Unit:2
Image Processing

Prepared By:
Prof. Janki Patel
Department of IT
SPCE, Bakrol

Content

- Pixel transforms,
- Color transforms,
- Histogram processing and equalization,
- Filtering
- Convolution
- Fourier transformation and its applications in sharpening
- Blurring and noise removal

What is the image processing?

"One picture is worth more than ten thousand words"

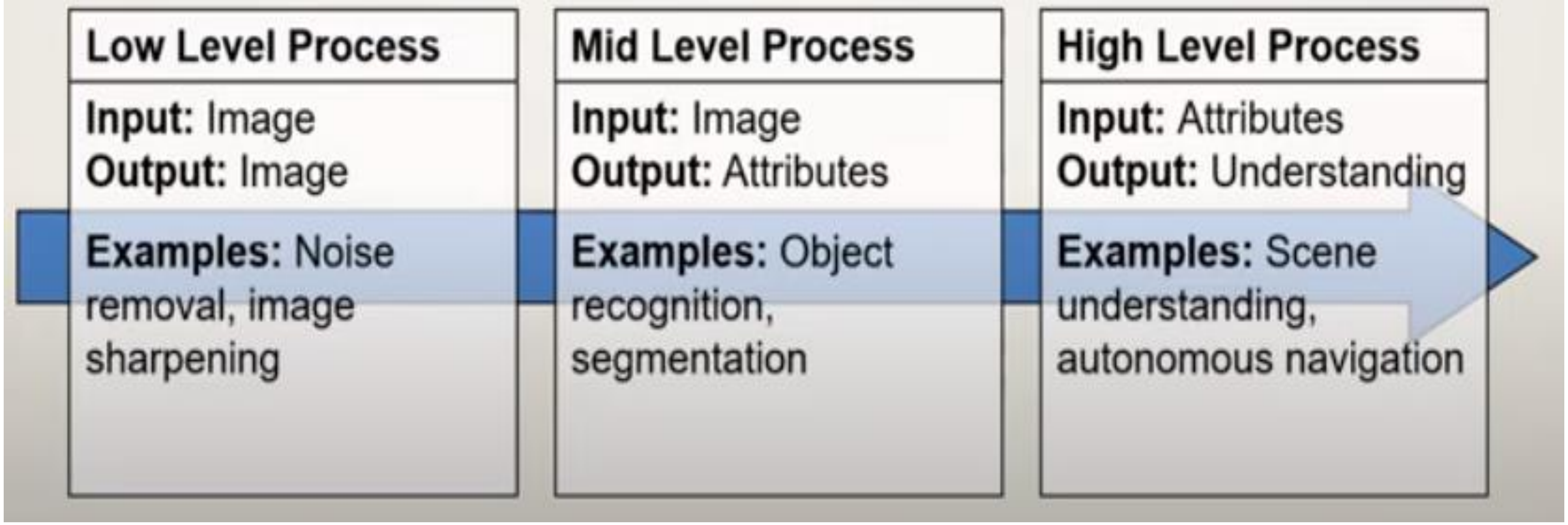
Anonymous

What is Digital Image Processing?

- Digital image processing focuses on two major tasks
  - Improvement of pictorial information for human interpretation
  - Processing of image data for storage, transmission and representation for autonomous machine perception.

Cont…

- The continuum from image processing to computer vision can be broken up into low, mid and high-level processes.



History of Digital Image Processing

- Early 1920s: One the first applications of digital imaging was in the news paper industry.
- The Bart lane cable picture transmission service
- Image were transferred by submarine cable between London and New York.
- Pictures were coded for cable transfer and reconstructed at the receiving end on a telegraph printer

Early digital image

Cont…

- Mid to late 1920s: Improvements to Bartlane system resulted in higher quality images
  - New reproduction processes based on photographic techniques
  - Increased number of tones in reproduced images

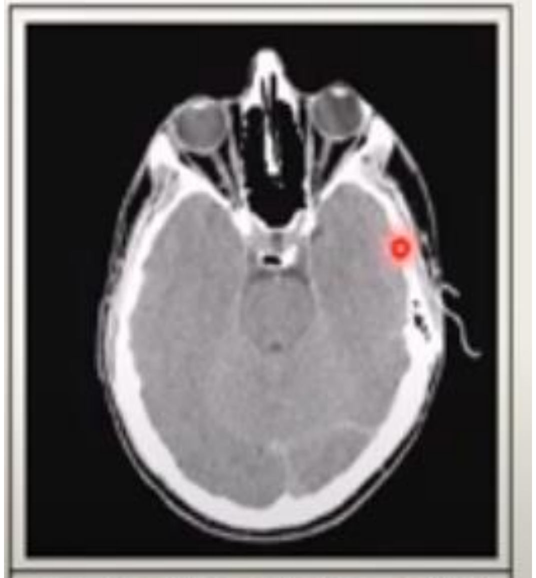
Cont…

- 1960s: Improvements in computing technology and onset of the space race led to a surge of work in digital image processing.
- 1964: Computers used to improve the quality of image of the moon taken by the Ranger 7 probe
- Such techniques were used in other space missions including the Apollo landings

A picture of the moon taken by the Ranger 7 probe minutes before landing

Cont…

- 1970s: Digital image processing begins to be used in medical applications.
  - 1979: Sir Godfrey N. Hounsfield & Prof. Allan M. Cormack share the Nobel Prize in medicine for the invention of tomography, the technology behind Computerized Axial Tomography (CAT)scans



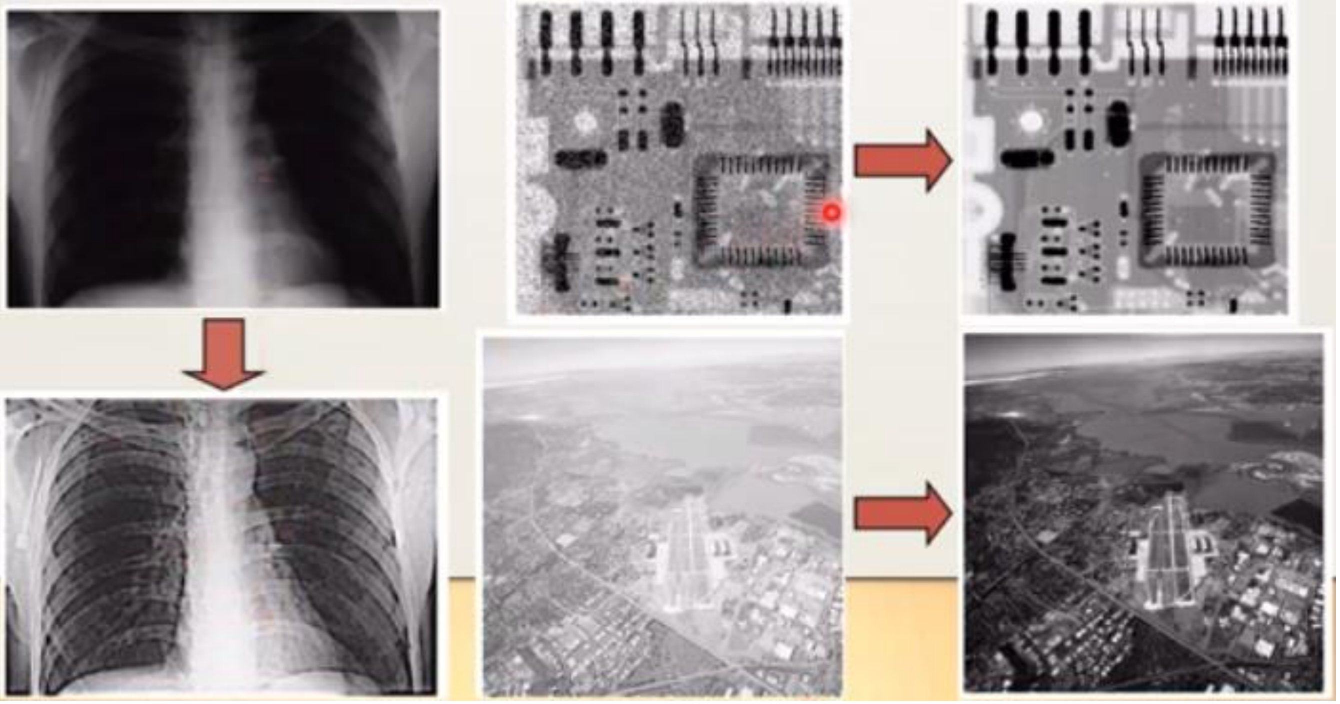
Typical head slice CAT image

Cont…

- 1980s-Today: The use of digital image processing techniques has exploded and they are now used for all kinds of tasks in all kinds of areas.
  - Image enhancement/restoration
  - Artistic effects
  - Medical visualization
  - Industrial inspection
  - Law enforcement
  - Human computer interfaces

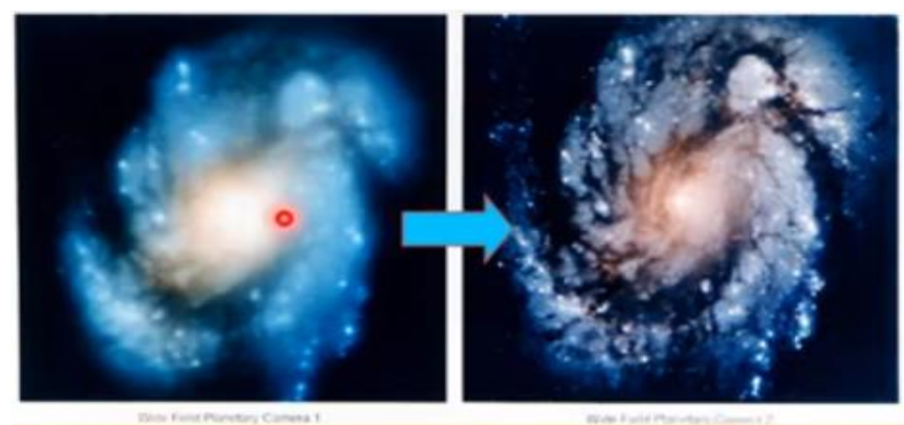
Examples: Image Enhancement

- One of the most common of DIP techniques: improve quality, remove noise etc.



Examples: The Hubble Telescope

- Launched in 1990 the Hubble telescope can take images of very distant objects.
- However, an incorrect mirror made many of Hubble’s images useless
- Image processing techniques were used to fix this

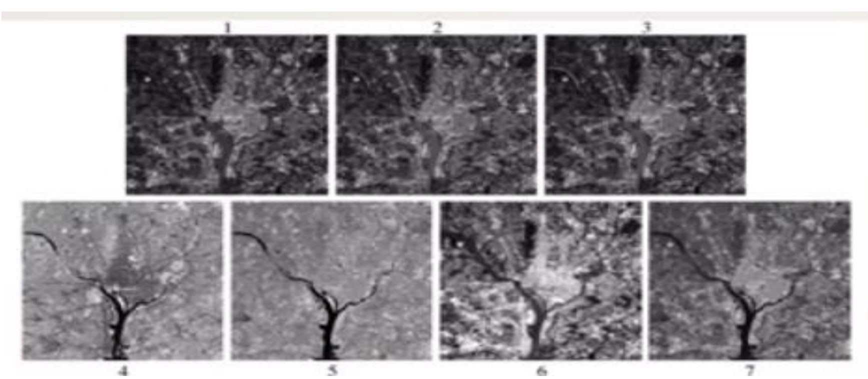


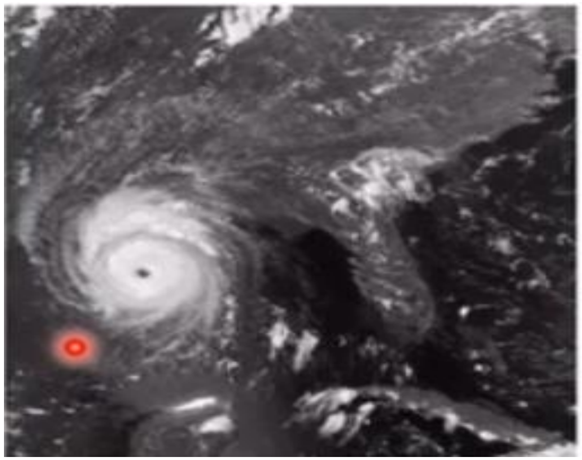
Examples: Artistic Effects

- Artistic effects are used to make images more visually appealing, to add special effects and to make composite images.

Examples: GIS

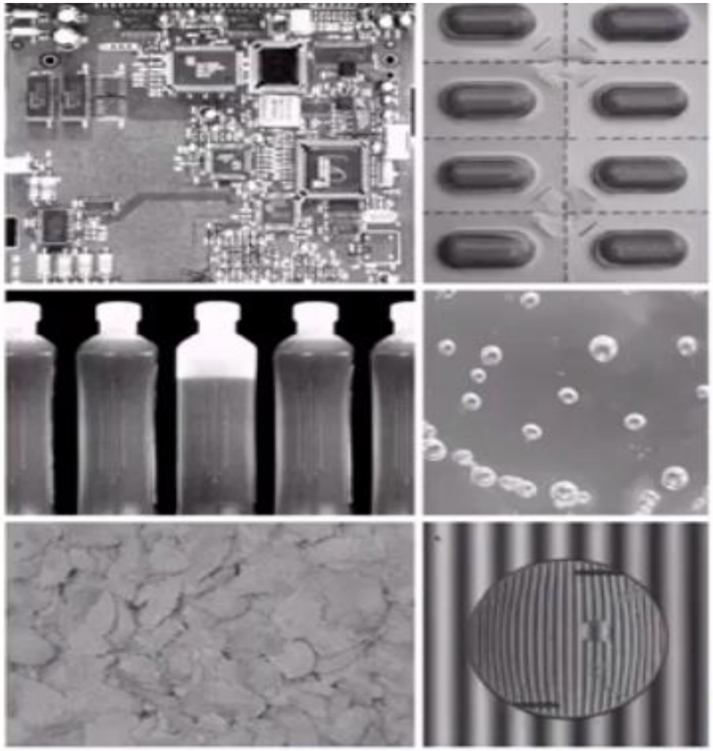
- **Geographic Information System**
  - Digital image processing techniques are used extensively to manipulate satellite imagery
  - Terrain classification
  - Meteorology





Examples: Industrial Inspection

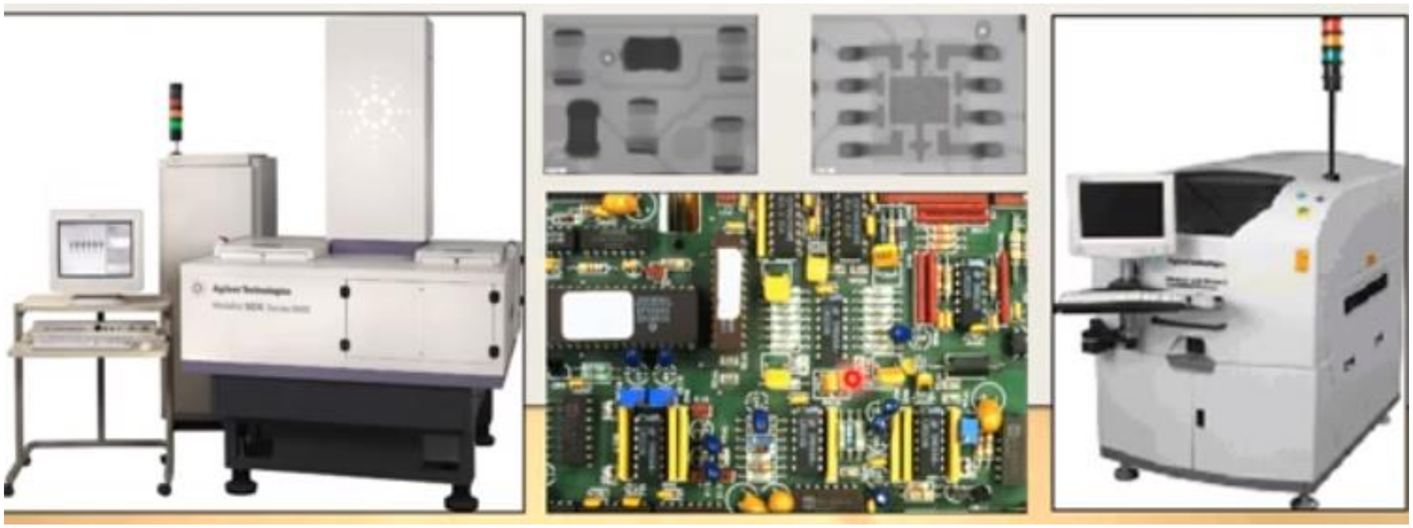
- Human operators are expensive, slow and unreliable
- Make machines do the job instead
- Industrial vision systems are used in all kinds of industries



Examples: PCB Inspection

Printed Circuit Board(PCB)inspection

- Machine inspection is used to determine that all components are present and that all solder joints are acceptable
- Both conventional imaging and x- ray imaging are used



Examples: Law Enforcement

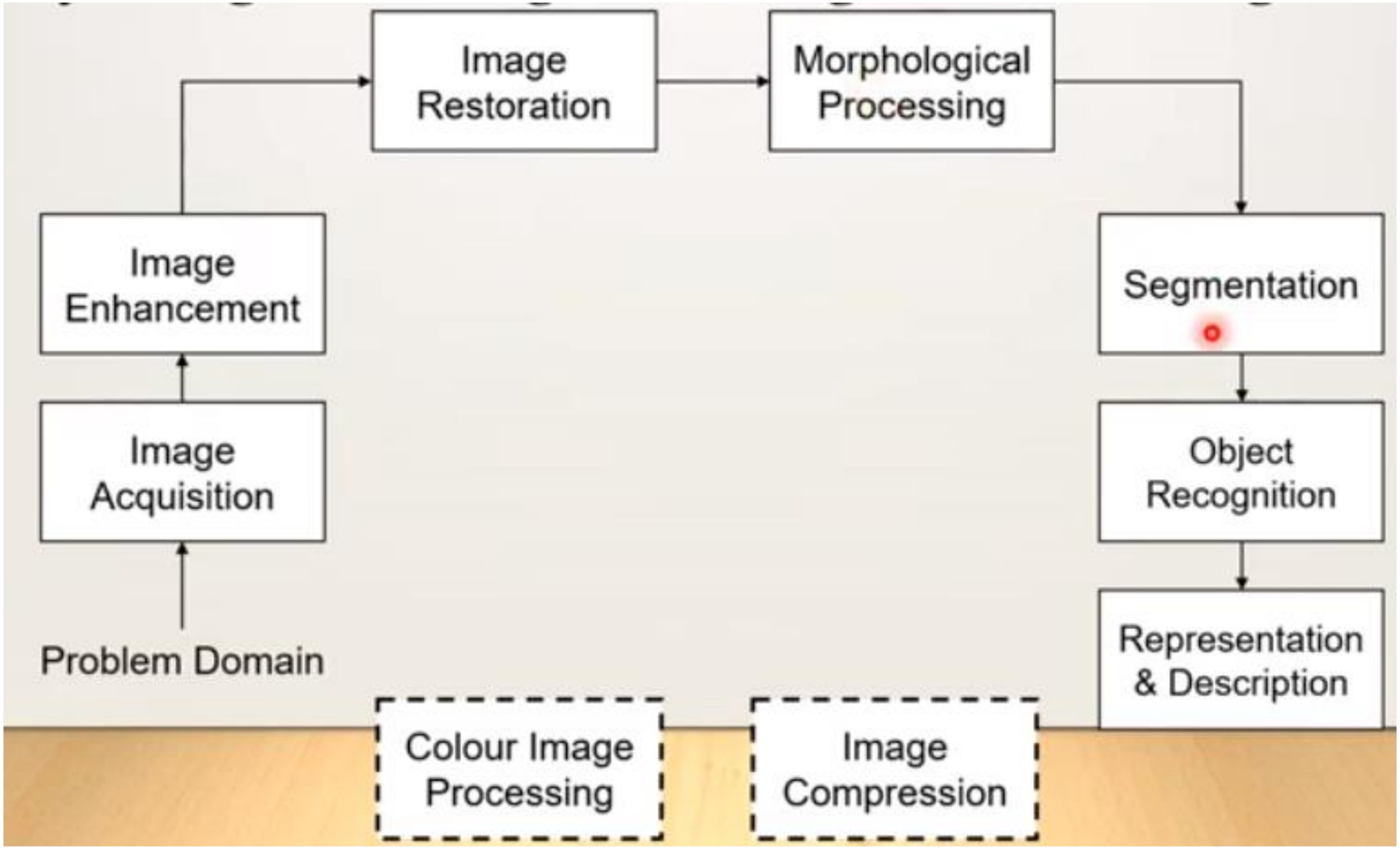
- Image processing techniques are used extensively by law enforcers
  - Number plate recognition for speed cameras/automated toll systems
  - Fingerprint recognition
  - Enhancement of CCTV images

Examples: HCI

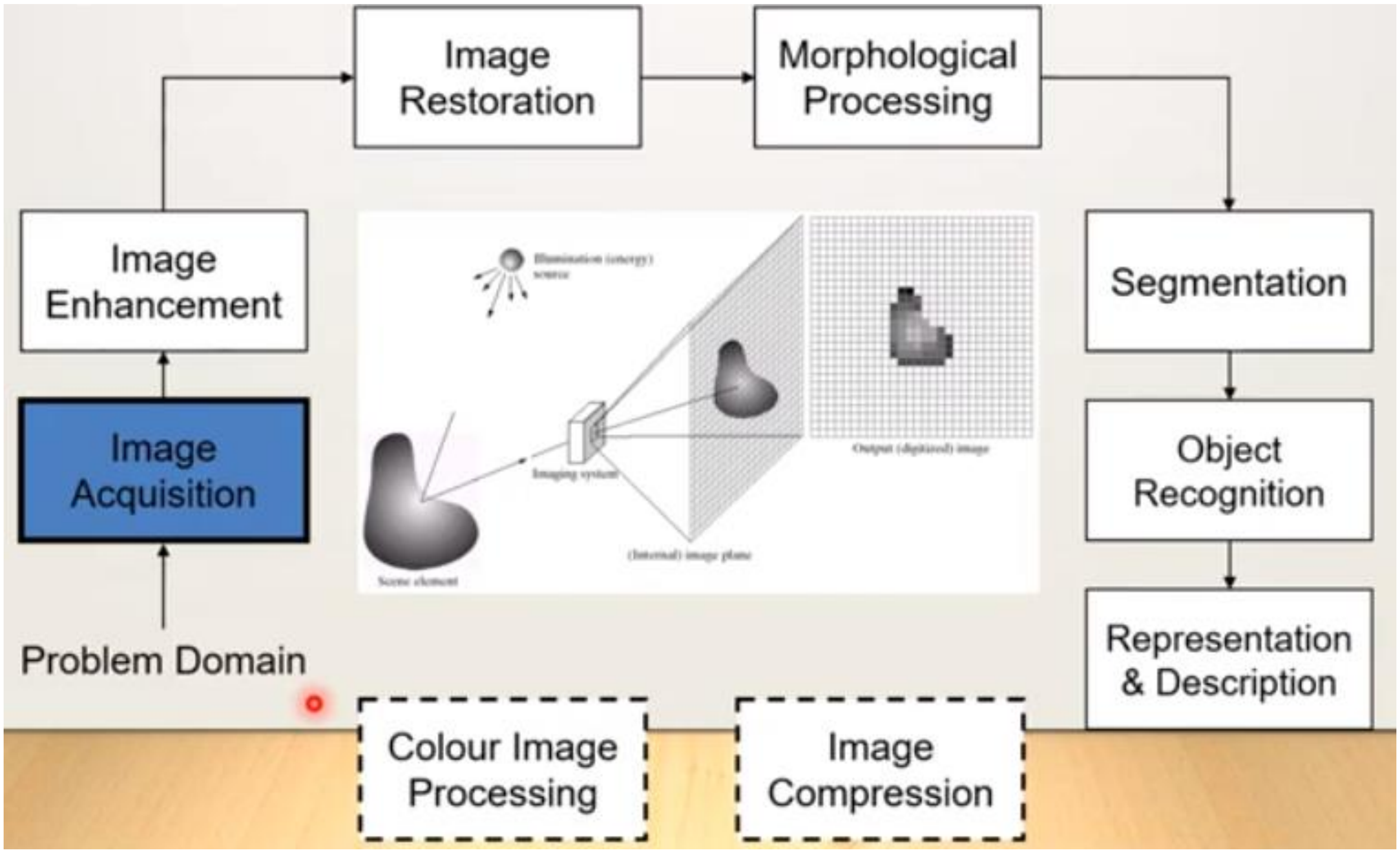
- Try to make human computer interfaces more natural
  - Face recognition
  - Gesture recognition

Key Stages in Digital Image Processing

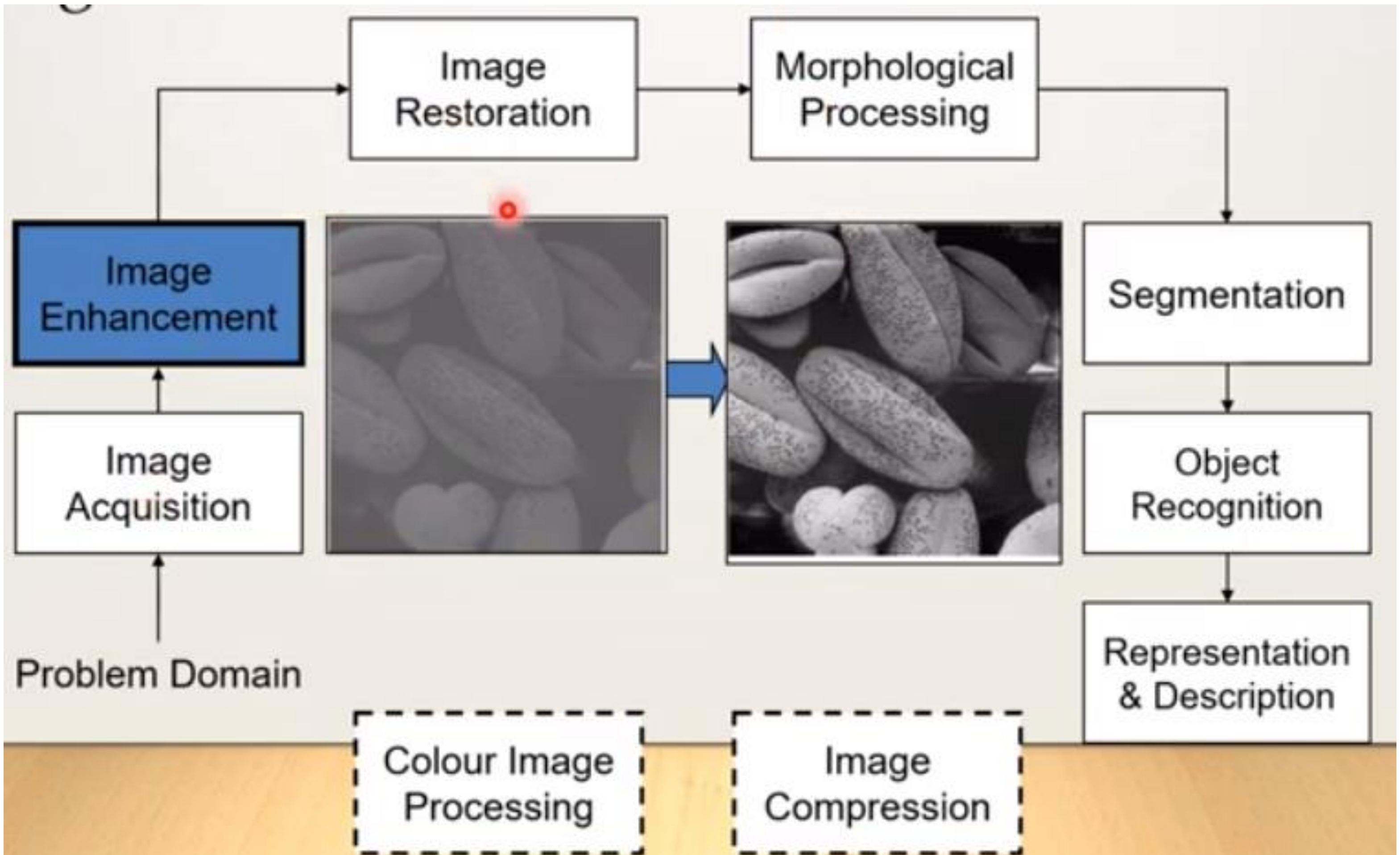
Key Stages in Digital Image Processing



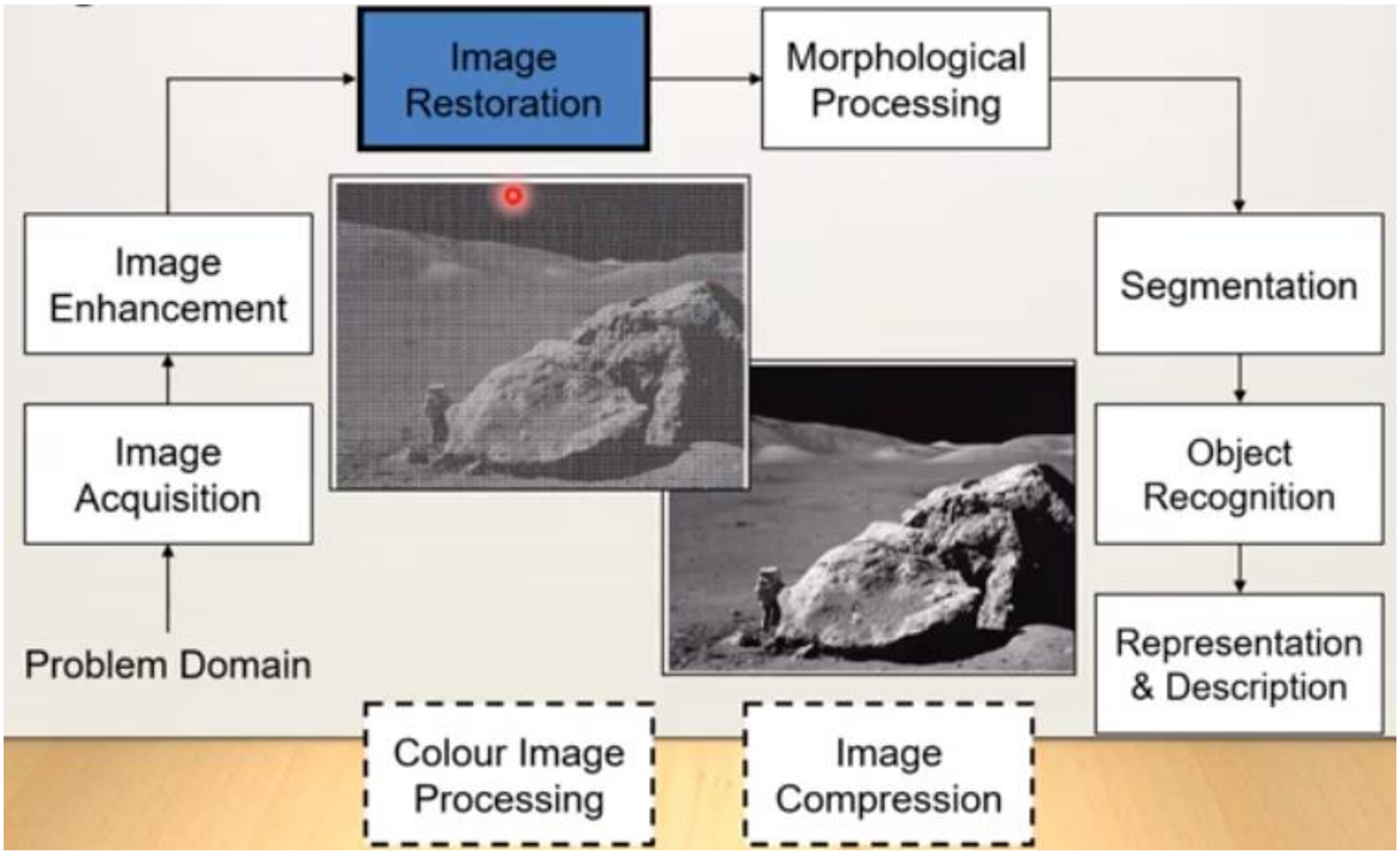
Key Stages in Digital Image Processing: Image Acquisition



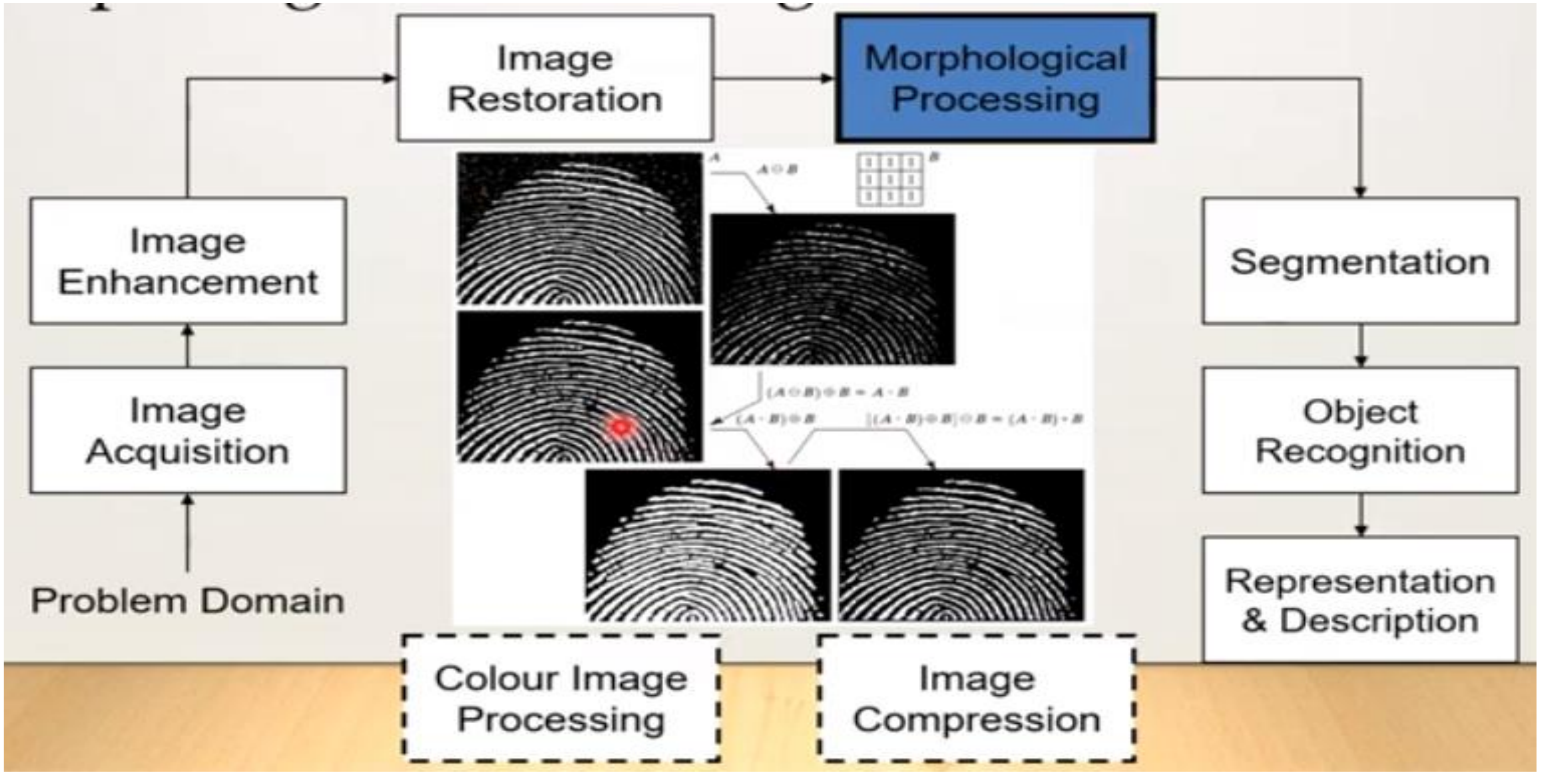
Key Stages in Digital Image Processing: Image Enhancement



