AP 186 Activity 6
Enhancement of Color Images

Marc Jerrone Castro
2015-07420

Why do colored images degrade through time?

Old photographs or aged photographs usually degrade overtime due to exposure to UV light that break up the chemical bonds present within the dyes of a picture dye [1] .

Figure 1. Side-by-side comparison of an old and faded photograph with its intact original photograph.
SOURCE: St. Petersburg Photo Restoration & Retouching Services

[1] https://www.photoancestry.com/why-do-my-photos-fade-and-deteriorate.html

These photographs hold sentimental value to its owners and as such it is imperative to restore them.

In this study, we explore the use of three particular methods for restoring color to these photographs.

- Contrast Stretching
- Gray World Algorithm
- White Patch Algorithm

Contrast Stretching

Contrast Stretching

Contrast stretching is method which utilizes the maximum and minimum intensity values of a poor contrast image to correct and rescale the contrast of the image using the following equation,

$$I'(x,y) = \frac{I(x,y) - I_{min}}{I_{max} - I_{min}}$$

Input Image

The following image was selected from a large collection of photographs that my father had stored.

Fig 2. An old faded photograph of my father back in the mid-20th century. The image shown is a class photograph they took back in elementary school. As one may observe, the image is tainted with a red hue as it was left exposed to air for quite some time.

Using this input image of shape 502px by 702 px by 3. I extracted each channel using the following snippet of code.

```
img_red = img[:,:,0]
img_green = img[:,:,1]
img_blue = img[:,:,2]
```

Whereas, each individual channel would look something like this

Fig 2. An old faded photograph of my father back in the mid-20th century. The image shown is a class photograph they took back in elementary school. As one may observe, the image is tainted with a red hue as it was left exposed to air for quite some time.

Using this input image of shape 502px by 702 px by 3. I extracted each channel using the following snippet of code.

```
1 img_red = img[:,:,0]
2 img_green = img[:,:,1]
3 img_blue = img[:,:,2]
```

Whereas, each individual channel would look something like this

Fig 3. Red channel image extracted from the faded photograph. The graylevel of each pixel indicates how intense the red light must be in order to reconstruct the image.

Using this input image of shape 502px by 702 px by 3. I extracted each channel using the following snippet of code.

```
1 img_red = img[:,:,0]
2 img_green = img[:,:,1]
3 img_blue = img[:,:,2]
```

Whereas, each individual channel would look something like this

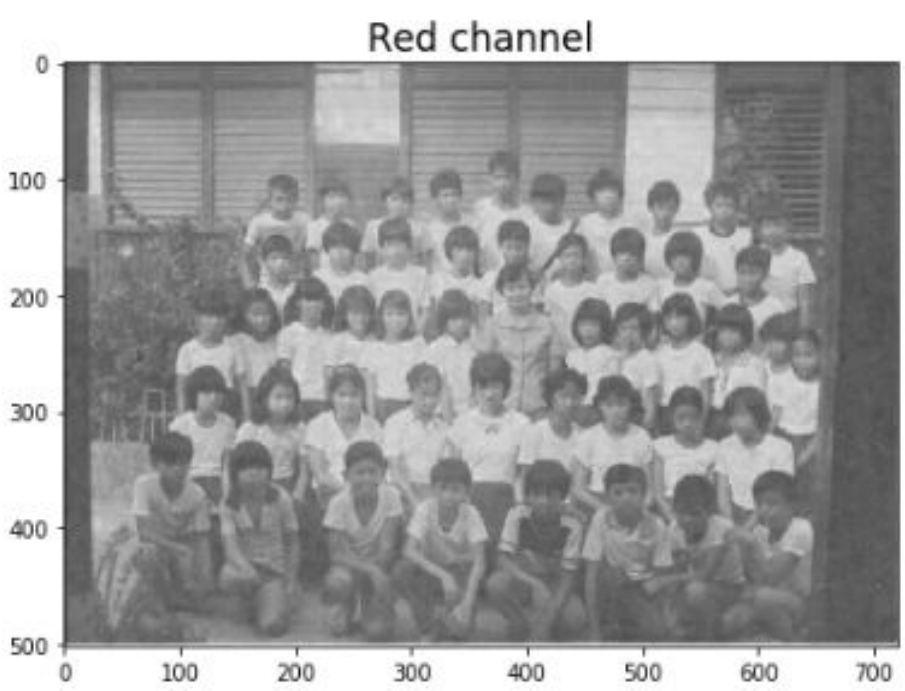
Fig 4. Green channel image extracted from the faded photograph. The graylevel of each pixel indicates how intense the green light must be in order to reconstruct the image.

