Introduction to Digital Image Processing

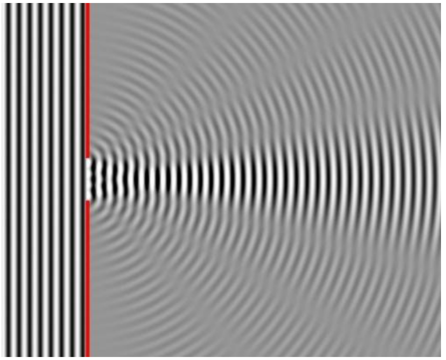
Xiaohui Yuan

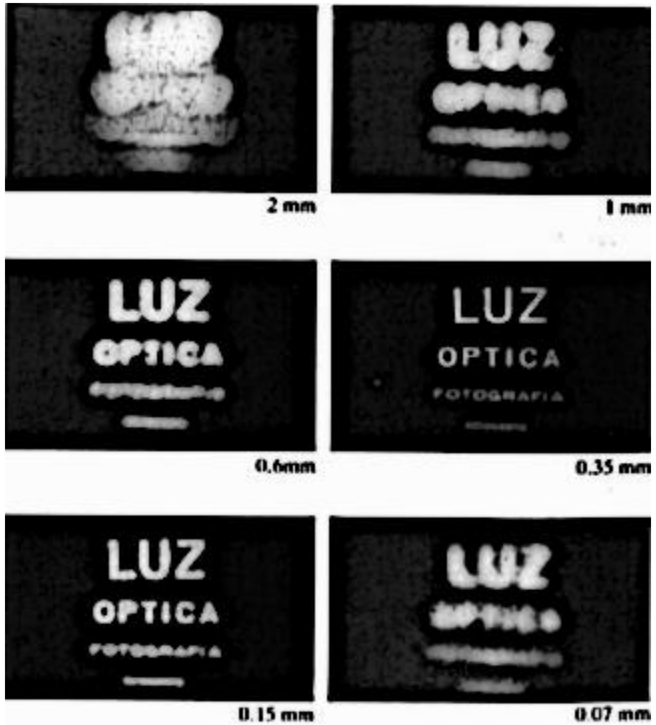
Department of Computer Science and Engineering
University of North Texas
xiaohui.yuan@unt.edu

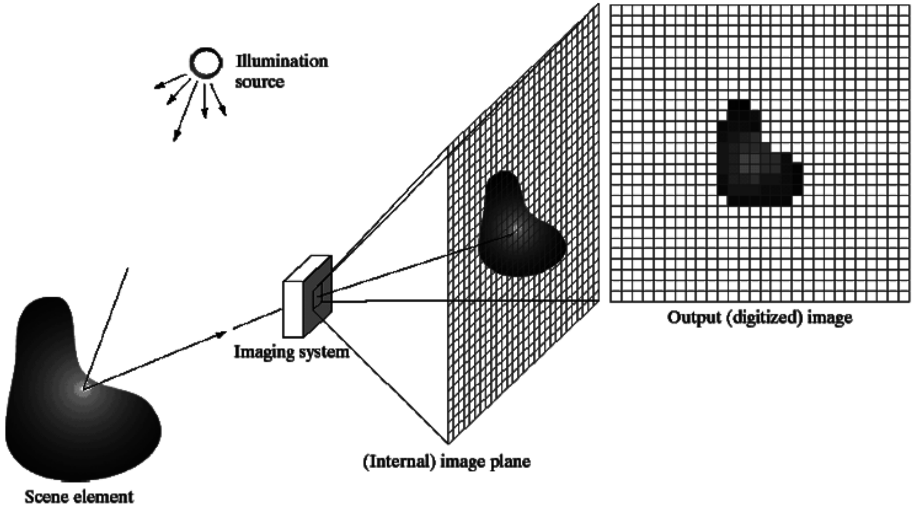
Image Acquisition

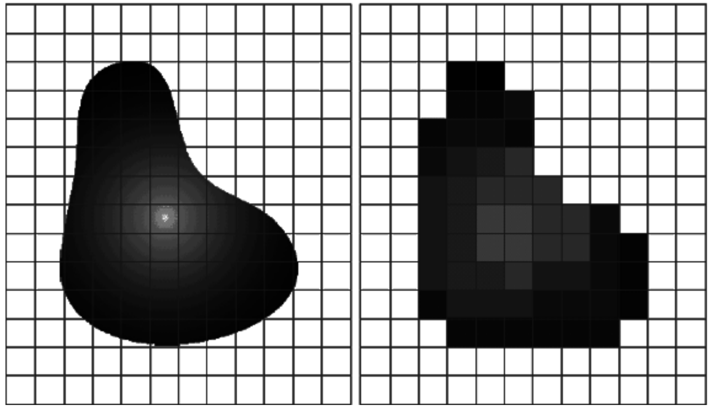
- A lens focuses light onto the film
- The thin lens model:
  - Rays passing through the center do not deviate

