



# Operações Pontuais

Guillermo Cámara-Chávez

# Um modelo simples de formação de uma imagem



Para que a **imagem** de uma **cena real** possa ser **processada ou armazenda** na forma digital deve passar por **dois processos** de **discretização** denominados de:

- Amostragem
- Quantização

# Digitalização de uma cena (Sampling & Quantization)

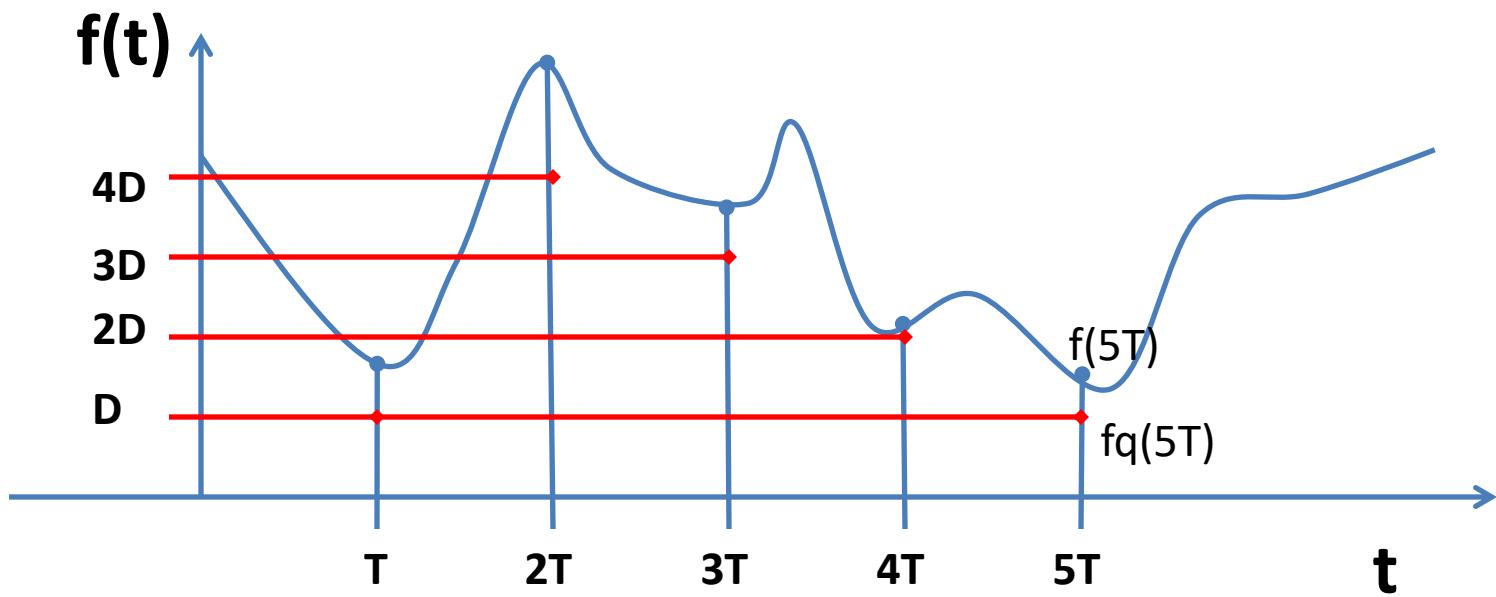


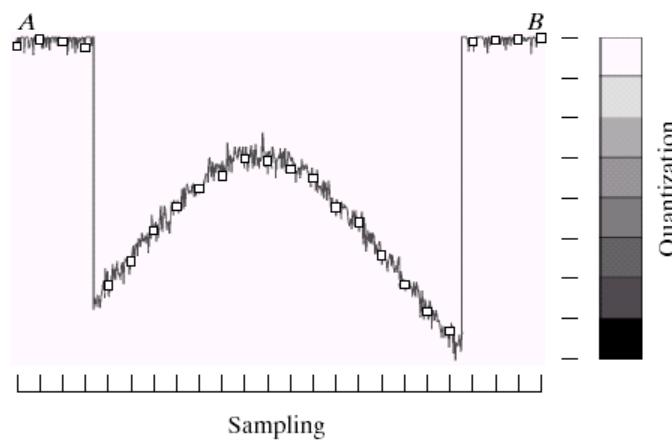
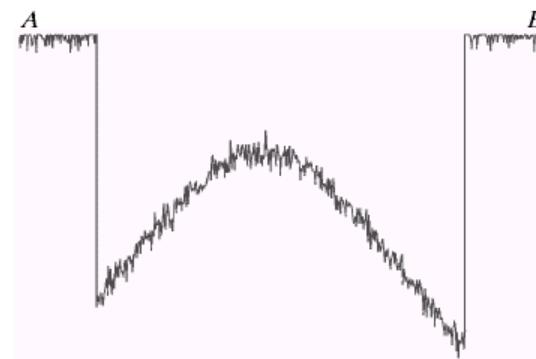
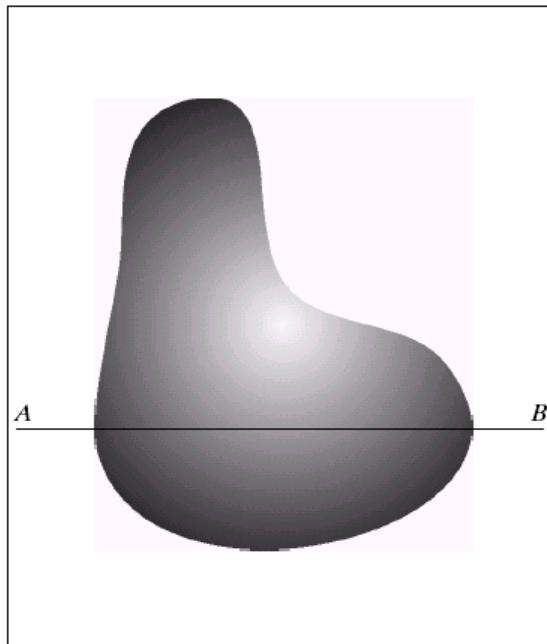
A  $f(x,y)$  é digitalizada : - **espacialmente**  
- **amplitude**

**Amostragem** : processo de discretização espacial das coordenadas  $x$  e  $y$  e dos valores de  $f$

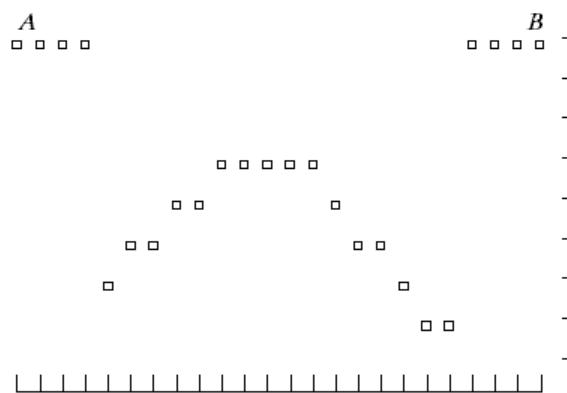
**Quantização** : processo de conversão das amplitudes (“brilho/intensidade”) de cada amostra em um conjunto de níveis ou tons de cinza (*gray level*).

# Digitalização de uma cena (Sampling & Quantization)





a b  
c d

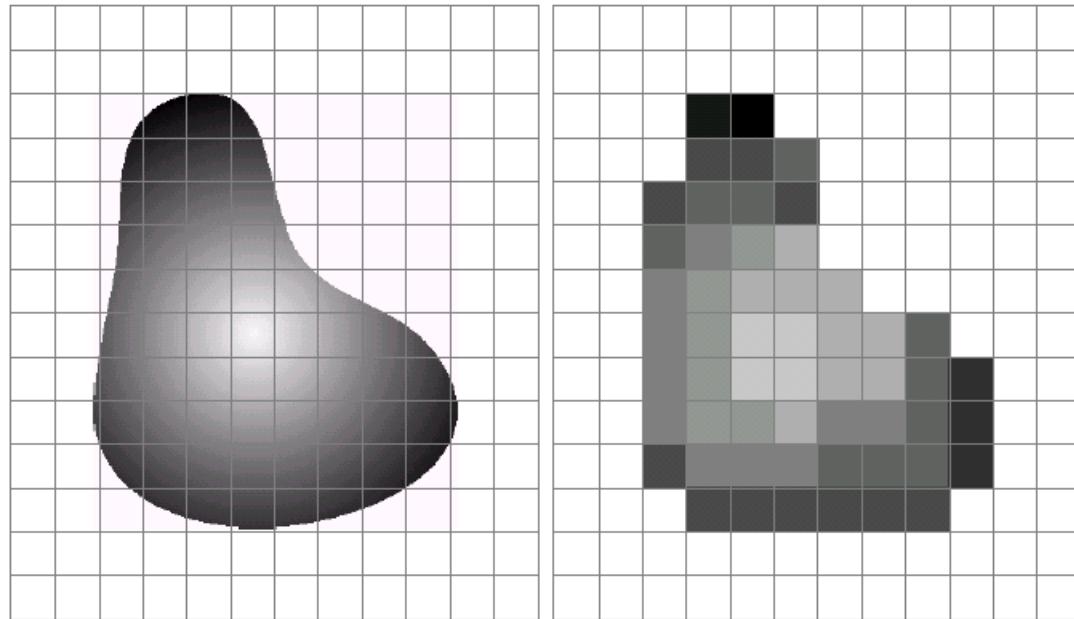


**FIGURE 2.16** Generating a digital image. (a) Continuous image. (b) A scan line from *A* to *B* in the continuous image, used to illustrate the concepts of sampling and quantization. (c) Sampling and quantization. (d) Digital scan line.

# Amostragem e Quantização



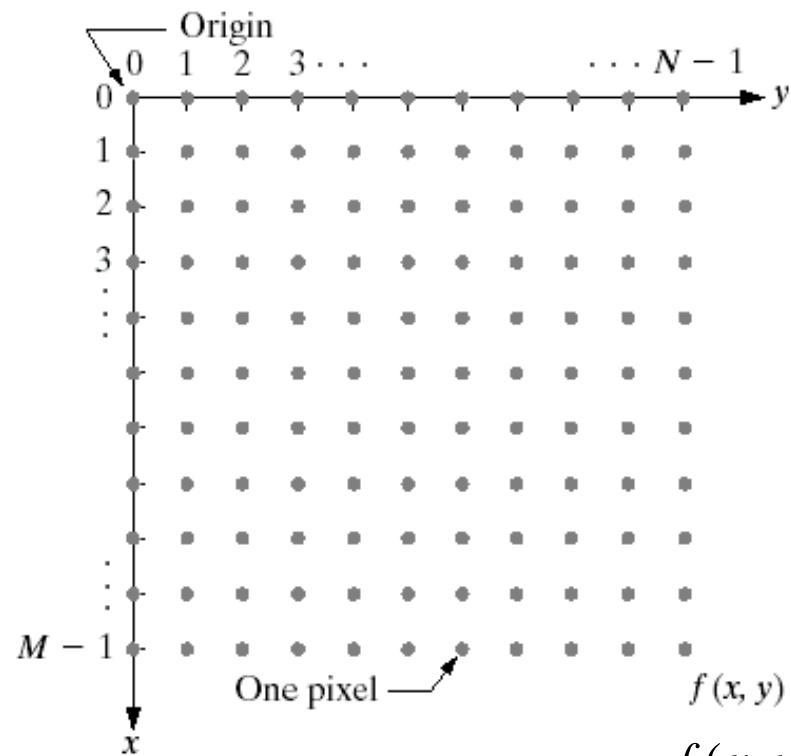
Continuous  
image project  
onto a sensor  
array.



a b

**FIGURE 2.17** (a) Continuos image projected onto a sensor array. (b) Result of image sampling and quantization.

O resultado do processo de amostragem e quantização é uma matriz de números reais, representado em uma matriz onde as coordenadas os elementos correspondem ao centro de cada sensor.



**FIGURE 2.18**  
Coordinate convention used in this book to represent digital images.

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0, N-1) \\ f(1,0) & f(1,1) & \cdots & f(1, N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(m-1, N-1) \end{bmatrix}$$

# Amostragem e Quantização



- **Amostragem:** refere-se ao número de pontos amostrados de uma imagem digitalizada (resolução)
- **Quantização:** quantidade de níveis de tons que pode ser atribuído a cada ponto digitalizados.

# Amostragem e Quantização (cont.)



- As imagens reais possuem um número ilimitado de cores ou tons.
- No processamento de imagens é necessário limitar os níveis de cores ou tons possíveis.

# Amostragem e Quantização (cont.)



- O sistema visual humana não é muito sensível a variações de intensidade (no máximo 30 diferentes tons de cinza)
- O computador consegue diferenciar qualquer quantidade de níveis de cinza.

# Dimensões da matriz e número de níveis de cinza



- O número  $L$ , resultante do processo de quantização, por razões do *hardware*, é um inteiro potência de 2:

$$L = 2^k$$

# Dimensões da matriz e número de níveis de cinza



- Os níveis de cinza são igualmente espaçados e inteiros no intervalo fechado:

$$\text{Nível de cinza} = [ 0 , L - 1 ]$$

- A **faixa de valores** resultante do processo de quantização é denominada de **faixa dinâmica**.
- Quando a faixa dinâmica **cobre toda a escala de cinza permitida**, diz-se que a imagem possui uma **faixa dinâmica ampla**.

