EE663
Image Processing
Histogram Equalization II

Dr. Samir H. Abdul-Jauwad
Electrical Engineering Department
King Fahd University of Petroleum & Minerals

Image Enhancement: Histogram Based Methods

What is the histogram of a digital image?

- The histogram of a digital image with gray values $r_0, r_1, \cdots, r_{L-1}$

is the discrete function

$$p(r_k) = \frac{n_k}{n}$$

n_k: Number of pixels with gray value r_k

n: total Number of pixels in the image

- The function $p(r_k)$ represents the fraction of the total number of pixels with gray value r_k.

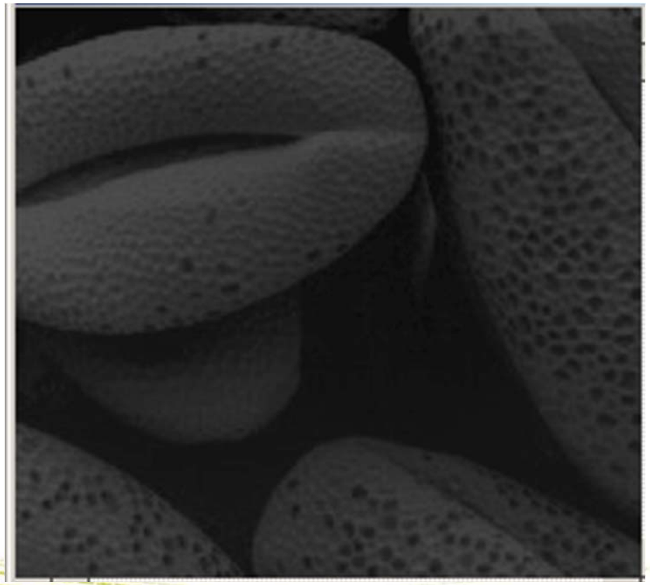
- Histogram provides a global description of the appearance of the image.

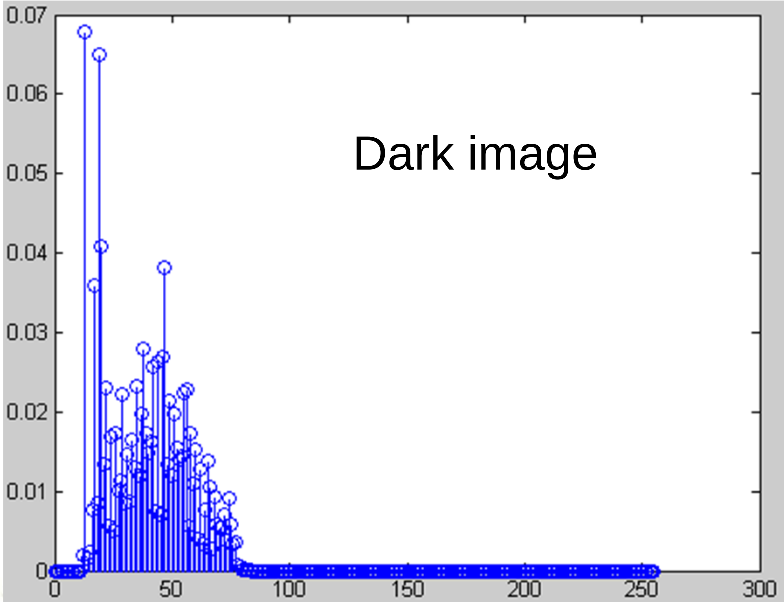
- If we consider the gray values in the image as realizations of a random variable R, with some probability density, histogram provides an approximation to this probability density. In other words,

$$\Pr(R = r_k) \approx p(r_k)$$

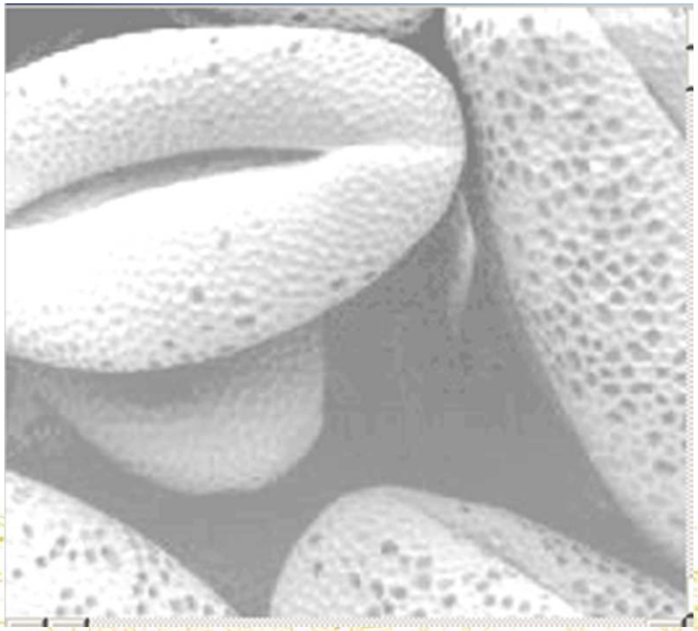
Some Typical Histograms

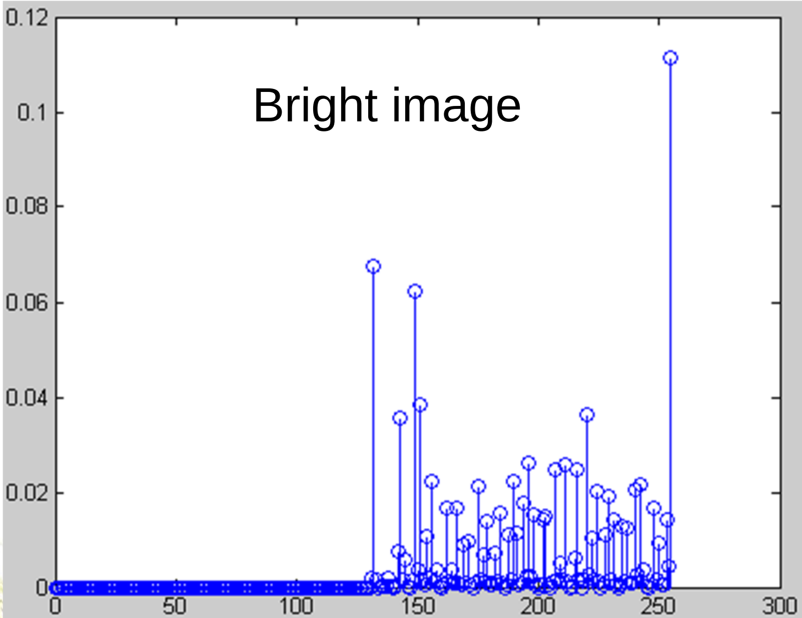
- The shape of a histogram provides useful information for contrast enhancement.