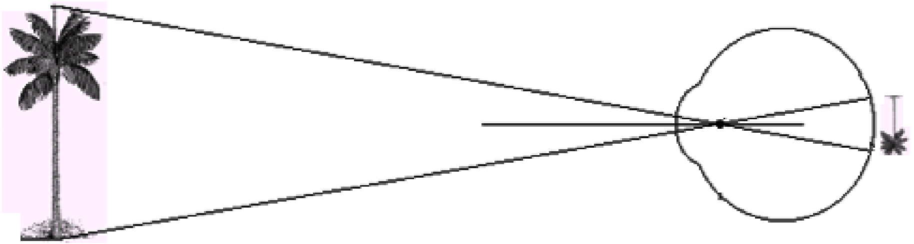
Diffraction

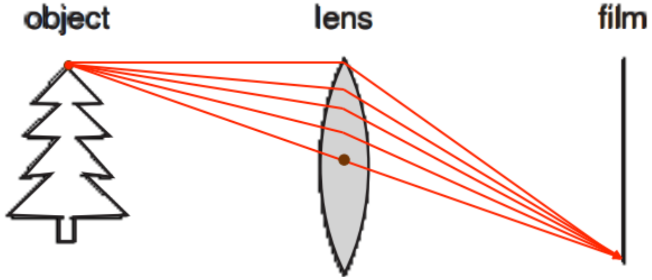


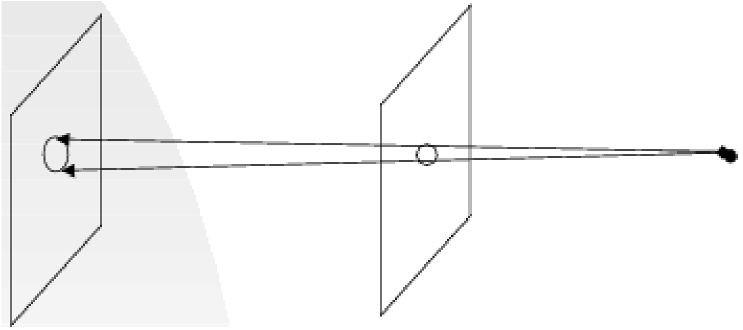










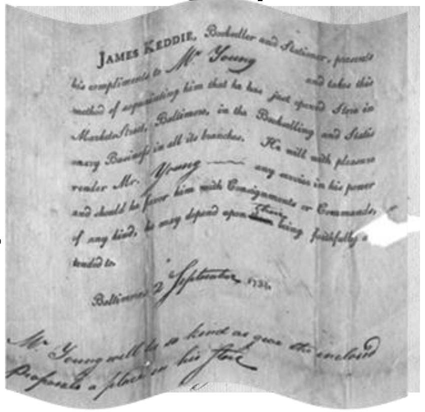


What is an Image Pixel

- Digital Images are composed of elements called **pixels**.
  - A pixel is the basic component of every digital image.
  - Each pixel consists of one (or more) numerical value that represents the color of a specific spot in the image.
  - They are organized into a grid to convey image contents.
- Each pixel is a string of binary code.
  - Each 0 or 1 is called a bit.
  - The number of bits in a pixel determines the size of the color palette, which is called color depth.
- If a pixel has more than one value, each value of all pixels forms an image plane known as a channel.
- Given an image I, a pixel is denoted as I(x, y), where x and y are the coordinates or the indices of the row and column of the pixel.