# Resolução



128x128

64x64

32x32

16x16

# Quantização



256 tons



128 tons



64 tons

# Quantização (cont)



```
function nimg = Quantize(img, nbites)
```

**% quantiza em número de bits**

```
nbitsShift = 8 - nbites;
```

```
nimg = bitshift(img, - nbitsShift);
```

```
nimg = bitshift(nimg, nbitsShift);
```

```
nimg = uint8(Mapping(nimg));
```

# Quantização (cont)



```
function nimg = Mapping(img)
```

```
  nimg = double(img);
```

```
  mmin = min(nimg(:));
```

```
  nimg = nimg - mmin;
```

```
  mmax = max(nimg(:));
```

```
  nimg = (255 * nimg) / mmax;
```

# Histograma de imagem digital



- Indica o **percentual de pixels** que a imagem tem de um **determinado nível de cinza** ou cor.
- O **histograma** de uma imagem cinza é uma **função discreta**  $h(l)$  (vetor) que produz o **número de ocorrências** de **cada nível de cinza**  $0 \leq l \leq 2^b - 1$  na imagem

# Histograma de imagem digital



- Pseudocódigo para calcular o histograma

**Início**

$h(f(x,y)) = 0$  ; {zera contadores de nív. de cinza }

**Para** cada valor  $f(x,y)$  **faça**

$h(f(x,y)) = h(f(x,y)) + 1$

**Fim-Para**

**Fim**

# Histograma de imagem digital



```
function mhist = myHist(img)
% histograma para imagem em níveis de cinza
[row, col] = size(img);
mhist = zeros(1, 256);
for i = 1 : row
    for j = 1: col
        mhist(img(i,j)+1) = mhist(img(i,j)+1) + 1;
    end
end
```

# Histograma de imagem digital



- Através da função **imhist** podemos calcular o histograma de uma imagem em níveis de cinza

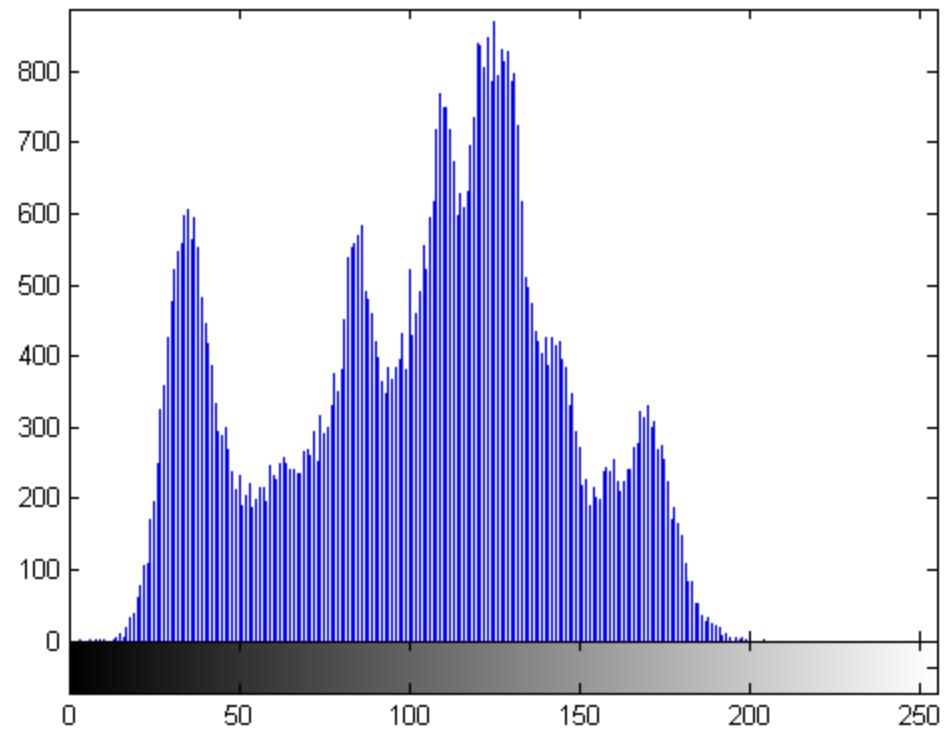
```
img = imread('lenna.png');
```

```
nbins = 256;
```

```
histImg = imhist(img, nbins);
```

```
plot(histImg);
```

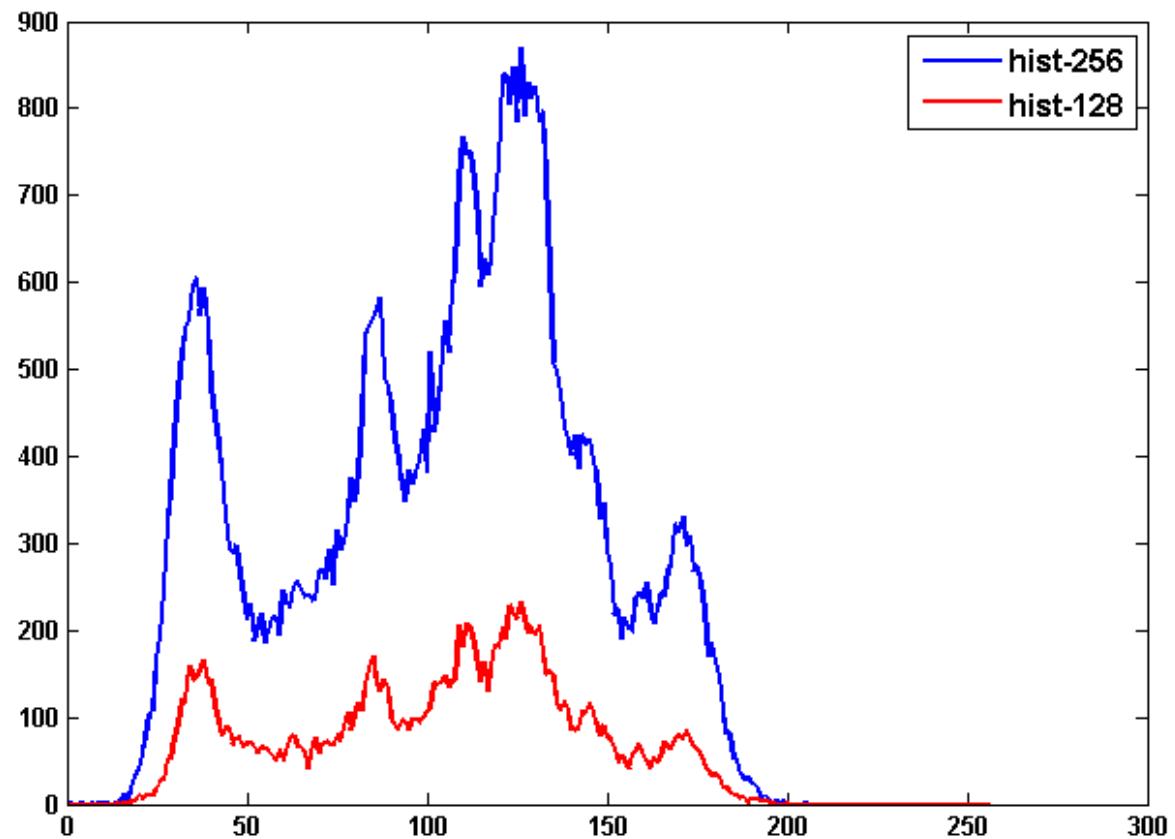
# Histograma de imagem digital



# Histograma de imagem digital



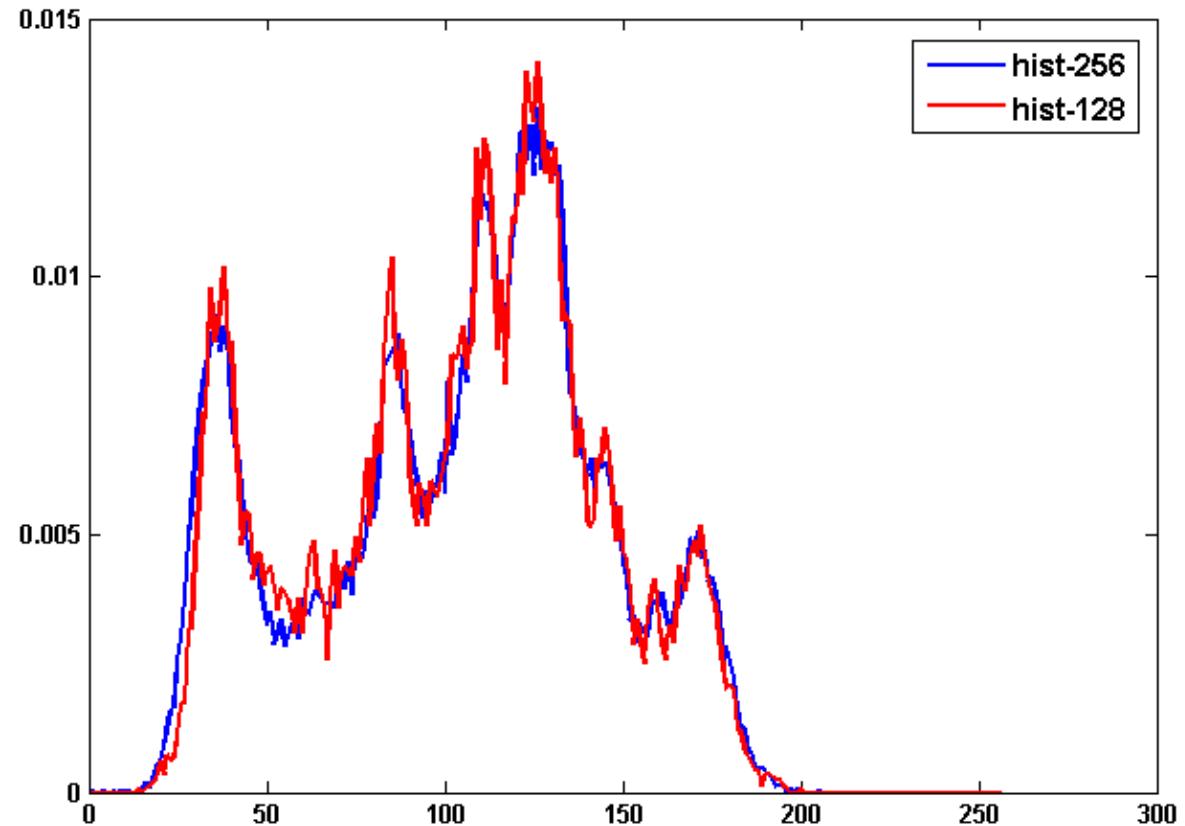
Histogramas da  
imagem lenna em  
resoluções de  
256x256 e 128x128



# Histograma de imagem digital



O histograma  
normalizado  
 $h(l)/|l|$   
representa a  
distribuição de  
probabilidade dos  
valores dos pixels,  
sendo que  $l$  é a  
imagem e  $|l|$  a  
cardinalidade



# Histograma de imagem digital



- Sua informação é **invariante** com as operações de **rotação** e **translação**.
- Podemos considerar um **histograma para cada banda** espectral ou um **histograma 3-D**, por exemplo, referente às componentes RGB de uma **imagem colorida**.

# Histograma de imagem digital



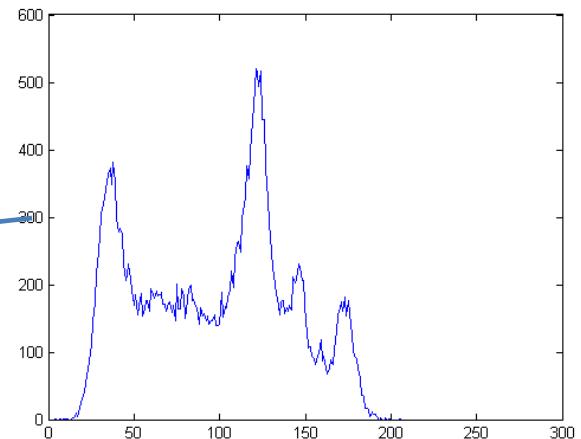
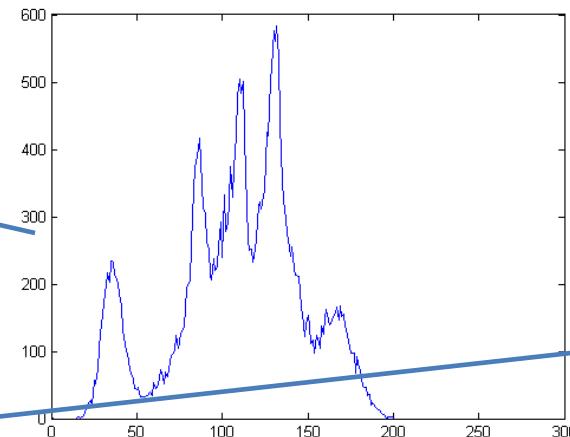
## Propiedades

1. A **informação espacial é perdida**, apenas contém a **quantidade de pixels** com um determinado valor
2. A *pdf (probability density function)* da imagem é obtida **dividindo o histograma pela sua área**.

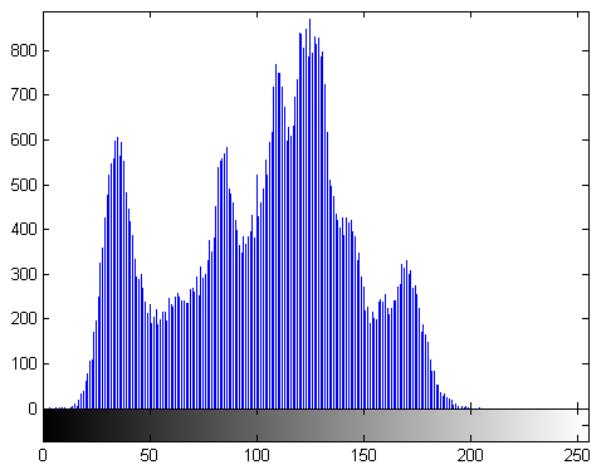
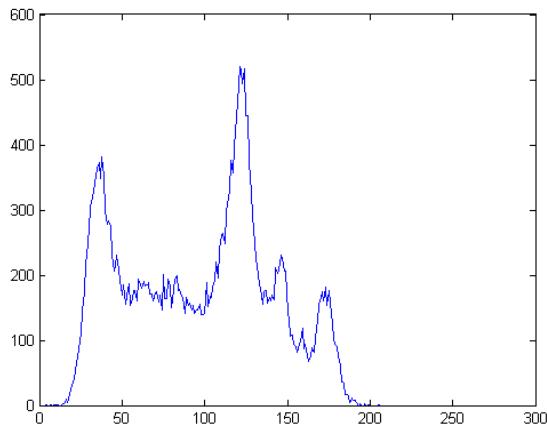
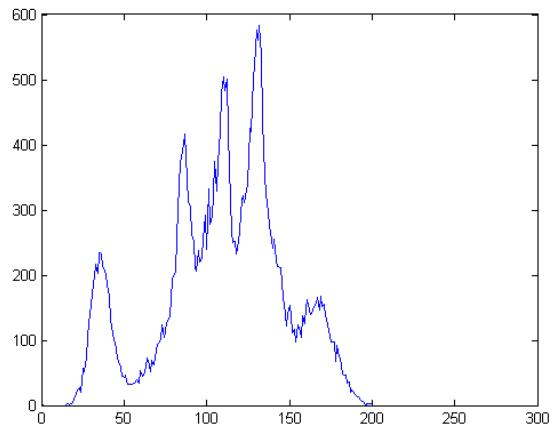
# Histograma de imagem digital



3. Se a imagem for construída por **duas ou mais regiões disjuntas**, o histograma pode ser obtido através da **soma dos histogramas** destas regiões



