Image Enhancement Techniques

Objective – process an image so that the result is more suitable than the original image for a specific application.

Methods

1. Spatial Domain
   direct manipulation of pixels of the image
2. Frequency Domain
   modifying the Fourier Transform of an image

Image Enhancement in Spatial Domain

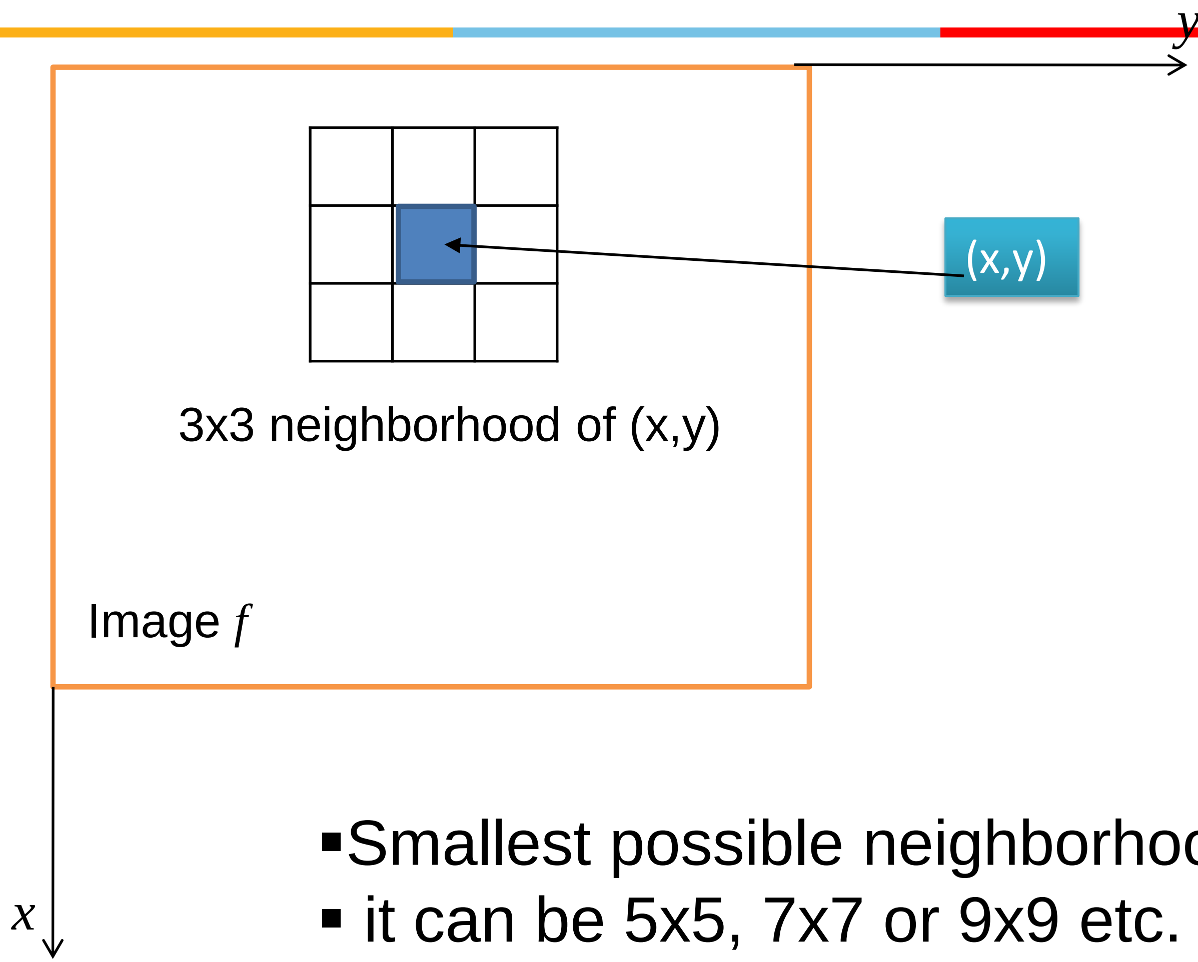
- These techniques operate directly on the pixels.
- More efficient computation and requires less processing resources to implement
- Spatial Domain Process is defined by $g(x,y)=T[f(x,y)]$

T is an operator on f defined over a neighborhood of point (x,y)

OUTPUT IMAGE

INPUT IMAGE

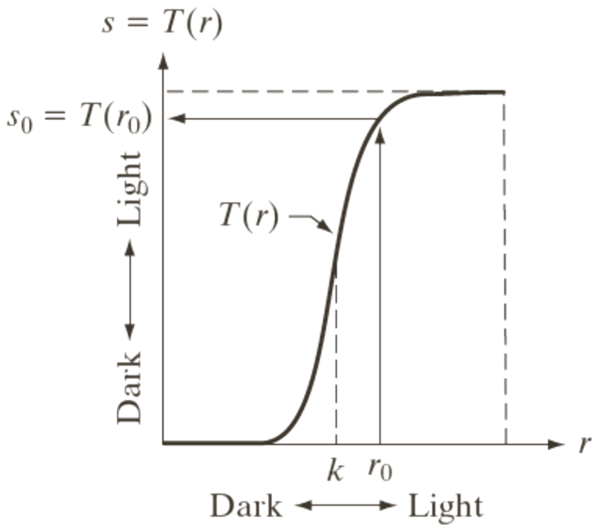
Image Enhancement in Spatial Domain



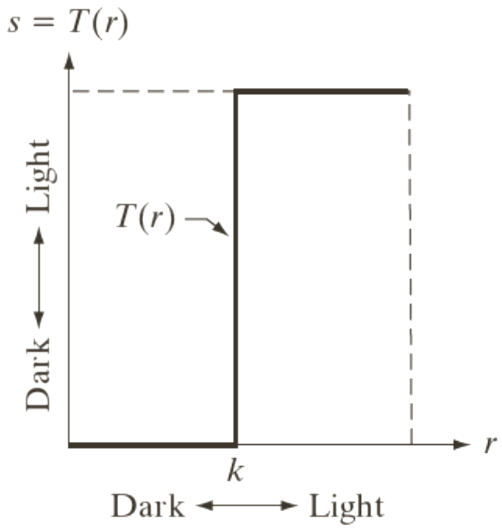
- Smallest possible neighborhood size is 1x1,
- it can be 5x5, 7x7 or 9x9 etc.

Image Enhancement in Spatial Domain

1x1 neighborhood operation is called as point processing and is represented by the transformation function s= T(r). Where s and r represents the intensity of g and f respectively



Contrast stretching function

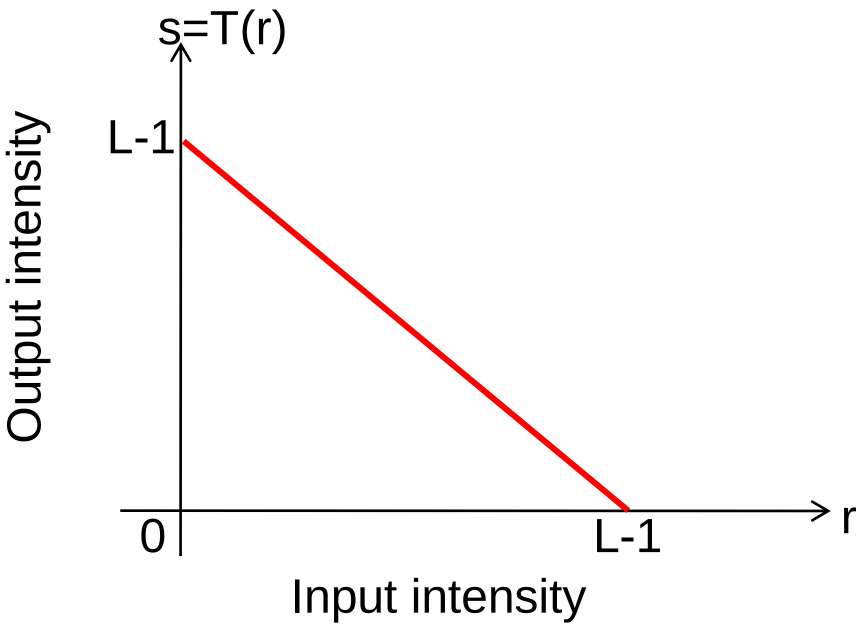


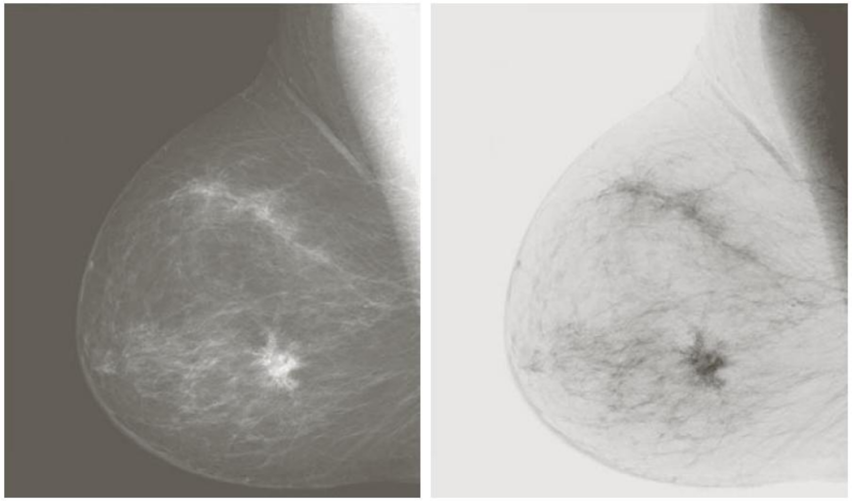
Thresholding Function

Intensity Transformation Function

Image Negative

Let the image has an intensity level in the range [0 L-1], then the intensity transformation is given by s=L-1-r

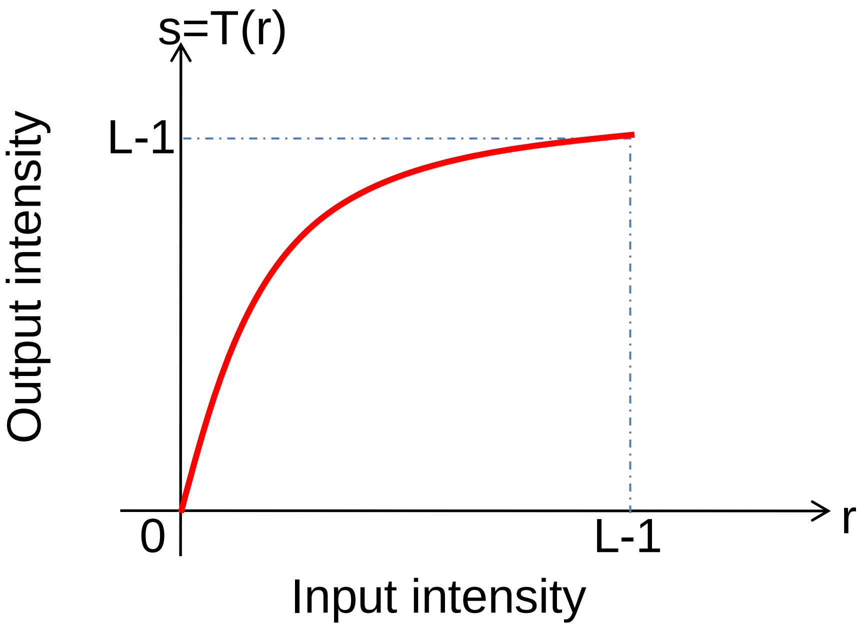




Intensity Transformation Function

Log Transformations

For an image having intensity ranging from [0 L-1], log transformation is given by **s=c log(1+r) ,** where c is a constant



- Maps the narrow range of low intensity values of input levels to wider range of output levels.

- Higher range of high intensity input levels is mapped to narrow range of out put levels.