Using this input image of shape 502px by 702 px by 3. I extracted each channel using the following snippet of code.

```
1 img_red = img[:,:,0]
2 img_green = img[:,:,1]
3 img_blue = img[:,:,2]
```

Whereas, each individual channel would look something like this

Fig 5. Blue channel image extracted from the faded photograph. The graylevel of each pixel indicates how intense the blue light must be in order to reconstruct the image.



Contrast Stretching

Afterwards by determining the minimum and maximum intensities of each pixel at each channel, we use the contrast stretching equation,

$$I'(x,y) = \frac{I(x,y) - I_{min}}{I_{max} - I_{min}}$$

To transform each color channel of the input image.

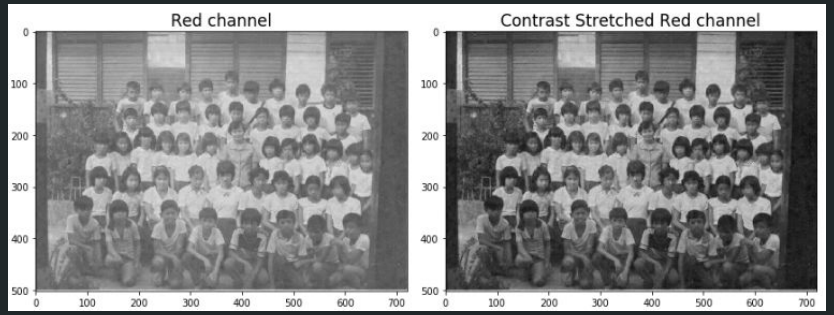
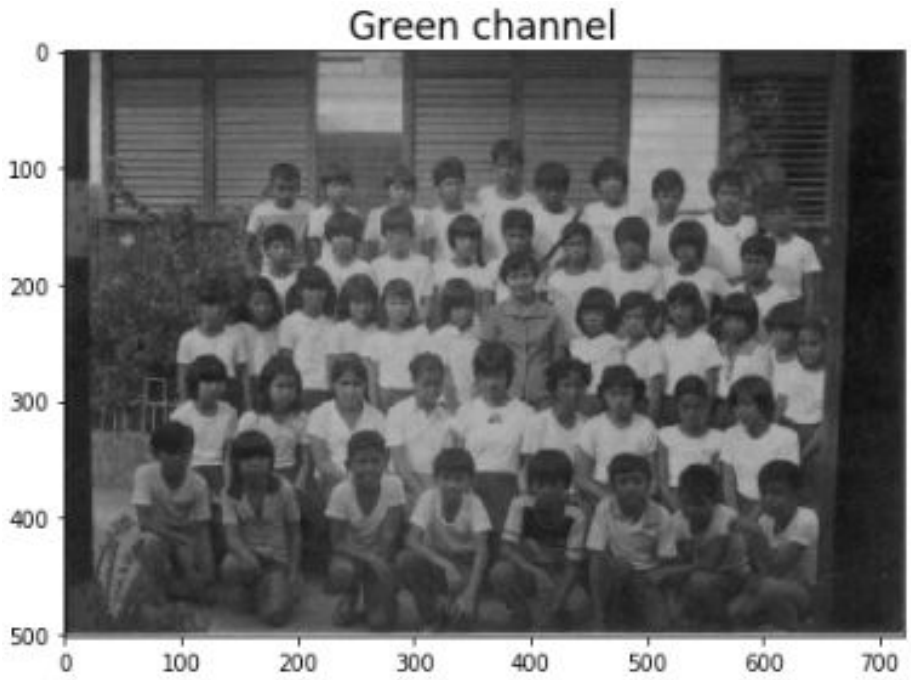
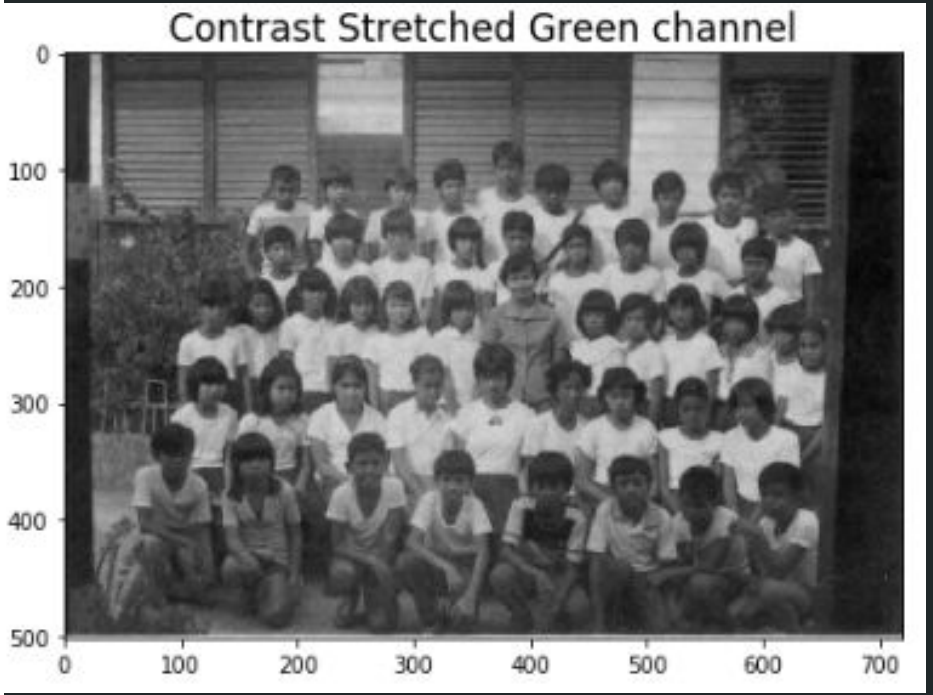