1-bit, black and white

8-bit grayscale



24-bit color

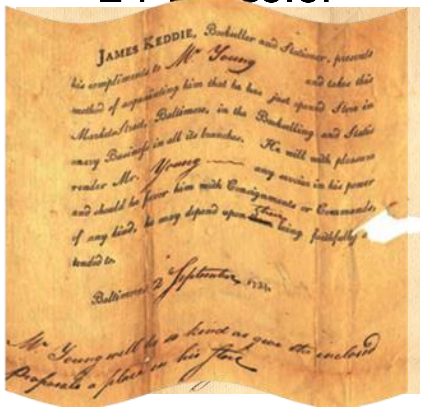
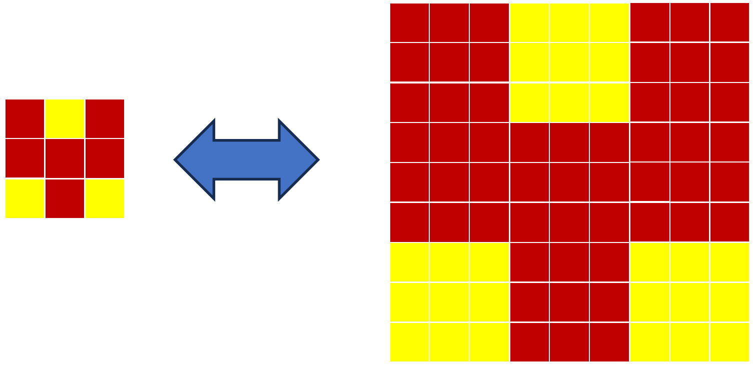


Image Resizing

- A digital image is saved as a matrix in computer memory spaces.
- Resizing an image often requires interpolation
- The interpolation process fills the voids (pixels without a value) generated from enlarging an image.
  - Zero-order interpolation (replication)
  - First-order interpolation (linear)
  - Third-order interpolation (cubic)
- Linear interpolation
  - The closer to a pixel, the higher the weight is assigned

$$I(n+a)=(1-a)\times I(n)+a\times I(n+1), \quad 0<a<1$$

- When an image is resized smaller, interpolation is also used to provide a smooth color transition between adjacent pixels.



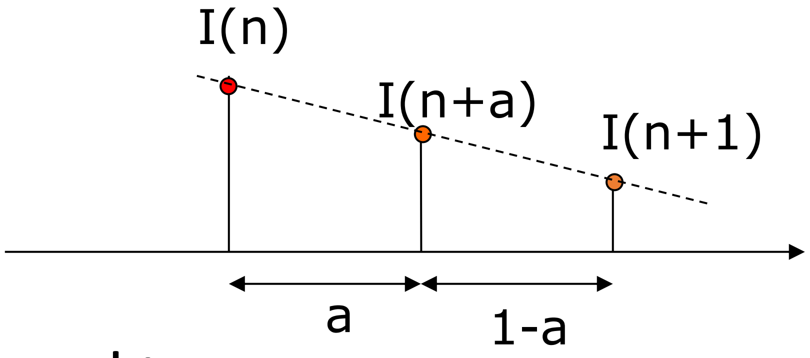
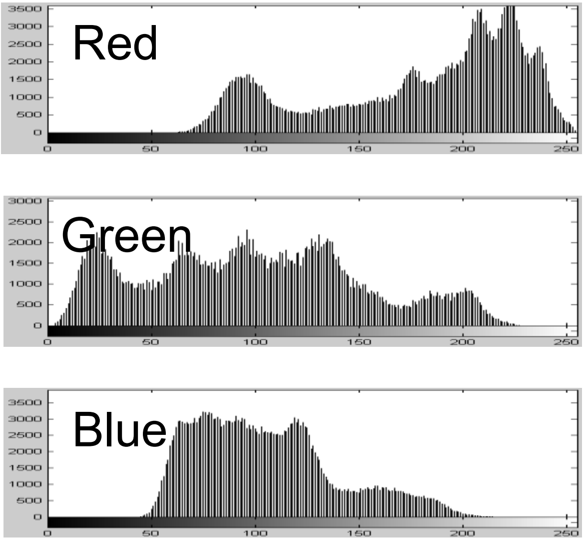


Image Histogram

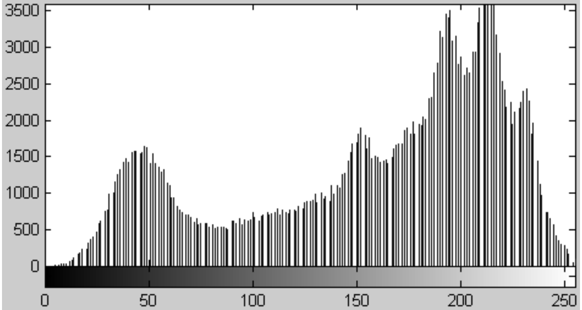
- The histogram of an image with gray levels in the range [0, L-1] is a discrete function
$$h(r_k) = n_k$$
where n_k is the count of pixels that have the gray level of r_k.
- The algorithm for creating the histogram of an image uses a single loop

Given an image A, its value range is [0 L-1].
1. Create a one-dimensional array h of size L with an initial value of zero
2. Loop through all pixels in the image A
3. For a pixel value v, increase the value at h(v) by one
4. Continue until all pixels in the image A are visited.

