2 Weeks prior starting the final project

            On October 10th we met up with Professor Martin. We all pitched our final project ideas to her. Shane’s idea was to create a program that tells you the elements of a image. He suggested that we can also work out way up to making a video that also finds elements each frame if time permitted. His idea is great for seeing statistics of how often a color is being used. Sometimes it can be boring seeing the same color repeatedly. Especially in cartoons. Writing this program will allow users to see how often a color shows up within a picture or video. Yao’s idea was to use an Arduino to make a product that can be used as an alternative source of energy. Yao mentioned how salt and water can be used for an alternative source of energy. Yao’s idea can be used for those who cannot afford an expensive portable charger. Alternative energy can not only be used for charging phones, but it can also be used to supply power to the house. If we ended up doing his project then it wouldn’t be used for something big, but rather small like to charge a phone. However, if we chose to carry this project on, we could one day make it into a business. Trush’s idea was to create an app that allows you to drag and drop code. His reasoning was that there are a lot of people that would like to create an app for business or personal reasons. However, most people often get discouraged and must buy numerous amount of books to learn how to code. Coding can be hard for some, so he stated that we can make it easier for them to understand. He mentioned that he wanted to create something like the program Alice. A user friendly way to learn how to code.  Professor Martin gave everyone a good resource for our project ideas.

Week 1

This week was our second time we met up. We discussed our ideas the first meeting. Second meeting was with Professor Martin, we all pitched our ideas and she gave feedback on how we can expand our ideas. Shane's idea was to create a way for a user to take pictures and see the elements. Yao's idea was to create an alternate source of energy and use Arduinos to operate. My idea was to create a drag and drop app that allows you to code with ease. The idea is sort of like the game designing application Alice. Today we narrowed it down to one project, and created deadlines for the future and some meeting dates. We have chosen Shane’s project about identifying elements of a certain picture or video. We thought it was a good project that can keep on getting enhanced. The possibilities with his project are endless. If we chose to start a business with his idea, we can not only create a bar graph with elements shown, but also go a step further and allow the user to point to a picture and have it display the elements.

Week 2

The first day we found links to image manipulation. It was very detailed and can be very useful. This one link we found: <https://www.google.com/url?q=https%3A%2F%2Fprocessing.org%3A8443%2Ftutorials%2Fpixels%2F&sa=D&sntz=1&usg=AFQjCNE54zfW7XzSV5zeYzoObC8WtFOh0Q>         has everything we can possibly need. It has the code which we can refer to when working on this project. We made good progress and almost began to start coding if it wasn't for our tight schedule. The link that we found can help us with color saturation (one of the main topics from that Disney Animation article that Professor Martin sent us. Also, we took a look at some courses because we plan on using a program called Matlab that is used primarily for image processing. Although we had to spend some out of our pockets, it was well worth the investment and can make our project much more reasonable with the resources this udemy course provides. The next meeting for week 2, nobody in our group had experience with MATLAB, Therefore, we needed a good foundation in order to start programming. We looked through numerous courses to help build up the foundation. After looking through a couple of courses on Udemy, we found one that attracted us. Just from looking at the course description and the content, it looked like it was very useful for our project. Unfortunately, only the first section came to use, the rest of the sections were primarily used for editing pictures. We were hoping that this course would show us how to plot data, create things such as a 3D graph, and help us for our idea on using our phone or any kind of camera to do certain things. We made sure that we finished the entire course before moving on. Trying to collect as much information. We ended up having to request a refund because even after we finished the entire course it did not come to use at all. We are currently looking for more courses out there on Udemy because we believe that Udemy offers lessons that are much more in-depth than you would find on YouTube.

Week 3

We continued to watch Matlab videos on Udemy that were for free. However, we wanted a course that was longer than just an hour and a half. Therefore, we used YouTube. We thought we wouldn't find anything useful because there isn't a way to build up a foundation. We would constantly have to watch videos that only have to do with things that we want. In other words, no video is going to completely give us the results we want. We have to gain some knowledge first and apply that knowledge to alter some code that we watch on YouTube to have it output what we imagine it. Matlab is not as popular as other programming languages, but it's good enough to have some videos out there to use. Matlab is easy to adapt to and the features it has to make it such a wonderful programming language to learn. We can create bar graphs, manipulate images, create 3D plots, do some interesting things such as have it detect your fingers, or face. It can be used to help create a more secure system as it can be used as an alternative way to lock a device.

On Tuesday 11/14, we had the 3D cluster up and running, the bar graph, and the image manipulated. Not only did we get out a program to did we get the bar graph to display the original image, but we also got it to show up for the manipulated image. Underneath both those images, we had the bar graph displayed on both the original and manipulated images. What we did after that was we took both the original and the manipulated image and did some calculations to find the difference between the two graph results. With results of the differences, we created a separate bar graph showing the variation of the number of colors from the original to the manipulated. The difference values were absolute, meaning the chart does not graph below 0. It would just take the most significant amount of one of the graphs and subtract it from the other. Below the bar graphs, we included one 3D figure. The graph displays 100 pixels. It presents the results obtained from the difference.

