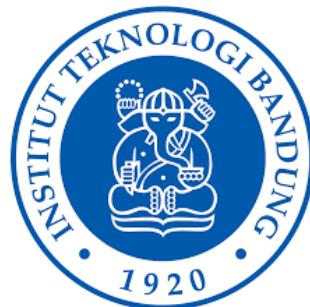


08 - Image Enhancement

(Bagian 1 – Update 2023)

IF4073 Interpretasi dan Pengolahan Citra

Oleh: Rinaldi Munir



Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
2023

Image Enhancement

- *Image enhancement* = perbaikan kualitas citra agar tampak lebih baik
- Tujuan: memperoleh citra yang lebih sesuai digunakan untuk aplikasi lebih lanjut (misal untuk mengenali objek di dalam citra).
- Merupakan satu proses awal (*preprocessing*) di dalam computer vision
- Mengapa memerlukan *image enhancement*?
 - citra sering mengandung derau (*noise*)
 - citra terlihat terlalu terang/gelap, citra kurang tajam, kabur (*blur*)
 - ada cacat saat akuisisi citra disebabkan oleh:
 - lensa: *object blurring* atau *background blurring*
 - objek bergerak/kamera bergerak: *motion blurring*
 - Distorsi geometrik disebabkan oleh lensa atau sudut pengambilan gambar



Noisy image



Citra dengan kontras terlalu gelap



Motion blur



Dark face image for recognition

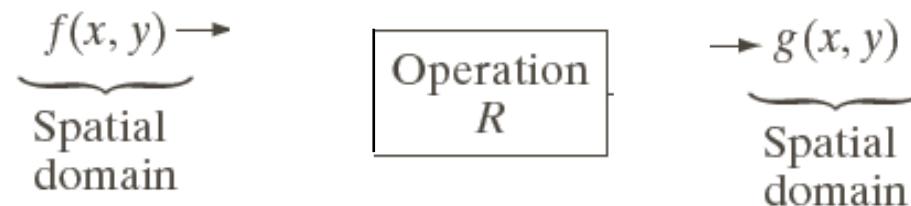


Blur vehicle plate number

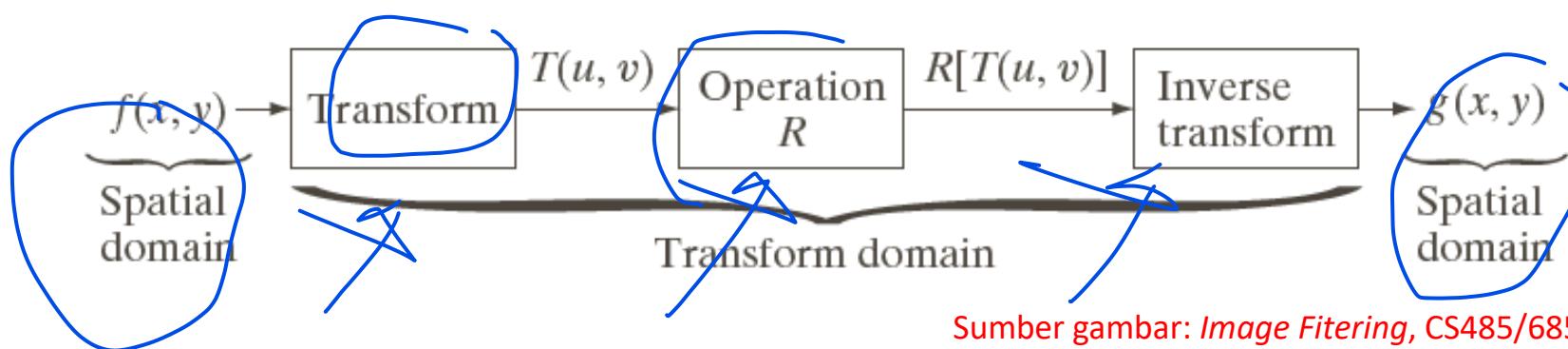
- Berdasarkan ranah (domain) operasinya, metode-metode untuk perbaikan kualitas citra dapat dikelompokkan menjadi dua kategori:

- Image enhancement* dalam ranah spasial
- Image enhancement* dalam ranah frekuensi

- Spatial Domain



- Frequency Domain (misalnya menggunakan *Fourier Transform*)



- Metode-metode *image enhancement* dalam ranah spasial dilakukan dengan memanipulasi secara langsung pixel-pixel di dalam citra.
- Metode-metode *image enhancement* dalam ranah frekuensi dilakukan dengan mengubah citra terlebih dahulu dari ranah spasial ke ranah frekuensi, baru kemudian memanipulasi nilai-nilai frekuensi tersebut.
- Masing-masing ranah operasi digunakan untuk tujuan spesifik, karena tidak semua perbaikan citra dapat dilakukan dalam ranah spasial.
- Materi di dalam PPT ini membahas metode-metode *image enhancement* dalam ranah spasial terlebih dahulu.

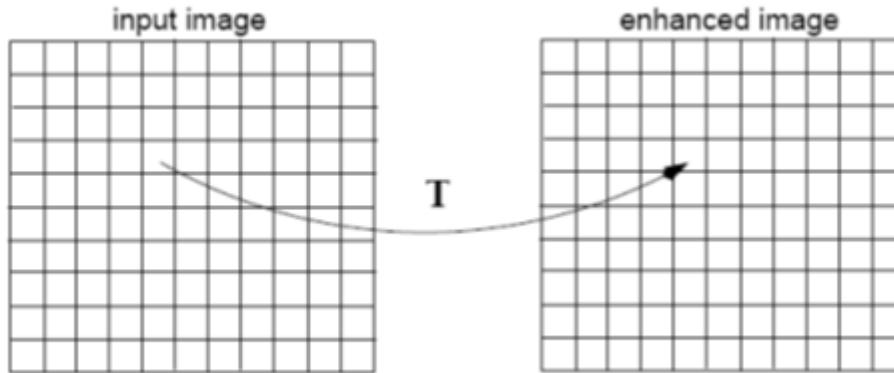
Metode dalam Ranah Spasial

- Misalkan:
 - $f(x,y)$: citra input
 - $g(x,y)$: citra output
 - T adalah operator terhadap f
- Metode pemrosesan citra dalam ranah spasial dinyatakan sebagai:

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

- T bisa beroperasi pada satu *pixel*, sekelompok *pixel* bertetangga, atau keseluruhan pixel di dalam citra.
- Jadi, metode dalam ranah spasial dapat dilakukan pada aras titik (*pixel*), aras lokal, dan aras global.

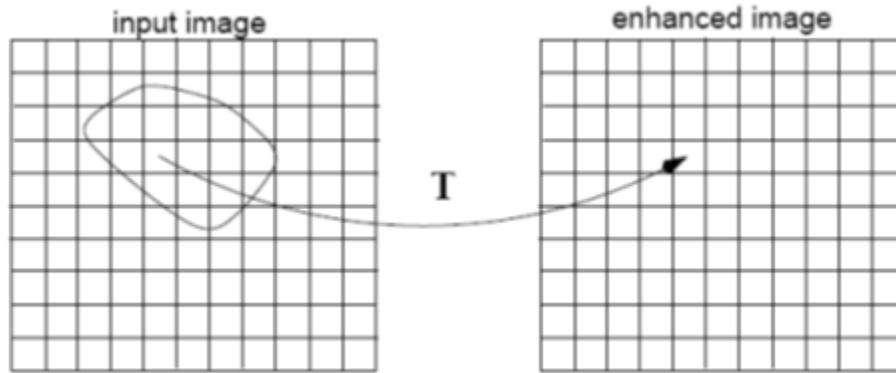
Aras titik



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

T operates on 1 pixel

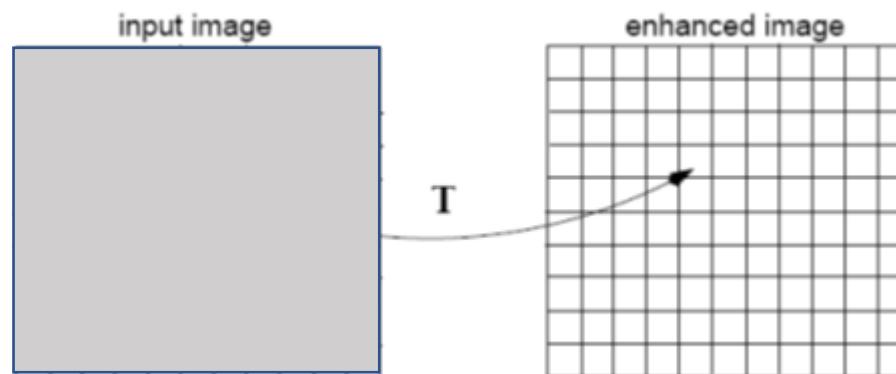
Aras lokal



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

T operates on a neighborhood of pixels

Aras global



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

T operates on entire of pixels

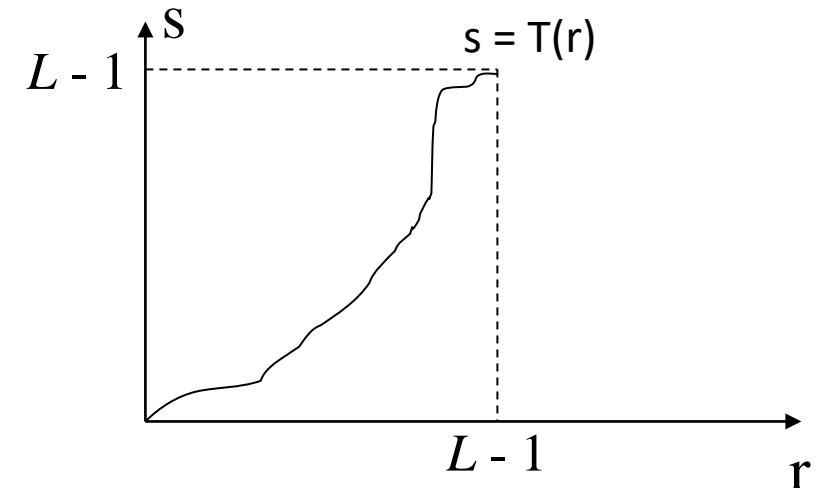
- Proses-proses yang termasuk ke dalam perbaikan kualitas citra:
 - Pengubahan kecerahan gambar (~~image brightening~~)
 - Citra negatif (image negatives)
 - Peregangan kontras (contrast stretching)
 - Pengubahan histogram citra.
 - Pelembutan citra (~~image smoothing~~)
 - Penajaman citra (~~image sharpening~~)
 - Perbaikan distorsi geometrik
 - Dll
- Operasi perbaikan kualitas citra dapat dilakukan dalam aras titik, aras lokal, atau aras global, bergantung pada metodenya.

Pemrosesan dalam aras titik (*pixel wise*)

- $g(x,y) = T [f(x,y)]$
- T hanya beroperasi pada *pixel* tunggal
- T adalah fungsi transformasi nilai *grayscale*, sehingga ditulis:

$$s = T(r)$$

r : variabel yang menyatakan nilai *grayscale* $f(x,y)$
 s : variabel yang menyatakan nilai *grayscale* $g(x,y)$

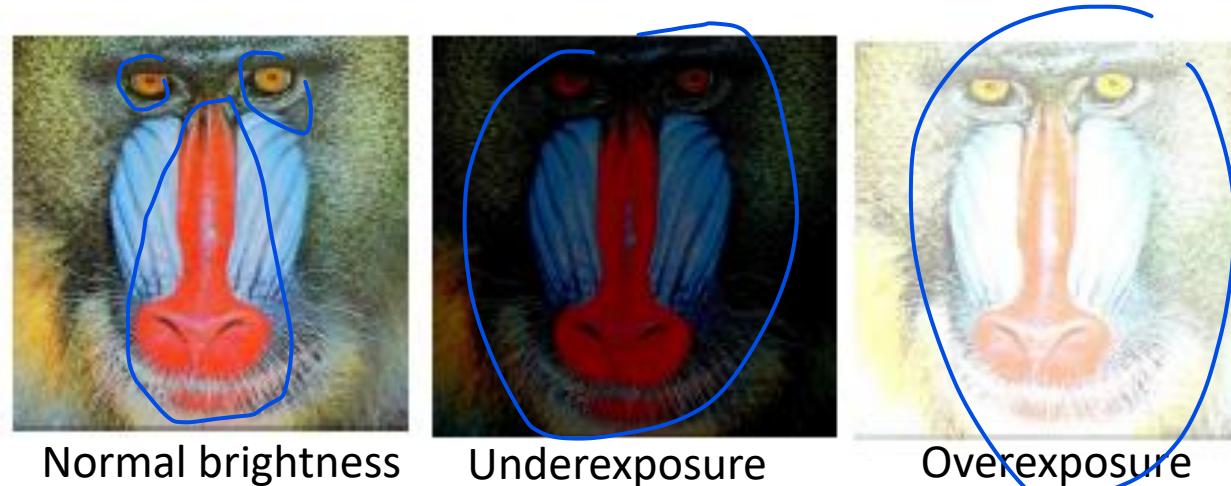


$L = 256$: pada citra grayscale 8-bit

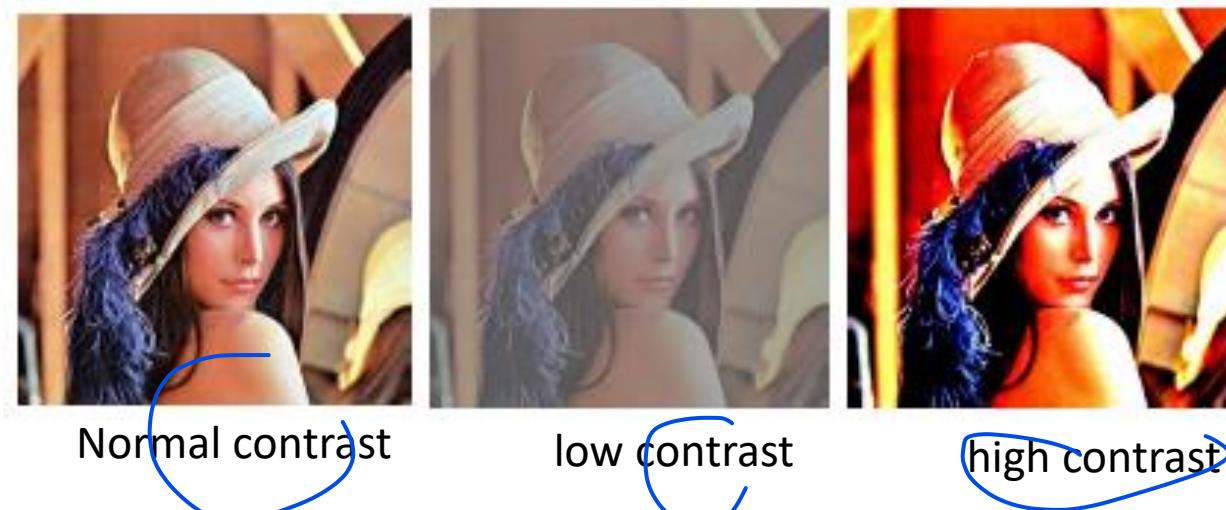
Contoh-contoh *image enhancement* dalam aras titik:

1. Mencerahkan citra (*image brightening*)
2. Menegatifkan citra (*image negatives*)
3. Peregangan kontras (*contrast stretching*)
4. *Gamma correction*
5. dll

- Perlu dipahami perbedaan antara kecerahan (*brightness*) dan kontras (*contrast*).
- *Brightness* adalah intensitas *pixel* relatif dengan *pixel* lainnya



- Kontras adalah perbedaan antara nilai maksimum (terang) dan nilai minimum (gelap) pixel



Overexposed



Correctly Exposed

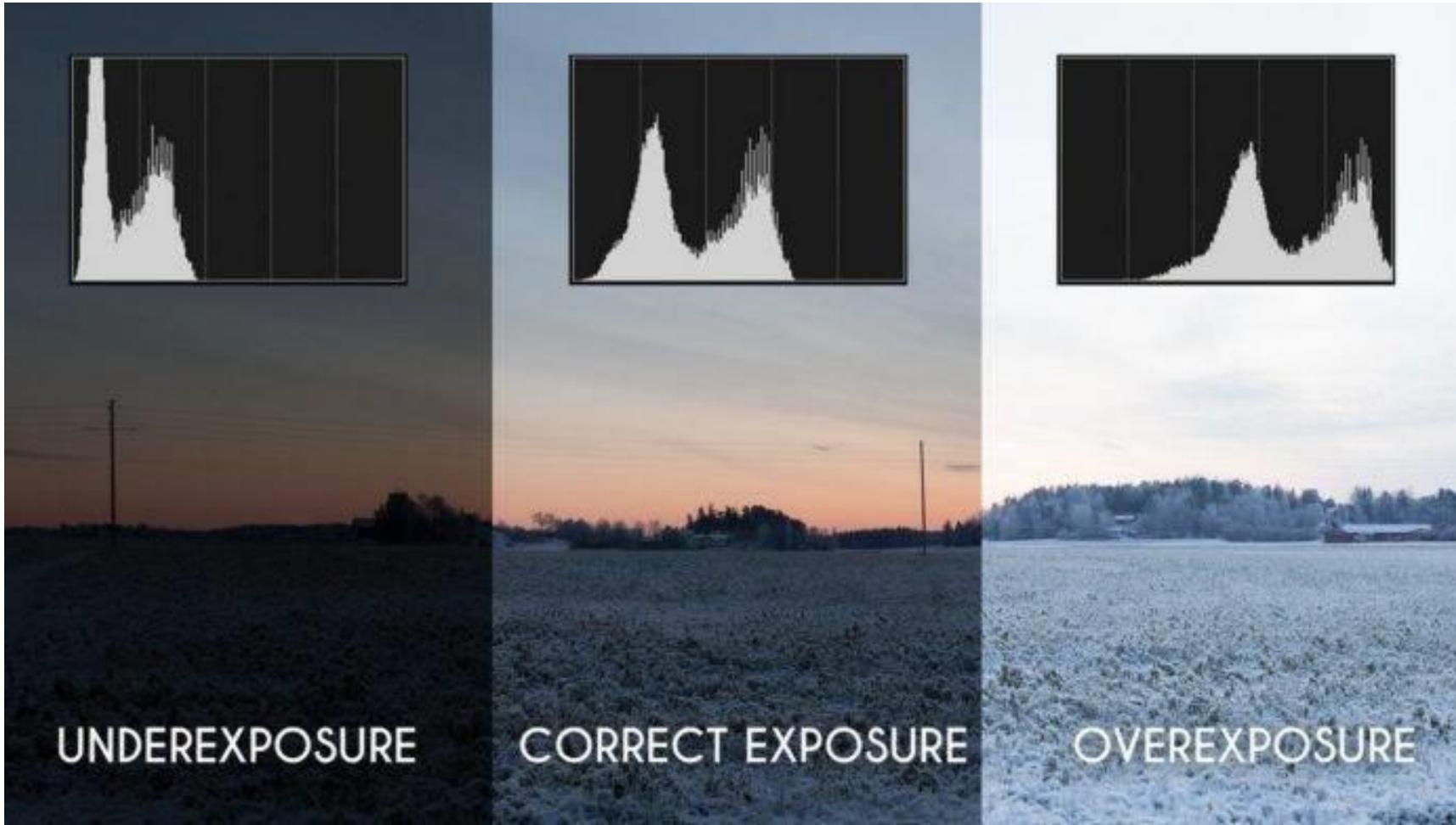


Underexposed



<https://shuttermuse.com/glossary/overexposure/>

- Perbedaan histogram pada *underexposure*, *normal*, dan *overexposure* image



UNDEREXPOSURE

Menumpuk di kiri

CORRECT EXPOSURE

Tersebar lebih merata

OVEREXPOSURE

Menumpuk di kanan



Low Contrast Image



High Contrast Image

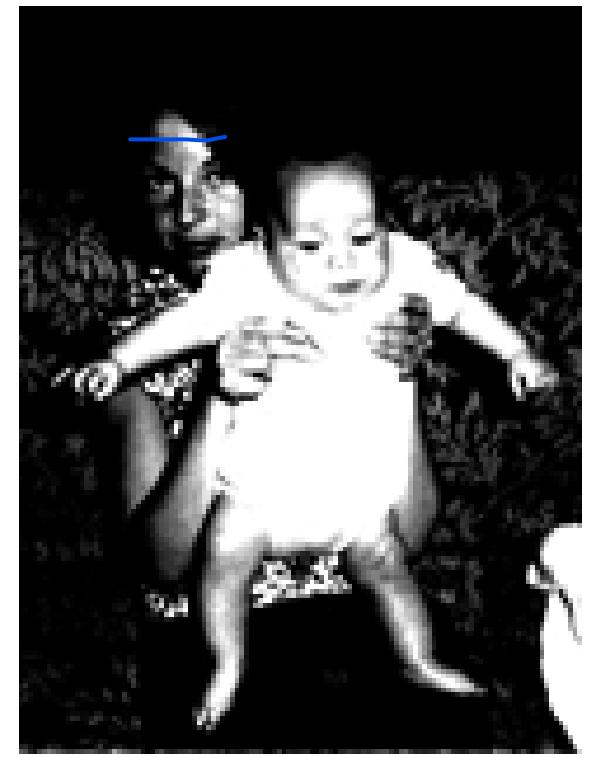
<https://theailearner.com/2019/01/30/what-is-contrast-in-image-processing/>



