewing these videos, we then decided that the best way to visualize important moments throughout the video in a way that teachers would understand what was happening will be with the use of video clips. In order to achieve those important moments, we had to filter out the data and select which would be the most useful.

When we manipulated and filtered out the data, we used Python and Python libraries to help create the video clips. We first created a smoothing algorithm where it would cancel out the "noise" from the raw data then pass it into a detect peaks function. In the detect peaks function, we figure out what is a peak value by first finding the average of all the values, and if they are higher than that average, then it is a peak value. Once we were able to figure out how to filter the peak values within each emotion, we then would extract information from the rest of the values in the raw data that correlate with that peak value in order to cut the entire video clip to one area. In order to create the smaller sub-clips, we used movie.py to access functions that will manipulate mp4 files. Once we are able to get our list of mp4 files that are the new sub-clips, we will then pass that on to the UI side.

In order to create a cleaner design that will work well with the newly created clips, we had a lot of different types of software to choose from to build the UI. The software that we used to create our UI was bootstrap. In order to use bootstrap, we used JavaScript to type up the code. We used bootstrap to create an HTML site where we pushed our changes on the repository and where people could access the prototype through a URL. We conducted a design-focused activity where we explored features that would benefit teachers in classroom video review. Some elements identified as potentially promising included a collapse sidebar, student profile button, a main section for the video, and filtering emotions and top three clips for each emotion for the student profiles. With a good understanding on what we would want for the UI and the sources we will be using for the project, we would be able to integrate the data end and the front-end.

0.3 Results

In the process of filtering out the important values of the data, we went into the already given raw data file and determined which values we will need to extract in order to make the video clips. We determined that we will create one singular JSON file that will contain all the objects needed for the front end. One object is considered each clip and the object contains values for emotion, time value/thumbnail, start and stop value, the naming convention, the ID of the profile, and the name of the directory of where the new mp4 is located. In order to create the start and stop values to create the clip, we had to add a few extra seconds to both values so all the clips will be about seven seconds each. It is good to have additional time added to both ends because that way, we are able to see what triggered that specific emotion and for how long they expressed it. Once we were able to extract all the necessary values, we put all those objects in arrays that we will then put in to create a CSV file. Once we created the CSV file, we then converted that file into a JSON file so it would be easier for the front-end to read the data. With the clips cut and the JSON file created, we were then able to start the integration between the new data and the UI.

The integration of the front-end and the data was a very smooth process due to the JSON file having all the values needed right away. The integration of the data and the front-end has made our UI become very interactive and not only with the user, but also between the different types of clips. The top portion of the prototype HTML page holds the video we were working on and the profiles of all the profiles of all of the students and teachers seen in the video. Once a user presses on a specific profile, the other faces then grays out to provide an emphasis on who's emotions we are focusing on. In addition, below the main video are the different emotions tabs and each tab has the top three time periods where the person expressed that specific emotion that the user clicked on. On the other hand we also implemented a way to display the list of various students and teachers and all of their peak times where they showed every single emotion or just one emotion and the user can filter it between all of the different emotions. Lastly, we also implemented the confidence level bar to show how accurate the machine learning AI is when it comes to detecting specific emotions. As we put the project together throughout the course of the year, we hope that the prototype can be useful for future ideas regarding the design of COILS.

0.4 Recommendations and Conclusion

From our collected results, we aspire future researchers that will work on COILS to incorporate various different ideas when it comes to adding on to this project or changing the current state of the project.

Our recommendations for the current system listed are:

1. The system can develop a more confident AI that detects emotions
2. While clicking on a specific profile, a new page can pop-up where there are multiple detailed visualization techniques that represent different things
3. Changing the survey questions that teachers need to respond to in order to get to the next page

When it comes to our project our recommendations listed are:

1. Creating clips for the gaze interactions
2. Showing a value next to the confidence bar with a percentage of how accurate the AI is
3. Producing the confidence level of the gaze interaction
4. Provide tools and tips tutorial for how to navigate out new UI

We aim for more teachers to continue using COILS in order to get a better understanding on how to provide a good environment for their students. In order to do so, the data visualization on the COILS system needs to be understandable for those who do not know that much when it comes to data visualization. We suggest video clips would be the best representation to visualize the peaks of when emotions are being expressed because it would help our users understand when and why they are expressing those emotions. Ideally, the new UI that we created for this project will help teachers have an easier time navigating through the COILS interface. With this type of visualization and training, it can provide understandable data visualization for anything and for anyone.

Chapter 1

Introduction

As teachers strive to implement a better classroom environment for students, they have undergone various training and programs that would help them provide equity of attention to all the students in the room rather than focus on one or a few students. At points there can be students that take up a lot of the teacher's undivided attention because they are very energetic or need the extra help but this can cause an unbalanced learning environment for other students. Having a balanced learning environment for all students leads to student success not only academically, but also socially. In order to make sure that they are providing that stable and positive learning environment, teachers will go through various training programs that will help them create a better classroom dynamic between all students and between students and teachers.

The COILS (Classroom Observation Interactive Learning System) project is a combination of machine learning, natural language processing, psychology, and educational theory to provide feedback and emotions seen in the classroom from teacher-student interactions (O'Conell, 2018). With this type of technology provided, teachers have the ability to understand classroom dynamics and how their teachings can create either a positive or negative emotion for students. Teachers are also able to analyze the reactions of students when something either good or bad happens, the interactions between students and other students, and overall the classroom experience and see if there is anything that needs to be improved. A COILS prototype has been developed through the use of artificial intelligence and data visualization techniques that will provide precise data that can be used for teachers to analyze in order to improve their teachings and interactions with students as seen in Figure 1.1.

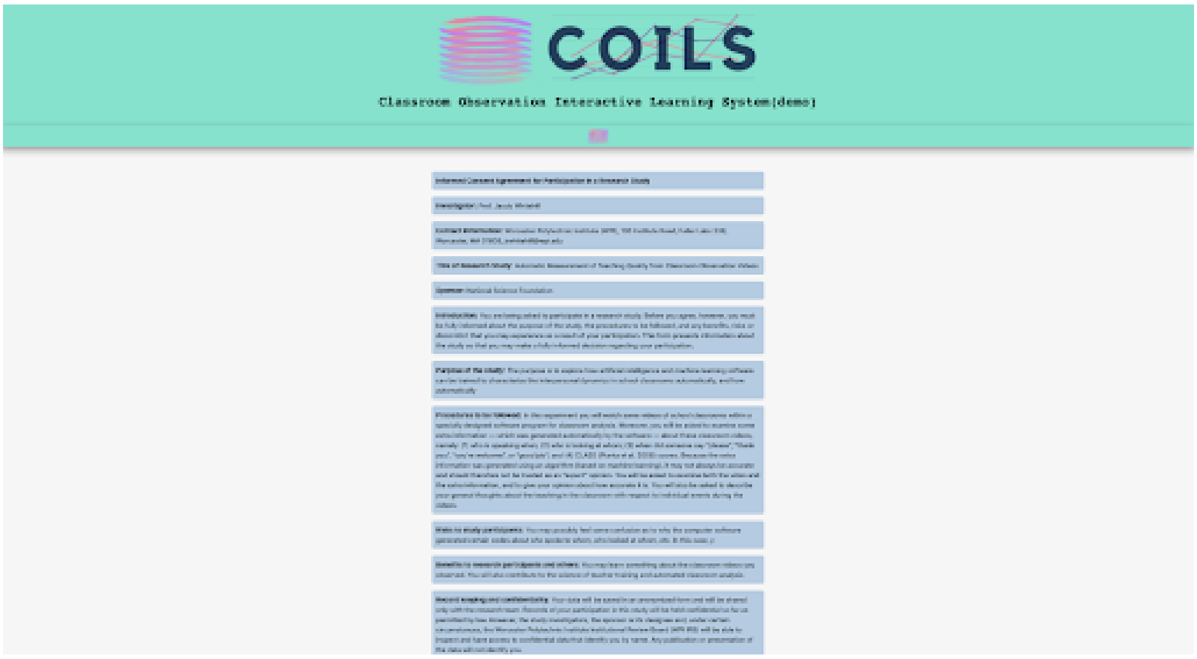
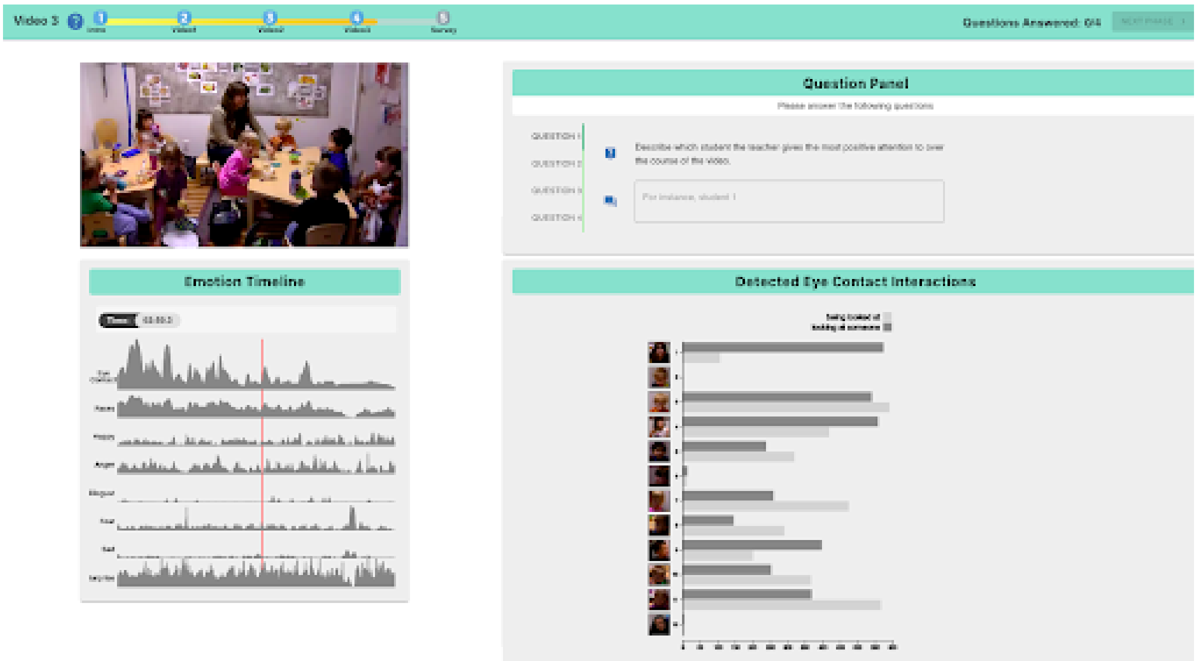


Figure 1.1: *COILS Main Page*
This is the main page of the current COILS main page demo interface. On the main page it contains aspects such as the research study and the investigator, the sponsor, purpose of the study, procedures of the study, risks and benefits for participants, confidentiality, and participant signature.

The two different parts to this teacher training is the ACORN and COILS projects. ACORN is known as the Automatic Classroom Observation Recognition Neural Network platform which recognizes teachers and students behaviors related to emotional support using machine learning. ACORN analyzes videos of school classrooms that process data of emotions regarding behaviors that are displayed on heat maps (Messier, 2018). COILS trains teachers to review classroom dynamics. As a team, we are focusing more on the COILS aspect where teachers will volunteer to participate in an interactive training development experience that takes the data shown by ACORN so they can analyze it. As COILS becomes the base of teacher training, there has become a problem where understanding the data is too complex and not as understanding and can be very time consuming. With this problem, our team has developed an improved and more user friendly COILS interface that will be helpful for teachers. A result of this prototype is that the complex visualizations such as the gaze charts and the emotions timeline can be somewhat intimidating and confusing for participants to interpret. Our goal is to make the visualizations of the clips to be clear and concise for participants going through this study. To do so, our overarching goal is to extract data and make the interface contain the top five clips of each person and their emotion. Through analyzing the demo, our team has the insight to explore a new possible design for COILS as a video-clip focused "visualization" tool to lower the barrier to use and exploration of key moments in videos. Our team has specifically focused on video three to test out the new interface of the video-clip visualization method. The current video for the interface is seen in Figure 1.2.



Please answer the following questions

Describe which student the teacher gives the most positive attention to over the course of the video.

For instance, student 1

Figure 1.2: *Initial Design*
This is the initial design for the video three page. The current page contains the video, the questions that the participants will answer, gaze interactions, and the emotions timeline.

In order to develop a new and improved user-friendly COILS interface, our goal is to create a more simplified experience for teachers to view students and their top emotions throughout the video training. With this, teachers will be able to assess why the student felt like that at the specific time and can assess how they can improve their teaching styles. With this new interface and simple way of detecting student emotions, we strive to help teachers implement a better classroom environment for students.

Chapter 2

Background

In this chapter, we explore the overall importance of data visualization and the impacts of teacher studies. Detecting human emotions in a classroom can be useful in understanding how emotions can affect students' ability to learn and teachers' ability to teach. ACORNS/COILS has performed studies, gathered data, and created a platform primarily for teachers to test for usability. Currently, the project has room for improvement in visualizing the data in a manner that users can intuitively understand and for effective usability. To improve on these aspects, teacher's preferences for the tool must first be explored and data visualization techniques must be researched.

2.1 Teacher Studies

As teachers try to study whether or not there is enough equity of attention in the classroom, they participate in various training in order to learn how to improve the classroom experience. Although some of the training that the teachers participate in contains them just watching videos of their interactions with the students, that can become very time-consuming and very tedious for them when they conduct emotional analysis [1]. In the classrooms, although it may seem unfortunate, there are biases when it comes to teachers and students. Whether it is implicit bias, racial bias, or gender bias, it can impact the student's learning [2]. In order to eliminate these biases, teachers are able to go through training programs to improve their teaching.

Alongside the WPI ACORN/COILS project, the University of Virginia Curry School of Education has also implemented various programs and training that will help teachers with social and emotional learning that will help improve the high school experience. In addition to our work on the ACORN/COILS project and the program put on by the University of Virginia, Harvard University also has a big training called The Teacher Education Program, also known as TEP, that helps teachers incorporate the relationship between the classroom and equity [3].

Currently, with the COILS project, teachers are finding it too complicated to understand the data visualized. When reviewing the different feedback from teachers, we are able to incorporate their ideas/suggestions to implement a new and improved user-friendly interactive system. We will help better reflect the raw data of the graphs and timeline to have it easily readable for the teachers. In order to understand the data, we have dug up research on data visualization and how it has helped analyze raw data and the techniques used with it.

2.2 Data Visualization Techniques

The use of data visualization helps by being able to shape data into an easier form to understand. Data visualization is defined as the graphical representation of information and data by using visual elements such as charts, graphs, and maps that provide an accessible way to see and understand trends, outliers, and patterns in data [4]. Throughout the years, data visualization has been increasing rapidly because it shows promising results of better data analysis for researchers and companies. Big data visualization can be practiced in two different directions: interactivity and real-time updating. Interactivity (interactive elements) is where data analysis tools contain interactive visualization modules where researchers can visualize a specific set of data points in various spatial organizations [5]. The second data visualization practice is real-time updating where new data is incorporated into data sets instantaneously through API access [5]. In order to build the charts, graphs, etc. there are various data visualization tools that can help researchers and companies analyze their data.

Data visualization tools can help convert data into interactive graphs, charts, maps, and more. Some of the main data visualization tools are Tableau, D3, Excel, Plotly, and more. Tableau is a data visualization tool that creates interactive graphs and charts in the form of dashboards and worksheets. This specific tool is used (but not limited to) for business purposes that helps companies gain business insight on their data. Excel is a data visualization tool because in the spreadsheet you are able to highlight the data that you wish to use and then produce various graphs and charts that you would like to visualize [3]. Another visualization tool is Plotly. Plotly is an open-source data visualization library to create interactive and publication-quality charts and graphs [6]. Plotly is used in various programming languages such as Python, R, MatLab, JavaScript, and more. With many different data visualizations tools offered, the current COILS project focuses on using the D3 visualization tool in order to help provide an interactive analysis of the data.

