CIS 4720 ASSIGNMENT 1 (PART 1 & PART 2)

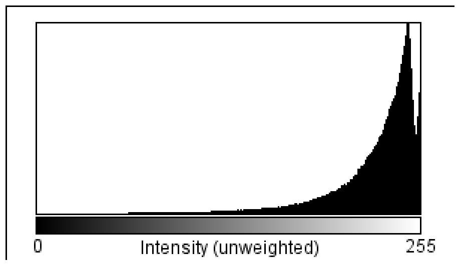
ABHISHEK JHOREE (0986820)

FEBRUARY 11, 2019

Part 1

Overexposed Image

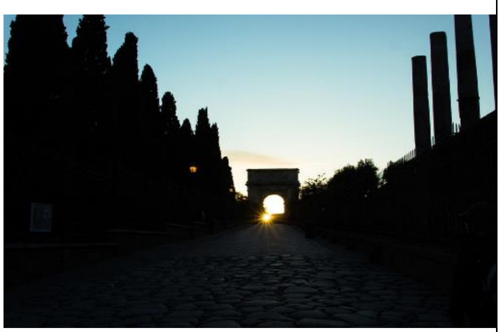
Overexposed image. Too much white light.



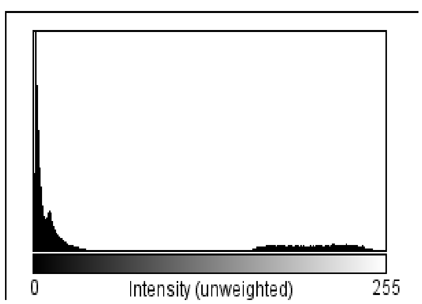
We can see that most of the values lie to the right of the histogram. This is because the picture is overexposed and hence lets in too much white light which explains why most values lie on the right

As the contrast has been reduced, we can now see the facial features mre distinctly that before.

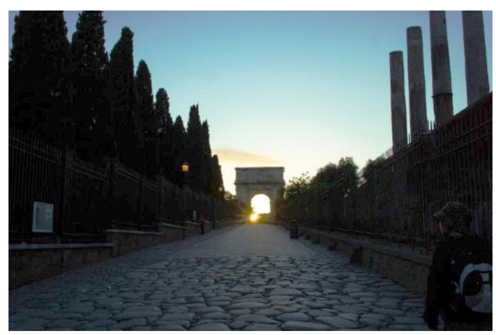
Underexposed image



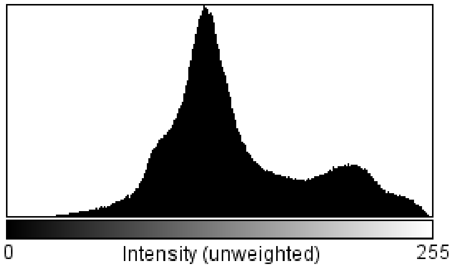
Before Image has been processed. We can see that this image is underexposed. There is not enough light and as a result the image looks dark



This is the histogram of the underexposed image. We can see that it is concentrated to the left showing the absence of light.

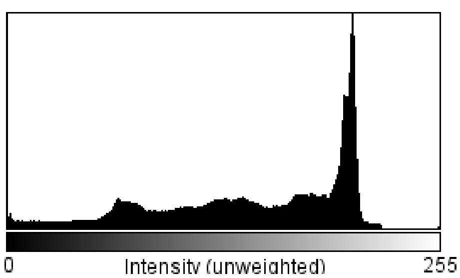


This is the image after it has been enhanced. We can see that the details are more visible now. Contrast has been increased.

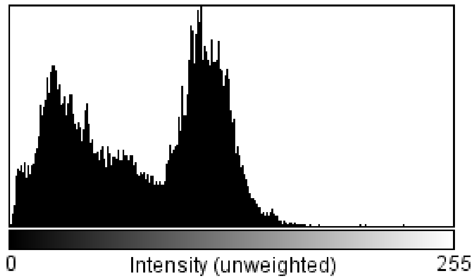
 We can see that the image is blurred due to the motion of the wings and the water droplets splashing. Since	 Histogram is concentrated in the middle. This is because the background is blurry. The bulk represents the eagle.	 Here the image has been corrected. The water is more focused, and we can now make out the eagle's eyes.

