上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

Institute of Media, Information, and Network

生命科学技术学院
School of Life Sciences and Biotechnology

图像滤波与增强

Hongkai Xiong
熊红凯
http://min.sjtu.edu.cn

Department of Electronic Engineering
Shanghai Jiao Tong University

14 February. 2020

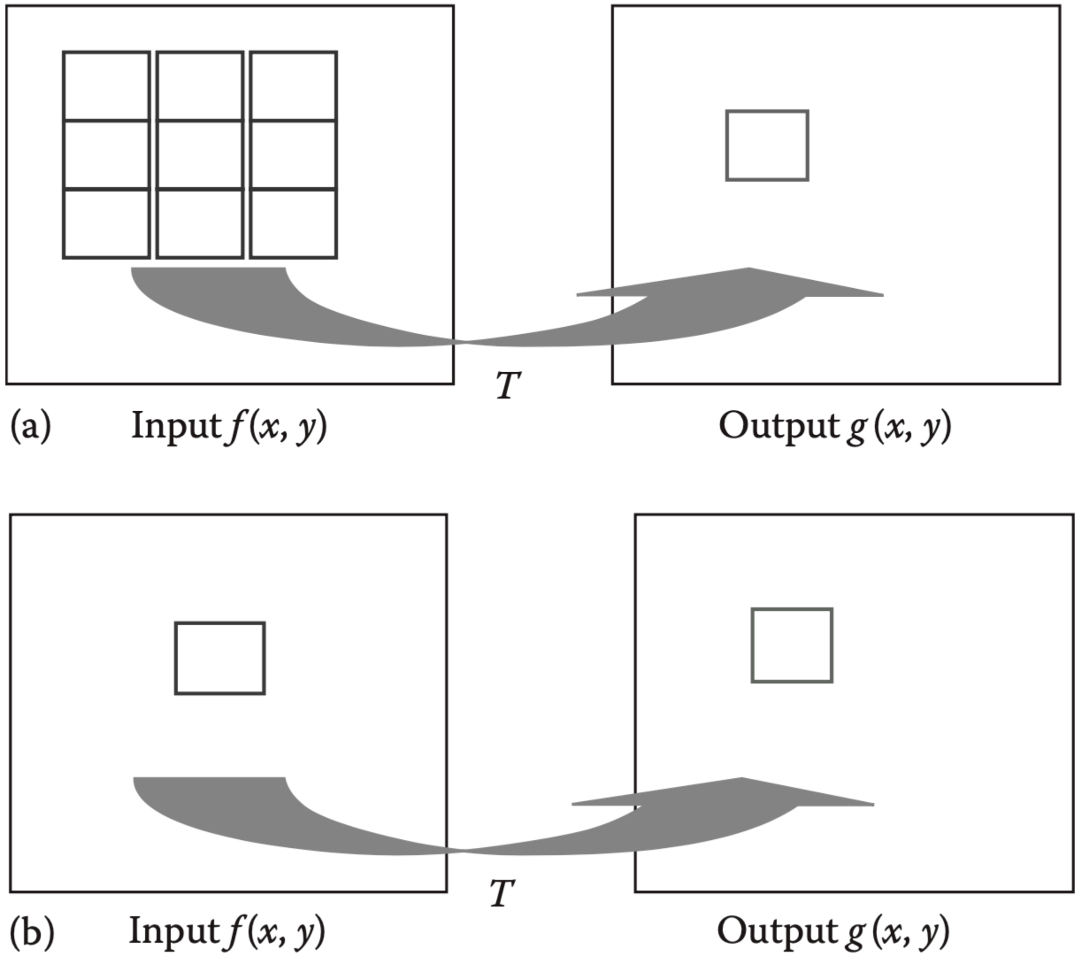
Topical today

- Image Enhancement
- Spatial Filtering
- Frequency Filtering

Image Enhancement

Point processing and mask processing

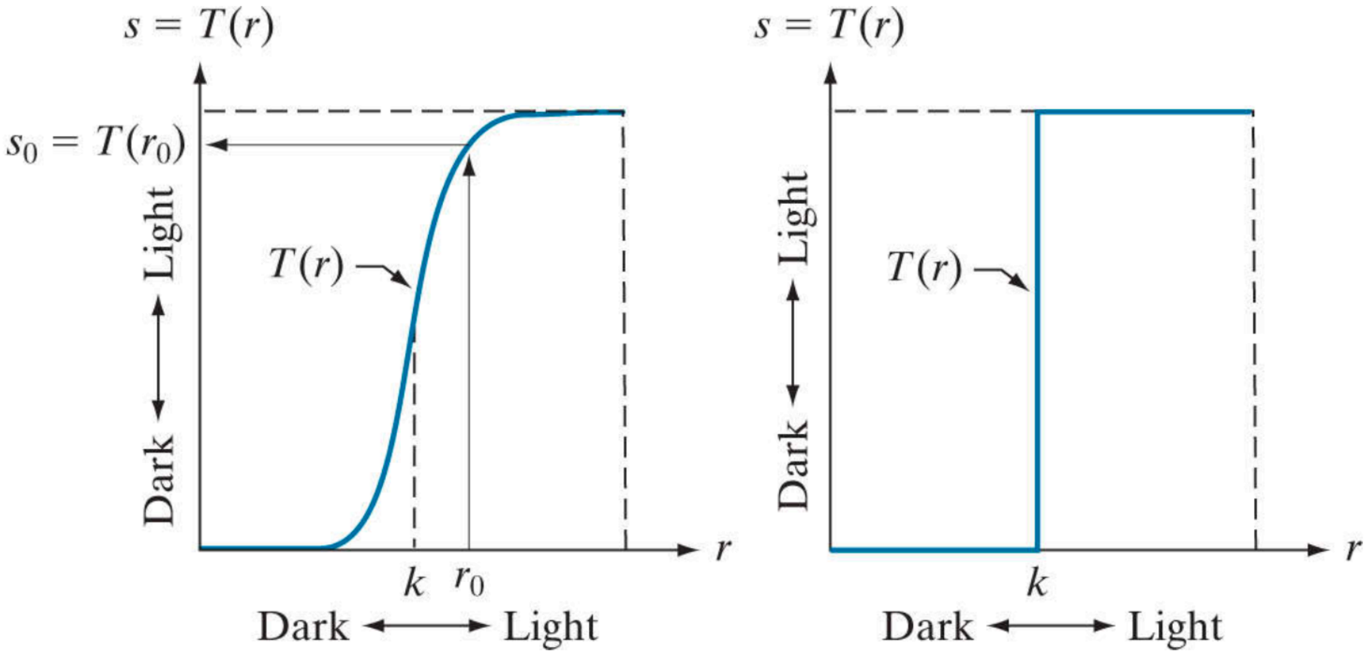
- Space-domain image enhancement techniques can be further classified into two general categories: point processing and mask processing



Schematic diagram of (a) mask processing and (b) point processing techniques

Point processing

- What can they do?
- What's the form of T?
- **Important:**every pixel for himself, which means spatial information completely lost!

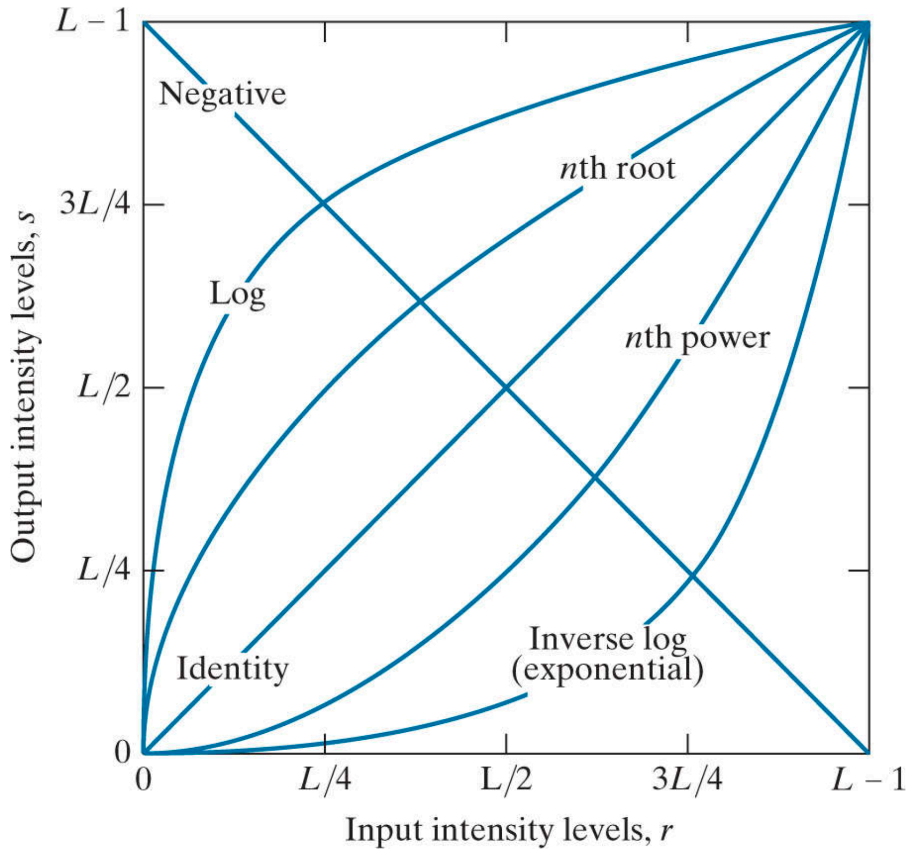


Intensity transformation functions. (a) Contrast stretching function. (b) Thresholding function.

Point processing

- Some basic intensity transformation functions

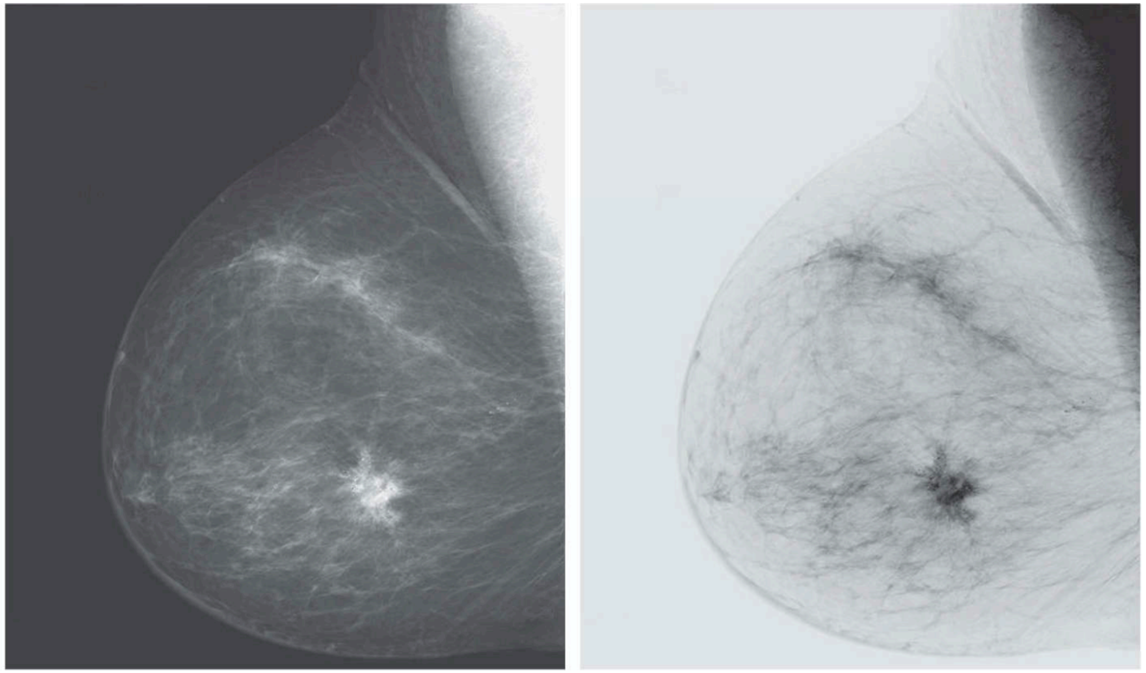
Each curve was scaled independently so that all curves would fit in the same graph. Our interest here is on the shapes of the curves, not on their relative values.



Point processing

- Image negatives

Reversing the intensity levels of a digital image in this manner produces the equivalent of a photographic negative. This type of processing is used, for example, in enhancing white or gray detail embedded in dark regions of an image, especially when the black areas are dominant in size.



Left: A digital mammogram. Right: Negative image obtained using s=L-1-r . (Image left Courtesy of General Electric Medical Systems.)