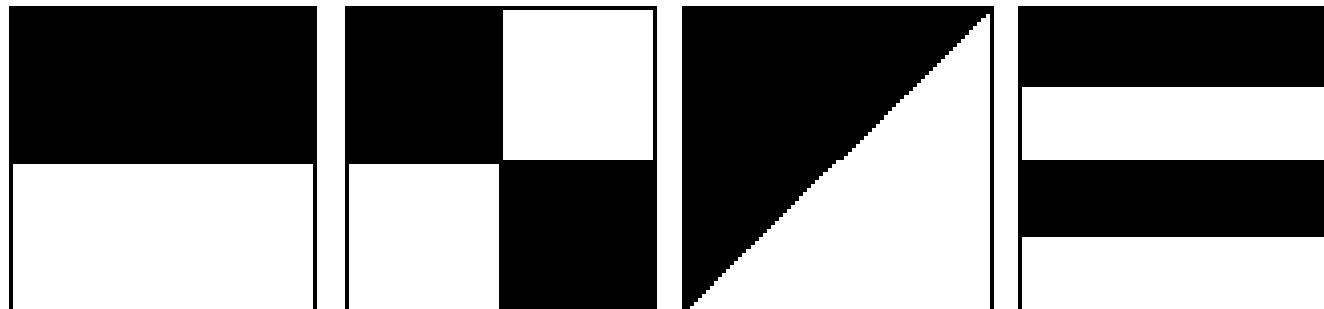
# Histograma de imagem digital



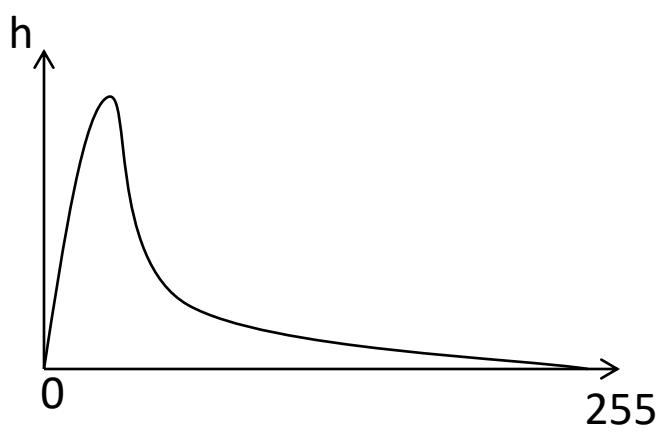
# Histograma de imagem digital



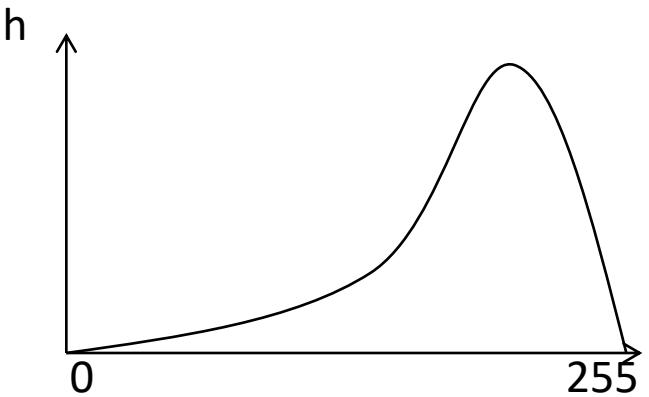
4. Uma imagem possui apenas um histograma, porém um histograma pode pertencer a mais de uma imagem



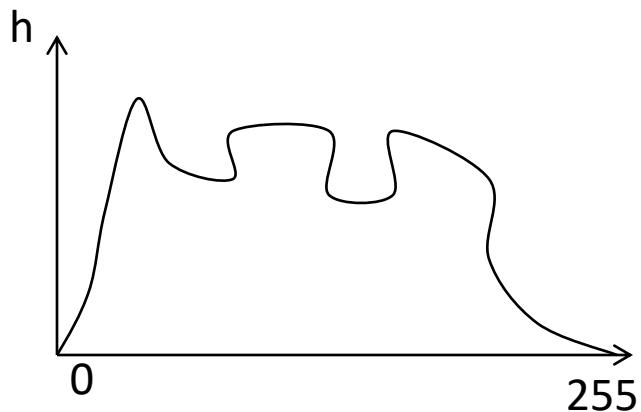
# Histograma de imagem digital



↑  
Imagen escura



↑  
Imagen clara



# Transformações Radiométricas



- Operadores ponto a ponto
- Independem da localização dos pixels na imagem.
- Em termos de implementação, podem ser representadas por *look-up-tables (LUT)*.
- Transformam um pixel de nível de cinza  $g_i$  em um nível de cinza  $g_f$ .

# Uma transformação radiométrica $r$ :



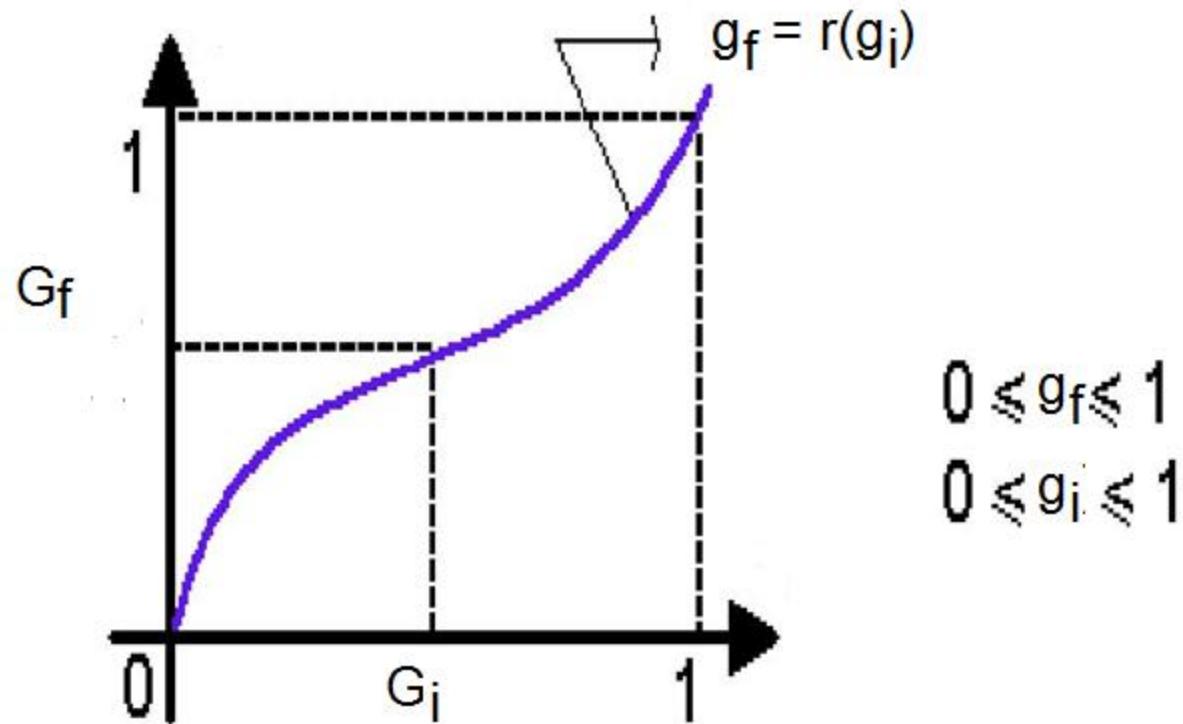
- É uma aplicação  $I(G_i) \rightarrow F(G_f)$ , tal que

$$G_i = [0, 1, \dots, N_i],$$

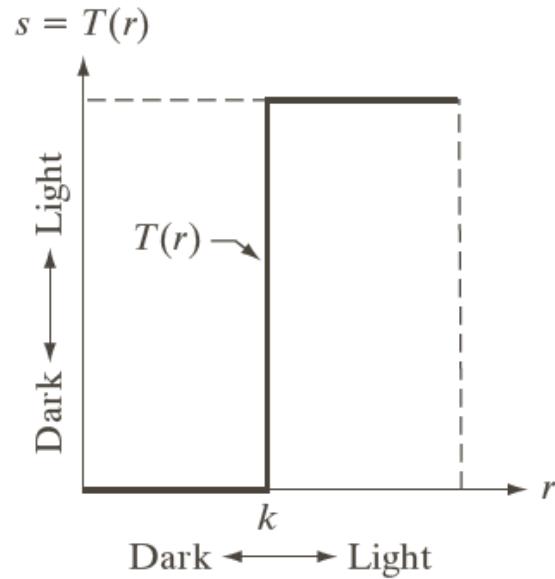
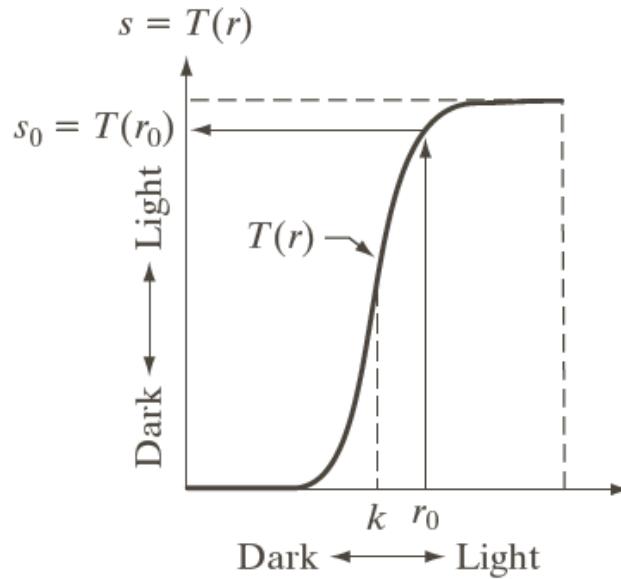
$$G_f = [0, 1, \dots, N_f] \quad \text{e}$$

$$\forall g_i \in G_i, \exists g_f \in G_f, g_f = r(g_i)$$

# Uma transformação radiométrica $r$ :



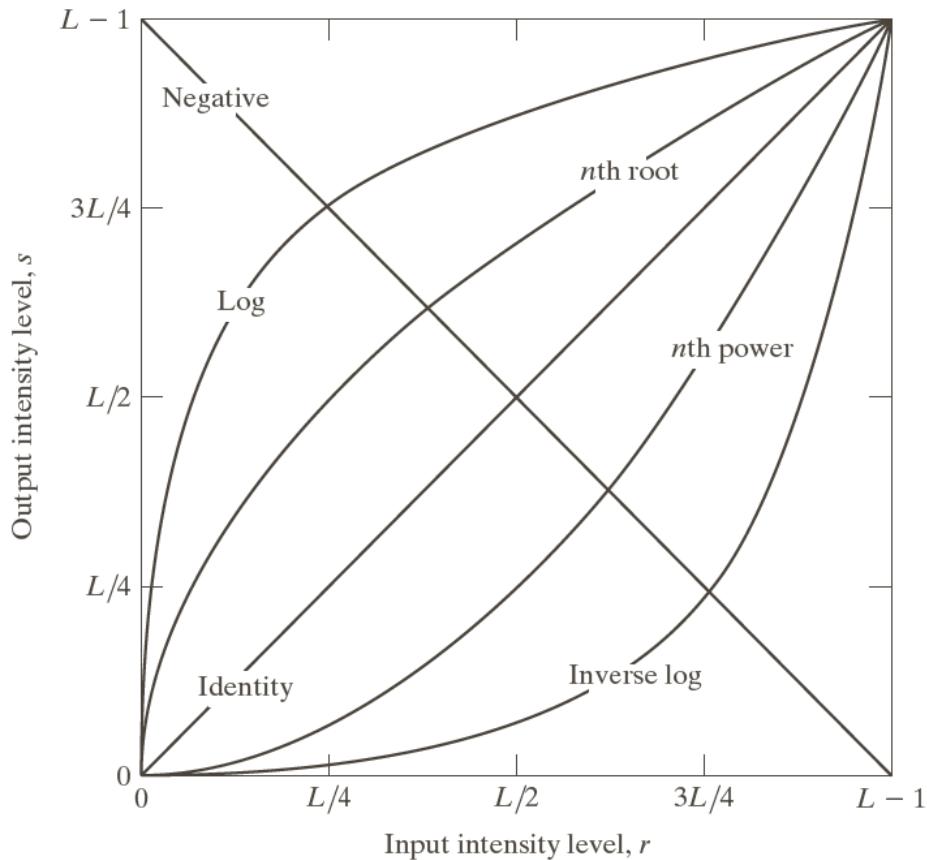
# Uma transformação radiométrica $r$ :



a | b

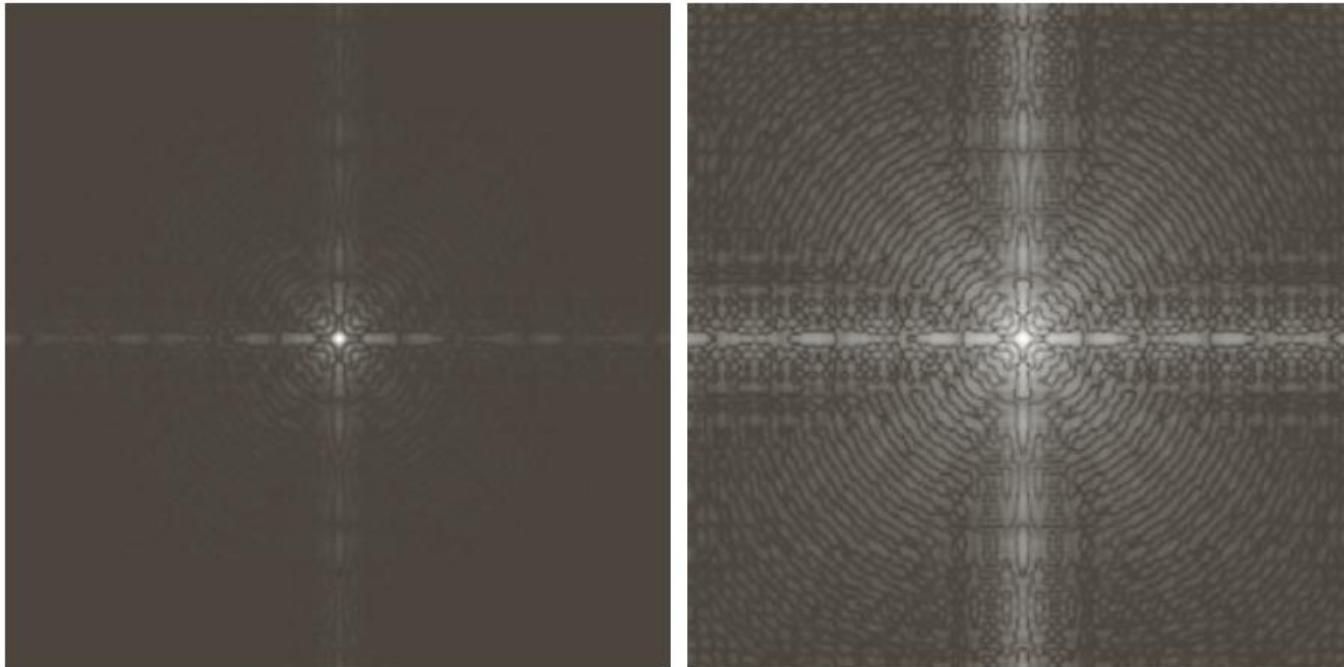
**FIGURE 3.2**  
Intensity transformation functions.  
(a) Contrast-stretching function.  
(b) Thresholding function.

