CTT310: Digital Image Processing

Intensity Transformations and Spatial Filtering

Dr. Nguyen Ngoc Thao
Department of Computer Science, FIT, HCMUS

Outline

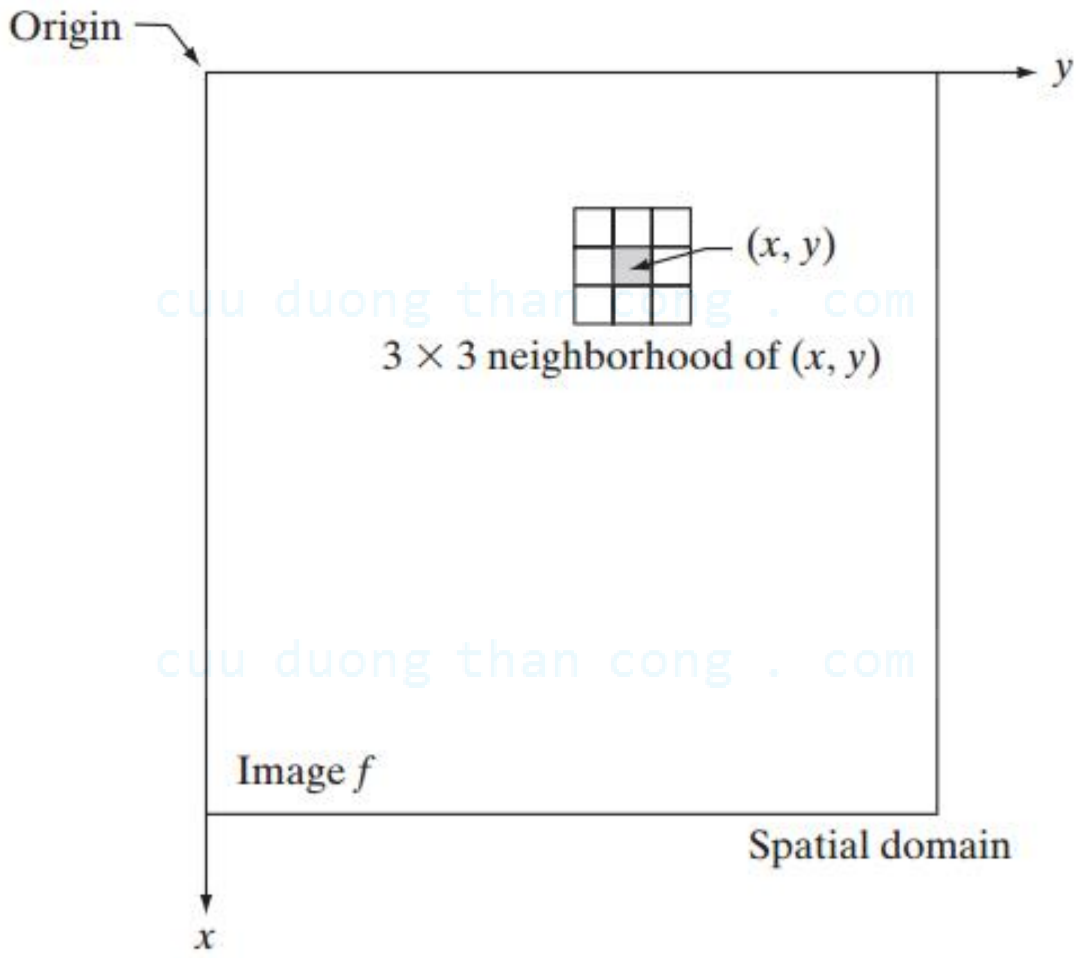
- Background
- Some basic intensity transformation functions
- Histogram processing
- Fundamentals of spatial filtering
- Image enhancement using spatial filters
  - Image smoothing
  - Image sharpening
- Combining spatial enhancement methods

Section 3.1

BACKGROUND

Spatial domain

- **Spatial domain** is the plane containing the pixels of an image, on which spatial domain techniques operate directly.



Spatial filtering

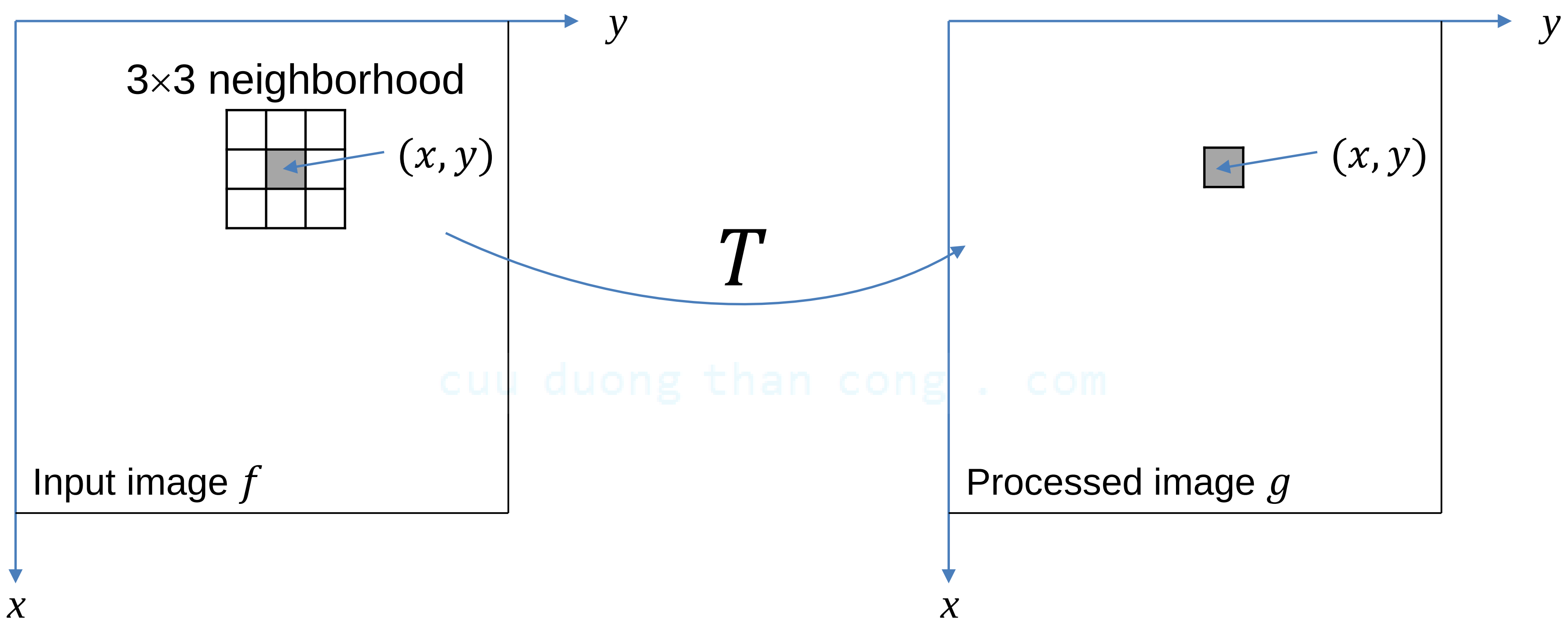
- The **spatial domain process** (or **spatial filtering**) can be denoted by

$$\boldsymbol{g(x,y) = T[f(x,y)]}$$

  - where $f(x,y)$ is the input image, $g(x,y)$ is the output image, and T is an operator on f defined over a neighborhood of point (x,y)

- The neighborhood, along with a predefined operation T, is called a **spatial filter**
  - Other terms: spatial mask, kernel, template, or window
- Spatial domain techniques are computationally efficient and require less processing resources to implement

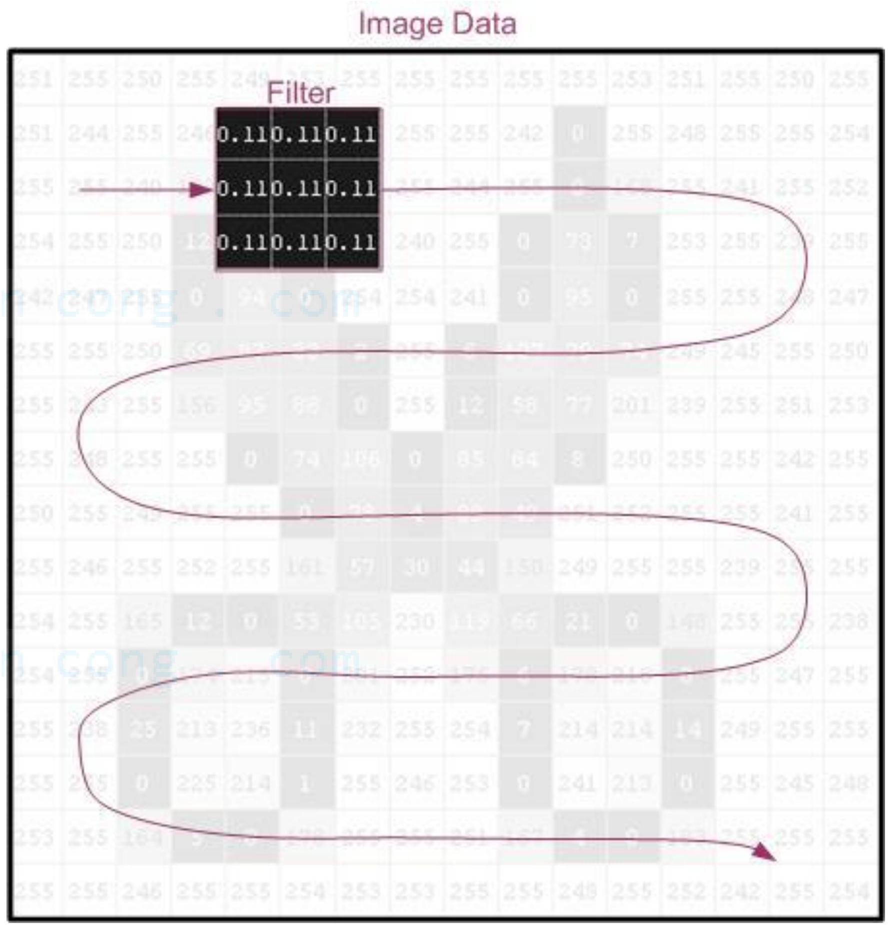
Spatial filtering: An example

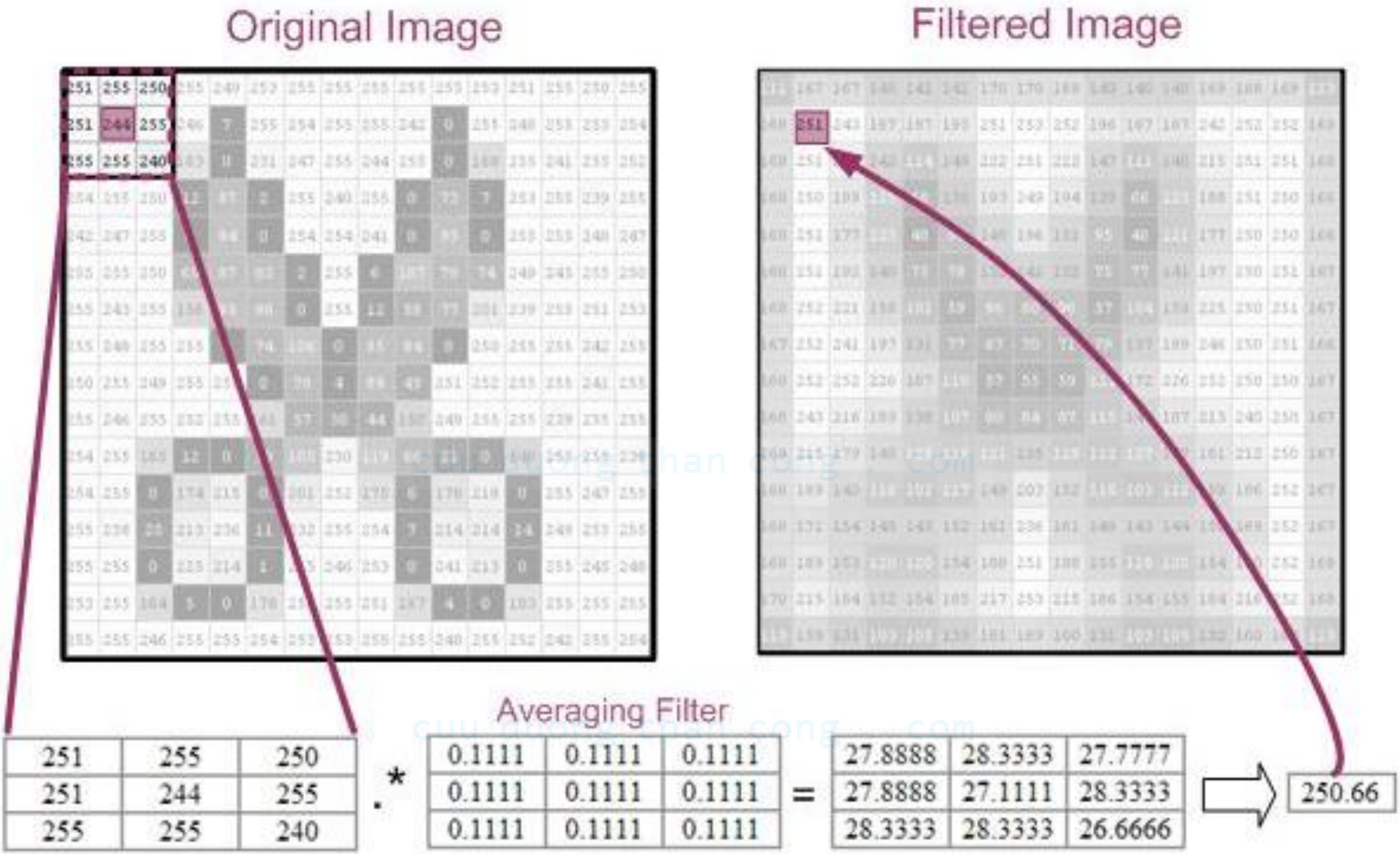


- The neighborhood is a square of size 3×3
- The operator T is defined as "*compute the average intensity of the neighborhood*"

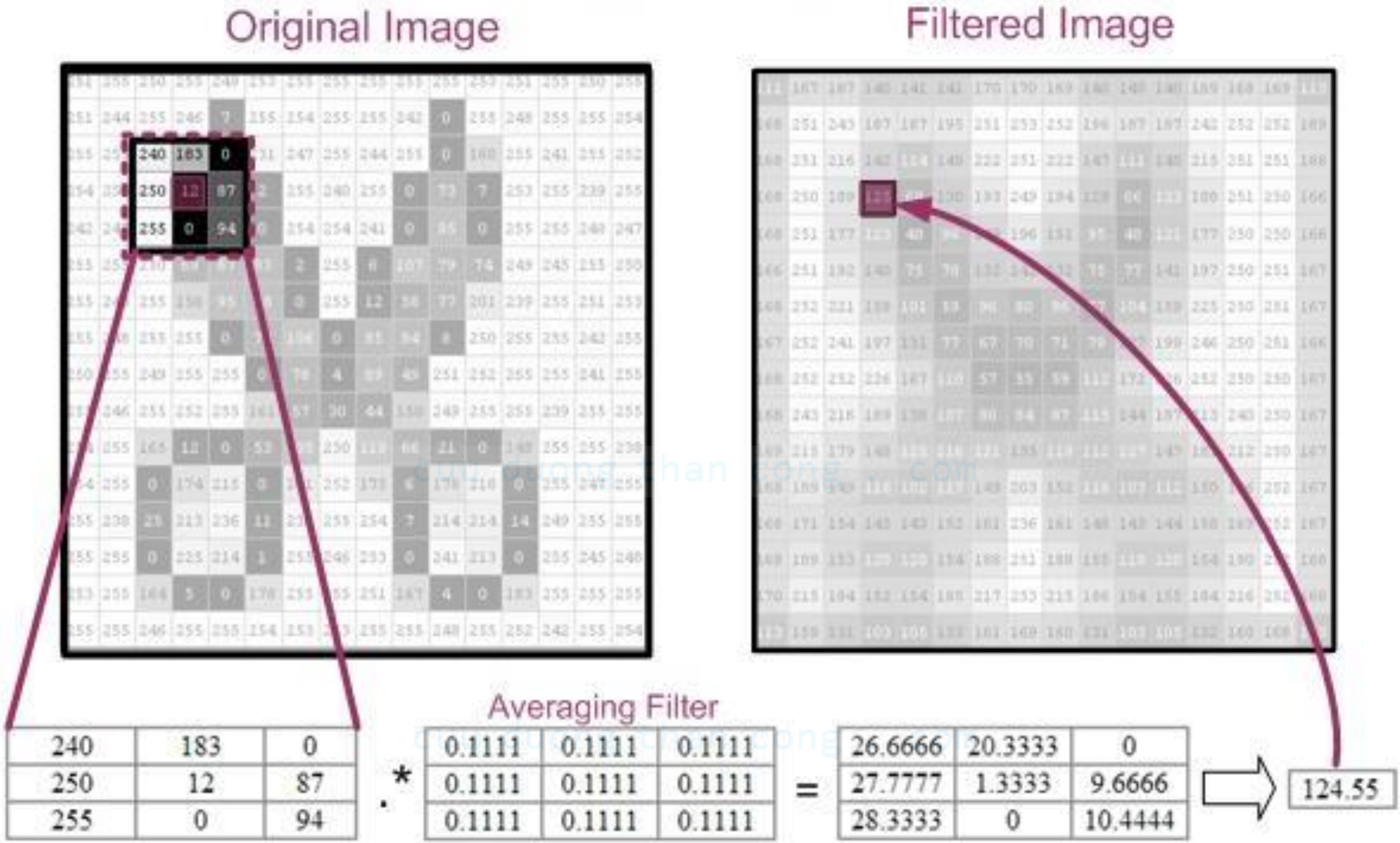
The process of spatial filtering

- Start at the top left of the input image and proceed pixel by pixel in a horizontal scan, one row at a time.
- At each location (x, y), where the origin of the neighborhood is positioned, the operator T uses the pixels in the neighborhood to yield the output at that location





Source: http://www.cs.uregina.ca/Links/class-info/425/Lab3/



Source: http://www.cs.uregina.ca/Links/class-info/425/Lab3/

Section 3.2

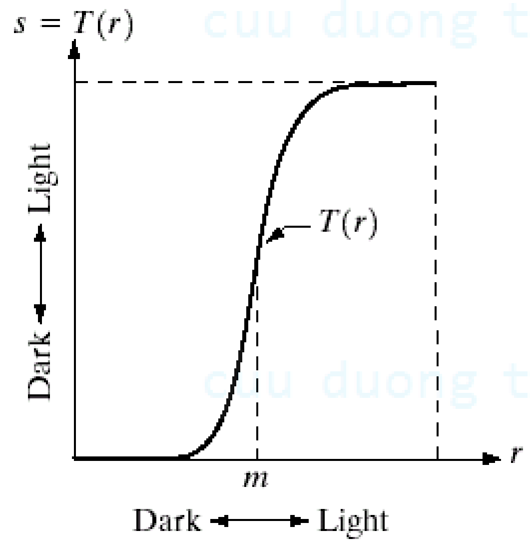
SOME BASIC INTENSITY TRANSFORMATION FUNCTIONS

Intensity transformations

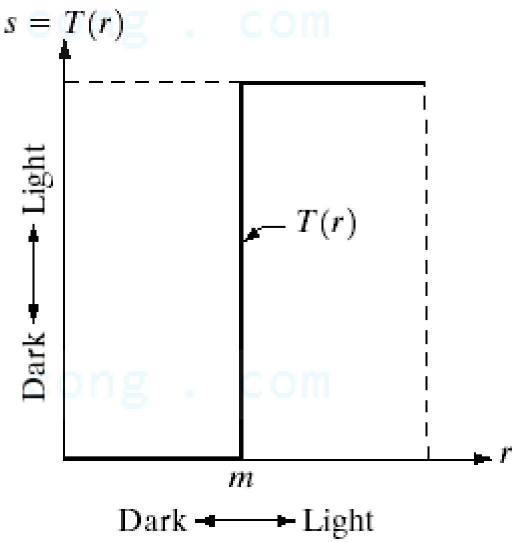
- Among the simplest of all image processing techniques
- The **transformation function T** is of the form

$$s = T(r)$$

  - s and r are the intensity of g and f at any point (x, y), respectively,



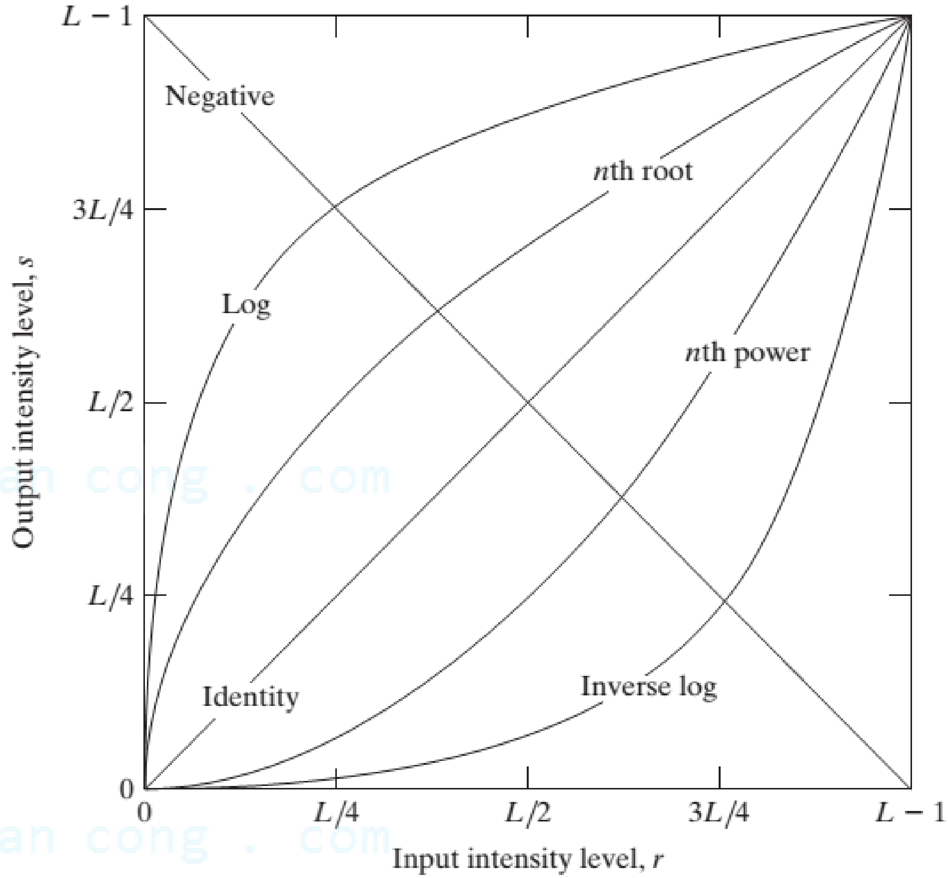
Contrast-stretching function



Thresholding function

Types of transformation functions

- **Linear** (negative and identity transformations)
- **Logarithmic** (log and inverse-log transformations)
- **Power-law** (nth power and nth root transformations)



Some basic intensity transformation functions. All curves were scaled to fit in the range shown.

Image negatives

- The **negative transformation** is given by

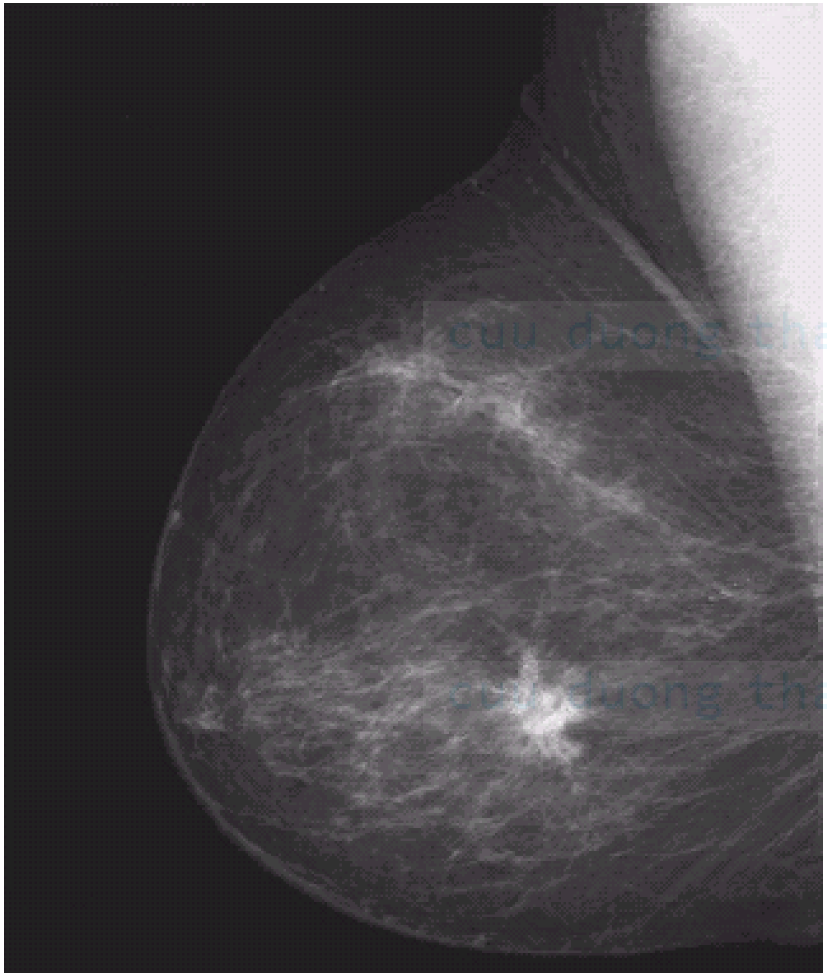
$$\boldsymbol{s = L - 1 - r}$$

  - where the intensity levels r are in the range $[0, L - 1]$

- Reverse the intensity levels of an image to produce the equivalent of a photographic negative
- Particularly suited for enhancing white or gray details embedded in dark regions
  - Especially when the black areas are dominant in size

a b

(a) Original digital mammogram
(b) Negative image obtained using the negative transformation. (Courtesy of G.E. Medical Systems.)