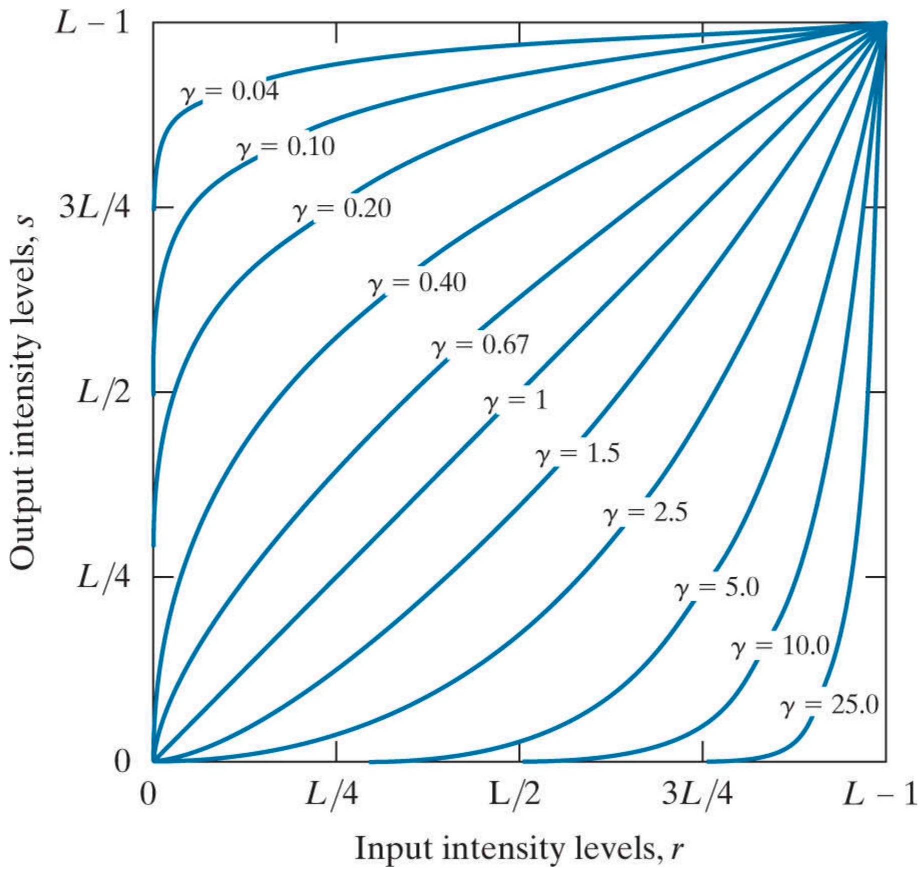
Point processing

- Power-Law(Gamma) Transformations

Power-law transformations have the form

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

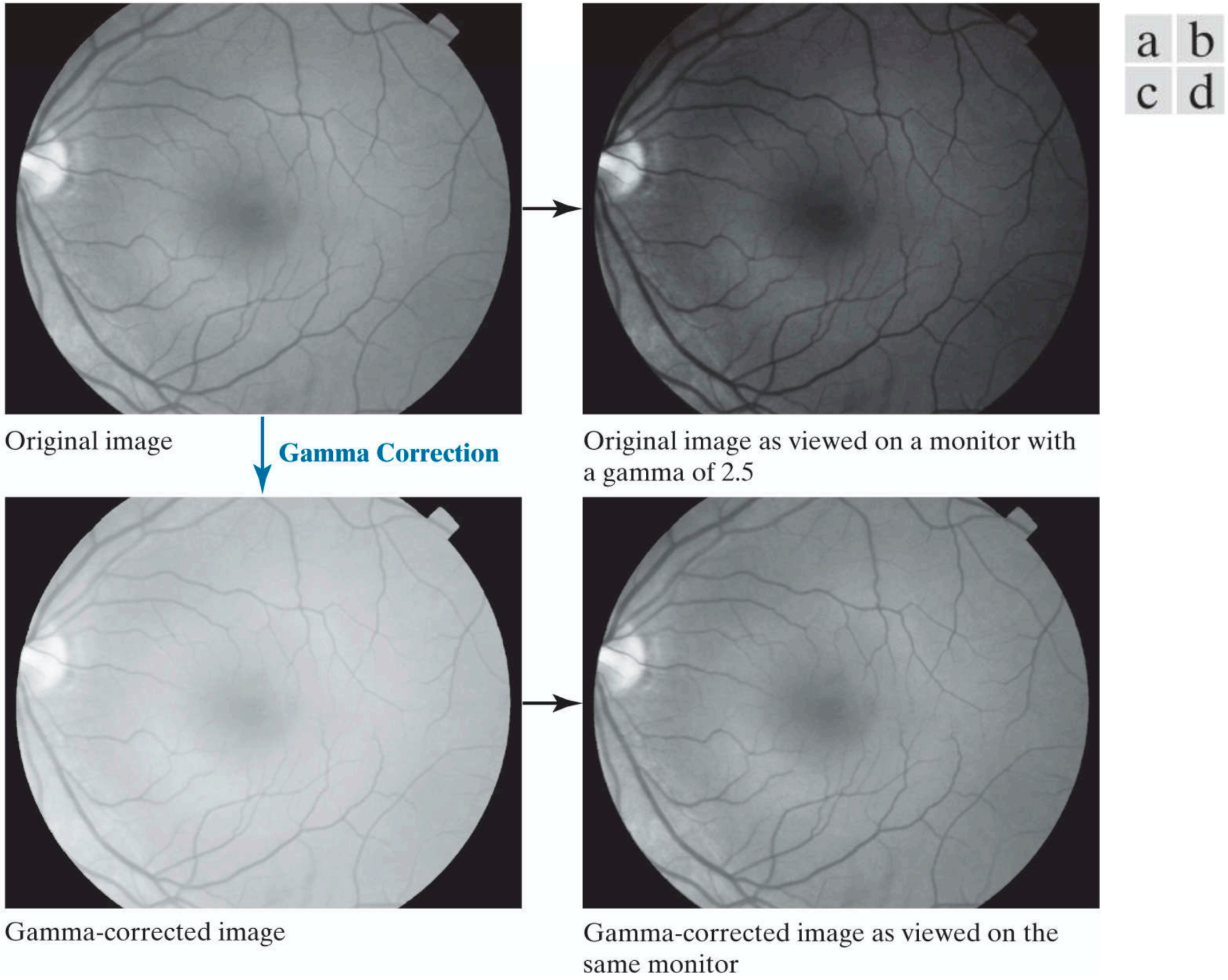
where c and are positive constants. Figure right shows plots of s as a function of r for various values of γ. The response of many devices used for image capture, printing, and display obey a power law. By convention, the exponent in a power- law equation is referred to as gamma. The process used to correct these power-law response phenomena is called gamma correction or gamma encoding.



Point processing

- Power-law transformations

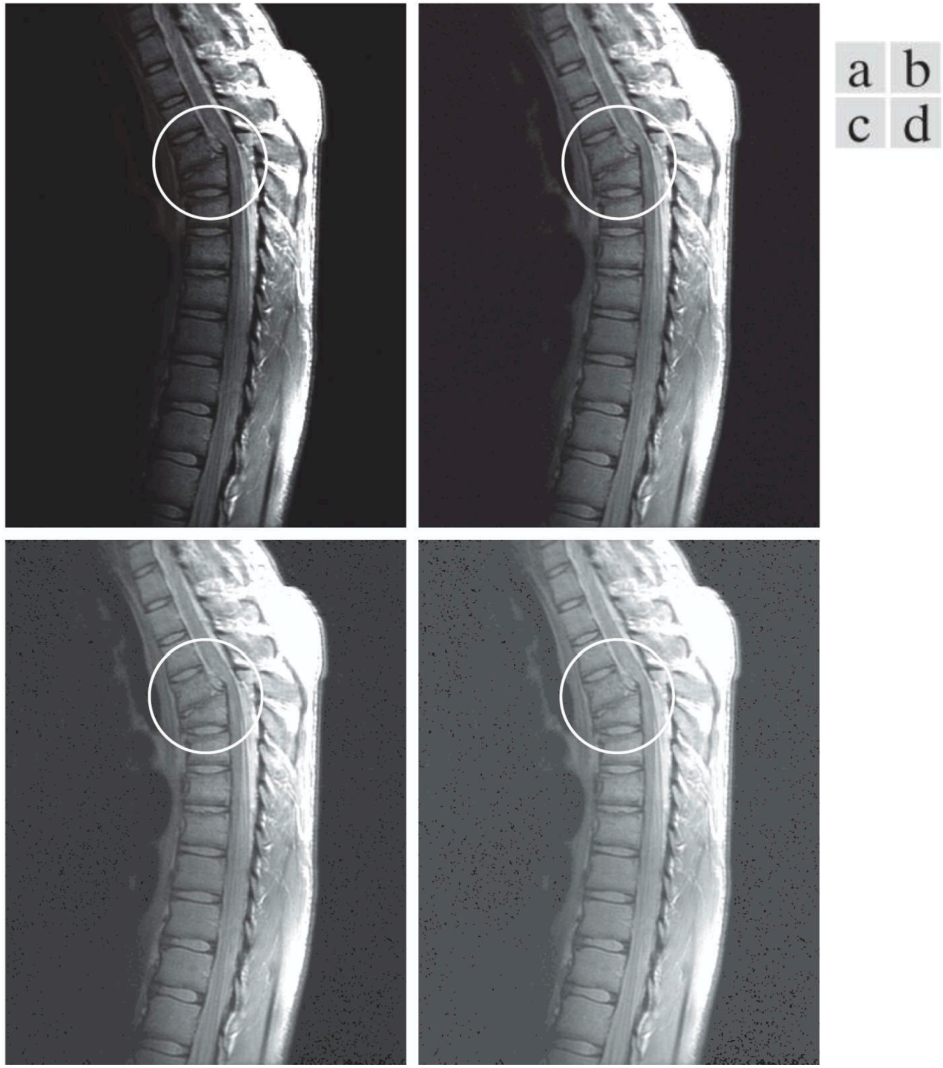
(a) Image of a human retina. (b) Image as as it appears on a monitor with a gamma setting of 2.5 (note the darkness). (c) Gamma- corrected image. (d) Corrected image, as it appears on the same monitor (compare with the original image). (Image (a) courtesy of the National Eye Institute, NIH)



Point processing

- Power-law transformations

  In addition to gamma correction, power-law transformations are useful for general-purpose contrast manipulation. shows a magnetic resonance image (MRI) of a human upper thoracic spine with a fracture dislocation. The fracture is visible in the region highlighted by the circle. Because the image is predominantly dark, an expansion of intensity levels is desirable. This can be accomplished using a power-law transformation with a fractional exponent. The values of gamma corresponding to images (b) through (d) are 0.6, 0.4, and 0.3, respectively (c = 1 in all cases). Observe that as gamma decreased from 0.6 to 0.4, more detail became visible.



Point processing

- **Power-law transformations**

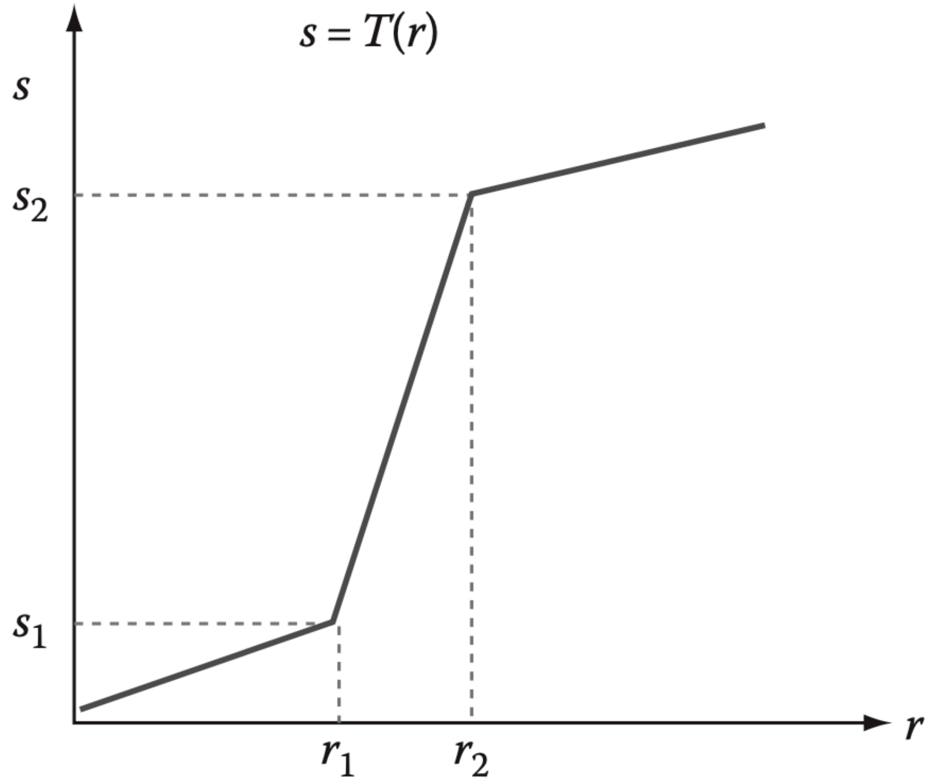
  Figure right shows the opposite problem of that presented above. The image to be processed now has a washed-out appearance, indicating that a compression of intensity levels is desirable. This can be accomplished with $s = cr^{\gamma}$ using values of γ greater than 1. The results of processing (a) with = 3.0, 4.0, and 5.0 are shown in (b) through (d), respectively. Suitable results were obtained using gamma values of 3.0 and 4.0. The latter result has a slightly more appealing appearance because it has higher contrast.