# Uma transformação radiométrica $r$ :



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

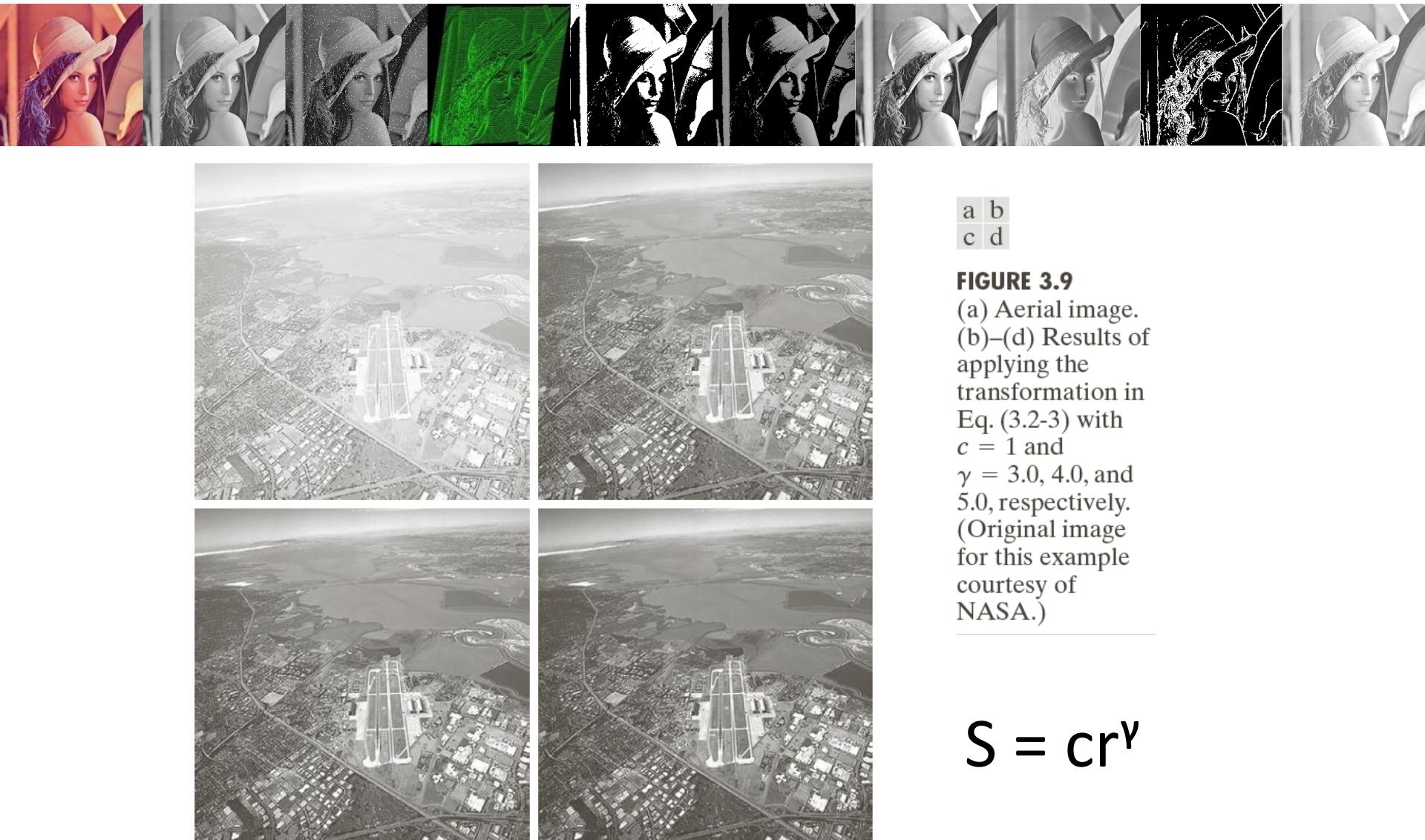
# Uma transformação radiométrica r :



a b

**FIGURE 3.5**  
(a) Fourier spectrum.  
(b) Result of applying the log transformation in Eq. (3.2-2) with  $c = 1$ .

# Uma transformação radiométrica r :

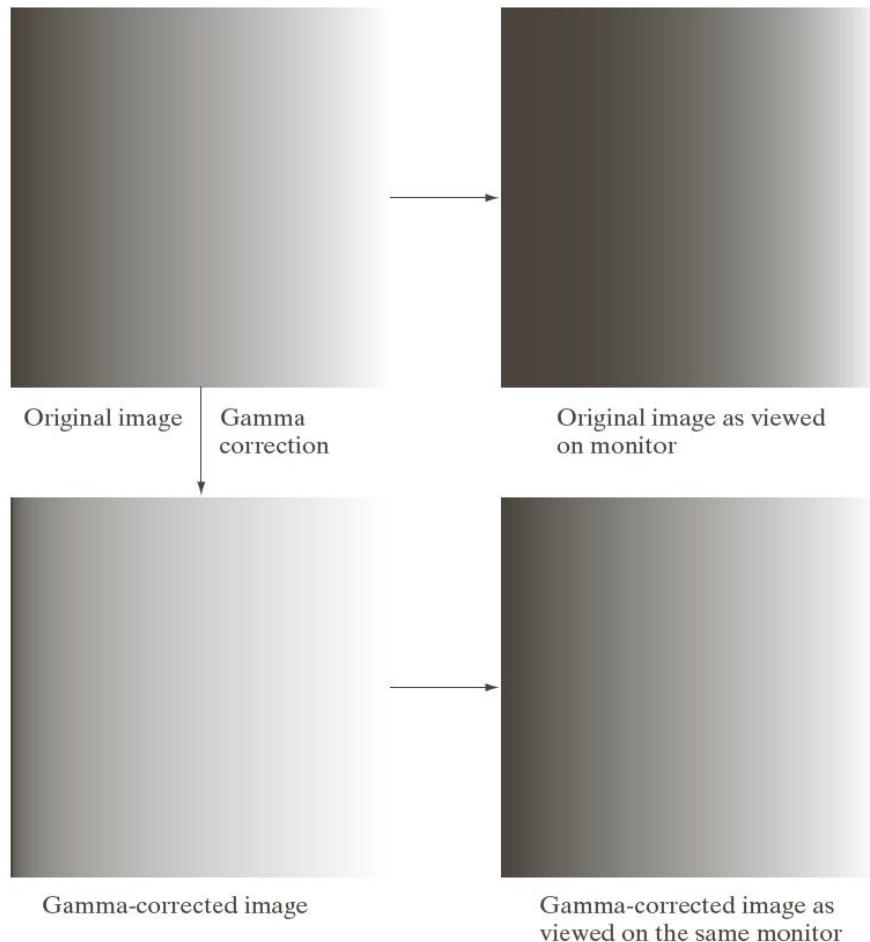


**FIGURE 3.9**

(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.)

$$S = cr^\gamma$$

# Uma transformação radiométrica r :



a  
b  
c  
d

**FIGURE 3.7**

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

# Gamma correction



# Gamma correction



$$\lambda = 1/2$$



$$\lambda = 1/4$$

# Transformações Lineares



- Sejam  $[l_1, l_2]$ ,  $l_1 \leq l_2$ , e  $[k_1, k_2]$  dois intervalos de cinza no conjunto de valores de  $I$  e  $J$ .
- A transformação linear (*stretching linear*) é definida por:

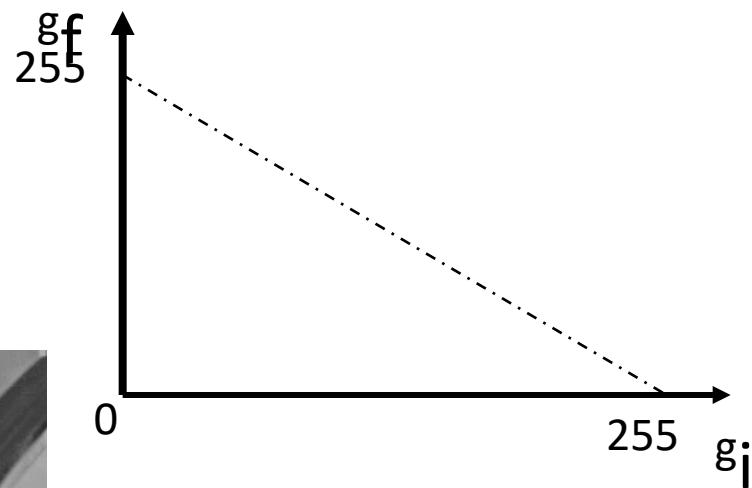
$$k = \begin{cases} k_1, & \text{se } l < l_1, \\ \frac{(k_2 - k_1)}{(l_2 - l_1)}(l - l_1) + k_1, & \text{se } l_1 \leq l < l_2, \\ k_2, & \text{se } l \geq l_2. \end{cases}$$

# Transformações Lineares

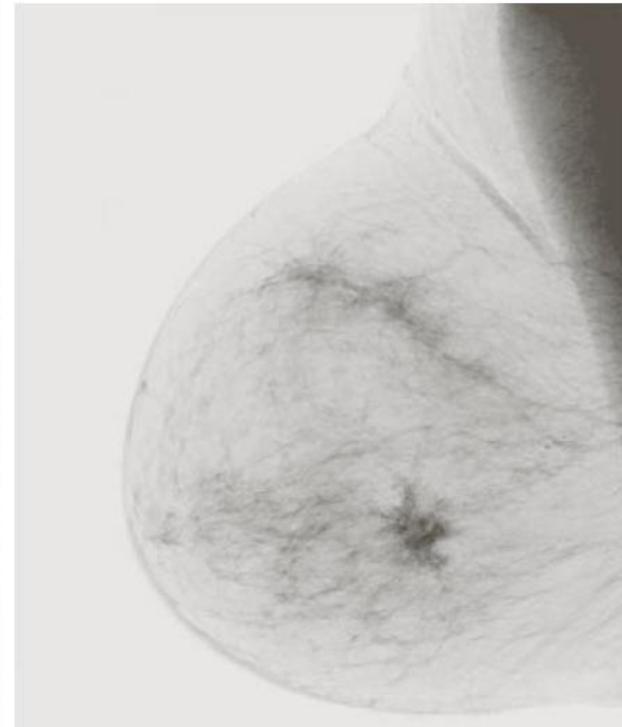
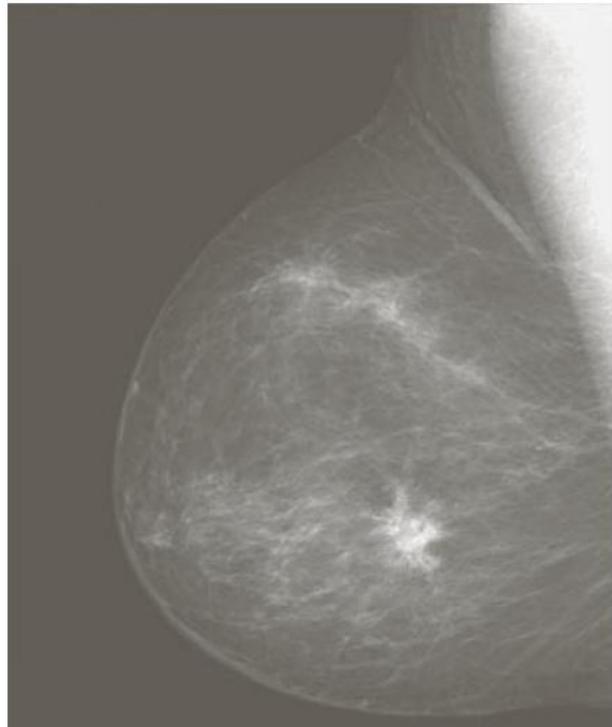


- Normalização em  $[0, H]$  (e.g.,  $H = 255$ ):  $k_2 = H$ ,  $k_1 = 0$ ,  $I_1 = I_{min}$ , e  $I_2 = I_{max}$ , onde  $I_{min}$  e  $I_{max}$  são os valores mínimo e máximo da  $I$
- Negativo:  $k_2 = I_{min}$ ,  $k_1 = I_{max}$ ,  $I_1 = I_{min}$ , e  $I_2 = I_{max}$
- Limiarização (*thresholding*):  $k_2 = H$ ,  $k_1 = 0$  e  $I_1 = I_2$

