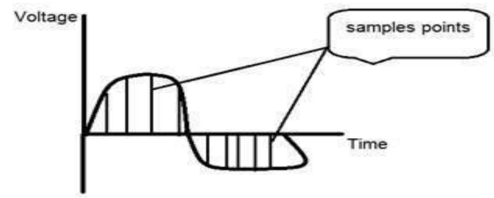
Chapter 2

Image enhancement in spatial domain

****

Digitizing a signal

Digitizing an analog signal into a digital, requires two basic steps.



Sampling and quantization

Sampling is done on **x axis**. It is the conversion of x axis (infinite values) to digital values.

The figure shows sampling of a signal.

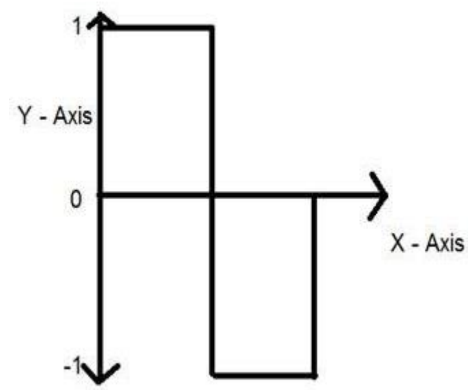
Sampling with relation to digital images

The concept of sampling is directly related to **zooming**. The more samples you take, the more pixels, you get.

Oversampling can also be called as zooming.

But the story of digitizing a signal does not end at sampling too, there is another step involved which is known as Quantization.

What is quantization



Quantization is opposite to sampling. It is done on **y axis**.

When you are quantizing an image, you are actually dividing a signal into ***quanta (partitions)***.

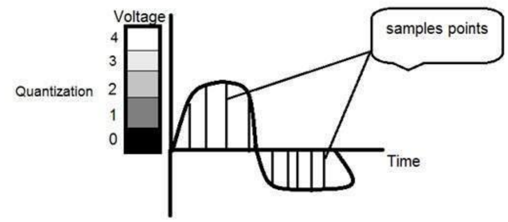
On the *x axis of the signal*, are the co-ordinate values,

and on the *y axis*, we have amplitudes. So digitizing the amplitudes is known as Quantization.

Here how it is done:

You can see in this image, that the signal has been quantified into three different levels. That means that when we sample an image, we actually gather a lot of values, and in quantization, we set levels to these values. This can be more clear in the image below.

In the figure shown in sampling, although the samples have been taken, but they were still spanning vertically to a continuous range of gray level values.



In the figure shown, these vertically ranging values have been quantized into 5 different levels or partitions. Ranging from 0 black to 4 white. This level could vary according to the type of image you want.

Relation of Quantization with gray level resolution:

The quantized figure shown above has 5 different levels of gray. It means that the image formed from this signal, would only have 5 different colors. It would be a black and white image more or less with some colors of gray.

Now if you were to make the quality of the image more better, there is one thing you can do here. Which is, to increase the levels, or gray level resolution up. If you increase this level to 256, it means you have a gray scale image. Which is far better then simple black and white image?

Now 256, or 5 or whatever level you choose is called gray level. Remember the formula that we discussed in the previous tutorial of gray level resolution which is,

$$L = 2^k$$

We have discussed that gray level can be defined in two ways. Which were these two.

- *Gray level = number of bits per pixel (BPP).(k in the equation)*
- *Gray level = number of levels per pixel.*

In this case we have gray level is equal to 256.

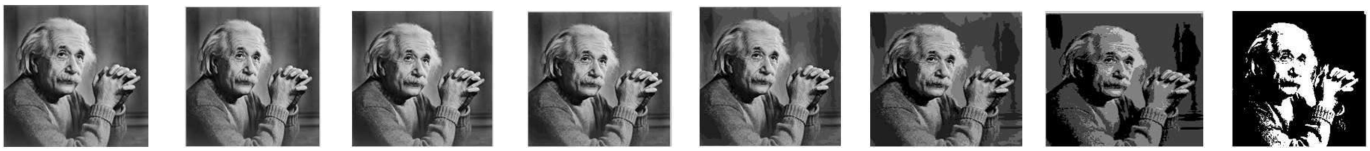
If we have to calculate the number of bits, we would simply put the values in the equation. In case of 256levels, we have 256 different shades of gray and 8 bits per pixel, hence the image would be a gray scale image.

Reducing the gray level

Now we will reduce the gray levels of the image to see the effect on the image.

For example

Lets say you have an image of 8bpp, that has 256 different levels. It is a grayscale image and the image looks something like this.



A B C D E F G H

(a) 256 Gray Levels **(b) 128 Gray Levels**

(c) 64 Gray Levels **(d) 32 Gray Levels**

(e) 16 Gray Levels **(f) 8 Gray Levels**

(g) 4 Gray Levels **(h) 2 Gray Levels**

(a) Now we will start reducing the gray levels. We will first reduce the gray levels from 256 to 128.

(b) There is not much effect on an image after decrease the gray levels to its half. Lets decrease some more.

(c) Still not much effect, then lets reduce the levels more.

(d) Surprised to see, that there is still some little effect. May be its due to reason, that it is the picture of Einstein, but lets reduce the levels more.

(e) The image finally reveals, that it is effected by the levels.