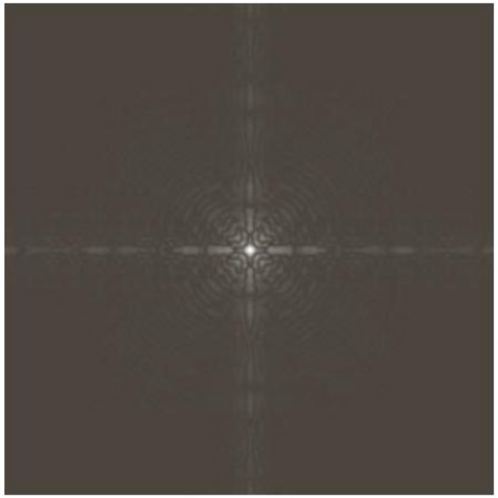
Intensity Transformation Function

Log Transformations

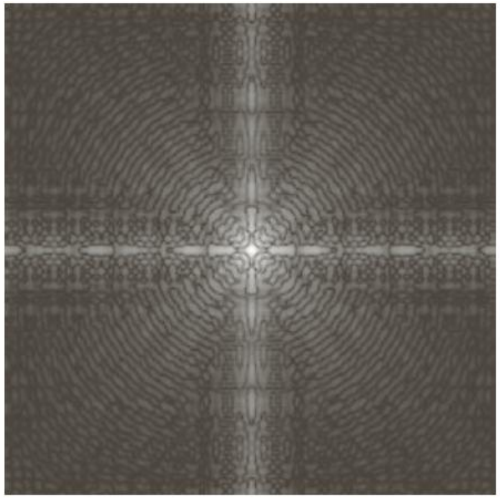
➢The Log function has the important characteristic that it compresses the dynamic range of images with large variation in the pixel value.

Classical example is displaying Fourier spectrum.

▪Fourier spectrum has the values in the range 0 to 1.5x106. These values are scaled linearly for the display in 8 bit system.



Fourier spectrum

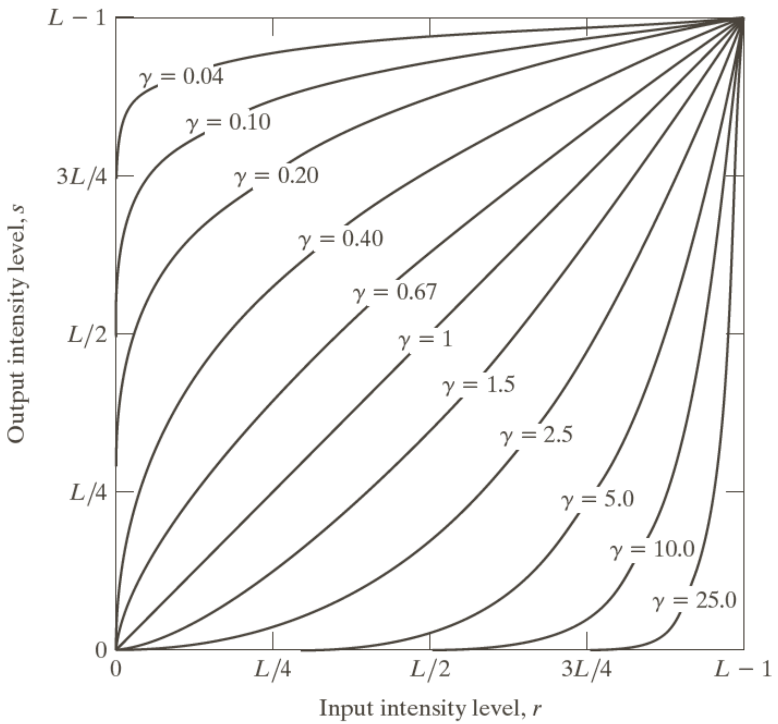


Log transformation with c=1

Intensity Transformation Function

Power-law (Gamma) Transformations

This has the basic form $s = c\, r^{\gamma}$,where c and γ are positive constants



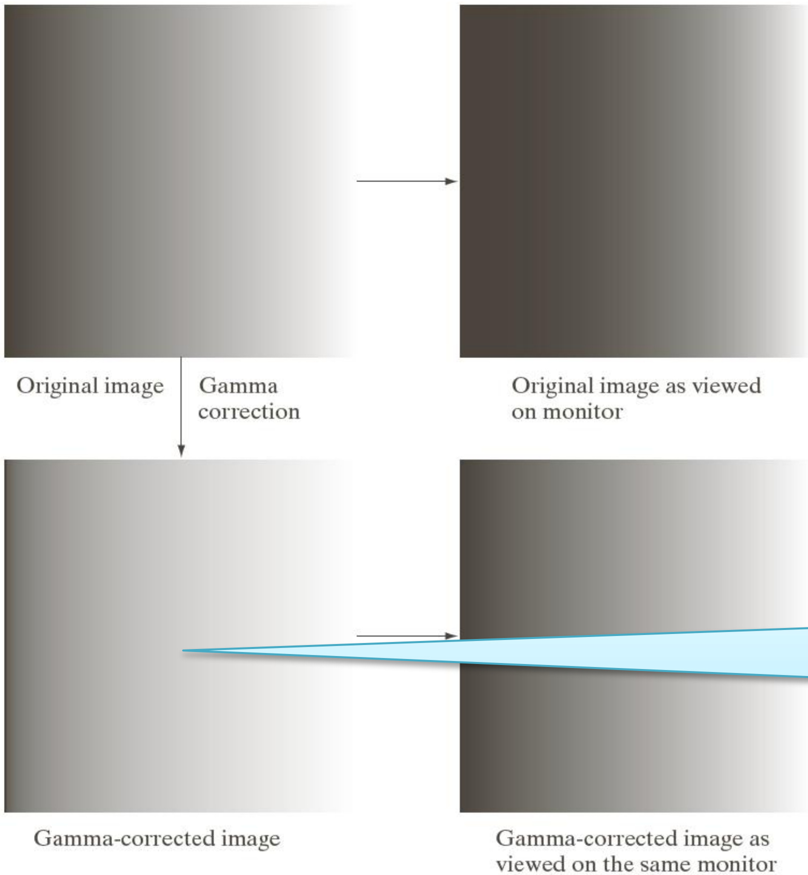
Plot for c=1

Fractional values of γ maps a narrow range of dark input values into a wider range of output values. Opposite of this also true for higher values of input levels.

These are also called as gamma correction due to the exponent in the power law equation.

Intensity Transformation Function

Power-law (Gamma) Transformations



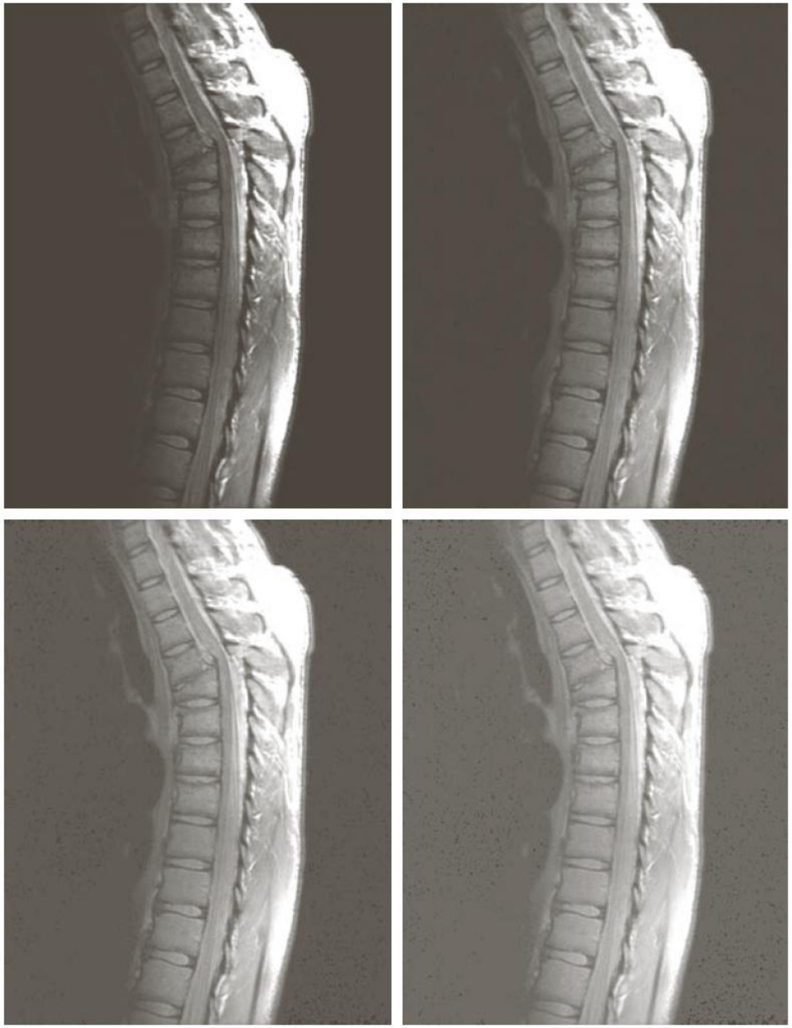
CRT device have an intensity to voltage response that is a power function with exponent varying from approximately 1.8 to 2.5. Such display system would produce images that are darker than intended.

Pre-processed image with gamma correction $s=r^{1/2.5}=r^{0.4}$. before input to the display device

Intensity Transformation Function

- Gamma correction is very important when to reproduce an image exactly on a display system.
- Power-law transformations are also used in general purpose contrast manipulation.

```
close all
clear all;
clc;
[filename, pathname] = uigetfile('*.tif');
im = imread([pathname filename]);
imshow(im);
im1=double(im).^0.3;
im1=mat2gray(im1);
figure,imshow(im1);
```



Intensity Transformation Function

MRI of fractured human spine

Result of a transformation for γ=0.6

Result of a transformation for γ=0.4

Result of a transformation for γ=0.3

Intensity Transformation Function

Arial image

Result of a transformation for c=1 and $\gamma=3$

Intensity Transformation Function

Result of a transformation for c=1 and $\gamma=4$

Result of a transformation for c=1 and $\gamma=5$

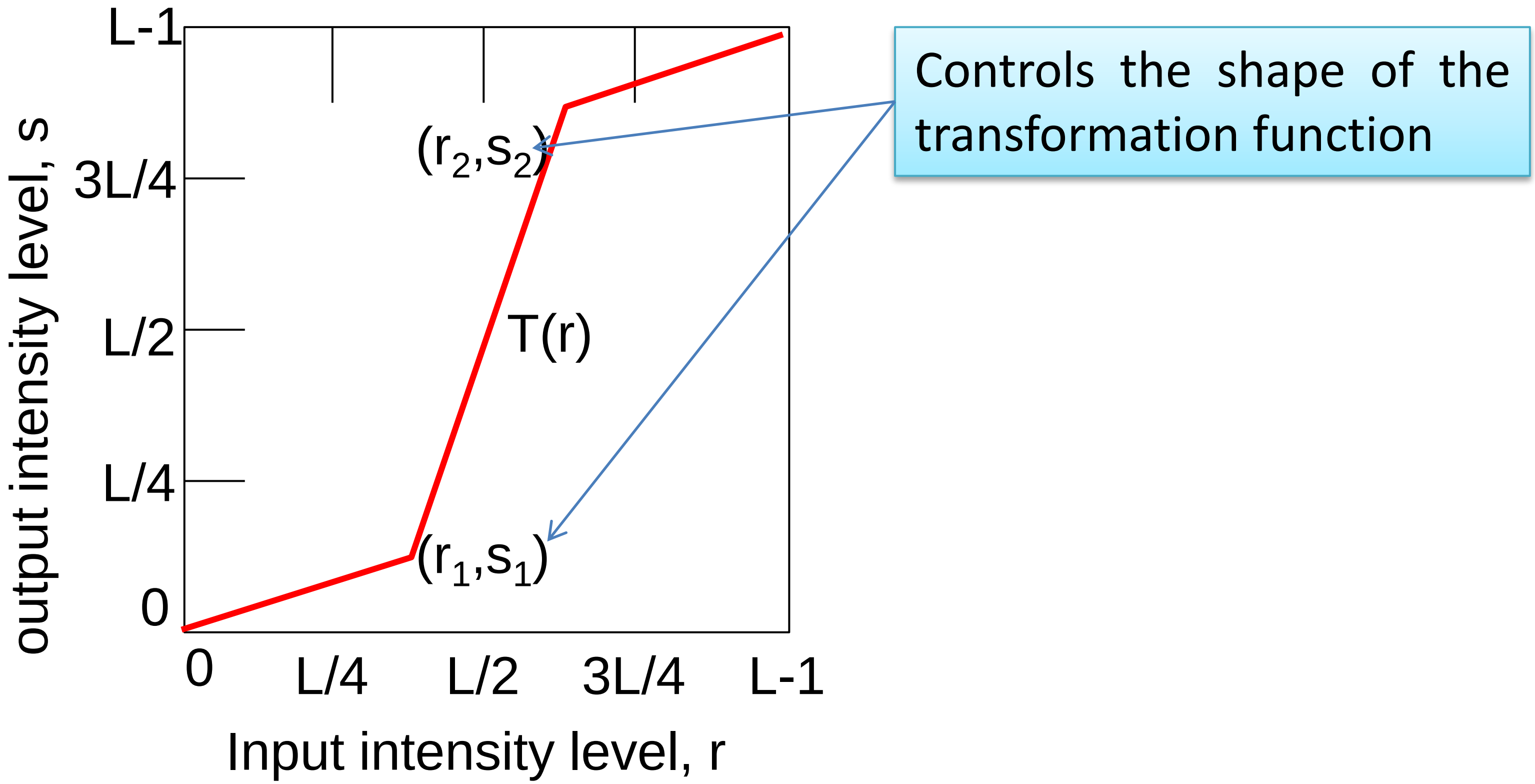
Piecewise Linear transformation functions