a	b
c	d

Point processing

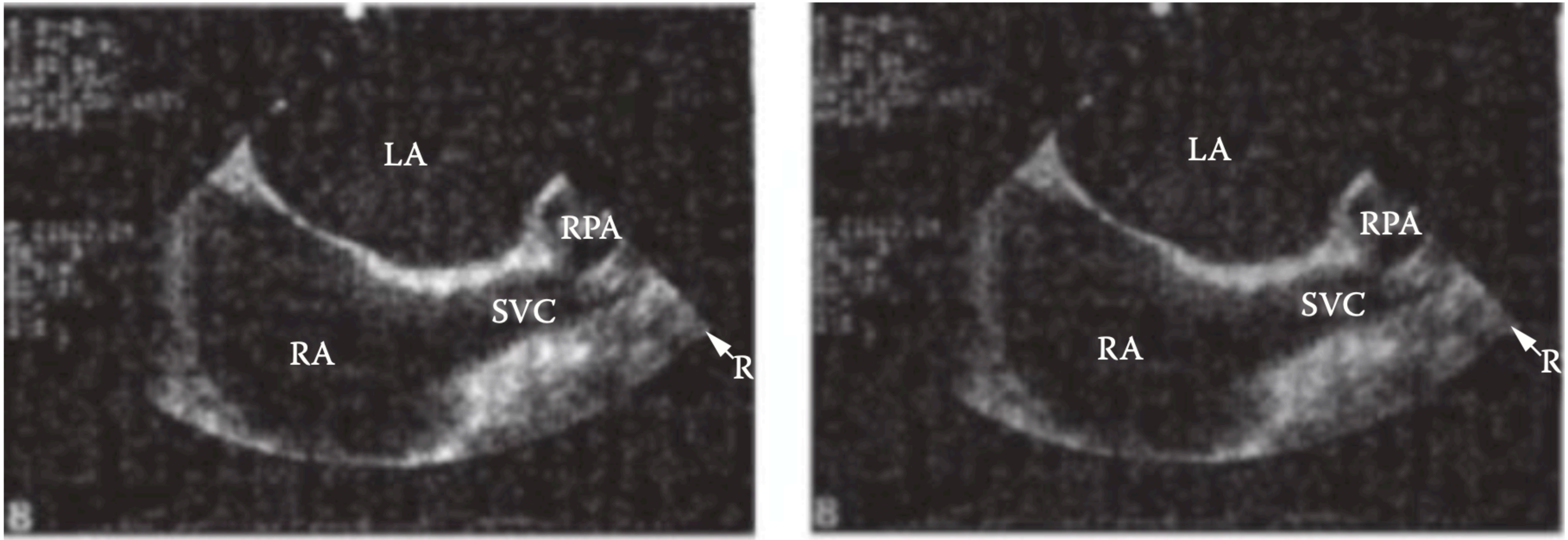
- Contrast Stretching

Contrast enhancement with stretching is a method to create better visibility of a particular range of gray level that corresponds to the object to be studied. In practice, we often select these values such that the interval [s1, s2] covers the gray-level range of the object of interest and the interval [r1, r2] provides the desired range of gray level for a better visibility of the object in the target image



Contrast enhancement using point processing

Contrast Enhancement



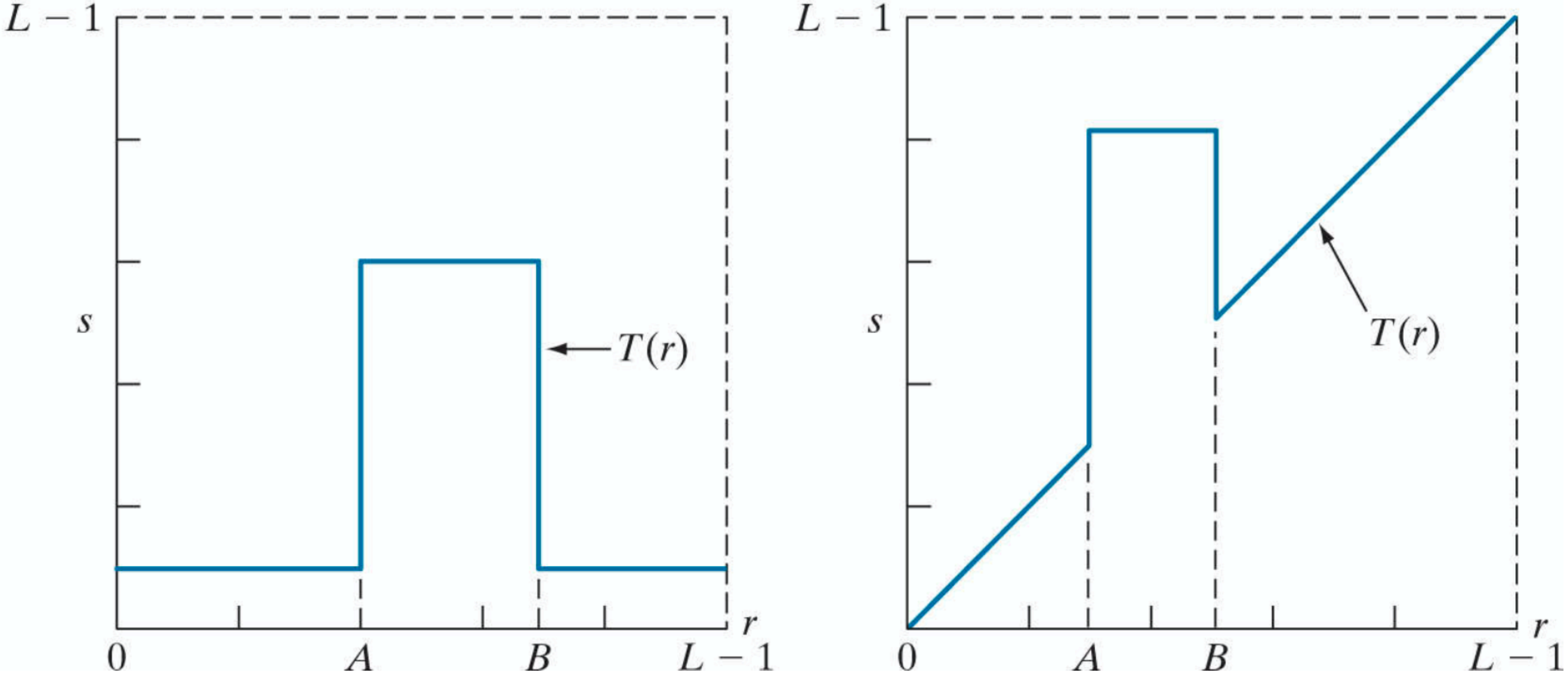
(a) (b)

(a) Original image and (b) image after point processing (Courtesy of Andre D'Avila, MD, Heart Institute (InCor), University of Sao Paulo, Medical School, Sao Paulo, Brazil)

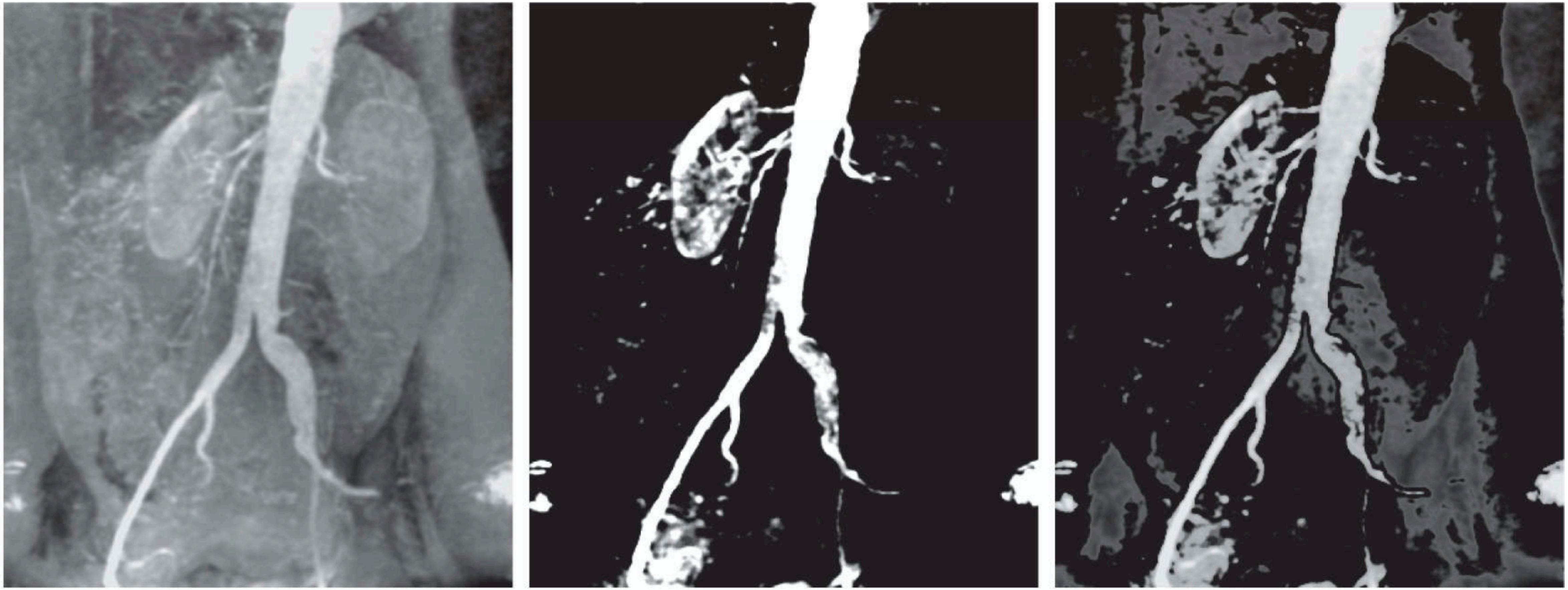
Point processing

- Intensity-level slicing

There are applications in which it is of interest to highlight a specific range of intensities in an image. Some of these applications include enhancing features in satellite imagery, such as masses of water, and enhancing flaws in X-ray images. (a) This transformation function highlights range [A, B] and reduces all other intensities to a lower level. (b) This function highlights range [A, B] and leaves other intensities unchanged.



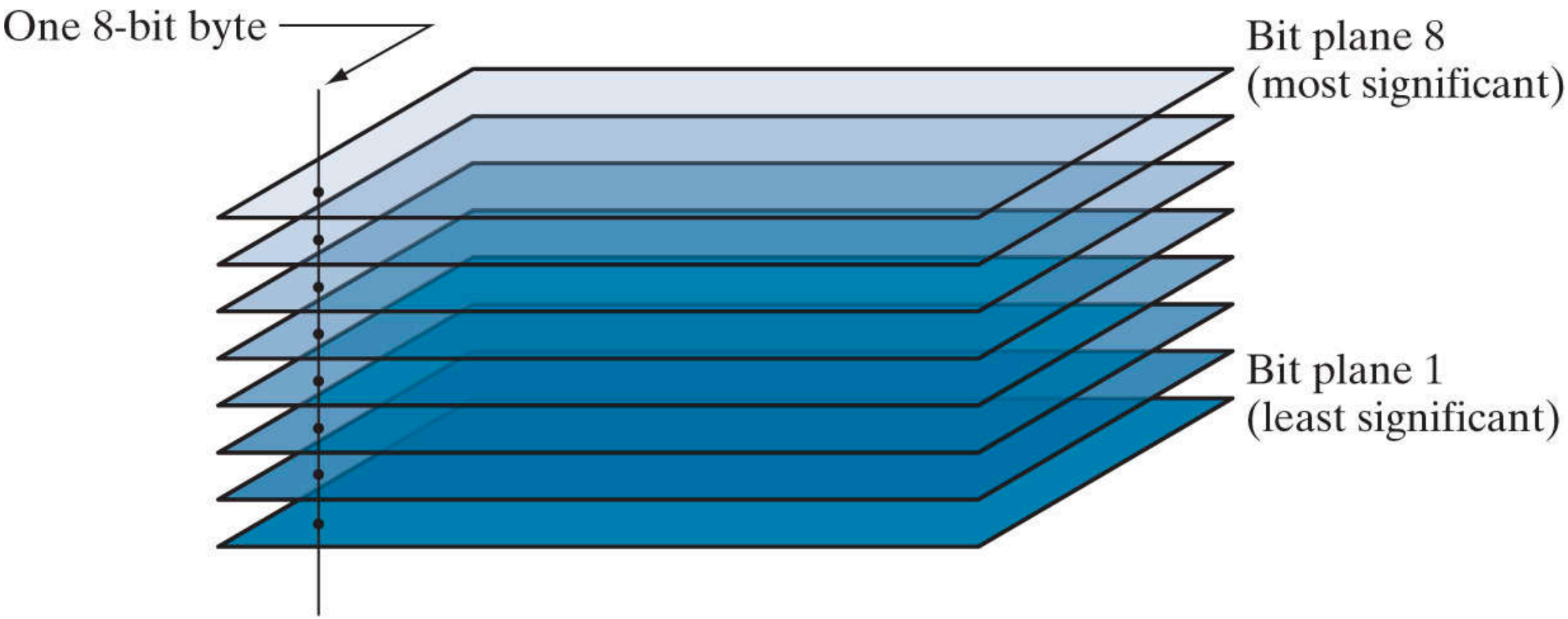
Intensity-Level Slicing



(a) Aortic angiogram. (b) Result of using a slicing transformation of the type illustrated in above (a) , with the range of intensities of interest selected in the upper end of the gray scale. (c) Result of using the transformation in above (b) , with the selected range set near black, so that the grays in the area of the blood vessels and kidneys were preserved.