Histograms of a color image



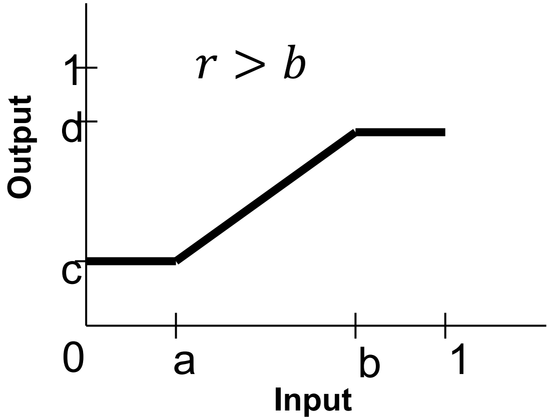
Histograms of a grayscale image

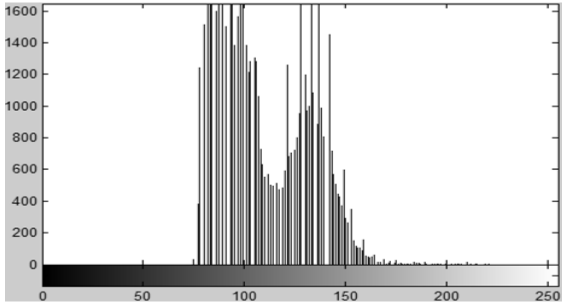


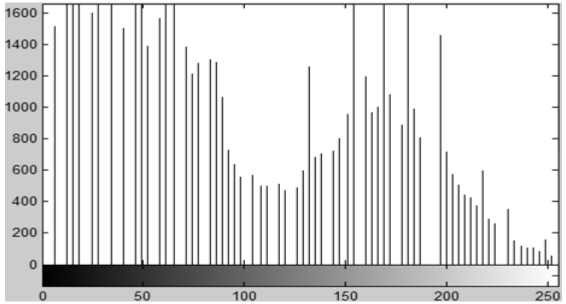
Histogram Stretching

- Image contrast is the gray level difference between neighboring pixels.
  - A larger gray level difference of an image often results in a greater contrast.
- The histogram stretching maps the value of all pixels to a new value that spans the full grayscale range.
- The mapping function can be a linear function or a nonlinear function.

$$s = \begin{cases} c & r < a \\ (r-a)\left(\dfrac{d-c}{b-a}\right)^{\gamma} + c & a \le r \le b \\ d & r > b \end{cases}$$







Histogram Equalization

- The goal is to maximize the usage of the full brightness range
  - This usually results in an enhanced image, with an increase in the dynamic range of pixel values.
  - The maximum contrast is achieved when the image histogram follows a uniform distribution.
- Assume continuous gray level r of an image and r is in the range of [0, 1].
- The enhanced image has a gray level s through a transformation T(·):

$$s = T(r), \qquad 0 < r < 1$$

- The transformation function satisfies the following two criteria:
  - T(·) is monotonically increasing in the range of [0, 1];
  - $0 < T(r) < 1$ and $0 < r < 1$

Histogram Equalization – Continuous Case

- Given two random variables r and s, their probability density functions are $p_r(r)$ and $p_s(s)$, respectively.
  - We want to compute the transformation function $T(r)$ that maps $p_r(r)$ to $p_s(s)$
- If $p_s(s)$ and $T(r)$ are known, $p_s(s)$ is computed as follows:

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

- Since we know $p_s(s) = 1$, we have

$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right| = 1$$

- Multiply both sides with ds ($ds > 0$), the above equation becomes

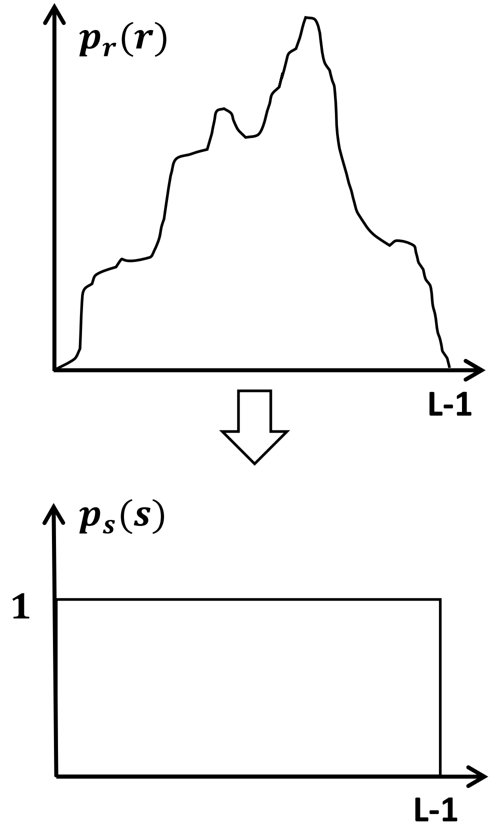
$$p_s(s)ds = p_r(r)dr = ds$$

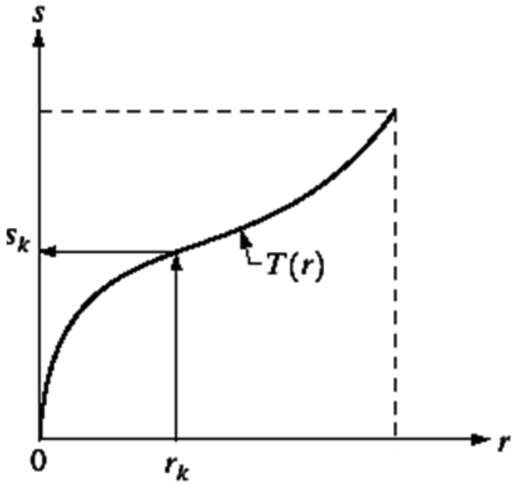
- Since $s = T(r)$, we have

$$dT(r) = p_r(r)dr$$

- Take integration on both sides and we get

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





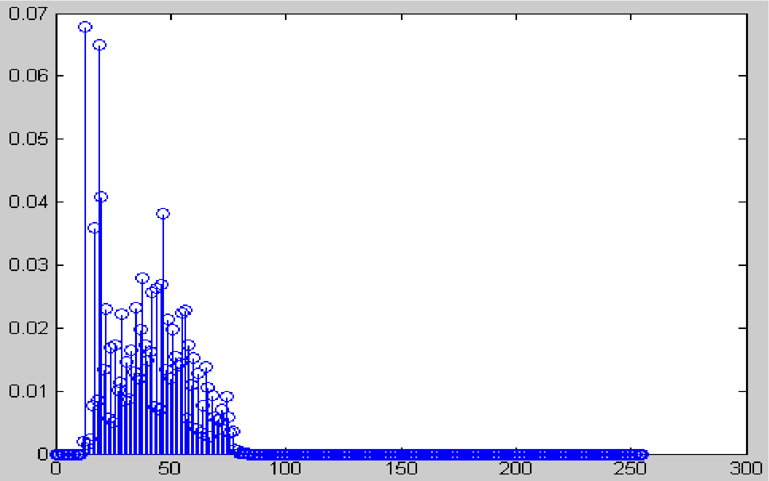
Histogram Equalization – Discrete Case

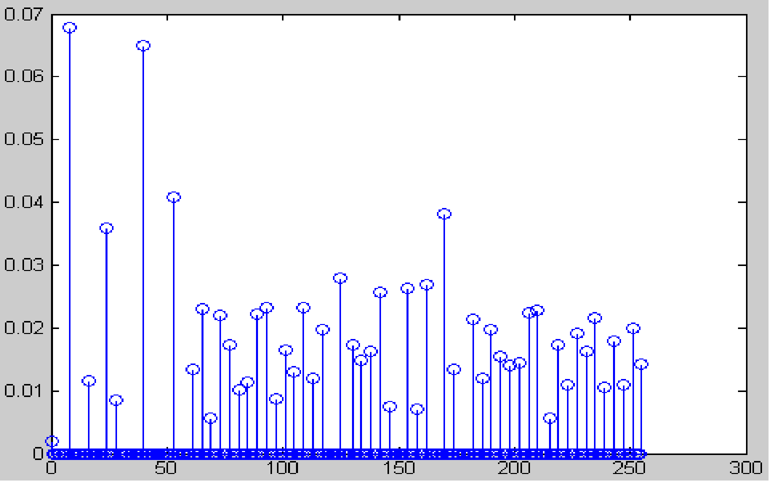
- In the discrete case, a uniform distribution in the output can hardly be achieved.
- Approximation is usually used:

$$p_r(r_k) = \frac{n_k}{n} \qquad k = 0,1,2,\ldots,L-1$$

- The discrete version of the transformation function is computed as follows:

$$s_k = T(r_k) = \sum_{j=0}^{k} p_r(r_j) = \sum_{j=0}^{k} \frac{n_j}{n}$$



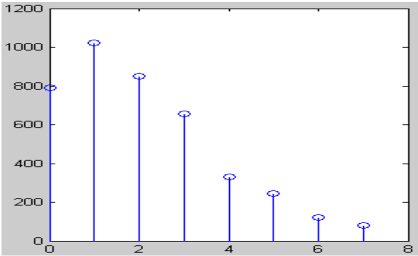


A Numerical Example

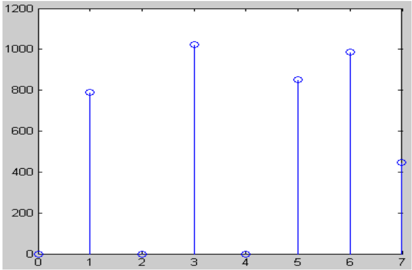
- Consider an 8-level 64 x 64 image with gray values (0, 1, …, 7). The normalized gray values are (0, 1/7, 2/7, …, 1).

Following the transformation: $s = \sum_{j=0}^{k} \frac{n_j}{n}$

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



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



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

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

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

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

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

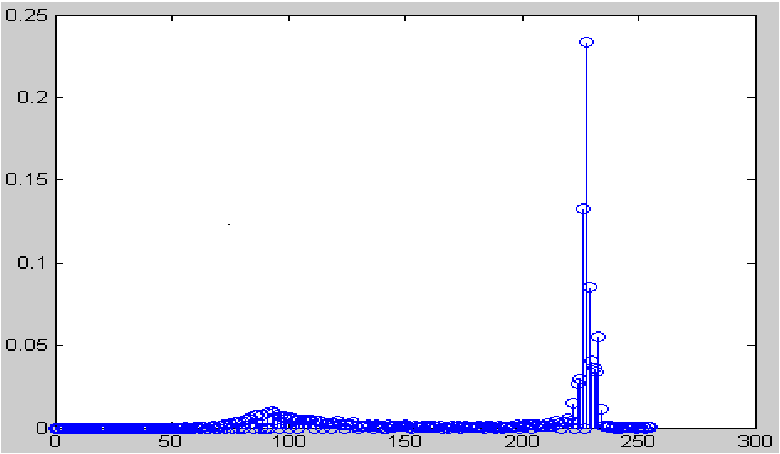
$$s_5 = T(r_5) = \sum_{j=0}^{5} p_{\text{in}}(r_j) = p_{\text{in}}(r_0) + p_{\text{in}}(r_1) + \cdots + p_{\text{in}}(r_5) = 0.95 \rightarrow 1$$

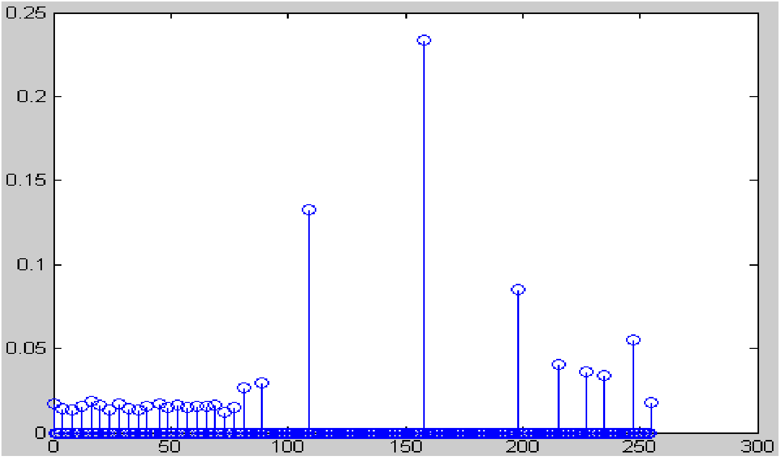
$$s_6 = T(r_6) = \sum_{j=0}^{6} p_{\text{in}}(r_j) = p_{\text{in}}(r_0) + p_{\text{in}}(r_1) + \cdots + p_{\text{in}}(r_6) = 0.98 \rightarrow 1$$

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

Issues of Histogram Equalization

- Histogram equalization offers advantages of full automation and makes full range of color.
- It may not always produce desirable results
  - when image histogram has long tails.
- It can produce false edges and regions. It can also increase image “graininess.”