Contrast stretching

- Low contrast images result from the following
  - Poor illumination
  - lack of dynamic range in the imaging sensor
  - Wrong settings of the lens aperture during acquisition
- It is a process that expands the range of intensity levels in an image so that it spans full intensity range of the recording medium or display device

Piecewise Linear transformation functions

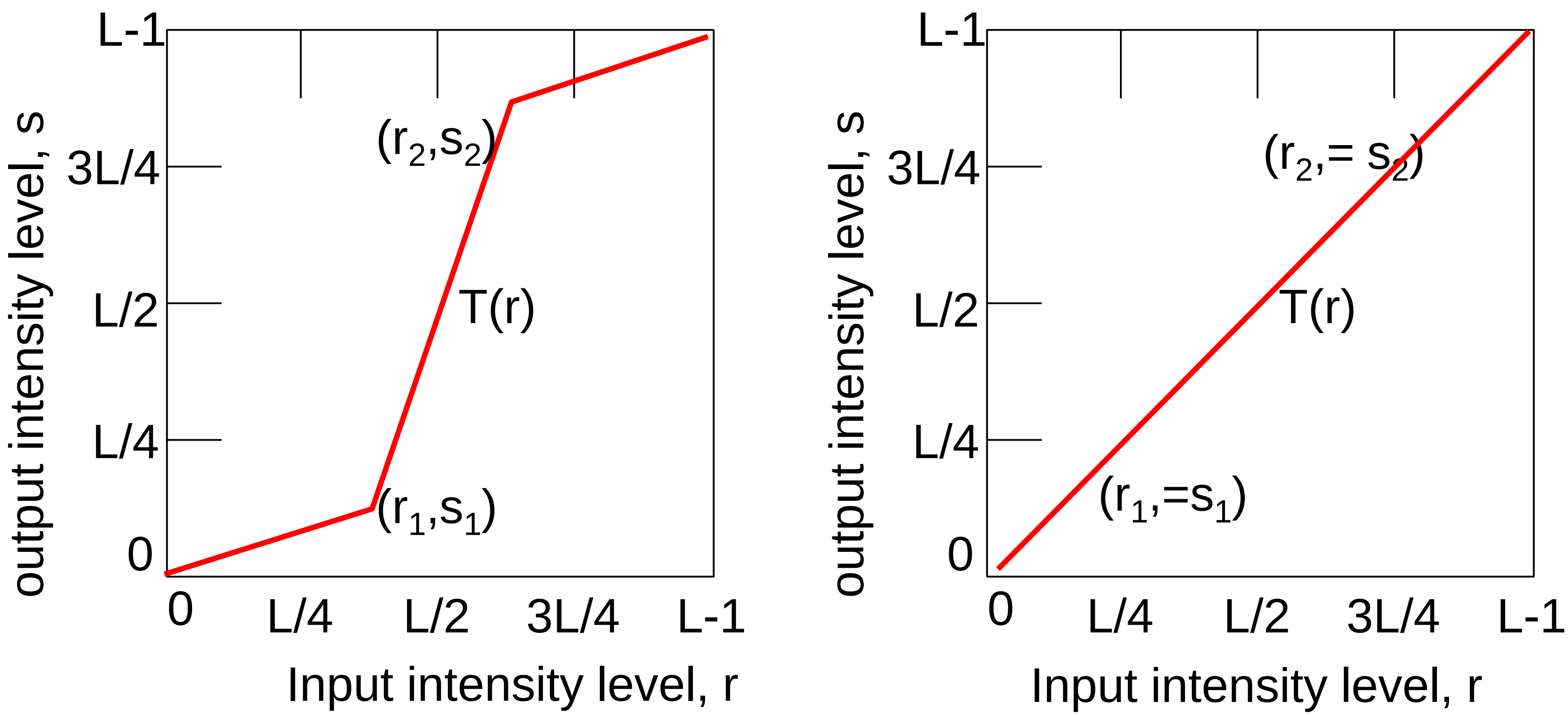
Contrast stretching



Piecewise Linear transformation functions

Contrast stretching

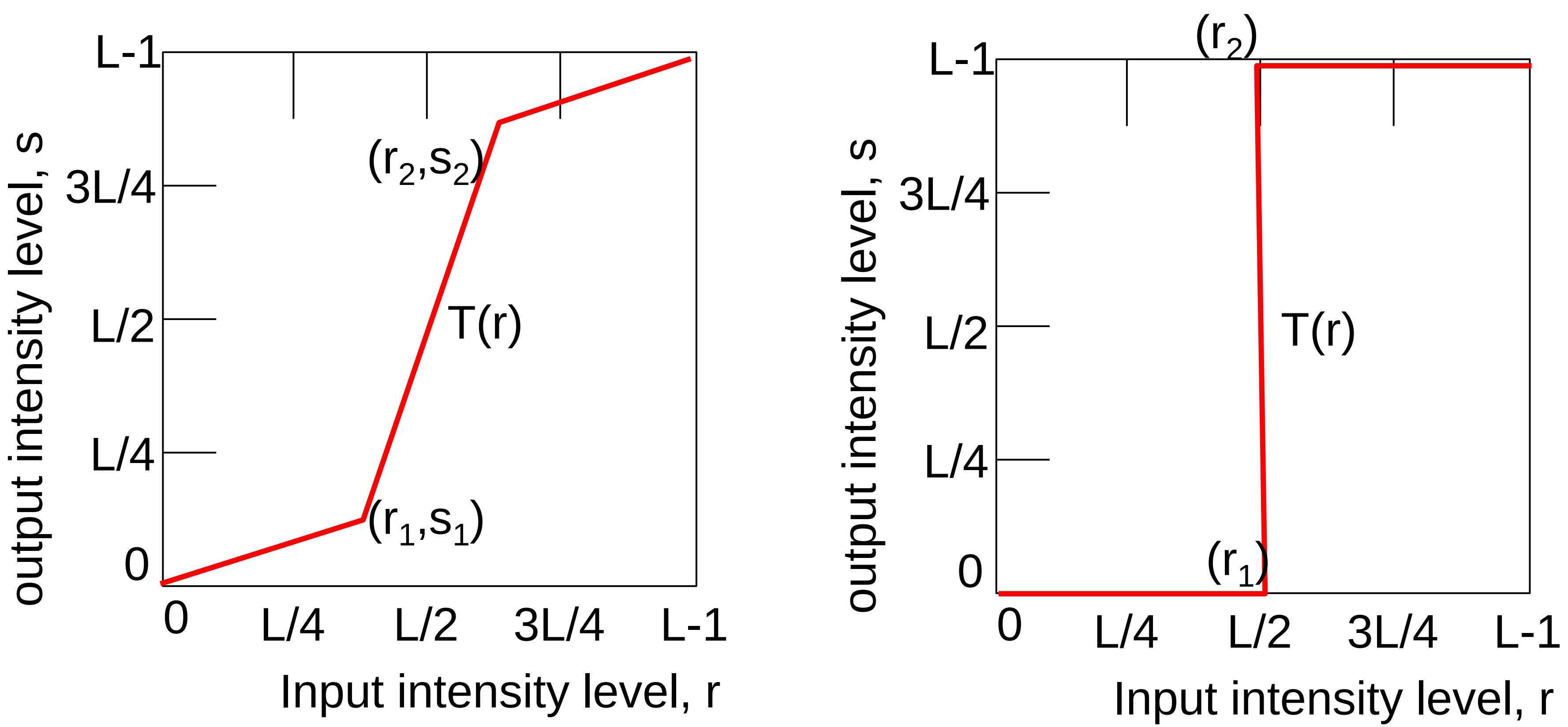
Suppose $r_1=s_1$ and $r_2=s_2$



Piecewise Linear transformation functions

Contrast stretching

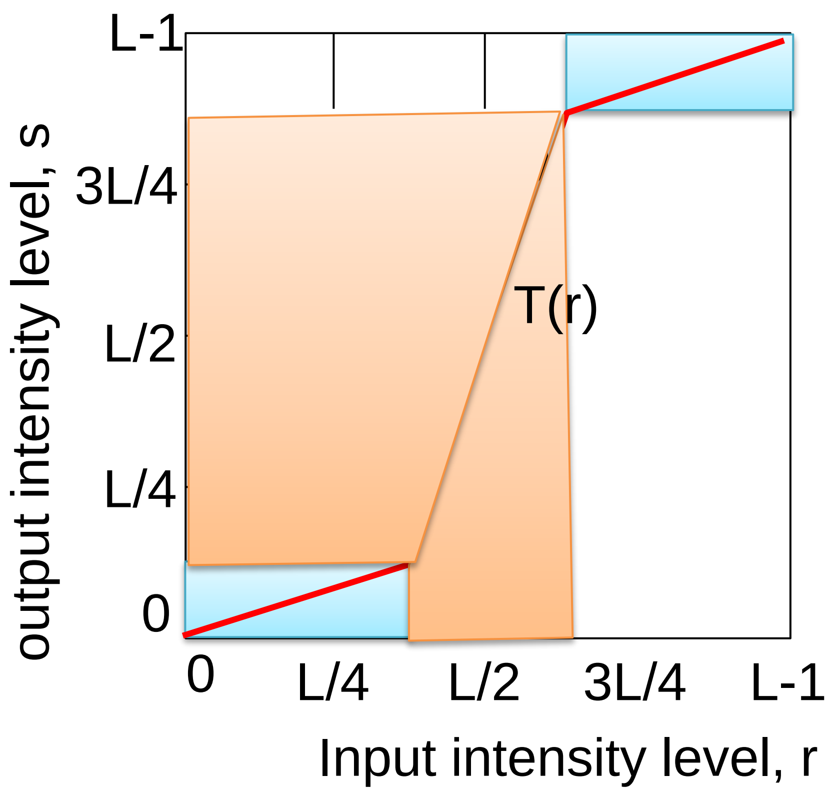
Suppose $r_1=r_2$ and $s_1=0$ and $s_2=L-1$