One of the most important data visualization techniques used throughout our project is D3. D3 is short for “Data-Driven Documents” and is a JavaScript library for visualizing data with HTML, SVG, and CSS [7]. D3 was developed by Mike Bostock, Jeff Heer, and Vadim Ogievetsky and was released in August 2011. As per Figure 3, D3 can visualize data in multiple different visuals ranging from bar charts, line charts, stream graphs, circle packing, bubble charts and more. The use of D3 provides many advantages for developers. First, because it is a JavaScript library, it can be used with any JavaScript framework such as React.js, Ember.js, or Angular.js. It is also an open-source library therefore you are able to to work with the source code and if needed, you can add features that tailor to your project. Lastly, it works with web standards therefore any additional plugins are needed and will work well with large datasets. As it provides many advantages, there are also limitations with D3. For example, it might contain some data-source limitations and cannot easily conceal original data. As D3 was significantly used throughout the project, as mentioned before it can be used with any JavaScript framework and the framework that was used for this project currently is React.js. React.js is an open-source JavaScript library that is used for building user interfaces [8]. React.js was originally developed by Jordan Walke and it launched in 2013 [8]. React.js is such a huge framework that applications and websites such as Facebook, Instagram, Khan Academy, and Imperial College London use it [9]. As these different data visualization tools help implement an easy understanding of data, by being able to interpret visualization analysis and design can help put the data in better perspective.

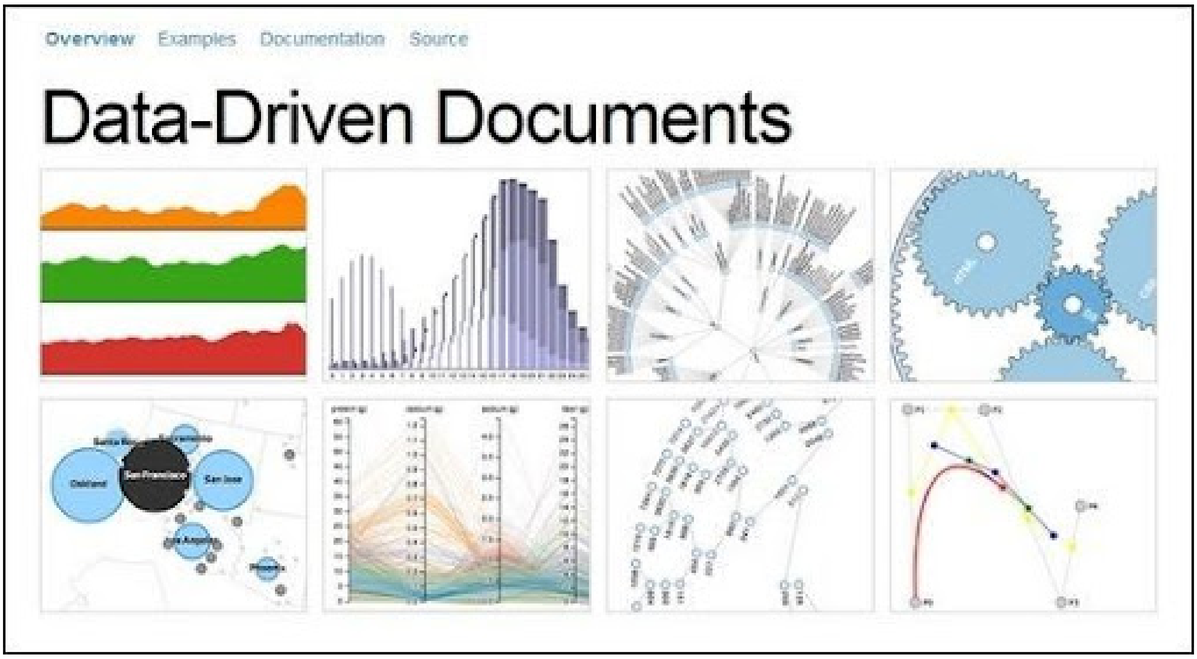


Figure 2.1: *D3 Data Visualization Examples*
These examples range from area charts, to line charts, etc.

2.3 Video-Clip Visualization Techniques

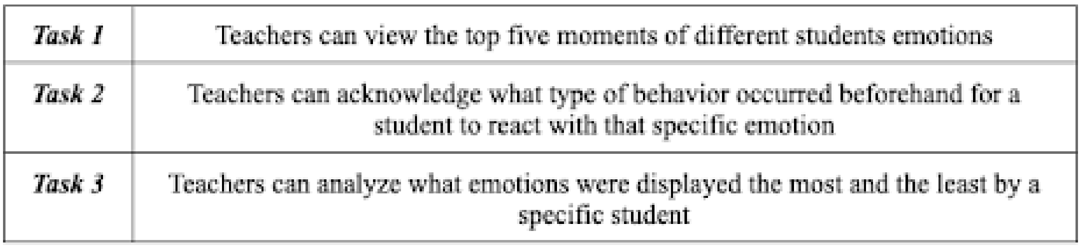
As data visualization techniques come with a variety of expressing given data, one of the most beneficial ways of expressing important content through a large set of data is through video-clip visualizations. This type of technique is what our team is focusing on for our project as we are extracting specific information from a large data file from a video then creating new videos from that extracted data set to present. Video visualization is a computation process that extracts meaningful information from original video data sets and conveys the extracted information to users in appropriate visual representations [10].

The video clip visualization has been a great method of data visualization because rather than interpreting various graphs and charts, participants can understand through a video specifically what they are looking at. When deciding to implement a video-clip visualization method, one important main question that researchers ask is how long should the video-clip be? When answering this question, most researchers recommend that the video-clips range from around ten to fifteen seconds of a clip [11]. The reason for this is that viewers need an initial settling phase as their attention span does not automatically begin at the beginning of the clip therefore the longer the clip the better. By creating new clips from videos that are over five minutes long, the small duration of a few seconds is favorable because it helps viewers get straight to the point and allow them to understand only the necessary important information rather than every single aspect of the video that may not be as important.

2.4 Visualization Analysis and Design

By being able to analyze the given designs, we are able to get a better understanding of the data and its impact on the projects. Data visualization analysis encompasses connecting low-level tasks with high-level tasks in a classification system to answer the questions “Why?”, “How?”, and “What?” the tasks are being performed for visualization. The description of tasks is an important aspect for visualization analysis because it leads to effective designs for problem driven design studies [12].

The use of multi-level typology of abstract visualization tasks is used to determine why the task is being performed, how the task is being performed, and what the task pertains to. The way that the typology supports complex visualization is because the sequence of tasks are used within each other as the output of one task is the input of the following task [12]. The first part of analyzing the question “Why?” in the scope of high-level tasks encompasses consuming and producing information. Visualizations first consume information in order to either present information or analyze new information. The produced information is what is generated as it transforms or derives data into visual effects. The second way to analyze visual designs is by asking the question of “How?”. By understanding the how, we are able to understand the methods used to visualize the data and that can be through manipulating or encoding data. An example of how we are implementing this into our project is that we are manipulating the given raw data by extracting specific points of where emotions are being detected by the AI and then cutting those points down and visualizing the top given points. Lastly, by answering the question of “What?” in the typology, we are able to relate the task to what it pertains to based on the outputs given. At the beginning of our implementation of the project, we created our list of top three tasks in order to help with our process to create a new visual representation of the data as seen in Figure 2.2.



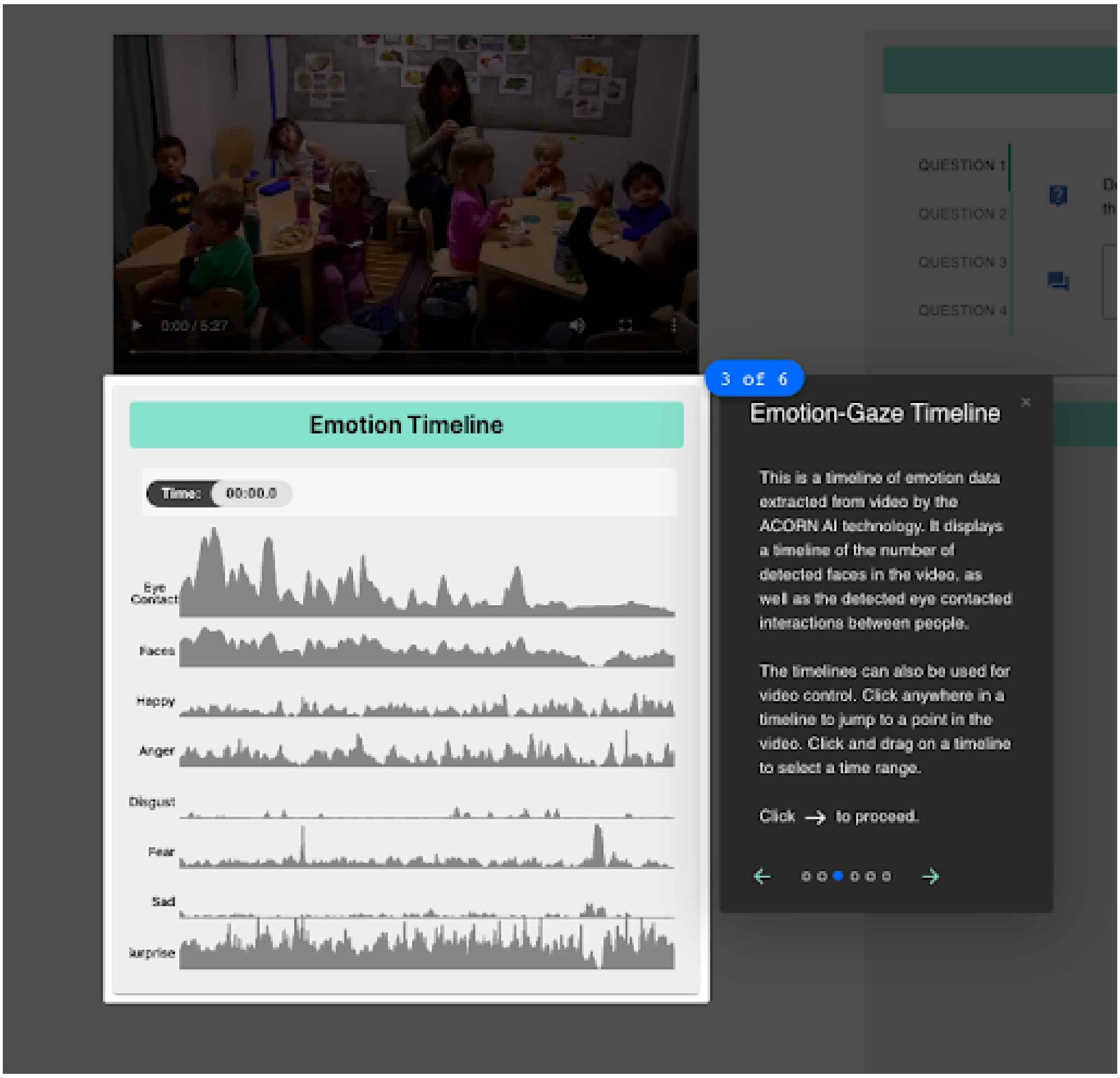
Task 1	Teachers can view the top five moments of different students emotions
Task 2	Teachers can acknowledge what type of behavior occurred beforehand for a student to react with that specific emotion
Task 3	Teachers can analyze what emotions were displayed the most and the least by a specific student

Figure 2.2: *Project's Visualization Analysis and Design Tasks*
These tasks are what we hope teachers can achieve by going through the new interface that we visualize. All these tasks helped us with understanding how to visualize our project.

2.5 Current ACORN/COILS Project

The ACORN platform and COILS training has been developed in tandem to help train teachers to pick up on classroom dynamics [13]. As teachers go through this training, they are able to identify which students they interact with the most, the reactions of students when something either good or bad happens, the interactions between students and other students, and overall the classroom experience and see if there is anything that needs to be improved.

A team of graduate students has worked on designing the project and carrying out experiments to collect data necessary for the purpose of this project. The analytical program that analyzes facial emotions from mock classroom experiments has been used to collect data. Alongside the raw data that classifies emotions detected from the mock experiment, think-out-loud sessions have been recorded. Essentially, think-out-loud sessions are ones with users that utilize a platform containing videos of the classroom experiments and a dashboard containing visualizations of different facial and emotional cues. These sessions were designed to have the user think as they watch the videos and interact. Based on the user's statements and user experience, information can be gathered and be considered for potential ideas to improve the usability of the platform and the effectiveness of the data visualizations. Currently, ACORNS/COILS is collecting data through various data visualizations tools that are then used by artificial intelligence. Much of these visualizations are constructed by the D3 JavaScript Library. Across the website's pages, there are timeline graphs, bar graphs, and arc graphs. The timeline graphs are used to show the change in emotions throughout the given videos. We use this so teachers are able to visualize moments of emotional change so that they are able to investigate what went on by going to that part in the video. The bar graph is used to display the eye contact detected by the software. With this, teachers may be able to deepen their understanding of the student-to-student and teacher-to-student interactions. The arc graphs are used to visualize these changes in eye contact throughout the video in a live fashion. As a teacher scrolls through the video, they are able to see which interactions were detected. For example, if the teacher looked at student one or two when anger occurred. By being able to visualize these interactions, D3 and React.js were used.



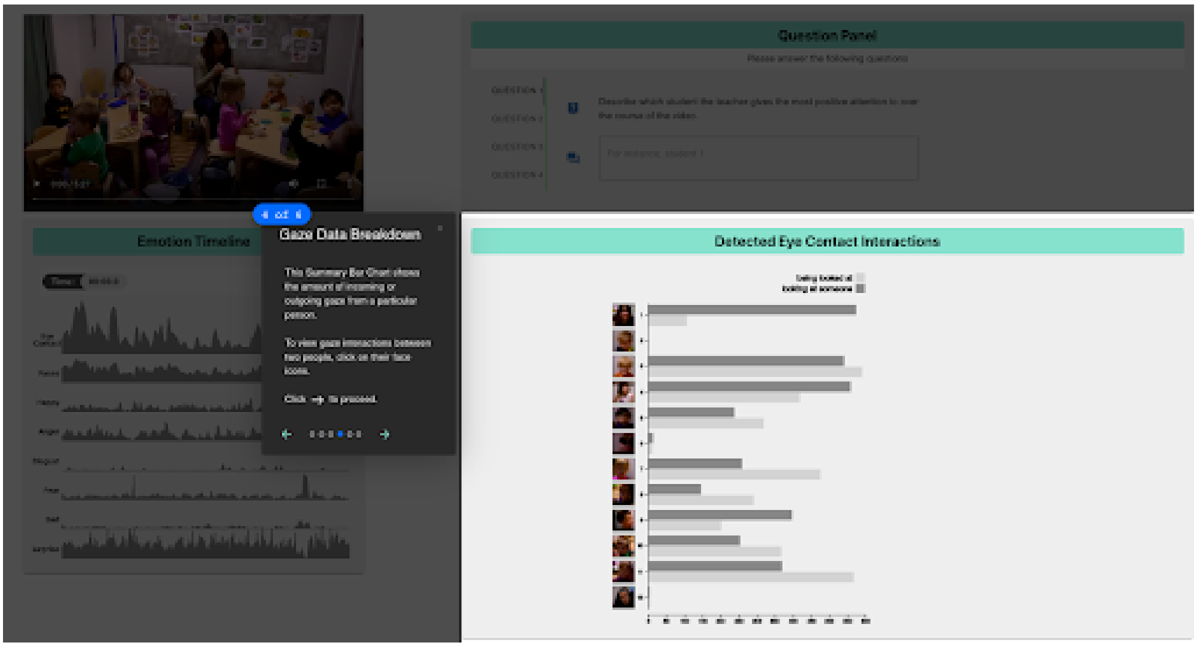
Emotion-Gaze Timeline

This is a timeline of emotion data extracted from video by the ACORN AI technology. It displays a timeline of the number of detected faces in the video, as well as the detected eye contacted interactions between people.

The timelines can also be used for video control. Click anywhere in a timeline to jump to a point in the video. Click and drag on a timeline to select a time range.

Click → to proceed.

Figure 2.3: *Visualization of Emotions*
As users drag their mouse across the timeline, it correlates to the time given in the video. As you can see through some of these peaks/humps in the timeline, at specific moments there are emotions that occur a lot and can be seen clearly through the video. In addition, before going through the questions and the video, the demo contains a very useful tutorial to try to help participants understand each feature.



Gaze Data Breakdown

This Summary Bar Chart shows the amount of incoming or outgoing gaze from a particular person.

To view gaze interactions between two people, click on their face icons.

Click → to proceed.

Figure 2.4: *Visualization of Eye-Gaze Interactions*
As users drag from the emotion timeline, the eye contact/gaze interactions between different people can be analyzed. The lighter gray shows how often that person is being looked at and the darker gray states how often that person is looking at someone else. Like in the previous figure, before going through the questions and the video, the demo contains a very useful tutorial to try to help participants understand this feature.

