

## Princípios e representação de imagens

**Prof. Luiz Otavio Murta Jr.**  
**Informática Biomédica**

**Depto. de Física e Matemática (FFCLRP/USP)**

- **Processamento de imagens**
- **Percepção visual**
- **Modelos de imagens**
  - Representação
  - Profundidade de pixels
- **Brilho e contraste**
- **Histograma**
  - Algoritmo de contagem de histograma
  - Equalização de histograma

# Tipos de processamento de imagem

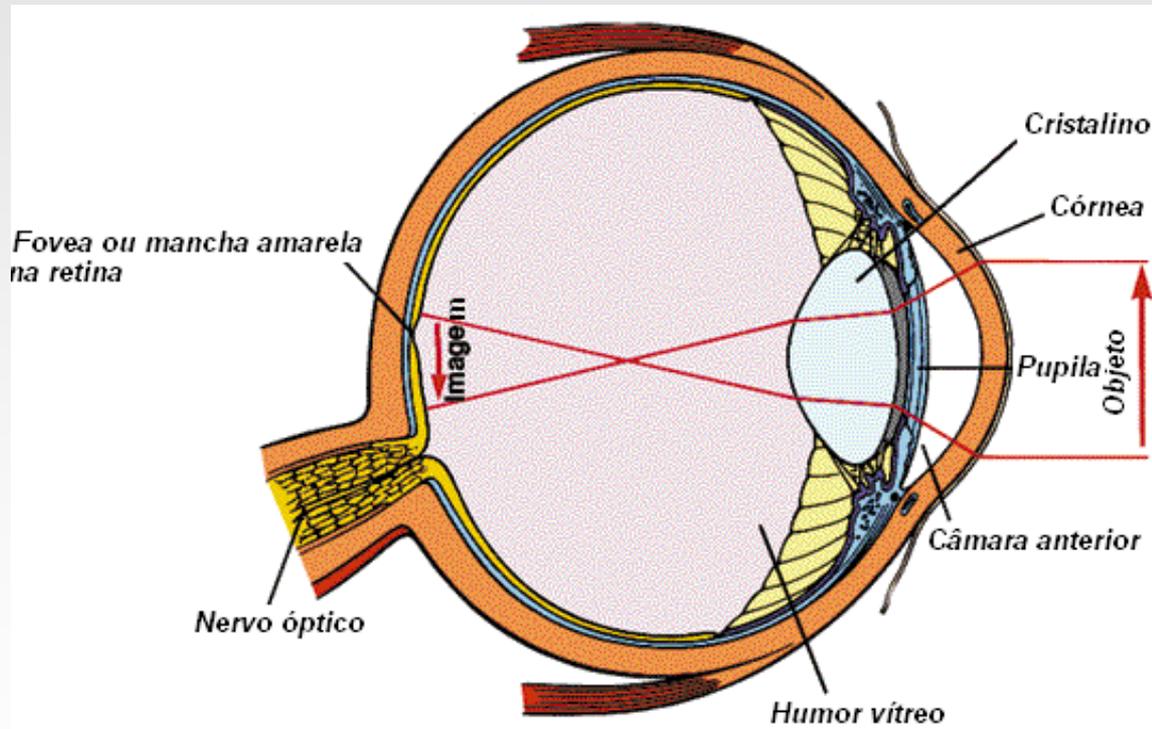
## ◆ Processamento Digital de Imagem (descrição)

Designação	Entrada	Saída
Síntese de Imagem	<i>Descrição</i>	<i>Imagem</i>
Processamento de Imagem	<i>Imagem</i>	<i>Imagem</i>
Análise de Imagem	<i>Imagem</i>	<i>Medidas</i>
Compreensão de Imagem	<i>Imagem</i>	<i>Descrição</i>

Domínio de aplicações 2D e 3D

# O Fenômeno Visual

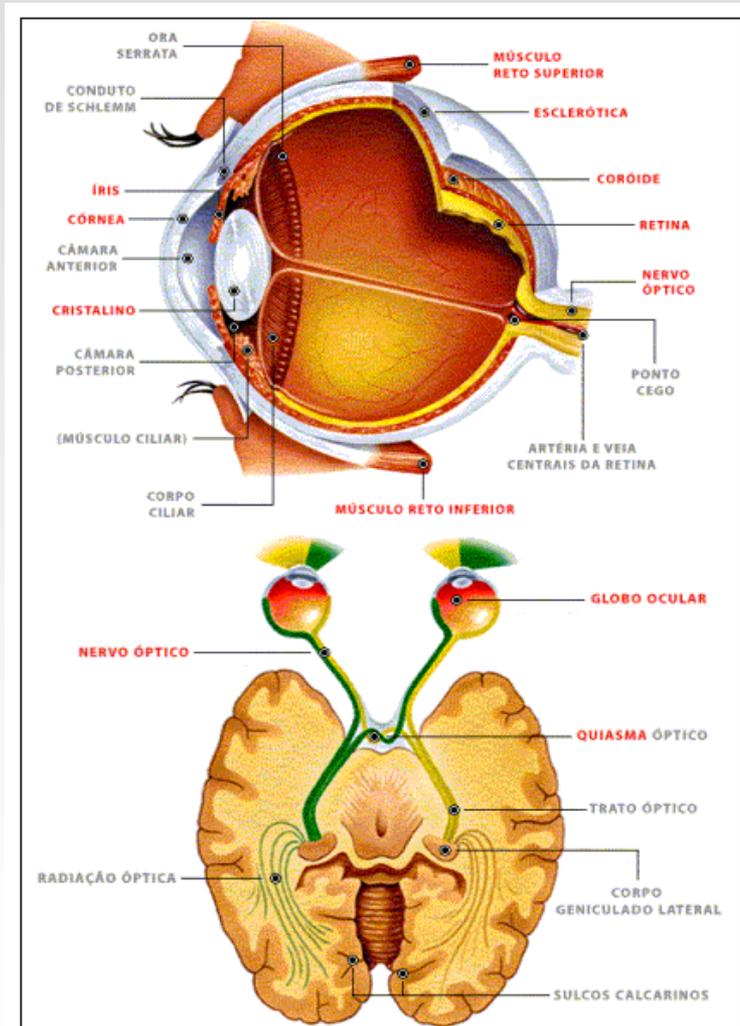
- Percepção sensorial da visão



# O Fenômeno Visual

## Anatomia da visão:

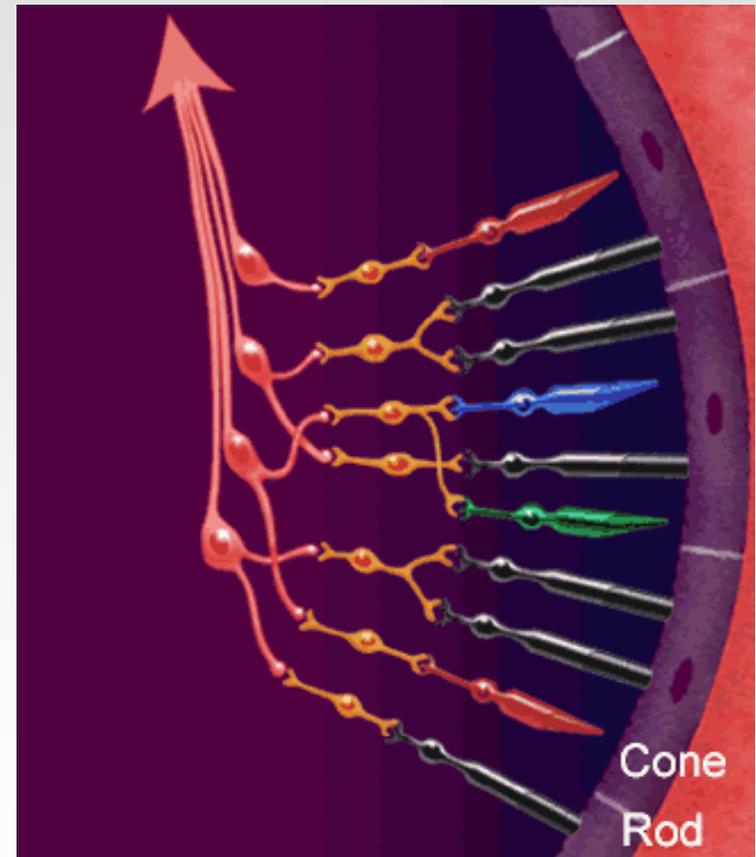
- Córnea
- Cristalino
- Retina
- Nervo ótico
- Córtex visual



# O Fenômeno Visual

## Percepção sensorial da visão:

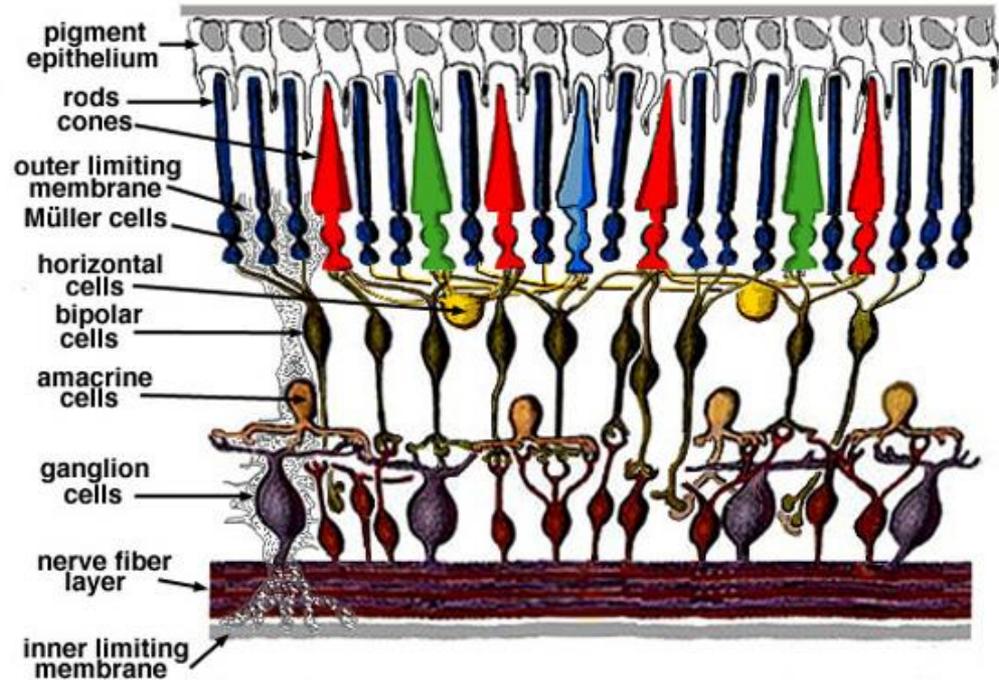
- Incidência de luz na retina
- Excitação dos cones e bastonetes
- Captação do nervo ótico
- Transporte de informação para o cortex visual (cérebro)



