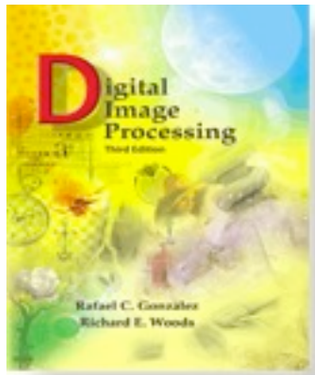


Chapter 3

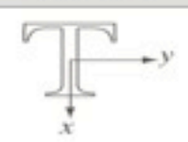
Intensity Transformations & Spatial Filtering

Last time:

Affine transforms (linear spatial transforms)

$$[\ x \quad y \quad 1\ ] = [\ v \quad w \quad 1\ ] \begin{bmatrix} t_{11} & t_{12} & 0 \\ t_{21} & t_{22} & 0 \\ t_{31} & t_{32} & 1 \end{bmatrix}$$

TABLE 2.2
Affine transformations based on Eq. (2.6.–23).

Transformation Name	Affine Matrix, T	Coordinate Equations	Example
Identity	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v$ $y = w$	
Scaling	$\begin{bmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = c_x v$ $y = c_y w$	
Rotation	$\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v\cos\theta - w\sin\theta$ $y = v\cos\theta + w\sin\theta$	
Translation	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ t_x & t_y & 1 \end{bmatrix}$	$x = v + t_x$ $y = w + t_y$	
Shear (vertical)	$\begin{bmatrix} 1 & 0 & 0 \\ s_v & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v + s_v w$ $y = w$	
Shear (horizontal)	$\begin{bmatrix} 1 & s_h & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x = v$ $y = s_h v + w$	

Chapter 3

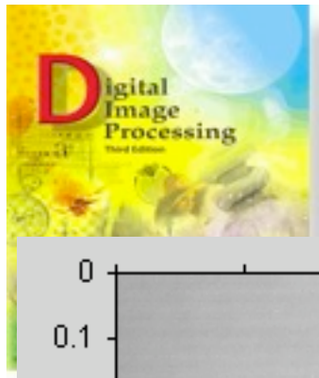
Intensity Transformations & Spatial Filtering

```
IMTRANSFORM Apply 2-D spatial transformation to image.
    B = IMTRANSFORM(A,TFORM) transforms the image A according to the 2-D
    spatial transformation defined by TFORM, which is a tform structure
    as returned by MAKETFORM or CP2TFORM.  If ndims(A) > 2, such as for
    an RGB image, then the same 2-D transformation is automatically
    applied to all 2-D planes along the higher dimensions.
```

$$\begin{bmatrix} x & y & 1 \end{bmatrix} = \begin{bmatrix} v & w & 1 \end{bmatrix} \begin{bmatrix} t_{11} & t_{12} & 0 \\ t_{21} & t_{22} & 0 \\ t_{31} & t_{32} & 1 \end{bmatrix}$$

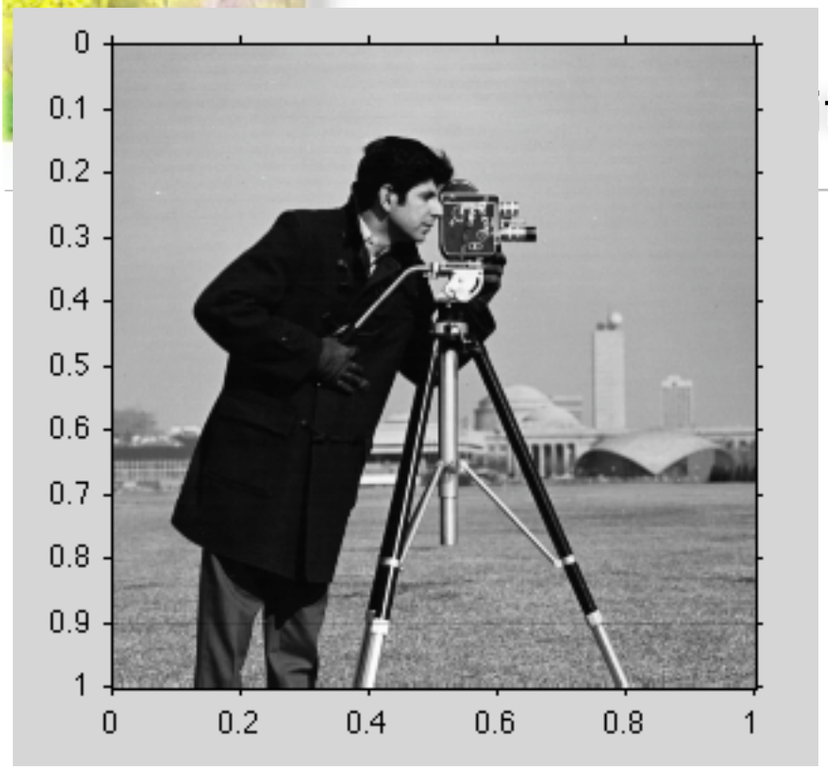
$$\begin{aligned} x &= vt_{11} + wt_{21} + t_{31} \\ y &= vt_{12} + wt_{22} + t_{32} \end{aligned}$$

```
T =

     2     0     0
     1     1     0
    50    -1     1

>> tform = maketform('affine',T);

tform =

       ndims_in: 2
      ndims_out: 2
    forward_fcn: @fwd_affine
    inverse_fcn: @inv_affine
          tdata: [1x1 struct]

>> IMG2= imtransform(IMG, tform)
```



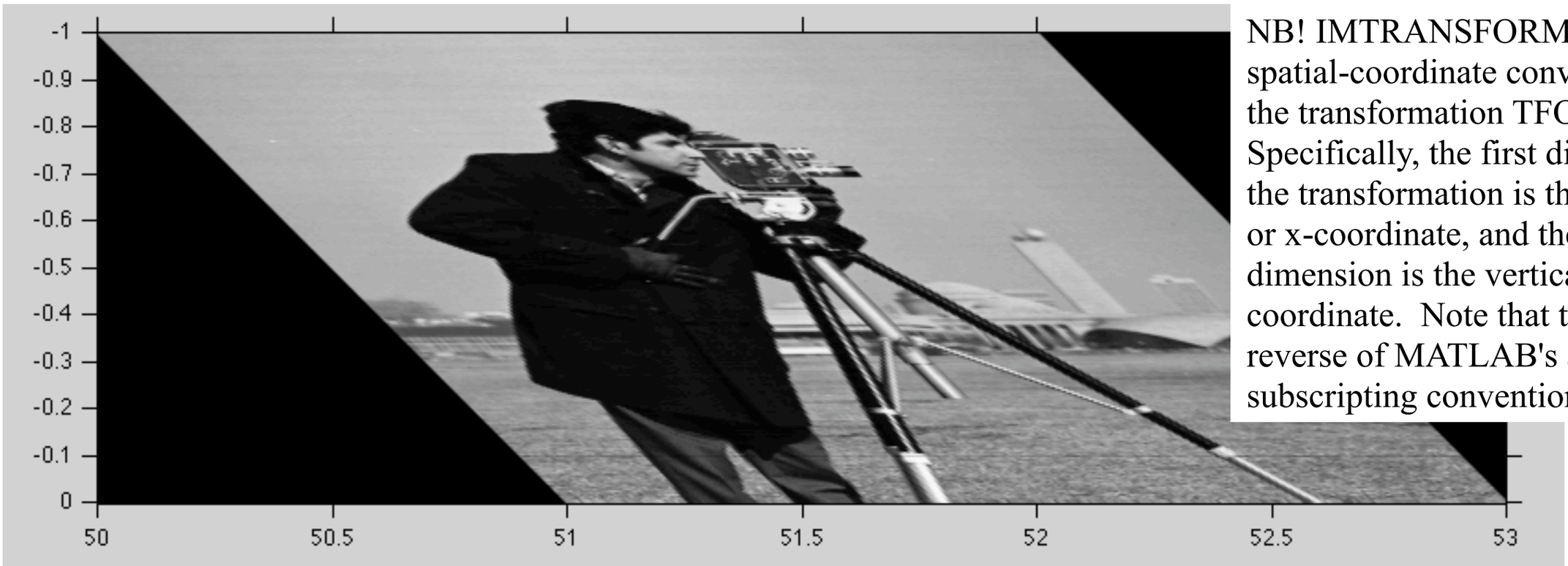
Chapter 3

ty Transformations & Spatial F

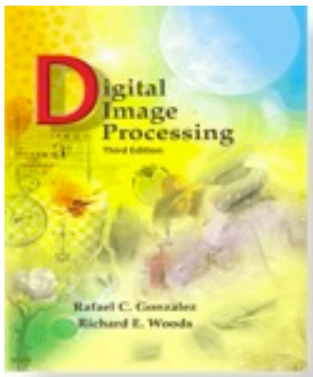


$$x = vt_{11} + wt_{21} + t_{31}$$
$$y = vt_{12} + wt_{22} + t_{32}$$

```
T =

     2     0     0
     1     1     0
    50    -1     1

>> tform = maketform('affine',T);

tform =

        ndims_in: 2
       ndims_out: 2
     forward_fcn: @fwd_affine
     inverse_fcn: @inv_affine
           tdata: [1x1 struct]

>> IMG2= imtransform(IMG, tform)
```



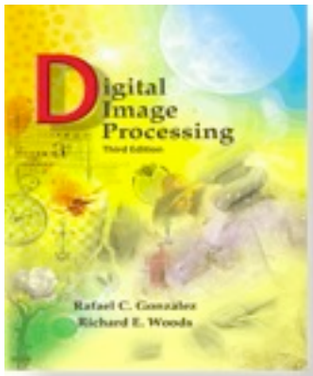
NB! IMTRANSFORM assumes spatial-coordinate conventions for the transformation TFORM. Specifically, the first dimension of the transformation is the horizontal or x-coordinate, and the second dimension is the vertical or y-coordinate. Note that this is the reverse of MATLAB's array subscripting convention.



Chapter 3

Intensity Transformations & Spatial Filtering

The process of applying a geometric transformation to an image (often called study) to match it to another image (called template/reference) is called *image registration*.

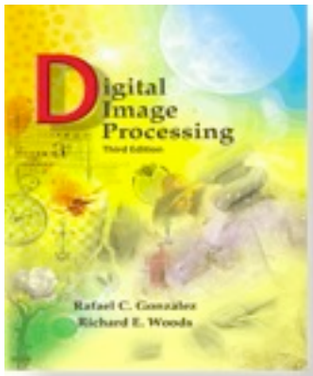


Chapter 3

Intensity Transformations & Spatial Filtering

The process of applying a geometric transformation to an image (often called study) to match it to another image (called template/reference) is called *image registration*.

Affine/rigid transformations can be used to perform image registration.



Chapter 3

Intensity Transformations & Spatial Filtering

The process of applying a geometric transformation to an image (often called study) to match it to another image (called template/reference) is called *image registration*.

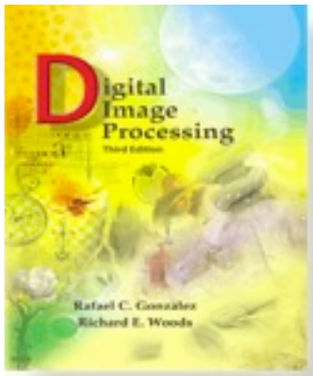
Affine/rigid transformations can be used to perform image registration.

There exist also other types of registration (nonlinear)

-bilinear, bicubic, polynomials

-spline representations

-fully nonlinear (by PDEs)



Chapter 3

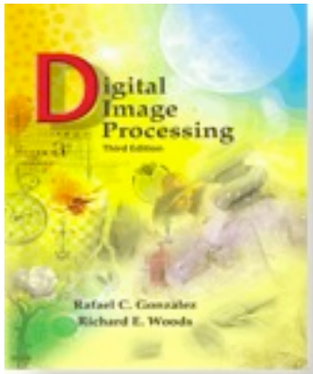
Intensity Transformations & Spatial Filtering

The process of applying a geometric transformation to an image (often called study) to match it to another image (called template/reference) is called *image registration*.

Affine/rigid transformations can be used to perform image registration.

There exist also other types of registration (nonlinear)
- -bilinear, bicubic, polynomials
- -spline representations
- -fully nonlinear (by PDEs)

For the time being, we will focus only on linear and bilinear transformations, other registration methods will be considered at the end of the course (MAT262 only).



Chapter 3

Intensity Transformations & Spatial Filtering

Image registration

Linear (affine):

we have 6 free parameter
we need 6 conditions to determine the parameters

Bilinear:

$$\begin{aligned} x &= c_1 v + c_2 w + c_3 vw + c_4 \\ y &= c_5 v + c_6 w + c_7 vw + c_8 \end{aligned}$$

$$(av + b)(cw + d)$$

we have 8 free parameter
we need 8 conditions to determine the parameters

A typical method to set up the parameters is through *landmarks* also called (*tie, control*, ... points).
This can be done either manually/automatically.