(f,g,h) Now before reducing it, further two 2 levels, you can easily see that the image has been distorted badly by reducing the gray levels. Now we will reduce it to 2 levels, which is nothing but a simple black and white level. It means the image would be simple black and white image.

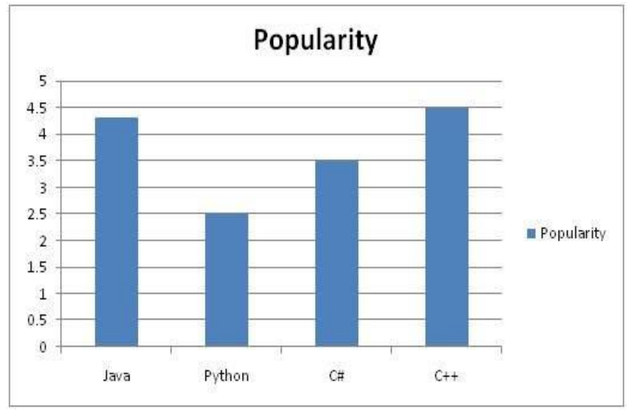
****

Histograms

Histogram Introduction:

A histogram is a graph. A graph that shows frequency of anything.

Usually histogram have bars that represent frequency of occurring of data in the whole data set.



A Histogram has two axes the x axis and the y axis.

The x axis contains event whose **frequency you have to count.** The y axis contains **frequency.**

The different heights of bar show different frequency of occurrence of data. Now we will see an example of this histogram is build

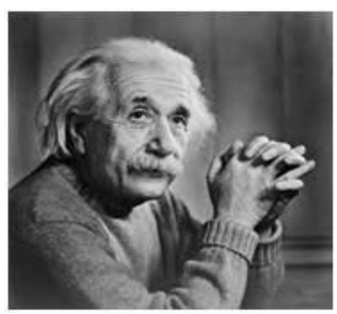
Histogram of an image

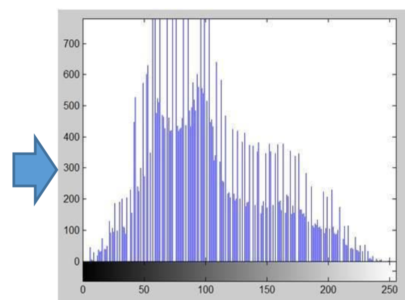
Histogram of an image, like other histograms also shows frequency.

But an image histogram, shows frequency of pixels intensity values.

In an image histogram, the x axis shows the gray level intensities and the y axis shows the frequency of these intensities.

For example





The histogram of the picture of the Einstein would be something like this

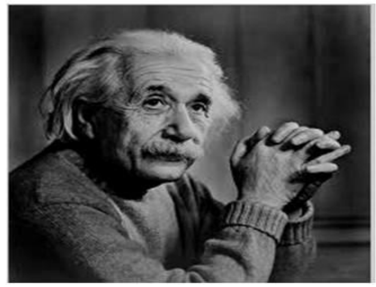
The x axis of the histogram shows the range of pixel values. Since its an 8 bpp image, that means it has 256 levels of gray or shades of gray in it.

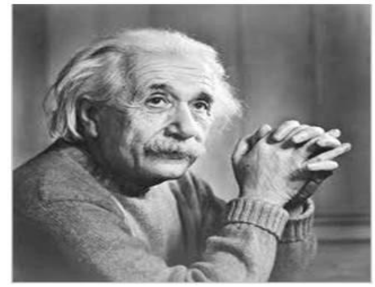
That’s why the range of x axis starts from 0 and end at 255 with a gap of 50. Whereas on the y axis, is the count of these intensities.

As you can see from the graph, that most of the bars that have high frequency lies in the first half portion which is the darker portion.

That means that the image we have got is darker. And this can be proved from the image too.

Applications of Histograms





Histograms has many uses in image processing.

The first use as it has also been discussed above is the analysis of the image such that we can predict about an image by just looking at its histogram.

The second use of histogram is for brightness purposes.

The histograms have wide application in image **brightness.** Not only in brightness, but histograms are also used in adjusting **contrast** of an image.

Another important use of histogram is to **equalize** an image.

And last but not the least, histogram has wide use in **thresholding.** This is mostly used in ***computer vision*.**

Brightness

Brightness can be defined as the amount of energy output by a source of light relative to the source we are comparing it to.

In some cases, we can easily say that the image is **bright**, and in some cases, it's not easy to **perceive.**

For example

Just have a look at both of these images, and compare which one is brighter.

We can easily see, that the image on the right side is brighter as compared to the image on the left.

But if the image on the right is made darker then the first one, then we can say that the image on the left is brighter than the left.

How to make an image brighter?

Brightness can be simply ***increased or decreased*** *by simple addition or subtraction, to the image* ***matrix.***

Consider this black image of 5 rows and 5 columns

STEP 1:

Since we already know, that each image has a matrix at its behind that contains the pixel values. This image matrix is given below.

0 0 0 0 0

0 0 0 0 0

0 0 0 0 0

0 0 0 0 0

0 0 0 0 0

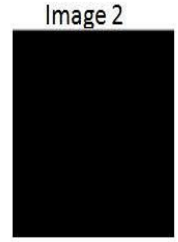
Since the whole matrix is filled with zero, and the image is very much darker.

