



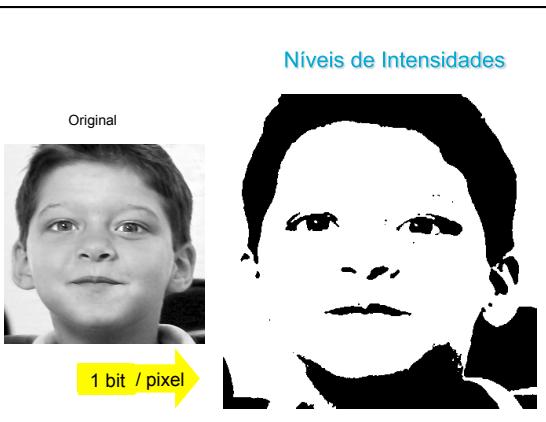
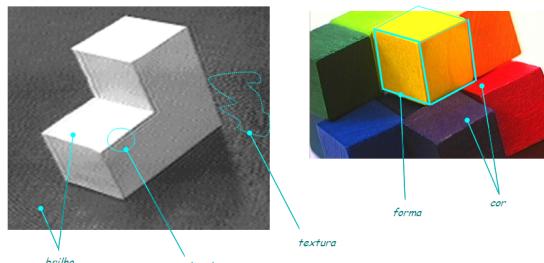
## Tentativa de Organização

- Aulas 1 e 2 (15 e 17 março) - Introdução, caracterização da imagem digital, brilho, contraste e análise de histograma, Python.
- Aulas 3 e 4 (22 e 24 de março) - Operações pontuais em imagens e topologia discreta.
- Aulas 5 e 6 (29 e 31 de março) – Operadores lineares e teoria da informação.
- Aulas 7 e 8 (5 e 7 de abril) – Aprendizado de máquina aplicado à imagens.
- Prova 1 (19 de abril) – Toda a matéria até esta data.
- Aulas 9 e 10 (26 e 28 de abril) – Morfologia matemática.
- Aulas 11 e 12 (3 e 5 de maio) – Morfologia e segmentação de imagens.
- Aulas 13 e 14 (10 e 12 de maio) - Representação de Fourier
- Aulas 15 e 16 (**17 e 19 de maio**) - Representação de Fourier
- Aulas 17 e 18 (24 e 26 de maio) – Atributos locais e globais
- Aulas 19 e 20 (31 de maio e 2 de junho) – Princípios físicos da visão
- Aulas 21 e 22 (7 e 9 de junho) - Princípios físicos da visão e cores
- Aula 23 e Prova 2 (21 e 23 de junho) - Textura e segunda prova
- Aulas 24 e 25 (28 e 30 de junho) - Visão de movimento

## Propriedades da imagem

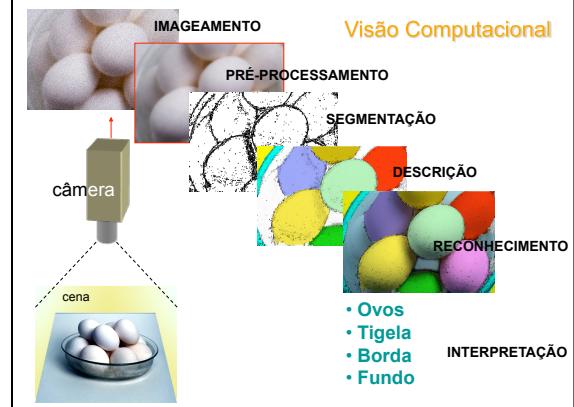
- **Atributos físicos**
  - Resolução espacial
    - De quantos pixels é essa imagem ?
  - Faixa dinâmica – ou resolução de amplitude
    - De quantos bits é essa imagem ?
  - Tamanho
    - Quanta memória ocupa essa imagem ?
      - Tamanho = (Resolução) x (Faixa Dinâmica)
- **Atributos perceptuais**
  - Qualidades percebidas pela visão

## Atributos perceptuais da visão

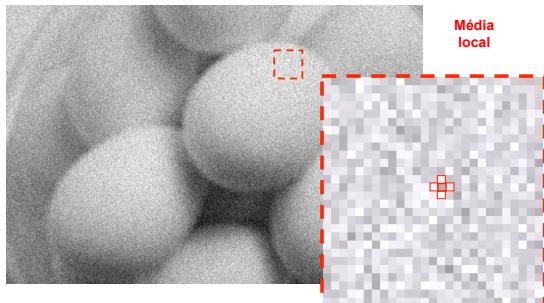


## Visão Computacional

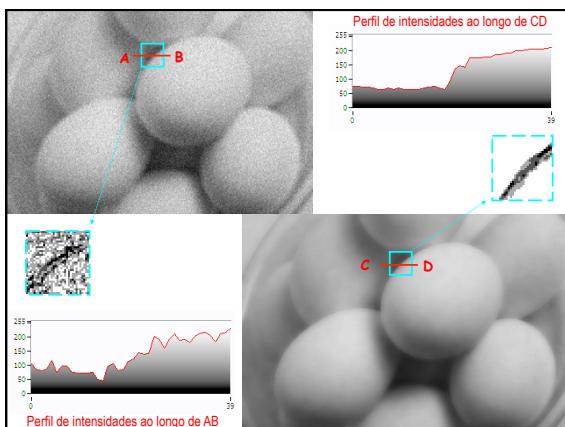
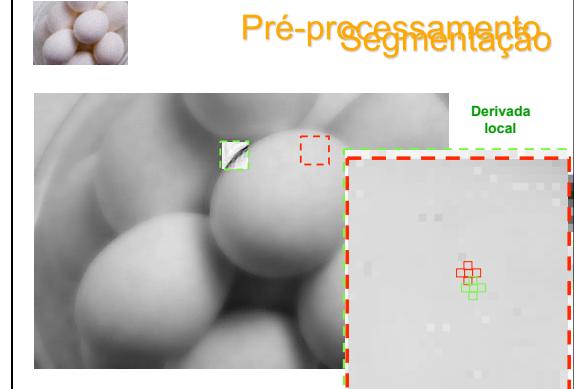
- Lida com diversos tipos de problemas
  - Inspeção, identificação, monitoração, rastreamento, mensuração, etc
- Busca representações e modelos adequados a cada tipo de problema
  - Problemas de aplicação tecnológica
  - Problemas de modelagem comportamental e/ou fisiológica

