Image Enhancement

Dr. Ratnakar Dash

CSE Dept. of NIT Rourkela

Sometimes Image is not perfect!!

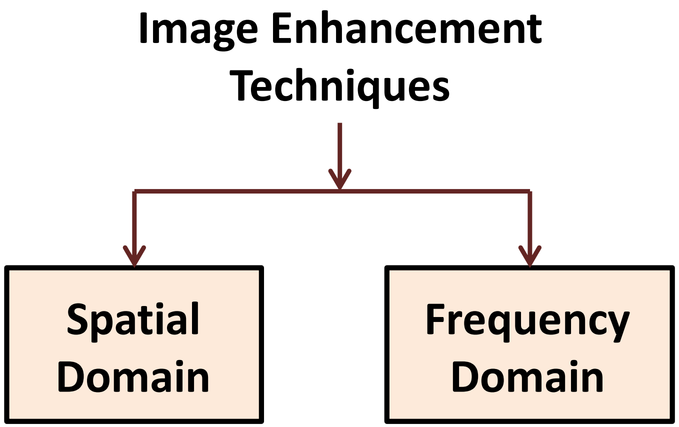
What is Image Enhancement

- Processing an image to enhance certain features of the image

- Result become more suitable than the original image for specific application

- Processing techniques are highly application dependent
  - e.g Best technique for the enhancement of X-ray images may not be the best one for microscopic images

Different Enhancement Techniques



- Spatial Domain Techniques
  - Work on image plane itself
  - Direct manipulation of the image pixels
- Frequency Domain Techniques
  - Modify the Fourier Transform coefficients of an image
  - Take inverse Fourier Transform of the modified coefficients to obtain the enhanced image

What is a good image?

- For Human Visual
  - The visual evaluation of image quality is a highly subjective process
  - It is hard to standardize the definition of a good image
- For Machine Perception
  - The evaluation task is easier
  - The good image is one which gives the best machine recognition result
- Before choosing a particular image enhancement method for certain applications, some amount of trial & error is required

Spatial Domain Enhancement Techniques

- These techniques are being operated directly on image pixels
- Mathematically,

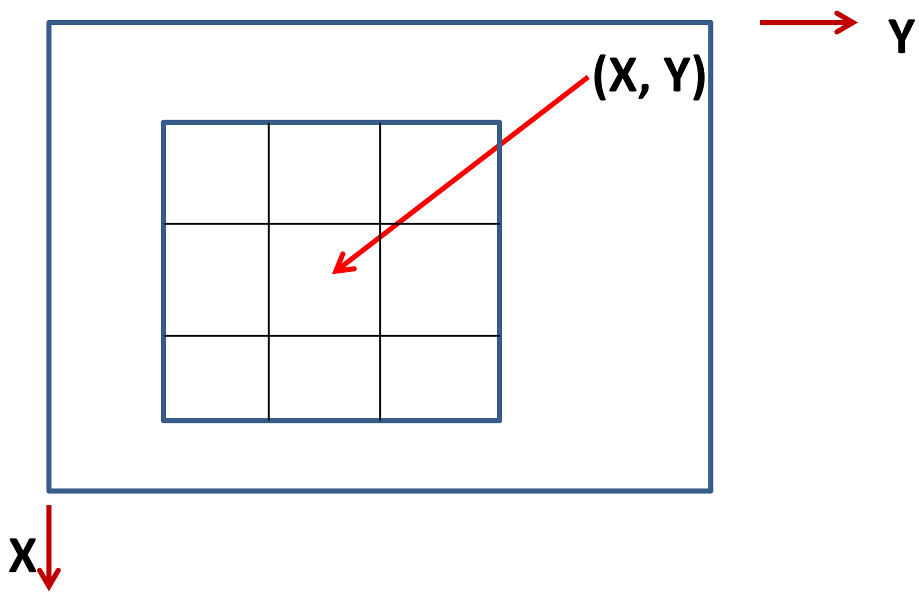
$$g(x, y) = T[f(x, y)]$$

  - Where, g (x, y) = Enhanced image
    f (x, y) = Original image
    T = Transformation function defined over the *neighbourhood* of a pixel

- Again can be broadly categorized into following types
  - Point Processing Techniques
  - Histogram Based Techniques
  - Mask Processing Techniques

Neighbourhood

- 3×3 Neighbourhood of a pixel (X, Y)



Point Processing Techniques

- Neighborhood = 1x1 pixel
- Point operations are zero-memory operations
- g depends on only the value of f at (x, y)
- T = gray level (or intensity or mapping) transformation function

$$s = T(r)$$

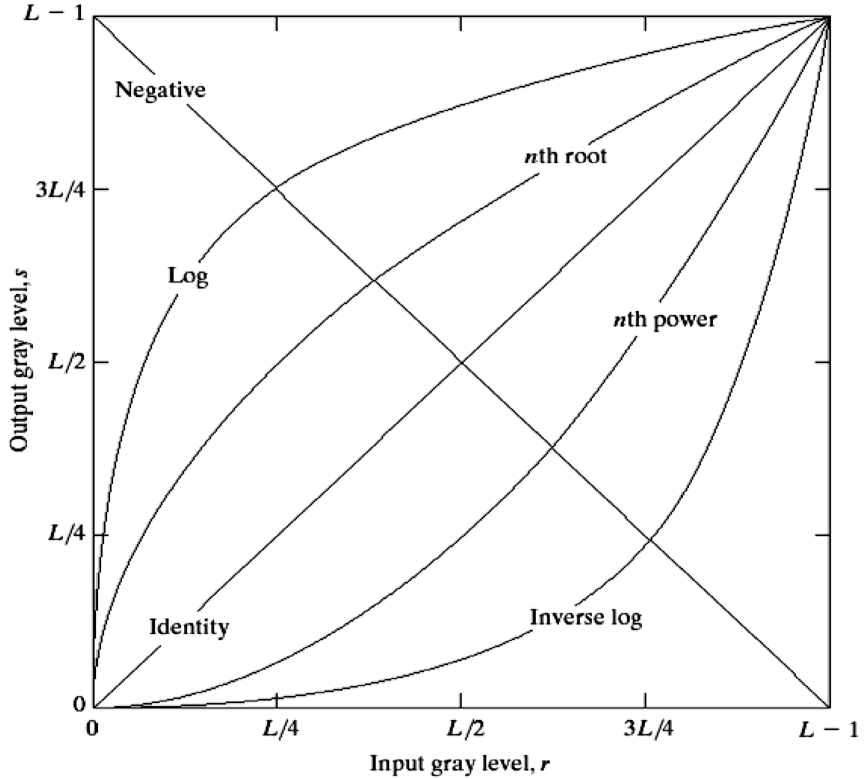
Where

r = gray level of $f(x, y)$

s = gray level of $g(x, y)$

Point Processing Techniques(cont…)

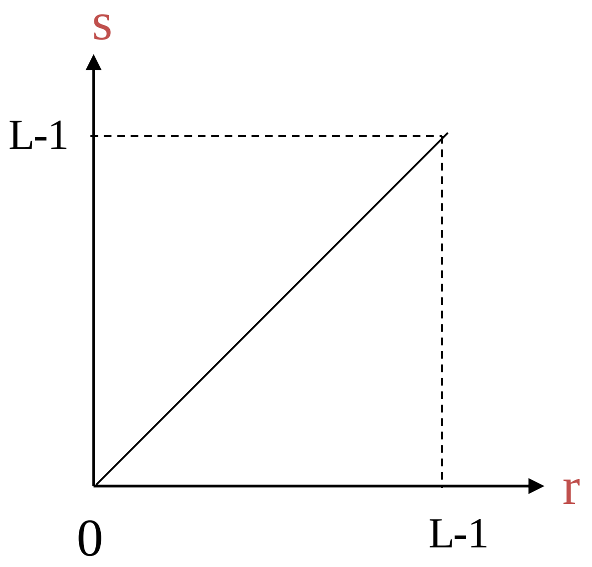
- 3-basic Gray level Transformation Functions
  - Linear Functions
    - Identity transformation
    - Negative transformation
  - Logarithm function
    - Log transformation
    - Inverse-log transformation
  - Power-law function
    - n^{th} power transformation
    - n^{th} root transformation



Point Processing Techniques(cont…)

- Identity Function (or Lazy man operation)
  - An original image with gray level in the range ***[0, L-1]***
  - Identity transformation :

$$s = T(r) = r$$

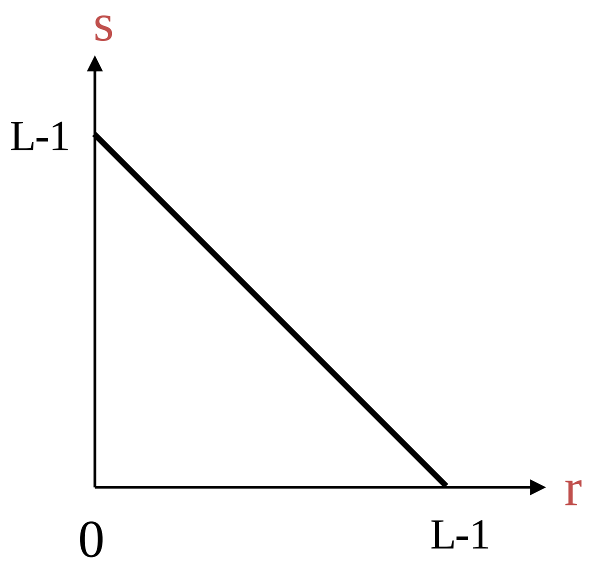


  - Absolutely no effect on the visual quality of the image.

Point Processing Techniques(cont…)

- Image Negative
  - An original image with gray level in the range ***[0, L-1]***
  - Negative transformation :

$$s = T(r) = L - 1 - r$$



  - Reversing the intensity levels of an image.
  - Suitable for enhancing white or gray detail embedded in dark regions of an image, especially when the black area dominant in size

Point Processing Techniques(cont…)

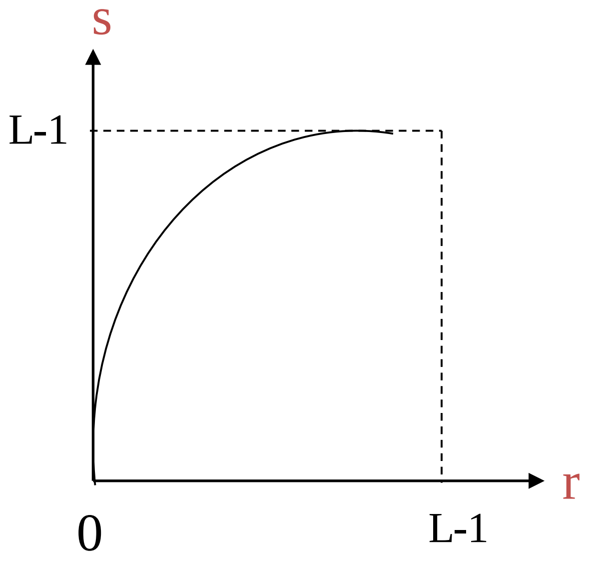
- Example of Image Negative

Point Processing Techniques(cont...)