STEP 2:

Now we will compare it with another same black image to see this image got brighter or not.

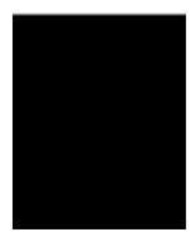
Image 1

Image 2



Still both the images are same, now we will perform some operations on image1, due to which it becomes brighter than the second one.

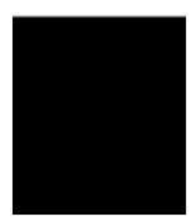
STEP 3:



What we will do is, that we will simply add a value of 1 to each of the matrix value of image 1.

After adding the image 1 would something like this.

STEP 4:



Now we will again compare it with image 2, and see any difference.

Sanjeev Thapa :- RHCSA,RHCE,RHCSA-Openstack,MTCNA,MTCTCE,USRS,3CX CE,**HE IPv6** ;)
CCT-BCIS,6th Sem, Spring 2017, maximus.innase@gmail.com

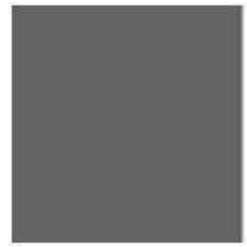
STEP 5:

We see, that still we cannot tell which image is brighter as both images looks the same.

Now what we will do, is that we will add 50 to each of the matrix value of the image 1 and see what the image has become.

STEP 6:

Consider the final image1 in brightness as, after adding 100 to each of the matrix value



The matrix of this image is:

100	**100**	**100**	**100**	**100**
100	**100**	**100**	**100**	**100**
100	**100**	**100**	**100**	**100**
100	**100**	**100**	**100**	**100**
100	**100**	**100**	**100**	**100**

The maximum value in this matrix is 100 and The minimum value in this matrix is 100.

Contrast = maximum pixel intensity (subtracted by) minimum pixel intensity

= 100 (subtracted by) 100

= 0

*0 means that this image has **0 contrast***

****

Histogram sliding

In histogram sliding, we just **simply shift** a complete histogram **rightwards or leftwards.**

Due to shifting or sliding of histogram towards right or left, a clear change can be seen in the image.

In this scenario we are going to use histogram sliding for manipulating **brightness.**

Brightness: Brightness is a relative term. Brightness can be defined as intensity of light emit by a particular light source.

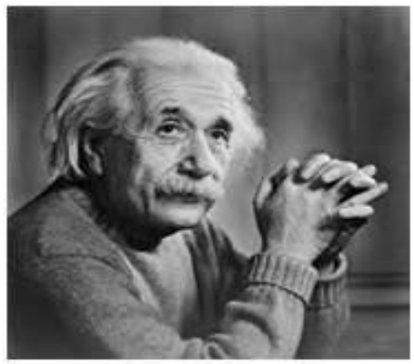
Contrast: Contrast can be defined as the difference between maximum and minimum pixel intensity in an image.

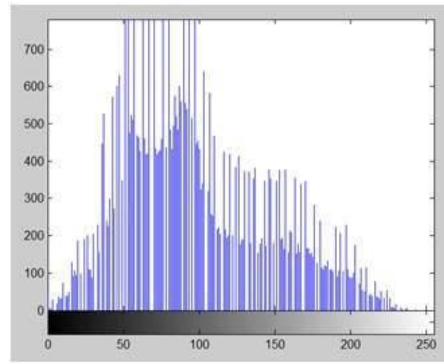
Sliding Histograms: (Example scenario for brightness)

STEP 1: Increasing brightness using histogram sliding

On the y axis of this histogram are the **frequency or count.**

And on the x axis, we have **gray level values.**



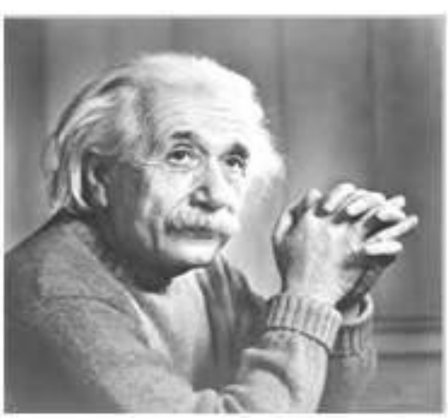


As you can see from the above histogram, that those gray level intensities whose count is more then ***700, lies in the first half portion***, means towards **blacker portion.** That's why we got an image that is a bit darker.

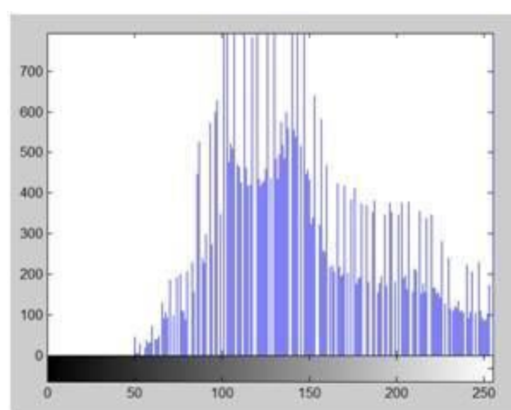
In order to bright it, we will slide its histogram ***towards right***, or towards whiter portion. In order to do we need to add at least a value of **50** to this image. Because we can see from the histogram above, that this image also has 0 pixel intensities, that are pure black.

