CMP_SC 7650: Digital Image Processing
Homework 1A: Point Processes in Python
By: Mikey Joyce
Due: 9/7/2023

Abstract:

Given an image of hela cells, there are many different colors present. To simplify the image, it would be better to convert these bright images down to a grayscale image (only shades of white, gray, and black). To simplify it even greater, the image can be converted to a binary format where a pixel is either fully black or white. Two modules were created, one was a module that can convert the color hela image into a grayscale hela image. The other module can convert the grayscale hela image into a binary hela image.

Introduction:

In this project 3 different experiments will be performed. The first experiment will be to convert *hela_cells_RGB.jpg* to a grayscale format. The second experiment will be to convert *hela_cells_gray.jpg* to a binary image format with threshold=60. This means that any pixel value below 60 will be converted to 0, which is black. Any value greater than or equal to 60 will be converted to 255, which is white. The last experiment will be to convert *hela_cells_gray.jpg* to a binary image format with threshold=140.

Experiments and Results:

For this project, the following input RGB image was given:

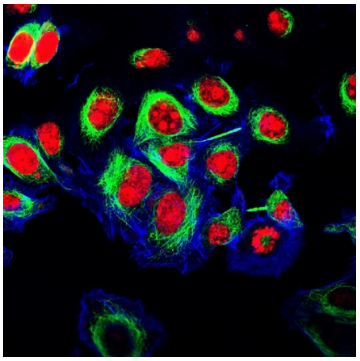

Figure 1 (hela_cells_RGB.jpg): RGB image of hela cells

For the first experiment, *hela_cells_RGB.jpg* needs to be converted to a grayscale format. The following equation was given to make this possible:

Gray[x][y] = (0.2989 * Red[x][y]) + (0.5871 * Green[x][y]) + (0.114 * Blue[x][y])

Utilizing the command:

```
python3 rgb2gray.py <input_file_path> <output_file_path>
```

The following image was produced:

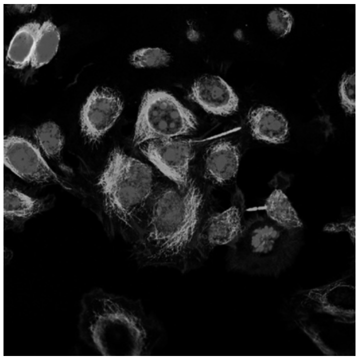

Figure 2 (hela_cells_gray.jpg): Grayscale image of hela cells

For the second experiment, *hela_cells_gray.jpg* needs to be converted to a binary image format. Choosing a threshold of 60, we will make this possible by converting any pixel value below 60 to black and any pixel value greater than or equal to 60 will be converted to white. Utilizing the command:

```
python3 gray2binary.py <input_file_path> <output_file_path> 60
```

The following image was produced:

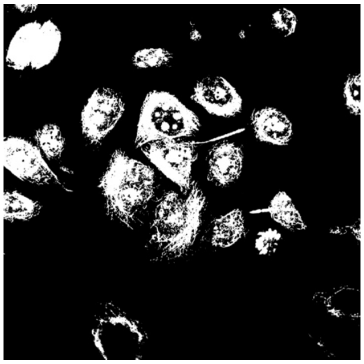

Figure 3 (hela_cells_binary_T=60.jpg): Binary image of hela cells with threshold=60

For the last experiment, *hela_cells_gray.jpg* needs to be converted to a binary image format. This time the threshold will be 140. Utilizing the command:

```
python3 gray2binary.py <input_file_path> <output_file_path> 140
```

The following image was produced:

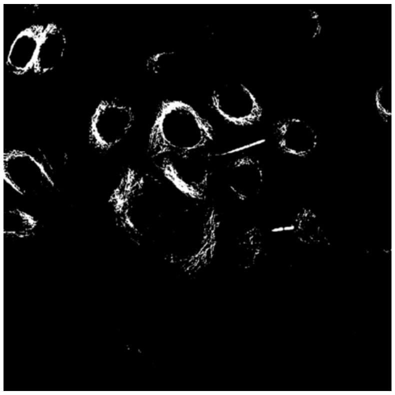

Figure 3 (hela_cells_binary_T=140.jpg): Binary image of hela cells with threshold=140

The difference between setting a threshold of 60 vs. a threshold of 140 is quite clear. The threshold of 60 allows for more white pixels to be present than the threshold of 140, which allows us to see more of the overall structure of the hela cells.

Conclusion:

In this homework assignment two module were created where an image could be manipulated. The first image manipulation consisted of converting an RGB image to a grayscale image. It was important that no library functions were utilized that way it was evident how on a low level these images are changing. The second image manipulation module consisted of converting a grayscale image to a binary image based on a given threshold. This assignment also got us familiar with the assignment structure that this class will be utilizing. Overall, this assignment was able to get me familiar with the OpenCV library and basic image manipulations.

References:

- Libraries and tools: PyCharm, OpenCV, NumPy, Matplotlib, Preview.
- DIP-CV-python-tutorial.pdf
- Lec03_IntensityTransformations.pdf