# Negativo



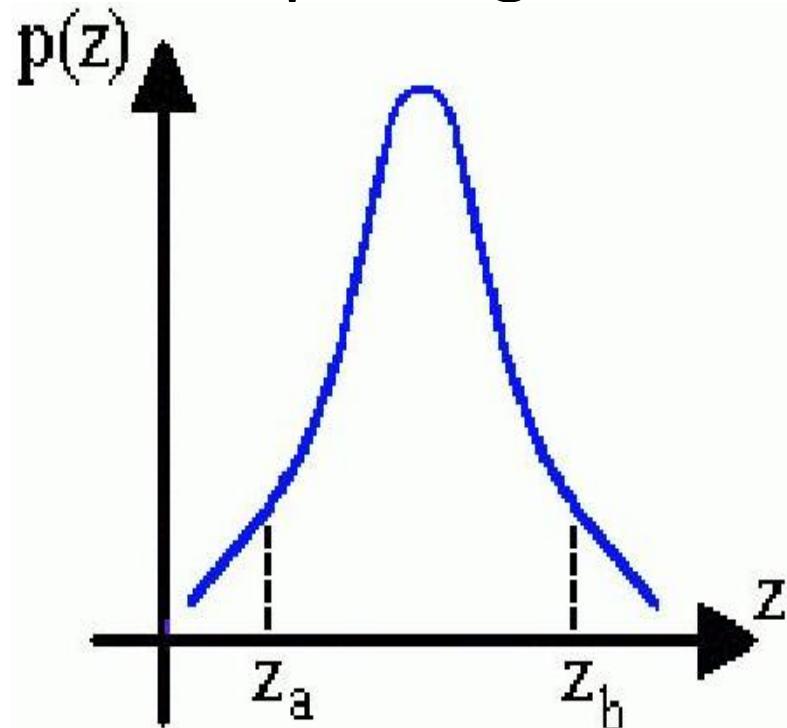
# Negativo



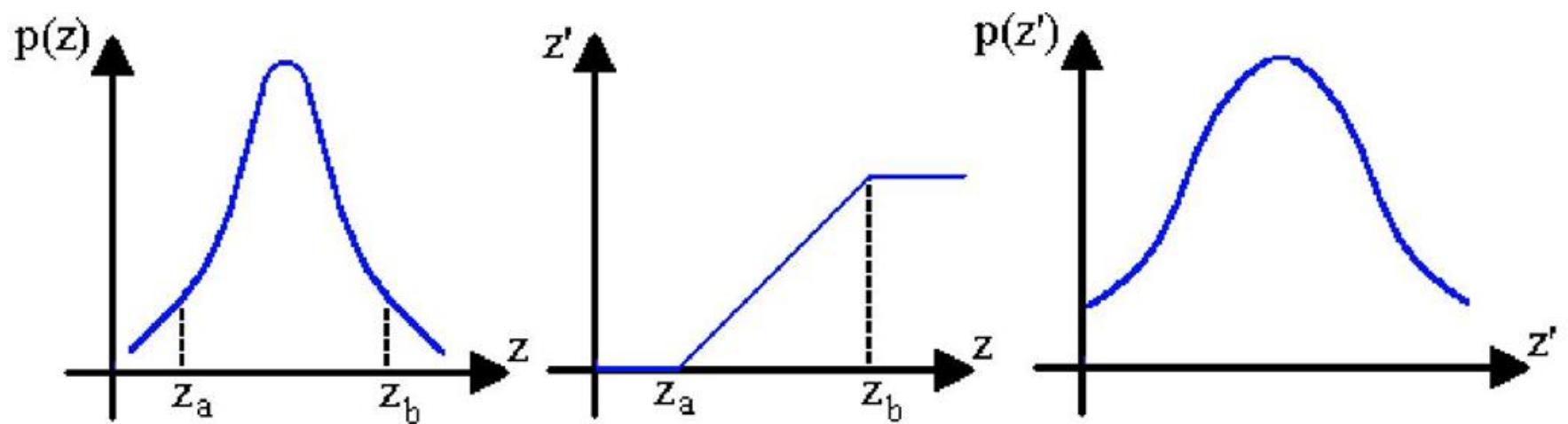
# Expansão de contraste linear



- Adequada para imagens com histogramas de forma gaussiana ou quase-gaussiana



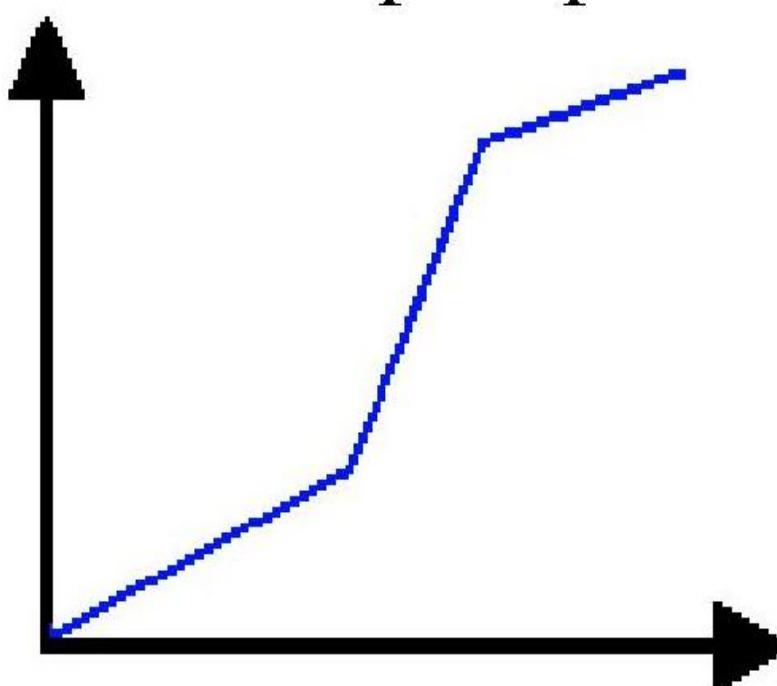
# Expansão de contraste linear



# Expansão de contraste linear



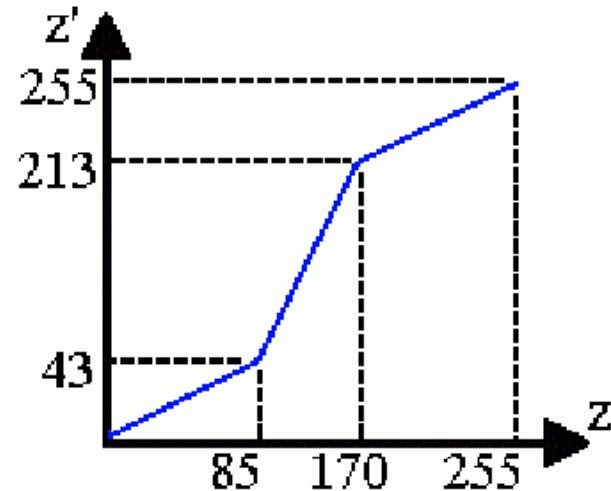
- Para imagens com histogramas bimodais, fazer expansão linear por partes



# Expansão de contraste linear



- Compressão e expansão



$$z' = \begin{cases} z / 2 , & \text{se } z \leq 85 \\ 2z - 127 , & \text{se } 85 < z < 170 \\ (z / 2) + 128 , & \text{se } z \geq 170 \end{cases} \quad [0 , 255 ]$$

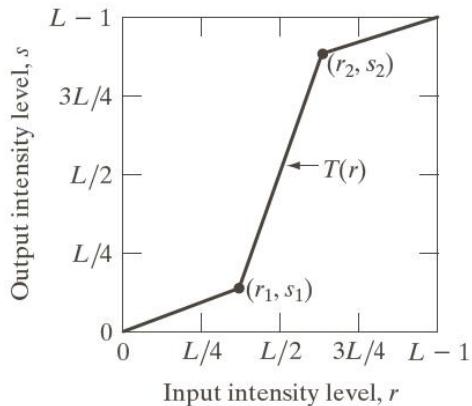
# Expansão de contraste linear



# Expansão de contraste linear



# Expansão de contraste linear



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.)

# Constraste e Brilho



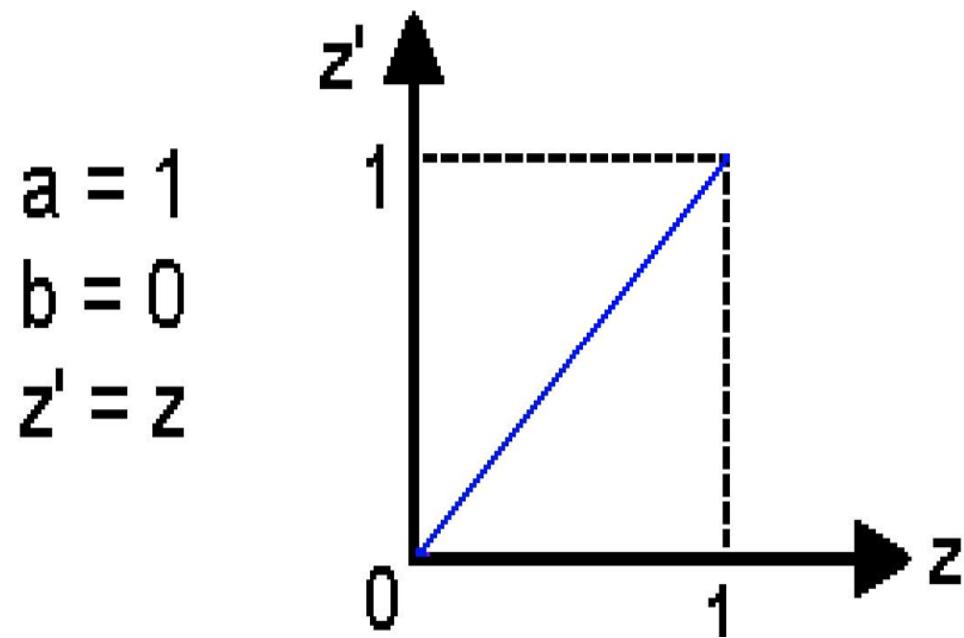
Lineares:

- $z' = a.z + b$ , onde
  - a: contraste e
  - b: brilho

# Constraste e Brilho



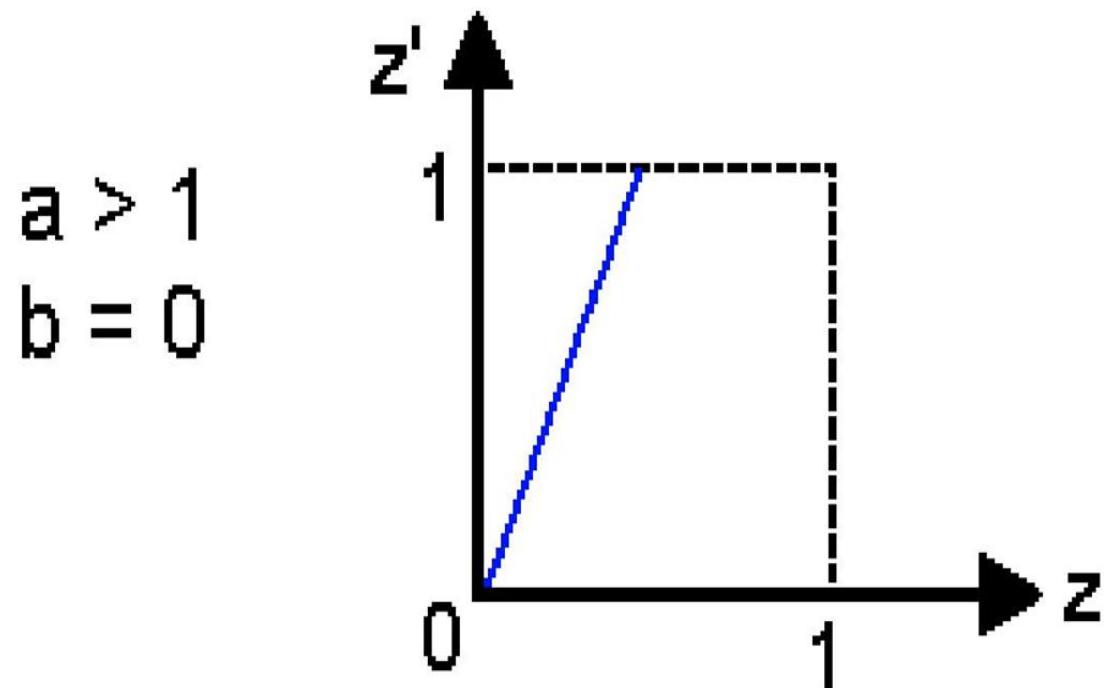
- Operação identidade



# Constraste e Brilho



- Expansão do contraste



# Constraste e Brilho



- Exemplo  $z' = 1.5 * z$

```
Img = imread('lenna.png');
```

```
nimg = 1.5 * double(img);
```

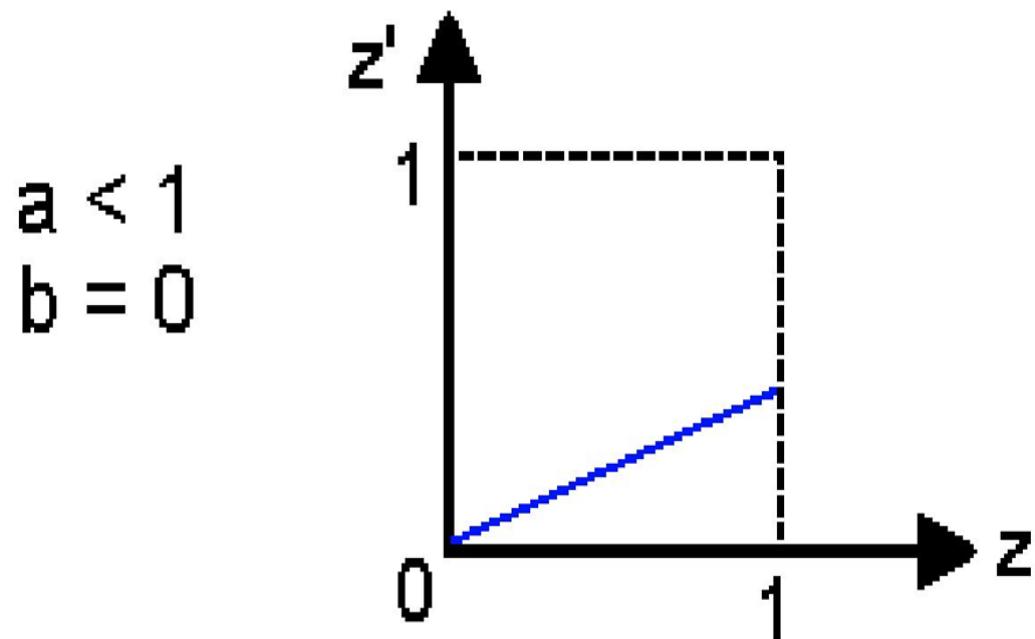
```
imshow(uint8(nimg));
```



# Constraste e Brilho



- Compressão do contraste



# Constraste e Brilho

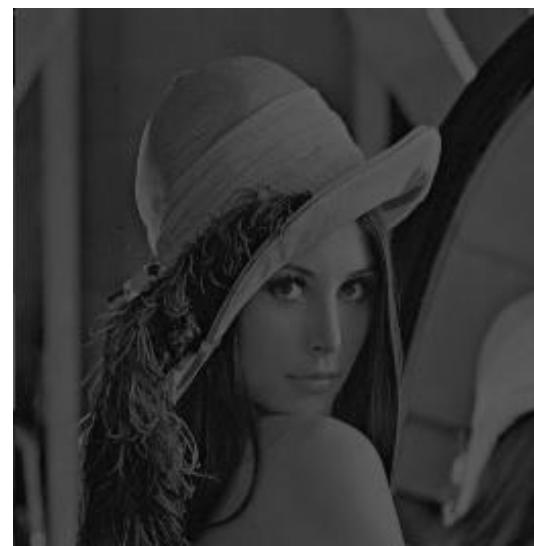


- Ejemplo  $z' = 0.5 * z$

```
img = imread('lenna.png');
```

```
nimg = double(img) * 0.5;
```

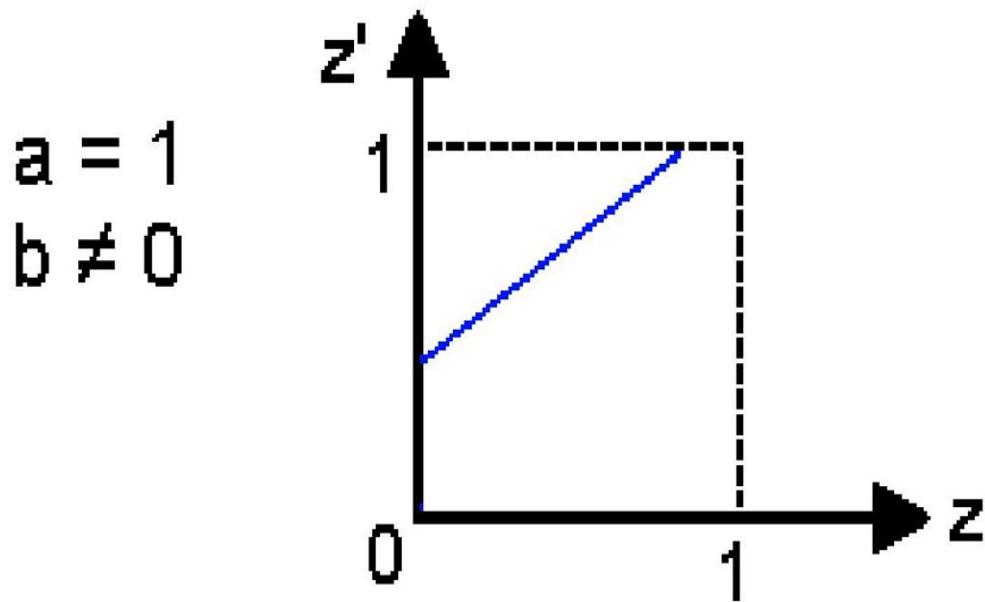
```
imshow(uint8(nimg));
```



# Constraste e Brilho



- Operação de escala



# Constraste e Brilho



- Ejemplo  $z' = z + 80$

```
img = imread('lenna.png');
```

```
nimg = double(img) + 80;
```