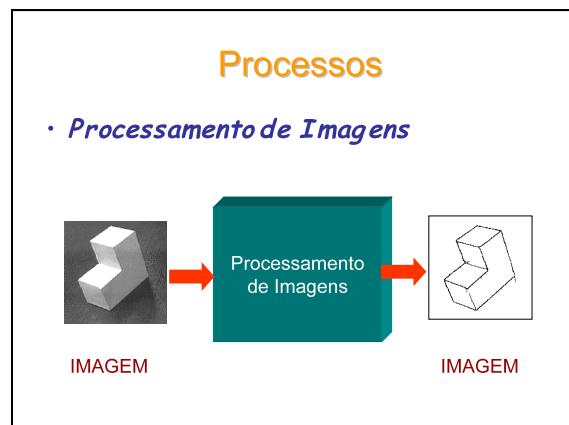
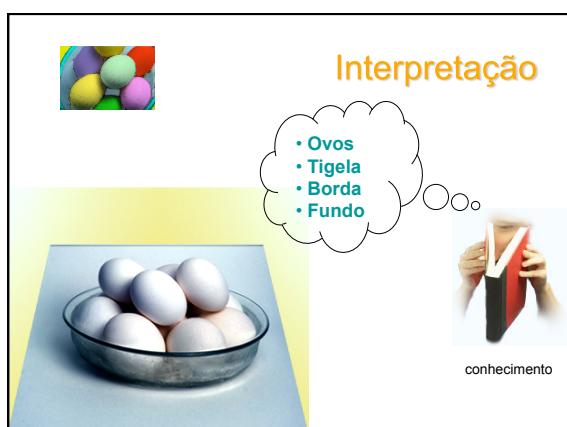
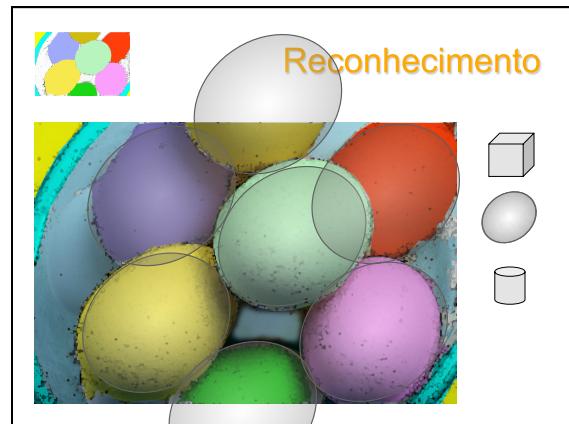
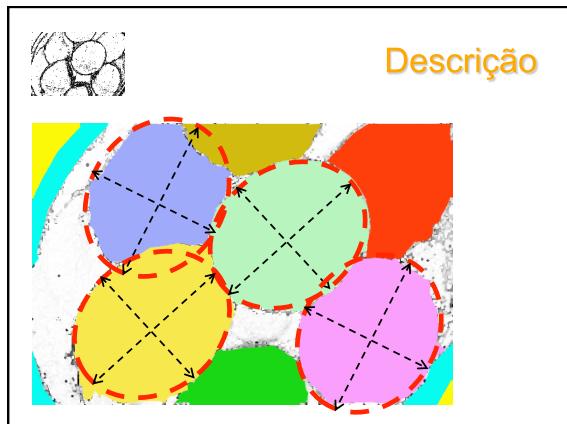


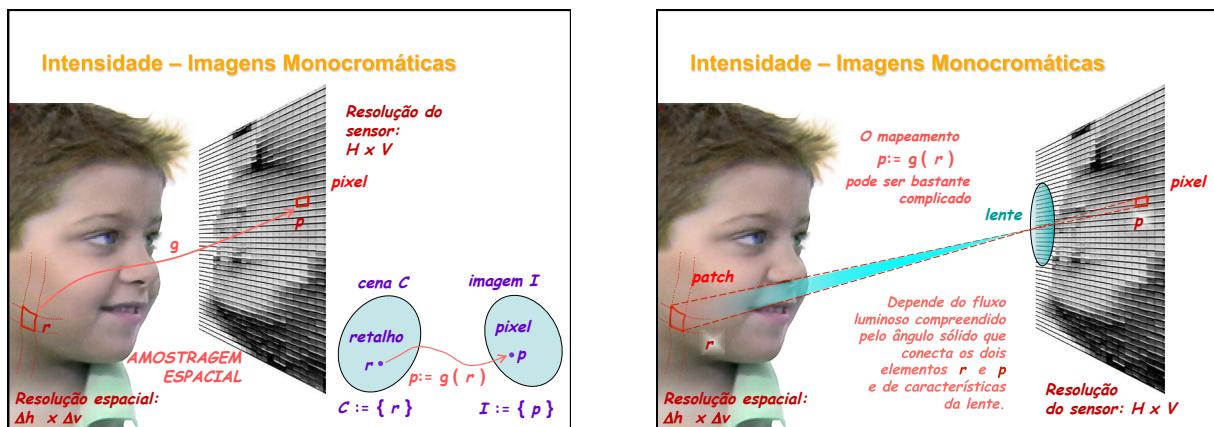
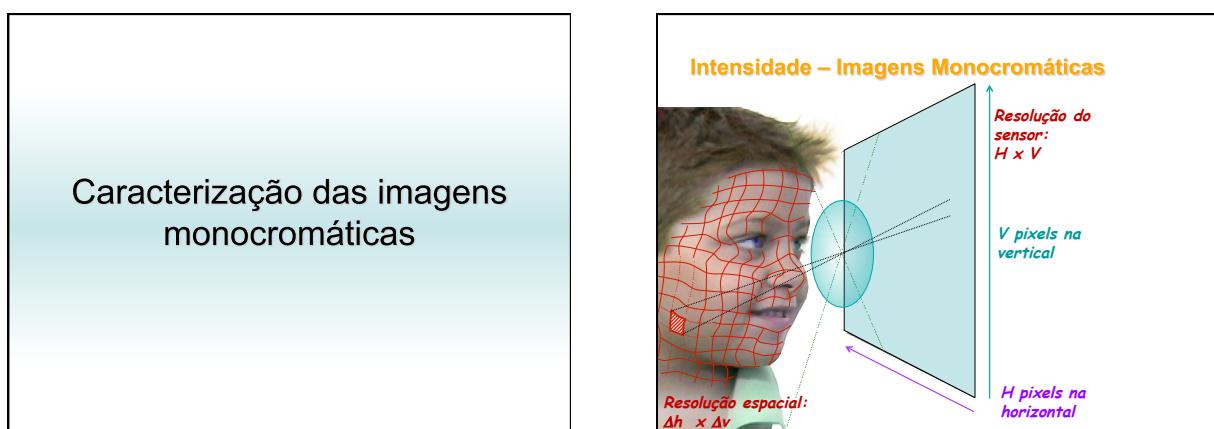
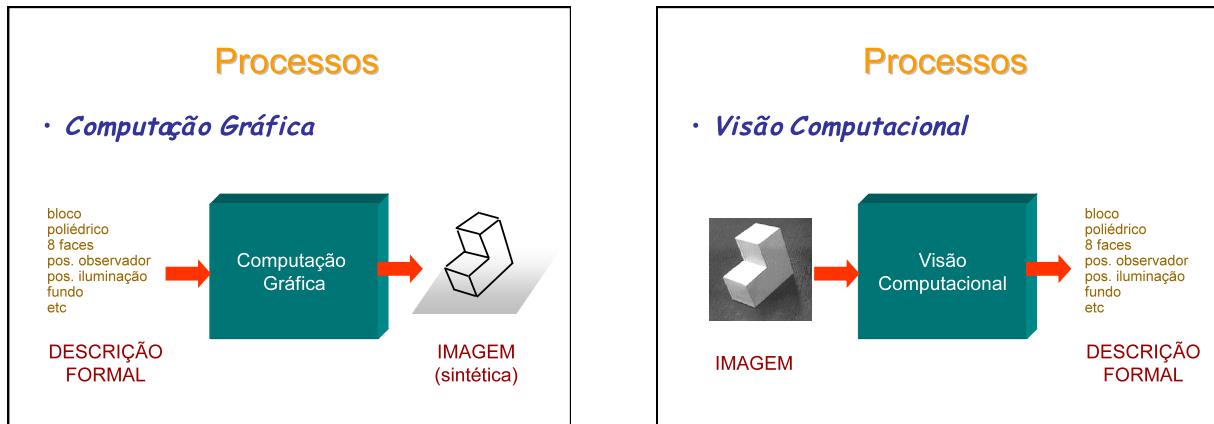
## Imageamento Pré-processamento