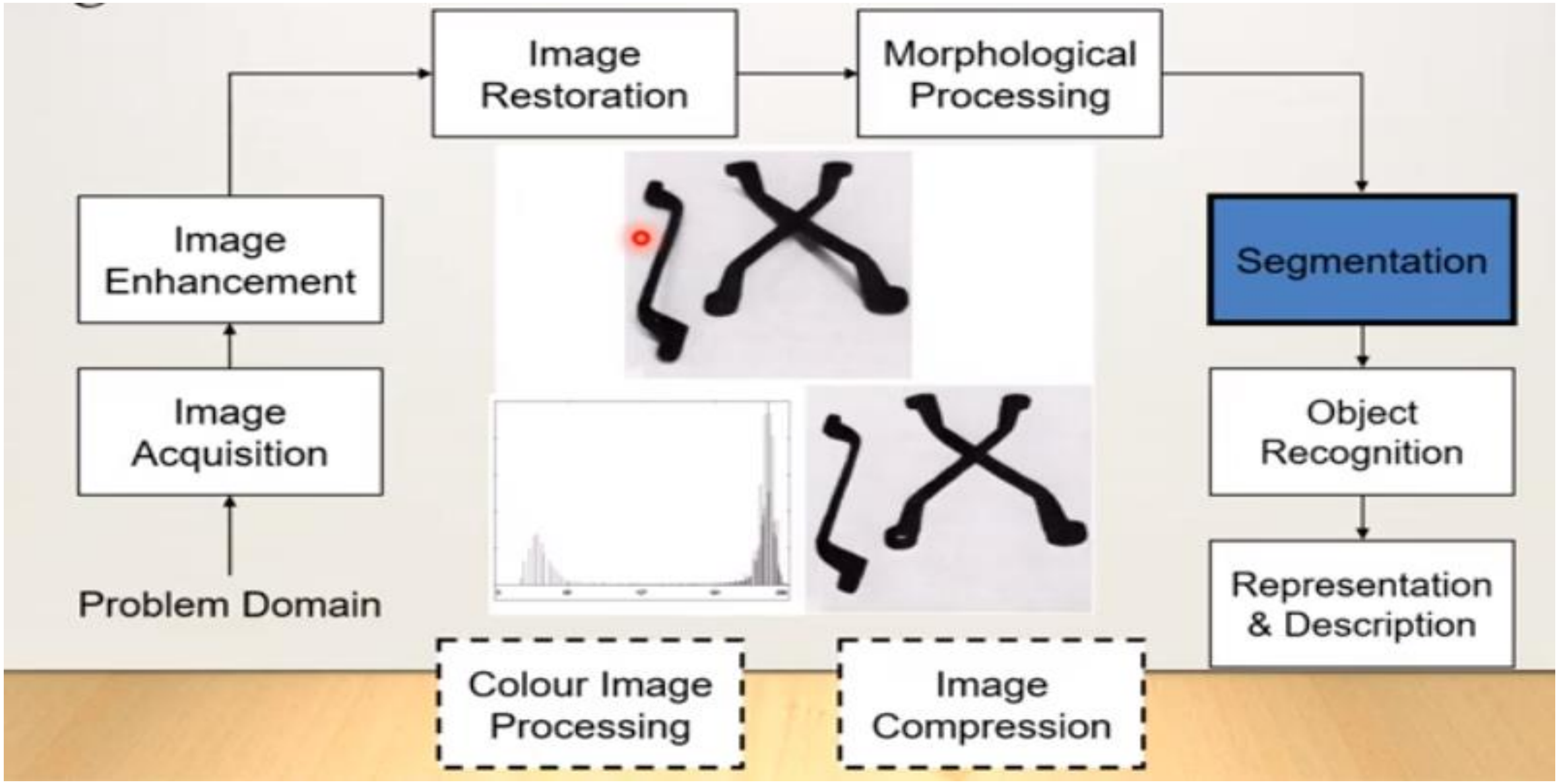
Key Stages in Digital Image Processing: Image Restoration



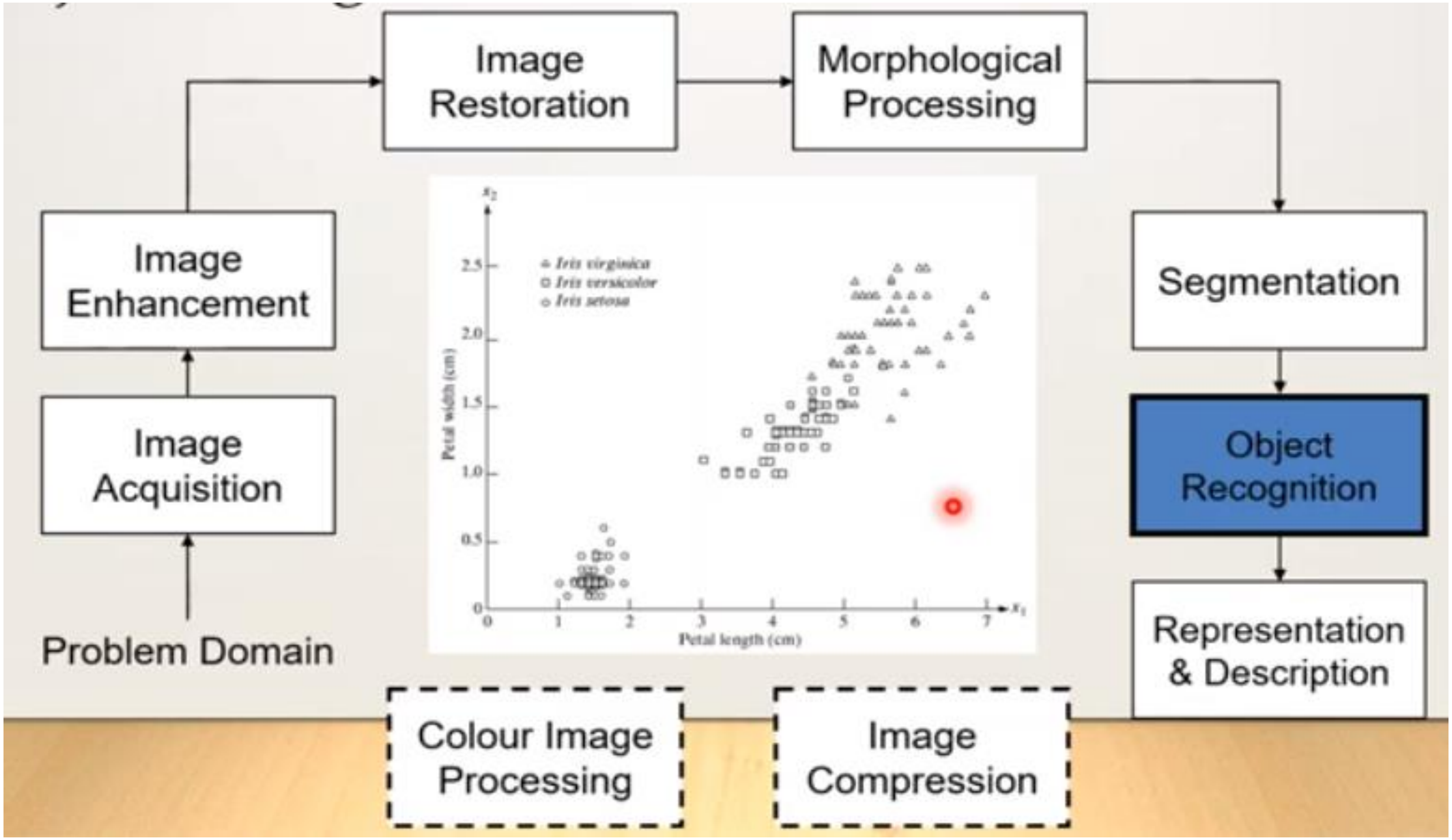
Key Stages in Digital Image Processing: Morphological Processing



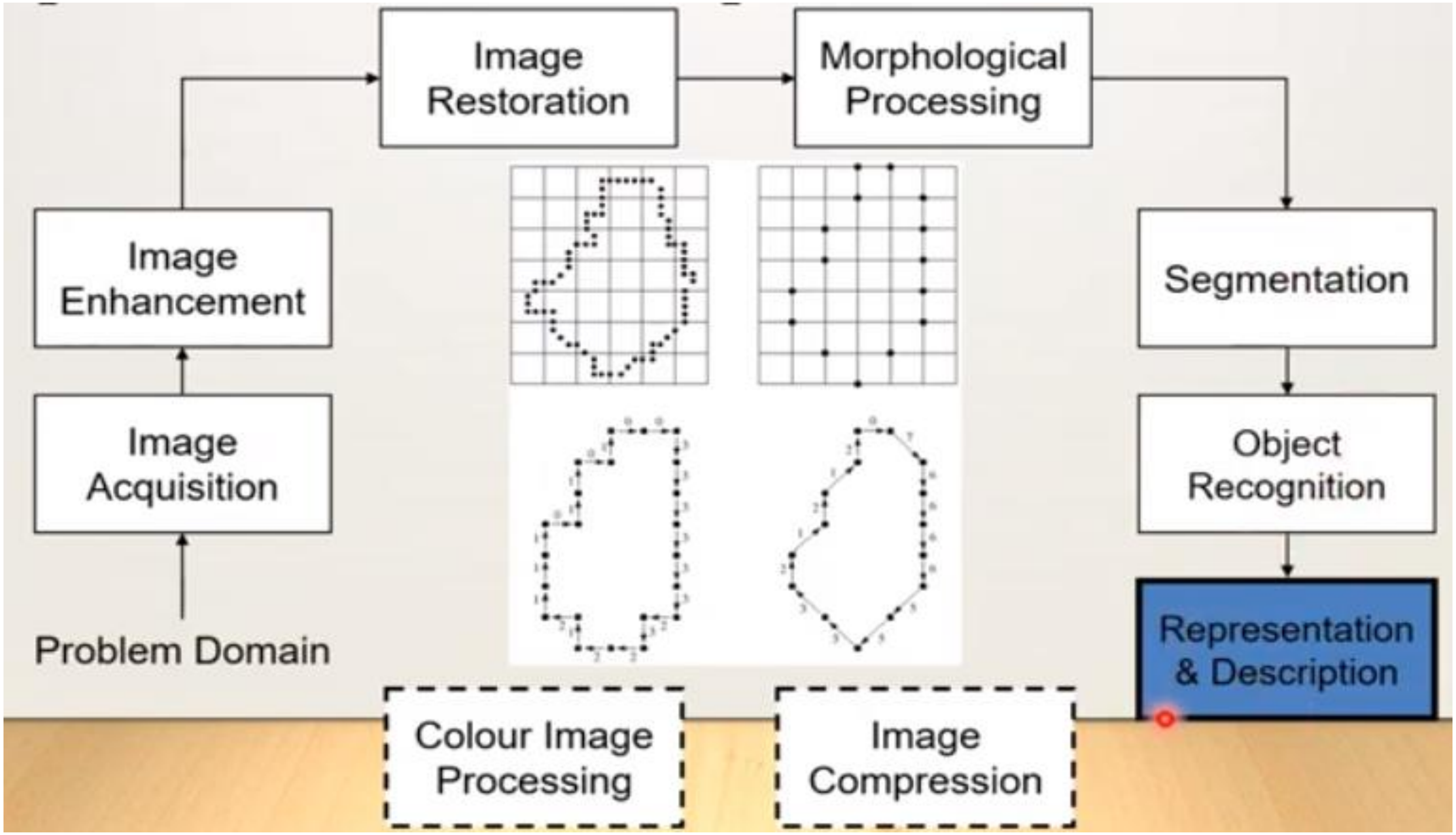
Key Stages in Digital Image Processing: Segmentation



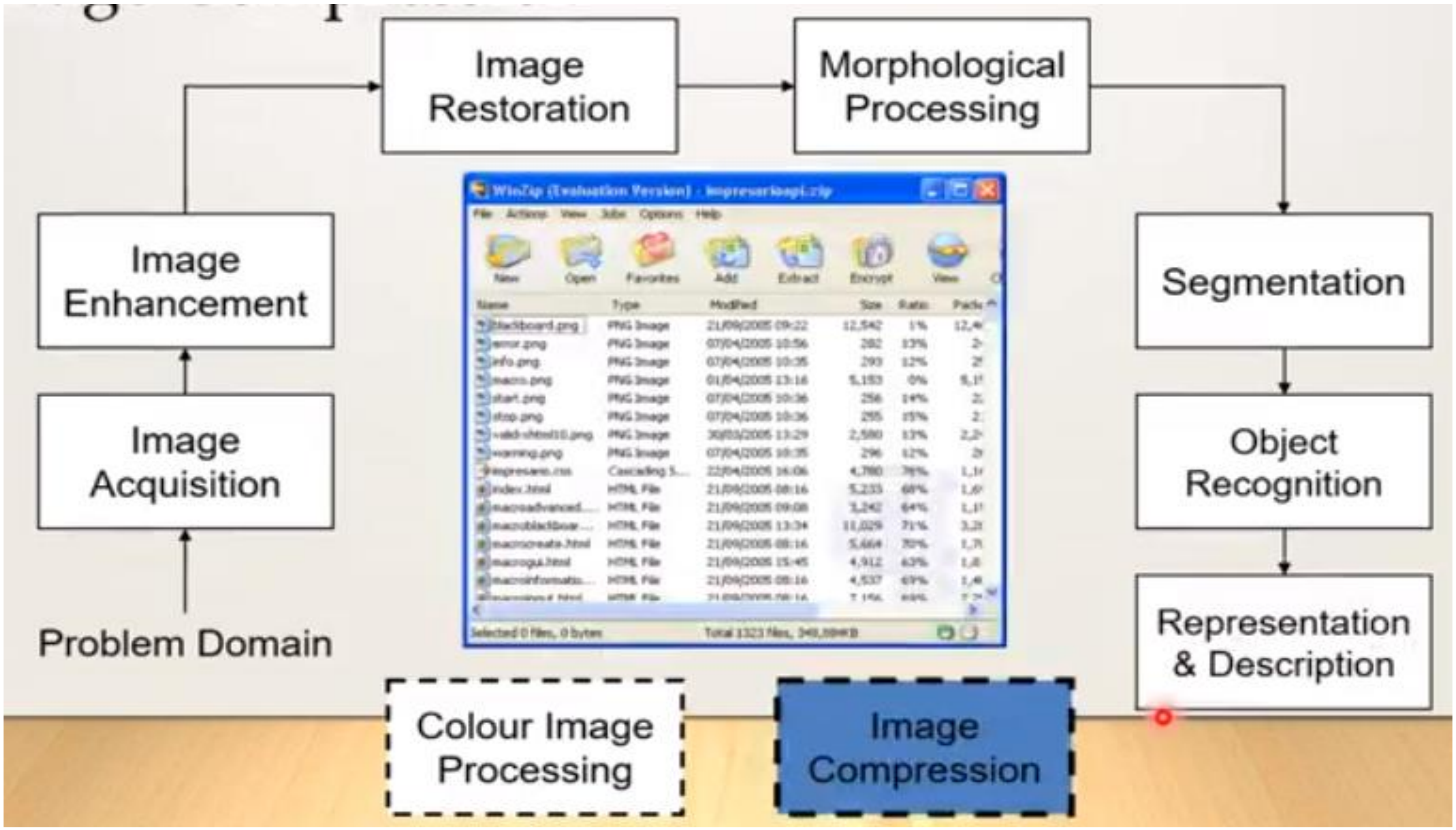
Key Stages in Digital Image Processing: Object Recognition



Key Stages in Digital Image Processing: Representation & Description



Key Stages in Digital Image Processing: Image Compression



Key Stages in Digital Image Processing: Color Image Processing

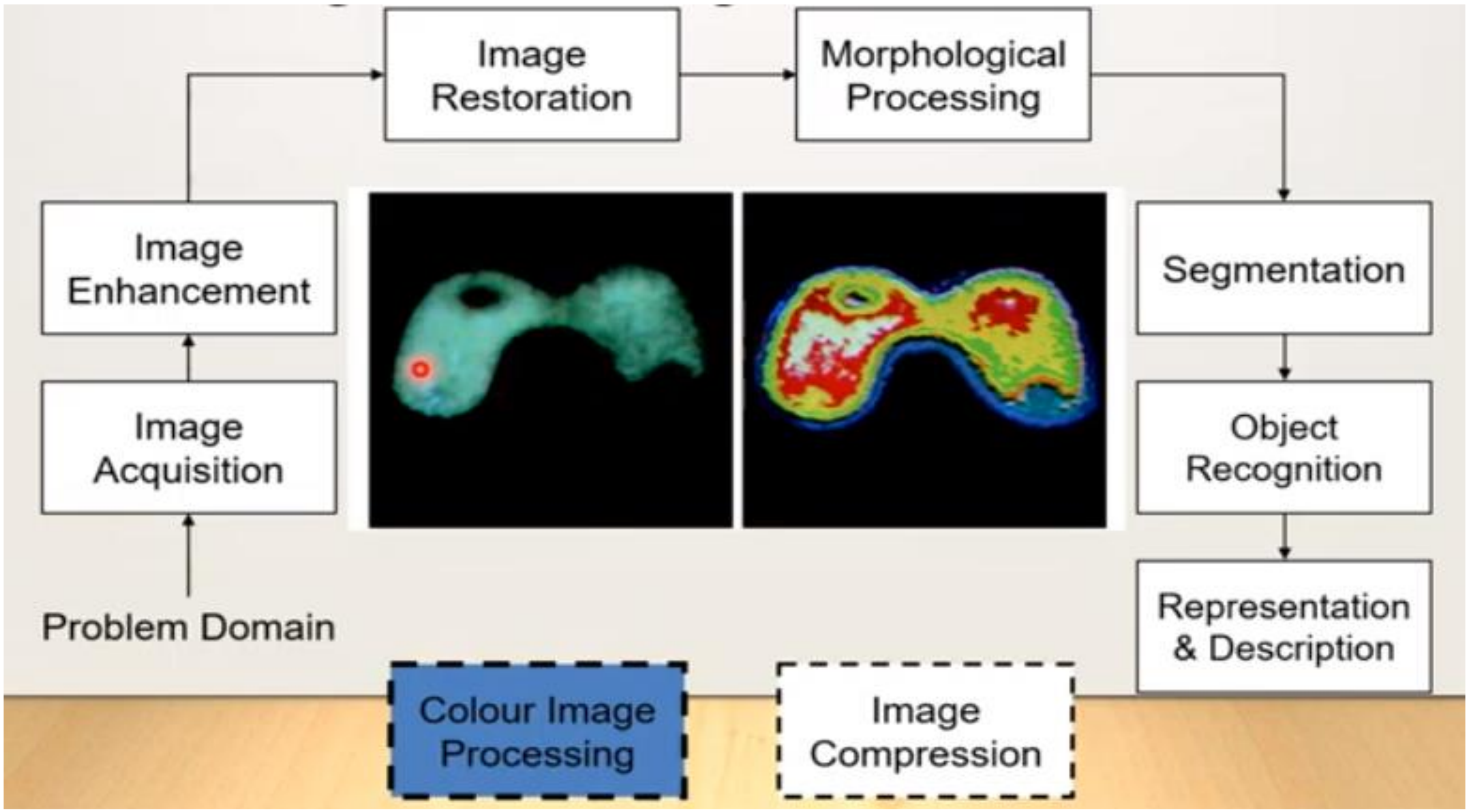


Image Enhancement

What is Image Enhancement?

- Image enhancement is the process of making images more useful
- The reasons for doing this include:
  - Highlighting interesting detail in images
  - Removing noise from images
  - Making image more visually appealing

Image Enhancement Examples

Image Enhancement Examples(Cont…)

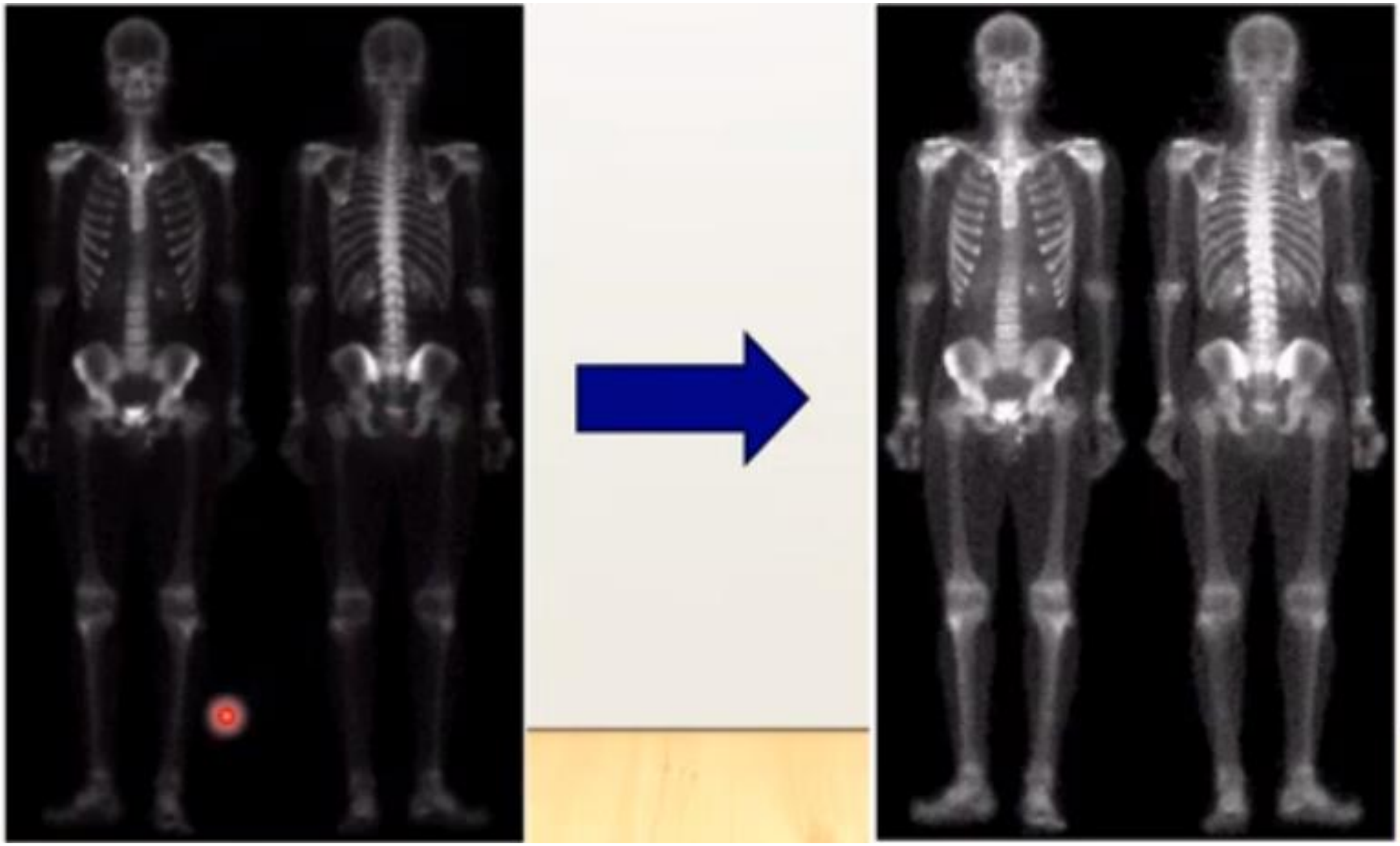


Image Enhancement Examples(Cont…)

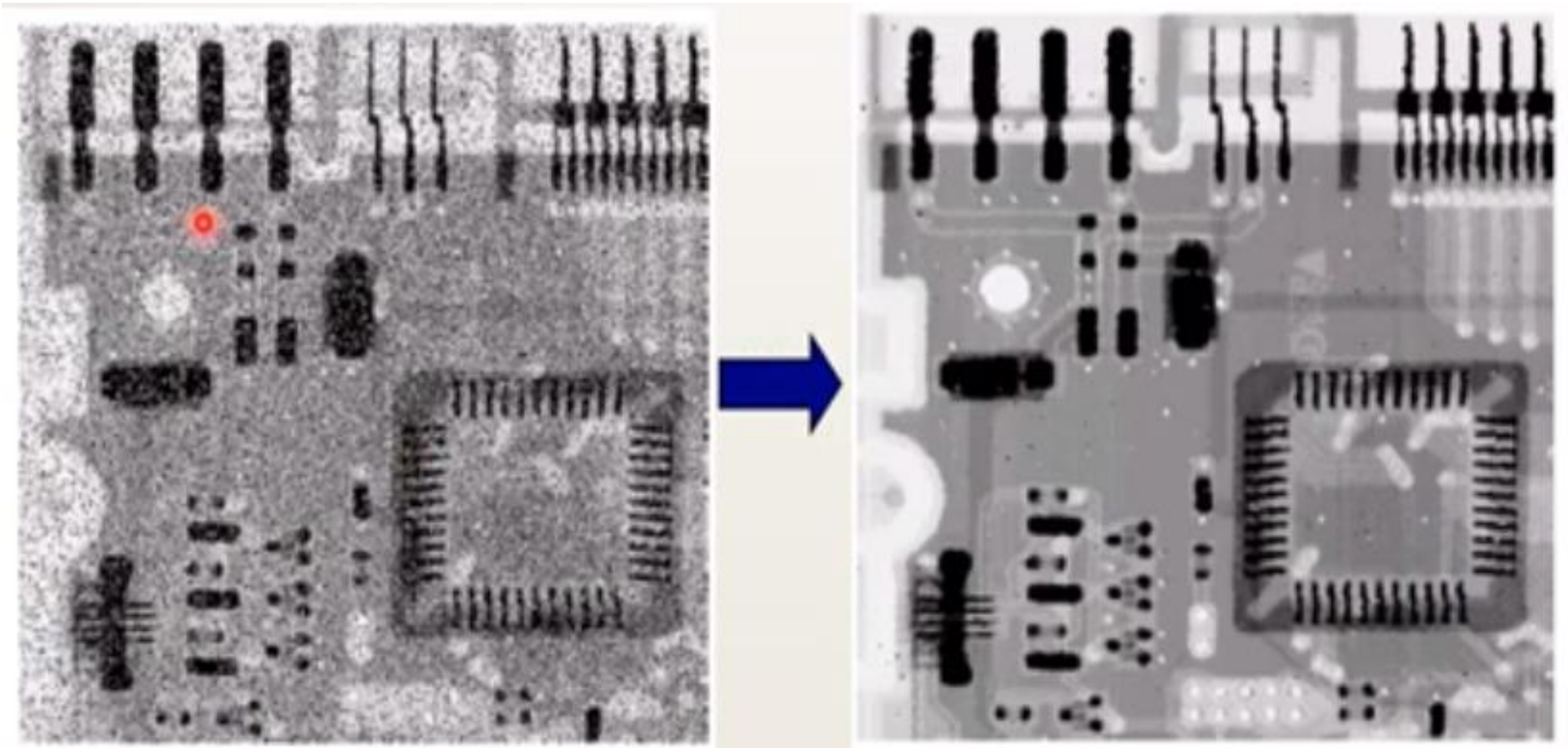


Image Enhancement Examples(Cont…)

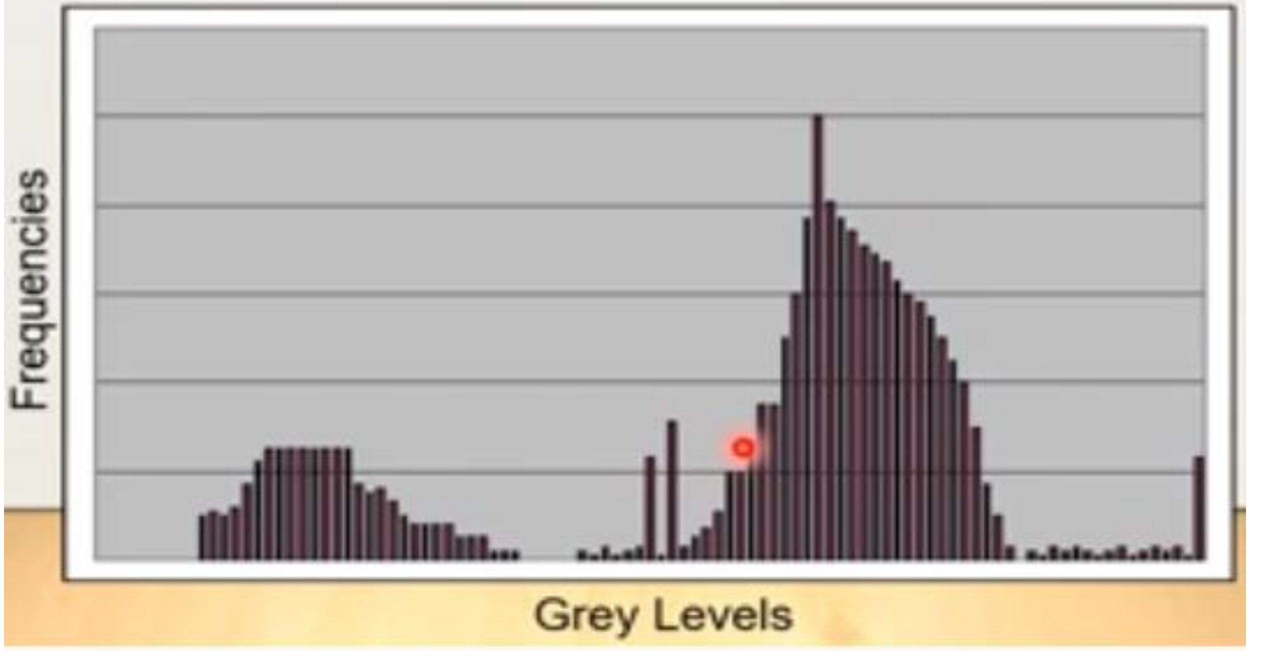
Spatial & Frequency Domains

- There are two broad categories of image enhancement techniques
- Spatial domain techniques
  - Direct manipulation if image pixels
- Frequency domain techniques
  - Manipulation of Fourier transform or wavelet transform of image an image

Image Enhancement
Image Histograms

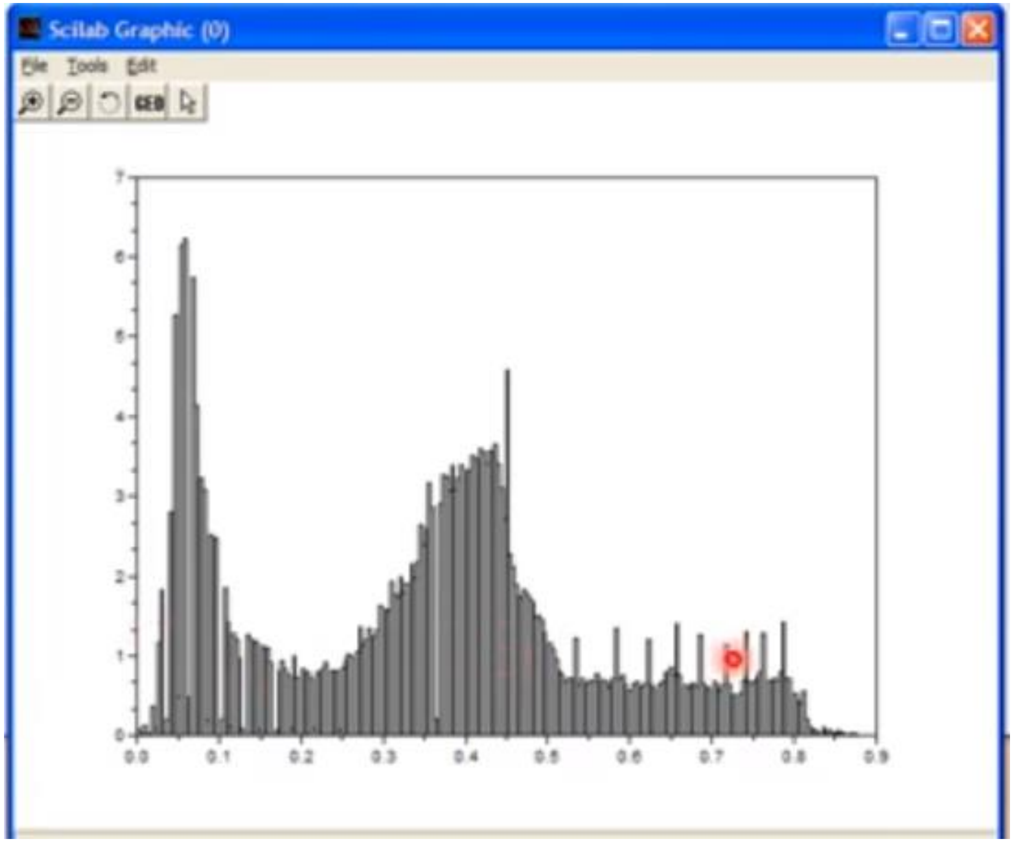
Image Histograms

- The histogram of an image shows us the distribution of grey levels in image
- Massively useful in image processing especially in segmentation.