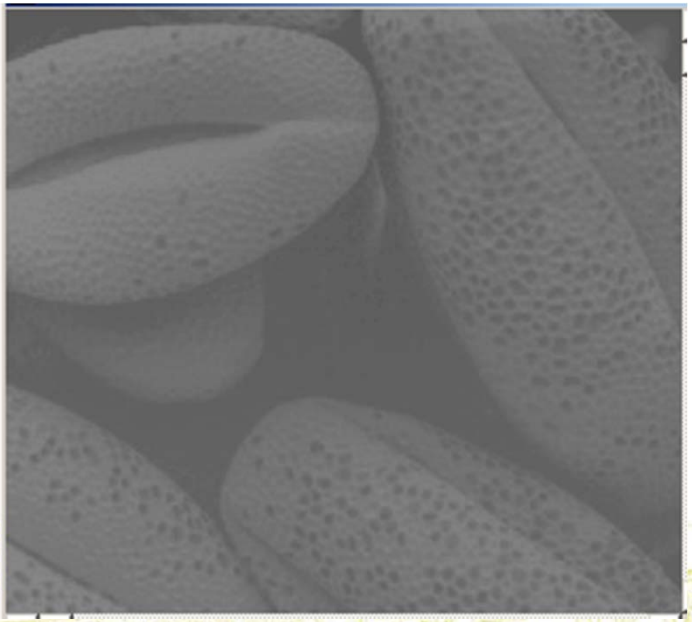


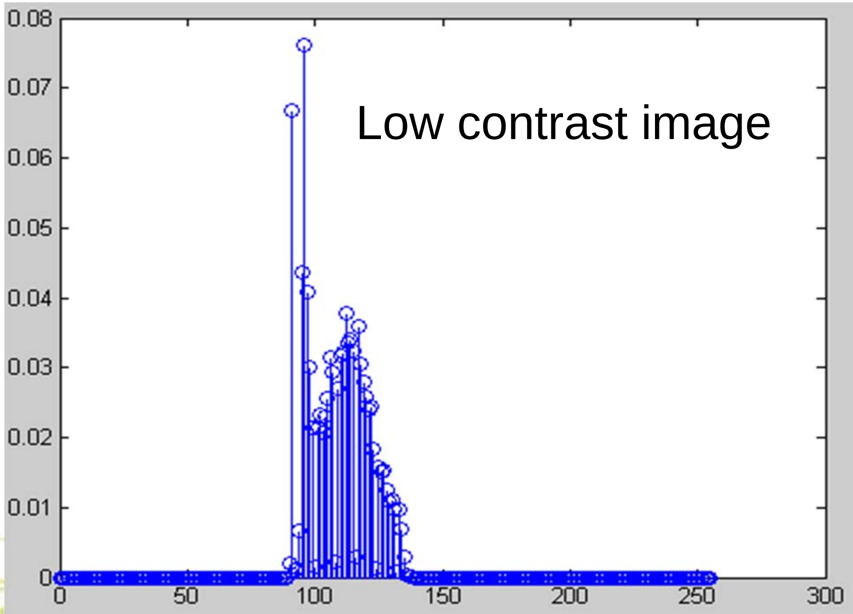



```
>> clear;
[ix,map]=imread('Fig3_15a.jpg');
imshow(ix)
figure;
ix=double(ix);
h=histogram(ix);
figure
stem(0:255,h);
```




Bright image
0.12
0.1
0.08
0.06
0.04
0.02
0
0
50
100
150
200
250
300




0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
0
50
100
150
200
250
300
Low contrast image

Histogram Equalization

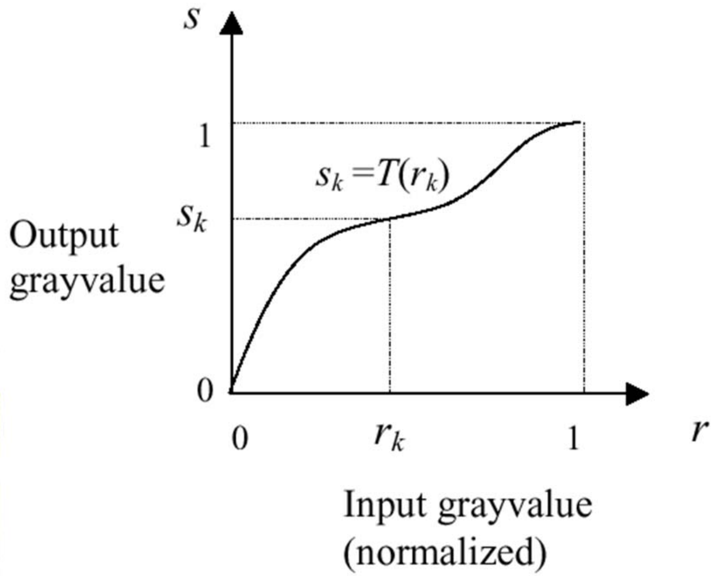
- What is the histogram equalization?

- The histogram equalization is an approach to enhance a given image. The approach is to design a transformation *T(.)* such that the gray values in the output is uniformly distributed in [0, 1].

- Let us assume for the moment that the input image to be enhanced has continuous gray values, with $r = 0$ representing black and $r = 1$ representing white.

- We need to design a gray value transformation $s = T(r)$, based on the histogram of the input image, which will enhance the image.

• As before, we assume that:
(1) *T(r)* is a monotonically increasing function for $0 \leq r \leq 1$
(preserves order from black to white).
(2) *T(r)* maps [0,1] into [0,1] (preserves the range of allowed Gray values).



- Let us denote the inverse transformation by $r = T^{-1}(s)$. We assume that the inverse transformation also satisfies the above two conditions.
- We consider the gray values in the input image and output image as random variables in the interval [0, 1].
- Let $p_{in}(r)$ and $p_{out}(s)$ denote the probability density of the Gray values in the input and output images.

- If $p_{in}(r)$ and T(r) are known, and $r = T^{-1}(s)$ satisfies condition 1, we can write (result from probability theory):

$$p_{out}(s) = \left[p_{in}(r) \frac{dr}{ds} \right]_{r=T^{-1}(s)}$$

- One way to enhance the image is to design a transformation *T(.)* such that the gray values in the output is uniformly distributed in [0, 1], i.e. $p_{out}(s) = 1, \quad 0 \leq s \leq 1$
- In terms of histograms, the output image will have all gray values in “equal proportion” .
- This technique is called **histogram equalizatio**n.

Next we derive the gray values in the output is uniformly distributed in [0, 1].

- Consider the transformation

$$s = T(r) = \int_0^r p_{in}(w)dw, \qquad 0 \le r \le 1$$

- Note that this is the cumulative distribution function (CDF) of $p_{in}(r)$ and satisfies the previous two conditions.

- From the previous equation and using the fundamental theorem of calculus,

$$\frac{ds}{dr} = p_{in}(r)$$

- Therefore, the output histogram is given by

$$p_{out}(s) = \left[p_{in}(r) \cdot \frac{1}{p_{in}(r)} \right]_{r=T^{-1}(s)} = [1]_{r=T^{-1}(s)} = 1, \qquad 0 \le s \le 1$$

- The output probability density function is uniform, regardless of the input.

- Thus, using a transformation function equal to the CDF of input gray values r, we can obtain an image with uniform gray values.

- This usually results in an enhanced image, with an increase in the dynamic range of pixel values.

How to implement histogram equalization?