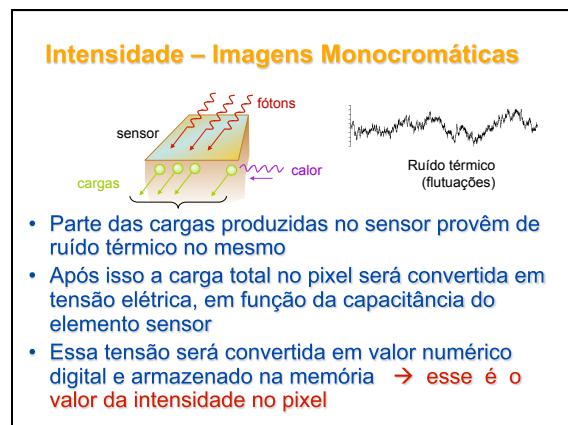
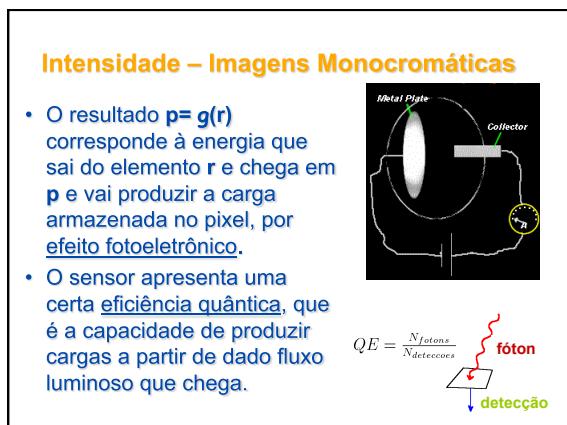
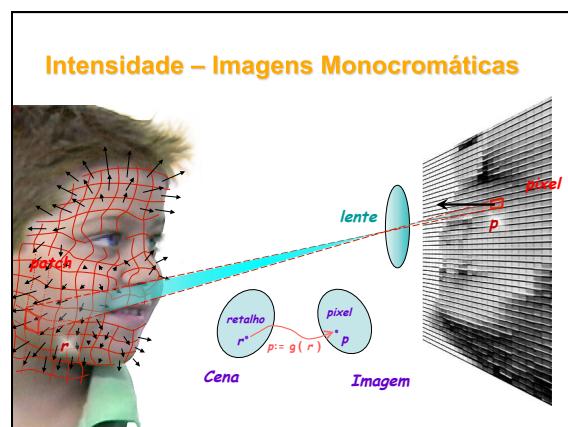
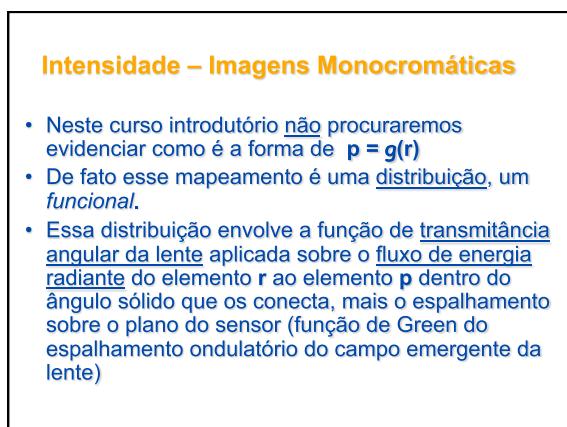
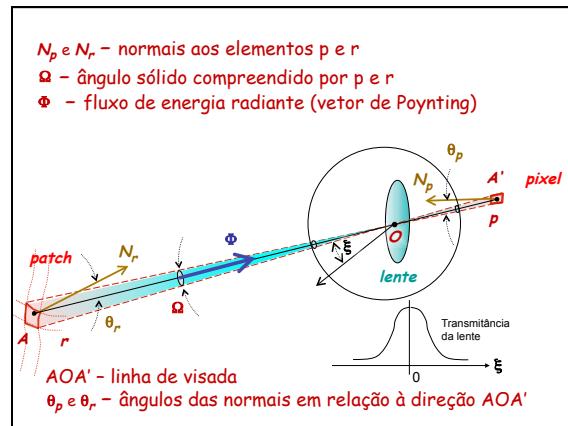
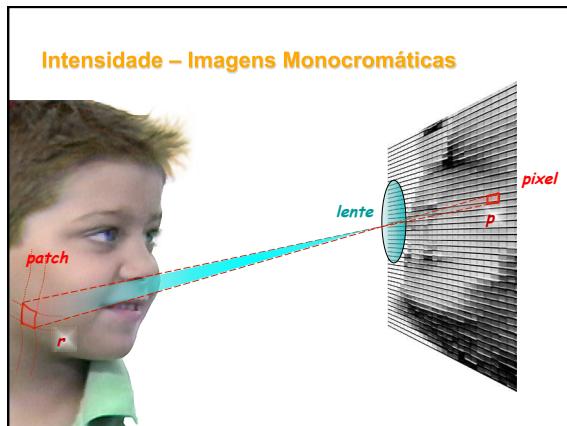


## Pré-processamento Segmentação









## Unidades de medida

- Radiometria**

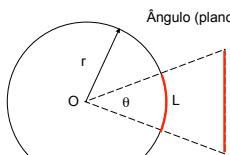
- Fluxo radiante  $\rightarrow W$  (Watt)
- Irradiância  $\rightarrow W / m^2$
- Intensidade radiante  $\rightarrow W / sr$  (sr  $\rightarrow$  sterorad)
- Radiânci  $\rightarrow (W / sr) / m^2$

$$1 \text{ watt} = 683 \text{ lumens}$$

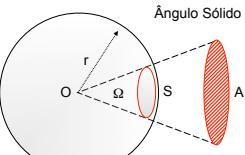
- Fotometria**

- Fluxo luminoso  $\rightarrow I$  (lúmen - lm)
- Iluminância  $\rightarrow I / m^2$  (lux - lx)
- Intensidade luminosa  $\rightarrow I / sr$  (candela - cd)
- Luminância  $\rightarrow I / sr / m^2$  (cd / m<sup>2</sup>)

## Ângulo Sólido



$$\theta = \frac{L}{r}$$

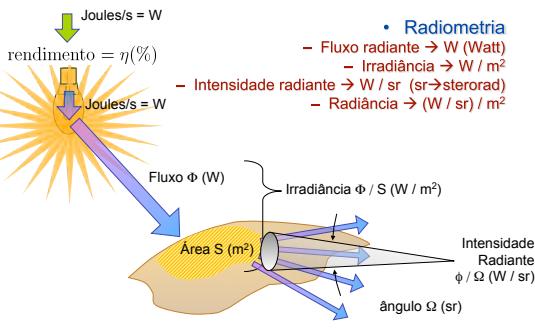


$$\Omega = \frac{A}{r^2}$$

- Exercício**

- Demonstre que o ângulo sólido que cobre a esfera toda vale  $4\pi$  (estero-radianos, sr).