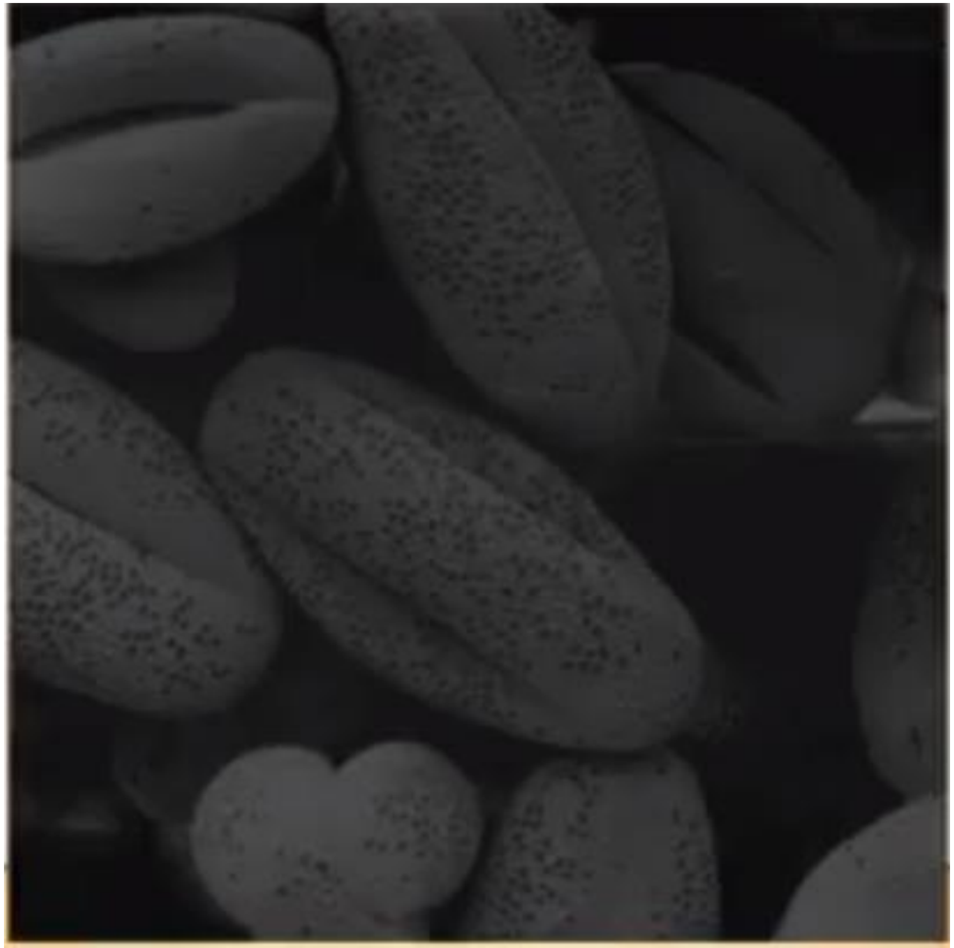


Histogram Examples

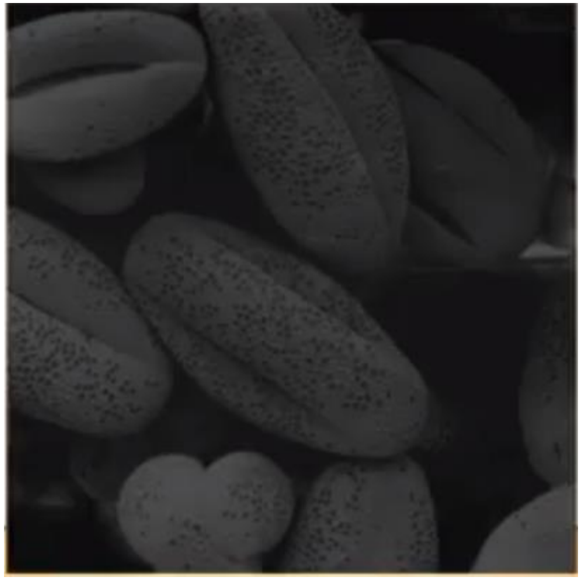
Histogram Examples(Cont…)

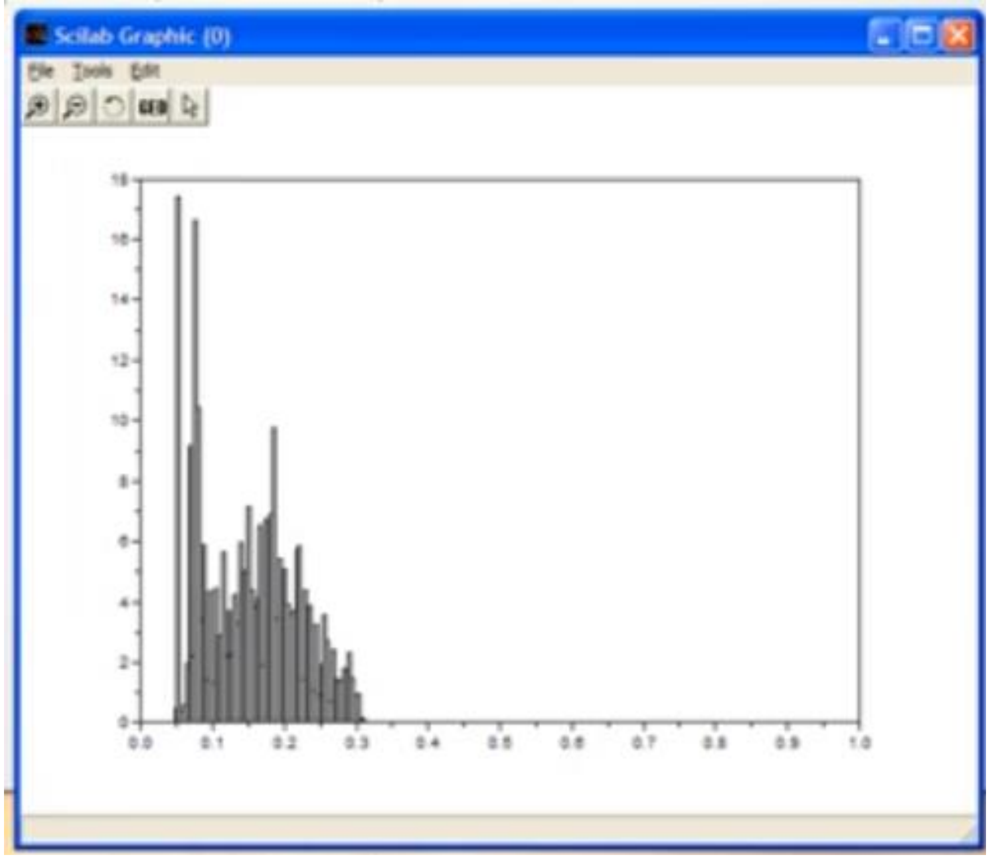


Histogram Examples(Cont…)

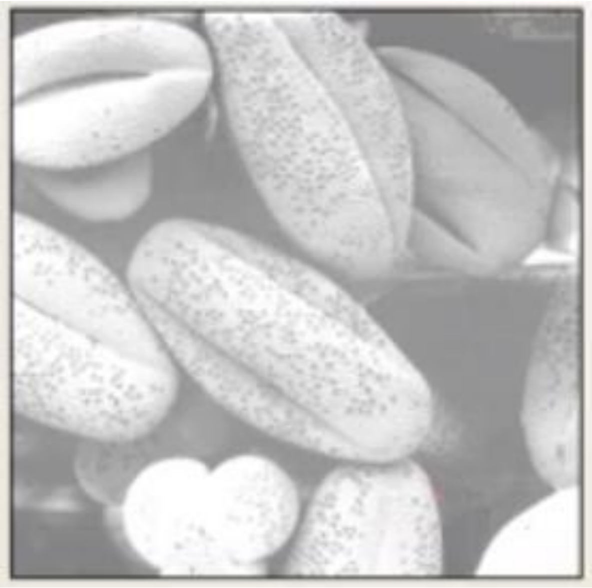


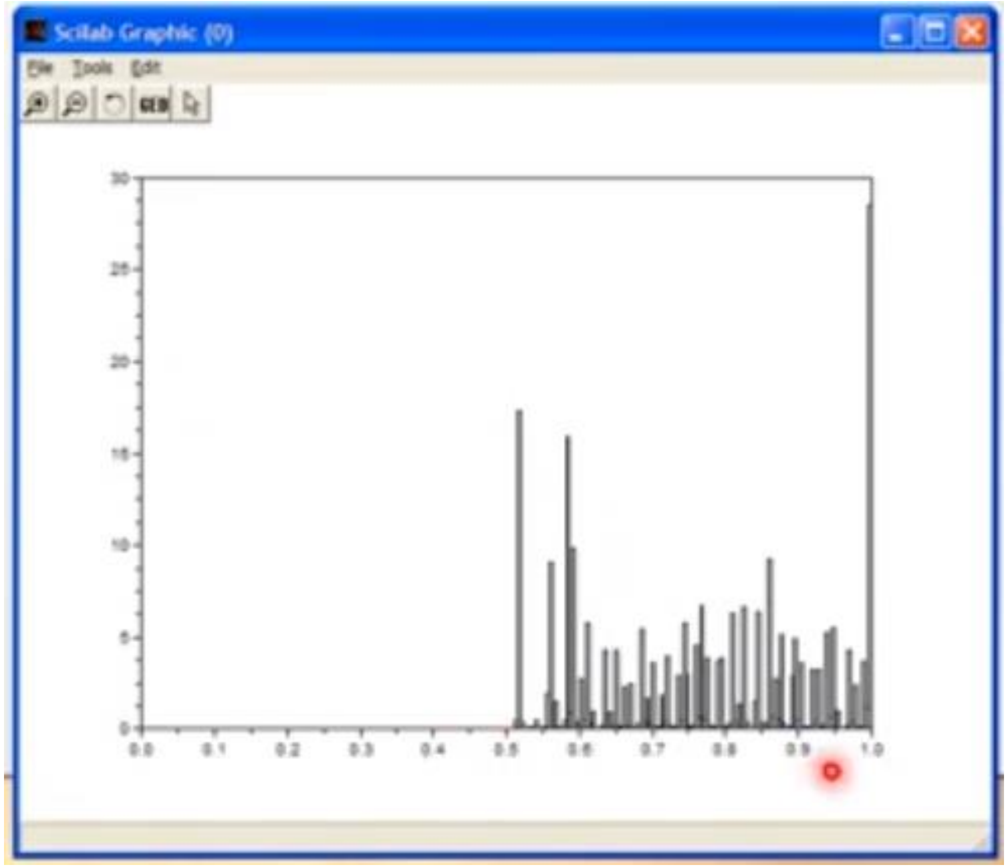
Histogram Examples(Cont…)



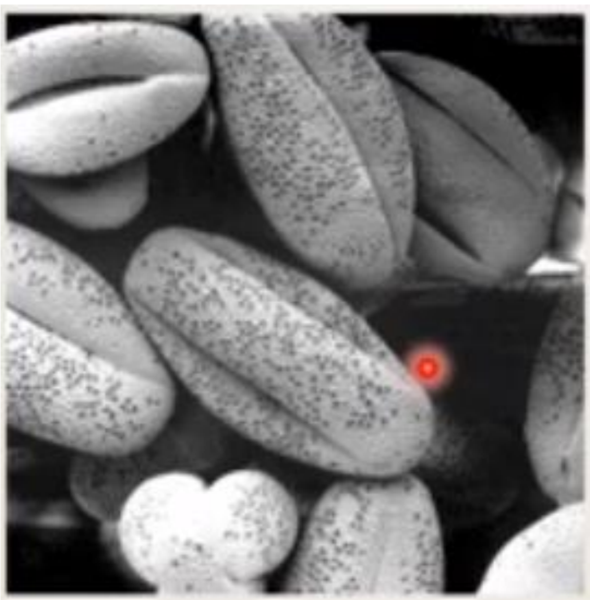


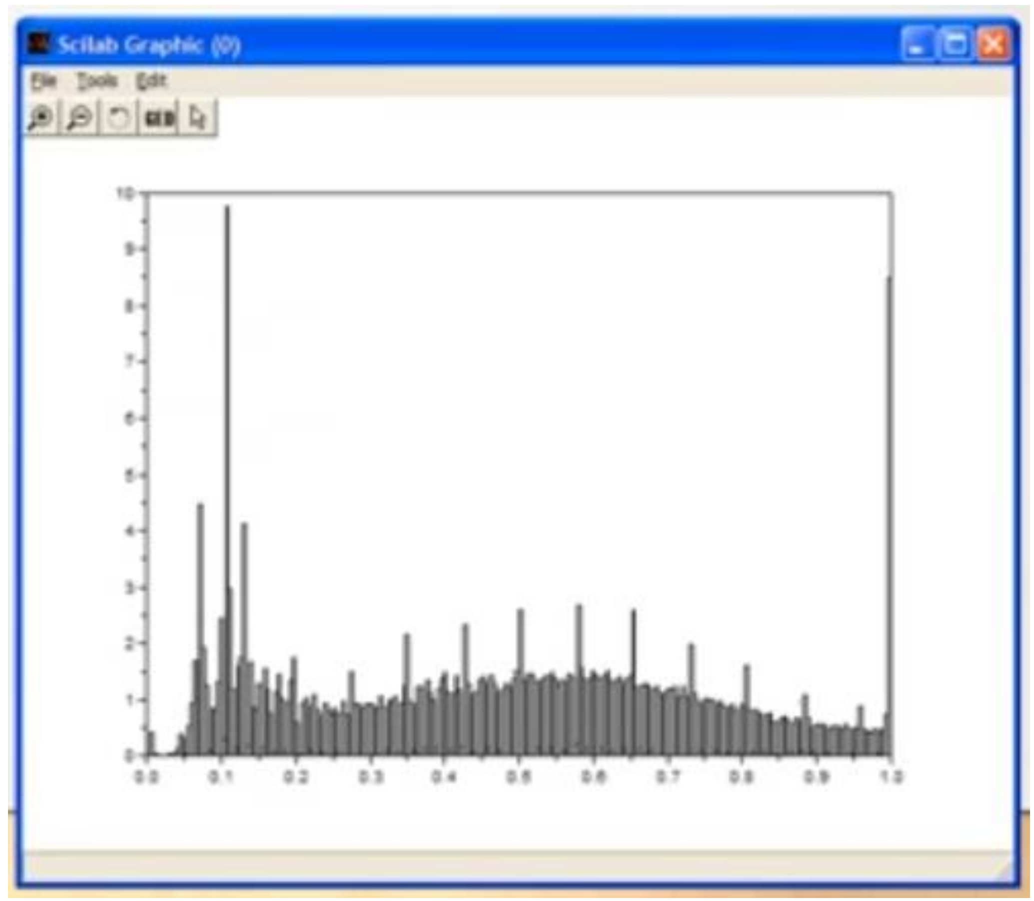
Histogram Examples(Cont…)





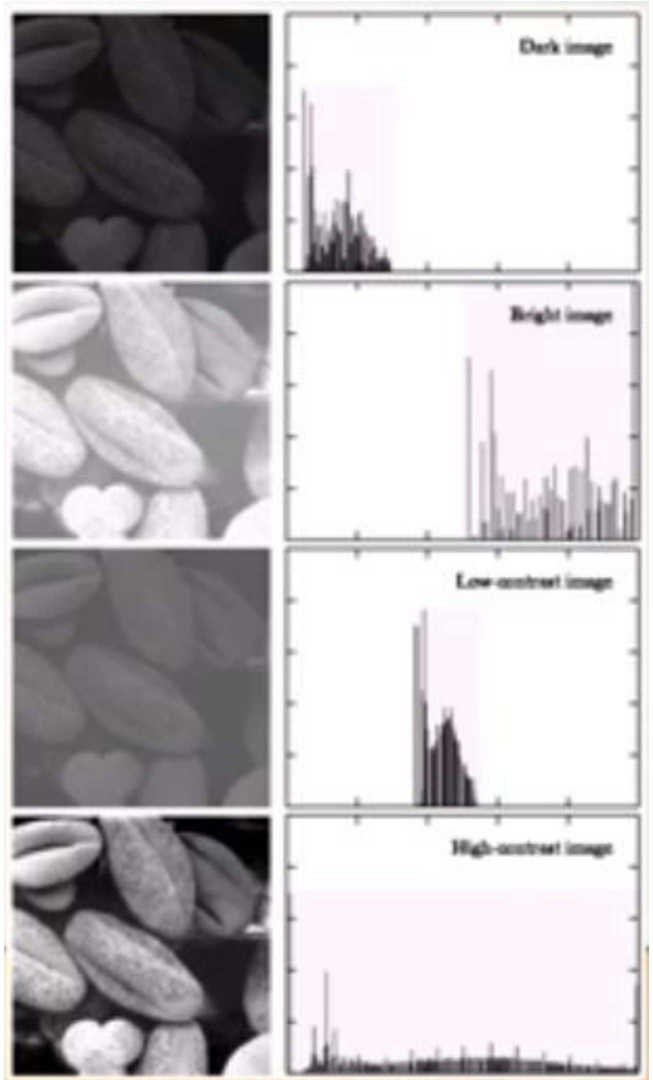
Histogram Examples(Cont…)



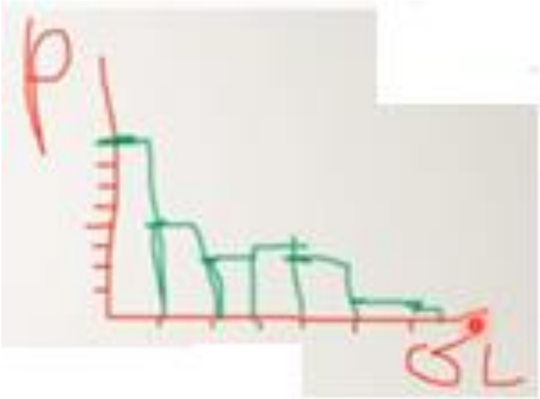


Histogram Examples(Cont…)

- A selection of images and their histogram
- Notice the relationships between the images and their histograms
- Note that the high contrast image has the most evenly spaced histogram



Histogram Examples



- Plot the histogram for following data.

Grey Level	Number of Pixels
0	40
1	20
2	10
3	15
4	10
5	3
6	2

Histogram Examples

- Calculate the probability of occurrence of the grey level(normalized histogram)

Grey Level	Number of Pixels(nk)	P(rk) = nk/n
0	40	0.4
1	20	0.2
2	10	0.1
3	15	0.15
4	10	0.1
5	3	0.03
6	2	0.02
	n=100	

Image Enhancement
Histograms Stretching

Histogram Examples

- Calculate the probability of occurrence of the grey level(Probability Density Functions).

Grey Level	Number of Pixels(nk)	P(rk) = nk/n
0	40	
1	20	
2	10	
3	15	
4	10	
5	3	
6	2	
	n=100	

Linear Stretching

- One way to increase the dynamic range is by using a technique known as histogram stretching.
- In this method, we do not alter the basic shape of the histogram, but we spread it so as to cover the entire dynamic range.

$$s = T(r) = \frac{S_{max} - S_{min}}{R_{max} - R_{min}} (R - R_{min}) + S_{min}$$

- Smax -> Maximum grey level of output image
- Smin -> Minimum grey level of output image
- rmax -> Maximum grey level of input image
- rmin -> Minimum grey level of input image

Histogram Stretching : Examples

- Perform histogram stretching so that new image has a dynamic range of [0,7].

Grey Level	Number of Pixels
0	0
1	0
2	50
3	60
4	50
5	20
6	10
7	0

Histogram Stretching : Examples

$R_{min} = 2$ $\quad S_{min} = 0$ $\quad r = [2, 6]$

$R_{max} = 6$ $\quad S_{max} = \cancel{8}\,7$

$$s = \frac{s_{max} - s_{min}}{r_{max} - r_{min}} (r - r_{min}) + s_{min}$$

Histogram Stretching : Examples

Put the value of r,

$$s = \frac{7-0}{6-2}(r-2) + 0$$

$$s = \frac{7}{4}(r-2) \qquad \Rightarrow r = 2 : \; s = \frac{7}{4}(2-2) = 0$$

$$\Rightarrow r = 3 : \; s = \frac{7}{4}(3-2) = \frac{7}{4} = 1.75 \approx 2$$

$$\Rightarrow r = 4 : \; s = \frac{7}{4}(4-2) = \frac{7}{4} \times 2 = 3.5 \approx 4$$

$$\Rightarrow r = 5 : \; s = \frac{7}{4}(5-2) = \frac{7}{4} \times 3 = 5.25 \approx 5$$

$$\Rightarrow r = 6 : \; s = \frac{7}{4}(6-2) = \frac{7}{4} \times 4 = 7$$

Here we find the all slope value at different range.

Histogram Stretching : Examples

Gr. L	Old pixel	New pixel
0	0	50
1	0	0
2	50	60
3	60	0
4	50	50
5	20	20
6	10	0
7	0	10

$S = \{0, 2, 4, 5, 7\}$

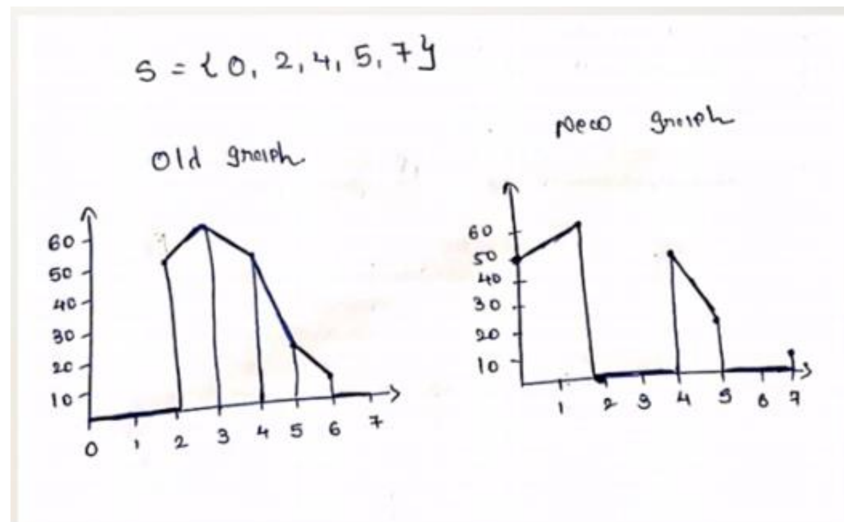


Image Enhancement
Histogram Equalization

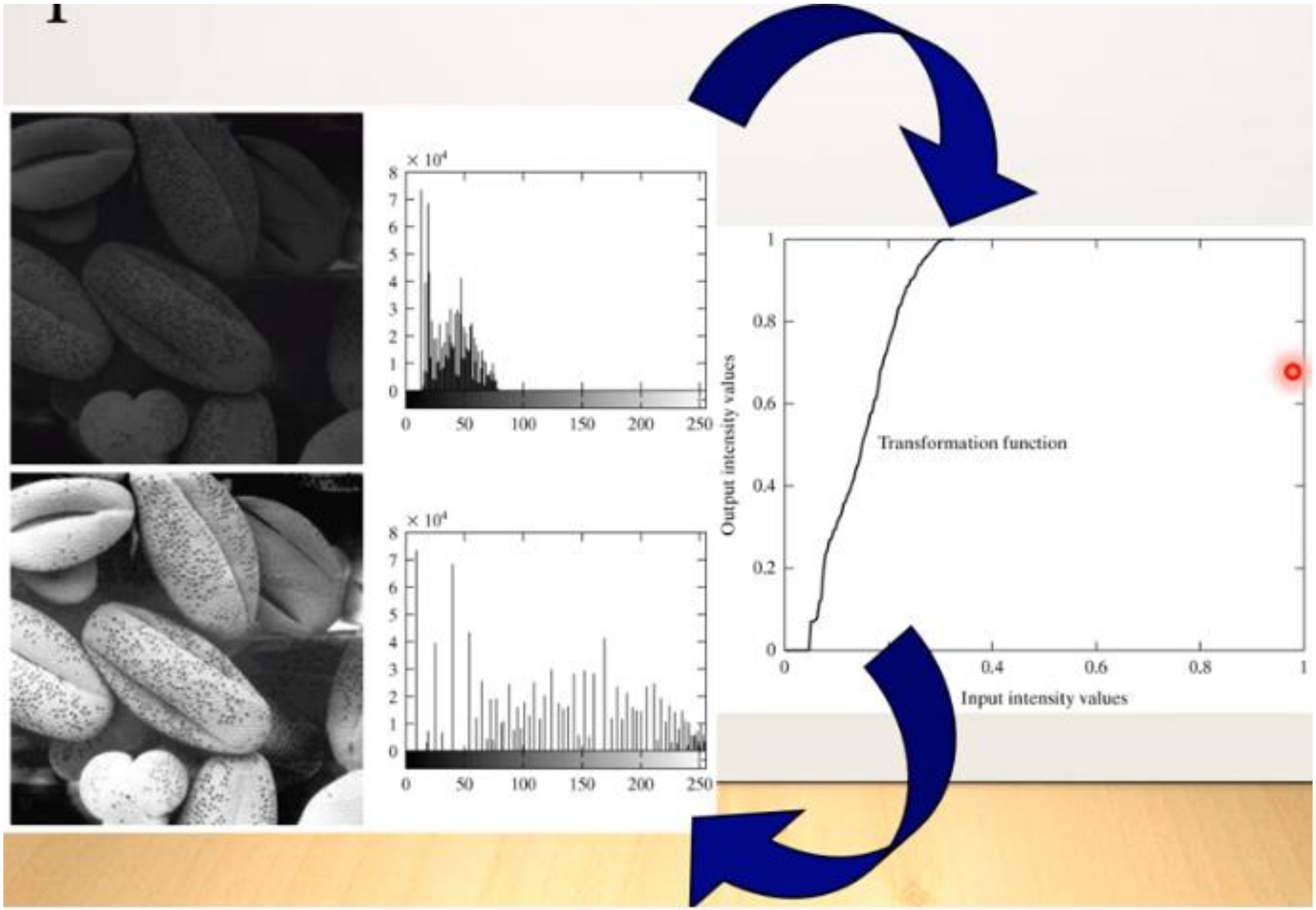
Histogram Equalisation

- Spreading out the frequencies in an image(or equalizing the image) is a simple way to improve dark or washed out images
- The formula for histogram Equalisation is given where

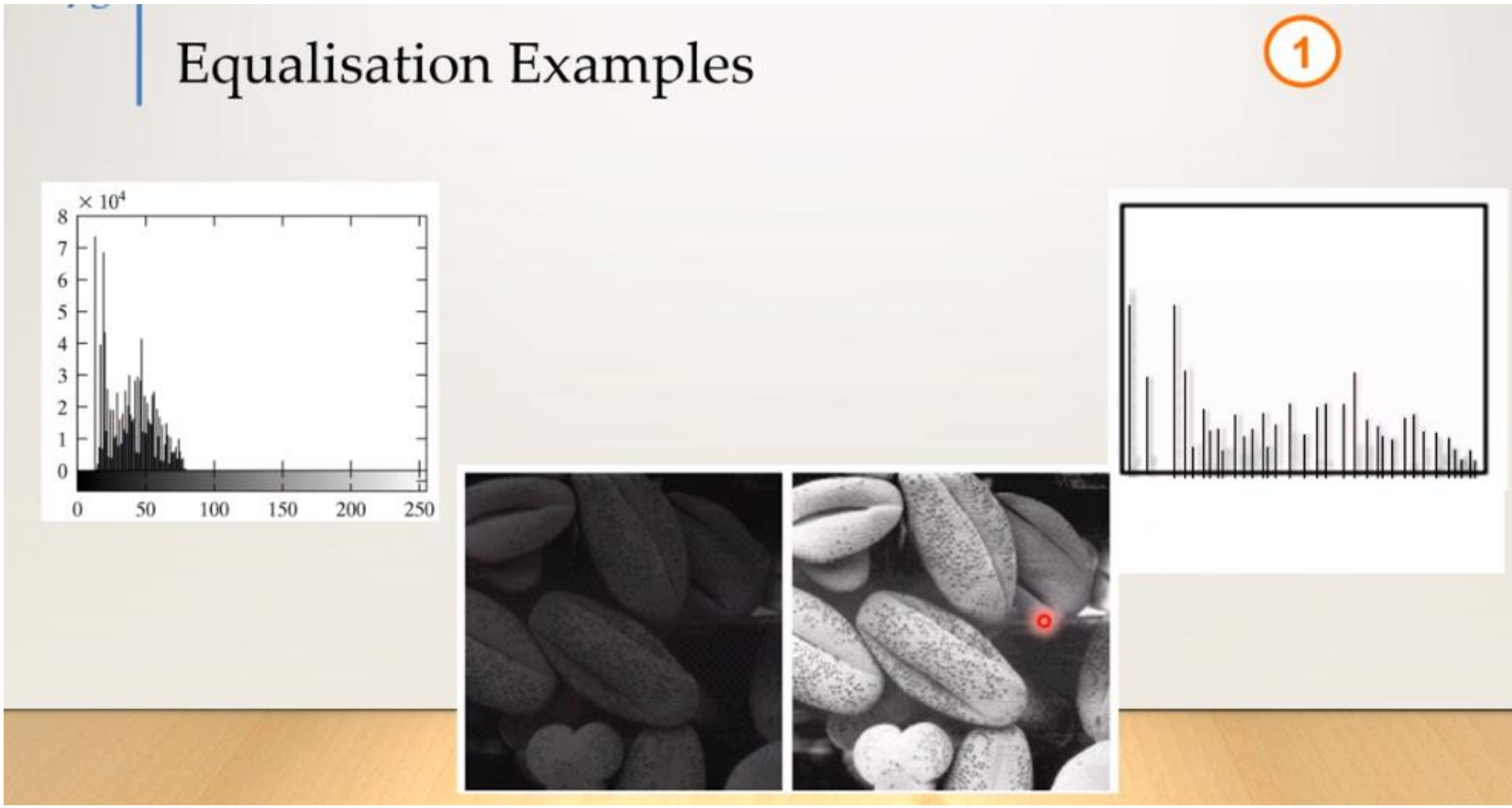
$$
\begin{aligned}
s_k &= T(r_k) \\
&= \sum_{j=1}^{k} p_r(r_j) \\
&= \sum_{j=1}^{k} \frac{n_j}{n}
\end{aligned}
$$

- r_k: input intensity
- s_k: processed intensity
- k: the intensity range (e.g 0.0 – 1.0)
- n_j: the frequency of intensity j
- n: the sum of all frequencies

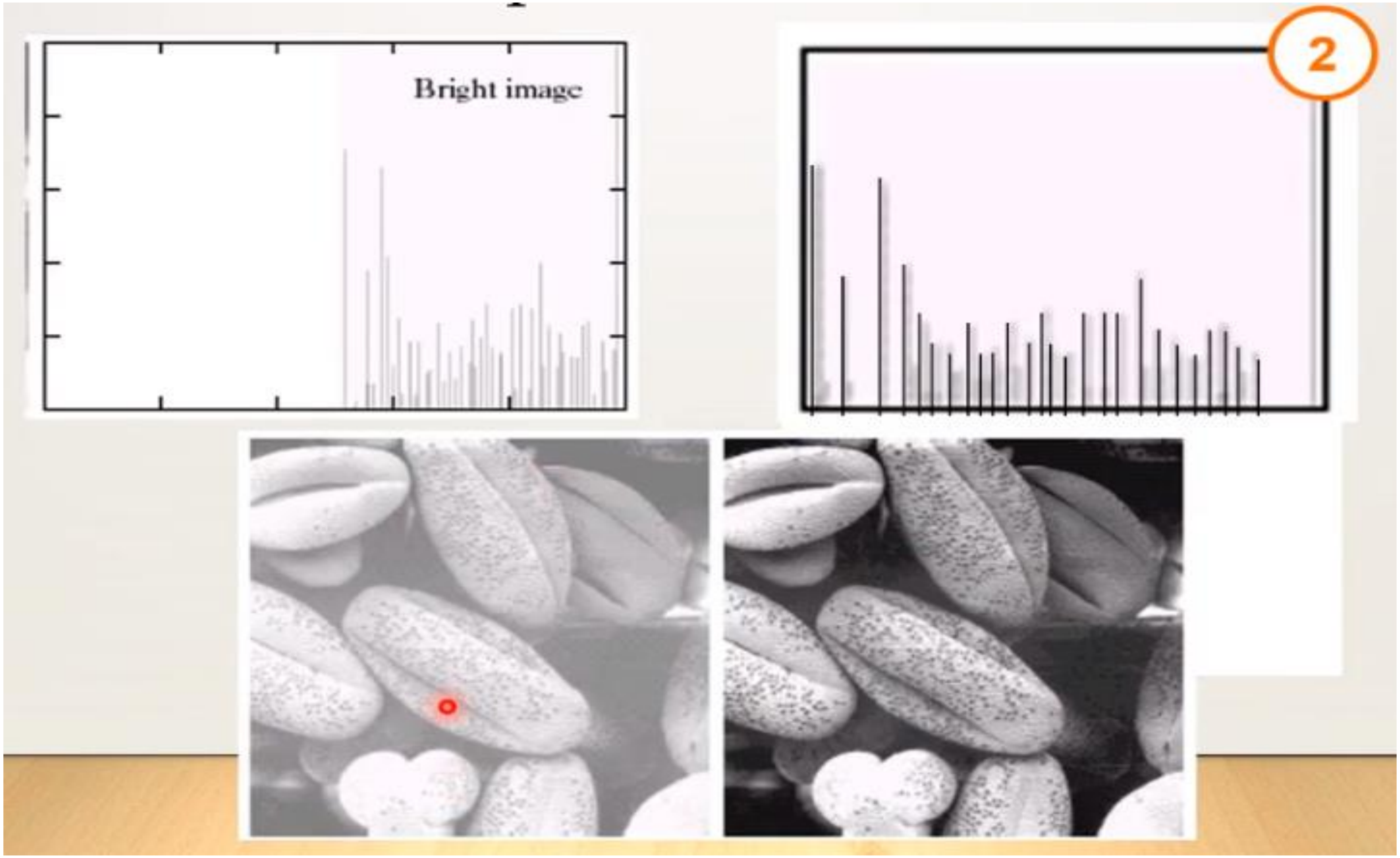
Equalisation Transformation Function



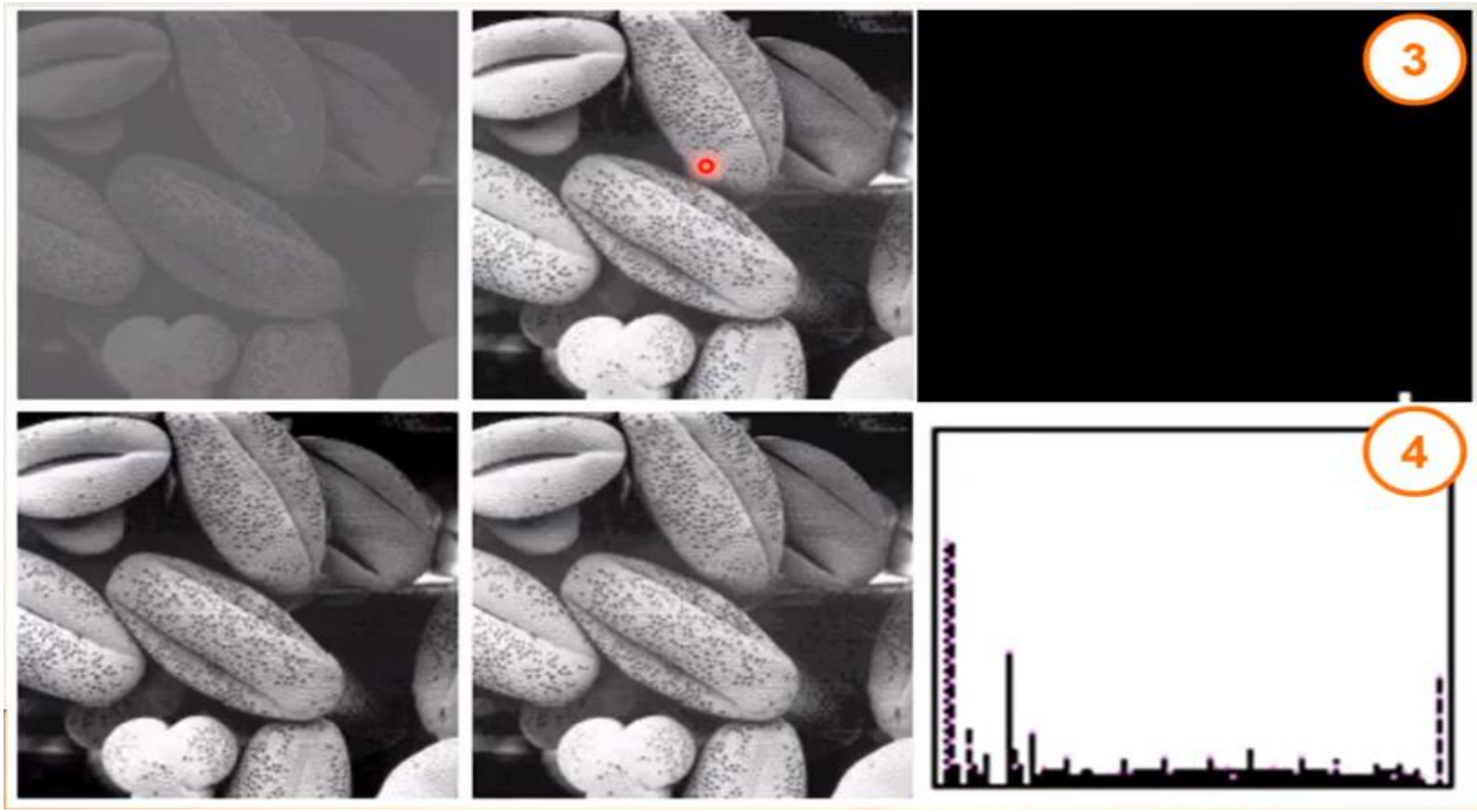
Equalisation Examples



Equalisation Examples



Equalisation Examples



Histogram Equalisation

- Histogram--->PDF --- > CDF --- > Equalized histogram
- CDF – cumulative density function

$$S_k = T(r_k) = \sum_{0}^{r} p_r(r)$$

Equalize the given histogram

Grey Level	Number of Pixels(nk)	PDF P(rk) = nk/n	CDF	(L-1)*Sk	Round off
0	790				
1	1023				
2	850				
3	656				
4	329				
5	245				
6	122				
7	81				
	4096				