- Log Transformation
  - An original image with gray level in the range ***[0, L-1]***
  - Log transformation :

$$s = c \log (1+r)$$

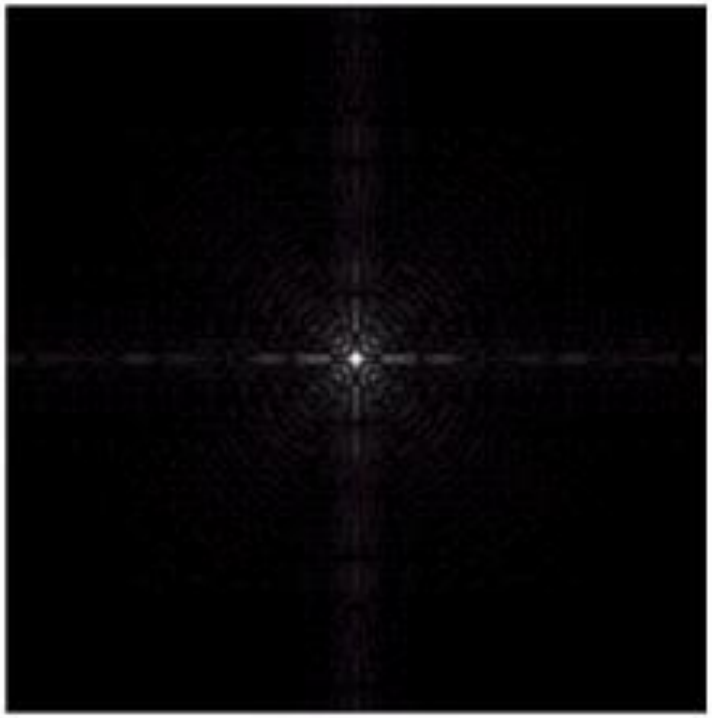
$$\textit{Where } c = \textit{const. and} \quad r \geq 0$$



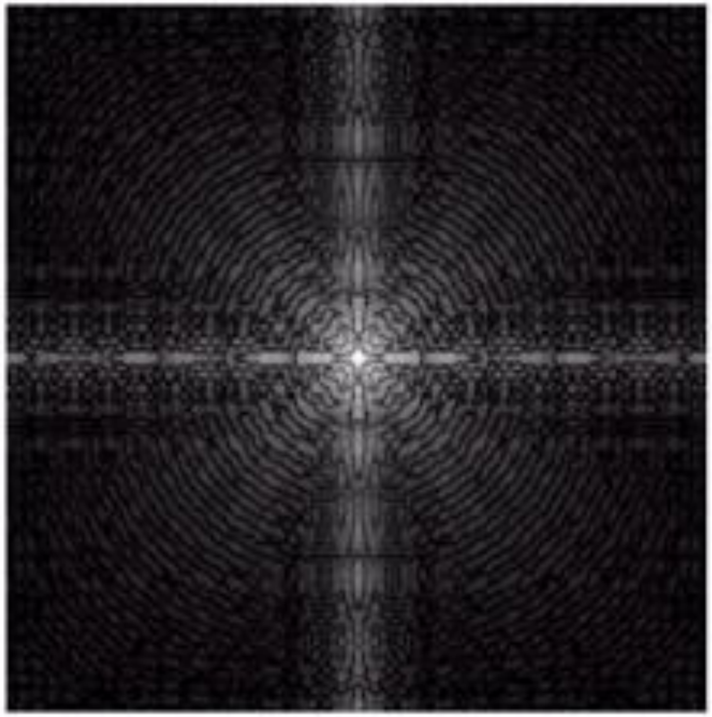
  - Log curve maps a narrow range of low gray-level values in the input image into a wider range of output levels .
  - Suitable for enhancing white or gray detail embedded in dark regions of an image, especially when the black area dominant in size

Point Processing Techniques(cont…)

- Example of Log Transformation



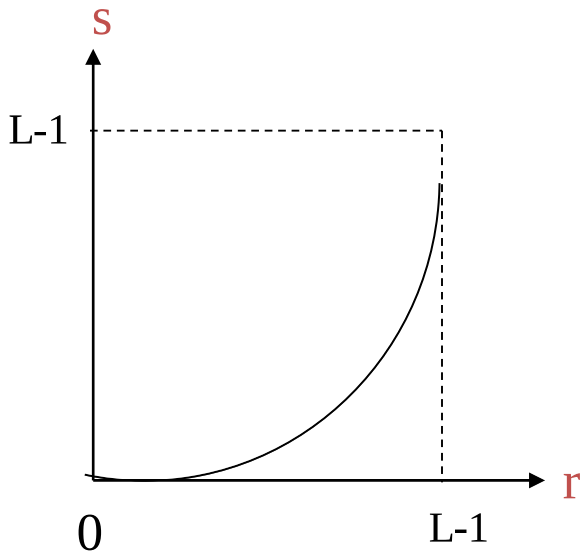
Fourier Spectrum with range = 0 to 1.5 x 10^6



Result after apply the log transformation with c = 1, range = 0 to 6.2

Point Processing Techniques(cont…)

- Inverse-Log Transformation
  - An original image with gray level in the range ***[0, L-1]***
  - Opposite of the log transform
  - Used to expand the values of high pixels in an image while compressing the darker-level values.
  - Inverse-Log curve maps a wider range of low gray-level values in the input image into a narrow range of output levels .



Point Processing Techniques(cont…)

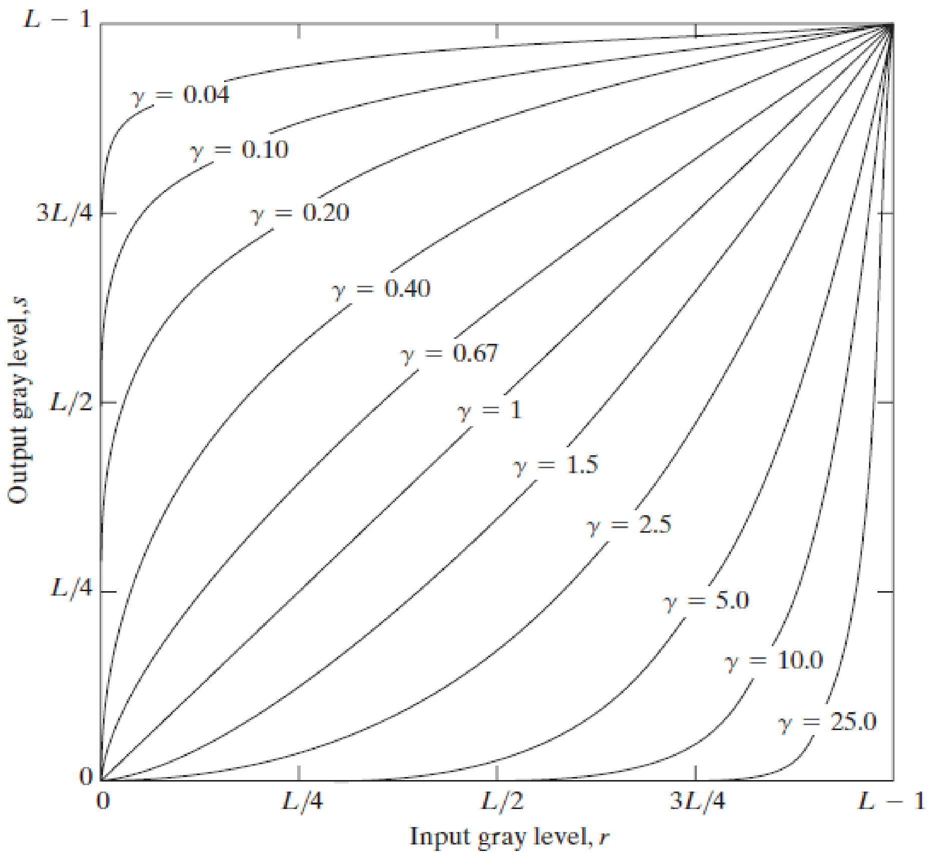
- Power-law (GAMMA) transformation
  - An original image with gray level in the range ***[0, L-1]***
  - Has the basic mathematical form

$$s = c \cdot r^{\gamma}$$

  - Power-law curves with fractional values of γ map a narrow range of dark input values into a wider range of output values, with the opposite being true for higher values of input levels.
  - $c = \gamma = 1$ ⇨ Identity function

Point Processing Techniques(cont…)

- Power-law (GAMMA) transformation



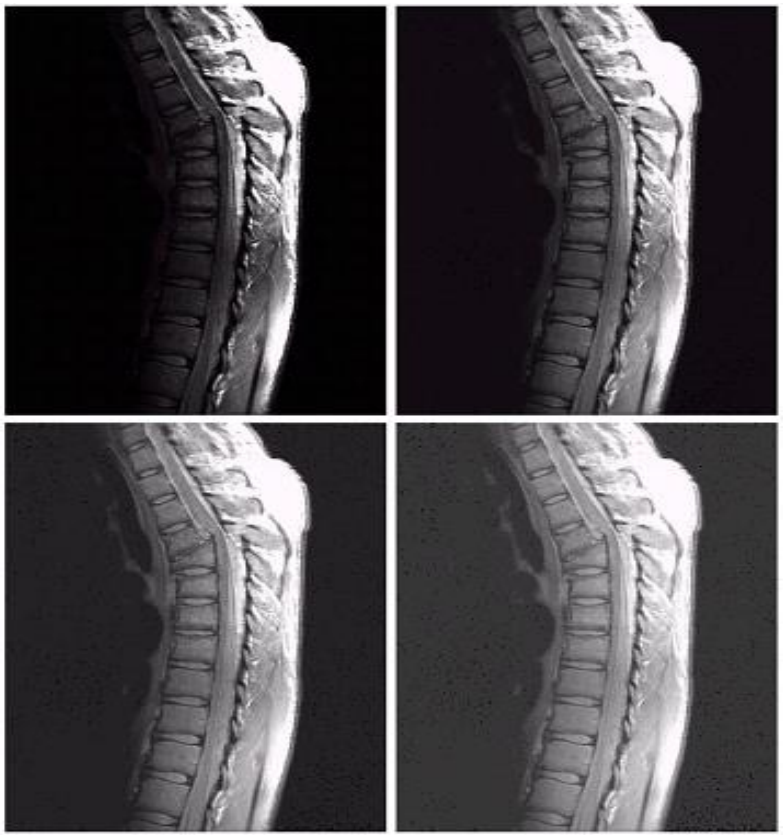
Input gray level, r

Plots of $s = cr^{\gamma}$ for various values of γ (c = 1 in all cases)

Point Processing Techniques(cont…)

- Examples Power-law (GAMMA) transformation

a	b
c	d



(a) A magnetic resonance image of an upper thoracic human spine with a fracture dislocation and spinal cord impingement

- The picture is predominately dark
- An expansion of gray levels are desirable ⇨ needs $\gamma < 1$

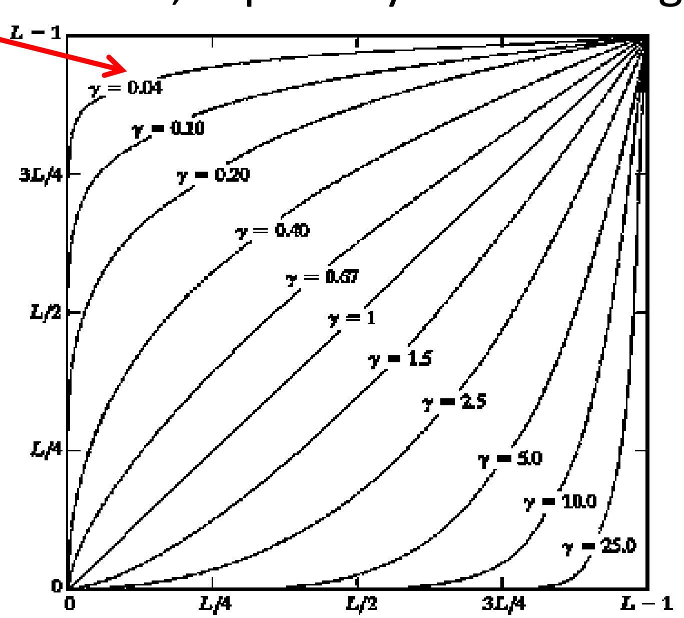
(b) Result after power-law transformation with $\gamma = 0.6$, c=1

(c) Transformation with $\gamma = 0.4$ (best result)

(d) transformation with $\gamma = 0.3$ (under acceptable level)

Point Processing Techniques(cont…)

- Effect of Decreasing Gamma
  - When the γ is reduced too much, the image begins to reduce contrast to the point where the image started to have very slight "wash-out" look, especially in the background



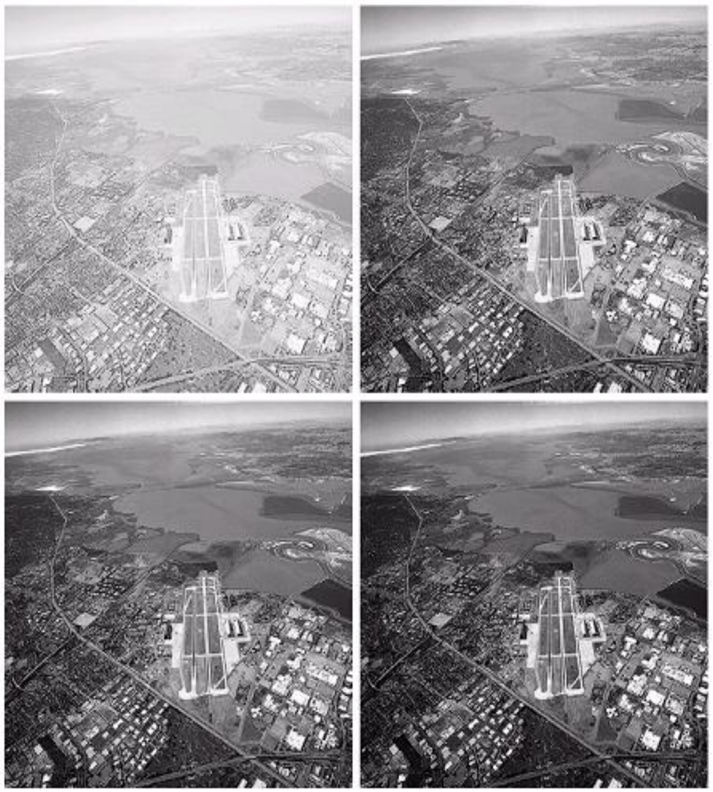
Input gray level, r

Plots of $s = cr^{\gamma}$ for various values of γ (c = 1 in all cases)

Point Processing Techniques(cont…)

- Another Examples Power-law transformation

a	b
c	d



(a) image has a washed-out appearance, it needs a compression of gray levels ⇨ needs $\gamma > 1$

(b) result after power-law transformation with $\gamma = 3.0$ (suitable)

(c) transformation with $\gamma = 4.0$ (suitable)

(d) transformation with $\gamma = 5.0$ (high contrast, the image has areas that are too dark, some detail is lost)

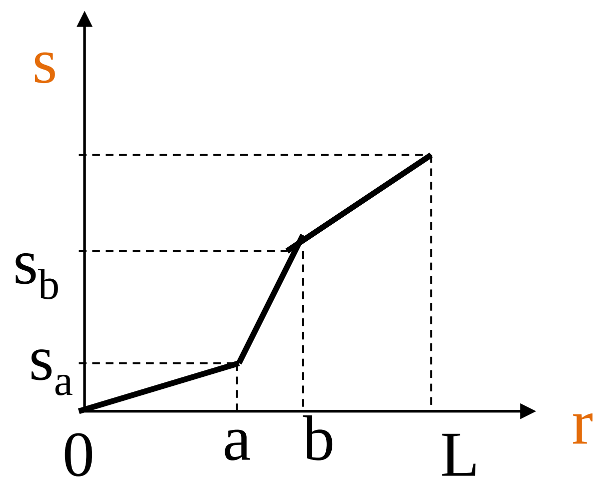
Point Processing Techniques(cont…)

- Piecewise-Linear Transformation Functions
  - Contrast Stretching
  - Thresholding
  - Gray Level Slicing
  - Bit Plane Slicing

Point Processing Techniques(cont...)

