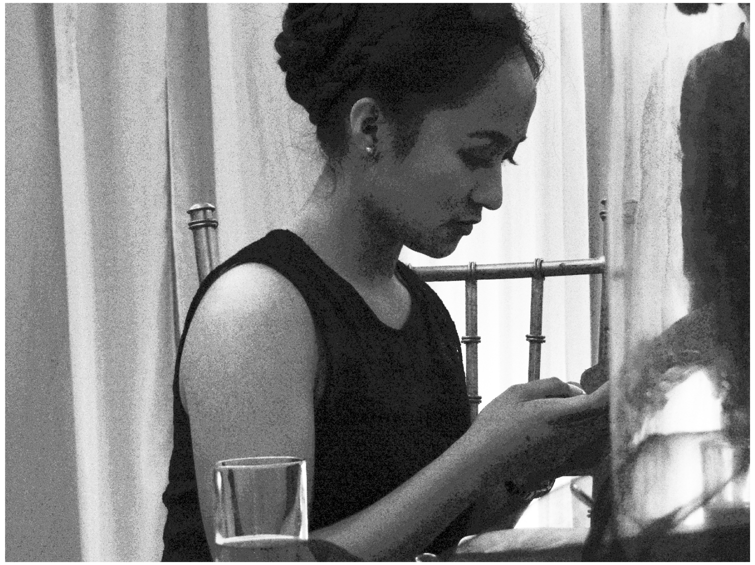
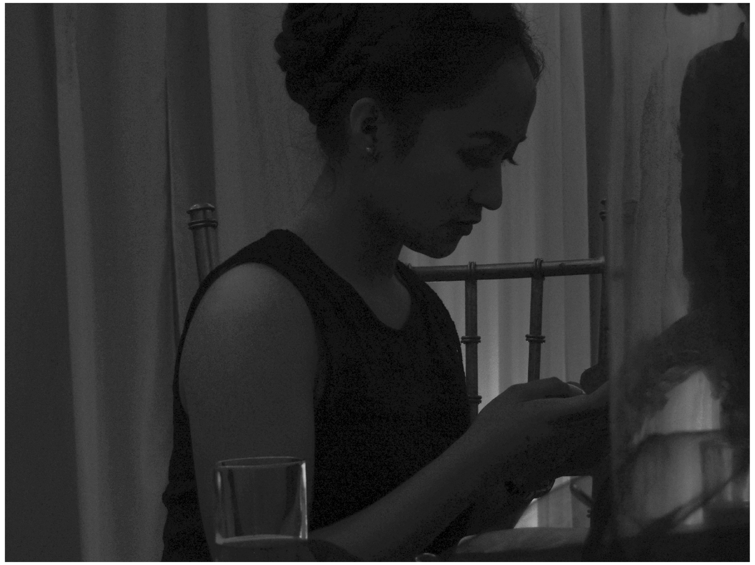
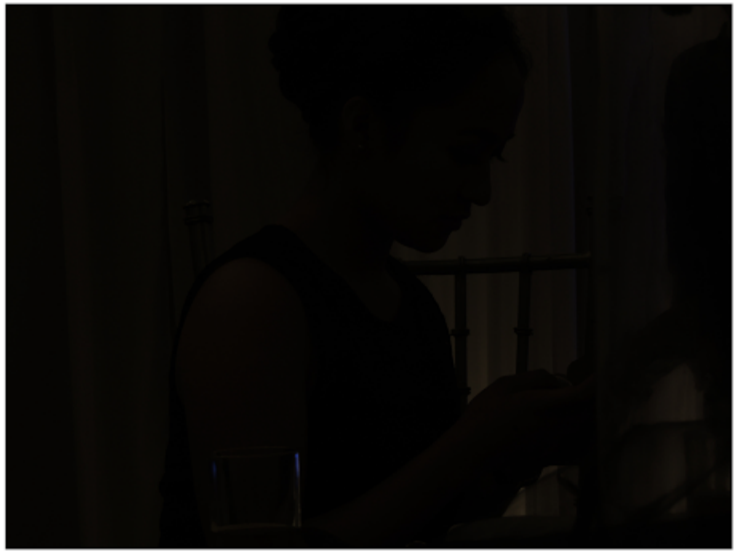
Activity 5 - Enhancement by Histogram Manipulation