Log transformation

- The **log transformation** is generally given by

$$\boldsymbol{s = clog(1 + r)}$$

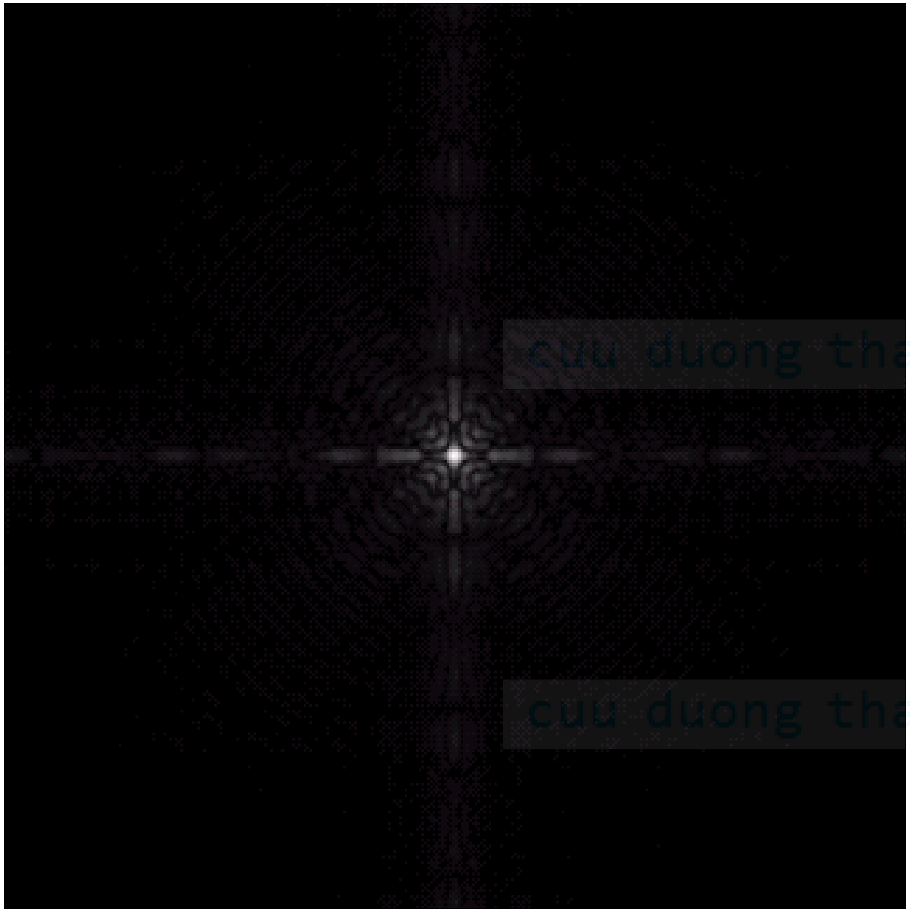
  - where c is a constant, and it is assumed that $r \geq 0$

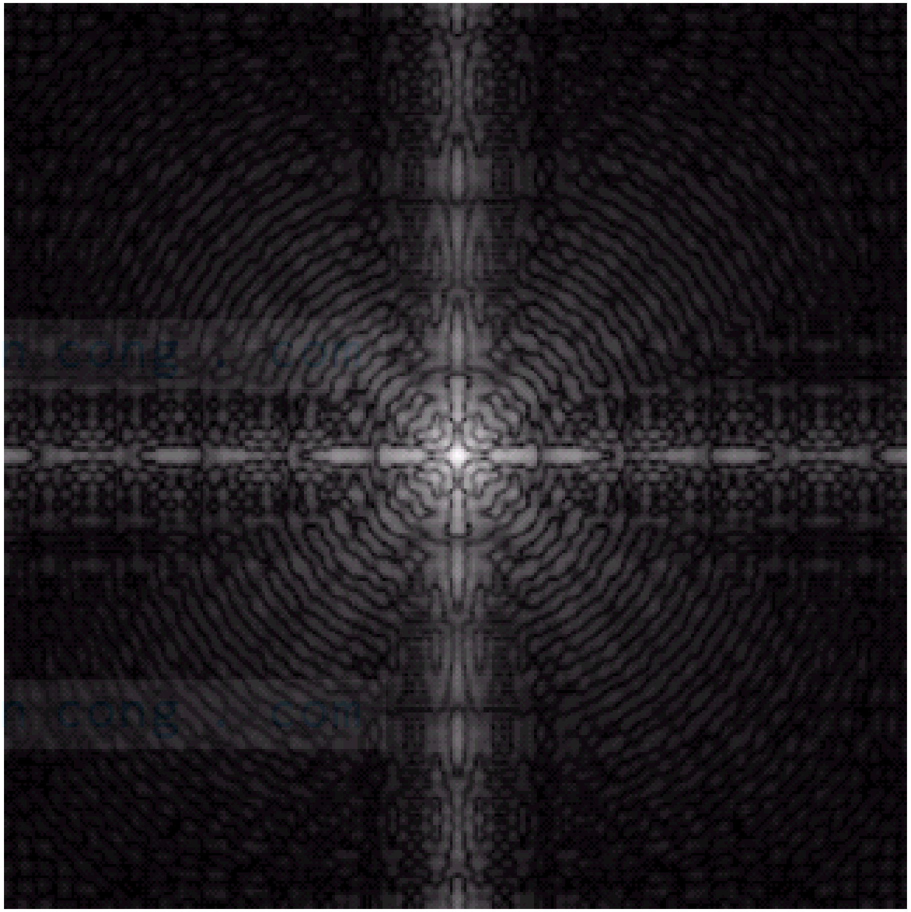
- Map narrow range of low intensity values in the input into a wider range of output levels.
  - The opposite is true of higher values of input levels.
  - The values of dark pixels in an image are expanded while higher-level values are compressed.
- The **inverse log transformation** performs inversely to log transformation

a b

(a) Fourier spectrum.

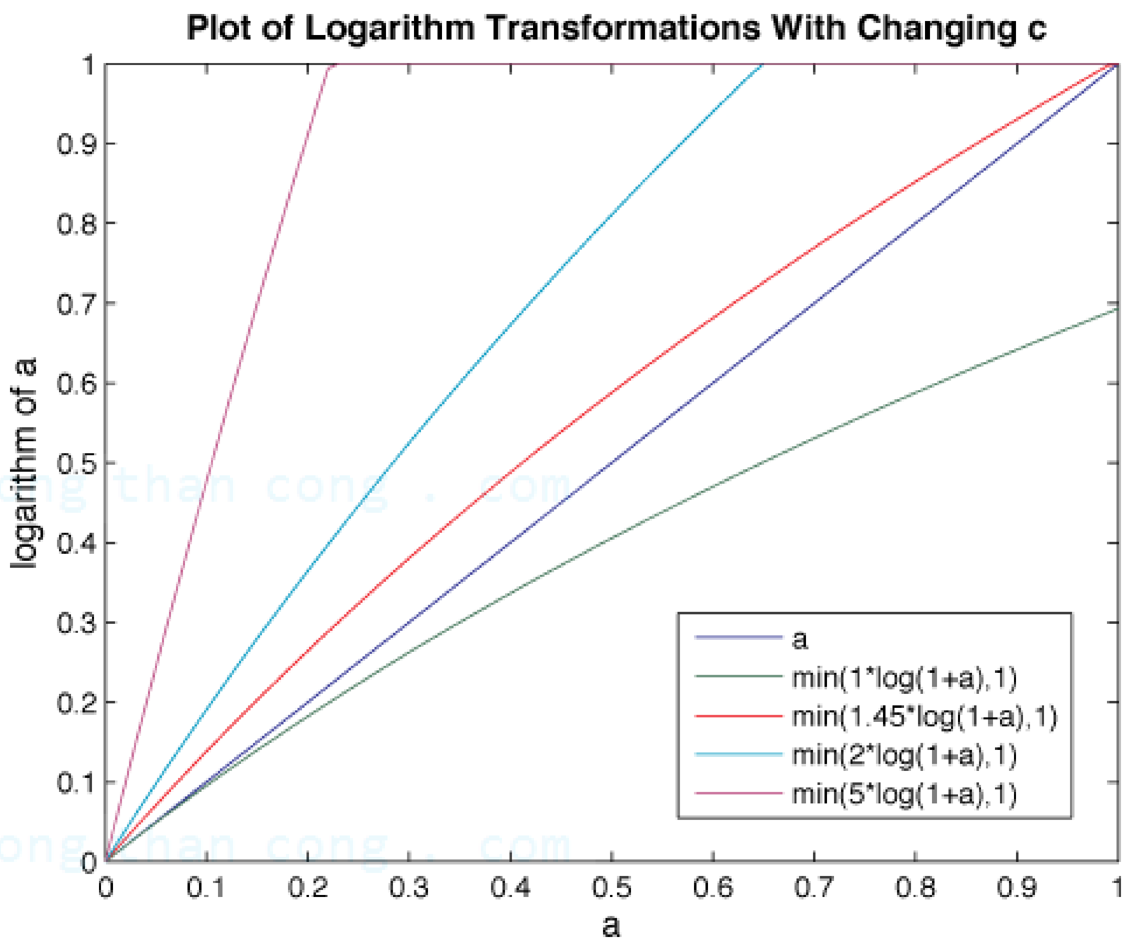
(b) Result of applying the log transformation with c = 1.



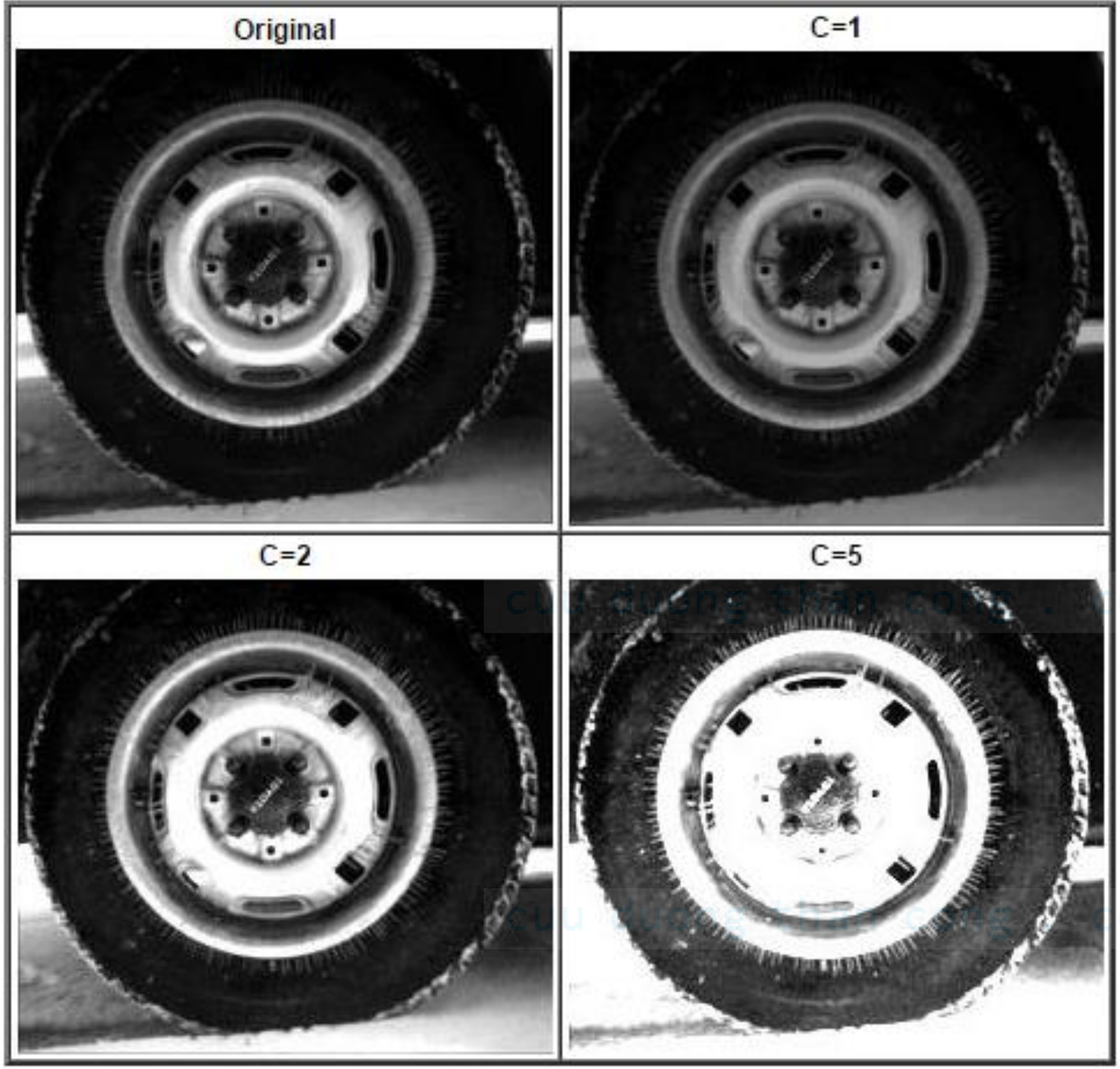


Log transformation: The constant c

- Usually used to scale the range of the log function to match the input domain
  - The higher the c, the brighter the image will appear



Source: http://www.cs.uregina.ca/Links/class-info/425/Lab3/lesson.html

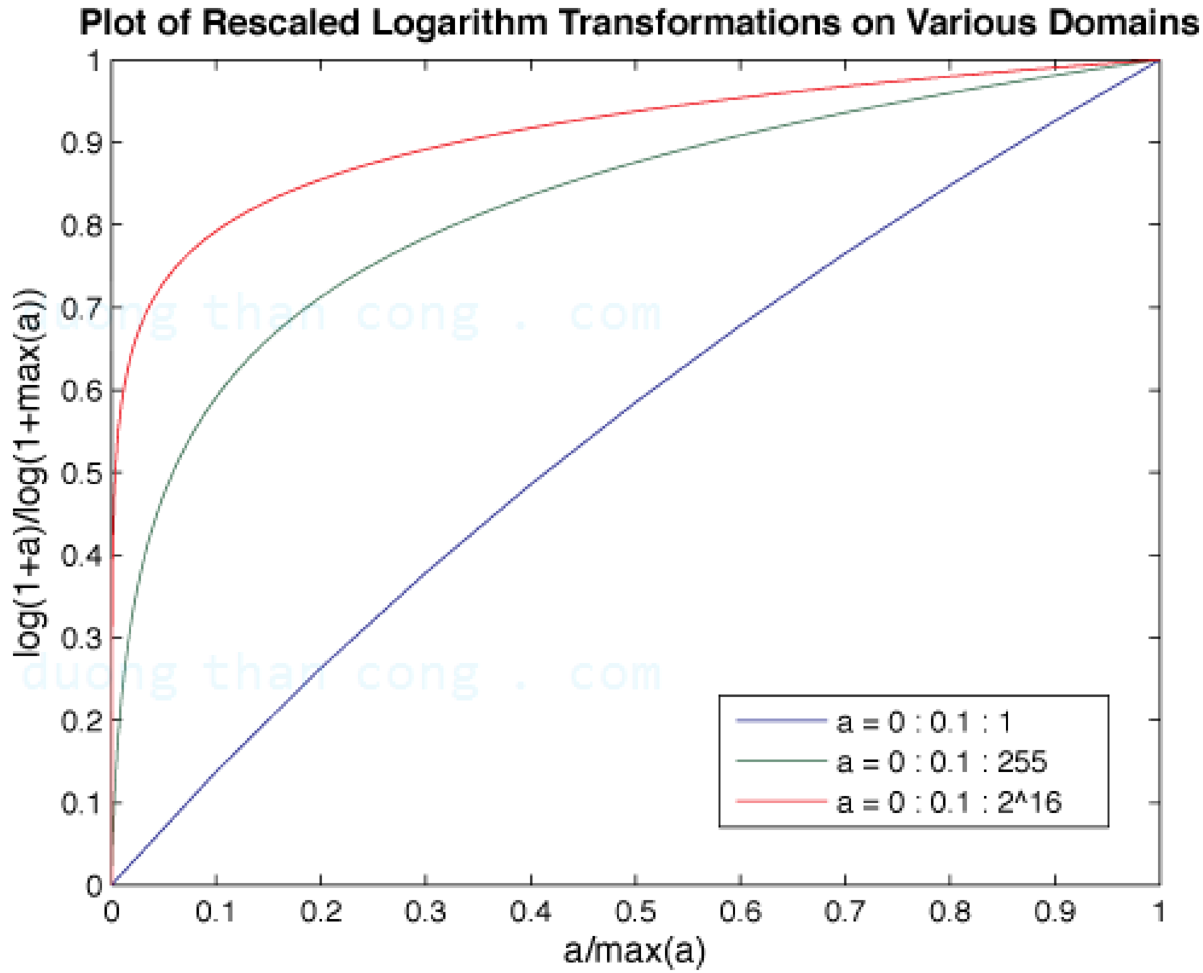


Source: http://www.cs.uregina.ca/Links/class-info/425/Lab3/lesson.html

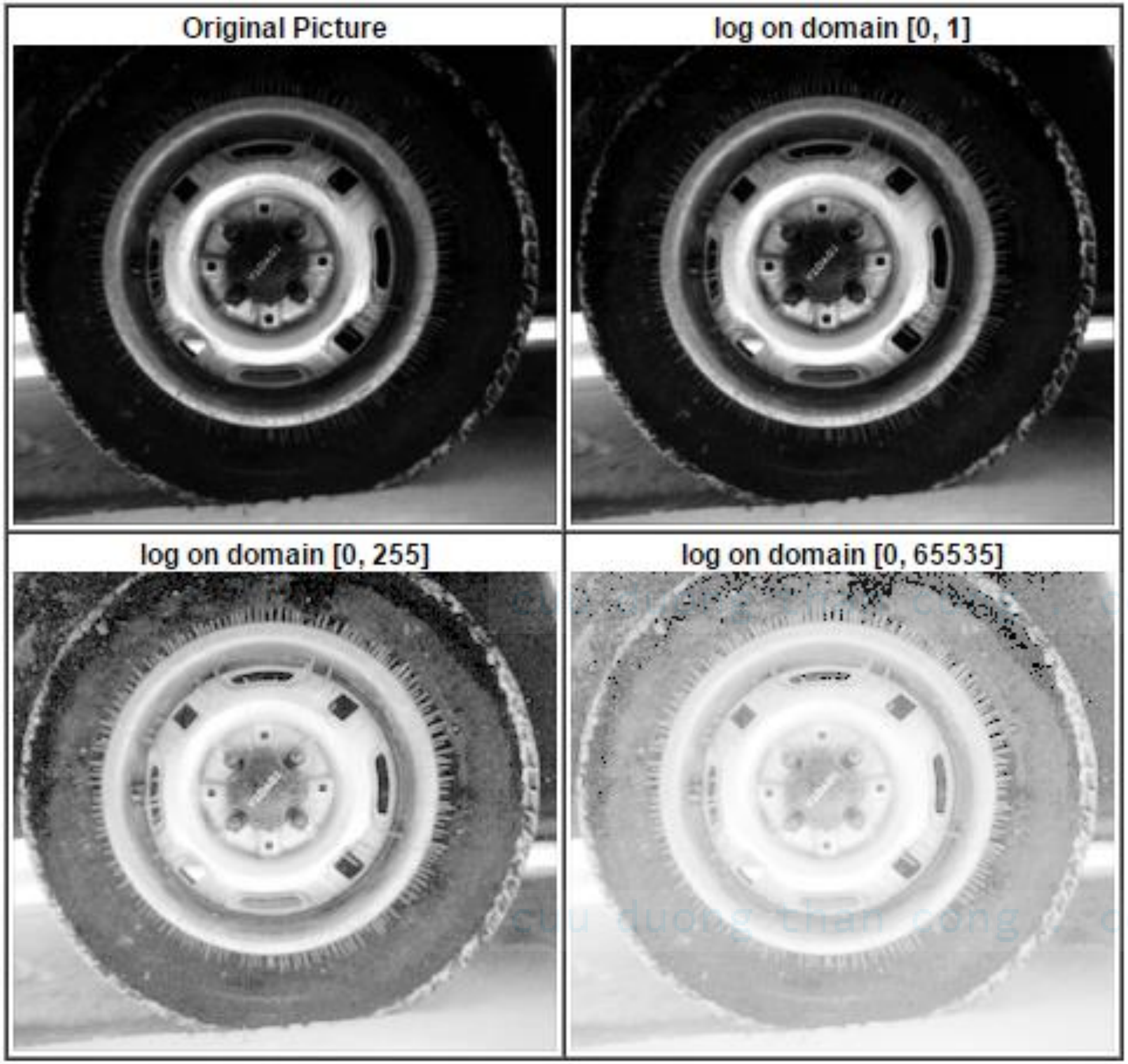
Notice that when c = 5, the image is the brightest and the radial lines on the inside of the tire can be seen easily

Log transformation: The range of values

- The shape of the curve depends on the range of values it is applied to.



Source: http://www.cs.uregina.ca/Links/class-info/425/Lab3/lesson.html



Source: http://www.cs.uregina.ca/Links/class-info/425/Lab3/lesson.html

Unlike with linear scaling and clamping, gross detail is still visible in light areas

Power-law (Gamma) transformation

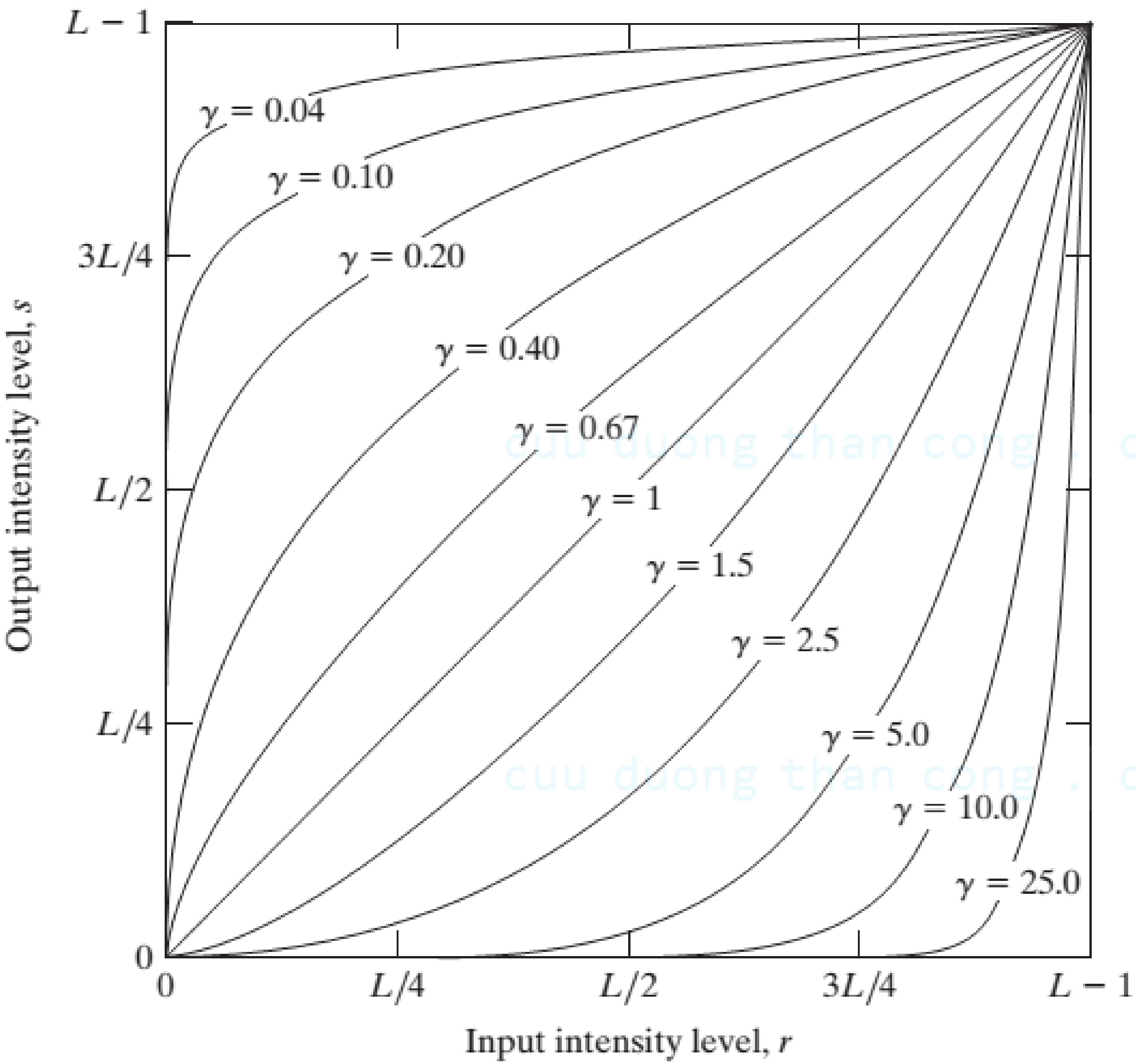
- The **power-law transformations** is given by

$$s = cr^{\gamma}$$

  - where c and γ are positive constants

- Map a narrow range of dark input values into a wider range of output values
  - The opposite is true for higher values of input levels
- Varying γ produces a family of transformation curves.
  - Curves generated with values of $\gamma > 1$ have exactly the opposite effect as those generated with values of $\gamma < 1$

Power-law transformation: the constant γ



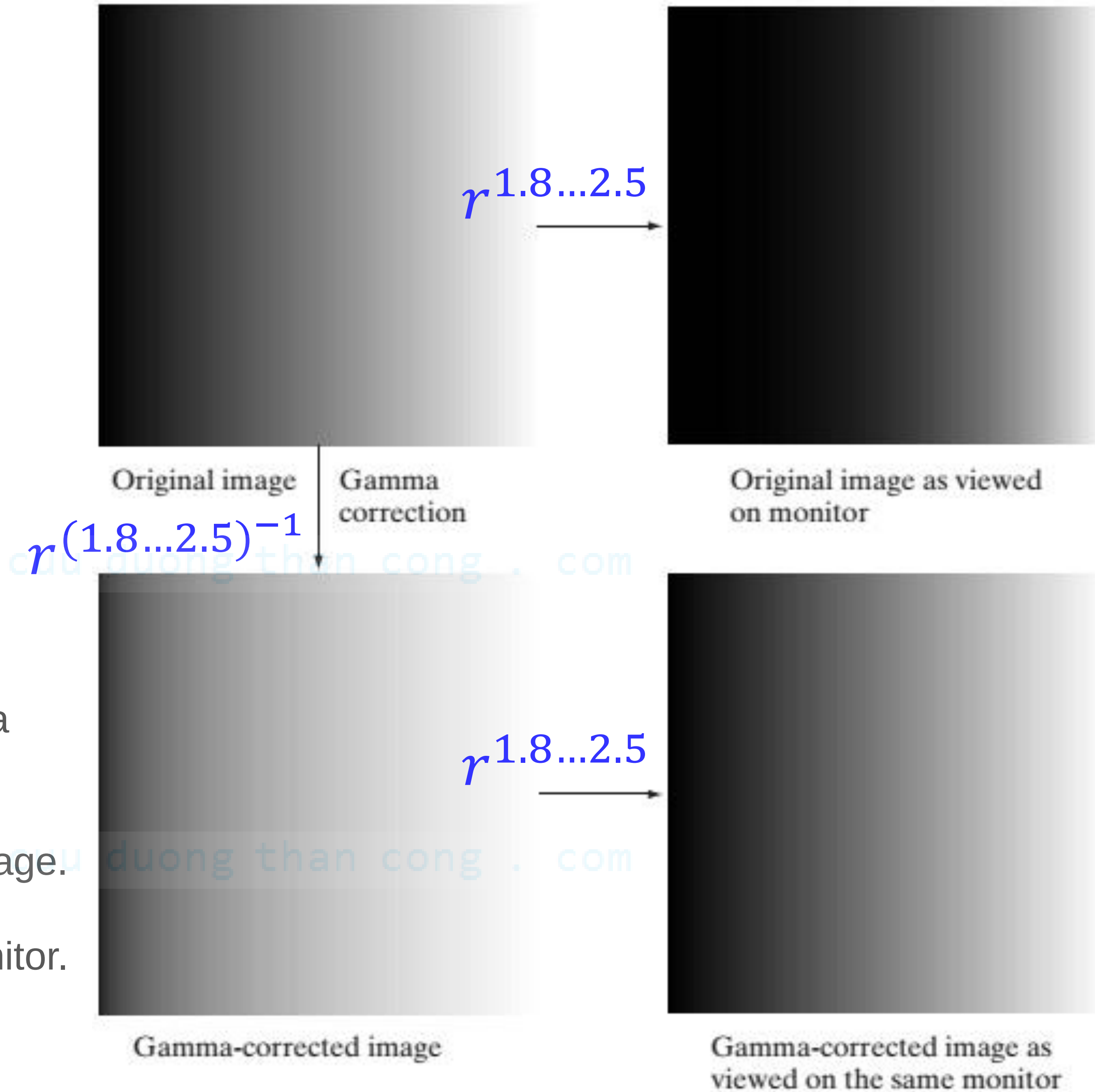
Plots of the equation $s = cr^{\gamma}$ for various values of γ. All curves were scaled to fit in the range shown.