- Contrast Stretching

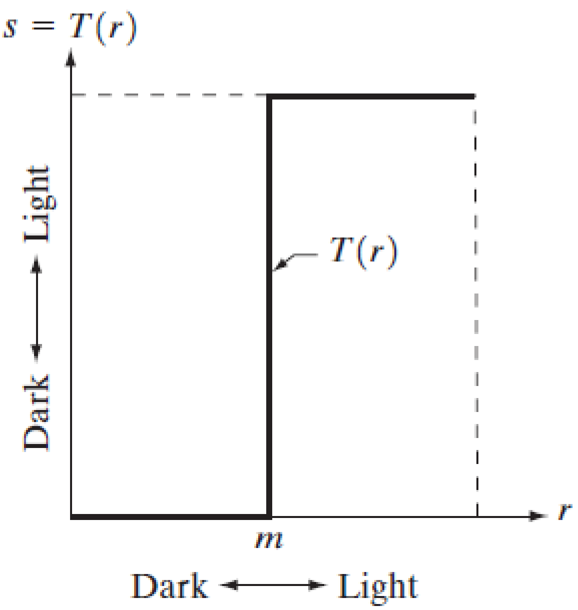
$$s = \begin{cases} \alpha r & 0 \le r < a \\ \beta(r-a) + s_a & a \le r < b \\ \gamma(r-b) + s_b & b \le r < L-1 \end{cases}$$



  - Increases the dynamic range of gray levels in the image
  - Thresholding is a special case of contrast stretching

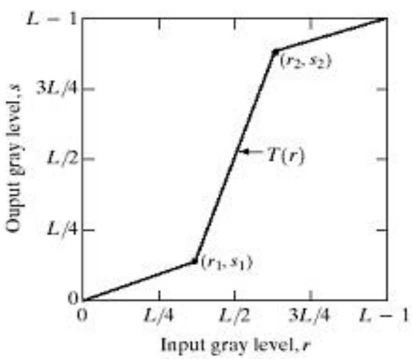
Point Processing Techniques(cont…)

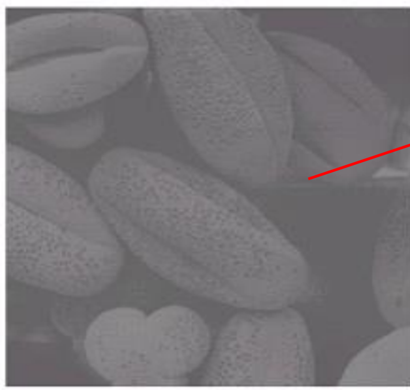
- Thresholding
  - Produces a two level (binary) image
  - Pixels above threshold grouped into one class and below threshold to another class

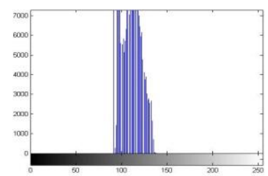


Point Processing Techniques(cont…)

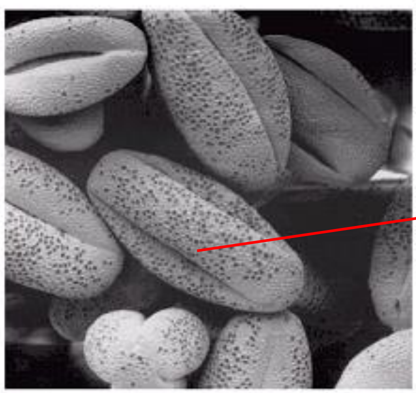
- Contrast Stretching & Thresholding Example

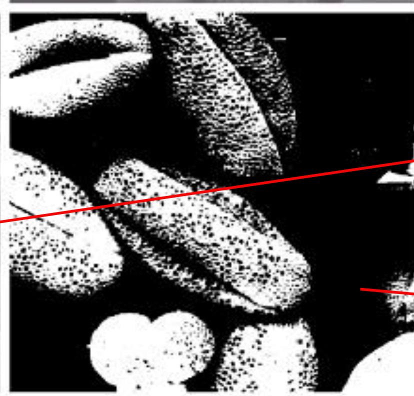






a low-contrast image : result from poor illumination, lack of dynamic range in the imaging sensor, or even wrong setting of a lens aperture of image acquisition



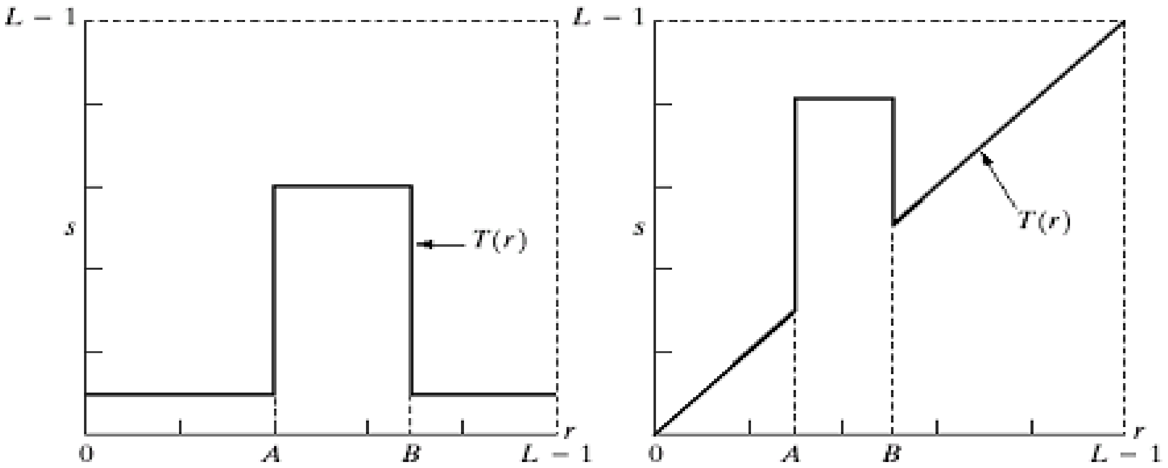


result of contrast stretching:
$(r_1,s_1) = (r_{min},0)$ and
$(r_2,s_2) = (r_{max},L-1)$

result of thresholding

Point Processing Techniques(cont…)

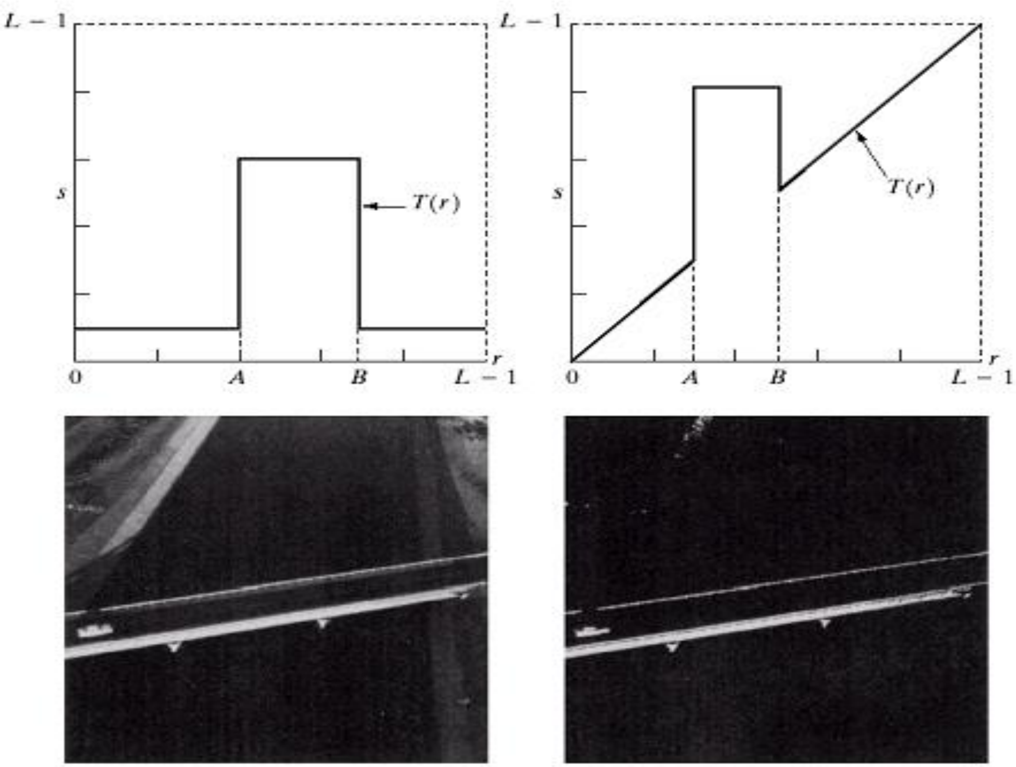
- Gray-level slicing
  - Highlighting a specific range of gray levels in an image
  - Display a high value of all gray levels in the range of interest and a low value for all other gray levels



Point Processing Techniques(cont…)

- Gray-level slicing Example

a	b
c	d

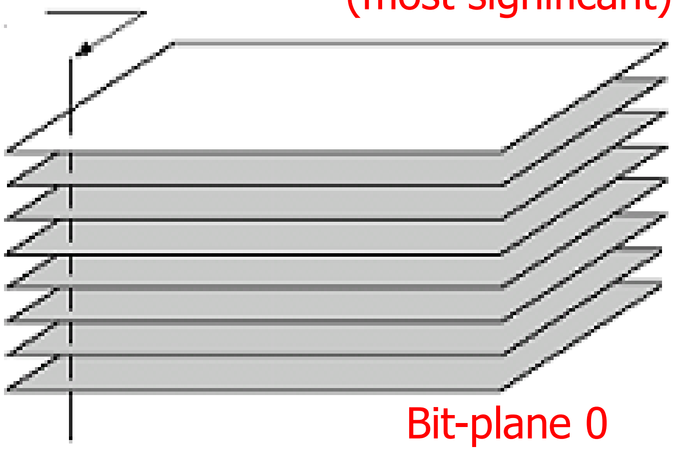


(a) transformation highlights range [A,B] of gray level and reduces all others to a constant level
(b) transformation highlights range [A,B] but preserves all other levels
(c) Original image
(d) Result of applying the transformation (a) on the image at (c)

Point Processing Techniques(cont…)

- Bit-plane Slicing
  - Highlights the contribution made to total image appearance by specific bits