Some imaging systems include some sensor/markers to help recognizing the points.

It suffices:

Linear: 6 free parameters → 3 couples of points

Bilinear: 8 free parameters → 4 couples of points

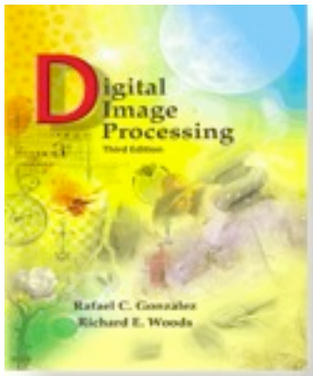
Each couple of coordinate *(x,y)* gives 2 conditions.

Solve the two linear systems

$$\begin{bmatrix} v_1 & w_1 & v_1w_1 & 1 \\ v_2 & w_2 & v_2w_2 & 1 \\ v_3 & w_3 & v_3w_3 & 1 \\ v_4 & w_4 & v_4w_4 & 1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}, \qquad \begin{bmatrix} v_1 & w_1 & v_1w_1 & 1 \\ v_2 & w_2 & v_2w_2 & 1 \\ v_3 & w_3 & v_3w_3 & 1 \\ v_4 & w_4 & v_4w_4 & 1 \end{bmatrix} \begin{bmatrix} c_5 \\ c_6 \\ c_7 \\ c_8 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}$$

(note that they have the same matrix of coefficients)

to find the parameters of the transformation.



Digital Image Processing, 3rd ed.

Gonzalez & Woods

www.ImageProcessingPlace.com

Chapter 3

Intensity Transformations & Spatial Filtering

We could use more points (overdetermined system)
in which case we have a Least Squares linear problem, that can be solved by standard methods (normal equations, QR, SVD, etc.).

We could use more points (overdetermined system)
in which case we have a Least Squares linear problem, that can be solved by standard methods (normal equations, QR, SVD, etc.).

These transformations might also be combined to work on subregions of the images, which then have to be patched together.

Chapter 3: Intensity transformations and spatial filtering

Generic form:

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

new intensity value at *(x,y)*

is a function of the old intensity value at *(x,y)*, or S_{xy}

Chapter 3

Intensity Transformations & Spatial Filtering

Example of spatial filtering: blurring by 3x3 neighborhood

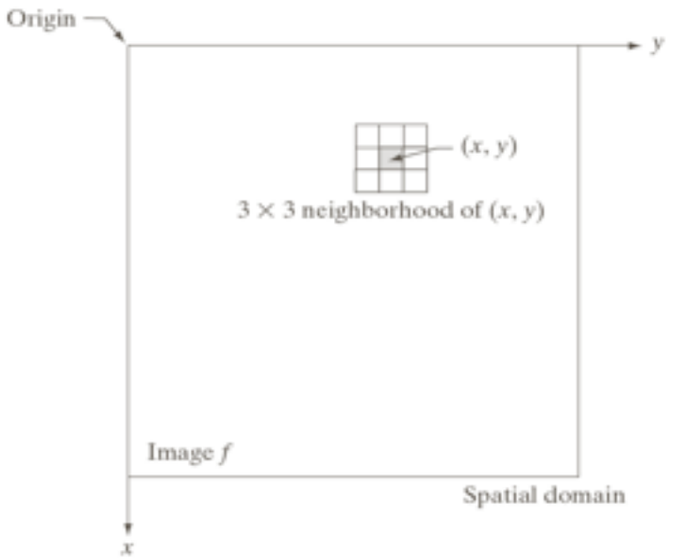
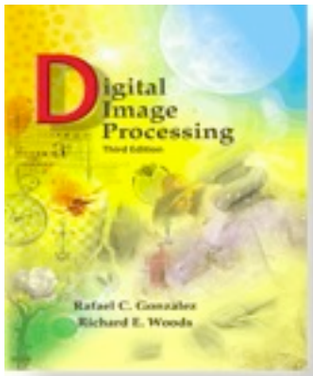


FIGURE 3.1 A 3 × 3 neighborhood about a point (x, y) in an image in the spatial domain. The neighborhood is moved from pixel to pixel in the image to generate an output image.

Create a 3x3 mask, and run it over the whole image to generate the new image (can also use conv2)

Here I have run the mask only on the left half of the picture.



Chapter 3

Intensity Transformations & Spatial Filtering

Example of spatial filtering: blurring by 3x3 neighborhood

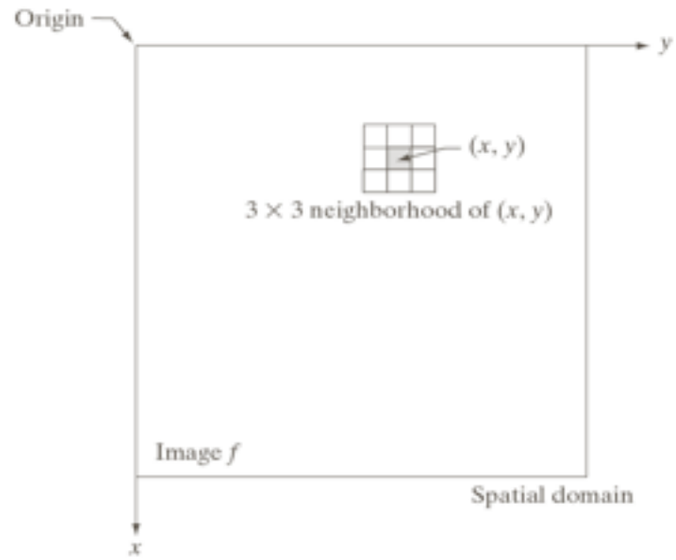
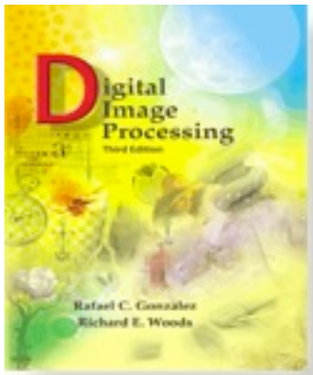


FIGURE 3.1 A 3 × 3 neighborhood about a point (x, y) in an image in the spatial domain. The neighborhood is moved from pixel to pixel in the image to generate an output image.

Create a 3x3 mask, and run it over the whole image to generate the new image (can also use conv2)

Here I have run the mask only on the left half of the picture.

Example of a blurred image by a 3x3 mask



Chapter 3

Intensity Transformations & Spatial Filtering

Simplest example:
1-pixel neighborhood (the pixel itself), in which case we operate directly on the intensity of the pixel.

This is called *point processing* as opposed to *neighborhood processing*.

Typical operations we will perform:

contrast stretching
thresholding

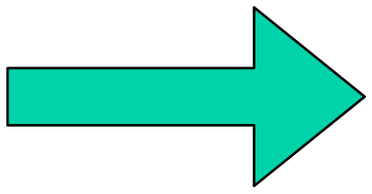
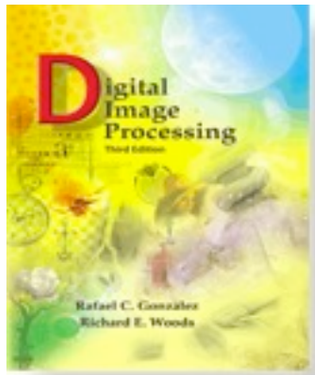


Image enhancement

Is the task of manipulating an image so that the result is more suitable for specific applications.
Mostly visual (subjective), no general quantitative assessment.

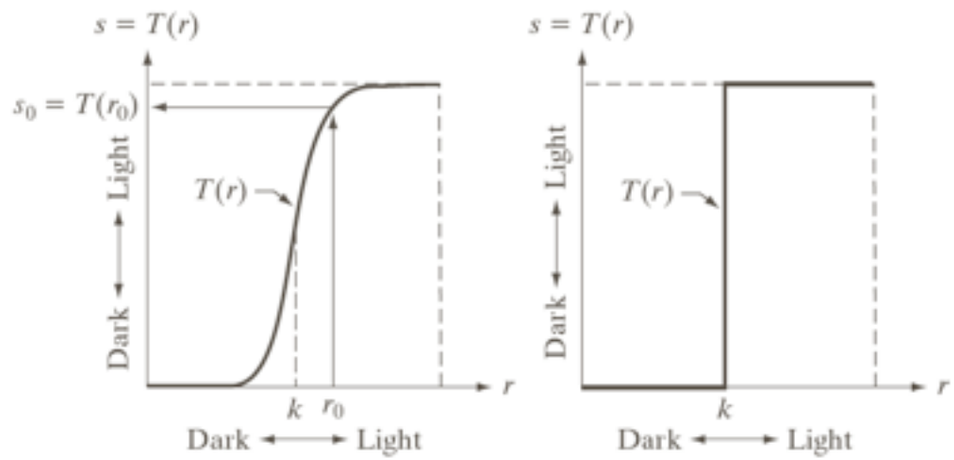


Chapter 3

Intensity Transformations & Spatial Filtering

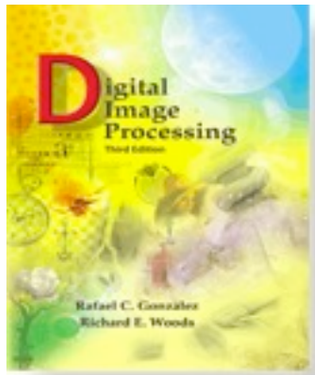
Contrast stretching (sigmoid):

low intensity areas are mapped
to even lower
high intensity areas are mapped
to even higher
medium intensity areas are
"stretched".



a b

FIGURE 3.2 Intensity transformation functions. (a) Contrast-stretching function. (b) Thresholding function.

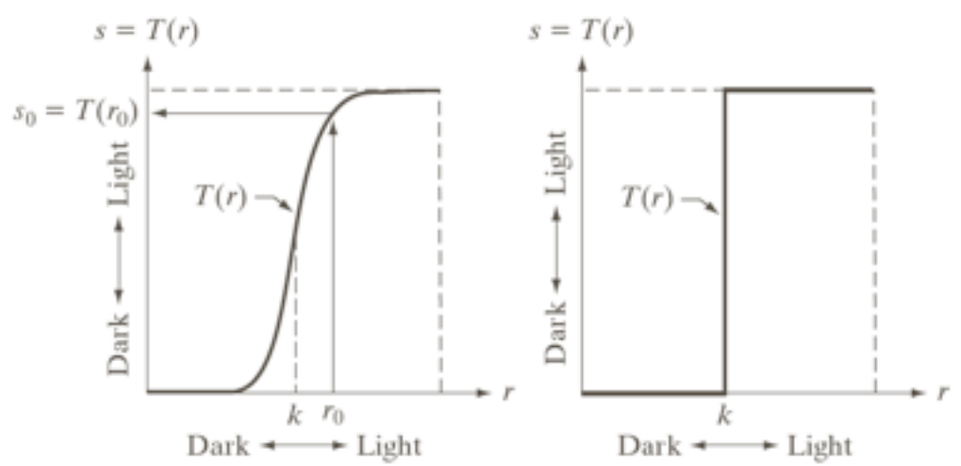


Chapter 3

Intensity Transformations & Spatial Filtering

Contrast stretching (sigmoid):

low intensity areas are mapped
to even lower
high intensity areas are mapped
to even higher
medium intensity areas are
"stretched".

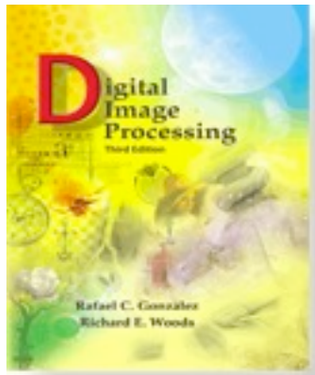


a b

FIGURE 3.2 Intensity transformation functions. (a) Contrast-stretching function. (b) Thresholding function.