So if we **add 0 to 50**, we will shift all the values lies at 0 intensity to 50 intensities and all the rest of the values will be shifted accordingly.

STEP 2: Here what we got after adding 50 to each pixel intensity.

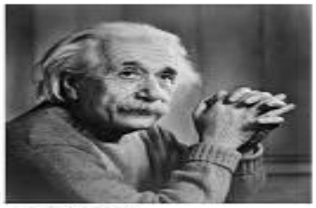


The image and histogram has been shown



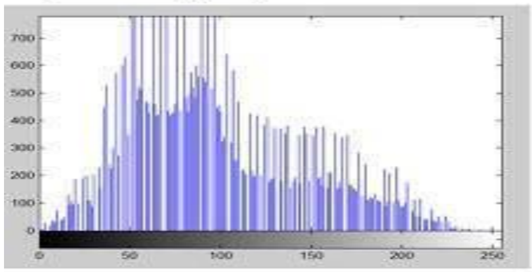
And its complete histogram sliding has been shown below.

Old image

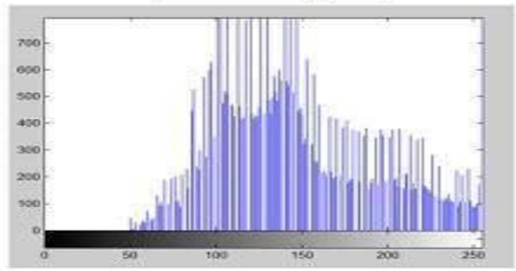


New image

Old histogram



New Histogram



Conclusion

As we can clearly see from the new histogram that all the pixel's values have been **shifted towards right** and its effect can be seen in the new image.

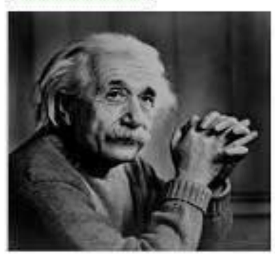
Decreasing brightness using histogram sliding

Now if we were to **decrease brightness** of this new image to such an extent that the old image looks brighter, we got to subtract some value from all the matrix of the new image.

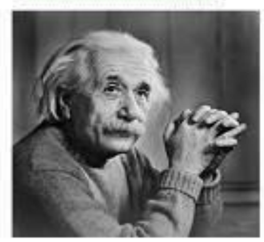
The value which we are going to ***subtract is 80.***

Because we already add 50 to the original image and we got a new brighter image, now if we want to make it darker, we have to subtract at least more than 50 from it.

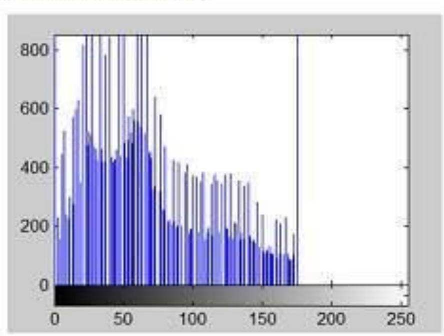
New image.



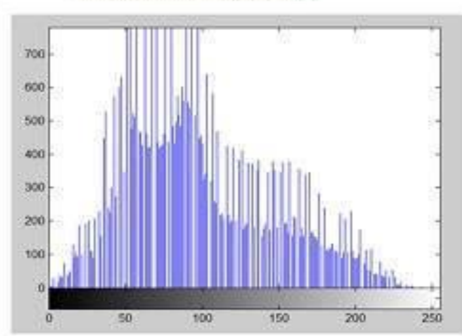
Original image.



New Histogram.



Original Histogram.



Conclusion

It is clear from the histogram of the new image, that all the pixel values have been ***shifted towards right*** and thus, it can be validated from the image that new image is darker and now the original image look brighter as compare to this new image

Histogram stretching

One of the other advantage of Histogram s that we discussed is "introduction to ***histograms is contrast enhancement*** ".

There are two methods of enhancing contrast.

The **first** one is called ***Histogram stretching that increase contrast***.

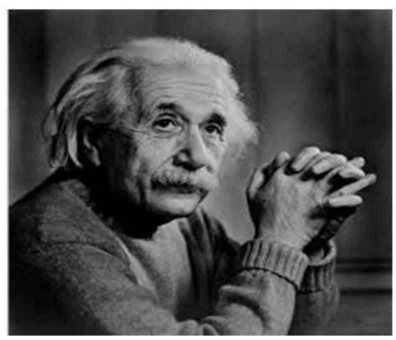
The **second** one is called ***Histogram equalization that enhance contrast***.

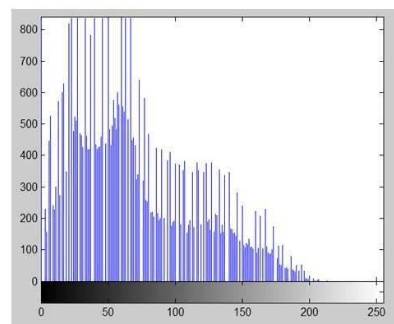
Before we will discuss the histogram stretching to increase contrast, we will briefly define contrast.

Contrast

Contrast is the difference between maximum and minimum pixel intensity.