Piecewise Linear transformation functions

Contrast stretching

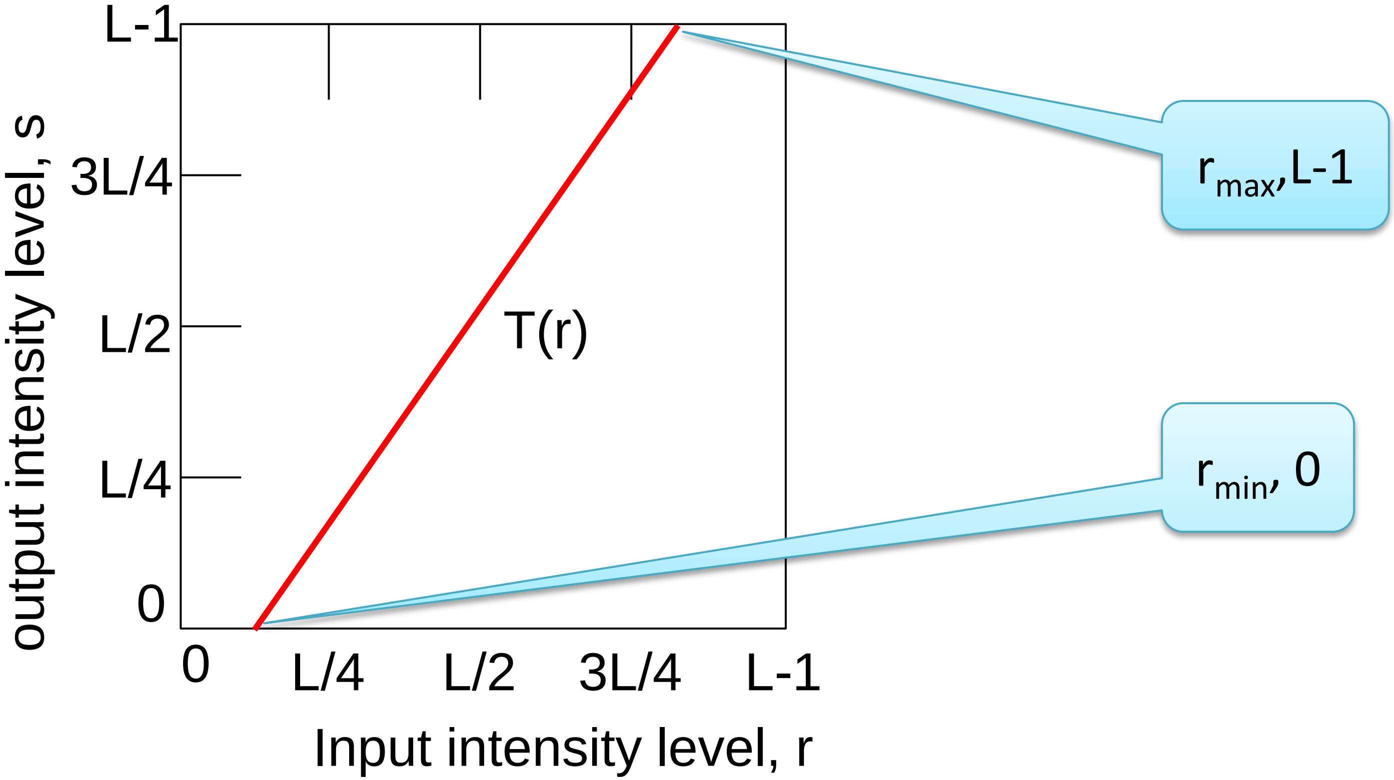
Intermediate values of (r_1,s_1) and (r_2,s_2) produces various degree of spread in the intensity



Piecewise Linear transformation functions

Contrast stretching (Example)

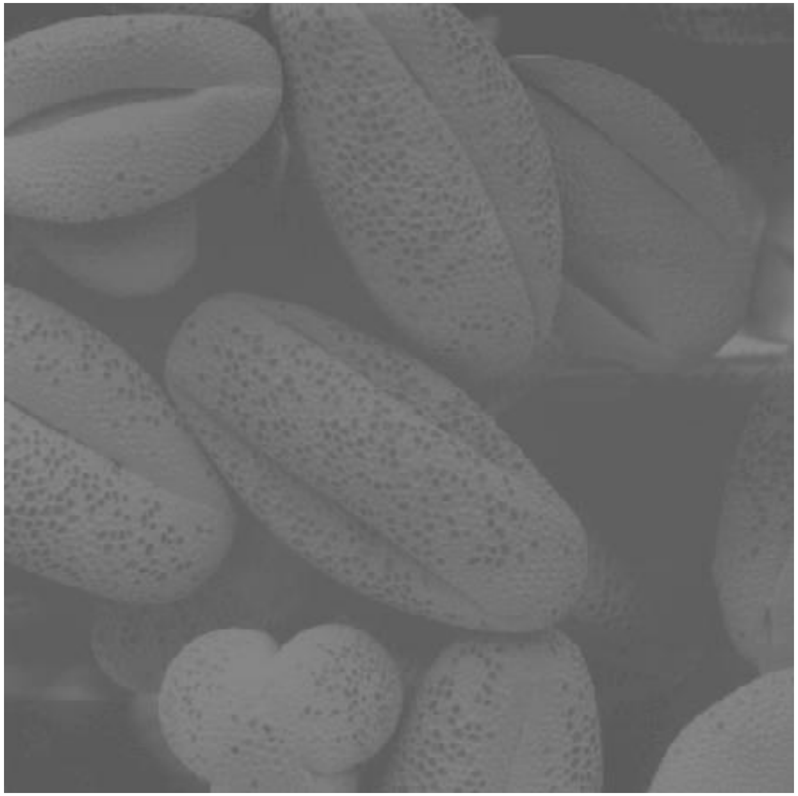
$(r_1,s_1)=(r_{min}, 0)$ and $(r_2,s_2)=(r_{max},L-1)$



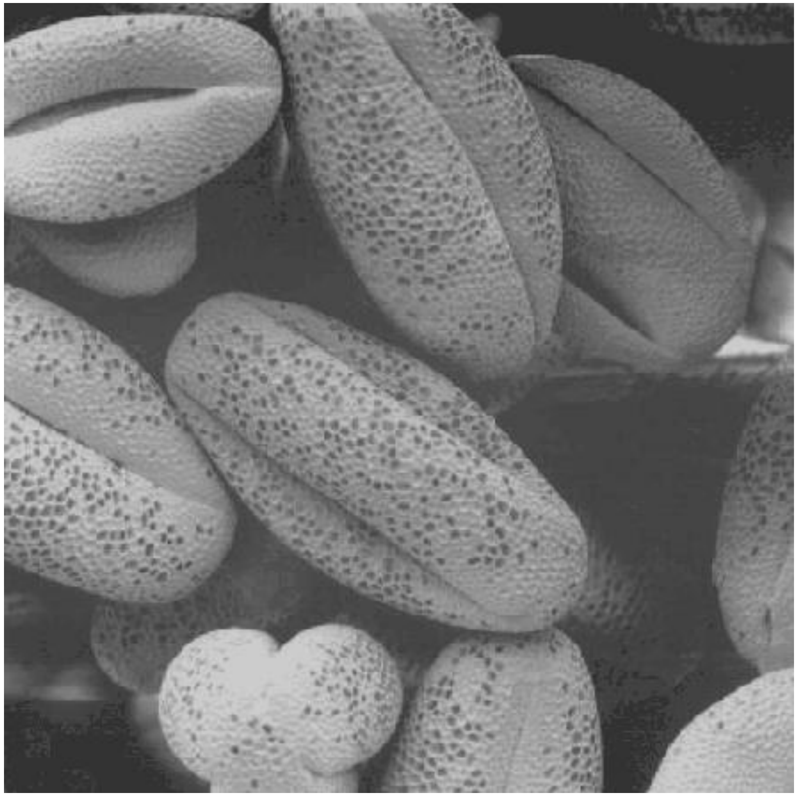
Piecewise Linear transformation functions

Contrast stretching (Example)

POOR CONTRAST IMAGE



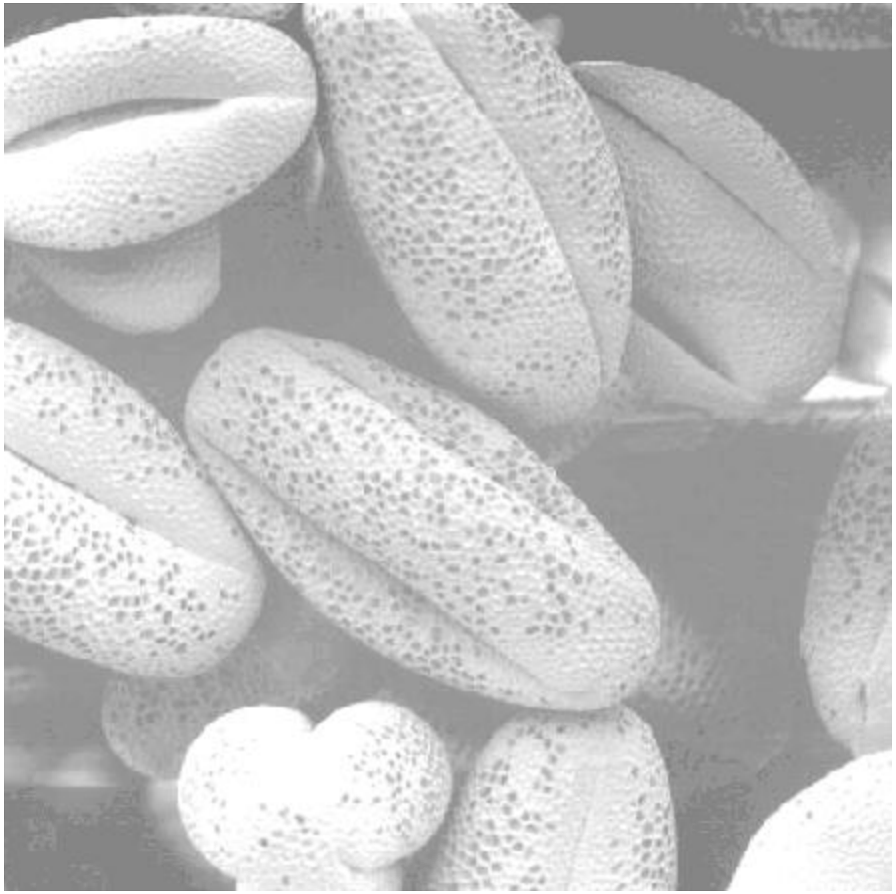
CONTRAST STRETCHED IMAGE



Piecewise Linear transformation functions

Contrast stretching (Example)

CONTRAST STRETCHED IMAGE



Piecewise Linear transformation functions

Contrast stretching (Example MATLAB PROGRAM)

```
close all
clear all;
clc;
[filename, pathname] = uigetfile('*.tif');
im = imread([pathname filename]);
imshow(im);
title('POOR CONTRAST IMAGE');
J = imadjust(im,[0.2 0.5],[0 1]);
figure,imshow(J);
title('CONTRAST STRETCHED IMAGE');
```

Piecewise Linear transformation functions

Contrast stretching (Example)

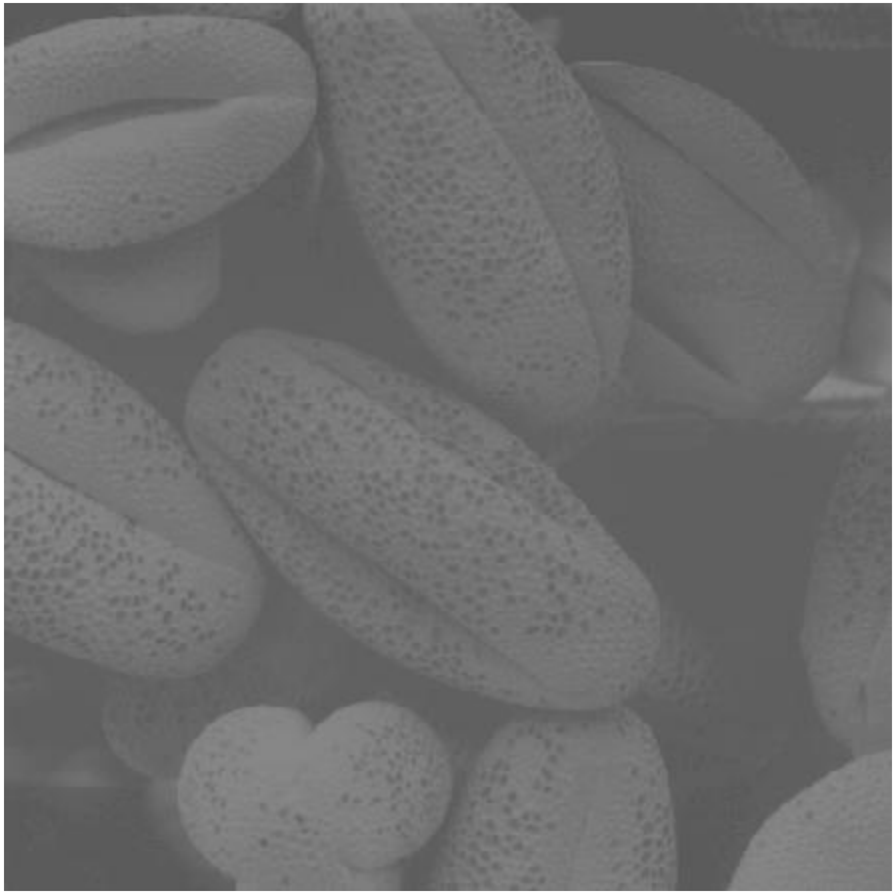
```
close all
clear all;
clc;
[filename, pathname] = uigetfile('*.tif');
im = imread([pathname filename]);
imshow(im);
title('POOR CONTRAST IMAGE');

K=im2bw(im,0.42);
figure,imshow(K)
```