Point processing

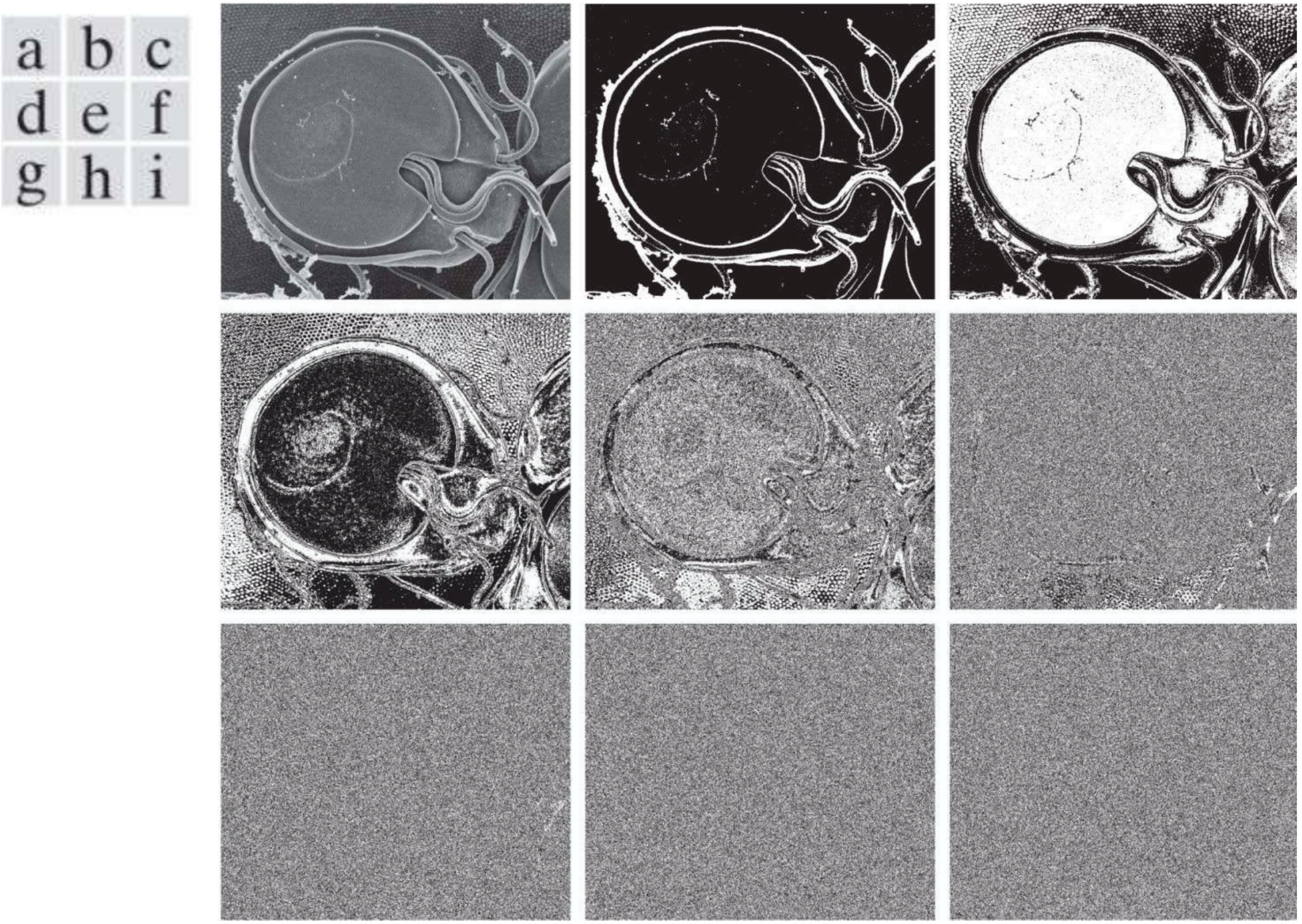
- Bit-plane slicing



Pixel values are integers composed of bits. For example, values in a 256-level gray-scale image are composed of 8 bits (one byte). Instead of highlighting intensity-level ranges, we could highlight the contribution made to total image appearance by specific bits.

Point processing

- Bit-plane slicing



(a) An 8-bit gray-scale image of size 837 × 988 pixels. (b) through (i) Bit planes 8 through 1, respectively, where plane 1 contains the least significant bit. Each bit plane is a binary image. Figure (a) is an SEM image of a trophozoite that causes a disease called giardiasis.

(a) shows an 8-bit gray-scale image and (b) through (i) are its eight, one-bit planes, with (b) corresponding to the highest (most significant) bit plane. Observe that the highest-order four planes, especially the higher two, contain a great deal of the visually significant data.

Point processing

- Bit-plane slicing

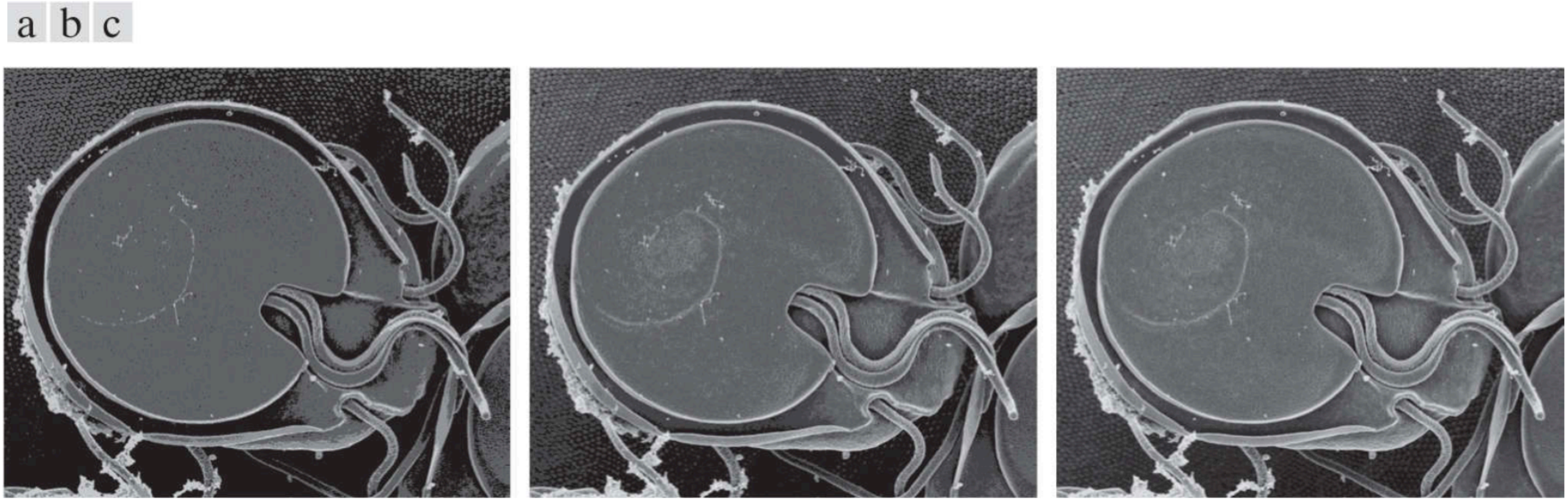


Image reconstructed from bit planes: (a) 8 and 7; (b) 8, 7, and 6; (c) 8, 7, 6, and 5.

Point processing

- Histogram Processing
  - Histograms are the basis for numerous spatial domain processing techniques
  - Histograms are the statistical diagrams of gray level distribution
- For continuous gray level

$$p(r) = \lim_{\Delta r \to 0} \frac{A(r+\Delta r) - A(r)}{\Delta r \cdot A}$$

Point processing

- Histogram Processing
  - For discrete gray level

$$h(r_k) = n_k, \quad r_k \in [0, L-1]$$

where r_k is the k-th intensity value and n_k is the number of pixels in the image with intensity r_k. Thus, a normalized histogram is given by

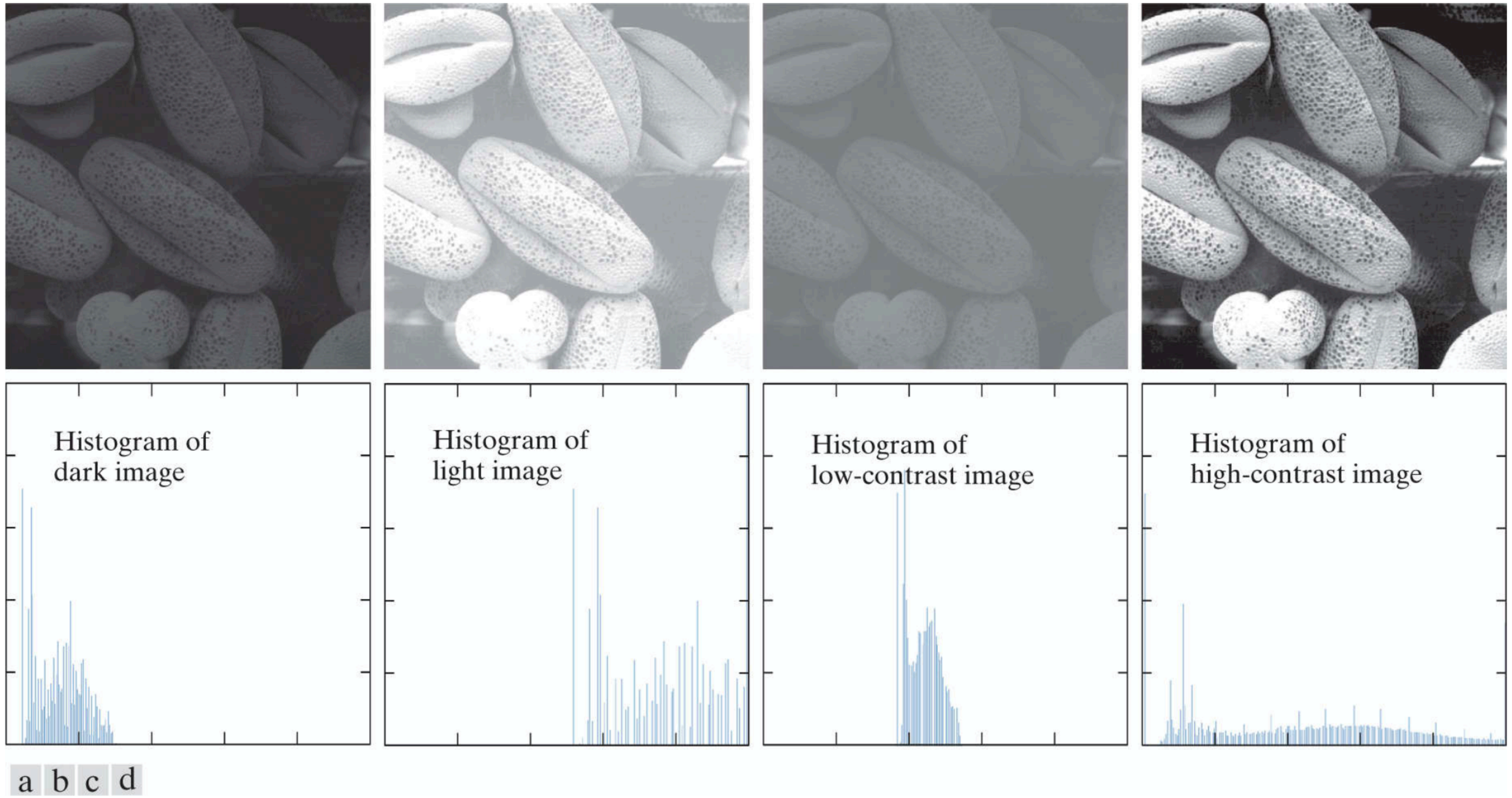
$$p(r_k) = n_k / MN, \; for \; k = 0, 1, 2, \ldots, L-1$$