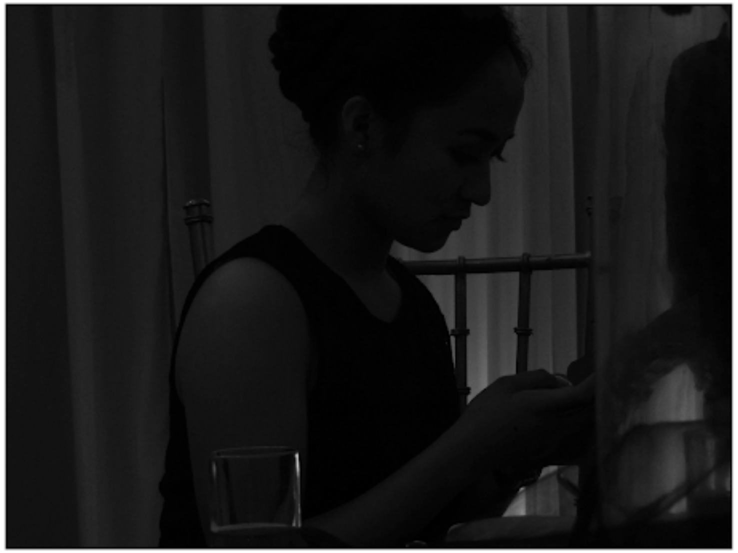
Kenneth M. Leo







Original RGB Image



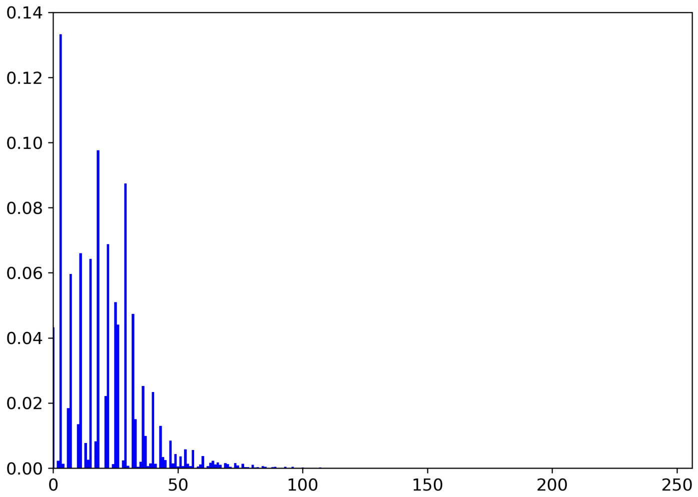
Original Grayscale Image

```
Global minimum: 0.0
Global maximum: 68.9188
```

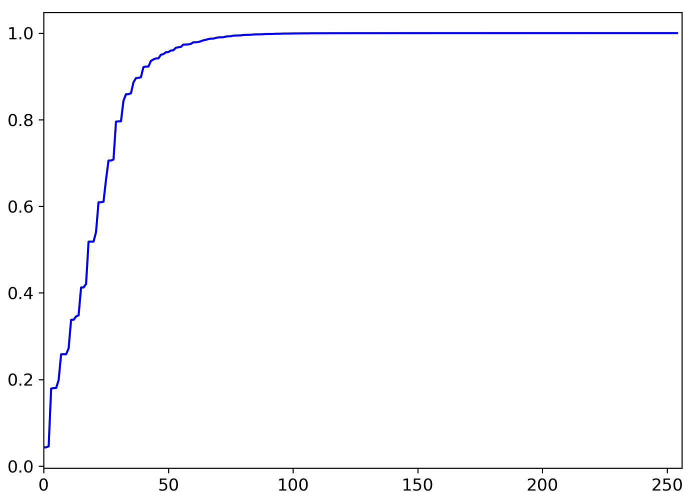
Stretched Grayscale Image

```
Global minimum: 0.0
Global maximum: 255.0
```

Looking at the original image, we can almost see nothing. Converting it to grayscale, we can already see more details because removed the bias caused by the RGB channels. Performing contrast stretching to the image will rescale the range of values to 0-255, from black to white.