Power-law transformation: Applications

- Gamma correction
  - Correct power-law intensity-to-voltage responses
    - A variety of devices used for image capture, printing, and display respond according to a power law, e.g., cathode ray tube (CRT) devices
  - Important for displaying an image accurately on a computer screen
- General-purpose contrast manipulation
  - Enhance the image contrast for better observation
  - If the image is predominantly dark $\rightarrow$ expand intensity levels. If it has washed-out appearance $\rightarrow$ apply the opposite procedure.

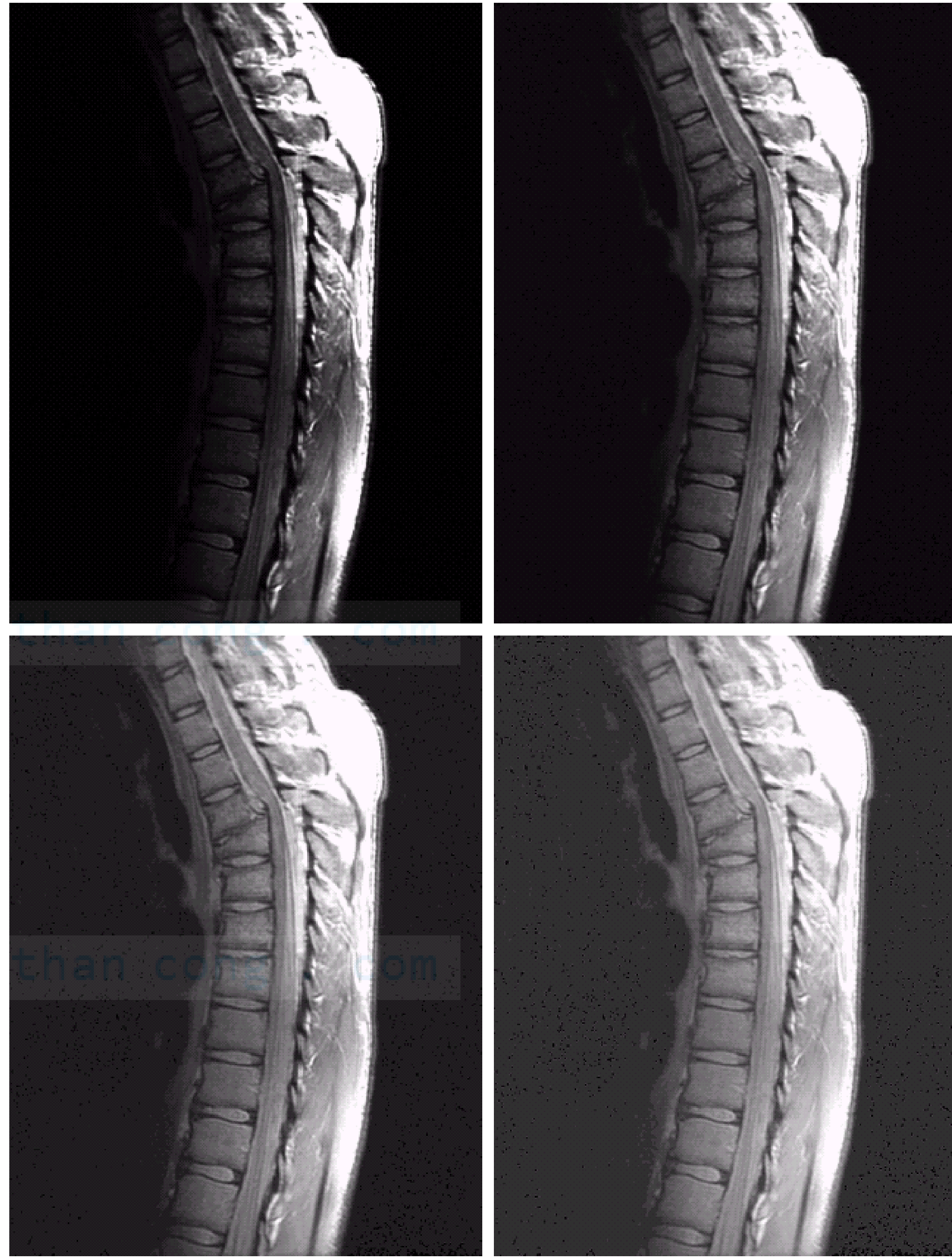
a	b
c	d

(a) Intensity ramp image.
(b) Image as viewed on a simulated monitor with a gamma of 2.5.
(c) Gamma corrected image.
(d) Corrected image as viewed on the same monitor. Compare (d) and (a).



a b
c d

(a) Magnetic resonance image (MRI) of a fractured human spine.
(b)–(d) Results of applying the power-law transformation with c = 1 and γ = 0.6, 0.4, and 0.3, respectively.
(Original image courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)



a	b
c	d

(a) Aerial image. (b)–(d) Results of applying the power-law transformation with $c = 1$ and $\gamma = 3.0$, 4.0, and 5.0, respectively. (Original image for this example courtesy of NASA.)

Piecewise-linear transformation functions

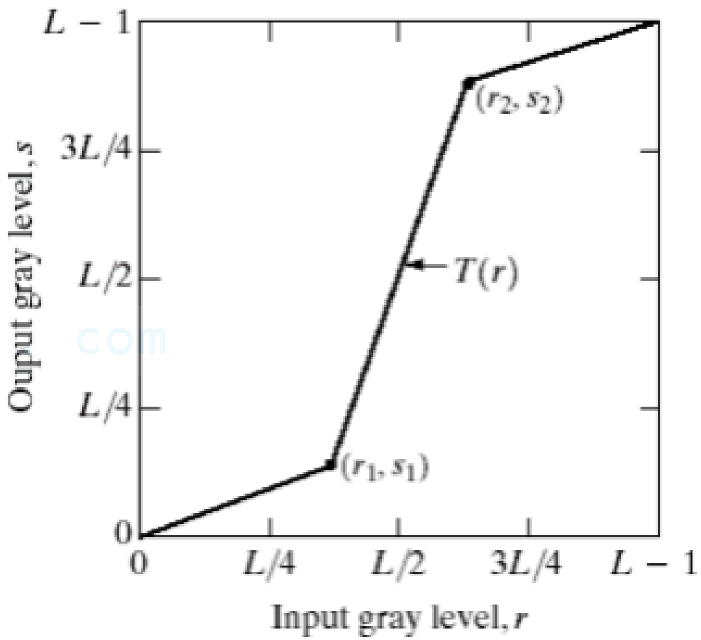
- A complementary approach to the aforementioned methods
- The form of these functions can be arbitrarily complex
- However, their specification requires considerably more user input

- Some common piecewise-linear transformation functions
  - Contrast stretching
  - Intensity-level slicing
  - Bit-plane slicing

Contrast stretching

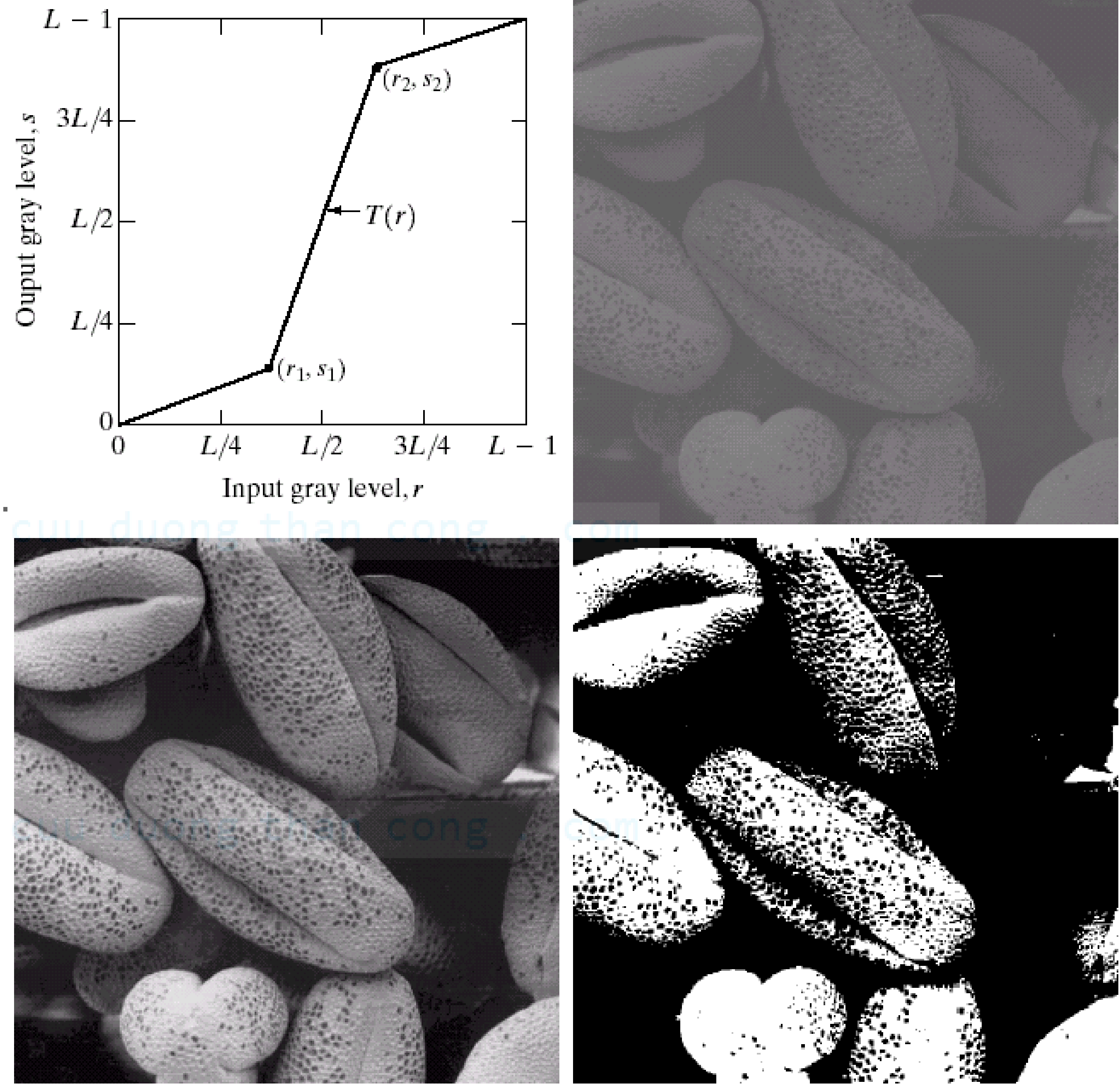
- Expand the range of intensity levels so that it spans the full intensity range of the recording medium or display device.
  - Low-contrast images can result from poor illumination, lack of dynamic range in the imaging sensor, or even the wrong setting of a lens aperture during image acquisition.

- In general, $r_1 \leq r_2$ and $s_1 \leq s_2$ is assumed so that the function is *single-valued* and *monotonically increasing*



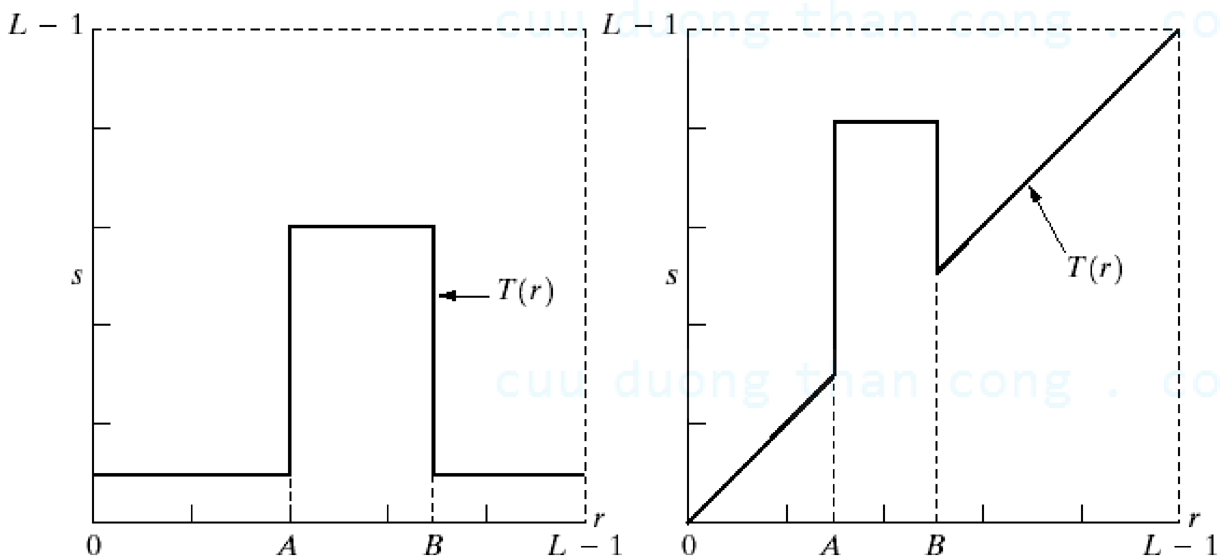
a b
c d

Contrast stretching.
(a) Form of transformation function.
(b) A low-contrast image.
(c) Result of contrast stretching.
(d) Result of thresholding.
(Original image courtesy of Dr. Roger Heady, Research School of Biological Sciences, Australian National University, Canberra, Australia.)



Intensity slicing

- Highlight a specific range of intensities in an image
  - E.g., enhancing features such as masses of water in satellite imagery or flaws in X-ray images
- Two basic themes of implement intensity slicing are

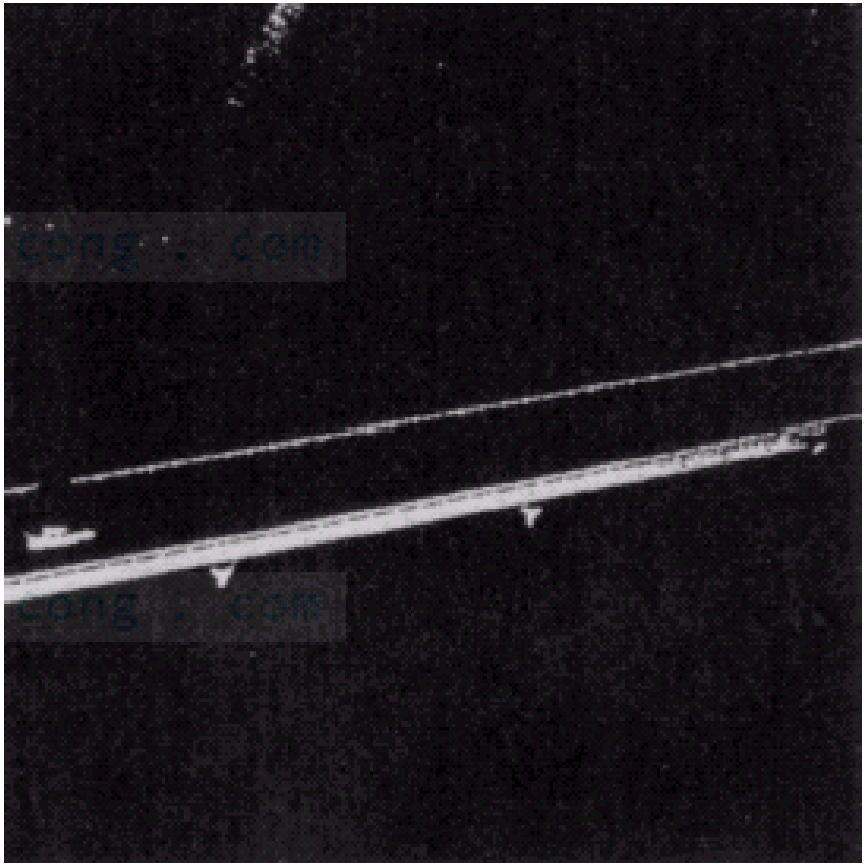


a | b

(a) Highlighting intensity range [*A*, *B*] and reducing all other intensities to a lower level.
(b) Highlighting intensity range [*A*, *B*] and preserving all other intensity levels.

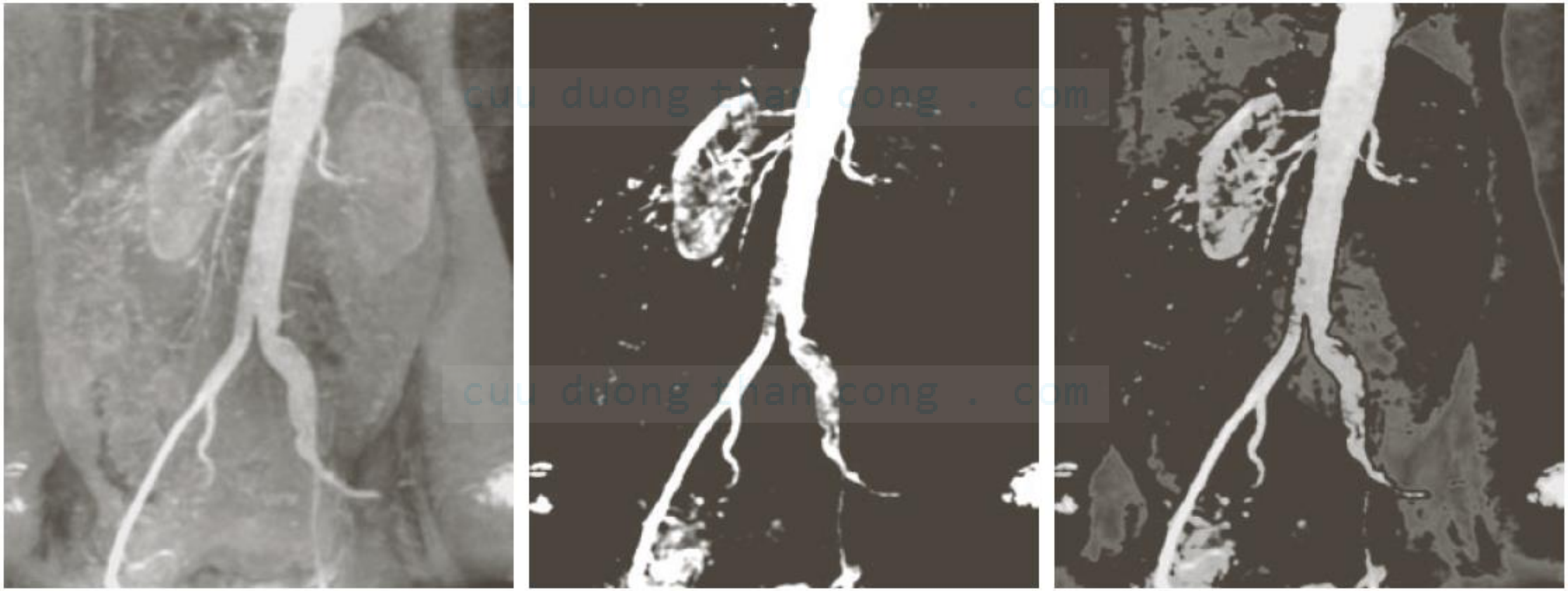
a b

(a) Original image. (b) Result of using the first intensity slicing theme with the range of intensities of interest selected in the upper end of the gray scale.



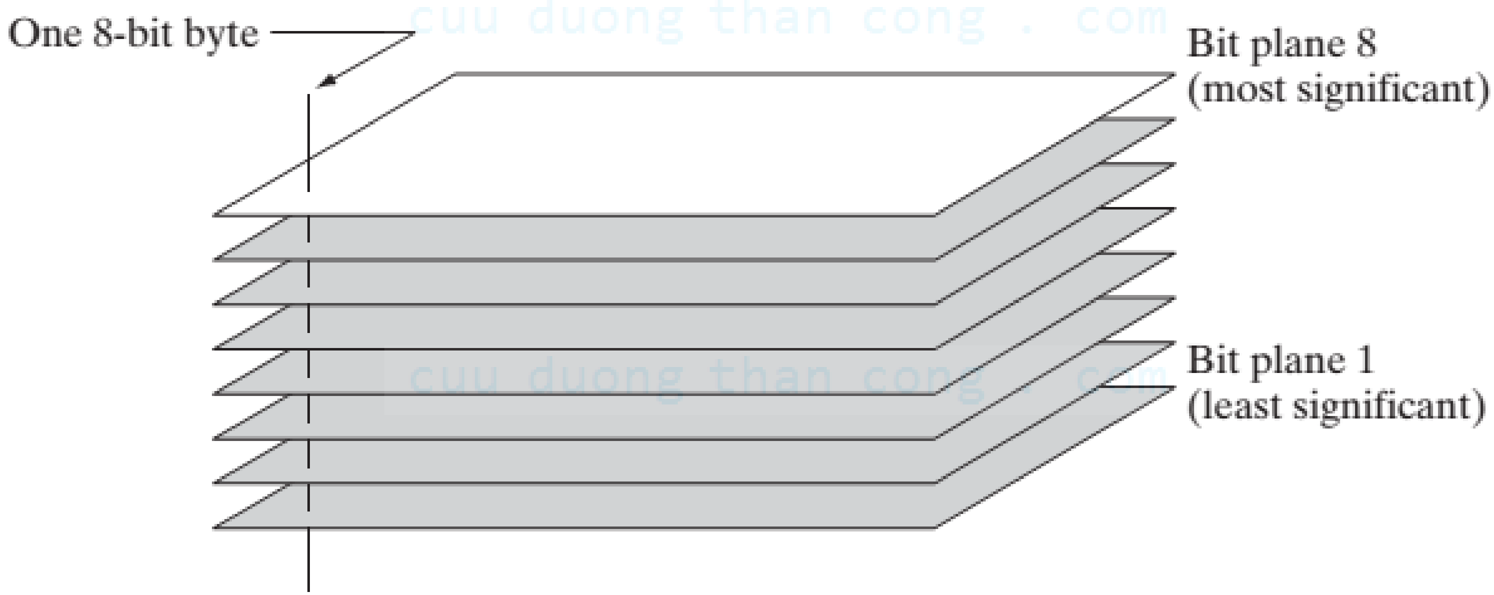
a b c

(a) Aortic angiogram. (b) Result of using the first intensity slicing theme with the range of intensities of interest selected in the upper end of the gray scale. (c) Result of using the second theme, with the selected area set to black, so that grays in the area of the blood vessels and kidneys were preserved. (Original image courtesy of Dr. Thomas R. Gest, University of Michigan Medical School.)