M and N are the row and column dimensions of the image. And we have

$$\sum_{k=0}^{L-1} p_k(r_k) = 1$$

Point processing

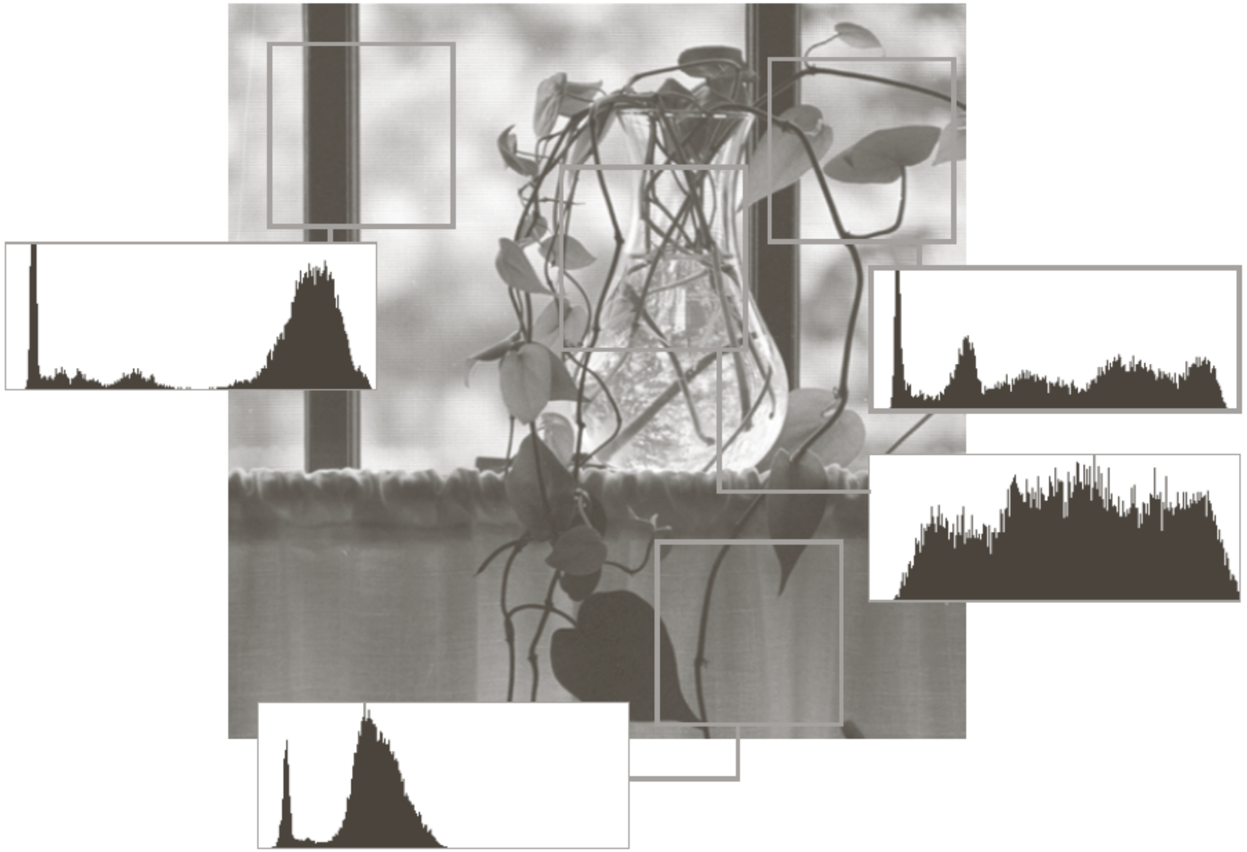
- Histogram Processing



Four image types and their corresponding histograms. (a) dark; (b) light; (c) low contrast; (d) high contrast. The horizontal axis of the histograms are values of and the vertical axis are values of $p(r_k)$.

Point processing

- Histogram Processing



Point processing

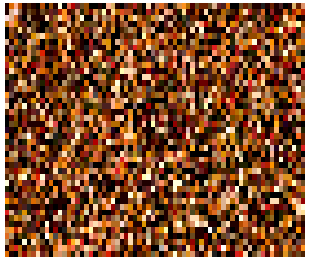
- In many cases histograms are needed for local areas in an image
- Examples:
  - Pattern detection
  - adaptive enhancement
  - adaptive thresholding
  - tracking

Histogram Usage

- Digitizing parameters
- Measuring image properties:
  - Average
  - Variance
  - Entropy
  - Contrast
  - Area (for a given gray-level range)
- Threshold selection
- Image distance
- Image Enhancement
  - Histogram equalization
  - Histogram stretching
  - Histogram matching

Point processing

- Histogram Processing
  - Histograms haven't any position information. Reshuffling all pixels within the image



Its histogram won't change. No point processing will be affected…

Spatial information is discarded

Point processing

- Histogram Processing
  - Histogram equalization
    - The histogram of is a uniform histogram
  - Histogram specification
    - The histogram of has a specified shape
  - Histogram equalization is a special example of histogram specification
  - Theoretical argument of histogram equalization
    - Principle of the biggest entropy

$$H_c = -\int_{r_{min}}^{r_{max}} p(r) \log p(r)\, dr$$

    - While the histogram is equalized, the entropy of the image is biggest, which means that the human visual system can obtain the maximum information

Histogram Equalization

Transformation function

$$s = T(r) \quad 0 \le r \le L-1$$

Assume that:

(a) T(r) is a monotonically increasing function in the interval

$$0 \le r \le L-1$$

(b)

$$0 \le T(r) \le L-1 \quad for \quad 0 \le r \le L-1$$

In some formulations to be discussed later, we use the inverse

$$r = T^{-1}(s) \quad 0 \le s \le L-1$$

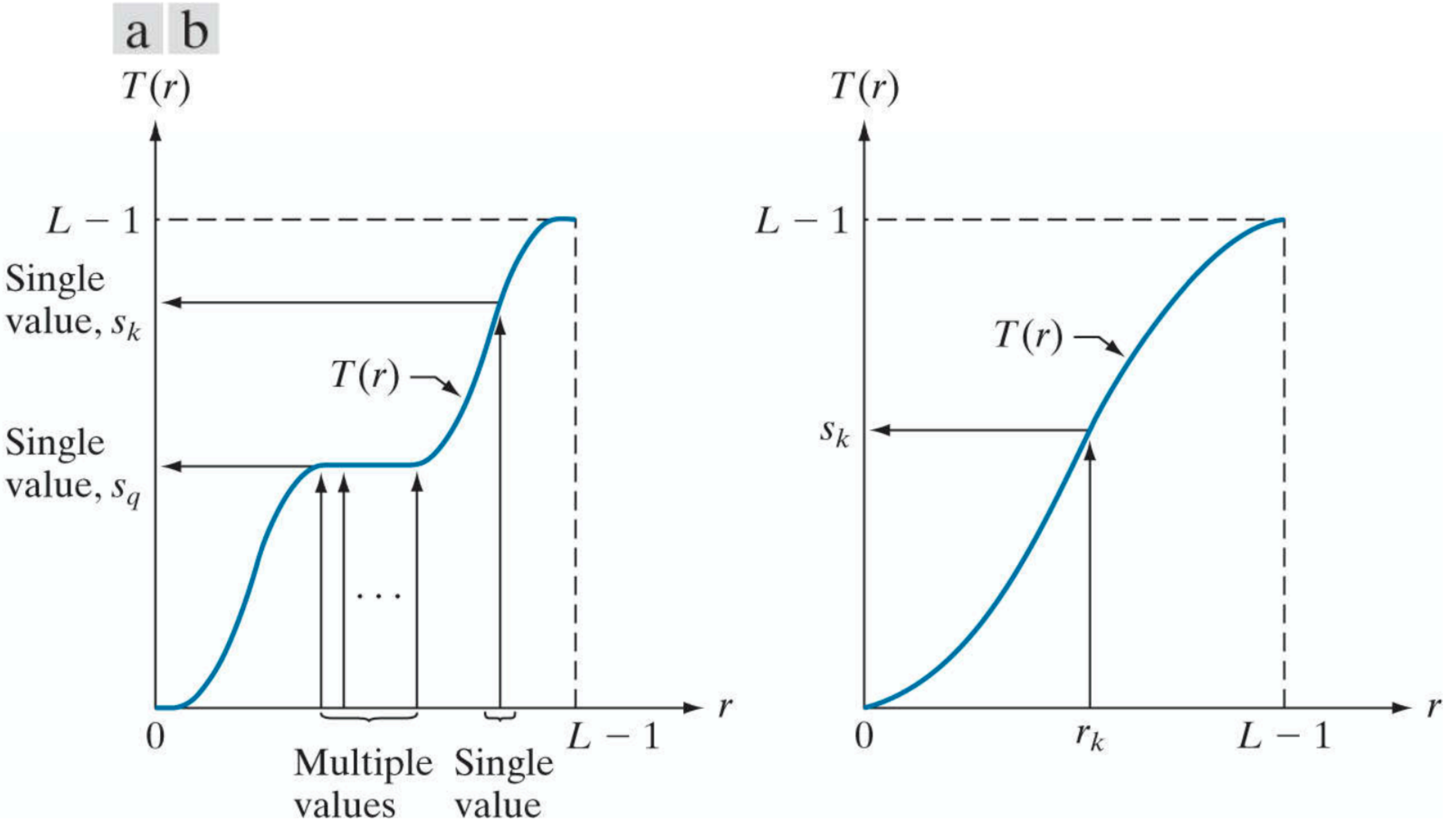
change condition (a) to

(a') T(r) is a strictly monotonically increasing function in the interval

$$0 \le r \le L-1$$

Histogram Equalization

(a) Monotonic increasing function, showing how multiple values can map to a single value. (b) Strictly monotonic increasing function. This is a one-to-one mapping, both ways.



Histogram Equalization

- $p_r(r)$ and $p_s(s)$ denote the PDFs of r and s, respectively. $p_r(r)$ and T(r) are known and T(r) is continuous and differentiable. Then the PDF of the transformed variable s can be obtained using the simple formula

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

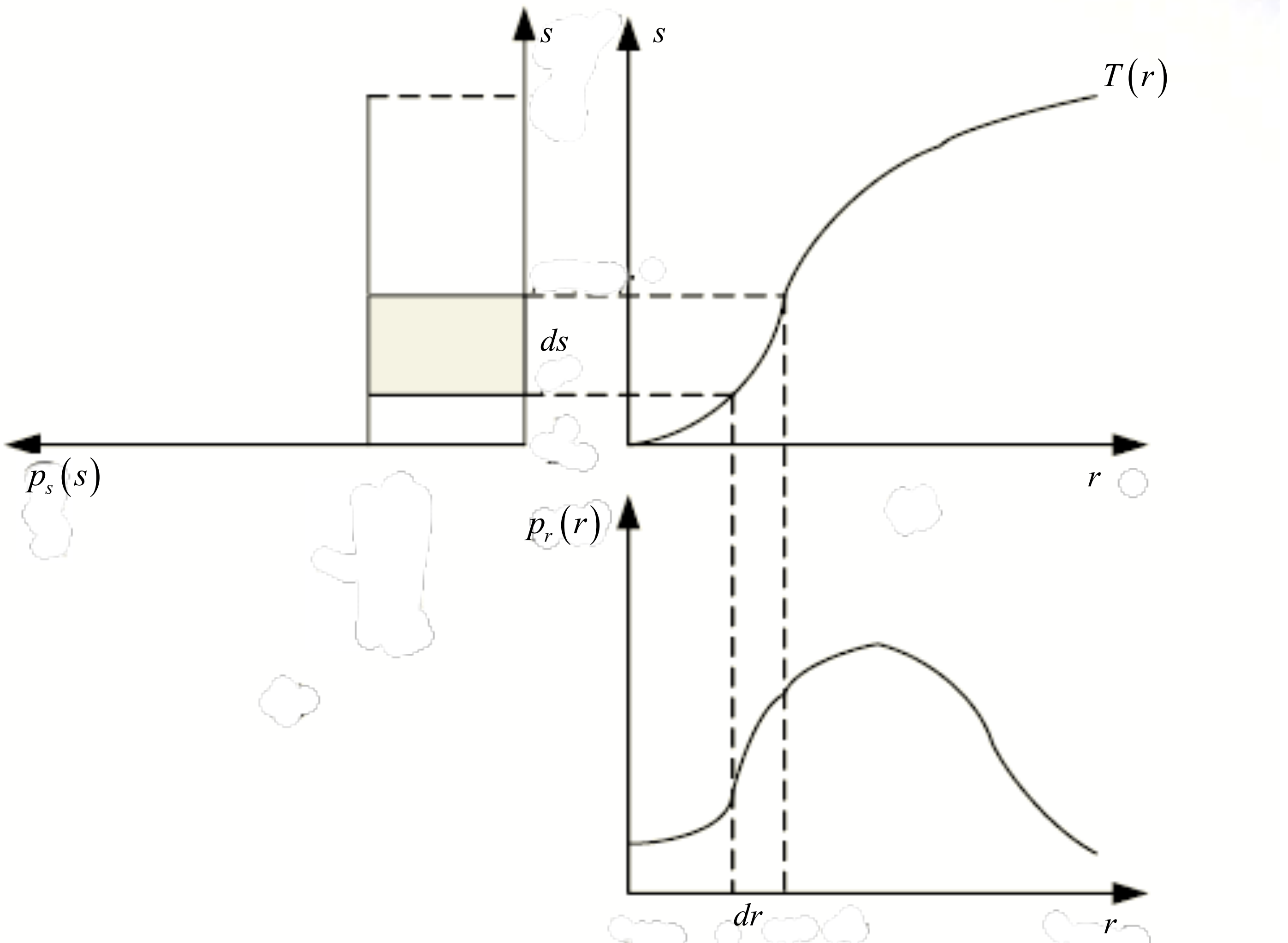
- A transformation function of particular importance in image processing has the form