Step 1:For images with discrete gray values, compute:

$$p_{in}(r_k) = \frac{n_k}{n} \qquad 0 \le r_k \le 1 \qquad 0 \le k \le L-1$$

L: Total number of gray levels

n_k: Number of pixels with gray value r_k

n: Total number of pixels in the image

Step 2: Based on CDF, compute the discrete version of the previous transformation :

$$s_k = T(r_k) = \sum_{j=0}^{k} p_{in}(r_j) \qquad 0 \le k \le L-1$$

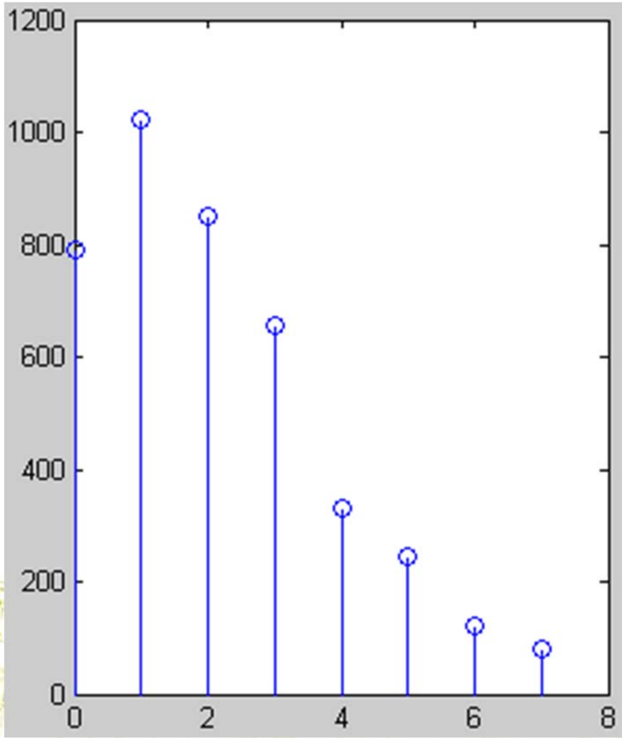
Example:

- Consider an 8-level 64 x 64 image with gray values (0, 1, …, 7). The normalized gray values are (0, 1/7, 2/7, …, 1). The normalized histogram is given below:

k	r_k	n_k	$p(r_k) = n_k/n$
0	0	790	0.19
1	1/7	1023	0.25
2	2/7	850	0.21
3	3/7	656	0.16
4	4/7	329	0.08
5	5/7	245	0.06
6	6/7	122	0.03
7	1	81	0.02

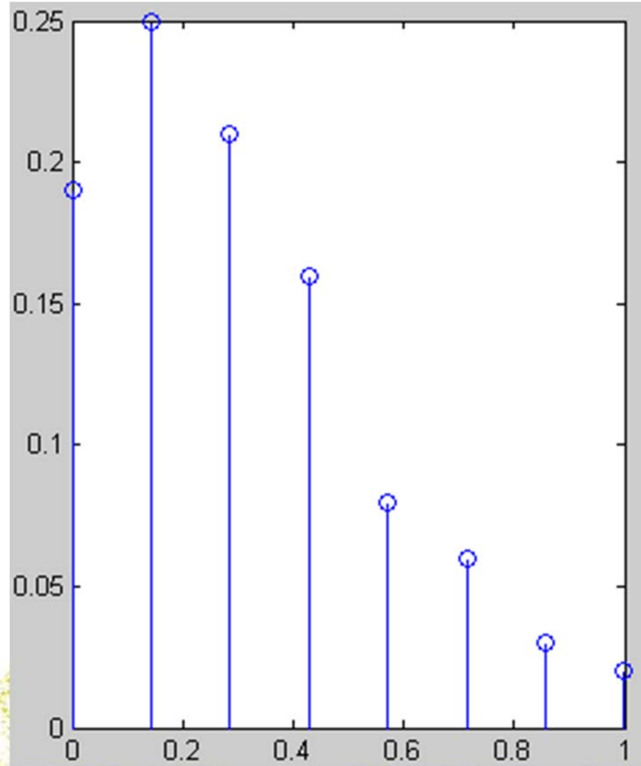
NB: The gray values in output are also (0, 1/7, 2/7, …, 1).

pixels



Gray value

Fraction of # pixels



Normalized gray value

- Applying the transformation,

$$s_k = T(r_k) = \sum_{j=0}^{k} p_{in}(r_j) \qquad \text{we have}$$

$$s_0 = T(r_0) = \sum_{j=0} p_{\text{in}}(r_j) = p_{\text{in}}(r_0) = 0.19 \rightarrow 1/7$$

$$s_1 = T(r_1) = \sum_{j=0}^{1} p_{\text{in}}(r_j) = p_{\text{in}}(r_0) + p_{\text{in}}(r_1) = 0.44 \rightarrow 3/7$$

$$s_2 = T(r_2) = \sum_{j=0}^{2} p_{\text{in}}(r_j) = p_{\text{in}}(r_0) + p_{\text{in}}(r_1) + p_{\text{in}}(r_2) = 0.65 \rightarrow 5/7$$

$$s_3 = T(r_3) = \sum_{j=0}^{3} p_{\text{in}}(r_j) = p_{\text{in}}(r_0) + p_{\text{in}}(r_1) + \cdots + p_{\text{in}}(r_3) = 0.81 \rightarrow 6/7$$

$$s_4 = T(r_4) = \sum_{j=0}^{4} p_{\text{in}}(r_j) = p_{\text{in}}(r_0) + p_{\text{in}}(r_1) + \cdots + p_{\text{in}}(r_4) = 0.89 \rightarrow 6/7$$

$$s_5 = T(r_5) = \sum_{j=0}^{5} p_{\text{in}}(r_j) = p_{\text{in}}(r_0) + p_{\text{in}}(r_1) + \cdots + p_{\text{in}}(r_5) = 0.95 \rightarrow 1$$

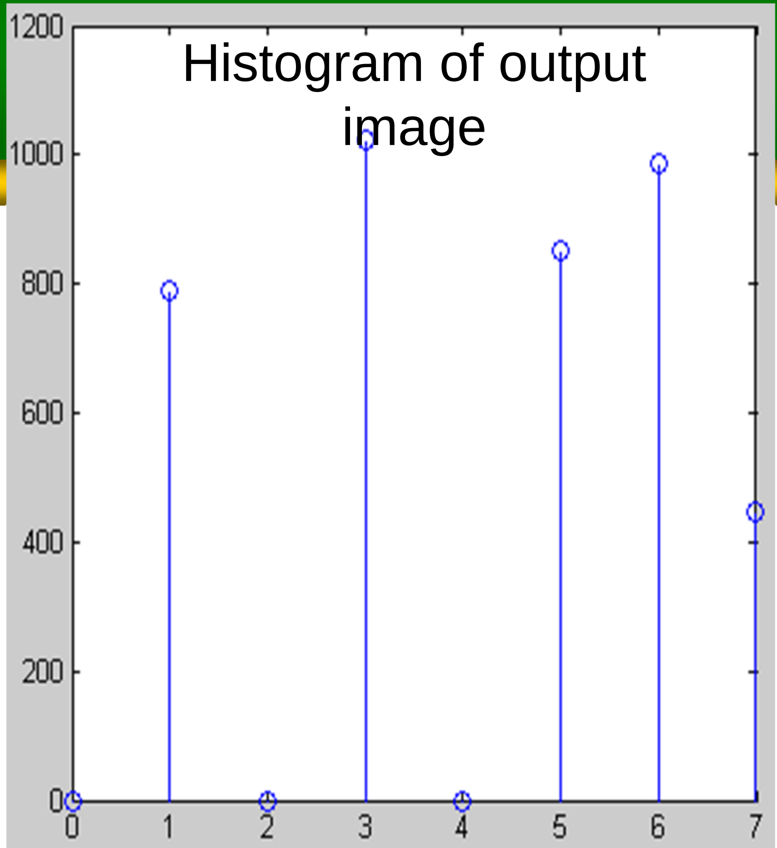
$$s_6 = T(r_6) = \sum_{j=0}^{6} p_{\text{in}}(r_j) = p_{\text{in}}(r_0) + p_{\text{in}}(r_1) + \cdots + p_{\text{in}}(r_6) = 0.98 \rightarrow 1$$