Finally, on Wednesday, we met up with Professor Martin. We showed her out finished product for the image portion of our project. We showed her the image manipulated, the bar graphs, and the 3D graph. She told us what we should do from here on out. She mentioned that we should look into some API’s, have the code converted to JAVA, and to look into the app called Disney Animated. Just after we were about to leave, she gave us Arduino boards that we can use for our project after we get the video portion done.

Week 4

This week, we have gotten a lot done. We researched how to get more colors into our program. We want more colors so customers can see a more detailed analysis. The reason we I say, customers, is because we are trying to sell this idea, not just created because it's a project that we will get graded on. If we treated like this project was to get a good grade on the final then we would have low expectations. If we treated it like we were selling this, then we would be able to put more effort. So to make it a product worth selling, we included more colors to get a more detailed analysis. So far what we did was, we researched the RGB values of different colors, then we looked at the proportions of each RGB value to get it to display specific colors. Yao did a very nice job while researching the different color values. All we had to do essentially was to find more colors to add and then use the proportions. Wasn't too difficult, but was time-consuming as we had to make sure the values were correct. Next, We contacted the creator of Disney Animation. He was able to show us how he got the color line to show up for his video. It wasn't easy to get a response from someone who is very popular. However, with an extensive 2 hours just to come up with what to write on the email body, and subject, we were finally able to get his attention and give us part of his code! Here is the email we got back from him:

I think this blog post will answer all or most of your questions:

<http://home.theodoregray.com/blog/2013/08/13/a-visit-to-disneys-magic-kingdom?rq=lion>

(If you’re seen it already, go back and look again, because I just fixed all the broken links and videos.)

I used Mathematica, so my code would not work for you in Matlab, but in case it’s useful, this is a stripped-down version of the core function, which takes frames from the movie and identifies the color clusters, then plots them in 3D. The blog post explains how this is used to create the overall color maps. As you will see from the blog post, this is all about clustering, which is done by magic in the FindClusters command. I have no idea how that function works, because it’s built in to Mathematica.

data = Flatten[  
   ImageData[Import["Lion\_King.SceneColorMaps/077750-078003.png"]], 1];  
  
dataCut = data[[1 ;; -1 ;; 10]];  
  
clusters = FindClusters[data];  
  
colorGroups = Map[Mean, clusters];  
  
Graphics3D[{  
  Point[dataCut, VertexColors -> dataCut],   
  Map[{Apply[RGBColor, #], Sphere[#, 0.05]} &, colorGroups]}

Hope that helps,

Theodore

Thirdly, we had to get the video processing part done. Trush had the video manipulated since week 1, and had the video displayed on the figure. All we had to do now was to add the data analysis part to the program. Shane was able to get the original video to show up on the top left and the manipulated video to show up on the right. Below both video is a bar graph that shows the color proportions for every frame until the video was over. Always it calculated the differences live! Below the bar graphs, we had the 3D cluster to show up. It would show the 3D cluster being updated every frame for the original video and manipulated video. Also we included the color line variation in order to have it look similar to the Disney Animated app created by Theodore Grey. We struggled with this, but the man himself was able to help us create it! Here are the links to the Manipulated video Data Analysis: <https://www.youtube.com/watch?v=2DLCQYCy_aw> <https://www.youtube.com/watch?v=oD7uPBSo3xM>

Currently, we are working with Arduino in order to display color proportions received from MATLAB. Unfortunately, MATLAB requires us to purchase a subscription in order to let MATLAB communicate with Arduino. Therefore, we thought outside the box and got MATLAB to send a text file of the output, and have it sent to JAVA. Now that we got the proportions in JAVA, we can now allow JAVA and Arduino to communicate with each other. So far we have got the servo to spin at certain angles. All that remains is just to have the Arduino get actual data from JAVA and have it applied to Arduino. Trush got the Arduino to move at a certain degree, and Shane got the color proportions from MATLAB to Java. We sort of switched roles. Trush was good with Arduino and Shane was good with JAVA. So Trush was basically the lead programmer and Shane was the documenter.

Weeks 5&6

This week, we are just about to finish wrapping everything up. As of now we are just making sure that our code works and is ready to use during the presentation. We are currently working on getting the Arduino to take in a variable from Java and have it sent to the Arduino. Currently we are making an Arduino move at certain positions depending on numbers we send to a text variables. We are using the Processing IDE and Arduino IDE. The problem we are currently face is getting Arduino to read the text file. We must manually type in numbers ourselves. Just to summarize, we are trying to get a text file that only contains ONE number. Have the processing app read it and then have Arduino move at certain positions.

Here is what we have tried so far:

1.Delete the if statement with the switch

2.Changed myport ('0') to myport ('9') since that's where our wire is connected

3.Tried to use ByteRead on Arduino but it does nothing

4.Tried to convert ByteRead into an int instead of a byte

We have checked on Google, but no luck the only link we found was this <http://arduinobasics.blogspot.com/2012/05/reading-from-text-file-and-sending-to.html>

Processing IDE Code: <https://pastebin.com/xtFF1rJd>

Arduino code: <https://pastebin.com/BShcNRg5>

The problem as of now is having the USB port that connects Arduino to the PC. Only one of these can run that port at a time. Note that Arduino UNO, Nano etc. with ATmega328 chip have only one hardware serial. In this case we must use some software UART emulation to get the second serial port to work. Finally, we are going to do a presentation on Friday.