Equalize the given histogram

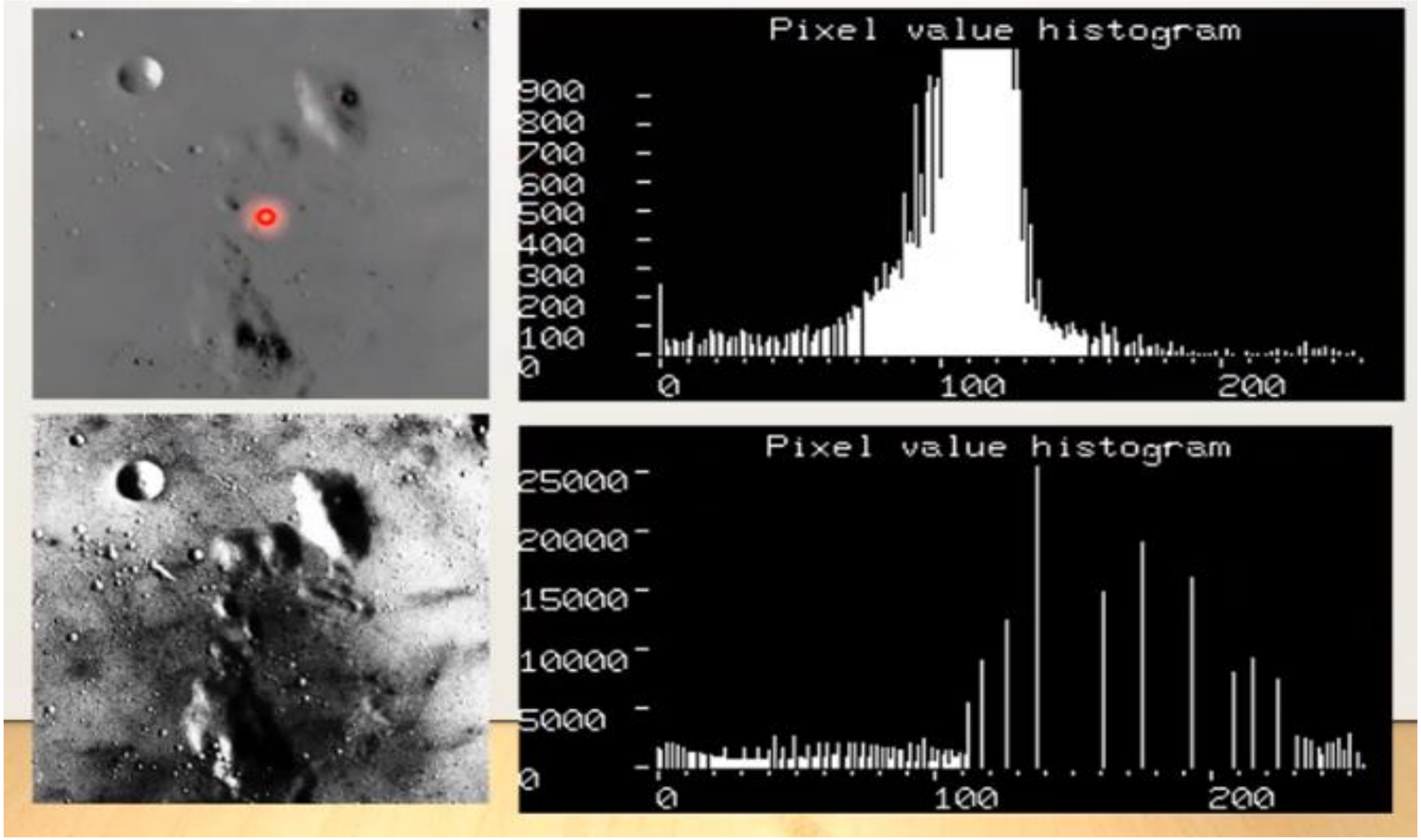
Grey Level	Number of Pixels(nk)	PDF P(rk) = nk/n	CDF Sk = $\sum_{0}^{r} p_r(r)$	(L-1)*Sk	Round off
0	790	0.19	0.19	1.33	1
1	1023	0.25	0.44	3.08	3
2	850	0.21	0.65	4.55	5
3	656	0.16	0.81	5.67	6
4	329	0.08	0.89	6.23	6
5	245	0.06	0.95	6.65	7
6	122	0.03	0.98	6.86	7
7	81	0.02	1	7	7
	4096				

Equalized Grey Level

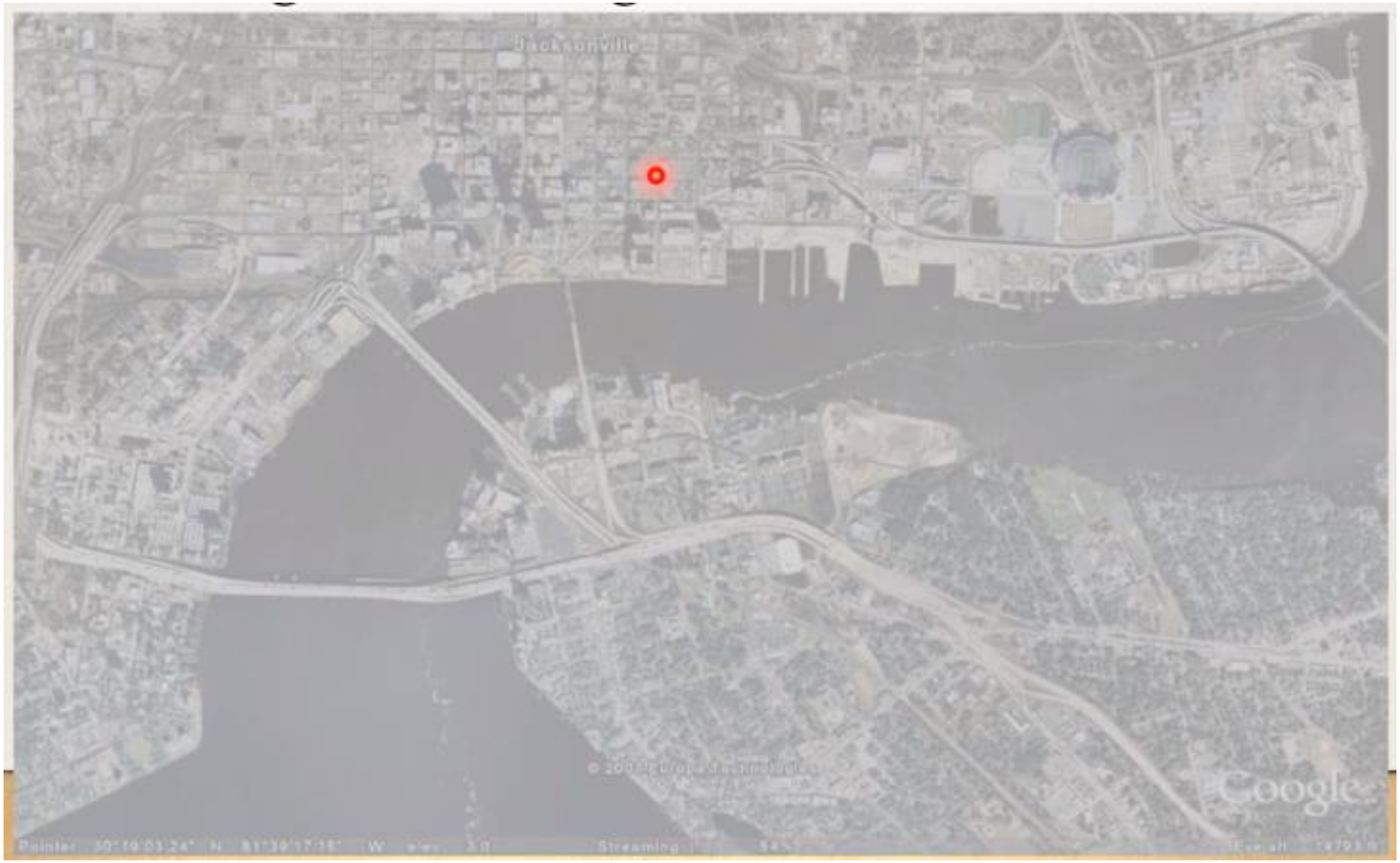
Grey Level	Round off
0	1
1	3
2	5
3	6
4	6
5	7
6	7
7	7

Grey Level	Old Number of Pixels	New Number of Pixels
0	790	0
1	1023	790
2	850	0
3	656	1023
4	329	0
5	245	850
6	122	656+329 = 985
7	81	245+122+81 = 448

Histogram Equalization



Satellite Original Image



Histogram Equalized Image

Noise Removal and Blurring

What is Image Restoration?

- Image restoration attempts to restore images that have been degraded
  - Identify the degradation process and attempts to reverse it.
  - Similar to image enhancement, but more objective.

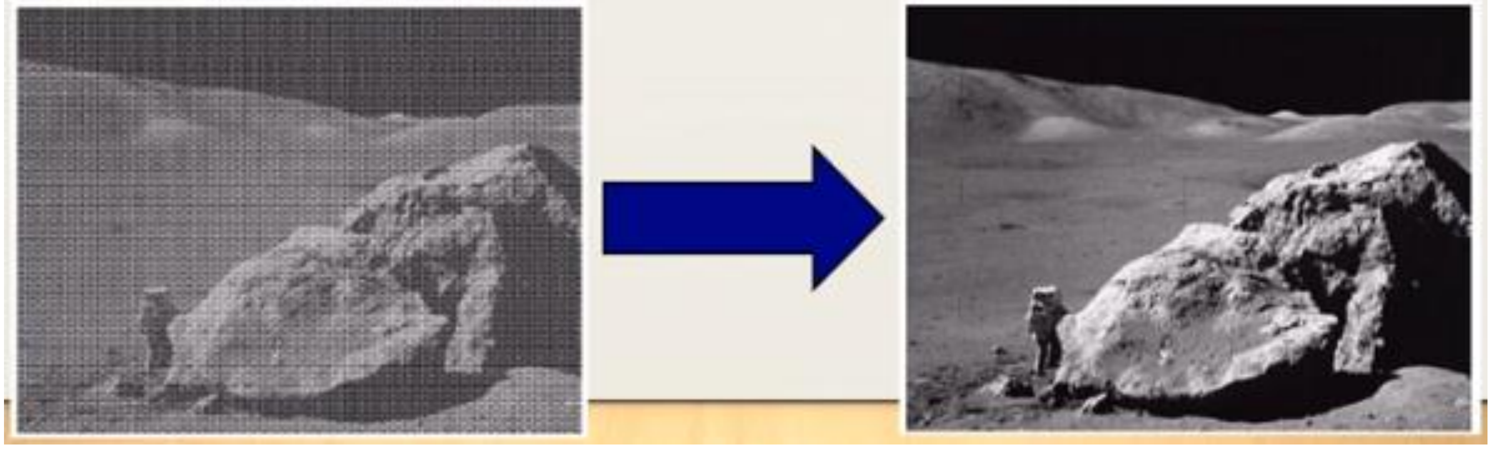


Image restoration vs. Image enhancement

- **Enhancement:**
  - Largely a subjective process
  - Priori knowledge about the degradation is not a must(sometime no degradation is involved)
  - Procedures are heuristic and take advantage of the psychophysical aspects of human visual system
- **Restoration:**
  - More an objective process
  - Images are degraded
  - Tries to recover the image by using the knowledge about the degradation

Noise and Images

- The sources of noise in digital images arise during image acquisition(digitization) and transmission
  - Imaging sensors can be affected by ambient conditions
  - Interference can be added to an image during transmission

Noise Model

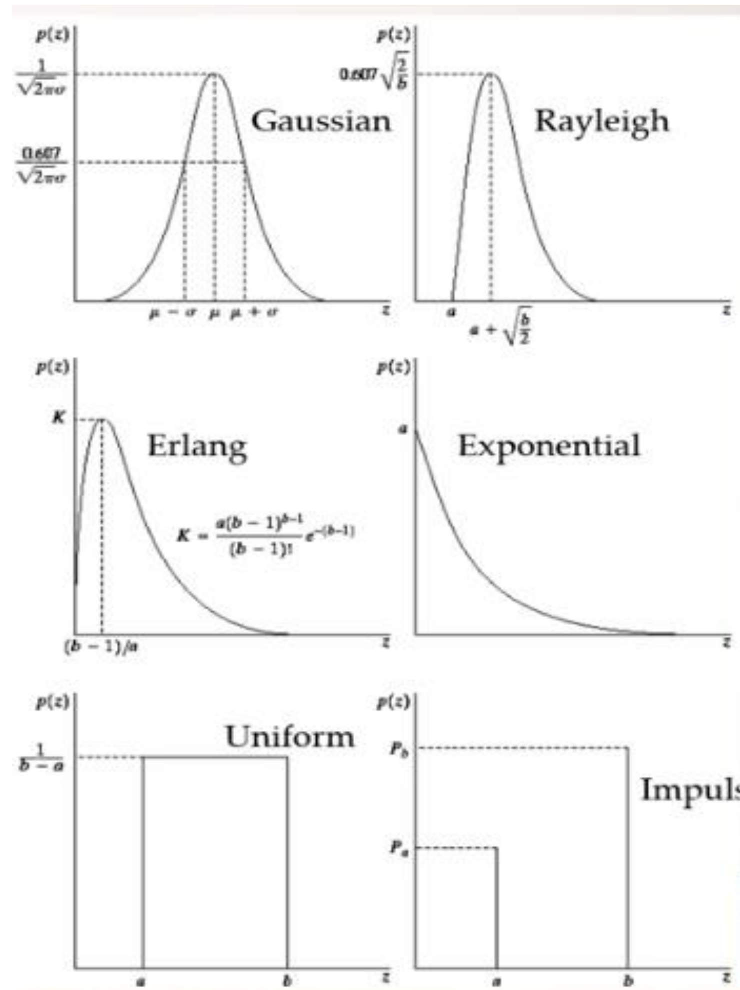
- We can consider a noisy image to be modelled as follows:

$$g(x, y) = f(x, y) + \eta(x, y)$$

- where $f(x, y)$ is the original image pixel, $\eta(x, y)$ is the noise term and $g(x, y)$ is the resulting noisy pixel
- If we can estimate the model the noise in an image is based on this will help us to figure out how to restore the image

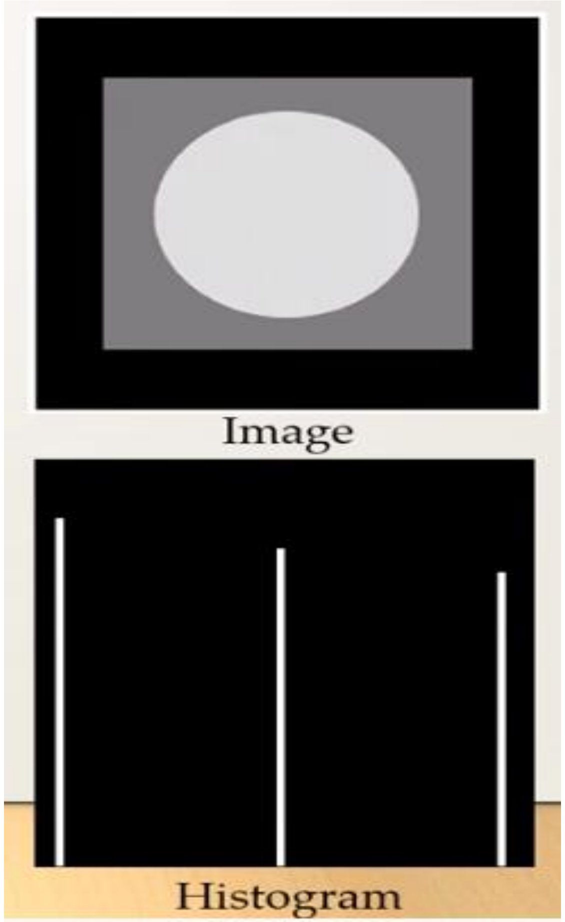
Noise Models

- There are many different models for the image noise term $\eta(x,y)$:
- Gaussian
  - Most common mode
- Rayleigh
- Erlang
- Exponential
- Uniform
- Impulse
  - Salt and pepper noise

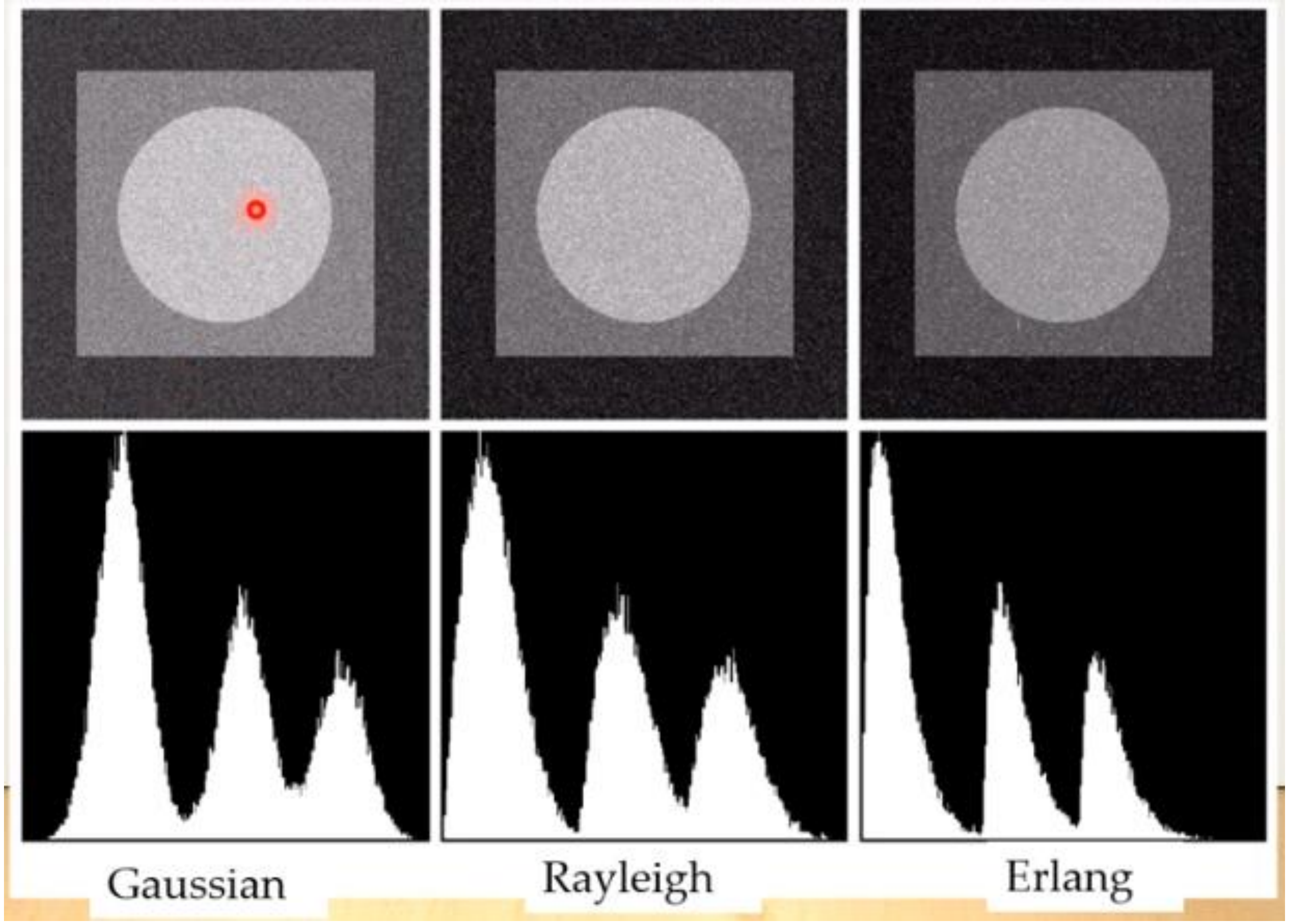


Noise Example

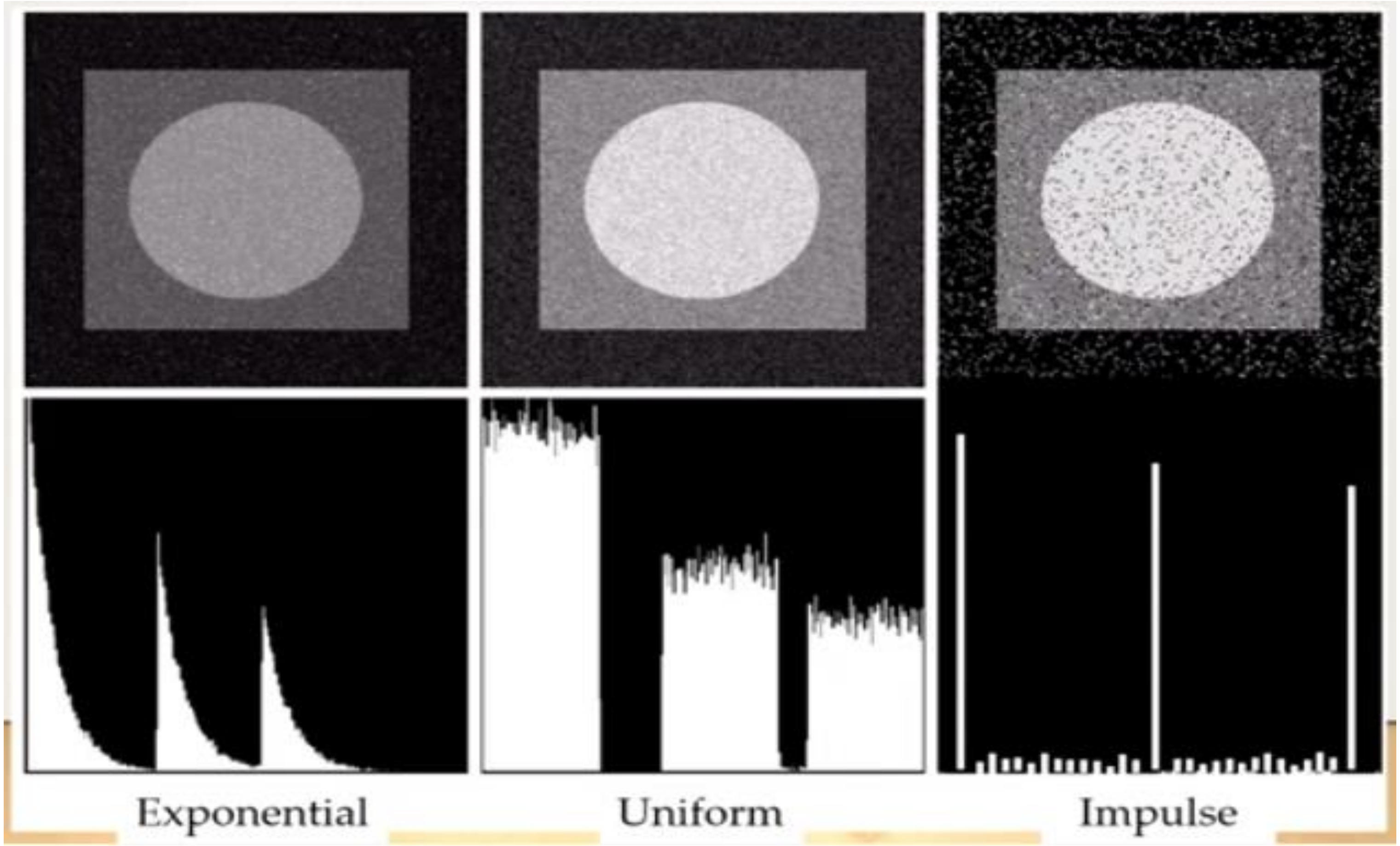
- The test pattern to the right is ideal for demonstrating the addition od noise.
- The following slides will show the result of adding noise based on various models to this image.



Noise Example(Cont…)



Noise Example(Cont…)



Noise Removal
Arithmetic Mean Filter

Filtering to Remove Noise : Arithmetic Mean Filter

- We can use spatial filters of different kinds to remove different kinds of noise
- The *arithmetic mean* filter is a very simple one and is calculated as follows:

$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$

This is implemented as the simple smoothing filter

Blurs the image to remove noise

$$\hat{f}(x,y) = \frac{1}{mn} \sum_{(s,t)\in S_{xy}} g(s,t)$$

Image Capture Device

i - Image capture device

↳ Noise model

$$\underbrace{g(x,y)}_{\text{resulting noisy image}} = \underbrace{f(x,y)}_{\text{Original}} + \underbrace{\eta(x,y)}_{\text{Noise}}$$

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

Filtering to remove Noise

→ Filtering to remove noise

- We can use spatial filters of different kinds to remove different kinds of noise.

1. Arithmetic mean filter:-

$$\hat{f}(x,y) = \frac{1}{mn} \sum g(s,t)$$

At axis x -

$$\hat{f}(x) = \frac{1}{m} \sum g(s)$$

At axis y -

$$\hat{f}(y) = \frac{1}{n} \sum g(t)$$

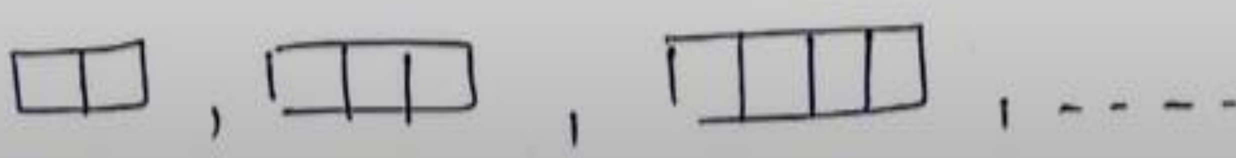
Example: Arithmetic Mean Filter

3, 9, 4, 52, 3, 8, 6, 2, 2, 9.

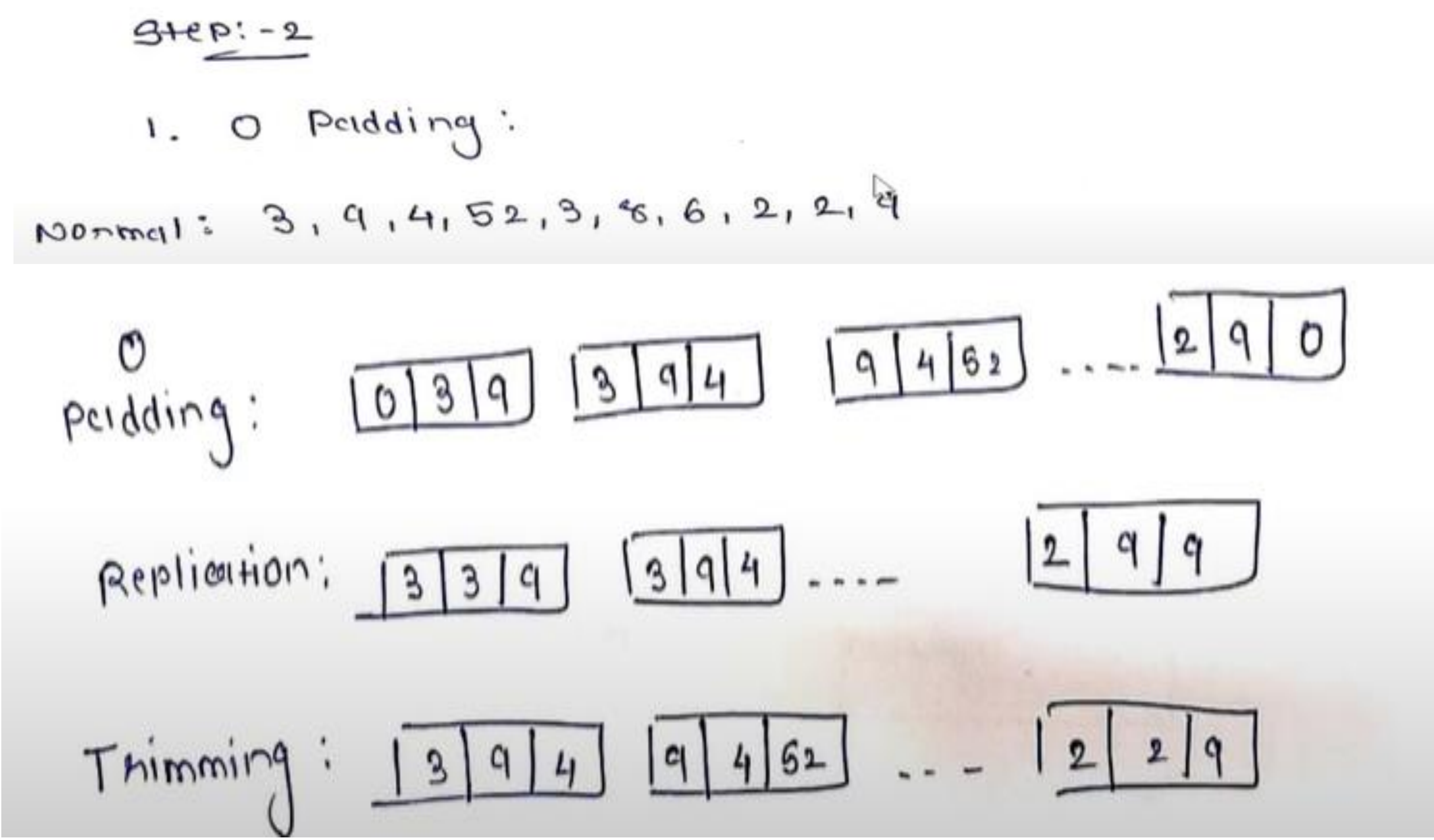
Step:-1

1st take windowing size,

:- we can take any size for windowing.



Step:2


Step:-2
1. 0 Padding :
Normal: 3, 9, 4, 52, 3, 8, 6, 2, 2, 9
0 Padding:
0 | 3 | 9
3 | 9 | 4
9 | 4 | 52
....
2 | 9 | 0
Replication:
3 | 3 | 9
3 | 9 | 4
....
2 | 9 | 9
Trimming:
3 | 9 | 4
9 | 4 | 52
...
2 | 2 | 9

Example: Arithmetic Mean Filter

Step 1: 0 Padding

solve example using arithmetic mean filter :

(i) 0 Padding :

| 0 | 3 | 9 | $= \frac{0+3+9}{3} = 4$

| 3 | 9 | 4 | $= \frac{16}{3} = 5.33 = 5$

| 9 | 4 | 52 | $= 21.66 = 22$

| 4 | 52 | 3 | $= 19.66 = 20$

| 52 | 3 | 8 | $= 21$

| 3 | 8 | 6 | $= 5.66 \approx 6$

| 8 | 6 | 2 | $= 5.33 \approx 5$

| 6 | 2 | 2 | $= 3.33 \approx 3$

| 2 | 2 | 9 | $= 4.33 \approx 4$

| 2 | 9 | 0 | $= 3.66 \approx 4$

| 4 | 5 | 22 | 20 | 21 | 6 | 5 | 3 | 4 | 4 |

Step 2: Replication

(ii) Replication:

| 3 | 3 | 9 | $= \frac{3+3+9}{3} = 5$

other are same only change boundry

| 2 | 9 | 9 | $= 6.66 \approx 7$

5	5	22	20	21	6	5	3	4	7

Step 3: Trimming

(iii) trimming :

Does not change boundary pixel all other pixels are change,

3	5	22	20	21	6	5	3	4	9

Other Means

- There are different kinds mean filters all of which exhibit slightly different behavior:
  - Geometric Mean
  - Harmonic Mean
  - Contrahamonic Mean

Geometric Mean:

- Geometric Mean:

$$\hat{f}(x,y) = \left[\prod_{(s,t)\in S_{xy}} g(s,t) \right]^{\frac{1}{mn}}$$

- Achieves similar smoothing to the arithmetic mean, but tends to lose less image detail

② Geometric mean :-

$$\hat{f}(x,y) = \left(\prod g(s,t)\right)^{1/mn}$$

at x axis,

$$\hat{f}(x) = \left(\prod g(s)\right)^{1/m} = \left(\prod x_i\right)^{1/m}$$

at y axis,

$$\hat{f}(y) = \left(\prod g(t)\right)^{1/n} = \left(\prod y_i\right)^{1/n}$$

Geometric Mean: Example

(i) Replication :-

3, 9, 4, 52, 3, 87, 6, 2, 2, 9

$\boxed{3 \mid 3 \mid 9}$ $= \left((3 \times 3)(9)\right)^{1/3} = (81)^{1/3} = 4.32$

$\boxed{3 \mid 9 \mid 4}$ $= 4.76 \approx 5$

$\boxed{9 \mid 4 \mid 52}$ $= 12.32 \approx 12$

$\boxed{4 \mid 52 \mid 3}$ $= 8.54 \approx 9$

Cont…

| 4 | 52 | 3 | = 8.54 ≈ 9

| 52 | 3 | 8 | = 10.76 ≈ 11

| 3 | 8 | 6 | = 5.24 ≈ 5

| 8 | 6 | 2 | = 4.57 ≈ 5

| 6 | 2 | 2 | = 2.88 ≈ 3

| 2 | 2 | 9 | = 3.30 ≈ 3

| 2 | 9 | 9 | = 5.45 ≈ 5

| 4 | 5 | 12 | 9 | 11 | 5 | 5 | 3 | 3 | 5 |

Cont…

(ii) Padding:

0	3	9

= 0

2	9	0

= 0

0	5	12	9	11	5	5	3	3	0

(iii) Trimming:

3	9	4

= 4.76 ≈ 5

2	2	9

= 3.30 ≈ 3

3	5	12	9	11	5	5	3	3	9

Harmonic Mean

Harmonic Mean:

$$\hat{f}(x,y) = \frac{mn}{\displaystyle\sum_{(s,t)\in S_{xy}} \frac{1}{g(s,t)}}$$

- Works well for salt noise, but fails for pepper noise
- Also does well for other kinds of noise such as Gaussian noise

Cont…

③ Harmonic mean :- work for salt noise but fails for pepper noise

$$\hat{f}(x,y) = \frac{mn}{\sum \frac{1}{g(s,t)}} = \frac{mn}{\sum \frac{1}{(x_i + y_i)}}$$

at x axis,

$$\hat{f}(x) = \frac{m}{\sum \frac{1}{g(s)}} = \frac{m}{\sum \frac{1}{x_i}}$$

Gaussian, salt.

at y axis

$$\hat{f}(y) = \frac{n}{\sum \frac{1}{g(t)}} = \frac{n}{\sum \frac{1}{y_i}}$$

Example: Harmonic Mean Filter

[3, 9, 4, 52, 3, 8, 6, 2, 2, 9]

(i) Replicating:

$\boxed{3 \mid 3 \mid 9} = \frac{3}{(1/15)} = 45 = \frac{3}{\frac{1}{3}+\frac{1}{3}+\frac{1}{9}} = \frac{3(9)}{3+3+1} = 3.85 = 4$

$\boxed{3 \mid 9 \mid 4} = \frac{3}{(1/16)} = 48 = \frac{3}{\frac{1}{3}+\frac{1}{9}+\frac{1}{4}} = 4.32$

$\boxed{9 \mid 4 \mid 52} = \frac{3}{(1/65)} = 195 = \frac{3}{\frac{1}{9}+\frac{1}{4}+\frac{1}{52}}$ - -

$\boxed{\quad}$ as well all belowed;

Contrahamonic Mean

- **Contraharmonic Mean:**

$$\hat{f}(x,y) = \frac{\sum_{(s,t)\in S_{xy}} g(s,t)^{Q+1}}{\sum_{(s,t)\in S_{xy}} g(s,t)^{Q}}$$

- Q is the *order* of the filter and adjusting its value changes the filter's behaviour
- Positive values of Q eliminate pepper noise
- Negative values of Q eliminate salt noise

Contrahamonic Mean Equation

(4) contraharmonic mean :-

$$\hat{f}(x,y) = \frac{\sum g(s,t)^{Q+1}}{\sum g(s,t)^{Q}}$$

Q is order of the filter & adjusting its value changes the filter's behaviour

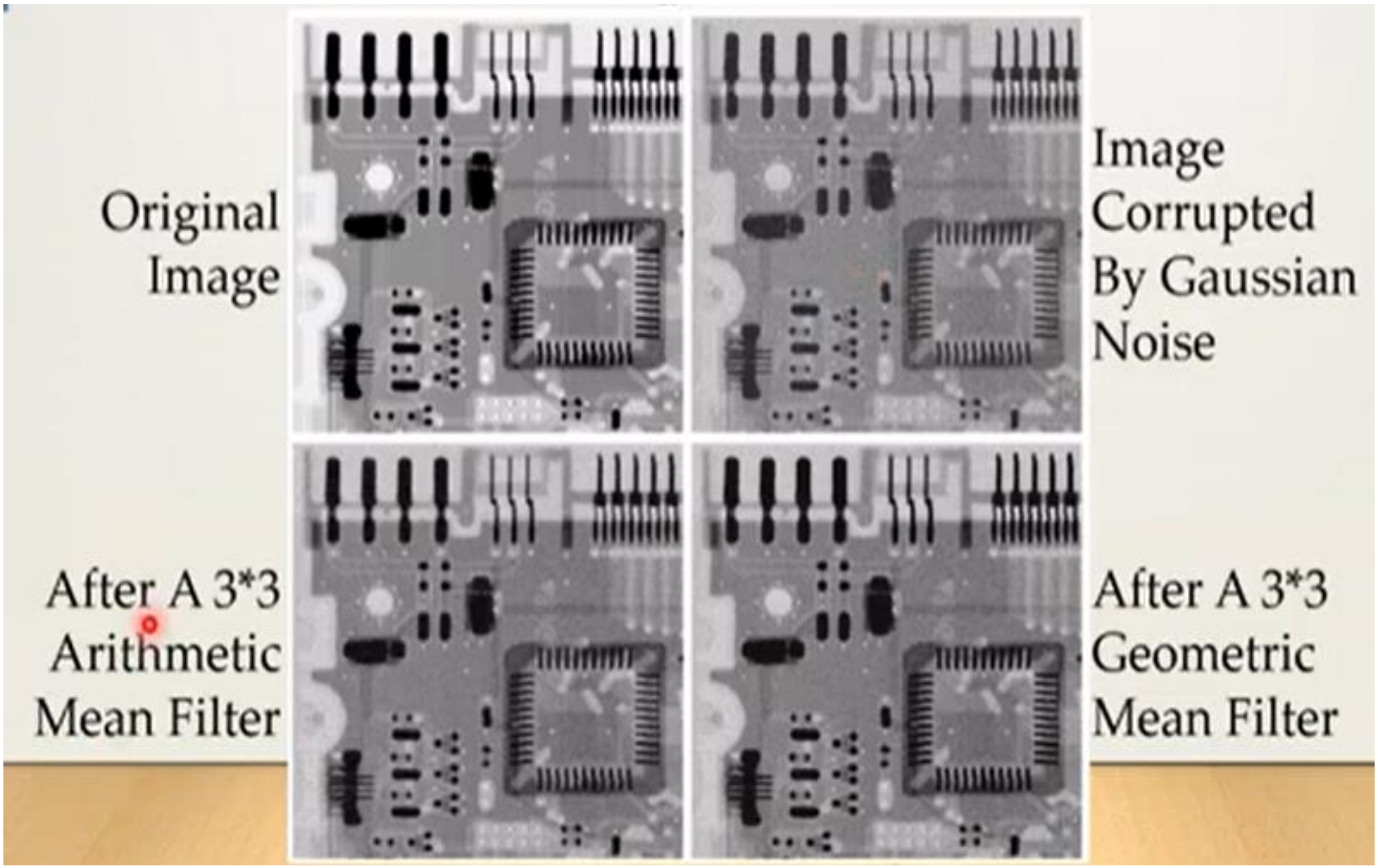
if $Q > 0$ = Q eliminate pepper noise

$Q < 0$ = " salt "