$$s_7 = T(r_7) = \sum_{j=0}^{7} p_{\text{in}}(r_j) = p_{\text{in}}(r_0) + p_{\text{in}}(r_1) + \cdots + p_{\text{in}}(r_7) = 1.00 \rightarrow 1$$

- Notice that there are only five distinct gray levels --- (1/7, 3/7, 5/7, 6/7, 1) in the output image. We will relabel them as (s_0, s_1, …, s_4).

- With this transformation, the output image will have histogram

k	s_k	n_k	$p(s_k) = n_k/n$
0	1/7	790	0.19
1	3/7	1023	0.25
2	5/7	850	0.21
3	6/7	985	0.24
4	1	448	0.11



pixels

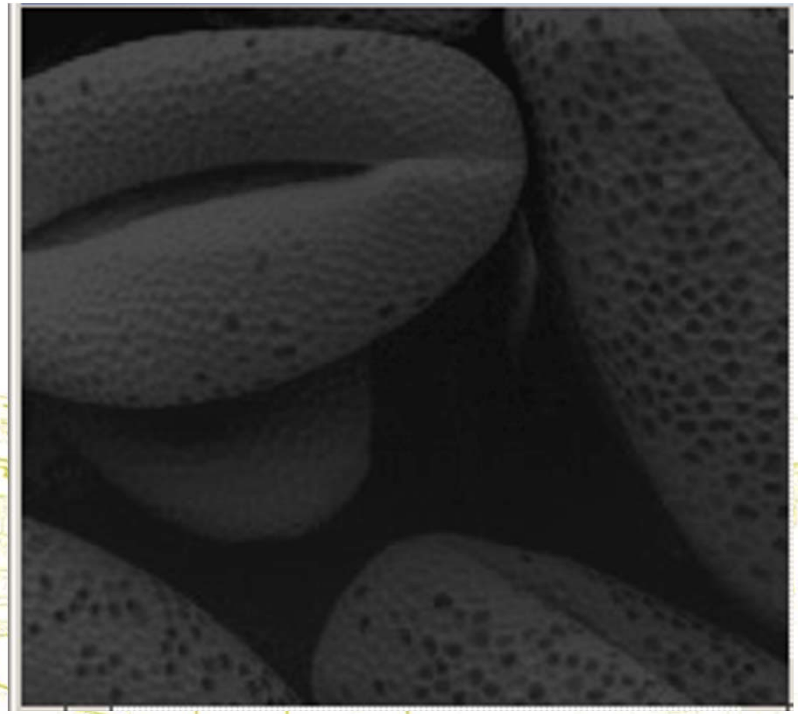
Gray values

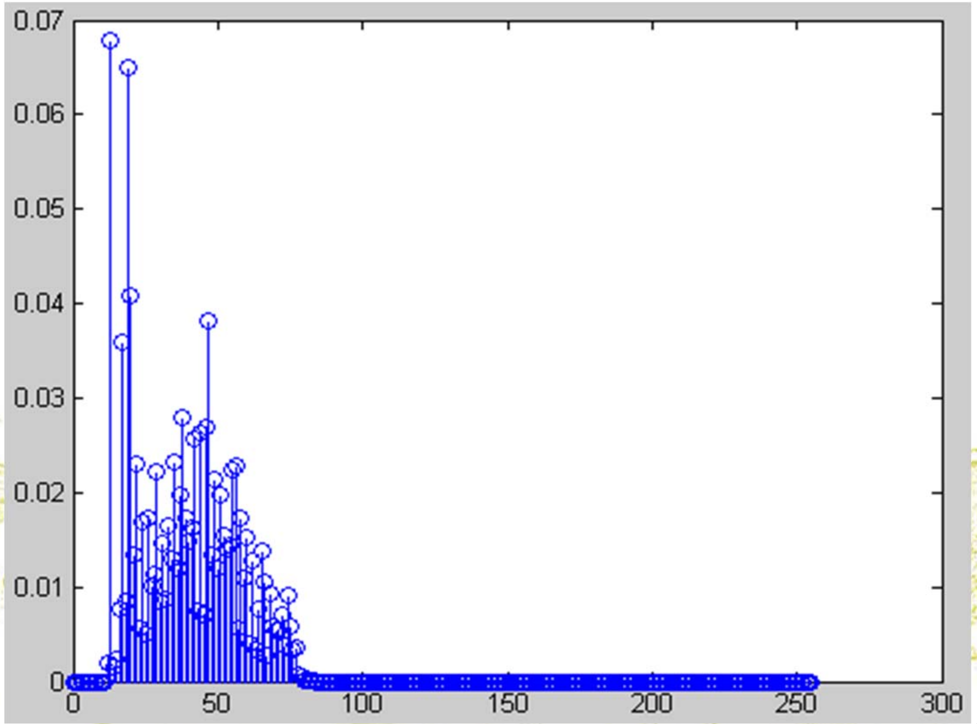
```
>> clear
>> h=[0 790 0 1023 0 850 985 448];
>> stem(0:7,h)
```

- Note that the histogram of output image is only approximately, and not exactly, uniform. This should not be surprising, since there is no result that claims uniformity in the **discrete** case.