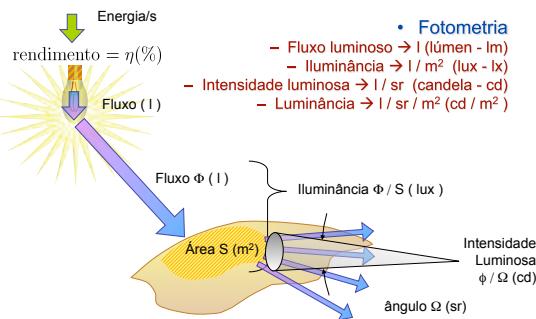
## Unidades Radiométricas



- Radiometria**

- Fluxo radiante  $\rightarrow W$  (Watt)
- Irradiância  $\rightarrow W / m^2$
- Intensidade radiante  $\rightarrow W / sr$  (sr  $\rightarrow$  sterorad)
- Radiânci  $\rightarrow (W / sr) / m^2$

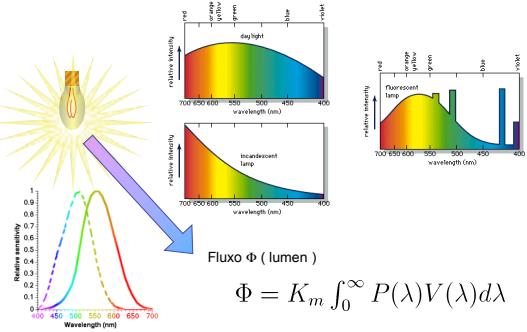
## Unidades Fotométricas



- Fotometria**

- Fluxo luminoso  $\rightarrow I$  (lúmen - lm)
- Iluminância  $\rightarrow I / m^2$  (lux - lx)
- Intensidade luminosa  $\rightarrow I / sr$  (candela - cd)
- Luminância  $\rightarrow I / sr / m^2$  (cd / m<sup>2</sup>)

## Unidades Fotométricas



## Fatos sobre intensidade

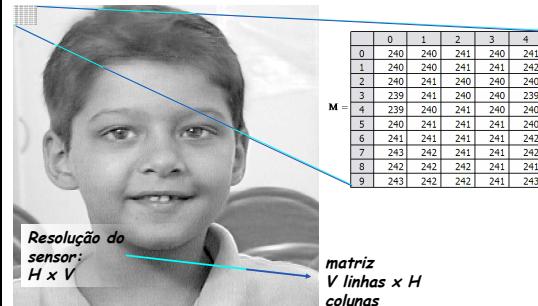
- Distribuição de luz natural:  $10000 \text{ cd/m}^2$
- Dist. Luz (amb. interno):  $100 \text{ cd/m}^2$
- Superfície clara (ambiente int.):  $50 \text{ cd/m}^2$
- Superfície escura (amb. interno):  $1 \text{ cd/m}^2$
- Faixa dinâmica requerida:  $1 \sim 10000 \text{ cd/m}^2$

Quantos bits por pixel seriam necessários para se discriminar valores nessa faixa dinâmica ?

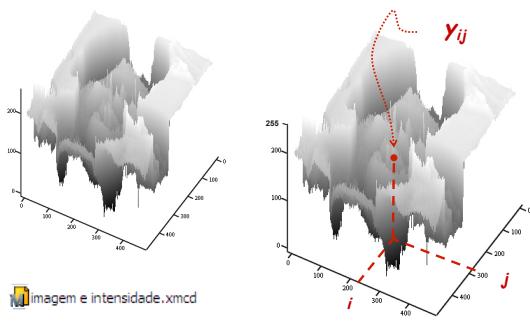
### Intensidade – Imagens Monocromáticas



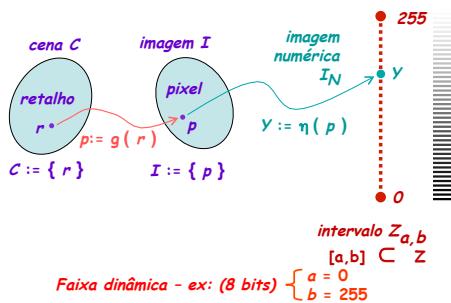
### Intensidade – Imagens Monocromáticas



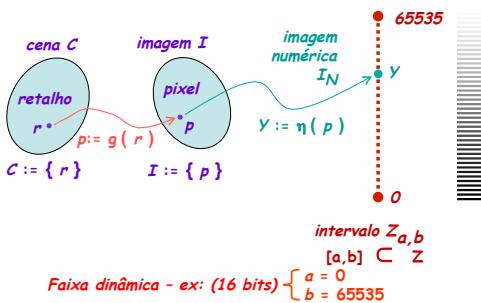
### Imagen digital



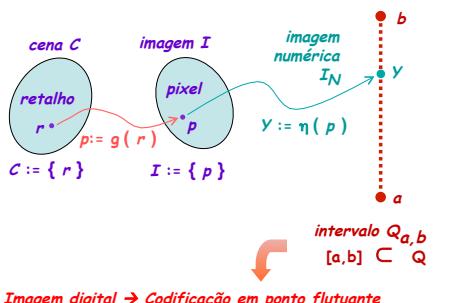
### Imagens Monocromáticas de 8 bits



### Imagens Monocromáticas de 16 bits



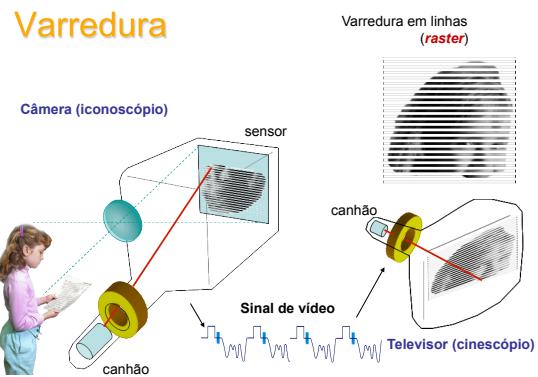
### Imagens Monocromáticas - Float



## Imagen → Sinal de Vídeo

- A tecnologia de imagem hoje empregada em computadores é **filha da tecnologia do vídeo em televisão**
- Em sua origem o vídeo em TV era produzido explorando-se as linhas da imagem formada em um sensor usando-se um feixe de elétrons
- A passagem do feixe de elétrons é chamada de **varredura**
- A informação resultante era um sinal variável no tempo, carregando o conteúdo de luminosidade em cada linha de varredura

## Varredura



## Varredura - 1

- Na câmera de TV (iconoscópio) a imagem era **focalizada** sobre um **sensor**
- O sensor acumulava cargas que eram **lidas** com um feixe
- Essa atividade chama-se **exploração**
- A exploração era feita **linha a linha**
- O processo de varrer a imagem linha a linha chama-se **rastering (rasterização)**

## Varredura - 2

- As linhas eram transmitidas ao receptor de TV via ondas curtas (MHz)
- O receptor exibe na tela do cinescópio também linha a linha, com um feixe de elétrons que excita material **fotoluminescente** aplicado sobre a tela (máscara de fósforo)
- O aparelho de TV e a câmera têm de estar **sincronizados** para que a imagem lida no emissor seja entendida no receptor
- Quando isso não acontecia ....