Out of Focus Image due to motion blur

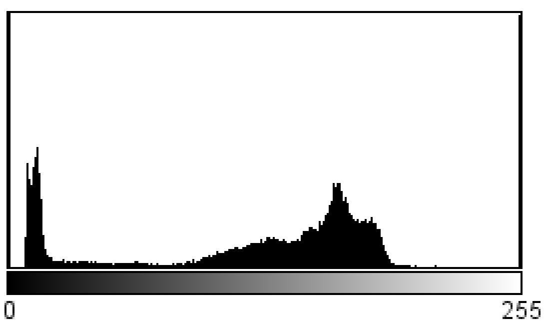
Out of focus

 The picture is out of focus. We can tell this by the blurry background		 In the improved image , the background has been sharpened and hence appears less blurry.

Incorrect white balance

 This picture has incorrect white balance. The picture is too warm and hence has this reddish glow		 There the picture is less warn. White balance has been attenuated.

Noisy Image

 This image is noisy. There are lots of speckles all over the picture. These “specks” are referred to as the noise of the image. Since we want a clearer picture of the guy taking the photograph, denoising is required.		 Noise has been removed from the picture. The specks are no longer present on the image

CIS 4720 ASSIGNEMNT 1 (PART 2)

Analysis and comparison of Contrast Enhancing Image Processing Algorithms

Introduction

For this section, I have decided to work with enhancing Contrast.

Contrast can simply be defined as the difference in luminance between objects. This property makes the objects easily distinguishable from one another. We can determine the contrast by calculating the difference in luminosity between objects in the same field of view or in this case photographs. Incorrect contrast manifests itself in a poor difference in luminosity or color and as a result the photograph will either appear too light or too dark in those regions lacking good contrast. As a consequence of this we will then attempt to fix those discrepancies by applying contrast enhancing algorithms to the photograph in order to fix the contrast and make objects more distinguishable. In this report, I will be comparing several algorithms and document my results so as to have a better understanding as to which ones work best and how so.

In this report we will be looking at 4 algorithms: **bi-histogram equalization (BBHE), Adaptive Histogram Equalization (AHE), contrast limited adaptative histogram equalization(CLAHE) and Histogram Hyperbolization.**

Contrast Enhancement Using Brightness Preserving Bi-Histogram Equalization (BBHE)

Overview of the algorithm

The general BBHE method is usually used for protecting of brightness of an image. Preserving brightness is one of the most important characteristics of an image. Hence, this method splits the image's histogram into two independently equalized parts. Hence, the intensities are arranged equally as well.
However, the brightness of the image can still be changed in transit after the HE. This is because of the straightening property of the Histogram Equalization.
Histogram equalization is one of the most widely used techniques for contrast enhancement. This is because of its simplicity and effectiveness and does not require much computation power to run. One disadvantage of the histogram equalization can be found on the fact that the brightness of an image can be changed after the histogram equalization, this is because of the flattening property of the histogram equalization. BHHE differs from the mainstream histogram equalization techniques. It utilizes independent histogram equalizations separately over two sub images. These are obtained by decomposing the input image based on its mean with a constraint that the resulting equalized sub images are bounded by each other around the input mean. It can be proven that the algorithm preserves the mean brightness of a given image significantly well compared to typical histogram equalization while enhancing the contrast and, thus, provides much natural enhancement that can be utilized in consumer electronic products which is one of the main reasons it was devised for. The ultimate goal BBHE is to preserve the mean brightness of a given image while the contrast is enhanced. This trait of BHHE is what makes it better than conventional histogram equalization algorithms.

How BBHE works

BBHE firstly decomposes an input image into two sub images based on the mean of the input image. One of the sets will be less than the mean whereas the other one will be greater than the mean. The 2 sub images are then distinctively equalized based on their respective histograms. At this point one major constraint is applied. The samples in the formal set are mapped into the range from the minimum gray level to the input mean and the samples in the latter set are mapped into the range from the mean to the maximum gray level. This process preserves the mean brightness of the image and gives BBHE its distinct property of brightness preservation.

Testing procedure for BBHE

The first test was to see how well the algorithm worked for exactly what it is meant to do. That is to preserve brightness. Therefore, I chose an Image that satisfied this need.

Figure 1
Before the algorithm was applied

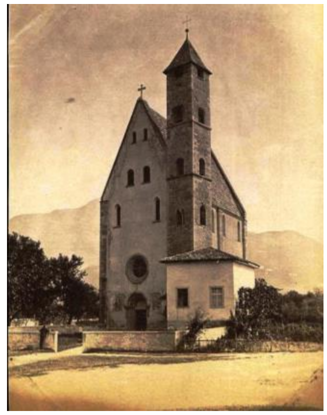


Figure 2 After the algorithm was used

The above (figure 1 and figure 2) demonstrates how the algorithm works on the image. We can see that the brightness has been preserved. This was verified with the intensity histogram of both images. The brightness level was relatively the same at around 21000 units which showed that the brightness was preserved. We can also see from this picture that the details are much more distinct than they were before. The contrast between the rooftop and the building has dramatically improved. Results were obtained relatively quick as well (1.36 seconds). Making this algorithm quite viable. Calculating the correlation between the 2 images yielded a score of 0.6963 indicating a significant difference.