LOW
CONTRAST

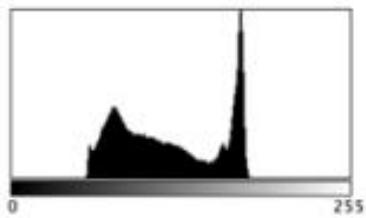


BALANCED

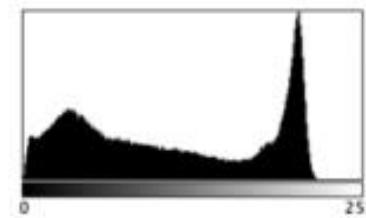


HIGH
CONTRAST

- Perbedaan histogram pada citra *low-contrast*, *high contrast*, dan *normal contrast*



low contrast

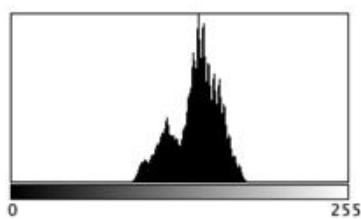


normal contrast

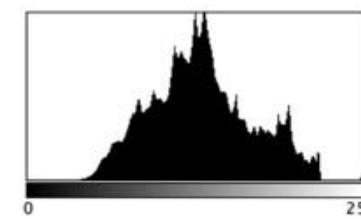


high contrast

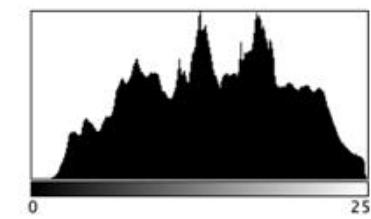
low contrast image



medium contrast image



high contrast image

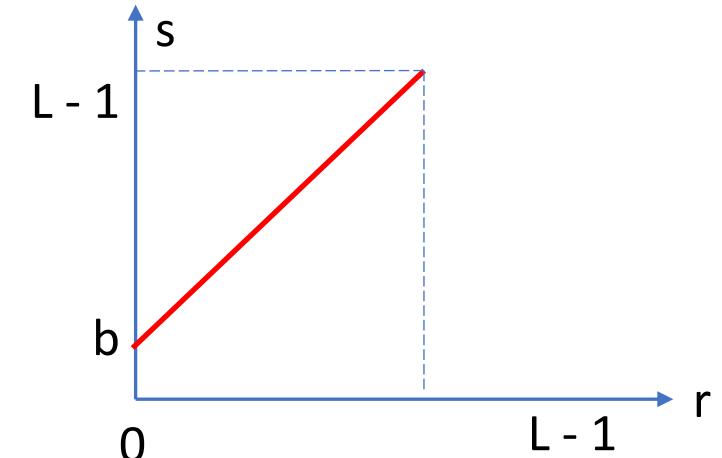


1. Pencerahan citra (*image brightening*)

- Kecerahan citra dapat diperbaiki dengan menambahkan/mengurangkan sebuah konstanta kepada (atau dari) setiap *pixel*, atau mengalikan sebuah konstansta ke setiap *pixel*.

$$s = r + b$$

- Jika b positif, kecerahan citra bertambah,
Jika b negatif kecerahan citra berkurang
- Perlu operasi *clipping* jika nilai $r + b$ berada di bawah nilai intensitas minimum atau di atas nilai intensitas maksimum:
 - jika $r + b > 255$, maka $s = 255$
 - jika $r + b < 0$, maka $s = 0$



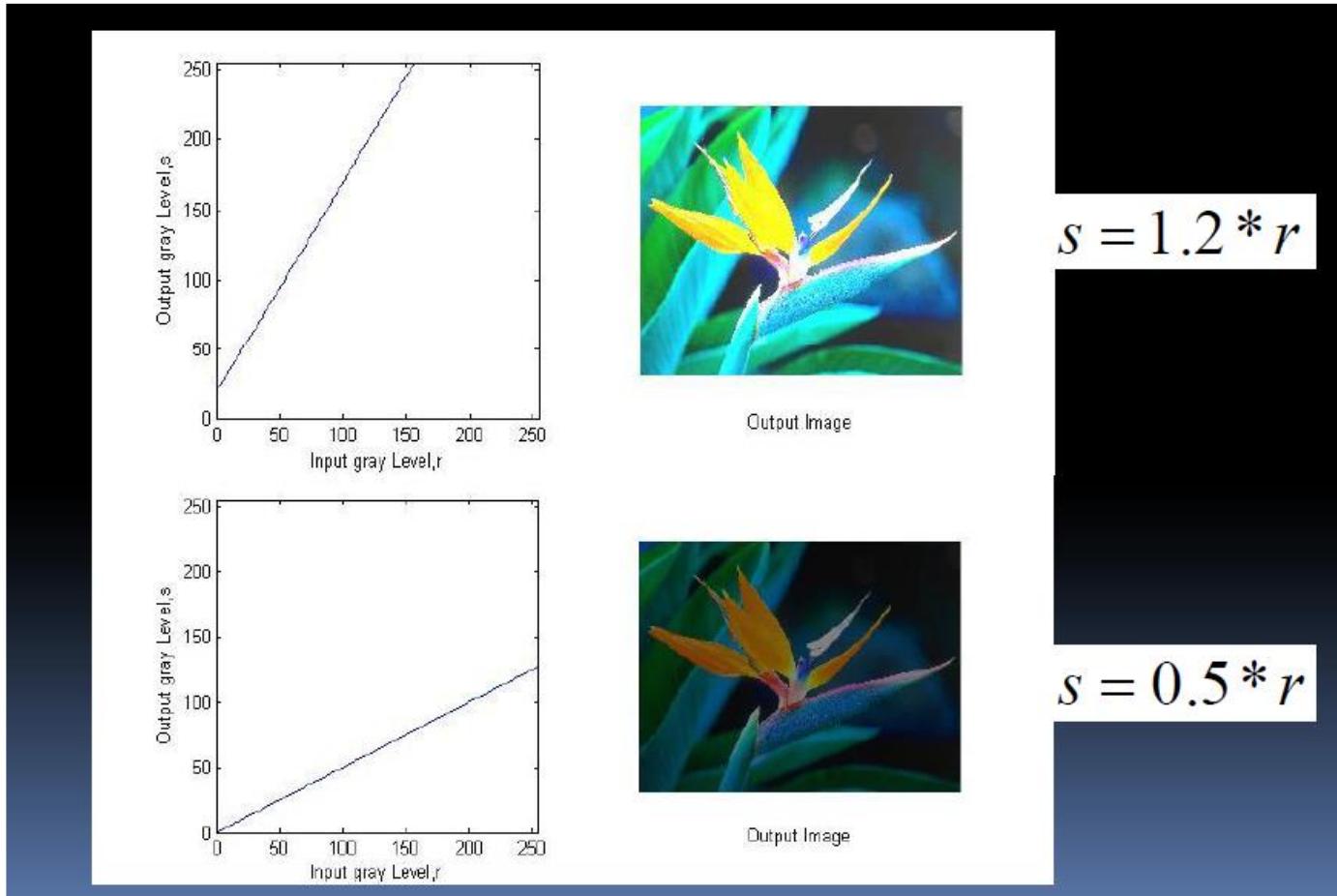


Gambar Kiri: citra Zelda (agak gelap); kanan: citra Zelda setelah operasi pencerahan citra, $b = 100$

- Operasi pencerahan yang lain adalah menggunakan rumus:

$$s = ar + b$$

a dan b adalah konstanta



Sumber gambar: Ehsan Khoramshahi,
Image enhancement in spatial domain

f = imread('lada256.bmp');
imshow(f)
g = 1.5 * f + 30;
figure, imshow(g)
h = 0.5 * f + 10;
figure, imshow(h)



f



g = 1.5 * *f* + 30;

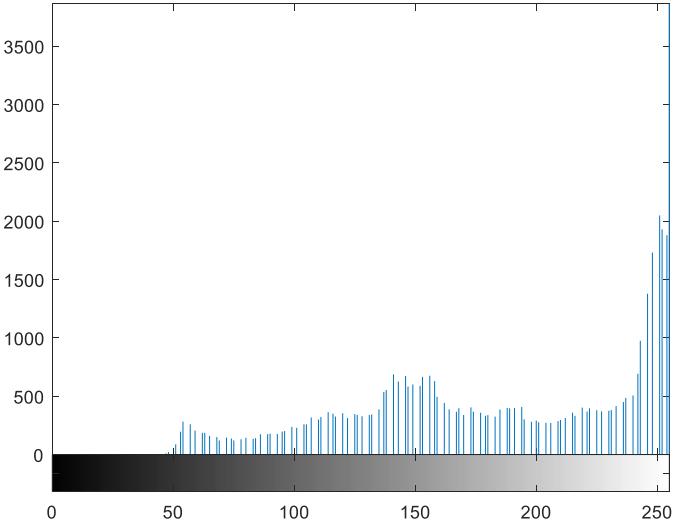
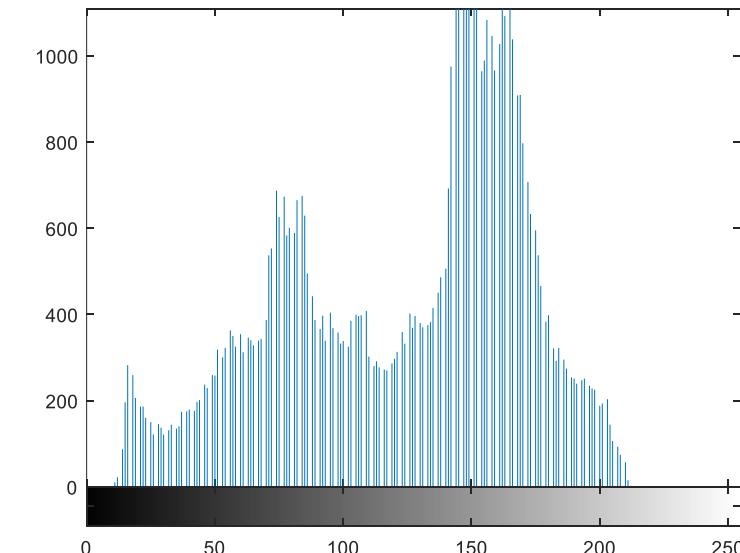


h = 0.5 * *f* + 10;

```
f = imread('bird.bmp');  
imshow(f);  
figure, imhist(f);  
g = 1.5 * f + 30;  
figure, imshow(g)  
figure, imhist(g);  
h = 0.5 * f - 10;  
figure, imshow(h);  
figure, imhist(h);
```



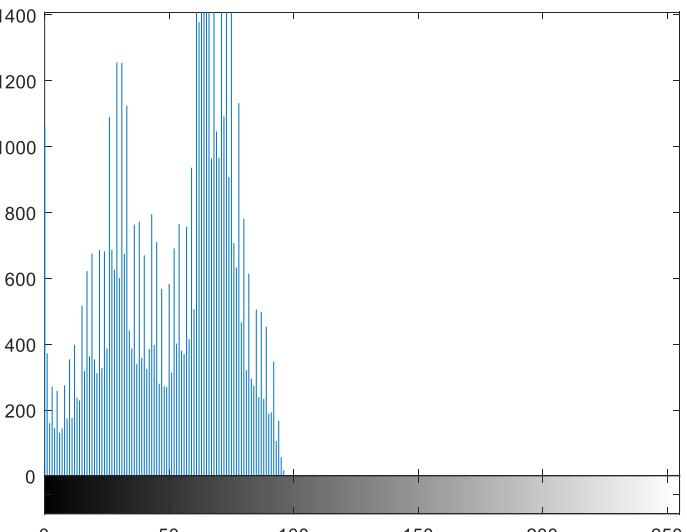
f



*g = 1.5 * f + 30;*

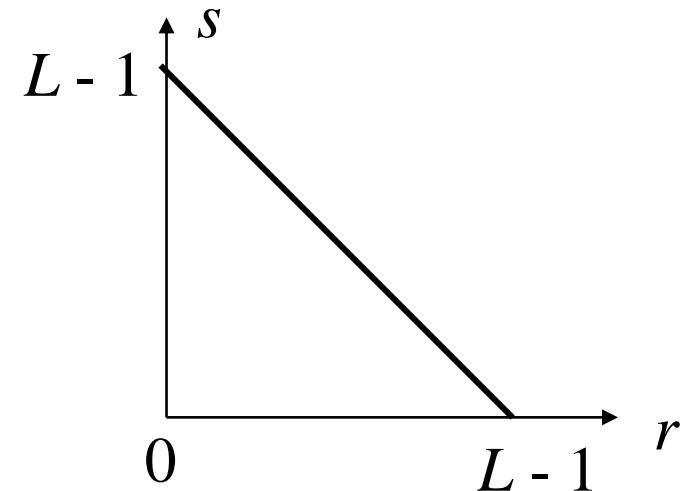


*h = 0.5 * f - 10;*



2. Menegatifkan Citra (*Image Negatives*)

- Seperti film negatif pada fotografi.
- Misalkan citra memiliki L derajat keabuan
- Caranya: kurangi nilai intensitas *pixel* dari nilai keabuan maksimum ($L - 1$)



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

Contoh pada citra *grayscale* 8-bit:

$$s = 255 - r$$



```
f = imread('girl.jpg');
g = 255 - f;
imshow(f), title('Original image');
figure, imshow(g), title('Negative image');
```

Original image



Negative image



```
f = imread('gedung-sate.jpg');  
g = 255 - f;  
imshow(f), title('Original image');  
figure, imshow(g), title('Negative image');
```

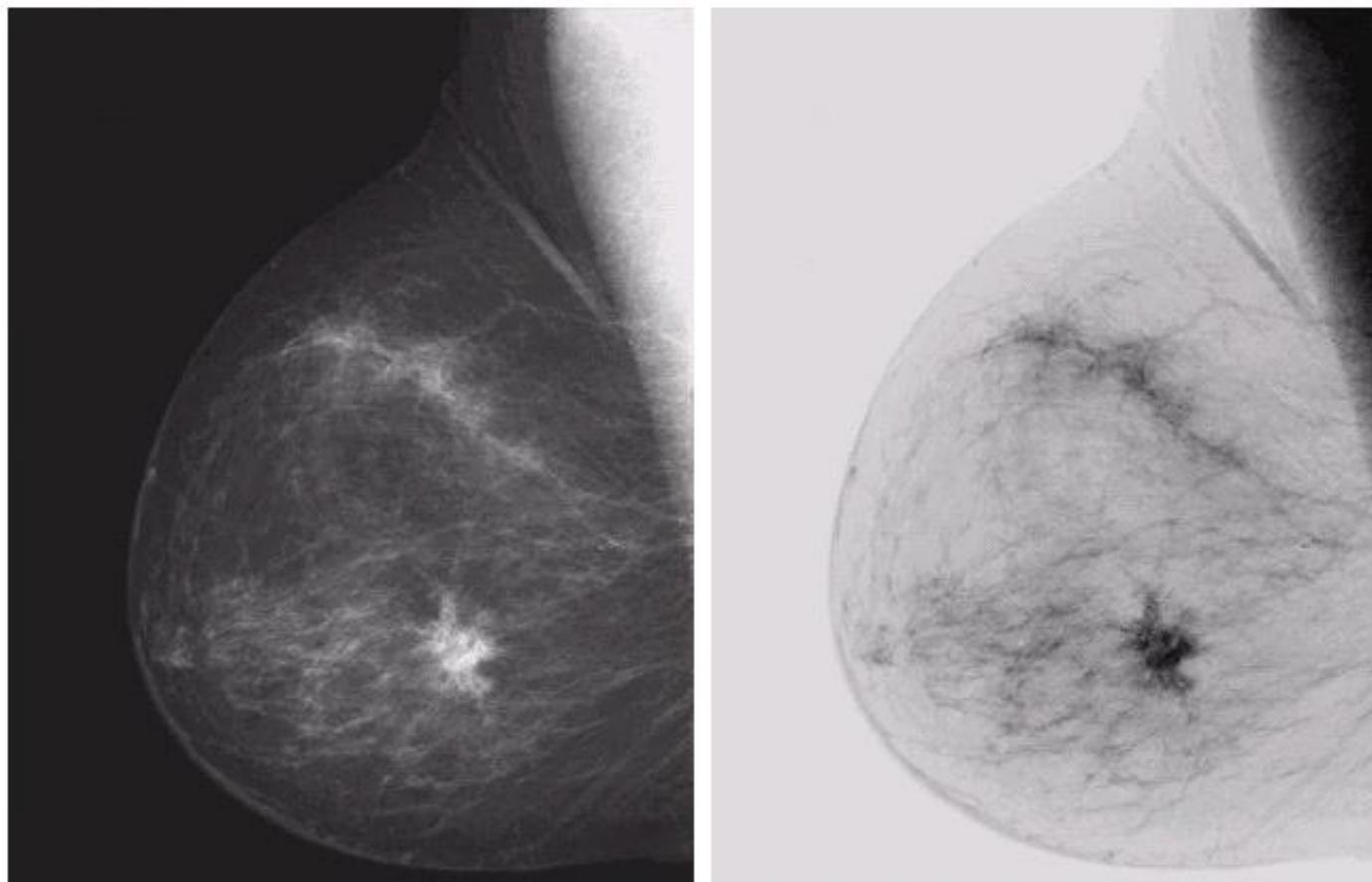
Original image



Negative image



- Sebagai proses *image enhancement*, menegatifkan citra bermanfaat bila area hitam sangat dominan di dalam citra, misalnya foto sinar-X dan citra mammografi.



a | b

FIGURE 3.4
(a) Original digital mammogram.
(b) Negative image obtained using the negative transformation in Eq. (3.2-1).
(Courtesy of G.E. Medical Systems.)