# O Fenômeno Visual

## Fisiologia da visão:

- Cones
- Bastonetes

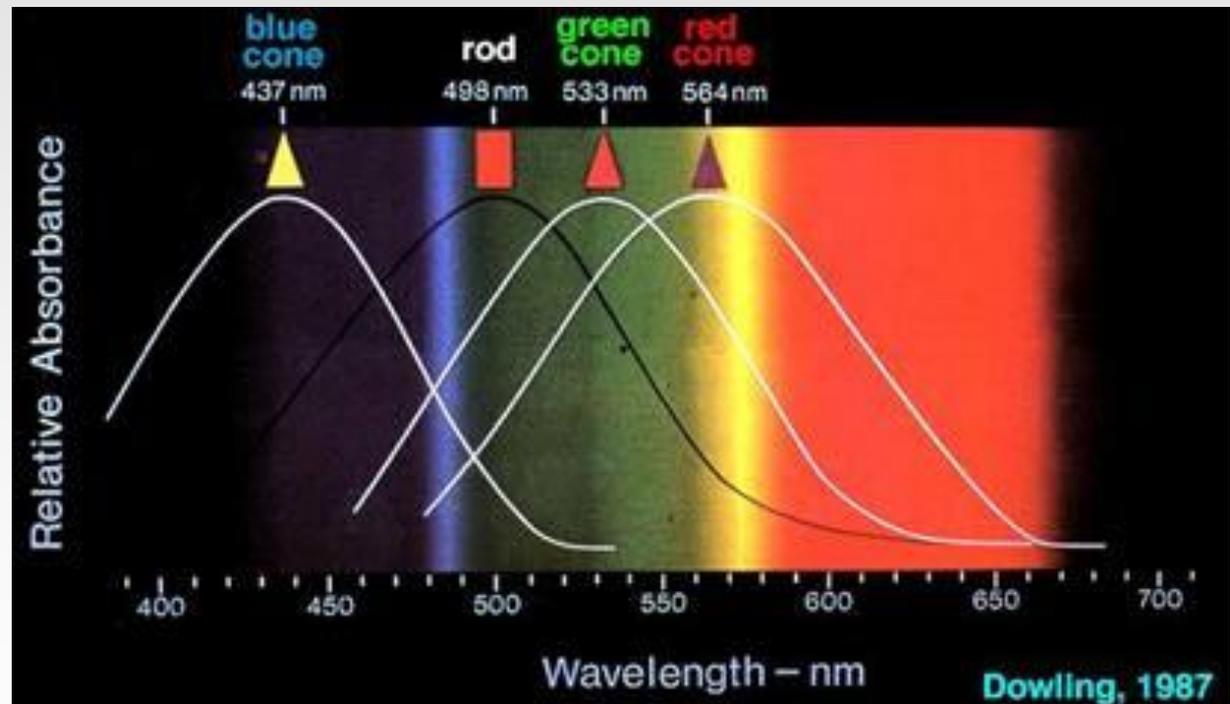


*Fig. 2. Simple diagram of the organization of the retina.*

# O Fenômeno Visual

## Fisiologia da visão:

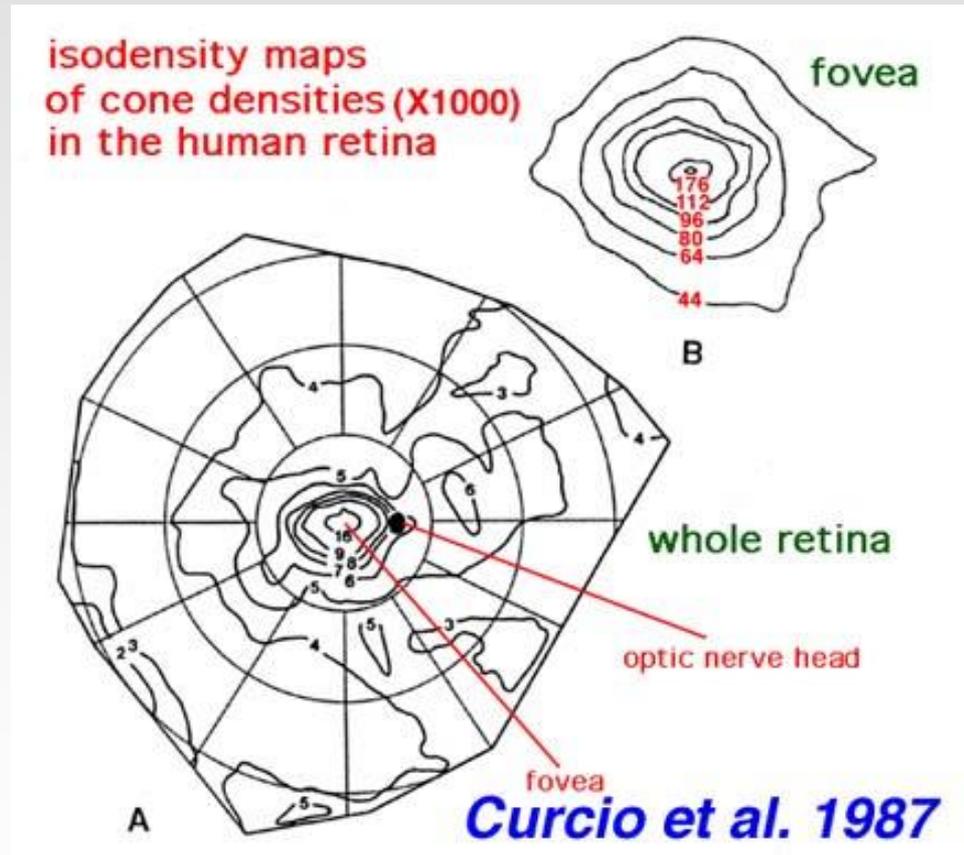
- Cones
- Bastonetes



# O Fenômeno Visual

## Fisiologia da visão:

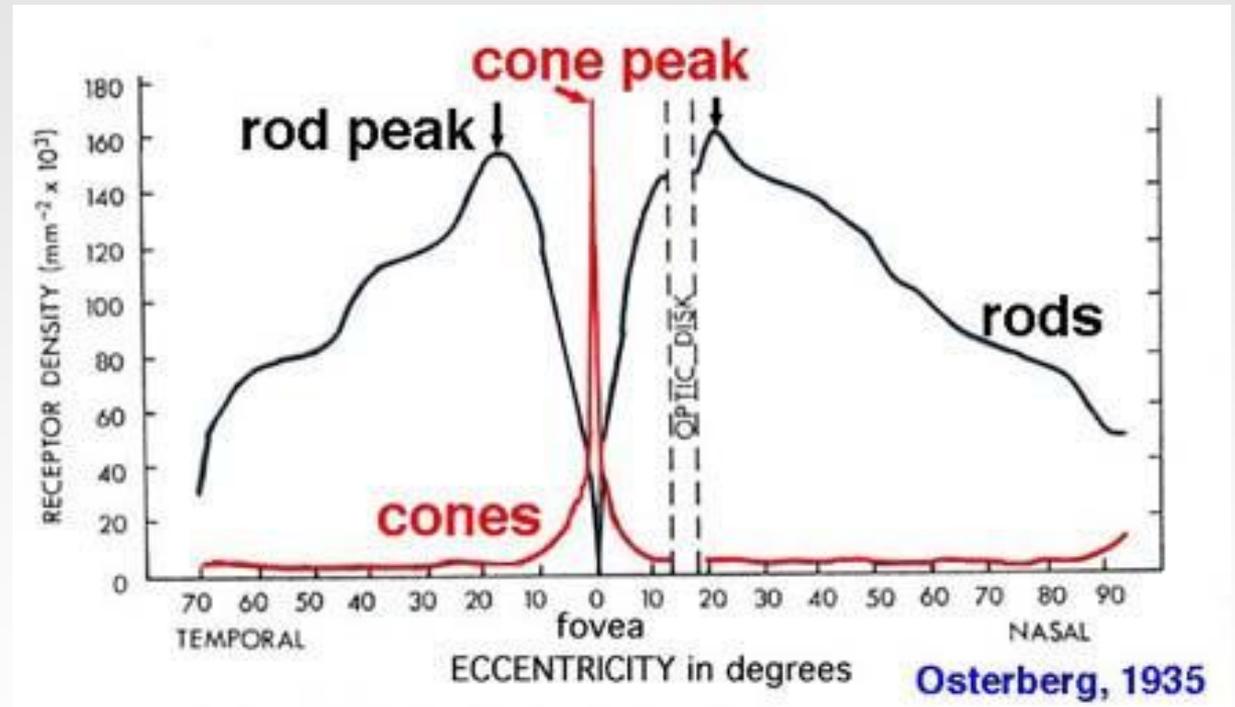
- Cones
- Bastonetes



# O Fenômeno Visual

## Fisiologia da visão:

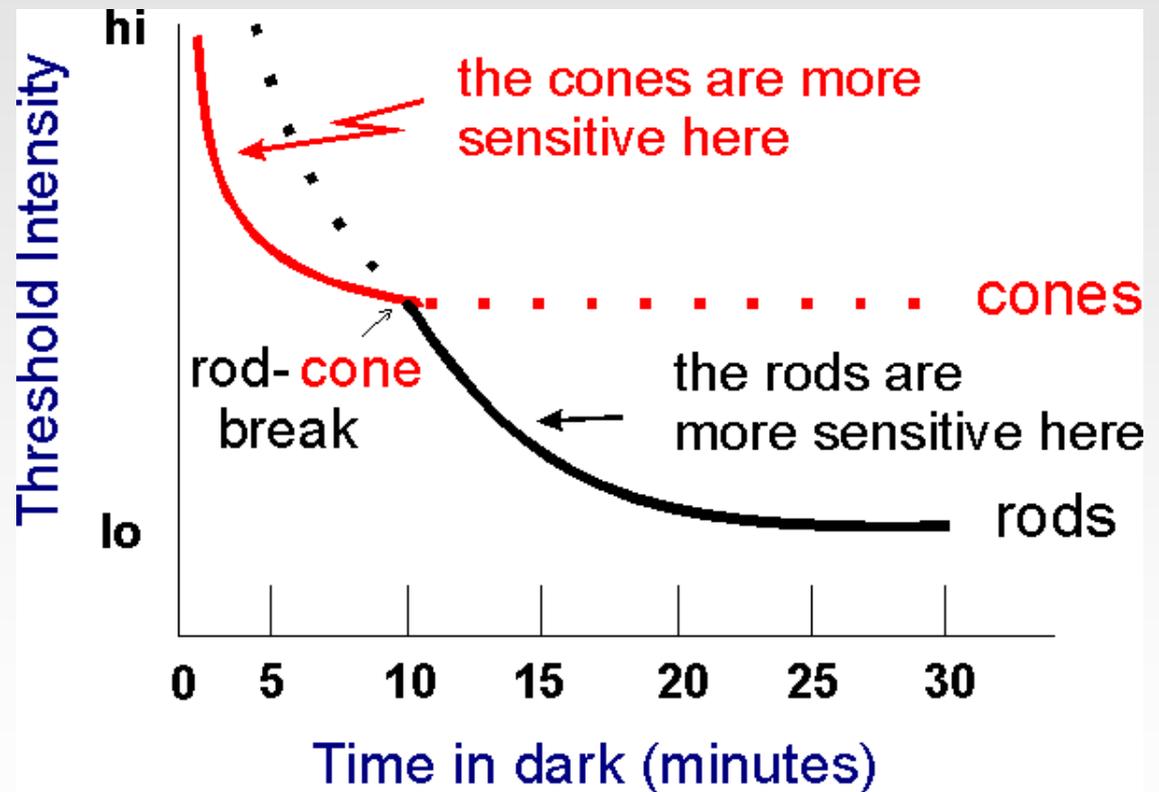
- Cones
- Bastonetes



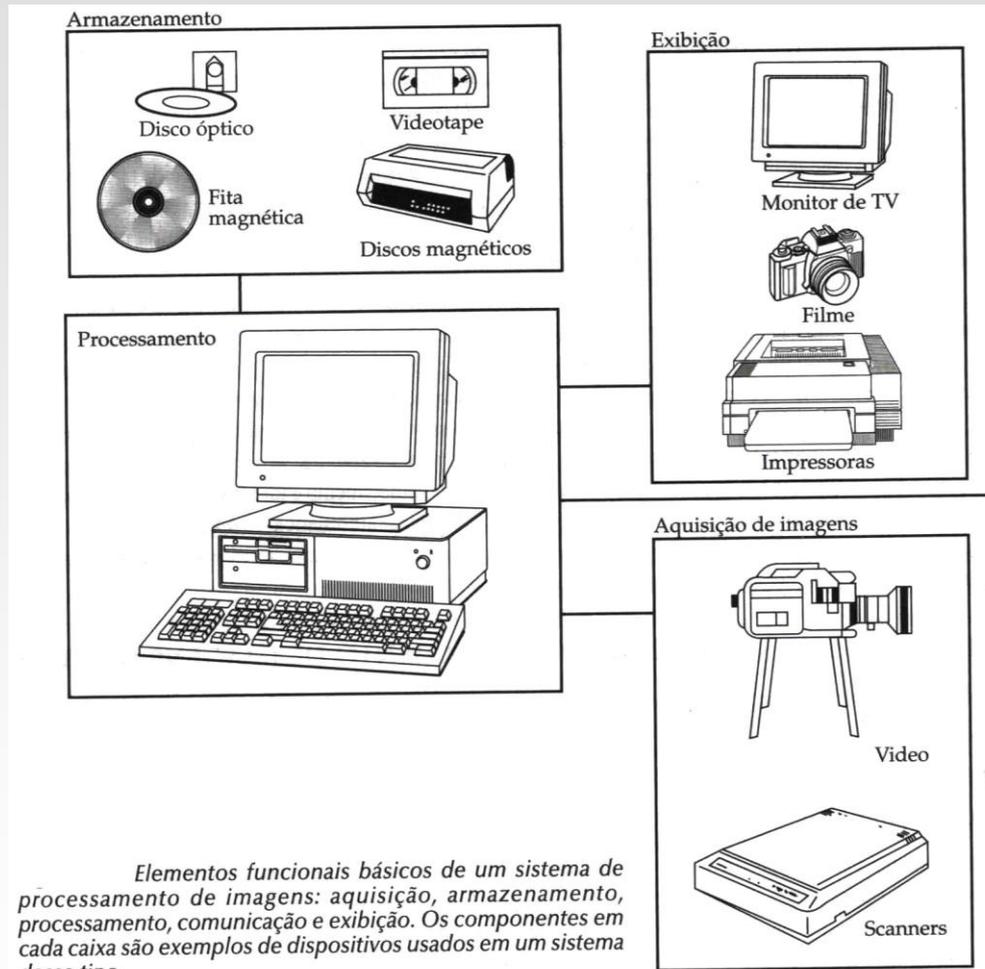
# O Fenômeno Visual

## Fisiologia da visão:

- Cones
- Bastonetes



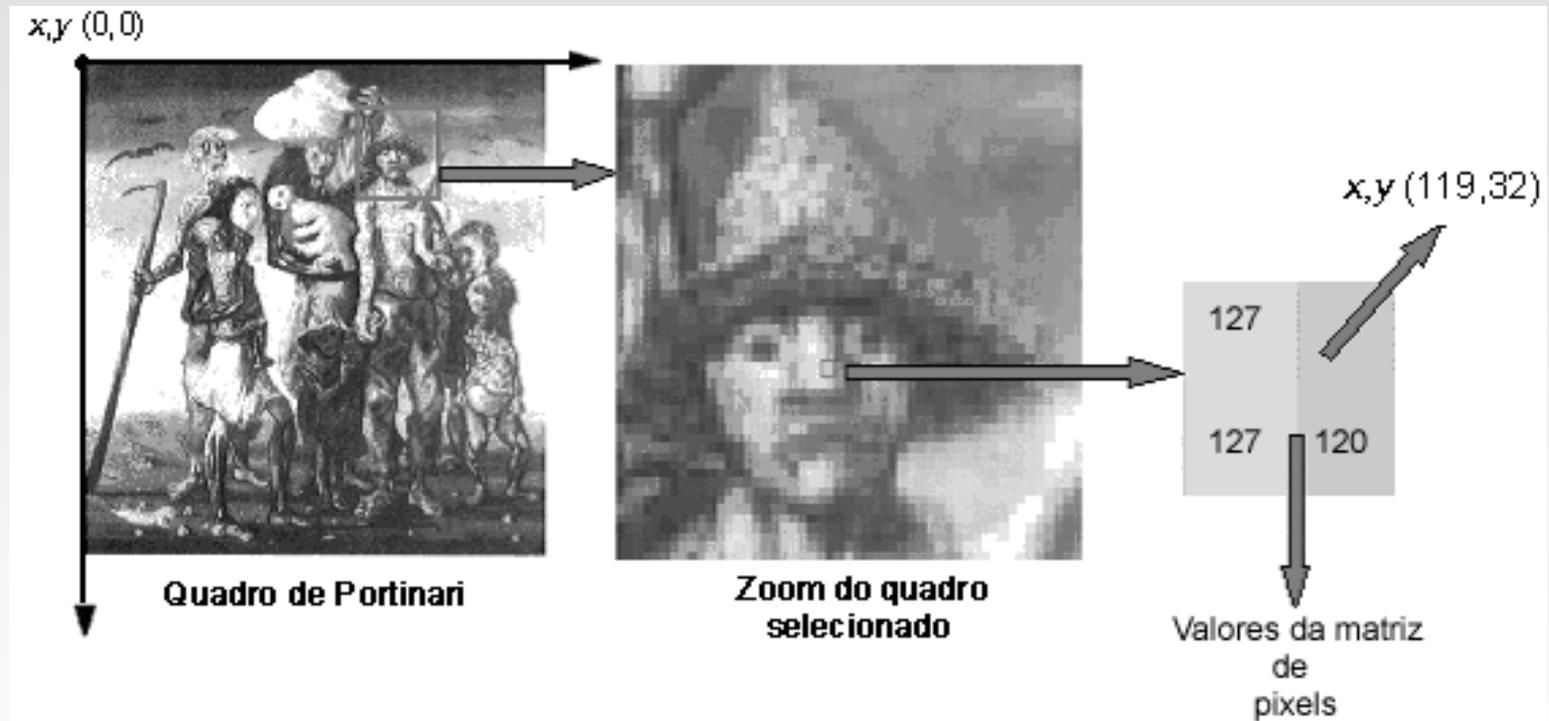
# Processamento de imagem



## Imagens médicas:

- ◆ Raios X