Fig 6. Comparison of red channel image with its contrast stretched image. As one may observe a significant difference in the intensities of the white color is seen. This is in addition to the contrast stretched image being a lot more clearer in comparison. One may attribute this to the input image's original hue being red and as a result the red filter would have a large significance in reconstructing the image.





Contrast Stretching

Afterwards by determining the minimum and maximum intensities of each pixel at each channel, we use the contrast stretching equation,

$$I'(x,y) = \frac{I(x,y) - I_{min}}{I_{max} - I_{min}}$$

To transform each color channel of the input image.

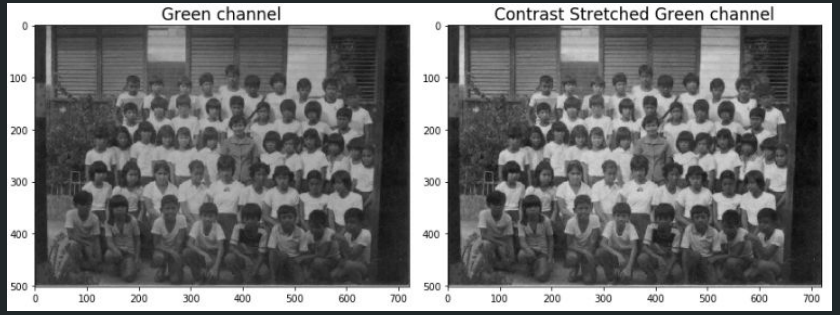
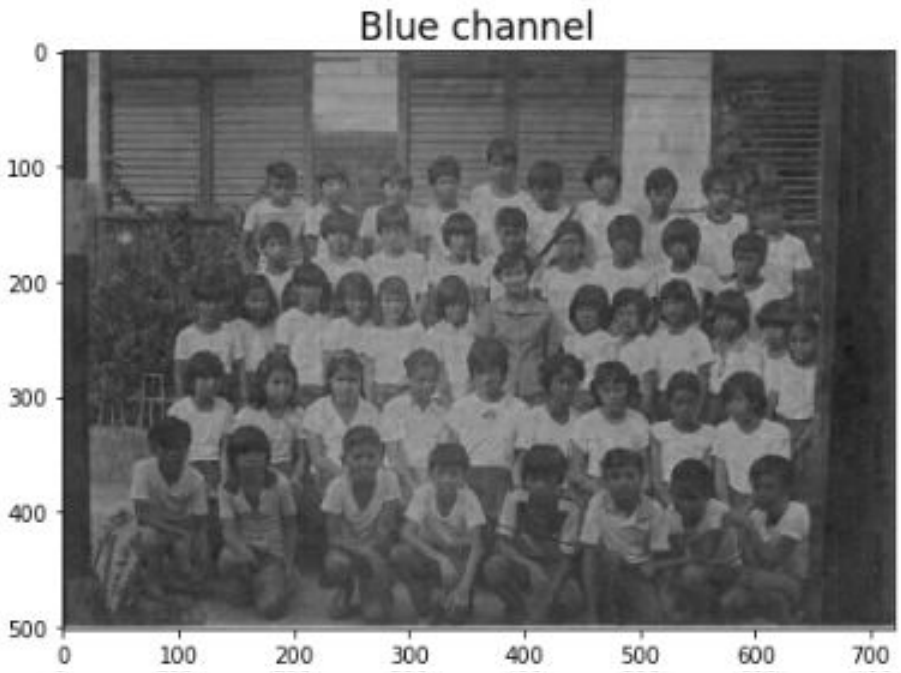
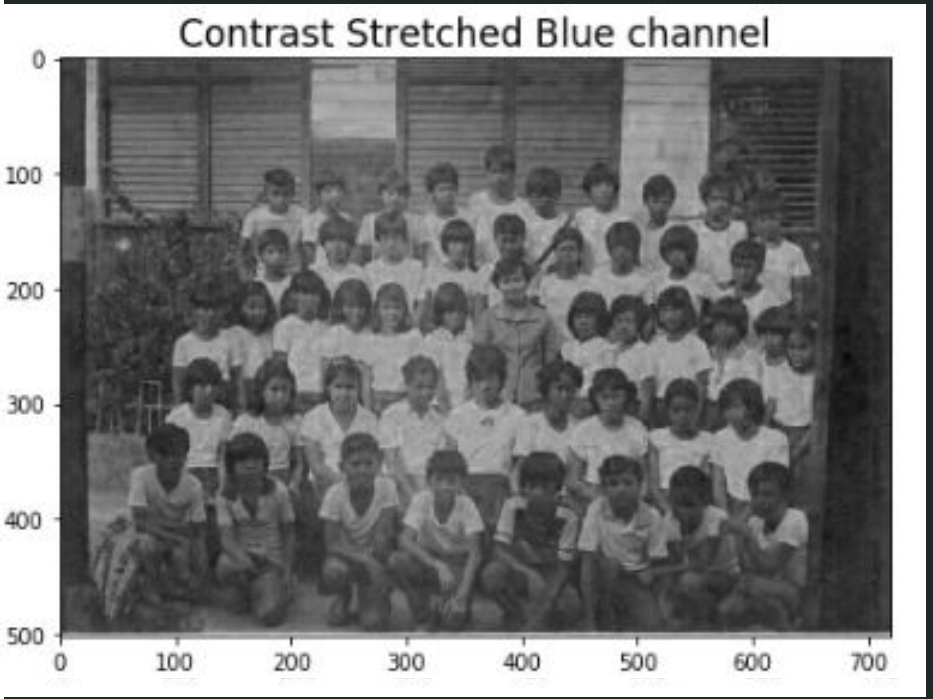