Input image (X-ray image)



Output image (negative)

- Menegatifkan citra adalah salah satu transformasi linier. Selain transformasi linier, terdapat tiga fungsi transformasi dasar keabuan:

1. Fungsi linier

- Transformasi negatif dan transformasi identitas

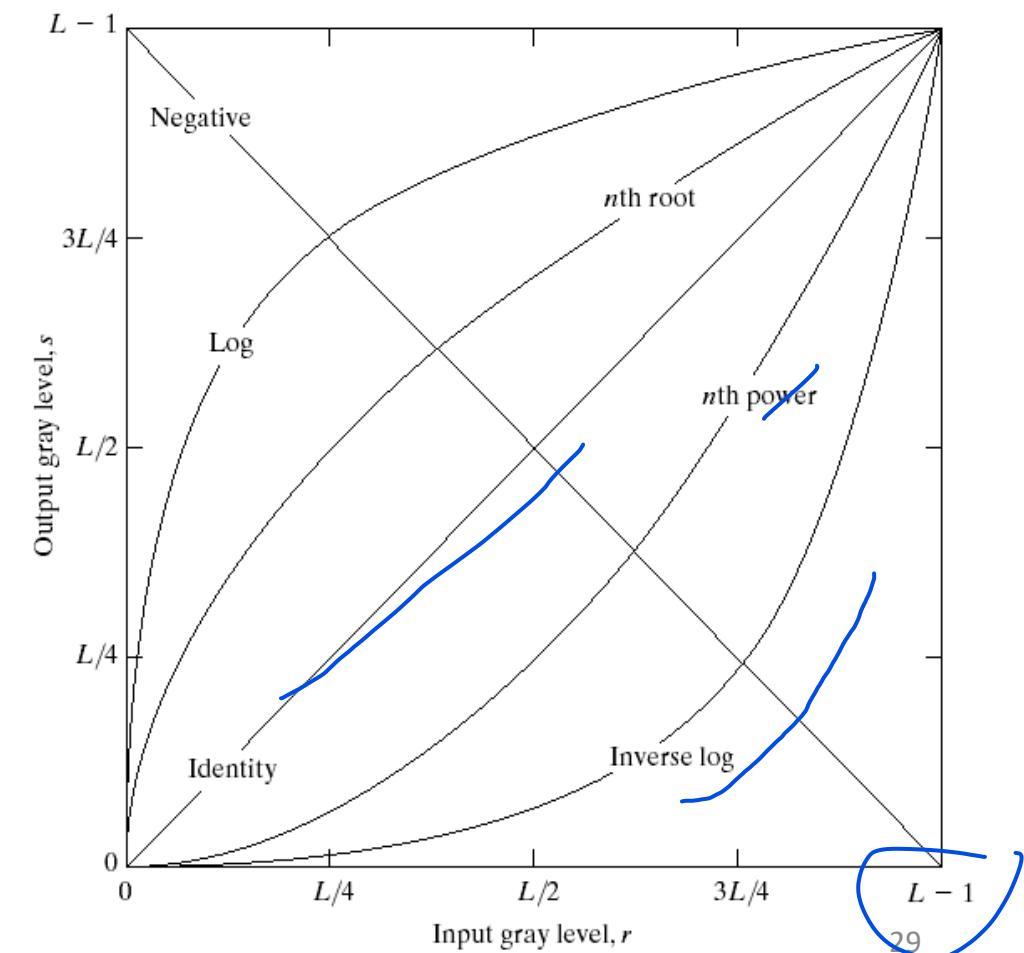
2. Fungsi logaritma

- Transformasi log dan inverse-log

3. Fungsi pangkat

- Transformasi pangkat n dan transformasi akar pangkat n

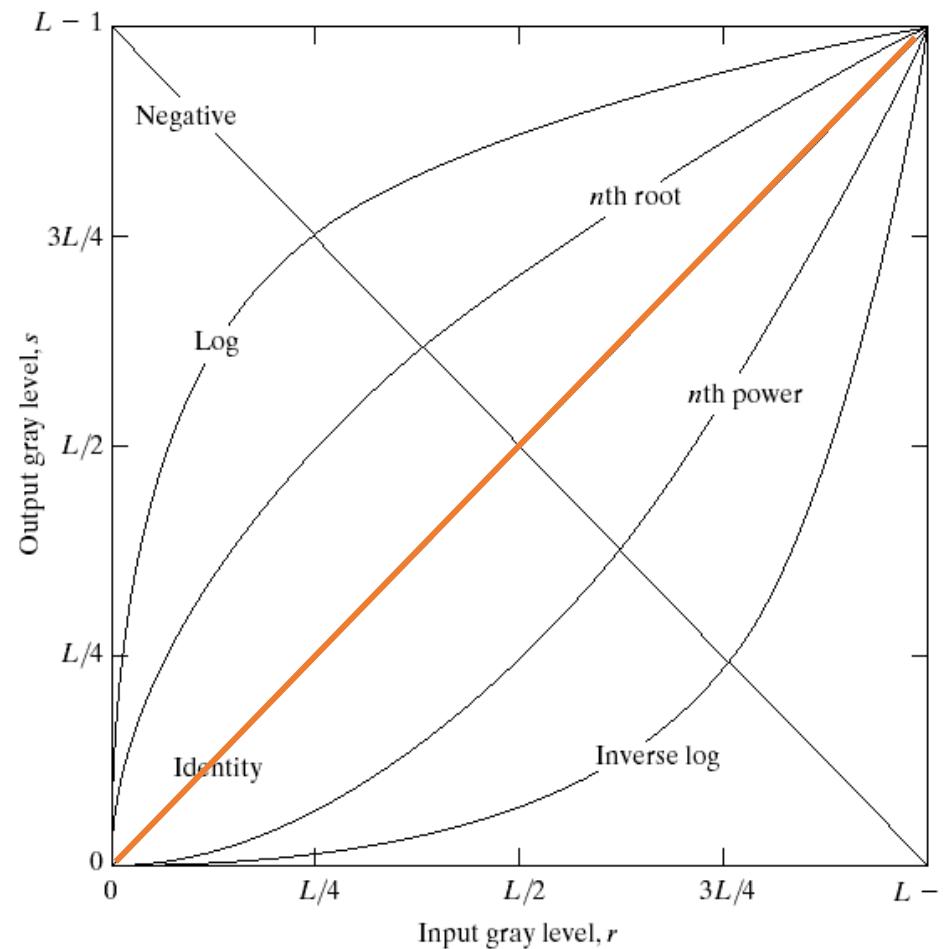
FIGURE 3.3 Some basic gray-level transformation functions used for image enhancement.



a) Transformasi identitas

- Nilai keabuan citra *output* sama dengan keabuan citra *input*
- Dimasukkan ke dalam grafik hanya untuk melengkapi

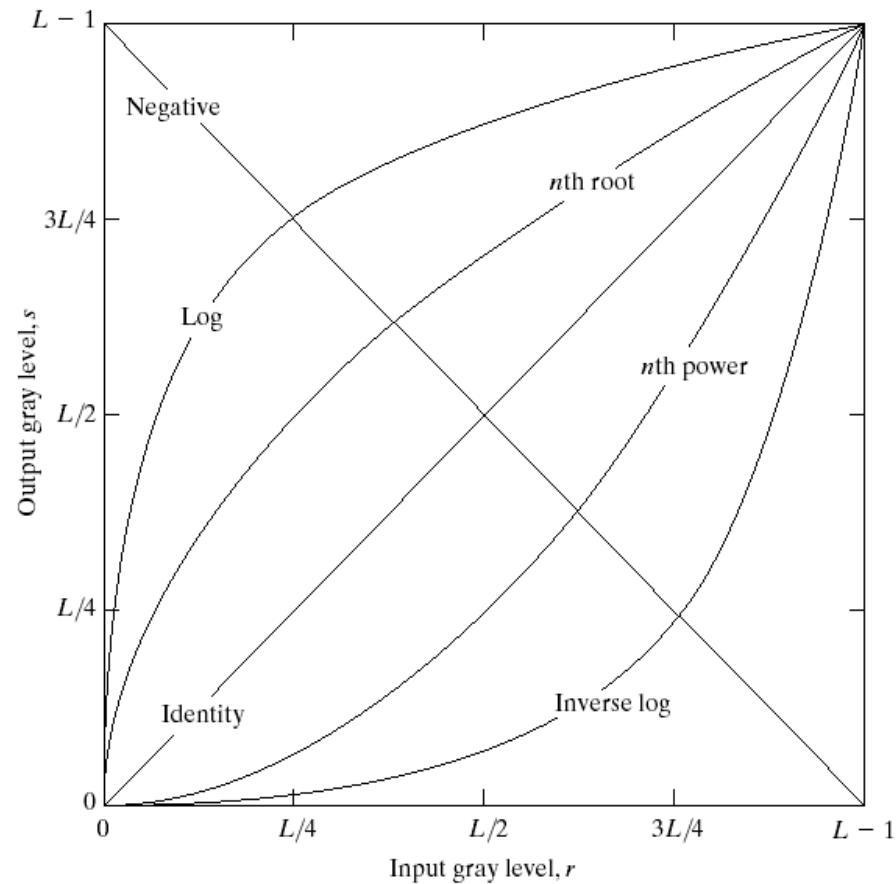
FIGURE 3.3 Some basic gray-level transformation functions used for image enhancement.

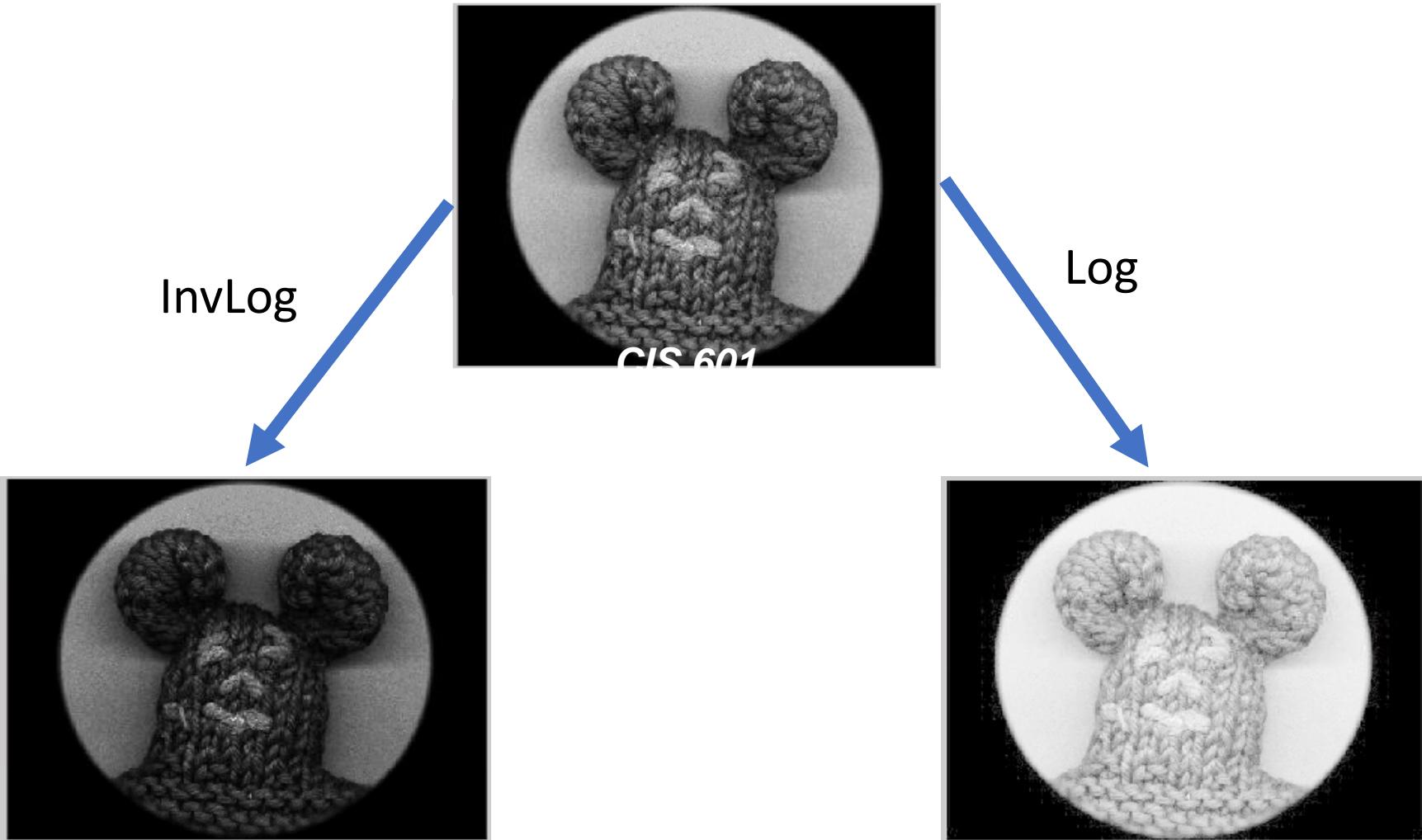


b) Transformasi Log

- Fungsi $s = c \log(1+r)$
- Transformasi log memiliki sifat:
 1. Untuk citra yang memiliki rentang yang sempit untuk nilai-nilai keabuan yang rendah (gelap), dipetakan menjadi rentang yang lebih luas pada citra luaran.
 2. Untuk citra yang memiliki rentang yang lebar untuk nilai-nilai keabuan yang tinggi (terang), dipetakan menjadi rentang yang lebih sempit pada citra luaran
- Pada transformasi log balikan (*inverse*), yang terjadi adalah kebalikannya.

FIGURE 3.3 Some basic gray-level transformation functions used for image enhancement.





Sumber gambar: CIS 601, Image ENHANCEMENT in the SPATIAL DOMAIN, Dr. Rolf Lakaemper

Transformasi Log

```
clc  
clear  
a=imread('cameraman.bmp');  
imshow(a), title ('Original image');  
a=im2double(a);  
[row,col]=size(a);  
c = 2;  
for i=1:row  
    for j=1:col  
        s(i,j)= c*log(a(i,j) + 1);  
    end  
end  
figure, imshow(s), title ('Log  
transformation image');
```

Original image



Log transformation image



Transformasi Inverse Log

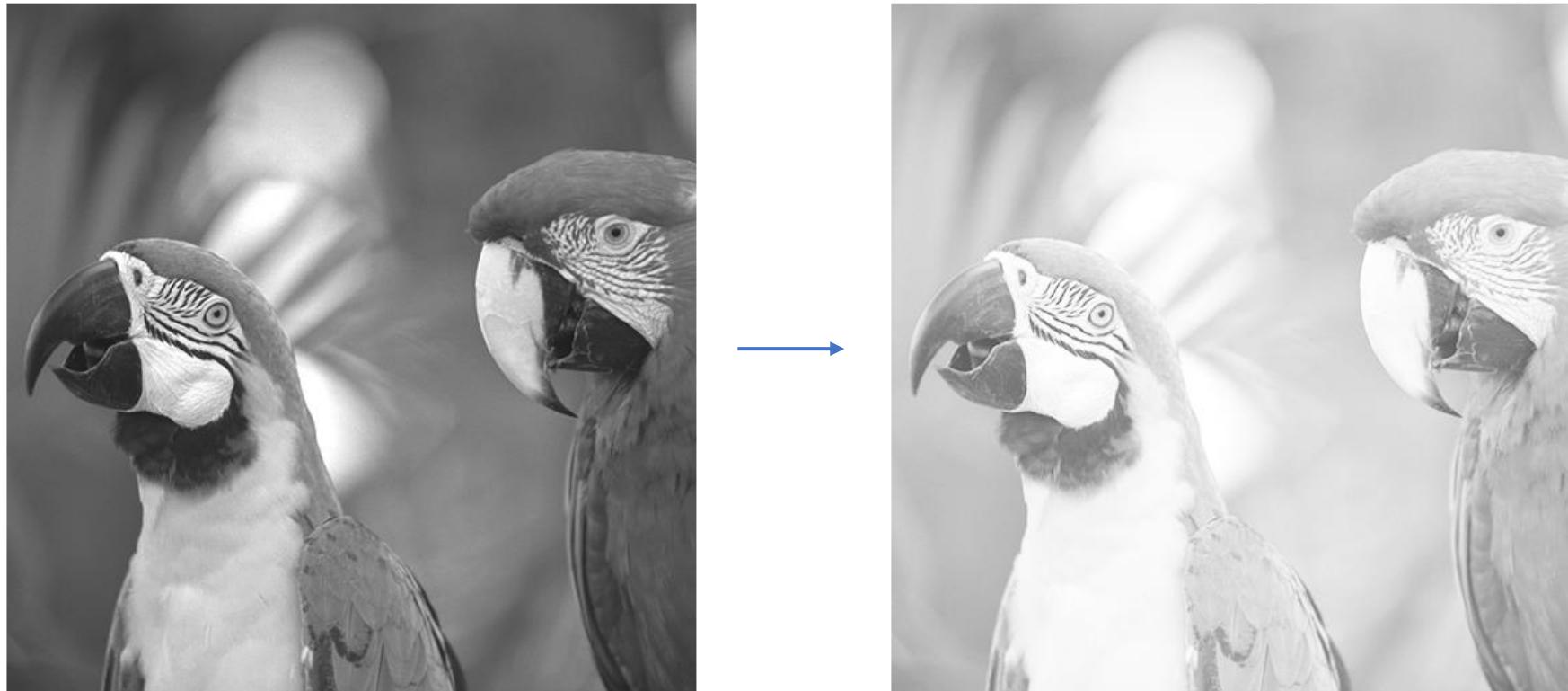
```
clc
clear
a=imread('cameraman.bmp');
imshow(a), title ('Original image');
a=im2double(a);
[row,col]=size(a);
c = 2;
for i=1:row
    for j=1:col
        s(i,j)= exp(a(i,j)^c) - 1;
    end
end
figure, imshow(s), title ('Inverse log
transformation image');
```

Original image



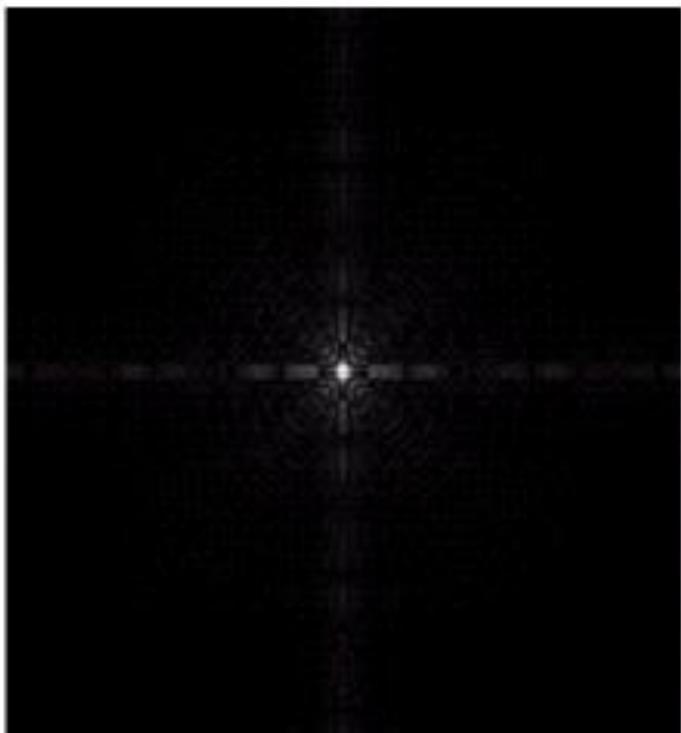
Inverse log transformation image



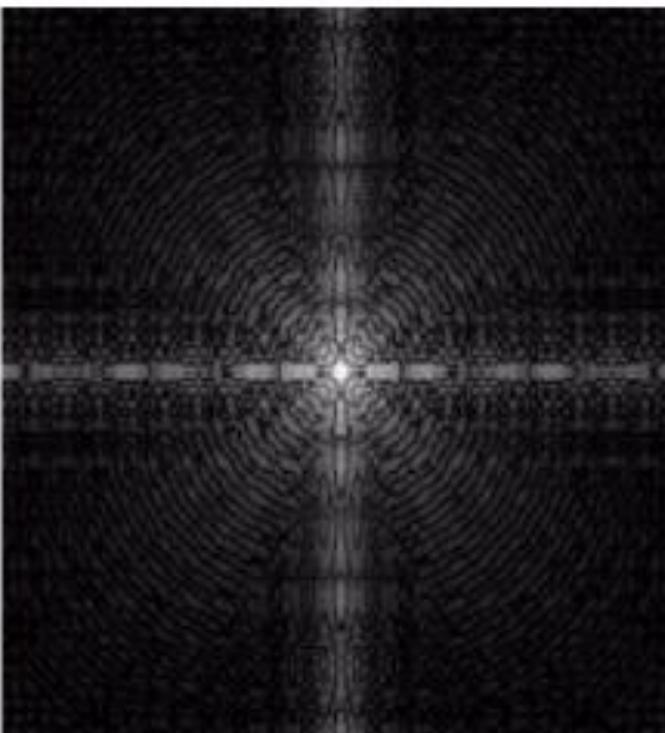


Application:

- *This transformation is suitable for the case when the dynamic range of a processed image far exceeds the capability of the display device (e.g. display of the Fourier spectrum of an image)*
- *Also called “dynamic-range compression / expansion”*



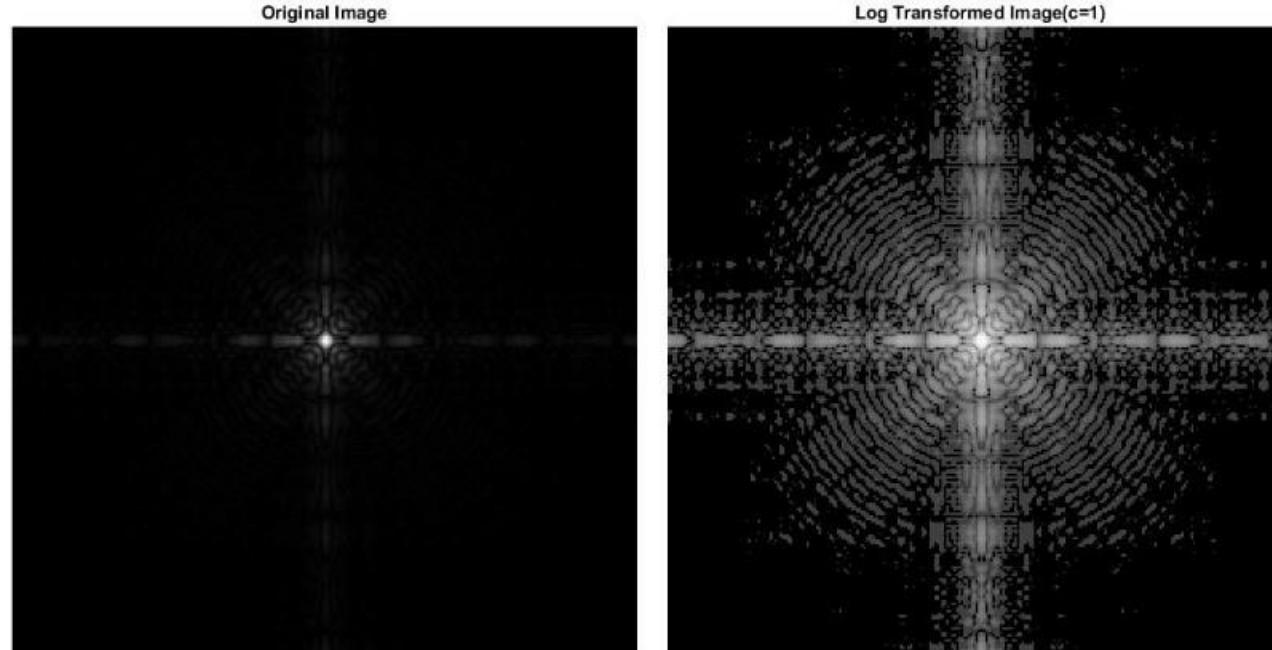
Fourier spectrum with values of range 0 to 1.5×10^6 scaled linearly



The result applying log transformation,
 $c = 1$

```
clc; clear all;

in=imread('fourier.jpg');
c=input('Enter the constant value, c = ');
a=im2double(in);
a=a*255;
out=c*log10(1+a);          % s=T(r)=clog(1+r)
out=out/max(max(out));    % Normalization
subplot(121), imshow(in), title('Original Image')
subplot(122), imshow(out), title('Log Transformed Image(c=1)')
```



c) Transformasi Pangkat

- Fungsi pangkat:

$$s = cr^\gamma$$

c dan γ adalah konstanta positif.