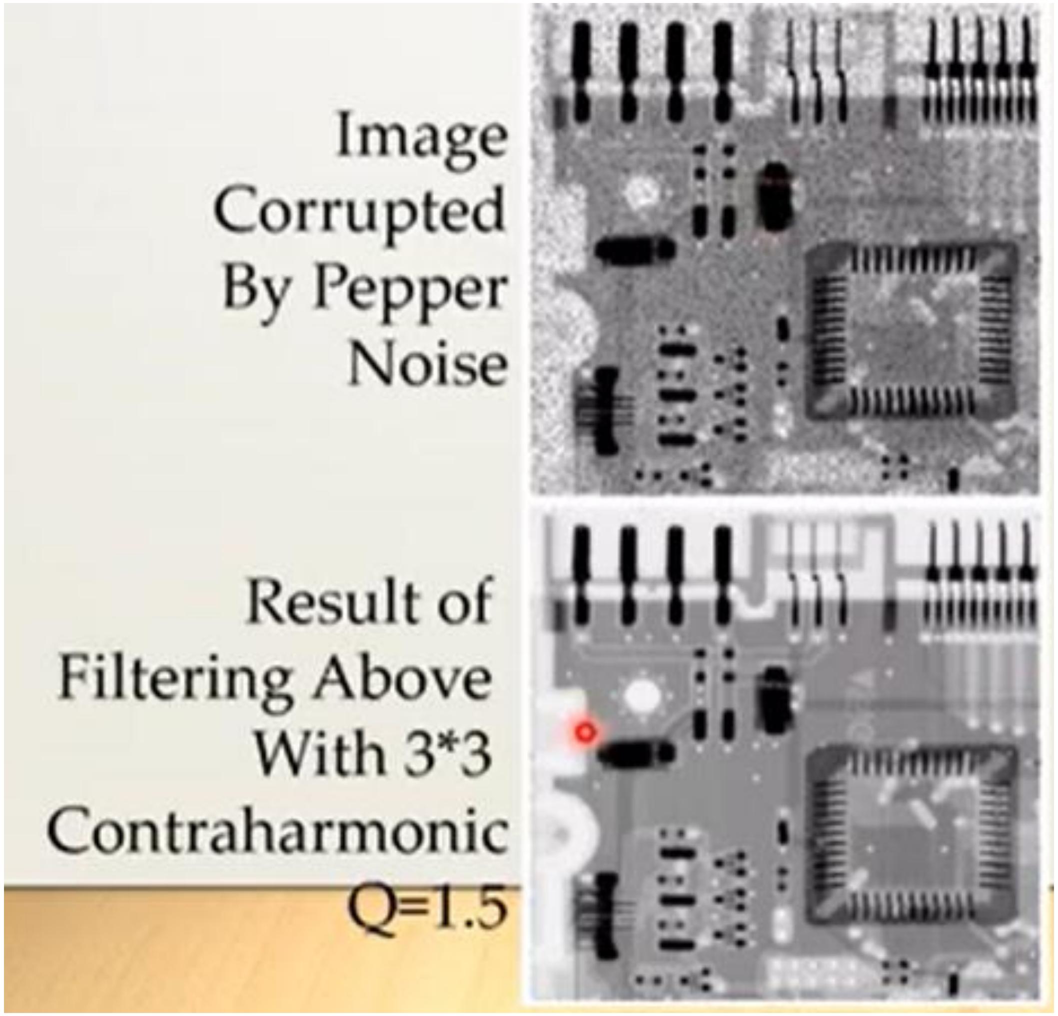
$Q = 0$ = arithmetic mean

$Q = -1$ = harmonic filter

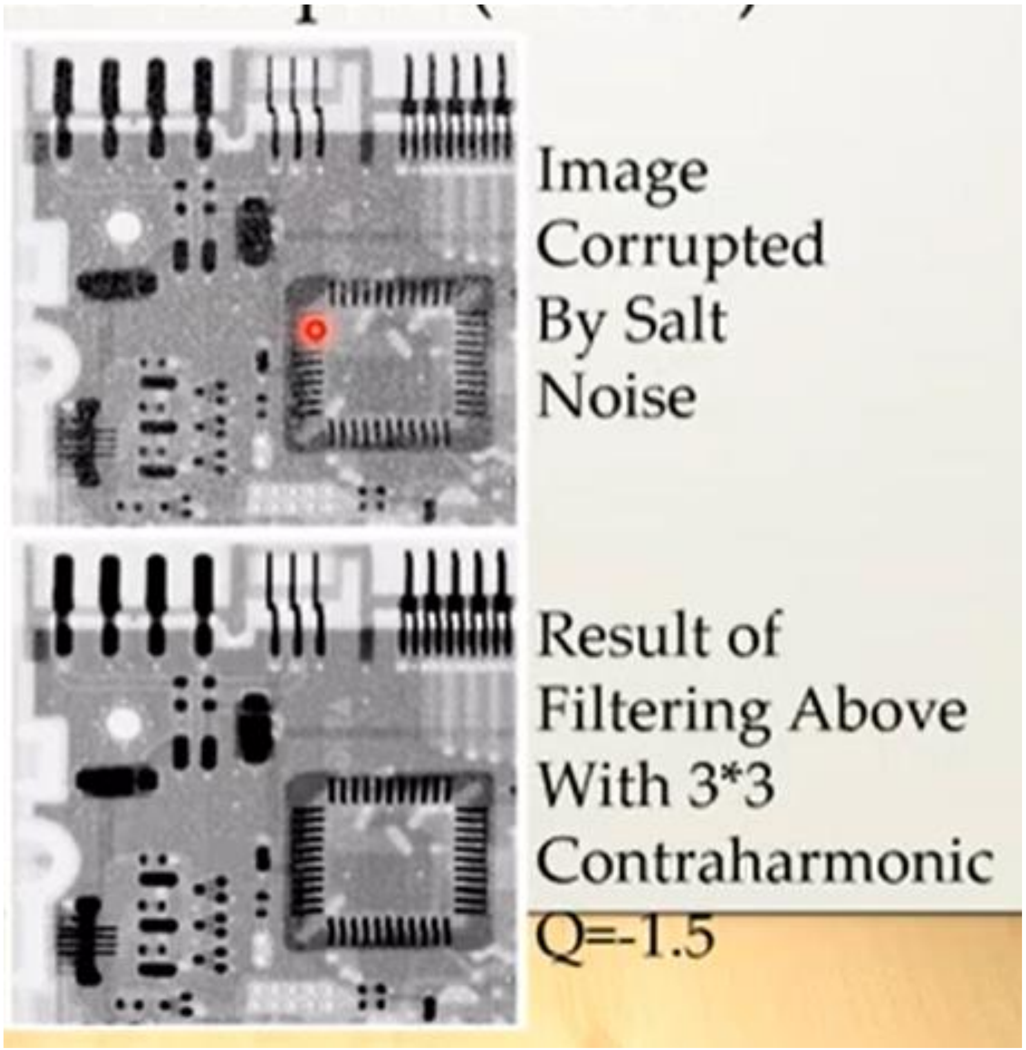
Noise Removal Examples



Noise Removal Examples

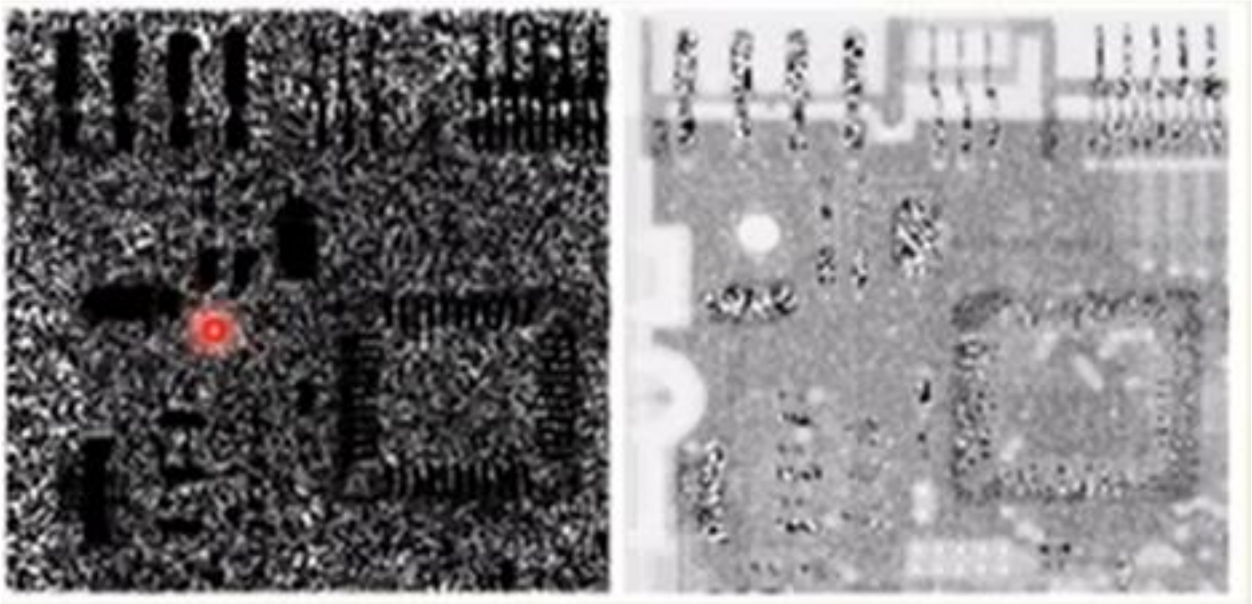


Noise Removal Examples



Contraharmonic Filter

- Choosing the wrong value for Q when using contra harmonic filter can have drastic results



Noise Removal
Order Statistics Filters

Order Statistics Filters

- Spatial filters that are based on ordering the pixel values that make up the neighborhood operated on by the filter
- Useful spatial filters include
  - Median filter
  - Max and Min filter
  - Midpoint filter
  - Alpha trimmed mean filter

Median Filter

- **Median Filter:**

$$\hat{f}(x,y) = \underset{(s,t)\in S_{xy}}{median}\{g(s,t)\}$$

- Excellent at noise removal, without the smoothing effects that can occur with other smoothing filters.
- Particularly good when salt and pepper noise is present.

Median Filter Example
Step 1:Replication

(1) Median filter:

$\hat{f}(x,y) = \text{median}(g(s,t))$

3, 9, 4, 52, 3, 8, 6, 2, 2, 9

(1) Replication:-

|3|3|9| = 3

|3|9|4| → |3|4|9| = 4

|9|4|52| → |4|9|52| = 9

|4|52|3| → |3|4|52| = 4

|52|3|8| → |3|8|52| = 8

|3|8|6| → |3|6|8| = 6

|8|6|2| → |2|6|8| = 6

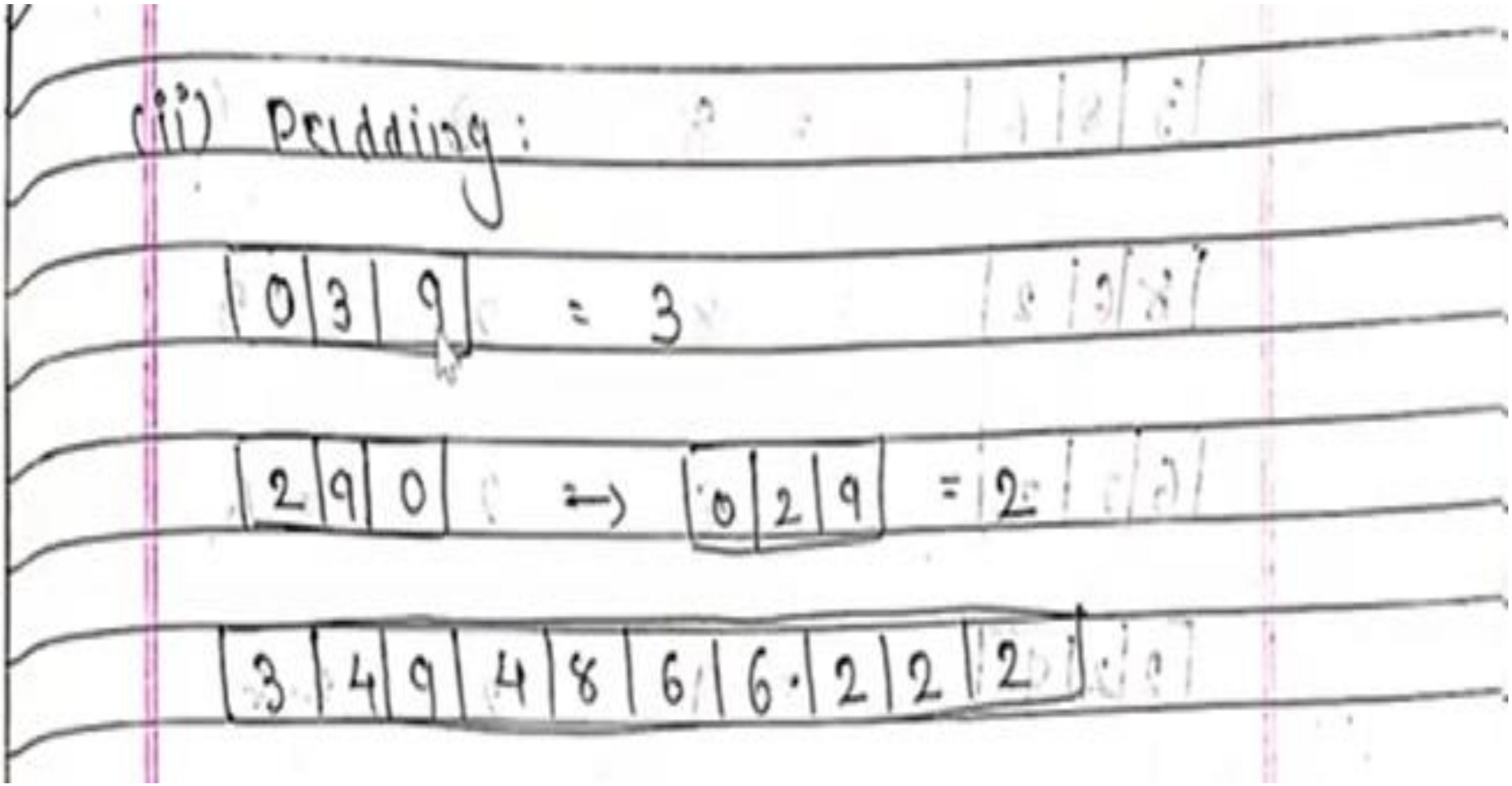
|6|2|2| → |2|2|6| = 2

|2|2|9| → |2|2|9| = 2

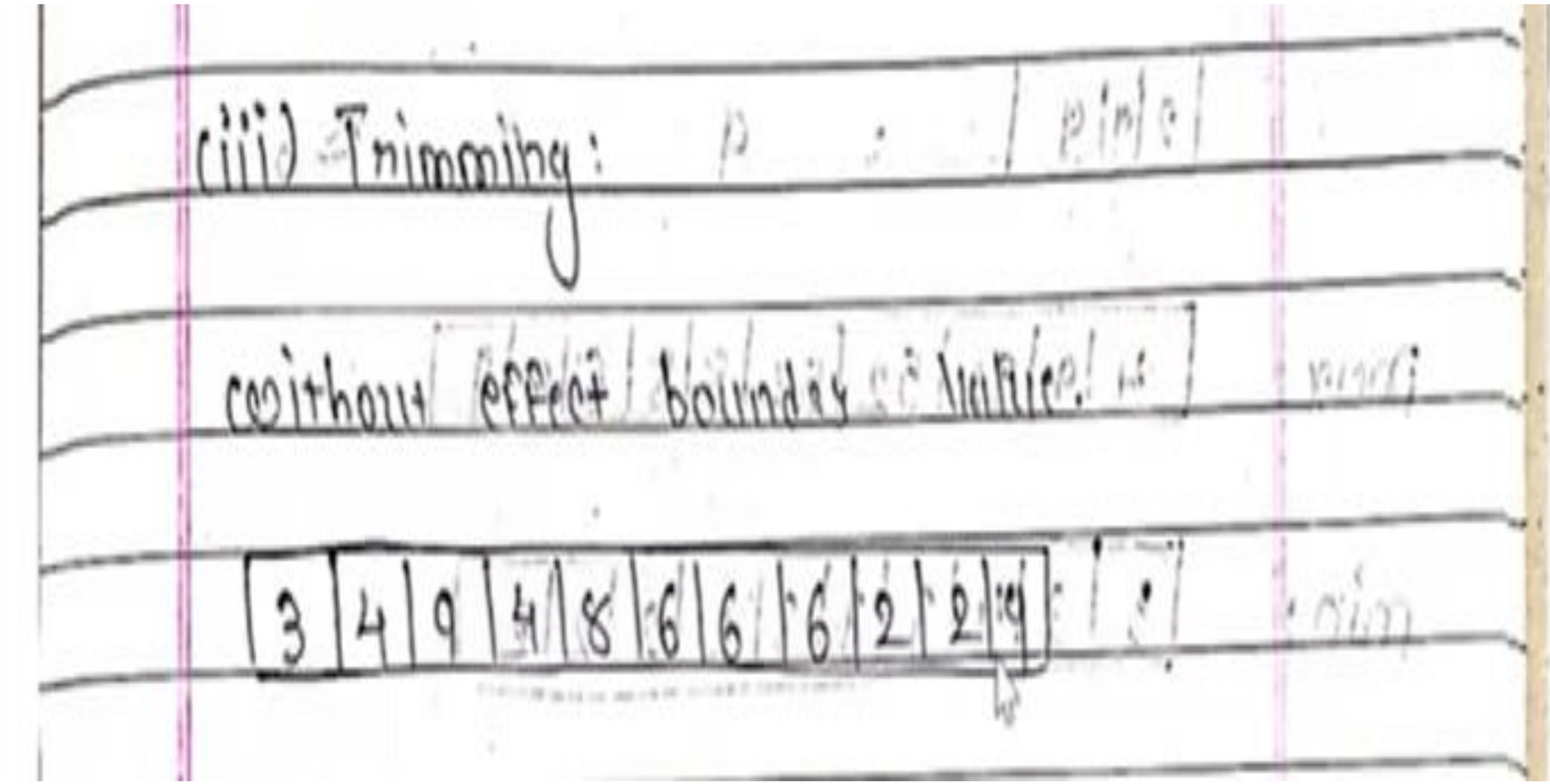
|2|9|9| = 9

|3|4|9|4|8|6|6|2|2|9|

Step 2: Padding



Step 3: Trimming



Max and Min Filter

Max Filter:

$$\hat{f}(x,y) = \max_{(s,t)\in S_{xy}} \{g(s,t)\}$$

Min Filter:

$$\hat{f}(x,y) = \min_{(s,t)\in S_{xy}} \{g(s,t)\}$$

Max filter is good for pepper noise and min is good for salt noise

Max and Min Filter Example
Step 1: Replication

3, 9, 4, 52, 3, 8, 6, 2, 2, 9

Given

(i) Replication :

	max	min	mid
$\boxed{3 \mid 3 \mid 9}$	= 84	3	3.5 ≈ 4
$\boxed{3 \mid 9 \mid 4}$	=> 9	3	6
$\boxed{9 \mid 4 \mid 52}$	=> 52	4	28
$\boxed{4 \mid 52 \mid 3}$	= 52	3	27.5 ≈ 28

52 | 3 | 8	= 52	3	≈ 28
3 | 8 | 6	= 8	3	≈ 6
8 | 6 | 2	= 8	2	≈ 5
6 | 2 | 2	= 6	2	≈ 6
2 | 2 | 9	= 9	2	≈ 6
2 | 9 | 9	= 9	2	≈ 6

max: | 4 | 9 | 52 | 52 | 52 | 8 | 8 | 6 | 9 | 9 |

min: | 3 | 3 | 4 | 3 | 3 | 3 | 2 | 2 | 2 | 2 |

Step 2: Trimming

(ii) Trimming :

3	3	9	= 3	3	= 3
2	2	9	= 9	9	= 9

max :	3	9	5 2	5 2	5 2	8	8	6	9	3
min :	3	3	4	3	3	3	2	2	2	9

Step 3: Padding

(iii) Padding :

| 0 | 3 | 9 | = 9   0   = 5

| 2 | 9 | 0 | = 9   0   = 5

max	9	9	52	52	52	8	8	6	9	3	9
min	0	3	34	3	3	3	2	2	2	0	

Midpoint Filter

Midpoint Filter:

$$\hat{f}(x,y) = \frac{1}{2}\left[\max_{(s,t)\in S_{xy}}\{g(s,t)\} + \min_{(s,t)\in S_{xy}}\{g(s,t)\}\right]$$

Good for random Gaussian and uniform noise

Midpoint Filter Example

(i) Replication:

max min

~~3|3|9~~

midpoint:

5	6	28	28	28	6	5	6	6	6

(ii) Trimming:

midpoint:

3	6	28	28	28	6	5	6	6	9

(iii) Padding:

midpoint:

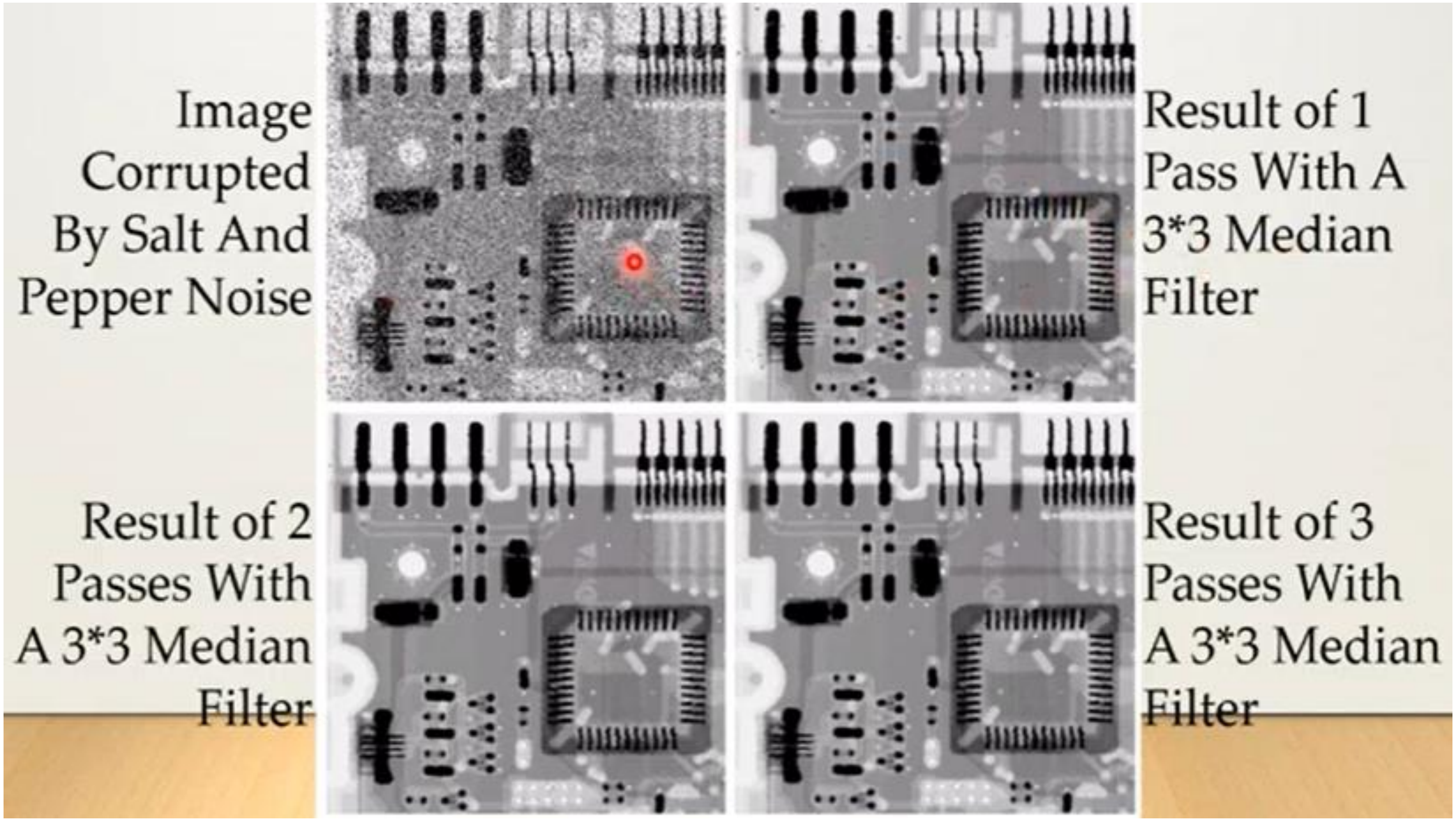
5	6	28	28	28	6	5	6	6	5

Alpha-Trimmed Mean Filter

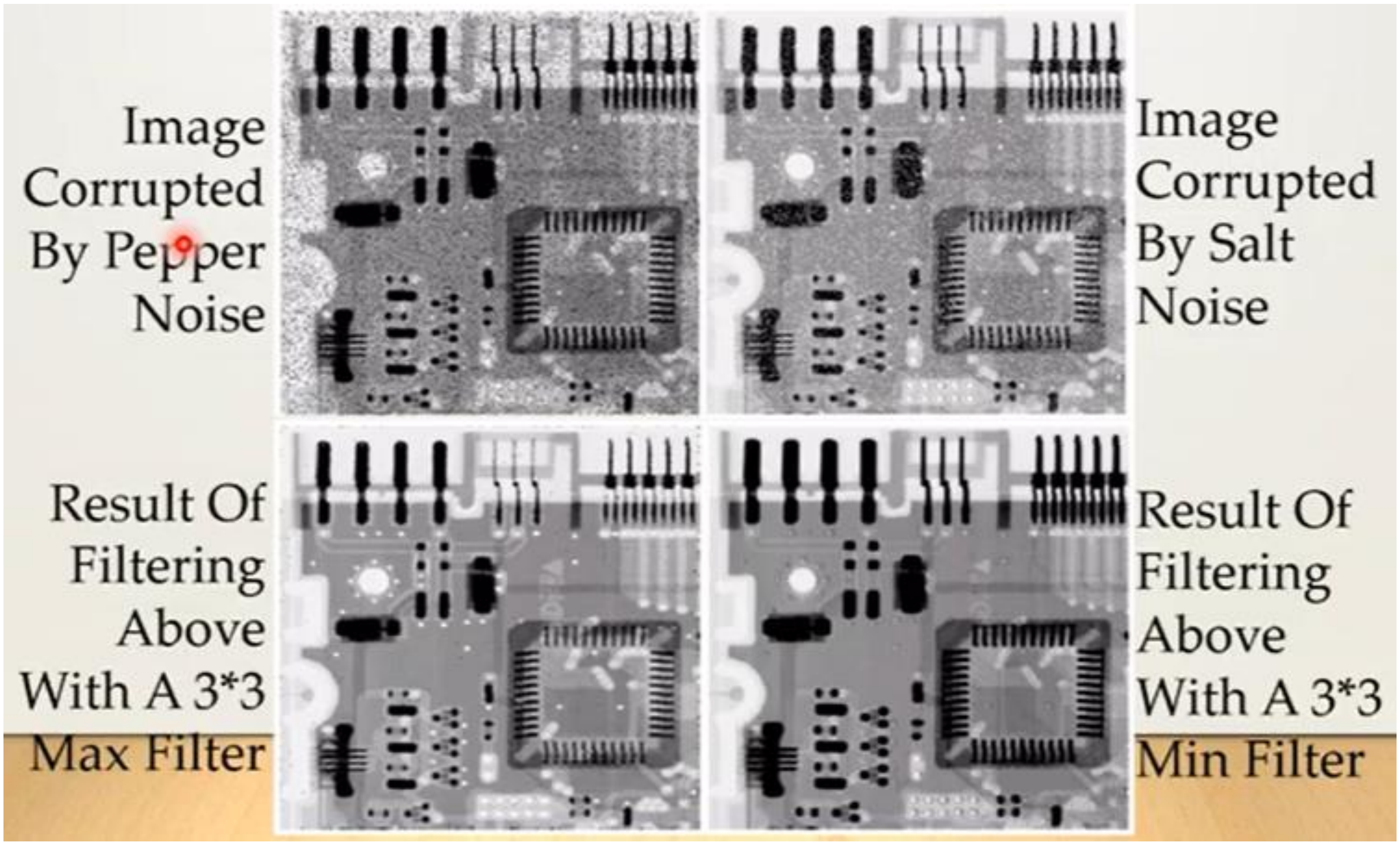
$$\hat{f}(x,y) = \frac{1}{mn-d} \sum_{(s,t)\in S_{xy}} g_r(s,t)$$

- We can delete the d/2 lowest and d/2 highest grey levels
- So $g_r(s,t)$ $represents\ the\ remaining\ mn - d\ pixels$

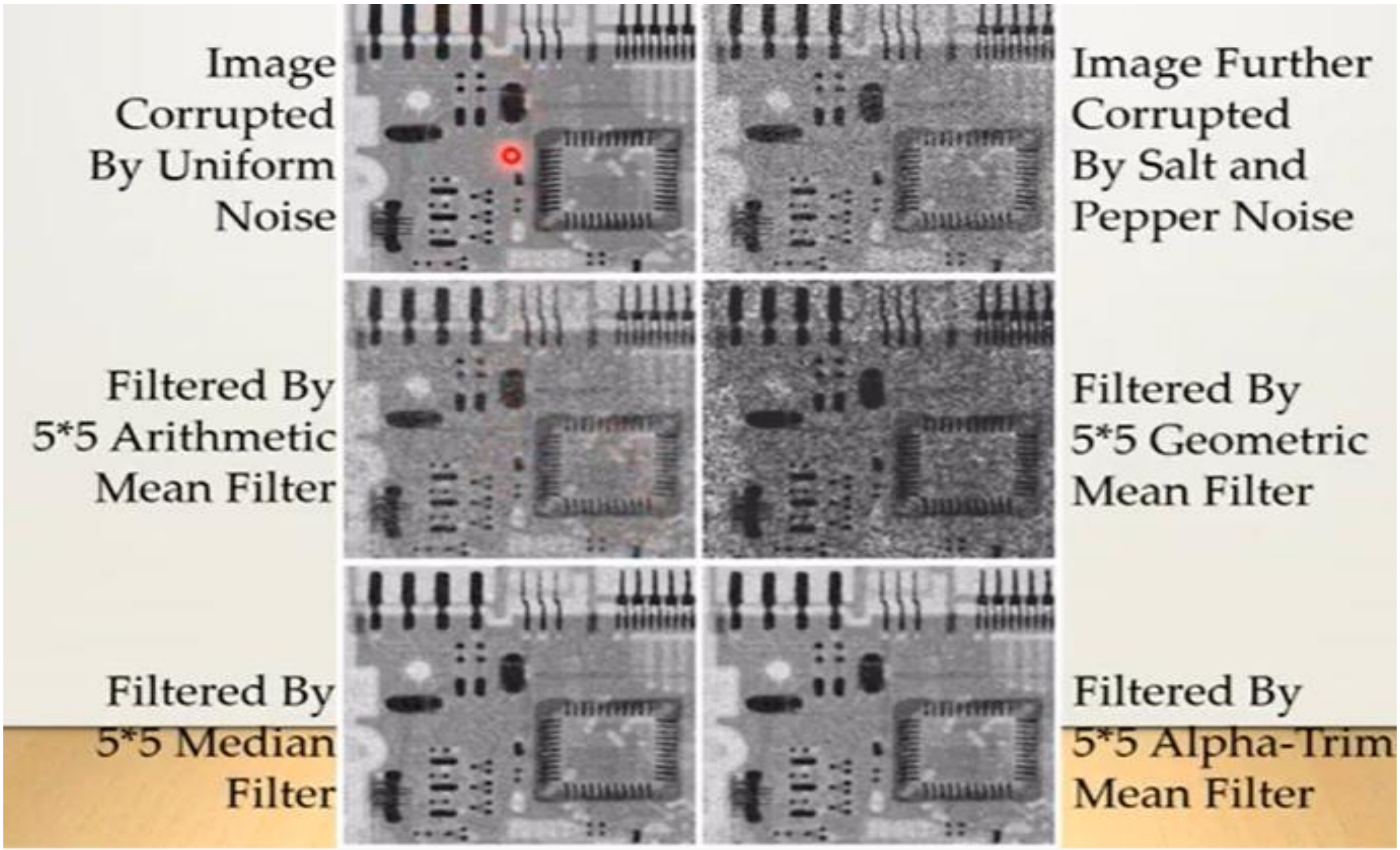
Noise Removal Examples



Noise Removal Examples



Noise Removal Examples



Periodic Noise

- Typically arises due to electrical or electromagnetic interference
- Gives rise to regular noise patterns in image
- Frequency domain techniques in the Fourier domain are most effective at removing periodic noise

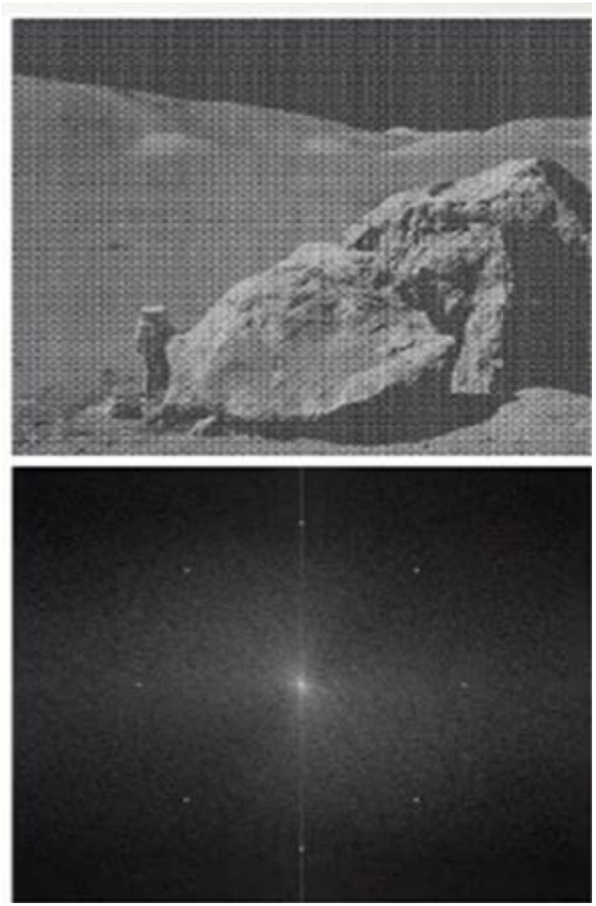


Image Enhancement
Fourier Transformation

Filter and Frequency

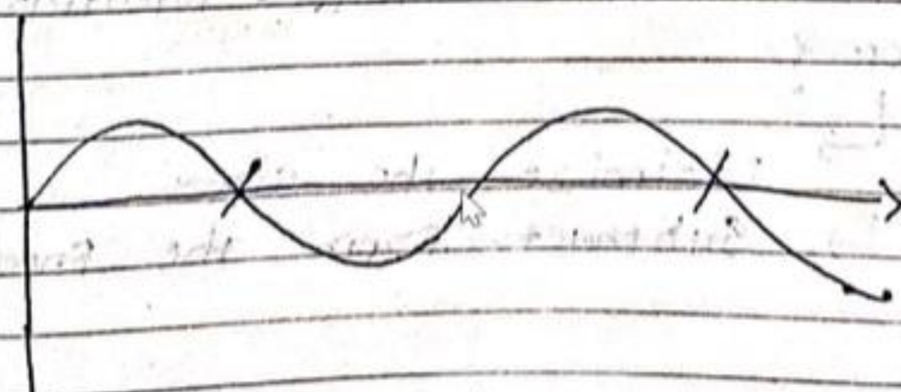
- **Frequency:** The number of times that a periodic function repeats the same sequence of values during a unit variation of the independent variable.
- **Filter:** A device or material for suppressing or minimizing waves or oscillations of certain frequencies

↳ ① Frequency :-

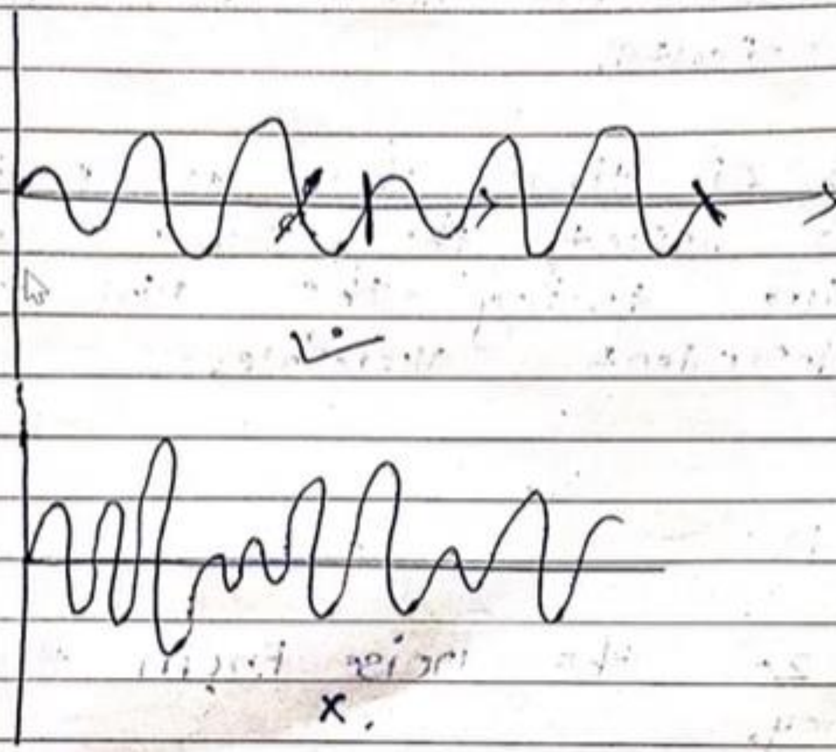
:- Same set of data how many time repeats.