Fig 7. Comparison of green channel image with its contrast stretched image. As one may observe a significant difference in the intensities of the white color is seen. This is in addition to the contrast stretched image being a lot more clearer in comparison. Although in comparison to the red channel, the difference between the image on the left and the image on the right is not as large.





Contrast Stretching

Afterwards by determining the minimum and maximum intensities of each pixel at each channel, we use the contrast stretching equation,

$$I'(x,y) = \frac{I(x,y) - I_{min}}{I_{max} - I_{min}}$$

To transform each color channel of the input image.

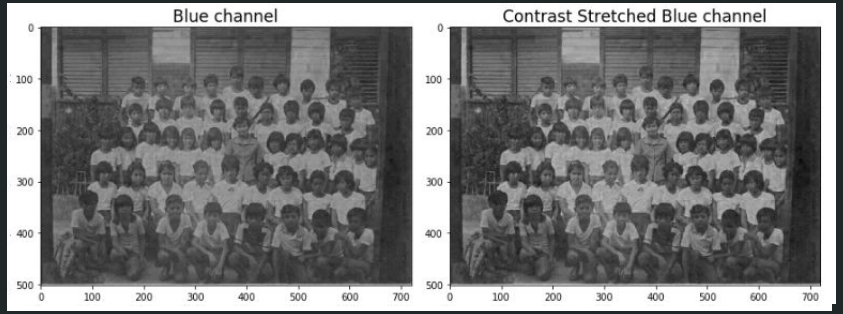


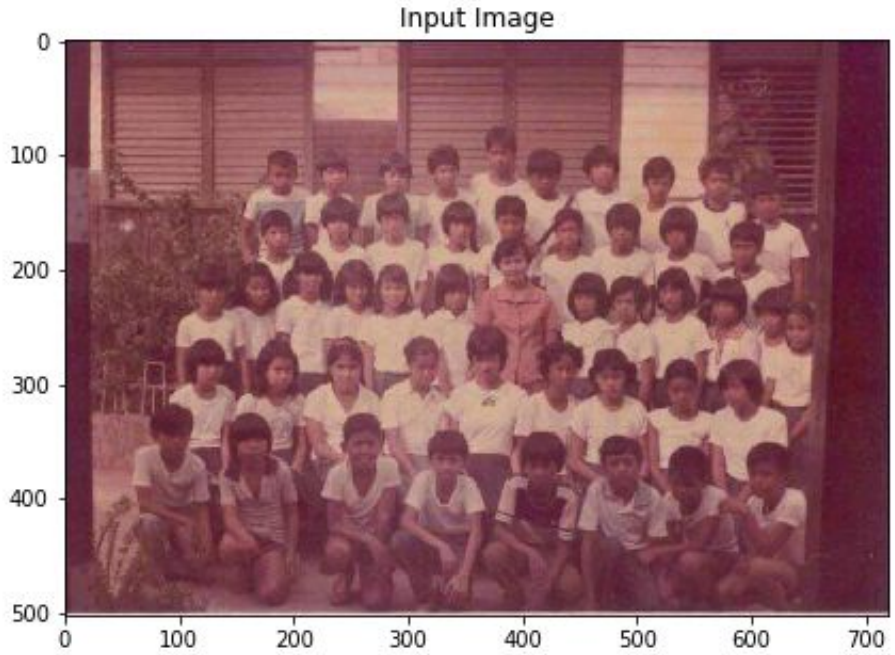
Fig 8. Comparison of blue channel image with its contrast stretched image. As one may observe, there is only a subtle difference between the image on the left in comparison to the image on the right. This may be attributed to the lack of blue shade within the original image and as such the difference between the contrast stretched image and the unstretched image is only minimal.