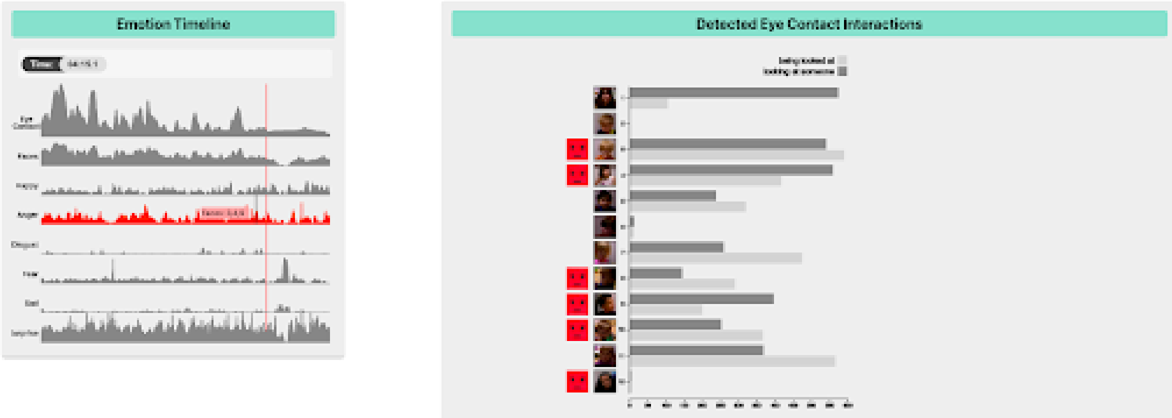
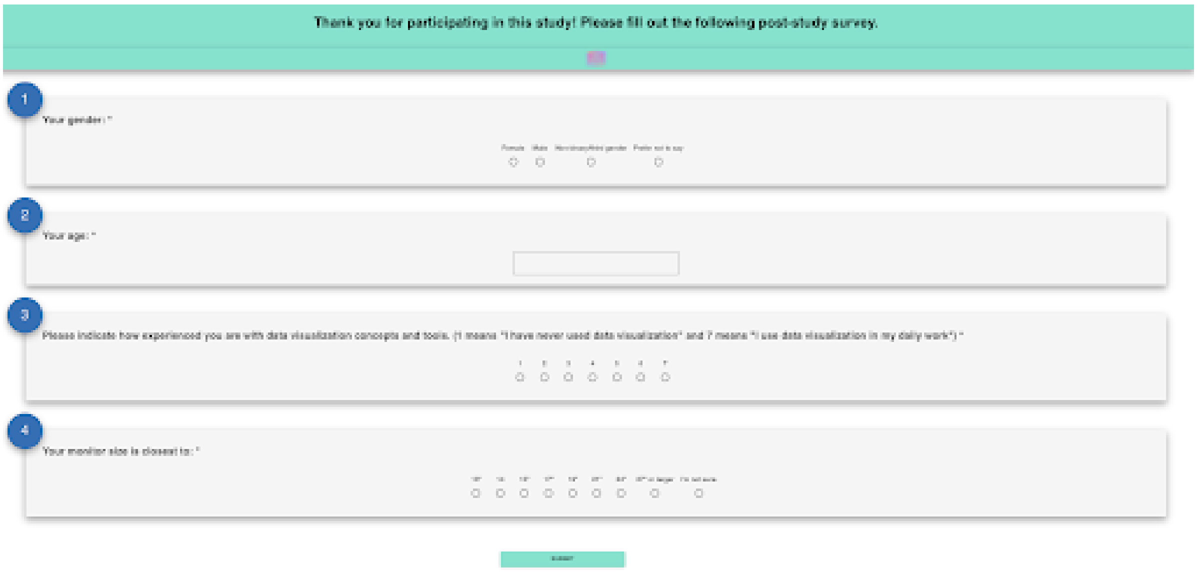


Figure 2.5: *Visualizations Interactions*
This figure shows how the two different visualization designs interact with each other. As seen in this example, state that you are looking at the anger emotion at time 4 minutes and 15 seconds, it shows the face numbers on the timeline and the eye contact interaction charts shows specifically between who.

As this demo has been distributed to teachers around the Worcester area, the participants had the ability to fill out a survey as seen in Figure 2.6. After the survey, they also had feedback on how to improve the demonstration and through this the ideas of implementing smaller clips was brought up. After thinking about the clips idea, April, a student who has worked on this project prior to our team's contribution, created some beginning phase sketches on how clips can be integrated into this project as seen in Figure 2.7, Figure 2.8, and Figure 2.9.



Thank you for participating in this study! Please fill out the following post-study survey.

1 Your gender: *

Female ○ Male ○ Non-binary/third gender ○ Prefer not to say ○

2 Your age: *

3 Please indicate how experienced you are with data visualization concepts and tools. (1 means "I have never used data visualization" and 7 means "I use data visualization in my daily work") *

1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7 ○

4 Your monitor size is closest to: *

[illegible] ○ ○ ○ ○ ○ ○ ○ ○ ○

Submit

Figure 2.6: *Current Survey*
This is the survey page at the end of the demo study where participants fill out basic information such as gender, age, experience with data visualization, and their monitor size.

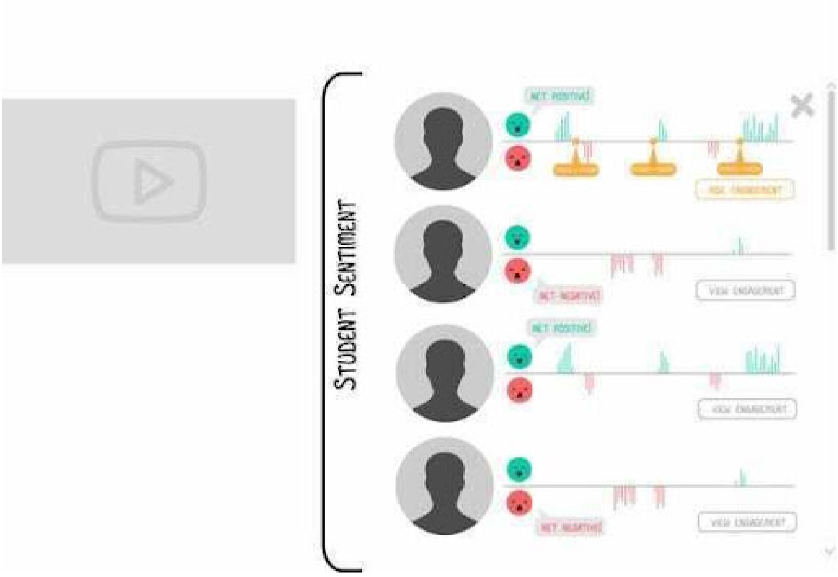


Figure 2.9: *Initial Eye-Gaze Design*
This initial prototype design done before was done in relation to student gazes within each other. For example, it would produce a line/arrow between the two students who are having that interaction.

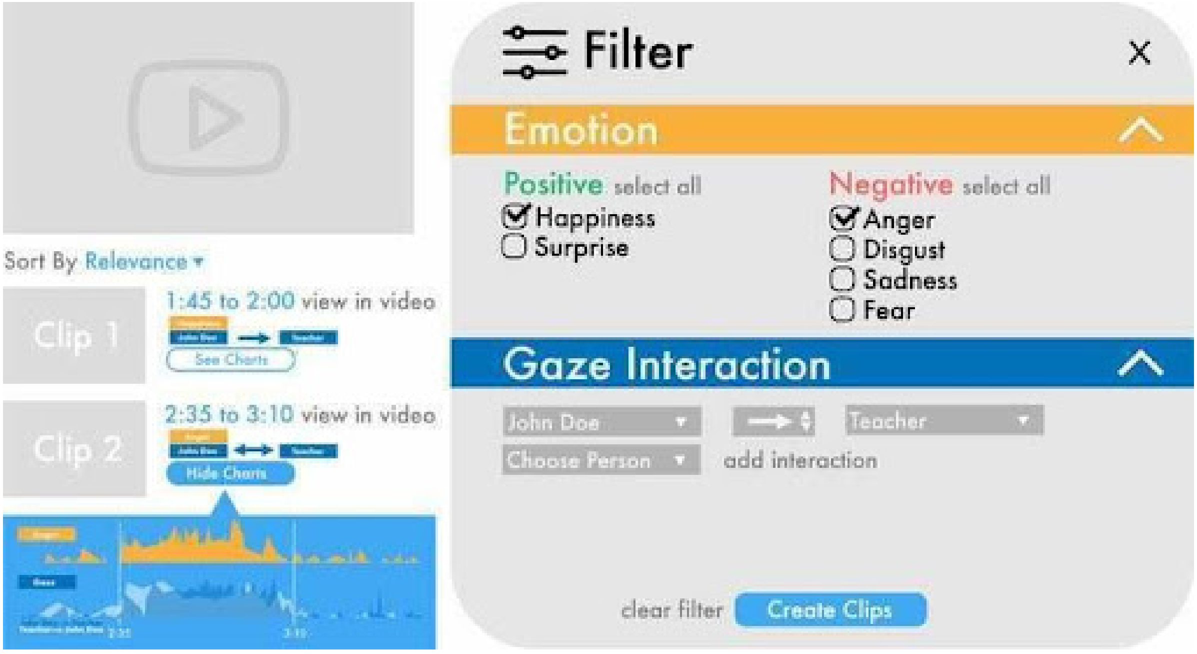


Figure 2.7: *Initial Clip Design*
This first initial prototype of the project shows the main video on one end and then the visualization on the other side. Here there is separation of positive and negative emotions and then whichever is clicked, the clip of those emotions pop up under the main video.

2.6 Impacted Stakeholders

As we move along this project, we think of new ideas on how the project can benefit our stakeholders. Our stakeholders are teachers from grades K-12 that go through the given COILS training. The project will impact teachers in many ways but the overall goal is to improve their interactions in the classroom. We hope that with the project, teachers can improve their teachings by picking up on classroom dynamics and how students react to teachers' cues. The ability to visualize students' emotions through a variety of clips can help teachers obtain a broader sense of student behaviors. The new UI by letting teachers see snippets of clips would show teachers the actions beforehand and what can trigger specific emotions within their students and how they react later.

2.7 Conclusion

Through the COILS project, data visualization techniques have been a critical role for displaying important moments throughout the videos. Transitioning from the current visualization

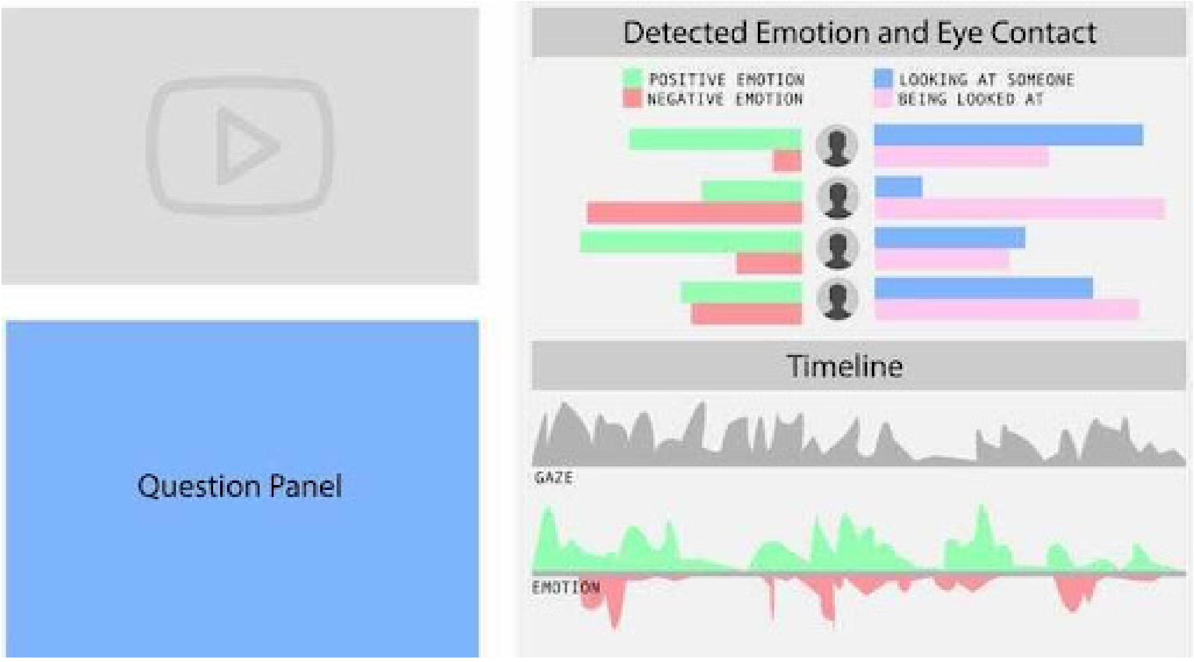


Figure 2.8: *Initial Project Interactions*
This image of the gaze and emotion timeline shows all of the emotions happening with a student as well as who they are looking at as well if they are being looked at by someone else. In addition there is the question panel to the side where teachers input their answers to the questions given or comments.

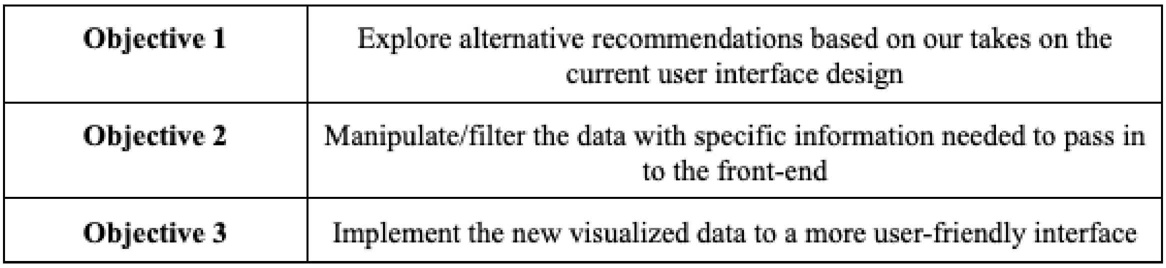
of timelines and graphs, the use of video-clip visualizations can be a great benefit for participants to understand the project overall. With the use of our video-clip visualization method, we are looking to produce the top five moments per emotion for each person seen in the video. By being able to extract those top five moments through the data file and convert them into videos, we are able to improve the current COILS project.

Chapter 3

Methodology

The goal of our project is to redesign the user interface of the Classroom Observation Interactive Learning System (COILS). The team has taken on this project in order to help WPI researchers to explore how artificial intelligence and machine learning software can successfully characterize interpersonal dynamics in school classrooms automatically and how. This project will help teachers analyze how their teaching styles impact their student's emotions and their interactions with their students alongside how students react with one another.

In order to carefully plan how to redesign the interactive demonstration, our team will accomplish three main objectives:



Objective 1	Explore alternative recommendations based on our takes on the current user interface design
Objective 2	Manipulate/filter the data with specific information needed to pass in to the front-end
Objective 3	Implement the new visualized data to a more user-friendly interface

After accomplishing these four main objectives, we further recommend different alternatives that can be reviewed for further research.

3.1 Objectives

We each individually decided to brainstorm our own ideas on how to improve the current user interface of the interactive learning system. By brainstorming our own ideas, it gave each of us the liberty to creatively interpret different visions of how the interface has the possibility of looking like. We analyzed the videos of teachers interacting with the current learning system in order to help advance our designs. With the help of watching these videos, we were able to understand the perspectives of teachers and what they hope to gain with the interactive system. For example, they hope to see what they can do differently in the classroom after reviewing the emotional timelines and how often they interact with specific students.

