

Sabrina Bergsten

Dr. Rivas

Software Development 1

Final Project Milestone

Photo Filter Milestone

After a long period of consideration, I decided to base my final project off of the iconic filters I've seen incorporated onto so many pictures throughout history. Whether it be the history making obama filter¹ used throughout his entire first election in 2008, or the many 'filters' used by artists like Andy Warhol in his Marilyn Monroe pieces², filters and artistic concepts being applied to images have shaped history as we know it. For my project, I have been working on a program that allows a user to upload an image and my personally created filter will edit each pixel until the entire photo is transformed. To do this I plan to implement many imported classes to use the prewritten methods made already by fellow computer scientists, as to not 'recreate the wheel'. To utilize these methods such as, PaintComponent, from the java Graphics class, on each and every pixel, I know I'll be needing some kind of for loop that can run through every pixel in the image and add them all to a list. To do so, I've imported the java BufferedImage class. This class has methods like getRGB() that takes parameters, row and column, to get the red, green, and blue values of a pixel at the given (row and column) position. The method I created to read into each of these pixels is labeled readPixels and once run returns a two-dimensional array filled

¹ Barack Obama 'Hope' Poster, became the iconic campaign image for the first African American President of the United States: http://npg.si.edu/object/npg_NPG.2008.52

² Marilyn Monroe Series by Andy Warhol featured on Masterworks Fine Art: <https://www.masterworksfineart.com/educational-resources/andy-warhol/warhol-marilyn-monroe-series-1967/>

with each pixel in the given image along with their RGB values. I created it using nested for loops that first run through the row and then column of the image's pixels and then saving them into the new `pixel[][]` two dimensional array. Once both of the for loops have been completed and all of the pixel data has been saved to the 2d array, the method returns the `pixels [][]` array and all of the newly gathered information. I next plan to implement another method, called `applyFilter`. This method will be the one in which the filter is applied to the image and the user submitted photo is transformed. This method will take in the `pixels` array and another for loop will run through each element and direct each pixel in the array to an if-statement. This if statement will ask if the red, green, and blue values of a pixel are above a certain level, for example it might ask, if `pixel[j][k]` has a red value over 100 (out of 256), a green value over 150, and a blue value over 70. Once this method determines if a pixel meets this criteria it will change the RGB values to something of my choosing. Eventually, this method will print out the updated picture using a `paintComponent` method from the `Graphics` class, specifically the `drawImage` function that takes in your image, `xLocation`, `yLocation`, image width, and image height as parameters. The newly updated picture will signify the end of my program and if users want to try it with a new image they can simply run the program again. Since many, if not all, of these methods I plan to create have not been perfected yet, I plan to see the general plan for my photo filter program change and develop as I continue to realize my resources and limitations throughout the semester.

UML Diagram:

