## Final Project – Colorization Using Optimization

Site: https://www.cs.huji.ac.il/w~yweiss/Colorization/

## The Goal

The goal of this project is to create a program that utilizes colorization via optimization. This is done from the premise of, "neighboring pixels in space-time that have similar intensities should have similar colors.". This is implemented using a few functions that get neighboring pixels, calculate the weights needed, and colorize the pixels to get our output.

## The Functions

- Idx from coord
- Adjacent\_coords
- Transform\_images
- Calculate\_weights
- Populate
- Color\_image

The function, Idx\_from\_coord, is simple and just converts the coordinates given via parameters to the proper index using m\_dim.

The function, adjacent\_coords, find the proper adjacent coordinates to a point from the parameters inside of the dimensions.

The function, transform\_images, converts the gray scale and scribble image given to the proper color space so that the rest of the program can work properly.

The function calculate\_weights, calculates the weight for the given parameters and returns the normalized weights.

The function, populate, populates the matrix given as a parameter which is used later in the color\_images function.

Finally the color\_image function, is what will actually color the images given initially. After using the transform\_images function so that we can properly start, we initialize the needed matrices and vectors, calls the populate function so that they aren't empty, and solve for the color channels, and combine them with the grayscale image to create the result of the colorized image.

The biggest issue I had while going about this was actually starting off since the directions weren't as simple as say a PA. But after some research and looking overs some other articles I ended up getting some headway and figuring out the rest piece by piece.

Results
Images from the site



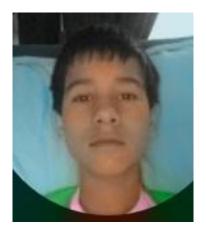




The following are my own images







Tyler Roosth CSCE 448 4/12/202<u>3</u>





This one ended up a little strange, but still worked out.