Original

contrast stretched

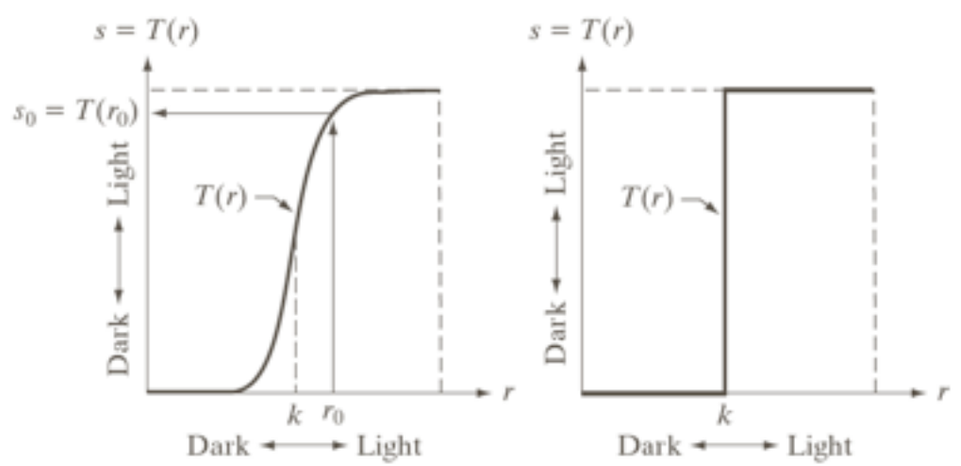


Chapter 3

Intensity Transformations & Spatial Filtering

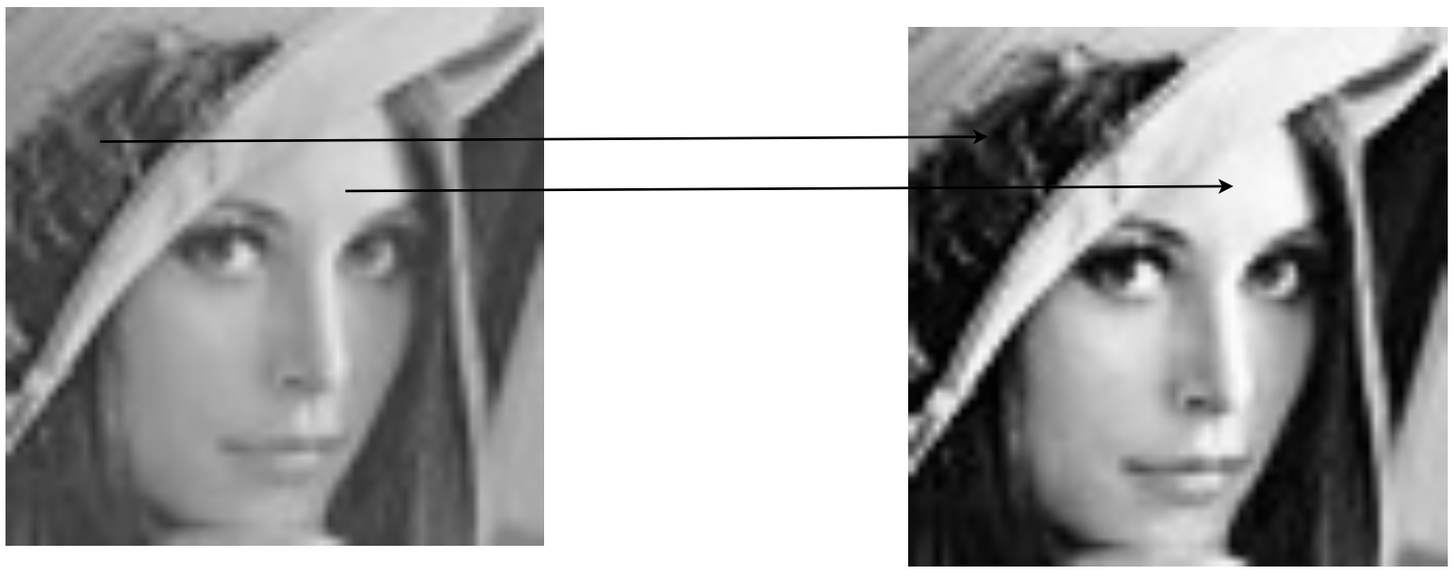
Contrast stretching (sigmoid):

low intensity areas are mapped
to even lower
high intensity areas are mapped
to even higher
medium intensity areas are
"stretched".



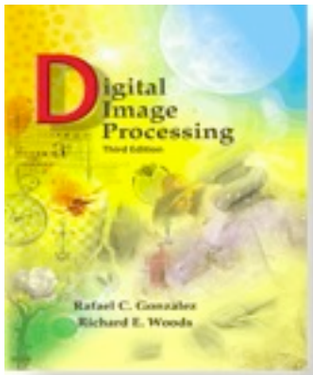
a b

FIGURE 3.2 Intensity transformation functions. (a) Contrast-stretching function. (b) Thresholding function.



Original

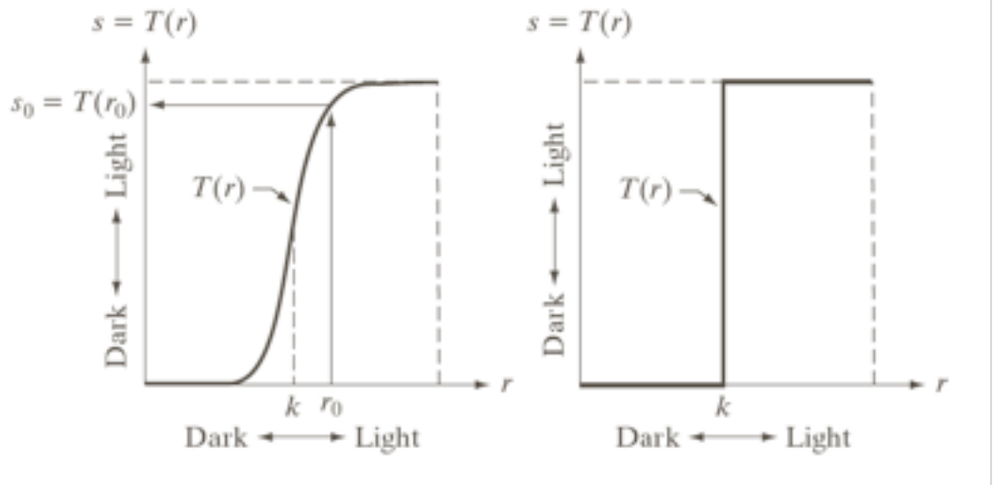
contrast stretched



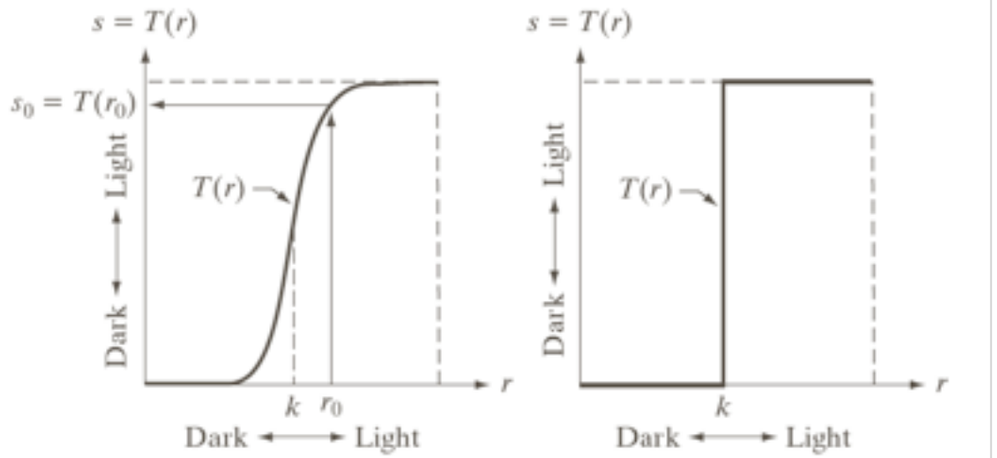
Chapter 3

Intensity Transformations & Spatial Filtering

Thresholding: we choose a threshold parameter k: intensities below this parameters are set to 0, those above are set to 1.
This produces a binary image.



Thresholding: we choose a threshold parameter k: intensities below this parameters are set to 0, those above are set to 1.
This produces a binary image.



Thresholded image (setting to 1 the values above 150, 0 the others)

Other maps of the type $s=T(r)$

Negative map: $s = (L-1) - r$

useful when one is interested in gray/white areas and the black areas are dominating

Other maps of the type $s=T(r)$

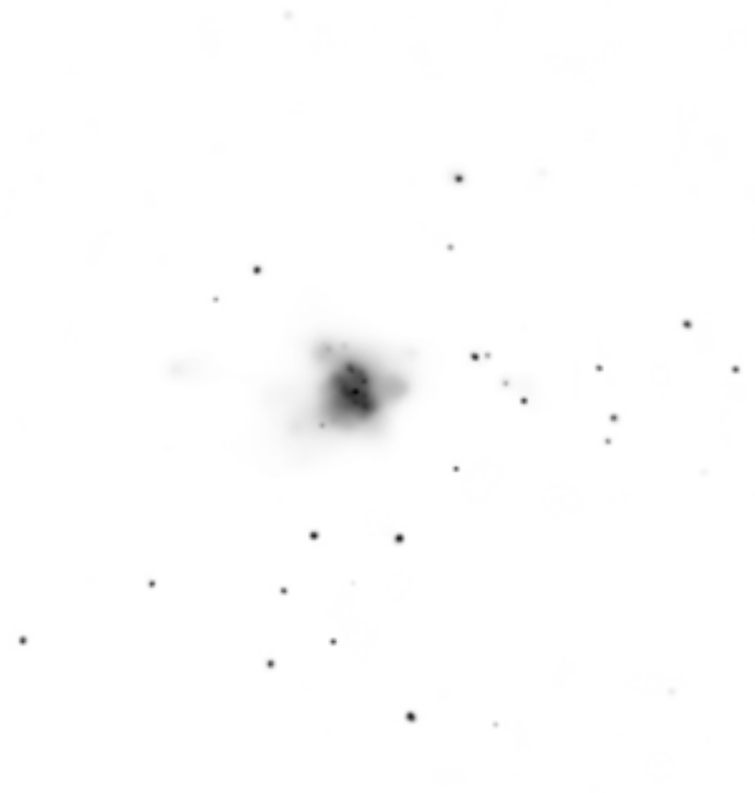
Negative map: $s = (L-1) - r$

useful when one is interested in gray/white areas and the black areas are dominating

Other maps of the type $s=T(r)$

Negative map: $s = (L-1) - r$

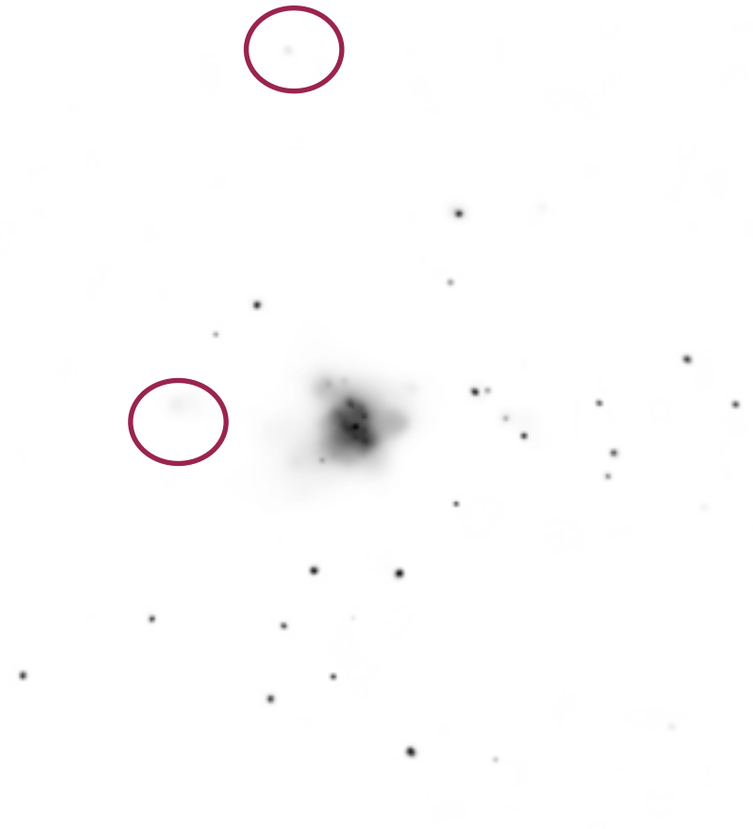
useful when one is interested in gray/white areas and the black areas are dominating



Other maps of the type $s=T(r)$

Negative map: $s = (L-1) - r$

useful when one is interested in gray/white areas and the black areas are dominating

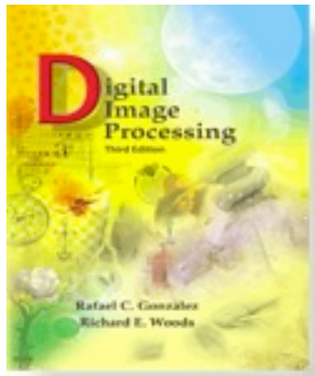


Chapter 3

Intensity Transformations & Spatial Filtering

Log map: $s = c\log(1 + r)$

maps lower intensities to a wider range, while white intensities are mapped to a smaller range

