CS 6643 - Computer Vision, Spring 2021 **James Fishbaugh**

Project 1

Out: 2 / 11 / 2021
Due: 2 / 22 / 2021 (deadline: 11:55PM)

Late submissions: Late submissions result in 10% deduction for each day. The assignment will no longer be accepted 3 days after the deadline.

Grading: 25% for implementation and 75% for written report

Office hours:

		Mon	Tue	Wed	Thur	Fri
James Fishbaugh	jf146@nyu.ed	4-5 PM		4-5 PM		
Michael Lally	mfl340@nyu.ed		12-1 PM		10-11 AM	
Akshat Khare	ak7674@nyu.edu	2-3 PM				10-11 AM

Please read the instructions carefully. Note: we will not be running code. Rather, we will check your code to make sure your implementation is your own, and it matches your results. Your grade is primarily based on your written report. This means going beyond just showing results, but also describing them. You should produce a standalone lab report, describing results in enough detail for someone else (outside of class) to follow. **Please submit a single PDF/HTML with all code included as an appendix.**

A) Programming Questions

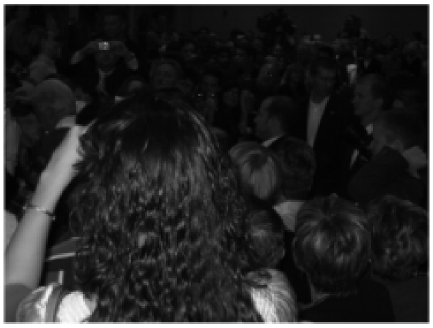
The purpose of this project is to get familiar with the coding environment for the rest of the course, reading/writing images, displaying images, and making graphs. You will implement histogram equalization.

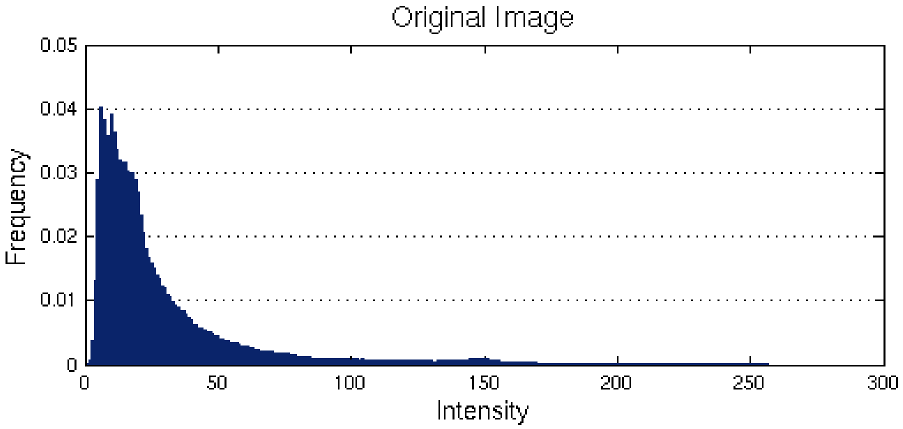
Hint for implementation: You can assume an intensity range [0-255] and hardcode the size of the histogram to be 256. You can make sure you have an image of type uint8 with min 0 and max 255 with the python code:

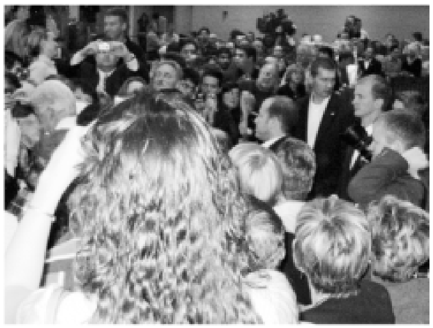
```python
# Convert the image to type uint8 and scale intensity values to the range 0-255
# *Note* numpy min/max flatten the 2D array, so you obtain the min/max of the entire image
im_uint8 = ((im - np.min(im)) * (1/(np.max(im) - np.min(im)) * 255)).astype('uint8')
```

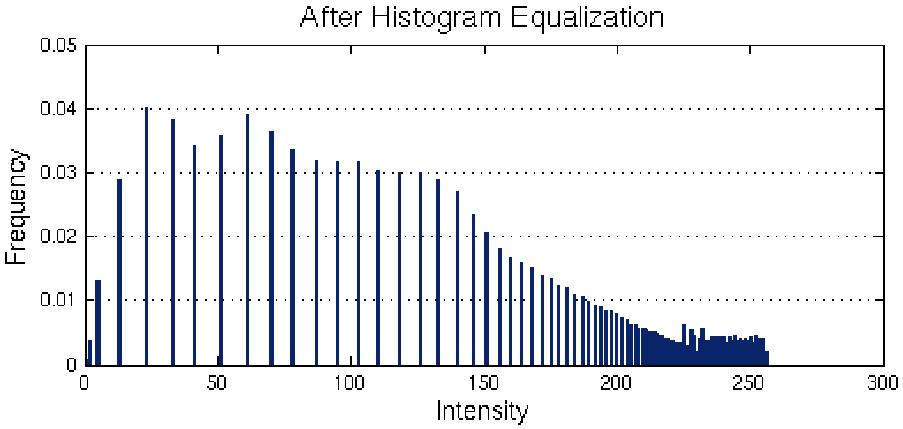
A1) Histogram Equalization

Implement histogram equalization and apply to image **crowd.png**, which you will find uploaded on NYUClasses. You may use the programming environment of your choice. However, we recommended Matlab, Python, and C++, as these are languages we can best support. You are encouraged to use libraries/built in functions to read/and write images and display figures and graphs, **but the rest of the implementation must be your own.** That specifically includes creating an image histogram (probability density function), computing the corresponding cumulative distribution function (CDF), and the creation of a new contrast adjusted image.









Implement the following functions:

```
def create_pdf(im_in):
        # Create normalized intensity histogram from an input image
        return pdf

def create_cdf(pdf):
        # Create the cumulative distribution function from an input pdf
        return cdf

def histogram_equalization(im_in):
        pdf = create_pdf(im_in) # Your previously implemented function
        cdf = create_cdf(pdf) # Your previously implemented function
        # Create a histogram equalized image using your computed cdf
        return equalized_im
```

Write up a report including the following:

- Include a brief introduction and description of how histogram equalization works.
- Show **crowd.png** before and after histogram equalization, and the corresponding histograms (PDFs).
- Discuss how the image and histogram have changed, and connect it back to your description in 1).
- Show the cumulative distribution function before and after histogram equalization on the same figure. Describe what you see. Explain the shape of each CDF and relate it back to image contrast and intensity histogram shape.
- Reapply the histogram equalization procedure on the corrected image. Show and discuss the results.
- Apply histogram equalization to another low contrast image (greyscale). Show and discuss the results.
- Histogram equalization is a global solution, modifying contrast with respect to the CDF of the entire image. Imagine you split the image into smaller regions (e.g. patches of 50x50 pixels) and apply histogram equalization to each local patch independently. **In your own words**, describe advantages and disadvantages of this strategy. There are no right or wrong answers here, the idea is to think critically about the method and answer honestly in your own words.
- Include all code as an appendix, i.e. copy and paste all your code at the end of your report.