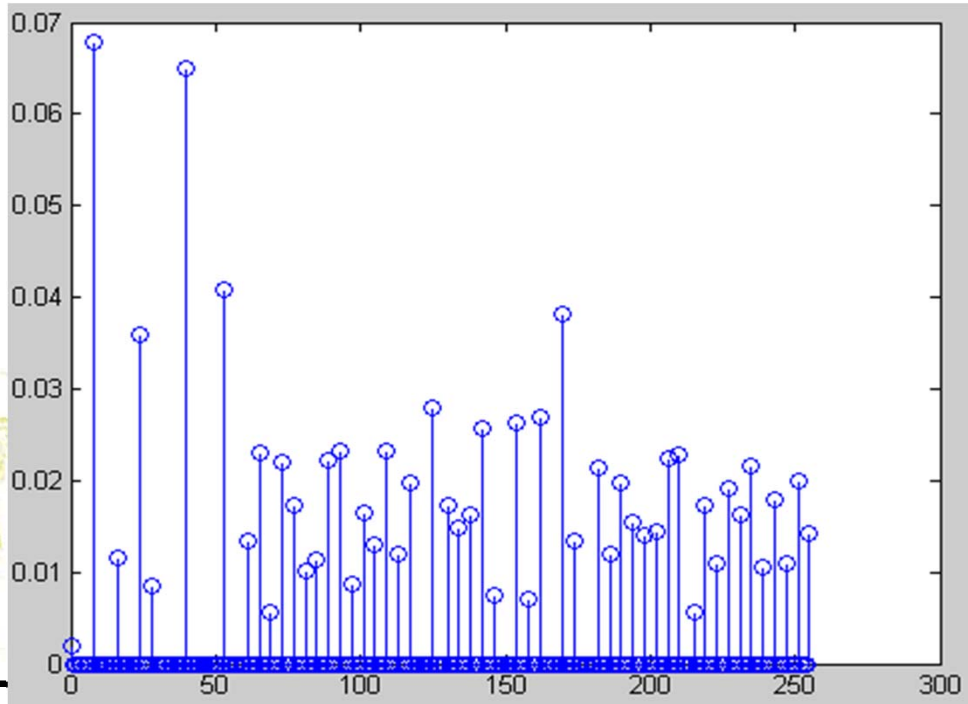
Example

Original image and its histogram



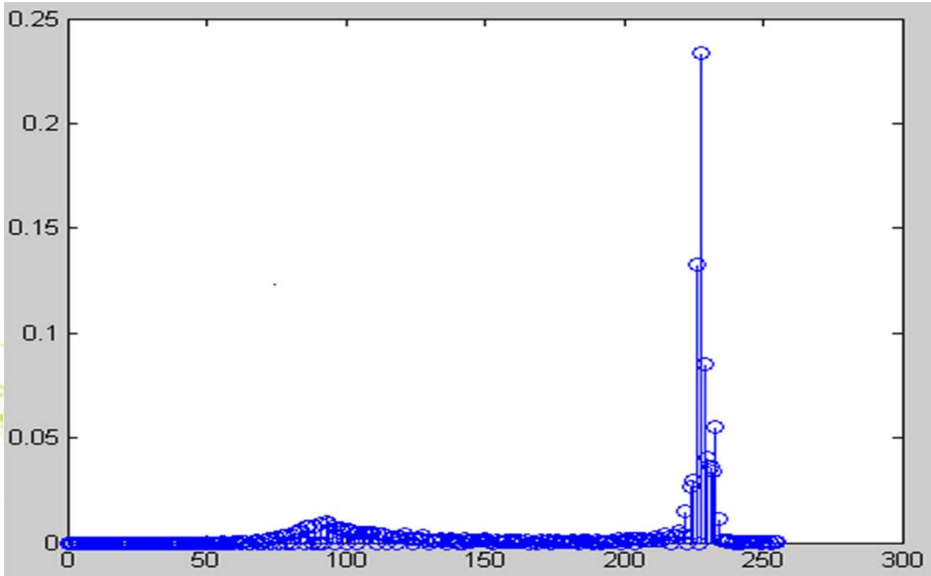


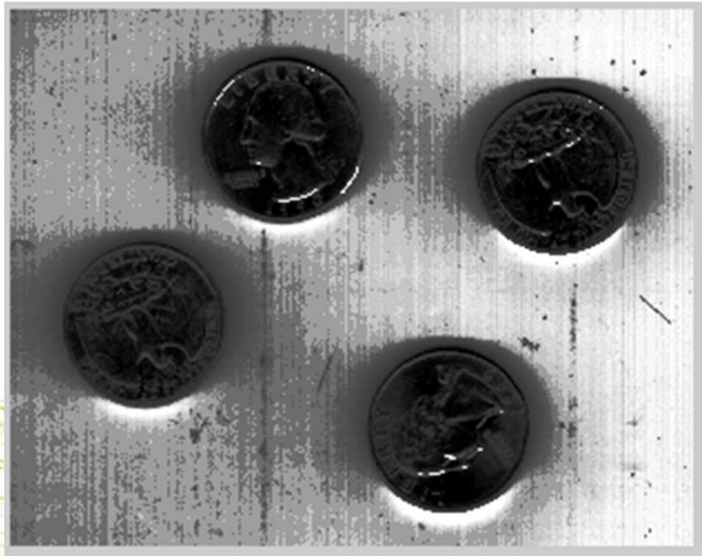
Histogram equalized image and its histogram

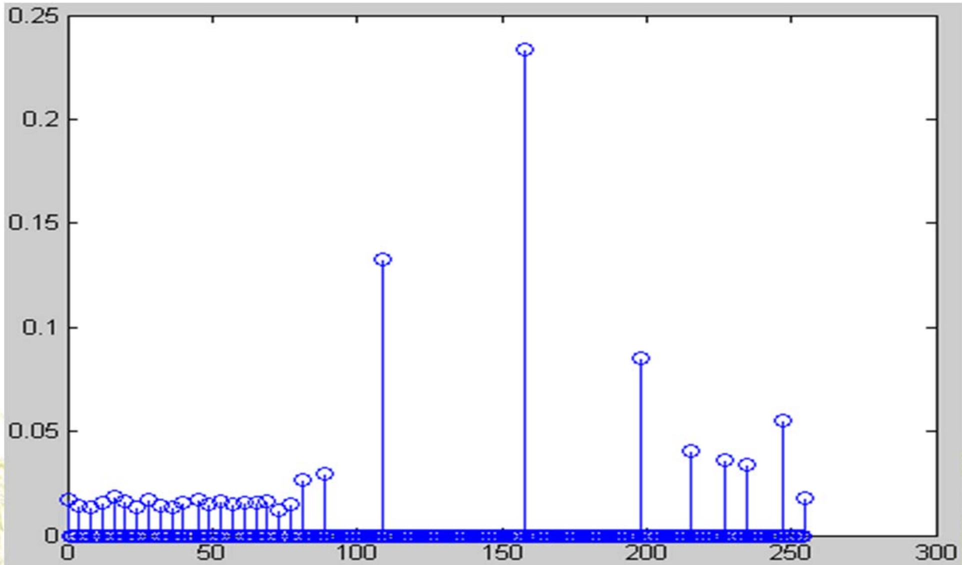


• Comments:

Histogram equalization may not always produce desirable results, particularly if the given histogram is very narrow. It can produce false edges and regions. It can also increase image "graininess" and "patchiness."






0.25
0.2
0.15
0.1
0.05
0
0
50
100
150
200
250
300

Histogram Specification (Histogram Matching)

- Histogram equalization yields an image whose pixels are (in theory) uniformly distributed among all gray levels.

- Sometimes, this may not be desirable. Instead, we may want a transformation that yields an output image with a pre-specified histogram. This technique is called **histogram specificatio**n.

- Given Information

(1) Input image from which we can compute its histogram .

(2) Desired histogram.

- Goal

Derive a point operation, $H(r)$, that maps the input image into an output image that has the user-specified histogram.

- Again, we will assume, for the moment, continuous-gray values.

Approach of derivation

$$z = H(r) \quad = G^{-1}(v = s = T(r))$$

$$\begin{tikzcd}
\boxed{\text{Input image}} \arrow[r, "s=T(r)"'] \arrow[rr, bend left] & \boxed{\text{Uniform image}} & \boxed{\text{Output image}} \arrow[l, "v=G(z)"]
\end{tikzcd}$$

• Suppose, the input image has probability density in *p(r)* . We want to find a transformation $z = H(r)$, such that the probability density of the new image obtained by this transformation is $p_{out}(z)$, which is not necessarily uniform

• First apply the transformation

$$s = T(r) = \int_0^r p_{in}(w)dw, \qquad 0 \le r \le 1 \qquad (*)$$

This gives an image with a uniform probability density.