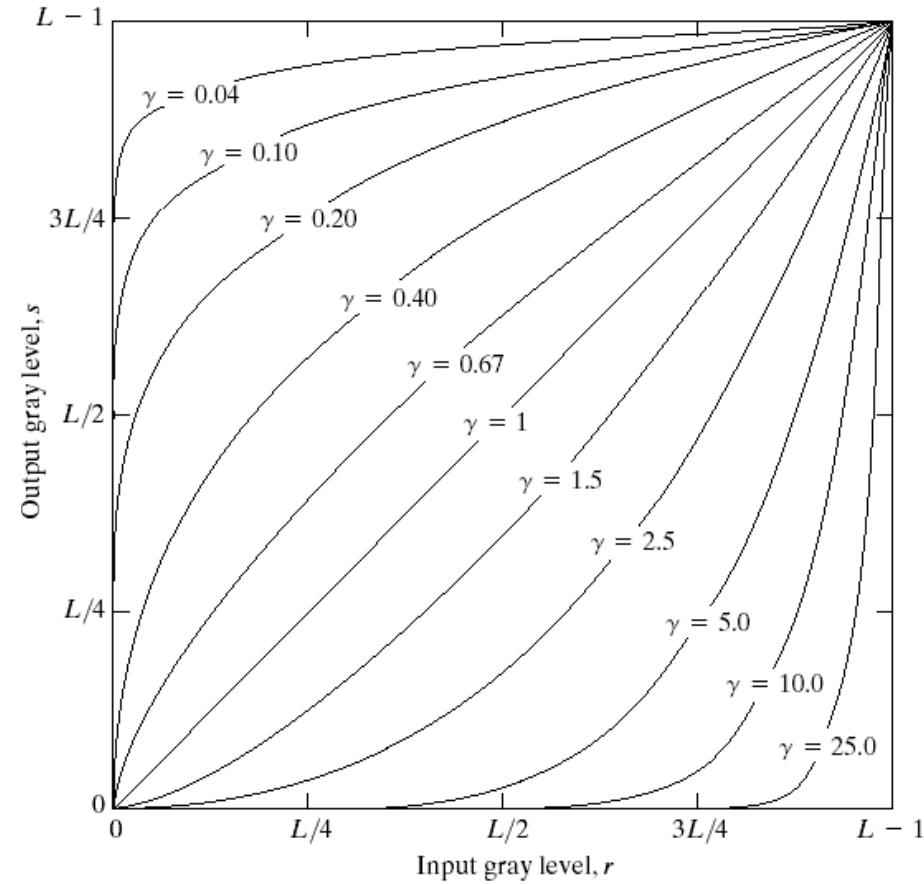
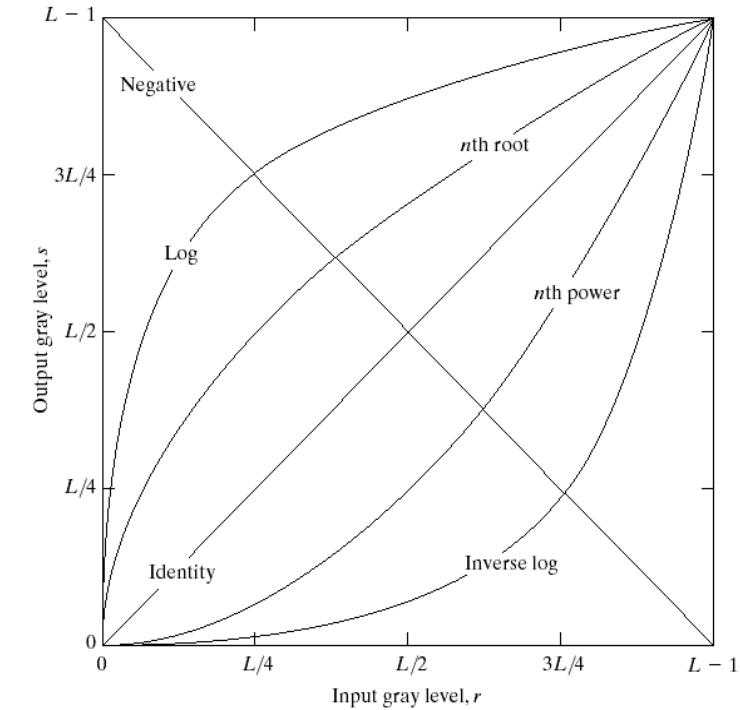


FIGURE 3.6 Plots of the equation $s = cr^\gamma$ for various values of γ ($c = 1$ in all cases).

FIGURE 3.3 Some basic gray-level transformation functions used for image enhancement.



Hukum pangkat (*power-law*):

Untuk $\gamma < 1$: Membuat citra menjadi lebih terang

Untuk $\gamma > 1$: Membuat citra menjadi lebih gelap

Jika $\gamma = 1$ & $c=1$: Transformasi identitas ($s = r$)

Beberapa devais (*image capture, printing, display*) melakukan respon berdasarkan hukum-pangkat dan perlu dikoreksi

Power transformation ($\gamma < 1$)

```
clc
clear
a=imread('cameraman.bmp');
imshow(a), title ('Original image');
a=im2double(a);
[row,col]=size(a);
gamma = 0.3;
c = 1;
for i=1:row
    for j=1:col
        s(i,j)= c *(a(i,j)^gamma);
    end
end
figure, imshow(s), title ('Power
transformation image');
```

Original image



Power transformation image



Power transformation ($\gamma > 1$)

```
clc
clear
a=imread('cameraman.bmp');
imshow(a), title ('Original image');
a=im2double(a);
[row,col]=size(a);
gamma = 3;
c = 1;
for i=1:row
    for j=1:col
        s(i,j)= c * (a(i,j)^gamma);
    end
end
figure, imshow(s), title ('Power
transformation image');
```

Original image



Power transformation image



Original image



Power transformation image



$$\gamma = 0.4$$

Citra berwarna

```
clc  
clear  
a=imread('gedung-sate.jpg');  
imshow(a), title ('Original image');  
a=im2double(a);  
gamma = 0.5; c = 1;  
s = c * a.^gamma;  
figure, imshow(s), title ('Power tranformation image');
```

Original image



Power tranformation image





MRI image of
fractured human
spine

Result of applying
power-law
transformation

$$c = 1, \gamma = 0.6$$

Result of applying
power-law
transformation

$$c = 1, \gamma = 0.4$$

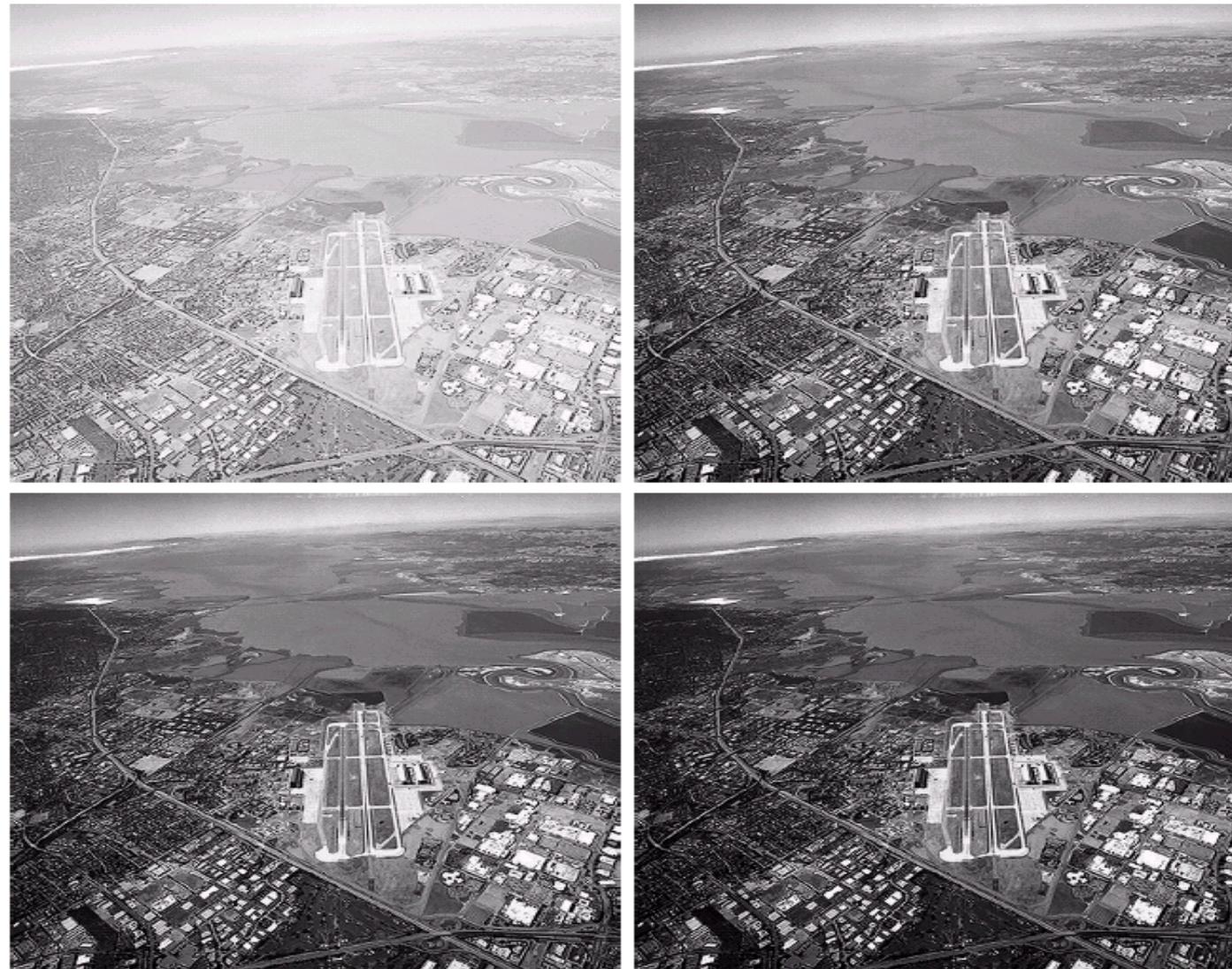
Result of applying
power-law
transformation

$$c = 1, \gamma = 0.3$$

a b
c d

FIGURE 3.9

(a) Aerial image.
(b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 3.0, 4.0$, and 5.0 , respectively. (Original image for this example courtesy of NASA.)



Rincian:



Arial image



Result of a transformation
for $c=1$ and $\gamma=3$

Rincian:



Result of a transformation
for $c=1$ and $\gamma=4$



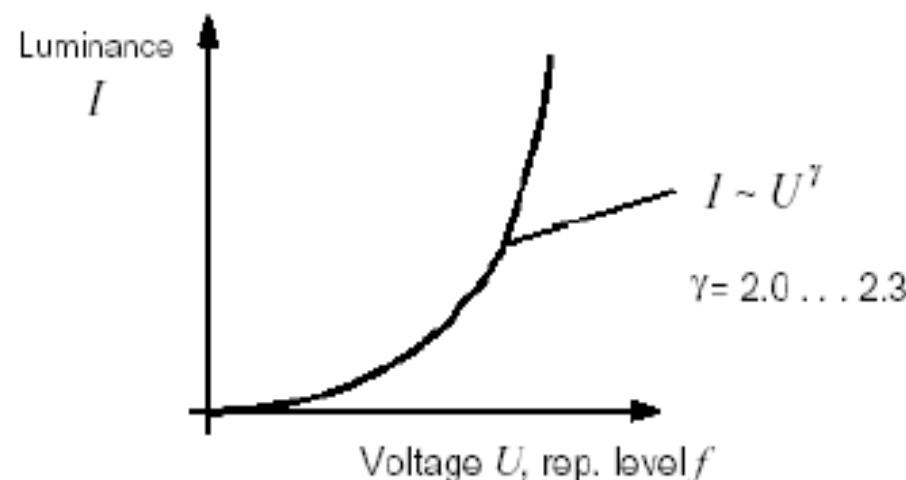
Result of a transformation
for $c=1$ and $\gamma=5$

Gamma correction

Gamma (γ) correction: Proses yang digunakan untuk mengoreksi fenomena hukum-pangkat

- Example of gamma correction

Cathode ray tubes (CRT) are nonlinear



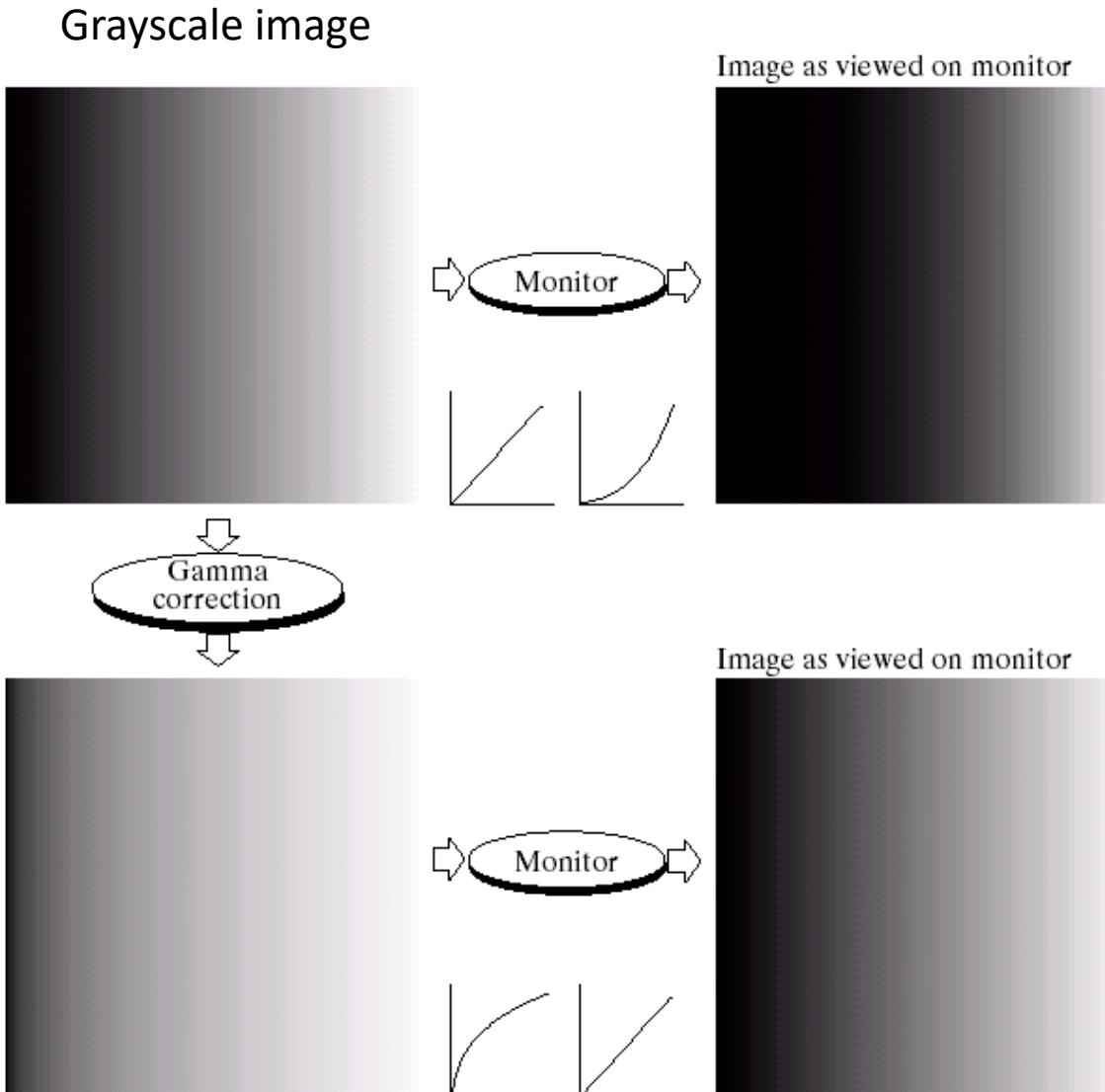
- To linearize the CRT response a pre-distortion circuit is needed

$$s = cr^{1/\gamma}$$

Gamma correction

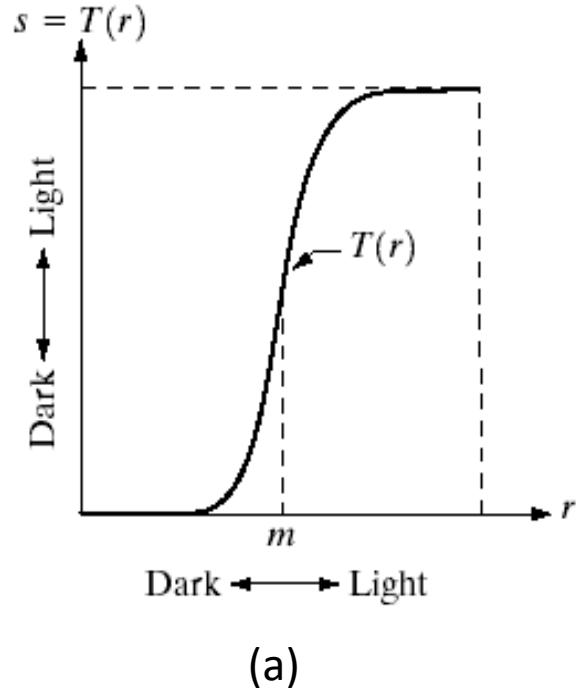
a
b
c
d

FIGURE 3.7
(a) Linear-wedge gray-scale image.
(b) Response of monitor to linear wedge.
(c) Gamma-corrected wedge.
(d) Output of monitor.