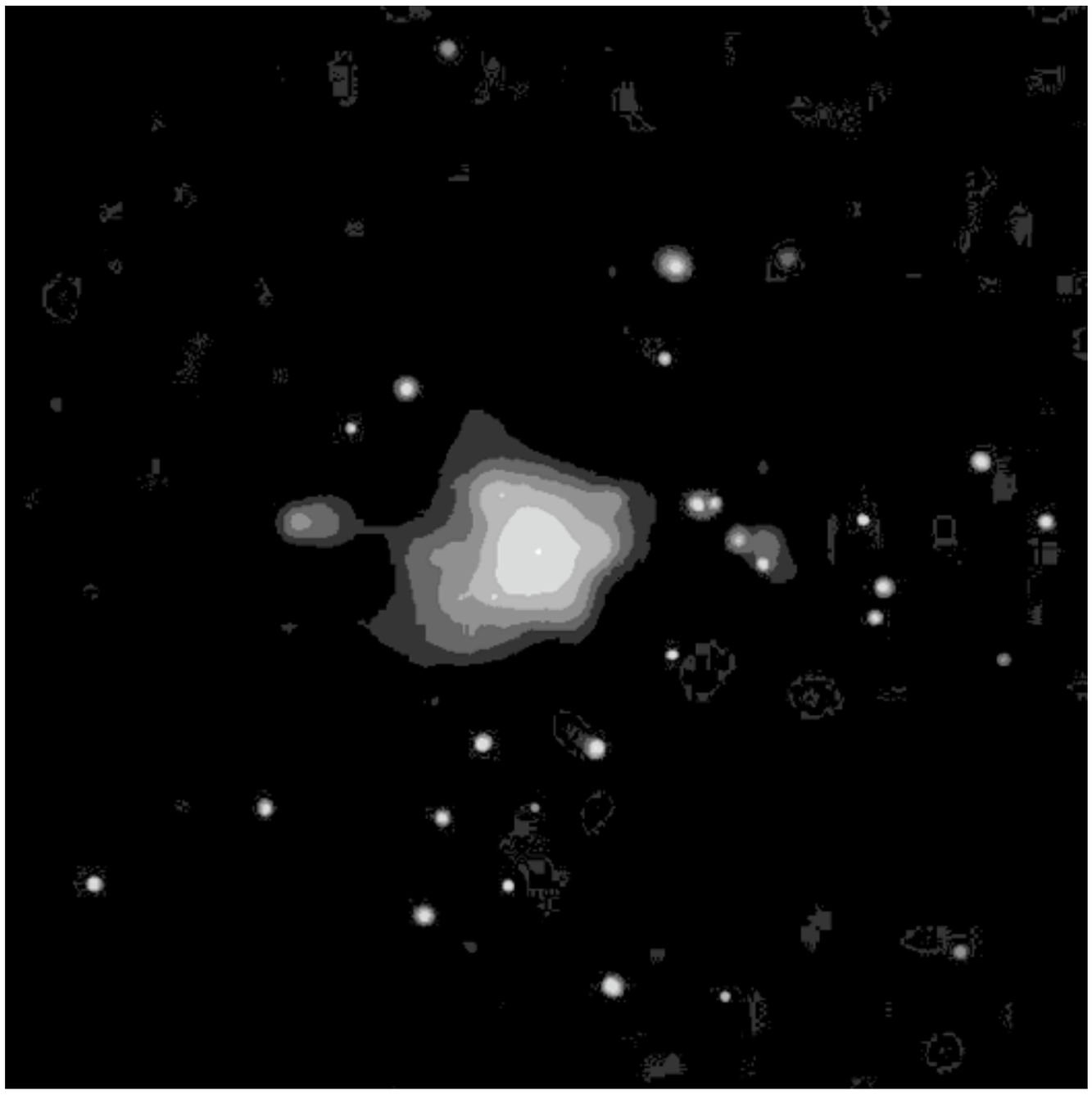


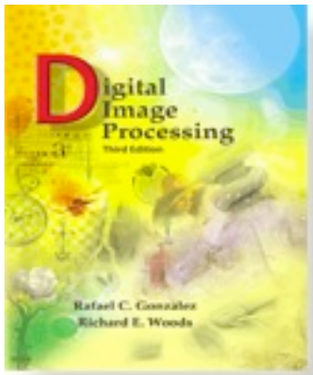
Chapter 3

Intensity Transformations & Spatial Filtering

Log map: $s = c\log(1 + r)$

maps lower intensities to a wider range, while white intensities are mapped to a smaller range





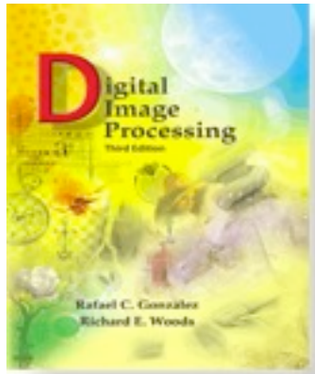
Chapter 3

Intensity Transformations & Spatial Filtering

Power-law (gamma) map:

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

These subsume the log and inverse log, but are more powerful, as just varying the gamma one gets a whole family of transformations.



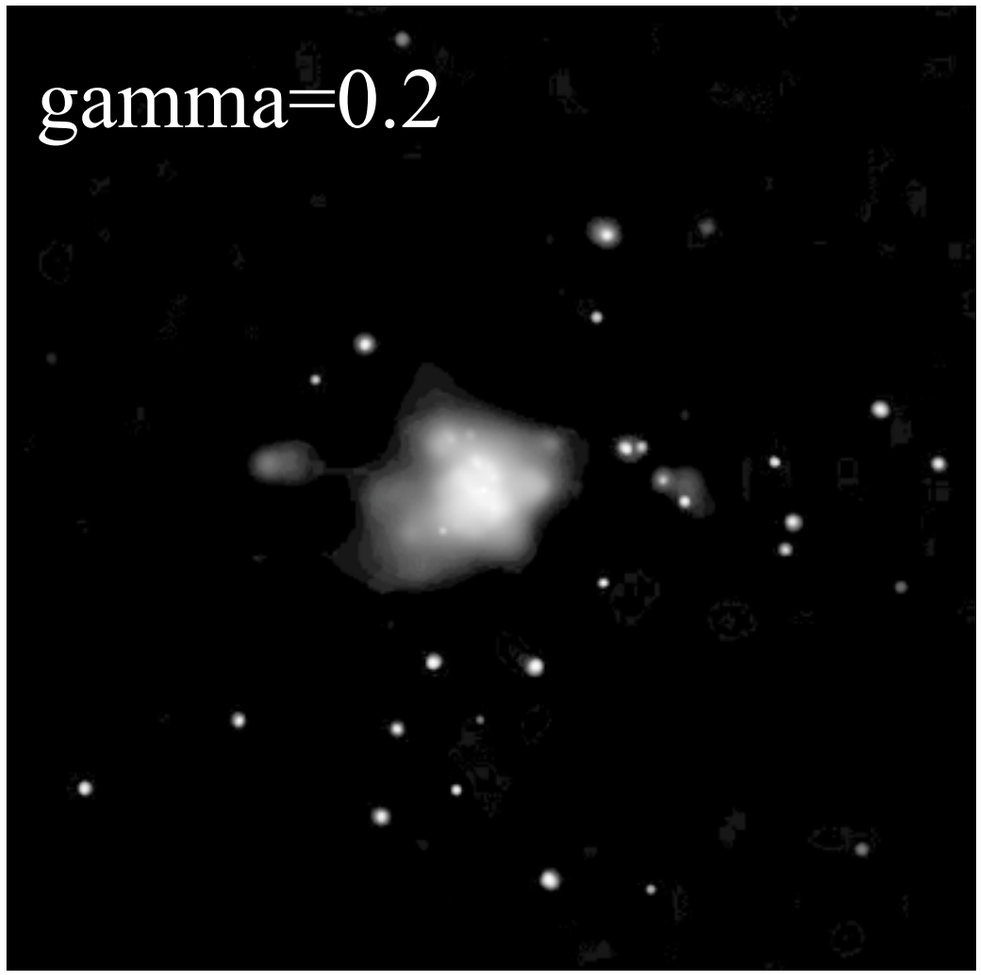
Chapter 3

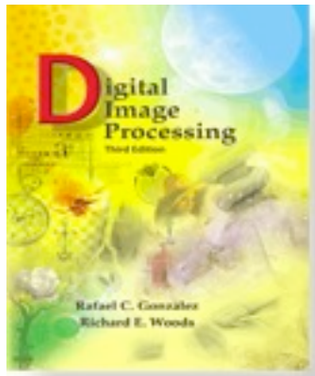
Intensity Transformations & Spatial Filtering

Power-law (gamma) map:

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

These subsume the log and inverse log, but are more powerful, as just varying the gamma one gets a whole family of transformations.





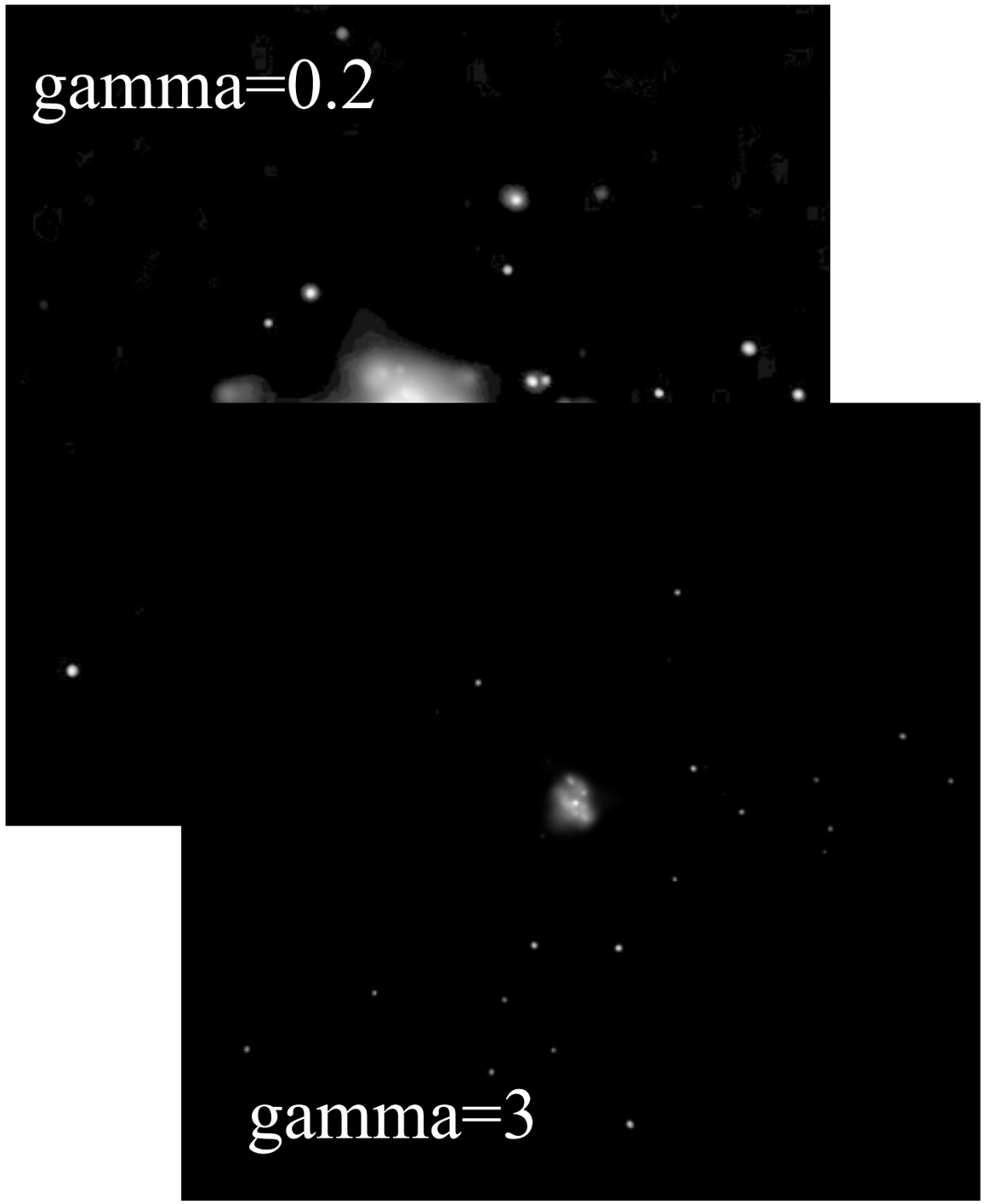
Chapter 3

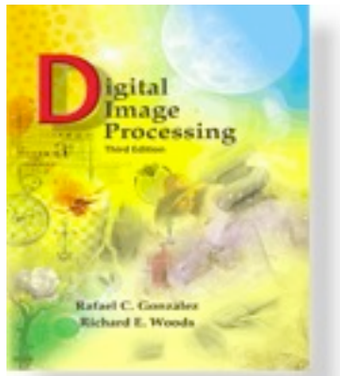
Intensity Transformations & Spatial Filtering

Power-law (gamma) map:

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

These subsume the log and inverse log, but are more powerful, as just varying the gamma one gets a whole family of transformations.





