

Image Enhancement

(Bagian 1)

IF4073 Interpretasi dan Pengolahan Citra

Oleh: Rinaldi Munir

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
2019

Image Enhancement

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



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:

1. *Image enhancement* dalam ranah spasial

2. *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.

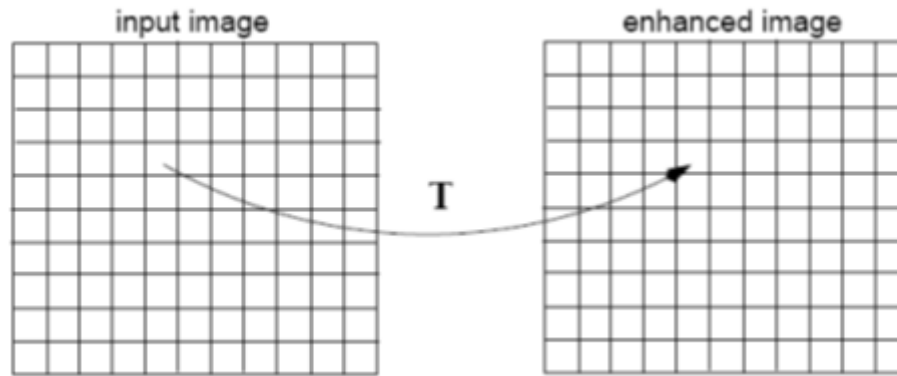
Metode dalam Ranah Spasial

- 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.

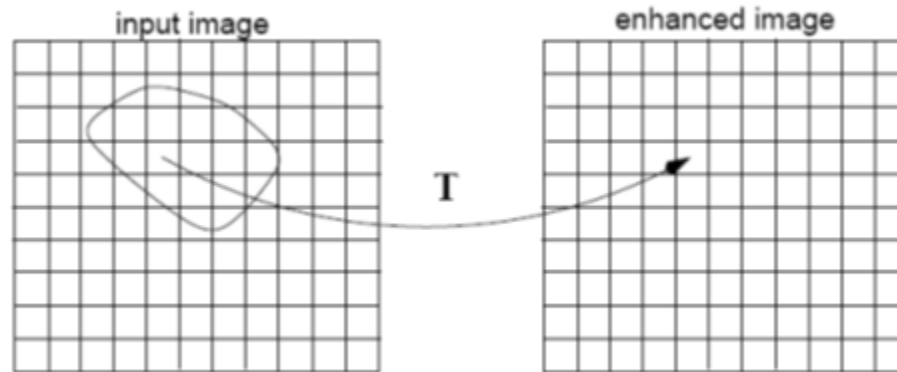
Aras lokal



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

T operates on 1 pixel

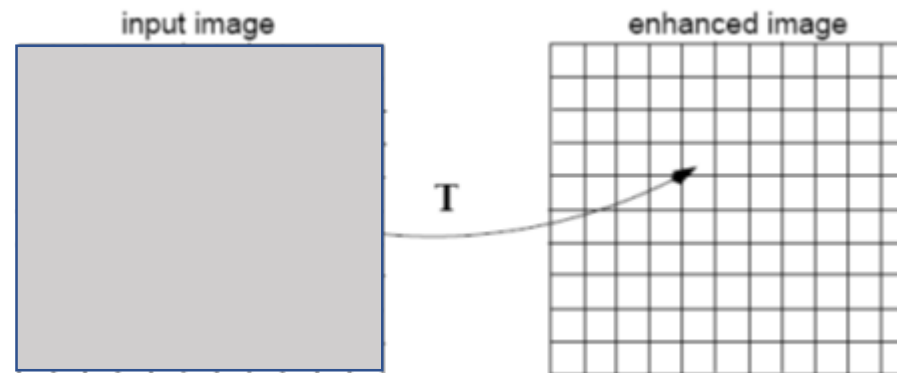
Aras lokal



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

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 (*sharpening*) tepi (*edge*).
- Pewarnaan semu (*pseudocolouring*)
- Pengubahan geometrik
- dll

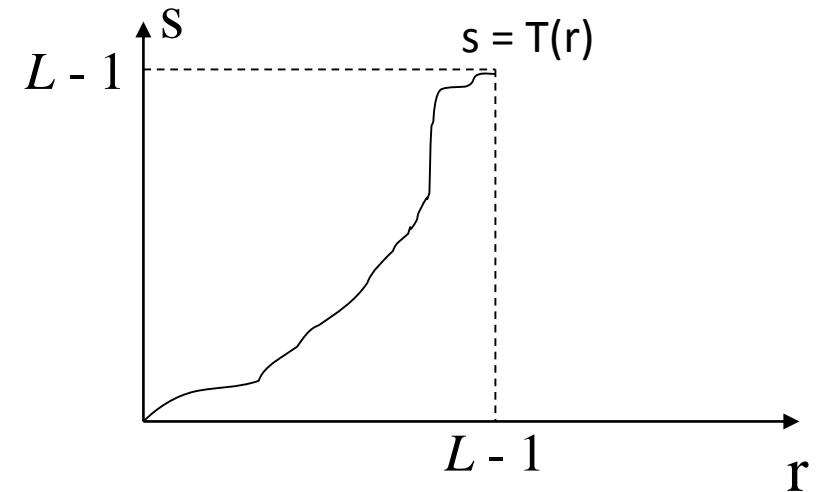
Pemrosesan dalam aras titik

- $g(x,y) = T [f(x,y)]$
- T hanya beroperasi pada satu pixel
- 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

1. Pencerahan citra (*image brightening*)

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

$$r = s + b$$

- 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 $s = 0$

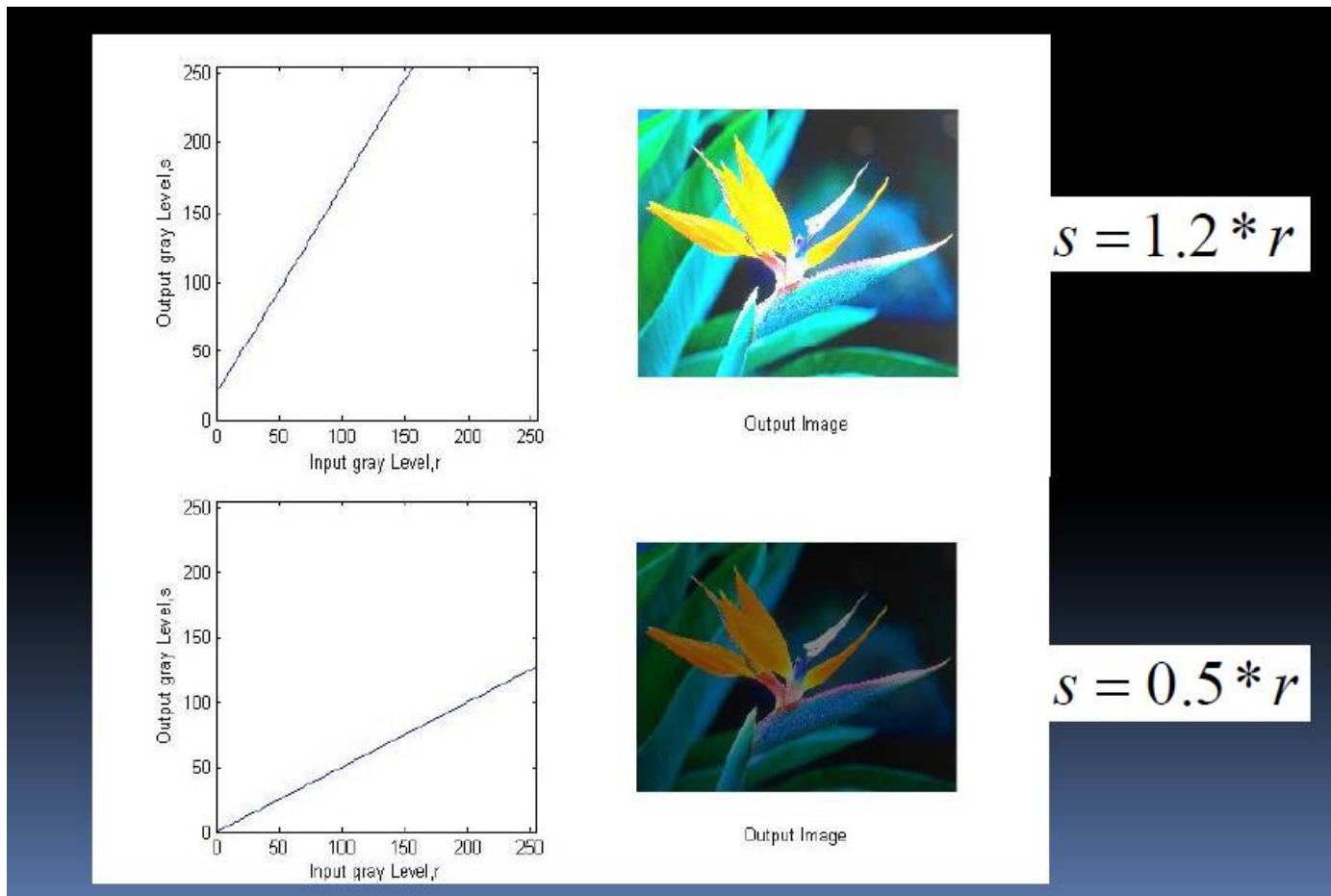


Gambar Kiri: citra Zelda (agak gelap); **kanan:** citra Zelda setelah operasi pencerahan citra, $b = 100$

- Operasi pencerahan yang lain adalah menggunakan rumus:

$$r = as + b$$

a dan b adalah konstanta



Sumber gambar: Ehsan Khoramshahi,
Image enhancement in spatial domain

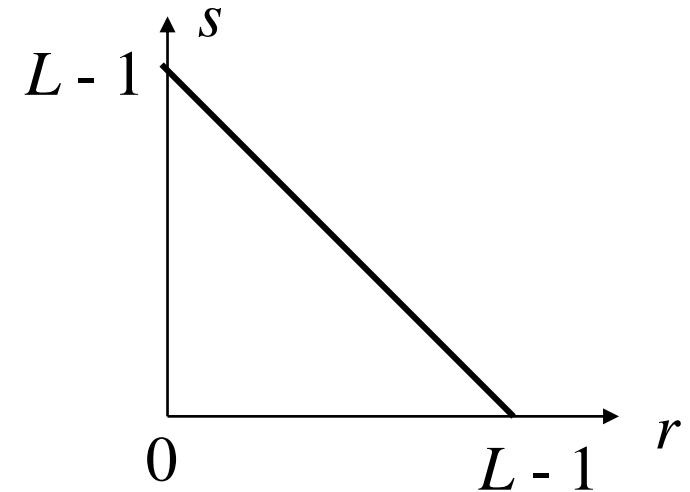
2. Menegatifkan Citra (*Image Negatives*)

- 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:

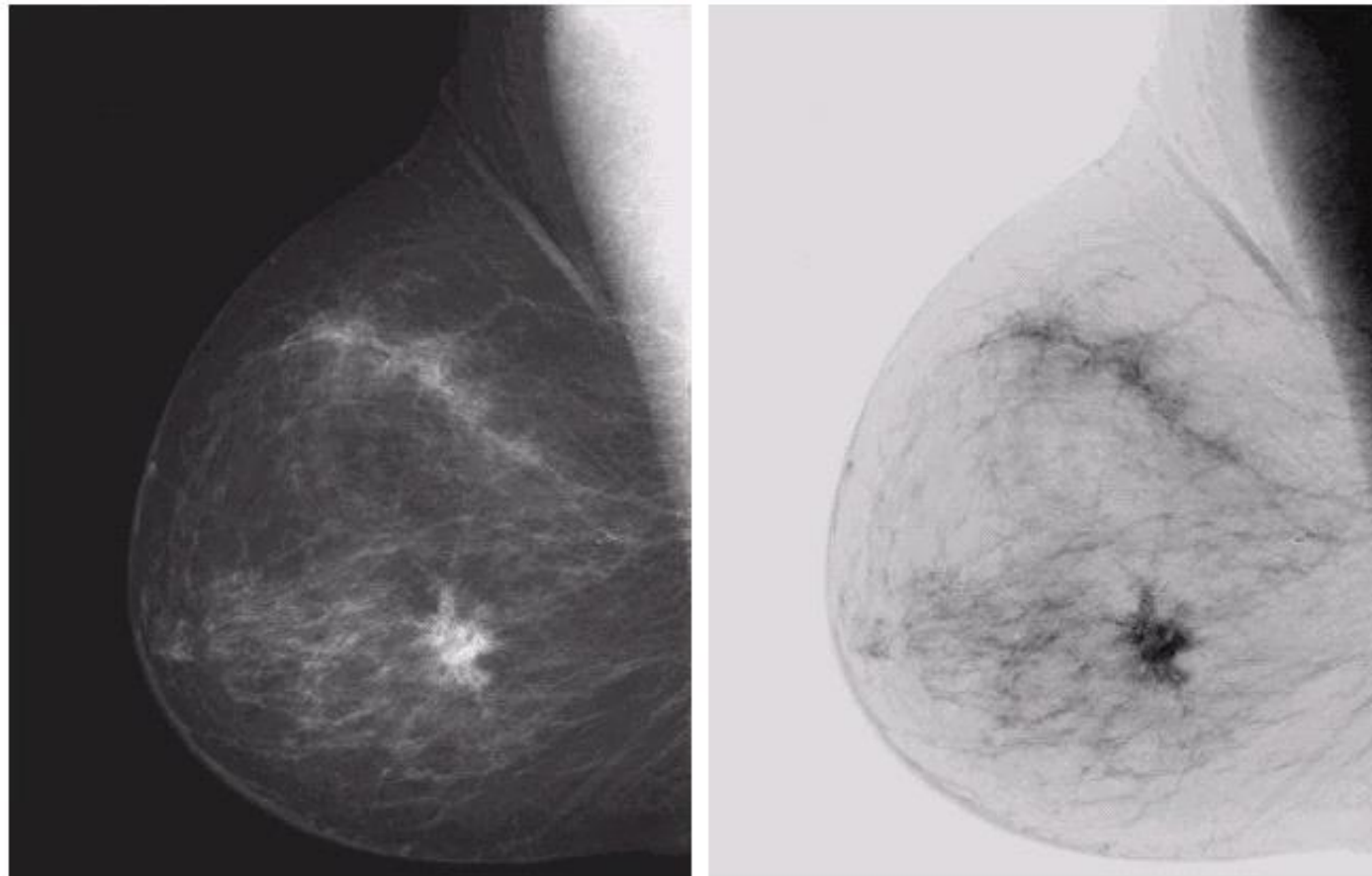
$$s = 255 - r$$







- 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 negative dan transformasi identitas

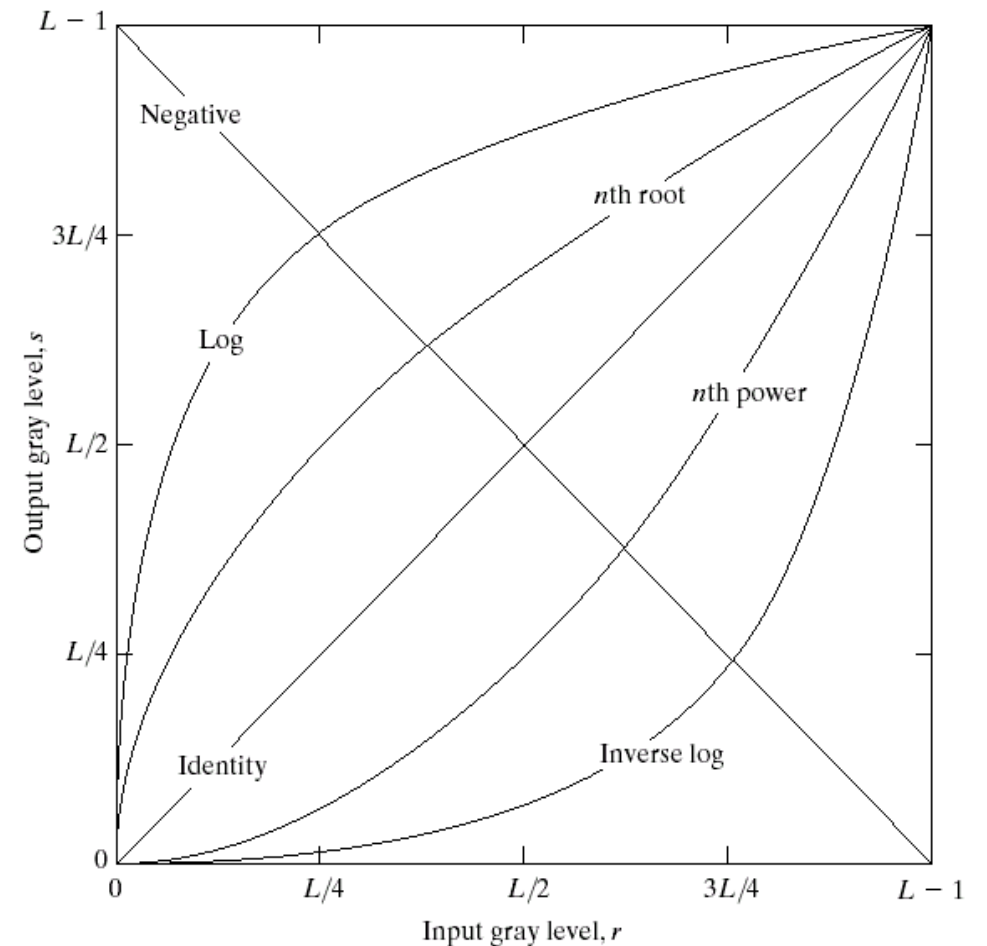
2. Fungsi logaritma

- Transformasi log dan inverse-log

3. Fungsi pangkat

- Transformasi pangkat n dan transformasi akar pangkat n

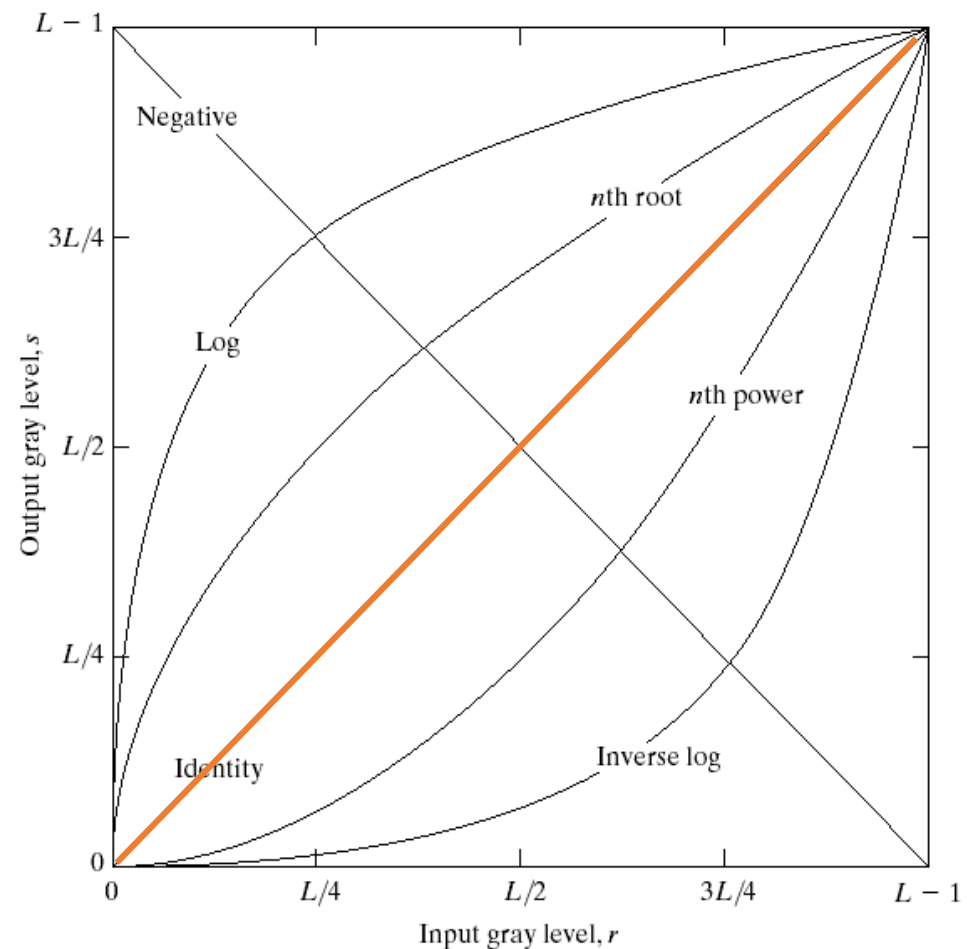
FIGURE 3.3 Some basic gray-level transformation functions used for image enhancement.



a) Transformasi identitas

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

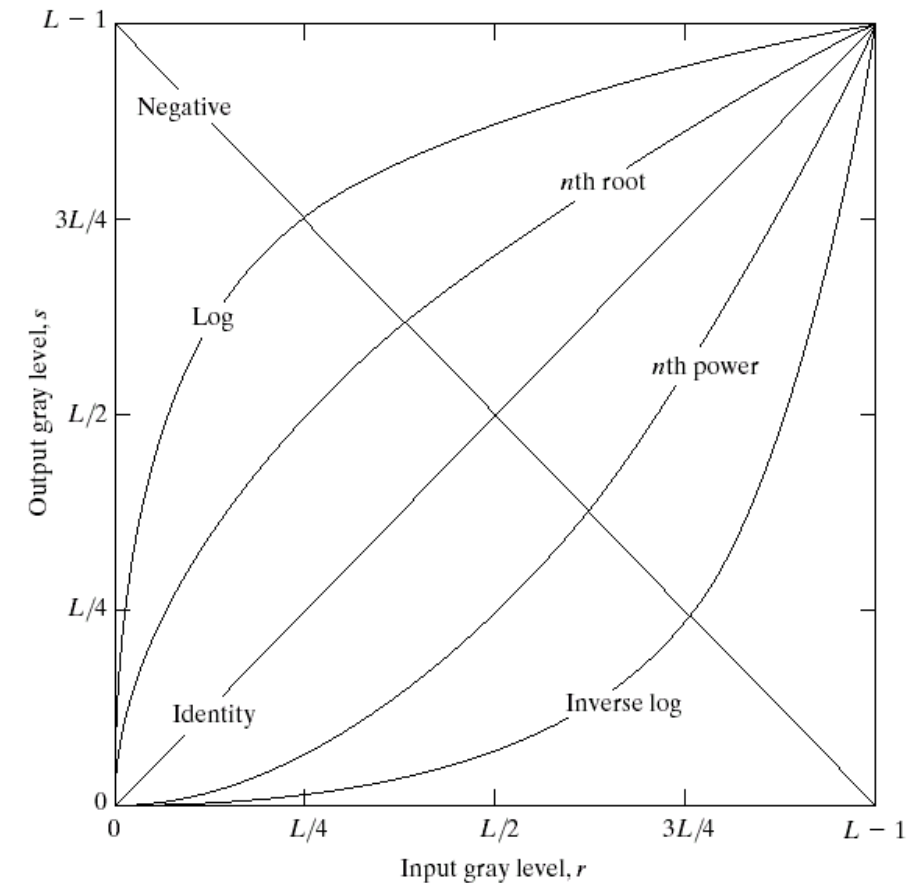
FIGURE 3.3 Some basic gray-level transformation functions used for image enhancement.



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.

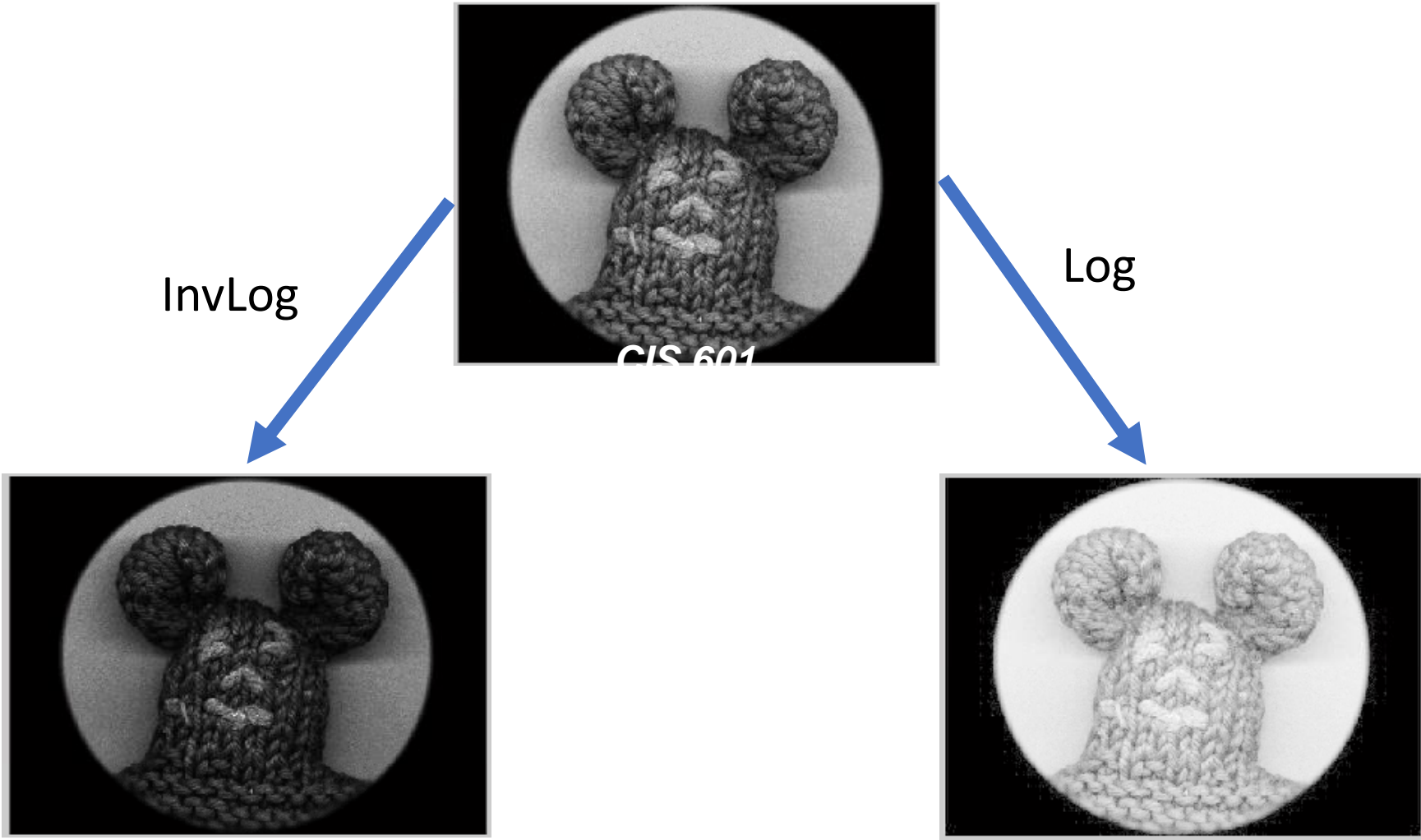
FIGURE 3.3 Some basic gray-level transformation functions used for image enhancement.





c = 100

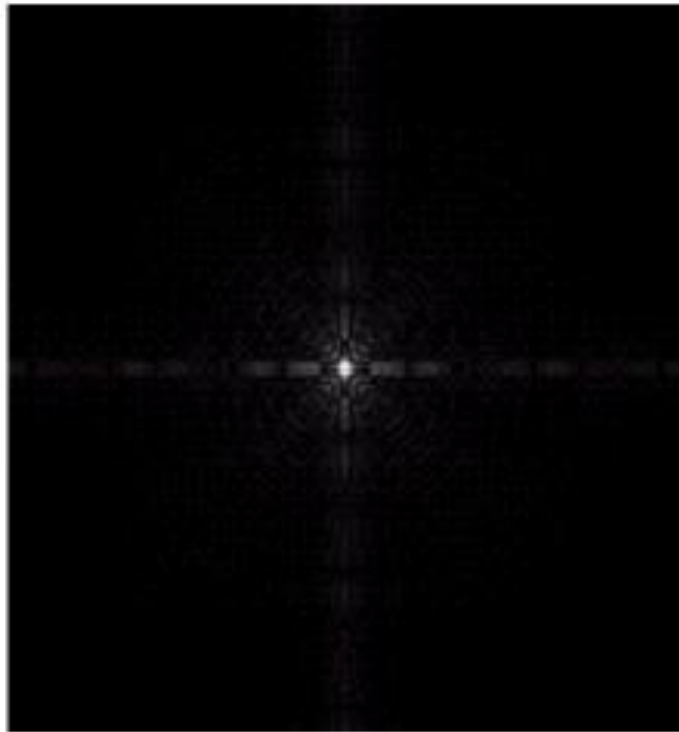




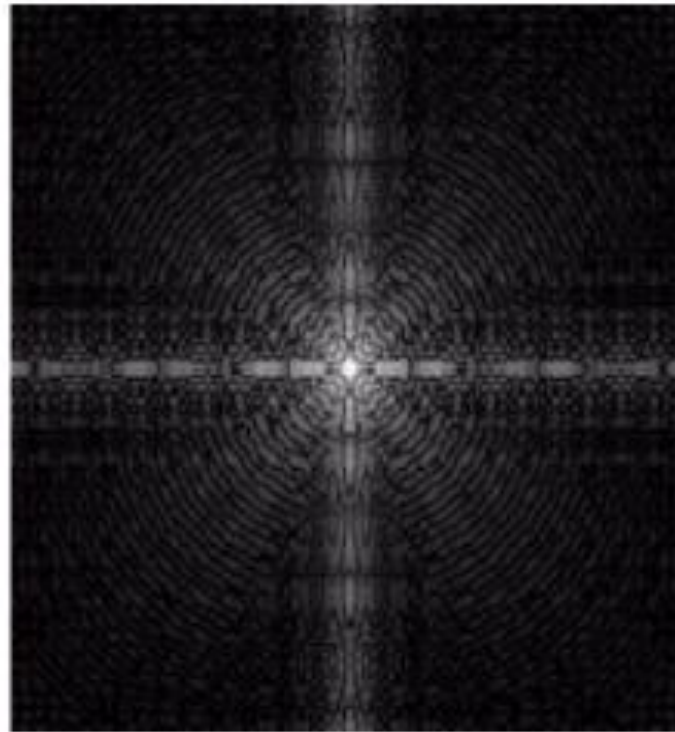
Sumber gambar: CIS 601, Image ENHANCEMENT in the SPATIAL DOMAIN, *Dr. Rolf Lakaemper*

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, $c = 1$

c) Transformasi Pangkat

- Fungsi pangkat:

$$s = cr^\gamma$$

c dan γ adalah konstanta positif.

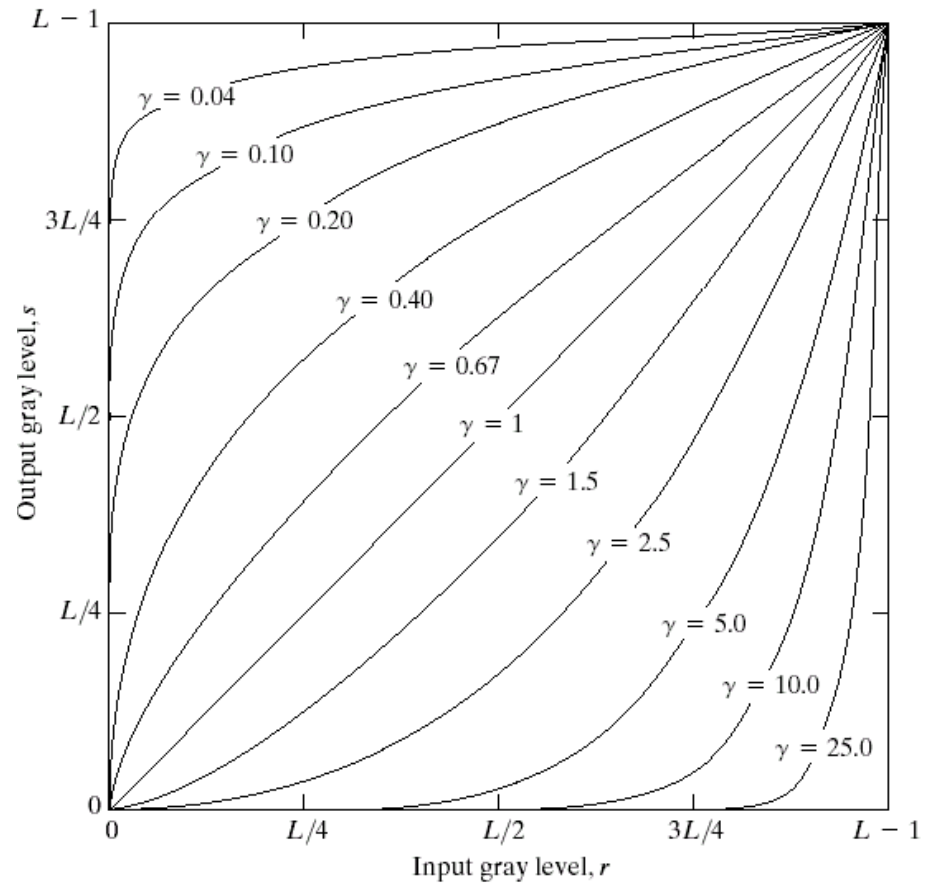


FIGURE 3.6 Plots of the equation $s = cr^\gamma$ for various values of γ ($c = 1$ in all cases).

Hukum pangkat (power-law):

Untuk $\gamma < 1$: Mengekspansi nilai-nilai pixel gelap, mengurangi nilai-nilai pixel terang

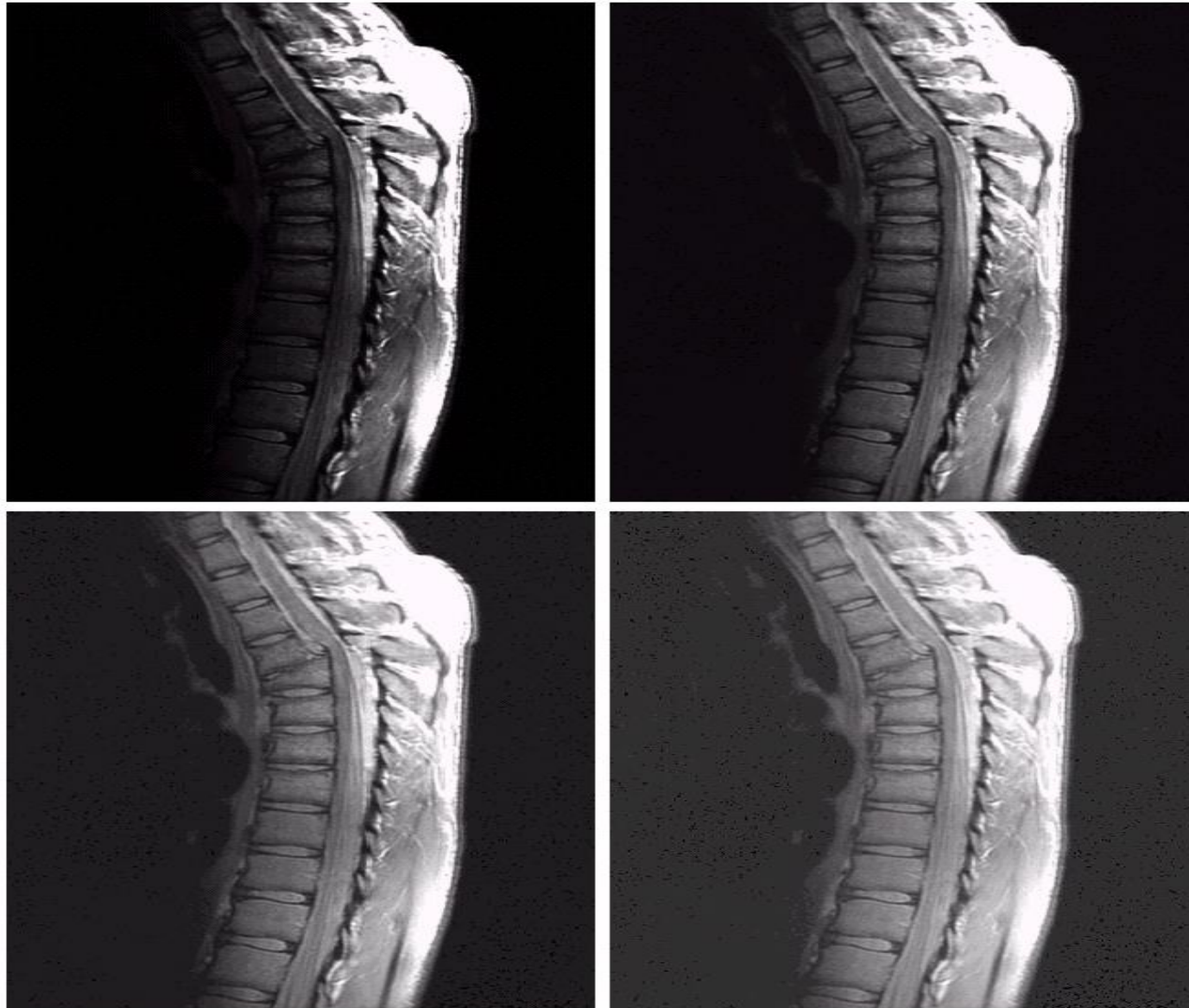
Untuk $\gamma > 1$: Mengurangi nilai-nilai pixel gelap, mengekspansi nilai-nilai pixel terang

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

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

Gamma (γ) correction

Proses yang digunakan untuk mengoreksi fenomena hukum-pangkat



a	b
c	d

FIGURE 3.8

(a) Magnetic resonance (MR) image of a fractured human spine.