Consider this image with its histogram.





Now we calculate contrast from this image.

Contrast = 225 - 0 = 255

Now we will increase the contrast of the image.

Increasing the contrast of the image:

The formula for stretching the histogram of the image to increase the contrast is:

$$g(x,y) = \frac{f(x,y) - fmin}{fmax - fmin} * 2^{bpp}$$

The formula requires finding the minimum and maximum pixel intensity multiply by levels of gray. In our case the image is 8bpp, so levels of gray are 256.

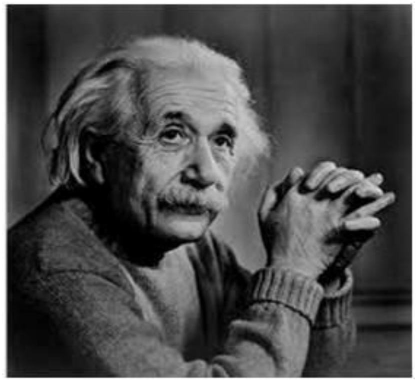
$$g(x,y) = \frac{f(x,y) - 0}{225 - 0} * 255$$

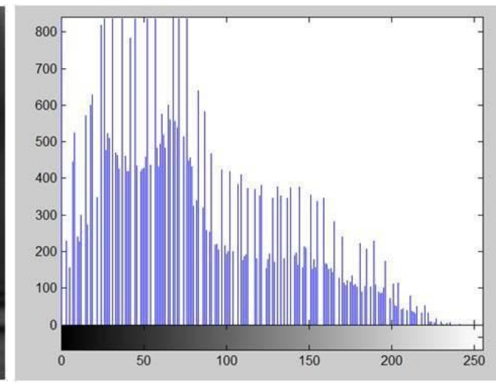
The minimum value is 0 and the maximum value is 225. So the formula in our case is

where **f(x,y)** denotes the value of each **pixel intensity.** For each f(x,y) in an image , we will calculate this formula.

After doing this, we will be able to enhance our contrast.

The following image appear after applying histogram stretching.





The stretched histogram of this image has been shown.

Note the shape and **symmetry** of histogram. The histogram is now ***stretched or in other means expand***. Have a look at it.

In this case the contrast of the image can be calculated as

Contrast = 240

Hence we can say that the contrast of the image is increased.

Note: this method of increasing contrast does not work always, but it fails on some cases.

Failing of histogram stretching:

As we have discussed, that the algorithm fails on some cases. Those cases include images with when there is pixel intensity 0 and 255 are present in the image

Because when pixel intensities 0 and 255 are present in an image, then in that case they become the minimum and maximum pixel intensity which ruins the formula like this.

$$g(x,y) = \frac{f(x,y) - fmin}{fmax - fmin} * 2^{bpp}$$

Original Formula

Putting fail case values in the formula:

$$g(x,y) = \frac{f(x,y) - 0}{255 - 0} * 255$$

Simplify that expression gives

$$g(x,y) = \frac{f(x,y)}{255} * 255$$

$$g(x,y) = f(x,y)$$

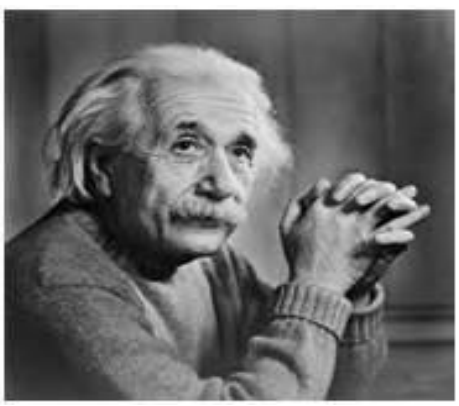
That means the output image is equal to the **processed image.** That means there is **no effect** of histogram stretching has been done at this image.

Histogram Equalization

We have already seen that contrast can be increased using histogram stretching. In this mechanism we will see that how **histogram equalization** can be used to **enhance contrast.**

Histogram equalization is used to **enhance contrast.** It is not necessary that contrast will always be increase in this. There may be some cases were histogram equalization can be worse. In that cases the contrast is decreased.

Let's start histogram equalization by taking this image below as a simple image.



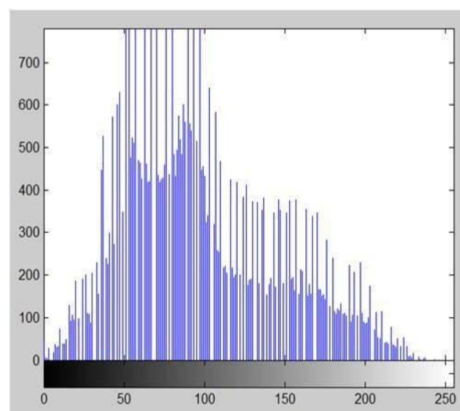


Image **Histogram of this image**

Now we will perform histogram equalization to it.

PMF: First we have to calculate the PMF (probability mass function) of all the pixels in this image.

CDF: Our next step involves calculation of CDF (cumulative distributive function

Gray Level Value	CDF
0	0.11
1	0.22
2	0.55
3	0.66
4	0.77
5	0.88
6	0.99
7	1

STEP 1: Calculate CDF according to gray levels

Let's for instance consider this, that the CDF calculated in the second step looks like this table.