:- The No. of time that a periodic function repeat the same sequence of value during the unit variation of independent variable.

① Frequency :-



:- Data must be same type.



:- Different data must be repeat at the same periods of time T.

→ ② Filter :-

:- Minimize the noise from the frequency.

:- A device / material for suppressing or minimizing waves or oscillations of certain frequency.

:- eliminate noise from frequency

↳ How?

↳ Minimize the size

↳ Subtract from the frequency.

i/P Digital → Sampling (DFT) ↓ Frequency Domain method (Complex No.) 0.10 + i5 → Filter ↓ Smoothing / Sparthing (convolution method) ↓ O/P complex (x + yi)

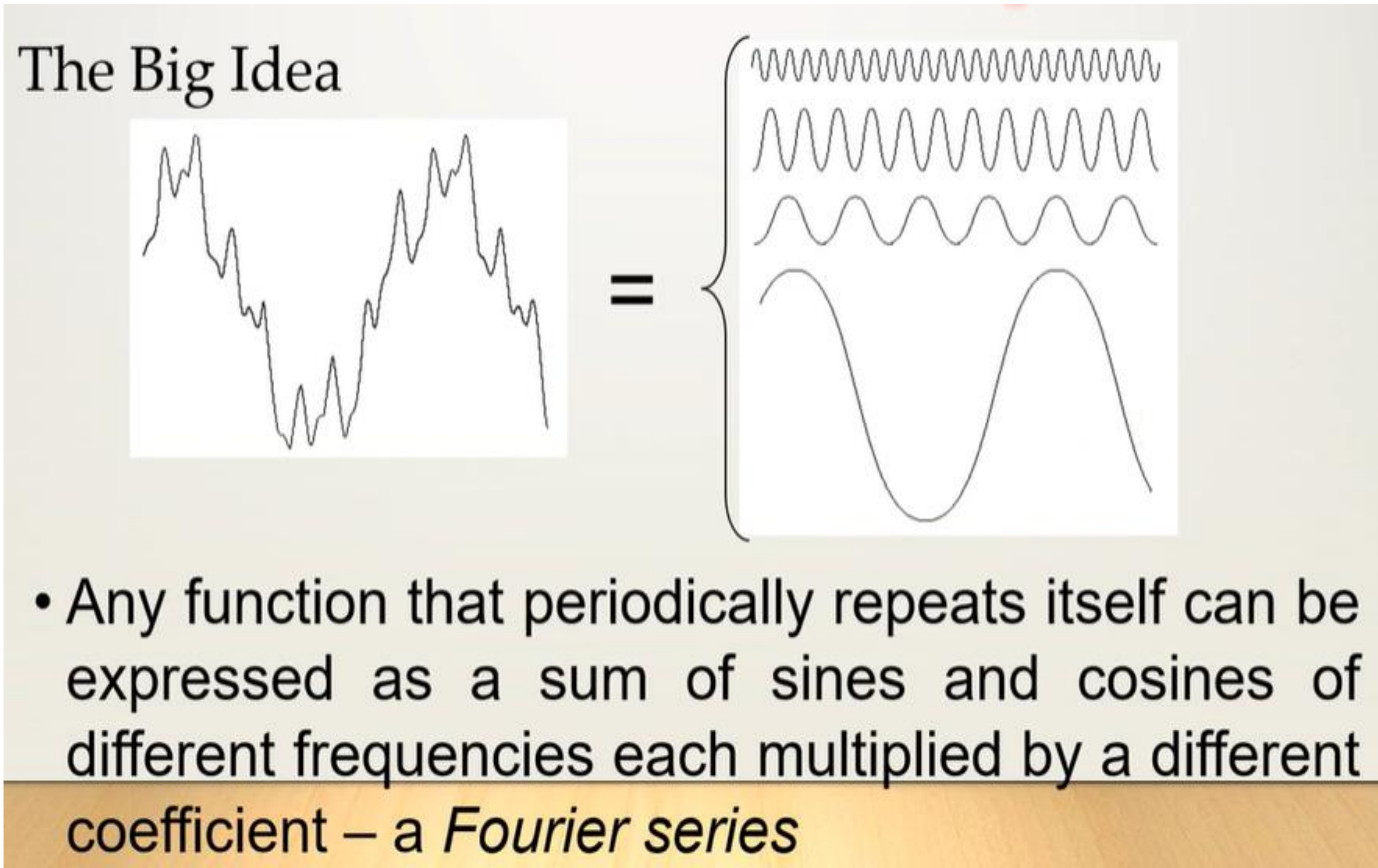
Reverse ←

(In DFT always apply reversible method)

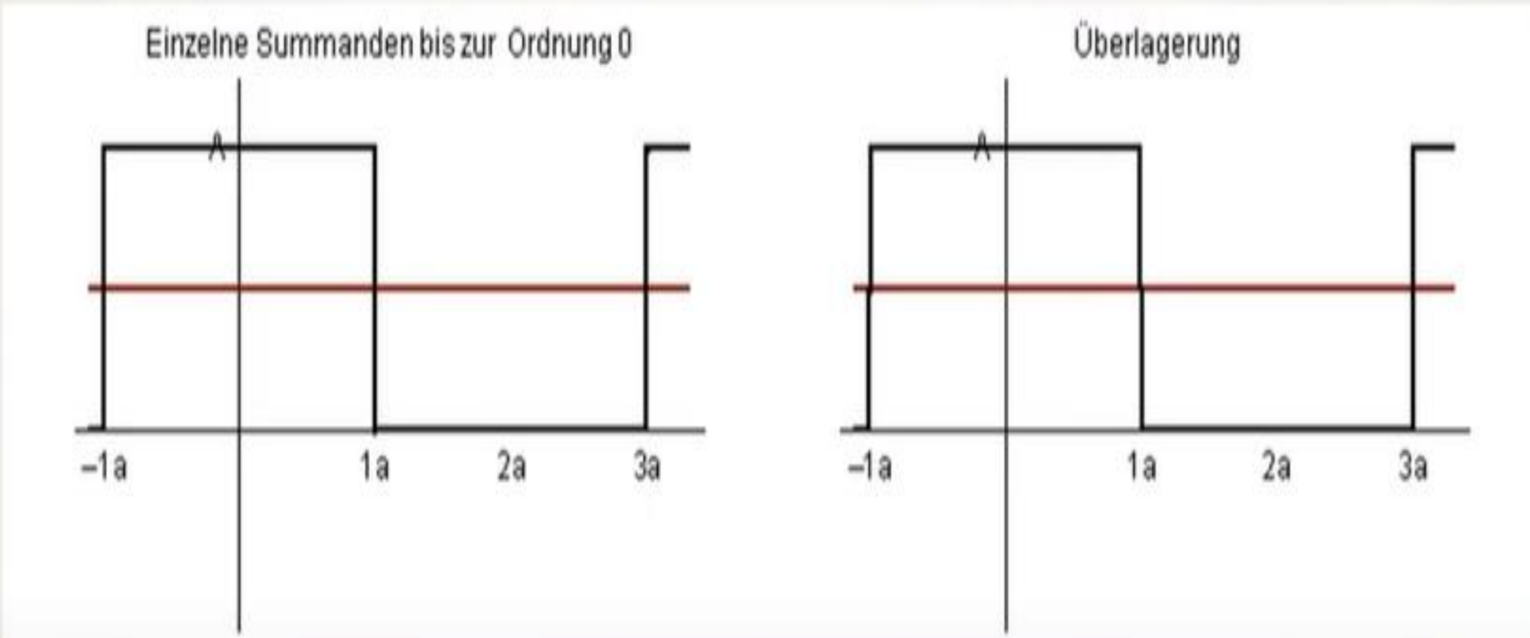
machine can not disply to, the complex No. then we negalate to imaginary part.

generate ← ~~method~~ Enhance ~~can't~~ be image

The Big Idea

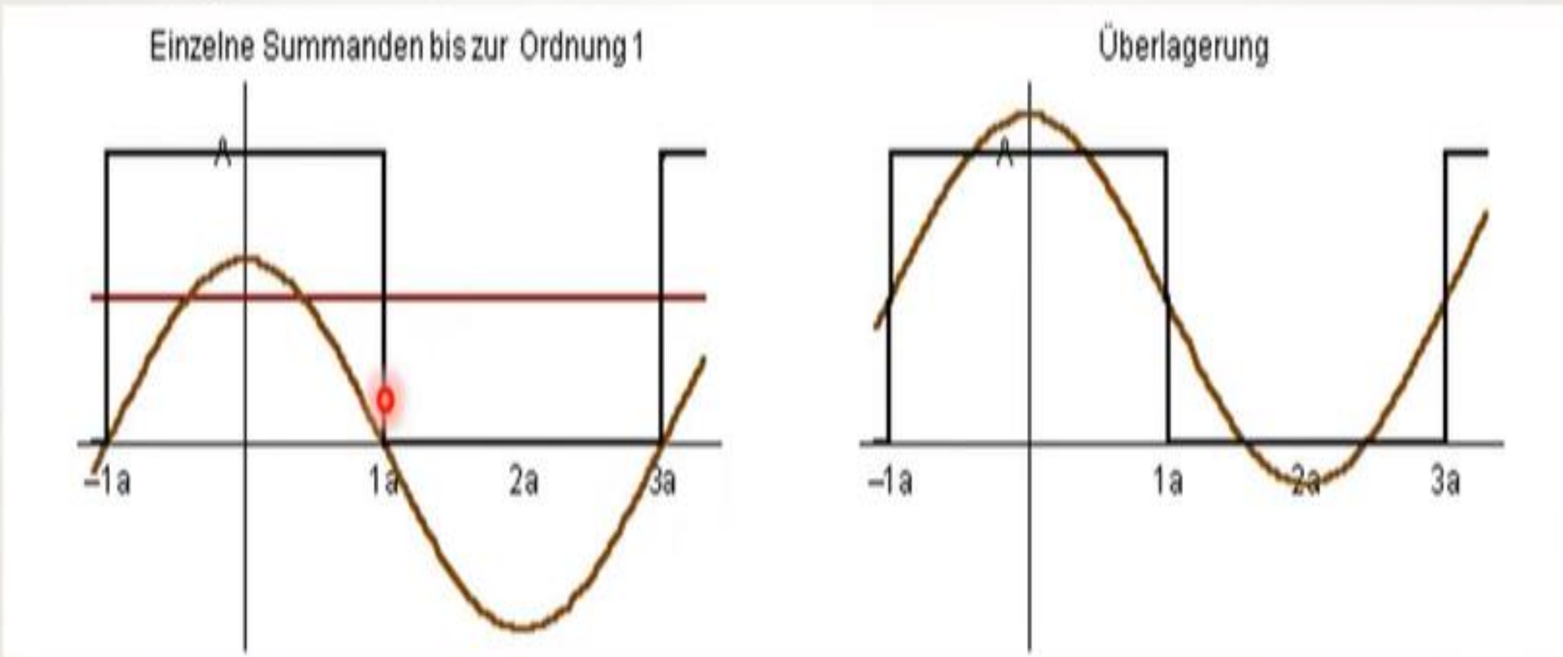

The Big Idea
=
• Any function that periodically repeats itself can be expressed as a sum of sines and cosines of different frequencies each multiplied by a different coefficient – a *Fourier series*

The Big Idea(Cont…)



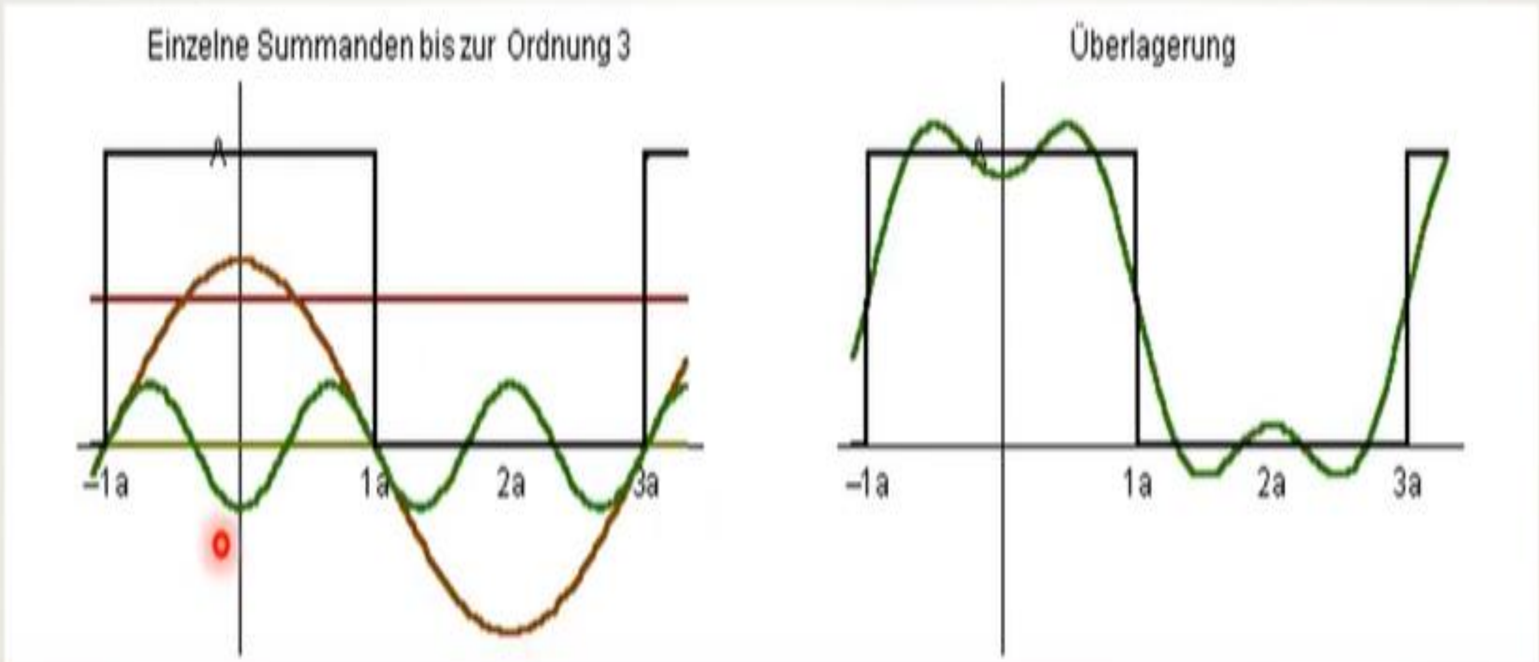
Notice how we get closer and closer to the original function as we add more and more frequencies

The Big Idea(Cont…)



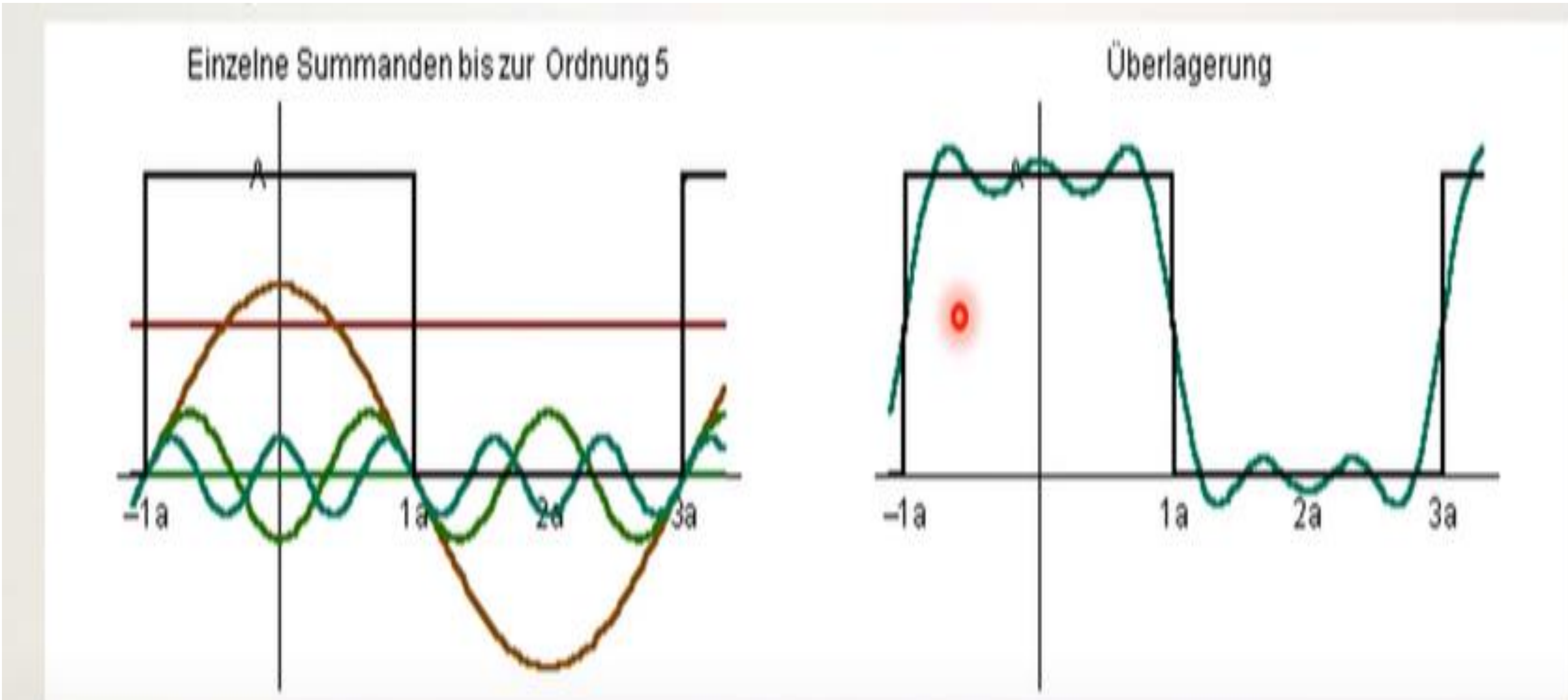
Notice how we get closer and closer to the original function as we add more and more frequencies

The Big Idea(Cont…)



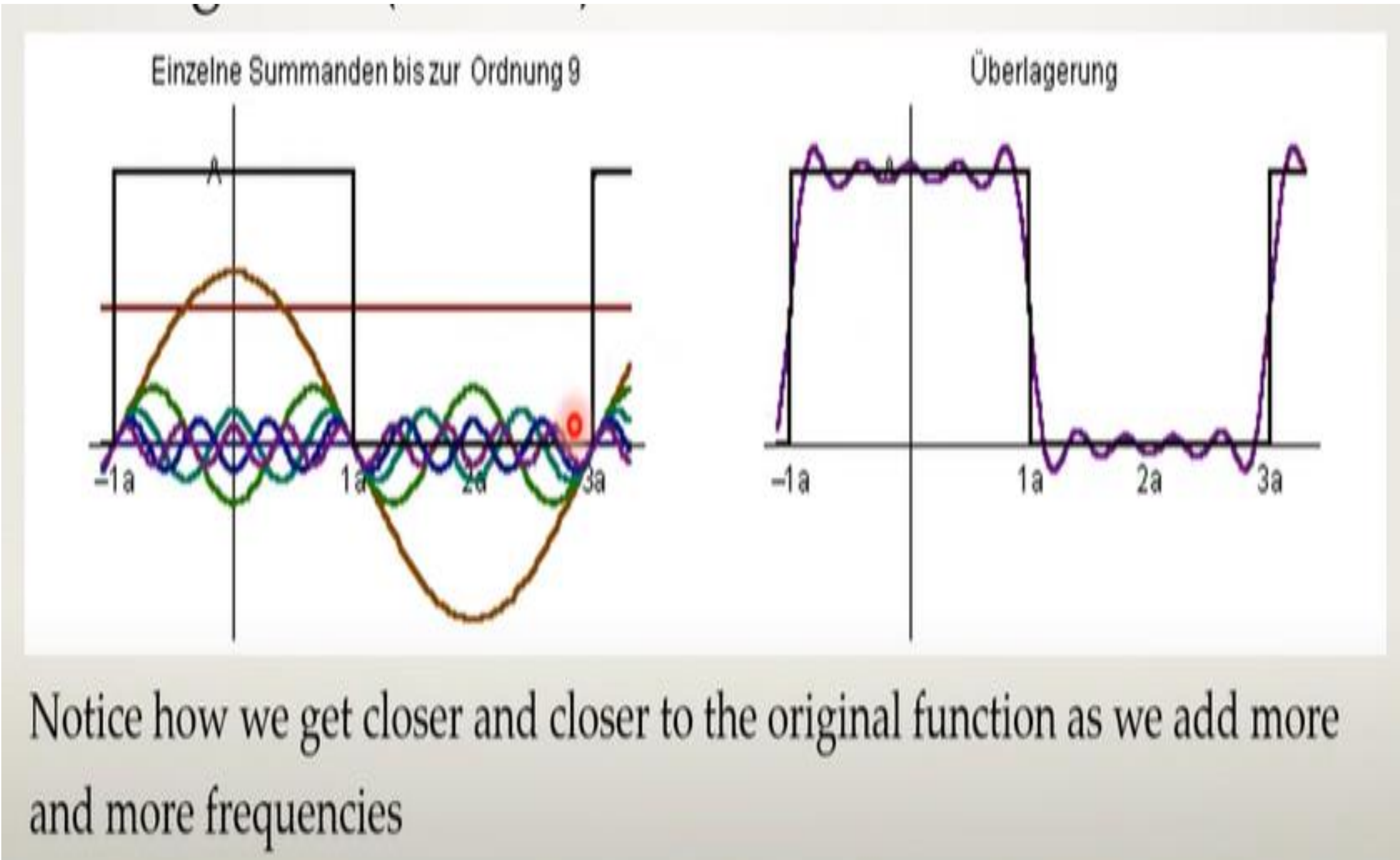
Notice how we get closer and closer to the original function as we add more and more frequencies

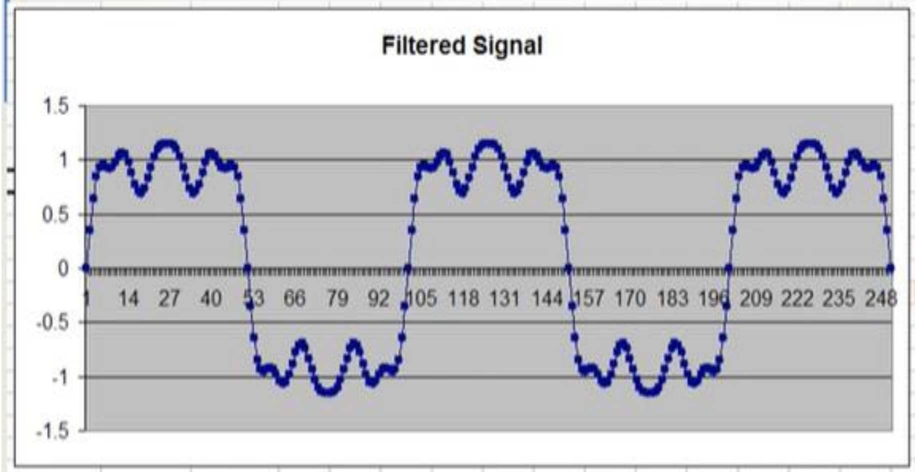
The Big Idea(Cont…)

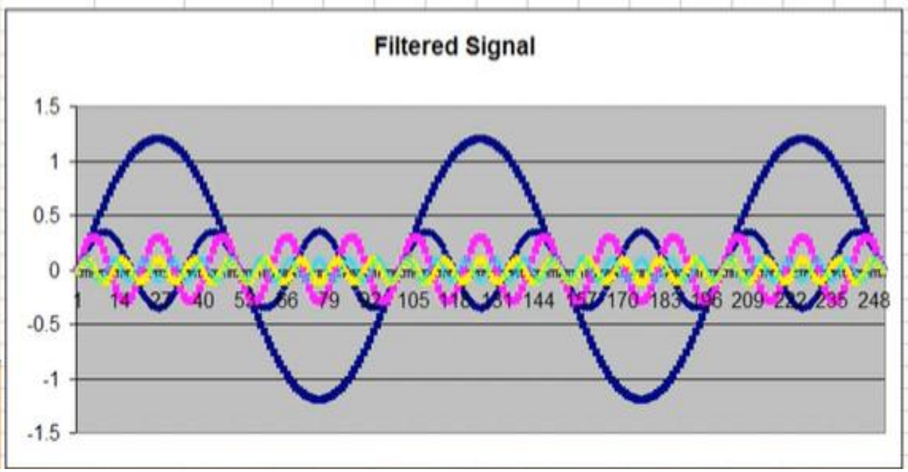


Notice how we get closer and closer to the original function as we add more and more frequencies

The Big Idea(Cont…)







Frequency domain signal processing example in Excel

Discrete Fourier Transform(DFT)

- The Discrete Fourier Transform of *f(x,y)*for *x*=0,1,2…M-1 and *y*=0,1,2,,,N-1,denoted by *F(u,v)* is given by the equation

$$F(u,v)=\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi(ux/M+vy/N)}$$

for $u = 0, 1, 2 \ldots M-1$ and $v = 0, 1, 2 \ldots N-1$.

Discrete Fourier transform(DFT)

↳ Discrete Fourier transform (DFT)

$x = 0, 1, 2, \ldots M-1$

$y = 0, 1, 2 \ldots N-1$

2D image

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\, e^{-2\pi j\left(\frac{ux}{M} + \frac{vy}{N}\right)}$$

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) \cdot e^{-2\pi j\left(\frac{ux}{M} + \frac{vy}{N}\right)}$$

$u = 0, 1, \ldots M-1$

$v = 0, 1 \ldots N-1$

1D image for x

$$F(u) = \sum_{x=0}^{M-1} f(x) \cdot e^{-2\pi j\left(\frac{ux}{M}\right)}$$

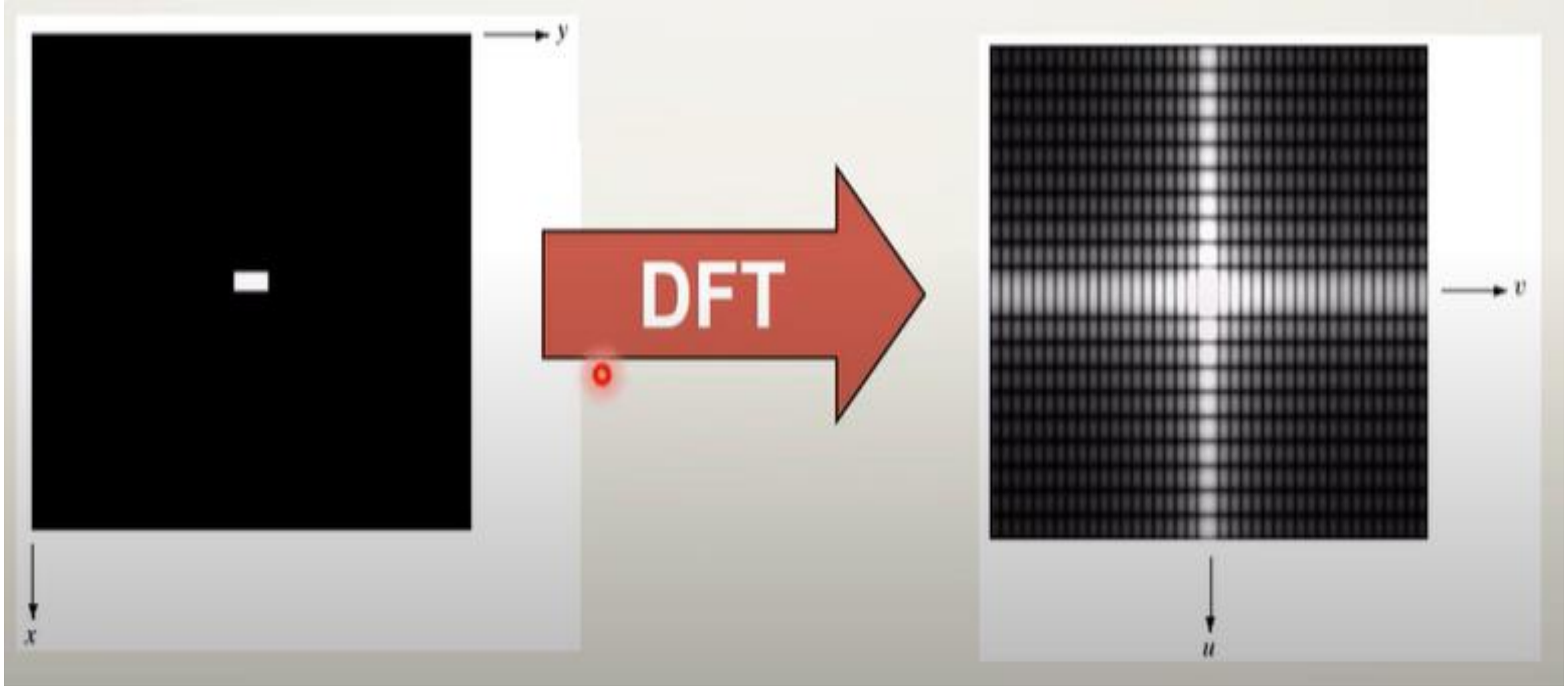
$u = 0, 1, 2, \ldots M-1$

1D image

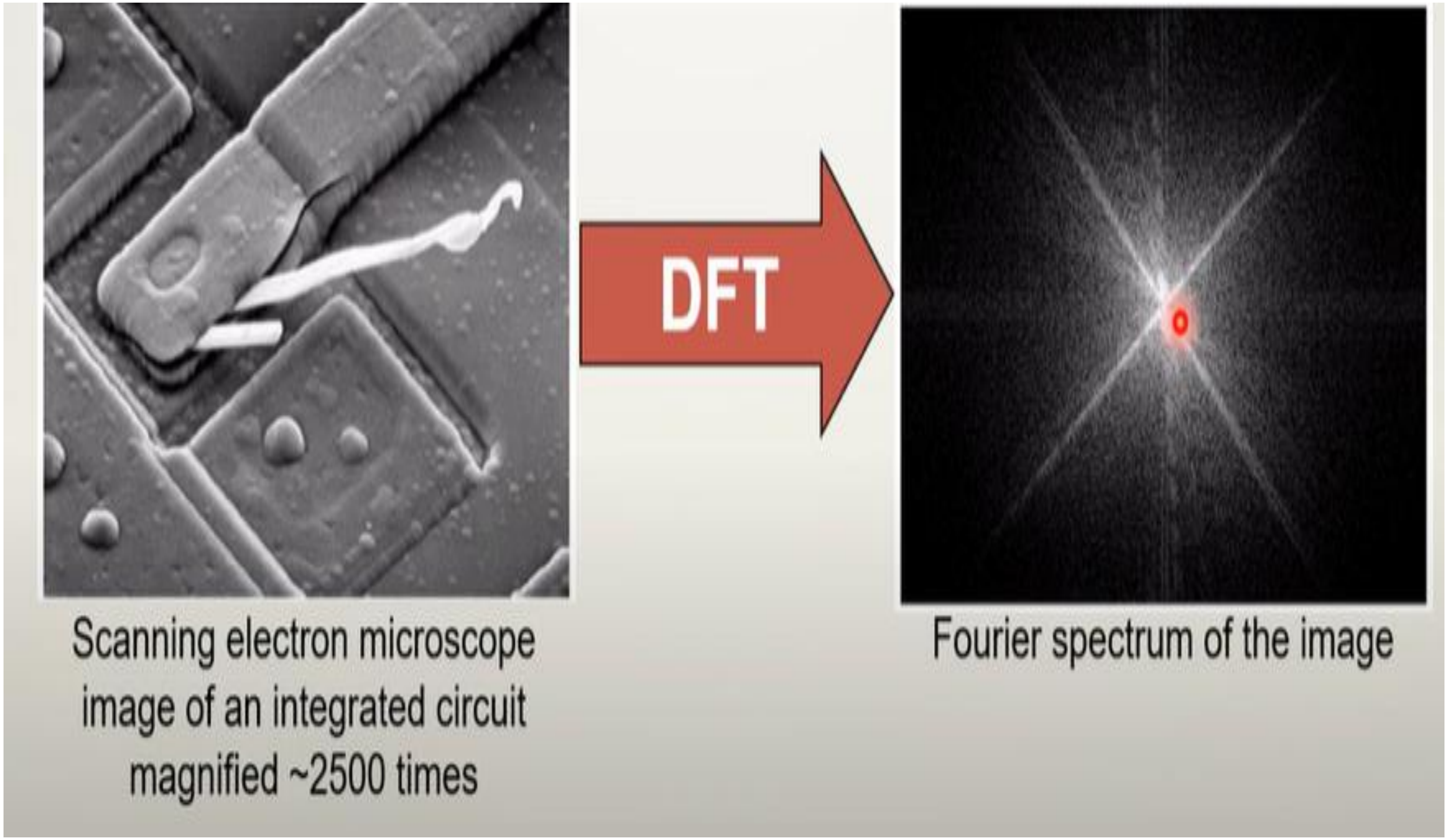
$$F(v) = \sum^{N-1} f(y) \cdot e^{-2\pi j\left(\frac{vy}{N}\right)}$$

DFT & Images

- The DFT of a two dimensional image can be visualised by showing the spectrum of the image component frequencies



DFT & Image(Cont…)



Inverse DFT

- It is really important to note that the Fourier transform is completely reversible
- The inverse DFT is given by:

$$f(x,y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{j2\pi(ux/M + vy/N)}$$

- For x=0,1,2…M-1 and y=0,1,2…N-1

→ The DFT & image processing:

$f(x,y) \xrightarrow[\text{(Fourier transform)}]{\text{DFT}} F(u,v) \xrightarrow{\text{Filter()}} H(u,v)$

mask → $H(u,v)$ (smoothing, sharpening)

$f(x,y)$ ↑ i/p image

$F(u,v)$ (In form of complex)
Extract only real information
remove the imaginary part
$(x + iy)$

$\underset{\text{mask}}{I(u,v)} \cdot H(u,v)$

$G(u,v)$ enhance image

$F^{-1}(u,v) \xleftarrow[G(u,v)]{\text{Inverse}} G(u,v)$

Inverse Fourier transform. → $f(x,y)$

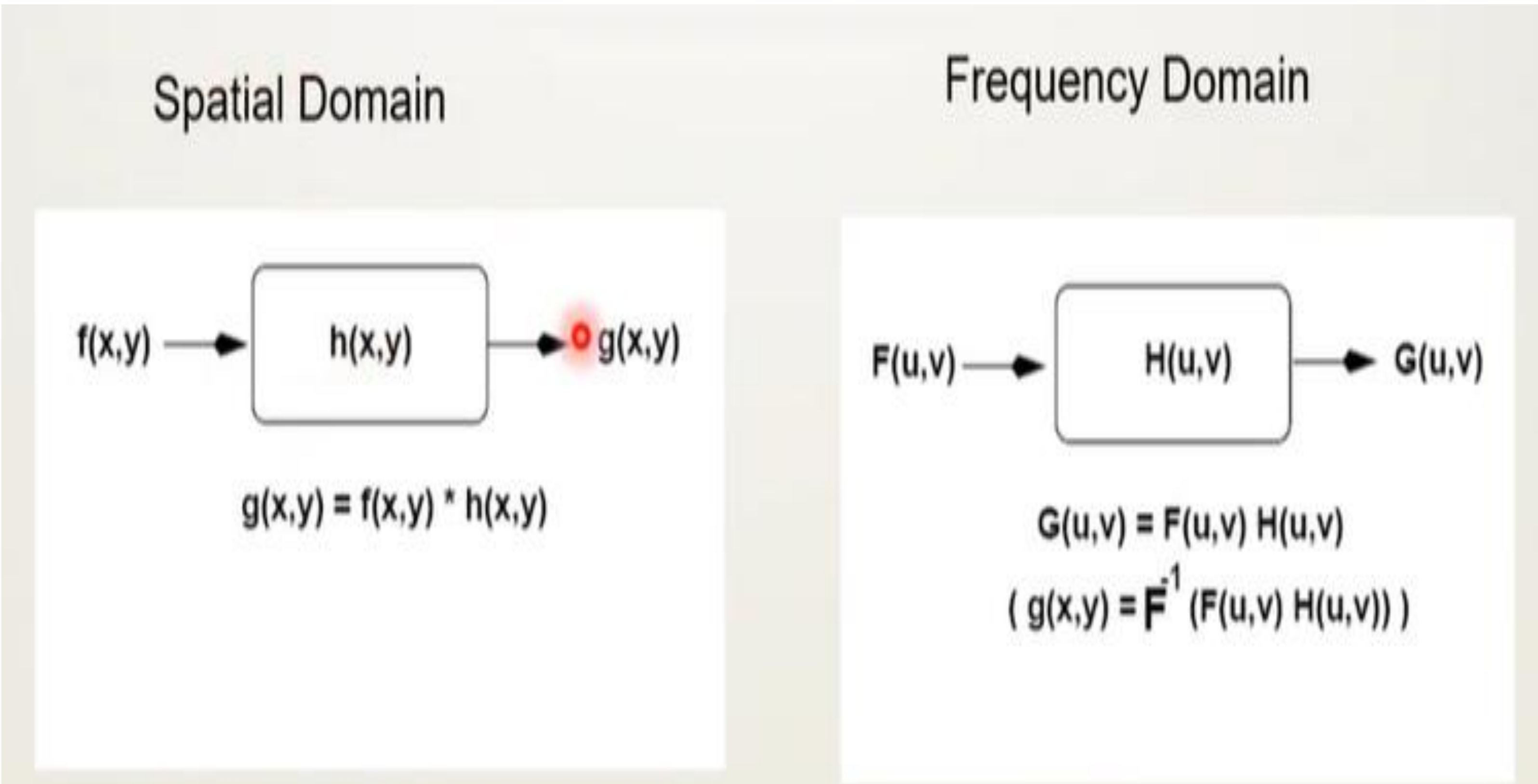
Steps

Steps :-

1. Apply DFT on $f(x,y)$ = (real image)
2. Give us $F(u,v)$ = (DFT image)
   ↳ Remove imaginary part only real value
3. Apply filter on $F(u,v)$
   ↳ create $H(u,v)$
4. $H(u,v)$ Apply mask $I(u,v)$ for the
   ↳ sharpening & smoothing.
5. Give us $G(u,v)$ = (enhance image)

→ $G(u,v)$ apply inverse $F^{-1}(u,v)$ → invers Fourier transform & give real $f(x,y)$ image.