(b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 0.6, 0.4,$ and 0.3 , respectively.

(Original image for this example courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)

Rincian:



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 $\gamma=4$

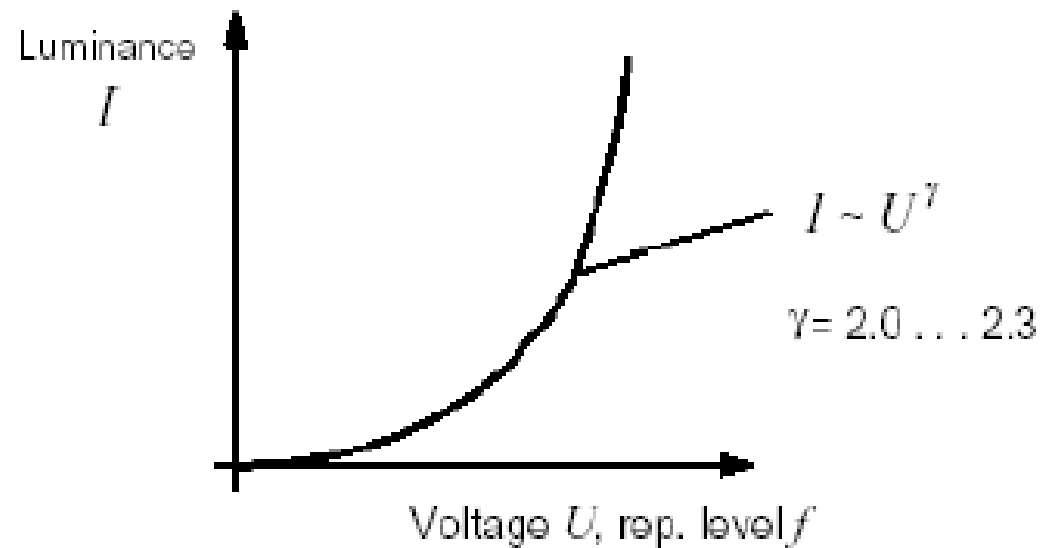


Result of a transformation
for $c=1$ and $\gamma=5$

Gamma correction

- Example of gamma correction

Cathode ray tubes (CRT) are nonlinear

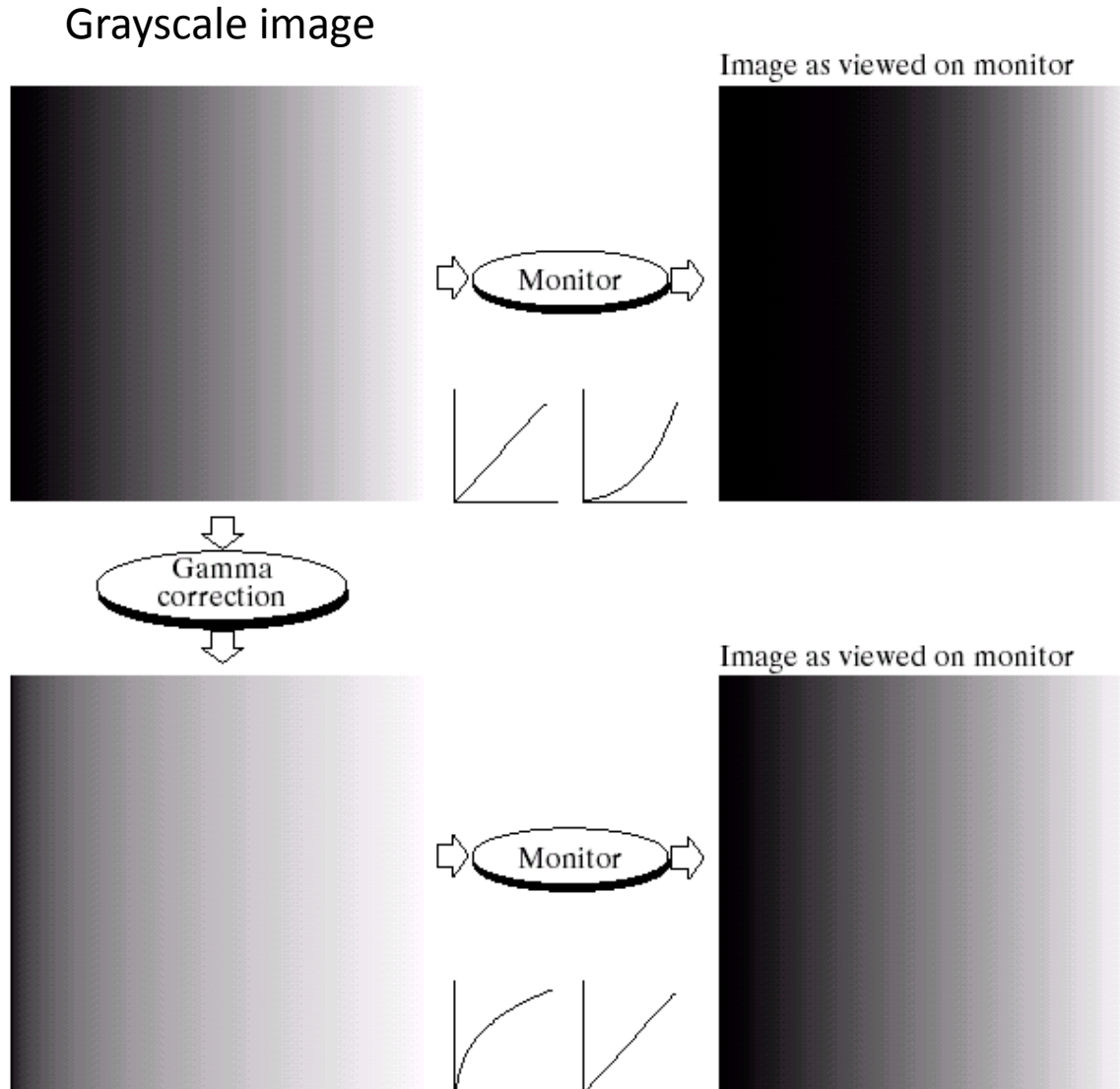


- To linearize the CRT response a pre-distortion circuit is needed $s = cr^{1/\gamma}$

Gamma correction

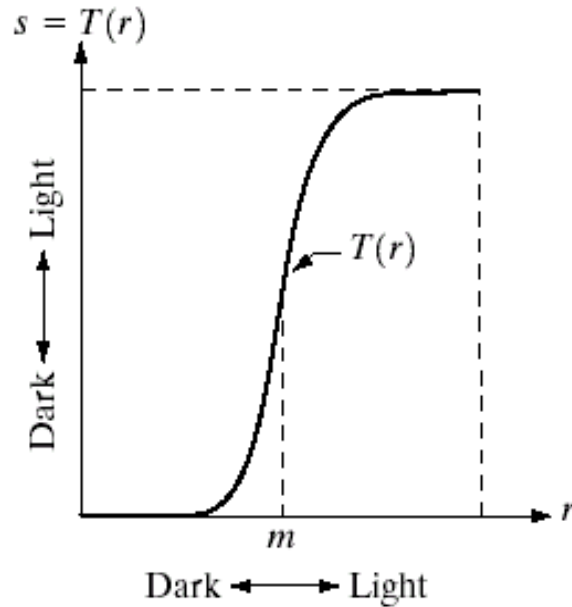
a b
c d

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

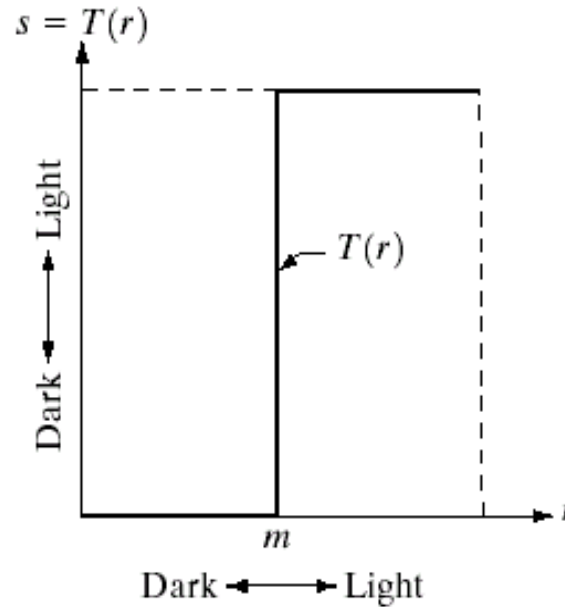
3. Perbaikan kontras (*contrast enhancement*)



(a)

Contrast Stretching

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



(b)

Thresholding

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

- $r = \text{graylevel}$ citra masukan
- $s = \text{graylevel}$ citra luaran
- $T =$ fungsi perbaikan kontras
- $m =$ nilai ambang



Original image



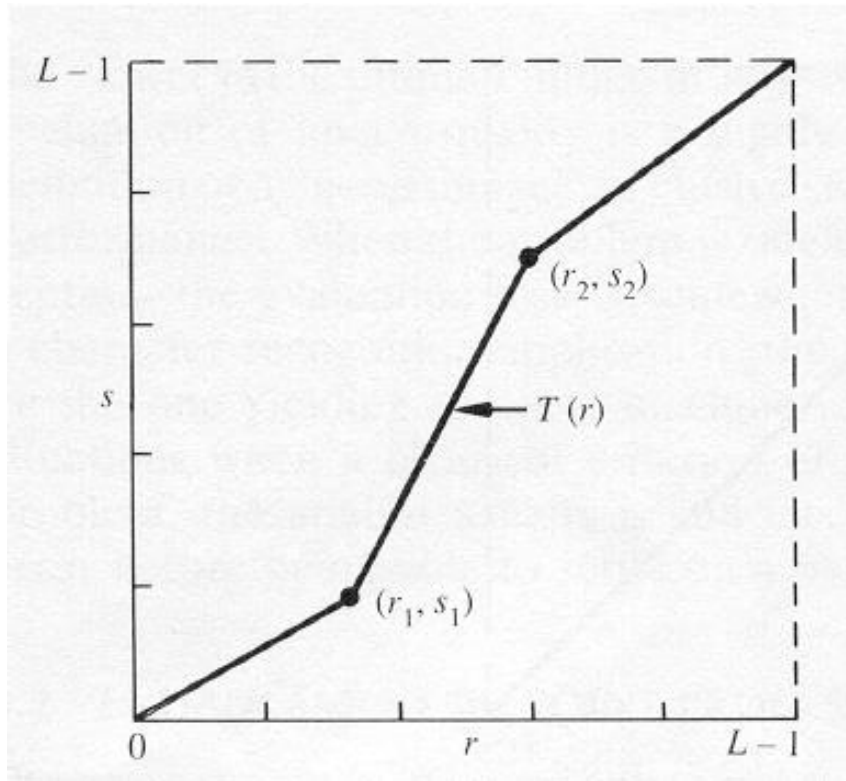
Peregangan kontras



Pengambangan

Peregangan kontras (*contrast stretching*)

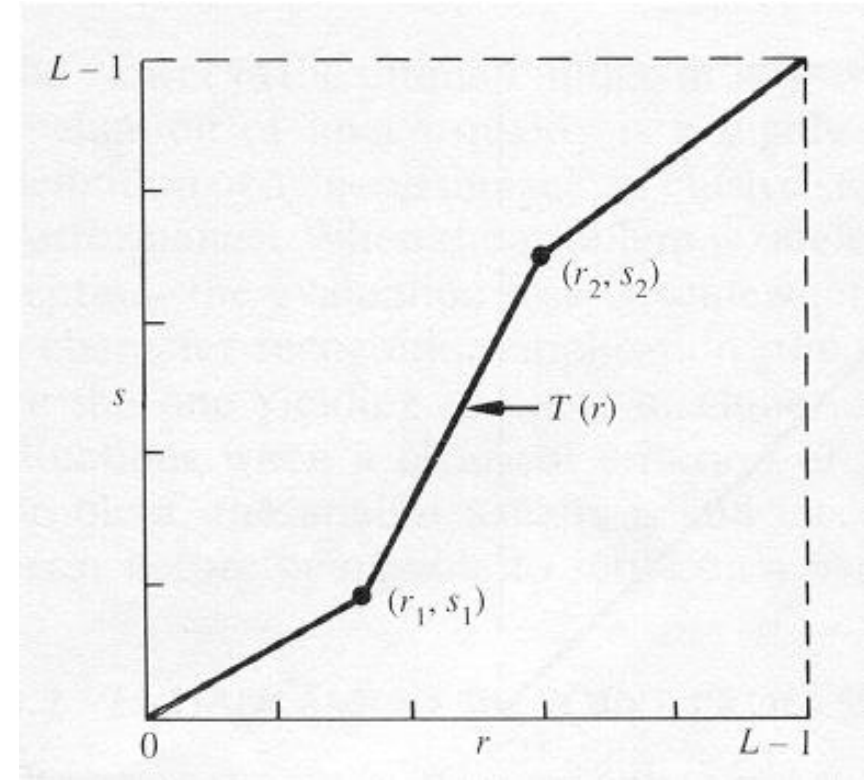
- Tujuan: meningkatkan rentang nilai-nilai keabuan untuk citra kontras-rendah (terentang dari 0 sampai 255 pada citra 8-bit)