• If the desired output image were available, then the following transformation would generate an image with uniform density:

$$V = G(z) = \int_0^z p_{out}(w)dw, \qquad 0 \le z \le 1 \qquad (**)$$

- From the gray values ν we can obtain the gray values *z* by using the inverse transformation, $z = G^{-1}(v)$

- If instead of using the gray values ν obtained from (**), we use the gray values *s* obtained from (*) above (both are uniformly distributed !), then the point transformation

$$Z = H(r) = G^{-1}[\ v = s = T(r)]$$

will generate an image with the specified density out *p(z)* , from an input image with density in *p(r)* !

$$s_k = T(r_k) = \sum_{j=0}^{k} p_{in}(r_j) \qquad 0 \le k \le L-1$$

- For discrete gray levels, we have

$$v_k = G(z_k) = \sum_{j=0}^{k} p_{out}(z_j) = s_k \qquad 0 \le k \le L-1$$

- If the transformation $z_k \rightarrow G(z_k)$ is one-to-one, the inverse transformation $s_k \rightarrow G^{-1}(s_k)$, can be easily determined, since we are dealing with a small set of discrete gray values.

- In practice, this is not usually the case (i.e.,) $z_k \rightarrow G(z_k)$ is not one-to-one) and we assign gray values to match the given histogram, as closely as possible.

Algorithm for histogram specification:

(1) Equalize input image to get an image with uniform gray values, using the discrete equation:

$$s_k = T(r_k) = \sum_{j=0}^{k} p_{in}(r_j) \qquad 0 \le k \le L-1$$

(2) Based on desired histogram to get an image with uniform gray values, using the discrete equation:

$$v_k = G(z_k) = \sum_{j=0}^{k} p_{out}(z_j) = s_k \qquad 0 \le k \le L-1$$

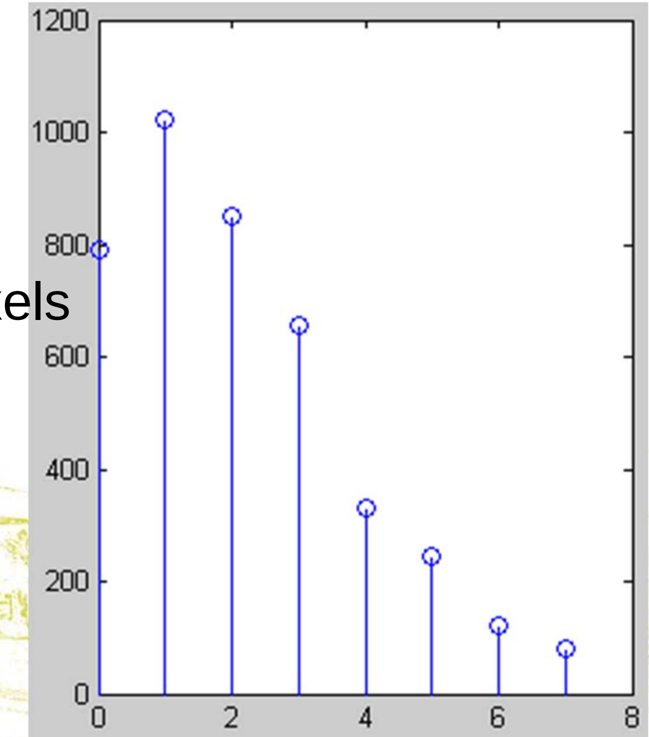
$$z = G^{-1}(v = s) \rightarrow z = G^{-1}[T(r)]$$

(3)

Example:

- Consider an 8-level 64 x 64 previous image.

k	r_k	n_k	$p(r_k) = n_k/n$
0	0	790	0.19
1	1/7	1023	0.25
2	2/7	850	0.21
3	3/7	656	0.16
4	4/7	329	0.08
5	5/7	245	0.06
6	6/7	122	0.03
7	1	81	0.02

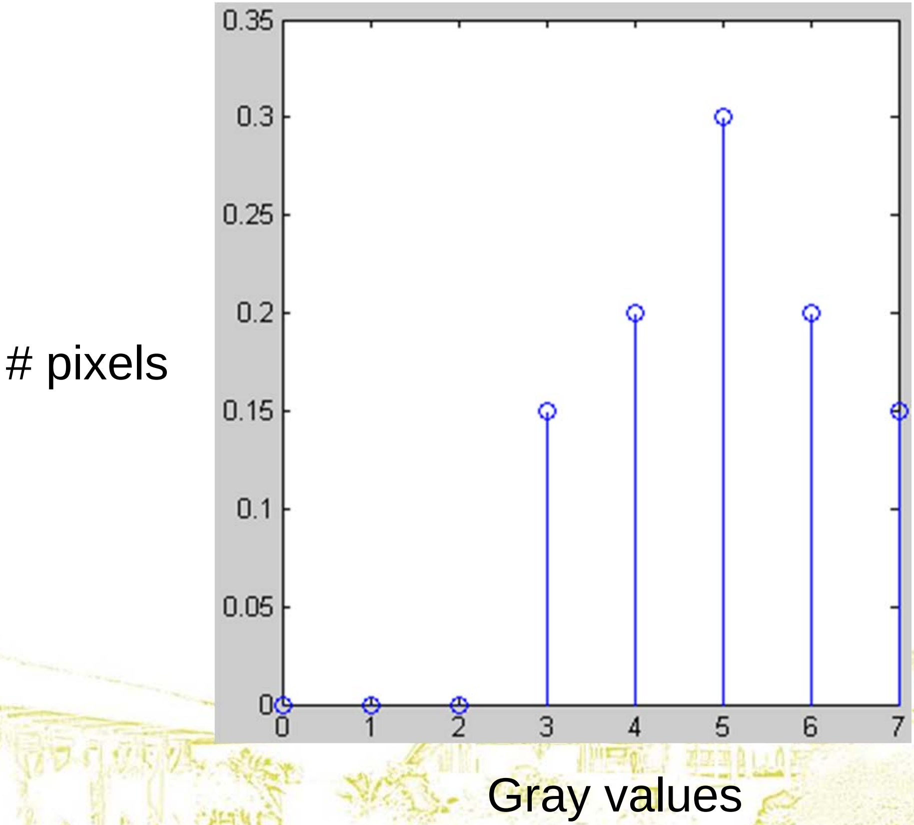


pixels

Gray value

• It is desired to transform this image into a new image, using a transformation $Z=H(r)=G^{-1}[T(r)]$, with histogram as specified below:

k	z_k	$p_{out}(z_k)$
0	0	0.00
1	1/7	0.00
2	2/7	0.00
3	3/7	0.15
4	4/7	0.20
5	5/7	0.30
6	6/7	0.20
7	1	0.15



- The transformation $T(r)$ was obtained earlier (reproduced below):

