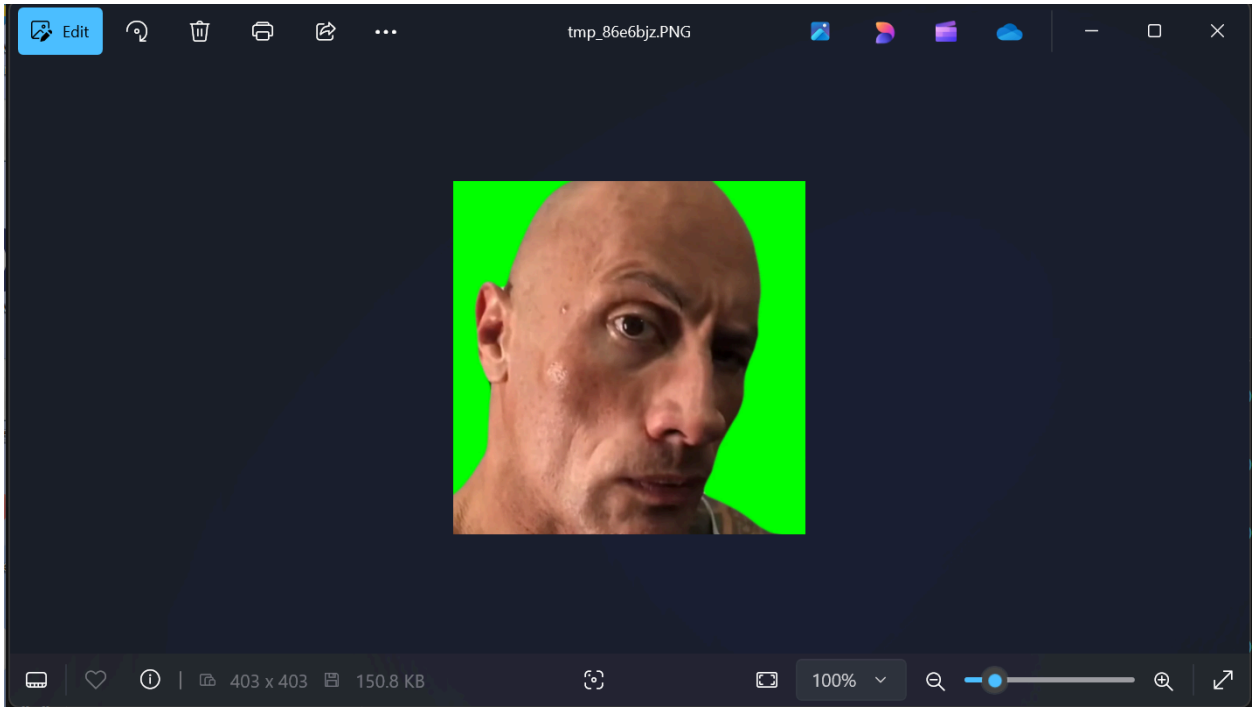
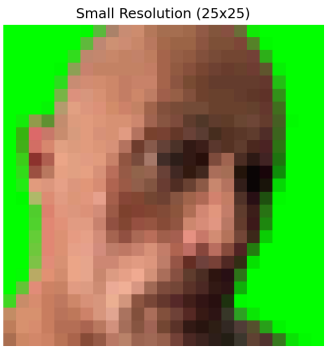
1. **Download a color digital image (or copy one from your phone). Read in the image using a python script, specifically use the PIL library.**

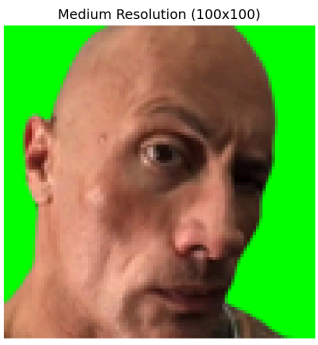
```
gbaks@gb_laptop MINGW64 ~/Downloads/cis365/assignments/assignment10-computer-vision (main)
$ python assignment10.py
Image loaded successfully!
Format: PNG, Size: (403, 403), Mode: RGBA
```