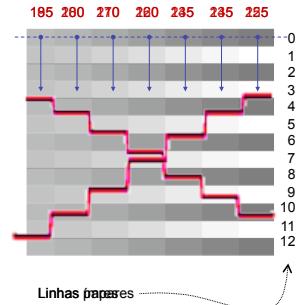
## Sinal de Vídeo (analógico)

A origem de diversos conceitos empregados hoje com imagens digitais provém da tecnologia analógica

- Muitos desses conceitos ainda são utilizados, bem como a própria tecnologia analógica (p.ex, em câmeras de segurança, emissoras de TV)
- As razões disso ser assim são:
  - Compatibilidade na troca de tecnologias
  - Mudança lenta dos conceitos
  - Preservação de conceitos julgados úteis
  - Custos "menores" da tecnologia analógica

## Sinal de Vídeo (analógico)

- Exemplo ilustrativo
  - Suponha que você tenha uma imagem composta da seguinte maneira: todas as linhas pares têm um degradê de cinzas em um sentido e as linhas ímpares têm o mesmo degradê só que em outro sentido



### Sinal de Vídeo (análogo)

- Isso poderia ser o sinal de vídeo, entretanto...
  - É preciso incluir a informação de sincronismo para que sua TV saiba como a câmera produziu a imagem

### Sinal de Vídeo (análogo)

O sinal de vídeo contém as informações das intensidades das linhas (as escadas) e pulsos de sincronismo, indicando quando começa uma nova linha :

### Exibição do vídeo

- O vídeo é exibido lendo-se eletronicamente o sinal de vídeo e transformando-o no movimento do feixe de elétrons que varre a tela

### Varredura Rasterizada

### Varredura - 3

- O sinal de vídeo transporta da câmera para o receptor a informação de sincronismo

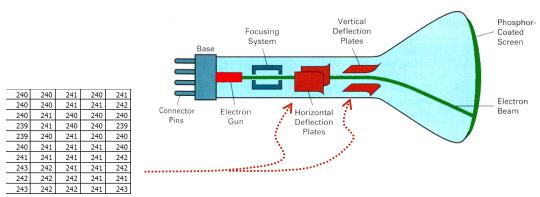
### Vídeo digital

- O vídeo é exibido lendo-se os dados da memória de vídeo e transformando-os no movimento do feixe de elétrons que varre a tela (feito pelo controlador de vídeo + circuitos do monitor ou display)

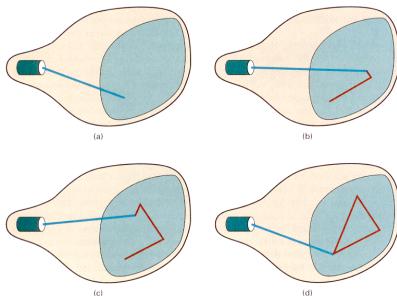
240	240	241	240	241
240	240	241	241	242
240	241	240	240	240
239	241	240	240	239
240	240	241	240	240
240	240	241	241	240
241	241	241	241	242
243	242	241	241	242
242	242	241	241	241
243	240	242	241	243

## Vídeo digital

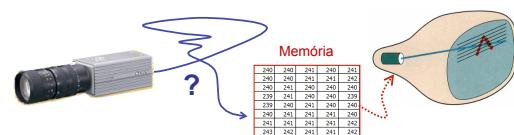
- Também existem monitores, muitas vezes de tecnologia usando tubo de raios catódicos (TRC) que operam de forma vetorial → varredura vetorializada



## Varredura Vetorializada (Vector Graphics)



## ... mas, como os dados do vídeo digital chegam à memória?



- Depende da fonte
  - Câmera analógica x câmera digital
- Depende da transmissão
  - Analógica – NTSC, PAL, outros
  - Digital – Ethernet, USB, Firewire, outros

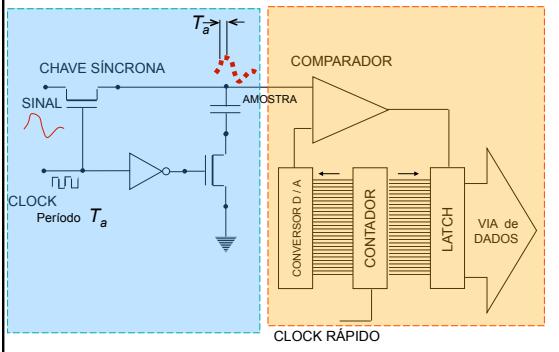
## Câmera Analógica

- O sinal de vídeo é analógico
  - É transportado em um cabo coaxial
- Precisa ser digitalizado
  - Placa de captura de vídeo analógico
    - Amostrador
    - Conversor Analógico/Digital



Amostrador e Conversor A/D

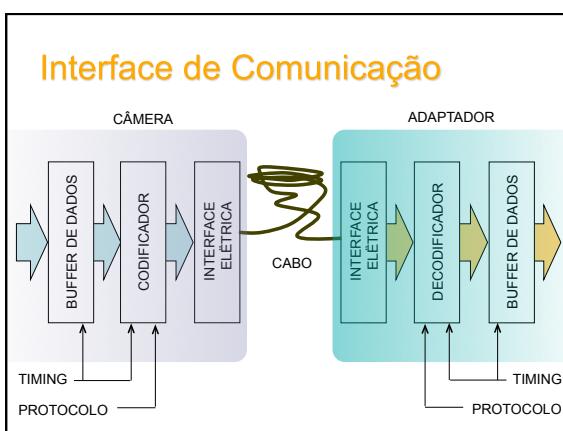
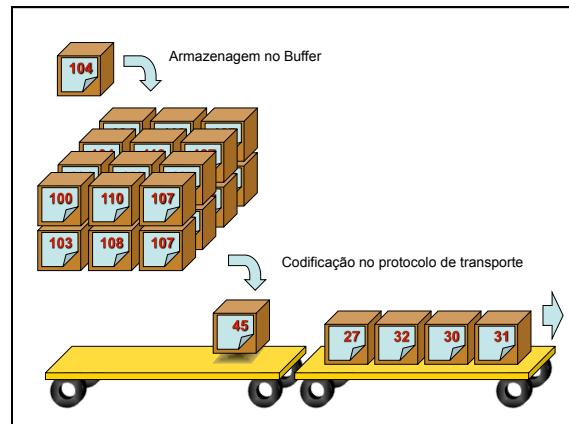
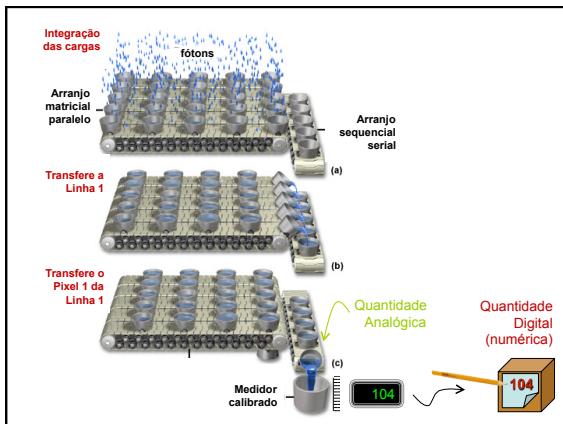
## Amostrador e Conversor A/D