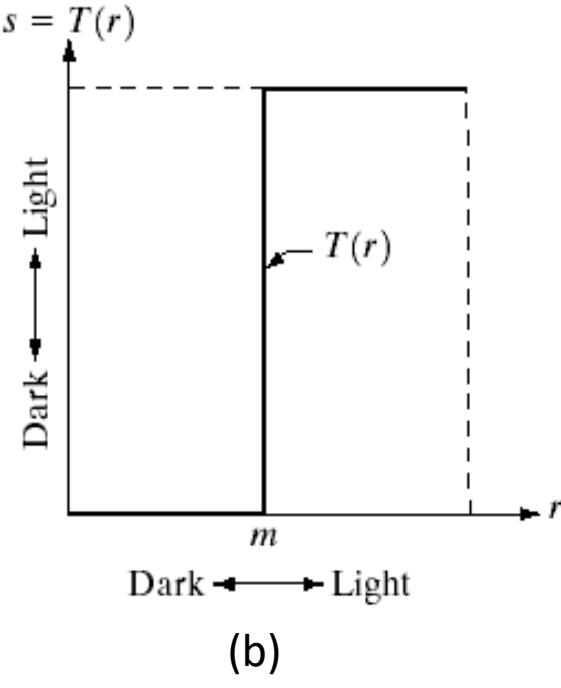
- Cathode ray tube (CRT) devices have an intensity-to-voltage response that is a power function, with γ varying from 1.8 to 2.5
- The picture will become darker.
- Gamma correction is done by preprocessing the image before inputting it to the monitor with $s = cr^{1/\gamma}$

3. Perbaikan kontras (*contrast enhancement*)



Contrast Stretching

- (a) Nilai-nilai pixel $< m$ dibuat lebih gelap
Nilai-nilai pixel $\geq m$ dibuat lebih terang
- Operasi peregangan kontras (*contrast stretching*)



Thresholding

- (b) Nilai-nilai pixel $< m$ dibuat menjadi hitam
Nilai-nilai pixel $\geq m$ dibuat menjadi putih
- Operasi pengambangan (*thresholding*)

- r = graylevel citra masukan
- s = graylevel citra luaran
- T = fungsi perbaikan kontras
- m = nilai ambang



Original image

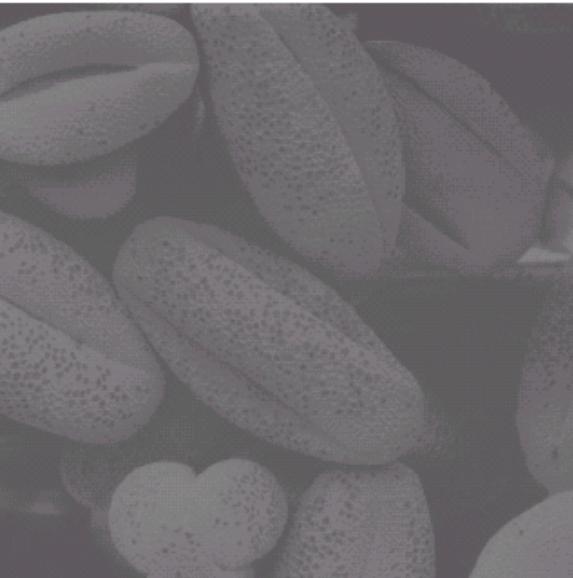
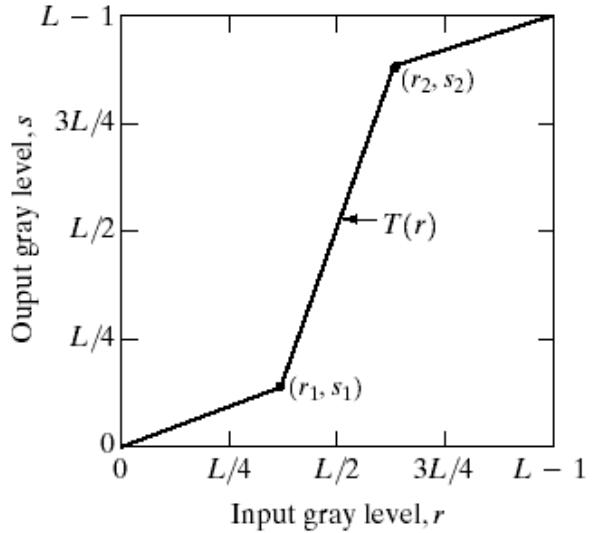


Peregangan kontras



Pengambangan

Contoh:



a b
c d

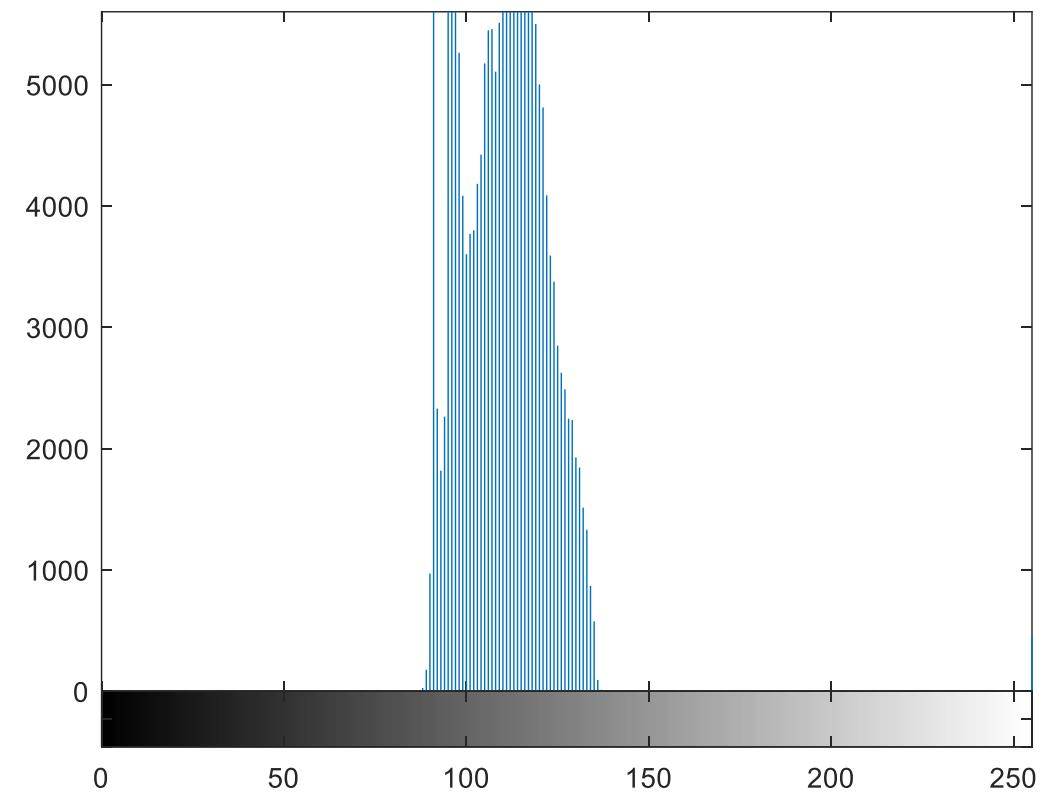
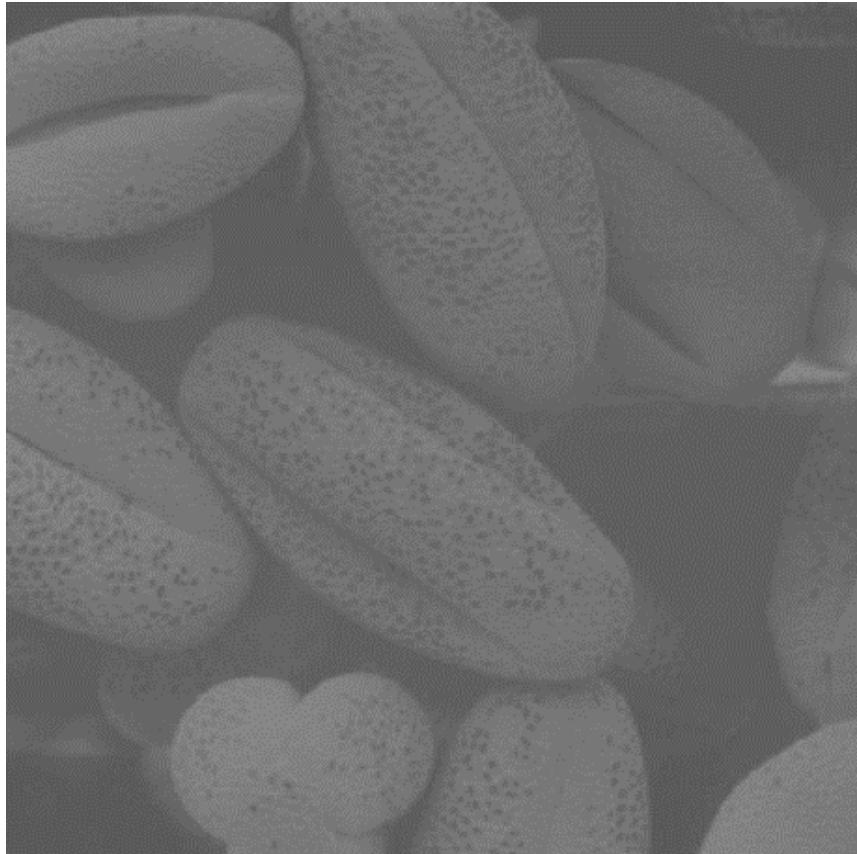
FIGURE 3.10

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.)

Jika $r_1 = r_2 = m$, maka hasilnya sama dengan operasi pengambangan, menghasilkan citra biner, seperti gambar d

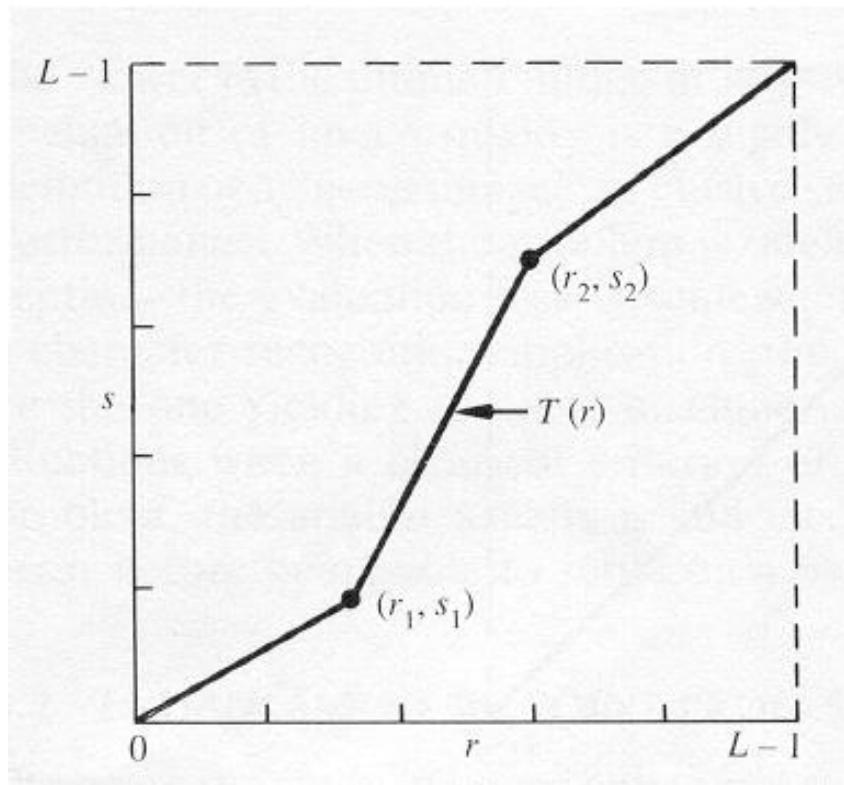
Peregangan kontras (*contrast stretching*)

- Peregangan kontras merupakan metode sederhana untuk memperbaiki citra yang memiliki kontras rendah



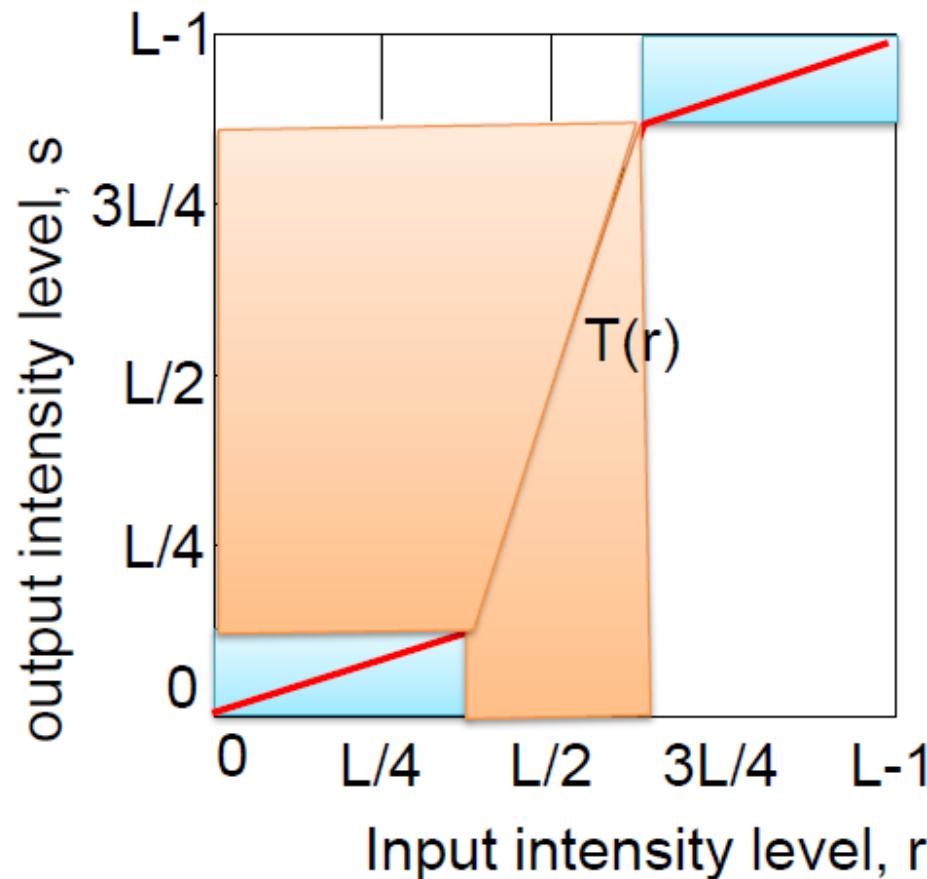
Ciri-ciri citra kontras-rendah: histogram sempit menumpuk di tengah

- Tujuan peregangan kontras: meningkatkan rentang nilai-nilai keabuan untuk citra kontras-rendah (terentang dari nilai r_1 sampai r_2 pada citra dengan nilai keabuan 0 sampai $L - 1$)



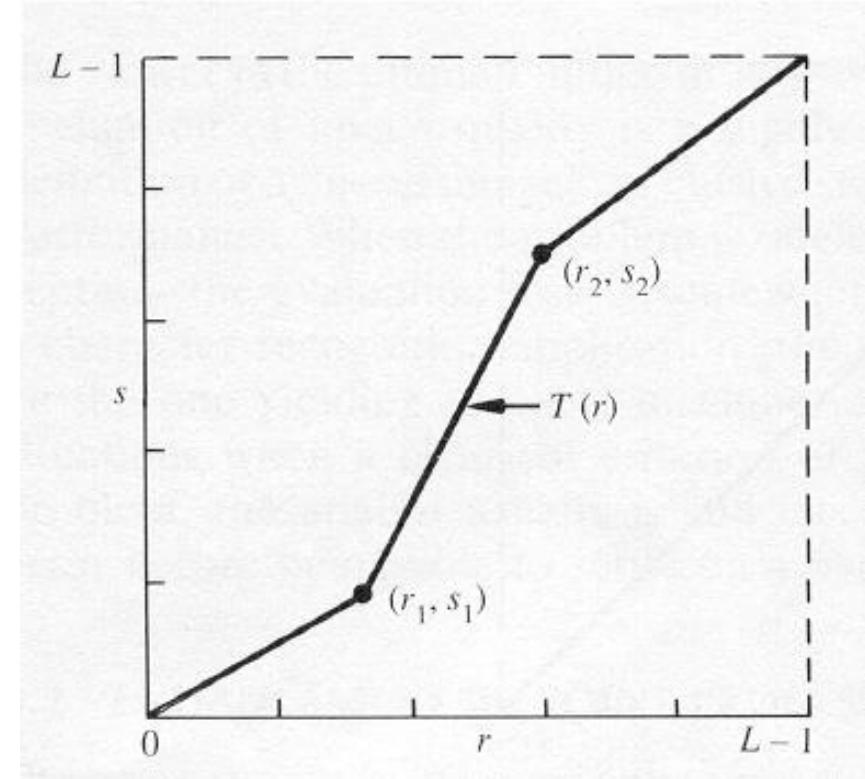
- Citra kontras-rendah dihasilkan dari
 - pencahayaan yang kurang
 - kekurangan pada rentang dinamis di dalam *imaging sensor*
 - kesalahan *setting lensa* selama akuisisi gambar
- Nilai-nilai pixel antara r_1 sampai r_2 akan dipetakan menjadi nilai antara s_1 sampai s_2

- Selang $[r_1, r_2]$ yang sempit diregang menjadi selang $[s_1, s_2]$ yang lebih lebar

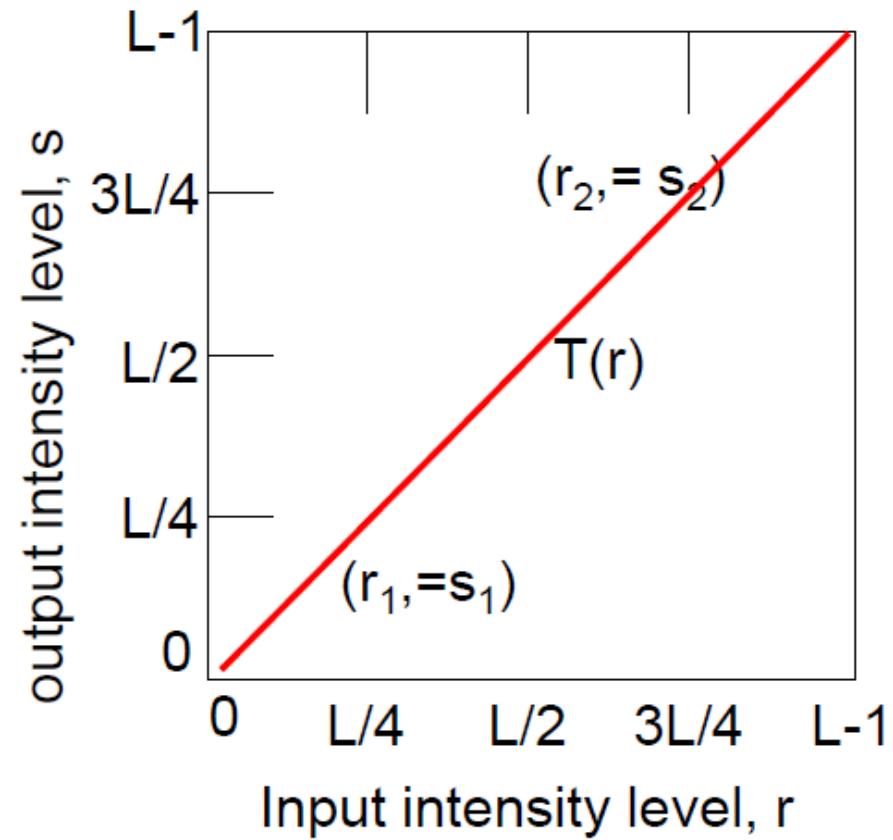
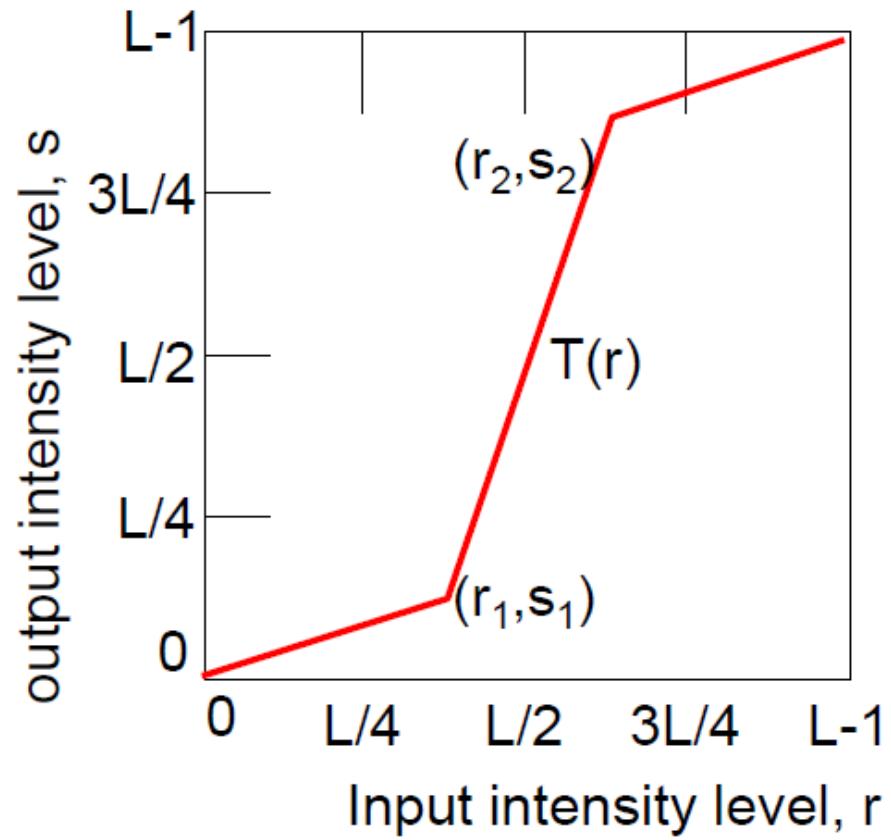


- Nilai-nilai di antara (r_1, s_1) and (r_2, s_2) menghasilkan penyebaran nilai keabuan citra luaran.