For the next text, I used a large image to check the difference in computational time of the algorithm.

Figure 3: before restauration

Figure 4: after the filter has been applied

There was no vast difference in computational speeds even though this image was more that 10 times bigger that Figure 1. The output was similar in nature as well. There is a definite improvement in the quality of the processed image. Execution time was 3.09 seconds. We can see no vast difference given the 10x factor of the image. Hence works quite well even for larger images.

Finally, the algorithm was tested with an image with a defect. Since BBHE preserves brightness a defective image (log function applied to it, rendering it dull) was tested to see the render.
it was also tested with the same image that has been rendered excessively bright.

(left side = before, right side = after)

Figure 5

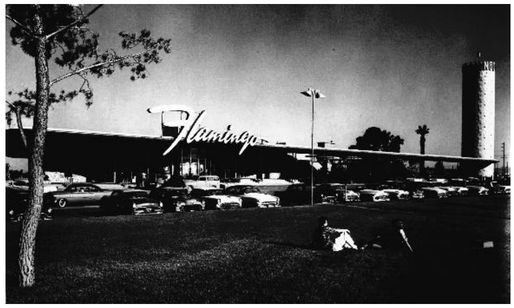


Figure 6

Figure 7

Figure 8

Ground truth (original picture)

As we can see comparing figure 5 (obtained from *ground truth*), the image has been poorly restored due to the lack of native brightness in the photo. This illustrates a clear limitation o the BBHE algorithm.

Figure 7 is excessively bright. The restored picture (figure 8) is equally as bright and it's hard to discern the contrast between the sky and the building. This is problematic if the lack of contrast in an image is due to the fact that several regions are overly bright. This shows that this algorithm is limited in this way. **More extensive test images can be found in the folder named BBHE**

Adaptative Histogram Equalization (AHE)

Adaptive Histogram Equalization is utilized for enhancing contrast as a standalone part of images. It varies from Histogram Equalization in the way that AHE uses versatile strategy. It computes a few histograms and every one of them relates to a particular segment of an image. The difference of area for an image won't be adequately improved by Histogram Equalization. AHE enhances this upgrade by changing every pixel with a change capacity got from an area locally. It is utilized to solve a few impediments of worldwide direct min-max windowing strategy. In this way it lessens the measure of commotion in districts of the image. Furthermore, AHE have the capacity for enhancing the difference of grayscale and color image.
However, AHE has a tendency to overamplify noise in relatively homogeneous regions of an image. A variant of adaptive histogram equalization called **contrast limited adaptive histogram equalization** (CLAHE) prevents this by limiting the amplification.
Ordinary AHE tends to overamplify the contrast in near-constant regions of the image, since the histogram in such regions is highly concentrated. As a result, AHE may cause noise to be amplified in near-constant regions. Contrast Limited AHE (CLAHE) is a variant of adaptive histogram equalization in which the contrast amplification is limited, so as to reduce this problem of noise amplification.

We will therefore be testing the AHE algorithm. At the same time, we will also be looking at the variant CLAHE and compare their results.

First test on an image with little contrast in the middle



before algorithm AHE algorithm CLAHE algorithm

As we can see, the original picture has several homogenous sections. The building, and the sky are 2 standalone homogenous regions. When AHE has been applied to it we can see that the resulting image has a little bit of noise specially in the sky region. In the CLAHE image (obtained from ImageJ) the noise is attenuated as depicted in the picture labelled CLAHE. A big drawback of AHE is that it takes a while to compute. The picture above took almost a minute to be generated. This is problematic for larger images. However, as a whole this picture has been processed quite well by the AHE algorithm.

The second test will focus on testing an image with added defects to test how the algorithm can handle images that have been transformed to look relatively uniform; thus, testing the weakness of the algorithm for uniform backgrounds

Ground truth Image

Original image used to generate defective images.

AHE ALGORITHM

Figure 1 (1st defect) 	**Figure 2 (After AHE)** 
Figure 3 (2nd defect) 	**Figure 4 (After AHE)** 

CLAHE

before 	**after** 
before 	**after** 

As we can see, AHE does a very poor job of processing images that are uniform in contrast. The images generated have a lot of noise making them almost indistinguishable from the **ground truth** image. This is very apparent when **figure 3** is used as input. The image looks very uniform and the resulting output has a lot of noise in it. Hence, we can clearly see the limitations of AHE when it comes to processing images with large homogenous regions in this test. We end up with images that have a lot of noise in them.

