**CS 7650 – Digital Image Processing**

Assignment 4G – Histogram Equalization and Histogram Matching

Kaitlyn Zahn

October 19, 2021

**Abstract**

The purpose of this assignment was to gain an understanding of histograms, histogram equalization, and histogram matching.

***PART A***

 a) Read an image (grayscale or color) and plot its histogram(s) in graph format with the x-axis of the graph representing gray value (ranging from 0 to 255) and the y-axis representing the frequency (or when normalized the probability) of occurrence of that grey value in the image. For color images plot three histograms one for each channel

 b) Support a different number of bins in displaying the histogram. For example, in part (a) the default binning would be 256 grayvalues for (each channel) of the image

c) Take the log of the image grayvalues (log can be to the base 10). Then display the transformed image with appropriate scaling such as multiplying by 100 and its new histogram

**Input Image**

**A person wearing a hat

Description automatically generated with medium confidenceA person wearing a hat

Description automatically generated with medium confidence**

**Output**

**Chart, line chart

Description automatically generatedChart, histogram

Description automatically generated**

**Chart, line chart, histogram

Description automatically generated**

**Chart, line chart

Description automatically generatedA person wearing a hat

Description automatically generated with medium confidence**

***PART B***

Apply histogram equalization to an input image. Display the original image and histogram equalized image

**Output**

**A couple of women wearing hats

Description automatically generated with low confidence**

**Chart, line chart

Description automatically generated**

**Observations**

Histogram equalization works the best when applied to images with a high color depth.

***PART C***

Apply histogram matching to two or more input images. Select one of the images as the preferred target image

*Not yet implemented*

**Input Image**

**Output**