STEP 2: Then in this step you will multiply the CDF value with (Gray levels (minus) 1).

Considering we have a 3 bpp image. Then number of levels we have are 8. And 1 subtracts 8 is 7. So we multiply CDF by 7. Here what we got after multiplying.

Gray Level Value	CDF	CDF * (Levels-1)
0	0.11	0
1	0.22	1
2	0.55	3
3	0.66	4
4	0.77	5
5	0.88	6
6	0.99	6
7	1	7

STEP 3: Now we have is the last step, in which we have to map the new gray level values into number of pixels.

Let's assume our old gray levels values has these number of pixels

Gray Level Value	Frequency
0	2
1	4
2	6
3	8
4	10
5	12
6	14
7	16

STEP 4: Now if we map our new values to , then this is what we got.

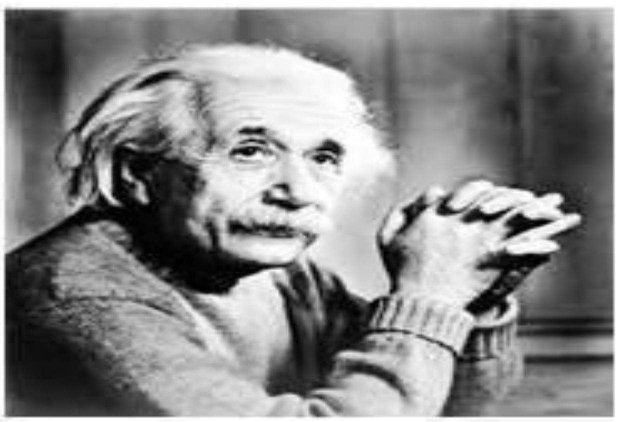
.

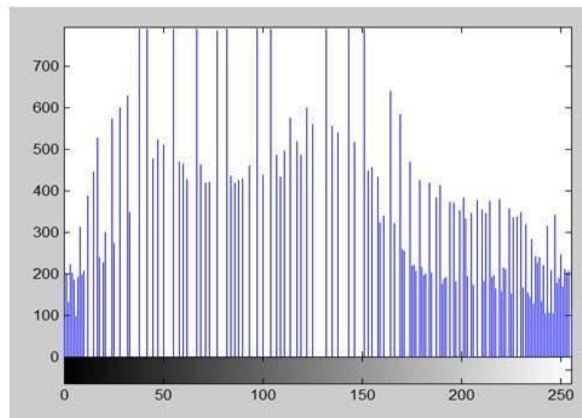
Gray Level Value	New Gray Level Value	Frequency
0	0	2
1	1	4
2	3	6
3	4	8
4	5	10
5	6	12
6	6	14
7	7	16

After all, Now map these new values you are onto histogram, and you are done.

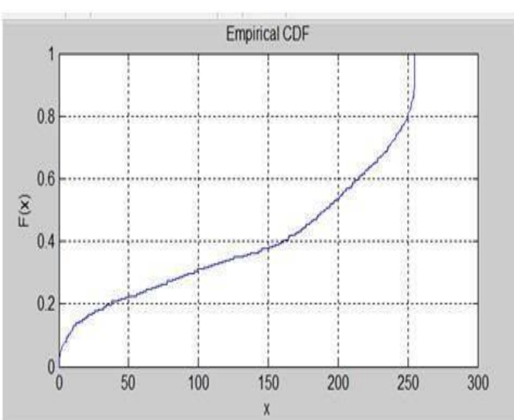
Let's apply this technique to our original image. After applying we got the following image and its following histogram.

Histogram Equalization Image Histogram Equalization histogram

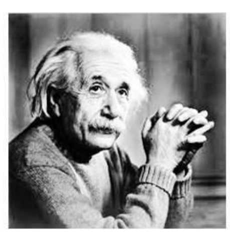




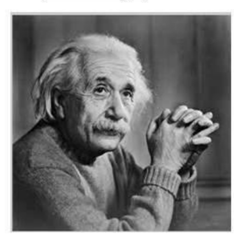
Cumulative Distributive function of this image



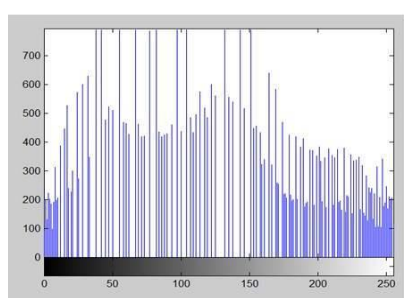
New Image



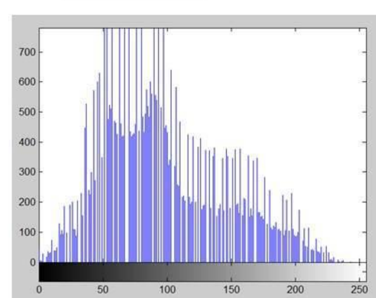
Old image



New Histogram



Old Histogram



Comparing both the histograms and images

Conclusion:

As you can clearly see from the images that the new image **contrast has been enhanced** and its histogram has also been equalized.

There is also one important thing to be note here that during histogram equalization **the overall shape** of the histogram changes, where as in histogram stretching the overall shape of histogram remains same.