PDF (normalized histogram) of the grayscale image

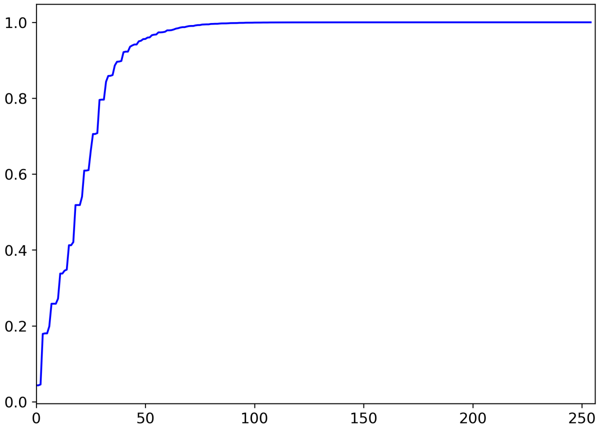


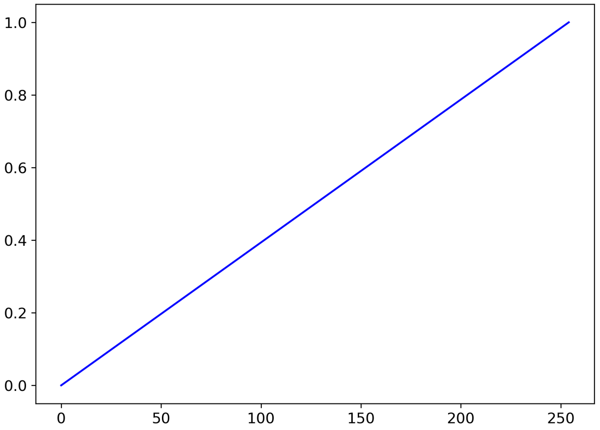
Normalized CDF of the grayscale image

We can see from the normalized histogram of the grayscale image that most of its values are less than 50, which corresponds to darkness. This is expected because the original image is very dark.

The figure on the left shows the cumulative distribution function of the histogram. The rapid change in the CDF located on the left side of the graph shows that most of the grayscale values are below 50.

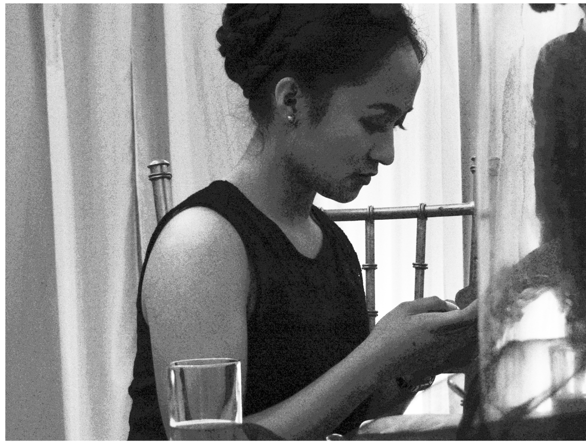
Desired CDF: Linear



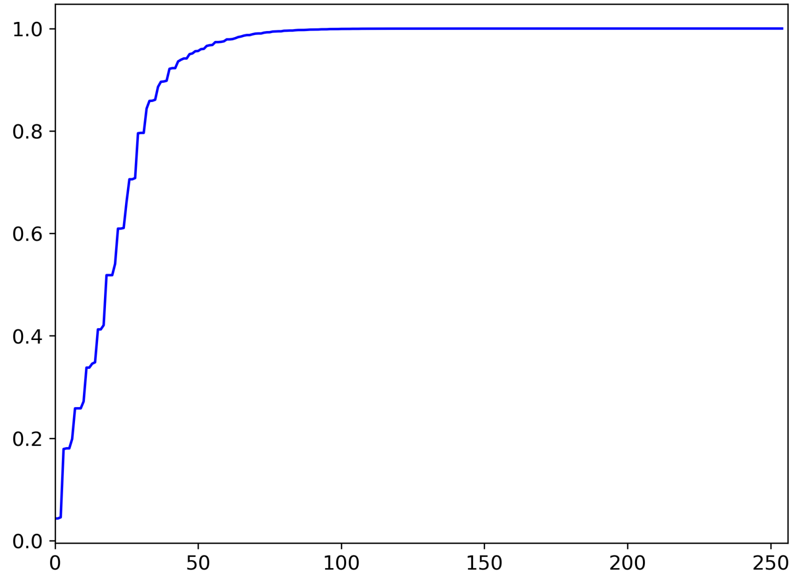


My ultimate goal is to see the details from my picture. And to do that I need to manipulate the histogram or CDF of the image. First I will use a linear CDF as my ‘desired’ CDF. A linear CDF corresponds to a uniform CDF, and I want to make the CDF of my image look uniform. I created the this by creating values from 0 to 255 that are linearly spaced and normalizing it by dividing it with 255.