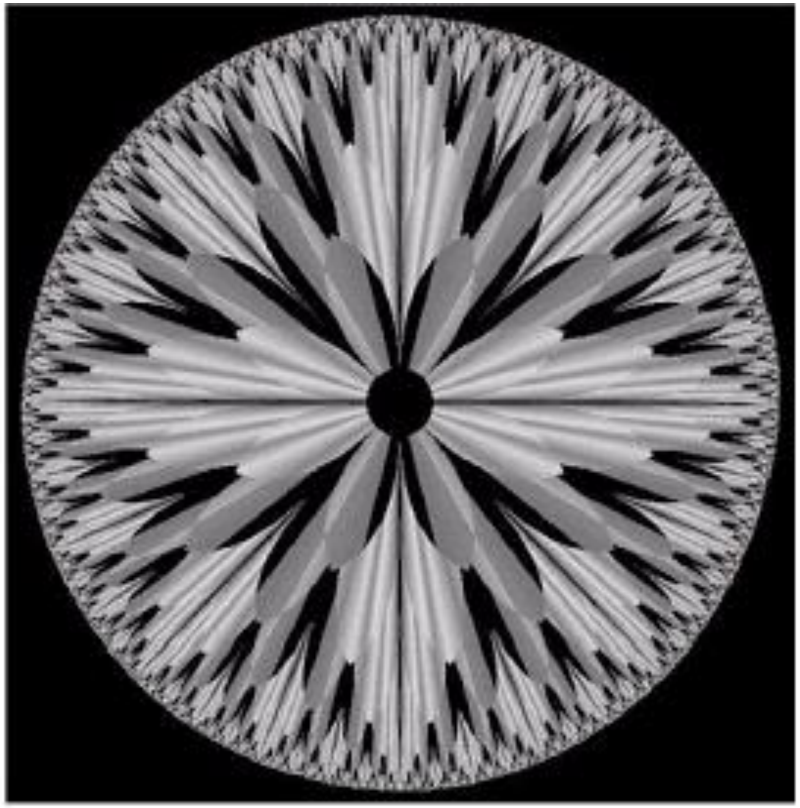
Suppose each pixel is represented by 8 bit



  - Higher-order bits contain the majority of the visually significant data.
  - Useful for analyzing the relative importance played by each bit of the image

Point Processing Techniques(cont…)

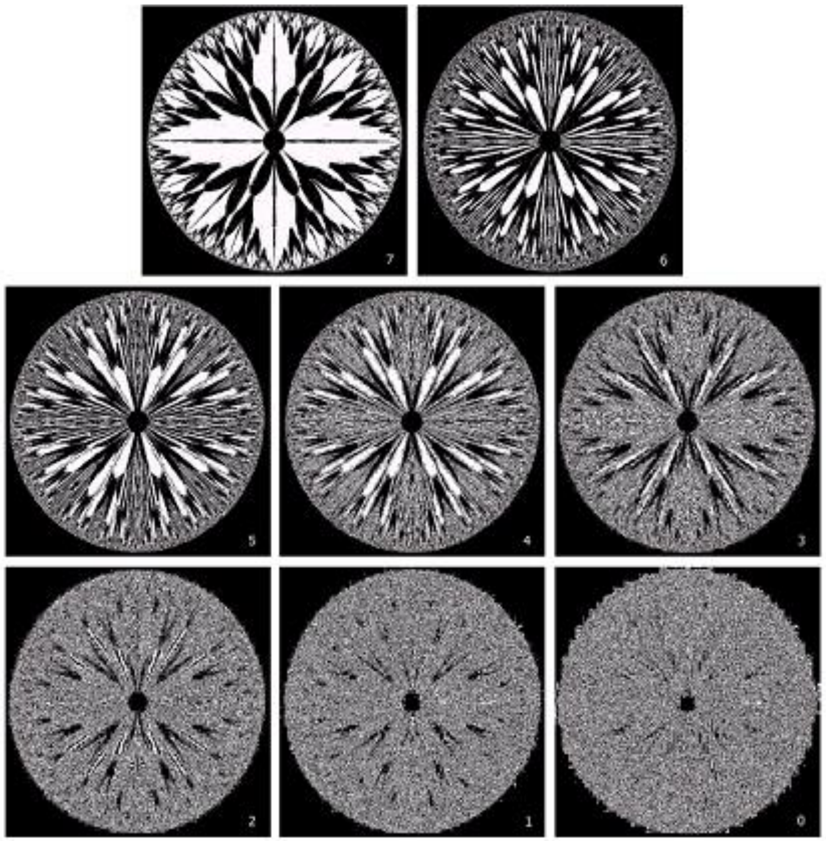
- Bit-plane Slicing Example
  - The (binary) image for bit-plane 7 can be obtained by processing the input image with a thresholding gray-level transformation.
    - Map all levels between 0 and 127 to 0
    - Map all levels between 128 and 255 to 255



An 8-bit fractal image

Point Processing Techniques(cont....)

- Bit-plane Slicing Example (cont....)



Bit-plane 7	Bit-plane 6	
Bit-plane 5	Bit-plane 4	Bit-plane 3
Bit-plane 2	Bit-plane 1	Bit-plane 0

[8- Bit planes]

Summary of Point Processing Techniques

- So far, we have discussed various forms of mapping function T(r) that leads to different enhancement results.

- The natural question is: How to select an appropriate T(r) for an arbitrary image?

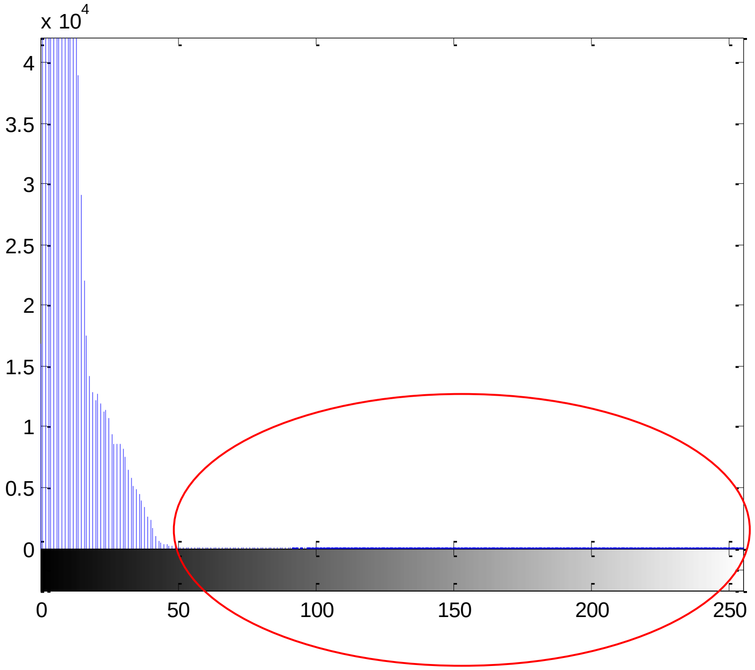
- One systematic solution is based on the histogram information of an image

Histogram Processing Techniques

- Histogram of an image represents the relative frequency of occurrence of various gray levels in the image
- Takes care the global appearance of an image
- Basic method for numerous spatial domain processing techniques
- Used effectively for image enhancement

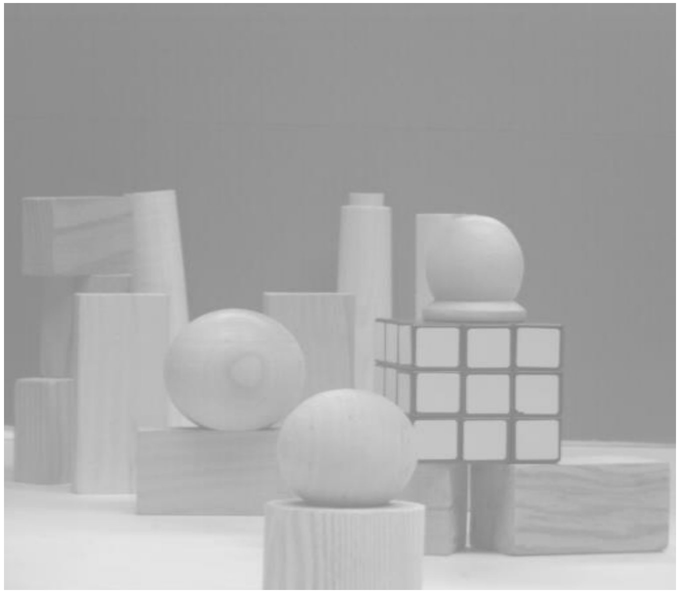
Why Histogram Processing Techniques?

It is a baby in the cradle!

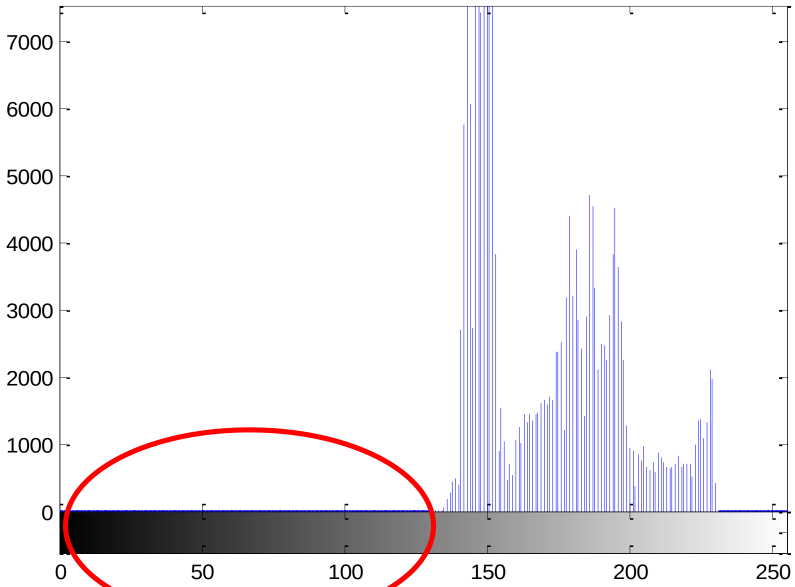


- Histogram information reveals that image is under-exposed

Why Histogram Processing Techniques?



Over-exposed image



- Histogram information reveals that image is over-exposed

Histogram Processing Techniques (cont….)

- Histogram of an Image
  - Histogram of a digital image with gray levels in the range [0 L-1], is a discrete function

$$h(r_k) = n_k$$

  - Where

    r_k : the k[th] gray level

    n_k : the number of pixels in the image having gray level r_k

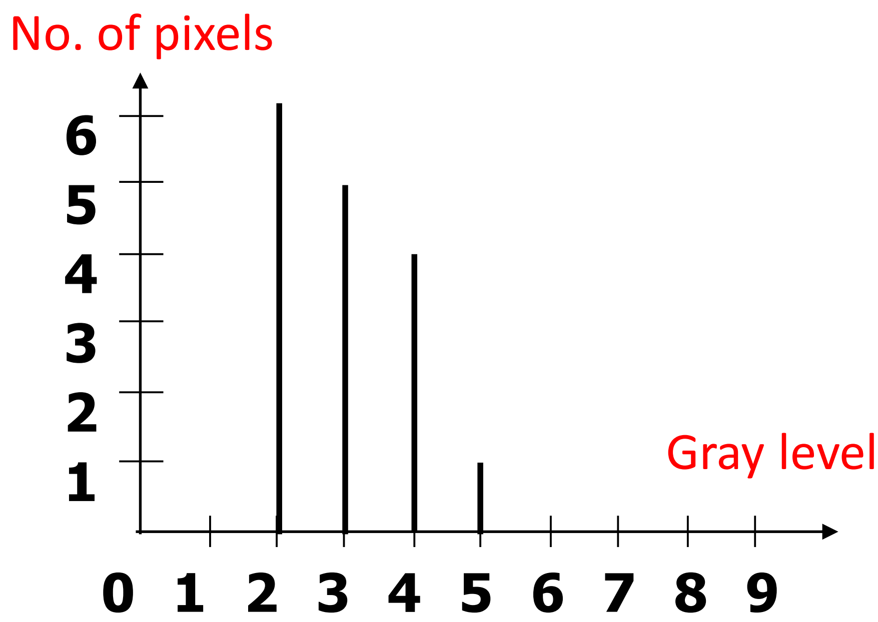
    $h(r_k)$: histogram of a digital image with gray levels r_k

Histogram Processing Techniques (cont....)

- A Basic Histogram Example

2	3	3	2
4	2	4	3
3	2	3	5
2	4	2	4

4x4 image
Gray scale = [0,9]



histogram

Histogram Processing Techniques (cont....)

- Normalized Histogram
  - dividing each of histogram at gray level rk by the total number of pixels in the image, n

$$p(r_k) = n_k/n$$

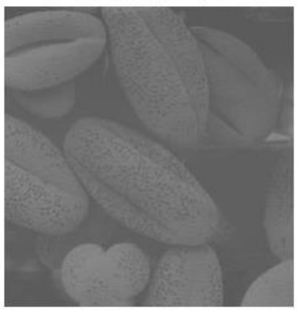
For k = 0,1,…,L-1

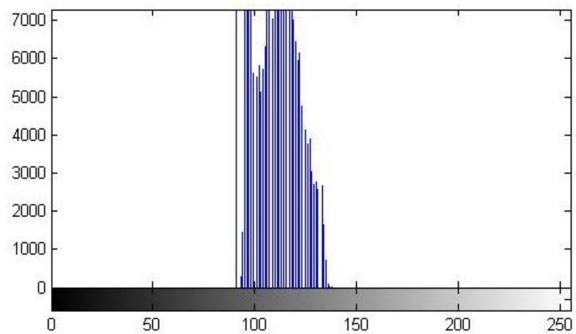
  - $p(r_k)$ gives an estimate of the probability of occurrence of gray level r_k
  - The sum of all components of a normalized histogram is equal to 1

Histogram Processing Techniques (cont….)