As for CLAHE, it does a better job at avoiding excessive noise for those homogenous regions (**first picture demonstrates this).** However, it still has a lot of trouble for the extreme cases (**last picture)** where the regions are very uniform. It generates a slightly better result than AHE.

More extensive restored images for AHE can be found in the folder AHE

More extensive restored images for CLAHE can be found in the folder CLAHE

Histogram Hyperbolization

The 3rd algorithm we are going to be looking at is Histogram Hyperbolization. It consists of a memoryless nonlinear, space-invariant transformation of the pixel brightness values. This transformation is based upon the histogram of the picture to be processed and the nature of human brightness perception Because the visibility of image features is directly affected by such relative differences, it is often possible to enhance the intelligibility of pictures by an appropriate transformation J(I) of the pixel brightness values I(x, y).

First test on non-defective images lacking contrast

Fig 1: Before

Fig 1: After

Fig 2: Before

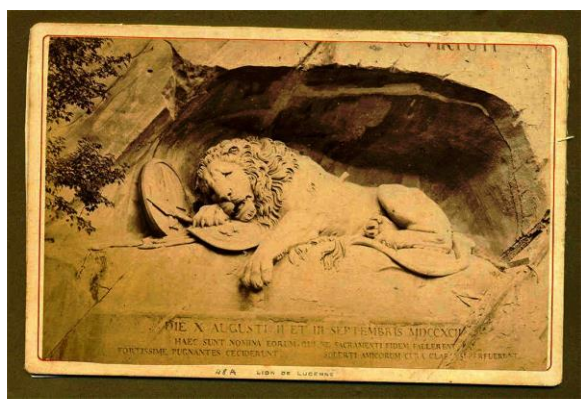


Fig 2: After

The results computed those images were very impressive. Fig 1 as we can see has very little contrast. After being processed the image is much more distinct. Rocks can be discerned; the different tones of the ground can also be seen. Comparing the histograms of Fig 1 before and after yields a score of 0.4739, indicating a change of more than 50 % from the original image which is very good.

Fig 2 has also seen a good improvement, we can now see that the rock has different tones and the texture of the cove can also be noticed. We can hence say that this algorithm works very well with photos that have no defects and little contrast.

Testing the algorithm with defective images that have very little contrast (log function applied to them)

Before (fig 1) (fig 1) After

Ground truth image used:

Before (uniformly bright) fig 2 **After(excessive noise) fig 2**

Fig 3 before

Fig 3 after

We can see that as contrast becomes less distinct, the algorithm has more trouble restoring the image. The difference in the histogram for the 4 defective pictures provided yield scores ranging from 0.528 to 0.319, indicating large changes for all of the images with the best results being for fig 3. **Fig 3 scored an impressive 0.919 when compared to the original ground truth image.** Fig 3 had a log function applied to it and uniformly made darker. Thus, we can conclude that the algorithm works best for this scenario type. The uniformly dull image (fig 1) was also very close but the worst one was the bright image which produced a lot of noise. Hence, we can see that overly bright images don't work well with that particular algorithm.

Extensive test images for Histogram Hyperbolization can be found in the folder Hyperbolization.

End of Report

Difficulties with contrast in Image Processing

The process behind enhancing the images lacking contrast in image processing is not straightforward. For this assignment, a lot of research had to be done in order to find and implement those algorithms that best enhance contrast. Furthermore, it was difficult dealing with different size of pictures. Some of them required special functions to handle then and resize them to an appropriate size. A lot of work also went into selecting the best images for each test case if order to exhaustively test these algorithms. Coming up with the best match wasn't an easy task. However this was nonetheless very enjoyable as I got to write my own functions in Python and write my own algorithms.

References:

Kim, Y.-T. "Contrast enhancement using brightness preserving bi-histogram equalization", IEEE Transactions on Consumer Electronics, Vol.43, pp.1-8 (1997).

Frei, W., "Image enhancement by histogram hyperbolization", Computer Graphics and Image Processing, Vol.6(3), pp.286-294 (1977)

S. M. Pizer, E. P. Amburn, J. D. Austin, et al.: *Adaptive Histogram Equalization and Its Variations*. Computer Vision, Graphics, and Image Processing 39 (1987) 355-368.

K. Zuiderveld: Contrast Limited Adaptive Histogram Equalization. In: P. Heckbert: Graphics Gems IV, Academic Press 1994, ISBN 0-12-336155-9