Frequency Domain Methods



↳ Frequency Domain filters :-

(1) low pass :-
:- low Frequency : pass.

(2) High pass :-
:- High frequency pass. / (Aptitute pass)

↳ Frequency domain method :-

(1) Spatial domain :-

$$f(x,y) \longrightarrow \boxed{h(x,y)} \longrightarrow g(x,y)$$

$$g(x,y) = f(x,y) * h(x,y).$$

Major Filter Categories

- Typically, filters are classified by examining their properties in the frequency domain:

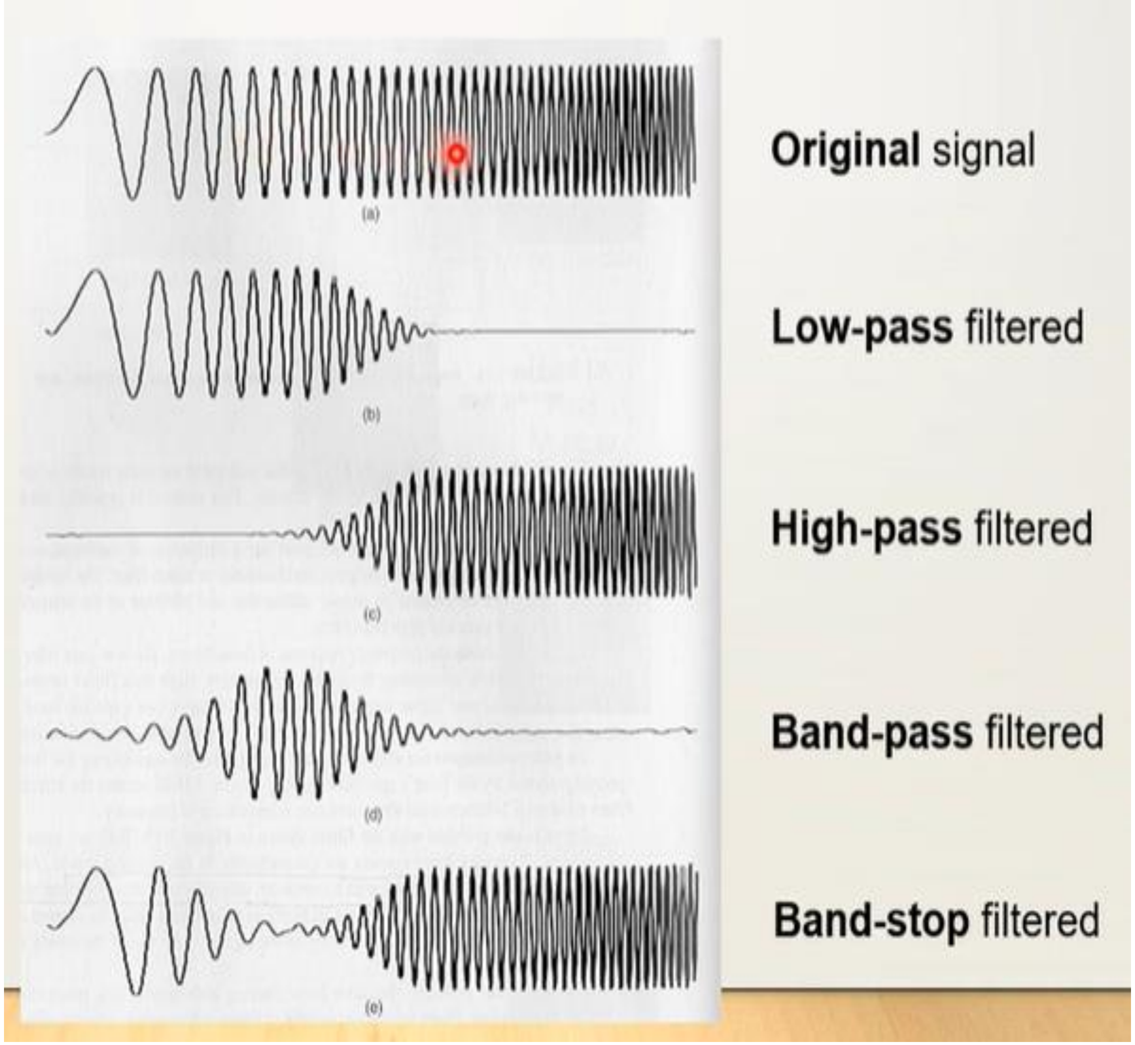
    (1)Low-pass

    (2)High-pass

    (3)Band-pass

    (4)Band-stop

Example

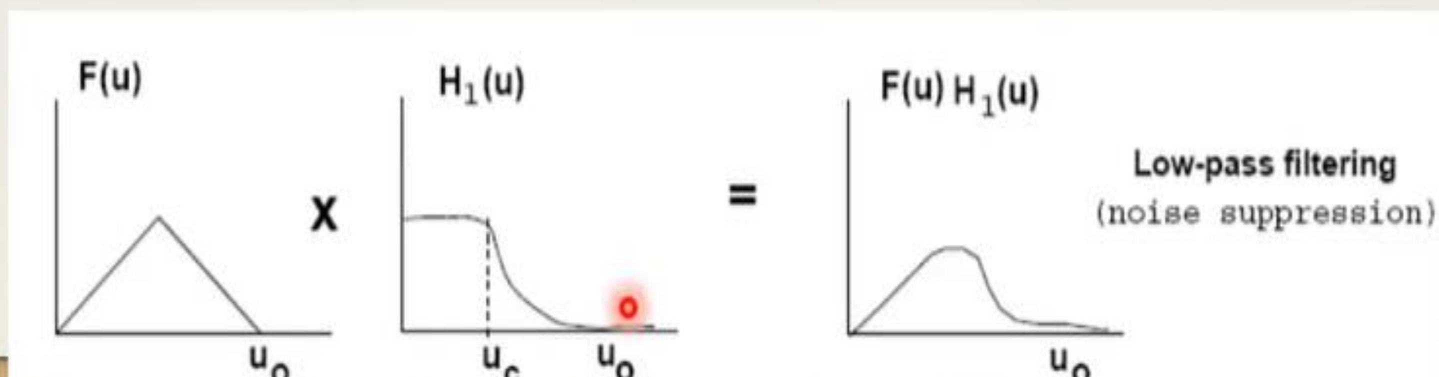


(a)
Original signal
(b)
Low-pass filtered
(c)
High-pass filtered
(d)
Band-pass filtered
(e)
Band-stop filtered

Low-pass filters(i.e., Smoothing filter)

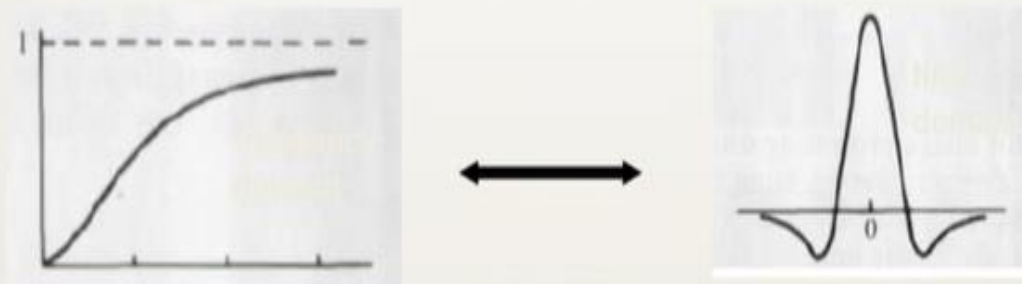
- Preserve low frequencies - useful for noise suppression

Example:

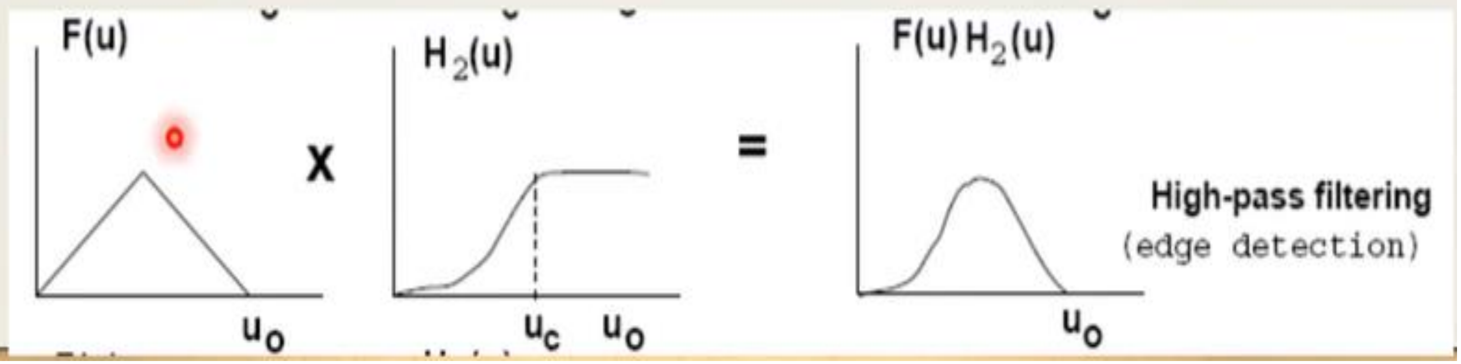


High-pass filters(i.e., Sharpening filters)

- Preserves high frequencies - useful for edge detection

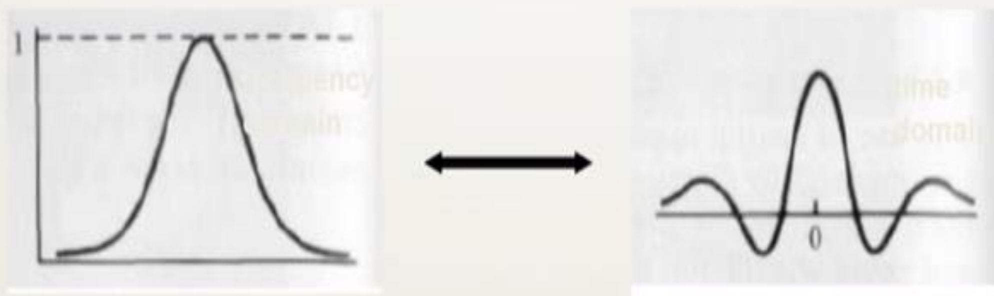


Example:

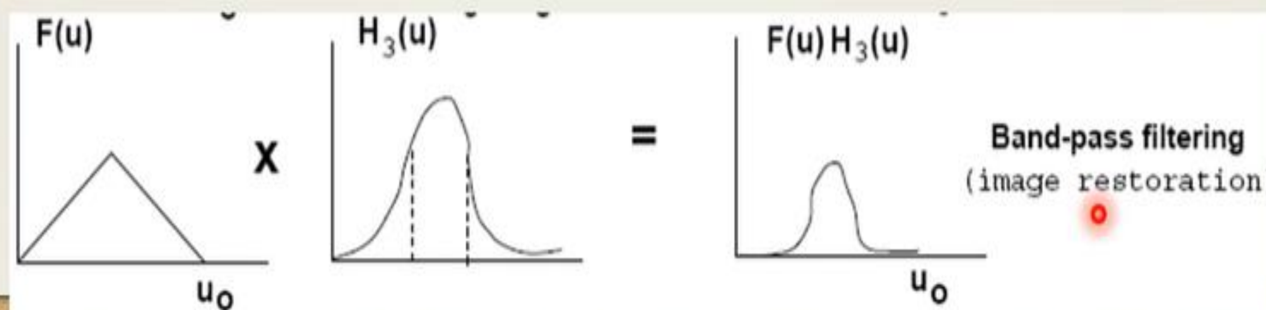


Band-Pass Filters

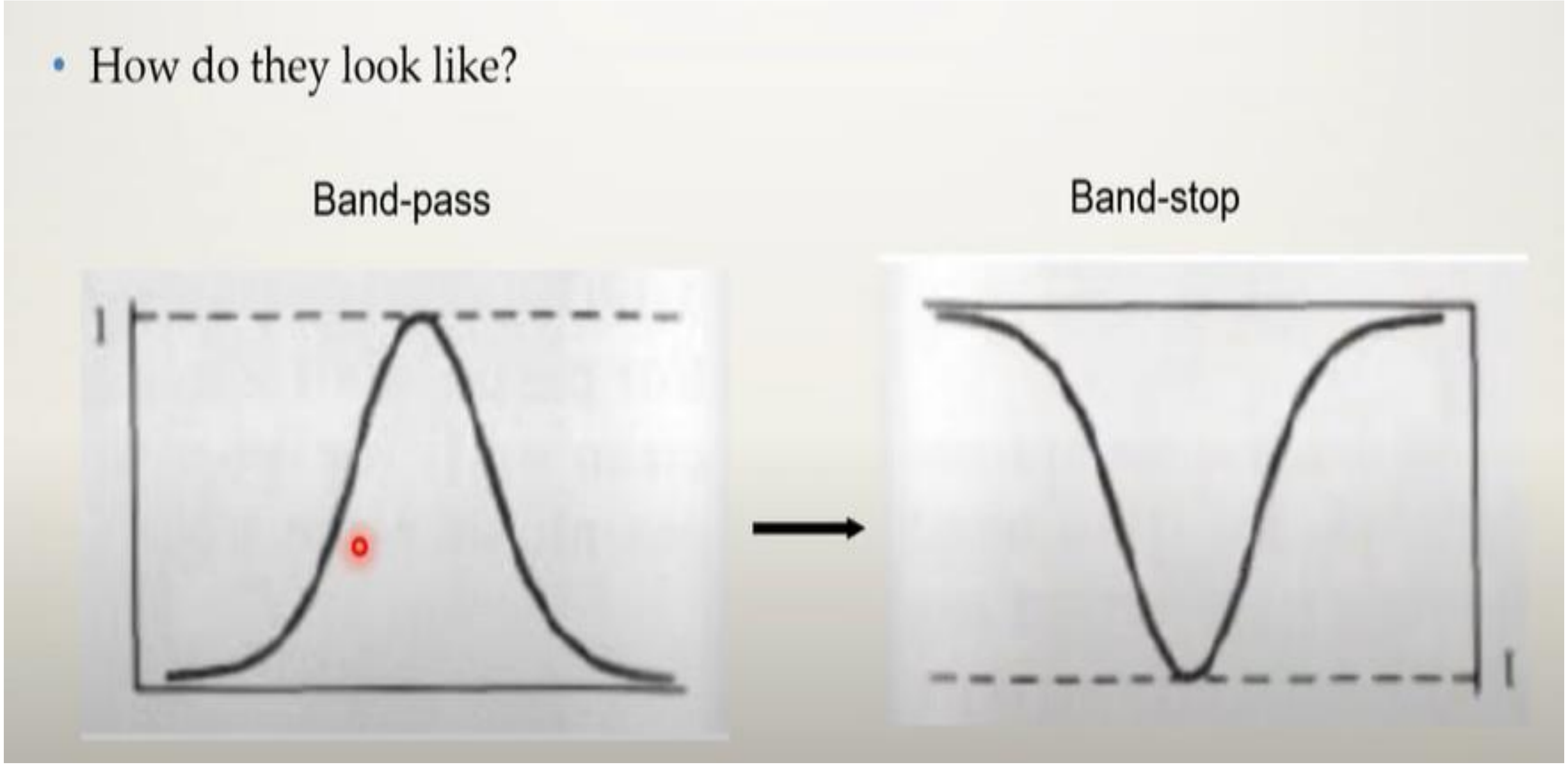
- Preserves frequencies within a certain band



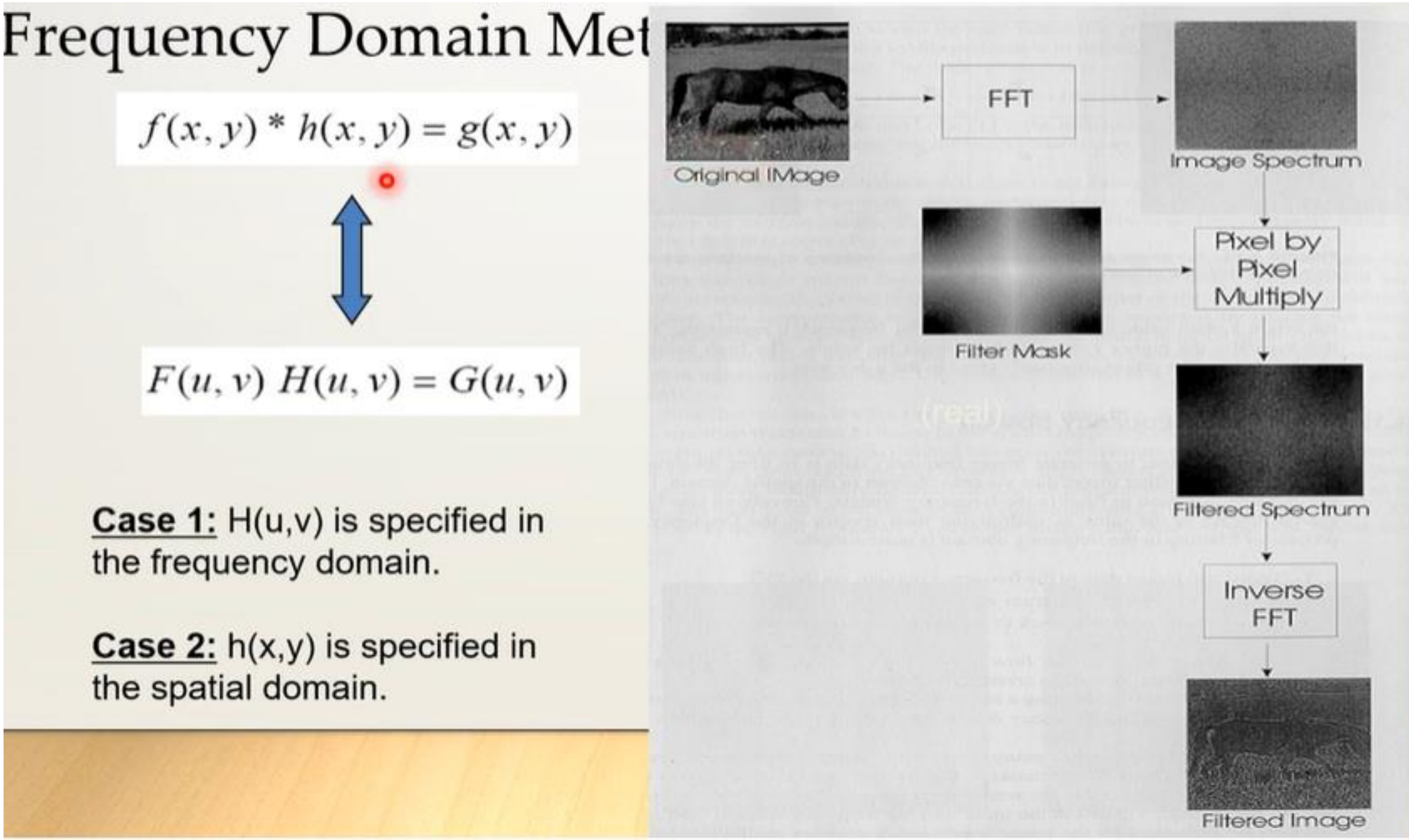
Example:



Band-stop Filters



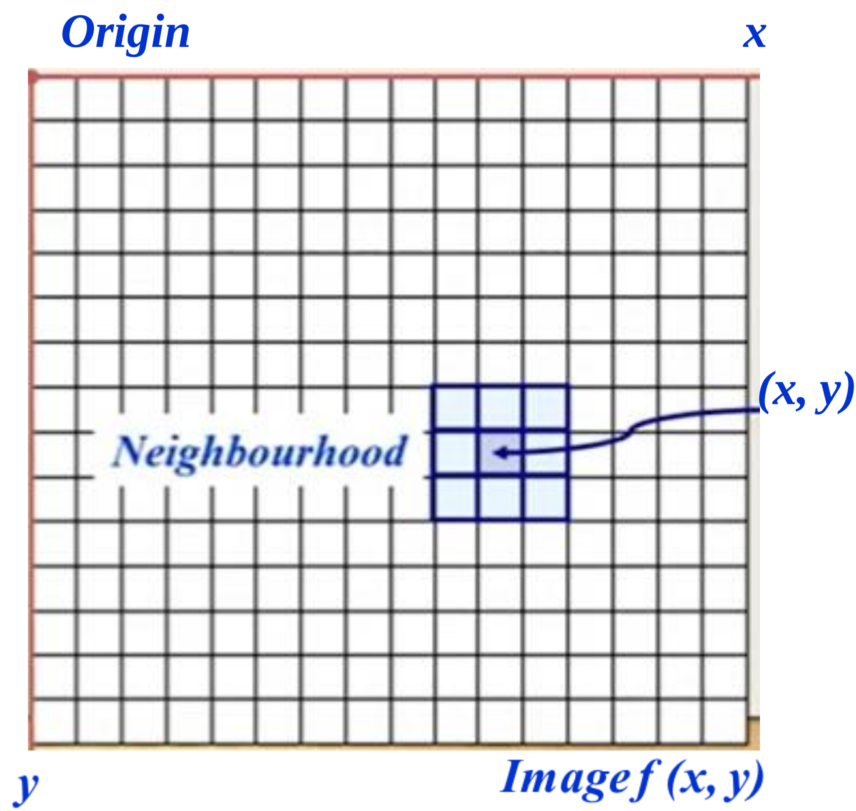
Frequency Domain Method



Filtering

Neighbourhood Operations

- Neighborhood operations simply operate on a larger neighborhood of pixels than point operations
- Neighborhoods are mostly a rectangle around a central pixel
- Any size rectangle and any shape filter are possible

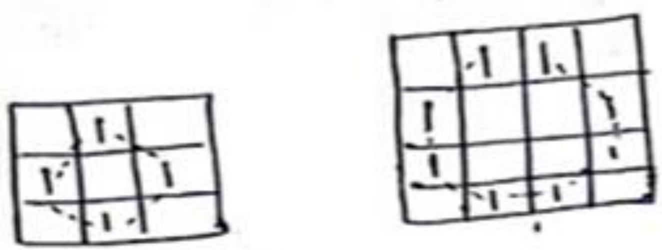


1) Neighbourhood

1) Neighbourhood :-

:- Mask is always in rectangle / square &
any n x m .

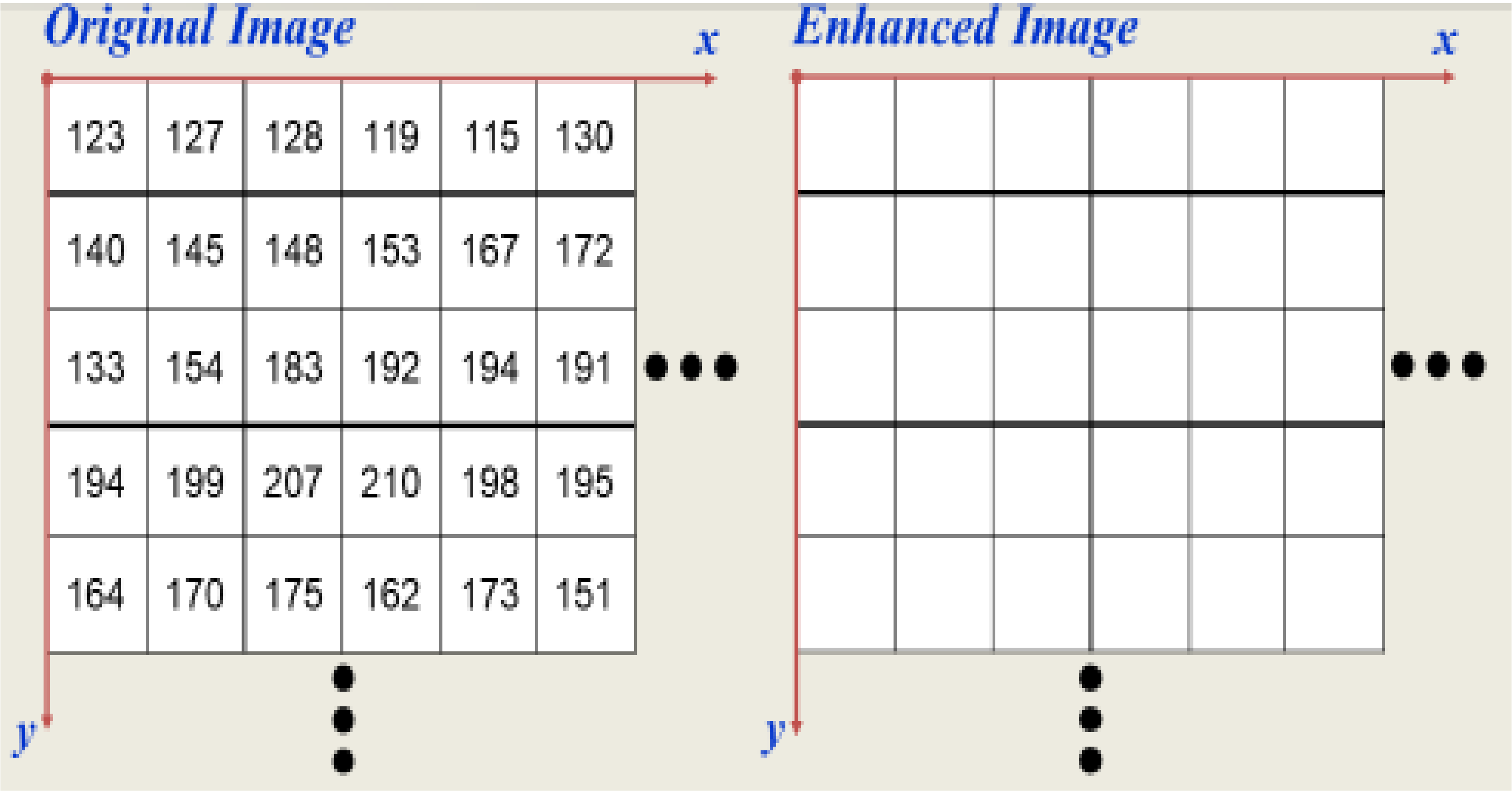
:- Shape



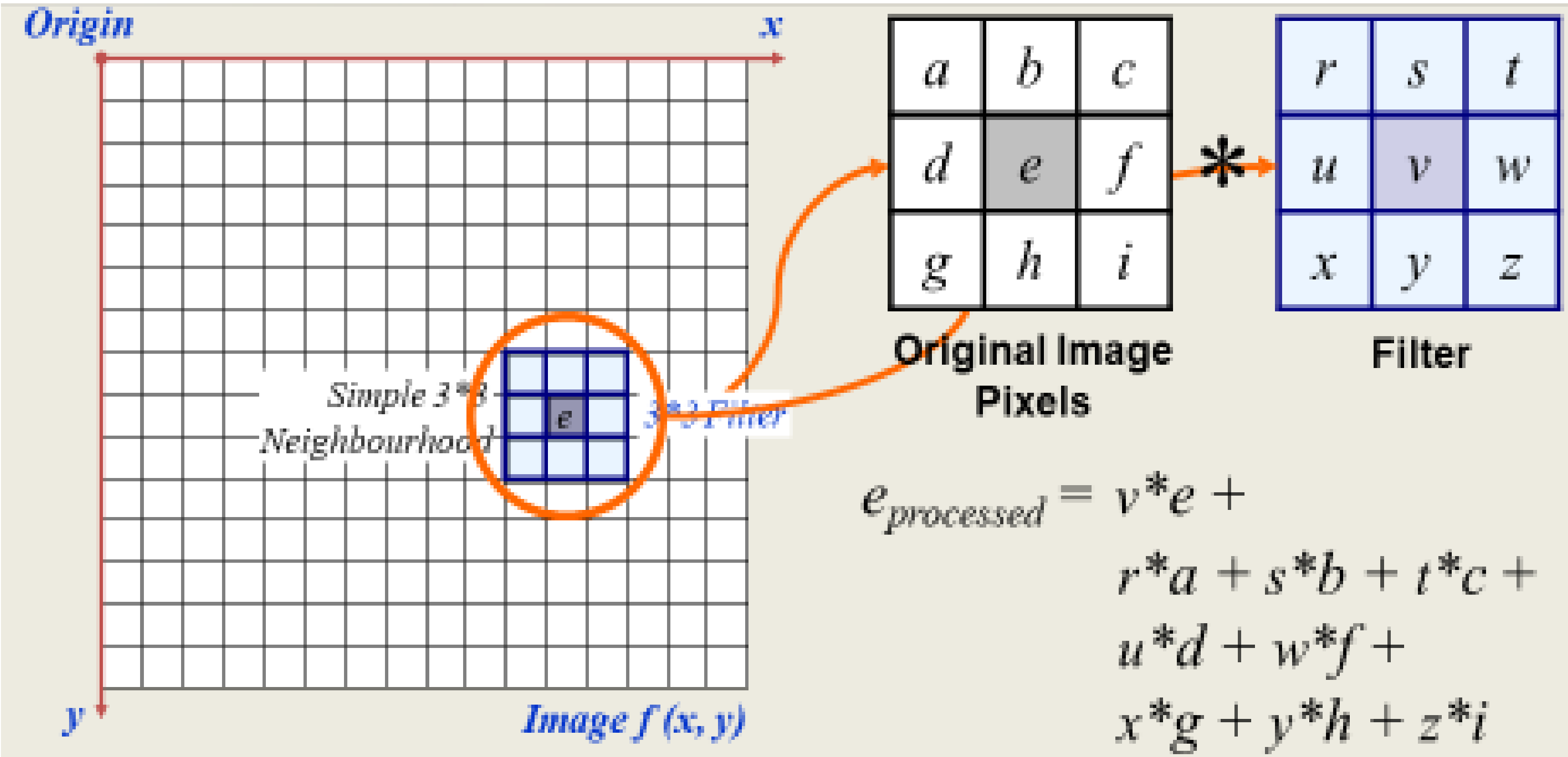
Simple Neighbourhood Operations

- Some simple neighborhood operations include:
  - **Min:** Set the pixel value to the minimum in the neighborhood
  - **Max:** Set the pixel value to the maximum in the neighborhood
  - **Median:** The median value of a set of numbers is the midpoint value in that set (e.g. from the set [1, 7, 15, 18, 24] 15 is the median). Sometimes the median works better than the average

Simple Neighbourhood Operations Example



The Spatial Filtering Process



The above is repeated for every pixel in the original image to generate the filtered image