Digital Image Processing, 3rd ed.

Gonzalez & Woods

www.ImageProcessingPlace.com

Chapter 3

Intensity Transformations & Spatial Filtering

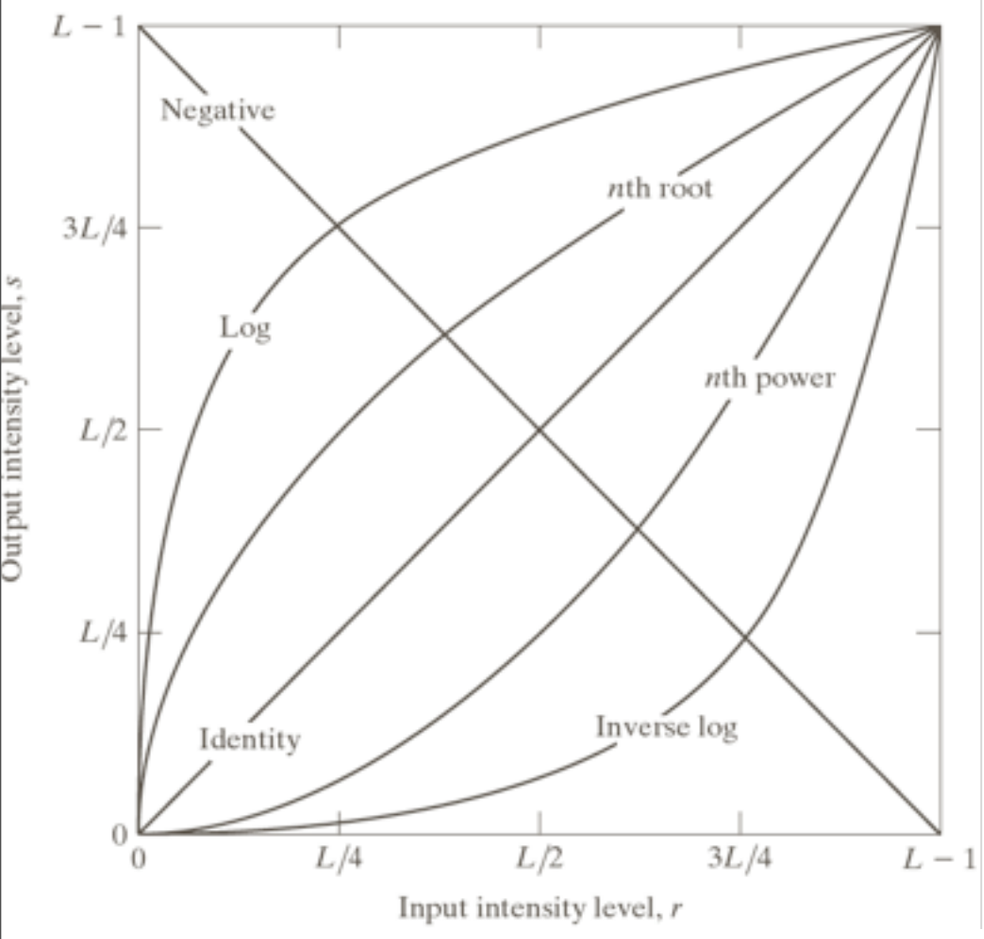


FIGURE 3.3 Some basic intensity transformation functions. All curves were scaled to fit in the range shown.

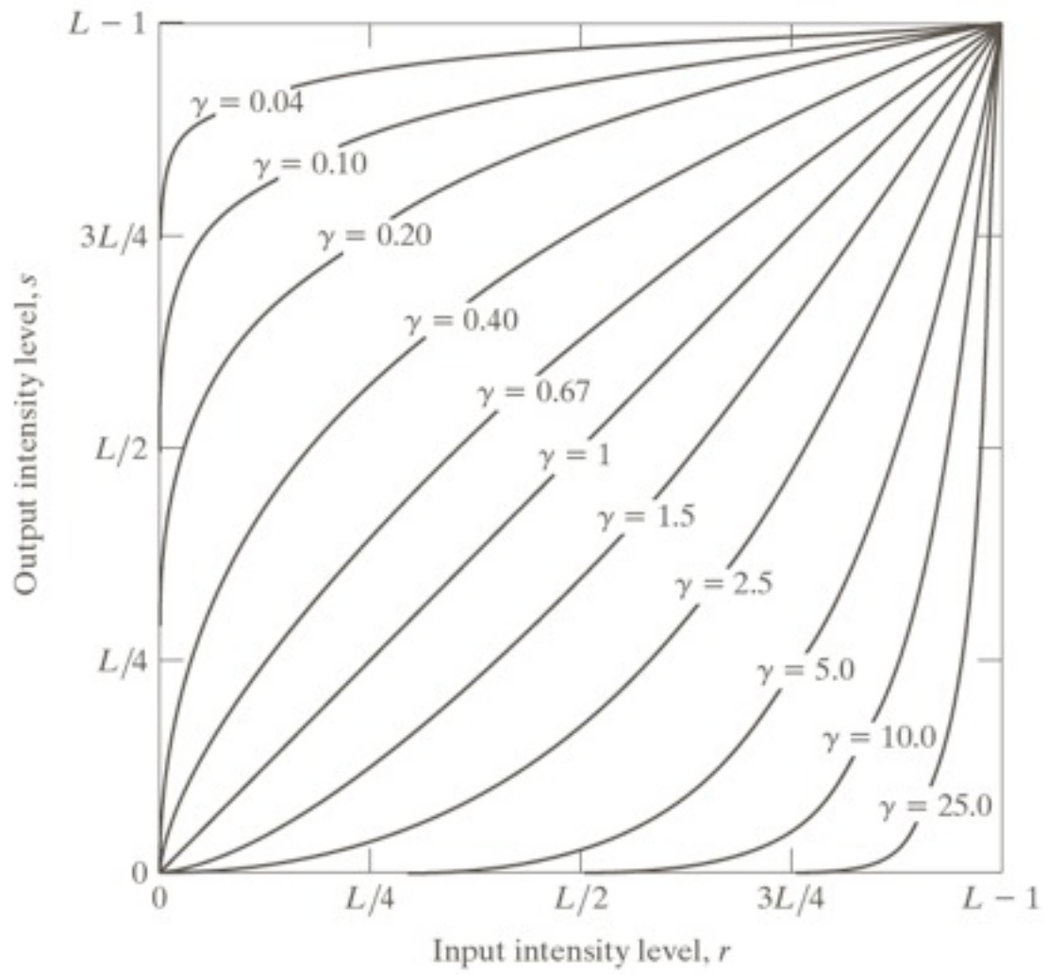
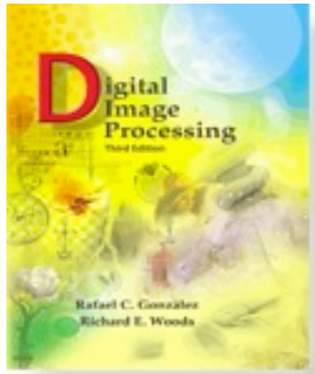


FIGURE 3.6 Plots of the equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases). All curves were scaled to fit in the range shown.

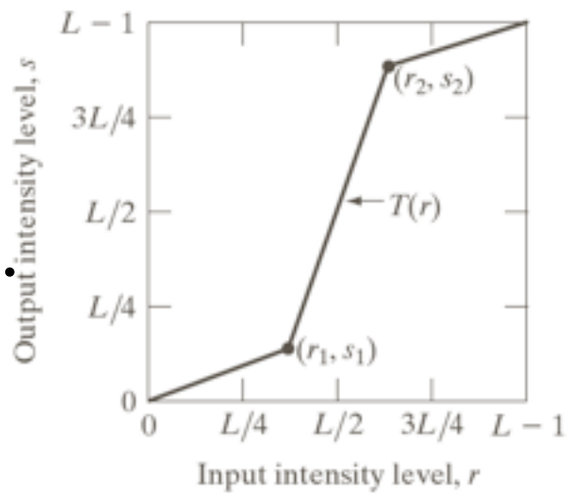


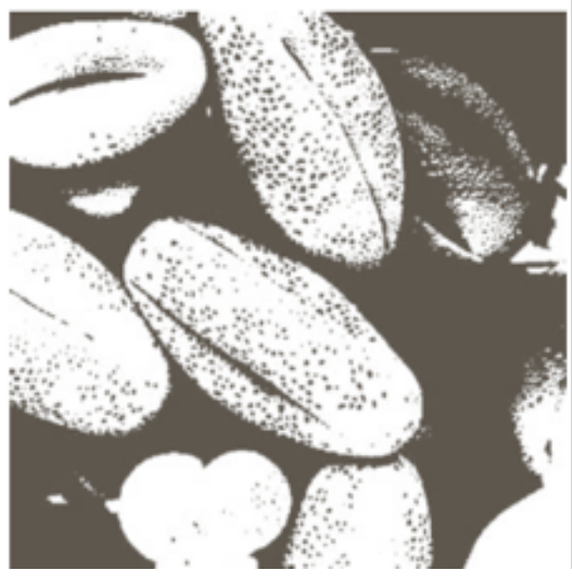
Chapter 3

Intensity Transformations & Spatial Filtering

Piecewise linear transformations:

Contrast stretching by piecewise linear transformations.

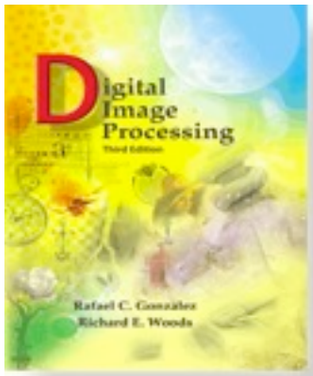




(r_1, s_1)

(r_2, s_2)

Control points



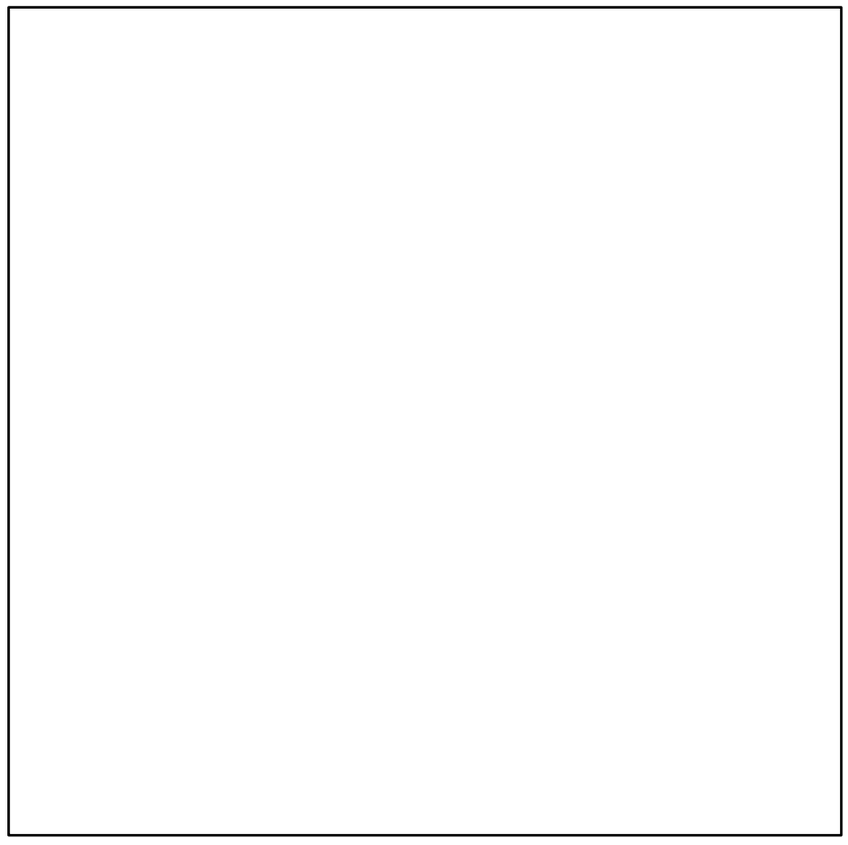
Digital Image Processing, 3rd ed.