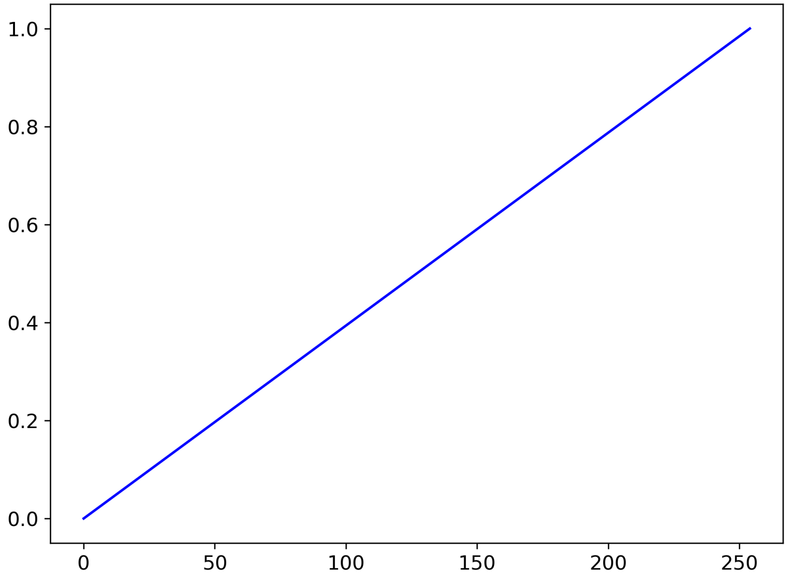
The two images on the right are the side by side comparison of the original image and the enhanced grayscale image using the linear CDF. We can already see some details that are not seen from the original grayscale image. We can now see the eyes and the ears.