Histogram Matching

- Histogram matching is used to specify the shape of the histogram that we wish an image to have.
  - We can specify the target histogram by giving a sample image.
- In the continuous case, if we map the histograms of two images (input and target) into the uniform distribution following the histogram equalization, we can have two transformation functions T and G for input and target images, respectively:

$$s = T(r) = \int_0^r p_{in}(w)dw \quad \text{and} \quad v = G(z) = \int_0^z p_{tg}(w)dw$$

- We apply the inverse transformation of the target image, G^{-1}, to the input image, such that the histogram of the input image matches the histogram of the target image:

$$\mathrm{z} = H(r) = G^{-1}(T(r))$$

Histogram Matching – Discrete Case

- Obtain the histogram of the input image, $p(r_k)$.
- Map gray level r_k into s_k following the histogram equalization scheme

$$s_k = T(r_k) = \sum_{j=0}^{k} p_r(r_j) = \sum_{j=0}^{k} \frac{n_j}{n}$$

- Obtain the transformation function G from the target image using

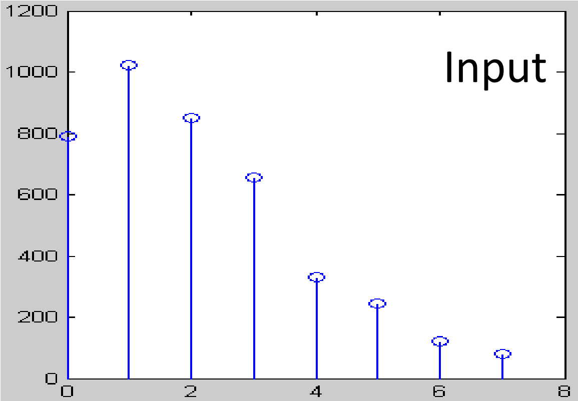
$$v_k = G(z_k) = \sum_{j=0}^{k} p_z(z_j) = s_k$$

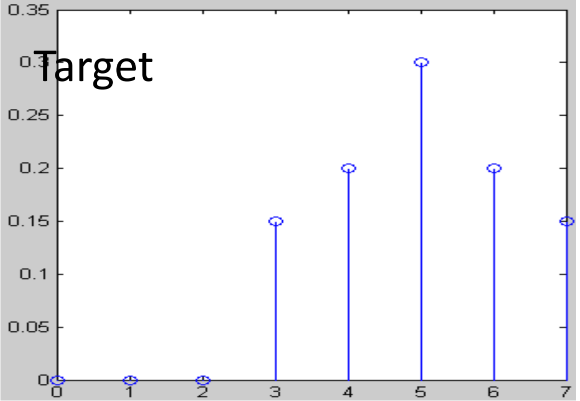
- Pre-compute z_k for each value of s_k using the iterative scheme defined in connection with

$$(G(\hat{z}) - s_k) \geq 0$$

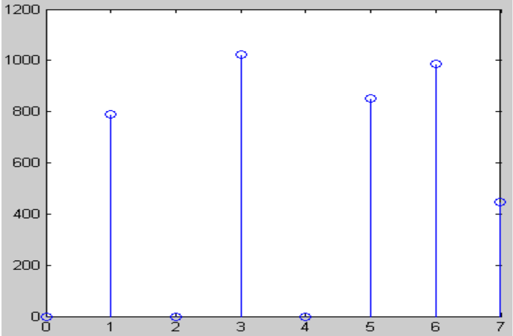
- For each pixel in the input image, if the value of that pixel is r_k, map it to the corresponding level s_k; then map s_k into the final level z_k.

A Numerical Example





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



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

k	z_k	$p_{\text{out}}(z_k)$
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