## Câmera Digital

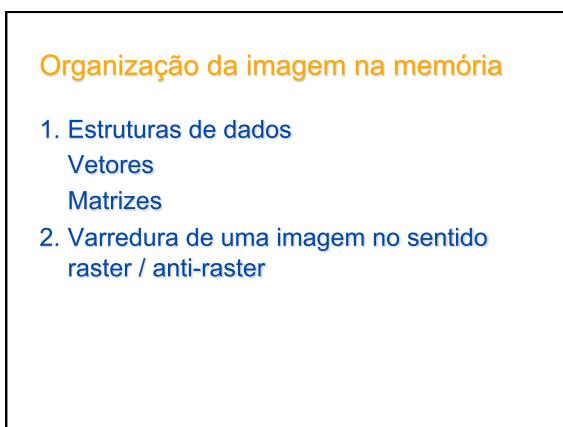
- O sinal é digitalizado na câmera
- Os dados numéricos são transmitidos digitalmente segundo um protocolo dependente da interface usada
  - Ethernet → TCP
  - USB → Serial assíncrono
  - Firewire → Pacotes IEEE-1394





### Interfaces de Vídeo Digital

Interface	Banda (MBytes/s)	Alcance Máximo (m)
Firewire - IEEE-1394a	50	4,5
USB 2.0	60	5 (sem hub)
Firewire-IEEE-1394b	100	4,5 (Cobre), 100 (fibra op.)
GigE	125	110
Camera Link	400	10
Barramento	Banda (MBytes/s)	-
PCI	128	-
PCI Express	4096	-

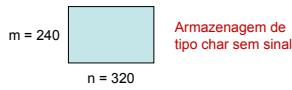


- ### Armazenagem na memória
- A armazenagem da imagem na memória é realizada através da alocação de um array
    - Unidimensional → vetor
    - Bidimensional → matriz
  - Os arrays são alocados com o tipo de dado que corresponde ao tipo da imagem → inteiro de 8, 16, 32 ou 64 bits ou ainda ponto flutuante

## Armazenagem como Vetor

- A imagem pode ser armazenada como vetor (array unidimensional)

– Exemplo: (em C)



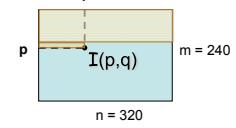
```
unsigned int m = 240, n = 320 ;
unsigned char *pimg ;
pimg = (unsigned char *) malloc(sizeof(char)*m*n) ;
if (pimg==NULL) { ERRO }
```

## Acesso como Vetor

- Acesso a um ponto  
 $P = I(p,q) , ( 0 \leq p , q \leq n )$   
 da imagem:

$$\text{value} = *(\text{pimg} + p * n + q)$$

para se chegar ao ponto P,  
 pulam-se p linhas e mais q colunas



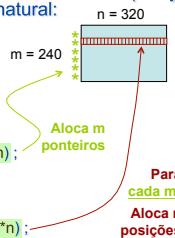
Toda operação pontual sobre a imagem armazenada como vetor pode ser feita com um único loop

## Armazenagem como Matriz

- A imagem também pode ser armazenada como matriz (array bidimensional), que é a estrutura mais natural:

```
unsigned int i, m = 240, n = 320 ;
unsigned char **ppimg, *pimg ;
ppimg = (unsigned char **) malloc(sizeof(char**) * m) ;
if (ppimg == NULL) { Tratamento de erro }

for (i=0; i<m; i++) {
    ppimg[i] = (unsigned char *) malloc(sizeof(char*) * n) ;
    if (ppimg[i]==NULL) { Tratamento de erro mais complexo }
}
```



## Acesso como matriz

- Matriz: uma operação pontual sobre um pixel  $I(p,q)$  da imagem pode ser feita com um simples acesso a  $\text{ppimg}$ ,

$$\text{value} = \text{ppimg}[p][q]$$

Toda operação pontual sobre a imagem armazenada como matriz é ser feita com dois loops

## Arquivamento de imagens

- Há diversos formatos para se guardar as imagens em arquivos
- Os formatos conhecidos têm as mais diversas origens e razões
- Existem formatos para exploração raster e para exploração vetorizada

## Exemplo : Sun raster image header

```
• struct rasterfile {
    int ras_magic;
    int ras_width;
    int ras_height;
    int ras_depth;
    int ras_length;
    int ras_type;
    int ras_maptype;
    int ras_maplength;
}
#define RAS_MAGIC 0x59a66a95
```

## Sun raster

- /\* Sun supported ras\_type's \*/
  - #define RT\_OLD 0 /\* Raw pixrect image in 68000 byte order \*/
  - #define RT\_STANDARD 1 /\* Raw pixrect image in 68000 byte order \*/
  - #define RT\_BYTE\_ENCODED 2 /\* Run-length compress. of bytes \*/
  - #define RT\_FORMAT\_RGB 3 /\* XRGB or RGB instead of XBGR or BGR \*/
  - #define RT\_FORMAT\_TIFF 4 /\* iff <-> standard rasterfile \*/
  - #define RT\_FORMAT\_IFF 5 /\* iff (TAAC format) <-> standard rasterfile \*/
  - #define RT\_EXPERIMENTAL 0xffff /\* Reserved for testing \*/
  - /\* Sun registered ras\_maptype's \*/
    - #define RMT\_RAW 2 /\* Sun supported ras\_maptype's \*/
    - #define RMT\_NONE 0 /\* ras\_maplength is expected to be 0 \*/
    - #define RMT\_EQUAL\_RGB 1 /\* red[ras\_maplength/3],green[],blue[] \*/

## Sun raster

- **ras\_length** field é o comprimento dos dados (comprimento do arquivo menos o comprimento do cabeçalho e do colormap)
  - Este valor é zero algumas vezes por questões de compatibilidade.
- **ras\_maptype** e **ras\_maplength** contém o tipo e o comprimento do mapa de cores (colormap)
  - **ras\_maptype** pode ser **RMT\_NONE**, **RMT\_RAW**, ou **RMT\_EQUAL\_RGB**
- Note que, independente de width, as linhas são armazenadas em múltiplos de 16bits.

