Assignment 1

# General layout/design:

For this project I decided to use and old Facebook profile picture and isolate and vertical ellipse around myself as the subject. With area outside of the ellipse fading from all white to black at the furthest points from the center of the ellipse.

The actual ellipse utilized the size of the given photo to scale appropriately, I had to hard code in and offset from the center of the image to be sure to highlight the desired profile. Then using a helper function I scanned through the pixels to determine weather they lie inside the ellipse or not. If not the return value was then scaled as a variable multiplier of the pixels intensity. I spoofed the BGR into a grayscale by synchronizing the intensities of each color.

The final result is and image with a color image of myself on the inlay of a gradient oval shape. I enjoyed this exercise; regrettably I ran short on time to “enhance” the project but will have more time for the class overall in the near future.

A person taking a selfie

Description automatically generatedA picture containing person, indoor

Description automatically generated

# Source Code:

*"""  
Assignment 1 Test program takes input pic pup keep in color  
generates an elliptical portrait with gradient grayscale exterior  
"""*import os  
import numpy  
import cv2  
import math  
  
windowname = 'me'  
filename = 'outputMe.jpg'  
img = cv2.imread('prof pic.jpg')  
LENGTH, WIDTH, COLORS = img.shape # global variables of imported pic  
  
  
def ellipseFunction(x, y):  
 *""" helper function takes a pixel location as a co-ordinate and returns  
 the value of the equation (x-h)^2/a^2 + (y-k)^2/b^2 the ellipse equation"""* h = LENGTH/2  
 k = WIDTH/3  
 a = ((x-k)\*\*2)/((WIDTH/3)\*\*2)  
 b = ((y-h)\*\*2)/(((7\*h)/8)\*\*2)  
 return a+b  
  
  
for i in range(LENGTH):  
 for j in range(WIDTH):  
 val = ellipseFunction(j, i)  
 if val > 1.0: # if return value of a given pixel is >1 the pixel isn't inside the ellipse  
 img[i, j, 0] = 255 - ((val - 1) \* 50) # using 255-max val given by ellipse we can gradient out  
 img[i, j, 1] = 255 - ((val - 1) \* 50) # the exterior of the ellipse to created a fake grayscale  
 img[i, j, 2] = 255 - ((val - 1) \* 50) # 50 hard coded in/could be found by entering 0,0 pixel in  
 # ellipseFunc and dividing 255 by given val  
  
cv2.imshow(windowname, img)  
cv2.imwrite(filename, img)  
print(LENGTH, WIDTH)  
cv2.waitKey()  
cv2.destroyAllWindows()