****

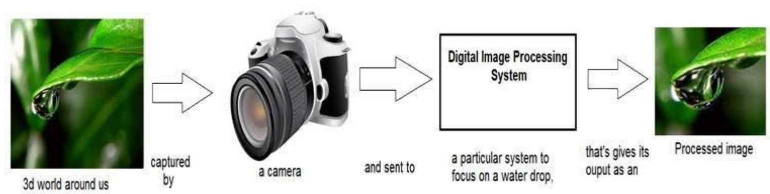
Gray Level Transformation

Before we discuss, what is image transformation, we will discuss **what a transformation is?**

Transformation:

Transformation is a **function that maps one set to another set after performing some operations.**

Digital Image Processing system



We have already seen in the introductory part, that in digital image processing, we will develop a system that whose **input would be an image** and **output would be an image too.** And the system would perform some **processing** on the input image and gives its output as a processed image. It is shown above.

Now **function** applied inside this digital system that process an image and convert it into output can be called as **transformation function.**

As it shows transformation or relation, that how an **image1** is converted to **image2.**

Image transformation.

Consider this equation **G(x,y) = T { f(x,y) }**

In this equation,

F(x,y) = input image on which transformation function has to be applied.

G(x,y) = the output image or processed image.

T is the transformation function.

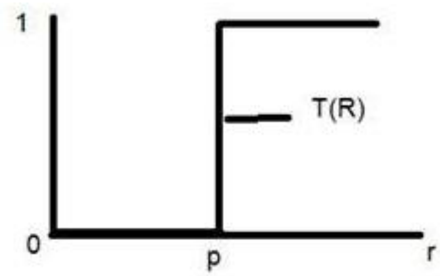
This relation between input image and the processed output image can also be represented as. ***s = T (r)***

Where **r** is actually the pixel value or gray level intensity of f(x,y) at any point. And **s** is the pixel value or gray level intensity of g(x,y) at any point.

Now we are going to discuss some of the very basic transformation functions.

Examples

Consider this transformation function.



Let's take the point r to be 256, and the point p to be 127.

Consider this image to be a one bpp image. That means we have only two levels of intensities that are 0 and 1.

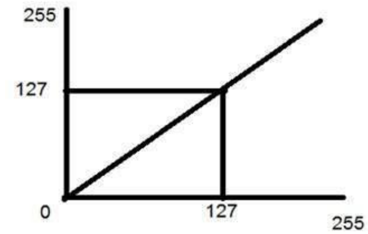
So in this case the transformation shown by the graph can be explained as.

All the pixel intensity values that are below 127 (point p) are 0, means black. And all the pixel intensity values that are greater than 127, are 1, that means white. But at the exact point of 127, there is a sudden change in transmission, so we cannot tell that at that exact point, the value would be 0 or 1.

$$g(x,y) = \begin{cases} 0 & f(x,y) < 127 \\ 1 & f(x,y) > 127 \end{cases}$$

Mathematically this transformation function can be denoted as:

Consider another transformation like this



Now if you will look at this particular graph, you will see a straight transition line between input image and output image.

It shows that for each pixel or intensity value of input image, there is a same intensity value of output image. That means the output image is exact replica of the input image.

It can be mathematically represented as: **$g(x,y) = f(x,y)$**

the input and output image would be in this case are shown below.

Input image

Output image

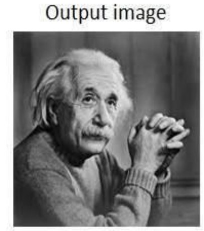


Image enhancement:

As we have discussed above some of the basic transformations in. In this image enhancement we will look at some of the basic gray level transformations.

Enhancing an image provides better contrast and a more detailed image as compare to non-enhanced image.

Image enhancement has very applications. It is used to enhance medical images, images captured in remote sensing, images from satellite etc.

The transformation function has been given as **$s = T(r)$**

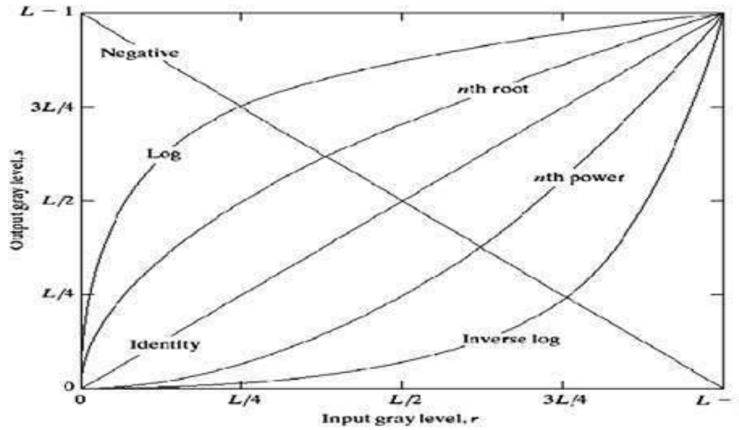
Where r is the pixels of the input image and s is the pixels of the output image. T is a transformation function that maps each value of r to each value of s. Image enhancement can be done through gray level transformations which are discussed below.