$r_j \rightarrow s_k$	n_k	$p(s_k)$
$r_0 \rightarrow s_0 = 1/7$	790	0.19
$r_1 \rightarrow s_1 = 3/7$	1023	0.25
$r_2 \rightarrow s_2 = 5/7$	850	0.21
$r_3, r_4 \rightarrow s_3 = 6/7$	985	0.24
$r_5, r_6, r_7 \rightarrow s_4 = 1$	448	0.11

- Now we compute the transformation G as before.

$$v_0 = G(z_0) = \sum_{j=0}^{0} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) = 0.00 \rightarrow 0$$

$$v_1 = G(z_1) = \sum_{j=0}^{1} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) + p_{\text{out}}(z_1) = 0.00 \rightarrow 0$$

$$v_2 = G(z_2) = \sum_{j=0}^{2} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) + p_{\text{out}}(z_1) + p_{\text{out}}(z_2) = 0.00 \rightarrow 0$$

$$v_3 = G(z_3) = \sum_{j=0}^{3} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) + p_{\text{out}}(z_1) + \cdots + p_{\text{out}}(z_3) = 0.15 \rightarrow {}^{1}\!/_{7}$$

$$v_4 = G(z_4) = \sum_{j=0}^{4} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) + p_{\text{out}}(z_1) + \cdots + p_{\text{out}}(z_4) = 0.35 \rightarrow {}^{2}\!/_{7}$$

$$v_5 = G(z_5) = \sum_{j=0}^{5} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) + p_{\text{out}}(z_1) + \cdots + p_{\text{out}}(z_5) = 0.65 \rightarrow {}^{5}\!/_{7}$$

$$v_6 = G(z_6) = \sum_{j=0}^{6} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) + p_{\text{out}}(z_1) + \cdots + p_{\text{out}}(z_6) = 0.85 \rightarrow {}^{6}\!/_{7}$$

$$v_7 = G(z_7) = \sum_{j=0}^{7} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) + p_{\text{out}}(z_1) + \cdots + p_{\text{out}}(z_7) = 1.00 \rightarrow 1$$

- Compute $z=G^{-1}(s)$, Notice that G is not invertible.

$G^{-1}(0) = ?$

$G^{-1}(1/7) = 3/7$

$G^{-1}(2/7) = 4/7$

$G^{-1}(3/7) = ?$

$G^{-1}(4/7) = ?$

$G^{-1}(5/7) = 5/7$

$G^{-1}(6/7) = 6/7$

$G^{-1}(1) = 1$

$$v_0 = G(z_0) = \sum_{j=0}^{0} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) = 0.00 \rightarrow 0$$

$$v_1 = G(z_1) = \sum_{j=0}^{1} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) + p_{\text{out}}(z_1) = 0.00 \rightarrow 0$$

$$v_2 = G(z_2) = \sum_{j=0}^{2} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) + p_{\text{out}}(z_1) + p_{\text{out}}(z_2) = 0.00 \rightarrow 0$$

$$v_3 = G(z_3) = \sum_{j=0}^{3} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) + p_{\text{out}}(z_1) + \cdots + p_{\text{out}}(z_3) = 0.15 \rightarrow 1/7$$

$$v_4 = G(z_4) = \sum_{j=0}^{4} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) + p_{\text{out}}(z_1) + \cdots + p_{\text{out}}(z_4) = 0.35 \rightarrow 2/7$$

$$v_5 = G(z_5) = \sum_{j=0}^{5} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) + p_{\text{out}}(z_1) + \cdots + p_{\text{out}}(z_5) = 0.65 \rightarrow 5/7$$

$$v_6 = G(z_6) = \sum_{j=0}^{6} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) + p_{\text{out}}(z_1) + \cdots + p_{\text{out}}(z_6) = 0.85 \rightarrow 6/7$$

$$v_7 = G(z_7) = \sum_{j=0}^{7} p_{\text{out}}(z_j) = p_{\text{out}}(z_0) + p_{\text{out}}(z_1) + \cdots + p_{\text{out}}(z_7) = 1.00 \rightarrow 1$$

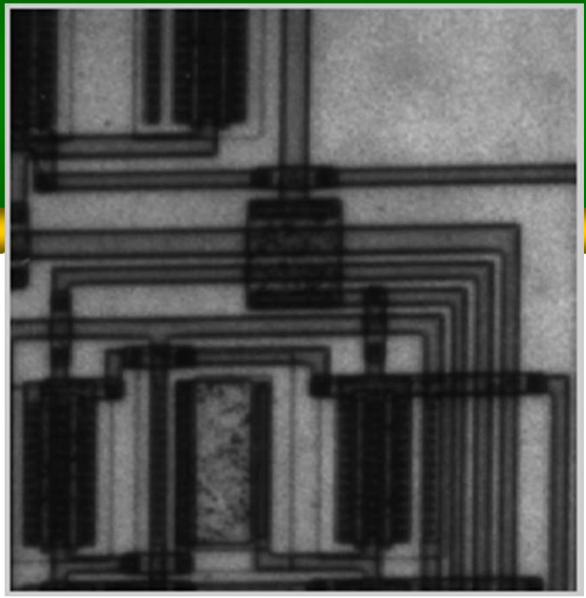
- Combining the two transformation T and G^{-1}, compute $z=H(r)=G^{-1}[v=s=T(r)]$

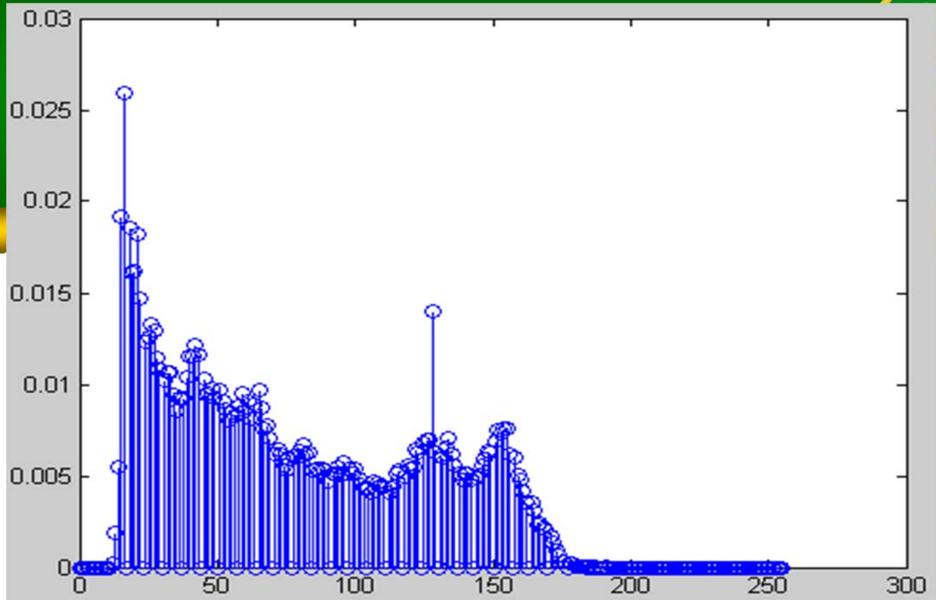
$r \rightarrow T(r)=s$	$s \rightarrow G^{-1}(s)=z$	$r \rightarrow z=H(r)=G^{-1}[T(r)]$
$r_0=0 \rightarrow 1/7$	$s_0=0 \rightarrow ?$	$r_0=0 \rightarrow G^{-1}[1/7]=3/7$
$r_1=1/7 \rightarrow 3/7$	$s_1=1/7 \rightarrow 3/7$	$r_1=1/7 \rightarrow G^{-1}[3/7]=?\ 4/7$
$r_2=2/7 \rightarrow 5/7$	$s_2=2/7 \rightarrow 4/7$	$r_2=2/7 \rightarrow G^{-1}[5/7]=5/7$
$r_3=3/7 \rightarrow 6/7$	$s_3=3/7 \rightarrow ?$	$r_3=3/7 \rightarrow G^{-1}[6/7]=6/7$
$r_4=4/7 \rightarrow 6/7$	$s_4=4/7 \rightarrow ?$	$r_4=4/7 \rightarrow G^{-1}[6/7]=6/7$
$r_5=5/7 \rightarrow 1$	$s_5=5/7 \rightarrow 5/7$	$r_5=5/7 \rightarrow G^{-1}[1]=1$
$r_6=6/7 \rightarrow 1$	$s_6=6/7 \rightarrow 6/7$	$r_6=6/7 \rightarrow G^{-1}[1]=1$
$r_7=1 \rightarrow 1$	$s_7=1 \rightarrow 1$	$r_7=1 \rightarrow G^{-1}[1]=1$