Piecewise Linear transformation functions

Contrast stretching (Example)

POOR CONTRAST IMAGE



Piecewise Linear transformation functions

Intensity Level slicing

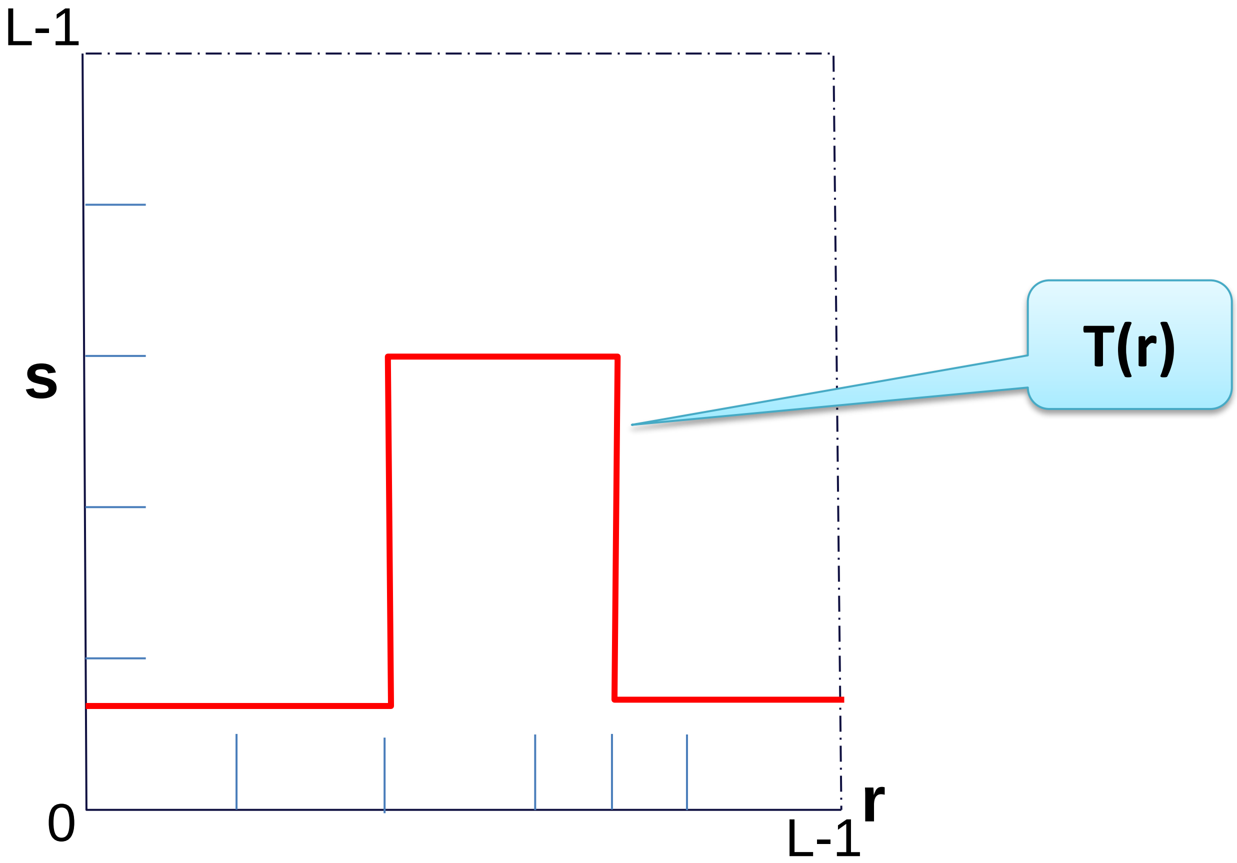
- Highlighting specific range of intensities

Example :

- Enhancing features such as masses of water in the satellite imagery
- Enhancing flaws in X-ray images.

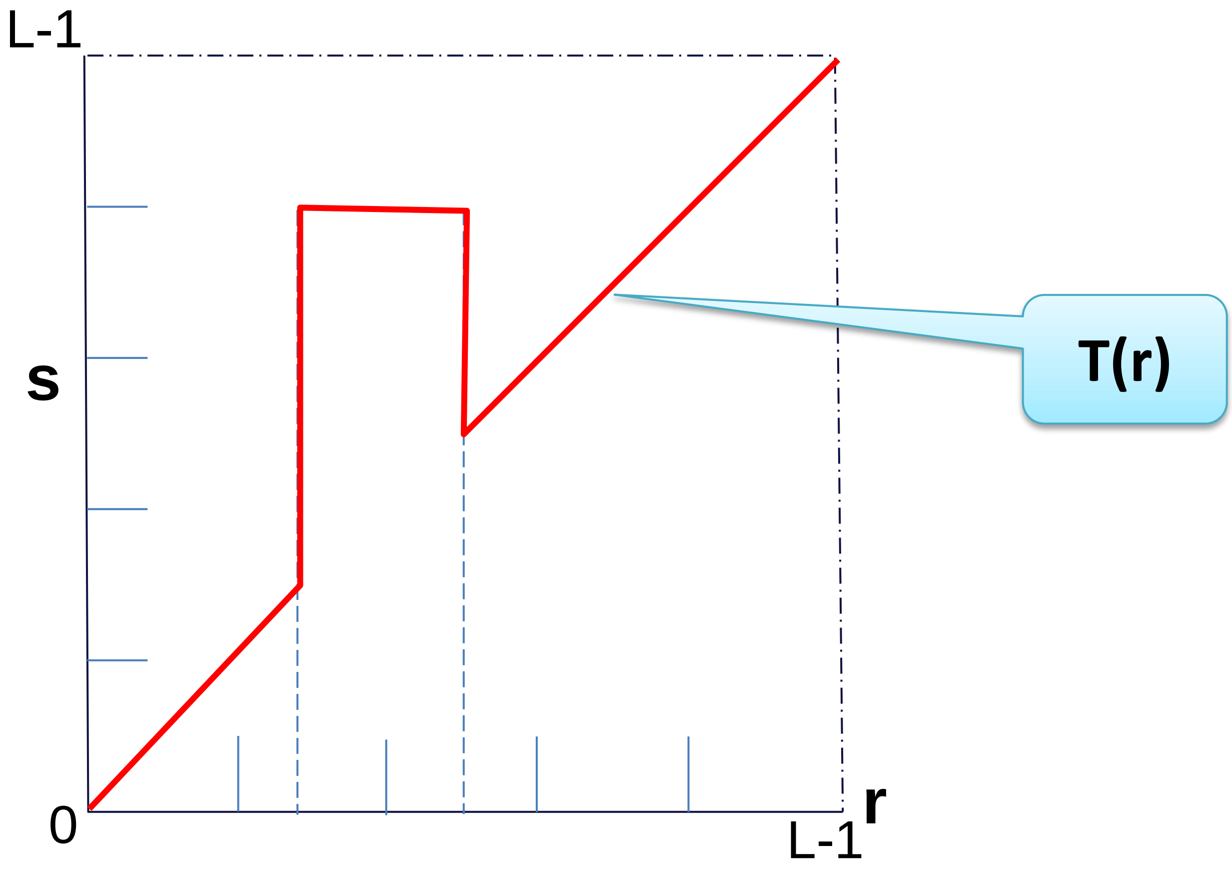
Piecewise Linear transformation functions

Intensity Level slicing



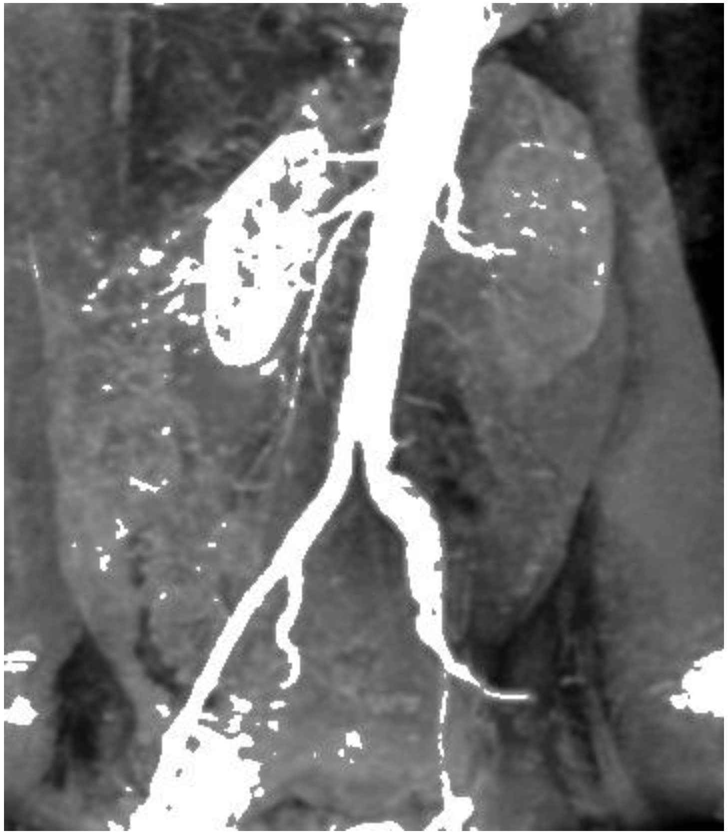
Piecewise Linear transformation functions

Intensity Level slicing



Piecewise Linear transformation functions

Intensity Level slicing (Example)



Piecewise Linear transformation functions

Intensity Level slicing (Example)

```
clear all ;
clc;
[filename, pathname] = uigetfile('*.tif');
im = imread([pathname filename]);
z=double(im);
[row,col]=size(z);
for i=1:1:row
for j=1:1:col
if((z(i,j)>142)) && (z(i,j)<250)
z(i,j)=255;
else
z(i,j)=im(i,j);
end
end
end
figure(1); %----------Original Image------------%
imshow(im);
figure(2); %----------Gray Level Slicing With Background------------%
imshow(uint8(z));
```

Piecewise Linear transformation functions

Intensity Level slicing (Example)

Piecewise Linear transformation functions

Intensity Level slicing (Example)

```
clear all ;
clc;
[filename, pathname] = uigetfile('*.tif');
im = imread([pathname filename]);
z=double(im);
[row,col]=size(z);
for i=1:1:row
for j=1:1:col
if((z(i,j)>142)) && (z(i,j)<250)
z(i,j)=255;
else
z(i,j)=0;
end
end
end
figure(1); %-----------Original Image------------%
imshow(im);
figure(2); %-----------Gray Level Slicing With Background------------%
imshow(uint8(z));
```