$$s = T(r) = (L-1)\int_0^r p_r(w)\,dw$$

Histogram Equalization

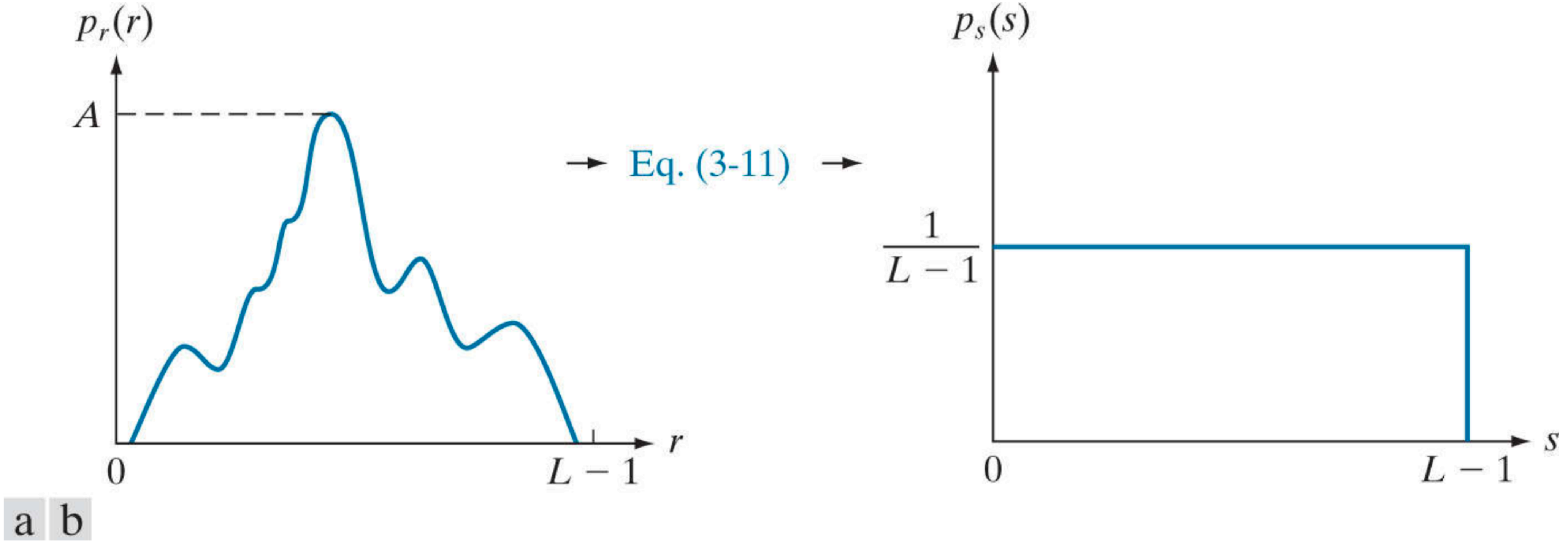
$$\frac{ds}{dr} = \frac{dT(r)}{dr}$$
$$= (L-1)\frac{d}{dr}\left[\int_0^r p_r(w)\,dw\right]$$
$$= (L-1)\,p_r(r)$$

$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right|$$
$$= p_r(r)\left|\frac{1}{(L-1)\,p_r(r)}\right|$$
$$= \frac{1}{L-1} \qquad 0 \le s \le L-1$$



Histogram Equalization

$$s = T(r) = (L-1)\int_0^r p_r(w)\,dw$$



(a) An arbitrary PDF. (b) Result of applying equation upper-right to the input PDF. The resulting PDF is always uniform, independently of the shape of the input.

Histogram Equalization

- For discrete values, we deal with probabilities (histogram values) and summations instead of probability density functions and integrals. The condition s of monotonicity stated earlier apply also in the discrete case

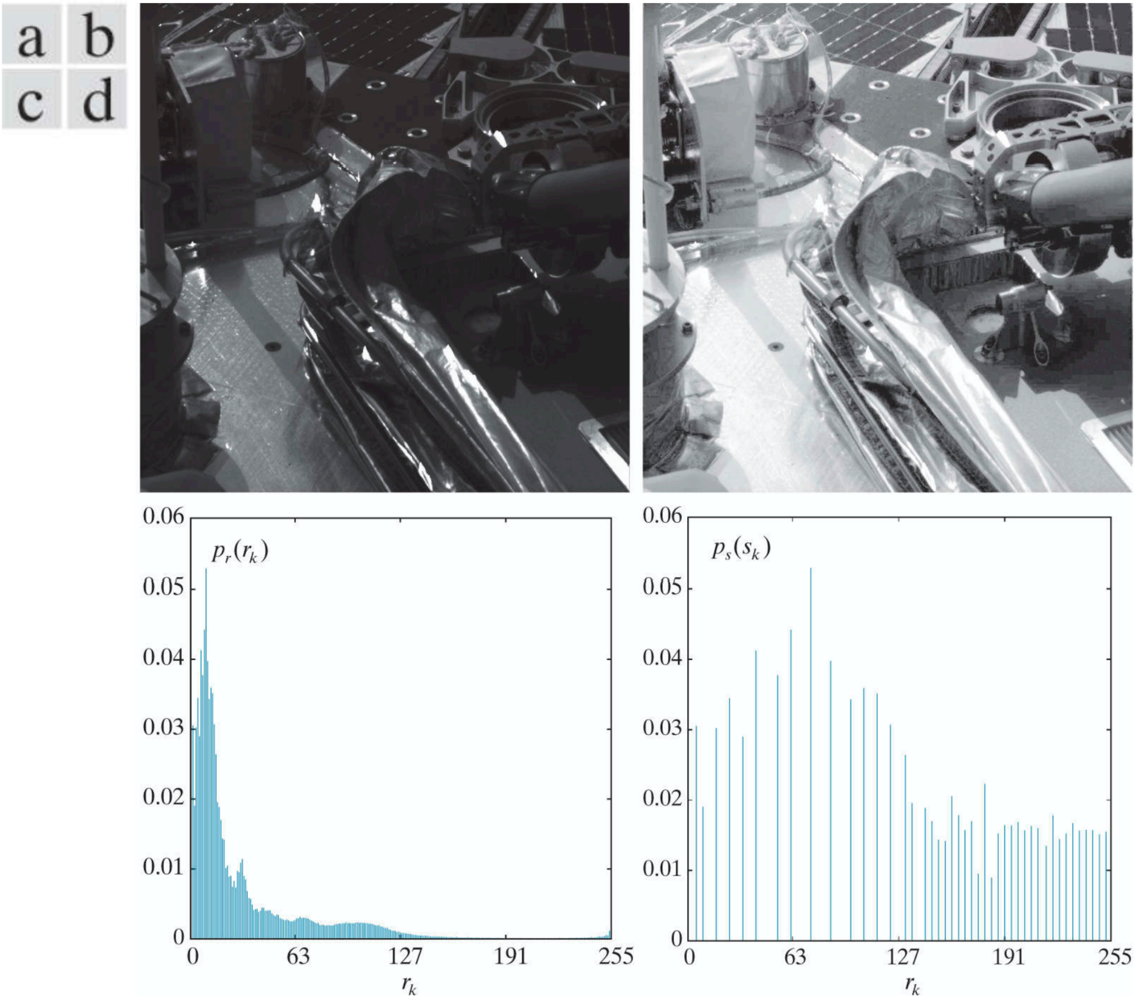
$$p_r(r_k) = \frac{n_k}{MN} \quad k = 0,1,2,\ldots,L-1$$

- A plot of $p_r(r_k)$ versus r_k is commonly referred to as a histogram
- The discrete form of the transformation is

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

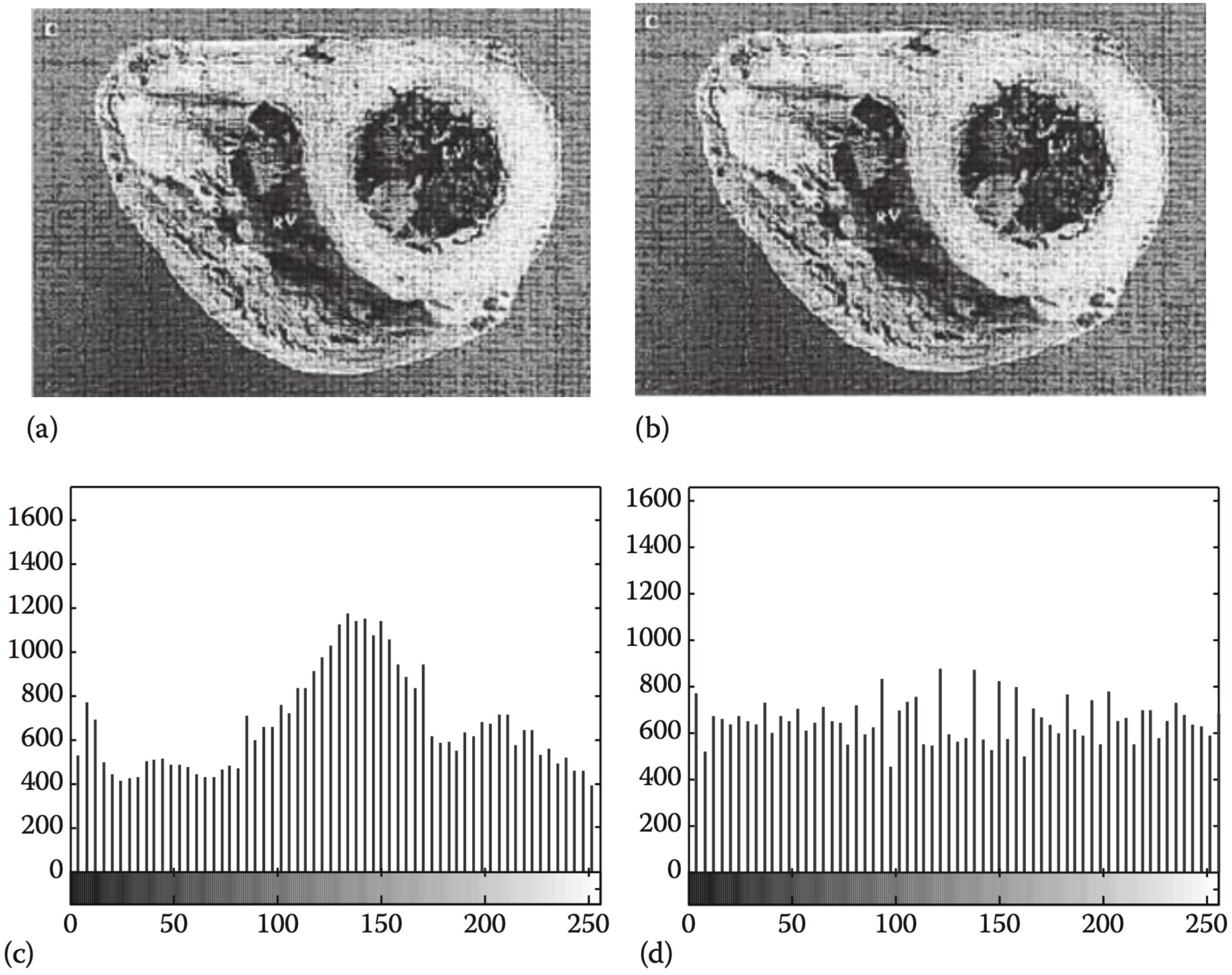
$$= \frac{L-1}{MN}\sum_{j=0}^{k} n_j \qquad k = 0,1,2,\ldots,L-1$$

Histogram Equalization



(a) Image from Phoenix Lander. (b) Result of histogram equalization. (c) Histogram of image (a). (d) Histogram of image (b). (Original image courtesy of NASA.)