- ◆ Ressonância
- ◆ Ultra-som
- ◆ Medicina nuclear
- ◆ .....

# Modelo simples de imagem



# Modelo simples de imagem



$$F(x, y)$$

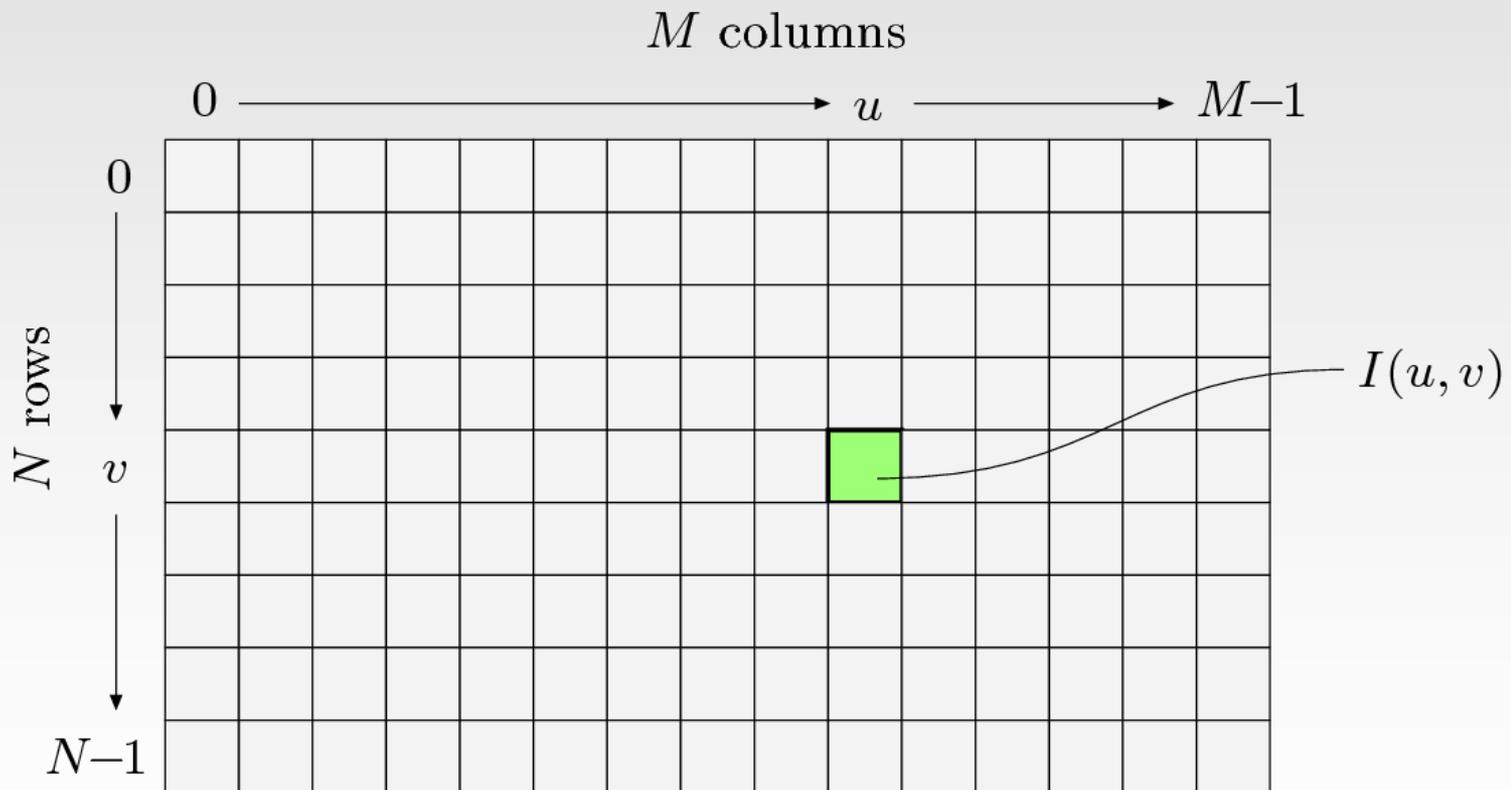


148	123	52	107	123	162	172	123	64	89	...
147	130	92	95	98	130	171	155	169	163	...
141	118	121	148	117	107	144	137	136	134	...
82	106	93	172	149	131	138	114	113	129	...
57	101	72	54	109	111	104	135	106	125	...
138	135	114	82	121	110	34	76	101	111	...
138	102	128	159	168	147	116	129	124	117	...
113	89	89	109	106	126	114	150	164	145	...
120	121	123	87	85	70	119	64	79	127	...
145	141	143	134	111	124	117	113	64	112	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

$$I(u, v)$$

# Modelo simples de imagem

Localização de pixels:



# Representação de imagem

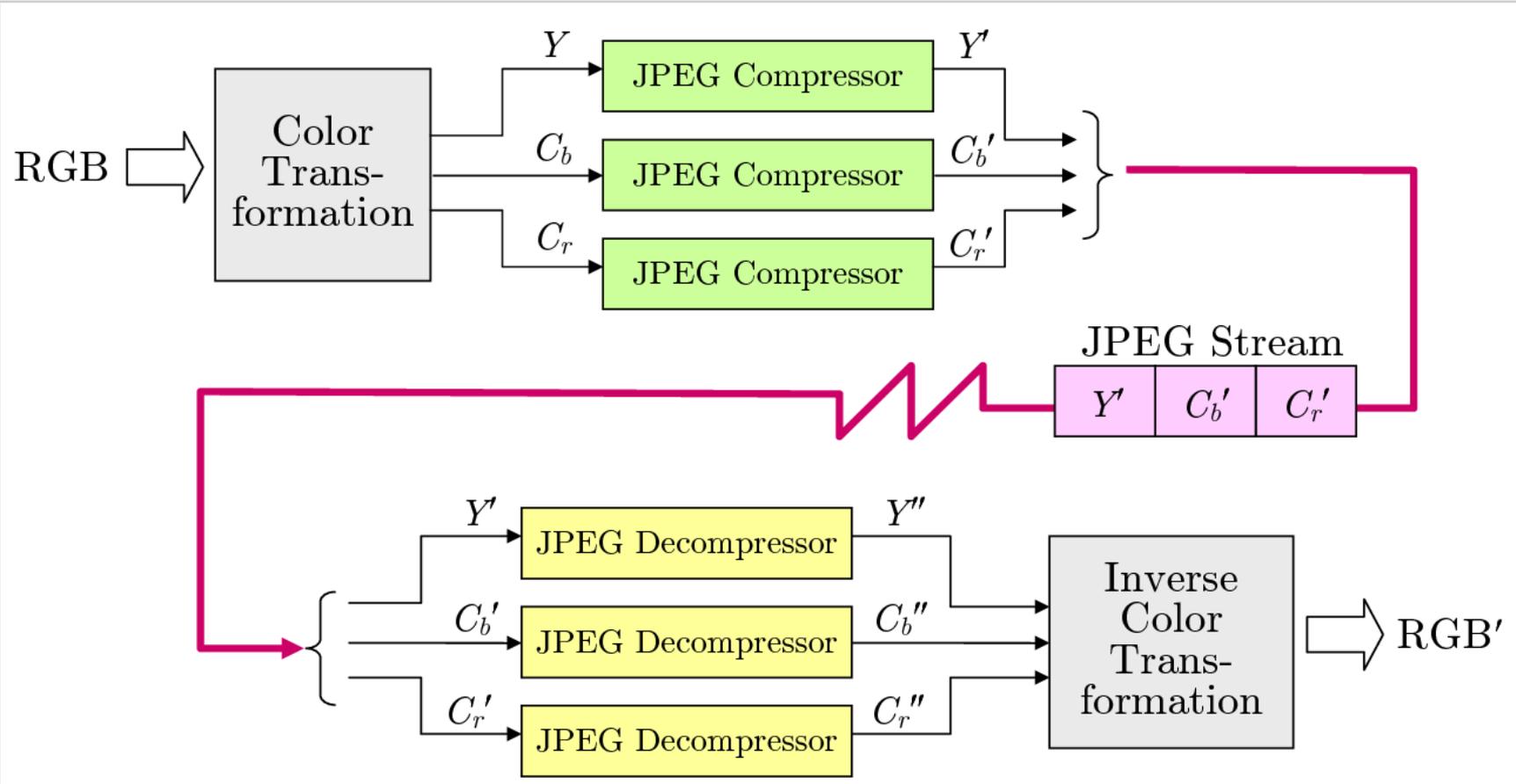
## Tons de cinza

Canais	Bits/pix.	Escopo	Uso
1	1	0...1	Imagem binária, documento, fax...
1	8	0...255	Universal, foto, scan, impressão...
1	12	0...4095	Foto alta qualidade, scan, impressão...
1	14	0...16383	Foto profissional, scsn, impressão
1	16	0...65535	Fotos de altíssima qualidade,...

## Imagens coloridas

Canais	Bits/pix.	Escopo	Uso
3	24	0...255	RGB universal, foto, scan, impressão...
3	36	0...4095	RGB foto alta qualidade, scan, impressão...
3	42	0...16383	RGB foto profissional, scsn, impressão
4	32	0...255	CMYK pre-impressão digital

# Representação de imagem



# Representação de imagem

## Arquivo:

```

P2
# oie.pgm
17 7
255
0 13 13 13 13 13 13 13 0 0 0 0 0 0 0 0
0 13 0 0 0 0 0 13 0 7 7 0 0 81 81 81 81
0 13 0 7 7 7 0 13 0 7 7 0 0 81 0 0 0
0 13 0 7 0 7 0 13 0 7 7 0 0 81 81 81 0
0 13 0 7 7 7 0 13 0 7 7 0 0 81 0 0 0
0 13 0 0 0 0 0 13 0 7 7 0 0 81 81 81 81
0 13 13 13 13 13 13 13 0 0 0 0 0 0 0 0
    
```

## Imagem:



<i>Format</i>	<i>Signature</i>
PNG	0x89504e47 □PNG
JPEG/JFIF	0xffd8ffe0 □□□□
TIFF <sub>little</sub>	0x49492a00 II*□
TIFF <sub>big</sub>	0x4d4d002a MM□*

<i>Format</i>	<i>Signature</i>
BMP	0x424d BM
GIF	0x4749463839 GIF89
Photoshop	0x38425053 8BPS
PS/EPS	0x25215053 %!PS

# Amostragem e Quantização

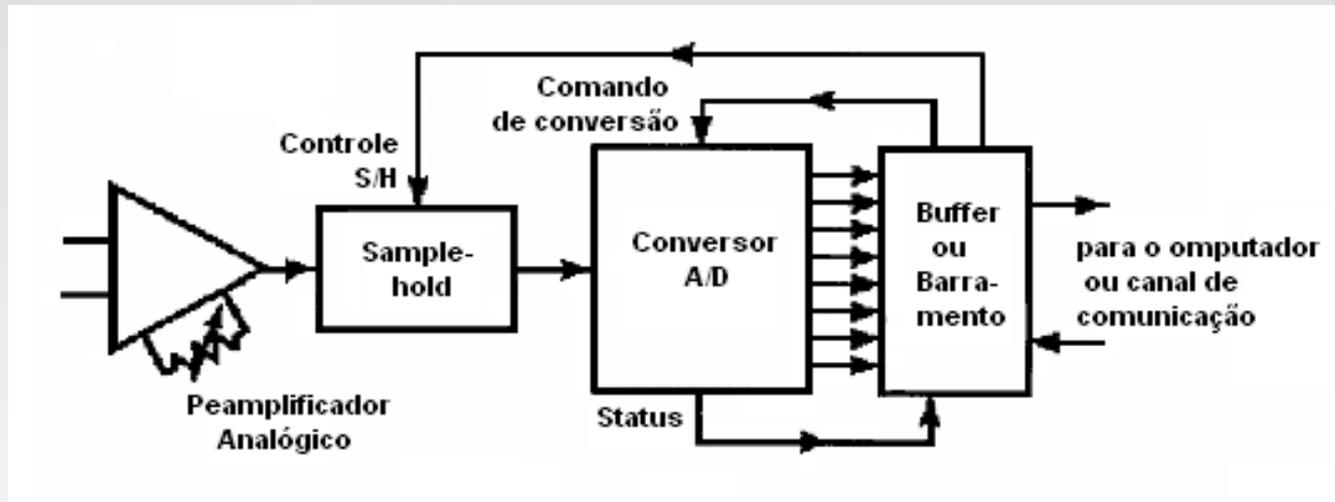
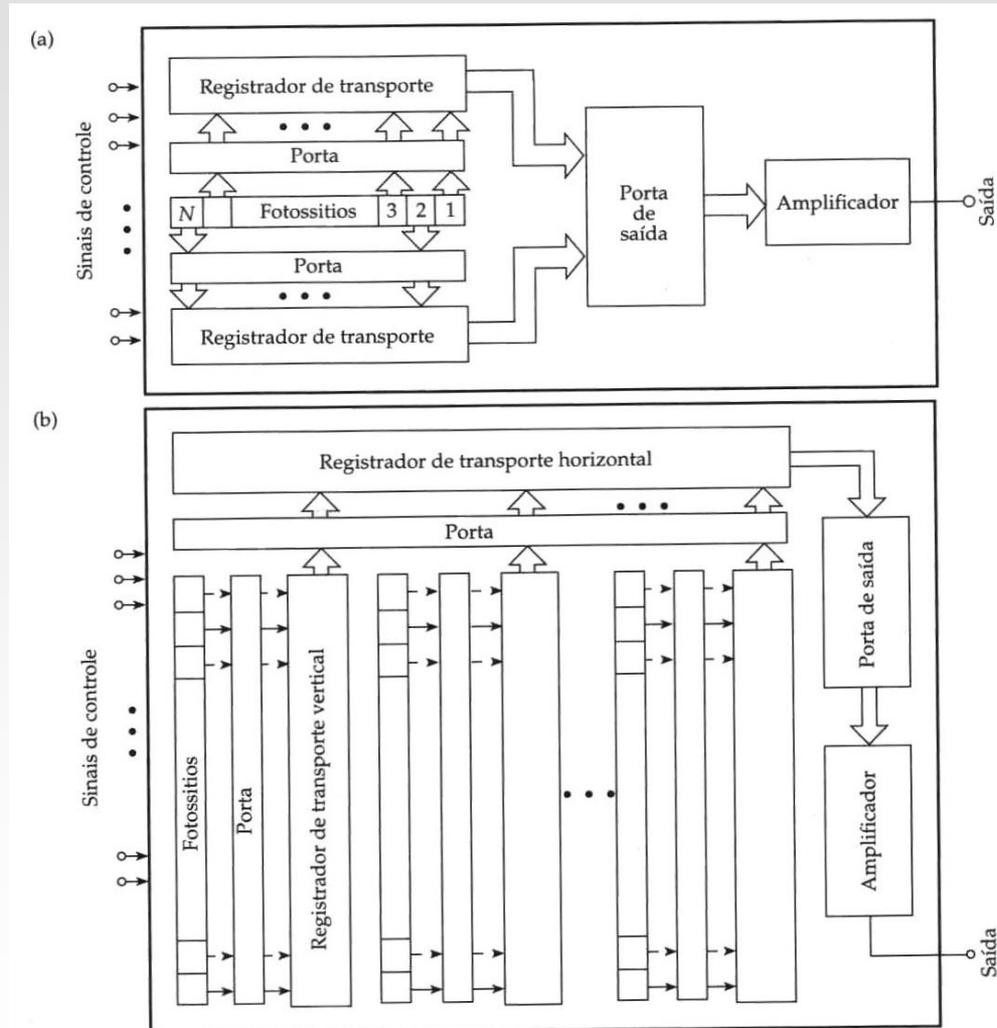