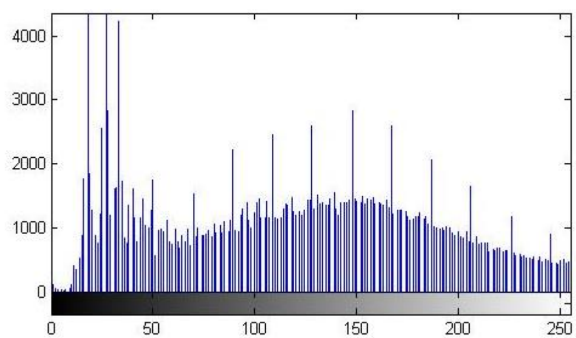
- Histograms of different types of images

Low Contrast Image





High Contrast Image



- Low-contrast image

  histogram is narrow and centered toward the middle of the gray scale

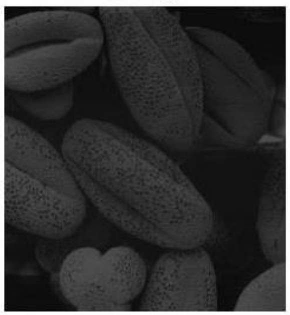
- High-contrast image

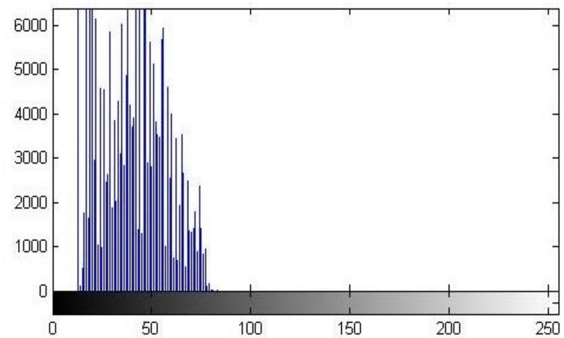
  histogram covers broad range of the gray scale and the distribution of pixels is not too far from uniform, with very few vertical lines being much higher than the others

Histogram Processing Techniques (cont….)

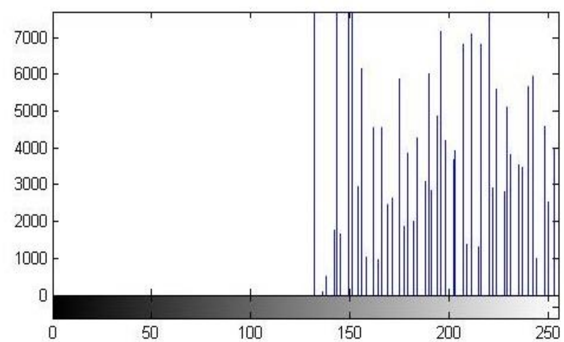
- Histograms of different types of images

Dark Image





Bright Image



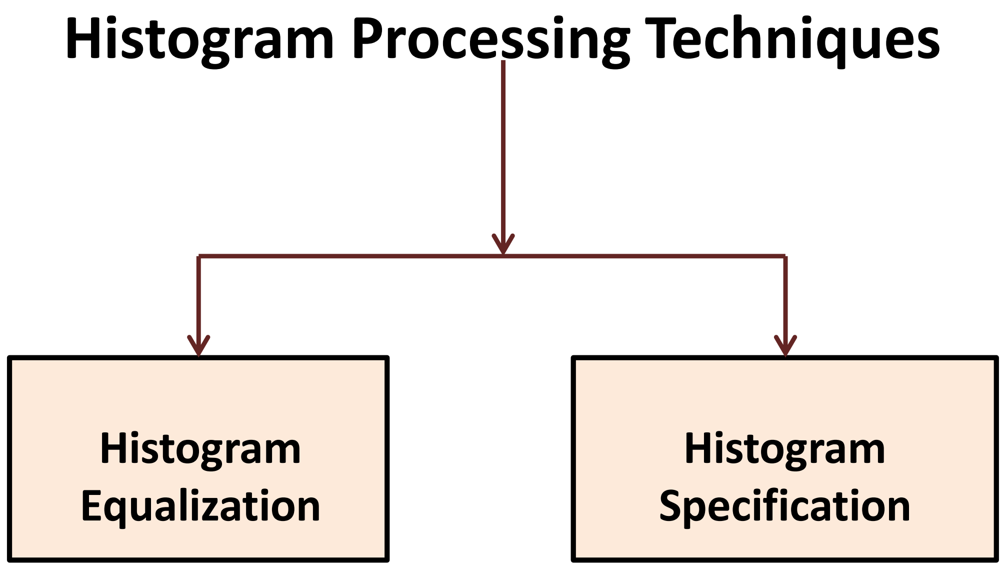
- Dark image

  Components of histogram are concentrated on the low side of the gray scale.

- Bright image

  Components of histogram are concentrated on the high side of the gray scale.

Histogram Processing Techniques (cont….)



- Histogram Specification can also be called *histogram modification* or *histogram matching*

Histogram Processing Techniques (cont....)

- Histogram Equalization

  - *Basic Idea:* find a map *f(x)* such that the histogram of the modified (equalized) image is flat (uniform).

  - *Key Motivation:* probability density function (pdf) of a random variable approximates a uniform distribution.

Suppose h(t) is the histogram (pdf), $s(x) = \sum_{t=0}^{x} h(t)$

Histogram Processing Techniques (cont....)

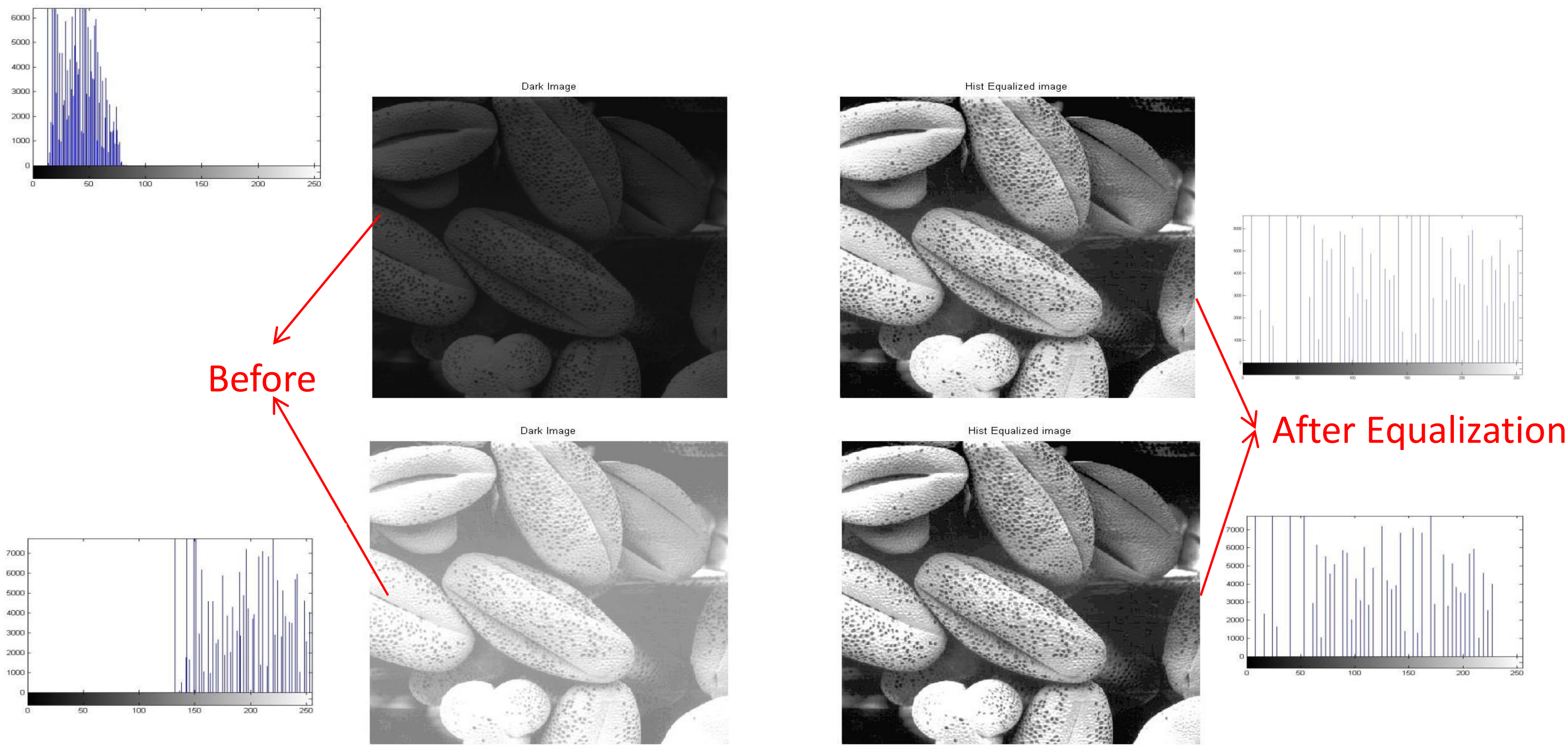
- Histogram Equalization (cont....)

  - As the low-contrast image's histogram is narrow and centred toward the middle of the gray scale, if we distribute the histogram to a wider range the quality of the image will be improved.

  - We can do it by adjusting the probability density function of the original histogram of the image so that the probability spread equally

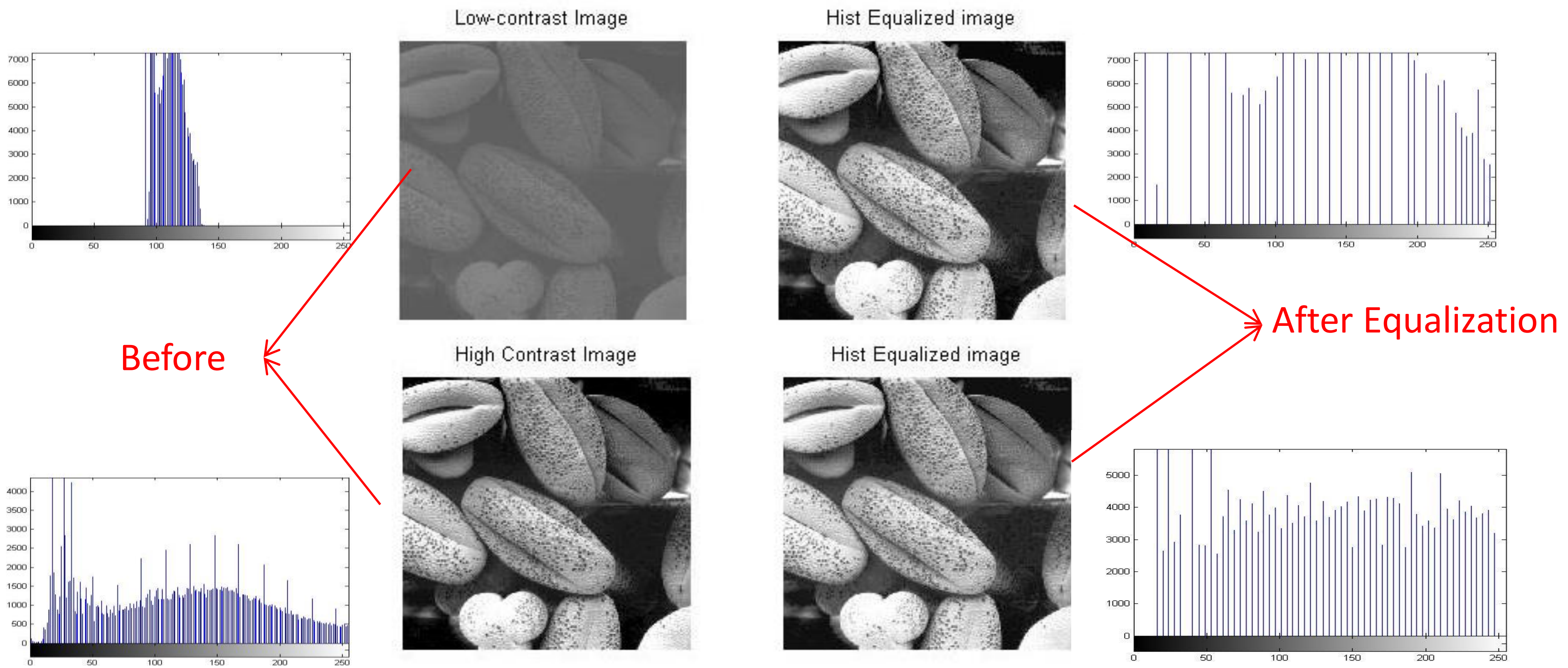
Histogram Processing Techniques (cont....)

- Histogram Equalization Example



Histogram Processing Techniques (cont….)

- Histogram Equalization Example (cont…..)