## Radiometria e Fotometria

- **Radiometria**
  - medidas de radiação emitida por uma fonte ou espalhadas por uma superfície
- **Fotometria**
  - medidas de radiação emitida por uma fonte ou espalhadas por uma superfície percebidas pelo ser humano
  - A fotometria é perceptual
    - Os equipamentos fotométricos são calibrados de acordo com parâmetros perceptuais medidos de forma sistemática

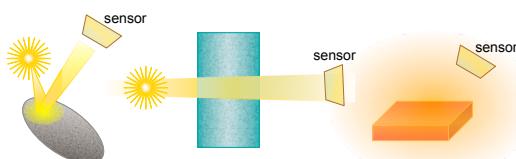
## Mapa de intensidades



- Estamos estudando as propriedades radiométricas das cenas
  - Manifestam-se nos valores dos pixels da imagem da cena
  - **Valores das intensidades nos pixels**
  - **Dependem da faixa dinâmica da imagem**
    - Imagens Monocromáticas: 8, 10 e 16 bits/pixel
    - Eventualmente → ponto flutuante

## Mapa de intensidades

- O que aparece nos pixels da imagem é o que foi captado pelo sensor (da câmera)
  - Espalhado pelas superfícies → Refletâncias
  - Transmitido através dos materiais → Transmitâncias
  - Emitidos pelas superfícies → Emitâncias



## Mapa de intensidades



- As intensidades dos pixels traduzem:
  - As propriedades da iluminação da cena
    - O sensor capta a quantidade de luz proveniente dos diversos pontos e direções
  - As propriedades dos materiais da cena
    - O sensor mede as radiâncias correspondentes aos diversos tipos de materiais → refletores, transmissores e emissores de radiação

## Mapas de intensidades

- Propriedades radiométricas dos mapas de intensidades (monocromáticas)
  - Brilho → qualidade subjetiva da imagem ser mais clara ou mais escura
  - Contraste → qualidade subjetiva das regiões da imagem apresentarem fronteiras de transição de luminosidade
    - São qualidades **GLOBAIS**, da imagem
    - Apesar do brilho e o contraste serem subjetivos, podemos definir medidas objetivas dos mesmos

## Mapas de Intensidades - exemplos



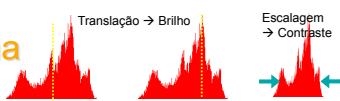
## Mapas de Intensidades - exemplos



## Mapas de Intensidades



## Histograma

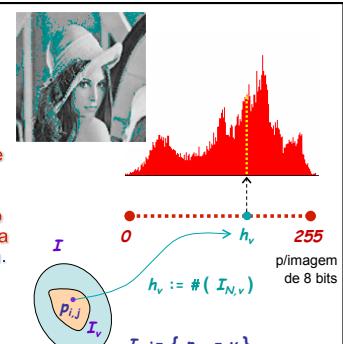


- Balancear as qualidades visuais de uma imagem (monocromática) significa
  - modificar consistentemente propriedades globais da imagem que produzem melhor julgamento da qualidade (perceptual)
    - Como brilho, contraste
- A busca de um método objetivo para tanto começa com a análise de como se distribuem os valores dos pixels
  - Isto é, seus tons monocromáticos

## Histograma

- É uma relação que mapeia, para cada valor de intensidade que um pixel possivelmente possa ter, o número de vezes em que ela aparece na imagem.

$I \rightarrow$  imagem  
 $I_v \rightarrow$  subconjunto da imagem  
 $h_v \rightarrow$  cardinalidade de  $I_v$



## Construção do histograma

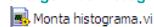
- O histograma é uma tabela das frequências de cada valor ou faixa de valores de intensidade nos pixels da imagem.

– Pode ser apresentado graficamente



– Demonstração

▪ Montagem de histograma



Monta histograma.vi

v	H
0	1
1	0
2	0
3	3
4	1
5	7
6	3
7	2
8	4
9	6
10	8
11	10
12	9
13	7
14	1
15	2

## Construção do Histograma

- Dividir o intervalo de dados em classes
  - Contar quantos dados existem em cada classe
- No caso de imagens cujos tons são valores inteiros, usualmente as classes são cada valor de intensidade presente
- Ex: imagem de 8 bits/pixel → 256 classes
  - imagem de 4 bits/pixel → 16 classes

## Construção do Histograma

- Exemplo em C:

```
void histograma(unsigned char *pimg, unsigned int size,
               unsigned int *hist) {
    unsigned int p;

    for (p=0; p<256; p++) hist[p] = 0; } Inicialização de hist

    for (p=0; p<size; p++)
        hist[pimg[p]]++; }
```

## Exemplos de histogramas

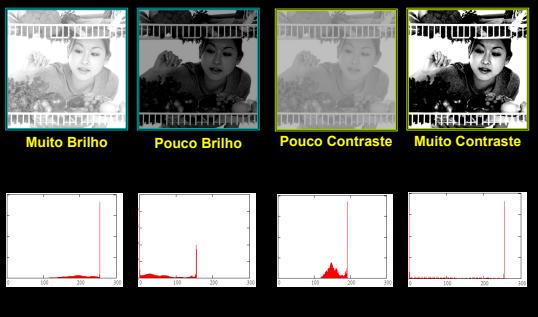
### – Exemplo 2

▪ [Estudos de casos](#)

▪ Compararemos os histogramas de imagens que apresentem brilhos e contrastes bastante distintos, de modo a aprendermos a interpretar a relação entre propriedades dos histogramas e essas qualidades perceptuais expressas nas imagens

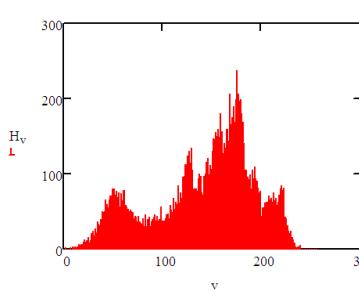
▪ Alguns dos casos estão nos slides seguintes

## Exemplos de Histogramas



## Histograma – outro exemplo

$$H_v := n\_pixels(v)$$



m := rows(I) m = 128  
n := cols(I) n = 128  
N := m·n N = 16384  
b := 8 bits  $2^b = 256$

## Comparação de Histogramas

Consideremos seguintes imagens



I → Lena

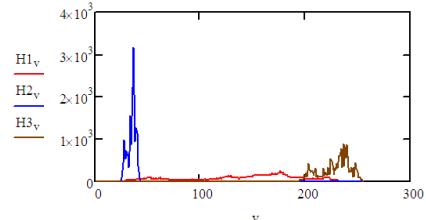
I2 → Lena dark

I3 → Lena bright

As imagens I2 e I3 são mais homogêneas que I. Notamos que elas apresentam menos **contraste** do que a I. Das três imagens, I3 é a que apresenta maior luminosidade, ou **brilho**. A mais escura é I2.