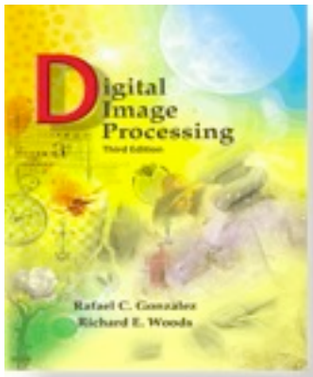
Gonzalez & Woods

www.ImageProcessingPlace.com

Chapter 3

Intensity Transformations & Spatial Filtering





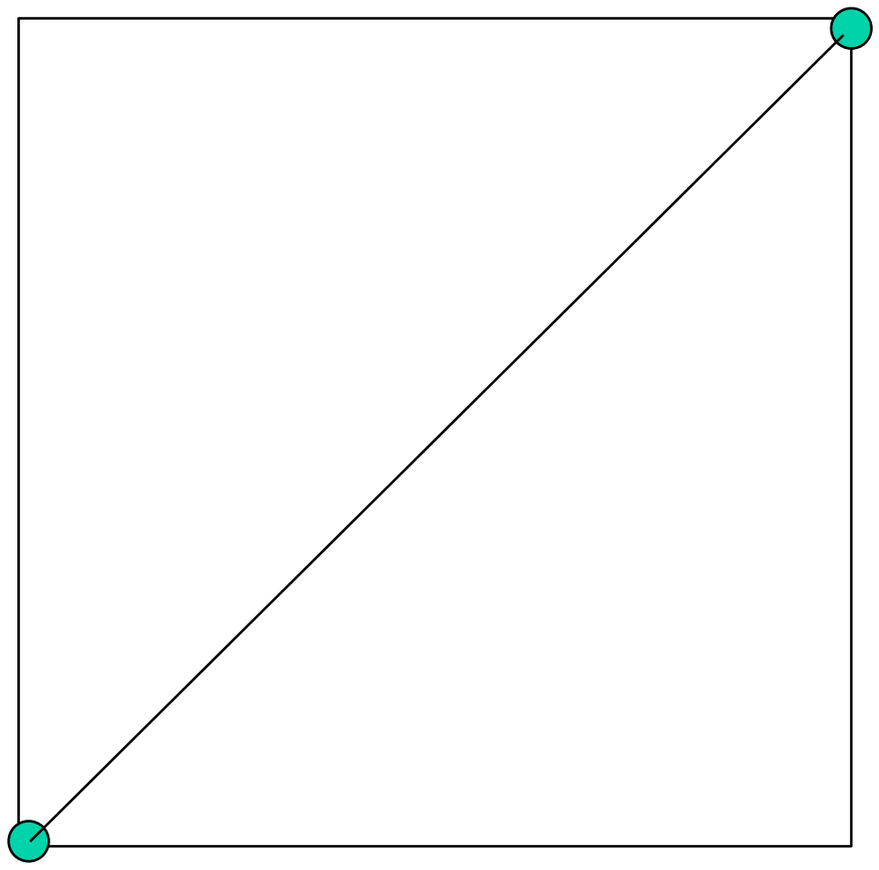
Chapter 3

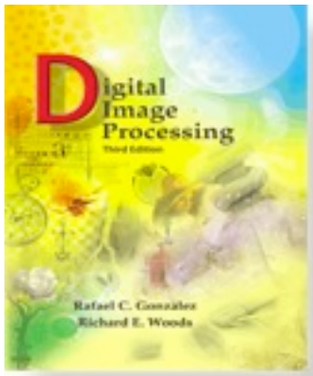
Intensity Transformations & Spatial Filtering

r1=0=s1

r2=L-1=s2

Identity





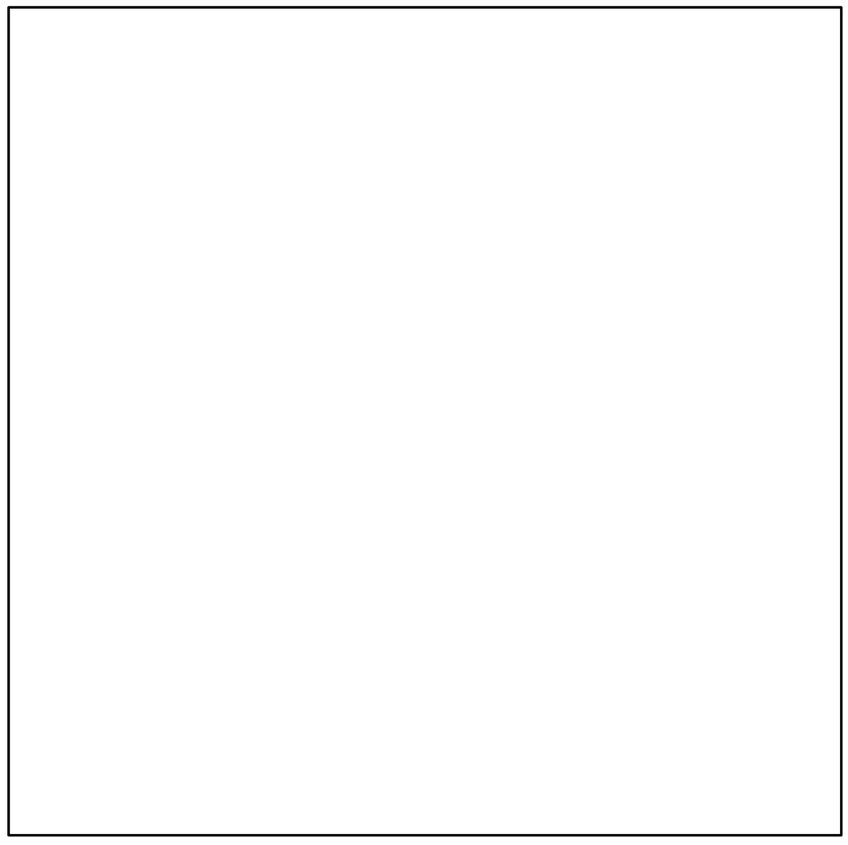
Digital Image Processing, 3rd ed.

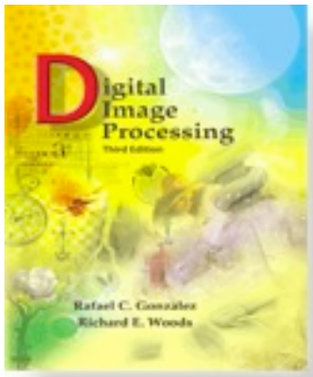
Gonzalez & Woods

www.ImageProcessingPlace.com

Chapter 3

Intensity Transformations & Spatial Filtering

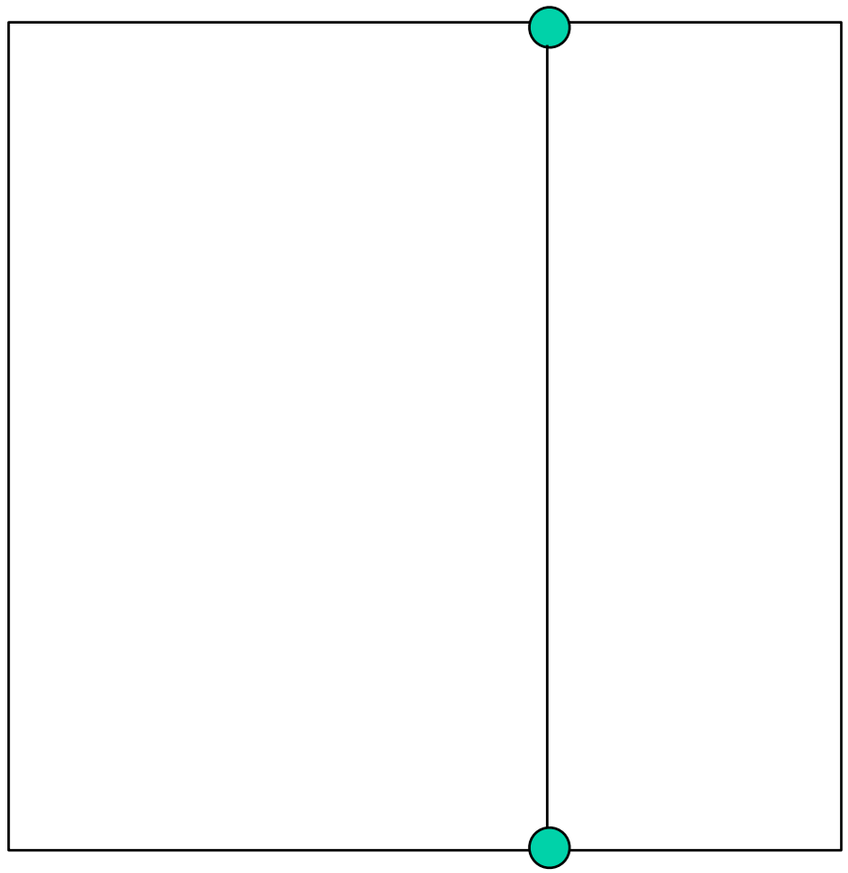




Chapter 3

Intensity Transformations & Spatial Filtering

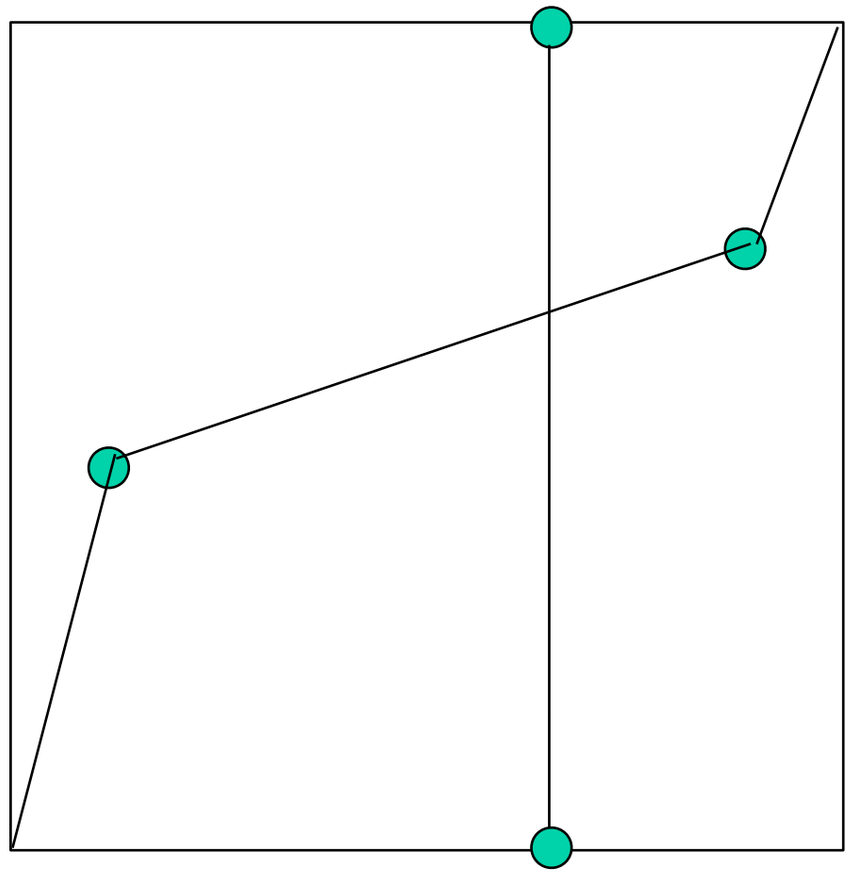
r1=r2
s1=0, s2=L-1
Thresholding



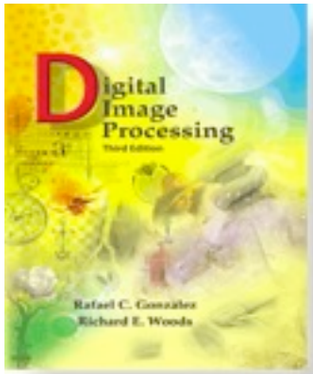
Chapter 3

Intensity Transformations & Spatial Filtering

r1=r2
s1=0, s2=L-1
Thresholding

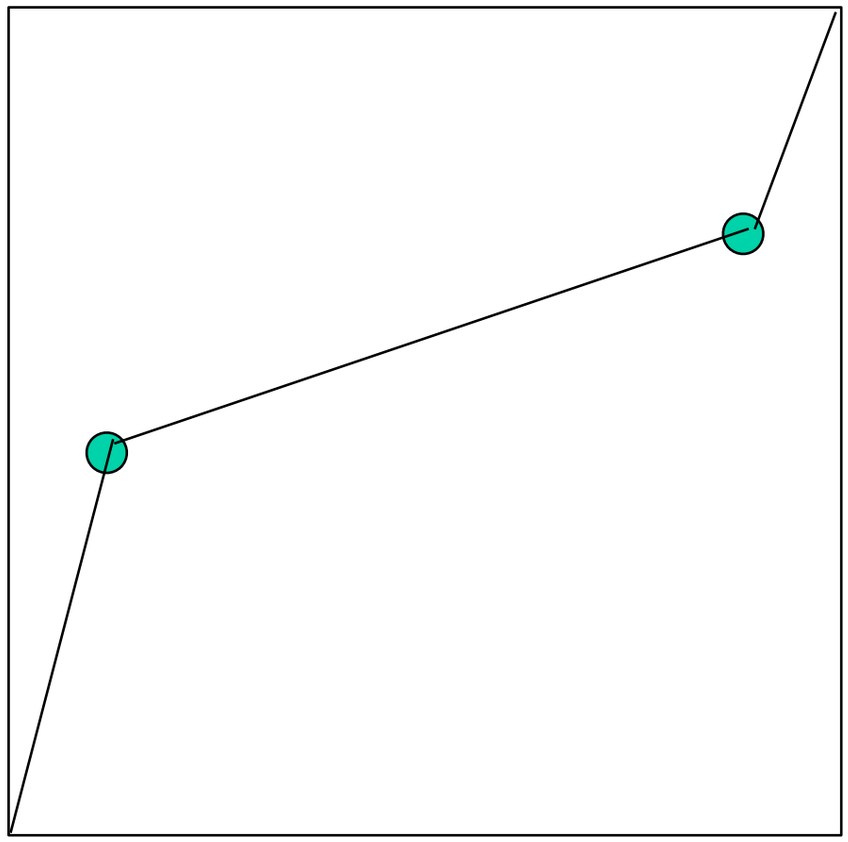


Other combinations give different effects on the image.