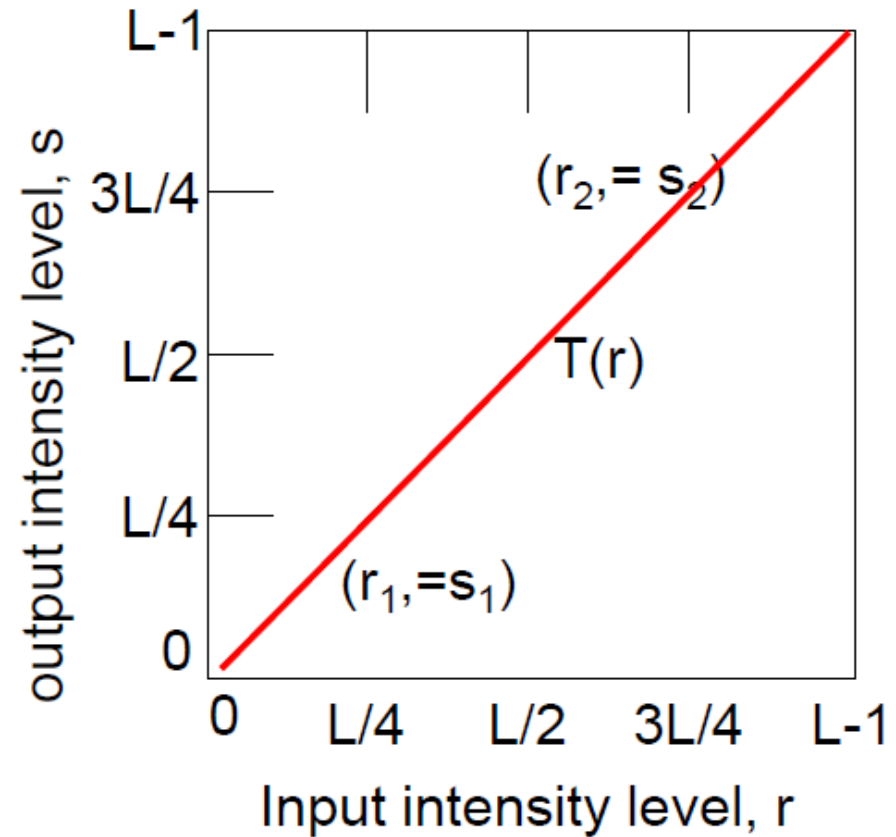
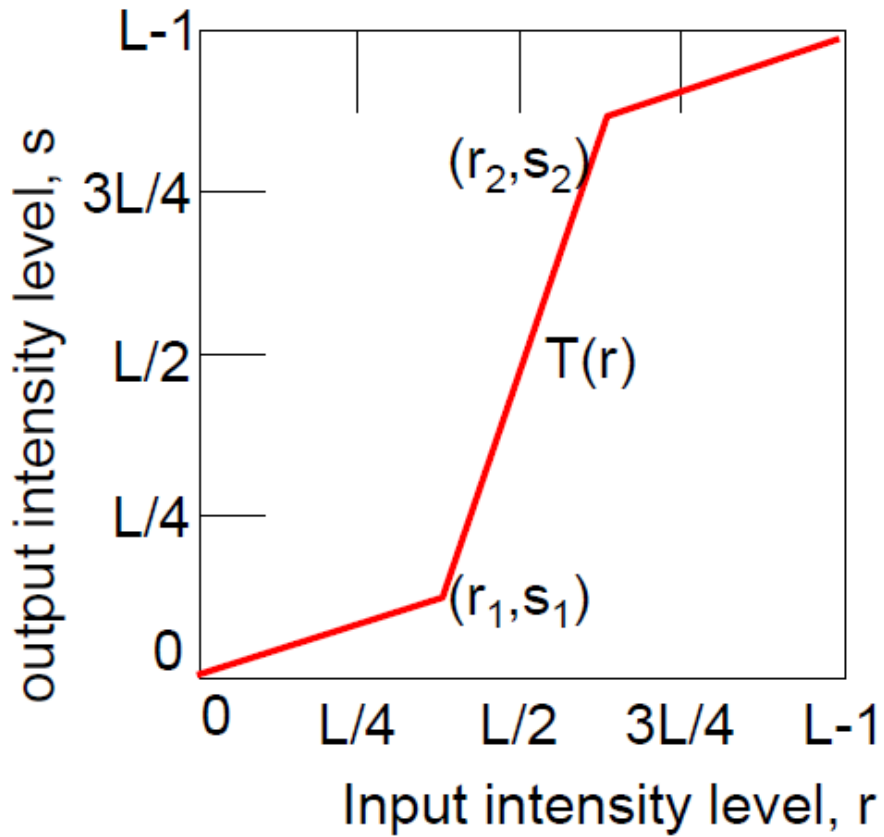
Citra kontras-rendah dihasilkan dari

- pencahayaan yang kurang
- kekurangan pada rentang dinamis di dalam *imaging sensor*
- kesalahan *setting* lensa selama akuisisi gambar

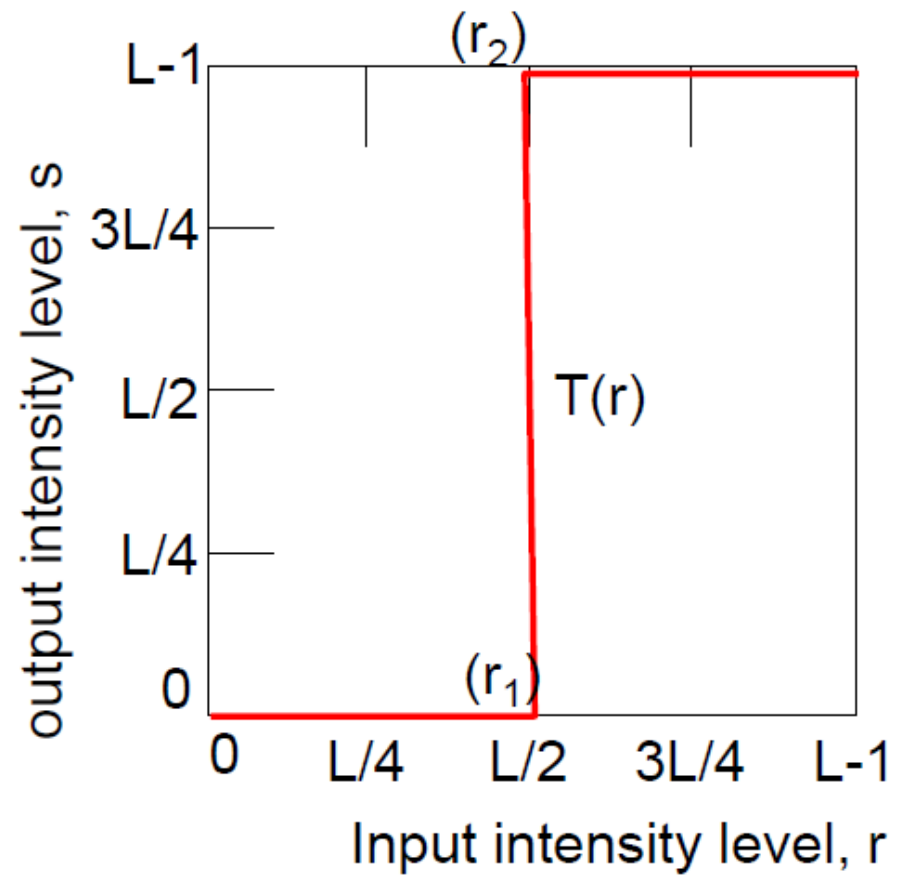
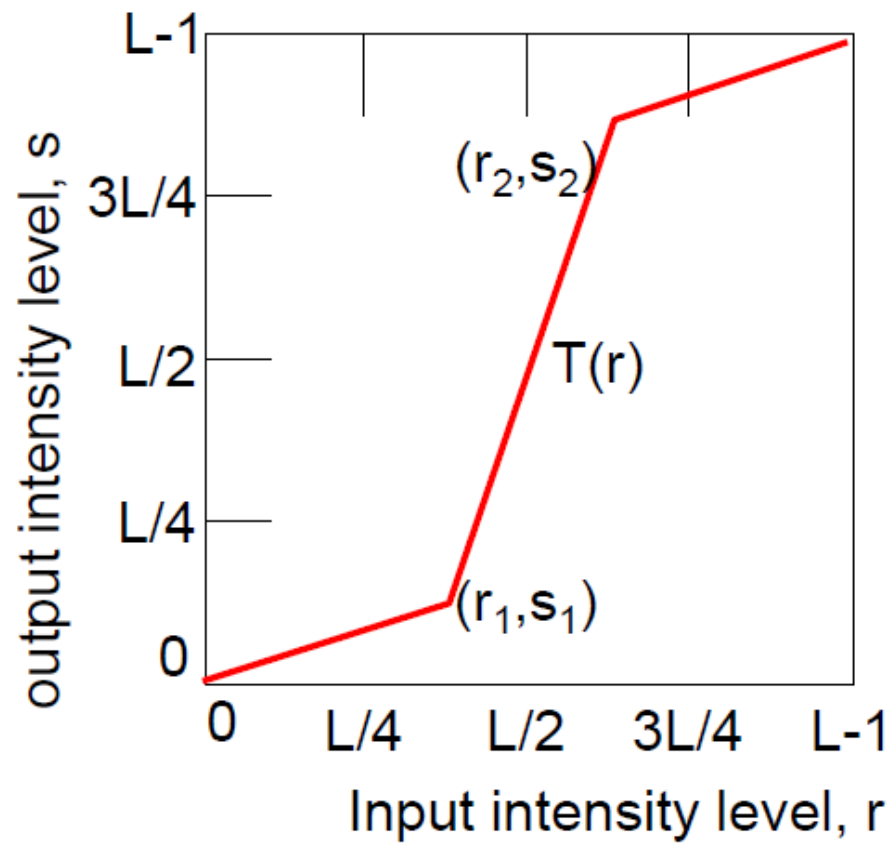
- Lokasi (r_1, s_1) dan (r_2, s_2) menentukan bentuk fungsi transformasi.
- Jika $r_1 = s_1$ dan $r_2 = s_2$ maka transformasi adalah fungsi linier sehingga tidak menghasilkan perubahan.
- Jika $r_1 = r_2$, $s_1 = 0$ dan $s_2 = L-1$, transformasi menjadi fungsi pengambangan yang menghasilkan citra biner.
- 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 $s_1 \leq s_2$