2. **Display the image to the screen.**

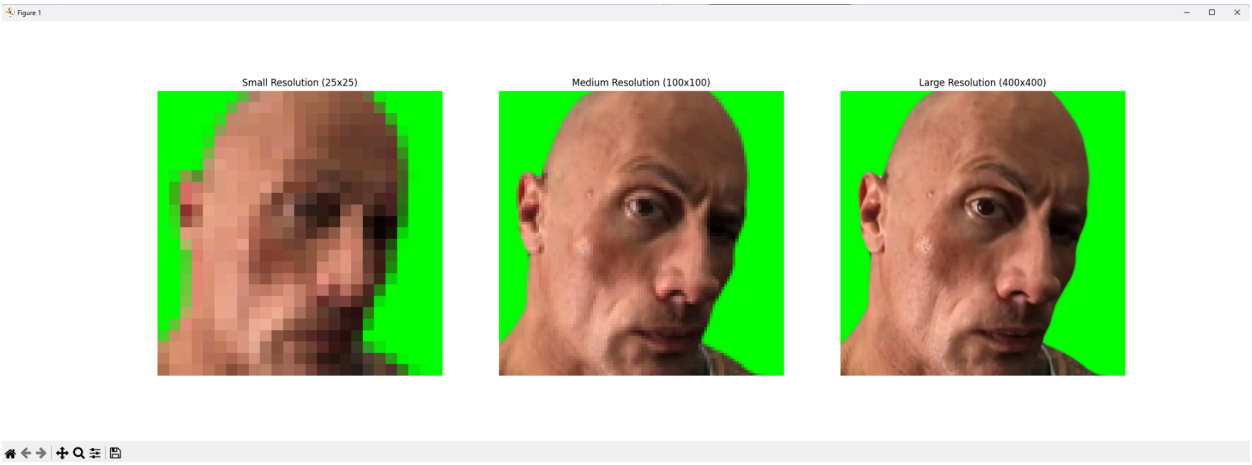


3. **Display the image at different resolutions, small, medium, large (you may select the actual dimensions). Display them on separate plots/windows.**

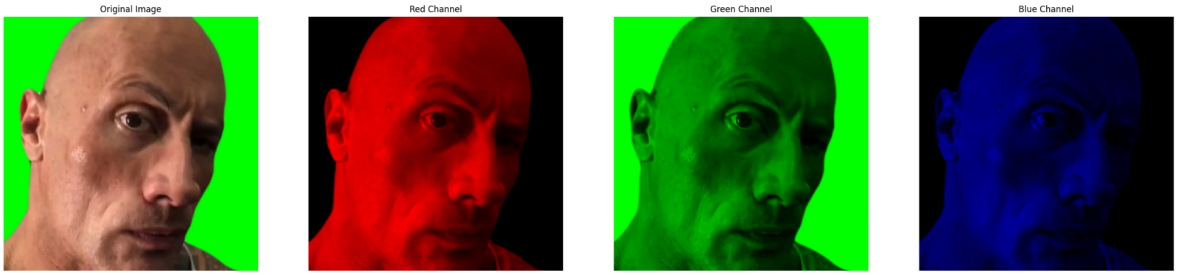




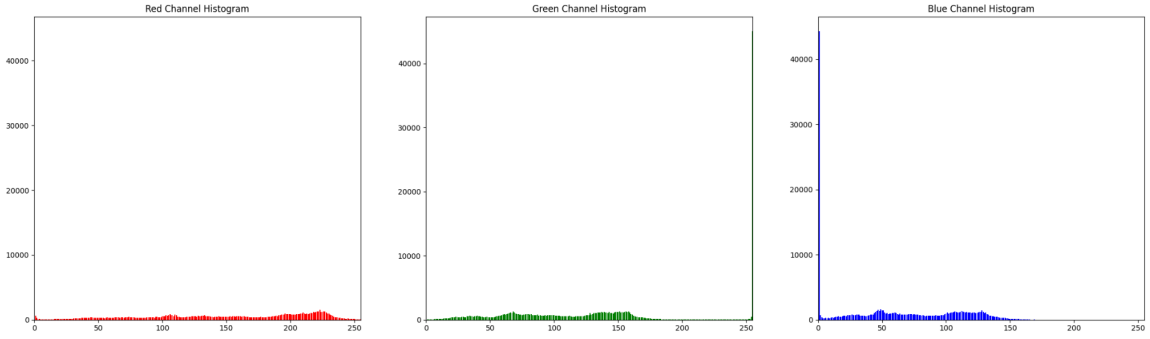
4. **Repeat step 3, but this time place all 3 images on a single plot/window.**