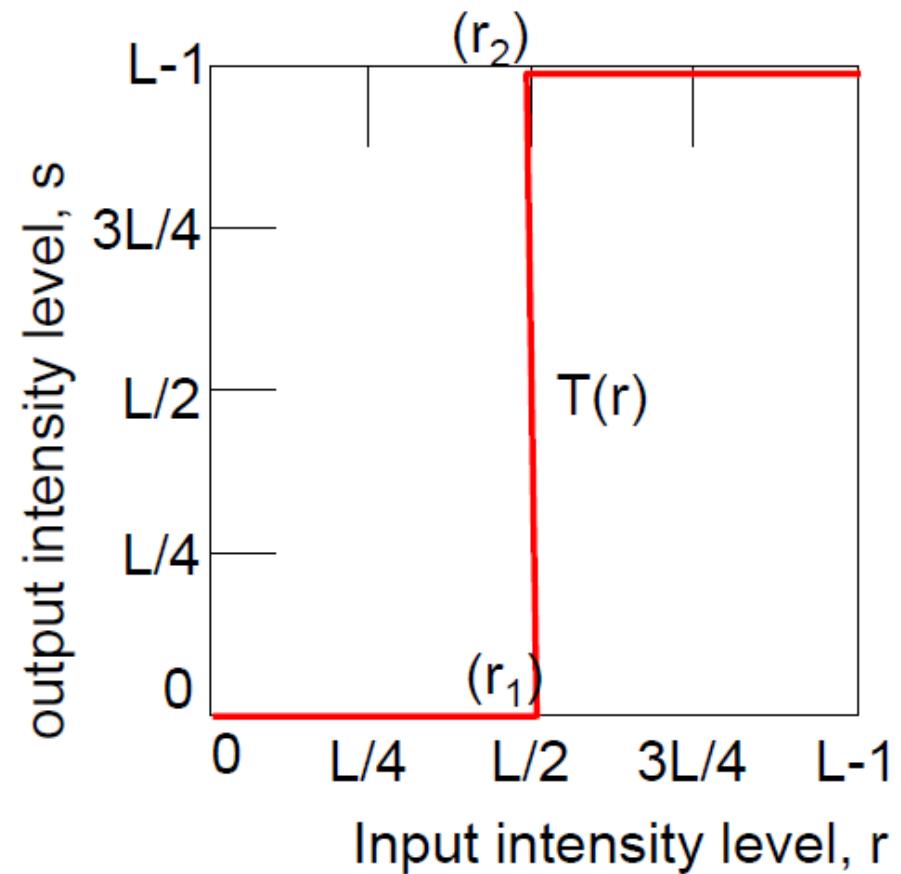
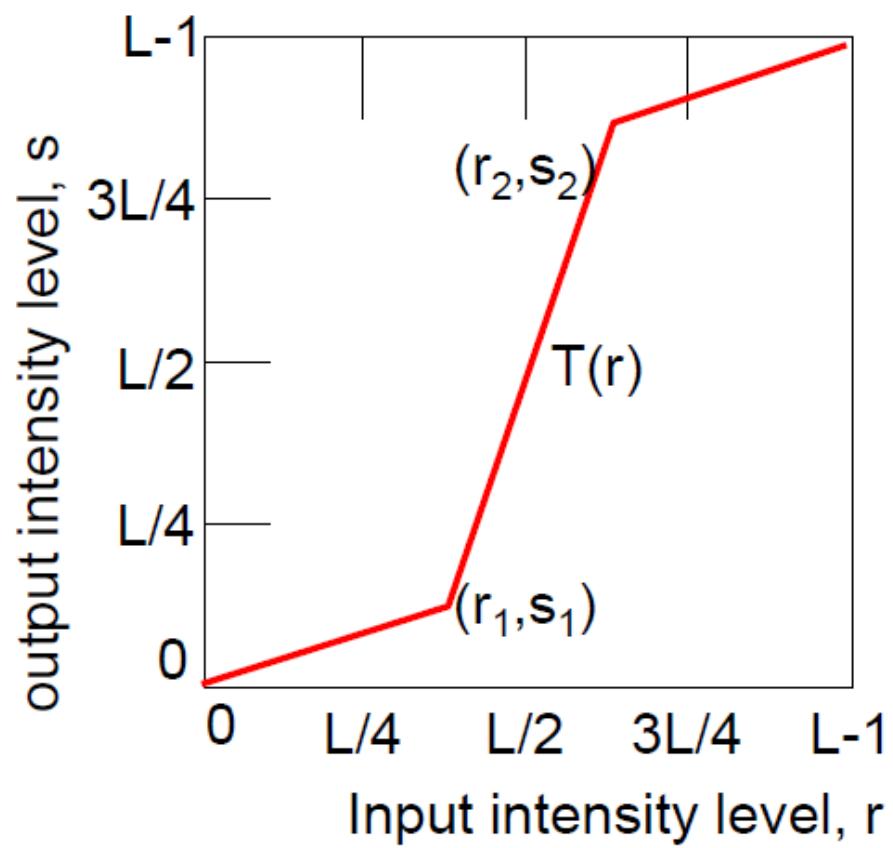
- Lokasi (r_1, s_1) dan (r_2, s_2) menentukan bentuk fungsi transformasi.
 - Jika $r_1 = s_1$ dan $r_2 = s_2$ maka transformasi adalah fungsi linier sehingga tidak menghasilkan perubahan.
 - Jika $r_1 = r_2$, $s_1 = 0$ dan $s_2 = L-1$, transformasi menjadi fungsi pengambangan yang menghasilkan citra biner.
 - Nilai-nilai di antara (r_1, s_1) and (r_2, s_2) menghasilkan penyebaran nilai keabuan citra luaran.
 - Umumnya diasumsikan $r_1 \leq r_2$ dan $s_1 \leq s_2$



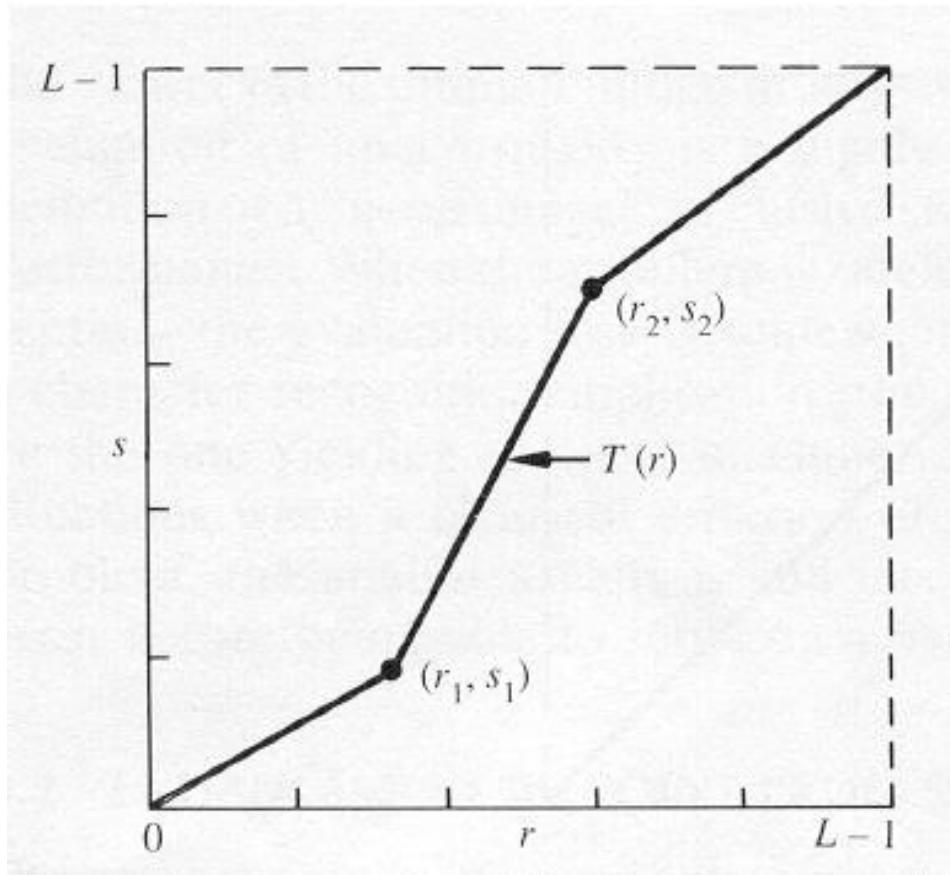
Jika $r_1 = s_1$ dan $r_2 = s_2$



Jika $r_1=r_2$, $s_1=0$ dan $s_2=L - 1$

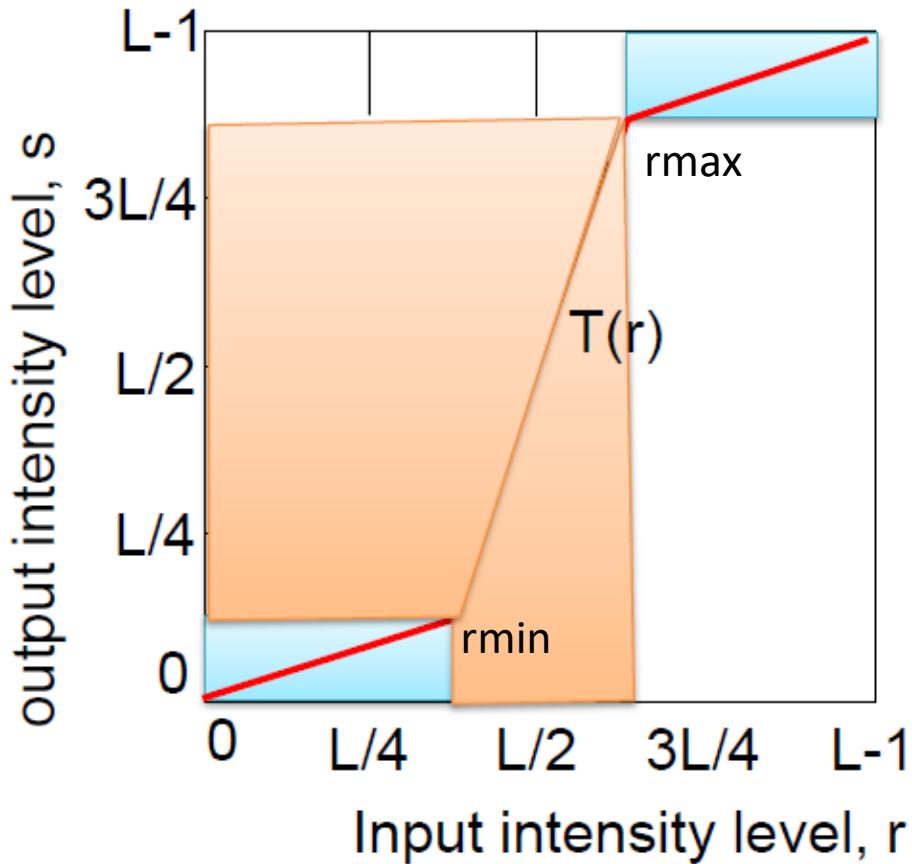


- Bagaimana menentukan r_1 dan r_2 ?



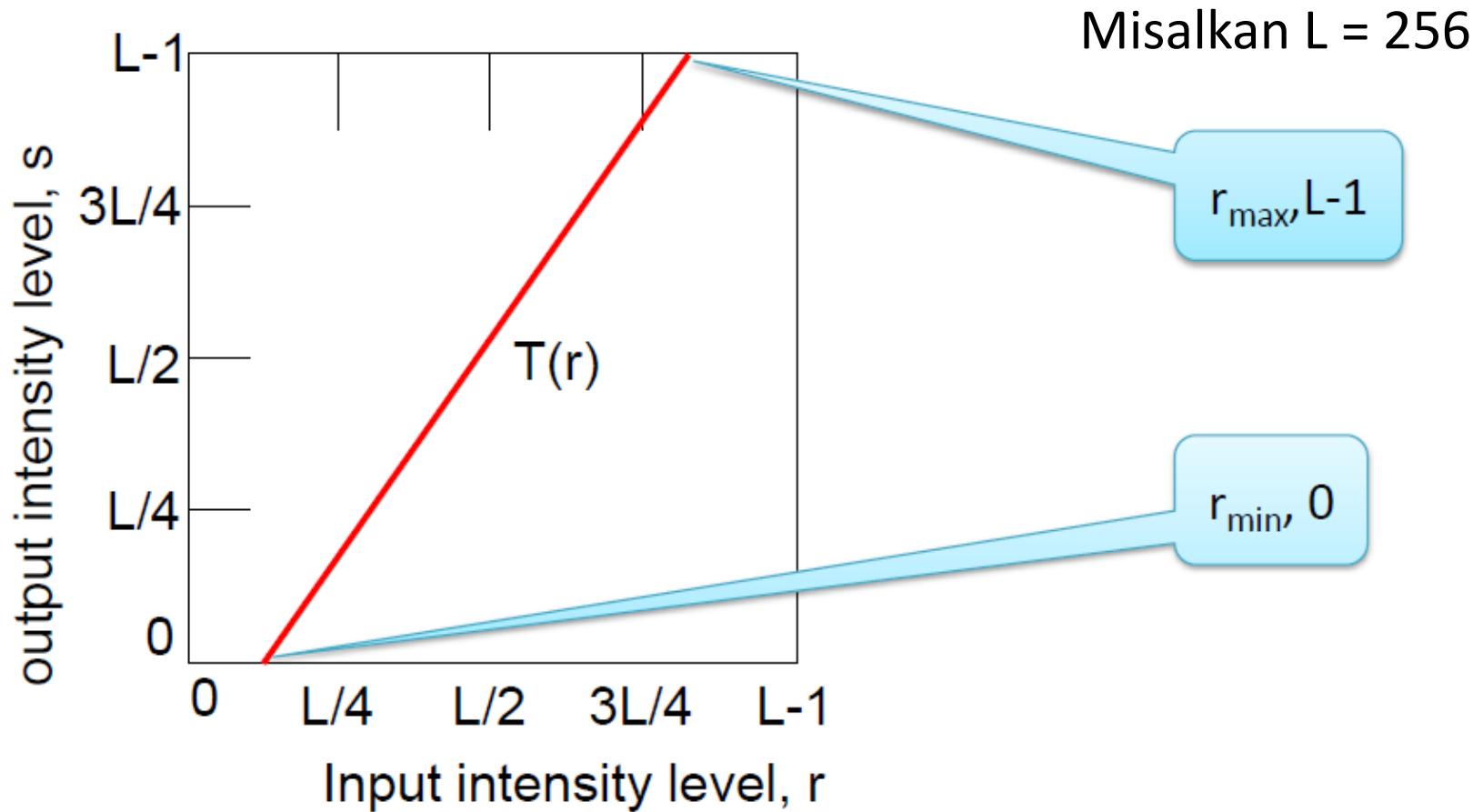
Salah satu pendekatan:

- pindai histogram citra, atau pindai pixel-pixel di dalam citra
- cari pixel bernilai minimum, misalkan r_{min}
- cari pixel bernilai maksimum, misalkan r_{max}
- pixel-pixel di bawah r_{min} diset 0
- Pixel-pixel di atas r_{max} diset $L - 1$
- $r_1 = r_{min}$, $r_2 = r_{max}$
- tentukan persamaan garis yang menghubungkan titik $(r_{min}, 0)$ dan $(r_{max}, L - 1)$ dengan persamaan umum garis lurus yang melalui titik (x_1, y_1) dan (x_2, y_2) :
$$\frac{y - y_1}{y_2 - y_1} = \frac{x - x_1}{x_2 - x_1}$$
- Petakan nilai keabuan yang lain di antara $(r_{min}, 0)$ dan $(r_{max}, L - 1)$ dengan menggunakan persamaan tersebut

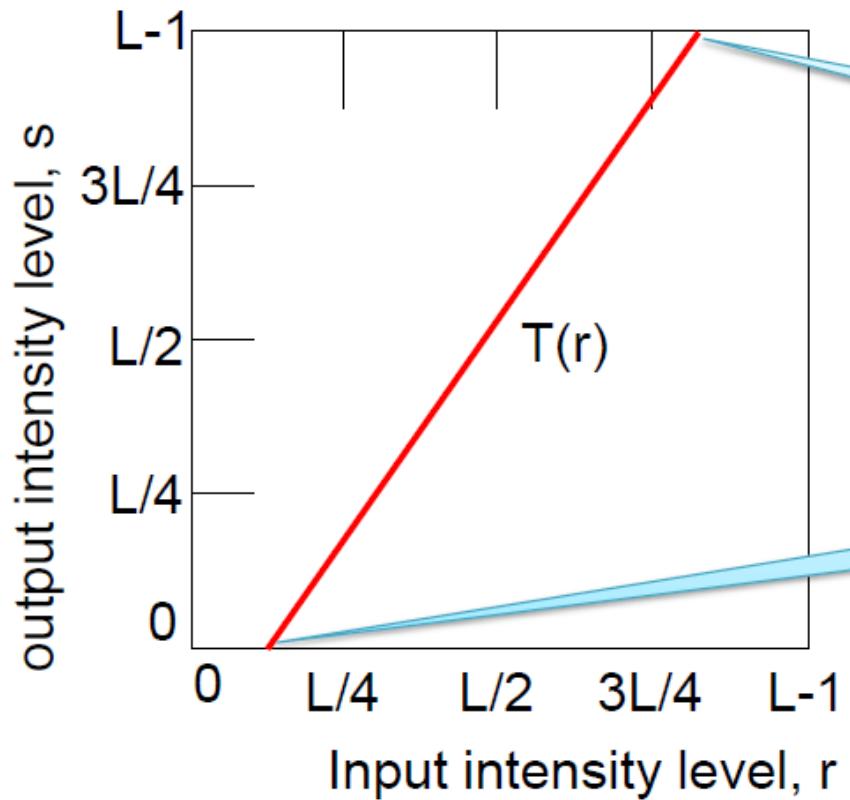


- pixel-pixel di bawah r_{min} diset 0
- Pixel-pixel di atas r_{max} diset $L - 1$

Contoh: $(r_1, s_1) = (r_{\min}, 0)$ and $(r_2, s_2) = (r_{\max}, L-1)$

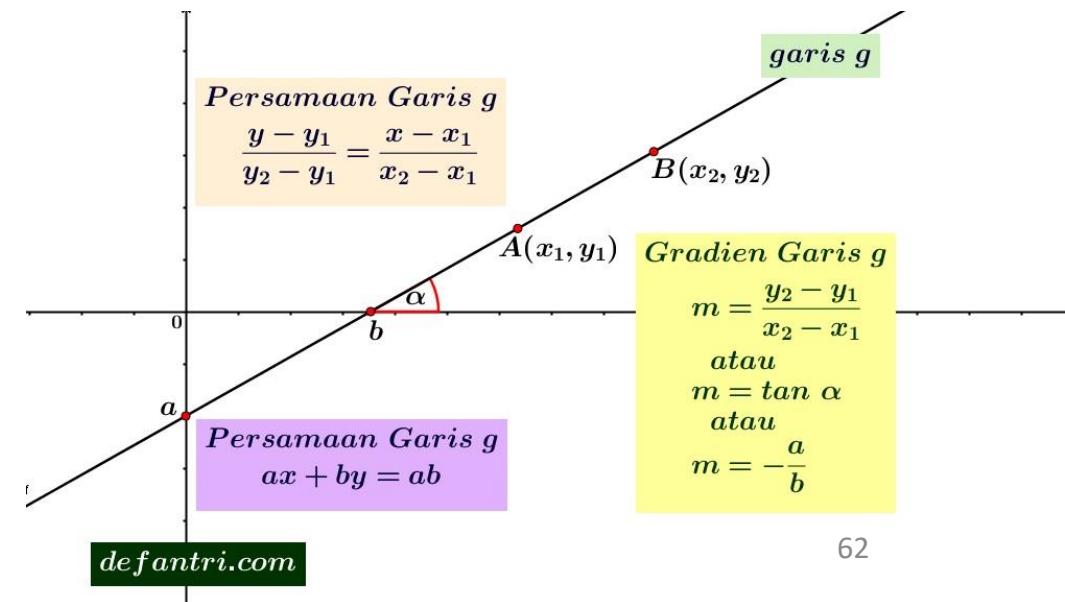


$(r_1, s_1) = (r_{\min}, 0)$ and $(r_2, s_2) = (r_{\max}, L-1)$



Persamaan garis yang melalui $(r_{\min}, 0)$ dan $(r_{\max}, 255)$:

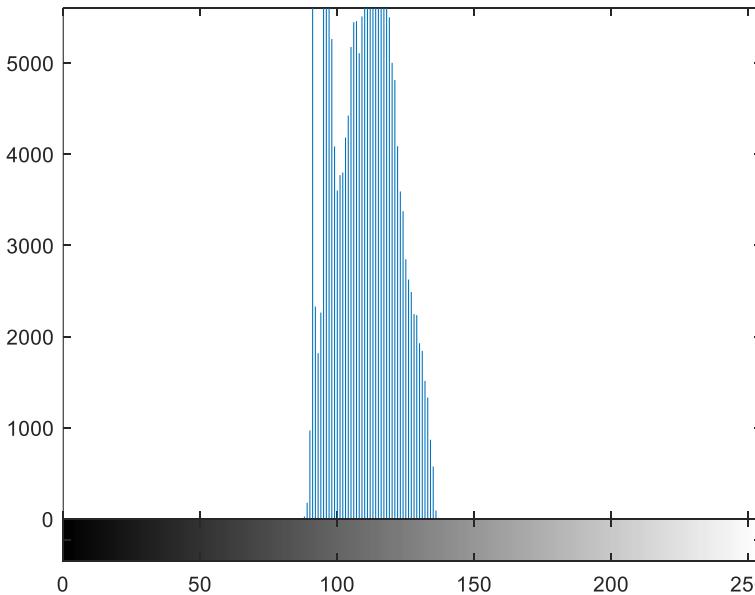
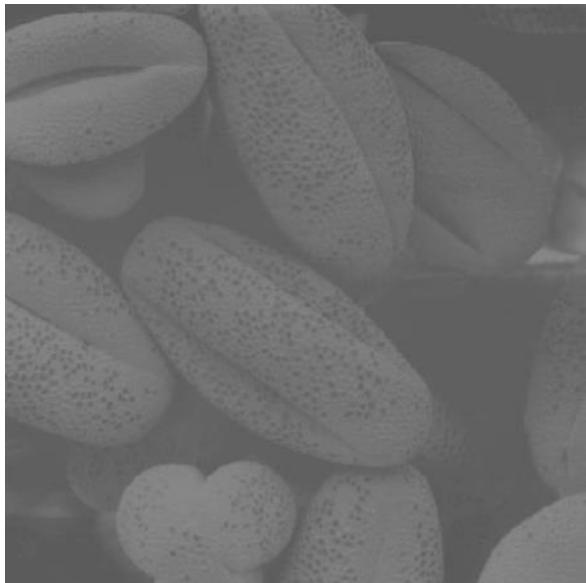
$$\frac{s-0}{255-0} = \frac{r-r_{\min}}{r_{\max}-r_{\min}} \rightarrow s = 255 \frac{(r-r_{\min})}{(r_{\max}-r_{\min})}$$



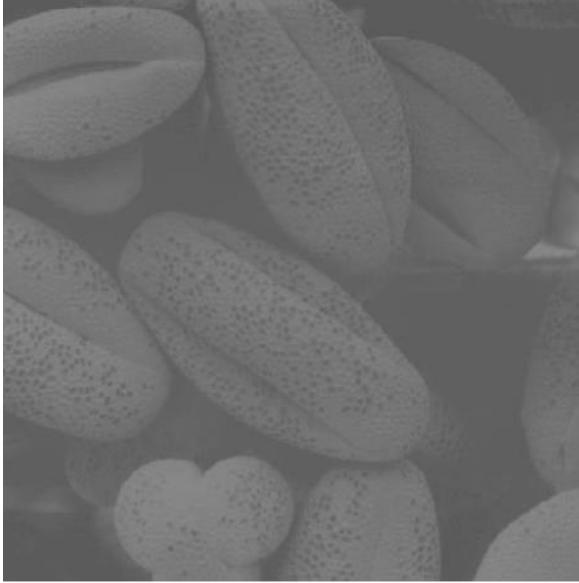
```

clear all;
clc;
I = imread('image1.bmp'); %read the image
rmin = 87;           % find the min. value of pixel in the image
rmax = 135;           % find the max. value of pixel in the image
I_new = (I - rmin).* (255/(rmax - rmin)); % transform the image
figure,imshow(I);      % display original image
figure,imhist(I);      % display histogram of original image
figure,imshow(I_new);    % display transformed image
figure,imhist(I_new);    % display histogram of transformed image

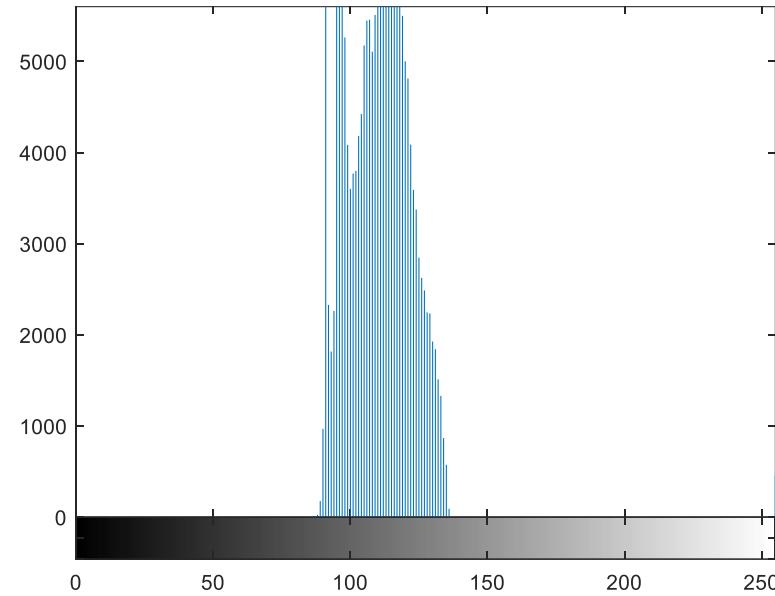
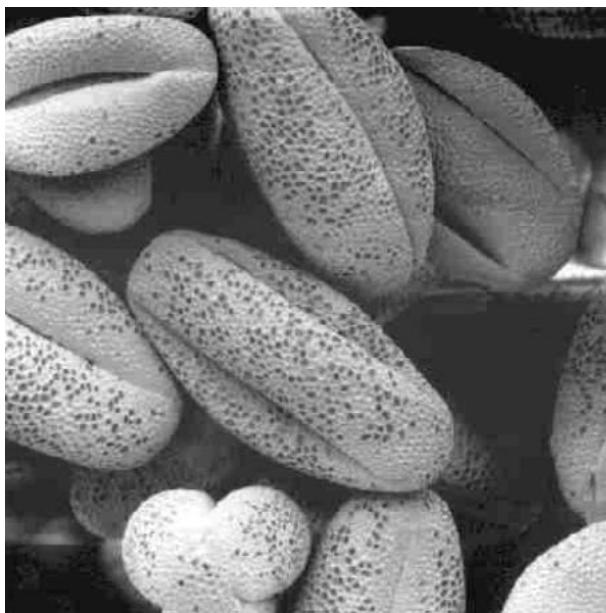
```



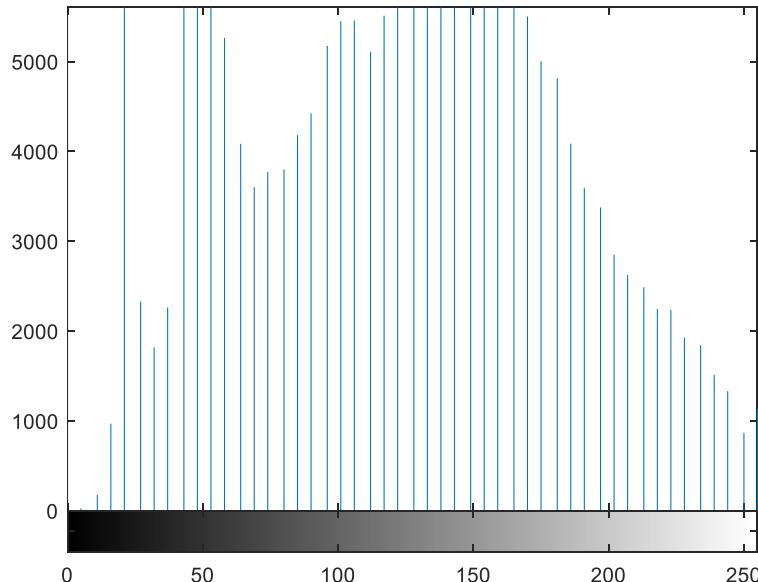
Sebelum

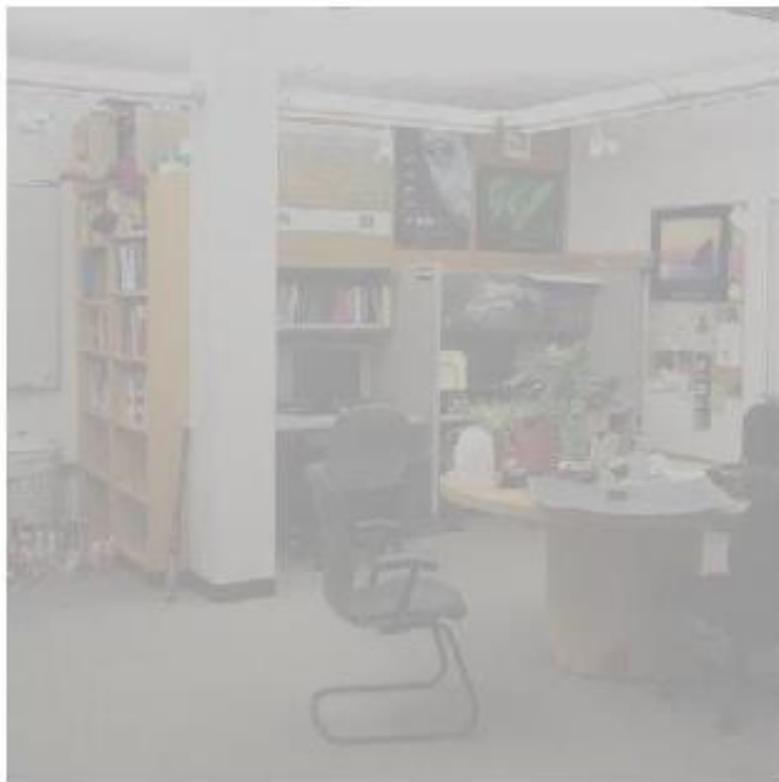


Sesudah

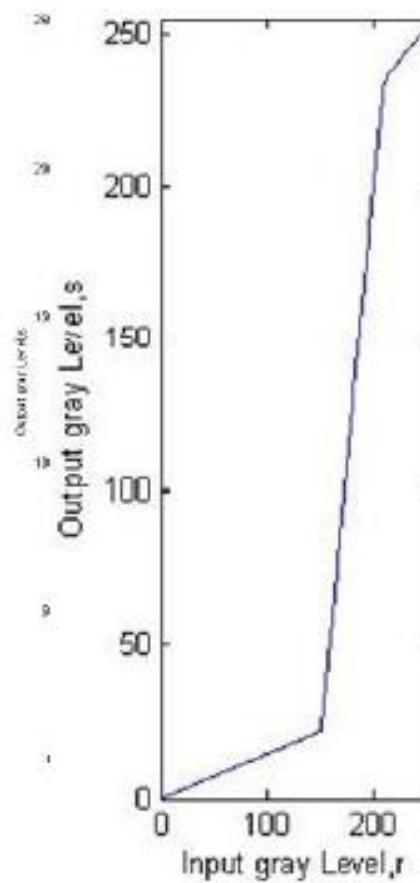


$$r_{\min} = 87$$
$$r_{\max} = 135$$





Original Image

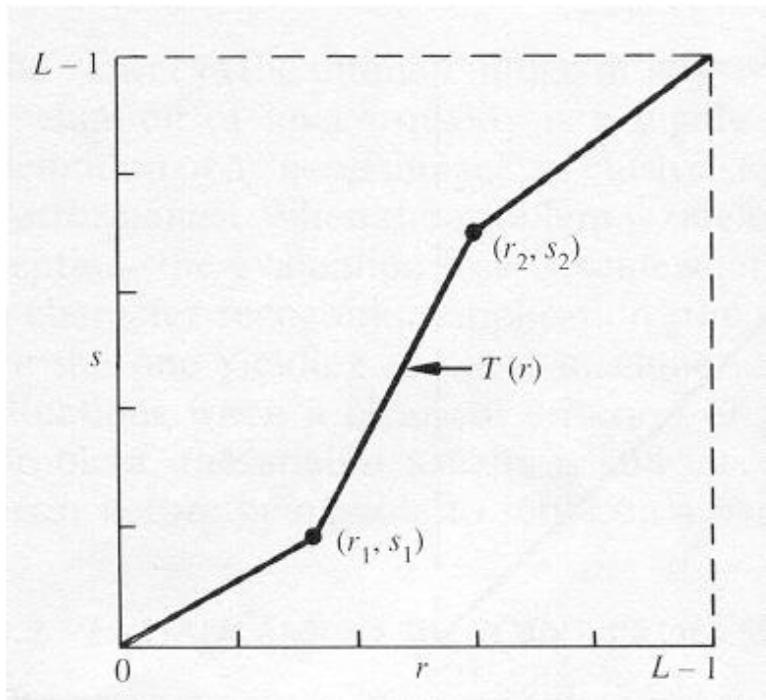


Enhanced Image

Sumber gambar: Ehsan Khoramshahi,
Image enhancement in spatial domain

Piece-wise linier transformation function

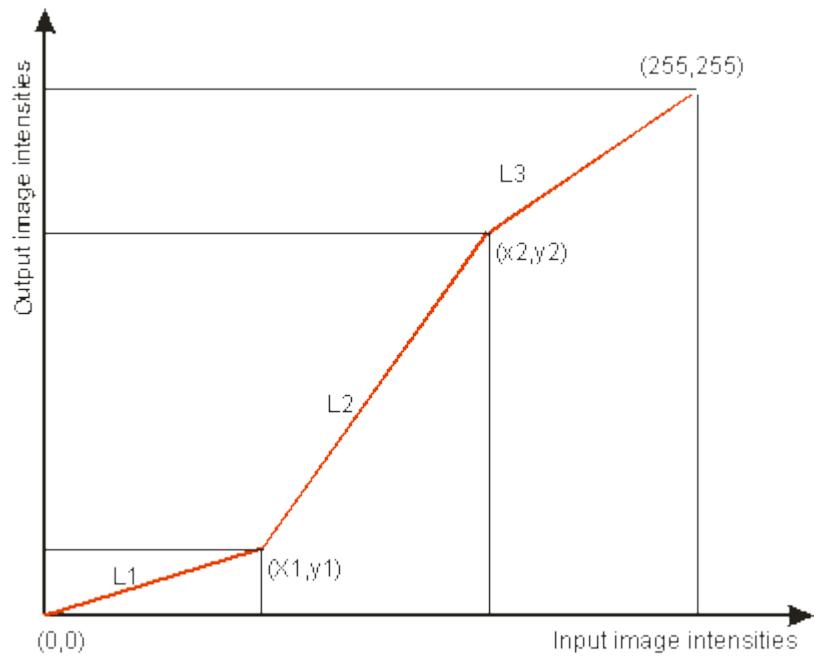
- Peregangan kontras termasuk ke dalam fungsi transformasi sepotong-sepotong (*piece-wise linier transformation function*)



Fungsi transformasi linier sepotong-sepotong:

1. *Contrast stretching*
2. *Gray-level slicing*
3. *Bit-plane slicing*

- Alternatif lain fungsi transformasi sepotong-sepotong:



$$y = \begin{cases} \frac{y_1}{x_1}x, & 0 \leq x \leq x_1 \\ \frac{y_2 - y_1}{x_2 - x_1}x + y_1, & x_1 < x < x_2 \\ \frac{255 - y_2}{255 - x_2}x + y_2, & x_2 < x < 255 \end{cases}$$

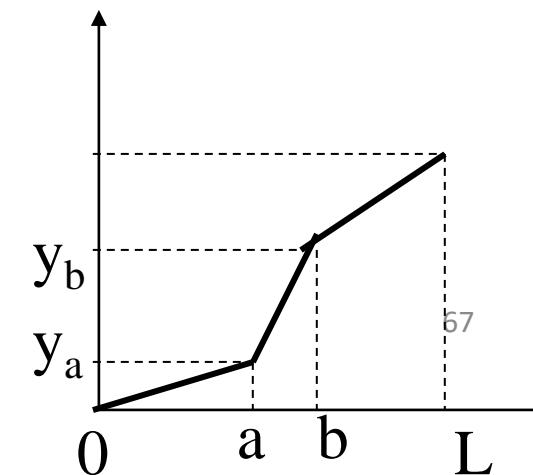
↓

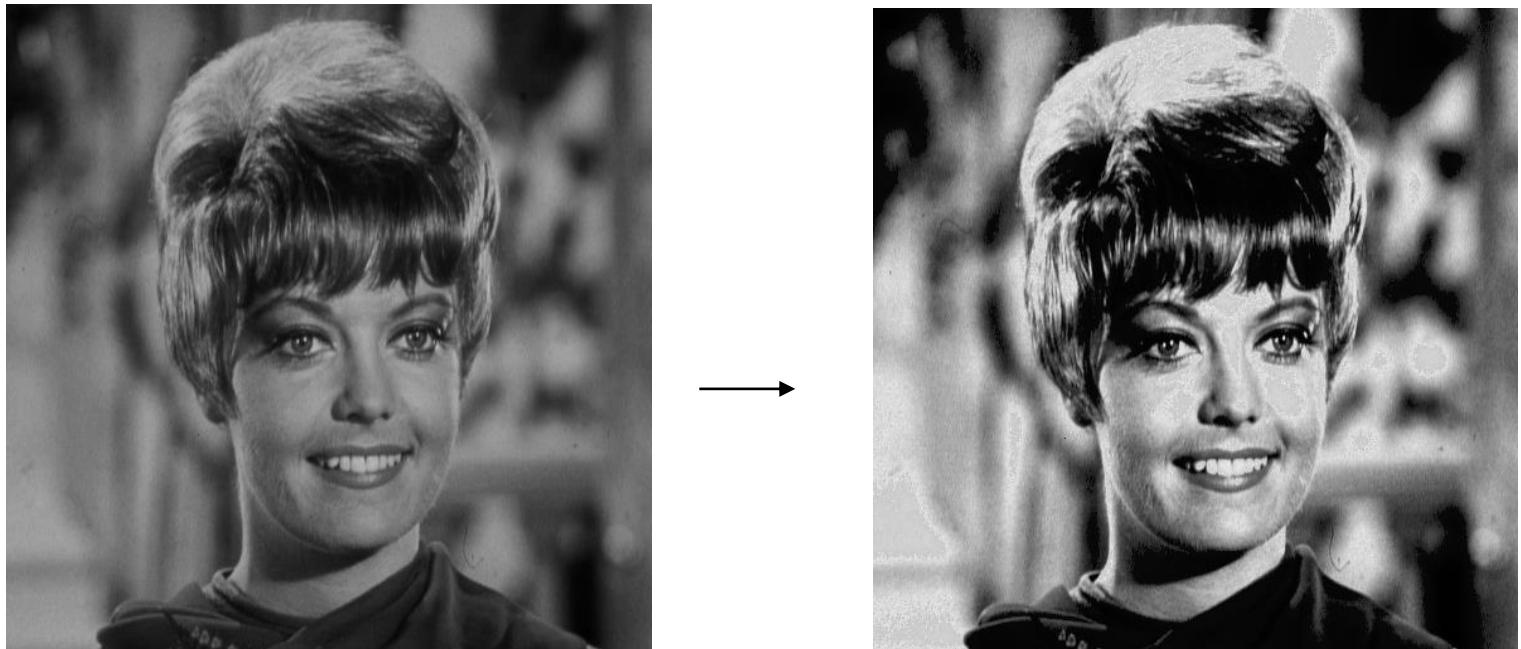
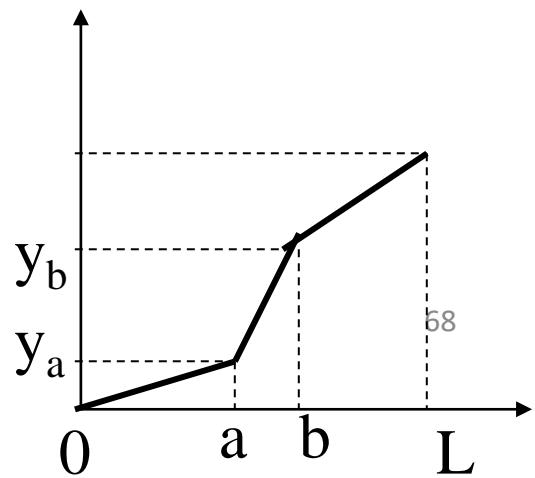
$$y = \begin{cases} \alpha x & 0 \leq x < a \\ \beta(x-a) + y_a & a \leq x < b \\ \gamma(x-b) + y_b & b \leq x < L \end{cases}$$