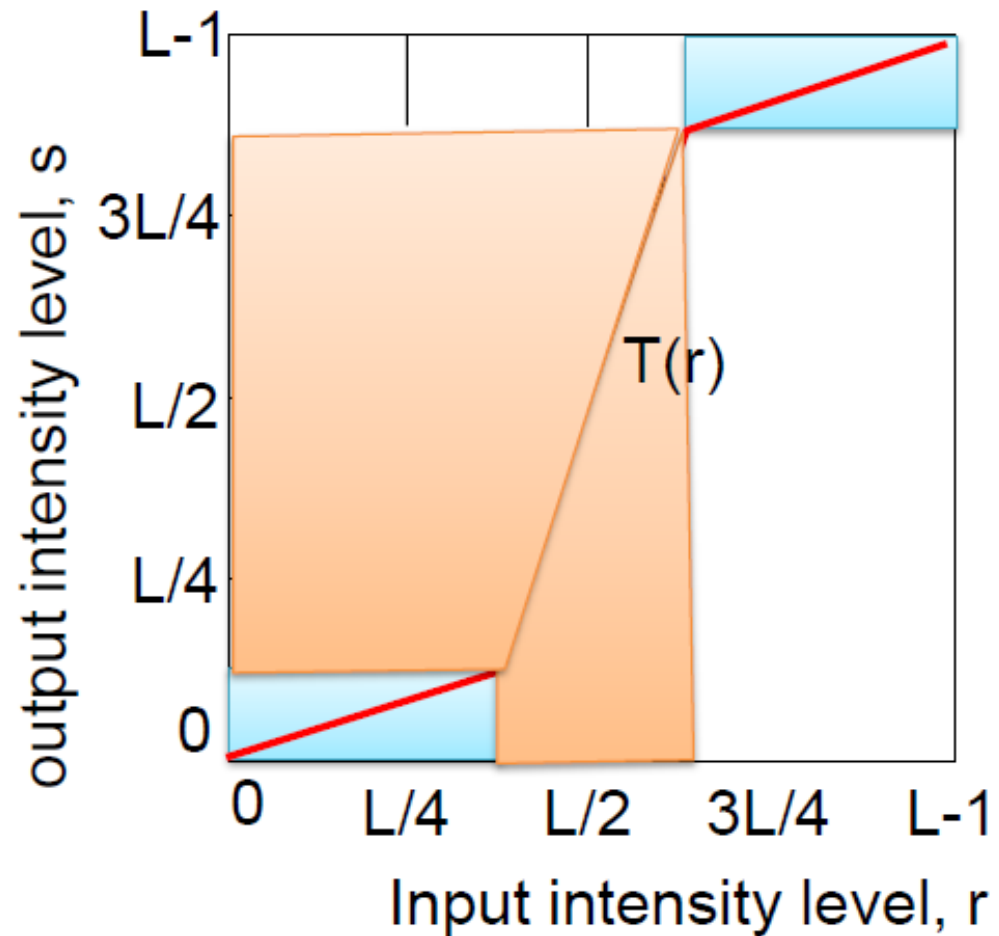
Jika $r_1 = s_1$ dan $r_2 = s_2$



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



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



Algoritma peregangan kontras

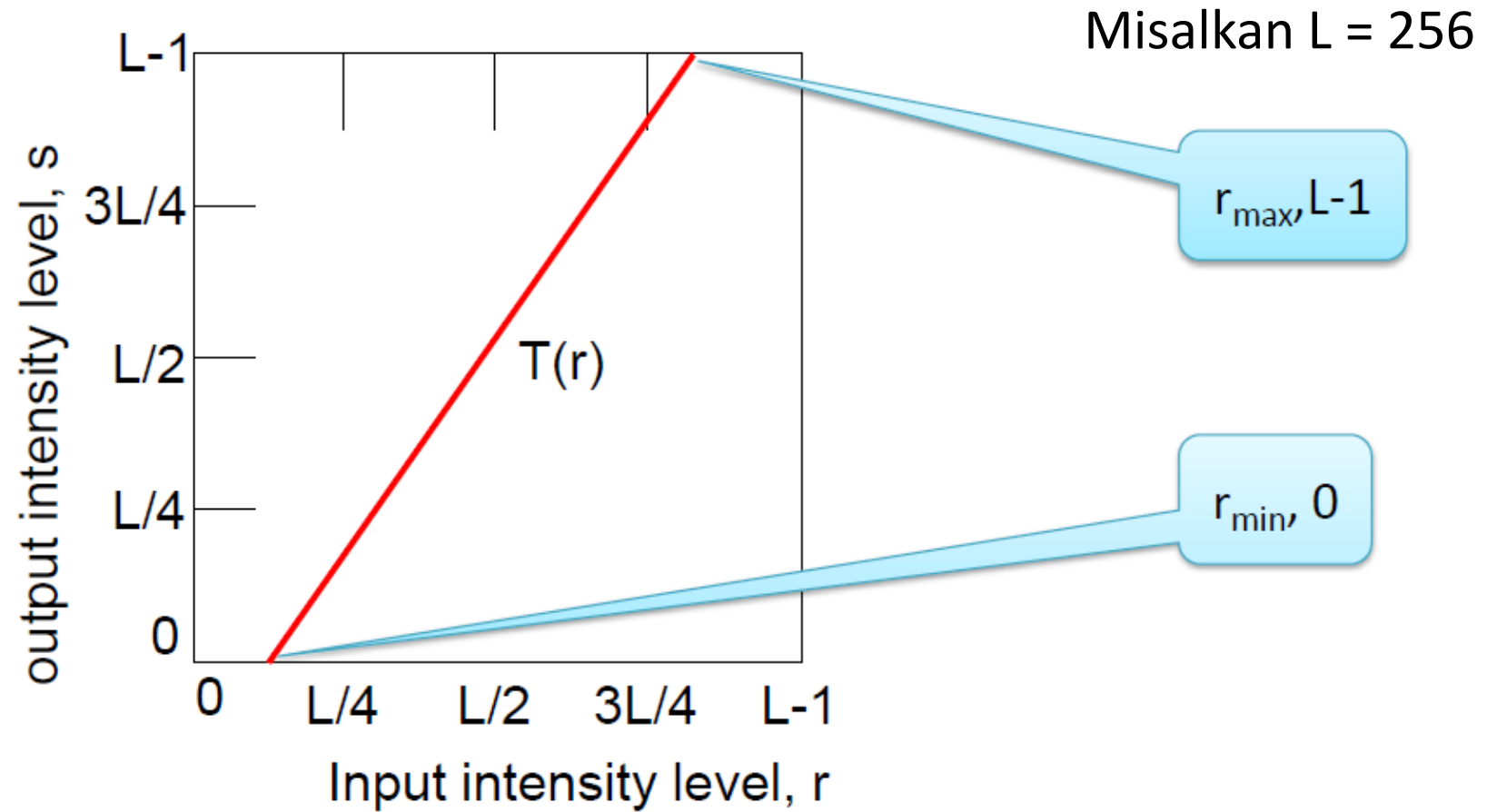
Input: I (citra masukan), r_{\min} dan r_{\max} (nilai ambang)

Luaran: O (citra luaran)

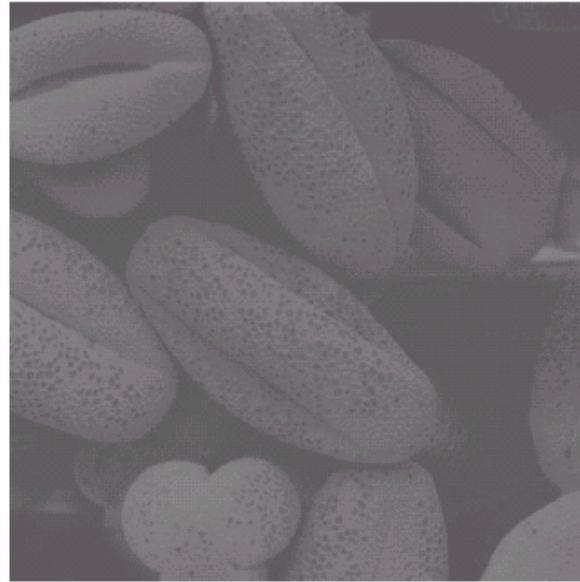
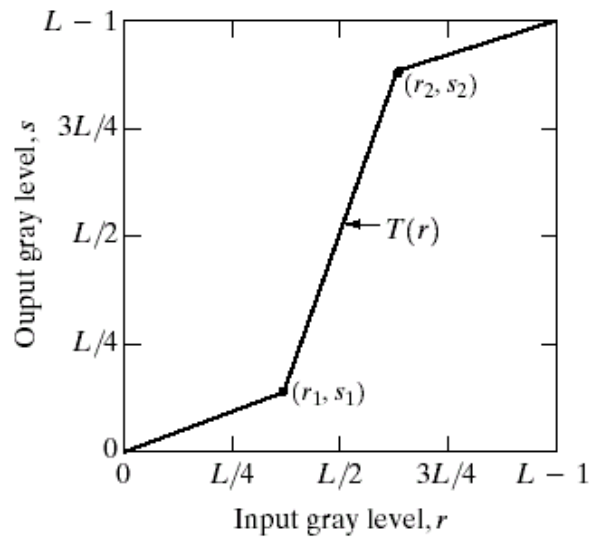
1. Misalkan citra I memiliki 256 derajat keabuan ($L = 256$).
2. Tentukan dua nilai ambang (a dan b). (a dan b menyatakan persentase, yaitu $a\%$ akan dinyatakan sebagai pixel hitam dan antara $100\% - b\%$ sebagai pixel putih)
3. Cari batas bawah pengelompokan *pixel* dengan cara memindai (*scan*) histogram dari nilai keabuan terkecil ke nilai keabuan terbesar (0 sampai 255) untuk menemukan *pixel* pertama yang melebihi nilai ambang pertama yang telah dispesifikasikan.
4. Cari batas atas pengelompokan *pixel* dengan cara memindai histogram dari nilai keabuan tertinggi ke nilai keabuan terendah (255 sampai 0) untuk menemukan *pixel* pertama yang lebih kecil dari nilai ambang kedua yang dispesifikasikan.
5. Misalkan r_{\min} adalah nilai keabuan terendah dari kelompok *pixel*, dan r_{\max} adalah nilai keabuan tertinggi dari kelompok *pixel*
6. *Pixel-pixel* yang nilainya $\leq r_{\min}$ di-set sama dengan 0, sedangkan *pixel-pixel* yang nilainya $\geq r_{\max}$ di-set sama dengan 255. Jadi, $(r_1, s_1) = (r_{\min}, 0)$ dan $(r_2, s_2) = (r_{\max}, 255)$.
4. *Pixel-pixel* yang berada di antara r_{\min} dan r_{\max} dipetakan (diskalakan) secara proporsional untuk memenuhi rentang nilai-nilai keabuan yang lengkap (0 sampai 255), misalnya dengan persamaan:

$$s = \frac{r - r_{\max}}{r_{\min} - r_{\max}} \times 255$$

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

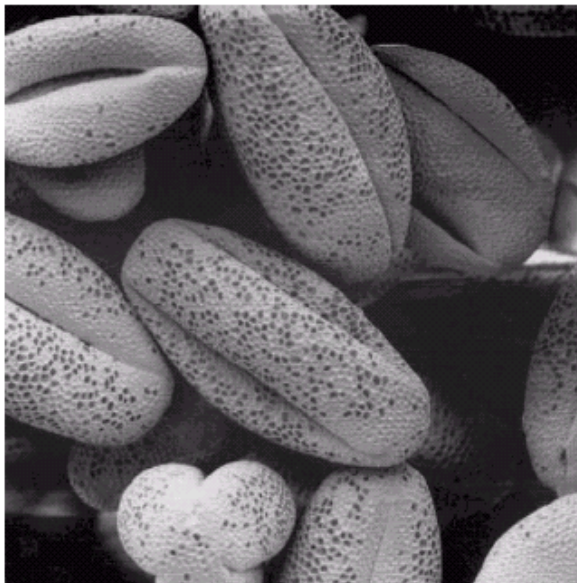


Contoh:

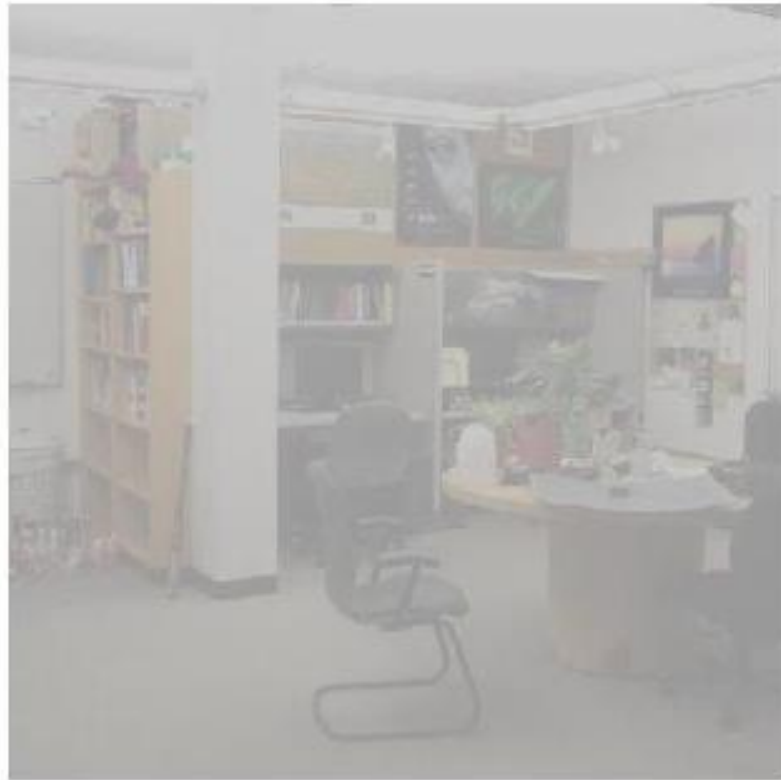


a b
c d

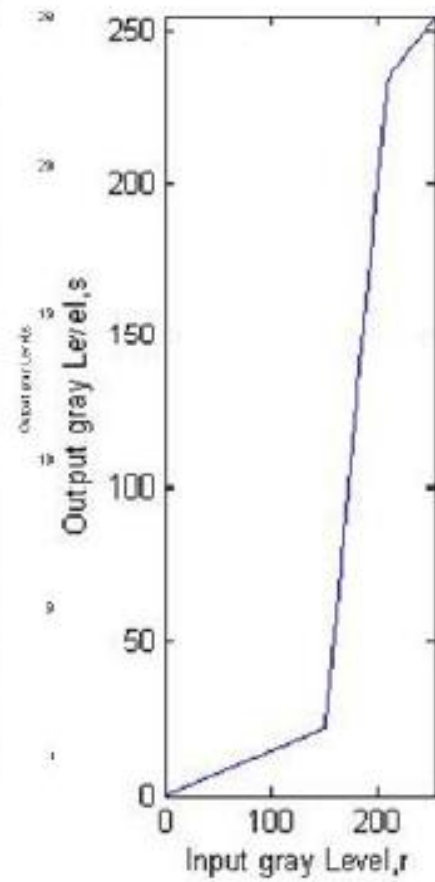
FIGURE 3.10
Contrast stretching.
(a) Form of transformation function. (b) A low-contrast image. (c) Result of contrast stretching. (d) Result of thresholding. (Original image courtesy of Dr. Roger Heady, Research School of Biological Sciences, Australian National University, Canberra, Australia.)