Bit-plane slicing

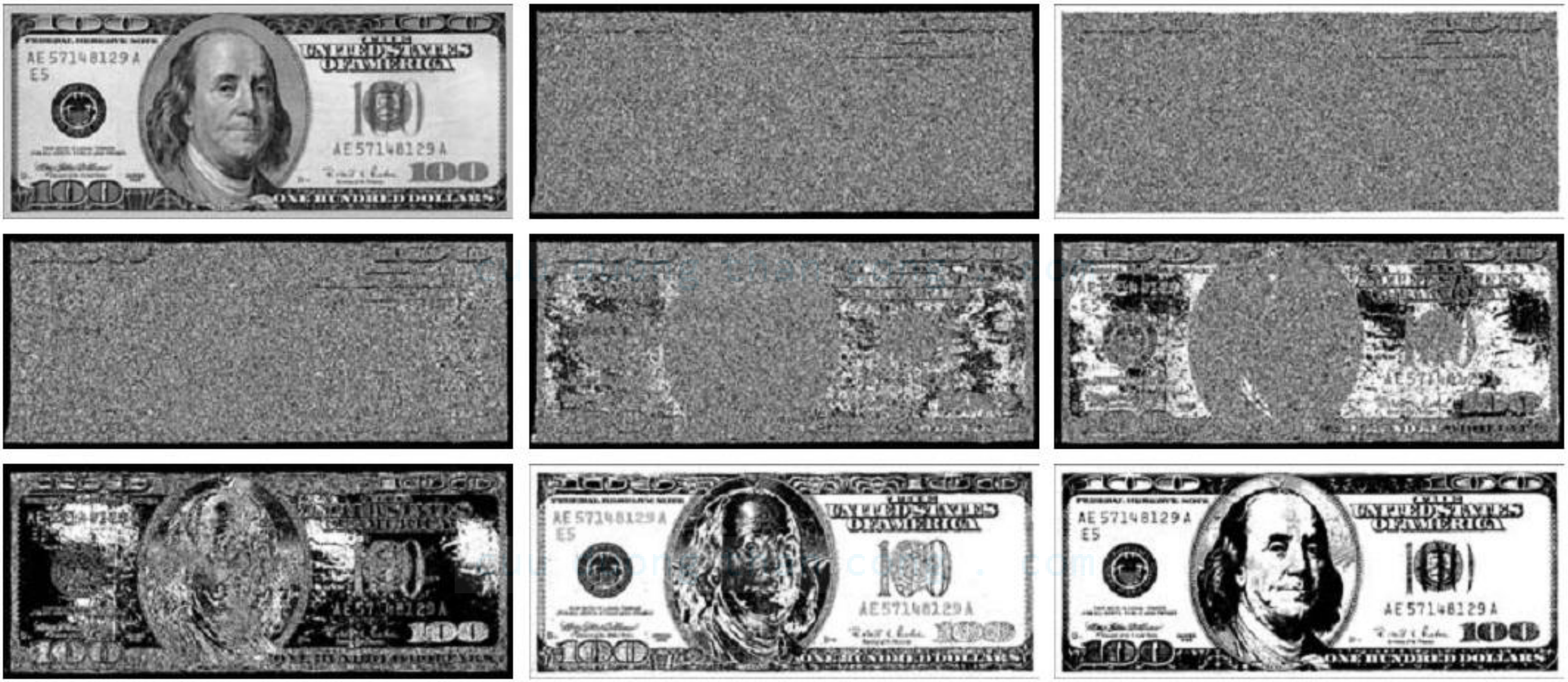
- Highlight the contribution made to total image appearance by specific bits
  - The four higher-order bit planes, especially the last two, contain a significant amount of the **visually significant data**, while the lower-order planes contribute to more **subtle intensity details**



Bit-plane representation of an 8-bit image.

a	b	c
d	e	f
g	h	i

(a) An 8-bit gray-scale image of size pixels. (b) through (i) Bit planes 1 through 8, with bit plane 1 corresponding to the least significant bit. Each bit plane is a binary image.



Bit-plane slicing

- **Image compression:** Fewer than all planes are used to reconstruct the image
  - The four highest-order bit planes reconstruct the original image in acceptable detail.
  - Storing these four planes requires 50% less storage (ignoring memory architecture issues)

a b c

Images reconstructed using (a) bit planes 8 and 7; (b) bit planes 8, 7, and 6; and (c) bit planes 8, 7, 6, and 5. Compare (c) with Fig. 3.14(a).

Section 3.3

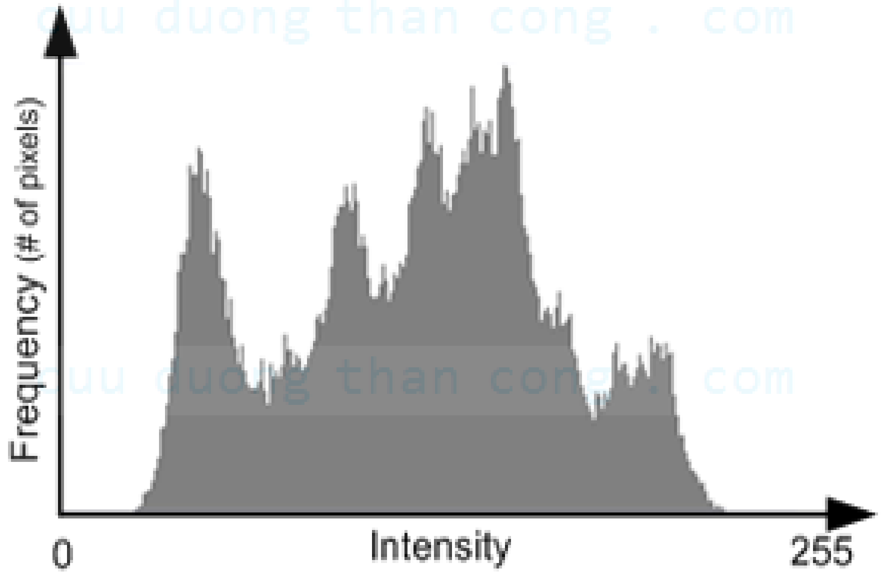
HISTOGRAM PROCESSING

Histogram $h(r_k)$

- The **histogram** of a digital image with intensity levels in the range $[0, L-1]$ is a discrete function of the form

$$h(r_k) = n_k$$

  - where r_k is the kth intensity value and n_k is the number of pixels in the image with intensity r_k



An example of histogram of a 8-bit image

Normalized histogram $p(r_k)$

- It is common practice to normalize a histogram using

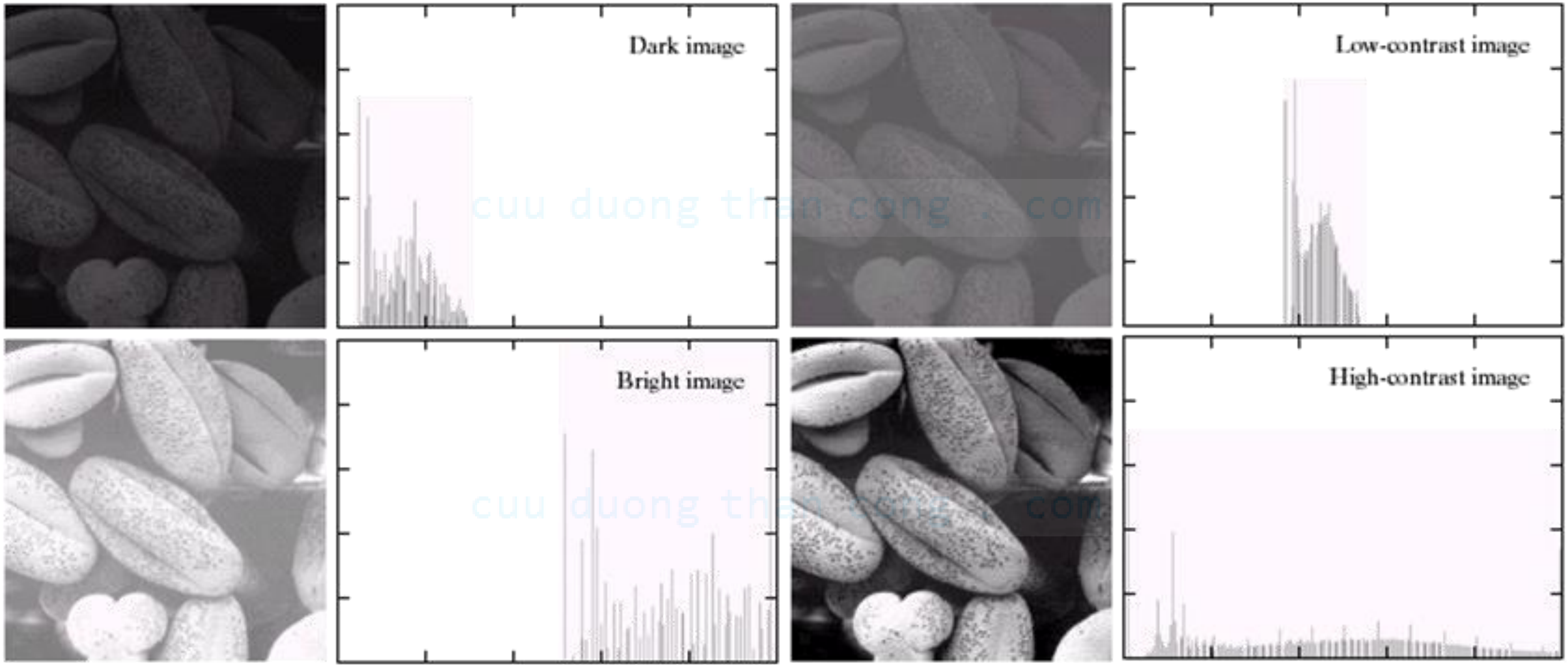
$$p(r_k) = \frac{n_k}{MN}$$

  - where M and N are the row and column dimensions of the image, and $k = 0, 1, 2, \ldots, L-1$

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

a b
c d

Four image types: dark, light, low contrast, high contrast, and their corresponding histograms



Histogram equalization

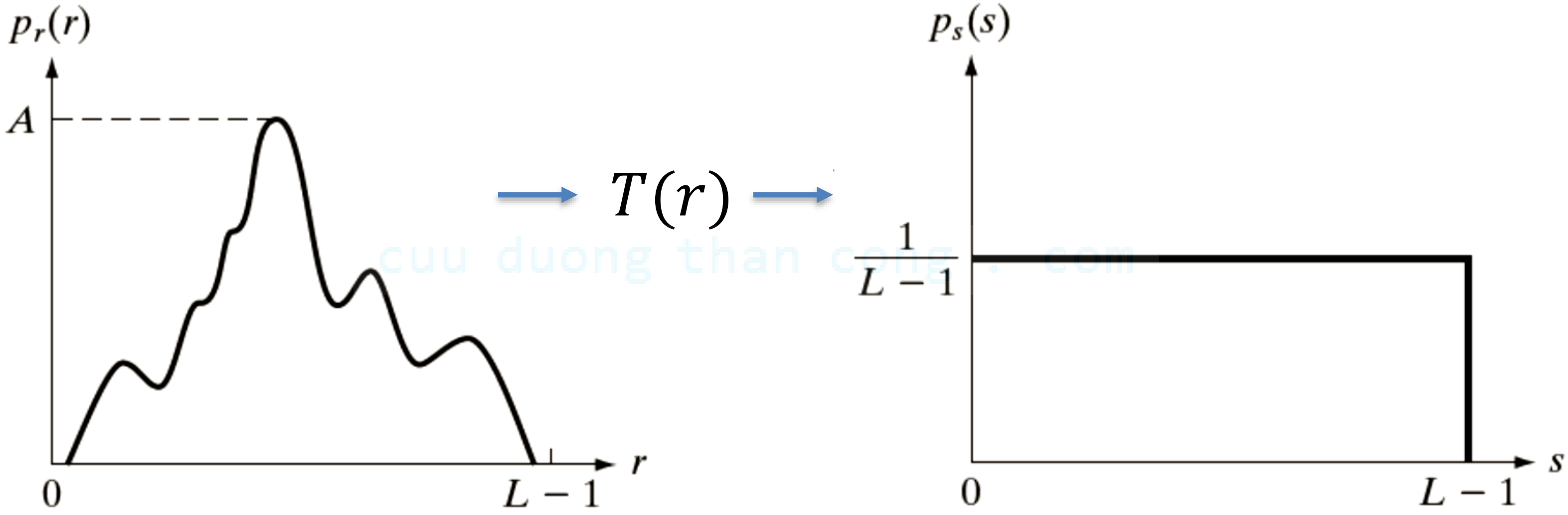
- Each pixel in the input image with intensity r_k is mapped into a corresponding pixel of level s_k in the output image

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

  - where $k = 0, 1, 2, \dots, L-1$
- Also called histogram linearization

Histogram equalization

- The transformation function T is determined **automatically** to produce an output image that has a uniform histogram.



a b

(a) An arbitrary PDF. (b) Result of applying histogram equalization to all intensity levels, r. The resulting intensities, s, have a uniform PDF, independently of the form of the PDF of the r's.

3-bit ($L = 8$) image of size 64×64 ($MN = 4096$). Intensities are in the range $[0, L-1] = [0, 7]$

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

Values of the histogram equalization transformation function s_k are computed

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

$$s_1 = T(r_1) = 7\sum_{j=0}^{1} p_r(r_j) = 7p_r(r_0) + 7p_r(r_1) = 3.08$$

Similarly, $s_2 = 4.55$, $s_3 = 5.67$, $s_4 = 6.23$, $s_5 = 6.65$, $s_6 = 6.86$, $s_7 = 7.00$

Round the s values to the nearest integers

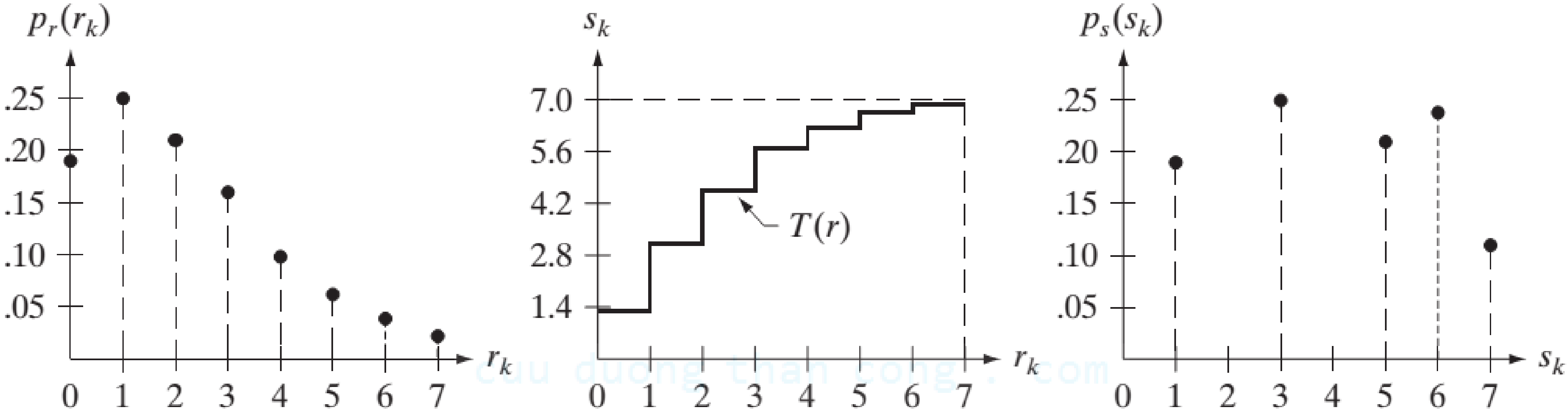
$s_0 = 1.33 \rightarrow 1$ $\quad s_4 = 6.23 \rightarrow 6$

$s_1 = 3.08 \rightarrow 3$ $\quad s_5 = 6.65 \rightarrow 7$

$s_2 = 4.55 \rightarrow 5$ $\quad s_6 = 6.86 \rightarrow 7$

$s_3 = 5.67 \rightarrow 6$ $\quad s_7 = 7.00 \rightarrow 7$

s_k	m_k	$p_s(s_k) = m_k/MN$
$s_0 = 1$	790	0.19
$s_1 = 3$	1023	0.25
$s_2 = 5$	850	0.21
$s_{3,4} = 6$	985	0.24
$s_{5,6,7} = 7$	448	0.11

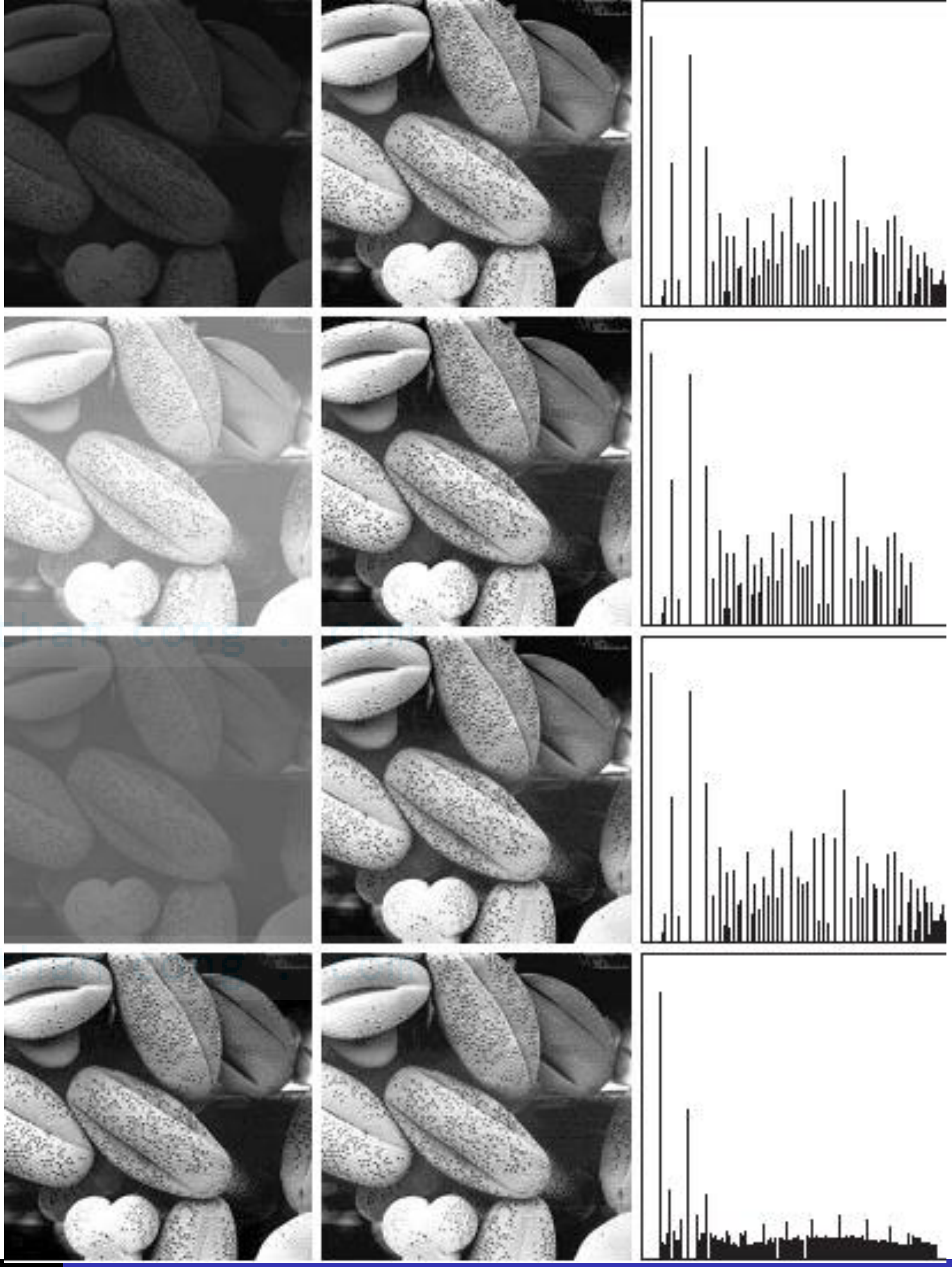


a b c

Illustration of histogram equalization of a 3-bit (8 intensity levels) image. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.

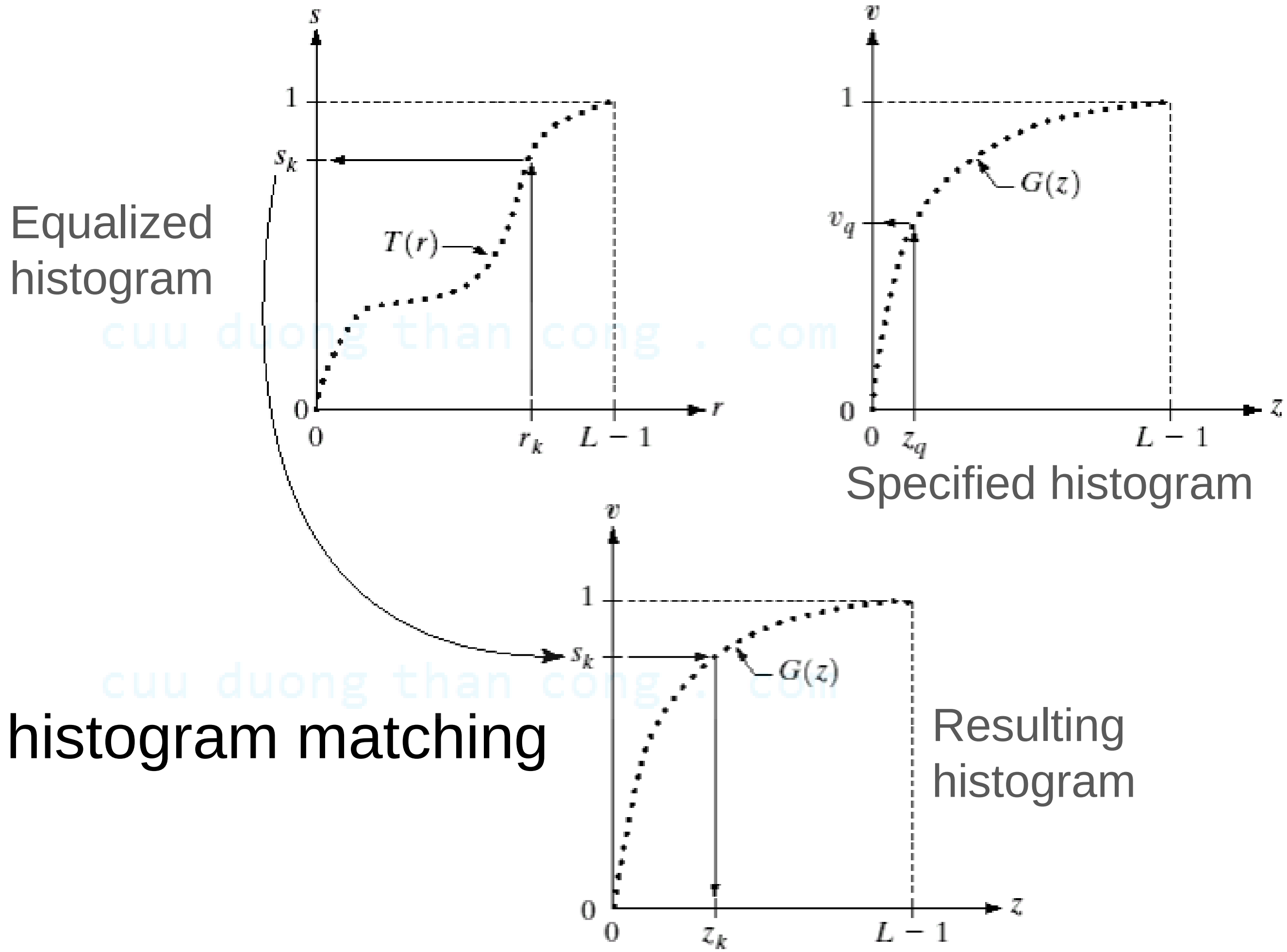
Note: *It cannot be proved (in general) that discrete histogram equalization results in a uniform histogram, yet it has a general tendency to spread the histogram of the input image* a b c

Left column: original images.
Center column: corresponding histogram-equalized images.
Right column: histograms of the images in the center column.



Histogram specification

- Create a processed image whose histogram shape is specified.