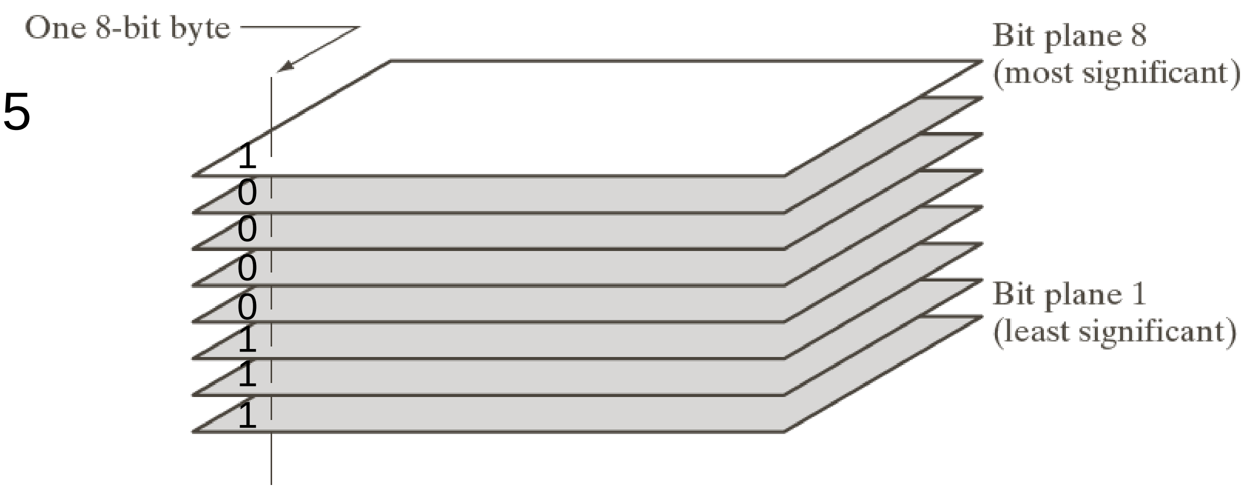
Piecewise Linear transformation functions

Bit Plane slicing (Example)

- Each pixels are digital number comprising of bits
- For a 256 level gray-scale image there are 8 bits for each pixel
- We can highlight the contribution of these bits to total image appearance

Example pixel value =135

1 0 0 0 0 1 1 1



Piecewise Linear transformation functions

Bit Plane slicing (Example)

An 8 bit gray scale image

Contribution of bit plane 8

Histogram Processing

Let the intensity level in the image be in the range from [0 L-1]

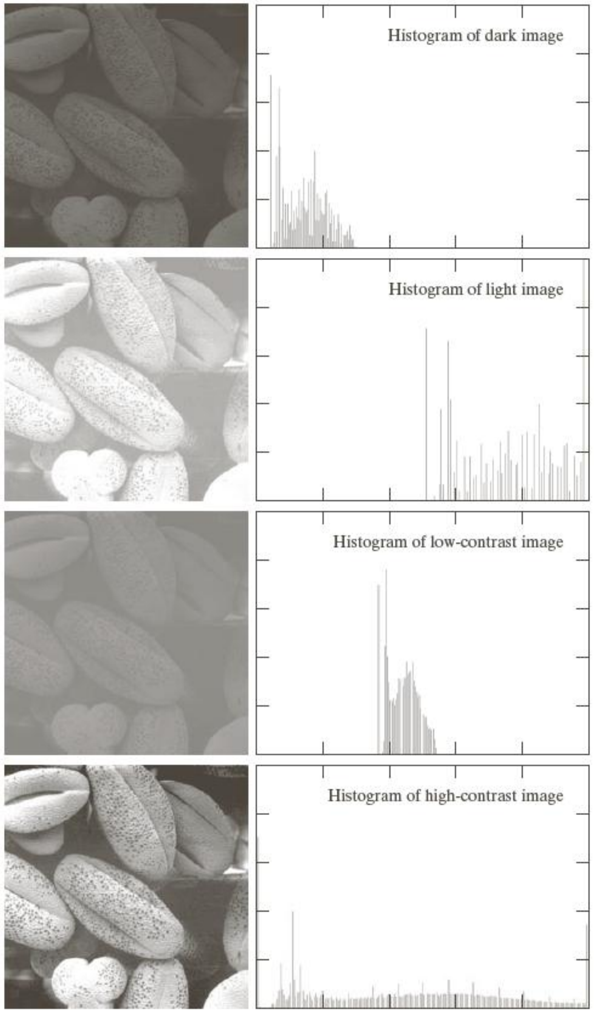
Histogram is a discrete function $h(r_k)=n_k$, where r_k is the k^{th} intensity value and n_k is the number of pixels in the image with pixel level r_k.

This histogram is normalized by dividing each component by total number of pixels in the image. Thus normalized histogram is given by,

$$p(r_k) = \frac{n_k}{MN} \quad for \;\; k = 0,1,2,3......L-1$$

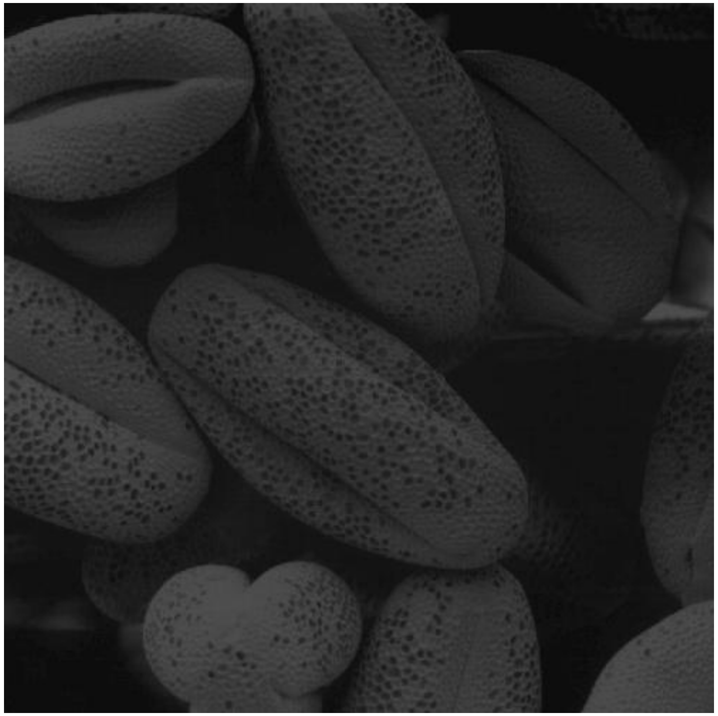
$p(r_k)$ is an estimate of the probability of occurrence of intensity level rk in an image. (Sum all the components=1)

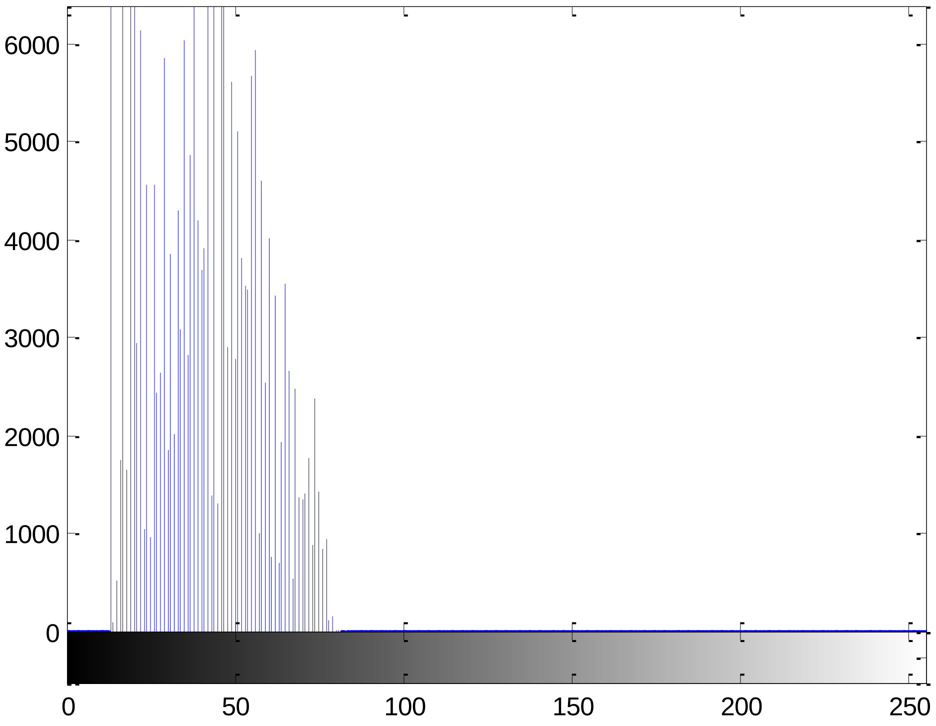
Histogram Processing



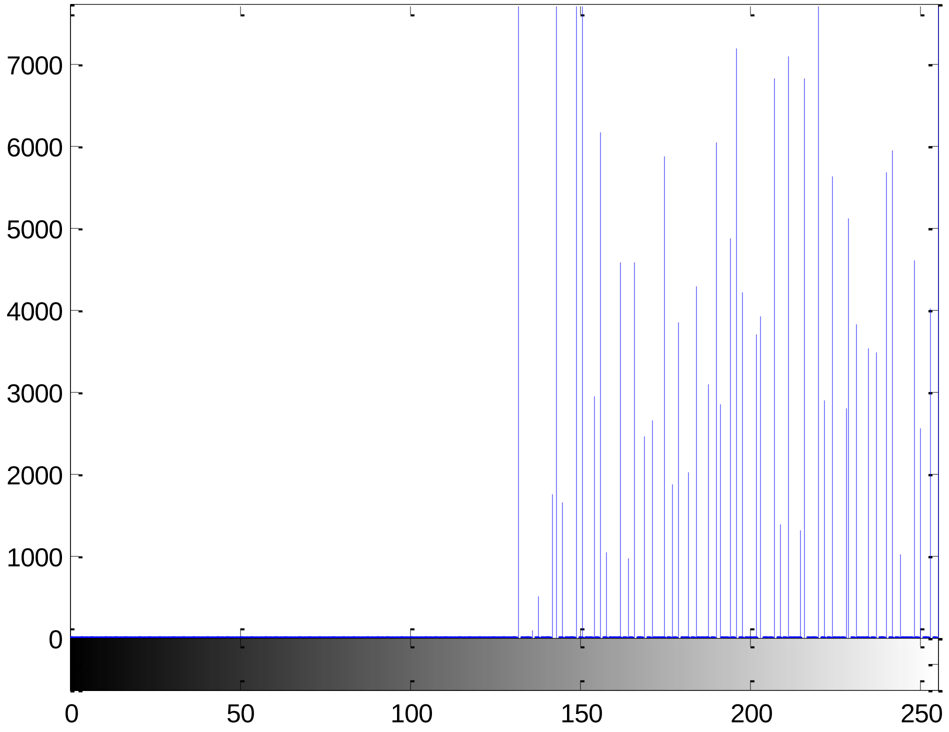

```
clear all ;
clc;
[filename, pathname] = uigetfile('*.tif');
im = imread([pathname filename]);
imshow(im);
figure, imhist(im);
```