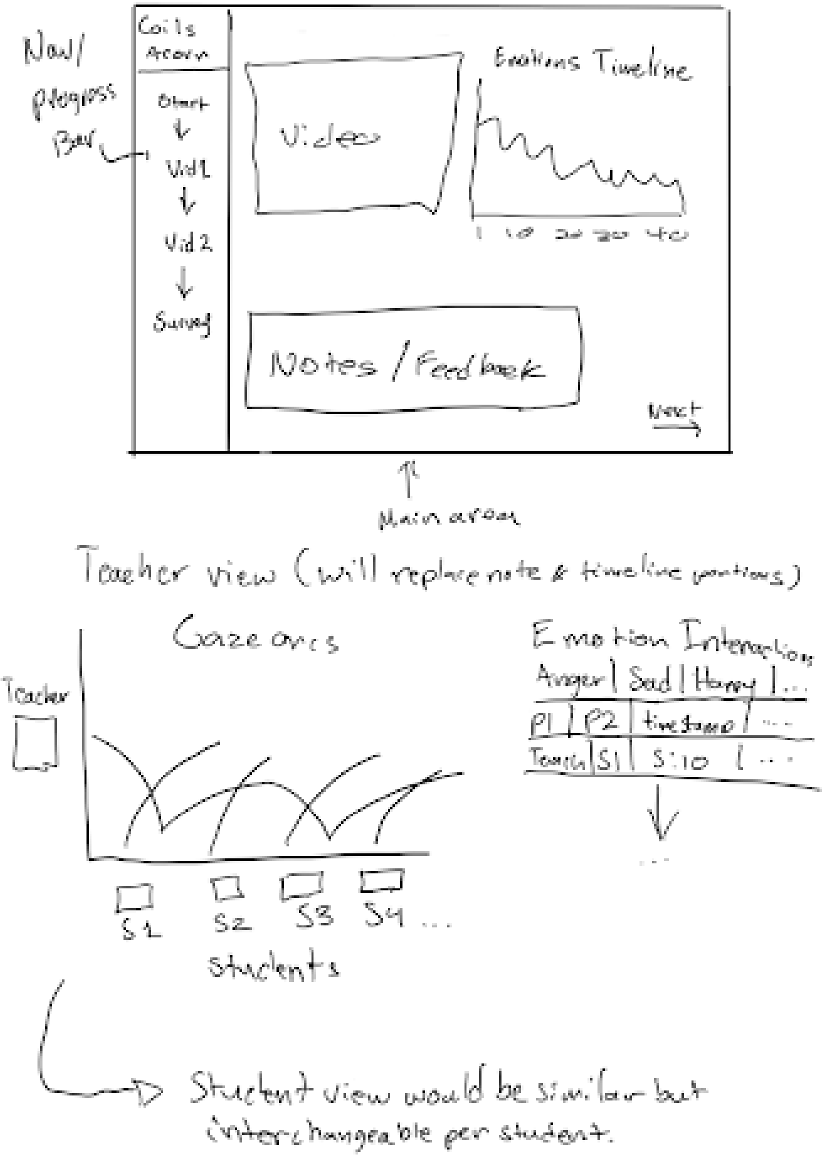
In order to go through with our design, we needed to filter out specific information in relation to finding the peak of our emotion values. Some of the included information are time values, start and end times, emotion name, and faceID. With all this information for every person ID and each of the emotions, we will then integrate that with the new UI.

By reviewing their interaction with the interface and taking all of our individual ideas put together, we aim to create a newly updated interface of the classroom observation learning system so it can be more user friendly for teachers. This new user interface will encompass next level usability for users. It will in turn provide features that will help teachers better understand the data visualization and easily navigate through the system.

3.2 User Interface Brainstorm Session

In order to have an idea of what we wanted our user interface (UI) to look like, we did various brainstorming sessions individually, within each other, and with our advisor. We analyzed what the current interface entailed and saw what we would want to keep and what we can change. Our brainstorm first started off through hand drawn ideas that can be seen in Figure 3.1 and Figure 3.2.

Inspiration for our brainstorming sessions were based on looking at the student views of the current COILS project that can be seen in Figure 14. We looked through how the current student view accessed all of the current students faceID value to make sure that the participant had an image to match the ID. After talking about these designs, we had another brainstorm session where we decided to change up our ideas and wanted to simplify the design. We started off by having the video and then also face numbers (FaceID) on the side so the user can click on a specific person. When they click the face number, then there would be a timeline at the bottom that shows the various emotions stacked on top of each other and then the top three clips from that specific emotion will be displayed. That clip will be a gif that shows about ten seconds of the student expressing the emotion. This type of display can be seen in Figure 3.3 and Figure 3.4.



Anger	Sad	Happy	...
P1	P2	timestamp	...
Teach	S1	5:10	...

Figure 3.1: *Sean's Initial Design*
Sean's initial ideas drawing. This idea encompasses a detailed description of all the values at a given time. This design shows that there can be interchangeable visualizations.

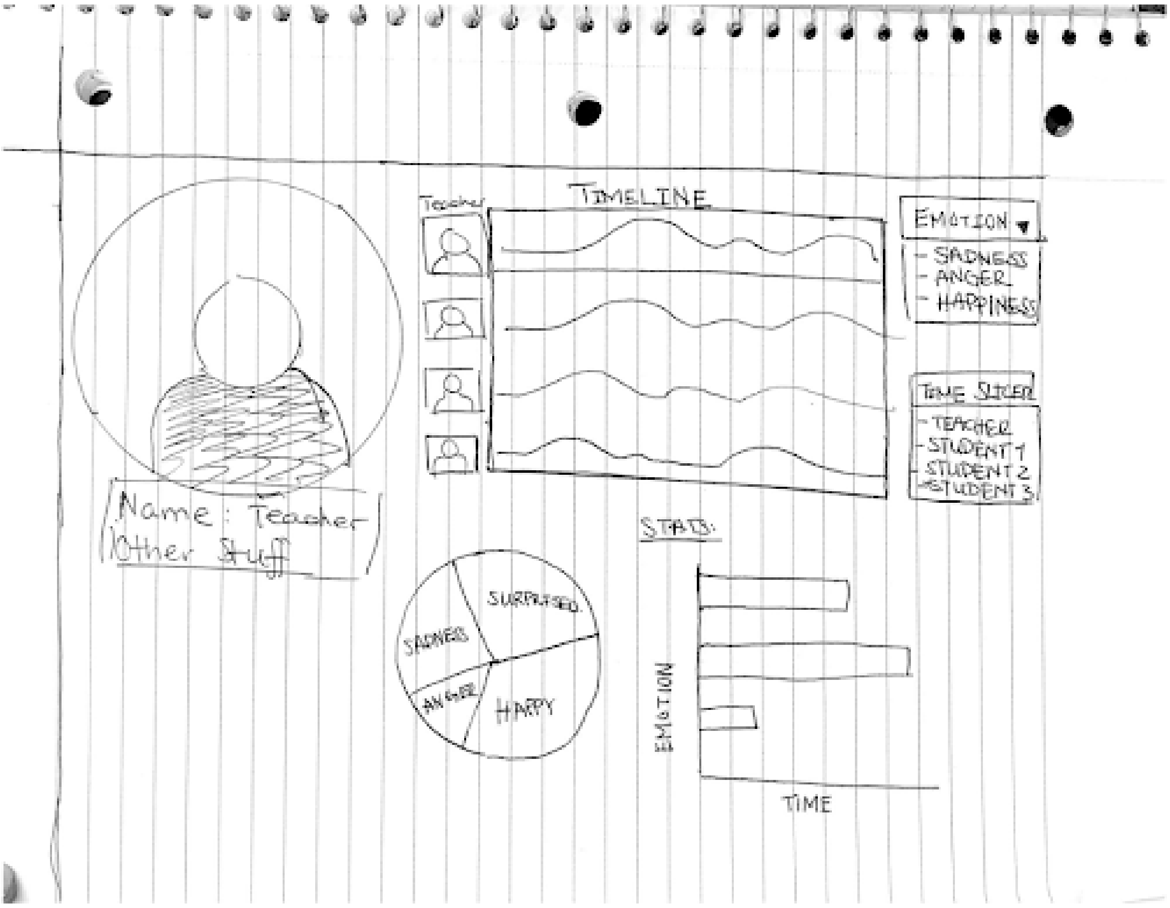


Figure 3.2: *Johvanni's Initial Design*
Johvanni's initial ideas drawing. This design is inspired by creating a profile per person and when that profile is chosen, it releases the visual elements in various different representations. Ex. bar graph, timeline, pie chart.

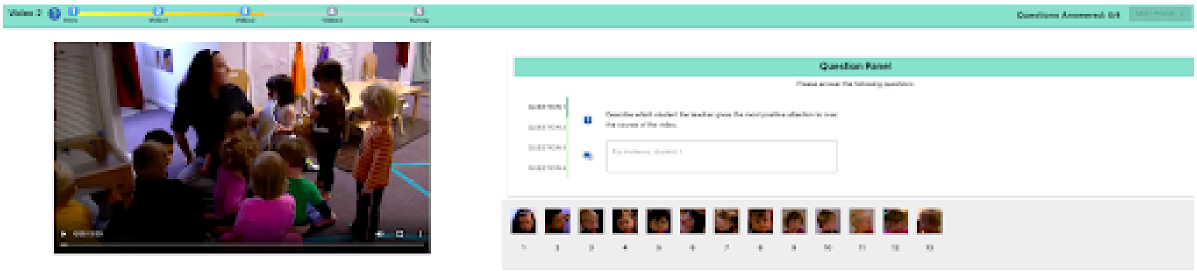


Figure 3.3: *COILS Current Demo*
Current interface of Video 2 in the COILS demo. This interface contains a variety of questions and matches the faceID value that is used in the data files to students seen in the video.

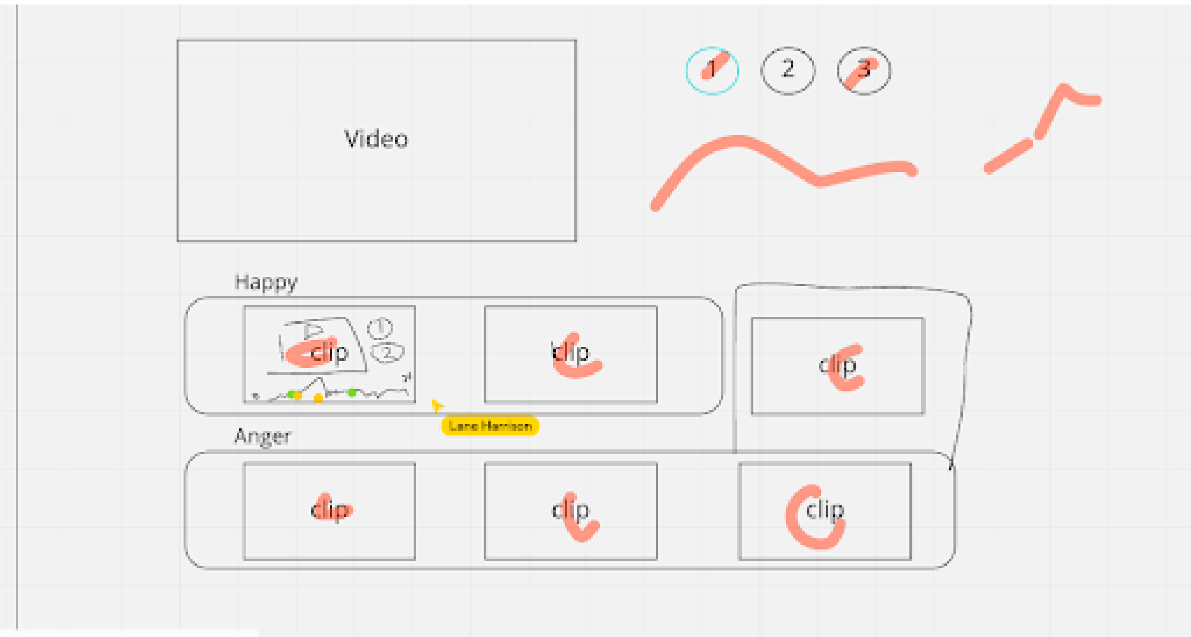


Figure 3.4: *The Team's Initial Clip Design*
Overall page initial idea drawing. This design entails the idea of clicking on a specific faceID and then all the emotions are stacked on top of each other with the top three clips of that emotion.

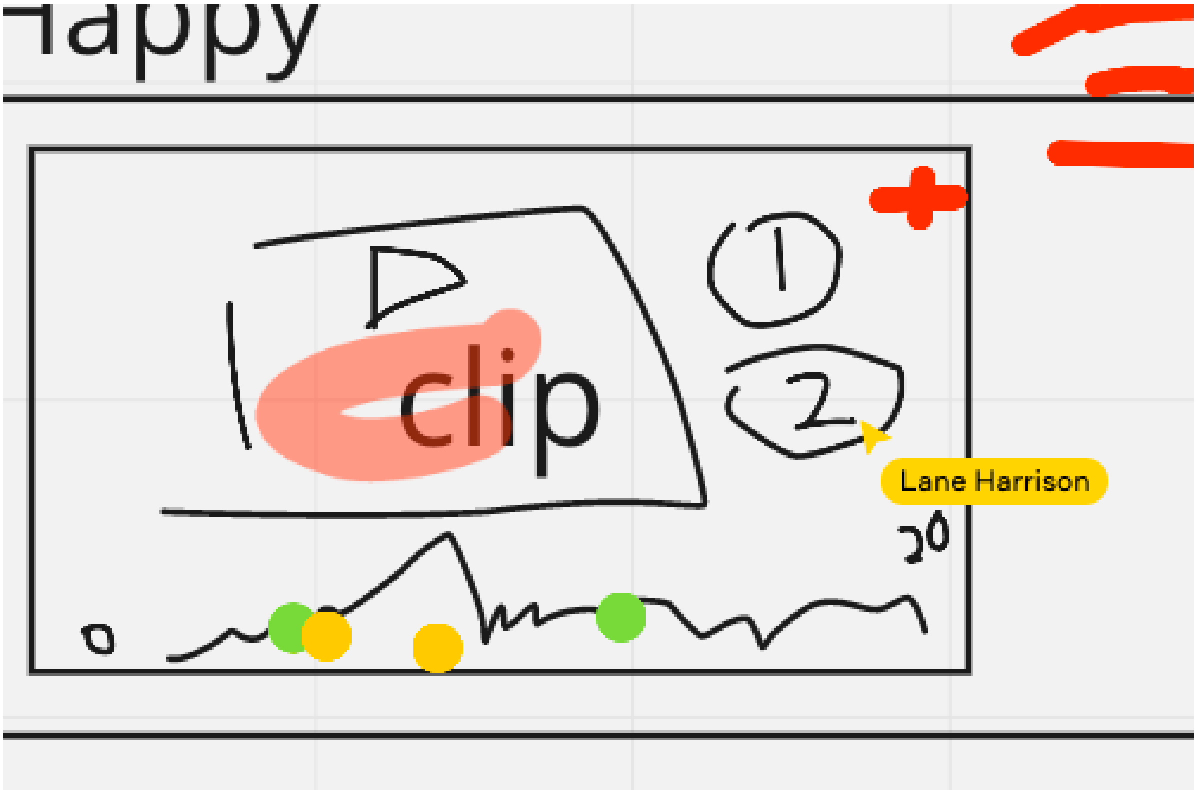


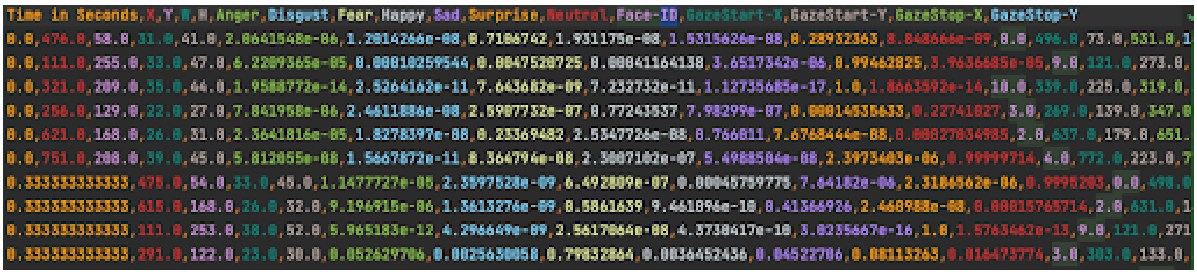
Figure 3.5: *Individual Clip Design*
Smaller clip details initial idea. Here you can see that it contains a timeline where the emotion is occurring as well as contains the faceID value of the person we are looking at.

Throughout our process of brainstorming ideas for the new UI, our main goal was to make sure that we can have the videos create a story that the teachers can conceptualize. In order to do so, using software that was dedicated to designing websites became an important aspect of our project. Bootstrap gave us that leisure to test out our initial design and how it looked on a web page. Bootstrap is an open-source CSS framework for front-end web development. As the front-end of the project came along with the use of bootstrap and it's functionality, the other integral portion of the project became the data manipulation in order to extract the correct information that we wanted to display.