Chapter 3

Intensity Transformations & Spatial Filtering



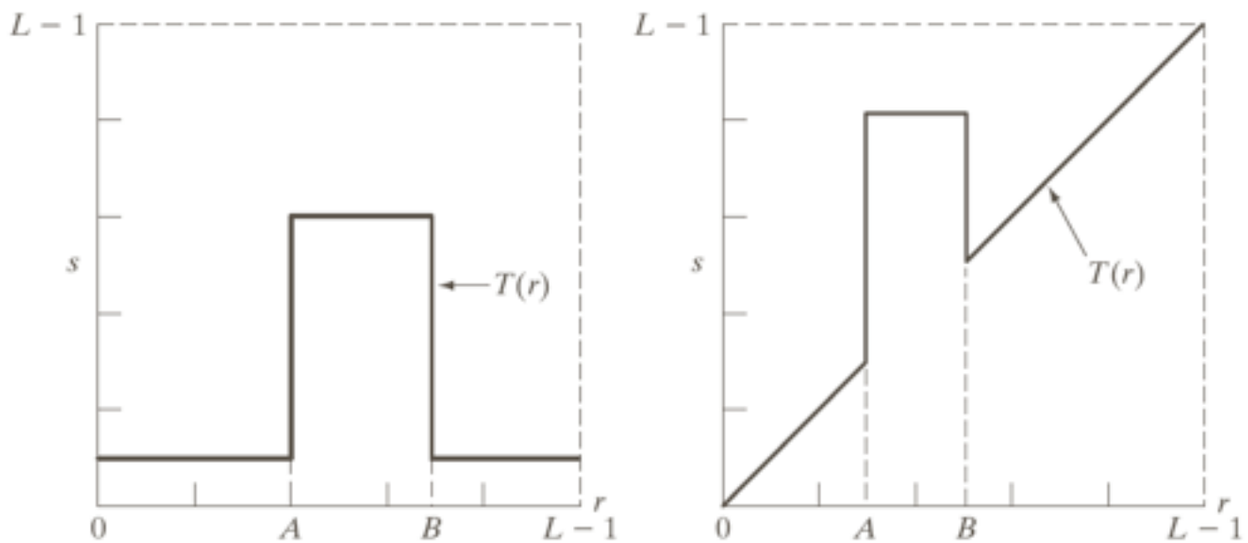
Other combinations give different effects on the image.

Chapter 3

Intensity Transformations & Spatial Filtering

Intensity level slicing:
to highlight a specific range of intensities.

Most popularly implemented as:
showing white in one range and black in the rest
showing white in one range and graytone in the rest

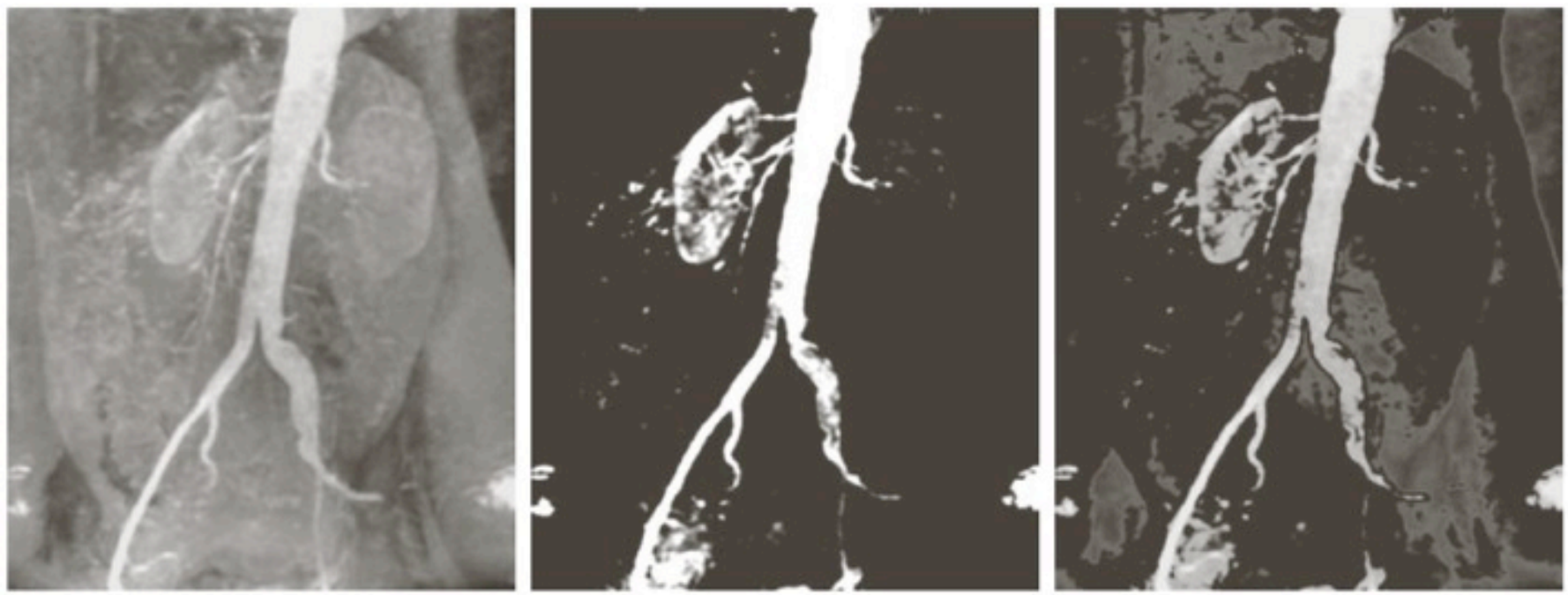


a b

FIGURE 3.11 (a) This transformation highlights intensity range $[A, B]$ and reduces all other intensities to a lower level. (b) This transformation highlights range $[A, B]$ and preserves all other intensity levels.

Chapter 3

Intensity Transformations & Spatial Filtering



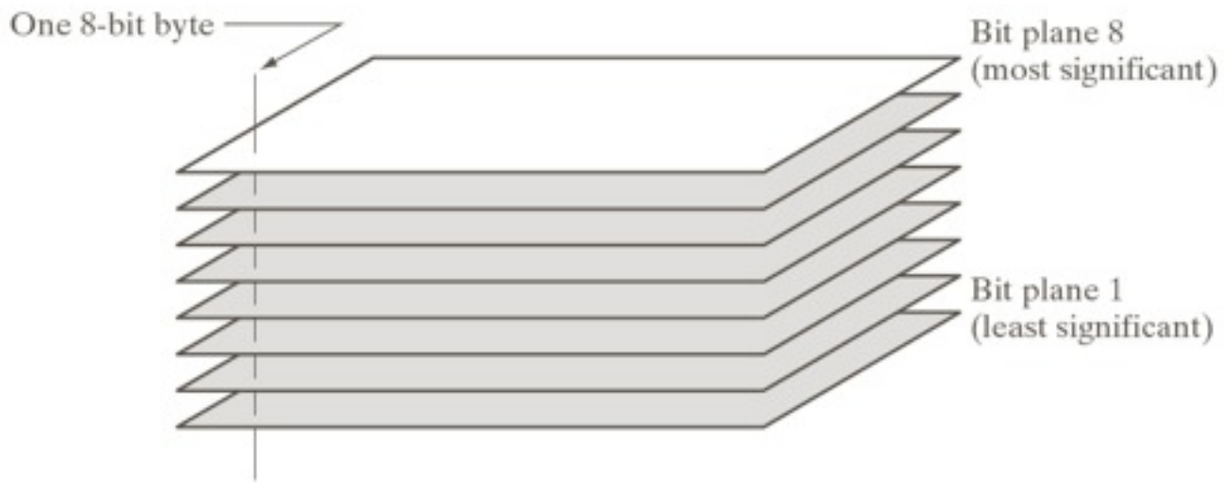
a b c

FIGURE 3.12 (a) Aortic angiogram. (b) Result of using a slicing transformation of the type illustrated in Fig. 3.11(a), with the range of intensities of interest selected in the upper end of the gray scale. (c) Result of using the transformation in Fig. 3.11(b), 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.)

Chapter 3

Intensity Transformations & Spatial Filtering

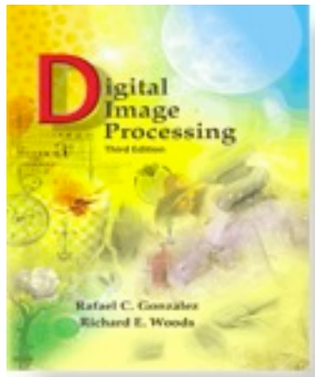
Bit slicing:



2^7 2^6 2^5 2^4 2^3 2^2 2^1 2^0

0	1	0	1	0	0	1	0

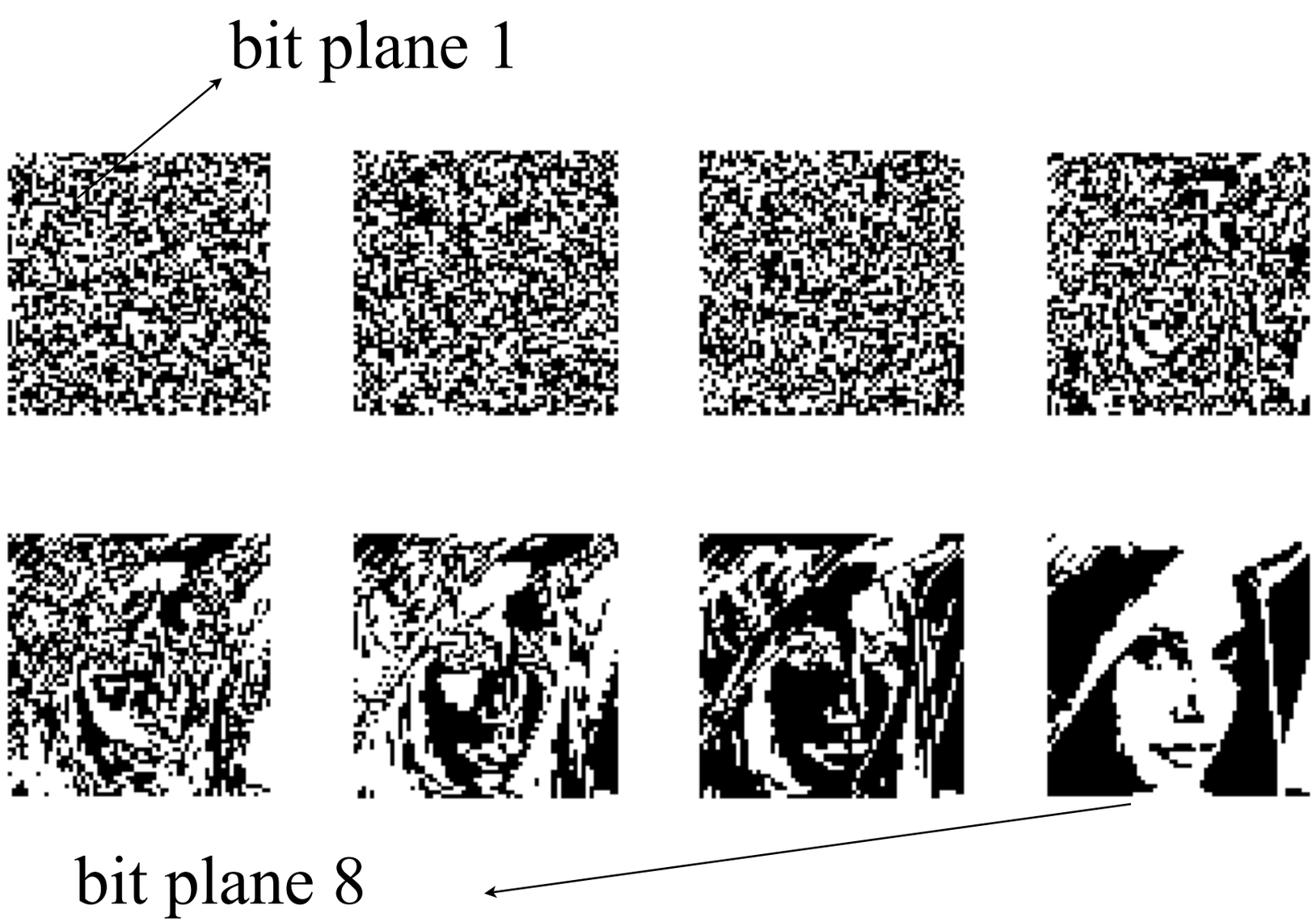
82=64+16+2



Chapter 3

Intensity Transformations & Spatial Filtering

The reconstructed image is obtained as
2^0*bit_plane1 + 2^1*bit_plane2 + …. + 2^7*bit_plane8