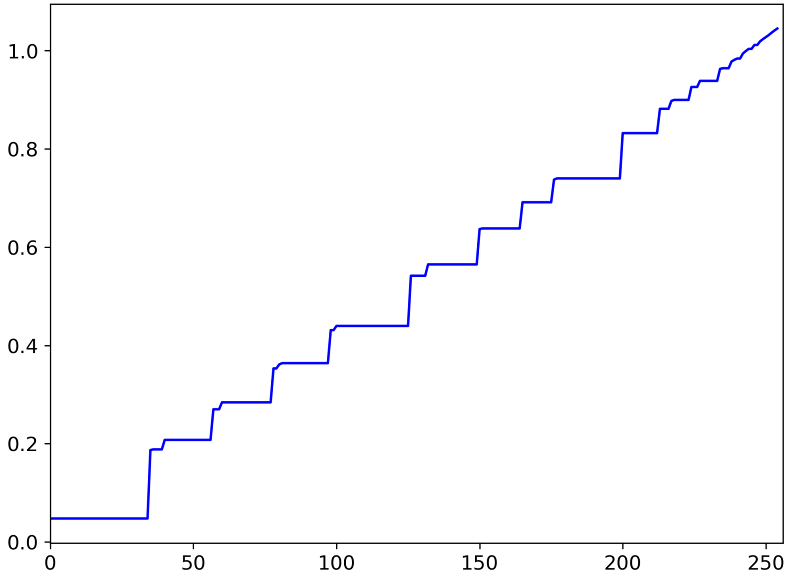
Desired CDF: Linear



CDF of Original Image

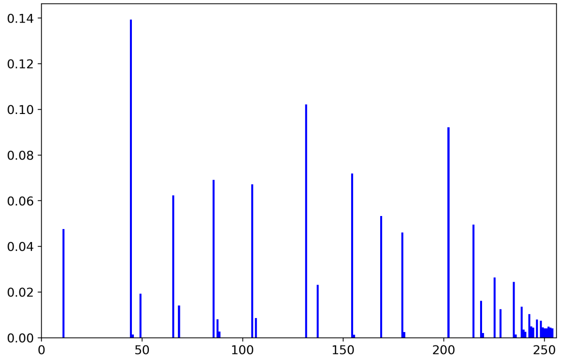


Desired linear CDF

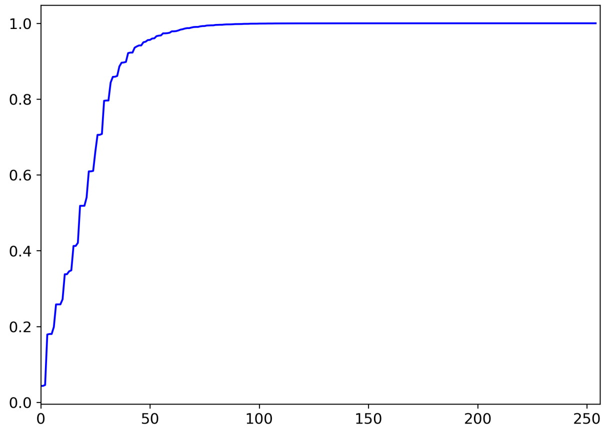


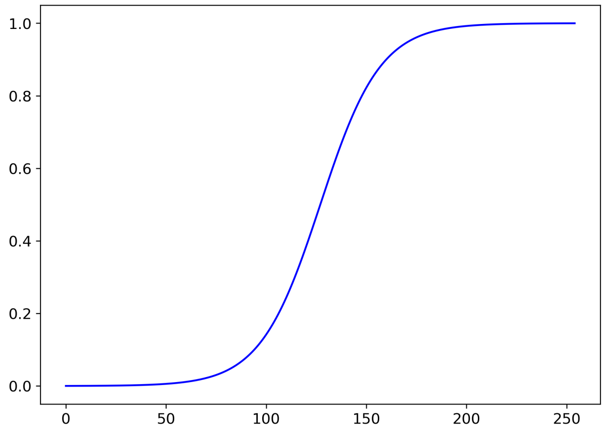
CDF of Resulting Image

We see that the CDF of the resulting image is what we expected, a linear-looking function. We can still see that there are sharp increase in the left part of the image. The flat parts of the CDF shows that there are no pixels in the image that have values in the flat areas.



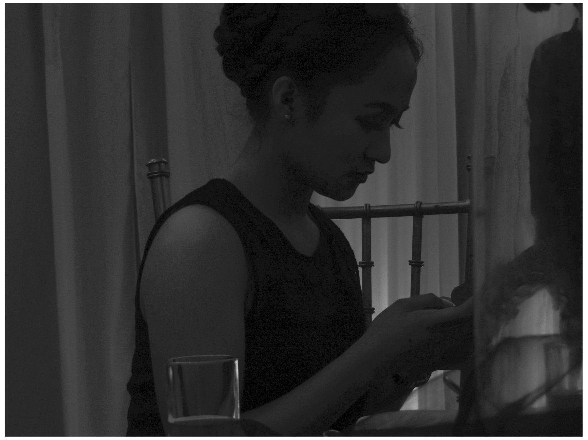
Desired CDF: Nonlinear (Sigmoid)



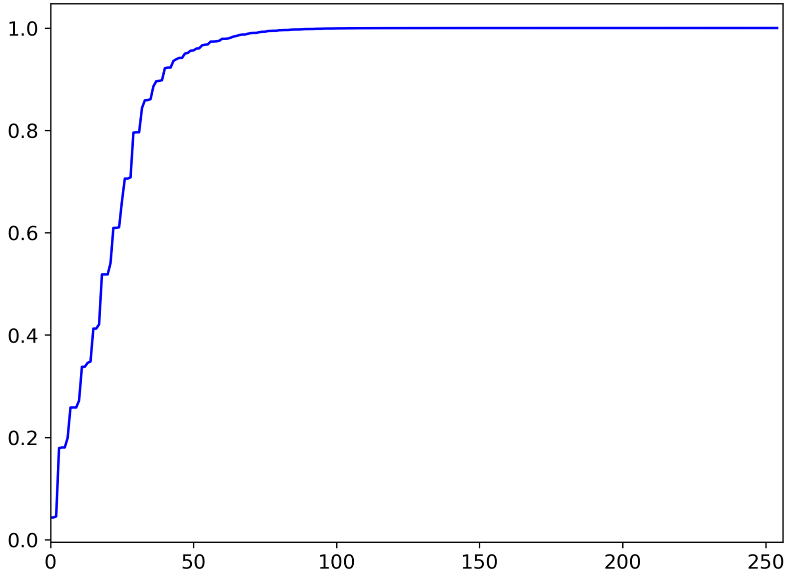


Next, I created a nonlinear function (a sigmoid function) to imitate the response of the human eye. First thing that I noticed is that the CDF of the sigmoid function is the same as the Gaussian, which means I can assume that the PDF of a sigmoid function also resembles that of a Gaussian. With this knowledge, I can hypothesize that the enhanced image will have grayscale values around 150.