3.3 Data Manipulation: Filtering and Processing the Initial Data Files

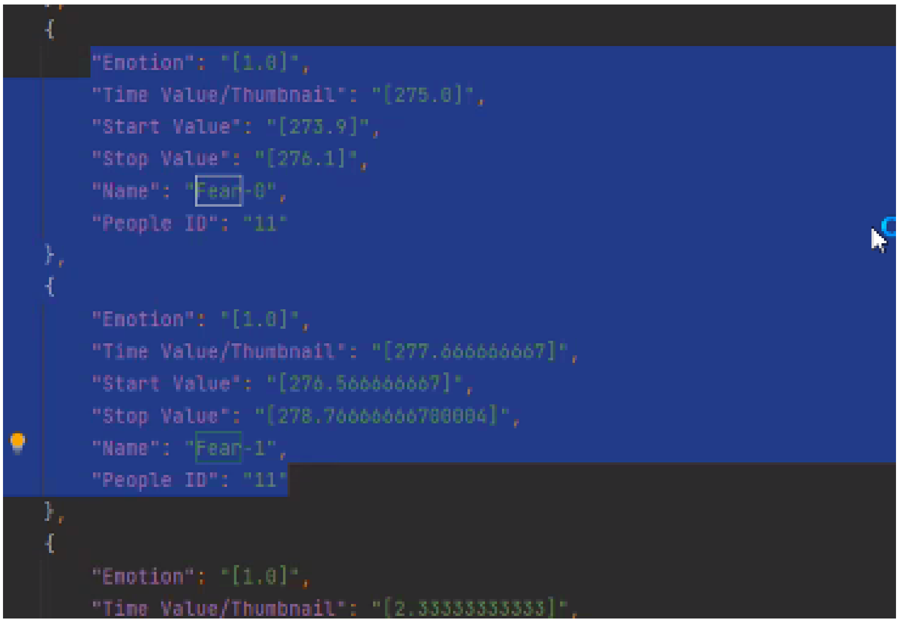
One of the most integral parts of our project is manipulating the data files in order to get the exact clips we need to integrate to the front end. Although the front-end portion code is written using JavaScript, for our data we used Python. In order to provide a gif/clip of a specific moment a student or a teacher is expressing through the video, we had to filter out the raw data then convert that filtered information into a JSON file to pass to the front-end. The values that the raw data file contains can be seen in Figure 3.6.

The JSON file contains a number of important information: emotion value, time/thumbnail value, a start time value, end time value, a name for that emotion, and the faceID value. We started off by smoothing out the values in the emotions column to get rid of noise. We then took the peak values that we detected and we looked for their timestamps. With those timestamps, we then did a small calculation where it calculates about a few seconds before and after that timestamp so then we can create the new gif/clip produced that will be shown on the emotions timeline. We took all of the information gathered and ran it through a createJSON function that will incorporate all of the peaks' values. As we went through this process, we first tried it with a singular emotion and then we had to create various loops in order to go through every emotion for every faceID as seen in Figure 3.7. After creating various JSON files for all twelve faceID's and each of their emotions, we will then combine them all into a singular JSON file that we can then integrate the data side and the front-end.



```
Time in Seconds,X,Y,W,H,Anger,Disgust,Fear,Happy,Sad,Surprise,Neutral,Face-ID,GazeStart-X,GazeStart-Y,GazeStop-X,GazeStop-Y
0.0,476.0,58.0,31.0,41.0,2.0641548e-06,1.2814266e-08,0.7106742,1.931175e-08,1.5315626e-08,0.28932363,8.848666e-09,0.0,496.0,73.0,531.0,1
0.0,111.0,255.0,33.0,47.0,6.2209365e-05,0.00010259544,0.0047520725,0.00041164138,3.6517342e-06,0.99462825,3.9636685e-05,9.0,121.0,273.0,
0.0,321.0,209.0,35.0,44.0,1.9588772e-14,2.5264162e-11,7.643682e-09,7.232732e-11,1.1273568e-17,1.0,1.8663592e-14,10.0,339.0,225.0,319.0,
0.0,256.0,129.0,22.0,27.0,7.841958e-06,2.4611886e-08,2.5907732e-07,0.77243537,7.98299e-07,0.00014535633,0.22741027,3.0,269.0,139.0,347.0
0.0,621.0,168.0,26.0,31.0,2.3641816e-05,1.8278397e-08,0.23369482,2.5347726e-08,0.766011,7.6768444e-08,0.00027034985,2.0,637.0,179.0,651.
0.0,751.0,208.0,39.0,45.0,5.812055e-08,1.5667872e-11,8.364794e-08,2.3007102e-07,5.4988584e-08,2.3973403e-06,0.99999714,4.0,772.0,223.0,7
0.333333333333,475.0,54.0,33.0,45.0,1.1477727e-05,2.3597528e-09,6.492809e-07,0.00045759775,7.64182e-06,2.3186562e-06,0.9995203,0.0,498.0
0.333333333333,615.0,168.0,26.0,32.0,9.196915e-06,1.3613276e-09,0.5861639,9.461896e-10,0.41366926,2.468988e-08,0.00015765714,2.0,631.0,1
0.333333333333,111.0,253.0,38.0,52.0,5.965183e-12,4.296649e-09,2.5617064e-08,4.3730417e-10,3.0235667e-16,1.0,1.5763462e-13,9.0,121.0,271
0.333333333333,291.0,122.0,23.0,30.0,0.052629706,0.0025630058,0.79032864,0.0036452436,0.04522706,0.00113263,0.016473774,3.0,303.0,133.0,
```

Figure 3.6: *Current Raw Data*
Snippet of the raw data we used The raw data contains all the information needed of each individual represented in the video. We used this raw data for the "*hire_TE.mp4*" video.



```
{
    "Emotion": "[1.0]",
    "Time Value/Thumbnail": "[275.0]",
    "Start Value": "[273.9]",
    "Stop Value": "[276.1]",
    "Name": "Fear-0",
    "People ID": "11"
},
{
    "Emotion": "[1.0]",
    "Time Value/Thumbnail": "[277.666666667]",
    "Start Value": "[276.566666667]",
    "Stop Value": "[278.7666666667000004]",
    "Name": "Fear-1",
    "People ID": "11"
},
{
    "Emotion": "[1.0]",
    "Time Value/Thumbnail": "[2.33333333333]",
```

Figure 3.7: *First JSON File Produced*

This is a sample result of the JSON file that calculated the peak for a specific emotion and extracted specific information like the time values and then we created the start and stop values.

3.4 UI Implementation

After our brainstorming session, we created various prototypes on how we wanted our front-end to look like. In order to do so, we used bootstrap and coded in JavaScript to create the page. Our front-end page obtained various features ranging from a collapse sidebar, student profile buttons, a main section for the video, and the top three clips for every emotion that the student profile possessed. Features can be seen in Figures 3.8, 3.9, and 3.10.

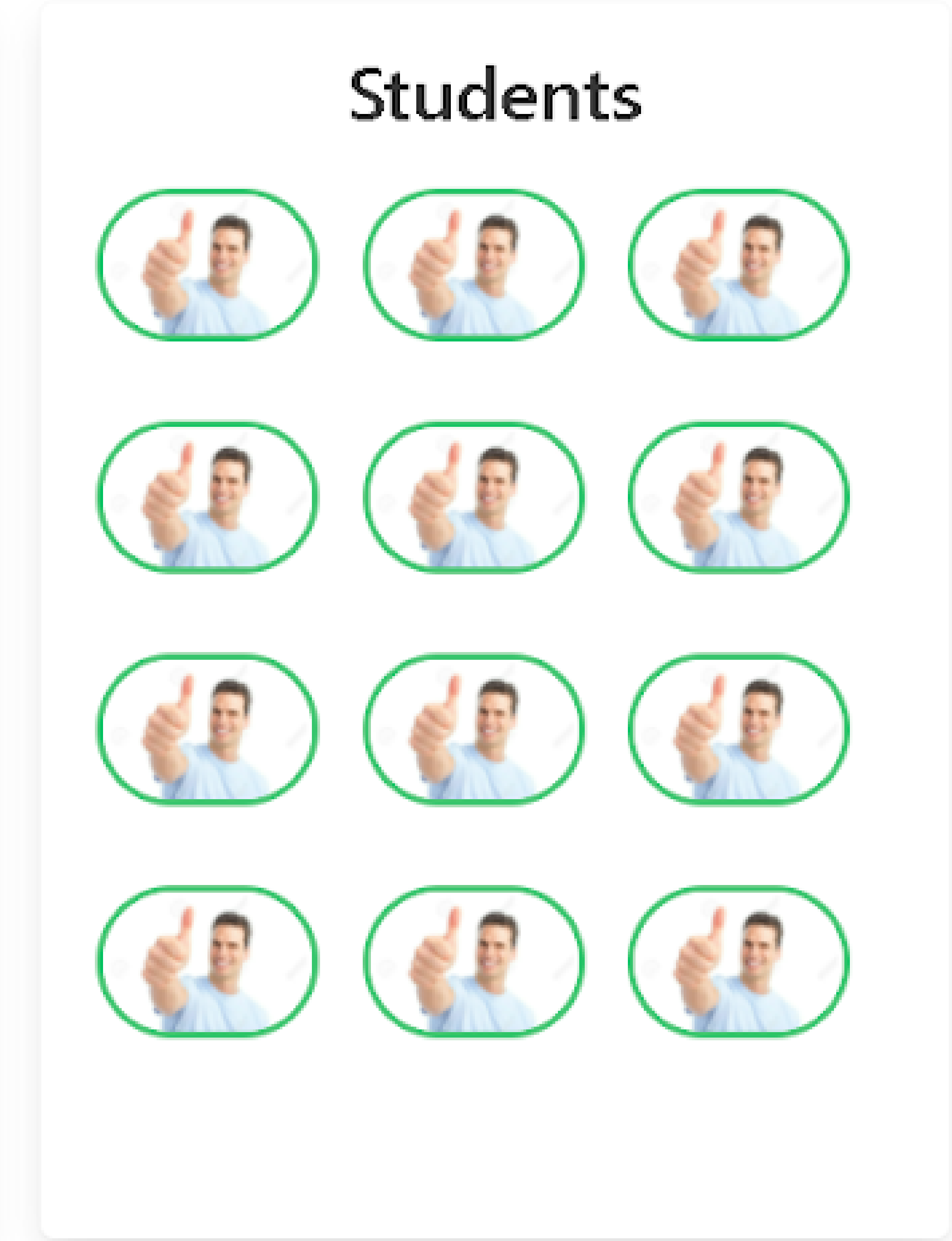


Figure 3.8: *Initial Student Profiles*
A prototype on how there would be different student profiles. Each of the buttons will have an image or number of the student correlating with their face ID number and once the button is pressed the list of emotions and their clips will appear.

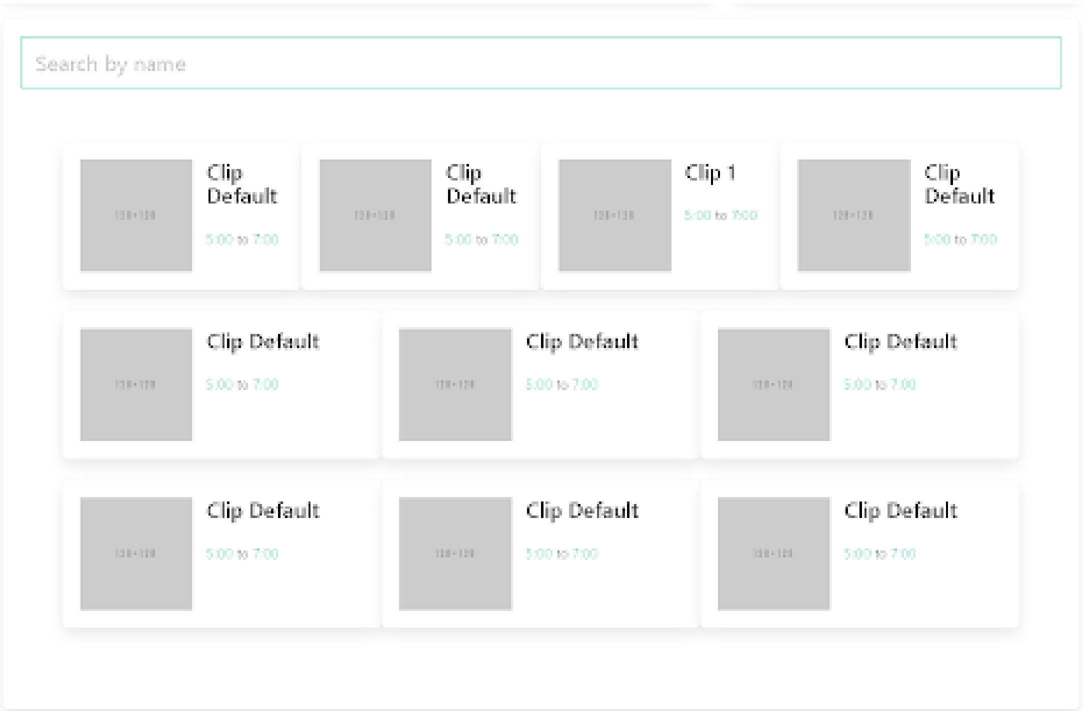


Figure 3.9: *Initial Clips Design*
This is a prototype on how the emotions and each of their clips will appear once the student profile is clicked. Each of the emotions as stated on the raw data file will be listed and then given by the new manipulated data, we will populate and insert clips of the top moments where that student/teacher expressed that emotion. In addition, with the search bar, you can also search for a specific emotion rather than just scrolling through the list.

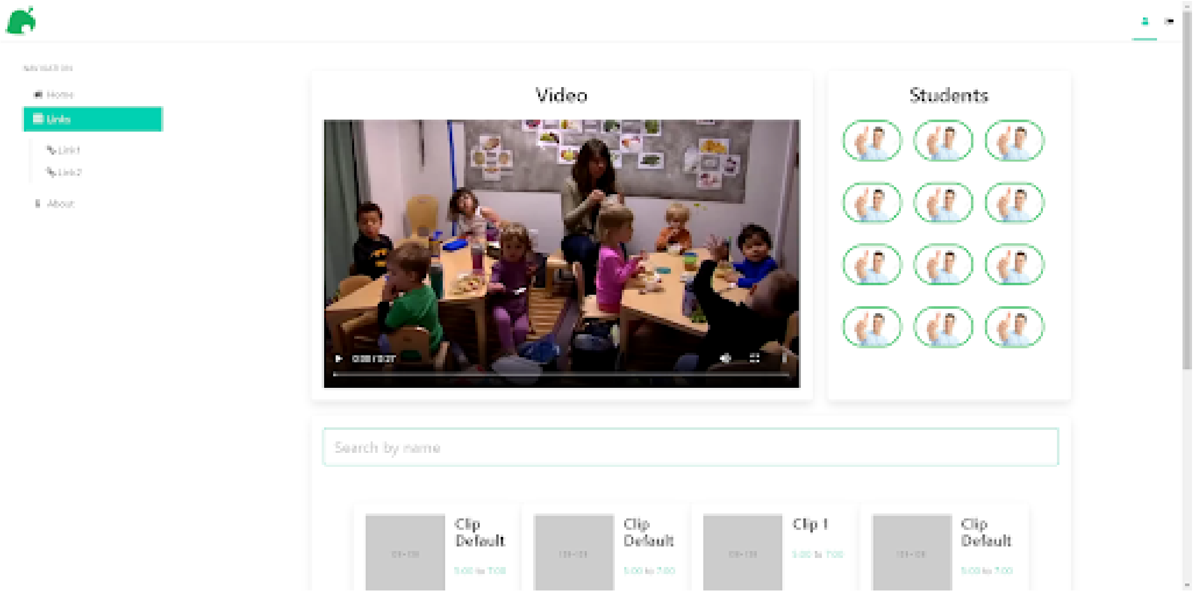


Figure 3.10: *Overall Initial UI View*
This is an overall view with the page as you see the different components as stated above surrounds the main video we are analyzing.

3.5 Development of New User Interface

We analyzed the data files gathered from the classroom's videos and analyzed the feedback given from participants and with that information, we were able to create a strong foundation for the new UI for new visualization techniques. We were able to extract values from the data files that correlate with the peaks for each person and their different emotions throughout the video. By extracting this information then implementing it all in a JSON file, we are able to pass that JSON file to the front end and display small clips of that given time period. Implementing this new method of video-clip visualization will benefit teachers to understand students' behaviors and how to improve their teaching styles in the classroom.