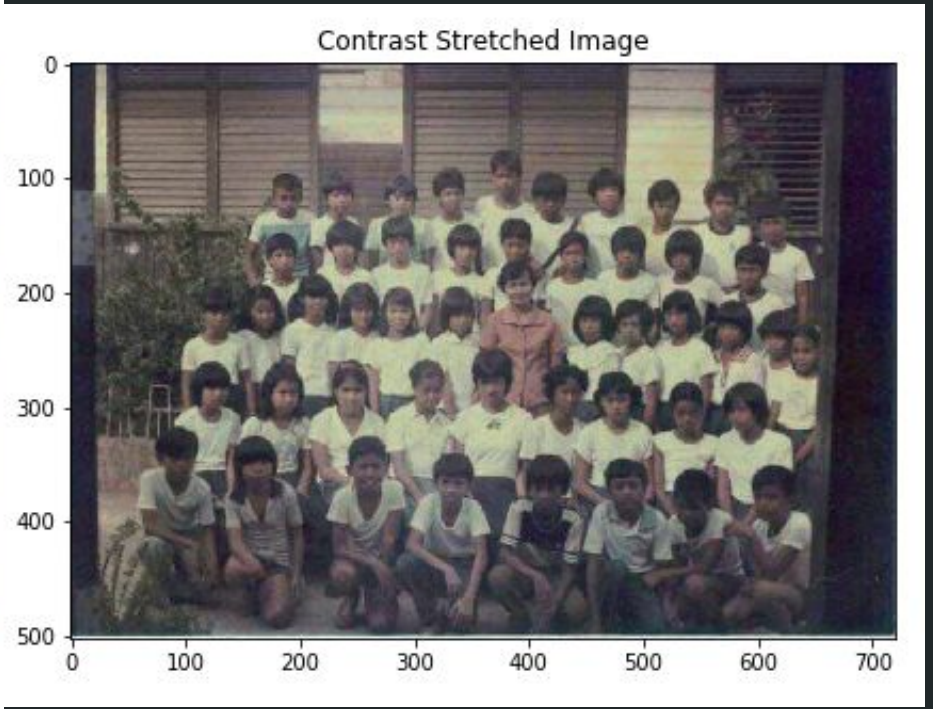
We then overlay each contrast stretched channel into one matrix using the following snippet of code,

```
img_CS = np.zeros([np.shape(img)[0],np.shape(img)[1],3], dtype = float)
img_CS[:,:,0] = imgred_cs
img_CS[:,:,1] = imggreen_cs
img_CS[:,:,2] = imgblue_cs
io.imshow(img_CS)
```

Which resulted to the following colored and restored image . A side by side comparison of the original image and the resulting image can be seen in the next slide.

Fig 9. Resulting image which resulted from overlaying the contrast stretched channels. As one may observe, the color of the image had been restored and is no longer tinted with a red hue. The white uniforms of the students, the red shirt of the teacher, as well as the green color of the leaves at the background had been restored.