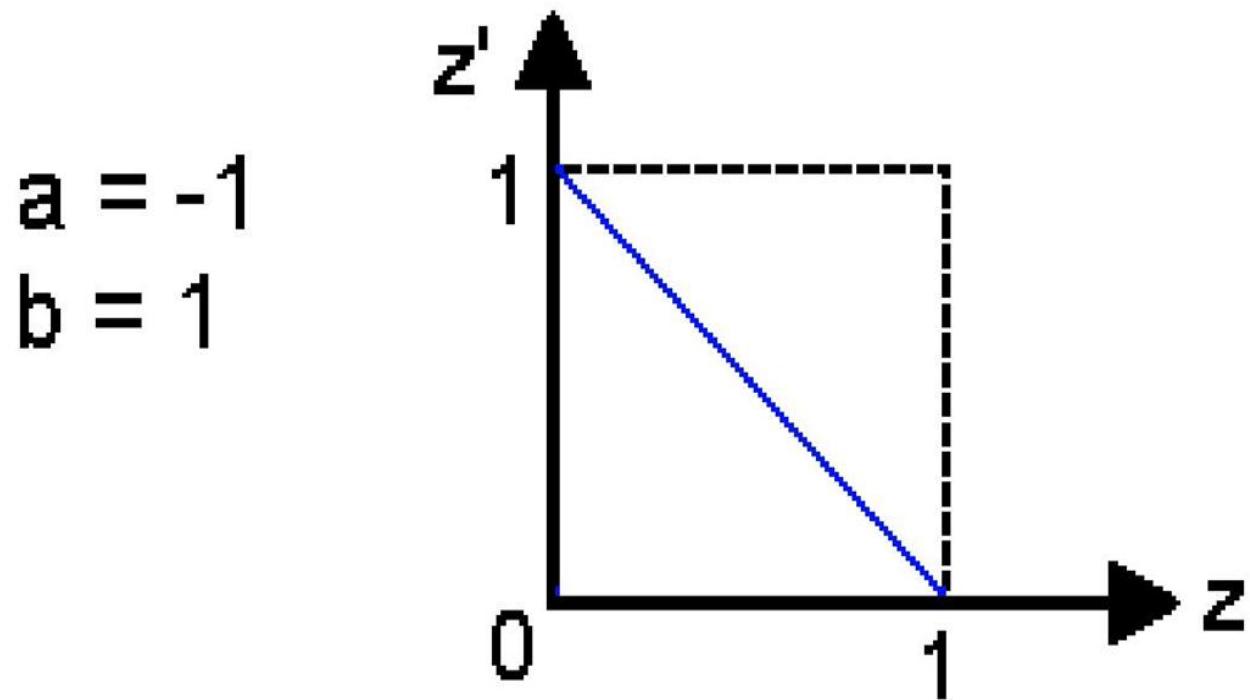
```
imshow(uint8(nimg));
```



# Constraste e Brilho



- Imagem negativa



# Constraste e Brilho



- Ejemplo  $z' = -z + 255$

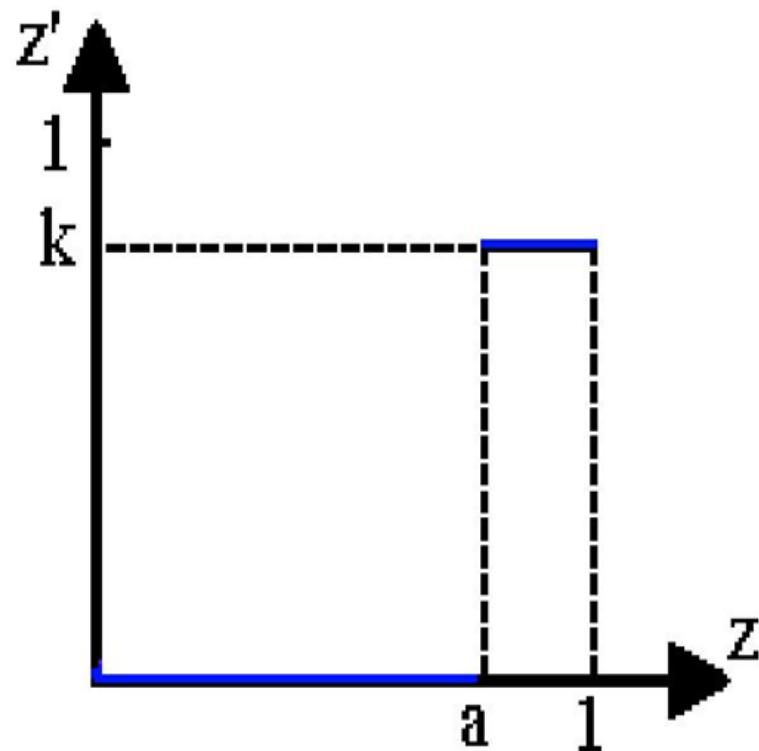
```
img = imread('lena.gif');  
nimg = 255 - double(img);  
imshow(uint8(nimg));
```



# Aplicação de limiar



- 2 níveis



$$z' = \begin{cases} 0, & \text{se } z \leq a \\ k, & \text{se } z > a \end{cases}$$

# Aplicação de limiar



## Algoritmo

```
function new_img = Binariza(img, thresh)
[fil, col] = size(img);
new_img = zeros(fil, col);
ind = img >= thresh;
new_img(ind) = 255;
```

# Aplicação de limiar



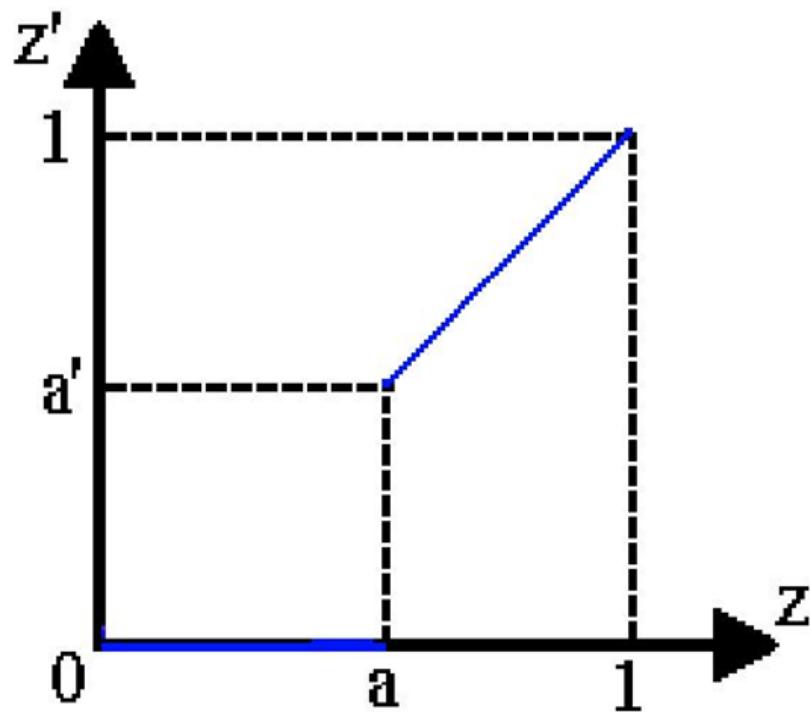
Exemplo : limiar = 127



# Aplicação de limiar



- Limiar com fundo



$$z' = \begin{cases} 0, & \text{se } z \leq a \\ z, & \text{se } z > a \end{cases}$$

# Aplicação de limiar



## Algoritmo

```
function new_img = BinarizaFondo(img,  
thresh)  
[fil, col] = size(img);  
new_img = zeros(fil, col);  
ind = img >= thresh;  
new_img(ind) = img(ind);
```

# Aplicação de limiar



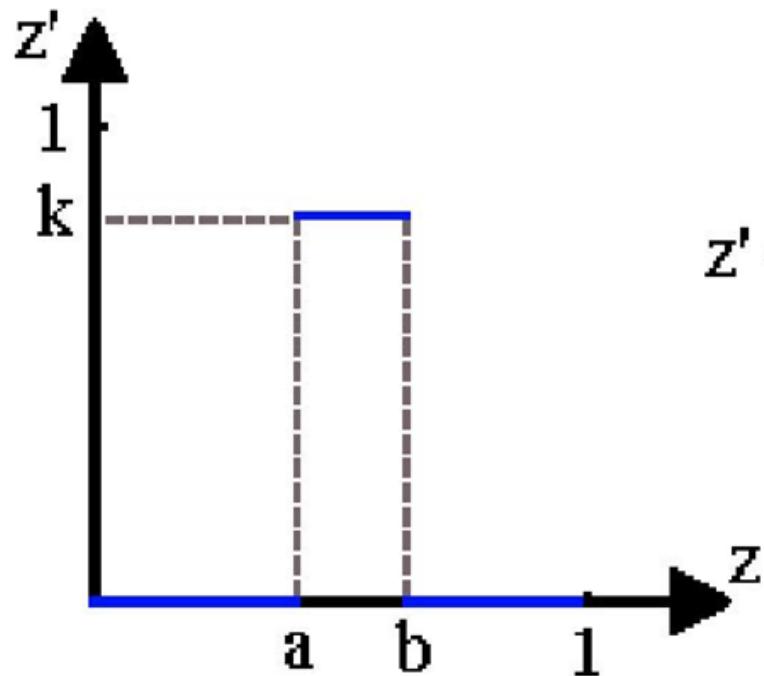
- Exemplo:  $a = 127$



# Aplicação de limiar



- Fatiamento por plano



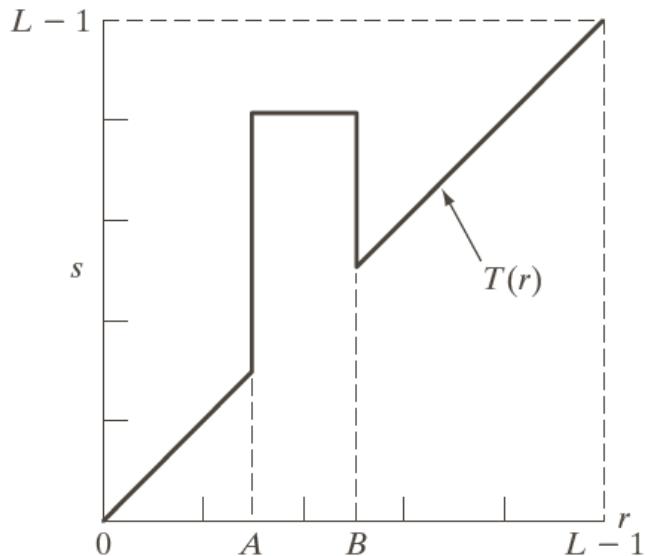
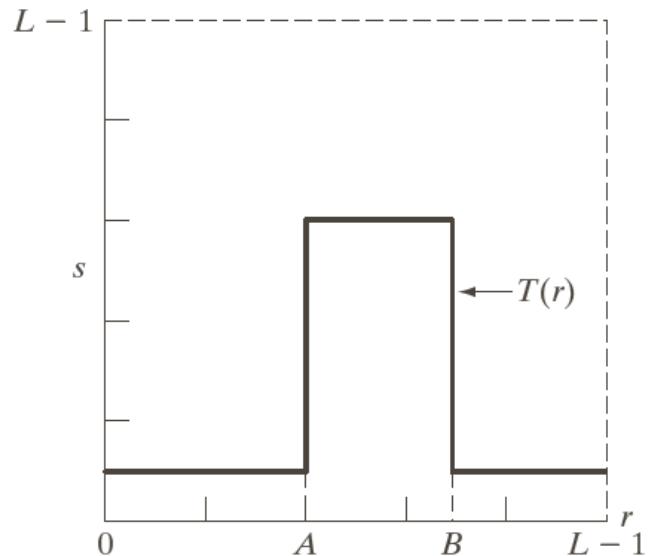
$$z' = \begin{cases} 0, & \text{se } z \leq a \text{ ou } z \geq b \\ k, & \text{se } a < z < b \end{cases}$$

# Aplicação de limiar

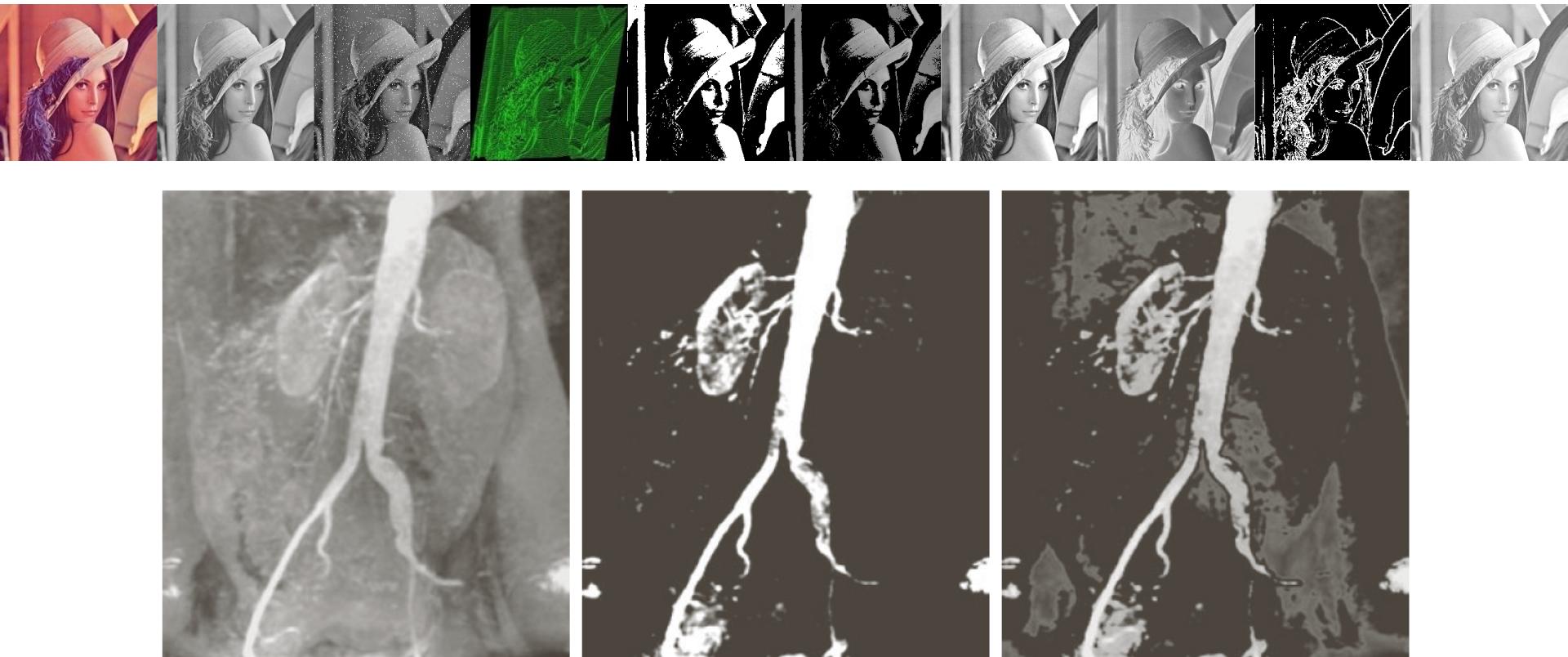


a b

**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.



# Aplicação de limiar



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.)