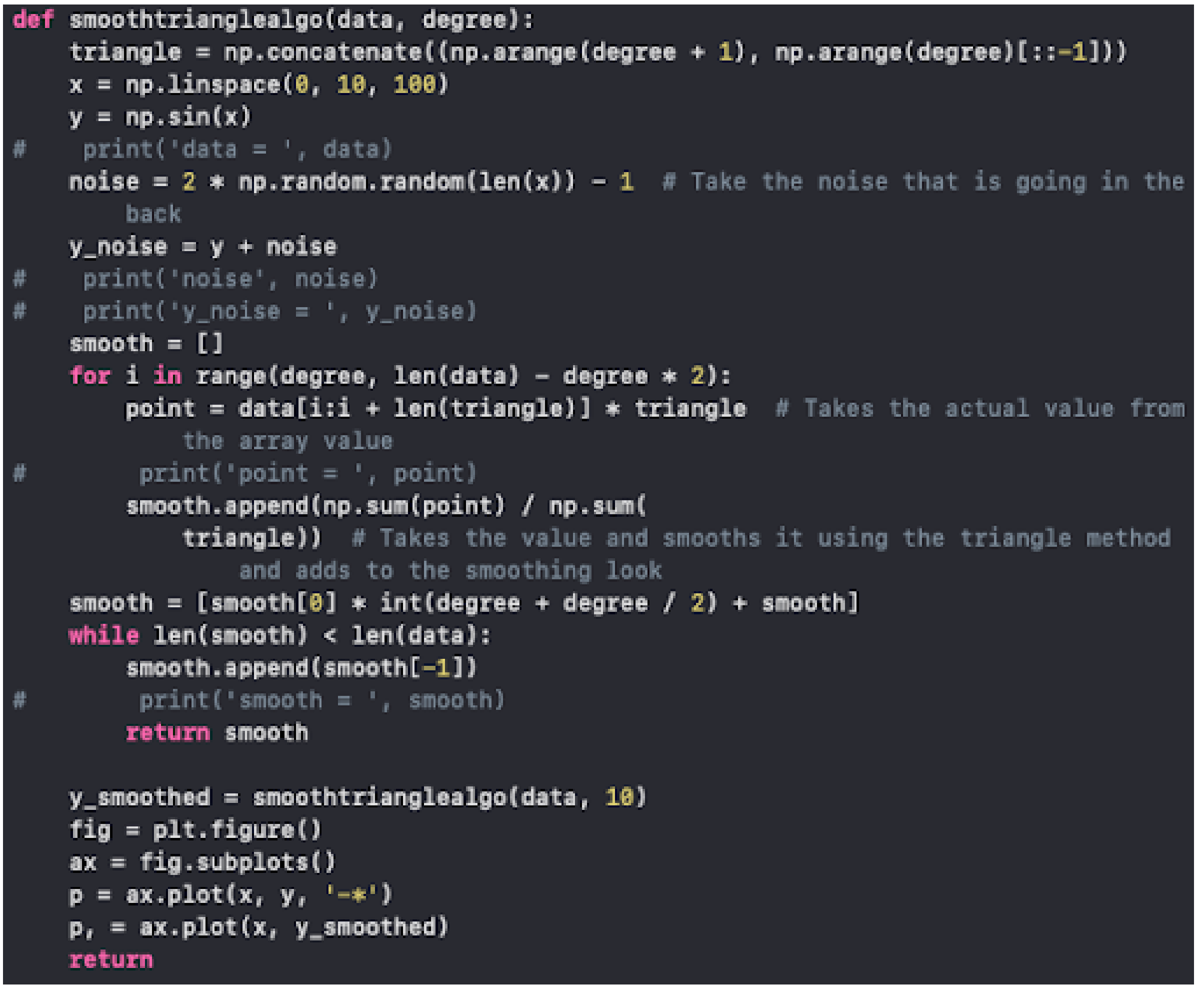
Chapter 4

Results

When it comes to visualizing data that is seen for the first time, making it easy to understand for those who are not experienced in data visualization is very important. In order to do so, we started off by manipulating data so that we can visualize solely the specific times where the emotions that are being expressed are visible. We then learned about how to use Bulma to create a new UI for the system that would provide areas that would allow the user to filter what is being shown on the screen by people or by emotions. In addition, we then incorporated a way on how to integrate all of the data with the necessary information and video clips into the new UI. With this integration, users are able to identify the top three emotions that the different people express throughout the time of the video.

4.1 Video Clips Creation

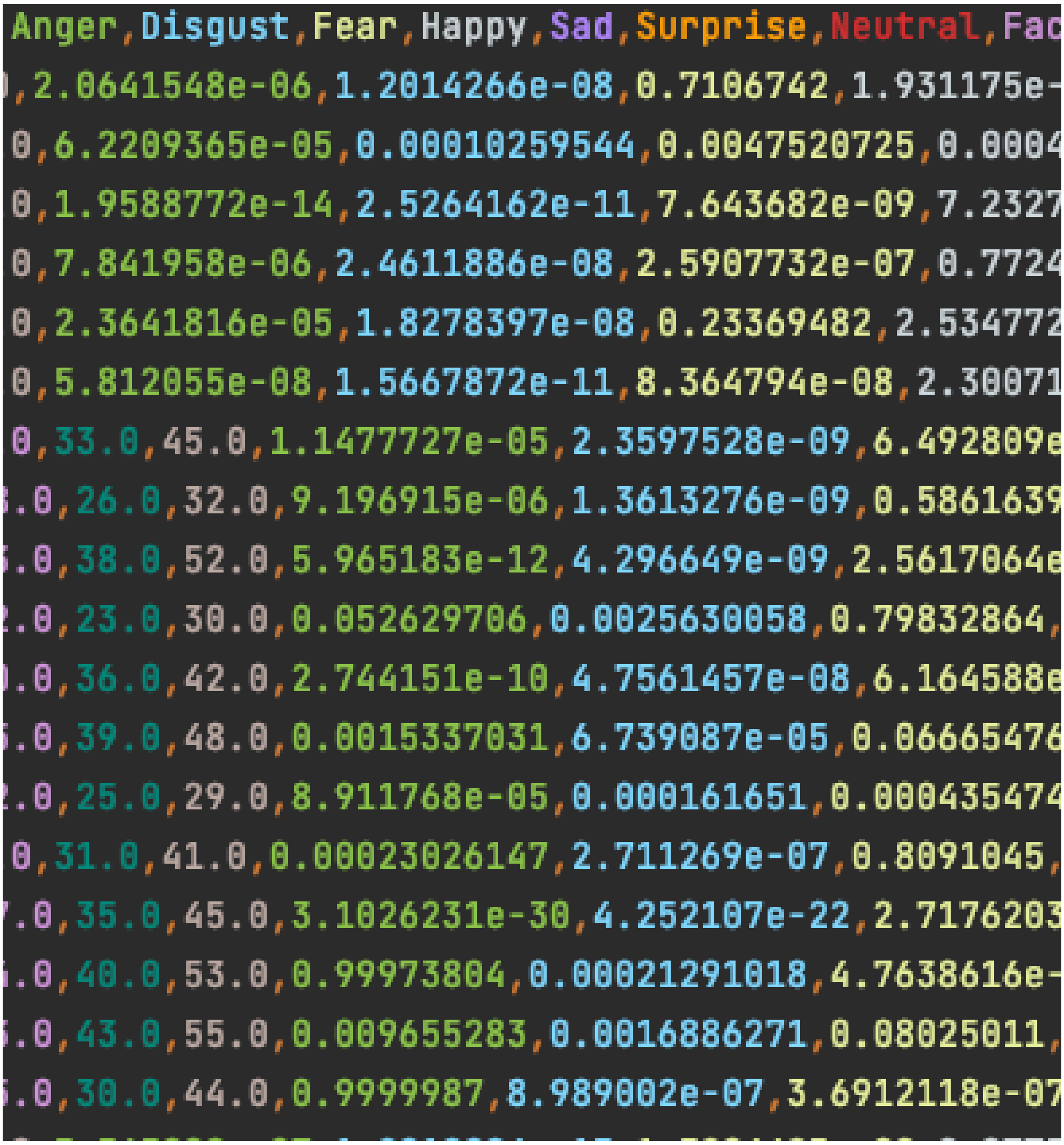
We started off our project by first analyzing the data that was already created during the beginning of the COILS project where the machine learning was already identifying faces and emotions. With this we took the data that was produced in the “*T_1_L_3_FaceTracks_6_26_2020.csv*” file and we used the video “*hire_TE.mp4*.” We first created a smoothing algorithm code that would clear off the noise in the data and would extract the peak emotion values as seen in Figure 4.1.



```
def smoothtrianglealgo(data, degree):
    triangle = np.concatenate((np.arange(degree + 1), np.arange(degree)[::-1]))
    x = np.linspace(0, 10, 100)
    y = np.sin(x)
#    print('data = ', data)
    noise = 2 * np.random.random(len(x)) - 1  # Take the noise that is going in the
        back
    y_noise = y + noise
#    print('noise', noise)
#    print('y_noise = ', y_noise)
    smooth = []
    for i in range(degree, len(data) - degree * 2):
        point = data[i:i + len(triangle)] * triangle  # Takes the actual value from
            the array value
#        print('point = ', point)
        smooth.append(np.sum(point) / np.sum(
            triangle))  # Takes the value and smooths it using the triangle method
                and adds to the smoothing look
    smooth = [smooth[0] * int(degree + degree / 2) + smooth]
    while len(smooth) < len(data):
        smooth.append(smooth[-1])
#        print('smooth = ', smooth)
        return smooth

    y_smoothed = smoothtrianglealgo(data, 10)
    fig = plt.figure()
    ax = fig.subplots()
    p = ax.plot(x, y, '-*')
    p, = ax.plot(x, y_smoothed)
    return
```

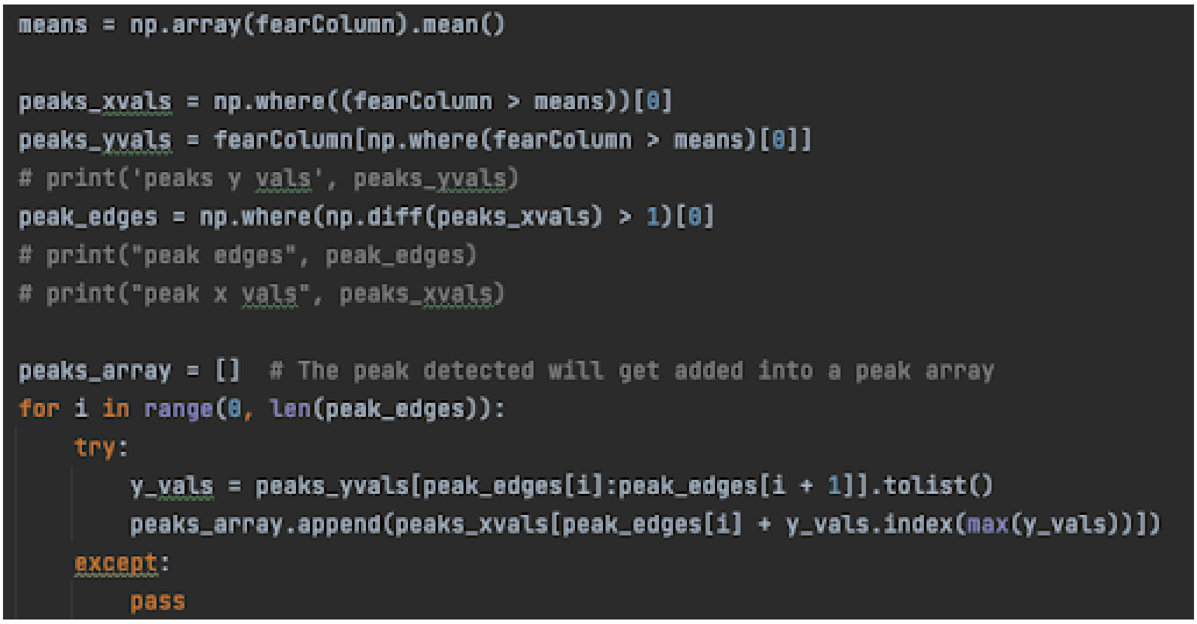
Figure 4.1: *Smoothing Algorithm*
This smoothing algorithm helps detect the peaks by first smoothing out the data that was provided to us and providing a cleaner curve of the emotion values that we will then pass through a detect peaks function that finds the actual peaks.

To get a more in depth understanding about how we are determining emotion values, in Figure 4.2, there are multiple columns that state different types of emotions, the values in each respective column are the values for each emotion that we are trying to extract the highest values that represent the peak. We calculated the peaks as values that are greater than the mean values of the emotion values as seen represented in Figure 4.3. Once we have then found all of those peaks, we then inserted all of those peak values in an array that holds all of those peak values that were extracted. With these values we can then store important information such as time values when that specific peak occurs and store it in a CSV file that then can be sent over to a JSON file. In addition with those time values, we then created clips of those certain time periods.



```
Anger,Disgust,Fear,Happy,Sad,Surprise,Neutral,Fac
,2.0641548e-06,1.2014266e-08,0.7106742,1.931175e-
0,6.2209365e-05,0.00010259544,0.0047520725,0.0004
0,1.9588772e-14,2.5264162e-11,7.643682e-09,7.2327
0,7.841958e-06,2.4611886e-08,2.5907732e-07,0.7724
0,2.3641816e-05,1.8278397e-08,0.23369482,2.534772
0,5.812055e-08,1.5667872e-11,8.364794e-08,2.30071
0,33.0,45.0,1.1477727e-05,2.3597528e-09,6.492809e
.0,26.0,32.0,9.196915e-06,1.3613276e-09,0.5861639
.0,38.0,52.0,5.965183e-12,4.296649e-09,2.5617064e
.0,23.0,30.0,0.052629706,0.0025630058,0.79832864,
.0,36.0,42.0,2.744151e-10,4.7561457e-08,6.164588e
.0,39.0,48.0,0.0015337031,6.739087e-05,0.06665476
.0,25.0,29.0,8.911768e-05,0.000161651,0.000435474
0,31.0,41.0,0.00023026147,2.711269e-07,0.8091045,
.0,35.0,45.0,3.1026231e-30,4.252107e-22,2.7176203
.0,40.0,53.0,0.99973804,0.00021291018,4.7638616e-
.0,43.0,55.0,0.009655283,0.0016886271,0.08025011,
.0,30.0,44.0,0.9999987,8.989002e-07,3.6912118e-07
```

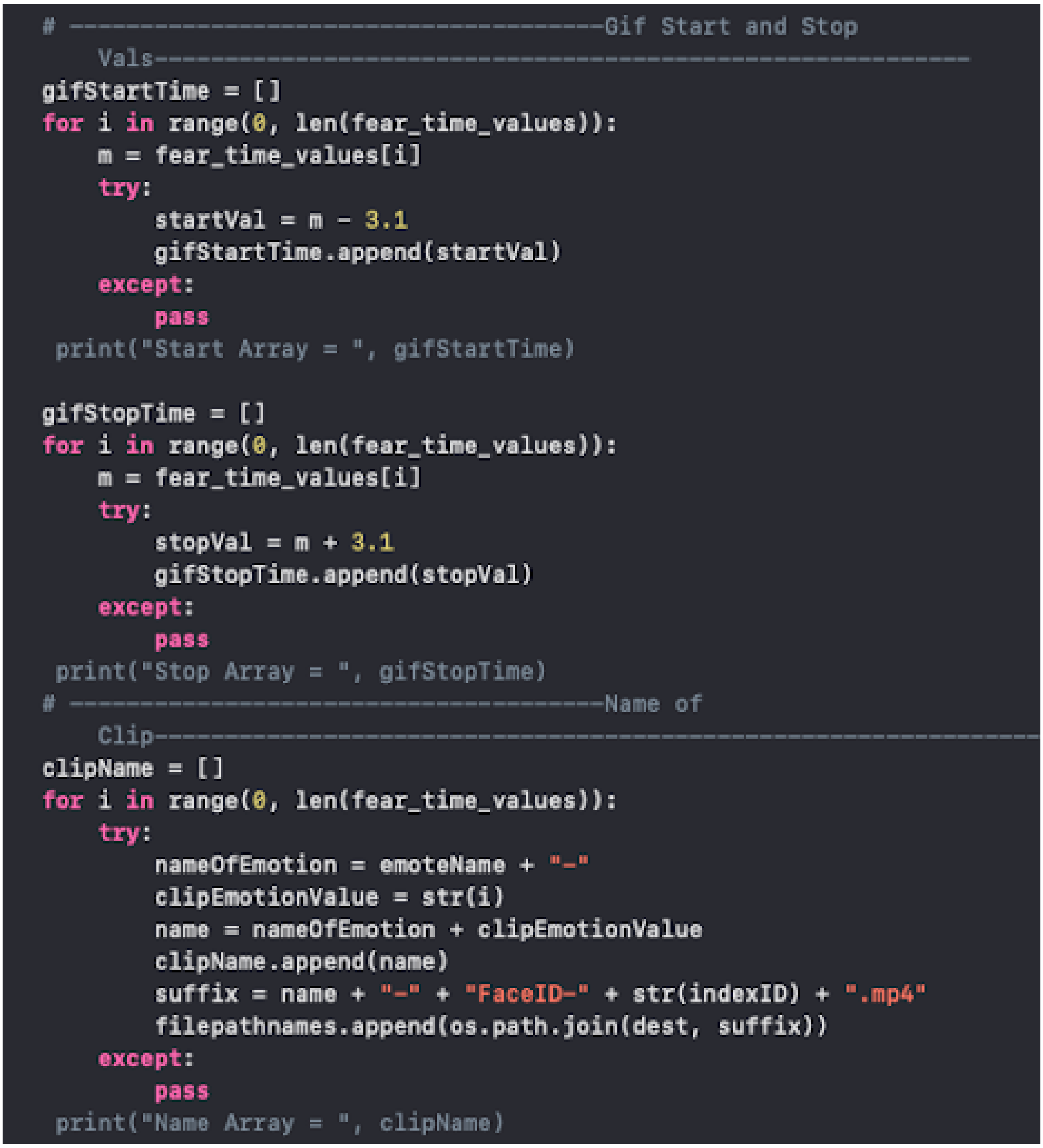
Figure 4.2: *Specific Emotion Values*
This is a quick image where the emotion values are represented in each column. The name of the emotion is at the very top and in PyCharm, each column has a different color representation to help visualize and differentiate values. For example, all of the anger values are green, all of the disgust values are light blue, etc.



```
means = np.array(fearColumn).mean()

peaks_xvals = np.where((fearColumn > means))[0]
peaks_yvals = fearColumn[np.where(fearColumn > means)[0]]
# print('peaks y vals', peaks_yvals)
peak_edges = np.where(np.diff(peaks_xvals) > 1)[0]
# print("peak edges", peak_edges)
# print("peak x vals", peaks_xvals)

peaks_array = []  # The peak detected will get added into a peak array
for i in range(0, len(peak_edges)):
    try:
        y_vals = peaks_yvals[peak_edges[i]:peak_edges[i + 1]].tolist()
        peaks_array.append(peaks_xvals[peak_edges[i] + y_vals.index(max(y_vals))])
    except:
        pass
```