The Spatial Filtering Process

2) Spatial Filtering Process :-

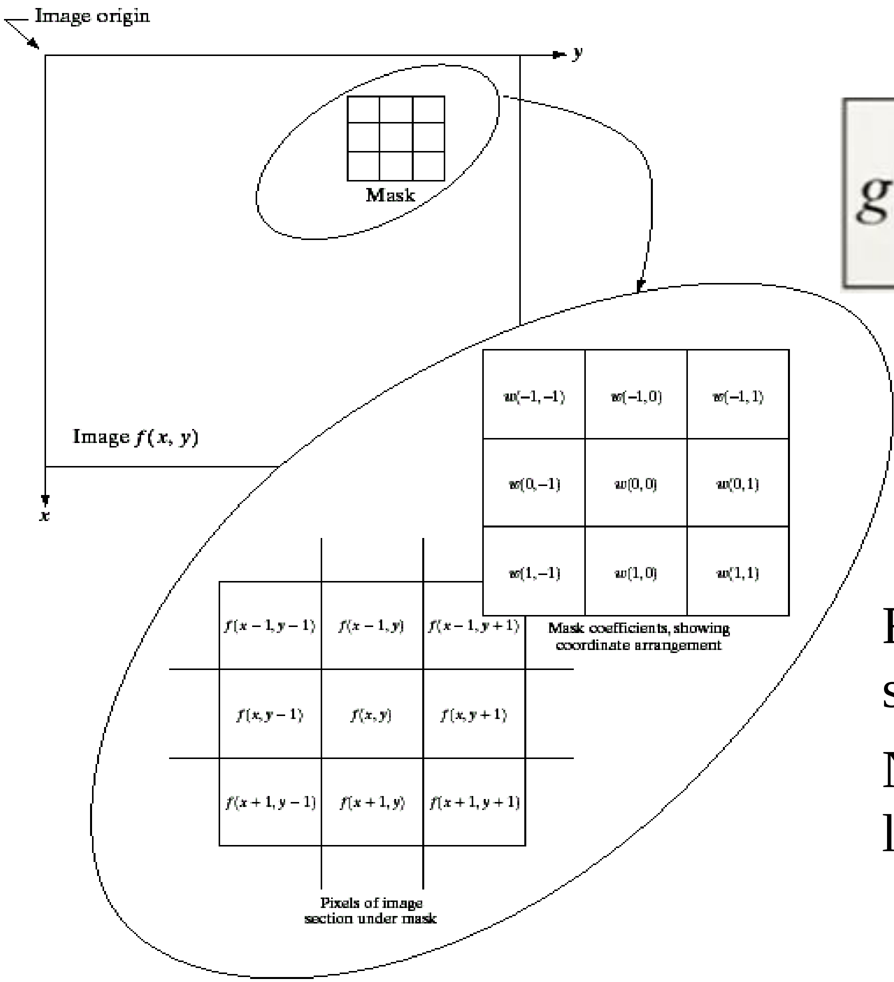
e Process = selected pixel with neighbour + Filter

$$g(x,y) = T \cdot f(x,y)$$

Filtered Image — Mask — Original Image.

$$g(x,y) = \sum_{s=-a} \sum_{s=-b} w(s,t) \cdot f(x+s, y+t)$$

Spatial Filtering: Equation Form



$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$$

Filtering can be given in equation form as shown above

Notations are based on the image shown to the left

Smoothing Spatial Filters

One of the simplest spatial filtering operations we can perform is a smoothing operation

- Simply average all of the pixels in a neighbourhood around a central value
- Especially useful in removing noise from images
- Also useful for highlighting gross detail

$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$
$^1/_9$	$^1/_9$	$^1/_9$

Simple averaging filter

(Box filter)

Smoothing Spatial Filters

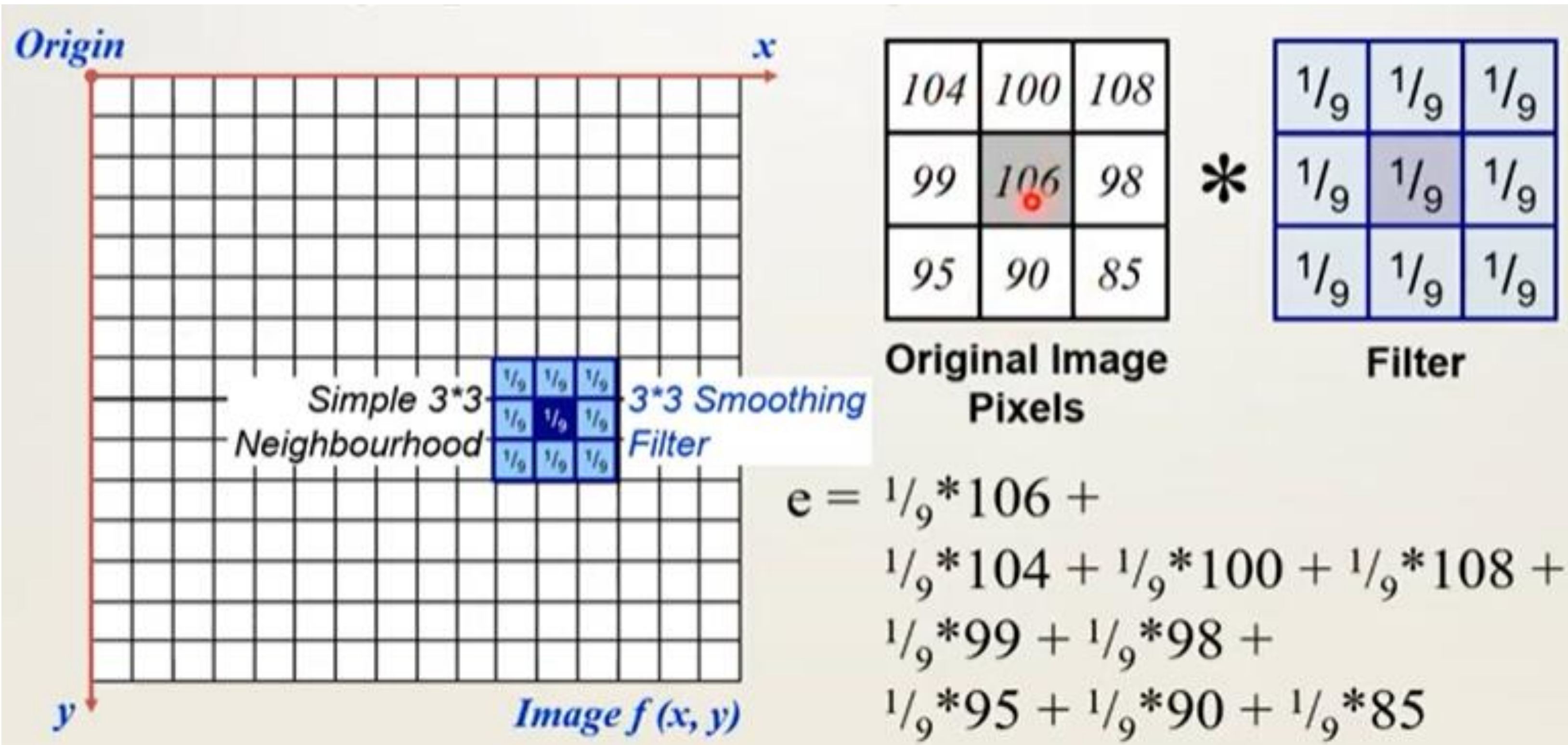
3) Smoothing ~~ff~~ Spatial Filters :-

:- IF we want to derease the pixel value
(divide pixel)

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \xrightarrow[\text{this filter}]{\text{Apply}} \begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

Original
Image

Smoothing Spatial Filtering

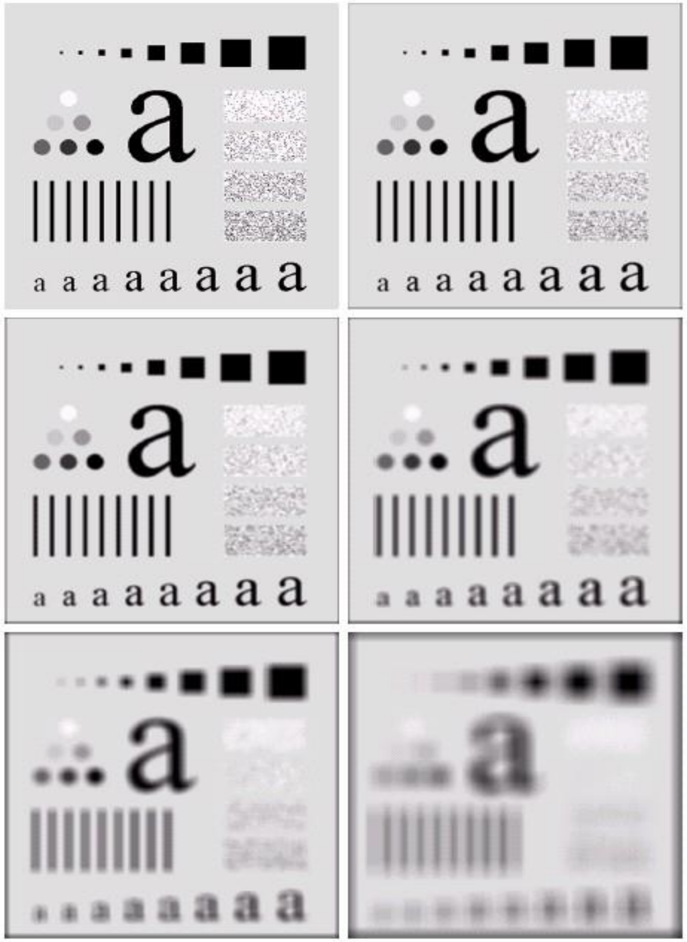


= 98.3333

The above is repeated for every pixel in the original image to generate the smoothed image

Image Smoothing Example

- The image at the top left is an original image of size 500*500 pixels
- The subsequent images show the image after filtering with an averaging filter of increasing sizes
- 3, 5, 9, 15 and 35
- Notice how detail begins to disappear



Filtering
Correlation

Correlation

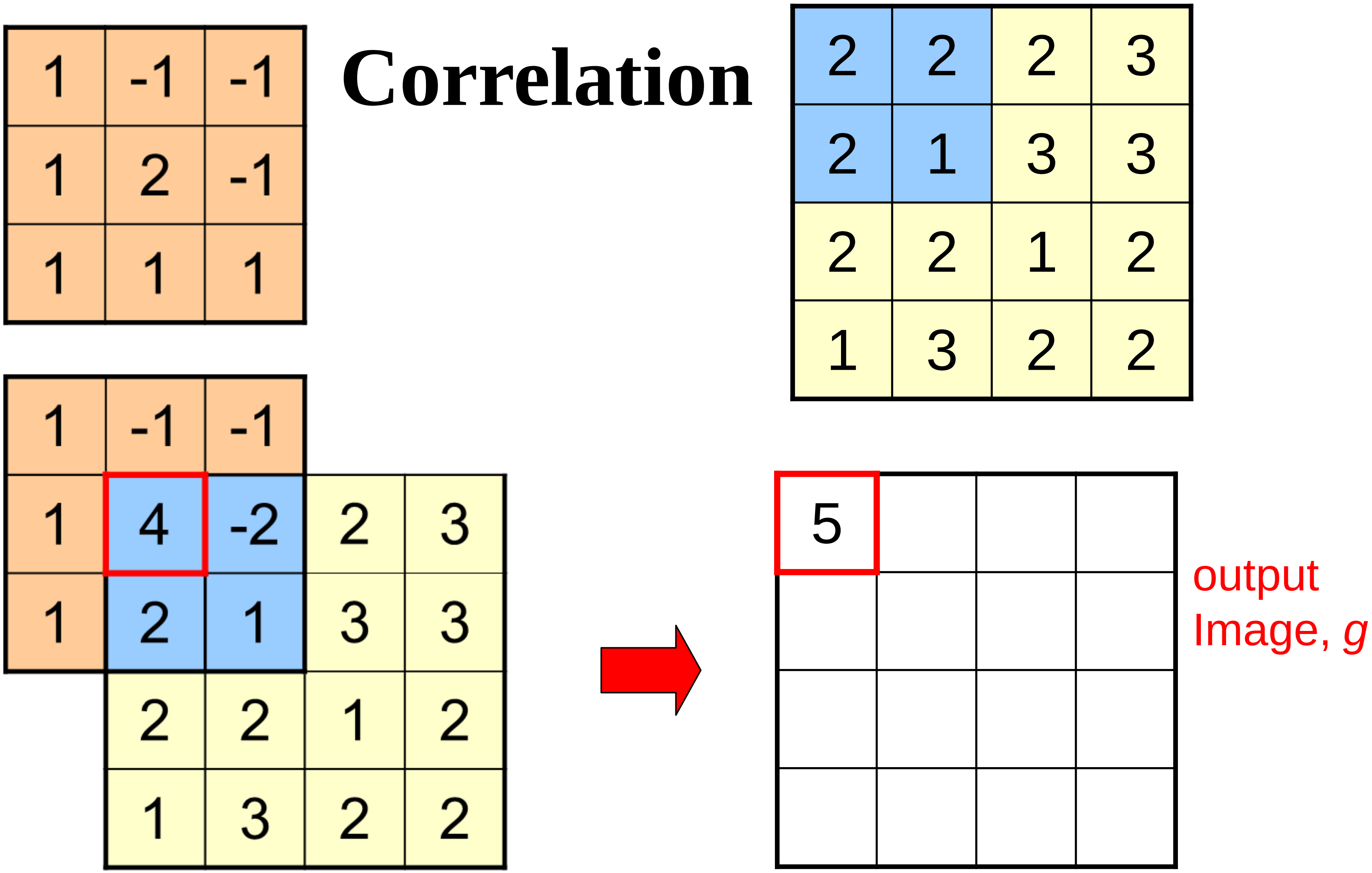
- The equation of correlation is defined as follows :

$$g(x,y)=\sum_{s=-a}^{a}\sum_{t=-b}^{b}\omega(s,t)f(x+s,y+t)$$

$$g=\omega\circ f$$

- It performs dot product between weights and function

Correlation



2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

Original

1	-1	-1
1	2	-1
1	1	1

Mask.

Step:-1.

$$\begin{matrix} \boxed{2} & 2 \\ 2 & 1 \end{matrix} \qquad \begin{matrix} \boxed{2} & -1 \\ 1 & 1 \end{matrix}$$

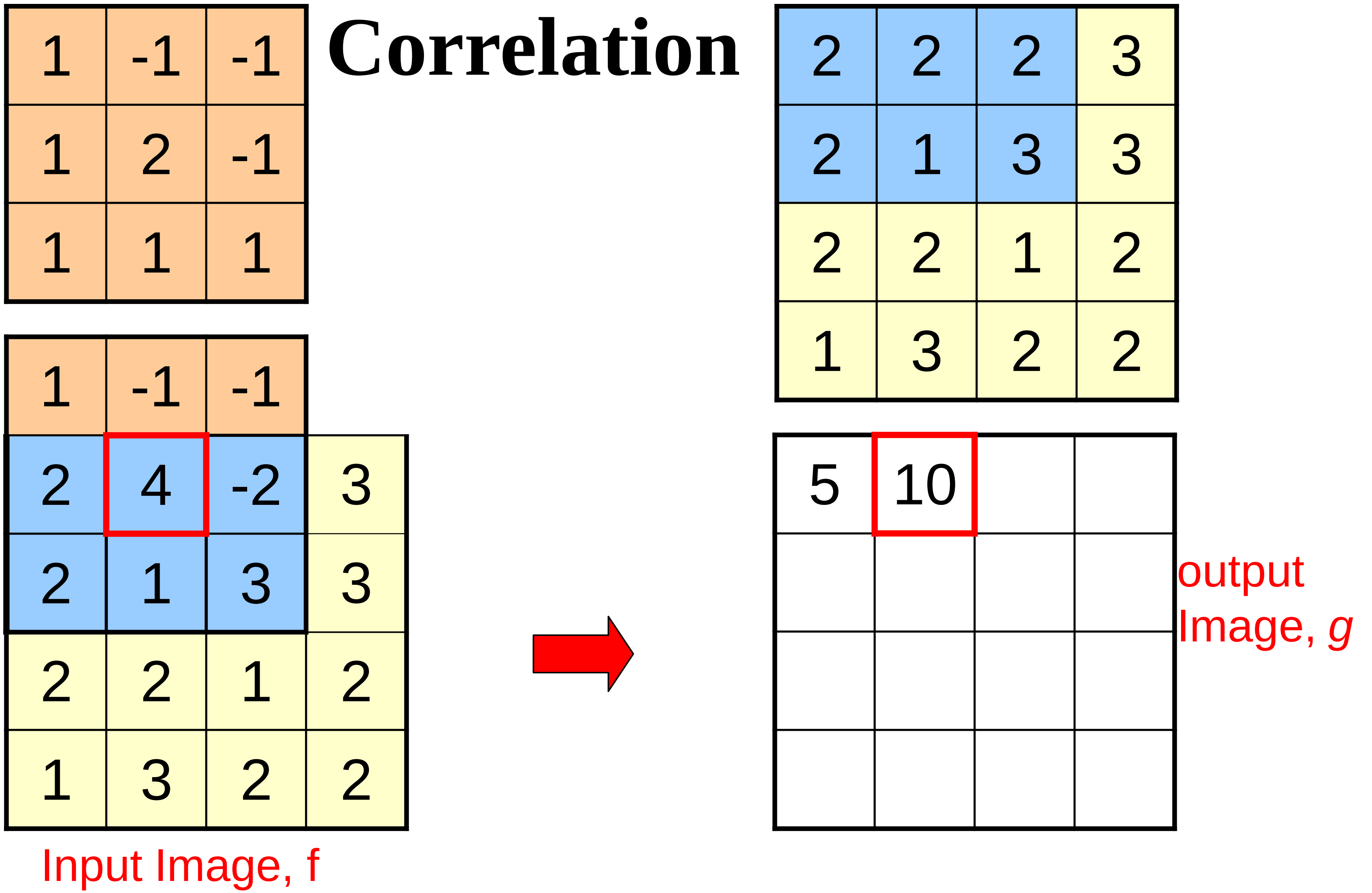
$$= (2)(2) + 2(-1) + 2(1) + 1(1)$$

$$= 4 - \cancel{2} + \cancel{2} + 1$$

$$= 5$$

5			

Correlation



Step:- 2

$$\begin{matrix} 2 & \boxed{2} & 2 \\ 2 & 1 & 3 \end{matrix} \qquad \begin{matrix} 1 & \boxed{2} & -1 \\ 1 & 1 & 1 \end{matrix}$$

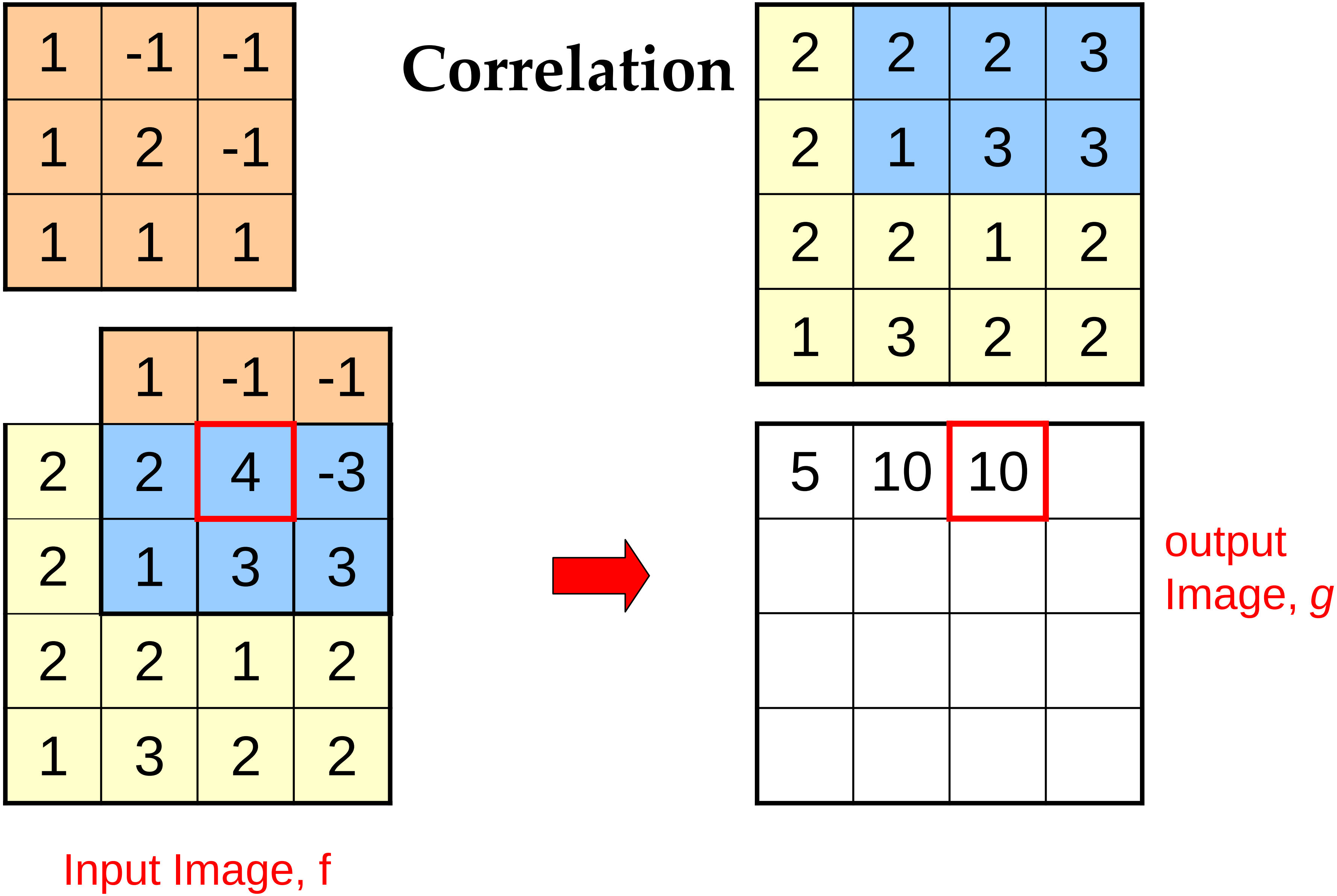
$$= 2(1) + 2(2) + 2(-1) + 2(1) + 1(1) + 3(1)$$

$$= 2 + 4 - \cancel{2} + \cancel{2} + 1 + 3$$

$$= 10$$

5	10		

Correlation



Step:- 3

$$\begin{matrix} 2 & \boxed{2} & 3 \\ 1 & 3 & 3 \end{matrix} \qquad \begin{matrix} 1 & \boxed{2} & -1 \\ 1 & 1 & 1 \end{matrix}$$

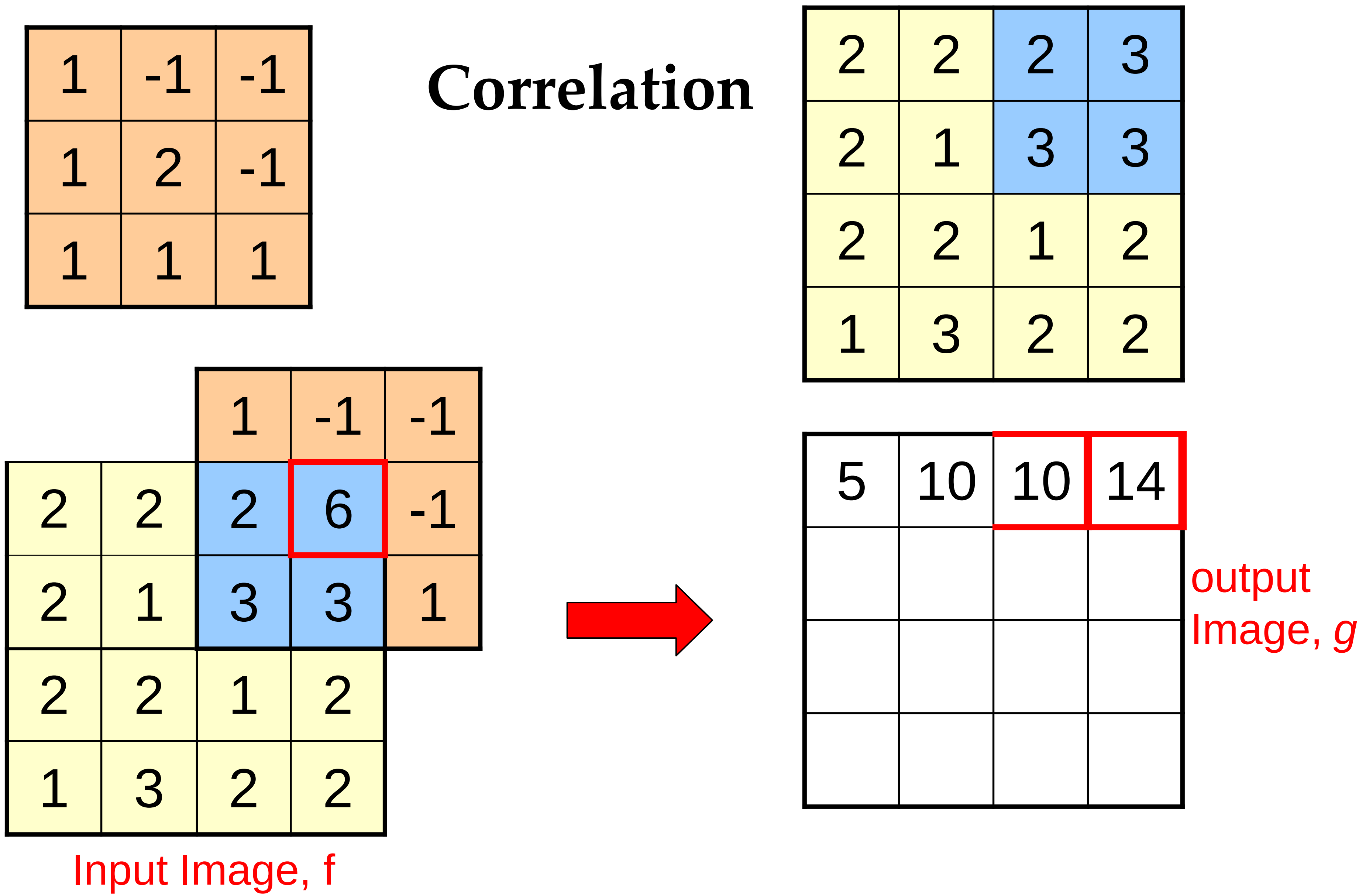
$$= 2(1) + 2(2) + 3(-1) + 1(1) + 3(1) + 3(1)$$

$$= 2 + 4 - \cancel{3} + 1 + \cancel{3} + 3$$

$$= 10$$

5	10	10	

Correlation



Step:-4

$$\begin{matrix} \boxed{2} & 3 \\ 3 & 3 \end{matrix} \qquad \begin{matrix} 2 & \boxed{3} \\ 3 & 3 \end{matrix} \qquad \begin{matrix} 1 & \boxed{2} \\ 1 & 1 \end{matrix}$$

$= 2(1) + 3(2) + 3(1) + 3(1)$

$= 2 + 6 + 3 + 3$

$= 14$

5	10	14	14

Correlation

1	-1	-1
1	2	-1
1	1	1

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

1	-2	-2	2	3
1	4	-1	3	3
1	2	2	1	2
	1	3	2	2

Input Image, f

5	10	10	15
3			

output Image, *g*

Step:- 5

$$\begin{array}{cc} 2 & 2 \\ \boxed{2} & 1 \\ 2 & 2 \end{array} \qquad \begin{array}{cc} -1 & -1 \\ \boxed{2} & -1 \\ 1 & 1 \end{array}$$

$$= 2(-1) + 2(-1) + 2(2) + 1(-1) + 2(1) + 2(1)$$

$$= -\cancel{2} - \cancel{2} + \cancel{4} - 1 + 2 + 2$$

$$= 3$$

5	10	10	14
3			

Correlation

1	-1	-1
1	2	-1
1	1	1

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

2	-2	-2	3
2	2	-3	3
2	2	1	2
1	3	2	2

Input Image, f

5	10	10	15
3	4		

output Image, *g*

Last step

$$\begin{matrix} 2 & 2 & 2 \\ 2 & \boxed{1} & 3 \\ 2 & 2 & 1 \end{matrix} \qquad \begin{matrix} 1 & -1 & -1 \\ 1 & \boxed{2} & -1 \\ 1 & 1 & 1 \end{matrix}$$

$$= 2(1) + 2(-1) + 2(-1) + 2(1) + 1(2) + 3(-1) + 2(1) + 2(1) + 1(1)$$

$$= \cancel{2} - \cancel{2} - \cancel{2} + \cancel{2} + 2 - 3 + 2 + 2 + 1$$

$$= 4$$

6	10	10	14
3	4		

As per this sequence,
we found final corelation O/P for given image.

Correlation

5	10	10	15
3	4	6	11
7	11	4	9
-5	4	4	5

Final output Image, *g*

Filtering Convolution

Correlation & Convolution

- The filtering we have been talking about so far is referred to as *correlation* with the filter itself referred to as the *correlation kernel*
- *Convolution* is a similar operation, with just one subtle difference
- For symmetric filters it makes no difference

a	b	c
d	e	e
f	g	h

Original Image Pixels

$*$

r	s	t
u	v	w
x	y	z

Filter

$$\begin{aligned} e_{processed} = {} & v*e + \\ & z*a + y*b + x*c + \\ & w*d + u*e + \\ & t*f + s*g + r*h \end{aligned}$$

Convolution

- The equation of correlation is defined as follows :

$$g(x,y)=\sum_{s=-a}^{a}\sum_{t=-b}^{b}\omega(s,t)f(x-s,y-t)$$

$$g=\omega * f$$

- It performs cross product between weights and function

Convolution

Convolution kernel, ω

1	-1	-1
1	2	-1
1	1	1

Rotate 180°

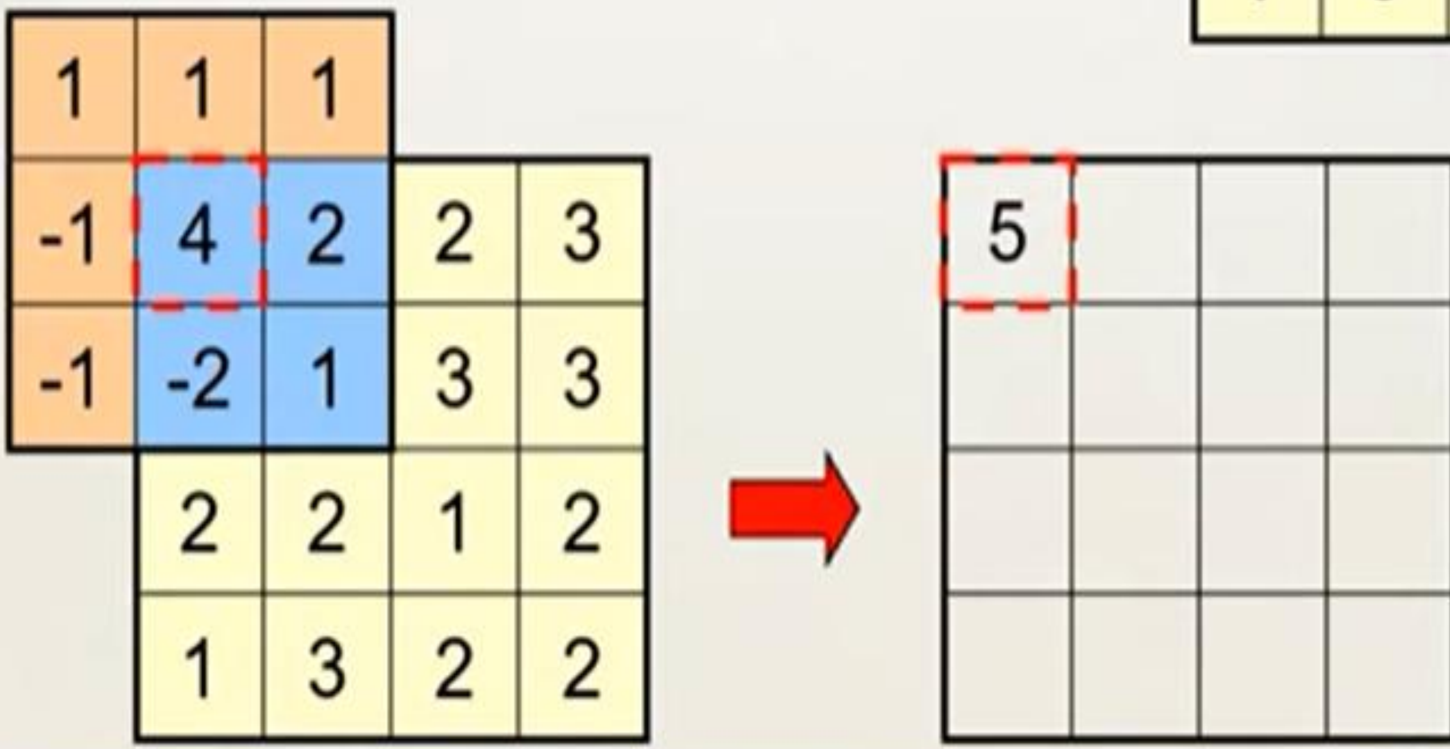
1	1	1
-1	2	1
-1	-1	1

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

Convolution

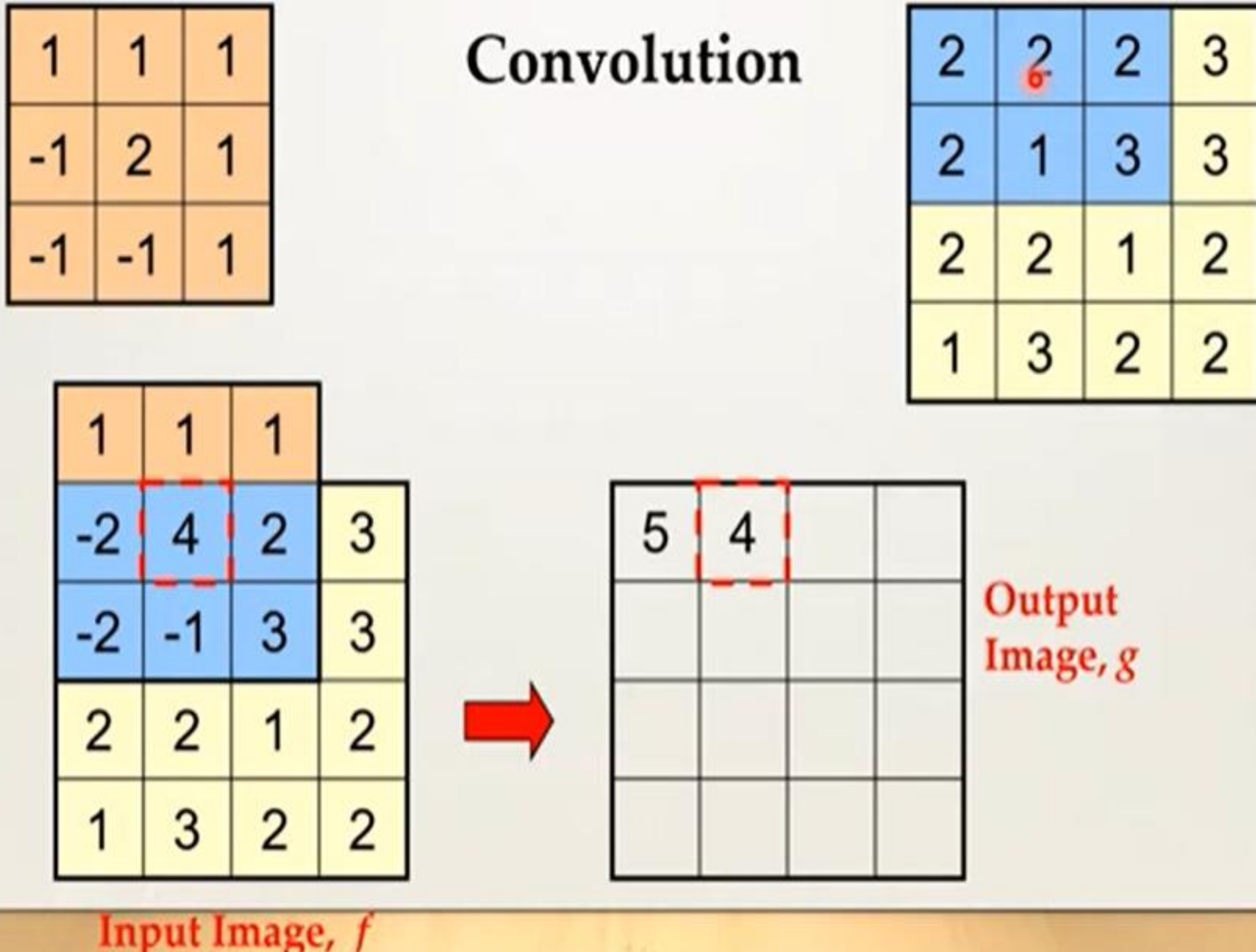
1	1	1
-1	2	1
-1	-1	1

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2



Output Image, g

Input Image, f


Convolution
Input Image, f
Output Image, g

Convolution

1	1	1
-1	2	1
-1	-1	1

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

2	2	2	3
-2	2	3	3
-2	-2	1	2
1	3	2	2

Input Image, *f*

5	4	4	-2
9	6		

Output Image, *g*

Convolution

5	4	4	-2
9	6	14	5
11	7	6	5
9	12	8	5

Final output Image, *g*

Thank You!!!