Ilustração esquemática de um sistema de conversão analógico / digital

# Aquisição via CCD

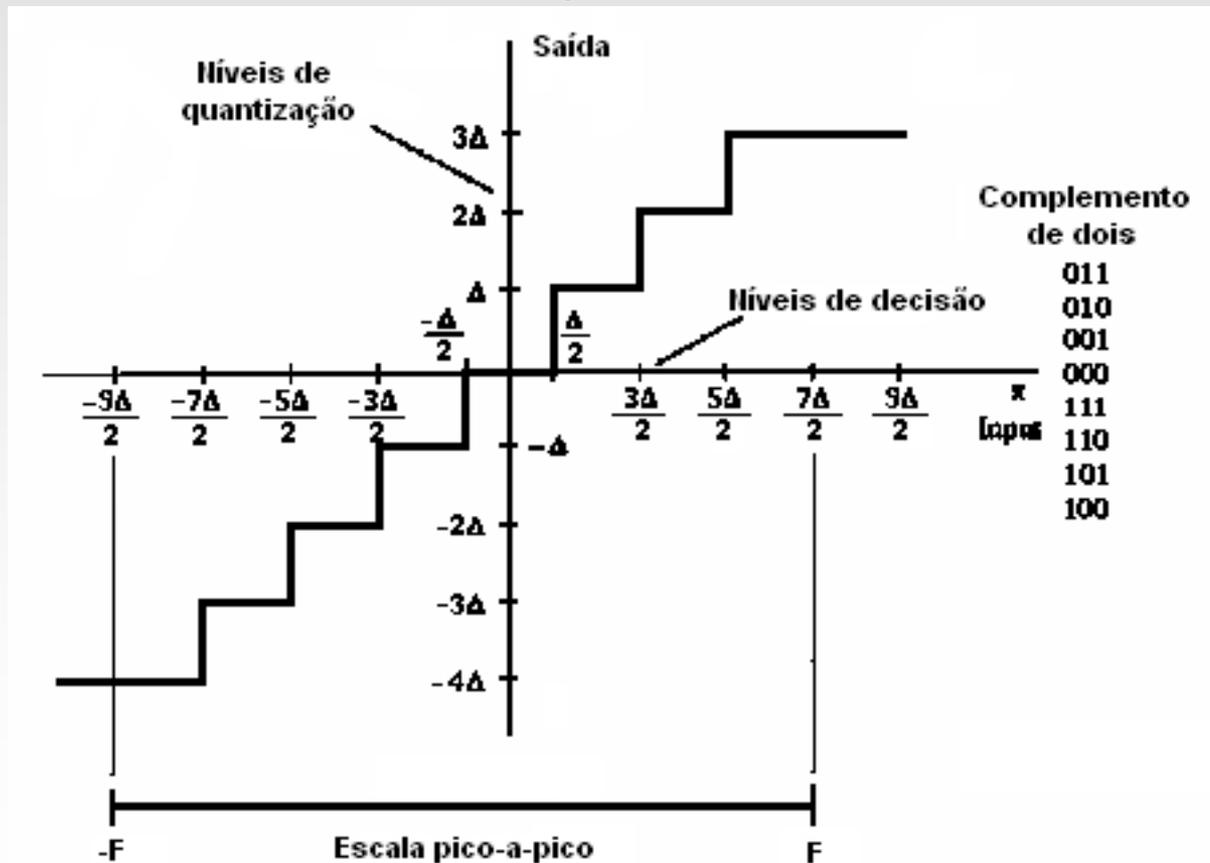


– a) Sensor CCD por varredura de linha; b) sensor CCD de área.

# Amostragem e Quantização

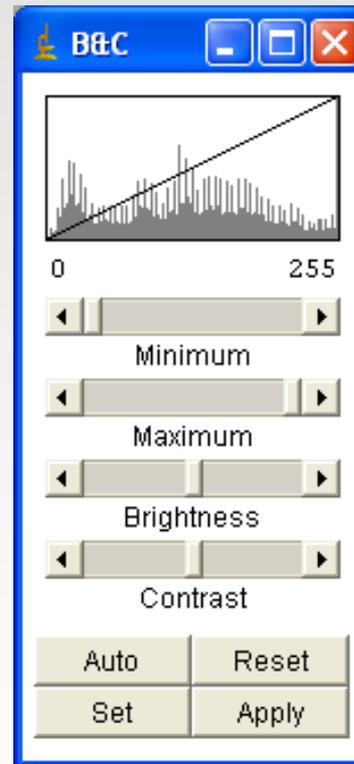
## Codificação

A codificação na conversão A/D se dá atribuindo a cada valor quantizado um número binário, como no exemplo abaixo.