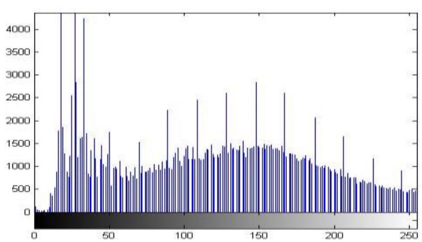


Histogram Processing Techniques (cont....)

- Histogram Equalization Example (cont.....)

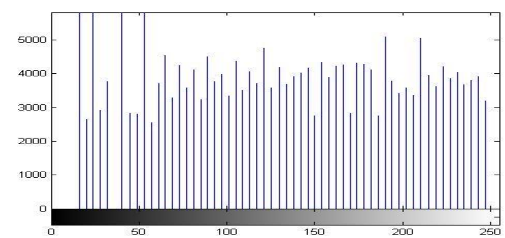
Before

After Equalization



High Contrast Image

Hist Equalized image



- The quality is not improved much because the original image already has a broaden gray-level scale

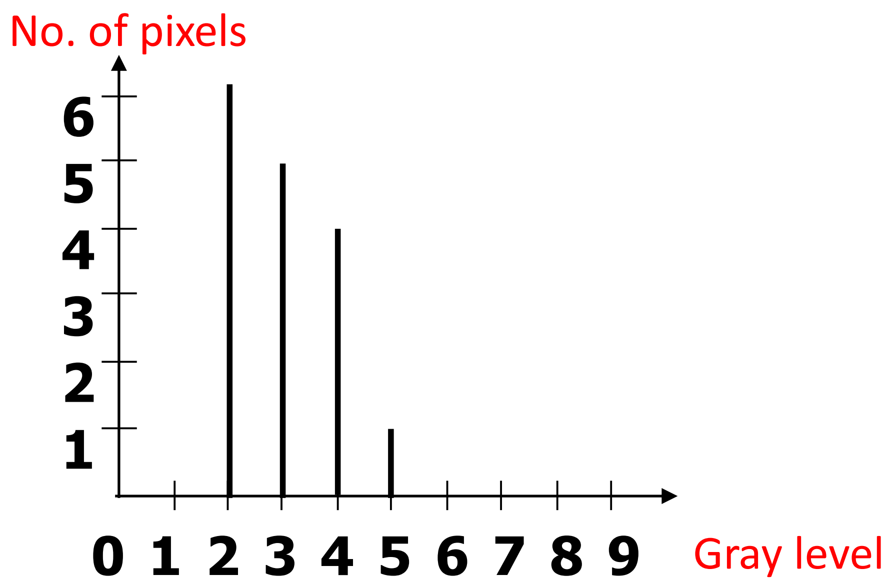
Histogram Processing Techniques (cont....)

- A Sample Example of Histogram Equalization (cont....)

2	3	3	2
4	2	4	3
3	2	3	5
2	4	2	4

4x4 image

Gray scale = [0,9]



histogram

Histogram Processing Techniques (cont....)

- A Sample Example of Histogram Equalization (cont....)

Gray Level(j)	0	1	2	3	4	5	6	7	8	9
No. of pixels	0	0	6	5	4	1	0	0	0	0
$\sum_{j=0}^{k} n_j$	0	0	6	11	15	16	16	16	16	16
$s = \sum_{j=0}^{k} \frac{n_j}{n}$	0	0	6 / 16	11 / 16	15 / 16	16 / 16	16 / 16	16 / 16	16 / 16	16 / 16
s x 9	0	0	3.3 ≈3	6.1 ≈6	8.4 ≈8	9	9	9	9	9

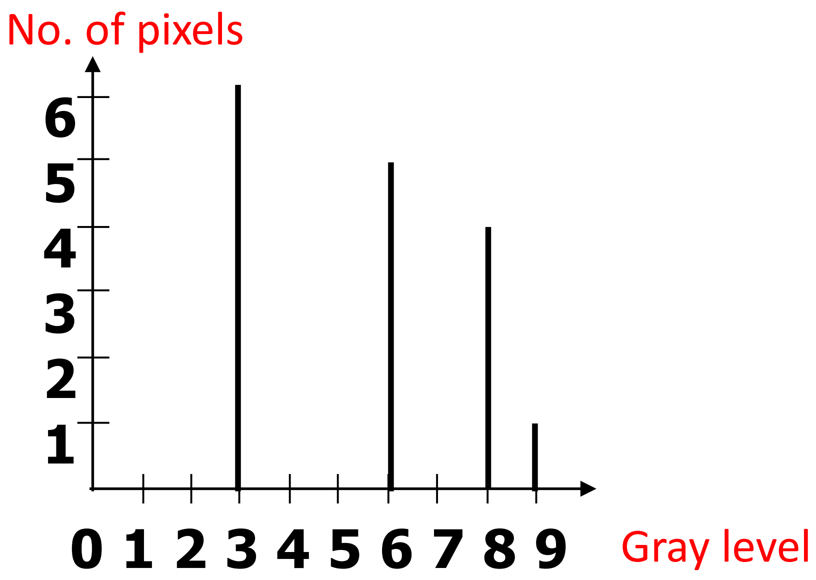
Histogram Processing Techniques (cont....)

- A Sample Example of Histogram Equalization (cont....)

3	6	6	3
8	3	8	6
6	3	6	9
3	8	3	8

Output image

Gray scale = [0,9]



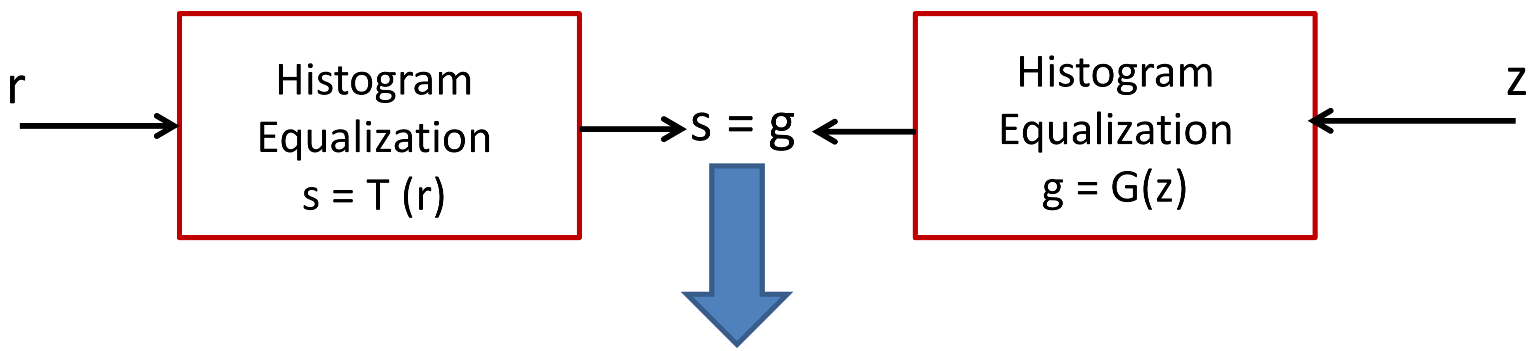
Histogram equalization

Histogram Processing Techniques (cont….)

- Histogram Specification
  - *Basic Idea:* To generate an image which has a specified histogram that we wish the processed image to have
  - *Key Motivation:* The disadvantage of histogram equalization, i.e. it can generate only one type of out put image

Histogram Processing Techniques (cont….)

- Histogram Specification (Block Diagram)



$$z = G^{-1}(s) = G^{-1}(T(r))$$

Histogram Processing Techniques (cont….)

- Histogram Specification (Continuous Domain)

  - Let ***$p_r(r)$*** denote continuous probability density function of gray-level of input image, ***r***
  - Let **$p_z(z)$** denote desired (specified) continuous probability density function of gray-level of output image, ***z***

  - Let ***s*** be a random variable with the property

$$s = T(r) = \int_0^r p_r(w)\,dw \Longrightarrow \text{Histogram equalization}$$

Where ***w*** is a dummy variable of integration

Histogram Processing Techniques (cont….)

- Histogram Specification (Continuous Domain) [cont..]
  - Next, we define a random variable ***z*** with the property

$$g(z) = \int_0^z p_z(t)\,dt = s \quad \Longrightarrow \quad \text{Histogram equalization}$$

Where ***t*** is a dummy variable of integration

$$s = T(r) = G(z)$$

Therefore, z must satisfy the condition

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

Assume G^{-1} exists. Now, we can map an input gray level ***r*** to output gray level ***z***

Histogram Processing Techniques (cont….)

- Summary of Histogram Specification Procedure
  - *Step :1* Obtain the transformation function ***T(r)*** by calculating the histogram equalization of the input image

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

  - *Step :2* Obtain the transformation function ***G(z)*** by calculating histogram equalization of the desired density function

$$G(z) = \int_0^z p_z(t)\,dt = s$$

  - *Step :3* Obtain the inversed transformation function G^{-1}

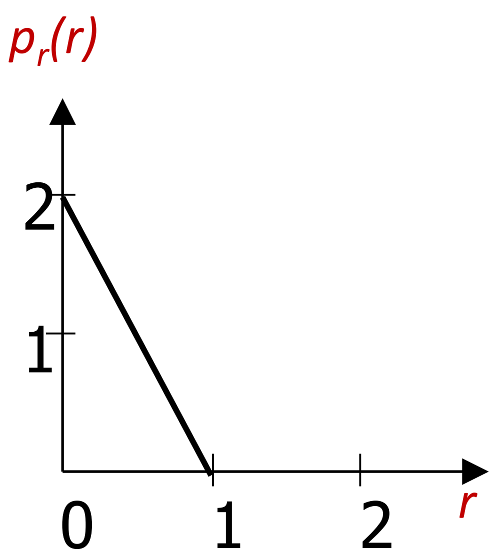
$$z = G^{-1}(s) = G^{-1}[T(r)]$$

  - *Step :4* Obtain the output image by applying the processed gray-level from the inversed transformation function to all the pixels in the input image

Histogram Processing Techniques (cont....)

- An Example of Histogram Specification

Assume an image has a gray level probability density function ***$p_r(r)$*** as shown.



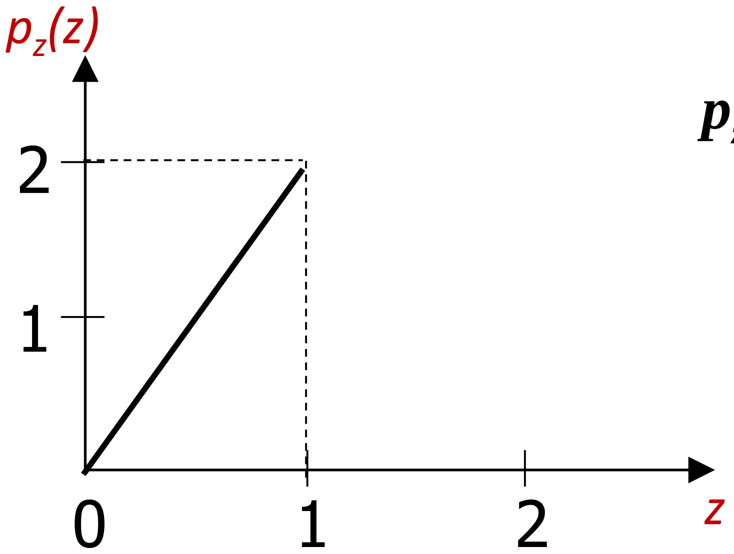
$$p_r(r) = \begin{cases} -2r + 2 & ;0 \le r \le 1 \\ 0 & ;\text{elsewhere} \end{cases}$$

$$\int_0^r p_r(w)\,dw = 1$$

Histogram Processing Techniques (cont....)

An Example of Histogram Specification

- We would like to apply the histogram specification with the desired probability density function $p_z(z)$ as shown.

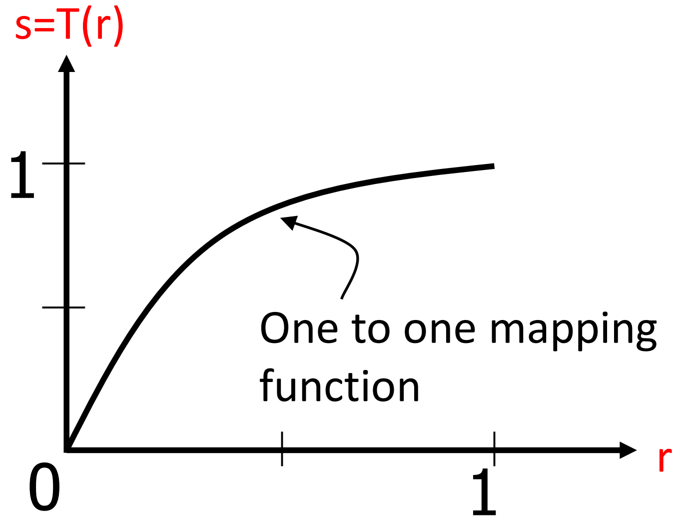


$$p_z(z) = \begin{cases} 2z & ; 0 \le z \le 1 \\ 0 & ; \text{elsewhere} \end{cases}$$

$$\int_0^z p_z(w)\,dw = 1$$

Histogram Processing Techniques (cont....)

