Michael Cruz

December 4, 2023

Programming Assignment 5

**Experimental Design:**

The Otsu threshold algorithm was implemented in google colab. Images were imported as grayscale images, and then pixel intensity histograms were calculated using the OpenCV calcHist function. After calculation, the histograms were graphed with pyplot in order to analyze the data. The output of the calcHist function is a numpy array representing the number of pixels for each intensity value; this array was used in the Otsu algorithm to calculate the best threshold. The Otsu algorithm calculates the following equation in each iteration:

For 256 iterations, the algorithm calculates the intra class variance between two distributions. The distributions are determined by a threshold value u, that increments each iteration starting from 0. After the calculation, the variance calculated is compared to the previous best variance, and if determined to be higher, then the variables holding the best variance and best threshold values are updated. Once the algorithm finishes, the image is altered. All pixel values less than the determined threshold value are set to 0 and pixel values greater than the threshold are set to 255.

**Results:**

The algorithm was very successful in the binary segmentation of images. For most cases, the correct intensity threshold was calculated and the foreground and background of the image was identified. Some images were less successful in segmenting the foreground and background. Initially, it was presumed that this error was random, and possibly due to the algorithm. However, through experimentation it was discovered that images with histograms that had equal distributions between light and dark pixels performed better than images with either more white pixels or more dark pixels. This hypothesis was further proven by analyzing the histogram graphs and the resultant segmented image after the threshold was calculated. Even though equal representation of light and dark pixels alters the overall effectiveness of the algorithm, the results were still successful for the majority of images.

Trial 1: This trail was successful and the best threshold was calculated to be 96, which seems to be accurate judging by the histogram graphs

Histogram of pixel intensities

|  |  |
| --- | --- |

Original image Segmented image

|  |  |
| --- | --- |

Trial 2: Trial 2 was less successful. The two distributions for the image were much more skewed towards darker pixels than lighter pixels. The threshold was set at a pixel intensity of 40 and the segmented image is much less defined compared to the first trial

Histogram of pixel intensities

|  |  |
| --- | --- |

Original image Segmented image

|  |  |
| --- | --- |

Trial 3: This trial resulted in an image with clear segmented regions, unlike the previous trial. The intensity threshold was calculated to be 52.

Histogram of pixel intensities

|  |  |
| --- | --- |

Original image Segmented image

|  |  |
| --- | --- |