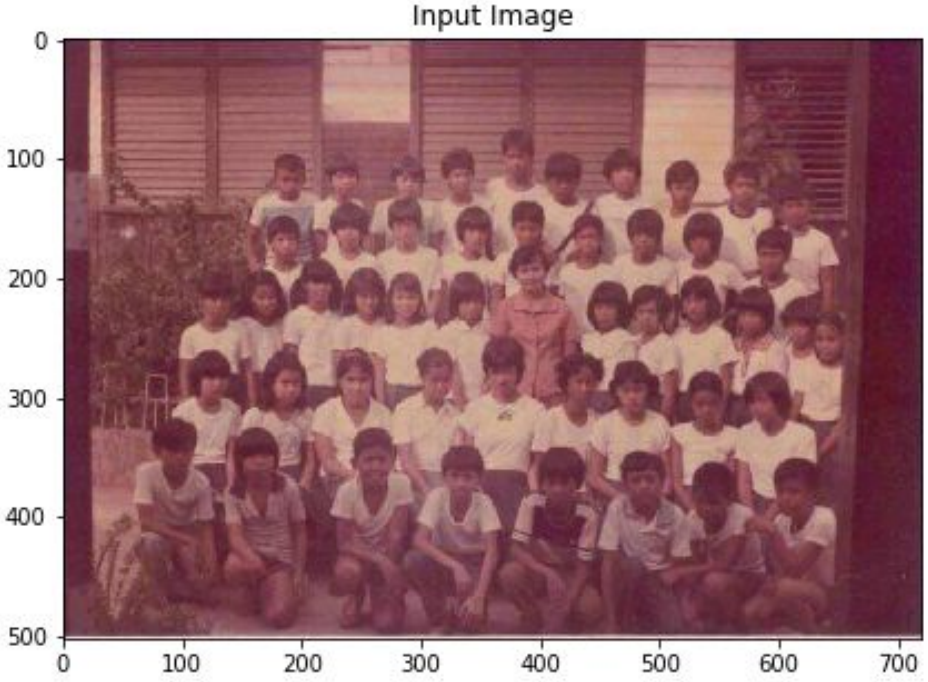


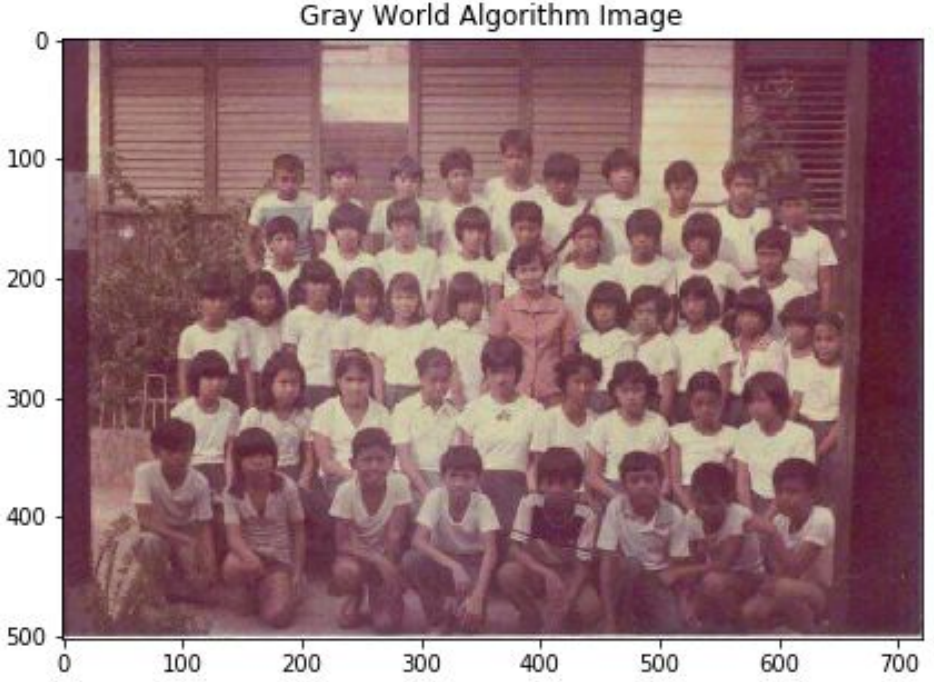
Contrast Stretching

Gray World Algorithm

Gray World Algorithm

The gray world algorithm utilizes the average value of each channel and afterwards scaling each pixel value such that each pixel in each channel is divided by their corresponding average value. Each channel is then normalized by dividing the maximum value of the transformed channel. Afterwards the normalized version of each transformed channel is overlayed as to recreate an RGB image.





Gray World Algorithm

We then overlay each normalized transformed channel into one matrix using the following snippet of code,

```
img_GWA = np.zeros([np.shape(img)[0],np.shape(img)[1],3], dtype = float)
img_GWA[:,:,0] = RGW/np.max(RGW)
img_GWA[:,:,1] = GGW/np.max(GGW)
img_GWA[:,:,2] = BGW/np.max(BGW)
io.imshow(img_GWA)
```

Which resulted to the following colored and restored image . A side by side comparison can be seen in Figure 10.

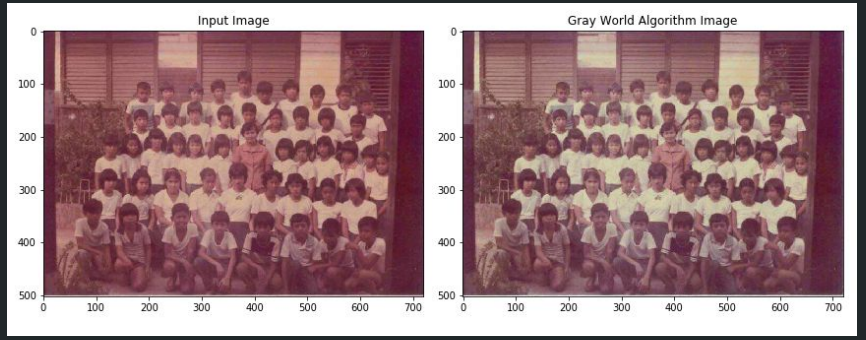


Fig 10. Resulting image which resulted from overlaying the normalized transformed channels. As one may observe, the color of the image had been slightly improved , whereas, the image is now slightly less red and is now able to show a better representation of the white uniforms. However in contrast to contrast stretching, the gray world algorithm performed slightly worse.