- Also called histogram matching

Histogram specification procedure

1. Compute the histogram of the given image, and then the histogram equalization transformation. Round the resulting values, s_k, to the integer range $[0,\ L-1]$
2. Compute all values of the transformation function G using

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

- where $q = 0, 1, 2, \ldots, L-1$, $p_z(z_i)$ are the values of the specified histogram

Round the values of G to integers in the range $[0,\ L-1]$ then store them in a table

z_q	Specified $p_z(z_q)$
$z_0 = 0$	0.00
$z_1 = 1$	0.00
$z_2 = 2$	0.00
$z_3 = 3$	0.15
$z_4 = 4$	0.20
$z_5 = 5$	0.30
$z_6 = 6$	0.20
$z_7 = 7$	0.15

Compute all the values of the transformation function, G

$$G(z_0) = 7\sum_{i=0}^{0} p_z(z_i) = 0.00$$

$$G(z_1) = 7\sum_{j=0}^{1} p_z(z_i) = 7p_z(z_0) + 7p_z(z_1) = 0.00$$

Similarly,

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

$$G(z_5) = 4.55 \quad G(z_6) = 5.95 \quad G(z_7) = 7.00$$

z_q	$G(z_q)$
$z_0 = 0$	0
$z_1 = 1$	0
$z_2 = 2$	0
$z_3 = 3$	1
$z_4 = 4$	2
$z_5 = 5$	5
$z_6 = 6$	6
$z_7 = 7$	7

Round the $G(z)$ values to the nearest integers

$$G(z_0) = 0.00 \rightarrow 0 \quad G(z_1) = 0.00 \rightarrow 0$$

$$G(z_2) = 0.00 \rightarrow 0 \quad G(z_3) = 1.05 \rightarrow 1$$

$$G(z_4) = 2.45 \rightarrow 2 \quad G(z_5) = 4.55 \rightarrow 5$$

$$G(z_6) = 5.95 \rightarrow 6 \quad G(z_7) = 7.00 \rightarrow 7$$

Histogram specification procedure

3. For every value of s_k, use the stored values of G from step 2 to find the corresponding value of z_q so that $G(z_q)$ is closest to s_k and store these mappings from s to z.
   - When more than one value of z_q satisfies the given s_k (i.e., the mapping is not unique), choose the smallest value by convention

4. Map every equalized pixel value, s_k, of the input image to the corresponding value z_q in the histogram-specified image using the mappings found in step 3.
   - The intermediate step of equalizing the input image can be skipped by combining the two transformation functions, T and G^{-1}

From step 1

$s_0 = 1 \quad s_2 = 5 \quad s_4 = 6 \quad s_6 = 7$

$s_1 = 3 \quad s_3 = 6 \quad s_5 = 7 \quad s_7 = 7$

From step 2

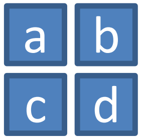
z_q	$G(z_q)$
$z_0 = 0$	0
$z_1 = 1$	0
$z_2 = 2$	0
$z_3 = 3$	1
$z_4 = 4$	2
$z_5 = 5$	5
$z_6 = 6$	6
$z_7 = 7$	7

From step 3

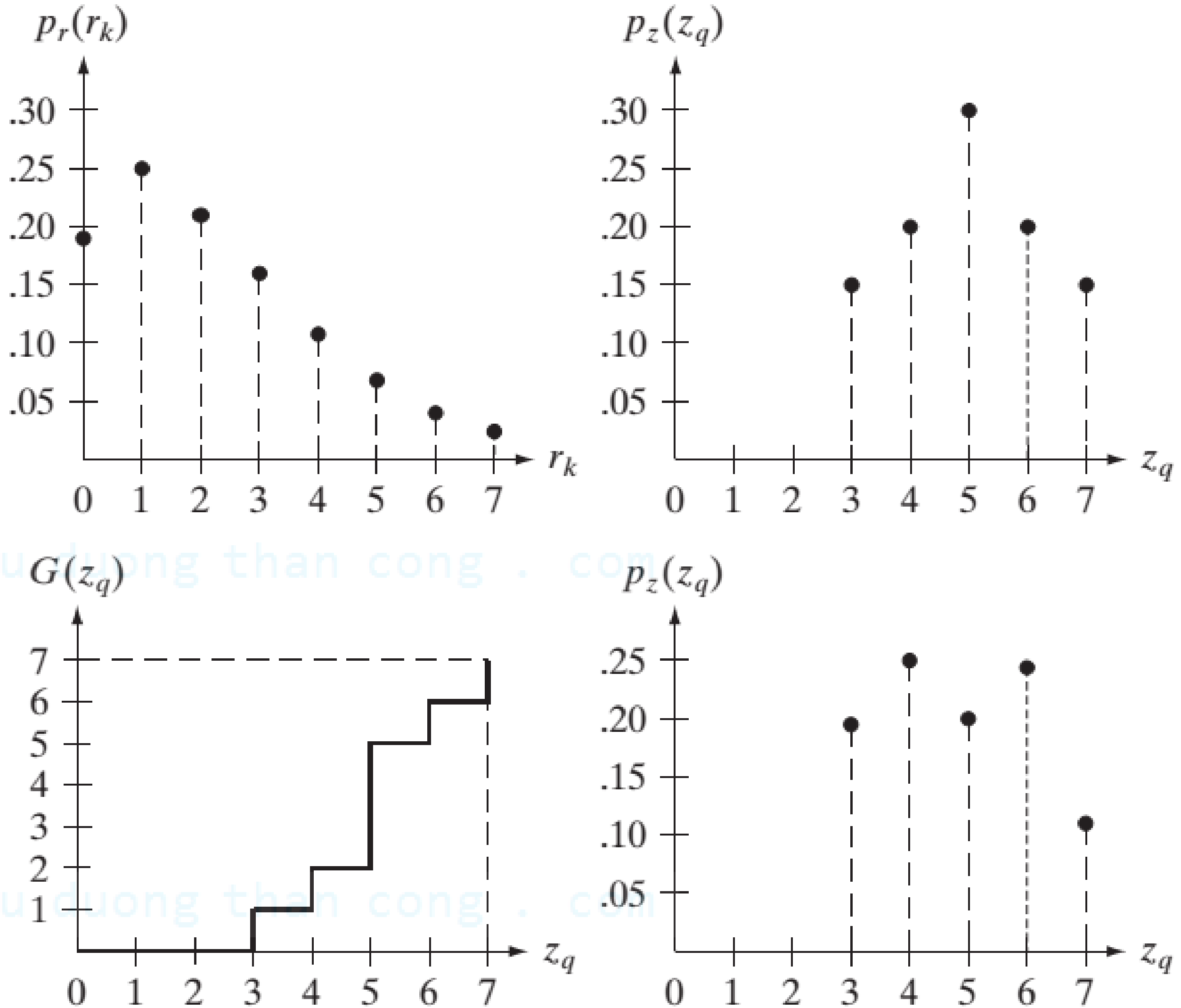
s_k	$\rightarrow$	z_q
1	$\rightarrow$	3
3	$\rightarrow$	4
5	$\rightarrow$	5
6	$\rightarrow$	6
7	$\rightarrow$	7

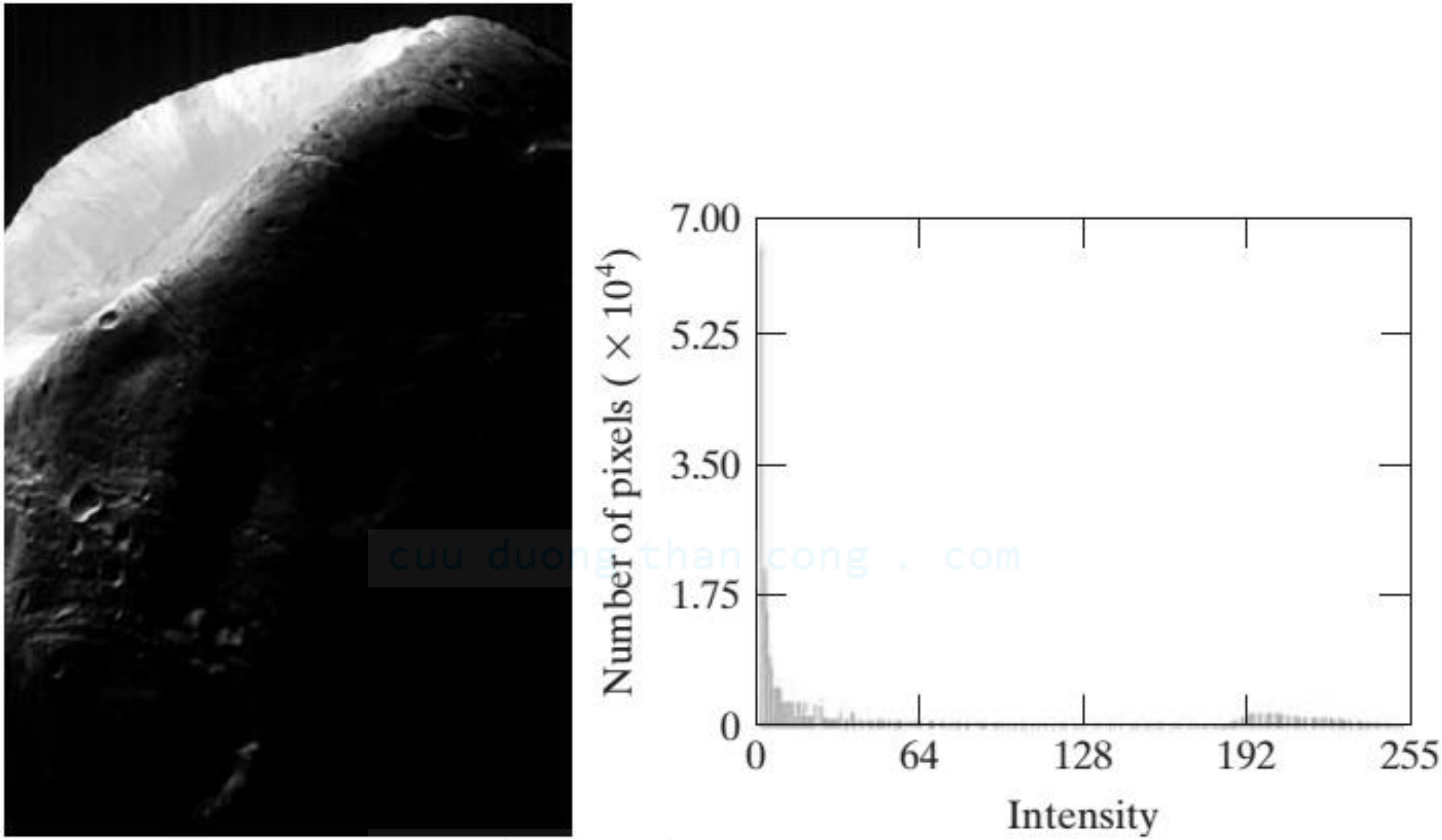
From step 4

z_q	Specified $p_z(z_q)$	Actual $p_z(z_k)$
$z_0 = 0$	0.00	0.00
$z_1 = 1$	0.00	0.00
$z_2 = 2$	0.00	0.00
$z_3 = 3$	0.15	0.19
$z_4 = 4$	0.20	0.25
$z_5 = 5$	0.30	0.21
$z_6 = 6$	0.20	0.24
$z_7 = 7$	0.15	0.11



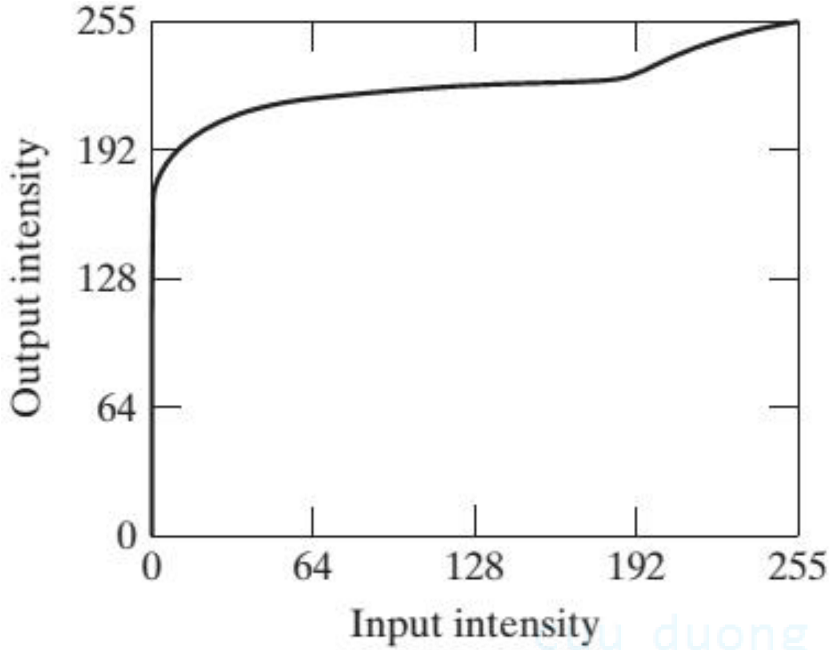
(a) Histogram of a 3-bit image.
(b) Specified histogram.
(c) Transformation function obtained from the specified histogram.
(d) Result of performing histogram specification. Compare (b) and (d).

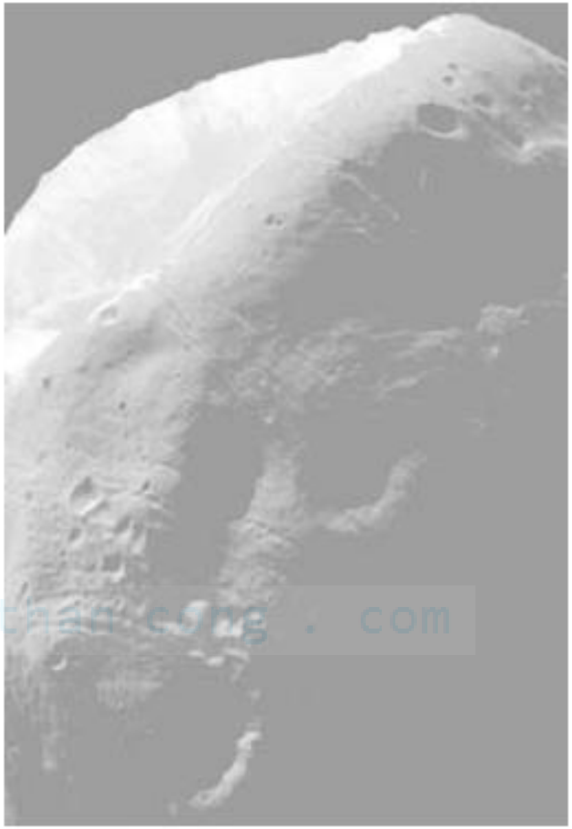


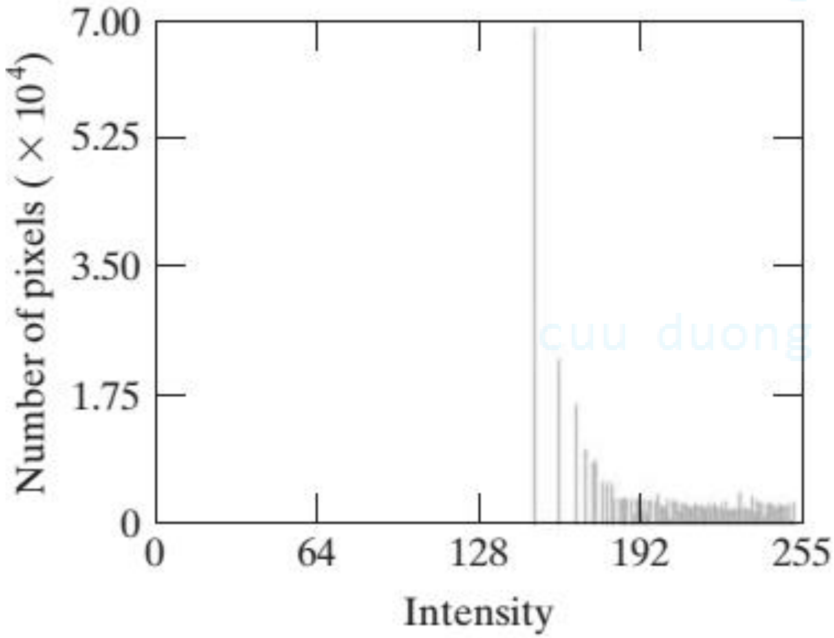


a b

(a) Image of the Mars moon Phobos taken by NASA's Mars Global Surveyor. (b) Histogram. (Original image courtesy of NASA.)

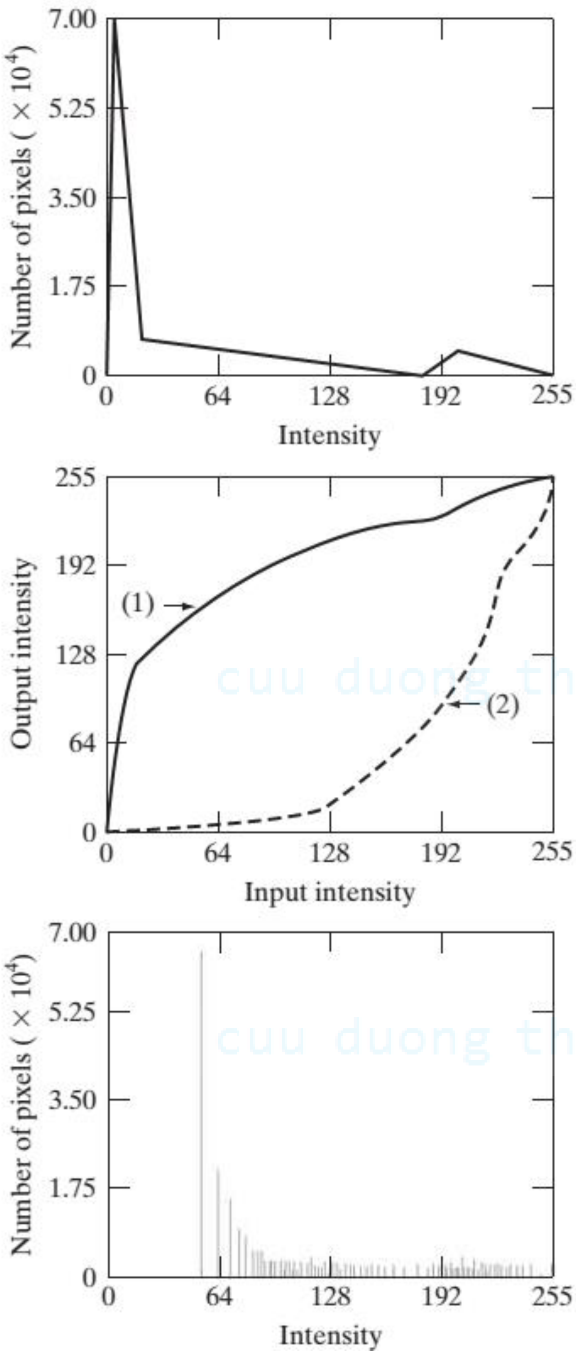






a b
c

(a) Transformation function for histogram equalization.
(b) Histogram-equalized image (note the washed-out appearance).
(c) Histogram of (b).

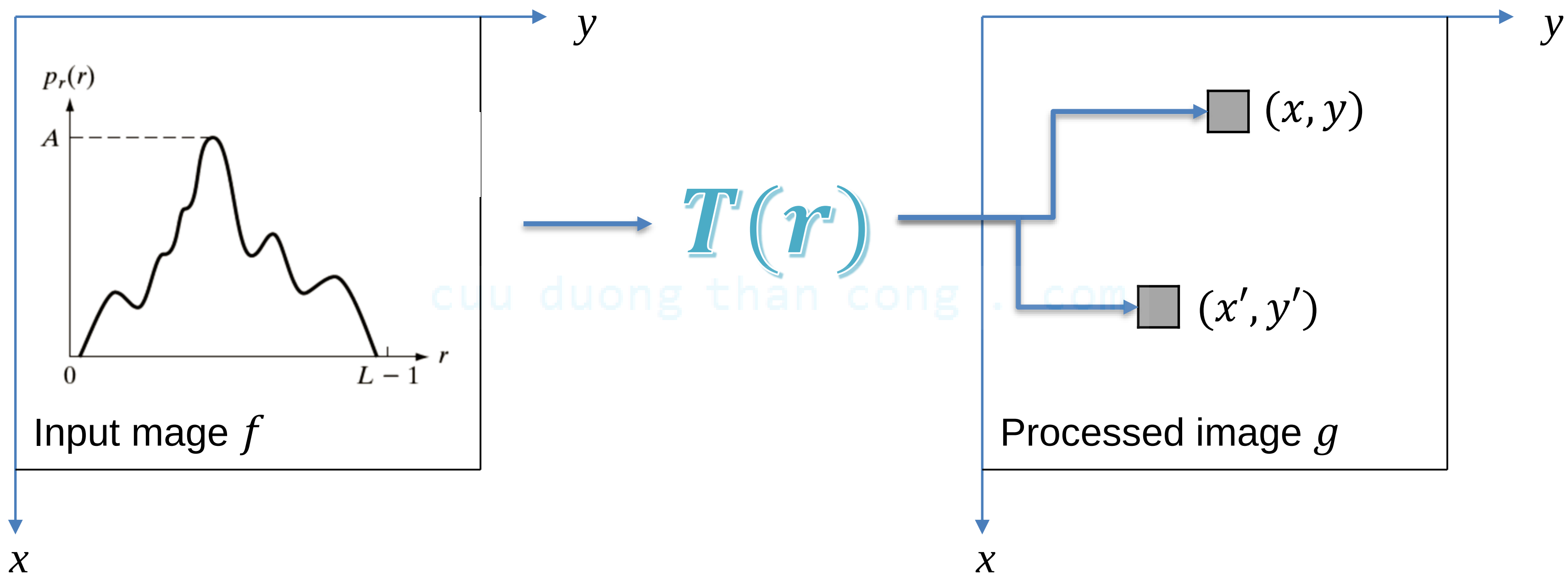


a c
b
d

(a) Specified histogram.
(b) Transformations.
(c) Enhanced image using mappings from curve (2).
(d) Histogram of (c).

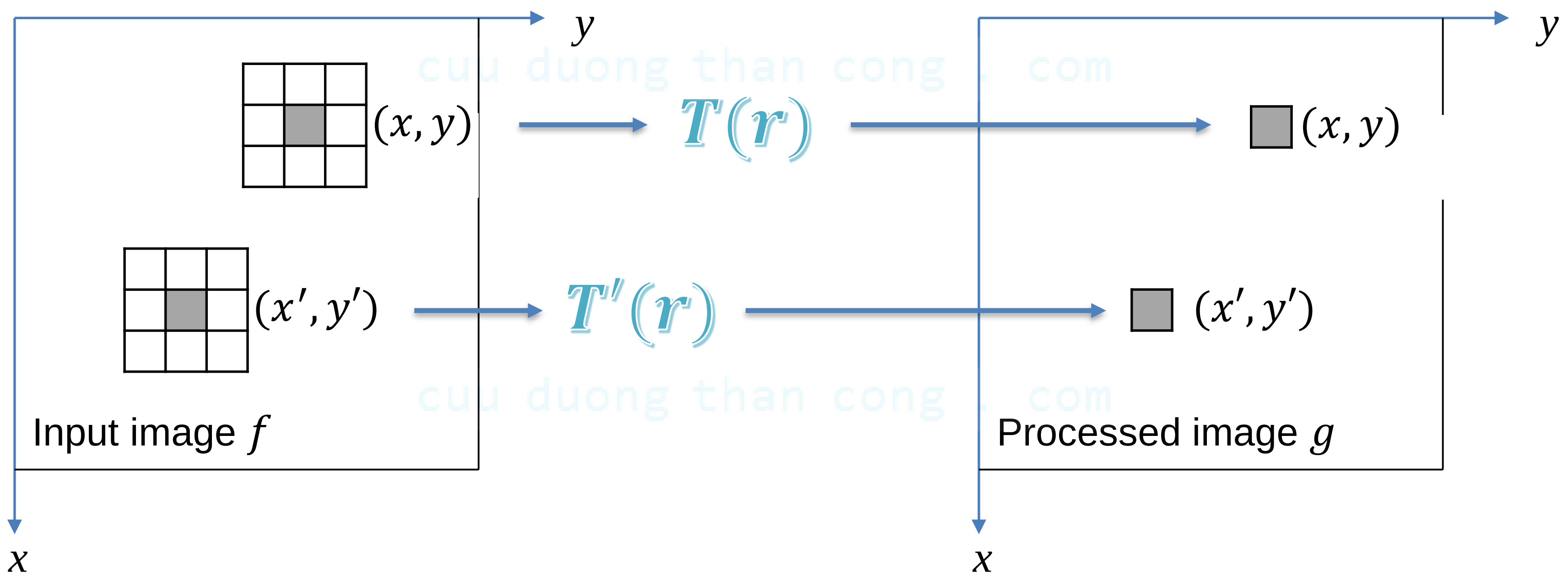
Histogram processing: global vs. local

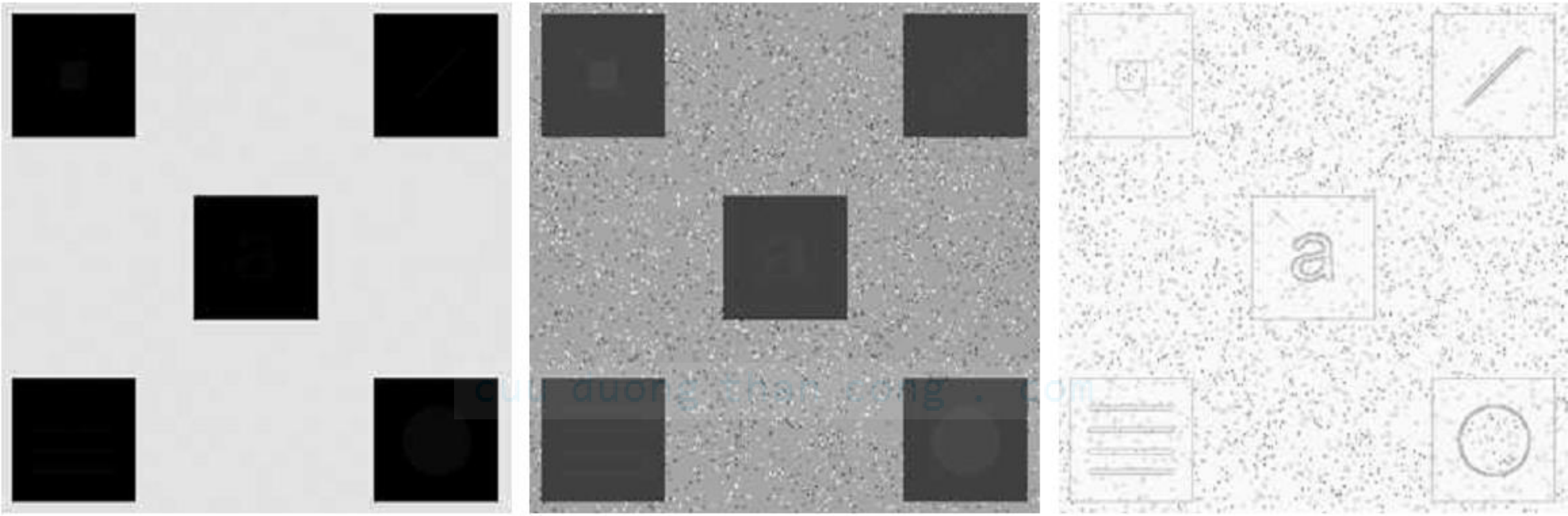
- **Global histogram processing:** Pixels are modified by a transformation function based on the intensity distribution of an **entire image**.
- Suitable for overall enhancement while less effective to details over small areas



Histogram processing: global vs. local

- **Local histogram processing:** A pixel at location (x, y) is modified by a transformation function based on the intensity distribution **in the neighborhood** centered on (x, y)





a b c

(a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization applied to (a), using a neighborhood of size 3×3.