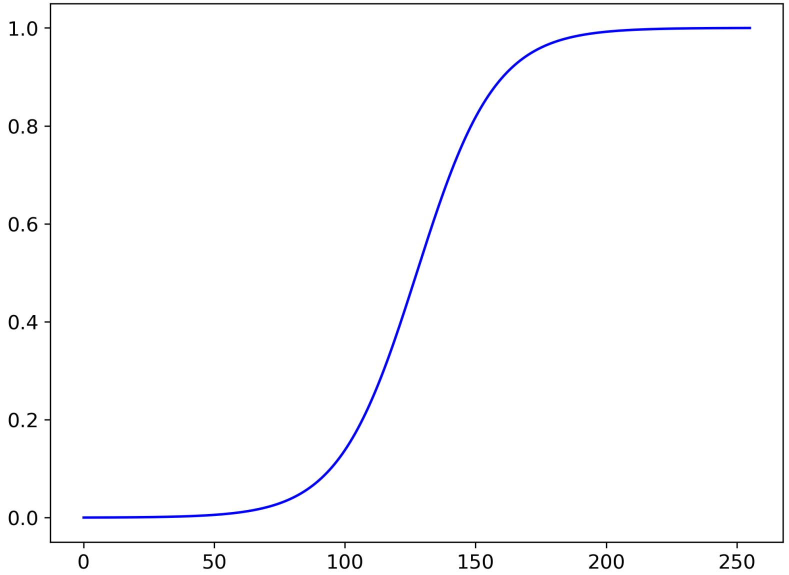
The two images on the right are the side by side comparison of the original image and the enhanced grayscale image using the sigmoid CDF. We can already see that the image has an overall gray appearance since looking at the histogram, there are high intensities in the middle grayscale values (gray area). This enhanced image is better than the image enhanced by linear CDF because you can see more details here.



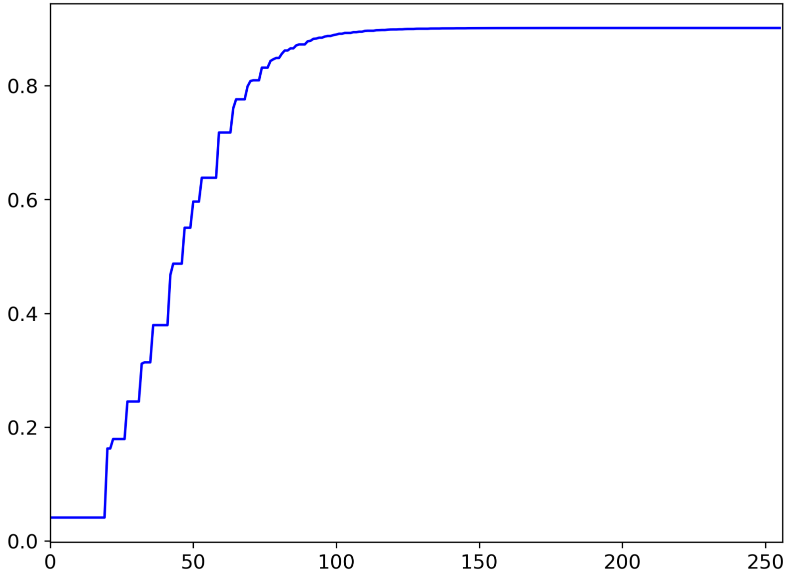
Desired CDF: Nonlinear (Sigmoid)



CDF of Original Image

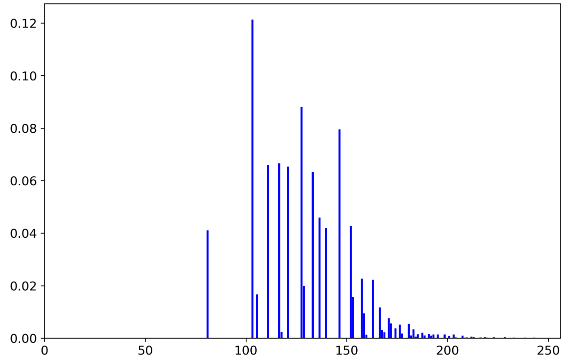


Desired sigmoid CDF



CDF of Resulting Image

We see that the resulting CDF somewhat resembles the desired CDF. We still see a sharp increase in the left side of the graph, which makes sense because the original image is really really dark.



Reflection and self-evaluation

I enjoyed doing the activity. I was able to learn new things like manipulate the histogram of an image to fix the contrast of the image. I also enjoyed this because I can see its real life applications. I think I might be able to use this to some of my photos that I took.’

Link to code used: http://tiny.cc/Activity5AP186

Technical correctness = 5 / 5

Quality of presentation = 5 / 5

Initiative = 1