• Applying the transformation *H* to the original image yields an image with histogram as below:

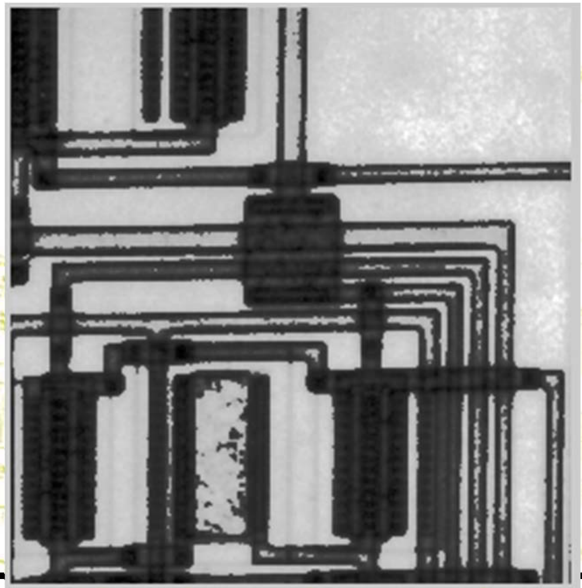
k	z_k	n_k	n_k/n (actual hist.)	$p_{out}(z_k)$ (specified hist.)
0	0	0	0.00	0.00
1	1/7	0	0.00	0.00
2	2/7	0	0.00	0.00
3	3/7	790	0.19	0.15
4	4/7	1023	0.25	0.20
5	5/7	850	0.21	0.30
6	6/7	985	0.24	0.20
7	1	448	0.11	0.15

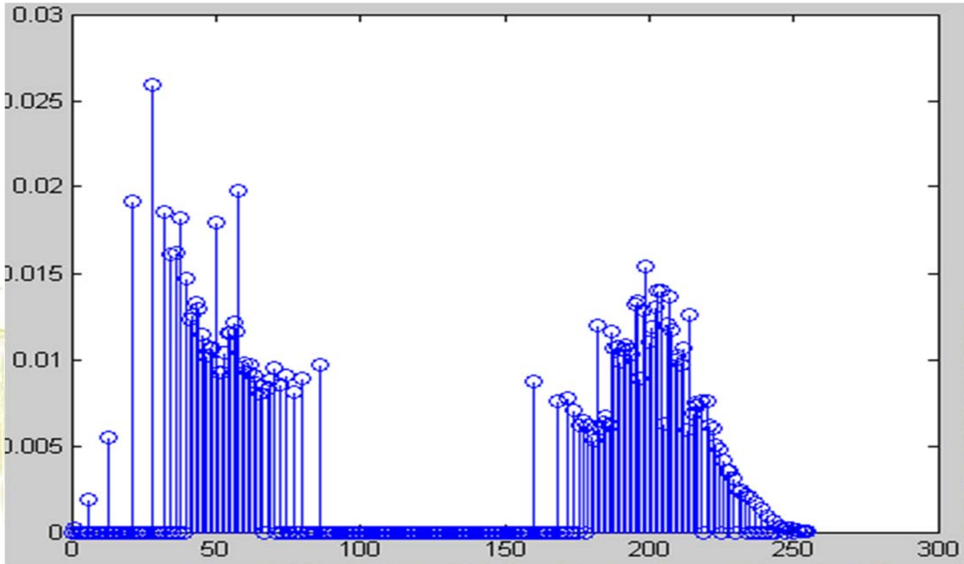
• Again, the actual histogram of the output image does not exactly but only approximately matches with the specified histogram. This is because we are dealing with discrete histograms.



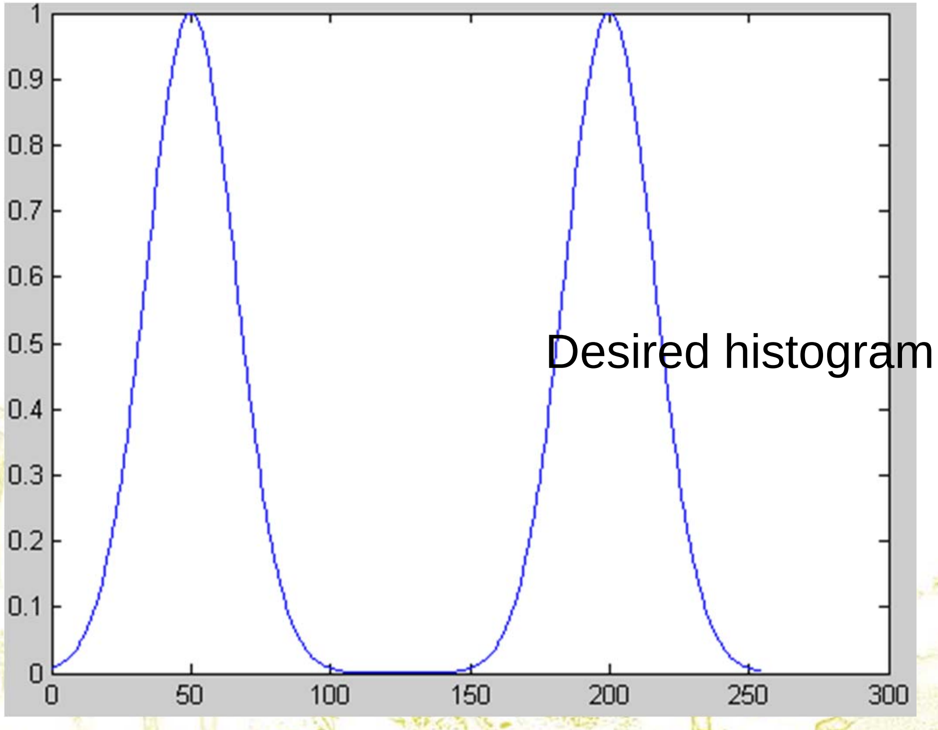


Original image and its histogram





Histogram specified image and its histogram



Desired histogram

Questions !