

Fundamentos de Imagem Digital

Prof. Alexandre Xavier Falcão

Aula 02

1 Representação em memória

Uma imagem pode ser armazenada em memória como um arranjo de spels. Abaixo seguem algumas possibilidades para imagens cinza 2D, 3D (imagens médicas), colorida, e vídeo digital.

```
typedef struct _grayimage {
    int  nx,ny;    /* dimensoes da imagem */
    int  **val;    /* matriz com os valores dos pixels */
    float dx,dy;  /* tamanho do pixel em unidades de comprimento */
    char unid[10]; /* unidade de comprimento */
} GrayImage;

typedef struct _medicalimage {
    int  nx,ny,nz; /* dimensoes da imagem */
    int  ***val;   /* volume com os valores dos voxels */
    float dx,dy,dz; /* tamanho do voxel em unidades de comprimento */
    char unid[10]; /* unidade de comprimento */
} MedicalImage;

typedef struct _cor {
    int val[3];
} Cor;

typedef struct _colorimage {
    int  nx,ny;    /* dimensoes da imagem */
    Cor **cor;     /* matriz com a cor dos pixels */
    float dx,dy;  /* tamanho do pixel em unidades de comprimento */
    char unid[10]; /* unidade de comprimento */
} ColorImage;
```

```

typedef struct _videodigital {
    int    nx,ny,nt;    /* dimensoes espaciais e temporal do video */
    Cor ***cor;        /* sequencia de quadros com a cor dos pixels */
    float dx,dy;       /* tamanho do pixel em unidades de comprimento */
    float dt;          /* distancia entre quadros em unidades de tempo */
    char ud[10],ut[10]; /* unidades de comprimento e tempo */
} VideoDigital;

```

O armazenamento em memória também pode ser vetorial. Neste caso, o índice p do vetor se relaciona com as coordenadas espaciais (x, y, z) por

- Caso 2D (Figura 1)

$$p = x + n_x * y \quad (1)$$

$$x = p \% n_x \quad (2)$$

$$y = p / n_x \quad (3)$$

- Caso 3D

$$p = x + n_x * y + n_x * n_y * z \quad (4)$$

$$z = p / (n_x * n_y) \quad (5)$$

$$j = p \% (n_x * n_y) \quad (6)$$

$$x = j \% n_x \quad (7)$$

$$y = j / n_x \quad (8)$$

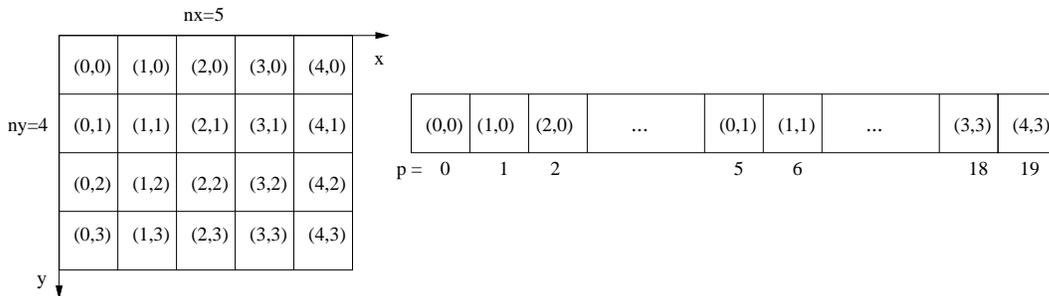


Figura 1: Representação matricial e vetorial de uma imagem 2D.

2 Representação em disco

No disco, as imagens são sempre armazenadas na forma vetorial com um cabeçalho cujos campos indicam informações sobre a imagem e a forma como está armazenada.

Por exemplo, imagens cinza de até $b = 8$ bits com formato PGM, do tipo P5, são armazenadas em disco da seguinte forma:

```
P5
nx ny
255
vetor de bytes com 1 byte por pixel
```

As três primeiras linhas são armazenadas em ASCII e o vetor de pixels em binário.

Outro exemplo é uma imagem colorida de $b = 24$ bits com formato PGM, do tipo P6.

```
P6
nx ny
255
vetor de bytes com 1 byte para cada banda de cada pixel.
```

(Para mais exemplos sobre formatos de imagem, veja libmc920 na página do curso.)

3 Histograma

O histograma de uma imagem cinza $\hat{I} = (D_I, I)$ 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. O histograma normalizado $h(l)/|D_I|$ representa a distribuição de probabilidade dos valores dos spels (Figura 2).

Imagens claras (escuras) possuem histogramas com altas concentrações de pixels de alto (baixo) brilho. O contraste maior está associado a um grau maior de dispersão do histograma.

Todas as operações com imagens podem ser escritas em forma de algoritmos. No caso do histograma temos.

Entrada: Imagem cinza $\hat{I} = (D_I, I)$.

Saída: Vetor com o histograma h .

Auxiliar: Variável l .

1. Para $l = 0$ até $2^b - 1$ faça
2. $h(l) \leftarrow 0$.
3. Para todo $p \in D_I$ faça
4. $h(I(p)) \leftarrow h(I(p)) + 1$.