Global histogram statistics

- Let r denote a discrete random variable representing intensity values in the range $[0, L-1]$ and $p(r_i)$ denote the normalized histogram component corresponding to r_i
- The **mean** (a measure of average intensity) is given by

$$\boldsymbol{m = \sum_{i=0}^{L-1} r_i \, p(r_i)}$$

- The **variance** (a measure of contrast) is given by

$$\boldsymbol{\sigma^2 = \sum_{i=0}^{L-1} (r_i - m)^2 \, p(r_i)}$$

Global histogram statistics

- The mean and variance can be estimated directly from the sample values, without computing the histogram
- Let M and N be row and column dimensions of the image, and $f(x, y)$ is the pixel intensity at (x, y)
  - where $x = 0, 1, 2, \dots, L-1, y = 0, 1, 2, \dots, N-1$

- The **mean**

$$\boldsymbol{m = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x, y)}$$

- The **variance**

$$\boldsymbol{\sigma^2 = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} [f(x, y) - m]^2}$$

2-bit image ($L = 4$). Intensities are in the range $[0, L-1]$

0	0	1	1	2
1	2	3	0	1
3	3	2	2	0
2	3	1	0	0
1	1	3	2	2

The values of normalized histogram component $p(r_k)$

$$p(r_0) = \frac{6}{25} = 0.24 \qquad p(r_2) = \frac{7}{25} = 0.28$$

$$p(r_1) = \frac{7}{25} = 0.28 \qquad p(r_3) = \frac{5}{25} = 0.20$$

The (global) mean and variance

$$m = \sum_{i=0}^{3} r_i\, p(r_i) = (0)(0.24) + (1)(0.28) + (2)(0.28) + (3)(0.20) = 1.44$$

$$\sigma^2 = \sum_{i=0}^{L-1} (r_i - m)^2\, p(r_i)$$

$$= (0-1.44)^2(0.24) + (1-1.44)^2(0.28) + (2-1.44)^2(0.28) + (3-1.44)^2(0.20)$$

$$= 1.1264$$

Estimate them directly from the sample values $m = \frac{1}{25}\sum_{x=0}^{4}\sum_{y=0}^{4} f(x,y) = 1.44$

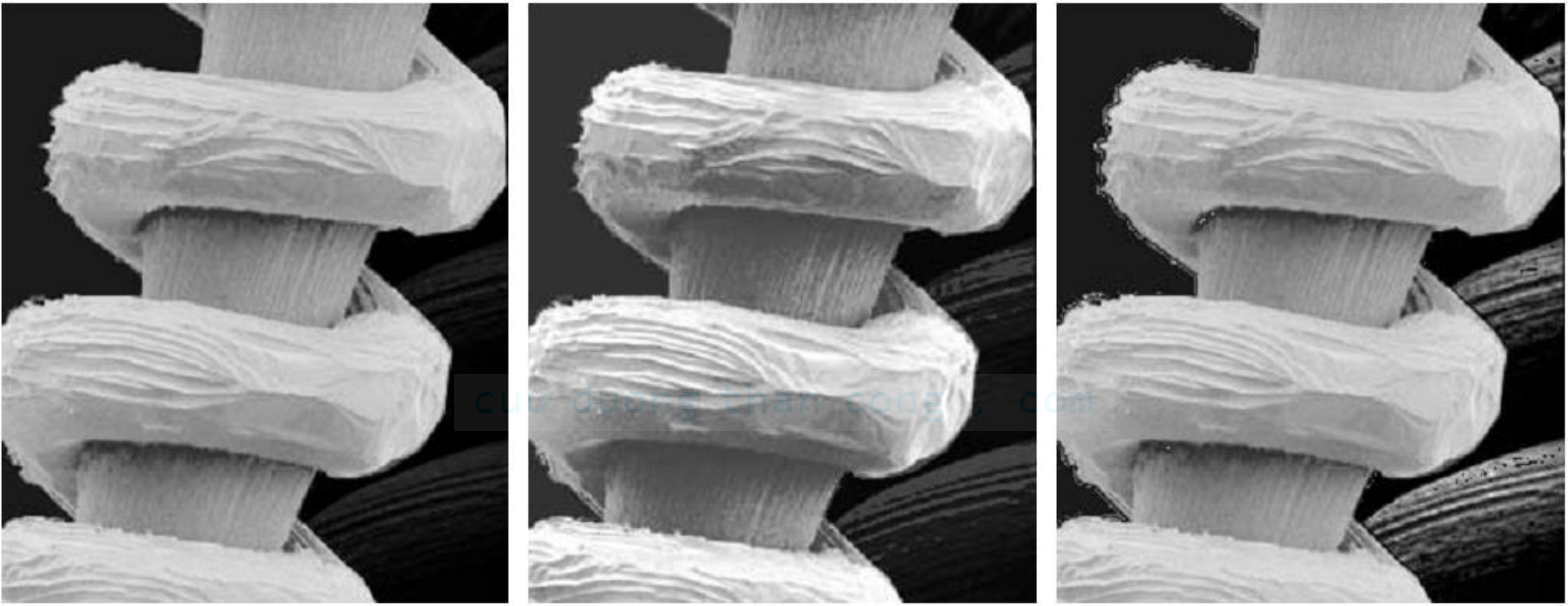
Local histogram statistics

- Let (x, y) denote the coordinates of any pixel in a given image and S_{xy} denote a neighborhood of specified size, centered on (x, y)
- Let p_{xy} the histogram of the pixels in region S_{xy}
- The **mean** value of the pixels in this neighborhood is

$$\boldsymbol{m_{S_{xy}} = \sum_{i=0}^{L-1} r_i \, p_{S_{xy}}(r_i)}$$

- The variance of the pixels in the neighborhood similarly is

$$\boldsymbol{\sigma^2_{S_{xy}} = \sum_{i=0}^{L-1} \left(r_i - m_{S_{xy}}\right)^2 p_{S_{xy}}(r_i)}$$



a b c

(a) SEM image of a tungsten filament magnified approximately. (b) Result of global histogram equalization. (c) Image enhanced using local histogram statistics. (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)

Section 3.4

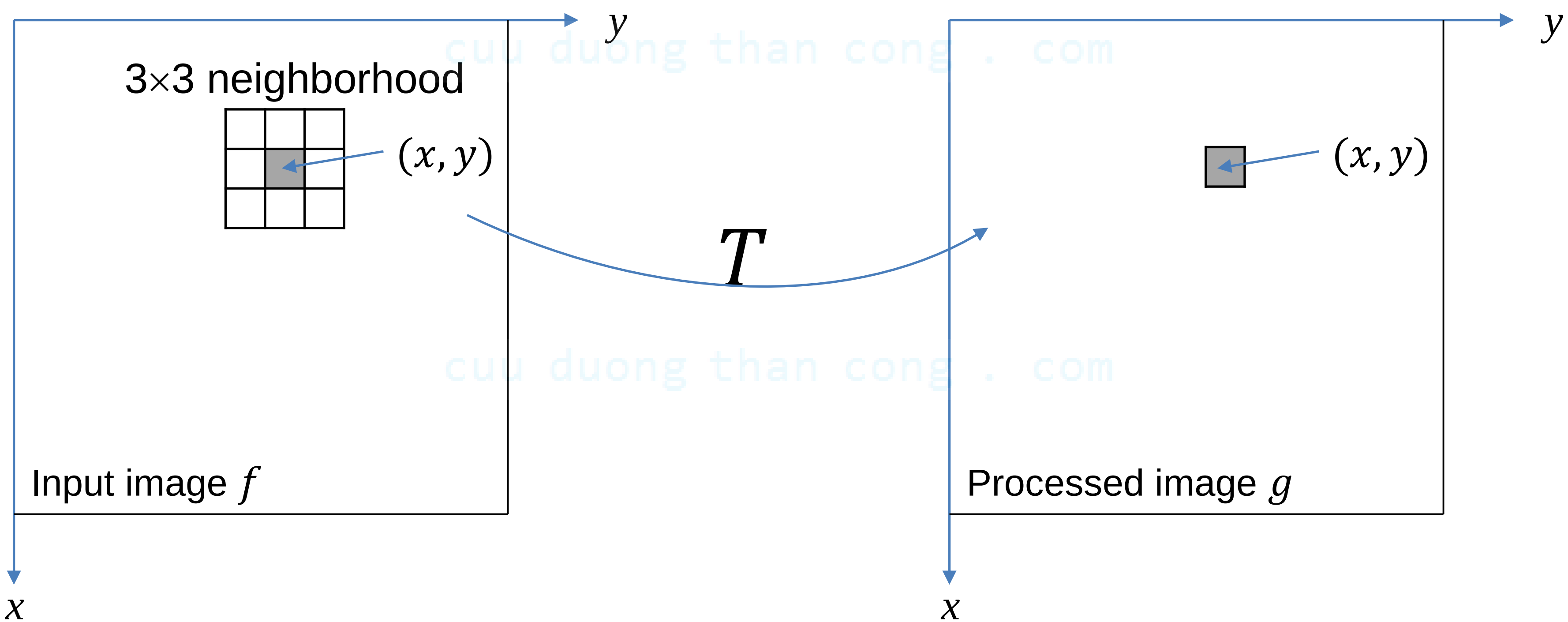
FUNDAMENTALS OF SPATIAL FILTERING

The mechanics of spatial filtering

- A spatial filter consists of

  (1) a neighborhood, (typically a small rectangle), and

  (2) a predefined operation that is performed on the image pixels encompassed by the neighborhood.



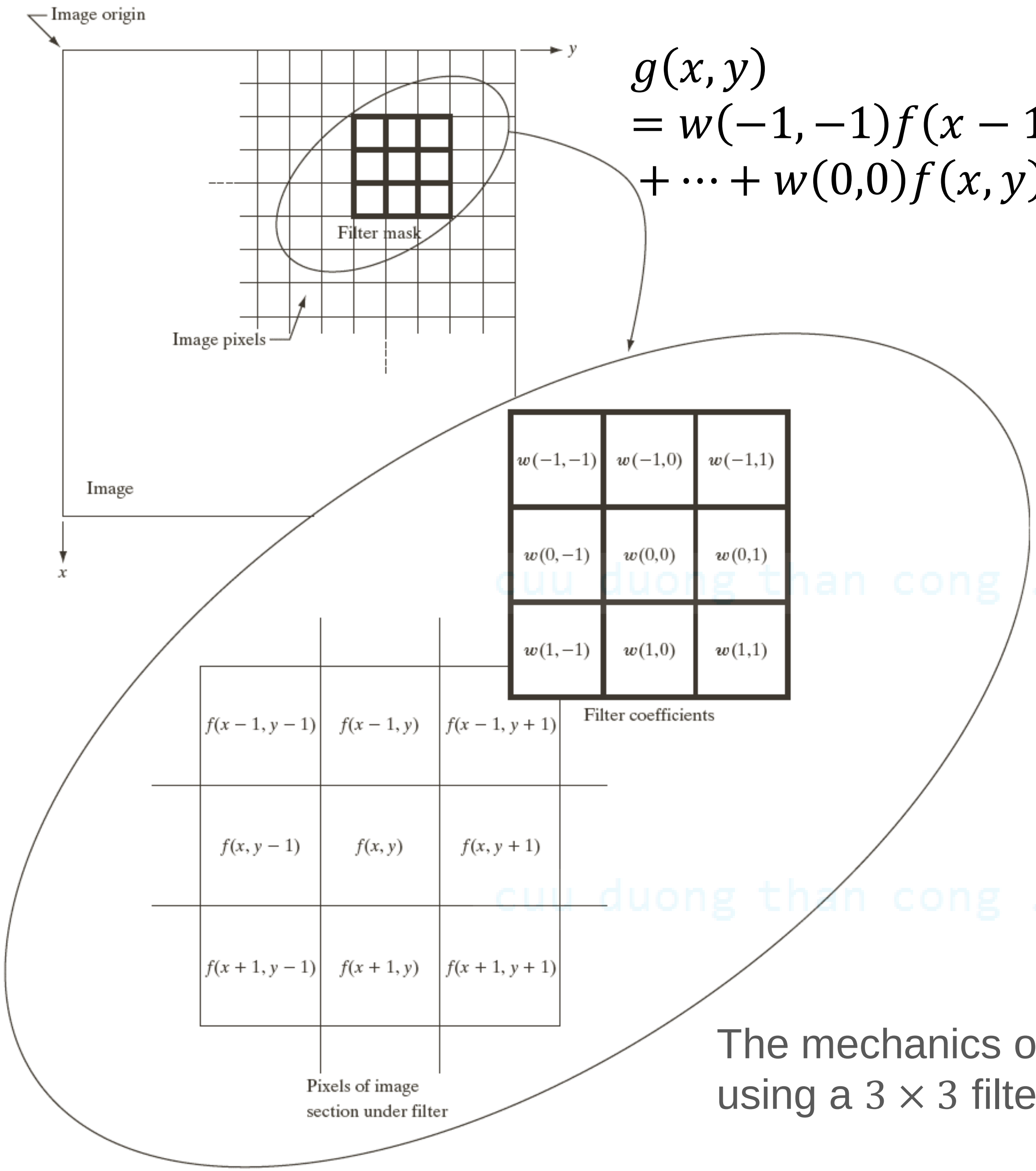
Linear spatial filtering

- **Linear spatial filtering** of an image of size $M \times N$ with a filter of size $m \times n$ is given by the expression

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

  - where $m = 2a + 1$ and $n = 2b + 1$, where a and b are positive integers, and x and y are varied so that each pixel in w visits every pixel in f

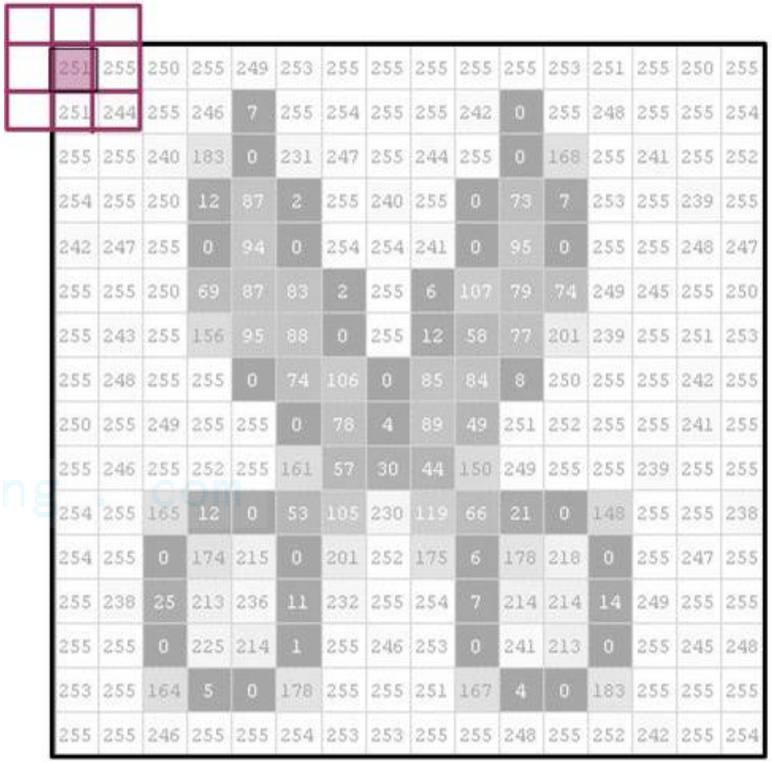
$$g(x,y) = w(-1,-1)f(x-1,y-1) + w(-1,0)f(x-1,y) + \cdots + w(0,0)f(x,y) + \cdots + w(1,1)f(x+1,y+1)$$



The mechanics of linear spatial filtering using a 3×3 filter mask

Image padding

- The origin of the neighborhood is at the image border → part of the neighborhood will reside outside the image.



251	255	250	255	249	253	255	255	255	255	255	253	251	255	250	255
251	244	255	246	7	255	254	255	255	242	0	255	248	255	255	254
255	255	240	183	0	231	247	255	244	255	0	168	255	241	255	252
254	255	250	12	87	2	255	240	255	0	73	7	253	255	239	255
242	247	255	0	94	0	254	254	241	0	95	0	255	255	248	247
255	255	250	69	87	83	2	255	6	107	79	74	249	245	255	250
255	243	255	156	95	88	0	255	12	58	77	201	239	255	251	253
255	248	255	255	0	74	106	0	85	84	8	250	255	255	242	255
250	255	249	255	255	0	78	4	89	49	251	252	255	255	241	255
255	246	255	252	255	161	57	30	44	150	249	255	255	239	255	255
254	255	165	12	0	53	105	230	119	66	21	0	148	255	255	238
254	255	0	174	215	0	201	252	175	6	178	218	0	255	247	255
255	238	25	213	236	11	232	255	254	7	214	214	14	249	255	255
255	255	0	225	214	1	255	246	253	0	241	213	0	255	245	248
253	255	164	5	0	178	255	255	251	167	4	0	183	255	255	255
255	255	246	255	255	254	253	253	255	255	248	255	252	242	255	254

- Solution
  - Ignore the outside neighbors in the computations of T, or
  - **Pad the image** with a border of some other specified intensity values (e.g. 0s)

Image padding: Zero-padded vs. Replicate

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	251	255	250	255	249	253	255	255	255	255	255	253	251	255	250	255	0
0	251	244	255	246	7	255	254	255	255	242	0	255	248	255	255	254	0
0	255	255	240	183	0	231	247	255	244	255	0	168	255	241	255	252	0
0	254	255	250	12	87	2	255	240	255	0	73	7	253	255	239	255	0
0	242	247	255	0	94	0	254	254	241	0	95	0	255	255	248	247	0
0	255	255	250	69	87	83	2	255	6	107	79	74	249	245	255	250	0
0	255	243	255	156	95	88	0	255	12	58	77	201	239	255	251	253	0
0	255	248	255	255	0	74	106	0	85	84	8	250	255	255	242	255	0
0	250	255	249	255	255	0	78	4	89	49	251	252	255	255	241	255	0
0	255	246	255	252	255	161	57	30	44	150	249	255	255	239	255	255	0
0	254	255	165	12	0	53	105	230	119	66	21	0	148	255	255	238	0
0	254	255	0	174	215	0	201	252	175	6	178	218	0	255	247	255	0
0	255	238	25	213	236	11	232	255	254	7	214	214	14	249	255	255	0
0	255	255	0	225	214	1	255	246	253	0	241	213	0	255	245	248	0
0	253	255	164	5	0	178	255	255	251	167	4	0	183	255	255	255	0
0	255	255	246	255	255	254	253	253	255	255	248	255	252	242	255	254	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Zero-padded

251	251	255	250	255	249	253	255	255	255	255	255	253	251	255	250	255	255
251	251	255	250	255	249	253	255	255	255	255	255	253	251	255	250	255	255
251	251	244	255	246	7	255	254	255	255	242	0	255	248	255	255	254	254
255	255	255	240	183	0	231	247	255	244	255	0	168	255	241	255	252	252
254	254	255	250	12	87	2	255	240	255	0	73	7	253	255	239	255	255
242	242	247	255	0	94	0	254	254	241	0	95	0	255	255	248	247	247
255	255	255	250	69	87	83	2	255	6	107	79	74	249	245	255	250	250
255	255	243	255	156	95	88	0	255	12	58	77	201	239	255	251	253	253
255	255	248	255	255	0	74	106	0	85	84	8	250	255	255	242	255	255
250	250	255	249	255	255	0	78	4	89	49	251	252	255	255	241	255	255
255	255	246	255	252	255	161	57	30	44	150	249	255	255	239	255	255	255
254	254	255	165	12	0	53	105	230	119	66	21	0	148	255	255	238	238
254	254	255	0	174	215	0	201	252	175	6	178	218	0	255	247	255	255
255	255	238	25	213	236	11	232	255	254	7	214	214	14	249	255	255	255
255	255	255	0	225	214	1	255	246	253	0	241	213	0	255	245	248	248
253	253	255	164	5	0	178	255	255	251	167	4	0	183	255	255	255	255
255	255	255	246	255	255	254	253	253	255	255	248	255	252	242	255	254	254
255	255	255	246	255	255	254	253	253	255	255	248	255	252	242	255	254	254

Replicate

Source: http://www.cs.uregina.ca/Links/class-info/425/Lab3/

Image padding: Demonstration

Original

Zero-padded

Symmetric

The results of the four different boundary options Exaggerated boundaries are made just for visual effect.

Replicate

Circular

Image padding: Demonstration

Original

Zero-padded

Replicate

The results of the four different boundary options. The filter used is a 5×5 averaging filter

Source: http://www.cs.uregina.ca/Links/class-info/425/Lab3/

Symmetric

Circular

Spatial correlation vs. convolution

- **Correlation** is the process of moving a filter mask over the image and computing the sum of products at each location

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

- The mechanics of **convolution** are the same, except that the filter is first rotated by 180°