# Aplicação de limiar



## Algoritmo

```
function new_img = CortePlano(img,  
a,b,k)  
[fil, col] = size(img);  
new_img = zeros(fil, col);  
ind = find(img > a & img < b);  
new_img(ind) = k;
```

# Aplicação de limiar



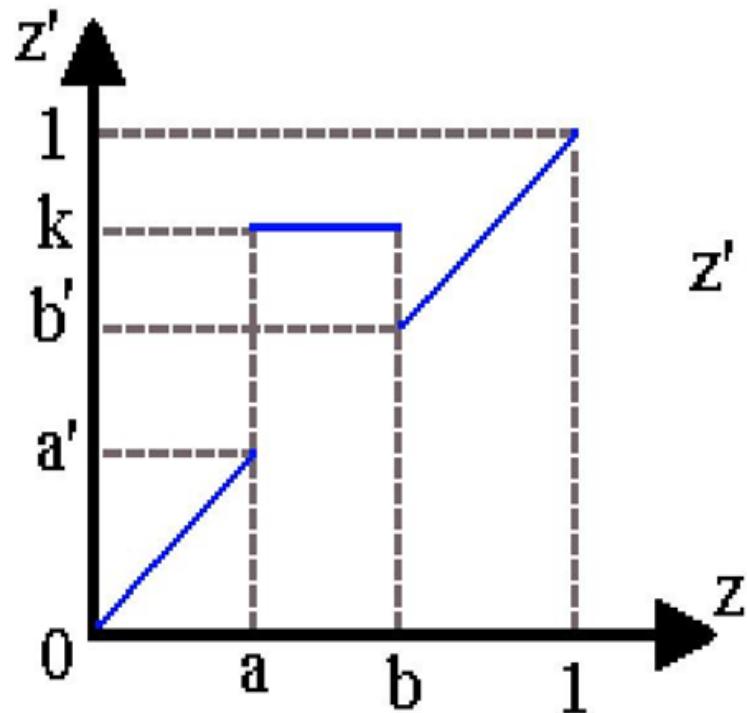
- Ejemplo:  $a = 150$ ,  $b = 200$ ,  $k = 255$



# Aplicação de limiar



- Fatiamento por plano com fundo

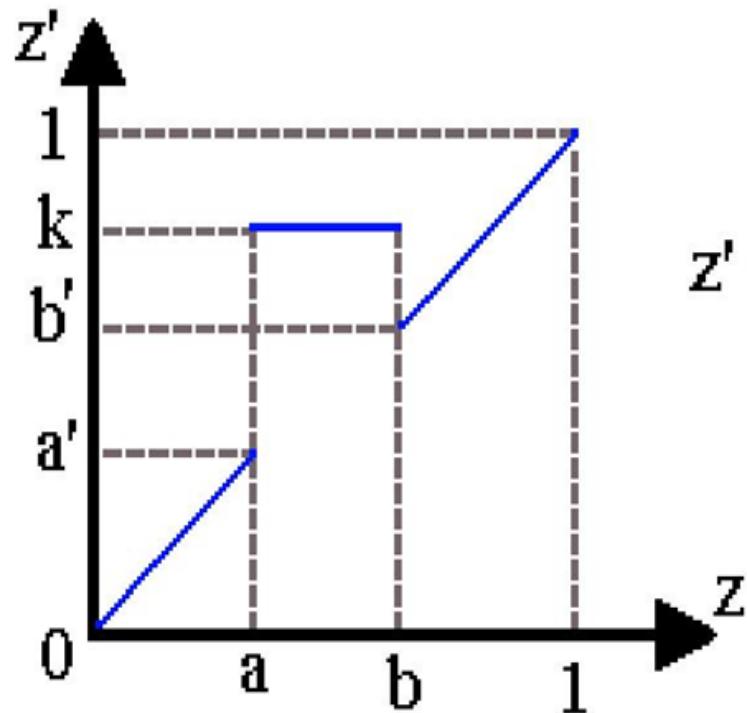


$$z' = \begin{cases} z, & \text{se } z \leq a \text{ ou } z \geq b \\ k, & \text{se } a < z < b \end{cases}$$

# Aplicação de limiar



- Fatiamento por plano com fundo



$$z' = \begin{cases} z, & \text{se } z \leq a \text{ ou } z \geq b \\ k, & \text{se } a < z < b \end{cases}$$

# Transformação Exponencial



- Pode ser definida por:

$$k = l_{\max} \exp\left(\frac{l - l_{\min}}{l_{\max} - l_{\min}}\right) - l_{\max} \text{ e}$$

$$k = H \exp\left(\frac{-(l - \mu)^2}{2\sigma^2}\right).$$

- O primeiro caso aumenta o contraste no intervalo  $[l_{\min}, l_{\max}]$ .
- O segundo aumenta o contraste em relação a um valor  $\mu$

# Transformação Logaritmica

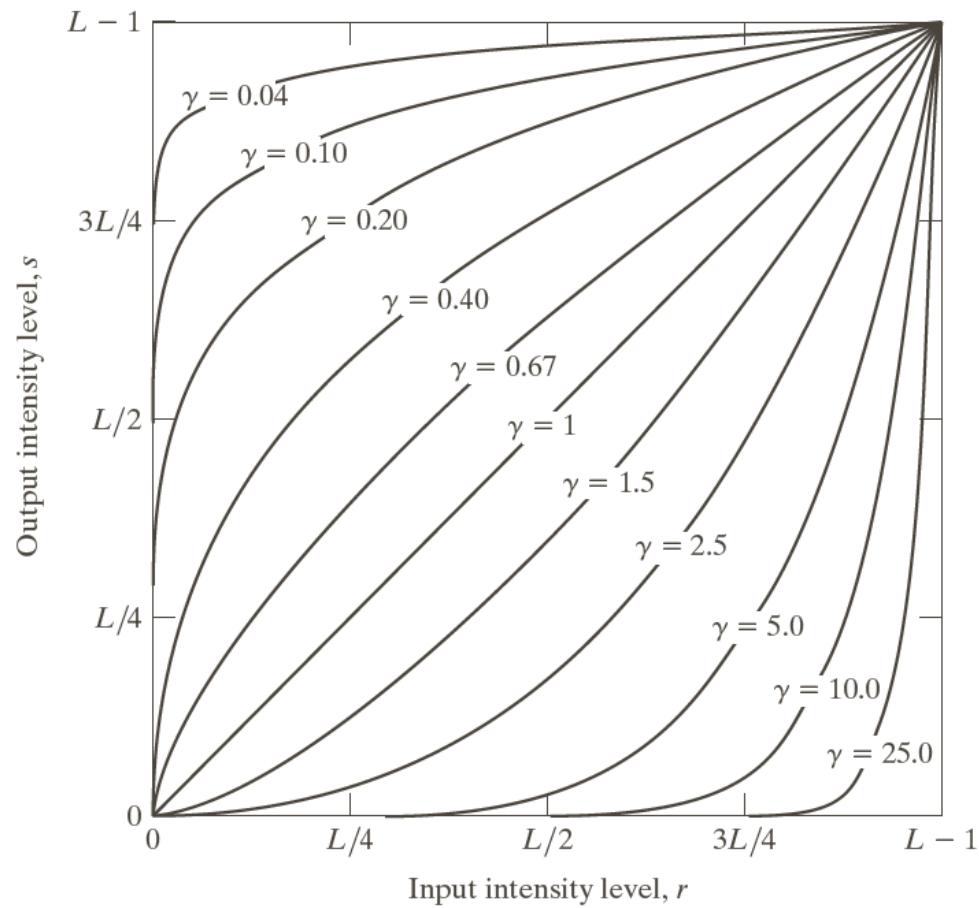


- Reduz a dinâmica da imagem (intervalo de brilho), sendo muito utilizada para visualizar a magnitude de Fourier

$$J(p) = H \log(1 + |\vec{I}(p)|),$$

onde  $I = \{I_1, I_2\}$  contém a parte real de  $I_1$  e a imaginária de  $I_2$  do espectro

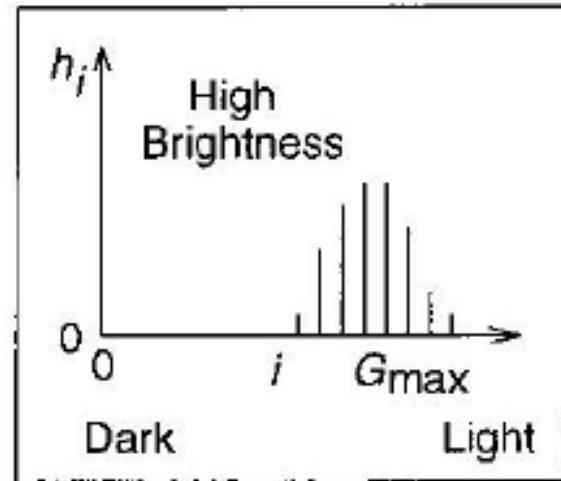
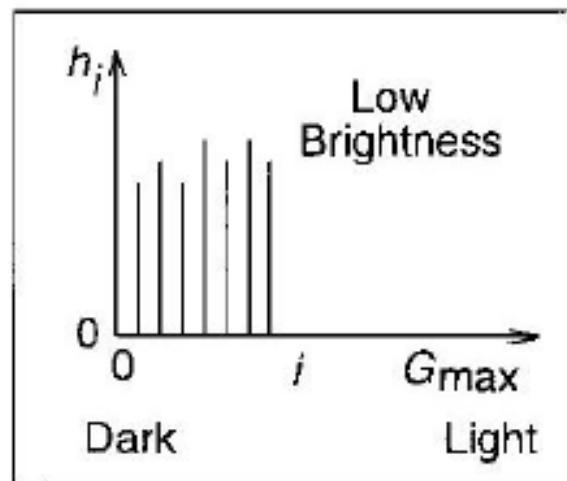
# Transformação Radiométricas



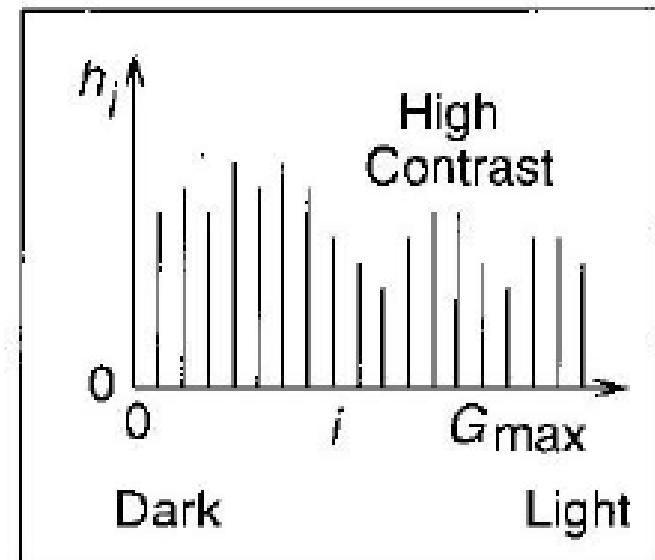
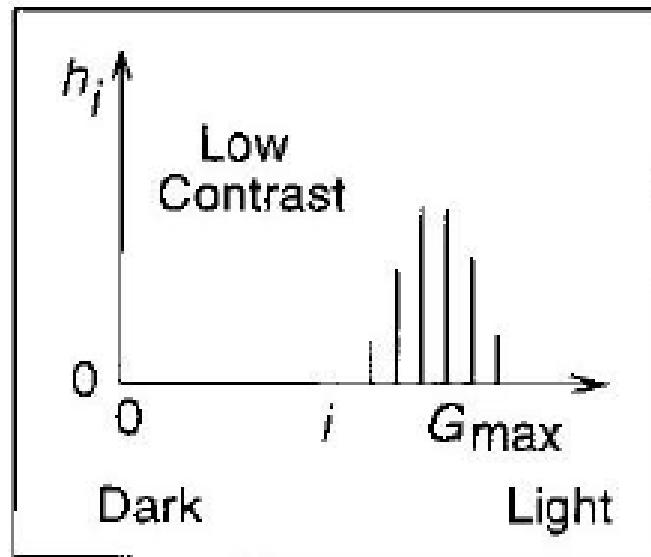
# Equalização de Histograma



- **Objetivo:** normalização e/ou aumento de contraste



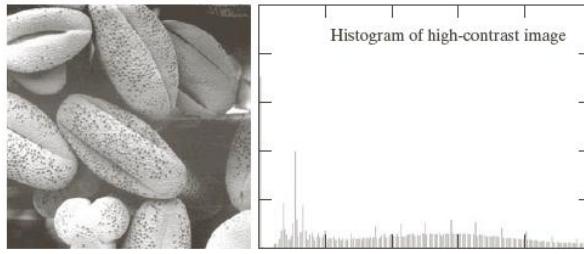
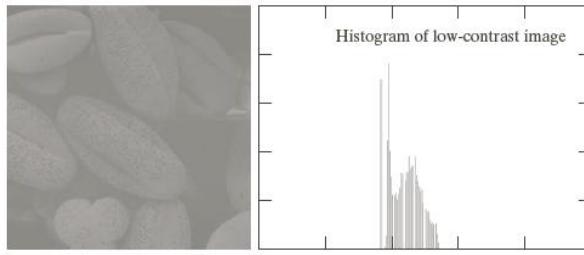
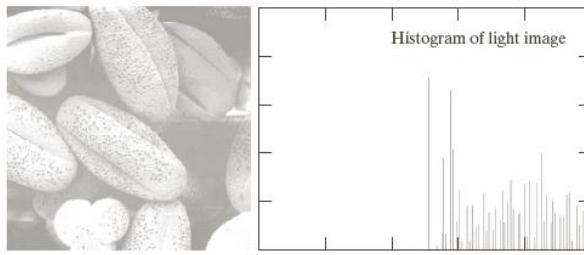
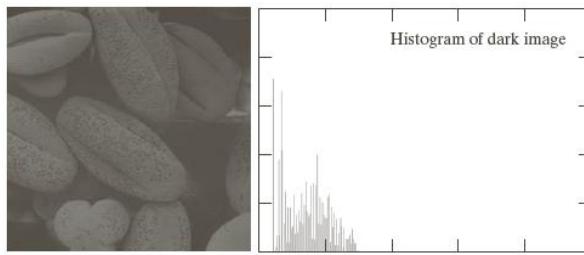
# Equalização de Histograma