Histogram Processing

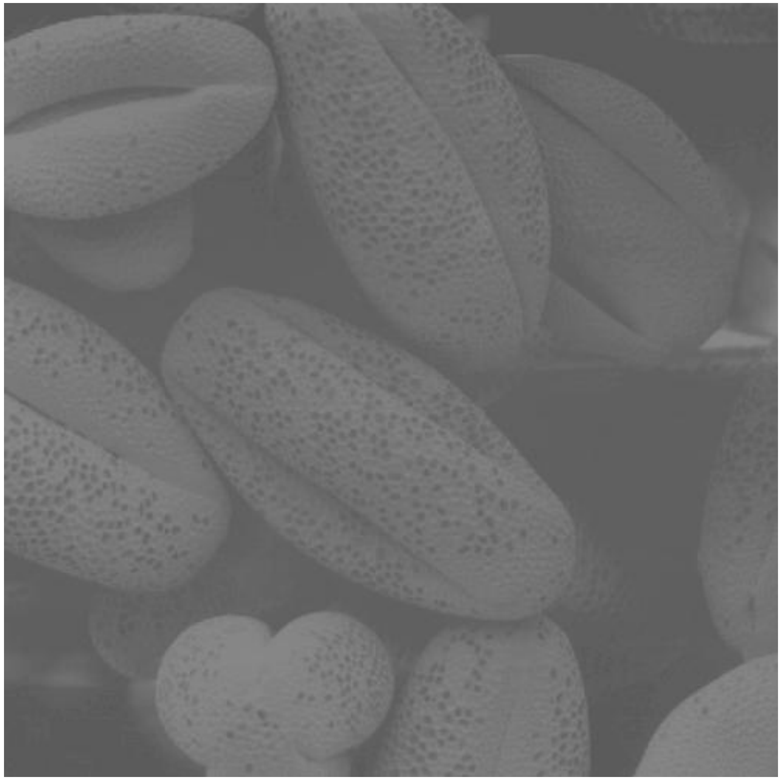


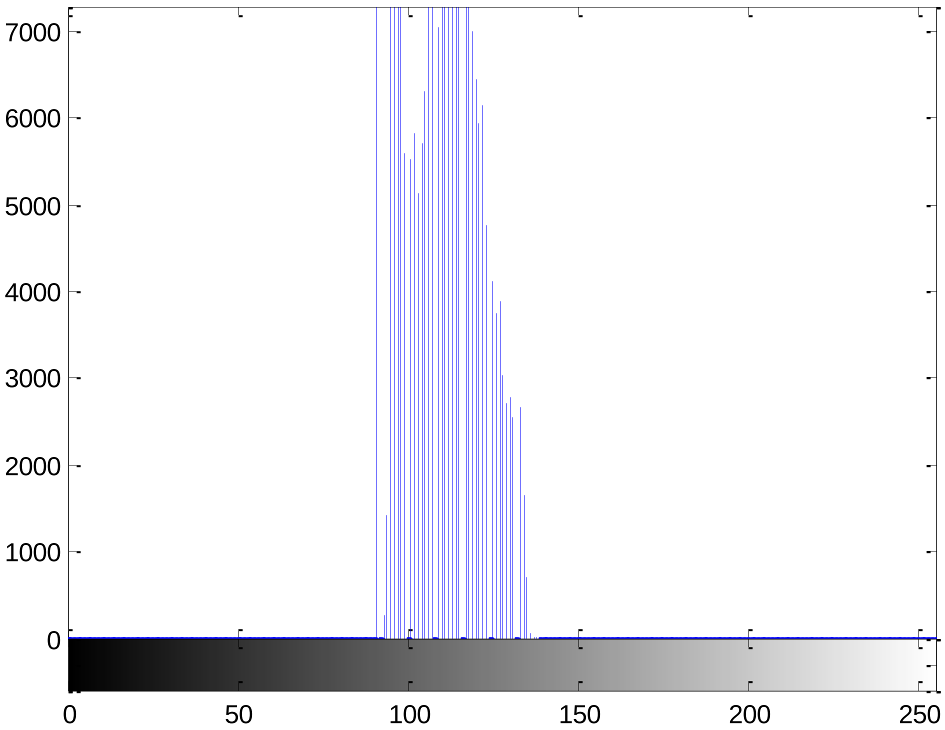


Histogram Processing

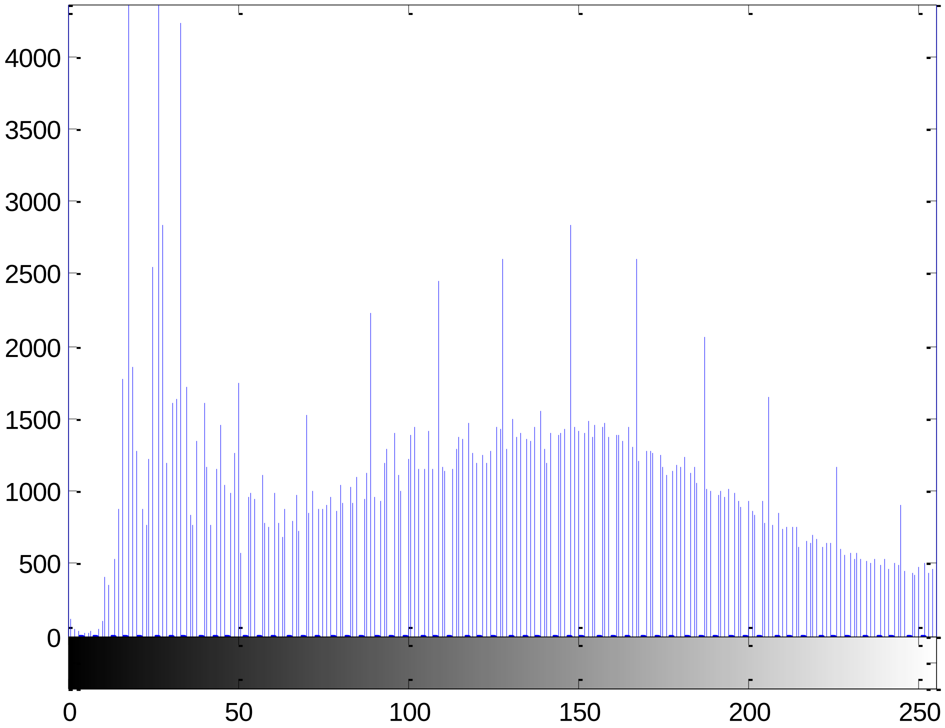


Histogram Processing





Histogram Processing



Histogram Equalization

Let us denote r [0 L-1] as intensities of the image to be processed

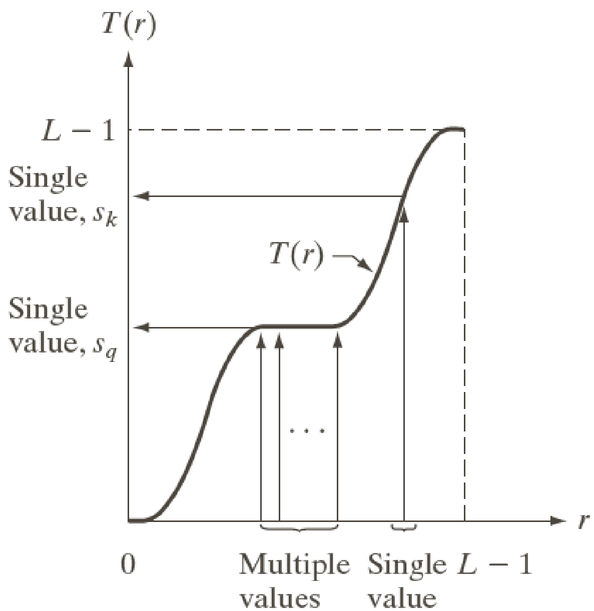
r=0 corresponding to black and r=L-1 representing white.

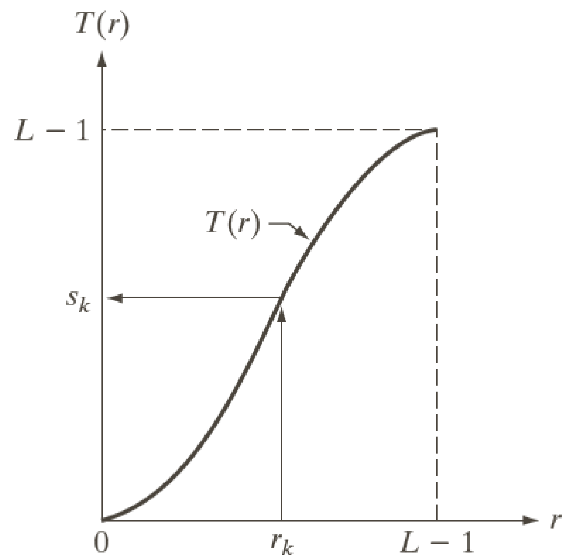
Let the intensity transformation is defined by s=T(r) , where $0 \le r \le L-1$

- T(r) is monotonically increasing function in the interval $0 \le r \le L-1$
- $0 \le T(r) \le L-1$ and $0 \le r \le L-1$

Suppose we use the inverse operation as $r=T^{-1}(s)$, then the condition should be strictly monotonically increasing.

Histogram Equalization





Satisfies the condition T(r) is monotonically increasing function in the interval 0 ≤ r ≤ L-1 and 0 ≤ T(r) ≤ L-1 and 0 ≤ r ≤ L-1

Strictly monotonically increasing

Mapping is one to one in both the directions.

Histogram Equalization

- Let us consider intensity levels in the image as random variables in the interval 0 to L-1.
- Let us defined the Probability Density Function (PDF) as $p_r(r)$ and $p_s(s)$ for r and s respectively.
- If $p_r(r)$ and T(r) is known, where T(r) is continuous and differentiable over the PDF range , then

$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right|$$

- The transformation function is of the form

$$s = T(r) = (L-1)\int_0^r p_r(w)dw$$

Cumulative Distribution Function (CDF) of random variable r

Histogram Equalization

- The transformation function of this form satisfies both the conditions we have seen.

Now let us compute $p_s(s)$, we know $s=T(r)$

$$\frac{ds}{dr} = \frac{dT(r)}{dr}$$

$$= (L-1)\frac{d}{dr}\left[\int_0^r p_r(w)dw\right]$$

$$= (L-1)p_r(r)$$

Substituting this for $p_s(s)$, we get

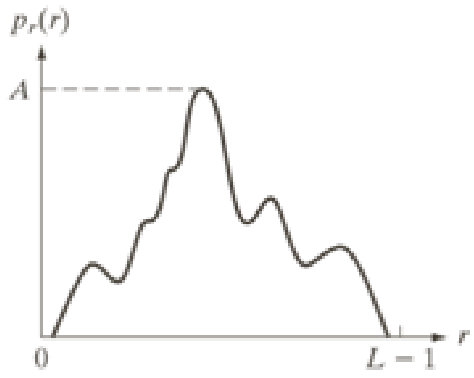
$$p_s(s) = p_r(r)\frac{dr}{ds}$$

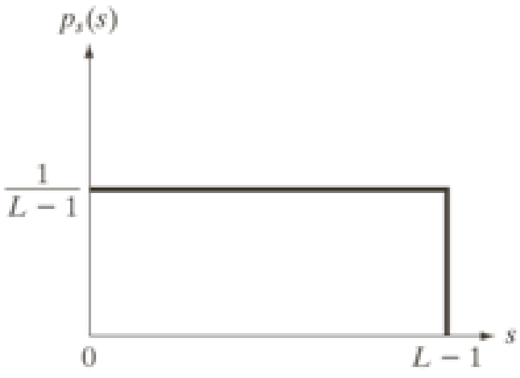
$$= p_r(r)\left|\frac{1}{(L-1)p_r(r)}\right|$$

$$= \frac{1}{L-1} \quad 0 \le s \le L-1$$

Histogram Equalization

- Which is a uniform probability density function, this means , performing intensity transformation yields a random variable s characterized by uniform PDF.
- It can be noted that T(r) depends on $p_r(r)$ but ps(s) is always uniform and independently of the form of $p_r(r)$.