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

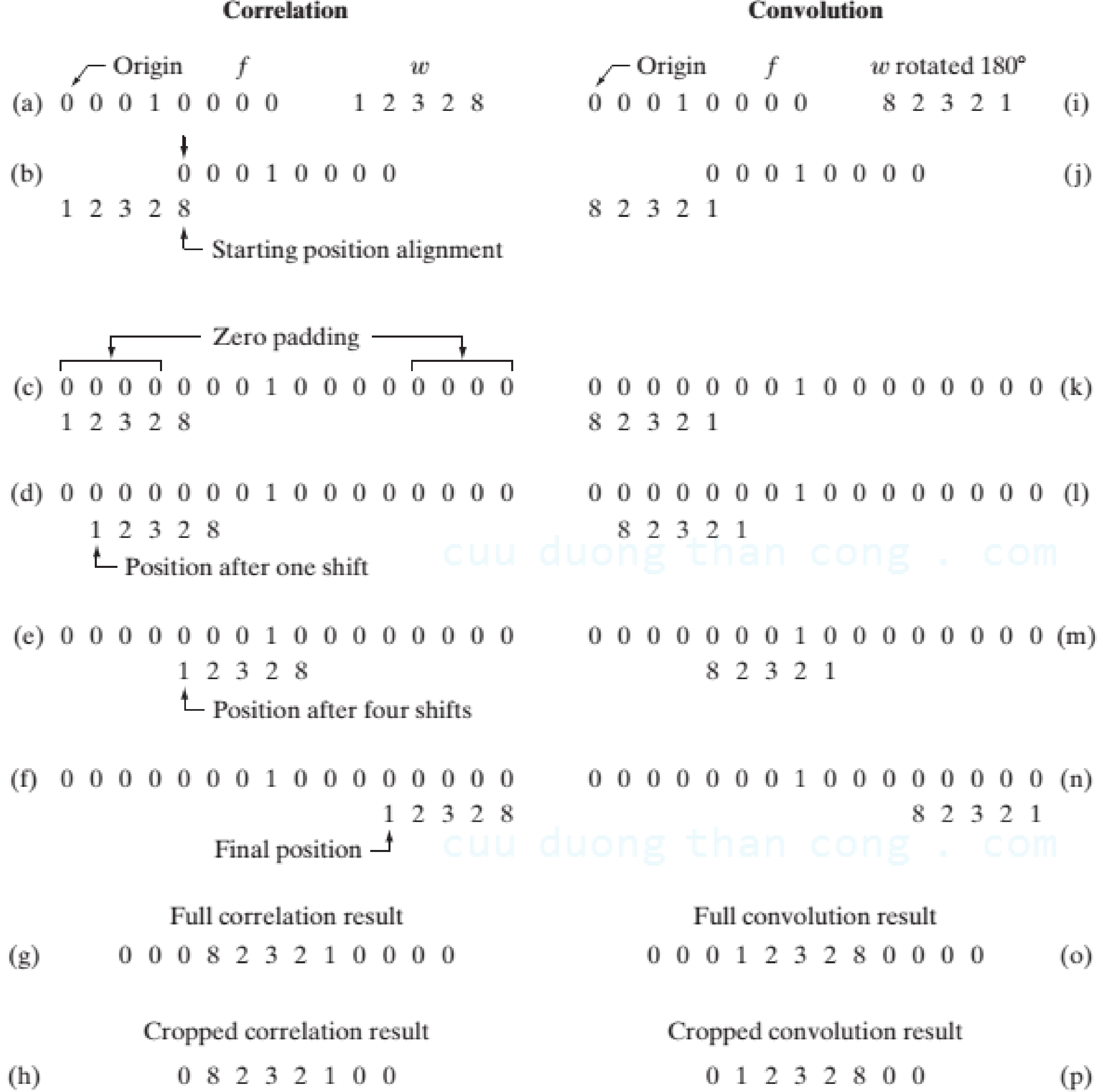
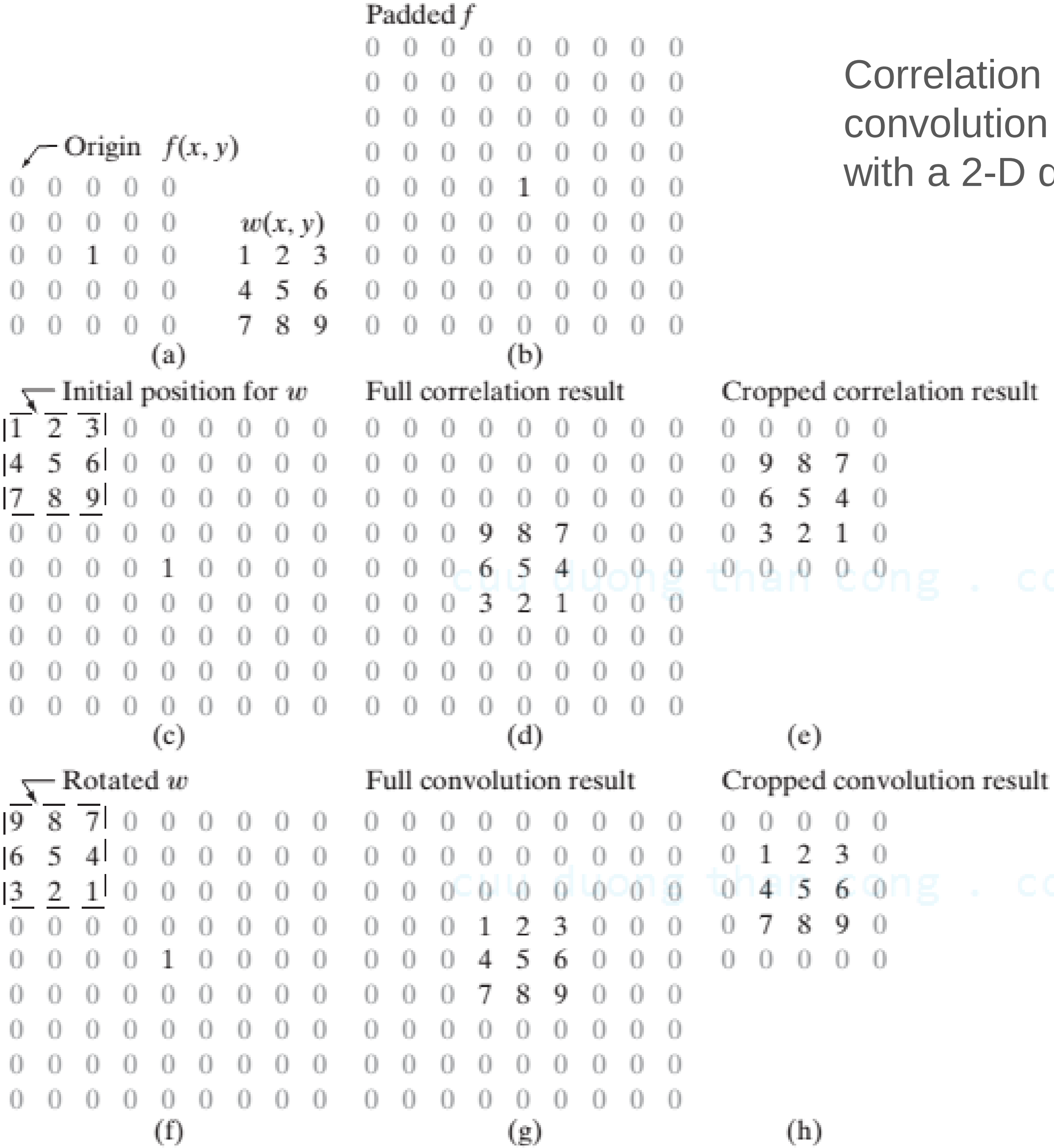
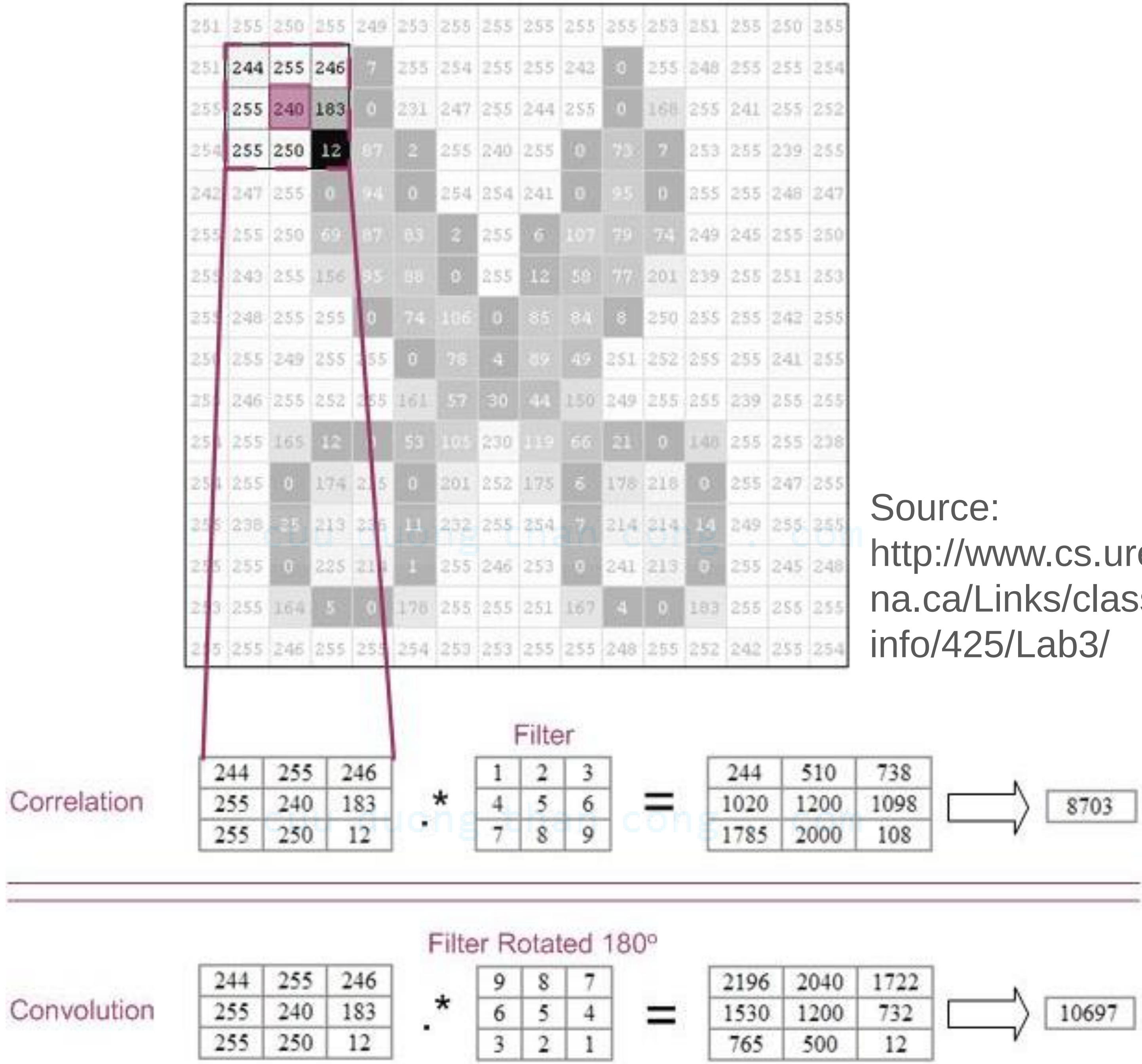


Illustration of 1-D correlation and convolution of a filter with a discrete unit impulse. Note that correlation and convolution are functions of *displacement*.

Correlation (middle row) and convolution (last row) of a 2-D filter with a 2-D discrete, unit impulse.





Source: http://www.cs.uregina.ca/Links/class-info/425/Lab3/

Spatial correlation vs. convolution

- Correlation is a function of displacement of the filter
  - The first value corresponds to zero displacement of the filter, the second corresponds to one unit displacement, and so on
- Correlation of a filter mask w with a **discrete unit impulse** yields a rotated version of w at the location of the impulse
  - Discrete unit impulse: a function that contains a single 1 with the rest being 0s
- If the filter mask is symmetric, correlation and convolution yield the same result

Section 3.5

SMOOTHING SPATIAL FILTERS

Averaging filter

- The value of every pixel in an image is replaced by the average of the intensity levels in the neighborhood defined by a filter mask

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

$\frac{1}{9} \times$

1	1	1
1	1	1
1	1	1

Averaging filter (box filter)

$\frac{1}{16} \times$

1	2	1
2	4	2
1	2	1

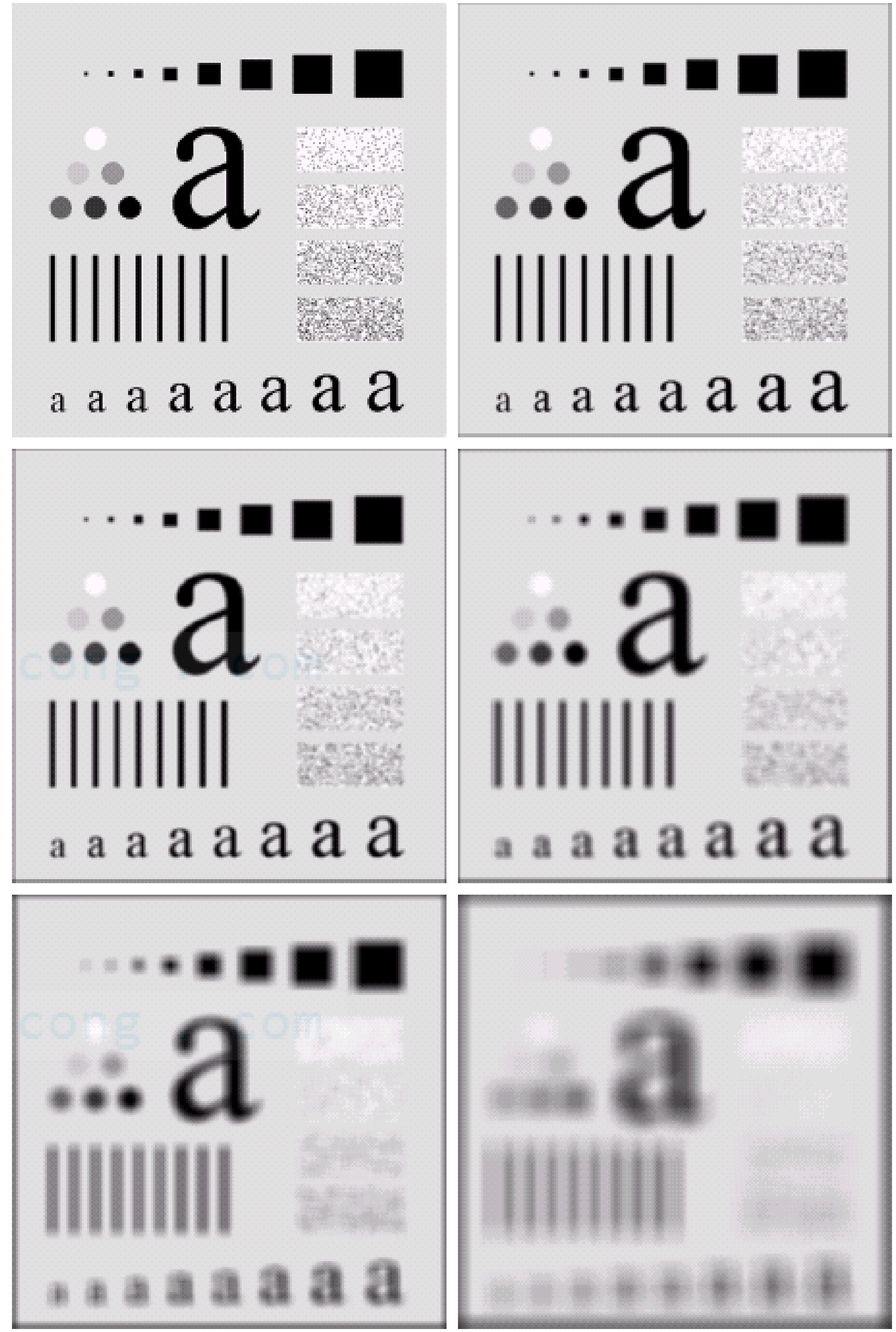
Weighted averaging filter

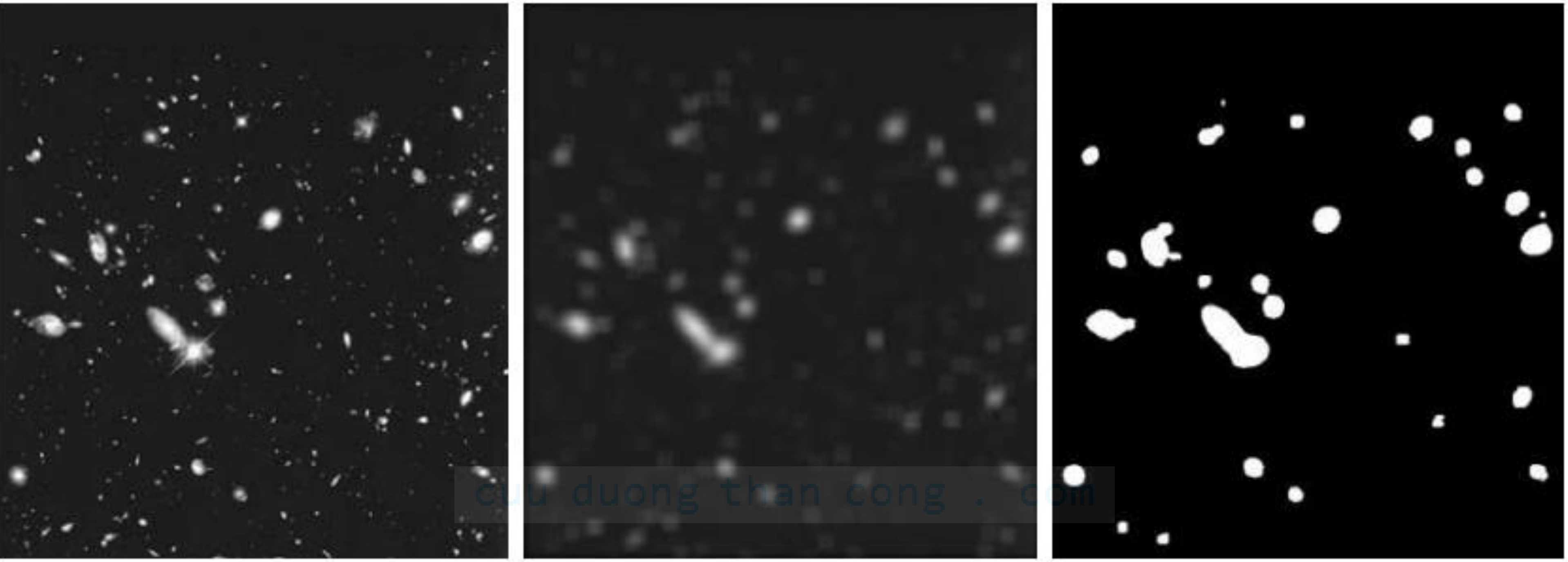
Averaging filter: Applications

- Smoothing results in an image with reduced “sharp” transitions in intensities
- Noise reduction: the most obvious application
  - Random noise typically includes sharp transitions in intensity levels
- Smoothing of false contours that result from using an insufficient number of intensity levels
- Reduction of “irrelevant” detail in an image
  - “irrelevant”: pixel regions that are small with respect to the size of the filter mask
- ***Side-effect:*** edges (which always are desirable features of an image) are also blurred

a	b
c	d
e	f

(a) Original image, of size 500×500 pixels.
(b)–(f) Results of smoothing with square averaging filter masks of sizes 3, 5, 9, 15, and 35, respectively.
The black squares at the top are of sizes 3, 5, 9, 15, 25, 35, 45, and 55 pixels, respectively; their borders are 25 pixels apart.
The letters at the bottom range in size from 10 to 24 points, in increments of 2 points; the large letter at the top is 60 points.
The vertical bars are 5 pixels wide and 100 pixels high; their separation is 20 pixels.
The diameter of the circles is 25 pixels, and their borders are 15 pixels apart; their intensity levels range from 0% to 100% black in increments of 20%.
The background of the image is 10% black.
The noisy rectangles are of size 50×120 pixels.

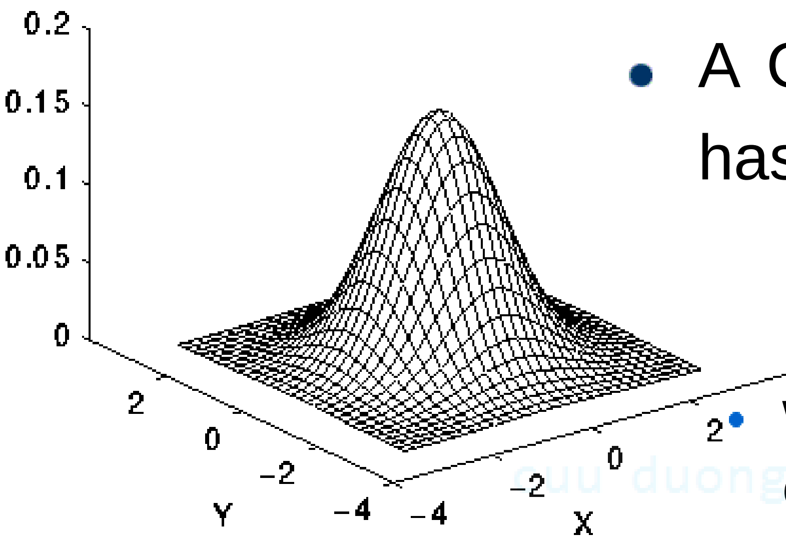




a b c

(a) Image of size 528×485 pixels from the Hubble Space Telescope. (b) Image filtered with a 15×15 averaging mask. (c) Result of thresholding (b).

Gaussian filter



- A Gaussian function of two variables has the basic form

$$h(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$$

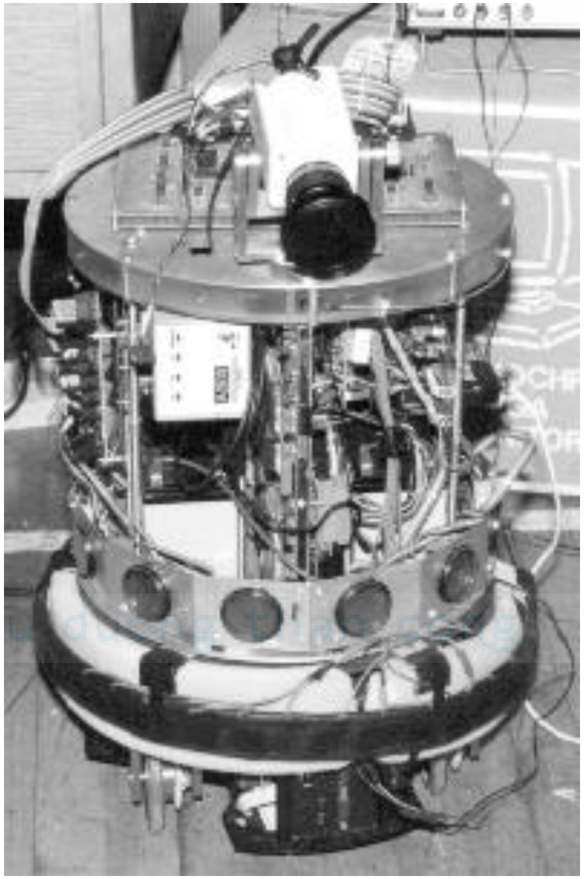
  - where σ is the standard deviation, and coordinates x and y are integers

- A $m \times n$ **Gaussian filter** is generated by sampling the Gaussian function about its center
  - E.g. a 3×3 Gaussian filter has $w_1 = h(-1,-1), w_2 = h(-1,0), \dots, , w_9 = h(1,1)$

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

$$\frac{1}{273}$$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1



a b c

(a) A 5×5 Gaussian mask of $\sigma = 1$. (b) Original image. (c) Image filtered with the Gaussian filter in (a).

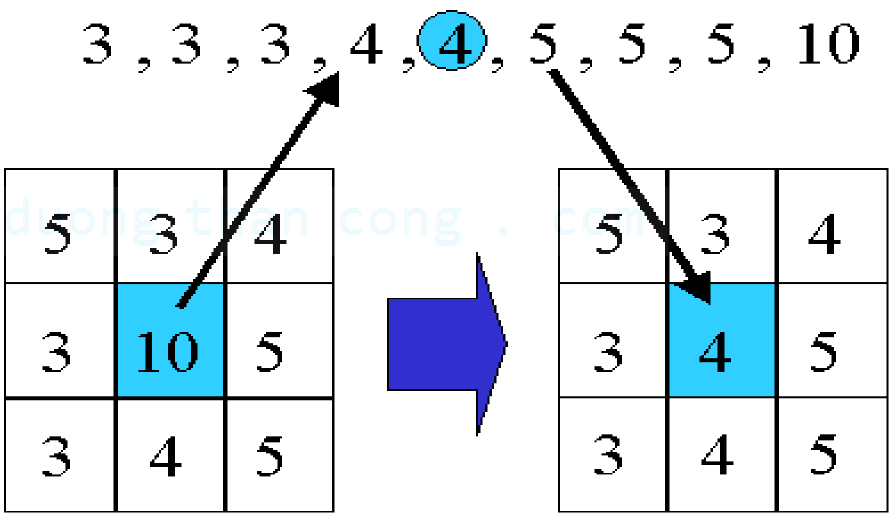
Source: http://homepages.inf.ed.ac.uk/rbf/HIPR2/gsmooth.htm

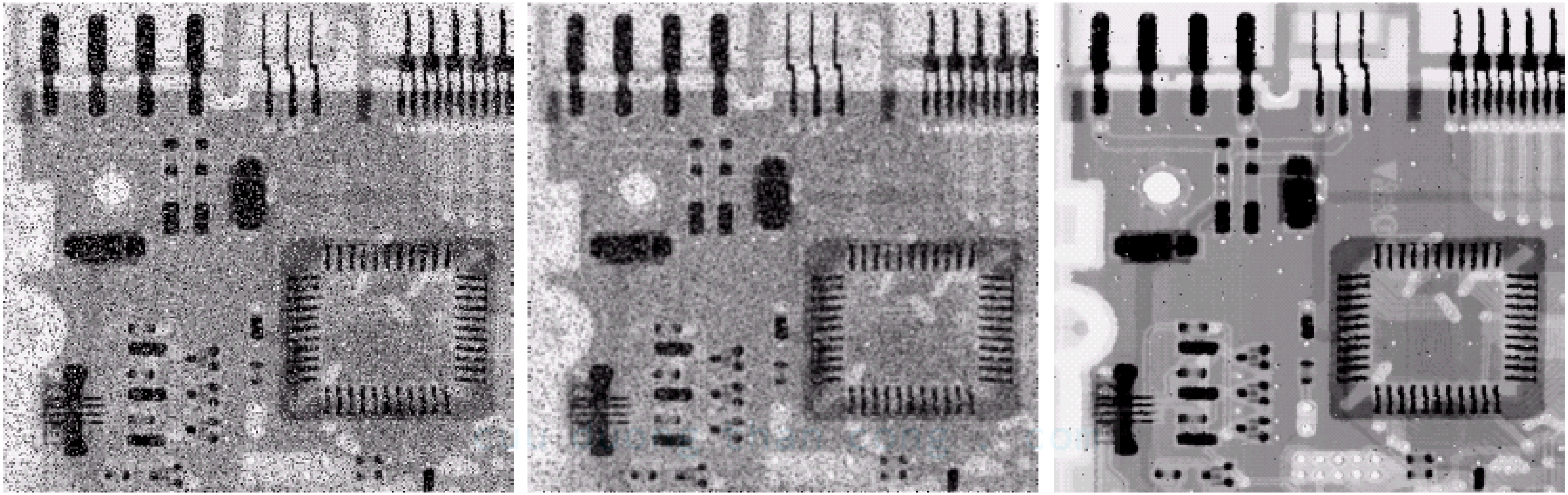
Order-statistic (nonlinear) filter

- The pixels in the image area encompassed by the filter are **ranked** and then the value of the center pixel are replaced with the value determined by the ranking result
- Common filters: **median**, **max** and **min filters**
  - The value of a pixel is replaced by the median/maximum/minimum of the intensity values in the neighborhood of that pixel, respectively

Order-statistic filter: Median filter

- Determine the median, ξ, of a set of values such that
  - Half the values in the set are less than or equal to ξ, and
  - Half are greater than or equal to ξ
- Excellent noise-reduction capabilities
  - Considerably less blurring than linear filters of similar size
  - Particularly effective in the presence of impulse noise (or salt-and-pepper noise)





a b c

(a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a averaging mask. (c) Noise reduction with a median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Section 3.6

SHARPENING SPATIAL FILTERS

Foundation

- Any definition for a first derivative $\frac{\partial f}{\partial x} = f(x+1) - f(x)$

  (1) must be zero in areas of constant intensity;

  (2) must be nonzero at the onset of an intensity step or ramp; and

  (3) must be nonzero along ramps.