- An Example of Histogram Specification (Solution)
  - *Step 1:*
    - Obtain the transformation function ***T(r)***



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

$$= \int_0^r (-2w + 2)\,dw$$

$$= -w^2 + 2w \Big|_0^r$$

$$= -r^2 + 2r$$

Histogram Processing Techniques (cont….)

- An Example of Histogram Specification (Solution)
  - *Step 2:*
    - Obtain the transformation function ***G(z)***

$$G(z) = \int_0^z (2w)dw \qquad = z^2\Big|_0^z \qquad = z^2$$

Histogram Processing Techniques (cont….)

- An Example of Histogram Specification (Solution)
  - *Step 3:*
    - Obtain the inversed transformation function G^{-1}

$$G(z) = T(r)$$

$$z^2 = -r^2 + 2r$$

$$z = \sqrt{2r - r^2}$$

    - We can guarantee that $0 \leq z \leq 1$ when $0 \leq r \leq 1$

Histogram Processing Techniques (cont….)

- Histogram Specification (Discrete Domain)

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

$$= \sum_{j=0}^{k} \frac{n_j}{n} \qquad k = 0,1,2,\ldots, L-1$$

$$G(z_k) = \sum_{i=0}^{k} p_z(z_i) = s_k \qquad k = 0,1,2,\ldots, L-1$$

$$z_k = G^{-1}\left[T(r_k)\right]$$

$$= G^{-1}\left[s_k\right] \qquad k = 0,1,2,\ldots, L-1$$

Histogram Processing Techniques (cont….)

- An Example Histogram Specification (Discrete Domain)

Intensity range
[0   L-1]=[0 7]

Input Histogram		Specified Histogram	
r_k	$pr(r_k)$	z_q	$pr(z_q)$
0	0.19	0	0.00
1	0.25	1	0.00
2	0.21	2	0.00
3	0.16	3	0.15
4	0.08	4	0.20
5	0.06	5	0.30
6	0.03	6	0.20
7	0.02	7	0.15

Histogram Processing Techniques (cont….)

- An Example Histogram Specification (Discrete Domain)

S T E P 1

Input Histogram		
r_k	$pr(r_k)$	$s_k = (L-1)\sum_{j=0}^{k} pr(r_j)$
0	0.19	$1.33 \approx 1$
1	0.25	$3.08 \approx 3$
2	0.21	$4.55 \approx 5$
3	0.16	$5.67 \approx 6$
4	0.08	$6.23 \approx 6$
5	0.06	$6.65 \approx 7$
6	0.03	$6.86 \approx 7$
7	0.02	$7.00 \approx 7$

Intensity range [0 L-1]=[0 7]

Mapping Table	
r_k	s_k
0	1
1	3
2	5
3	6
4	6
5	7
6	7
7	7

Histogram Processing Techniques (cont….)

- An Example Histogram Specification (Discrete Domain)

STEP 2

Specified Histogram		
z_q	$pr(z_q)$	$g_q = (L-1)\sum_{j=0}^{k} pr(z_j)$
0	0.00	$0.00 \approx 0$
1	0.00	$0.00 \approx 0$
2	0.00	$0.00 \approx 0$
3	0.15	$1.05 \approx 1$
4	0.20	$2.45 \approx 2$
5	0.30	$4.55 \approx 5$
6	0.20	$5.95 \approx 6$
7	0.15	$7.00 \approx 7$

Intensity range [0 L-1]=[0 7]

Mapping Table	
z_q	g_q
0	0
1	0
2	0
3	1
4	2
5	5
6	6
7	7

Histogram Processing Techniques (cont....)

- An Example Histogram Specification (Discrete Domain)

Intensity range
[0   L-1]=[0 7]

S
T
E
P

3

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

Map $\mathbf{s_k}$ to $\mathbf{z_q}$ based on common $\mathbf{g_q}$

Histogram Processing Techniques (cont….)

- An Example Histogram Specification (Discrete Domain)

Intensity range
[0 L-1]=[0 7]

S
T
E
P

4

Mapping Table		
r_k	s_k	z_q
0	1	3
1	3	4
2	5	5
3	6	6
4	6	6
5	7	7
6	7	7
7	7	7

Final Mapping

$$r_k \longrightarrow z_q$$

Histogram Processing Techniques (cont….)

- Comparison Between Histogram Equalization and specification using Example

Image of Mars moon

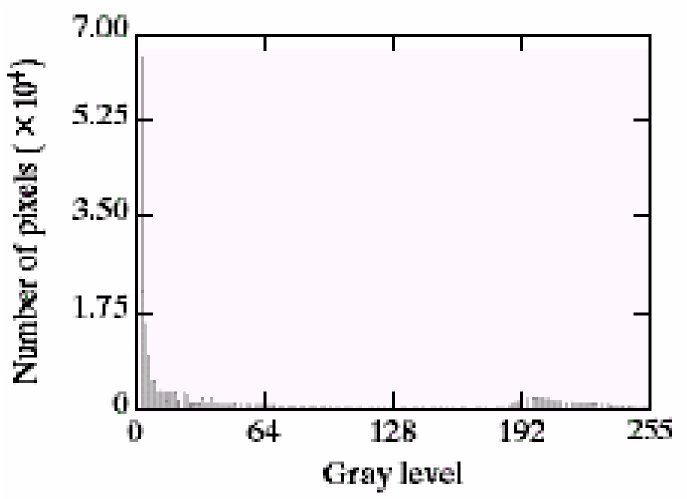
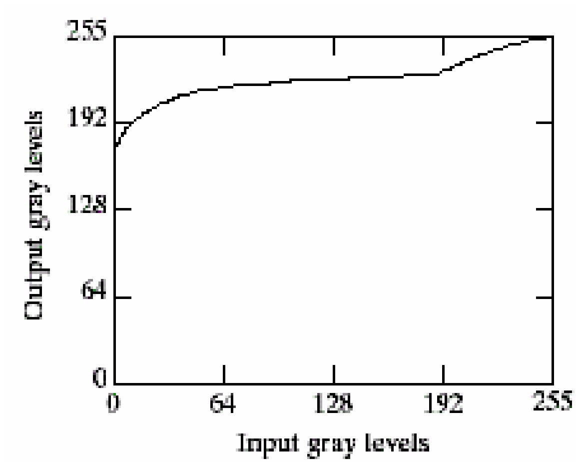


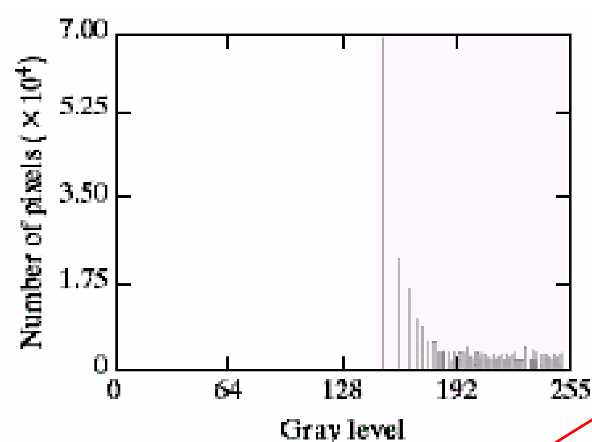
Image is dominated by large, dark areas, resulting in a histogram characterized by a large concentration of pixels in pixels in the dark end of the gray scale

Histogram Processing Techniques (cont.…)

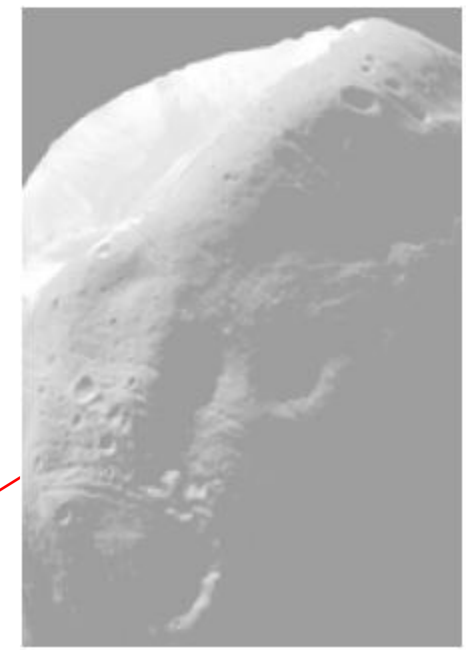
- Comparison Between Histogram Equalization and specification using Example



Transformation function for histogram equalization



Histogram of the result equalized image

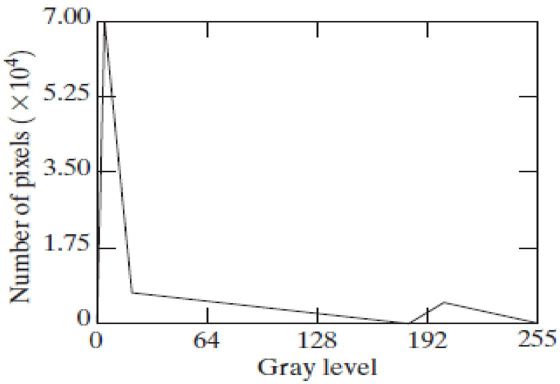


Resultant equalized image

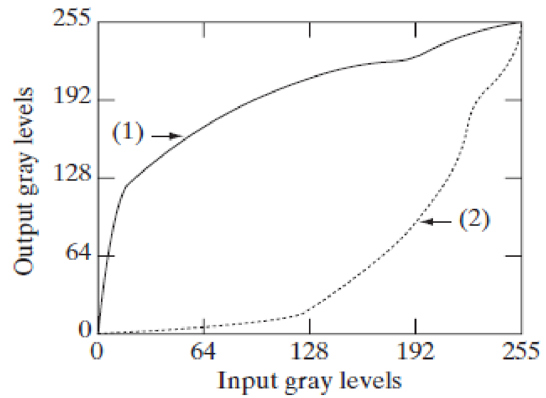
Look at the washed out appearance of equalized image which is not at all expected

Histogram Processing Techniques (cont….)

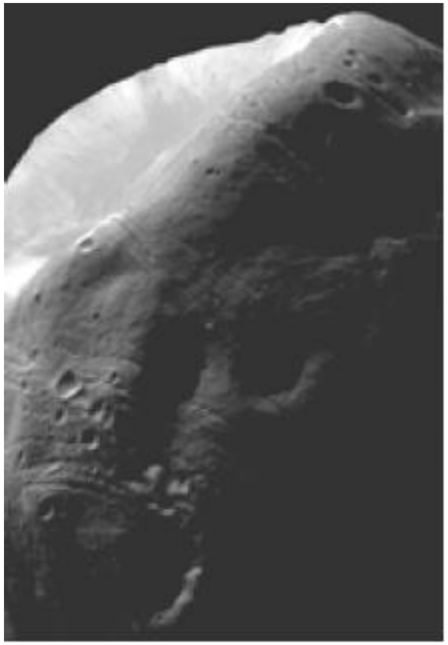
- Comparison Between Histogram Equalization and specification using Example



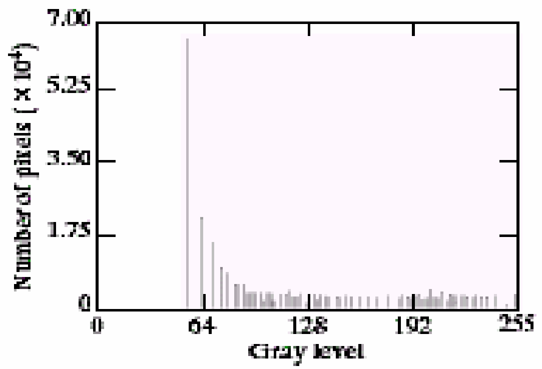
Specified Histogram



Transformations



Resultant Histogram



Resultant Specified image

Histogram Processing Techniques (cont....)

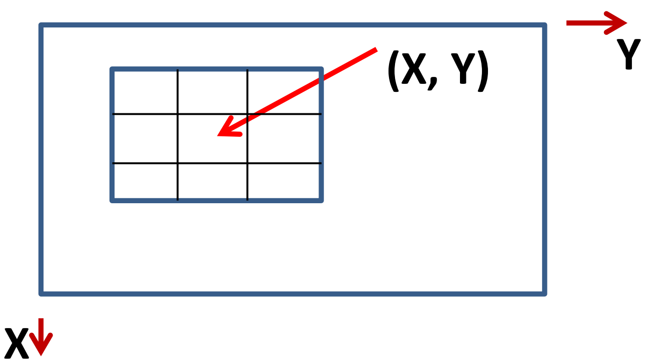
- A Special note on Histogram Specification
  - Histogram specification is a trial-and-error process
  - There are no rules for specifying histograms, and one must resort to analysis on a case-by-case basis for any given enhancement task.