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



Original Image

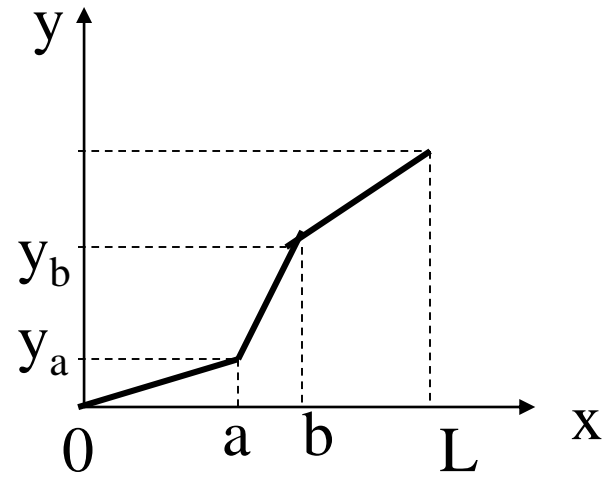


Enhanced Image

Sumber gambar: Ehsan Khoramshahi,
Image enhancement in spatial domain

- Peregangan kontras juga dapat dilakukan dengan rumus berikut:

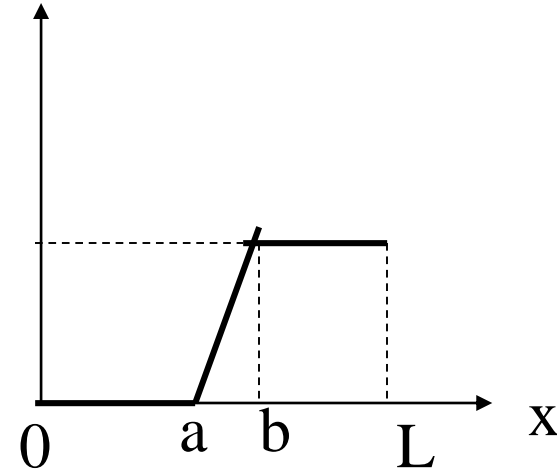
$$y = \begin{cases} \alpha x & 0 \leq x < a \\ \beta(x-a) + y_a & a \leq x < b \\ \gamma(x-b) + y_b & b \leq x < L \end{cases}$$



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

atau menggunakan rumus *clipping* berikut:

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

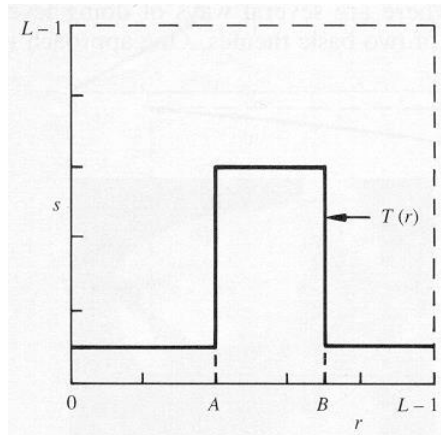


$$a = 50, b = 150, \beta = 2$$

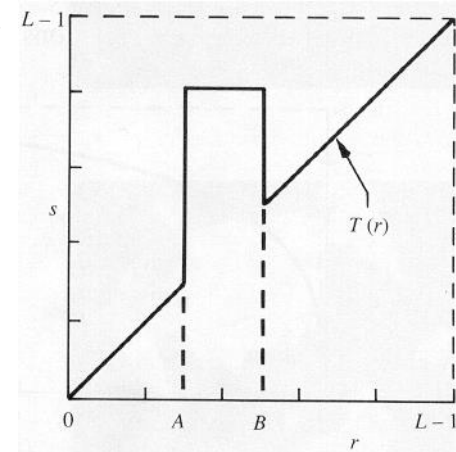
4. Gray-level Slicing

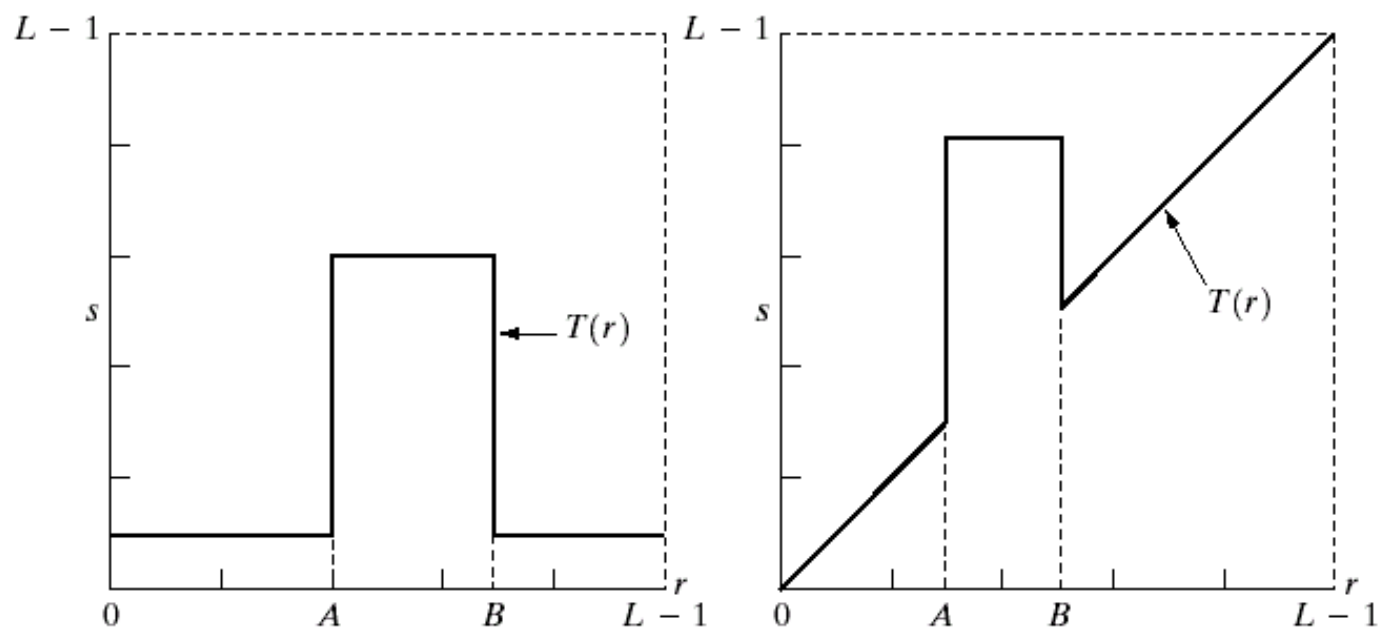
- Tujuan: menonjolkan (*highlight*) rentang keabuan tertentu di dalam citra.
- Contoh: menonjolkan gumpalan air yang ada pada citra satelit, menonjolkan cacat yang ada pada citra sinar X.
- Dua pendekatan di dalam *graylevel slicing*:

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

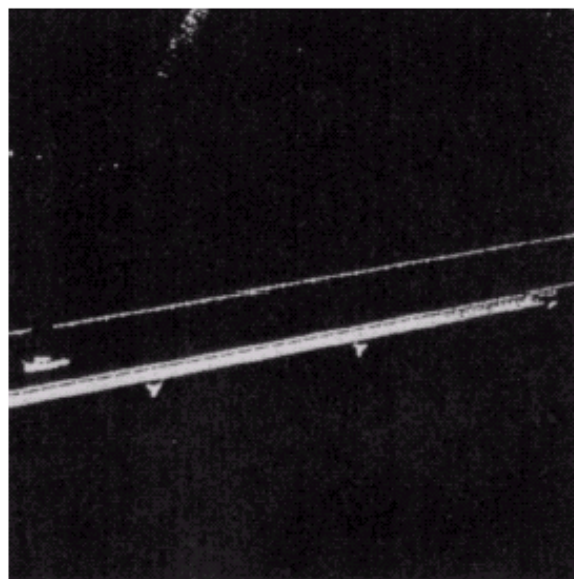
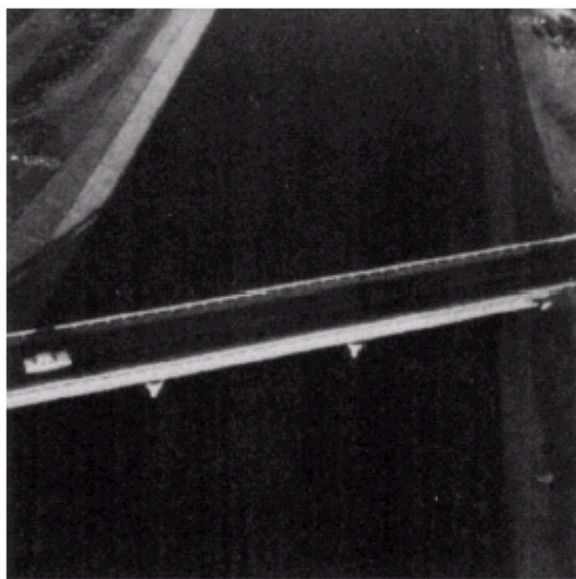
FIGURE 3.11

(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.

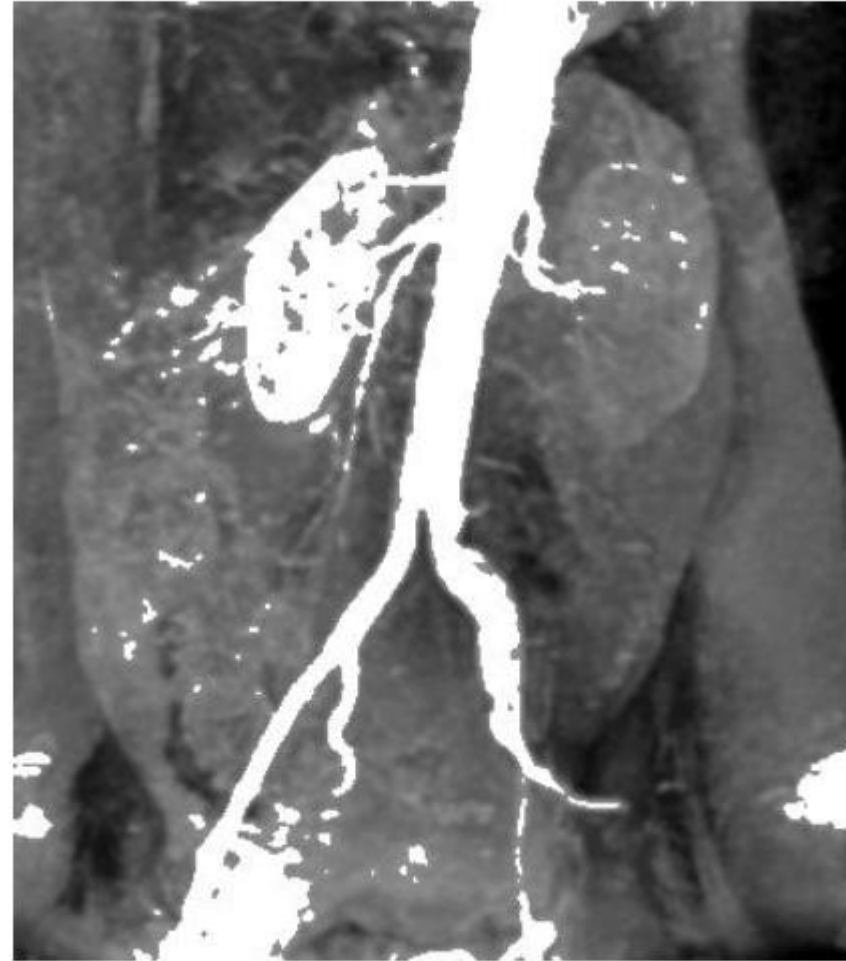
(d) Result of using the transformation in (a).



```
clear all ;
clc;
[filename, pathname] = uigetfile('*.tif');
im = imread([pathname filename]);
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
end
figure(1); %-----Original Image-----%
imshow(im);
figure(2); %-----Gray Level Slicing With Background-----%
imshow(uint8(z));
```