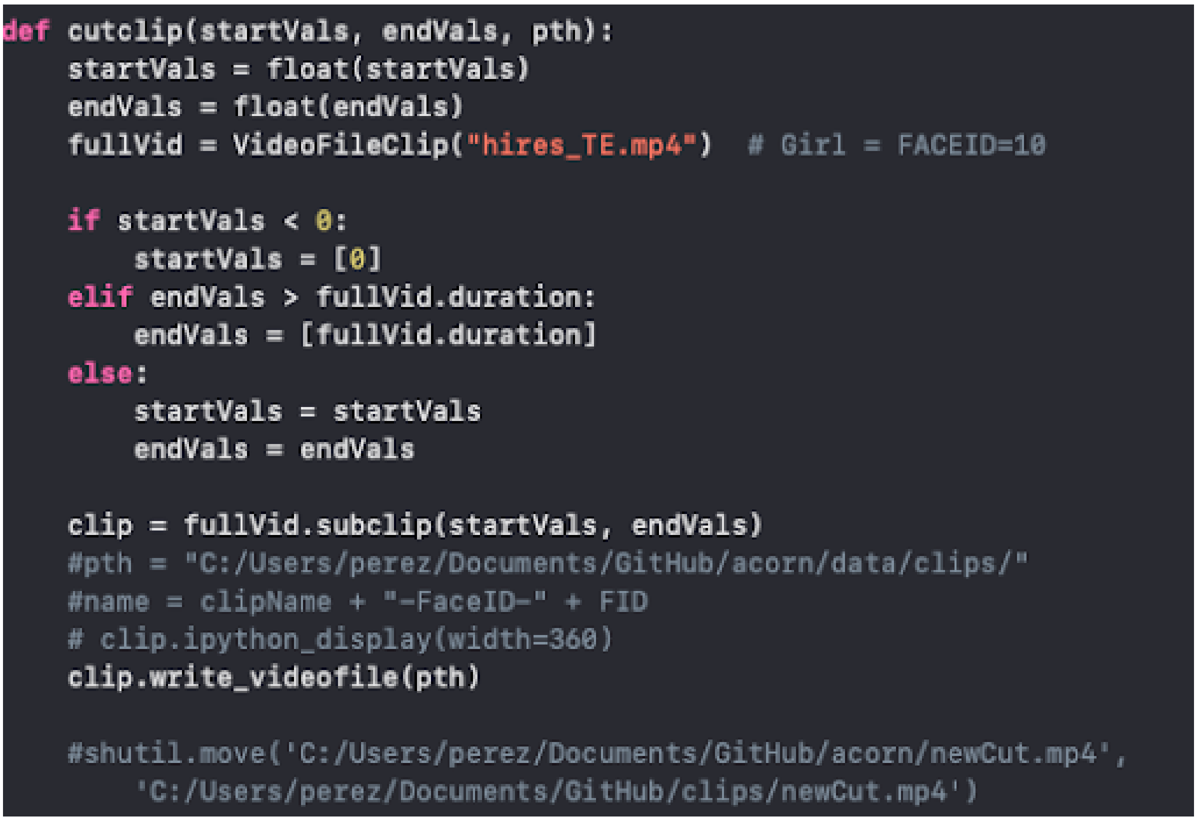
Figure 4.3: *Mean Calculation*
This representation of a quick snippet of the code where we calculated the mean and then went through and compared each value with the mean value and added it to the array.

With all of these values extracted, we then sorted throughout the csv file to find the start and end time for each of the values. We needed the start and end time because we would be compiling the clips with those values. In Figure 4.4, we used those time values to add a few additional seconds to both the start and end times into new different variables so the clips were not so short. Each clip that we created had a time span of about seven seconds. With those variables that we stored with the new start and end times, that is how we created our clips. We used movie.py in order to work with mp4 files. Movie.py is a python library for video editing, cutting, concatenating, and more. As seen in Figure 4.5, we used the time values to look through the "hire_TE.mp4" video and used the start and end files to create the sub-clips that we will use in the prototype.



```
# --------------------------------------Gif Start and Stop
    Vals---------------------------------------------------------
gifStartTime = []
for i in range(0, len(fear_time_values)):
    m = fear_time_values[i]
    try:
        startVal = m - 3.1
        gifStartTime.append(startVal)
    except:
        pass
 print("Start Array = ", gifStartTime)

gifStopTime = []
for i in range(0, len(fear_time_values)):
    m = fear_time_values[i]
    try:
        stopVal = m + 3.1
        gifStopTime.append(stopVal)
    except:
        pass
 print("Stop Array = ", gifStopTime)
# --------------------------------------Name of
    Clip-------------------------------------------------------------
clipName = []
for i in range(0, len(fear_time_values)):
    try:
        nameOfEmotion = emoteName + "-"
        clipEmotionValue = str(i)
        name = nameOfEmotion + clipEmotionValue
        clipName.append(name)
        suffix = name + "-" + "FaceID-" + str(indexID) + ".mp4"
        filepathnames.append(os.path.join(dest, suffix))
    except:
        pass
 print("Name Array = ", clipName)
```

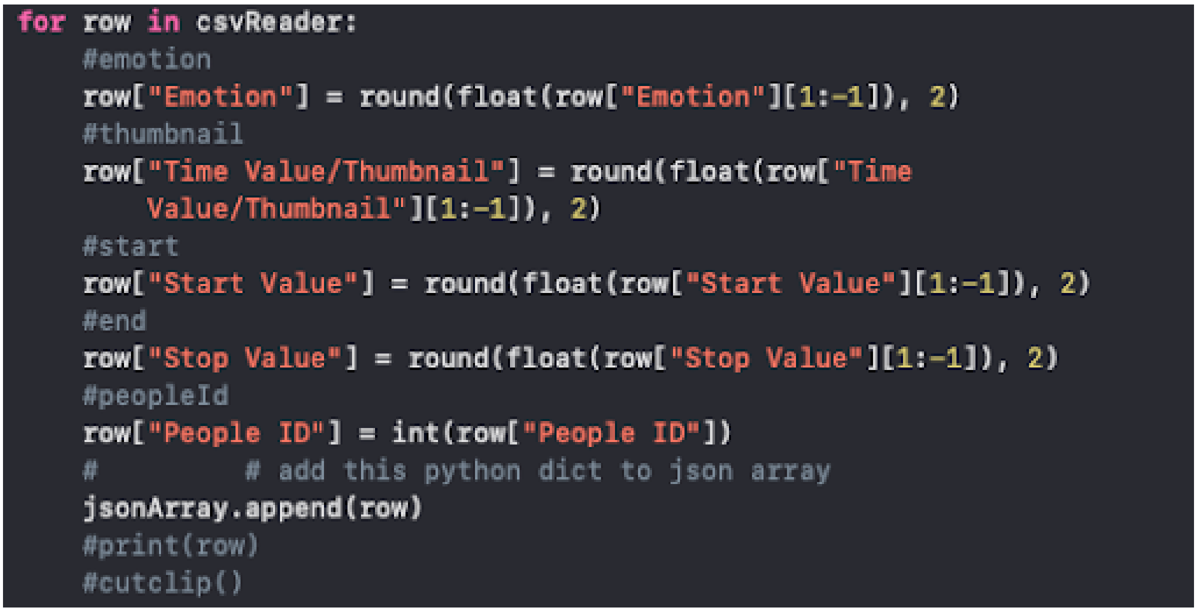
Figure 4.4: *Clips Start and Stop Times*
We created new variables gifStartTime and gifStopTime to increase the duration of the video clip. In addition, we also created the naming convention of the name of the clip so it is easy for the integration process.



```
def cutclip(startVals, endVals, pth):
    startVals = float(startVals)
    endVals = float(endVals)
    fullVid = VideoFileClip("hires_TE.mp4")  # Girl = FACEID=10

    if startVals < 0:
        startVals = [0]
    elif endVals > fullVid.duration:
        endVals = [fullVid.duration]
    else:
        startVals = startVals
        endVals = endVals

    clip = fullVid.subclip(startVals, endVals)
    #pth = "C:/Users/perez/Documents/GitHub/acorn/data/clips/"
    #name = clipName + "-FaceID-" + FID
    # clip.ipython_display(width=360)
    clip.write_videofile(pth)

    #shutil.move('C:/Users/perez/Documents/GitHub/acorn/newCut.mp4',
        'C:/Users/perez/Documents/GitHub/clips/newCut.mp4')
```

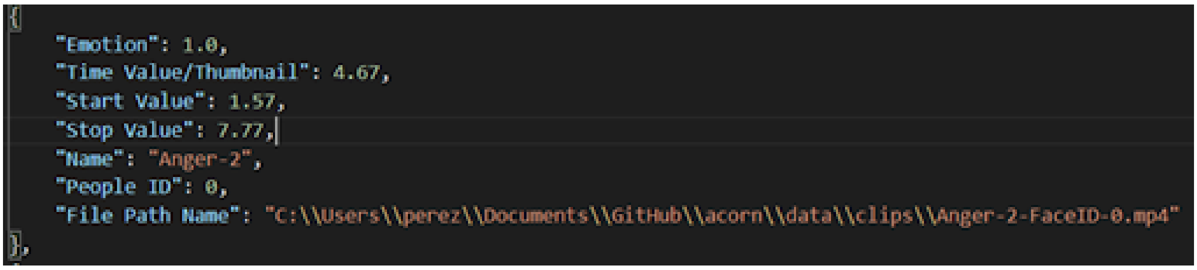
Figure 4.5: *cutClips Function*
The function cutClips uses functions in the movie.py to cut the video in questions to the values that we inputted. This then sends the newly created clip to a path/directory where we can find the new clip.

Once all of the values of each important were extracted from the original csv file and the creation of the various clips that were in relation to those various clips were created, we stored all of that respective information in one abject and into a csv file as seen in Figure 4.6. In Figure 4.7, an example of a singular object that holds all of the necessary information needed to pass into the UI section. The values that are needed for each object are the emotion values, the time values for that emotion, the start and stop value that we previously calculated, the name that we created for the clips, the face ID of who that clip represents, and the file path name of where the clip is stored. Once we were able to get all of the objects that represented all of the different emotions peaks for different people, we then converted that file into a JSON file. JSON files are much easier to extract information from so we created one singular file that contained all the objects of the clips. Once the JSON file was done, that was the end of our data extraction where we can then implement the clips into the front-end.



```python
for row in csvReader:
    #emotion
    row["Emotion"] = round(float(row["Emotion"][1:-1]), 2)
    #thumbnail
    row["Time Value/Thumbnail"] = round(float(row["Time
        Value/Thumbnail"][1:-1]), 2)
    #start
    row["Start Value"] = round(float(row["Start Value"][1:-1]), 2)
    #end
    row["Stop Value"] = round(float(row["Stop Value"][1:-1]), 2)
    #peopleId
    row["People ID"] = int(row["People ID"])
    #         # add this python dict to json array
    jsonArray.append(row)
    #print(row)
    #cutclip()
```

Figure 4.6: *Implementation to CSV File*
This code shows how to implement all of the necessary information into a csv file that will create objects.



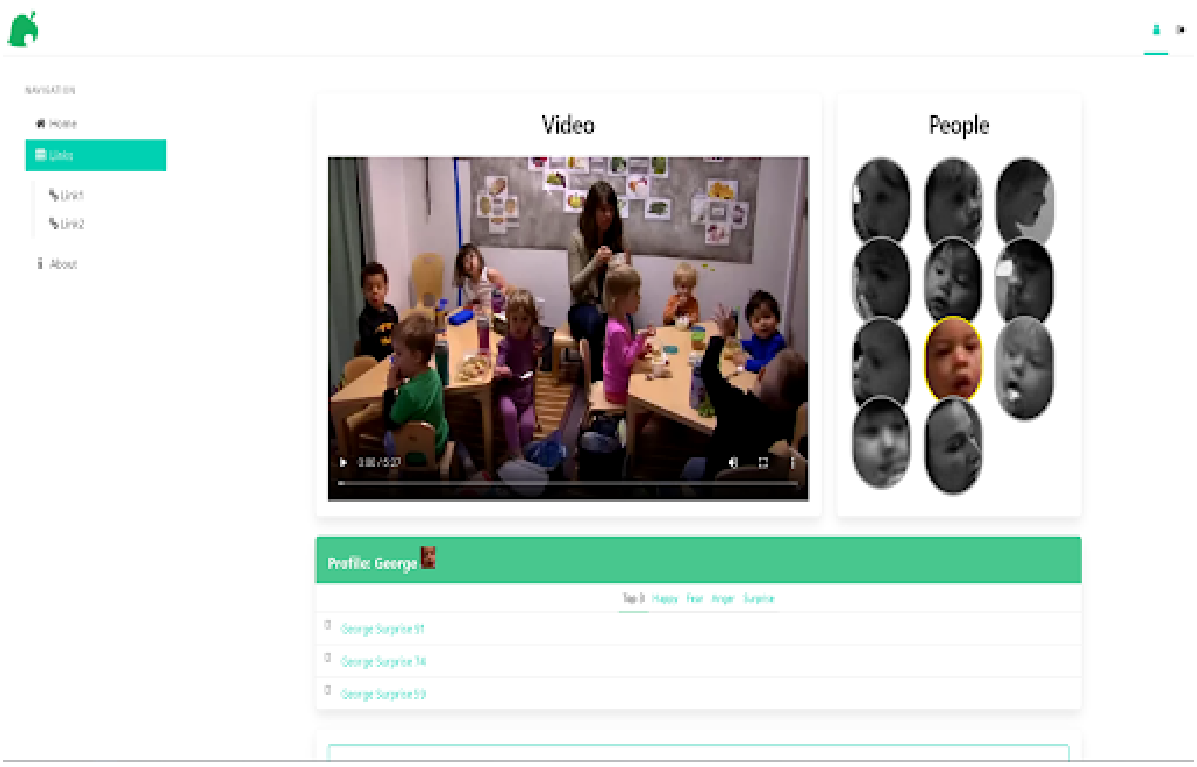
```json
{
    "Emotion": 1.0,
    "Time Value/Thumbnail": 4.67,
    "Start Value": 1.57,
    "Stop Value": 7.77,
    "Name": "Anger-2",
    "People ID": 0,
    "File Path Name": "C:\\Users\\perez\\Documents\\GitHub\\acorn\\data\\clips\\Anger-2-FaceID-0.mp4"
},
```

Figure 4.7: *Single Object*
A singular object that is in the JSON file that contains the necessary values needed to be easily accessed on the UI end such as the naming conventions and the path to where the new clips are.