## Comparação dos Histogramas

$H1 := \text{histo}(I)$     $H2 := \text{histo}(I2)$     $H3 := \text{histo}(I3)$

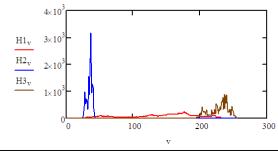


I → Lena

I2 → Lena dark

I3 → Lena bright

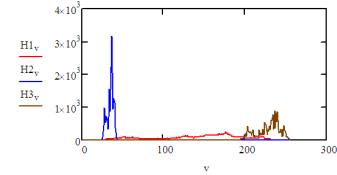
$H1 := \text{histo}(I)$     $H2 := \text{histo}(I2)$     $H3 := \text{histo}(I3)$



Dessa forma, só podemos fazer uma comparação qualitativa.

## Comparação dos Histogramas

- Para melhor comparar as imagens a partir de seus histogramas, introduziremos alguns parâmetros que caracterizam quantitativamente o histograma.



## Parâmetros do Histograma

- Os histogramas podem ser caracterizados por dois parâmetros (entre outros):
  - Parâmetro de posição:
    - **Média** → indica a posição (centralização) do histograma
  - Parâmetro de dispersão:
    - **Desvio-padrão** → indica a largura (dispersão, espalhamento) do histograma

## Parâmetros do Histograma

### Média

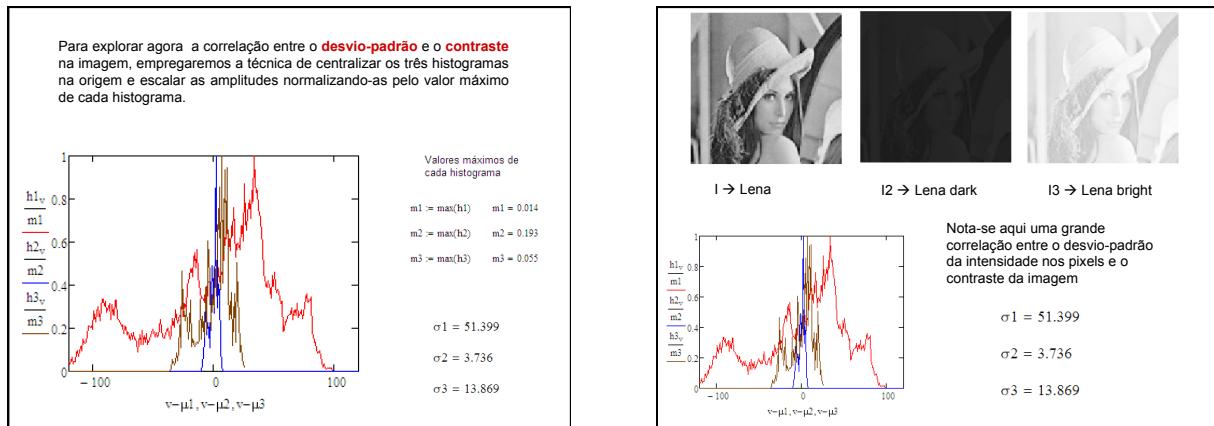
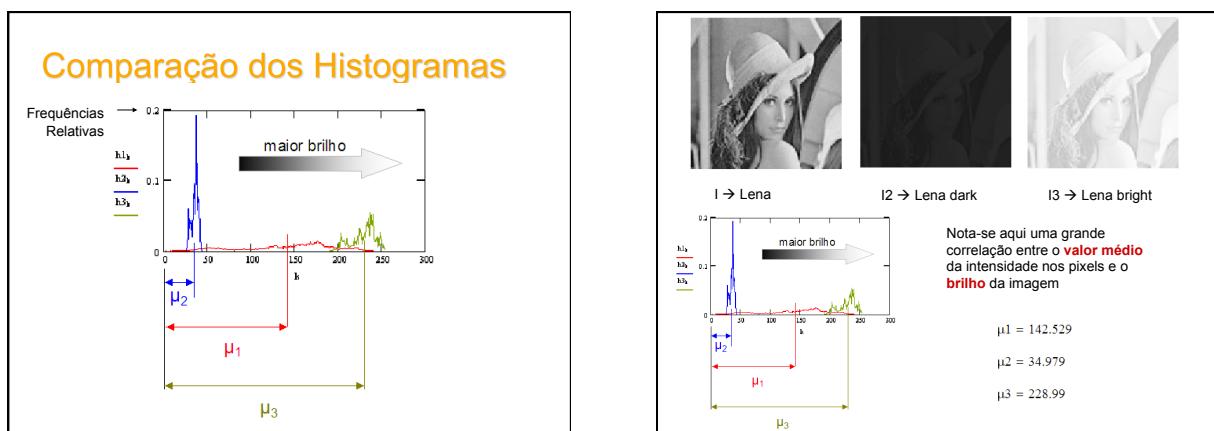
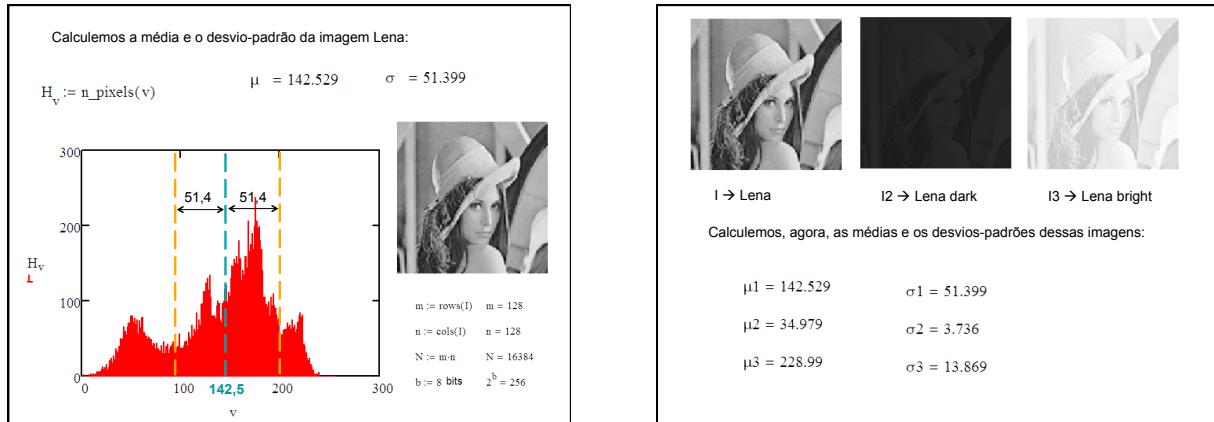
$$\mu = \frac{1}{N} \sum_{v=0}^{v_{\max}} (v H_v)$$

$v_{\max} = 2^b \rightarrow b = \text{nº de bits}$   
Para 8 bits →  $v_{\max} = 255$

Nessas expressões estão indicados, de fato, os chamados **estimadores** (de máxima verossimilhança) da média e do desvio-padrão. Essas expressões são livres de vício de estimação.

### Desvio-padrão

$$\sigma = \sqrt{\frac{\sum_{v=0}^{v_{\max}} [(v-\mu)^2 H_v]}{N-1}}$$



## Exercício sobre histogramas

- O próximos slides apresentarão conjuntos formados por 4 imagens e 4 histogramas cada um
- Associe a cada imagem de cada conjunto o seu respectivo histograma