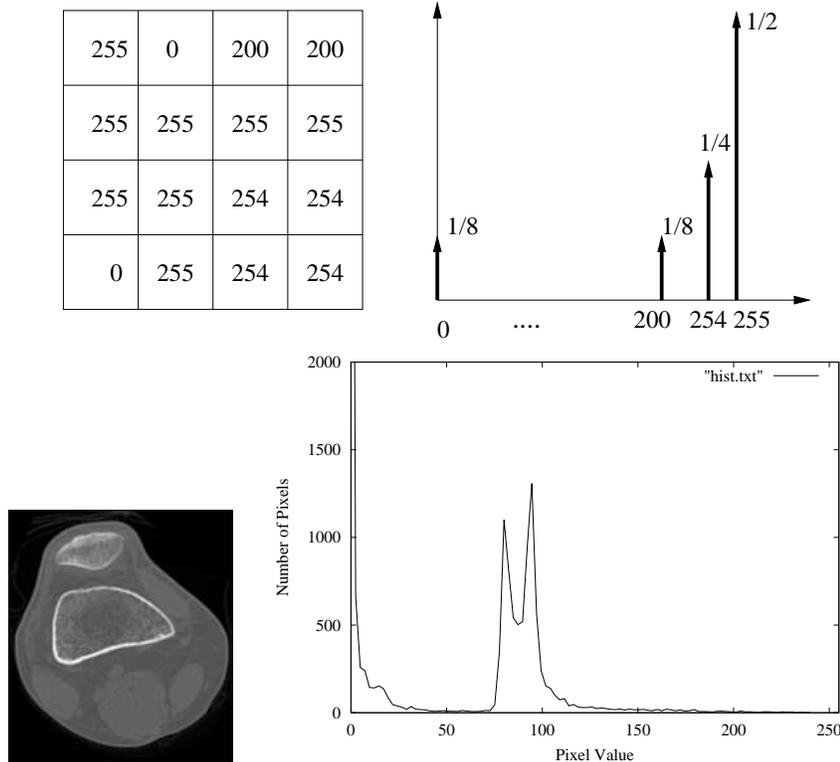


Figura 2: (acima) Exemplo de histograma de imagem cinza com $|D_I| = 4 \times 4$ e $b = 8$. Imagem clara e com pouco contraste. (abaixo) Histograma da imagem de tomografia de raios-X de um joelho.

No caso de imagens multiespectrais, cada banda é requantizada em um certo número de intervalos, de forma que o espaço de características Z^k é dividido em hipercubos (bins do histograma).

Por exemplo, a partir de uma imagem colorida $\hat{I} = (D_I, \vec{I})$, $\vec{I} = (I_1, I_2, I_3)$, 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 $[192, 255]$ (Figura 3). A contagem de cores em cada bin é usada no cálculo do histograma.

4 Operações matemáticas

Operações matemáticas são realizadas entre imagens cinza (binária) com um mesmo domínio. Elas equivalem às mesmas operações realizadas spel a spel, entre os valores dos spels.

Por exemplo, a raiz quadrada da soma de duas imagens $\hat{A} = (D_A, A)$ e $\hat{B} = (D_B, B)$, onde $D_A = D_B$, multiplicada por 2, gera uma imagem $\hat{C} = (D_C, C)$, onde $D_C = D_A = D_B$ e $C(p) = 2\sqrt{A(p) + B(p)}$ para todo $p \in D_C$.

As operações lógicas, and e or, entre \hat{A} e \hat{B} , por exemplo, geram \hat{C} e \hat{D} respectivamente, onde $C(p) = \min\{A(p), B(p)\}$, $\forall p \in D_C$, e $D(p) = \max\{A(p), B(p)\}$, $\forall p \in D_D$.

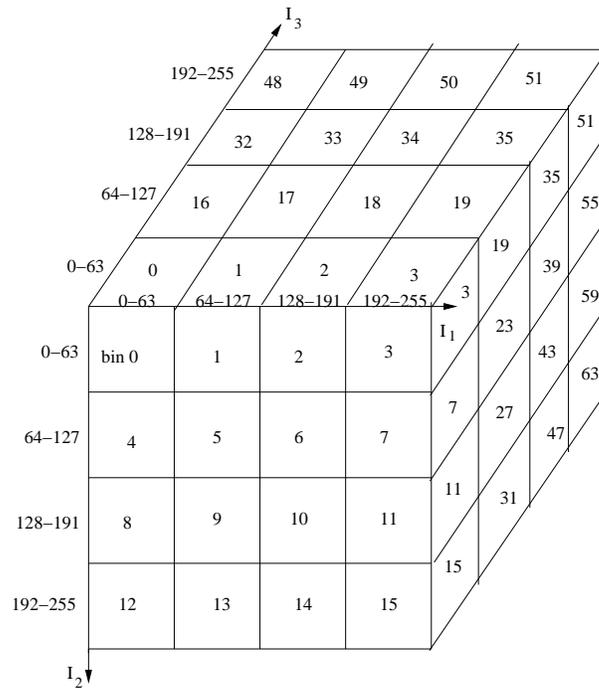


Figura 3: Exemplo de divisão do espaço paramétrico para cálculo de histograma de imagem colorida com 64 bins.

5 Exercícios

1. Escreva um algoritmo para calcular o histograma de cor com N bins para uma imagem colorida. Como ficam os intervalos de valor em cada eixo quando $N = 512$ bins.
2. Escreva um algoritmo para calcular o histograma de uma imagem com k bandas. O número de bins é dado como parâmetro.
3. Crie um exemplo de imagem escura e construa o seu histograma graficamente.
4. Escreva os algoritmos que implementam as operações matemáticas acima.