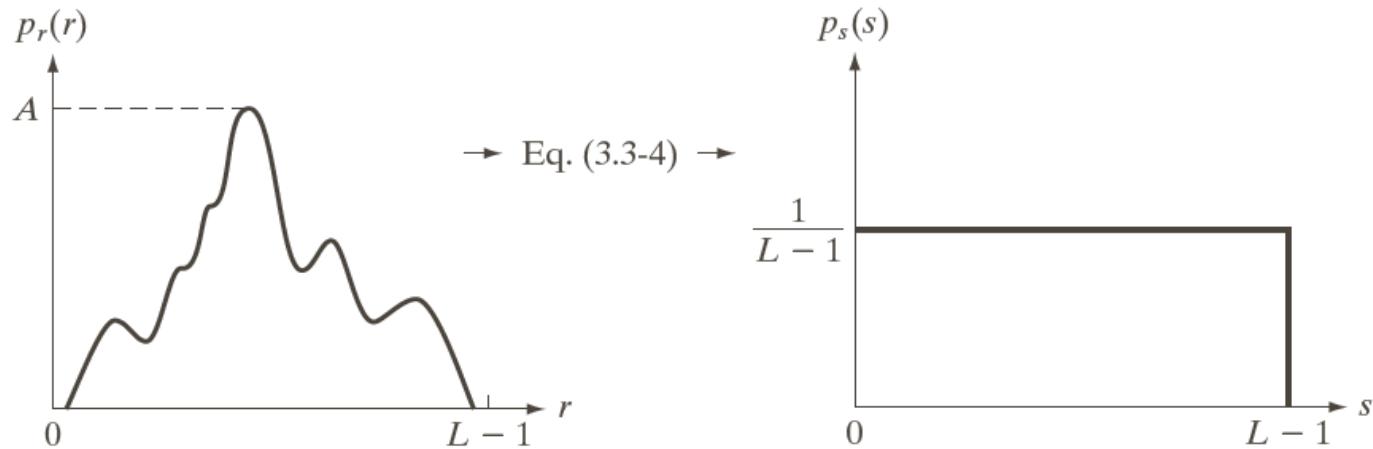
# Equalização de Histograma



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



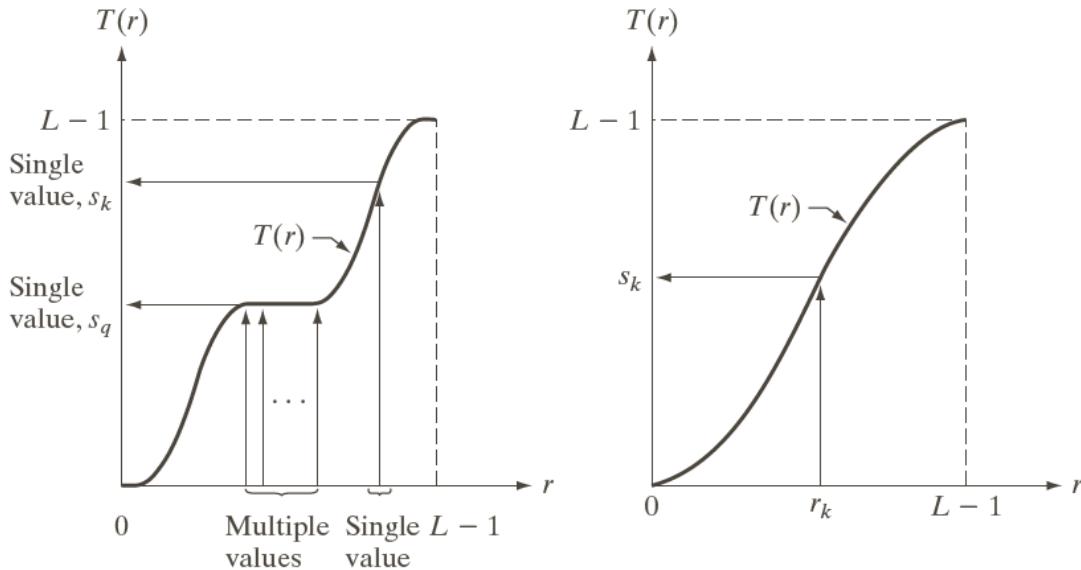
# Equalização histogramática



a | b

**FIGURE 3.18** (a) An arbitrary PDF. (b) Result of applying the transformation in Eq. (3.3-4) to all intensity levels,  $r$ . The resulting intensities,  $s$ , have a uniform PDF, independently of the form of the PDF of the  $r$ 's.

# Equalização histogramática



**FIGURE 3.17**  
(a) Monotonically increasing function, showing how multiple values can map to a single value.  
(b) Strictly monotonically increasing function. This is a one-to-one mapping, both ways.

# Equalização histográmica



## Fundamentos

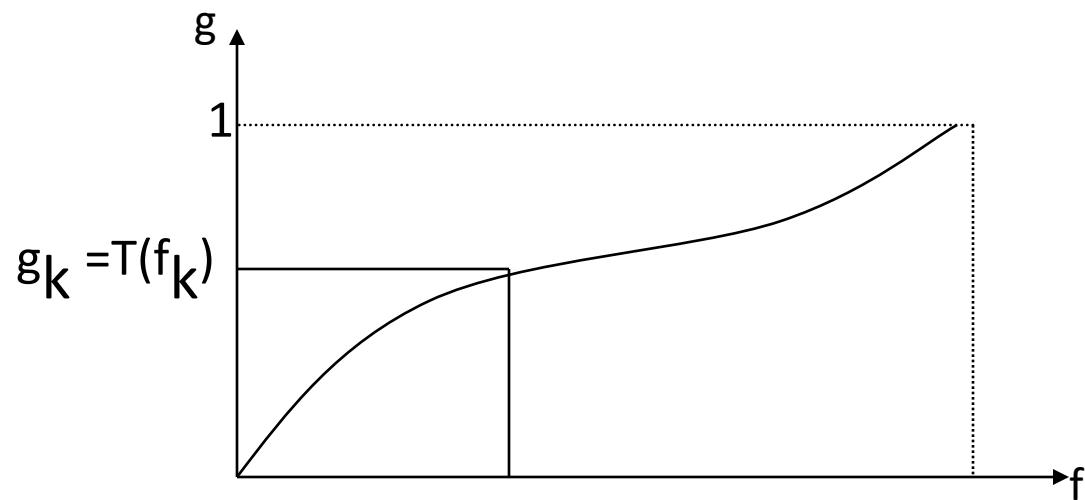
- Histograma: função de distribuição de probabilidade (DF),  $h = \text{imhist}(\text{img})$ ;
- Histograma normalizado: função de densidade de probabilidade (PDF) ,  $hn = \text{imhist}(\text{img}) / \text{size}(\text{img})$ ;
- Integração do HN: função cumulativa de distribuição,  $lhn = \text{cumsum}(hn)$ ;

# Equalização histográmica



- Transformação radiométrica que visa **aumentar a dinâmica** dos níveis de cinza **melhorando**, por exemplo, **o contraste** de imagens obtidas sob péssimas condições de iluminação.
- Idéia: gerar uma **distribuição mais uniforme** dos níveis de cinza → um histograma planar.

# Equalização histogramática



# Equalização histográmica



Consideremos, agora, a seguinte função  $T(f)$ :

$$g = T(f) = \int_0^f p_f(w) dw \quad 0 \leq f \leq 1$$

que representa a função de distribuição cumulativa (FDC) de  $f$  (esta função é monotonicamente crescente e varia de 0 a 1 em função de  $f$ ).

- Conclusão: se  $T(f)$  é uma FDC, então ela pode ser empregada na definição de uma nova imagem cuja distribuição dos níveis de cinza será mais uniforme.

# Equalização histográmica



- Imagem:

$n=M \times N$  pixels com valores discretos  $k = 0,1,\dots,L-1$ :

- $$p_f(f_k) = \frac{n_k}{n} \quad 0 \leq f_k \leq 1$$

onde:  $n_k$  = número de aparições do nível  $k$

$p_f(f_k)$  = probabilidade de ocorrência de  $f_k$

Assim:

$$g_k = T(f_k) = \sum_{j=0}^k \frac{n_j}{n} = \sum_{j=0}^k p_f(f_j), \quad 0 \leq f_k \leq 1 \text{ e } k = 0,1,\dots,L-1.$$

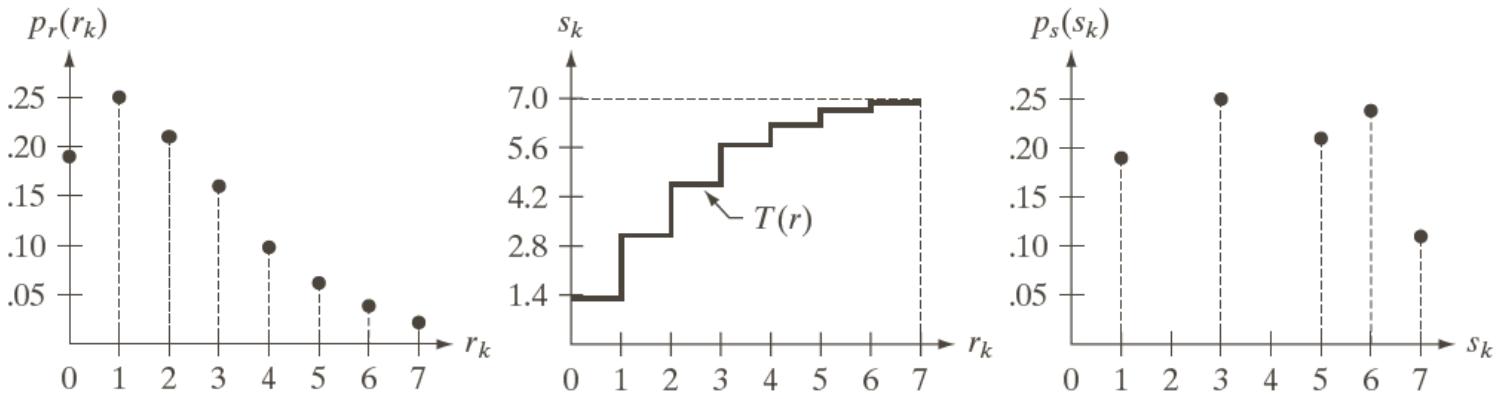
# Equalização histográmica



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

**TABLE 3.1**  
Intensity  
distribution and  
histogram values  
for a 3-bit,  
 $64 \times 64$  digital  
image.

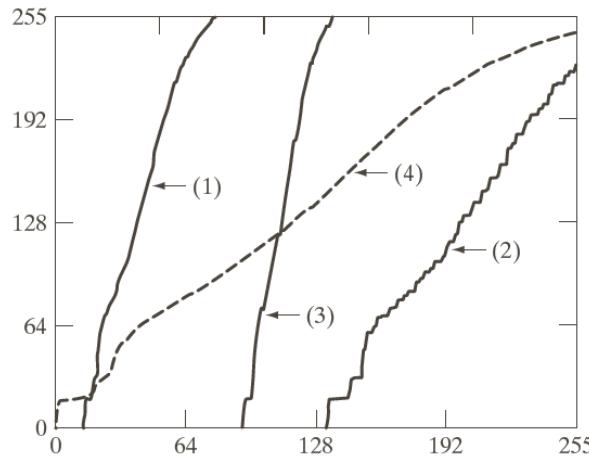
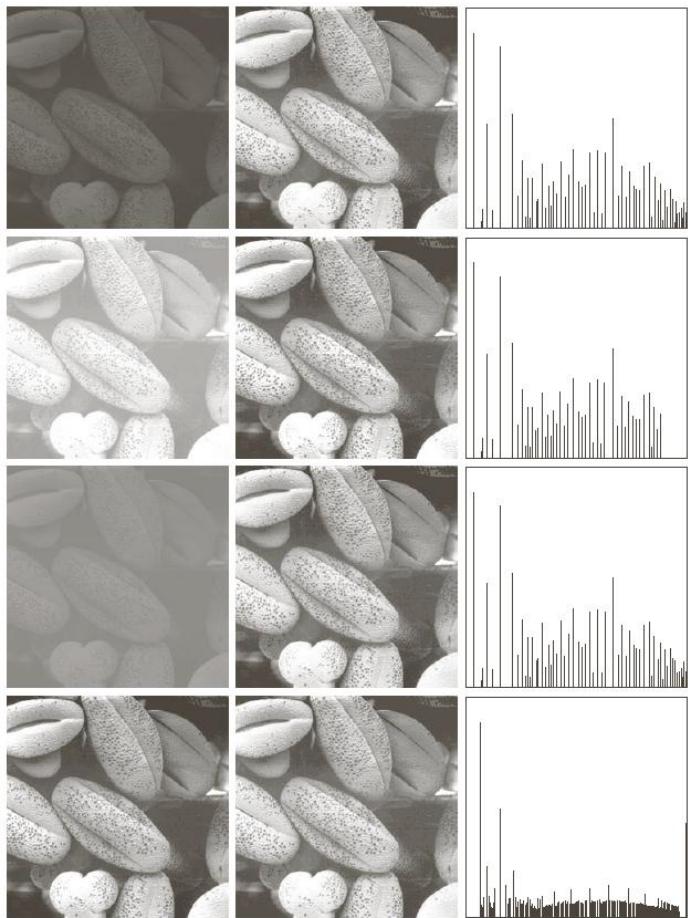
# Equalização histogramática



a b c

**FIGURE 3.19** Illustration of histogram equalization of a 3-bit (8 intensity levels) image. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.

# Equalização histográmica



**FIGURE 3.21**  
Transformation functions for histogram equalization. Transformations (1) through (4) were obtained from the histograms of the images (from top to bottom) in the left column of Fig. 3.20 using Eq. (3.3-8).

# Equalização histogramática



```
function nimg = EcualizacionHistograma (img)
[fil, col] = size(img); %dim. da imagen
N = fil * col; % numero total de pixels
hist = imhist(img) / N; %calcula o hist.norm
histAcumNorm = cumsum(hist); % hist. acum.
nimg = histAcumNorm(img+1);
nimg = uint8(nimg*255);
```

# Equalização histográmica



# Equalização histogramática



# Equalização histográmica



```
img = imread('lenna.png');  
nimg = histeq(img);  
imshow(nimg);
```

# Exercícios Propostos



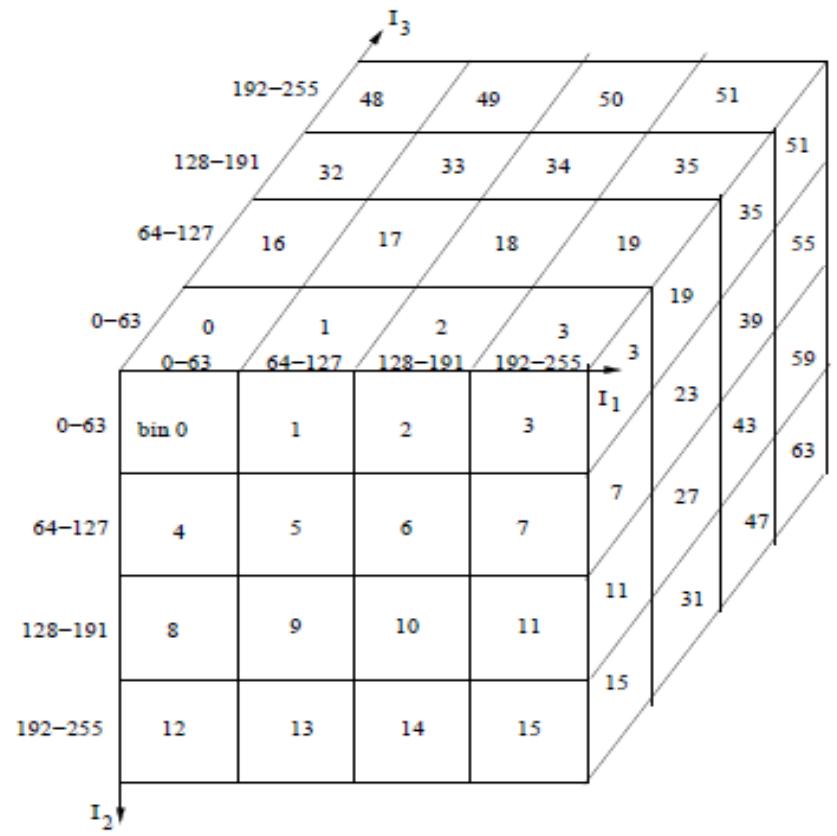
1. Implementar a função que permita quantizar uma imagem a  $n$  quantidade de cores, onde  $n$  não é necessariamente uma potência de 2.
2. Modifique a função Mapping(), de forma tal que é o usuário quem determina qual é o menor e o maior valor dentro na nova escala.

# Exercícios Propostos (cont)



3. Criar a função que calcula o histograma de uma imagem colorida.

A partir de uma imagem colorida com  $b = 8$  (0-255) bits para cada componente  $I_1$  (red),  $I_2$  (green), e  $I_3$  (blue), nós podemos dividir cada eixo do  $Z_3$  em 4 intervalos:  $[0, 63]$ ,  $[64, 127]$ ,  $[128, 191]$ , e  $[192, 255]$ . A contagem de cores em cada bin é usada no cálculo do histograma.



# Exercícios Propostos (cont)



## Sistema RGB