Computing Mapping Function G

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Gray Level Lookup Table

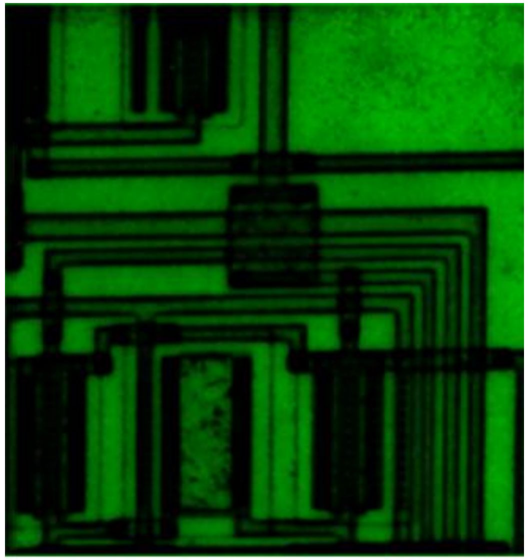
Inverse function

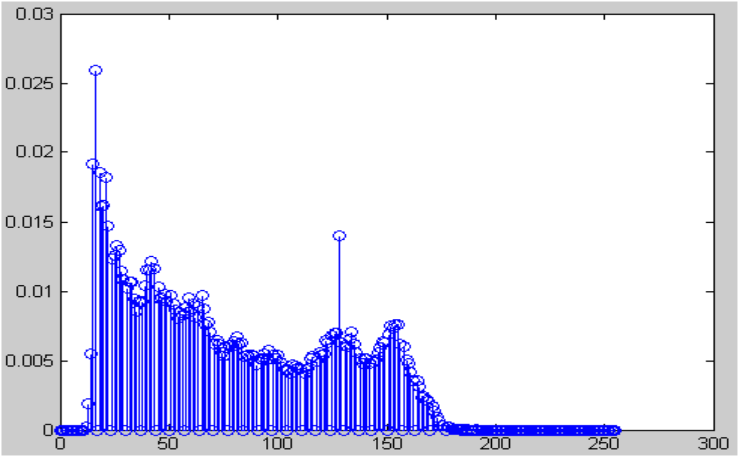
$G^{-1}(0)$	?
$G^{-1}(\frac{1}{7})$	$\frac{3}{7}$
$G^{-1}(\frac{2}{7})$	$\frac{4}{7}$
$G^{-1}(\frac{3}{7})$	?
$G^{-1}(\frac{4}{7})$	?
$G^{-1}(\frac{5}{7})$	$\frac{5}{7}$
$G^{-1}(\frac{6}{7})$	$\frac{6}{7}$
$G^{-1}(1)$	1

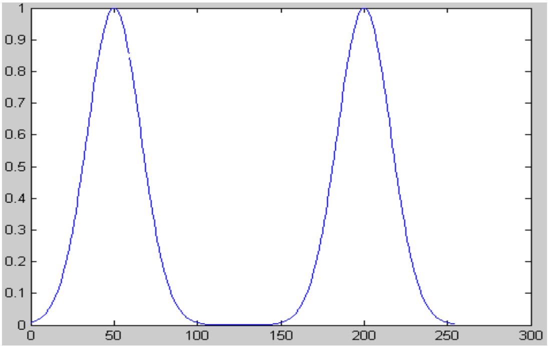
$r \to s$	
0	1/7
1/7	3/7
2/7	5/7
3/7	6/7
4/7	6/7
5/7	1
6/7	1
1	1

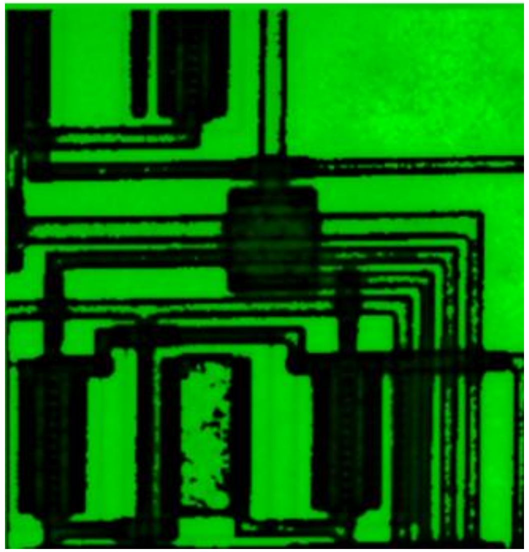
$s \to z$	
0	?
1/7	3/7
2/7	4/7
3/7	?
4/7	?
5/7	5/7
6/7	6/7
1	1

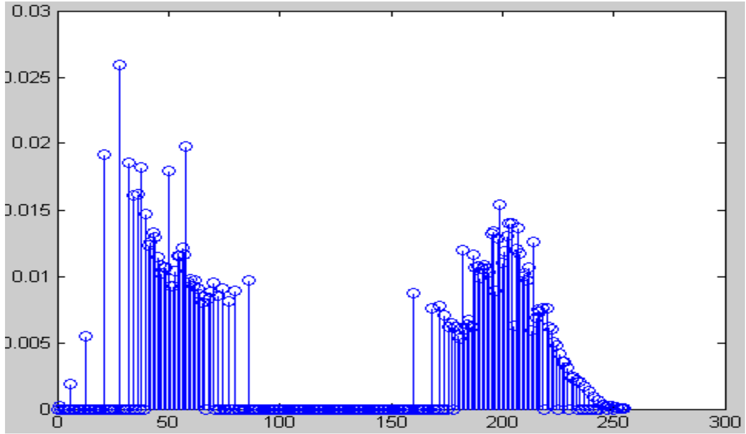
$r \to z$	
0	3/7
1/7	4/7
2/7	5/7
3/7	6/7
4/7	6/7
5/7	1
6/7	1
1	1




0.03
0.025
0.02
0.015
0.01
0.005
0
0 50 100 150 200 250 300


1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0 50 100 150 200 250 300




0.03
.025
0.02
.015
0.01
.005
0
0 50 100 150 200 250 300