Gray level transformation

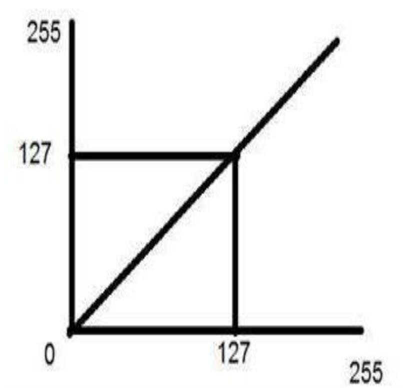
There are three basic gray level transformation.

- **Linear**
- **Logarithmic**
- **Power – law**

The overall graph of these transitions has been shown.



Linear transformation



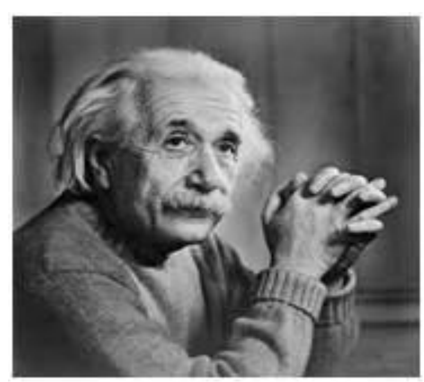
First we will look at the linear transformation. Linear transformation includes simple identity and negative transformation. Identity transformation has been discussed in our tutorial of image transformation, but a brief description of this transformation has been given here.

Identity transition is shown by a straight line. In this transition, each value of the input image is directly mapped to each other value of output image. That results in the same input image and output image. And hence is called identity transformation. It has been shown below:

Negative transformation

The second linear transformation is negative transformation, which is invert of identity transformation. In negative transformation, each value of the input image is subtracted from the L-1 and mapped onto the output image.

The result is somewhat like this.



Input Image

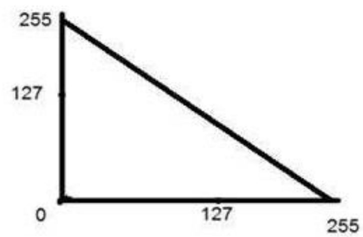
Output Image

In this case the following transition has been done ***s = (L – 1) – r***

since the input image of Einstein is an 8 bpp image, so the number of levels in this image are 256.

Putting 256 in the equation, we get this ***s = 255 – r***

So each value is subtracted by 255 and the result image has been shown above. So what happens is that, the ***lighter pixels*** become dark and the darker picture becomes light. And it results in **image negative.**



It has been shown in the graph.

Logarithmic transformations:

Logarithmic transformation further contains **two** type of transformation. **Log transformation** and **inverse log transformation.**

The log transformations can be defined by this formula

s = c log (r + 1)

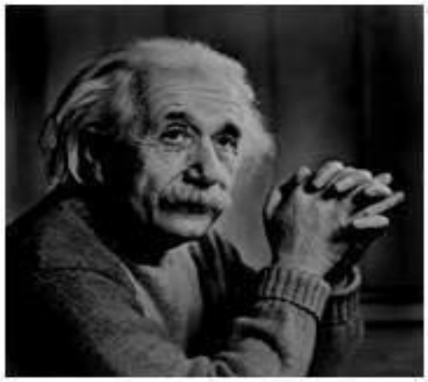
Where **s and r** are the pixel values of the **output** and the **input** image and c is a **constant.**

The value 1 is added to each of the pixel value of the input image because if there is a pixel intensity of 0 in the image, then log (0) is equal to **infinity.** So 1 is added, to make the minimum value at least 1.

During log transformation, the dark pixels in an image are expanded as compare to the higher pixel values. The higher pixel values are kind of compressed in log transformation. This result in following image enhancement.

The value of c in the log transform adjust the kind of enhancement you are looking for.

Input Image ***Log Transform Image***



The inverse log transform is opposite to log transform.

Power – Law transformations

There are further two transformations is power law transformations, that include nth power and nth root transformation.

These transformations can be given by the expression: $s=cr^\gamma$

This symbol **γ** is called gamma, due to which this transformation is also known as **gamma transformation**.

Variation in the value of γ varies the enhancement of the images. Different display devices / monitors have their own gamma correction, that's why they display their image at different intensity.

This type of transformation is used for enhancing images for different type of display devices. The gamma of different display devices is different.

For example, Gamma of CRT lies in between of **1.8 to 2.5**, that means the image displayed on CRT is dark.

Correcting gamma.

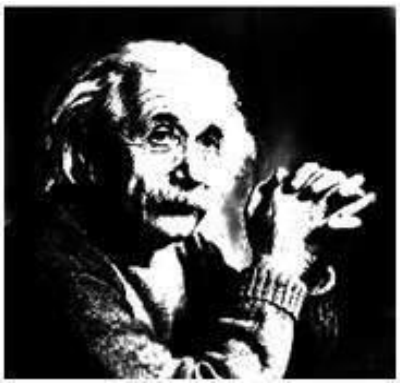
$$s=cr^\gamma$$

s=cr^(1/2.5)

The same image but with different gamma values has been shown here.

For example

Gamma = 10 **Gamma = 8** **Gamma = 6**



Sanjeev Thapa :- RHCSA,RHCE,RHCSA-Openstack,MTCNA,MTCTCE,USRS,3CX CE,**HE IPv6** ;)
CCT-BCIS,6th Sem, Spring 2017, maximus.innase@gmail.com