The most significant planes are often those with high bit number

Chapter 3

Intensity Transformations & Spatial Filtering

a b c
d e f
g h i

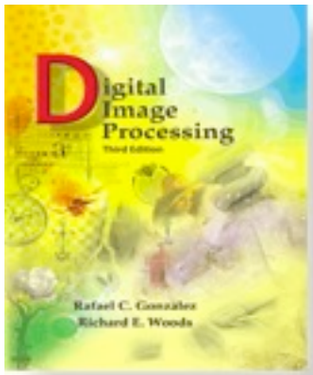
FIGURE 3.14 (a) An 8-bit gray-scale image of size 500 × 1192 pixels. (b) through (i) Bit planes 1 through 8, with bit plane 1 corresponding to the least significant bit. Each bit plane is a binary image.

Chapter 3

Intensity Transformations & Spatial Filtering

a b c

FIGURE 3.15 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).



Chapter 3

Intensity Transformations & Spatial Filtering

Histogram processing

Histogram: is a discrete function that counts how many pixels have a given intensity value.

$$h(r_k) = n_k$$

Number of pixels that have such intensity value

k-th intensity value

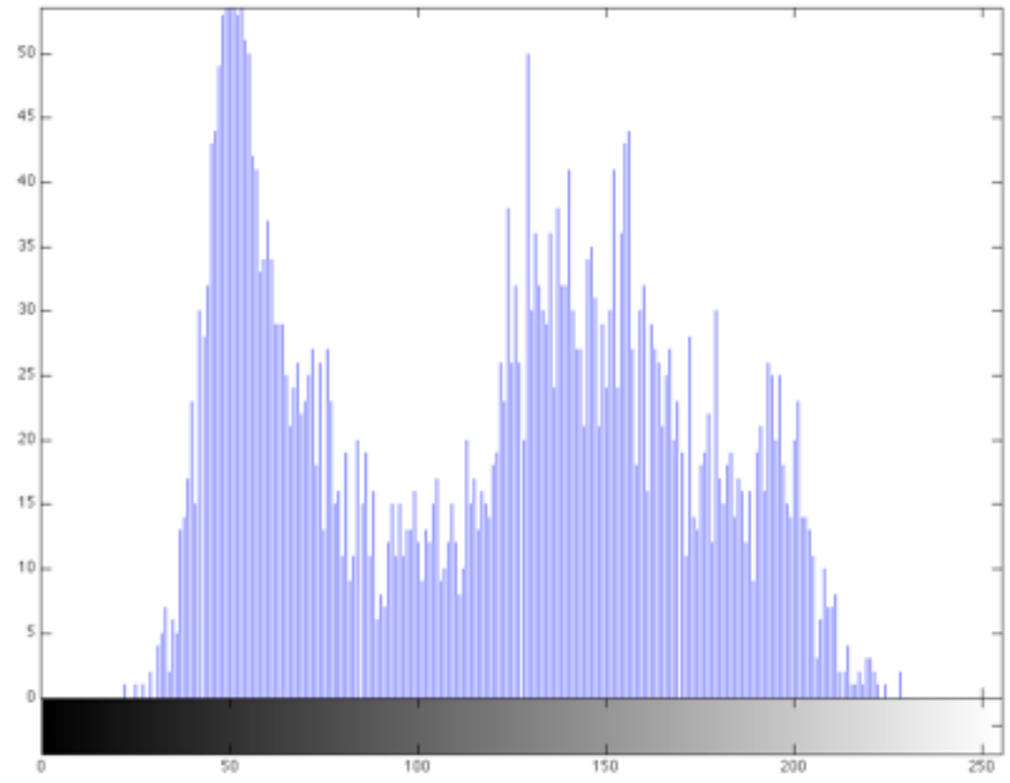
Histogram processing

Histogram: is a discrete function that counts how many pixels have a given intensity value.

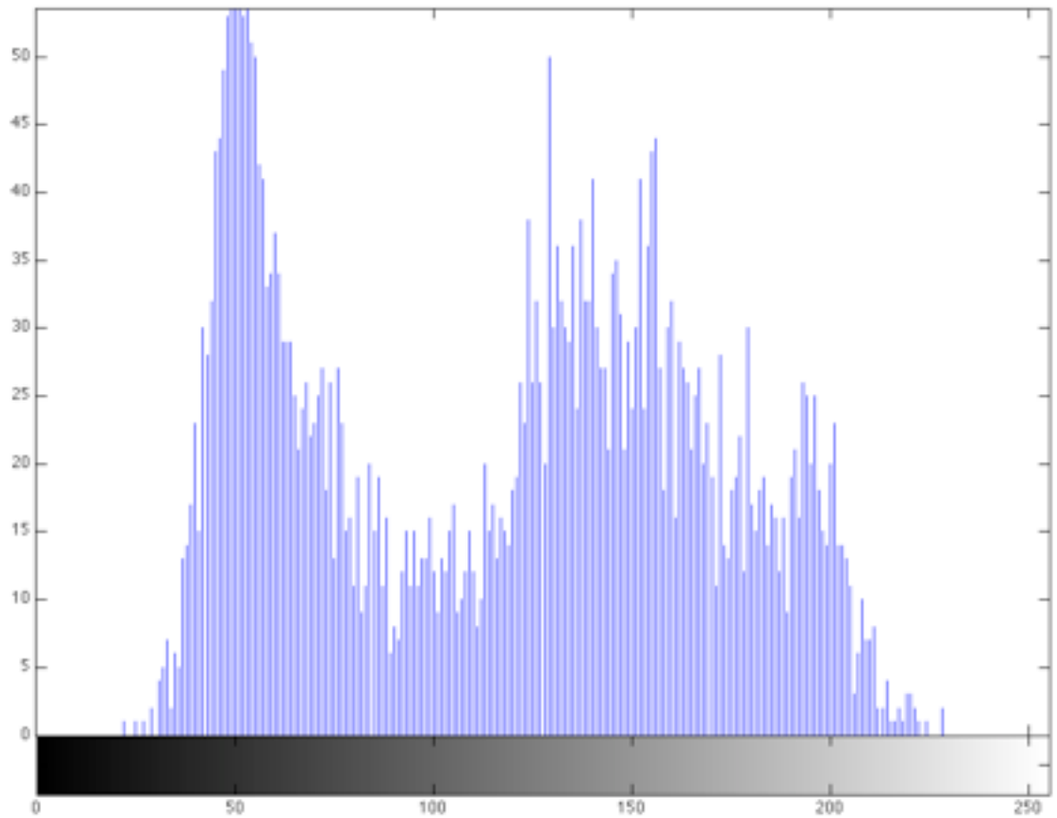
$$h(r_k) = n_k$$

Number of pixels that have such

k-th intensity value



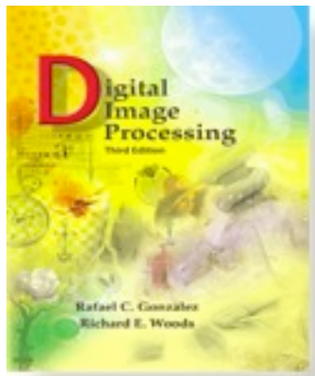
It is typical to scale n_k by MN, the number of pixels. The corresponding number represents the probability of a pixel having the given intensity.



$$\sum_k n_k = MN$$

$$p(r_k) = \frac{n_k}{MN} \qquad \sum_k p(r_k) = 1$$

Probability that a random pixels has intensity r_k.



Digital Image Processing, 3rd ed.

Gonzalez & Woods

www.ImageProcessingPlace.com

Chapter 3

Intensity Transformations & Spatial Filtering

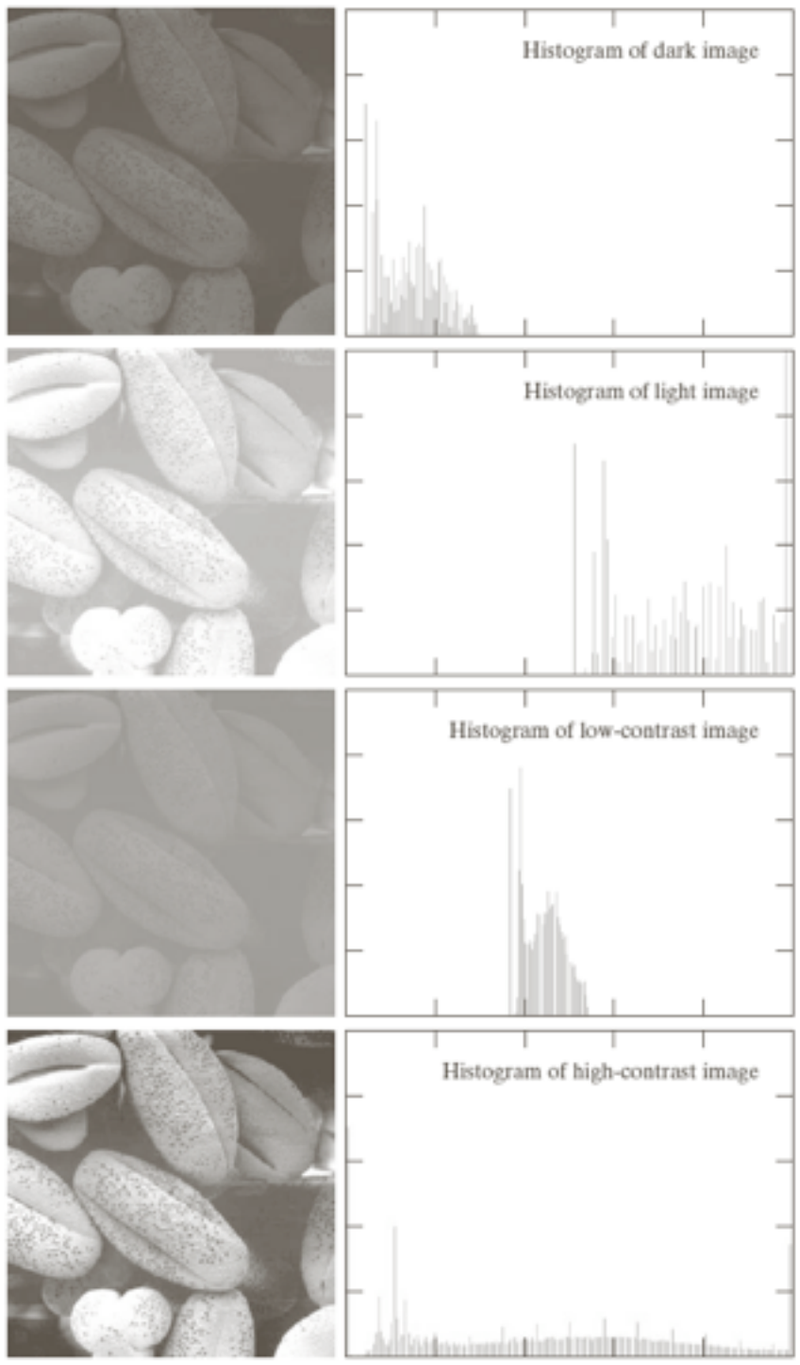


FIGURE 3.16 Four basic image types: dark, light, low contrast, high contrast, and their corresponding histograms.