- Similarly, any definition of a second derivative $\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x)$

  (1) must be zero in constant areas;

  (2) must be nonzero at the onset and end of an intensity step or ramp; and

  (3) must be zero along ramps of constant slope

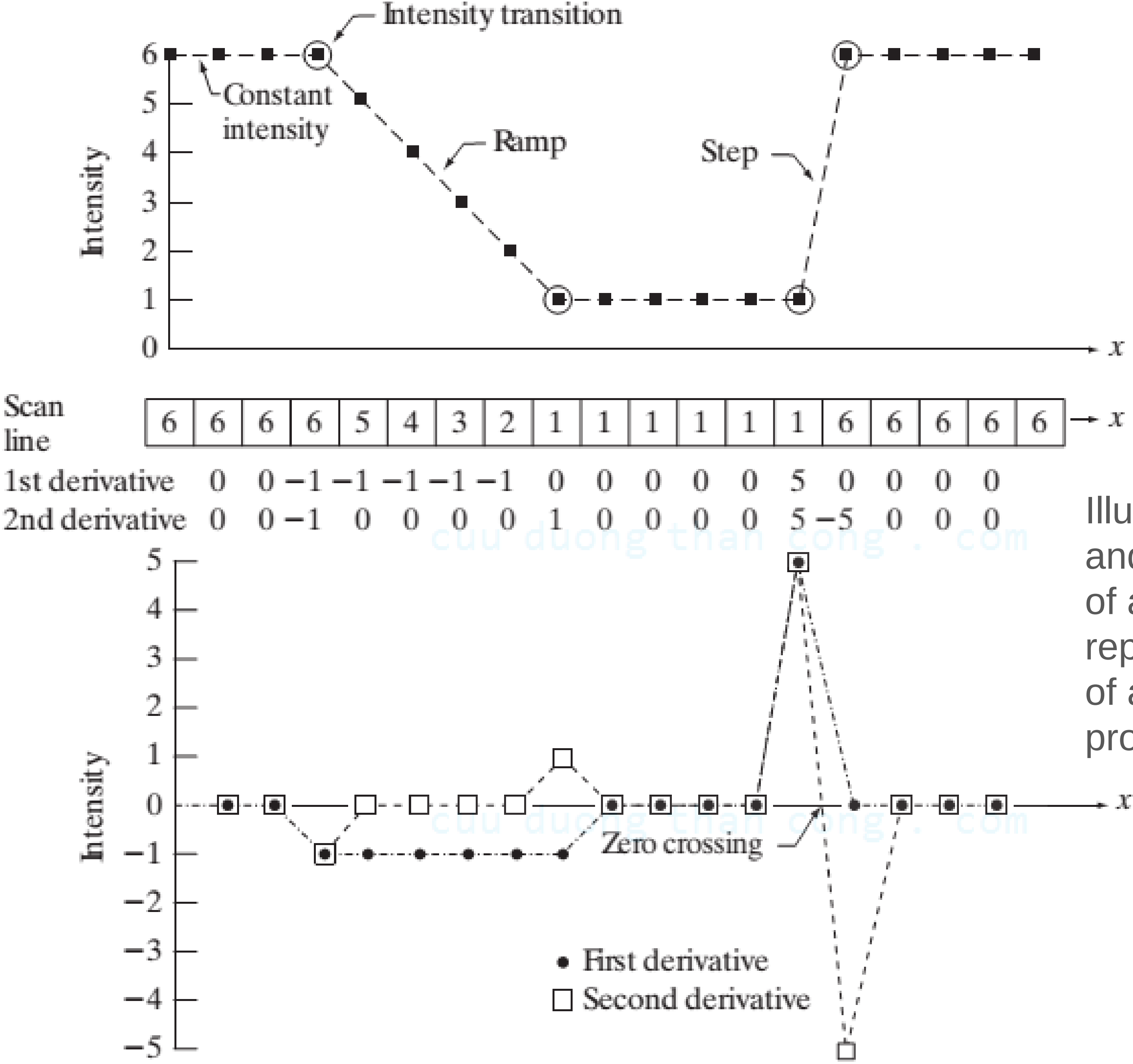
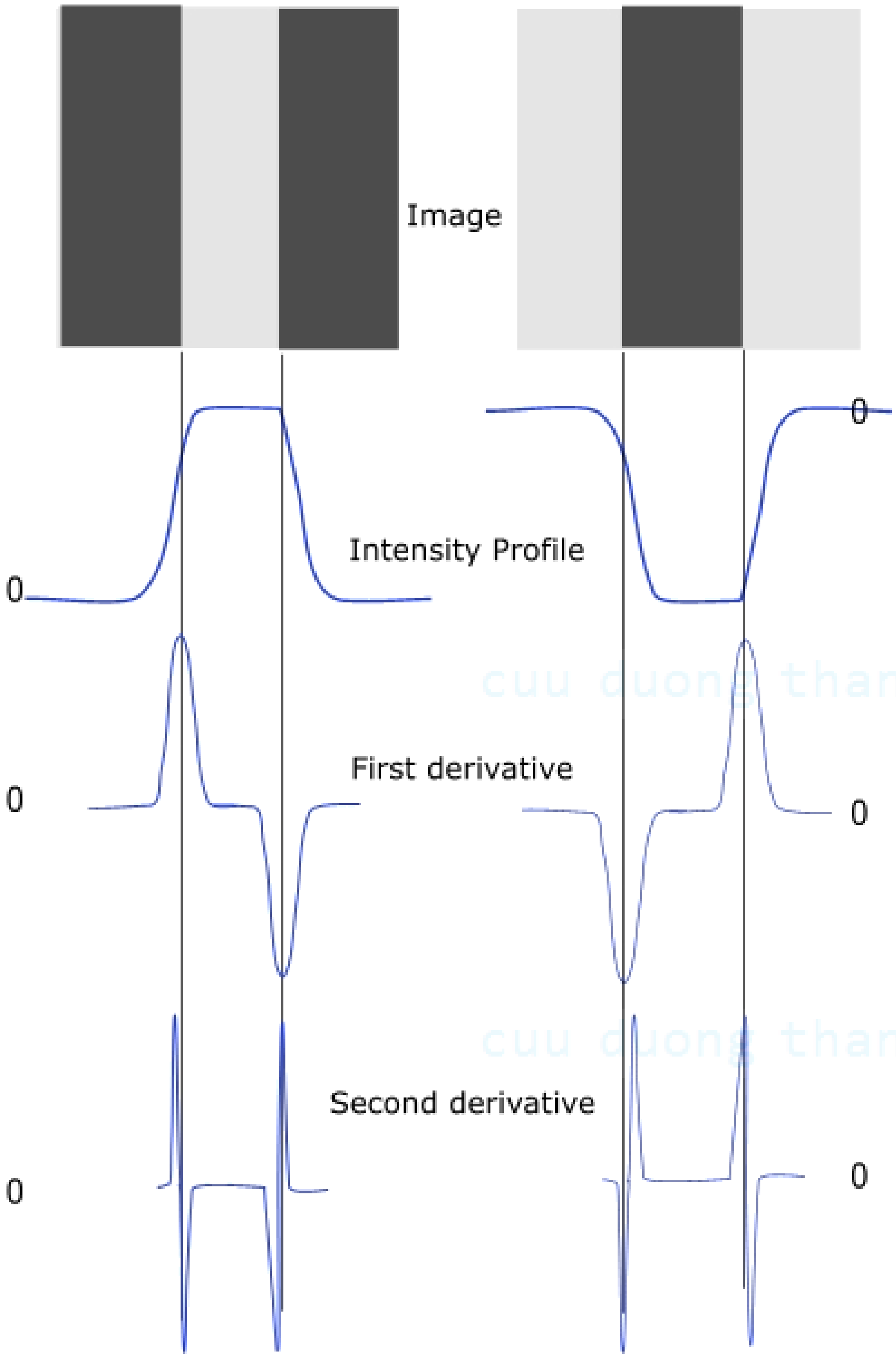


Illustration of the first and second derivatives of a 1-D digital function representing a section of a horizontal intensity profile from an image



Source: http://mipav.cit.nih.gov/pubwiki/index.php/Edge_Detection:_Zero_X_Laplacian

Second derivative: Laplacian operator

- The Laplacian for a function (image) of two variables is

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

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

- An isotropic linear operator
  - Rotating the image and then applying the filter gives the same result as applying the filter to the image and then rotating the result, and
  - Derivatives of any order are linear operations

Second derivative: Laplacian operator

- The discrete Laplacian of two variables is

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

0	1	0
1	−4	1
0	1	0

- The diagonal directions can also be incorporated

$$\nabla^2 f(x,y) = f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) + f(x-1,y-1) + f(x-1,y+1) + f(x+1,y-1) + f(x+1,y+1) \mathbf{-8f(x,y)}$$

1	1	1
1	−8	1
1	1	1

Second derivative: Laplacian operator

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

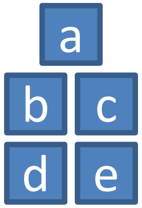
Two other implementations of Laplacian found frequently in practice

Second derivative: Laplacian filter

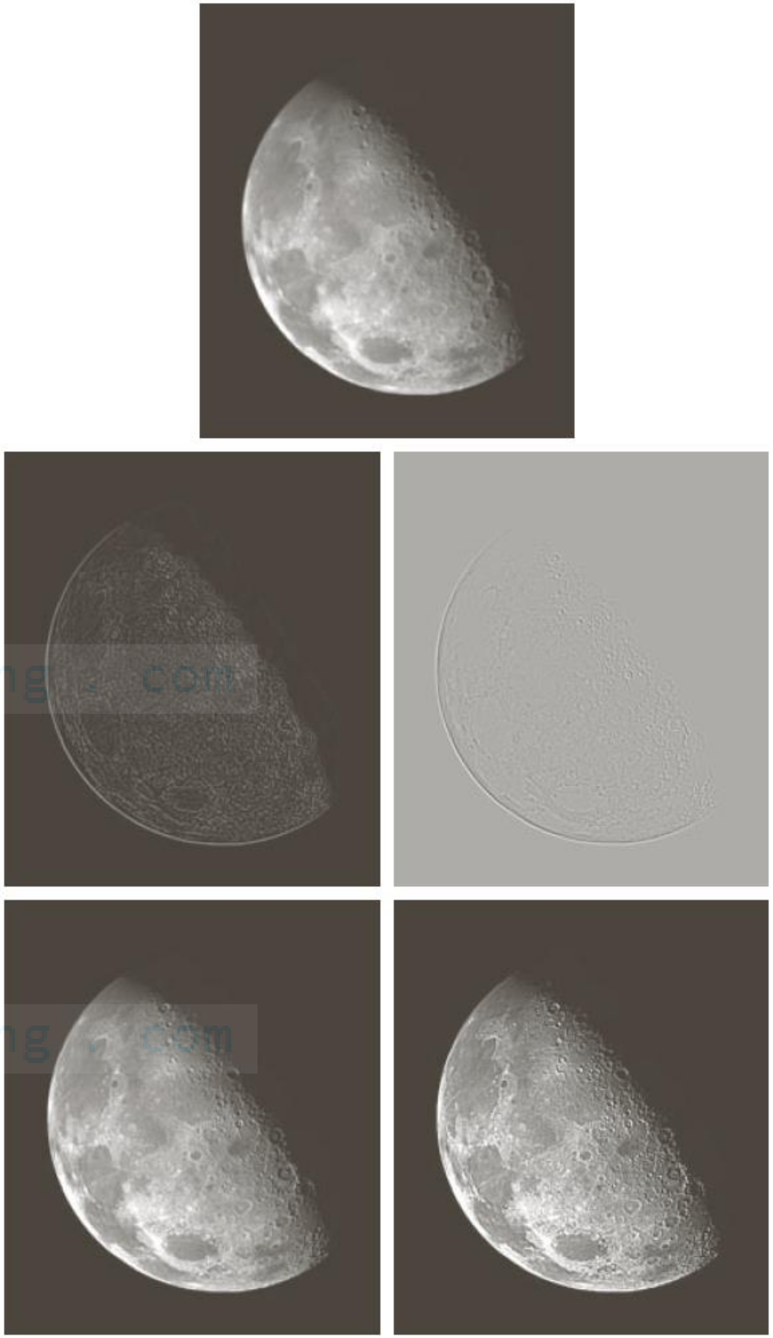
- The **Laplacian filter** for image sharpening is given by

$$\boldsymbol{g(x,y) = f(x,y) + c\left[\nabla^2 f(x,y)\right]}$$

  - where $f(x, y)$ and $g(x, y)$ are the input and sharpened images, respectively. The constant $c = -1$ if the term at the center of the filter is negative, otherwise, $c = 1$



(a) Blurred image of the North Pole of the moon. (b) Laplacian without scaling. (c) Laplacian with scaling. (d) Image sharpened using the 2-direction mask with negative central term. (e) Result of using the 4-direction mask with negative central term. (Original image courtesy of NASA.)



Unsharp masking and highboost filtering

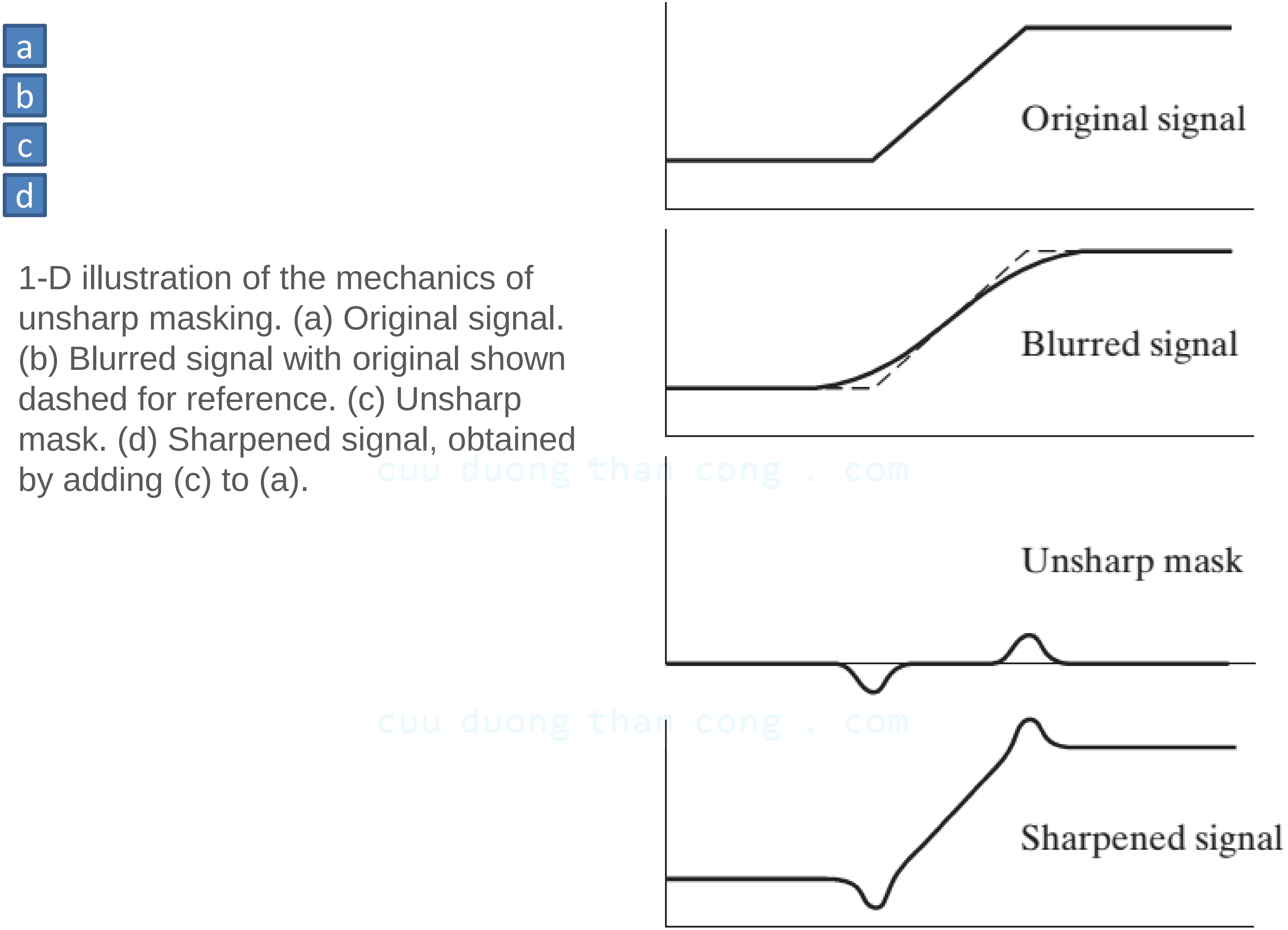
- The unsharp masking consists of the following steps:
  1. Blur the original image $f(x, y)$ to produce a blurred image $\bar{f}(x, y)$
  2. Subtract the blurred image from the original to create a mask

$$g_{mask}(x, y) = f(x, y) - \bar{f}(x, y)$$

  3. Add a weighted portion of the mask back to the original image

$$g(x, y) = f(x, y) + k * g_{mask}(x, y)$$

  - The process is referred to as *unsharp masking* when $k = 1$ or *highboost filtering* when $k > 1$. Meanwhile, $k < 1$ de-emphasizes the contribution of the unsharp mask
- This process has been used for many years by the printing and publishing industry to sharpen images



1-D illustration of the mechanics of unsharp masking. (a) Original signal. (b) Blurred signal with original shown dashed for reference. (c) Unsharp mask. (d) Sharpened signal, obtained by adding (c) to (a).

a
b
c
d
e

(a) Original image. (b) Result of blurring with a Gaussian filter. (c) Unsharp mask. (d) Result of using unsharp masking. (e) Result of using highboost filtering ($k = 4.5$)

First derivative: Gradient

- The gradient of a function f at location (x, y) is defined as

$$\nabla f \equiv grad(f) \equiv \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

- The magnitude (length) of vector ∇f is described by

$$M(x, y) = mag(\nabla f) = \sqrt{g_x^2 + g_y^2}$$

  - $M(x, y)$ is referred to as *gradient image*

First derivative: Gradient

- The gradient vector is linear, but the magnitude of the vector is not.
  - Derivatives are linear, square and square root operations are not
- The partial derivatives are not rotation invariant, but the magnitude of the gradient vector is

- Approximate the squares and square root operations by absolute values → more suitable computationally

$$M(x, y) \approx |g_x| + |g_y|$$

  - The relative changes in intensity is preserved, but the isotropic property is lost in general.

First derivative: Robert cross operators

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

−1	0
0	1

$$g_x = (z_9 - z_5)$$

0	−1
1	0

$$g_y = (z_8 - z_6)$$

a b c

(a) A region of an image (the zs are intensity values). (b)–(c) Roberts cross gradient operators. All the mask coefficients sum to zero, as expected of a derivative operator.

- $$M(x,y) = \left[(z_9 - z_5)^2 + (z_8 - z_6)^2\right]^{1/2}$$
  $$\approx |z_9 - z_5| + |z_8 - z_6|$$

First derivative: Sobel operators

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

$$g_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$

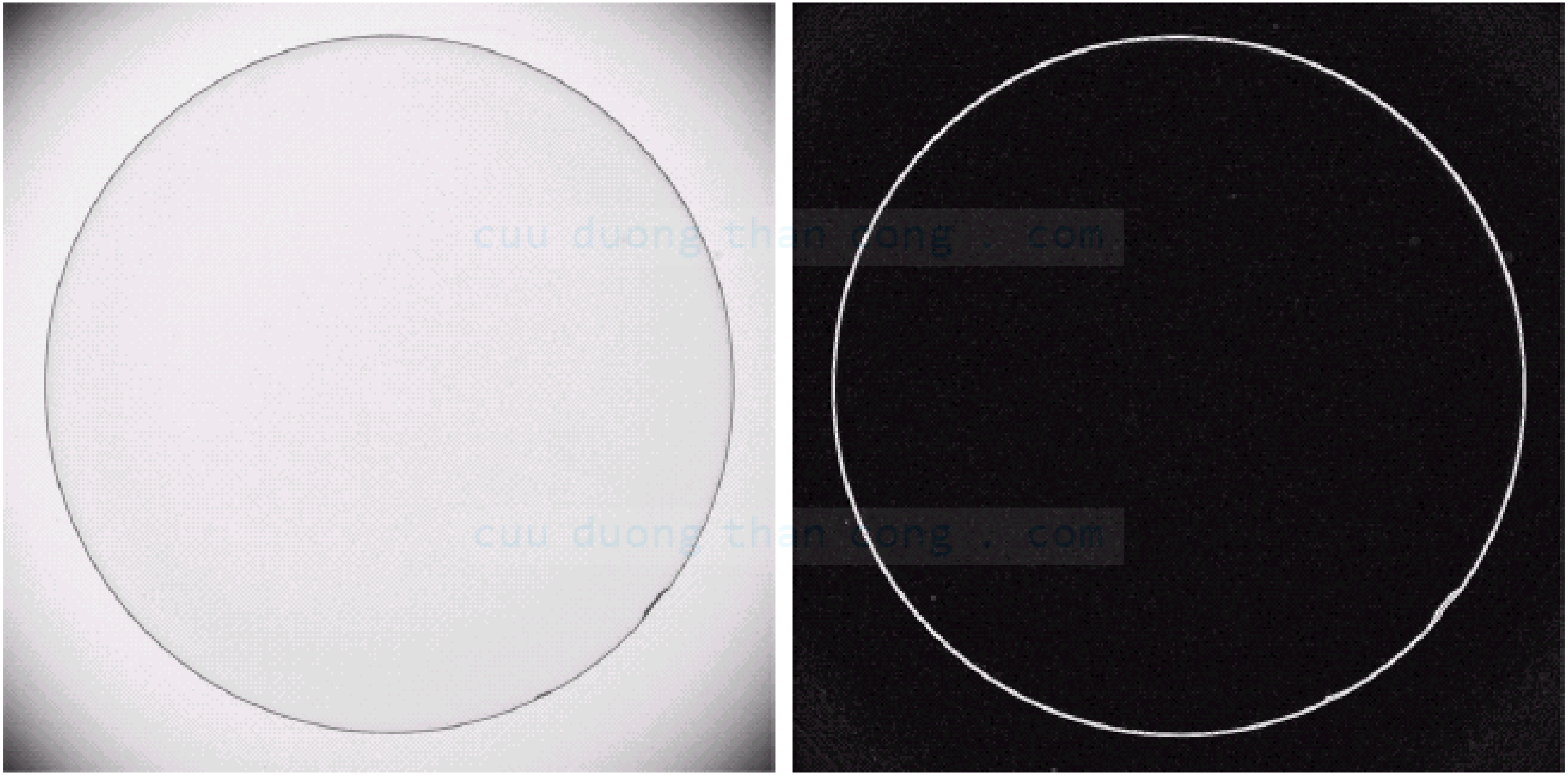
$$g_y = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$$

a b c (a) A region of an image (the zs are intensity values). (b)–(c) Sobel operators. All the mask coefficients sum to zero, as expected of a derivative operator.

- $M(x,y) \approx |(z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)| + |(z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)|$

a b

(a) Optical image of contact lens (note defects on the boundary at 4 and 5 o'clock). (b) Sobel gradient. (Original image courtesy of Pete Sites, Perceptics Corporation.)



Section 3.7

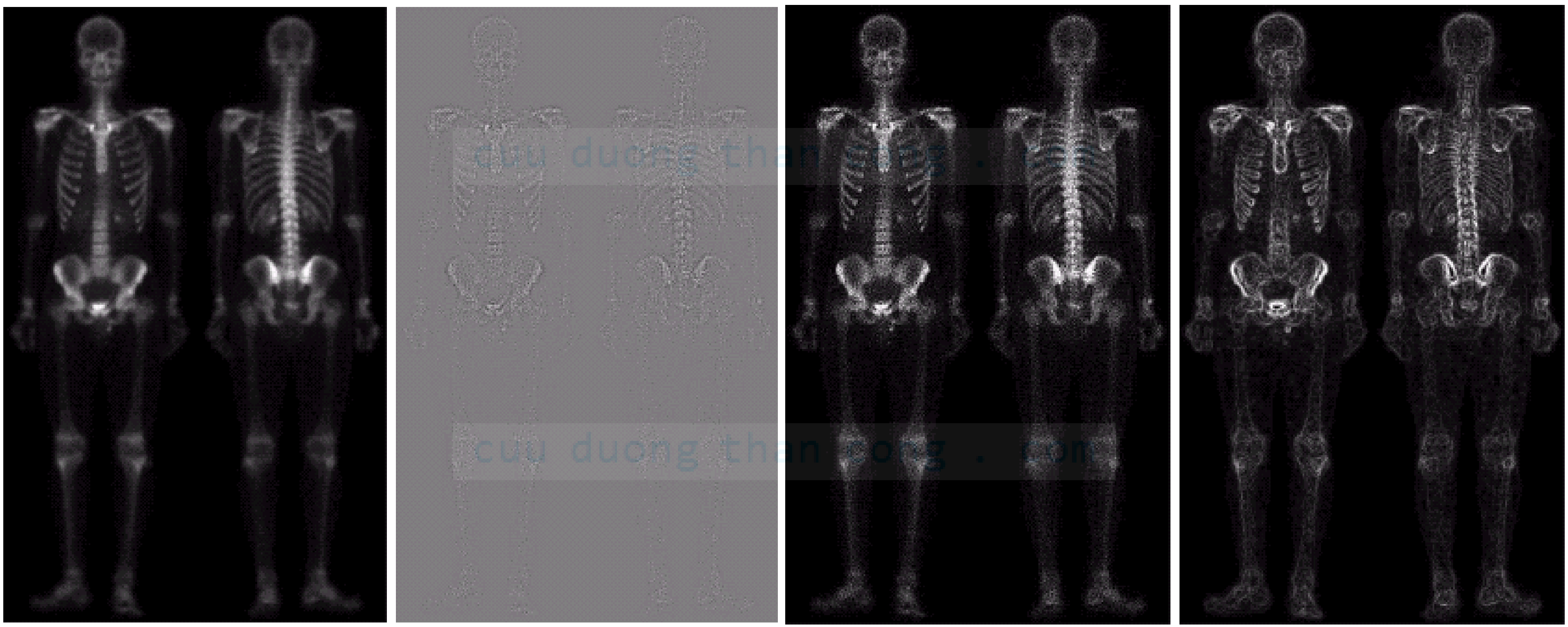
COMBINING SPATIAL ENHANCEMENT METHODS

Example of combining several methods

- **Objective:** Enhance the image by sharpening and by bringing out more of the skeletal detail
- **Ideas**
  - Utilize the Laplacian to highlight fine detail, and the gradient to enhance prominent edges
  - A smoothed version of the gradient image will be used to mask the Laplacian image
  - Finally, increase the dynamic range of the intensity levels by using an intensity transformation

a b c d

(a) Image of whole body bone scan.
(b) Laplacian of (a).
(c) Sharpened image obtained by adding (a) and (b).
(d) Sobel gradient of (a)



e f g h

(Continued)

(e) Sobel image smoothed with an averaging filter.

(f) Mask image formed by the product of (c) and (e).

(g) Sharpened image obtained by the sum of (a) and (f).

(h) Final result obtained by applying a power-law transformation to (g).

Compare (g) and (h) with (a). (Original image courtesy of G.E. Medical Systems.)

References

- Rafael C. Gonzalez, Richard E. Woods, “Digital Image Processing”, 3rd edition, 2008. Chapter 3

- Images are obtained from the above materials and Google