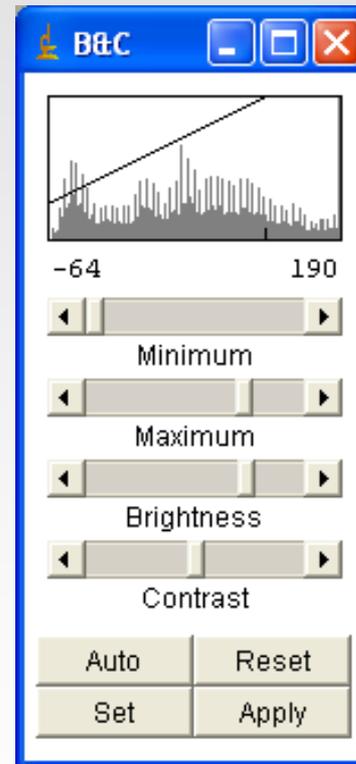
# Brilho, Contraste e Histograma

histograma de imagem com brilho e contraste originais



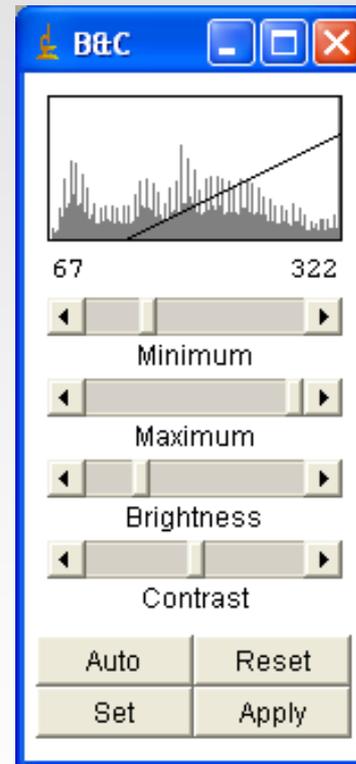
# Brilho, Contraste e Histograma

histograma de imagem com brilho incrementado



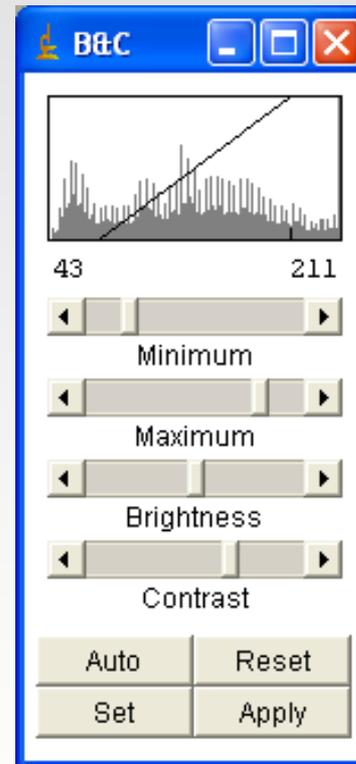
# Brilho, Contraste e Histograma

histograma de imagem com brilho reduzido



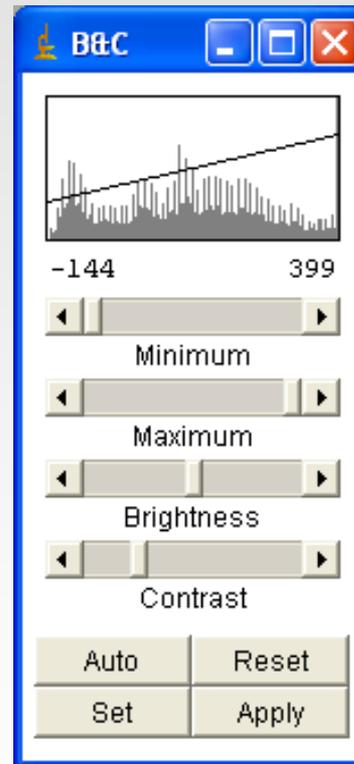
# Brilho, Contraste e Histograma

histograma de imagem com contraste incrementado

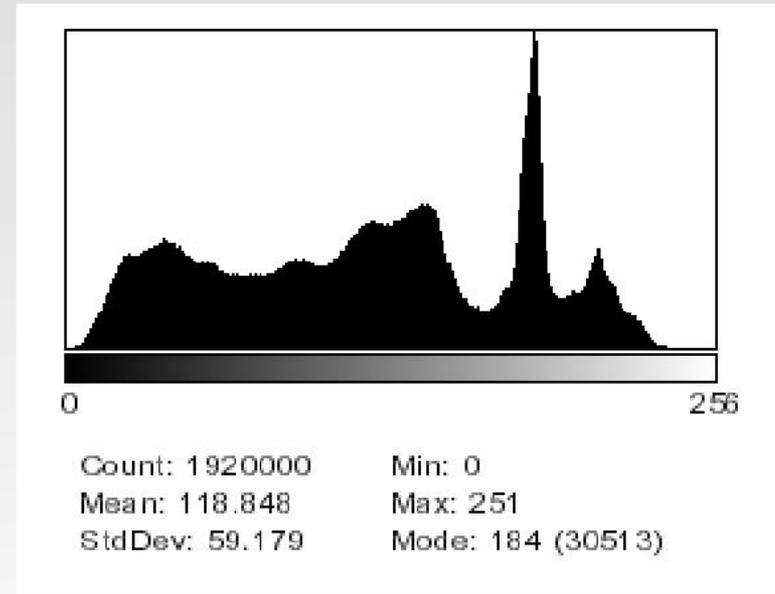


# Brilho, Contraste e Histograma

histograma de imagem com contraste reduzido

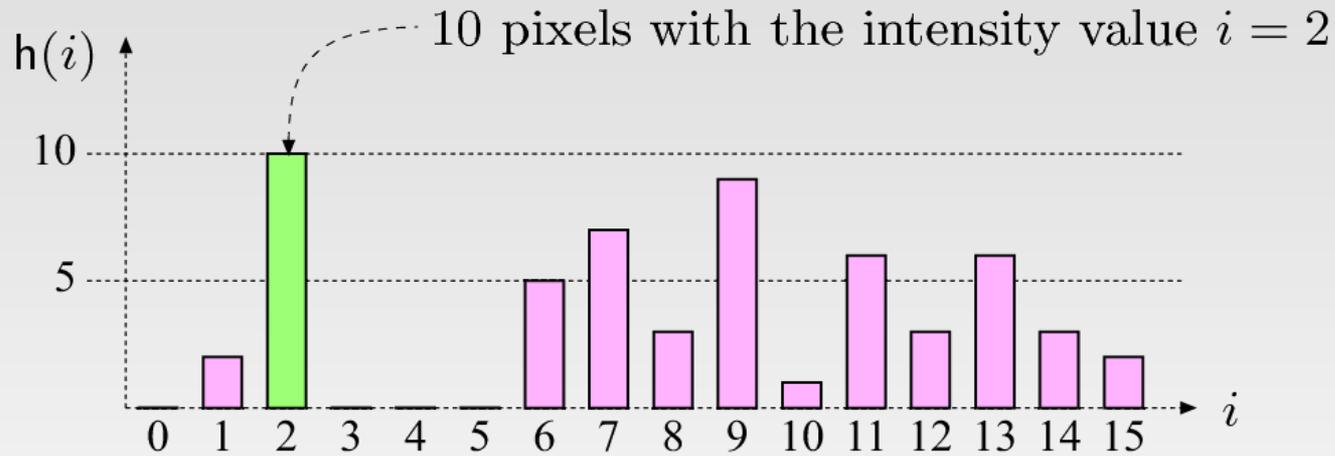


# Histograma



- Contagem de pixels em cada tom de cinza

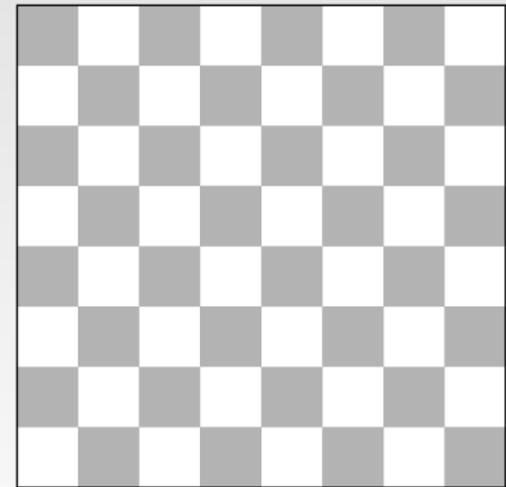
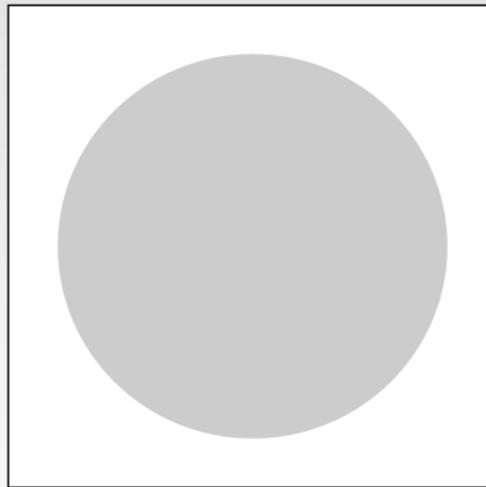
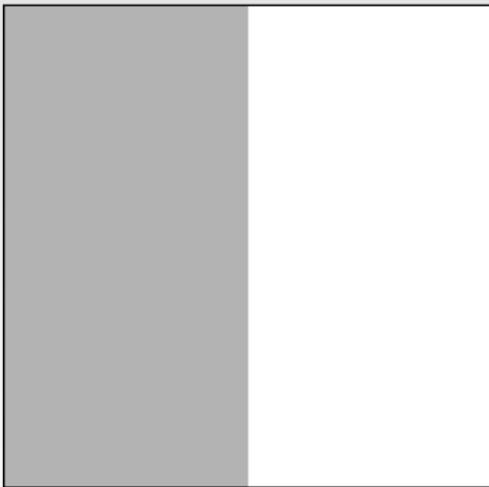
# Histograma



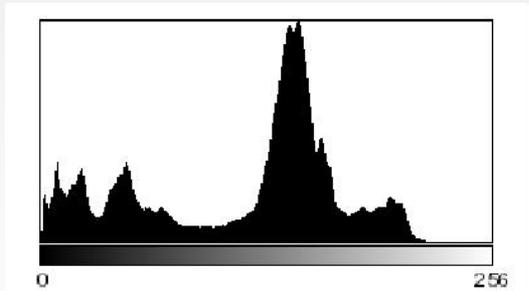
$h(i)$	0	2	10	0	0	0	5	7	3	9	1	6	3	6	3	2
$i$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

- Contagem de pixels em cada tom de cinza

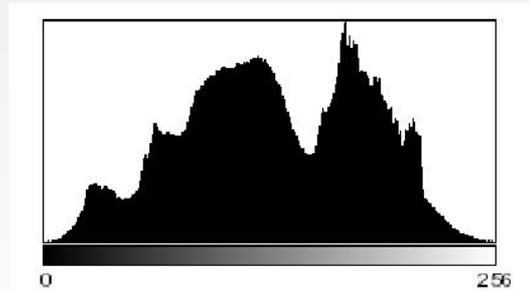
- Diferentes imagens podem ter o mesmo histograma?



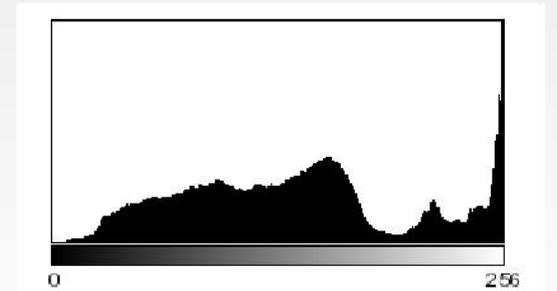
# Histograma



(a)

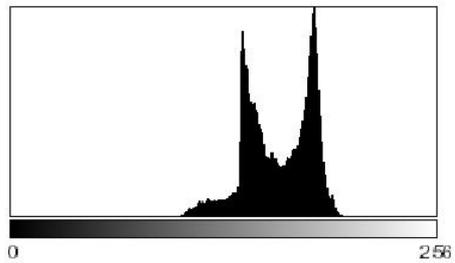
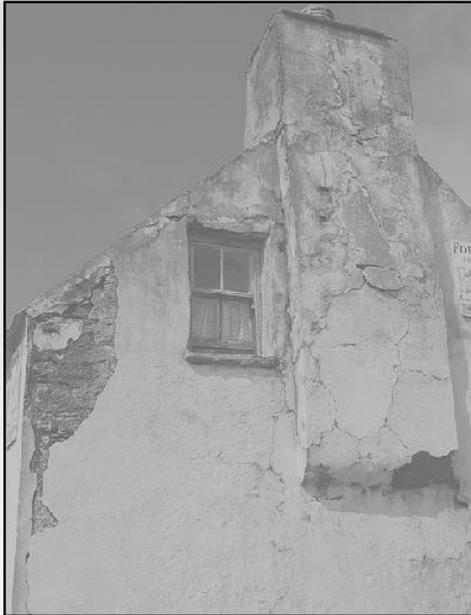


(b)

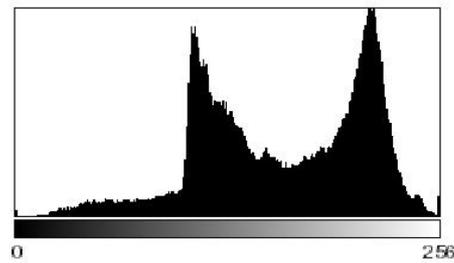


(c)

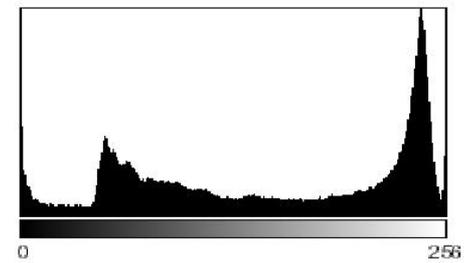
# Histograma



(a)

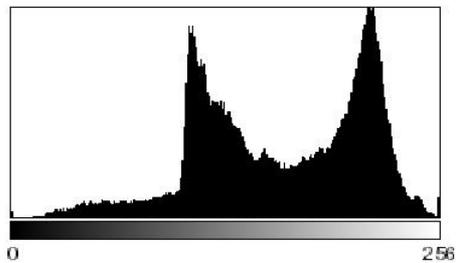


(b)

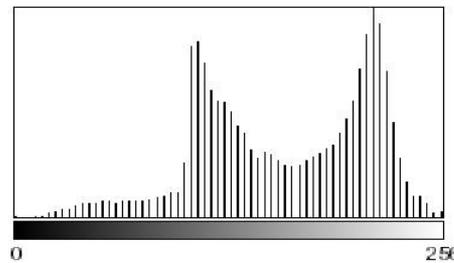


(c)

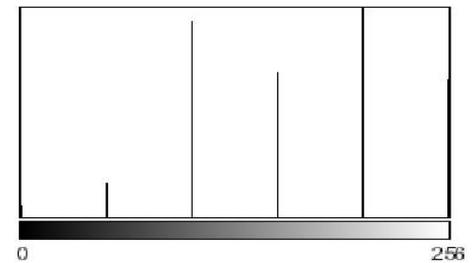
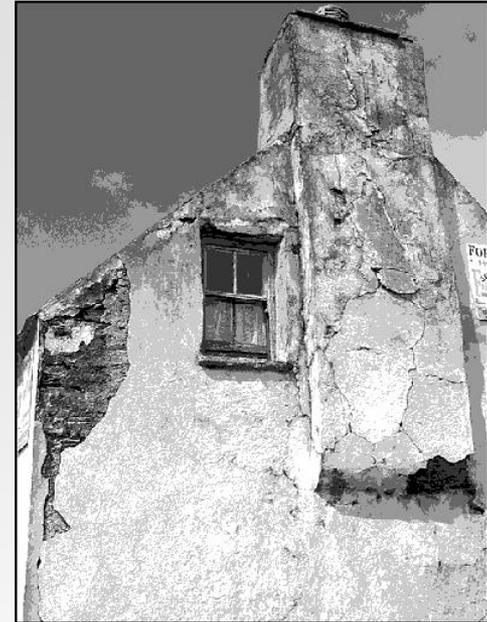
# Histograma



(a)



(b)

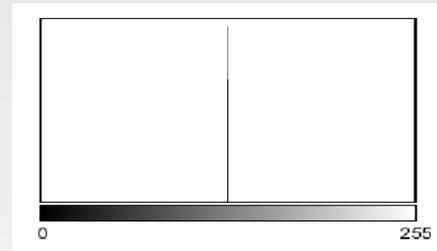


(c)

Histogramas distribuídos significam melhores imagens?



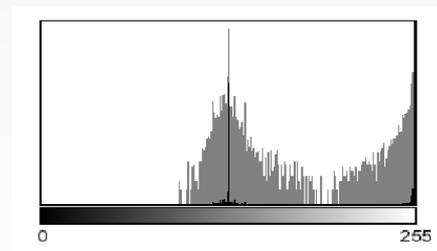
(a)



(b)



(c)



(d)

## Algoritmo:

```
1 public class Compute_Histogram implements PlugInFilter {
2
3     public int setup(String arg, ImagePlus img) {
4         return DOES_8G + NO_CHANGES;
5     }
6
7     public void run(ImageProcessor ip) {
8         int[] H = new int[256]; // histogram array
9         int w = ip.getWidth();
10        int h = ip.getHeight();
11
12        for (int v = 0; v < h; v++) {
13            for (int u = 0; u < w; u++) {
14                int i = ip.getPixel(u,v);
15                H[i] = H[i] + 1;
16            }
17        }
18        ... //histogram H[] can now be used
19    }
20
21 } // end of class Compute_Histogram
```

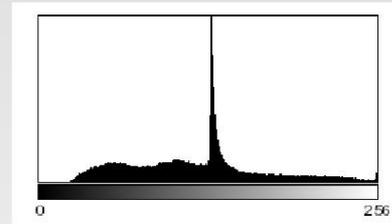
## Algoritmo de histograma de tamanho definido:

```
1  int[] binnedHistogram(ImageProcessor ip) {
2      int K = 256; // number of intensity values
3      int B = 32; // size of histogram, must be defined
4      int[] H = new int[B]; // histogram array
5      int w = ip.getWidth();
6      int h = ip.getHeight();
7
8      for (int v = 0; v < h; v++) {
9          for (int u = 0; u < w; u++) {
10             int a = ip.getPixel(u, v);
11             int i = a * B / K; // integer operations only!
12             H[i] = H[i] + 1;
13         }
14     }
15     // return binned histogram
16     return H;
17 }
```

## Histograma de imagens coloridas



(a)



