For my term project, I would like to build a program that has the capabilities to judge people’s moods based on their current facial expressions. My project would need user’s to have a webcam or an external camera. In addition, this project requires users to follow certain guidelines to ensure that the project works properly. Users must be in a well-lit area, do not cover their face, and are not wearing anything over their eyes, mouth, and face. Upon following these guidelines, my project will use the user’s eyes and mouth to determine their mood. The five moods that my program will be able to recognize are happy, sad, calm neutral, fear/stress, and angry/disgust.

Projects very similar to this one have been completed in the past. For instance, Carnegie Mellon University has developed a machine that uses a camera to determine people’s moods. This machine is very precise in detecting people’s top and bottom lips and eyes, and then uses the shape and direction of the these facial qualities to determine the users’ mood. A feature that I like about this machine is that it does not attempt to recognize all of a human’s emotions. Instead, it defines five emotions that it can detect, which are happy, sad, neutral, disgusted, surprised. I also would like to touch on five feasible moods that can be recognized, instead of trying to recognize an excess of moods. A feature that I enjoyed when using the machine was that it showed how much of a mood the user exhibiting by displaying a percentage out of a hundred. In my project, I would like to incorporate a similar, yet different feature. I want to display the next algorithmically close mood if the application yielded the wrong mood. However, one feature that I found quite extraneous was the number of blinks counter. The user’s number of blinks is not relevant to my project, which I decided not to include.

Another application was made in 2012 called PlayMood that used a phone’s front camera to determine people’s moods and based on people’s moods play music. This application is not in the app store anymore, so I was not able to extensively demo this application to determine how it worked. Although I did enjoy the idea of matching people’s moods to music, due to the time constraint of this project it may not be something that is feasible. One thing that I did not like about this application from what I read, however was that it was only able to detect three moods: happy, sad, and neutral. These are not the only moods of music and would be limiting users to small domain of music.

From these projects, I have determined what things I would like to include and what things I would not like to include in my term project. I have been able to determine what moods that I would like to collect and what data I would like to collect, in order to determine the user’s mood. One of the biggest differences that my project will have from both of these applications is that my project will not compute any data regarding the user’s mood unless the user has clicked on the button that takes a of picture of the user’s current facial expression. This will eliminate any inaccuracies that may arise because of sudden moments that the use might make. However, there will be a button that allows the users to take another picture, in this case does not accurately represent their current mood.

I plan to accomplish my project by using modules such as OpenCV, Tkinter, and PIL. OpenCV is the main module that I will be using throughout my project. This module will help me determine the different parts of a person’s face that I would like to use to gauge, especially the user’s eyes and mouth. I have used cascades, such as the haarcascade for frontal face and eyes, which give my program the information to detect what a face, mouth, and eyes look like. Using the data from these cascades, I have been able to find a face, mouth, and eyes from a webcam video and an image. However, from my testing I have found that finding these facial features from an image is more accurate than find these features from a video. This is because the webcam video refreshes its frames, so it is constantly updating the location of the user’s eyes and mouth. In order to fix this problem, I intend on allowing users to take pictures of their current facial expression by pressing a button, and then determining the location of the users’ mouth and eyes from this picture. Furthermore, to avoid strange bugs in my code, I plan to make sure that there is only one face, two eyes, and one mouth in these respective lists.

Then using the location of these facial features, I plan on computing the number of dark pixels in these regions. By counting the number of dark pixels in these regions I can determine if a mouth is opened or closed and how narrow or large people’s eyes are opened. This is because the size of these facial features can be determined by computing the number of dark pixels in this region. In addition, I plan making more precise outlines of people’s facial features. By making precise outlines, I can determine the shape of the eyes and the mouth, such as if the user is smiling or if the user is frowning.

Knowing how large a facial feature is opened and the shape of the features are the two metrics that are the most important to this project. This is because this data can, essentially, determine the mood of a person. A happy person will be defined by bright eyes, open mouth, and upward sloping lips, whereas a stressed person will be defined by open eyes, open mouth, and circular mouth. On the other hand small eyes, downward sloping lips, and closed mouth will define a sad person. Similarly, narrow eyes and narrow lips will define an angry person. Lastly, a neutral expression will be defined as straight lips and open eyes, but not too bright.

Once I finish the OpenCV portion of this application to detect mood, I plan on using Tkinter and PIL to make a user interface, with pictures from the Internet. This user interface will have instructions for the user to follow and will be completely inspired by Baymax from Disney’s Big Hero 6. Also, there will be a small button which if pressed will display the next algorithmically close mood if the previous mood was wrong.