5. **What software library did you use to display the images?**
   For displaying the images, I used the `matplotlib` library. Specifically, I used its `pyplot` module, which provides functions like `imshow()` for displaying images
6. **Display the red, green,and blue channels along with the original image in the same display window(so 1 by 4)**



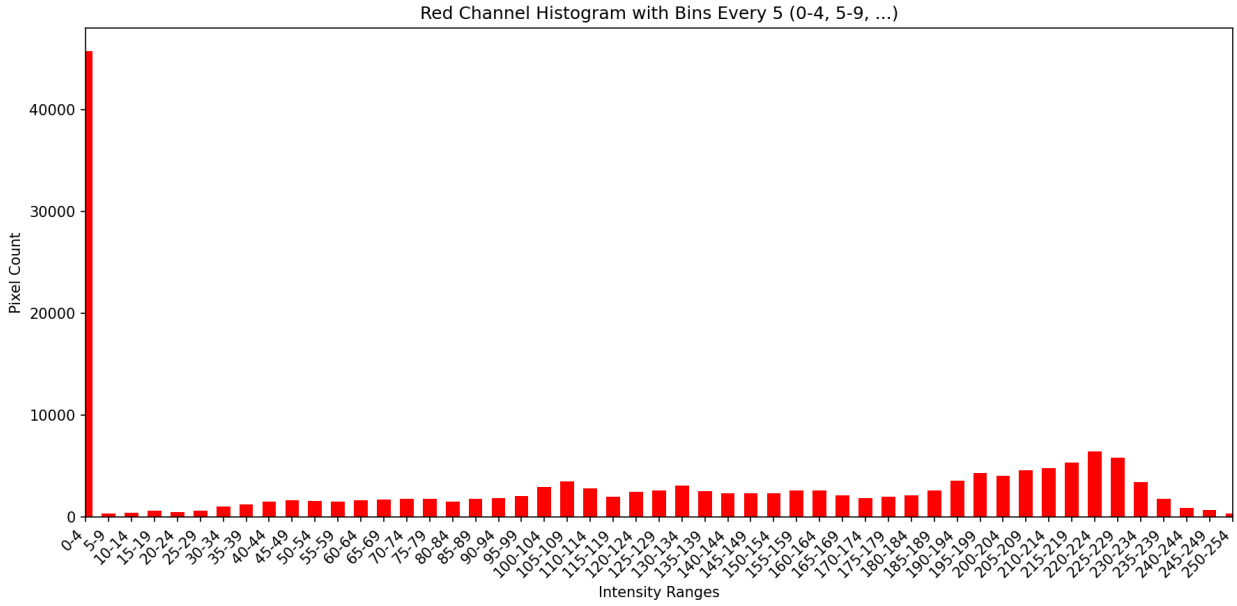
7. **Calculate the histogram manually (i.e., don't use a software library to calculate it) You can accomplish this by creating a double for loop that steps through the matrix (image) and counts the value for each channel. You should create a histogram for each of the 3 color channels (Red, Green, Blue). Display each of the histograms**