(b)  $h_{Lum}$



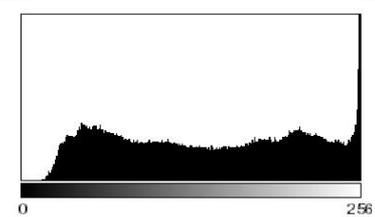
(c) R



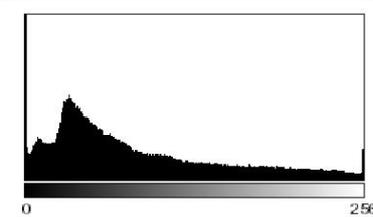
(d) G



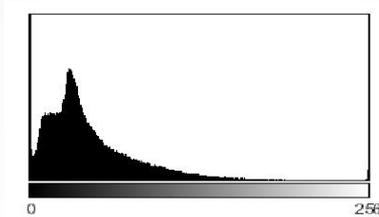
(e) B



(f)  $h_R$

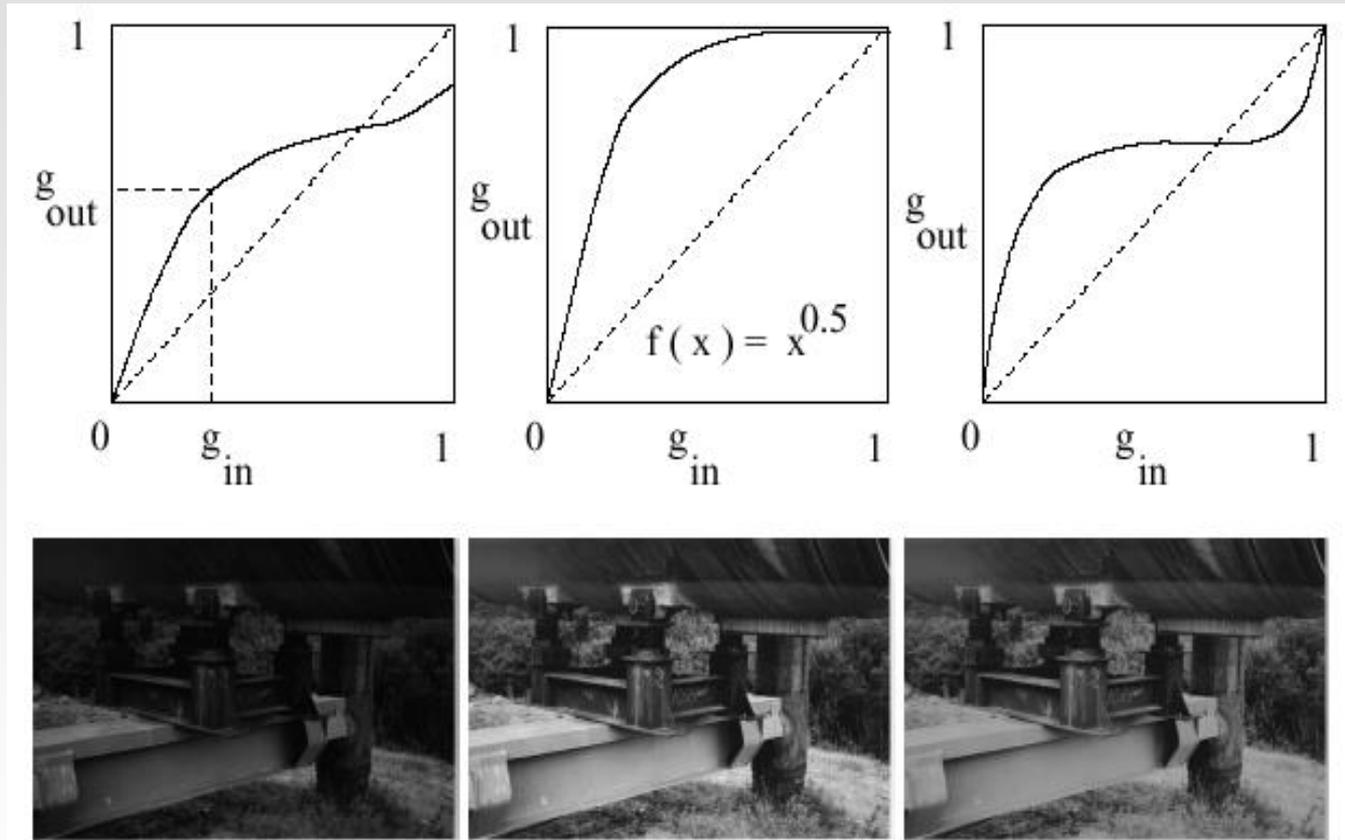


(g)  $h_G$



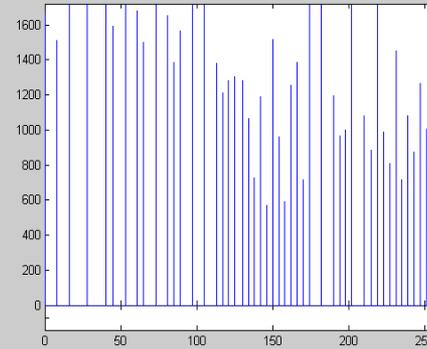
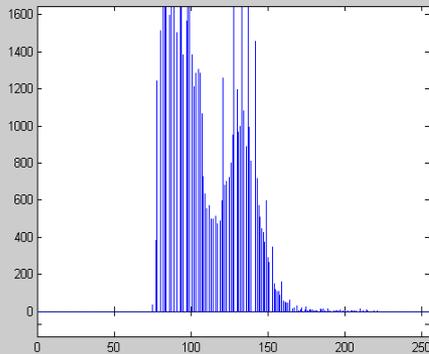
(h)  $h_B$

# Transformação de níveis de cinza

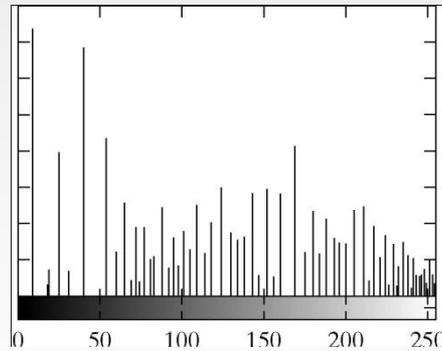
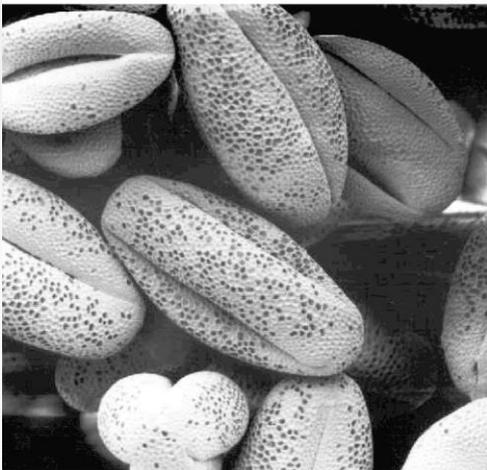
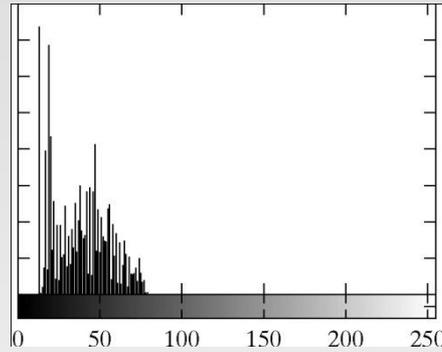


$$\longrightarrow f(x) = O_{\min} + \frac{O_{\max} - O_{\min}}{(I_{\max} - I_{\min})^\gamma} (x - I_{\min})^\gamma$$