Histogram Equalization



(a) Original image (b) Transformed image after histogram equalization (c) Gray-level histogram of the original image (d) Histogram of the image after histogram equalization (Courtesy of Andre D'Avila, MD, Heart Institute (InCor), University of Sao Paulo, Medical School, Sao Paulo, Brazil)

Spatial Filtering

Mask Processing: Learning Filtering In Space Domain

It is often the case that instead of linear processing of images using filters described in frequency domain, space-domain linear filters are used in typical image processing applications. This is mainly to the fact that frequency-domain description of two-dimensional (2-D) filters is often more complex than the one-dimensional (1-D) filters. In principle, space-domain linear filters approximate the impulse response of various kinds of typical frequency-domain filters with a 2-D mask In spatial filtering, as described before, a weight mask is used to express the effect of the filter on each pixel of the image in an insightful fashion.

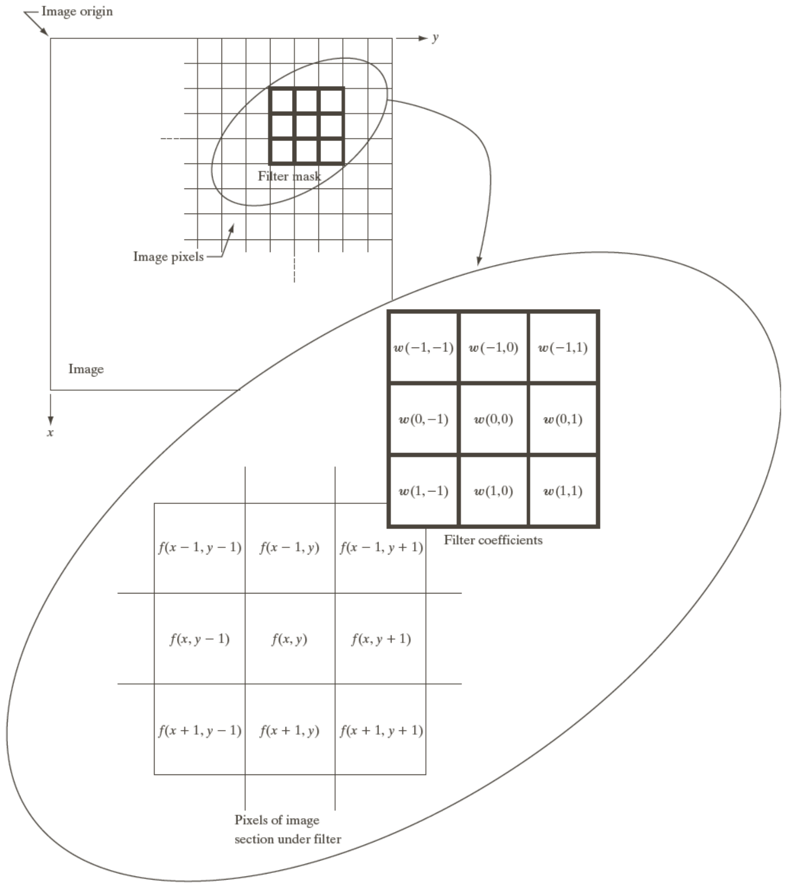
$\omega(-1,-1)$	$\omega(-1,0)$	$\omega(-1,1)$
$\omega(0,-1)$	$\omega(0,0)$	$\omega(0,1)$
$\omega(1,-1)$	$\omega(1,0)$	$\omega(1,1)$

Typical 3×3 mask.

Spatial Filtering

- The mechanics of spatial filtering
  - For a mask of size 3 by 3

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



Spatial Filtering

- Low-Pass Filters

	1	1	1
(1/9)*	1	1	1
	1	1	1

(a)

	1	2	1
(1/16)*	2	4	2
	1	2	1

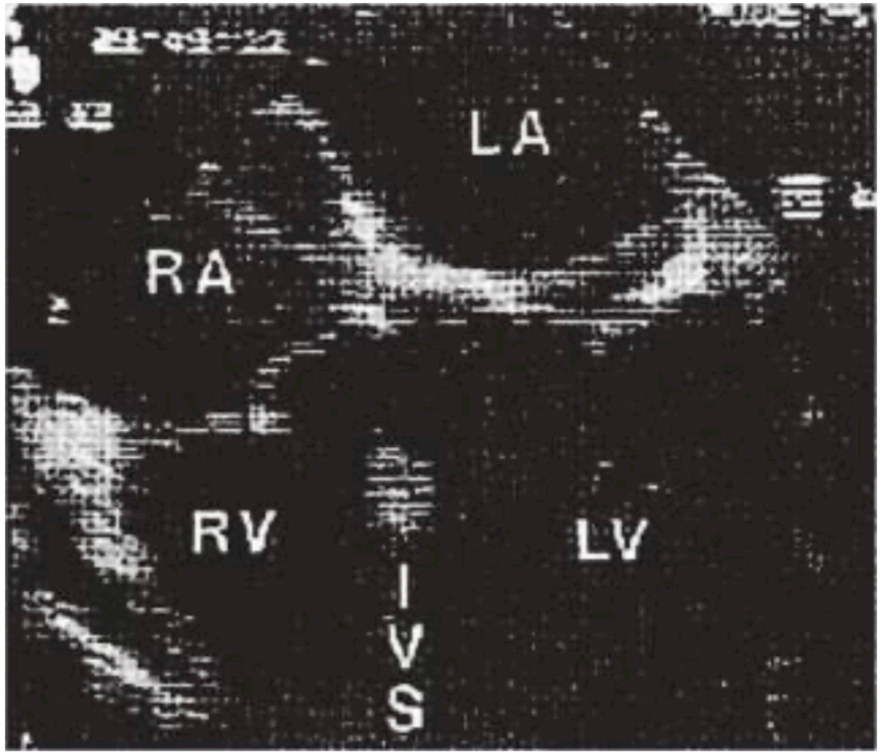
(b)

(a and b) Two typical masks used for low-pass filtering

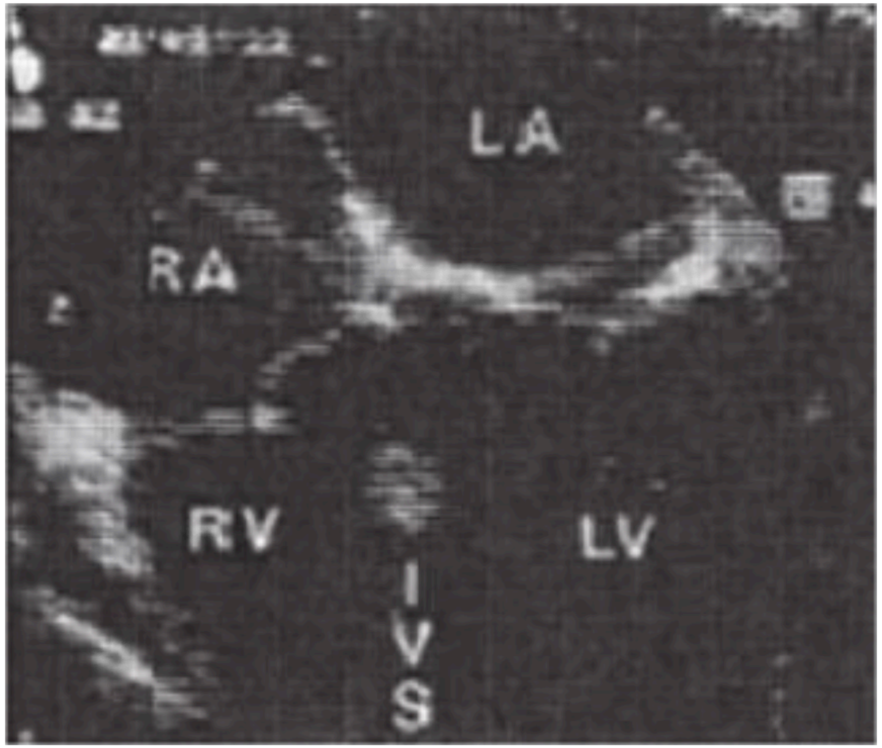
Low-pass filters attenuate or eliminate high-frequency components of an image such as edges, texture, and other sharp details. Low-pass filters that are often used for applications such as smoothing, blurring, and noise reduction provide a smooth version of the original image. A low-pass filter is sometimes used as a preprocessing step to remove unimportant details from an image before object extraction.

Spatial Filtering

- Low-Pass Filters



(a)



(b)

(a) Original image and (b) image after applying low-pass filter

Spatial Filtering

- ## Median Filters

Median filters are statistical nonlinear filters that are often described in the space domain Median filters are known to reduce the noise without eliminating the edges and other high-frequency contents Median filters (also referred to as order statistics filters) perform the following operations to find each pixel value in the processed image:

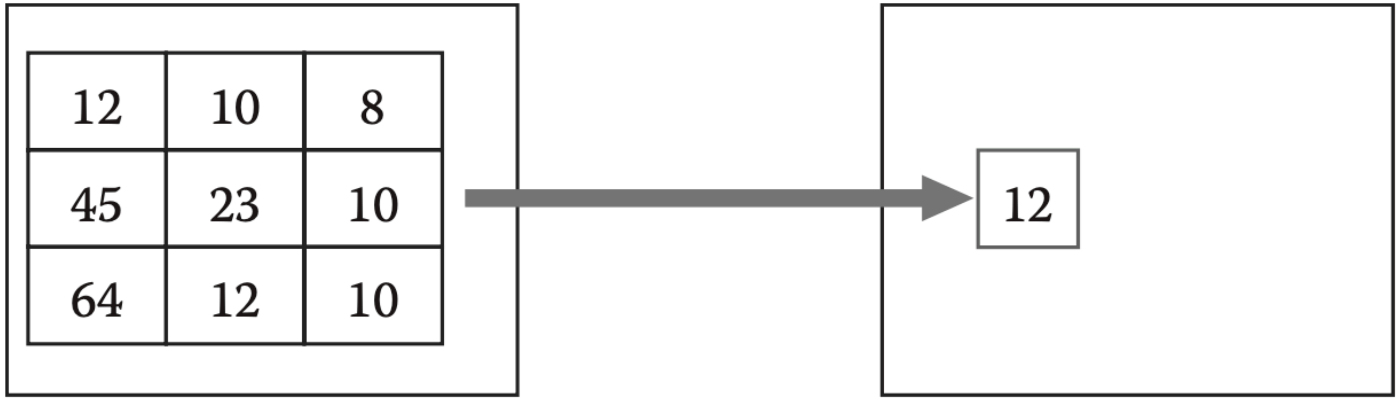
Step 1: All pixels in the neighborhood of the pixel in the original image (identified by the mask) are inserted in a list.
Step 2: This list is sorted in ascending (or descending) order.
Step 3: The median of the sorted list (i.e., the pixel in the middle of the list) is chosen as the pixel value for the processed image.

As defined earlier, median filter create pixel values of the filtered image based on the sort- ing of the gray level of pixels in the mask around the central pixels in the original image

Spatial Filtering

- Median Filters

The performance of a 3×3 median filter on a sub image is illustrated in Figure 3.12. As can be seen from Figure 3.12, the median filter selects the median of the gray- level values in the 3×3 neighborhood of the central pixel and assigns this value as the output. In this example, the median filter is to select the median of following set: {8 10 10 12 12 23 45 64}. According to the sorted list, the response of the filter is 12



Mask of median filter

Spatial Filtering

- Median Filters

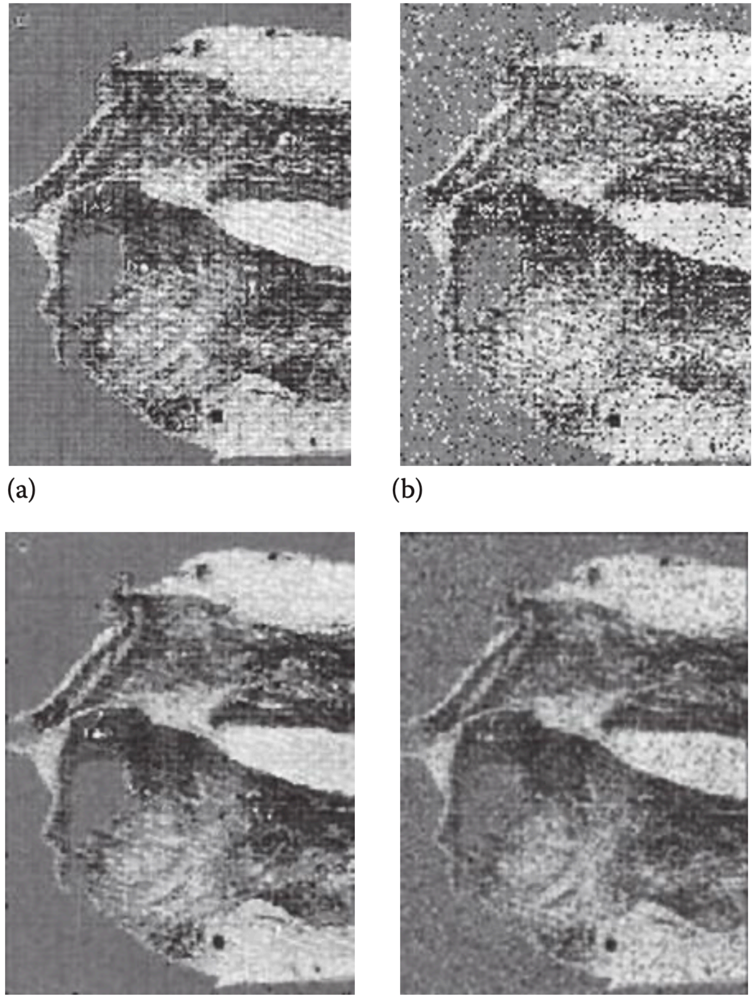
(a)

(b)

(a) Noisy image and (b) median-filtered image.