Persamaan garis L1: $y = \frac{y_1}{x_1}x$

Persamaan garis L2: $y = \frac{y_2 - y_1}{x_2 - x_1} \cdot x + y_1$

Persamaan garis L3: $y = \frac{255 - y_2}{255 - x_2} \cdot x + y_2$





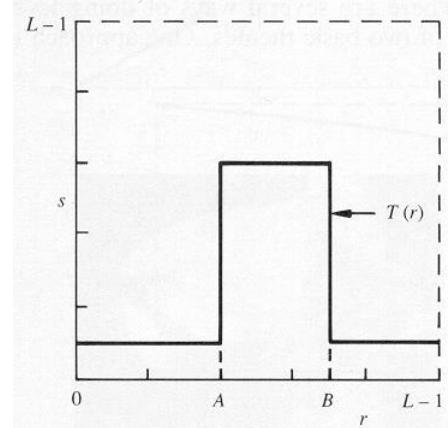
$$a = 50, b = 150, \alpha = 0.2, \beta = 2, \gamma = 1, y_a = 30, y_b = 200$$

$$y = \begin{cases} \alpha x & 0 \leq x < a \\ \beta(x-a) + y_a & a \leq x < b \\ \gamma(x-b) + y_b & b \leq x < L \end{cases}$$

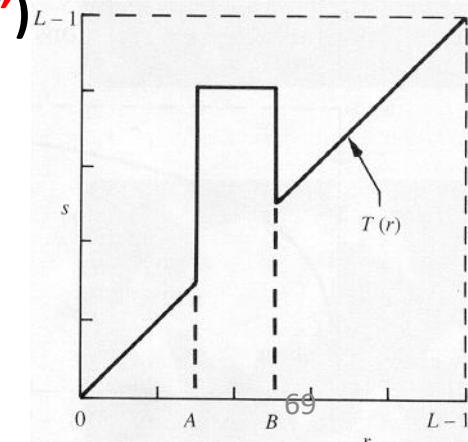
4. Gray-level Slicing

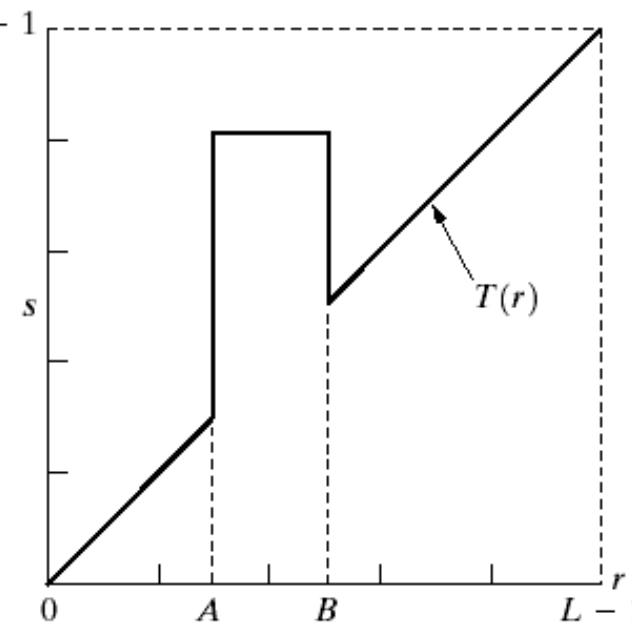
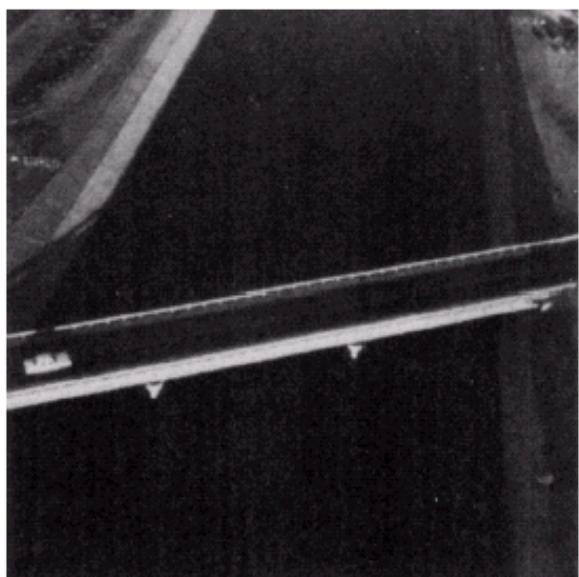
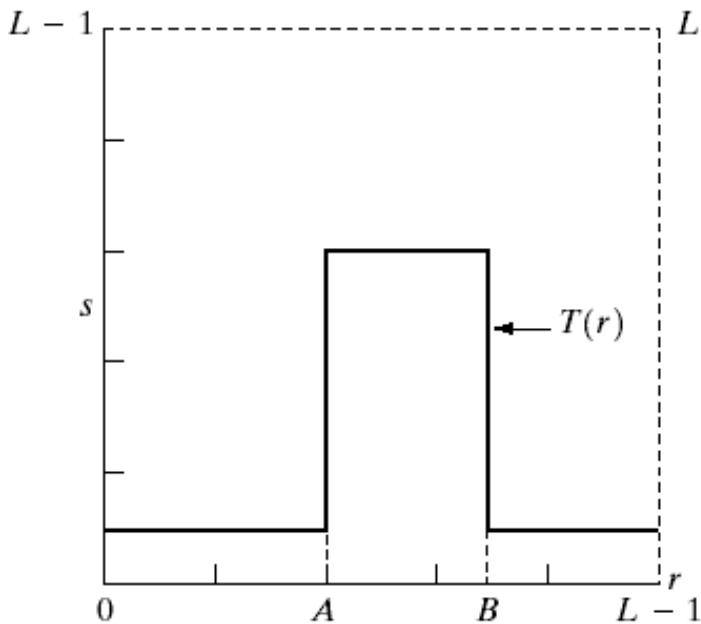
- Tujuan: menonjolkan (*highlight*) rentang keabuan tertentu di dalam citra.
- Contoh: menonjolkan gumpalan air yang ada pada citra satelit, menonjolkan cacat yang ada pada citra sinar X.
- Dua pendekatan di dalam *graylevel slicing*:

1. Menampilkan lebih terang semua *graylevel* di dalam rentang yang ingin ditonjolkan, dan menampilkan lebih gelap semua *graylevel* lainnya ('*discard background*').



2. Menampilkan lebih terang semua *graylevel* di dalam rentang yang ingin ditonjolkan, sembari tetap mempertahankan *graylevel* lainnya ('*preserve background*')





a	b
c	d

FIGURE 3.11

- (a) This transformation highlights range $[A, B]$ of gray levels and reduces all others to a constant level.
- (b) This transformation highlights range $[A, B]$ but preserves all other levels.
- (c) An image.
- (d) Result of using the transformation in (a).

Preserve Background

```
clear all ;
clc;
im = imread('kidney.tif');
z=double(im);
[row,col]=size(z);
for i=1:1:row
    for j=1:1:col
        if ((z(i,j)>142)) && (z(i,j)<250)
            z(i,j)=255;
        else
            z(i,j)=im(i,j);
        end
    end
end
figure(1); %-----Original Image-----%
imshow(im);
figure(2); %-----Gray Level Slicing With Background-----%
imshow(uint8(z));
```

Preserve Background

Intensity Level slicing (Example)



Sumber: Image Processing By Dr. Jagadish Nayak ,BITS Pilani, Dubai Campus

Discard Background

```
clear all ;
clc;
im = imread('kidney.tif');
z=double(im);
[row,col]=size(z);
for i=1:1:row
    for j=1:1:col
        if ((z(i,j)>142)) && (z(i,j)<250)
            z(i,j)=255;
        else
            z(i,j)=0;
        end
    end
end
figure(1); %-----Original Image-----%
imshow(im);
figure(2); %-----Gray Level Slicing With Background-----%
imshow(uint8(z));
```

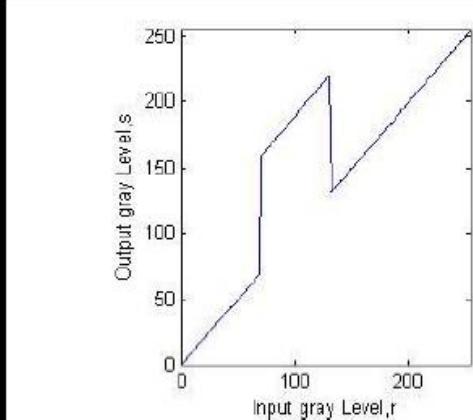
Discard Background

Intensity Level slicing (Example)



Sumber: Image Processing By Dr. Jagadish Nayak ,BITS Pilani, Dubai Campus

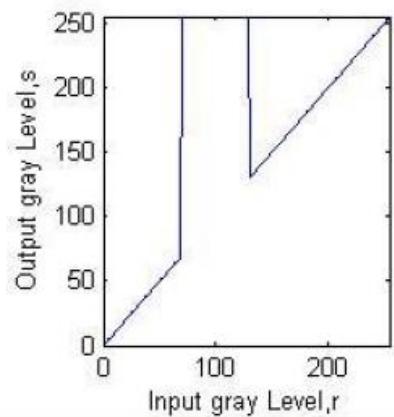
Slicing Example



Original Image



Enhanced Image



Original Image



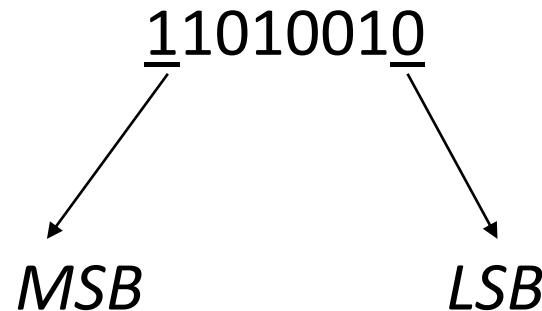
Enhanced Image

Sumber gambar: Ehsan Khoramshahi,
Image enhancement in spatial domain
75

5. Bit-plane Slicing

- Tujuan: Menonjolkan kontribusi dari bit tertentu di dalam citra.
- Misalkan satu pixel = 8 bit. Bit-bit tersusun dari kiri ke kanan dalam urutan yang kurang berarti (*least significant bits* atau *LSB*) hingga bit-bit yang berarti (*most significant bits* atau *MSB*).
- Susunan bit pada setiap *byte* adalah $b_7b_6b_5b_4b_3b_2b_1b_0$.

Contoh:



- Jika setiap bit dari setiap *pixel* diambil, maka diperoleh 8 buah bidang (*bit-plane*).

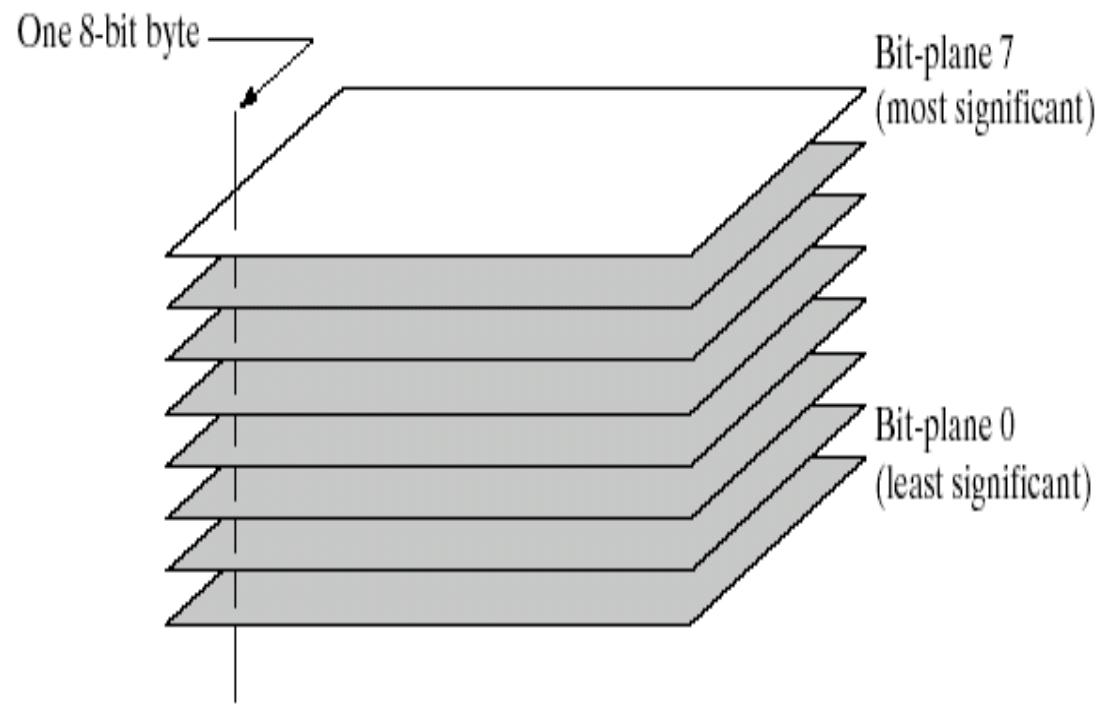


FIGURE 3.12
Bit-plane
representation of
an 8-bit image.

```
I = imread('cameraman.bmp');
imshow(I);
s = size(I);
for i=1:s(1)
    for j=1:s(2)
        for k=1:8
            P(i,j,k) = bitget(I(i,j), k);
        end
    end
end

P = logical(P);
for k=1:8
    figure, imshow(P(:,:,k))
end
```



Original image



Bitplane 7



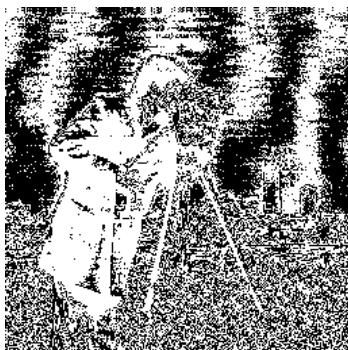
Bitplane 6



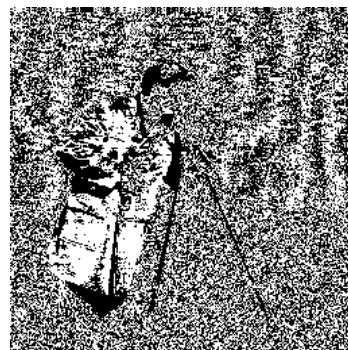
Bitplane 5



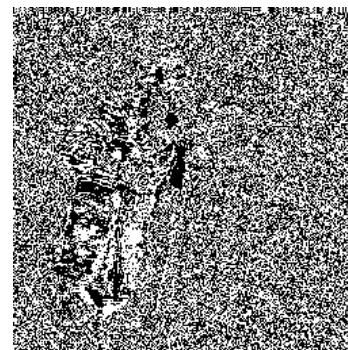
Bitplane 4



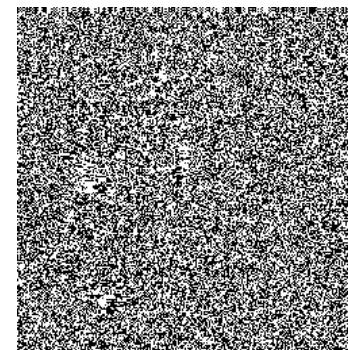
Bitplane 3



Bitplane 2



Bitplane 1



Bitplane 0

Contoh:

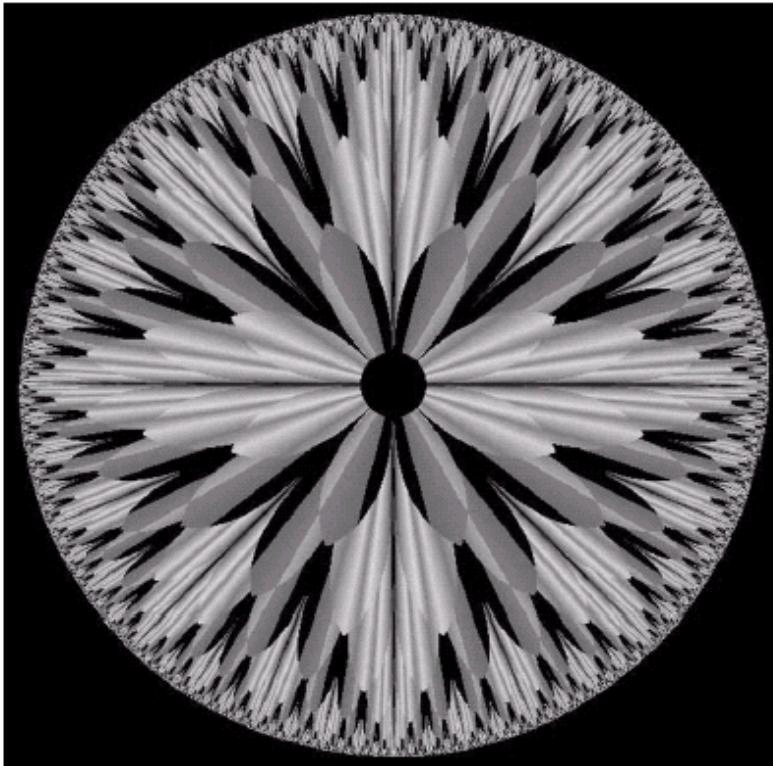
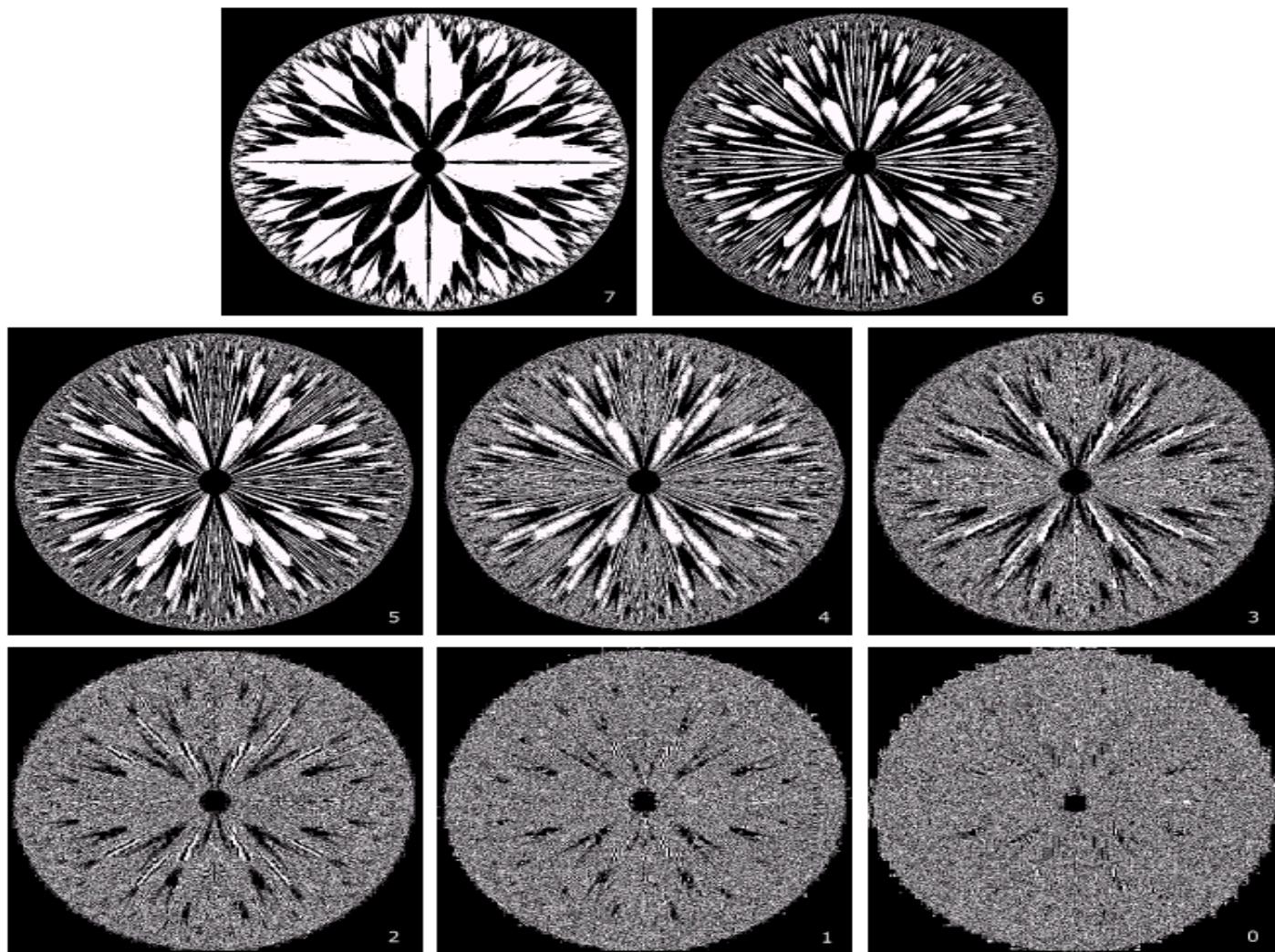


FIGURE 3.13 An 8-bit fractal image. (A fractal is an image generated from mathematical expressions). (Courtesy of Ms. Melissa D. Binde, Swarthmore College, Swarthmore, PA.)



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

FIGURE 3.14 The eight bit planes of the image in Fig. 3.13. The number at the bottom, right of each image identifies the bit plane.