Gray World Algorithm

White Patch Algorithm

White Patch Algorithm

The white patch algorithm utilizes the known portions of an image which are colored white. By extracting a portion of the input image which is known to be that of the color white and determining the average intensities of each channel of that portion, one can transform each color channel by dividing these averages to each channel of the input image. A similar normalization scheme to that of the Gray World Algorithm is performed before overlaying each channel to recreate the image.

Input Image

We first select a portion of the original image which we know to be white.

Fig 11. An old faded photograph of my father back in the mid-90s. Highlighted with a red box is a "patch" of the image that we know to be white. We then use this to determine a scaling factor which we could use to enhance the color of our input image.

Input Image

We first select a portion of the original image which we know to be white.

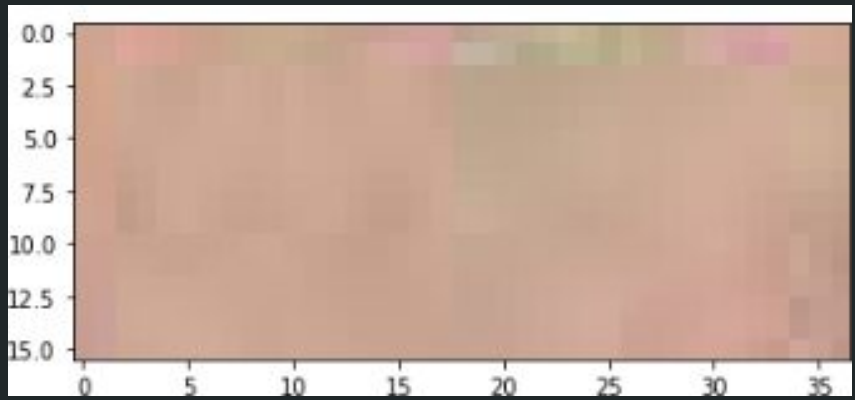
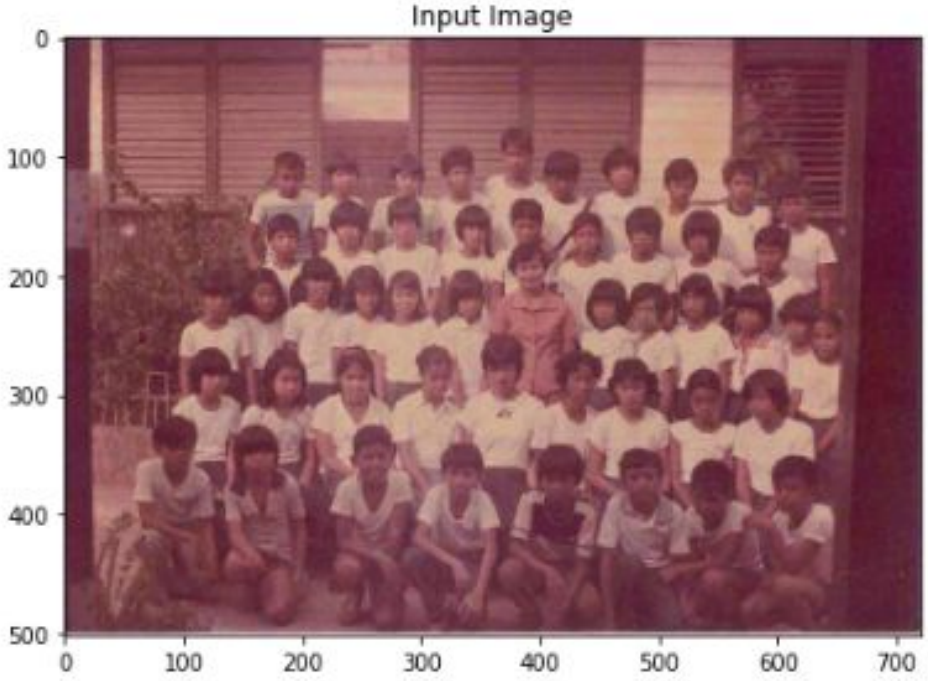


Fig 11.5. A close up of the patch we selected.