4.2 Integration

The integration between the new data and the UI side of the project was the key piece for the completion of the project. With the JSON file and the path that held all of the clips, the prototype of the new clip visualization page was created. The UI section of the project was created using Bulma and was coded in JavaScript. Bulma is an open source CSS framework that can be used to create html classes. We used JavaScript to code the framework as well as extract the pieces needed in the JSON file. With the naming conventions that we used that differentiated the emotions and the face IDs, we were able to then filter out clips based on the face ID value and/or emotions depending on whichever the user wanted to view. In addition, we were also able to extract the top three emotions per face ID.

The prototype has many interactive features for the user. The top portion of the prototype html page holds the video we were working on and the profiles of all the profiles of all of the students and teachers seen in the video. Once a user presses on a specific profile, the other faces then grays out to provide an emphasis on who's emotions we are focusing on. Once a profile is clicked upon, a series of tabs appear as seen in Figure 4.8. The first tab shows the top three emotions that the person expressed. The following tabs are the different emotions and each tab has the top three time periods where the person expressed that specific emotion that the user clicked on. In addition to this filter, we also implemented filtering of emotions rather than by face ID. For example as seen in Figure 4.9, it displays the list of various students and teachers but it shows all of the peak times where they showed fear for example and the user can filter it between all of the different emotions. We also implemented the confidence level of how accurate the machine learning AI is when it comes to detecting specific emotions. Figure 4.10 is similar to Figure 4.9 but the differences are that it is filtered by solely the anger emotion for everyone and the confidence level of the AI is in red rather than blue. Each emotion has a varying color to distinguish from one another. On this prototype, the clips each have an interaction between the clip and the video shown at the top.



Profile: George

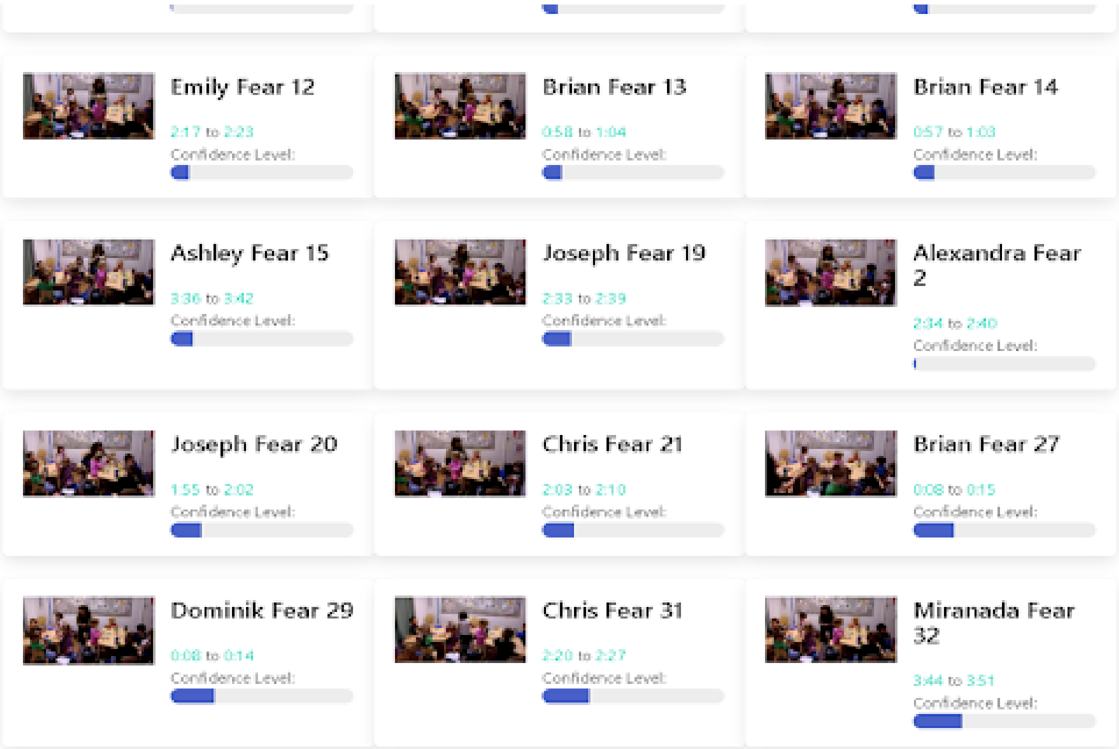
George Surprise 91

George Surprise 74

George Surprise 59

Figure 4.8: *Top View of Prototype*

The video that we used is the “$hires\_TE.mp4$” and the entirety of the video is seen on the left side while the profile images of people are seen on the right side. Once one of the profiles is clicked, the rest gray out and the tab of the name of the person and a smaller picture appear with the top three emotions. The following tops next to the top three are the rest of the emotions and top three times of that specific emotion are displayed.



Emily Fear 12
2:17 to 2:23
Confidence Level:

Brian Fear 13
0:58 to 1:04
Confidence Level:

Brian Fear 14
0:57 to 1:03
Confidence Level:

Ashley Fear 15
3:36 to 3:42
Confidence Level:

Joseph Fear 19
2:33 to 2:39
Confidence Level:

Alexandra Fear 2
2:34 to 2:40
Confidence Level:

Joseph Fear 20
1:55 to 2:02
Confidence Level:

Chris Fear 21
2:03 to 2:10
Confidence Level:

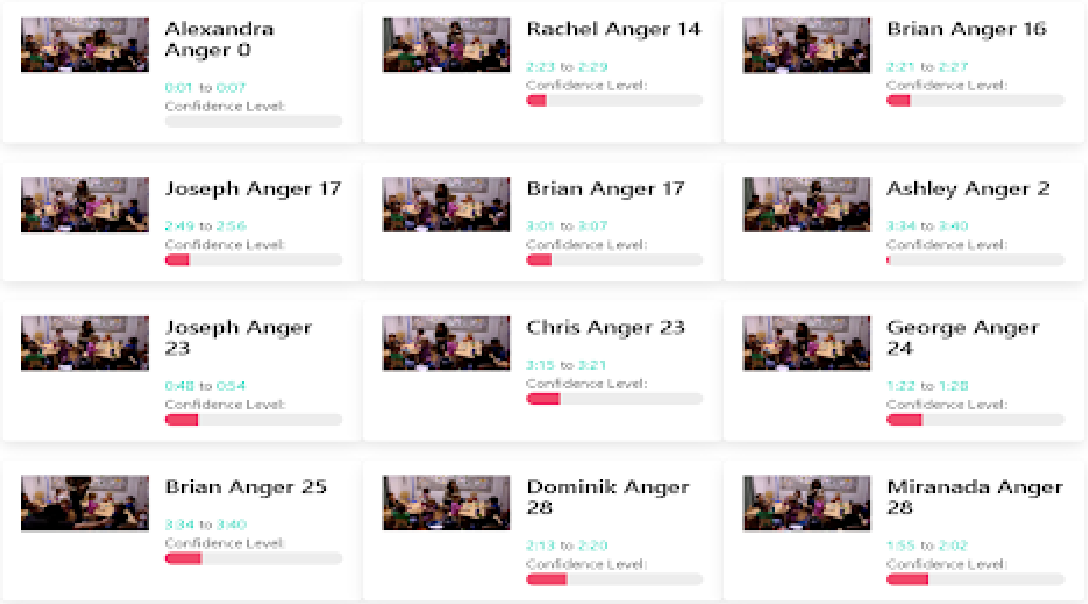
Brian Fear 27
0:08 to 0:15
Confidence Level:

Dominik Fear 29
0:08 to 0:14
Confidence Level:

Chris Fear 31
2:20 to 2:27
Confidence Level:

Miranada Fear 32
3:44 to 3:51
Confidence Level:

Figure 4.9: *Students vs Emotions*
The various different profiles appear but they all display the fear emotions.



Alexandra Anger 0
0:01 to 0:07
Confidence Level:

Rachel Anger 14
2:23 to 2:29
Confidence Level:

Brian Anger 16
2:21 to 2:27
Confidence Level:

Joseph Anger 17
2:49 to 2:56
Confidence Level:

Brian Anger 17
3:01 to 3:07
Confidence Level:

Ashley Anger 2
3:34 to 3:40
Confidence Level:

Joseph Anger 23
0:48 to 0:54
Confidence Level:

Chris Anger 23
3:15 to 3:21
Confidence Level:

George Anger 24
1:22 to 1:28
Confidence Level:

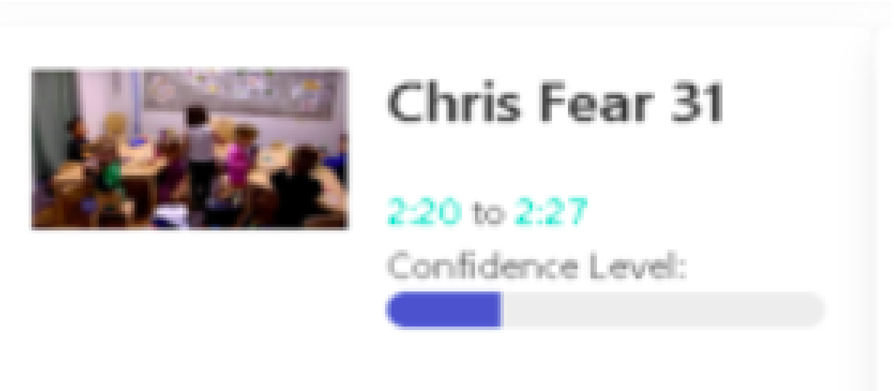
Brian Anger 25
3:34 to 3:40
Confidence Level:

Dominik Anger 28
2:13 to 2:20
Confidence Level:

Miranada Anger 28
1:55 to 2:02
Confidence Level:

Figure 4.10: *Confidence Levels*
Similar to Figure 29, there are confidence levels shown within each clip, all of the different emotions display a different color when displaying confidence levels.

As the user can be interactive when it comes to filtering out different emotions and specifying which person to focus on, the clips and the video can interact with each other as well. To have a more up close look at the clips, in Figure 4.11, the very first thing that is noticed when it comes to looking at the individual clips is the thumbnail image and the start and stop times of the clip in relation to the time of the longer video. The thumbnail is a single png image of a point in time in the video. Once the thumbnail is clicked, the video clip is played but also the video at the top of the HTML file cuts to the respected time of where the clip starts. This is very cool to visualize because with the additional seconds of the clip, we can see what caused that person to express that emotion and how long that emotion lasted. This will prove to be beneficial for teachers when they are going through this training system because it can help them either change their ways of teaching or help them better understand what types of actions they can avoid doing that could possibly impact a student. For creative purposes, the names that we used for the students are not their actual names but we included names because it would be easier than stating face ID numbers. With this prototype set, we hope that it can lead to additional ideas for the future of COILS.


Chris Fear 31

2:20 to 2:27

Confidence Level:

Figure 4.11: *Clips Details*
A representation of a singular object from the JSON file that is now a clip. Each contains a thumbnail of an image, name of person, emotion, the confidence level, and the start and stop times of when that emotion is being represented in the actual video. When a user clicks on the time, it makes the video at the top of the page skip to that time.

CHAPTER 5

RECOMMENDATIONS AND CONCLUSIONS

5.1 Recommendations

The future of the COILS can provide an endless list of different data visualization techniques. There can be many changes done to the current system made and also there are additional recommendations that can be made regarding our specific prototype. In addition to our prototype, there are many other data techniques that we wish that future teams can explore when continuing this project.

For the current COILS system, we have a list of recommendations that we did not incorporate into our project but could be explored in the future. First, the system can develop a more confident AI that detects emotions. As seen on the confidence levels that we show, the confidence levels seem to be somewhat lower than the halfway mark so if that can be improved, the more accurate the study could be. This could help our sponsors and the users to believe the AI more rather than doubting whether or not it is accurate. Another recommendation is that while clicking on a specific profile, a new page can pop-up where there are multiple detailed visualization techniques that represent different things. If a page with various different visualization techniques popped up such as the gaze interactions, emotions timeline, top clips, etc. with more detailed captions describing what that visualization represents, then the users who are not experienced in data visualization will have a better overall understanding. One last recommendation that we believe would benefit the project would be changing the survey questions that teachers need to respond to in order to get to the next page. It would be more beneficial if it was more of a fun interactive question such as making the tool-tip pop ups into question pop ups so the users can know exactly what they're looking at rather than guessing where it is on the page. Although the current COILS sections would benefit with additional improvements, so can our prototype.

If any additional teams would like to continue on with more of the video clip visualization, there are many improvements they can make to our prototype because it is far from perfect. One recommendation we would like to see be accomplished is creating clips for the gaze interactions. For example, a user can pick two people from the given profile pictures and then clips that involve those two profiles having a gaze interaction would appear. It can also work vice-versa. For example, there can be an option where users would like to know at what point a student is not making eye contact with anyone and they can analyze the reasons why through the new clips. Another recommendation we would also like to add is on the confidence level bar that we show with each individual clip, there can be a value next to the bar showing the percentage of how accurate the AI is rather than users having to guess based on how empty or full the bar is. One last recommendation to our recommendation is producing the confidence level of the gaze interaction and how accurate the AI is when analyzing that. Lastly, we would also like to recommend a tools and tips tutorial on how to navigate our new UI. With these wide range of recommendations, we have no doubt that this project can be used for great things and will be beneficial and more understandable for its users.

5.2 Conclusion

Many people in the NSF and in our computer and data science department believe that visualizing data and events will be the newest and most efficient form of training people. At the moment, there is the COILS website demonstration that is the starting point for all new types of visualization techniques. In order for COILS to open its doors to new ways to visualize data, we hoped that our technique of using video clips could give those on this research project a new perspective on data representation.

There were many different aspects that we had to verify and extract when it came to creating a new visualization. Although the front-end UI is the main area that users are seeing change in the project, the back end and the data organizations are getting a huge change as well. Those changes can be similar to the changes that we had such as creating new files, omitting some information, changing one or two values, concatenating values, or even converting the bigger picture into smaller pieces. Once users extract what information they need, that is when the visualizing of that data can start. We suggested video clips would be the best representation to visualize the peaks of when emotions are being expressed because it would help our users understand when and why they are expressing those emotions. Ideally, the new UI that we created for this project will help teachers have an easier time navigating through the COILS interface. With this type of visualization and training, it can provide understandable data visualization for anything and for anyone.

Bibliography

[1] H. Zeng, X. Shu, Y. Wang, Y. Wang, L. Zhang, T.-C. Pong, and H. Qu, “Emotioncues: Emotion-oriented visual summarization of classroom videos,” *IEEE transactions on visualization and computer graphics*, vol. 27, no. 7, pp. 3168–3181, 2020.

[2] (2018) Teacher bias: The elephant in the classroom. [Online]. Available: https://marcolearning.com/teacher-bias-the-elephant-in-the-classroom/

[3] K. Miller. (2021, oct) What to consider when creating data visualizations in excel. HBS Online. [Online]. Available: https://online.hbs.edu/blog/post/data-visualizations-in-excel

[4] Tableau, “What is data visualization? definition examples | tableau.” 2022. [Online]. Available: https://www.tableau.com/learn/articles/data-visualization

[5] K. Hepworth, “Editorial big data visualization: Promises pitfalls,” dec 2016.

[6] V. Verma. (2021, apr) Creating interactive visualizations using plotly in python. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/04/creating-interactive-visualizations-using-plotly-in-python/

[7] M. Bostock. (2012) Data driven documents. [Online]. Available: https://d3js.org/

[8] (2022) React – a javascript library for building user interfaces. [Online]. Available: https://reactjs.org/

[9]

[10] M. Chen, R. Botchen, R. Hashim, and D. Weiskopf, “Visual signatures in video visualization,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 5, 2006.

[11] P. Frohlich, S. Egger, R. Schatz, M. Mühlegger, K. Masuch, and B. Gardlo, “Qoe in 10 seconds: Are short video clip lengths sufficient for quality of experience assessment?” *International Workshop on Quality of Multimedia Experience, QoMEx*, 2012.

[12] M. Brehmer and T. Munzner, “A multi-level typology of abstract visualization tasks,” *IEEE transactions on visualization and computer graphics*, vol. 19, no. 12, pp. 2376–2385, 2013.

[13] J. Messier. (sept) Giving teachers a better lens to perceive classroom dynamics. Worcester Polytechnic Institute.