Median filter has reduced the noise in the image without destroying the edges This is the main advantage of the median filters over the linear low-pass filters

Spatial Filtering



(a) (b) (c) (d)

(a) Original image, (b) noisy image, (c) image after low-pass filter, and (d) image after median filter.

Evidently, the low-pass filter has reduced the noise level in the image; however, at the same time, the filter has blurred the image. On the other hand, while median filter has also reduced the noise, it has preserved the edges of the image almost entirely. Again, this difference is due to the fact that the median filter forces the pixels with distinct intensities to be more like their neighbors and therefore eliminates isolated intensity spikes Such a smoothing criterion will not result in significant amount of filtering across edges.

Median filters, however, have certain disadvantages When the number of noisy pixels is greater than half of the total pixels, median filters give a poor performance This is because, in such cases, median value will be much more influenced by dominating noisy values than the non-noisy pixels In addition, when the additive noise is Gaussian in nature, median filters may fail to provide a desirable filtering performance

Spatial Filtering

- Sharpening spatial filters
  - High-pass filters

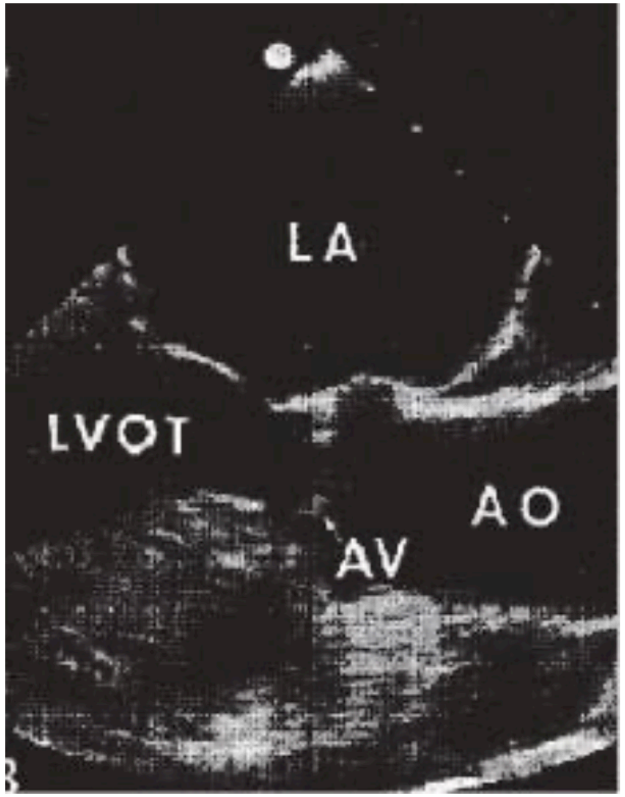
As in linear low-pass filters, the masks used for high-pass filtering are nothing but the truncated approximations of the space-domain representation of the typical ideal high-pass filters As such, in high-pass filters the shape of impulse response should have (+) coefficients near its center and (–) coefficients in the outer periphery

	−1	−1	−1
(1/9)*	−1	8	−1
	−1	−1	−1

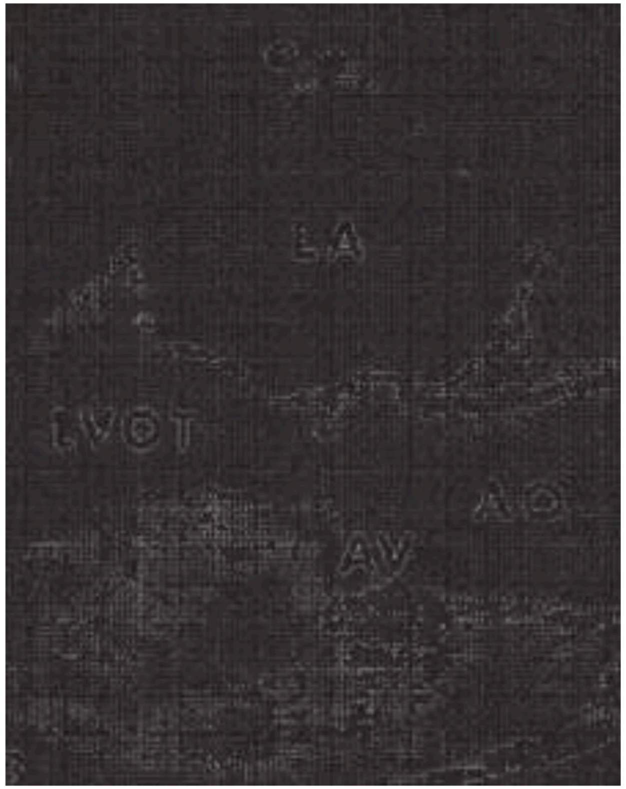
Typical mask of linear high-pass spatial filter

Spatial Filtering

- Sharpening spatial filters



(a)



(b)

(a) Original image and (b) image after sharpening spatial filter

Spatial Filtering

- Sharpening spatial filters
  - High-boost filters

	0	0	0
$(A-1)$	0	1	0
	0	0	0

	−1	−1	−1
+1/9	−1	8	−1
	−1	−1	−1

	−1	−1	−1
=1/9	−1	W	−1
	−1	−1	−1

High-boost mask

Some extensions of high-pass filters, while highlighting the high frequencies, preserve some low-frequency components and avoid negative pixel values The most commonly used extensions of high-pass filters are high-boost filters that are also referred to as high-frequency emphasis filters

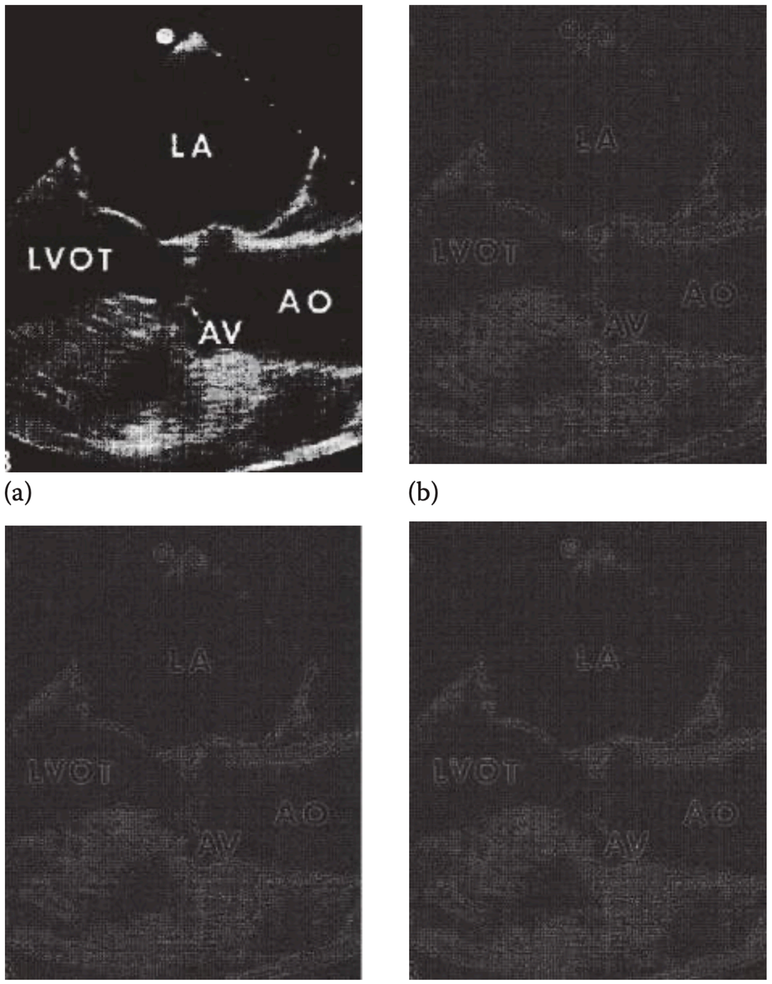
Spatial Filtering

- Sharpening spatial filters
  - High-boost filters

High-boost filtered image = (A −1) original image + high-pass filtered image.
Or simply, in terms of filters:

$$\text{High-boost filter} = (A-1)\ \text{all-pass filter} + \text{high-pass filter}$$

Spatial Filtering



(a) (b) (c) (d)

(a) Original image, (b) filtered image with A = 1 1, (c) filtered image with A = 1 15, and (d) filtered image with A = 1 2

Spatial Filtering

- Sharpening spatial filters
  - Derivative filters

As we saw in the previous sections, image blurring can be caused by averaging Since averaging is simply the discrete version of spatial integration, one can expect that spatial differentiation would result in image sharpening This observation about spatial differentiation is the main idea behind a family of sharpening filters called "derivative filters " In order to find suitable masks for spatial differentiation, we need to study the concept of differentiation in digital 2-D spaces more closely

Derivative Filters

Since an image is a 2-D signal, instead of simple 1-D differentiation, the directional differentiations must be calculated in both horizontal and vertical directions. This leads to spatial gradient defined as follows:

$$\nabla f = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix} \tag{1}$$

Derivative Filters

The partial differentiations such as ∂f/∂x can be approximated in discrete image simply by calculating the difference in the gray level of two neighboring pixels, i.e.,

$$\begin{aligned}\frac{\partial f(x,y)}{\partial x} &\cong \frac{f(x,y)-f(x-1,y)}{x-(x-1)} \\ &= \frac{f(x,y)-f(x-1,y)}{1} \\ &= f(x,y)-f(x-1,y)\end{aligned} \tag{2}$$

Derivative Filters

Note that since the smallest value to approximate ∂x is one pixel, we ended up replacing this value with 1. Also note that the final value in (1) is an integer (positive or negative). This is due to the fact that subtraction of two integers (i.e., f(x, y) − f(x − 1, y)) would always give an integer value Back to the continuous gradient, the magnitude of the gradient vector is given by

$$|\nabla f| = \left[\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2\right]^{1/2} \qquad (3)$$

Derivative Filters

Now, note that storing integers in digital computers is significantly more efficient than storing real numbers that require floating points In addition, performing calculations with integers are faster and more efficient than doing calculations with real numbers These two observations strongly encourage the use of integers for image processing in which large images must be stored and processed Since the result of (2) is almost always a real number, we need to approximate this operation such that the resulting number stays an integer.
The approximation of (2) typically used in image processing is as follows:

$$|\nabla f| \cong \left|\frac{\partial f}{\partial x}\right| + \left|\frac{\partial f}{\partial y}\right| \qquad (4)$$

This approximation does not only give us a positive integer, but it also reduces the time complexity of calculating the magnitude of the gradient vector

Frequency Filtering

Frequency Filtering

- Why filtering in the frequency domain

  - The filtering methods in different domains can correspond to each other, but some operations will be better implemented in a certain domain.

Frequency Filtering

- Smooth filters in frequency domain
  - The main objective in smoothing an image is to decrease the noisy fast variations in the gray levels of the image Since the fast variations in gray level of digital images correspond to high frequencies in DFT of the image, a filter that attenuates the high- frequency values of the DFT of the original image is simply a low-pass filter

Frequency Filtering

- Smooth filters in frequency domain
  - Idea low-pass filter

$$H(u,v) = \begin{cases} 1 & D(u,v) \le D_0 \\ 0 & D(u,v) > D_0 \end{cases}$$

where

$$D(u,v) = (u^2 + v^2)^{1/2}$$

Frequency Filtering

- Smooth filters in frequency domain
  - Butterworth low-pass filters

$$H(u,v) = \frac{1}{1+\left[D(u,v)/D_0\right]^2}$$

where D(u, v) is the distance from the origin in the frequency domain, as previously defined.

Frequency Filtering

- Sharpening filters in frequency domain
  - Ideal high-pass filters

$$H(u,v) = \begin{cases} 0 & D(u,v) \le D_0 \\ 1 & D(u,v) > D_0 \end{cases}$$

Frequency Filtering

- Sharpening filters in frequency domain
  - Butterworth high-pass filters

$$H(u,v) = \frac{1}{1 + \left[D_0 / D(u,v) \right]^2}$$

where D(u, v) is the distance from the origin in the frequency domain

Thank You!