Histogram Equalization (Example)

Suppose intensity values in an image have the PDF

$$p_r(r) = \begin{cases} \dfrac{2r}{(L-1)^2} & \text{for } 0 \le r \le (L-1) \\ 0 & \text{otherwise} \end{cases}$$

$$s = T(r) = (L-1)\int_0^r p_r(w)\,dw = \frac{2}{(L-1)}\int_0^r w\,dw = \frac{r^2}{(L-1)}$$

Suppose L=9 and pixel at location say (x,y) has the value $r=3$, then

$s=T(r)= r^2/9=1$

Histogram Equalization (Example)

The PDF of the intensities in the new image is

$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right| = \frac{2r}{(L-1)^2}\left|\left[\frac{ds}{dr}\right]^{-1}\right|$$

$$= \frac{2r}{(L-1)^2}\left|\left[\frac{d}{dr}\frac{r^2}{L-1}\right]^{-1}\right|$$

$$= \frac{2r}{(L-1)^2}\left|\frac{(L-1)}{2r}\right| = \frac{1}{L-1}$$

Assume r is positive and L>1

Result is uniform PDF

Histogram Equalization

innovate achieve lead

- For the discrete values of the histogram , we deal with summation instead of integration

$$p(r_k) = \frac{n_k}{MN} \quad k = 0,1,2,\ldots\ldots L-1$$

The discrete form of transformation is given by

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j)$$

$$= \frac{(L-1)}{MN}\sum_{j=0}^{k} n_j \quad k = 0,1,2,\ldots\ldots L-1$$

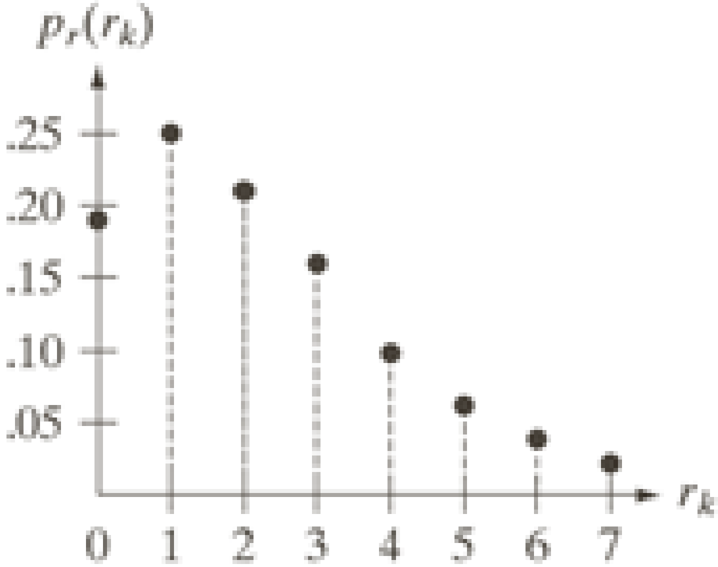
Histogram Equalization

- The input pixel r_k is mapped to output pixel s_k
- The transformation (mapping) T(r_k) is called as histogram equalization or histogram linearization.

Histogram Equalization (Example)

Let us consider a 3 bit image (L=8) of 64 x 64 (MN=4096), has the intensity distribution shown below.

r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02



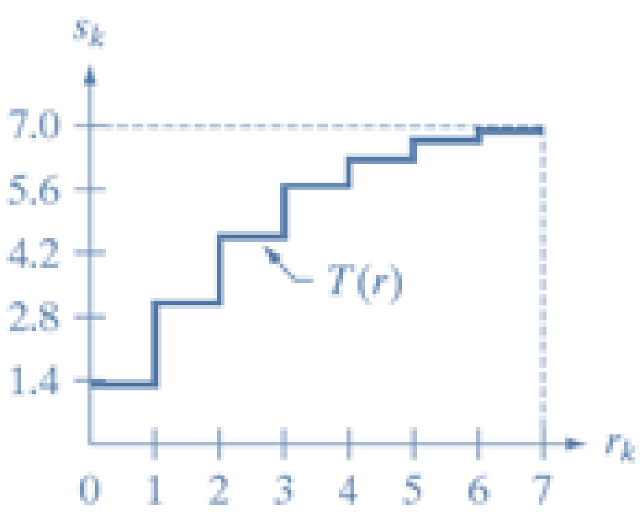
Histogram Equalization (Example)

From the equation of histogram equalization , we have

$$s_0 = T(r_0) = 7\sum_{j=0}^{0} p_r(r_j) = 7p_r(r_0) = 1.33$$

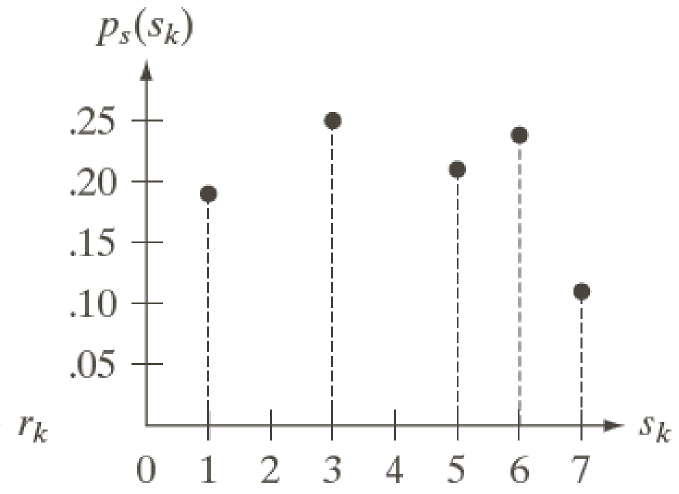
$$s_1 = T(r_1) = 7\sum_{j=0}^{1} p_r(r_j) = 7p_r(r_0) + 7p_r(r_1)$$

Similarly compute $s_2, s_3, s_4, s_5, s_6, s_7$



Histogram Equalization (Example)

s_0	1.33	1
s_1	3.08	3
s_2	4.55	5
s_3	5.67	6
s_4	6.23	6
s_5	6.65	7
s_6	6.86	7
s_7	7.00	7



Histogram matching (Specification)

- Histogram equalization is an automatic enhancement.
- Some times shape of the histogram can be specified based on the requirement.
- The method used to generate a processed image that has a specified histogram is called histogram matching or histogram specification

$$p(r_k) = \frac{n_k}{MN} \quad k = 0,1,2,......L-1$$

The discrete form of transformation is given by

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j)$$

$$= \frac{(L-1)}{MN}\sum_{j=0}^{k} n_j \quad k = 0,1,2,.......L-1$$

Histogram matching (Specification)

Let $p_z(z)$ is the specified PDF, which is going to be the PDF of the output image. So we have

$$G(z_q) = (L-1)\sum_{j=0}^{q} p_z(z_i) = s_k$$

Desired value $z_q = G^{-1}(s_k)$

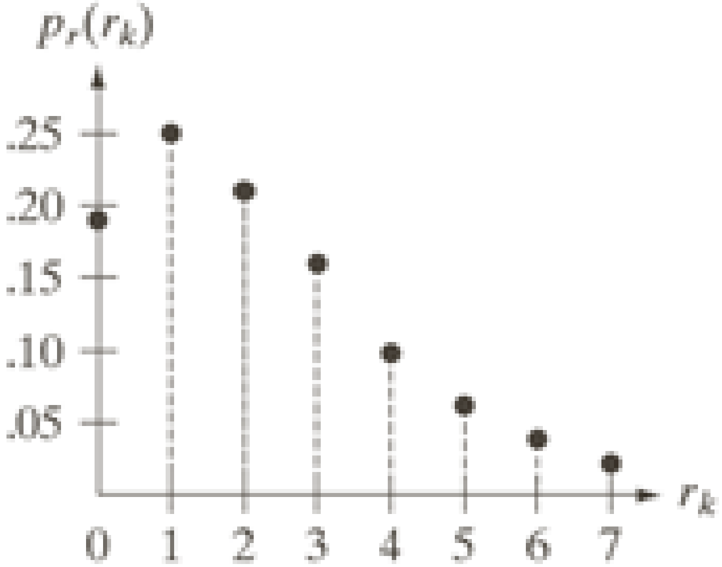
This will give value of z for each value of s, by performing mapping of s to z

Let us understand it by an example

Histogram matching (Specification)

Let us consider a 3 bit image (L=8) of 64 x 64 (MN=4096), has the intensity distribution shown below.

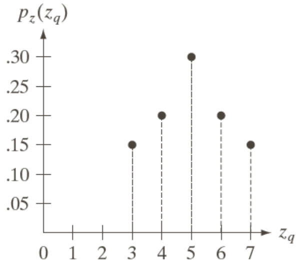
r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02



Histogram matching (Specification)

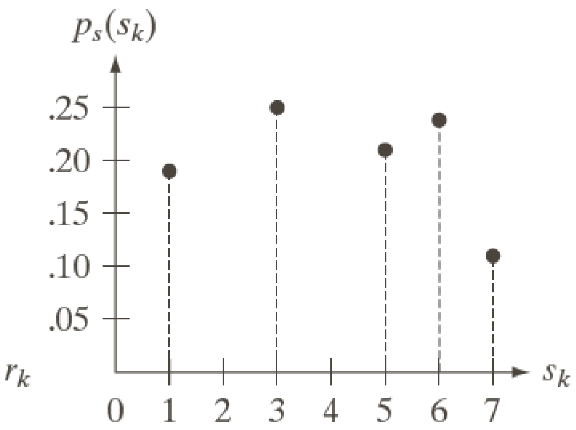
Specified histogram is given as follows

z_q	$P_z(z_q)$
$Z_0=0$	0.00
$Z_1=1$	0.00
$Z_2=2$	0.00
$Z_3=3$	0.15
$Z_4=4$	0.20
$Z_5=5$	0.30
$Z_6=6$	0.20
$Z_7=7$	0.15



STEP 1 : Scaled histogram-equalized values

s_0	1.33	1
s_1	3.08	3
s_2	4.55	5
s_3	5.67	6
s_4	6.23	6
s_5	6.65	7
s_6	6.86	7
s_7	7.00	7



Histogram matching (Specification)

STEP 2 : Compute all the values of transformation function G,

$$G(z_0) = 7\sum_{j=0}^{0} p_z(z_j)$$

$$G(z_1) = 7\sum_{j=0}^{0} p_z(z_j) = 7[p(z_0) + p(z_1)]$$

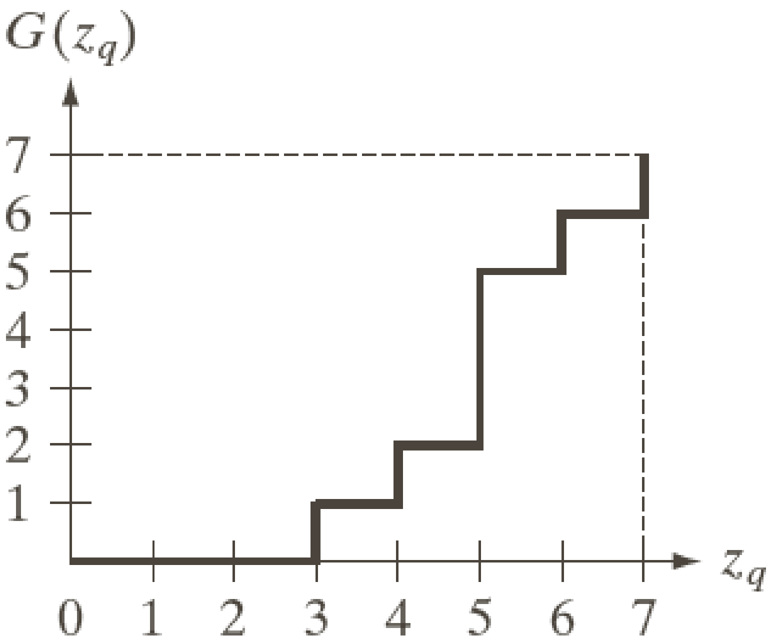
$$G(z_2) = 0.00 \quad G(z_3) = 1.05 \quad G(z_4) = 2.45 \quad G(z_5) = 4.55$$

$$G(z_6) = 5.95 \quad G(z_7) = 7.00$$

These fractional values are converted to integer values as shown

Histogram matching (Specification)

$G(z_0)$	0.00	0
$G(z_1)$	0.00	0
$G(z_2)$	0.00	0
$G(z_3)$	1.05	1
$G(z_4)$	2.45	2
$G(z_5)$	4.55	5
$G(z_6)$	5.95	6
$G(z_7)$	7.00	7



- The condition of strictly monotonic is violated

Histogram matching (Specification)

To handle this situation following procedure is used

Find the smallest value of z_q so that the value $G(z_q)$ is closest to s_k.

For example $s_0=1$, and $G(z_3)=1$, which is a perfect match for this case, here s_0 -> z_3 ,i. e every pixel whose value is 1 in the histogram equalized image is mapped to pixel valued 3 in the histogram specified image. Continuing this we get,

s_k	z_q
1	3
3	4
5	5
6	6
7	7