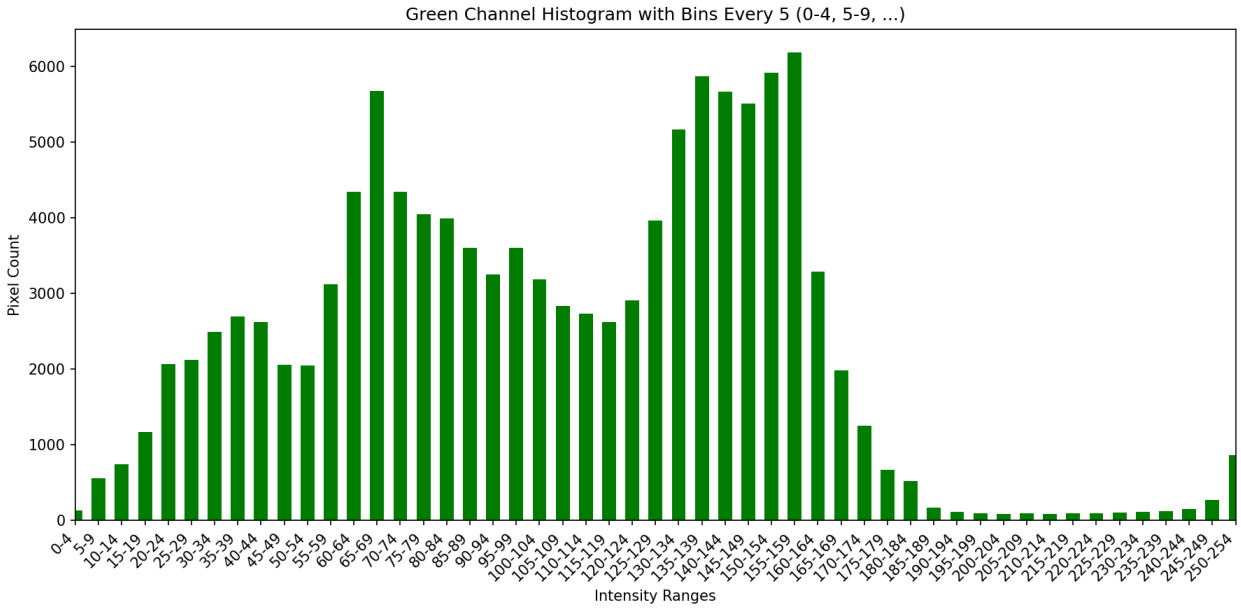


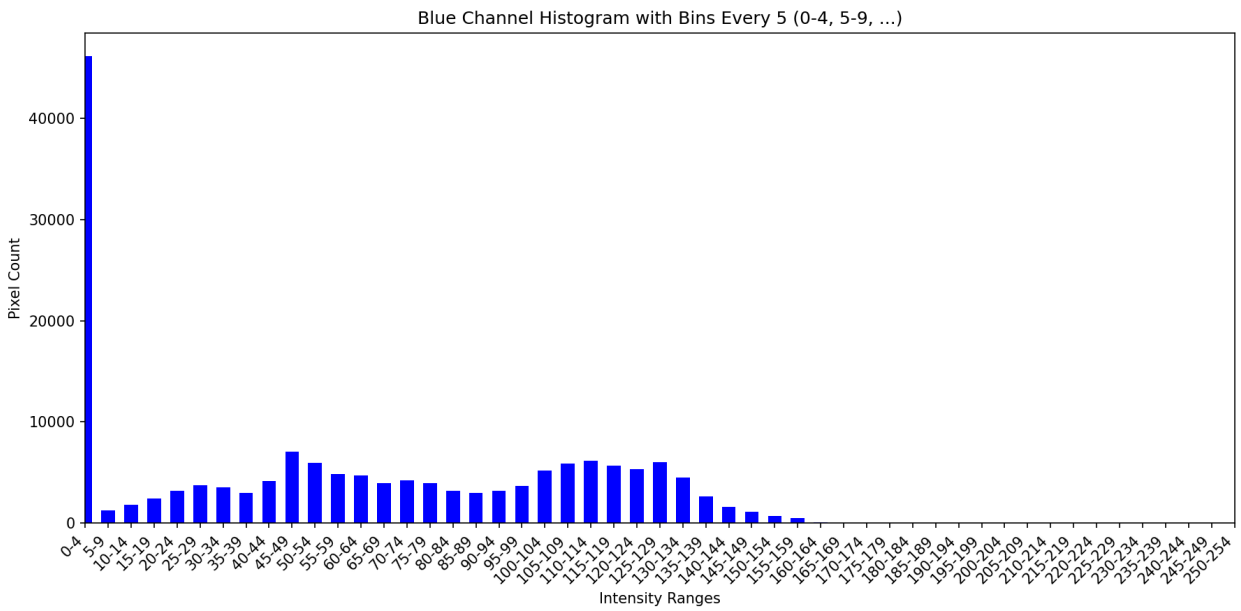
8. **Binning places all the values within a range inside a certain bin. For example, if the bin was from 0-4, all the image intensity values that fall in the range 0-4 would**

count forthat bin. Recalculate the histogram, but this time using bins 0-4, 5-9, 10-14, etc.

9. **Display the binned histogram for each of the colors.**







10. Create a custom 3x3 smoothing filter and apply it to the image. Output the resulting image and the original image in the same window. (Note: by custom I mean define what the 3x3 matrix is).