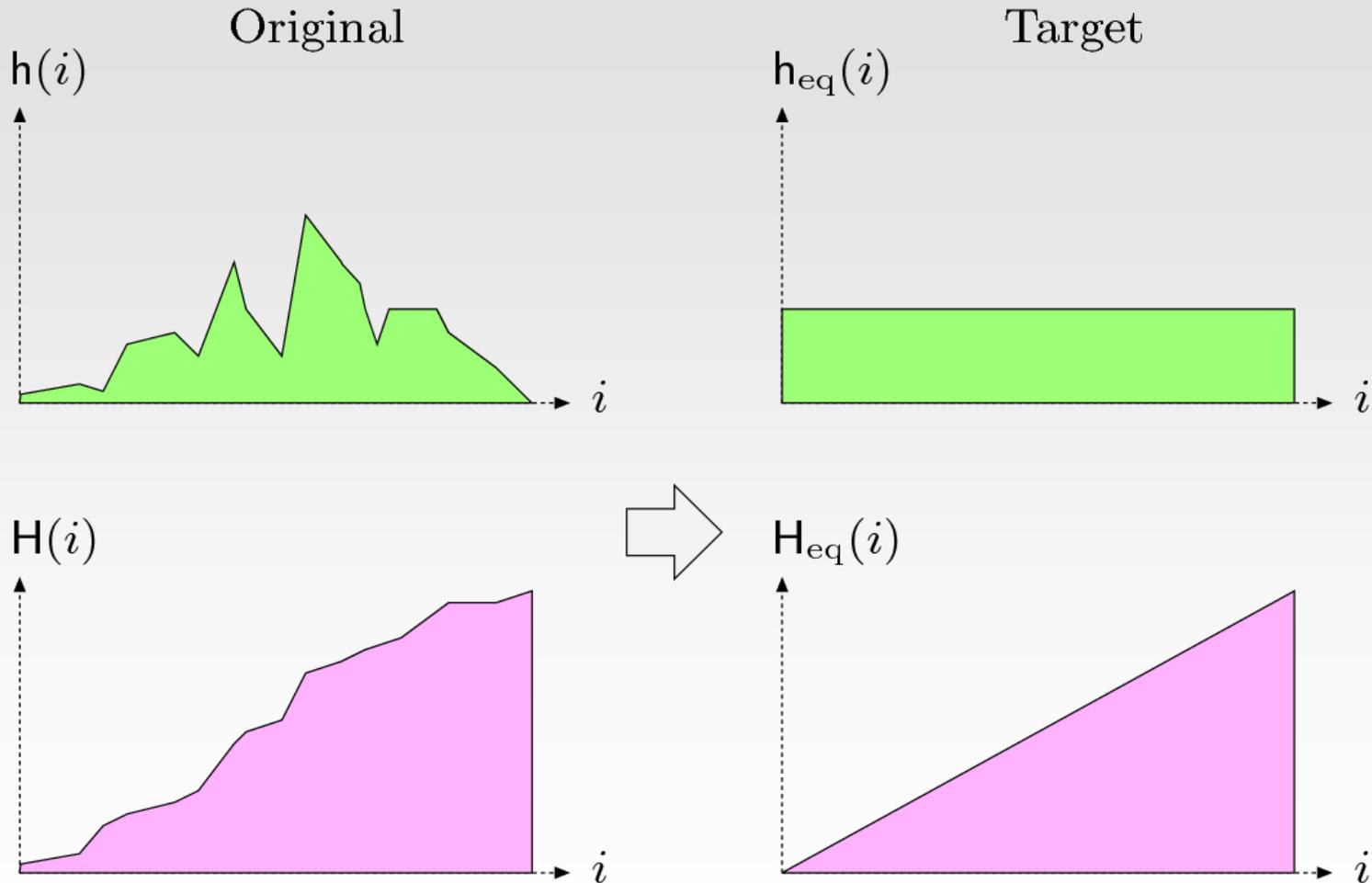
# Equalização de histograma



# Equalização de histograma



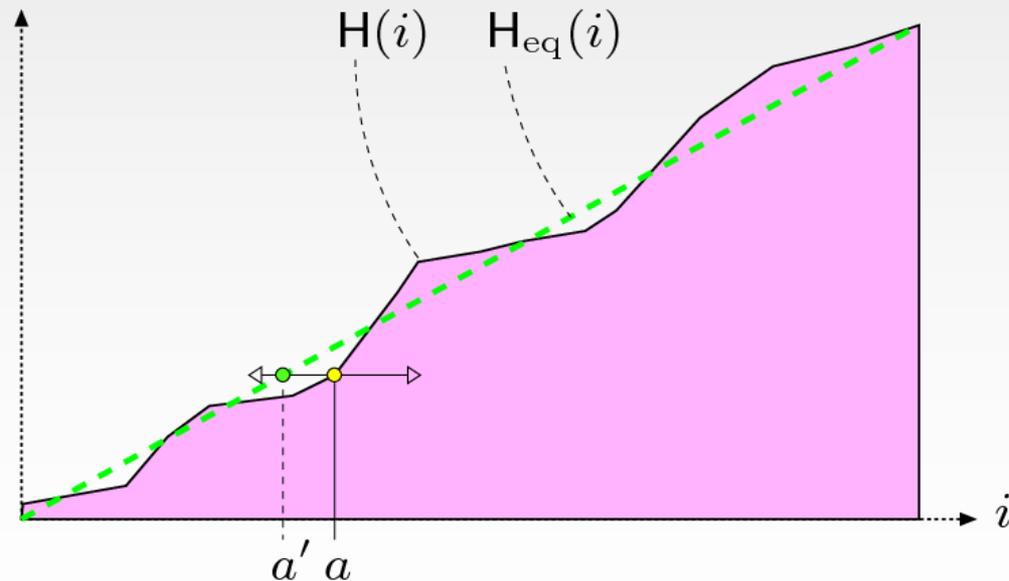
# Equalização de histograma



# Equalização de histograma

**Objetivo:** encontrar uma função que equalize o histograma

$$f_{\text{eq}}(a) = \left\lfloor H(a) \cdot \frac{K-1}{MN} \right\rfloor$$



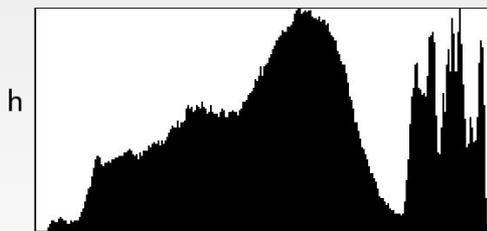
# Equalização de histograma



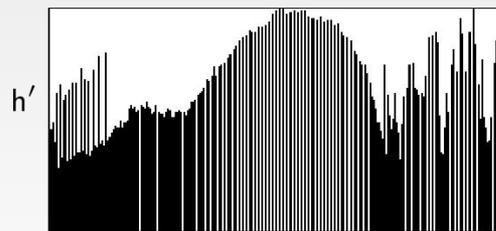
(a)



(b)



(c)



(d)



(e)



(f)

# Equalização de histograma

Assuma por um momento que níveis de intensidade são quantidades contínuas normalizadas para o escopo [0, 1],  
- função de densidade de probabilidade (PDF)

$$\sum_i h(i) = M \cdot N \quad p(i) = \frac{h(i)}{MN} \quad \sum_i p(i) = 1$$

$$P(i) = \frac{H(i)}{H(K-1)} = \frac{H(i)}{MN} = \sum_{j=0}^i \frac{h(j)}{MN} \\ = \sum_{j=0}^i p(j) \quad \text{para } 0 \leq i < K$$

Desejado:

$$p_s(s) = \begin{cases} 1 & \text{for } 0 \leq s \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

Transformação:

para  $k = 1, 2, \dots, L$  onde  $s_k$  é o valor de intensidade na imagem de saída (processada) que corresponde aos valores  $r_k$  na imagem de entrada

$$s_k = T(r_k) \\ = \sum_{j=1}^k p_r(r_j) \\ = \sum_{j=1}^k \frac{n_j}{n}$$

# Equalização de histograma

## Histograma cumulativo

```
1: CDF(h)
   Returns the cumulative distribution function  $P(i) \in [0, 1]$  for a discrete histogram  $h(i)$ , with  $i = 0, \dots, K-1$ .
2: Let  $K \leftarrow \text{Size}(h)$ 
3: Let  $n \leftarrow \sum_{i=0}^{K-1} h(i)$ 
4: Create table P of size  $K$ 
5: Let  $c \leftarrow h(0)$ 
6:  $P(0) \leftarrow c/n$ 
7: for  $i \leftarrow 1 \dots (K-1)$  do
8:      $c \leftarrow c + h(i)$                                 ▷ cumulate histogram values
9:      $P(i) \leftarrow c/n$ 
10: return P.
```

# Equalização de histograma

```
1  public void run(ImageProcessor ip) {
2      int w = ip.getWidth();
3      int h = ip.getHeight();
4      int M = w * h;    // total number of image pixels
5      int K = 256;     // number of intensity values
6
7      // compute the cumulative histogram:
8      int[] H = ip.getHistogram();
9      for (int j = 1; j < H.length; j++) {
10         H[j] = H[j-1] + H[j];
11     }
12
13     // equalize the image:
14     for (int v = 0; v < h; v++) {
15         for (int u = 0; u < w; u++) {
16             int a = ip.get(u, v);
17             int b = H[a] * (K-1) / M;
18             ip.set(u, v, b);
19         }
20     }
21 }
```

# Equalização de histograma

## Histograma cumulativo

```
1  public void run(ImageProcessor ip) {
2      int w = ip.getWidth();
3      int h = ip.getHeight();
4      int M = w * h;    // total number of image pixels
5      int K = 256;     // number of intensity values
6
7      // compute the cumulative histogram:
8      int[] H = ip.getHistogram();
9      for (int j = 1; j < H.length; j++) {
10         H[j] = H[j-1] + H[j];
11     }
12
13     // equalize the image:
14     for (int v = 0; v < h; v++) {
15         for (int u = 0; u < w; u++) {
16             int a = ip.get(u, v);
17             int b = H[a] * (K-1) / M;
18             ip.set(u, v, b);
19         }
20     }
21 }
```

# Equalização de histograma

$x_i$	$n_i$
0/7	790
1/7	1023
2/7	850
3/7	656
4/7	329
5/7	245
6/7	122
7/7	81

Histograma de imagem octal (3 bits de resolução): escopo 0...7

Tamanho:  $64 \times 64 = 4096$  pixels

# Equalização de histograma

$x_i$	$n_i$	$P_i = n_i/N$
0/7	790	0.19
1/7	1023	0.25
2/7	850	0.21
3/7	656	0.16
4/7	329	0.08
5/7	245	0.06
6/7	122	0.03
7/7	81	0.02

$P_i$  = frecuencia normalizada

# Equalização de histograma

$x_i$	$n_i$	$P_i = n_i/N$	$y' = F_i$
0/7	790	0.19	0.19
1/7	1023	0.25	0.44
2/7	850	0.21	0.65
3/7	656	0.16	0.81
4/7	329	0.08	0.89
5/7	245	0.06	0.95
6/7	122	0.03	0.98
7/7	81	0.02	1.00

$y_i$  = distribuição acumulada

# Equalização de histograma

$x_i$	$n_i$	$P_i = n_i/N$	$y' = F_i$	$y^1_j$
0/7	790	0.19	0.19	1/7
1/7	1023	0.25	0.44	3/7
2/7	850	0.21	0.65	5/7
3/7	656	0.16	0.81	6/7
4/7	329	0.08	0.89	6/7
5/7	245	0.06	0.95	7/7
6/7	122	0.03	0.98	7/7
7/7	81	0.02	1.00	7/7

$y^1_i = y_i$  aproximado a  $x_i$

# Equalização de histograma

$x_i$	$n_i$	$P_i = n_i/N$	$y' = F_i$	$y^1_j$	$P^1_j$
0/7	790	0.19	0.19	1/7	0.19
1/7	1023	0.25	0.44	3/7	0.25
2/7	850	0.21	0.65	5/7	0.21
3/7	656	0.16	0.81	6/7	
4/7	329	0.08	0.89	6/7	0.24
5/7	245	0.06	0.95	7/7	
6/7	122	0.03	0.98	7/7	
7/7	81	0.02	1.00	7/7	0.11

Repetindo o processo...

# Equalização de histograma

$x_i$	$n_i$	$P_i = n_i/N$	$y' = F_i$	$y^1_j$	$P^1_j$	$F^1_j$
0/7	790	0.19	0.19	1/7	0.19	0.19
1/7	1023	0.25	0.44	3/7	0.25	0.44
2/7	850	0.21	0.65	5/7	0.21	0.65
3/7	656	0.16	0.81	6/7		
4/7	329	0.08	0.89	6/7	0.24	0.89
5/7	245	0.06	0.95	7/7		
6/7	122	0.03	0.98	7/7		
7/7	81	0.02	1.00	7/7	0.11	1.00

# Equalização de histograma

$x_i$	$n_i$	$P_i = n_i/N$	$y' = F_i$	$y^1_j$	$P^1_j$	$F^1_j$	$y^2_j$
0/7	790	0.19	0.19	1/7	0.19	0.19	0/7
1/7	1023	0.25	0.44	3/7	0.25	0.44	2/7
2/7	850	0.21	0.65	5/7	0.21	0.65	4/7
3/7	656	0.16	0.81	6/7			5/7
4/7	329	0.08	0.89	6/7	0.24	0.89	6/7
5/7	245	0.06	0.95	7/7			7/7
6/7	122	0.03	0.98	7/7			7/7
7/7	81	0.02	1.00	7/7	0.11	1.00	7/7

# Equalização de histograma

$x_i$	$n_i$	$P_i=n_i/N$	$y'=F_i$	$y^1_j$	$P^1_j$	$F^1_j$	$y^2_j$	$P^2_j$
0/7	790	0.19	0.19	1/7	0.19	0.19	0/7	0.19
1/7	1023	0.25	0.44	3/7	0.25	0.44	2/7	0.25
2/7	850	0.21	0.65	5/7	0.21	0.65	4/7	0.21
3/7	656	0.16	0.81	6/7			5/7	0.16
4/7	329	0.08	0.89	6/7	0.24	0.89	6/7	0.08
5/7	245	0.06	0.95	7/7			7/7	
6/7	122	0.03	0.98	7/7			7/7	
7/7	81	0.02	1.00	7/7	0.11	1.00	7/7	0.11

# Equalização de histograma

$x_i$	$n_i$	$P_i=n_i/N$	$y'=F_i$	$y^1_j$	$P^1_j$	$F^1_j$	$y^2_j$	$P^2_j$	$F^2_j$
0/7	790	0.19	0.19	1/7	0.19	0.19	0/7	0.19	0.19
1/7	1023	0.25	0.44	3/7	0.25	0.44	2/7	0.25	0.44
2/7	850	0.21	0.65	5/7	0.21	0.65	4/7	0.21	0.65
3/7	656	0.16	0.81	6/7			5/7	0.16	0.81
4/7	329	0.08	0.89	6/7	0.24	0.89	6/7	0.08	0.89
5/7	245	0.06	0.95	7/7			7/7		
6/7	122	0.03	0.98	7/7			7/7		
7/7	81	0.02	1.00	7/7	0.11	1.00	7/7	0.11	1.00