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

Intensity Level slicing (Example)



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

```
clear all ;
clc;
[filename, pathname] = uigetfile('*.tif');
im = imread([pathname filename]);
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
end
figure(1); %-----Original Image-----%
imshow(im);
figure(2); %-----Gray Level Slicing With Background-----%
imshow(uint8(z));
```

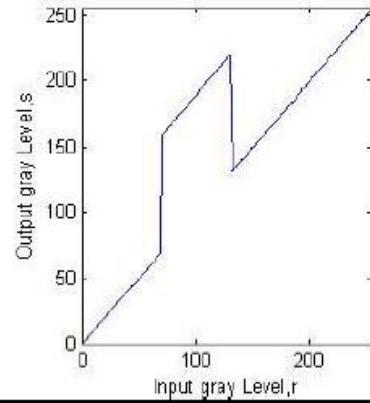
Sumber: Image Processing By Dr. Jagadish Nayak ,BITS Pilani, Dubai Campus

Intensity Level slicing (Example)



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

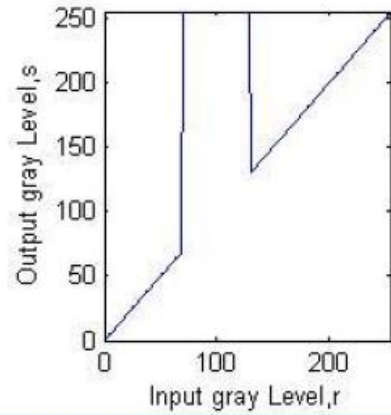
Slicing Example



Original Image



Enhanced Image



Original Image



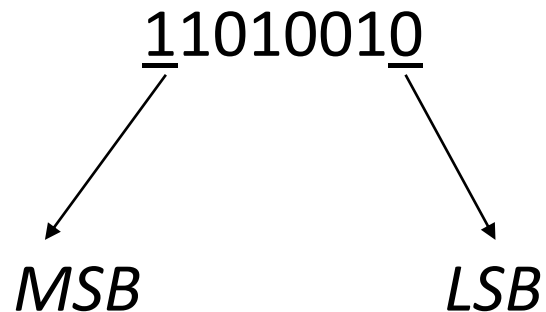
Enhanced Image

Sumber gambar: Ehsan Khoramshahi,
Image enhancement in spatial domain

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*).

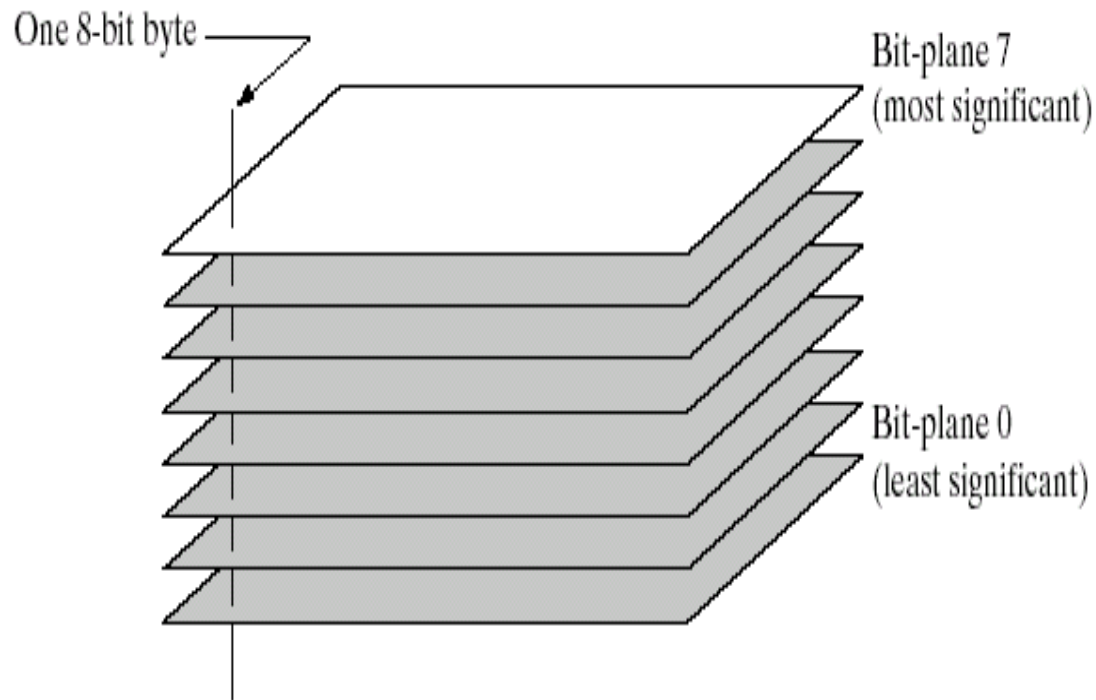


FIGURE 3.12
Bit-plane
representation of
an 8-bit image.

Contoh:

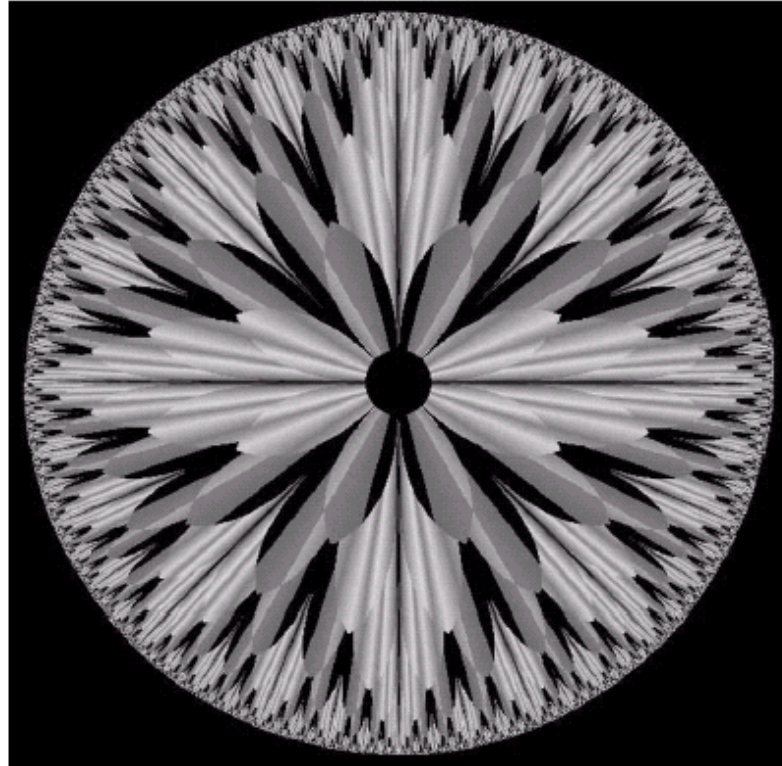
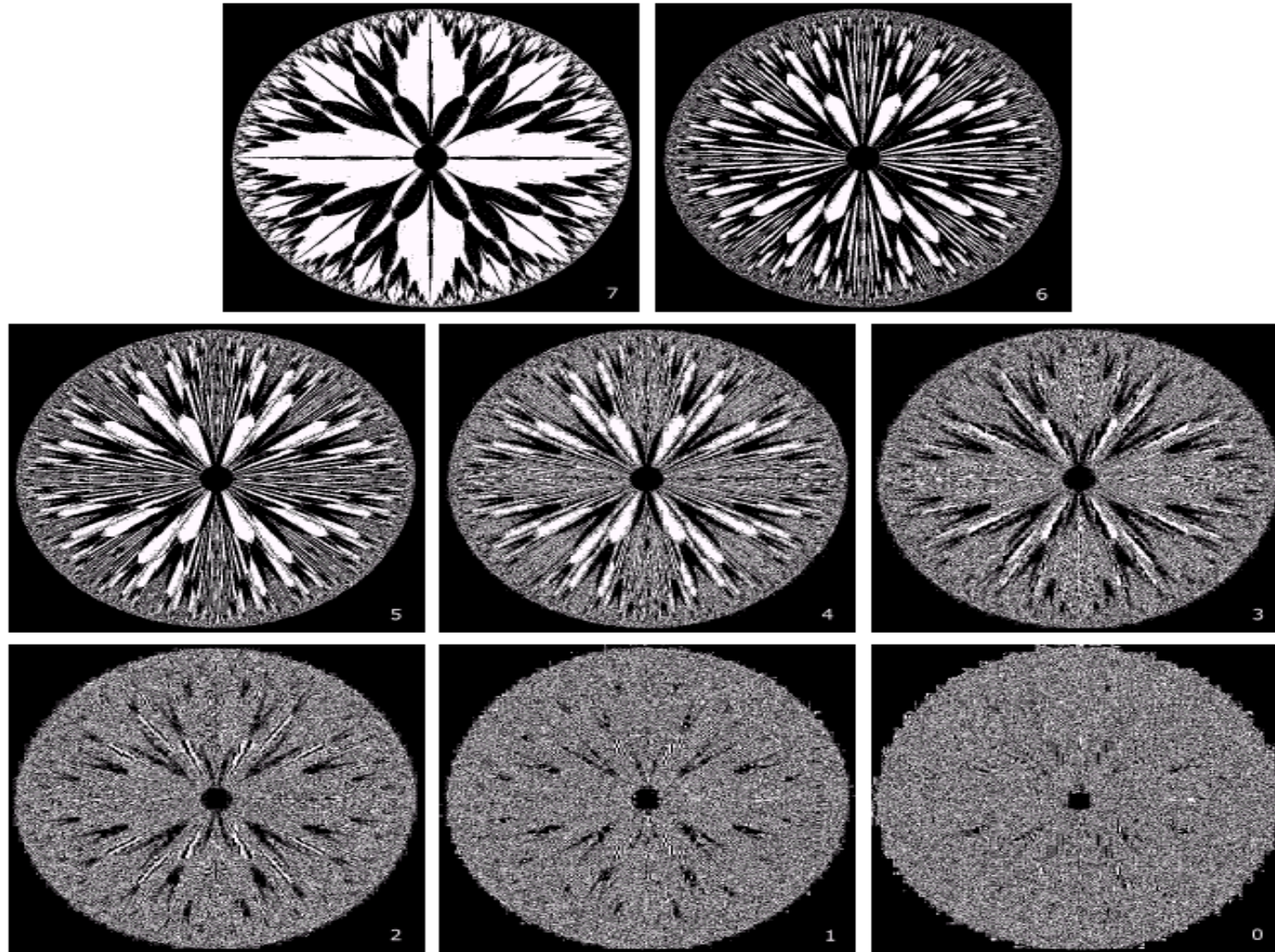


FIGURE 3.13 An 8-bit fractal image. (A fractal is an image generated from mathematical expressions). (Courtesy of Ms. Melissa D. Binde, Swarthmore College, Swarthmore, PA.)



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.



Original image



Bitplane 7



Bitplane 6



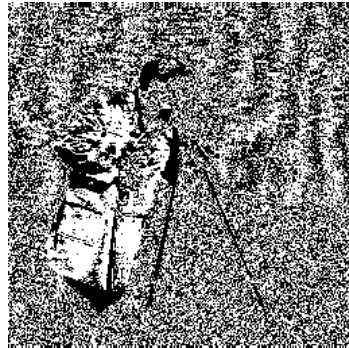
Bitplane 5



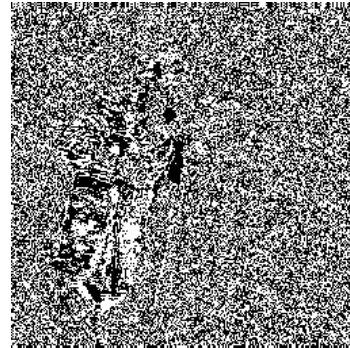
Bitplane 4



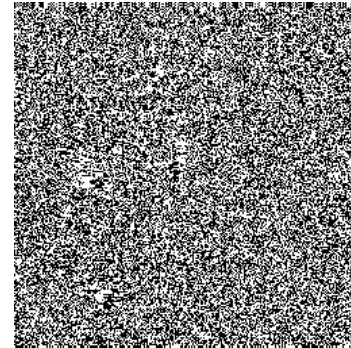
Bitplane 3



Bitplane 2



Bitplane 1



Bitplane 0

Tugas

Buatlah program *image enhancement*:

1. *Image brightening*
2. *Contrast stretching*
3. Transformasi log dan inverse log
4. Transformasi pangkat
5. *Gray-level slicing*
6. *Bit-plane slicing*