White Patch Algorithm

We then overlay each normalized transformed channel into one matrix using the following snippet of code,

```
img_GWP = np.zeros([np.shape(img)[0],np.shape(img)[1],3], dtype = float)
img_GWP[:,:,0] = RWB/np.max(RWB)
img_GWP[:,:,1] = GWB/np.max(GWB)
img_GWP[:,:,2] = BWB/np.max(BWB)
io.imshow(img_GWP)
```

Which resulted to the following colored and restored image . A side by side comparison can be seen in Figure 12.

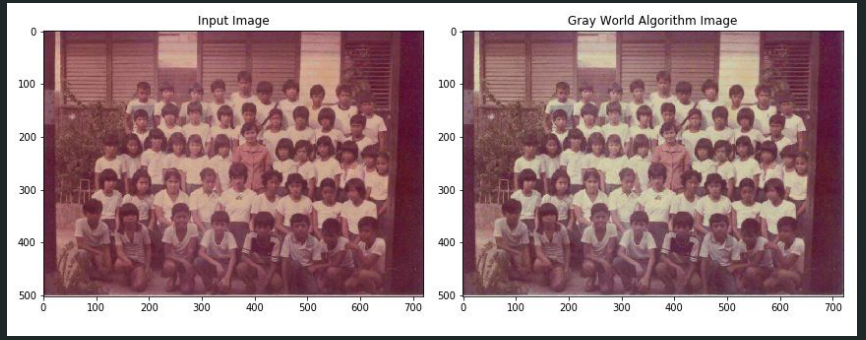


Fig 12. Resulting image which resulted from overlaying the normalized transformed channels using a white patch. Similar observations can be made with respect to the gray world algorithm, whereas, the image is now relatively brighter and clearer in contrast to the input image. However, it differs in the amount of brightness which occured for the white portions as we had defined a snippet that we were sure to be of a white color. As such, all intensities with similar values in each channel had similar transformation such that the algorithm is biased towards the whiter or lighter portions of the image.

White Patch Algorithm

Comparison

Comparison of Methods

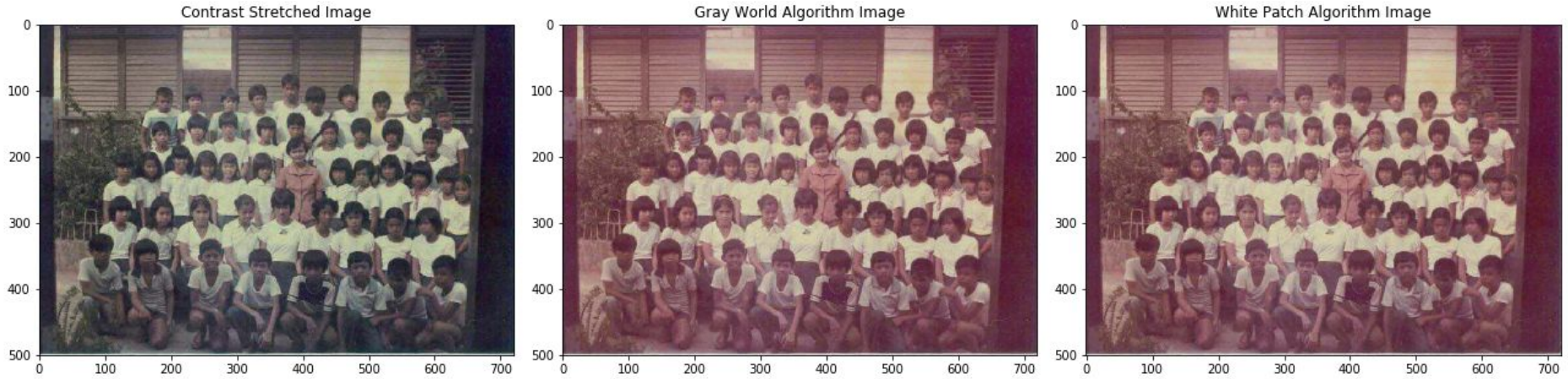


Fig 13. Side by side comparison of color enhancement methods. From the images above, we observed that contrast stretching performed the best as it had also removed the red tint that was overlayed on the image. In contrast to the two latter algorithms which were unable to account for the dominant red tint of the image. The white patch algorithm performed slightly better than the gray world algorithm as it was able to use a point of reference to perform a similar method - which in this case is the white t-shirt of one of the students. White patch algorithm resulted to a better representation of the color white, however, this is subjected to a systematic bias of the user such that (1) it requires for the color white to be present somewhere in the image, (2) that the user must be entirely sure that the “patch” is of the color white, and (3) no other color other than white must be included in the patch.

To give an objective critique on the performance of each method:

Contrast Stretching > White Patch* > Gray World

* This would be only true if the user is confident that the patch is of a color white only and that it exists on the input image

How do I score myself?

Technical Correctness: 5

Quality of Presentation: 5

Initiative: 2

AP 186 Activity 6
Enhancement of Color Images

Marc Jerrone Castro
2015-07420