Mask Processing Techniques

- Neighborhood under consideration is greater than 1×1 pixel.
- That means the neighbourhood must be of size 3×3, 5×5 or 7×7 etc.
- Instead of acting on a single pixel, the transformation function is acting over the neighbourhood of the pixel under consideration
- The transformation function is in the form of a ***mask.***
- ***Mask*** value shows what kind of enhancement is going to be achieved.

Mask Processing Techniques (cont....)

- 3 × 3 Neighbourhood of a pixel (X, Y) and the Mask



$W_{-1,-1}$	$W_{-1,0}$	$W_{-1,1}$
$W_{0,-1}$	$W_{0,0}$	$W_{0,1}$
$W_{1,-1}$	$W_{0,1}$	$W_{1,1}$

3×3 MASK

- Mathematically, the enhance image can be written as

$$g(x,y)=\sum_{i=-1}^{1}\sum_{j=-1}^{1} w_{i,j}\, f(x+i, y+j)$$

Mask Processing Techniques (cont....)

- Mask processing techniques can be of the following types
  - Linear Smoothing/ Linear Filtering
  - Median Filtering
  - Sharpen Filtering

Mask Processing Techniques (cont....)

- Linear Smoothing
  - Image Averaging Operation
    - Replaces the intensity value of the pixel under consideration with the *average* of the intensity values of its neighbourhood.
    - The mask used here is

$$Mask = \frac{1}{9} \times \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

    - O/p Image is a smooth image i.e. sharp edges are blurred.
    - Blur will be more if mask size is getting bigger and bigger.

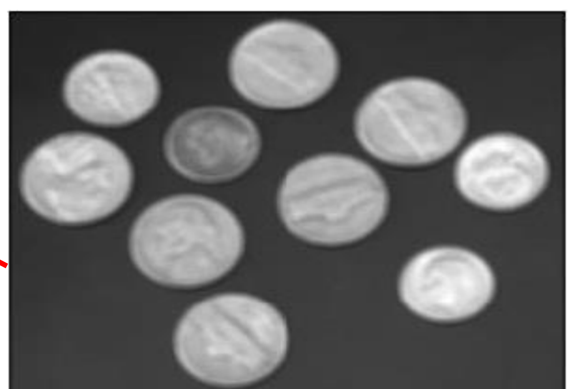
Mask Processing Techniques (cont….)

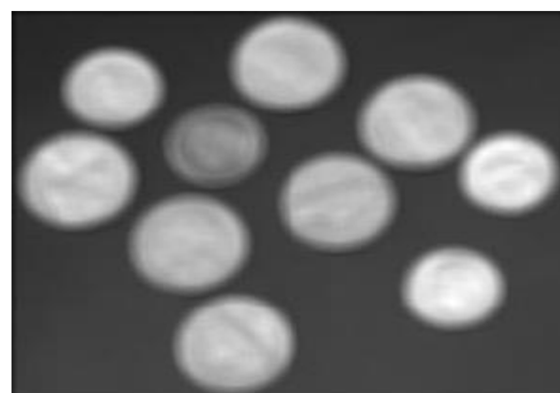
- Examples of Linear Smoothing
  - Image Averaging Operation

Original Image

Using 3×3 mask

Using 5×5 mask





Using 9×9 mask

Mask Processing Techniques (cont....)

- Linear Smoothing (cont....)
  - Weighted Averaging Operation
    - Replaces the intensity value of the pixel under consideration with the *weighted-average* of the intensity values of its neighbourhood.
    - The mask used here is

$$Mask = \frac{1}{\sum_{i=-1}^{1}\sum_{j=-1}^{1} w_{i,j}} \times \begin{array}{|c|c|c|} \hline W_{-1,-1} & W_{-1,0} & W_{-1,1} \\ \hline W_{0,-1} & W_{0,0} & W_{0,1} \\ \hline W_{1,-1} & W_{0,1} & W_{1,1} \\ \hline \end{array}$$

$$Mask = \frac{1}{16} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

[An Example]

    - Blur will be some how reduced as compared to simple averaging.

Mask Processing Techniques (cont....)

- Examples of Linear Smoothing
  - Weighted Averaging Operation

Original Image

Simple Avg
Using 3×3 mask

Weighted Avg
Using 3×3
mask

Mask Processing Techniques (cont….)

- Median Filtering
  - Based on order statistics and hence is a non-linear filter
  - Reduces salt & pepper noise effectively
  - Response is based on ordering of the intensities in the neighbourhood of the point under consideration

[Before Filtering]

100	85	98
99	95	102
90	101	108

Sorting of neighboring pixels in Ascending Order

85	90	95	98	99	100	101	102	108

Median

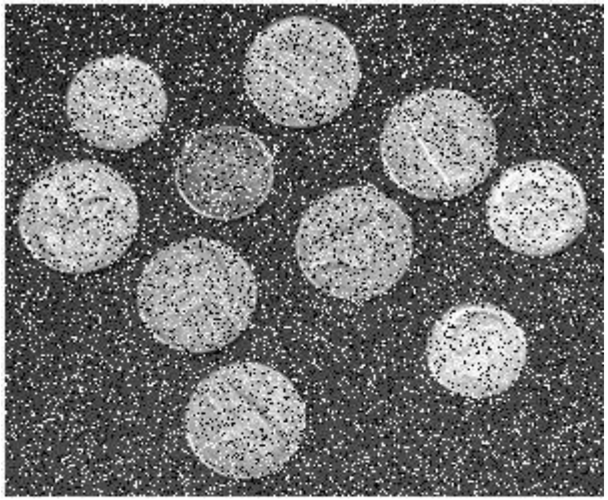
99

[After Filtering]

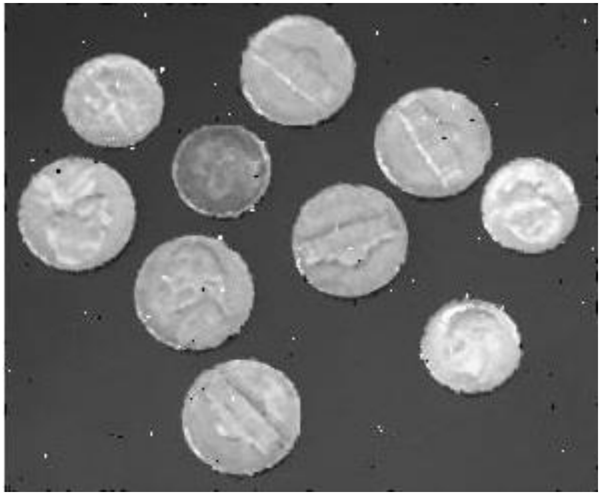
100	85	98
99	99	102
90	101	108

Mask Processing Techniques (cont….)

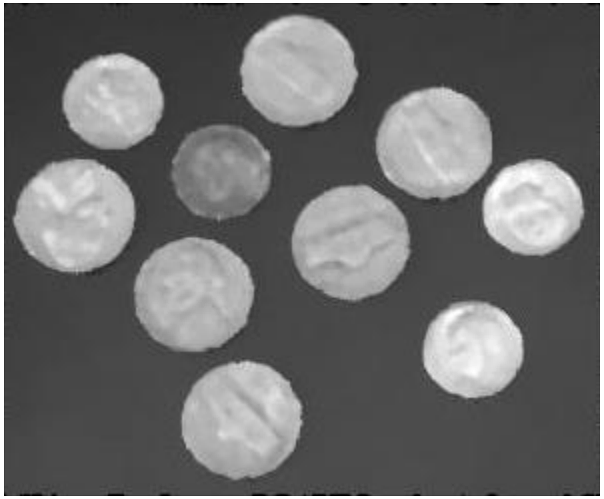
- Example of Median Filtering
  - Helps in reducing salts & pepper noise



Coin with 20% salt & Pepper Noise



Using a 3×3 Median Filter



Using a 5×5 Median Filter

Mask Processing Techniques (cont....)

- Sharpen Filtering
  - Increases the sharpness of the image
  - Objective is to highlight the intensity details of an image
  - Derivative operators generally increase the sharpness of the image
  - First order and second order derivative operators are used for this purpose

Mask Processing Techniques (cont....)

- Sharpen Filtering (cont....)

First Order Derivative Filter	Second Order Derivative Filter
- Must be zero in areas of constant gray level - Non-zero at the *onset* of a gray-level step or ramp - Non-zero along ramp	- Zero in flat areas - Non-zero at the *onset and end* of a gray level step or ramp - Zero along ramp of constant slope

Mask Processing Techniques (cont....)

- Sharpen Filtering (cont....)
  - Derivative in continuous domain

$$\frac{df}{dx} = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

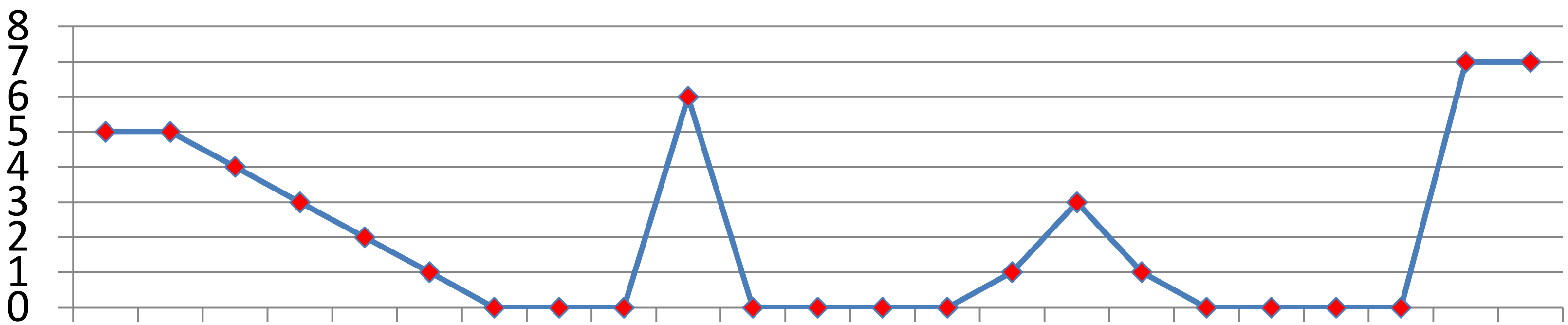
  - Derivative in discrete domain
    - In image, the smallest possible value of ***'h'*** is **1,** then the discrete version of derivative is

$$\frac{df}{dx} = f(x+1) - f(x) \longrightarrow$$ First Order Derivative

$$\frac{d^2 f}{dx^2} = f(x+1) + f(x-1) - 2f(x) \longrightarrow$$ Second Order Derivative

Mask Processing Techniques (cont....)

- Sharpen Filtering (cont....)



5	5	4	3	2	1	0	0	0	6	0	0	0	0	1	3	1	0	0	0	0	7	7	**1D Signal**

0	-1	-1	-1	-1	-1	0	0	6	-6	0	0	0	1	2	-2	-1	0	0	0	7	0	→ *f'*

-1	0	0	0	0	1	0	6	-12	6	0	0	1	1	-4	1	1	0	0	7	-7	→ *f''*

[Observation of 1st order and 2nd order Derivative over a 1-D signal]

Mask Processing Techniques (cont....)

- Sharpen Filtering (cont....)
  - Observations from the previous Example
    - 1st order derivative generally produces thicker edges in an image
    - 2nd order derivative gives stronger response to fine details such as thin lines and isolated points
    - 1st order derivative has stronger response to gray level step
    - 2nd order derivative produces a double response at step edges
  - We can conclude that 2nd order derivative are better suited for image enhancement
    - Most widely used 2nd order derivative is ***Laplacian Operator***

Mask Processing Techniques (cont....)

- Sharpen Filtering (cont....)
  - Formulation of Laplacian operator
    - Laplacian of a function in continuous domain

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

    - Laplacian in discrete domain

$$\frac{\partial^2 f}{\partial x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y)$$

$$\frac{\partial^2 f}{\partial y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y)$$

Mask Processing Techniques (cont....)

- Sharpen Filtering (cont....)
  - Formulation of Laplacian operator (cont....)
    - Laplacian of a function in discrete domain

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y)$$

    - Laplacian masks

0	1	0
1	-4	1
0	1	0

0	-1	0
-1	4	-1
0	-1	0

For Horizontal & vertical Lines

1	1	1
1	-8	1
1	1	1

-1	-1	-1
-1	8	-1
-1	-1	-1

For Horizontal & vertical and diagonal Lines

Mask Processing Techniques (cont....)

- Sharpen Filtering (cont....)
  - Example using Laplacian Operator

Blurry Moon image

Enhanced image using Laplacian

Thanks for your kind attention !!!